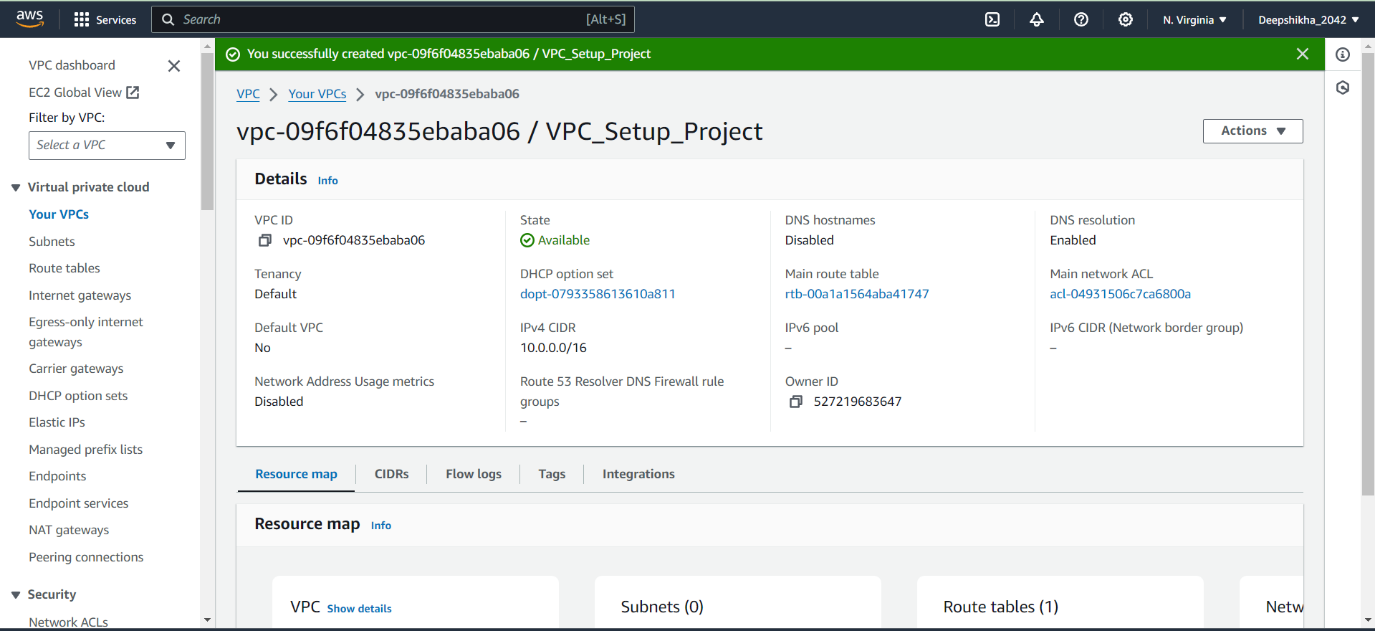
**AWS Fortified Cloud Architecture  
( BY : Deepshikha Paty )**

* **STEP 1:**

Created a VPC network of CIDR range 10.0.0.0/16



* **STEP-2:**

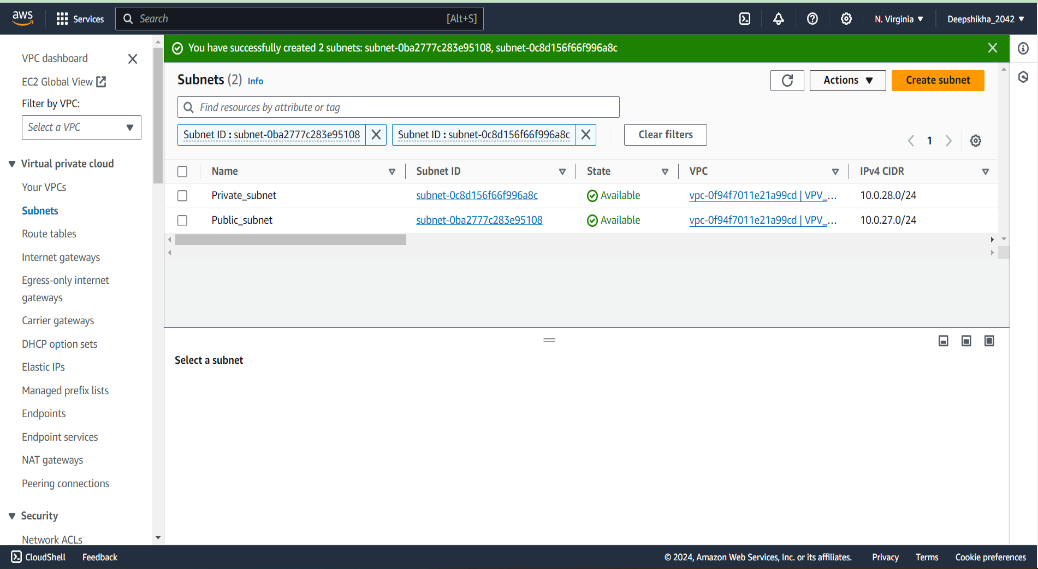
Inside this VPC, created two subnets.

1. Subnet-1 (Public)

* Name- Public\_subnet
* CIDR: 10.0.27.0/24
* Web-server
* Availability-Zone: us-east-1a

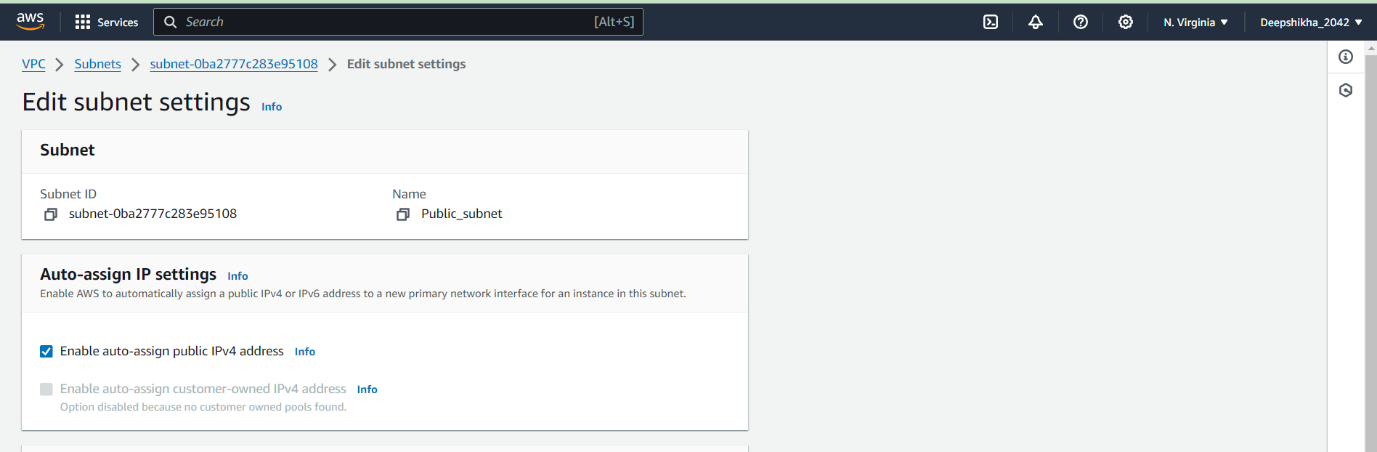
1. Subnet-2 (Private)

* Name- Private\_subnet
* CIDR:10.0.28.0/24
* Database-server
* Availability-Zone:us-east-1b



* **STEP-3:**

Made the subnet-1 as public by enabling public access.

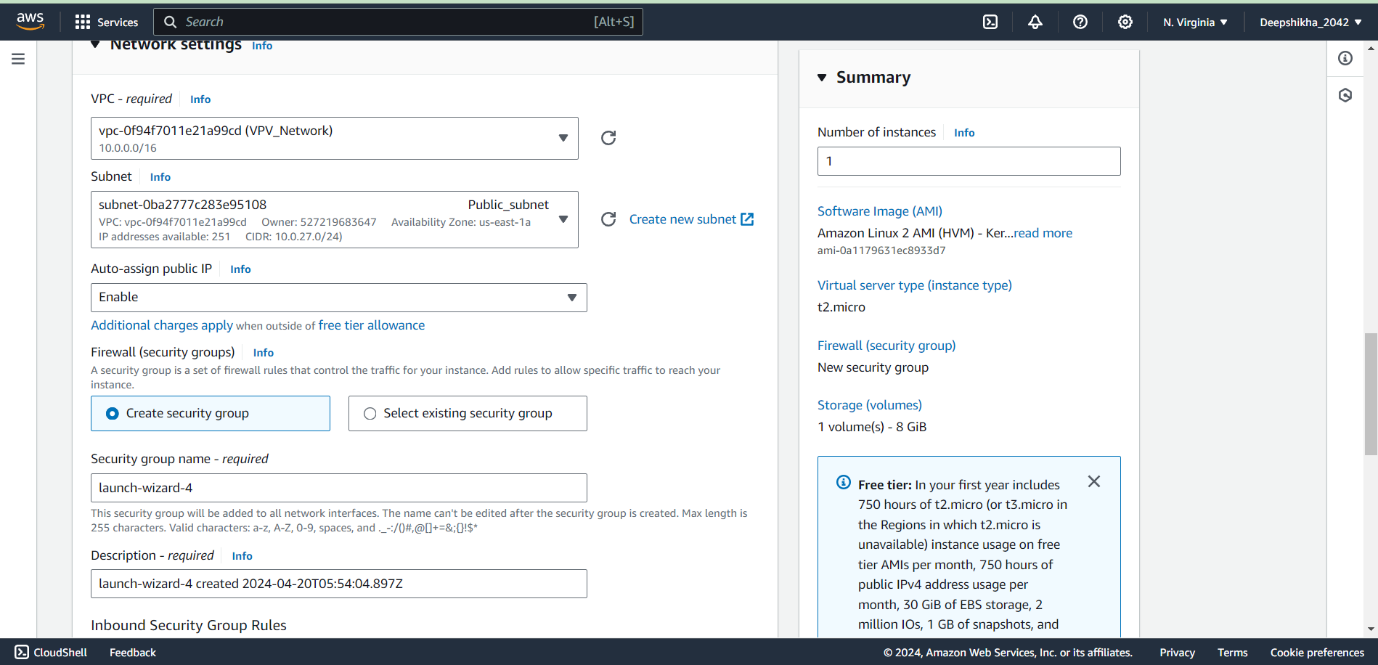


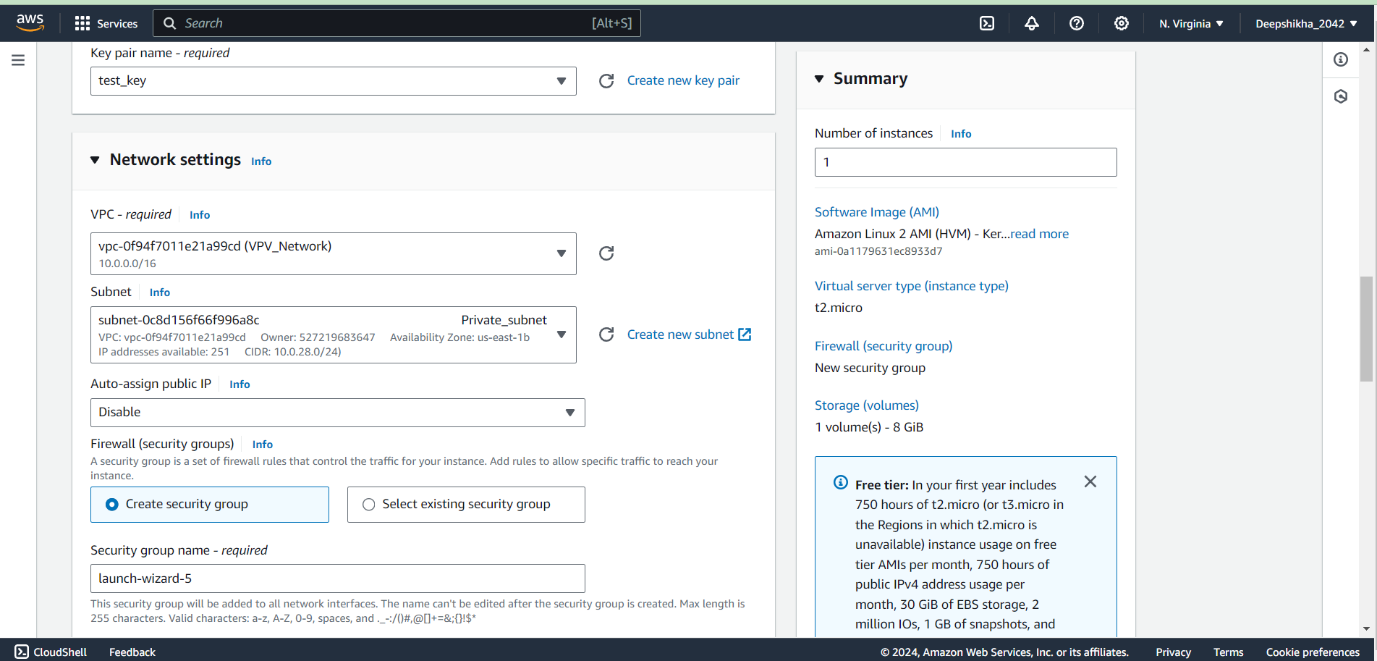
* **STEP-4:**

Then, I deployed a web-server (EC2-instance) and a database-server (EC2-instance) in the public and private subnet respectively.

**NOTE:**

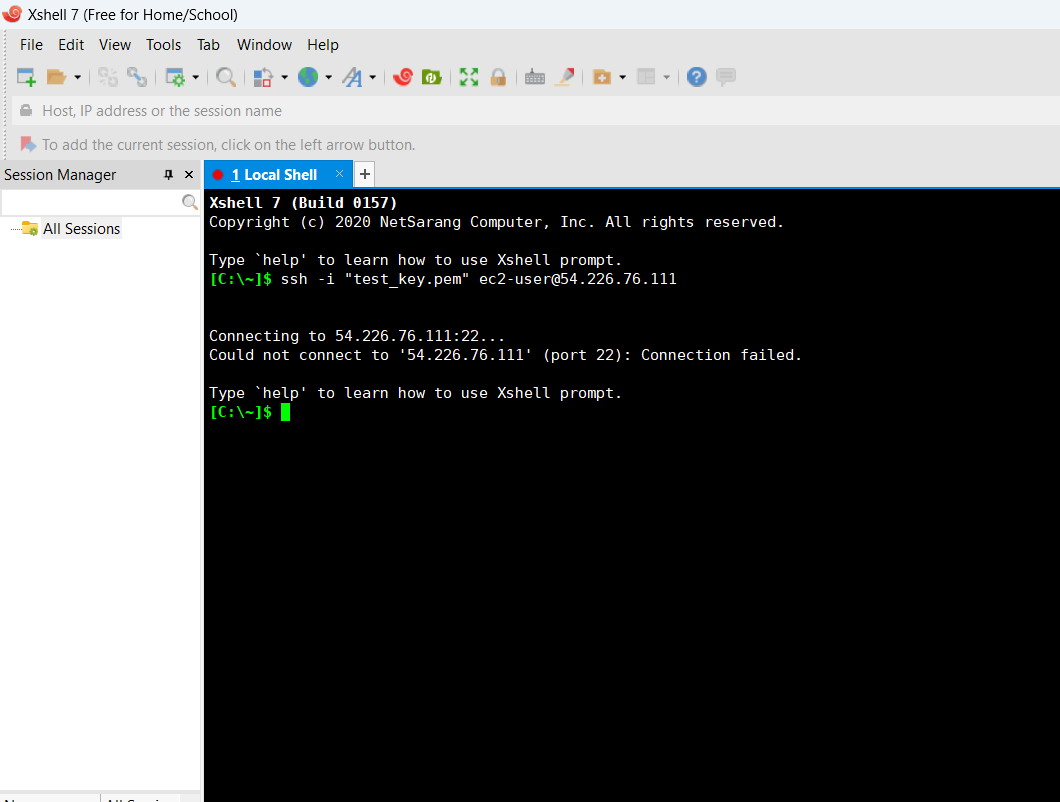
* **The public and private subnet, both are into different availability zone.**
* **The resources of public subnet and private subnet are hence present in different availability zone.**





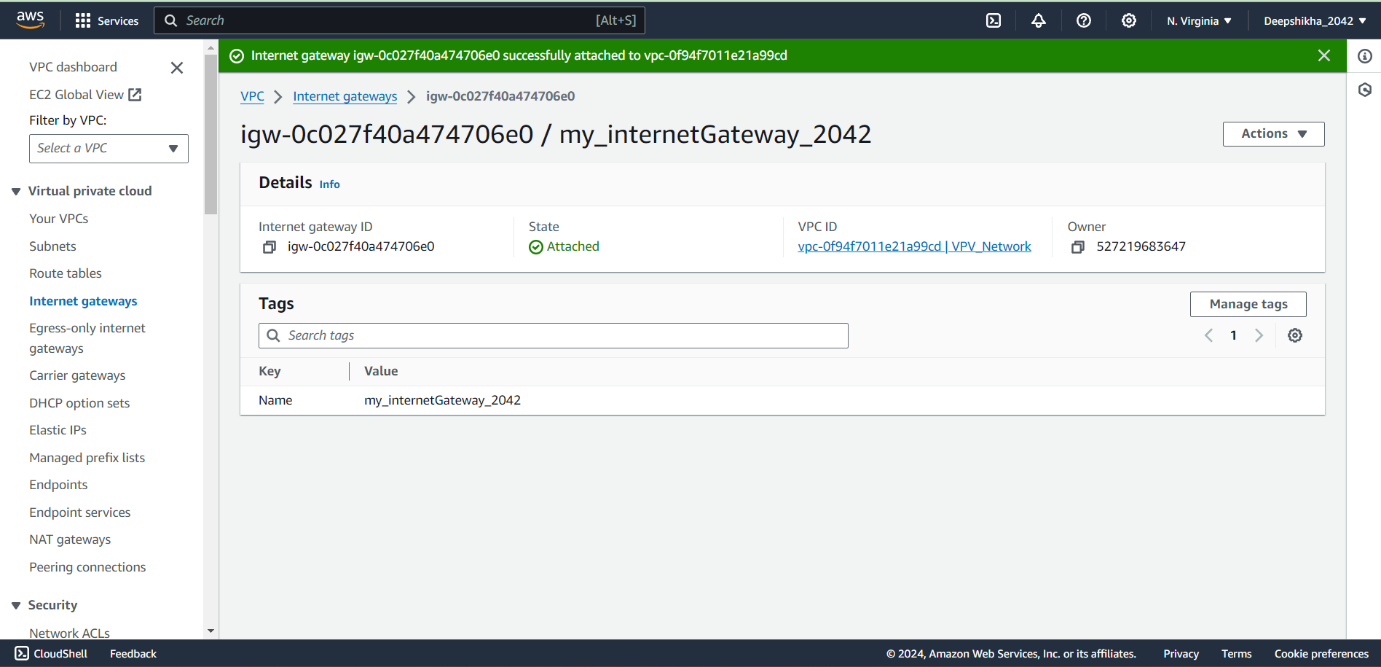
* **STEP-5:**

After that, I tried connecting the web-server, but could not connect. This is because, the internet-gateway is not defined yet.



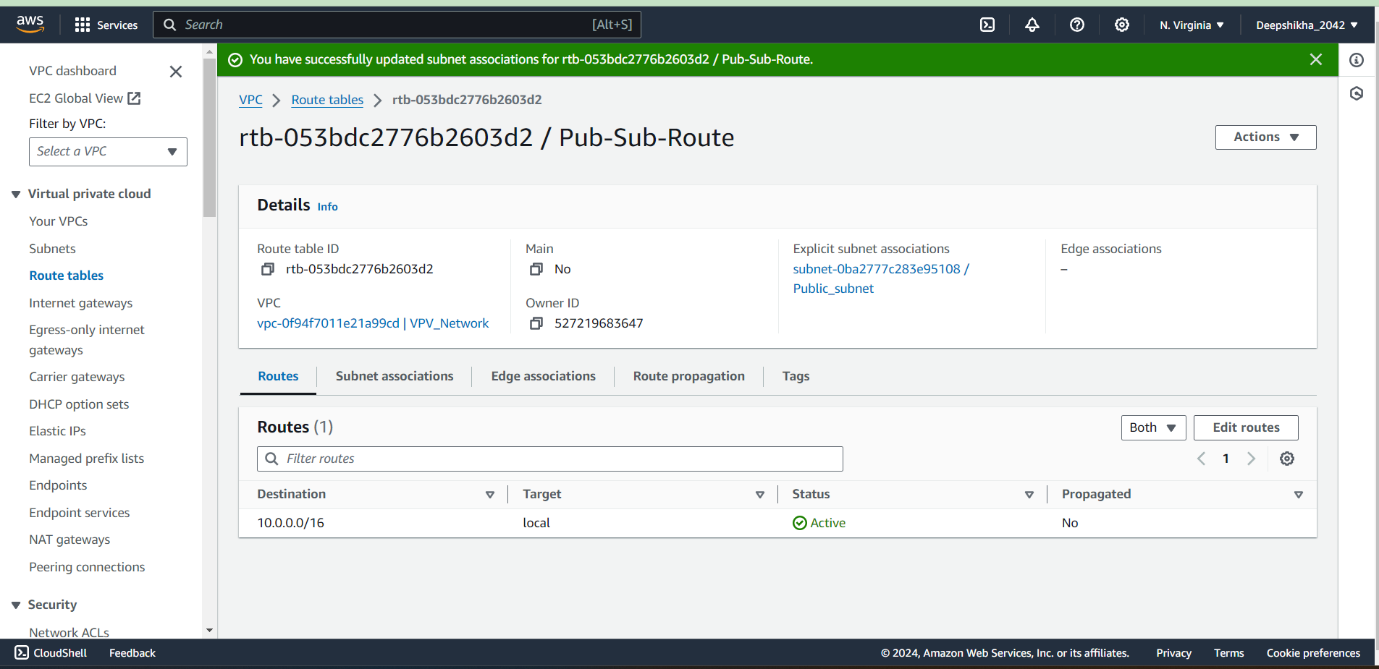
* **STEP-6:**

Hence, I created internet-gateway and attached it to the VPC that I just created.

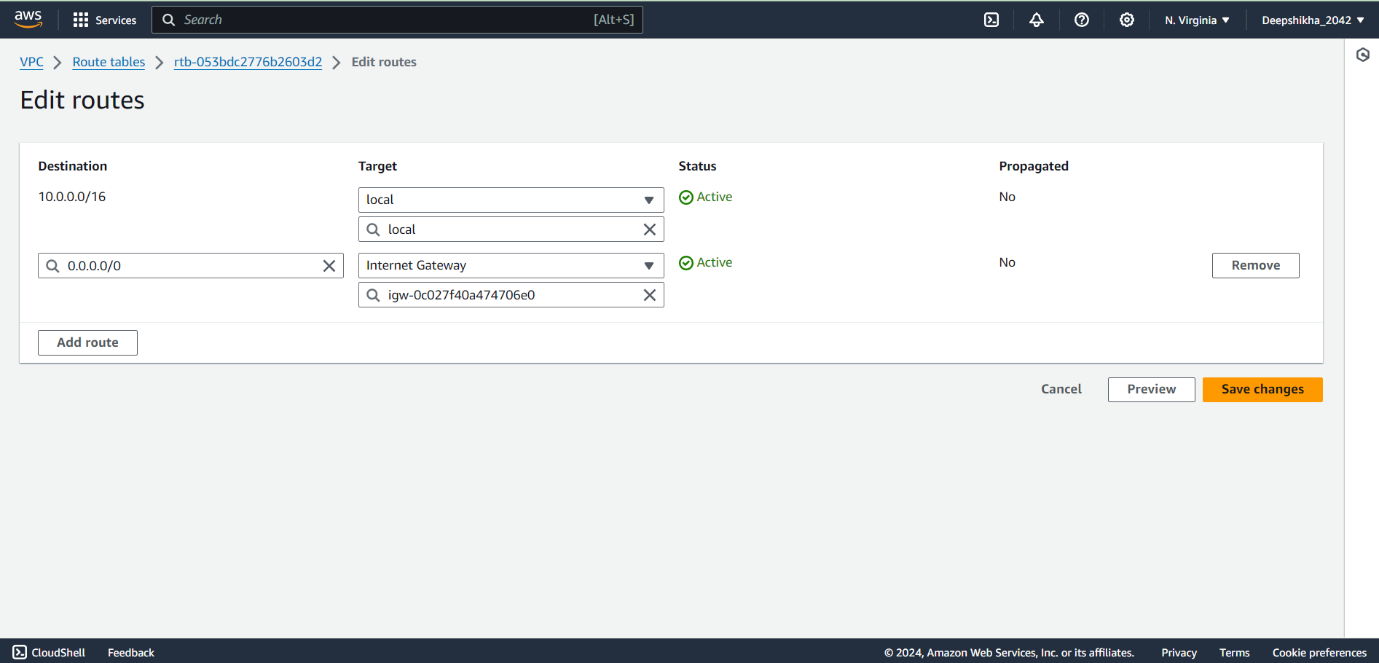


* **STEP-7:**

Then, I defined routing. I created a route table for the public subnet and associated that route-table with the public subnet.

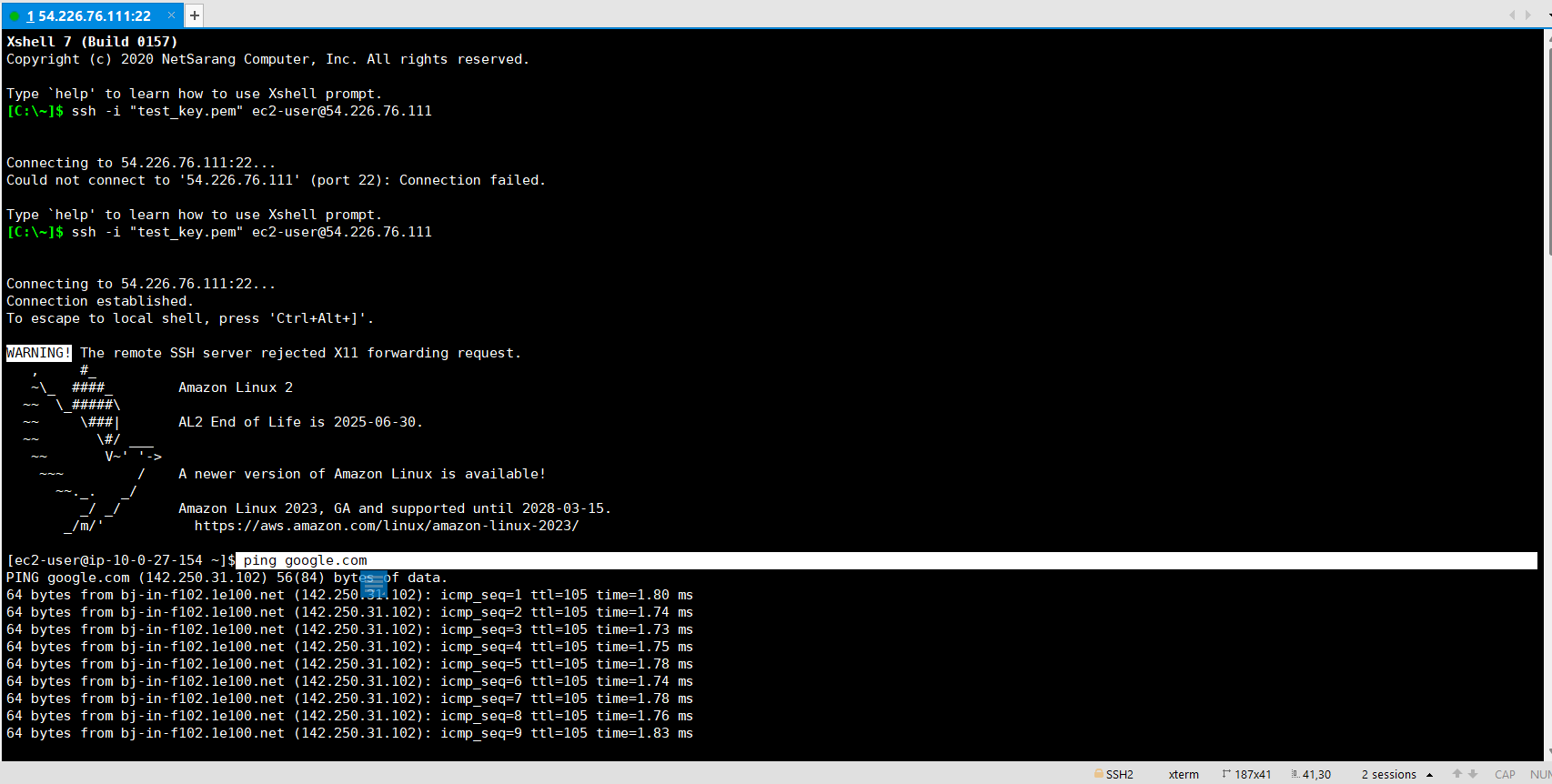


In the route-table, I added the policy for the subnet to access the internet via the internet-gateway.



* **STEP-8:**

Since I have the internet access now, I can connect to the web-server present in the public subnet. Also, the internet can be accessed from the web-server.

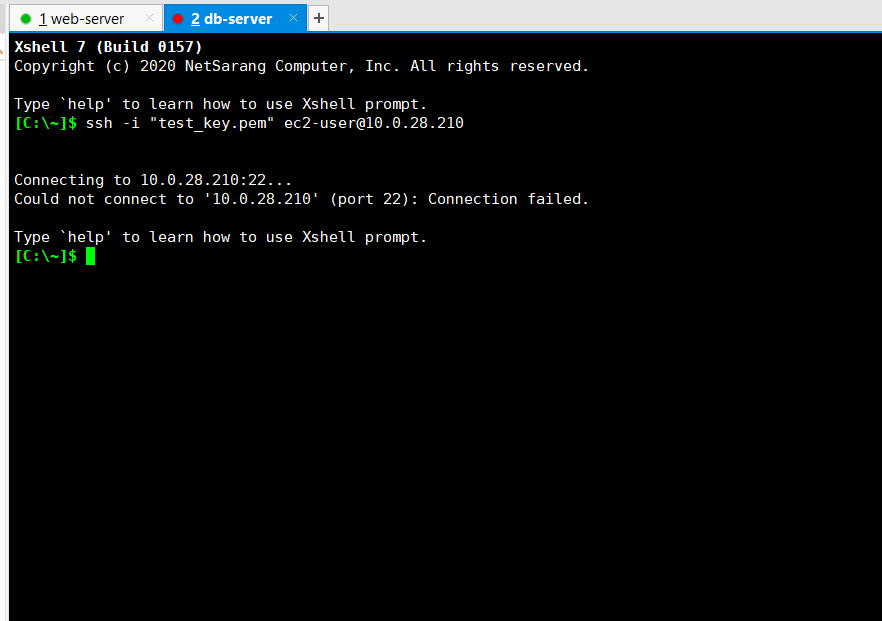


Web-server Internet

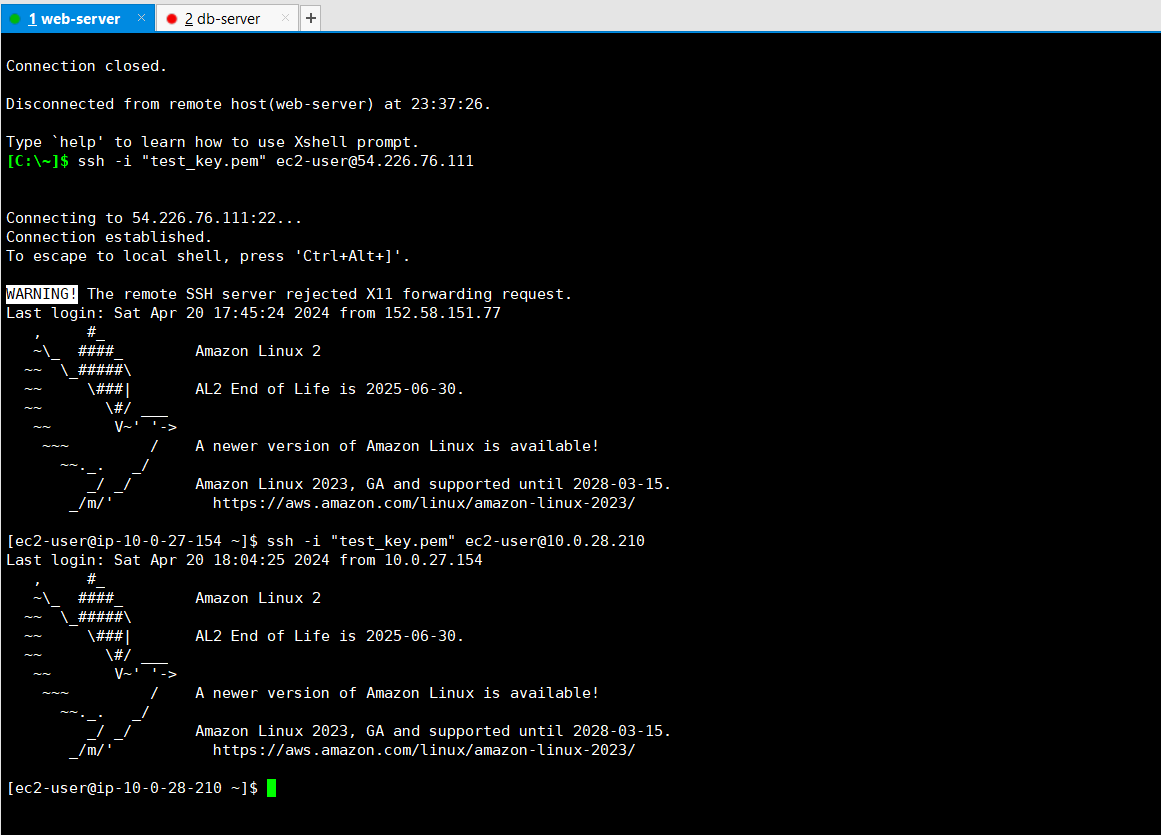
TWO-WAY COMMUNICATION

* **STEP-9:**

Then, I tried connecting the database-server present in the private subnet using the SSH command but I could not because it has only private IP address, hence cannot be accessed publicly.



Then I tried accessing the database-server present in private-server from the web-server present in the public subnet. I observed that I can access it (database-server) via the web-server since they are connected to each other because both of them are under same network and the web-server has the internet access.

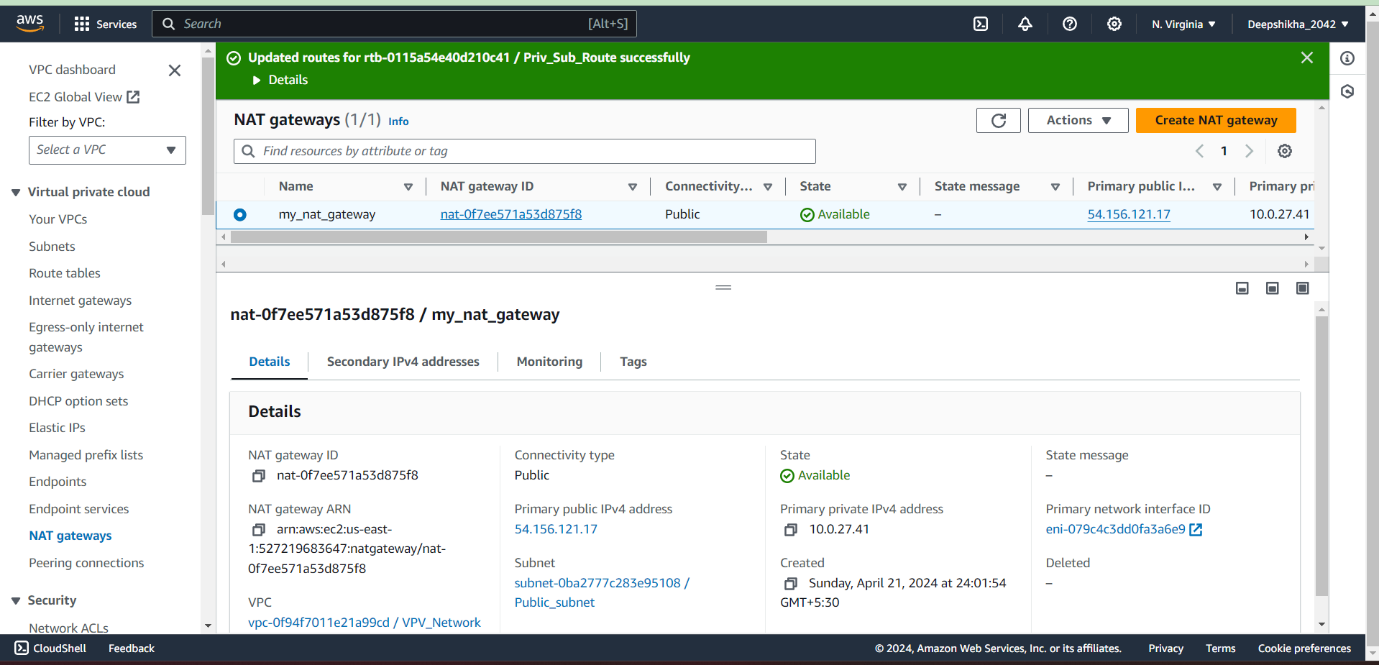


Neither the database-server can be accessed from outside as it has only private IP address, nor one can access internet from the database-server.

* **STEP-10:**

(First way to access the database-server using NAT-GATEWAY)

To access the internet from the database-server, we need to define a NAT-Gateway. So, I created a NAT-Gateway inside the public subnet.

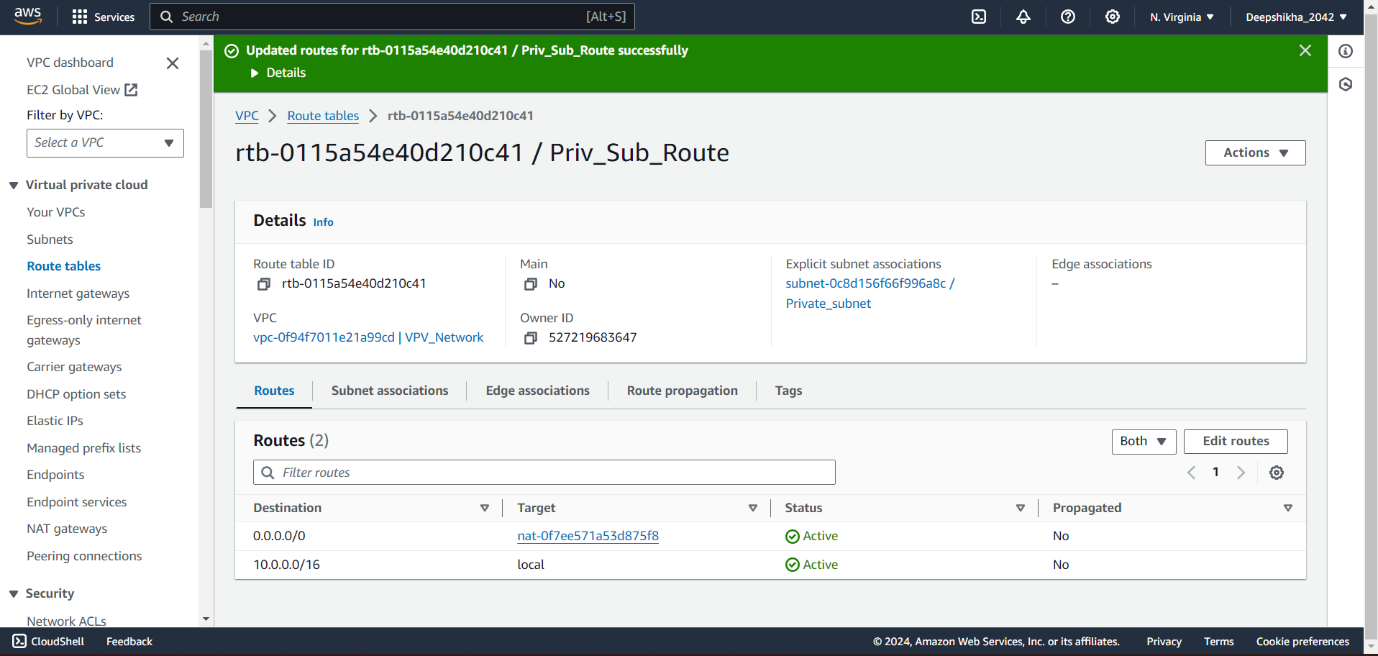


* **STEP-11:**

Firstly, I created a route-table and associated that with the private subnet.

In the route-table, I added routing policies for private subnet to access the internet via NAT-Gateway.

Now, I can access internet from the database-server.

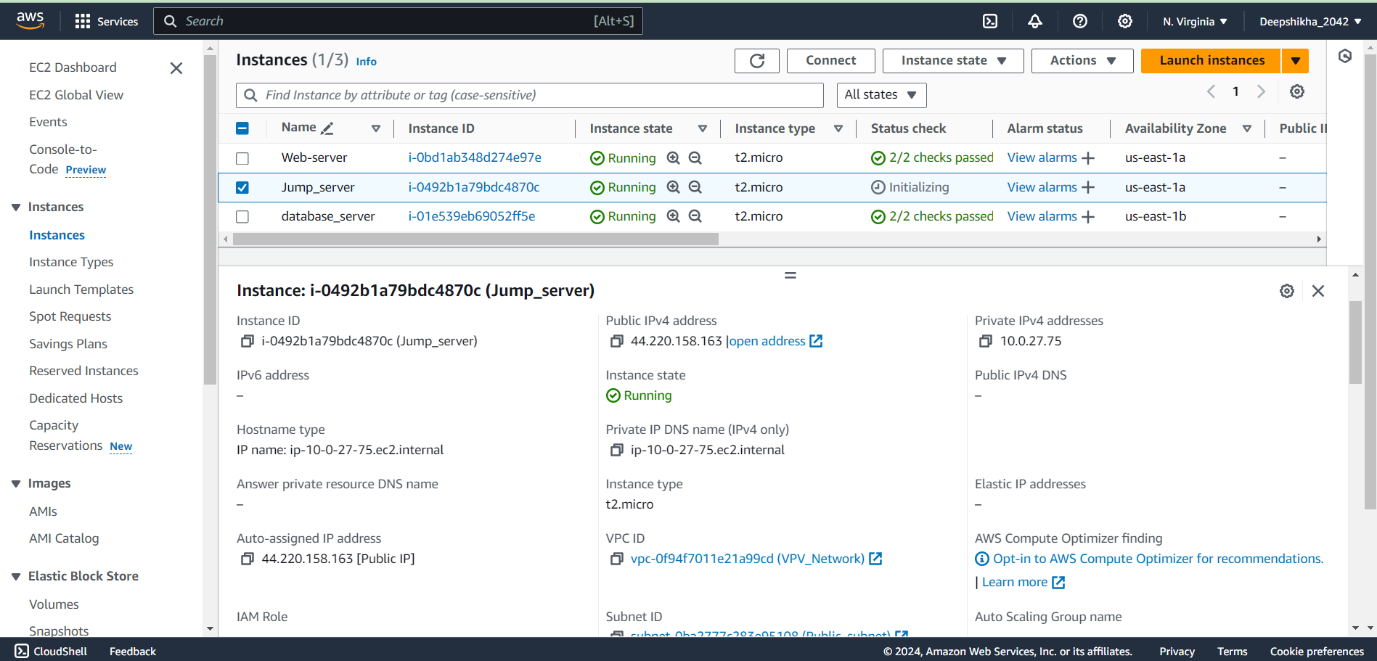


* **STEP-12:**

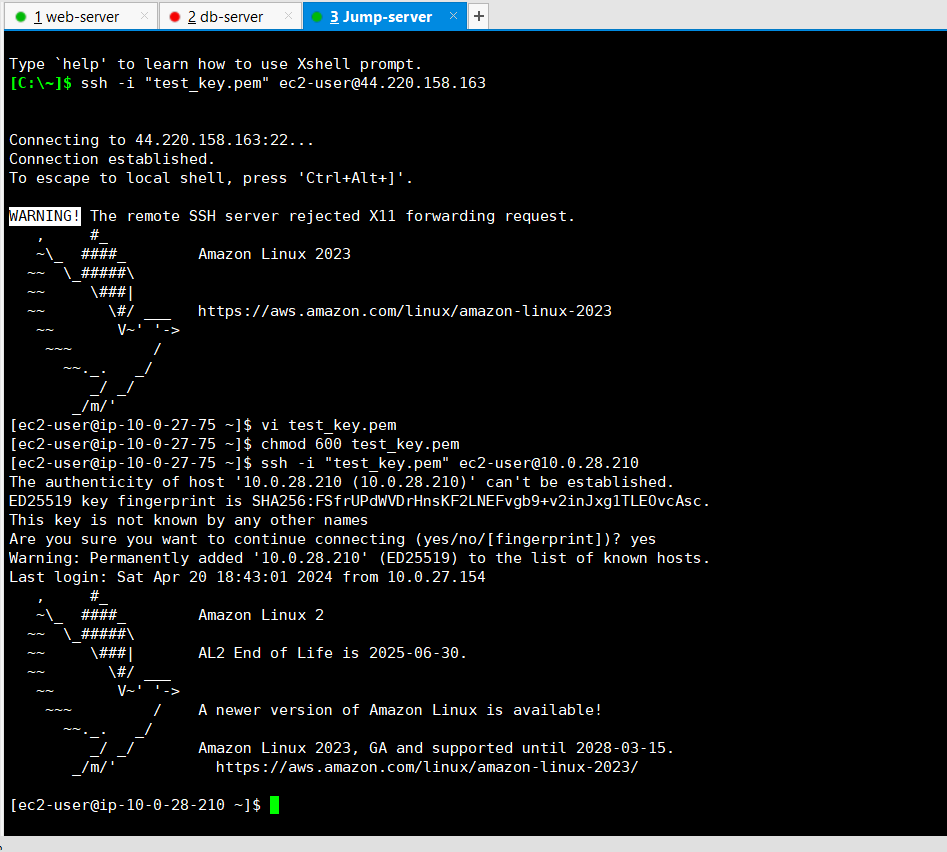
But, what if a developer (a database administrator may be) want the access to the database-server? Does he need to access it via the web-server every single time?

We want to give the access of the database-server to the developer only, not to the public.

For this, I created a jump server.



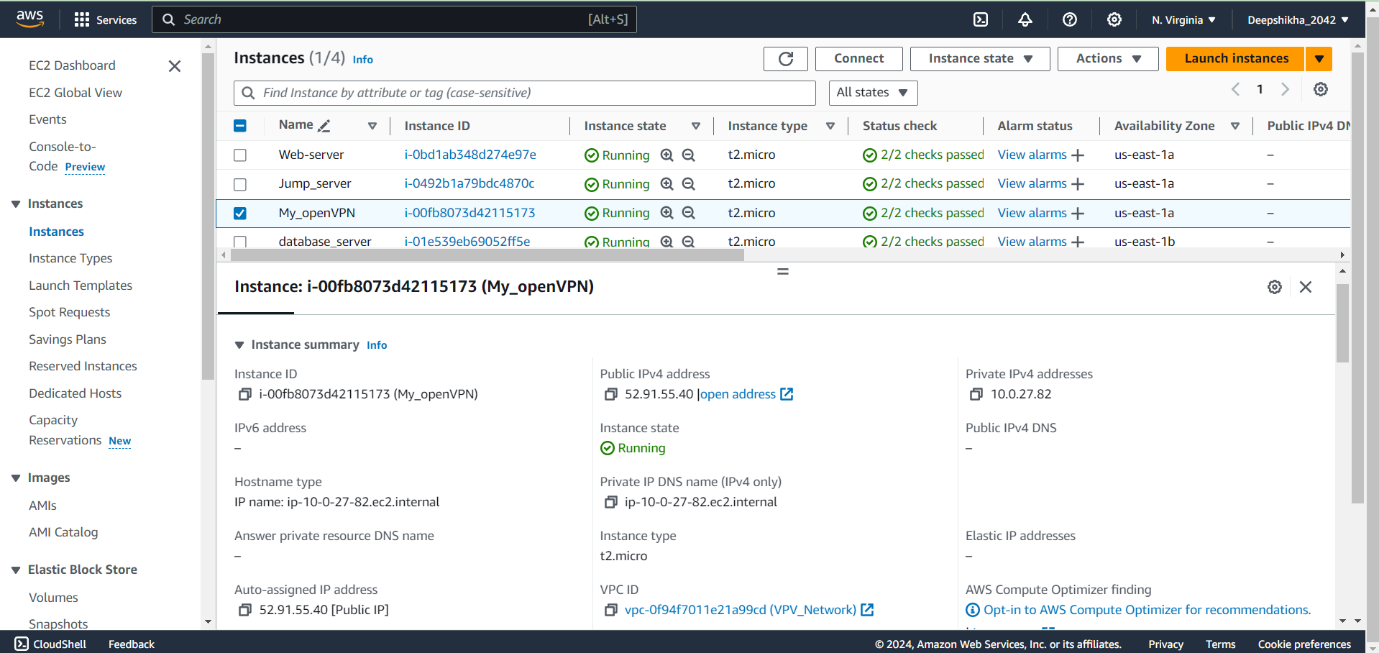
One can access the database-server via the web-server, but it is better to have a different server (jump-server) for accessing purpose. This is because, the web-server already have different important roles (application purpose)



* **STEP-13:**

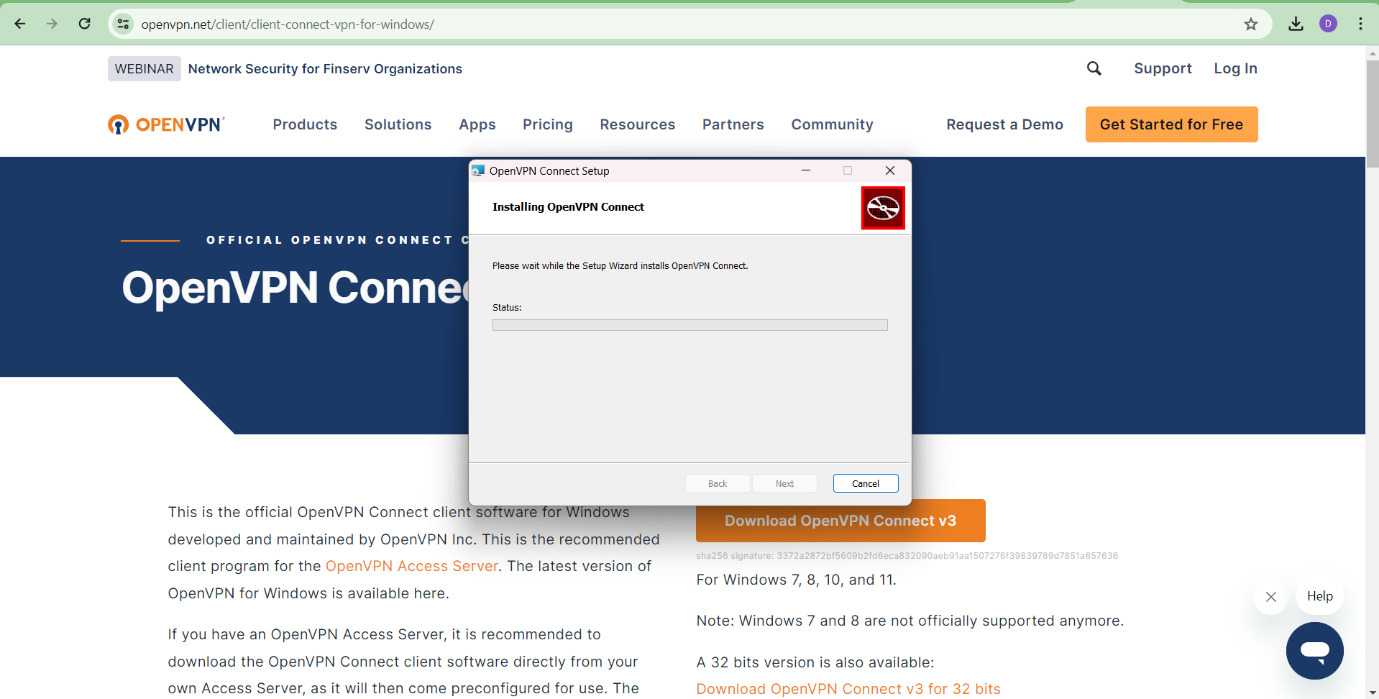
(Second way to access the database-server using OpenVPN)

I created OpenVPN in the public subnet.



* **STEP-14:**

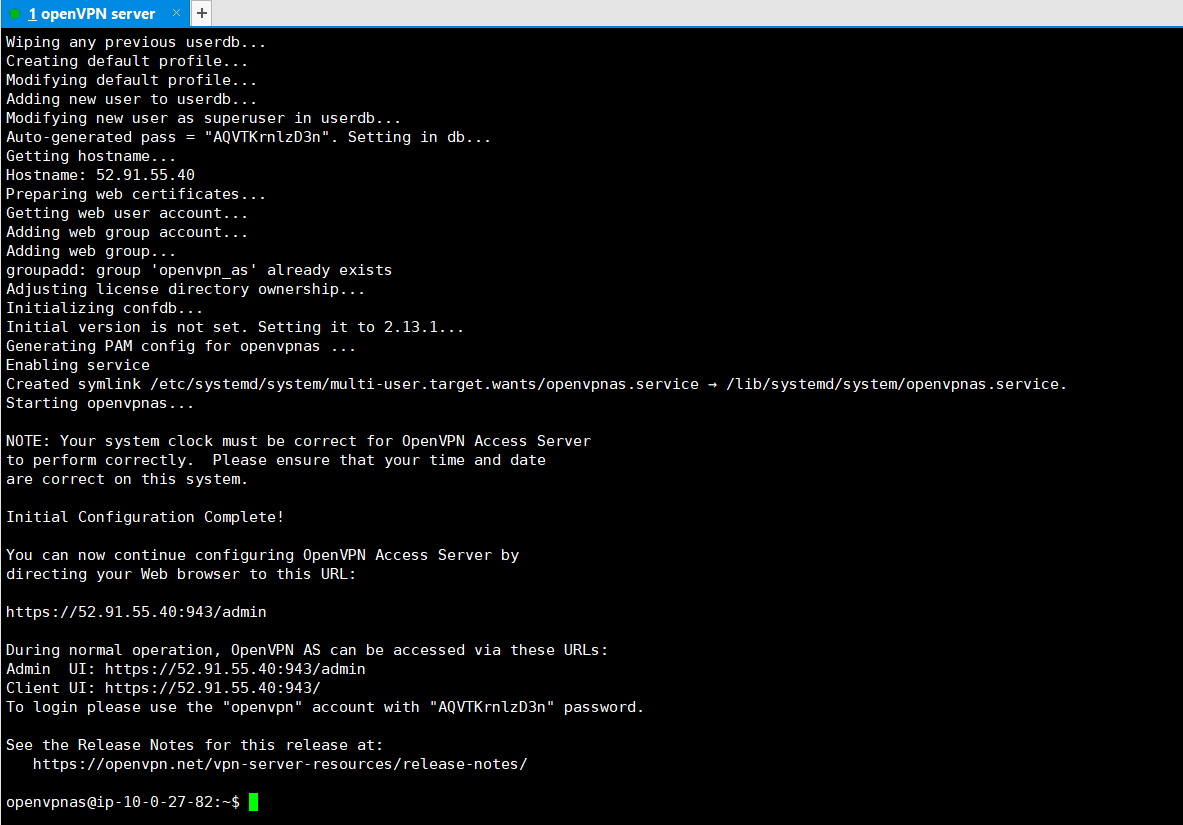
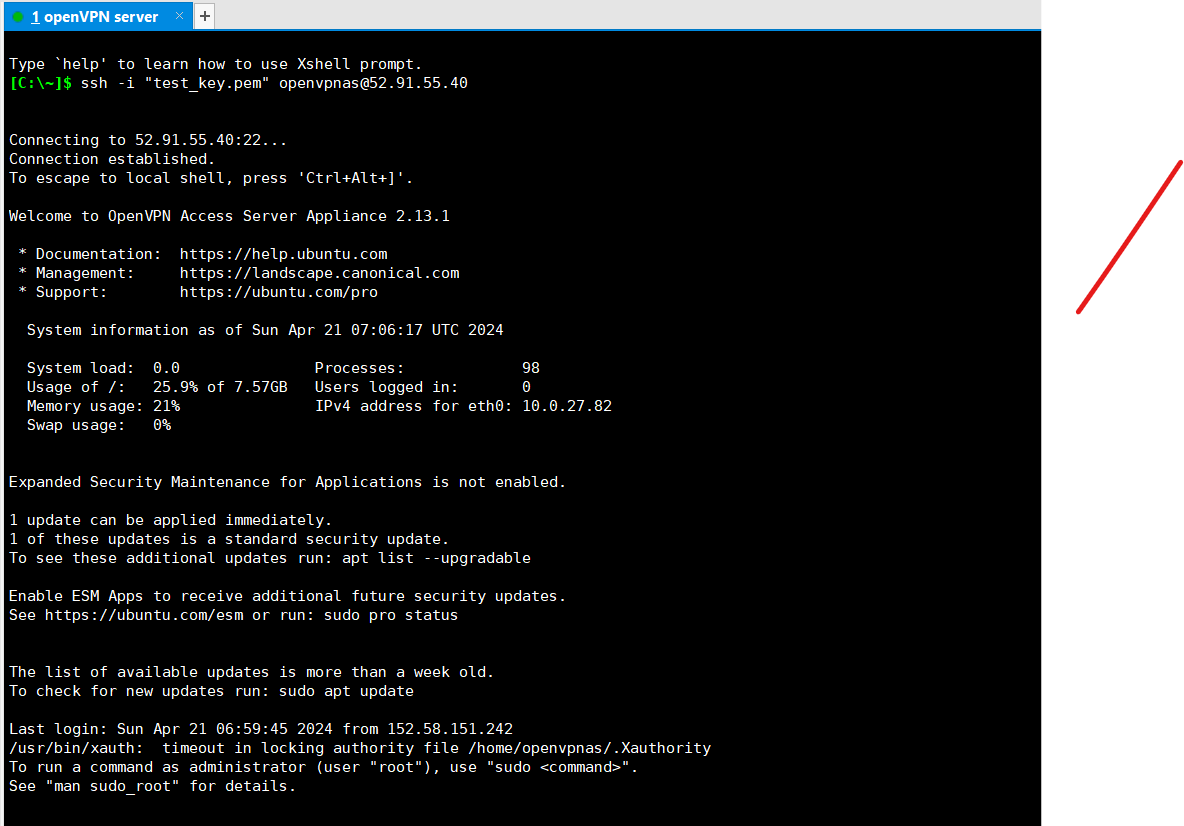
Installed VPN Client in the system.

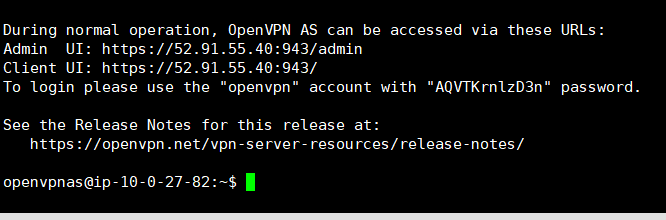


* **STEP-15:**

Then, I connected to the VPN server and I got the following credentials:

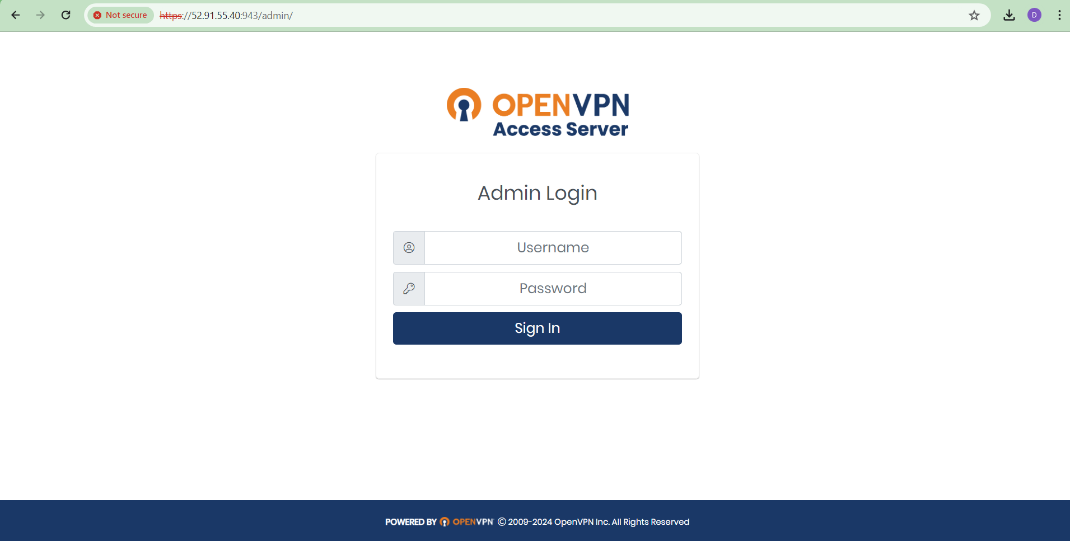
* Admin UI
* Client UI
* Access ID and Password



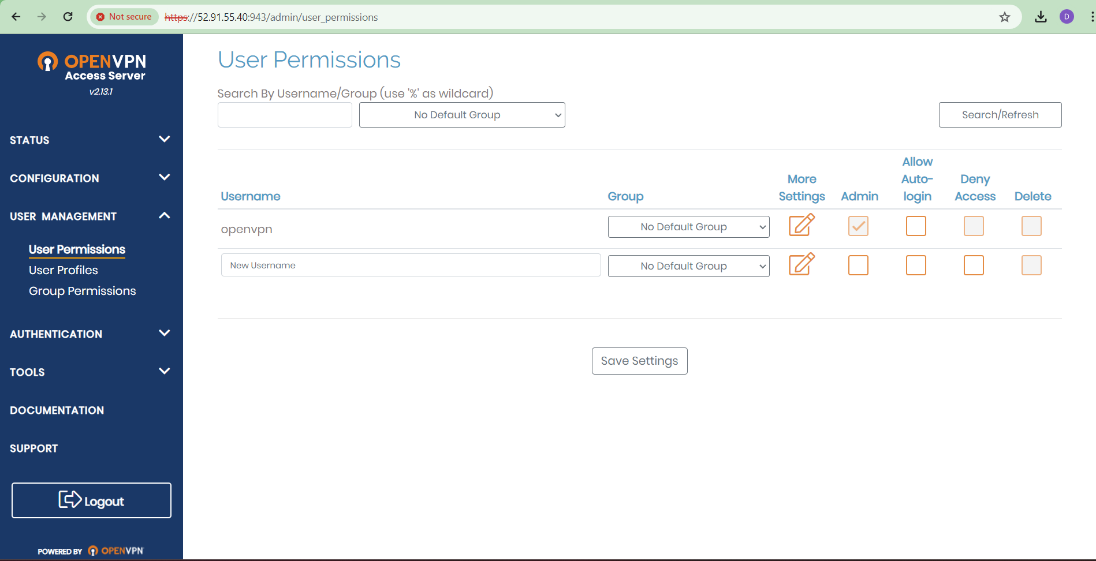


* **STEP-16:**

Then, I just pasted the Admin UI in the search engine and got the following page.

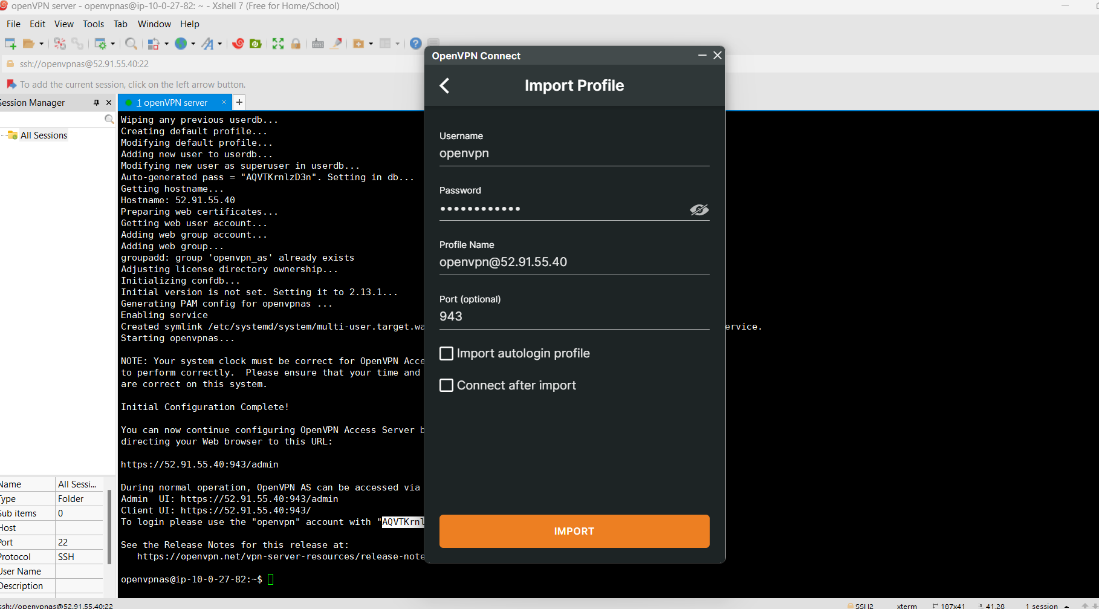


From here, one can handle user permission, group permission, user profile etc.

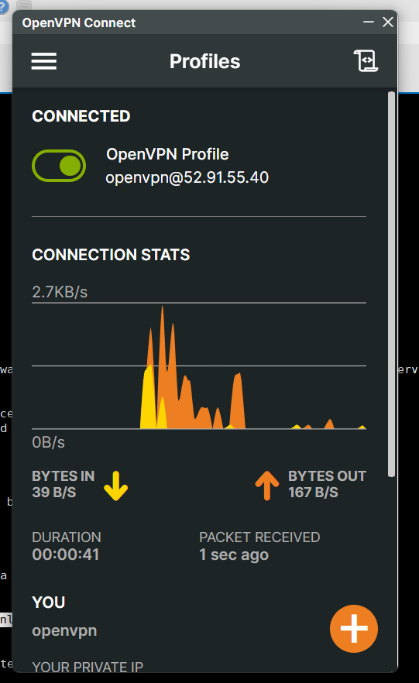
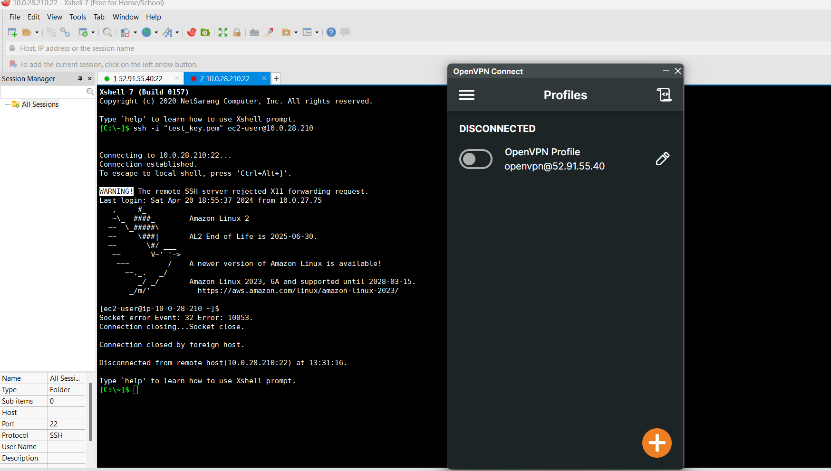


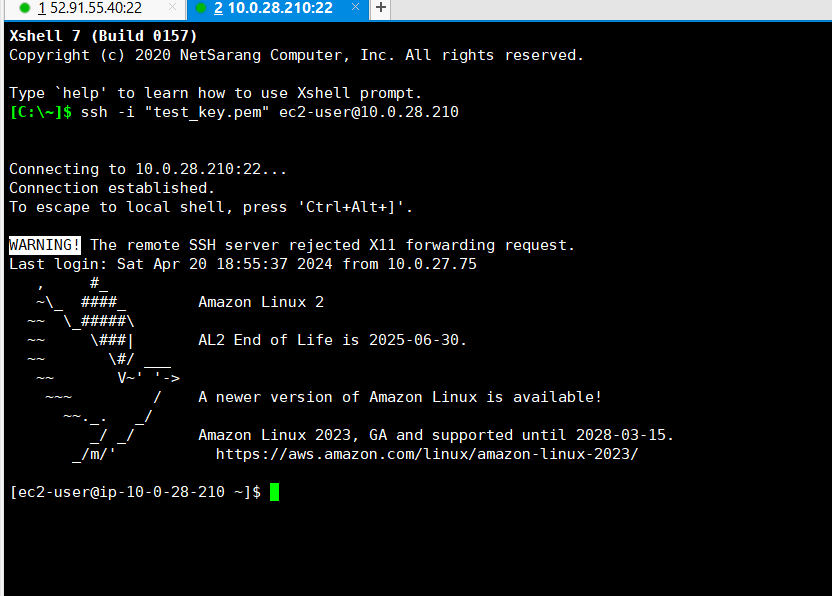
* **STEP-17:**

Using the Client UI, Access id and password, I was able to connect to the AWS account. Now I (or may be the database-administrator) can connect to the AWS account even using the private IP address.



The database-administrator can connect and disconnect to the AWS account based on the requirements.



**\*\*\*\* END \*\*\*\***