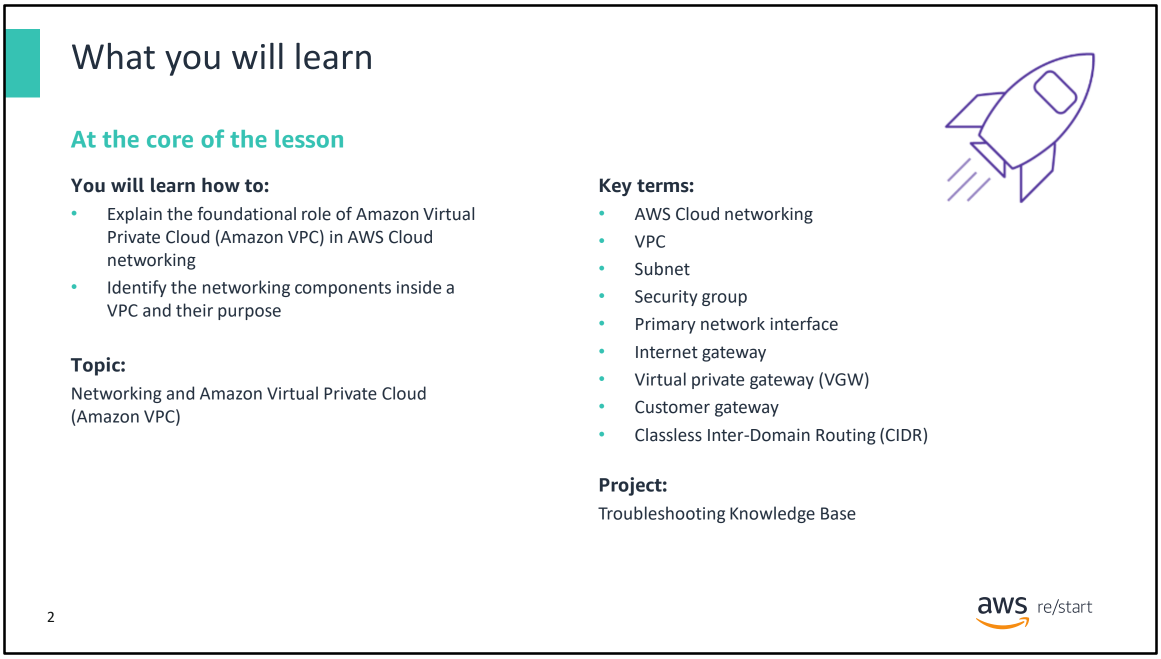
aws re/start

AWS Cloud Networking and Amazon Virtual Private Cloud

Networking

Welcome to AWS Cloud Networking and Amazon Virtual Private Cloud. This lesson will introduce cloud networking and Amazon Virtual Private Cloud (Amazon VPC).



What you will learn

At the core of the lesson

You will learn how to:

- Explain the foundational role of Amazon Virtual Private Cloud (Amazon VPC) in AWS Cloud networking
- Identify the networking components inside a VPC and their purpose

Topic:

Networking and Amazon Virtual Private Cloud (Amazon VPC)

Key terms:

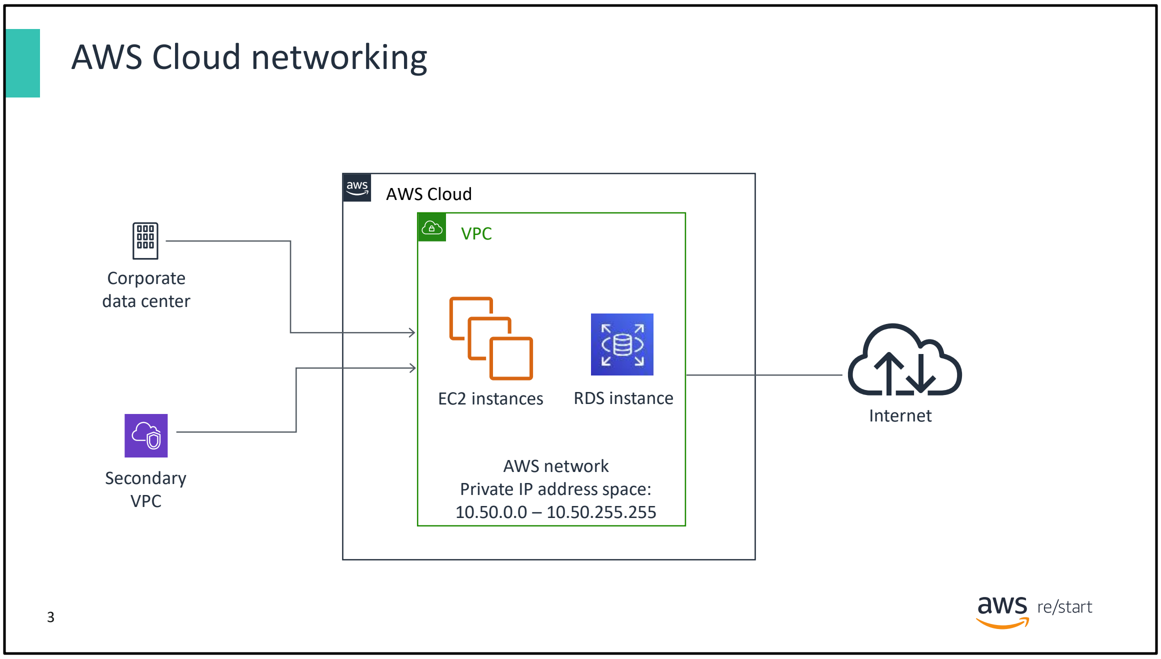
- AWS Cloud networking
- VPC
- Subnet
- Security group
- Primary network interface
- Internet gateway
- Virtual private gateway (VGW)
- Customer gateway
- Classless Inter-Domain Routing (CIDR)

Project:

Troubleshooting Knowledge Base

At the end of this module, you will be able to:

- Explain the foundational role of Amazon Virtual Private Cloud (VPC) in AWS Cloud networking.
- Identify the networking components inside a VPC and their purpose.



AWS Cloud networking

Corporate
data center

Secondary
VPC

AWS Cloud

VPC

EC2 instances

RDS instance

AWS network
Private IP address space:
10.50.0.0 – 10.50.255.255

Internet

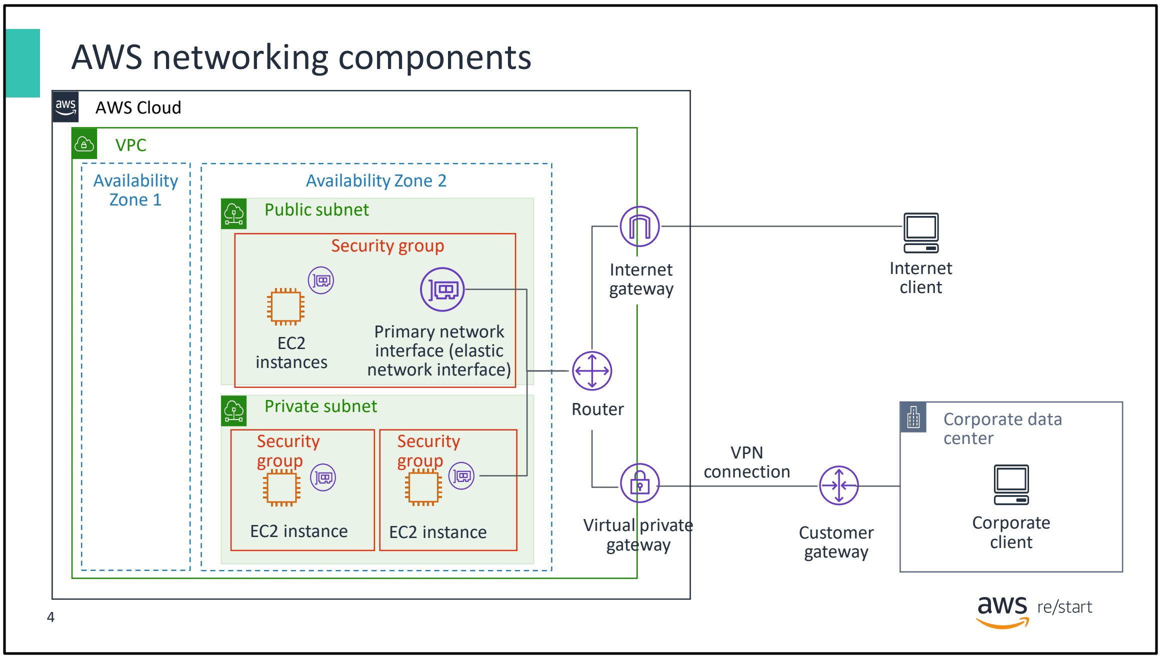
In its most basic form, a cloud-based network is a private IP address space where you can deploy computing resources. In Amazon Web Services (AWS), a *virtual private cloud (VPC)* component provides this private network space. A VPC enables you to define a virtual network in your own logically isolated area within the AWS Cloud. Inside this virtual network, you can deploy AWS computing resources. These resources include, for example, Amazon Elastic Compute Cloud (Amazon EC2) or Amazon Relational Database Service (Amazon RDS) instances. You can also define how—and whether—your private network space connects to endpoints in your network topology.

In the example, a VPC contains three EC2 instances and an RDS instance. It is connected to the internet and the corporate data center. It is also connected to a secondary VPC.

What networking components can be defined in a VPC?

As a systems operator, you have control over these components. Thus, it is important that you understand how to configure and use them. You might have been previously introduced to many of them before. However, a brief review of what they are and how they relate to each other is beneficial before you examine them in more detail.

You review this topic next.



AWS networking components

4

This diagram shows the main AWS networking components that reside in a VPC. It also highlights the key components that enable it to connect to external clients.

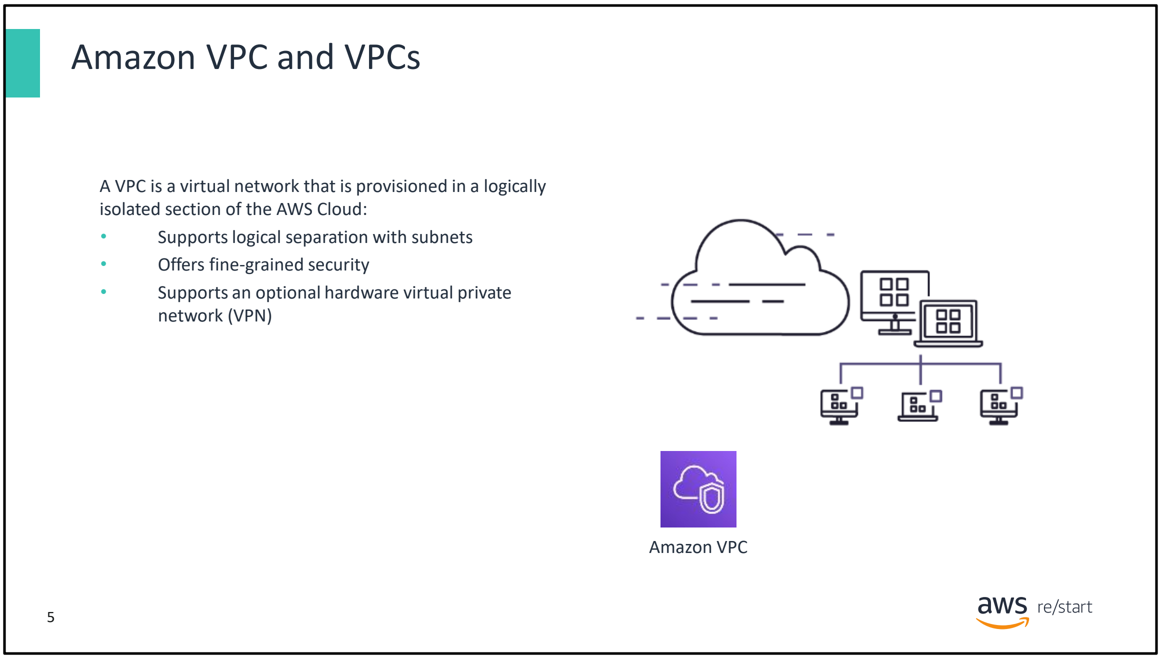
A VPC can span multiple Availability Zones, and its key component types include:

- *Subnet* – Subnets are logical network segments within your VPC. They enable you to subdivide your VPC network into smaller networks inside a single Availability Zone. A subnet is *public* if it is attached to an internet gateway, or *private* if it is not. A subnet is required to deploy an instance into a VPC.
- *Security group* – A security group is a set of *firewall* rules that secure instances. They allow or block inbound and outbound traffic into an instance (stateful). If you do not specify a particular group at launch time, an instance is automatically assigned to the default security group for the VPC. A security group is associated with an instance.
- *Primary network interface (elastic network interface)* – An elastic network interface is a virtual network interface (NIC) that connects an instance to a network. Each instance in a VPC has a default network interface, the *primary network interface*,

which cannot be detached from the instance.

- *Router* – A router is a component that routes traffic within the VPC.
- *Internet gateway* – An internet gateway is a VPC component that enables communication between instances in a VPC and the internet.
- *Virtual private gateway* – A virtual private gateway is the component that is defined on the *AWS side* of a *virtual private network (VPN)* connection. A VPN connection provides a secure and encrypted tunnel between two network endpoints.
- *Customer gateway* – A customer gateway is a physical device or software application that is defined *on the client side* of a VPN connection.

Throughout the rest of this topic, you examine the purpose and use of these networking components in more detail.



Amazon VPC and VPCs

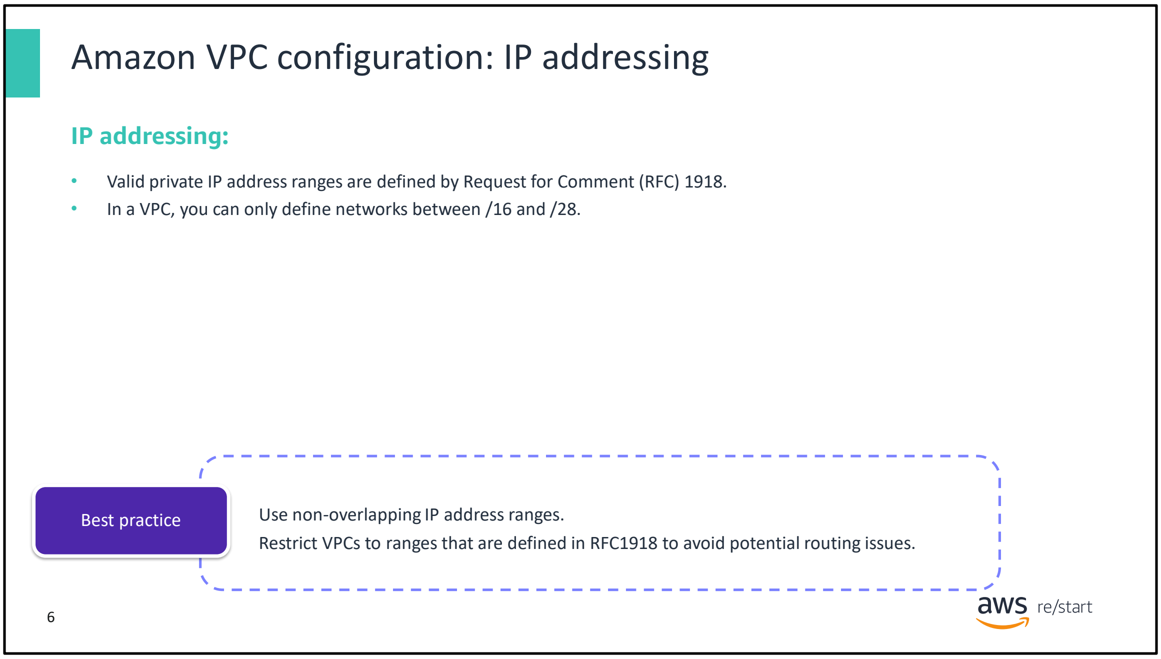
A VPC is a virtual network that is provisioned in a logically isolated section of the AWS Cloud:

- Supports logical separation with subnets
- Offers fine-grained security
- Supports an optional hardware virtual private network (VPN)

This slide shows the formal definition of a VPC and a description of some of its main characteristics.

A VPC is an isolated portion of the AWS Cloud. You provision a VPC so that you can deploy AWS infrastructure services. It is a virtual network and, as such, it supports multiple subnets, routing, and fine-grained security mechanisms.

When you create a VPC, you define its IP address range, subnets, and route tables. You can also optionally use network gateways or hardware VPN solutions to securely connect it to on-premises corporate networks.



Amazon VPC configuration: IP addressing

IP addressing:

- Valid private IP address ranges are defined by Request for Comment (RFC) 1918.
- In a VPC, you can only define networks between /16 and /28.

Best practice

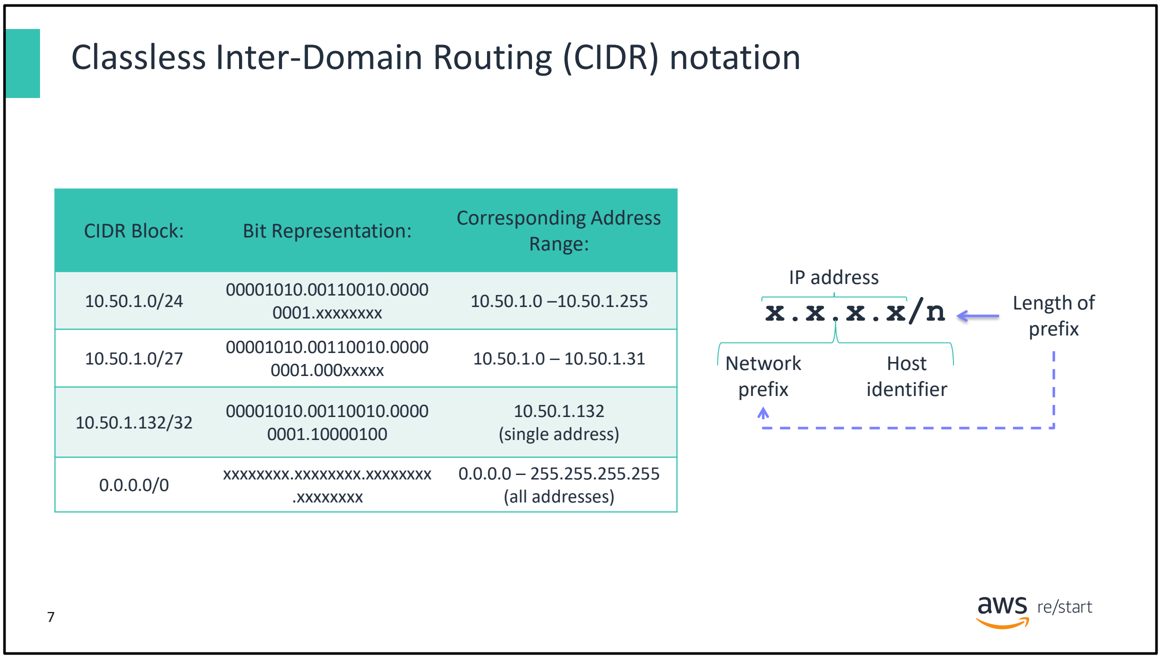
Use non-overlapping IP address ranges.
Restrict VPCs to ranges that are defined in RFC1918 to avoid potential routing issues.

6

When you create a VPC, you must specify its allowable IP address range. This range represents IPv4 addresses, and it is expressed in the form of a Classless Inter-Domain Routing (CIDR) block. This required range is known as the VPC's *primary* CIDR block. After the VPC is created, you can optionally add up to four *secondary* CIDR blocks to it. You can only define VPCs with CIDR blocks of /16 through /28 inclusive, which means that a VPC can contain 16 – 65,536 IP addresses.

As a best practice, restrict your range to addresses that are specified in the standards document Request for Comment (RFC) 1918. This document defines the address ranges that private networks can use. This standard exists to prevent potential IP collisions between resources in your private network, and public resources that might be available on the internet. In addition, it is recommended that you do not use a range that overlaps with addresses that other VPCs use. Doing so will stop you from connecting these VPCs to a common home network via a hardware VPN connection.

You can also optionally assign an IPv6 CIDR block that AWS provides to a VPC. The CIDR block uses a fixed prefix length of /56. You cannot choose the range of addresses or the IPv6 CIDR block size. AWS assigns the block to your VPC from the pool of IPv6 addresses from AWS.



Classless Inter-Domain Routing (CIDR) notation

CIDR Block:	Bit Representation:	Corresponding Address Range:
10.50.1.0/24	00001010.00110010.0000 0001.xxxxxxxx	10.50.1.0 –10.50.1.255
10.50.1.0/27	00001010.00110010.0000 0001.000xxxxx	10.50.1.0 – 10.50.1.31
10.50.1.132/32	00001010.00110010.0000 0001.10000100	10.50.1.132 (single address)
0.0.0.0/0	xxxxxxxx.xxxxxxxx.xxxxxxxx .xxxxxxxx	0.0.0.0 – 255.255.255.255 (all addresses)

The *Classless Inter-Domain Routing (CIDR)* format is used to specify IP address ranges when you create a VPC or a subnet. It specifies a block (known as a *CIDR block*) of IP addresses that use the format *x.x.x.x/n*, where:

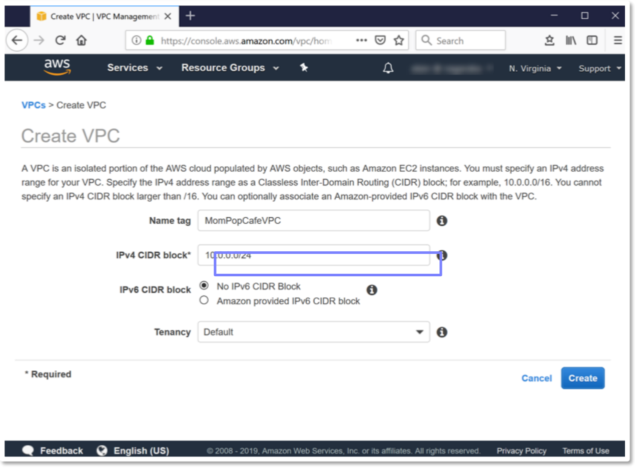
- *x.x.x.x* is an IP address. An IPv4 IP address is a 32-bit number that is represented as four numbers, which are separated by periods. Therefore, each *x* is an 8-bit number (a byte) that can have a value in the range 0 – 255. The IP address is logically divided into a *network prefix* and a *host identifier,* which identify the network and the host within the network, respectively.
- */n* specifies the *length* in bits of the network prefix portion of the IP address (starting from the leftmost bit). For an IPv4 IP address, the value of *n* can be in the range 0 – 32. In a VPC, the value of *n* is restricted to 16 – 28. In general, the larger the value of *n*, the smaller the range size becomes, which results in a smaller number of usable IP addresses.

The table provides examples of CIDR block ranges, their corresponding bit representation, and resulting address range.

Example AWS CLI command to create a VPC

Command

```
$ aws ec2 create-vpc --cidr-block 10.0.0.0/16
```



Create VPC | VPC Management
https://console.aws.amazon.com/vpc/hom
Search
aws
Services
Resource Groups
N. Virginia
Support

VPCs > Create VPC

Create VPC

A VPC is an isolated portion of the AWS cloud populated by AWS objects, such as Amazon EC2 instances. You must specify an IPv4 address range for your VPC. Specify the IPv4 address range as a Classless Inter-Domain Routing (CIDR) block; for example, 10.0.0.0/16. You cannot specify an IPv4 CIDR block larger than /16. You can optionally associate an Amazon-provided IPv6 CIDR block with the VPC.

Name tag: MomPopCafeVPC

IPv4 CIDR block*: 10.0.0.0/24

IPv6 CIDR block: ● No IPv6 CIDR Block ○ Amazon provided IPv6 CIDR block

Tenancy: Default

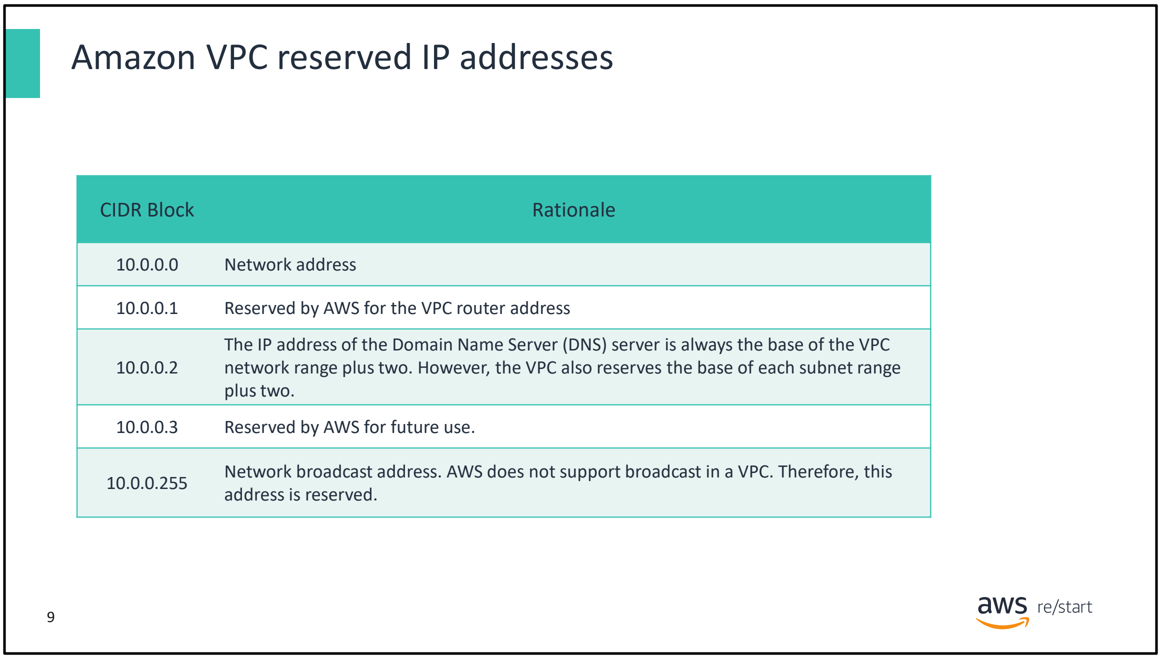
* Required

Cancel
Create
Feedback
English (US)
© 2008 - 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.
Privacy Policy
Terms of Use

The screen depicts the AWS Command Line Interface (AWS CLI) command that will generate the VPC that is shown in the console.

Expected result:

```
{
    "Vpc": {
        "CidrBlock": "10.0.0.0/16",
        "DhcpOptionsId": "dopt-32058557",
        "State": "pending",
        "VpcId": "vpc-012345678912",
        "InstanceTenancy": "default",
        "Ipv6CidrBlockAssociationSet": [],
        "CidrBlockAssociationSet": [
            {
                "AssociationId": "vpc-cidr-assoc-
04ae2a5af46e7bb7c",
                "CidrBlock": "10.0.0.0/16",
                "CidrBlockState": {
                    "State": "associated"
                }
            }
        ],
        "IsDefault": false,
        "Tags": []
    }
}
```

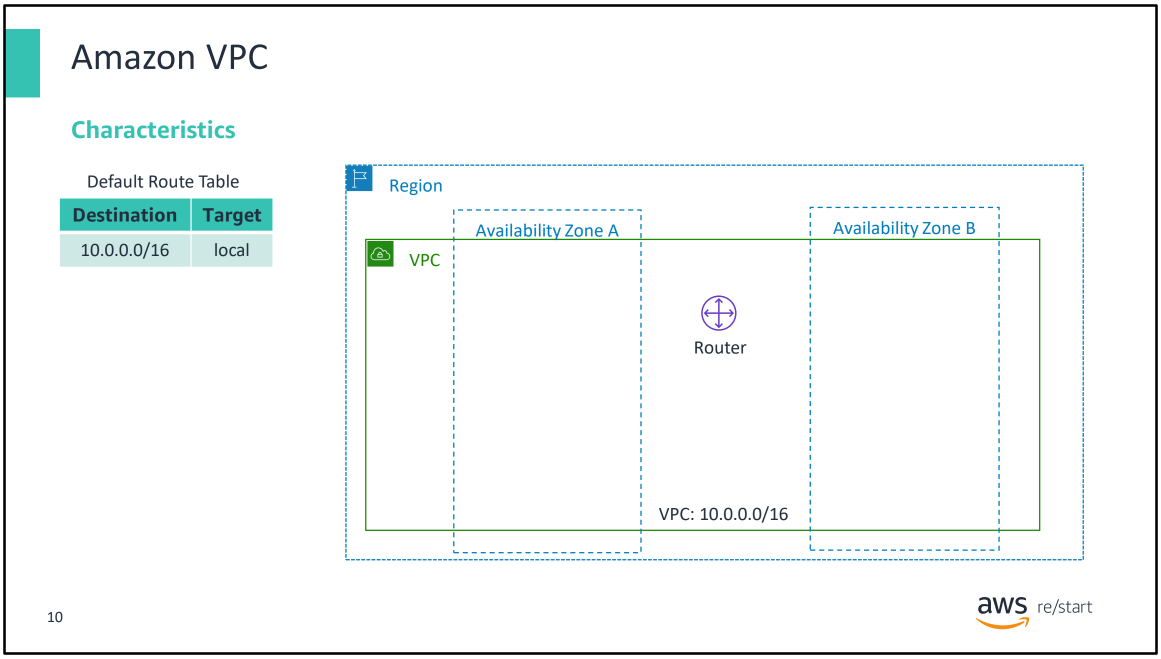


Amazon VPC reserved IP addresses

CIDR Block	Rationale
10.0.0.0	Network address
10.0.0.1	Reserved by AWS for the VPC router address
10.0.0.2	The IP address of the Domain Name Server (DNS) server is always the base of the VPC network range plus two. However, the VPC also reserves the base of each subnet range plus two.
10.0.0.3	Reserved by AWS for future use.
10.0.0.255	Network broadcast address. AWS does not support broadcast in a VPC. Therefore, this address is reserved.

The first four IP addresses and the last IP address in each subnet CIDR block are not available for you to use. They cannot be assigned to an instance. For example, in a subnet with CIDR block 10.0.0.0/24, the table depicts the five IP addresses that are reserved.

For VPCs with multiple CIDR blocks, the IP address of the Domain Name Server (DNS) server is in the primary CIDR.



Amazon VPC

Characteristics

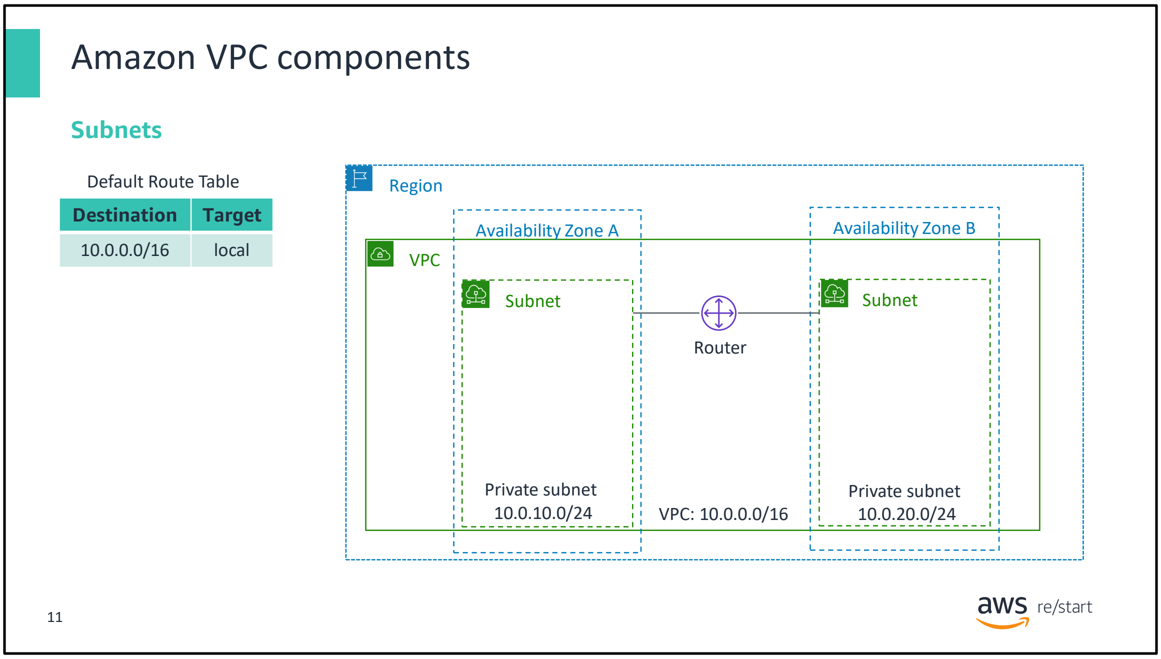
Default Route Table

Destination	Target
10.0.0.0/16	local

This diagram illustrates the main characteristics of a VPC:

- VPCs can span multiple Availability Zones in an AWS Region.
- VPCs have an implicit router that routes all traffic in the VPC.
- VPCs have a default route table that specifies the allowed routes out of the VPC. By default, this table defines a route (rule) that allows all traffic that is for its CIDR IP address range to be routed locally. In the example, the VPC has an address range of *10.0.0.0/16*. Therefore, its default route table has a rule to route all traffic that is destined for that range through the *local* route.

Next, you examine how subnets and route tables are used in a VPC.



Amazon VPC components

Subnets

Default Route Table

Destination	Target
10.0.0.0/16	local

Region

Availability Zone A

Availability Zone B

VPC

Subnet

Router

Subnet

Private subnet
10.0.10.0/24

VPC: 10.0.0.0/16

Private subnet
10.0.20.0/24

Subnets are used to further segment the VPC address range and to provide logical groupings for resources. For example, you can create a separate subnet for resources of different types. Examples of such types are Amazon Elastic Compute Cloud (Amazon EC2) instances and database instances, or resources with different visibility (public or private). Another example is to use subnets to divide resources by team or department.

You can create up to 200 subnets per VPC. For IPv4, the minimum size of a subnet is a /28 (or 16 IP addresses). For IPv6, the subnet size is fixed to be a /64, and only one IPv6 CIDR block can be allocated to a subnet.

Examples of private subnet ranges to use with Amazon VPC are:

- 10.0.0.0 – 10.255.255.255 (10/8 prefix)
- 172.16.0.0 – 172.31.255.255 (172.16/12 prefix)

- 192.168.0.0 – 192.168.255.255 (192.168/16 prefix)

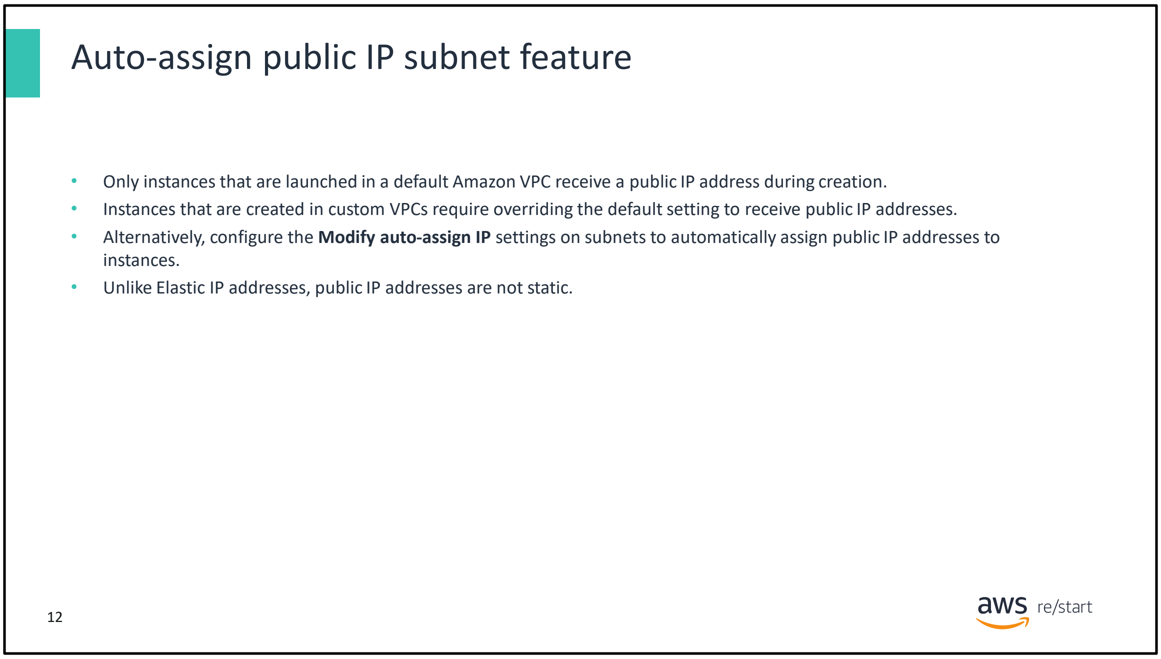
This diagram illustrates the following characteristics of subnets:

- Subnets can exist in one—and only one—Availability Zone.
- In a subnet, the address range of its CIDR block must be a subset of its VPC's address range. In the example, the CIDR block for the subnet in Availability Zone A supports IP addresses 10.0.10.0 – 10.0.10.255. The CIDR block in Availability Zone B supports IP addresses 10.0.20.0 – 10.0.20.255. Both of these ranges are subsets of the VPC address range, which supports IP addresses 10.0.0.0 – 10.0.255.255.
- Subnet CIDR blocks within a VPC must not overlap. This rule is true in the diagram for the two subnets (10.0.10.0/24 vs. 10.0.20.0/24).
- Traffic to and from each subnet flows through the implicit router of the VPC.

NOTE: Remember to only use a /16 or smaller from the private ranges that are listed. For example, you could use any of the following ranges:

- 10.0.0.0 – 10.0.255.255 (10.0.0.0/16) – 65534 usable hosts
- 10.1.0.0 – 10.1.255.255 (10.1.0.0/16) – 65534 usable hosts
- 172.16.0.0 – 172.16.255.255 (172.16.0.0/16) – 65534 usable hosts
- 172.17.0.0 – 172.17.255.255 (172.17.0.0/16) – 65534 usable hosts
- 192.168.0.0 – 192.168.127.255 (192.168.0.0/17) – 32766 usable hosts
- 192.168.128.0 – 192.168.255.255 (192.168.128.0/17) – 32766 usable hosts

For more information, refer to VPCs and subnets.



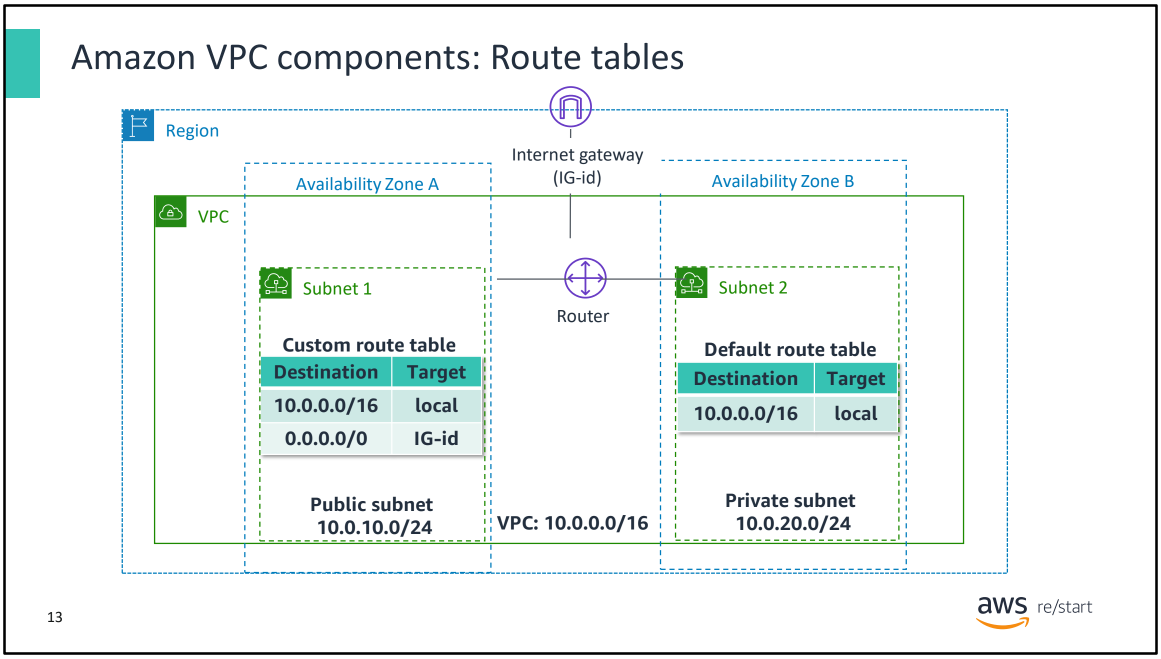
Auto-assign public IP subnet feature

- Only instances that are launched in a default Amazon VPC receive a public IP address during creation.
- Instances that are created in custom VPCs require overriding the default setting to receive public IP addresses.
- Alternatively, configure the **Modify auto-assign IP** settings on subnets to automatically assign public IP addresses to instances.
- Unlike Elastic IP addresses, public IP addresses are not static.

After you create a subnet, you can specify whether new EC2 instances that are launched into it should automatically have an assigned public IP address. This address is useful for a *public* subnet that is intended to host EC2 instances that are accessible to internet clients. Before an instance can be publicly accessed, it must be assigned a public IP address. By enabling this subnet feature, you can automate the assignment of an IPv4 or IPv6 address to instances that are launched into the subnet.

To enable this setting in the VPC console, go to the **Modify auto-assign IP settings** configuration page for the subnet. Select the **Enable auto-assign public IPv4/IPv6 address** check box.

Note: The assigned IP address is not an Elastic IP address, and it is released from the instance when the instance is stopped or terminated. An Elastic IP address is a static IPv4 address that is reachable from the internet.



Amazon VPC components: Route tables

Custom route table

Destination	Target
10.0.0.0/16	local
0.0.0.0/0	IG-id

Public subnet
10.0.10.0/24

Default route table

Destination	Target
10.0.0.0/16	local

Private subnet
10.0.20.0/24

Each VPC comes with an implicit router that directs traffic between resources in each subnet and out of the subnet. The router uses a *route table* to determine the valid IP destinations out of a subnet and the corresponding target to reach the destination.

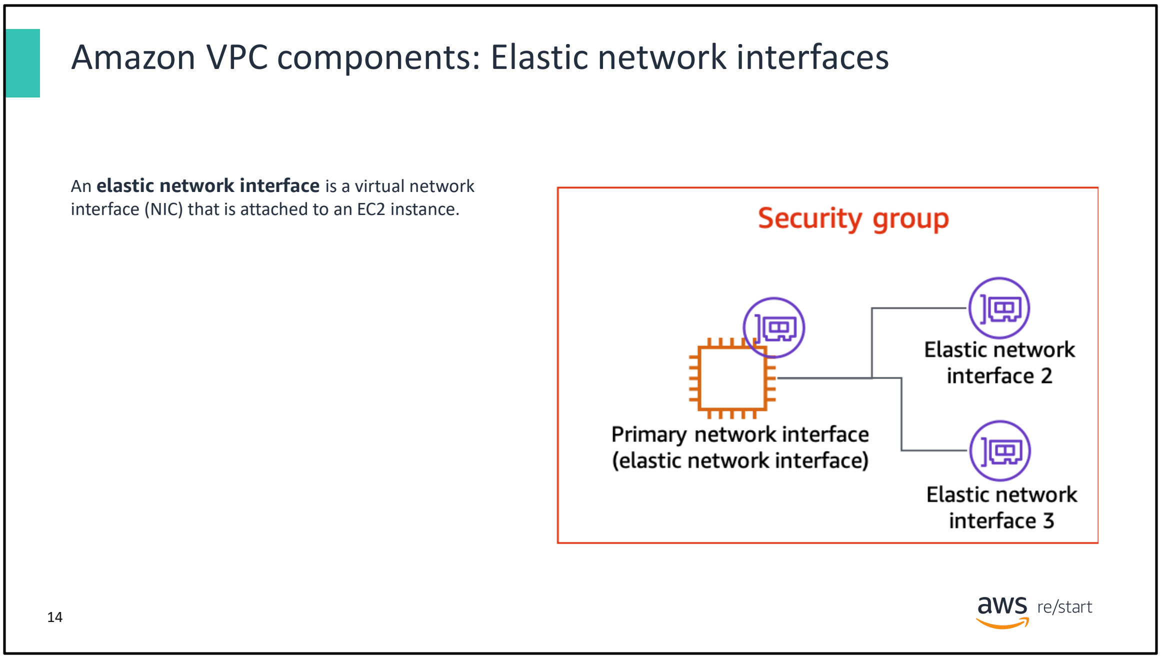
A route table is a mechanism used for routing traffic that originates from an associated subnet in a VPC. It contains a set of rules (also called *routes*) that determine where traffic is sent. Routes in a route table consist of a *destination* and a *target*. The router reads the route like this: “Any traffic that goes to *destination* should be routed through *target.”* A target can be a specific instance ID, an elastic network interface ID, an internet gateway, or a virtual private gateway.

When a VPC is created, a *default* route table is also created. This default route table has a rule that routes local traffic anywhere within the IP address range of the VPC. You can add more routes to the default route table. When you create a subnet, it is automatically associated with the default route table of the VPC. If you do not want to use this route table, you can create a *custom* route table and associate it with the subnet instead.

Each VPC might also have an internet gateway that is attached to it. An internet gateway is a service that routes traffic to the internet. When an internet gateway is attached to a VPC, route tables of public subnets might use it to forward traffic out to the internet.

In the diagram:

- Subnet 2 is associated with the default route table. A single route is defined in the table. It states that traffic is allowed out of the subnet only to the IP addresses in the VPC’s address range (*10.0.0.0/16*). This traffic is routed through a local path.
- Subnet 1 is associated with a custom route table that has two routes. The first route is the same as in the default route table. Thus, local traffic is allowed out of the subnet to the IP addresses in the VPC’s address range. In addition, the second route (*0.0.0.0/0*) states that traffic to any other IP addresses is also allowed. It should be routed through the internet gateway that is identified by *IG-id*.



Amazon VPC components: Elastic network interfaces

An **elastic network interface** is a virtual network interface (NIC) that is attached to an EC2 instance.

Security group
Primary network interface (elastic network interface)
Elastic network interface 2
Elastic network interface 3

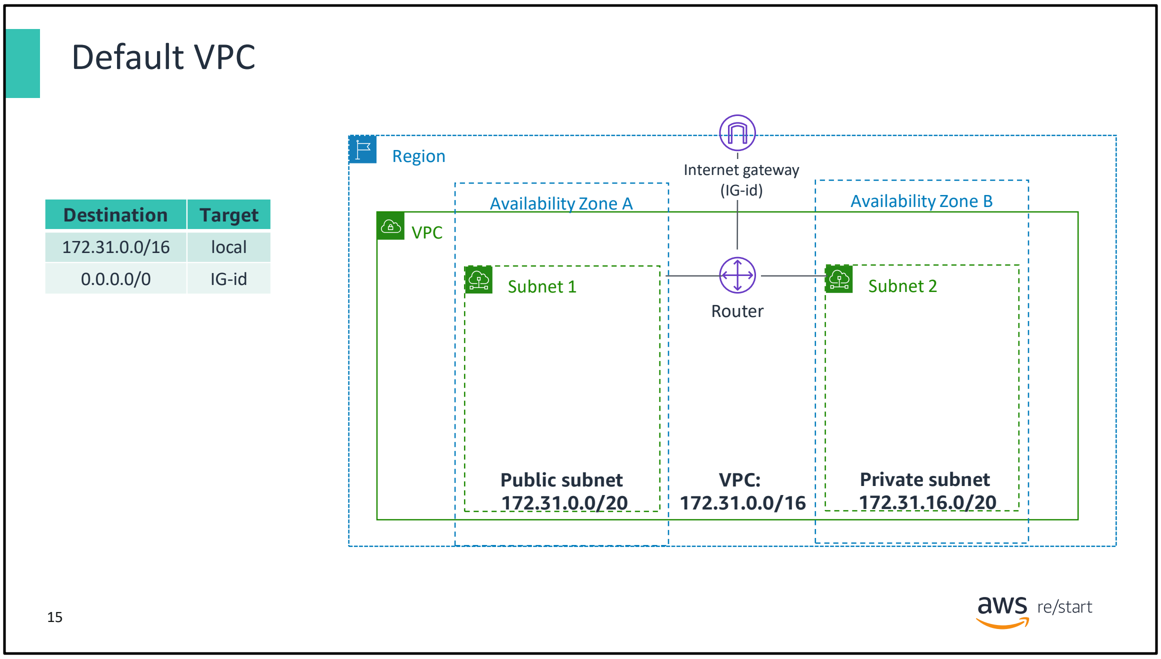
An *elastic network interface*, which is also called a *network interface (NIC),* is a virtual network interface (NIC) that is attached to an EC2 instance. It provides the connection point that enables an instance to communicate with a network. Each network interface has one primary IP address, plus additional secondary IP addresses. It also has its own media access control (MAC) address and security groups.

Each instance in a VPC has a default NIC, the *primary network interface*, which is assigned a private IPv4 address from the IPv4 address range of the VPC. You cannot detach a primary network interface from an instance.

You can create and attach additional NICs to an instance. The number of NICs that you can attach varies by instance type. An additional NIC can be attached to an instance, detached it from that instance, and attached to another instance. The NIC's attributes follow it when it is attached or detached from instances. When you move a NIC from one instance to another instance, network traffic is redirected to the new instance.

Use cases for multiple NICs on an instance include:

- *Using network and security appliances in a VPC* – Some network and security appliances—such as load balancers, network address translation (NAT) servers, and proxy servers—prefer to be configured with multiple NICs. You can create and attach secondary NICs to instances in a VPC that run these types of applications. Then, configure the additional interfaces with their own public and private IP addresses, and security groups.
- *Creating a management only network interface* – To ensure that bandwidth on a customer facing interface is not impacted by management activities (uploading new versions of software, downloading log files, and so on) a separate ENI is used for administrative work.



Default VPC

Destination	Target
172.31.0.0/16	local
0.0.0.0/0	IG-id

Region

Internet gateway
(IG-id)

Availability Zone A

Availability Zone B

VPC

Subnet 1

Router

Subnet 2

Public subnet
172.31.0.0/20

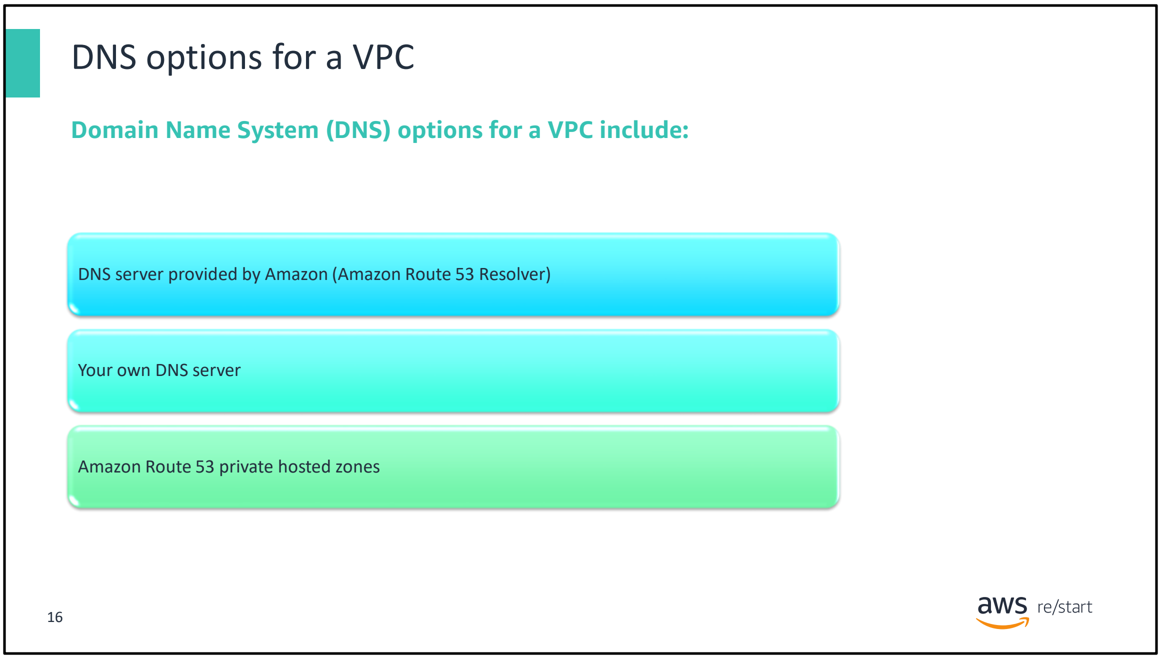
VPC:
172.31.0.0/16

Private subnet
172.31.16.0/20

When you create an AWS account, AWS automatically creates a *default VPC* for you with a CIDR block of *172.31.0.0/16*. This VPC provides up to 65,536 private IPv4 addresses in the range 172.31.0.0–172.31.255.255. This default VPC enables you to start using VPC features immediately.

AWS also automatically creates the following components in the default VPC:

- An internet gateway to enable communication with the internet.
- A default route table with rules to send traffic to IP addresses in the VPC's address range to a local route. It also sends traffic to any other IP address to the internet gateway.
- A public subnet in each Availability Zone with a /20 size, which provides up to 4,096 addresses. The **Auto-Assign Public IP** option is enabled for these subnets. Any instance that is launched in the default VPC automatically gets a public IP address. They are *public* subnets because they are associated with the default route table, which has a rule that allows traffic through an internet gateway.



DNS options for a VPC

Domain Name System (DNS) options for a VPC include:

- DNS server provided by Amazon (Amazon Route 53 Resolver)
- Your own DNS server
- Amazon Route 53 private hosted zones

A Domain Name System (DNS) server is used to resolve a DNS hostname (such as *www.example.com*) to its corresponding IP address (such as *192.0.2.1)*. When you create a VPC, AWS automatically assigns a DNS server (provided by Amazon) to it for resolving hostnames in the VPC. As of December 2018, this DNS server is now called *Amazon Route 53 Resolver*. By default, Resolver directly answers DNS queries for domain names within the VPC. It performs recursive lookups against public name servers for all other domain names.

If you want to use a different DNS server for a VPC, the available options are:

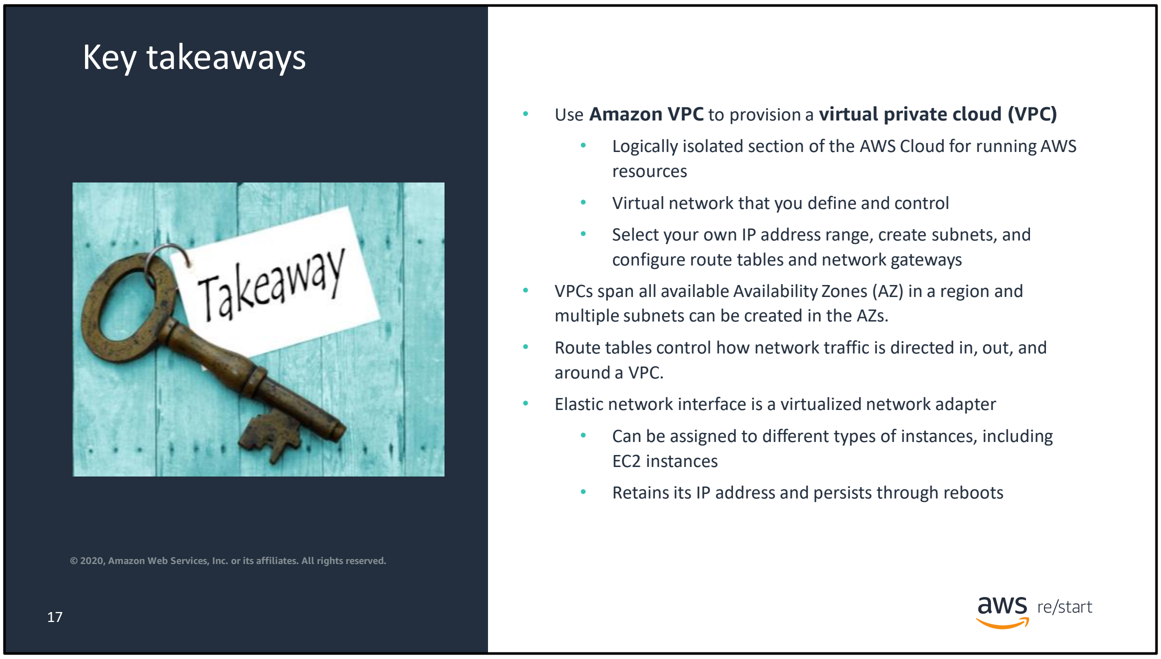
- Use *your own DNS server*. This option requires specifying a special set of Dynamic Host Configuration Protocol (DHCP) options for the VPC.
- Use an *Amazon Route 53 private hosted zone*. A *hosted zone* is a container that holds information about how Amazon Route 53 routes traffic for the domains in a VPC. A *public* hosted zone allows Amazon Route 53 to route internet traffic to resources inside a VPC. For example, a customer can view the company website, which is running on an EC2 instance. Amazon Route 53 uses a *private* hosted zone to route

traffic within one or more VPCs without exposing resources to the internet.

One common use case for creating a private hosted zone DNS is called a *split-horizon DNS*. A split-horizon DNS pairs a private hosted zone DNS with a public hosted zone DNS. With this implementation, a given DNS hostname resolves differently depending on whether the lookup comes from inside or outside the VPC. If a lookup is initiated from within the VPC, the DNS hostname resolves to a specified IP address. However, if the lookup originates from outside the VPC, the same DNS hostname resolves to a different IP address. An example scenario for using a split-horizon DNS is when you want to maintain an internal and external version of the same website. Split-horizon DNS enables you to access the internal version of the website by using the same domain name that is used for the public website.

For more information about VPC, DNS, and Route 53 private hosted zones, refer to:

- Using DNS with your VPC
- Working with private hosted zones



Key takeaways

- Use **Amazon VPC** to provision a **virtual private cloud (VPC)**
  - Logically isolated section of the AWS Cloud for running AWS resources
  - Virtual network that you define and control
  - Select your own IP address range, create subnets, and configure route tables and network gateways
- VPCs span all available Availability Zones (AZ) in a region and multiple subnets can be created in the AZs.
- Route tables control how network traffic is directed in, out, and around a VPC.
- Elastic network interface is a virtualized network adapter
  - Can be assigned to different types of instances, including EC2 instances
  - Retains its IP address and persists through reboots

Some key takeaways for this module are:

- Use **Amazon VPC** to provision a **virtual private cloud (VPC)**
  - Logically isolated section of the AWS Cloud for running AWS resources
  - Virtual network that you define and control
  - Select your own IP address range, create subnets, and configure route tables and network gateways
- VPCs can span multiple Availability Zones, and VPCs can also have multiple subnets.
- Route tables control how network traffic is directed in, out, and around a VPC.
- Elastic network interface is a virtualized network adapter
  - Can be assigned to different types of instances, including EC2 instances

- Retains its IP address and persists through reboots