Chapter 4
Amazon Virtual Private Cloud (Amazon VPC)

THE AWS CERTIFIED SOLUTIONS ARCHITECT ASSOCIATE EXAM OBJECTIVES COVERED IN THIS CHAPTER MAY INCLUDE, BUT ARE NOT LIMITED TO, THE FOLLOWING:

Domain 1.0: Designing highly available, cost-efficient, fault-tolerant, scalable systems

✓1.1 Identify and recognize cloud architecture considerations, such as fundamental components and effective designs.

Content may include the following:

- How to design cloud services
- Planning and design
- Familiarity with:
  - Best practices for AWS architecture
  - Architectural trade-off decisions (for example, high availability vs. cost, Amazon Relational Database Service [RDS] vs. installing your own database on Amazon Elastic Compute Cloud—EC2)
  - Hybrid IT architectures (for example, Direct Connect, Storage Gateway, VPC, Directory Services)

Domain 2.0: Implementation/Deployment

✓2.1 Identify the appropriate techniques and methods using Amazon EC2, Amazon S3, AWS Elastic Beanstalk, AWS CloudFormation, AWS OpsWorks, Amazon Virtual Private Cloud (VPC), and AWS Identity and Access Management (IAM) to code and implement a cloud solution.

Content may include the following:

- Operate and extend service management in a hybrid IT architecture
- Configure services to support compliance requirements in the cloud

Domain 3.0: Data Security

✓3.1 Recognize and implement secure practices for optimum cloud deployment and maintenance.

Content may include the following:

- AWS security attributes (customer workloads down to the physical layer)
- Amazon Virtual Private Cloud (VPC)
- Ingress vs. egress filtering, and which AWS services and features fit

- “Core” Amazon EC2 and S3 security feature sets
- Incorporating common conventional security products (Firewall and VPNs)
- Complex access controls (building sophisticated security groups, ACLs, and so on)

Introduction

The *Amazon Virtual Private Cloud (Amazon VPC)* is a custom-defined virtual network within the AWS Cloud. You can provision your own logically isolated section of AWS, similar to designing and implementing a separate independent network that would operate in an on-premises data center. This chapter explores the core components of Amazon VPC and, in the exercises, you learn how to build your own Amazon VPC in the cloud. A strong understanding of Amazon VPC topology and troubleshooting is required to pass the exam, and we highly recommend that you complete the exercises in this chapter.

Amazon Virtual Private Cloud (Amazon VPC)

Amazon VPC is the networking layer for Amazon Elastic Compute Cloud (Amazon EC2), and it allows you to build your own virtual network within AWS. You control various aspects of your Amazon VPC, including selecting your own IP address range; creating your own *subnets*; and configuring your own route tables, network gateways, and security settings. Within a region, you can create multiple Amazon VPCs, and each Amazon VPC is logically isolated even if it shares its IP address space.

When you create an Amazon VPC, you must specify the IPv4 address range by choosing a *Classless Inter-Domain Routing (CIDR)* block, such as 10.0.0.0/16. The address range of the Amazon VPC cannot be changed after the Amazon VPC is created. An Amazon VPC address range may be as large as /16 (65,536 available addresses) or as small as /28 (16 available addresses) and should not overlap any other network with which they are to be connected.

The Amazon VPC service was released after the Amazon EC2 service; because of this, there are two different networking platforms available within AWS: EC2-Classic and EC2-VPC. Amazon EC2 originally launched with a single, flat network shared with other AWS customers called EC2-Classic. As such, AWS accounts created prior to the arrival of the Amazon VPC service can launch instances into the EC2-Classic network and EC2-VPC. AWS accounts created after December 2013 only support launching instances using EC2-VPC. AWS accounts that support EC2-VPC will have a default VPC created in each region with a default subnet created in each Availability Zone. The assigned CIDR block of the VPC will be 172.31.0.0/16.

Figure 4.1 illustrates an Amazon VPC with an address space of 10.0.0.0/16, two subnets with different address ranges (10.0.0.0/24 and 10.0.1.0/24) placed in different Availability Zones, and a route table with the local route specified.

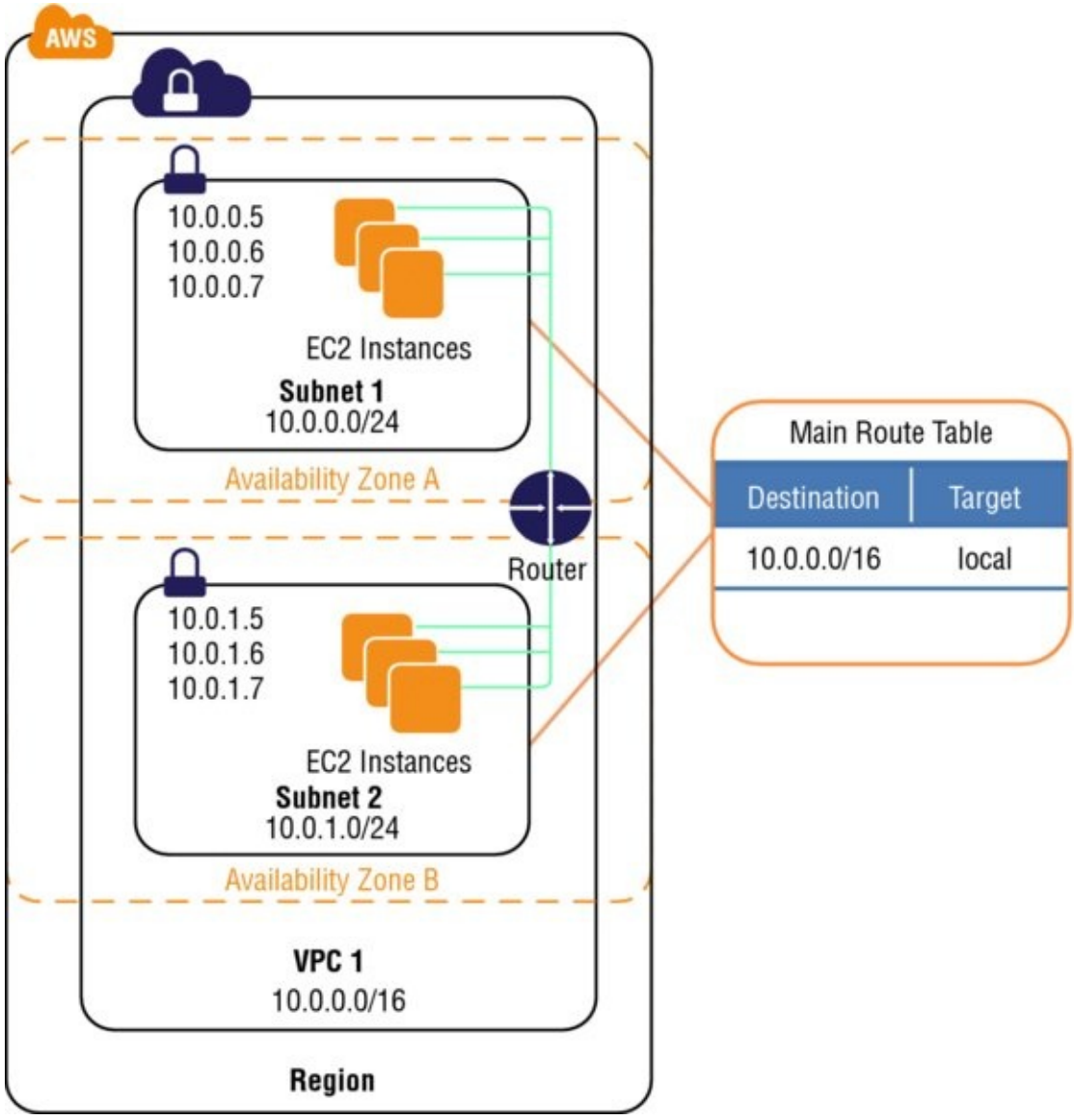

FIGURE 4.1 VPC, subnets, and a route table

An Amazon VPC consists of the following components:

- Subnets
- Route tables
- Dynamic Host Configuration Protocol (DHCP) option sets
- Security groups
- Network Access Control Lists (ACLs)

An Amazon VPC has the following optional components:

- Internet Gateways (IGWs)
- Elastic IP (EIP) addresses
- Elastic Network Interfaces (ENIs)
- Endpoints
- Peering
- Network Address Translation (NATs) instances and NAT gateways

- Virtual Private Gateway (VPG), Customer Gateways (CGWs), and Virtual Private Networks (VPNs)

Subnets

A *subnet* is a segment of an Amazon VPC's IP address range where you can launch Amazon EC2 instances, Amazon Relational Database Service (Amazon RDS) databases, and other AWS resources. CIDR blocks define subnets (for example, 10.0.1.0/24 and 192.168.0.0/24). The smallest subnet that you can create is a /28 (16 IP addresses). AWS reserves the first four IP addresses and the last IP address of every subnet for internal networking purposes. For example, a subnet defined as a /28 has 16 available IP addresses; subtract the 5 IPs needed by AWS to yield 11 IP addresses for your use within the subnet.

After creating an Amazon VPC, you can add one or more subnets in each Availability Zone. Subnets reside within one Availability Zone and cannot span zones. This is an important point that can come up in the exam, so remember that one subnet equals one Availability Zone. You can, however, have multiple subnets in one Availability Zone.

Subnets can be classified as public, private, or VPN-only. A public subnet is one in which the associated route table (discussed later) directs the subnet's traffic to the Amazon VPC's IGW (also discussed later). A private subnet is one in which the associated route table does not direct the subnet's traffic to the Amazon VPC's IGW. A VPN-only subnet is one in which the associated route table directs the subnet's traffic to the Amazon VPC's VPG (discussed later) and does not have a route to the IGW. Regardless of the type of subnet, the internal IP address range of the subnet is always private (that is, non-routable on the Internet).

Default Amazon VPCs contain one public subnet in every Availability Zone within the region, with a netmask of /20.

Route Tables

A *route table* is a logical construct within an Amazon VPC that contains a set of rules (called routes) that are applied to the subnet and used to determine where network traffic is directed. A route table's routes are what permit Amazon EC2 instances within different subnets within an Amazon VPC to communicate with each other. You can modify route tables and add your own custom routes. You can also use route tables to specify which subnets are public (by directing Internet traffic to the IGW) and which subnets are private (by not having a route that directs traffic to the IGW).

Each route table contains a default route called the local route, which enables communication within the Amazon VPC, and this route cannot be modified or removed. Additional routes can be added to direct traffic to exit the Amazon VPC via the IGW (discussed later), the VPG (discussed later), or the NAT instance (discussed later). In the exercises at the end of this chapter, you can practice how this is accomplished.

You should remember the following points about route tables:

- Your VPC has an implicit router.
- Your VPC automatically comes with a main route table that you can modify.
- You can create additional custom route tables for your VPC.
- Each subnet must be associated with a route table, which controls the routing for the subnet. If you don't explicitly associate a subnet with a particular route table, the subnet uses the main route table.
- You can replace the main route table with a custom table that you've created so that each new subnet is automatically associated with it.
- Each route in a table specifies a destination CIDR and a target; for example, traffic destined for 172.16.0.0/12 is targeted for the VPG. AWS uses the most specific route that matches the traffic to determine how to route the traffic.

Internet Gateways

An *Internet Gateway (IGW)* is a horizontally scaled, redundant, and highly available Amazon VPC component that allows communication between instances in your Amazon VPC and the Internet. An IGW provides a target in your Amazon VPC route tables for Internet-routable traffic, and it performs network address translation for instances that have been assigned public IP addresses.

Amazon EC2 instances within an Amazon VPC are only aware of their private IP addresses. When traffic is sent from the instance to the Internet, the IGW translates the reply address to the instance's public IP address (or EIP address, covered later) and maintains the one-to-one map of the instance private IP address and public IP address. When an instance receives traffic from the Internet, the IGW translates the destination address (public IP address) to the instance's private IP address and forwards the traffic to the Amazon VPC.

You must do the following to create a public subnet with Internet access:

- Attach an IGW to your Amazon VPC.
- Create a subnet route table rule to send all non-local traffic (0.0.0.0/0) to the IGW.
- Configure your network ACLs and security group rules to allow relevant traffic to flow to and from your instance.

You must do the following to enable an Amazon EC2 instance to send and receive traffic from the Internet:

- Assign a public IP address or EIP address.

You can scope the route to all destinations not explicitly known to the route table (0.0.0.0/0), or you can scope the route to a narrower range of IP addresses, such as the public IP addresses of your company's public endpoints outside of AWS or the EIP addresses of other Amazon EC2 instances outside your Amazon VPC.

Figure 4.2 illustrates an Amazon VPC with an address space of 10.0.0.0/16, one subnet with an address range of 10.0.0.0/24, a route table, an attached IGW, and a single Amazon EC2 instance with a private IP address and an EIP address. The route table contains two routes: the local route that permits inter-VPC communication and a route that sends all non-local traffic to the IGW (`igw-id`). Note that the Amazon EC2 instance has a public IP address (EIP = 198.51.100.2); this instance can be accessed from the Internet, and traffic may originate and return to this instance.

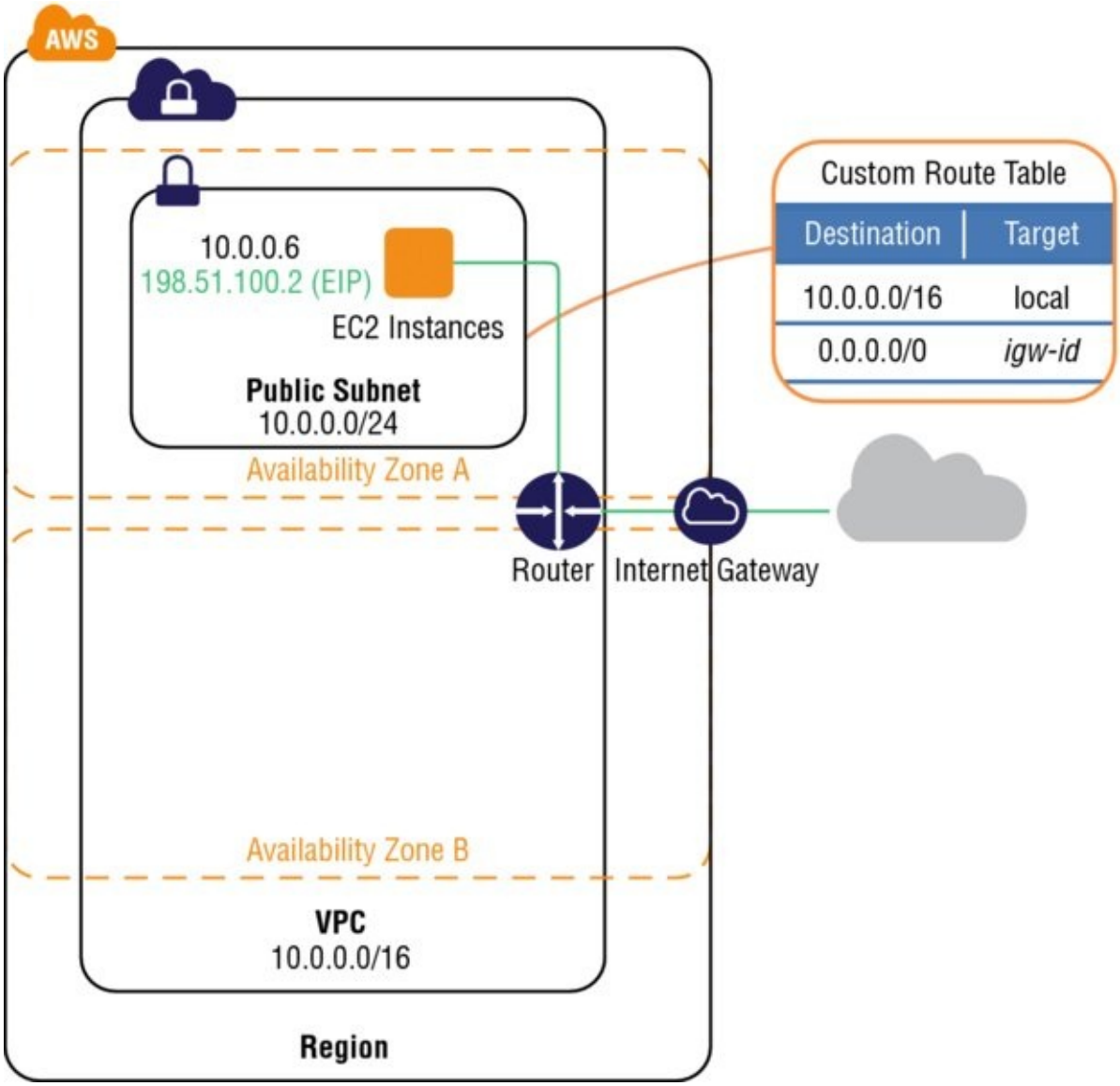

FIGURE 4.2 VPC, subnet, route table, and an Internet gateway

Dynamic Host Configuration Protocol (DHCP) Option Sets

Dynamic Host Configuration Protocol (DHCP) provides a standard for passing configuration information to hosts on a TCP/IP network. The options field of a DHCP message contains the configuration parameters. Some of those parameters are the domain name, domain name server, and the `netbios-node-type`.

AWS automatically creates and associates a DHCP option set for your Amazon VPC upon creation and sets two options: domain-name-servers (defaulted to AmazonProvidedDNS) and domain-name (defaulted to the domain name for your region). AmazonProvidedDNS is an Amazon Domain Name System (DNS) server, and this option enables DNS for instances that need to communicate over the Amazon VPC's IGW.

The DHCP option sets element of an Amazon VPC allows you to direct Amazon EC2 host name assignments to your own resources. To assign your own domain name to your instances, create a custom DHCP option set and assign it to your Amazon VPC. You can configure the following values within a DHCP option set:

- ***domain-name-servers***—The IP addresses of up to four domain name servers, separated by commas. The default is AmazonProvidedDNS.
- ***domain-name***—Specify the desired domain name here (for example, `mycompany.com`).
- ***ntp-servers***—The IP addresses of up to four Network Time Protocol (NTP) servers, separated by commas
- ***netbios-name-servers***—The IP addresses of up to four NetBIOS name servers, separated by commas
- ***netbios-node-type***—Set this value to 2.

Every Amazon VPC must have only one DHCP option set assigned to it.

Elastic IP Addresses (EIPs)

AWS maintains a pool of public IP addresses in each region and makes them available for you to associate to resources within your Amazon VPCs. An *Elastic IP Addresses (EIP)* is a static, public IP address in the pool for the region that you can allocate to your account (pull from the pool) and release (return to the pool). EIPs allow you to maintain a set of IP addresses that remain fixed while the underlying infrastructure may change over time. Here are the important points to understand about EIPs for the exam:

- You must first allocate an EIP for use within a VPC and then assign it to an instance.
- EIPs are specific to a region (that is, an EIP in one region cannot be assigned to an instance within an Amazon VPC in a different region).
- There is a one-to-one relationship between network interfaces and EIPs.
- You can move EIPs from one instance to another, either in the same Amazon VPC or a different Amazon VPC within the same region.
- EIPs remain associated with your AWS account until you explicitly release them.
- There are charges for EIPs allocated to your account, even when they are not associated with a resource.

Elastic Network Interfaces (ENIs)

An *Elastic Network Interface (ENI)* is a virtual network interface that you can attach to an instance in an Amazon VPC. ENIs are only available within an Amazon VPC, and they are associated with a subnet upon creation. They can have one public IP address and multiple private IP addresses. If there are multiple private IP addresses, one of them is primary. Assigning a second network interface to an instance via an ENI allows it to be dual-homed (have network presence in different subnets). An ENI created independently of a particular instance persists regardless of the lifetime of any instance to which it is attached; if an underlying instance fails, the IP address may be preserved by attaching the ENI to a replacement instance.

ENIs allow you to create a management network, use network and security appliances in your Amazon VPC, create dual-homed instances with workloads/roles on distinct subnets, or create a low-budget, high-availability solution.

Endpoints

An Amazon VPC *endpoint* enables you to create a private connection between your Amazon VPC and another AWS service without requiring access over the Internet or through a NAT instance, VPN connection, or AWS Direct Connect. You can create multiple endpoints for a single service, and you can use different route tables to enforce different access policies from different subnets to the same service.

Amazon VPC endpoints currently support communication with Amazon Simple Storage Service (Amazon S3), and other services are expected to be added in the future.

You must do the following to create an Amazon VPC endpoint:

- Specify the Amazon VPC.
- Specify the service. A service is identified by a prefix list of the form `com.amazonaws.<region>.<service>`.
- Specify the policy. You can allow full access or create a custom policy. This policy can be changed at any time.
- Specify the route tables. A route will be added to each specified route table, which will state the service as the destination and the endpoint as the target.

Table 4.1 is an example route table that has an existing route that directs all Internet traffic (0.0.0.0/0) to an IGW. Any traffic from the subnet that is destined for another AWS service (for example, Amazon S3 or Amazon DynamoDB) will be sent to the IGW in order to reach that service.

TABLE 4.1 Route Table with an IGW Routing Rule

Destination	Target
10.0.0.0/16	Local
0.0.0.0/0	igw-1a2b3c4d

Table 4.2 is an example route table that has existing routes directing all Internet traffic to an IGW and all Amazon S3 traffic to the Amazon VPC endpoint.

TABLE 4.2 Route Table with an IGW Routing Rule and VPC Endpoint Rule

Destination	Target
10.0.0.0/16	Local
0.0.0.0/0	igw-1a2b3c4d
pl-1a2b3c4d	vpce-11bb22cc

The route table depicted in Table 4.2 will direct any traffic from the subnet that's destined for Amazon S3 in the same region to the endpoint. All other Internet traffic goes to your IGW, including traffic that's destined for other services and for Amazon S3 in other regions.

Peering

An Amazon VPC *peering* connection is a networking connection between two Amazon VPCs that enables instances in either Amazon VPC to communicate with each other as if they are within the same network. You can create an Amazon VPC peering connection between your own Amazon VPCs or with an Amazon VPC in another AWS account within a single region. A peering connection is neither a gateway nor an Amazon VPN connection and does not introduce a single point of failure for communication.

Peering connections are created through a request/accept protocol. The owner of the requesting Amazon VPC sends a request to peer to the owner of the peer Amazon VPC. If the peer Amazon VPC is within the same account, it is identified by its VPC ID. If the peer VPC is within a different account, it is identified by Account ID and VPC ID. The owner of the peer Amazon VPC has one week to accept or reject the request to peer with the requesting Amazon VPC before the peering request expires.

An Amazon VPC may have multiple peering connections, and peering is a one-to-one relationship between Amazon VPCs, meaning two Amazon VPCs cannot have two peering agreements between them. Also, peering connections do not support transitive routing. Figure 4.3 depicts transitive routing.

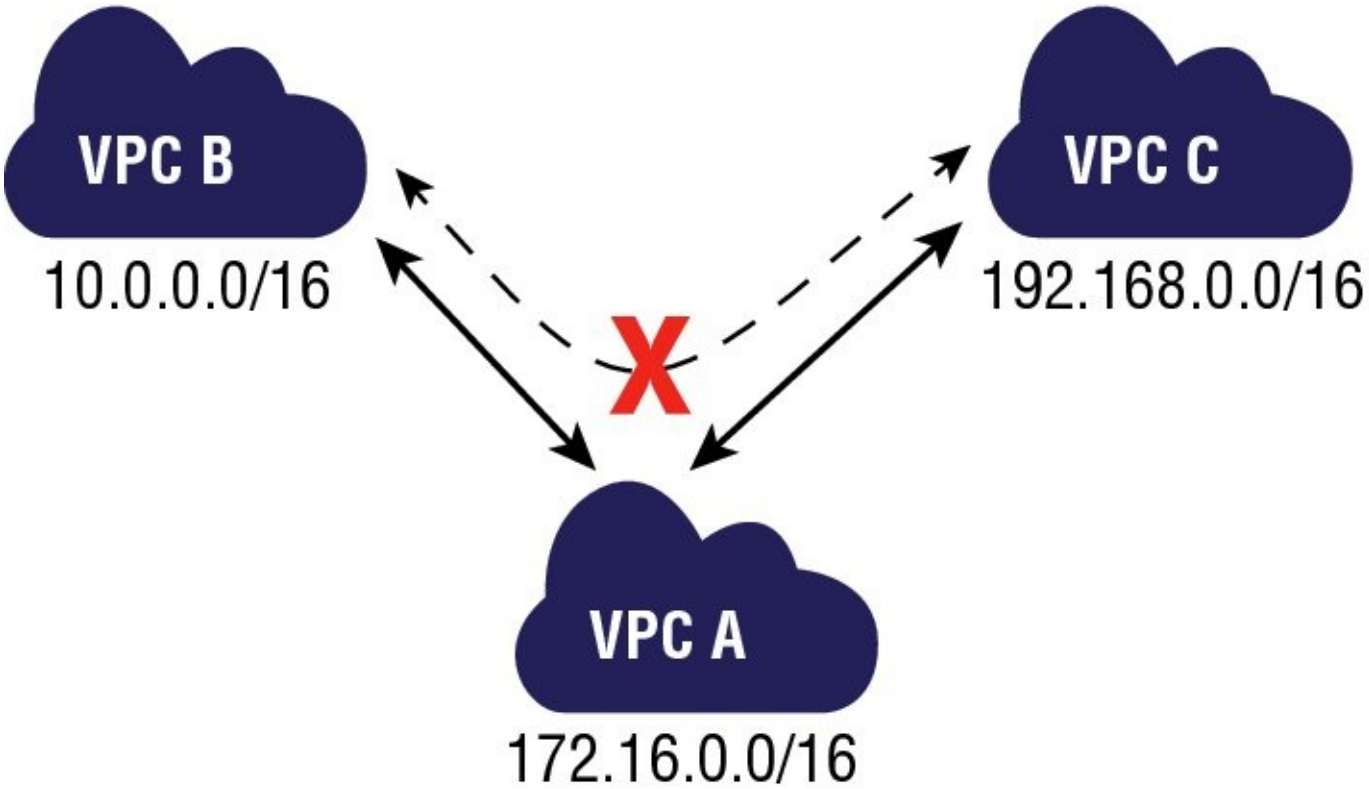

FIGURE 4.3 VPC peering connections do not support transitive routing

In Figure 4.3, VPC A has two peering connections with two different VPCs: VPC B and VPC C. Therefore, VPC A can communicate directly with VPCs B and C. Because peering connections do not support transitive routing, VPC A cannot be a transit point for traffic between VPCs B and C. In order for VPCs B and C to communicate with each other, a peering connection must be explicitly created between them.

Here are the important points to understand about peering for the exam:

- You cannot create a peering connection between Amazon VPCs that have matching or overlapping CIDR blocks.
- You cannot create a peering connection between Amazon VPCs in different regions.
- Amazon VPC peering connections do not support transitive routing.
- You cannot have more than one peering connection between the same two Amazon VPCs at the same time.

Security Groups

A *security group* is a virtual stateful firewall that controls inbound and outbound network traffic to AWS resources and Amazon EC2 instances. All Amazon EC2 instances must be launched into a security group. If a security group is not specified at launch, then the instance will be launched into the default security group for the Amazon VPC. The default security group allows communication between all resources within the security group, allows all outbound traffic, and denies all other traffic. You may change the rules for the default security group, but you may not delete the default security group. Table 4.3 describes the settings of the default security group.

TABLE 4.3 Security Group Rules

Inbound			
Source	**Protocol**	**Port Range**	**Comments**
`sg-xxxxxxxx`	All	All	Allow inbound traffic from instances within the same security group.
Outbound			
Destination	**Protocol**	**Port Range**	**Comments**
`0.0.0.0/0`	All	All	Allow all outbound traffic.

For each security group, you add rules that control the inbound traffic to instances and a separate set of rules that control the outbound traffic. For example, Table 4.4 describes a security group for web servers.

TABLE 4.4 Security Group Rules for a Web Server

Inbound			
Source	**Protocol**	**Port Range**	**Comments**
`0.0.0.0/0`	TCP	80	Allow inbound traffic from the Internet to port 80.
Your network's public IP address range	TCP	22	Allow Secure Shell (SSH) traffic from your company network.
Your network's public IP address range	TCP	3389	Allow Remote Desktop Protocol (RDP) traffic from your company network.
Outbound			
Destination	**Protocol**	**Port Range**	**Comments**
The ID of the security group for your MySQL database servers	TCP	3306	Allow outbound MySQL access to instances in the specified security group.
The ID of the security group for your Microsoft SQL Server database servers	TCP	1433	Allow outbound Microsoft SQL Server access to instances in the specified security group.

Here are the important points to understand about security groups for the exam:

- You can create up to 500 security groups for each Amazon VPC.
- You can add up to 50 inbound and 50 outbound rules to each security group. If you need to apply more than 100 rules to an instance, you can associate up to five security groups with each network interface.
- You can specify allow rules, but not deny rules. This is an important difference between security groups and ACLs.
- You can specify separate rules for inbound and outbound traffic.
- By default, no inbound traffic is allowed until you add inbound rules to the security group.
- By default, new security groups have an outbound rule that allows all outbound traffic. You can remove the rule and add outbound rules that allow specific outbound traffic only.
- Security groups are stateful. This means that responses to allowed inbound traffic are allowed to flow outbound regardless of outbound rules and vice versa. This is an important difference between security groups and network ACLs.
- Instances associated with the same security group can't talk to each other unless you add rules allowing it (with the exception being the default security group).
- You can change the security groups with which an instance is associated after launch,

and the changes will take effect immediately.

Network Access Control Lists (ACLs)

A network *access control list (ACL)* is another layer of security that acts as a stateless firewall on a subnet level. A network ACL is a numbered list of rules that AWS evaluates in order, starting with the lowest numbered rule, to determine whether traffic is allowed in or out of any subnet associated with the network ACL. Amazon VPCs are created with a modifiable default network ACL associated with every subnet that allows all inbound and outbound traffic. When you create a custom network ACL, its initial configuration will deny all inbound and outbound traffic until you create rules that allow otherwise. You may set up network ACLs with rules similar to your security groups in order to add a layer of security to your Amazon VPC, or you may choose to use the default network ACL that does not filter traffic traversing the subnet boundary. Overall, every subnet must be associated with a network ACL.

Table 4.5 explains the differences between a security group and a network ACL. You should remember the following differences between security groups and network ACLs for the exam.

TABLE 4.5 Comparison of Security Groups and Network ACLs

Security Group	Network ACL
Operates at the instance level (first layer of defense)	Operates at the subnet level (second layer of defense)
Supports allow rules only	Supports allow rules and deny rules
Stateful: Return traffic is automatically allowed, regardless of any rules	Stateless: Return traffic must be explicitly allowed by rules.
AWS evaluates all rules before deciding whether to allow traffic	AWS processes rules in number order when deciding whether to allow traffic.
Applied selectively to individual instances	Automatically applied to all instances in the associated subnets; this is a backup layer of defense, so you don't have to rely on someone specifying the security group.

Network Address Translation (NAT) Instances and NAT Gateways

By default, any instance that you launch into a private subnet in an Amazon VPC is not able to communicate with the Internet through the IGW. This is problematic if the instances within private subnets need direct access to the Internet from the Amazon VPC in order to apply security updates, download patches, or update application software. AWS provides NAT instances and NAT gateways to allow instances deployed in private subnets to gain Internet access. For common use cases, we recommend that you use a NAT gateway instead of a NAT instance. The NAT gateway provides better availability and higher bandwidth, and requires less administrative effort than NAT instances.

NAT Instance

A *network address translation (NAT) instance* is an Amazon Linux Amazon Machine Image (AMI) that is designed to accept traffic from instances within a private subnet, translate the source IP address to the public IP address of the NAT instance, and forward the traffic to the IGW. In addition, the NAT instance maintains the state of the forwarded traffic in order to return response traffic from the Internet to the proper instance in the private subnet. These instances have the string `amzn-ami-vpc-nat` in their names, which is searchable in the Amazon EC2 console.

To allow instances within a private subnet to access Internet resources through the IGW via a NAT instance, you must do the following:

- Create a security group for the NAT with outbound rules that specify the needed Internet resources by port, protocol, and IP address.
- Launch an Amazon Linux NAT AMI as an instance in a public subnet and associate it with the NAT security group.
- Disable the Source/Destination Check attribute of the NAT.
- Configure the route table associated with a private subnet to direct Internet-bound traffic to the NAT instance (for example, `i-1a2b3c4d`).
- Allocate an EIP and associate it with the NAT instance.

This configuration allows instances in private subnets to send outbound Internet communication, but it prevents the instances from receiving inbound traffic initiated by someone on the Internet.

NAT Gateway

A *NAT gateway* is an Amazon managed resource that is designed to operate just like a NAT instance, but it is simpler to manage and highly available within an Availability Zone.

To allow instances within a private subnet to access Internet resources through the IGW via a NAT gateway, you must do the following:

- Configure the route table associated with the private subnet to direct Internet-bound

traffic to the NAT gateway (for example, `nat-1a2b3c4d`).

- Allocate an EIP and associate it with the NAT gateway.

Like a NAT instance, this managed service allows outbound Internet communication and prevents the instances from receiving inbound traffic initiated by someone on the Internet.

 To create an Availability Zone-independent architecture, create a NAT gateway in each Availability Zone and configure your routing to ensure that resources use the NAT gateway in the same Availability Zone.

The exercises will demonstrate how a NAT gateway works.

Virtual Private Gateways (VPGs), Customer Gateways (CGWs), and Virtual Private Networks (VPNs)

You can connect an existing data center to Amazon VPC using either hardware or software VPN connections, which will make Amazon VPC an extension of the data center. Amazon VPC offers two ways to connect a corporate network to a VPC: VPG and CGW.

A *virtual private gateway (VPG)* is the *virtual private network (VPN)* concentrator on the AWS side of the VPN connection between the two networks. A *customer gateway (CGW)* represents a physical device or a software application on the customer's side of the VPN connection. After these two elements of an Amazon VPC have been created, the last step is to create a VPN tunnel. The VPN tunnel is established after traffic is generated from the customer's side of the VPN connection. Figure 4.4 illustrates a single VPN connection between a corporate network and an Amazon VPC.

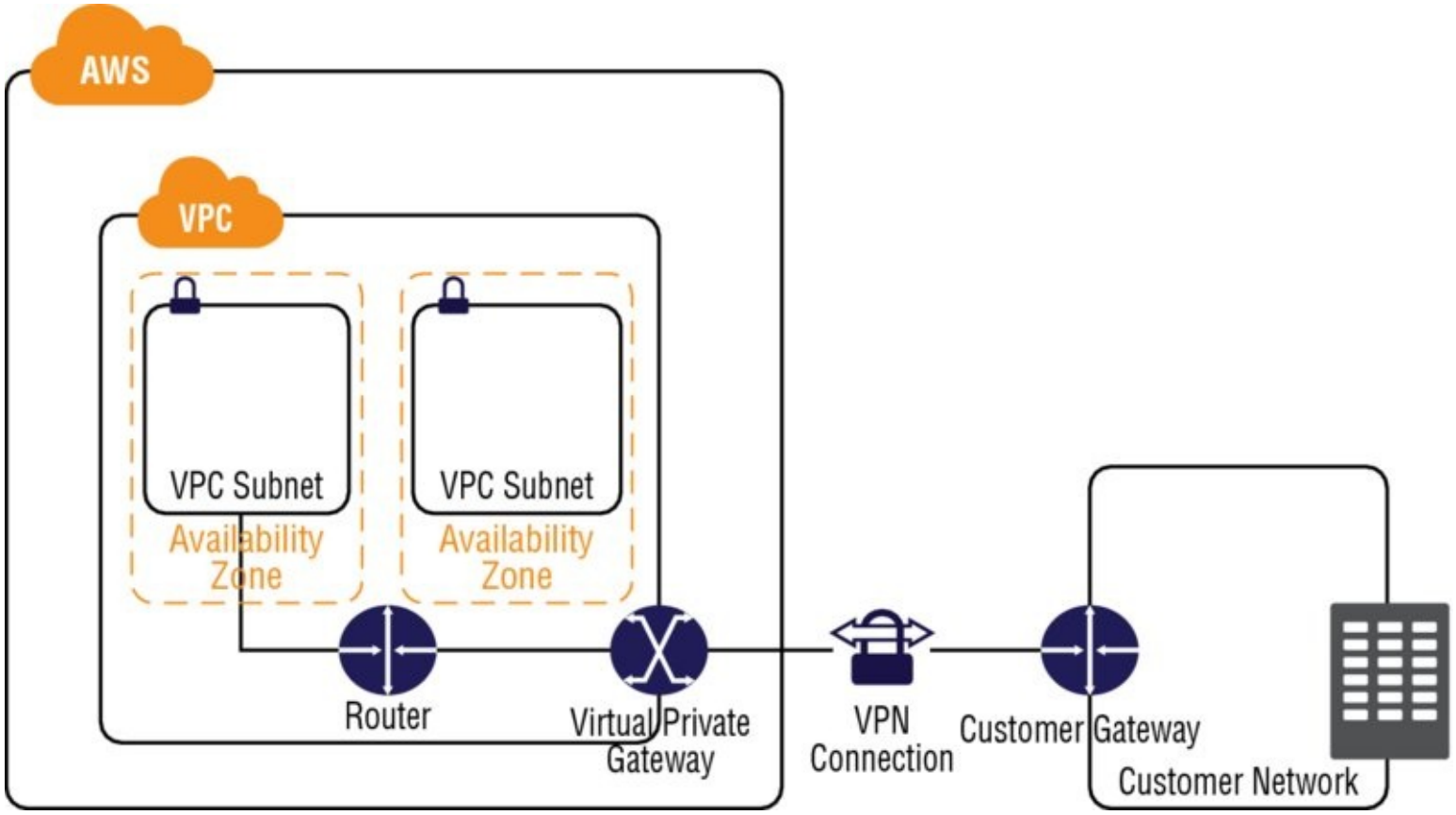

FIGURE 4.4 VPC with VPN connection to a customer network

You must specify the type of routing that you plan to use when you create a VPN connection. If the CGW supports Border Gateway Protocol (BGP), then configure the VPN connection for dynamic routing. Otherwise, configure the connections for static routing. If you will be using static routing, you must enter the routes for your network that should be communicated to the VPG. Routes will be propagated to the Amazon VPC to allow your resources to route network traffic back to the corporate network through the VGW and across the VPN tunnel.

Amazon VPC also supports multiple CGWs, each having a VPN connection to a single VPG (many-to-one design). In order to support this topology, the CGW IP addresses must be unique within the region.

Amazon VPC will provide the information needed by the network administrator to configure the CGW and establish the VPN connection with the VPG. The VPN connection consists of two Internet Protocol Security (IPSec) tunnels for higher availability to the Amazon VPC.

Following are the important points to understand about VPGs, CGWs, and VPNs for the exam:

- The VPG is the AWS end of the VPN tunnel.
- The CGW is a hardware or software application on the customer's side of the VPN tunnel.
- You must initiate the VPN tunnel from the CGW to the VPG.
- VPGs support both dynamic routing with BGP and static routing.
- The VPN connection consists of two tunnels for higher availability to the VPC.

Summary

In this chapter, you learned that Amazon VPC is the networking layer for Amazon EC2, and it allows you to create your own private virtual network within the cloud. You can provision your own logically isolated section of AWS similar to designing and implementing a separate independent network that you'd operate in a physical data center.

A VPC consists of the following components:

- Subnets
- Route tables
- DHCP option sets
- Security groups
- Network ACLs

A VPC has the following optional components:

- IGWs
- EIP addresses
- Endpoints
- Peering
- NAT instance and NAT gateway
- VPG, CGW, and VPN

Subnets can be public, private, or VPN-only. A public subnet is one in which the associated route table directs the subnet's traffic to the Amazon VPC's IGW. A private subnet is one in which the associated route table does not direct the subnet's traffic to the Amazon VPC's IGW. A VPN-only subnet is one in which the associated route table directs the subnet's traffic to the Amazon VPC's VPG and does not have a route to the IGW. Regardless of the type of subnet, the internal IP address range of the subnet is always private (non-routable on the Internet).

A route table is a logical construct within an Amazon VPC that contains a set of rules (called routes) that are applied to the subnet and used to determine where network traffic is directed. A route table's routes are what permit Amazon EC2 instances within different subnets within an Amazon VPC to communicate with each other. You can modify route tables and add your own custom routes. You can also use route tables to specify which subnets are public (by directing Internet traffic to the IGW) and which subnets are private (by not having a route that directs traffic to the IGW). An IGW is a horizontally scaled, redundant, and highly available Amazon VPC component that allows communication between instances in your Amazon VPC and the Internet. IGWs are fully redundant and have no bandwidth constraints. An IGW provides a target in your Amazon VPC route tables for Internet-routable traffic, and it performs network address translation for instances that have been assigned public IP addresses.

The DHCP option sets element of an Amazon VPC allows you to direct Amazon EC2 host

name assignment to your own resources. In order for you to assign your own domain name to your instances, you create a custom DHCP option set and assign it to your Amazon VPC.

An EIP address is a static, public IP address in the pool for the region that you can allocate to your account (pull from the pool) and release (return to the pool). EIPs allow you to maintain a set of IP addresses that remain fixed while the underlying infrastructure may change over time.

An Amazon VPC endpoint enables you to create a private connection between your Amazon VPC and another AWS service without requiring access over the Internet or through a NAT instance, VPN connection, or AWS Direct Connect. You can create multiple endpoints for a single service, and you can use different route tables to enforce different access policies from different subnets to the same service.

An Amazon VPC peering connection is a networking connection between two Amazon VPCs that enables instances in either Amazon VPC to communicate with each other as if they were within the same network. You can create an Amazon VPC peering connection between your own Amazon VPCs or with an Amazon VPC in another AWS account within a single region. A peering connection is neither a gateway nor a VPN connection and does not introduce a single point of failure for communication.

A security group is a virtual stateful firewall that controls inbound and outbound traffic to Amazon EC2 instances. When you first launch an Amazon EC2 instance into an Amazon VPC, you must specify the security group with which it will be associated. AWS provides a default security group for your use, which has rules that allow all instances associated with the security group to communicate with each other and allow all outbound traffic. You may change the rules for the default security group, but you may not delete the default security group.

A network ACL is another layer of security that acts as a stateless firewall on a subnet level. Amazon VPCs are created with a modifiable default network ACL associated with every subnet that allows all inbound and outbound traffic. If you want to create a custom network ACL, its initial configuration will deny all inbound and outbound traffic until you create a rule that states otherwise.

A NAT instance is a customer-managed instance that is designed to accept traffic from instances within a private subnet, translate the source IP address to the public IP address of the NAT instance, and forward the traffic to the IGW. In addition, the NAT instance maintains the state of the forwarded traffic in order to return response traffic from the Internet to the proper instance in the private subnet.

A NAT gateway is an AWS-managed service that is designed to accept traffic from instances within a private subnet, translate the source IP address to the public IP address of the NAT gateway, and forward the traffic to the IGW. In addition, the NAT gateway maintains the state of the forwarded traffic in order to return response traffic from the Internet to the proper instance in the private subnet.

A VPG is the VPN concentrator on the AWS side of the VPN connection between the two networks. A CGW is a physical device or a software application on the customer's side of the VPN connection. After these two elements of an Amazon VPC have been created, the last step is to create a VPN tunnel. The VPN tunnel is established after traffic is generated from the

customer’s side of the VPN connection.

Exam Essentials

Understand what a VPC is and its core and optional components. An Amazon VPC is a logically isolated network in the AWS Cloud. An Amazon VPC is made up of the following core elements: subnets (public, private, and VPN-only), route tables, DHCP option sets, security groups, and network ACLs. Optional elements include an IGW, EIP addresses, endpoints, peering connections, NAT instances, VPGs, CGWs, and VPN connections.

Understand the purpose of a subnet. A subnet is a segment of an Amazon VPC's IP address range where you can place groups of isolated resources. Subnets are defined by CIDR blocks—for example, 10.0.1.0/24 and 10.0.2.0/24—and are contained within an Availability Zone.

Identify the difference between a public subnet, a private subnet, and a VPN-Only subnet. If a subnet's traffic is routed to an IGW, the subnet is known as a public subnet. If a subnet doesn't have a route to the IGW, the subnet is known as a private subnet. If a subnet doesn't have a route to the IGW, but has its traffic routed to a VPG, the subnet is known as a VPN-only subnet.

Understand the purpose of a route table. A route table is a set of rules (called routes) that are used to determine where network traffic is directed. A route table allows Amazon EC2 instances within different subnets to communicate with each other (within the same Amazon VPC). The Amazon VPC router also enables subnets, IGWs, and VPGs to communicate with each other.

Understand the purpose of an IGW. An IGW is a horizontally scaled, redundant, and highly available Amazon VPC component that allows communication between instances in your Amazon VPC and the Internet. IGWs are fully redundant and have no bandwidth constraints. An IGW provides a target in your Amazon VPC route tables for Internet-routable traffic and performs network address translation for instances that have been assigned public IP addresses.

Understand what DHCP option sets provide to an Amazon VPC. The DHCP option sets element of an Amazon VPC allows you to direct Amazon EC2 host name assignment to your own resources. You can specify the domain name for instances within an Amazon VPC and identify the IP addresses of custom DNS servers, NTP servers, and NetBIOS servers.

Know the difference between an Amazon VPC public IP address and an EIP address. A public IP address is an AWS-owned IP that can be automatically assigned to instances launched within a subnet. An EIP address is an AWS-owned public IP address that you allocate to your account and assign to instances or network interfaces on demand.

Understand what endpoints provide to an Amazon VPC. An Amazon VPC endpoint enables you to create a private connection between your Amazon VPC and another AWS service without requiring access over the Internet or through a NAT instance, a VPN connection, or AWS Direct Connect. Endpoints support services within the region only.

Understand Amazon VPC peering. An Amazon VPC peering connection is a networking connection between two Amazon VPCs that enables instances in either Amazon VPC to communicate with each other as if they are within the same network. Peering connections

are created through a request/accept protocol. Transitive peering is not supported, and peering is only available between Amazon VPCs within the same region.

Know the difference between a security group and a network ACL. A security group applies at the instance level. You can have multiple instances in multiple subnets that are members of the same security groups. Security groups are stateful, which means that return traffic is automatically allowed, regardless of any outbound rules. A network ACL is applied on a subnet level, and traffic is stateless. You need to allow both inbound and outbound traffic on the network ACL in order for Amazon EC2 instances in a subnet to be able to communicate over a particular protocol.

Understand what a NAT provides to an Amazon VPC. A NAT instance or NAT gateway enables instances in a private subnet to initiate outbound traffic to the Internet. This allows outbound Internet communication to download patches and updates, for example, but prevents the instances from receiving inbound traffic initiated by someone on the Internet.

Understand the components needed to establish a VPN connection from a network to an Amazon VPC. A VPG is the VPN concentrator on the AWS side of the VPN connection between the two networks. A CGW represents a physical device or a software application on the customer's side of the VPN connection. The VPN connection must be initiated from the CGW side, and the connection consists of two IPSec tunnels.

Exercises

The best way to become familiar with Amazon VPC is to build your own custom Amazon VPC and then deploy Amazon EC2 instances into it, which is what you'll be doing in this section. You should repeat these exercises until you can create and decommission Amazon VPCs with confidence.

For assistance completing these exercises, refer to the Amazon VPC User Guide located at `http://aws.amazon.com/documentation/vpc/`.

EXERCISE 4.1

Create a Custom Amazon VPC

1. Sign in to the AWS Management Console as an administrator or power user.
2. Select the Amazon VPC icon to launch the Amazon VPC Dashboard.
3. Create an Amazon VPC with a CIDR block equal to `192.168.0.0/16`, a name tag of **`My First VPC`**, and default tenancy.

You have created your first custom VPC.

EXERCISE 4.2

Create Two Subnets for Your Custom Amazon VPC

1. Create a subnet with a CIDR block equal to `192.168.1.0/24` and a name tag of **`My First Public Subnet`**. Create the subnet in the Amazon VPC from Exercise 4.1, and specify an Availability Zone for the subnet (for example, `US-East-1a`).
2. Create a subnet with a CIDR block equal to `192.168.2.0/24` and a name tag of **`My First Private Subnet`**. Create the subnet in the Amazon VPC from Exercise 4.1, and specify a different Availability Zone for the subnet than previously specified (for example, `US-East-1b`).

You have now created two new subnets, each in its own Availability Zone. It's important to remember that one subnet equals one Availability Zone. You cannot stretch a subnet across multiple Availability Zones.

EXERCISE 4.3

Connect Your Custom Amazon VPC to the Internet and Establish Routing

For assistance with this exercise, refer to the Amazon EC2 key pair documentation at:

http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/ec2-key-pairs.html

For additional assistance with this exercise, refer to the NAT instances documentation at:

http://docs.aws.amazon.com/AmazonVPC/latest/UserGuide/VPC_NAT_Instance.html#NATInstance

1. Create an Amazon EC2 key pair in the same region as your custom Amazon VPC.
2. Create an IGW with a name tag of **`My First IGW`** and attach it to your custom Amazon VPC.
3. Add a route to the main route table for your custom Amazon VPC that directs Internet traffic (0.0.0.0/0) to the IGW.
4. Create a NAT gateway, place it in the public subnet of your custom Amazon VPC, and assign it an EIP.
5. Create a new route table with a name tag of **`My First Private Route Table`** and place it within your custom Amazon VPC. Add a route to it that directs Internet traffic (0.0.0.0/0) to the NAT gateway and associate it with the private subnet.

You have now created a connection to the Internet for resources within your Amazon VPC. You established routing rules that direct Internet traffic to the IGW regardless of the originating subnet.

EXERCISE 4.4

Launch an Amazon EC2 Instance and Test the Connection to the Internet

1. Launch a t2.micro Amazon Linux AMI as an Amazon EC2 instance into the public subnet of your custom Amazon VPC, give it a name tag of **`My First Public Instance`**, and select the newly-created key pair for secure access to the instance.
2. Securely access the Amazon EC2 instance in the public subnet via SSH with the newly-created key pair.
3. Execute an `update` to the operating system instance libraries by executing the following command:

   ```
   # sudo yum update -y
   ```

4. You should see output showing the instance downloading software from the Internet and installing it.

You have now provisioned an Amazon EC2 instance in a public subnet. You can apply patches to the Amazon EC2 instance in the public subnet, and you have demonstrated connectivity to the Internet.

Review Questions

1. What is the minimum size subnet that you can have in an Amazon VPC?
   A. /24
   B. /26
   C. /28
   D. /30
2. You are a solutions architect working for a large travel company that is migrating its existing server estate to AWS. You have recommended that they use a custom Amazon VPC, and they have agreed to proceed. They will need a public subnet for their web servers and a private subnet in which to place their databases. They also require that the web servers and database servers be highly available and that there be a minimum of two web servers and two database servers each. How many subnets should you have to maintain high availability?
   A. 2
   B. 3
   C. 4
   D. 1
3. Which of the following is an optional security control that can be applied at the subnet layer of a VPC?
   A. Network ACL
   B. Security Group
   C. Firewall
   D. Web application firewall
4. What is the maximum size IP address range that you can have in an Amazon VPC?
   A. /16
   B. /24
   C. /28
   D. /30
5. You create a new subnet and then add a route to your route table that routes traffic out from that subnet to the Internet using an IGW. What type of subnet have you created?
   A. An internal subnet
   B. A private subnet
   C. An external subnet
   D. A public subnet

6. What happens when you create a new Amazon VPC?
   A. A main route table is created by default.
   B. Three subnets are created by default—one for each Availability Zone.
   C. Three subnets are created by default in one Availability Zone.
   D. An IGW is created by default.
7. You create a new VPC in `US-East-1` and provision three subnets inside this Amazon VPC. Which of the following statements is true?
   A. By default, these subnets will not be able to communicate with each other; you will need to create routes.
   B. All subnets are public by default.
   C. All subnets will be able to communicate with each other by default.
   D. Each subnet will have identical CIDR blocks.
8. How many IGWs can you attach to an Amazon VPC at any one time?
   A. 1
   B. 2
   C. 3
   D. 4
9. What aspect of an Amazon VPC is stateful?
   A. Network ACLs
   B. Security groups
   C. Amazon DynamoDB
   D. Amazon S3
10. You have created a custom Amazon VPC with both private and public subnets. You have created a NAT instance and deployed this instance to a public subnet. You have attached an EIP address and added your NAT to the route table. Unfortunately, instances in your private subnet still cannot access the Internet. What may be the cause of this?
    A. Your NAT is in a public subnet, but it needs to be in a private subnet.
    B. Your NAT should be behind an Elastic Load Balancer.
    C. You should disable source/destination checks on the NAT.
    D. Your NAT has been deployed on a Windows instance, but your other instances are Linux. You should redeploy the NAT onto a Linux instance.
11. Which of the following will occur when an Amazon Elastic Block Store (Amazon EBS)-backed Amazon EC2 instance in an Amazon VPC with an associated EIP is stopped and started? (Choose 2 answers)
    A. The EIP will be dissociated from the instance.

B. All data on instance-store devices will be lost.

C. All data on Amazon EBS devices will be lost.

D. The ENI is detached.

E. The underlying host for the instance is changed.

12. How many VPC Peering connections are required for four VPCs located within the same AWS region to be able to send traffic to each of the others?

A. 3

B. 4

C. 5

D. 6

13. Which of the following AWS resources would you use in order for an EC2-VPC instance to resolve DNS names outside of AWS?

A. A VPC peering connection

B. A DHCP option set

C. A routing rule

D. An IGW

14. Which of the following is the Amazon side of an Amazon VPN connection?

A. An EIP

B. A CGW

C. An IGW

D. A VPG

15. What is the default limit for the number of Amazon VPCs that a customer may have in a region?

A. 5

B. 6

C. 7

D. There is no default maximum number of VPCs within a region.

16. You are responsible for your company's AWS resources, and you notice a significant amount of traffic from an IP address in a foreign country in which your company does not have customers. Further investigation of the traffic indicates the source of the traffic is scanning for open ports on your EC2-VPC instances. Which one of the following resources can deny the traffic from reaching the instances?

A. Security group

B. Network ACL

C. NAT instance

D. An Amazon VPC endpoint

17. Which of the following is the security protocol supported by Amazon VPC?
    A. SSH
    B. Advanced Encryption Standard (AES)
    C. Point-to-Point Tunneling Protocol (PPTP)
    D. IPsec
18. Which of the following Amazon VPC resources would you use in order for EC2-VPC instances to send traffic directly to Amazon S3?
    A. Amazon S3 gateway
    B. IGW
    C. CGW
    D. VPC endpoint
19. What properties of an Amazon VPC must be specified at the time of creation? (Choose 2 answers)
    A. The CIDR block representing the IP address range
    B. One or more subnets for the Amazon VPC
    C. The region for the Amazon VPC
    D. Amazon VPC Peering relationships
20. Which Amazon VPC feature allows you to create a dual-homed instance?
    A. EIP address
    B. ENI
    C. Security groups
    D. CGW