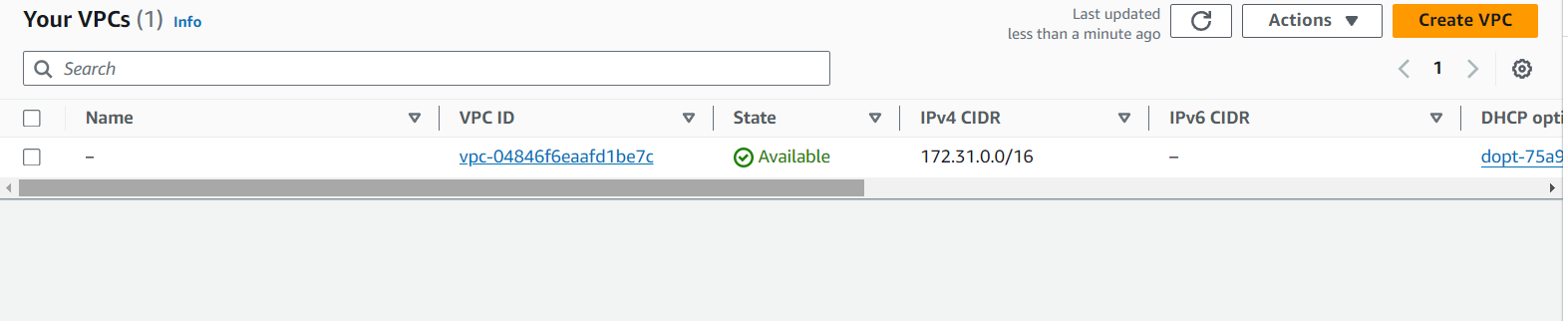
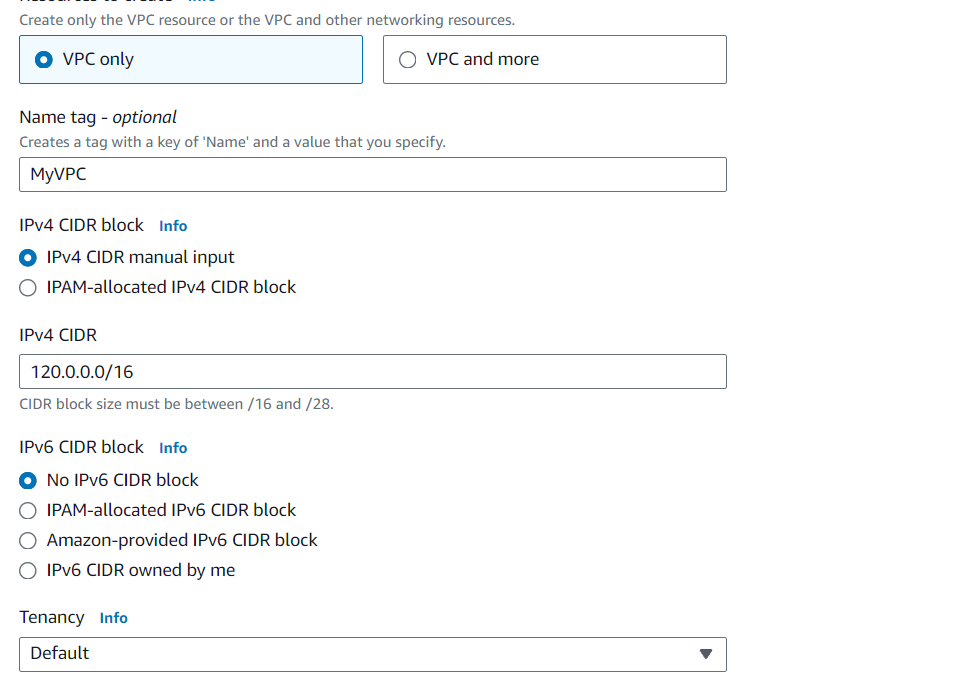
**1. Create a VPC**

* **Step 1**: Log in to the AWS Management Console and go to the VPC Dashboard.
* **Step 2**: Click on **Create VPC**.



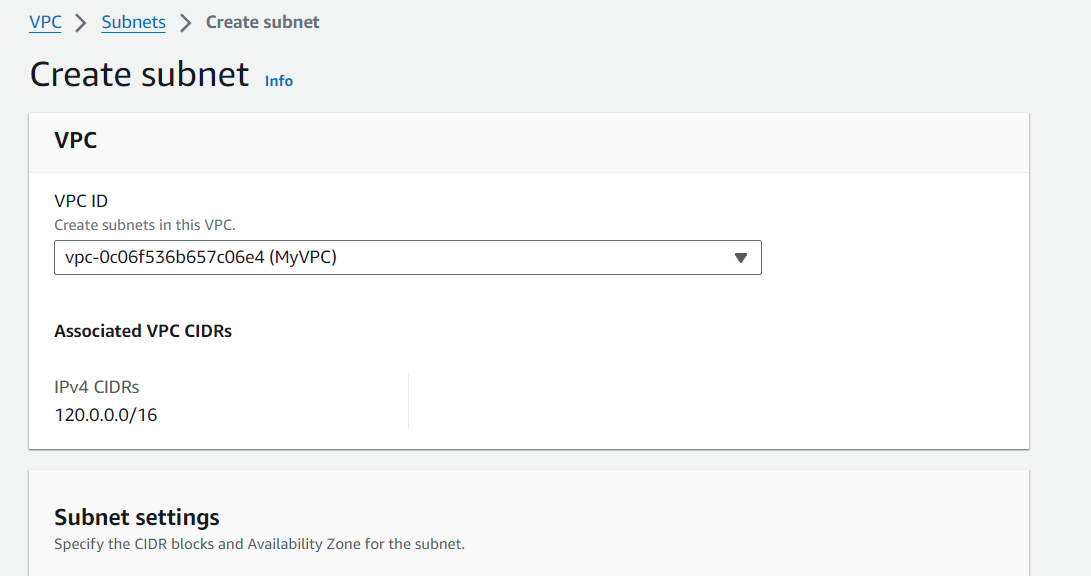
* **Step 3**: Choose **VPC only**.
* **Step 4**: Enter the following details:
  + **Name tag**: (e.g., MyVPC)
  + **IPv4 CIDR block**: 120.0.0.0/16



* **Step 5**: Click **Create VPC**.

**2. Create Subnets**

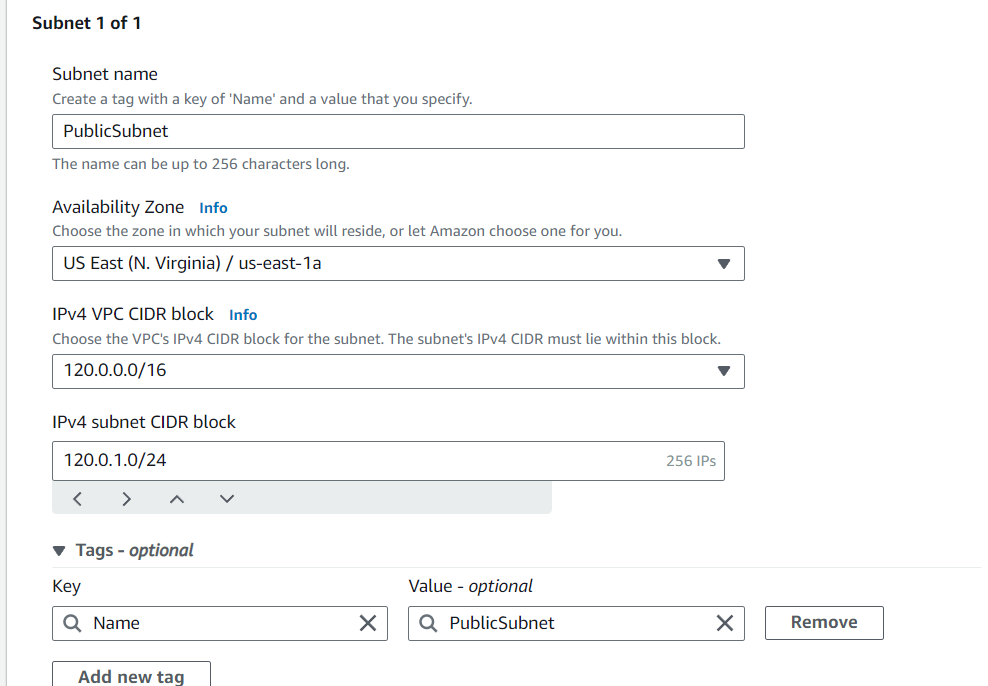
* **Step 1**: In the VPC Dashboard, click on **Subnets** in the left navigation pane, then click **Create Subnet**.
* **Step 2**: Select the VPC you just created.



* **Step 3**: Create the public and private subnets:

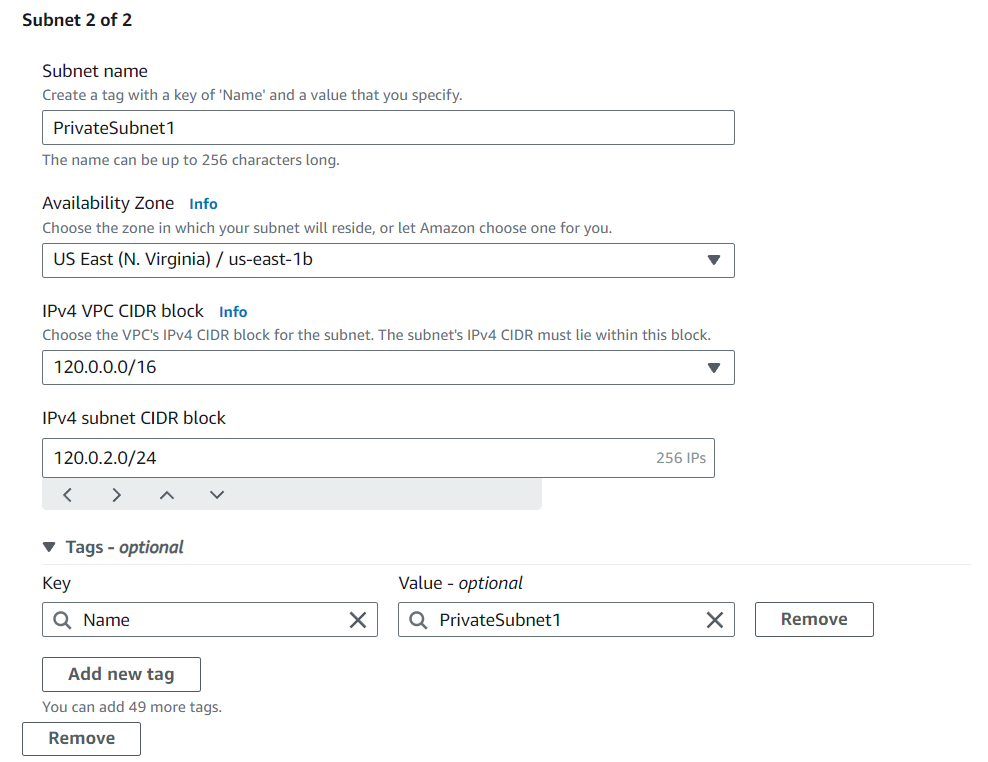
**Public Subnet**

* + **Name tag**: (e.g., PublicSubnet)
  + **Availability Zone**: Choose one (e.g., us-east-1a).
  + **IPv4 CIDR block**: (e.g., 120.0.1.0/24)
  + **Step 4**: Click **Create Subnet**.



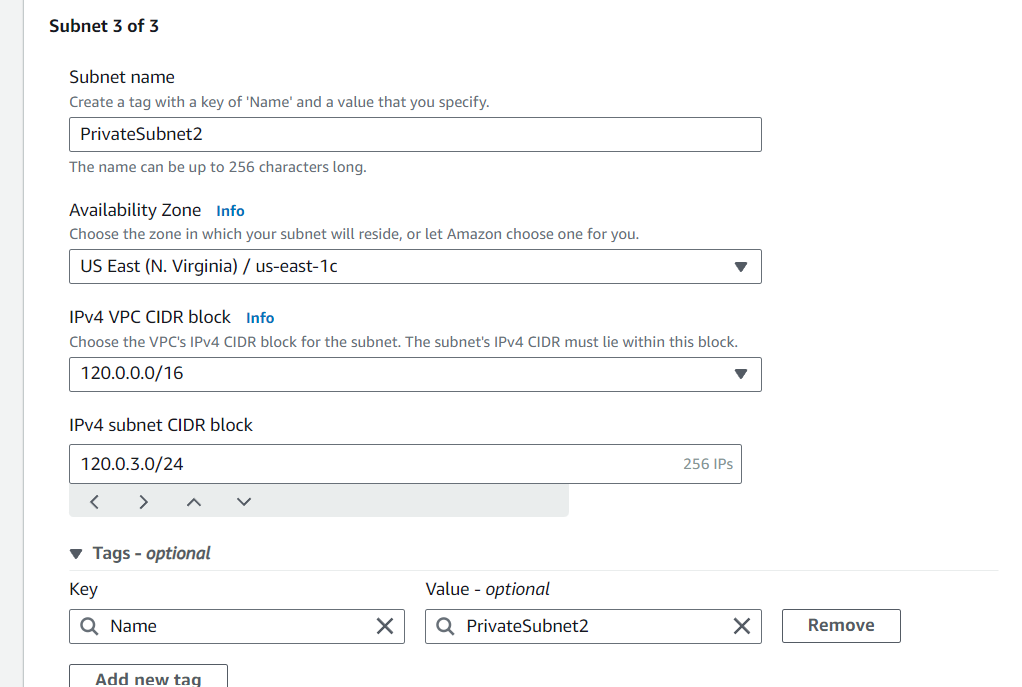
**Private Subnet 1**

* + **Name tag**: (e.g., PrivateSubnet1)
  + **Availability Zone**: Choose another one (e.g., us-east-1b).
  + **IPv4 CIDR block**: (e.g., 120.0.2.0/24)
  + **Step 4**: Click **Create Subnet**.

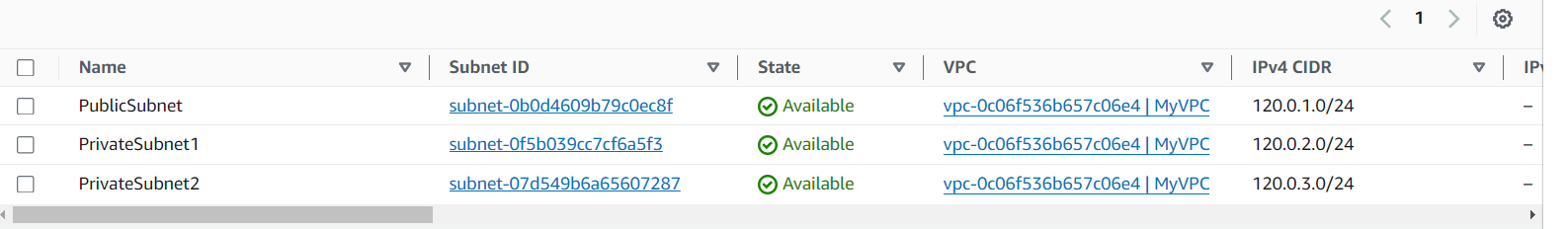


**Private Subnet 2**

* + **Name tag**: (e.g., PrivateSubnet2)
  + **Availability Zone**: Choose the third (e.g., us-east-1c).
  + **IPv4 CIDR block**: (e.g., 120.0.3.0/24)

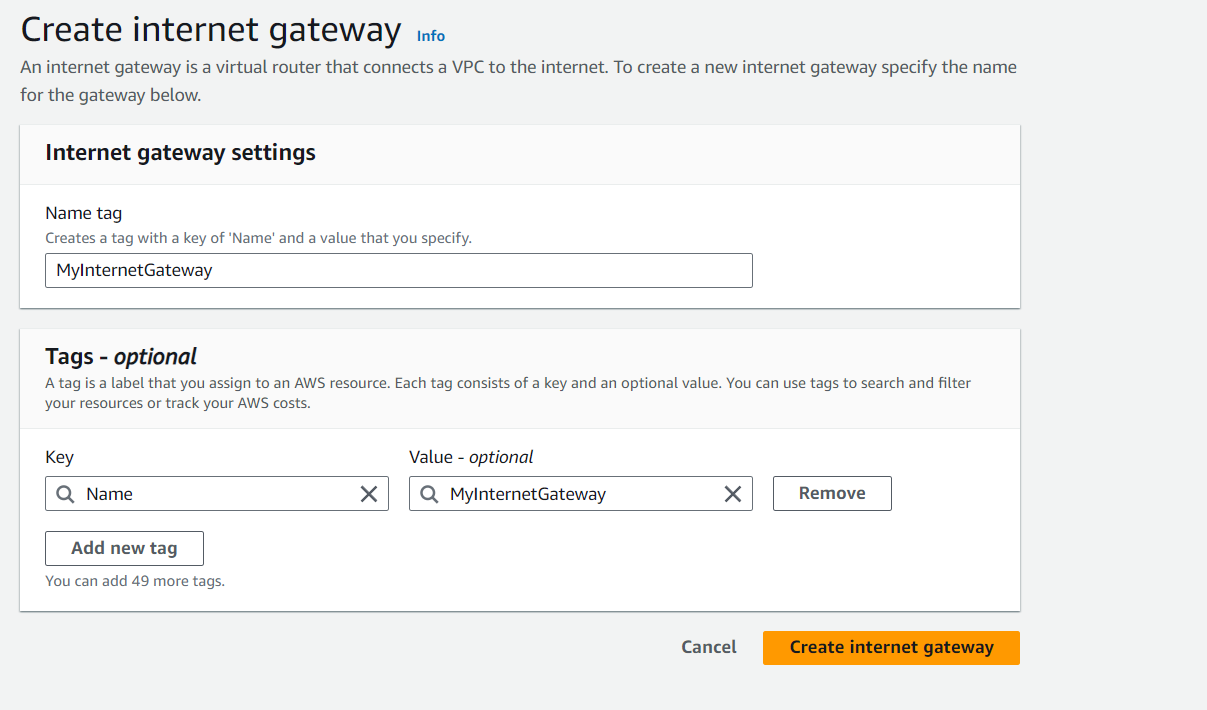


* + **Step 4**: Click **Create Subnet**.

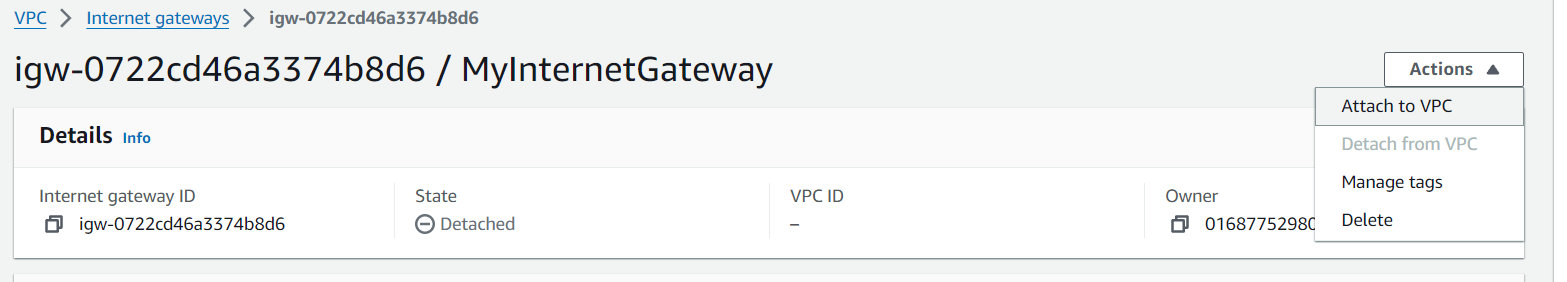


**3. Create an Internet Gateway and Attach it to the VPC**

* **Step 1**: In the VPC Dashboard, click on **Internet Gateways** in the left navigation pane, then click **Create internet gateway**.
* **Step 2**: Enter a name tag (e.g., MyInternetGateway), then click **Create internet gateway**.

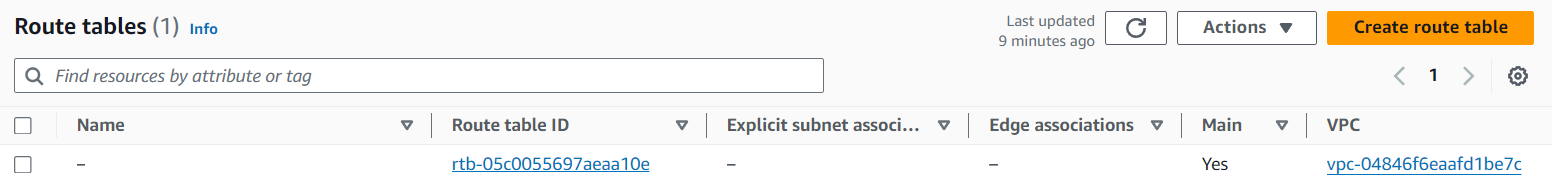


* **Step 3**: Click **Attach to VPC**, select the VPC you created, and click **Attach internet gateway**.

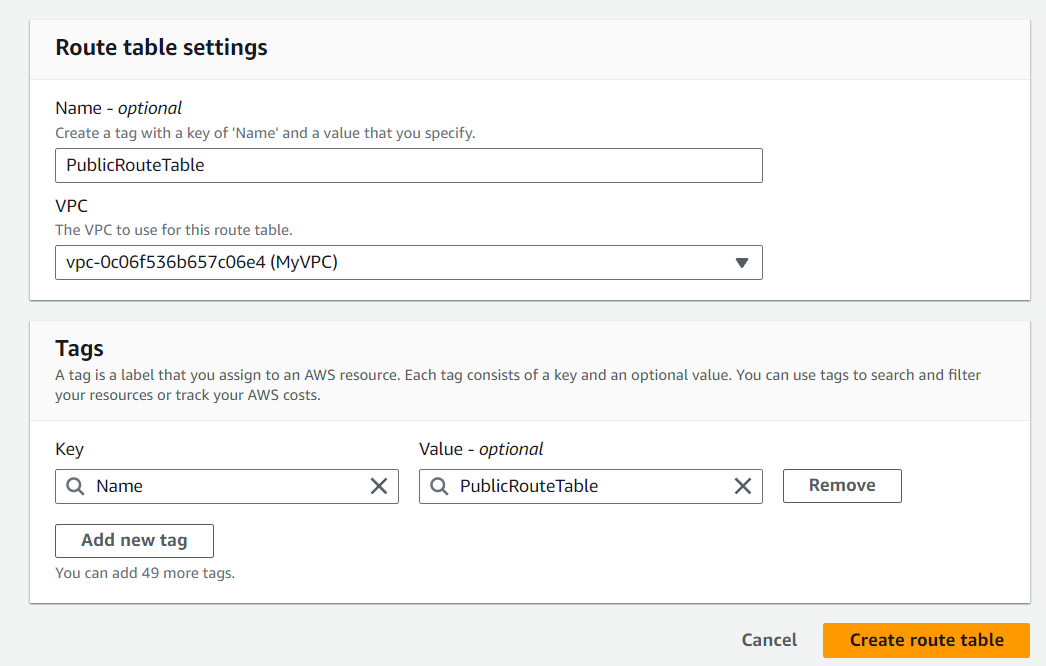


**4. Create a Route Table for the Public Subnet**

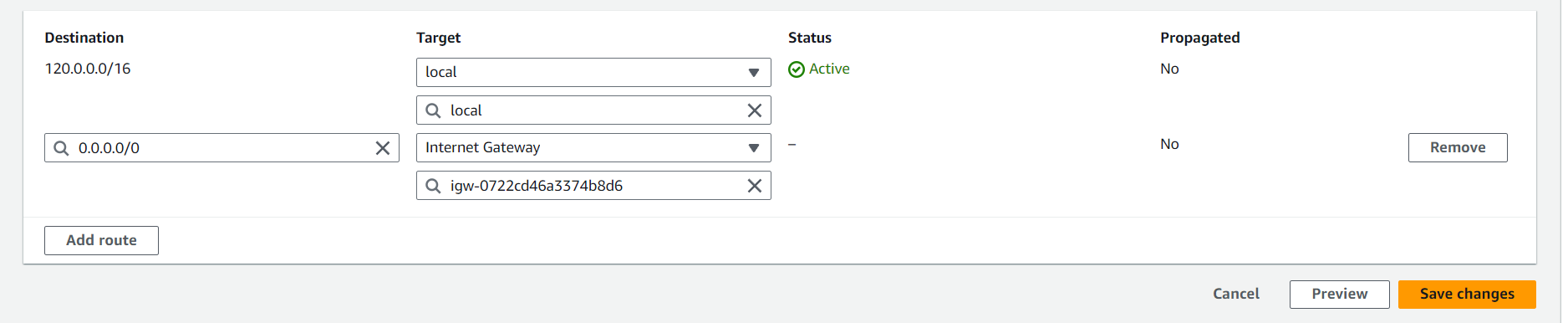
* **Step 1**: In the VPC Dashboard, click on **Route Tables** in the left navigation pane, then click **Create route table**.

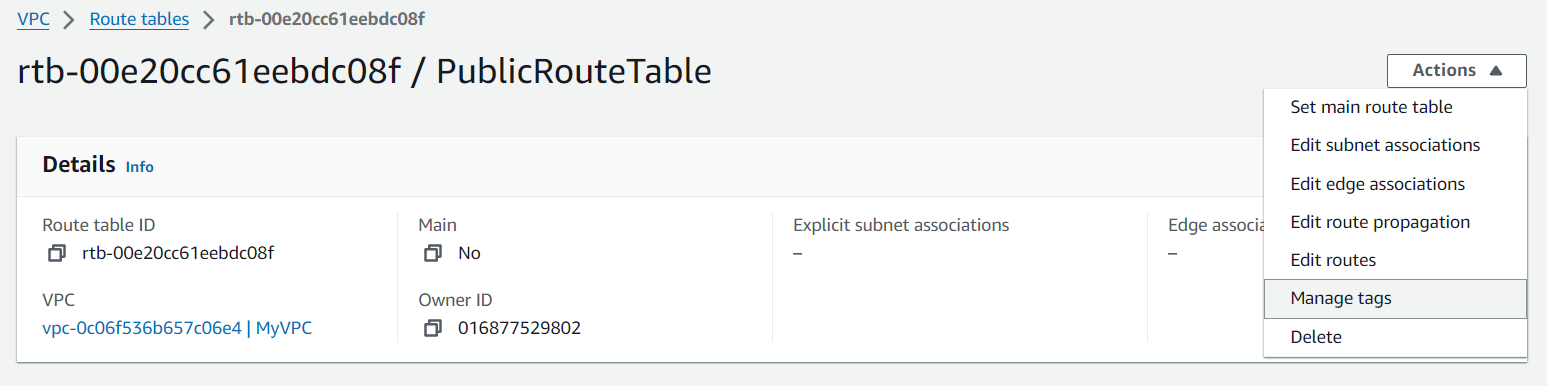


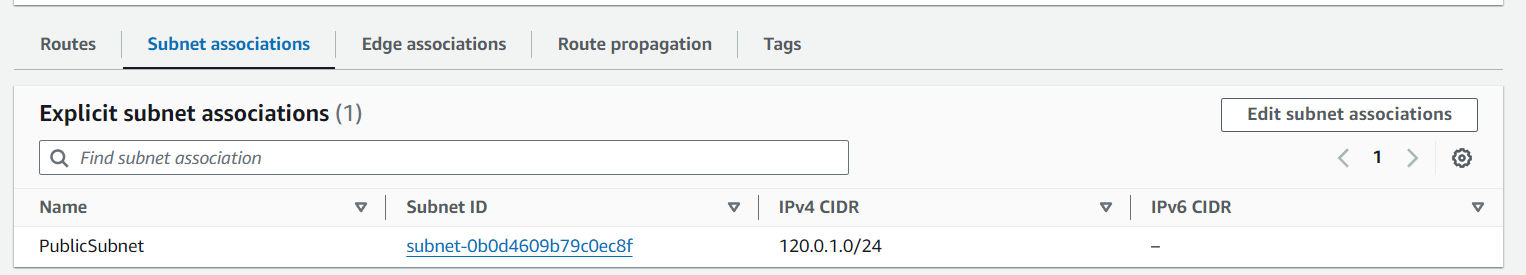
* **Step 2**: Select the VPC you created, and enter a name tag (e.g., PublicRouteTable).



* **Step 3**: Click **Create route table**.
* **Step 4**: Select the newly created route table, and under the **Routes** tab, click **Edit routes**.
* **Step 5**: Click **Add route**:
  + **Destination**: 0.0.0.0/0
  + **Target**: Select your Internet Gateway.
* **Step 6**: Click **Save changes**.

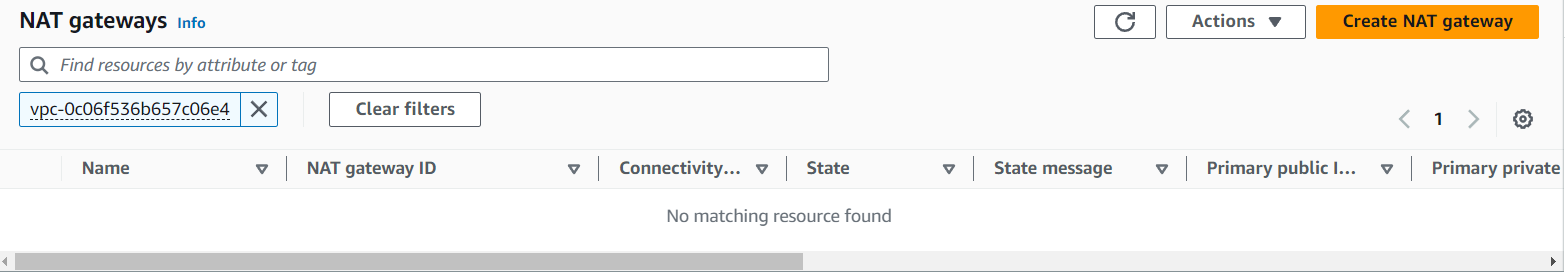


* **Step 7**: Under the **Subnets associations** tab, click **Edit subnet associations** 
* and select your public subnet.

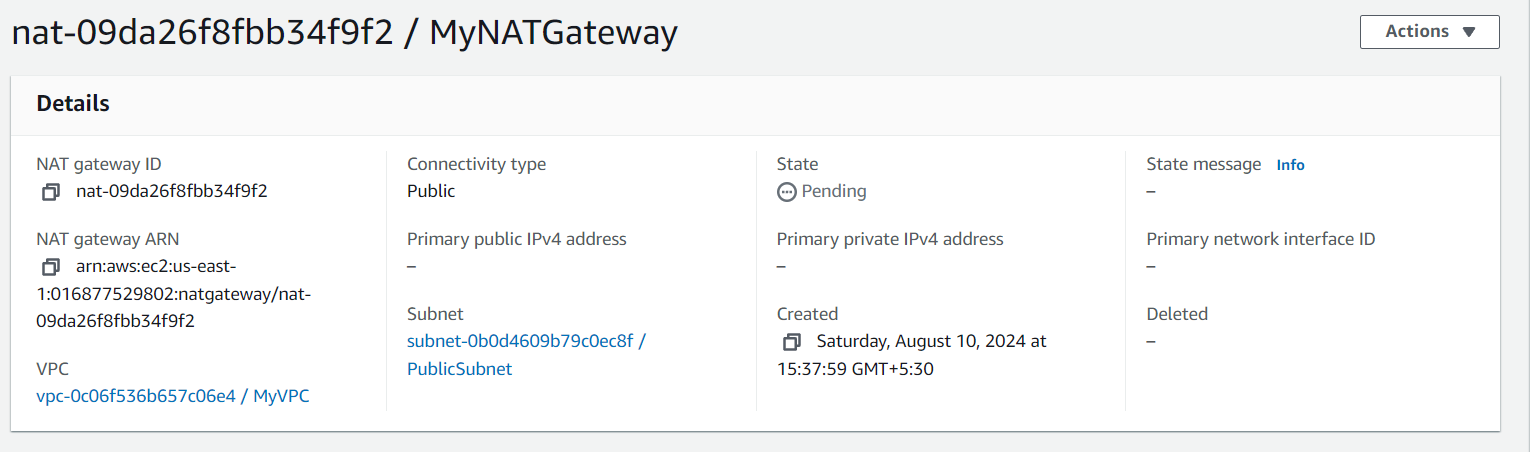


**5. Create a NAT Gateway**

* **Step 1**: In the VPC Dashboard, click on **NAT Gateways** in the left navigation pane, then click **Create NAT gateway**.

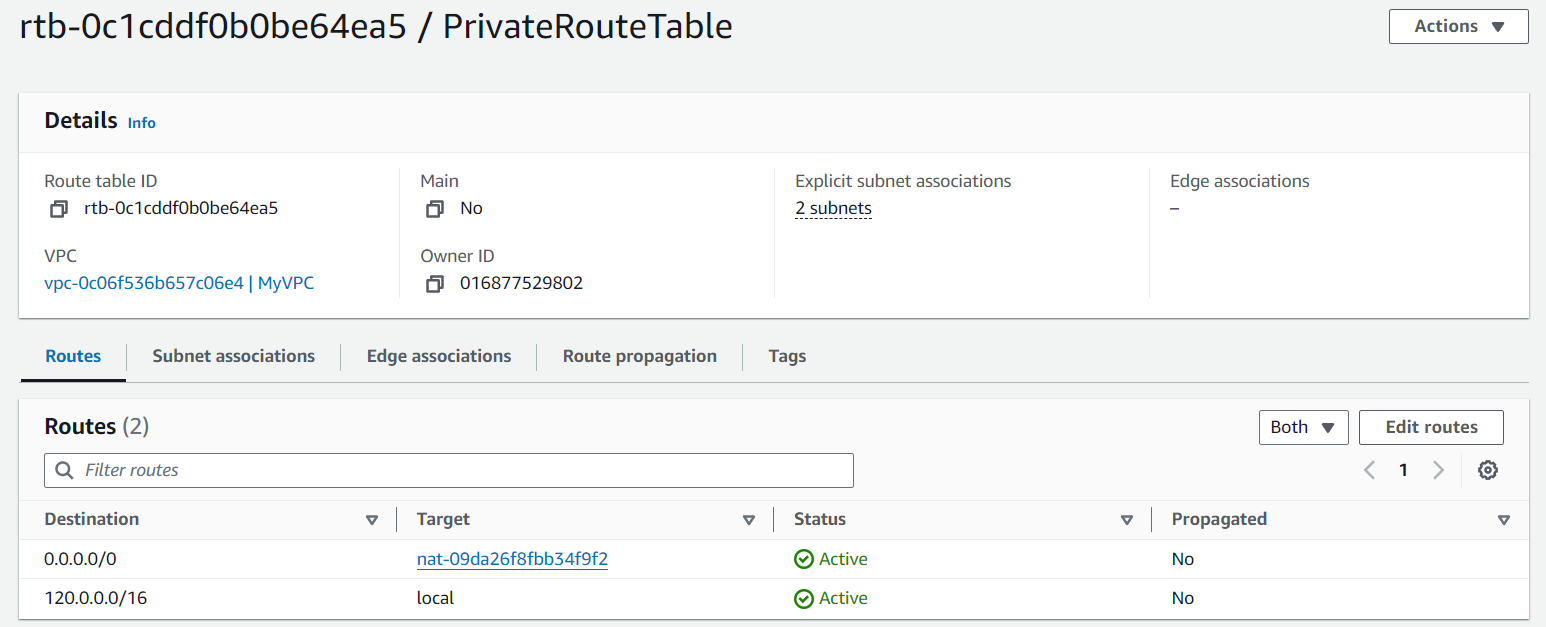


* **Step 2**: Enter the following details:
  + **Name tag**: (e.g., MyNATGateway)
  + **Subnet**: Select your public subnet.
  + **Elastic IP allocation ID**: Allocate a new Elastic IP or select an existing one.
* **Step 3**: Click **Create NAT gateway**.

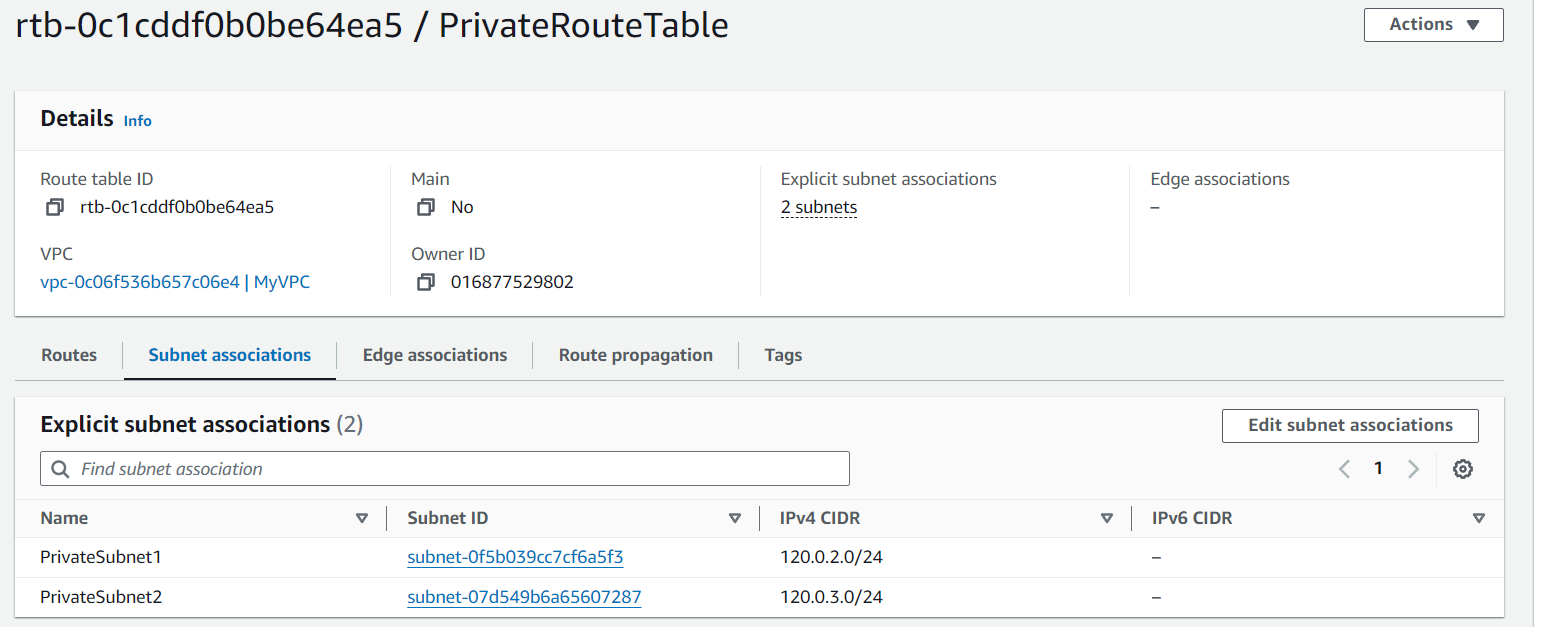


**6. Create a Route Table for the Private Subnets**

* **Step 1**: In the VPC Dashboard, click on **Route Tables** in the left navigation pane, then click **Create route table**.
* **Step 2**: Select the VPC you created, and enter a name tag (e.g., PrivateRouteTable).
* **Step 3**: Click **Create route table**.
* **Step 4**: Select the newly created route table, and under the **Routes** tab, click **Edit routes**.
* **Step 5**: Click **Add route**:
  + **Destination**: 0.0.0.0/0
  + **Target**: Select your NAT Gateway.
* **Step 6**: Click **Save changes**.

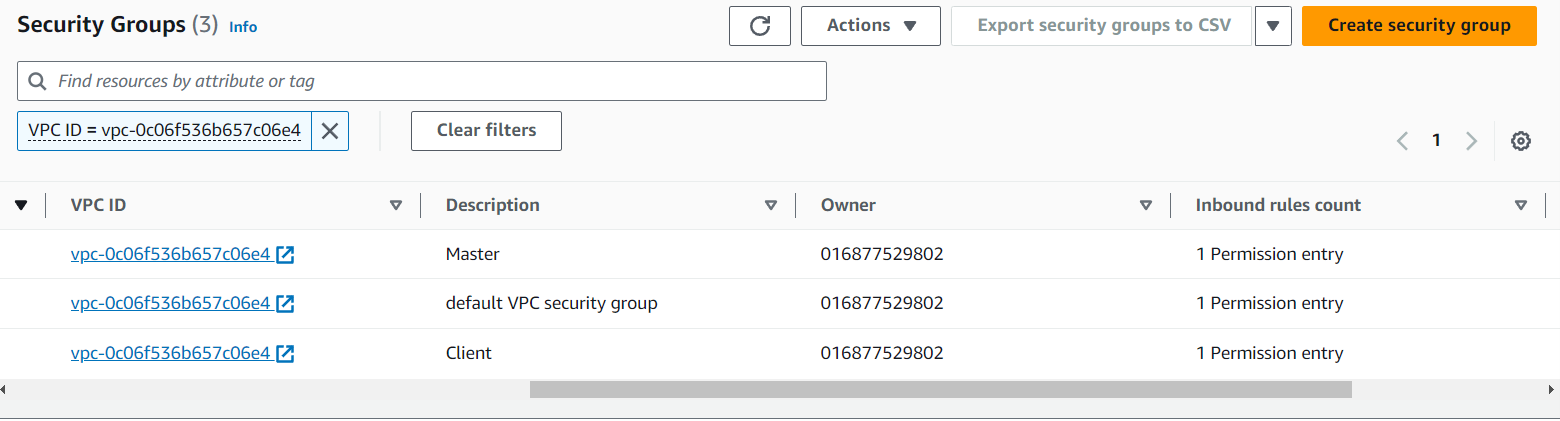


* **Step 7**: Under the **Subnets associations** tab, click **Edit subnet associations** and select your private subnets.

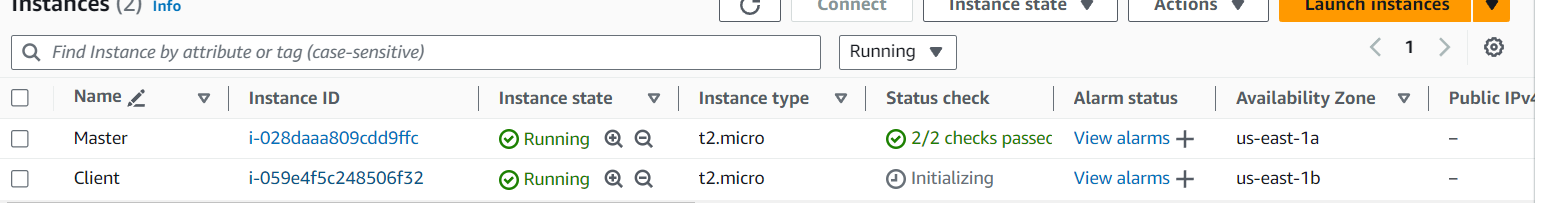


**1. Launch EC2 Instances**

* **Step 1**: Log in to the AWS Management Console and go to the **EC2 Dashboard**.
* **Step 2**: Click on **Launch Instance**.
* **Step 3**: Choose an Amazon Machine Image (AMI). For simplicity, use **Amazon Linux 2**.
* **Step 4**: Choose an Instance Type (e.g., t2.micro).
* **Step 5**: Configure Instance Details:
  + **Network**: Select the VPC where you want to create the instances (e.g., MYVPC1).
  + **Subnet**: Select a public subnet(for Master) and private subnet(for client) within the chosen VPC.
* **Step 6**: Add Storage if needed.
* **Step 7**: Add Tags:
  + **Key**: Name
  + **Value**: Master (for the first instance) and Client (for the second instance)
* **Step 8**: Configure Security Group:
  + For the **Master** instance, create a security group (e.g., MasterSG) that allows:
    - **SSH** access from anywhere (0.0.0.0/0).
  + For the **Client** instance, create a security group (e.g., ClientSG) that:
    - **Does not** allow direct SSH access from anywhere.

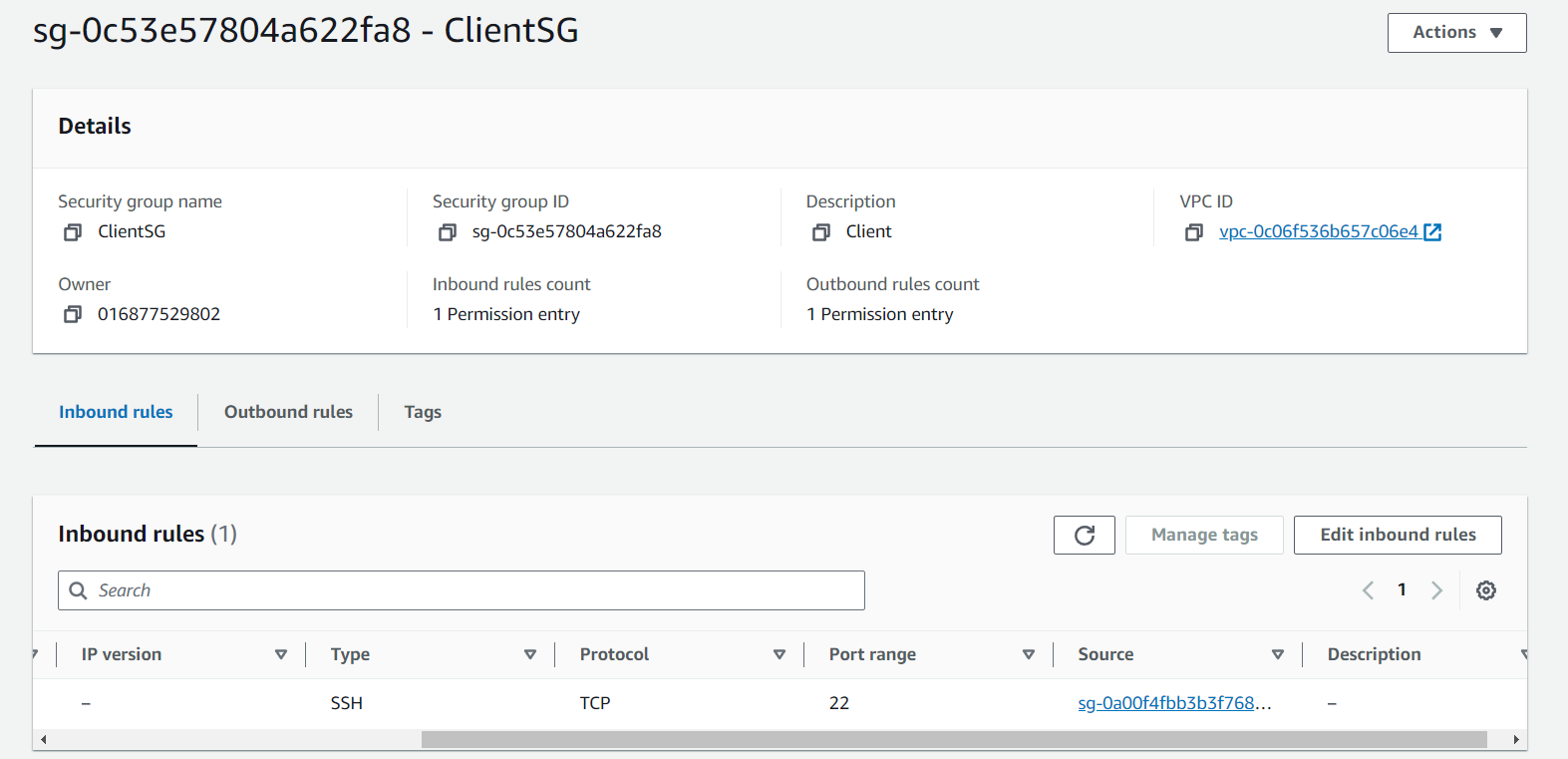


* **Step 9**: Review and Launch the instances.



**2. Configure Security Groups**

* **Step 1**: Go to the **Security Groups** section in the EC2 Dashboard.
* **Step 2**: Select the ClientSG security group.
* **Step 3**: Edit the **Inbound Rules**:
  + Add a new rule to allow **SSH** access:
    - **Type**: SSH
    - **Protocol**: TCP
    - **Port Range**: 22
    - **Source**: Select the MasterSG security group (This restricts SSH access to the Client instance only from the Master instance).



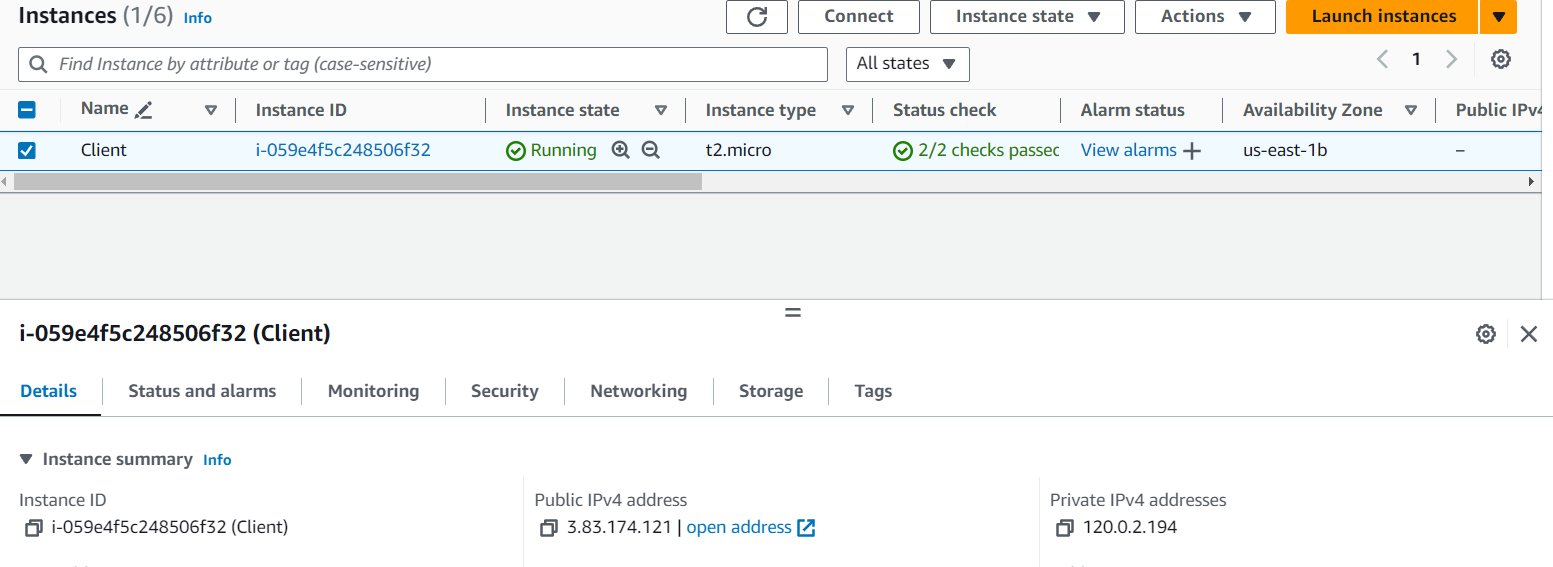
* **Step 4**: Save the changes.

**3. Test the Configuration**

* **Step 1**: SSH into the **Master** instance from your local machine using the command:

ssh -i "new.pem" ec2-user@120.0.1.25

* **Step 2**: From the **Master** instance, SSH into the **Client** instance using the private IP address of the Client instance:



ssh -i "new.pem" ec2-user@3.83.174.121

* **Step 3**: Verify that the Client instance is not directly accessible via SSH from your local machine, only through the Master instance.

