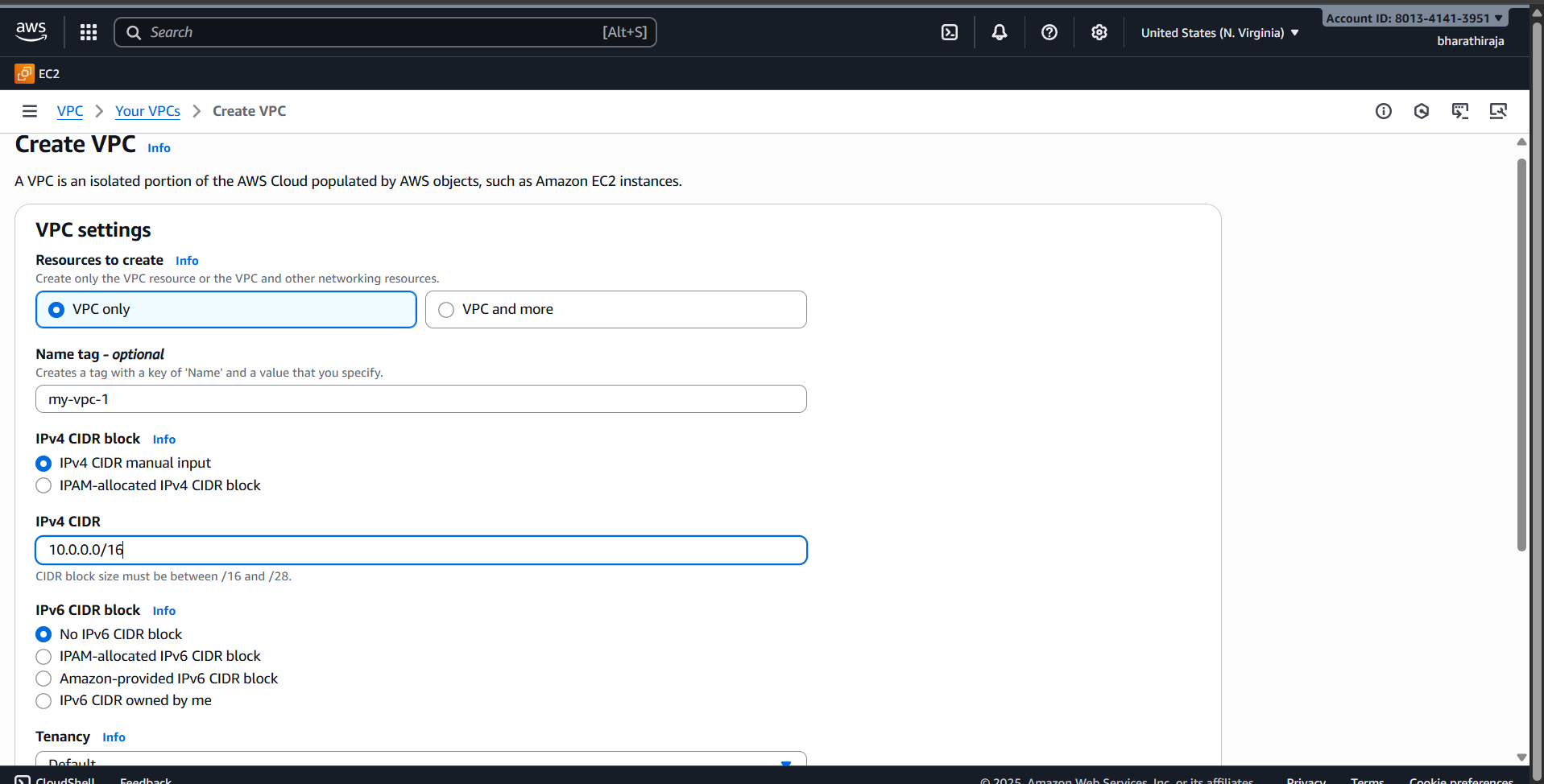
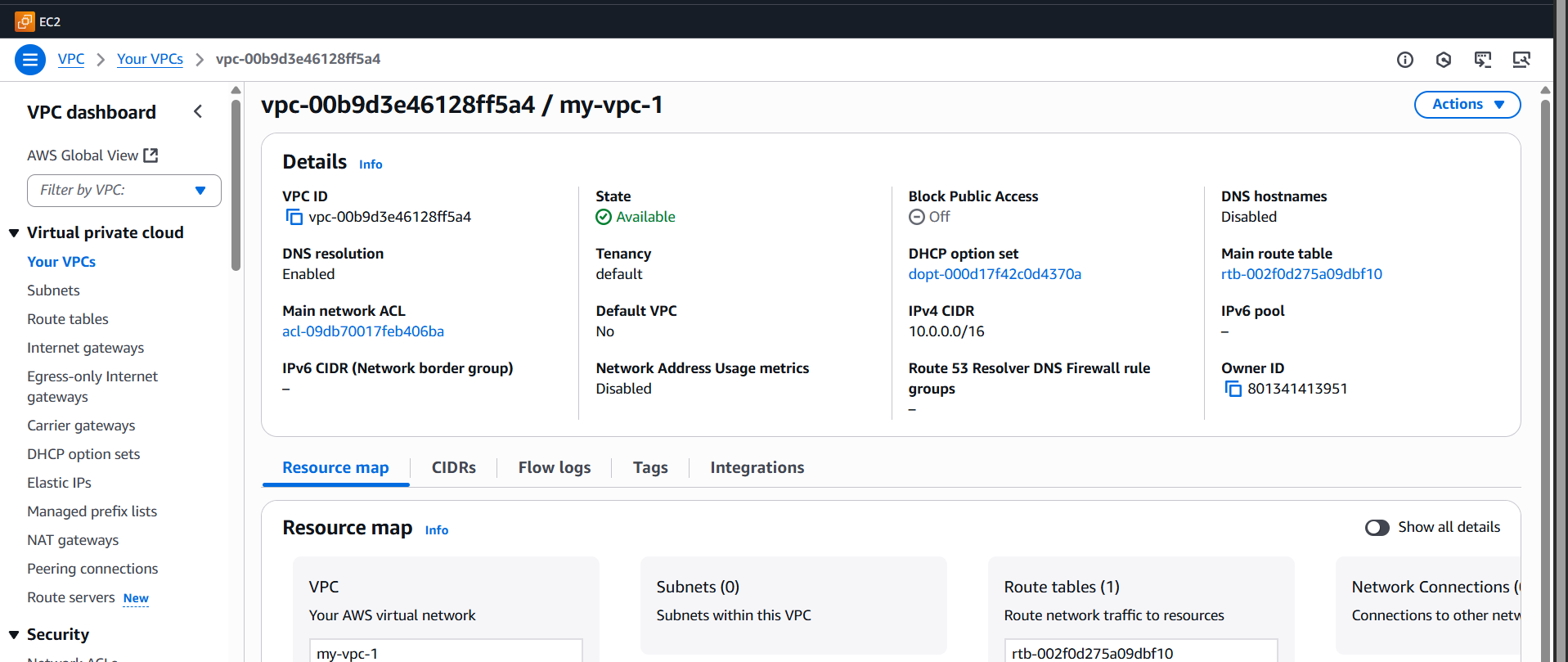
AUTO SCALLING GROUP

1. Create one VPC in the North Virginia region.

Go to the VPC part, click on it, then create the VPC   
then after you name it as you want or something related to your project, based on

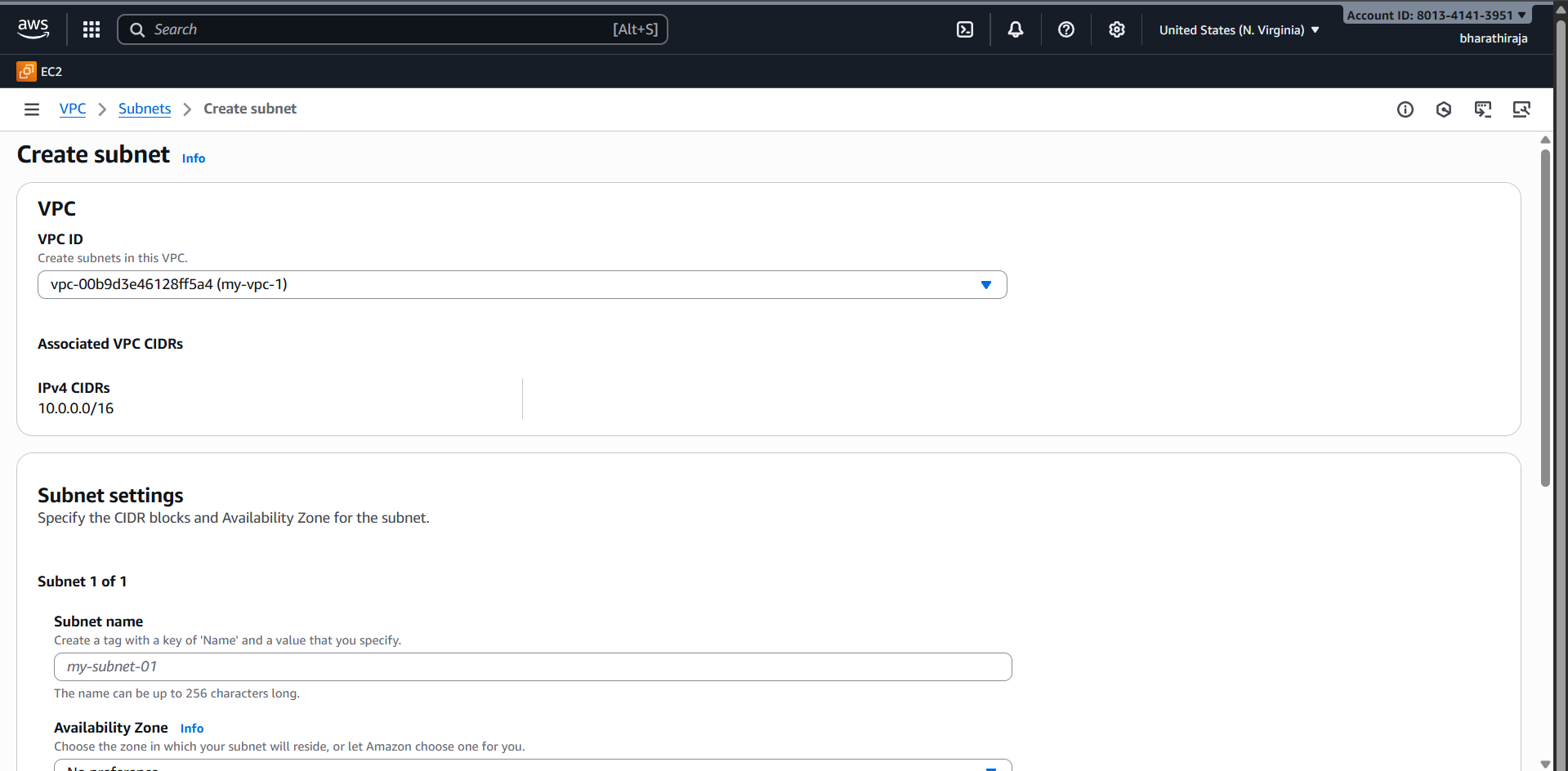
* + Gives ipv4 CIDR range as much as you want



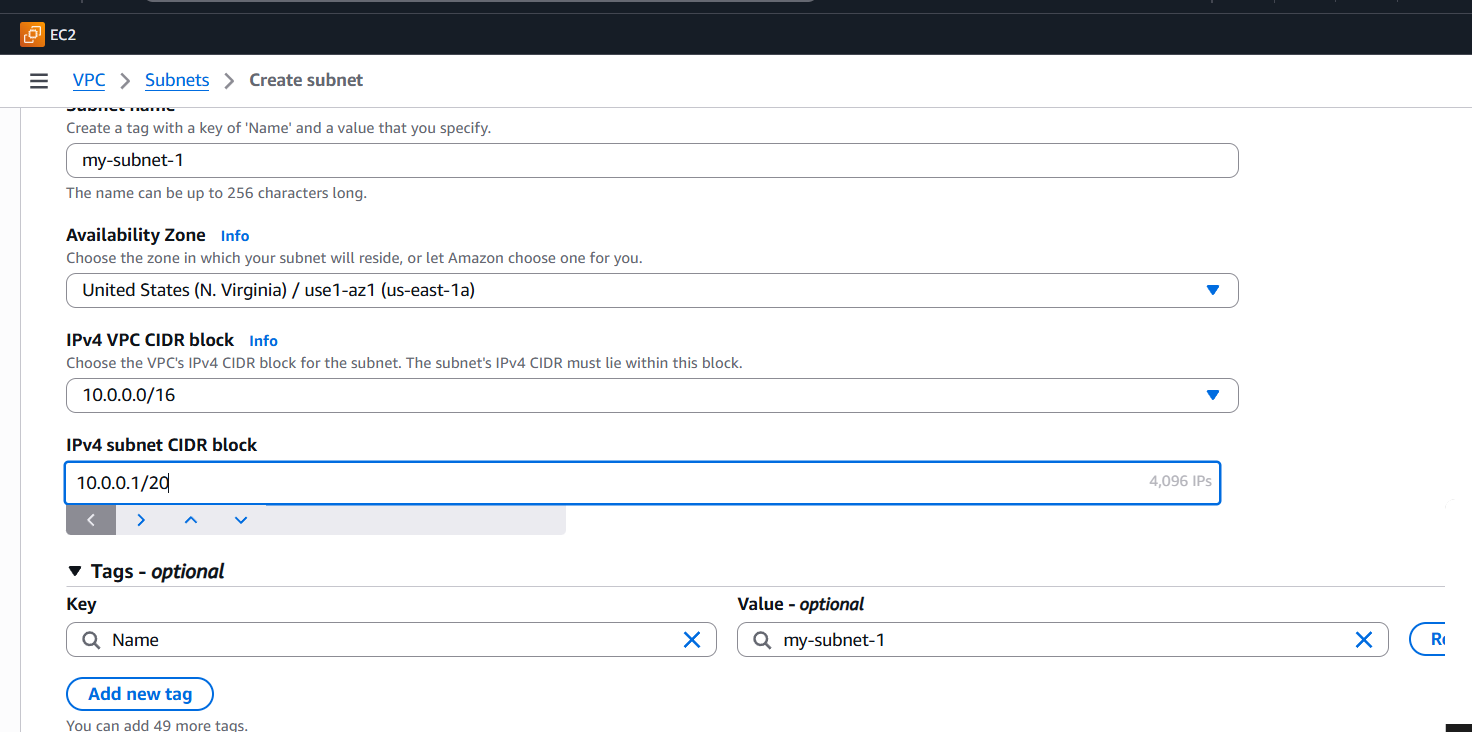
Then click on create and successfully created   


1. Create two subnets: one public subnet and one private subnet.

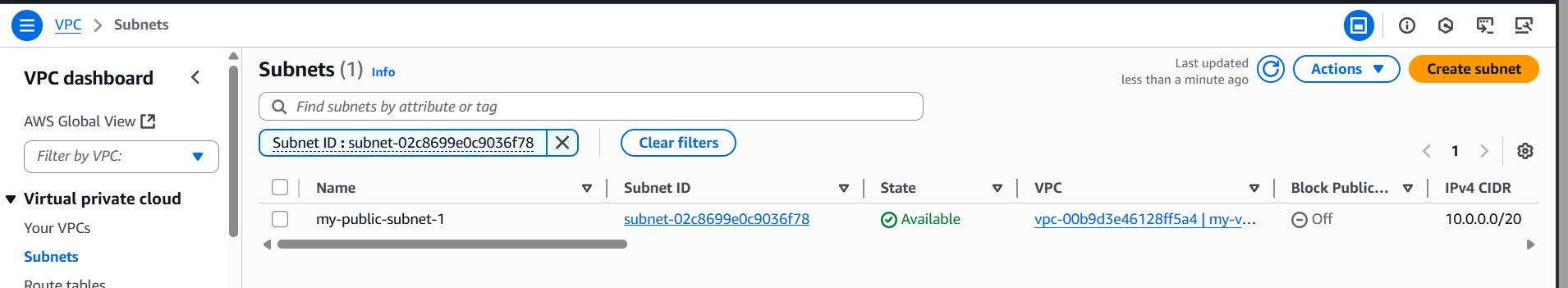
Go to the subnet part, click on it   
create the subnet, give details as needed



I give the range IPv4

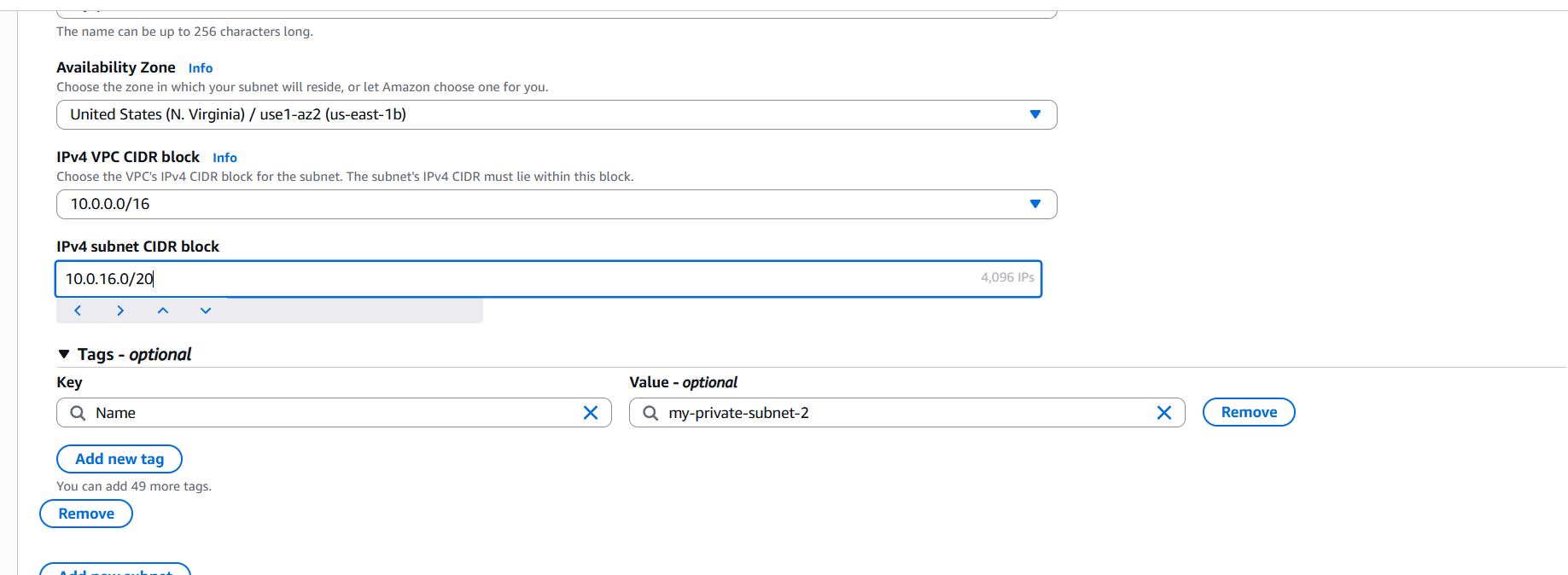


Then this public subnet-1



The Private subnet has been opened

We need to create the subnet



We can see the private and public subnets

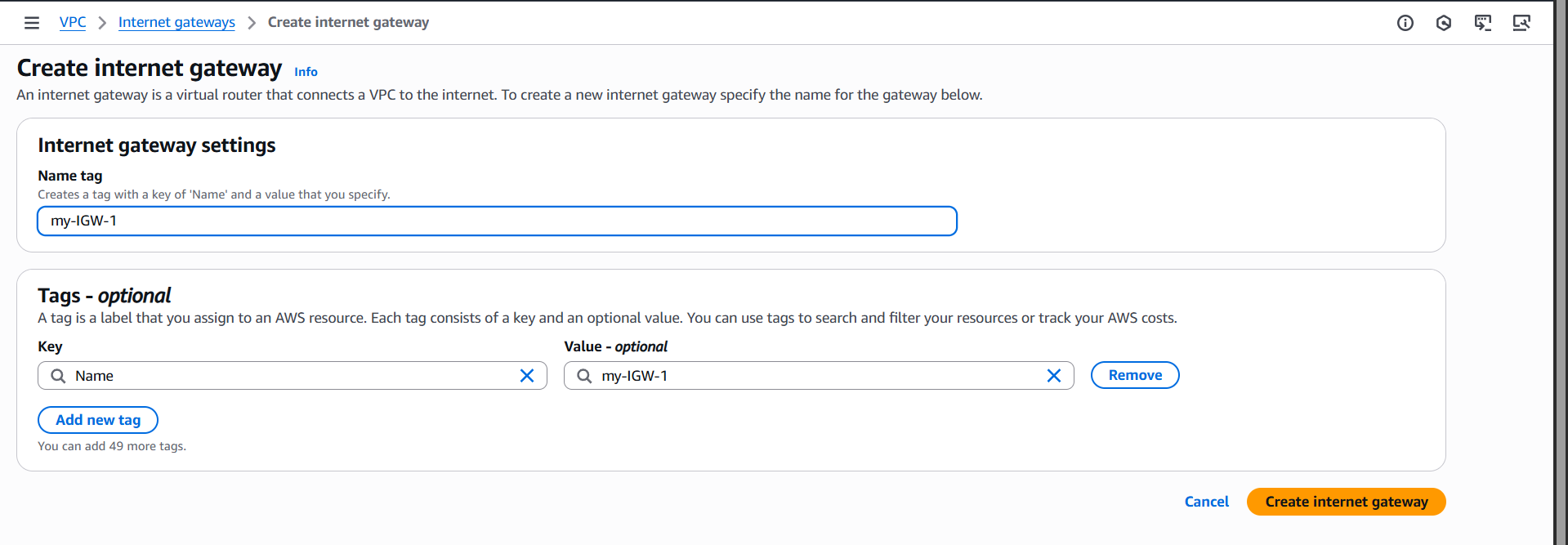


1. Attach an IGW to the VPC.

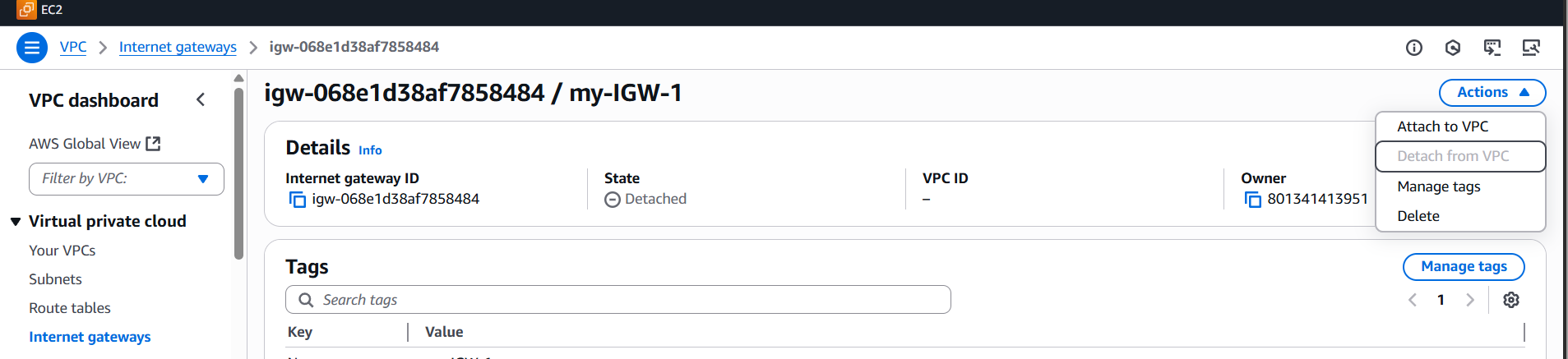
First, we need to create an internet gateway, then we have to attach to the VPC

Then, after we attach to the VPC

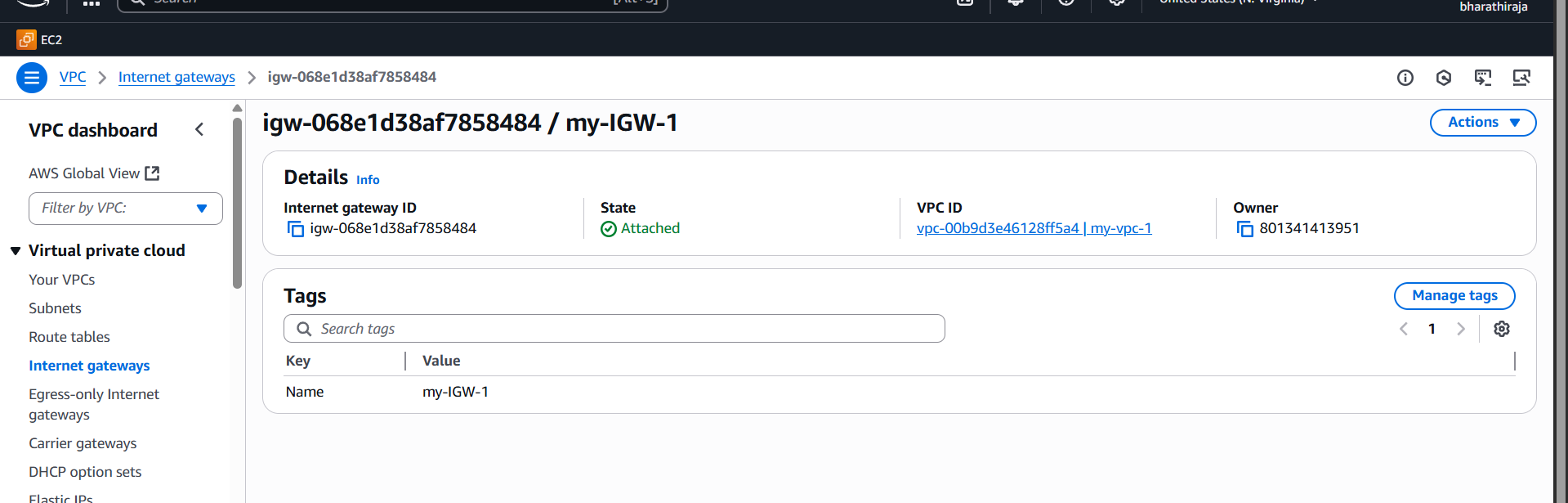
Click on Create IGW, go inside give the name as you want



Created after the right corner option showing attachment to VPC like that you have to do

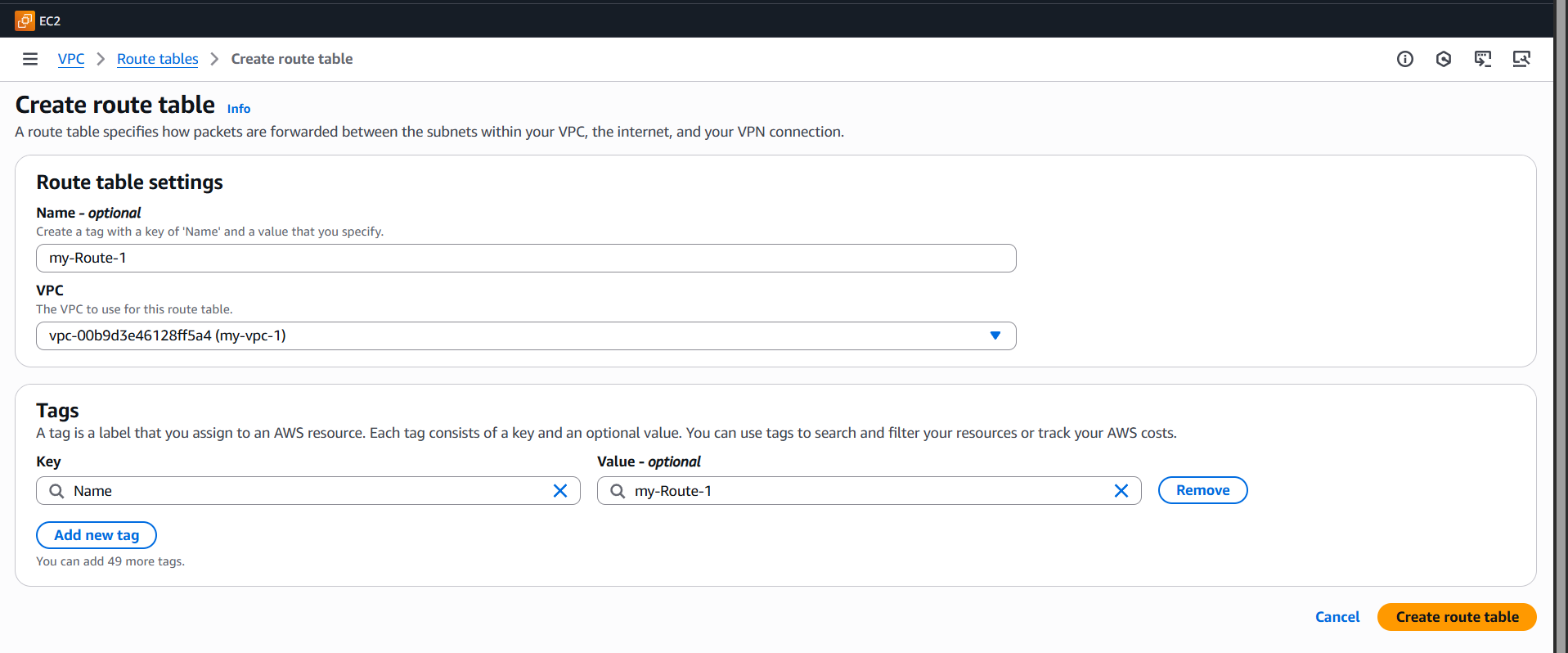


You see the difference between attaching the VPC and without attaching the VPC

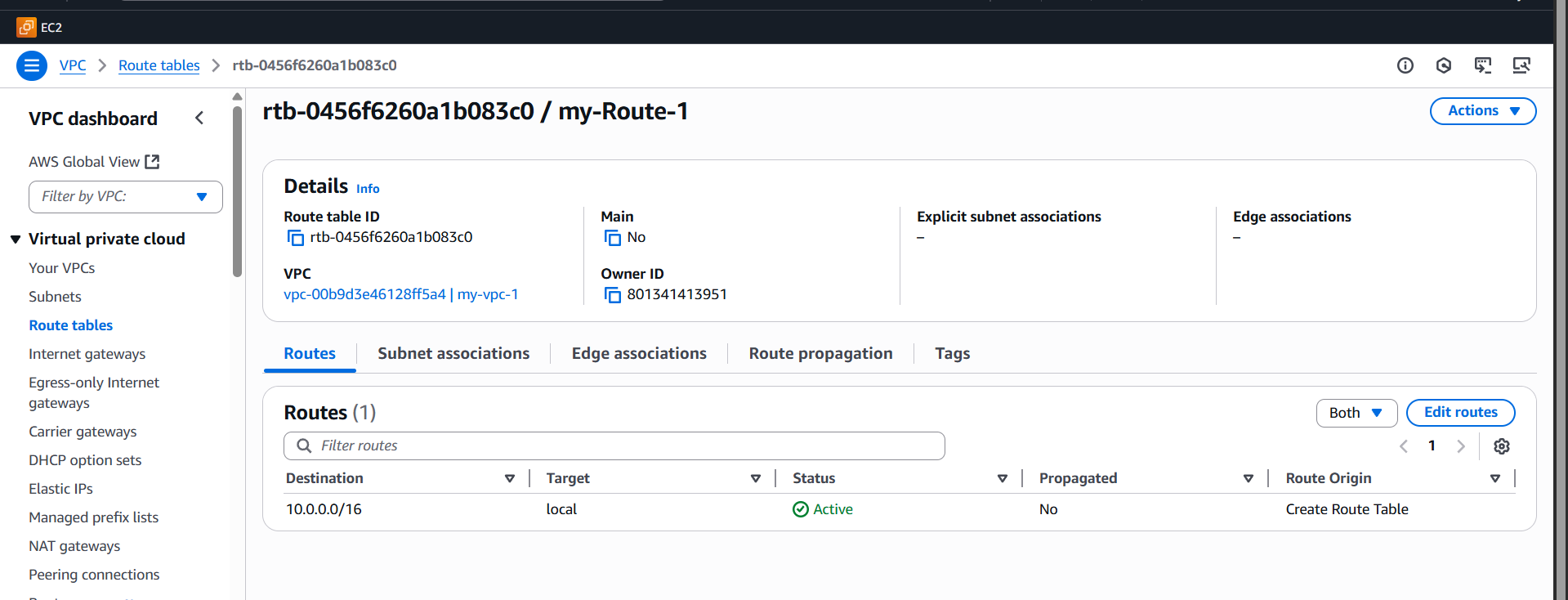


1. Create one public route table (RT) and one private route table.

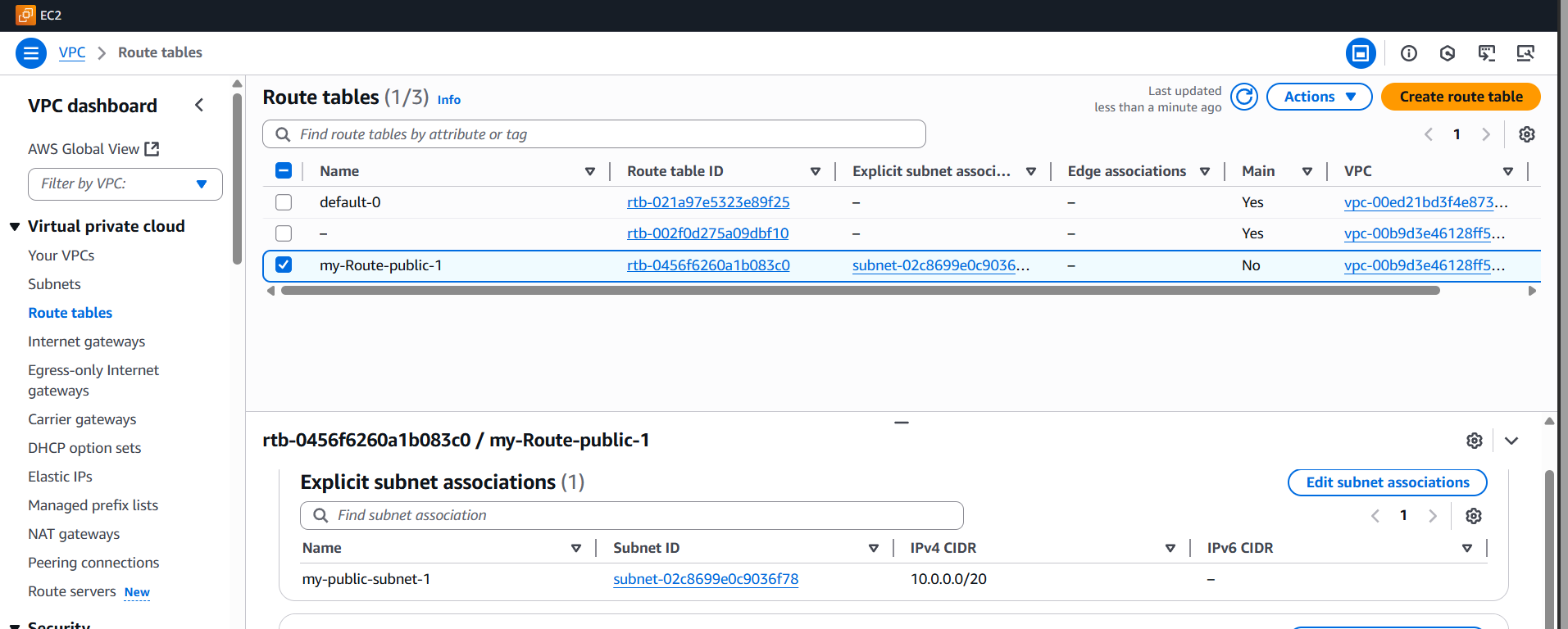
Go to the routable part, click on the Create route table   
which VPC do YOU ADD IT then create it



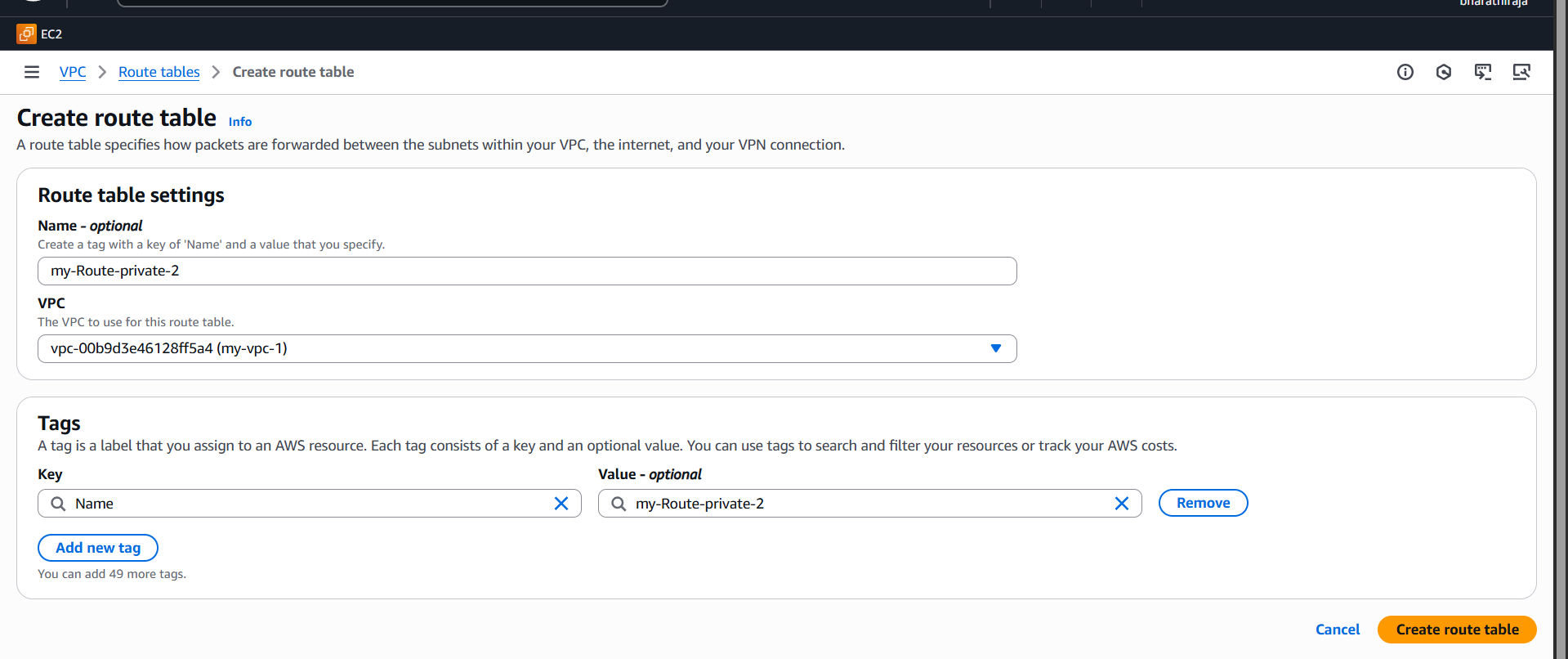
I created a subnet



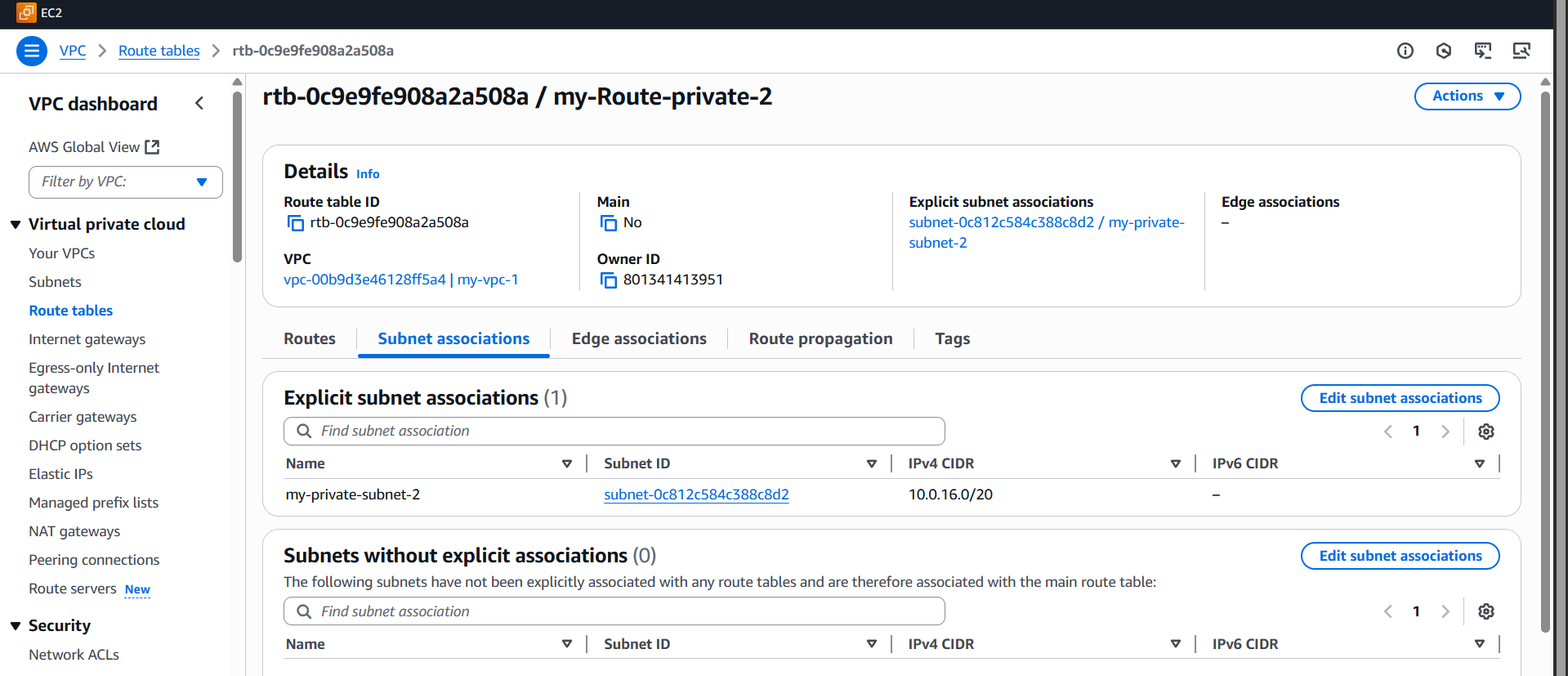
This is a public subnet. Once open route table then after, we have to add go to the inside subnet  
edit subnet association, then add public subnet. **Explicit subnet associations** (1)



This is a private



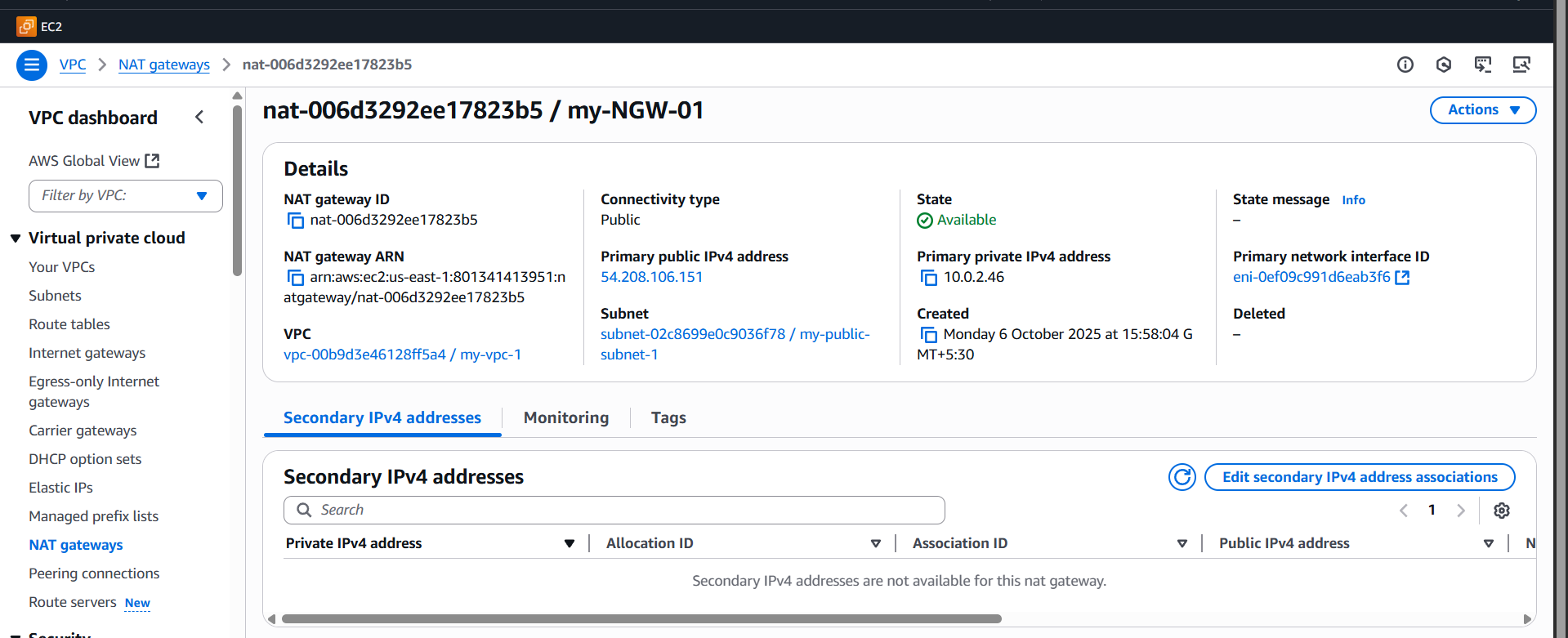
This is a private subnet that I have opened



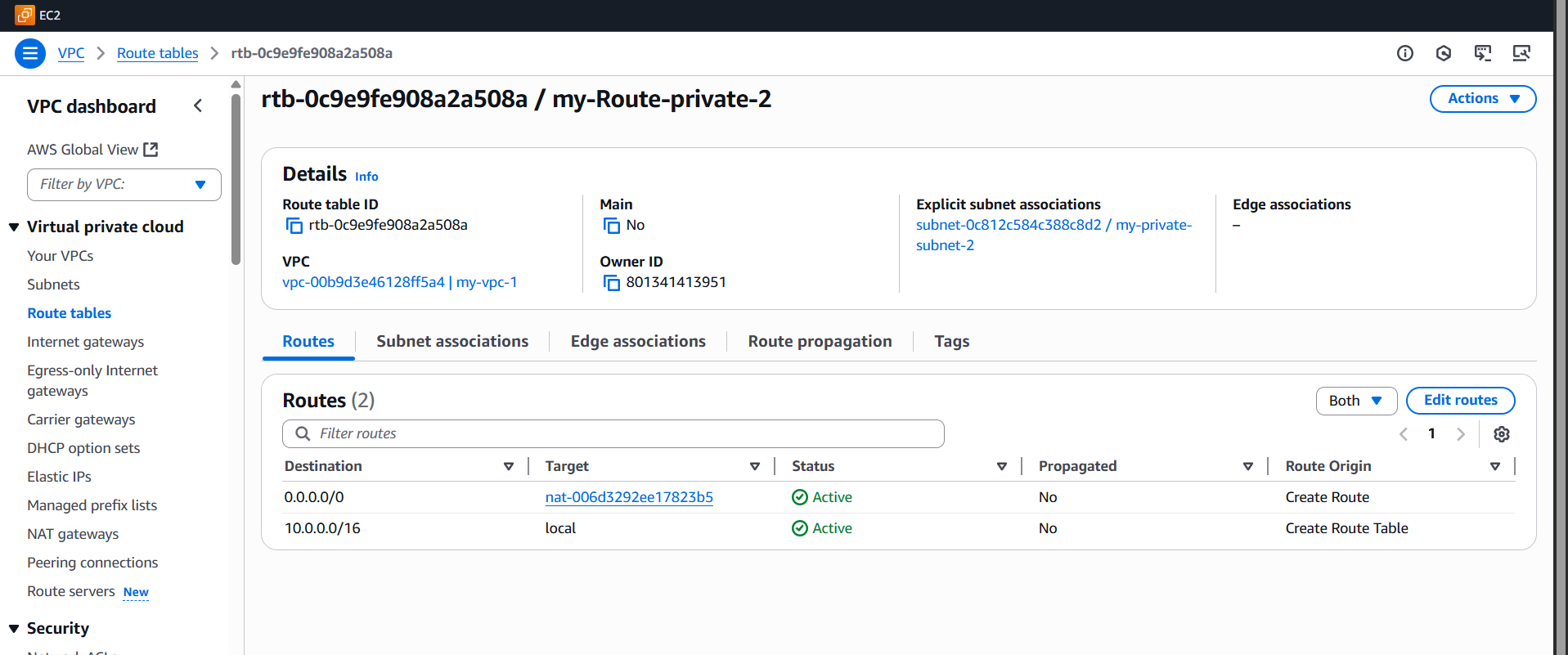
1. Deploy a NAT gateway in the public subnet and attach the NAT gateway to the private subnet

* I created a NAT gateway deployment in the public subnet **Name:** MyNATGateway
* **Subnet:** Select your **Public Subnet** (Example: 10.0.1.0/24)
* **Connectivity type:** Public
* **Elastic IP Allocation:**
  + Click **Allocate Elastic IP**
  + Then **Allocate and associate Elastic IP**

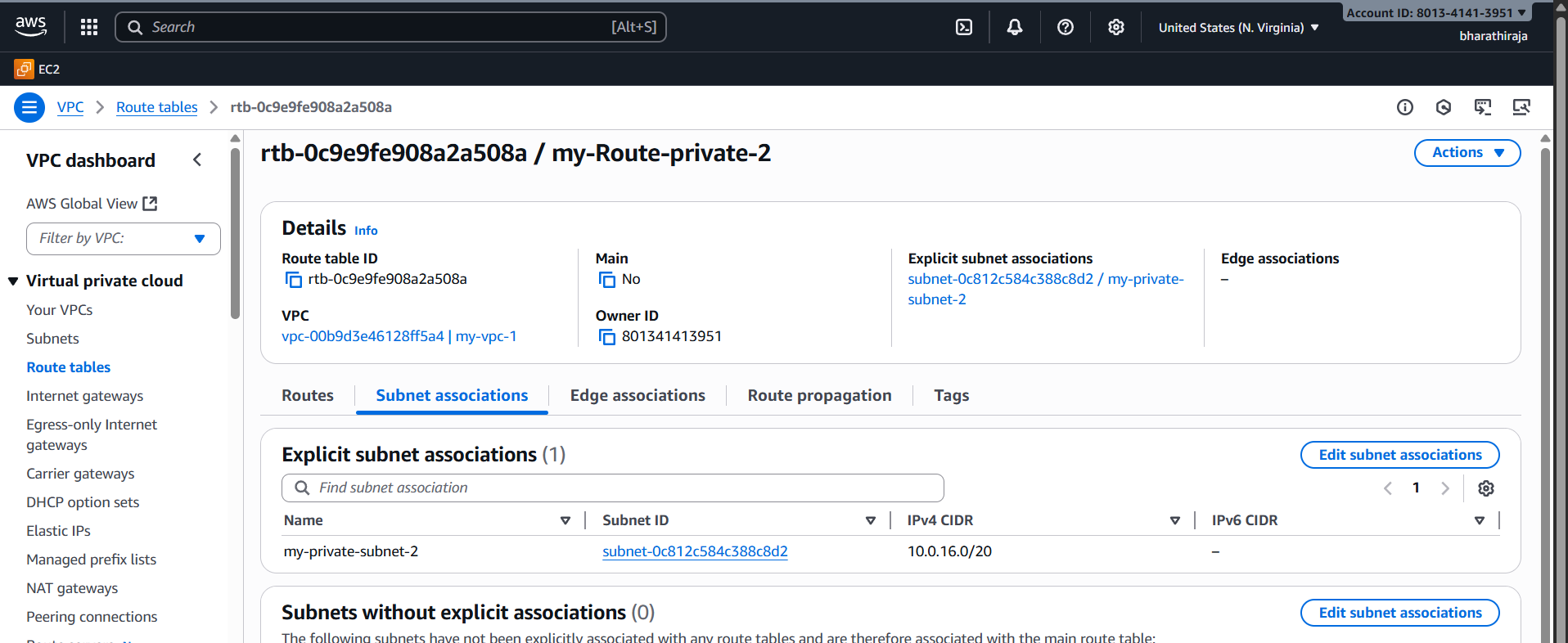
This gives your NAT Gateway a **public IP** (required to access the internet)



In the private route table, I attached the NAT gateway



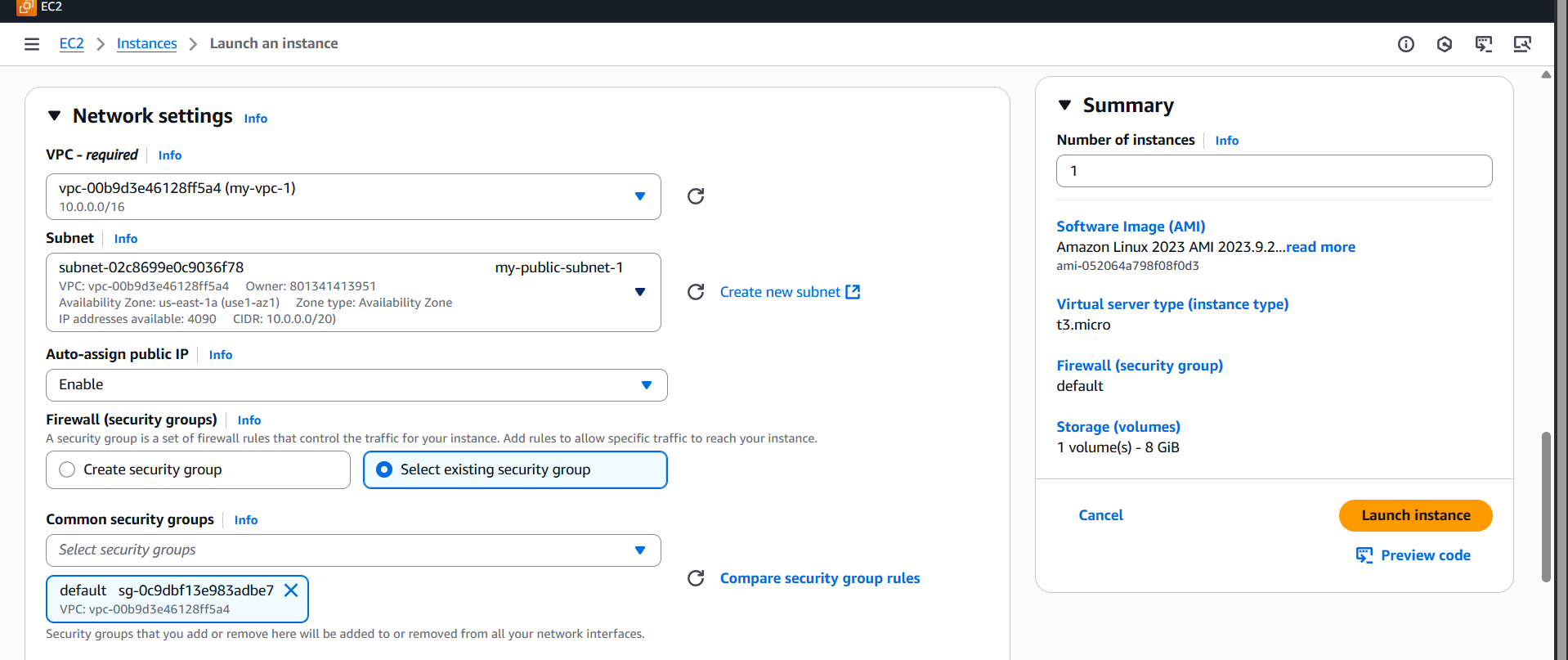
Verify subnet association



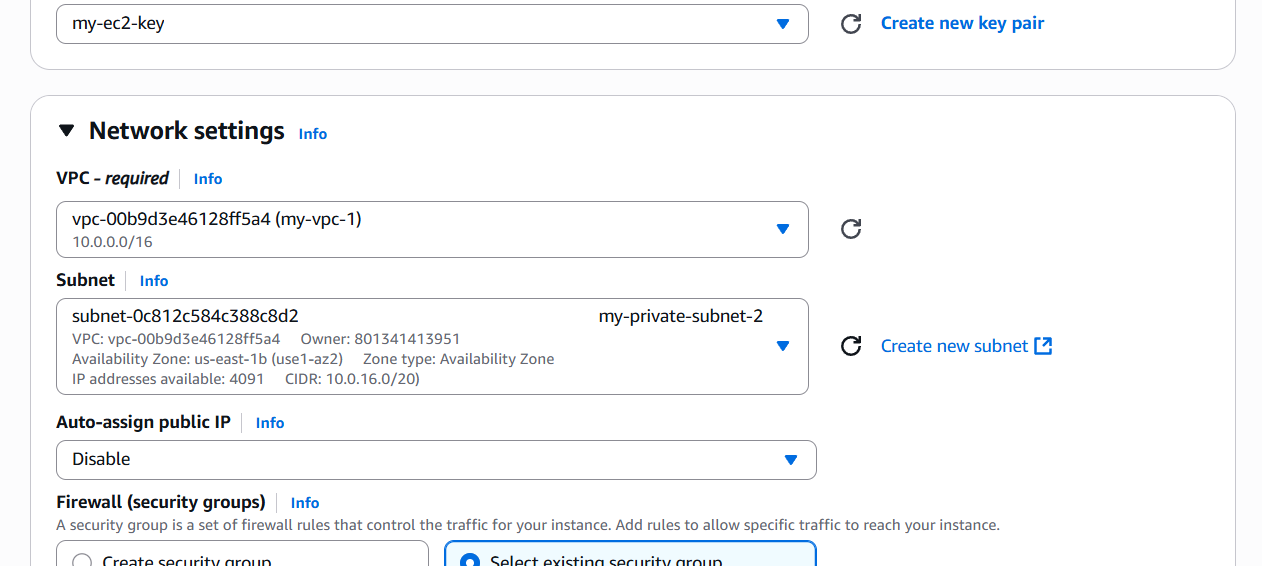
1. Create two instances, one in the public subnet and one in the private subnet.

This is a public EC2 instance

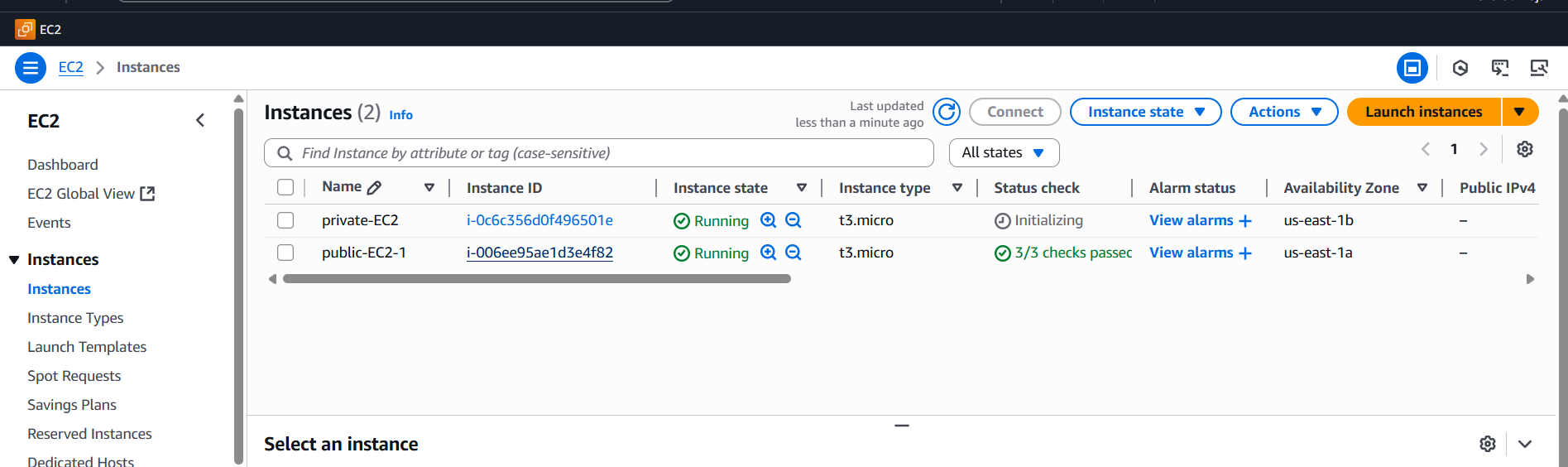
While creating a public subnet, we have to ensure the network configuration



While creating the private subnet, carefully add the network configuration

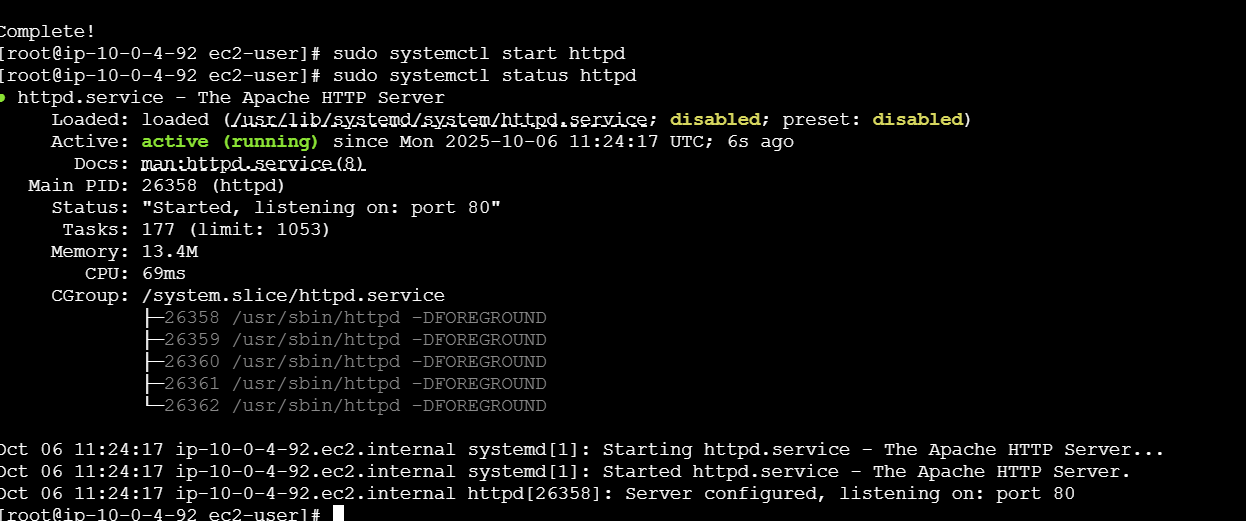


I created 2 EC2 instances public one and a private one ec2 instance



1. Deploy an Apache server on both EC2 instances with a sample index.html file.

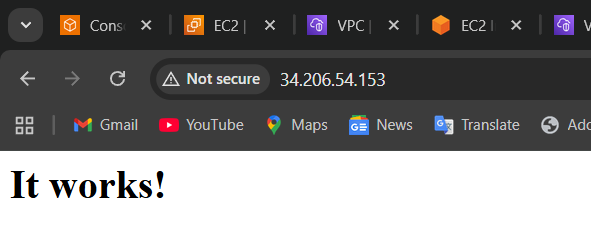
I use this command



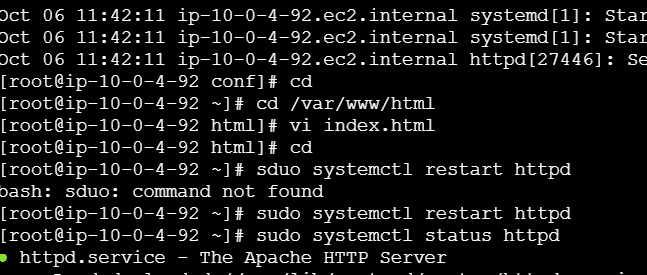
Port changing area

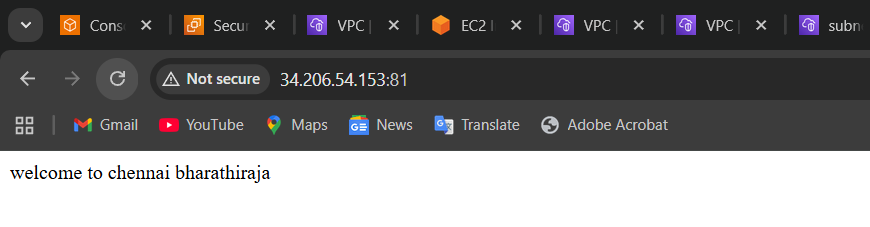


Its works



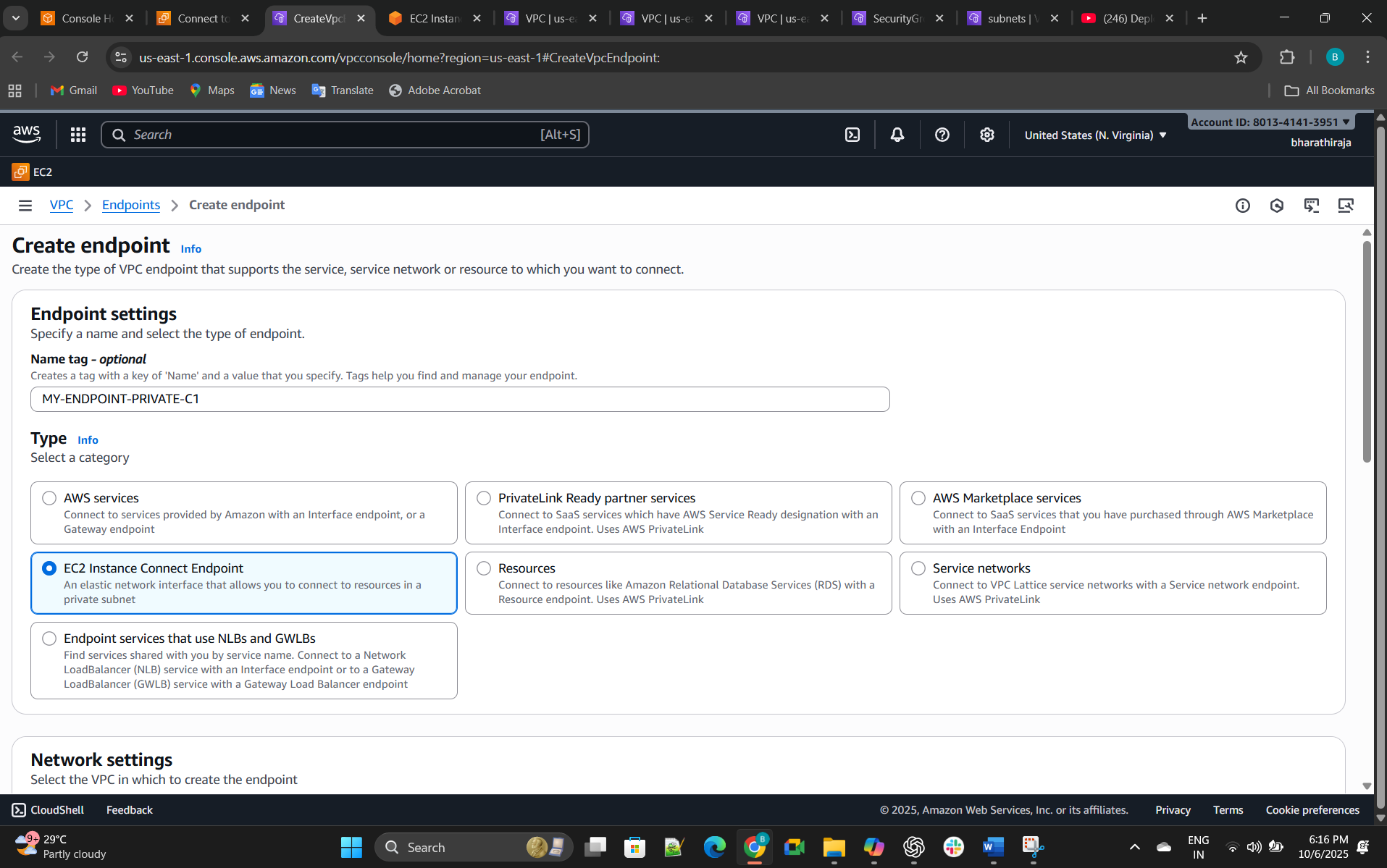
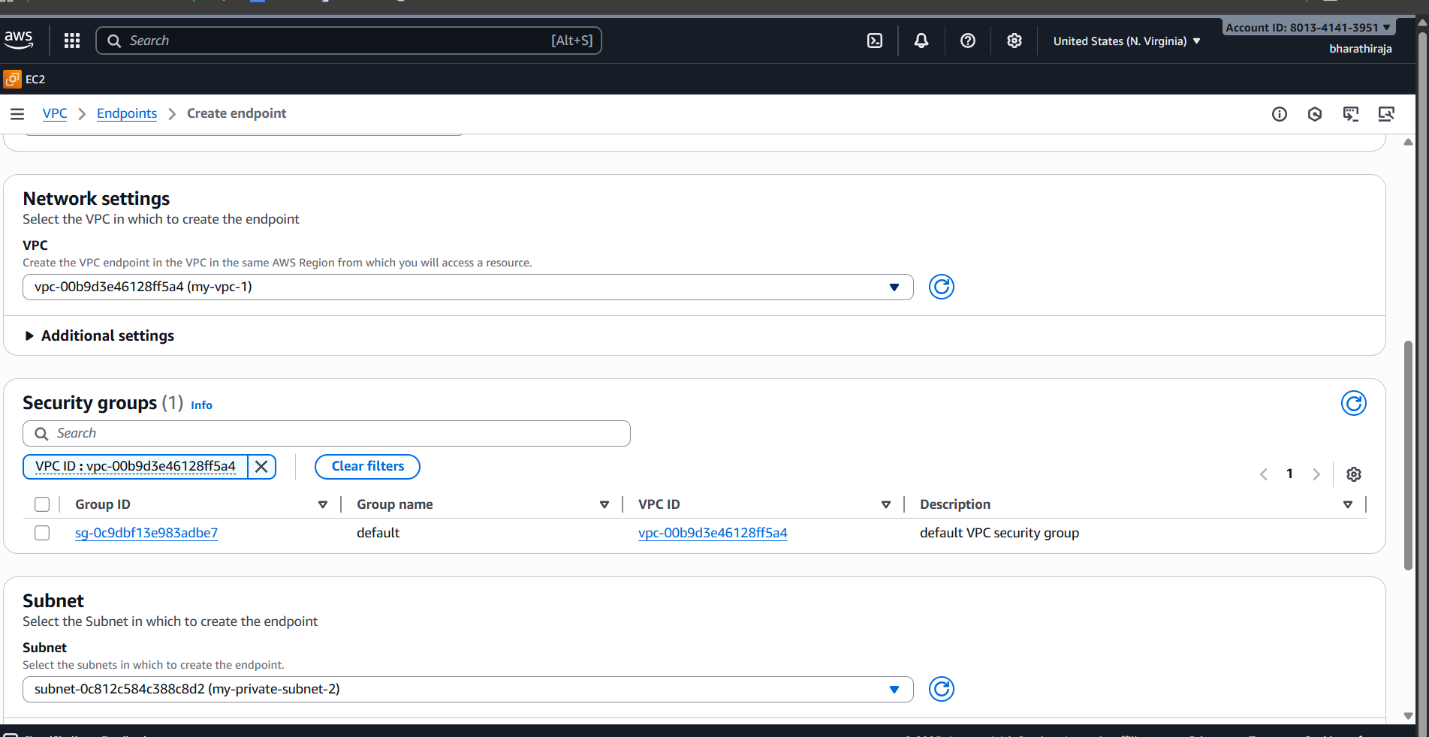
Name change area





First, we need to connect the private instance

Go to the endpoint, create an endpoint it will ask for some details give it carefully



**Step-by-Step Fix**

**Step 1: Check Subnet Route Table**

Go to:

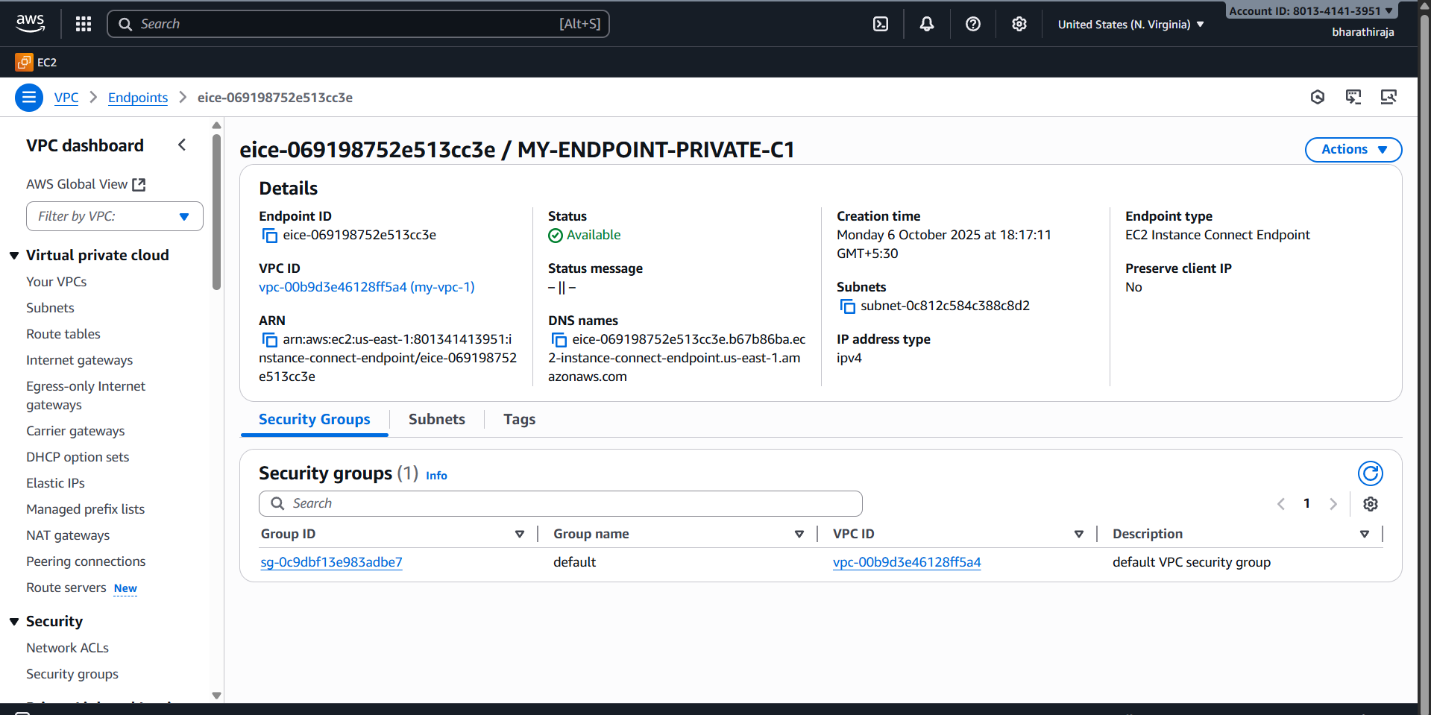
VPC → Route Tables → <associated with subnet-0c812c584c388c8d2>

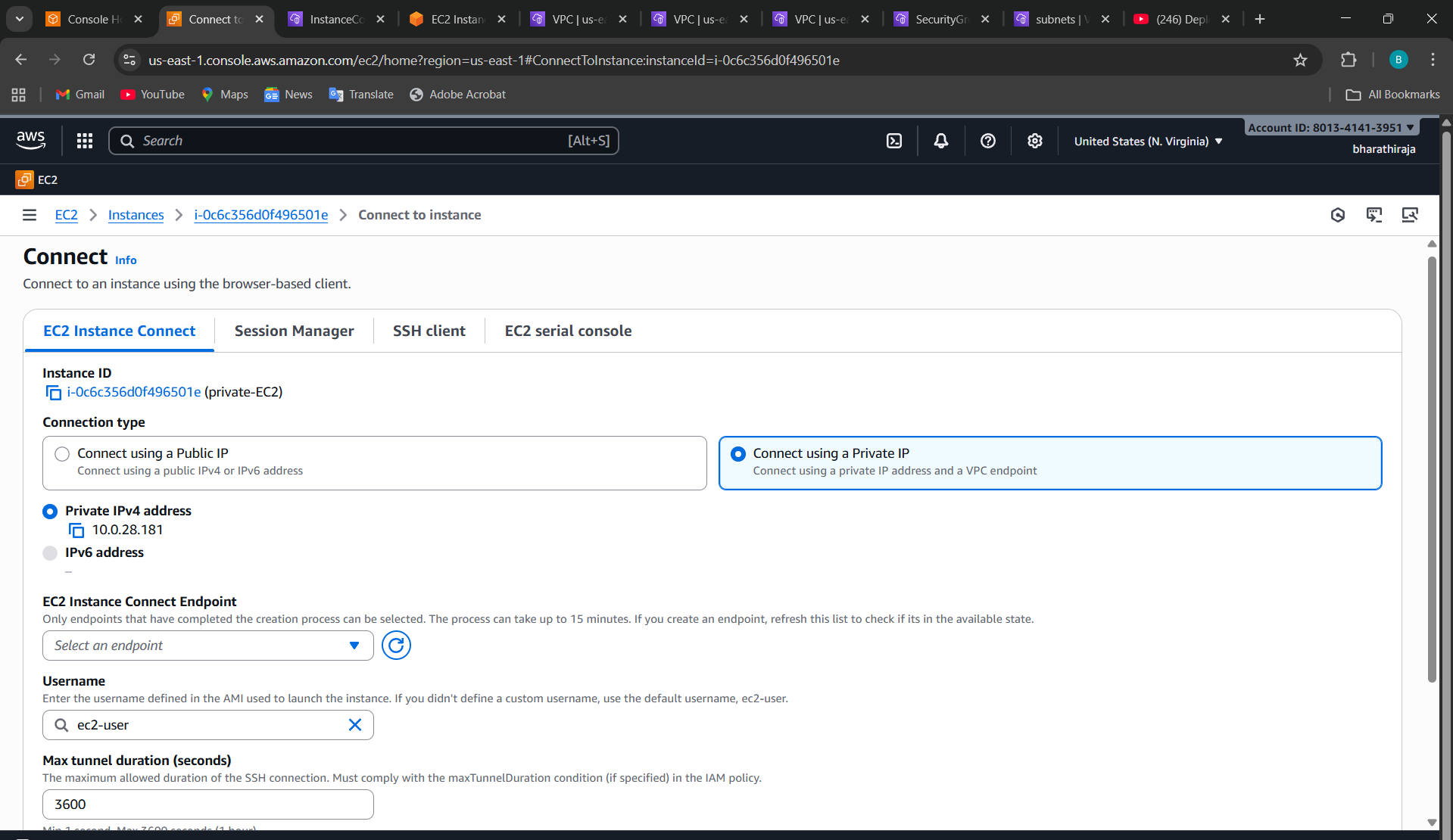
Make sure it has:

Destination Target

0.0.0.0/0 nat-xxxxxxxx (NAT Gateway ID)

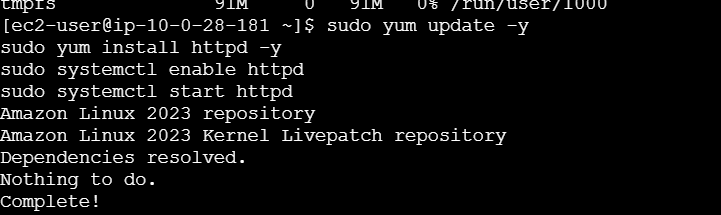
If it points to “local” only → that’s why it’s stuck.



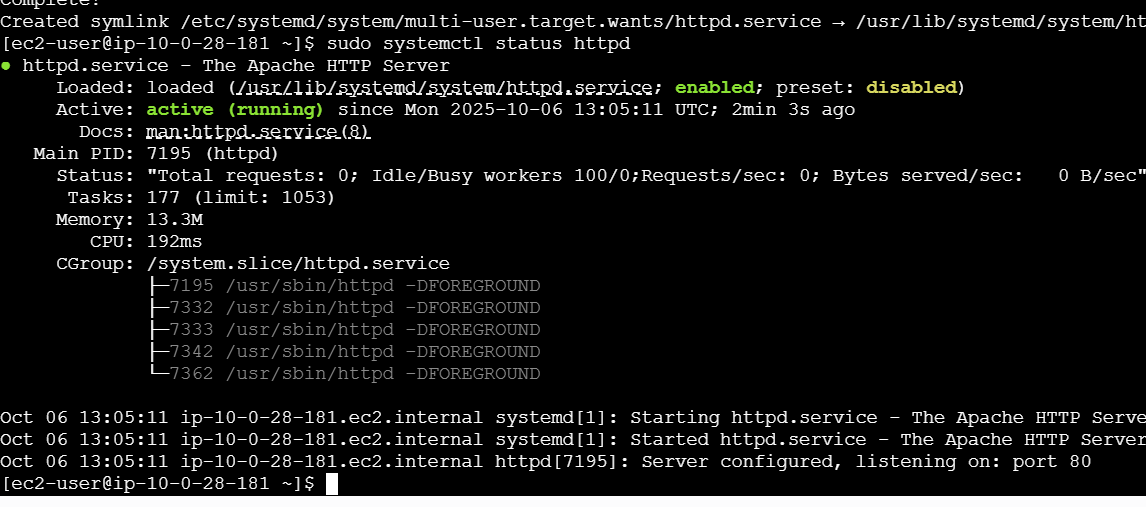


Finally I connected

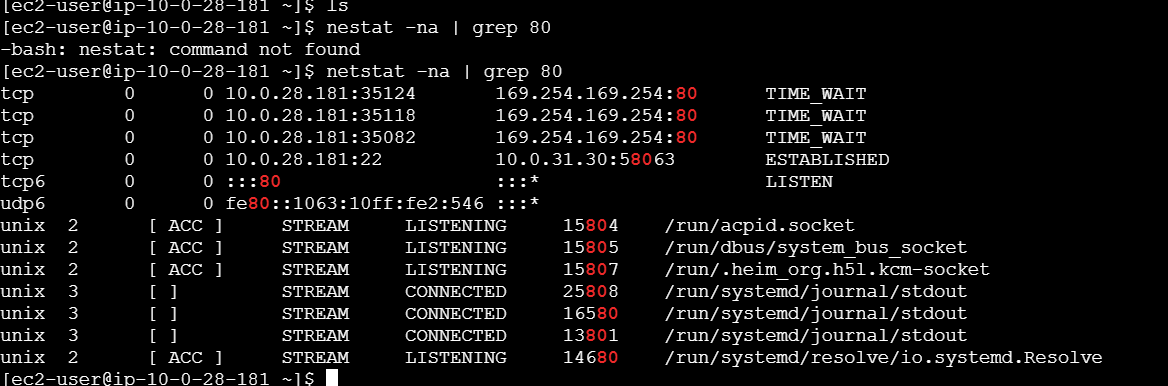
Now I'm installing Apache HTTPD  
Use this command



Now I am checking the status whether it installed or not

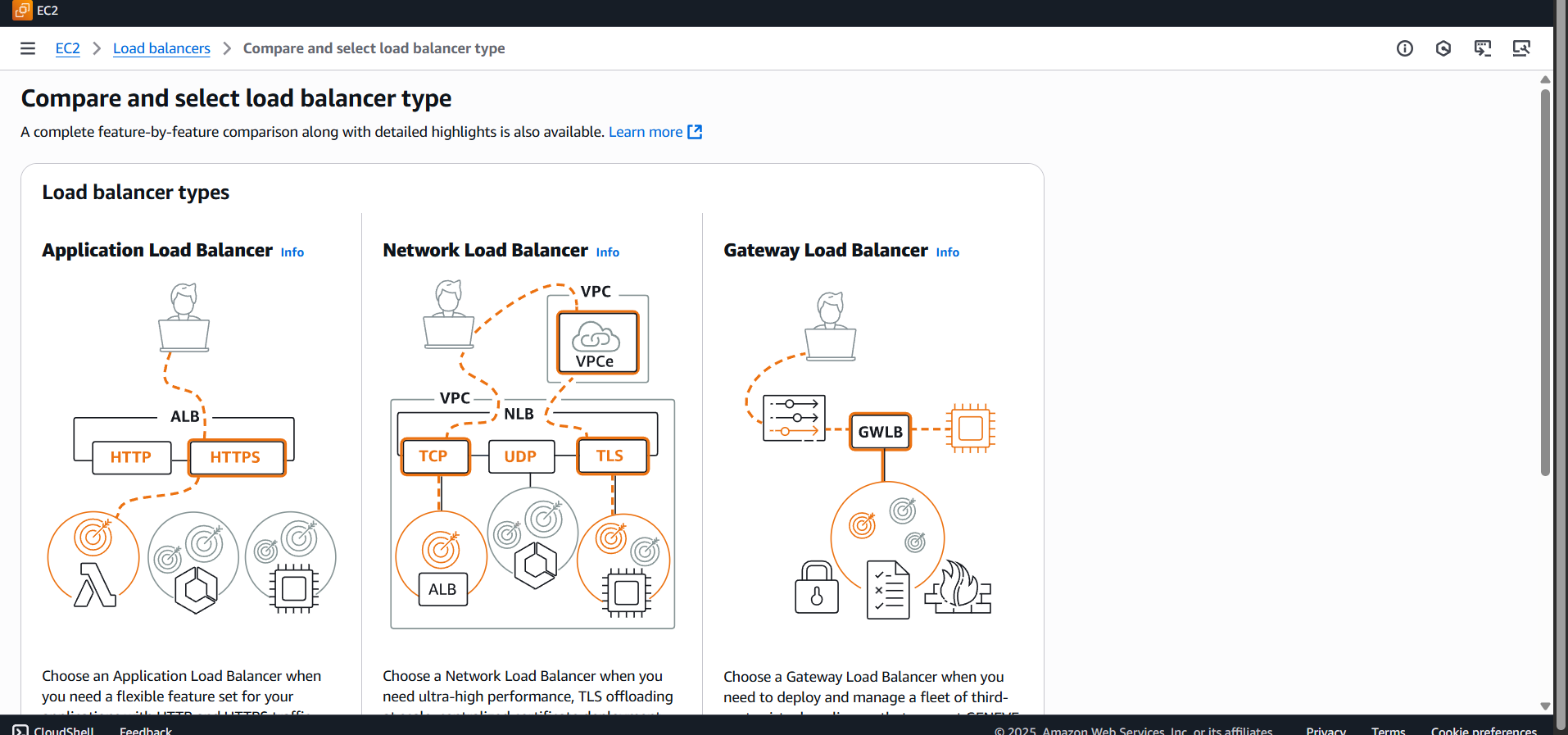


Now it's running

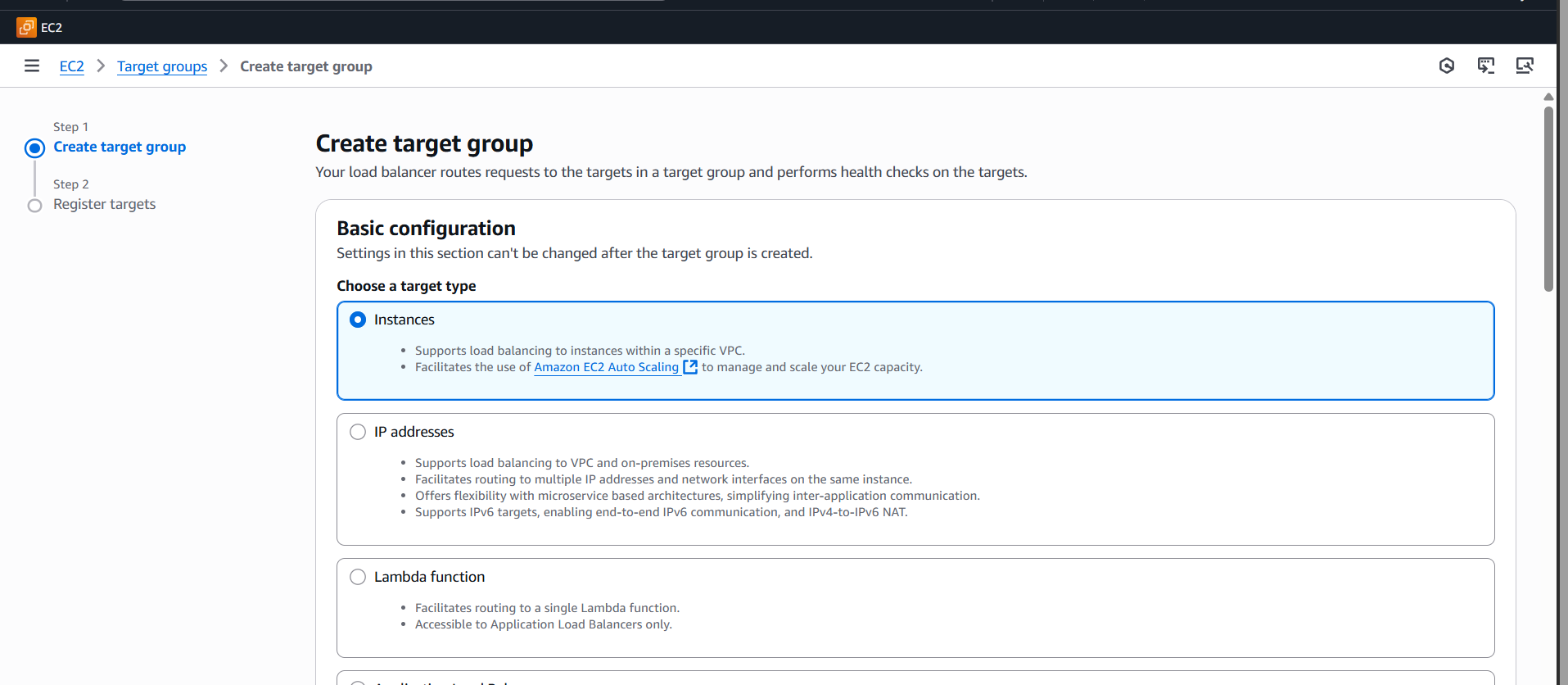


1. Create one application load balancer and attach it to both EC2 instances.

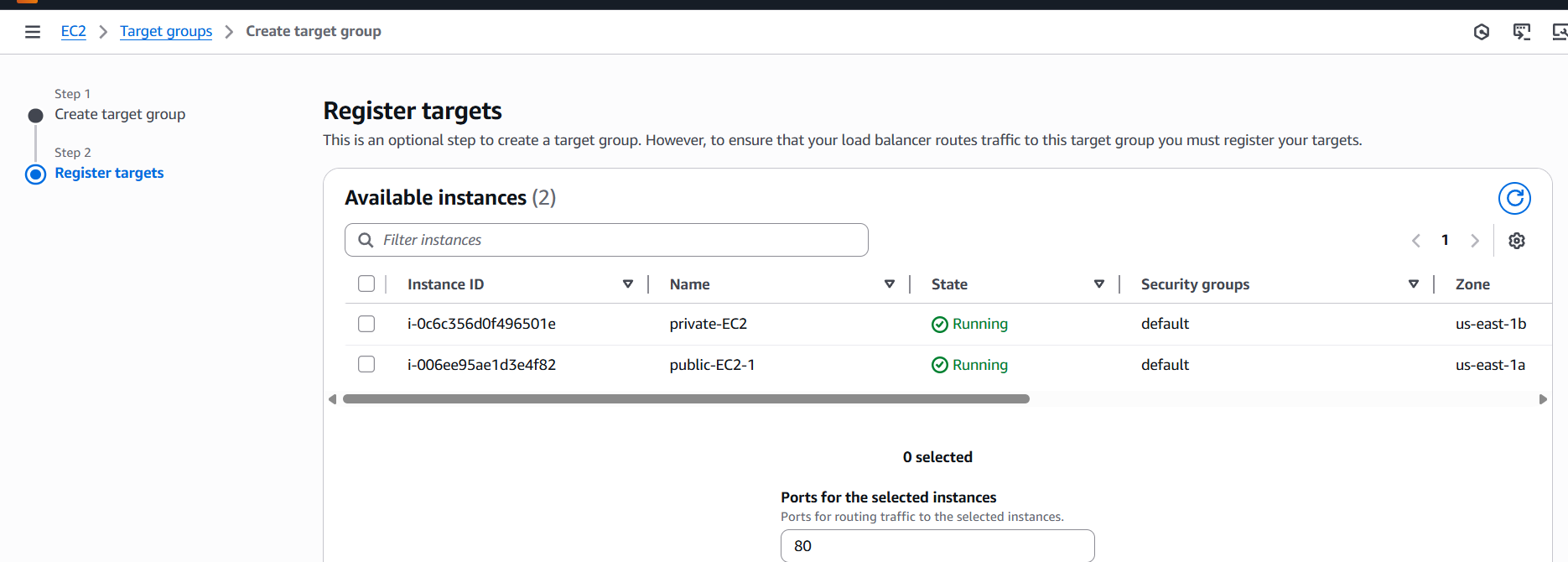
go to EC2 instances, and there is a load balancer, click on it



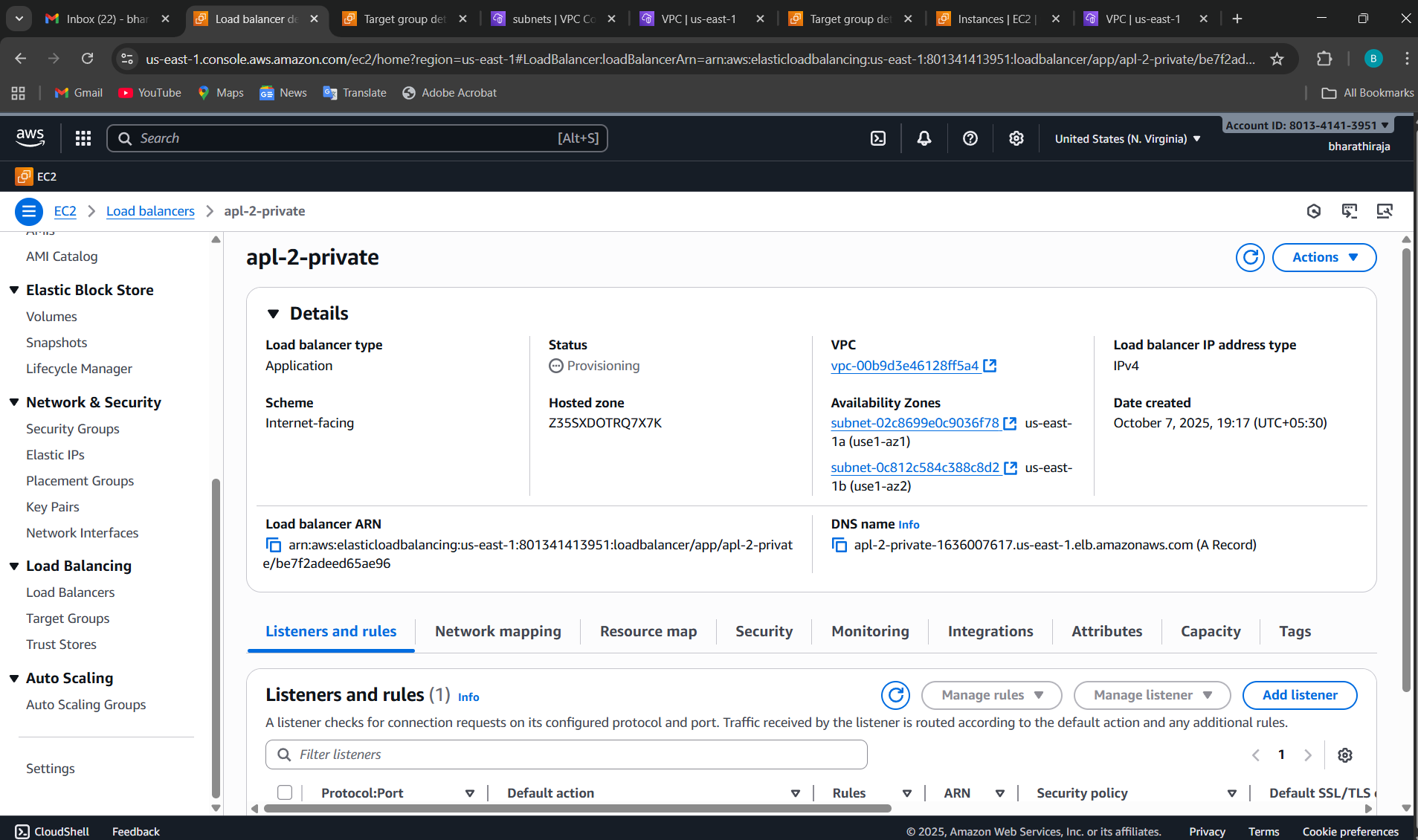
Before we create one target group

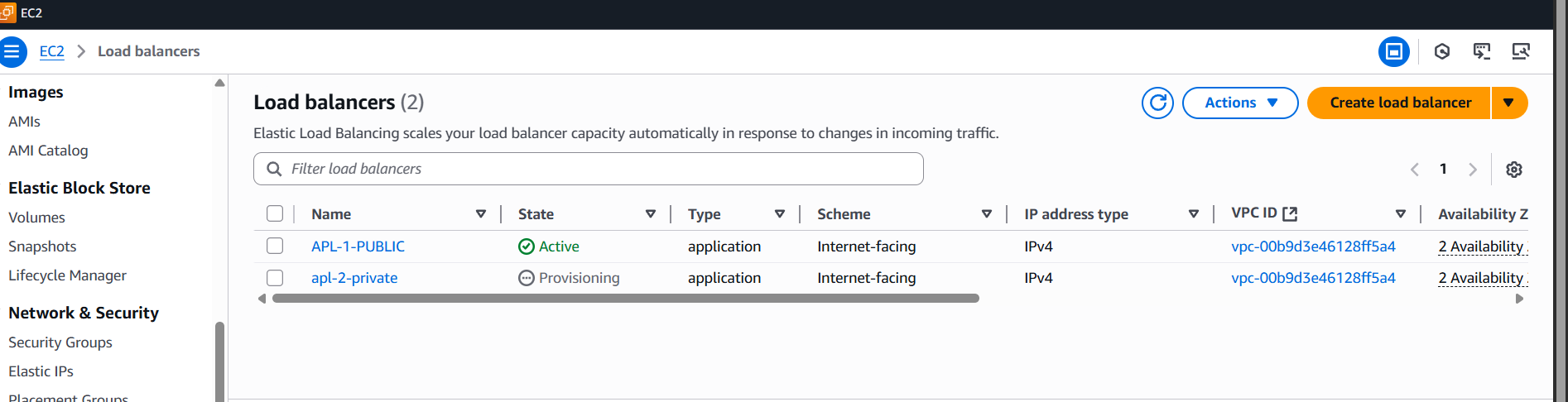


Created default target group\_1



2nd application load balancer





1. Store application load balancer logs in S3.

Go to **AWS Console → S3 → Create Bucket**.

Give the bucket a **unique name**, e.g., alb-logs-bharathiraja.

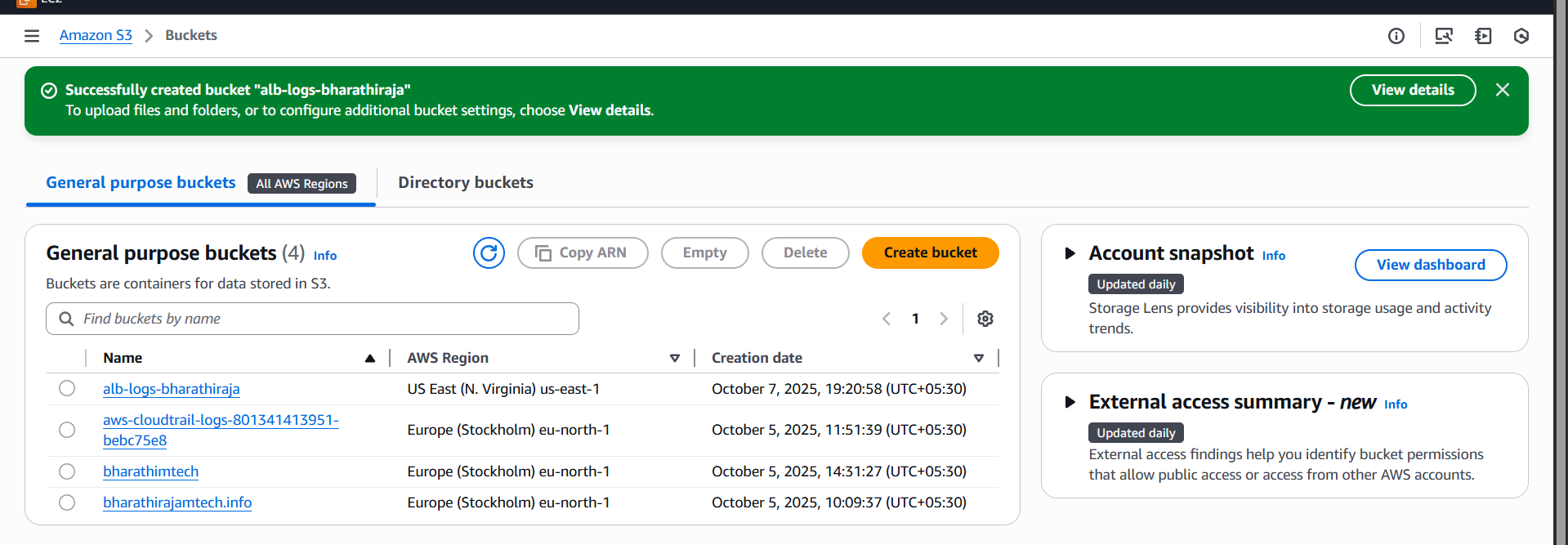
Choose **Your Region (the** same as your ALB is recommended).

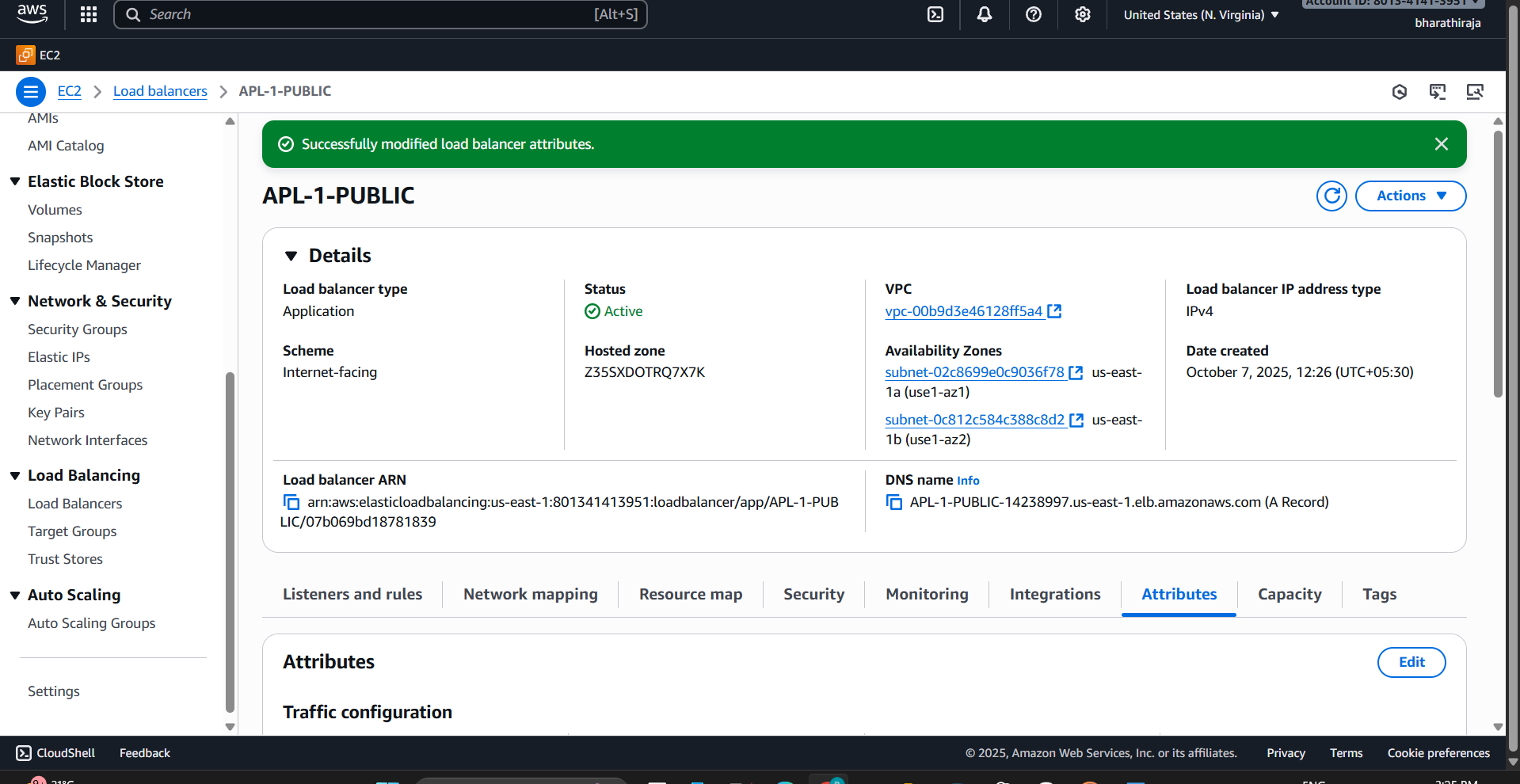
Disable **Block all public access** (logs don’t need to be public).

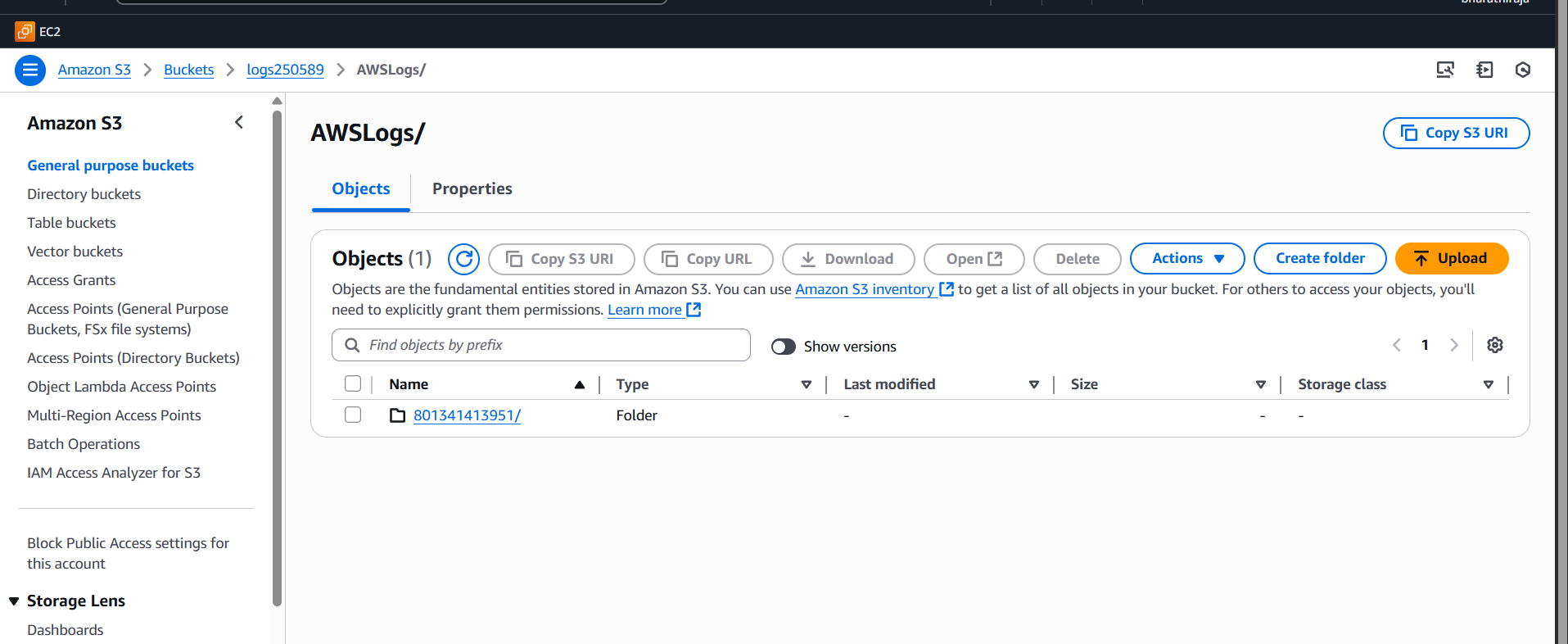
Click **Create bucket**.

Go to EC2 Dashboard → Load Balancers → Select your ALB  
1. Description tab → “Edit attributes”  
2. Enable Access logs

I created one s3 bucket





Logs happening 

1. Store the VPC flow logs in a CloudWatch log group.

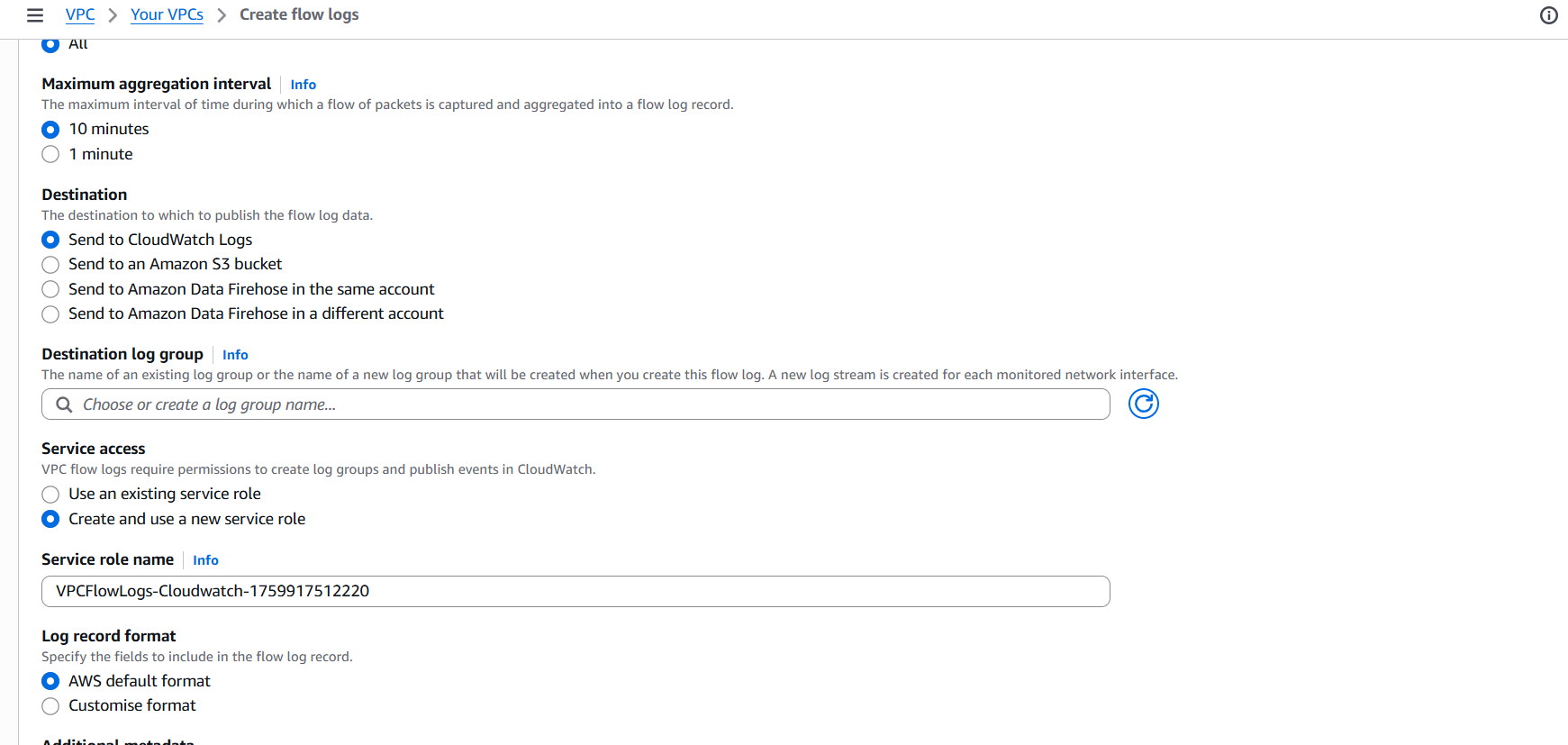
**Go to VPC in AWS Console**

1. Sign in to AWS Management Console.
2. Go to **VPC → Your VPCs**.
3. Select the **VPC** for which you want to enable flow logs.

Create a flow log



If already give exist if not, you can create new one

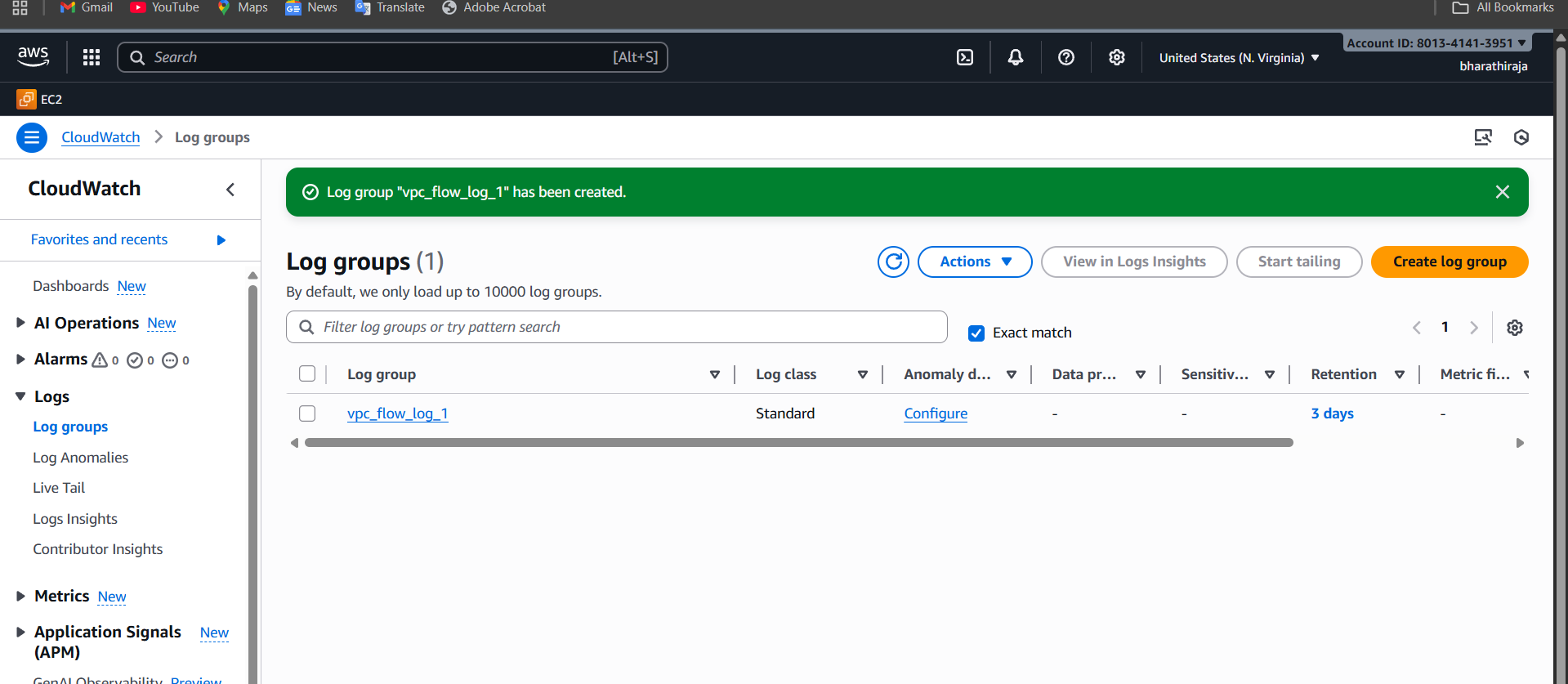


Open one cloud group logs

Go to **CloudWatch → Logs → Log groups → Create log group**

Enter a name, e.g., VPCFlowLogs-MyVPC.

Click **Create**.



I created flow logs

