Poll question

How many VPCs does your organization use?

A. <20

B. 20 to 100

C. >100

D. I'm not sure

 2

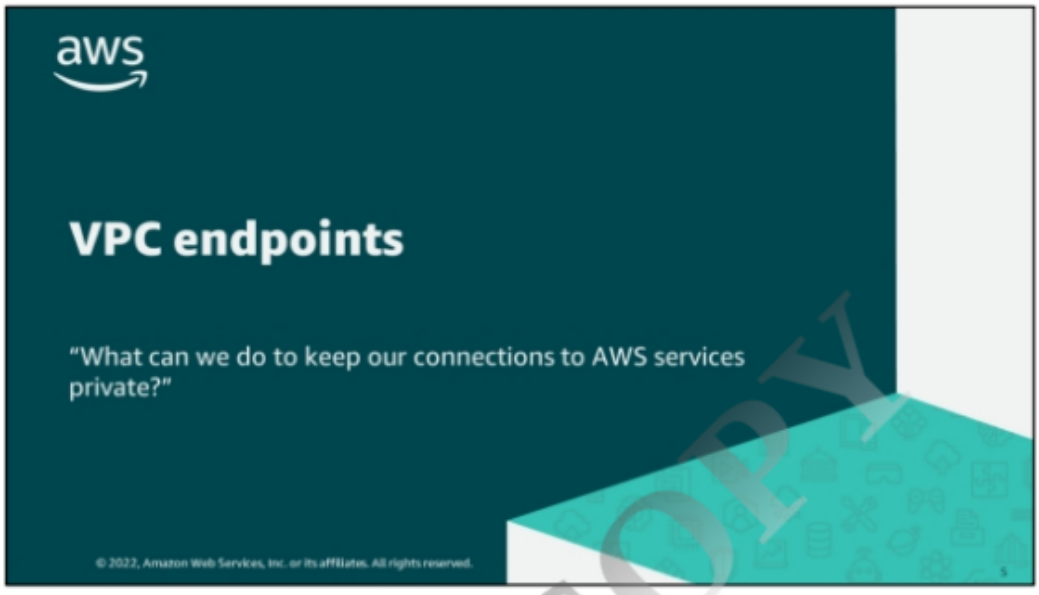
Module overview

- Business requests
- VPC endpoints
- VPC peering
- Hybrid networking
- AWS Transit Gateway
- Present solutions
- Knowledge check

 3

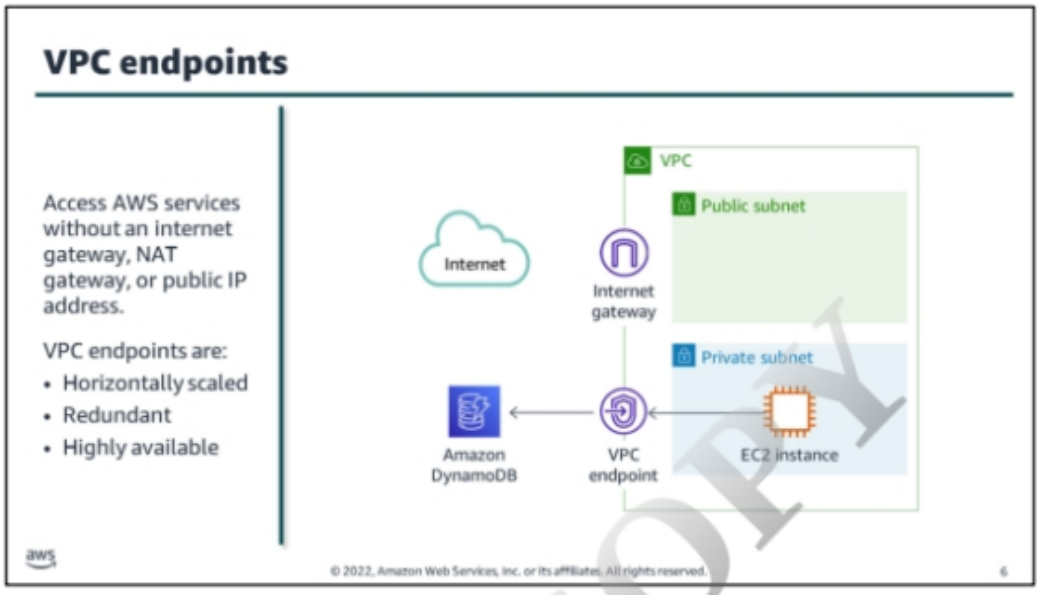
Imagine your network engineer meets with you to discuss how to connect multiple networks together. They also want to set up a hybrid environment. Here are some questions they are asking you about changes to Amazon Virtual Private Cloud (Amazon VPC) networking.

At the end of this module, you meet with the network engineer and present some solutions.



The network engineer asks, "What can we do to keep our connections to AWS services private?"

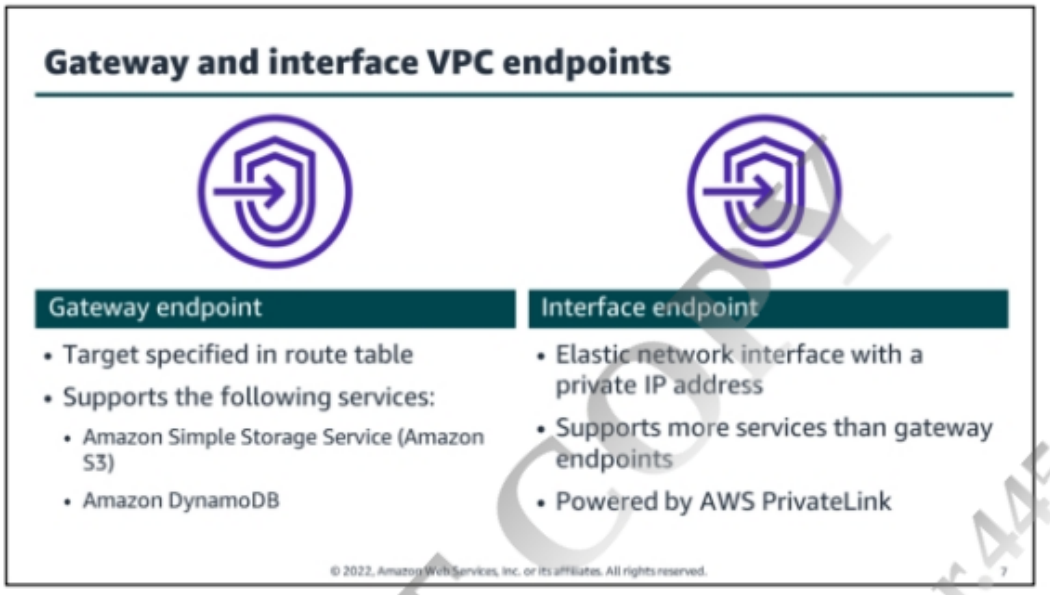
The networking team must build paths to protect traffic to and from AWS services such as Amazon Simple Storage Service (Amazon S3) and AWS Systems Manager.



Without VPC endpoints, a VPC requires an internet gateway and a NAT gateway, or a public IP address, to access serverless services outside of the VPC.

A VPC endpoint provides a reliable path between your VPC and supported AWS services. You do not need an internet gateway, NAT device, VPN connection, or AWS Direct Connect connection. Instances in your VPC do not require public IP addresses to communicate with resources in the service.

Endpoints are virtual devices. They are horizontally scaled, redundant, and highly available VPC components. They permit communication between instances in your VPC and services without imposing availability risks or bandwidth constraints on your network traffic.



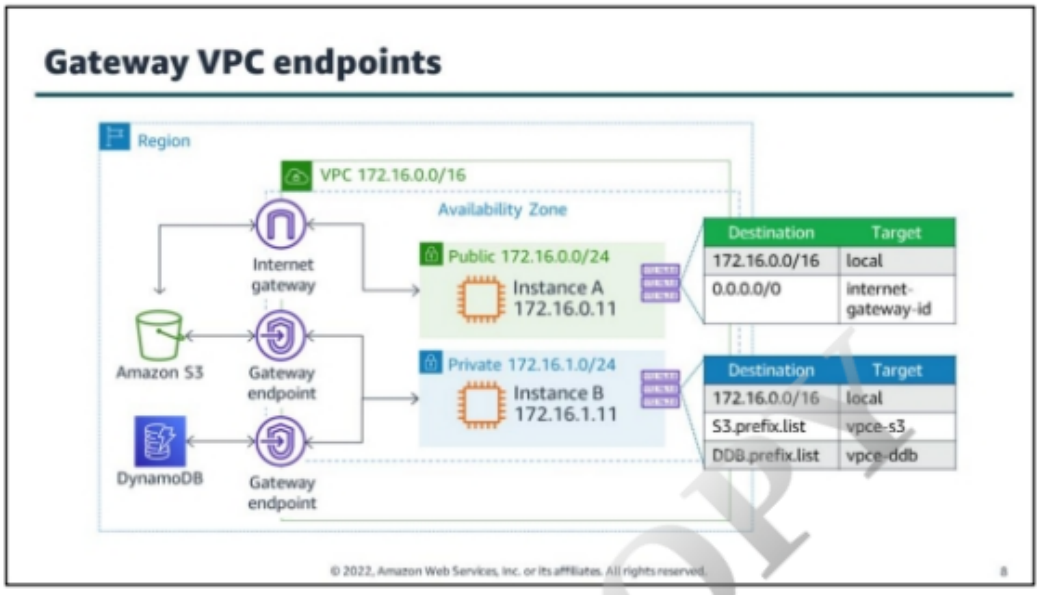
Gateway VPC endpoints and interface VPC endpoints help you access services over the AWS backbone.

Gateway endpoint
A *gateway VPC endpoint* (gateway endpoint) is a gateway that you specify as a target for a route in your route table for traffic destined for a supported AWS service. The following AWS services are supported: Amazon S3 and Amazon DynamoDB.

Interface endpoint
An *interface VPC endpoint* (interface endpoint) is an elastic network interface with a private IP address from the IP address range of your subnet. The network interface serves as an entry point for traffic destined to a supported service. AWS PrivateLink powers interface endpoints and it avoids exposing traffic to the public internet.

This course does not cover Gateway Load Balancer endpoints. For more information about Gateway Load Balancer endpoints, see "Module 10: Networking 2" in the *Online Course Supplement: Architecting on AWS* (https://explore.skillbuilder.aws/learn/course/external/view/elearning/8319/architecting-on-aws-online-course-supplement).

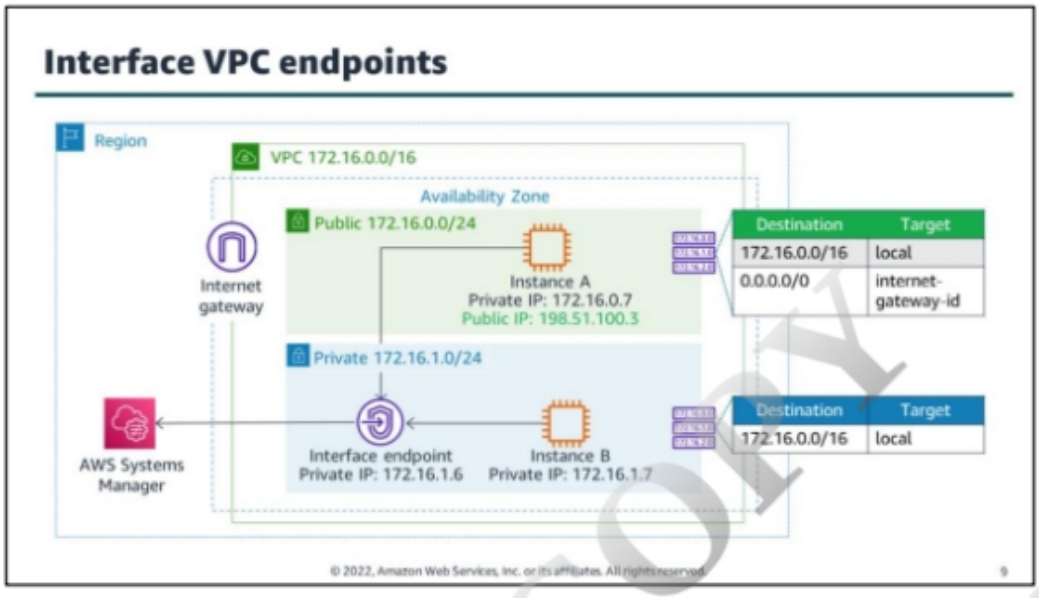
For more information about all services that integrate with interface endpoints, see "AWS PrivateLink and VPC endpoints" (https://docs.aws.amazon.com/vpc/latest/privatelink/endpoint-services-overview.html).



Specify a gateway VPC endpoint (gateway endpoint) as a route table target for traffic that is destined for Amazon S3 and DynamoDB. There is no additional charge for using gateway endpoints. Standard charges apply for data transfer and resource usage.

In the diagram, instance A in the public subnet communicates with Amazon S3 via an internet gateway. Instance A has a route to local destinations in the VPC. Instance B communicates with an Amazon S3 bucket and an Amazon DynamoDB table using unique gateway endpoints. The diagram shows an example of a private route table. The private route table directs your Amazon S3 and DynamoDB requests through each gateway endpoint using routes. The route table uses a prefix list to target the specific Region for each service.

For more information, see "Gateway endpoints" (https://docs.aws.amazon.com/vpc/latest/privatelink/gateway-endpoints.html).



With an interface VPC endpoint (interface endpoint), you can privately connect your VPC to services as if they were in your VPC. When the interface endpoint is created, traffic is directed to the new endpoint without changes to any route tables in your VPC.

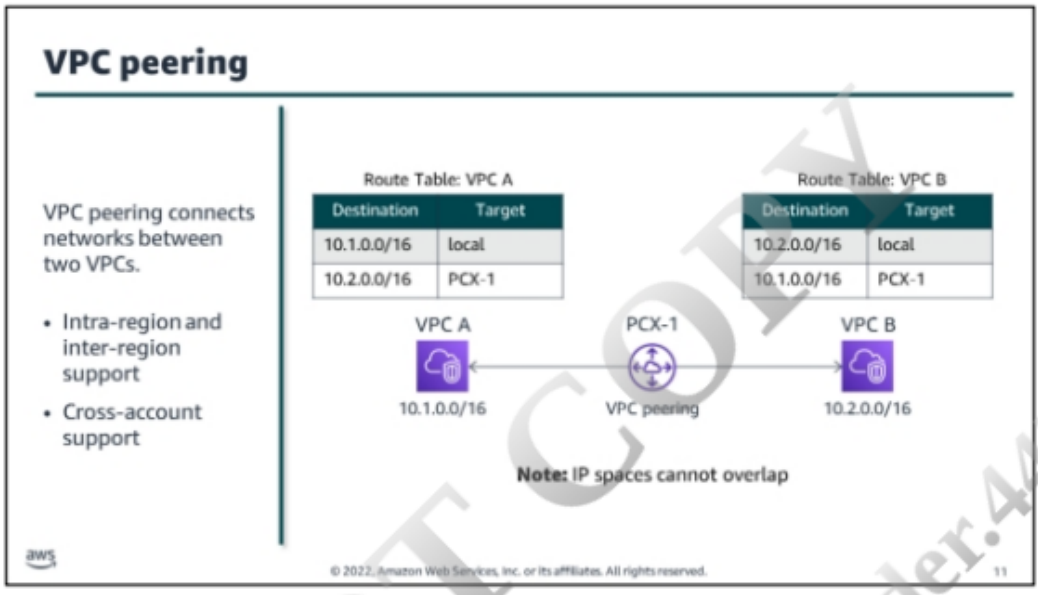
In this example, a Region is shown with Systems Manager outside of the example VPC. The example VPC has a public and private subnet with an Amazon Elastic Compute Cloud (Amazon EC2) instance in each. Systems Manager traffic sent to ssm.*region*.amazonaws.com is sent to an elastic network interface in the private subnet.

For more information about interface VPC endpoints, see "Access an AWS service using an interface VPC endpoint" (https://docs.aws.amazon.com/vpc/latest/privatelink/vpce-interface.html).

For a full list of AWS services that support interface endpoints, see "AWS services that integrate with AWS PrivateLink" (https://docs.aws.amazon.com/vpc/latest/privatelink/aws-services-privatelink-support.html).

The network engineer asks, "How can we privately route traffic between our VPCs?"

The networking team team is considering options for how to network across VPCs both in a single account and across multiple accounts and AWS Regions.

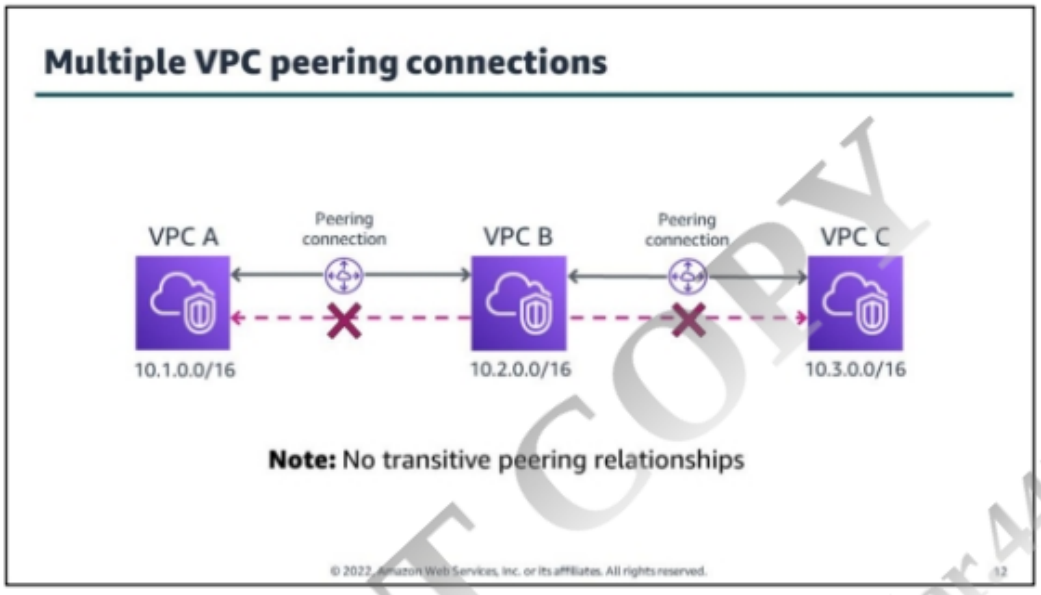


When your business or architecture becomes large enough, you will find the need to separate logical elements for security or architectural needs, or just for simplicity.

A VPC peering connection is a one-to-one relationship between two VPCs. There can only be one peering resource between any two VPCs. You can create multiple VPC peering connections for each VPC that you own.

VPC peering limitations and rules include the following:
- There is a limit on the number of active and pending VPC peering connections that you can have per VPC.
- You can have only one VPC peering connection between the same two VPCs.
- The maximum transmission unit (MTU) across a VPC peering connection is 1,500 bytes.

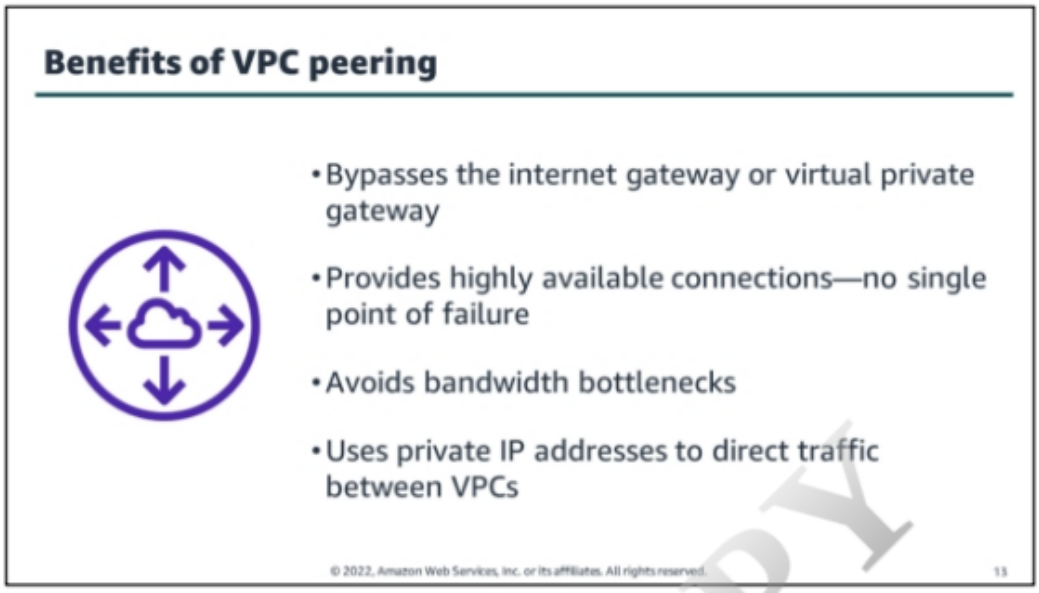
In the diagram, VPCs A and B are peered. The route table for each VPC has a route with the Classless Inter-Domain Routing (CIDR) range of the opposite VPC targeting the peering connection ID. In the diagram, the peering ID is PCX-1. Local traffic stays within each VPC.



In this diagram, VPCs A and B are peered, and B and C are peered. This does not mean that A can communicate with C. By default, VPC peering does not permit VPC A to connect to VPC C unless they are explicitly established as peers. You control which VPCs can communicate with each other.

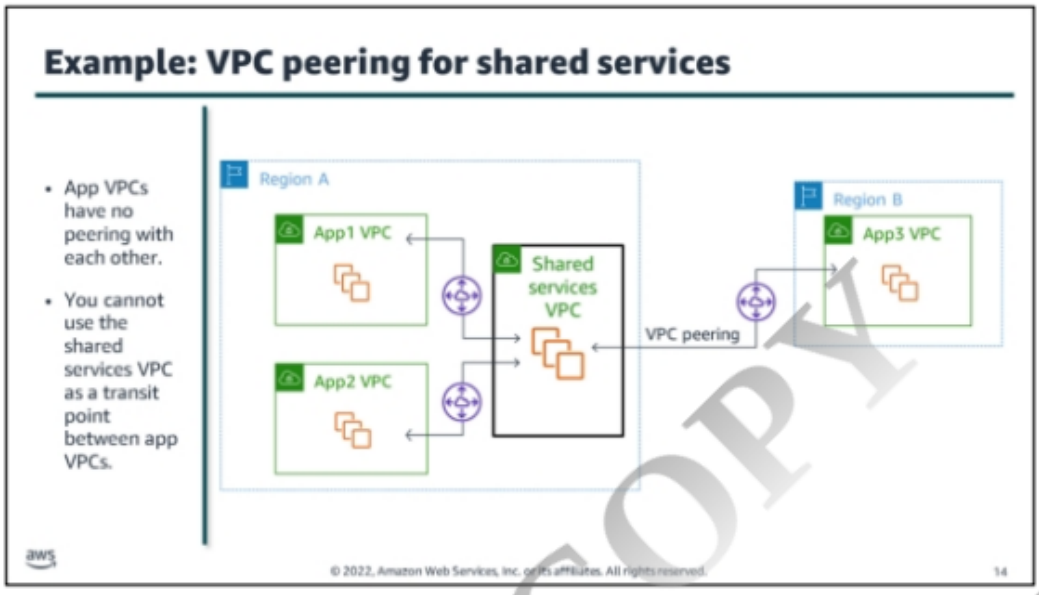
You can create multiple VPC peering connections for each VPC that you own, but transitive peering relationships are not supported. You will not have any peering relationship with VPCs that your VPC is not directly peered with. You can create a VPC peering connection between your own VPCs, or with a VPC in another AWS account within a single Region.

For more information about VPC peering limits, see "Amazon VPC quotas" in the *Amazon Virtual Private Cloud User Guide* (https://docs.aws.amazon.com/vpc/latest/userguide/amazon-vpc-limits.html).



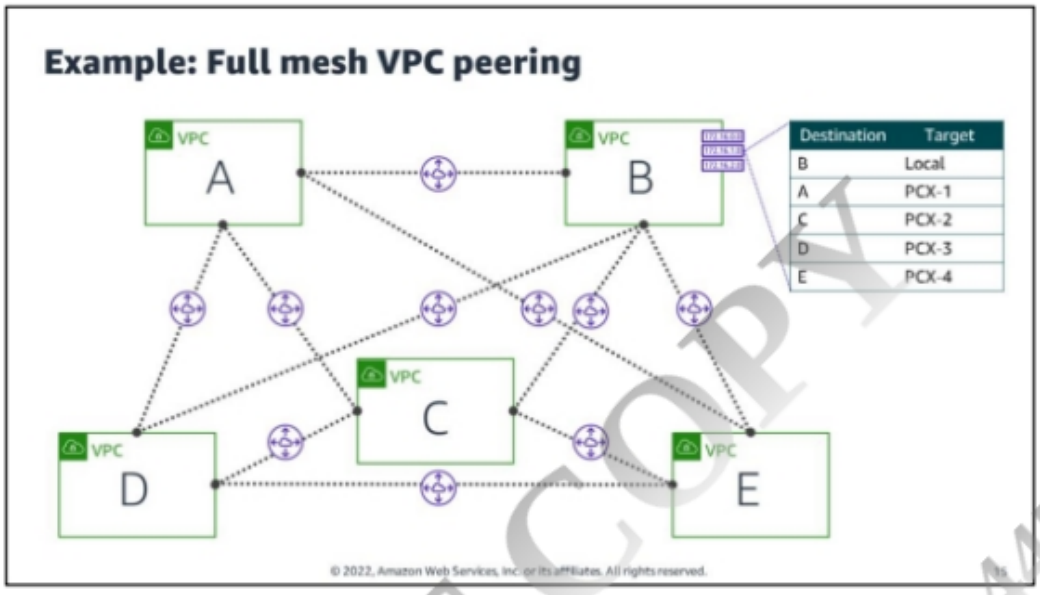
Review some of the benefits of using VPC peering to connect multiple VPCs together.

- Bypass the internet gateway or virtual private gateway. Use VPC peering to quickly connect two or more of your networks without needing other virtual appliances in your environment.
- Use highly available connections. VPC peering connections are redundant by default. AWS manages your connection.
- Avoid bandwidth bottlenecks. All inter-Region traffic is encrypted with no single point of failure or bandwidth bottlenecks. Traffic always stays on the global AWS backbone, and never traverses the public internet, which reduces threats, such as common exploits, and distributed denial of service (DDoS) attacks.
- Use private IP addresses to direct traffic. The VPC peering traffic remains in the private IP space.



In this example, your security team provides you with a *shared services VPC* that each department can peer with. This VPC allows your resources to connect to a shared directory service, security scanning tools, monitoring or logging tools, and other services.

A VPC peering connection with a VPC in a different Region is present. Inter-Region VPC peering allows VPC resources that run in different AWS Regions to communicate with each other using private IP addresses. You won't be required to use gateways, virtual private network (VPN) connections, or separate physical hardware to send traffic between your Regions.



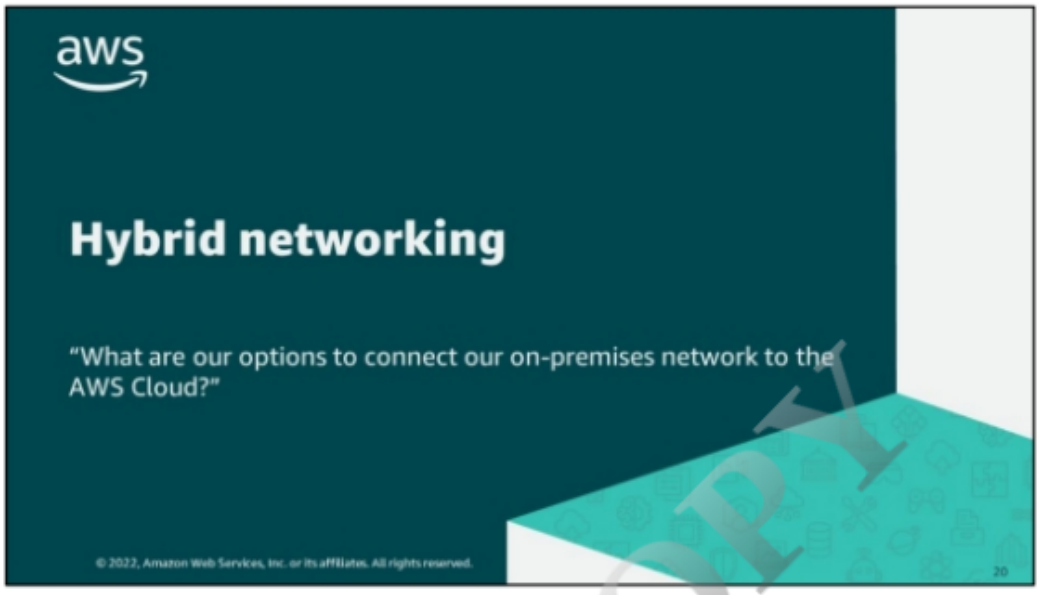
You can create a full mesh network design using VPC peering to connect each VPC to every other VPC in the organization.

In this architecture, each VPC must have a one-to-one connection with each VPC with which it is approved to communicate. This is because each VPC peering connection is nontransitive in nature and does not permit network traffic to pass from one peering connection to another.

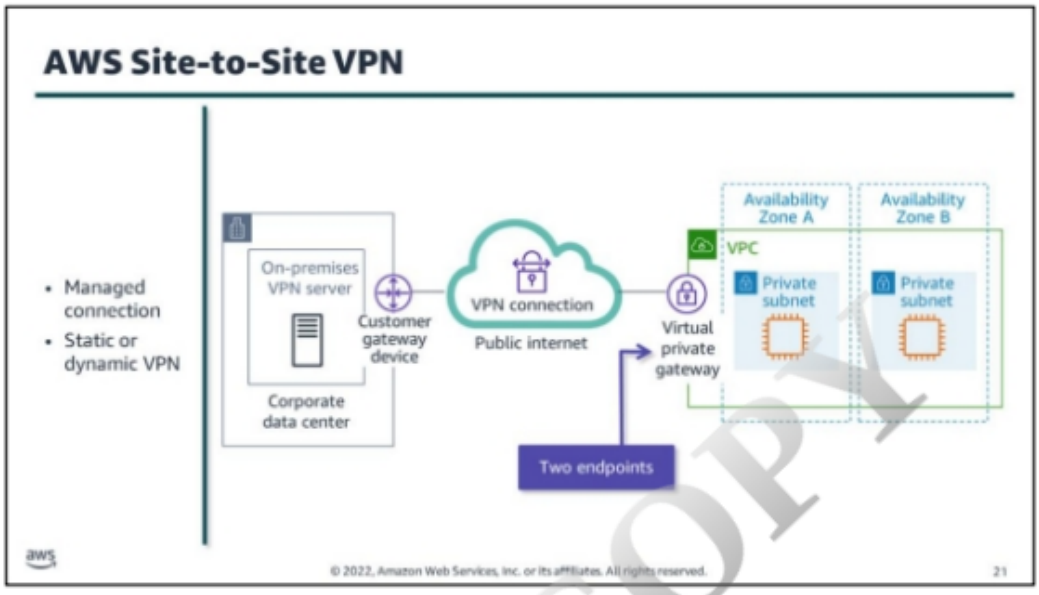
For example, VPC A is peered with VPC C, and VPC C is peered with VPC E. You cannot route packets from VPC A to VPC E through VPC C. To route packets directly between VPC A and VPC E, you must create a separate VPC peering connection between them.

The number of connections required has a direct impact on the number of potential points of failure and the requirement for monitoring. The fewer connections you need, the fewer you need to monitor and the fewer potential points of failure.

You should consider another option as your networking needs scale up. You learn about an alternate solution later in this module.



The network engineer asks, "What are our options to connect our on-premises network to the AWS Cloud?" The networking team has a requirement to create a hybrid environment that combines their on-premises data centers and their VPCs.

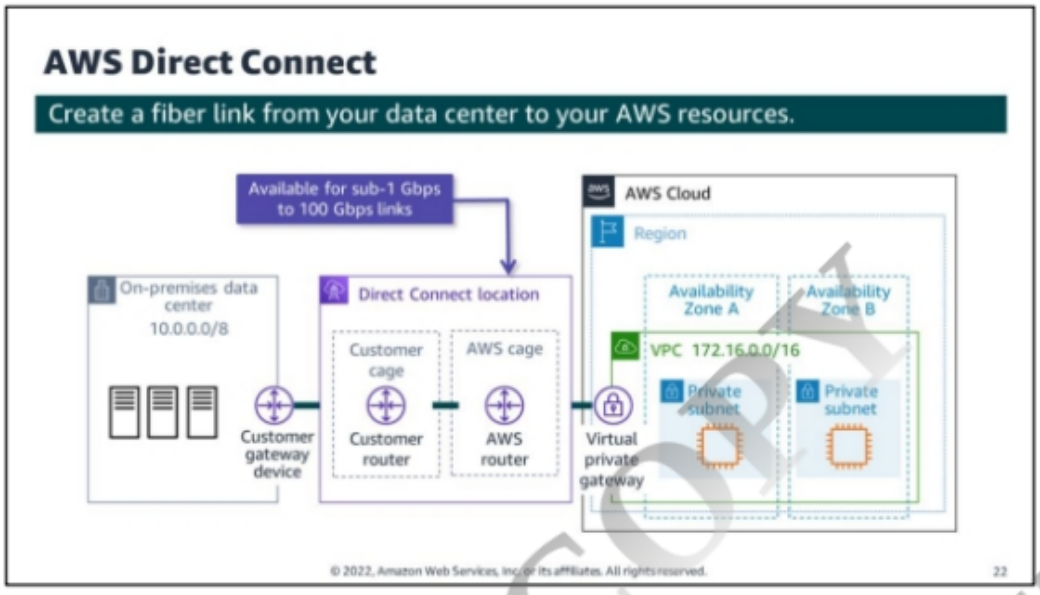


An AWS Site-to-Site VPN connection offers two VPN tunnels between a virtual private gateway (or a transit gateway) on the AWS side, and a customer gateway on the on-premises side.

- A *virtual private gateway* is the VPN concentrator on the AWS side of the AWS Site-to-Site VPN connection.
- The VPN tunnels per one VPN connection terminate in different Availability Zones.
- A *customer gateway* is a resource that you create in AWS. It represents the customer gateway device in your on-premises network. Your network administrator configures the customer gateway device or application in your remote network. AWS provides you with the required configuration information.
- You choose either static routing or dynamic routing based on the features of your customer gateway device. The dynamic routing option uses Border Gateway Protocol (BGP) to automatically discover routes.

Your customer gateway device must bring up the tunnels for your AWS Site-to-Site VPN connection by generating traffic and initiating the Internet Key Exchange (IKE) negotiation process. When you create a customer gateway, you provide information about your device to AWS.

For more information, see "What is AWS Site-to-Site VPN?" in the *AWS Site-to-Site VPN User Guide* (https://docs.aws.amazon.com/vpn/latest/s2svpn/VPC_VPN.html).



AWS Direct Connect links your internal network to a Direct Connect location over a standard Ethernet fiber-optic cable. One end of the cable is connected to your router, the other to a Direct Connect router. This is called the *cross-connect*. With this connection, you can create *virtual interfaces* directly to public AWS services (for example, to Amazon S3) or to Amazon VPC, bypassing internet service providers (ISPs) in your network path.

A Letter of Authorization and Connecting Facility Assignment (LOA-CFA) is required to begin the process of creating the cross-connect in the data center.

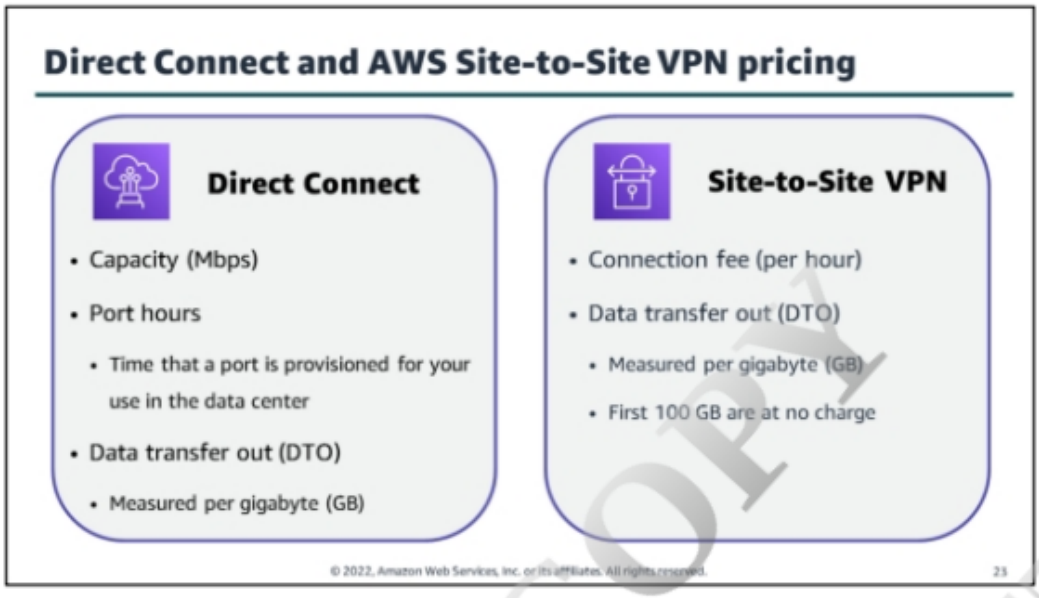
In the example, an on-premises data center holds your customer gateway device. In the data center, traffic is passed to your customer cage holding a router. It brings your traffic to an AWS router in an AWS cage. In the AWS Cloud, a virtual private gateway receives traffic over the AWS backbone, connecting the on-premises data center to the VPC. You can then create routes in your VPC to allow traffic to flow between your on-premises data center and the private subnets in your VPC.

A Direct Connect location provides access to AWS in the Region with which it is associated. You can use a single connection in a public Region or AWS GovCloud (US) to access public AWS services in all other public Regions.

To use Direct Connect in a Direct Connect location, your network must meet one of the following conditions:

- Your network is collocated with an existing Direct Connect location.
- You are working with a Direct Connect Partner.
- You are working with an independent service provider to connect to Direct Connect.

For more information about AWS Direct Connect connections, see "AWS Direct Connect" (https://aws.amazon.com/directconnect/).



It is important for you to consider pricing as a factor when deciding whether to use AWS Site-to-Site VPN or AWS Direct Connect.

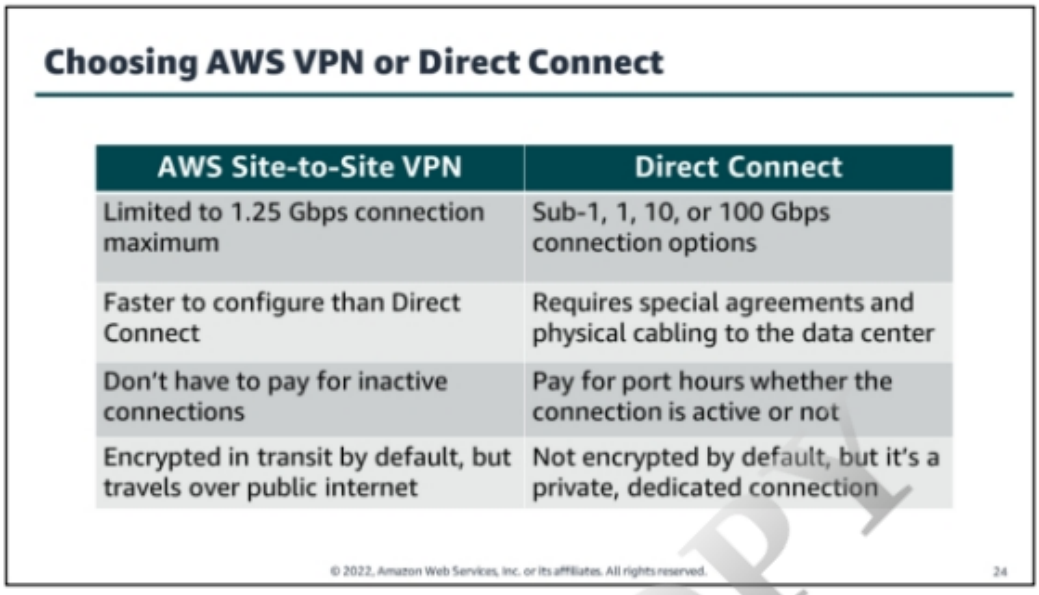
Direct Connect pricing factors include the following:

- *Capacity* is the maximum rate that data can be transferred through a network connection. The capacity of AWS Direct Connect connections are measured in megabits per second (Mbps) or gigabits per second (Gbps).
- *Port hours* measure the time that a port is provisioned for your use with AWS or an AWS Direct Connect Delivery Partner's networking equipment inside an AWS Direct Connect location. Even when no data is passing through the port, you are charged for port hours. Port hour pricing is determined by the connection type: dedicated or hosted.
- *Data transfer out (DTO)* refers to the cumulative network traffic that is sent through AWS Direct Connect to destinations outside of AWS. This is charged per GB, and unlike capacity measurements, DTO refers to the amount of data transferred, not the speed. When calculating DTO, exact pricing depends on the AWS Region and AWS Direct Connect location you are using.

AWS Site-to-Site VPN has simpler cost factors to calculate. You are charged a per-hour connection fee for your use, and you are also charged for DTO, similarly to Direct Connect. With AWS Site-to-Site VPN, you receive your first 100 GB of data transfer out at no charge.

For more information about Direct Connect pricing, see "AWS Direct Connect pricing" (https://aws.amazon.com/directconnect/pricing/).

For more information about AWS Site-to-Site VPN pricing, see "AWS VPN pricing" (https://aws.amazon.com/vpn/pricing/).



You should choose the product that best meets your hybrid connectivity needs. You may choose to use either Site-to-Site VPN, Direct Connect, or both, depending on your use case.

Choose AWS VPN solutions when you:

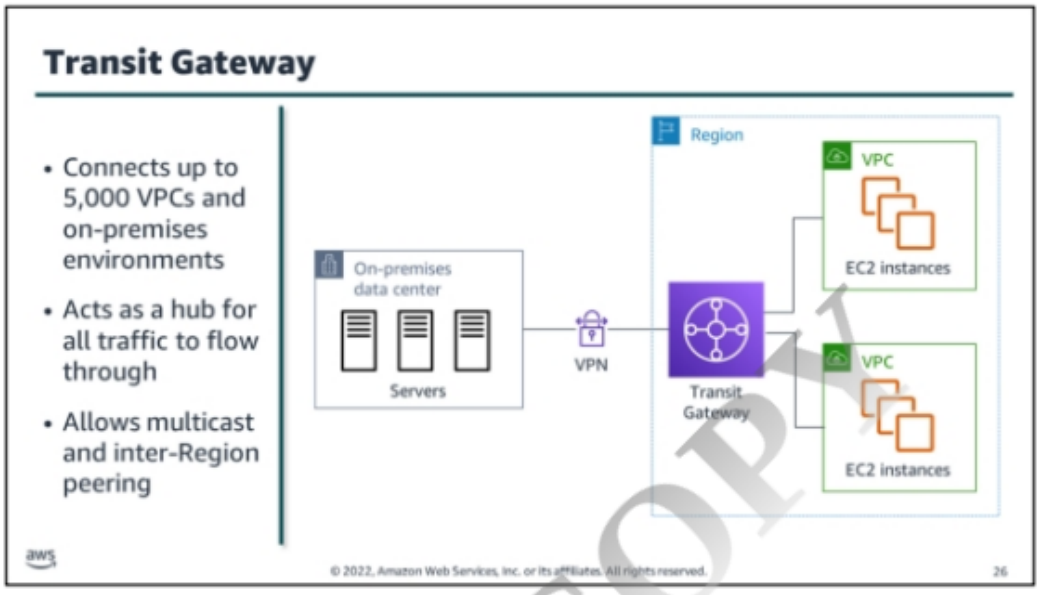
- Need a way to quickly establish a network connection between your on-premises networks and your VPC
- Need to stay within a small budget
- Require encryption in transit

Consider Direct Connect when you:

- Need faster connectivity options than what AWS Site-to-Site VPN can provide
- Are already in a collocation that supports Direct Connect
- Need predictable network performance

The network engineer asks, "Which services can reduce the number of route tables we need to manage our global network?"

The networking team must scale out the hybrid network in a way that reduces the number of route tables and connections they have to manage.

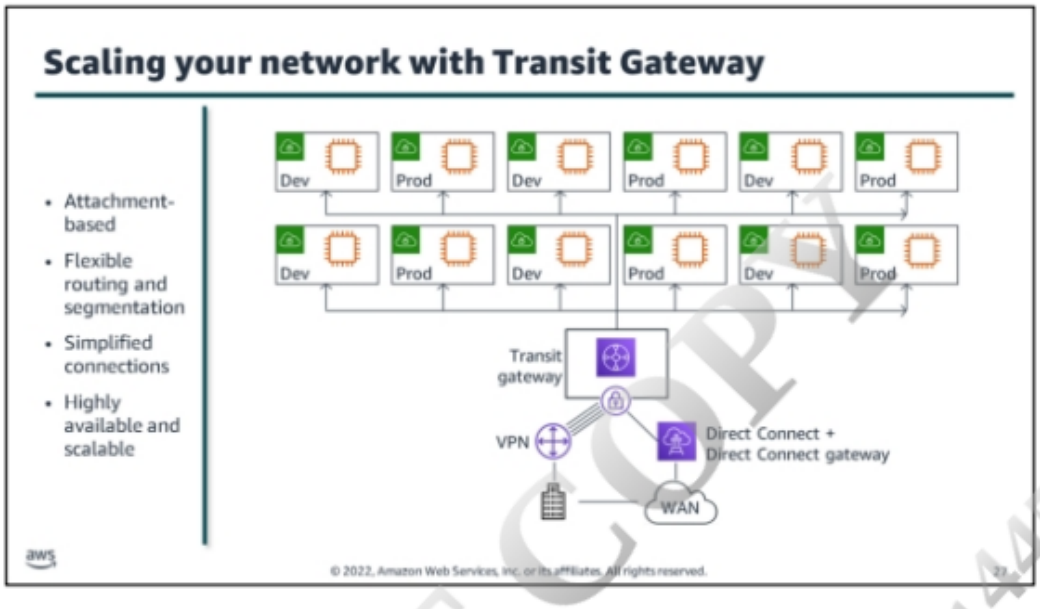


AWS Transit Gateway acts as a hub that controls how traffic is routed among all the connected networks, which act like spokes. This hub-and-spoke model significantly simplifies management and reduces operational costs because each network only has to connect to Transit Gateway and not to every other network. Any new VPC is connected to Transit Gateway and is then automatically available to every other connected network.

Routing through a transit gateway operates at Layer 3, where the packets are sent to a specific next-hop attachment based on their destination IP addresses. Your transit gateway routes Internet Protocol version 4 (IPv4) and Internet Protocol version 6 (IPv6) packets between attachments using transit gateway route tables. Configure route tables to propagate routes from the tables for the attached VPCs and VPN connections. You can add static routes to the transit gateway route tables.

A *transit gateway* is a network transit hub that you can use to interconnect your VPCs and on-premises networks, and it scales elastically, based on traffic.

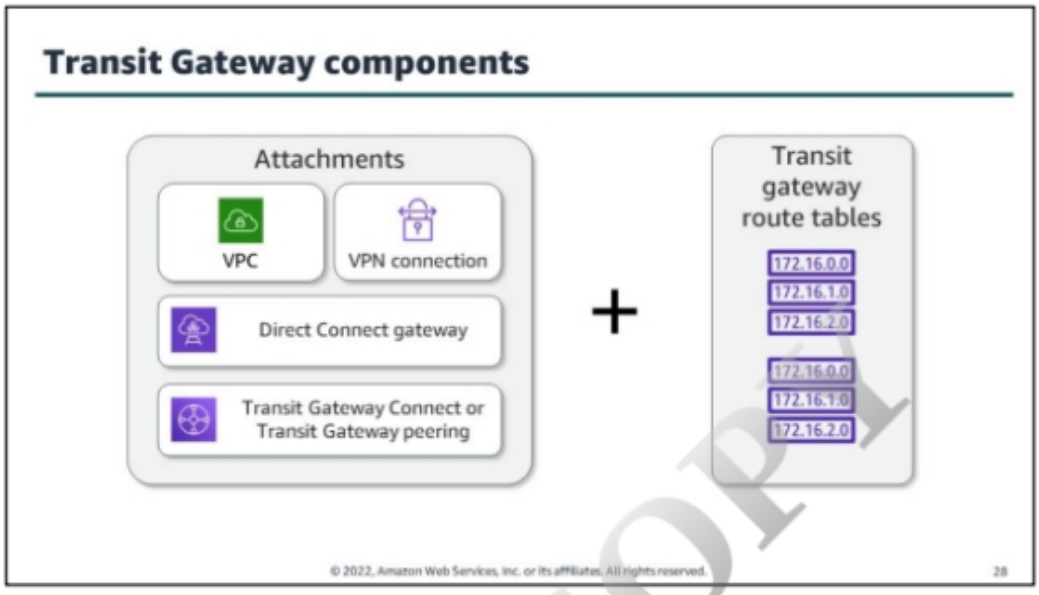
For more information about AWS Transit Gateway, see "AWS Transit Gateway" (https://aws.amazon.com/transit-gateway/).



A transit gateway acts as a cloud router to simplify your network architecture. As your network grows, the complexity of managing incremental connections doesn't slow you down. When building global applications, you can connect transit gateways using inter-Region peering.

With Transit Gateway Network Manager, you can monitor your VPCs and edge connections from a central console. Integrated with popular software-defined wide area network (SD-WAN) devices, Transit Gateway Network Manager helps you identify issues in your global network.

Traffic between a VPC and transit gateway remains on the AWS global private network and is not exposed to the public internet. Transit Gateway inter-Region peering encrypts all traffic. With no single point of failure or bandwidth bottleneck, it protects you against DDoS attacks and other common exploits.



Transit Gateway is made up of two important components: attachments and route tables.

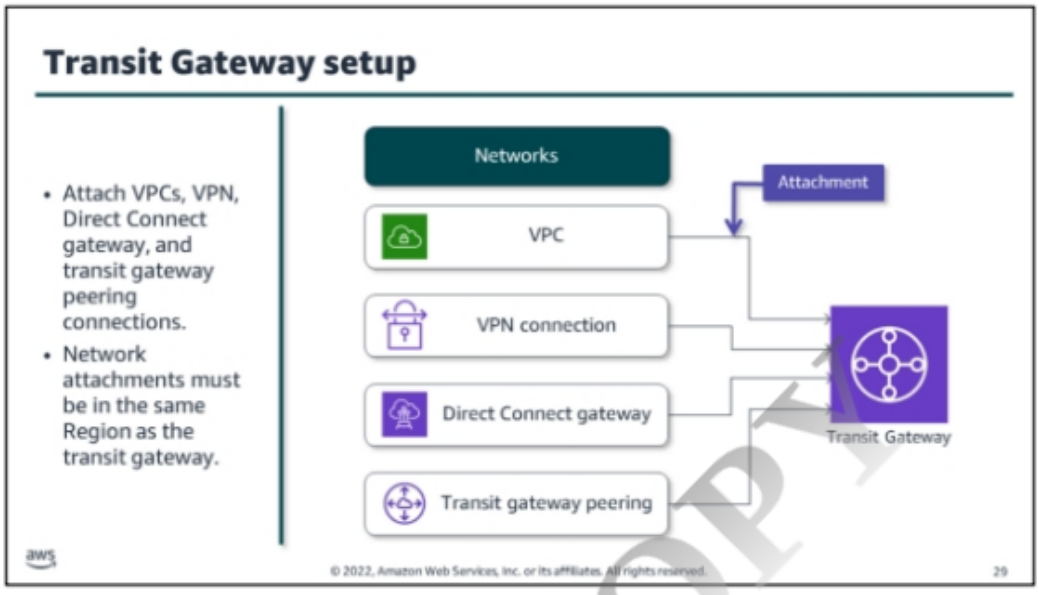
A transit gateway *attachment* is a source and a destination of packets. You can attach one or more of the following resources if they are in the same Region as the transit gateway:

- VPC
- VPN connection
- Direct Connect gateway
- Transit Gateway Connect
- Transit Gateway peering connection

You can use VPN connections and Direct Connect gateways to connect your on-premises data centers to transit gateways. With a transit gateway, you can connect with VPCs in the AWS Cloud creating a hybrid network.

A transit gateway has a default route table and can optionally have additional route tables. A route table includes dynamic and static routes that decide the next hop based on the destination IP address of the packet. The target of these routes could be any transit gateway attachment. By default, transit gateway attachments are associated with the default transit gateway route table.

Each attachment is *associated* with exactly one route table. Each route table can be associated with zero to many attachments.



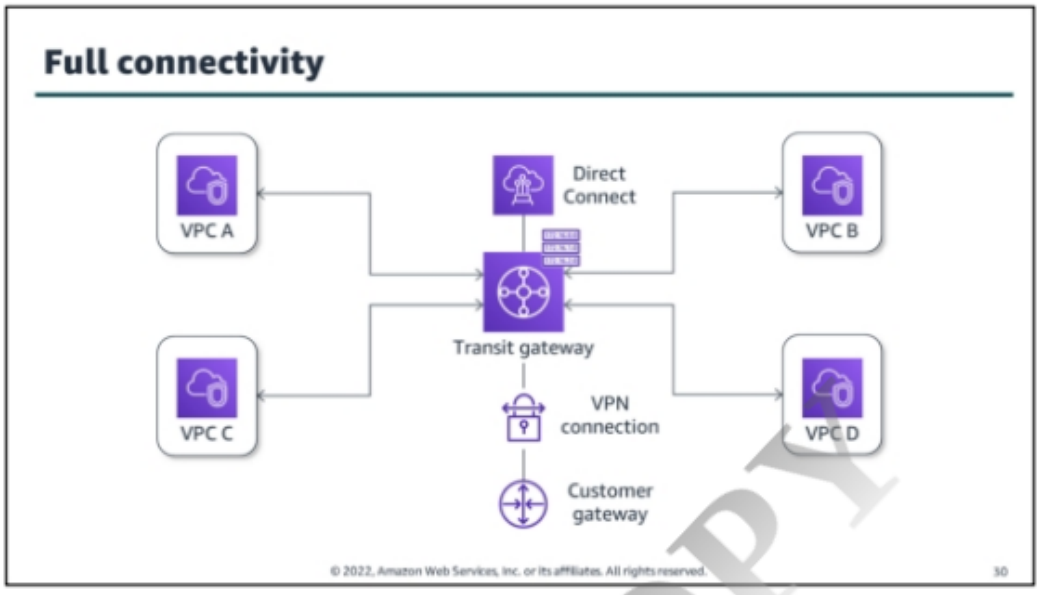
A transit gateway works across AWS accounts. You can use AWS Resource Access Manager to share your transit gateway with other accounts. After you share a transit gateway with another AWS account, the account owner can attach their VPCs to your transit gateway. A user from either account can delete the attachment at any time.

A transit gateway attachment is a source and a destination of packets. You can attach the following resources to your transit gateway:

- One or more VPCs
- One or more VPN connections
- One or more Direct Connect gateways
- One or more transit gateway peering connections

If you attach a transit gateway peering connection, the transit gateway must be in a different Region. Transit gateways support dynamic and static routing between attached VPCs and VPN connections. You can turn on or turn off route propagation for each attachment.

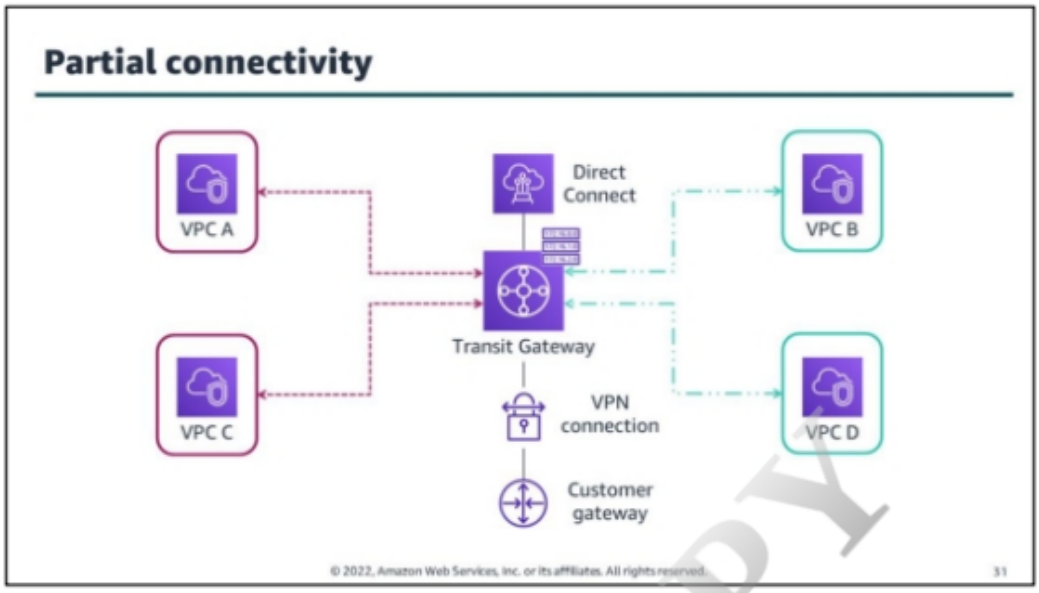
For more information about transit gateway setup, see "Examples" in the *Amazon VPC: AWS Transit Gateway* guide (https://docs.aws.amazon.com/vpc/latest/tgw/TGW_Scenarios.html).



Transit Gateway is the central hub that helps you control communication between attached resources.

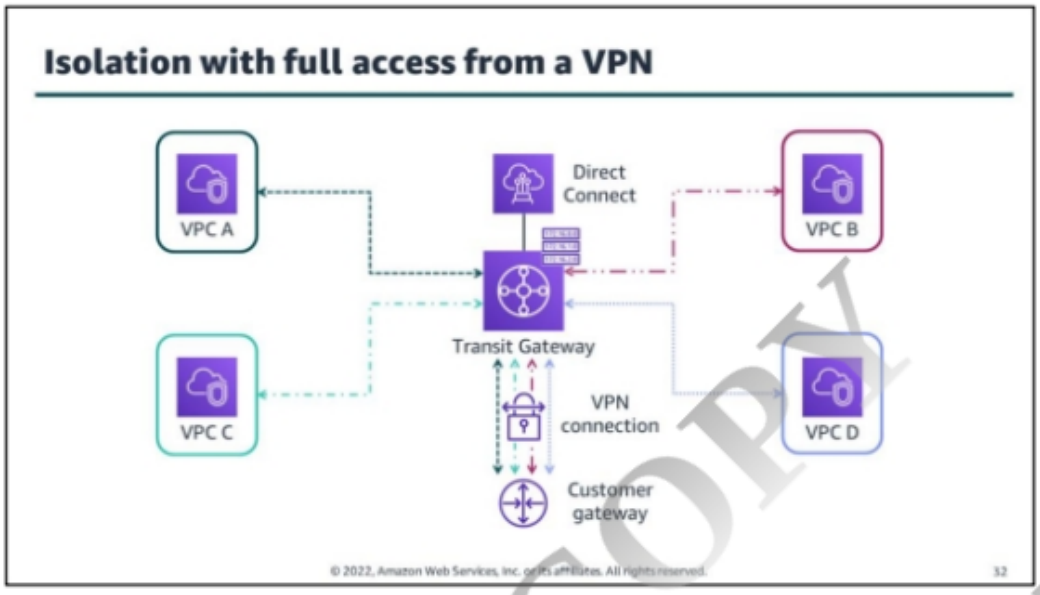
This diagram shows four VPCs (VPCs A, B, C, and D) with attachments to the transit gateway. A Direct Connect gateway and a VPN connection are also attached to the same transit gateway. A customer gateway device is on the other side of the VPN connection.

In this diagram, all of the VPCs can communicate with each other.

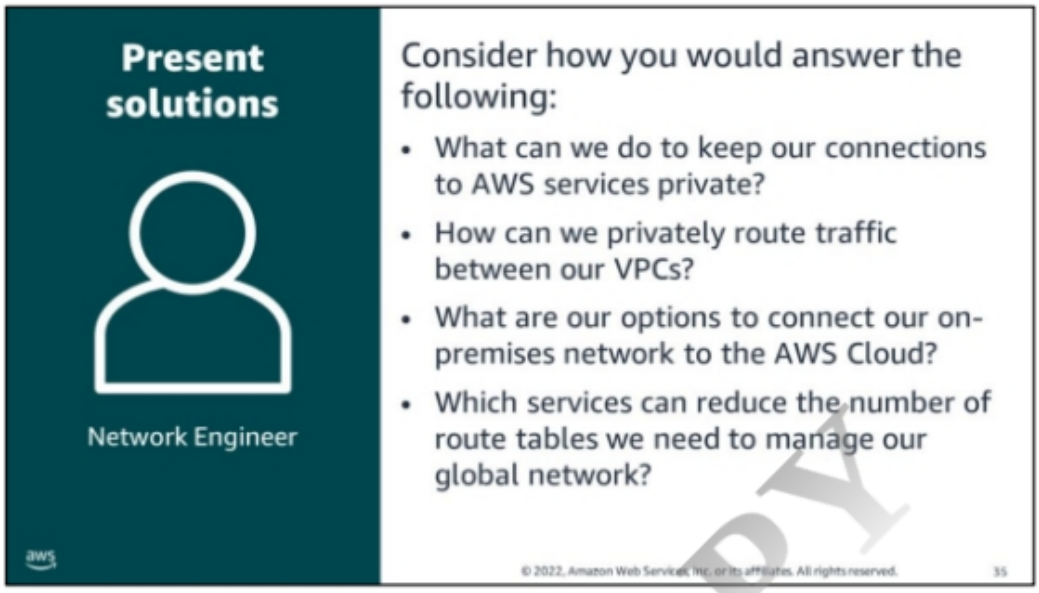


In this diagram, VPC A and VPC C can communicate with each other, but not with VPC B or VPC D.

VPC B and VPC D can communicate with each other, but not with VPC A or VPC C.



In this diagram, none of the VPCs can communicate with each other, but they can all be accessed through the VPN connection.

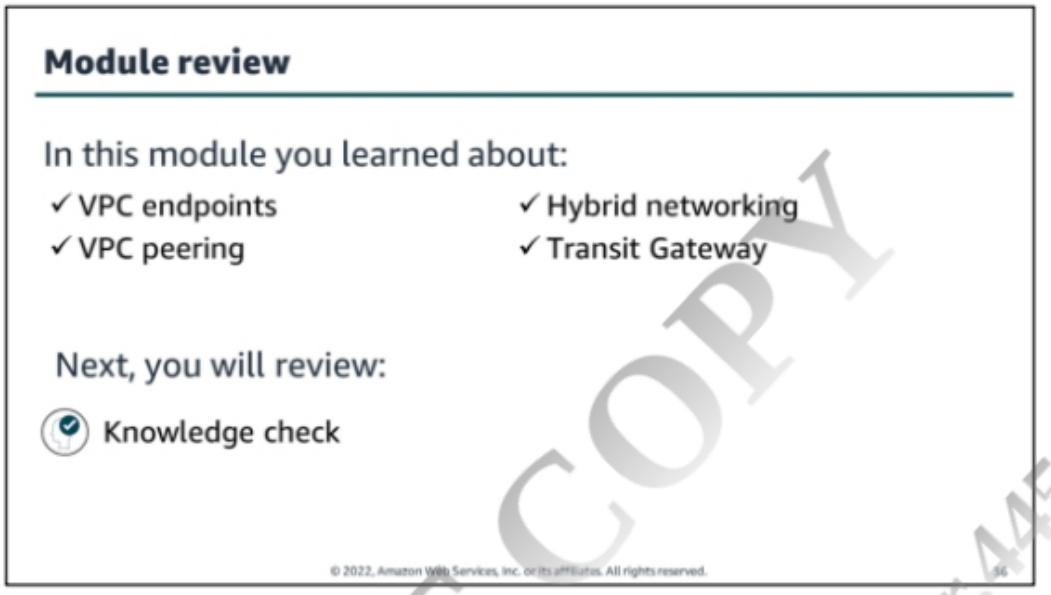


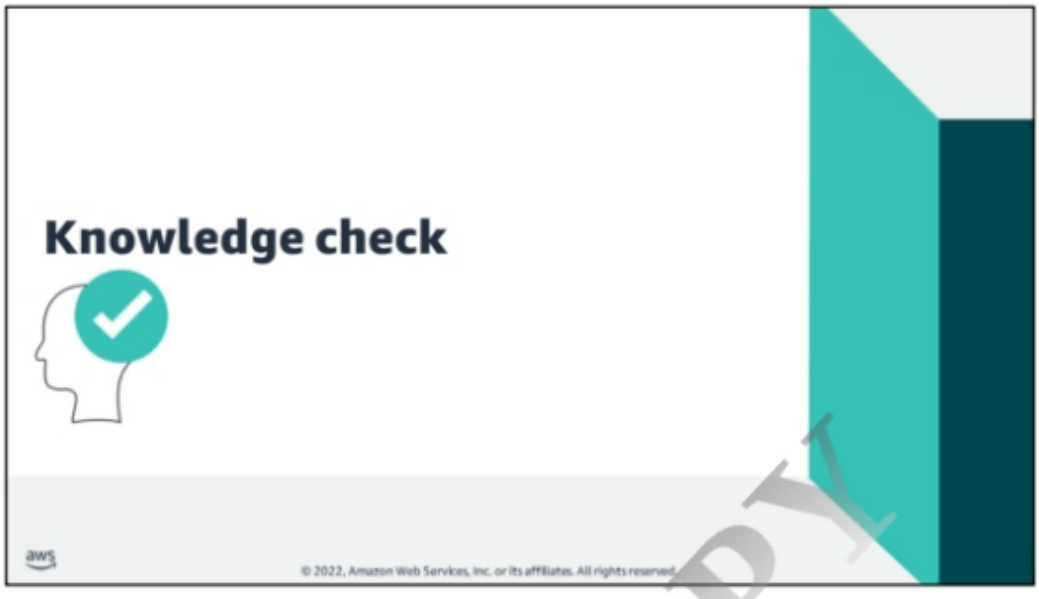
Imagine you are now ready to talk to the network engineer and present solutions that meet their architectural needs.

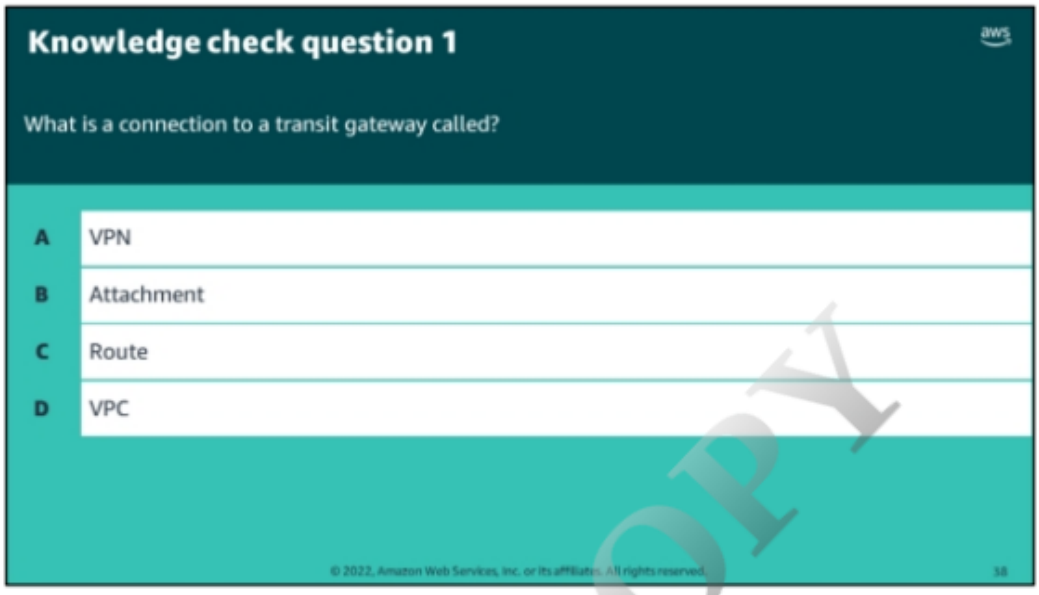
Think about how you would answer the questions from the beginning of the lesson.

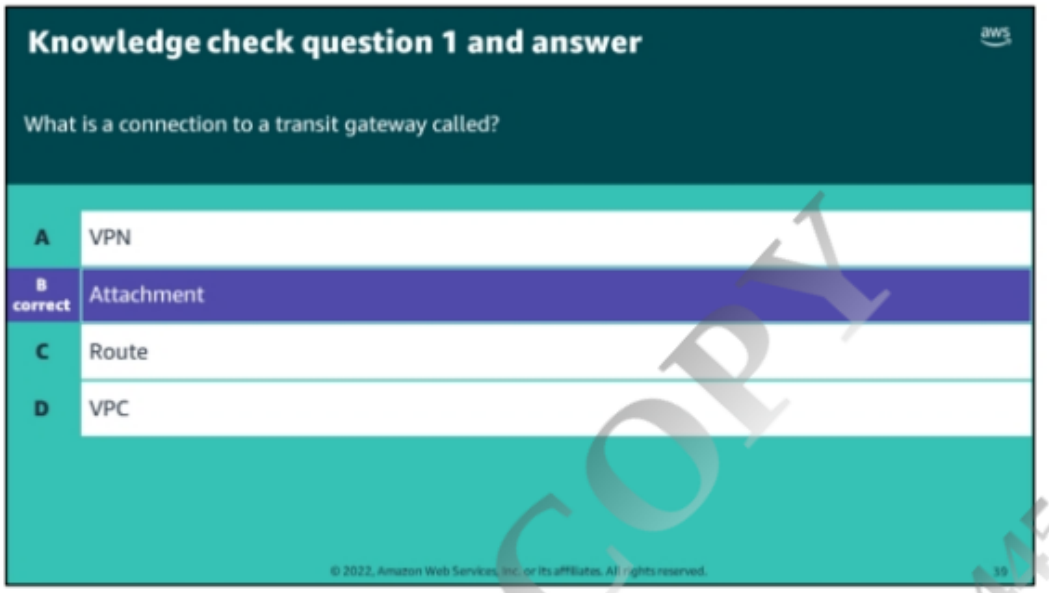
Your answers should include the following solutions:

- You can keep connections to AWS services private using interface VPC endpoints. You can also create gateway endpoints to allow connectivity in your VPCs to Amazon S3 or DynamoDB without using an internet gateway or NAT gateway.
- There are multiple solutions to privately route traffic between VPCs. Using VPC peering, you can quickly network two VPCs together, but it is difficult to scale. Consider using a Transit Gateway if you will be routing traffic between many VPCs.
- Two options to connect on-premises networks to VPCs include AWS Site-to-Site VPN and Direct Connect. Choose your solution based on your use case to optimize cost and performance.
- Transit Gateway route tables can be managed and shared between multiple VPCs in your global network. You can reduce the number of route tables you create and manage by associating existing route tables to new VPCs, when appropriate.









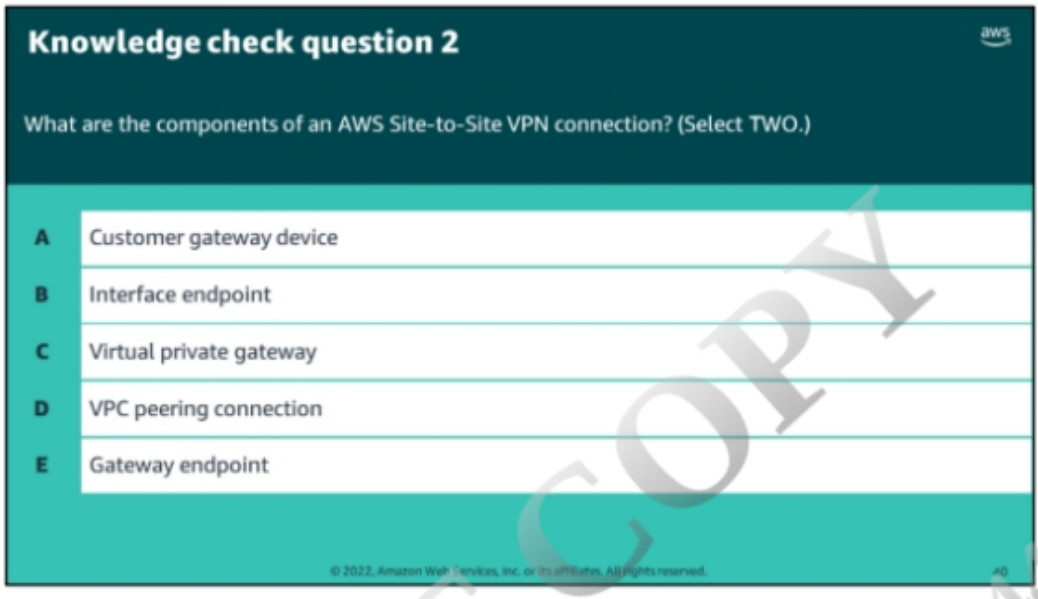
The answer is B, attachment.

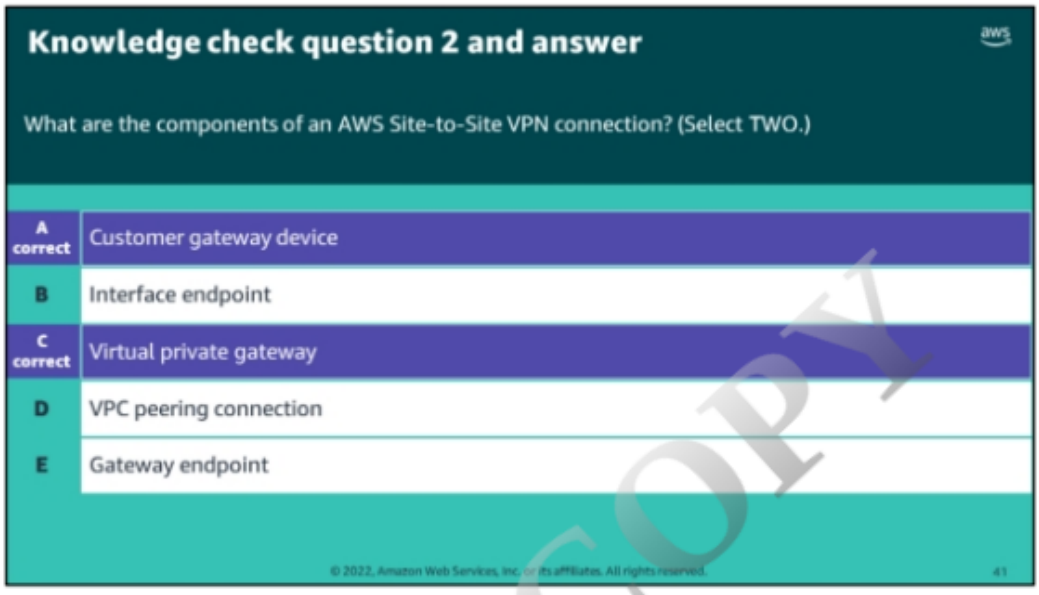
A transit gateway attachment is both a source and a destination of packets. You can attach the following resources to your transit gateway:

- One or more VPCs
- One or more VPN connections
- One or more AWS Direct Connect gateways
- One or more Transit Gateway Connect attachments
- One or more transit gateway peering connections

If you attach a transit gateway peering connection, the transit gateway must be in a different Region.

For more information about transit gateways and attachments, see "How transit gateways work" in the *Amazon VPC: AWS Transit Gateway* guide (https://docs.aws.amazon.com/vpc/latest/tgw/how-transit-gateways-work.html).



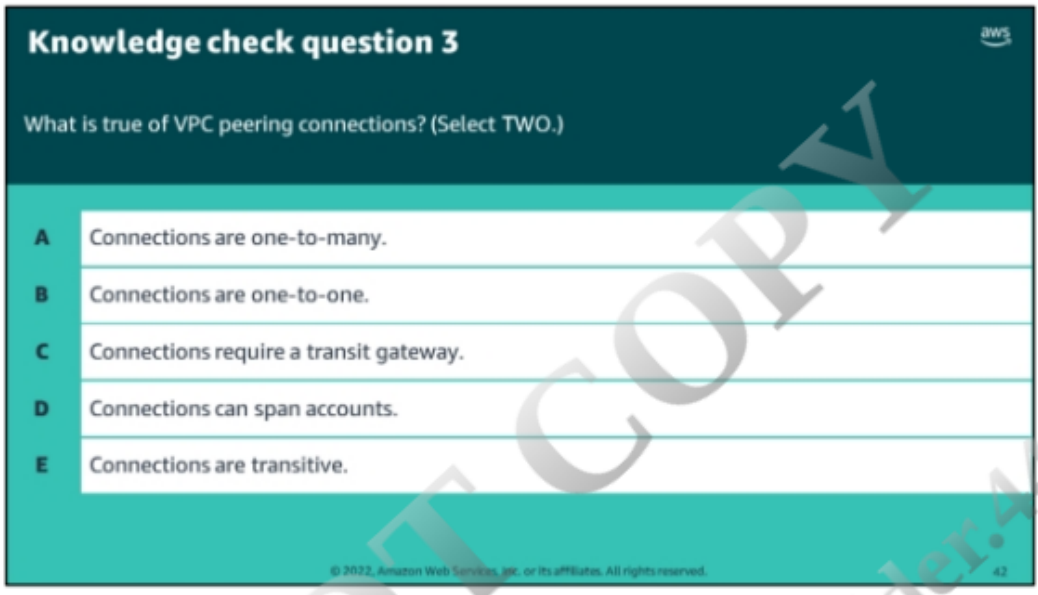


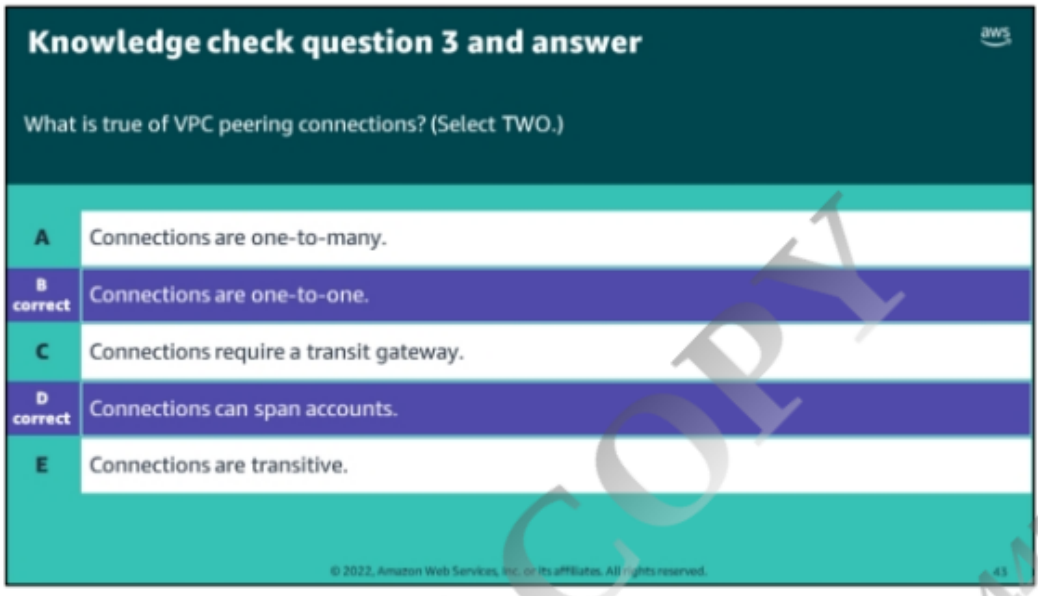
The answer is A and C, customer gateway device and virtual private gateway.

Customer gateway device: A physical device or software application on your side of the Site-to-Site VPN connection.

Virtual private gateway: The VPN concentrator on the Amazon side of the AWS Site-to-Site VPN connection. Use a virtual private gateway or a transit gateway as the gateway for the Amazon side of the Site-to-Site VPN connection.

For more information about AWS Site-to-Site VPN, see "What is AWS Site-to-Site VPN?" in the *AWS VPN User Guide* (https://docs.aws.amazon.com/vpn/latest/s2svpn/VPC_VPN.html).





The answer is B and D, connections are one-to-one and connections can span accounts.

A VPC peering connection is a one-to-one relationship between two VPCs. Only one peering resource can exist between any two VPCs. You can create multiple VPC peering connections for each VPC that you own, but transitive peering relationships are not supported.