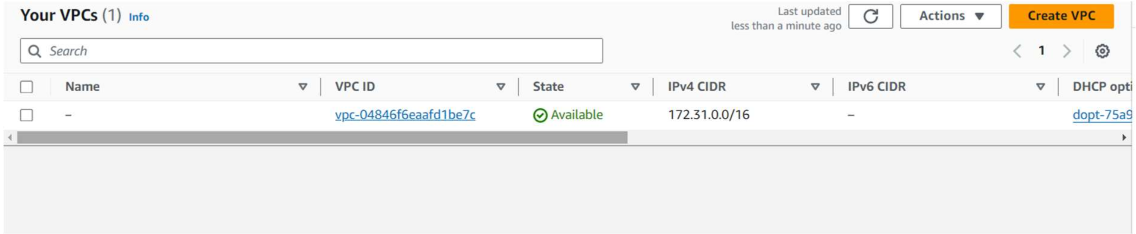
1. Create a VPC

- **Step 1**: Log in to the AWS Management Console and go to the VPC Dashboard.
- **Step 2**: Click on **Create VPC**.



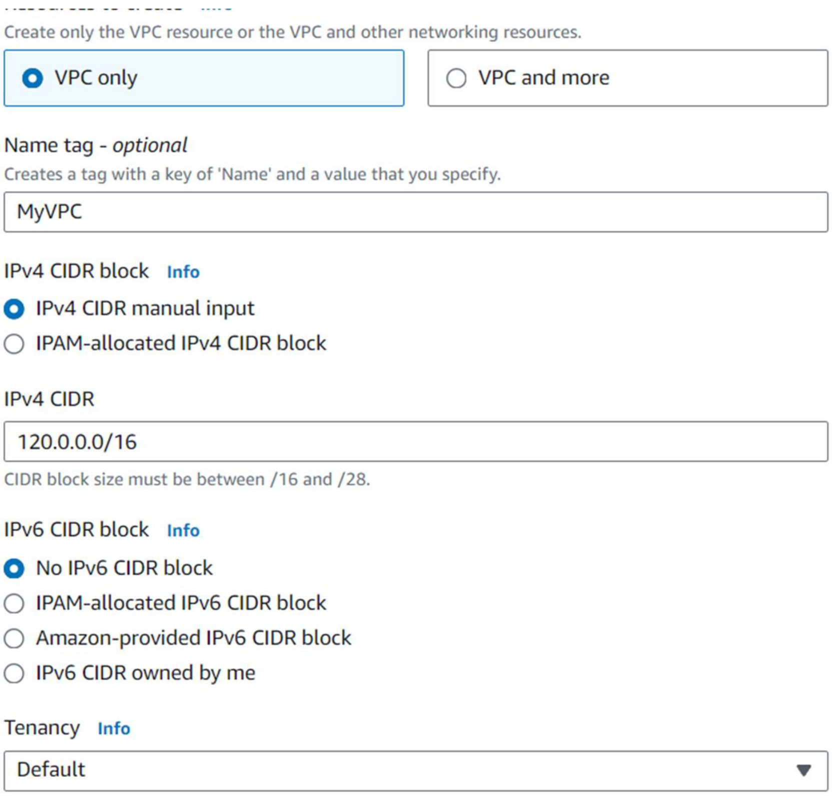
Your VPCs (1) Info

Last updated less than a minute ago | Actions ▼ | Create VPC

Search

□	Name	VPC ID	State	IPv4 CIDR	IPv6 CIDR	DHCP opti
□	–	vpc-04846f6eaafd1be7c	✓ Available	172.31.0.0/16	–	dopt-75a9

- **Step 3**: Choose **VPC only**.
- **Step 4**: Enter the following details:
  - **Name tag**: (e.g., MyVPC)
  - **IPv4 CIDR block**: 120.0.0.0/16



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

120.0.0.0/16

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

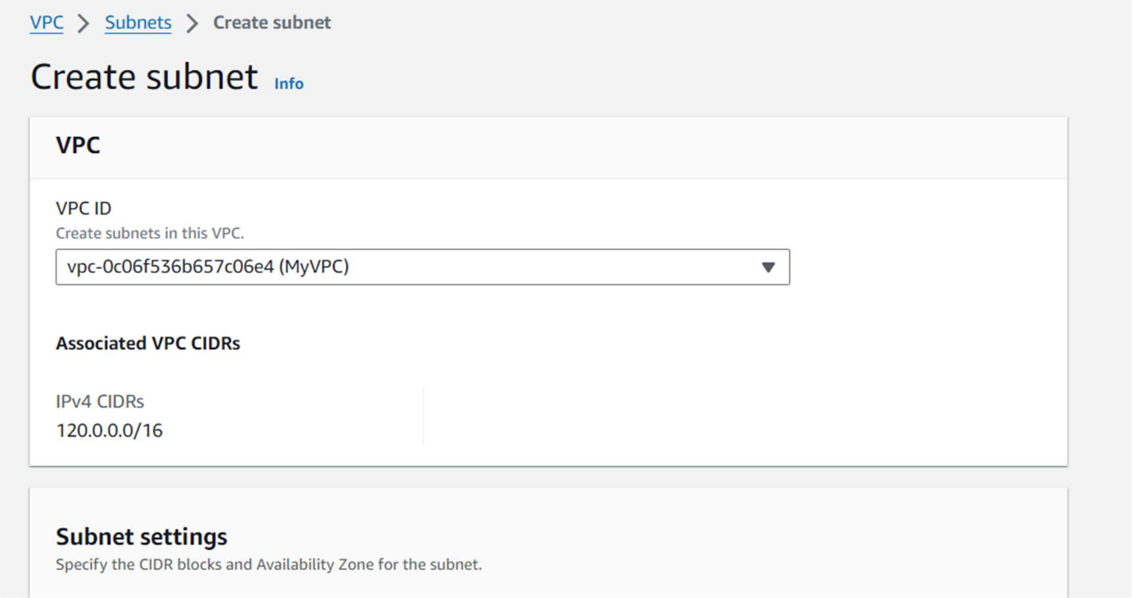
Tenancy Info

Default ▼

- **Step 5**: Click **Create VPC**.

2. Create Subnets

- **Step 1**: In the VPC Dashboard, click on **Subnets** in the left navigation pane, then click **Create Subnet**.
- **Step 2**: Select the VPC you just created.



VPC > Subnets > Create subnet

Create subnet Info

VPC

VPC ID

Create subnets in this VPC.

vpc-0c06f536b657c06e4 (MyVPC)

Associated VPC CIDRs

IPv4 CIDRs

120.0.0.0/16

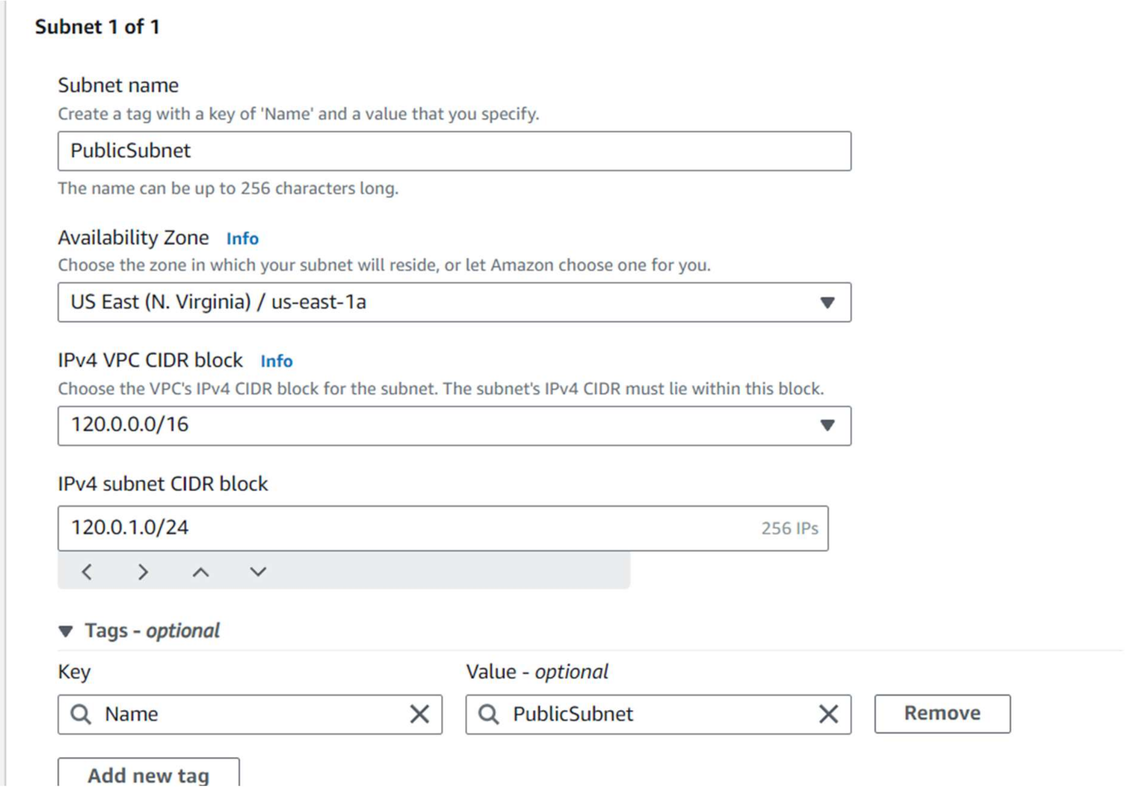
Subnet settings

Specify the CIDR blocks and Availability Zone for the subnet.

- **Step 3**: Create the public and private subnets:

Public Subnet

  - **Name tag**: (e.g., PublicSubnet)
  - **Availability Zone**: Choose one (e.g., us-east-1a).
  - **IPv4 CIDR block**: (e.g., 120.0.1.0/24)
  - **Step 4**: Click **Create 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.

US East (N. Virginia) / us-east-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.

120.0.0.0/16

IPv4 subnet CIDR block

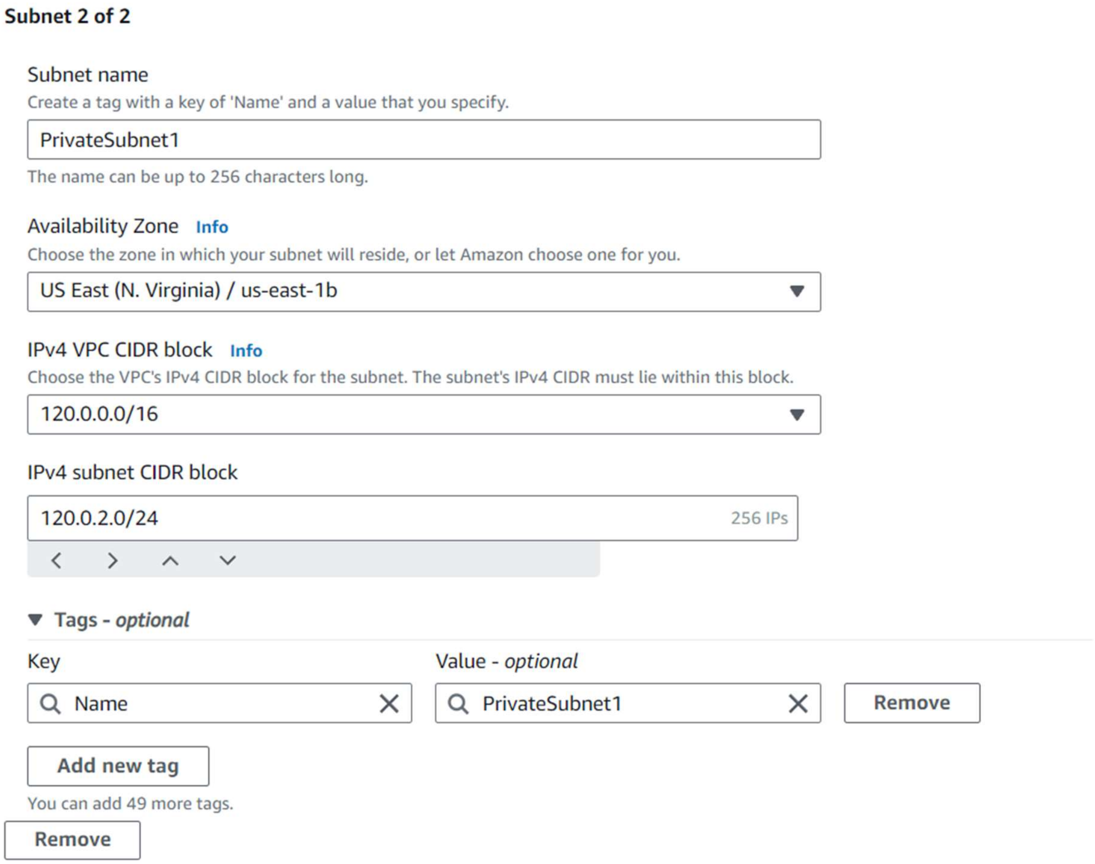
120.0.1.0/24 256 IPs

Tags - *optional*

Key	Value - *optional*
Name	PublicSubnet

Private Subnet 1

- **Name tag**: (e.g., PrivateSubnet1)
- **Availability Zone**: Choose another one (e.g., us-east-1b).
- **IPv4 CIDR block**: (e.g., 120.0.2.0/24)
- **Step 4**: Click **Create Subnet**.



Subnet 2 of 2

Subnet name

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

PrivateSubnet1

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.

US East (N. Virginia) / us-east-1b

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.

120.0.0.0/16

IPv4 subnet CIDR block

120.0.2.0/24 256 IPs

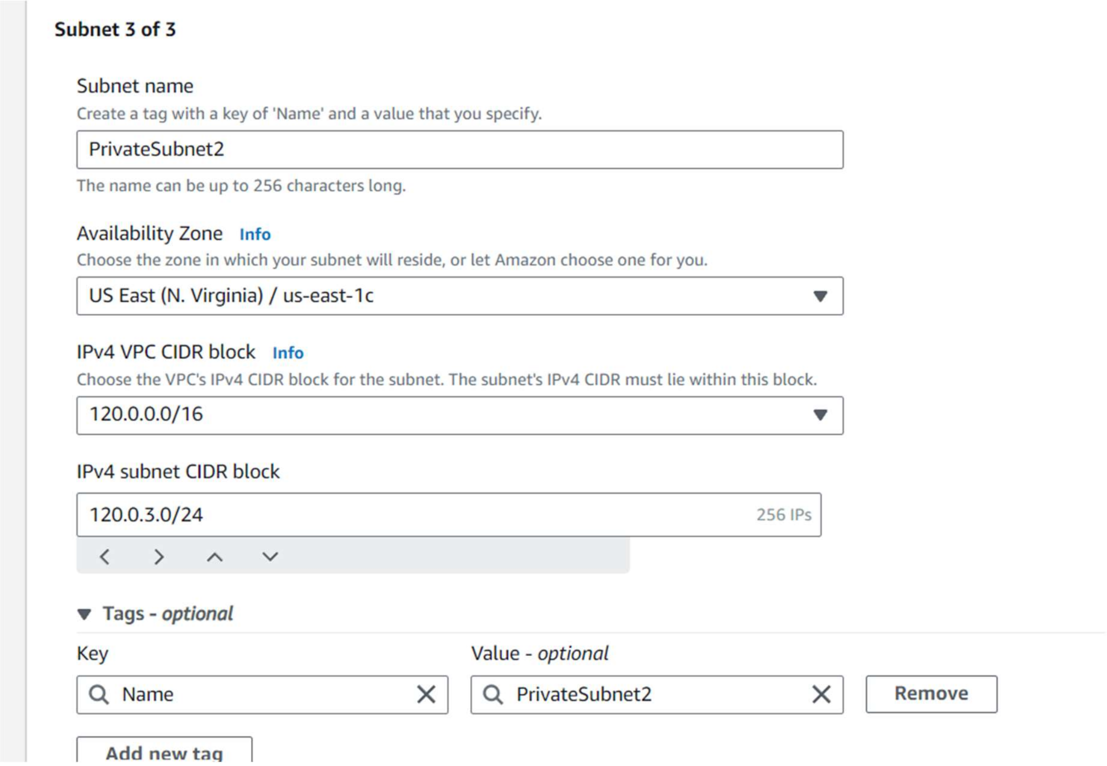
▼ Tags - *optional*

Key	Value - *optional*
Name	PrivateSubnet1

You can add 49 more tags.

Private Subnet 2

- **Name tag**: (e.g., PrivateSubnet2)
- **Availability Zone**: Choose the third (e.g., us-east-1c).
- **IPv4 CIDR block**: (e.g., 120.0.3.0/24)



Subnet 3 of 3

Subnet name

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

PrivateSubnet2

The name can be up to 256 characters long.

Availability Zone

Choose the zone in which your subnet will reside, or let Amazon choose one for you.

US East (N. Virginia) / us-east-1c

IPv4 VPC CIDR block

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

120.0.0.0/16

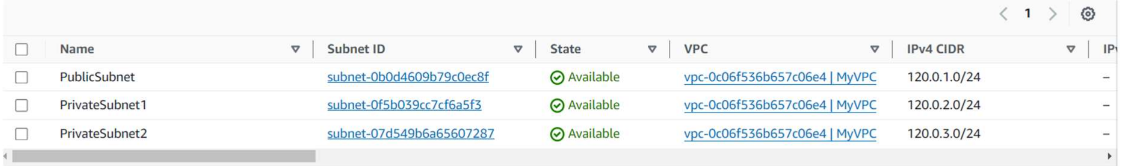
IPv4 subnet CIDR block

120.0.3.0/24 256 IPs

▼ Tags - *optional*

Key	Value - *optional*	
Name	PrivateSubnet2	Remove

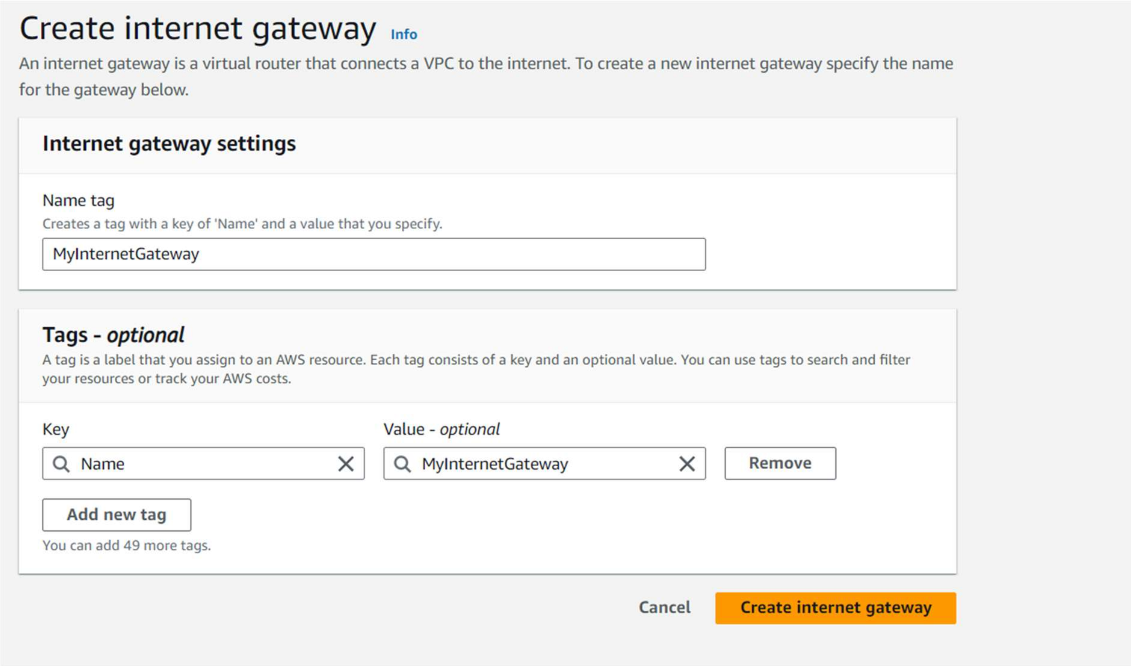
- **Step 4**: Click **Create Subnet**.



☐	Name	Subnet ID	State	VPC	IPv4 CIDR	IP
☐	PublicSubnet	subnet-0b0d4609b79c0ec8f	✓ Available	vpc-0c06f536b657c06e4 | MyVPC	120.0.1.0/24	–
☐	PrivateSubnet1	subnet-0f5b039cc7cf6a5f3	✓ Available	vpc-0c06f536b657c06e4 | MyVPC	120.0.2.0/24	–
☐	PrivateSubnet2	subnet-07d549b6a65607287	✓ Available	vpc-0c06f536b657c06e4 | MyVPC	120.0.3.0/24	–

3. Create an Internet Gateway and Attach it to the VPC

- **Step 1**: In the VPC Dashboard, click on **Internet Gateways** in the left navigation pane, then click **Create internet gateway**.
- **Step 2**: Enter a name tag (e.g., MyInternetGateway), then click **Create internet gateway**.



Create internet gateway Info

An internet gateway is a virtual router that connects a VPC to the internet. To create a new internet gateway specify the name for the gateway below.

Internet gateway settings

Name tag

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

MyInternetGateway

Tags - *optional*

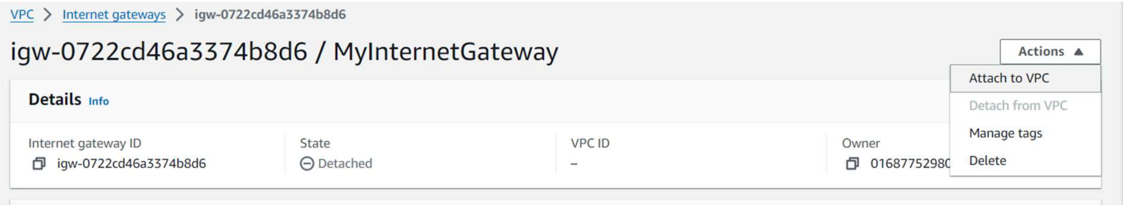
A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

Key	Value - *optional*	
Name	MyInternetGateway	Remove

Add new tag

You can add 49 more tags.

- **Step 3**: Click **Attach to VPC**, select the VPC you created, and click **Attach internet gateway**.



VPC > Internet gateways > igw-0722cd46a3374b8d6

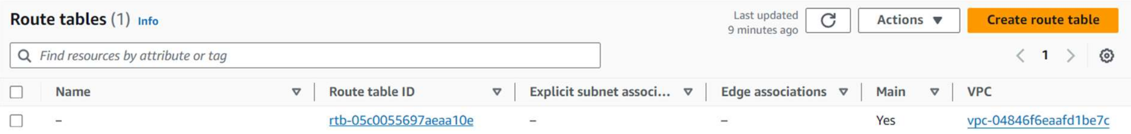
igw-0722cd46a3374b8d6 / MyInternetGateway

Details Info

Internet gateway ID	State	VPC ID	Owner
igw-0722cd46a3374b8d6	Detached	–	01687752980

4. Create a Route Table for the Public Subnet

- **Step 1**: In the VPC Dashboard, click on **Route Tables** in the left navigation pane, then click **Create route table**.

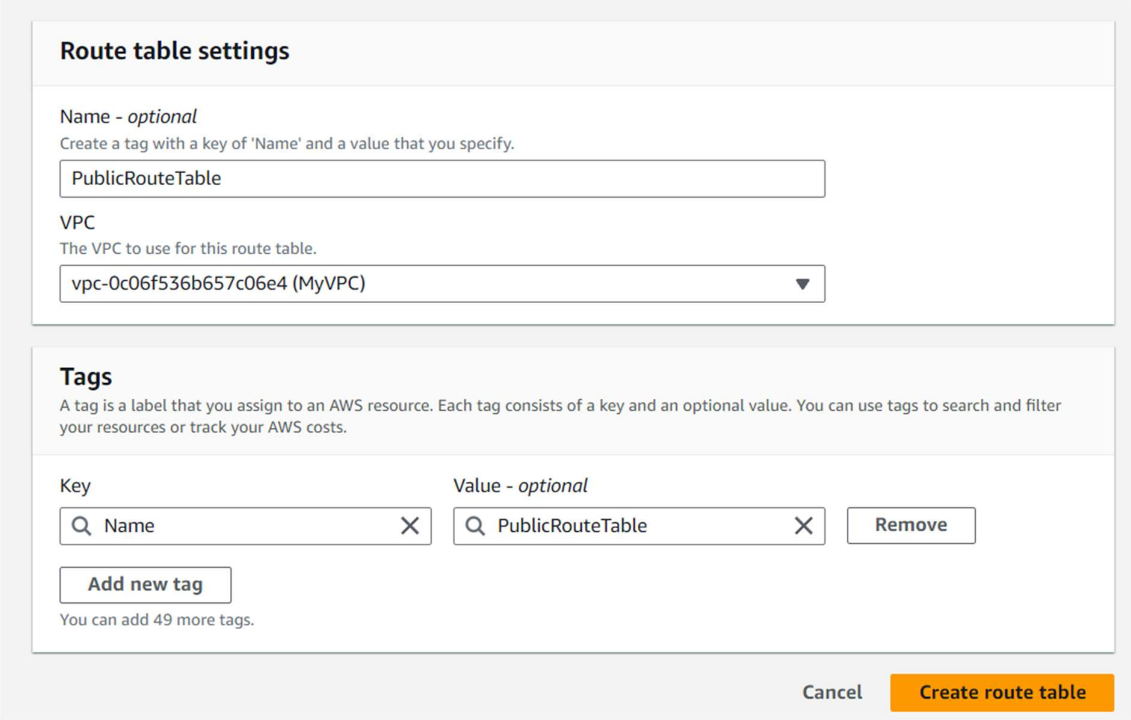


Route tables (1) Info

Last updated 9 minutes ago

□	Name	Route table ID	Explicit subnet associ...	Edge associations	Main	VPC
□	–	rtb-05c0055697aeaa10e	–	–	Yes	vpc-04846f6eaafd1be7c

- **Step 2**: Select the VPC you created, and enter a name tag (e.g., PublicRouteTable).



Route table settings

Name - *optional*

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

PublicRouteTable

VPC

The VPC to use for this route table.

vpc-0c06f536b657c06e4 (MyVPC)

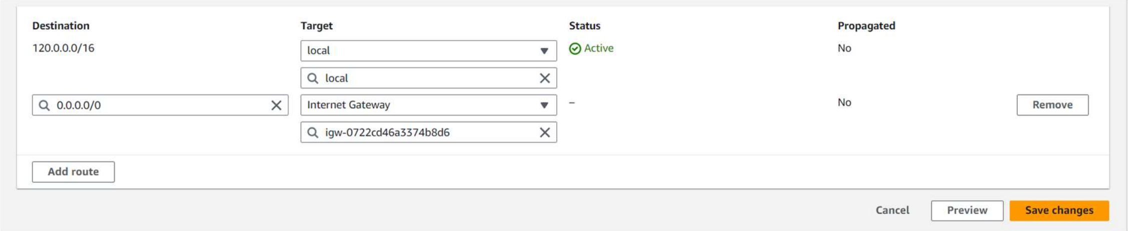
Tags

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	Value - *optional*
Name	PublicRouteTable

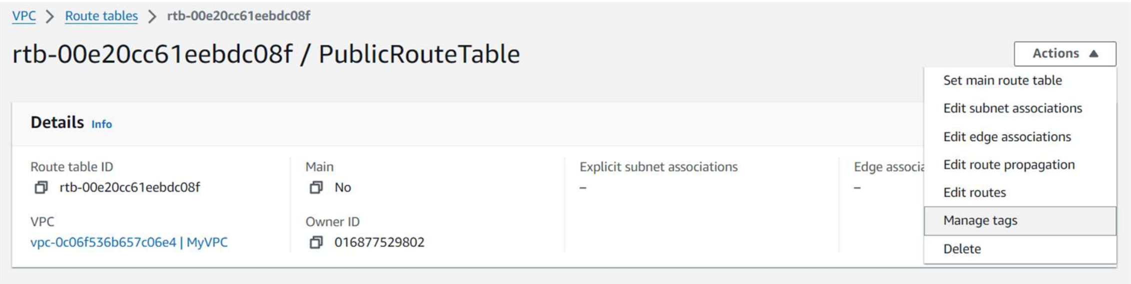
You can add 49 more tags.

- **Step 3**: Click **Create route table**.
- **Step 4**: Select the newly created route table, and under the **Routes** tab, click **Edit routes**.
- **Step 5**: Click **Add route**:
    - **Destination**: 0.0.0.0/0
    - **Target**: Select your Internet Gateway.
- **Step 6**: Click **Save changes**.



Destination	Target	Status	Propagated
120.0.0.0/16	local local	Active	No
0.0.0.0/0	Internet Gateway igw-0722cd46a3374b8d6	-	No

- **Step 7**: Under the **Subnets associations** tab, click **Edit subnet associations**



VPC > Route tables > rtb-00e20cc61eebdc08f

rtb-00e20cc61eebdc08f / PublicRouteTable

Details Info

Route table ID: rtb-00e20cc61eebdc08f

Main: No

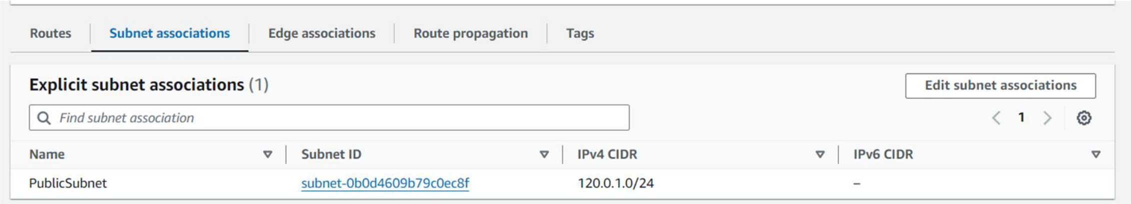
Explicit subnet associations: –

Edge associ: –

VPC: vpc-0c06f536b657c06e4 | MyVPC

Owner ID: 016877529802

- and select your public subnet.

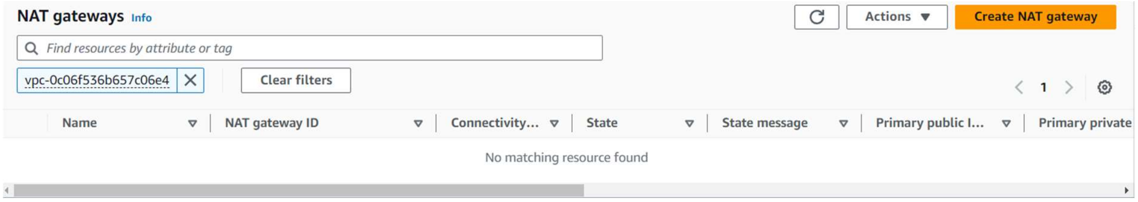


Explicit subnet associations (1)

Name	Subnet ID	IPv4 CIDR	IPv6 CIDR
PublicSubnet	subnet-0b0d4609b79c0ec8f	120.0.1.0/24	–

5. Create a NAT Gateway

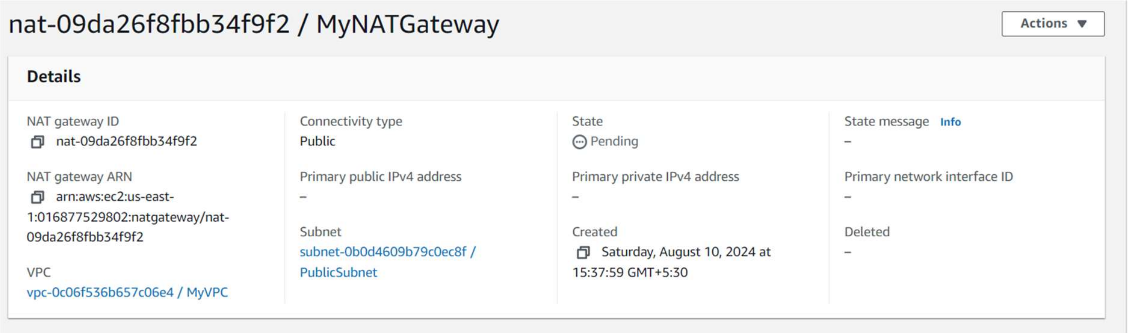
- **Step 1**: In the VPC Dashboard, click on **NAT Gateways** in the left navigation pane, then click **Create NAT gateway**.



Name	NAT gateway ID	Connectivity...	State	State message	Primary public I...	Primary private

No matching resource found

- **Step 2**: Enter the following details:
  - **Name tag**: (e.g., MyNATGateway)
  - **Subnet**: Select your public subnet.
  - **Elastic IP allocation ID**: Allocate a new Elastic IP or select an existing one.
- **Step 3**: Click **Create NAT gateway**.



nat-09da26f8fbb34f9f2 / MyNATGateway

Details

NAT gateway ID: nat-09da26f8fbb34f9f2

NAT gateway ARN: arn:aws:ec2:us-east-1:016877529802:natgateway/nat-09da26f8fbb34f9f2

VPC: vpc-0c06f536b657c06e4 / MyVPC

Connectivity type: Public

Primary public IPv4 address: –

Subnet: subnet-0b0d4609b79c0ec8f / PublicSubnet

State: Pending

Primary private IPv4 address: –

Created: Saturday, August 10, 2024 at 15:37:59 GMT+5:30

State message Info: –

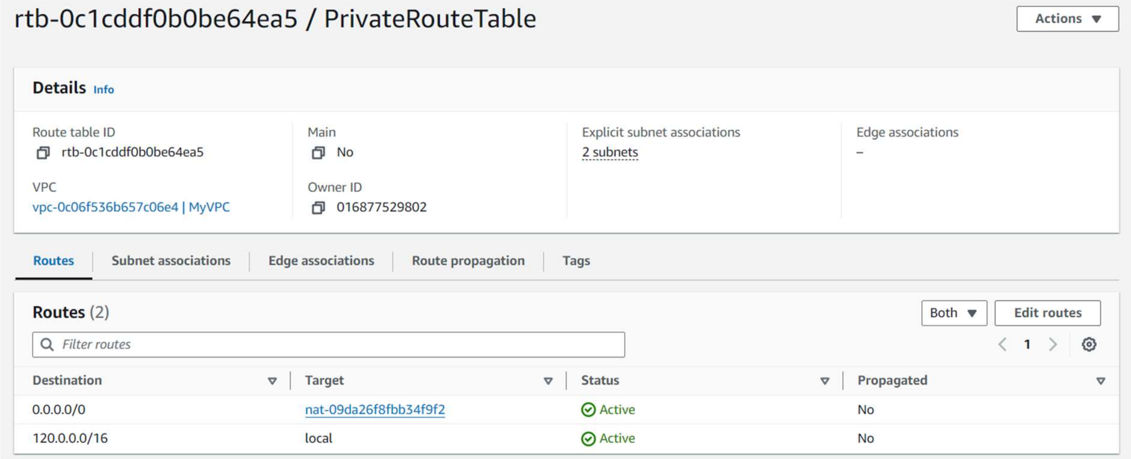
Primary network interface ID: –

Deleted: –

6. Create a Route Table for the Private Subnets

- **Step 1**: In the VPC Dashboard, click on **Route Tables** in the left navigation pane, then click **Create route table**.
- **Step 2**: Select the VPC you created, and enter a name tag (e.g., PrivateRouteTable).
- **Step 3**: Click **Create route table**.
- **Step 4**: Select the newly created route table, and under the **Routes** tab, click **Edit routes**.
- **Step 5**: Click **Add route**:
  - **Destination**: 0.0.0.0/0
  - **Target**: Select your NAT Gateway.

- **Step 6**: Click **Save changes**.



rtb-0c1cddf0b0be64ea5 / PrivateRouteTable

Details

Route table ID: rtb-0c1cddf0b0be64ea5

Main: No

Explicit subnet associations: 2 subnets

Edge associations: –

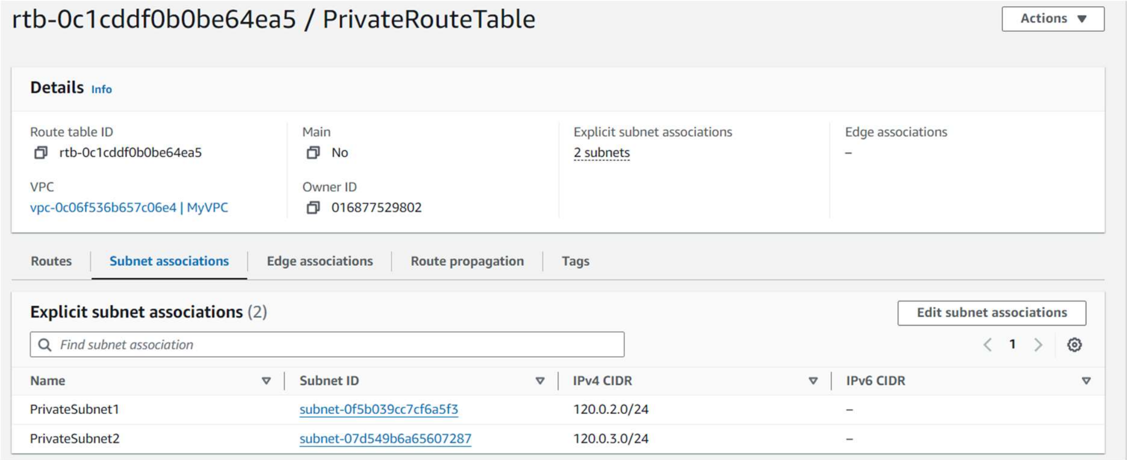
VPC: vpc-0c06f536b657c06e4 | MyVPC

Owner ID: 016877529802

Routes (2)

Destination	Target	Status	Propagated
0.0.0.0/0	nat-09da26f8fbb34f9f2	Active	No
120.0.0.0/16	local	Active	No

- **Step 7**: Under the **Subnets associations** tab, click **Edit subnet associations** and select your private subnets.



rtb-0c1cddf0b0be64ea5 / PrivateRouteTable

Details

Route table ID: rtb-0c1cddf0b0be64ea5

Main: No

Explicit subnet associations: 2 subnets

Edge associations: –

VPC: vpc-0c06f536b657c06e4 | MyVPC

Owner ID: 016877529802

Explicit subnet associations (2)

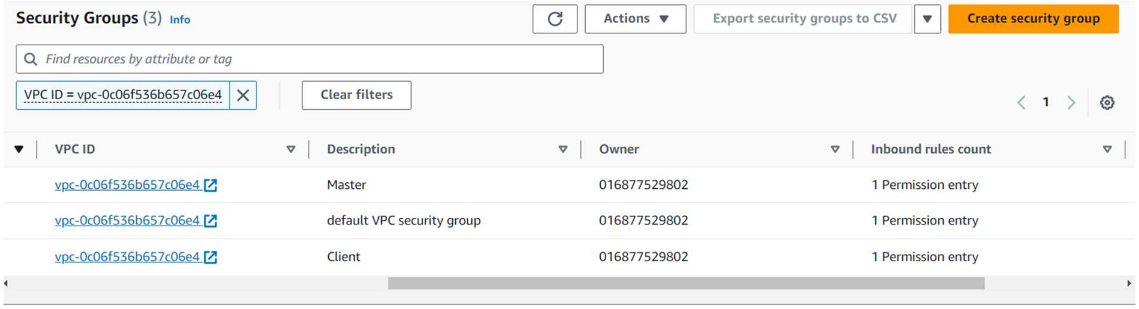
Name	Subnet ID	IPv4 CIDR	IPv6 CIDR
PrivateSubnet1	subnet-0f5b039cc7cf6a5f3	120.0.2.0/24	–
PrivateSubnet2	subnet-07d549b6a65607287	120.0.3.0/24	–

1. Launch EC2 Instances

- **Step 1**: Log in to the AWS Management Console and go to the **EC2 Dashboard**.
- **Step 2**: Click on **Launch Instance**.
- **Step 3**: Choose an Amazon Machine Image (AMI). For simplicity, use **Amazon Linux 2**.
- **Step 4**: Choose an Instance Type (e.g., t2.micro).
- **Step 5**: Configure Instance Details:
    - **Network**: Select the VPC where you want to create the instances (e.g., MYVPC1).
    - **Subnet**: Select a public subnet(for Master) and private subnet(for client) within the chosen VPC.
- **Step 6**: Add Storage if needed.
- **Step 7**: Add Tags:

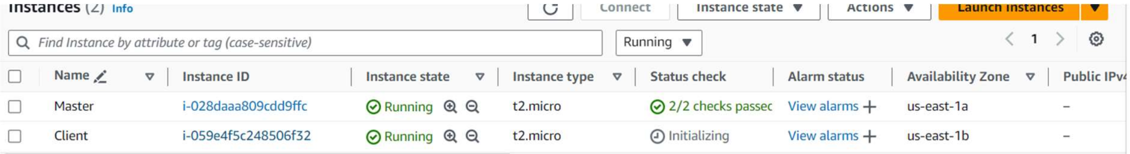
  - **Key**: Name
  - **Value**: Master (for the first instance) and Client (for the second instance)

- **Step 8**: Configure Security Group:
  - For the **Master** instance, create a security group (e.g., MasterSG) that allows:
    - **SSH** access from anywhere (0.0.0.0/0).
  - For the **Client** instance, create a security group (e.g., ClientSG) that:
    - **Does not** allow direct SSH access from anywhere.


Security Groups (3) Info
Actions ▼
Export security groups to CSV
Create security group
Find resources by attribute or tag
VPC ID = vpc-0c06f536b657c06e4
Clear filters
1

VPC ID	Description	Owner	Inbound rules count
vpc-0c06f536b657c06e4	Master	016877529802	1 Permission entry
vpc-0c06f536b657c06e4	default VPC security group	016877529802	1 Permission entry
vpc-0c06f536b657c06e4	Client	016877529802	1 Permission entry

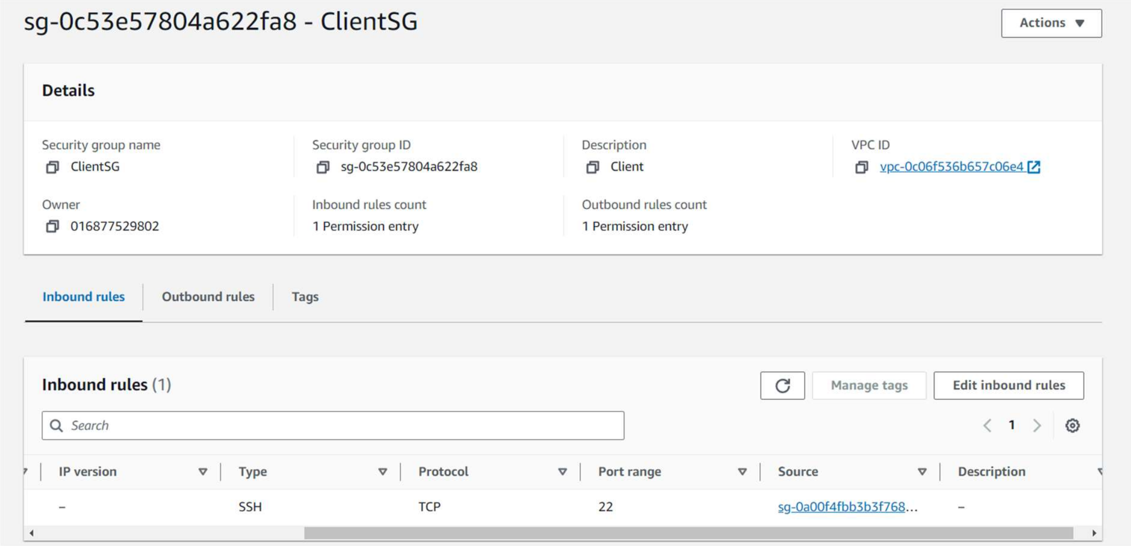
- **Step 9**: Review and Launch the instances.



□	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv
□	Master	i-028daaa809cdd9ffc	Running	t2.micro	2/2 checks passec	View alarms +	us-east-1a	–
□	Client	i-059e4f5c248506f32	Running	t2.micro	Initializing	View alarms +	us-east-1b	–

2. Configure Security Groups

- **Step 1**: Go to the **Security Groups** section in the EC2 Dashboard.
- **Step 2**: Select the ClientSG security group.
- **Step 3**: Edit the **Inbound Rules**:
  - Add a new rule to allow **SSH** access:
    - **Type**: SSH
    - **Protocol**: TCP
    - **Port Range**: 22
    - **Source**: Select the MasterSG security group (This restricts SSH access to the Client instance only from the Master instance).



sg-0c53e57804a622fa8 - ClientSG

Details

Security group name ClientSG	Security group ID sg-0c53e57804a622fa8	Description Client	VPC ID vpc-0c06f536b657c06e4
Owner 016877529802	Inbound rules count 1 Permission entry	Outbound rules count 1 Permission entry	

Inbound rules (1)

IP version	Type	Protocol	Port range	Source	Description
–	SSH	TCP	22	sg-0a00f4fbb3b3f768...	–

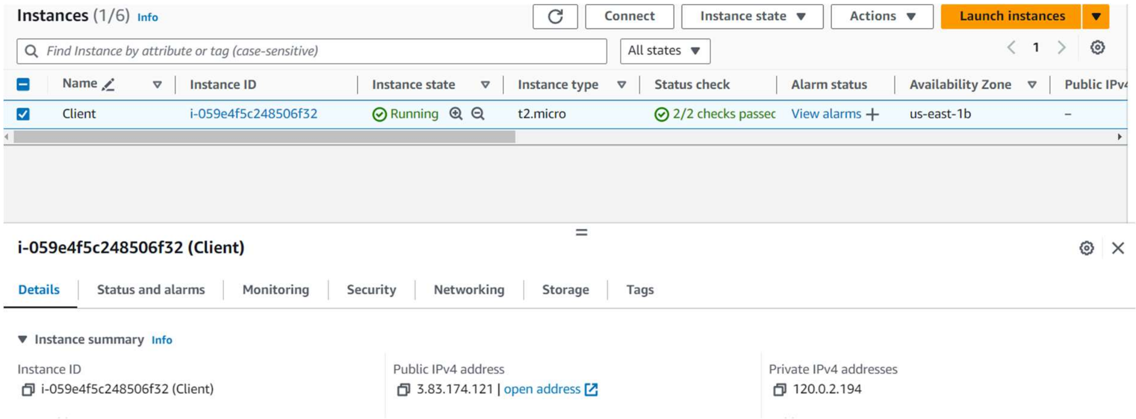
- **Step 4**: Save the changes.

3. Test the Configuration

- **Step 1**: SSH into the **Master** instance from your local machine using the command:

  ssh -i "new.pem" ec2-user@120.0.1.25

- **Step 2**: From the **Master** instance, SSH into the **Client** instance using the private IP address of the Client instance:



Instances (1/6) Info

☑	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4
☑	Client	i-059e4f5c248506f32	✓ Running	t2.micro	✓ 2/2 checks passec	View alarms +	us-east-1b	–

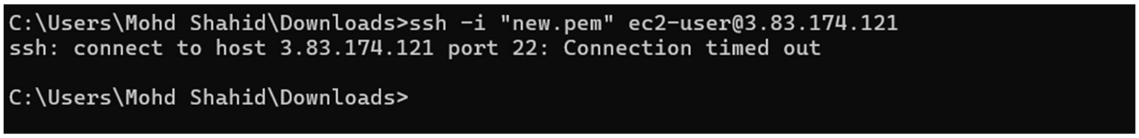
i-059e4f5c248506f32 (Client)

▼ Instance summary Info

Instance ID i-059e4f5c248506f32 (Client)	Public IPv4 address 3.83.174.121 | open address	Private IPv4 addresses 120.0.2.194

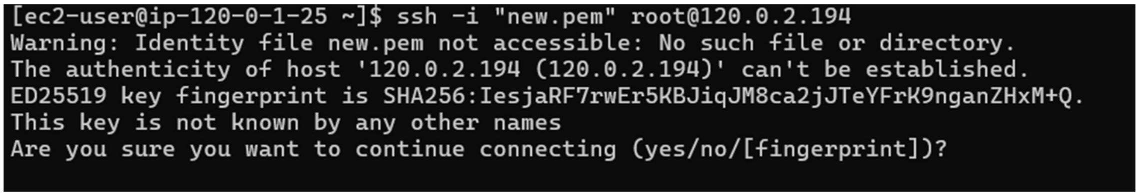
ssh -i "new.pem" ec2-user@3.83.174.121

- **Step 3**: Verify that the Client instance is not directly accessible via SSH from your local machine, only through the Master instance.



```
C:\Users\Mohd Shahid\Downloads>ssh -i "new.pem" ec2-user@3.83.174.121
ssh: connect to host 3.83.174.121 port 22: Connection timed out

C:\Users\Mohd Shahid\Downloads>
```



```
[ec2-user@ip-120-0-1-25 ~]$ ssh -i "new.pem" root@120.0.2.194
Warning: Identity file new.pem not accessible: No such file or directory.
The authenticity of host '120.0.2.194 (120.0.2.194)' can't be established.
ED25519 key fingerprint is SHA256:IesjaRF7rwEr5KBJiqJM8ca2jJTeYFrK9nganZHxM+Q.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])?
```