Application load balancers:

The Application Load Balancer is **a feature of Elastic Load Balancing that allows a developer to configure and route incoming end-user traffic to applications based** in the AWS public cloud. ... Application Load Balancer works in Layer 7 of the OSI reference model for how applications communicate over a network.

**Getting started with Application Load Balancers**

**Before you begin.**

**Step 1: Configure your target group.**

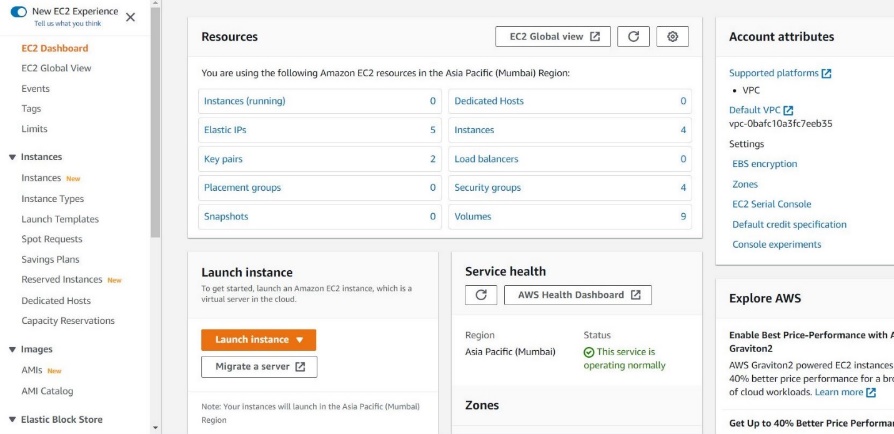
**Step 2: Choose a load balancer type.**

**Step 3: Configure your load balancer and listener.**

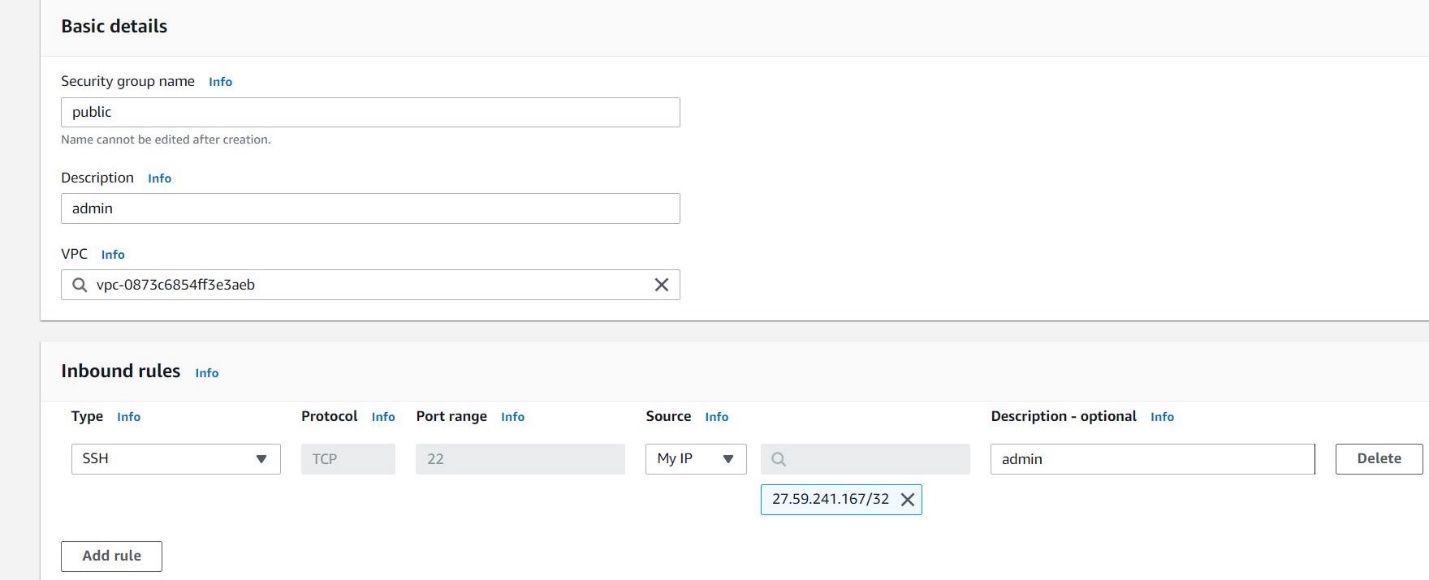
**Step 4: Test your load balancer.**

**Step 5: (Optional) Delete your load balancer.**

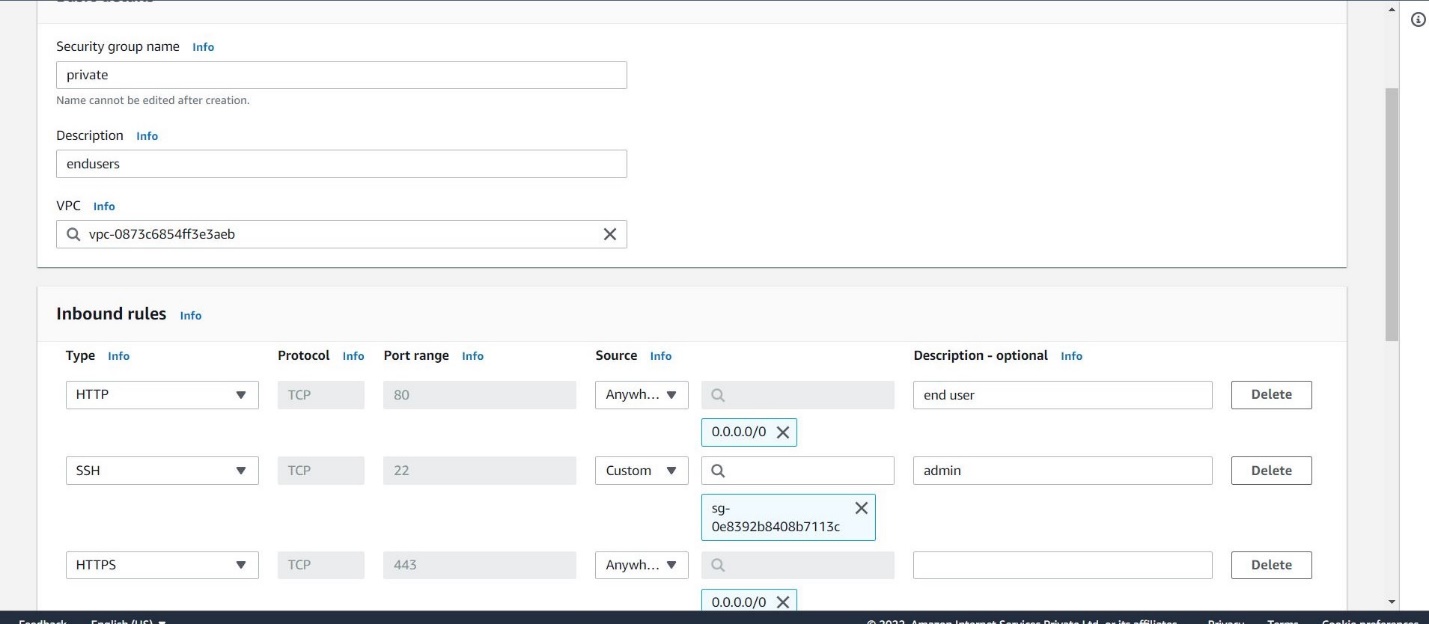
**Go to the Aws console here we create one security groups**

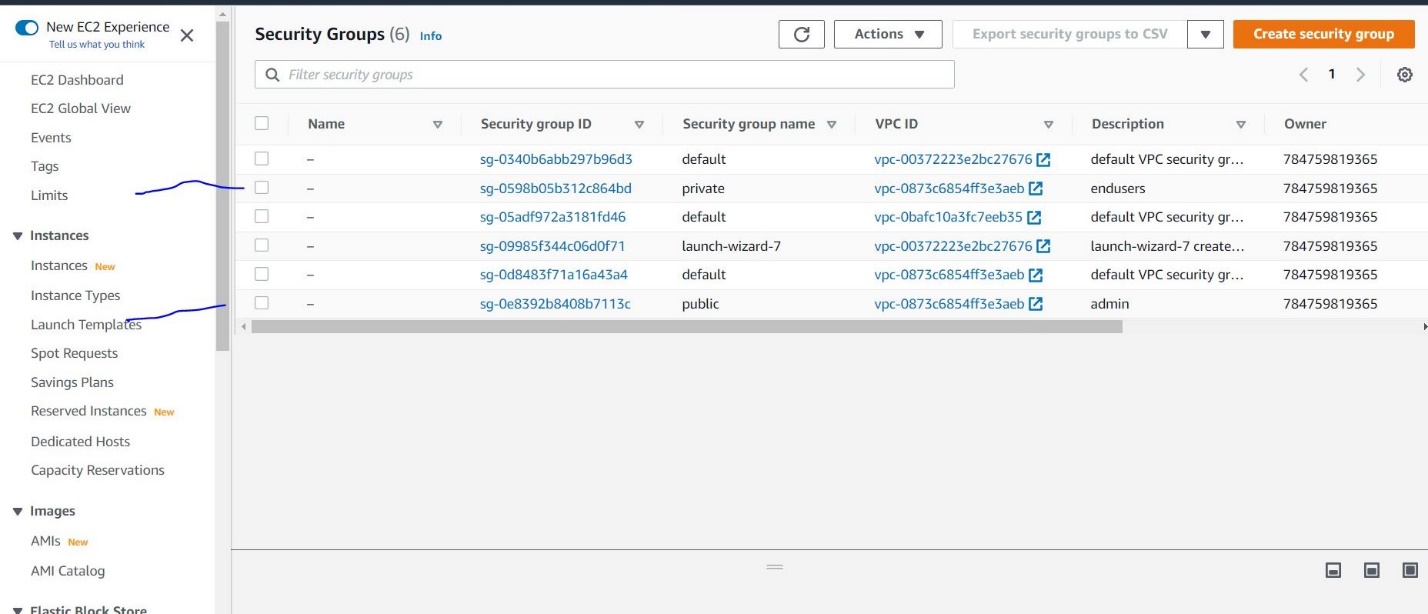


**Step 1 Now I am going to create one public security group with inbound rules**

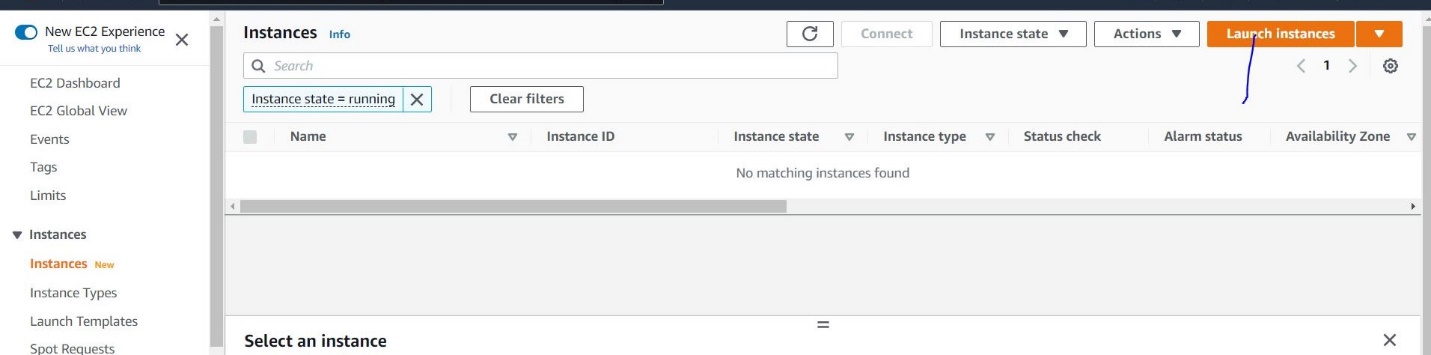
****

**Step 2 Now I am going to create one private security group with inbound rules**

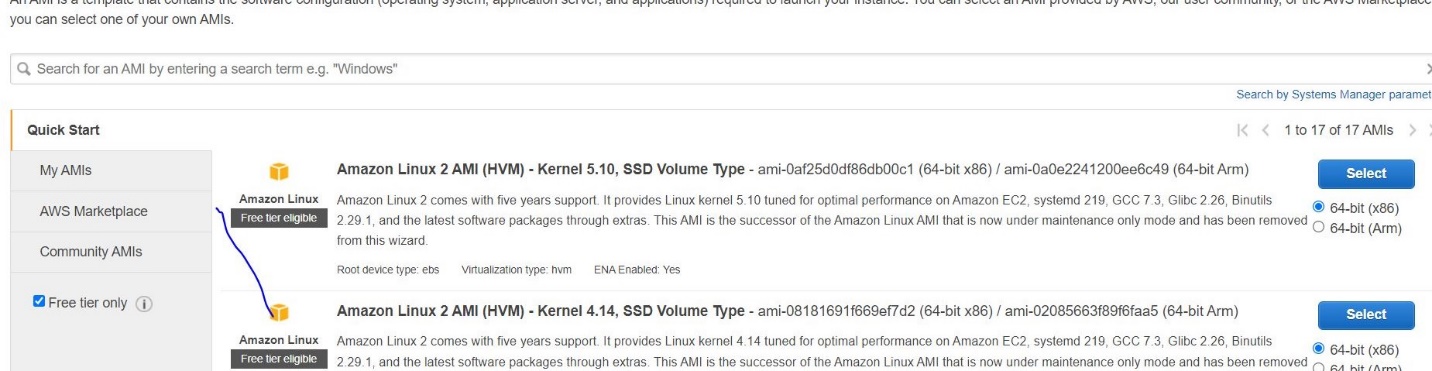
****

**Steps Here we can see the private subnets and public subnets created**

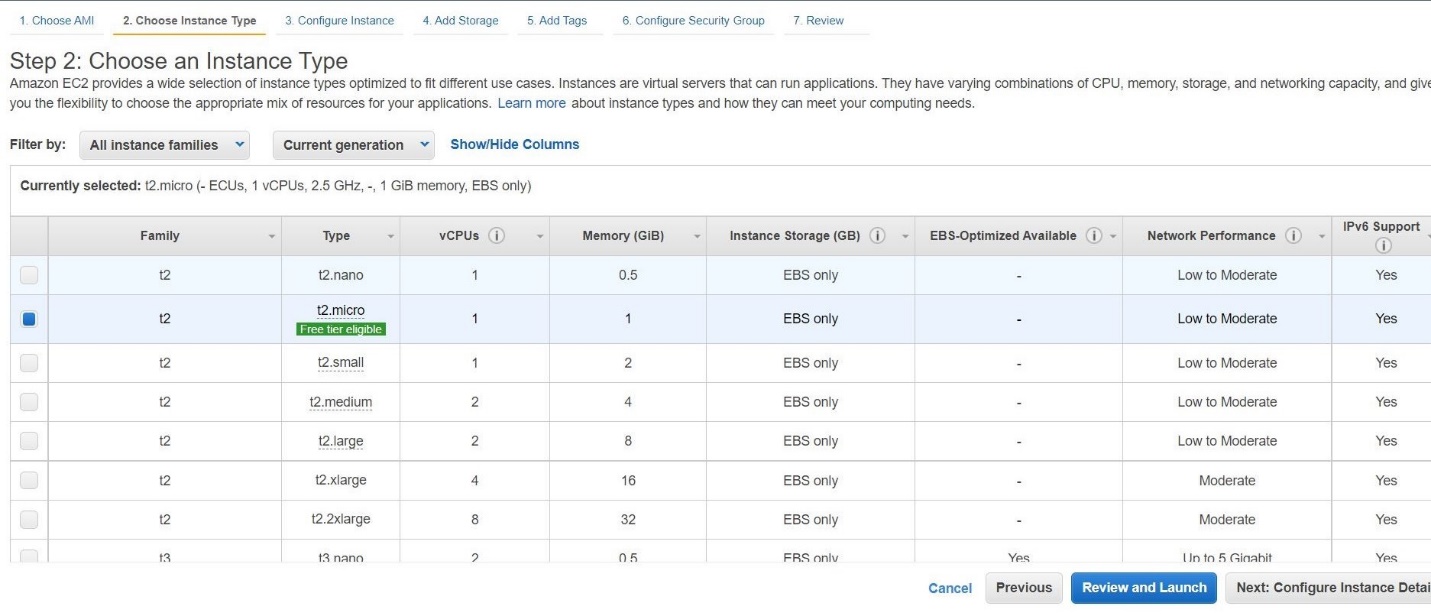
**Now I am going to created ec2 instance**

****

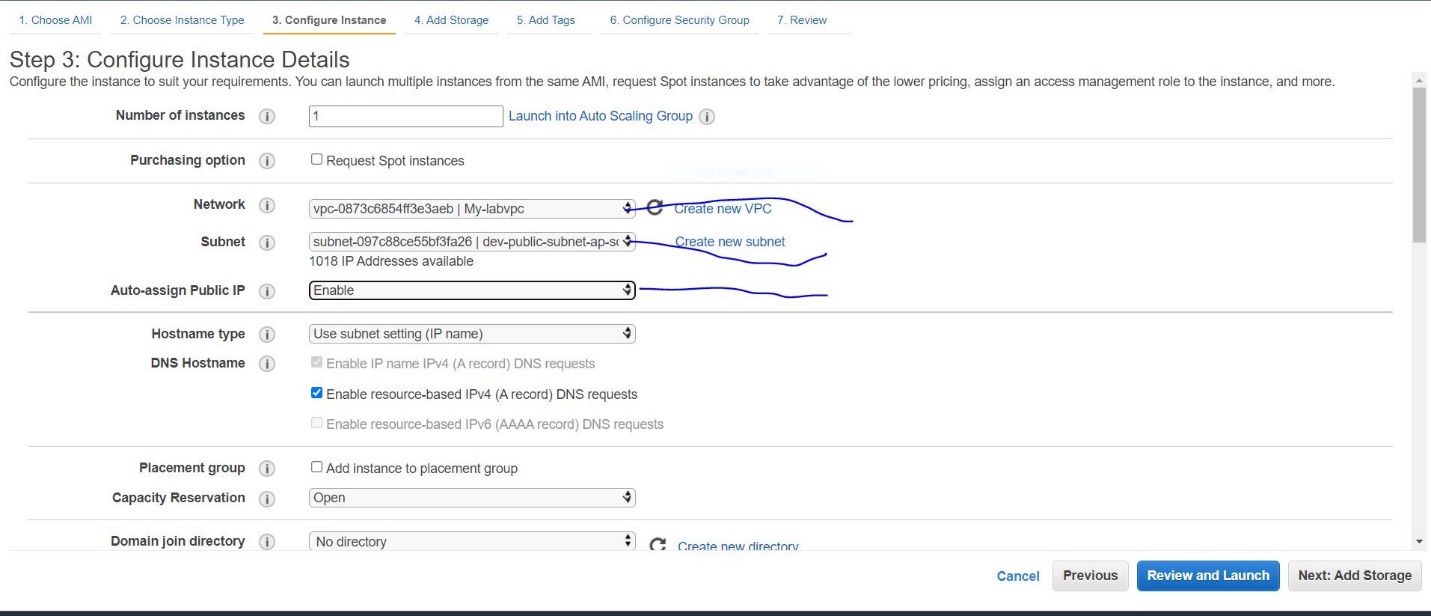
**Now I am going to selecting the ios image**

****

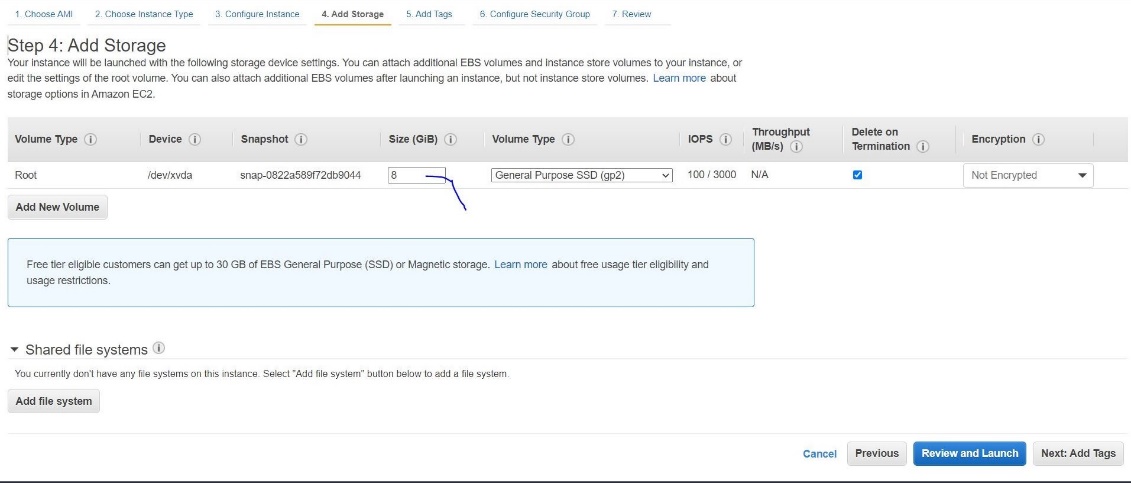
**You can choose which instance type you want ?**

****

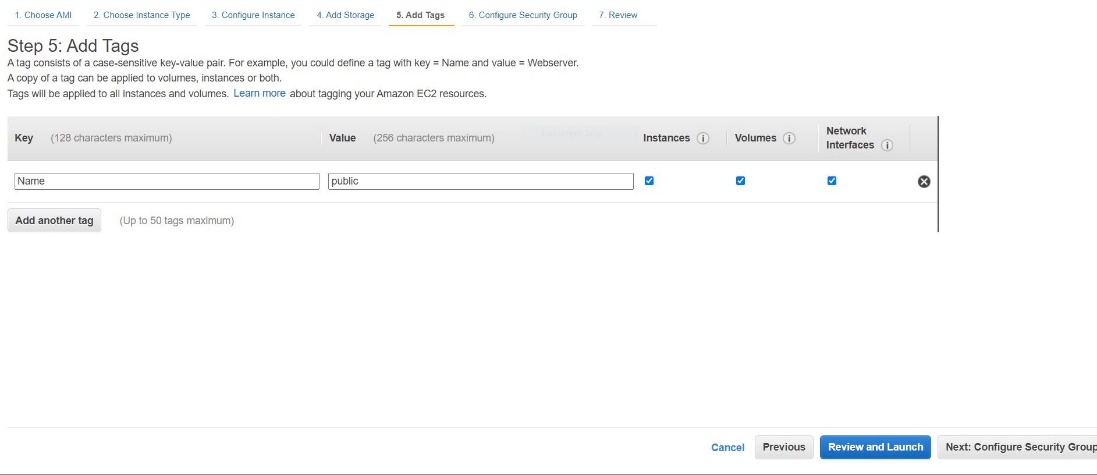
**Here we can I configure my vpc and subnet and enable auto assign public ip**

****

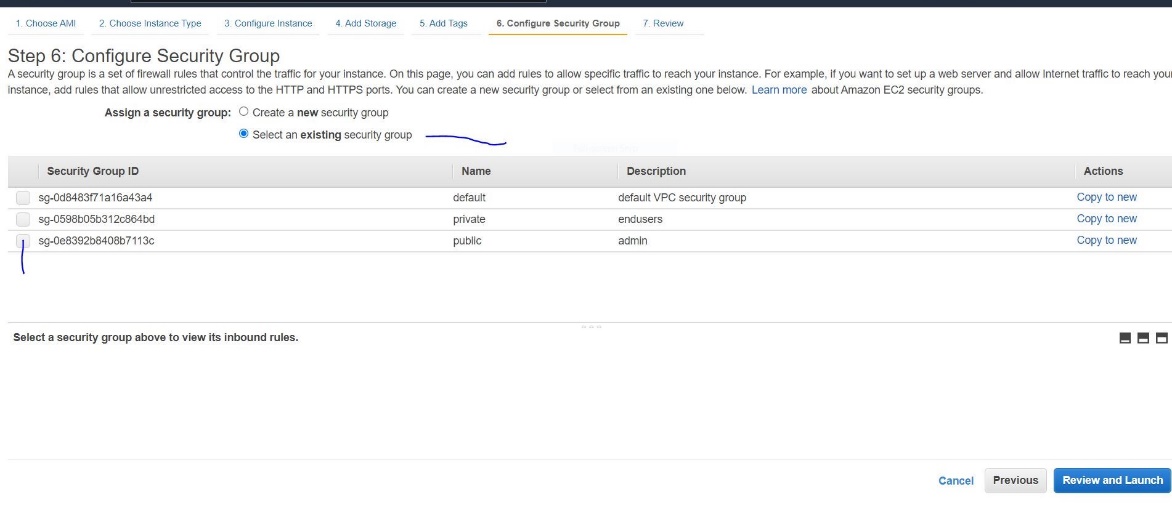
**If want you to give the size here can we give.**

****

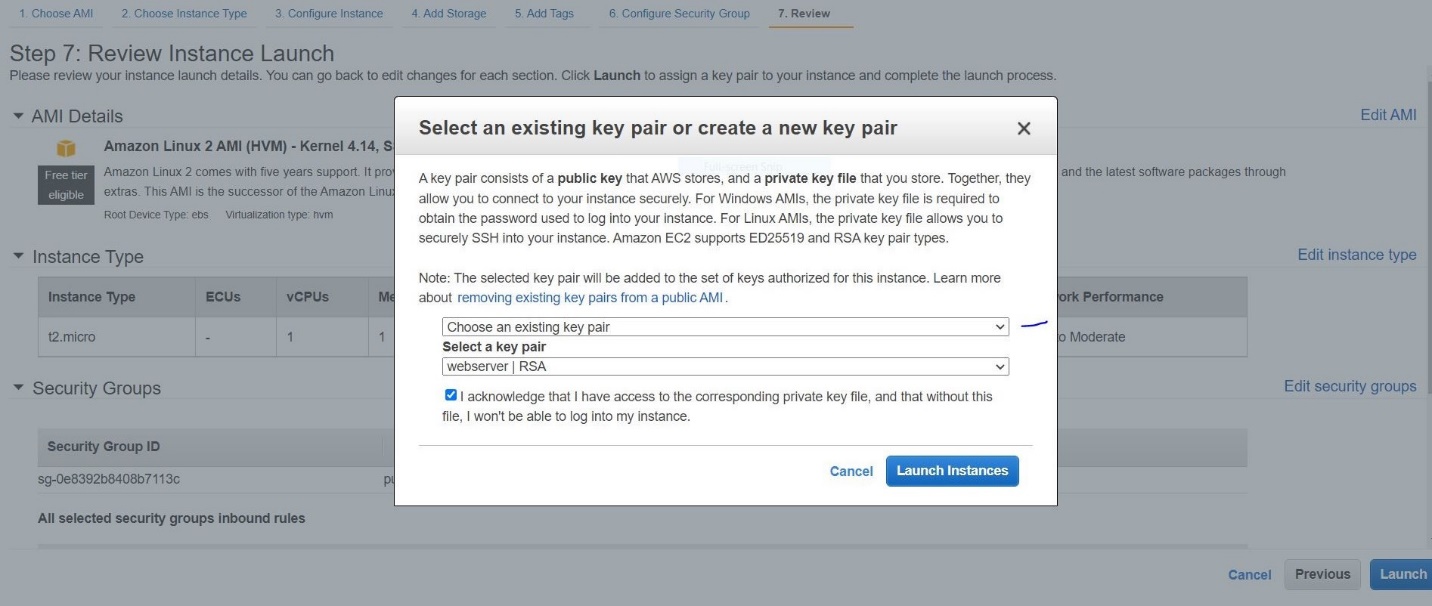
**Here we give Tag name public**

****

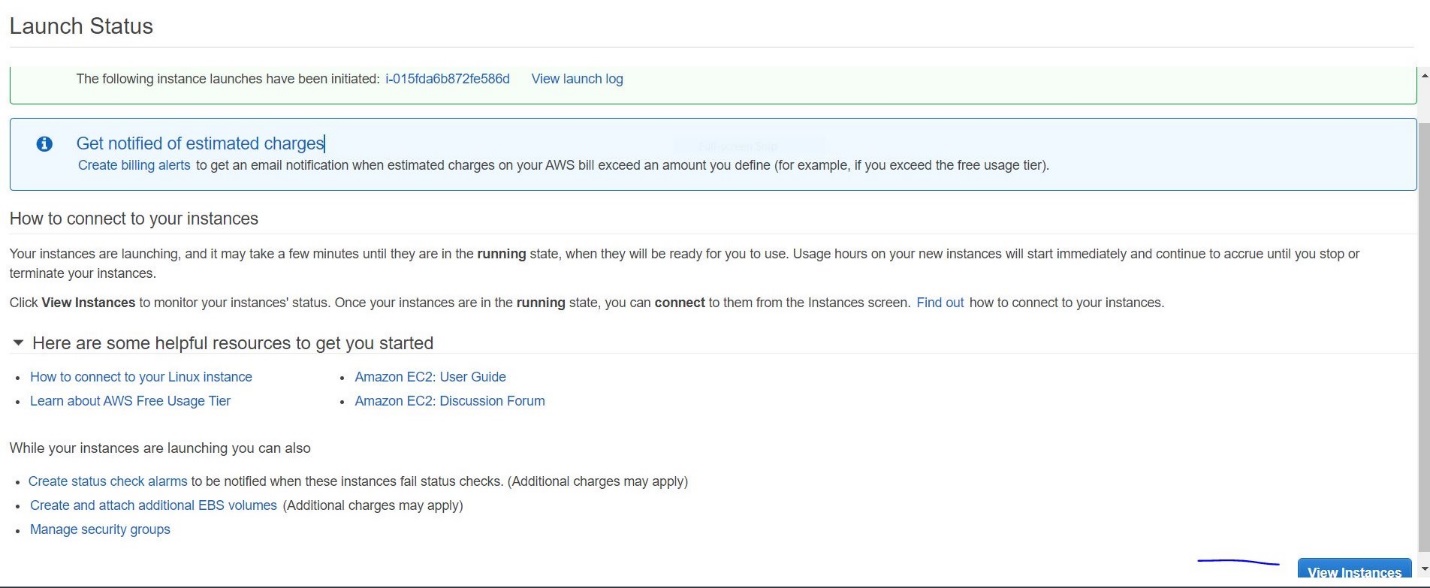
**We can select the existing security group and click the public save it review and lauch**

****

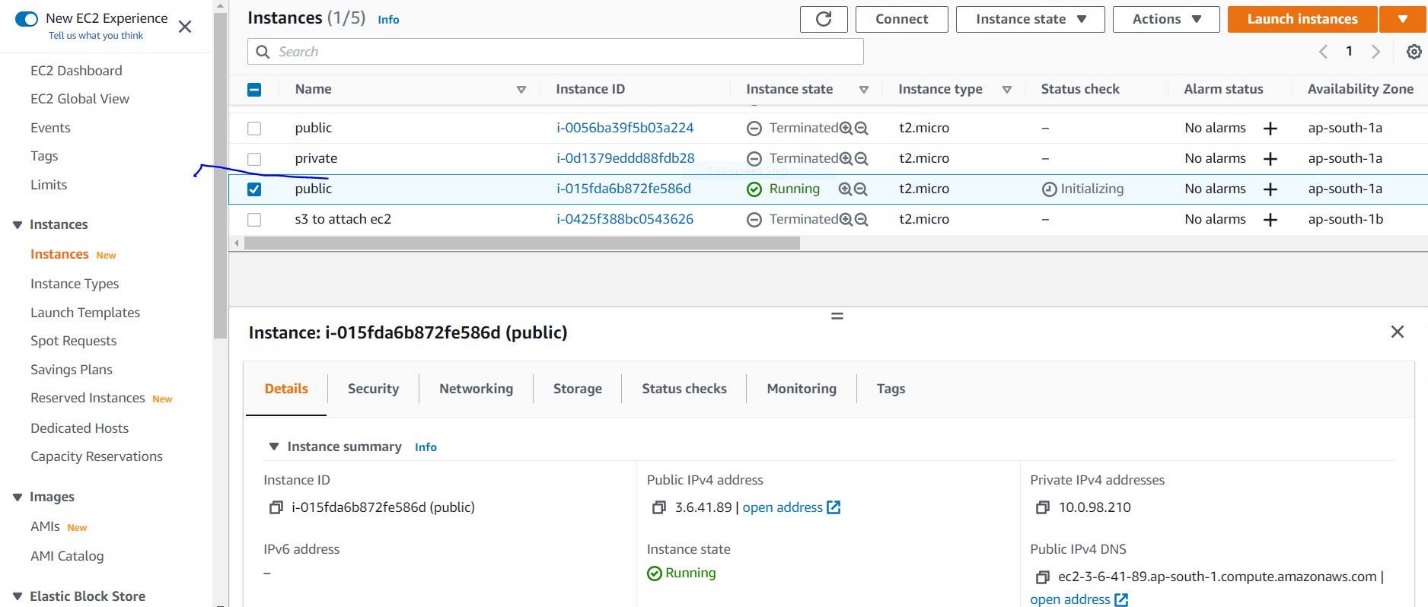
**Here we create new key also but I already created my key that wise I select existing key pair**

****

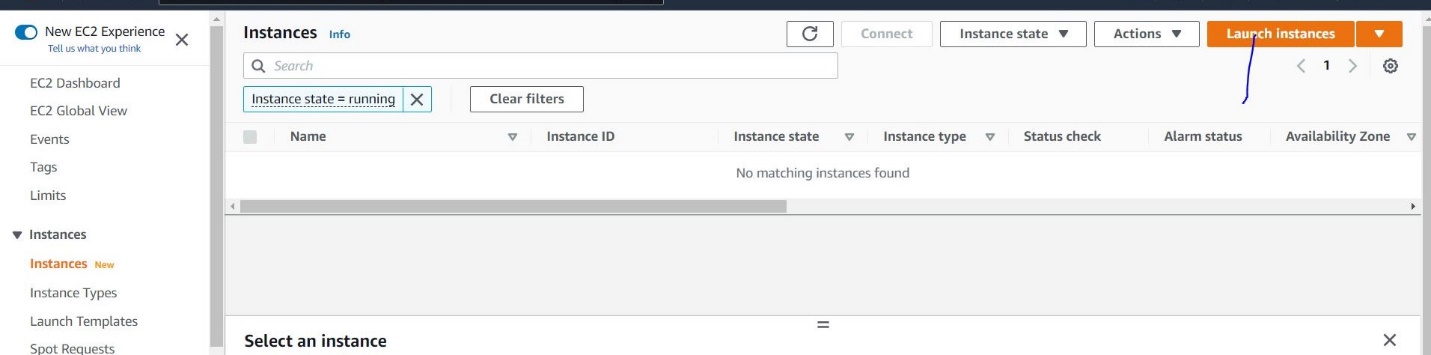
**Now I am lunch my public ec2 machine created this way we need to follow**

****

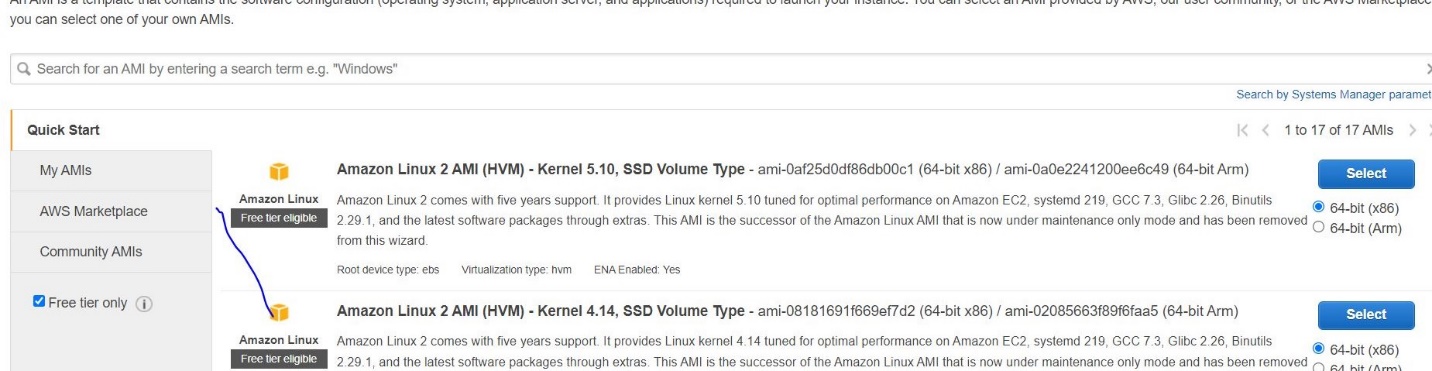
**Here we can see my public ec2 machine ready**

****

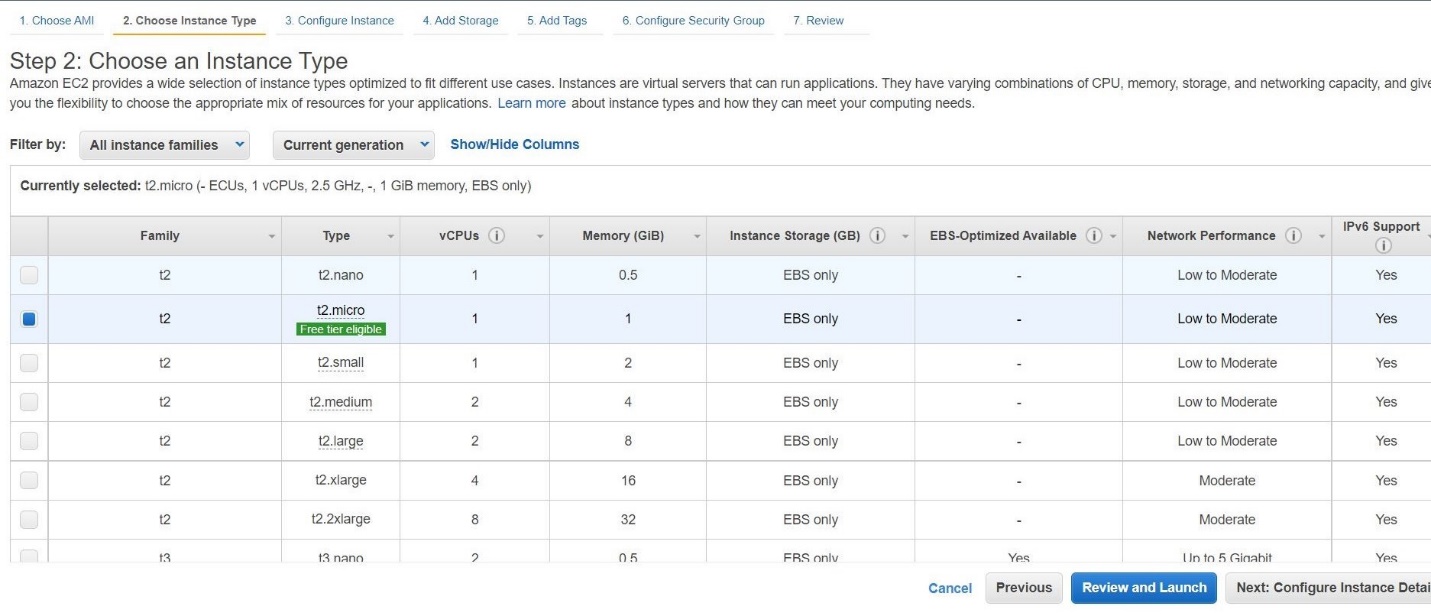
**Now I am creating private ec2 machine**

****

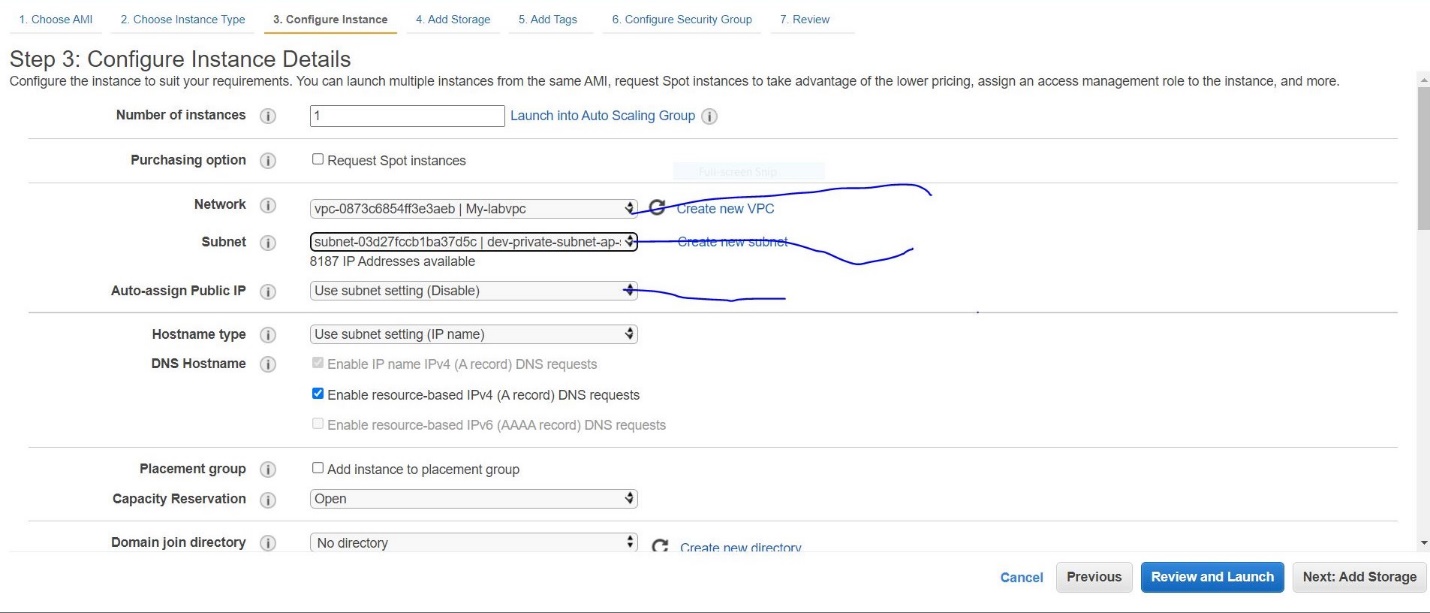
**Now I am going to selecting the ios image .**

****

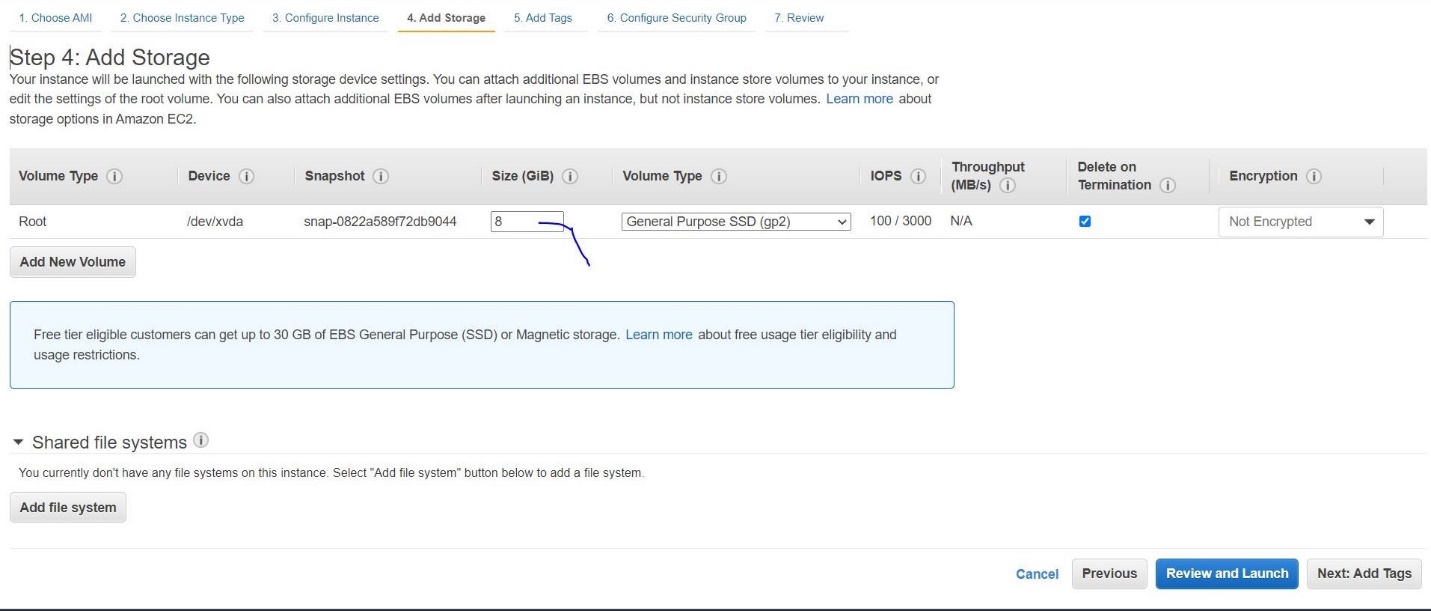
**You can choose which instance type you want ?you can select**

****

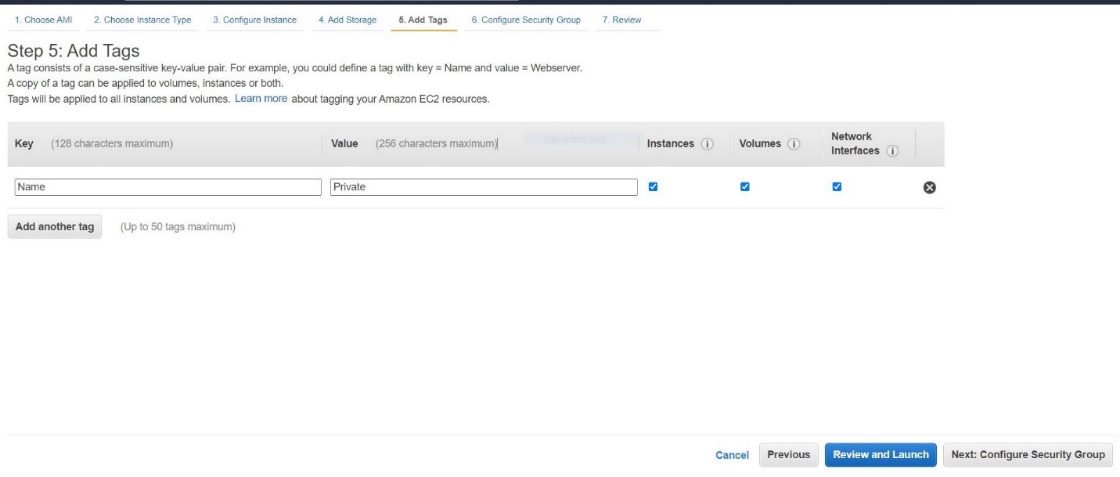
**Here we can I configure network my vpc and subnet private and auto assign disable.**

****

**If want, you to give the size here can we give.**

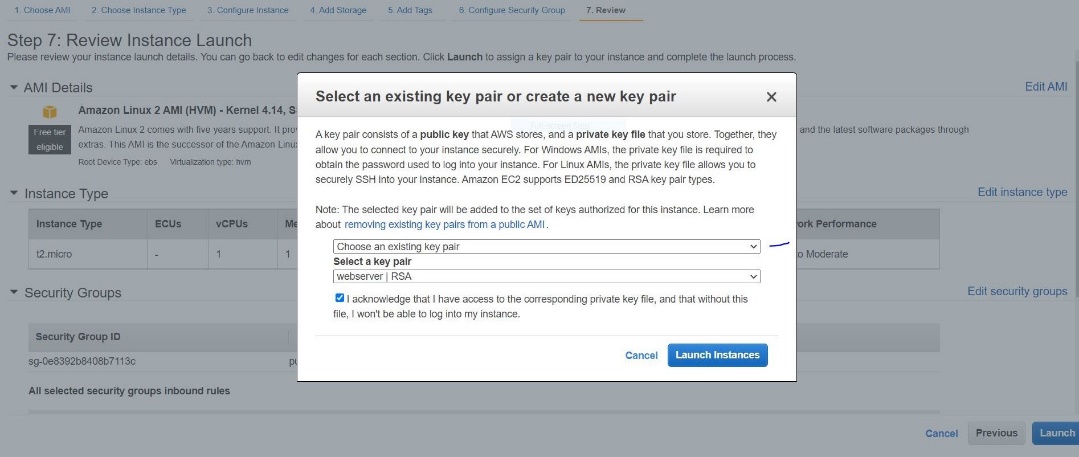
****

**Here we e Tag name private**

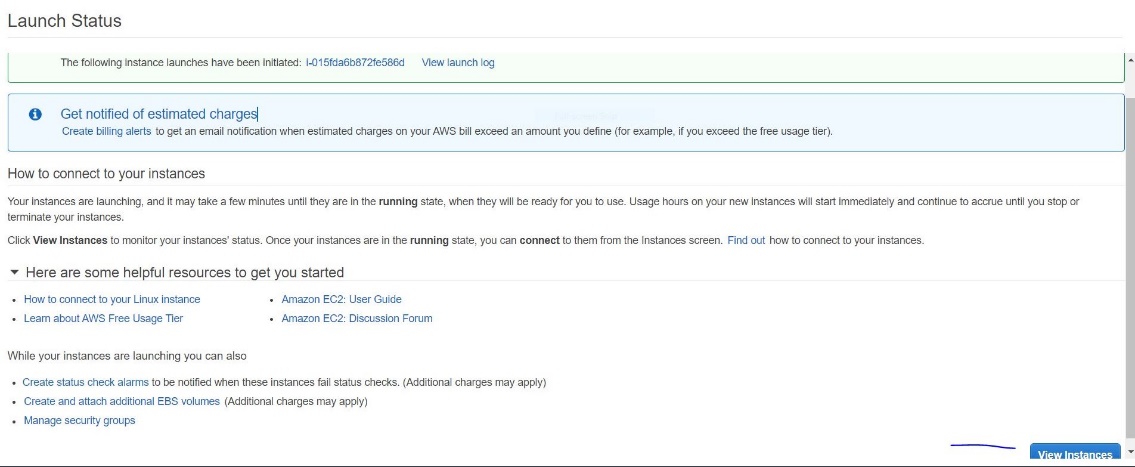
****

**We can select the existing security group and click the private save it review and launch**

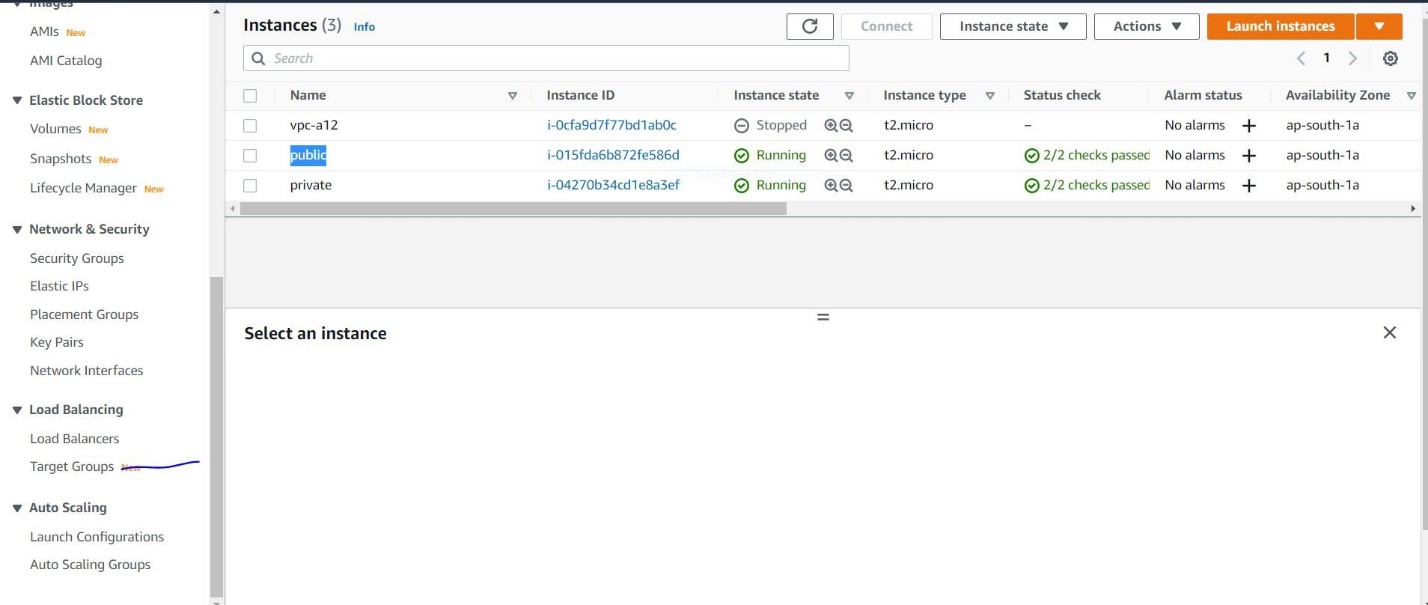
**Here we create new key also but I already created my key that wise I select existing key pair**

****

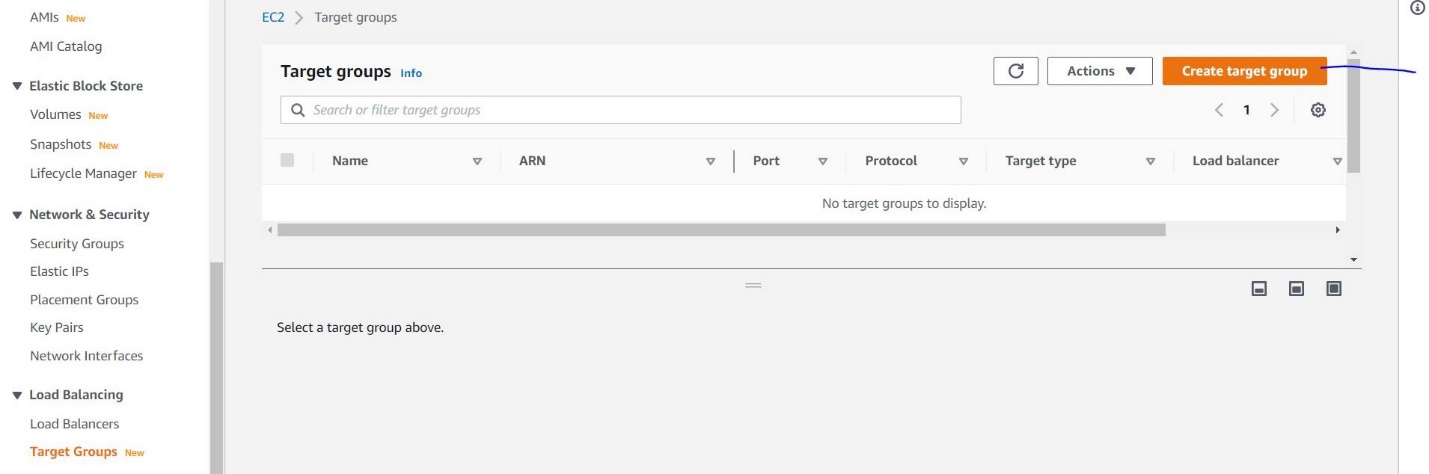
**Now I am lunch my private ec2 machine created this way we need to follow**

****

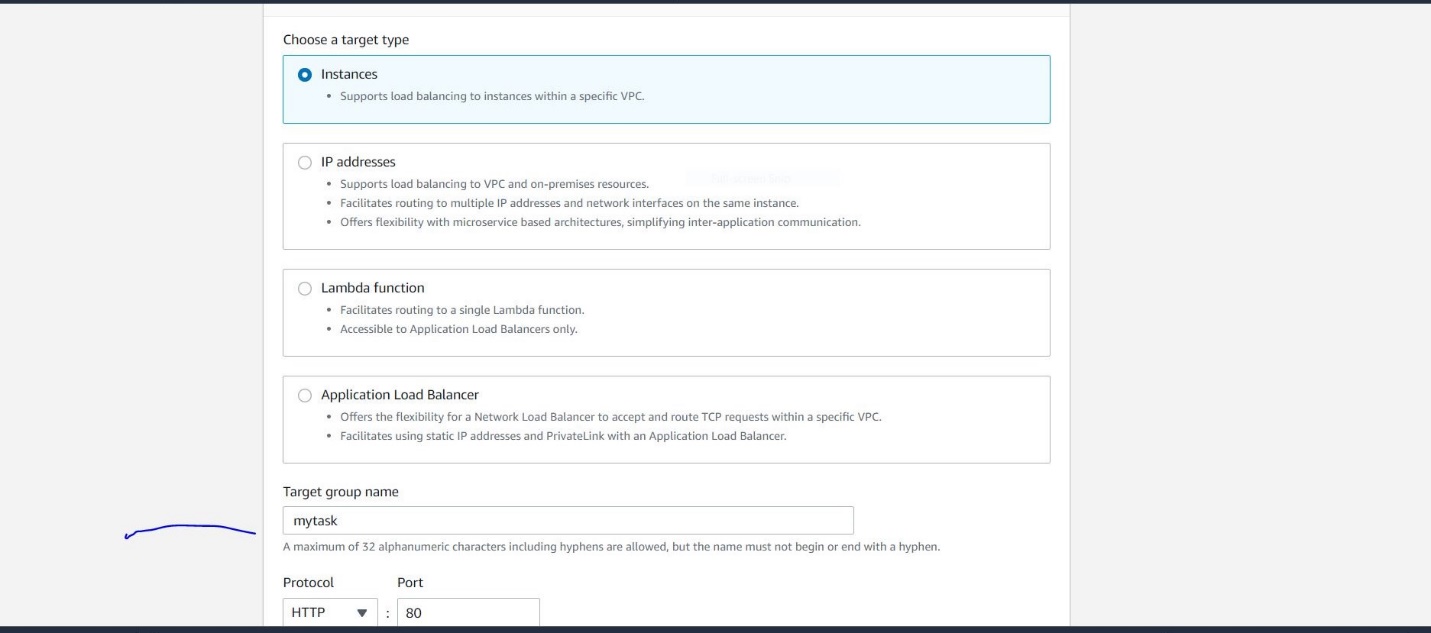
**Now I created both public and private ec2 machine now I’m going creating application load balancer.**

****

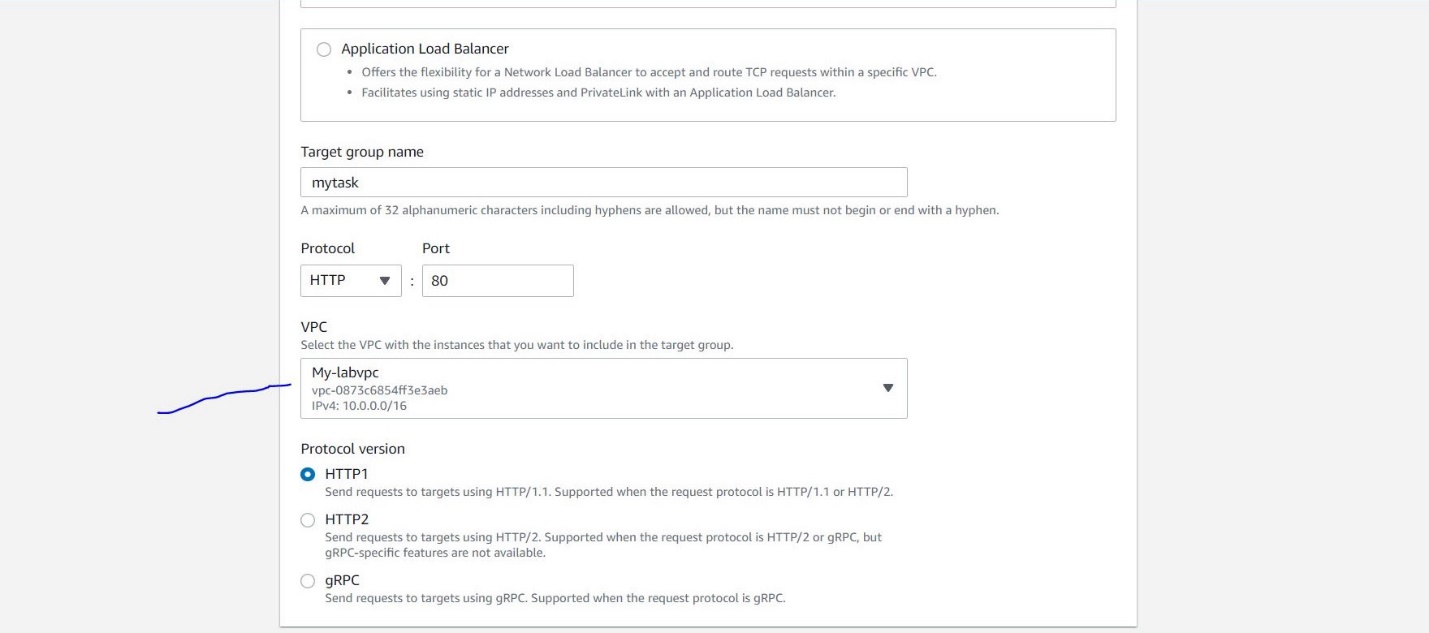
**Frist I creating the target group after I attaching the application load balancers**

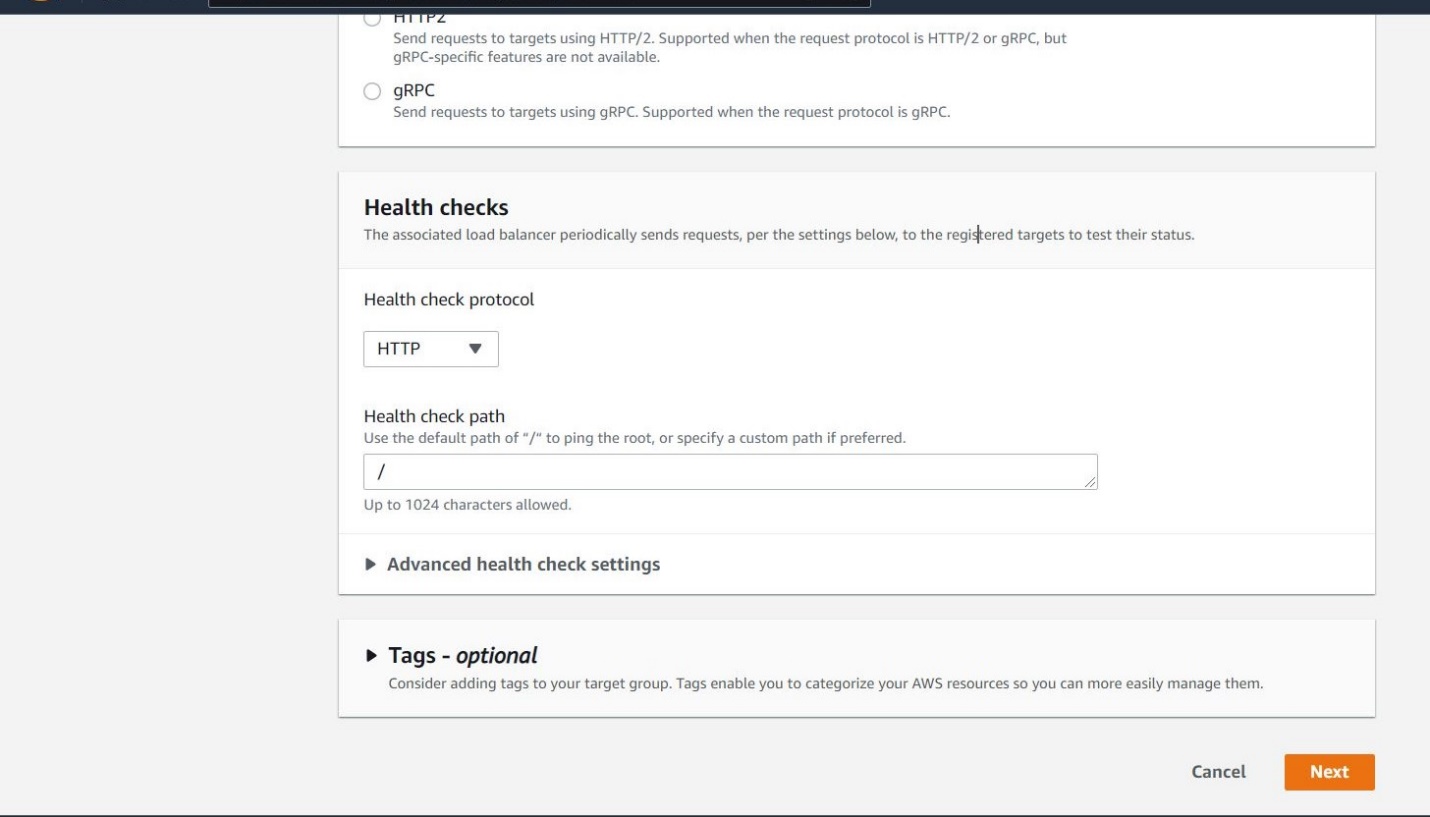
****

**Now here select the instance and give target name**

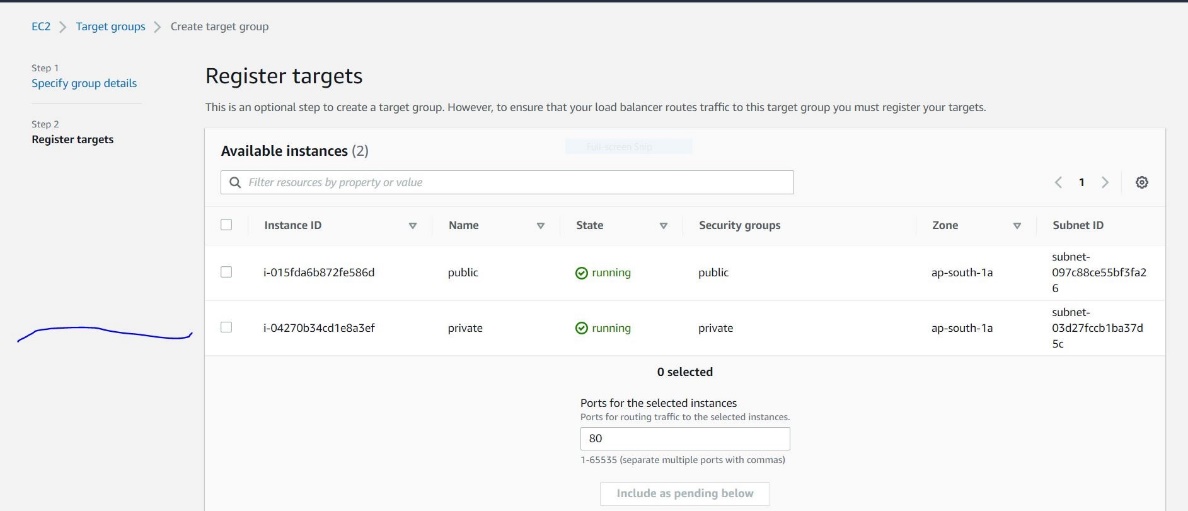
****

**Here we selecting the VPC.**

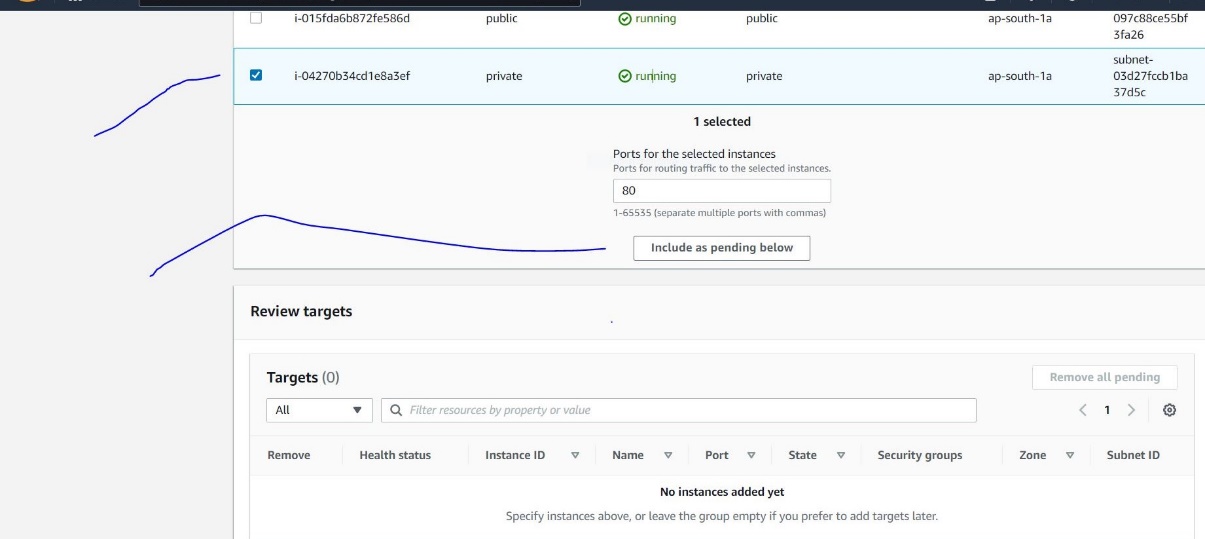
****

****

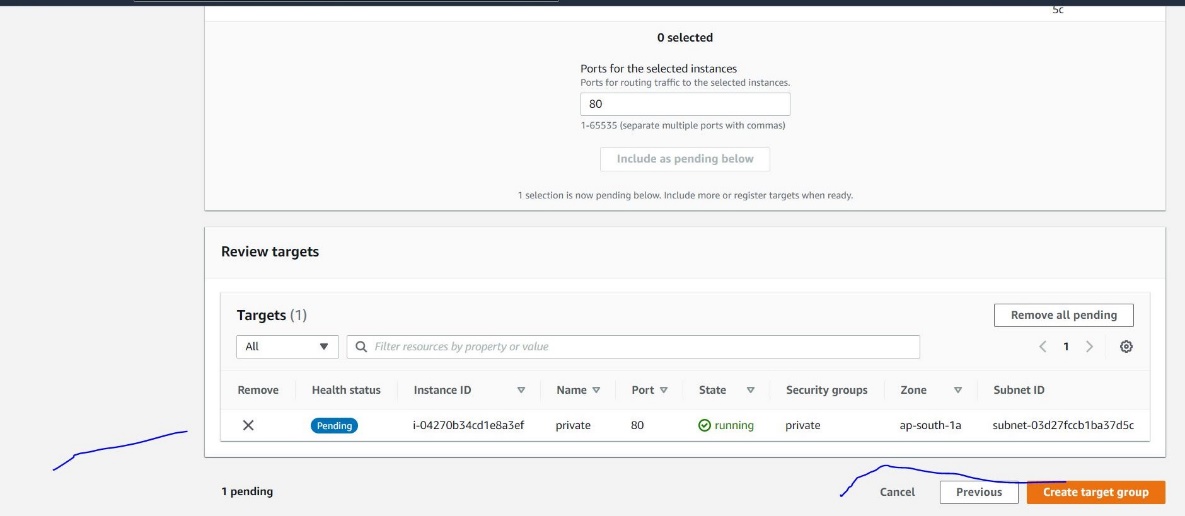
**Now I am going to Register the private click the check box click the include ad pending below button**

****

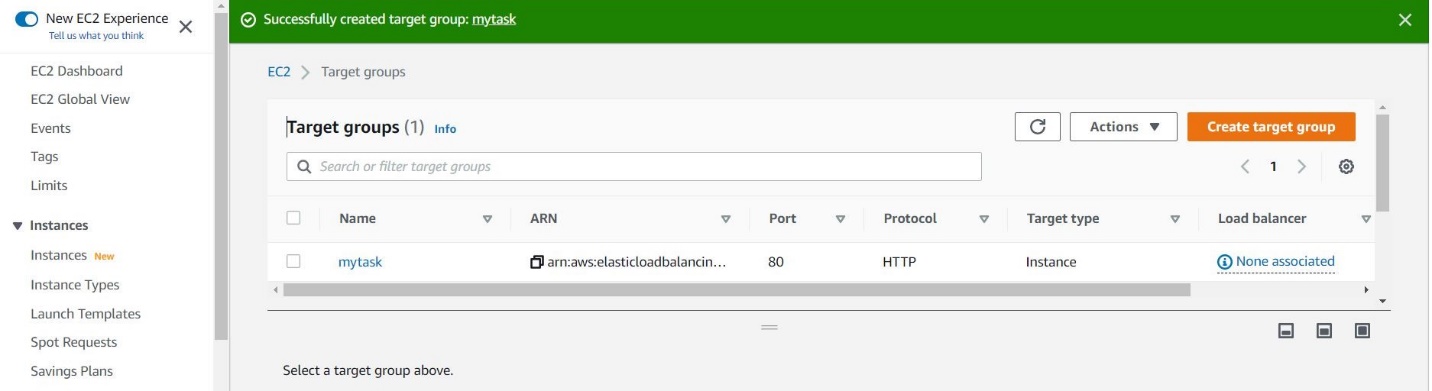
**After click the showing like this.**

****

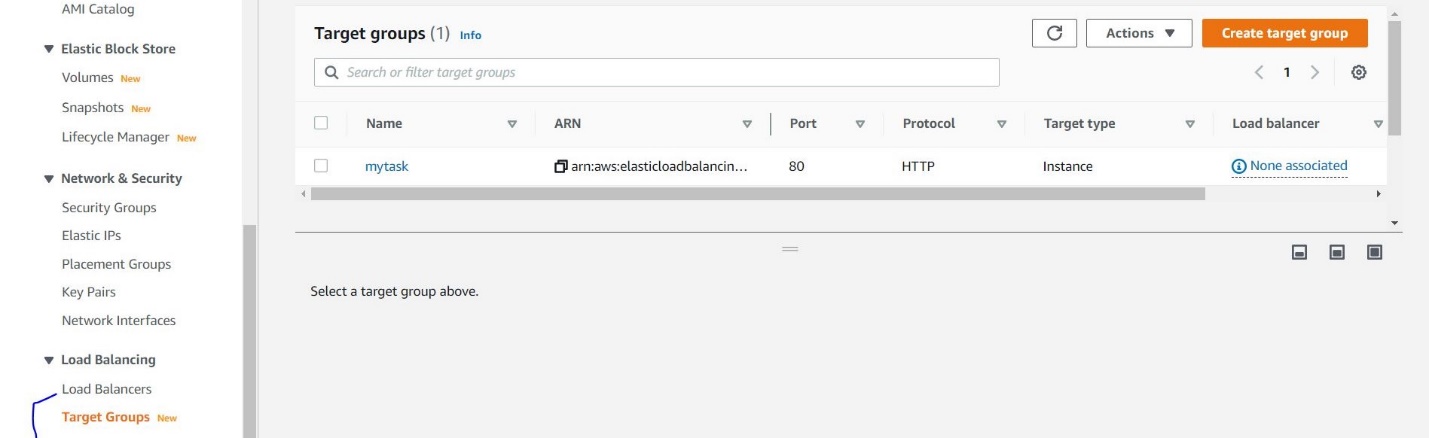
**It will showing the review target click the create target group now is created target group**

****

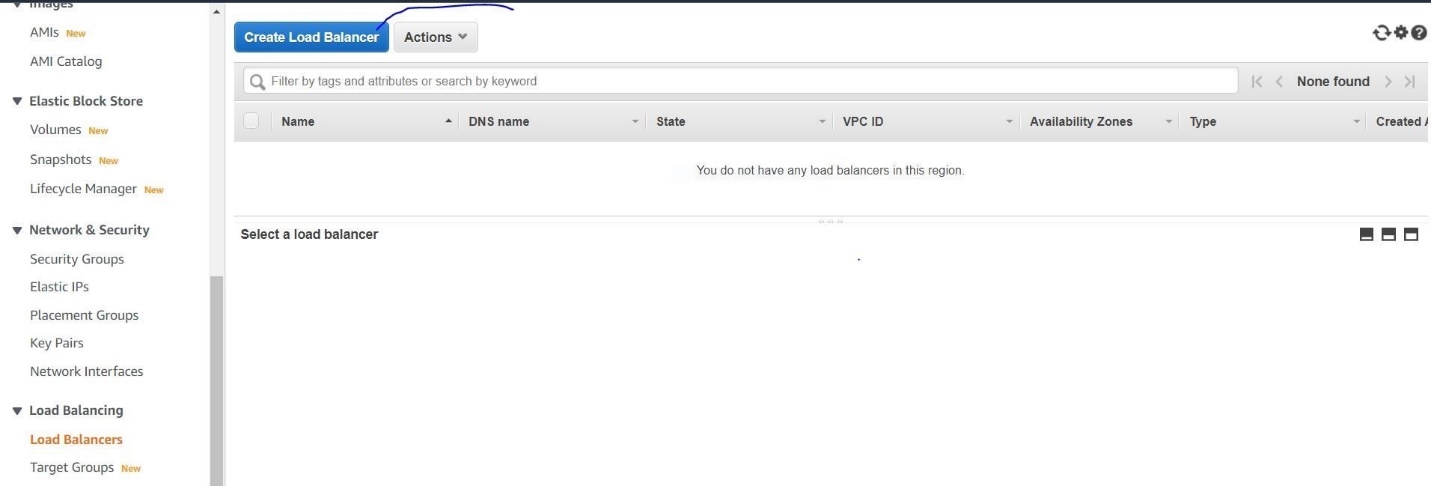
**Now successfully created target group and now I am going to creating application load balancer**

****

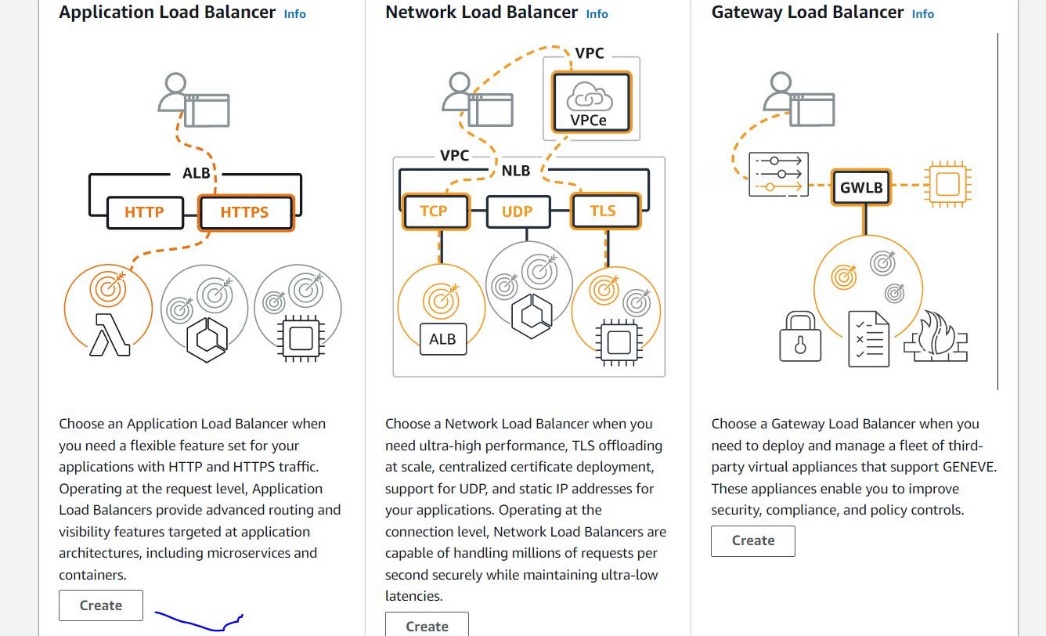
**Now I am creating load balancer click the**

****

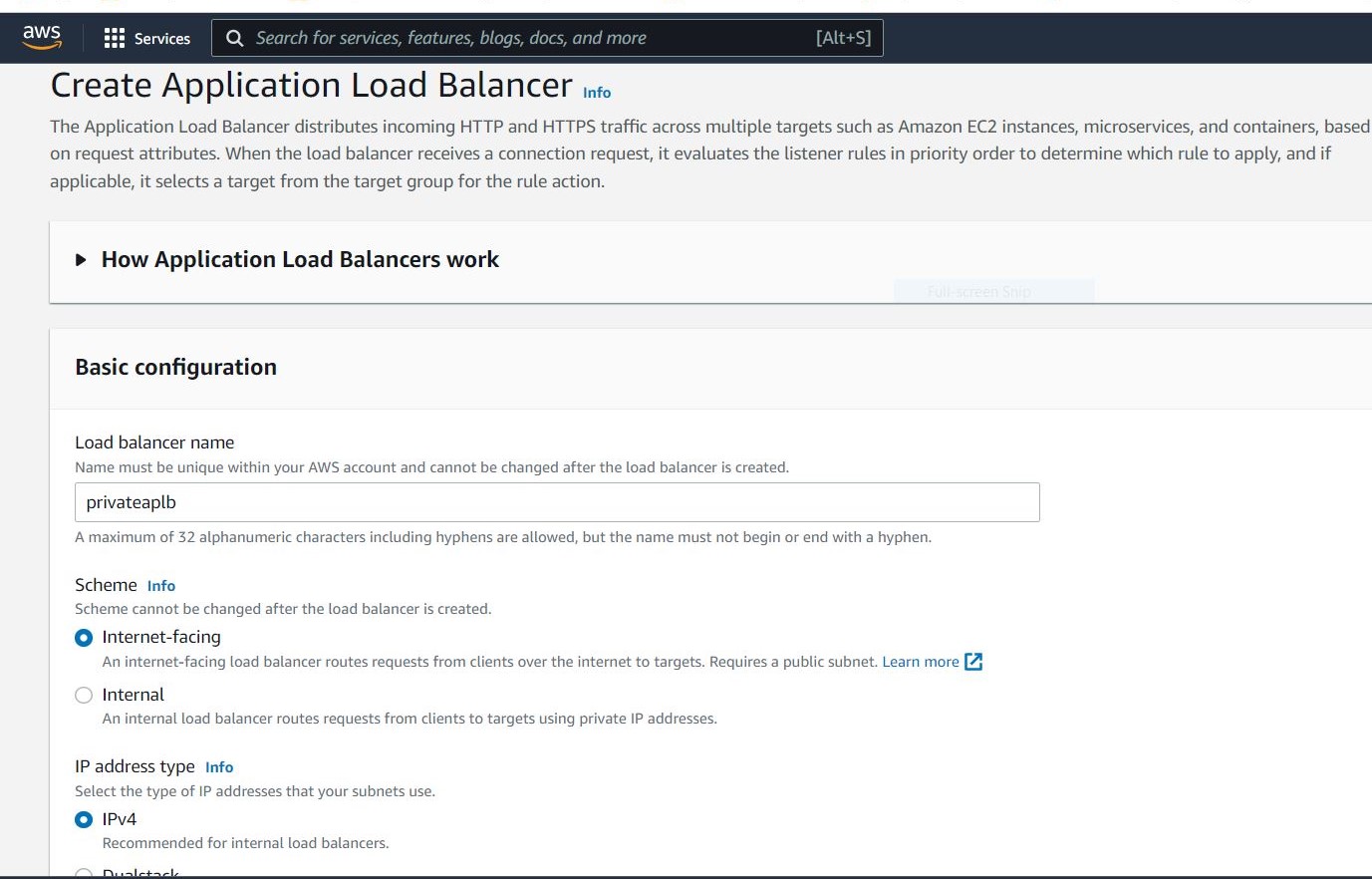
**Creating the load balancer**

****

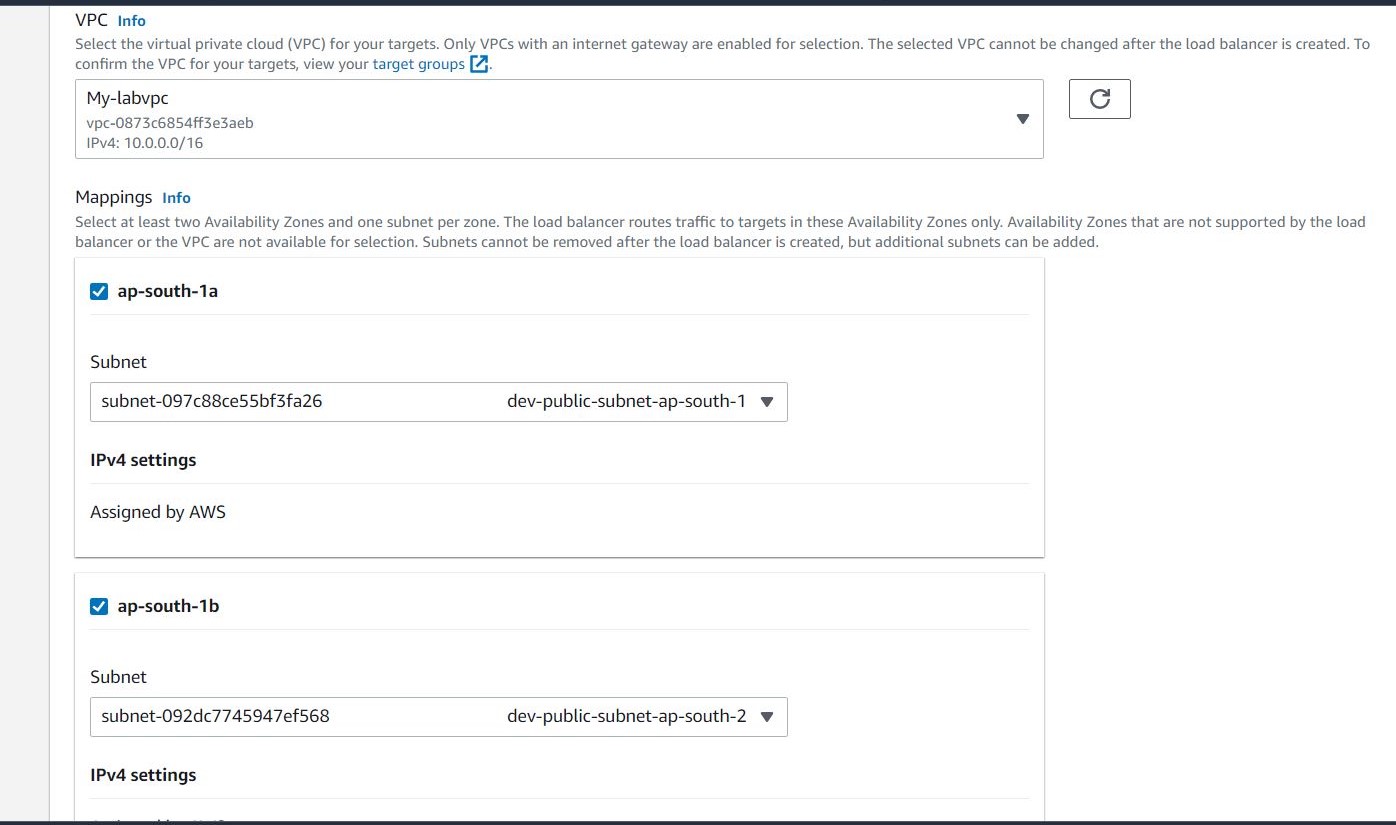
**Selecting the Application Load Balancer**

****

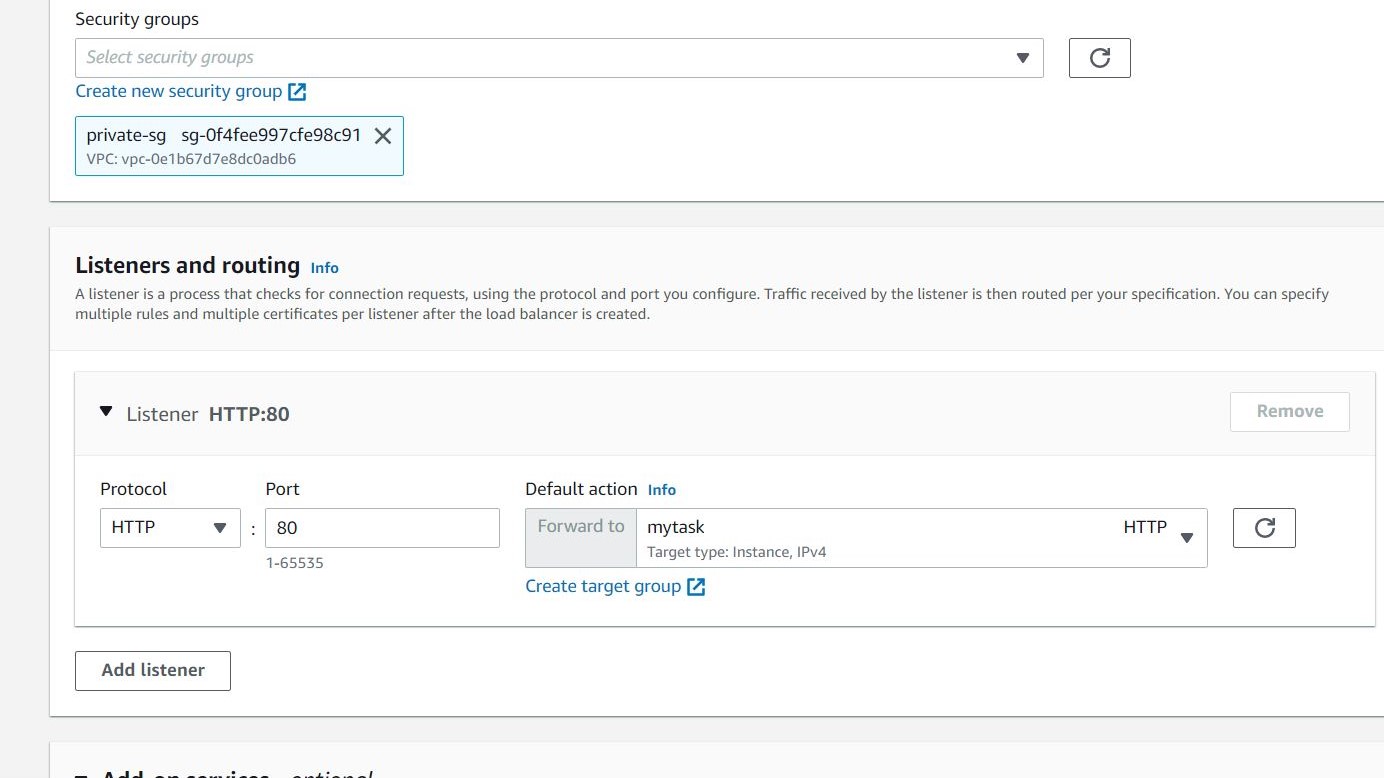
**Now I am creating load balancer name.**

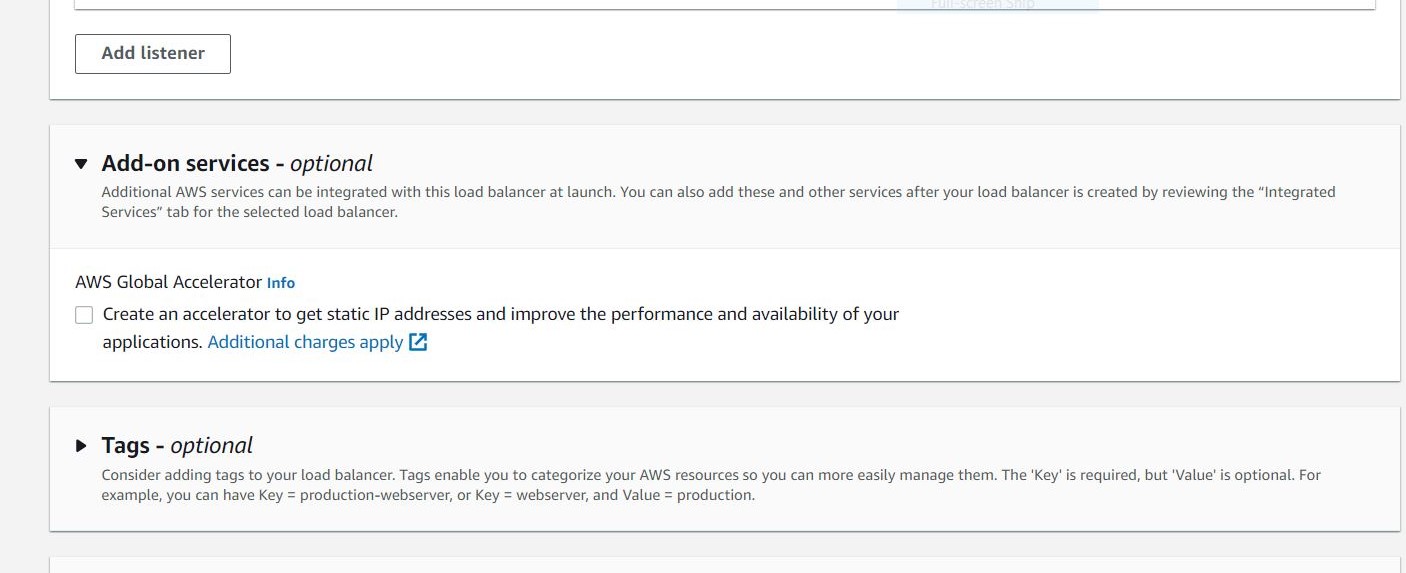
****

**Selecting the vpc the ap-south-1a Here we selecting all public subnets**

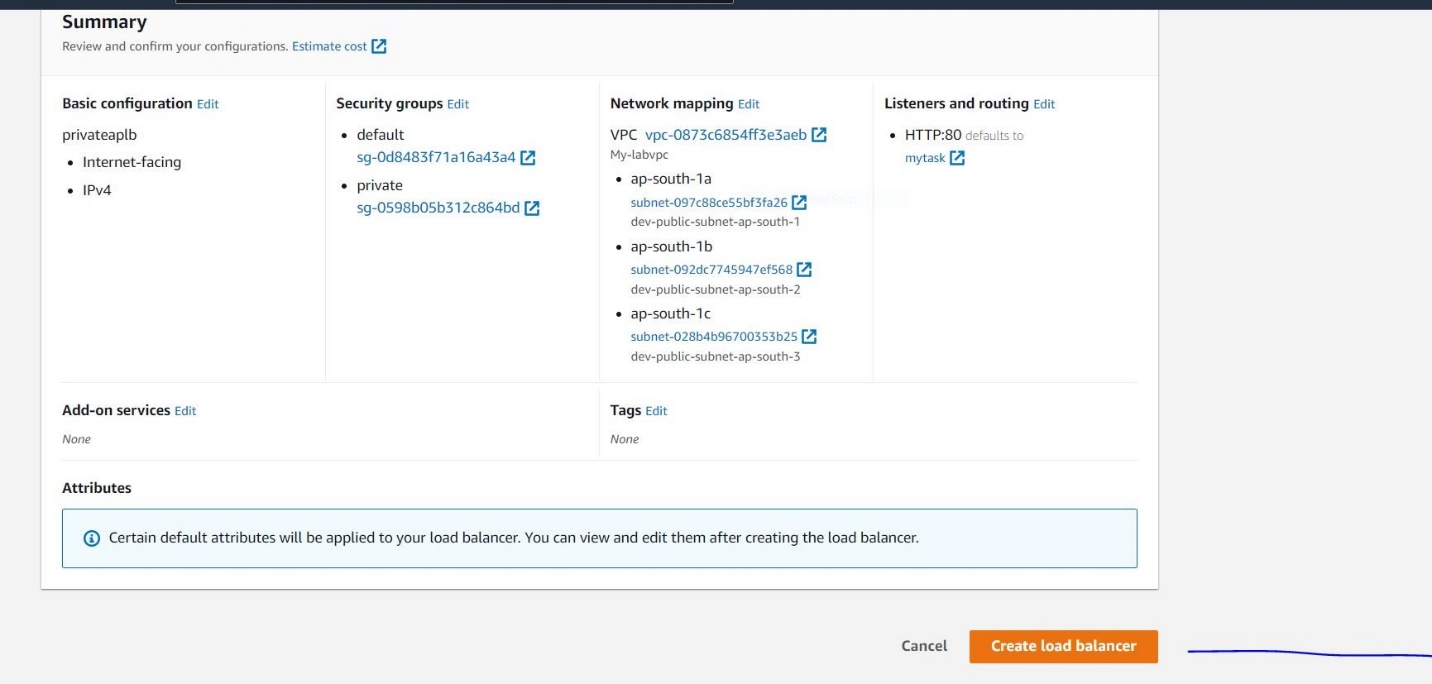
****

**Here we selecting the security groups and selecting the forward to target group**

****

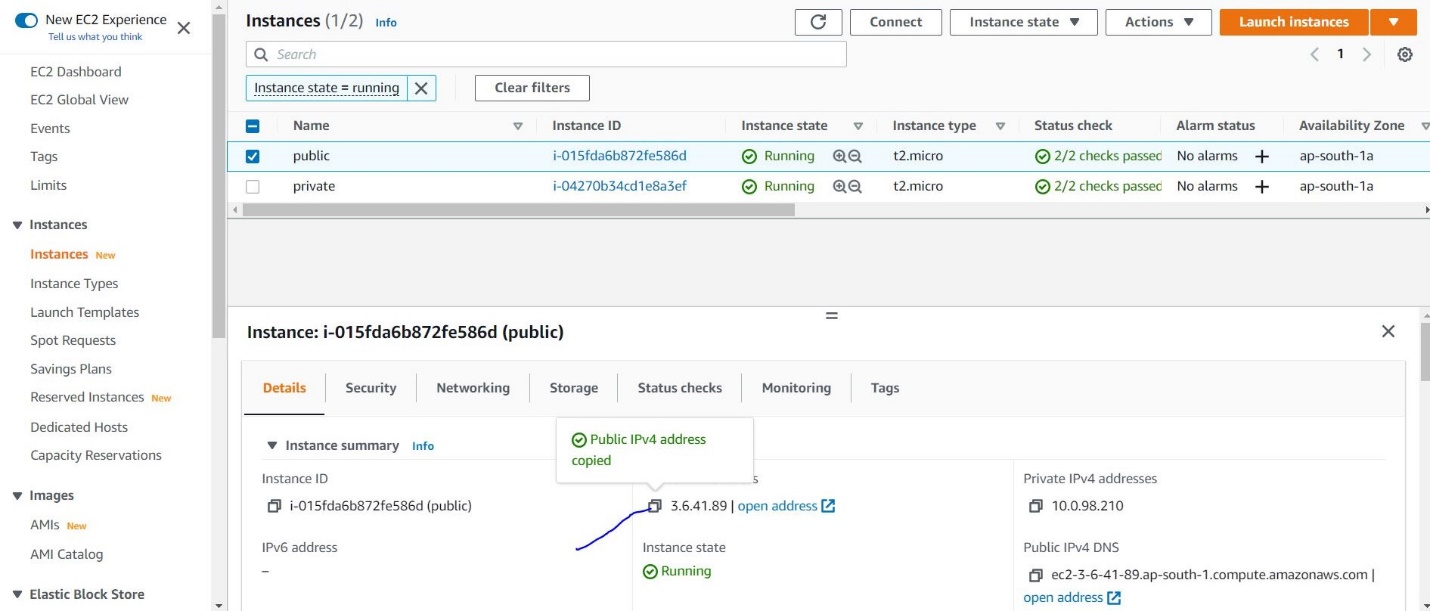
****

**Now we created load balancer now you going to public ec2 machine and connecting private machine**

****

**Here we selecting the public copy the public ip go to your Linux system , or gitbash, putty any one we**

**Choose. I select gitbash here I am doing .**

****

DELL@DESKTOP-6HOFNG0 MINGW64 /E (master)

$ ssh -i webserver.pem ec2-user@3.6.41.89

The authenticity of host '3.6.41.89 (3.6.41.89)' can't be established.

ED25519 key fingerprint is SHA256:eRXcGEdG6/3oEsfmXzvdK/YFxbz/3B7s8rn9hQ1+bV4.

This key is not known by any other names

Are you sure you want to continue connecting (yes/no/[fingerprint])? yes

Warning: Permanently added '3.6.41.89' (ED25519) to the list of known hosts.

\_\_| \_\_|\_ )

\_| ( / Amazon Linux 2 AMI

\_\_\_|\\_\_\_|\_\_\_|

https://aws.amazon.com/amazon-linux-2/

11 package(s) needed for security, out of 15 available

Run "sudo yum update" to apply all updates.

[ec2-user@ip-10-0-98-210 ~]$

[ec2-user@ip-10-0-98-210 ~]$ exit

logout

Connection to 3.6.41.89 closed.

**Here we copying pem file to the public ec2 machine because connecting public to private**

DELL@DESKTOP-6HOFNG0 MINGW64 /E (master)

$ scp -i webserver.pem webserver.pem ec2-user@3.6.41.89:/home/ec2-user

webserver.pem 100% 1704 12.1KB/s 00:00

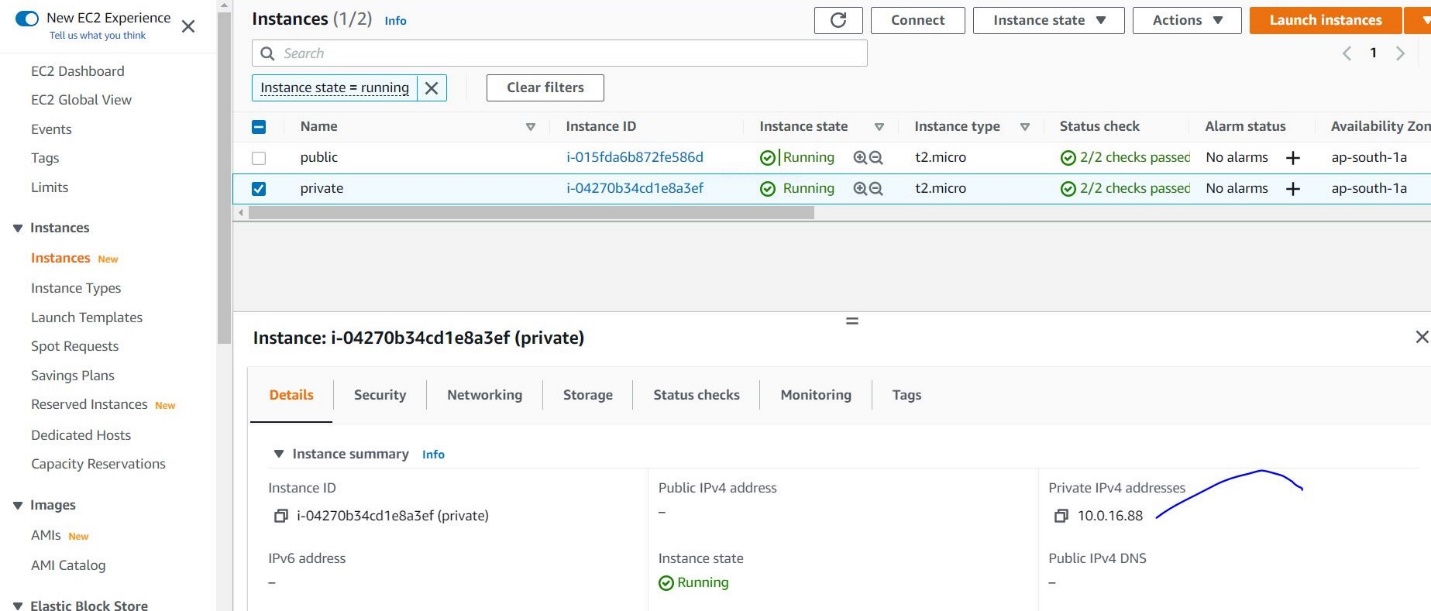
DELL@DESKTOP-6HOFNG0 MINGW64 /E (master)

$

[ec2-user@ip-10-0-98-210 ~]$ ls======public-ip

webserver.pem

Here we taking the private ip



[ec2-user@ip-10-0-98-210 ~]$ chmod 400 webserver.pem

[ec2-user@ip-10-0-98-210 ~]$ ssh -i webserver.pem [ec2-user@10.0.16.88](mailto:ec2-user@10.0.16.88) –private ip

The authenticity of host '10.0.16.88 (10.0.16.88)' can't be established.

ECDSA key fingerprint is SHA256:fZbPAXzqRjtMZuAL2fNrlk9KOn4z+uw20biJXU2uGZc.

ECDSA key fingerprint is MD5:4d:b6:ab:82:8e:b6:51:39:3e:6f:6d:84:9a:3d:34:3c.

Are you sure you want to continue connecting (yes/no)? yes

Warning: Permanently added '10.0.16.88' (ECDSA) to the list of known hosts.

\_\_| \_\_|\_ )

\_| ( / Amazon Linux 2 AMI

\_\_\_|\\_\_\_|\_\_\_|

https://aws.amazon.com/amazon-linux-2/

11 package(s) needed for security, out of 15 available

Run "sudo yum update" to apply all updates.

[ec2-user@ip-10-0-16-88 ~]$

[ec2-user@ip-10-0-16-88 ~]$ sudo su -

[root@ip-10-0-16-88 ~]# yum update -y && yum install httpd –y

Now packages installing wait minute

Once it’s done.

Then start service.

[root@ip-10-0-16-88 ~]# systemctl start httpd

[root@ip-10-0-16-88 ~]# systemctl status httpd

● httpd.service - The Apache HTTP Server

Loaded: loaded (/usr/lib/systemd/system/httpd.service; disabled; vendor preset: disabled)

Active: active (running) since Tue 2022-01-25 10:08:07 UTC; 25s ago

Docs: man:httpd.service(8)

Main PID: 446 (httpd)

Status: "Total requests: 2; Idle/Busy workers 100/0;Requests/sec: 0.105; Bytes served/sec: 428 B/sec"

CGroup: /system.slice/httpd.service

├─446 /usr/sbin/httpd -DFOREGROUND

├─447 /usr/sbin/httpd -DFOREGROUND

├─448 /usr/sbin/httpd -DFOREGROUND

├─449 /usr/sbin/httpd -DFOREGROUND

├─450 /usr/sbin/httpd -DFOREGROUND

└─451 /usr/sbin/httpd -DFOREGROUND

Jan 25 10:08:07 ip-10-0-16-88.ap-south-1.compute.internal systemd[1]: Startin...

Jan 25 10:08:07 ip-10-0-16-88.ap-south-1.compute.internal systemd[1]: Started...

Hint: Some lines were ellipsized, use -l to show in full.

[root@ip-10-0-16-88 ~]#

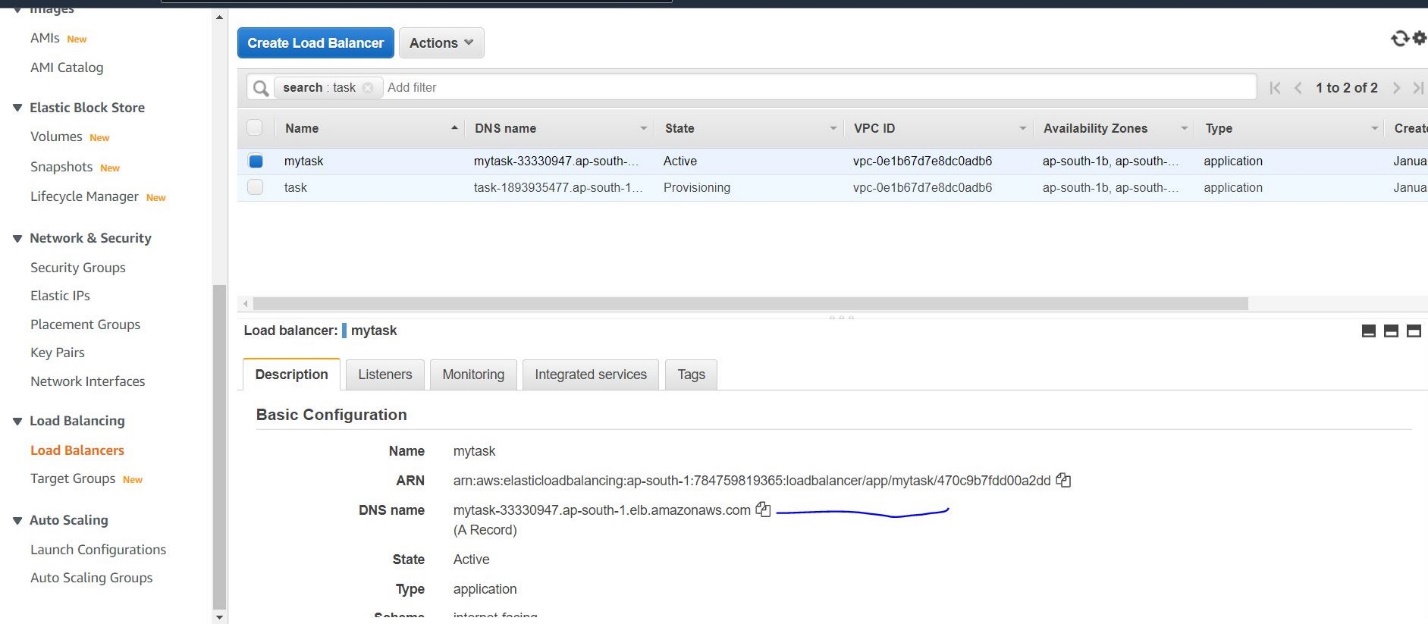
[root@ip-10-0-16-88 ~]# cd /var/www/html/

[root@ip-10-0-16-88 html]# vi index.html

“Hello World from ip-10-0-22-74.ap-south-1.compute.internal”

:wq!

Go to your load balancer take the copy DNS NAME go to your page



Enter here Dns NAMe

