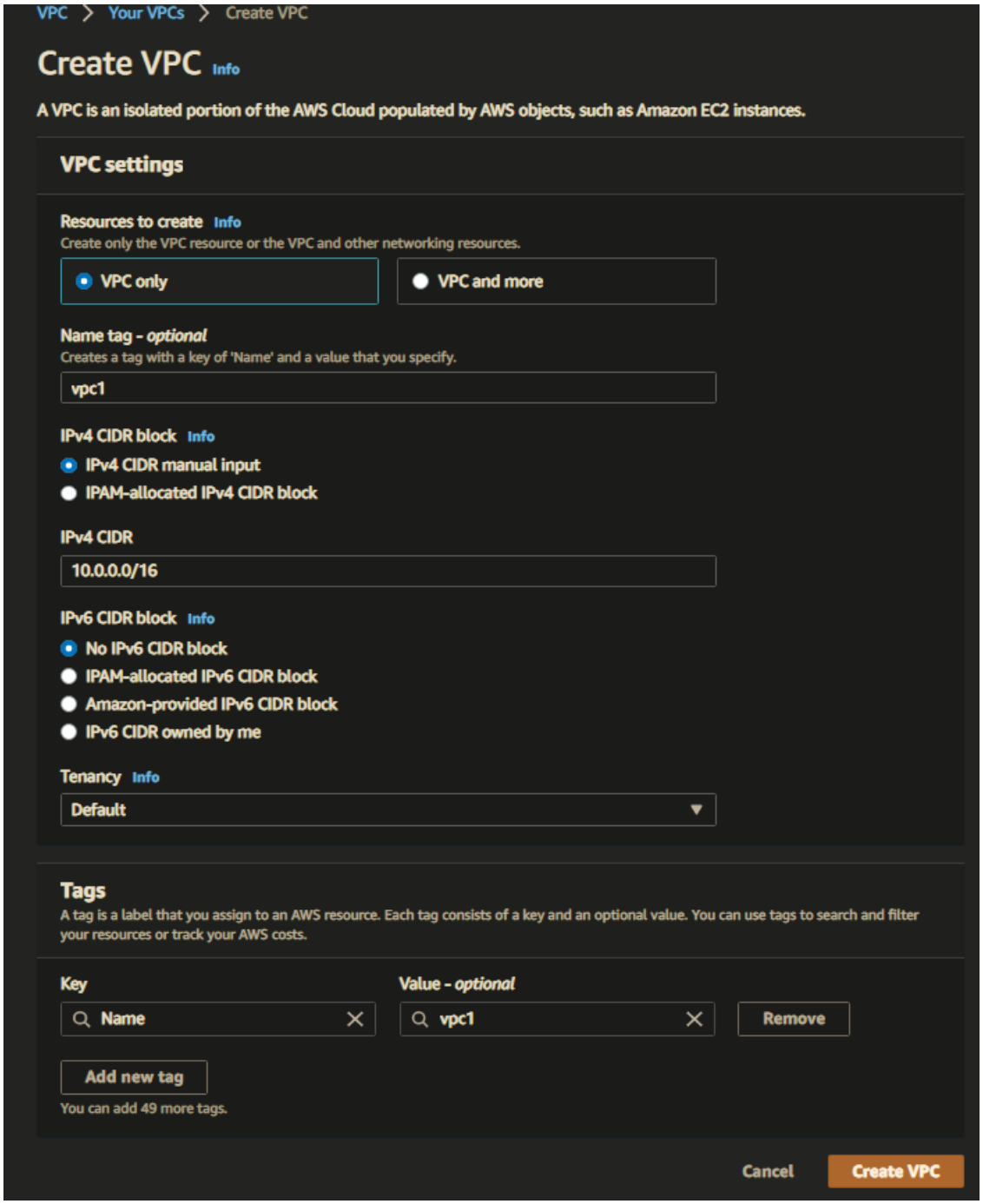
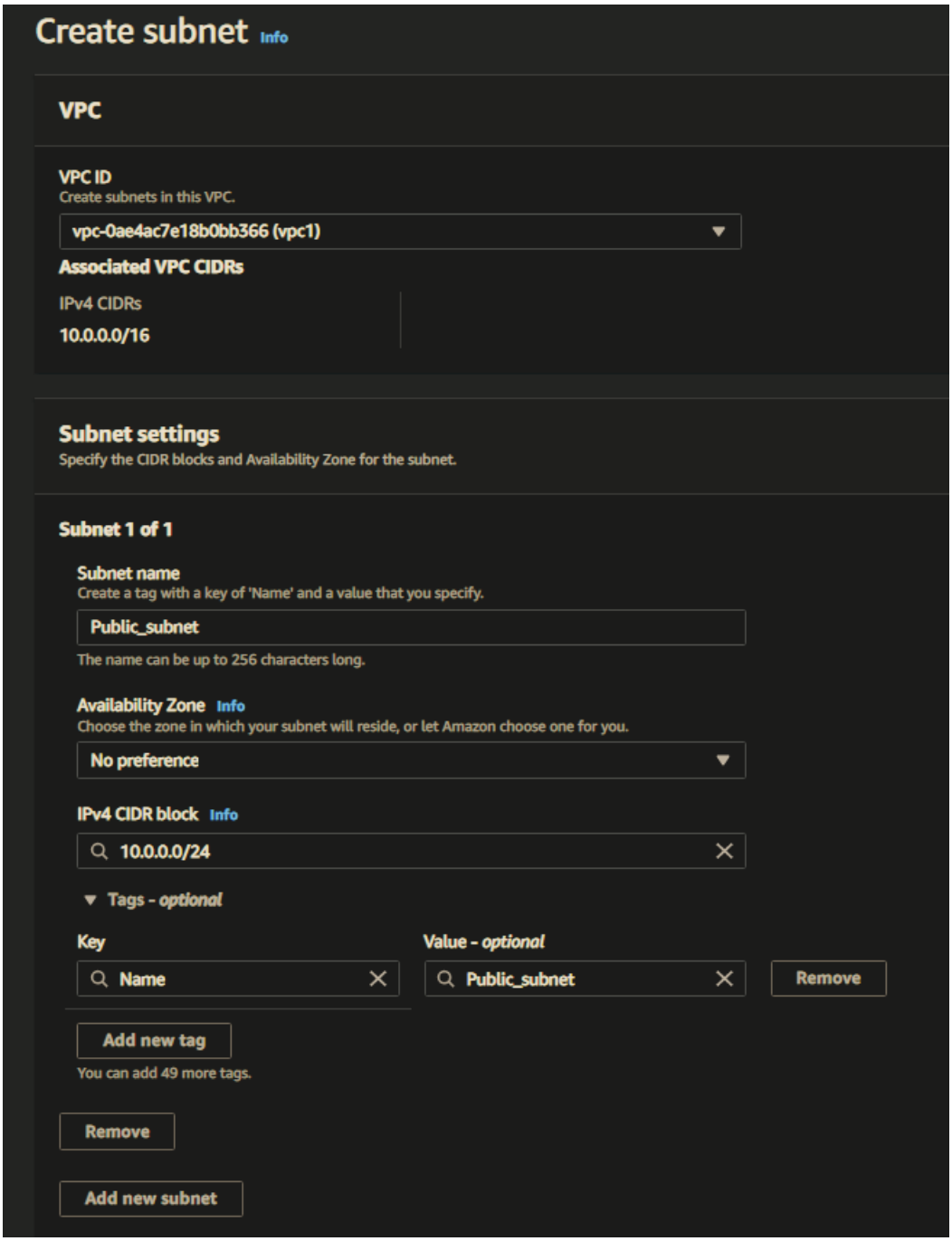
NAT GATEWAY LAB

Create a VPC

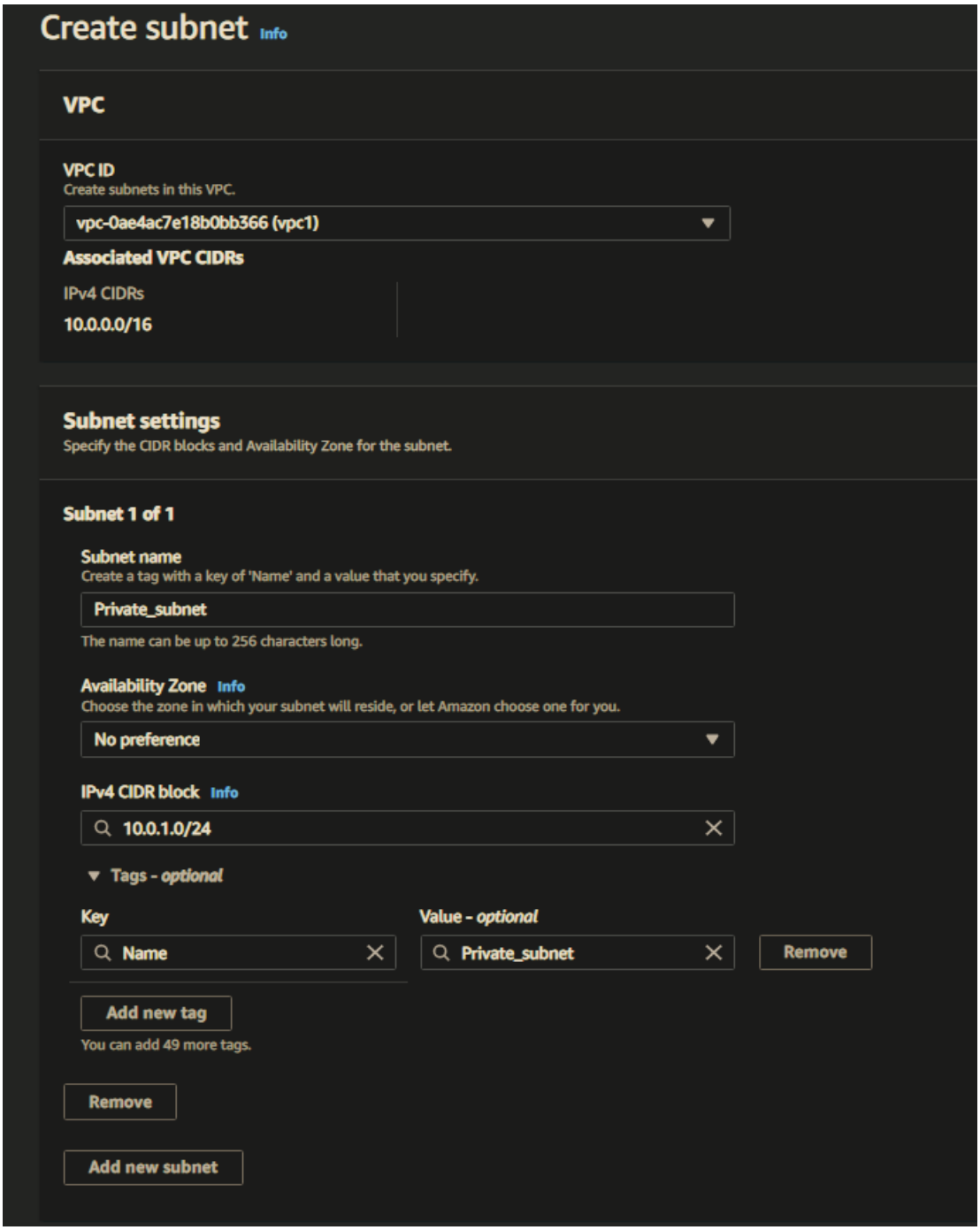


Create Subnets one for public and one for private

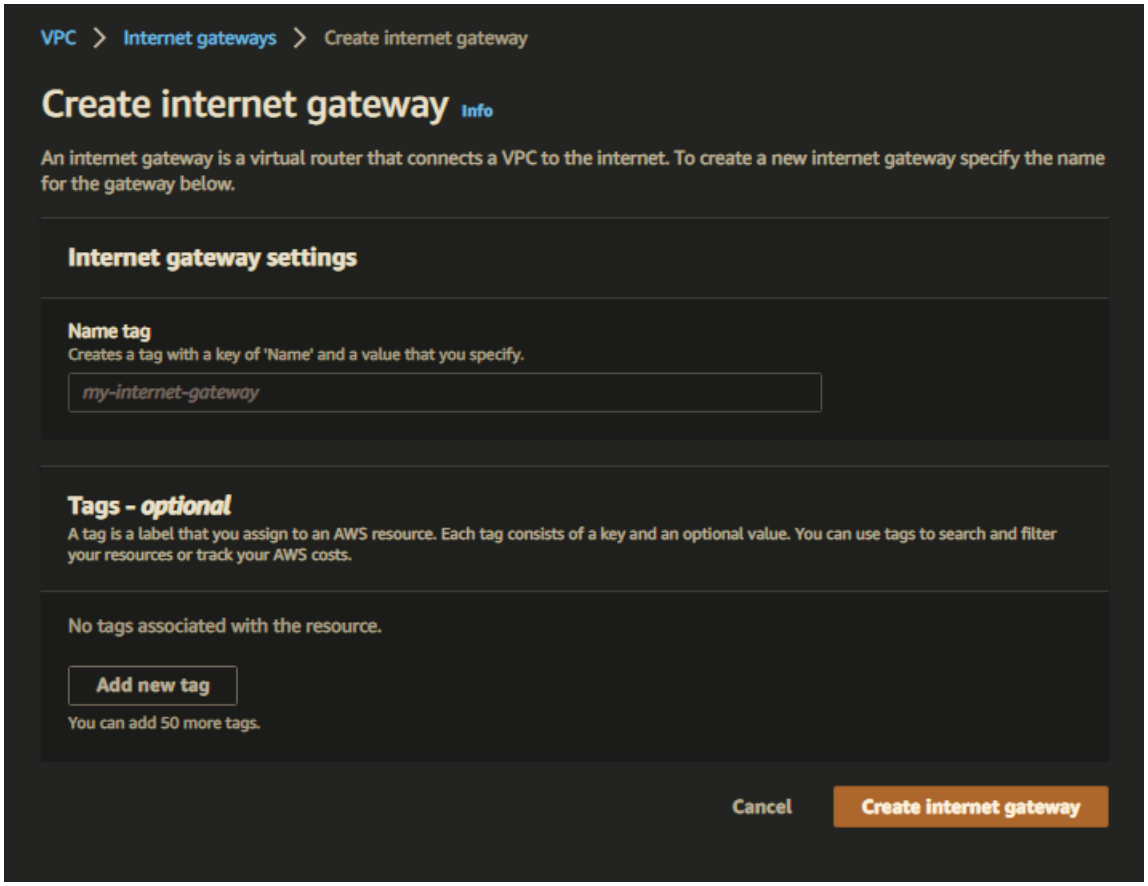
- Public



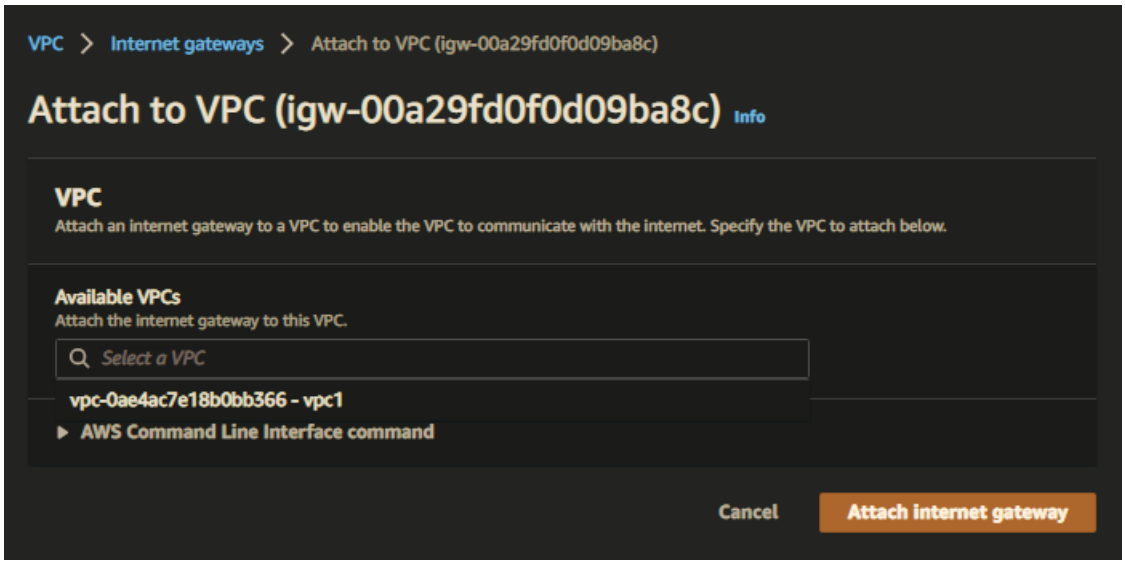
- Private


Create subnet Info
VPC
VPC ID
Create subnets in this VPC.
vpc-0ae4ac7e18b0bb366 (vpc1)
Associated VPC CIDRs
IPv4 CIDRs
10.0.0.0/16
Subnet settings
Specify the CIDR blocks and Availability Zone for the subnet.
Subnet 1 of 1
Subnet name
Create a tag with a key of 'Name' and a value that you specify.
Private_subnet
The name can be up to 256 characters long.
Availability Zone Info
Choose the zone in which your subnet will reside, or let Amazon choose one for you.
No preference
IPv4 CIDR block Info
10.0.1.0/24
Tags - optional
Key
Value - optional
Name
Private_subnet
Remove
Add new tag
You can add 49 more tags.
Remove
Add new subnet

Create Internet Gateway and attach it to the VPC you have created

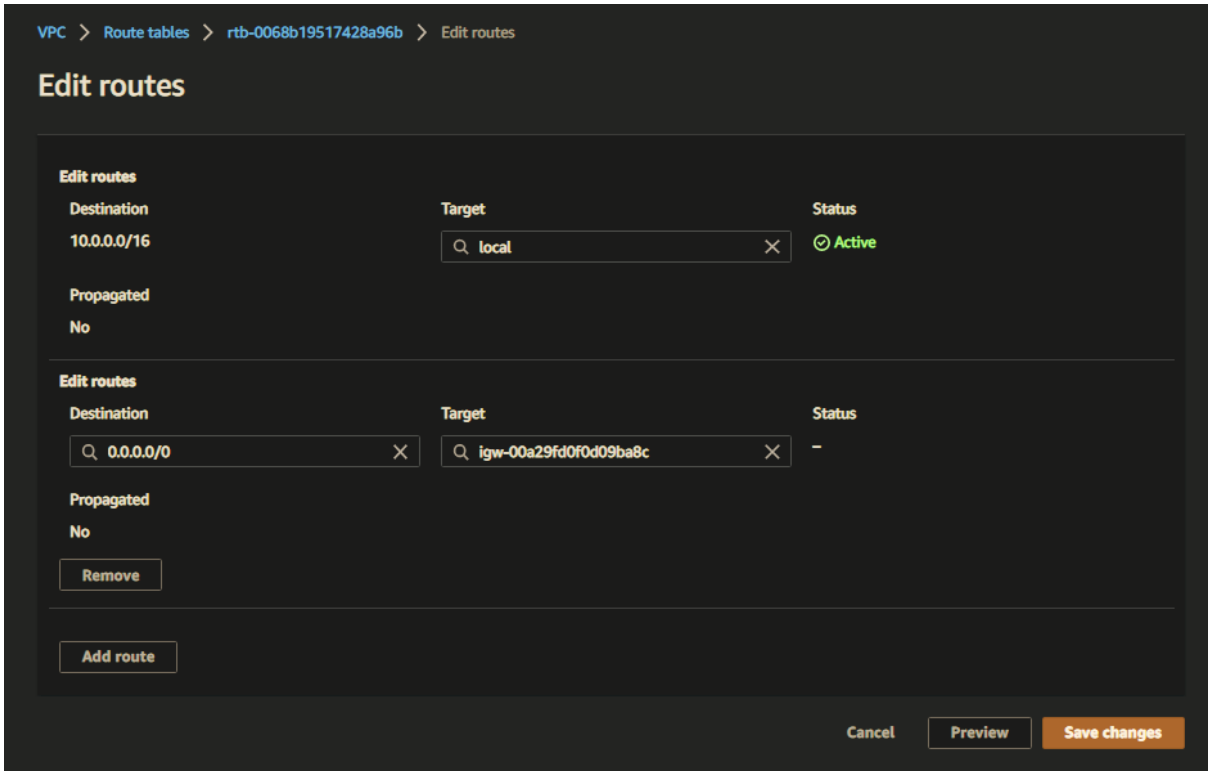


Select you Internet Gateway and click actions there you will find attach option, after clicking on it you can attach you IG to a VPC

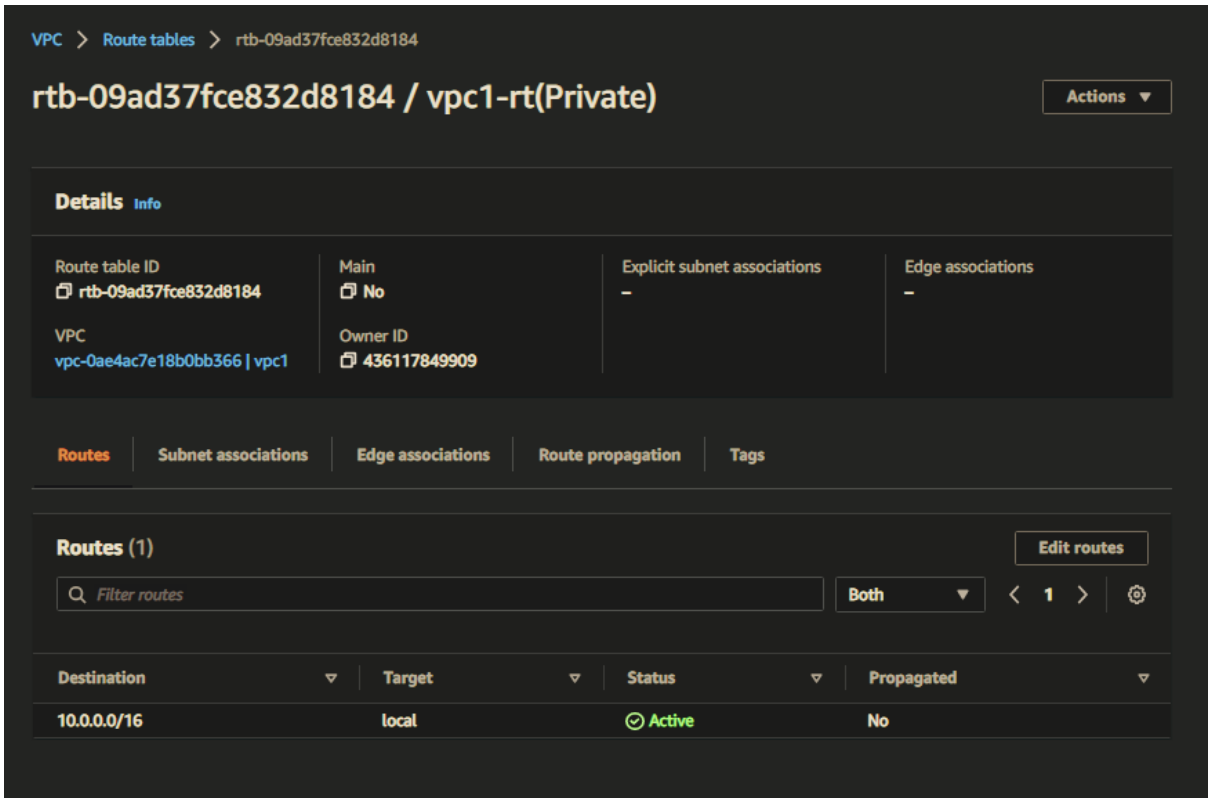


You can see here your IG is attached to the VPC you have created

Go to route table and add Internet gateway to your main route table:



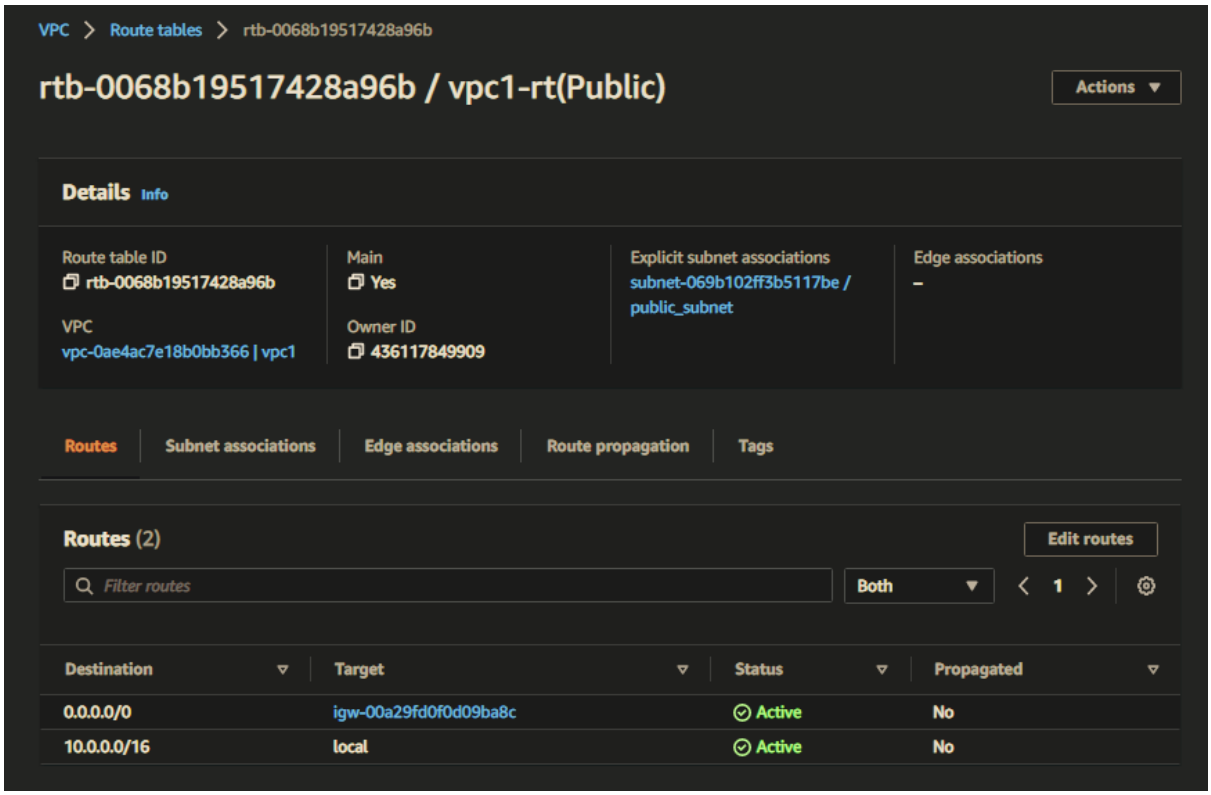
Create another Route table for the VPC with no IG attached to it



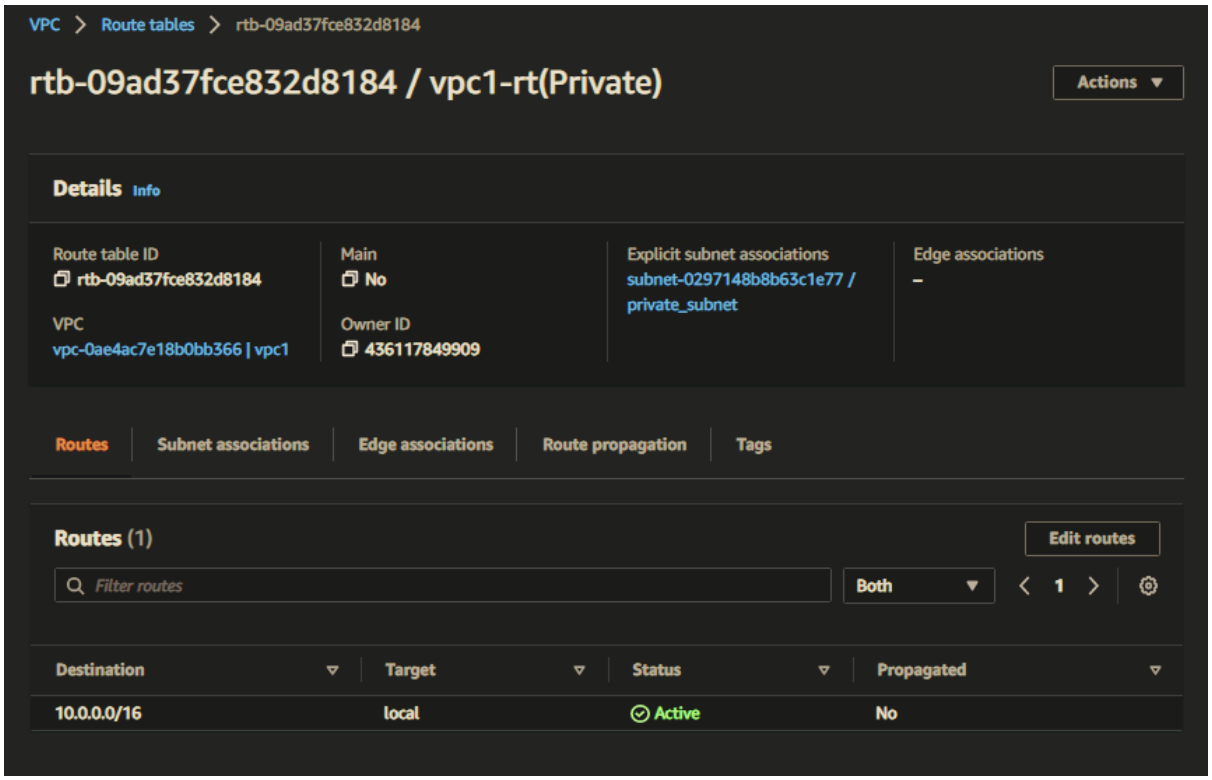
Associate Subnets to the route table by edit subnet association:

- Here public subnet will be attached to the route table with IG such that main route table of vpc1
- And private subnet to the route table with no IG association

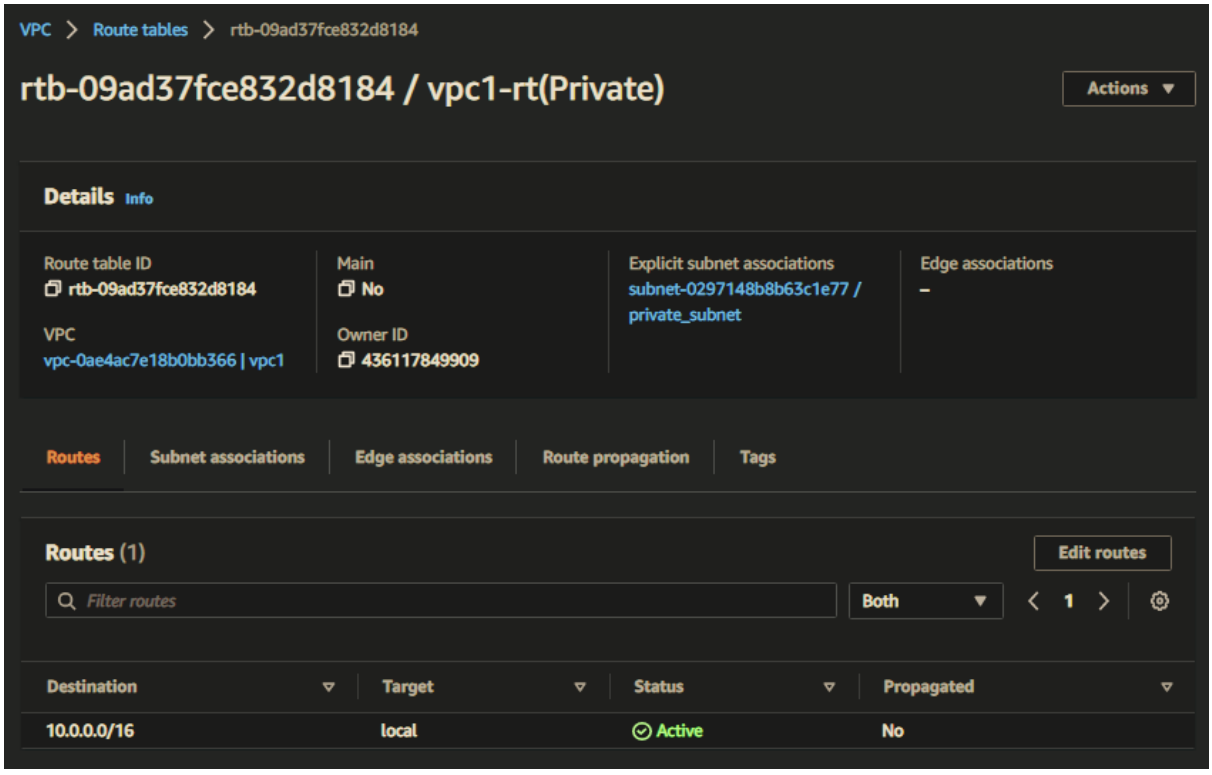
Below image shows the public subnet being associated to the route table with IG (click on the section of explicit subnet association to verify which subnet is attached to the route table)



Similarly attach the private subnet to the route table with no IG associated to it.

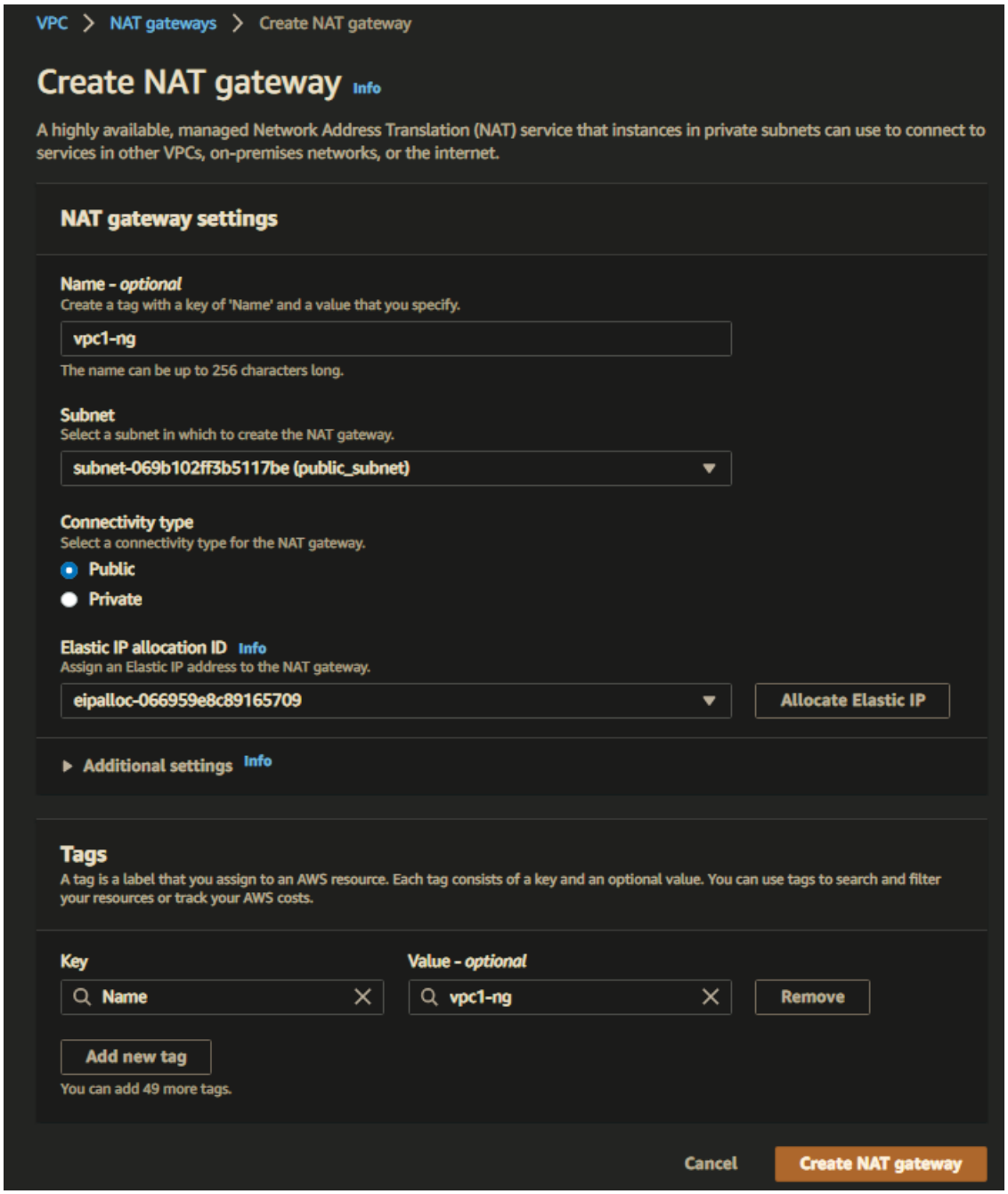


Below image shows your private subnet associated with the route table with no IG associated to it



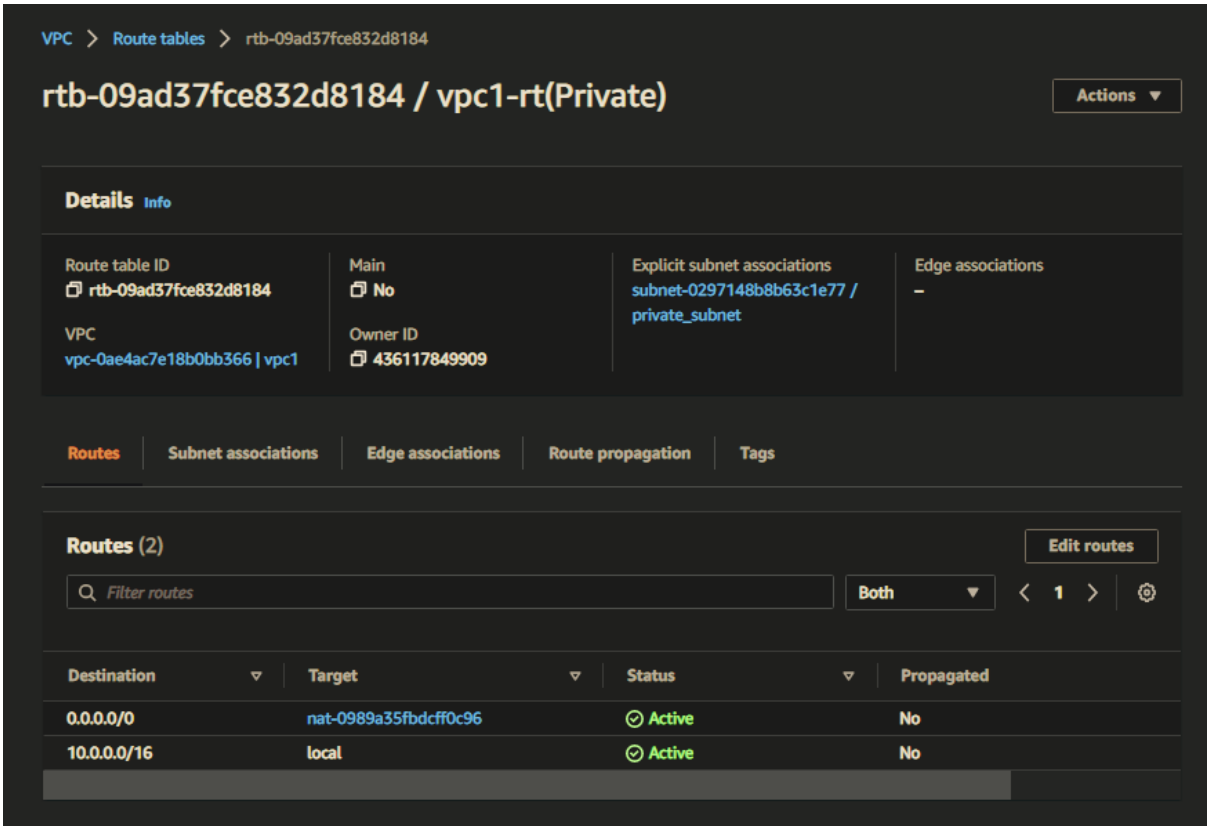
Create a NAT Gateway:

- ⚠️ NAT Gateway is created in a public subnet, hence we select public subnet in the Subnet option shown in figure.



Edit Route table attached to private subnet

- Add NAT gateway route to the route table



Check the resource map you will see the routes you have created

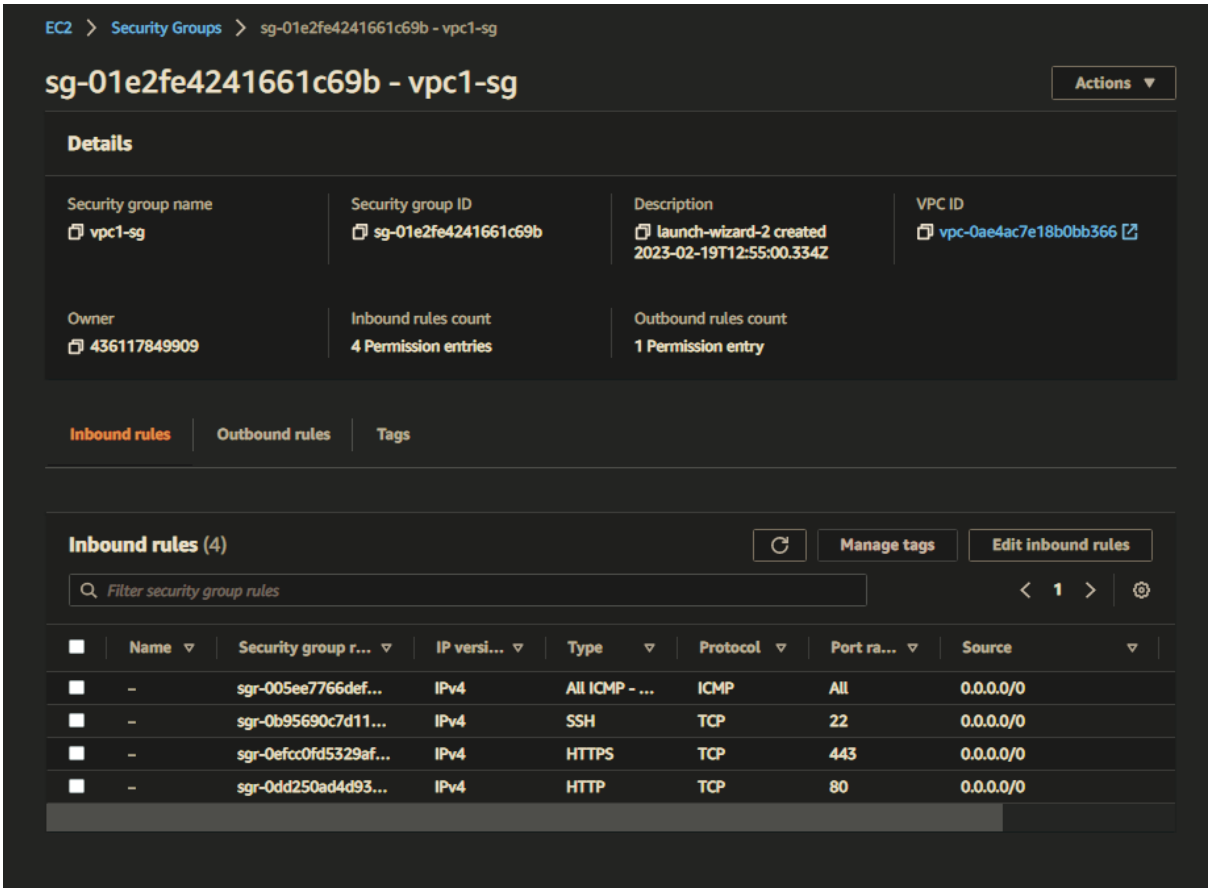
- Below you can see how you have associated your public subnet to the IG and private subnet with the NAT gateway

Create Linux EC2 instance(machine) in the public and private subnets

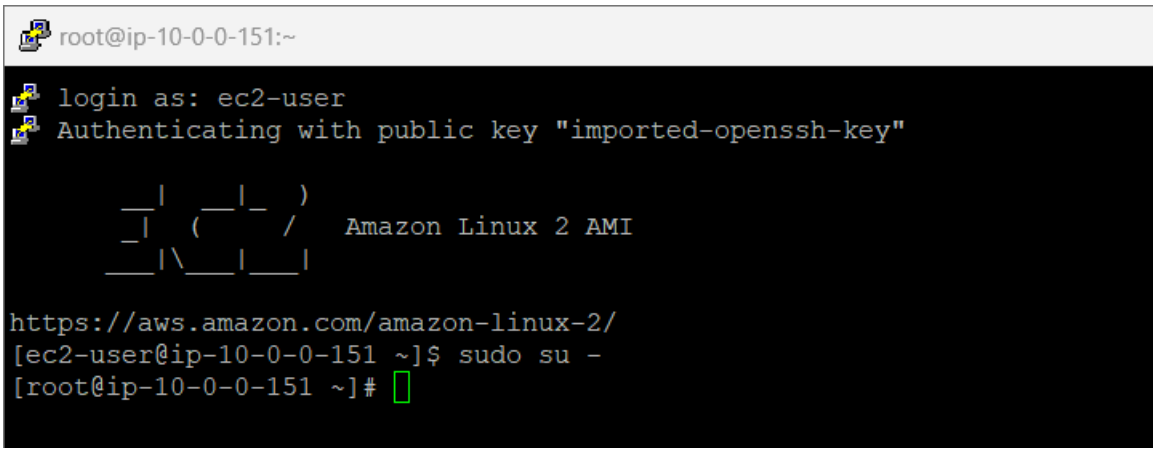
- Edit network settings while creating EC2 instances, select public subnet and enable auto-assign public IP to make ec2 publicly accessible and select private subnet for private ec2 and disable auto-assign public IP. Create common security group vpc1-sg for both instances

Edit the security group of the instances:

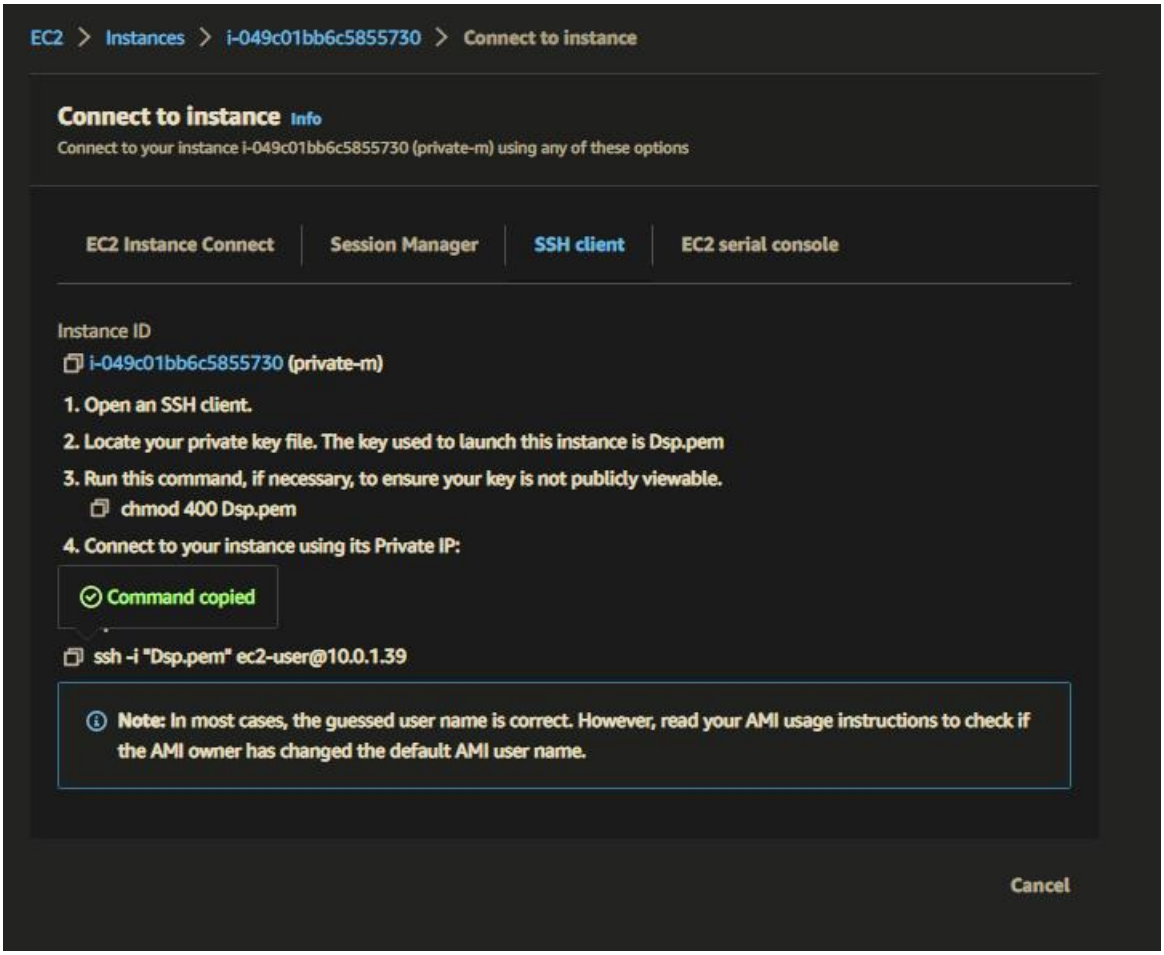
- Add following inbound rules to your Security groups as shown in the image below.



Open Public instance using putty



- enter “sudo su –” command to switch to root user.Now you have to access your private subnet from the public subnet for that go to private Ec2 instance we have created and click on connect and then click on SSH client, you will see the following procedure there



Execute following in your linux CLI :

- “vi Dsp.pem”
- open your .pem file of your key assigned to the instance in Notepad and copy the key
- In CLI press “I” you will enter into insert mode and now paste the copied key.
- Now copy the commands from the connect to instance, as shown in the image above.
- After successful connection you will see the following ,you are now into private instance.

```
[root@ip-10-0-0-151 ~]# chmod 400 Dsp.pem
[root@ip-10-0-0-151 ~]# ssh -i "Dsp.pem" ec2-user@10.0.1.39
The authenticity of host '10.0.1.39 (10.0.1.39)' can't be established.
ECDSA key fingerprint is SHA256:1W8UmaAj950MaecPxCw5MDepBEDP4iVP5WsvdqIoaEk.
ECDSA key fingerprint is MD5:18:f1:6b:21:d3:41:2f:36:b7:0a:f7:3d:34:eb:14:de.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '10.0.1.39' (ECDSA) to the list of known hosts.

       __|  __|_  )
       _|  (     /   Amazon Linux 2 AMI
      ___|\___|___|

https://aws.amazon.com/amazon-linux-2/
[ec2-user@ip-10-0-1-39 ~]$ sudo su -
```

- Type “sudo su –” command and switch to root user
- Now use the command “ping 8.8.8.8” to verify whether the private subnet can access internet, if your private subnet is connected to internet you will see the following output

```
[root@ip-10-0-1-39 ~]# ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=106 time=1.97 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=106 time=1.46 ms
64 bytes from 8.8.8.8: icmp_seq=3 ttl=106 time=1.44 ms
64 bytes from 8.8.8.8: icmp_seq=4 ttl=106 time=1.43 ms
64 bytes from 8.8.8.8: icmp_seq=5 ttl=106 time=1.43 ms
64 bytes from 8.8.8.8: icmp_seq=6 ttl=106 time=1.41 ms
64 bytes from 8.8.8.8: icmp_seq=7 ttl=106 time=1.41 ms
64 bytes from 8.8.8.8: icmp_seq=8 ttl=106 time=1.43 ms
64 bytes from 8.8.8.8: icmp_seq=9 ttl=106 time=1.38 ms
64 bytes from 8.8.8.8: icmp_seq=10 ttl=106 time=1.49 ms
```

- This means NAT gateway is working and now you can access internet from your private subnet.