aws training and certification

Introduction to Amazon Virtual Private Cloud (VPC)

SPL-84 - Version 2.0.6

Overview

This lab introduces you to Amazon Virtual Private Cloud (Amazon VPC). In this lab you will use the Amazon VPC wizard to create a VPC, attach an Internet Gateway, add a subnet and then define routing for the VPC so that traffic can flow between the subnet and the Internet gateway.

Topics covered

Upon completion of this lab, you will be able to:

- Create an Amazon VPC Using the **VPC Wizard**
- Explore the basic components of a VPC including:
- Public and private subnets
- Route tables and routes
- NAT Gateways
- Network ACLs
- Elastic IPs

What is Amazon Virtual Private Cloud (VPC)?

Amazon Virtual Private Cloud (Amazon VPC) lets you provision a logically isolated section of the Amazon Web Services (AWS) cloud where you can launch AWS resources in a virtual network that you define. You have complete control over your virtual networking environment, including selection of your own IP address range, creation of subnets, and configuration of route tables and network gateways. You can use both IPv4 and IPv6 in your VPC for secure and easy access to resources and applications.

Task 1: Create an Elastic IP address

Your VPC will launch a NAT Gateway to provide Internet access to private resources. The NAT Gateway will be assigned a static IP address, known as an **Elastic IP address**. In this task, you will create the Elastic IP address.

An Elastic IP address is a public IPv4 address, which is reachable from the Internet. It is a *static IP address*, which means that the IP address will not change. You can associate the Elastic IP address with a resource in your VPC, such as a NAT Gateway or an Amazon EC2 instance. You retain control of the Elastic IP address until you release it back to AWS.

- In the **AWS Management Console**, on the **Services** menu, click **VPC**.
- In the left navigation pane, click **Elastic IPs**.
- Click **Allocate new address**.
- Click **Allocate**.

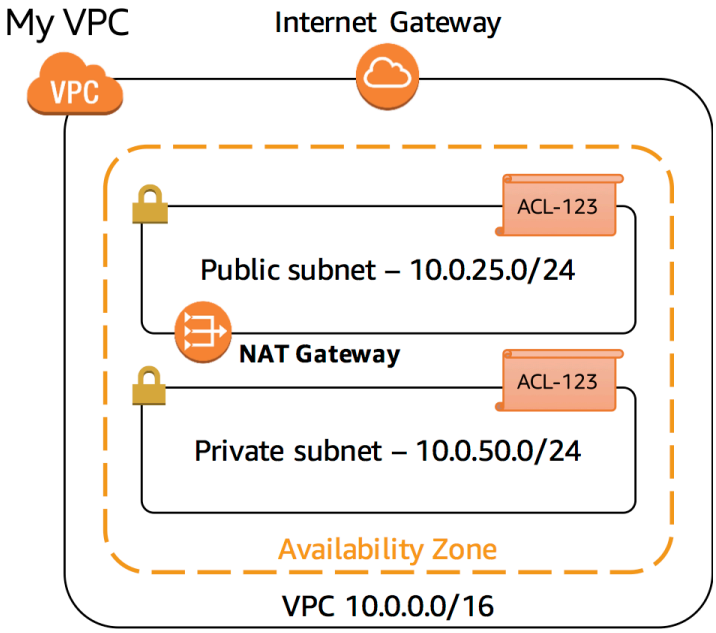
Your Elastic IP address is displayed. You will use it in the next task.

- Click **Close**.

Task 2: Create an Amazon VPC

In this task you will create an Amazon VPC using the **VPC wizard**. The wizard automatically creates a VPC based upon parameters you specify. Using the VPC Wizard is much simpler than manually creating each component of the VPC.

Here is an overview of the VPC you will create:



Each component will be explained in more detail later in this lab.

- Click **VPC Dashboard** in the top-left corner.
- Click **Launch VPC Wizard**.

The wizard offers four pre-defined configurations. Click each option in the Wizard to view their definition:

- **VPC with a Single Public Subnet:** A single public subnet connected to the Internet. This is ideal for applications that operate purely in the AWS cloud.
- **VPC with Public and Private Subnets:** A public subnet for Internet-facing resources and a private subnet for back-end resources. A NAT Gateway is also launched to provide Internet access for resources in the private subnet. This is ideal for keeping private resources separate from the Internet.
- **VPC with Public and Private Subnets and Hardware VPN Access:** A public subnet and a private subnet, plus a Virtual Private Network (VPN) connection to an existing Corporate Data Center. This is ideal when you have legacy infrastructure in a data center, which can connect to the AWS cloud as a combined network.
- **VPC with a Private Subnet Only and Hardware VPN Access:** A private subnet connected to a Corporate Data Center via a VPN connection. This is ideal for *bursting into the AWS cloud* to provide additional resources while remaining totally secure from Internet access. This design is often used for Development and Testing, where no direct Internet access is required.

This lab will use the **VPC with Public and Private Subnets** template.

- Click **VPC with Public and Private Subnets** (the second option).
- Click **Select**.

You are now presented with parameters to customize the VPC configuration. Configure the following settings, leaving other fields at their default values:

- **VPC name:**
- **Public subnet's IPv4 CIDR:**
- **Public Availability Zone:** Select the first Availability Zone in the list
- **Private subnet's IPv4 CIDR:**
- **Private Availability Zone:** Select the same availability zone as the public subnet
- **Elastic IP Allocation ID:** Click in the box and select the Elastic IP Address you created earlier
- Click **Create VPC**.

Your VPC will now be created. A status window displays progress. When the VPC completes, a status window confirms that your VPC has been successfully created. This may take a few minutes to create.

- Click **OK** to close the status window and return to the VPC dashboard.

Your newly created VPC is now displayed in the **VPC Dashboard**.

Task 3: Explore your VPC

In this task, you will explore the VPC components created by the VPC Wizard.

- In the top-left corner, under **Filter by VPC**, click in the **Select a VPC** field and select **My VPC**.

This limits the console display to only show components related to the VPC you created.

- In the left navigation pane, click **Internet Gateways**.

The Internet gateway for your VPC will be displayed.

An Internet gateway connects your VPC to the Internet. If the Internet Gateway was not present, then the VPC would have *no* connectivity to the Internet.
An Internet gateway is a horizontally scaled, redundant and highly available VPC component. It therefore imposes no availability risks or bandwidth constraints on your network traffic.

- In the left navigation pane, click **Subnets**.

A Subnet is a subset of a VPC. A subnet:

  - Belongs to a specific **VPC**
  - Exists in a single **Availability Zone** (while a VPC can span multiple Availability Zones)
  - Has a **range of IP addresses** (known as a CIDR range, which stands for Classless Inter-Domain Routing)

Two subnets will be displayed for your VPC: a Public subnet and a Private subnet.

- Select the **Public subnet**.

Examine the information displayed in the lower window pane:

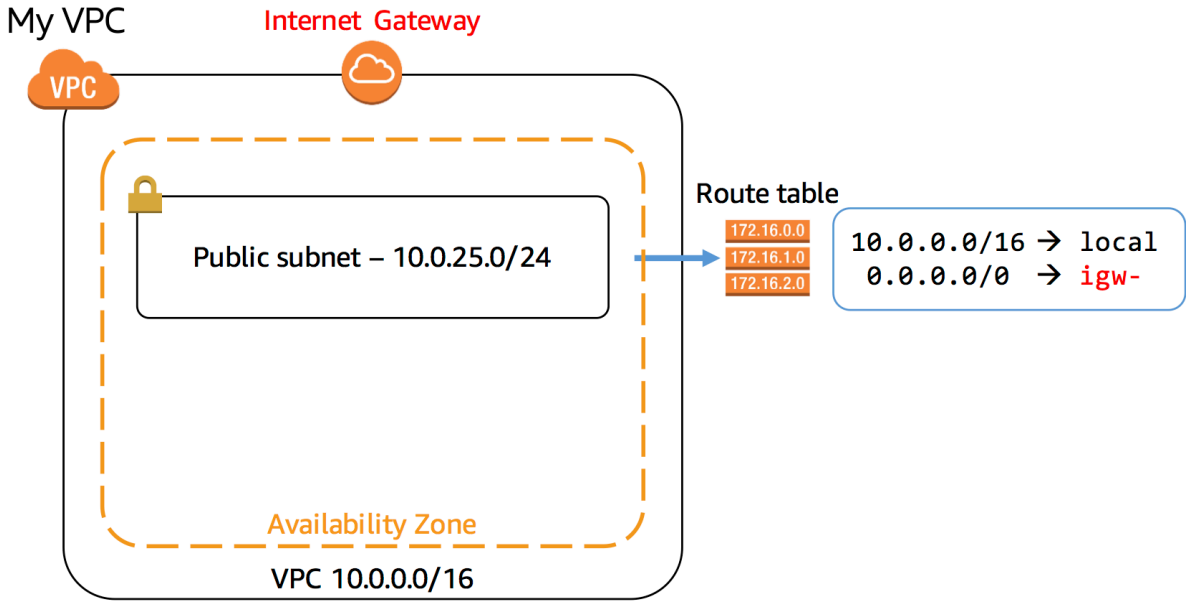
- Each subnet is assigned a unique **Subnet ID**.
- The **IPv4 CIDR** of *10.0.25.0/24* means that the subnet contains the range of IP addresses from *10.0.25.0* to *10.0.25.255*. (IPv6 is also supported, but is not part of this lab.)

- The subnet only has 250 **Available IPs** out of 256 possible addresses. This is because there are several reserved addresses in each subnet and one IP address has been consumed by the NAT Gateway.

Why is this subnet considered to be a *Public* subnet? The answer lies in the Subnet *Routing*.

- Click the **Route Table** tab.

  Each subnet is associated with a **Route Table**, which specifies the routes for outbound traffic leaving the subnet. Think of it like an address book that lists where to direct traffic based upon its destination.



There are two routes in the route table that is associated with your public subnet:

- **Route 10.0.0.0/16 l local** directs traffic destined for elsewhere in the VPC (which has a range of *10.0.0.0/16*) locally within the VPC. This traffic never leaves the VPC.
- **Route 0.0.0.0/0 l igw-** directs all traffic to the Internet gateway.

Routing rules are evaluated from the most restrictive (with the bigger number after the slash) through to the least restrictive (which is *0.0.0.0/0* since it refers to the entire Internet). Thus, traffic is first sent within the VPC if it falls within the range of the VPC, otherwise it is send to the Internet. The rules can further be edited based upon your particular network configuration.

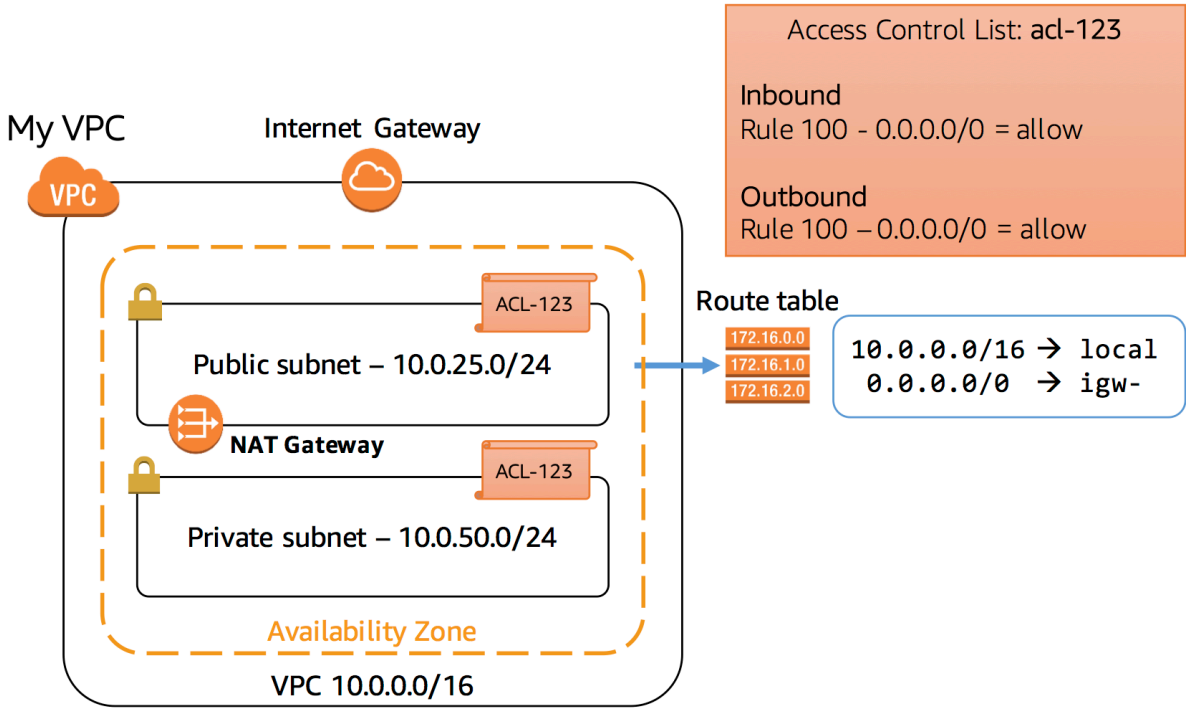
The fact that this subnet is *associated with a Route Table that has a route to an Internet gateway* makes it a ***Public Subnet***. That is, it is *reachable from the Internet*.

- Click the **Network ACL** tab.

A network access control list (ACL) is an optional layer of security for your VPC that acts as a firewall for controlling traffic in and out of subnets. Network

ACLs are normally left with their default settings that allow all traffic in and out of subnets:

- **Rule 100 Inbound** allows all inbound traffic into the **Public Subnet**.
- **Rule 100 Outbound** allows all traffic out of the **Public Subnet**.
- The second line in each ruleset shows an asterisk (*) that acts as a *catch-all* rule in case traffic does not match any of the earlier rules.

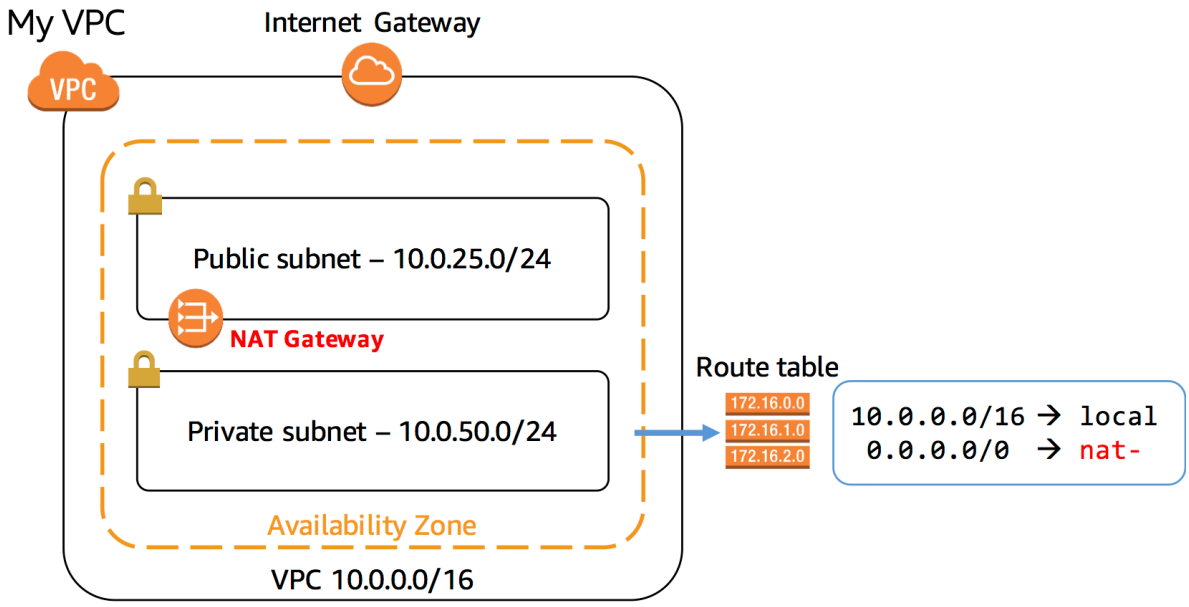


- Click the **Tags** tab.
  The subnet has been tagged with the key of **Name** having the value of **Public subnet**. Tags help you to manage and identify your AWS resources.
- At the top of the window, select **Private subnet** and ensure that it is the only line selected.
- Click the **Route Table** tab.

The Route Table for the Private subnet has the configuration:

- **Route 10.0.0.0/16 l local** is the same as the Public subnet.
- **Route 0.0.0.0 l nat-** directs traffic to the NAT Gateway.

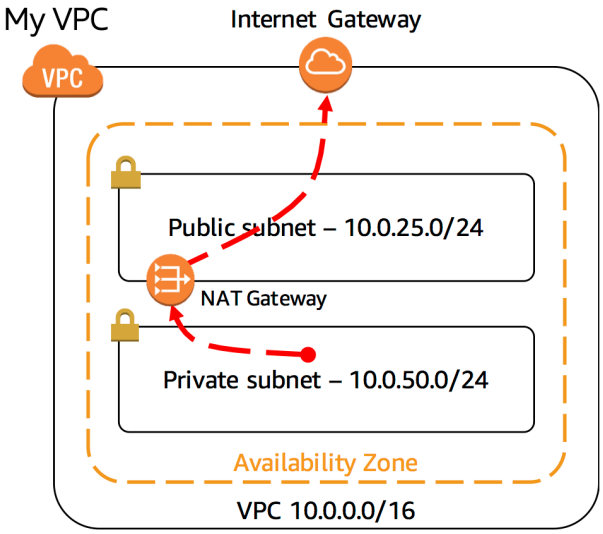
This subnet does not have a route to the Internet gateway. Therefore, it is a ***Private Subnet***.



23. In the left navigation pane, click **NAT Gateways**.

A NAT gateway is displayed.

A Network Address Translation (NAT) Gateway allows resources in a private subnet to connect to the Internet and other resources outside the VPC. This is an *outbound-only* connection, which means that the connection must be initiated from within the private subnet. Resources on the Internet cannot initiate an inbound connection. Therefore, it is a means of keeping resources private and improving security for VPC resources.



- In the left navigation pane, click **Security Groups**.
- Select the Security Group displayed and click the **Inbound Rules** tab.

Security groups act as virtual firewall for your instances to control inbound and outbound traffic. When you launch an Amazon EC2 instance into a VPC, you can assign up to five security groups to the instance. Security groups act at the instance level and not the subnet level. Your VPC automatically comes with a default security group. If you do not specify a different security group when you launch an Amazon EC2 instance, it will use the default security group.

The default security group permits *ALL traffic* to access associated resources, but only if the *Source* is the default security group. This self-reference might appear strange, but this configuration simply means that any EC2 instance associated with the default security group can communicate with any other EC2 instance that is associated with the default security group. All other traffic is denied. This is a very safe default setting because it limits any access from other resources.
When adding resources to the VPC, you can create additional security groups to permit desired access to resources such as web servers, application servers and database servers.

Launching Amazon EC2 instances in this lab is out of the scope of the lab. Please do not attempt to launch an Amazon EC2 instance. This lab will not allow you to launch EC2 instances.

End Lab