Introduction to Amazon Virtual Private Cloud (VPC)



**SPL-84 - Version 2.0.15**

© 2023 Amazon Web Services, Inc. or its affiliates. All rights reserved. This work may not be reproduced or redistributed, in whole or in part, without prior written permission from Amazon Web Services, Inc. Commercial copying, lending, or selling is prohibited. All trademarks are the property of their owners.

Note: Do not include any personal, identifying, or confidential information into the lab environment. Information entered may be visible to others.

Corrections, feedback, or other questions? Contact us at [*AWS Training and Certification*](https://support.aws.amazon.com/#/contacts/aws-training).

**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

**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.

**Start lab**

1. To launch the lab, at the top of the page, choose **Start lab**.

 You must wait for the provisioned AWS services to be ready before you can continue.

1. To open the lab, choose **Open Console**.

You are automatically signed in to the AWS Management Console in a new web browser tab.

**Do not change the Region unless instructed.**

COMMON SIGN-IN ERRORS

**Error: You must first sign out**



If you see the message, **You must first log out before logging into a different AWS account:**

* Choose the **click here** link.
* Close your **Amazon Web Services Sign In** web browser tab and return to your initial lab page.
* Choose **Open Console** again.

**Error: Choosing Start Lab has no effect**

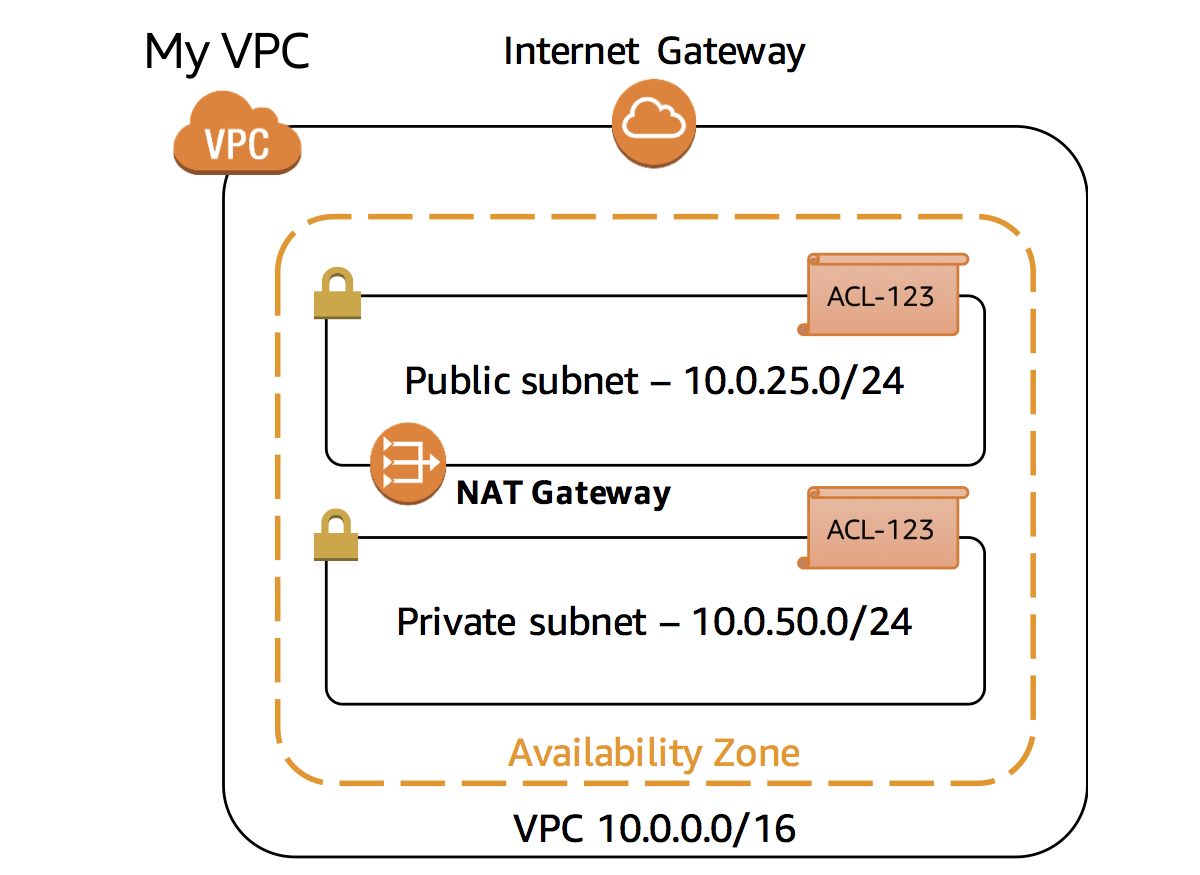
In some cases, certain pop-up or script blocker web browser extensions might prevent the **Start Lab** button from working as intended. If you experience an issue starting the lab:

* Add the lab domain name to your pop-up or script blocker’s allow list or turn it off.
* Refresh the page and try again.

**Task 1: 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.

1. At the top of the AWS Management Console, to the right of **Services** menu, in the search bar, search for

**VPC**

 and then choose **VPC** from the list.

1. Click **VPC Dashboard** in the top-left corner.
2. Click **Create VPC**
3. On **Create VPC** page, under **VPC settings** section, choose **VPC and more** (the second option).
4. Configure the following fields:

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

* Under **Name tag auto-generation:** Make sure  **Auto-generate** is selected and provide

Lab

 in the text box.

* **Number of Availability Zones (AZs):** Choose **1**.
* **Number of public subnets:** Choose **1**.
* **Number of private subnets:** Choose **1**.
* **Expand Customize subnets CIDR blocks:**
* **Customize public subnets CIDR blocks:**

10.0.25.0/24

* **Customize private subnets CIDR blocks:**

10.0.50.0/24

* **NAT gateways ($):** Choose **In 1 AZ**
* **VPC endpoints:** **None**

1. 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.

1. Choose **View VPC**.
2. Copy the **VPC ID** value and paste it into your text editor.

**Task 2: Explore your VPC**

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

1. Select **Your VPCs** from left panel.
2. Search for the **VPC ID**, saved in previous step.
3. Locate Your VPCs’s **Name** column, your VPC is created with the name **Lab-vpc**.
4. If **Lab-vpc** is not displayed under **Filter by VPC**, Refresh your screen, and then choose  **Select a VPC** again.
5. In the left navigation pane, choose **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.

1. In the left navigation pane, choose **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](https://en.wikipedia.org/wiki/Classless_Inter-Domain_Routing))

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

1. Select  the **Public subnet** which starts with **Lab-subnet-public** in the **Name** column.

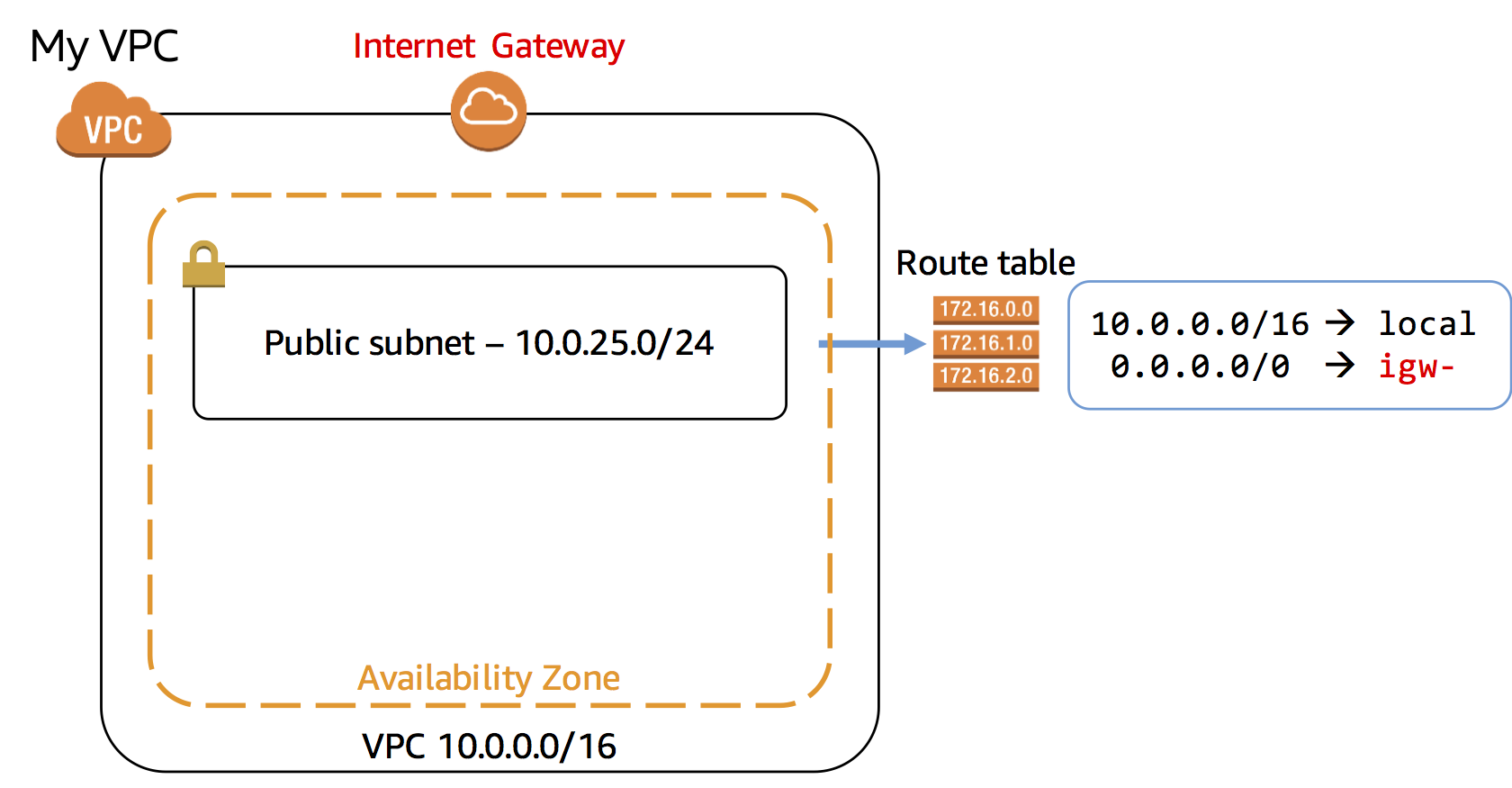
Examine the information displayed in the lower window pane:

* 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*.

1. Choose 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 | 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 | igw-** directs all traffic to the Internet gateway.

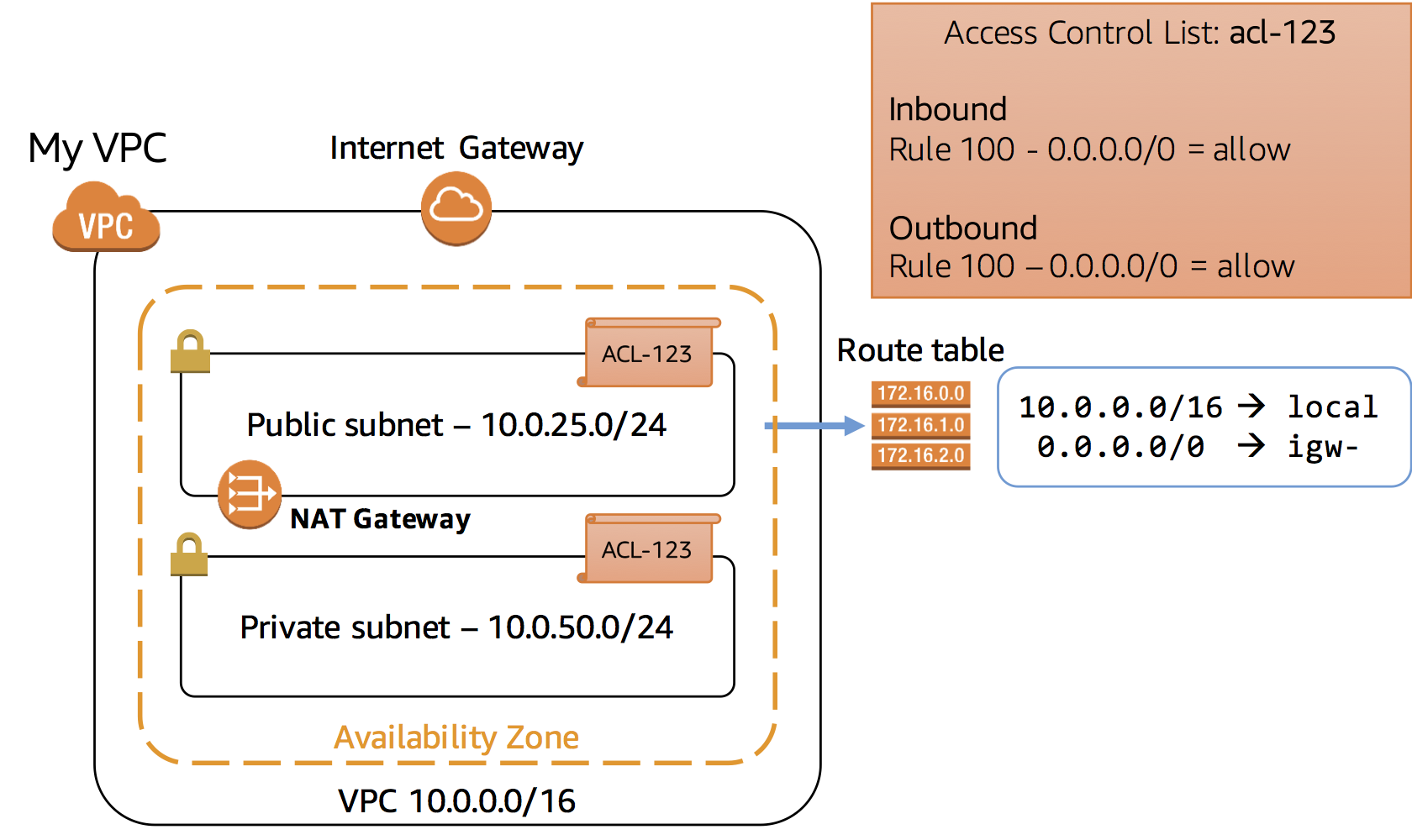
Routing rules are evaluated from the most restrictive (with the bigger number after the slash) through to the least restrictive (which is *0.0.0.0/0* since it refers to the entire Internet). Thus, traffic is first sent within the VPC if it falls within the range of the VPC, otherwise it is send to the 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*.

1. Choose 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.



1. In the left navigation pane, choose **Subnets**.
2. At the top of the window, select  **Private subnet** which starts with **Lab-subnet-private** in the **Name** column and ensure that it is the only line selected.
3. Choose the **Tags** tab.

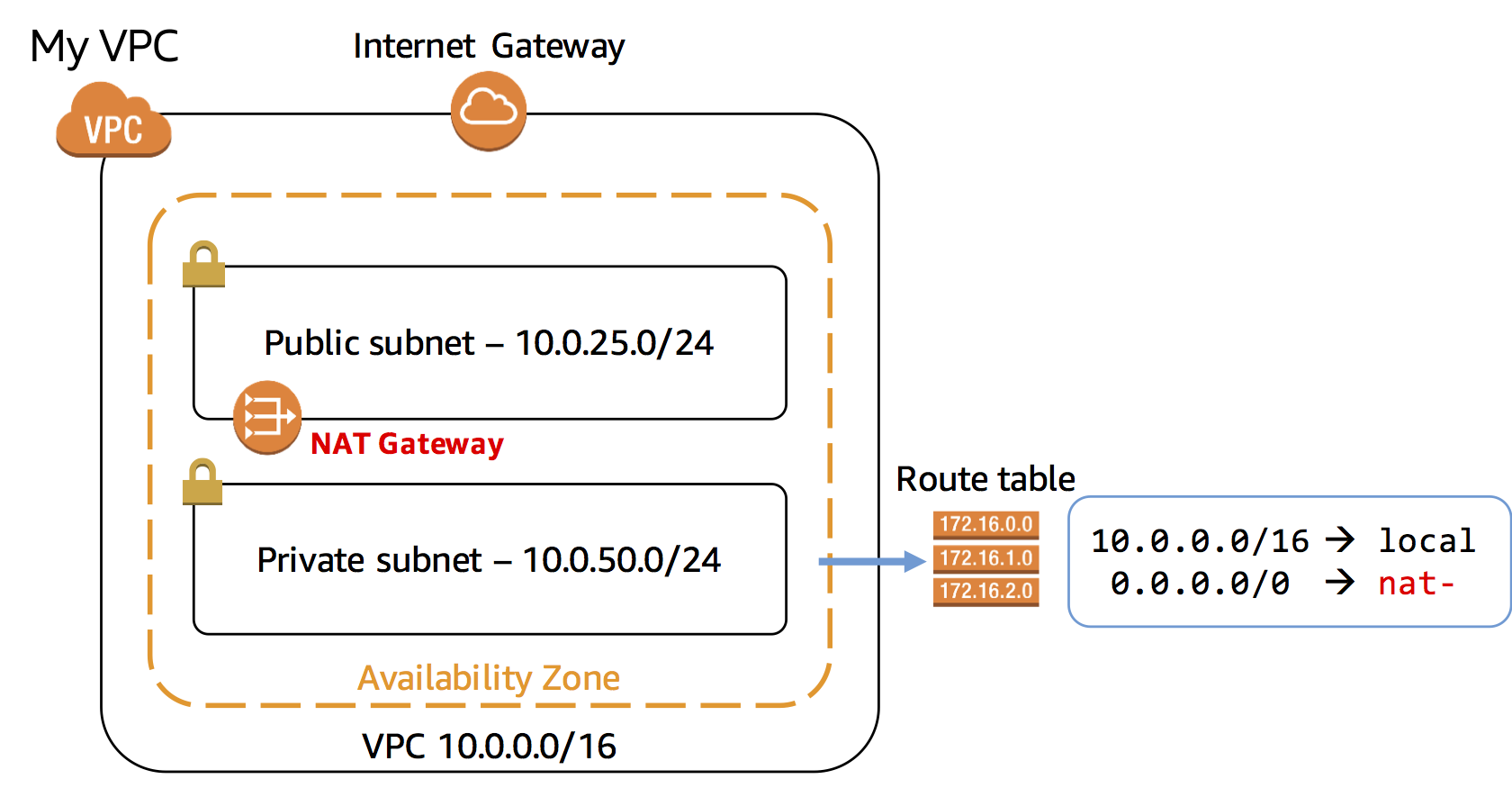
The subnet has been tagged with the key of **Name** starting with the value of **Lab-subnet-private**. Tags help you to manage and identify your AWS resources.

1. Choose the **Route table** tab.

The Route Table for the Private subnet has the configuration:

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

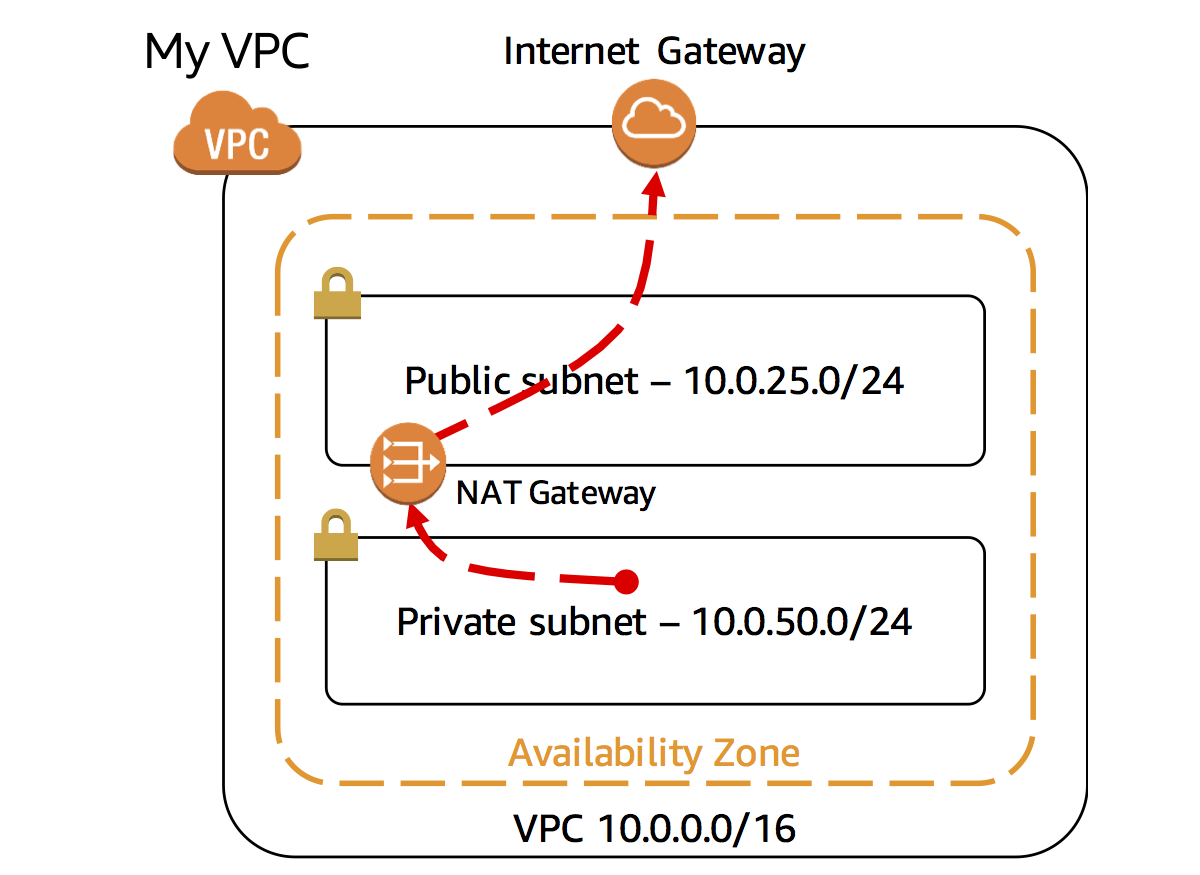
This subnet does not have a route to the Internet gateway. Therefore, it is a ***Private Subnet***.



1. In the left navigation pane, choose **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.



1. In the left navigation pane, choose **Security groups**.
2. Select  the Security group that matches with the **VPC ID** that you copied to your text editor and choose 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.

**Conclusion**

Congratulations! You now have successfully:

* Created an Amazon VPC Using the **VPC Wizard**.
* Explored the basic components of a VPC.

**End lab**

Follow these steps to close the console and end your lab.

1. Return to the **AWS Management Console**.
2. At the upper-right corner of the page, choose **AWSLabsUser**, and then choose **Sign out**.
3. Choose **End lab** and then confirm that you want to end your lab.

**Additional resources**

* For more information about Amazon VPC, see: http://aws.amazon.com/vpc/
* For more information on AWS Training & Certification, see: http://aws.amazon.com/training/

For more information about AWS Training and Certification, see [*https://aws.amazon.com/training/*](https://aws.amazon.com/training/).

*Your feedback is welcome and appreciated.*  
If you would like to share any feedback, suggestions, or corrections, please provide the details in our [*AWS Training and Certification Contact Form*](https://support.aws.amazon.com/#/contacts/aws-training).