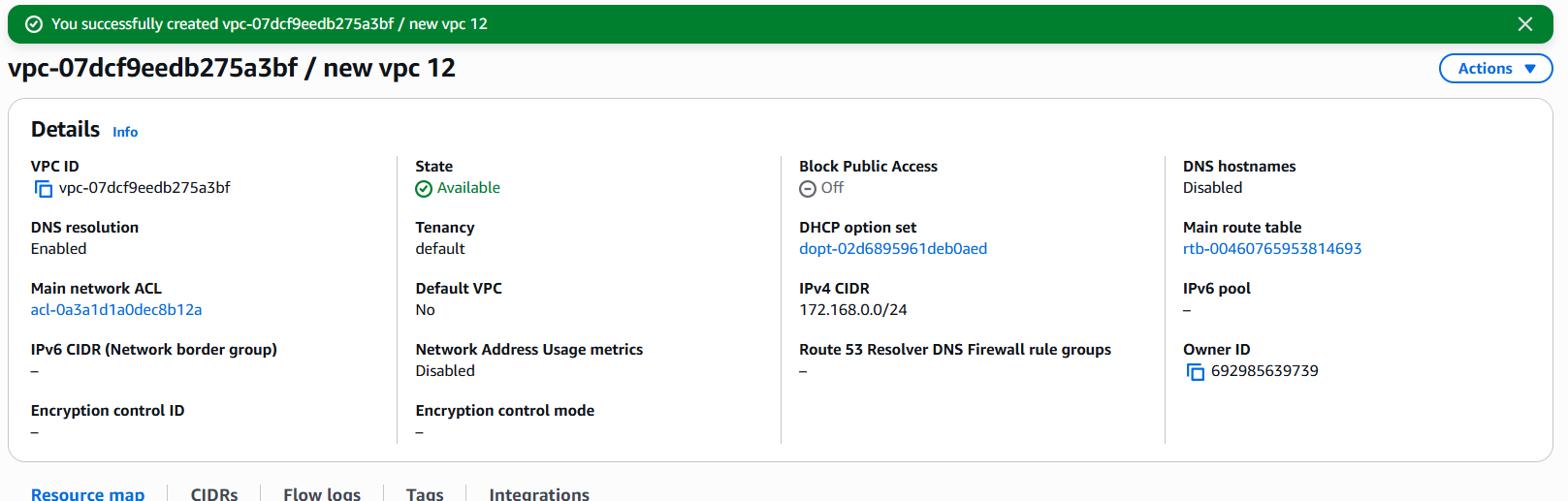
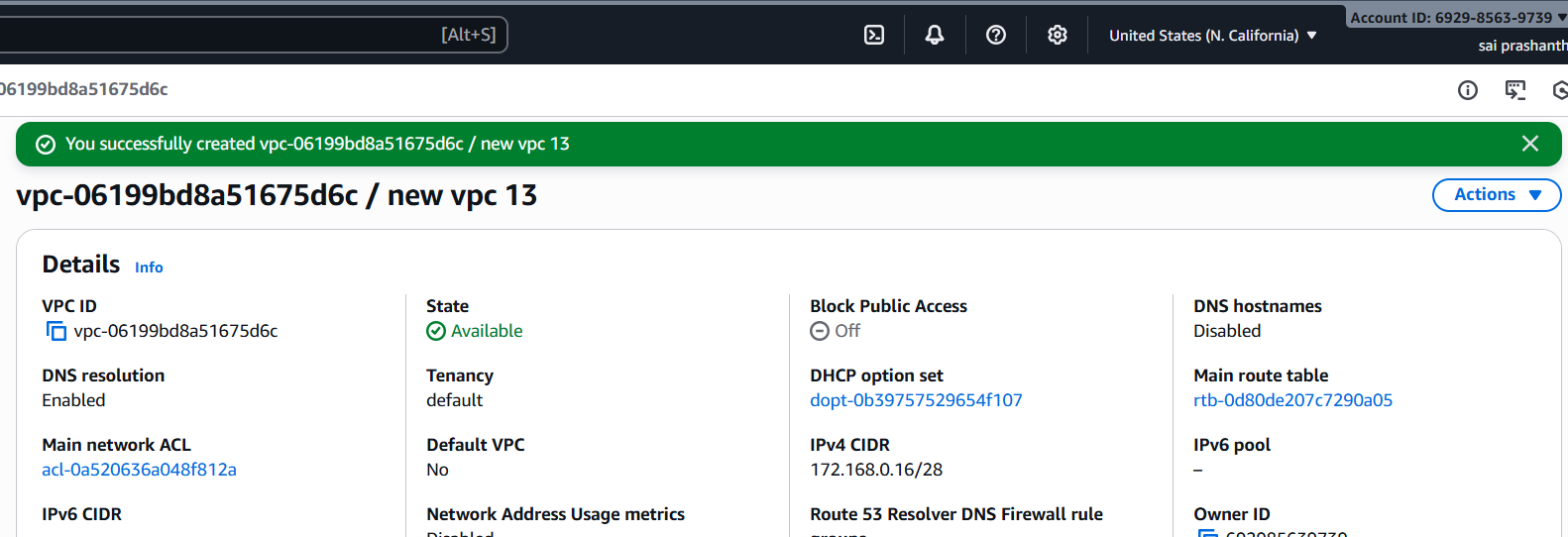
**Cloud Front and Route 53**

**Objective:** To design and implement a secure and scalable cloud networking and content delivery solution by configuring cross region VPC peering, deploying a static website on Amazon S3 enabling global content delivery using Amazon cloud front with SSL and mapping a custom domain purchased from Go Daddy through Amazon Route 53 ensuring successful domain-based access and connectivity validation.

1. **Configure VPC peering in cross regions.**

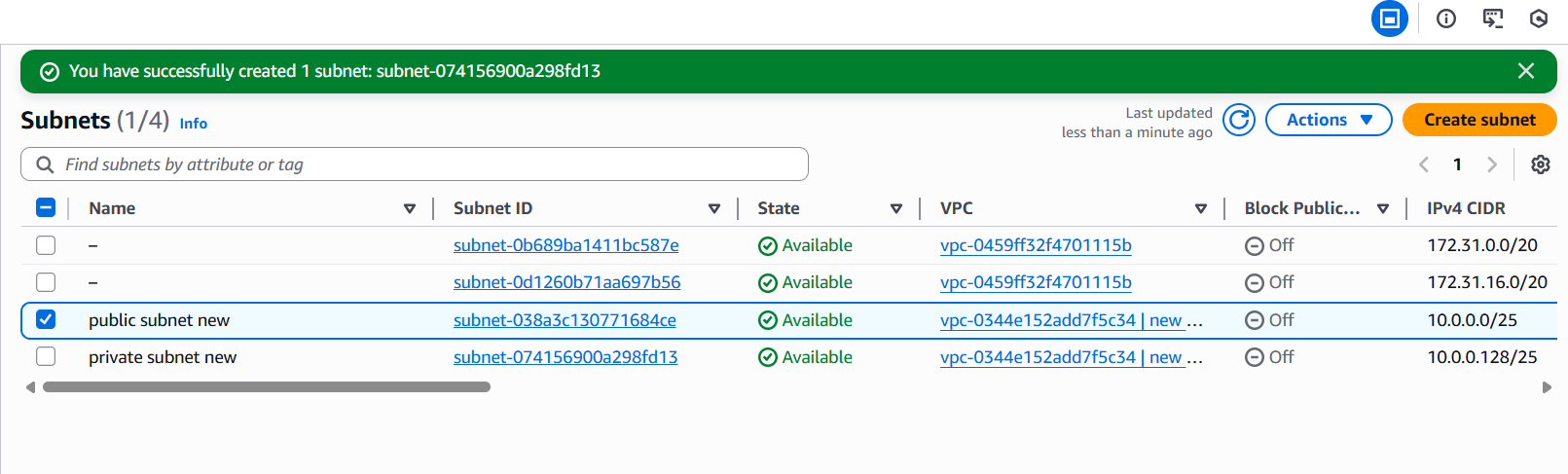
first created two vpc’s in 2 different regions



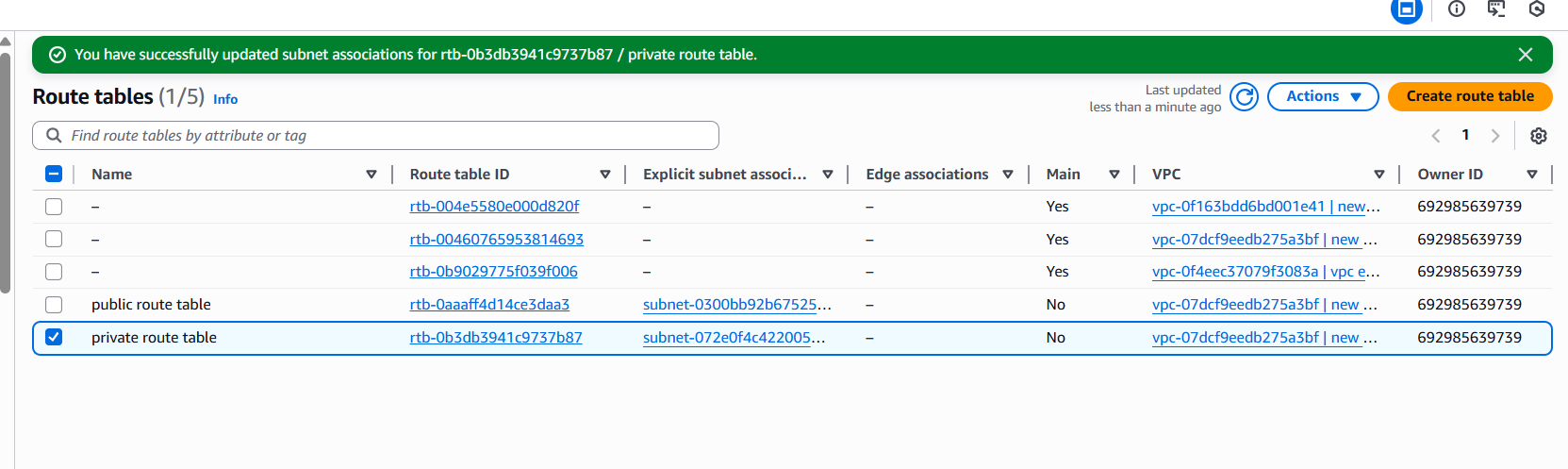


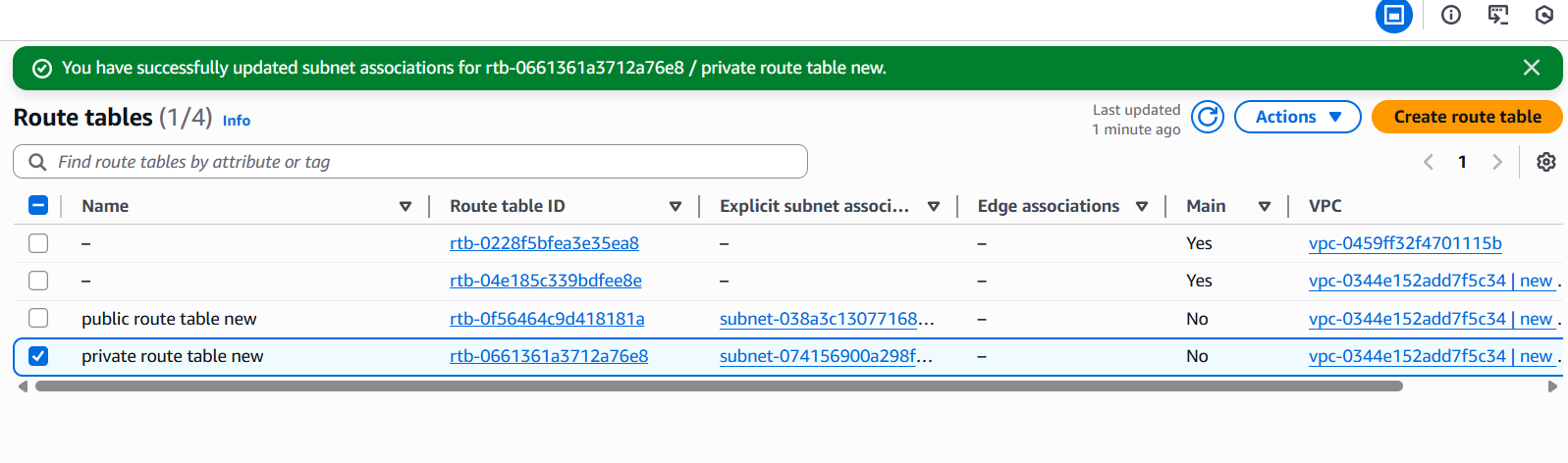
now created 2 subnets in both regions



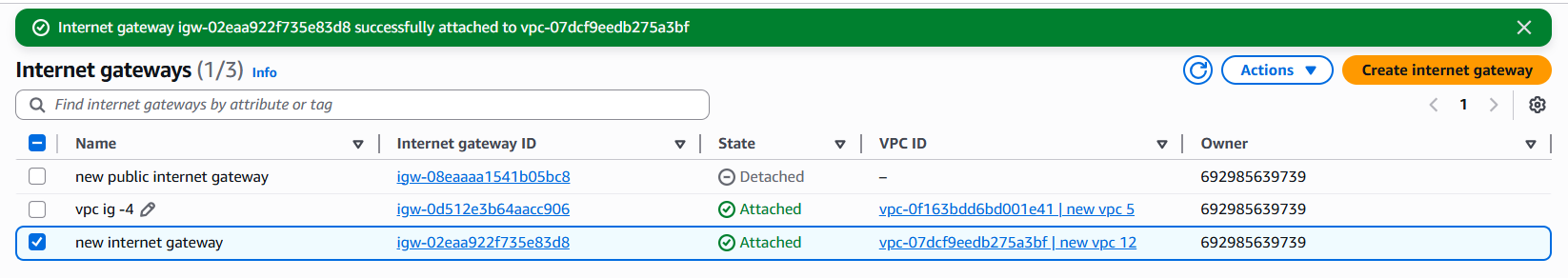


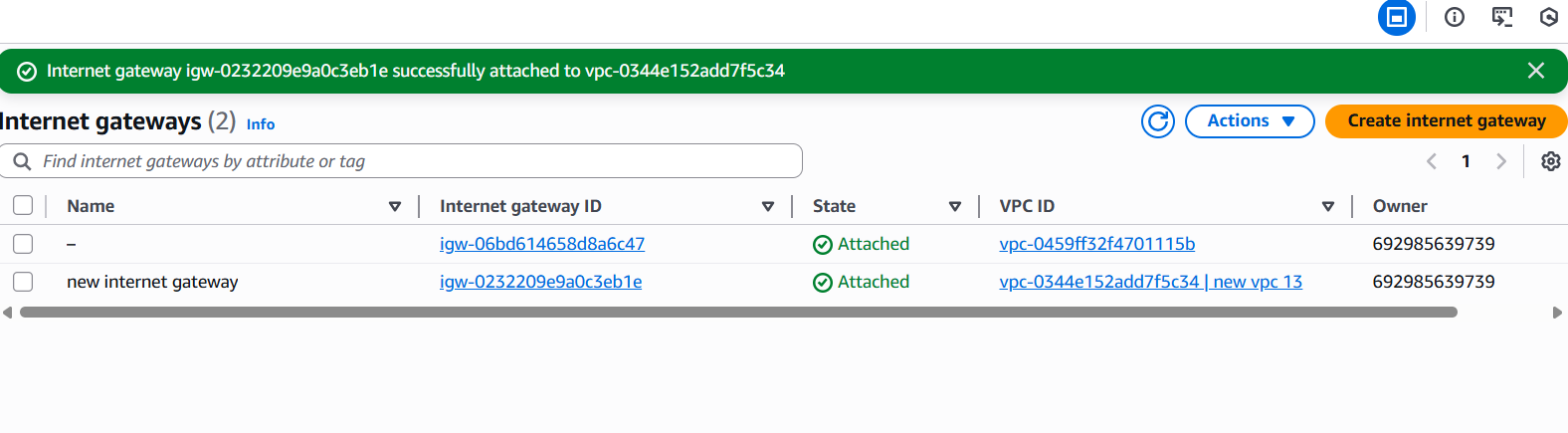
create two route tables public and pribate and add subnet associations accordingly





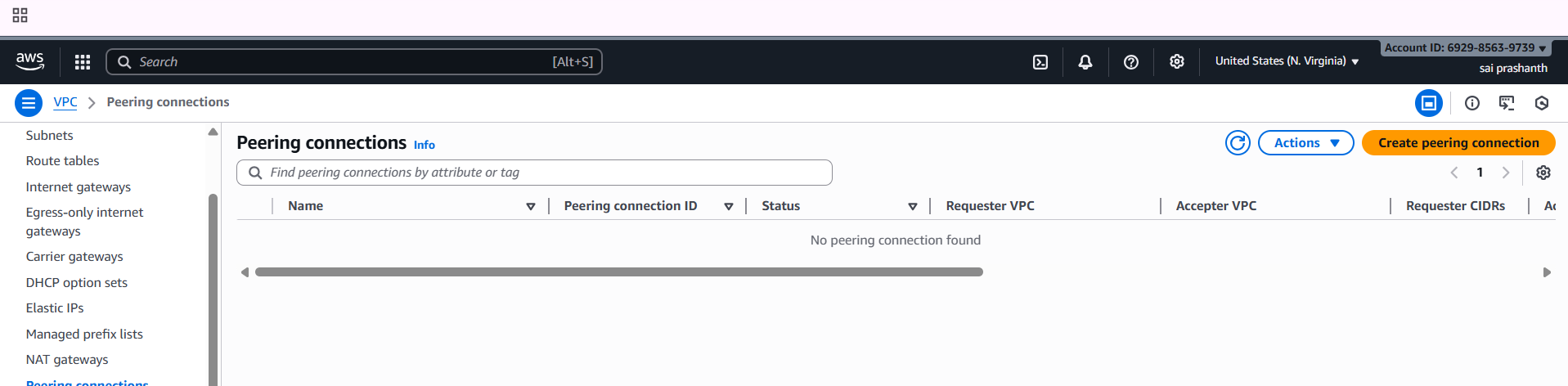
created internet gateway and attached to vpc in both regions and gave permissions to both oublic subnet in 2 regions



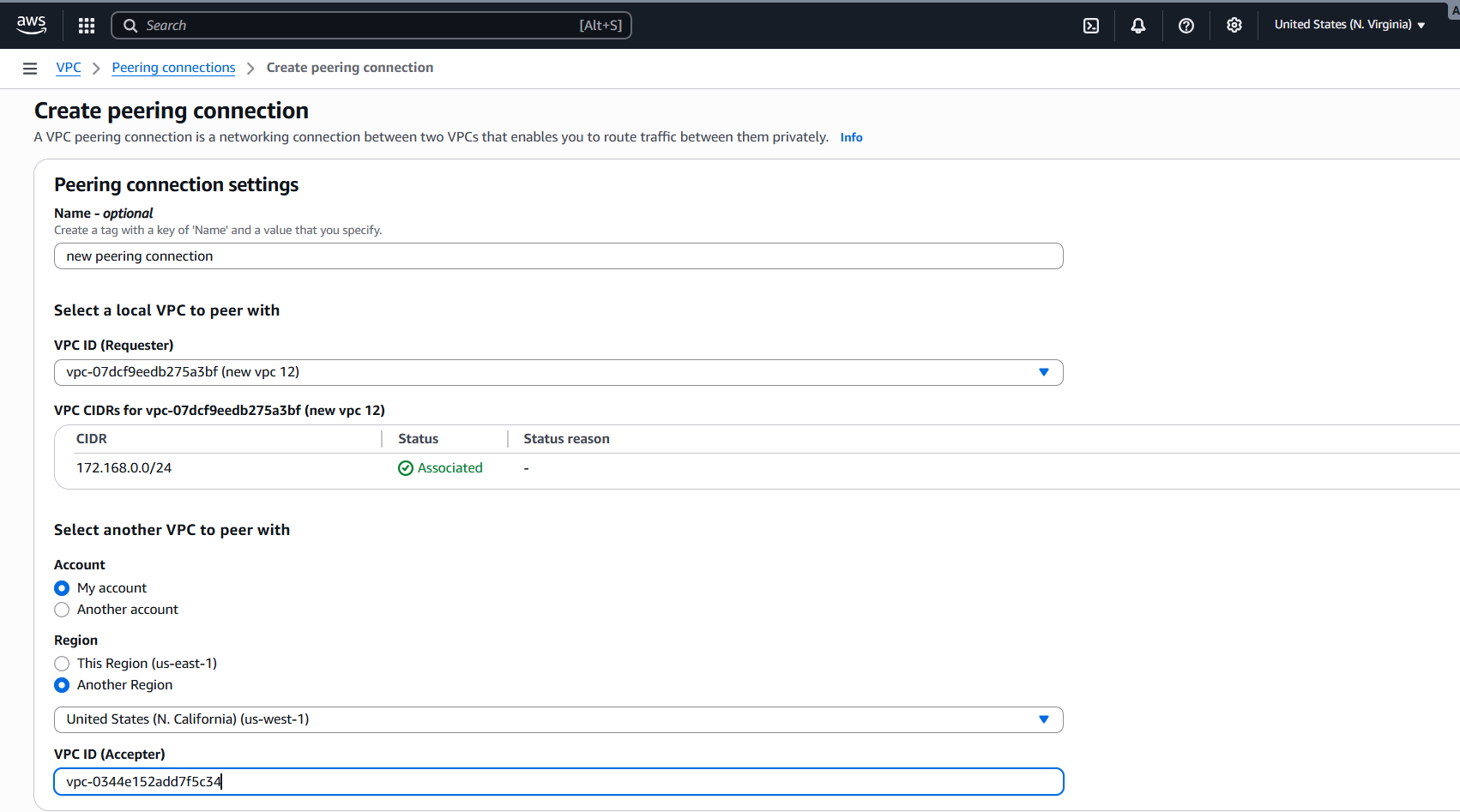


now we want to do peering connections

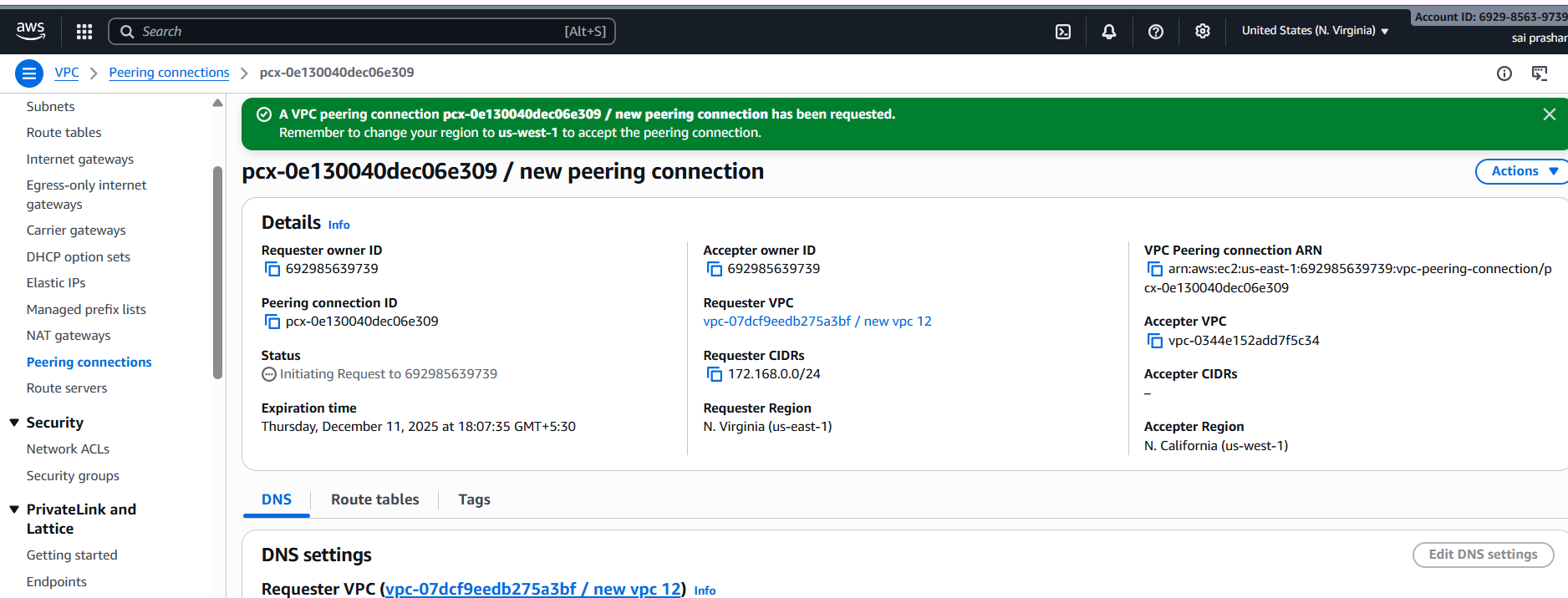
click on create peering connection



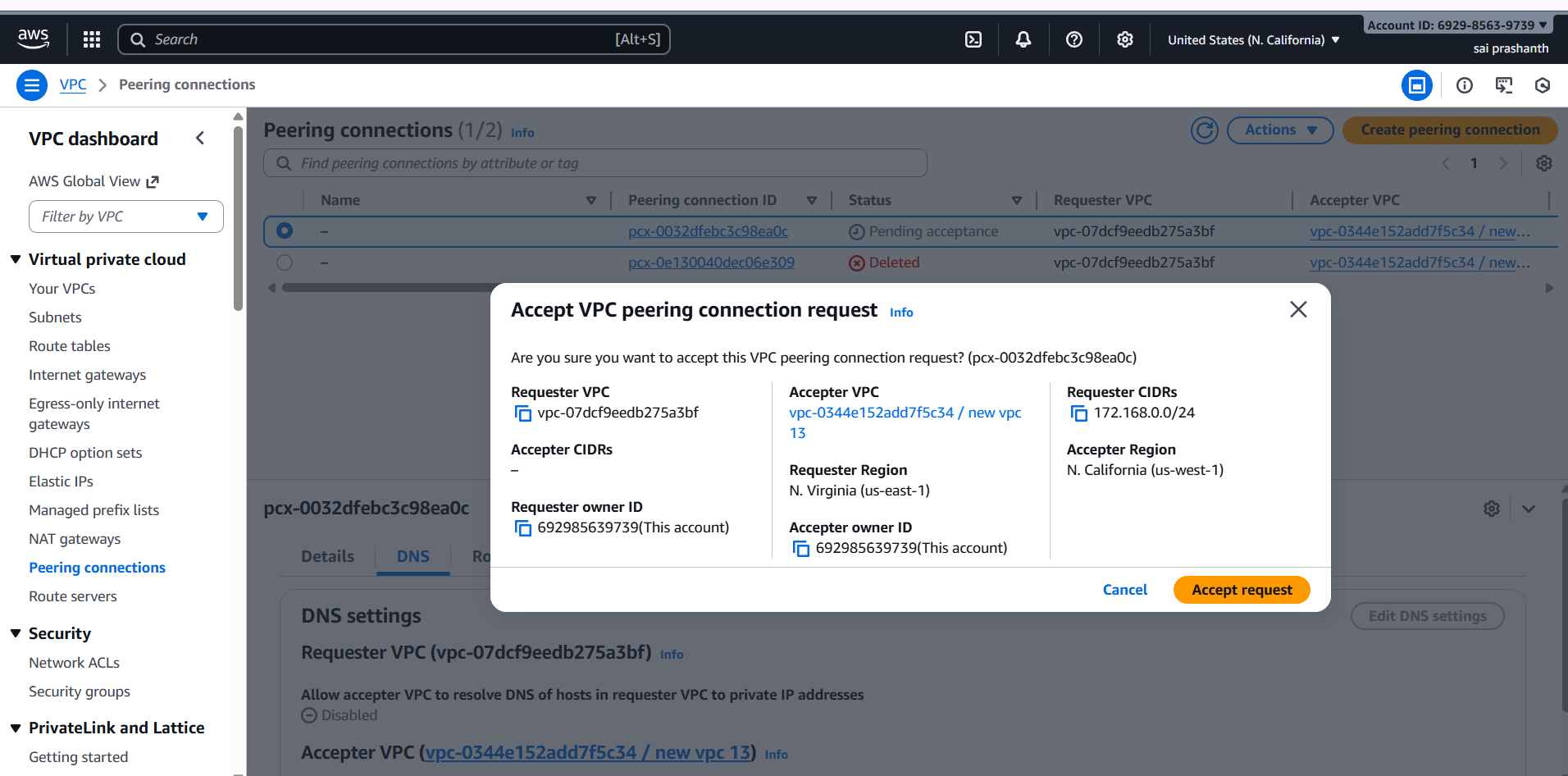
after clicking it enter the details of name select another region vpc and paste the vpc id



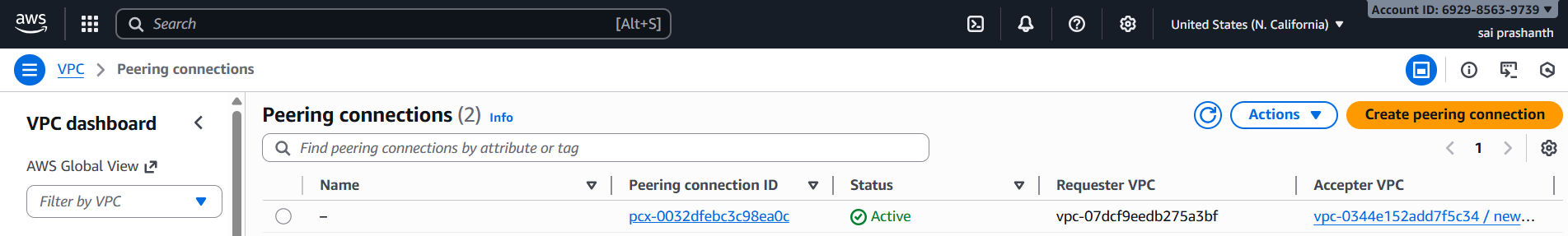
once done click on created, the peering connection has been created



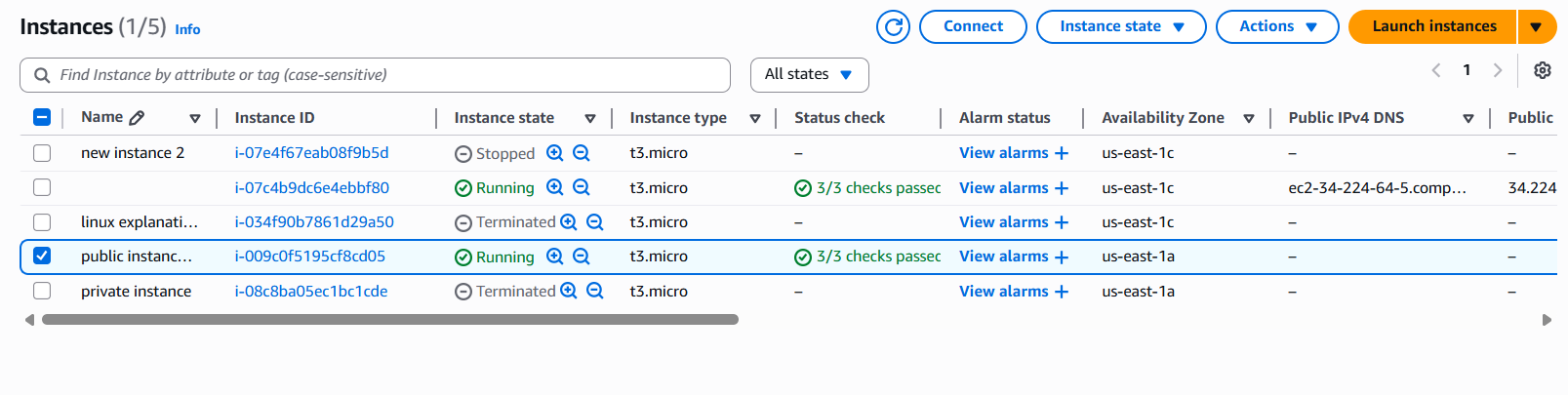
once done enable



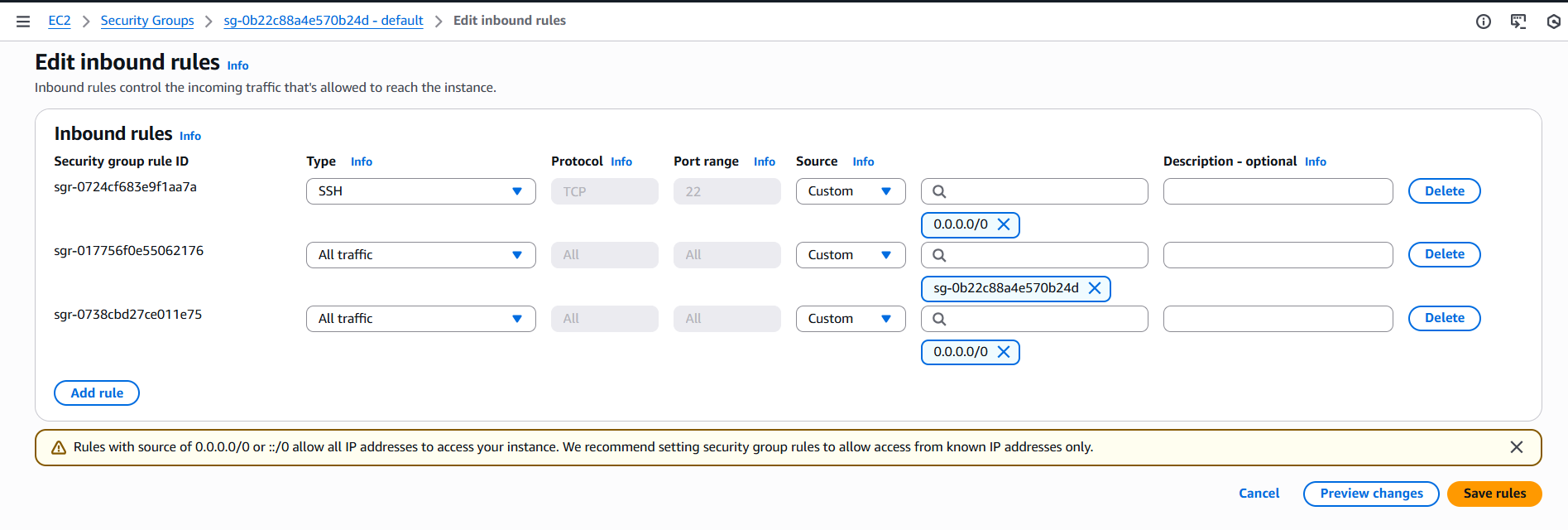
peering has been accepted in another region now…

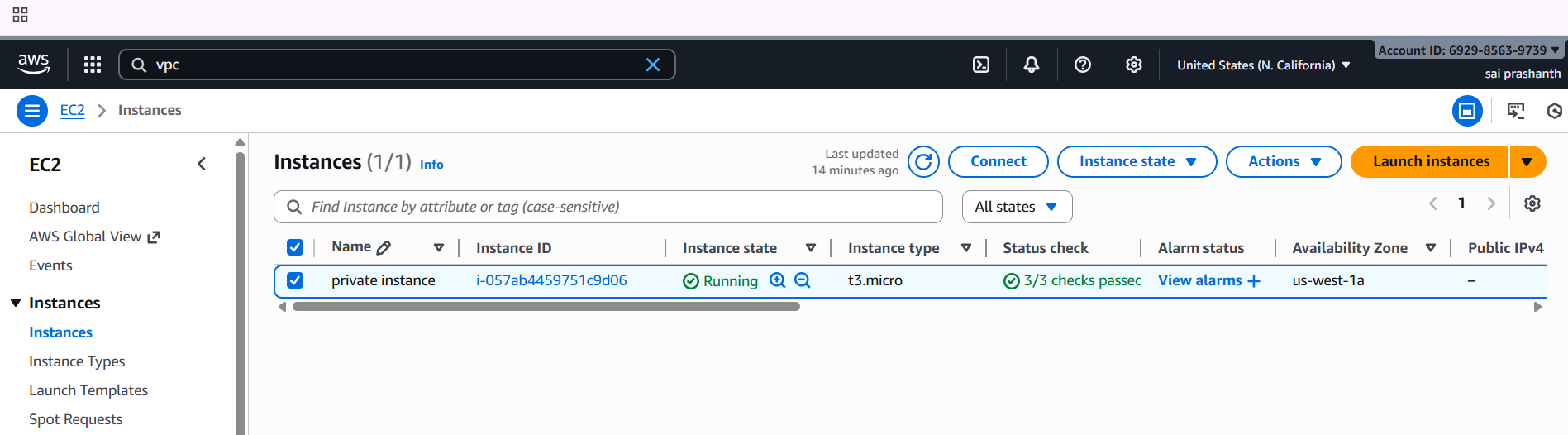


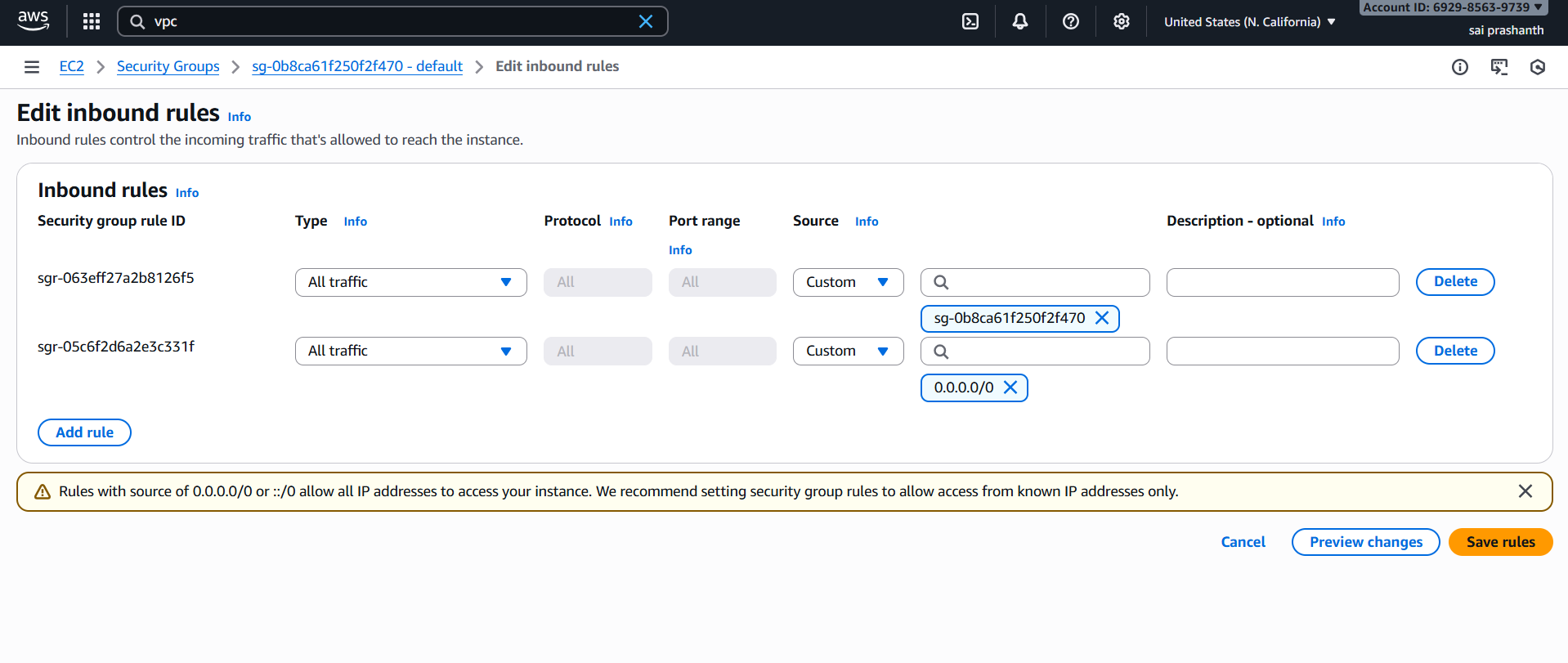
now create one public instance in one region another private instance in another region



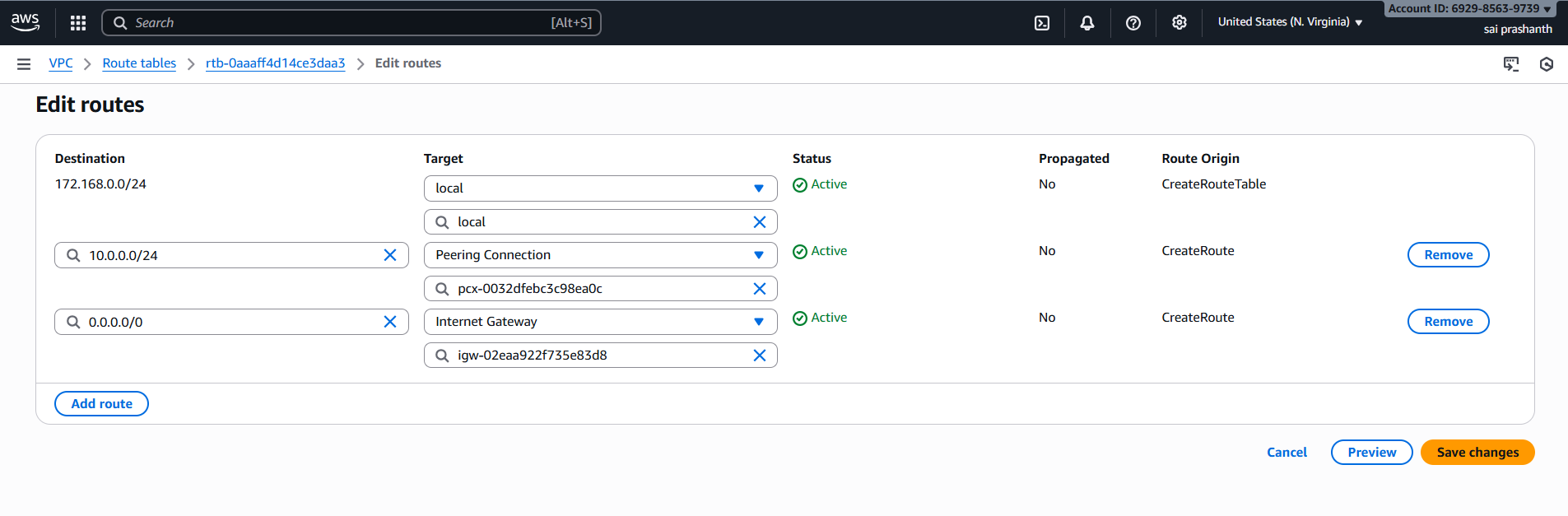
enable security groups





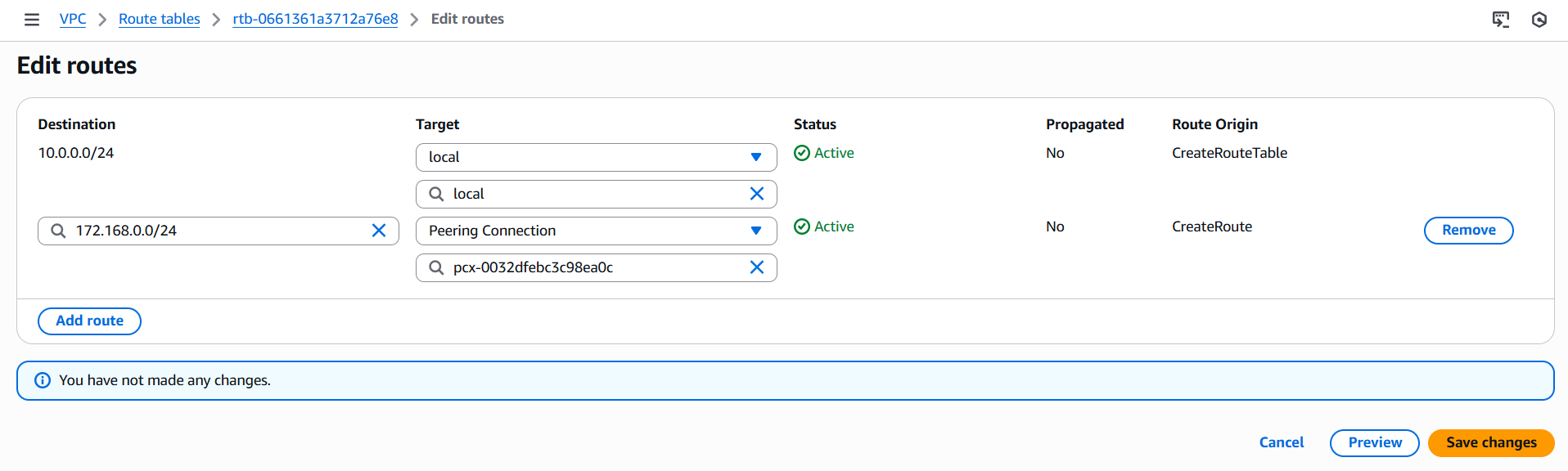


now in public instance which you have been created you need to edit the route table by adding the another instance vpc id and select peering connection



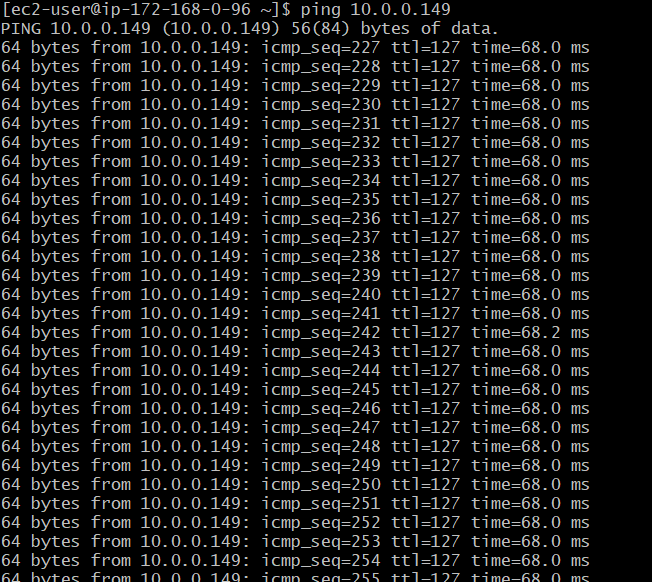
do the same in private instance now…

add the public instance vpc id and and add in private route table and peering connection and save



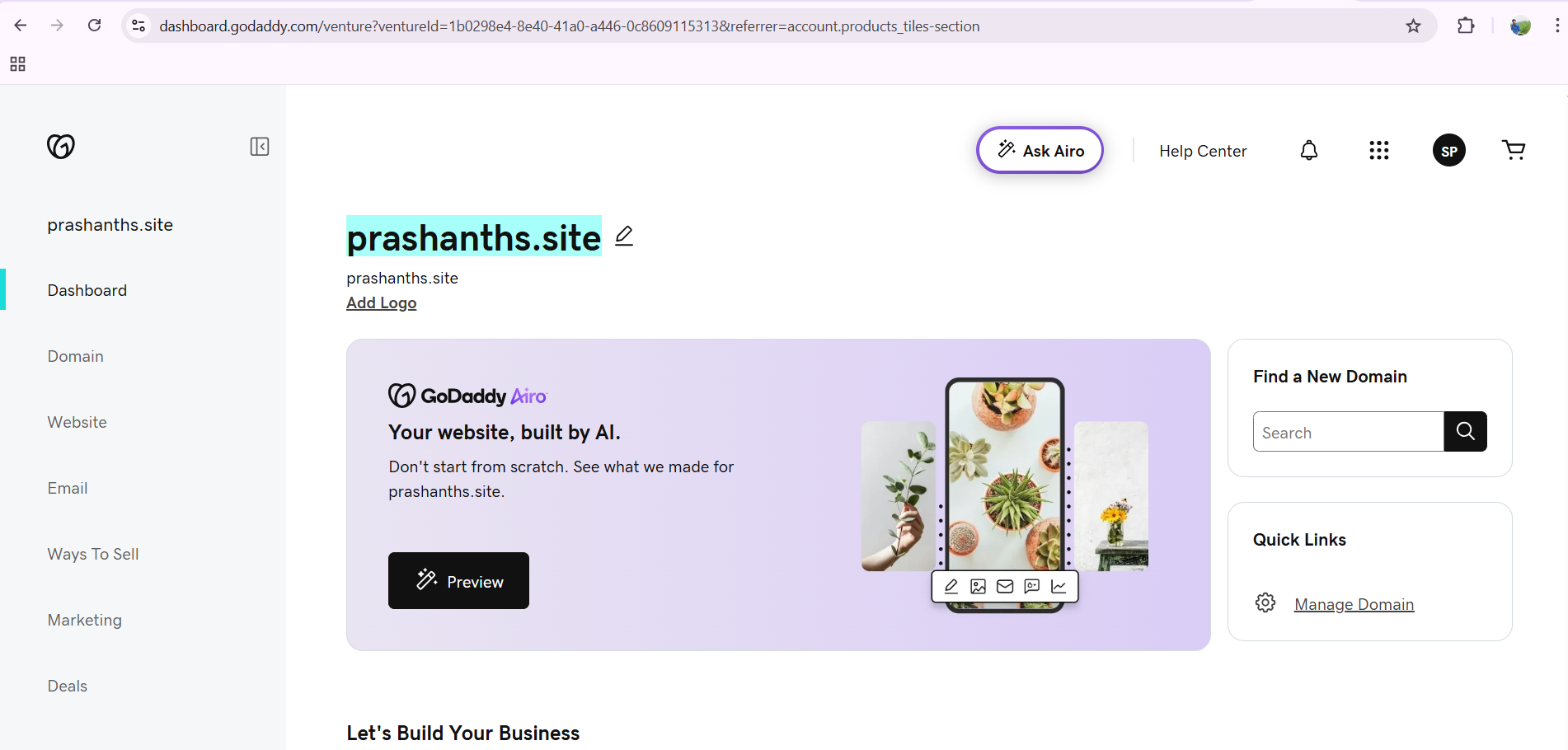
now in git bash connect and ping with private ip of another region

now the peering connection has been made



1. **Purchase one domain from GoDaddy.**

**Domain name : prashanths.site**

****

1. **Deploy static website in S3.**

first we need to create bucet

download any file from totoplate

and extract it

and now upload the objects

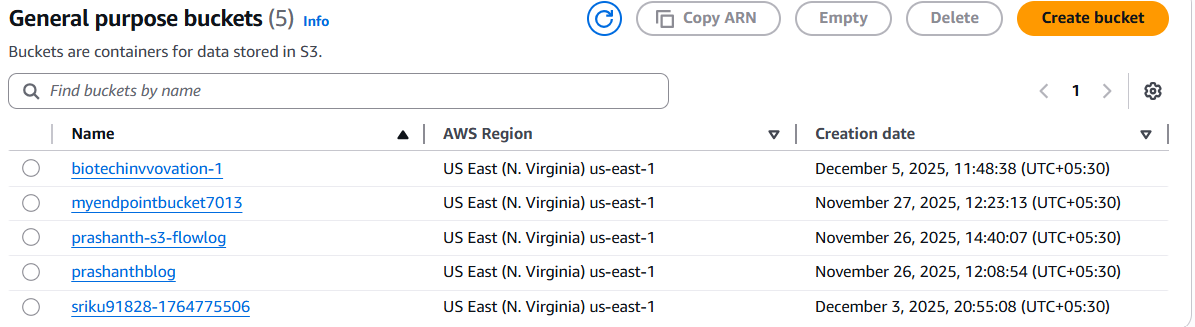
and in properties enable static website hosting

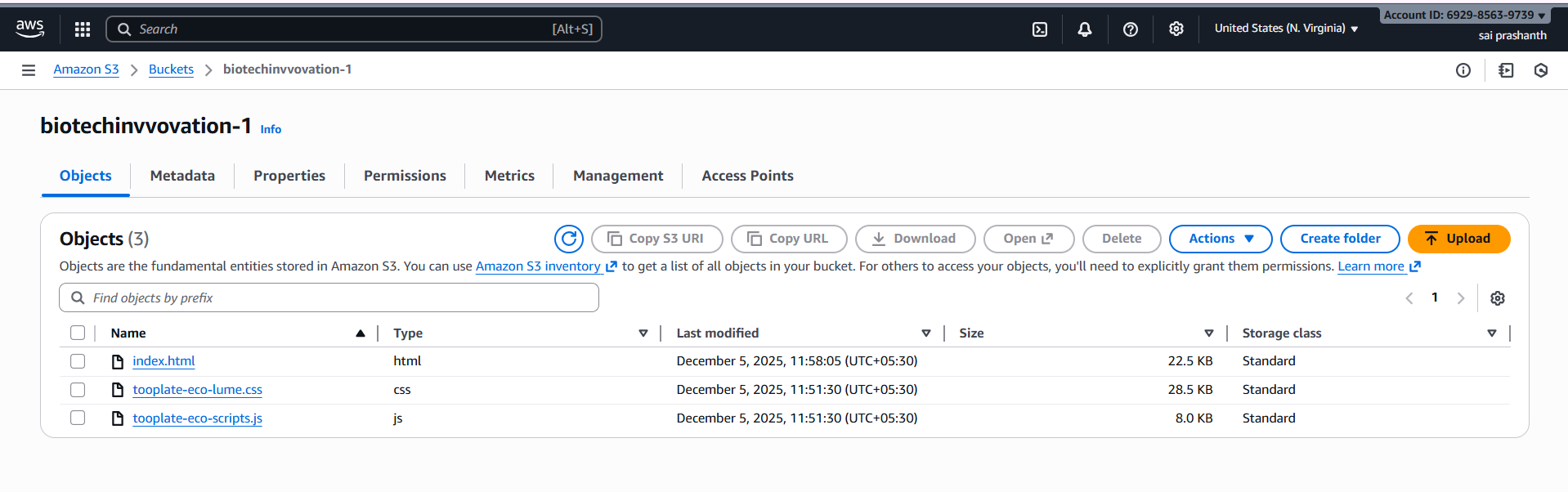
index.html

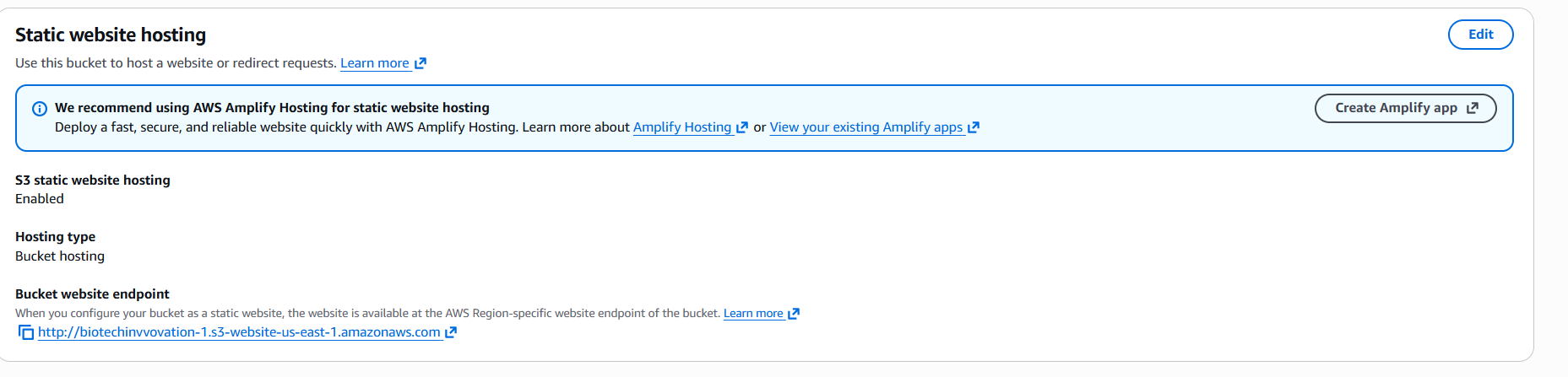
error.html

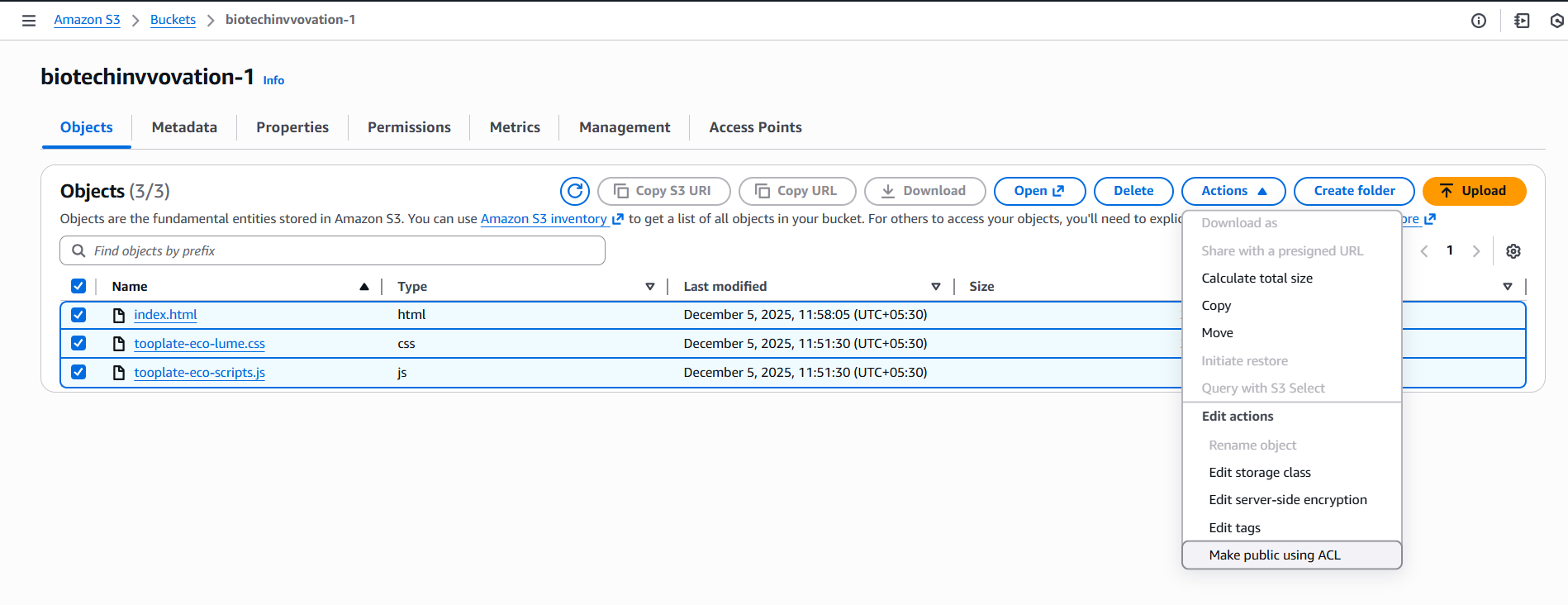
and now enable all the objects to public

now copy the link and paste it in your browser

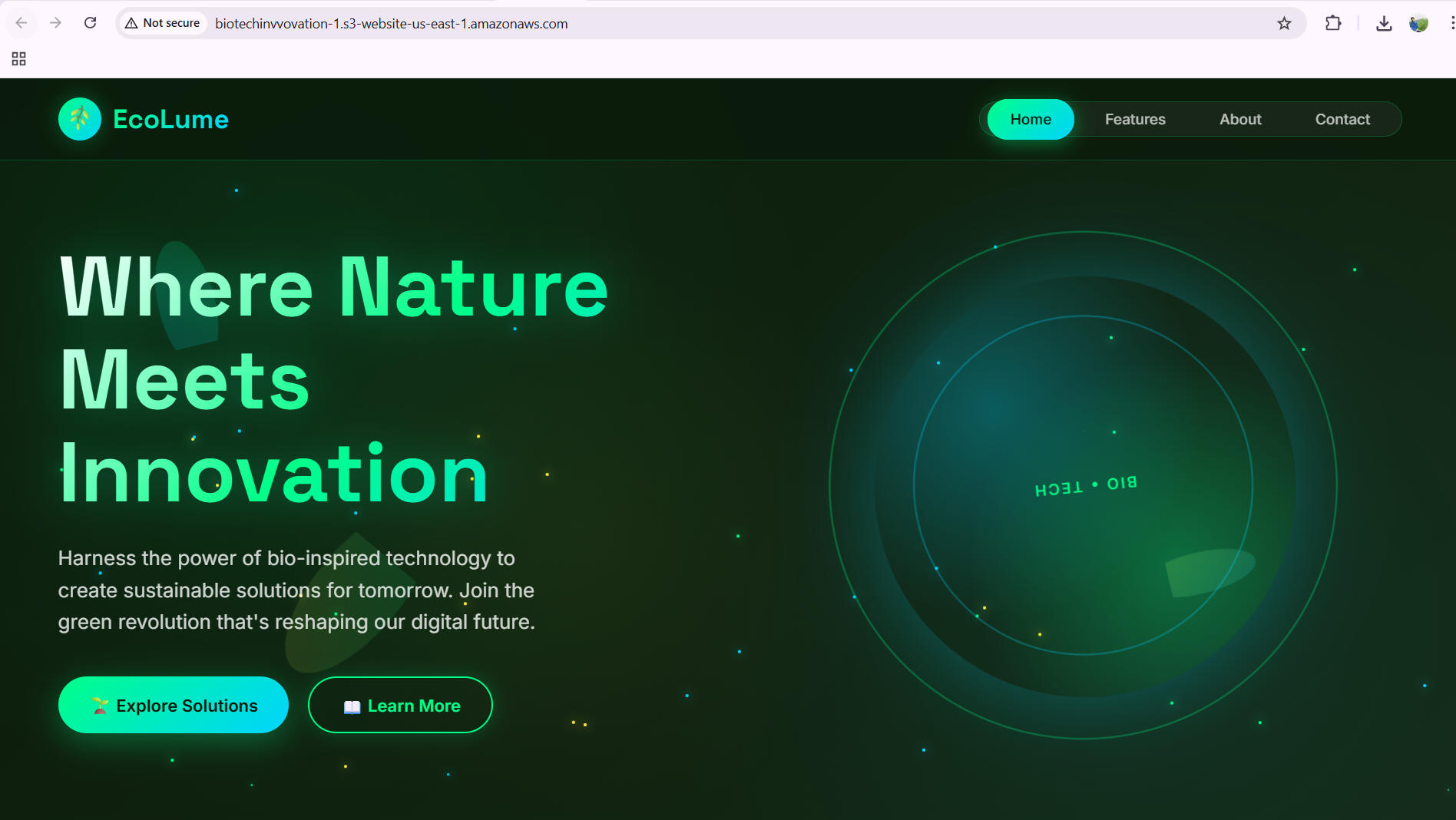










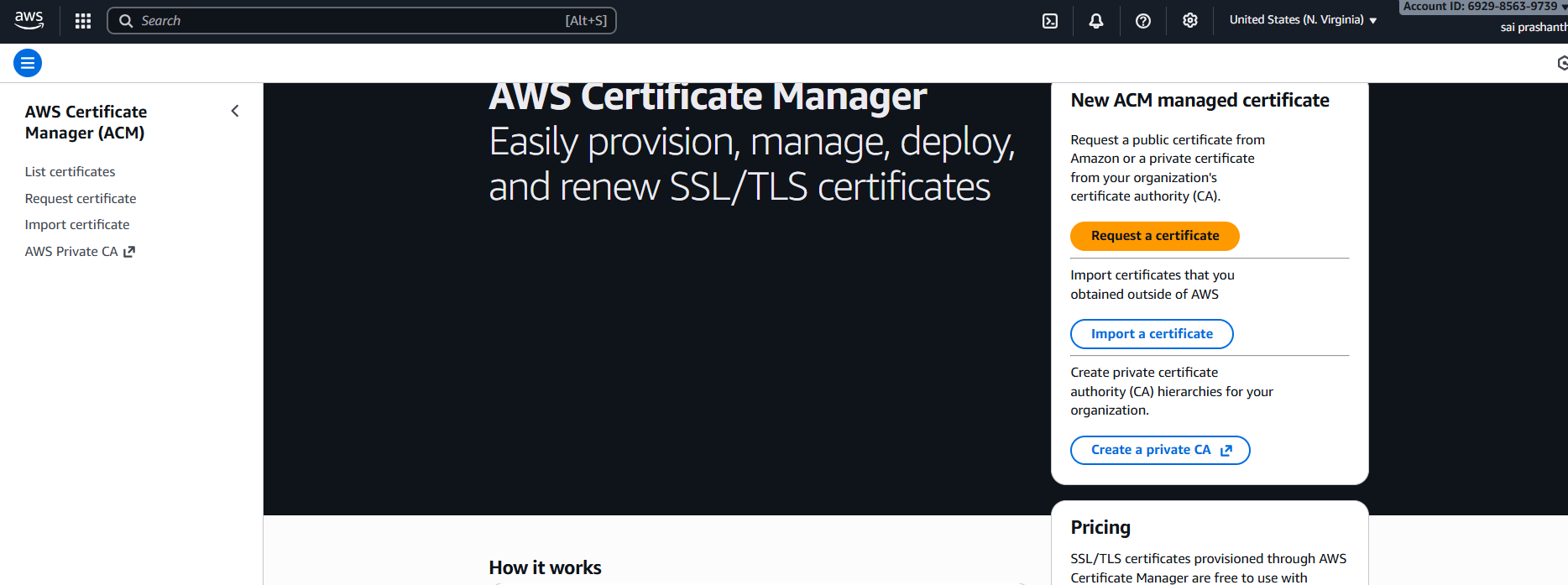


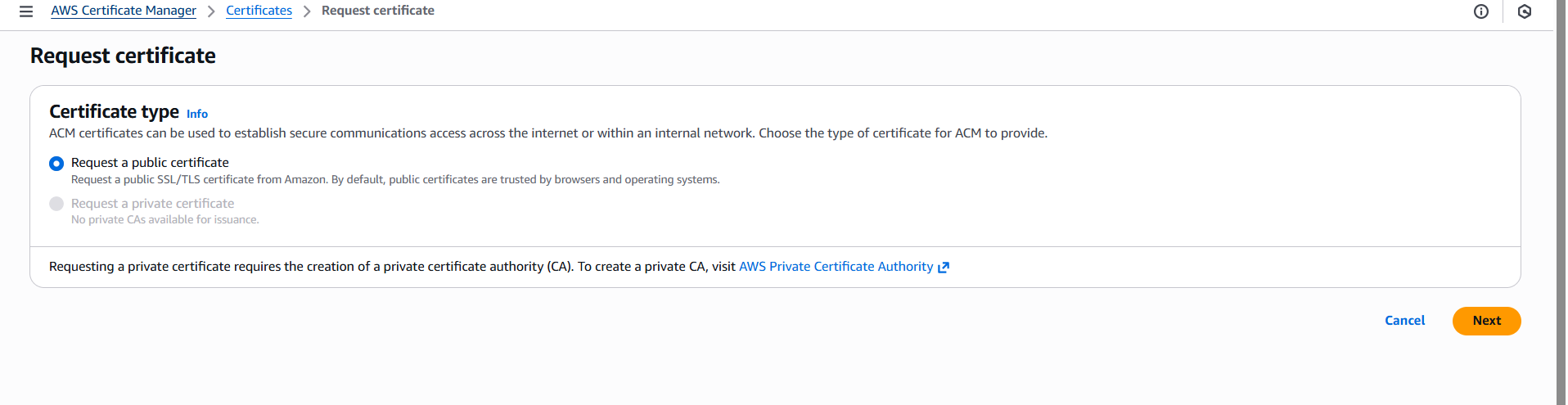
1. **Create a CDN and attach one SSL certificate.**

. Create a Route 53 hosted zone and map the domain with the CDN.

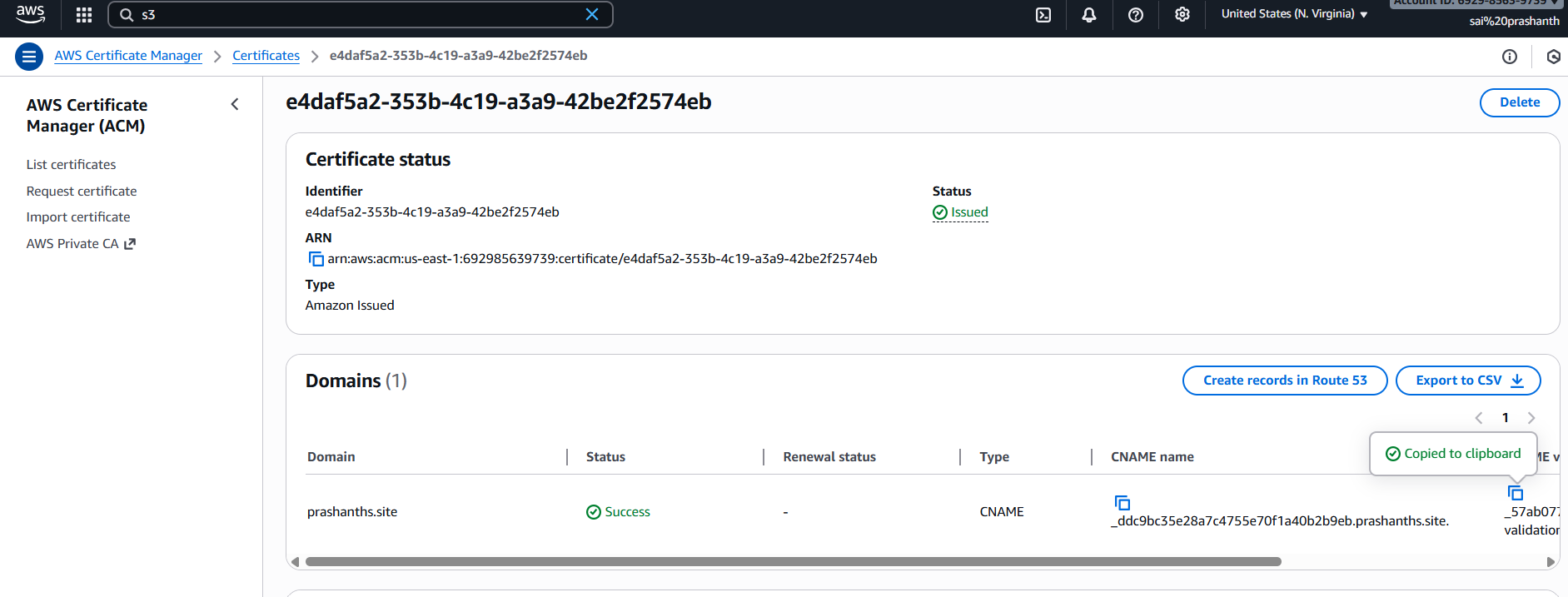
Update the index.html in the S3 bucket and ensure the updated file is accessible using the domain name.

first we need to create the certificate through ACM

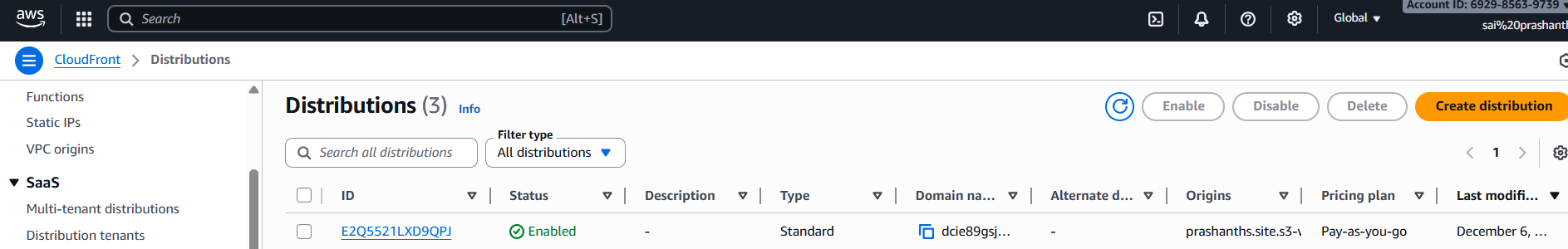
****

****

**now the certificate has been created**

****

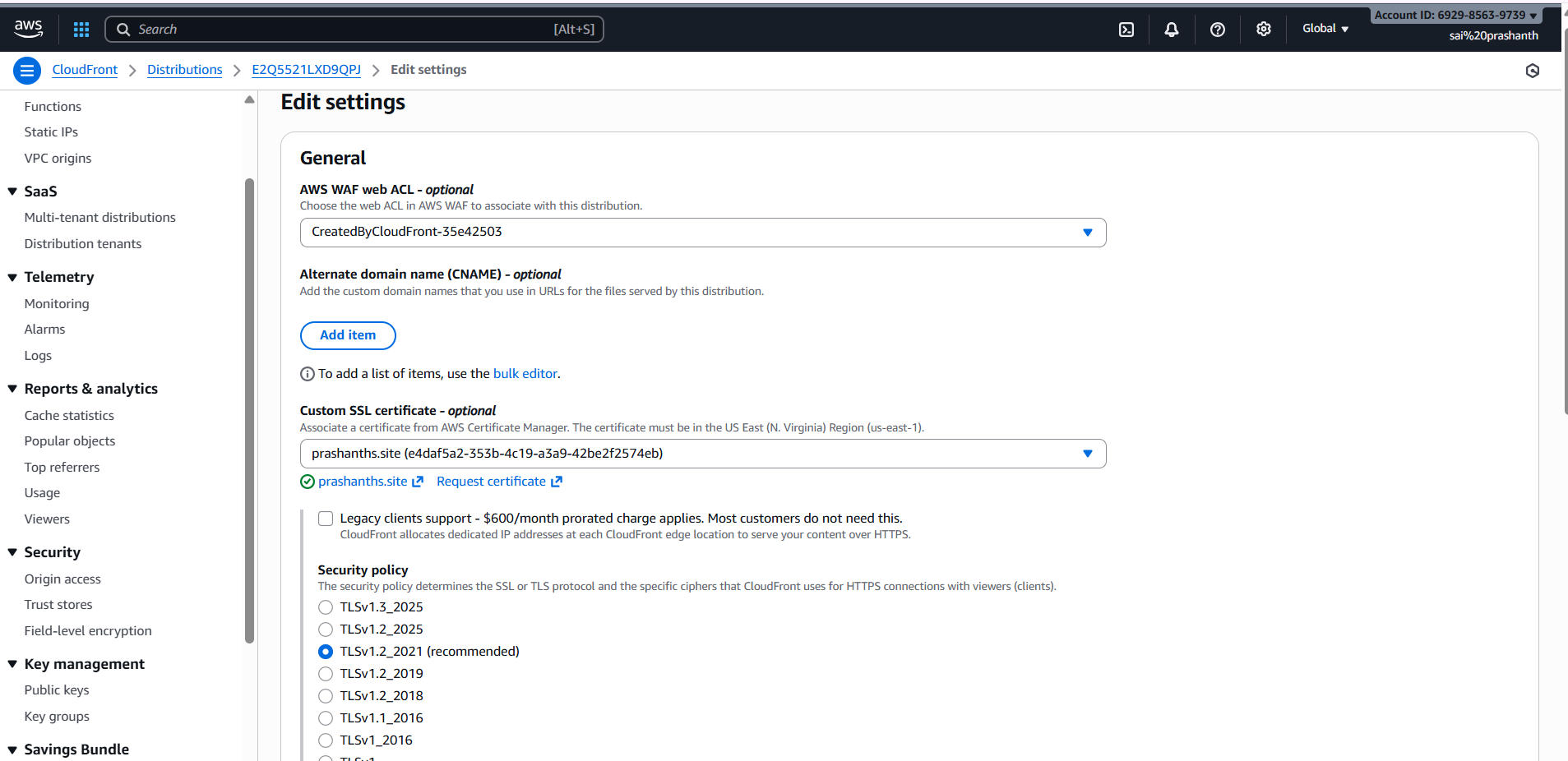
**after creating go to cloud front and create distribution**

****

**after clicking distribution, we need to add the certificate**

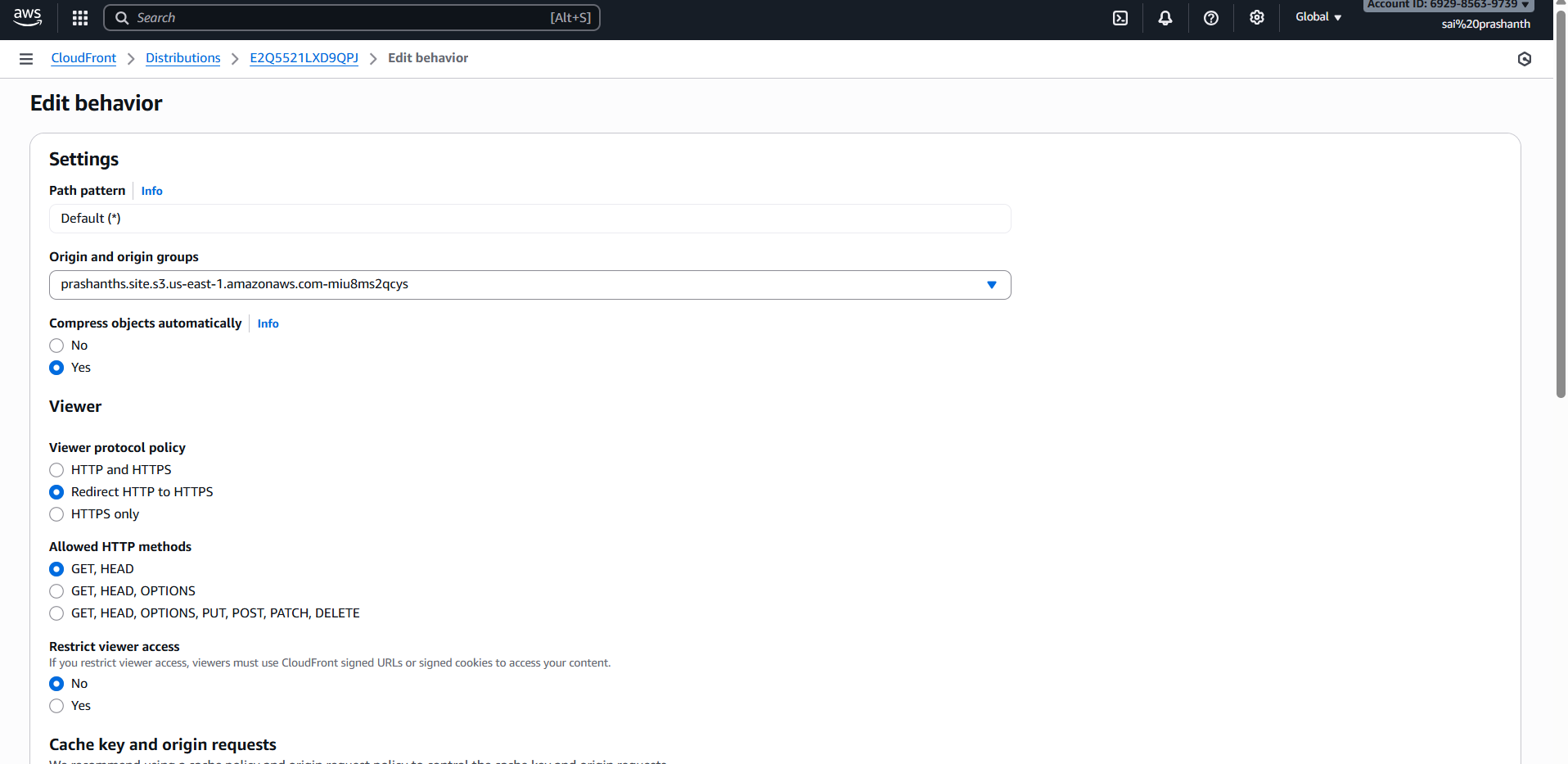
**we need to add security policy and supported http versions**

**we need to add the origin of S3 –it should be static website**

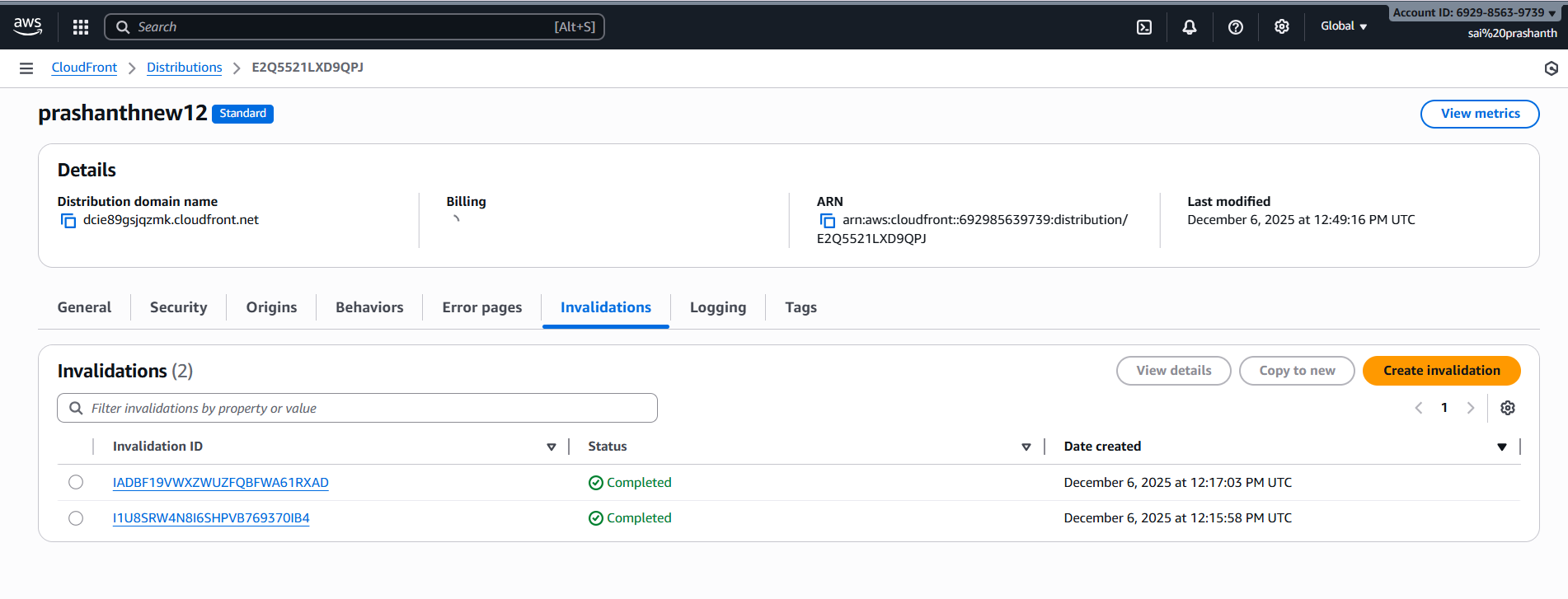
****

**and in behaviour we need to add the origin name ,protocol policy and allow http method**

**and we need to save**

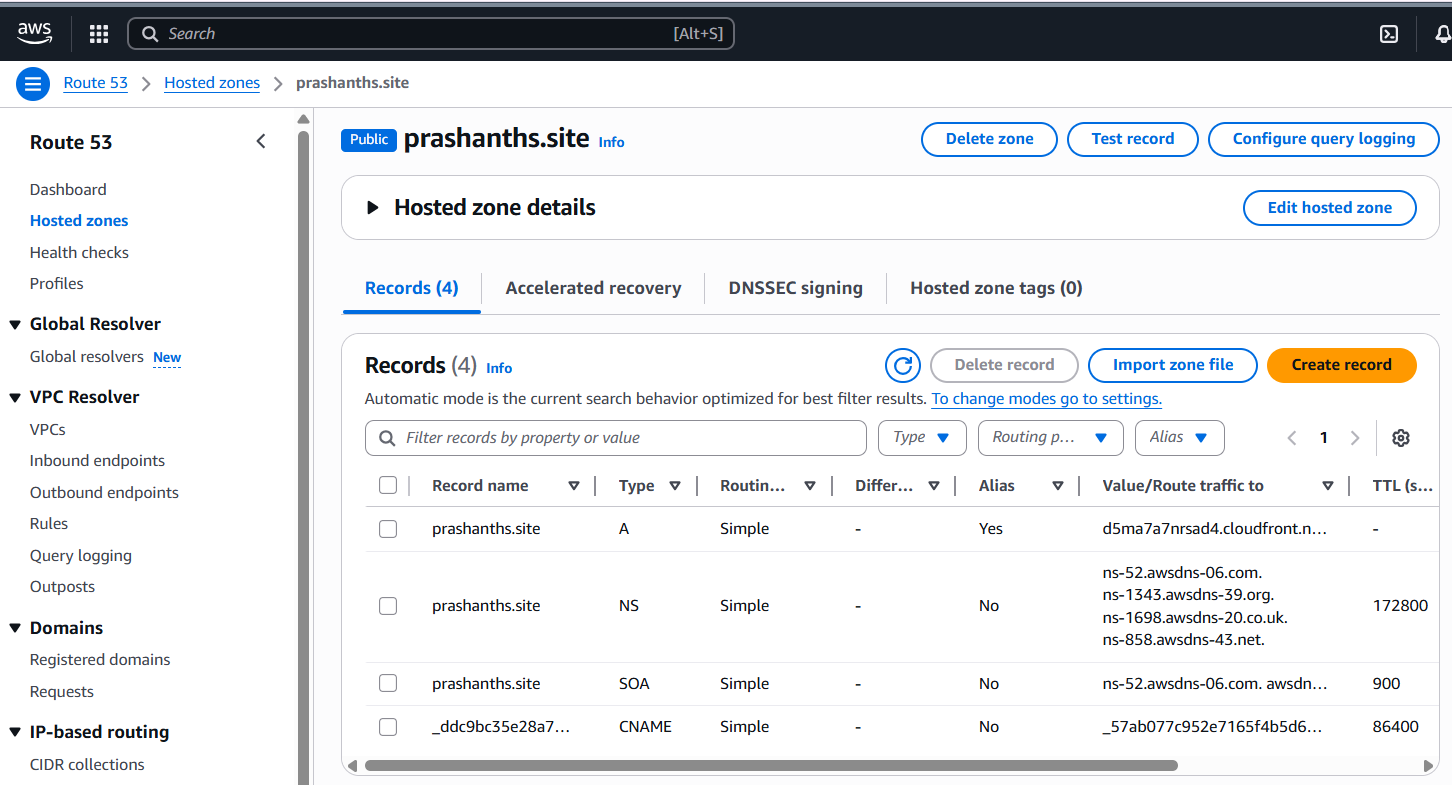
****

**and now we need to give invalidations with /index.html and /\***

****

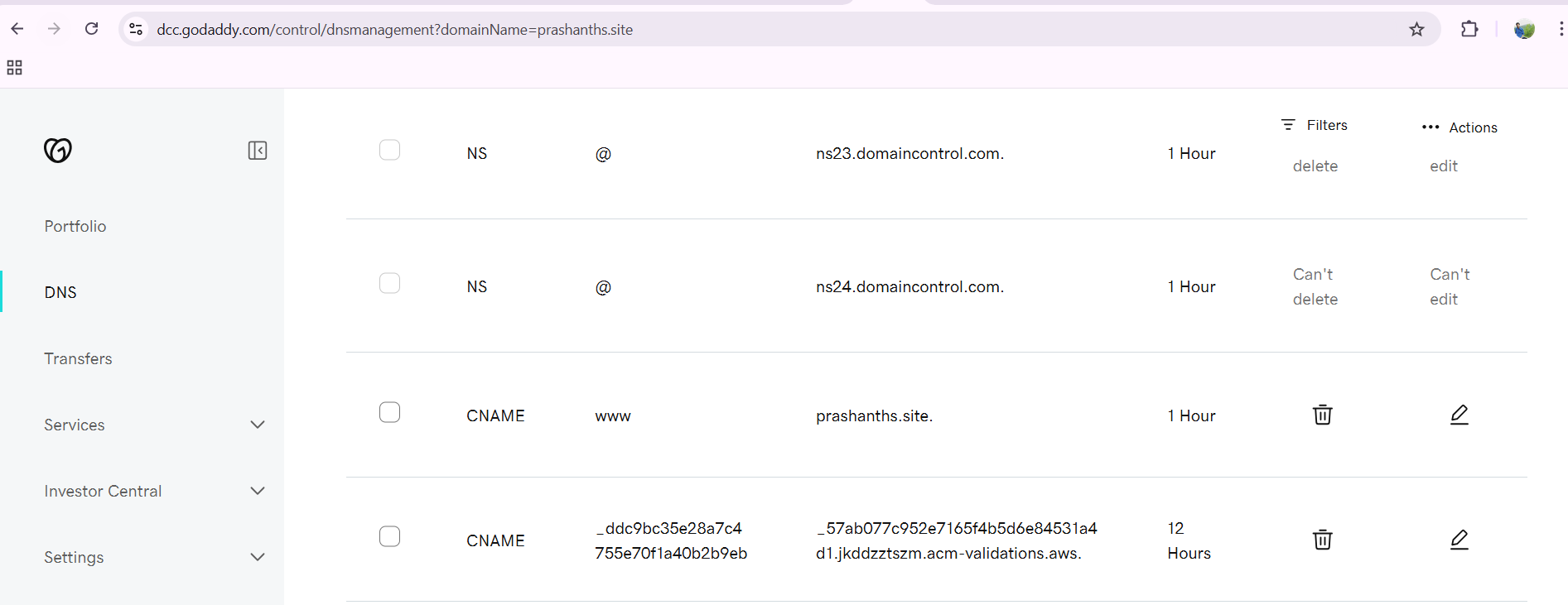
**go to route 53 –we need to create hosted zone by adding the zone name with same domain name what we have created**

**create record with c name ,A,AAA by adding the alias to region and distribution name which you have created**

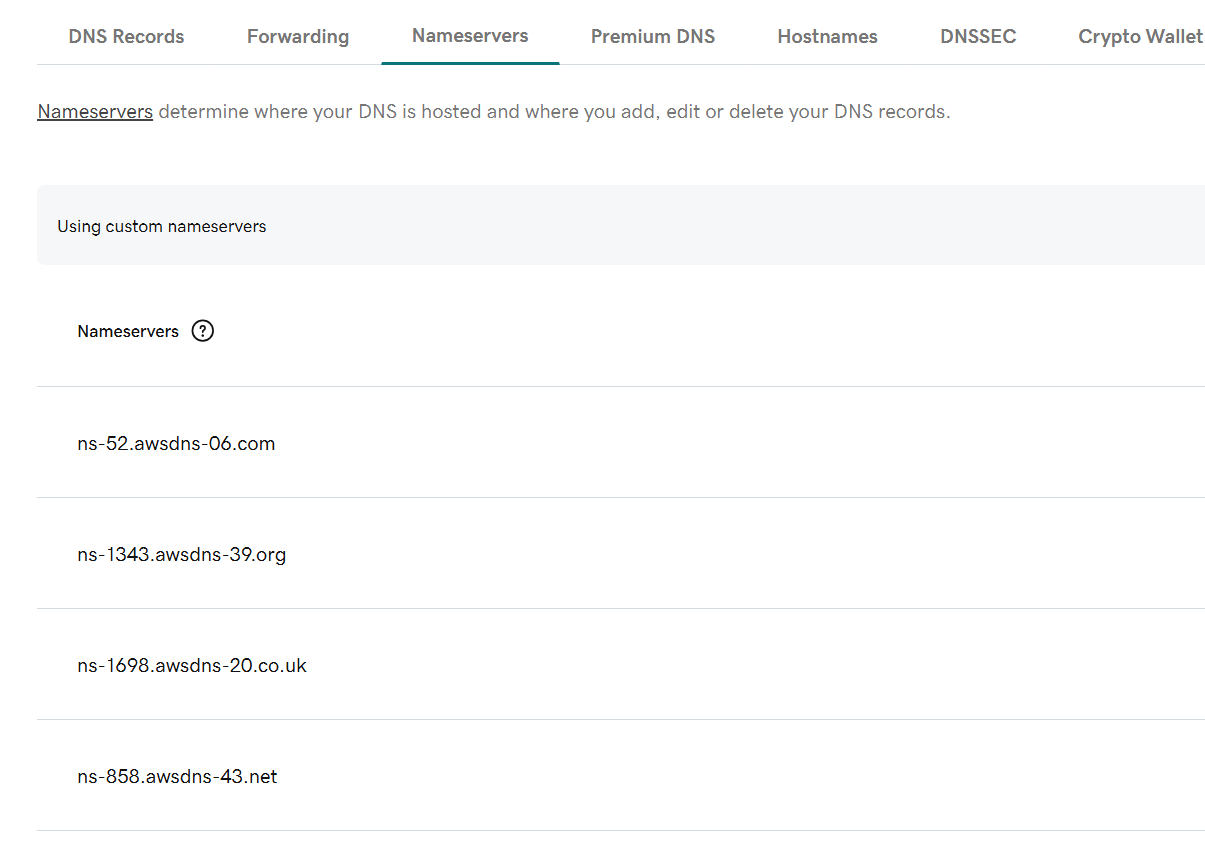
****

**now in go daddy we need to add the new record with cname and value**

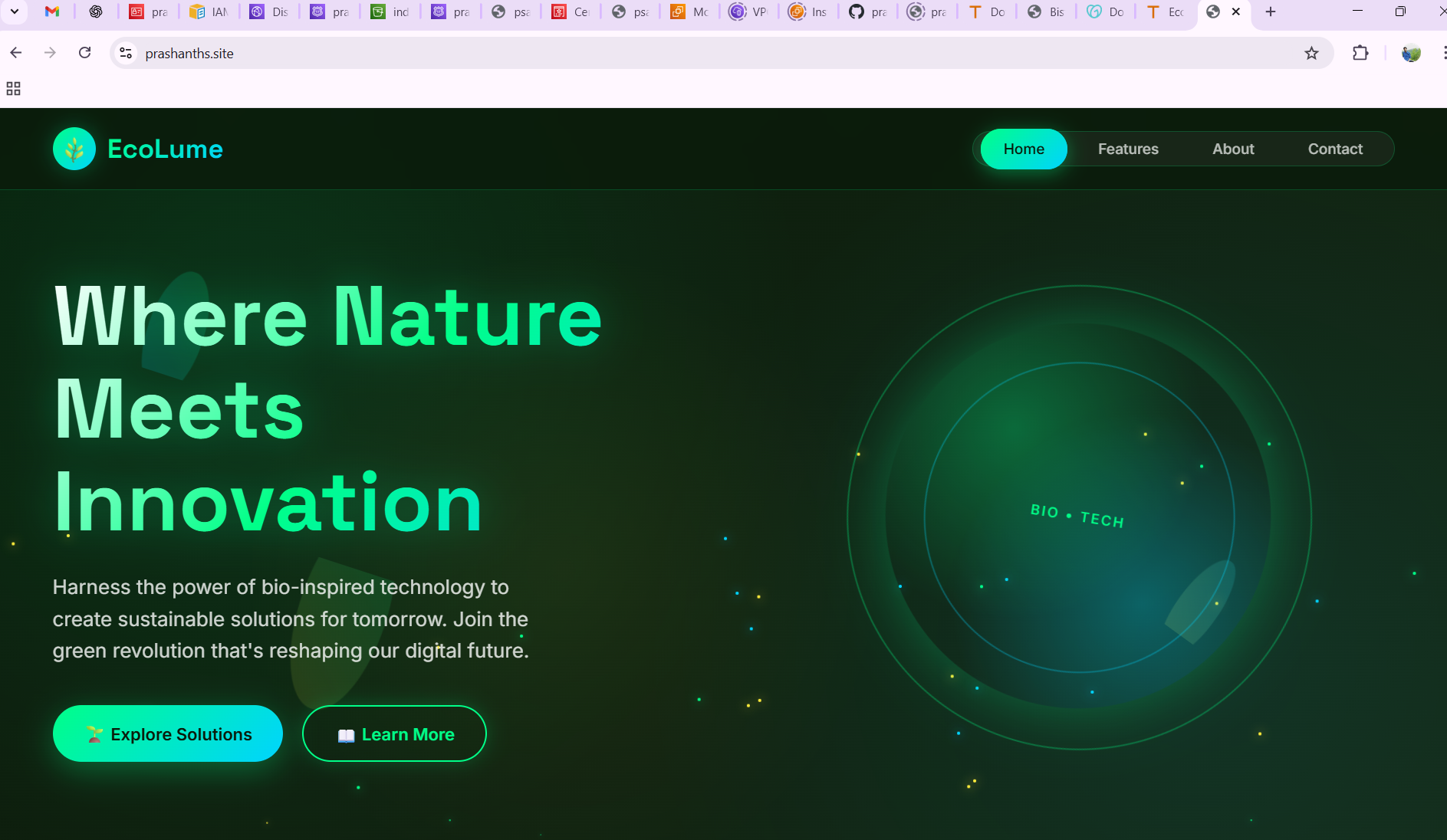
**and we need to add the link with www and domain name**

****

**and name servers in the go daddy**

****

**now try to refresh the browser and check with domain**



**domain name: prashanths.site**

**Note: sometimes browser is not accepting some domain names due to the issue with go daddy**