Cloud Architect, Cloud Network Engineer, Cloud Security Engineer, Cloud Administrator

Networking

**Lab Steps**

Task 1: Sign in to AWS Management Console

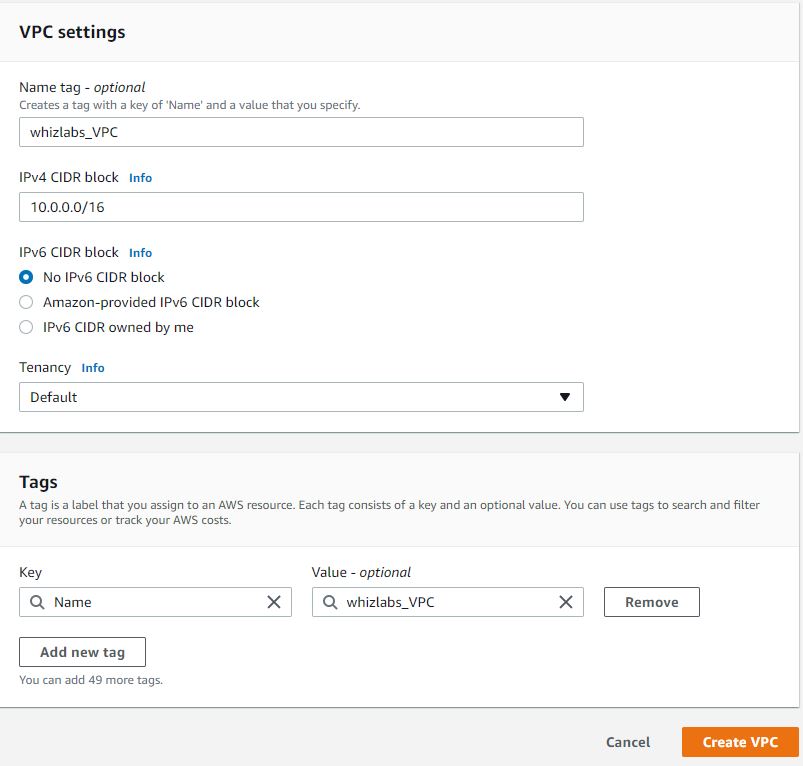
1. Click on the  button, and you will get redirected to AWS Console in a new browser tab.
2. On the AWS sign-in page,
   * Leave the Account ID as default. Never edit/remove the 12 digit Account ID present in the AWS Console. otherwise, you cannot proceed with the lab.
   * Now copy your **User Name** and **Password** in the Lab Console to the **IAM Username and Password** in AWS Console and click on the **Sign in** button.
3. Once Signed In to the AWS Management Console, Make the default AWS Region as **US East (N. Virginia) us-east-1.**
4. Select Maybe later in New AWS Console Home page pop-up

**Note: There is no Validation function for this lab.**

Task 2: Creating a new VPC

1. Make sure to choose the  **N.Virginia** region in the AWS Management Control dashboard, which is present in the top right corner.
2. Navigate and click on  **VPC** which will be available under the **Networking & Content Delivery**section of  
3. Click on **Your VPCs**from the left menu and Click on 
4. In **Create VPC** page fill the following details,

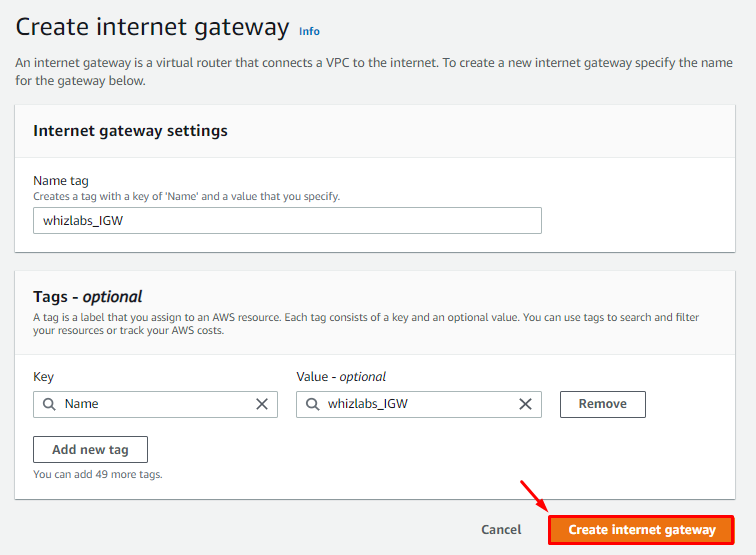
* Select **VPC Only**
* Name tag: Enter ***whizlabs\_VPC***
* IPV4 CIDR Block: Enter ***10.0.0.0/16***
* IPV6 CIDR block:  Select **No IPV6 CIDR block**
* Tenancy:  **Default**
* Click on **Create** **VPC**button**.**



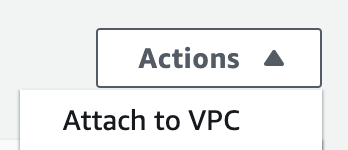
Task 3: Creating and attaching an Internet gateway

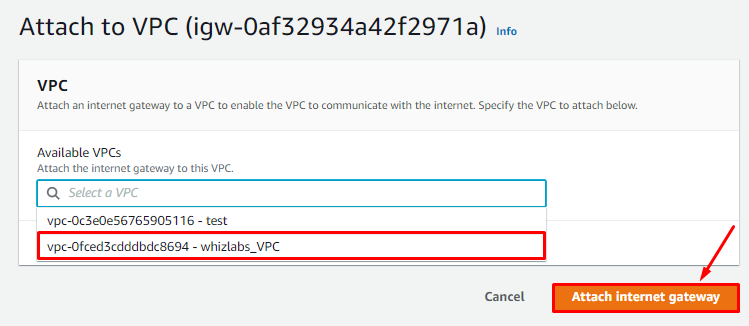
1. Click on **Internet Gateways**from the left menu and click  and enter the following details:

* Name tag: Enter ***whizlabs\_IGW***
* Click on **Create** **internet gateway** button.



1. Select the Internet gateway you created from the list.

* Click on **Actions**.
* Click on **Attach to VPC**.  
  
* Select MyVPC and click on **Attach internet gateway.**



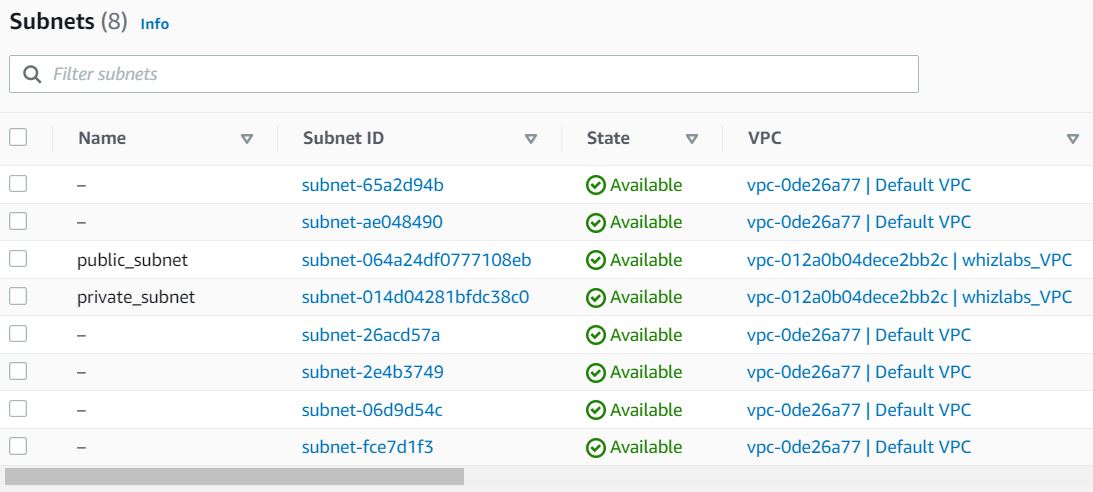
Task 4: Creating two Subnets

1. You will create **2 Subnets,** one for public and another for private resources. First, we will create a **public subnet**.
2. For the Public Subnet**,** click on **Subnets** from the left menu and click on ****

* VPC ID: Select **whizlabs\_VPC** (Select the VPC which you created from the dropdown)
* Name tag: Enter ***public\_subnet***
* Availability zone: Select **No Preference**
* IPv4 CIDR block: Enter ***10.0.1.0/24***
* Click on**Create subnet**button

1. For Private Subnet**,** click on **Create Subnet**again.

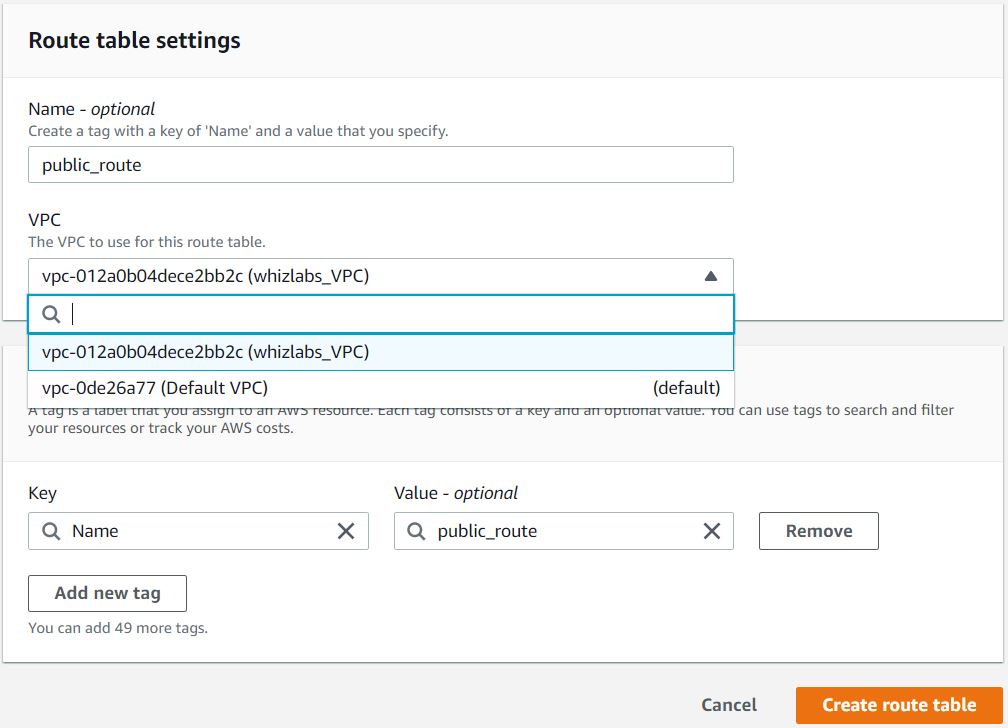
* VPC ID: Select **whizlabs\_VPC** (Select the VPC which you created from the dropdown)
* Name tag: Enter ***private\_subnet***
* Availability zone: Select **No Preference**
* IPv4 CIDR block: Enter ***10.0.2.0/24***
* Click on**Create subnet**button



Task 5: Creating Route tables, configuring routes and associating them with Subnets

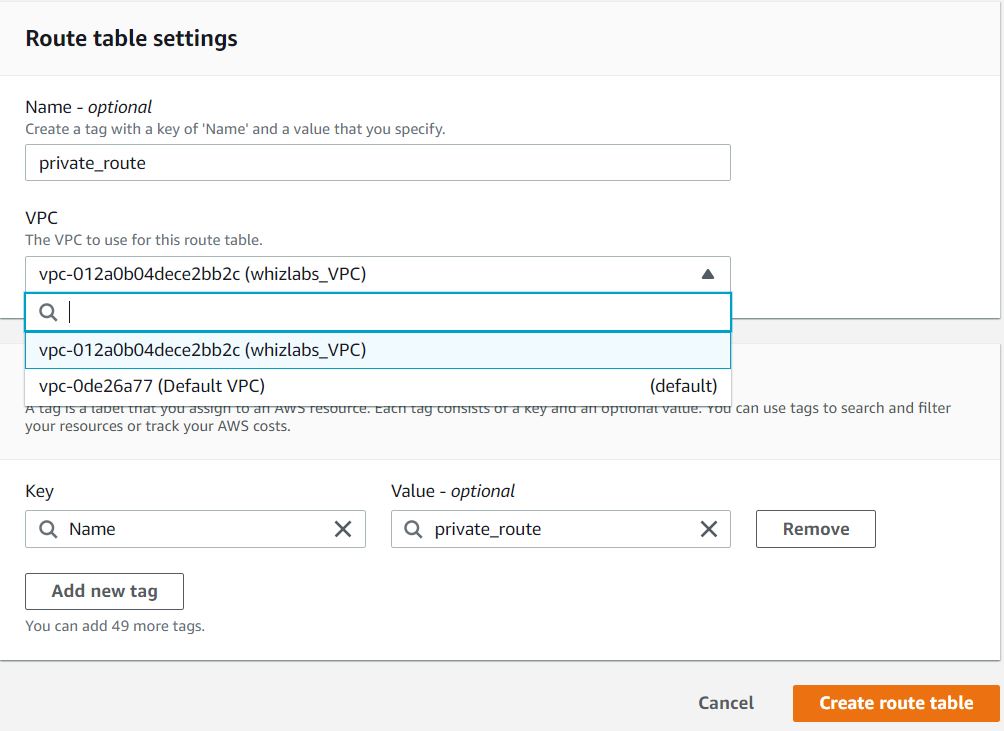
1. You will create **2 route tables,** one for public routes and another for private routes.
2. Go to **Route Tables** from the left menu and click on ****

* Name tag: Enter ***public\_route***
* VPC: Select **whizlabs\_VPC** (Select the VPC you created from the dropdown)
* Click on **Create** **route table**button**.**



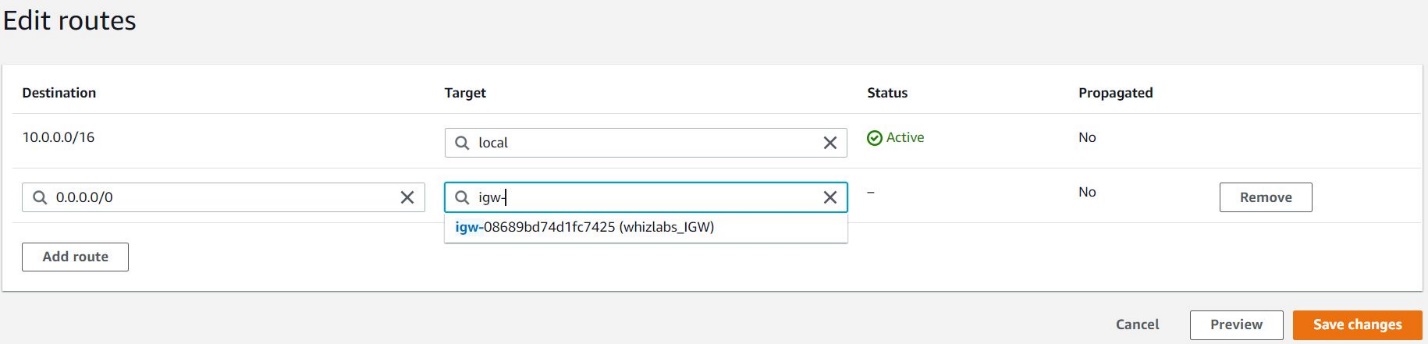
1. Similarly, go to **Route Tables** from the left menu and click on **Create route table.**

* Name tag: Enter ***private\_route***
* VPC: Select **whizlabs\_VPC** (Select the VPC you created from the dropdown)
* Click on **Create** **route table**button.

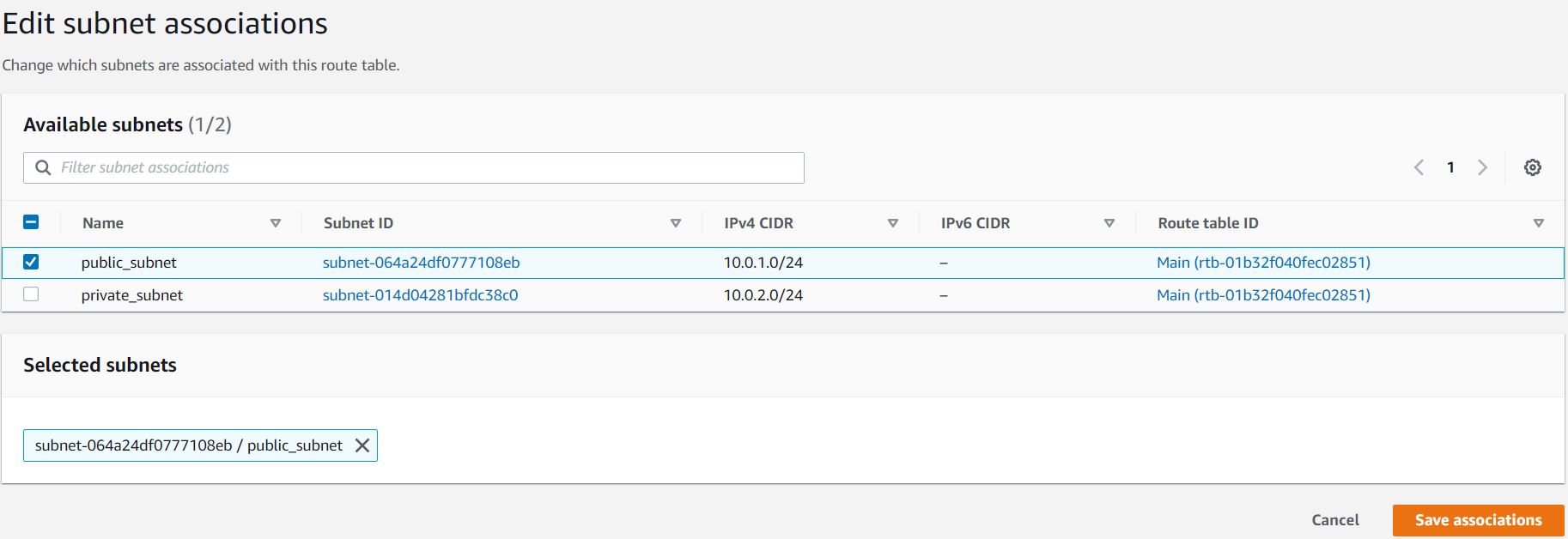


1. Now, you need to **add routes to the Route Tables**.

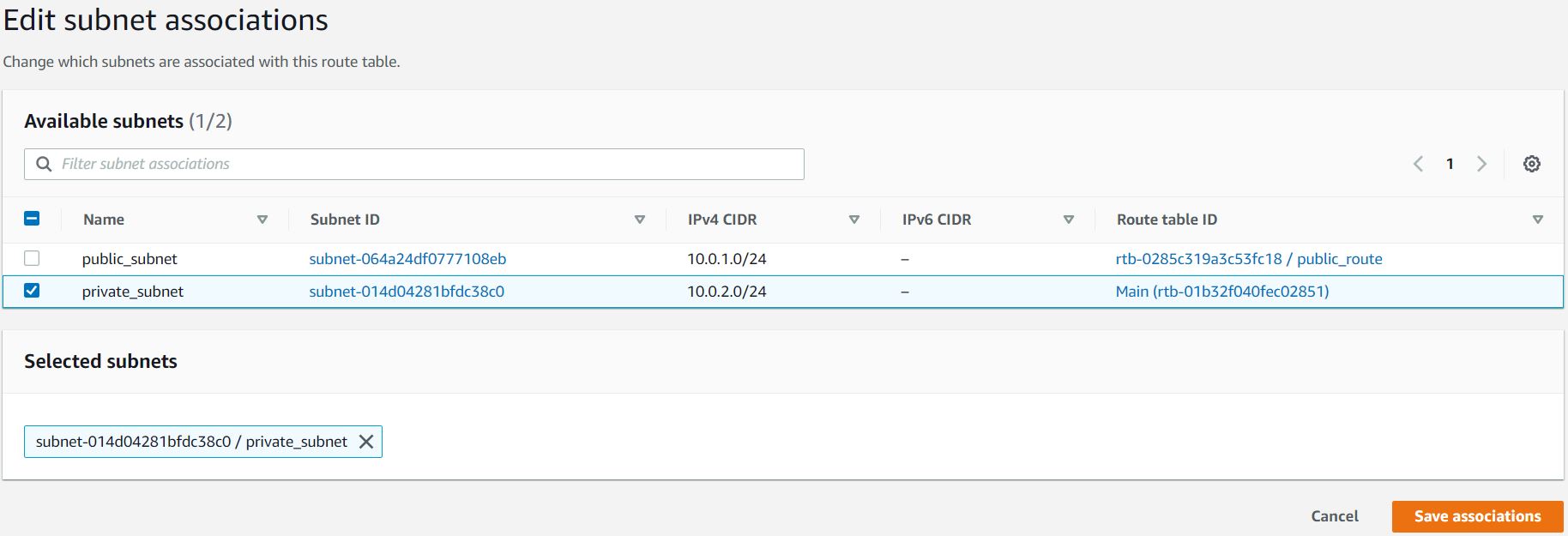
* Select **public\_route.**
* Go to the **Routes** tab click on **Edit routes**. On the next page, click on **Add route.**
* Specify the following values:
  + **Destination:** Enter ***0.0.0.0/0***
  + **Target:** Select **Internet Gateway** from the dropdown menu to select **whizlabs\_IGW**.
  + Click on **Save changes.**



1. Next, you need to associate the **public\_subnet**with this **public\_route**. Select the **public\_route** and go to the **Action** and in that go to **Edit Subnet Associations tab.**
   * Click on  **Edit Subnet Associations**.
   * Select **public\_subnet** from the list.
   * Click on **Save Associations**



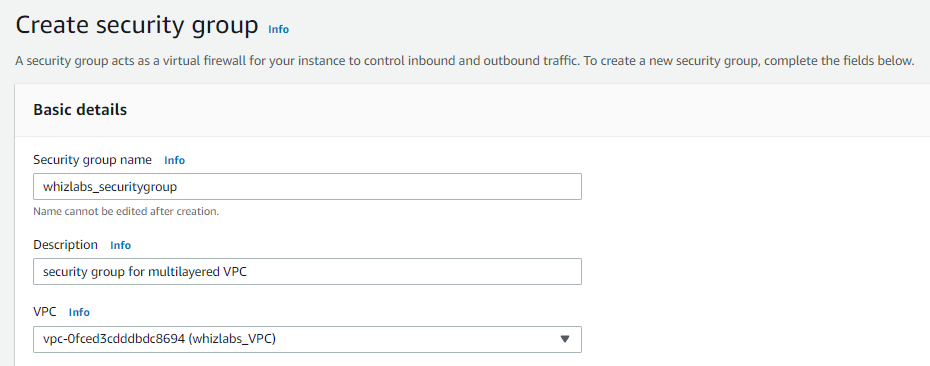
1. Similarly, you need to associate the **private\_subnet**with this **private\_route**. Select the **private\_route** and go to the **Action** and in that go to **Edit Subnet Associations tab.**
   * Click on  **Edit Subnet Associations**.
   * Select **private\_subnet** from the list.
   * Click on **Save Associations**

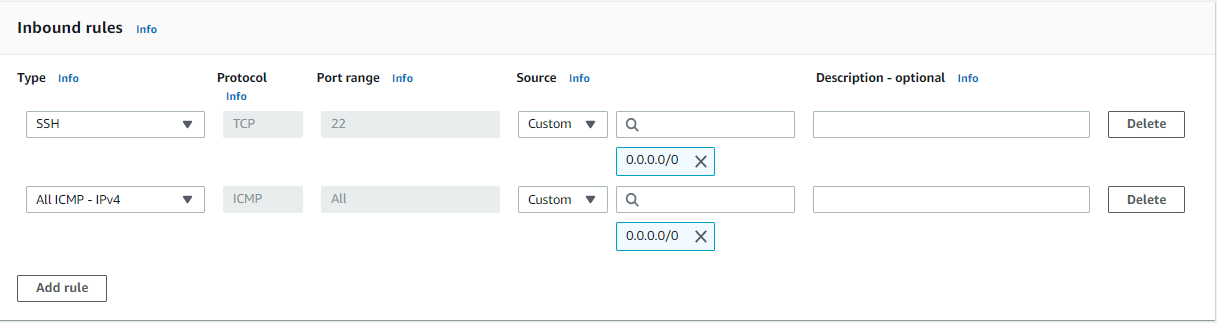


Task 6: Creating Security Group

1. Go to **Security Group** from the left menu and click on****and then provide the following details**:**

* Security group name: Enter ***whizlabs\_securitygroup***
* Description: Enter ***security group for multilayered VPC***
* VPC: Select **whizlabs\_VPC** (select from the dropdown)
* Under **Inbound Rules,** click on **Add Rule**.
* To add **SSH**,
  + Choose Type: 
  + Source: **Custom - 0.0.0.0/0**
* To add **All ICMP - IPv4,**
  + Click on **Add Rule**
  + Choose Type: **All ICMP - IPv4**
  + Source: **Anywhere IPv4**
* Click on 

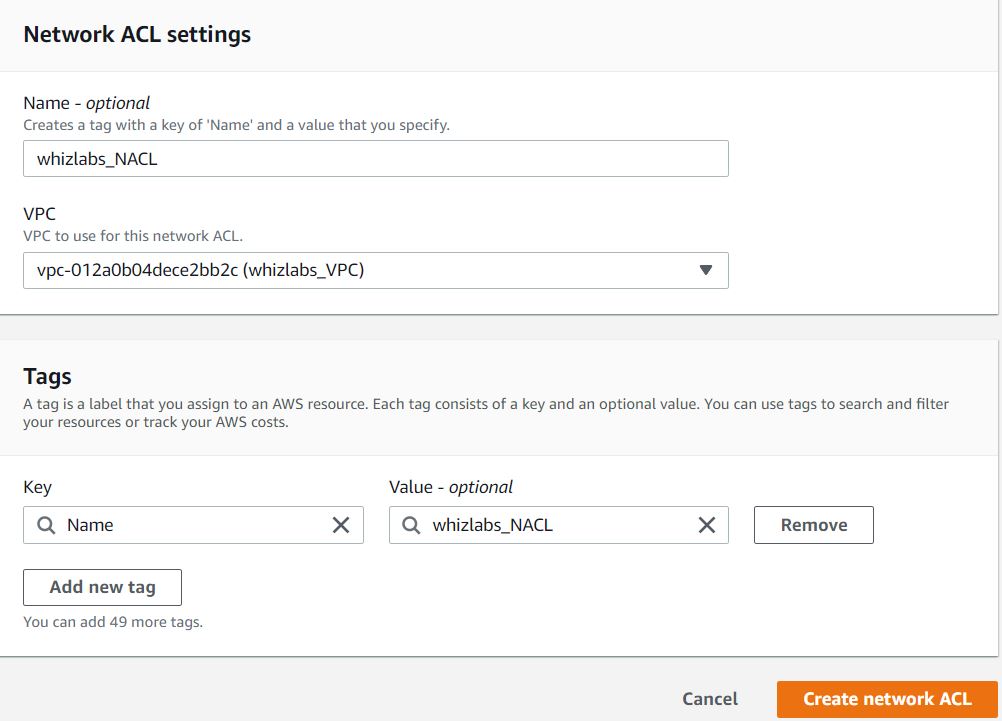




Task 7: Creating and configuring Network ACL

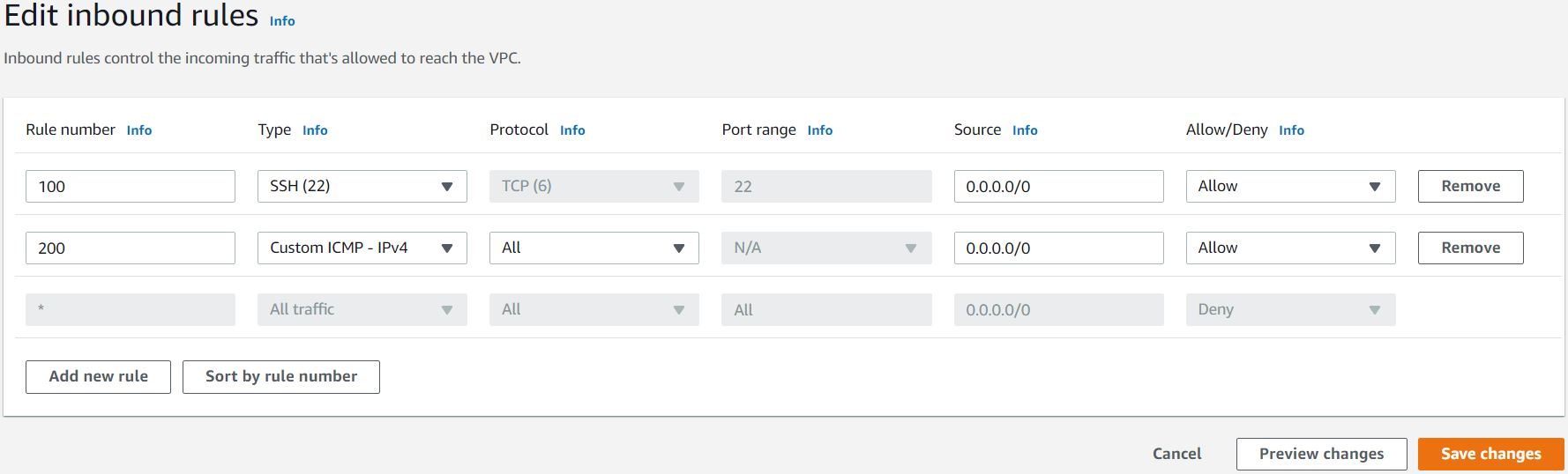
1. Go to **Network ACLs** from the left menu and click on **** Provide the following details:

* Name tag: Enter ***whizlabs\_NACL***
* VPC: Select **whizlabs\_VPC**(Select the VPC which you created from the dropdown)
* Click on **Create network ACL**button.



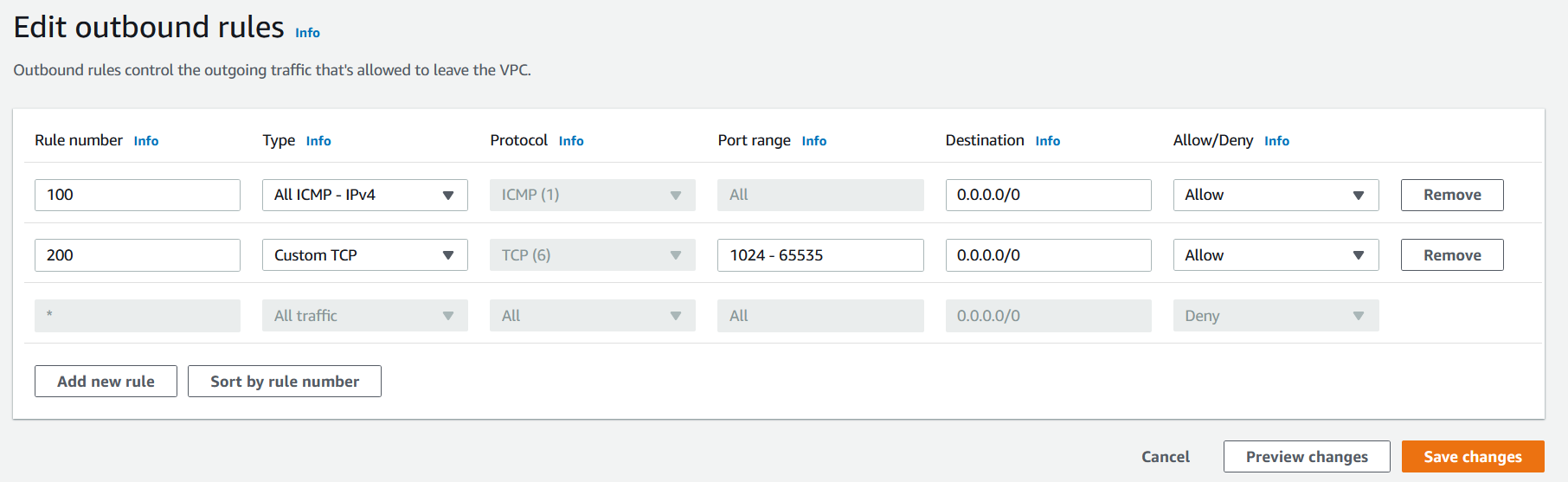
1. Select **whizlabs\_NACL**and go to **Inbound rules tab.**Click on**Edit inbound rules** and then click on the **Add new rule.**
2. Add the following rules:

* For **SSH**, click on **Add new rule**,
  + Rule number : Enter ***100***
  + Type: Choose ***SSH (22)***
  + Source: Enter ***0.0.0.0/0***
  + Allow / Deny: Select **Allow**
* For **ALL ICMP- IPv4**, click on **Add new rule**,
  + Rule number : Enter ***200***
  + Type: Choose ***ALL ICMP - IPv4***
  + Source: Enter ***0.0.0.0/0***
  + Allow / Deny: Select **Allow**
* Click on **Save changes*.***

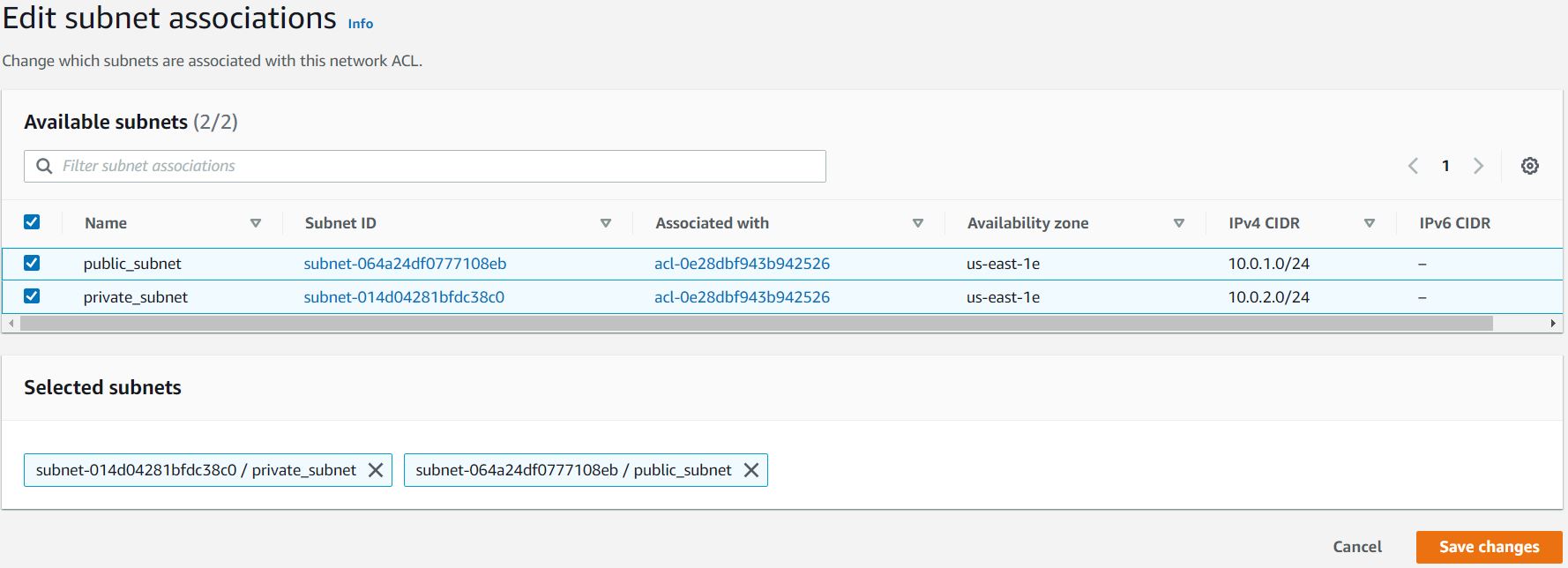


1. NACLs are stateless. You need to add the rules in Outbound rules too.
2. Select **whizlabs\_NACL**and go to **Onbound rules tab.**Click on**Edit onbound rules** and then click on the **Add Rule.**

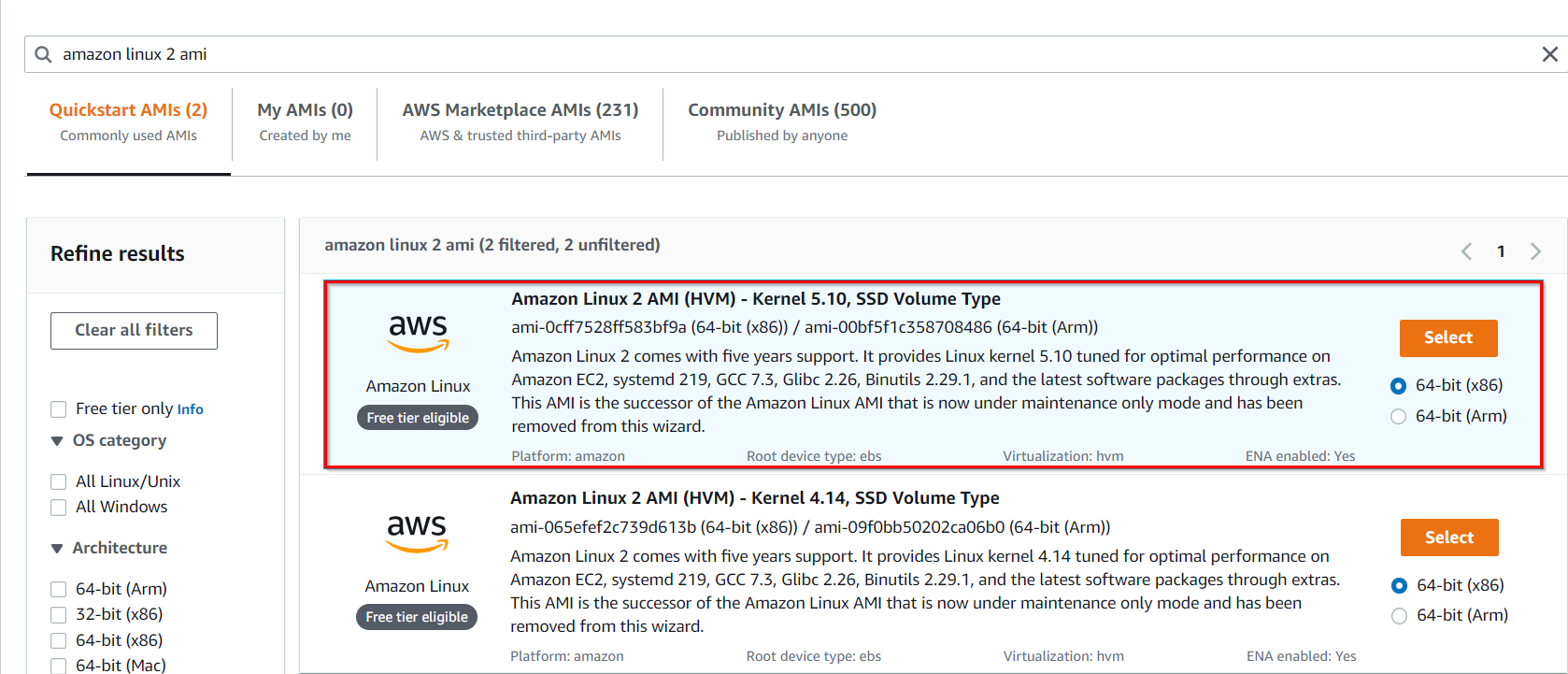
* For **ALL ICMP- IPv4**, click on **Add new rule**,
  + Rule# : Enter ***200***
  + Type: Choose ***ALL ICMP - IPv4***
  + Destination: Enter ***0.0.0.0/0***
  + Allow / Deny: Select **Allow**
* For **Custom TCP Rule**, click on **Add new rule**,
  + Rule# : Enter ***300***
  + Type: Choose ***Custom TCP Rule***
  + Port Range: Enter ***1024 - 65535***
  + Destination: Enter ***0.0.0.0/0***
  + Allow / Deny: Select **Allow**
* Click on **Save changes*.***



1. You need to associate both public and private subnets with the NACL.
2. Select **whizlabs\_NACL**and go to the **Subnet associations tab.**Click on**Edit subnet associations.**
3. Choose all the available subnets and then click on **Save changes** button.



Task 8: Launching 2 EC2 Instances

1. Navigate to **Services** and choose **EC2** under **Compute**
2. Navigate to **Instances** on the left panel and click on ****
3. Name : Enter ***public\_instance***
4. **For Amazon Machine Image (AMI):** Search for **Amazon Linux 2 AMI** in the search box and click on the **select** button.  
   

**Note: if there are two AMI's present for Amazon Linux 2 AMI, choose any of them.**

1. **For Instance Type:** select ***t2.micro***

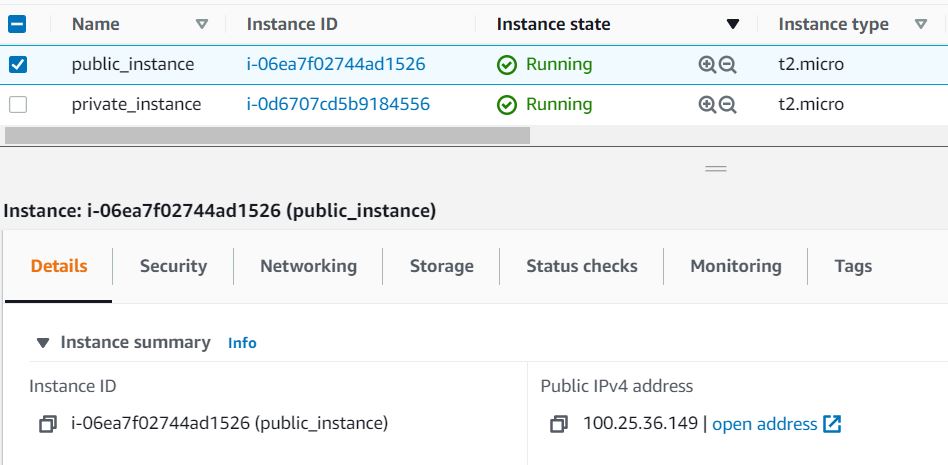
******

1. **For Key pair:**Select **Create a new key pair**Button
   1. Key pair name: **WhizKey**
   2. Key pair type:**RSA**
   3. Private key file format: **.pem**
2. Select **Create key pair** Button.

1. In Network Settings Click on **Edit**:
   1. VPC: Select **whizlabs\_VPC**
   2. Subnet: Select **public\_subnet**
   3. Auto-assign public IP: **Enable**
   4. Choose **Select an existing security group** and remove default one then select **whizlabs\_securitygroup**
2. Keep Rest thing Default and Click on**Launch Instance**Button.
3. Select **View all Instances** to View Instance you Created
4. Similar to the above, launch **another EC2 instance:**
   1. **Name** the instance as ***private\_instance***
   2. Key pair : Select Existing one
   3. VPC: Select **whizlabs\_VPC**
   4. Subnet: Select **private\_subnet**
   5. Auto-assign public IP: **Disable**
   6. Choose **Select an existing security group** and remove default one then select **whizlabs\_securitygroup**
5. Keep Rest thing Default and Click on**Launch Instance**Button.
6. Select **View all Instances** to View Instance you Created

Task 9: Testing the EC2 instances

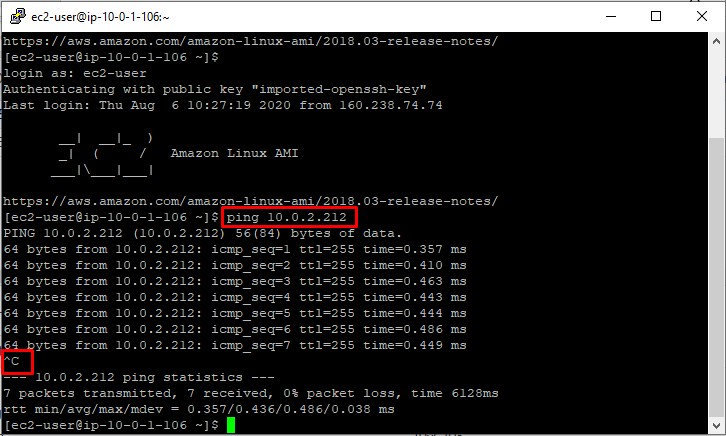
1. Select the **publc\_instance** from the EC2 dashboard and copy the public IPv4 address.



1. Similarly, copy the **Private IPv4 Address** of the **private\_instance**.
2. SSH into **public\_instance**EC2 Instance.

* Please follow the steps in [SSH into EC2 Instance](https://business.whizlabs.com/labs/support-document/ssh-into-ec-instance).

1. **Ping**the**private IP** of your **private\_instance** by using the below command:
   * **ping <your Private EC2 IPv4 address>**
2. Once you execute this command, you will receive a response from the IP.



1. Press [Ctrl] + C to cancel the process.

**Completion and Conclusion**

1. You have successfully created a new VPC.
2. You have successfully created and attached an Internet Gateway.
3. You have successfully created two subnets for public and private AWS instances.
4. You have successfully created and configured the Route Table.
5. You have successfully created a Security Group.
6. You have successfully created and configured Network ACL.
7. You have successfully launched 2 EC2 instances (one in a public subnet and one in a private subnet).
8. You have successfully tested the EC2 instance.

**End Lab**

1. Sign out from the AWS Account.
2. You have successfully completed the lab.
3. Once you have completed the steps click on