www.linkedin.com/in/arun-kumar-akula

VPC Setup and VPC Peering in AWS

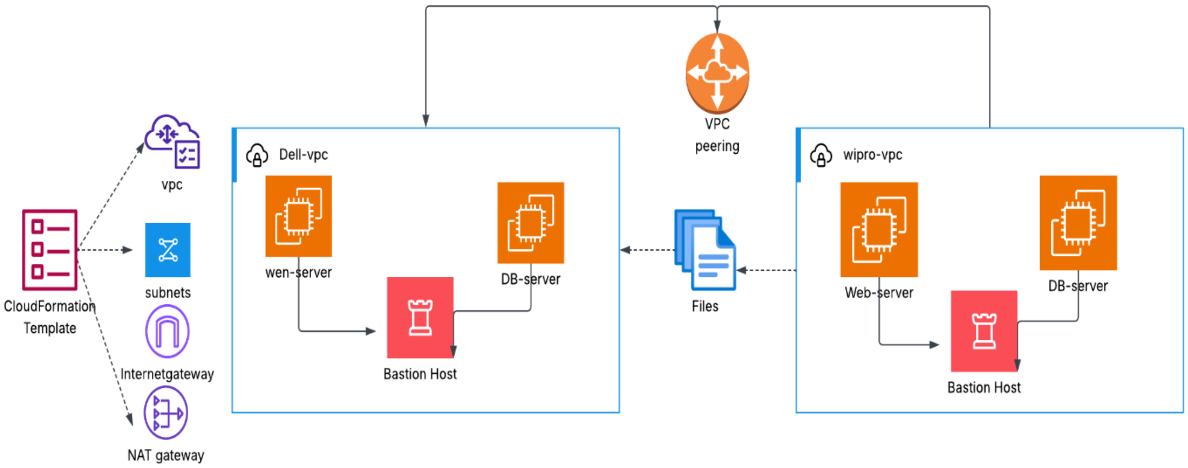
Overview:

Created a complete VPC network using **CloudFormation** including subnets, route tables, IGW, and NAT Gateway.

launched EC2 instances across two separate networks, connected to private instances via a **bastion host**, and securely shared PEM keys.

Established **VPC peering** to enable file sharing between the networks demonstrating secure communication and access across VPCs.

VPC Peering Architecture



Goal of This Implementation :

To securely connect two VPCs for private communication and file sharing between EC2 instances across networks using a CloudFormation Templates and Vpc setup.

What is VPC Peering : VPC Peering is a networking connection between two Virtual Private Clouds (VPCs) in AWS that allows them to communicate privately using their private IP addresses as if they were on the same network or different aws accounts or regions.

- Traffic stays within AWS, not over the internet.
- Commonly used for microservices, shared services, or multi-account setups.

- Works across regions and accounts.

Step-by-step Setup Guide:

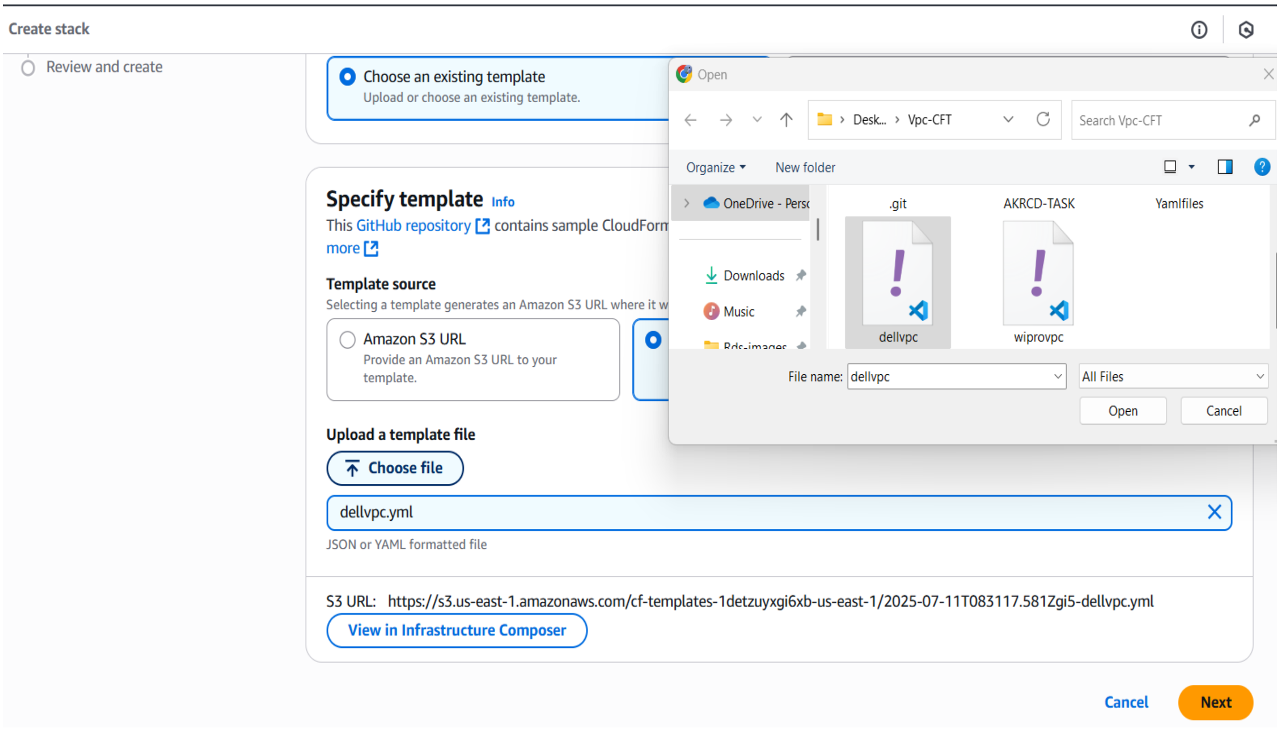
Step 1: Create VPC's (using CloudFormation)

The Vpc's was created using an AWS CloudFormation script to automate the provisioning of network components such as subnets, route tables, Internet Gateway, and NAT Gateway.

- You can find the CloudFormation templates used for this setup in the GitHub repository below:
- **GitHub Repository:** https://github.com/Arunkumarakula/Yamlfiles.git
- Use the provided script to create one VPC (e.g., Dell). To create a second VPC (e.g., Wipro), duplicate the template and update the resource names, tags, and CIDR blocks accordingly.

Step 2: Create a CloudFormation stack to create the required AWS resources like VPC, subnets, internet gateway, and route tables.

- Navigate to **CloudFormation** in the AWS Services search bar and open it
- Click **Create Stack**
  - Choose With new resources (standard).
- Choose a **Template** (Existing template)
- Specify template : **upload a template file & select file**



Create stack

○ Review and create

● Choose an existing template
Upload or choose an existing template.

Specify template Info
This GitHub repository contains sample CloudForm
more

Template source
Selecting a template generates an Amazon S3 URL where it w

○ Amazon S3 URL
Provide an Amazon S3 URL to your template.

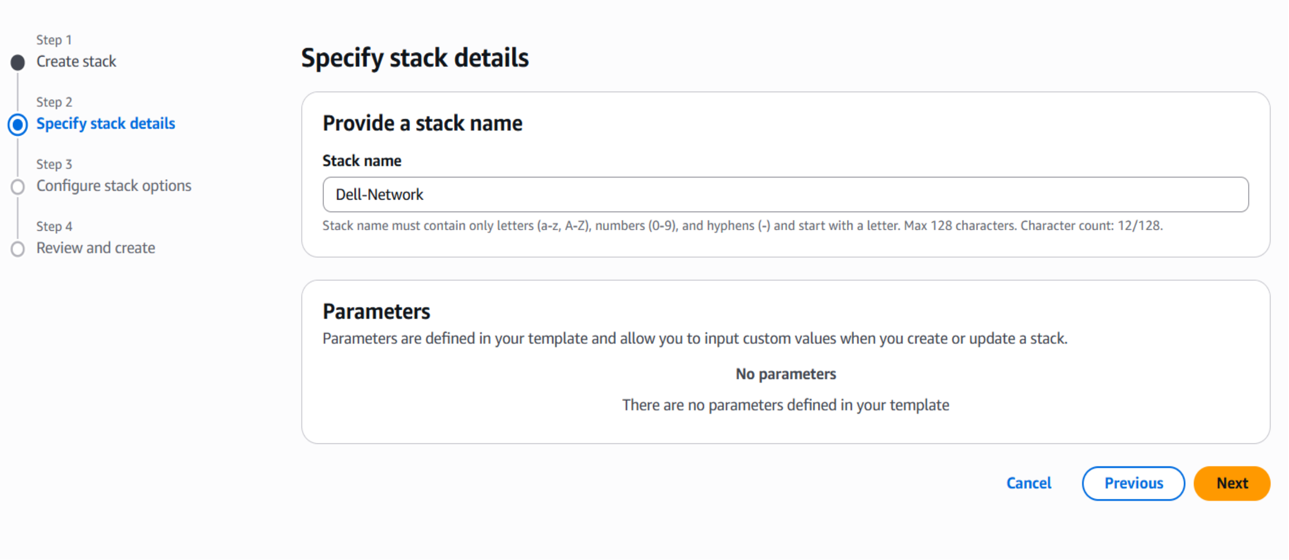
Upload a template file
Choose file
dellvpc.yml
JSON or YAML formatted file

S3 URL: https://s3.us-east-1.amazonaws.com/cf-templates-1detzuyxgi6xb-us-east-1/2025-07-11T083117.581Zgi5-dellvpc.yml

File name: dellvpc

All Files

- Click Next
- Specify Stack Details
  - **Stack name**: Give your stack name (e.g., Dell-Network-stack)



Step 1
Create stack
Step 2
Specify stack details
Step 3
Configure stack options
Step 4
Review and create

Specify stack details

Provide a stack name

Stack name

Dell-Network

Stack name must contain only letters (a-z, A-Z), numbers (0-9), and hyphens (-) and start with a letter. Max 128 characters. Character count: 12/128.

Parameters

Parameters are defined in your template and allow you to input custom values when you create or update a stack.

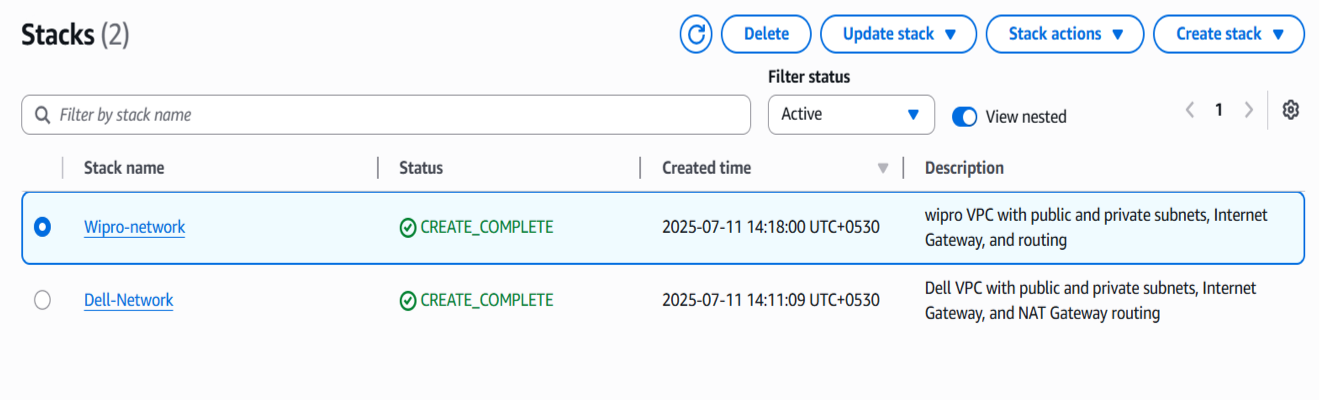
No parameters

There are no parameters defined in your template

- Click Next
- Configure stack options (optional) – click Next.
- **Review & Submit.**

Like above process create another network (Wipro Network-stack).

- Here we can observe that our stack has been created successfully.

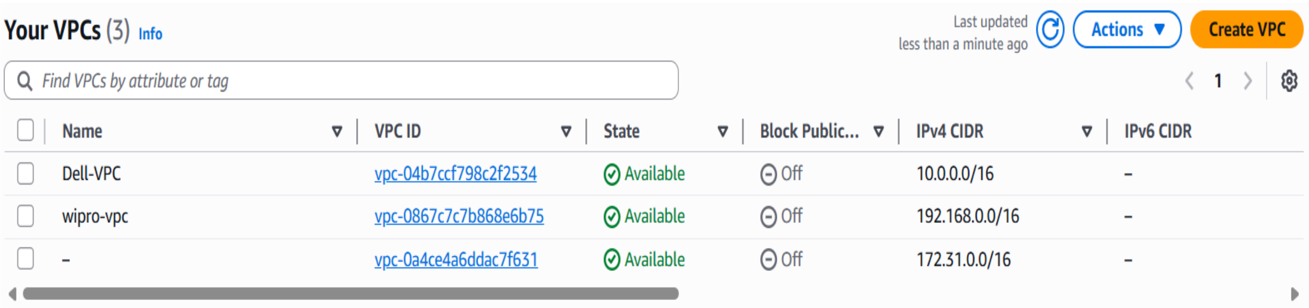


Stacks (2)

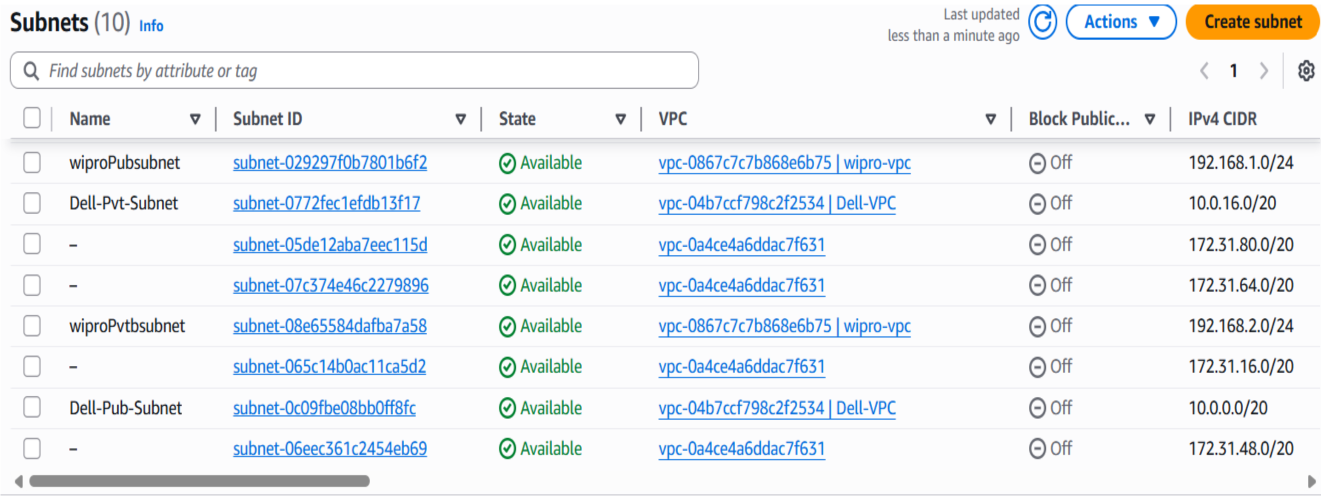
Filter status: Active

	Stack name	Status	Created time	Description
●	Wipro-network	✓ CREATE_COMPLETE	2025-07-11 14:18:00 UTC+0530	wipro VPC with public and private subnets, Internet Gateway, and routing
○	Dell-Network	✓ CREATE_COMPLETE	2025-07-11 14:11:09 UTC+0530	Dell VPC with public and private subnets, Internet Gateway, and NAT Gateway routing

➢ Now we can observe that our VPC environment has been created including the VPC, subnets, route tables, Internet gateway and NAT Gateway.


Your VPCs (3) Info
Last updated less than a minute ago
Actions ▼
Create VPC
Find VPCs by attribute or tag
1

□	Name	VPC ID	State	Block Public...	IPv4 CIDR	IPv6 CIDR
□	Dell-VPC	vpc-04b7ccf798c2f2534	✓ Available	⊖ Off	10.0.0.0/16	–
□	wipro-vpc	vpc-0867c7c7b868e6b75	✓ Available	⊖ Off	192.168.0.0/16	–
□	–	vpc-0a4ce4a6ddac7f631	✓ Available	⊖ Off	172.31.0.0/16	–


Subnets (10) Info
Last updated less than a minute ago
Actions ▼
Create subnet
Find subnets by attribute or tag
1

□	Name	Subnet ID	State	VPC	Block Public...	IPv4 CIDR
□	wiproPubsubnet	subnet-029297f0b7801b6f2	✓ Available	vpc-0867c7c7b868e6b75 | wipro-vpc	⊖ Off	192.168.1.0/24
□	Dell-Pvt-Subnet	subnet-0772fec1efdb13f17	✓ Available	vpc-04b7ccf798c2f2534 | Dell-VPC	⊖ Off	10.0.16.0/20
□	–	subnet-05de12aba7eec115d	✓ Available	vpc-0a4ce4a6ddac7f631	⊖ Off	172.31.80.0/20
□	–	subnet-07c374e46c2279896	✓ Available	vpc-0a4ce4a6ddac7f631	⊖ Off	172.31.64.0/20
□	wiproPvtbsubnet	subnet-08e65584dafba7a58	✓ Available	vpc-0867c7c7b868e6b75 | wipro-vpc	⊖ Off	192.168.2.0/24
□	–	subnet-065c14b0ac11ca5d2	✓ Available	vpc-0a4ce4a6ddac7f631	⊖ Off	172.31.16.0/20
□	Dell-Pub-Subnet	subnet-0c09fbe08bb0ff8fc	✓ Available	vpc-04b7ccf798c2f2534 | Dell-VPC	⊖ Off	10.0.0.0/20
□	–	subnet-06eec361c2454eb69	✓ Available	vpc-0a4ce4a6ddac7f631	⊖ Off	172.31.48.0/20

Step 3 : Create 4 EC2 Instances in 2 VPCs

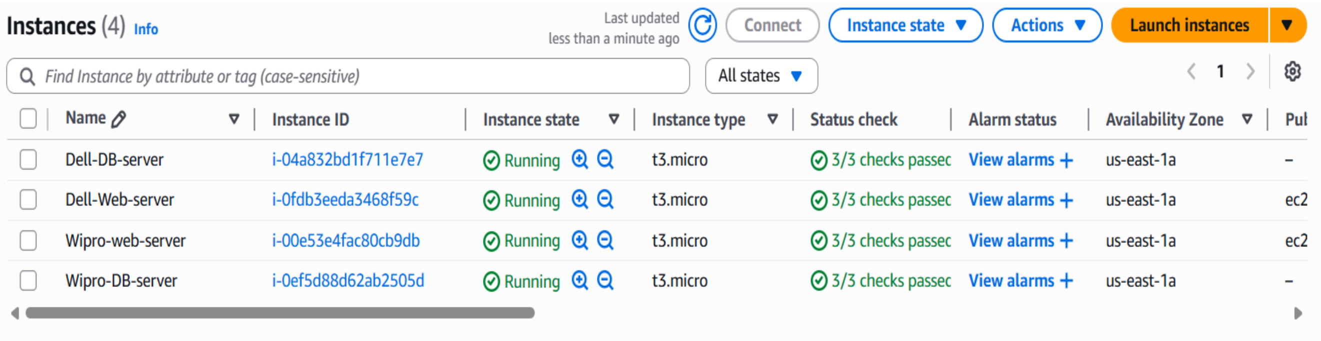
- To test secure communication between two VPCs.
  By launching EC2 instances in each VPC, we can verify that resources (like web and DB servers) can talk to each other using private IPs over the peering connection.

➢ In the Dell VPC Network create 2 instnaces.

- Server-1 name as **Dell-Web Server** and create in **Public Subnet**
- Server-2 name as **Dell-DB Server** and create in **Private Subnet**

➢ In the Wipro VPC Network create 2 instances.

- Server-1 name as **Wipro-Web Server** and create in **Public Subnet**
- Server-2 name as **wipro-DB Server** and create in **Private Subnet**



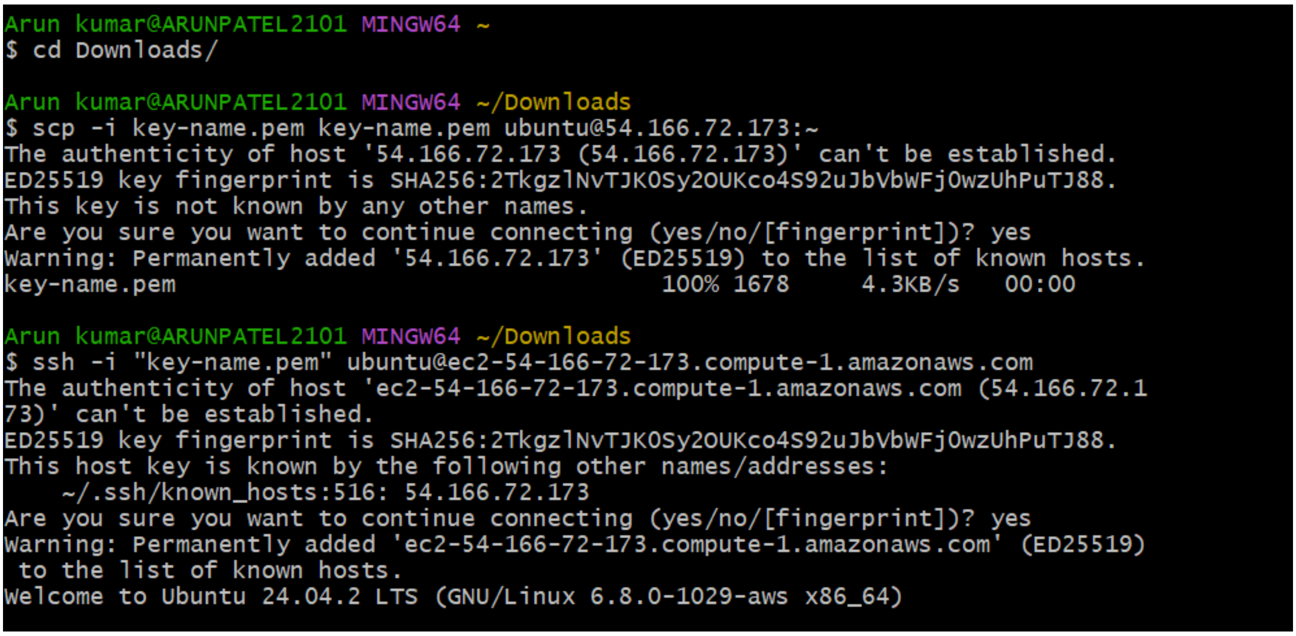
Instances (4) Info
Last updated less than a minute ago
Connect
Instance state ▼
Actions ▼
Launch instances
Find Instance by attribute or tag (case-sensitive)
All states ▼
1

□	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Pul
□	Dell-DB-server	i-04a832bd1f711e7e7	✓ Running	t3.micro	✓ 3/3 checks passec	View alarms +	us-east-1a	–
□	Dell-Web-server	i-0fdb3eeda3468f59c	✓ Running	t3.micro	✓ 3/3 checks passec	View alarms +	us-east-1a	ec2
□	Wipro-web-server	i-00e53e4fac80cb9db	✓ Running	t3.micro	✓ 3/3 checks passec	View alarms +	us-east-1a	ec2
□	Wipro-DB-server	i-0ef5d88d62ab2505d	✓ Running	t3.micro	✓ 3/3 checks passec	View alarms +	us-east-1a	–

Step 4: Copy the PEM file from your local machine to the Dell web server and connect to your **Dell web server EC2 instance.**

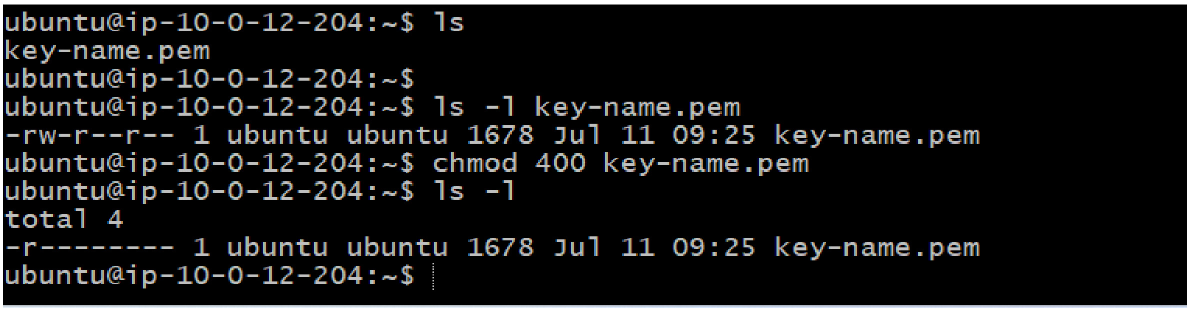
- Command to copy pem file from local machine to server :

```
scp -i key-name.pem key-name.pem ubuntu@public-ip:~
```



```
Arun kumar@ARUNPATEL2101 MINGW64 ~
$ cd Downloads/

Arun kumar@ARUNPATEL2101 MINGW64 ~/Downloads
$ scp -i key-name.pem key-name.pem ubuntu@54.166.72.173:~
The authenticity of host '54.166.72.173 (54.166.72.173)' can't be established.
ED25519 key fingerprint is SHA256:2TkgzlNvTJKOSy2OUKco4S92uJbVbWFjOwzUhPuTJ88.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '54.166.72.173' (ED25519) to the list of known hosts.
key-name.pem                                  100% 1678     4.3KB/s   00:00

Arun kumar@ARUNPATEL2101 MINGW64 ~/Downloads
$ ssh -i "key-name.pem" ubuntu@ec2-54-166-72-173.compute-1.amazonaws.com
The authenticity of host 'ec2-54-166-72-173.compute-1.amazonaws.com (54.166.72.1
73)' can't be established.
ED25519 key fingerprint is SHA256:2TkgzlNvTJKOSy2OUKco4S92uJbVbWFjOwzUhPuTJ88.
This host key is known by the following other names/addresses:
    ~/.ssh/known_hosts:516: 54.166.72.173
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'ec2-54-166-72-173.compute-1.amazonaws.com' (ED25519)
 to the list of known hosts.
Welcome to Ubuntu 24.04.2 LTS (GNU/Linux 6.8.0-1029-aws x86_64)
```

- Now check the pem file in web-server and change the permissions of a file readable by others. Setting it to chmod 400 ensures only you (the owner) can read it which is required for secure SSH access.

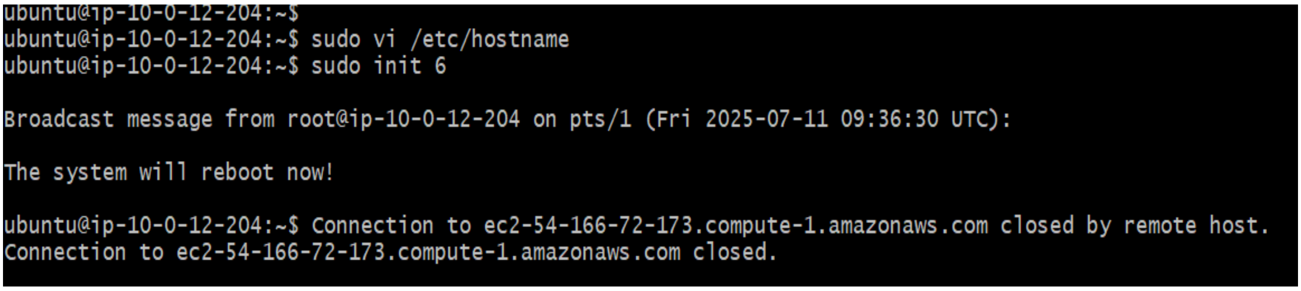


```
ubuntu@ip-10-0-12-204:~$ ls
key-name.pem
ubuntu@ip-10-0-12-204:~$
ubuntu@ip-10-0-12-204:~$ ls -l key-name.pem
-rw-r--r-- 1 ubuntu ubuntu 1678 Jul 11 09:25 key-name.pem
ubuntu@ip-10-0-12-204:~$ chmod 400 key-name.pem
ubuntu@ip-10-0-12-204:~$ ls -l
total 4
-r-------- 1 ubuntu ubuntu 1678 Jul 11 09:25 key-name.pem
ubuntu@ip-10-0-12-204:~$
```

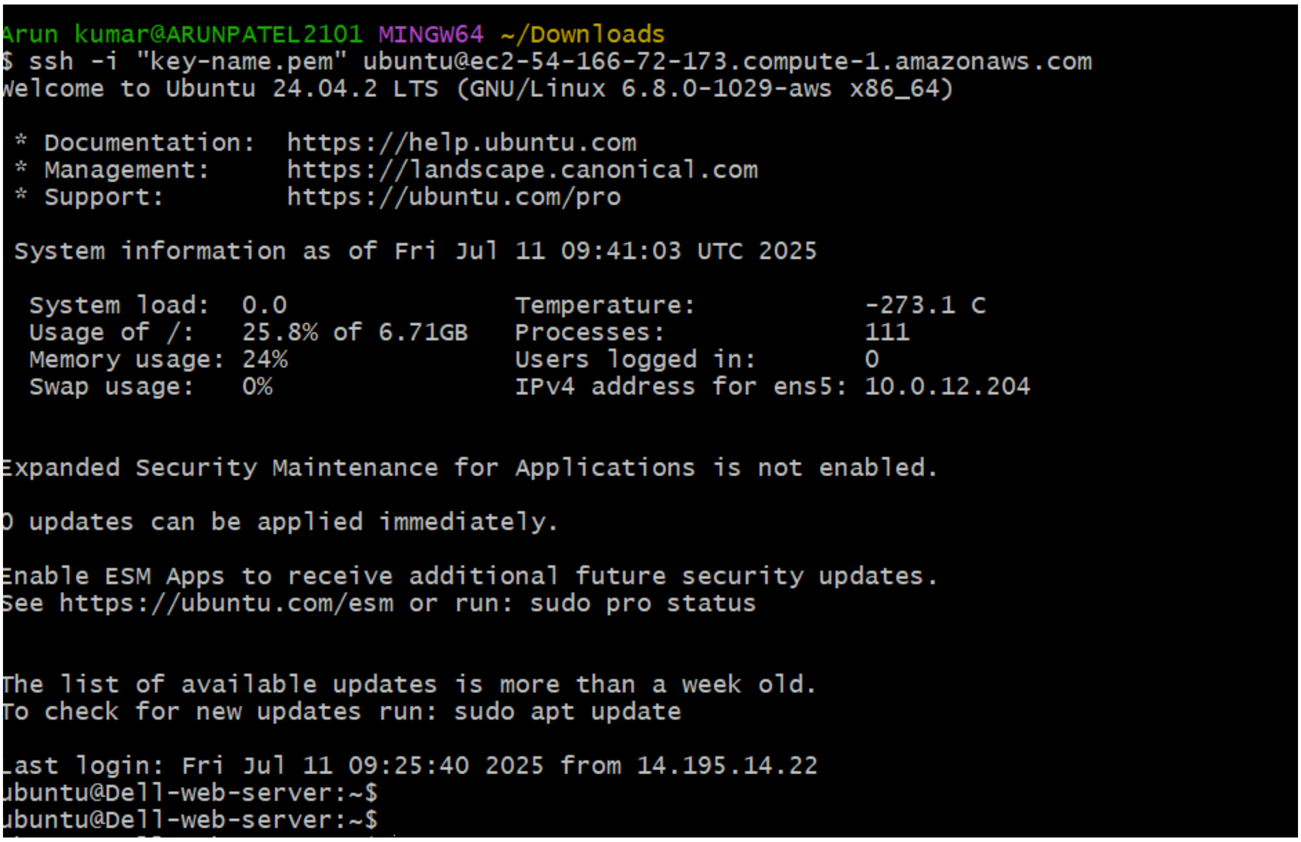
- Change the host-name to avoid the confusion of servers to identify in terminals.

# In the all servers which you have try to change the host-name(To identify easily).

- Sudo vi /etc/hostname (here change Ip-add to Dell-web-server)
- Sudo init 6 (reboot the server)
- Now login again with ssh (you will see the hostname)

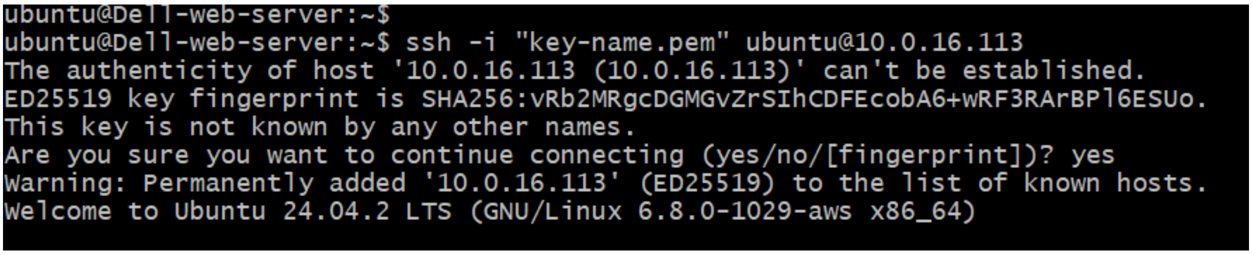


```
ubuntu@ip-10-0-12-204:~$
ubuntu@ip-10-0-12-204:~$ sudo vi /etc/hostname
ubuntu@ip-10-0-12-204:~$ sudo init 6

Broadcast message from root@ip-10-0-12-204 on pts/1 (Fri 2025-07-11 09:36:30 UTC):

The system will reboot now!

ubuntu@ip-10-0-12-204:~$ Connection to ec2-54-166-72-173.compute-1.amazonaws.com closed by remote host.
Connection to ec2-54-166-72-173.compute-1.amazonaws.com closed.
```



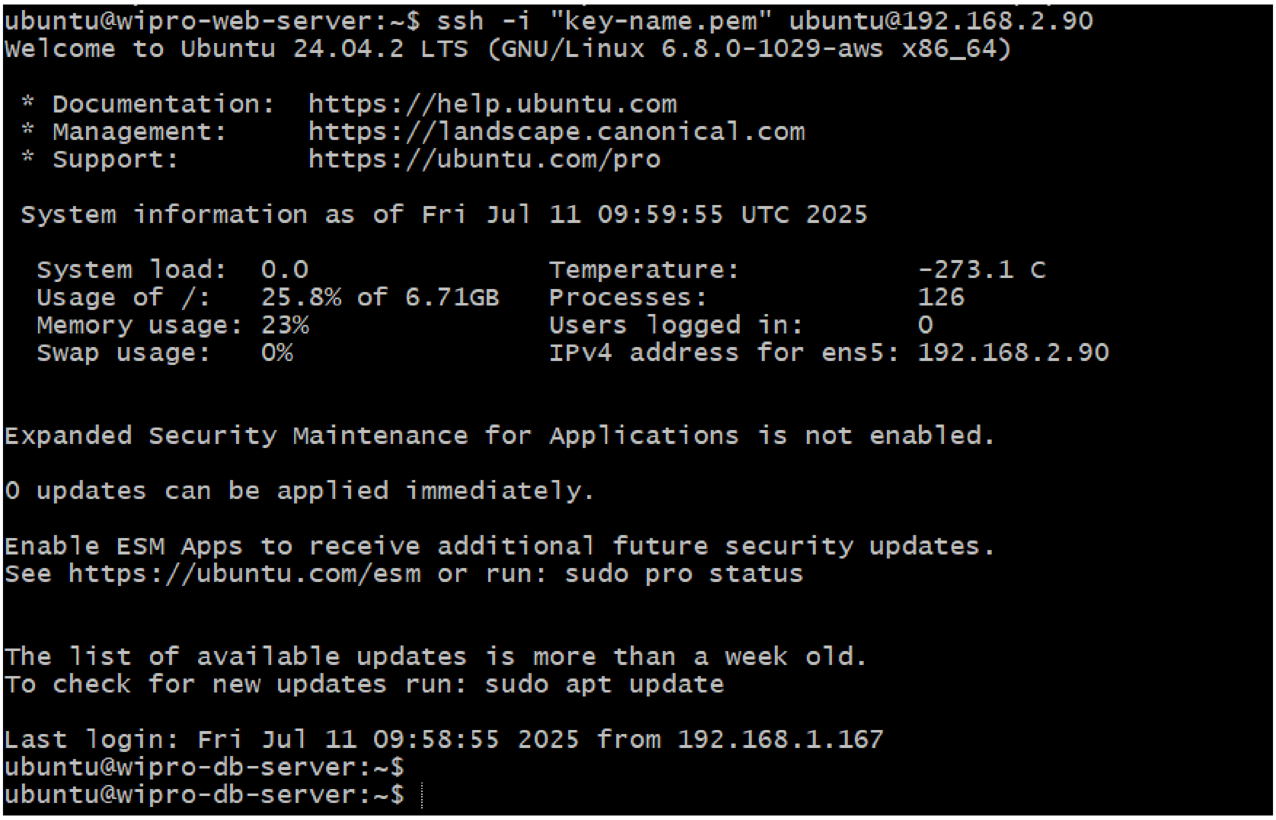
```
Arun kumar@ARUNPATEL2101 MINGW64 ~/Downloads
$ ssh -i "key-name.pem" ubuntu@ec2-54-166-72-173.compute-1.amazonaws.com
Welcome to Ubuntu 24.04.2 LTS (GNU/Linux 6.8.0-1029-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro

 System information as of Fri Jul 11 09:41:03 UTC 2025

  System load:  0.0               Temperature:           -273.1 C
  Usage of /:   25.8% of 6.71GB   Processes:             111
  Memory usage: 24%               Users logged in:       0
  Swap usage:   0%                IPv4 address for ens5: 10.0.12.204


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: Fri Jul 11 09:25:40 2025 from 14.195.14.22
ubuntu@Dell-web-server:~$
ubuntu@Dell-web-server:~$
```

- Now connect the Dell-DB-server from the Dell-web-server & change the host-name.



```
ubuntu@Dell-web-server:~$
ubuntu@Dell-web-server:~$ ssh -i "key-name.pem" ubuntu@10.0.16.113
The authenticity of host '10.0.16.113 (10.0.16.113)' can't be established.
ED25519 key fingerprint is SHA256:vRb2MRgcDGMGvZrSIhCDFEcobA6+wRF3RArBPl6ESUo.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.0.16.113' (ED25519) to the list of known hosts.
Welcome to Ubuntu 24.04.2 LTS (GNU/Linux 6.8.0-1029-aws x86_64)
```

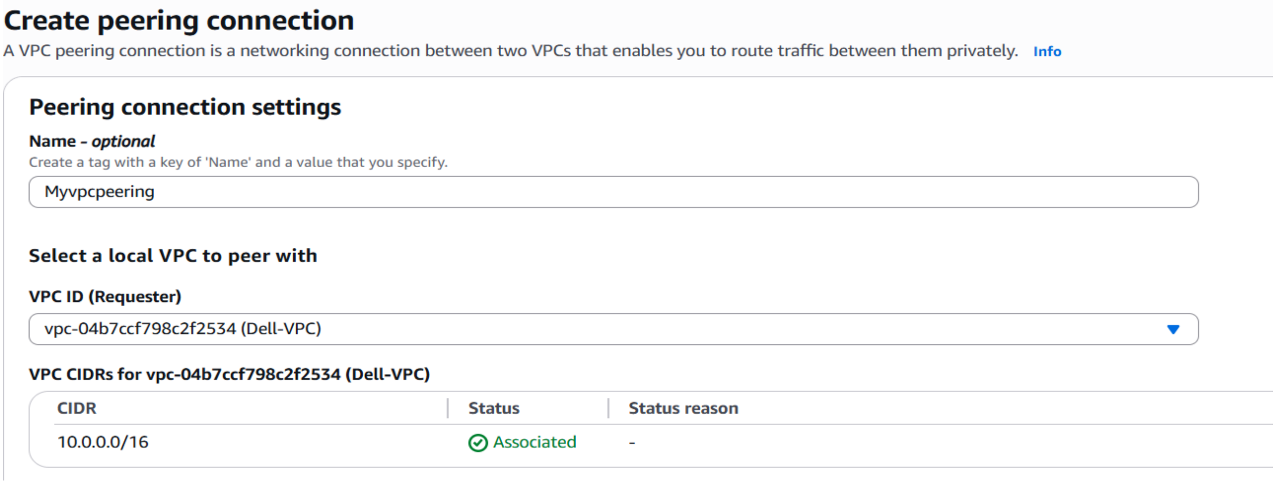
- Same like above Copy the PEM file from your local machine to the wipro web server and connect to your wipro web server EC2 instance , Follow **step-4** process.
- Here can observe successfully copied pemfile and changed the host-name, connected from web server to db server.



```
ubuntu@wipro-web-server:~$ ssh -i "key-name.pem" ubuntu@192.168.2.90
Welcome to Ubuntu 24.04.2 LTS (GNU/Linux 6.8.0-1029-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro

 System information as of Fri Jul 11 09:59:55 UTC 2025

  System load:  0.0               Temperature:           -273.1 C
  Usage of /:   25.8% of 6.71GB   Processes:             126
  Memory usage: 23%               Users logged in:       0
  Swap usage:   0%                IPv4 address for ens5: 192.168.2.90


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: Fri Jul 11 09:58:55 2025 from 192.168.1.167
ubuntu@wipro-db-server:~$
ubuntu@wipro-db-server:~$
```

Step 5 : Create a VPC Peering Connection

- Go to **VPC Console** → **Peering Connections** → Click **Create Peering Connection.**



Create peering connection

A VPC peering connection is a networking connection between two VPCs that enables you to route traffic between them privately. Info

Peering connection settings

Name - *optional*

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

Myvpcpeering

Select a local VPC to peer with

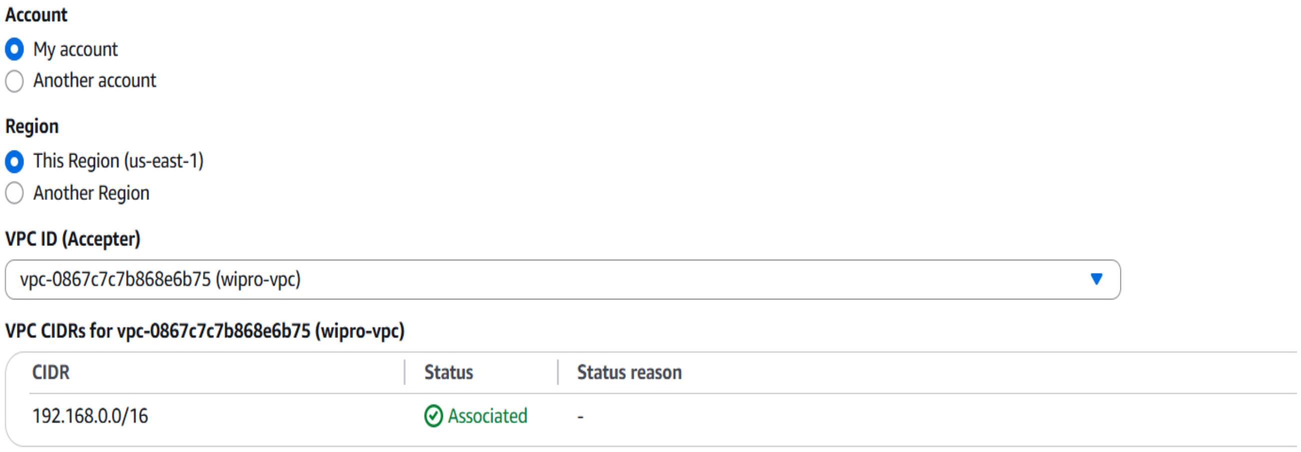
VPC ID (Requester)

vpc-04b7ccf798c2f2534 (Dell-VPC) ▼

VPC CIDRs for vpc-04b7ccf798c2f2534 (Dell-VPC)

CIDR	Status	Status reason
10.0.0.0/16	✓ Associated	-

- Provide:
  - **Name** tag (e.g., MyVpcpeering)
  - **Requester VPC**: Select Dell vpc



Account

● My account
○ Another account

Region

● This Region (us-east-1)
○ Another Region

VPC ID (Accepter)

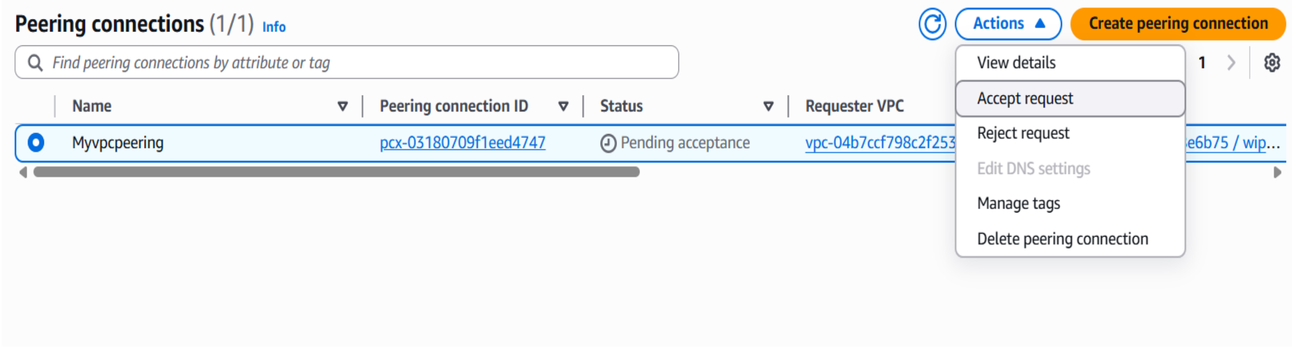
vpc-0867c7c7b868e6b75 (wipro-vpc)

VPC CIDRs for vpc-0867c7c7b868e6b75 (wipro-vpc)

CIDR	Status	Status reason
192.168.0.0/16	✓ Associated	-

  - **Accepter VPC**: Select wipro vpc

- If another account → provide Account ID and VPC ID of the other VPC
- Click **Create Peering Connection.**
- After creating , choose peering and accept the request.

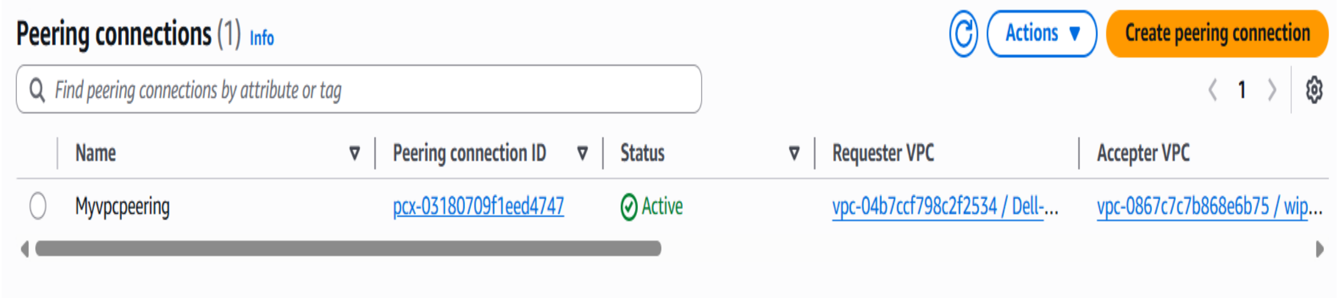


Peering connections (1/1) Info

Actions ▲
Create peering connection
Find peering connections by attribute or tag
View details
Accept request
Reject request
Edit DNS settings
Manage tags
Delete peering connection

	Name	Peering connection ID	Status	Requester VPC	
●	Myvpcpeering	pcx-03180709f1eed4747	Pending acceptance	vpc-04b7ccf798c2f253	e6b75 / wip...

- Now the vpc peering status will be active .



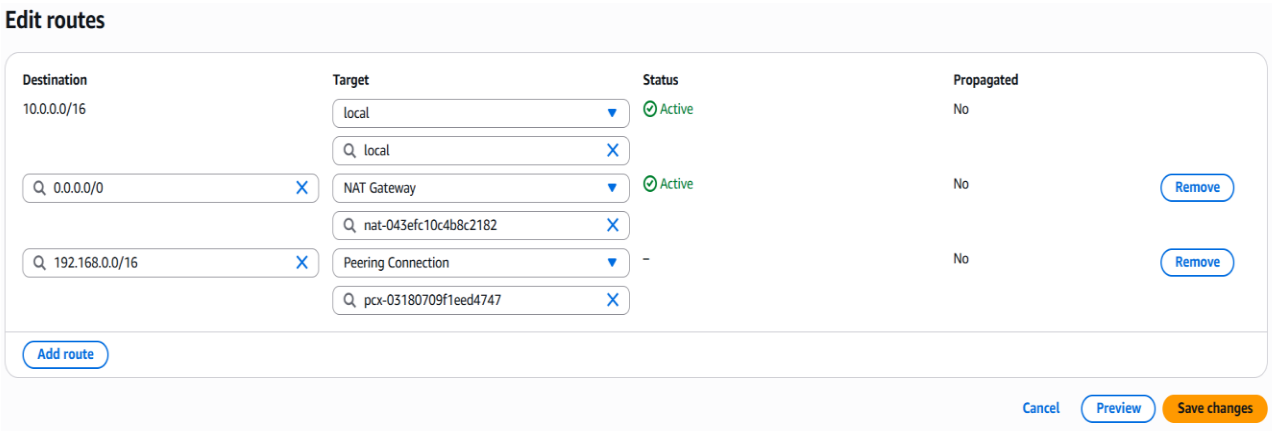
Peering connections (1) Info

Actions ▼
Create peering connection
Find peering connections by attribute or tag
1

	Name	Peering connection ID	Status	Requester VPC	Accepter VPC
○	Myvpcpeering	pcx-03180709f1eed4747	✓ Active	vpc-04b7ccf798c2f2534 / Dell-...	vpc-0867c7c7b868e6b75 / wip...

Step 6 : Now Edit the Route tables for DB-servers

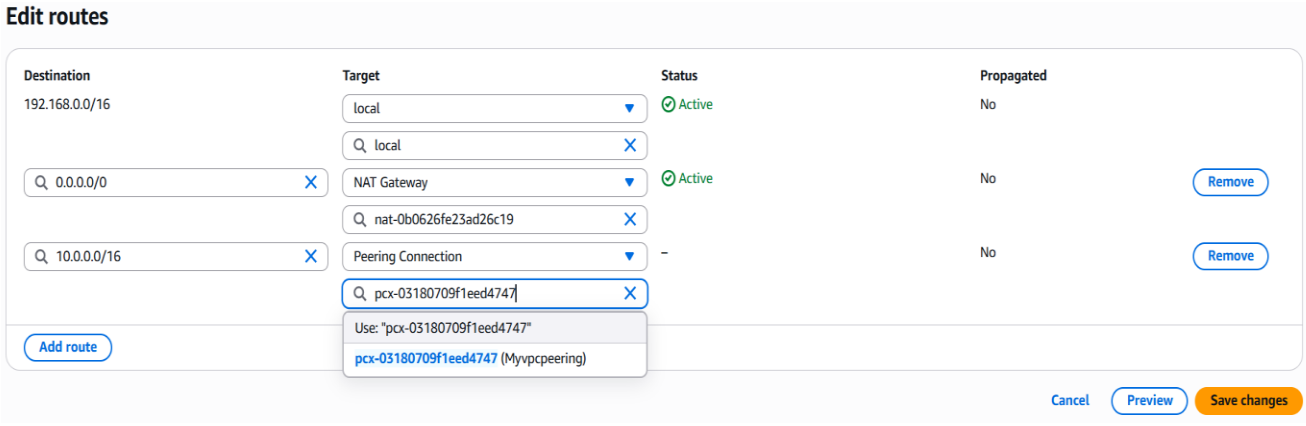
- To allow private EC2 instances (DB servers) to communicate with other networks (like web servers or peered VPCs) using private IPs while not giving them internet access.
- Dell private route table (give credentials of wipro vpc Cidr)



Edit routes

Destination	Target	Status	Propagated	
10.0.0.0/16	local local	Active	No	
0.0.0.0/0	NAT Gateway nat-043efc10c4b8c2182	Active	No	Remove
192.168.0.0/16	Peering Connection pcx-03180709f1eed4747	-	No	Remove

- Wipro private route table (give credentials of dell vpc cidr).



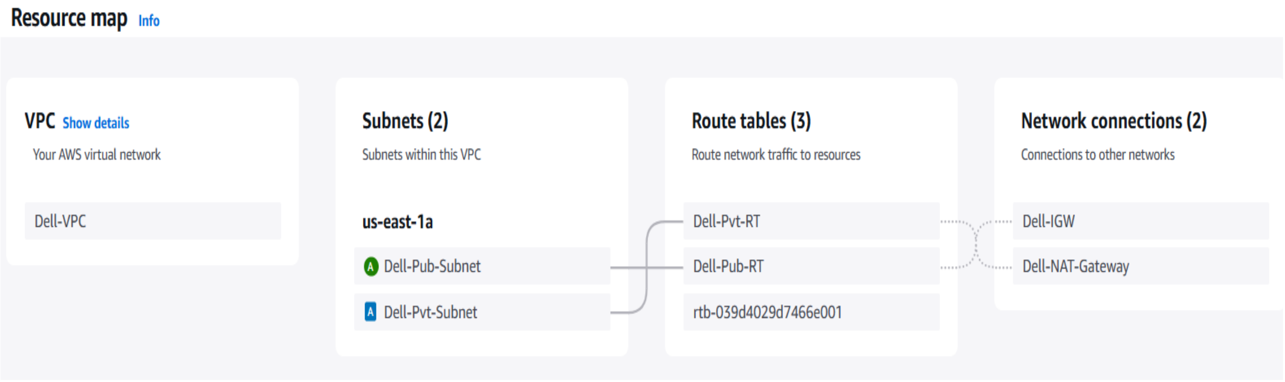
Edit routes

Destination	Target	Status	Propagated	
192.168.0.0/16	local local	Active	No	
0.0.0.0/0	NAT Gateway nat-0b0626fe23ad26c19	Active	No	Remove
10.0.0.0/16	Peering Connection pcx-03180709f1eed4747	-	No	Remove

Use: "pcx-03180709f1eed4747"

pcx-03180709f1eed4747 (Myvpcpeering)

- Now we can observe in the Vpc resource map



Resource map Info

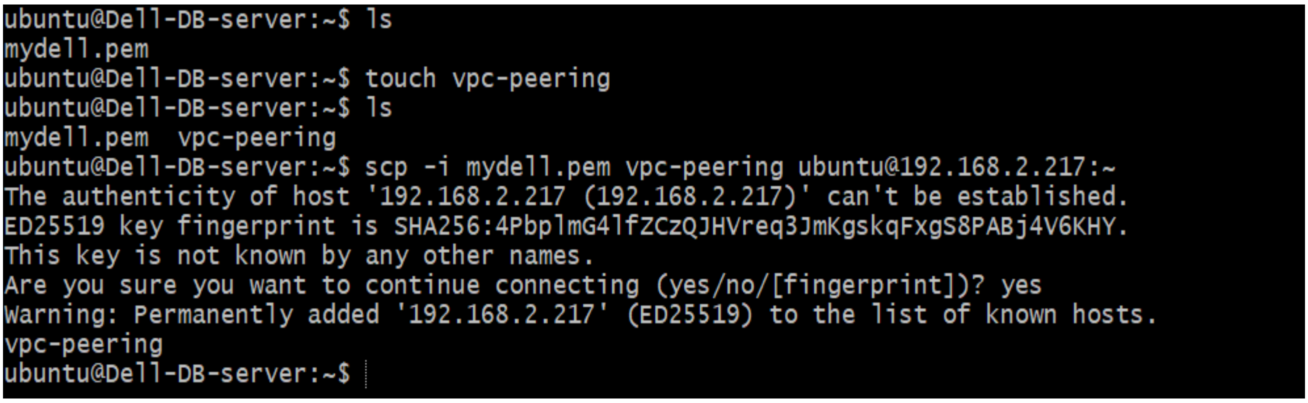
VPC Show details
Your AWS virtual network
Dell-VPC

Subnets (2)
Subnets within this VPC
us-east-1a
Dell-Pub-Subnet
Dell-Pvt-Subnet

Route tables (3)
Route network traffic to resources
Dell-Pvt-RT
Dell-Pub-RT
rtb-039d4029d7466e001

Network connections (2)
Connections to other networks
Dell-IGW
Dell-NAT-Gateway

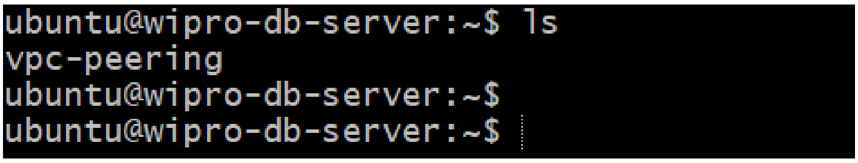
- Now create a file in the Dell-DB-server and share to Wipro-DB-server.



```
ubuntu@Dell-DB-server:~$ ls
mydell.pem
ubuntu@Dell-DB-server:~$ touch vpc-peering
ubuntu@Dell-DB-server:~$ ls
mydell.pem  vpc-peering
ubuntu@Dell-DB-server:~$ scp -i mydell.pem vpc-peering ubuntu@192.168.2.217:~
The authenticity of host '192.168.2.217 (192.168.2.217)' can't be established.
ED25519 key fingerprint is SHA256:4PbplmG4lfZCzQJHVreq3JmKgskqFxgS8PABj4V6KHY.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.2.217' (ED25519) to the list of known hosts.
vpc-peering
ubuntu@Dell-DB-server:~$
```

```
# File to share from Network-Network
scp -I key-name.pem key-name.pem ubuntu@private-ip:~
```

- Now we can see that which we created in the Dell-DB-server file will be in the wipro-DB-srever.



```
ubuntu@wipro-db-server:~$ ls
vpc-peering
ubuntu@wipro-db-server:~$
ubuntu@wipro-db-server:~$
```

---****---