



AWS Academy Cloud Architecting  
Module 08 Student Guide  
Version 3.0.0

200-ACACAD-30-EN-SG

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

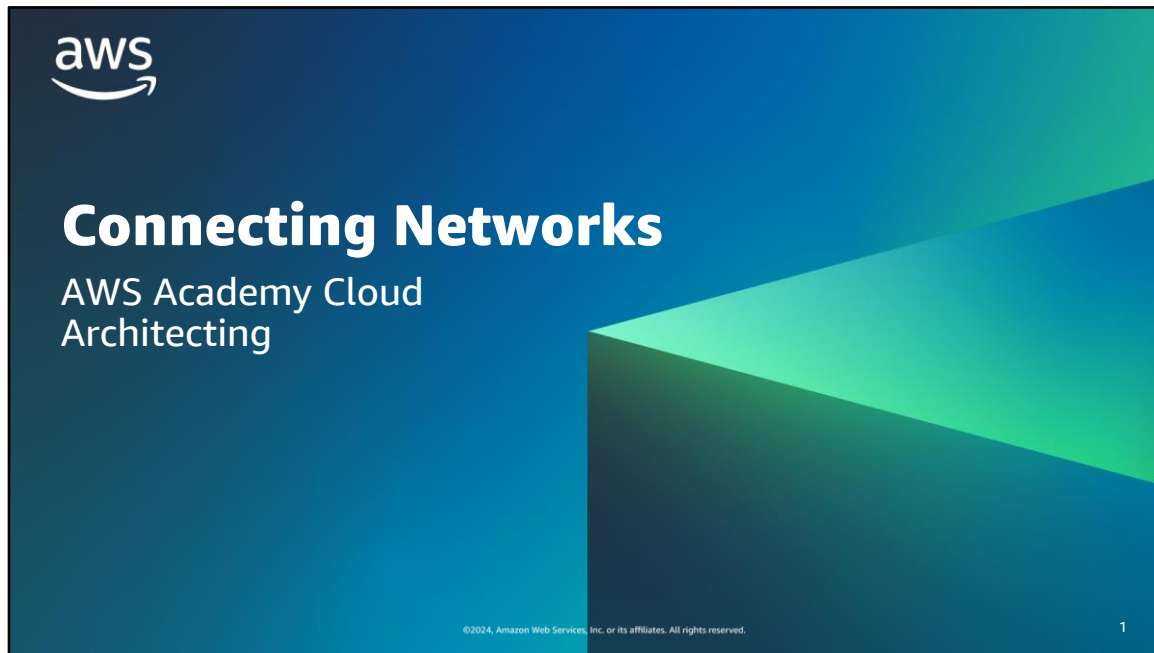
This work may not be reproduced or redistributed, in whole or in part,  
without prior written permission from Amazon Web Services, Inc.  
Commercial copying, lending, or selling is prohibited.

All trademarks are the property of their owners.

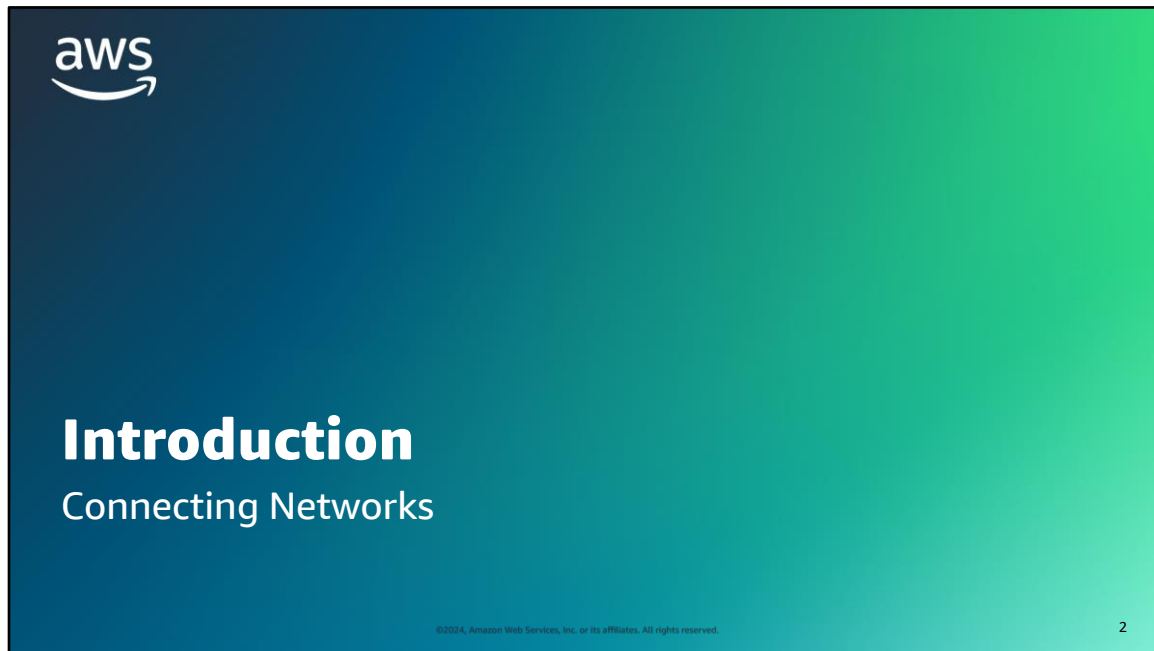
# Contents

[Module 8: Connecting Networks](#)

4



Welcome to the Connecting Networks module. This module focuses on connecting virtual private cloud (VPC) networks as well as connecting on-premises networks to VPCs.



This introduction section describes the content of this module.

## Module objectives



This module prepares you to do the following:

- Describe how to connect an on-premises network to the Amazon Web Services (AWS) Cloud.
- Describe how to connect multiple virtual private clouds (VPCs) in the AWS Cloud.
- Connect VPCs in the AWS Cloud by using VPC peering.
- Describe how to scale VPCs in the AWS Cloud.
- Use the AWS Well-Architected Framework principles when connecting networks.

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

3

## Module overview

### Presentation sections

- Scaling your VPC network with AWS Transit Gateway
- Connecting VPCs in AWS with VPC peering
- Connecting to your remote network with AWS Site-to-Site VPN
- Connecting to your remote network with AWS Direct Connect
- Applying AWS Well-Architected Framework to network connectivity

### Activity

- Configure AWS Transit Gateway Routes

### Knowledge checks

- 10-question knowledge check
- Sample exam question



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

4

The objectives of this module are presented across multiple sections.

In the activity in this module you will configure route tables for VPCs that you want to connect to each other.

The module wraps up with a 10-question knowledge check delivered in the online course and a sample exam question to discuss in class.

The next slide describes the lab in this module.

## Hands-on labs in this module

### Guided lab

- Creating a VPC Peering Connection





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

5

This module includes the guided lab that is listed. In this lab, you create and configure a peer connection between two VPCs. Additional information about this lab is included in the student guide where the lab takes place, and the lab environment provides detailed instructions.



**As a cloud architect designing connected networks:**

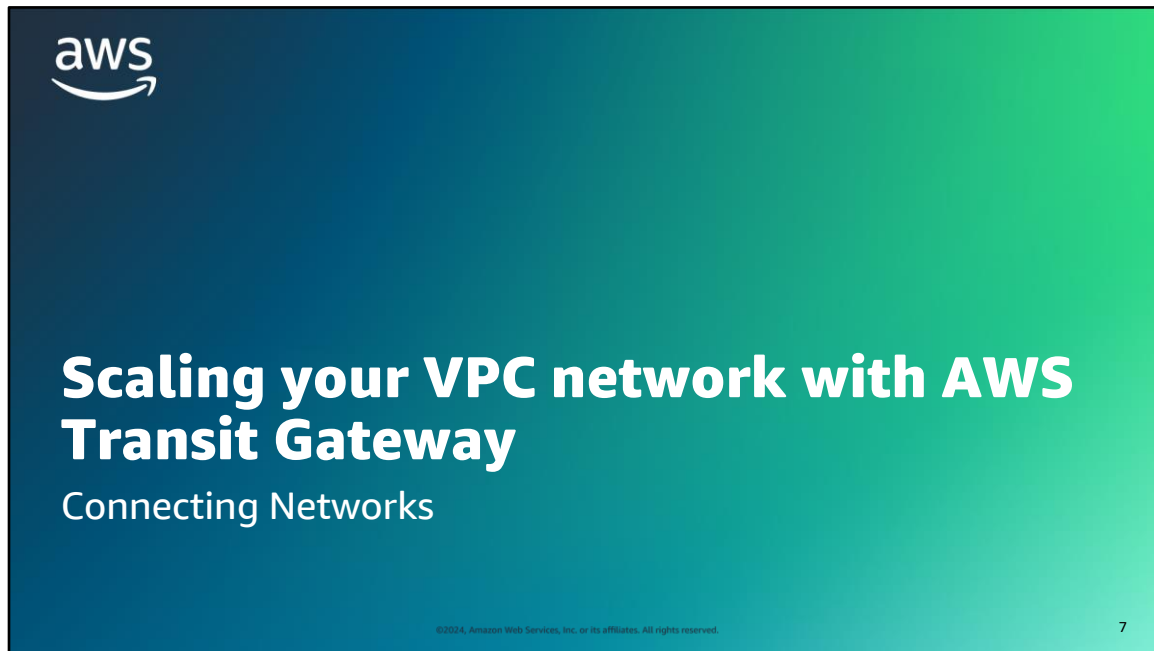


- I need to design for failover capability and adequate bandwidth to accommodate applications deployed on premises, in the cloud, or as a hybrid solution, so that my networks perform as needed.
- I need to choose network components that optimize performance and reduce data transfer costs between networks, so that I can maximize the business value.
- I need to protect data in transit between networks, so that I can meet data security compliance requirements.

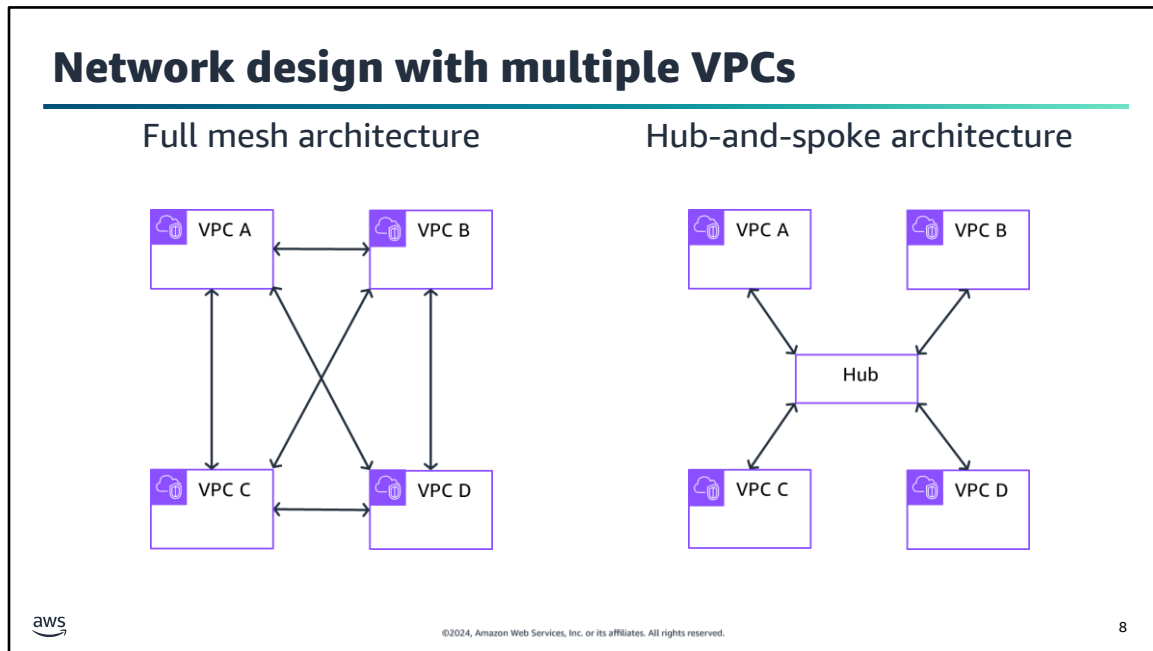
©2024, Amazon Web Services, Inc. or its affiliates. All rights reserved.

6

This slide asks you to take the perspective of a cloud architect as you think about how to approach cloud network design. Keep these considerations in mind as you progress through this module, remembering that the cloud architect should work backward from the business need to design the best architecture for a specific use case. As you progress through the module, consider the café scenario presented in the course as an example business need, and think about how you would address these needs for the fictional café business.



This section looks at connecting multiple VPCs with AWS Transit Gateway.




Now that you know how to create a VPC and its components, you should expand your network design to multiple networks. For example, when your business or architecture expands, you might need to separate logical elements for security, for architectural purposes, or for management simplicity. However, many VPC environments do need connectivity to other network environments. Many enterprise organizations deploy hundreds of VPCs that need connectivity to each other.

When your architecture requires multiple VPCs, you can design each VPC to connect to each of the other required VPCs. This design is called full mesh architecture where every node is directly connected to every other node. This design works well for networks with a small number of VPCs that require fast network speeds with little network latency. In the example on this slide, four VPCs are connected with six connections. The number of connection links required for a given number of nodes (N) can be calculated as  $N * (N - 1) / 2$ . Therefore, for 100 nodes, you will need to configure  $100 * 99 / 2 = 4,950$  connections. This architecture creates a heavy operational and maintenance effort for network engineers.

Another approach is to use a hub-and-spoke architecture where there is a central intermediary hub to manage connectivity. Each node requires only one connection to the hub. In the example on this slide, four VPCs need four connections to communicate through the hub. For a 100 nodes, you will need to configure 100 connections. This architecture significantly simplifies management and reduces operational effort. This design works well for networks with a large number of VPCs that can tolerate the added latency for hub processing.

## AWS Transit Gateway



Transit Gateway

- Is a centralized, Regional router to connect VPCs and on-premises networks based on hub-and-spoke architecture
- Is a managed AWS service that automatically scales based on the volume of network traffic
- Can be peered with other transit gateways in other AWS Regions and AWS accounts
- Incurs cost charges based on the number of connections and amount of traffic throughput
- Has a Transit Gateway Flow Logs feature to publish transit gateway traffic logs

aws

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

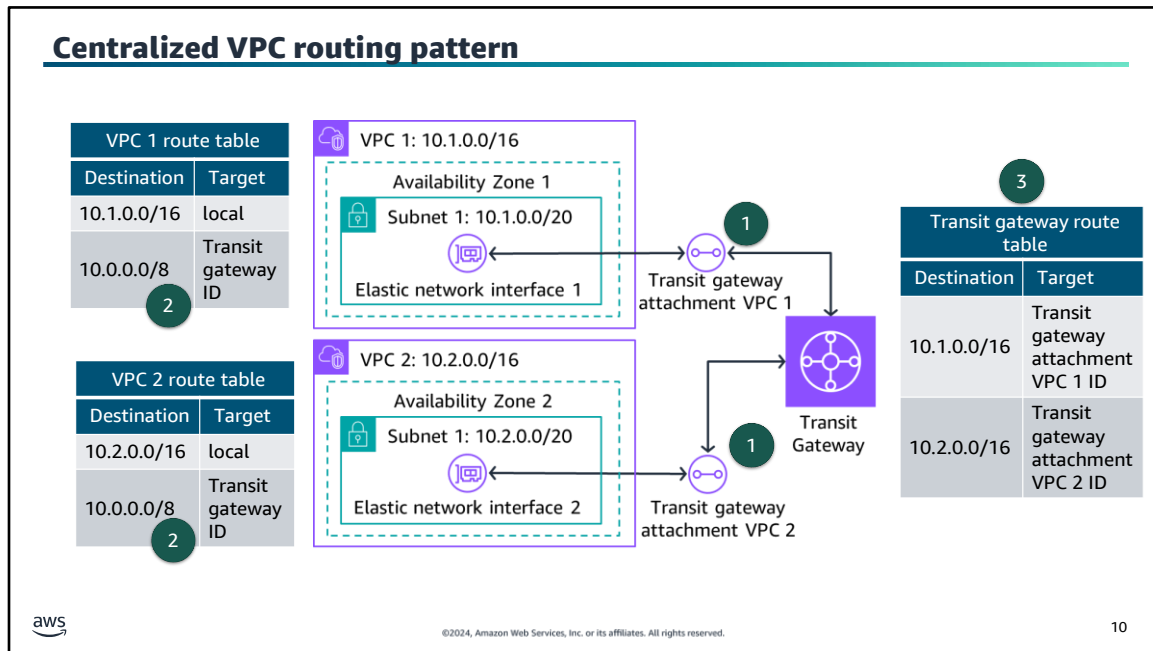
9

Transit Gateway is a managed AWS service that you can use to connect your VPCs and on-premises networks to a single gateway called a transit gateway. With the Transit Gateway service, you create and manage only a single connection from the central gateway into each VPC, on-premises data center, or remote office across your network. Transit Gateway simplifies network management and scaling as workloads expand because it automatically scales based on traffic throughput. You can use it to connect thousands of VPCs and on-premises networks. Transit Gateway charges per hour for the number of connections that you make to the transit gateway and the amount of traffic that flows through the transit gateway. For more information, see [Transit Gateway pricing link](#) in resource list.

Transit Gateway uses a hub-and-spoke architecture. Any new VPC connected to the transit gateway is automatically available to every other network that is connected to the transit gateway. Transit Gateway supports dynamic and static routing between attached VPCs. Static routes are configured before network traffic can be routed. Dynamic routing requires routers to exchange route information with other routers to discover routing paths when traffic is routed. Transit Gateway supports both IPv4 and IPv6 traffic.

Transit gateways can be peered with each other within the same AWS Region or between different AWS Regions. Transit Gateway traffic always stays on the global AWS backbone infrastructure and never traverses the public internet. This reduces threat vectors such as common exploits and distributed denial of service (DDoS) attacks.

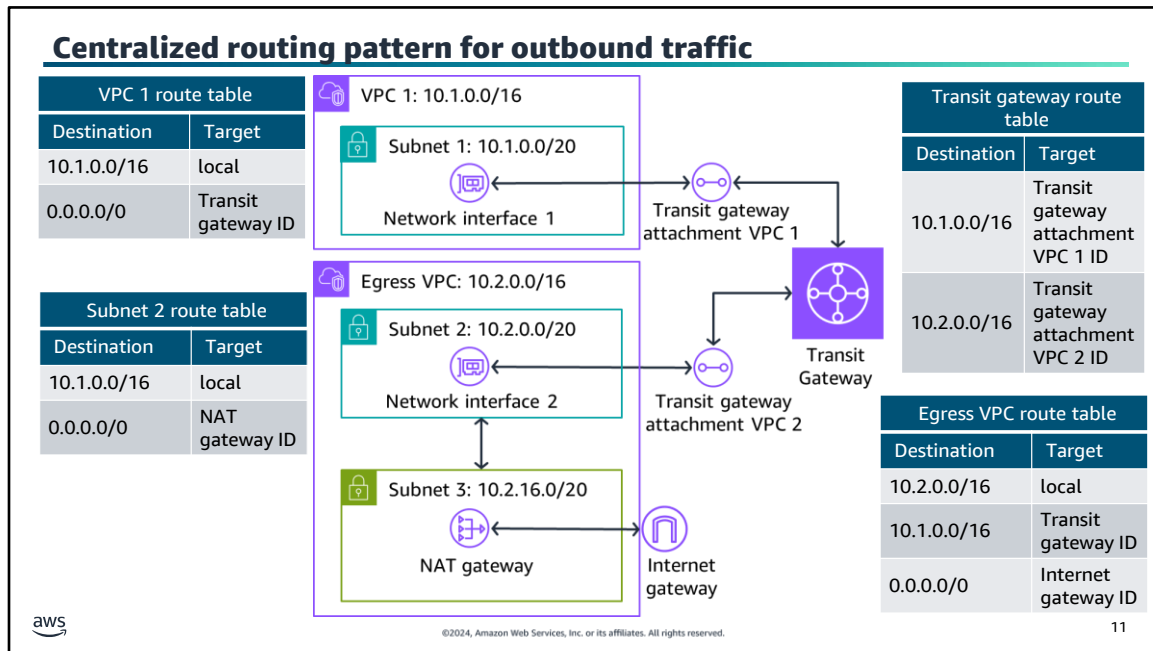
Transit Gateway Flow Logs is a feature that you can use to capture information about the IP traffic going to and from your transit gateways. Flow log data can be published to Amazon CloudWatch Logs, Amazon Simple Storage Service (Amazon S3), or Amazon Kinesis Data Firehose.



To understand how to connect multiple VPCs by using Transit Gateway, consider this scenario. You want to fully connect multiple VPCs in your network, and each VPC should have full access to the resources of all other VPCs. In this scenario, you will learn how to deploy Transit Gateway and the VPCs with nonoverlapping IP address space in a single Region. You will then learn how to attach the transit gateway to the VPCs. Every new VPC that is added to the transit gateway will be accessible from the existing VPCs:

1. Create a VPC attachment by using Transit Gateway. This step will connect the transit gateway to the VPC through an elastic network interface deployed in the subnets. You must ensure that every Availability Zone in the VPC has a network interface to connect the VPC to the transit gateway. You can do this by selecting at least one subnet from each Availability Zone for the network interface.
2. Add a transit gateway route to each VPC route table to send traffic that's destined for all of the other VPCs in the network to the transit gateway. In this example, the second line of the VPC 1 route table shows that traffic destined for the 10.0.0.0/8 network is sent to the transit gateway. This route makes it possible for any traffic from VPC 1 going to other VPCs to be sent to the transit gateway. This occurs because the Classless Inter-Domain Routing (CIDR) block 10.0.0.0/8 includes the 10.X.0.0/16 CIDR blocks, which are used by the individual VPCs. It is a wild card routing CIDR block and is not the same as a VPC CIDR block that has a maximum CIDR block range of /16.
3. Configure the transit gateway route table to route traffic to the connected VPCs. When you create a transit gateway, a default transit gateway route table is created. Each route in the transit gateway route table gives the transit gateway the ability to send traffic destined for one of the VPCs to a corresponding attachment, which is a reference to the network interface that is attached to the VPC itself. In this example, there is a route in the transit gateway route table that sends any traffic destined for the 10.1.0.0/16 network to the transit gateway attachment VPC 1. Similarly, any traffic destined for the VPC 2 network is sent to the VPC 2 transit gateway attachment. A transit gateway can have multiple route tables, and attachments are associated with a specific route table.

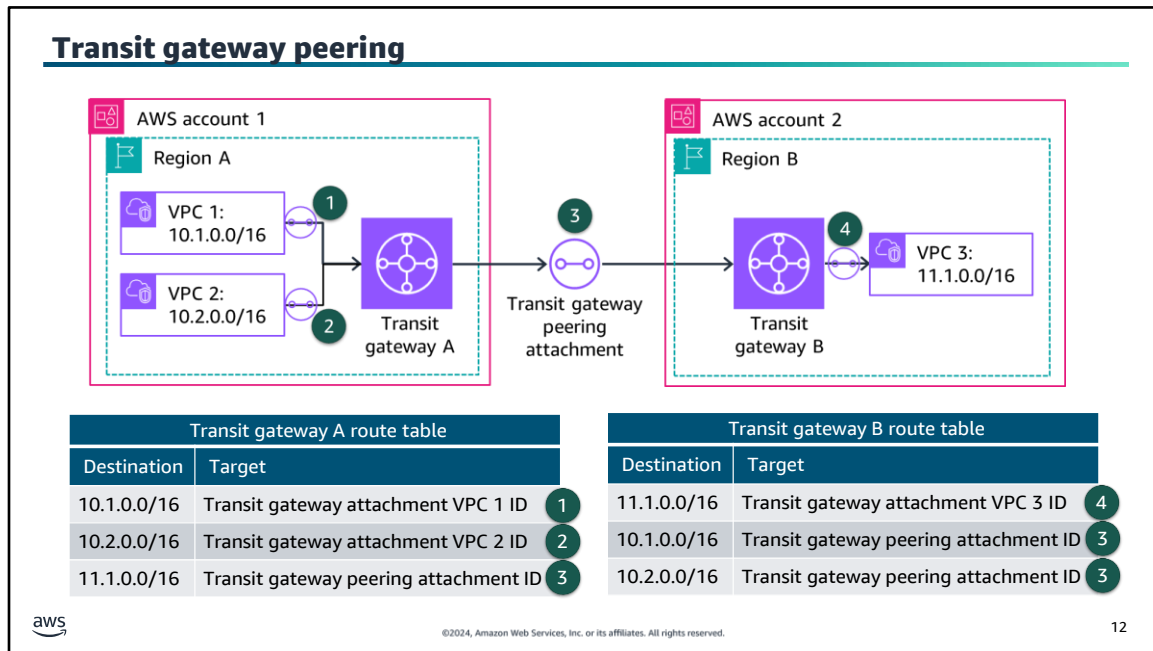
You can extend this example to include hundreds of VPCs within the routable 10.0.0.0/8 IP address range.



You can design a dedicated VPC to handle egress internet traffic for added outbound network security. This pattern is called centralized outbound routing and will route any internet traffic from VPC 1 to the egress VPC that contains a NAT gateway. In the example on this slide, applications in VPC 1 need outbound-only internet access. VPC 1 is already connected to the egress VPC with Transit Gateway based on the centralized VPC routing pattern. The egress VPC subnet 2 route table is configured to point to the NAT gateway in the public subnet 3 of the egress VPC. The egress VPC route table has a rule to route all traffic to the internet gateway. This pattern provides a centralized gate for outbound traffic for monitoring and security purposes. It is also cost efficient to run NAT gateways in one VPC instead of having a NAT gateway for each VPC. For redundancy, run a NAT gateway for each Availability Zone in the egress VPC.

You can also use this design to centralize access to shared services such as traffic inspection, or interface VPC endpoint access. One VPC is the designated service VPC that all other VPCs use. This centralization simplifies the complexity of managing these resources in several VPCs and provides better access control.

Transit gateway connectivity to on-premises networks is discussed later in the module.



**Image description:** In the example on this slide, transit gateway A has two VPC attachments and has one transit gateway peering attachment labelled as 3. Transit gateway attachment VPC 1 is labelled as 1. Transit gateway attachment VPC 2 is labelled as 2. Transit gateway B has one VPC attachment. Transit gateway attachment VPC 3 is labelled as 4. Network traffic from the subnets in VPC 1 and VPC 2 that have VPC 3 as a destination first route through transit gateway A, then transit gateway B, and then to VPC 3. **End of image description.**

If you need network traffic to flow between AWS Regions or different AWS accounts, you can create a transit gateway peering connection between transit gateways. You can then route traffic between the attachments for each of the transit gateways.

To provide traffic flow between two transit gateways, create a transit gateway peering attachment on your transit gateway, and specify a target transit gateway. The target transit gateway can be in your account or a different AWS account. After you create a peering attachment request, the owner of the target transit gateway (also referred to as the acceptor transit gateway) must accept the request. To route traffic between the transit gateways, add a static route to the transit gateway route table that points to the transit gateway peering attachment.

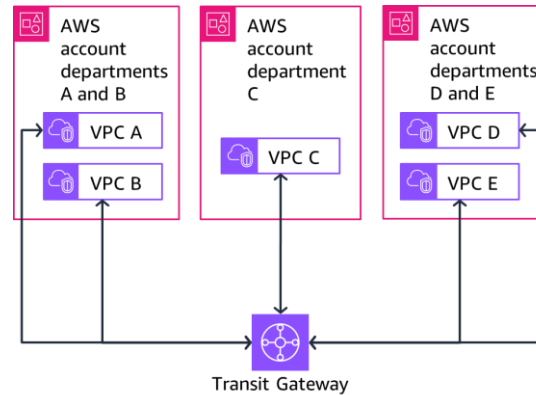
In the example on this slide, CIDR blocks for each VPC propagate to the transit gateway default route tables. The propagated routes are labelled as 1, 2, and 4 in the transit gateway route tables. Each transit gateway route table has a static route that points to the transit gateway peering attachment labelled as 3 in the routing tables.

A peered transit gateway can reside in a different AWS Region or AWS account than the original transit gateway. This provides seamless communication between VPCs located in different AWS accounts or Regions, making data transfers efficient and reliable. Traffic between Regions is secure because it does not traverse the public internet.

## Company group of departments use case

### Scenario:

A company has multiple IT departments, each with their own VPC. Some VPCs are located within the same AWS account, and others in a different AWS account. All the VPCs should be peered together to let the IT departments have full access to each others' resources. The company is considering adding accounts of other business groups in the future.



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

13

The solution that requires the least maintenance and effort for your company is connecting each department's VPC to a transit gateway. This solution would provide adequate capacity for future growth.

Alternatively, if one VPC needs access to only one other VPC, the route tables can be updated to limit traffic to the specified IP address range of the target VPC.



## Activity: Configure AWS Transit Gateway Routes



- Configure routes for five VPCs that you want to connect to each other through Transit Gateway.
- Configure routes in the VPC route tables.
- Configure routes in the transit gateway route table.

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

14

## Configure the VPC route tables

Which routes are necessary to add to each of the VPC route tables to provide full connectivity between all five VPCs?

VPC ID	VPC CIDR	Transit gateway VPC attachment ID	VPC route table ID	VPC route table destination	VPC route table target
vpc-a	10.1.0.0/16	tgw-attach-vpc-a	rtb-vpc-a	?	?
vpc-b	10.2.0.0/16	tgw-attach-vpc-b	rtb-vpc-a	?	?
vpc-c	10.3.0.0/16	tgw-attach-vpc-c	rtb-vpc-a	?	?
vpc-d	10.4.0.0/16	tgw-attach-vpc-d	rtb-vpc-a	?	?
vpc-e	10.5.0.0/16	tgw-attach-vpc-e	rtb-vpc-a	?	?



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

15

For each VPC route table, provide a destination and a target to let traffic flow from the VPC to transit gateway tgw-1. The transit gateway ID is tgw-1. VPC routing to all VPCs can be achieved with CIDR block 10.0.0.0/8.

The following are the definitions of the terms in the table columns:

- VPC ID: This is a unique identifier for each VPC.
- VPC CIDR: This represents the IP address range for the VPC.
- Transit gateway attachment ID: This ID represents the connection between the VPC and the transit gateway.
- VPC route table ID: Every VPC has a route table, and this ID uniquely identifies the route table.
- Destination: This is the IP address range that the route table directs traffic to.
- Target: This is where the traffic is directed to.

## Configure the transit gateway tgw-1 route table

Which routes are necessary to add to the transit gateway tgw-1 route table to provide full connectivity between all five VPCs?

Transit gateway route table destination	Transit gateway route table target
?	?
?	?
?	?
?	?
?	?



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

16

Provide the route destination and target for the transit gateway tgw-1 route table for each VPC in the previous slide to complete your transit gateway route configuration.

The following are the definitions of the terms in the table columns:

- Transit gateway route table destination: This is the destination CIDR to route to.
- Transit gateway route table target: This is the target attachment to route to.

## Solution: VPC route tables

VPC ID	VPC CIDR	Transit gateway VPC attachment ID	VPC route table ID	VPC route table destination	VPC route table target
vpc-a	10.1.0.0/16	tgw-attach-vpc-a	rtb-vpc-a	10.0.0.0/8	tgw-1
vpc-b	10.2.0.0/16	tgw-attach-vpc-b	rtb-vpc-a	10.0.0.0/8	tgw-1
vpc-c	10.3.0.0/16	tgw-attach-vpc-c	rtb-vpc-a	10.0.0.0/8	tgw-1
vpc-d	10.4.0.0/16	tgw-attach-vpc-d	rtb-vpc-a	10.0.0.0/8	tgw-1
vpc-e	10.5.0.0/16	tgw-attach-vpc-e	rtb-vpc-a	10.0.0.0/8	tgw-1



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

17

For each VPC route table, provide a destination and a target to provide traffic flow from the VPC to transit gateway tgw-1. The transit gateway ID is tgw-1. VPC routing to all VPCs can be achieved with CIDR block 10.0.0.0/8. Although 0.0.0.0/0 for all traffic could be used, it is better to limit the routing between VPCs CIDR ranges only.

## Solution: Transit gateway tgw-1 route table

Transit gateway route table destination	Transit gateway route table target
10.1.0.0/16	tgw-attach-vpc-a
10.2.0.0/16	tgw-attach-vpc-b
10.3.0.0/16	tgw-attach-vpc-c
10.4.0.0/16	tgw-attach-vpc-d
10.5.0.0/16	tgw-attach-vpc-e



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

18

The transit gateway route table destination is the CIDR block of the VPC with the target as the transit gateway VPC attachment ID.

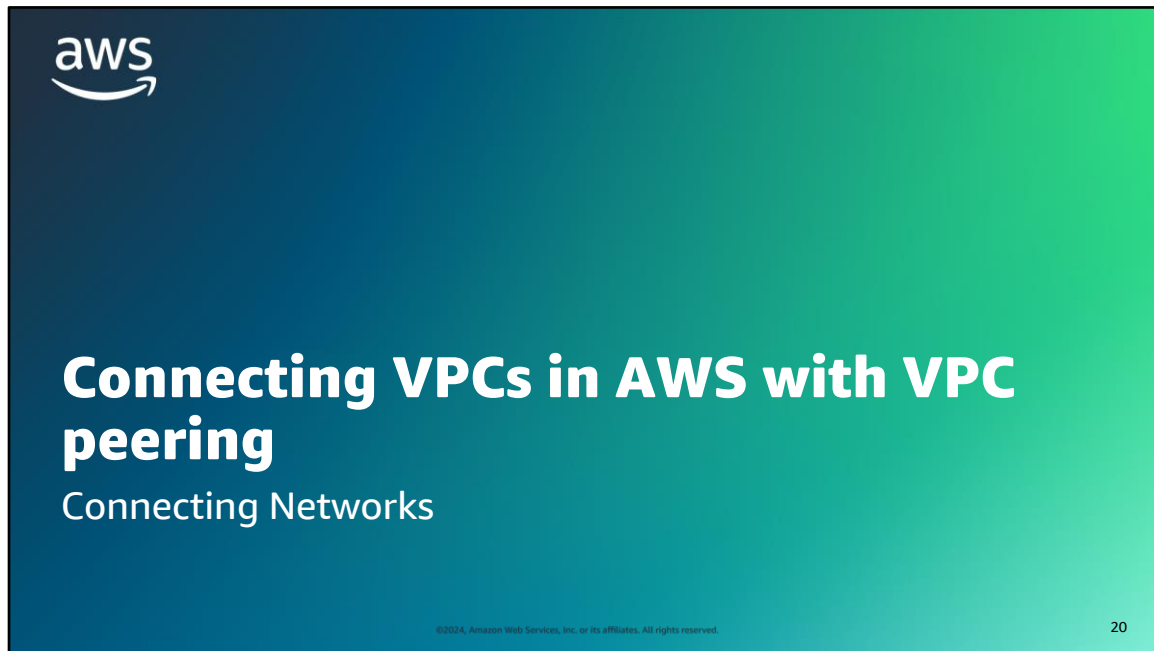
## Key takeaways: Scaling your VPC network with Transit Gateway



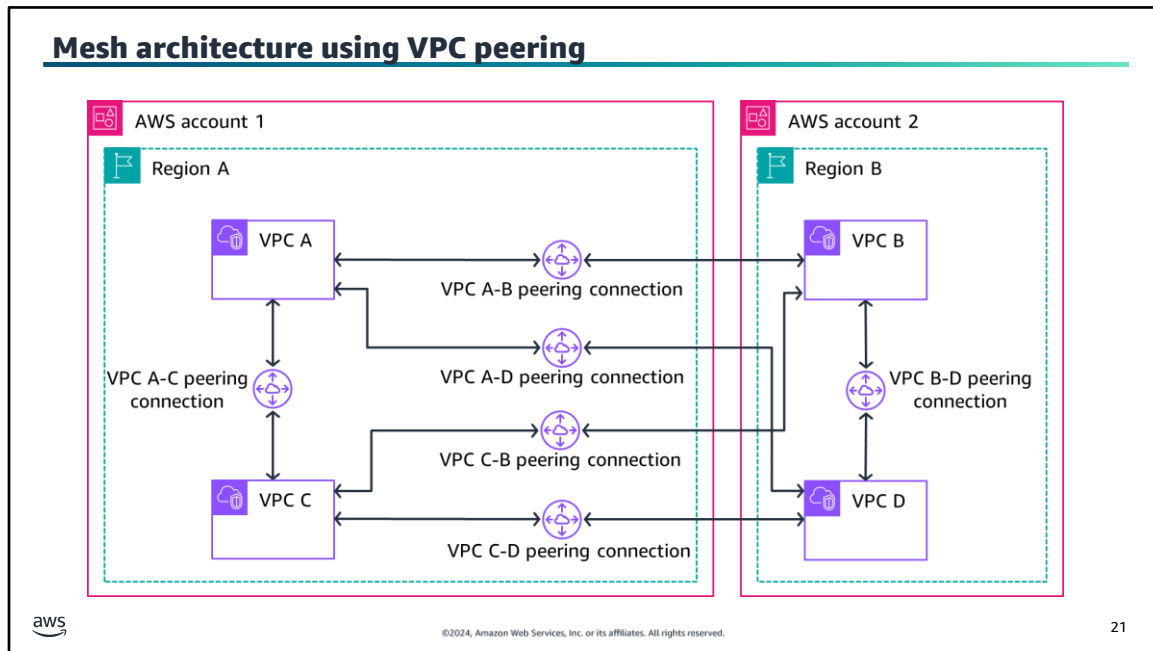
- Transit Gateway is a centralized Regional router to connect VPCs.
- Transit Gateway can be peered with other transit gateways in other AWS Regions and AWS accounts.
- Transit Gateway supports thousands of attachments.
- Transit Gateway charges per hour for the number of connections that you make to the transit gateway and the amount of traffic that flows through the transit gateway.

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

19



This section looks at connecting multiple VPCs in AWS by using the VPC peering feature.



21

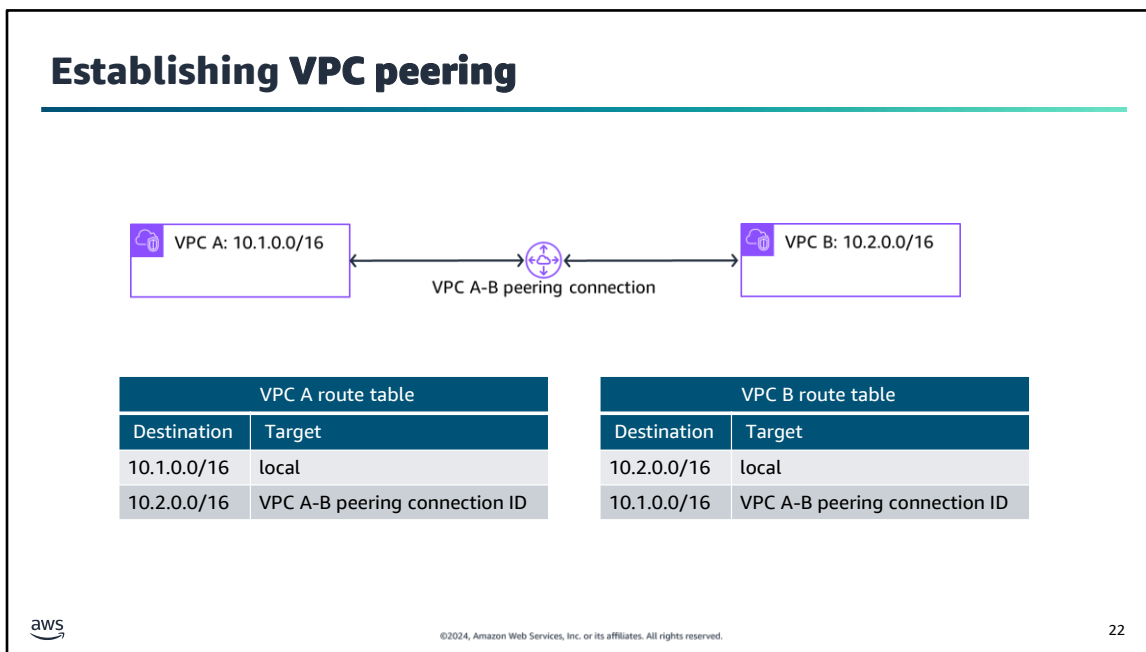
When you have a small number of VPCs or your networking budget is constrained by paying for a transit gateway, you can use the VPC peering feature to establish a one-to-one network connection between two VPCs. VPC peering is a feature of the Amazon Virtual Private Cloud (Amazon VPC) service, and creating a VPC peering connection does not incur any costs. Because it is a point-to-point peer, the network overhead is very small, and the network latency is low.

VPC peering lets Amazon Elastic Compute Cloud (Amazon EC2) instances in two VPCs to communicate with each other by using private IP addresses as if they are in the same network. You can create a VPC peering connection between your own VPCs, with a VPC in another AWS account, or with a VPC in a different AWS Region. Inter-Region VPC peering provides a cost-effective way to share resources between Regions or replicate data for geographic redundancy. Data that is transferred across inter-Region or inter-Availability Zone VPC peering connections is charged at the standard data transfer rates.

VPC peering connections do not share traffic with other VPC peering connections, and the traffic remains in the private IP address space. Traffic also always stays on the global AWS backbone. This means that traffic never traverses the public internet, which reduces threats, such as common exploits and DDoS attacks. All inter-Region traffic is encrypted, and there is no single point of failure or bandwidth bottlenecks.

A VPC peering connection helps you to facilitate the transfer of data. For example, if you have more than one AWS account, you can peer the VPCs across those accounts to create a file-sharing network. You can also use a VPC peering connection to give other VPCs the ability to access resources that you have in one of your VPCs.



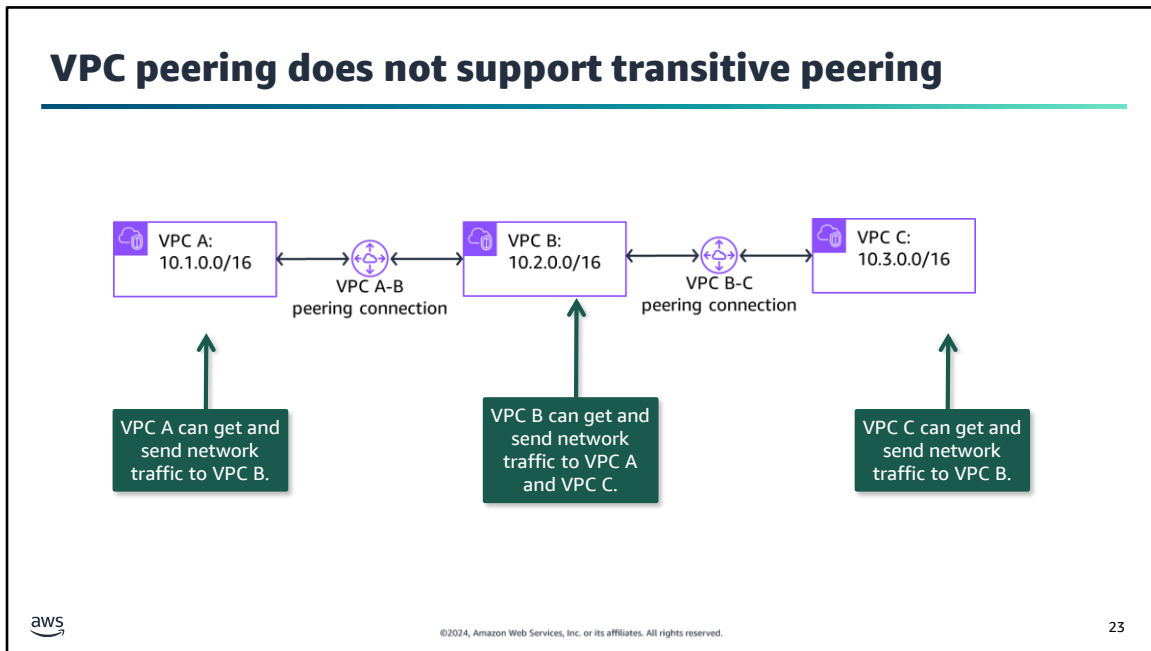


In the example on this slide, a VPC peering connection between VPC A and VPC B is required. The owner of VPC A sends a peering request to the owner of VPC B. To activate the peering connection, the owner of VPC B must accept the VPC A peering connection request. The CIDR block of VPC A cannot overlap with the CIDR block of VPC B.

To provide the flow of the traffic between the VPC peers by using private IP addresses, the owner of VPC A must add a route to the VPC A route table. This route destination is the IP address range of VPC B 10.2.0.0/16. The target of the route is the VPC A-B peering connection ID. The owner of VPC B should also add a route to the VPC B route table. This route destination is the IP address range of VPC A 10.1.0.0/16. The target of the route is the VPC A-B peering connection ID.

The owners of VPC A and VPC B might also need to update the security group rules that are associated with the VPC instances so that traffic to and from the peer VPC is not restricted.

A practical application of VPC peering would be two startups collaborating on a project. They can establish VPC peering to share resources securely. This setup lets them to communicate over the private network infrastructure of AWS, helping to ensure faster and more secure data exchange without exposing their resources to the public internet.

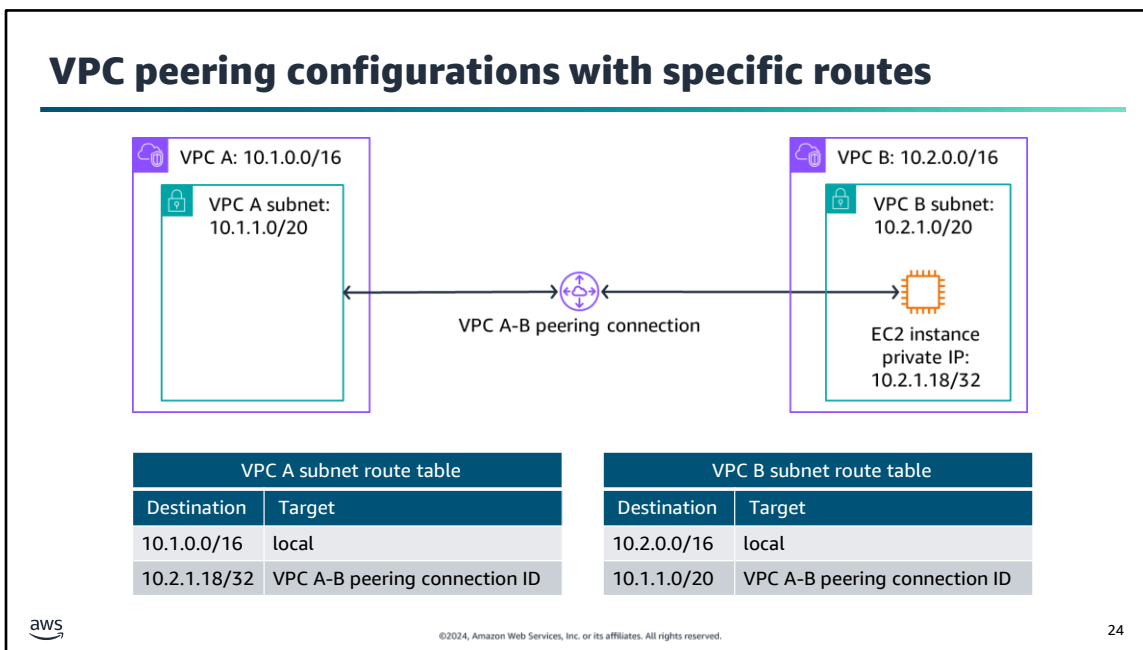


Transitive peering is not supported by VPC peering. This makes peering secure and manageable to isolate errors and limit the blast radius for network attacks.

For example, in the diagram, VPC A and VPC B are peered, and VPC B and VPC C are peered. However, this does not mean that the VPC A is connected to VPC C. By default, VPC peering does not let VPC A to connect to VPC C unless they are explicitly established as peers. Therefore, you control which VPCs can communicate with each other and which VPCs should be isolated with minimal access.

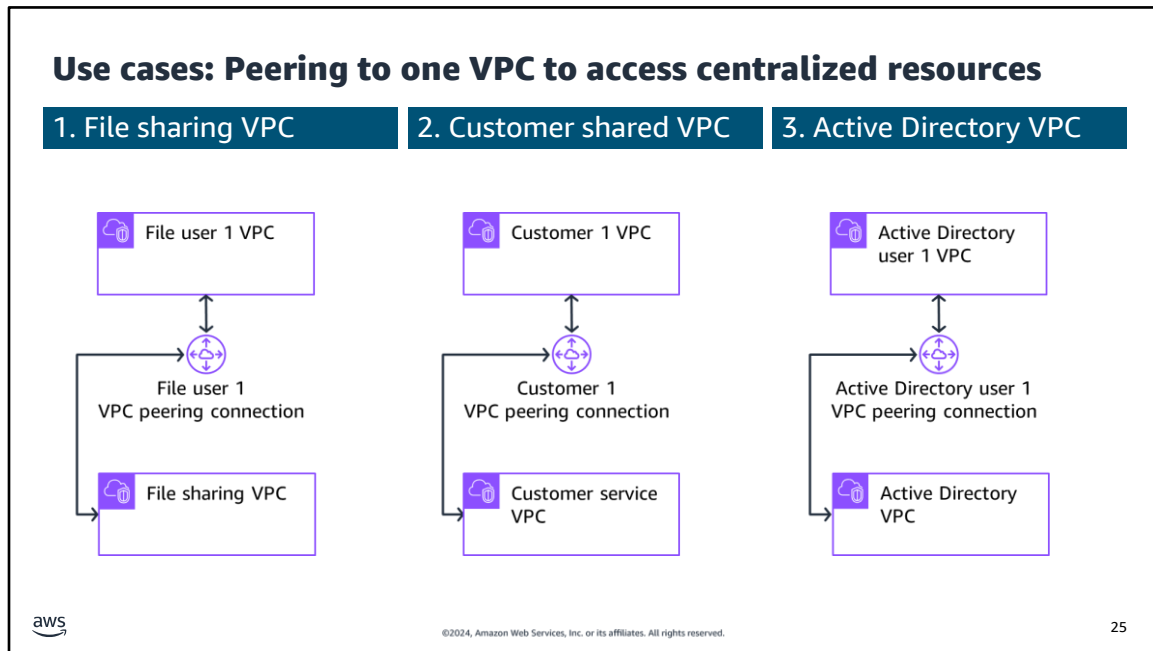
The following are other limitations for a VPC peering connection:

- VPC peering is not possible between VPCs that have matching or overlapping CIDR blocks.
- If either VPC has an internet or NAT gateway that is in a peering connection, the VPC does not have access to the internet or NAT gateway residing in the other VPC.
- Only VPC owners can work with their VPC peering connections.



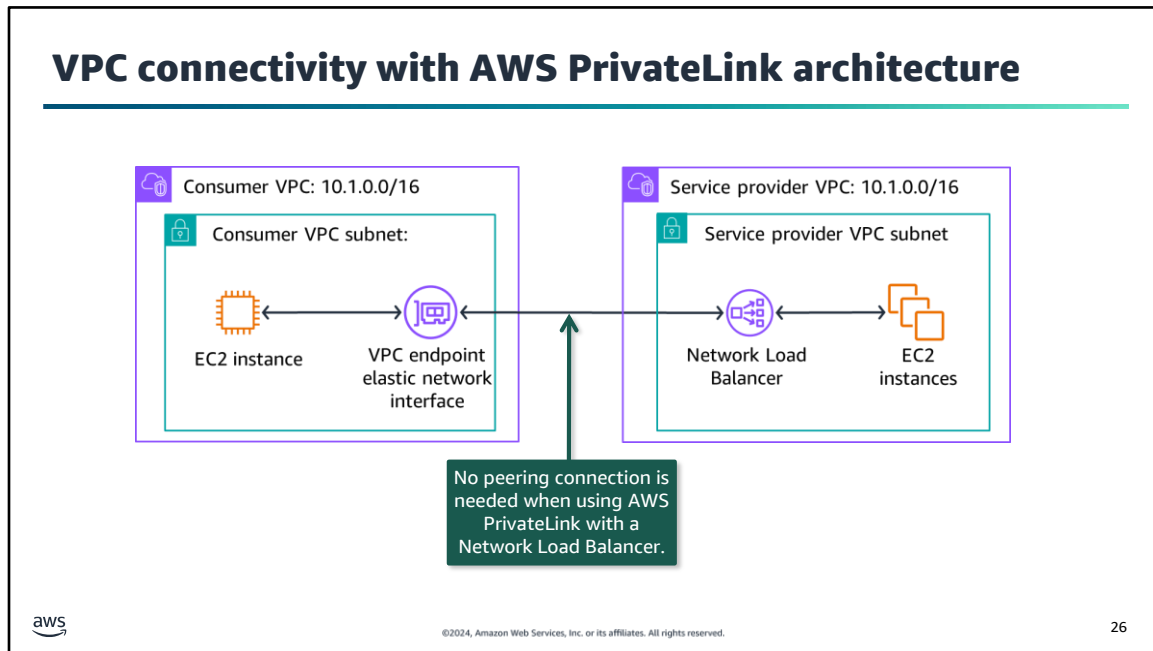
You can configure route tables for a VPC peering connection to restrict access to a subnet CIDR block, a specific CIDR block if the VPC has multiple CIDR blocks, or a specific resource IP address in the peer VPC.

In the example on this slide, the VPC A subnet route table has a route defined to the EC2 instance in VPC B. The route table destination is the private IP of the EC2 instance 10.2.1.18/32 with a target of the VPC A-B peering connection. The VPC B subnet route table has a route defined to the VPC A subnet. The route table destination is the VPC A subnet CIDR block of 10.1.1.0/20 with a target of the VPC A-B peering connection.



The following are examples of a central VPC that provides a service for other VPCs to access:

1. Your company's IT department has a VPC for file sharing. You want to peer other VPCs to that central VPC; however, you do not want the other VPCs to send traffic to each other.
2. Your company has a VPC that you want to share with your customers. Each customer can create a VPC peering connection with your VPC; however, your customers cannot route traffic to other VPCs that are peered to yours, nor are they aware of the other customers' routes.
3. You have a central VPC that is used for Active Directory services. Specific instances in peer VPCs send requests to the Active Directory servers and require full access to the central VPC. The central VPC does not require full access to the peer VPCs; it needs only to route response traffic to the specific instances.



Up to this point, you have connected VPCs at the network level, but what if you want to connect VPCs on an application level? Alternatively, what if you want to connect to a VPC with overlapping IP address ranges?

You can use AWS PrivateLink to privately connect to a service or application that resides in a service provider VPC from consumer VPCs within an AWS Region. The benefit of this architecture is that only consumer VPCs initiate connections to the service provider VPC.

In the example on this slide, the service provider VPC has a Network Load Balancer with EC2 instances as targets. The owner of the consumer VPC creates a VPC elastic network interface endpoint to the network load balancer endpoint. In this scenario, the consumer and service provider VPCs can have overlapping IP address ranges of 10.1.0.0/16.

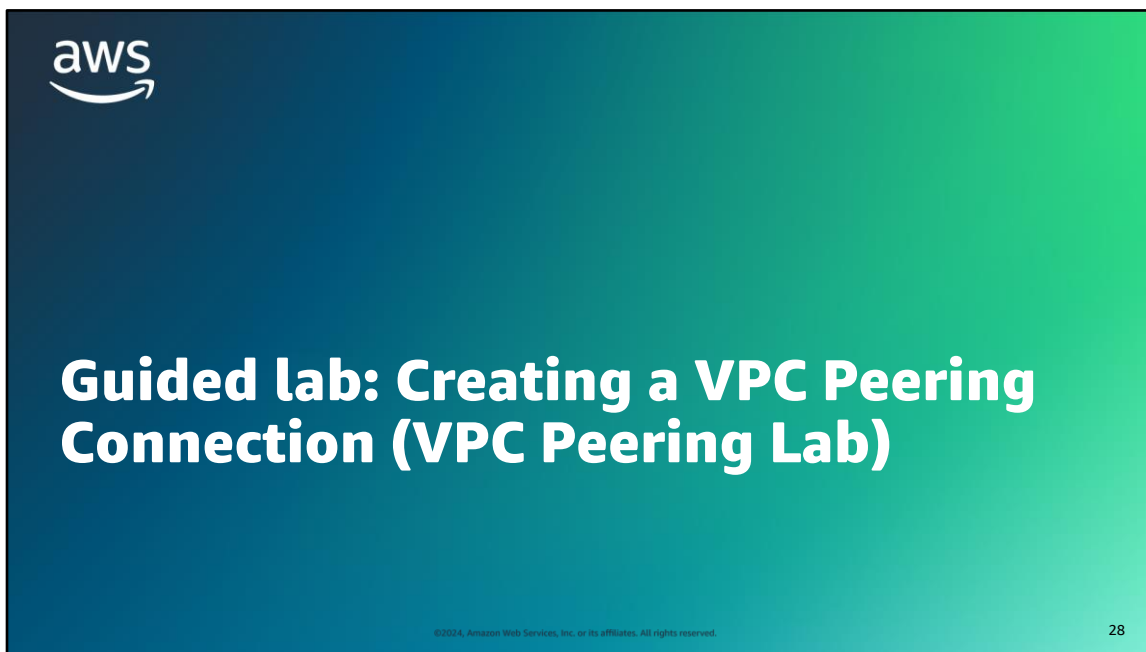
## Key takeaways: Connecting VPCs in AWS with VPC peering



- VPC peering establishes a one-to-one peering networking connection between two VPCs to provide private network traffic routes.
- VPC peering does not incur costs, but transferring data across Availability Zones and Regions does.
- VPC peering can provide network traffic flow between different AWS accounts and AWS Regions.
- VPC peering does not support transitive VPC peering relationships.
- If VPC Classless Inter-Domain Routing (CIDR) blocks overlap, use PrivateLink with a Network Load Balancer to establish VPC connectivity.



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

27



You will now complete a lab. The next slides summarize what you will do in the lab, and you will find the detailed instructions in the lab environment.

## VPC Peering Lab tasks



- In this lab, you will perform the following main tasks:
  - Create a peering connection between two VPCs.
  - Configure route tables to send traffic to the peering connection.
  - Test the peering connection.
- Open your lab environment to start the lab and find additional details about the tasks that you will perform.

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

29

Access the lab environment through your online course to get additional details and complete the lab.



## Debrief: VPC peering lab

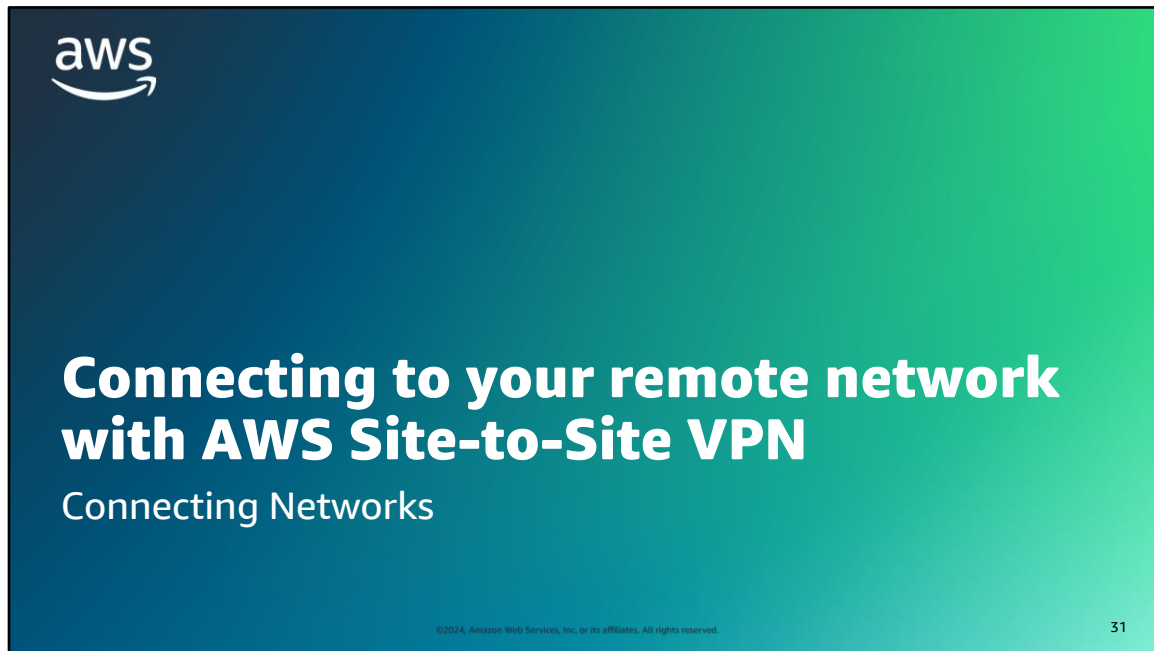
---

- What did you learn about the VPC peering connection process?
- How were you able to confirm that the VPC peering worked?

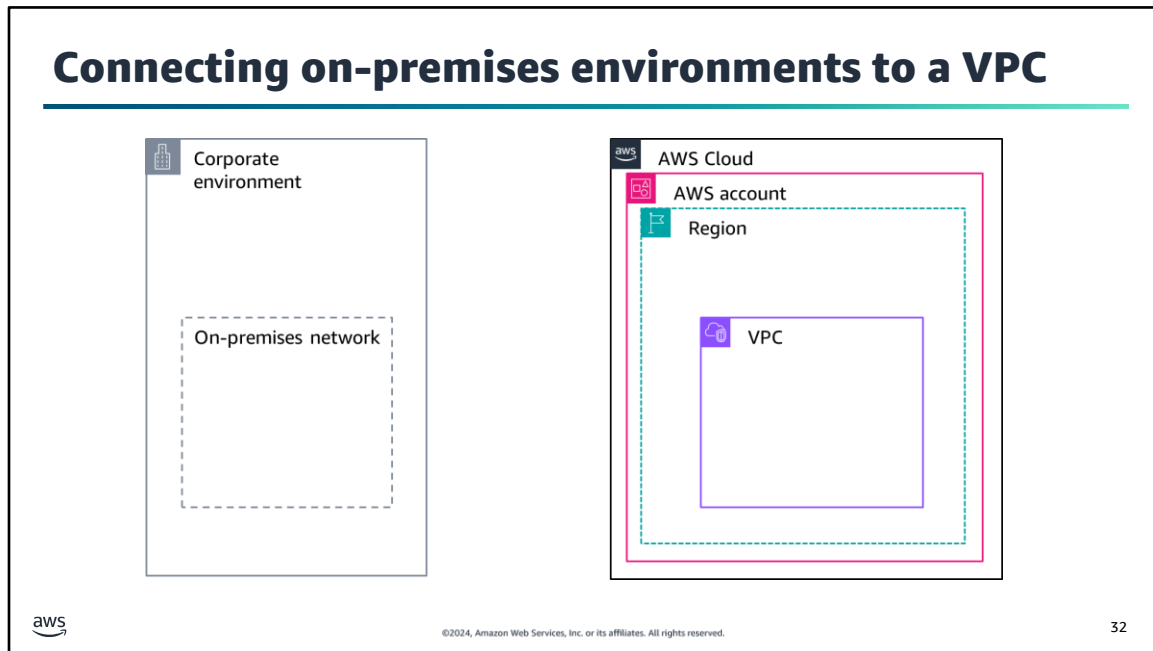


©2024, Amazon Web Services, Inc. or its affiliates. All rights reserved.

30



This section looks at how to connect to an Amazon virtual private cloud (VPC) from an on-premises remote network.



Now that you can connect VPCs to each other, how can you connect to your VPC from an on-premises corporate environment? By default, instances that you launch into a VPC on AWS cannot communicate with your on-premises network.

## AWS Site-to-Site VPN



Site-to-Site VPN

- Creates a secure connection between an on-premises customer gateway and AWS virtual private gateway (or transit gateway) for VPC access
- Creates two encrypted IPsec VPN tunnels for each connection across multiple Availability Zones
- Charges for each VPN connection-hour



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

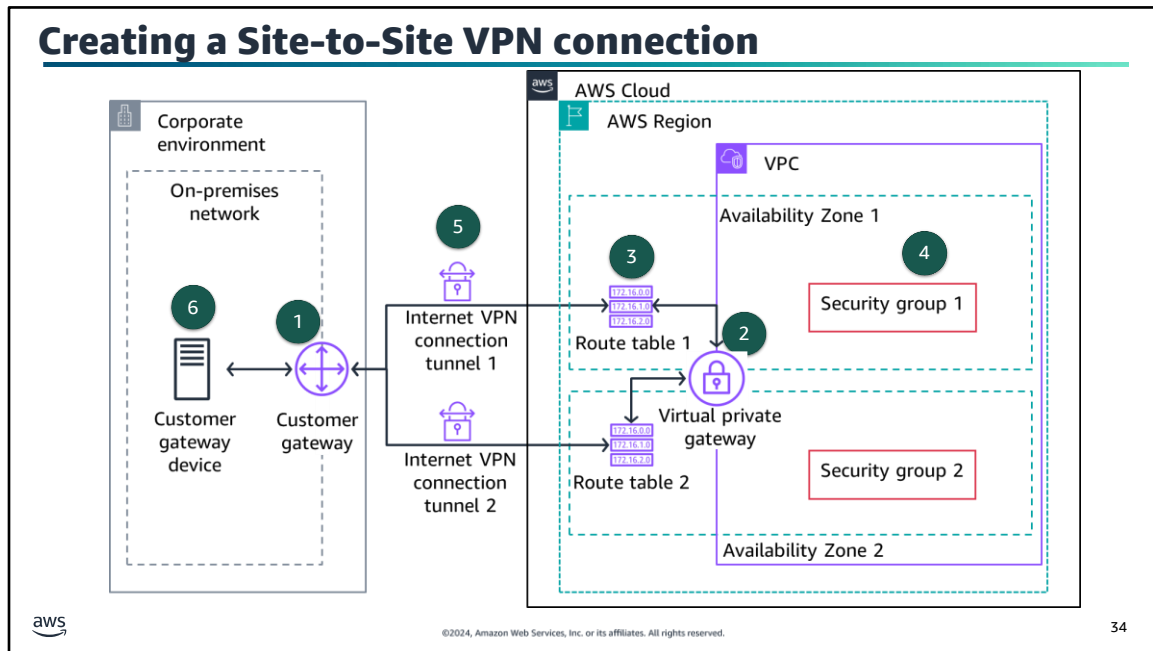
33

You can use AWS Site-to-Site VPN to securely connect your on-premises environment to your VPC. Each Site-to-Site VPN connection uses internet protocol security (IPsec) connections to create encrypted virtual private network (VPN) tunnels between two locations. A VPN tunnel is an encrypted link where data can pass from the customer network to or from AWS over the public internet. Site-to-Site VPN can be set up relatively quickly and can be available in hours.

The on-premises side of the connection is the customer gateway. The AWS side of the connection is the virtual private gateway. Instead of a virtual private gateway, you can also create a Site-to-Site VPN connection as an attachment on a transit gateway.

A Site-to-Site VPN connection provides two VPN tunnels across multiple Availability Zones that you can use simultaneously for high availability. You can stream primary traffic through the first tunnel and use the second tunnel for redundancy. If one tunnel goes down, traffic will still get delivered to your VPC.

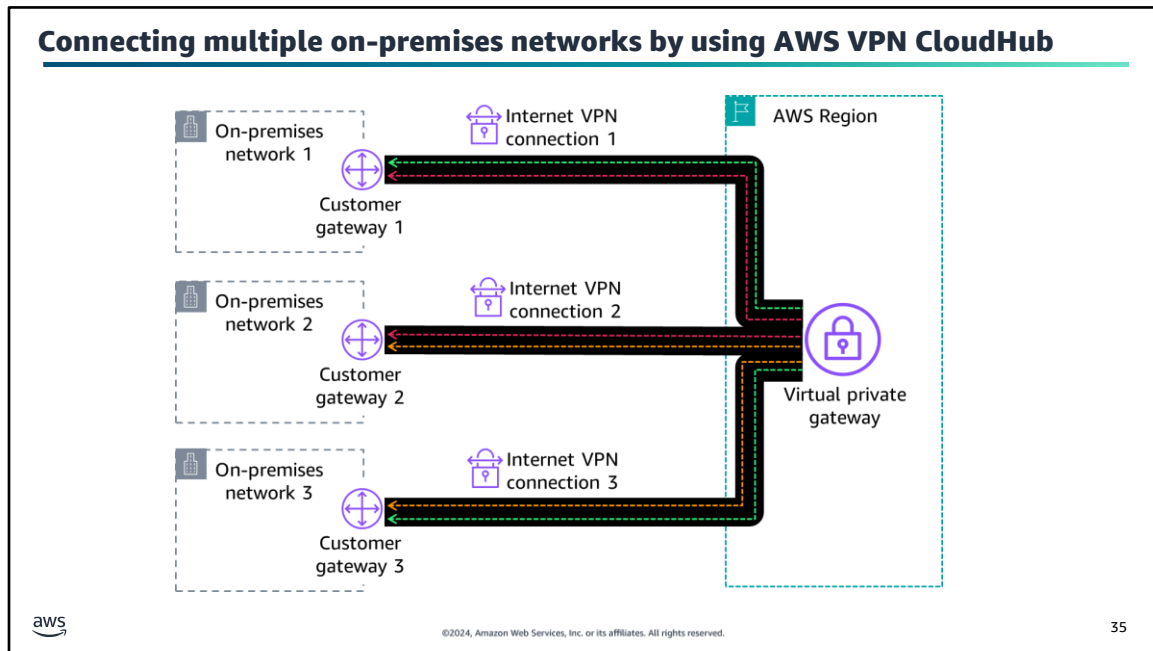
If you create a Site-to-Site VPN connection to your VPC, you are charged for each VPN connection-hour that your VPN connection is provisioned and available.



To create a Site-to-Site VPN connection, complete the following steps:

1. Create a VPC customer gateway. When you create a Site-to-Site VPN connection from your on-premises network to your VPC, a VPC customer gateway represents the customer gateway device. The customer gateway device can be a physical device or a software application residing in the on-premises network. Assign a Border Gateway Protocol (BGP) Autonomous System Number (ASN) to the customer gateway if the customer gateway device supports BGP.
2. Create a virtual private gateway and specify a custom private BGP ASN or use the Amazon default ASN. This ASN must be different from the ASN that you specified for the customer gateway. You can use Transit Gateway as an alternative solution instead of the virtual private gateway.
3. Configure routing to let instances in your VPC to reach your customer gateway. You must configure your route table to include the routes that your VPN connection uses and point them to your virtual private gateway. You can activate route propagation for your route table to automatically propagate Site-to-Site VPN routes. In the example on this slide, each VPN connection tunnel is routed to a different Availability Zone to help ensure high availability.
4. Update security groups to allow Secure Shell Protocol (SSH), Remote Desktop Protocol (RDP), Internet Control Message Protocol (ICMP), or other desired protocols access from the on-premises network.
5. Create the Site-to-Site VPN connection to contain the customer gateway, the virtual private gateway, and the VPN connection with two tunnels connecting to separate Availability Zones. It is important to note that the VPN uses internet connectivity for the VPN connection tunnels and is not guaranteed to be available.
6. Download the configuration file and use it to configure the customer gateway device.

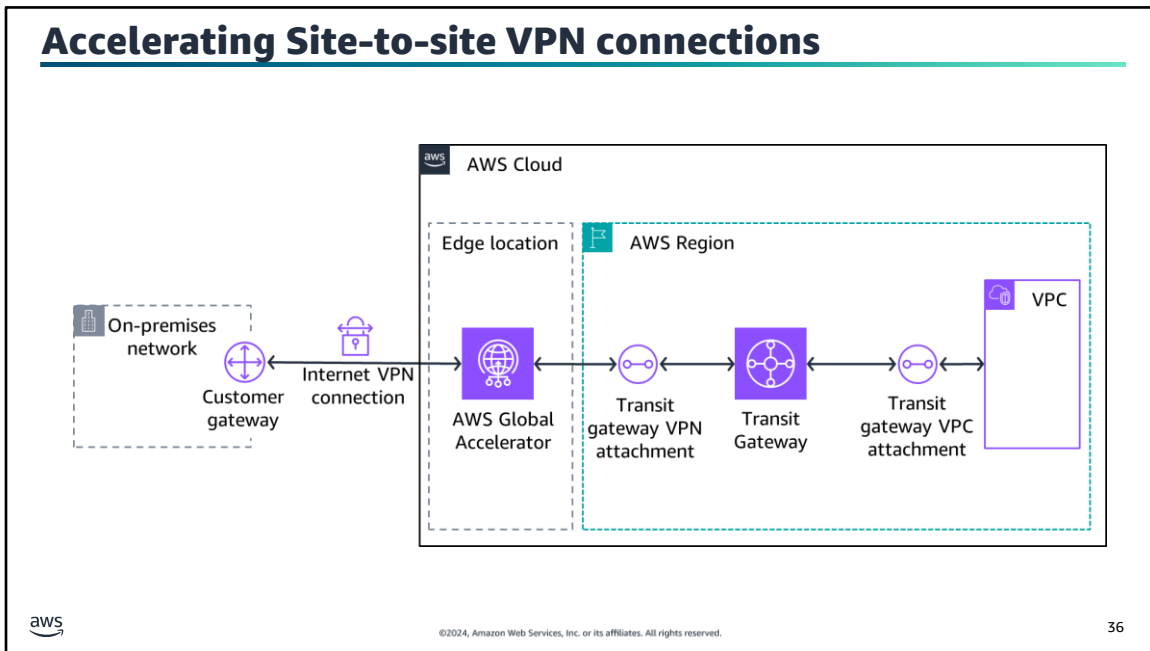
The VPN supports dynamic and static routing depending on the make and model of the customer gateway device. If the device supports BGP, specify dynamic routing. Dynamic routing uses the BGP to advertise routes to the virtual private gateway. If the device does not support BGP, specify static routing. Static routing requires that you specify the routes with IP prefixes for your network that should be communicated to the virtual private gateway. AWS will provide the required connection and tunnel configuration information for the specified customer gateway device. AWS recommends BGP-capable devices because BGP offers robust health checks to assist failover to another VPN connection tunnel.



Large corporate organizations usually have multiple on-premises network environments that can be physically located at different premises far apart. If the environments need primary or backup connectivity to each of the other environments, you need a centralized hub to facilitate connectivity.

Building on the AWS managed VPN options described previously, you can securely communicate from one environment to another by using the AWS VPN CloudHub. The AWS VPN CloudHub operates on a simple hub-and-spoke model that you can use with or without the Amazon VPC service. In the example on this slide, internet VPN connection 1 carries traffic from on-premises network 1 to on-premises networks 2 and 3 and back. Similarly, internet VPN connections 2 and 3 carry traffic to and from the other on-premises networks.

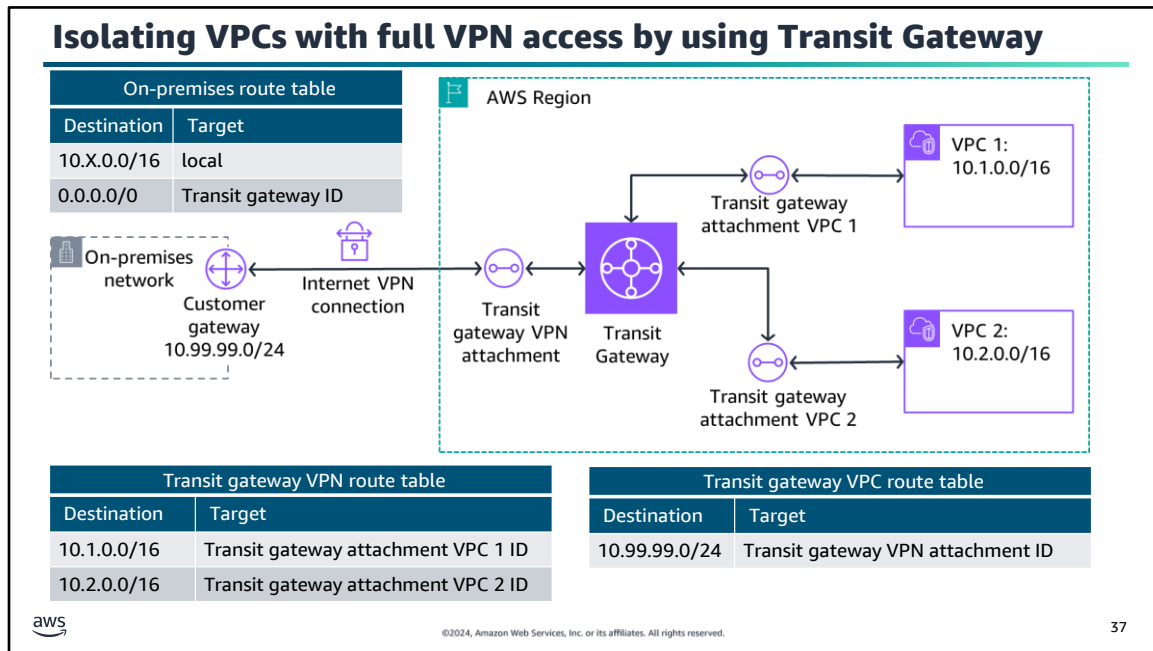
AWS VPN CloudHub uses a virtual private gateway with multiple customer gateways, each of which uses unique BGP ASNs. The customer gateways advertise the appropriate routes over their VPN connections. These routing advertisements are received and re-advertised to each BGP peer so that each site can send data to and receive data from the other environments. The remote sites must not have overlapping IP ranges.



Because public internet traffic can potentially have network disruptions, you can use AWS Global Accelerator to accelerate your Site-to-Site VPN connection. An accelerated VPN connection uses Global Accelerator to route traffic from your on-premises network to an AWS edge location that is closest to your customer gateway device. Network traffic is now using the AWS backbone infrastructure to efficiently route traffic from the edge location to the transit gateway with better response times. Note that acceleration is supported only for VPN connections that are attached to a transit gateway and not a virtual private gateway.

In the example on this slide, the on-premises network has a VPN connection that connects to Global Accelerator. Global Accelerator is connected to the Transit Gateway through the transit gateway VPN attachment. The Transit Gateway connects to the VPC through the Transit Gateway VPC attachment.

To test the difference between public internet speeds and Global Accelerator from various Regions to your location, see the AWS Global Accelerator Speed Comparison in resource links.



So far, you have learned how to create a network that is designed for full sharing of all resources in all the connected environments. What about a requirement that the on-premises network have full access to the VPCs but that the VPCs be isolated from each other?

Fortunately, you can configure your transit gateway as multiple isolated routers. This is similar to using multiple transit gateways but provides more flexibility in cases where the routes and attachments might change. For one transit gateway, you can create multiple route tables. Each transit gateway attachment is associated with a route table. Attachments associated with one isolated routing table can route packets to each other but cannot route packets to or receive packets from the attachments for another isolated router.

In the example on this slide, each VPC has a route table with the transit gateway ID as a target for all traffic. The transit gateway has two route tables. The transit gateway VPN route table is associated with the transit gateway VPN attachment. The transit gateway VPC route table is associated with the transit gateway VPC attachments. Network traffic from VPC 1 and VPC 2 that is destined for the on-premises network routes to the transit gateway and then to the internet VPN connection to arrive at the on-premises network. Network traffic from VPC 1 that has a destination of a subnet in VPC 2 10.2.0.0 routes through the transit gateway, where it is blocked because there is no route in the transit gateway VPC route table.



## Key takeaways: Connecting to your remote network with Site-to-Site VPN



- Site-to-Site VPN creates a secure connection between an on-premises customer gateway and an AWS virtual private gateway (or transit gateway) for VPC access.
- You can connect multiple on-premises networks to a single virtual private gateway.
- Use Global Accelerator to accelerate Site-to-Site VPN connections.
- Configure multiple Transit Gateway routing tables to isolate VPCs to provide full VPN access.


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

38



This section looks at how to connect to an Amazon virtual private cloud (VPC) from an on-premises remote network by using AWS Direct Connect.

## AWS Direct Connect



Direct Connect

- Is a dedicated, private, virtual local area network (VLAN) connection that extends on-premises network to include AWS resources
- Provides a consistent network experience with predictable performance and increased bandwidth and throughput

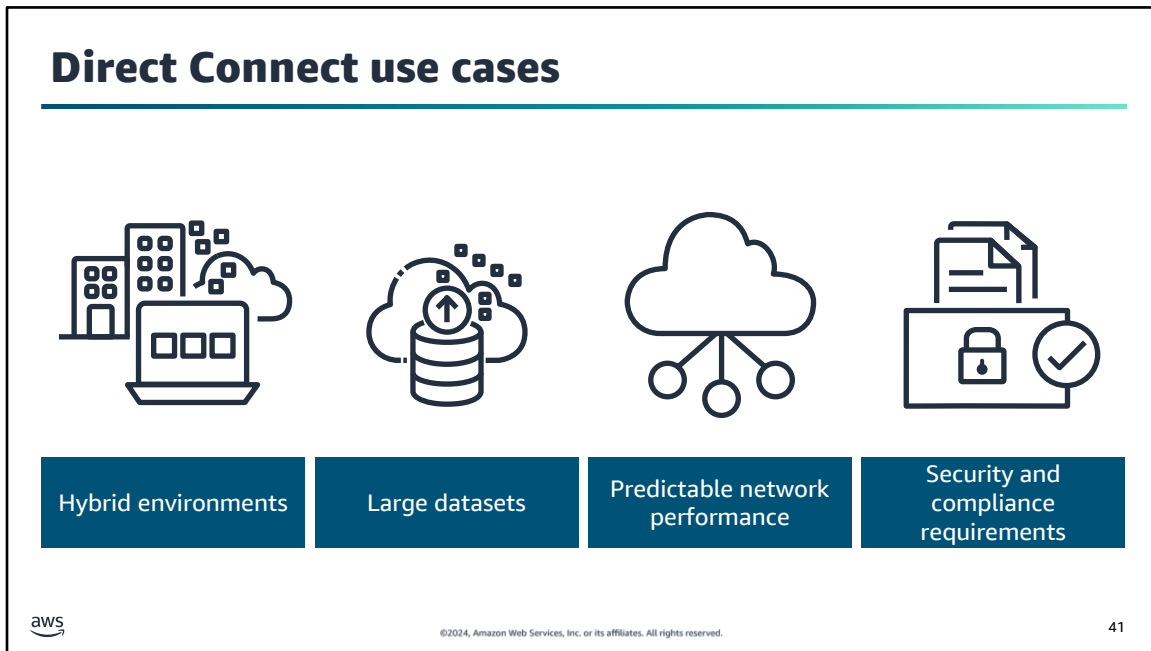
aws

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

40

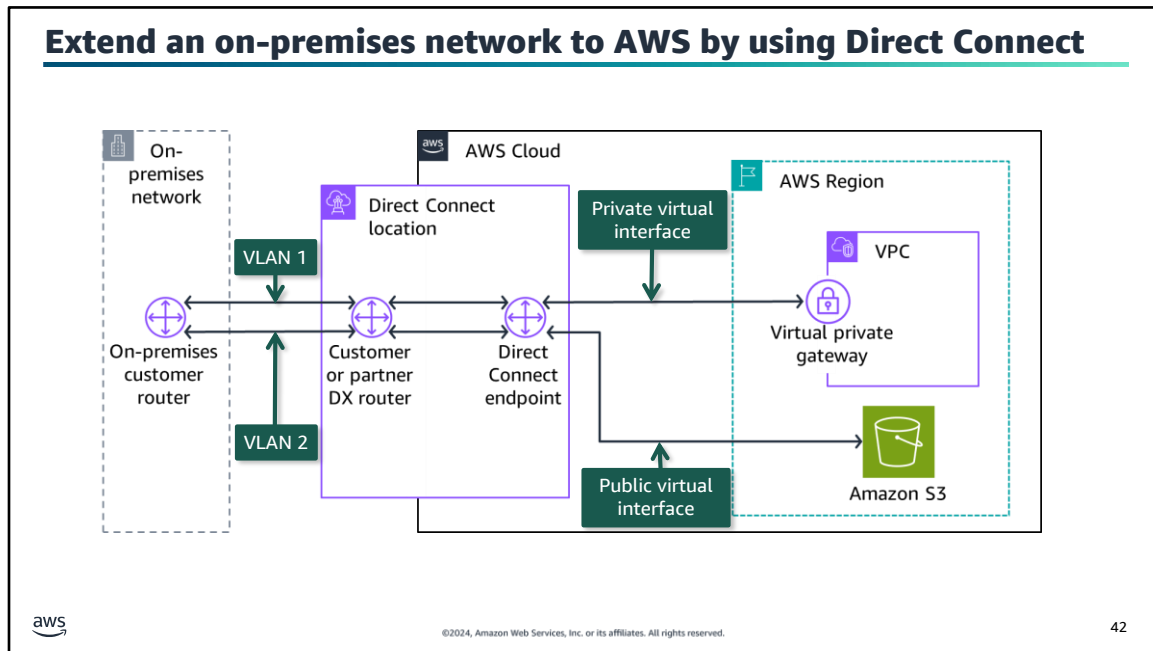
As you learned, Site-to-Site VPN is one option for connecting your on-premises network to the AWS global network. Data is transferred through encrypted tunnels over the public internet.

Direct Connect is another solution that goes beyond connectivity over the internet. Direct Connect uses virtual local area networks (VLANs) so that you can establish a dedicated, private network connection from your on-premises network to AWS. This private connection extends your private network to include AWS resources. The benefits of using Direct Connect are that it increases bandwidth throughput and provides a more consistent network experience than internet-based connections.



Direct Connect is useful for several scenarios, such as the following:

- For applications that require access to existing data center equipment, such as an on-premises database, Direct Connect gives you the ability to create a hybrid environment so that you can take advantage of the elasticity and economic benefits of AWS.
- For applications that operate on large datasets, such as high performance computing (HPC) applications, transferring large datasets over the internet between your data center and the AWS Cloud can be time consuming and expensive. For such applications, connecting to the AWS Cloud by using Direct Connect is a good solution for the following reasons:
  - Network transfers will not compete for internet bandwidth at your data center.
  - The high-bandwidth link reduces the potential for network congestion and degraded application performance.
  - By limiting the internet bandwidth that your applications use, you can reduce network fees that you pay to your internet service provider (ISP) and avoid having to pay for increased internet bandwidth commitments or new contracts. In addition, all data that is transferred over Direct Connect is charged at the reduced Direct Connect data transfer rate instead of internet data transfer rates, which can reduce your network costs.
- For applications that require predictable network performance, Direct Connect is a good solution. Examples include applications that operate on real-time data feeds, such as audio or video streams. In such cases, a dedicated network connection can provide more-consistent network performance than standard internet connectivity.
- For enterprise security or regulatory policies which sometimes require that applications hosted on the AWS Cloud should be accessed only through private network circuits. Direct Connect is a solution to this requirement because traffic between your data center and your application flows through the dedicated private network connection.



42

Direct Connect (DX) makes use of Direct Connect locations to extend your on-premises network to AWS by using industry-standard 802.1Q VLANs. By default, these locations are associated with an AWS Region. However, you can access any VPC or public AWS service in any Region (except China) from any supported Direct Connect location.

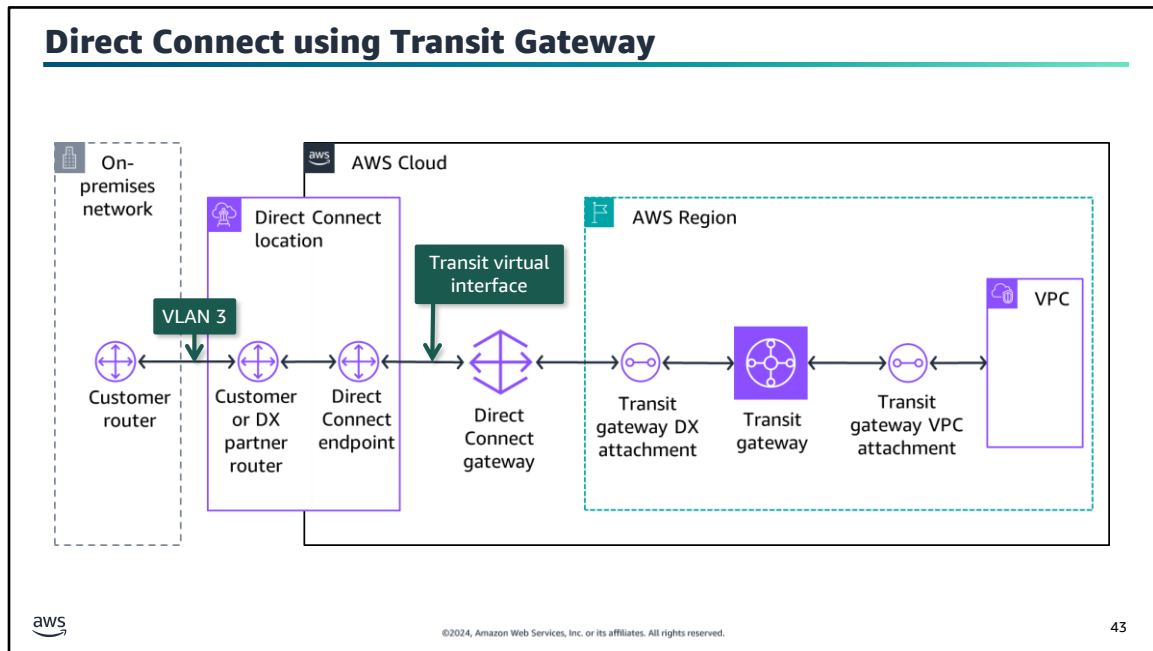
With Direct Connect, you have a choice of two types of connections to the Direct Connect location: dedicated or hosted connections. A dedicated connection is where a physical ethernet fiber-optic cable is provisioned for exclusive use by the customer. The on-premises customer router will be connected by the ethernet connection to the customer router in the Direct Connect location. A hosted connection is where a physical ethernet fiber-optic cable is provisioned by a Direct Connect Partner and shared with the customer. This provides a customer with more bandwidth options in smaller increments than a dedicated connection. The customer Direct Connect router is connected to a Direct Connect endpoint in the Direct Connect location to connect to AWS services, including VPCs. A Direct Connect connection takes planning and physical resources, so the implementation time typically spans multiple weeks.

With a Direct Connect connection, you can create three types of virtual interfaces: public, private, and transit virtual interfaces:

- A public virtual interface provides access to public AWS services, such as Amazon S3. In the example on this slide, the public virtual interface is designated by green arrows.
- A private virtual interface provides access to your VPC by using a virtual private gateway.
- A transit virtual interface provides access to your VPC by using a transit gateway.

In the example on this slide, the VLAN 1 connection is established from the on-premises customer router to the Direct Connect location customer Direct Connect router and Direct Connect endpoint. The Direct Connect endpoint connects to a VPC virtual private gateway through a private virtual interface. The VLAN 2 connection is established from the on-premises customer router to the Direct Connect location customer Direct Connect router and Direct Connect endpoint. The Direct Connect endpoint connects to Amazon S3 through a public virtual interface.

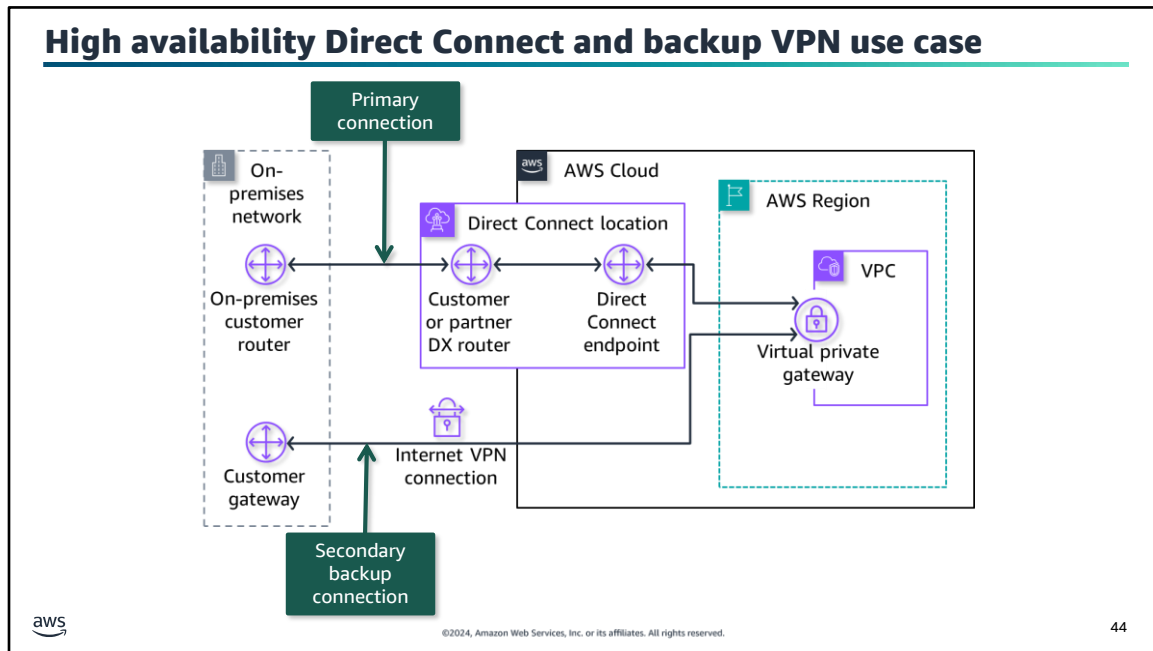
For more information, see AWS Direct Connect Locations on the content resources page of your online course.



43

You can also use Direct Connect to connect to a Transit Gateway instead of a virtual private gateway to simplify routing between the on-premises network and multiple VPCs. The transit gateway has a Transit Gateway Direct Connect attachment pointing to a Direct Connect gateway. You can use the Direct Connect gateway to connect to multiple Transit Gateways. The solution uses a transit virtual interface to communicate from the Direct Connect location to the Direct Connect gateway.

In the example on this slide, the VLAN 3 connection is established from the on-premises customer router to the Direct Connect location customer Direct Connect router and Direct Connect endpoint. The Direct Connect endpoint connects to a Direct Connect gateway endpoint through a transit virtual interface. The Direct Connect gateway is connected to the transit gateway through the transit gateway Direct Connect attachment. The Transit Gateway is connected to the VPC through the transit gateway VPC attachment.

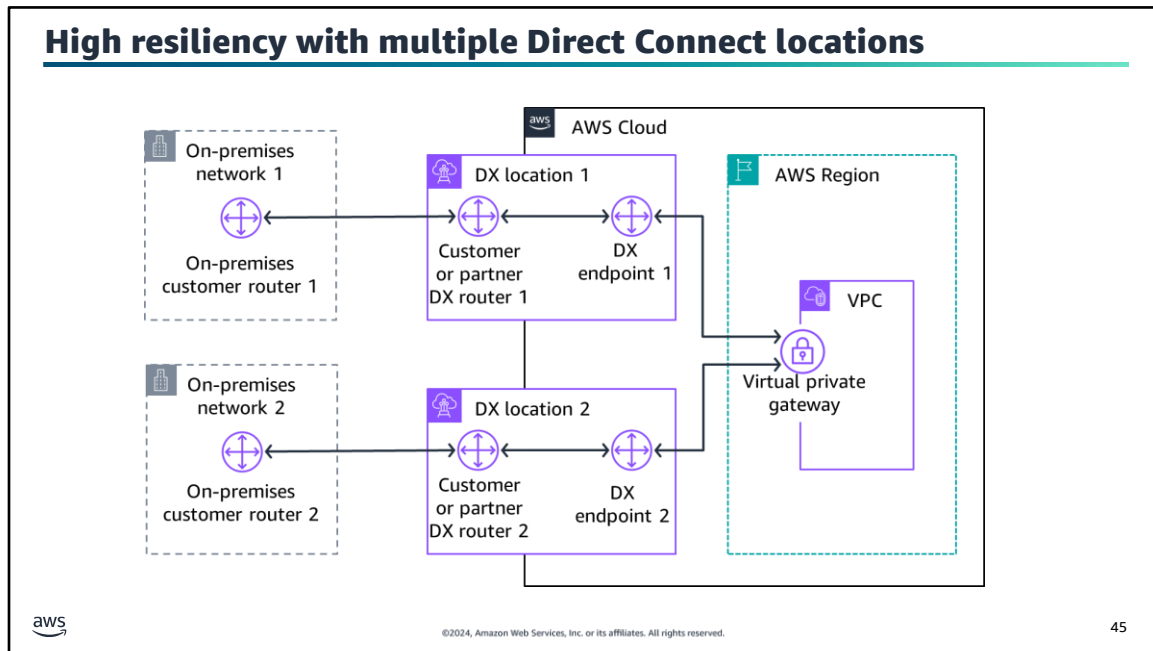


You can implement highly available connectivity between your data centers and your VPC by coupling one or more Direct Connect connections that you use for primary connectivity with a lower-cost backup VPN connection.

In this example, the configuration consists of two dynamically routed connections—one that uses Direct Connect and another that uses a VPN connection—from two different customer devices. AWS provides example router configurations to help you establish both Direct Connect and dynamically routed VPN connections. By default, AWS will always prefer to send traffic over your Direct Connect connection, so you do not need additional configurations specific to AWS to define primary and backup connections. However, you should configure Direct Connect and VPN-specific internal-route propagation to help ensure that internal systems select the appropriate paths.

With this approach, you can choose the primary network path and network provider for your AWS traffic, with the option of using a different provider for a backup VPN connection. Choose network providers and Direct Connect locations that align with your organization's risk tolerance, financial expectations, and data center connectivity policies.

There are many ways to increase your network availability. You can use multiple Direct Connect circuits and multiple VPN tunnels between separately deployed private IP address spaces. You can also use multiple Direct Connect locations for high availability. If you use multiple AWS Regions, you will also need multiple Direct Connect locations in at least two Regions. You might want to evaluate AWS Marketplace appliances that shut down VPNs.



45

For critical production workloads that require high resiliency, AWS recommends that you have one connection at multiple locations. As this architecture diagram shows, such a topology helps ensure resilience against connectivity failures due to a hardware failure or a complete location failure. In the example on this slide, on-premises network 1 connects to the virtual private gateway through Direct Connect location 1. If Direct Connect location 1 fails or there are issues with on-premises network 1, then on-premises network 2 can still connect to the virtual private gateway through Direct Connect location 2.

Highly resilient, fault-tolerant network connections are key to a well-architected system. AWS recommends connecting from multiple data centers for physical location redundancy. When you design remote connections, consider using redundant hardware and telecommunications providers.

Additionally, it is a best practice to use dynamically routed, active/active connections for automatic load balancing and failover across redundant network connections. Provision sufficient network capacity to help ensure that the failure of one network connection does not overwhelm and degrade redundant connections.



## Key takeaways: Connecting to your remote network with Direct Connect



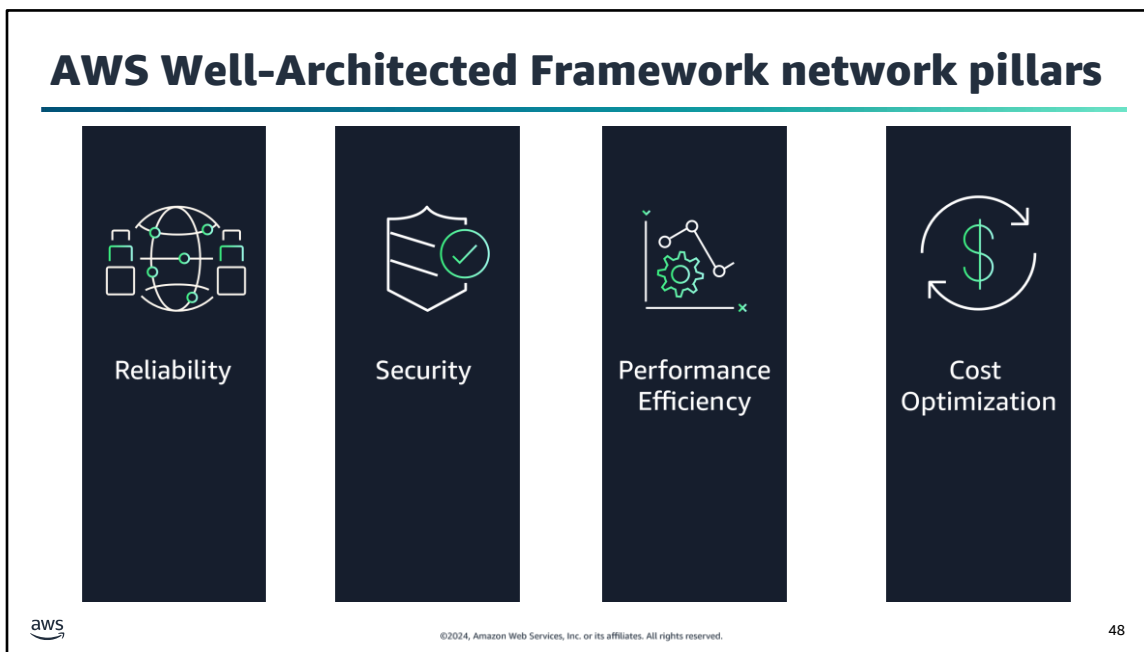
- Direct Connect is a dedicated, private, VLAN connection that extends an on-premises network to include AWS resources.
- Use a private virtual interface to connect a Direct Connect location to a virtual private gateway.
- Use a public virtual interface to connect a Direct Connect location to supported AWS services.
- Use a transit virtual interface to connect a Direct Connect location to a transit gateway through a Direct Connect gateway.
- Make your network highly available by using Direct Connect as a primary connection and a VPN as a backup connection.
- Make your network highly resilient by connecting from multiple on-premises networks to AWS by using multiple Direct Connect locations.

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

46




This section looks at how to apply the AWS Well-Architected Framework pillar principles to your network connectivity.



The AWS Well-Architected Framework has six pillars, and each pillar includes best practices and a set of questions that you should consider when you architect cloud solutions. This section highlights a few best practices from the pillars that are most relevant to this module. For more information about best practices by pillar, see the AWS Well-Architected website link in the course resources.

When you design connectivity to multiple networks, you should consider all the future planned workloads in your on-premises and cloud environments. How do you ensure that a workload will be resilient, secure, high performing, and cost effective when deployed over multiple networks? At the infrastructure level, you start by designing your network to be resilient, secure, high performing, and cost effective.

### Best practice approach: Foundations - plan your network topology




Reliability

#### Best practices

Provision redundant connectivity between private networks in the cloud and in on-premises environments.

Prefer hub-and-spoke topologies over many-to-many mesh.

©2024, Amazon Web Services, Inc. or its affiliates. All rights reserved.49

When you design your network, you should plan a resilient network topology. Resiliency implies that your network's workloads should function correctly and consistently when they are expected to. Your network should anticipate possible network failures and accommodate future traffic growth so that workloads can run on on-premises and AWS networks. This means resiliency is a shared responsibility between AWS and you. AWS is responsible for the resiliency of the cloud network backbone, and you are responsible for implementing resiliency on premises and in the AWS Cloud.

**Provision redundant connectivity between private networks in the cloud and in on-premises environments:**

AWS recommends having failover network mechanisms on premises and in the cloud for when an interruption occurs. When you connect your VPC to your on-premises data center through VPN, you should consider the resiliency and bandwidth requirements that you need. If you use a VPN appliance that is not resilient in its implementation, then you should have a redundant connection through a second appliance. If you choose to connect your VPC to your data center by using a Direct Connect connection and you need this connection to be highly available, implement redundant Direct Connect connections from each data center. The redundant connection should use a second Direct Connect connection from a different location than the first. If you have multiple data centers, ensure that the connections point to different locations. You can also use a VPN as a backup for a Direct Connect connection.

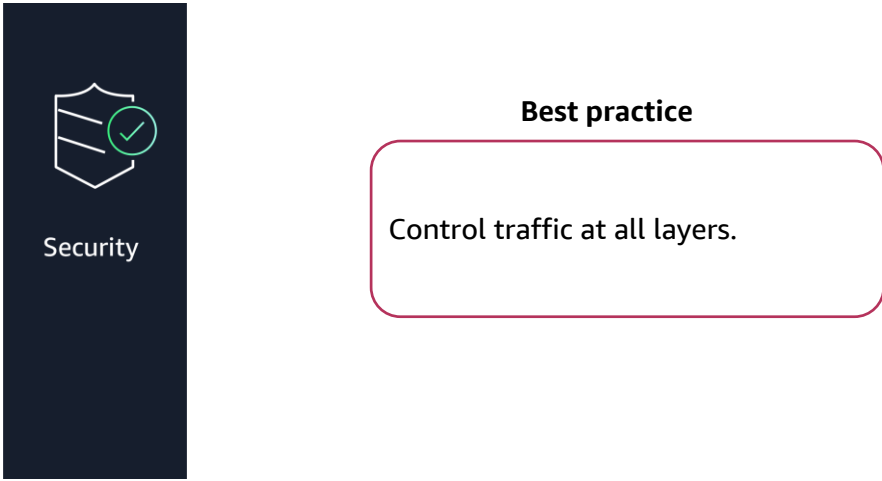
**Prefer hub-and-spoke topologies over many-to-many mesh:** AWS recommends using hub-and-spoke topologies such as Transit Gateway rather than many-to-many mesh such as Amazon VPC peering. Transit Gateway provides a hub-and-spoke model that routes traffic across your networks. Avoid implementing Amazon VPC peering to connect more than two VPCs. Do not establish multiple BGP sessions for each VPC to create connectivity to multiple VPCs across multiple AWS Regions.

In this module, you've learned about a number of AWS network services that support these best practices, including the following:

- Implement a VPN connection failover if Direct Connect between on-premises networks and AWS VPC networks is not available.
- For multiple VPCs, implement a hub-and-spoke model for less connection maintenance and ease of use.

For more information about reliability best practices, see the [AWS Well-Architected Reliability Pillar link](#) in the course resources.

### Best practice approach: Infrastructure protection – protecting networks



**Best practice**

Control traffic at all layers.

Security

aws

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

50

AWS recommends applying a zero trust approach to apply the principle of security at all layers. The careful planning and management of your network design forms the foundation of how you provide isolation and boundaries for resources within your hybrid workload running on premises and in the AWS Cloud.


**Control traffic at all layers:** AWS recommends establishing private network traffic by implementing Direct Connect or Site-to-Site VPN to protect communication. Site-to-Site VPN creates a secure connection with tunnels to traverse the public internet from an on-premises location to the AWS Cloud. Direct Connect establishes a dedicated private line from an on-premises location to a Direct Connect location.

In this module, you've learned about a number of AWS network services that support these best practices, including the following:

- Implement Site-to-Site VPN for private network traffic over the internet.
- Implement Direct Connect for a private, dedicated line to AWS.

For more information about security best practices, see [AWS Well-Architected Security Pillar](#) link in the course resources.

### Best practice approach: Data protection – protecting data in transit



Security

#### Best practice

Authenticate network communications.

Enforce encryption in transit.

aws

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

51

Data in transit is any data that is sent from one system to another. This includes communication between resources within your workload and communication between other services and your end users. By providing the appropriate level of protection for your data in transit, you protect the confidentiality and integrity of your workload's data.

**Authenticate network communications:** AWS recommends verifying the identity of communications by using protocols that support authentication, such as TLS or IPsec. Using network protocols that support authentication establishes trust between the parties. This adds to the encryption used in the protocol to reduce the risk of communications being altered or intercepted. Common protocols that implement authentication include TLS, which is used in many AWS services, and IPsec, which is used in AWS VPNs.


**Enforce encryption in transit:** AWS recommends using protocols with encryption when transmitting sensitive data outside of your VPC. Encryption helps maintain data confidentiality even when the data transits untrusted networks. All data should be encrypted in transit by using TLS protocols and cipher suites. AWS recommends using TLS version 1.3. Network traffic between your resources and the internet must be encrypted to mitigate unauthorized access to the data. Consider protecting network-to-network traffic with an IPsec VPN or Direct Connect to help ensure private network traffic.

In this module, you've learned about a number of AWS network services that support these best practices, including the following:

- Implement Site-to-Site VPN by using the IPsec protocol to authenticate traffic to establish encrypted VPN tunnels to AWS.
- Implement encryption and use Direct Connect for a private, dedicated line to AWS.

For more information about security best practices, see [AWS Well-Architected Security Pillar](#) link in the course resources.

### Best practice approach: Selection – network architecture selection



Performance Efficiency

#### Best practices

Choose appropriately sized dedicated connectivity or VPN for hybrid workloads.

Choose your workload's location based on network requirements.

aws

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

52

The optimal solution for a particular workload varies, and solutions often combine multiple approaches. Well-architected workloads use multiple solutions and include different features to improve performance.

**Choose appropriately sized dedicated connectivity or VPN for hybrid workloads:** AWS recommends sizing workload traffic that will need hybrid networking. There are multiple configuration options to choose from for connectivity, such as a dedicated connection or VPN. Select the appropriate connection type for each workload while verifying that you have adequate bandwidth and encryption requirements between your location and the cloud. Estimate the bandwidth and latency requirements for your hybrid workload. These numbers will drive the sizing requirements for your connectivity options.

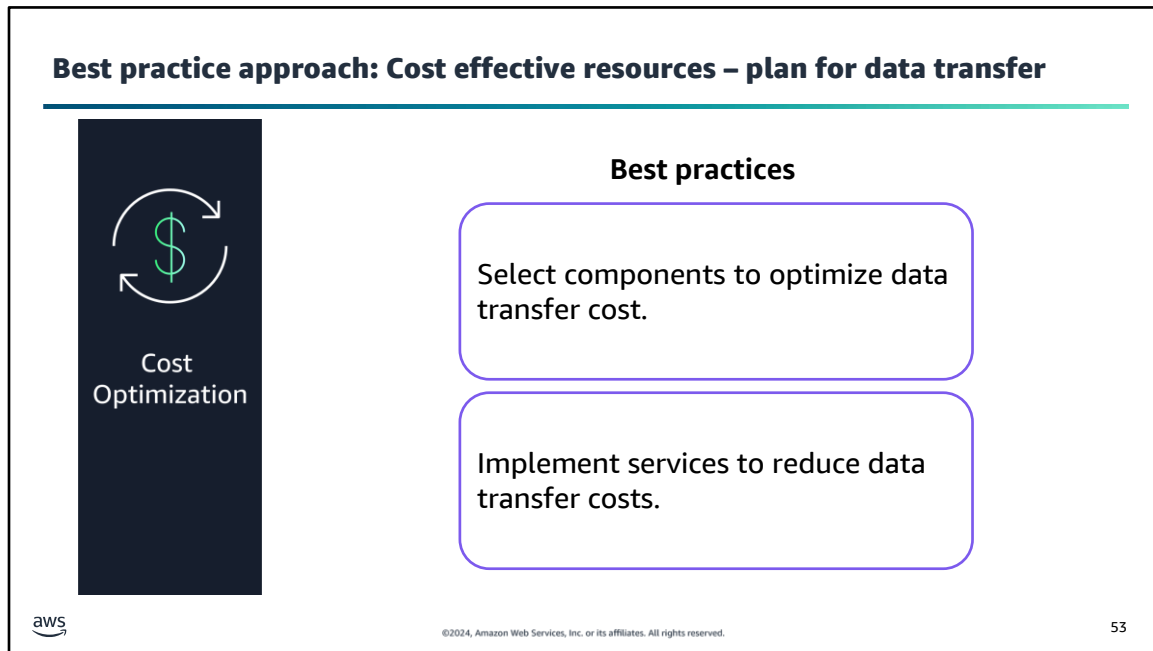
**Choose your workload's location based on network requirements:** AWS recommends evaluating options for resource placement to reduce network latency and improve throughput, providing an optimal user experience by reducing page load and data transfer times. If you have applications or users on-premises, you may benefit from having a dedicated network connection between your network and the cloud. A dedicated network connection provided by Direct Connect can reduce the chance of encountering public internet congestion or unexpected increases in latency. If you configure a Site-to-Site VPN to connect to your resources within AWS, you can optionally turn on acceleration. An accelerated Site-to-Site VPN connection uses Global Accelerator to route traffic from your on-premises network to an AWS edge location that is closest to your customer gateway device.

In this module, you've learned about a number of AWS network services that support these best practices, including the following:

- Evaluate workloads to estimate bandwidth and encryption requirements between a workload location and AWS.
- If you have users or applications on-premises, implement Direct Connect for predictable performance.

For more information about performance efficiency best practices, see [AWS Well-Architected Performance Efficiency Pillar](#) link in the course resources.





A workload on a network should fully use all resources, achieve outcomes at the lowest possible price, and meet functional requirements. Network costs should be included in the workload cost benchmark. The key to saving on costs is to select the best pricing model with the appropriate network configurations for your workloads.

**Select components to optimize data transfer cost:** AWS recommends architecting data transfer to minimize data transfer costs. This may involve using content delivery networks to locate data closer to users or using a dedicated network connection from your on-premises networks to AWS such as Direct Connect. You can also use WAN optimization and application optimization to reduce the amount of data that is transferred between components.

**Implement services to reduce data transfer costs:** AWS recommends implementing services to reduce data transfer. By using data transfer modeling, look at where the largest costs and highest volume flows are. Review the AWS services. Assess whether there is a service that reduces or removes the transfer, specifically networking and content delivery. AWS prefers that customers use Direct Connect instead of VPN to connect to AWS because Direct Connect provides predictable connectivity.

In this module, you've learned to use a dedicated network connection such as Direct Connect from an on-premises environment to VPCs on your AWS account in order to optimize data transfer costs and provide predictable data transfer cost.

For more information about cost optimization best practices, see [AWS Well-Architected Cost Optimization](#) link in the course resources.

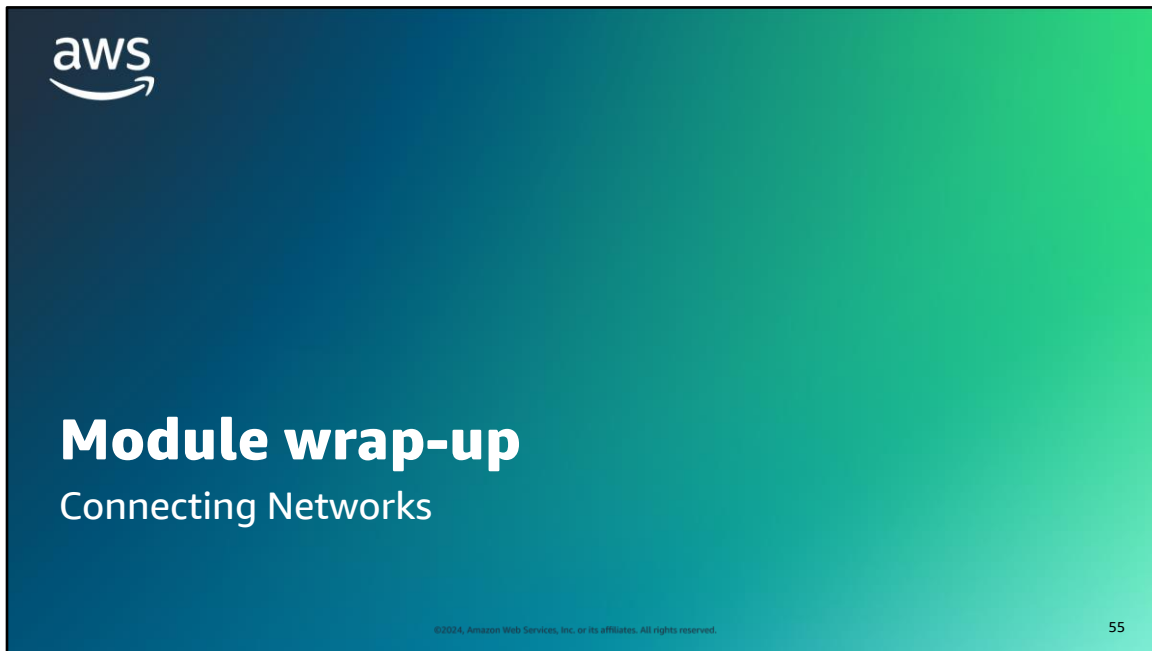
## Key takeaways: Applying Well-Architected Framework principles to your network



- Provision redundant connectivity between private networks in the cloud and in on-premises environments.
- Prefer hub-and-spoke topologies over many-to-many mesh.
- Control traffic at all layers.
- Authenticate network communications.
- Enforce data encryption in transit.
- Select components to optimize and reduce data transfer cost.
- Choose appropriately sized dedicated connectivity or VPN for hybrid workloads.
- Choose your workload's location based on network requirements.

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

54



This section summarizes what you have learned and brings the module to a close.

## Module summary

---

This module prepared you to do the following:

- Describe how to connect an on-premises network to the AWS Cloud.
- Describe how to connect multiple VPCs in the AWS Cloud.
- Connect VPCs in the AWS Cloud by using VPC peering.
- Describe how to scale VPCs in the AWS Cloud.
- Use the AWS Well-Architected Framework principles when connecting networks.



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

56

## Considerations for the café

---



- Discuss how you as a cloud architect might advise the café based on the key cloud architect concerns presented at the start of this module.



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

57

## Module knowledge check



- The knowledge check is delivered online within your course.
- The knowledge check includes 10 questions based on material presented on the slides and in the slide notes.
- You can retake the knowledge check as many times as you like.

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

58

Use your online course to access the knowledge check for this module.

## Sample exam question

An application running on Amazon EC2 instances in a virtual private cloud (VPC) processes sensitive information stored on Amazon S3. The information is accessed by using an Amazon S3 public Regional endpoint over the internet. The security team is concerned that the internet connectivity to Amazon S3 is a security risk. Which solution will resolve the security concern with the most efficient network route?

Identify the key words and phrases before continuing.

The following are the key words and phrases:

- EC2 instances in a VPC
- Sensitive information
- Internet connectivity to Amazon S3 is a security risk
- Most efficient network route



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


59

The question asks you to consider security and efficiency when deciding on the best solution.

## Sample exam question: Response choices

An application running on Amazon EC2 instances in a virtual private cloud (VPC) processes sensitive information stored on Amazon S3. The information is accessed by using an Amazon S3 public regional endpoint over the internet. The security team is concerned that the internet connectivity to Amazon S3 is a security risk. Which solution will resolve the security concern with the most efficient network route?

Choice	Response
A	Access the data through an internet gateway.
B	Access the data through a virtual private network (VPN) connection.
C	Access the data through a NAT gateway.
D	Access the data through a VPC endpoint for Amazon S3.

 ©2024, Amazon Web Services, Inc. or its affiliates. All rights reserved. 60


Use the key words that you identified on the previous slide, and review each of the possible responses to determine which one best addresses the question.



## Sample exam question: Answer

The answer is D.

Choice	Response
A	
B	
C	
D	Access the data through a VPC endpoint for Amazon S3.

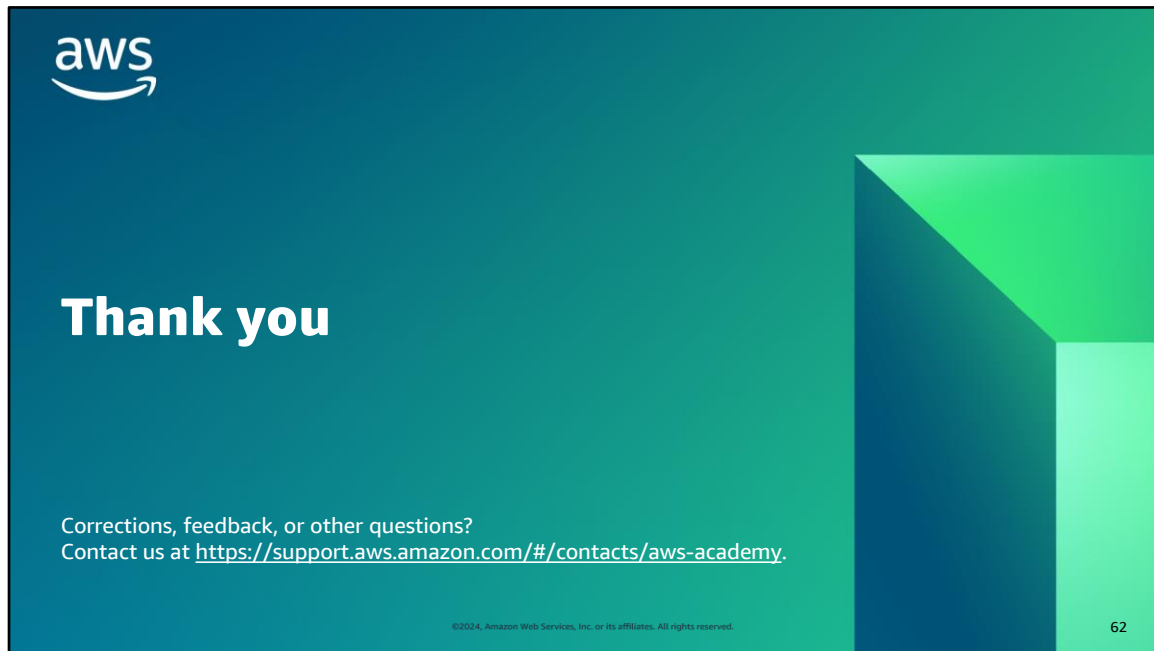
 ©2024, Amazon Web Services, Inc. or its affiliates. All rights reserved. 61

Choice A (Access the data through an internet gateway) can be eliminated because traffic will still be traversing the internet and it is not the most efficient route.

Choice B (Access the data through a virtual private network [VPN] connection) can also be eliminated because you cannot connect to Amazon S3 by VPN.

Choice C (Access the data through a NAT gateway) is not the most efficient way to route from an EC2 instance to an S3 bucket because a NAT gateway uses additional network nodes while traversing the AWS backbone network in a Region.

Choice D is the best choice. Use a gateway VPC endpoint for a no additional cost solution without bandwidth and packet size limits.



That concludes this module. The Content Resources page of your course includes links to additional resources that are related to this module.