

## **Cloud & Security**

### **Lab Assignment 2**

**Name :- Suraj Kumar**

**PNR:- 240840127041**

- 1. Create a VPC in AWS with address as 172.20.0.0. Create a public subnet and a private subnet. Connect Internet Gateway to the public subnet. Create an EC2 instance connect it to the public subnet. Create another EC2 instance and connect it to private subnet. Display that you are able to connect using putty to the EC2 instance in the public subnet.**
- 2. Copy the key of the second EC2 instance to the EC2 instance in the public subnet. Display that you are able to ssh to the EC2 instance in the private subnet from the EC2 instance in the public subnet.**

#### **Step :1 Create a VPC**

In the "Create VPC" wizard:

Name tag: "MyVPC"

IPv4 CIDR block: 172.20.0.0/16

Leave other options as default (IPv6, Tenancy).

Click Create VPC.

Browser tabs: CreateVpc | VPC Console, ChatGPT, 5. Cloud - Google Drive

URL: ap-south-1.console.aws.amazon.com/vpcconsole/home?region=ap-south-1#CreateVpc:createMode=vpcOnly

Navigation: VPC > Your VPCs > Create VPC

### Create VPC [Info](#)

A VPC is an isolated portion of the AWS Cloud populated by AWS objects, such as Amazon EC2 instances.

#### VPC settings

**Resources to create** [Info](#)  
Create only the VPC resource or the VPC and other networking resources.

☒ VPC only ☐ VPC and more

**Name tag - optional**  
Creates a tag with a key of 'Name' and a value that you specify.

MyVPC

**IPv4 CIDR block** [Info](#)  
☒ IPv4 CIDR manual input  
☐ IPAM-allocated IPv4 CIDR block

IPv4 CIDR  
172.20.0.0  
CIDR block size must be between /16 and /28.

**IPv6 CIDR block** [Info](#)  
☒ No IPv6 CIDR block  
☐ IPAM-allocated IPv6 CIDR block  
☐ Amazon-provided IPv6 CIDR block  
☐ IPv6 CIDR owned by me

**Tenancy** [Info](#)  
Default

#### Tags

A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

Key	Value - optional	
Q Name X	Q MyVPC X	Remove tag

Add tag

You can add 49 more tags

Cancel Preview code **Create VPC**

CloudShell Feedback

## Step 2: Create Public and Private Subnets

After creating the VPC, go to the Subnets section.

Create the Public Subnet:

Click Create subnet.

Select the VPC created in Step 1.

Name tag: "PublicSubnet"

Availability Zone: Choose one (e.g., us-east-1a).

CIDR block: 172.20.1.0/24

Click Create subnet.

**Create subnet** [Info](#)

**VPC**

VPC ID  
Create subnets in this VPC.  
vpc-0db17f1760532f9f3 (MyVPC)

**Associated VPC CIDRs**

IPv4 CIDRs  
172.20.0.0/16

**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.  
PublicSubnet  
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.  
172.20.0.0/16

IPv4 subnet CIDR block  
172.20.1.0/24 256 IPs

**Tags - optional**

Key Value - optional  
Name PublicSubnet Remove  
Add new tag  
You can add 49 more tags.  
Remove  
Add new subnet

Cancel Create subnet

CloudShell Feedback

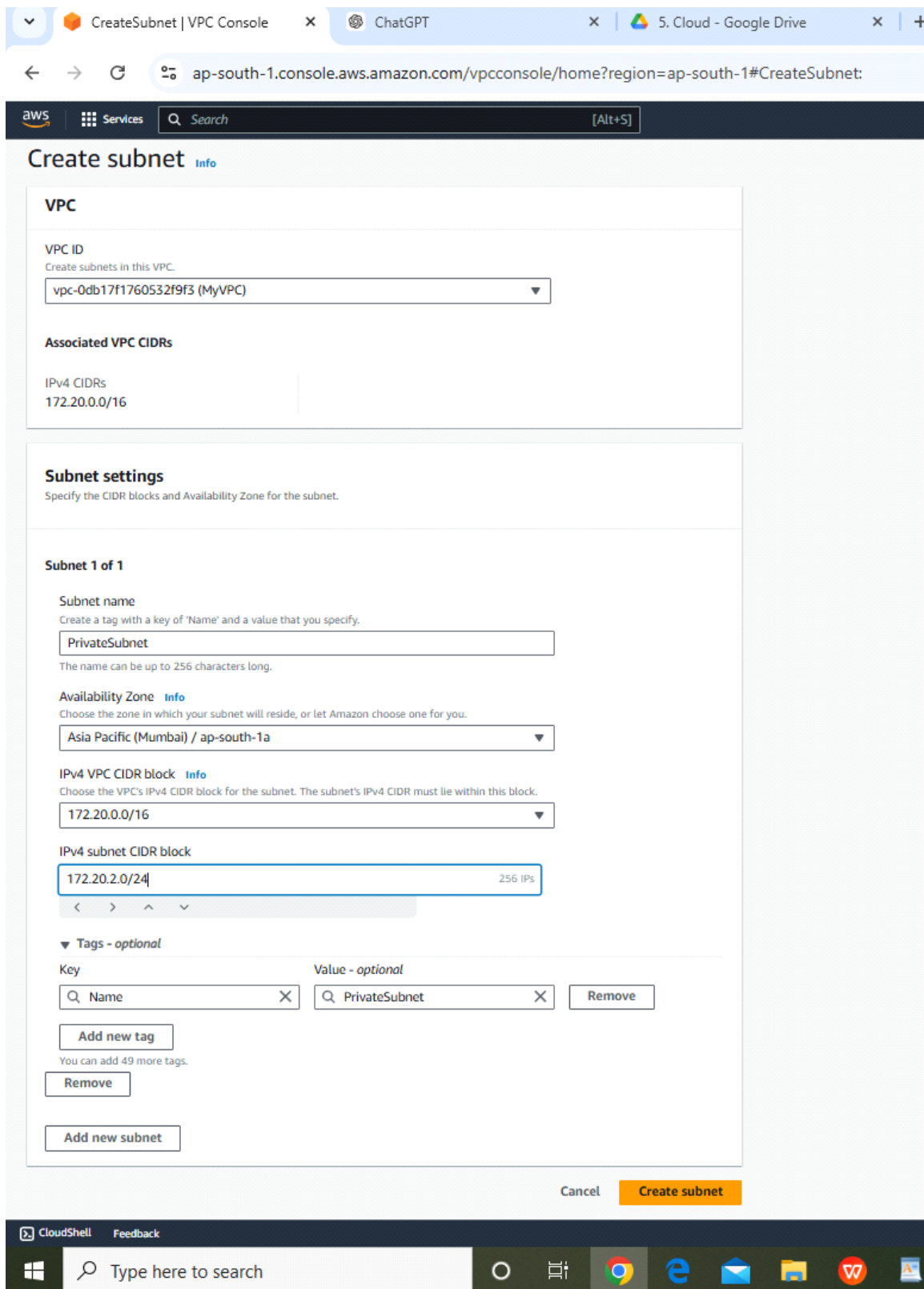
### **3 Create the Private Subnet:**

Click Create subnet again.

Select the same VPC.

Name tag: "PrivateSubnet"

Availability Zone: Choose the same or different (e.g., us-east-1b).CIDR block:  
172.20.2.0/24Click Create subnet.



### Step 3: Create an Internet Gateway (IGW)

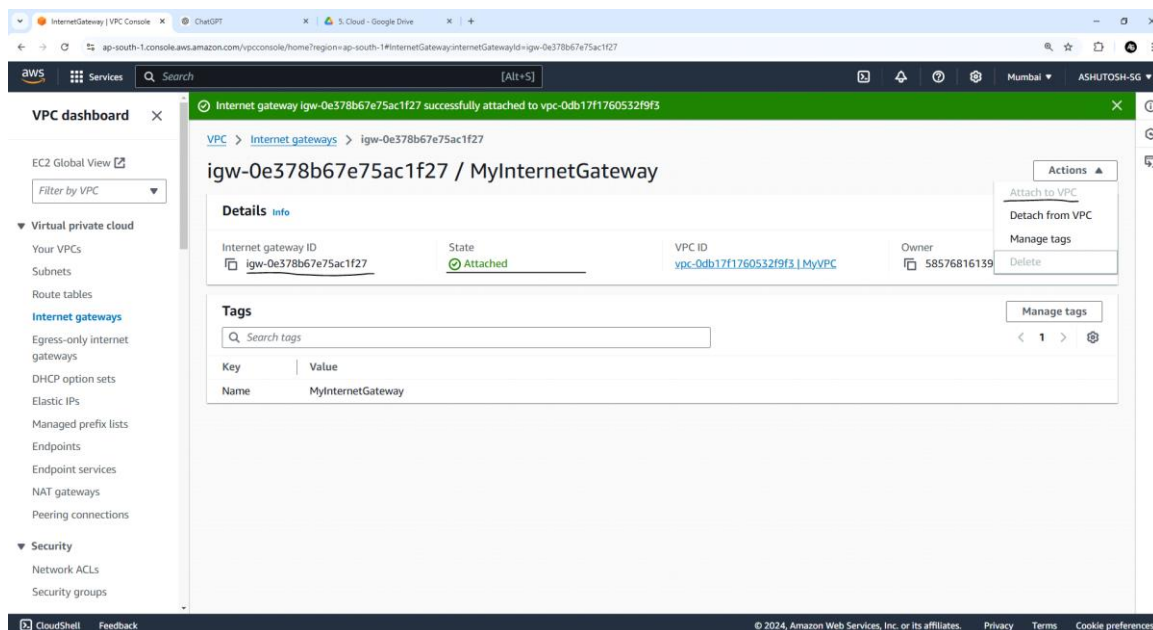
Go to the Internet Gateways section in the VPC Dashboard.

Click Create internet gateway.

Name tag: "MyInternetGateway"

Click Create and then Attach to VPC.

Select the VPC ("MyVPC") and click Attach.



### Step 4: Create Route Tables

*Create route tables to control traffic flow within your VPC.*

#### 1. Create Route Table for Public Subnet:

- In the VPC dashboard, go to Route Tables.
- Click Create route table.
- Name tag: Enter a name (e.g., "Public-Route-Table").
- VPC: Select your VPC.
- Click Create.

- *Edit Routes:*
  - Select the newly created route table and go to the Routes tab.
  - Click Edit routes > Add route.
  - Destination: Enter 0.0.0.0/0 (for all traffic).
  - Target: Select your Internet Gateway.
  - Click Save routes.
- Associate Route Table with Public Subnet:
  - Go to the Subnet Associations tab.
  - Click Edit subnet associations.
  - Select your Public Subnet and click Save.

## **2. Create Route Table for Private Subnet:**

- Click Create route table again.
- Name tag: Enter a name (e.g., "Private-Route-Table").
- VPC: Select your VPC.
- Click Create.
- Edit Routes and Associate Route Table with Public Subnet:
  - Private subnets typically don't have direct internet access, so leave the default route.

CreateRouteTable | VPC Console

ChatGPT

5. Cloud - Google Drive

ap-south-1.console.aws.amazon.com/vpcconsole/home?region=ap-south-1#CreateRouteTable:

aws

Services

Search

[Alt+S]

[VPC](#) > [Route tables](#) > Create route table

## Create route table [Info](#)

A route table specifies how packets are forwarded between the subnets within your VPC, the internet, and your VPN connection.

### Route table settings

**Name - optional**  
Create a tag with a key of 'Name' and a value that you specify.

Public Subnet Route Table

**VPC**  
The VPC to use for this route table.

vpc-0db17f1760532f9f3 (MyVPC)

### Tags

A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

**Key**

Q Name

X

**Value - optional**

Q Public Subnet Route Table

X

Remove

Add new tag

You can add 49 more tags.

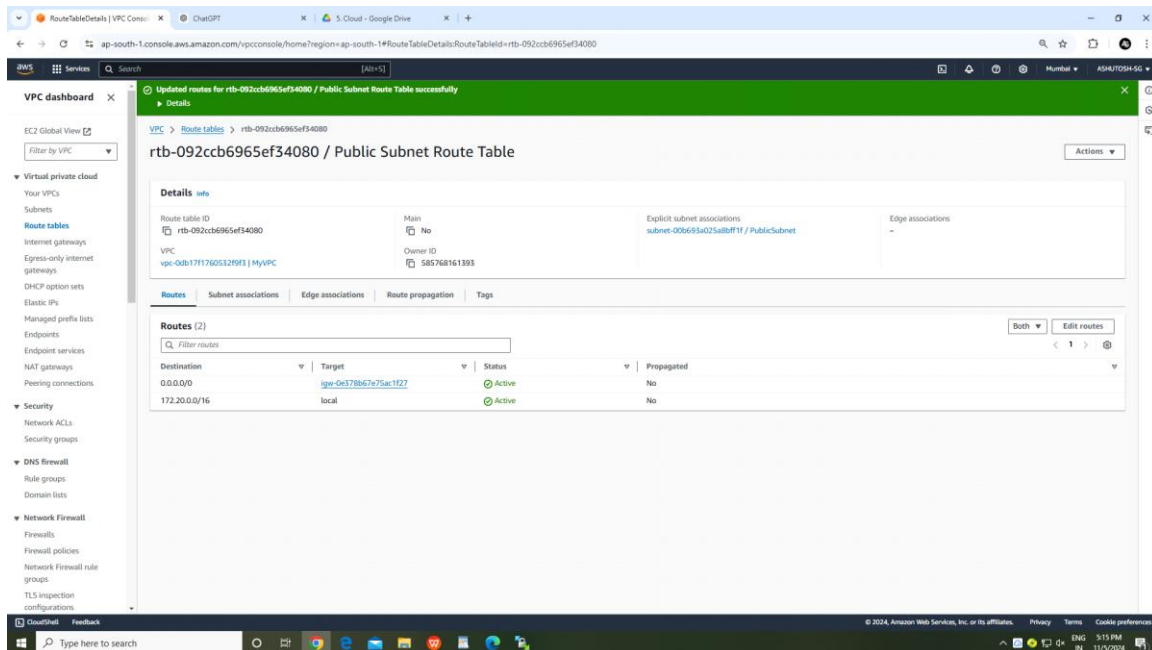
Cancel

Create route table

CloudShell

Feedback





## Step 5: Launch EC2 Instance in the Public Subnet

Go to the EC2 Dashboard and click Launch Instance.

Select an Amazon Machine Image (AMI) (e.g., Amazon Linux 2).

Choose an Instance Type (e.g., t2.micro).

In the Network section, select the VPC ("MyVPC") and Subnet ("PublicSubnet").

In the Configure Security Group section:

Create a new security group or select an existing one.

Add an inbound rule to allow SSH (port 22) from your IP address.

Key Pair: Create a new key pair ((.pem)download and save it; you will need it for SSH).

Click Launch Instance.

Launch an instance | EC2 | ap-south-1

public\_instance\_key Create new key pair

**Network settings** Info

VPC - required Info  
vpc-0db17f1760532f9f3 (MyVPC) 172.20.0.0/16

Subnet Info  
subnet-00b693a025a8bfff1f PublicSubnet  
VPC: vpc-0db17f1760532f9f3 Owner: 585768161393  
Availability Zone: ap-south-1a Zone type: Availability Zone  
IP addresses available: 251 CIDR: 172.20.1.0/24

Auto-assign public IP Info  
Enable

Additional charges apply when outside of free tier allowance

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

Security group name - required  
publiid

Description - required Info  
launch-wizard-1 created 2024-11-05T11:50:25.804Z

Inbound Security Group Rules  
Security group rule 1 (TCP, 22, 0.0.0.0/0) Remove

Type Info Protocol Info Port range Info  
ssh TCP 22

Source type Info Source Info Description - optional Info  
Anywhere 0.0.0.0/0 e.g. SSH for admin desktop

Rules with source of 0.0.0.0/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.

**Summary**

Number of instances Info  
1

Software Image (AMI)  
Canonical, Ubuntu, 24.04, amd64...read more  
ami-0dee22c13ea7a9a67

Virtual server type (instance type)  
t2.micro

Firewall (security group)  
New security group

Storage (volumes)  
1 volume(s) - 8 GiB

Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 750 hours of public IPv4 address usage per month, 30 GiB of EBS storage, 2 million I/Os, 1 GB of snapshots, and 100 GB of bandwidth to the internet.

Cancel Launch Instance Preview code

## Step 6: Launch EC2 Instance in the Private Subnet

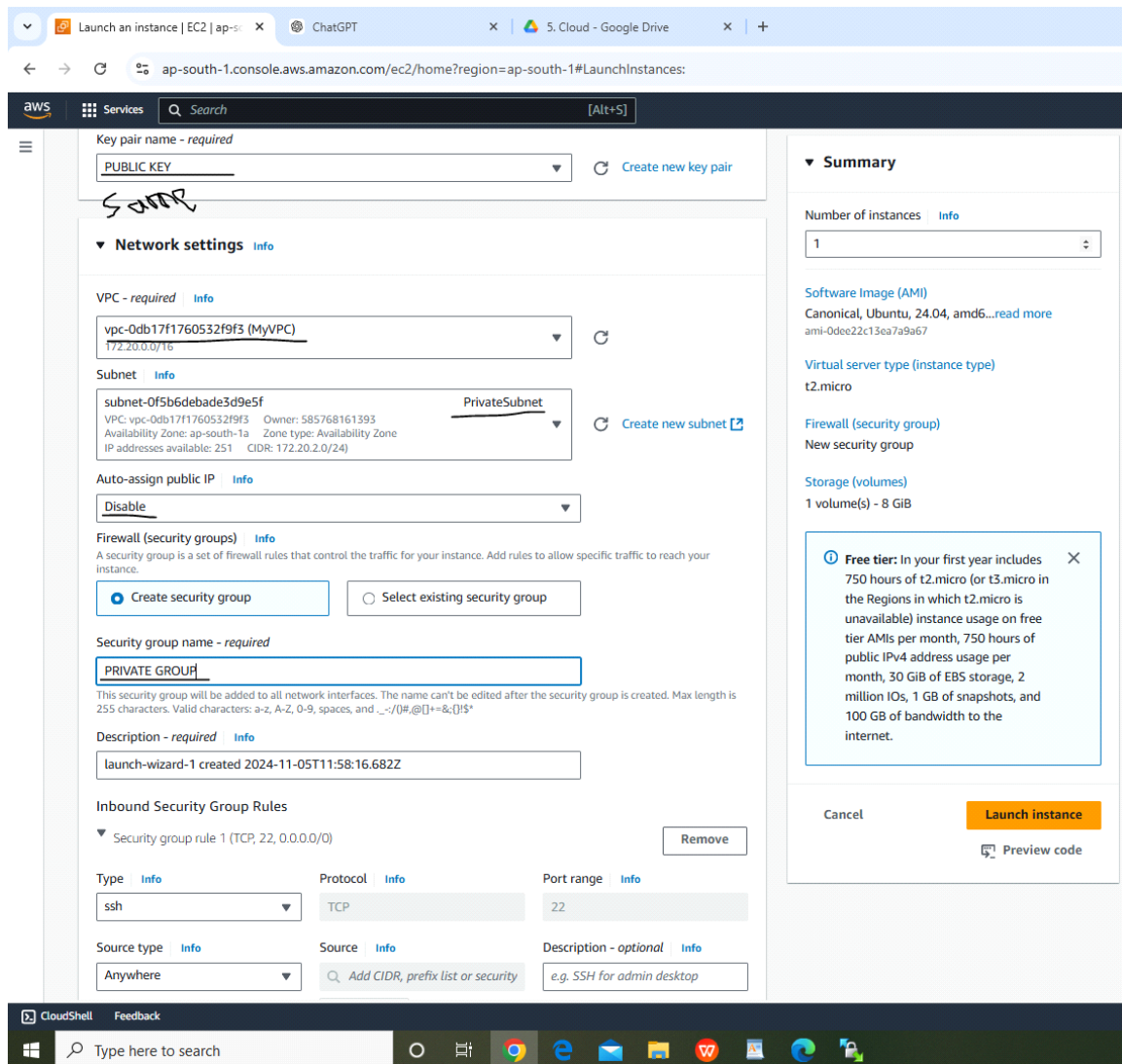
Follow the same steps as above to create another EC2 instance in the Private Subnet.

Only change is disable auto assign IP and choose private subnet

## EDIT NETWORK SETTINGS

In the Network section, select the VPC ("MyVPC") and Subnet ("PrivateSubnet").

Click Launch Instance.



USE PUTTY GEN TO CONVERT .PEM to .PPK

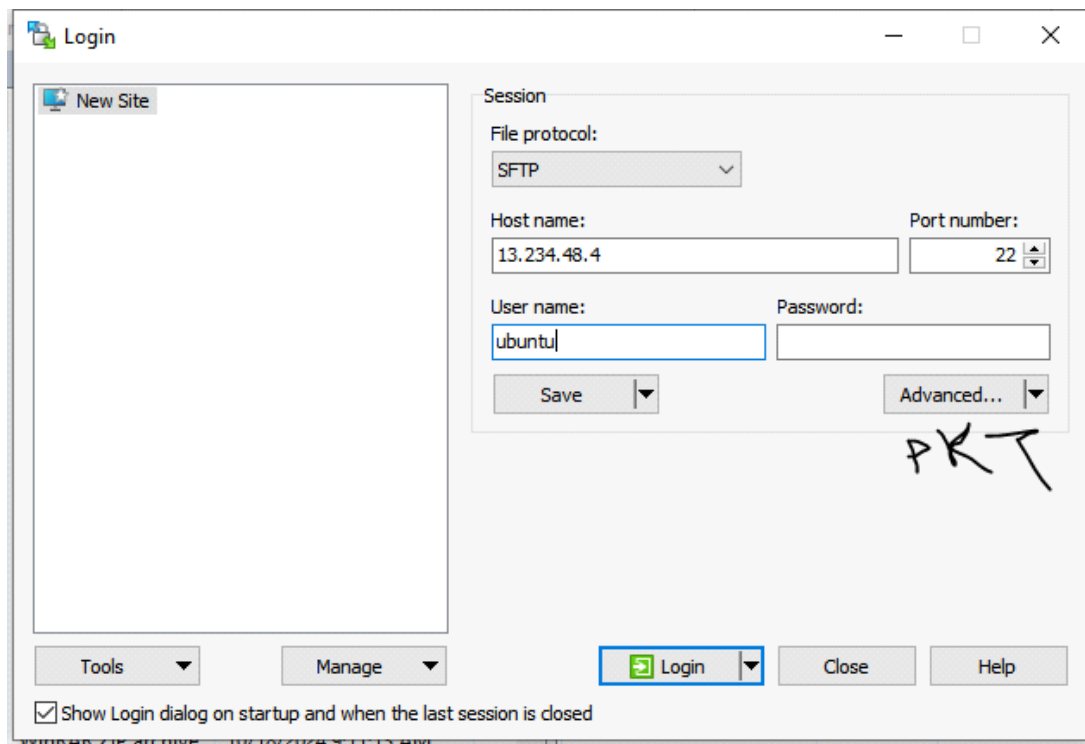
Step 7: Connect to the Public EC2 Instance via SSH (using Putty)

Open Putty (Windows) and enter the Public IP of the public EC2 instance.

Under Connection → SSH → Auth, browse to the private key file (in .pem format) for the public EC2 instance and select it.

Click Open to connect to the EC2 instance. You should be logged into the EC2 instance in the PublicSubnet.

by **using winSCP** copy pem file to local to ec2 instance



move pem file to the public instance BY DRAG AND DROP

Last Step:-

```
ubuntu@ip-172-20-1-231:~$ ls
```

```
'PUBLIC KEY.pem'
```

```
ubuntu@ip-172-20-1-231:~$ chmod 400 PUBLIC\ KEY.pem
```

```
ubuntu@ip-172-20-1-231:~$ ssh -i "PUBLIC KEY.pem" 172.20.2.223
```

**Finally we are connected to private instance via public instance**

```
ubuntu@ip-172-20-2-223: ~  
ubuntu@ip-172-20-1-231:~$ ls  
'PUBLIC KEY.pem'  
ubuntu@ip-172-20-1-231:~$ chmod 400 PUBLIC\ KEY.pem  
ubuntu@ip-172-20-1-231:~$ ssh -i "PUBLIC KEY.pem" 172.20.2.223  
Welcome to Ubuntu 24.04.1 LTS (GNU/Linux 6.8.0-1016-aws x86_64)  
  
* Documentation:  https://help.ubuntu.com  
* Management:    https://landscape.canonical.com  
* Support:        https://ubuntu.com/pro  
  
System information as of Tue Nov  5 12:35:38 UTC 2024  
  
System load:  0.0                Processes:            103  
Usage of /:   22.8% of 6.71GB    Users logged in:     0  
Memory usage: 21%                IPv4 address for enX0: 172.20.2.223  
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
```