Step-by-Step Guide to Creating an AWS VPC

Bastion Host Setup or Jump Server Setup

A VPC, or Virtual Private Cloud, is a logically isolated virtual network within the AWS Cloud where you can launch your AWS resources. It provides complete control over your virtual networking environment, including your own IP address range, subnets, route tables, and network gateways.

Prerequisites

- AWS Account.
- Basic understanding of networking concepts like ip and CIDR.

Learning Goals 🚀

- Creating the VPC by using VPC only option.
- The typical VPC setup creation and testing
    - **Public Subnet** → EC2 that can be reached from the internet (e.g. web server, bastion host, load balancer).The EC2 has Elastic Ip.
    - **Private Subnet** → EC2 that cannot be reached directly from the internet(e.g. database, application backend, cache).
- To reach private subnet, first SSH into public subnet than try SSH into the private subnet.

Step 1: Create a VPC

- Create a VPC -> Vpc only -> Name
- **Ipv4 CIDR block**
    - *IPv4 CIDR block* - Default and Manual selection of non-overlapping ip ✓
    - *IPv4 CIDR manual inputIPAM-allocated IPv4 CIDR block* - **IP Address Manager (IPAM)** to automatically assign a CIDR block to your VPC. Instead of you picking a range, IPAM pulls an available CIDR block from a predefined pool. This prevents IP address conflicts, ensures compliance with your organization's networking rules, and automates the process, making it much easier to scale in large, multi-account environments.
- **Ipv4 CIDR**
    - Here I used the Ip range between `10.0.0.0/28` (16 Ip's).
    - Vpc ip are working with RFC 1918.

RFC 1918

RFC 1918 is a standard that defines private IPv4 address ranges that are not routable over the public internet. They're meant for internal/private networks.

The three ranges are:

- `10.0.0.0 – 10.255.255.255 (CIDR: 10.0.0.0/8)`
- `172.16.0.0 – 172.31.255.255 (CIDR: 172.16.0.0/12)`
- `192.168.0.0 – 192.168.255.255 (CIDR: 192.168.0.0/16)`

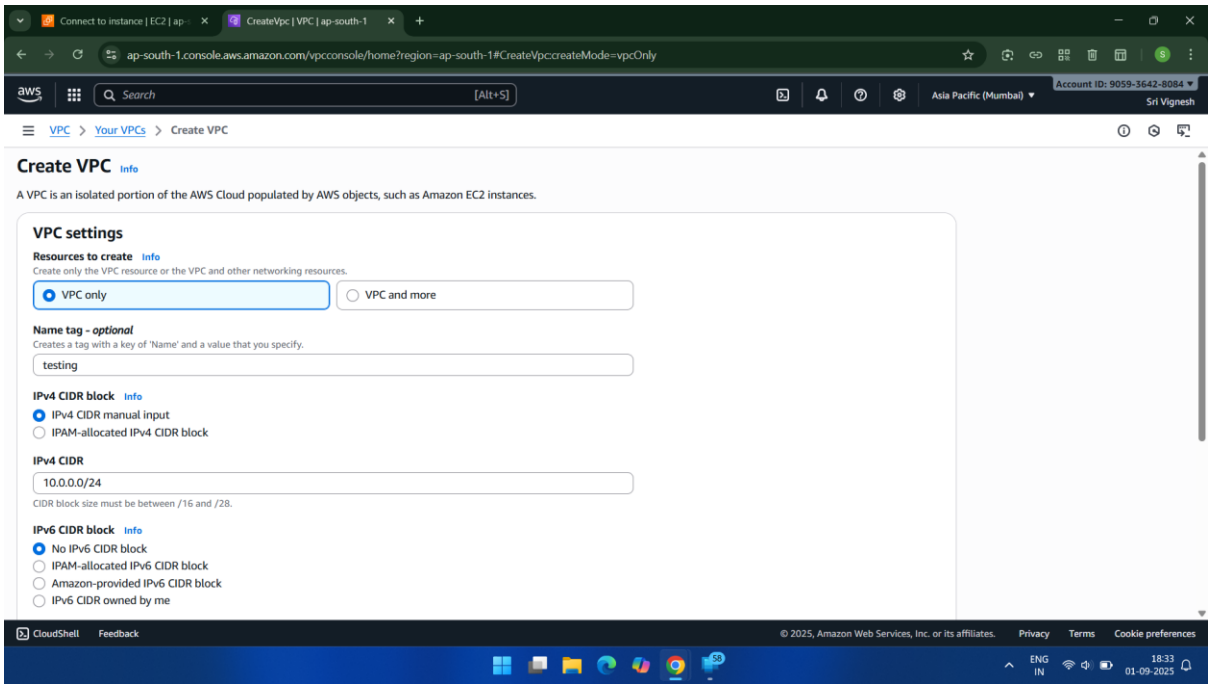
AWS recommends you choose your VPC CIDR block from one of these ranges to keep your VPC private.

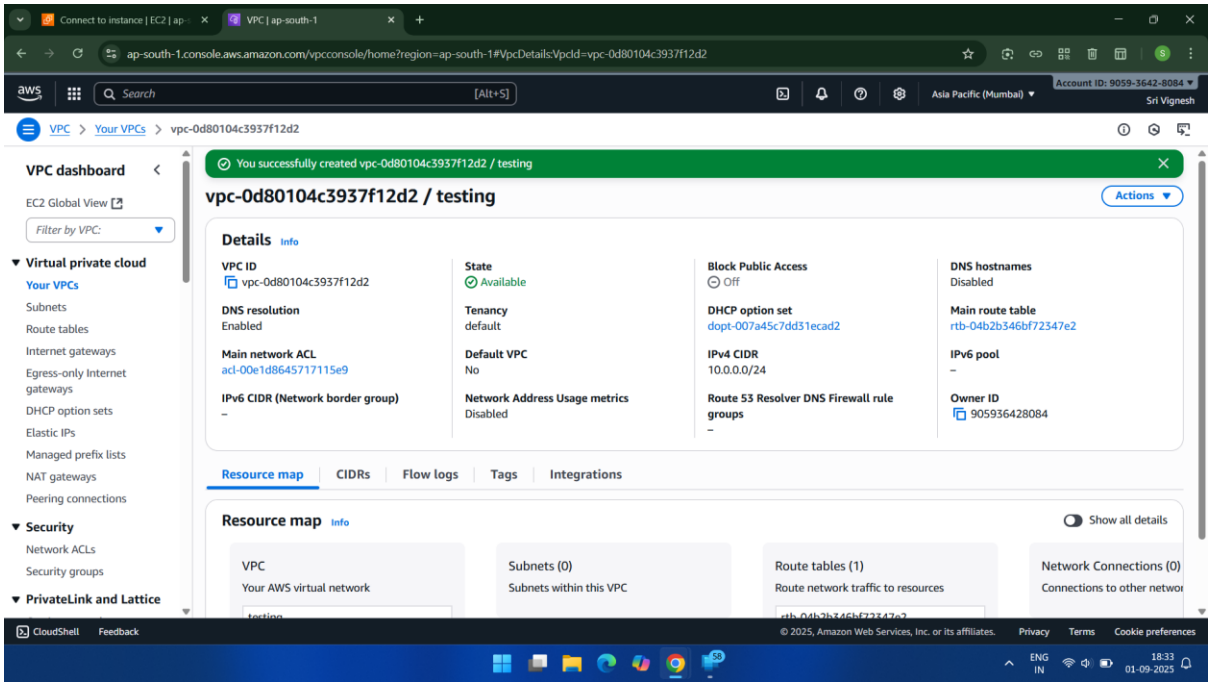
In AWS, every subnet loses 5 IPs that you can't assign to resources:

1. First IP (network address) – Identifies the subnet itself, not a device.
2. Second IP (VPC router) – Used by AWS as the default gateway for routing traffic.
3. Third IP (DNS server) – Reserved for the Amazon-provided DNS in that VPC.
4. Fourth IP (future use) – Held for AWS internal purposes.
5. Last IP (broadcast address) – Used for broadcasting to all devices in that subnet (even though AWS doesn't actually support broadcast traffic).

- **Ipv6 CIDR block**
  - *No IPv6 CIDR block:* No IPv6 address range is associated with the VPC. ✓
  - *IPAM-allocated IPv6 CIDR block:* An IPv6 CIDR block assigned to the VPC by AWS IP Address Manager (IPAM).
  - *Amazon-provided IPv6 CIDR block*: A globally unique IPv6 CIDR block provided by AWS that you can associate with your VPC.
  - *IPv6 CIDR owned by me:* A publicly routable IPv6 CIDR block that you own and have brought into AWS to use within your VPC.

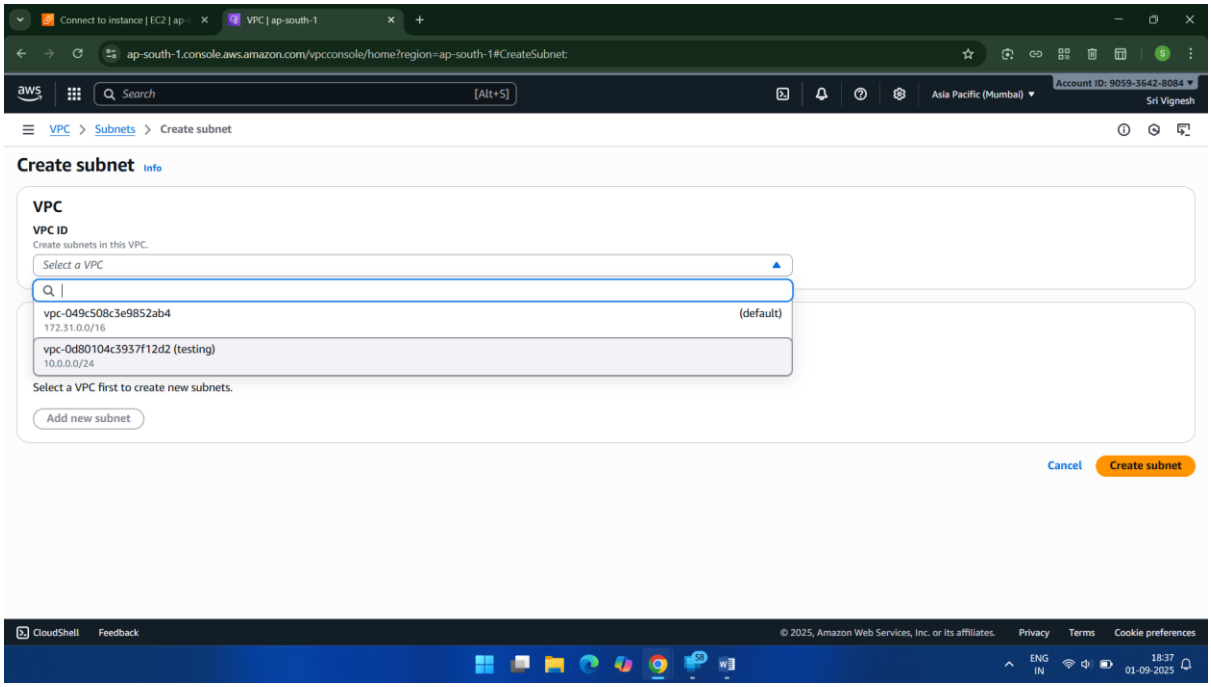
- Create a VPC.



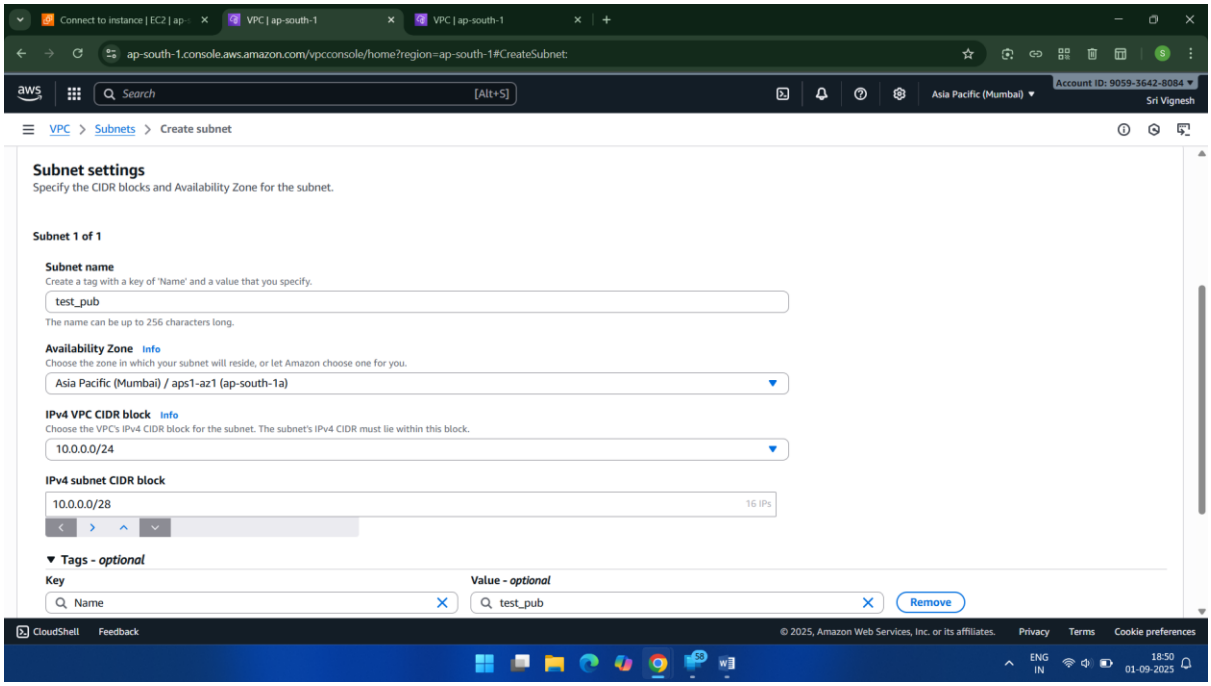

Connect to instance | EC2 | ap-
VPC | ap-south-1
ap-south-1.console.aws.amazon.com/vpcconsole/home?region=ap-south-1#VpcDetails:VpcId=vpc-0d80104c3937f12d2
Search
[Alt+S]
Asia Pacific (Mumbai)
Account ID: 9059-3642-8084
Sri Vignesh
VPC > Your VPCs > vpc-0d80104c3937f12d2
VPC dashboard
EC2 Global View
Filter by VPC:
Virtual private cloud
Your VPCs
Subnets
Route tables
Internet gateways
Egress-only Internet gateways
DHCP option sets
Elastic IPs
Managed prefix lists
NAT gateways
Peering connections
Security
Network ACLs
Security groups
PrivateLink and Lattice
You successfully created vpc-0d80104c3937f12d2 / testing
vpc-0d80104c3937f12d2 / testing
Actions
Details Info
VPC ID
vpc-0d80104c3937f12d2
State
Available
Block Public Access
Off
DNS hostnames
Disabled
DNS resolution
Enabled
Tenancy
default
DHCP option set
dopt-007a45c7dd31ecad2
Main route table
rtb-04b2b346bf72347e2
Main network ACL
acl-00e1d8645717115e9
Default VPC
No
IPv4 CIDR
10.0.0.0/24
IPv6 pool
–
IPv6 CIDR (Network border group)
–
Network Address Usage metrics
Disabled
Route 53 Resolver DNS Firewall rule groups
–
Owner ID
905936428084
Resource map
CIDRs
Flow logs
Tags
Integrations
Resource map Info
Show all details
VPC
Your AWS virtual network
Subnets (0)
Subnets within this VPC
Route tables (1)
Route network traffic to resources
Network Connections (0)
Connections to other networ
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences
ENG
IN
18:33
01-09-2025

Step 2: Create a Subnet

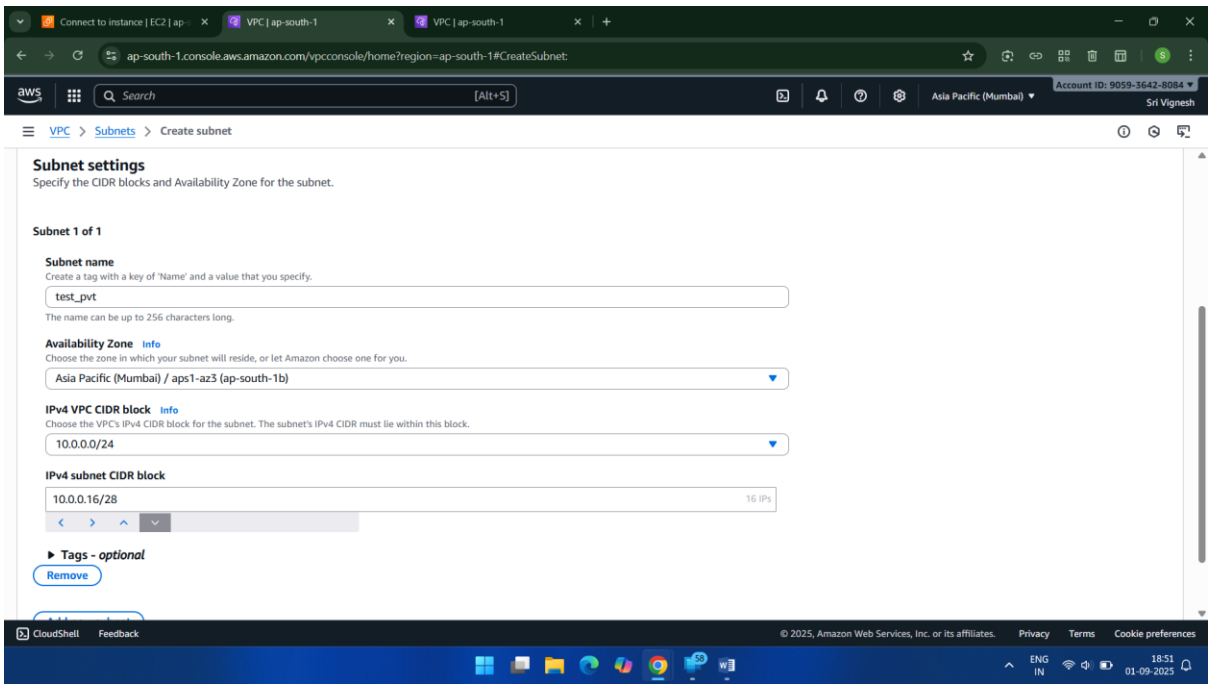
- Subnet -> Create Subnet.
- Select the VPC need to associated with this new subnet.
- Subnet name.
- *Select the avaliability zone* – 1 subnet per avaliability zone.
    - For **public** subnet select – **ap-south-1a**
    - For **private** subnet select – **ap-south-1b**
- *IPv4 subnet CIDR block* – Manually enter the ip range
    - Here I select the range between `10.0.0.0/28` for public ranges 16 IP's.
    - For private range between `10.0.0.16/28` range gives 16 IP's.
- Click -> Create Subnet.



Creting Public Subnet

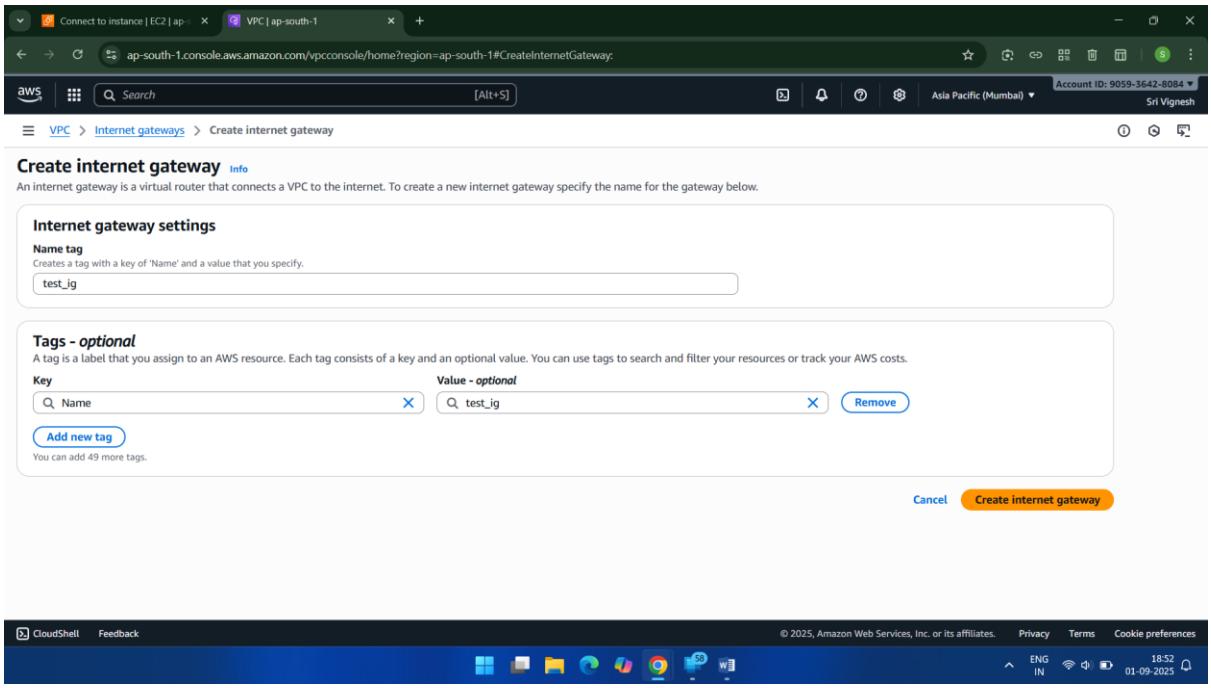


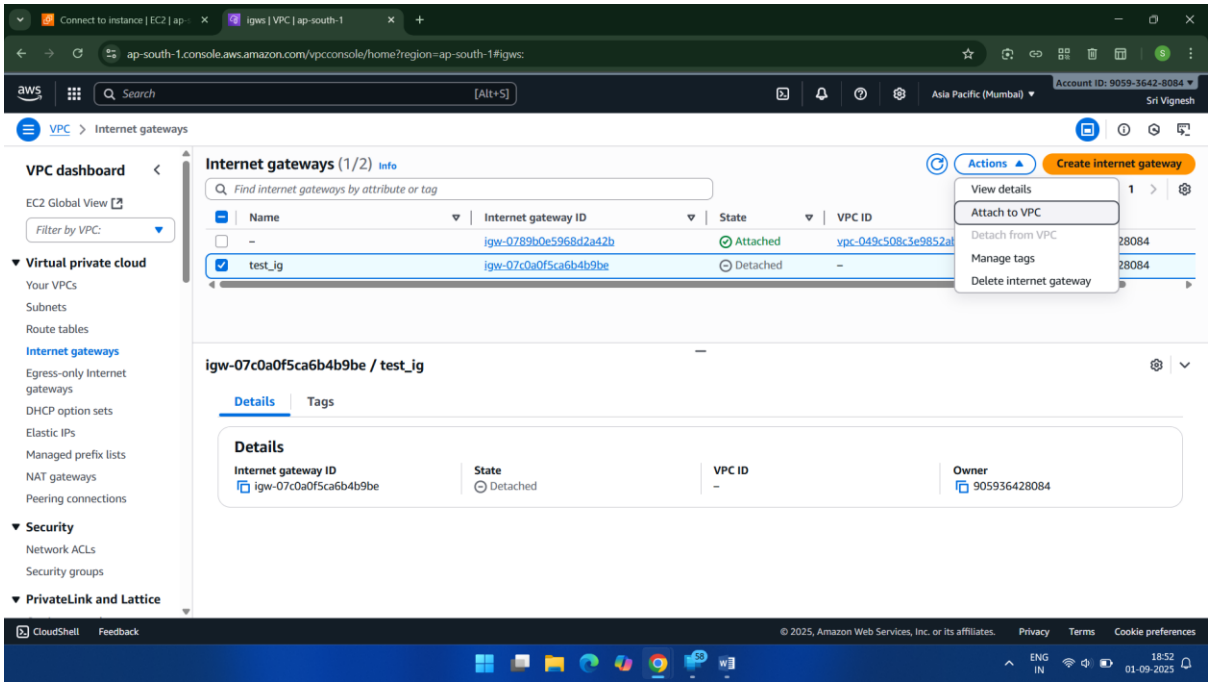
Creting Private Subnet

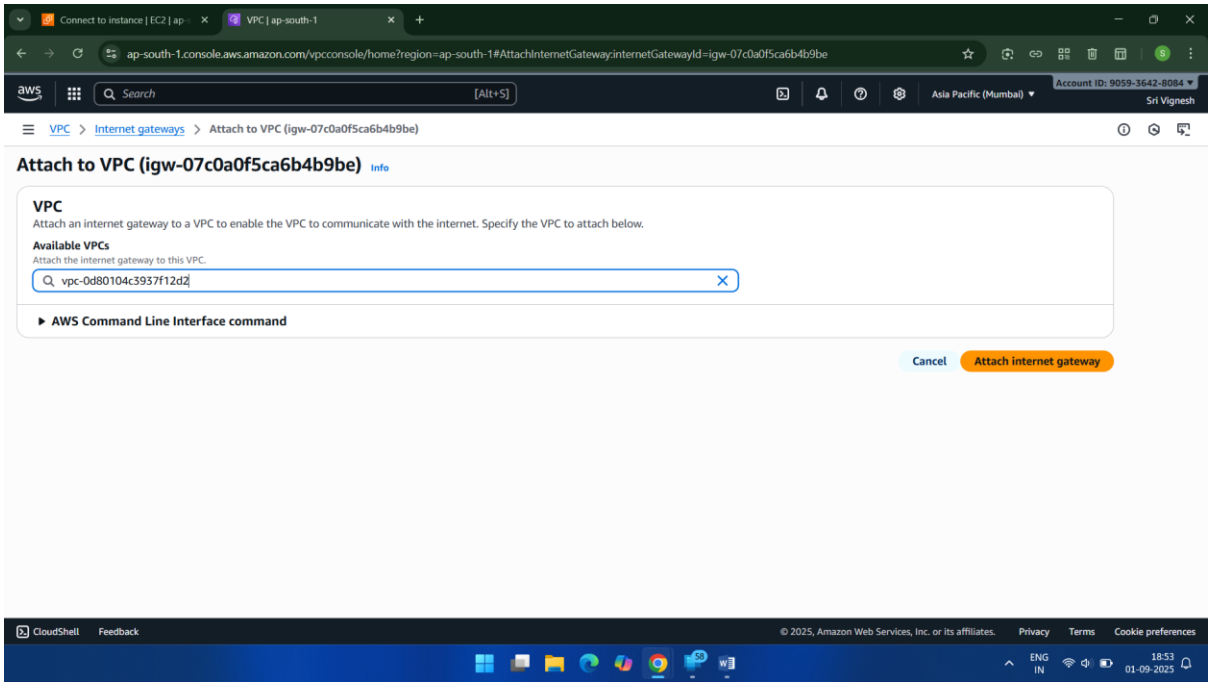


Step 3: Creating Internet gateway

- AWS –> Create an Internet Gateway
- Name the Internet gaetway
- Click -> Create Internet gateway
- *Attach the Internet Gateway to the VPC*
    - Select the Ig -> Action -> Attach to VPC.
    - Select the VPC
    - Click -> Attach to Intenet Gateway.

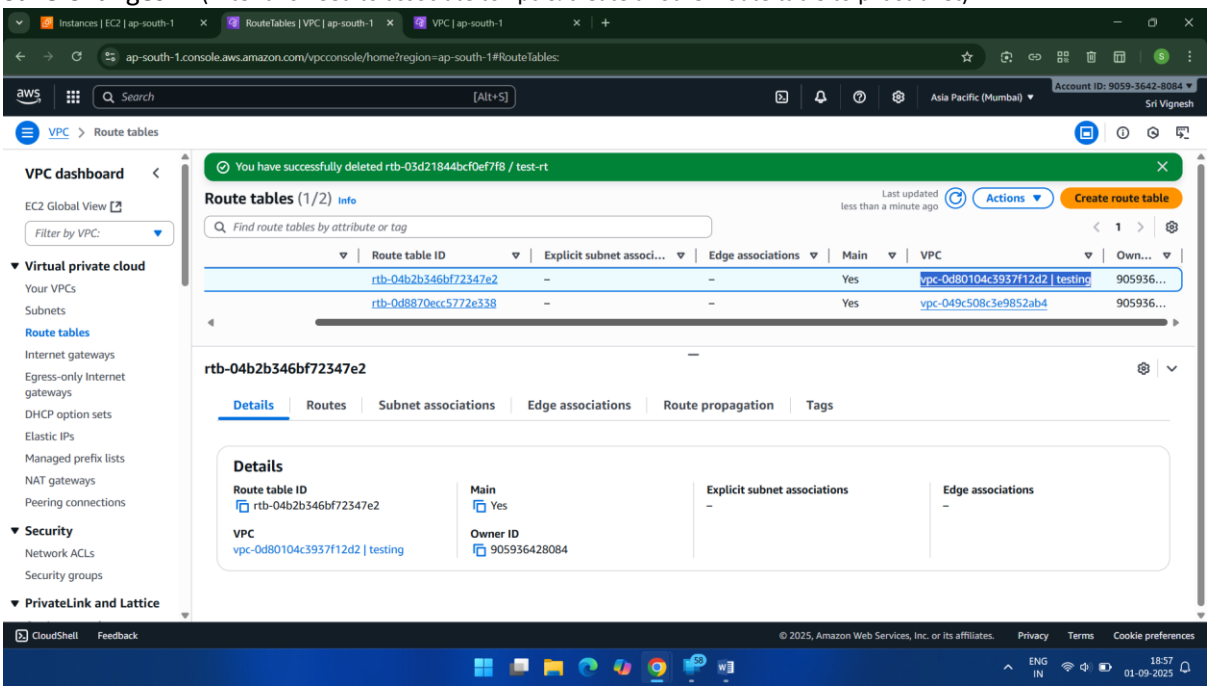


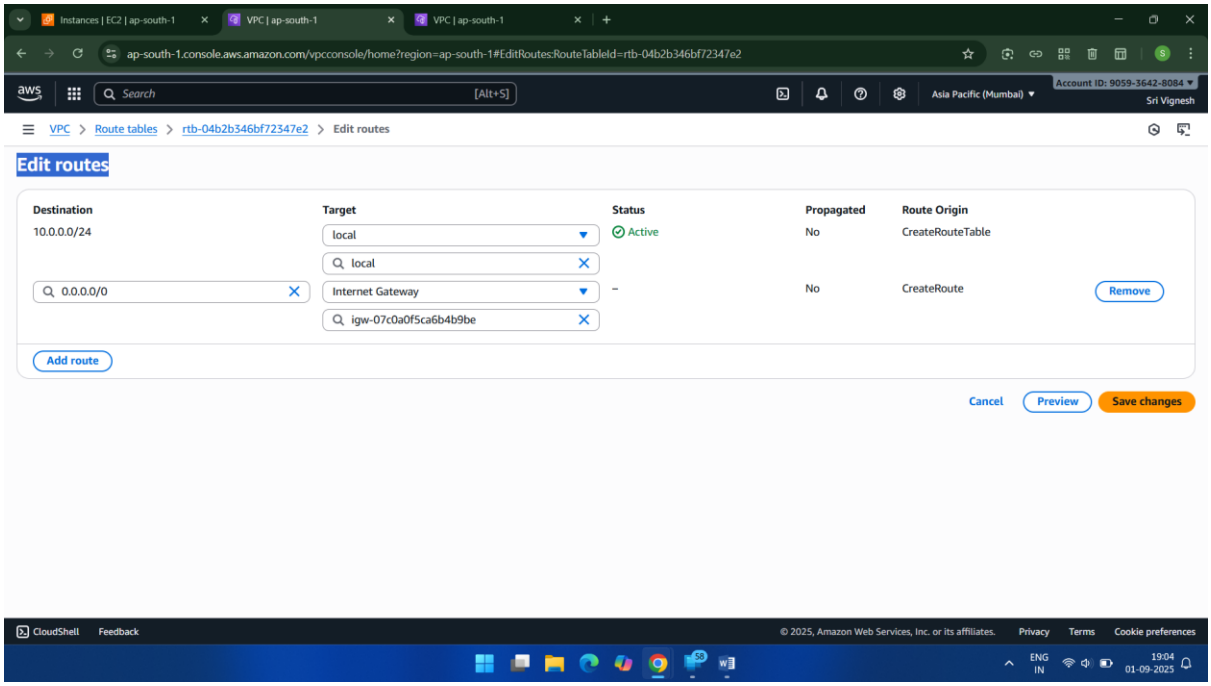




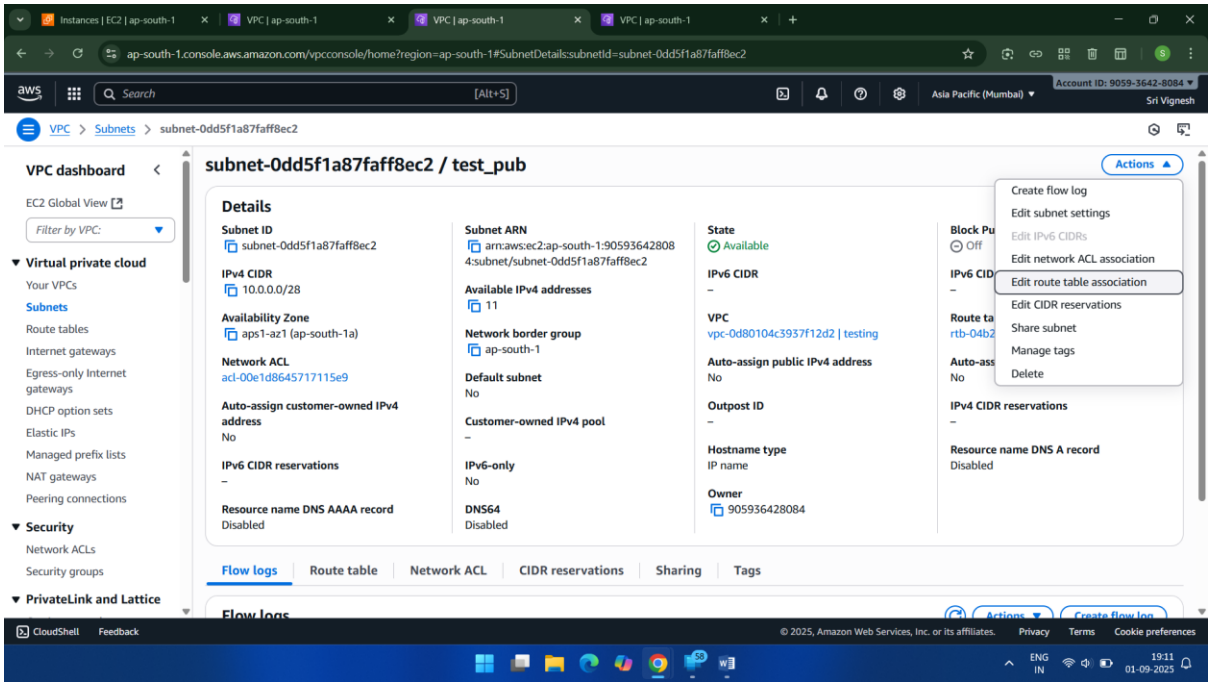
Step 4: Creating Route table

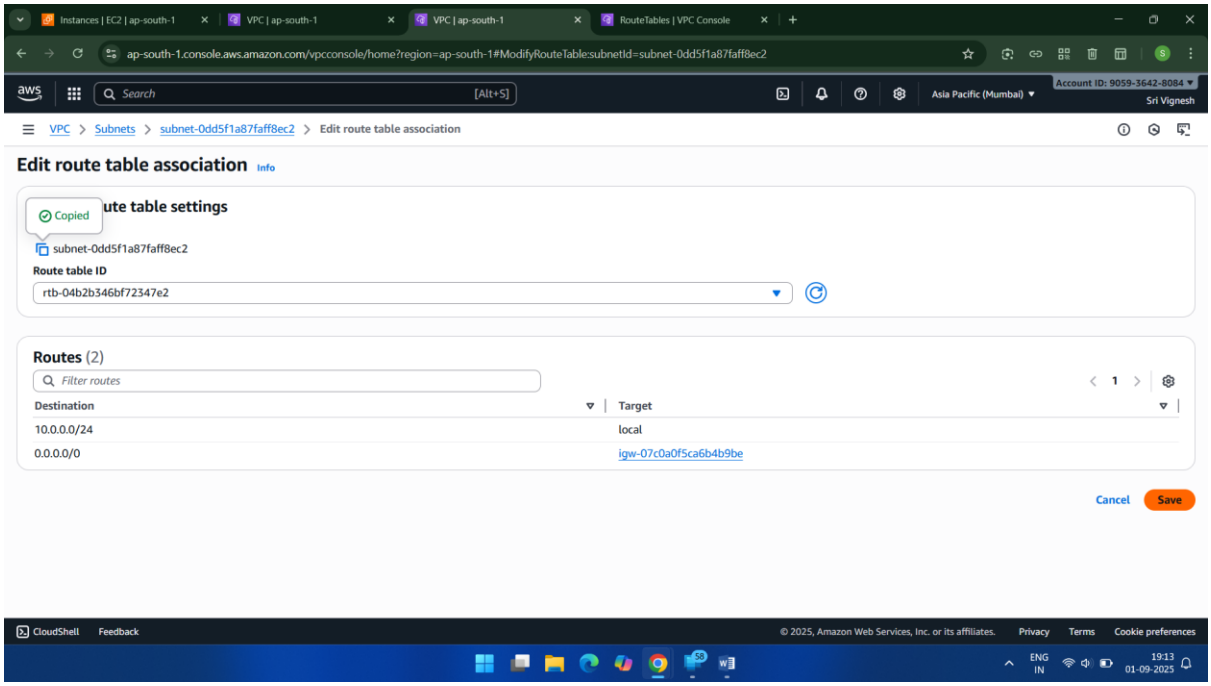
- By default there is a Route table and Need to create two route tables each for Public and Private subnet.
- This says **Main** that can be used for **Public subnet.**
- Select the RouteTable states as Main -> Actions -> Edit route.
- *Edit Route*
    - By default there is a Destination- this will route the traffic within the VPC.
    - Add another route
        - **Destination:** Enter `0.0.0.0/0`
        - **Target:** Select Internet Gateway
- Save Changes -> (After this need to associate to vpc & create another route table to pvt subnet)



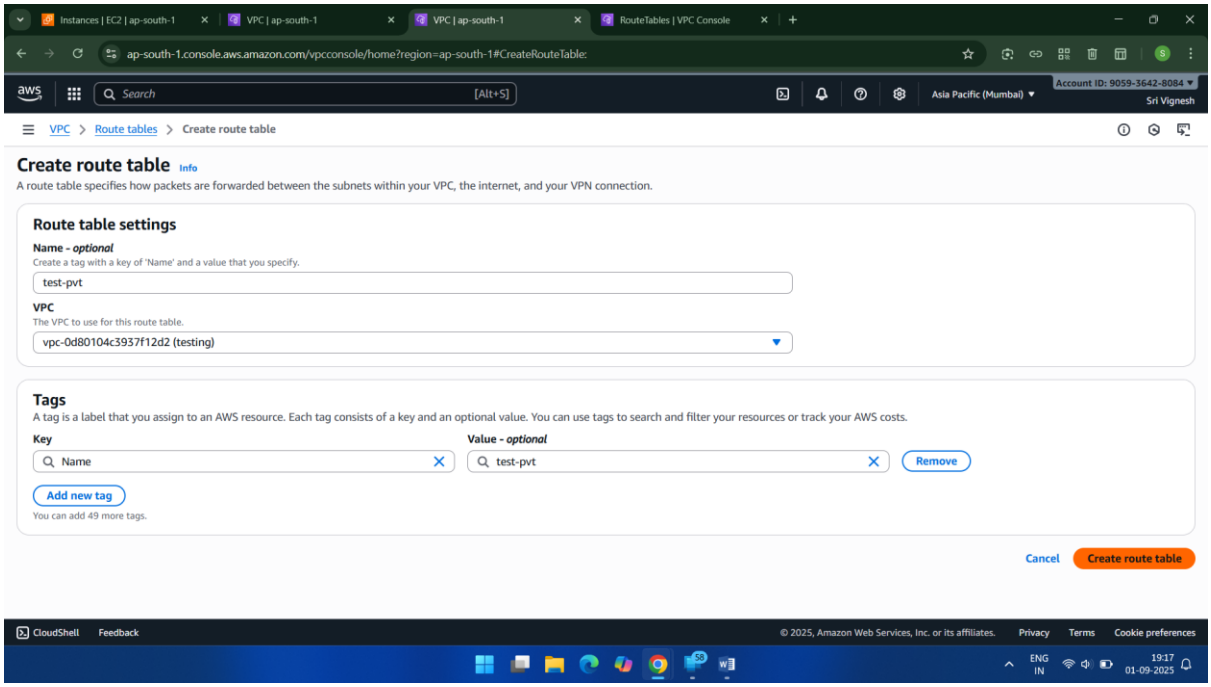


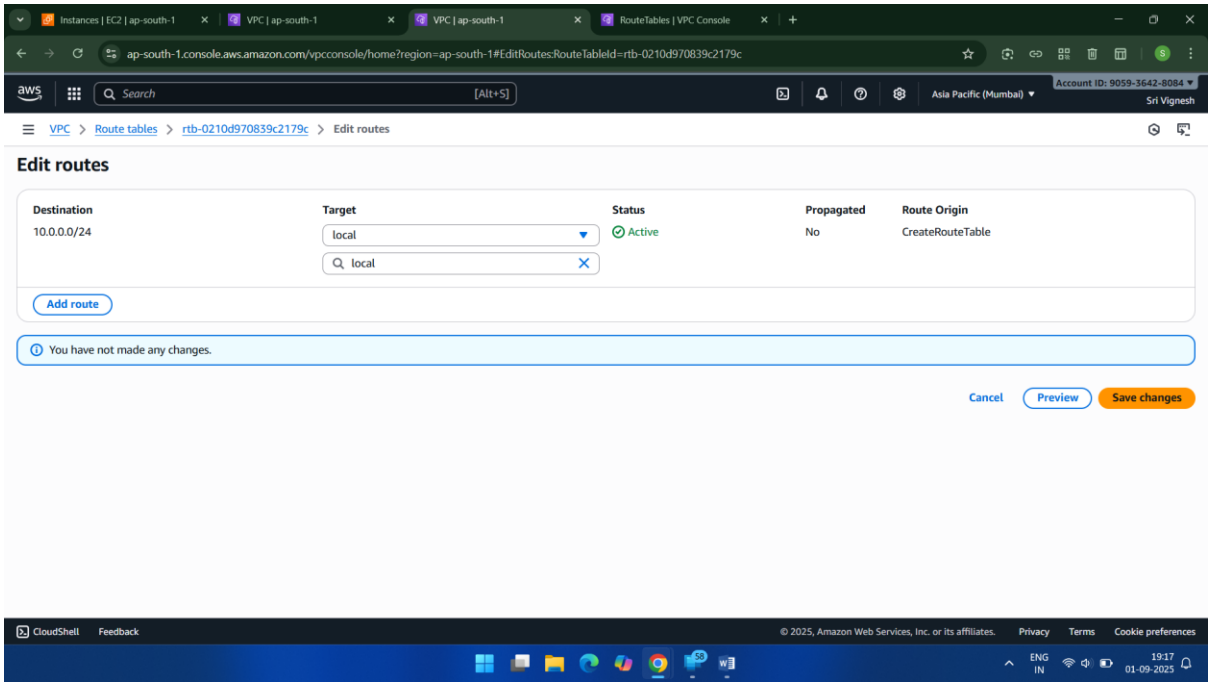
- Select the **Public** subnet.
- Action-> Edit route table association
- Select the correct route table id
- Click Save

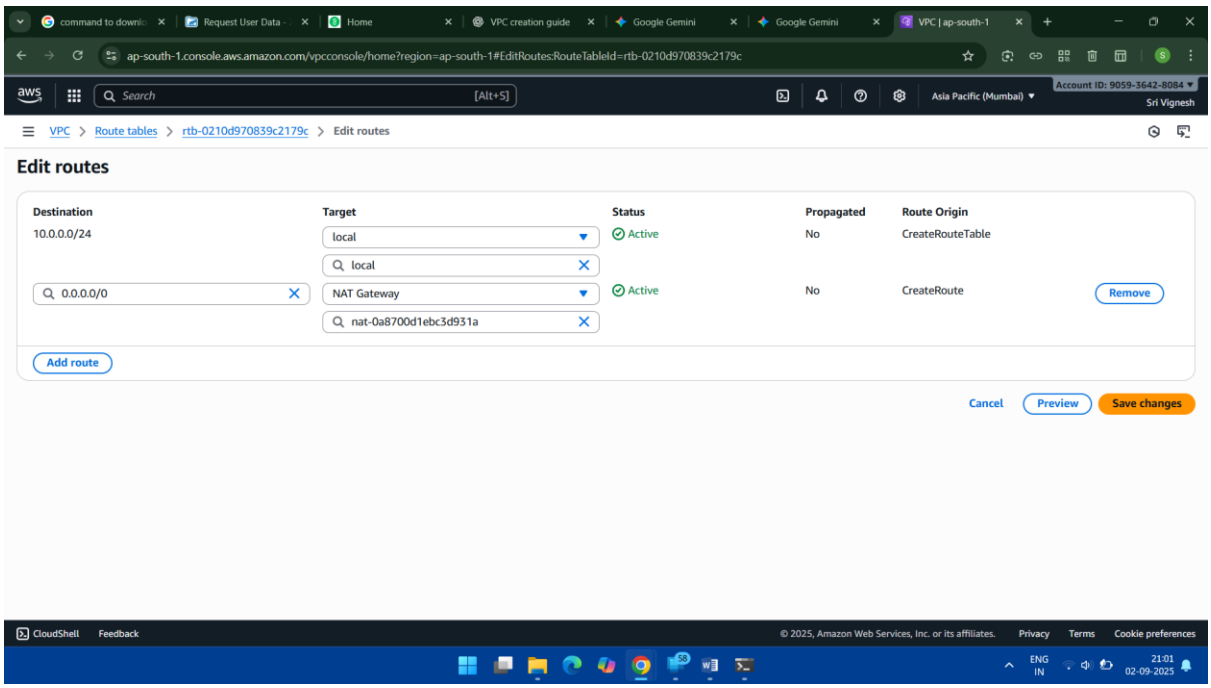




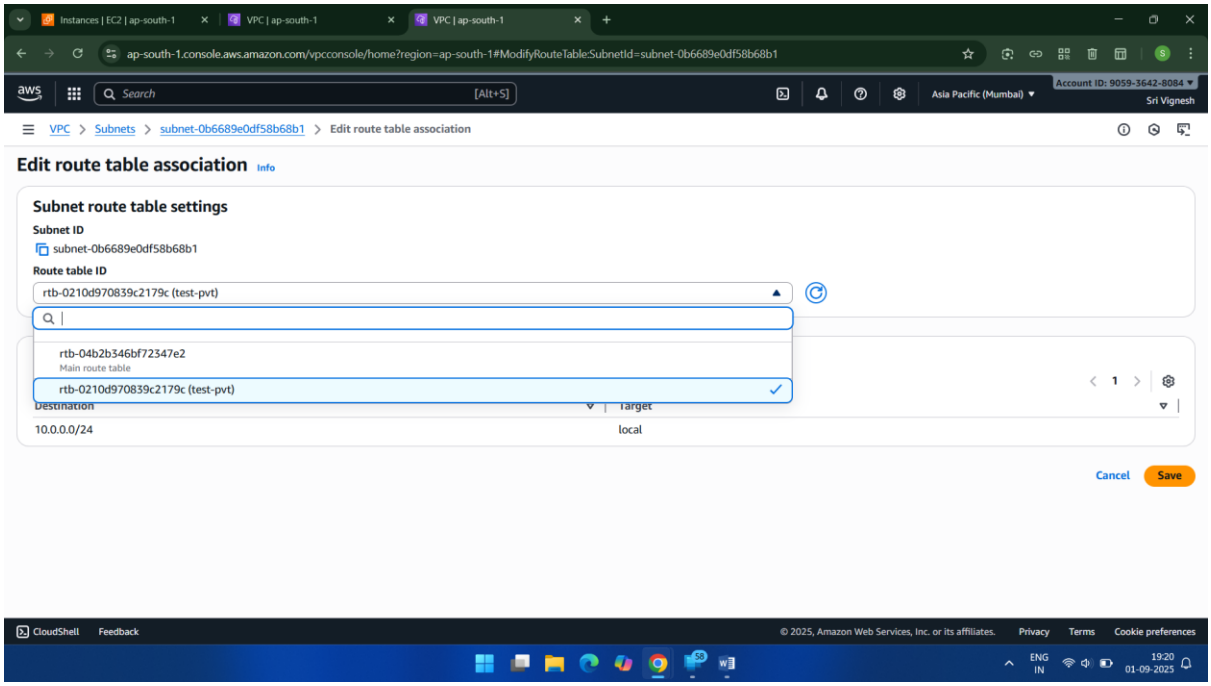
- Create Another Route table For **Private Subnet**.
- Name and Select the **VPC** .
- Click -> Create route table.
- Action -> *Edit route*
    - **Destination:** Enter `0.0.0.0/0`
    - **Target:** Select **Nat Gateway ✓** - Create NAT gateway
- Click -> Save changes.




Instances | EC2 | ap-south-1
VPC | ap-south-1
VPC | ap-south-1
RouteTables | VPC Console
ap-south-1.console.aws.amazon.com/vpcconsole/home?region=ap-south-1#EditRoutes:RouteTableId=rtb-0210d970839c2179c
Search
[Alt+S]
Asia Pacific (Mumbai)
Account ID: 9059-3642-8084
Sri Vignesh
VPC > Route tables > rtb-0210d970839c2179c > Edit routes
Edit routes
Destination
Target
Status
Propagated
Route Origin
10.0.0.0/24
local
Active
No
CreateRouteTable
local
Add route
You have not made any changes.
Cancel
Preview
Save changes
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences
ENG
IN
19:17
01-09-2025


command to downlo
Request User Data
Home
VPC creation guide
Google Gemini
Google Gemini
VPC | ap-south-1
ap-south-1.console.aws.amazon.com/vpcconsole/home?region=ap-south-1#EditRoutes:RouteTableId=rtb-0210d970839c2179c
Search
[Alt+S]
Asia Pacific (Mumbai)
Account ID: 9059-3642-8084
Sri Vignesh
VPC > Route tables > rtb-0210d970839c2179c > Edit routes
Edit routes
Destination
Target
Status
Propagated
Route Origin
10.0.0.0/24
local
Active
No
CreateRouteTable
local
0.0.0.0/0
NAT Gateway
Active
No
CreateRoute
Remove
nat-0a8700d1ebc3d931a
Add route
Cancel
Preview
Save changes
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences
ENG
IN
21:01
02-09-2025

- Select the **Private** subnet.
- Action-> Edit route table association
- Select the correct route table id
- Click Save.



Step 5: Creating NAT(Network Address Translation) gateway

- NAT gateway -> Name.
- Select the Private Subnet.
- *Connectivity type*
    - Public NAT
    - Private NAT

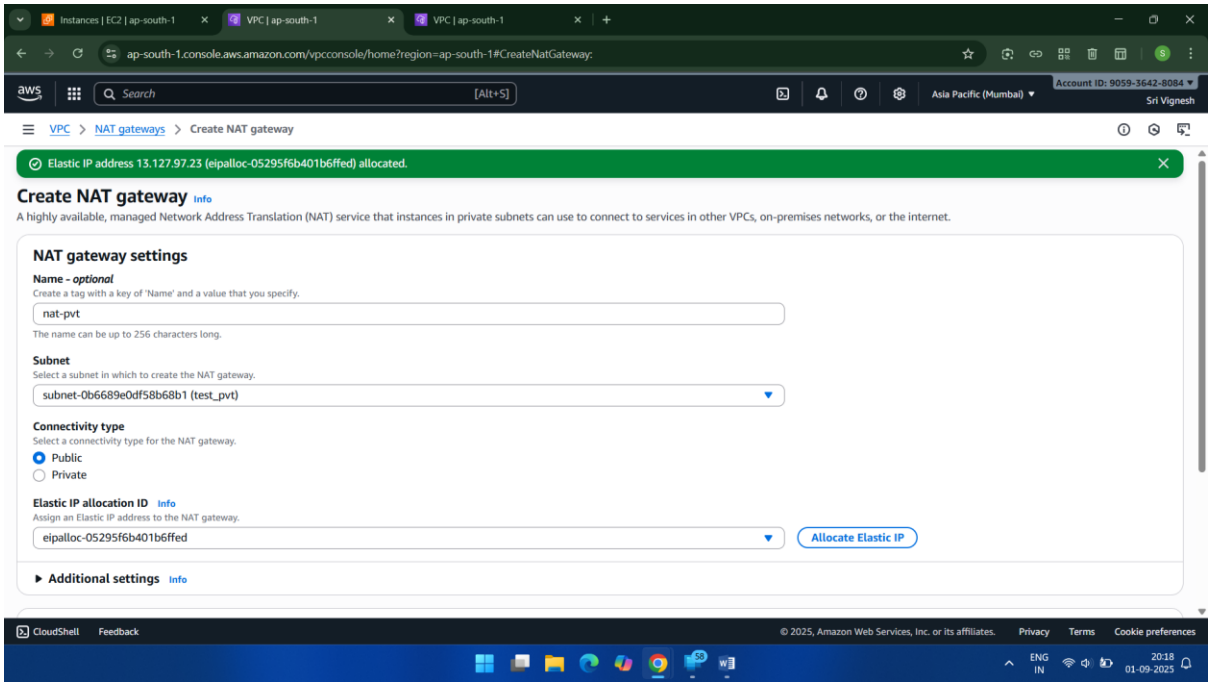
Public NAT Gateway

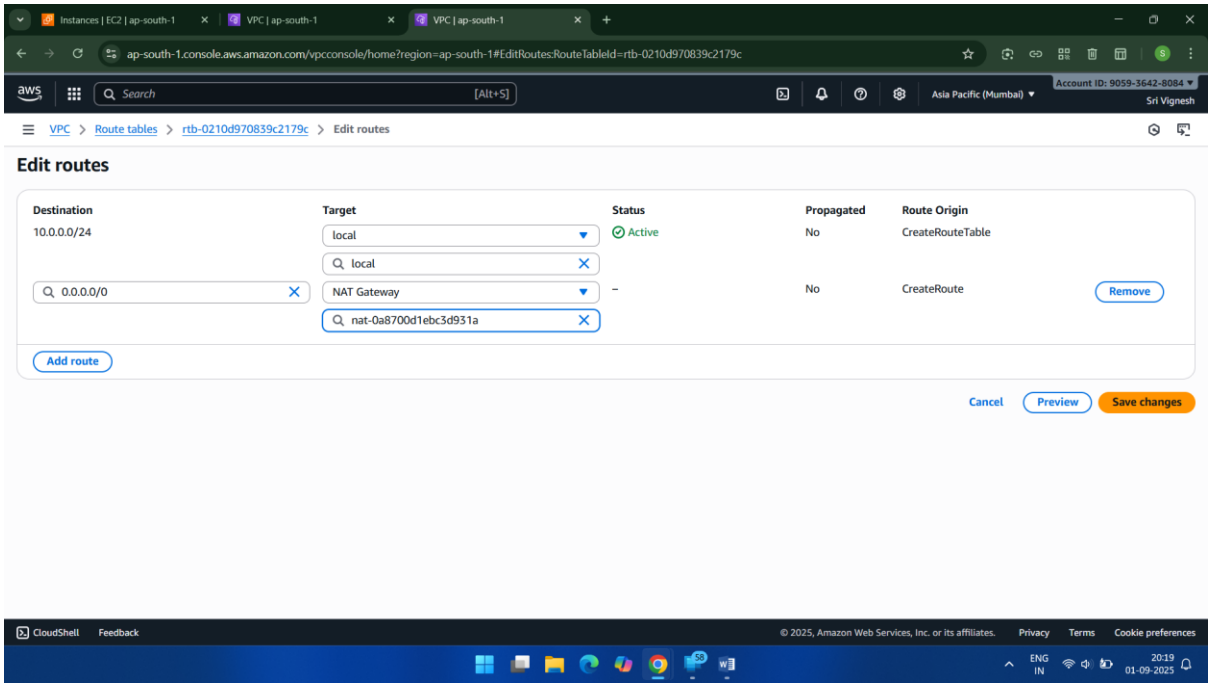
A **public** NAT gateway is used for instances in a private subnet that need to send outbound traffic to the internet while preventing any inbound connections from the internet. It is created in a public subnet and requires an Elastic IP address. When an instance in a private subnet sends a request, the public NAT gateway translates the instance's private IP address to its own Elastic IP address before sending the traffic to the internet gateway.

Private NAT Gateway

A **private** NAT gateway is used to enable communication between instances in a private subnet and other VPCs or your on-premises network. It is created in a private subnet and does **not** require an Elastic IP address. This type of NAT gateway can be used to route traffic through a transit gateway or a virtual private gateway, but traffic routed to an internet gateway will be dropped.

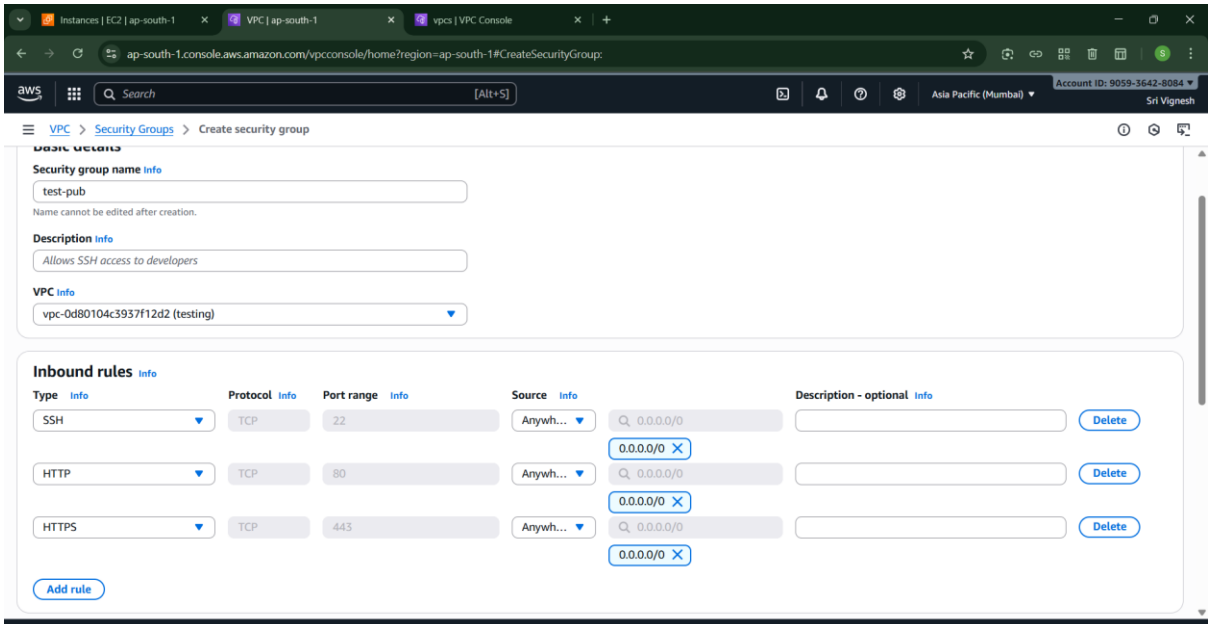
- Elastic IP allocation ID
    - Select -> Allocate Elastic IP.
    - This will give you an elastic IP.
- Save and **add** this in that **Private Route table** Destinations.

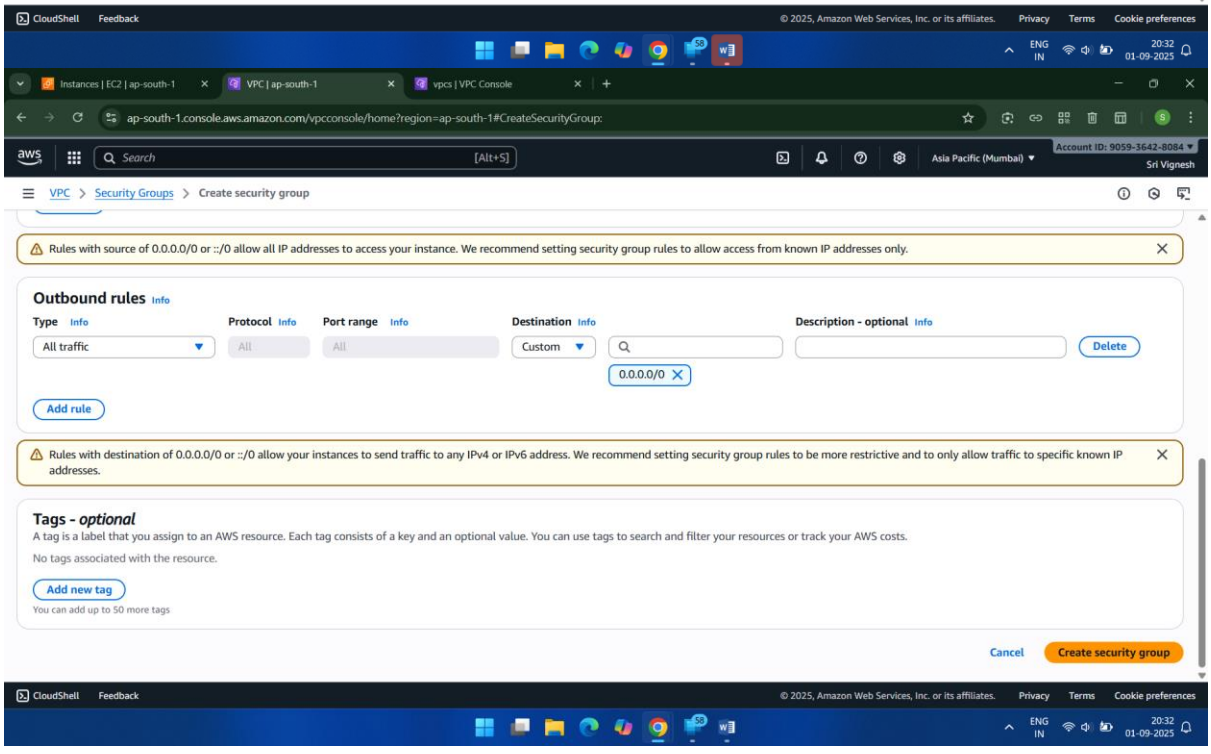




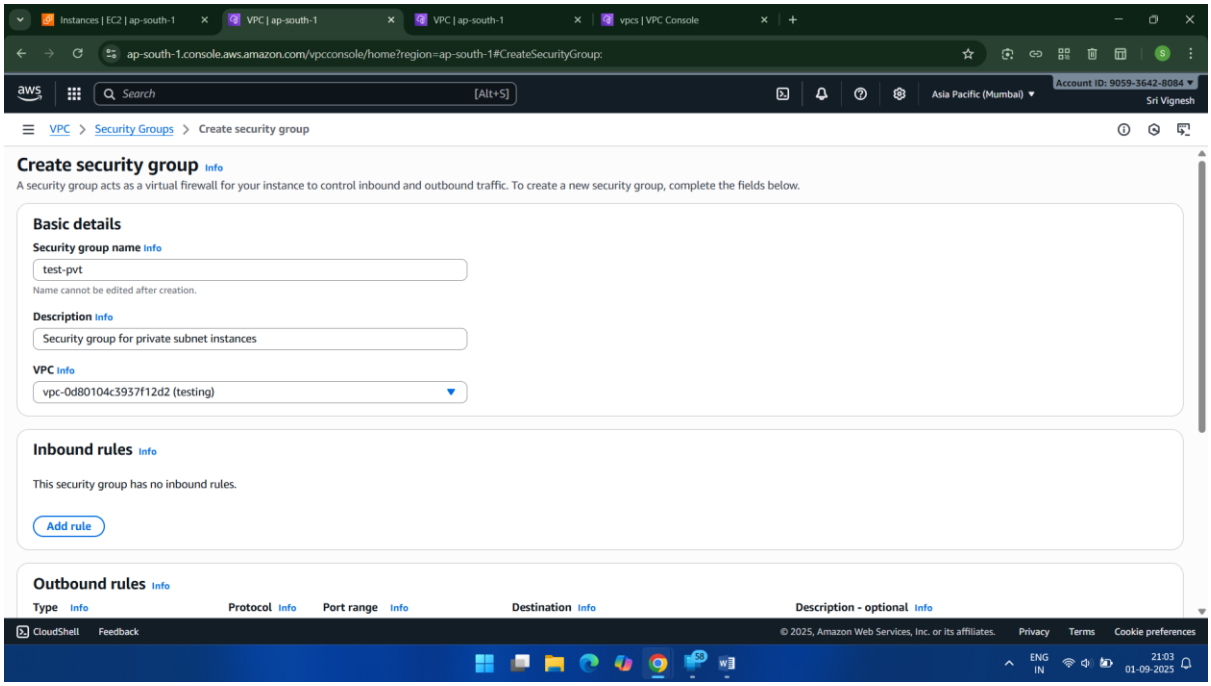
Step 6: Creating Security Group

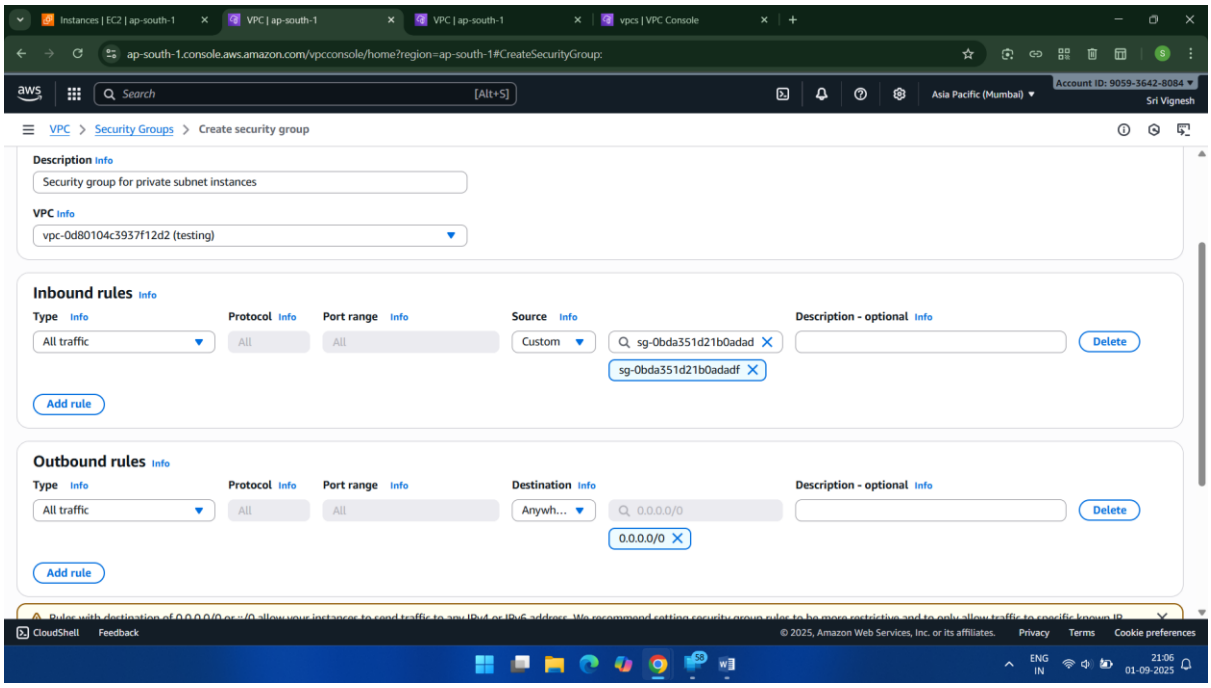
- Create Security group for **Public Subnet**.
- Name -> add Discription
- Select the VPC
- *Inbound rules*
    - Add type: SSH -> source: Anywhere `0.0.0.0/0`
    - Add type: HTTP -> source: Anywhere `0.0.0.0/0`
    - Add type: HTTPS -> source: Anywhere `0.0.0.0/0`
- *Outbound rules*
    - Add type: All Traffic-> source: Anywhere `0.0.0.0/0`
- Click -> Create security group.





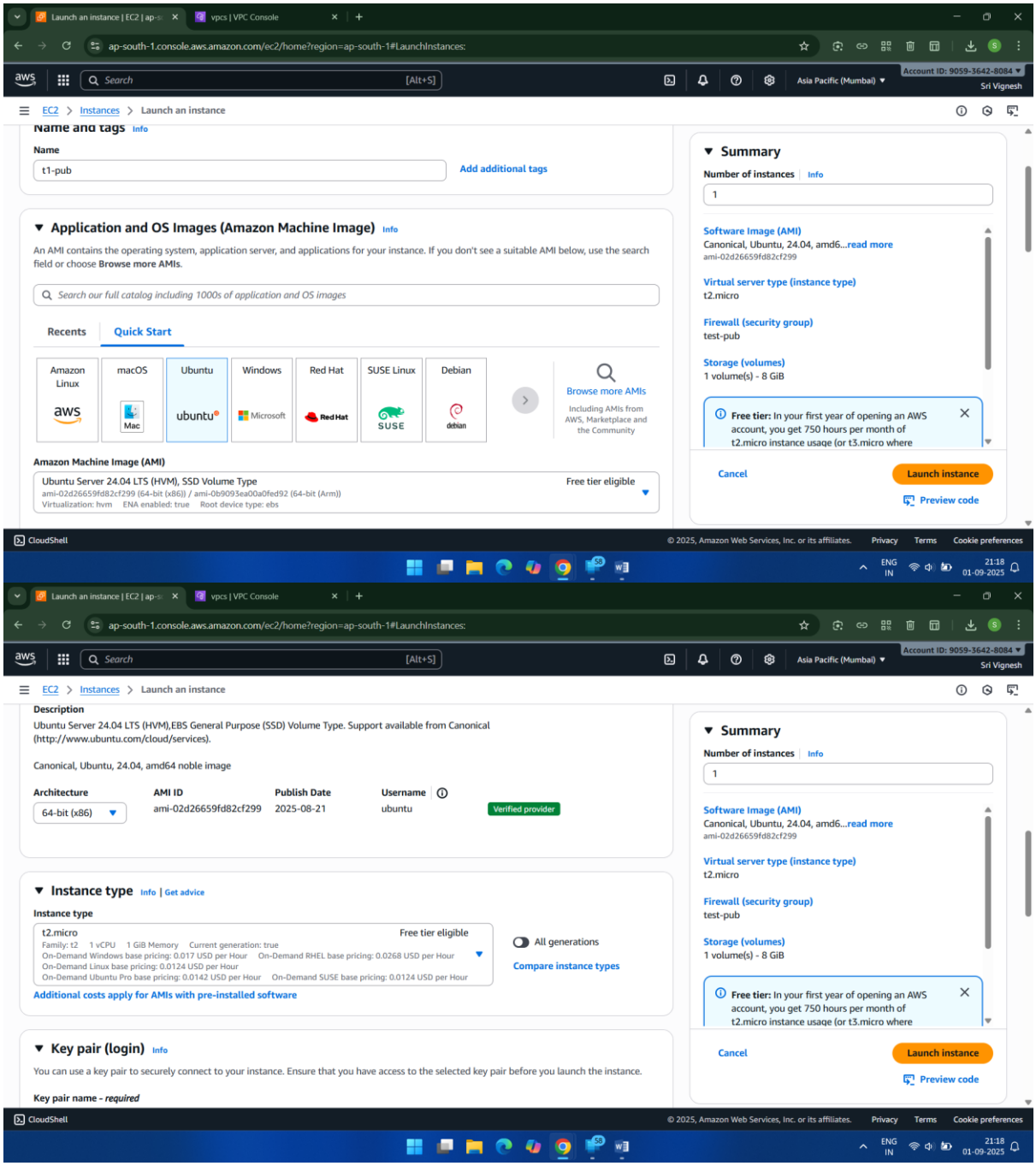
- Create Security group for **Private Subnet**.
- Name -> add Discription
- Select the VPC
- *Inbound rules*
    - Add type: All Traffic -> source: Custom -> **Public Security group**.
        - This will only allow traffic from Public subnet.
- *Outbound rules*
    - Add type: All Traffic-> source: Anywhere `0.0.0.0/0`
- Click -> Create security group.

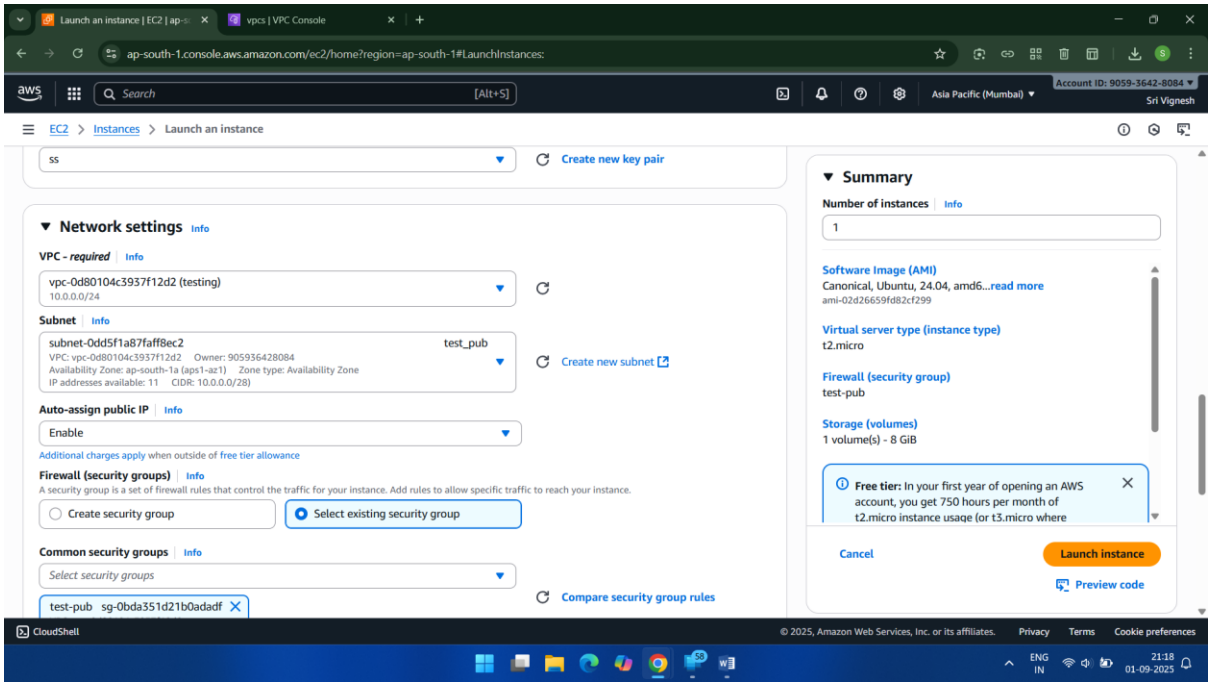


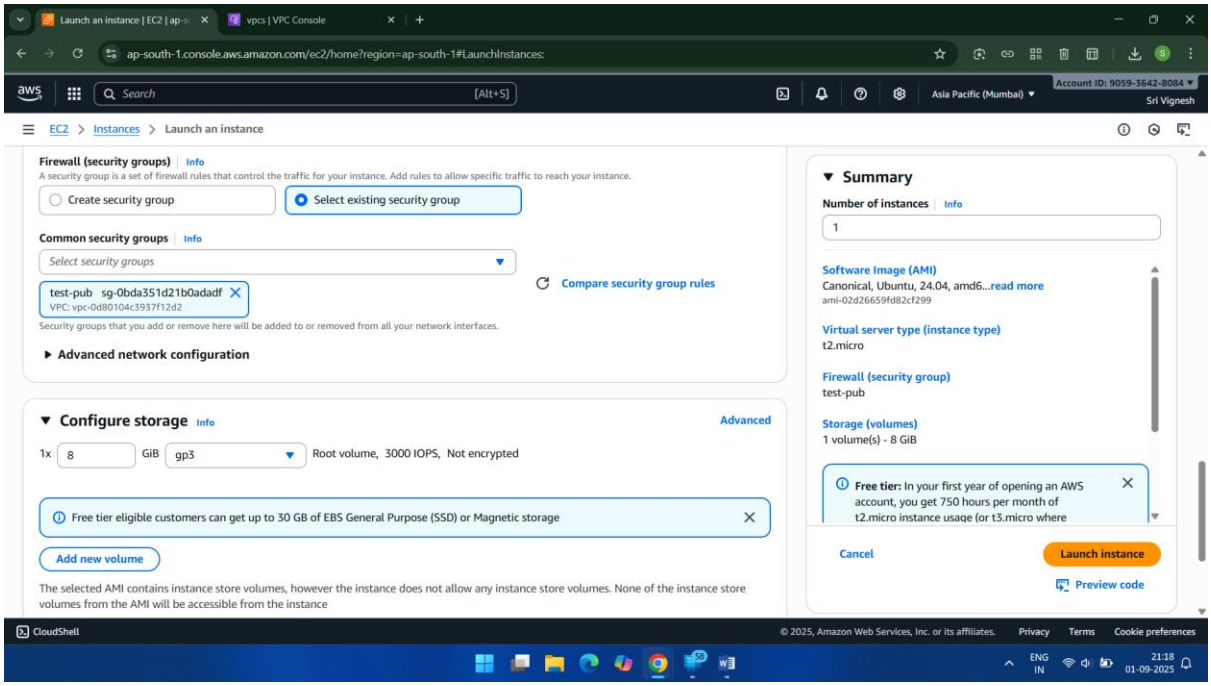


Step 7: Creating Security Group

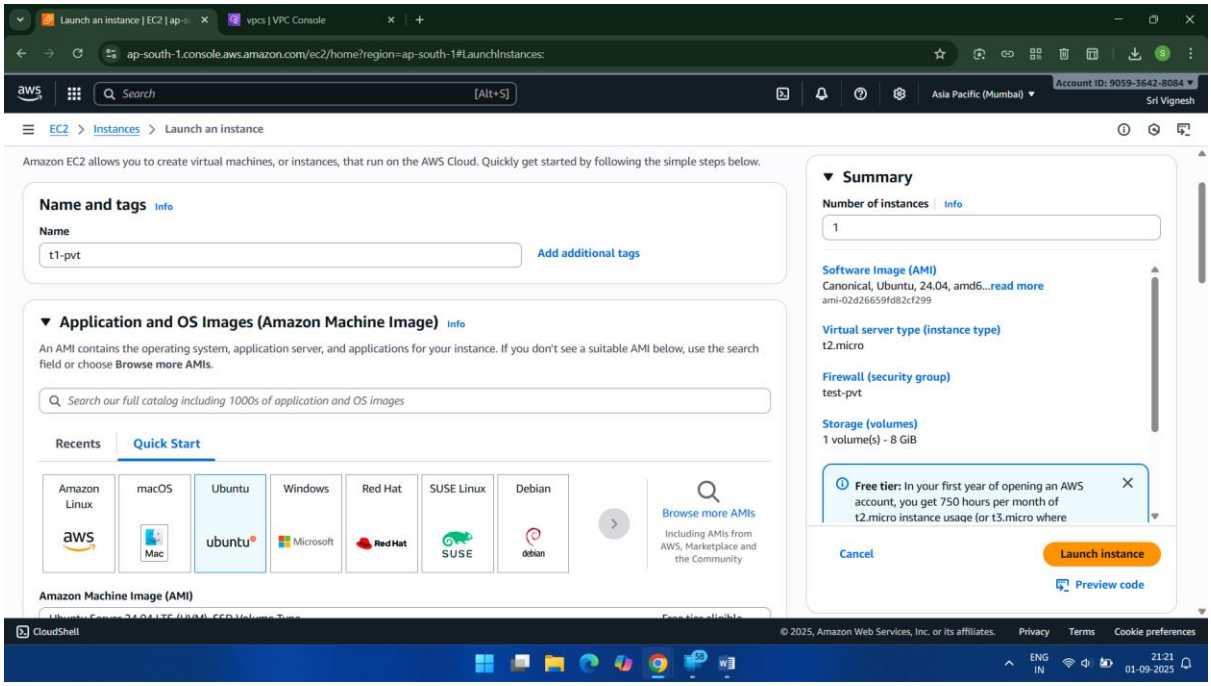
- Create 2 Instances with same OS and Key pair. One for **Public and Private subnet.**
- ***Network Settings***
    - Click Edit
    - Select the VPC
    - Select the **Public subnet** 1a.
    - Auto assign IP: **Enable** (For public subnet)
    - Firewall: Select existing Security Group.
    - Select -> Public Security group.
    - Click -> Launch Instance.

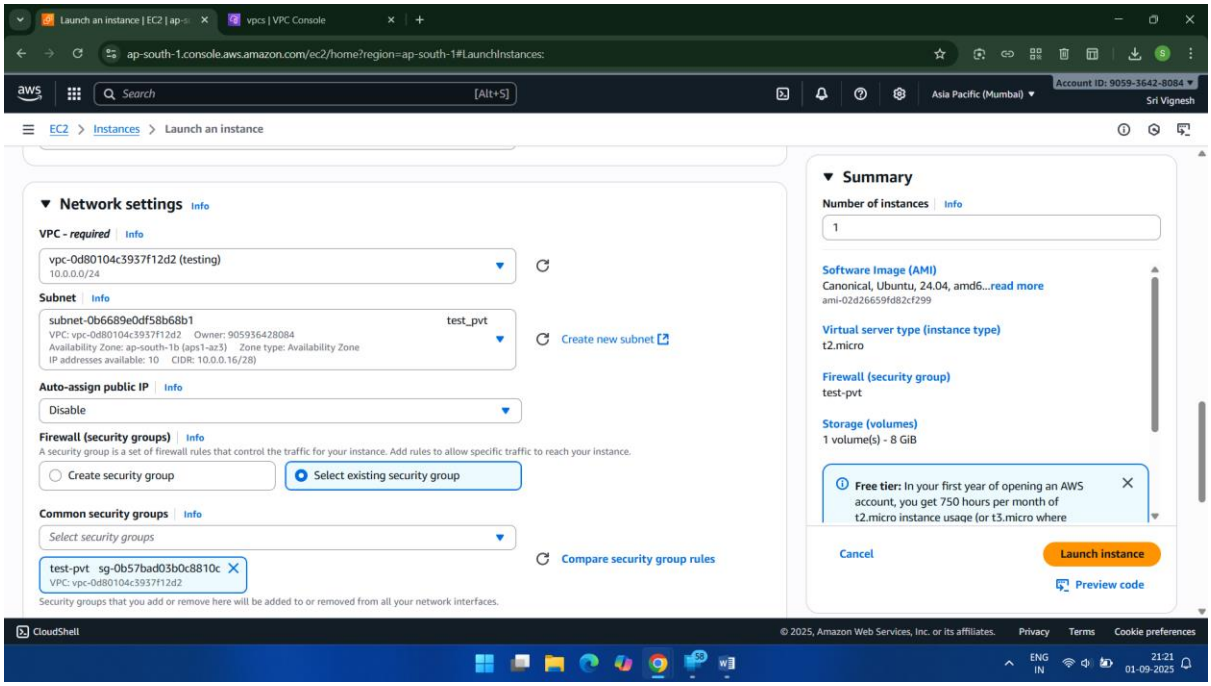






- Creation of instance for **Private subnet.**
- ***Network Settings***
  - Click Edit
  - Select the VPC
  - Select the **Private subnet** 1b.
  - Auto assign IP: **Disable** (For Private subnet)
  - Firewall: Select existing Security Group.
  - Select -> **Private Security group**.
  - Click -> Launch Instance.


Launch an instance | EC2 | ap-s
vpcs | VPC Console
ap-south-1.console.aws.amazon.com/ec2/home?region=ap-south-1#LaunchInstances:
Search
[Alt+S]
Asia Pacific (Mumbai)
Account ID: 9059-3642-8084
Sri Vignesh
EC2 > Instances > Launch an instance
Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following the simple steps below.
Name and tags Info
Name
t1-pvt
Add additional tags
Application and OS Images (Amazon Machine Image) Info
An AMI contains the operating system, application server, and applications for your instance. If you don't see a suitable AMI below, use the search field or choose Browse more AMIs.
Search our full catalog including 1000s of application and OS images
Recents
Quick Start
Amazon Linux
macOS
Ubuntu
Windows
Red Hat
SUSE Linux
Debian
Browse more AMIs
Including AMIs from AWS, Marketplace and the Community
Amazon Machine Image (AMI)
Summary
Number of instances Info
1
Software Image (AMI)
Canonical, Ubuntu, 24.04, amd6...read more
ami-02d26659fd82cf299
Virtual server type (instance type)
t2.micro
Firewall (security group)
test-pvt
Storage (volumes)
1 volume(s) - 8 GiB
Free tier: In your first year of opening an AWS account, you get 750 hours per month of t2.micro instance usage (or t3.micro where
Cancel
Launch instance
Preview code
CloudShell
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences
ENG
IN
21:21
01-09-2025


Launch an instance | EC2 | ap-s
vpcs | VPC Console
ap-south-1.console.aws.amazon.com/ec2/home?region=ap-south-1#LaunchInstances:
Search
[Alt+S]
Asia Pacific (Mumbai)
Account ID: 9059-3642-8084
Sri Vignesh
EC2 > Instances > Launch an instance
Network settings Info
VPC - required Info
vpc-0d80104c3937f12d2 (testing)
10.0.0.0/24
Subnet Info
subnet-0b6689e0df58b68b1
test_pvt
VPC: vpc-0d80104c3937f12d2 Owner: 905936428084
Availability Zone: ap-south-1b (aps1-az3) Zone type: Availability Zone
IP addresses available: 10 CIDR: 10.0.0.16/28)
Create new subnet
Auto-assign public IP Info
Disable
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
test-pvt sg-0b57bad03b0c8810c
VPC: vpc-0d80104c3937f12d2
Compare security group rules
Security groups that you add or remove here will be added to or removed from all your network interfaces.
Summary
Number of instances Info
1
Software Image (AMI)
Canonical, Ubuntu, 24.04, amd6...read more
ami-02d26659fd82cf299
Virtual server type (instance type)
t2.micro
Firewall (security group)
test-pvt
Storage (volumes)
1 volume(s) - 8 GiB
Free tier: In your first year of opening an AWS account, you get 750 hours per month of t2.micro instance usage (or t3.micro where
Cancel
Launch instance
Preview code
CloudShell
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences
ENG
IN
21:21
01-09-2025

Step 8: Testing

- **Step 1: Copying pem file into Public Instance**
    - From Local Copy ".Pem" File to the EC2 Instance

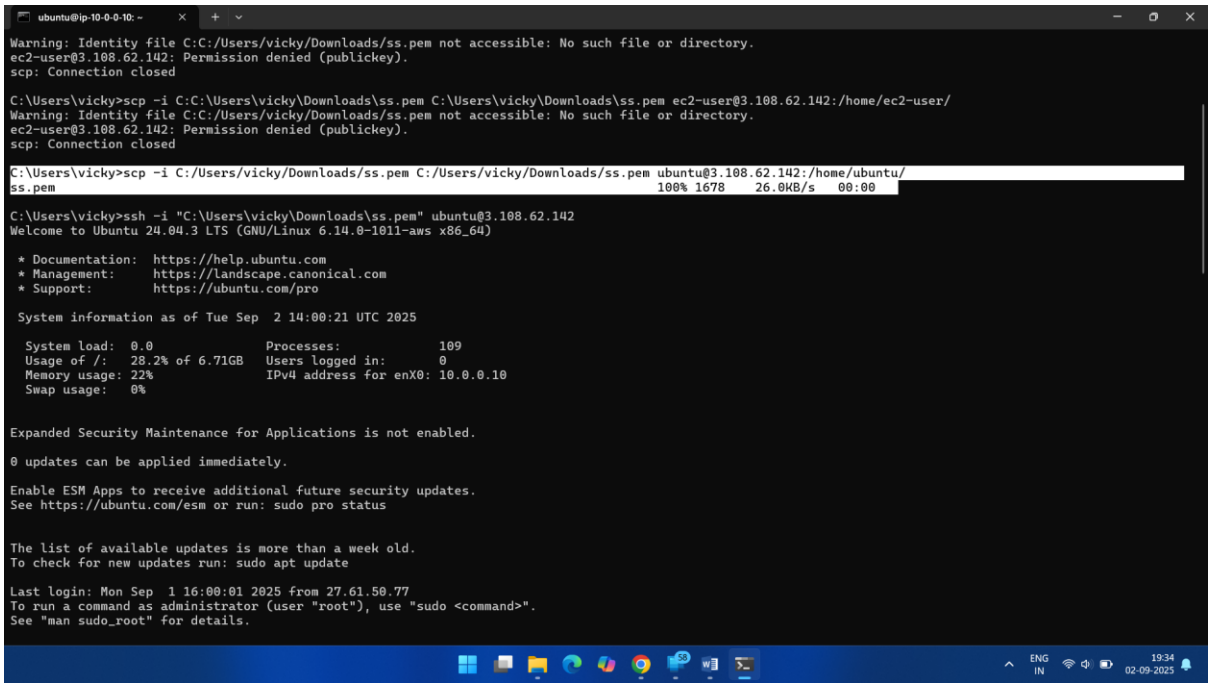
For **Amazon Linux** use the below code in Local:

```
scp -i C:/Users/vicky/Downloads/ss.pem C:/Users/vicky/Downloads/ss.pem ec2-user@<public-ec2-public-ip>:/home/ec2-user/
```

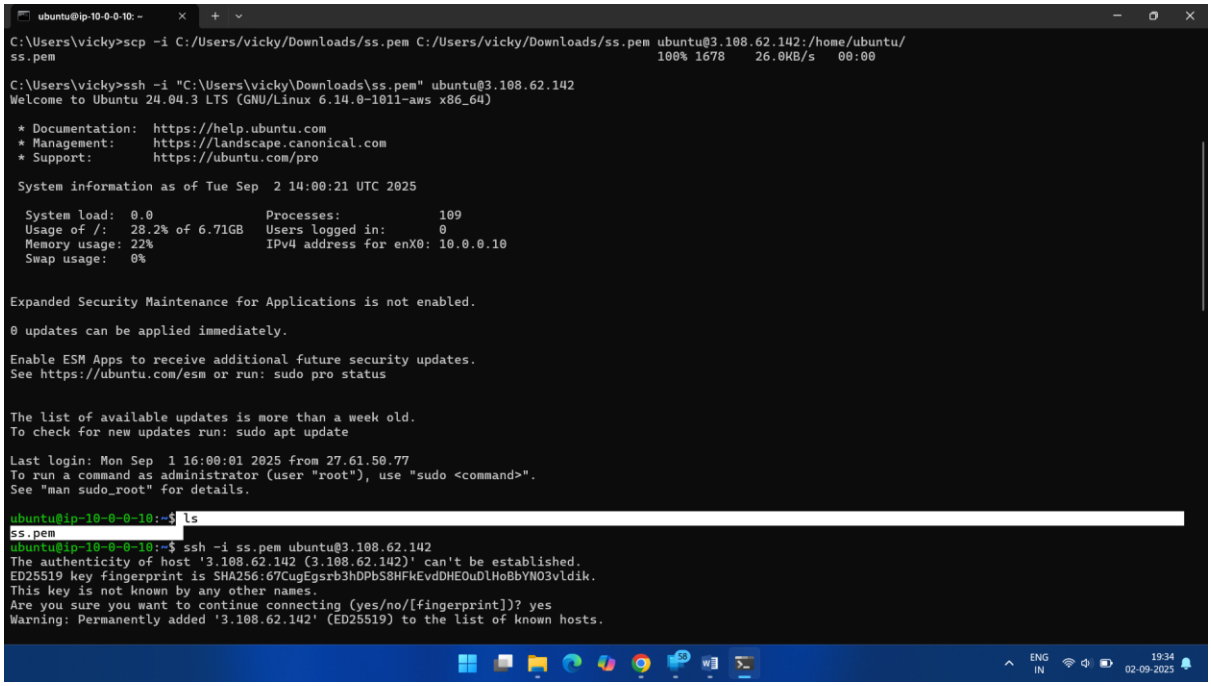
For **Ubuntu** use the below code in Local:

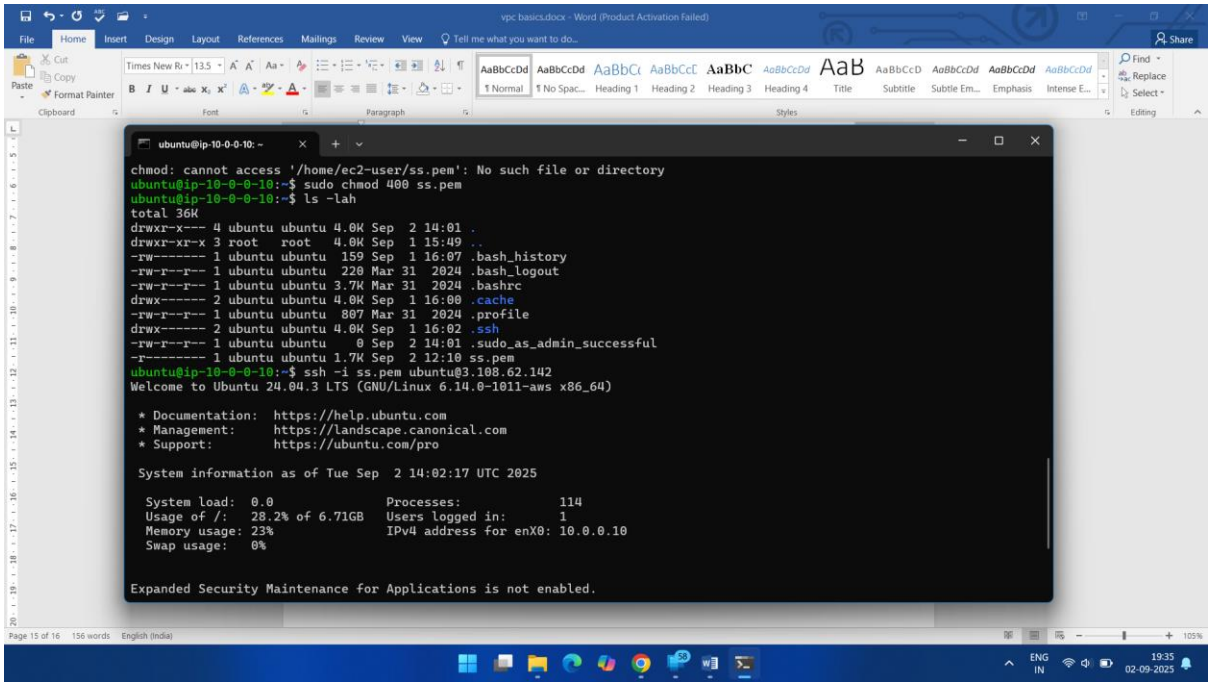
```
scp -i C:/Users/vicky/Downloads/ss.pem C:/Users/vicky/Downloads/ss.pem ubuntu@3.108.62.142:/home/ubuntu/
```

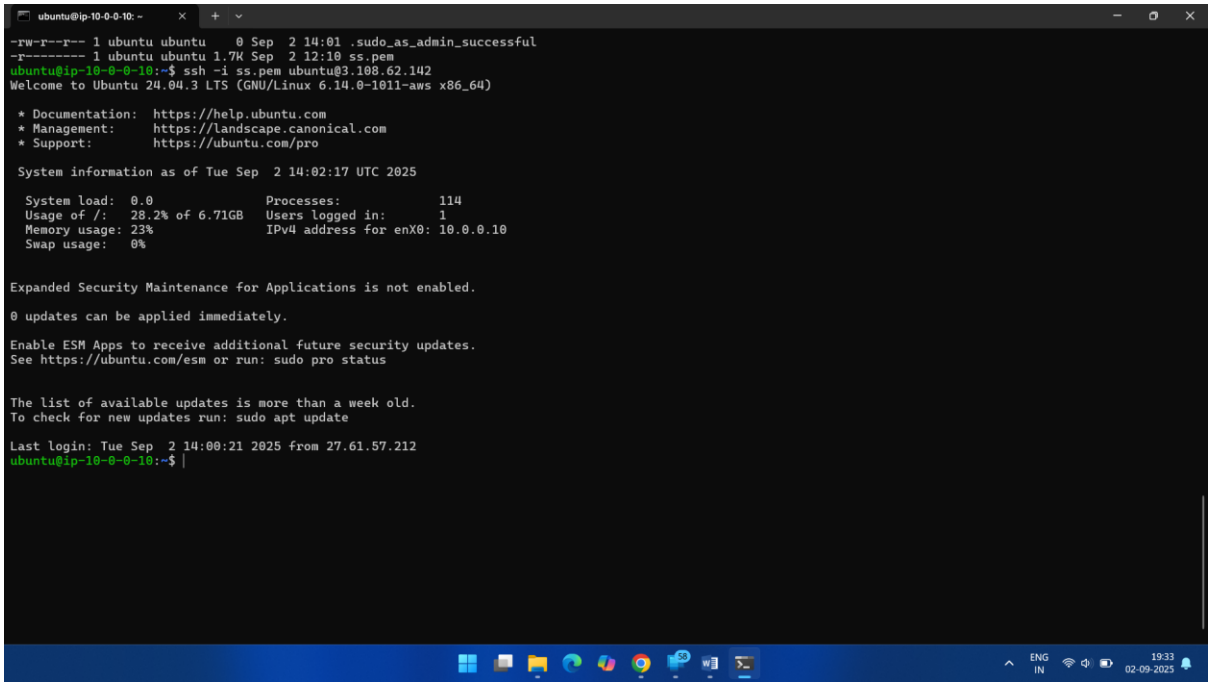
The above command will Store the **".pem"** file into the public EC2 instance in location : **"/home/ubuntu/"** from "**C:\Users\vicky\Downloads\ss.pem**".



- **Step 2: SSH into the public instance**
    - Command : `cd /home/ubuntu`
        - You can find the ".pem" file.
    - Command : `sudo chmod 400 <name>.pem`
        - For security purpose its mandatory to change the permissions
    - Command : `ssh -i <name>.pem ubuntu@3.108.62.142`
        - You can able to login to the private subnet. By which we can confirm by this process and login to the private instance.
    - Command : `curl ifconfig.me`
        - Using this command in public instance this'll return public ip of the instance.
        - While using this on private instance this'll return public ip of NAT gateway.


ubuntu@ip-10-0-0-10: ~
C:\Users\vicky>scp -i C:/Users/vicky/Downloads/ss.pem C:/Users/vicky/Downloads/ss.pem ubuntu@3.108.62.142:/home/ubuntu/
ss.pem 100% 1678 26.0KB/s 00:00
C:\Users\vicky>ssh -i "C:\Users\vicky\Downloads\ss.pem" ubuntu@3.108.62.142
Welcome to Ubuntu 24.04.3 LTS (GNU/Linux 6.14.0-1011-aws x86_64)
* Documentation: https://help.ubuntu.com
* Management: https://landscape.canonical.com
* Support: https://ubuntu.com/pro
System information as of Tue Sep 2 14:00:21 UTC 2025
System load: 0.0 Processes: 109
Usage of /: 28.2% of 6.71GB Users logged in: 0
Memory usage: 22% IPv4 address for enX0: 10.0.0.10
Swap usage: 0%
Expanded Security Maintenance for Applications is not enabled.
0 updates can be applied immediately.
Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status
The list of available updates is more than a week old.
To check for new updates run: sudo apt update
Last login: Mon Sep 1 16:00:01 2025 from 27.61.50.77
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.
ubuntu@ip-10-0-0-10:~$ ls
ss.pem
ubuntu@ip-10-0-0-10:~$ ssh -i ss.pem ubuntu@3.108.62.142
The authenticity of host '3.108.62.142 (3.108.62.142)' can't be established.
ED25519 key fingerprint is SHA256:67CugEgsrb3hDPbS8HFkEvdDHEOuDlHoBbYNO3vldik.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '3.108.62.142' (ED25519) to the list of known hosts.


vpc basics.docx - Word (Product Activation Failed)
ubuntu@ip-10-0-0-10: ~
chmod: cannot access '/home/ec2-user/ss.pem': No such file or directory
ubuntu@ip-10-0-0-10:~$ sudo chmod 400 ss.pem
ubuntu@ip-10-0-0-10:~$ ls -lah
total 36K
drwxr-x--- 4 ubuntu ubuntu 4.0K Sep 2 14:01 .
drwxr-xr-x 3 root root 4.0K Sep 1 15:49 ..
-rw------- 1 ubuntu ubuntu 159 Sep 1 16:07 .bash_history
-rw-r--r-- 1 ubuntu ubuntu 220 Mar 31 2024 .bash_logout
-rw-r--r-- 1 ubuntu ubuntu 3.7K Mar 31 2024 .bashrc
drwx------ 2 ubuntu ubuntu 4.0K Sep 1 16:00 .cache
-rw-r--r-- 1 ubuntu ubuntu 807 Mar 31 2024 .profile
drwx------ 2 ubuntu ubuntu 4.0K Sep 1 16:02 .ssh
-rw-r--r-- 1 ubuntu ubuntu 0 Sep 2 14:01 .sudo_as_admin_successful
-r-------- 1 ubuntu ubuntu 1.7K Sep 2 12:10 ss.pem
ubuntu@ip-10-0-0-10:~$ ssh -i ss.pem ubuntu@3.108.62.142
Welcome to Ubuntu 24.04.3 LTS (GNU/Linux 6.14.0-1011-aws x86_64)
* Documentation: https://help.ubuntu.com
* Management: https://landscape.canonical.com
* Support: https://ubuntu.com/pro
System information as of Tue Sep 2 14:02:17 UTC 2025
System load: 0.0 Processes: 114
Usage of /: 28.2% of 6.71GB Users logged in: 1
Memory usage: 23% IPv4 address for enX0: 10.0.0.10
Swap usage: 0%
Expanded Security Maintenance for Applications is not enabled.



Step 9: Testing

- Final Flow will be like