**Task on Load Balancer**

1. **title: Configure Classic Load balancer**

**objective:** we need to create one load balancer in classic type

**prerequisites:**

ec2

vpc

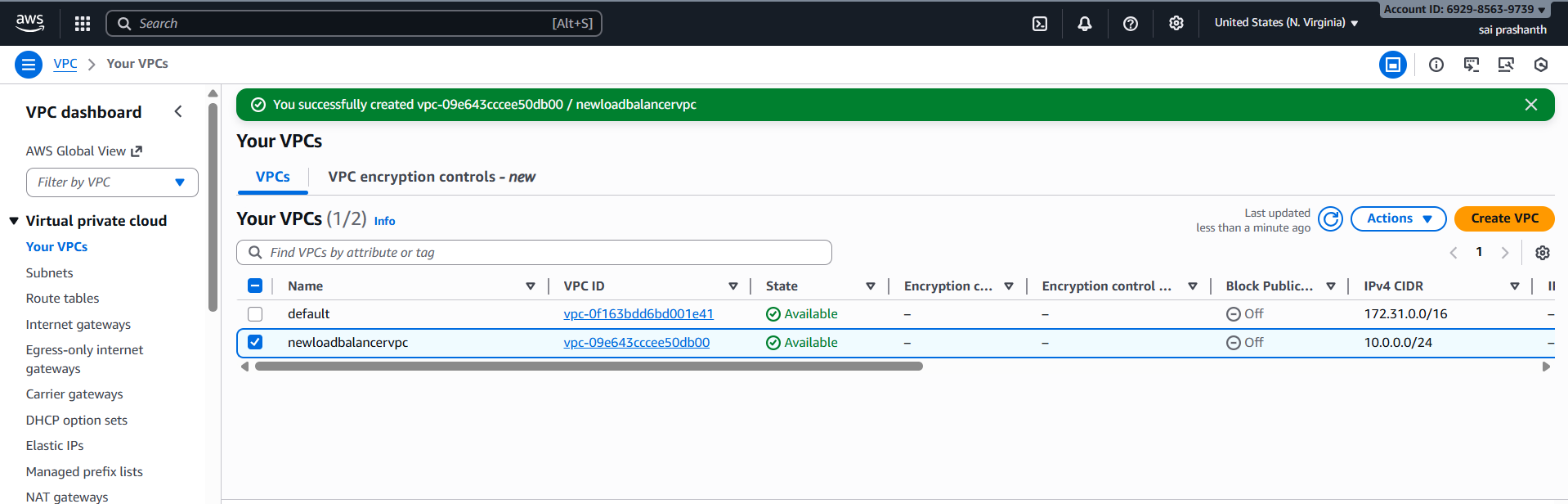
subnets

security group

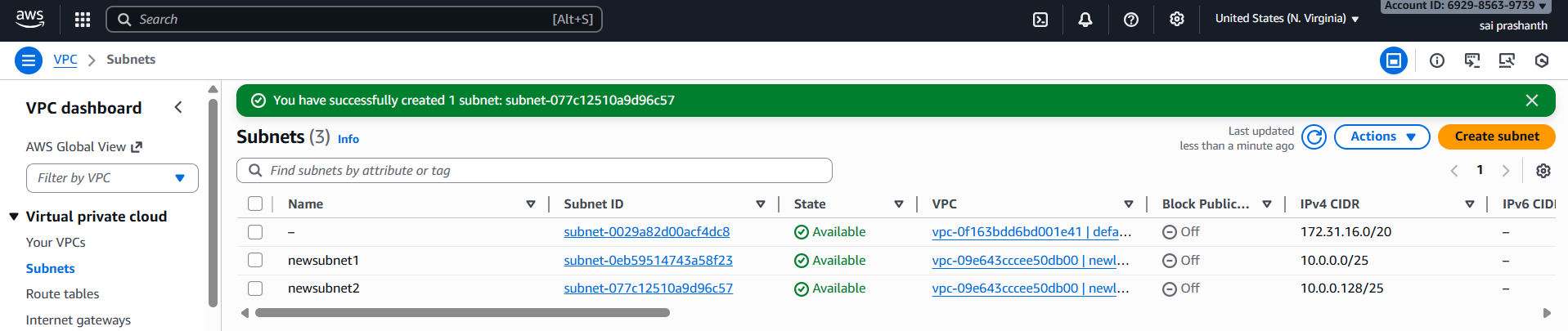
**step by step process:**

first create one vpc

vpc—create vpc—enter one name and cidr range and create

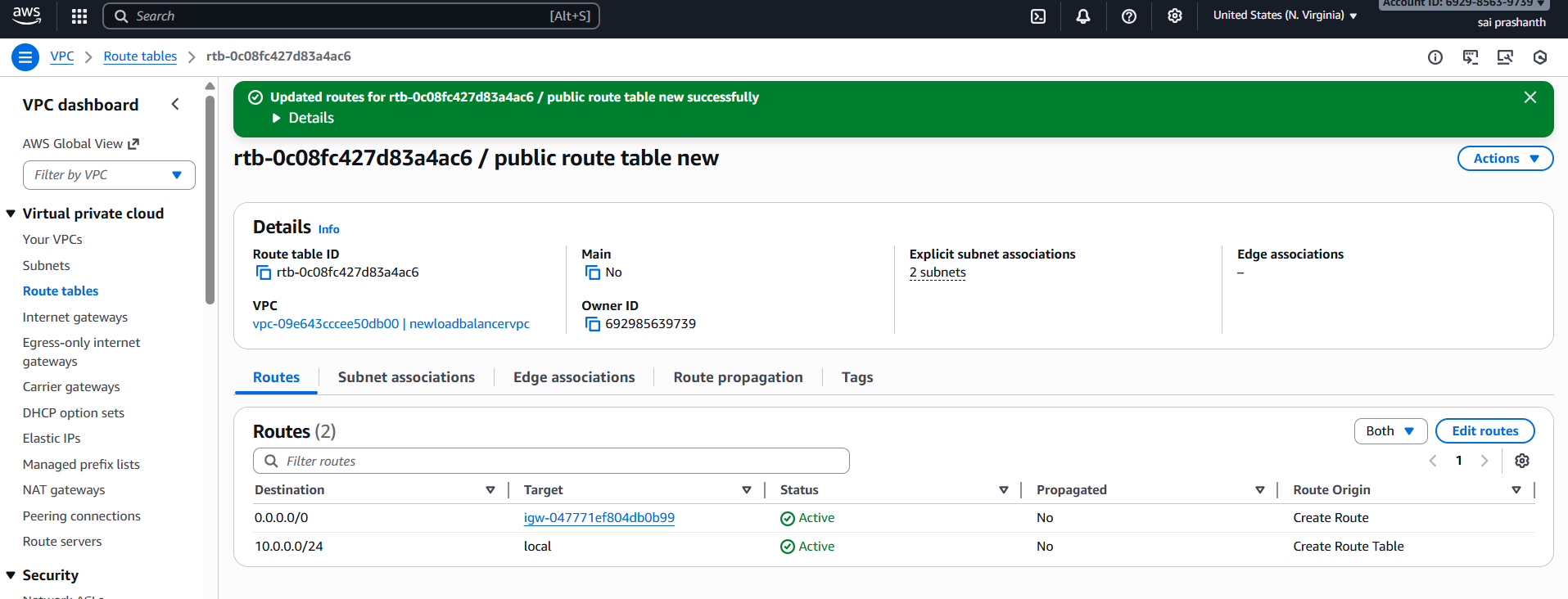


now we need to create subnets with attached vpc

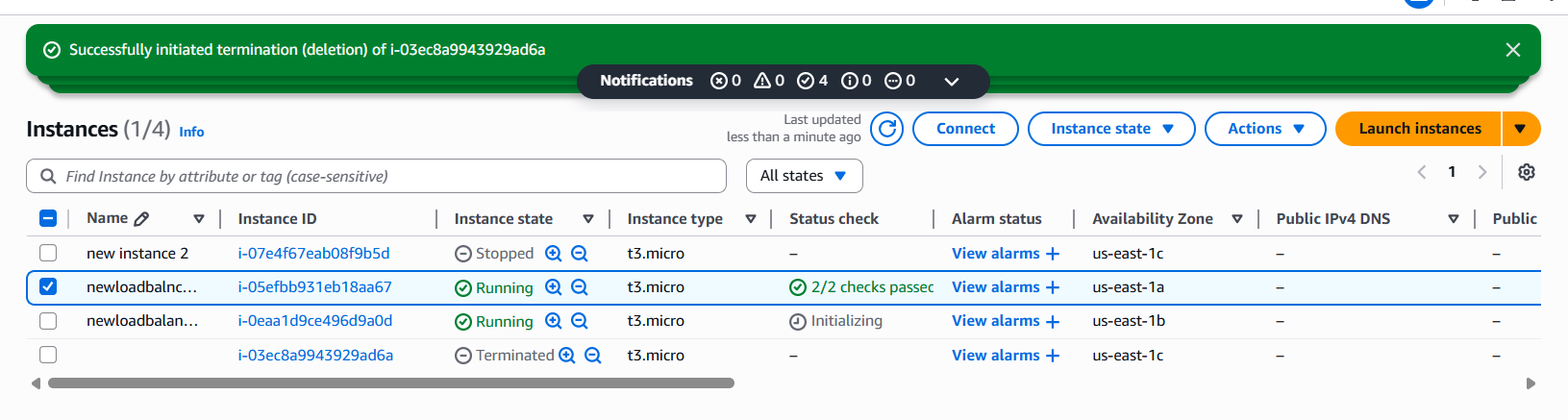


now created route tables and attached 2 subnets and gave access to internet gateway

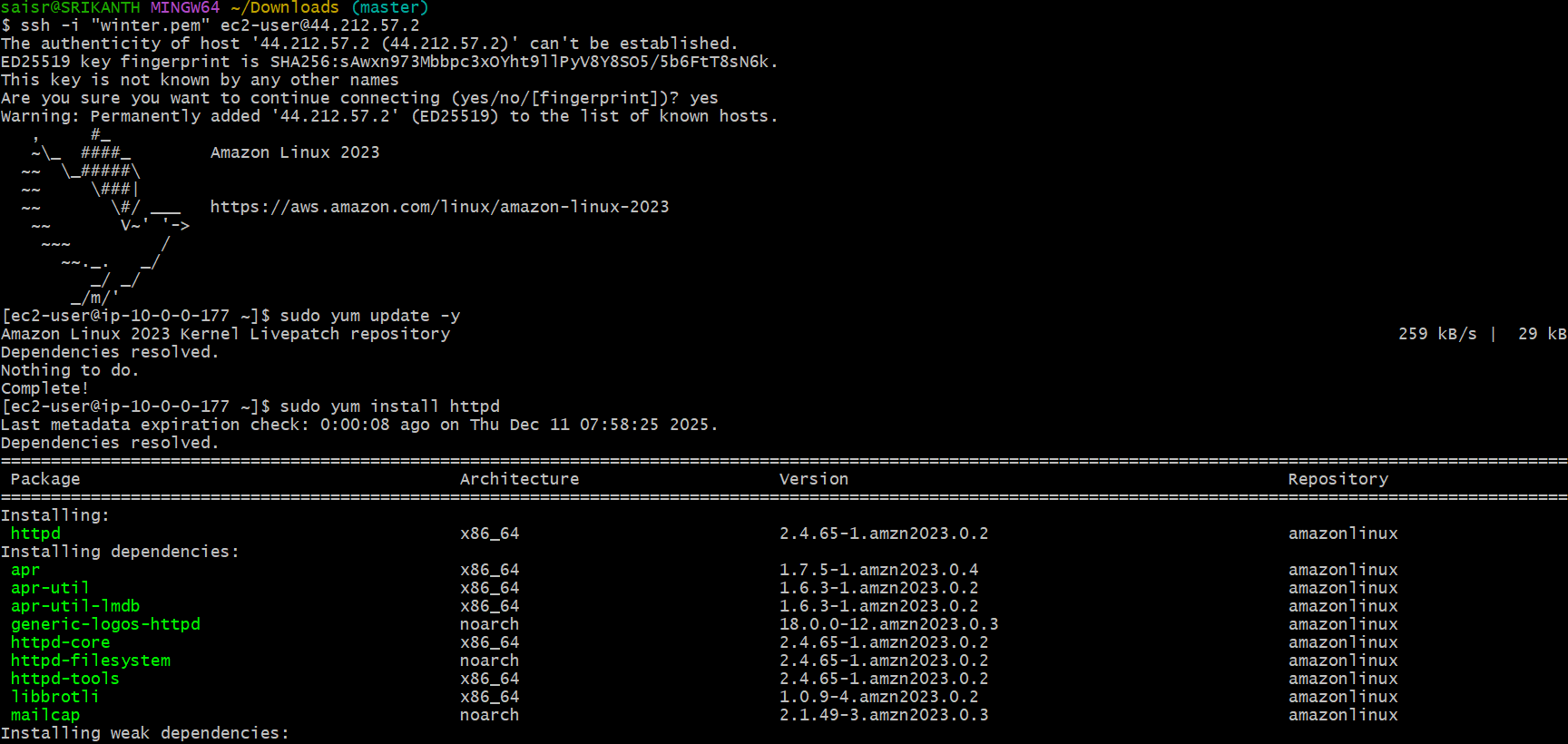
now route table has been created

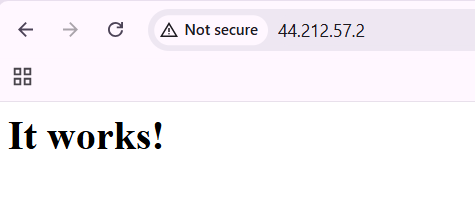


now we have created 2 instances with two different subnets as it is in different regions



after creating connect instance and run one web server apache or nginx in both instances and check

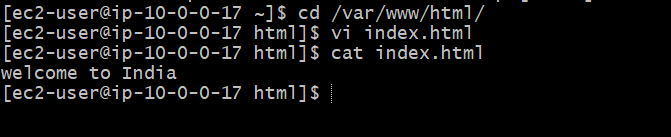




enable first you have created already instances with index.html for both instances

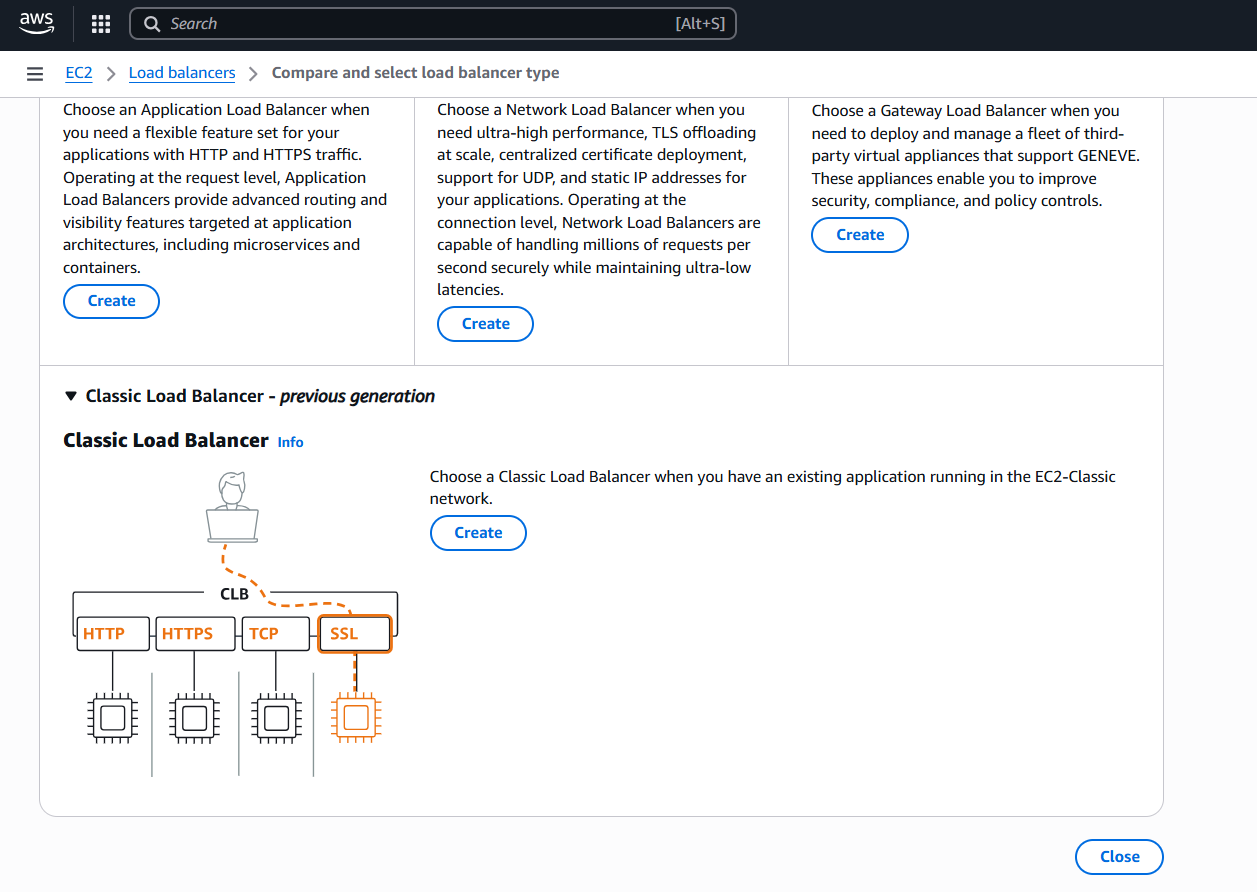
I created one instance index as welcome to India

another instance index as welcome to Tirupathi



now create classic load balancer

go to ec2—load balancer—create load balancer—classic load balancer--create

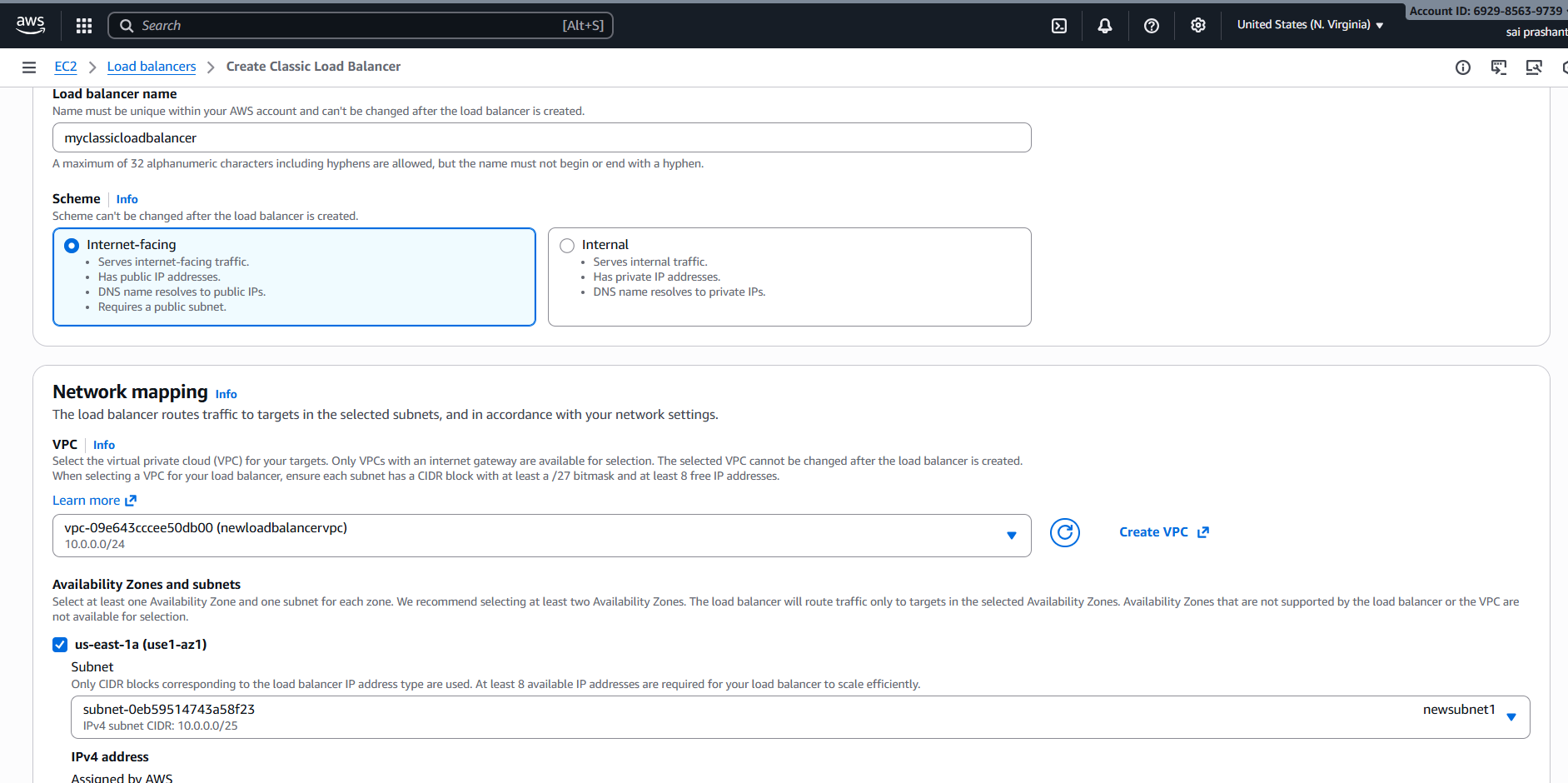


after creating enter load balancer name

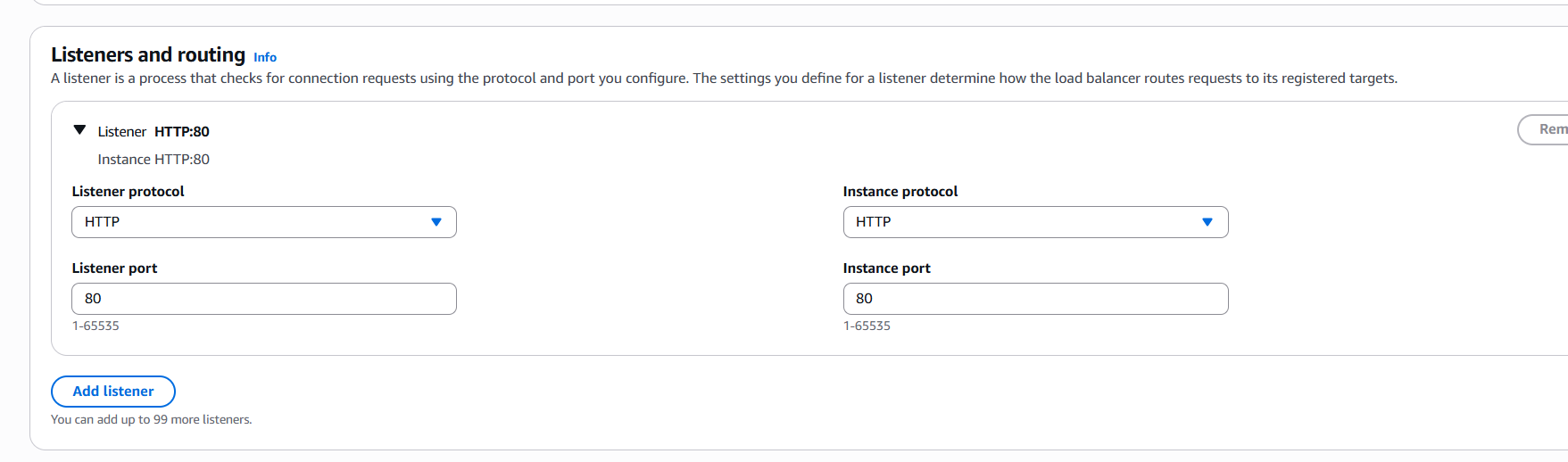
enter scheme internet facing

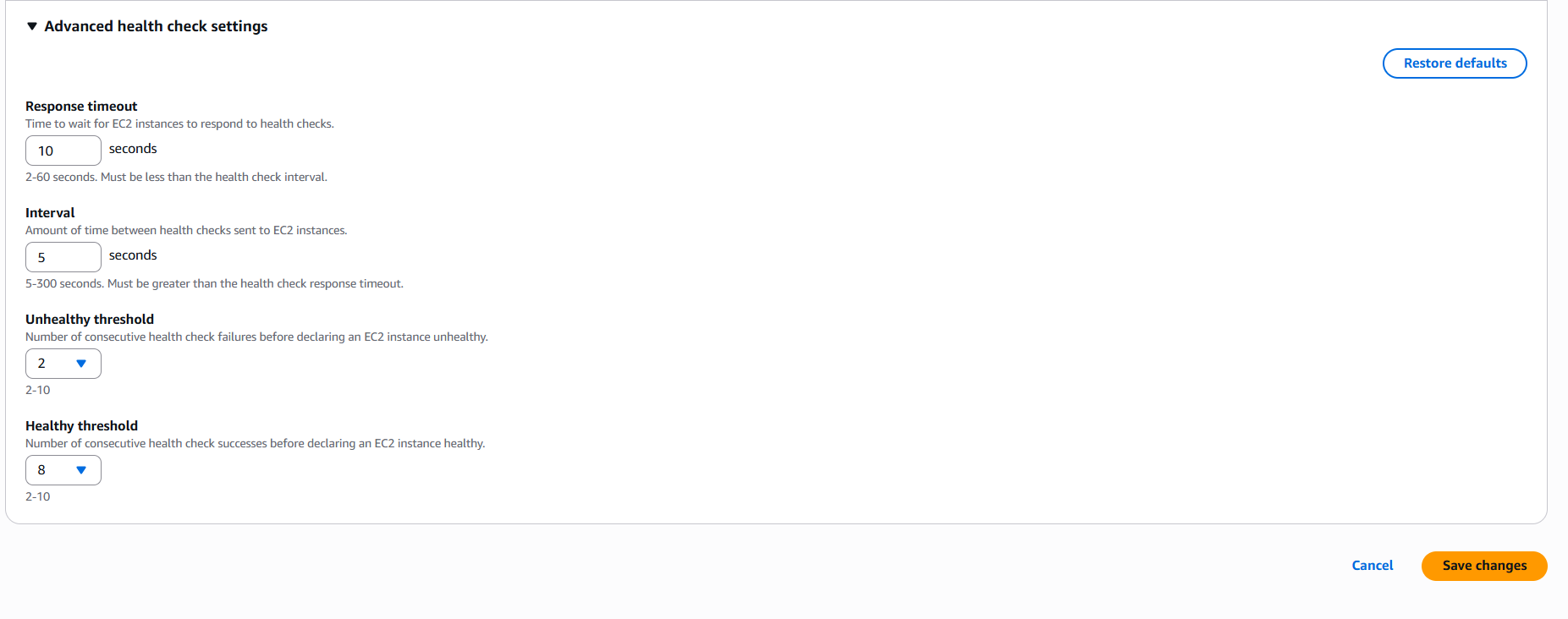
select vpc

subnets in two regions



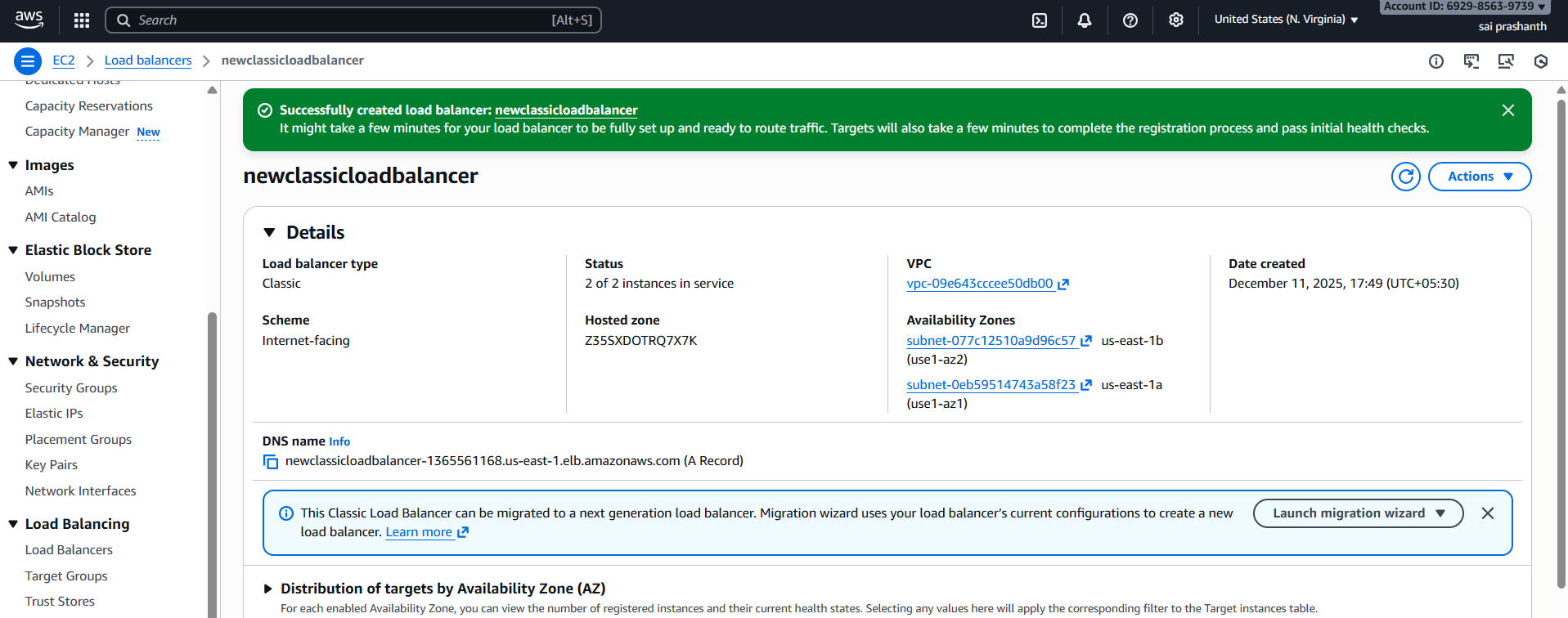
now listener protocol with 80 port number





add both instances

after save changes classic load balancer has been created



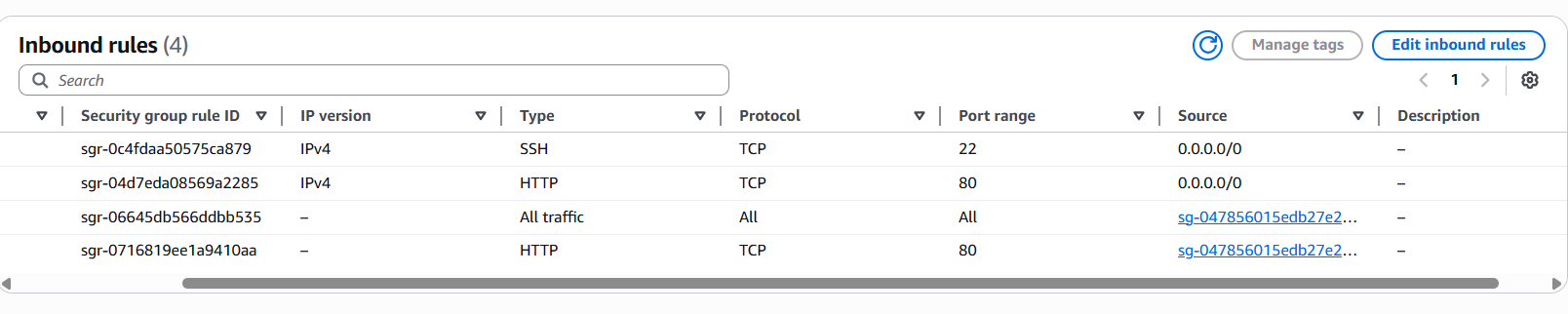
ensure both instances health status should be in service



now in security groups add http with classic load balancer security group

and http with custom 0000

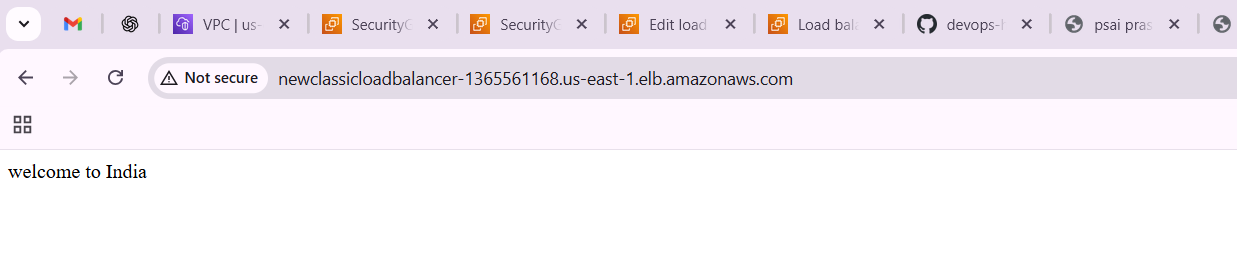
this is compulsory



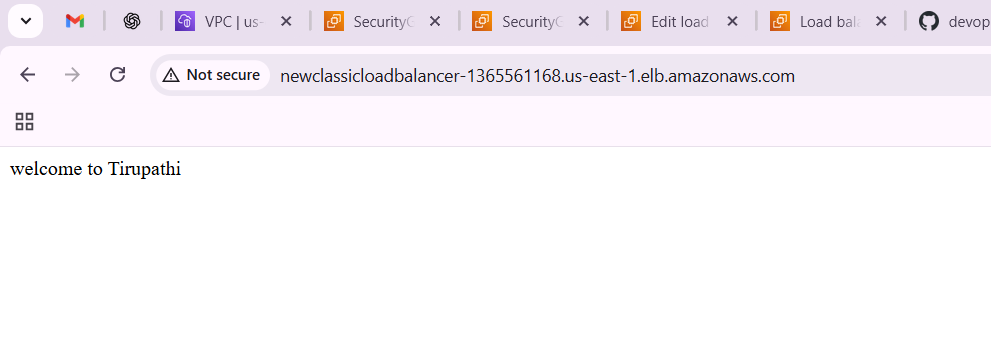
**validation steps:**

now you need to verify the classic load balancer link

for first instance index



for second instance index



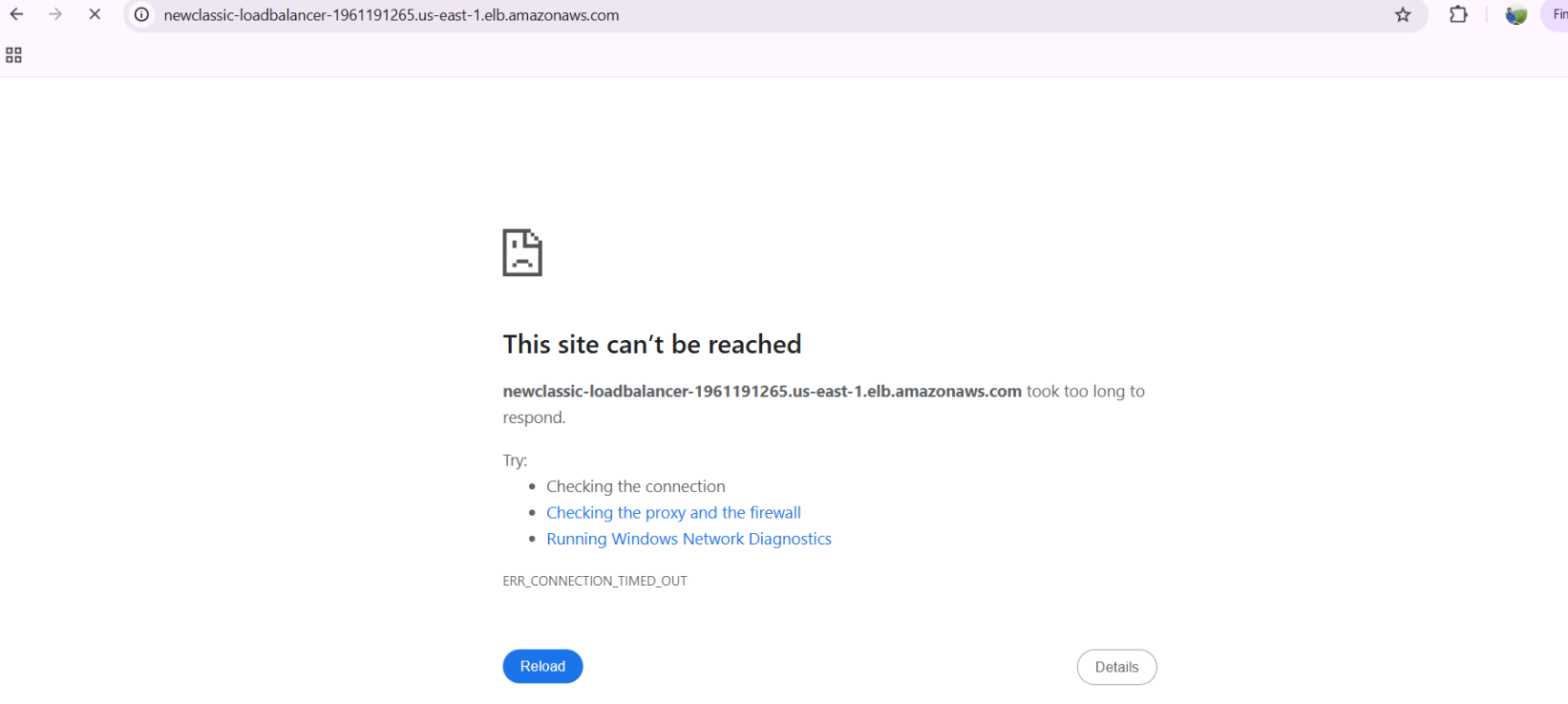
**trouble shooting:**

first, I tried with load balancer URL

then I make changes in security groups

even though faced same issue

again, I deleted existing load balancer and created again and added security groups with additional http with custom 0000



**conclusion**:

now with classic load balancer link we can run both instances as it is healthy.

1. **title: Configure Application Load balancer.**

**objective:** we need to create one application load balancer with the same instances which we have created**.**

**prerequisites:**

vpc

ec2

subnets

route tables

internet gateway

target groups

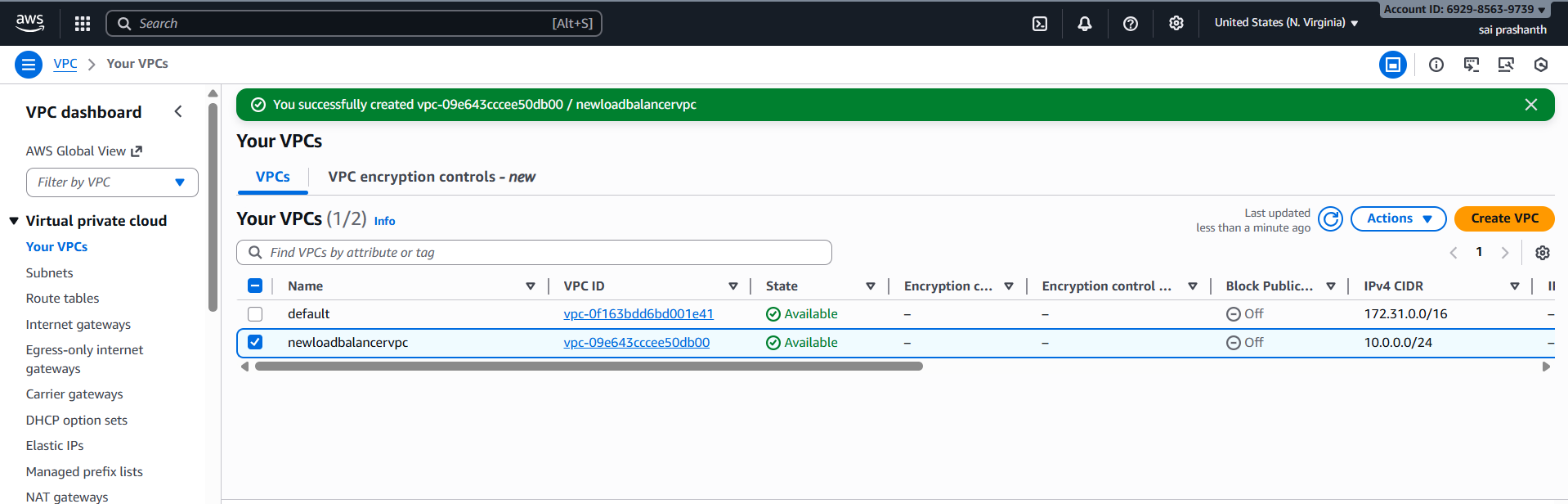
load balancer

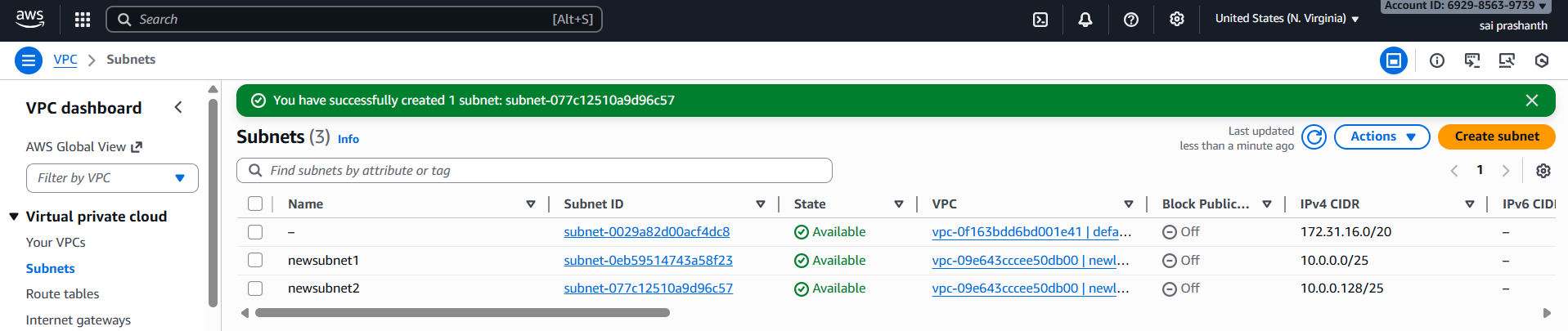
**step by step process:**

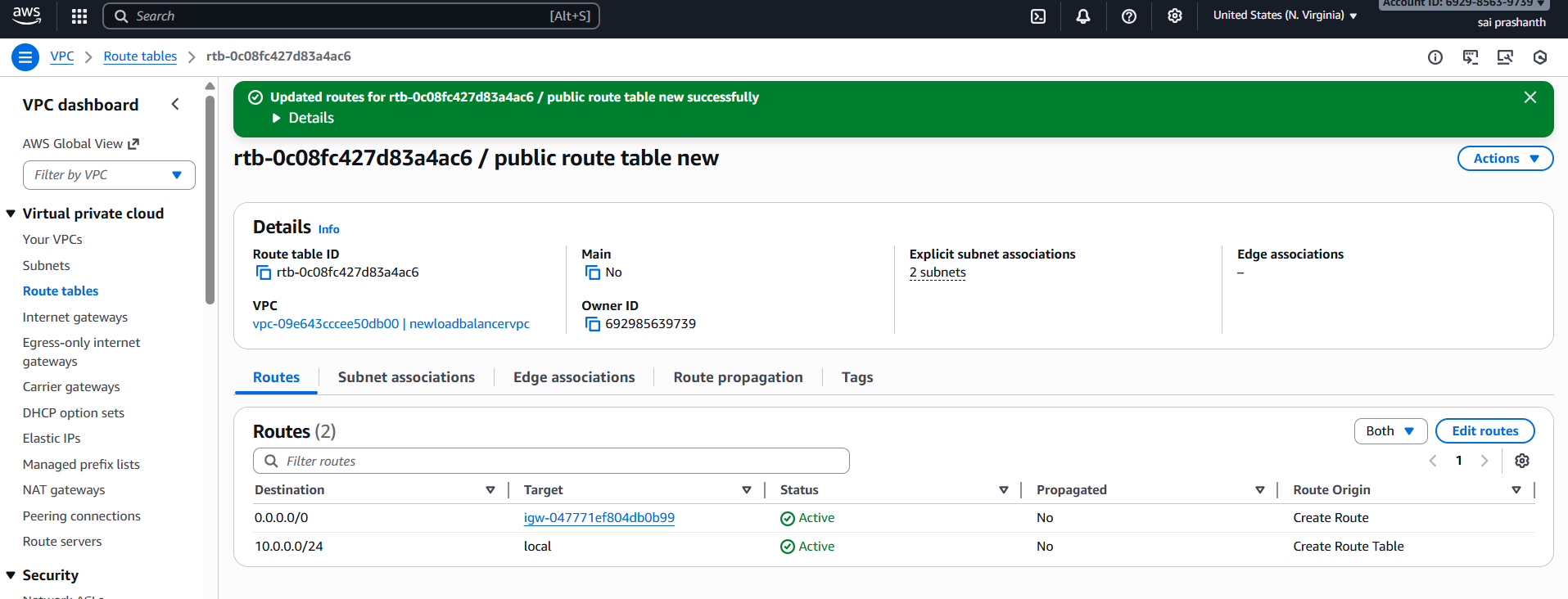
same process first we need to create vpc

2 subnets—attaching to route table public

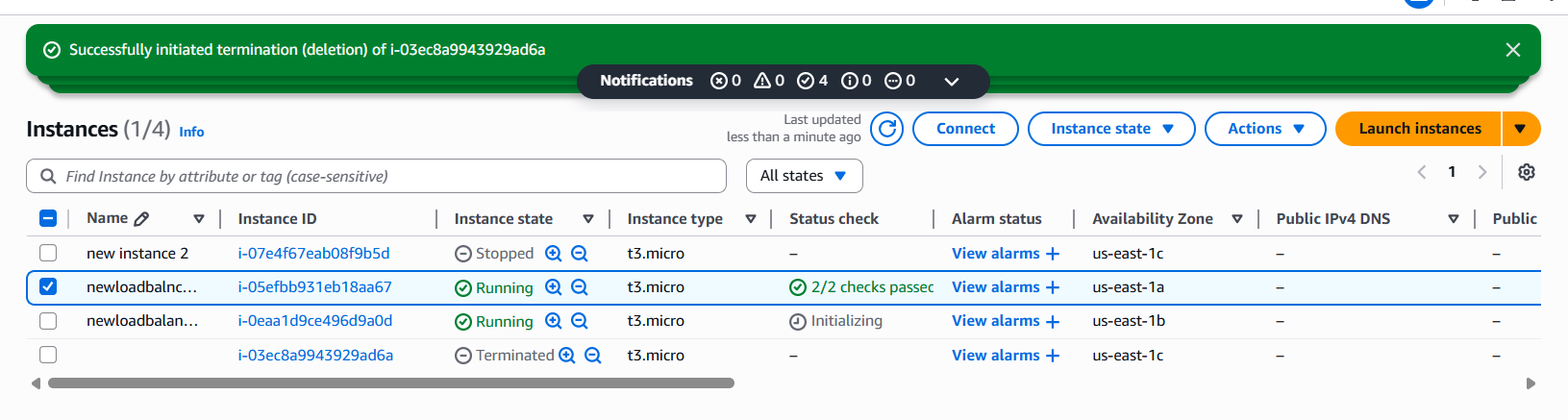
provide internet gateway



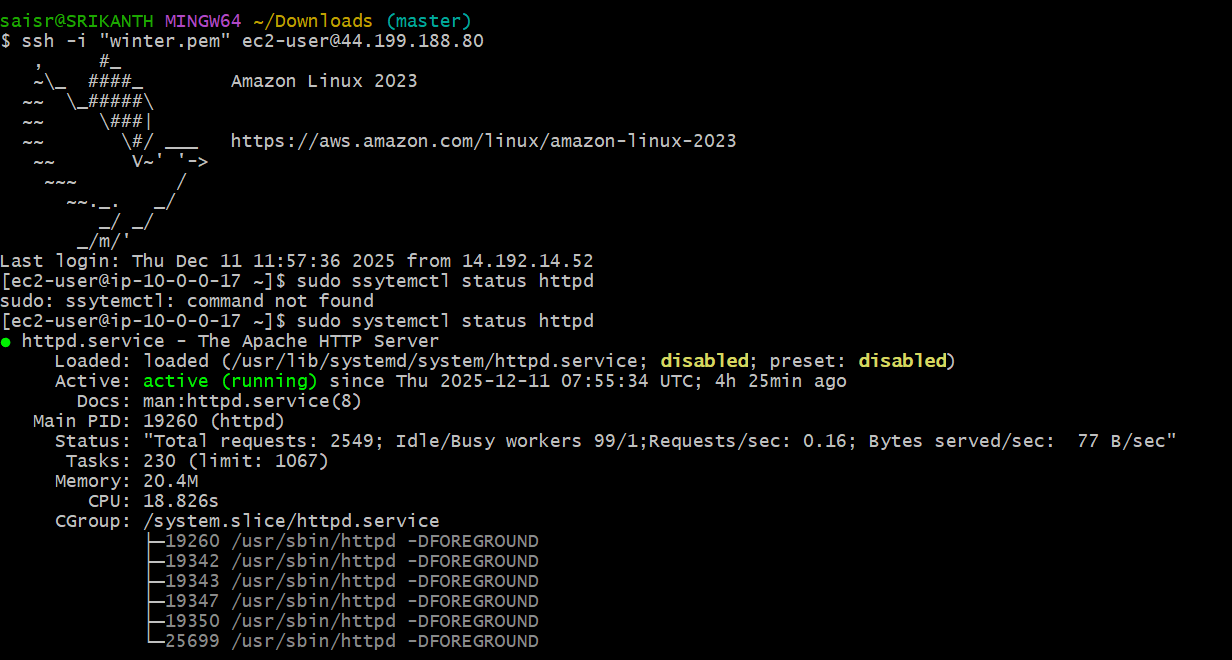




then we need to create two instances and edit the instances one with one subnet region and another instance with another region



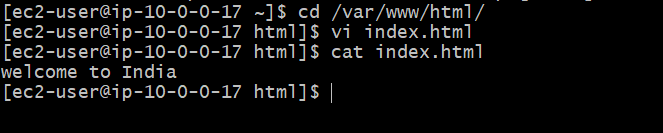
now connect the instance and run with one web server httpd and status should be active for httpd



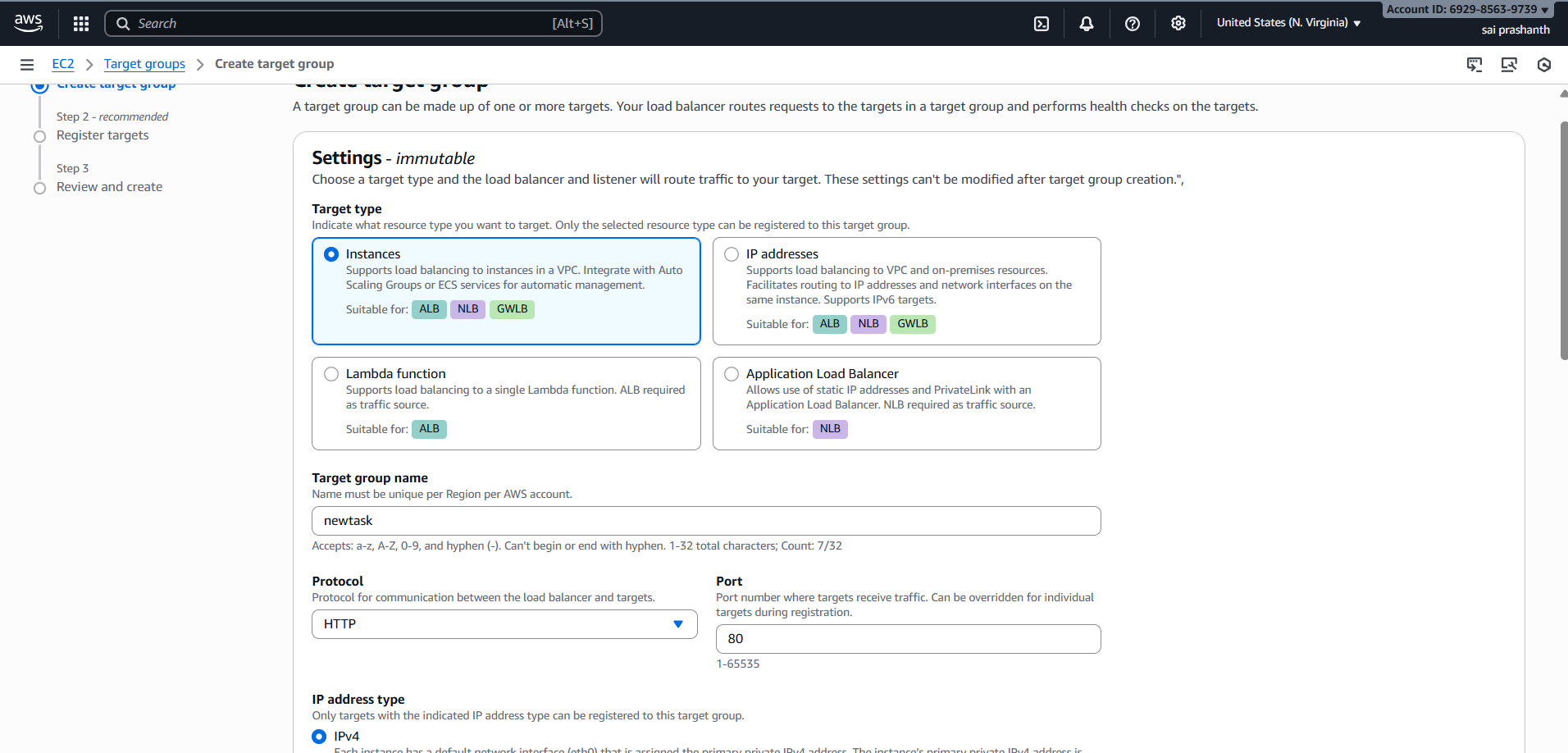
and make changes in index.html by using this path cd /var/www/html

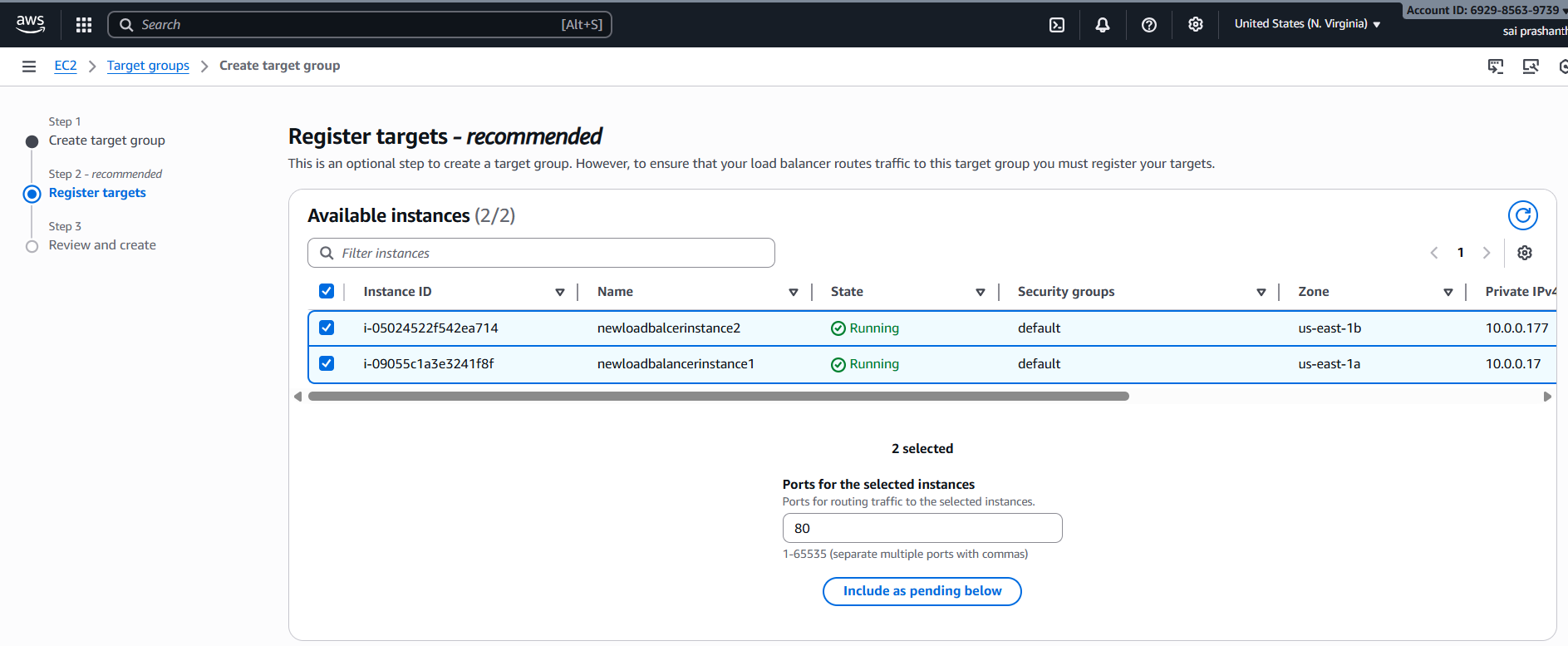
one I created welcome to India

another one with welcome to Tirupathi

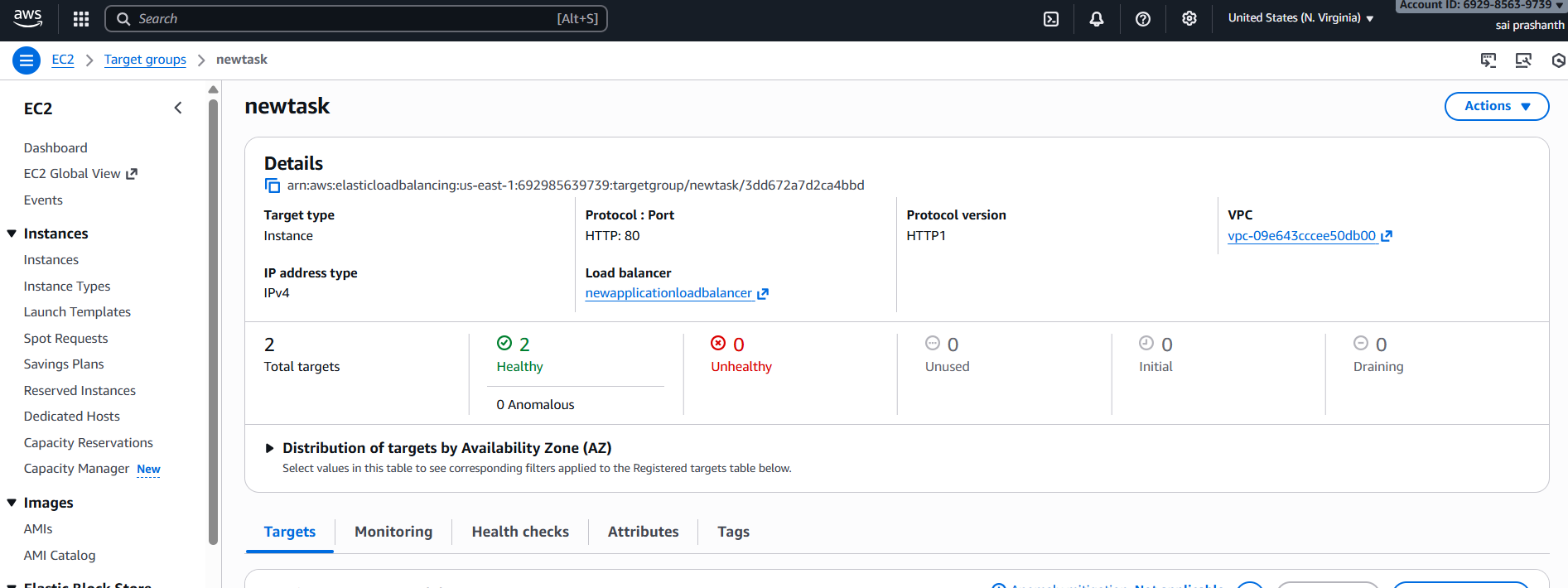


before creating load balancer, we need to create target groups and attach both instances and click on create target group

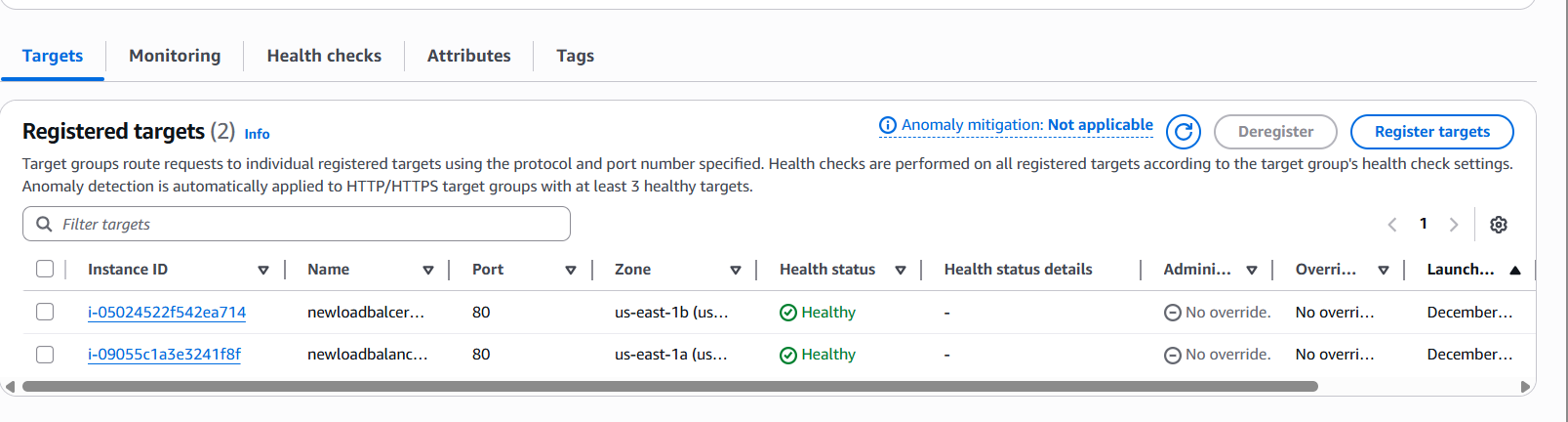




now you can see target group has been created



ensure both instances should be healthy



now we need to create one application load balancer

ec2—load balancer—application load balancer—create

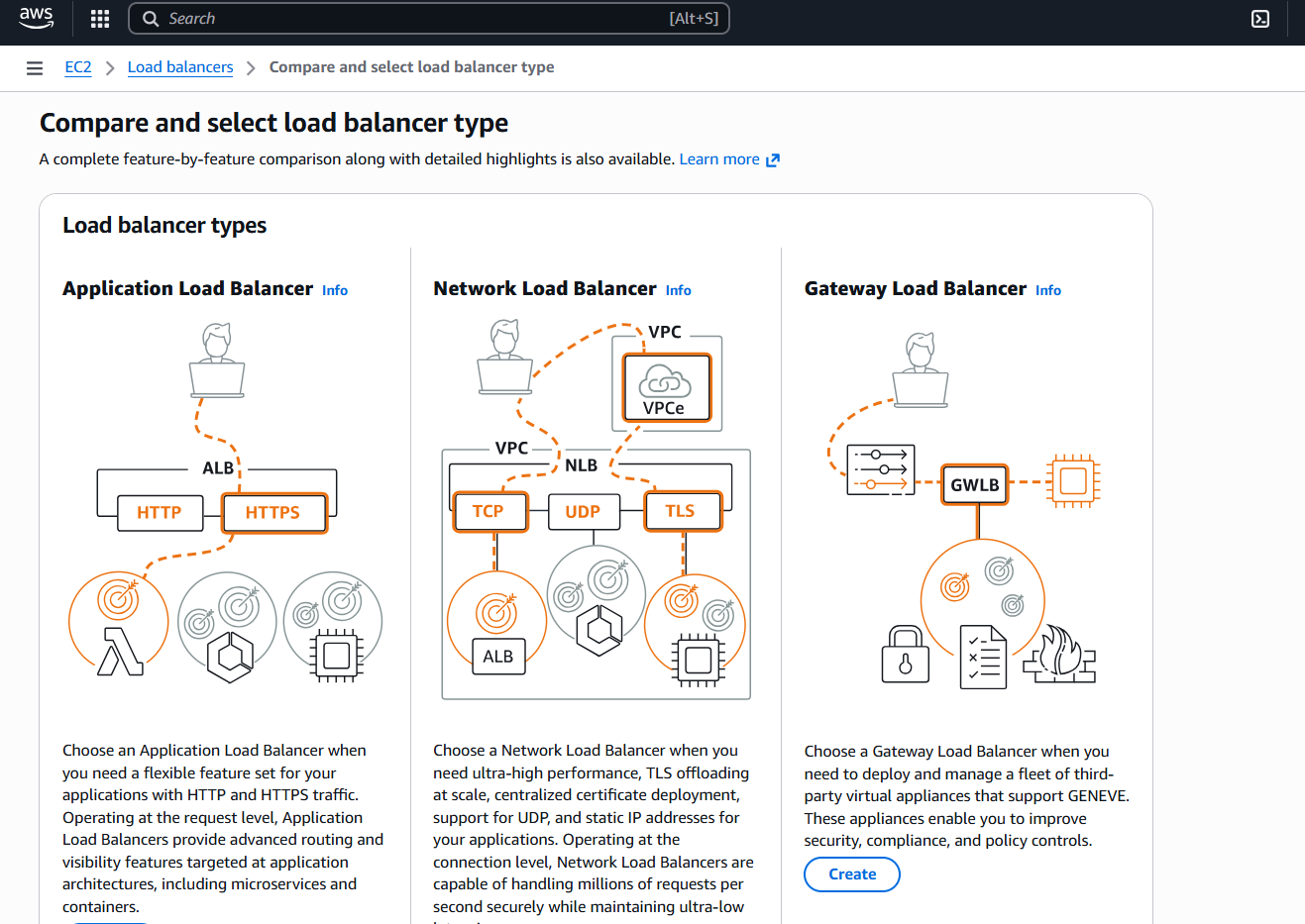
then provide one name

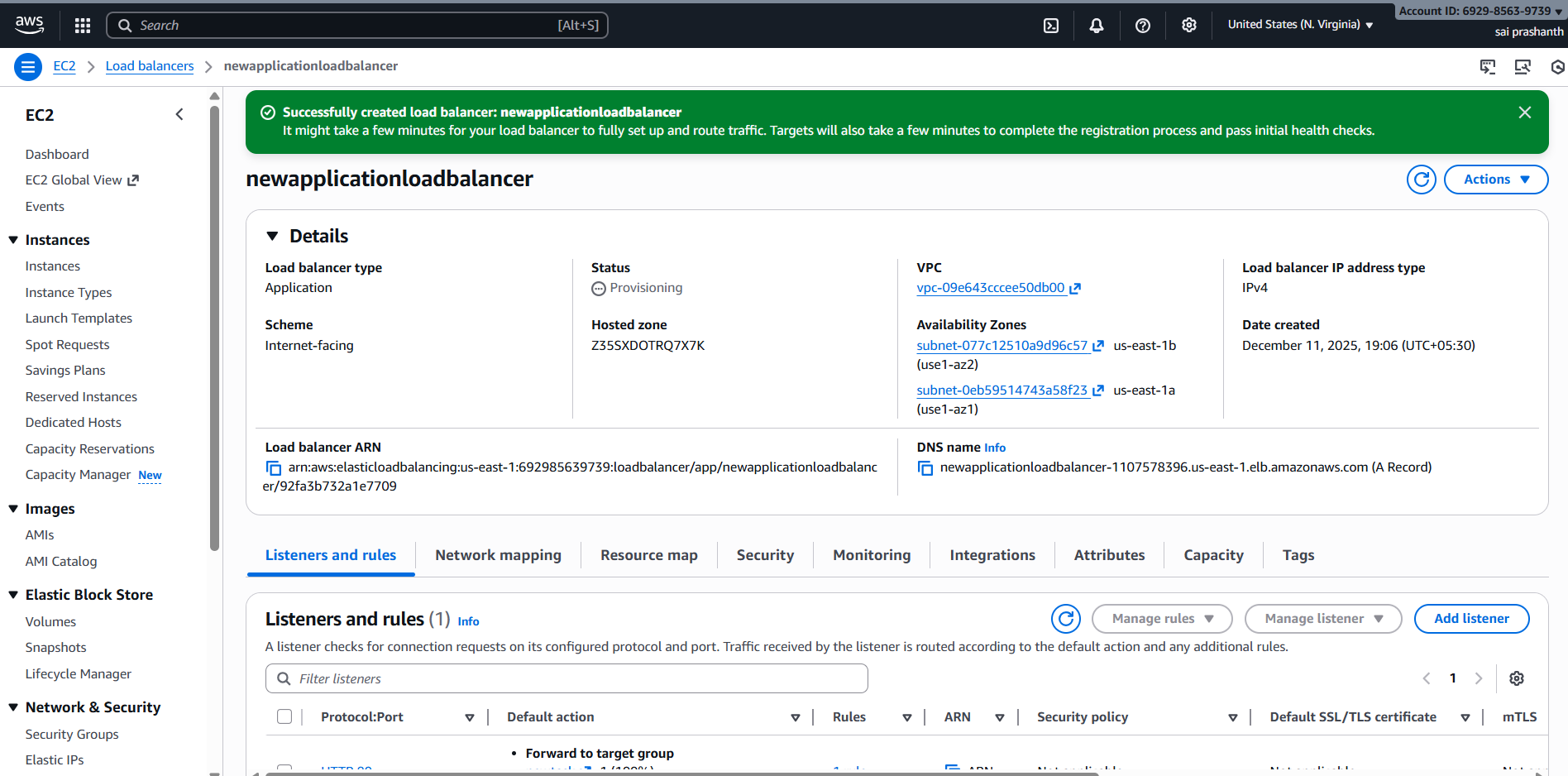
then select vpc

subnets in different regions

link to target group which you have created

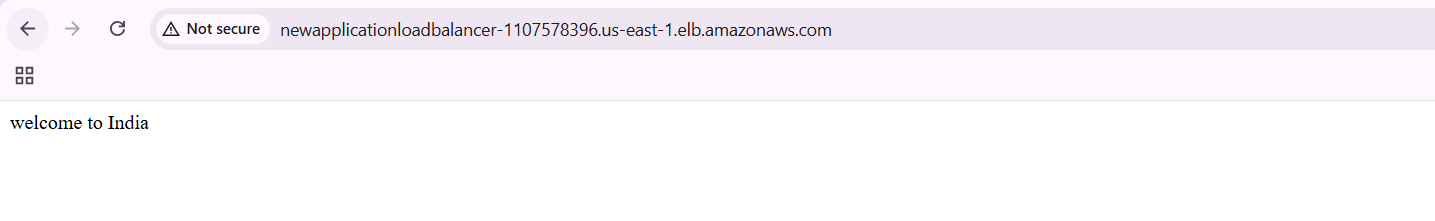
then click on create load balancer

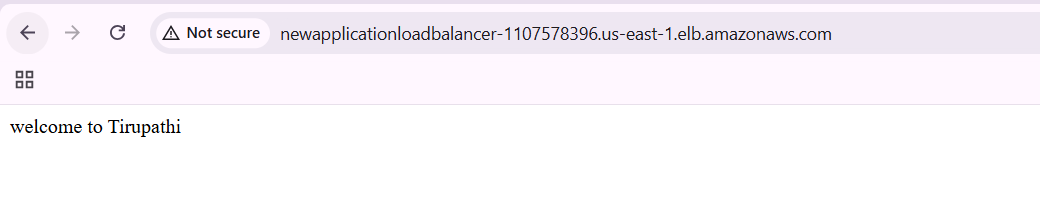




**validation steps:**

for first instance



for second instance  


**conclusion**:

now I have created application load balancer by attaching two instances in target groups

**trouble shoot:**

taking much time for getting display of our web page

this is due to security group configurations

we need to delete and add two times

and click again

1. **title: Configure Network Load balancer.**

**objective:** we need to create one network load balancer

**prerequisites:**

vpc

subnets

ec2 instances

security groups

key pair

route tables

target groups

**step by step process:**

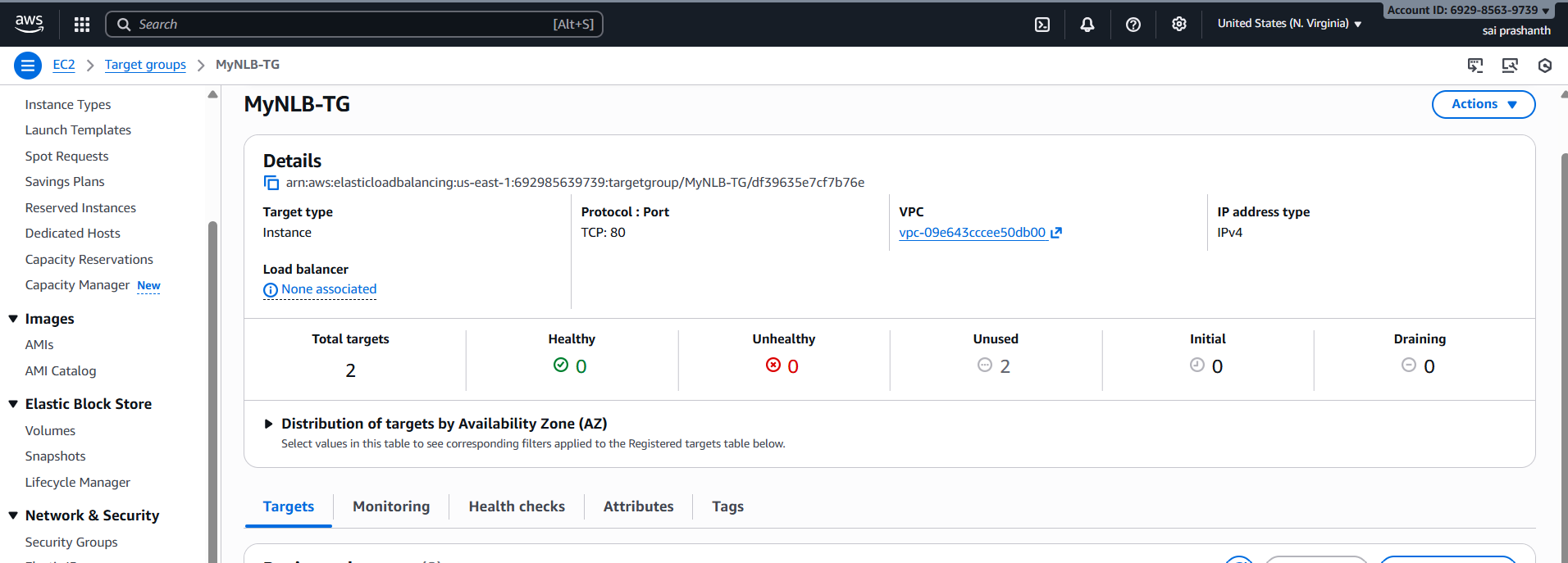
same process like we need to create vpc

2 subnets—attach to public route tables—internet gateway

ec2 instances with two subnets which is in different regions

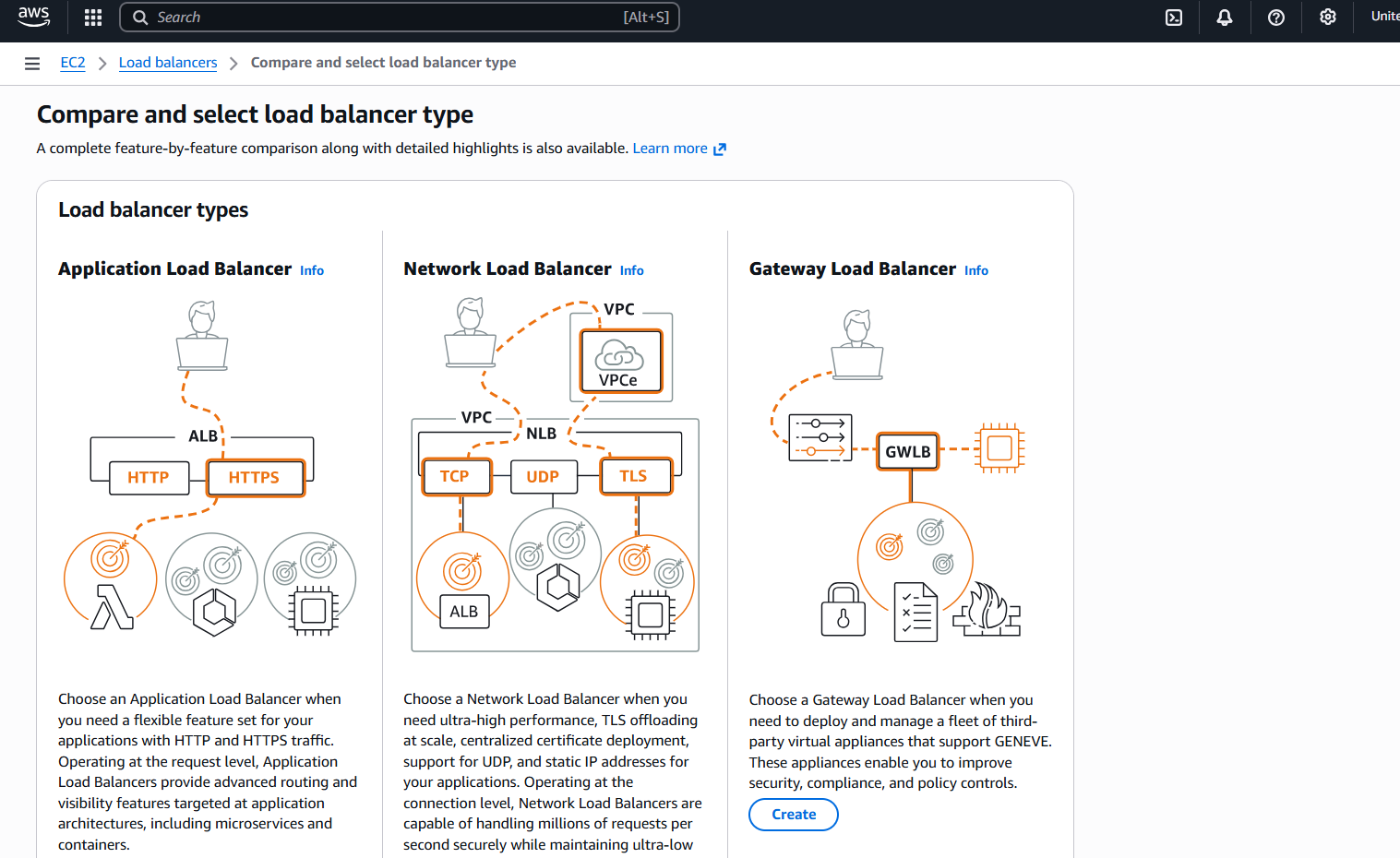
edit security groups with inbound rules like ssh, HTTP, all traffic in ec2

before creating we need to create target groups with protocol tcp

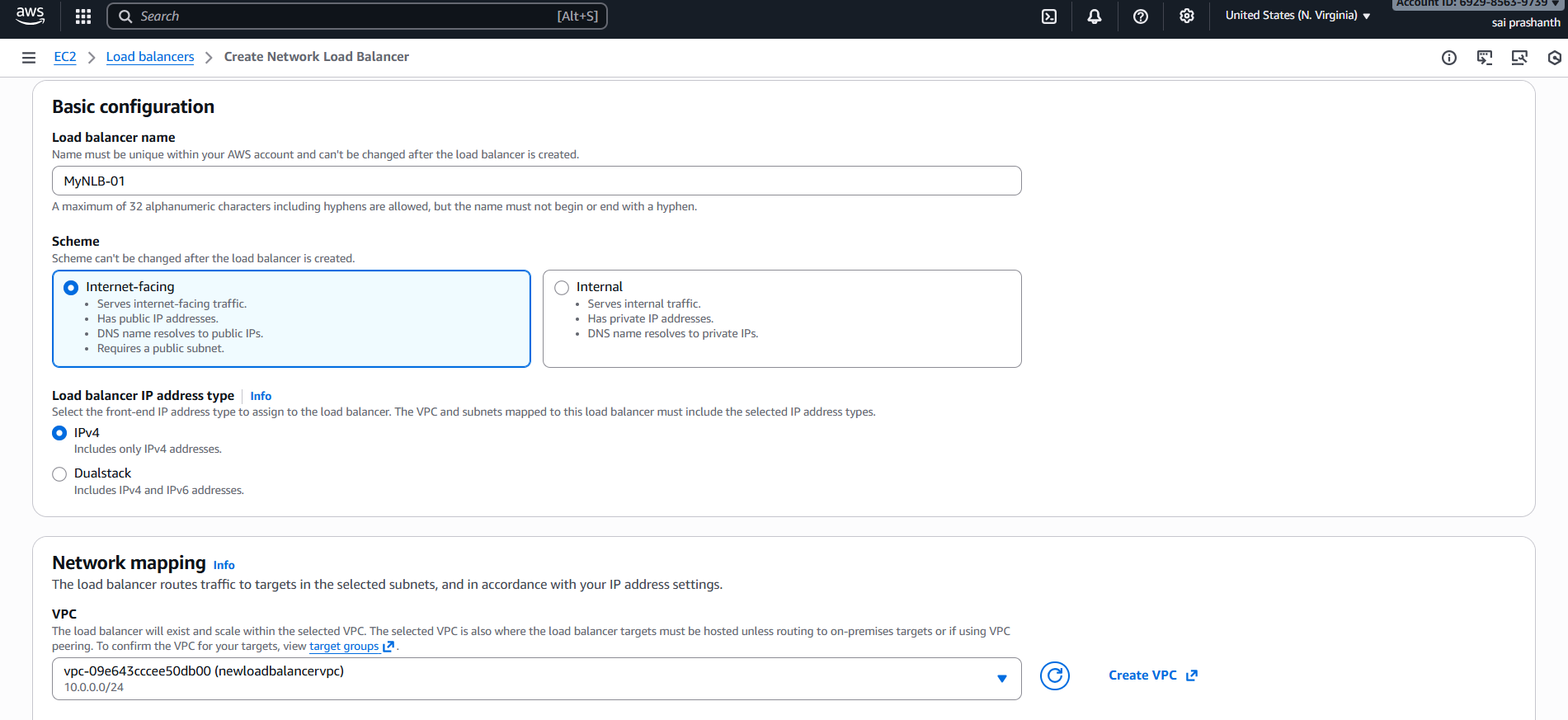


now we need to create network load balancer

go to ec2 console—load balancer—click on network load balancer—create

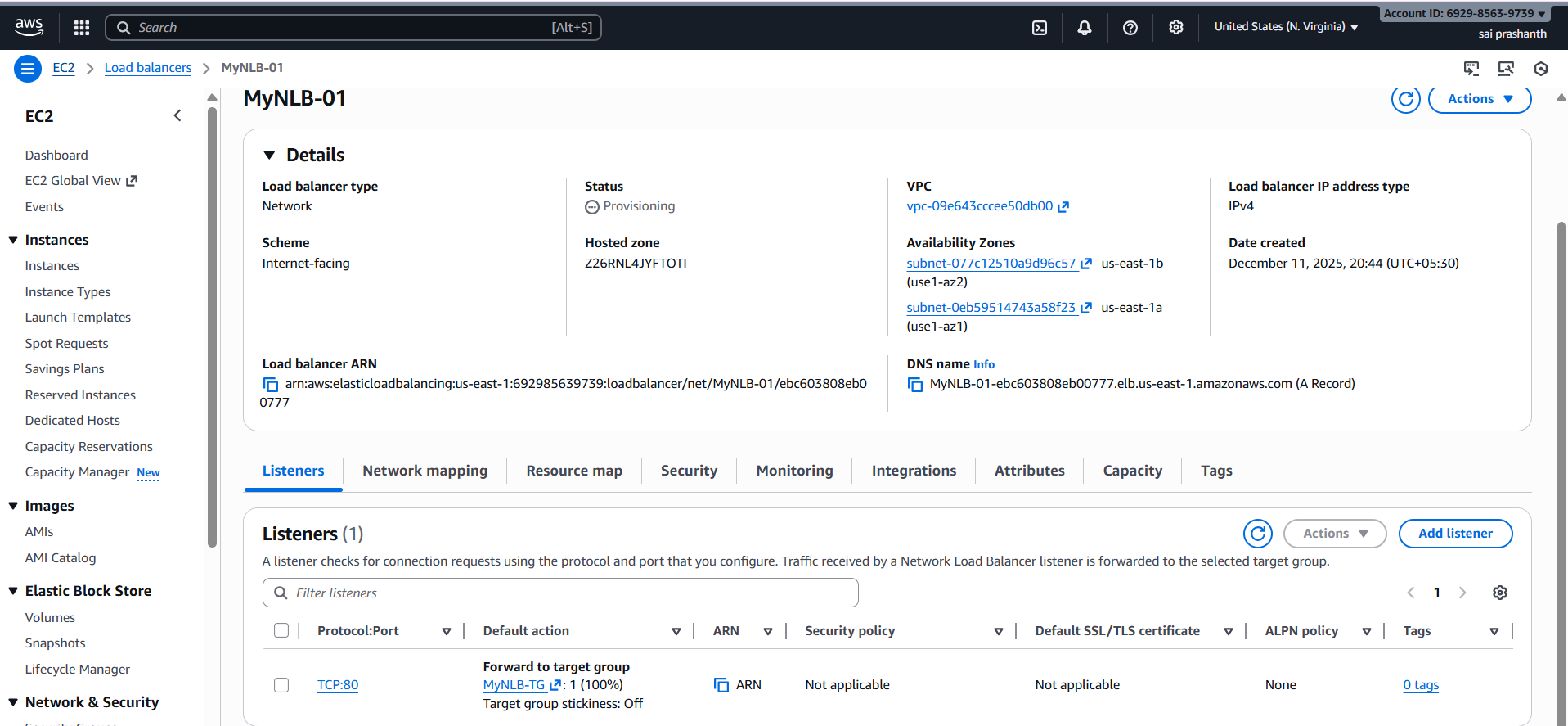


after clicking on create we need to provide name, protocol type with port number 80

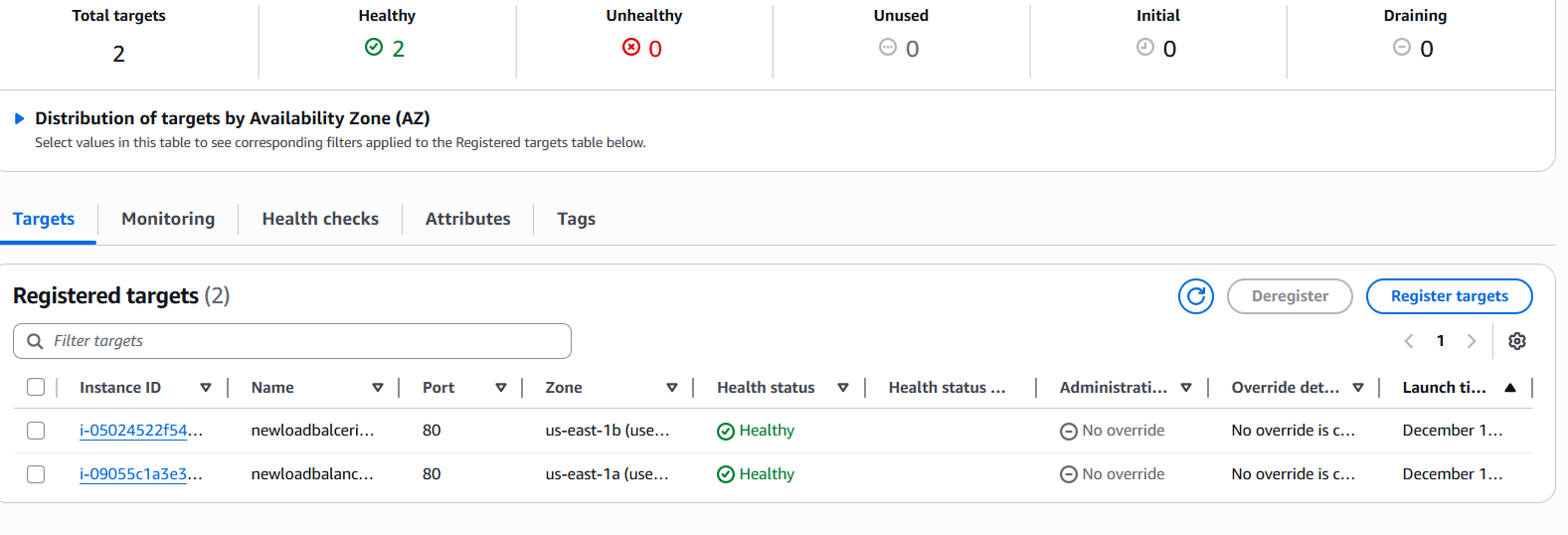


once you click on create load balancer

now you can see the network load balancer has been created successfully



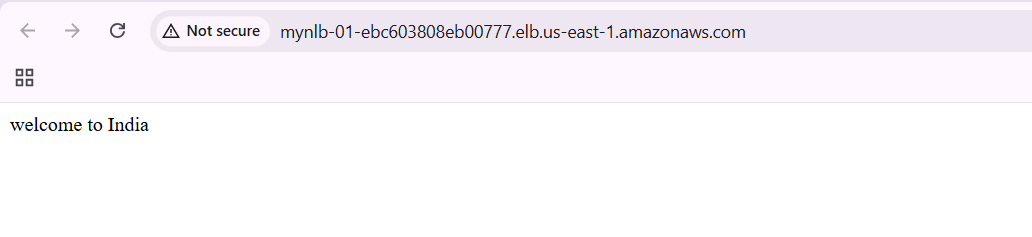
ensure the target group should be healthy



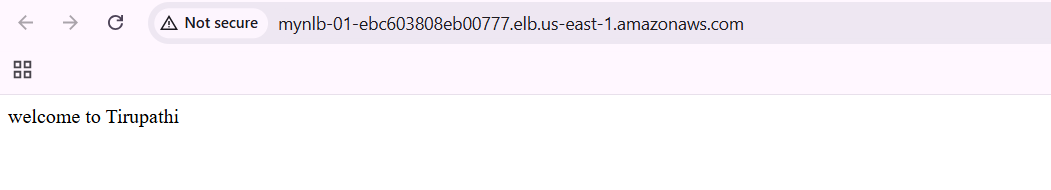
**validation steps:**

now verify the Network load balancer link

with first instance



now verify with the second instance



**conclusion**:

now we have executed NLB with TCP port number 80

1. **title: Attach SSL for application load balancer.**
2. **title: Map Application load balancer to R53.**

**objective: we need to create one SSL certificate and attach it to a load balancer and we need to map to route 53**

**prerequisites:**

acm certificate

domain name

hosted zone

route 53

target group load balancer

security group 443

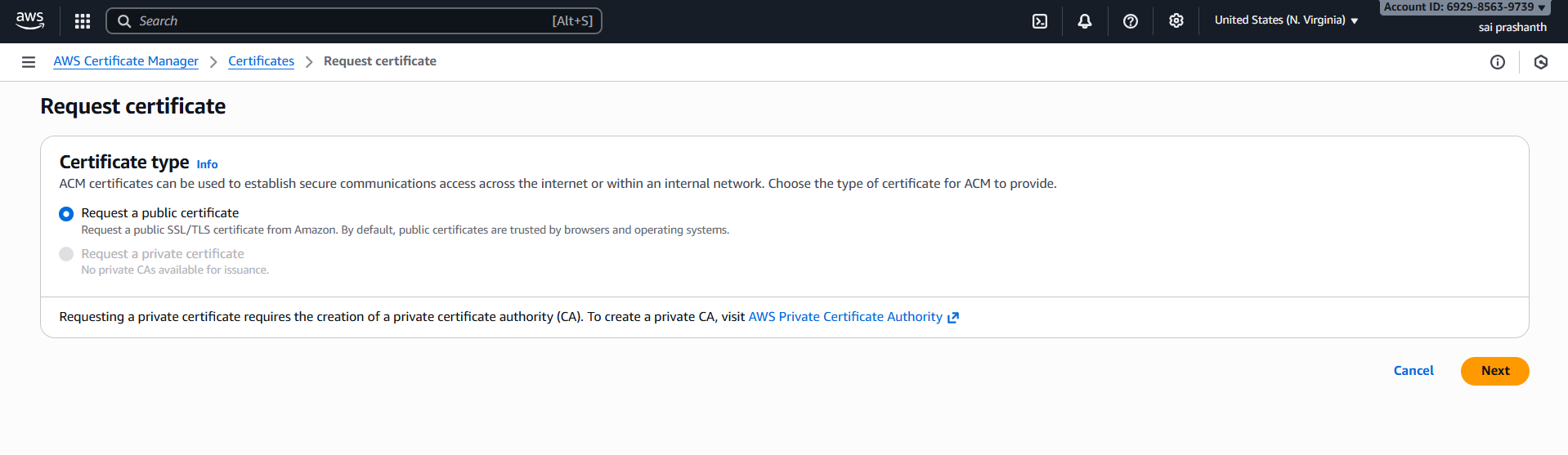
**step by step process:**

first we need to create certificate

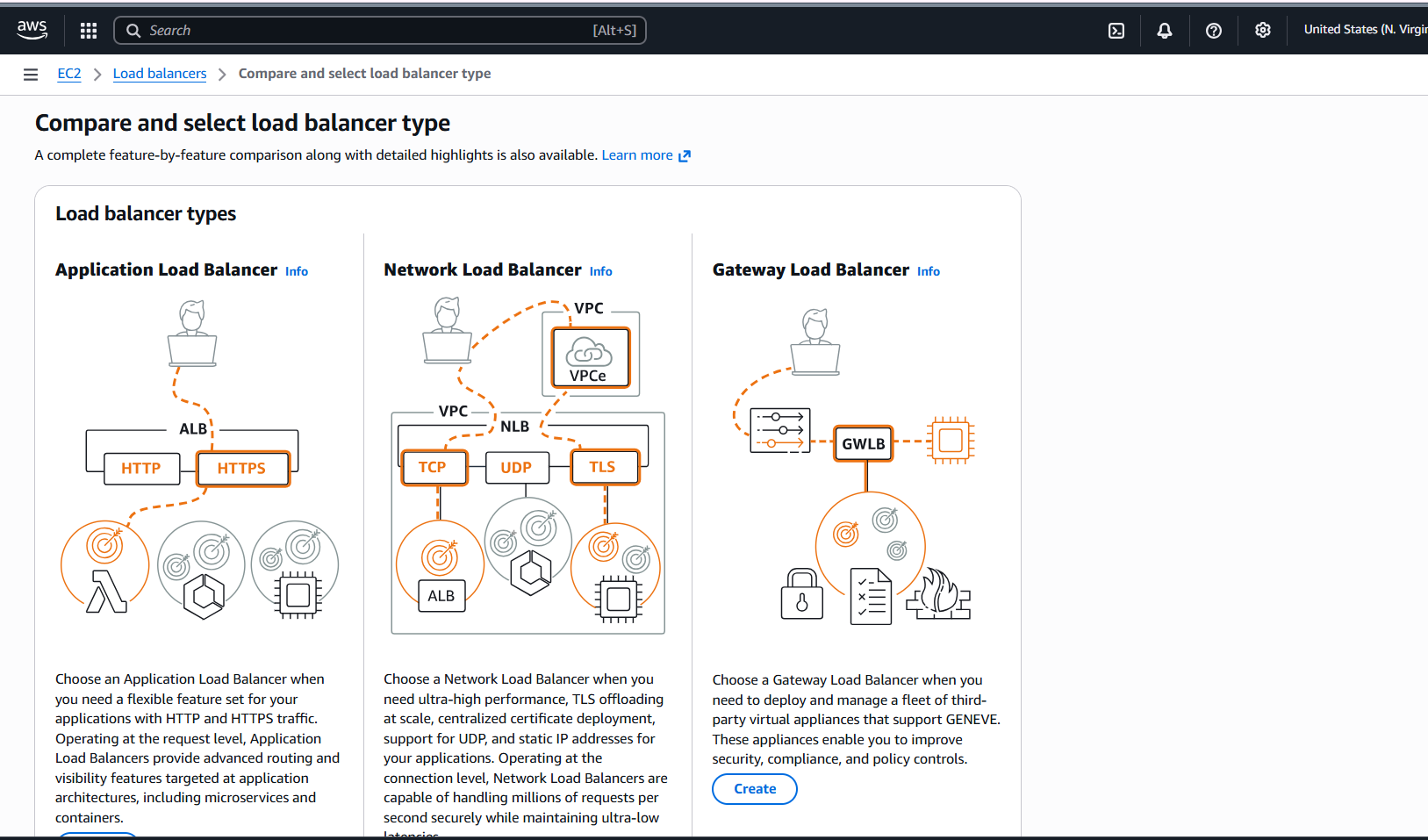
go to acm—create certificate

now enter your domain name

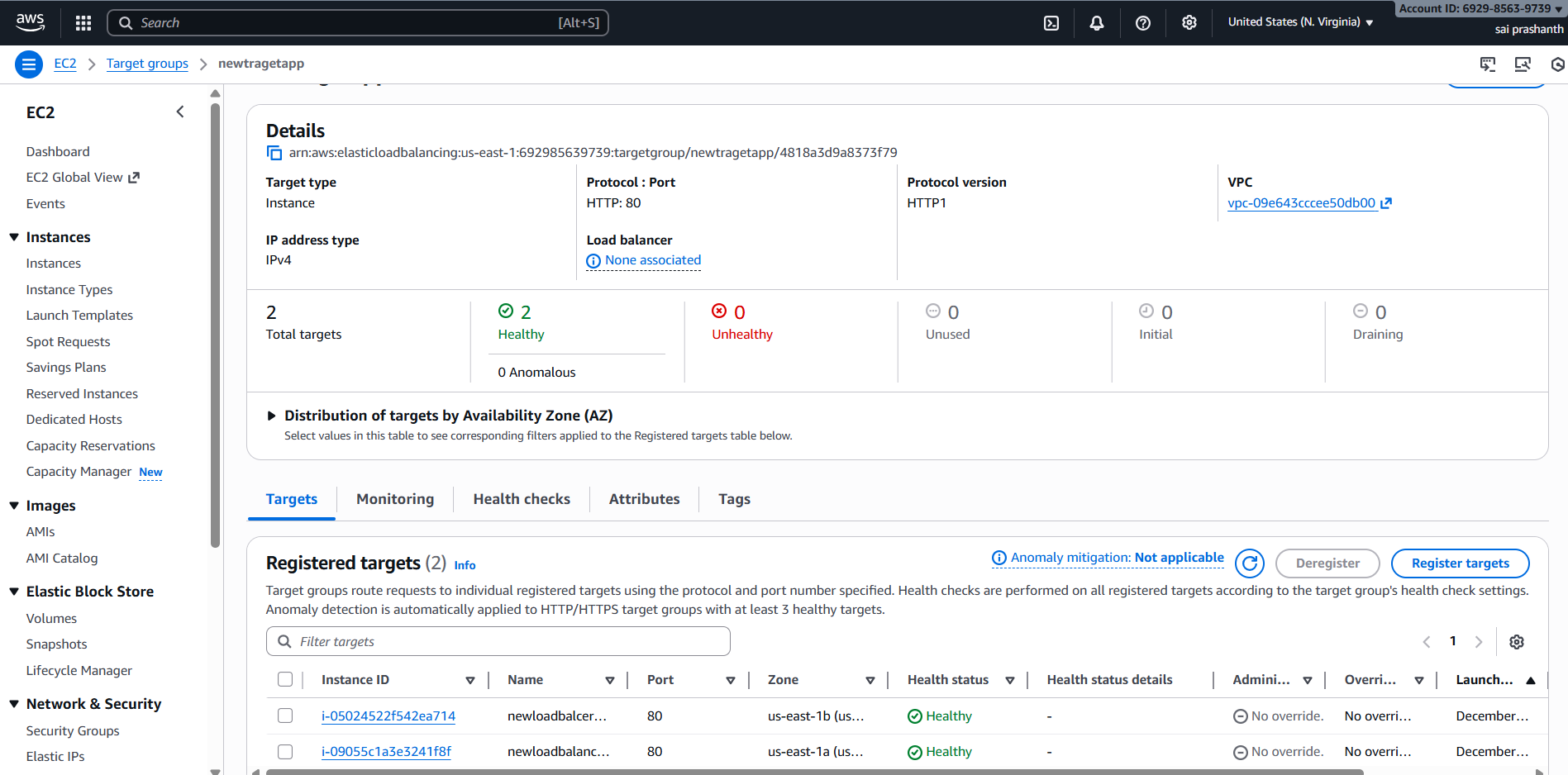
click request



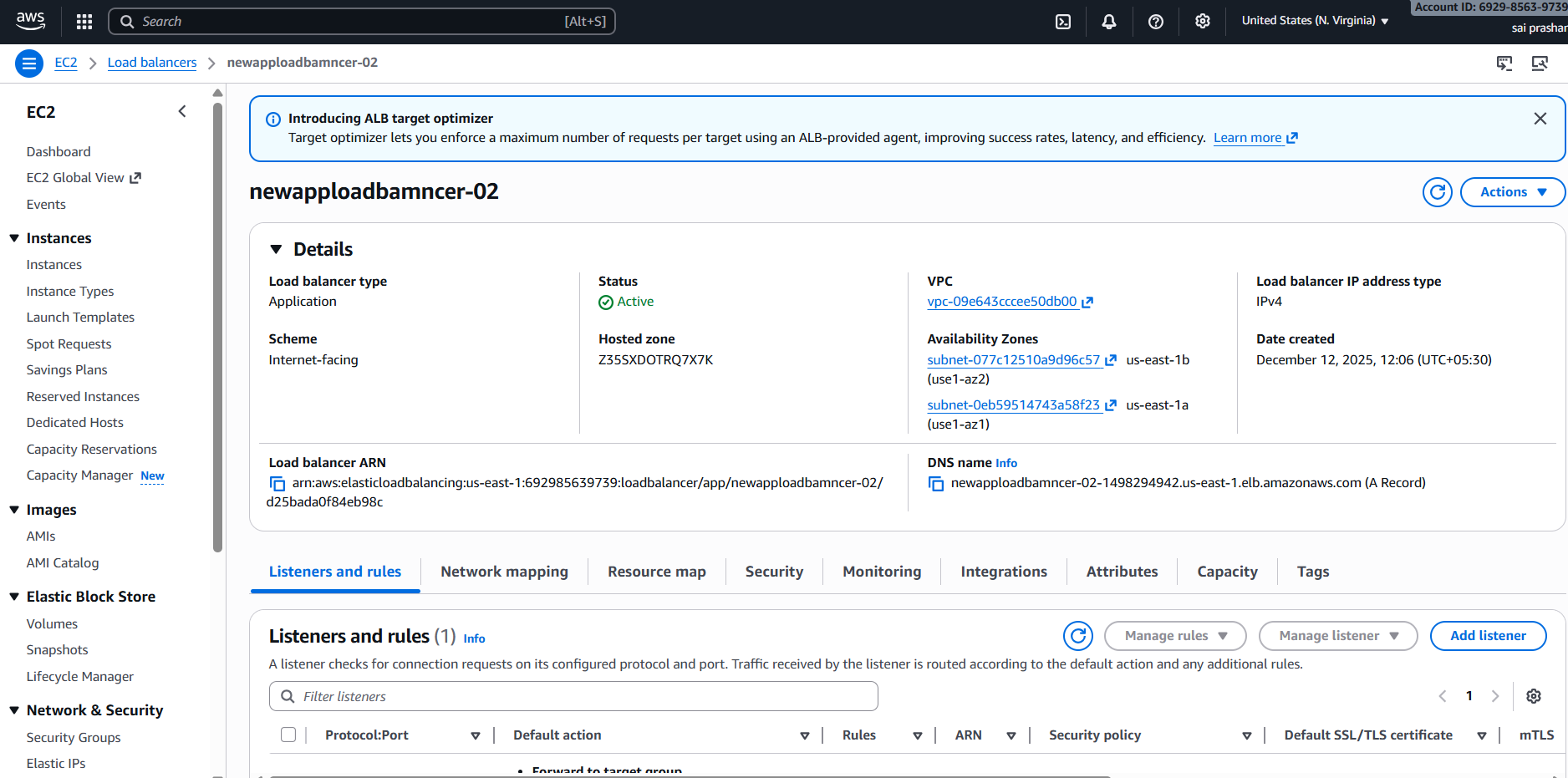
now create one application load balancer



create one target group and attach the instances and it should be healthy

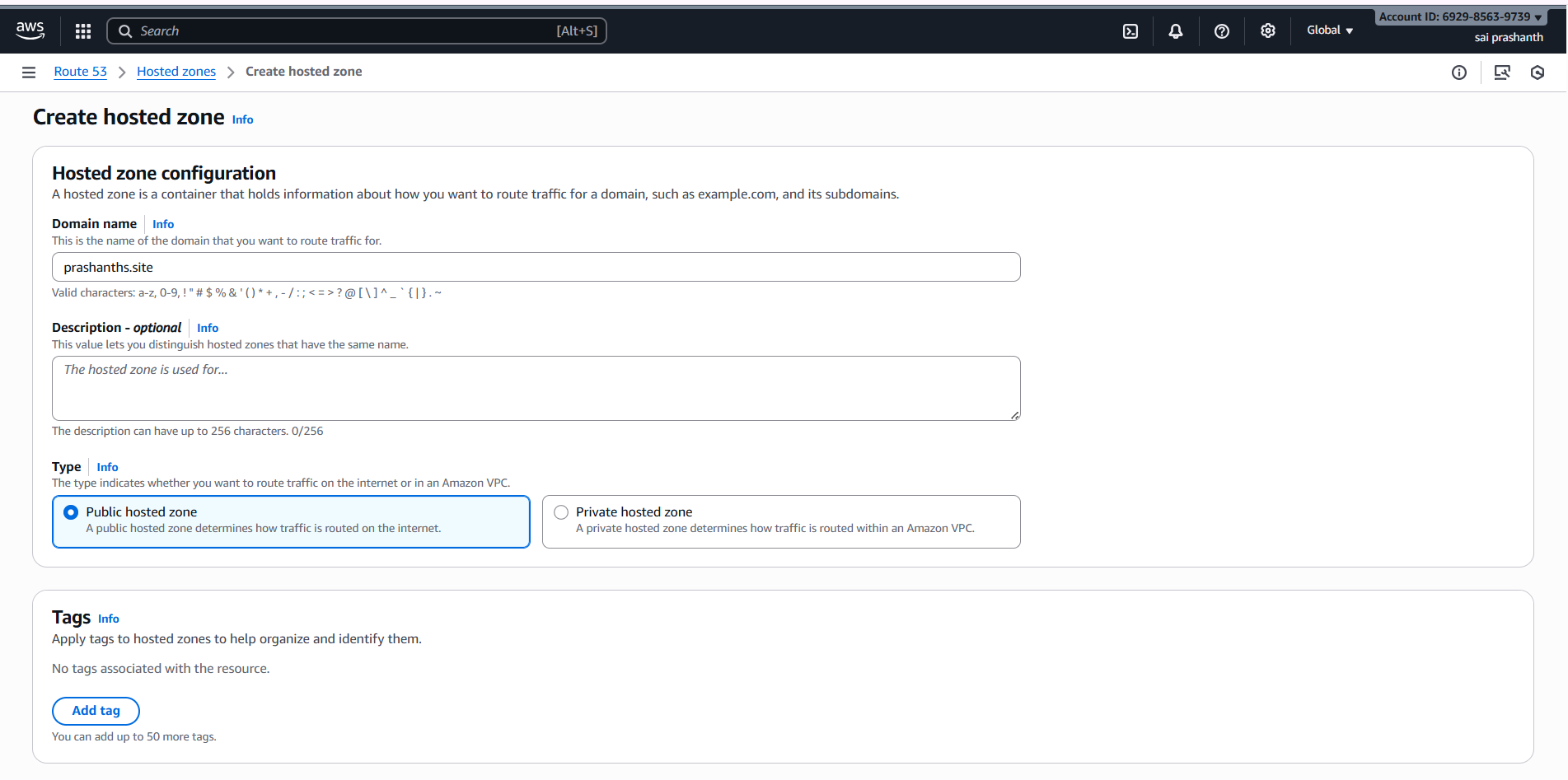


now create one load balancer and attach this target group

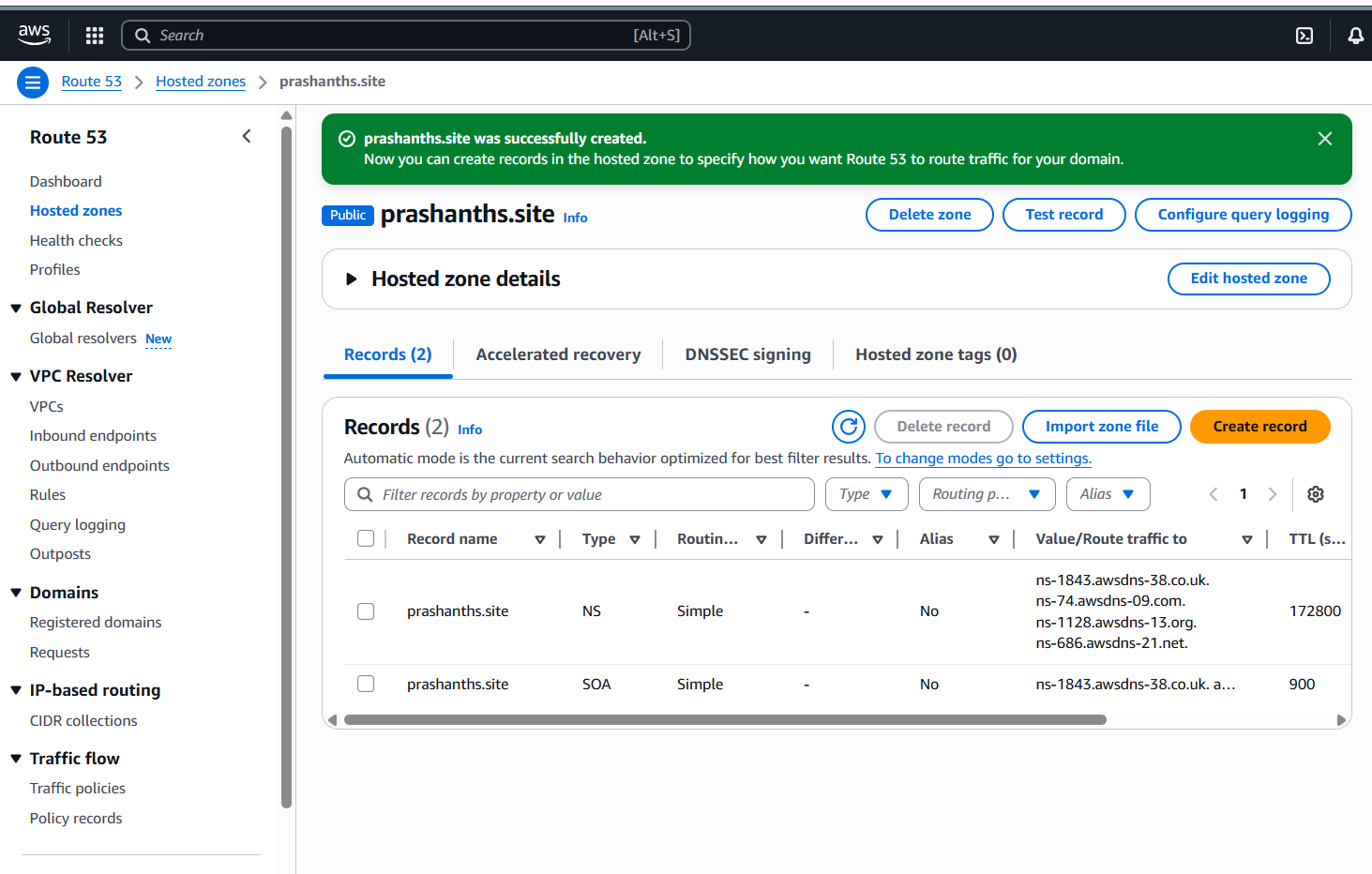


once it has been done now attach ssl certificate to route 53

go to route 53—create one hosted zone—click on create

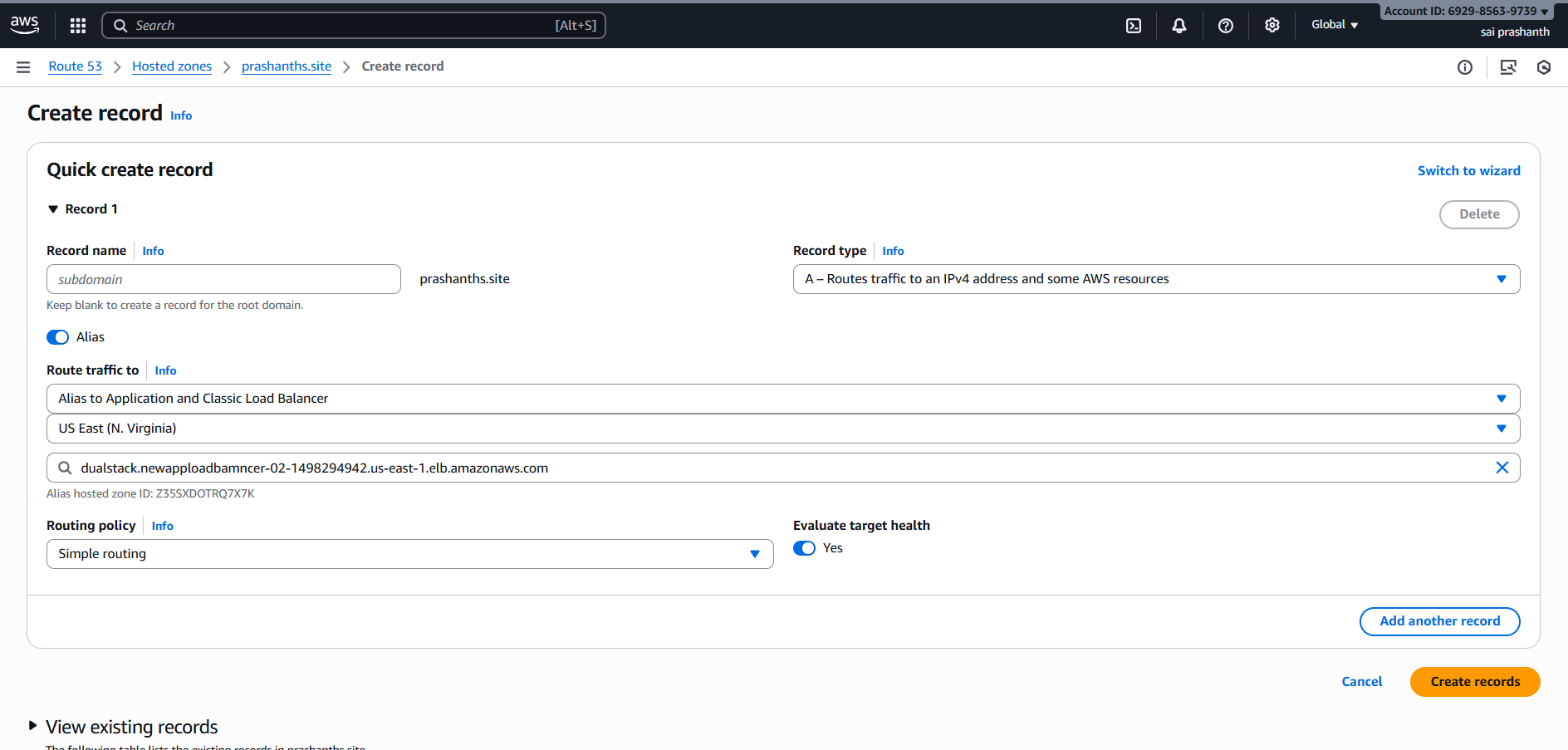


once you created hosted zone with your domain name

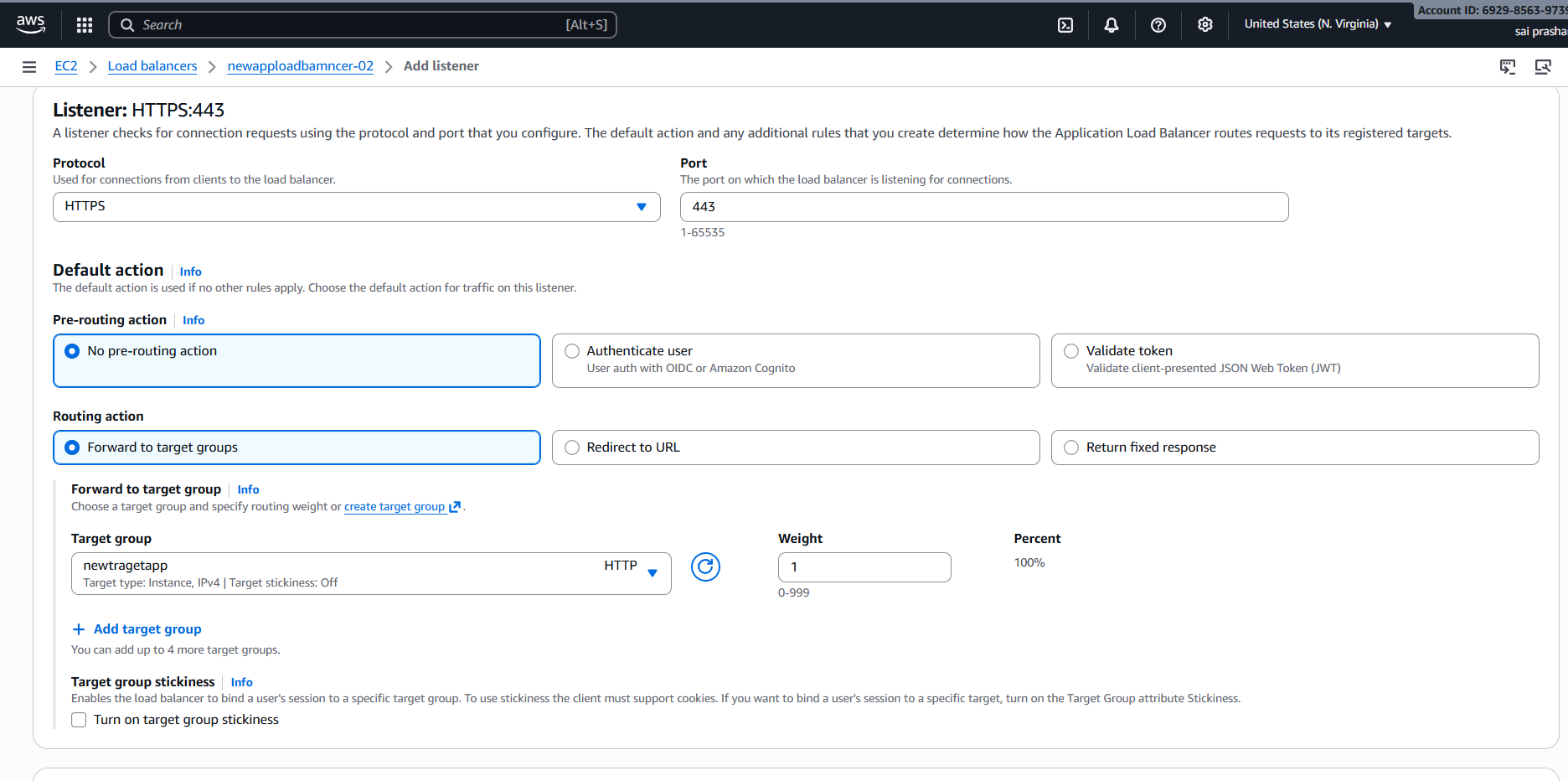


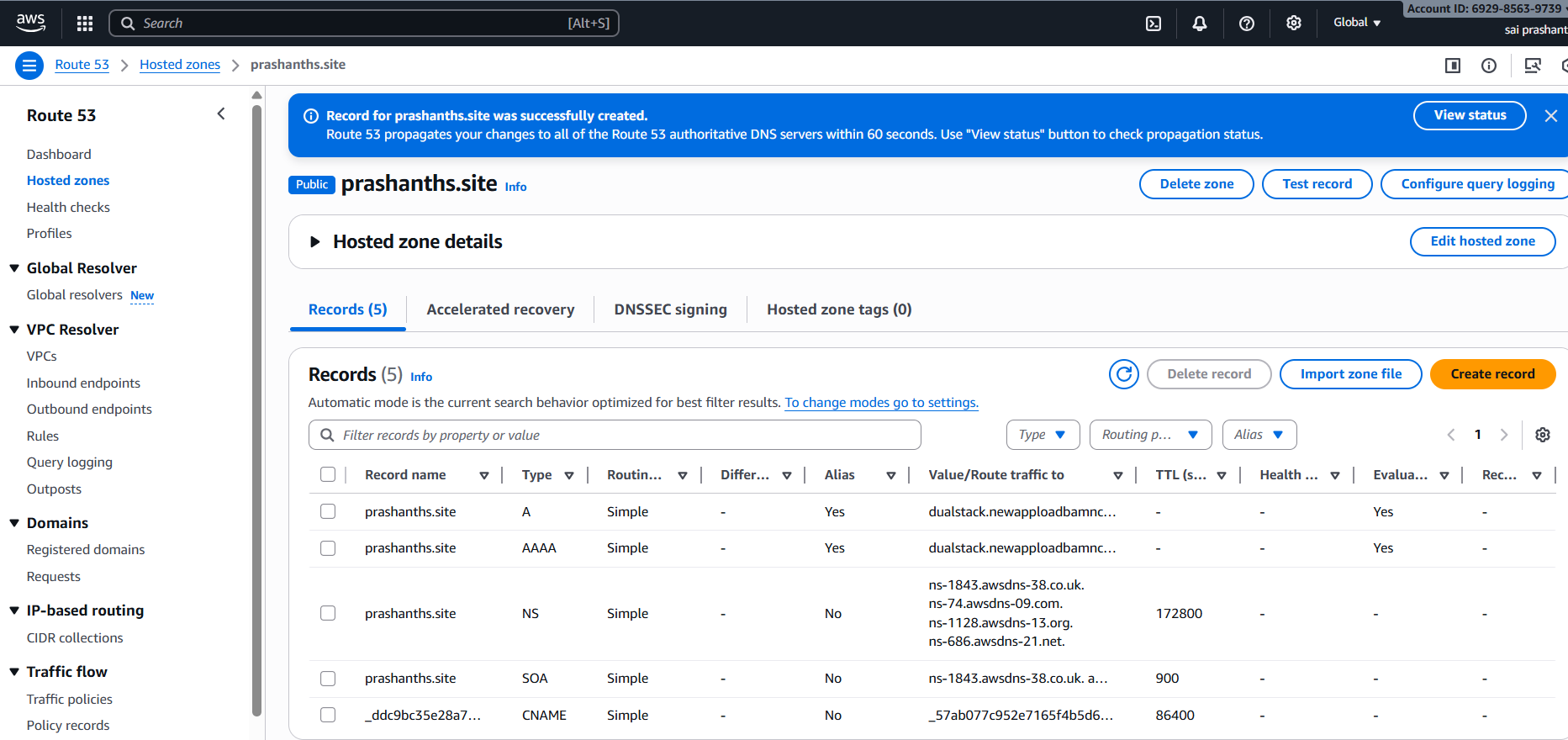
and add some more records

add a and aaaa and in alias choose your end point to application load balancer

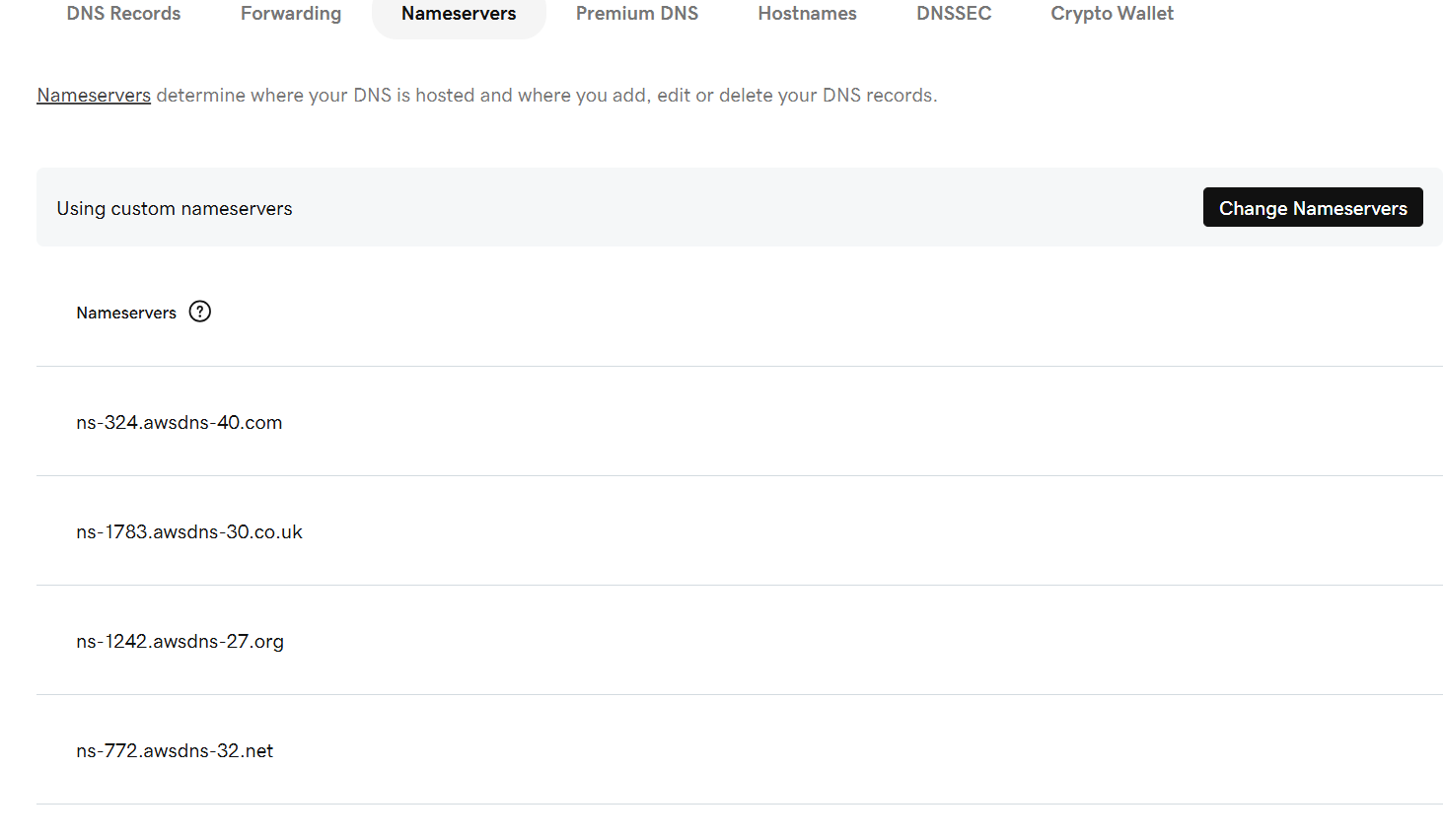


choose region in above column

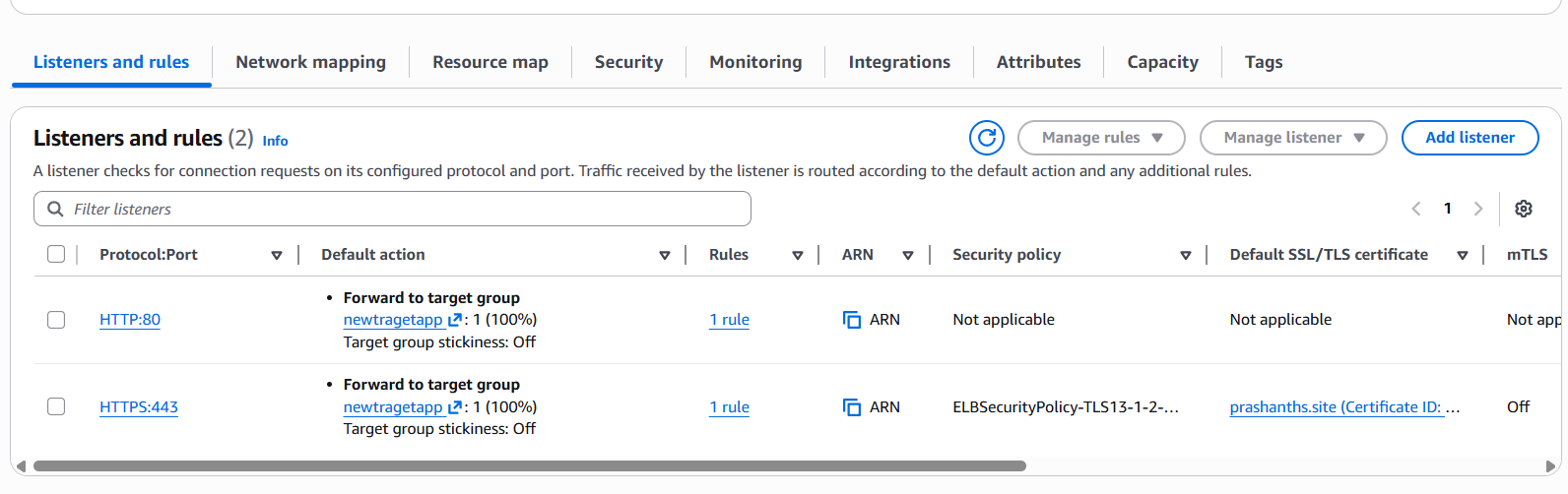




now add these name servers and in your go daddy website



now come to application load balancer—click on add listener—enable https with port number 443 and add the certificate name and click on add listener



**verification steps:**

now we need to check with domain name

it should display as what we have created in index in instance 1 and 2

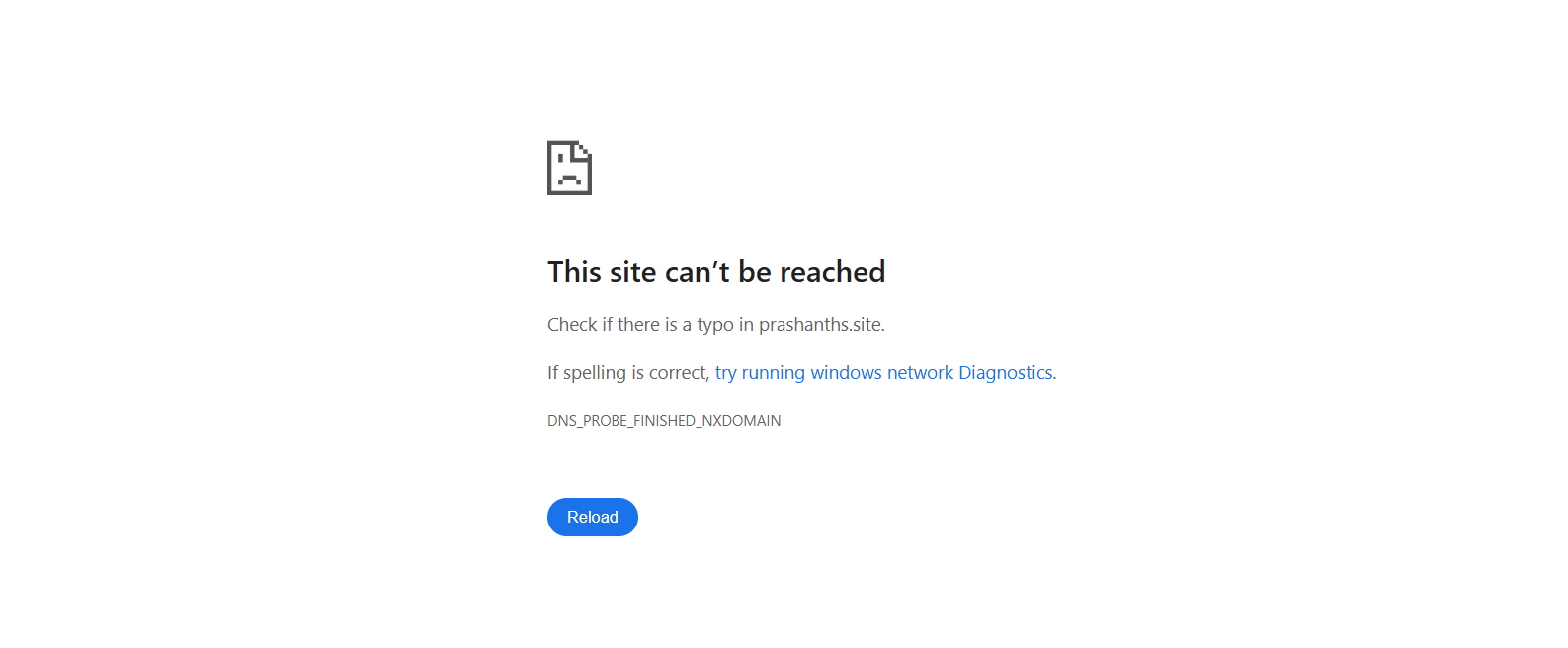
I have purchased new domain again and waiting for certificate issued status. once it has been moved to issue status i

ll upload again and paste the screen shot

**trouble shoot:**

some times dns working in browser

we need to do keep on refresh



1. **title: Push the application load balancer logs to S3.**

**objective:** we need to the logs of application load balancer to S3

prerequisites:

s3 buckets

application load balancer

target groups

ec2 instances attaching

Alb logs

**step by step process:**

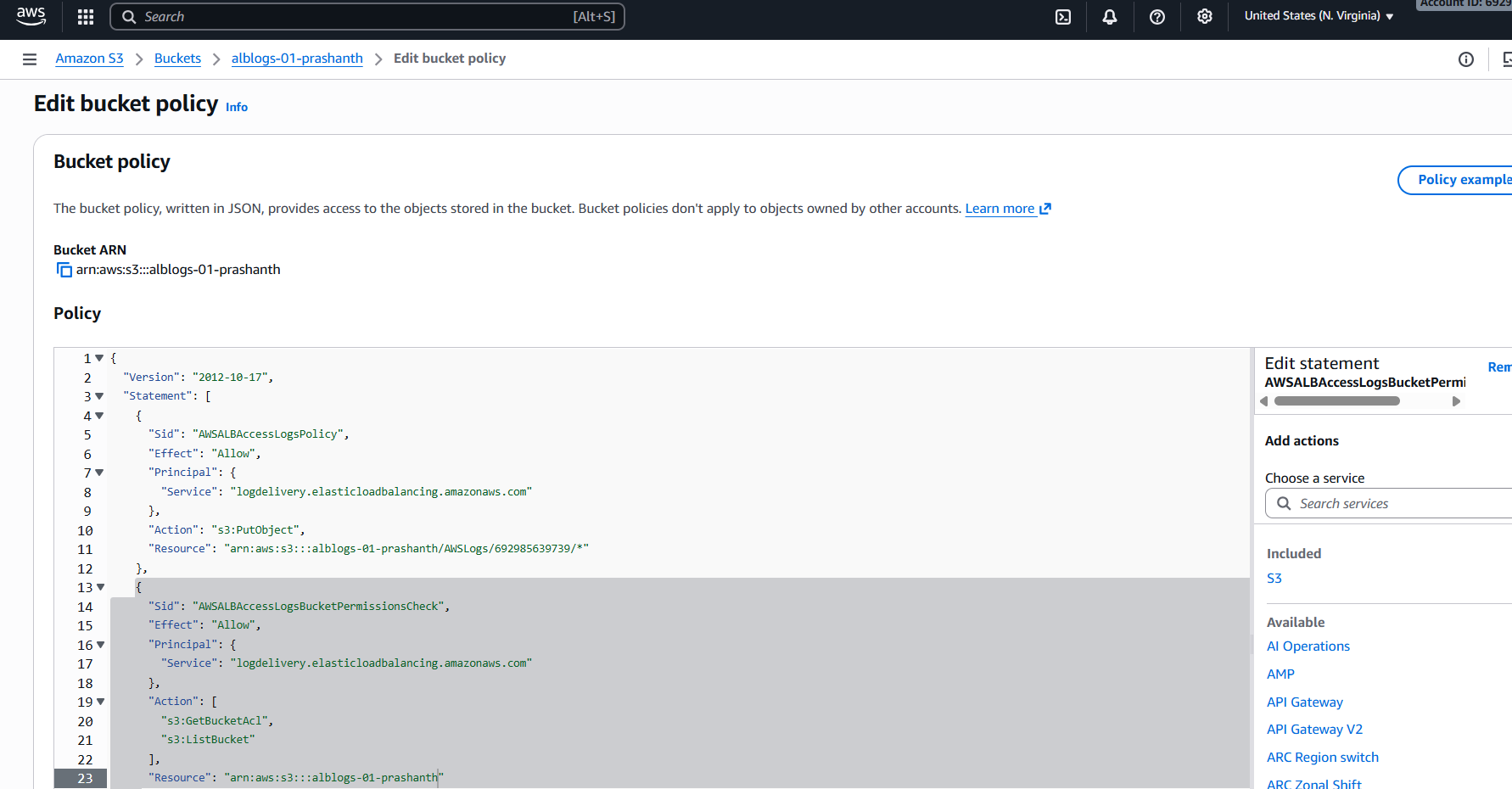
first go S3---create bucket—enter the bucket name and click on create



then click on your bucket---permissions—click on bucket policy---edit—in this policy enter your bucket name and account id

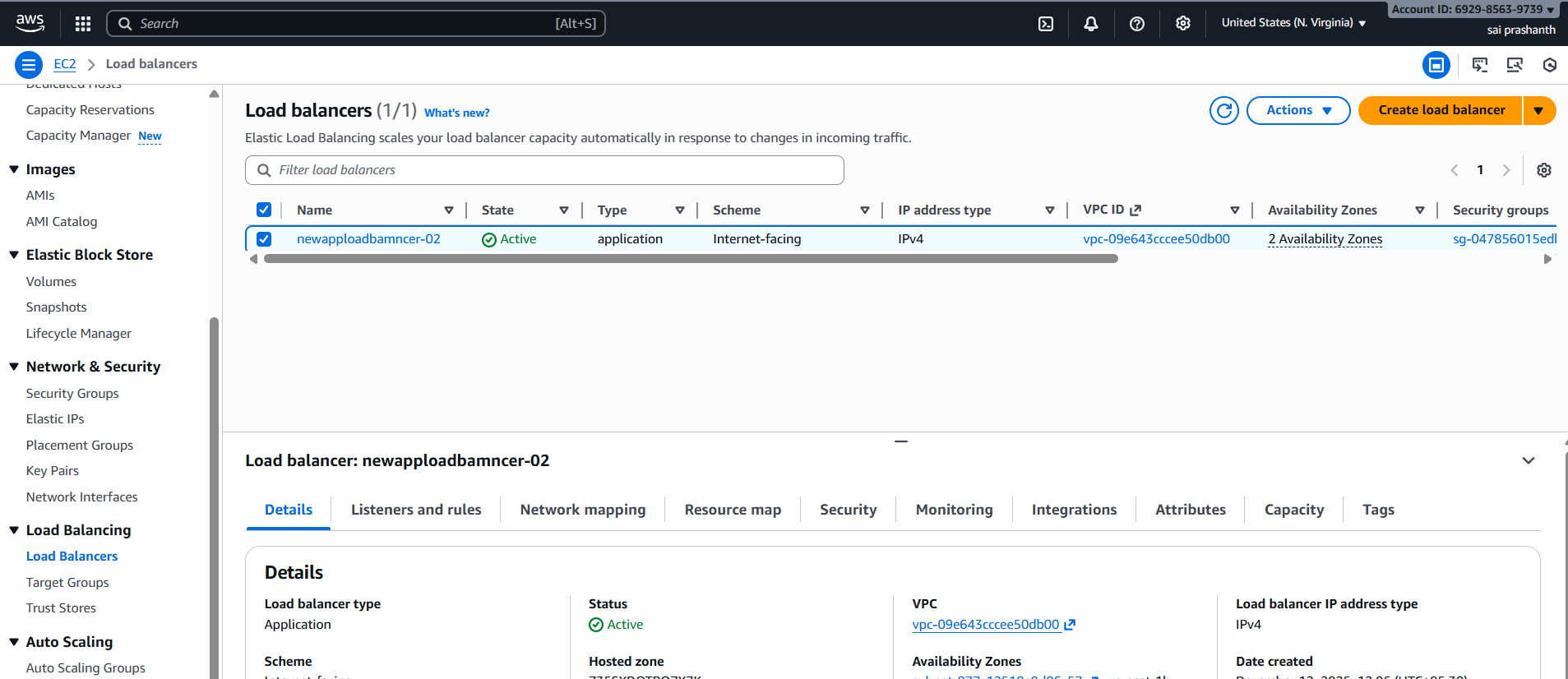
{  
  "Version": "2012-10-17",  
  "Statement": [  
    {  
      "Sid": "AWSALBAccessLogsPolicy",  
      "Effect": "Allow",  
      "Principal": {  
        "Service": "[logdelivery.elasticloadbalancing.amazonaws.com](http://logdelivery.elasticloadbalancing.amazonaws.com/)"  
      },  
      "Action": "s3:PutObject",  
      "Resource": "arn:aws:s3:::YOUR\_BUCKET\_NAME/AWSLogs/YOUR\_AWS\_ACCOUNT\_ID/\*"  
    },  
    {  
      "Sid": "AWSALBAccessLogsBucketPermissionsCheck",  
      "Effect": "Allow",  
      "Principal": {  
        "Service": "[logdelivery.elasticloadbalancing.amazonaws.com](http://logdelivery.elasticloadbalancing.amazonaws.com/)"  
      },  
      "Action": [  
        "s3:GetBucketAcl",  
        "s3:ListBucket"  
      ],  
      "Resource": "arn:aws:s3:::YOUR\_BUCKET\_NAME"  
    }  
  ]  
}

then click on save changes



now go to application load balancer which you have been already created

select you alb—go to attributes—click on edit

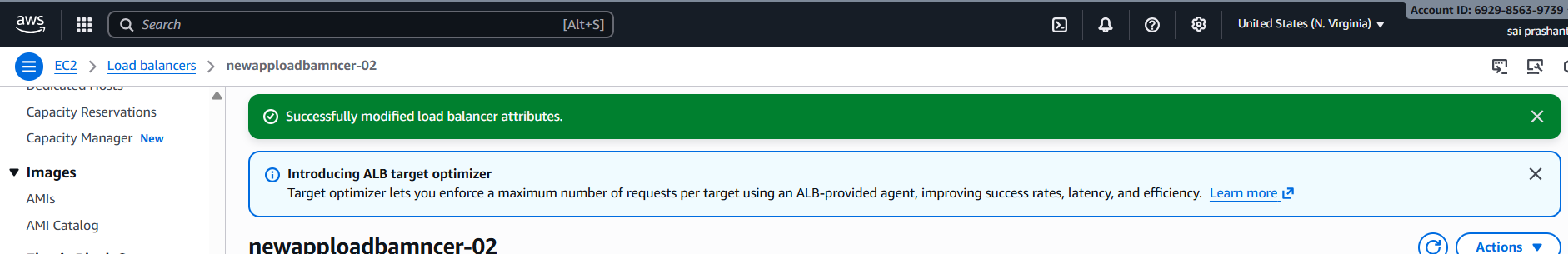


enable access logs and browse your s3 bucket

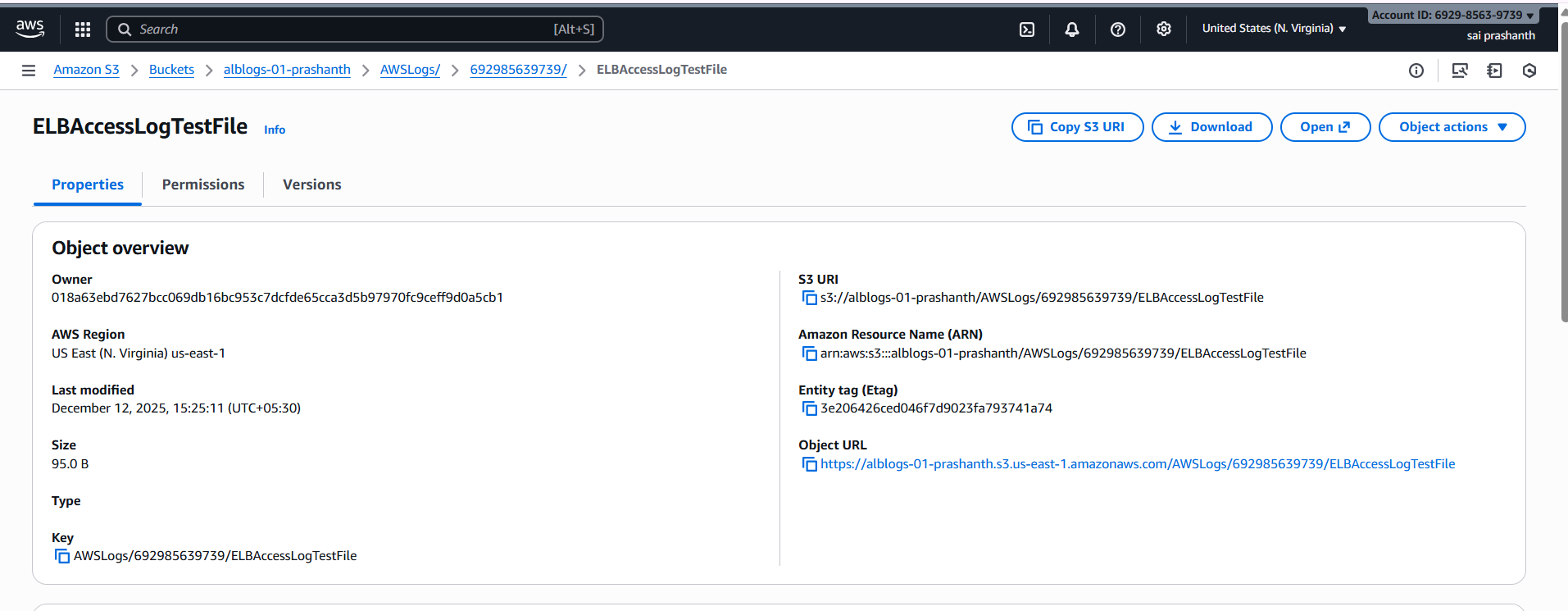
select and click on save changes



now you can see alb logs has been successfully modified



**validation steps:**

now you see alb logs has been moved to s3

**conclusion**:

**now you can see application load balancer logs has been moved to s3 for monitoring and secutiry purpose.**

**The logs have been stored in s3**