**Cloud Computing Lab**

**Experiment No.: 3**

[Create a Virtual Private Clouds and establish connections between each other. (Amazon VPC)](https://awseducate.instructure.com/courses/746/modules/items/13261)



# **Experiment No. 3**

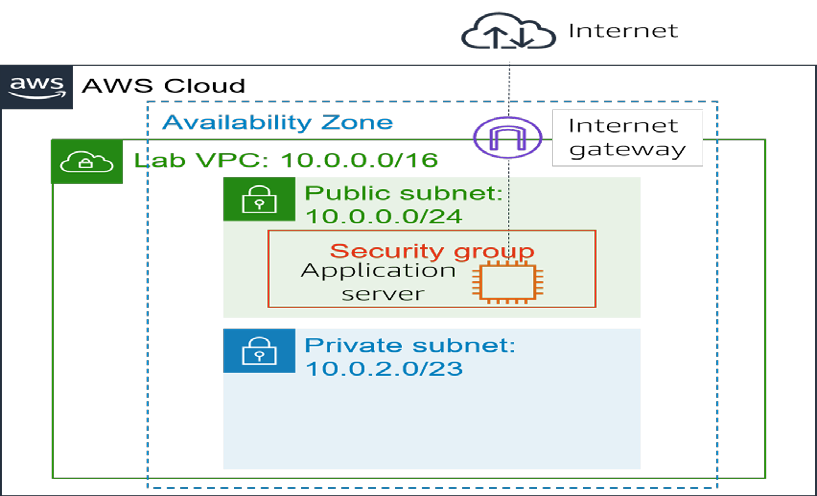
**Name : Sarvagya Singh**

**SAPID : 60009200030 BATCH : K1**

1. **Aim*:*** Create a Virtual Private Clouds and establish connections between each other.
2. **Objectives: In this lab, you will explore**
   * To create your own VPC and add additional components to produce a customized network.
   * To create a security group.
   * Configure and customize an EC2 instance to run a web server and launch the EC2 instance to run in a subnet in the VPC.
3. **Outcomes: After completion of lab**
   * Deploy a VPC
   * Create an internet gateway and attach it to the VPC
   * Create a public subnet
   * Create a private subnet
   * Create an application server to test the VPC
4. **Hardware / Software Required:** Internet, AWS console
5. **Theory:**

Amazon Virtual Private Cloud (Amazon VPC) enables you to launch AWS resources into a virtual network that you've defined. This virtual network closely resembles a traditional network that you'd operate in your own data center, with the benefits of using the scalable infrastructure of AWS.

In this lab students build the following infrastructure:



Task 1: Create Your VPC

A VPC is a virtual network that is dedicated to your Amazon Web Services (AWS) account. It is logically isolated from other virtual networks in the AWS Cloud. You can launch AWS resources, such as Amazon Elastic Compute Cloud (Amazon EC2) instances, into the VPC. You can configure the VPC by modifying its IP address range and can create subnets. You can also configure route tables, network gateways, and security settings.

1. In the AWS Management Console, on the **Services** menu, choose **VPC**.
2. In the left navigation pane, choose **Your VPCs**.

A default VPC is provided so that you can launch resources as soon as you start using AWS. There is also a shared VPC that you use later in the lab. However, you now create your own **Lab VPC**.

The VPC will have a Classless Inter-Domain Routing (CIDR) range of **10.0.0.0/16**, which includes all IP address that start with 10.0.x.x. It contains more than 65,000 addresses. You later divide the addresses into separate subnets.

1. Choose **Create VPC** and configure the following settings:
   * For **Name tag**, enter

Lab VPC

10.0.0.0/16

* + For **IPv4 CIDR block**, enter
  + Choose **Create VPC**.

1. From the **VPC Details** page, choose the **Tags** tab.
2. Choose **Actions** and select **Edit DNS hostnames**.

This option assigns a friendly Domain Name System (DNS) name to EC2 instances in the VPC, such as the following:

**ec2-52-42-133-255.us-west-2.compute.amazonaws.com**

1. Select **Enable**, and choose **Save changes Task 2: Creating subnets**

A subnet is a subrange of IP addresses in the VPC. AWS resources can be launched into a specified subnet. Use a public subnet for resources that must be connected to the internet, and use a private subnet for resources that must remain isolated from the internet.

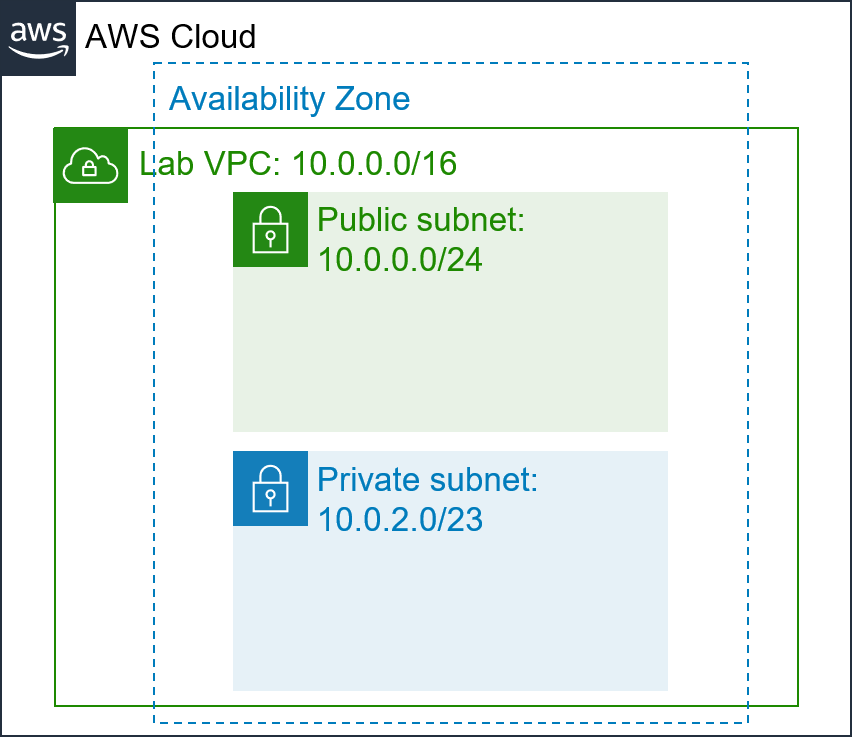
In this task, you create a public subnet and a private subnet shown in fig 7.2

**Create a public subnet**

1. In the left navigation pane, choose **Subnets**.
2. Choose **Create subnet** and configure the following settings:
   * For **VPC ID**, choose **Lab VPC**.
   * For **Subnet name**, enter Public Subnet
   * For **Availability zone**, select the first Availability Zone in the list. Do not choose **No Preference**.
   * For **IPv4 CIDR block**, enter

10.0.0.0/24

* + Choose **Create subnet**



**Figure 7.2 Public and Private Subnet**

**Create a private subnet**

1. Use what you learned in the previous steps to create another subnet with the following settings:
   * For **VPC ID**, choose **Lab VPC**.
   * For **Subnet name,** enter Private Subnet
   * For **Availability Zone**, select the first Availability Zone in the list. Do not choose **No Preference**.
   * For **IPv4 CIDR block**, enter

10.0.2.0/23

* + Choose **Create subnet**

VPC now has two subnets. However, the public subnet is totally isolated and cannot communicate with resources outside the VPC. Next, you configure the public subnet to connect to the internet via an internet gateway.

**Task 3: Creating an internet gateway**

1. In the left navigation pane, choose **Internet Gateways**.
2. Choose **Create internet gateway** and configure the following settings:

* For **Name tag**, enter

Lab IGW

* Choose **Create internet gateway**
  + You can now attach the internet gateway to your **Lab VPC**.

1. Choose **Actions** and then **Attach to VPC**, and configure the following settings:

* For **Available VPCs**, select **Lab VPC**.
* Choose **Attach internet gateway**
  + This action attaches the internet gateway to your **Lab VPC Task 4: Configuring route tables**

1. In the left navigation pane, choose **Route Tables**.
2. In the **VPC** column, find the route table that shows **Lab VPC**, and select the check box for this route table.
3. In the **Name** column, choose and then enter the name

Private Route Table

**Save**

1. In the lower half of the page, choose the **Routes** tab.
2. Choose **Create route table** and configure the following settings:
   * For **Name**, enter

Public Route Table

* + For **VPC**, choose **Lab VPC**.
  + Choose **Create route table**

1. In the **Routes** tab, choose **Edit routes**

and choose

You now add a route to direct internet-bound traffic (0.0.0.0/0) to the internet gateway.

1. Choose **Add route** and then configure the following settings:
   * For **Destination**, enter

0.0.0.0/0

* + For **Target**, select **Internet Gateway**, and then from the dropdown list select

Lab IGW**.**

* + Choose **Save changes**

The last step associates this new route table with the public subnet.

1. Choose the **Subnet associations** tab.
2. In the **Subnets without explicit associations** section, choose **Edit subnet associations**
3. Select the row with **Public Subnet**.
4. Choose **Save associations**

Task 5: Creating a security group for the application server

1. In the left navigation pane, choose **Security Groups**.
2. Choose **Create security group** and configure the following settings:
   * For **Security group name**, enter

App-SG

Allow HTTP traffic

* + For **Description**, enter
  + For **VPC**, choose **Lab VPC**.
  + Choose **Create security group**

1. Choose the **Inbound Rules** tab.
2. Choose **Edit inbound rules**
3. Choose **Add rule** and then configure the following settings:
   * For **Type**, choose **HTTP**.
   * From the **Source type** dropdown list, choose **Anywhere IPv4**.
   * For **Description**, enter Allow web access
   * Choose **Save rules**

You use this **App-SG** in the next task.

**Task 6: Launching an application server in the public subnet**

To test that your VPC is correctly configured, you now launch an EC2 instance into the public subnet. You also confirm that you can access the EC2 instance from the internet.

1. On the **Services** menu, choose **EC2**.
2. Choose **Launch instance** and then select **Launch instance** from the dropdown list. Configure the following options:
   * In the **Name and tags** pane, in the **Name** text box, enter

App Server

* + In the **Application and OS Images (Amazon Machine Image)** section, keep default selection, **Amazon Linux 2**.
  + In the **Instance type** section, keep the default instance type, **t2.micro**.
  + In the **Key pair (login)** section, from the **Key pair name - *required*** dropdown list, choose **Proceed without a key pair (not recommended)**.

In the **Network settings** section, choose **Edit**

* From the **VPC - *required*** dropdown list, choose **Lab VPC**.
* From the **Subnet** dropdown list, choose **Public Subnet**.
* Ensure that **Auto-assign public IP** is **Enable**.
* In the **Firewall (security groups)** section, choose **Select existing security group**
* From the **Common security groups** dropdown list, choose App-SG.
* In the **Configure storage** section, keep the default storage configuration.
* Expand the **Advanced details** section.
* For **IAM instance profile**, choose the role **Inventory-App-Role**.
* Scroll down to **User data** section, copy and paste the below code in the block.

#!/bin/bash

# Install Apache Web Server and PHP yum install -y httpd mysql

amazon-linux-extras install -y php7.2

# Download Lab files

wget https://aws-tc-largeobjects.s3-us-west-2.amazonaws.com/ILT-TF-200-

ACACAD-20-EN/mod6-guided/scripts/inventory-app.zip unzip inventory-app.zip -d /var/www/html/

# Download and install the AWS SDK for PHP

wget https://github.com/aws/aws-sdk-php/releases/download/3.62.3/aws.zip unzip aws -d /var/www/html

# Turn on web server chkconfig httpd on service httpd start

* + From the **Summary** section, choose **Launch instance**

1. Choose **View all instances**
2. Wait for the application server to fully launch. It should display the following status:

o **Instance State:** Running

1. Select **App Server**.
2. From the **Details** tab, copy the **Public IPv4 address** address.
3. Open a new browser tab, paste the IP address you just copied, and press Enter.

If you configured the VPC correctly, the Inventory application and this message should appear: **Please configure Settings to connect to database**.

1. **Result : Paste your screen shots for every task.**
2. **Conclusions :**

With Amazon Virtual Private Cloud (Amazon VPC), you can launch AWS resources in a logically isolated virtual network that you've defined. This virtual network closely resembles a traditional network that you'd operate in your own data center, with the benefits of using the scalable infrastructure of AWS.

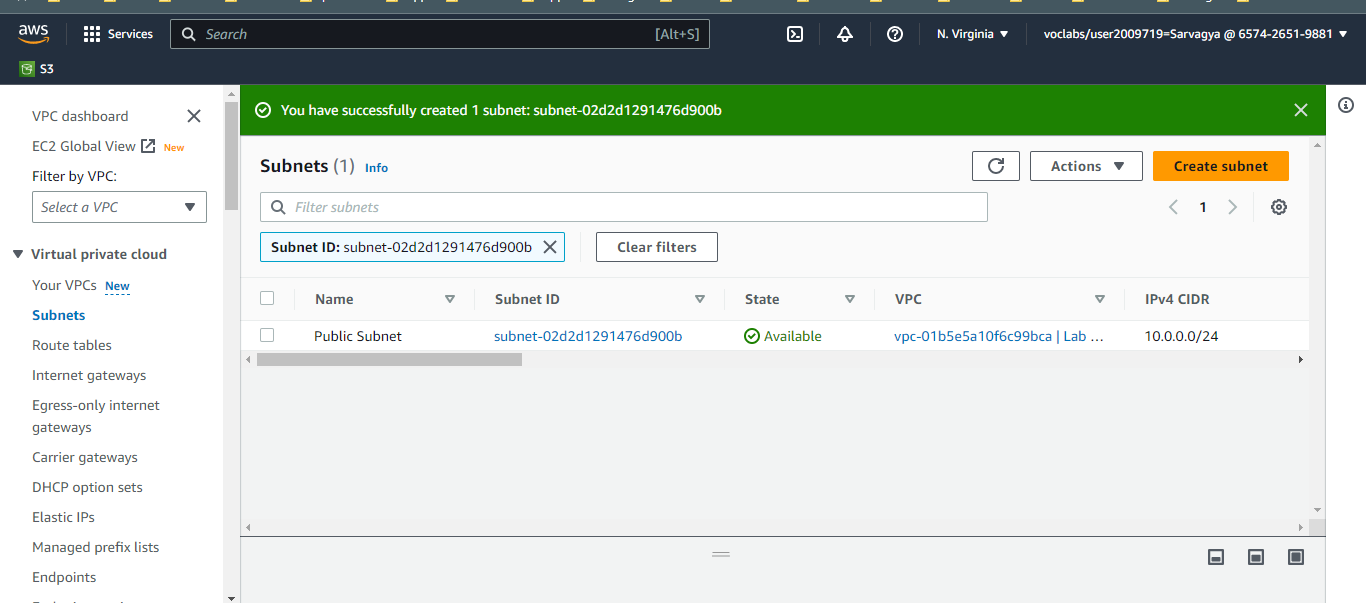
The following diagram shows an example VPC. The VPC has one subnet in each of the Availability Zones in the Region, EC2 instances in each subnet, and an internet gateway to allow communication between the resources in your VPC and the internet.

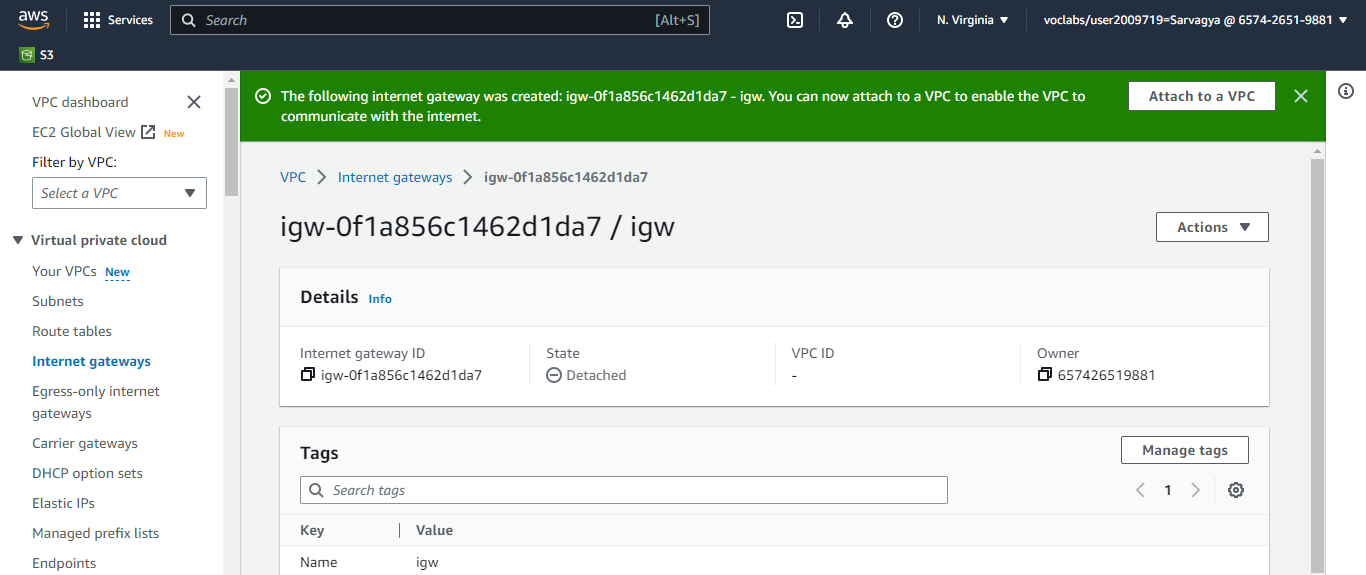
1. **Viva Questions:**
   * What are the components of Amazon VPC
   * What are Internet Gateways in VPC?
   * What do you know about VPC Peering?

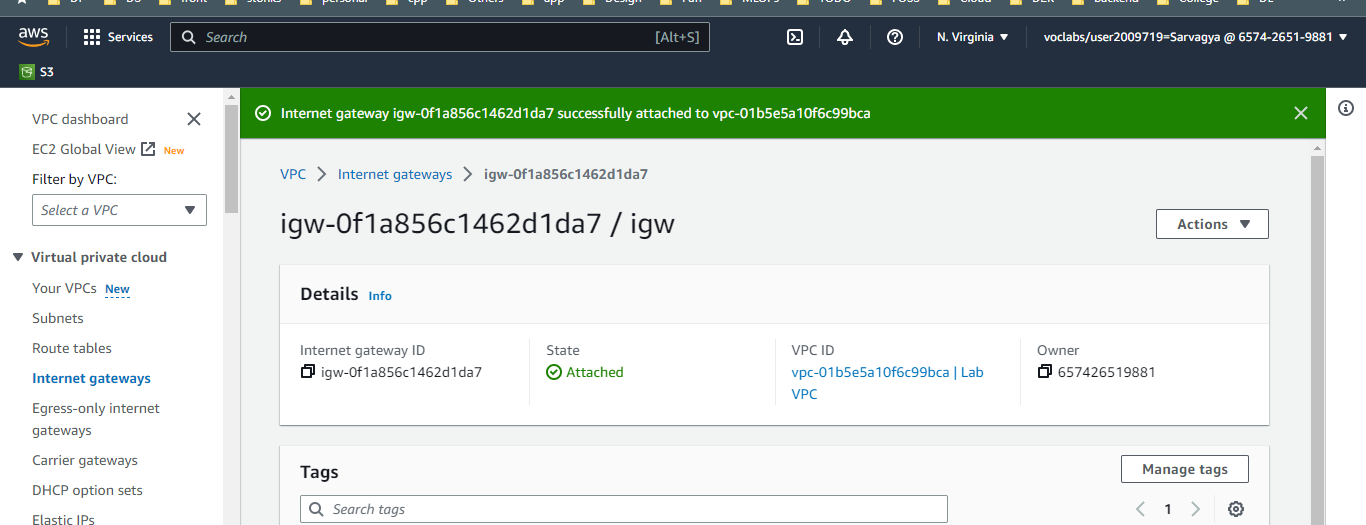
**References:** <https://docs.aws.amazon.com/vpc/latest/userguide/what-is-amazon-vpc.html>

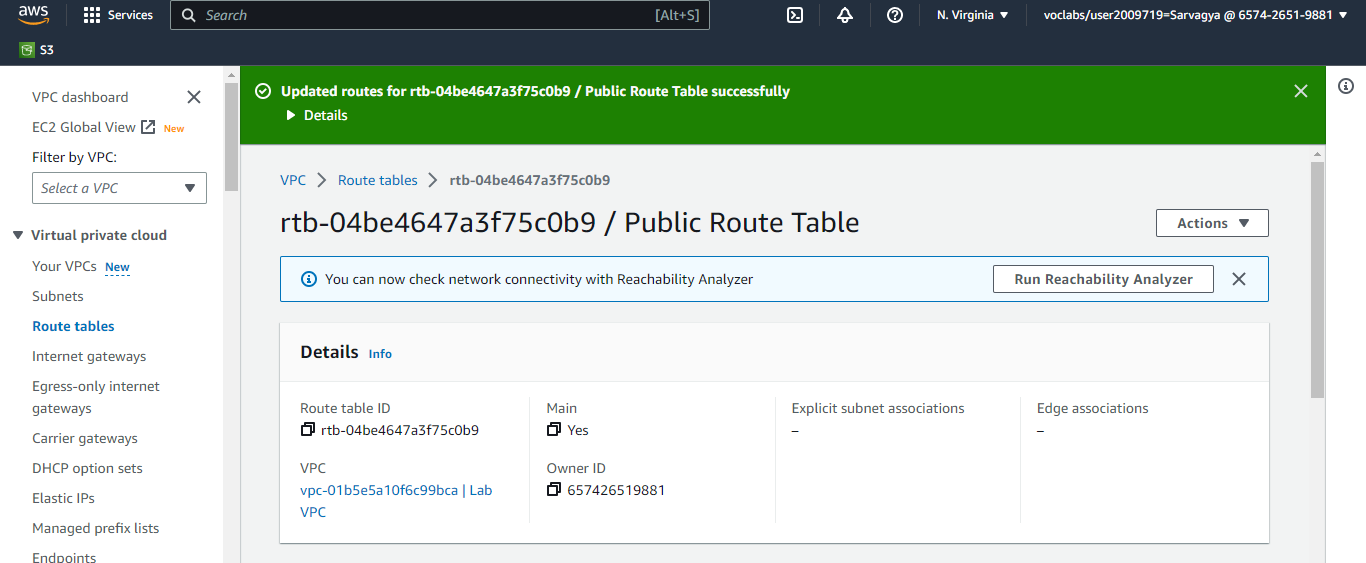
1. **Screenshots :**

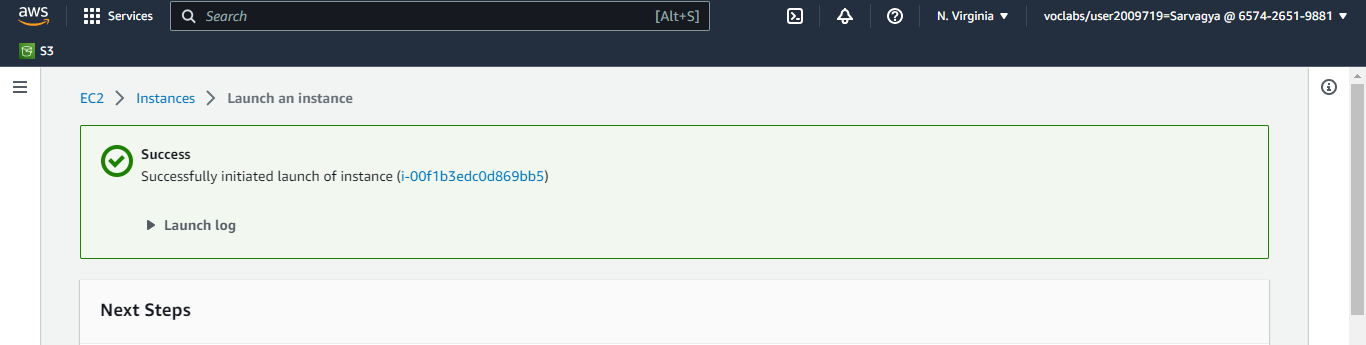
**Challenge lab:**

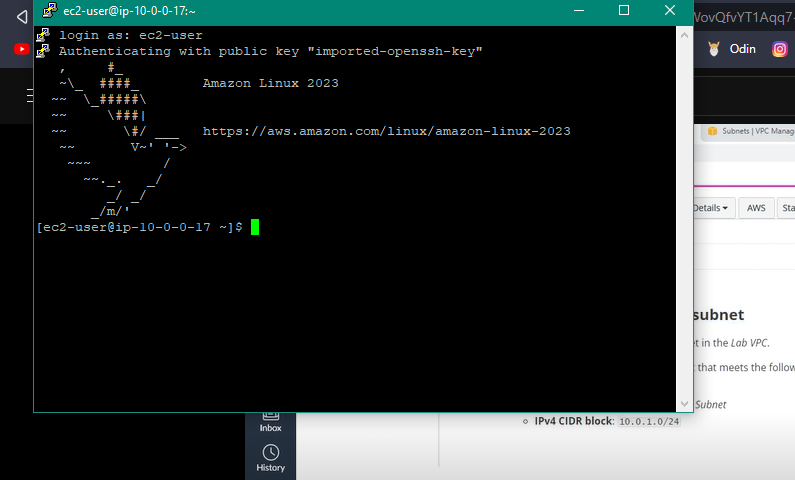
****

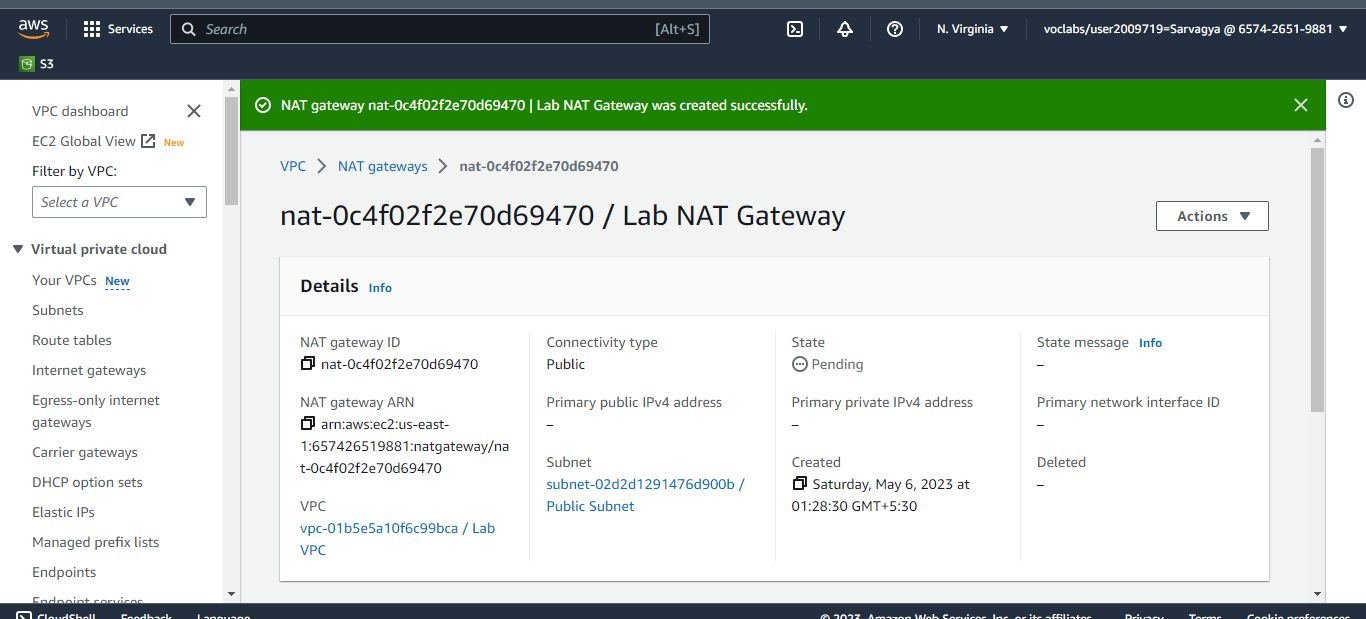
****

****

****

****

****

****