Creation of VPC and deployment of Public and Private Server.

Steps To Create Custom Network.

1.Create VPC

2.Create IG and Attach to VPC

3.Create Subnets

4.Create Route Table

5.Provide Routing from IG to Route Table

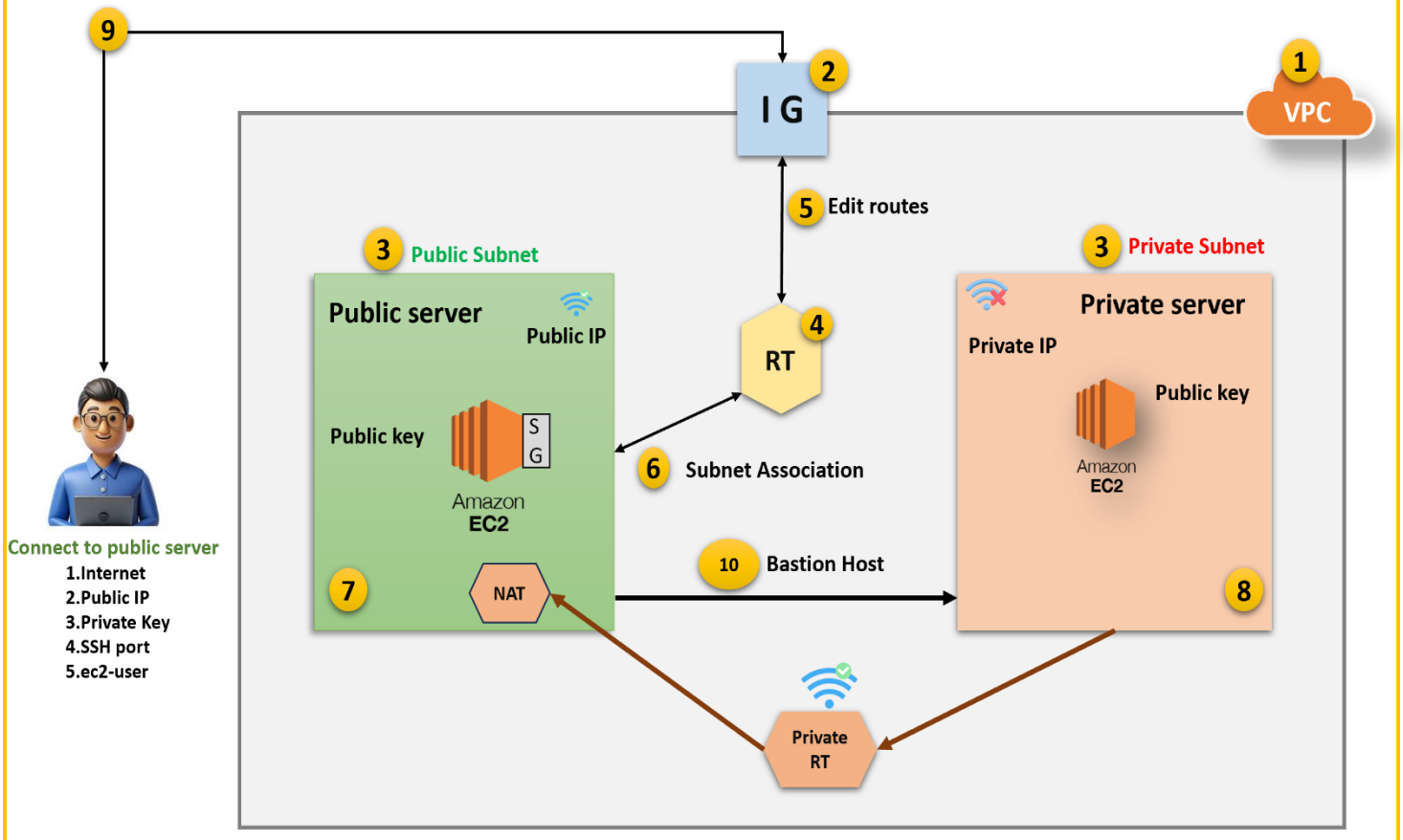
6.Provide Routing from RT to Required Subnet

7.Create Public Server in Public Subnet

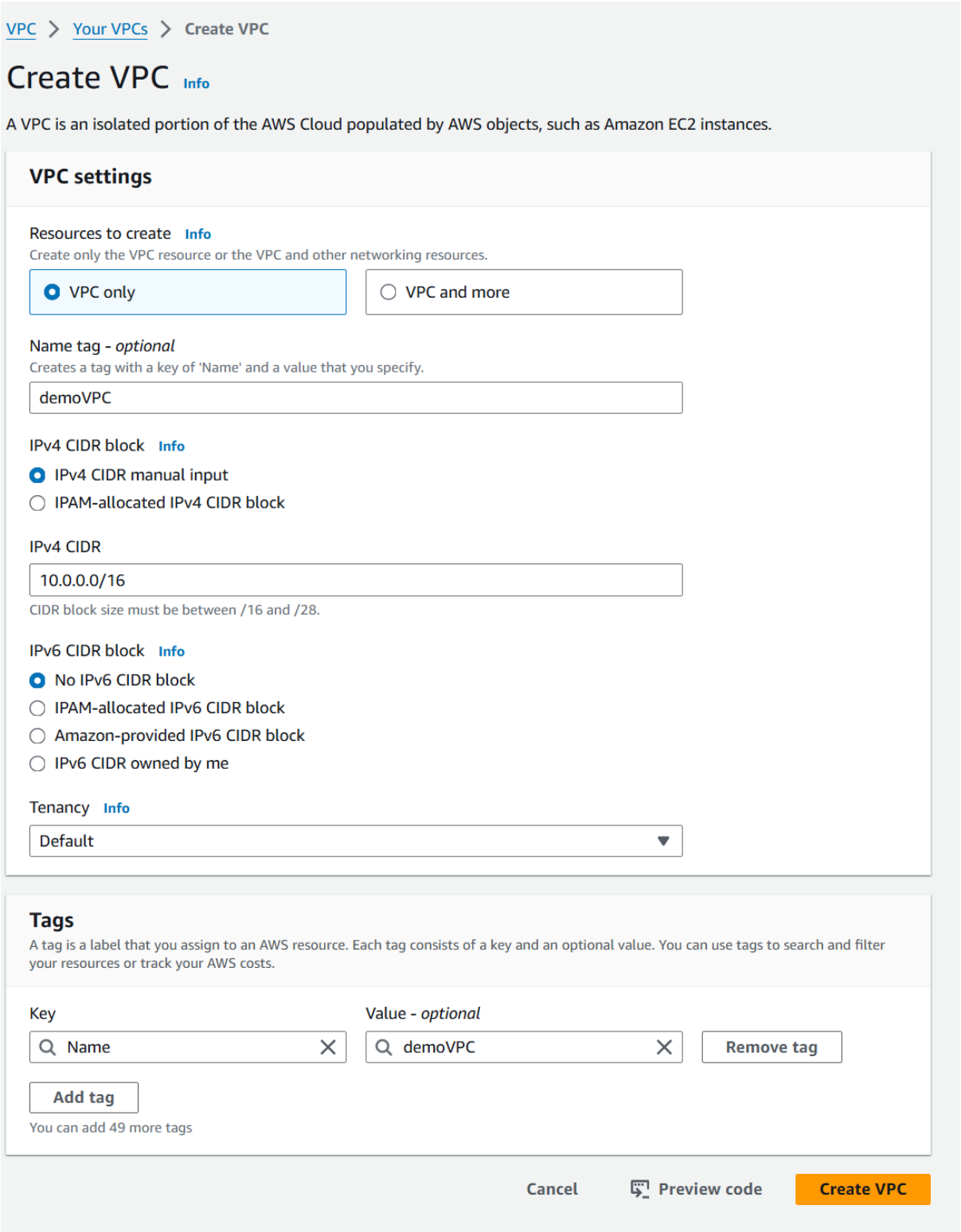
8. Create Private Server in Private Subnet

9.Connect to Public Server

10.Connect to Private Server

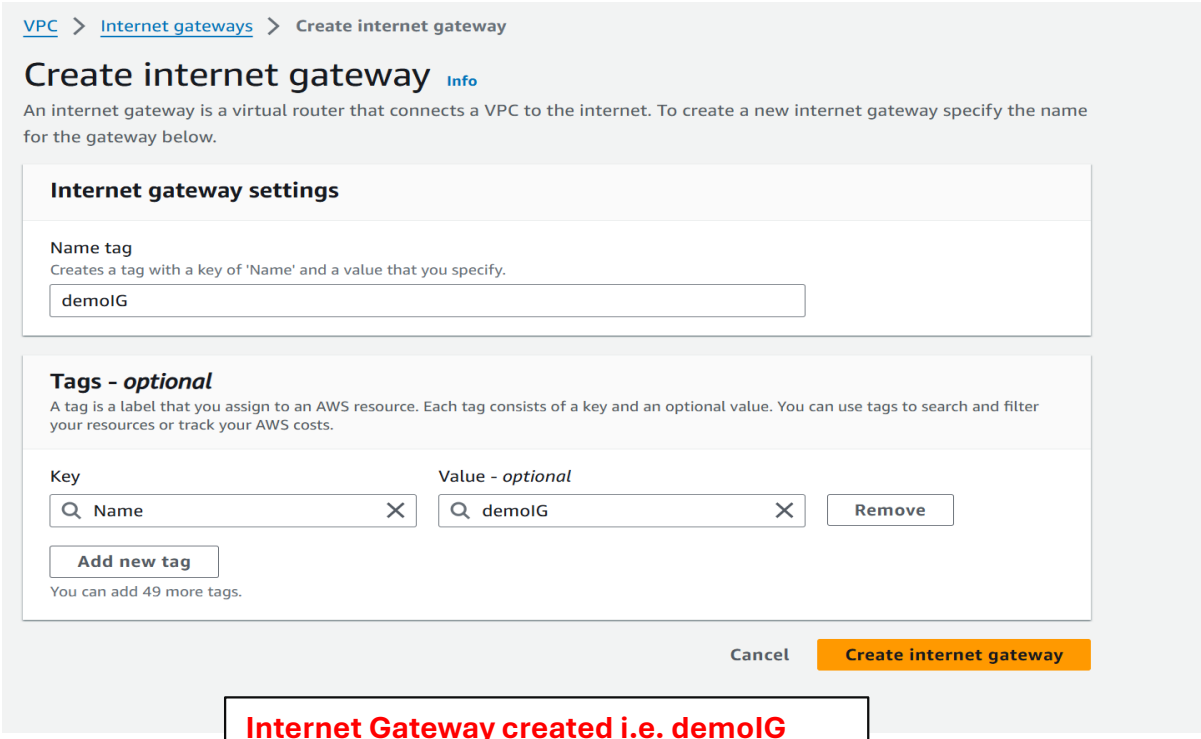


1.Create VPC:

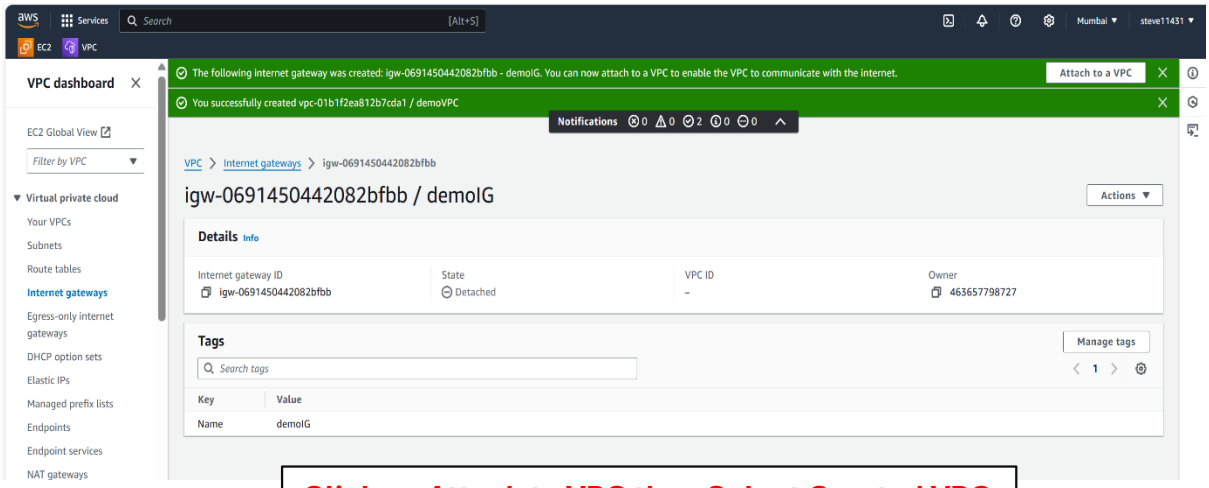


VPC Name = DemoVPC **IPV4 CIDR = 10.0.0.0/16**

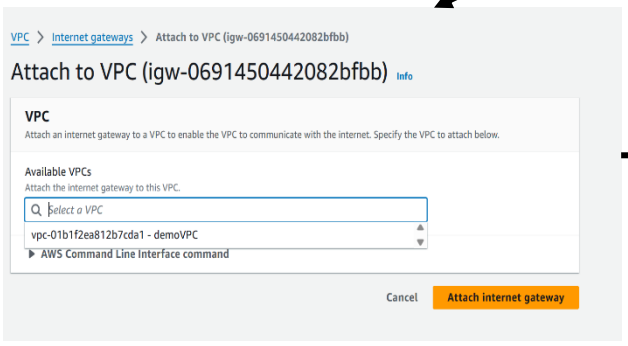
2.Create IG and Attach to VPC:

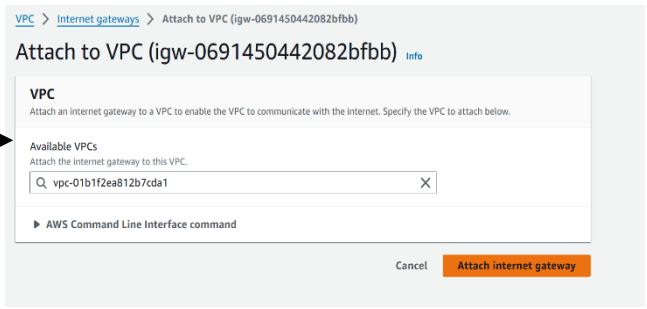


Internet Gateway created i.e. demoIG

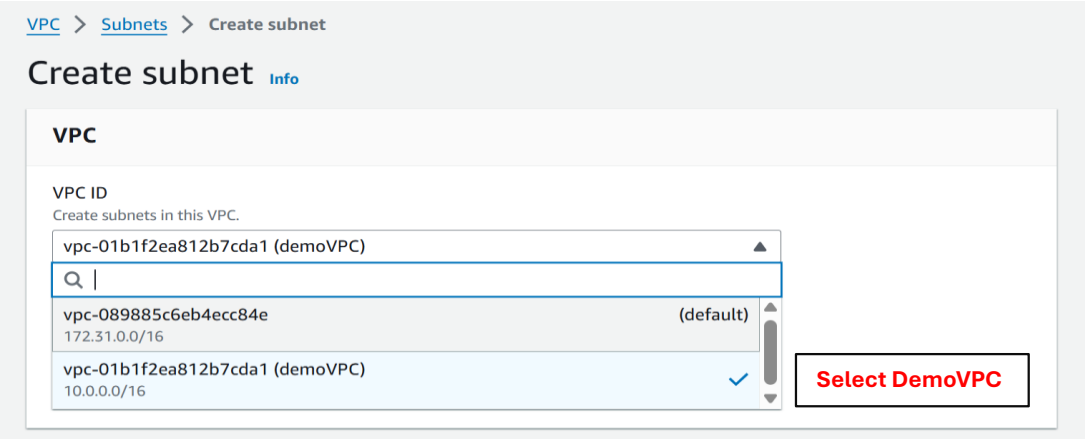


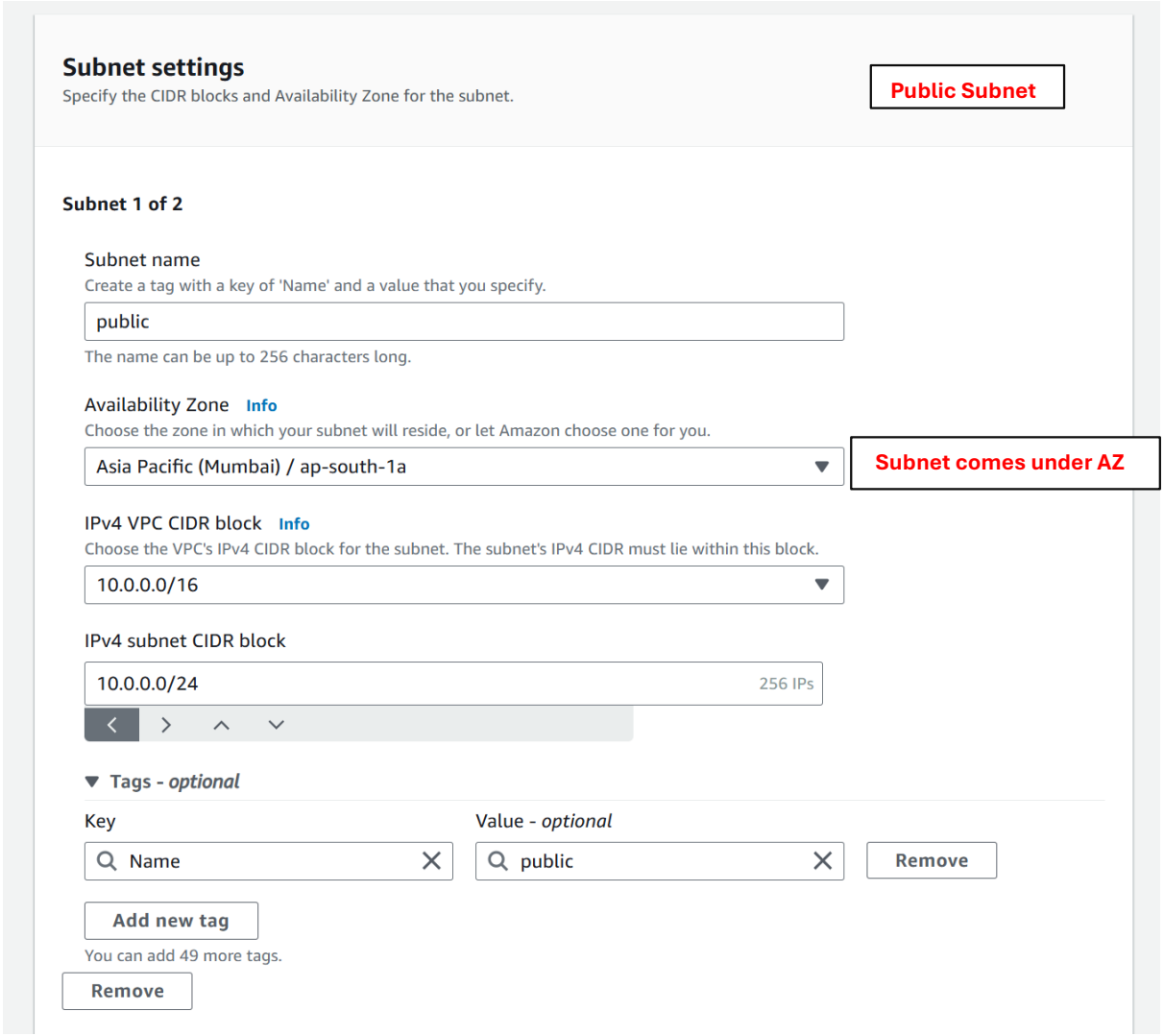
Click on Attach to VPC then Select Created VPC





3.Create Subnets





Click on Add Another Subnet i.e. Private subnet.

Subnet 2 of 2

Private Subnet

Subnet name
Create a tag with a key of 'Name' and a value that you specify.
private
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.
Asia Pacific (Mumbai) / ap-south-1a

IPv4 VPC CIDR block Info
Choose the VPC's IPv4 CIDR block for the subnet. The subnet's IPv4 CIDR must lie within this block.
10.0.0.0/16

IPv4 subnet CIDR block
10.0.1.0/24 256 IPs

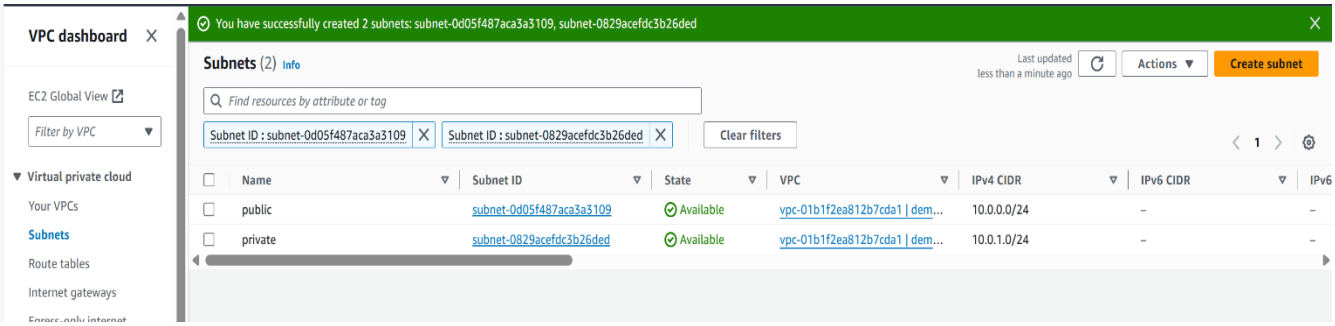
Tags - *optional*

Key	Value - *optional*	
Name	private	Remove

Add new tag
You can add 49 more tags.
Remove

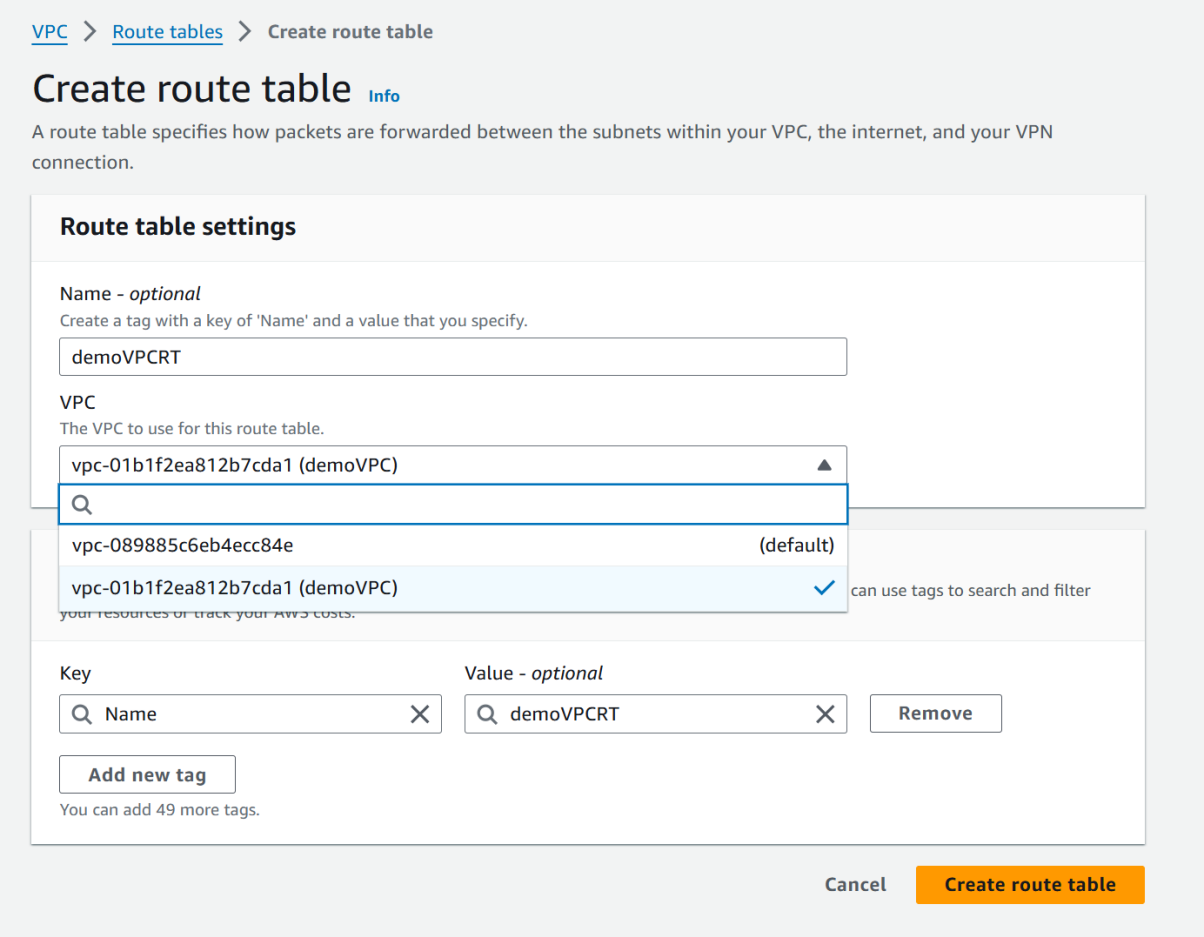
Add new subnet

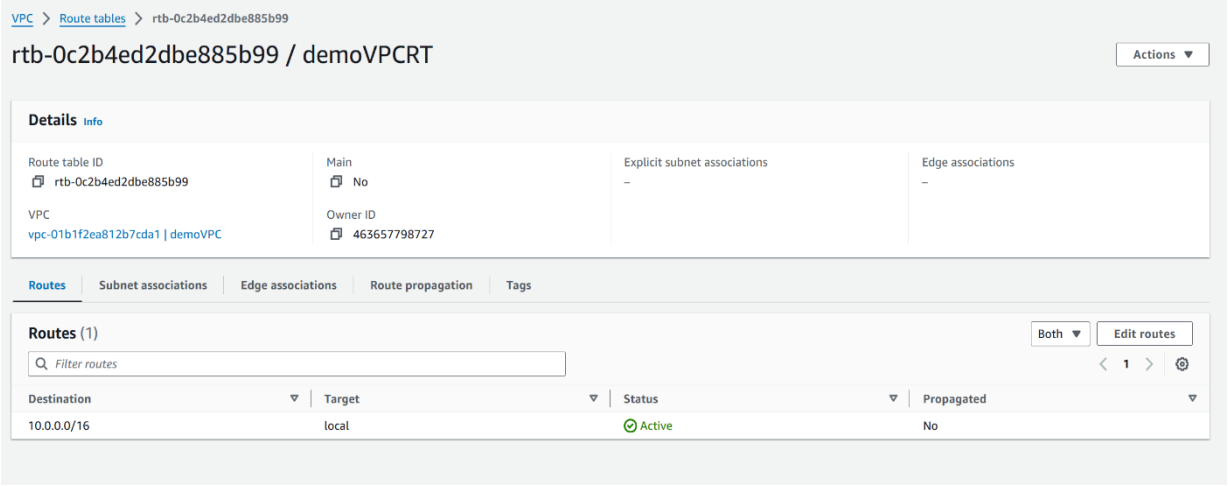
Cancel **Create subnet**



Public Subnet & Private Subnet created Successfully.

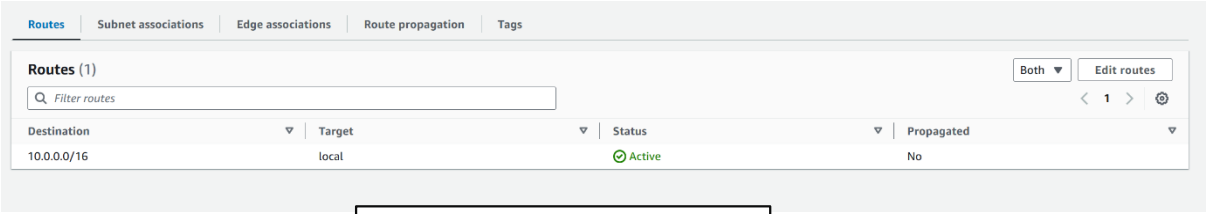
4.Create Route Table:



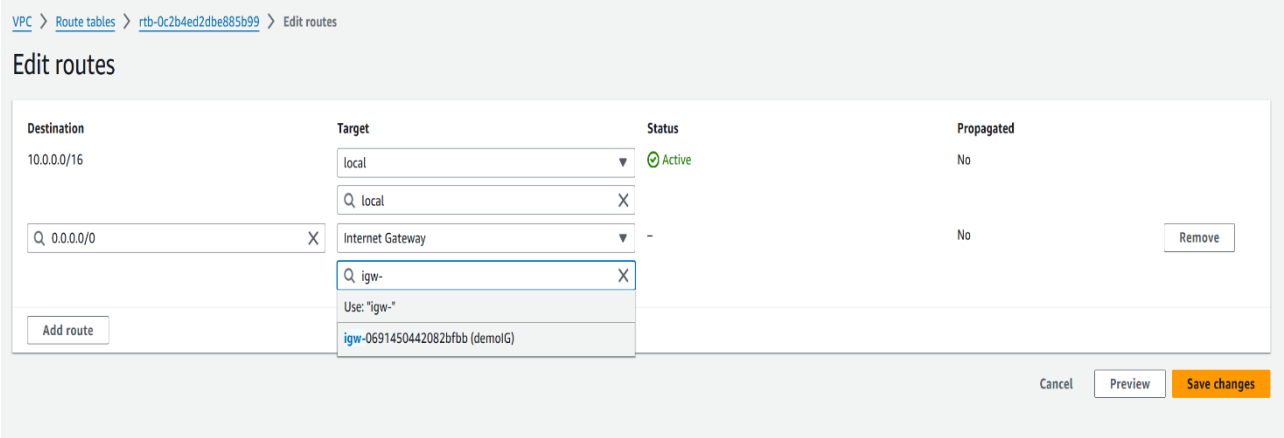


Public Route Table created Successfully I.e. demoVPCRT

5.Provide Routing from IG to Route Table:



Click on Edit Routes Option

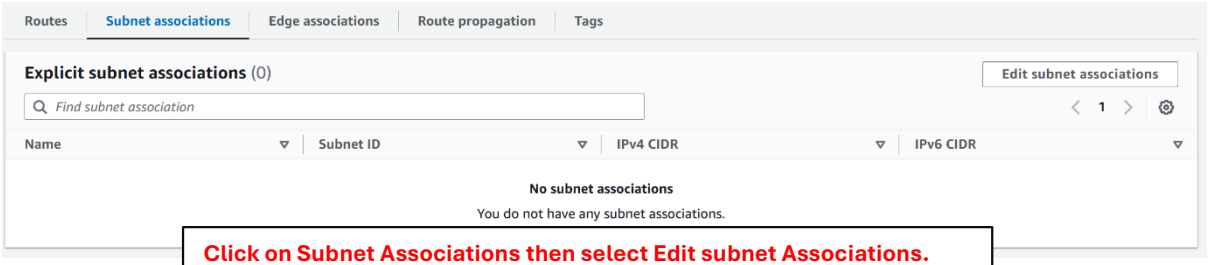


Click on Add Route

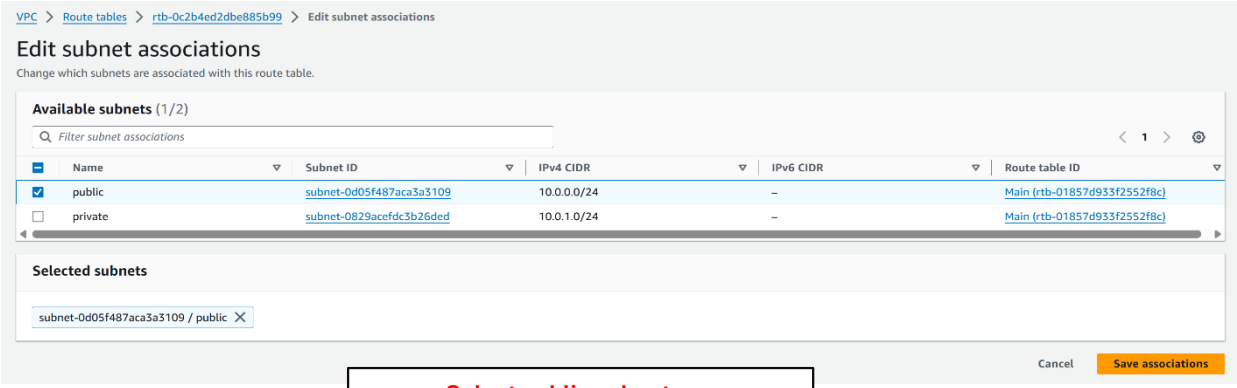
0.0.0.0/0 =Everyone can connect

Select Internet Gateway i.e. demoIG

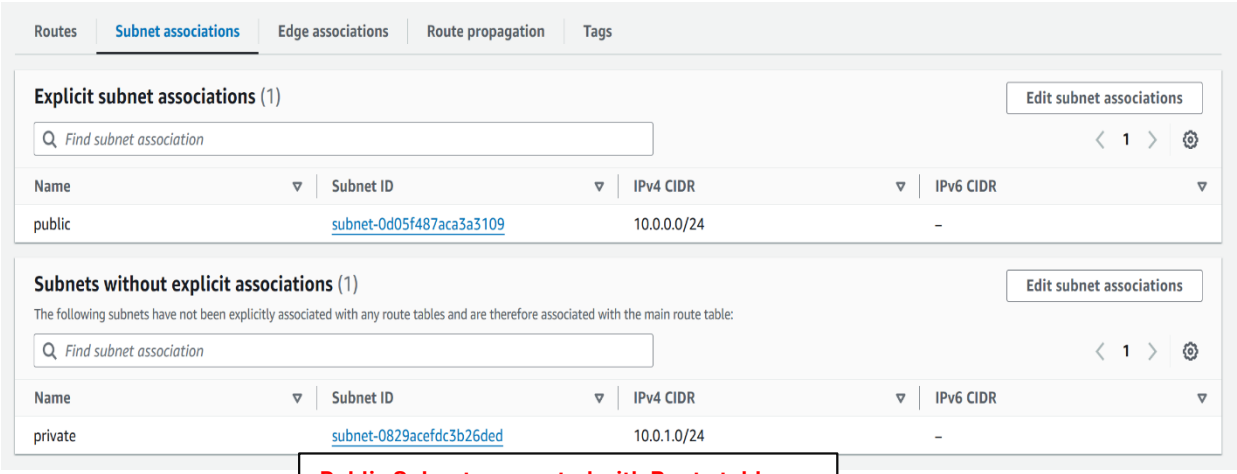
6.Provide Routing from RT to Required Subnet:



Click on Subnet Associations then select Edit subnet Associations.

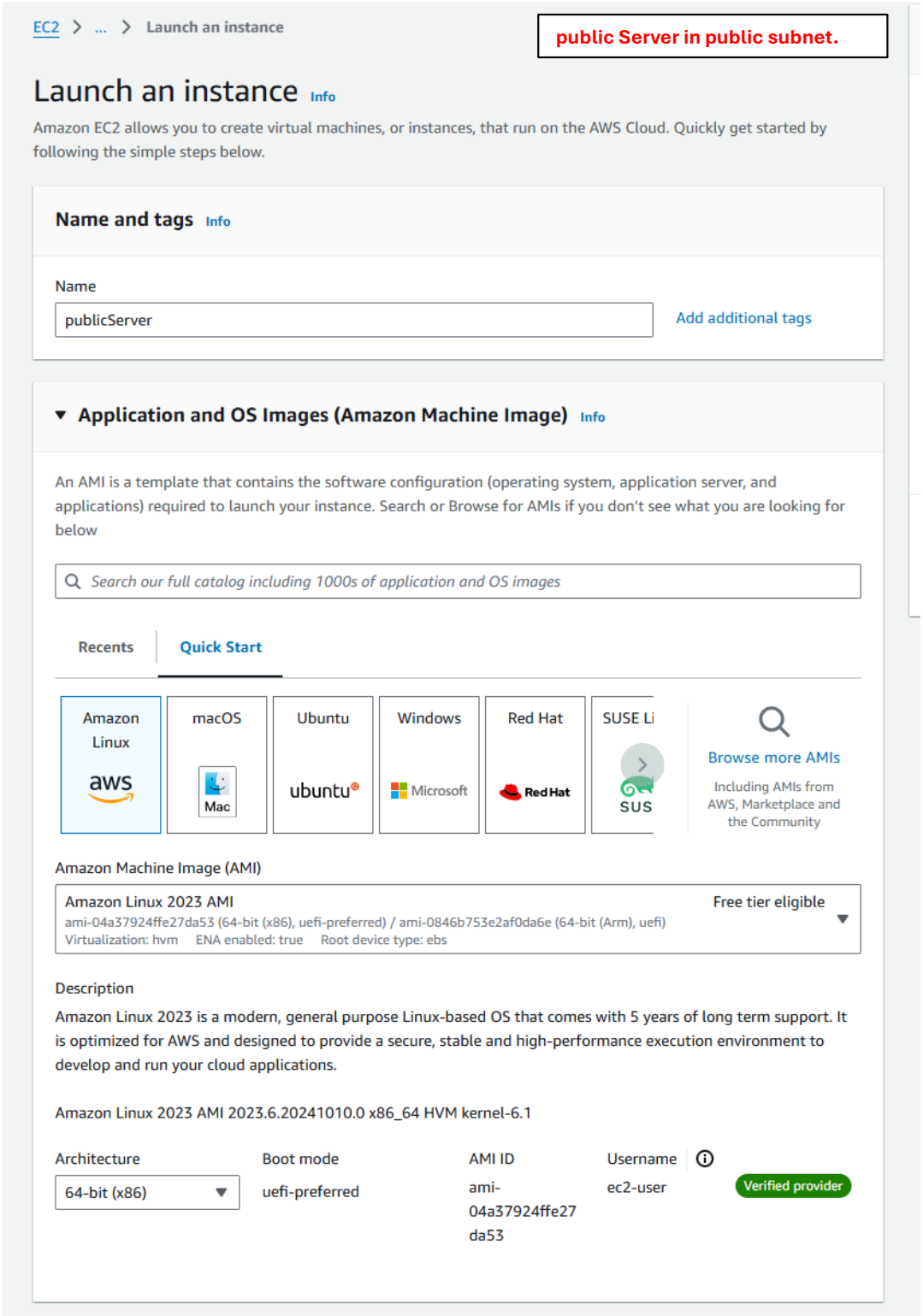


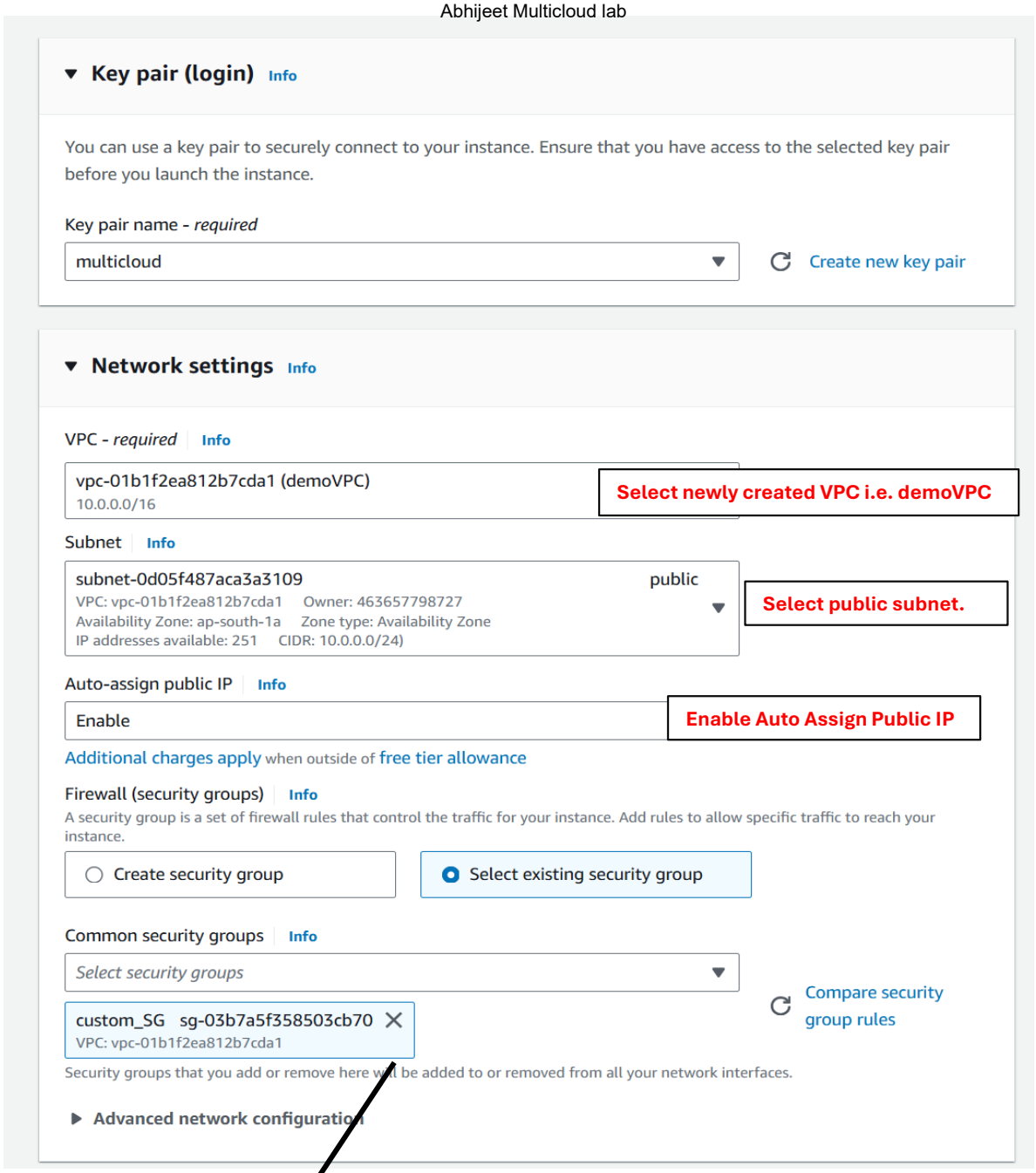
Select public subnet

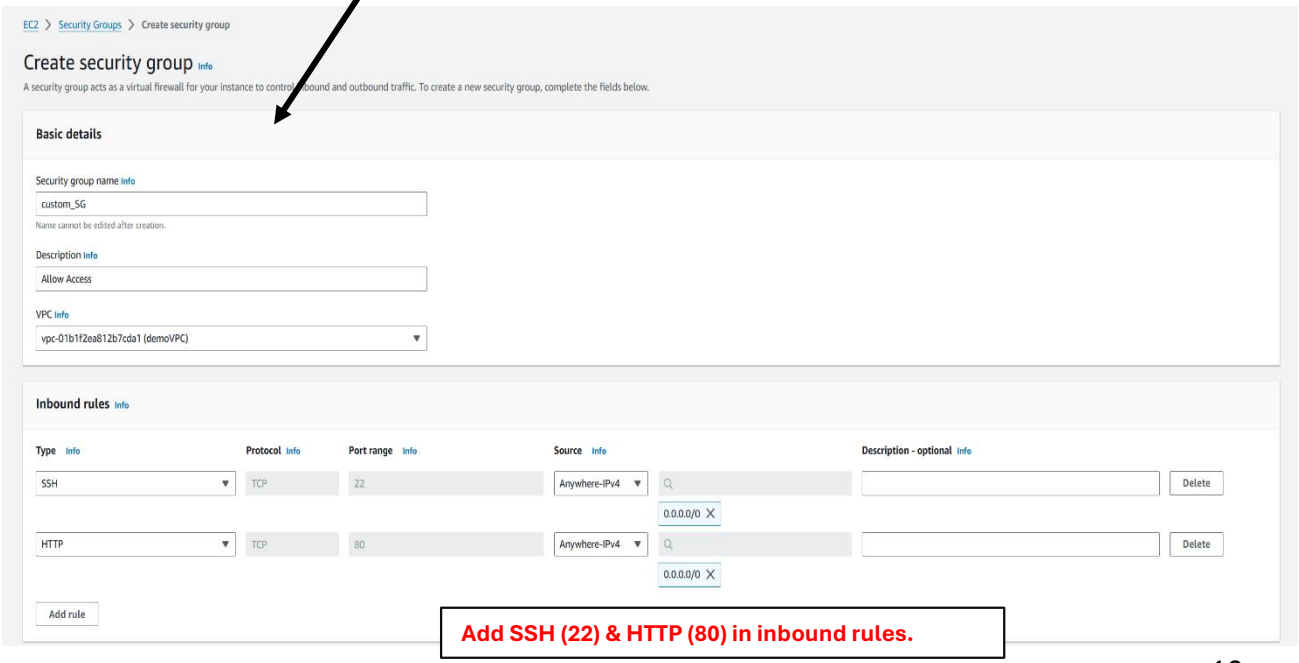


Public Subnet connected with Route table.

7.Create Public Server in Public Subnet:

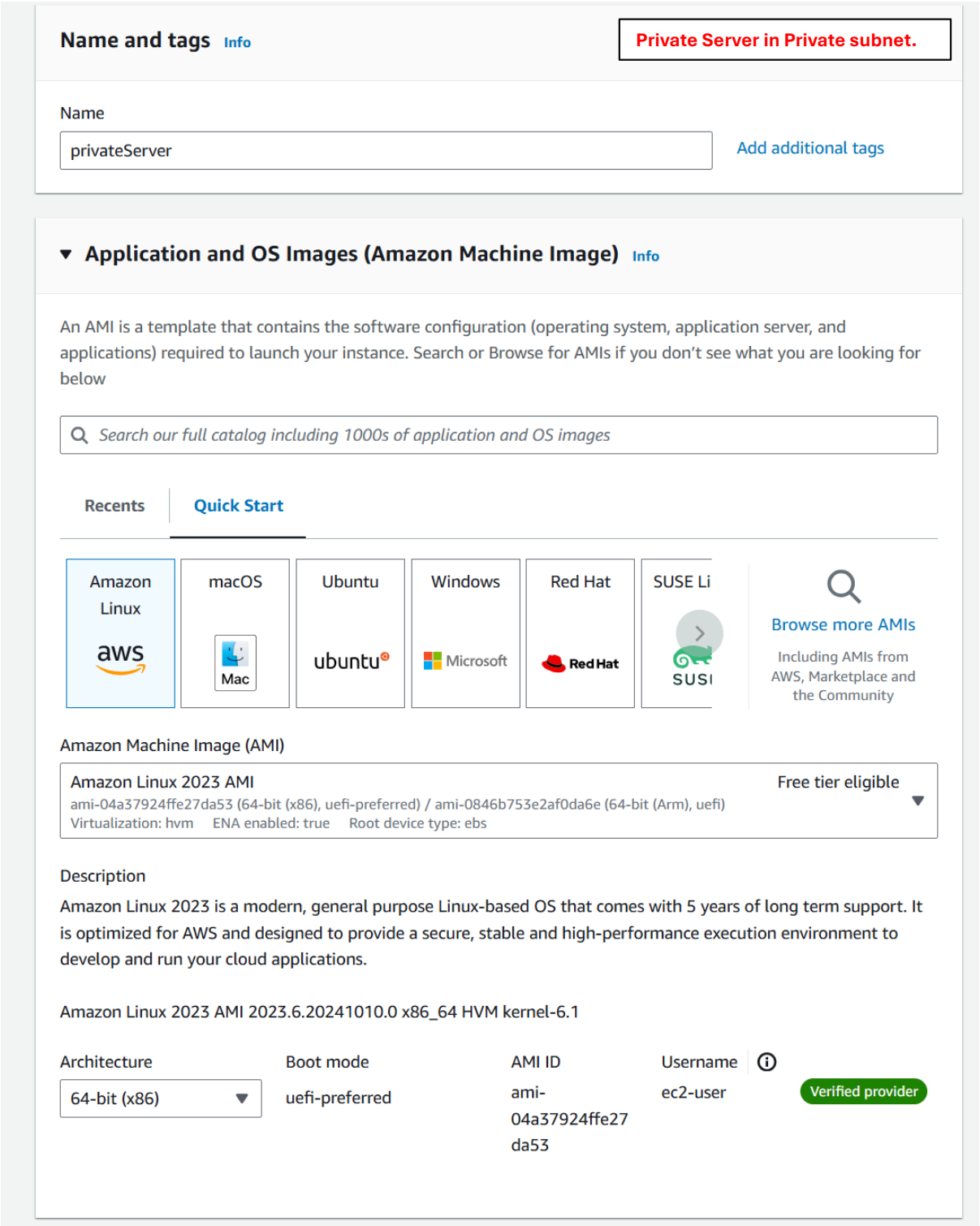






Add SSH (22) & HTTP (80) in inbound rules.

8. Create Private Server in Private Subnet:



Instance type Info | Get advice

Instance type

t2.micro — Free tier eligible
Family: t2 1 vCPU 1 GiB Memory Current generation: true
On-Demand Linux base pricing: 0.0124 USD per Hour
On-Demand Windows base pricing: 0.017 USD per Hour
On-Demand RHEL base pricing: 0.0268 USD per Hour
On-Demand Ubuntu Pro base pricing: 0.0142 USD per Hour
On-Demand SUSE base pricing: 0.0124 USD per Hour

All generations

Compare instance types

Additional costs apply for AMIs with pre-installed software

Key pair (login) Info

You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

Key pair name - *required*

multicloud

Create new key pair

Network settings Info

VPC - *required* Info

vpc-01b1f2ea812b7cda1 (demoVPC)
10.0.0.0/16

Subnet Info

subnet-0829acefdc3b26ded — private
VPC: vpc-01b1f2ea812b7cda1 Owner: 463657798727
Availability Zone: ap-south-1a Zone type: Availability Zone
IP addresses available: 251 CIDR: 10.0.1.0/24)

Select private subnet

Auto-assign public IP Info

Disable

Disable Auto Assign Public IP

Firewall (security groups) Info

A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

○ Create security group ◉ Select existing security group

Common security groups Info

Select security groups

custom_SG sg-03b7a5f358503cb70
VPC: vpc-01b1f2ea812b7cda1

Compare security group rules

Security groups that you add or remove here will be added to or removed from all your network interfaces.

▶ Advanced network configuration

Instances (2) Info

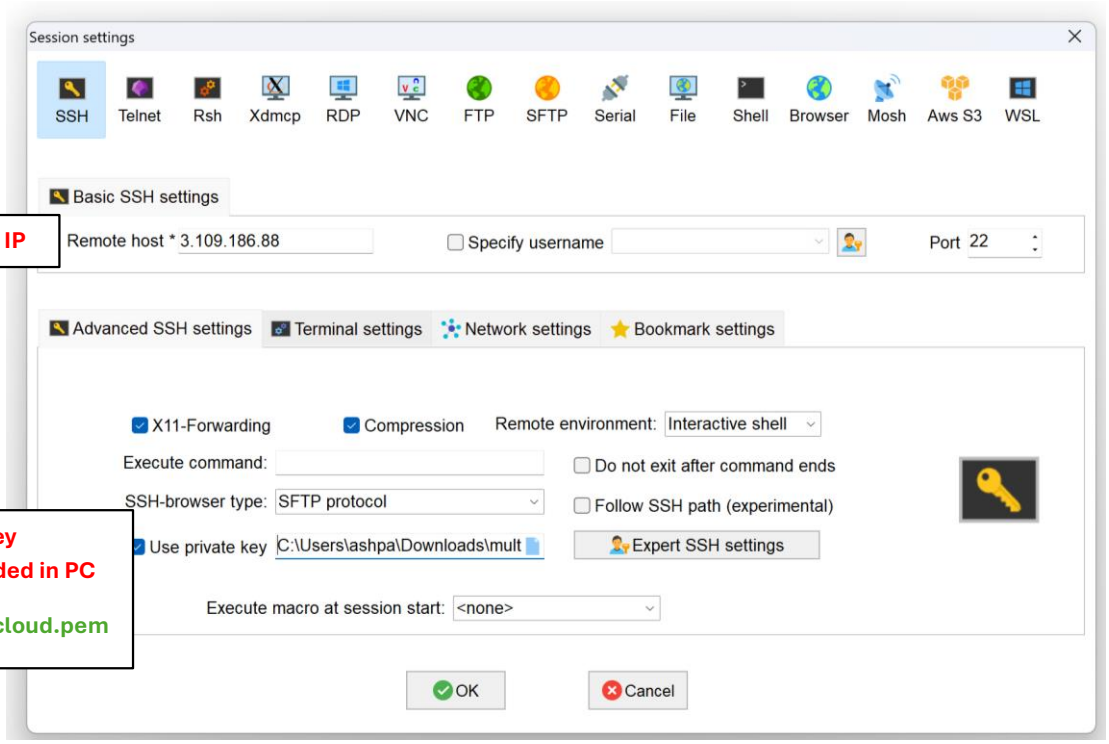
Last updated less than a minute ago | Connect | Instance state | Actions | Launch instances

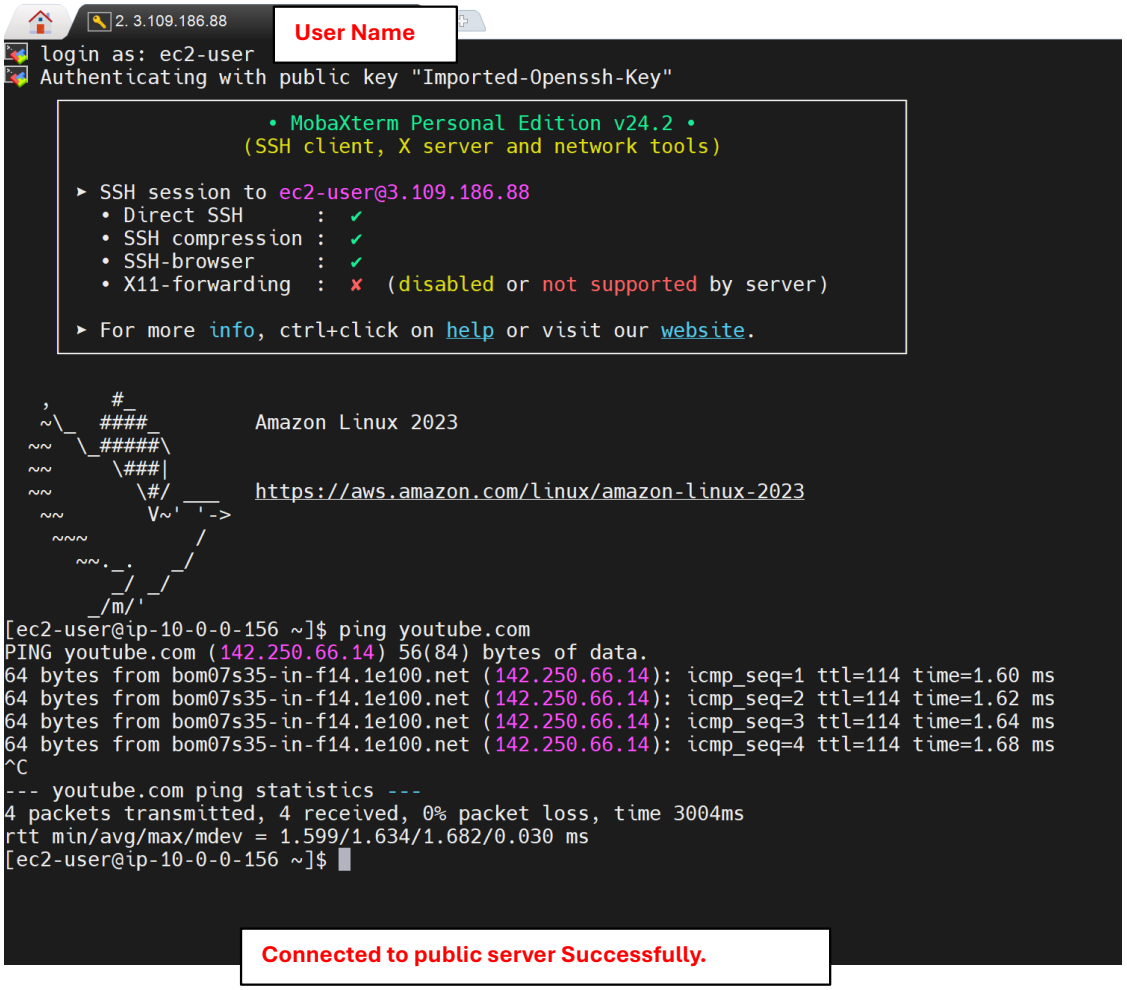
Find Instance by attribute or tag (case-sensitive) | All states | 1

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4 ...	Elastic IP
privateServer	i-0f7005b0deb25dd13	Running	t2.micro	Initializing	View alarms +	ap-south-1a	-	-	-
publicServer	i-043ed69b3fba5b91c	Running	t2.micro	2/2 checks passed	View alarms +	ap-south-1a	-	3.109.186.88	-

Public and private servers created successfully.

9.Connect to Public Server (using Mobaxterm):





10.Connect to Private Server with the help of Public Server:

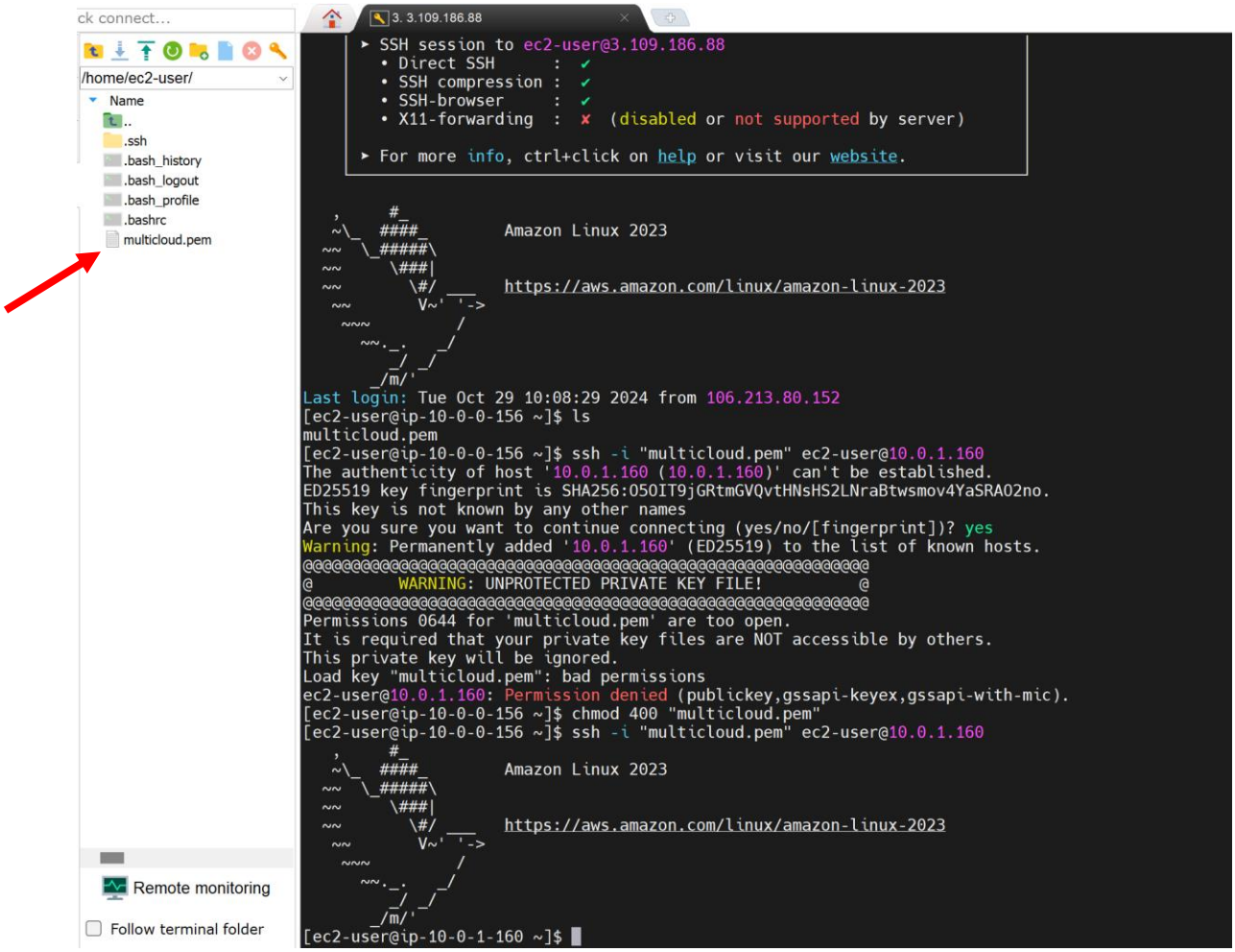
(Bastion Host)

Private IP - 10.0.1.160

Private Key- multicloude.pem

SSH -SG

Connecting Private server by using upload Private key option in mobaxterm.



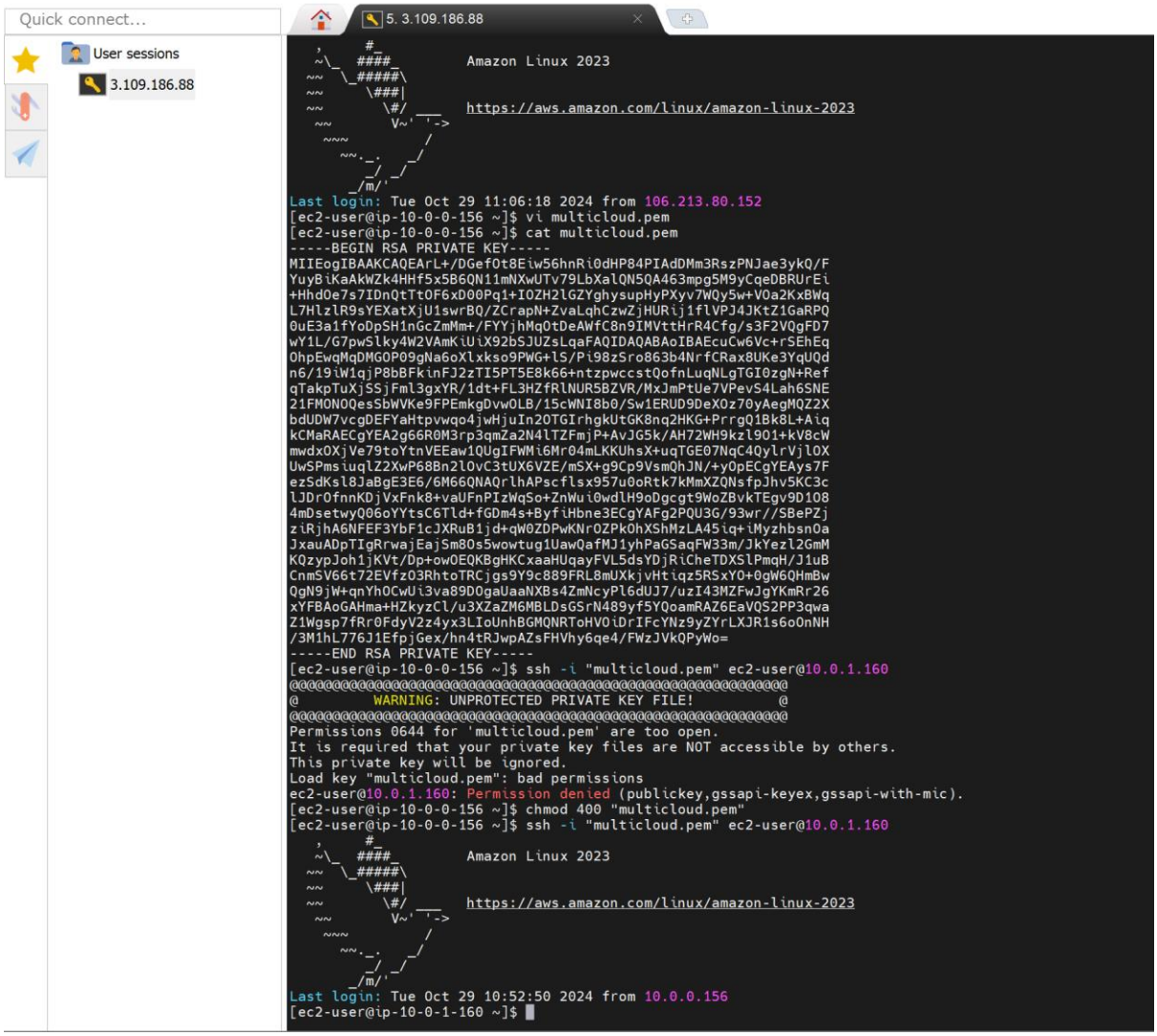
Commands used:

ls

chmod 400 "multicloud.pem"

ssh -I "multicloud.pem" ec2-user@10.0.1.160

By using vi command (create file inside and add content):



Commands used:

Vi multicloude.pem

Wq!

Cat multicloude.pem

chmod 400 “multicloud.pem”

ssh -i “multicloud.pem” ec2-user@10.0.1.160

Steps To Provide Internet to Private Server Using NAT Gateway.

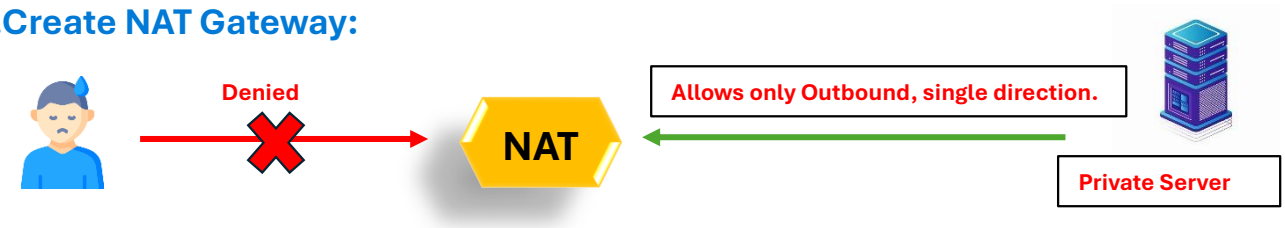
1.Create NAT Gateway

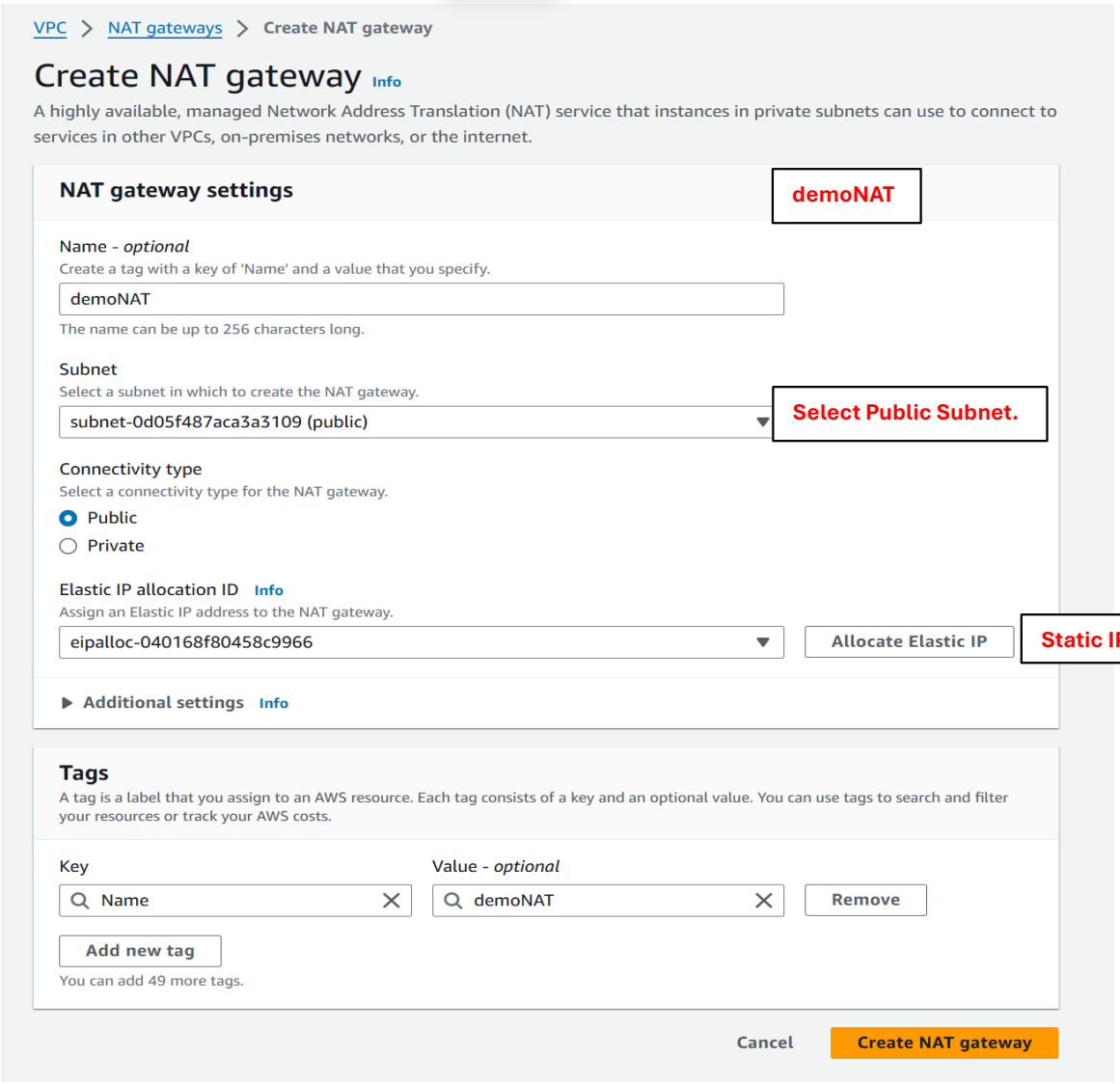
2.Create Route Table

3.Edit Routes (Provide Path from NAT Gateway to Private RT)

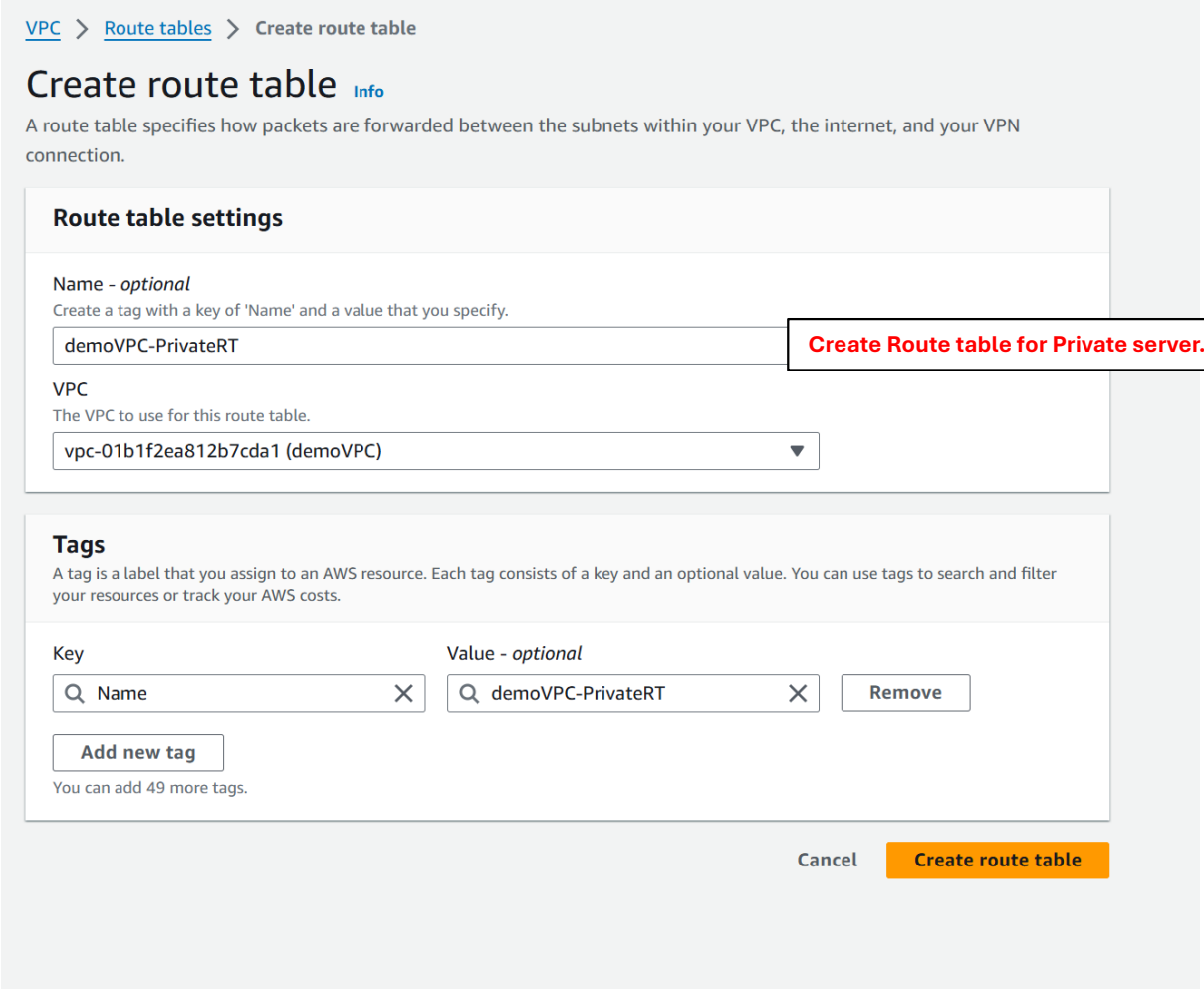
4.Provide Path from Private RT to Private Subnet

1.Create NAT Gateway:

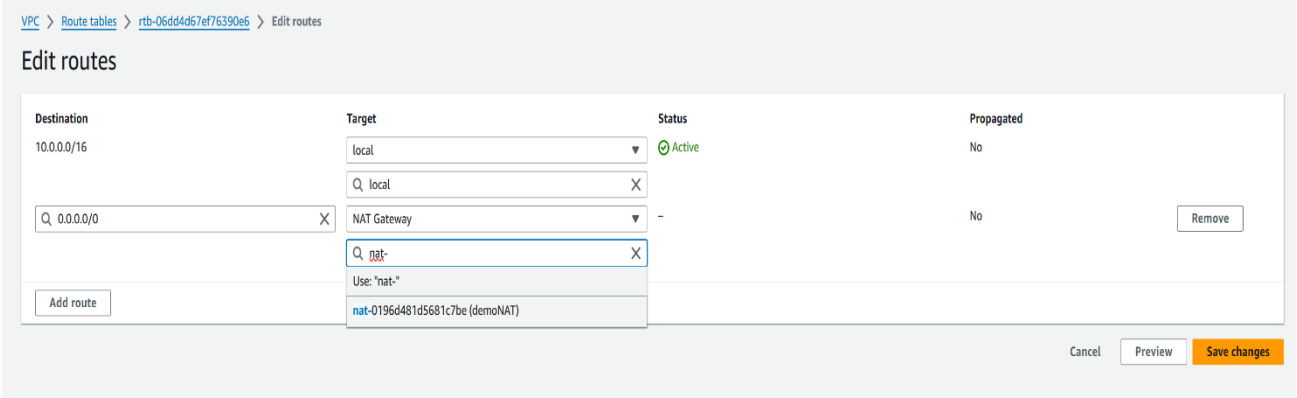


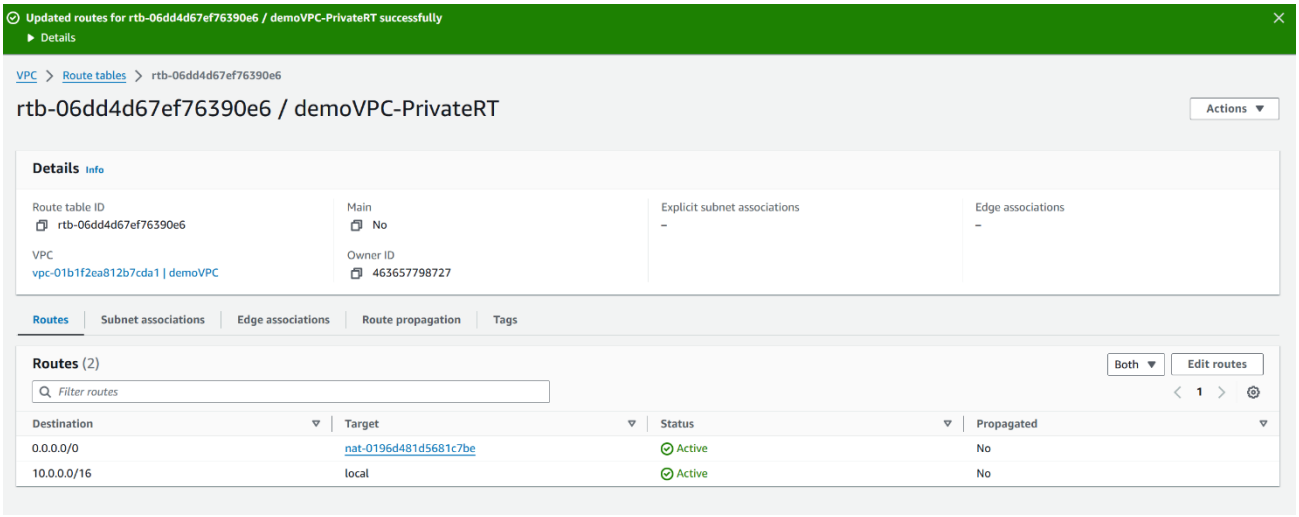


2.Create Route Table:



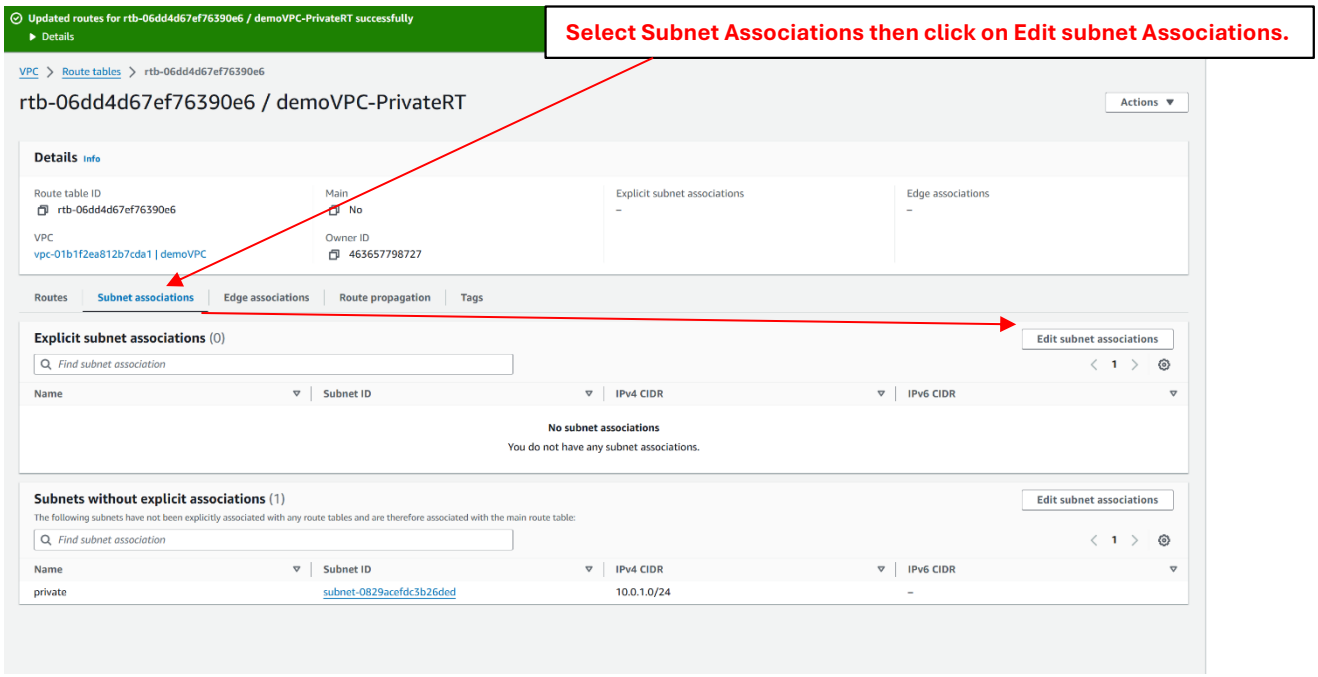
3.Edit Routes (Provide Path from NAT Gateway to Private RT):

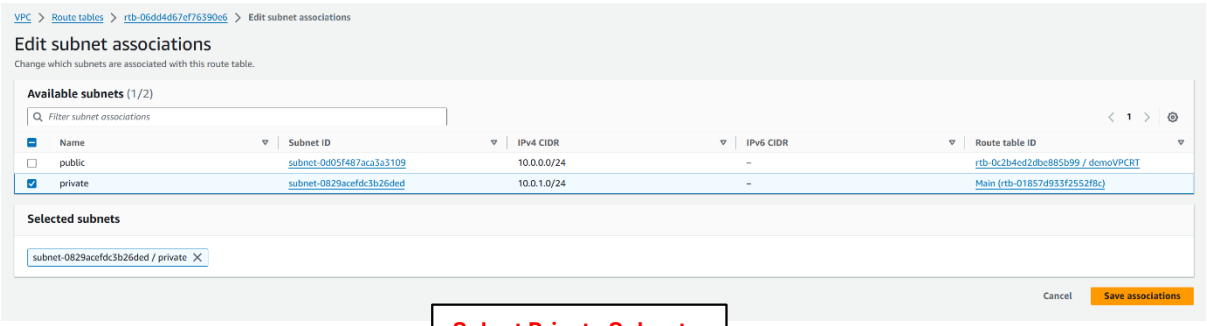




Path from NAT Gateway to Private RT Created Successfully.

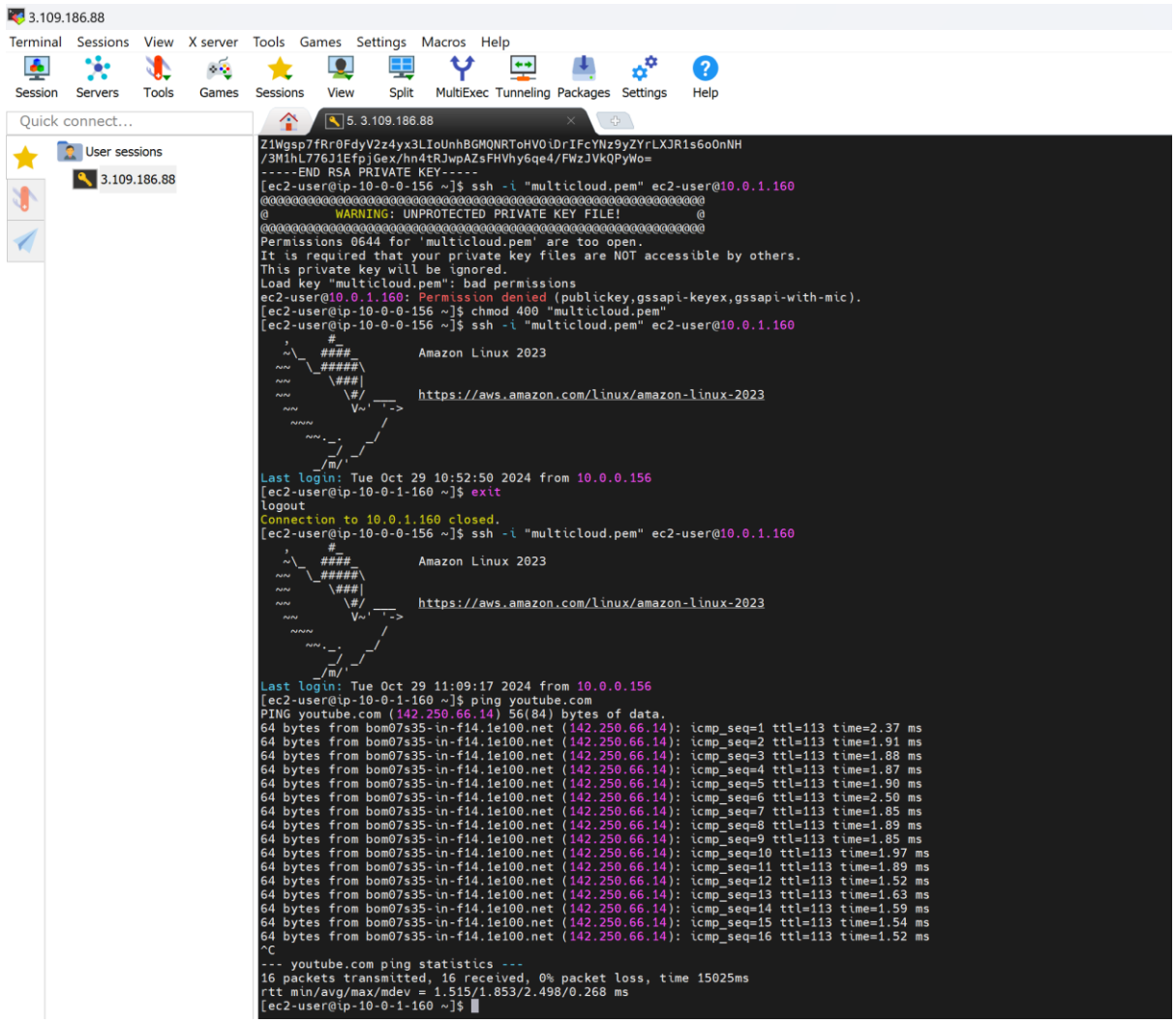
4.Provide Path from Private RT to Private Subnet:



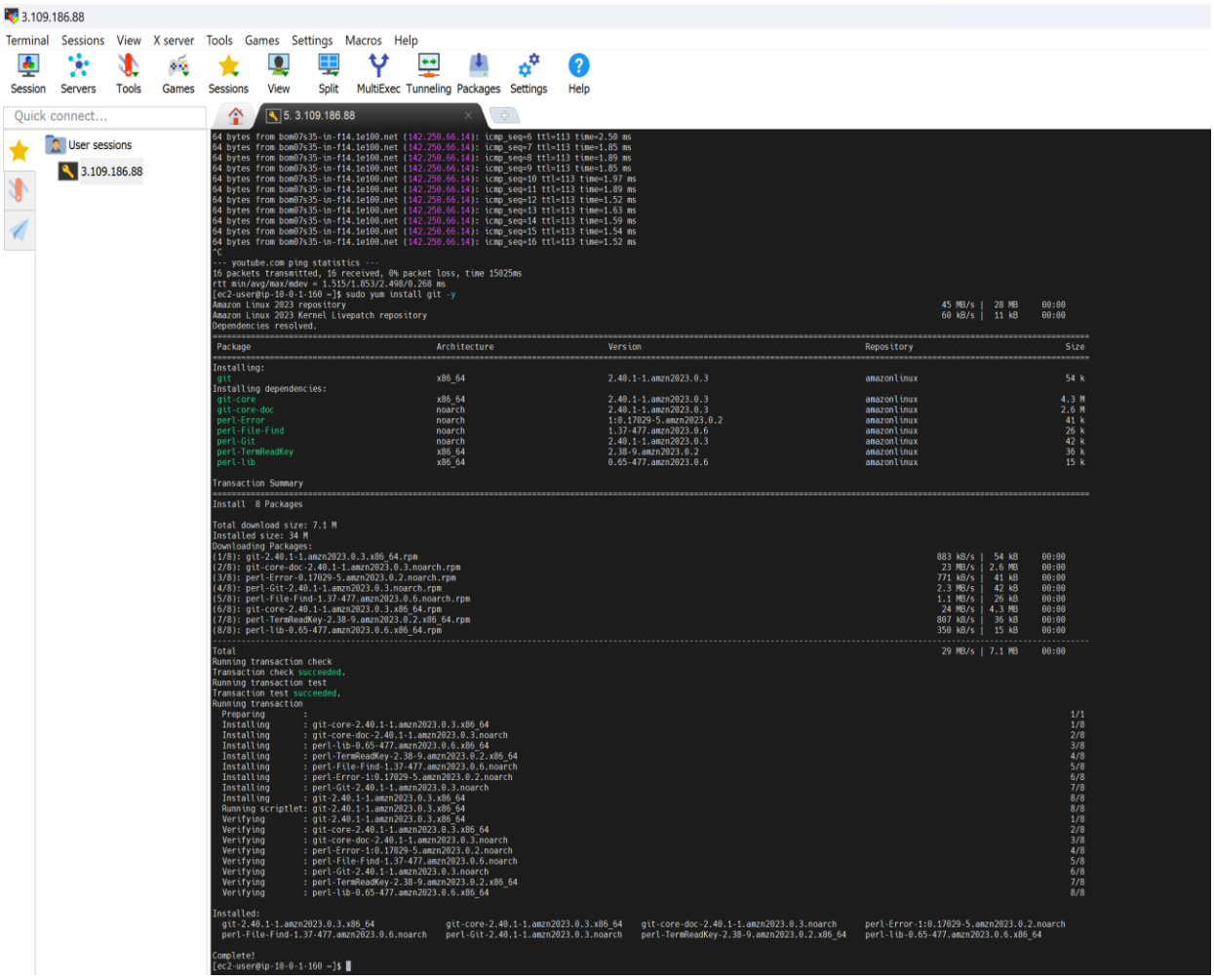


Select Private Subnet

Providing Internet to The Private Server with The Help of NAT Gateway.



Ping youtube.com Successfully through private server.



Installed git Successfully in Private Server.