Dual Stack and IPv6-only Amazon VPC Reference Architectures

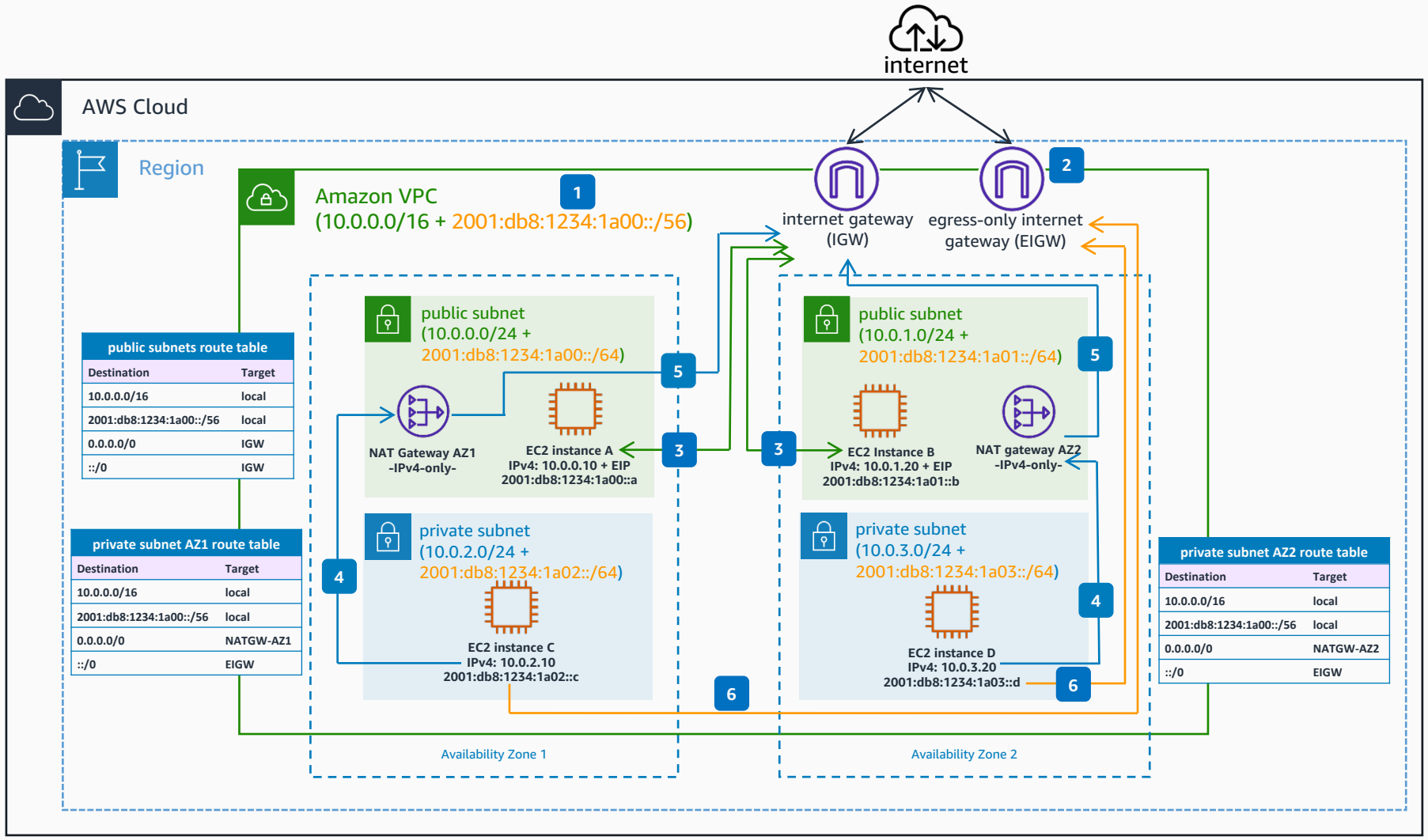
1. *Dual stack Amazon VPC internet connectivity*
2. *IPv6-only subnets in a dual stack Amazon VPC*
3. *Internet connectivity for IPv6-only subnets in a dual stack Amazon VPC*
4. *IPv4 targets for dual stack internet-facing Application Load Balancer*
5. *IPv6 targets for dual stack internet-facing Application Load Balancer*
6. *IPv4 targets for dual stack internet-facing Network Load Balancer*
7. *IPv6 targets for dual stack internet-facing Network Load Balancer*
8. *Dual stack internal Application and Network Load Balancers*
9. *DNS64*
10. *NAT64*
11. *Centralized egress traffic with NAT64* NEW
12. *Dual stack peering connectivity for the Amazon VPC*
13. *Dual stack VPC connectivity with AWS Transit Gateway*
14. *Dual stack VPC connectivity with AWS PrivateLink* NEW
15. *Dual stack hybrid connectivity with AWS Direct Connect*
16. *Dual stack VPN connectivity with AWS Transit Gateway*
17. *Dual stack AWS Transit Gateway Connect*

AWS Reference Architecture

Reviewed for technical accuracy June 23, 2022

1. Dual stack Amazon VPC internet connectivity

Enable IPv4 and IPv6 internet connectivity for your Amazon Virtual Private Cloud (VPC).



1 Associate an IPv6 Classless Inter-Domain Routing (CIDR) block to your **Amazon Virtual Private Cloud (Amazon VPC)**. This can be an AWS-assigned CIDR, or part of a Bring Your Own IPv6 Addresses (BYOIPv6) pool.

2 Associate an egress-only internet gateway (EIGW) to the VPC. This is the target for the IPv6 default route of private dual stack subnets.

3 Compute resources in public dual stack subnets use the internet gateway for dual-stack IPv4 and IPv6 internet connectivity. They can directly initiate outbound internet connections and accept inbound internet connections, to and from IPv4 and IPv6 hosts in the internet, using their associated Elastic IPv4 address or IPv6 addresses from the subnet CIDR. Note that security groups must allow both IPv4 and IPv6 traffic.

4 Resources in private dual stack subnets use the public NAT gateway in each Availability Zone for outbound IPv4 Internet connectivity. The NAT gateway allows only outbound IPv4 connections to be opened from private **Amazon Elastic Compute Cloud (Amazon EC2) instances** to internet IPv4 destinations, and the associated return traffic.

5 The NAT gateways send the translated IPv4 packets to the internet gateway, which sends the traffic out in the internet, to the respective IPv4 destinations.

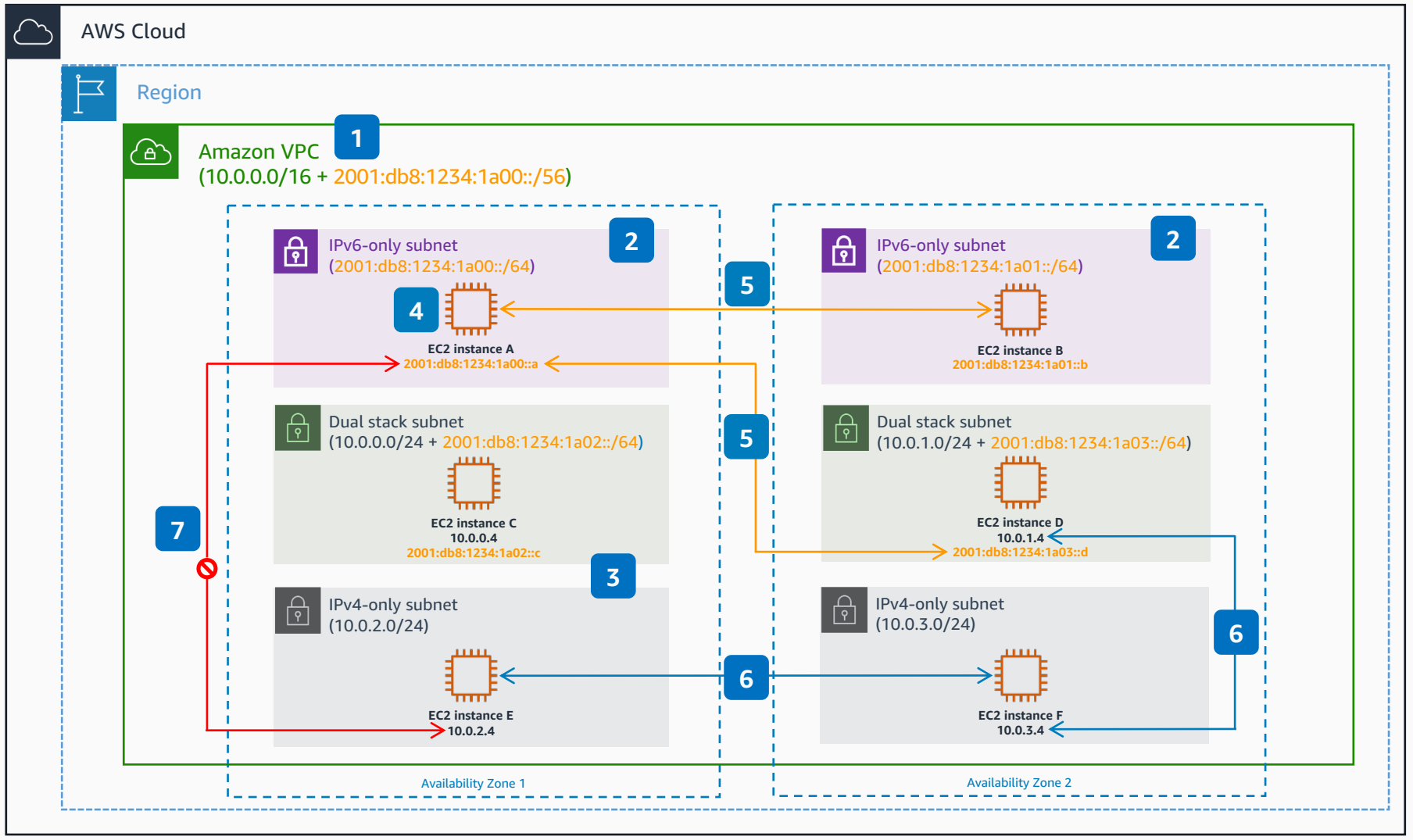
6 Resources in private dual stack subnets use the egress-only internet gateway for outbound IPv6 internet connectivity. The egress-only internet gateway allows only outbound IPv6 connections to be opened from private **EC2 instances** to internet IPv6 destinations, and the associated return traffic.

Reviewed for technical accuracy June 23, 2022

AWS Reference Architecture

2. IPv6-only subnets in a dual stack Amazon VPC

Integrate IPv6-only subnets in your dual stack VPC.



1. To create IPv6-only subnets, you start with a dual stack **Amazon VPC (Virtual Private Cloud)**. The VPC needs to have a secondary IPv6 CIDR associated with it.

2. IPv6-only subnets only have an IPv6 CIDRs associated with them, and do not need associated IPv4 CIDRs. Also, IPv6-only subnets are bound to an Availability Zone the same as dual stack or IPv4-only subnets are.

3. IPv6-only subnets can coexist in the same VPC as IPv4-only subnets and dual stack subnets. You can also choose to have dual stack VPCs with IPv6-only subnets, but keep in mind that the primary VPC CIDR is an IPv4 CIDR.

4. When you create an **EC2** instance in the IPv6-only subnet, the IPv6 address is either automatically assigned from the subnet CIDR through DHCPv6, or you can manually configure it.

5. IPv6-only resources in IPv6-only subnets can communicate natively over IPv6 with other IPv6-only resources or dual stack resources in other subnets in the VPC.

6. IPv4-only resources in the IPv4-only subnets can communicate natively over IPv4 with other IPv4-only resources or dual stack resources in other subnets in the VPC.

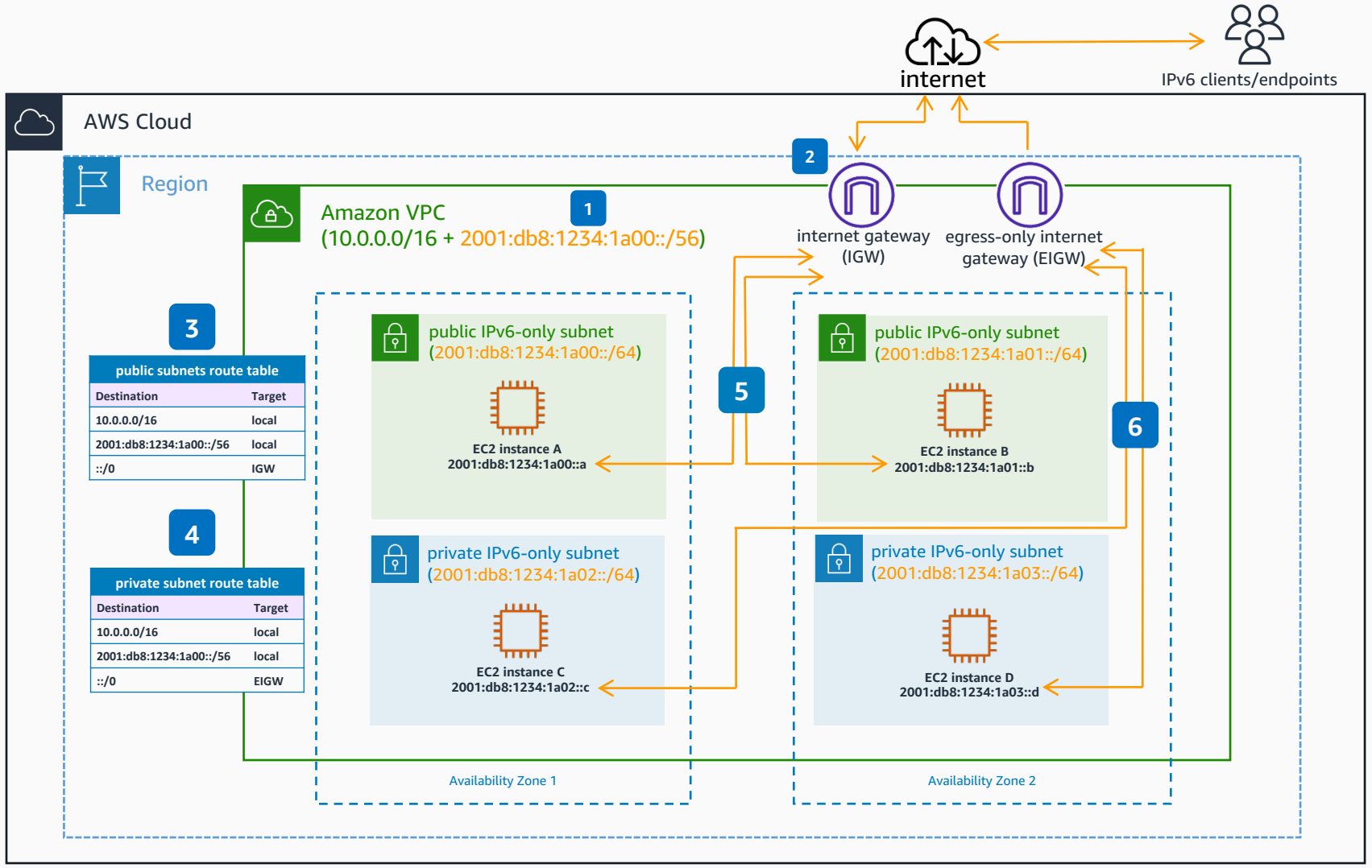
7. IPv6-only resources in the IPv6-only subnets cannot communicate with IPv4-only resources in other subnets in the VPC.

Reviewed for technical accuracy June 23, 2022

AWS Reference Architecture

3. Internet connectivity for IPv6-only subnets in a dual stack VPC

Enable IPv6 Internet connectivity for your IPv6-only subnets in your VPC.



1. Starting from your dual stack **Amazon VPC**, the primary CIDR is an IPv4 one, and the secondary IPv6 CIDR is used to create the IPv6-only subnets.

2. The internet gateway (IGW) and the egress-only internet gateway (EIGW) are attached to the dual stack VPC. Although the IPv6 addresses are Global Unicast Addresses (GUA), you can still create public and private subnets, controlling the IGW and EIGW routing and security group configuration.

3. The public IPv6-only subnets route tables have the default IPv6 route, `::/0`, with the internet gateway as the target.

4. The private IPv6-only subnets route tables have the default IPv6 route, `::/0`, with the EIGW as target.

5. Compute resources in public IPv6-only subnets use the internet gateway for IPv6 internet connectivity. They can directly initiate outbound internet connection and accept inbound internet connection, IPv6 endpoints in the internet, using their IPv6 addresses from the subnet CIDR. Note that security groups must allow IPv6 traffic.

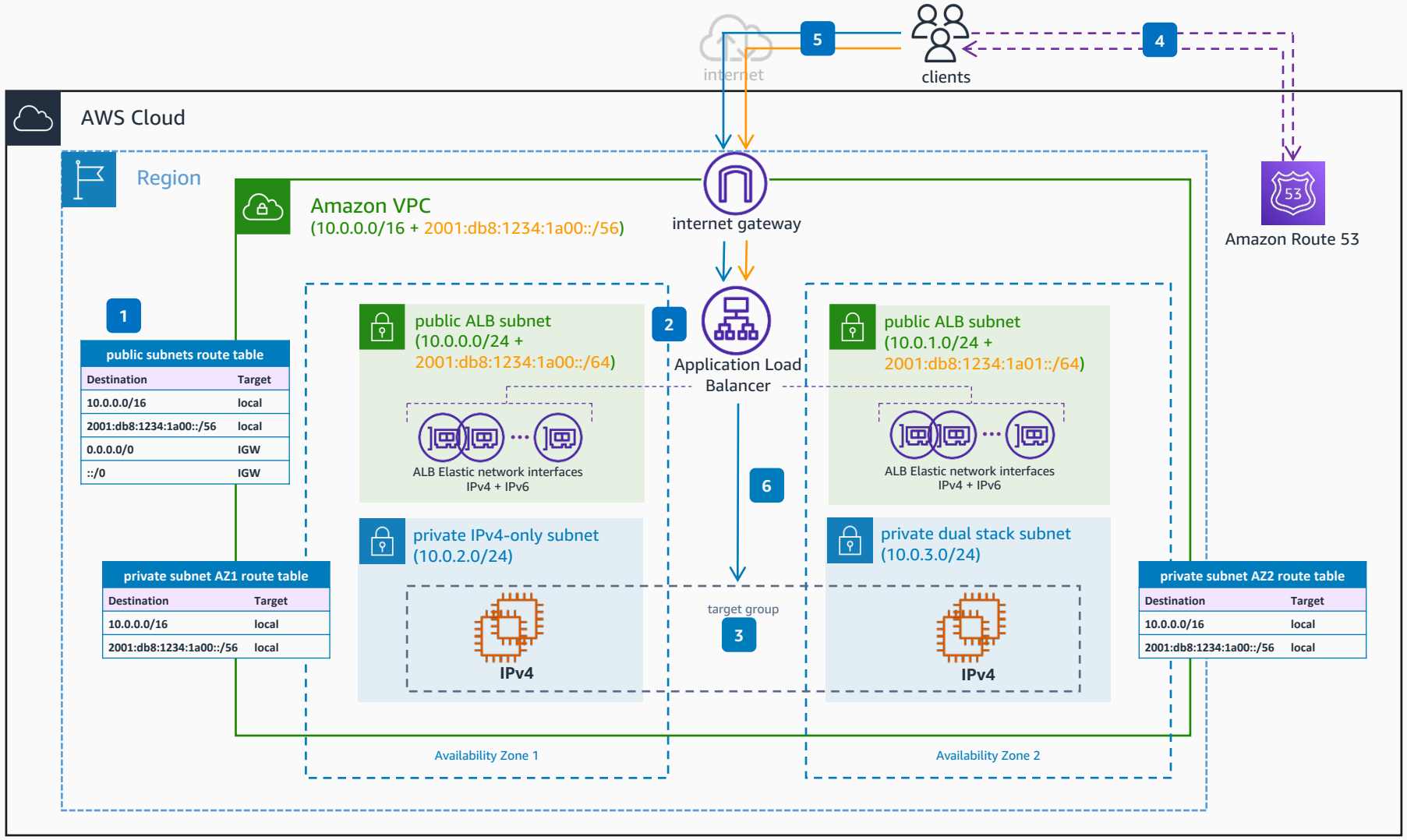
6. Resources in private IPv6-only subnets use the EIGW for outbound IPv6 internet connectivity. The egress-only internet gateway allows only outbound IPv6 connections to be opened from private **EC2** instances to internet IPv6 destinations, and the associated return traffic.

Reviewed for technical accuracy June 23, 2022

AWS Reference Architecture

4. IPv4 targets for dual stack internet-facing Application Load Balancer

Enable IPv4 and IPv6 internet connectivity to your application using Application Load Balancers (ALBs).



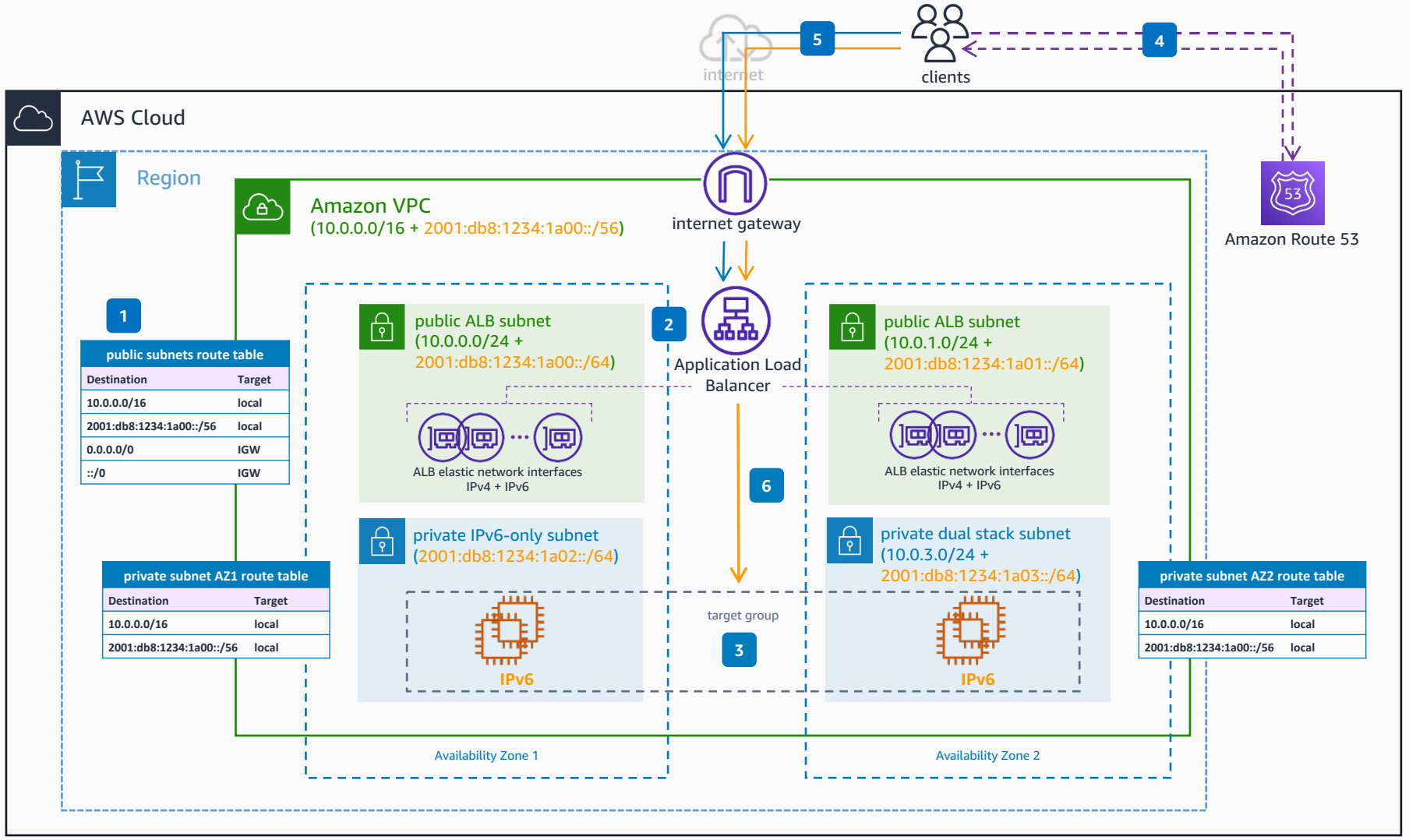
1. Configure your VPC **ALB** subnets for dual stack internet connectivity by adding the default routes for IPv4 and IPv6.
2. Deploy your dual stack internet-facing **ALB**, and select the dual stack **ALB** subnets in the VPC.
3. Your application stack remains unchanged with the dual stack added functionality for the application endpoints with **ALB**. The **ALB** and target group instances continue using IPv4 for communication.
4. The application clients query for the application name and receive the IPv4 or IPv6 address of the endpoint, based on their capabilities. For single stack customers, their stack determines the protocol to be used. For dual stack enabled clients, the operating system configuration determines the use of IPv4 or IPv6 for communication.
5. The clients open connections to the application endpoint, using either IPv4 or IPv6.
6. The **ALB** distributes traffic to the healthy targets in the target groups using IPv4 connections.

Reviewed for technical accuracy June 23, 2022

AWS Reference Architecture

5. IPv6 targets for dual stack internet-facing Application Load Balancer

Enable IPv4 and IPv6 internet connectivity to your application using ALBs.



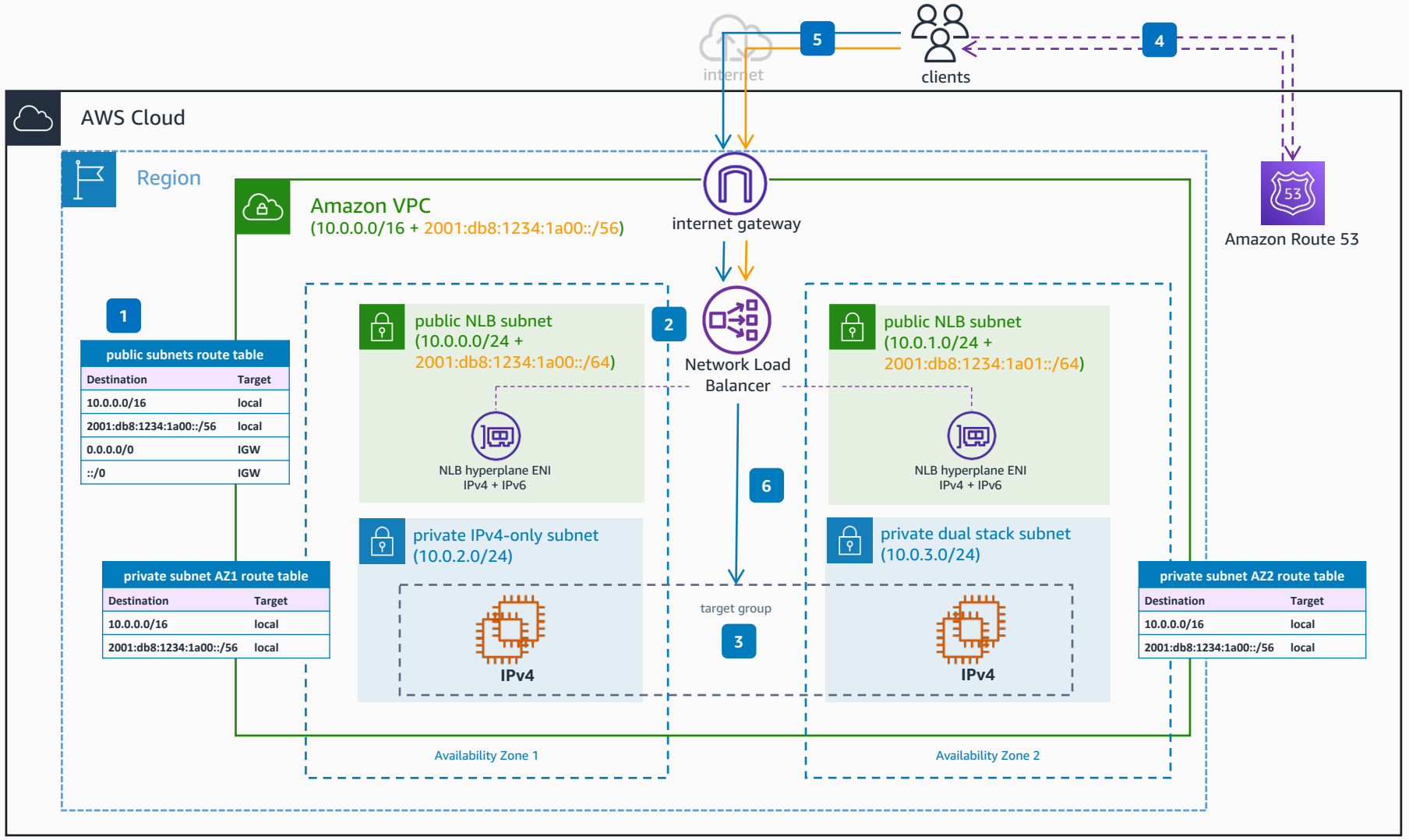
1. Configure your VPC **ALB** subnets for dual stack internet connectivity by adding the default routes for IPv4 and IPv6.
2. Deploy your dual stack internet-facing **ALB**, and select the dual stack **ALB** subnets in the VPC.
3. Your application stack can run natively on IPv6, as the **ALB** supports IPv6 targets. IPv6 target groups only support IP type targets.
4. The application clients query for the application name and receive the IPv4 or IPv6 address of the endpoint, based on their capabilities. For single stack customers, their stack determines the protocol to be used. For dual stack enabled clients, the operating system configuration determines the use of IPv4 or IPv6 for communication.
5. The clients open connections to the application endpoint, using either IPv4 or IPv6.
6. The **ALB** distributes traffic to the healthy targets in the target groups using IPv6 connections.

Reviewed for technical accuracy June 23, 2022

AWS Reference Architecture

6. IPv4 targets for dual stack Internet-facing Network Load Balancer

Enable IPv4 and IPv6 internet connectivity to your application using NLBs.



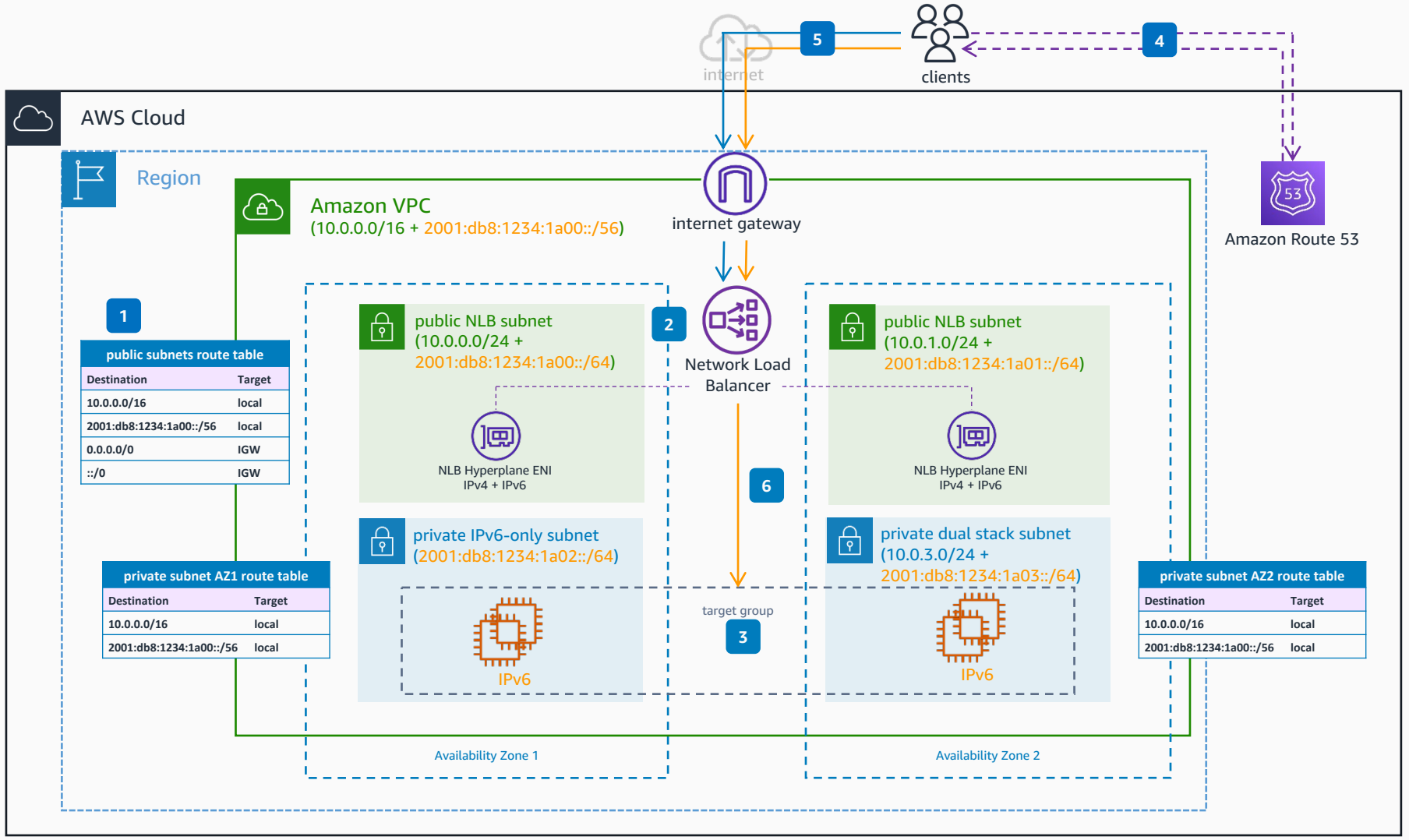
1. Configure your VPC **NLB** subnets for dual stack internet connectivity by adding the default routes for IPv4 and IPv6.
2. Deploy your dual stack internet-facing **NLB**, and select the dual stack **NLB** subnets in the VPC.
3. Your application stack remains unchanged with the dual stack added functionality for the for your application endpoints with **NLB**. The **NLB** and target group instances continue using IPv4 for communication.
4. The application clients query for the application name and receive the IPv4 or IPv6 address of the endpoint, based on their capabilities. For single stack customers, their stack determines the protocol to be used. For dual stack enabled clients, the operating system configuration determines the use of IPv4 or IPv6 for communication.
5. The clients open connections to the application endpoint, using wither IPv4 or IPv6.
6. The **NLB** distributes traffic to the healthy targets in the target groups using IPv4 connections.

Reviewed for technical accuracy June 23, 2022

AWS Reference Architecture

7. IPv6 targets for dual stack internet-facing Network Load Balancer

Enable IPv4 and IPv6 internet connectivity to your application using NLBs.



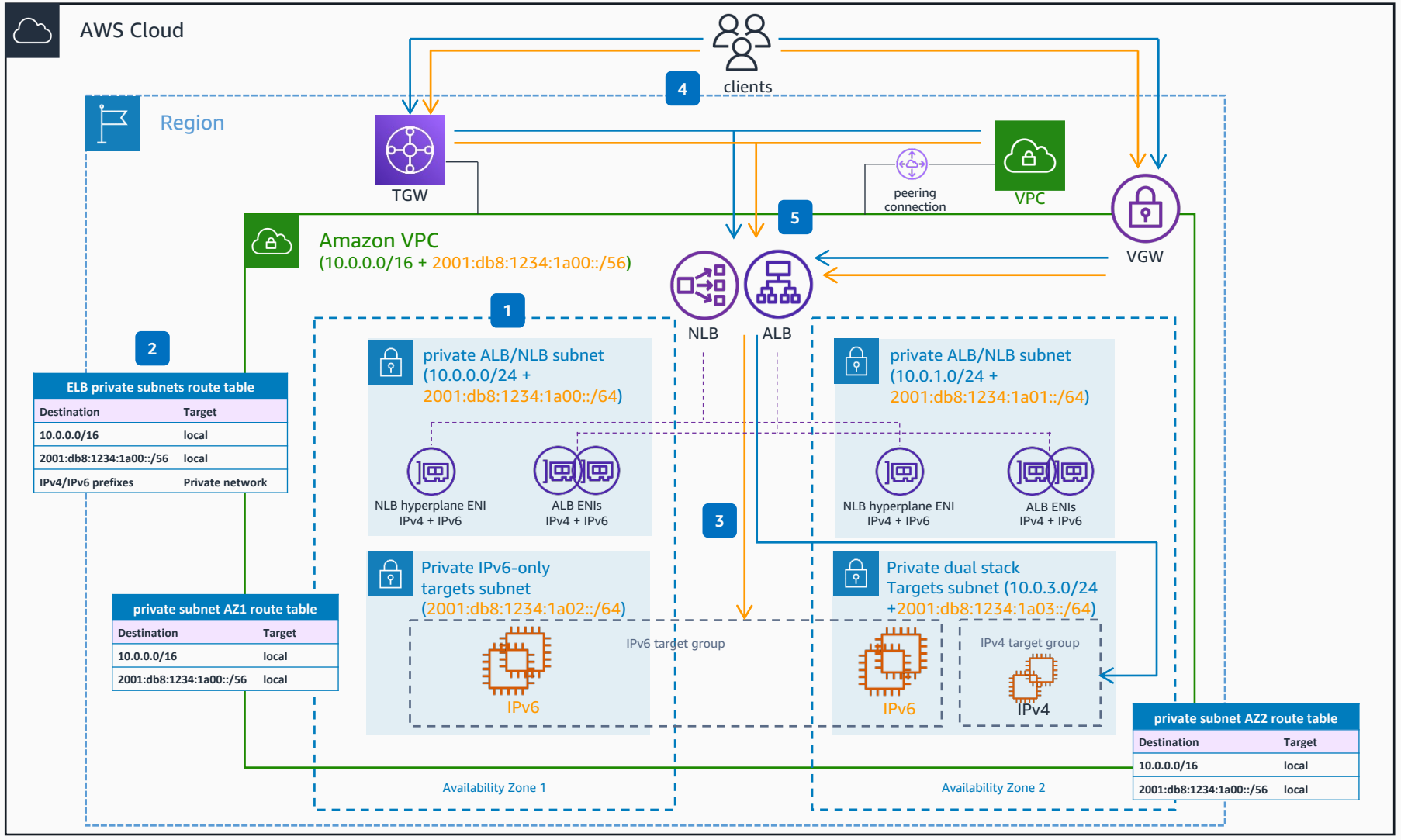
1. Configure your VPC **NLB** subnets for dual stack internet connectivity by adding the default routes for IPv4 and IPv6.
2. Deploy your dual stack internet-facing **NLB**, and select the dual stack **NLB** subnets in the VPC.
3. Your application stack can run natively on IPv6, as the **NLB** supports IPv6 targets. IPv6 target groups only support IP type targets.
4. The application clients query for the application name and receive the IPv4 or IPv6 address of the endpoint, based on their capabilities. For single stack customers, their stack determines the protocol to be used. For dual stack enabled clients, the operating system configuration determines the use of IPv4 or IPv6 for communication.
5. The clients open connections to the application endpoint, using wither IPv4 or IPv6.
6. The **NLB** distributes traffic to the healthy targets in the target groups using IPv6 connections.

Reviewed for technical accuracy June 23, 2022

AWS Reference Architecture

8. Dual stack internal Application and Network Load Balancers

Enable IPv4 and IPv6 private connectivity to your application using Application and Network Load Balancers.



1. Configure your VPC **NLB** or **Application Load Balancer (ALB)** subnets as dual stack, to accommodate for the internal **Elastic Load Balancing (ELB)** instances..

2. Depending on the private connectivity method you have in place for your VPC – VPC peering, **AWS Transit Gateway (TGW)**, Virtual Private Gateway, VPN, or **AWS Direct Connect** – the ALB/NLB private subnets must be configured with the appropriate routes, for both IPv4 and IPv6 stacks.

3. Target groups for both **ALBs** and **NLBs** can contain either only IPv6 targets or only IPv4 targets. You can't register an IPv4 target with an IPv6 target group.

4. Clients reaching out to the private **ELB** endpoints can be both IPv4 and IPv6 and can have connectivity over any private connectivity method supported by the Amazon VPC – VPC peering, **AWS Transit Gateway**, Virtual Private Gateway, VPN or **AWS Direct Connect**.

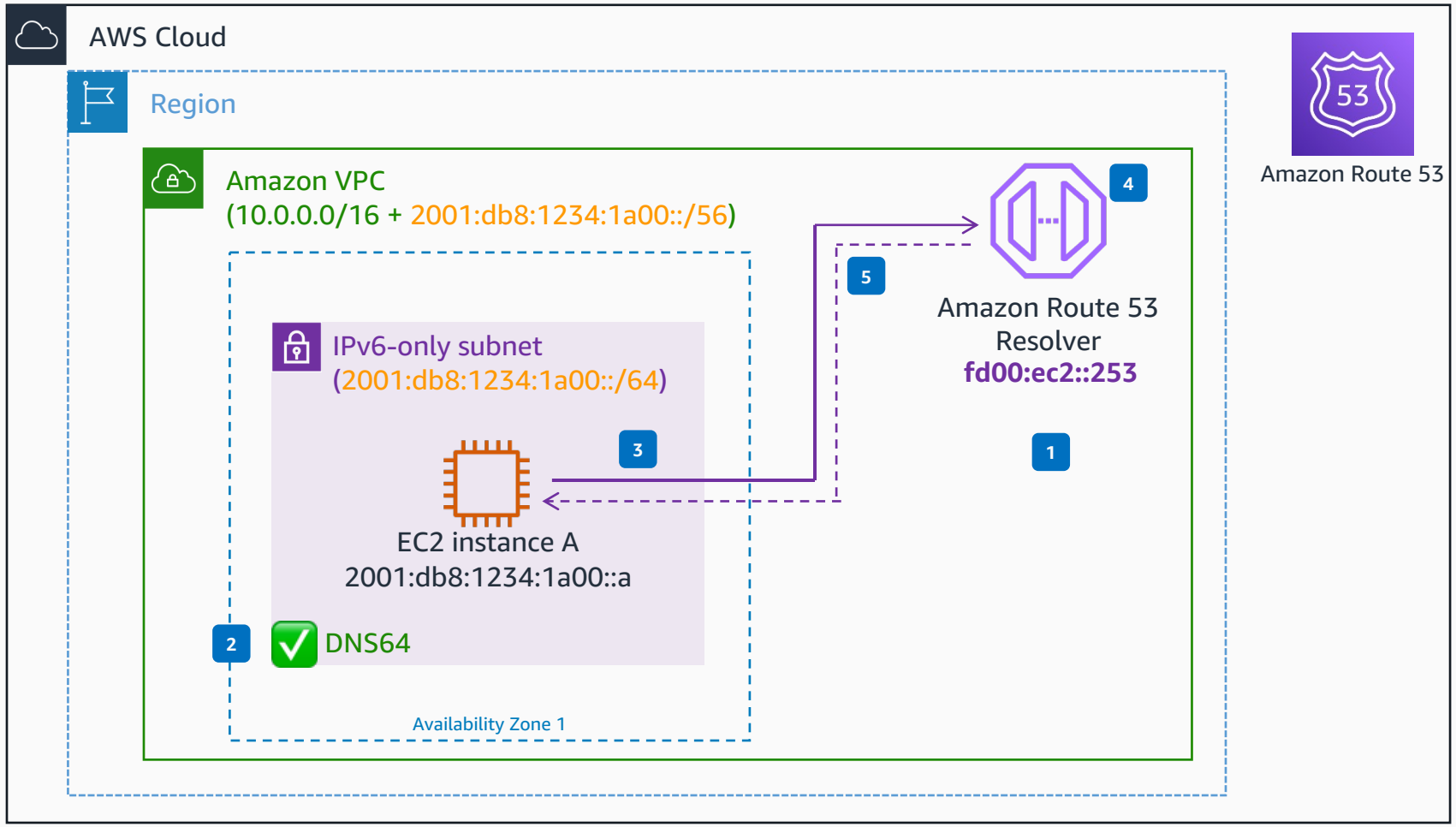
5. For internal ALBs and NLBs, the new attribute flag `ipv6.deny-all-igw-traffic` blocks internet gateway (IGW) access to the load balancer, preventing unintended access to your internal load balancer through an internet gateway. It is set to false for internet-facing load balancers and true for internal load balancers. This attribute does not prevent non-IGW internet access (such as, through peering, **Transit Gateway**, **AWS Direct Connect**, or **AWS VPN**).

Reviewed for technical accuracy June 23, 2022

AWS Reference Architecture

9. DNS64

Enable DNS resolution for queries from IPv6-only resources to IPv4-only names.



1. When you create a VPC, the **Amazon Route 53 Resolver** that is created by default, maps to a DNS server that runs on a reserved IPv4 address for the VPC network range, plus 2. For the IPv6 stack, the **Route 53 Resolver** can be reached at the local address `fd00:ec2::253`.

2. Without DNS64, a DNS query for an IPv4-only service will yield an IPv4 destination address in response and your IPv6-only service cannot communicate with it. To bridge this communication gap, you can enable DNS64 for a subnet and it applies to all the AWS resources within that subnet.

3. The IPv6-only instance sends a DNS query for on IPv4-only endpoint to the **Route 53 Resolver**.

4. With DNS64, the **Route 53 Resolver** looks up the DNS record for the service you queried for and does one of the following:

   - If the record contains an IPv6 address, it returns the original record and the connection is established without any translation over IPv6.

   - If there is no IPv6 address associated with the destination in the DNS record, the **Route 53 Resolver** synthesizes one by prepending the well-known /96 prefix, defined in RFC6052 (`64:ff9b::/96`), to the IPv4 address in the record.

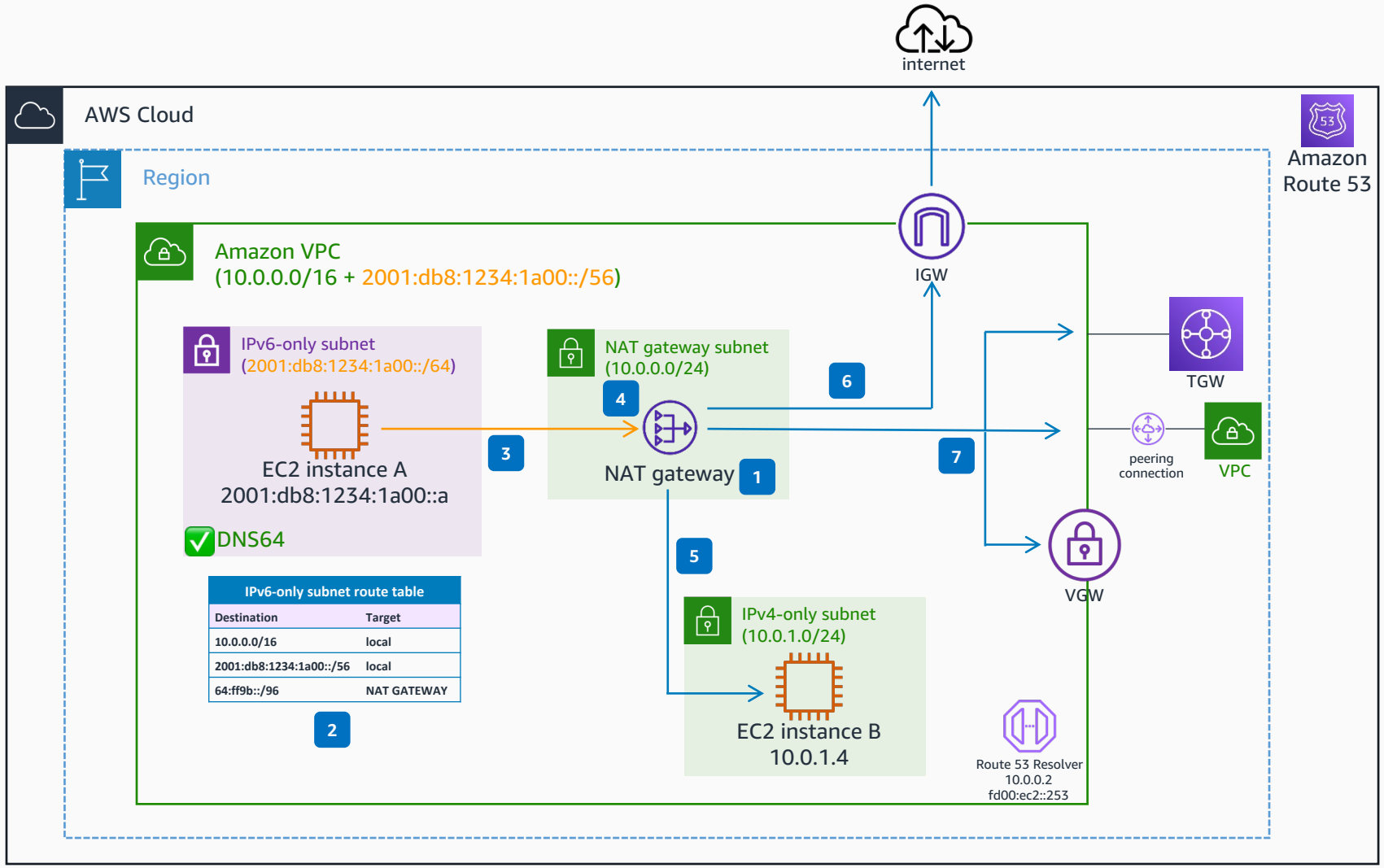
5. The **Amazon Route 53** resolver replies with a synthesized IPv6 address made of the well-known `64:ff9b::/96` prefix and the IPv4 address.

Reviewed for technical accuracy June 23, 2022

AWS Reference Architecture

10. NAT64

Enable communication between IPv6-only resources and IPv4-only endpoints.



1. NAT64 is automatically available on your existing NAT gateways or on any new NAT gateways you create. It's not a feature you enable or disable.

2. You need to route traffic for the well-known `64:ff9b::/96` prefix through the NAT gateway, which performs the necessary translation on the traffic to allow IPv6 services to access IPv4 services outside that subnet.

3. The IPv6 packet from the IPv6-only instance is sent to the NAT64 gateway. The IP is the instance IPv6 address, and the destination IP is the DNS64 synthesized IPv6 address, that was returned by the **Route 53 Resolver**.

4. From the `64:ff9b::/96` prefix, the NAT gateway recognizes that the original destination is IPv4 and translates the IPv6 packets to IPv4 by replacing:

   - The source IPv6 with its own private IPv4 address.
   - The destination IPv6 to IPv4 by truncating the 64:ff9b::/96 prefix.

5. Traffic can go to IPv4-only resources in the same VPC.

6. Traffic can go to IPv4-only endpoints in the Internet, if the NAT gateway is public and has an elastic IP associated.

7. Traffic can go to IPv4-only resources in the private network, over VPN, **AWS Direct Connect**, VPC peering, or **Transit Gateway**.

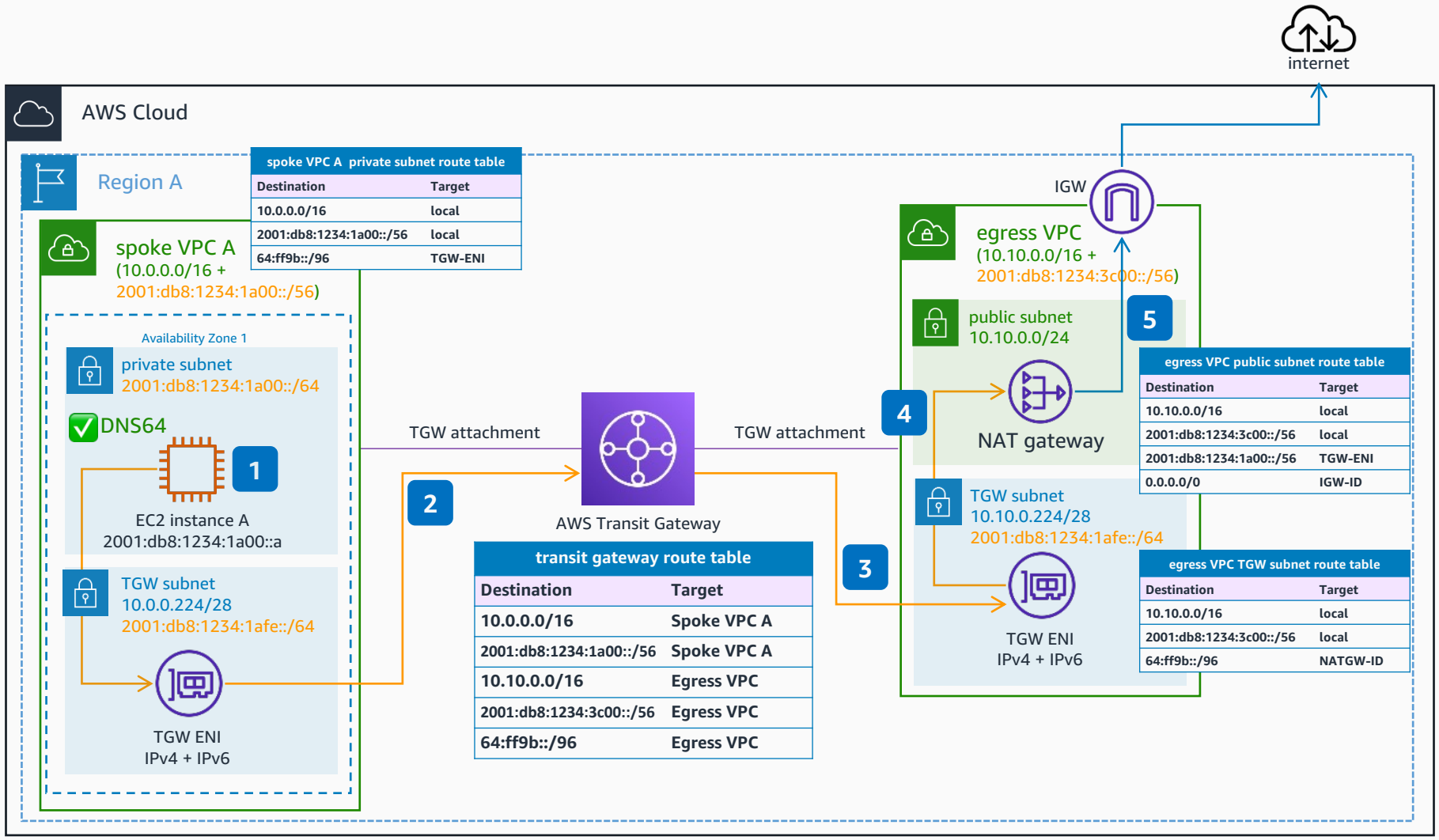
Reviewed for technical accuracy June 23, 2022

AWS Reference Architecture

NEW

11. Centralized egress traffic with NAT64

Centralize egress traffic to IPv4-only endpoints on the internet by forwarding all traffic to the well-known `64:ff9b::/96` prefix from all your spoke VPCs to a central egress VPC with a public NAT gateway through AWS Transit Gateway.



1. You need to enable DNS64 in all the subnets where you want your IPv6-only workloads to talk with IPv4-only destinations. That way, the **Amazon Route 53 Resolver** looks up the DNS record for the service queried. If there is no IPv6 address associated with the destination, it synthesizes one by prepending the well known `64:ff9b::/96` prefix, to the IPv4 address in the record.

2. You need to route traffic for the well-known `64:ff9b::/96` prefix through the NAT gateway located in the egress VPC. As per the spoke VPC A private subnet route table, all the traffic to the `64:ff9b::/96` prefix is routed first to the TGW ENI.

3. In the **AWS Transit Gateway** route table associated to the spoke VPC attachments, you need to add an static route sending all the traffic to the `64:ff9b::/96` prefix via the egress VPC attachment.

4. As per the egress VPC's TGW subnet route table, all the traffic to the `64:ff9b::/96` prefix is routed to the NAT gateway.

5. The NAT gateway recognizes that the original destination is IPv4 and translates the IPv6 packets to IPv4 by replacing:

   - The source IPv6 with its own public EIP IPv4 address.
   - The destination IPv6 to IPv4 by truncating the `64:ff9b::/96` prefix.

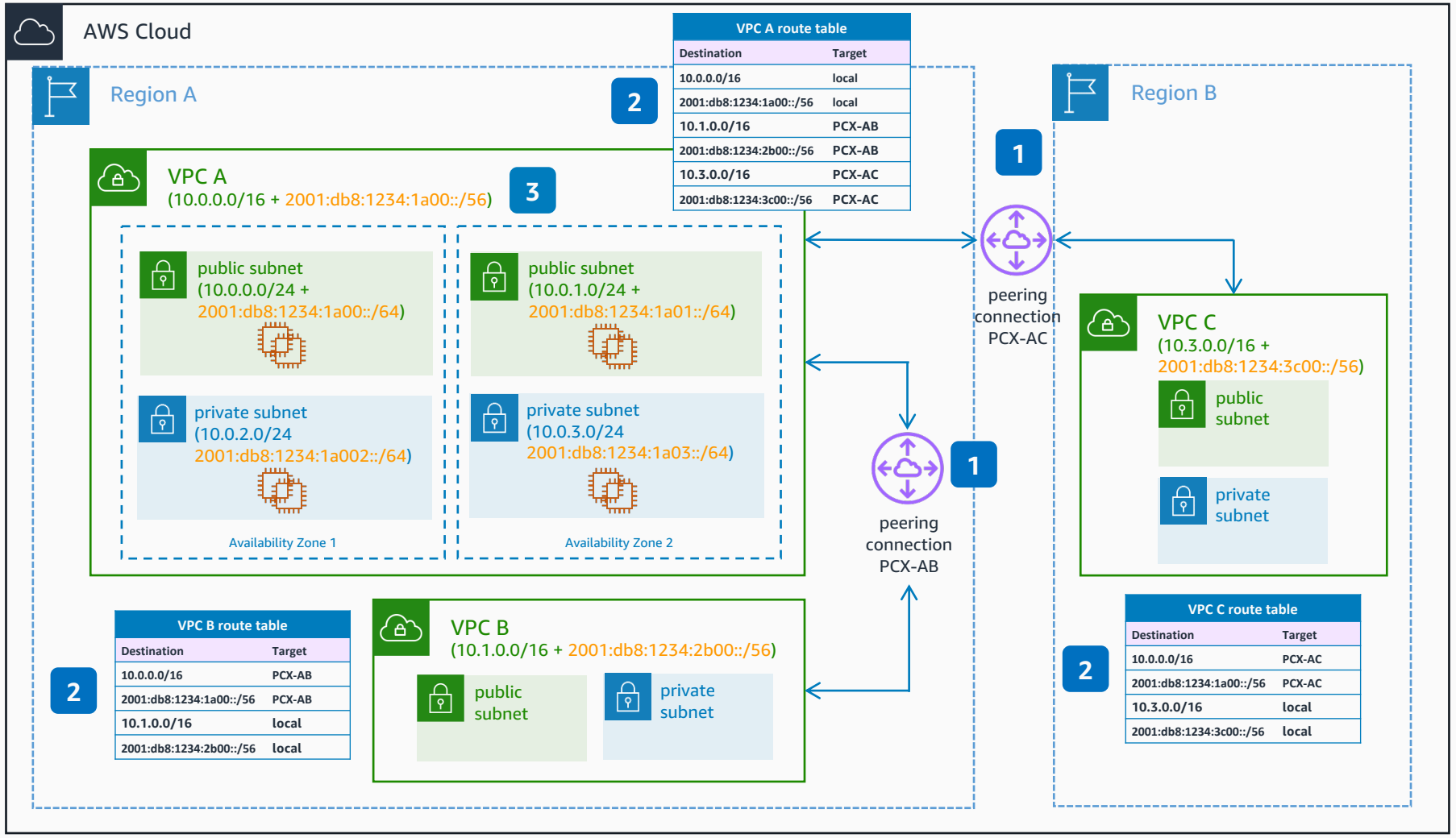
If only connectivity to IPv4 resources inside this topology is needed, a private NAT gateway can be used.

Reviewed for technical accuracy June 23, 2022

AWS Reference Architecture

12. Dual stack peering connectivity for the Amazon VPC

Configure IPv4 and IPv6 VPC peering connectivity.



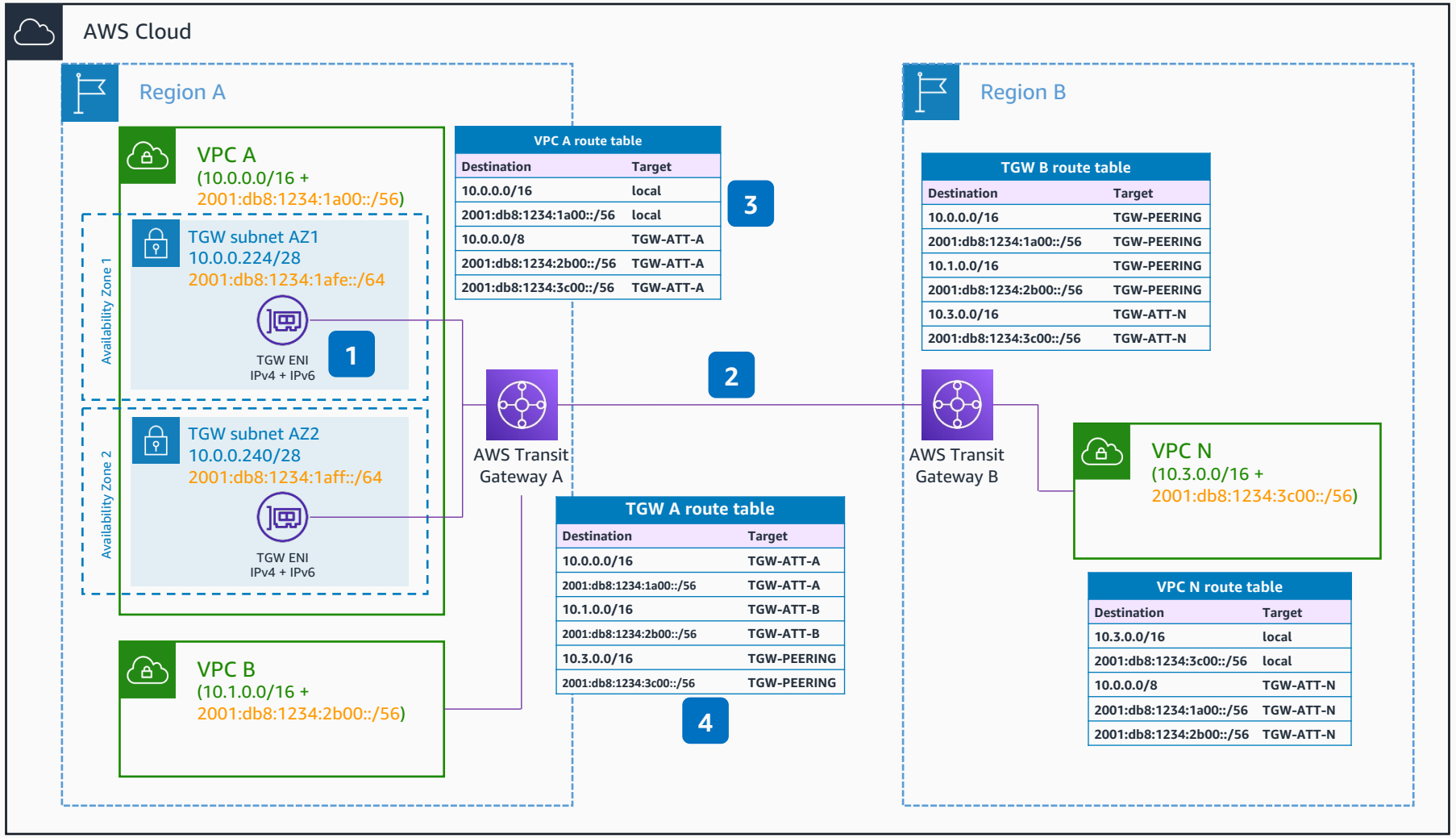
1. **Amazon VPC** peering natively supports routing for both IP stacks, and you can create both intra-Region and cross-Region VPC peering connections.
2. Update the route tables of VPCs with both the IPv4 and IPv6 CIDR blocks of the peered VPCs, with the respective peering connection ID as a target.
3. With VPC peering, IPv4 and IPv6 CIDR blocks of peered VPCs cannot overlap.

Reviewed for technical accuracy June 23, 2022

AWS Reference Architecture

13. Dual stack VPC connectivity with AWS Transit Gateway

Build global dual stack VPC connectivity with AWS Transit Gateway.



1. When you attach a VPC to an **AWS Transit Gateway**, you must specify one subnet from each Availability Zone to be used by the **Transit Gateway** to route traffic. For dual stack support, the **Transit Gateway** attachment subnets must have IPv6 CIDR blocks associated, and the attachment must be enable for IPv6 routing.

2. **Transit Gateway** peering connections natively support dual stack routing.

3. Update the VPC route tables with the necessary IPv4 and IPv6 routes.

4. If propagation is enabled for the VPC attachments to the transit gateway, the **Transit Gateway** route tables will be populated with both IPv4 and IPv6 VPC CIDR blocks.

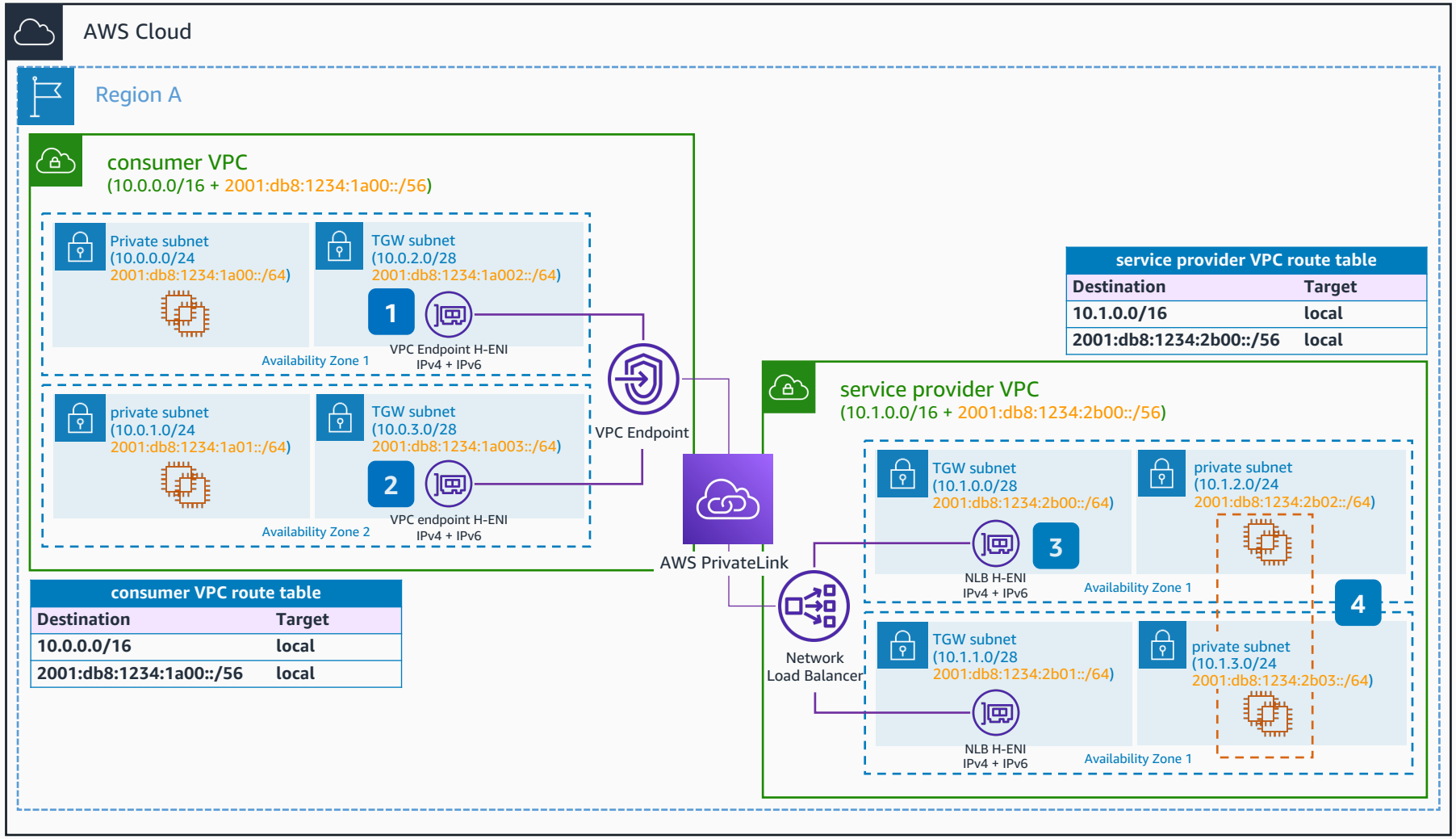
Reviewed for technical accuracy June 23, 2022

AWS Reference Architecture

NEW

14. Dual Stack VPC Connectivity with AWS PrivateLink

IPv6 support for AWS PrivateLink services and endpoints allows service providers to implement dual stack at the network border, and service consumers and service providers alike can adopt IPv6 at their own pace.



service provider VPC route table

Destination	Target
10.1.0.0/16	local
2001:db8:1234:2b00::/56	local

consumer VPC route table

Destination	Target
10.0.0.0/16	local
2001:db8:1234:1a00::/56	local

1 When you create **AWS PrivateLink** endpoints in the Consumer VPC, you need to specify the subnets to place them. For high-availability, we recommend you select at least two subnets in different Availability Zones. Those subnets can be IPv4-only, IPv6-only, or dual stack, and you can edit existing endpoints with IPv4 addresses to be dual stack.

2 **PrivateLink** endpoint's ENIs with IPv6 addresses always have `denyAllIGWTraffic` set on creation (this cannot be changed), making them unable to receive traffic from IGW, regardless of routing tables, and therefore private. The `denyAllIGWTraffic` attribute is not changeable after creation for any ENIs.

3 When you create the **Network Load Balancer** in the service provider VPC, its ENIs should be placed in dual stack subnets to enable the **NLB** to receive both IPv4 and IPv6 addresses. Remember that you should also create the **PrivateLink** endpoint service (which ties to the **NLB**) and share it with the consumers.

4 When placing your backend targets in the service provider VPC, you can use IPv4-only, IPv6-only or dual stack subnets. If you want to use IPv6 targets, remember that the target type of the **NLB** target group can only be IP.

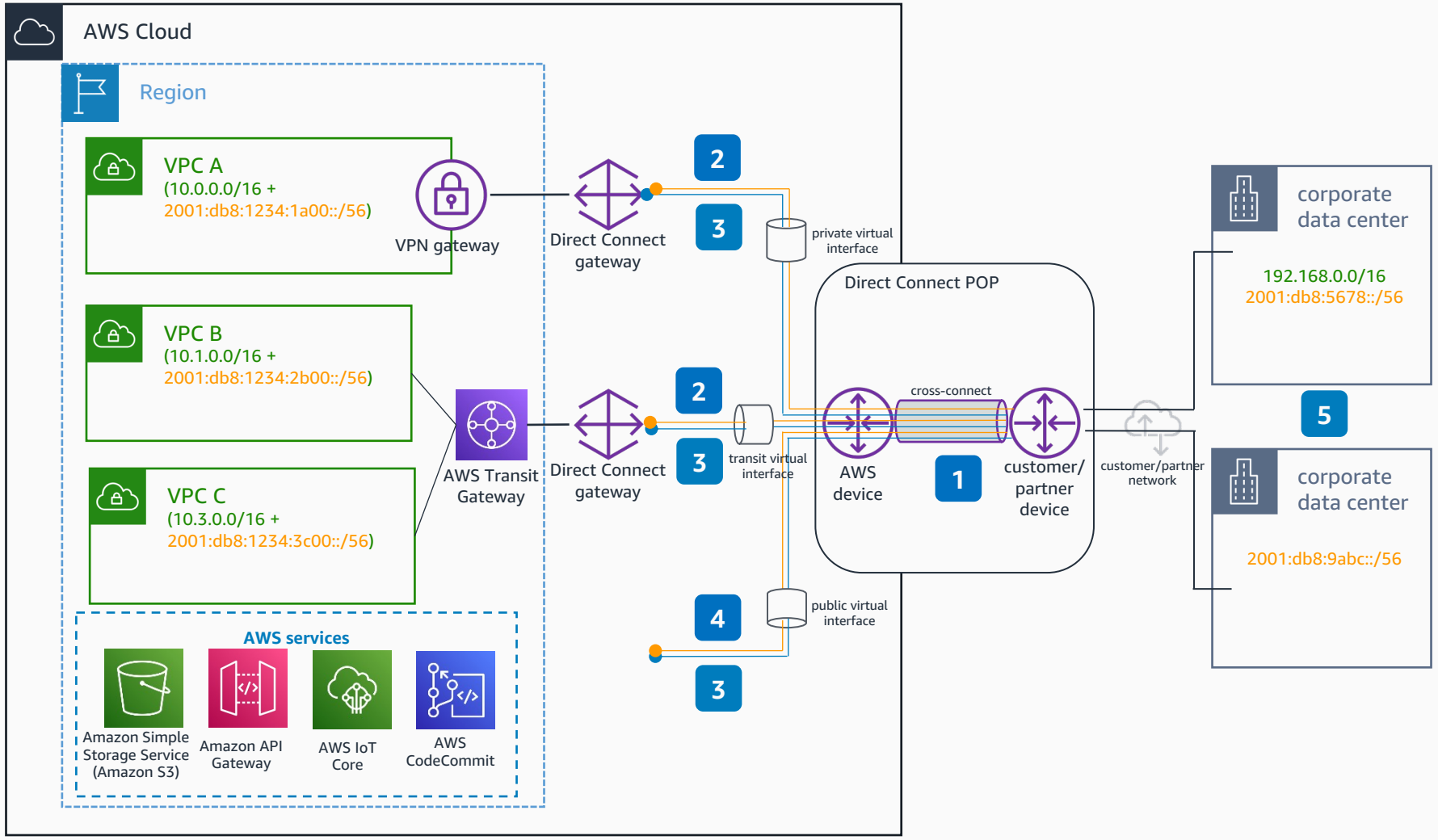
Note that the **NLB** will use its own IP address if any IP translation is needed (even if client preservation is on): for example if the connection to the **NLB** is used using IPv6 addresses, and the backend use IPv4.

Reviewed for technical accuracy June 23, 2022

AWS Reference Architecture

15. Dual stack hybrid connectivity with AWS Direct Connect

Build hybrid dual stack connectivity with AWS Direct Connect public, private, and transit virtual interfaces.



1. **AWS Direct Connect** virtual interfaces (VIFs) support both IPv4 and IPv6 Border Gateway Protocol (BGP) sessions for dual stack operation.

2. For all types of VIFs IPv6 BGP peering, Amazon assigns a /125 CIDR which is not configurable.

3. Private and transit VIFs IPv4 configuration make use of either Amazon-generated private IPv4 addresses, or addresses that you configure. If you specify your own, ensure that you specify private CIDRs for your router interface and the **AWS Direct Connect** interface only. (For example, do not specify other IP addresses from your local network.)

4. For public VIFs IPv4 BGP peering, you must specify unique public /31 IPv4 CIDR that you own, or submit a request to have a CIDR block assigned.

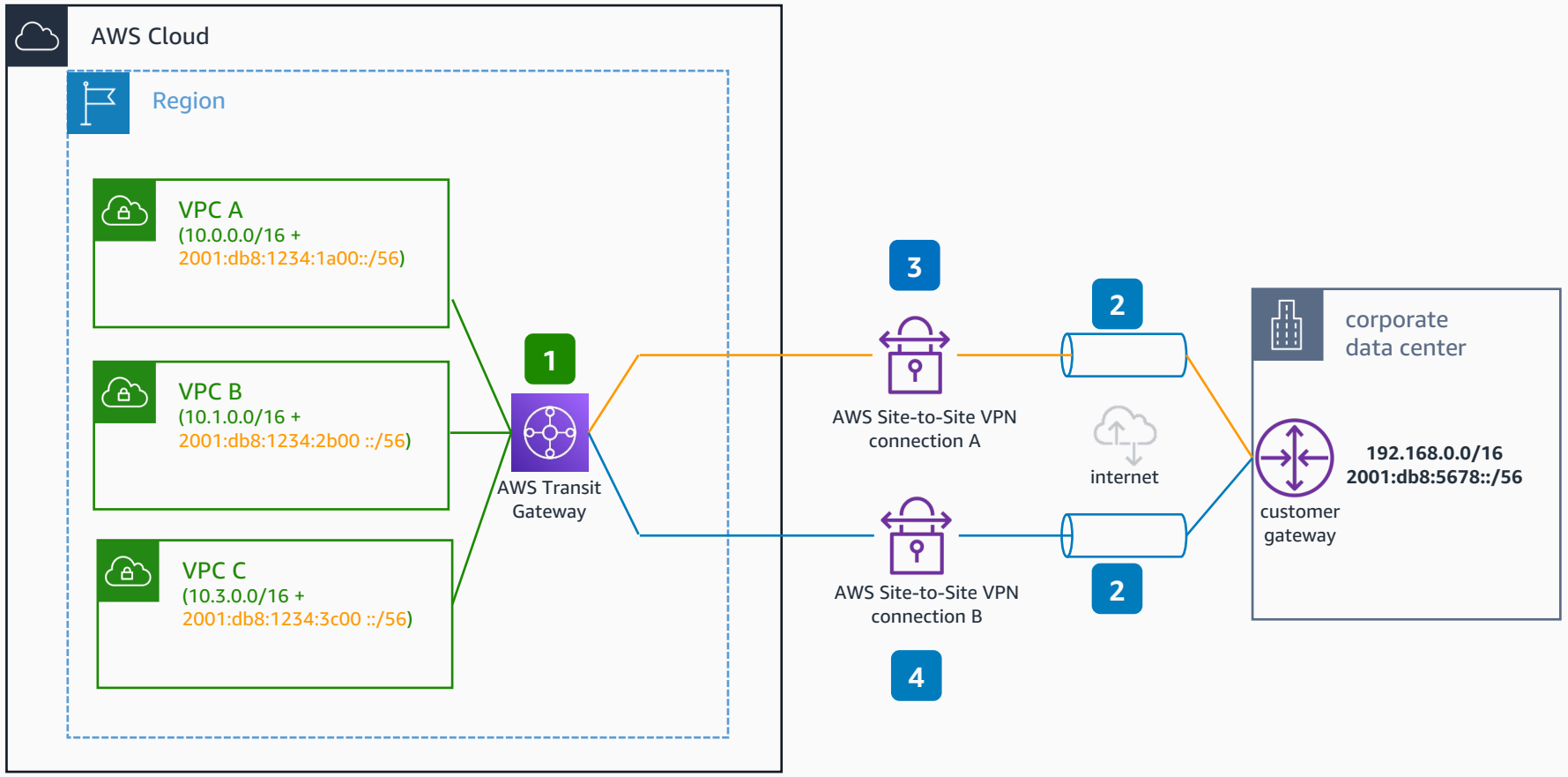
5. You can maintain both IPv6-only and dual stack on premises environment and use **AWS Direct Connect** for dedicated and private connectivity to your dual stack or IPv6-only workload footprint in AWS.

Reviewed for technical accuracy June 23, 2022

AWS Reference Architecture

16. Dual stack VPN connectivity with AWS Transit Gateway

Build hybrid dual stack VPN connectivity with AWS Transit Gateway.

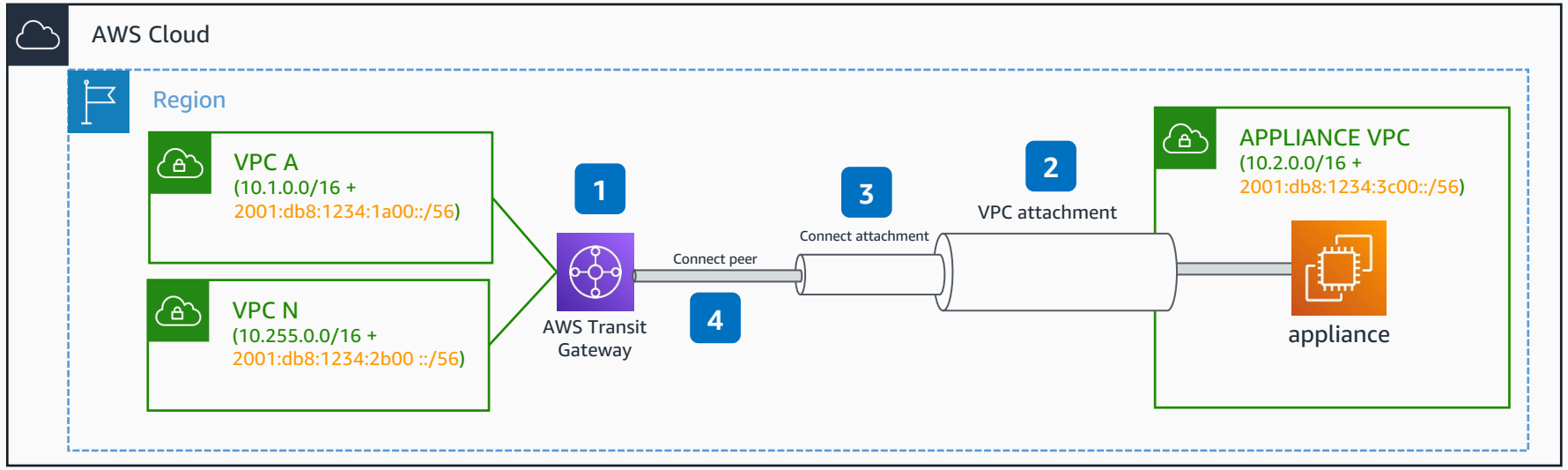


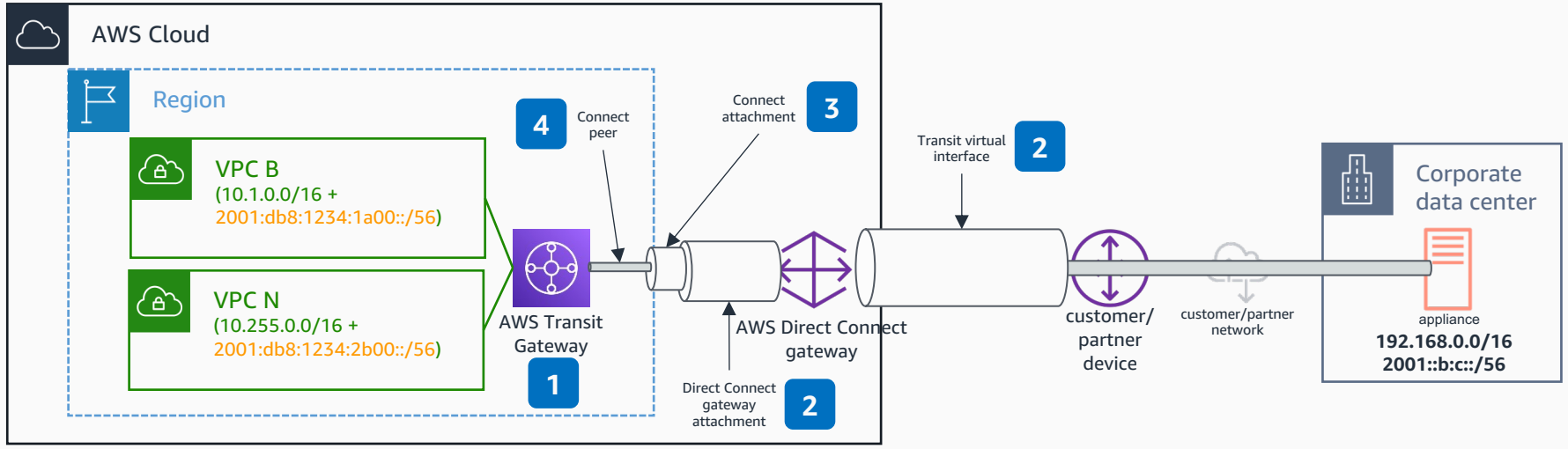
1. To configure dual stack support for VPN, create two transit gateway VPN attachments, one for each IP stack.
2. The outer IP addresses of the VPN connections are public IPv4 addresses.
3. One of the VPN tunnels is configured with inner IPv6 addresses, and routes IPv6 traffic. This enables you to maintain IPv6-only in on-premises environments, and configure IPv6-only connectivity with AWS environments, as long as you keep the outer VPN tunnel IPv4 public addresses.
4. The other VPN tunnel is configured with inner IPv4 addresses, routes IPv4 traffic, you require both IPv4 and IPv6 connectivity between your AWS environments and your on-premises workloads.

Reviewed for technical accuracy June 23, 2022

17. Dual stack AWS Transit Gateway Connect

Build dual stack appliance integration with AWS Transit Gateway Connect.





1. For **Transit Gateway** connect configuration, configure the transit gateway CIDR block. You can specify a size /24 CIDR block or larger for IPv4, or a size /64 CIDR block or larger for IPv6.
2. The VPC attachment to the **Transit Gateway** is dual stack enabled. The **Transit virtual interface (VIF)** can be configured with both IPv4 and IPv6 BGP peers.
3. The **Transit Gateway** connect attachment uses the VPC or **AWS Direct Connect Gateway** attachment of the as transport.
4. When you create the connect peers, you must specify the peer Generic Routing Encapsulation (GRE) address, which can be IPv4 or IPv6. The Border Gateway Protocol (BGP) inside CIDR blocks can be configured for both IPv4 and IPv6. For IPv4 you must specify a /29 CIDR, and for IPv6, a /125 CIDR.

Reviewed for technical accuracy June 23, 2022

AWS Reference Architecture