**VPC Task -02**

**1) Create one VPC, with 1 one public subnet and 1 private subnet.**

**Step 1: Create a VPC**

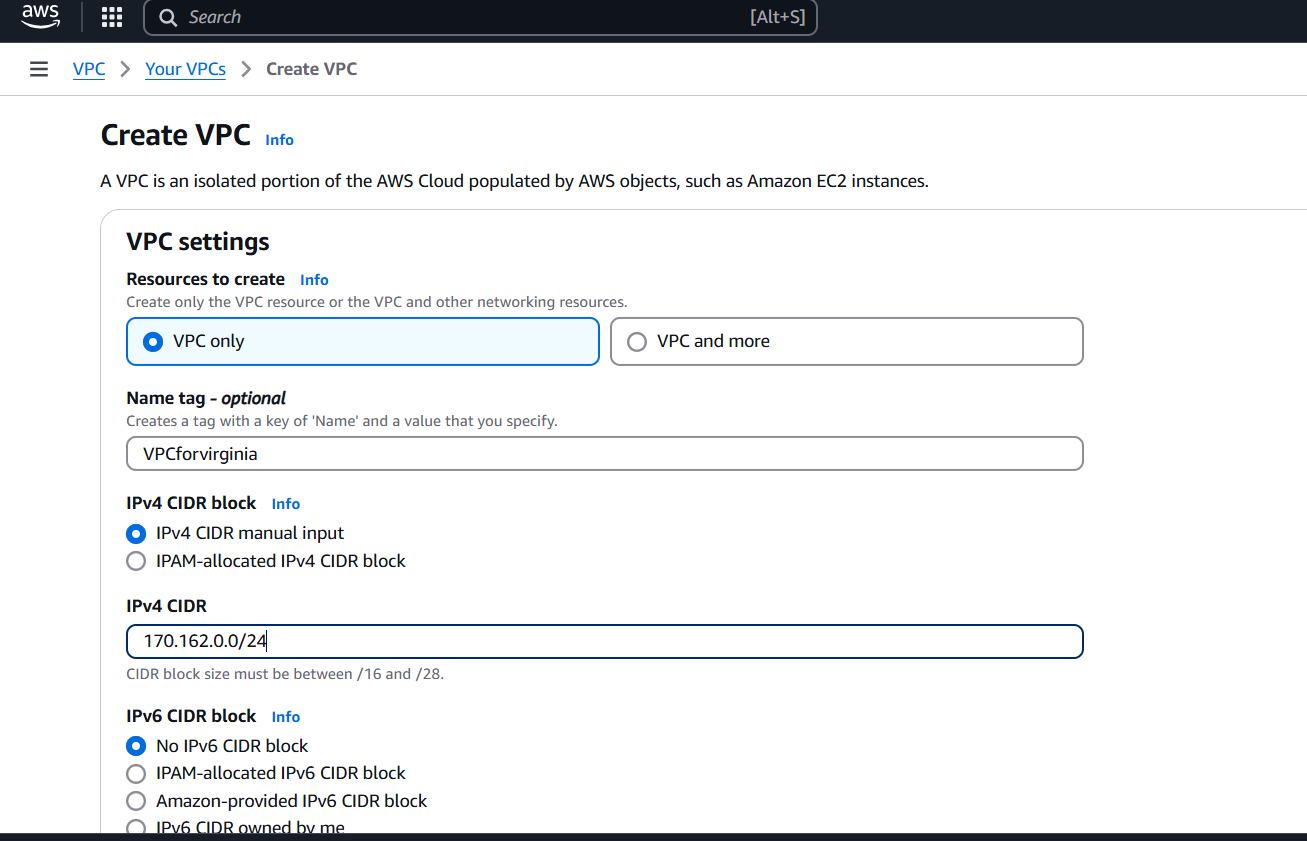
1. Go to the VPC console.

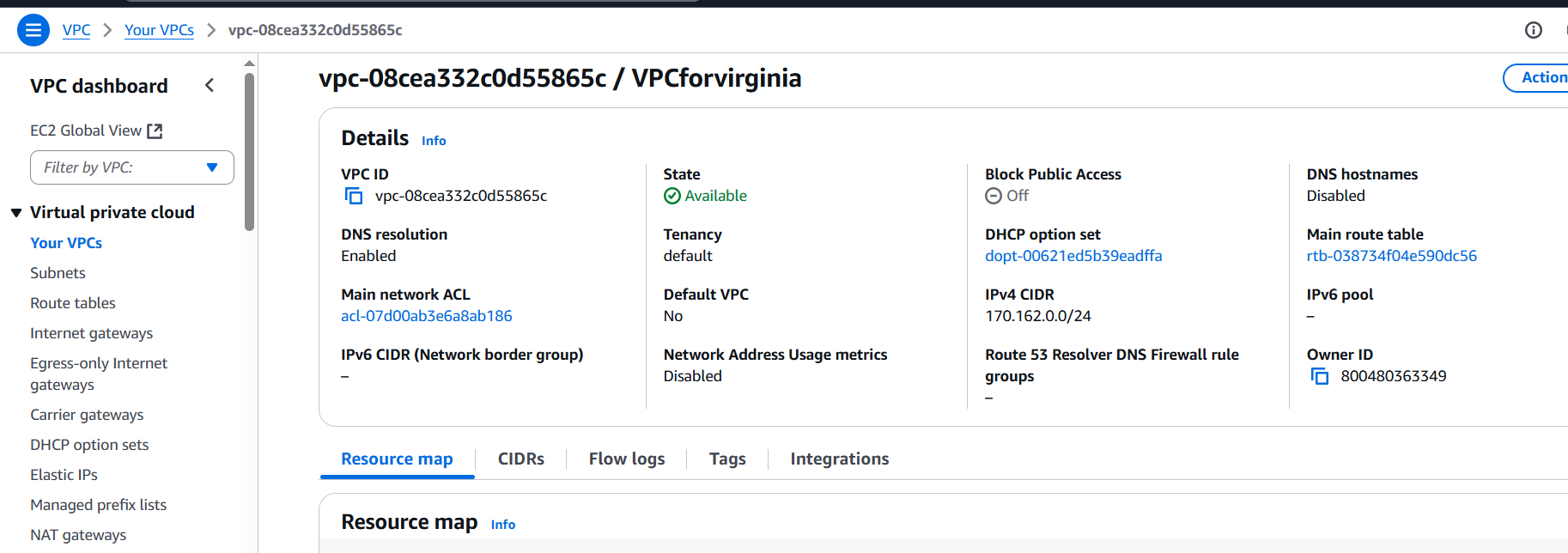
2. Click on "Create VPC".

3. Enter a name for your VPC.

4. Enter an IPv4 CIDR block.

5. Click "Create VPC".





**Step 2: Create a Public Subnet**

1. Go to the VPC console.

2. Select the VPC.

3. Click on "Subnets" in the left-hand menu.

4. Click on "Create subnet".

5. Enter a name for your public subnet.

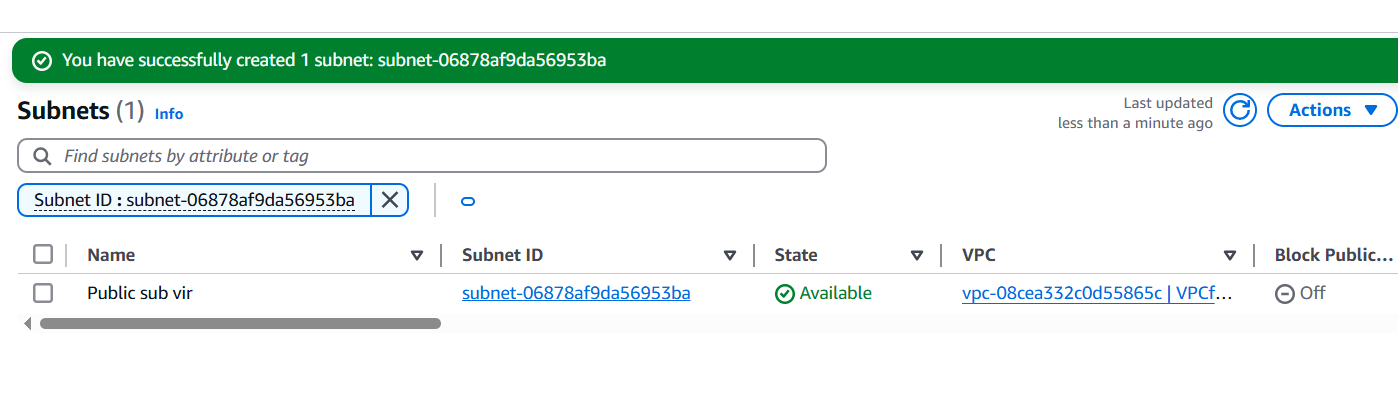
6. Select the VPC.

7. Enter an IPv4 CIDR block.

8. Enable auto-assign public IPv4 address.

9. Click "Create subnet".





**Step 3: Create an Internet Gateway**

1. Go to the VPC console.

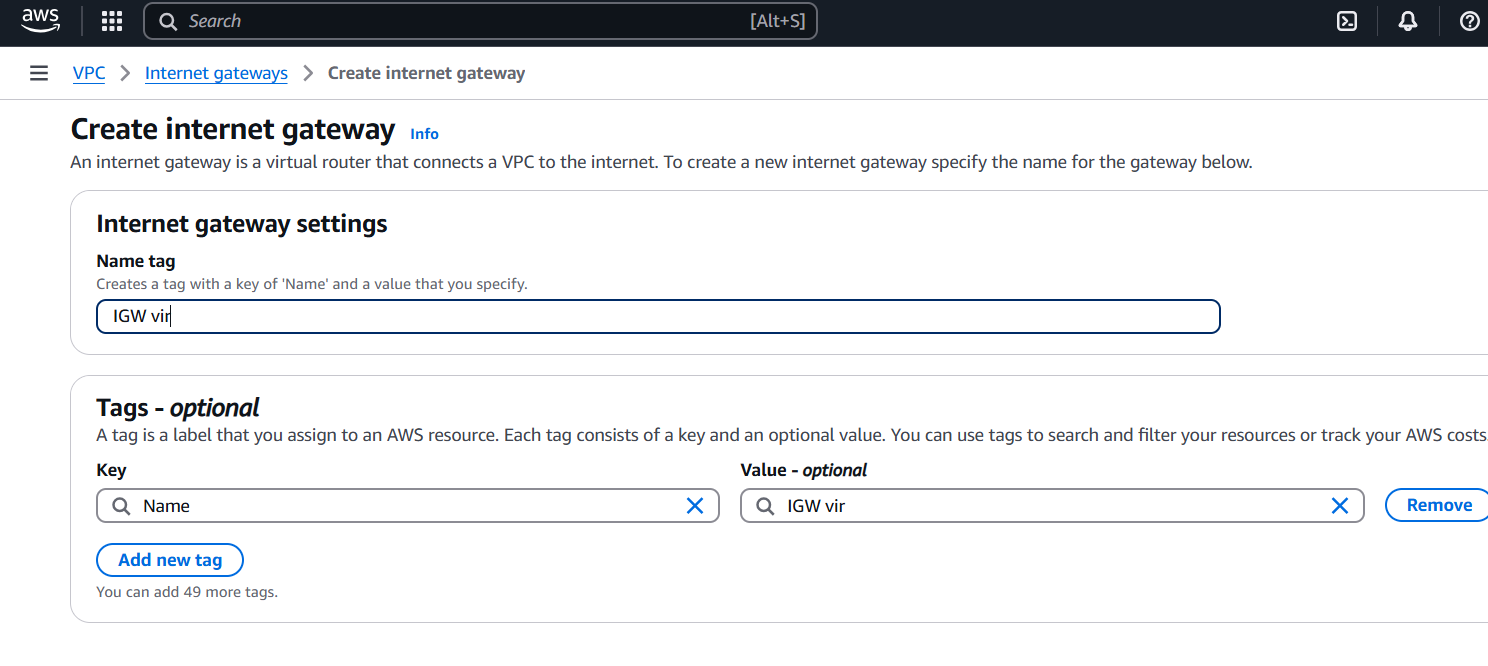
2. Click on "Internet Gateways" in the left-hand menu.

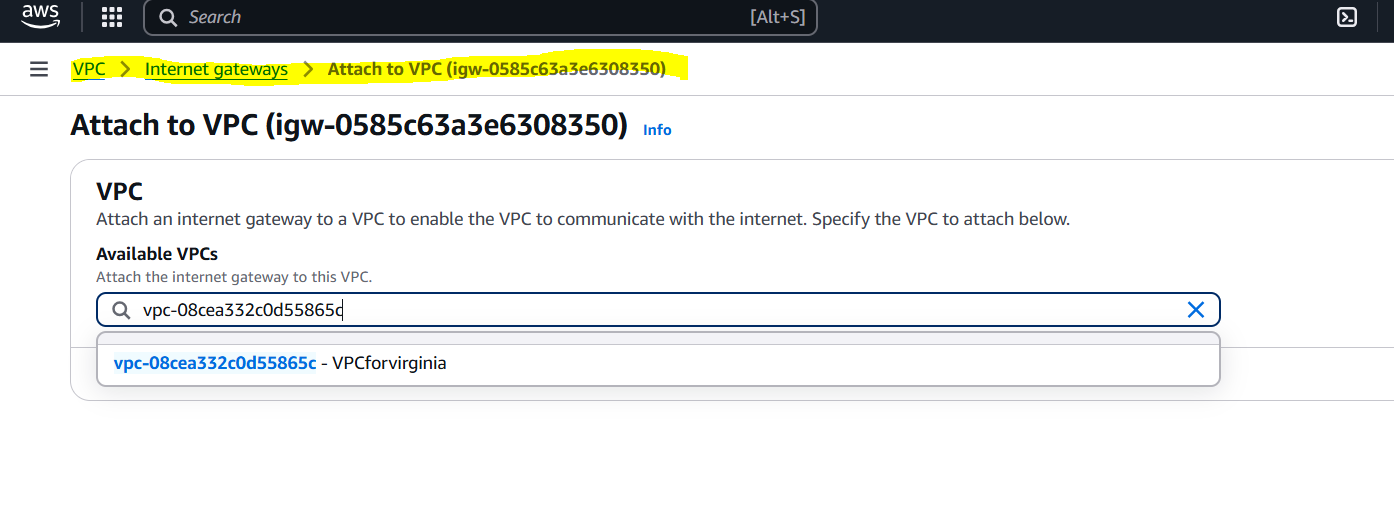
3. Click on "Create internet gateway".

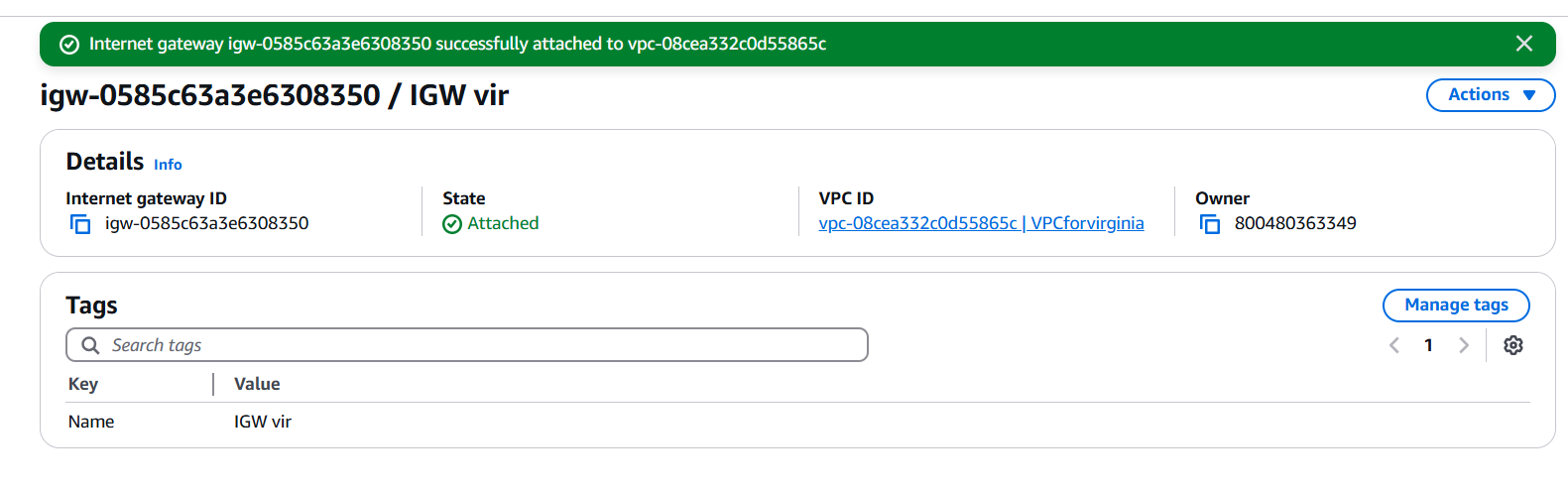
4. Enter a name for your internet gateway.

5. Click "Create internet gateway".

6. Attach the internet gateway to your VPC.







**Step 4: Update Route Table for Public Subnet**

1. Go to the VPC console.

2. Select the VPC.

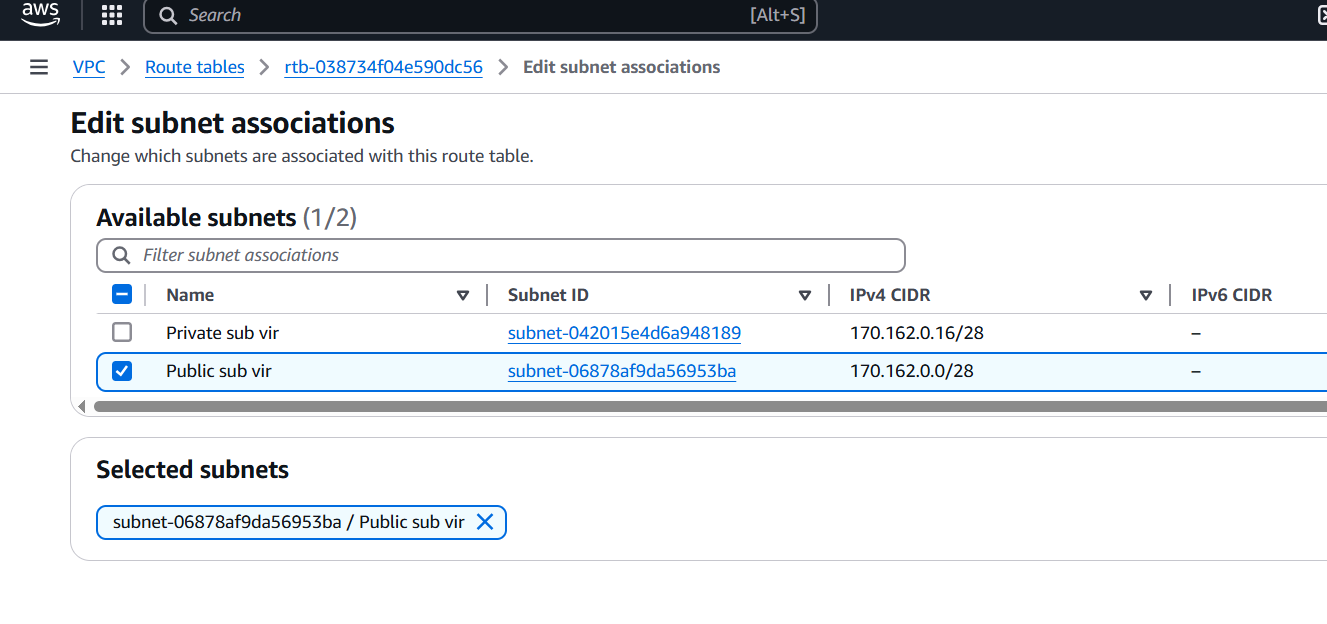
3. Click on "Route Tables" in the left-hand menu.

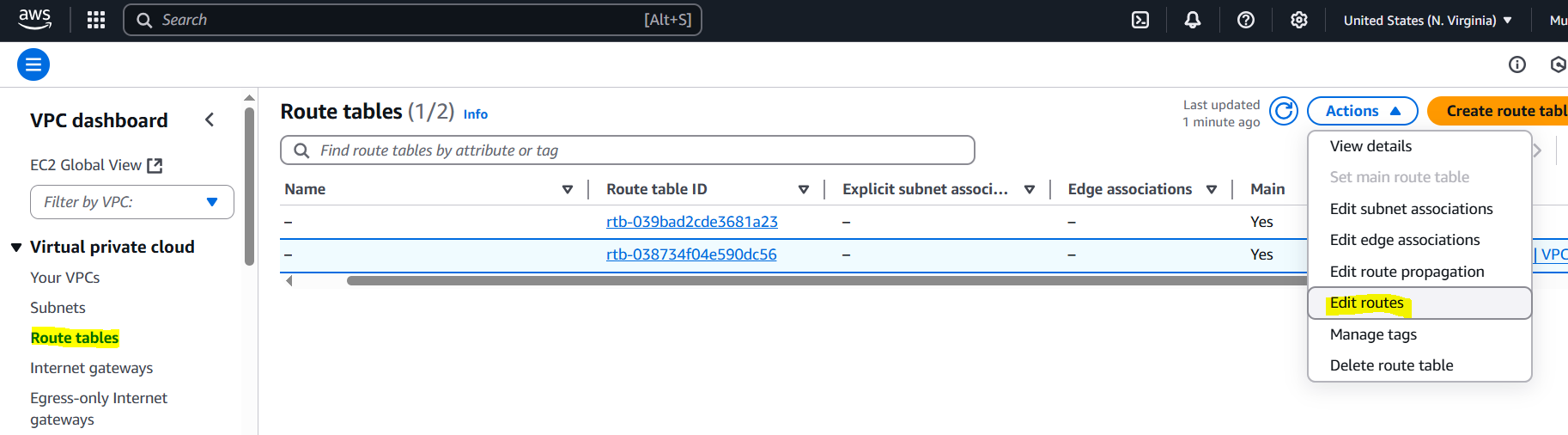
4. Select the route table associated with your public subnet.

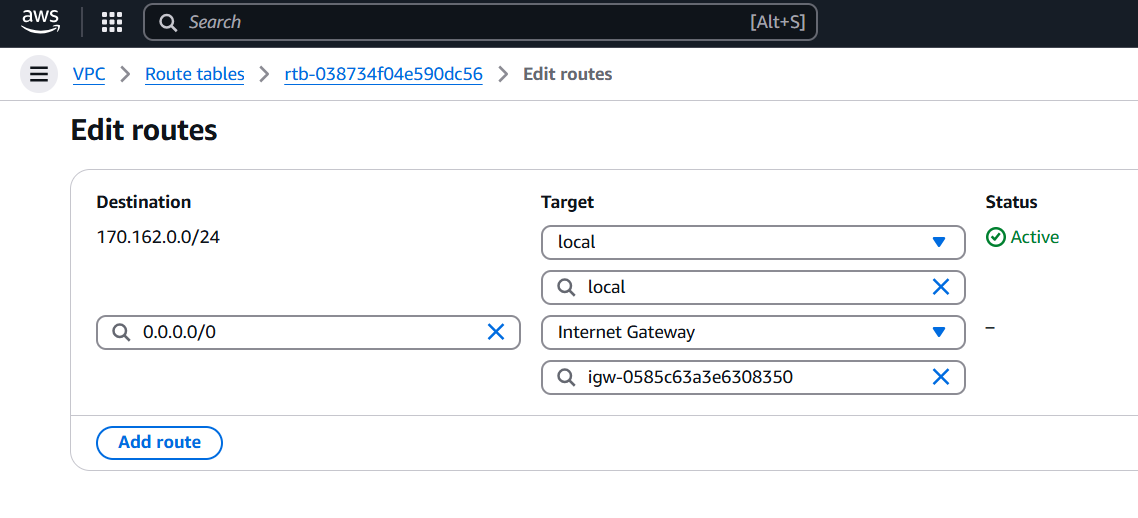
5. Click on "Actions" and then select "Edit route table".

6. Add a route with destination 0.0.0.0/0 and target as the internet gateway.

7. Click "Save changes".







**Step 5: Create a Private Subnet**

1. Go to the VPC console.

2. Select the VPC.

3. Click on "Subnets" in the left-hand menu.

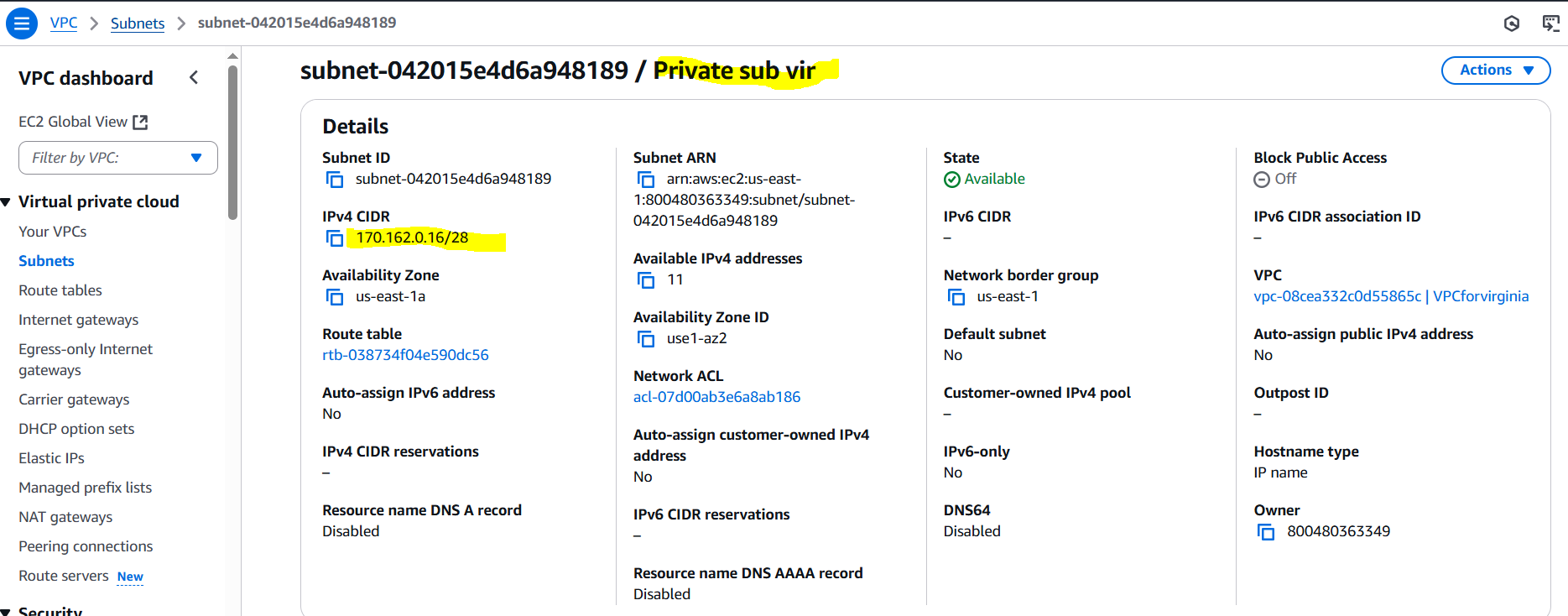
4. Click on "Create subnet".

5. Enter a name for your private subnet.

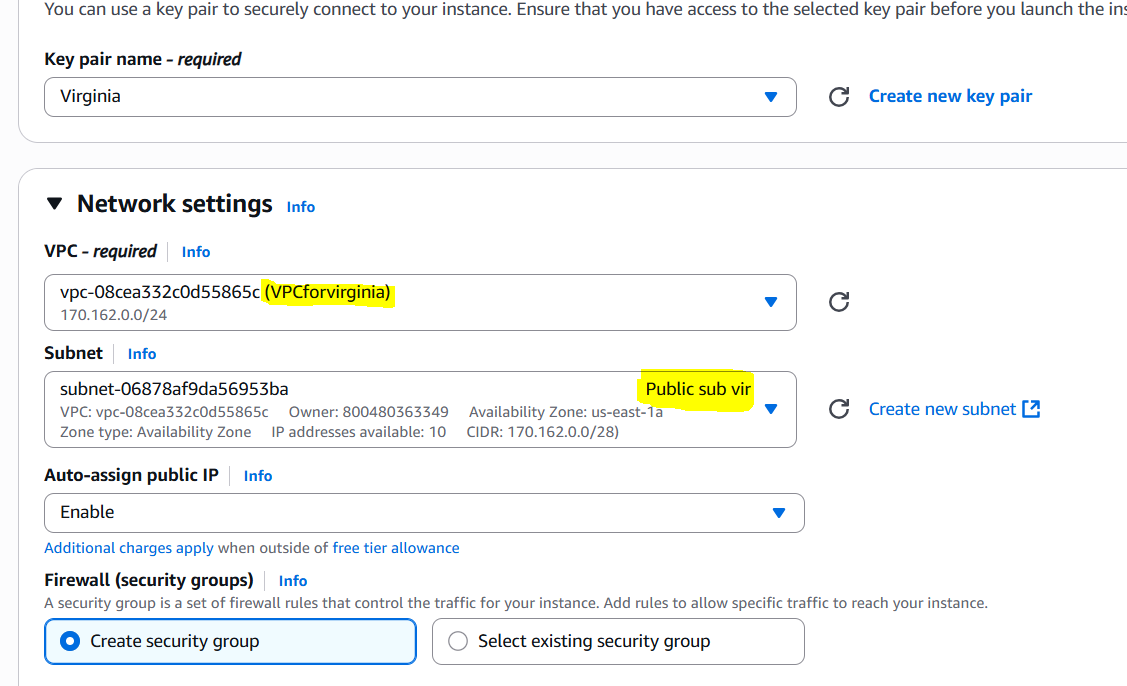
6. Select the VPC.

7. Enter an IPv4 CIDR block.

8. Click "Create subnet".



Launched one Public instance.



**2) Enable VPC peering for cross region.**

**Step 1: Create a VPC Peering Connection**

1. Go to the VPC console.

2. Click on "Peering Connections" in the left-hand menu.

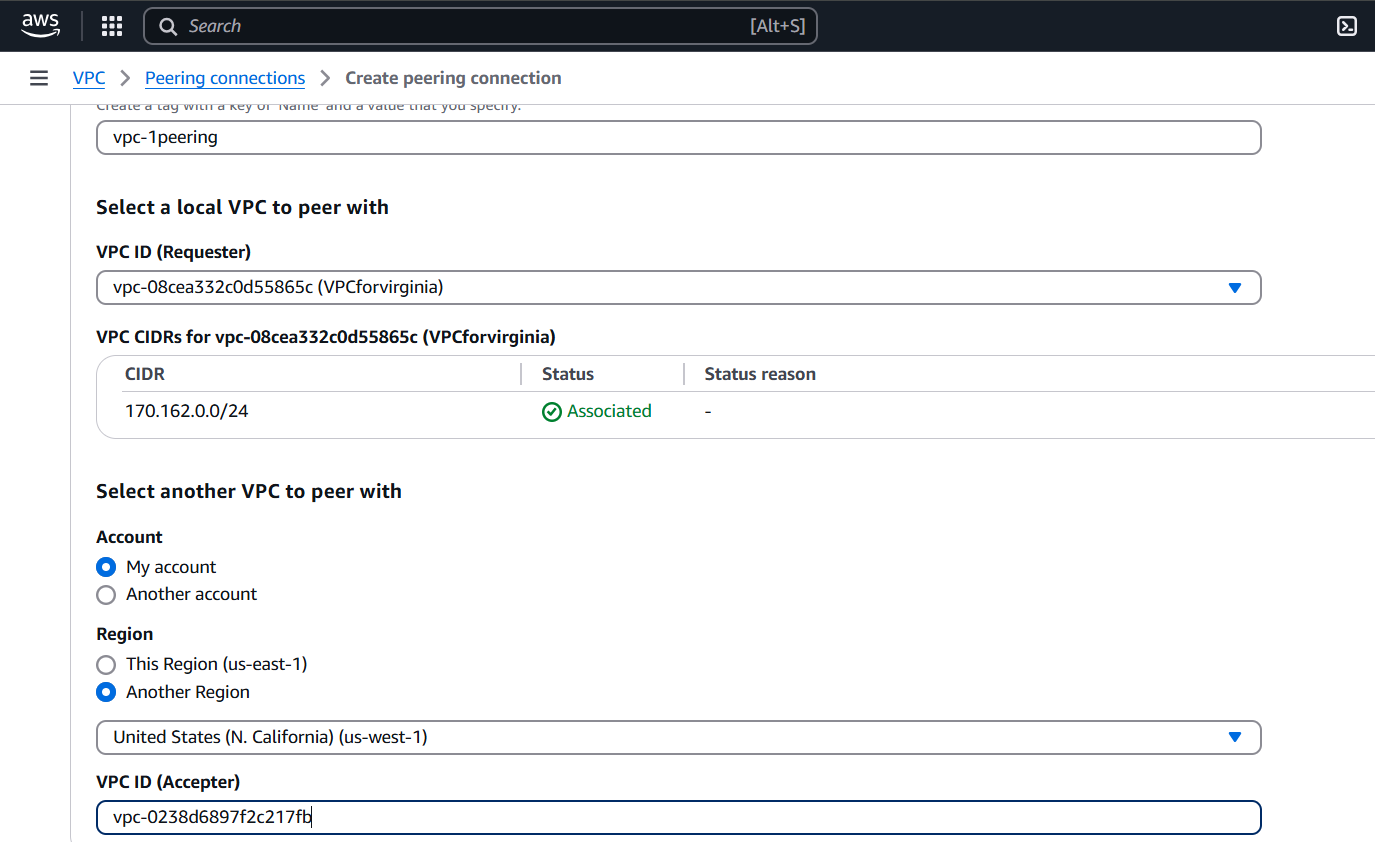
3. Click on "Create peering connection".

4. Select "Another AWS account" or "My account" depending on your use case.

5. Choose the VPC in the current region (Requester VPC).

6. Enter the VPC ID and region of the VPC you want to peer with (Accepter VPC).

7. Click "Create peering connection".



**Step 2: Accept the VPC Peering Connection**

1. Log in to the AWS account that owns the Accepter VPC.

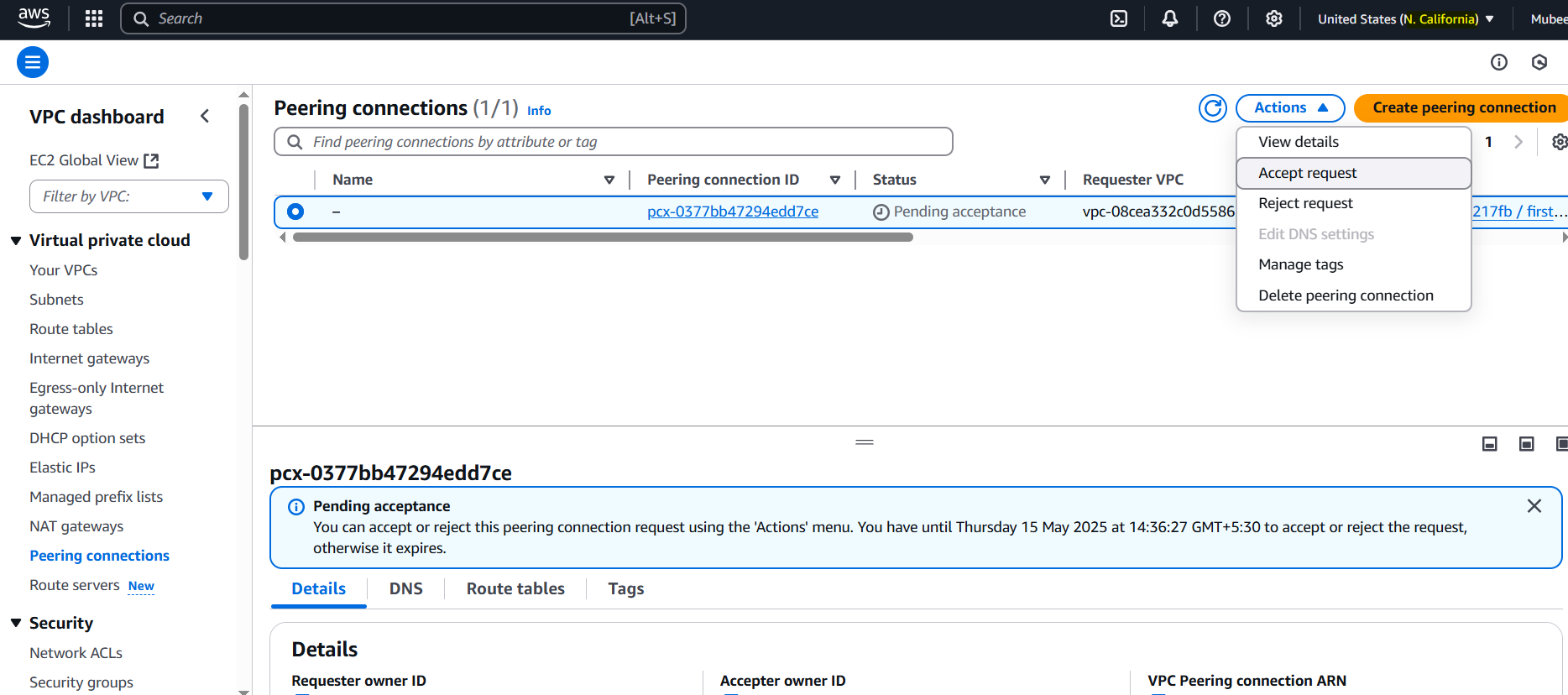
2. Go to the VPC console in the region of the Accepter VPC.

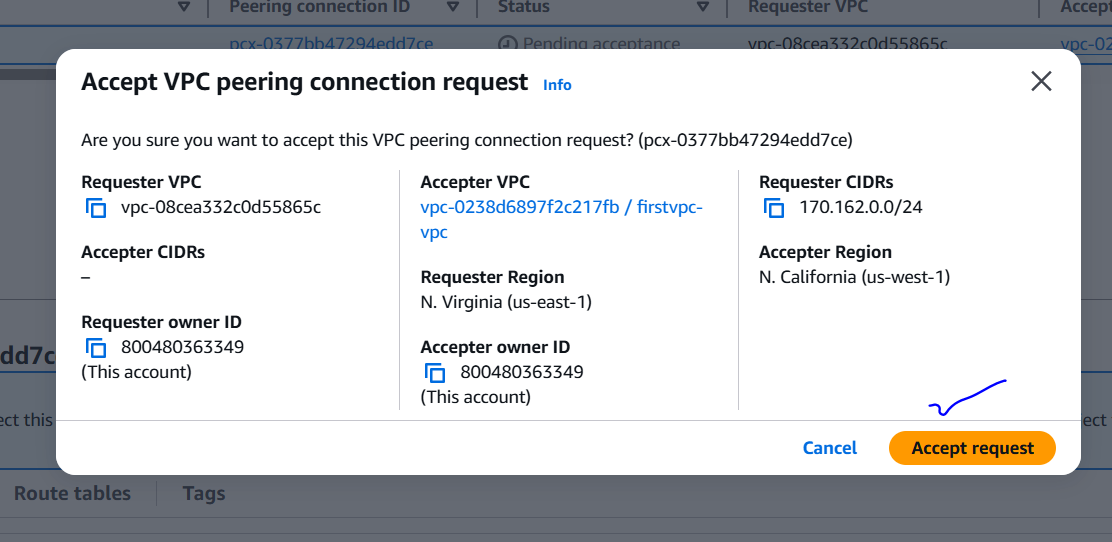
3. Click on "Peering Connections" in the left-hand menu.

4. Find the pending peering connection and select it.

5. Click on "Actions" and then select "Accept request".

6. Confirm that you want to accept the peering connection.

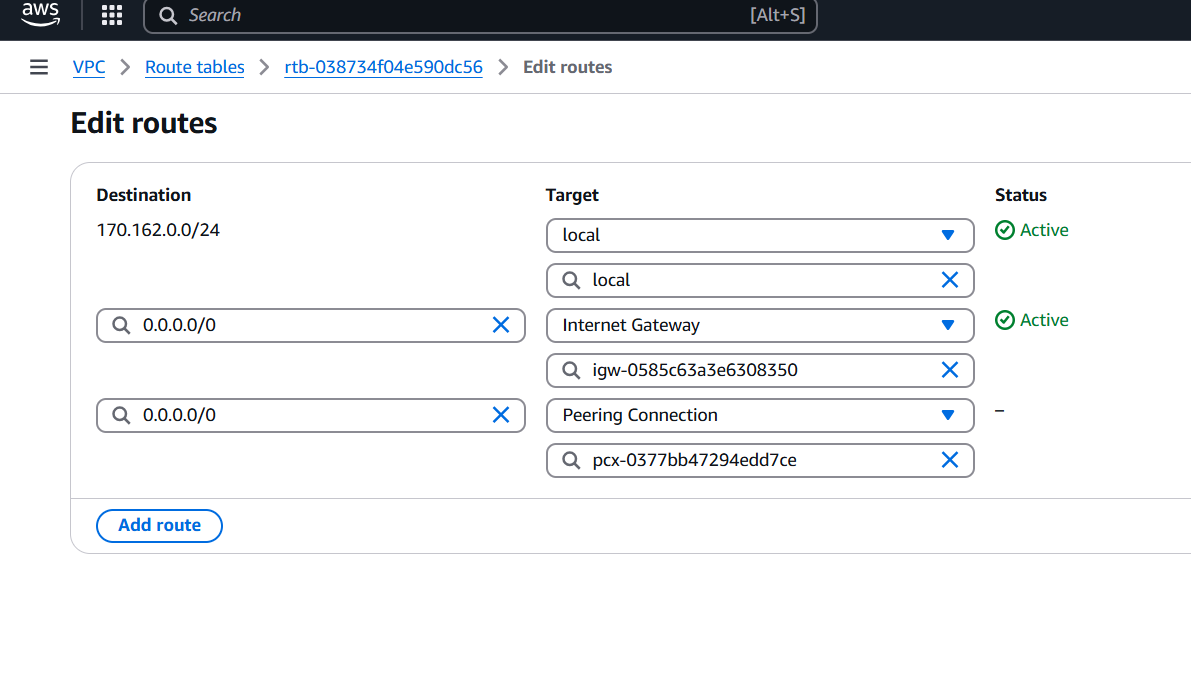


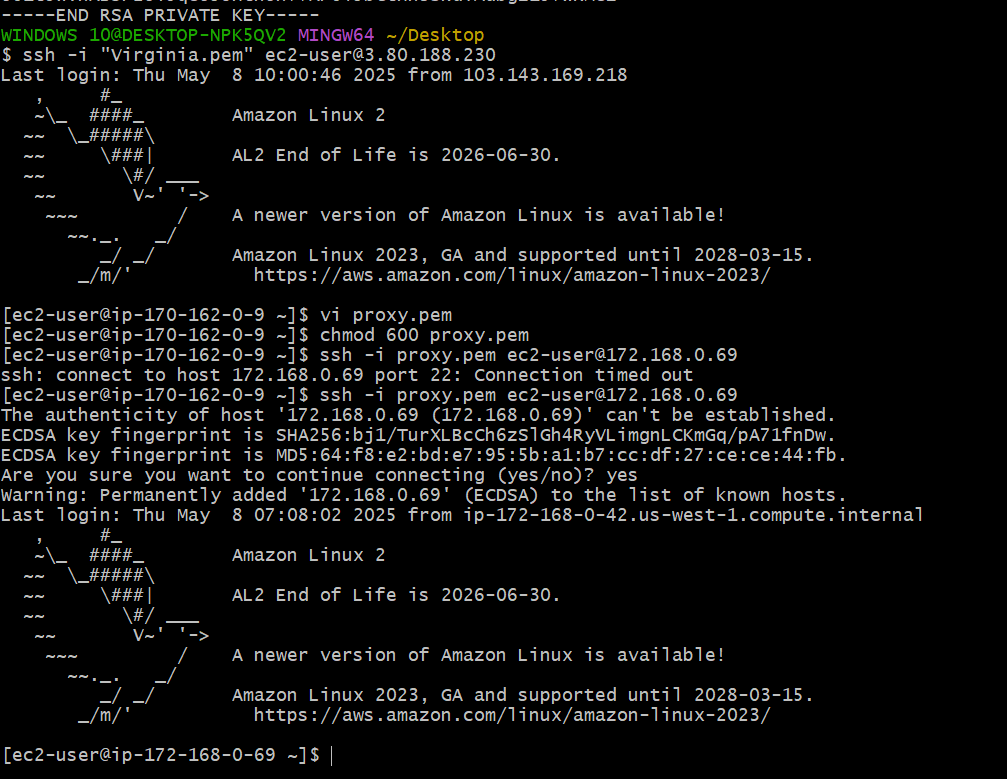


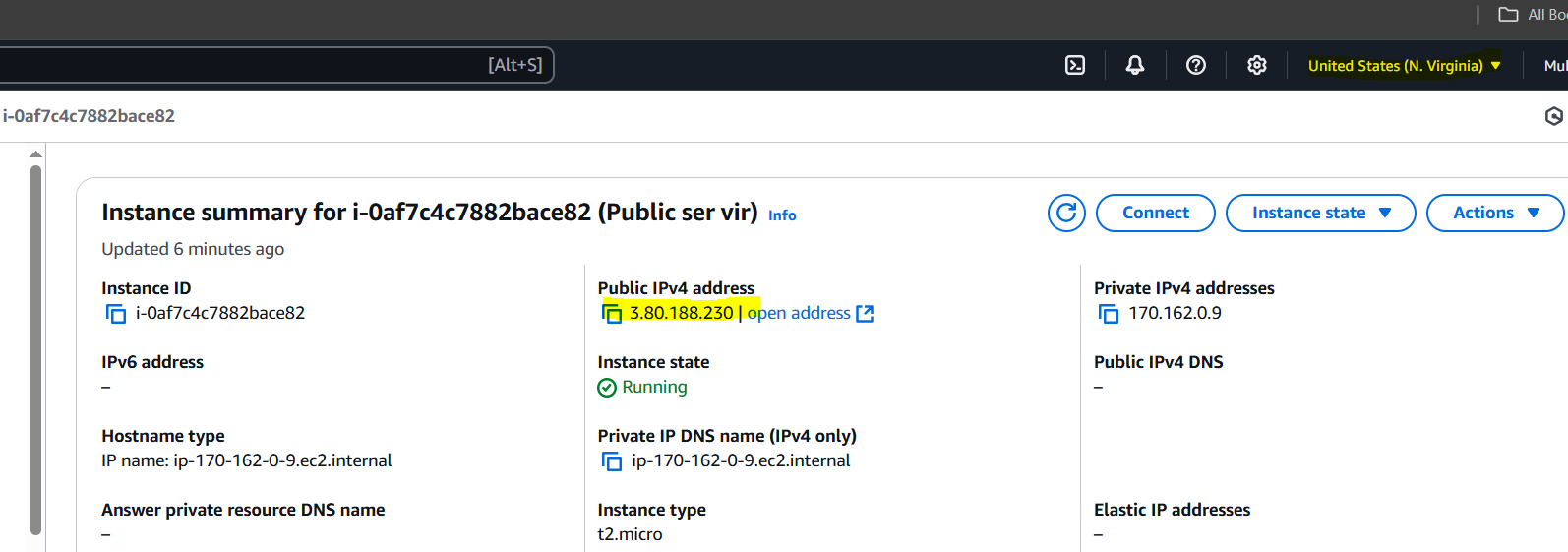
**Step 3: Update Route Tables**

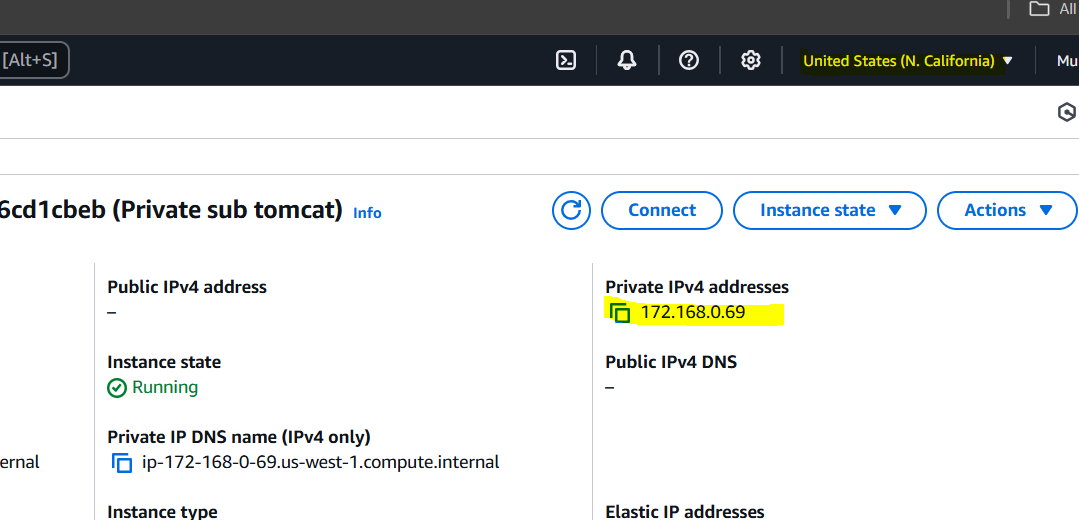
1. Update the route tables in both VPCs to include routes for the peering connection.

2. Add a route for the CIDR block of the peer VPC.









**3) Enable VPC peering for cross account. (You can collaborate with your friend and do this task).**

**Step 1: Create a VPC Peering Connection (Requester Account)**

1. Go to the VPC console in the requester account.

2. Click on "Peering Connections" and then "Create peering connection".

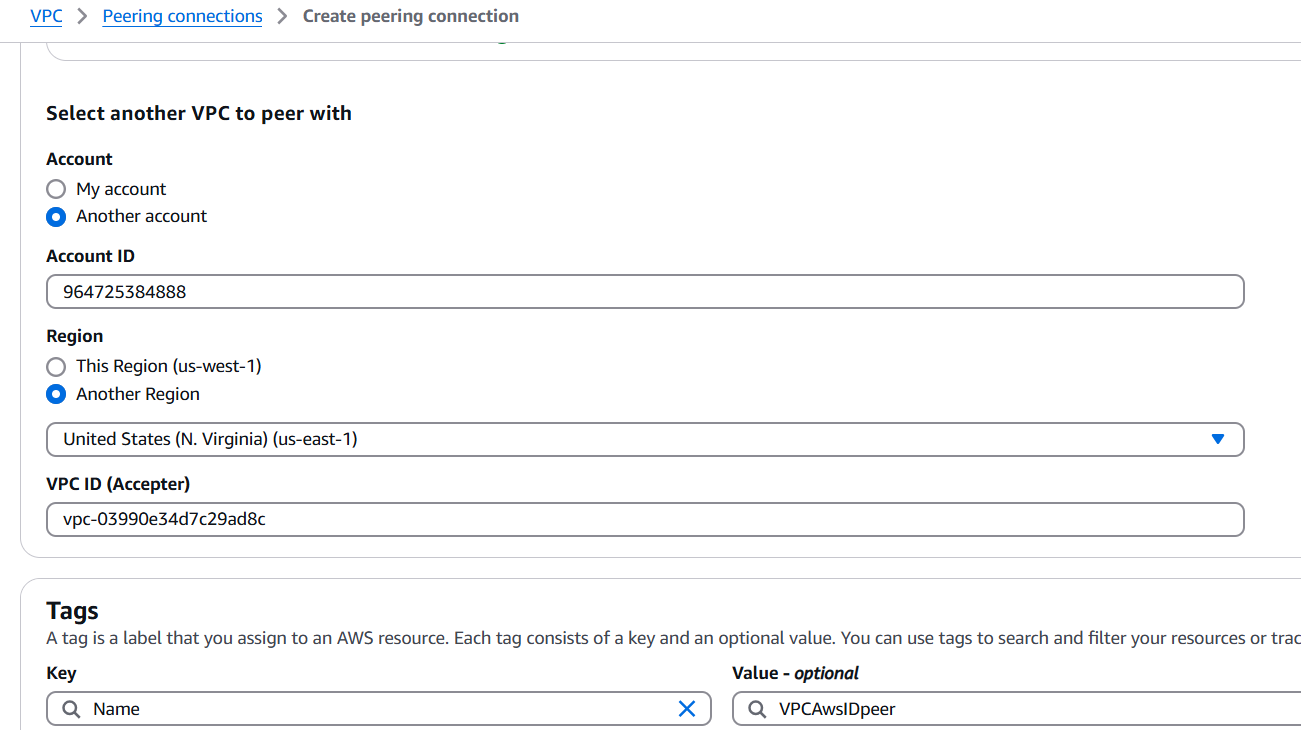
3. Select "Another AWS account".

4. Enter the AWS account ID and VPC ID of the accepter account.

5. Choose the VPC in the requester account.

6. Click "Create peering connection".





**Step 2: Accept the VPC Peering Connection (Accepter Account)**

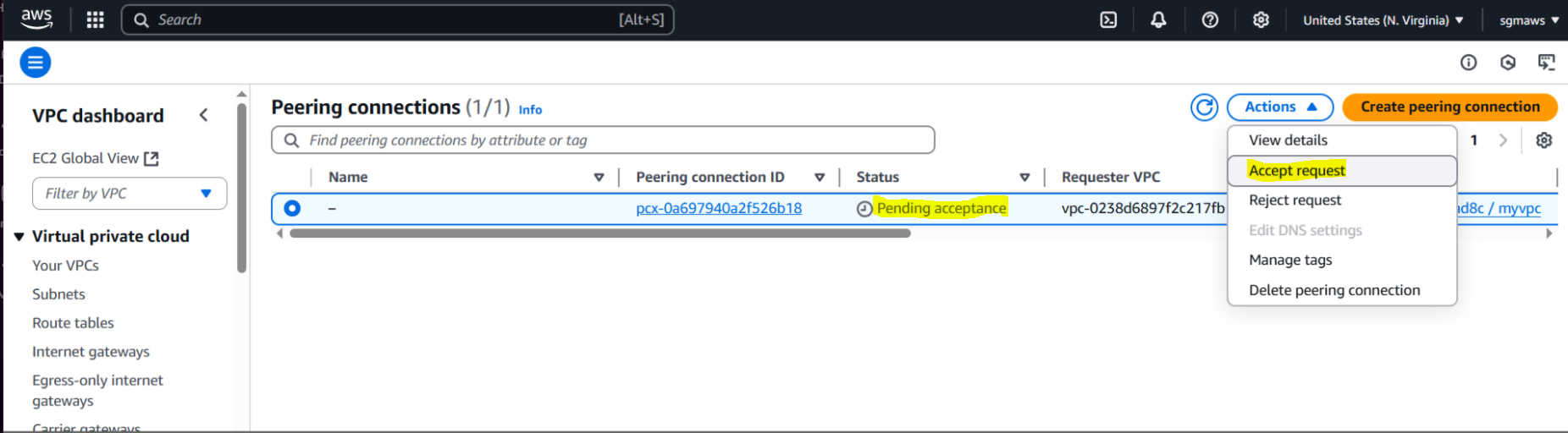
1. Log in to the AWS account that owns the accepter VPC.

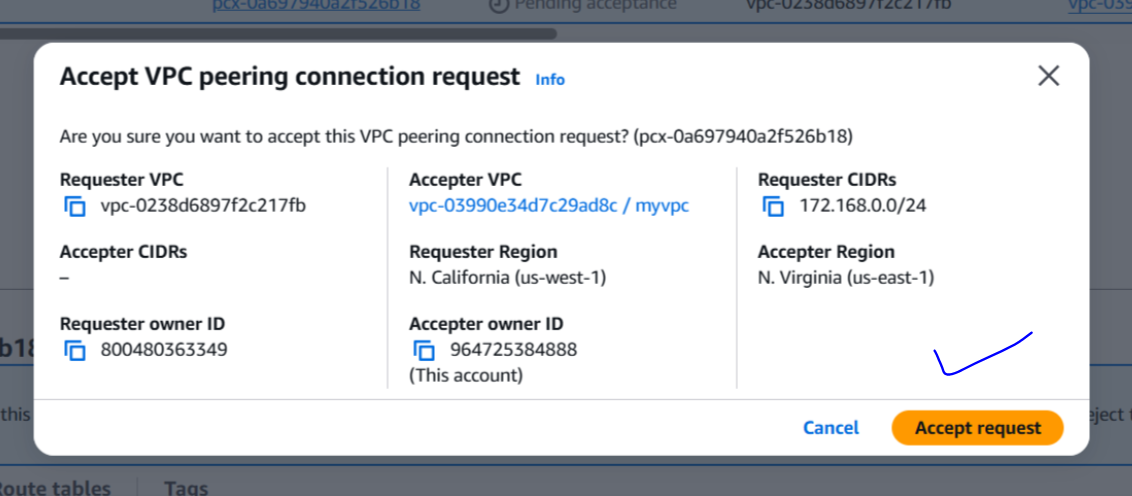
2. Go to the VPC console.

3. Click on "Peering Connections".

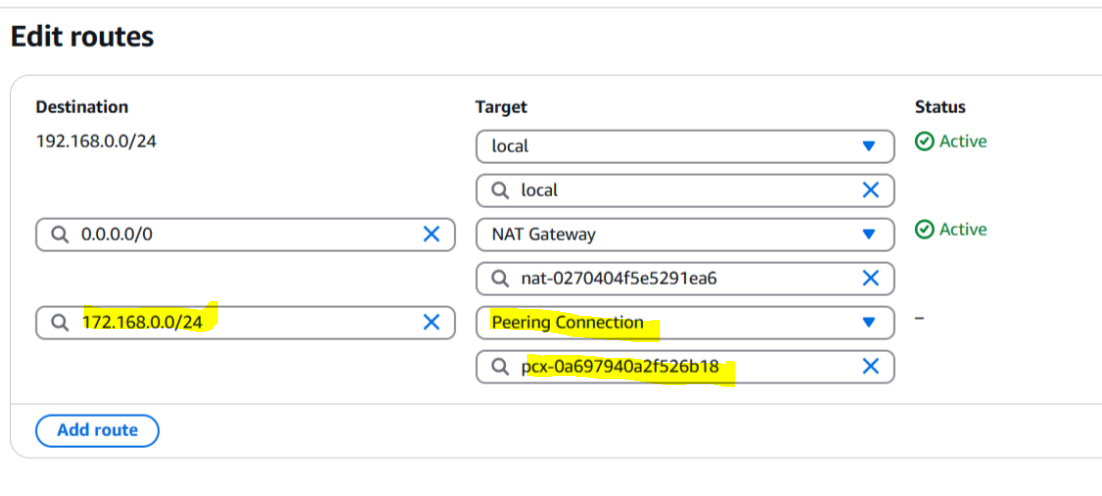
4. Find the pending peering connection and select it.

5. Click on "Actions" and then select "Accept request".

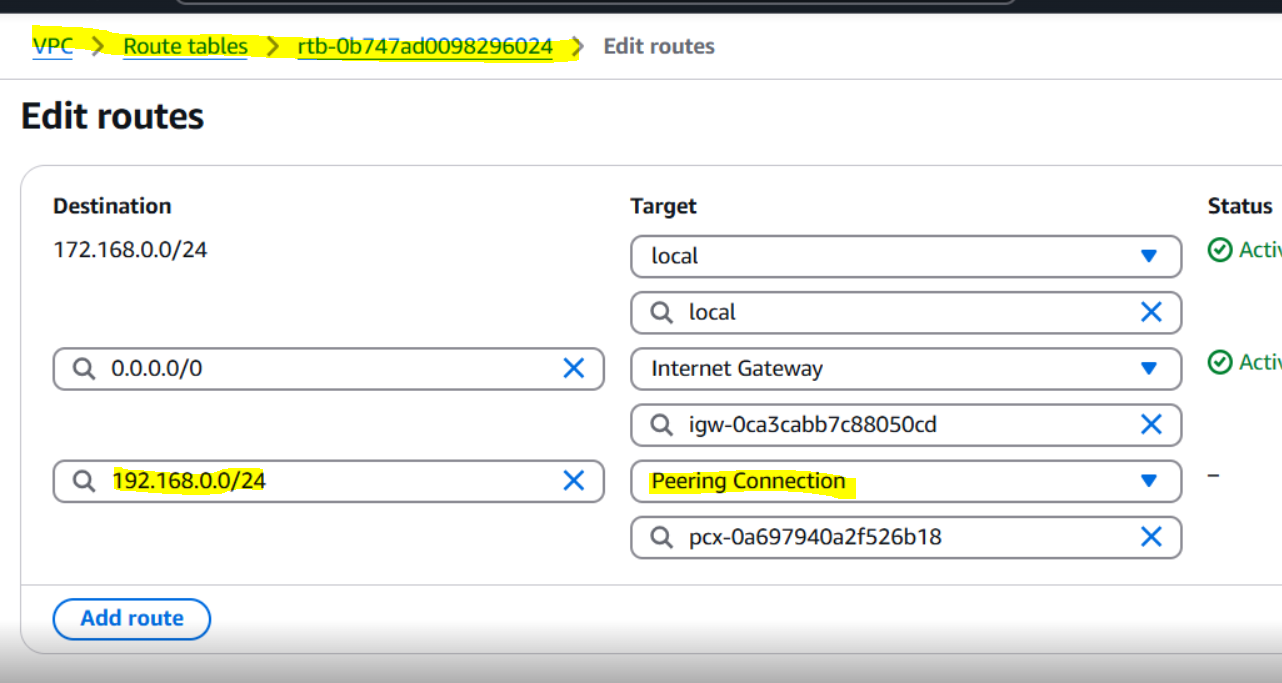




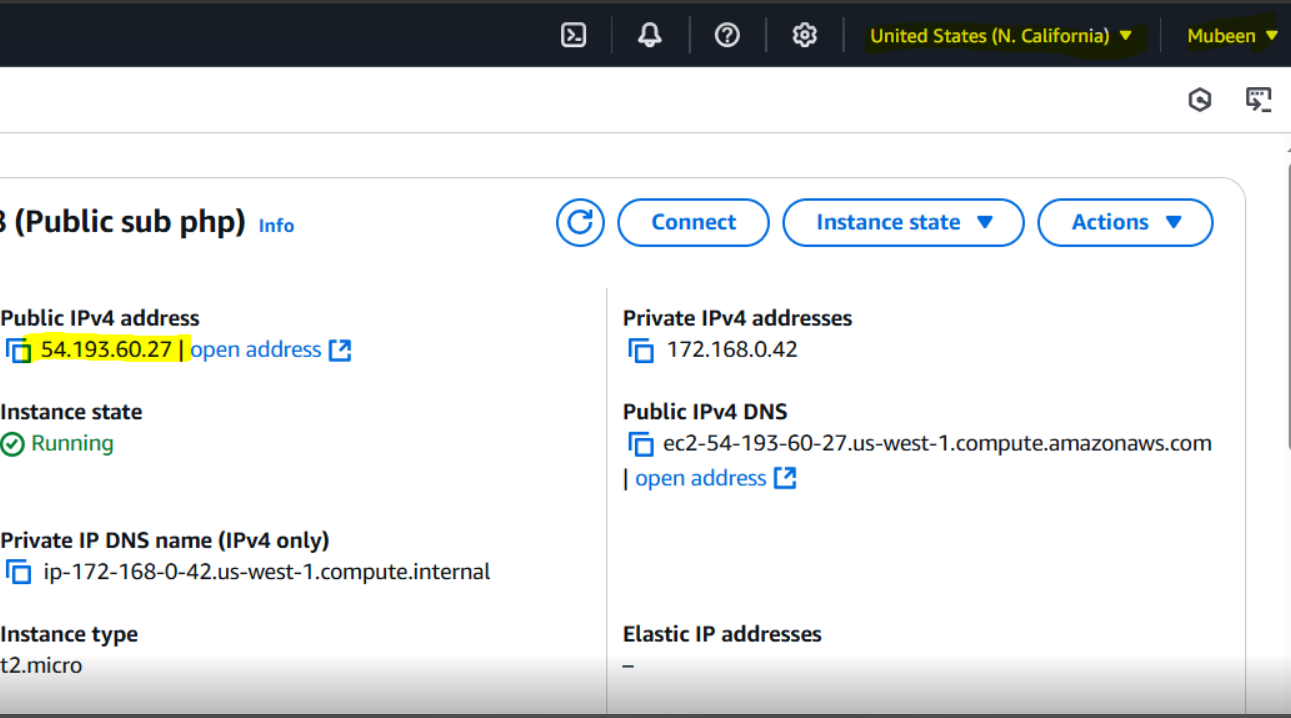
**Step 3: Update Route Tables**

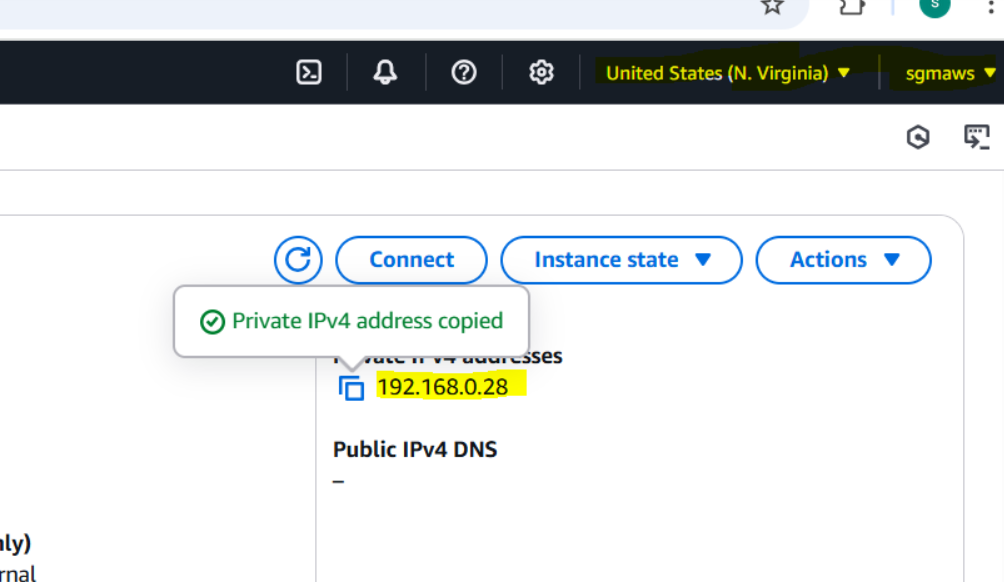


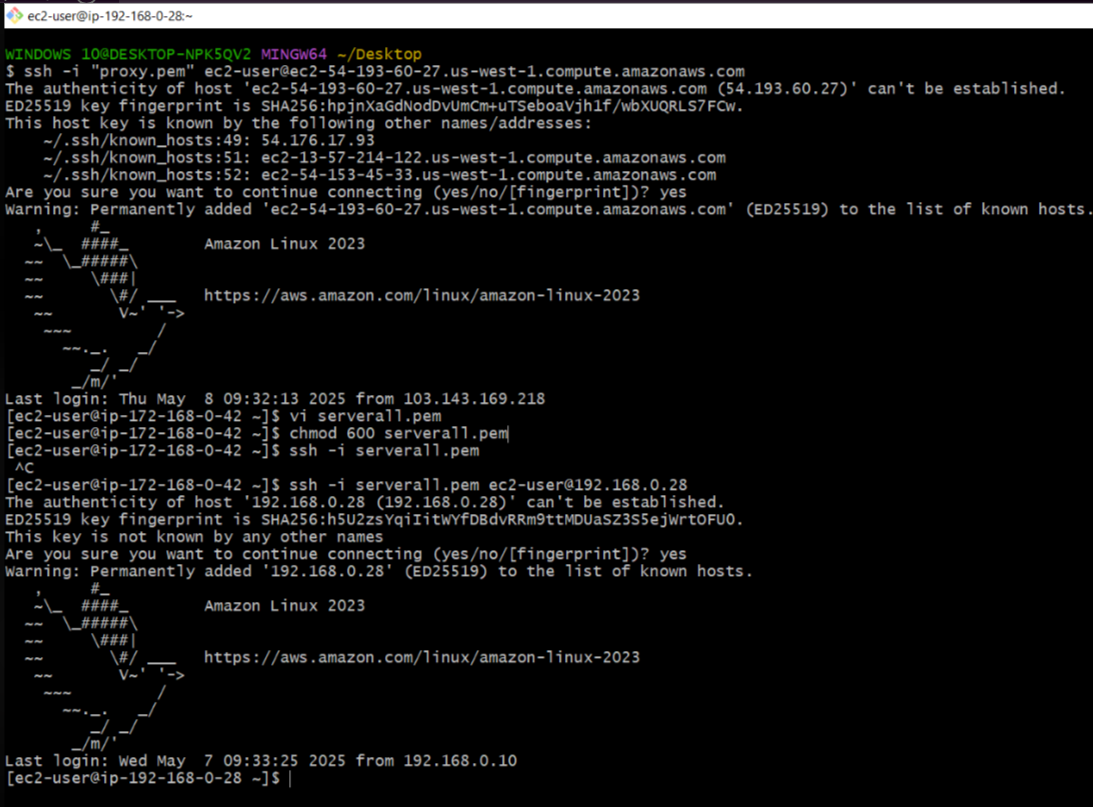
Update the route tables in both VPCs to include routes for the peering connection.



From **Mubeen ID- Region-California-Public server** to Ghouse **ID- Regions-virginia-Private ID**.







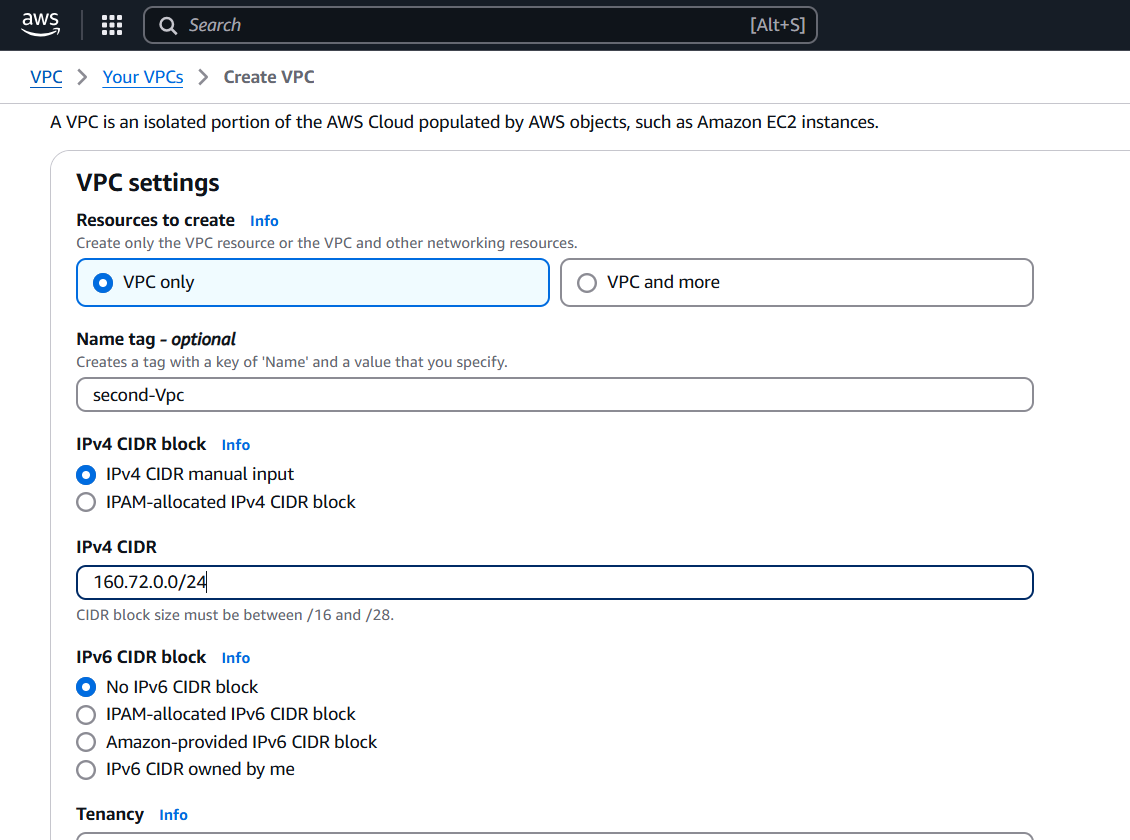
**4) Setup VPC Transist gateway.**

**Step 1: Create 4 VPCs**

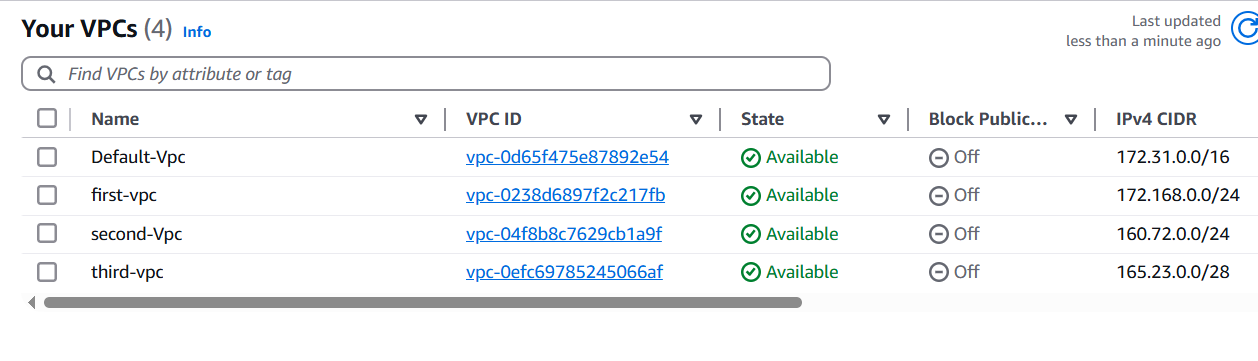
1. Go to VPC console.

2. Click "Create VPC".

3. Create 4 VPCs with unique CIDR blocks.



Similarly make other Vpc’s with different IP as shown in screenshot, Default VPC is also used.



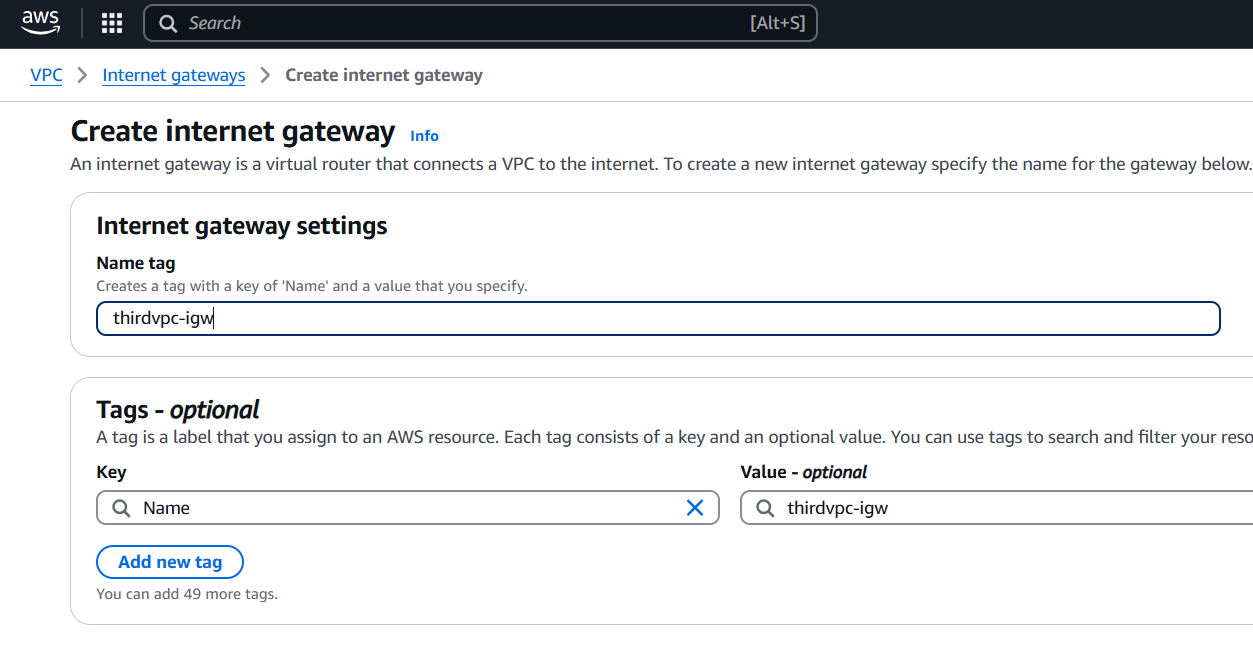
**Step 2: Create IGWs and Attach to VPCs**

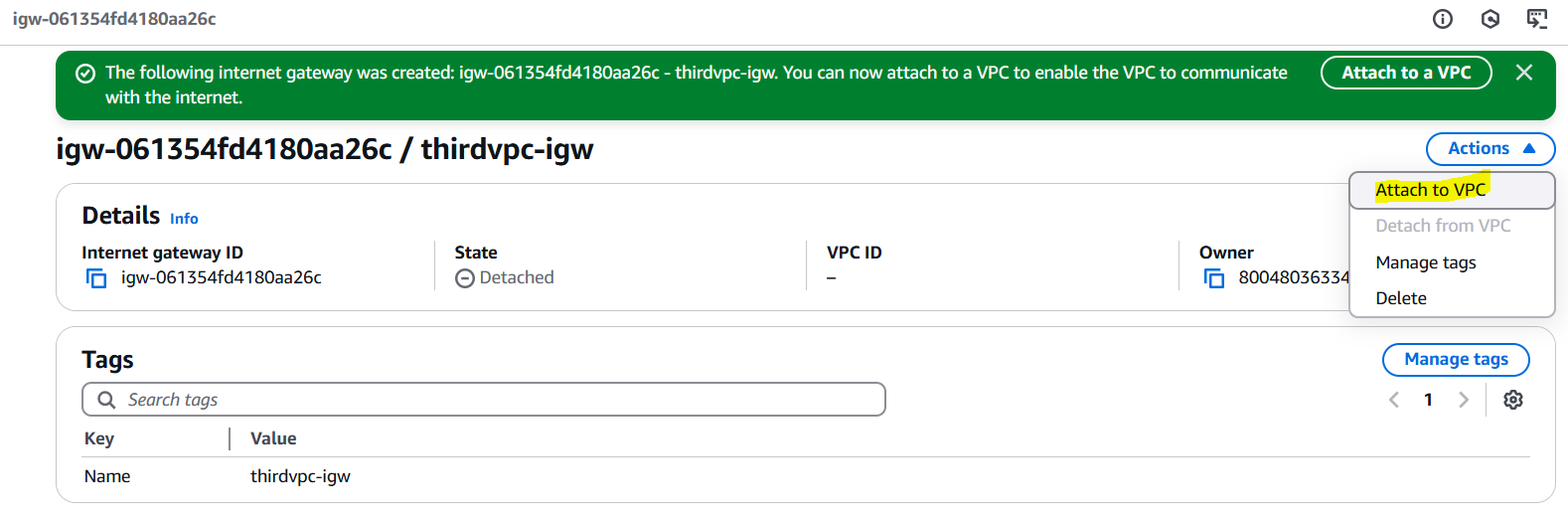
1. Create an IGW for each VPC:

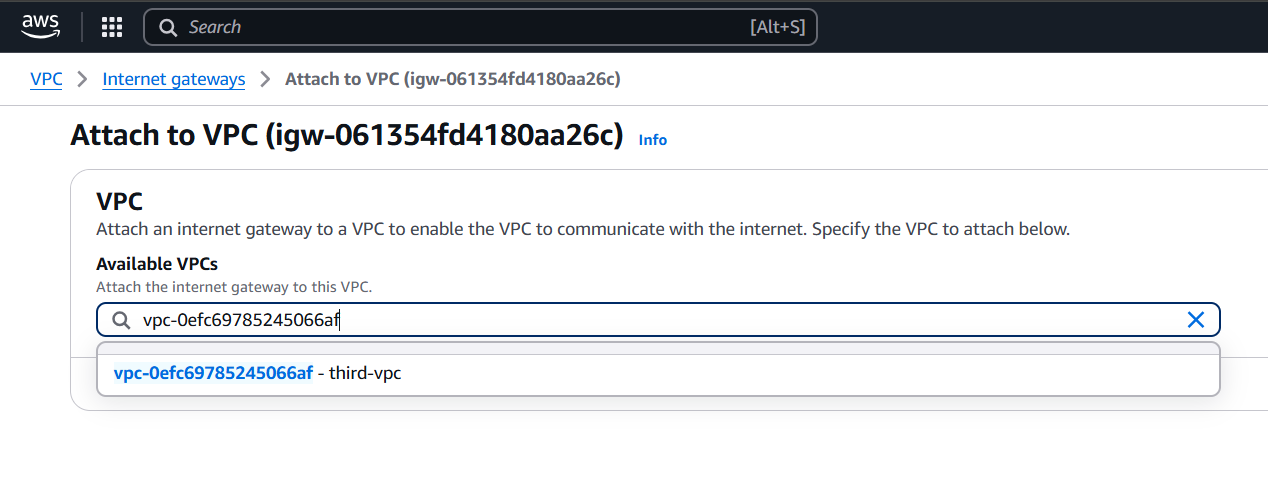
- Go to VPC console > Internet Gateways > Create internet gateway.

- Attach each IGW to a VPC:

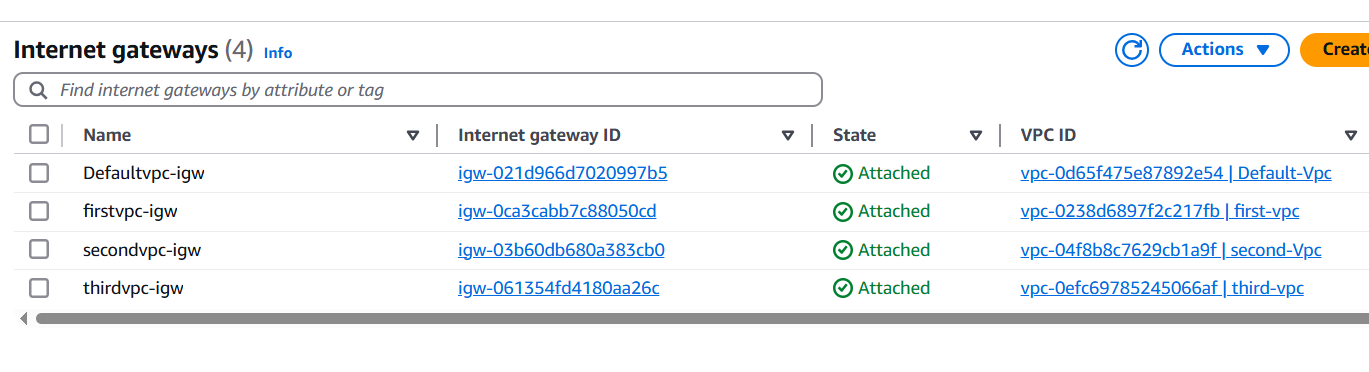
- Actions > Attach to VPC.







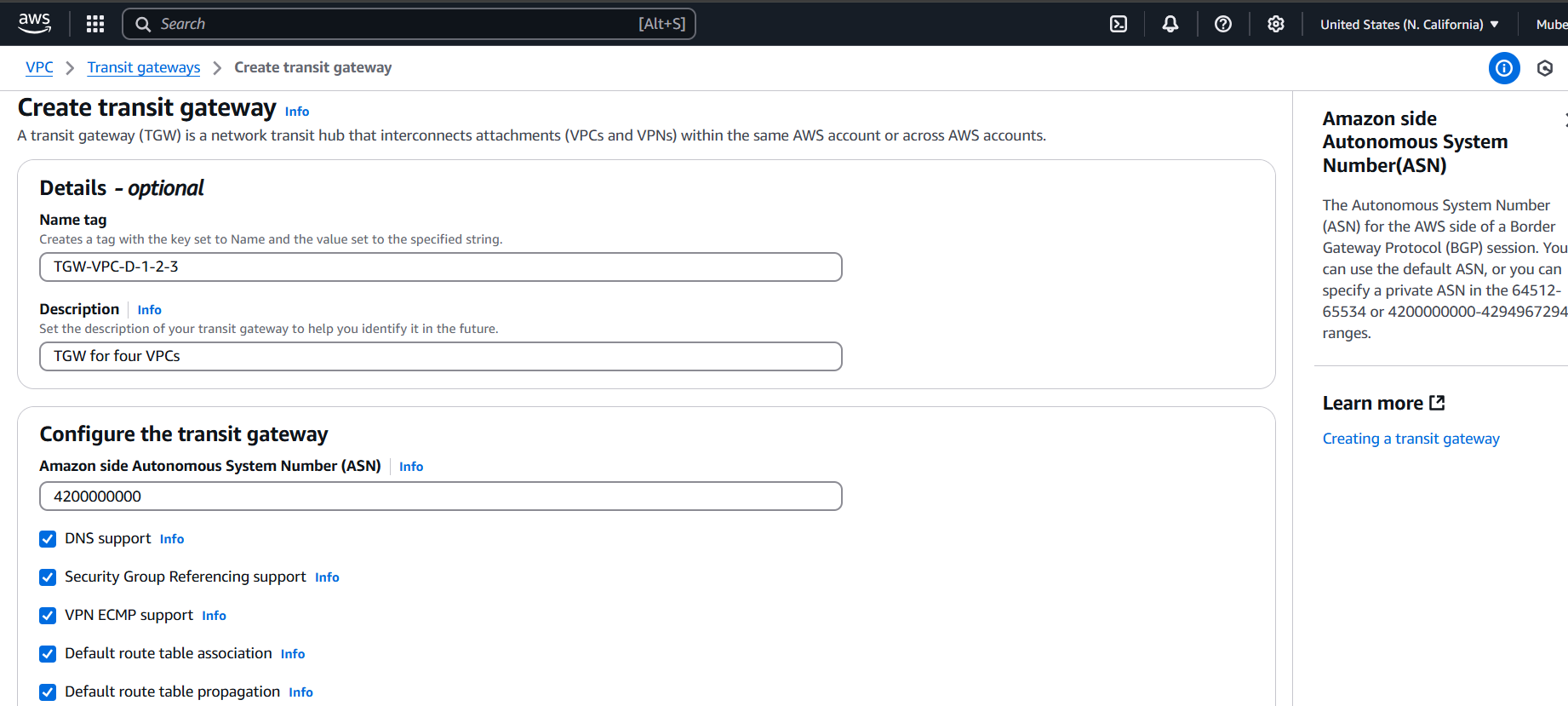
Similarly associated all VPC’s to IGW.



**Step 3: Create Transit Gateway (TGW)**

1. Go to VPC console > Transit Gateways > Create transit gateway.

2. Enter a name and description.



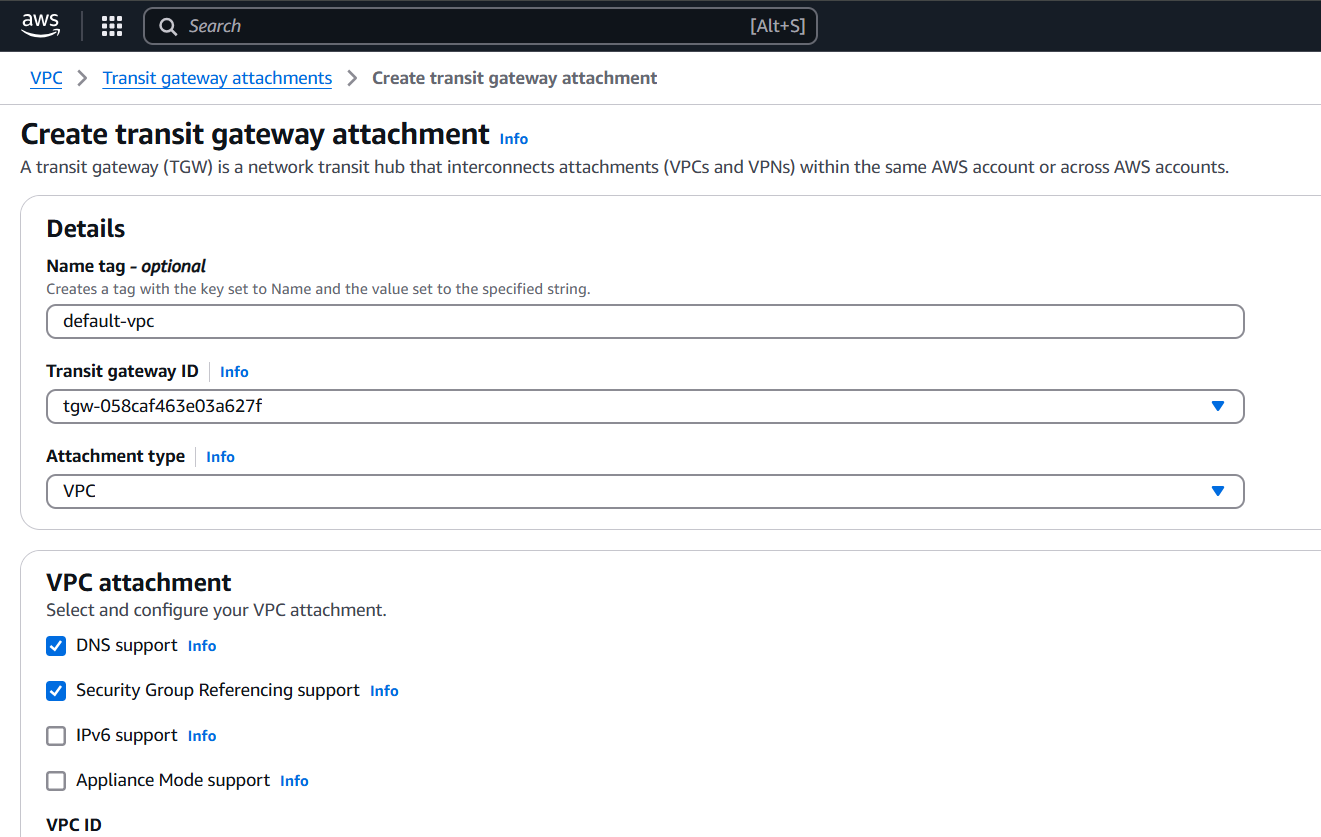
**Step 4: Create TGW Attachments**

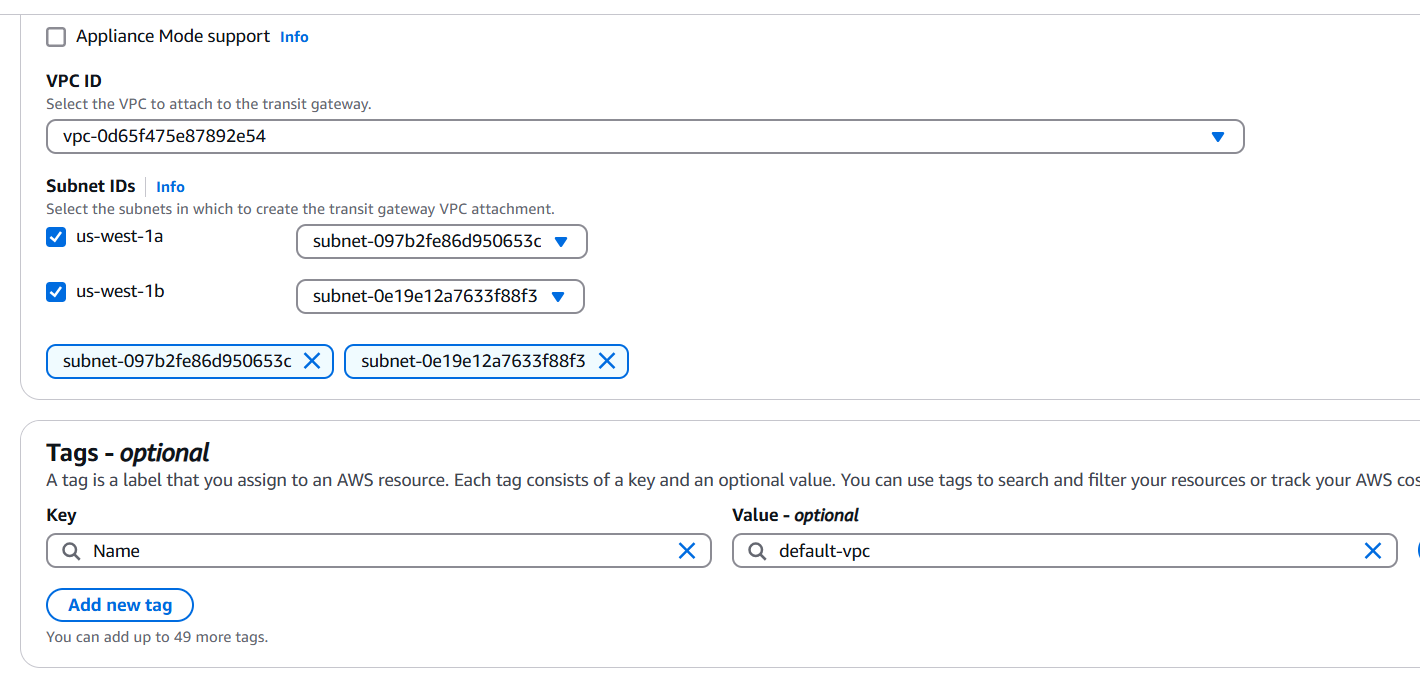
1. Create attachments for each VPC:

- Go to VPC console > Transit Gateways > Transit gateway attachments > Create transit gateway attachment.

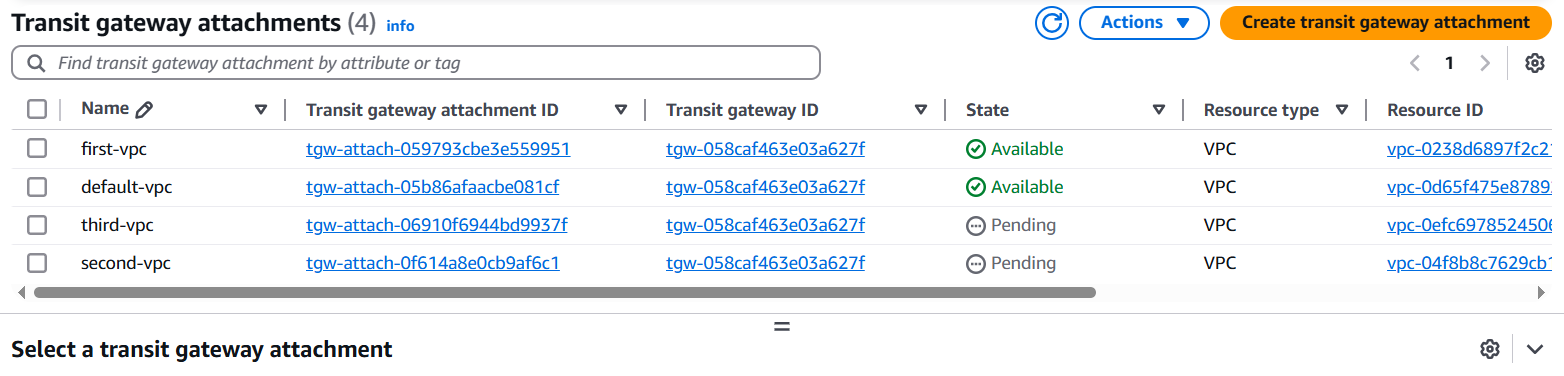
- Select the VPC and subnet(s) to attach.

- Repeat for each VPC.





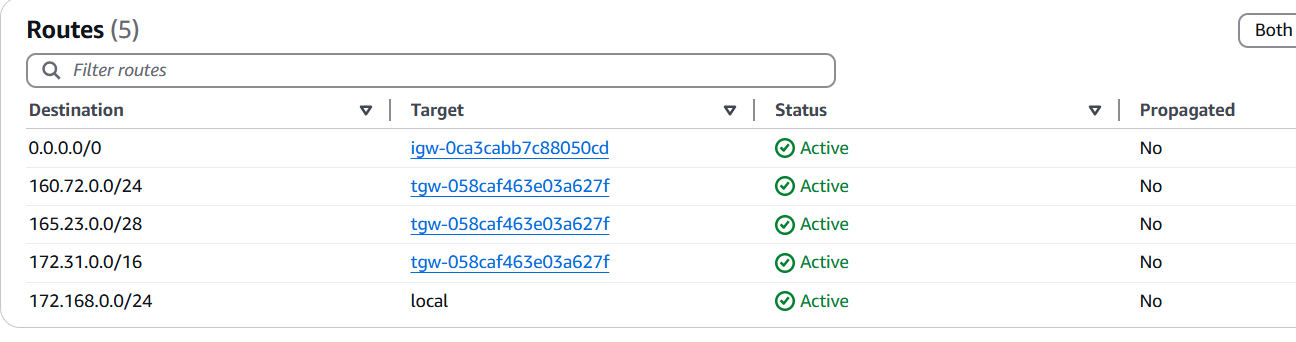
Similarly attach TGW to all VPCs.



**Step 5: Update Route Tables**

Update each VPC's route table:

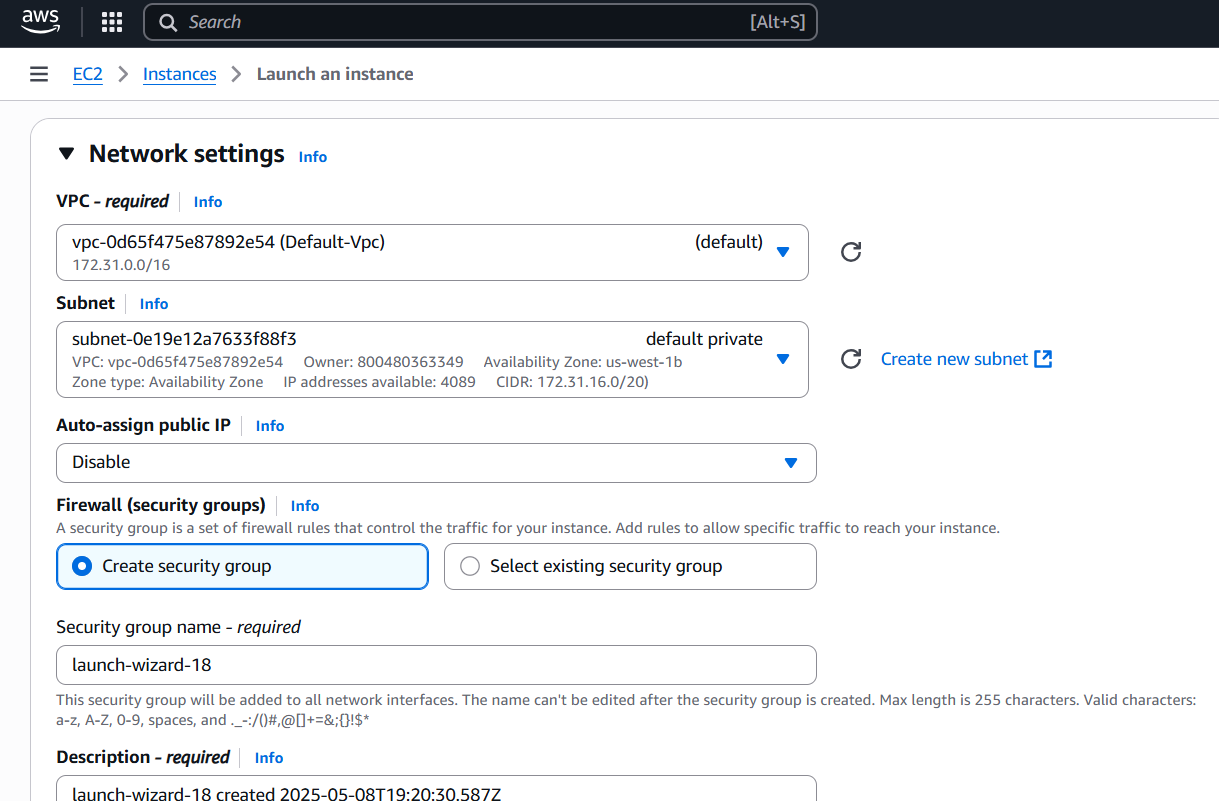
- Add a route for the TGW.



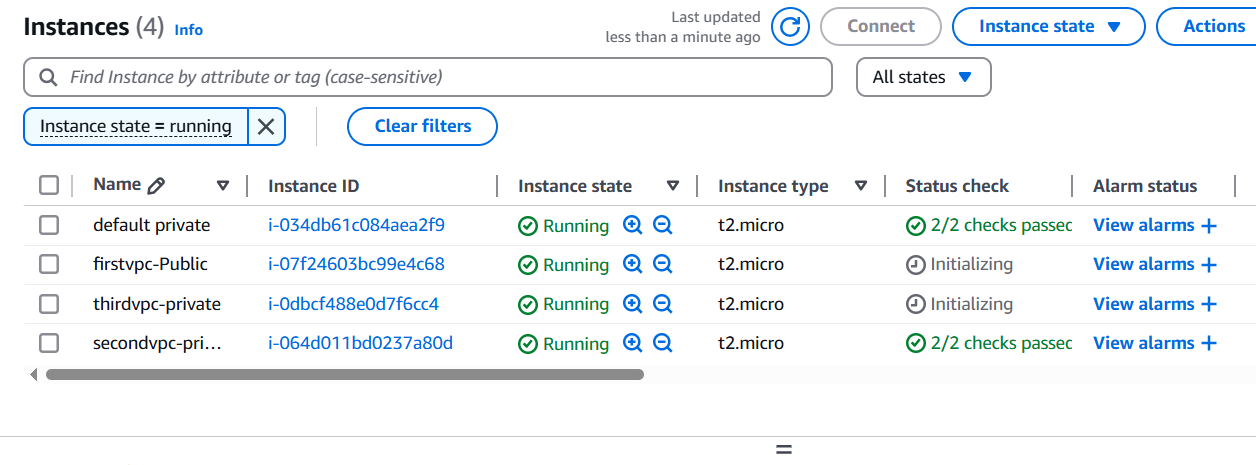
Similarly add TGW to all RTs of VPC.

**Step 6: Launch 4 instances**

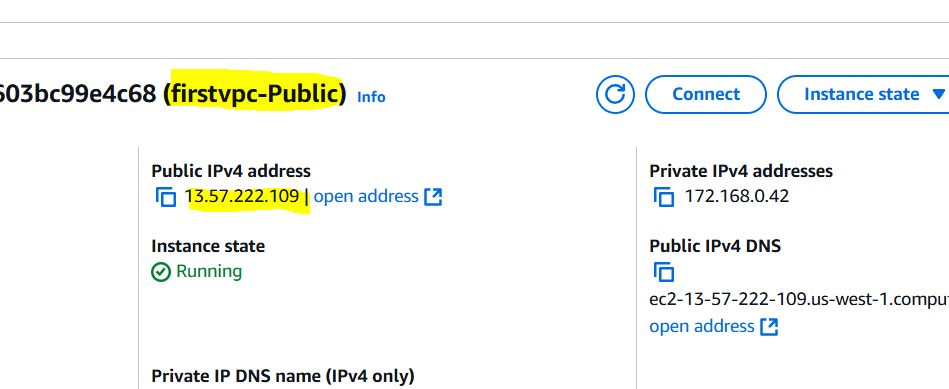
We are launching 1 public and 3 private instances.



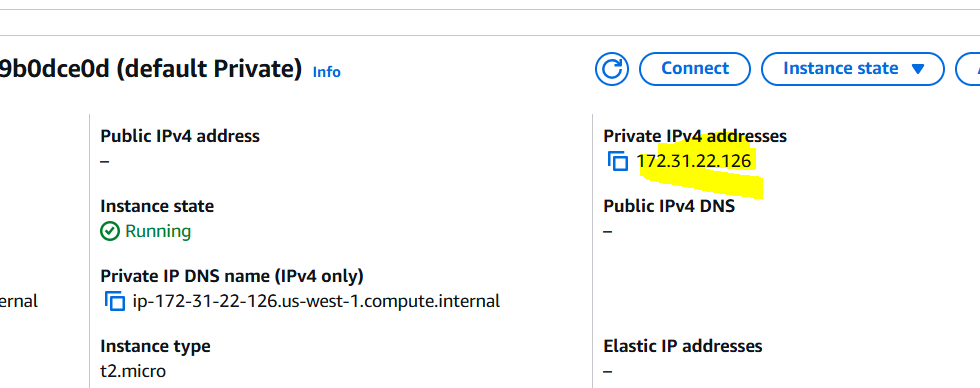
Similarly create all 4 instances

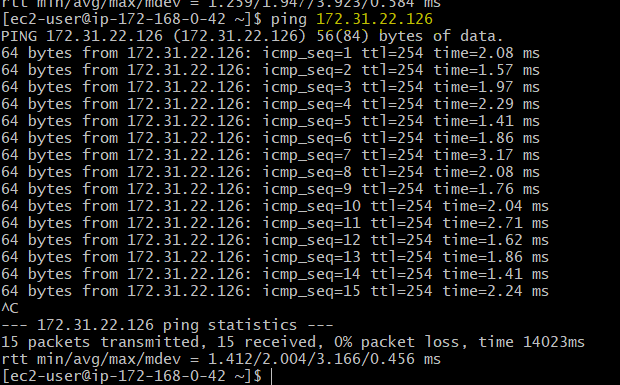


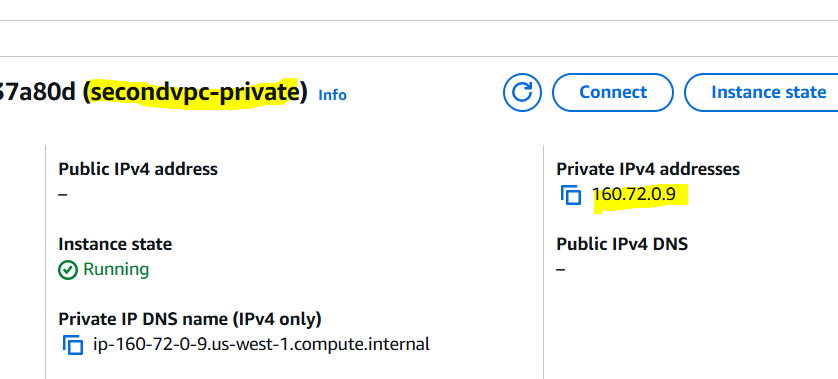
**1 Public server**

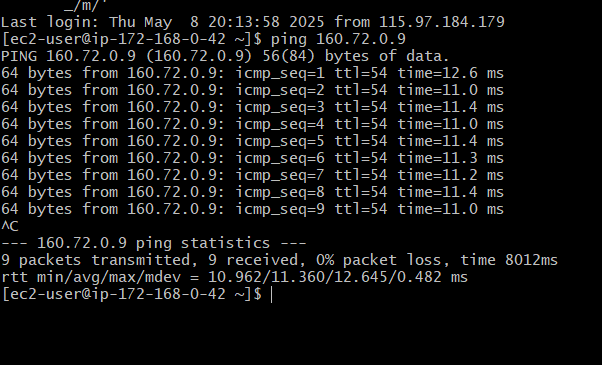


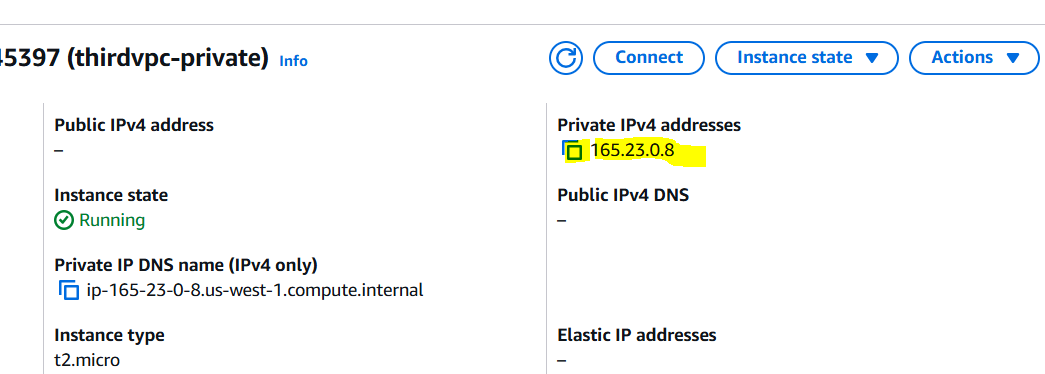
**3 Private servers connection checking through ping command and connecting the last one.**

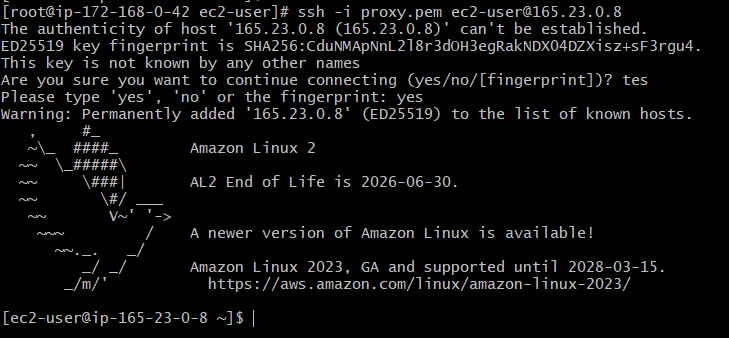








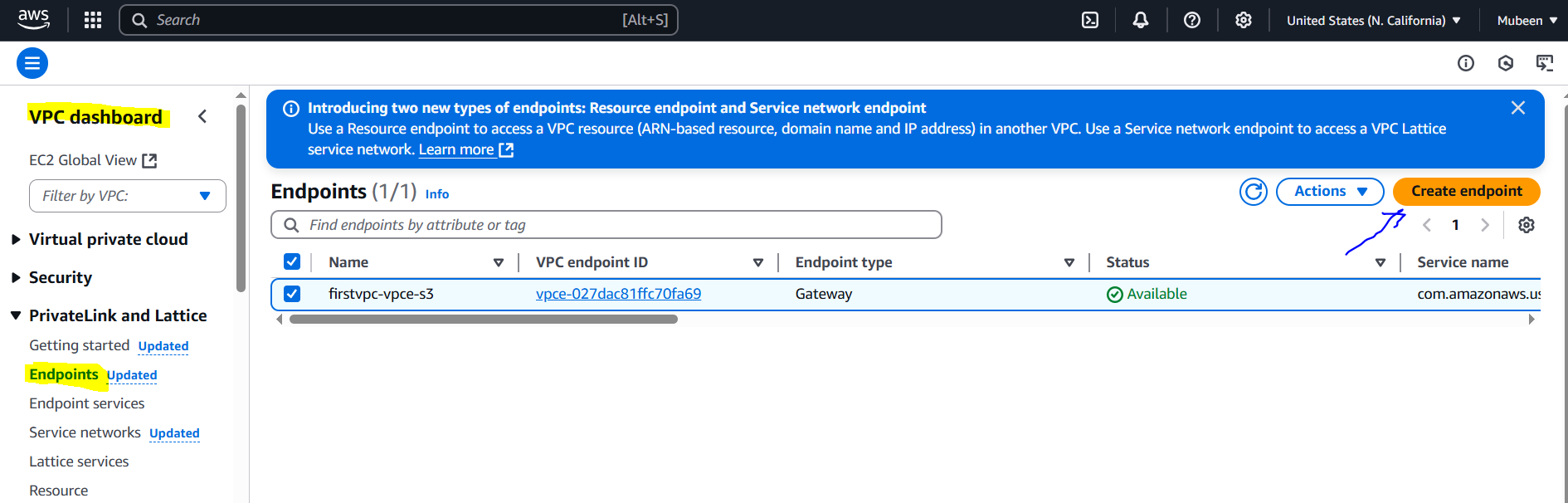


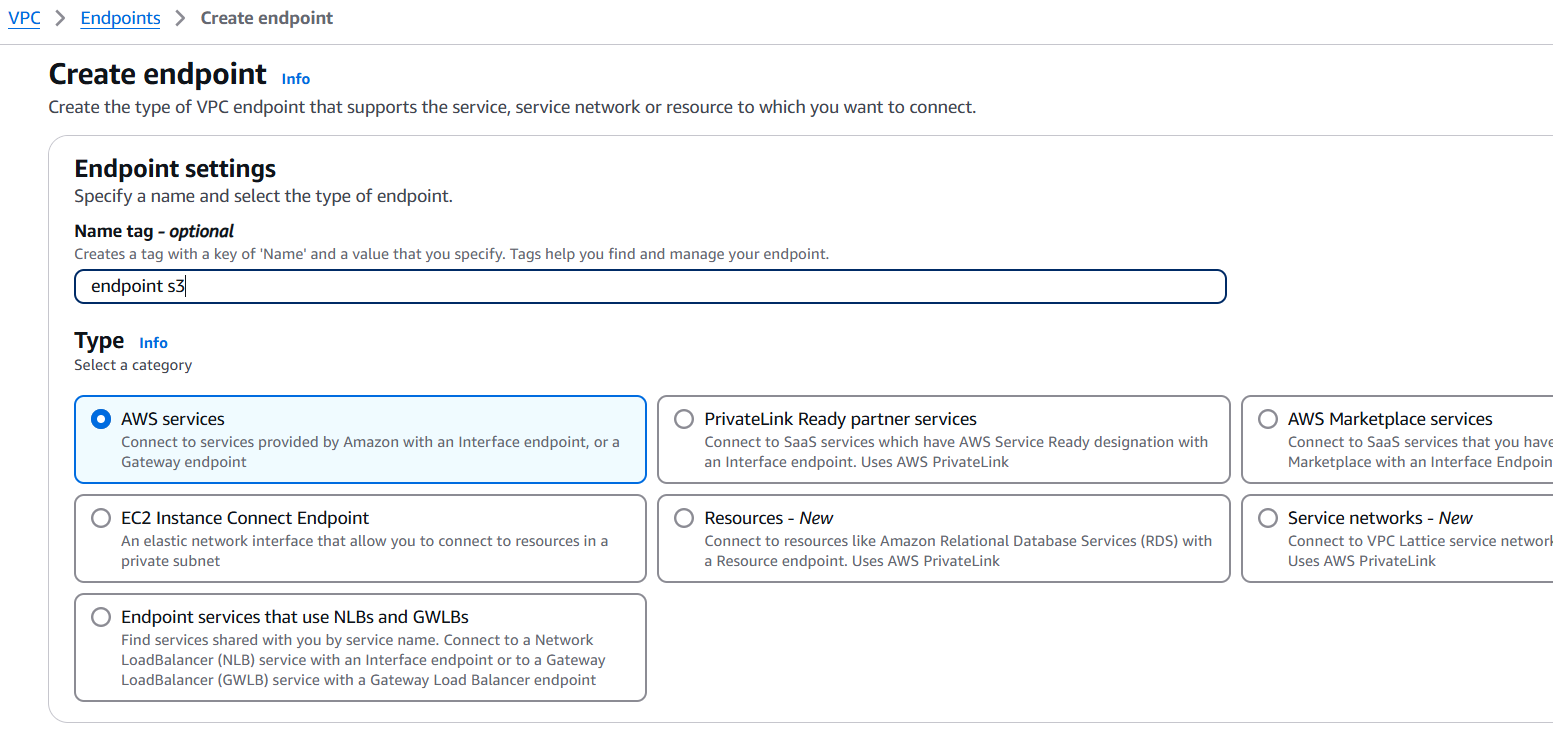


**5) Setup VPC End Point.**

1. Go to the VPC console.

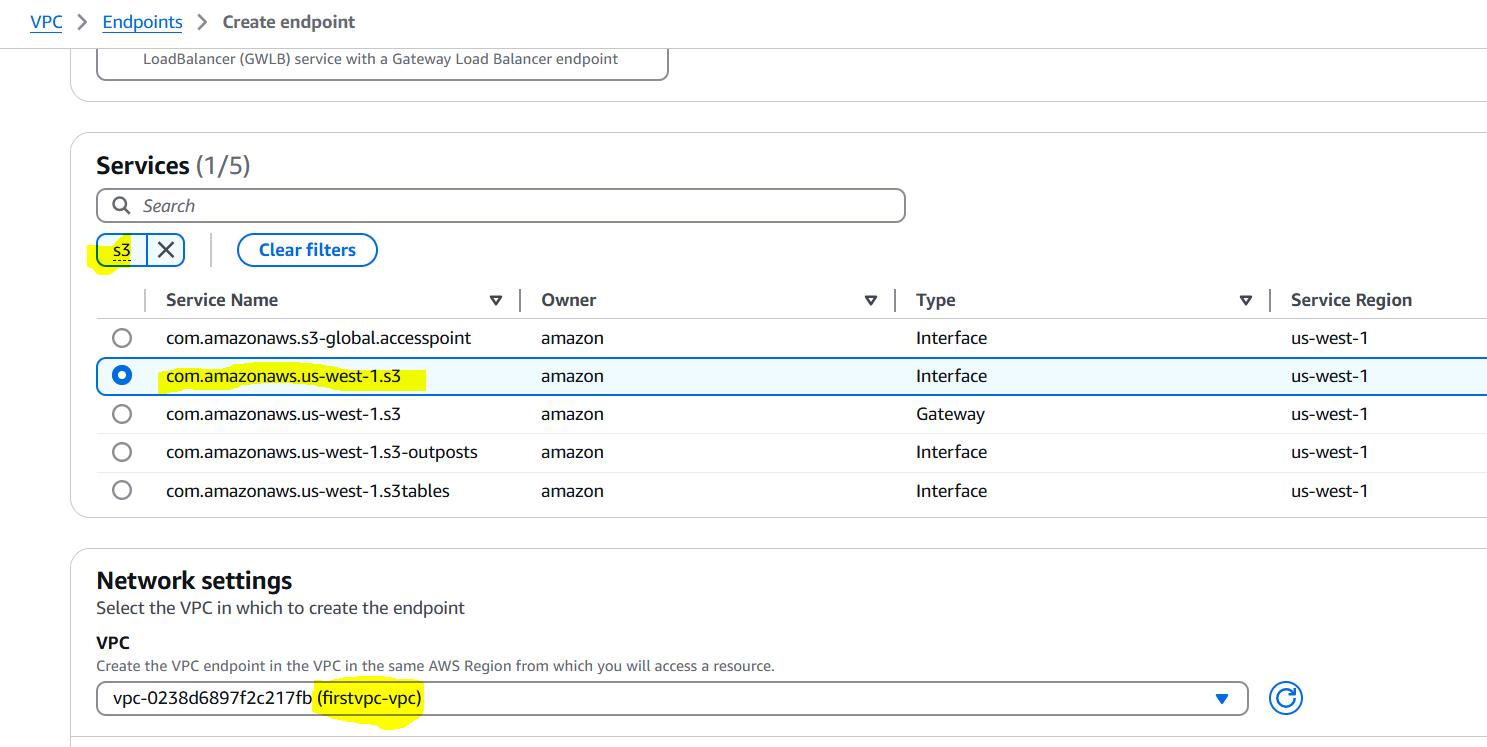
2. Click on "Endpoints" and then "Create endpoint".

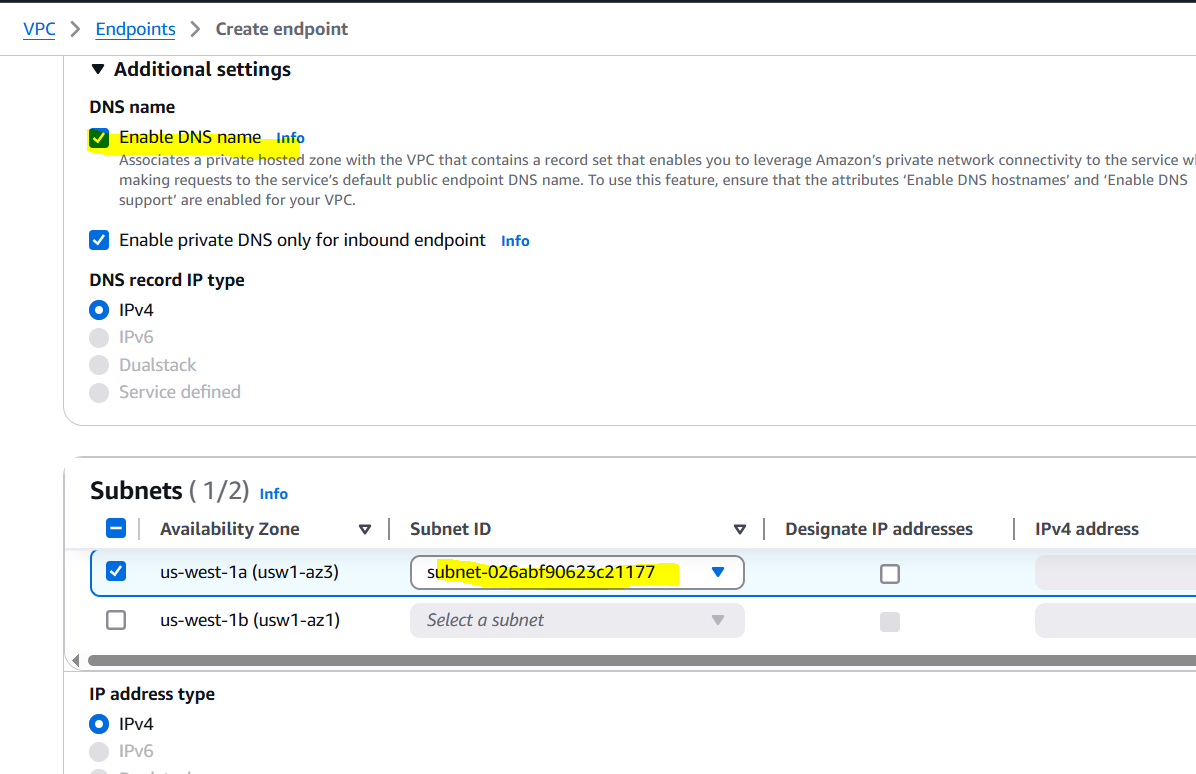




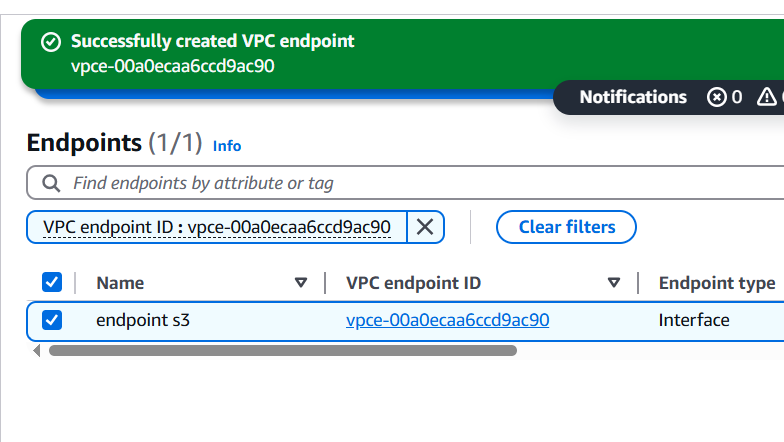
3. Select the service category S3.

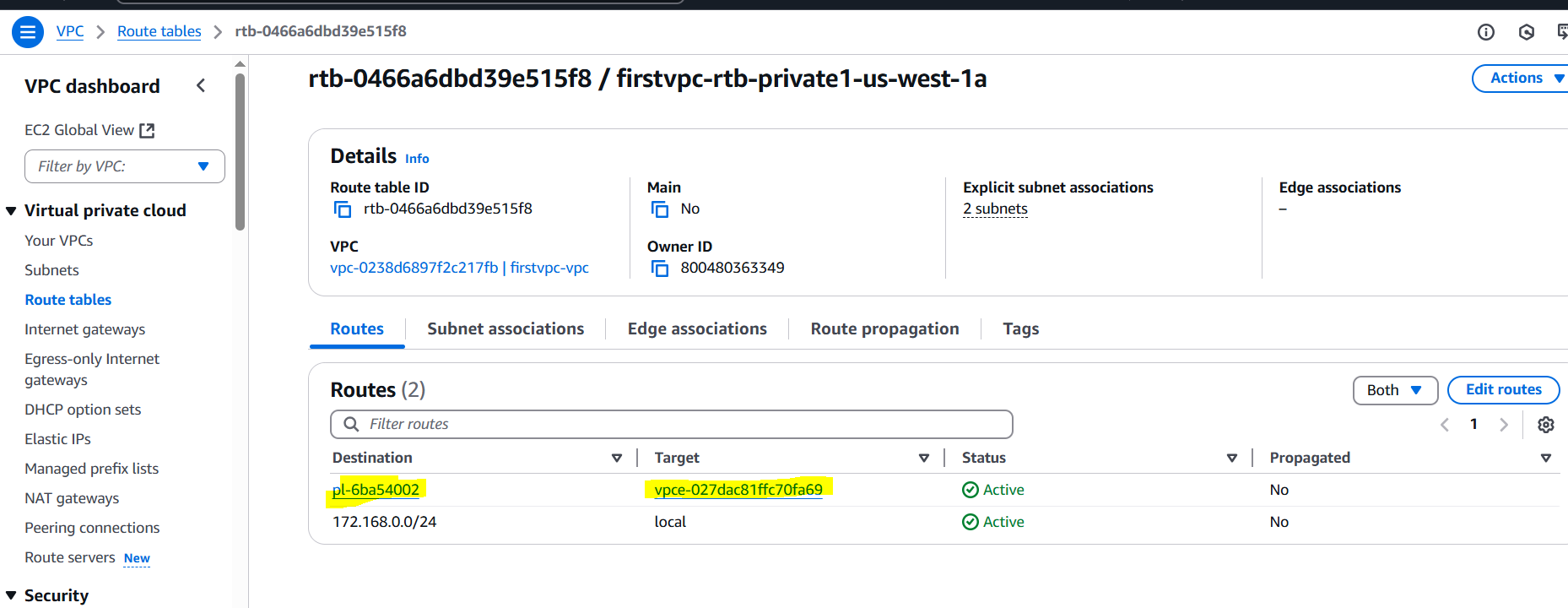
4. Choose the VPC and route table(s).



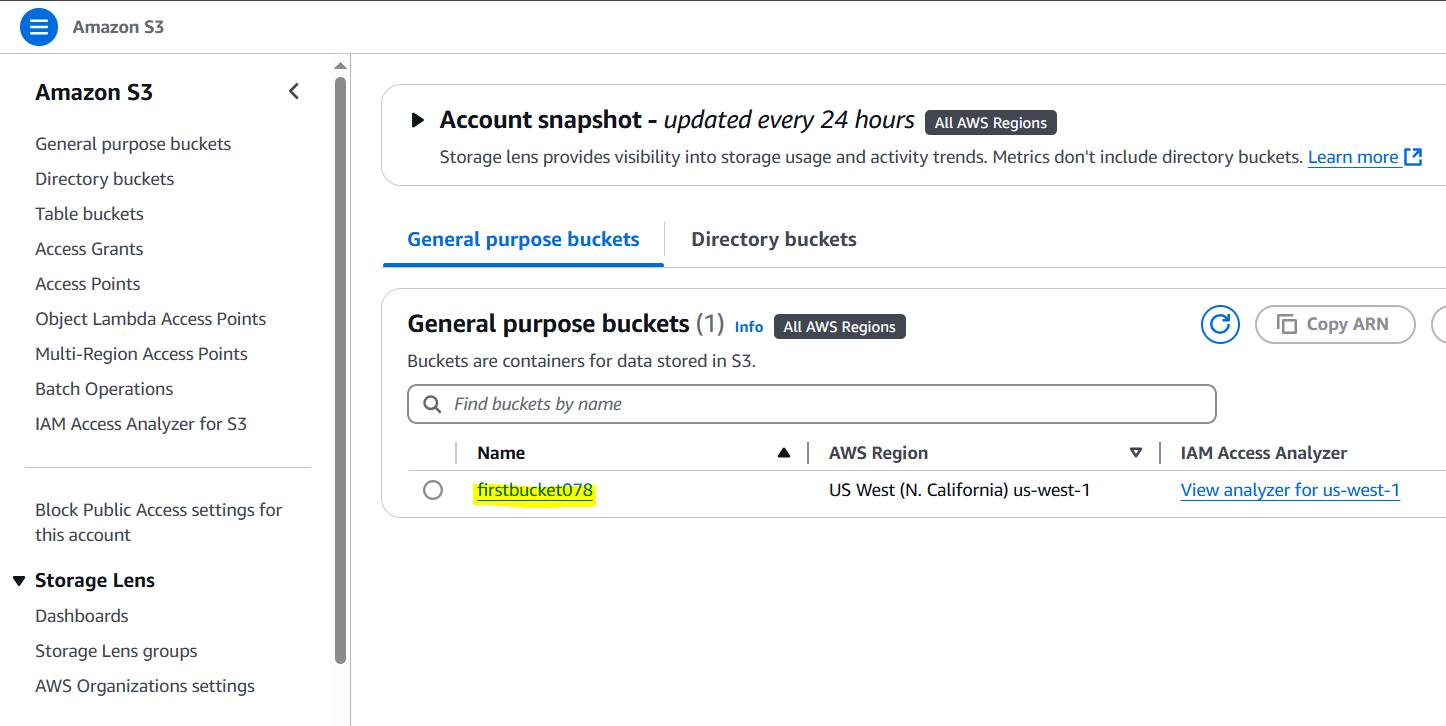


5. Click "Create endpoint".

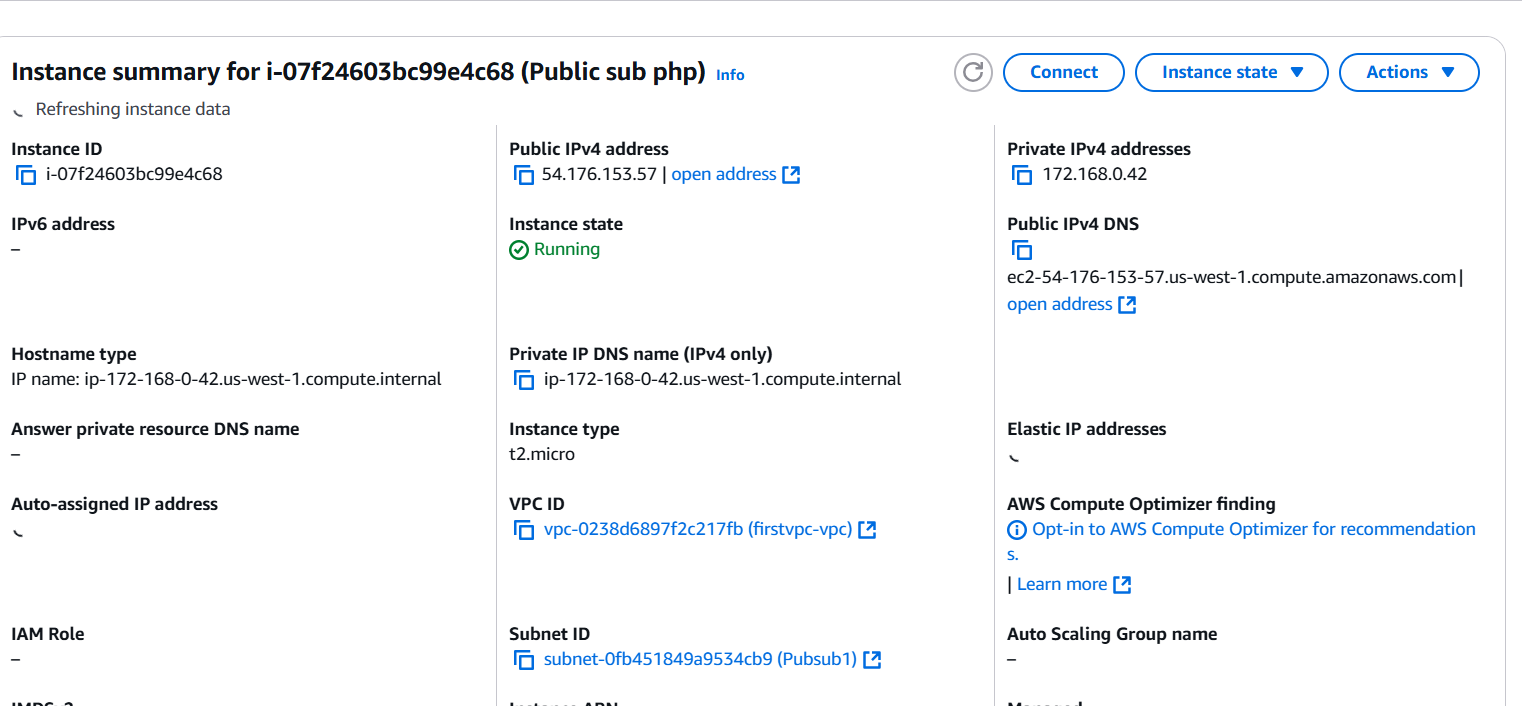


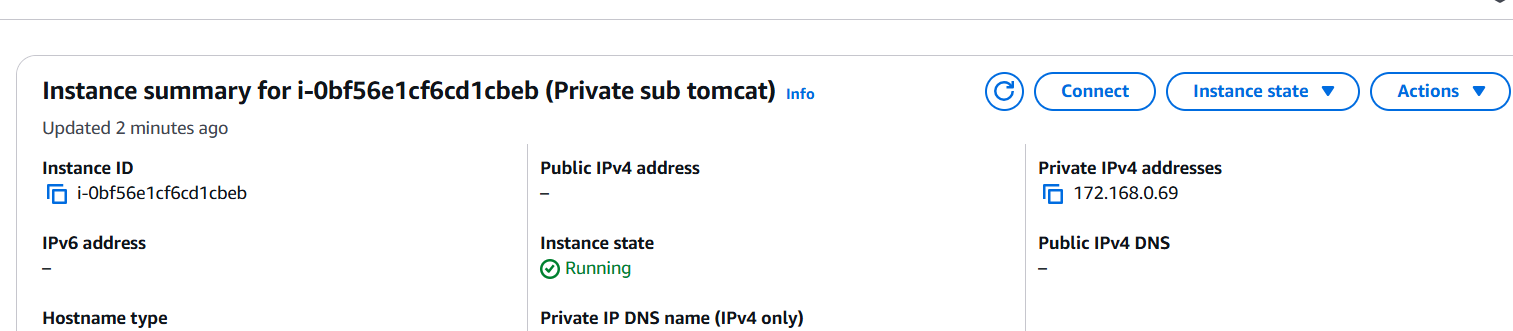
We can check that end point is attached to Route table.  
  


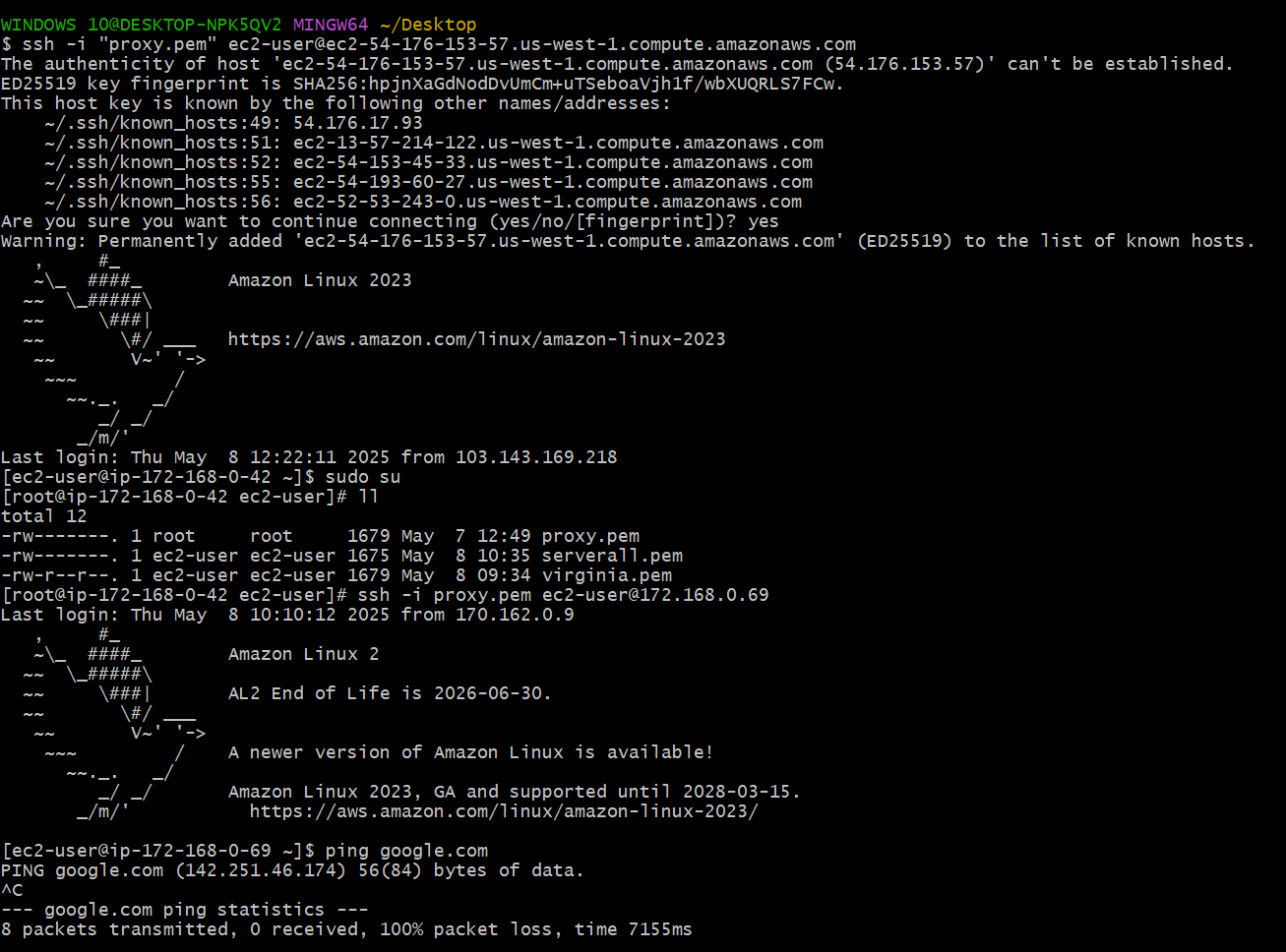
We have a existing S3 bucket.



Log in to the public server, then connect to the private server. Using existing servers.



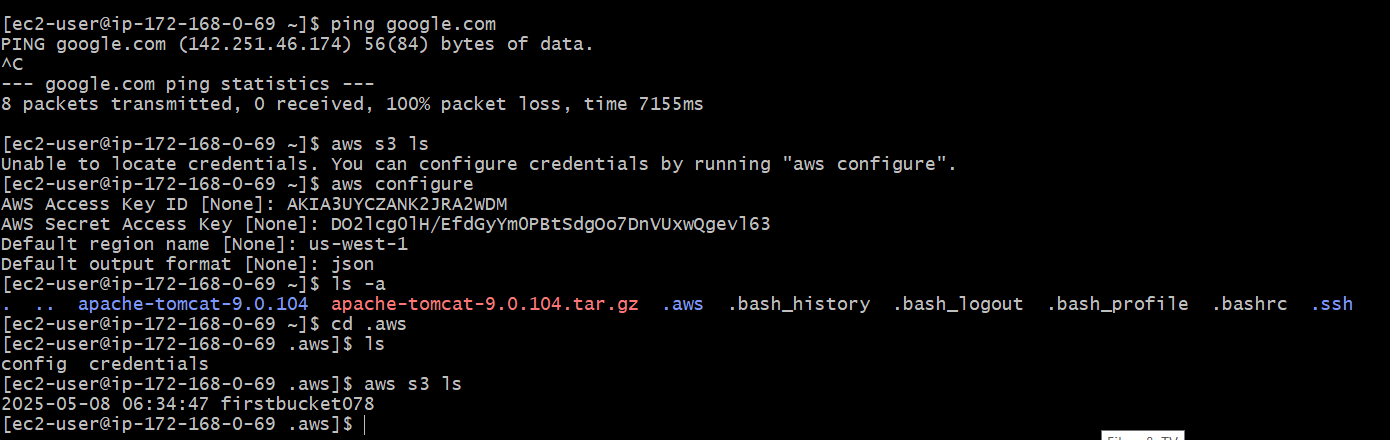




Create access key and security key to configure aws.



1. By using command aws configure we configure the service and .aws new directory with sub config and credentials is created automatically.
2. By using command aws s3 ls we can see the s3 bucket, that means we have completed Endpoint setup successfully.
3. For reference in the screenshot ping google.com is used to make it clear that there no internet access.



**The-End**