GitHub

Getting Started with Ingesting GitHub GHAS Alerts

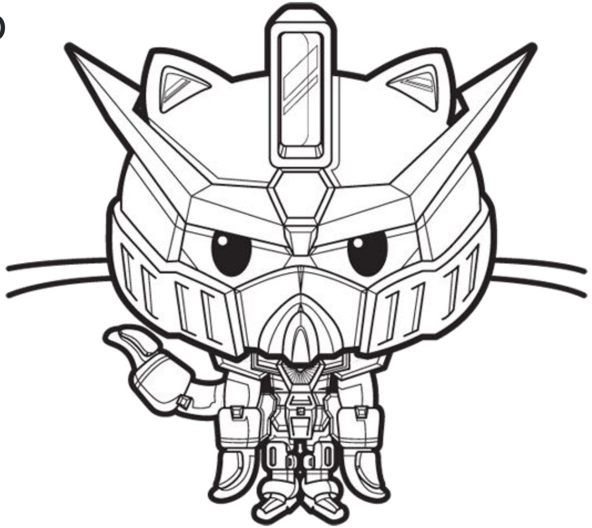
GitHub Partner Engineering

How to use this guide

This deck is meant as a starting point for ingesting GitHub Advanced Security (GHAS) Alerts. These alerts can be fed into 3rd party solutions

- Logging
- Observability
- Security Information and Event Management (SIEM)
- Business Intelligence (BI)

It contains links to documentation and sample code. The code samples leverage octokit.js.



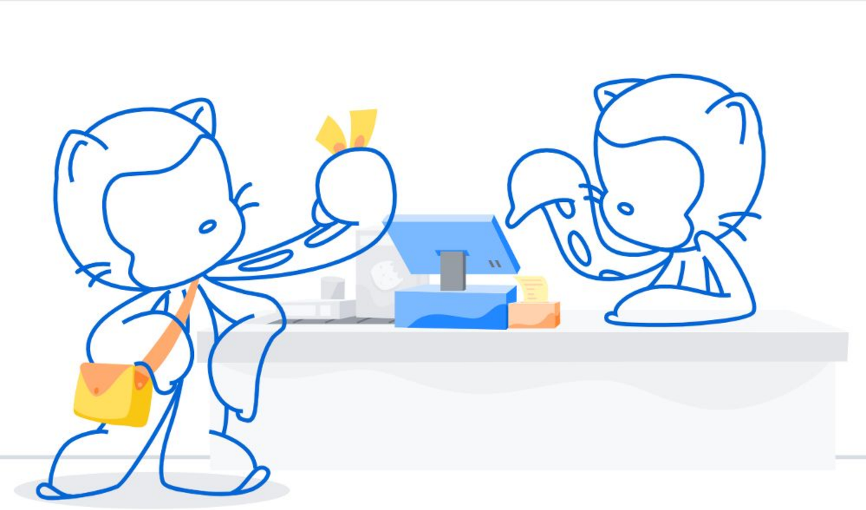
Topics

- 👋 GitHub Advanced Security platform overview
- 🤖 Polling with GitHub REST API
- Webhooks
- 🚀 Summary

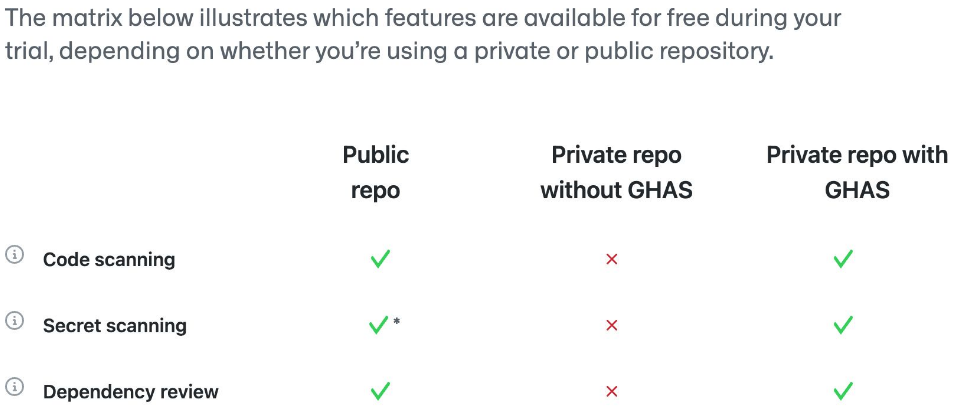
👋 GitHub platform overview

3 Types of GitHub GHAS Alerts

- Code Scanning Alerts
- Secret Scanning Alerts
- Dependabot Alerts



How to try GitHub Advanced Security



The matrix below illustrates which features are available for free during your trial, depending on whether you're using a private or public repository.

	Public repo	Private repo without GHAS	Private repo with GHAS
ⓘ Code scanning	✓	×	✓
ⓘ Secret scanning	✓ *	×	✓
ⓘ Dependency review	✓	×	✓

Ingestion Strategy Comparison

Polling

Pros

- Returns rich data set
- Has historical alerts

Cons

- Subject to rate limits
- Requires a dedicated host
- May require data sanitation (secret scanning alerts contain secrets)

Webhooks

Pros

- Push based
- Real time, event-driven

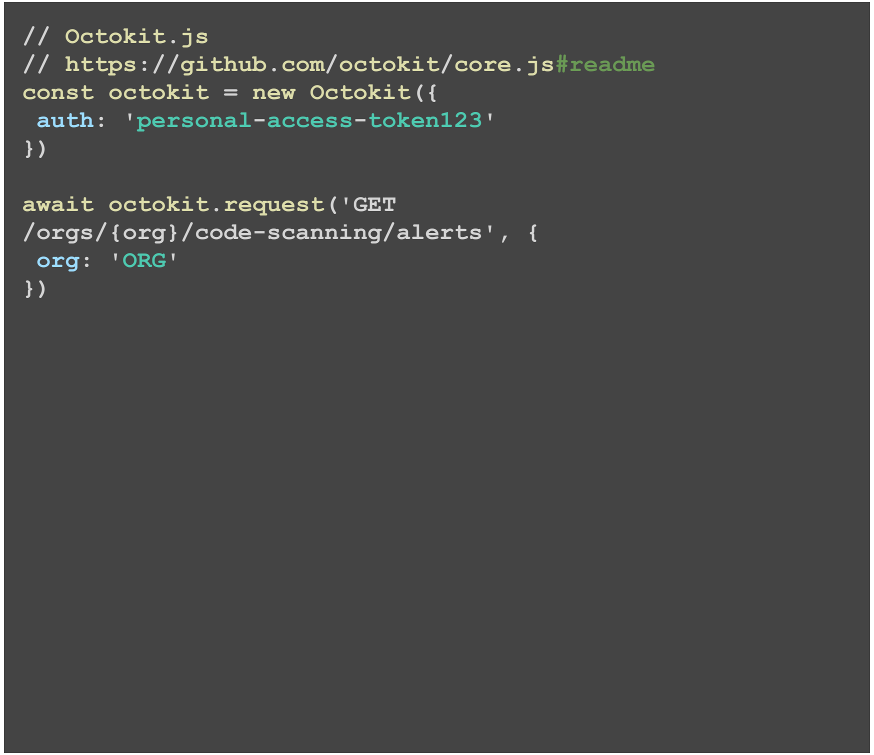
Cons

- Requires HTTP write endpoint
- No retry mechanism
- Returns only summaries
- No history, only new events

🤖 Polling with GitHub API’s

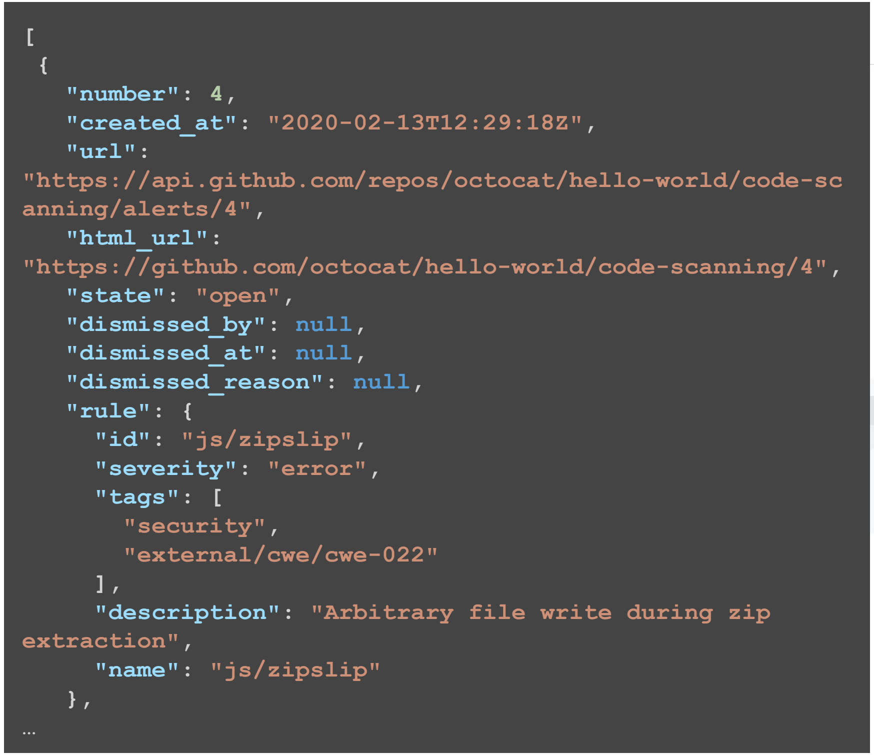
Code Scanning REST API docs example

Code Snippet



```
// Octokit.js
// https://github.com/octokit/core.js#readme
const octokit = new Octokit({
 auth: 'personal-access-token123'
})

await octokit.request('GET
/orgs/{org}/code-scanning/alerts', {
 org: 'ORG'
})
```

Example 200 Response



```
[
 {
   "number": 4,
   "created_at": "2020-02-13T12:29:18Z",
   "url":
"https://api.github.com/repos/octocat/hello-world/code-sc
anning/alerts/4",
   "html_url":
"https://github.com/octocat/hello-world/code-scanning/4",
   "state": "open",
   "dismissed_by": null,
   "dismissed_at": null,
   "dismissed_reason": null,
   "rule": {
     "id": "js/zipslip",
     "severity": "error",
     "tags": [
       "security",
       "external/cwe/cwe-022"
     ],
     "description": "Arbitrary file write during zip
extraction",
     "name": "js/zipslip"
   },
…
```

Secret Scanning REST API docs example

Code Snippet

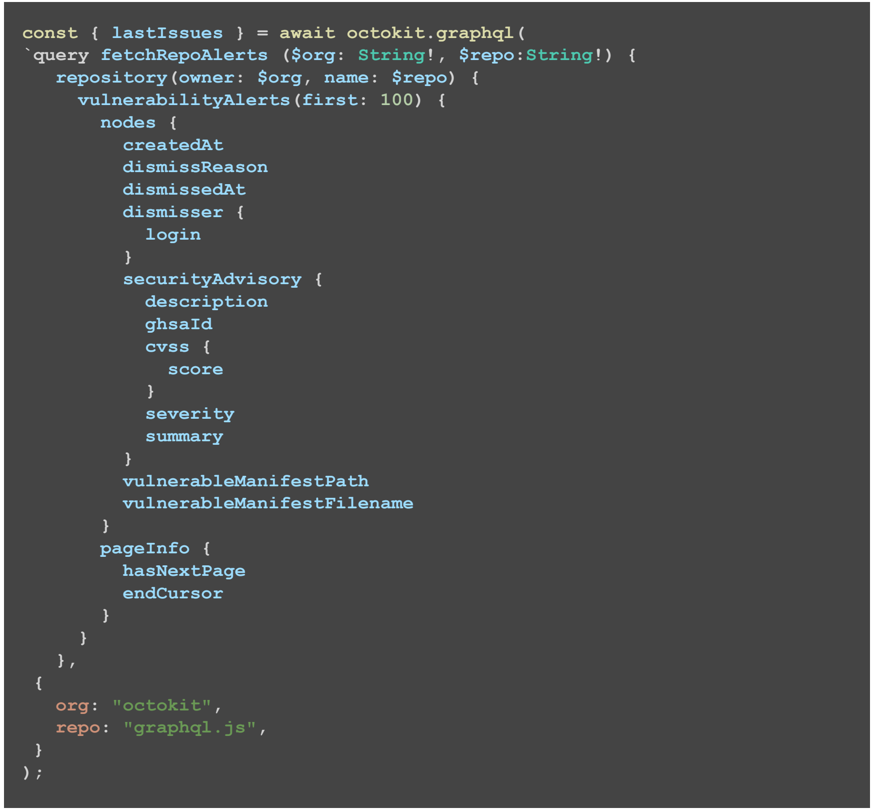
```
// Octokit.js
// https://github.com/octokit/core.js#readme
const octokit = new Octokit({
 auth: 'personal-access-token123'
})

await octokit.request('GET
/repos/{owner}/{repo}/secret-scanning/alerts', {
 owner: 'OWNER',
 repo: 'REPO'
})
```

Example 200 Response

```
[
 {
   "number": 2,
   "created_at": "2020-11-06T18:48:51Z",
   "url":
"https://api.github.com/repos/owner/private-repo/secret-s
canning/alerts/2",
   "html_url":
"https://github.com/owner/private-repo/security/secret-sc
anning/2",
   "locations_url":
"https://api.github.com/repos/owner/private-repo/secret-s
canning/alerts/2/locations",
   "state": "resolved",
   "resolution": "false_positive",
   "resolved_at": "2020-11-07T02:47:13Z",
   "resolved_by": {
     "login": "monalisa",
     "id": 2,
     "node_id": "MDQ6VXNlcjI=",
     "avatar_url":
"https://alambic.github.com/avatars/u/2?",
     "gravatar_id": "",
     "url": "https://api.github.com/users/monalisa",
…
```

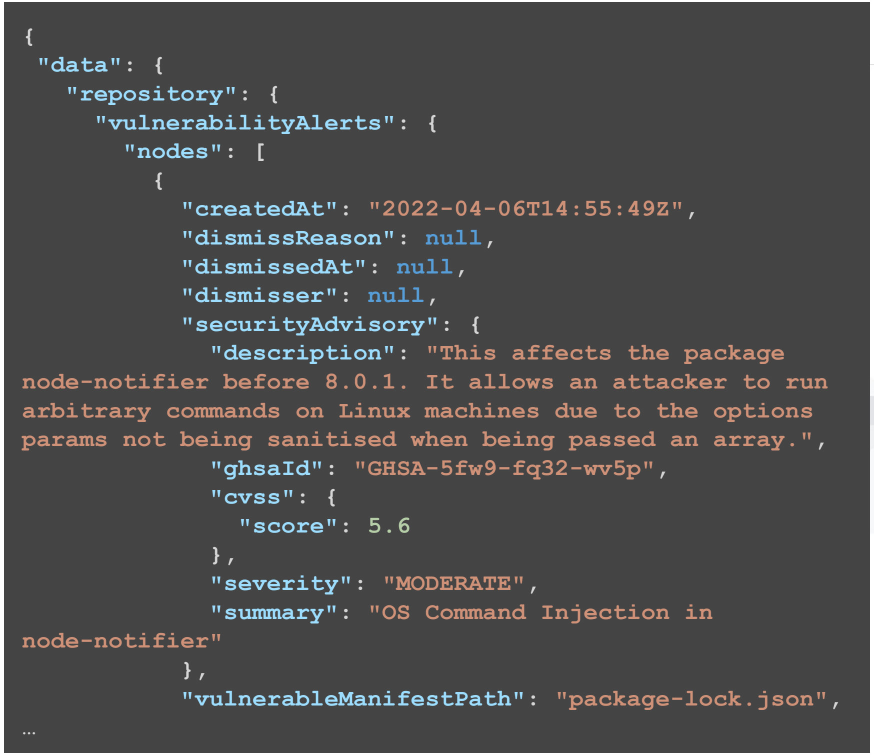
Dependabot GraphQL API docs example

Sample Query



```
const { lastIssues } = await octokit.graphql(
`query fetchRepoAlerts ($org: String!, $repo:String!) {
   repository(owner: $org, name: $repo) {
     vulnerabilityAlerts(first: 100) {
       nodes {
         createdAt
         dismissReason
         dismissedAt
         dismisser {
           login
         }
         securityAdvisory {
           description
           ghsaId
           cvss {
             score
           }
           severity
           summary
         }
         vulnerableManifestPath
         vulnerableManifestFilename
       }
       pageInfo {
         hasNextPage
         endCursor
       }
     }
   },
 {
   org: "octokit",
   repo: "graphql.js",
 }
);
```

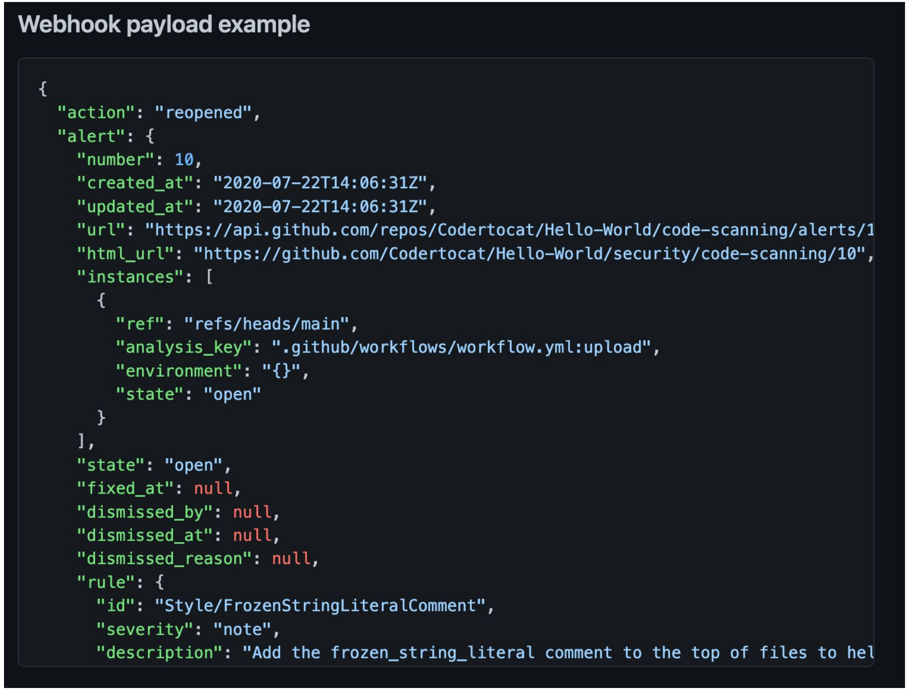
Example 200 Response



```
{
 "data": {
   "repository": {
     "vulnerabilityAlerts": {
       "nodes": [
         {
           "createdAt": "2022-04-06T14:55:49Z",
           "dismissReason": null,
           "dismissedAt": null,
           "dismisser": null,
           "securityAdvisory": {
             "description": "This affects the package
node-notifier before 8.0.1. It allows an attacker to run
arbitrary commands on Linux machines due to the options
params not being sanitised when being passed an array.",
             "ghsaId": "GHSA-5fw9-fq32-wv5p",
             "cvss": {
               "score": 5.6
             },
             "severity": "MODERATE",
             "summary": "OS Command Injection in
node-notifier"
           },
           "vulnerableManifestPath": "package-lock.json",
…
```

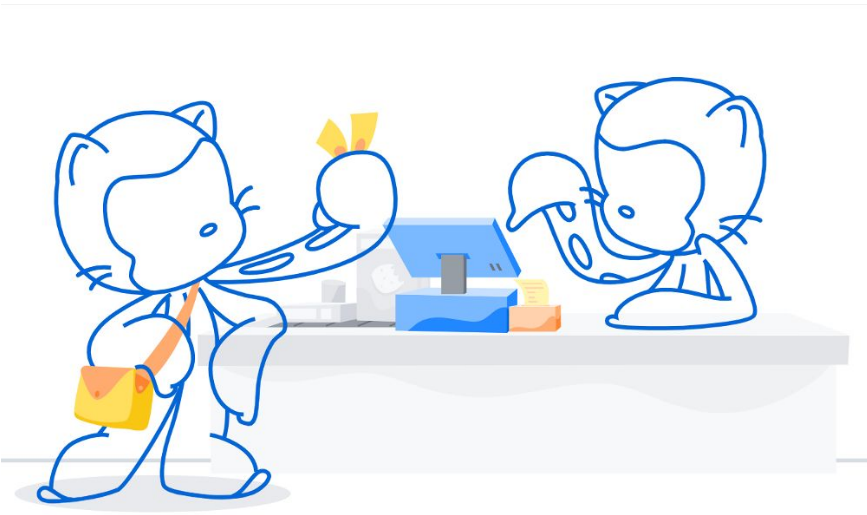
🤖 Webhook Events & Payloads

code_scanning_alert docs

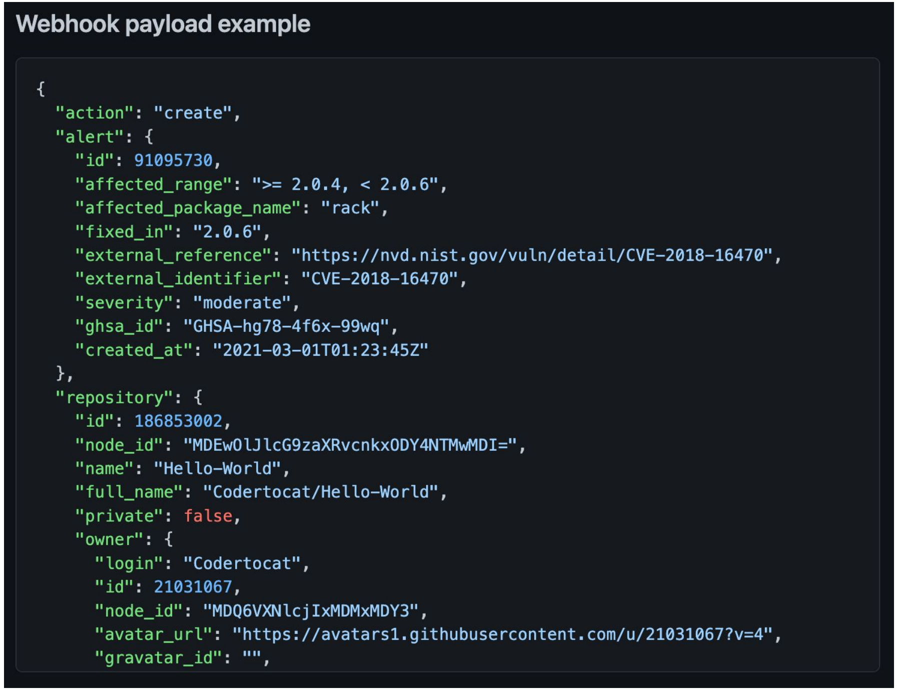


Webhook payload example

```
{
  "action": "reopened",
  "alert": {
    "number": 10,
    "created_at": "2020-07-22T14:06:31Z",
    "updated_at": "2020-07-22T14:06:31Z",
    "url": "https://api.github.com/repos/Codertocat/Hello-World/code-scanning/alerts/1
    "html_url": "https://github.com/Codertocat/Hello-World/security/code-scanning/10",
    "instances": [
      {
        "ref": "refs/heads/main",
        "analysis_key": ".github/workflows/workflow.yml:upload",
        "environment": "{}",
        "state": "open"
      }
    ],
    "state": "open",
    "fixed_at": null,
    "dismissed_by": null,
    "dismissed_at": null,
    "dismissed_reason": null,
    "rule": {
      "id": "Style/FrozenStringLiteralComment",
      "severity": "note",
      "description": "Add the frozen_string_literal comment to the top of files to hel
```



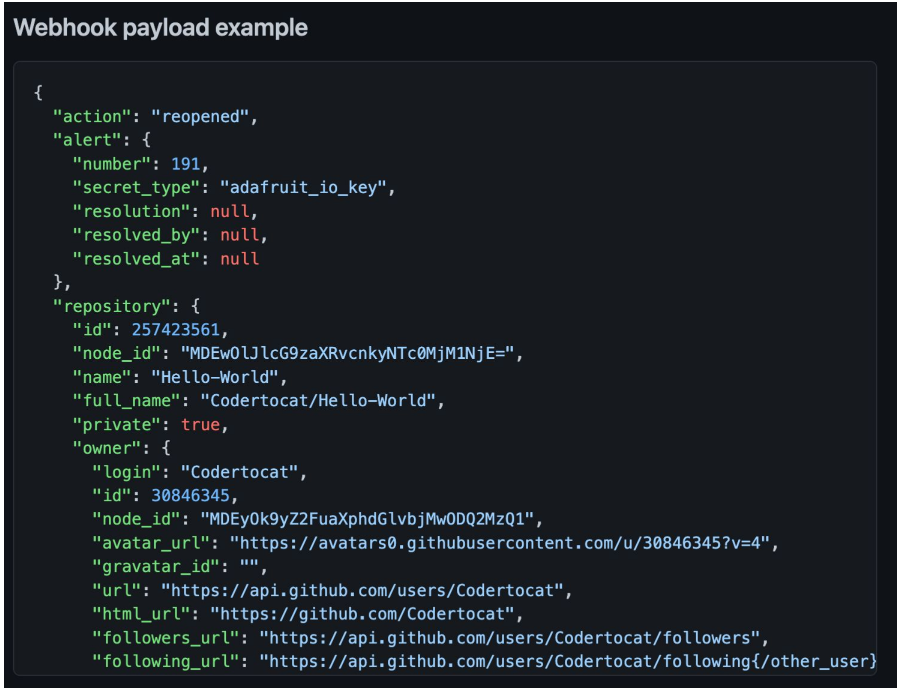
repository_vulnerability_alert docs



Webhook payload example

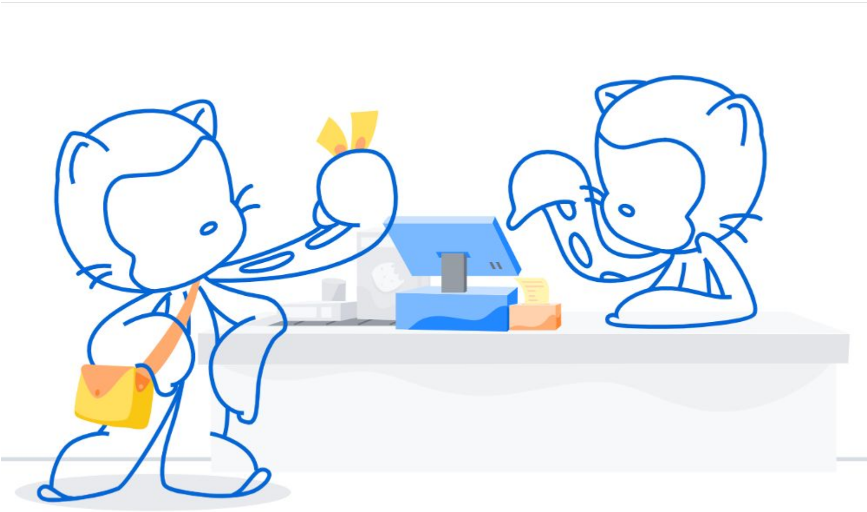
```
{
  "action": "create",
  "alert": {
    "id": 91095730,
    "affected_range": ">= 2.0.4, < 2.0.6",
    "affected_package_name": "rack",
    "fixed_in": "2.0.6",
    "external_reference": "https://nvd.nist.gov/vuln/detail/CVE-2018-16470",
    "external_identifier": "CVE-2018-16470",
    "severity": "moderate",
    "ghsa_id": "GHSA-hg78-4f6x-99wq",
    "created_at": "2021-03-01T01:23:45Z"
  },
  "repository": {
    "id": 186853002,
    "node_id": "MDEwOlJlcG9zaXRvcnkxODY4NTMwMDI=",
    "name": "Hello-World",
    "full_name": "Codertocat/Hello-World",
    "private": false,
    "owner": {
      "login": "Codertocat",
      "id": 21031067,
      "node_id": "MDQ6VXNlcjIxMDMxMDY3",
      "avatar_url": "https://avatars1.githubusercontent.com/u/21031067?v=4",
      "gravatar_id": "",
```

secret_scanning_alert docs



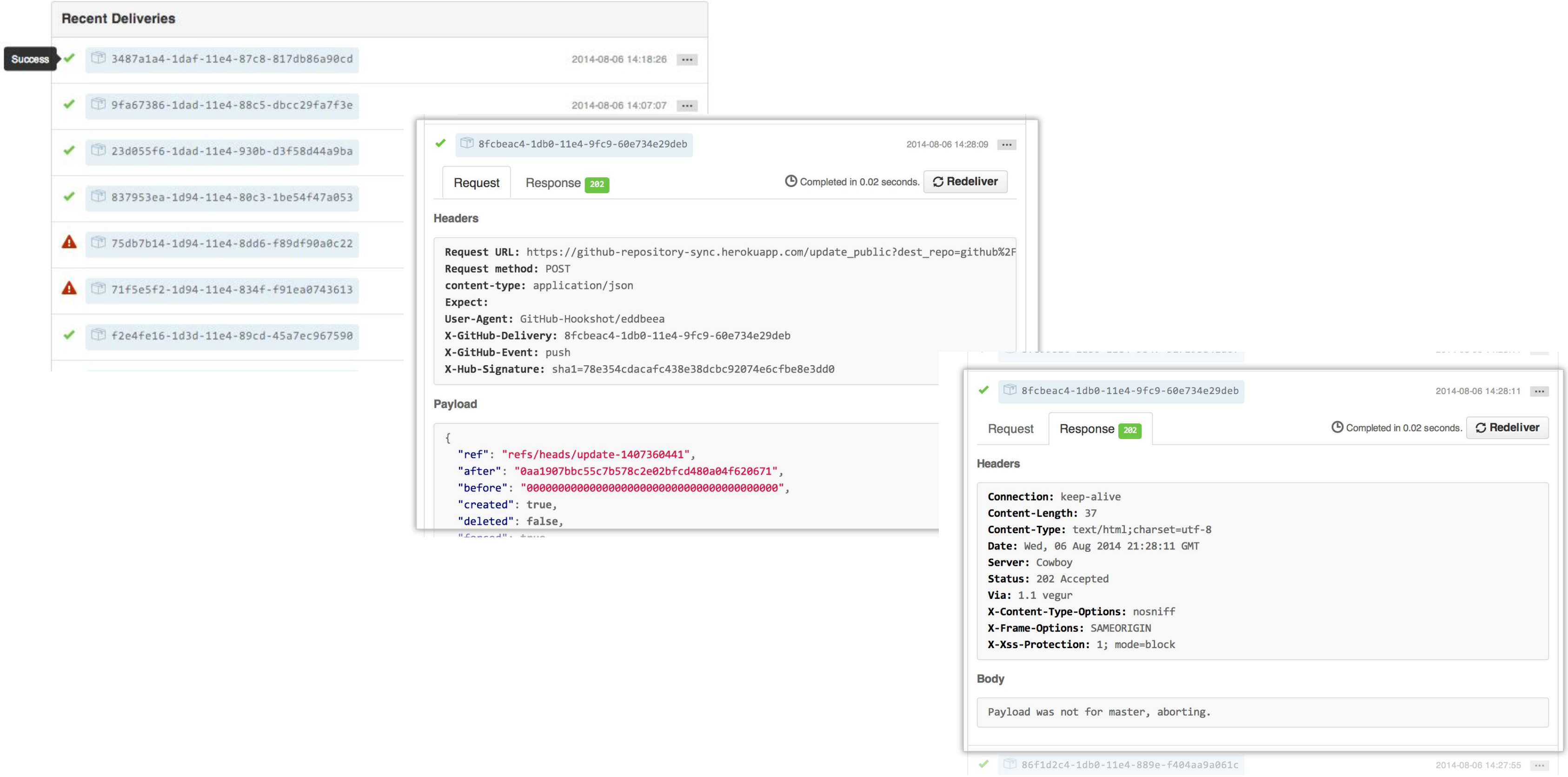
Webhook payload example

```
{
  "action": "reopened",
  "alert": {
    "number": 191,
    "secret_type": "adafruit_io_key",
    "resolution": null,
    "resolved_by": null,
    "resolved_at": null
  },
  "repository": {
    "id": 257423561,
    "node_id": "MDEwOlJlcG9zaXRvcnkyNTc0MjM1NjE=",
    "name": "Hello-World",
    "full_name": "Codertocat/Hello-World",
    "private": true,
    "owner": {
      "login": "Codertocat",
      "id": 30846345,
      "node_id": "MDEyOk9yZ2FuaXphdGlvbjMwODQ2MzQ1",
      "avatar_url": "https://avatars0.githubusercontent.com/u/30846345?v=4",
      "gravatar_id": "",
      "url": "https://api.github.com/users/Codertocat",
      "html_url": "https://github.com/Codertocat",
      "followers_url": "https://api.github.com/users/Codertocat/followers",
      "following_url": "https://api.github.com/users/Codertocat/following{/other_user}
```



Testing Webhooks

GitHub keeps a log of each webhook delivery for 30 days.



Recent Deliveries

Success ✓ 3487a1a4-1daf-11e4-87c8-817db86a90cd 2014-08-06 14:18:26 ···

✓ 9fa67386-1dad-11e4-88c5-dbcc29fa7f3e 2014-08-06 14:07:07 ···

✓ 23d055f6-1dad-11e4-930b-d3f58d44a9ba

✓ 837953ea-1d94-11e4-80c3-1be54f47a053

⚠ 75db7b14-1d94-11e4-8dd6-f89df90a0c22

⚠ 71f5e5f2-1d94-11e4-834f-f91ea0743613

✓ f2e4fe16-1d3d-11e4-89cd-45a7ec967590

✓ 8fcbeac4-1db0-11e4-9fc9-60e734e29deb 2014-08-06 14:28:09 ···

Request | Response 202 | Completed in 0.02 seconds. | Redeliver

Headers

```
Request URL: https://github-repository-sync.herokuapp.com/update_public?dest_repo=github%2F
Request method: POST
content-type: application/json
Expect:
User-Agent: GitHub-Hookshot/eddbeea
X-GitHub-Delivery: 8fcbeac4-1db0-11e4-9fc9-60e734e29deb
X-GitHub-Event: push
X-Hub-Signature: sha1=78e354cdacafc438e38dcbc92074e6cfbe8e3dd0
```

Payload

```
{
  "ref": "refs/heads/update-1407360441",
  "after": "0aa1907bbc55c7b578c2e02bfcd480a04f620671",
  "before": "0000000000000000000000000000000000000000",
  "created": true,
  "deleted": false,
  "forced": true
```

✓ 8fcbeac4-1db0-11e4-9fc9-60e734e29deb 2014-08-06 14:28:11 ···

Request | Response 202 | Completed in 0.02 seconds. | Redeliver

Headers

```
Connection: keep-alive
Content-Length: 37
Content-Type: text/html;charset=utf-8
Date: Wed, 06 Aug 2014 21:28:11 GMT
Server: Cowboy
Status: 202 Accepted
Via: 1.1 vegur
X-Content-Type-Options: nosniff
X-Frame-Options: SAMEORIGIN
X-Xss-Protection: 1; mode=block
```

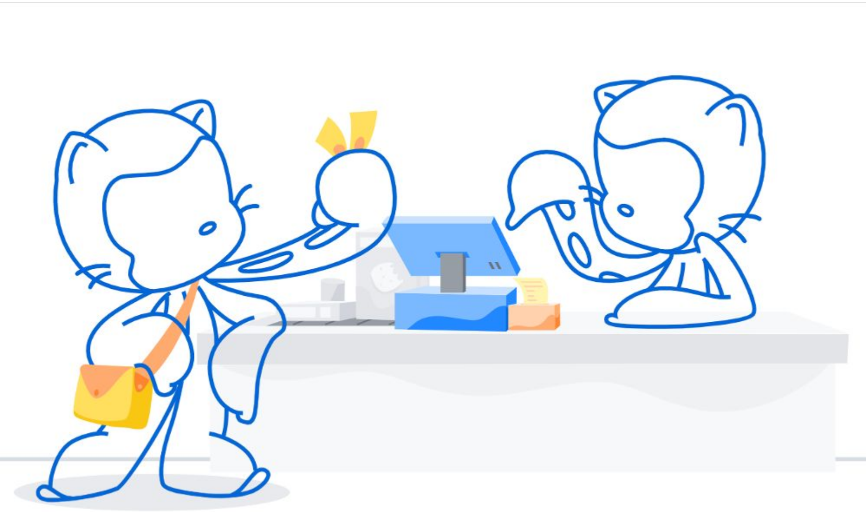
Body

```
Payload was not for master, aborting.
```

✓ 86f1d2c4-1db0-11e4-889e-f404aa9a061c 2014-08-06 14:27:55 ···

Best Practices

- Use Webhooks in conjunction with the REST API's to get the full picture
- Create a GitHub App for higher rate limits.



🤖 Appendix: Authentication

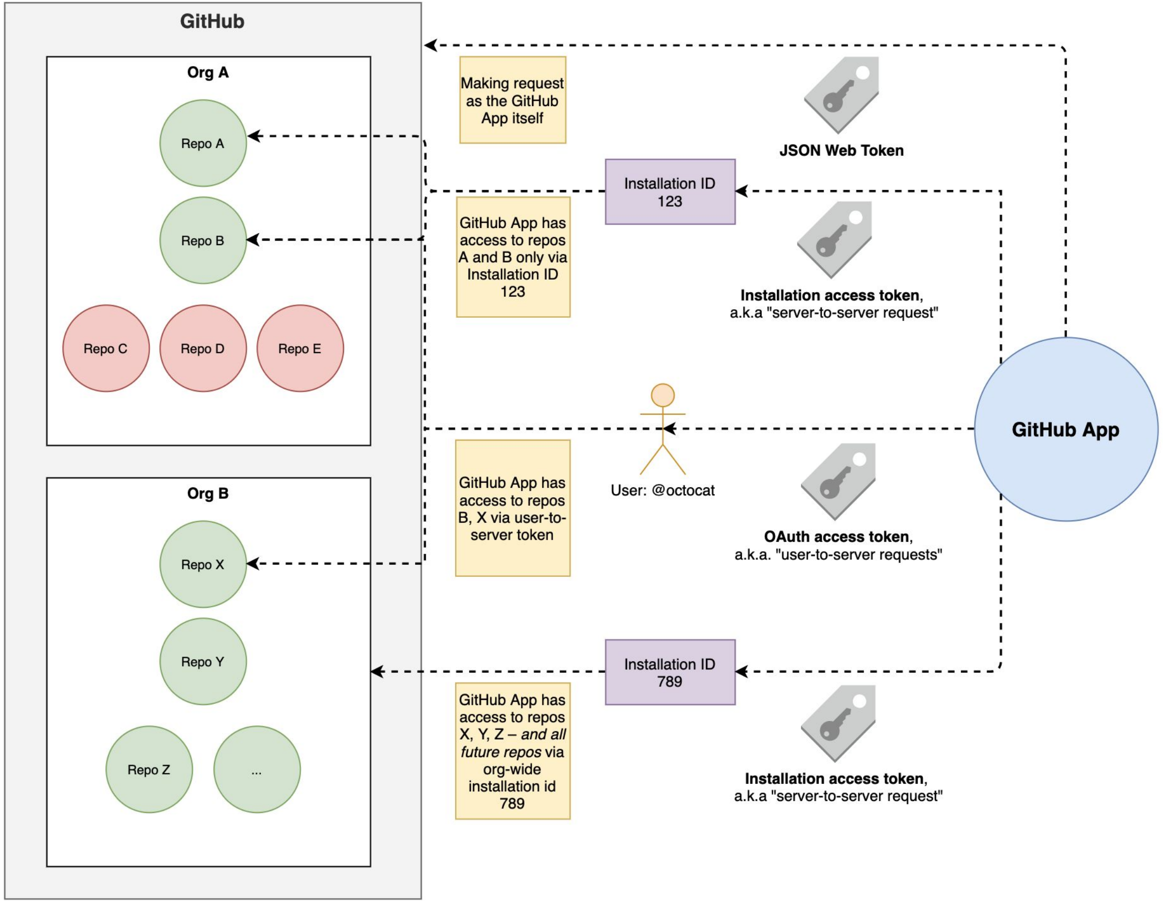
Authentication overview

Authentication Scheme	Also Known As	Description	How to Get It	Available Endpoints	Examples
JSON Web Token	JWT (pronounced "jot")	Authenticates as the GitHub App	GitHub docs, Octokit	List	Fetching application installation details or exchanging the **JWT** for an **installation access token**.
Installation access token	Server-to-server requests	Authenticates as a specific installation of the GitHub App	GitHub docs, Octokit	List	Opening an issue or providing feedback on a pull request
OAuth access token	User-to-server requests	Authenticates as a user of the GitHub App	GitHub docs	List	Authenticating as a user when a GitHub App needs to verify a user's identity or act on a user's behalf
Personal Access Token	PAT	Authenticates as a user	GitHub docs		PATs are an alternative to using passwords for authentication to GitHub

Authentication at a glance

Deciding which authentication type to use comes down to:

- What resource do I need to access?
- Who do I need to access it as?


GitHub
Org A
Repo A
Repo B
Repo C
Repo D
Repo E
Org B
Repo X
Repo Y
Repo Z
...
Making request as the GitHub App itself
JSON Web Token
Installation ID 123
GitHub App has access to repos A and B only via Installation ID 123
Installation access token, a.k.a "server-to-server request"
User: @octocat
GitHub App
GitHub App has access to repos B, X via user-to-server token
OAuth access token, a.k.a. "user-to-server requests"
Installation ID 789
GitHub App has access to repos X, Y, Z – and all future repos via org-wide installation id 789
Installation access token, a.k.a "server-to-server request"

Server-to-server requests

Server-to-server requests are those made from the perspective of an *installation* and are authenticated by **installation access tokens.**

Using your **JWT**, generate an **installation access token** via:

```
curl -i -X POST \
      -H "Authorization: Bearer YOUR_JWT" \
      -H "Accept: application/vnd.github.machine-man-preview+json" \
      https://api.github.com/app/installations/:installation_id/access_tokens
```

As a security measure, these tokens expire after 1 hour. They can be used like:

```
curl -i \
      -H "Authorization: token YOUR INSTALLATION ACCESS TOKEN" \
      -H "Accept: application/vnd.github.machine-man-preview+json" \
      https://api.github.com/installation/repositories
```

User-to-server requests

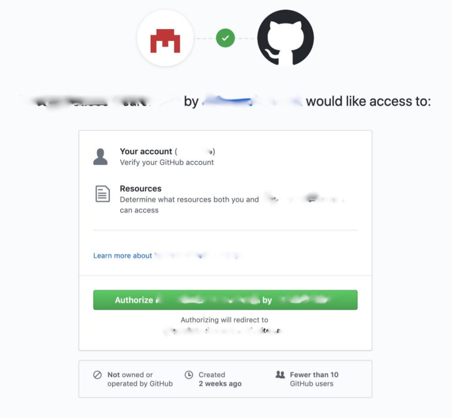
User-to-server requests act as a *user who has authorized your GitHub App* and are authenticated using an **OAuth access token.**

First, users authorize your GitHub App via OAuth and receive a `code`:

Then, your GitHub App trades the `code`, `client_id` and `client_secret` for an **OAuth access token** to be used like:

```
curl -H "Authorization: token OAUTH-TOKEN" https://api.github.com/user
```

Unlike typical OAuth, the scope is determined by the GitHub App.



[redacted] by [redacted] would like access to:

Your account ([redacted])
Verify your GitHub account

Resources
Determine what resources both you and [redacted] can access

Learn more about [redacted]

Authorize [redacted] **by** [redacted]

Authorizing will redirect to
[redacted]

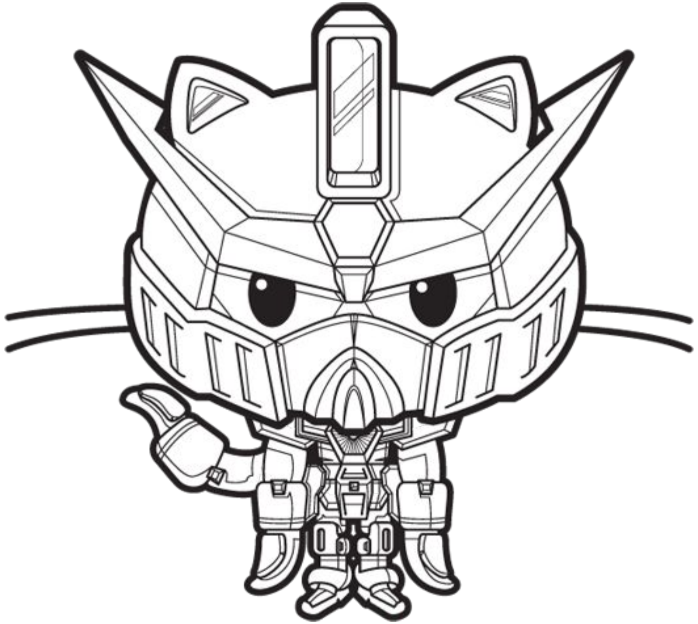
Not owned or operated by GitHub

Created **2 weeks ago**

Fewer than 10 GitHub users

Additional Resources

- Developer Documentation
- GitHub REST and GraphQL APIs
- GitHub Webhooks
- Octokit and 3rd party libraries
- smee.io Tool for testing Webhooks
- Platform Samples repo
- GitHub Advanced Security Workshop
- Webhook handler samples
  - github-webhook-handler node.js
  - python-github-webhooks python
  - github_webhook ruby
  - hookserve go
  - afterparty rust
  - Github-webhook-lambda (AWS lambda)
  - GitHub-Webhook-Function (Azure Function)
  - github-webhook-cloud-function (Google Cloud Functions)



🚀 Summary

- Polling the API is great for getting rich data sets
- Webhooks are great for getting alerts as soon as they happen