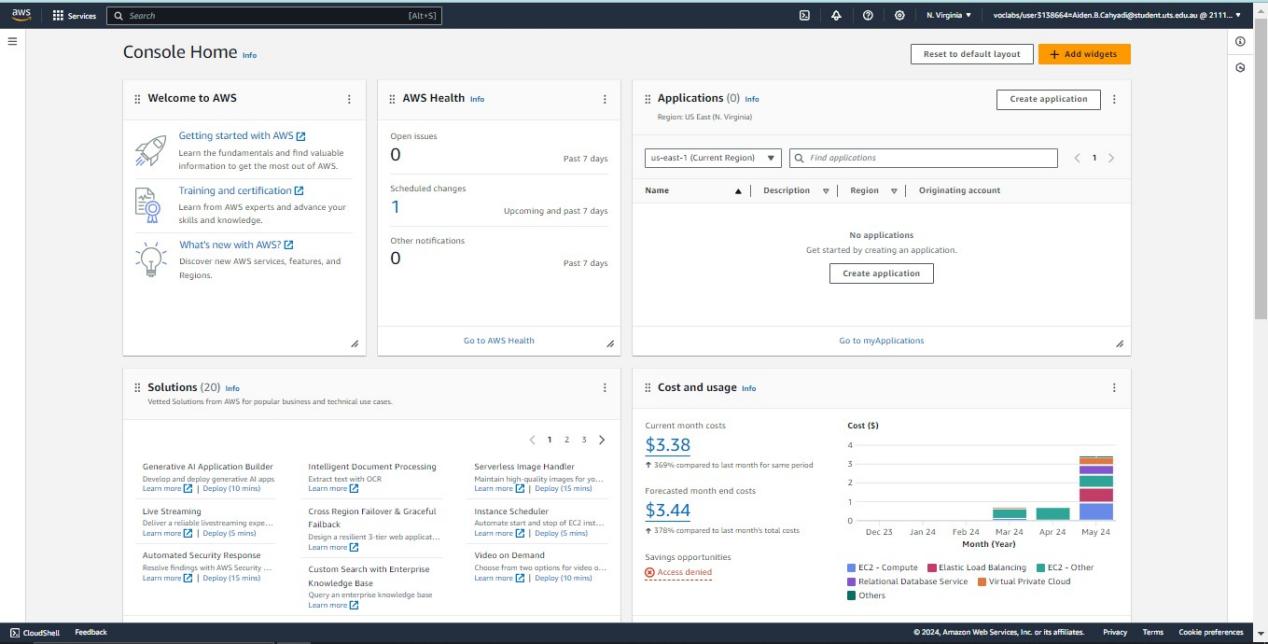
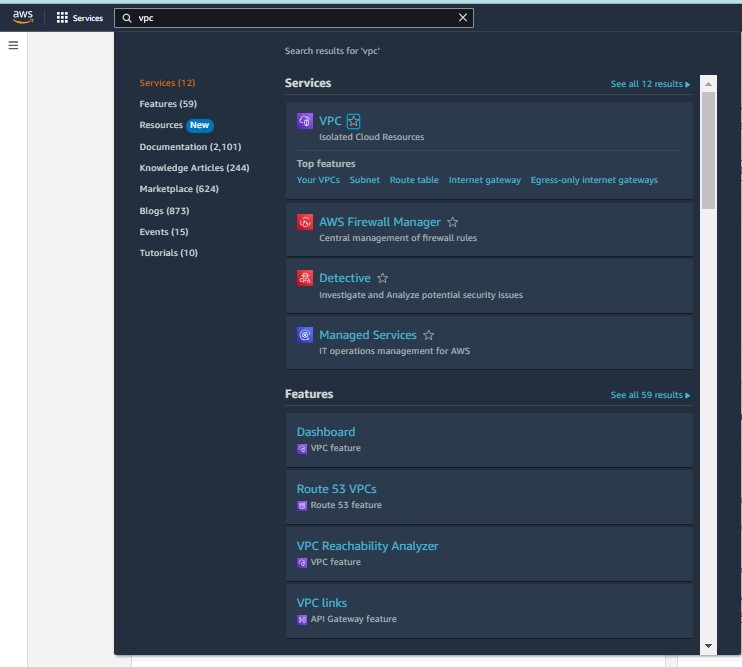
**AWS Application Load Balancer with Auto Scaling group**

Go to AWS console



Go to VPC



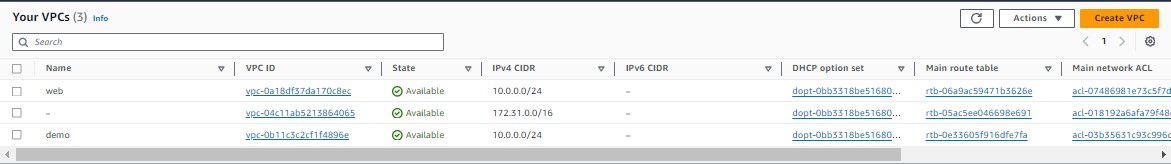
Click Create a new vpc



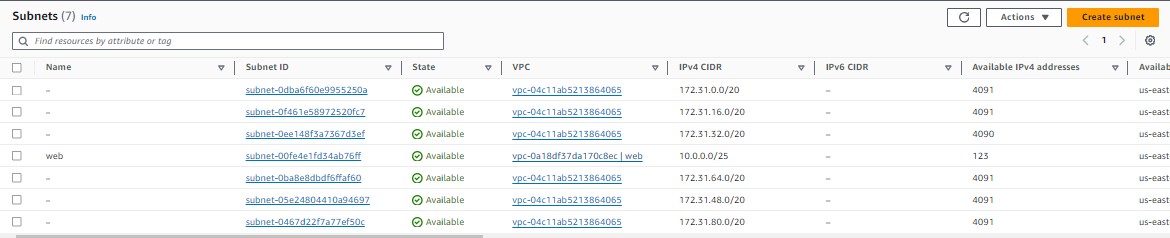
Now give the vpc details

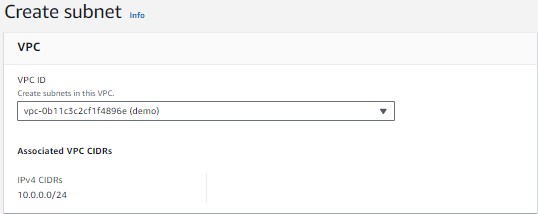


Now here we can see the created vpc

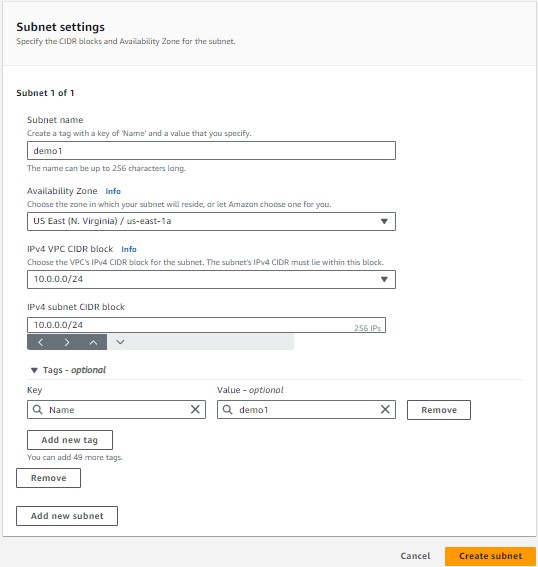


Now go to subnet

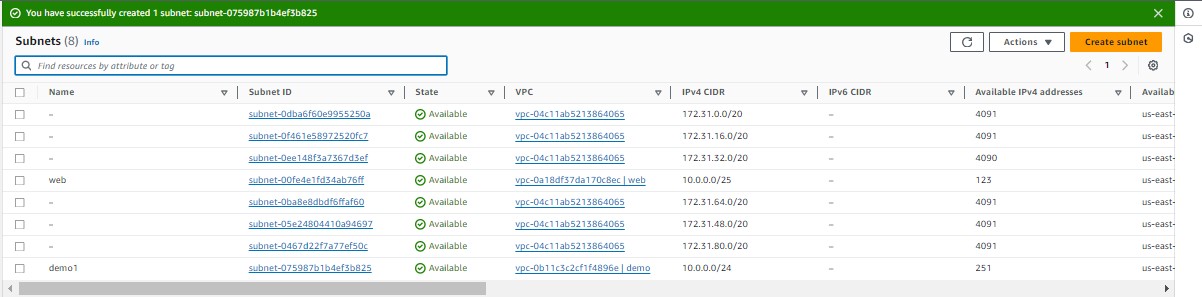


Connect our VPC 

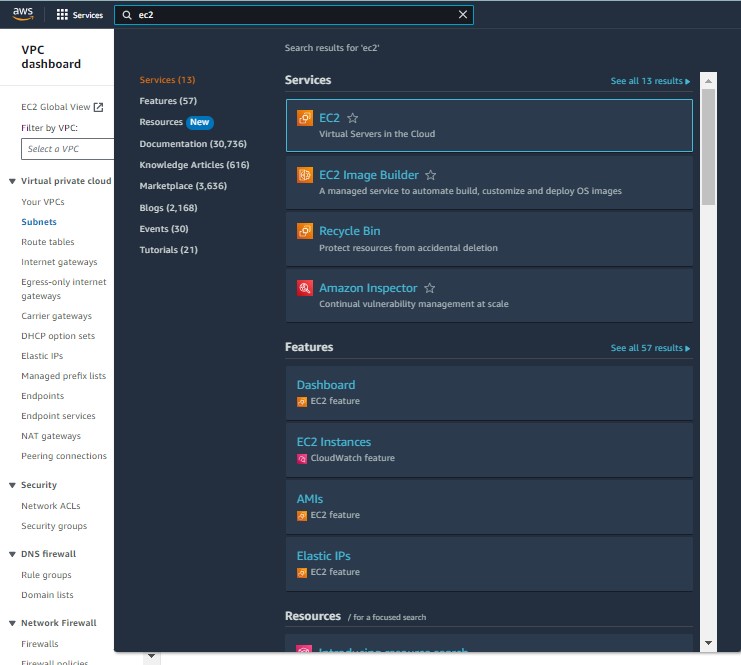
Now give the subnet details

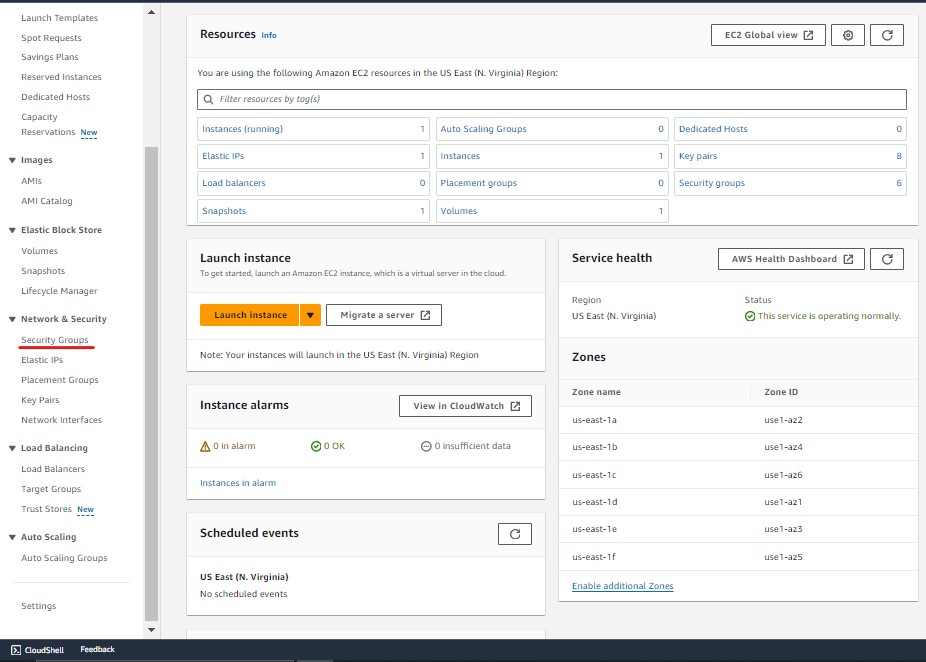


Here we can see our subnet

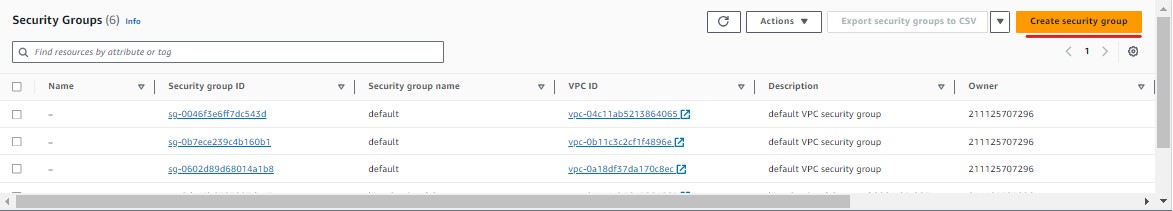


Now go to EC2

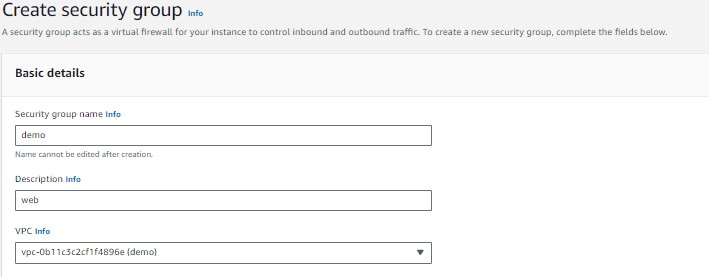


Now click the security group 

Now click on create security group



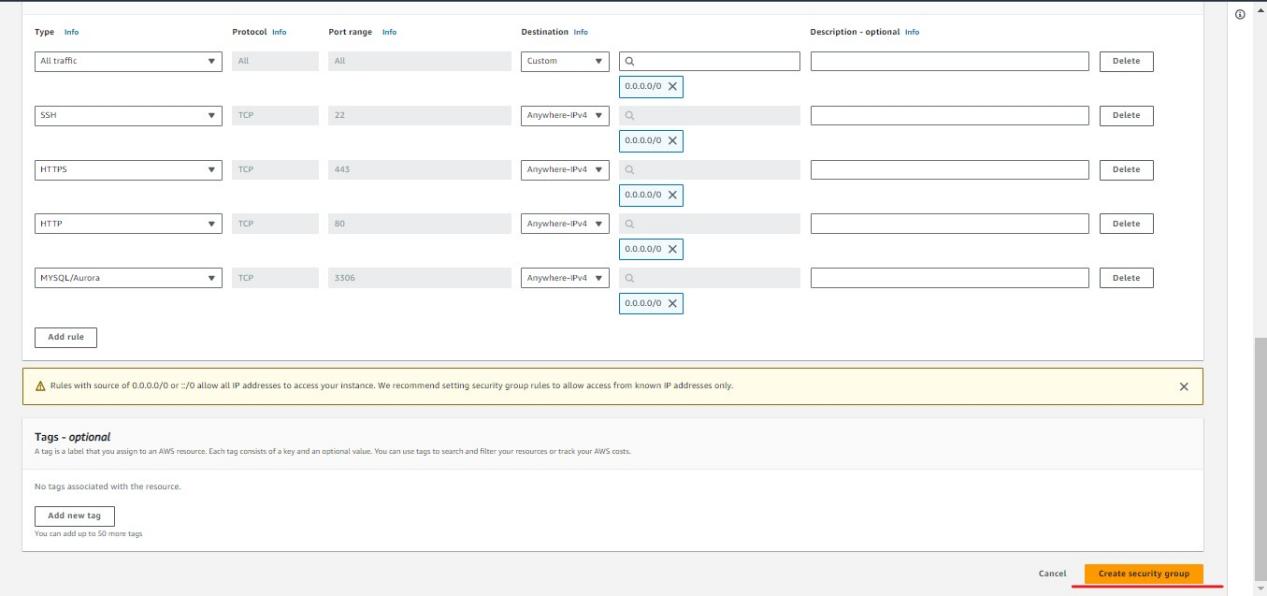
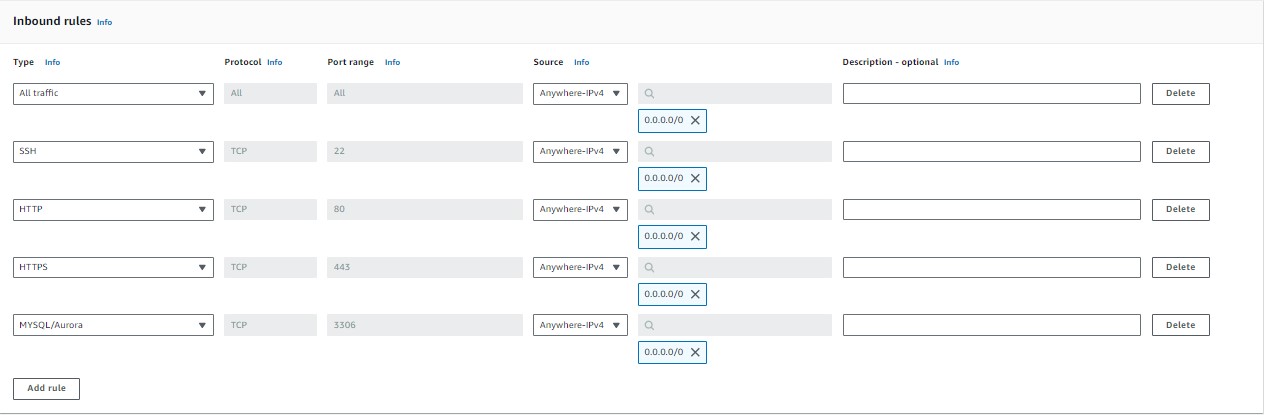
Give the security group details



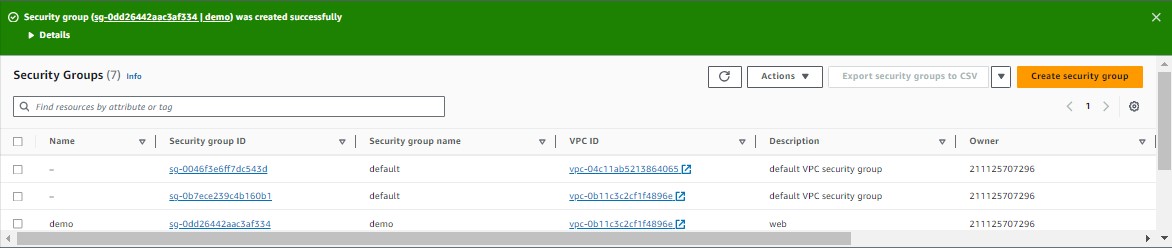
Add the inbound rules of our group   
>http - 80

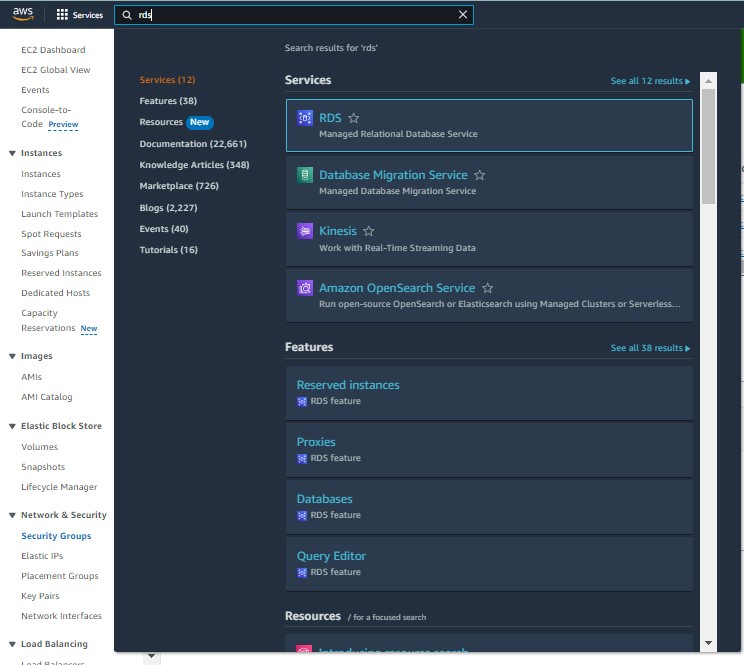
>https - 80

>ssh - 22  
>MYSQL - 3066



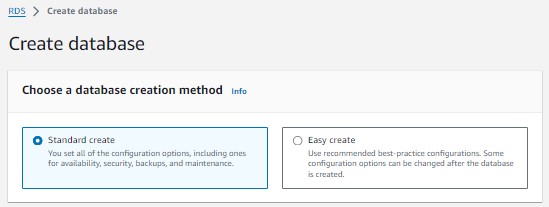
Here we can see our security group



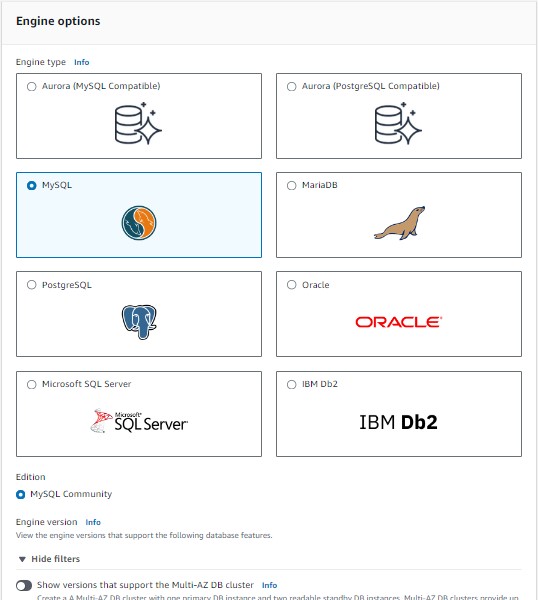
Now go to RDS 

Click on Create database



Now select the creation method 

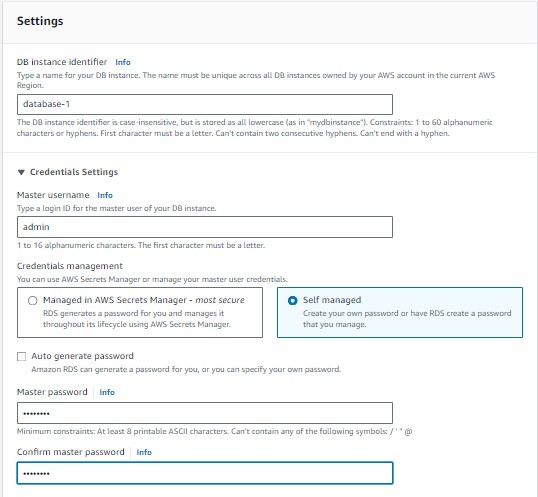
Now select the DB type

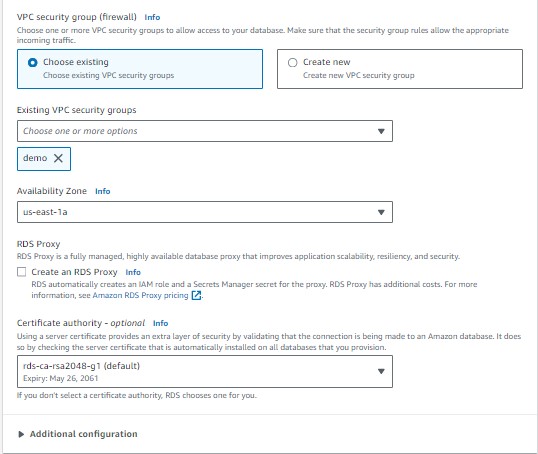


Select the free tier

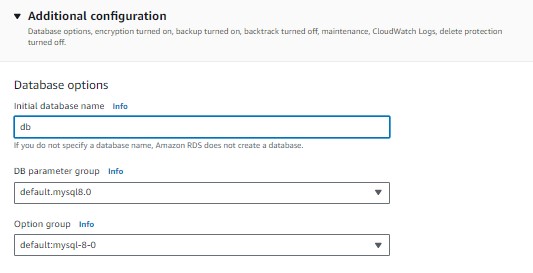


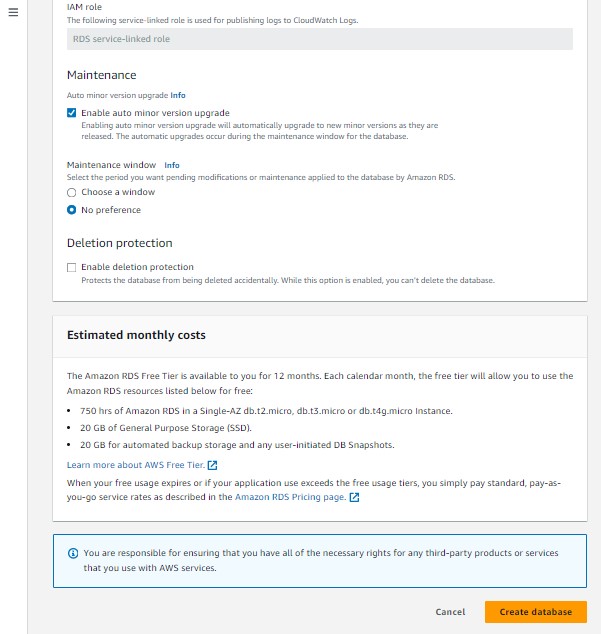
Give the user name and password



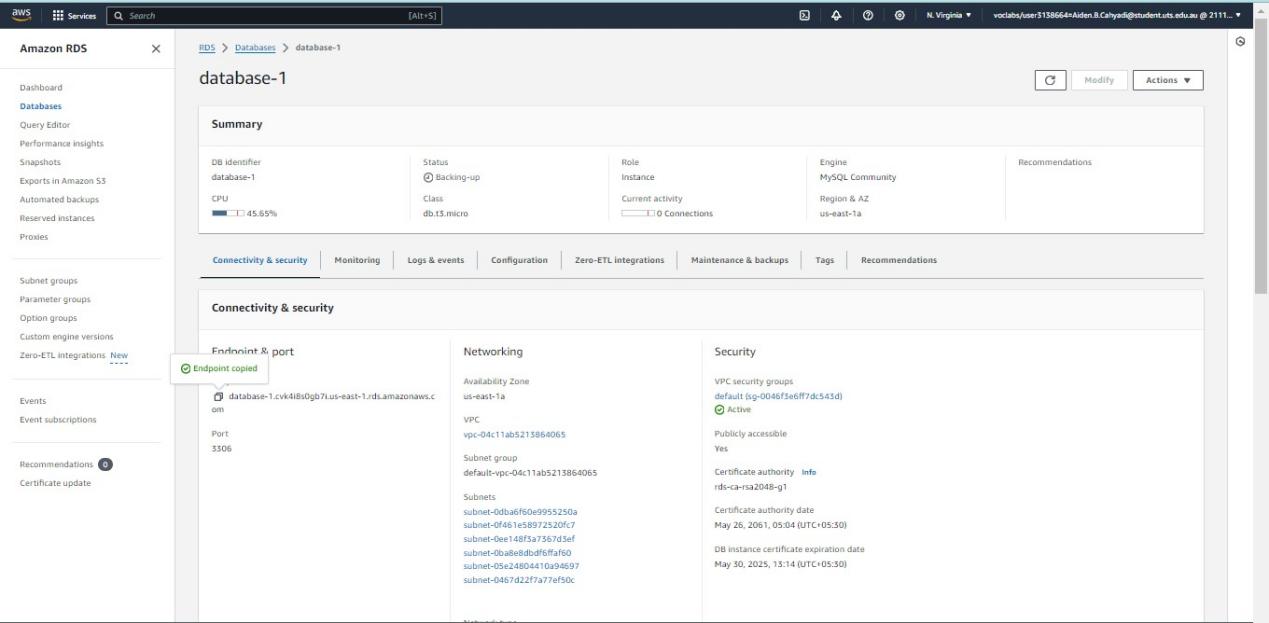
Now give network details 

Now give the db name



Now create the DB name 

Here we can see the DB now copy the DB end port link



Now add it in the link in the php code

<?php

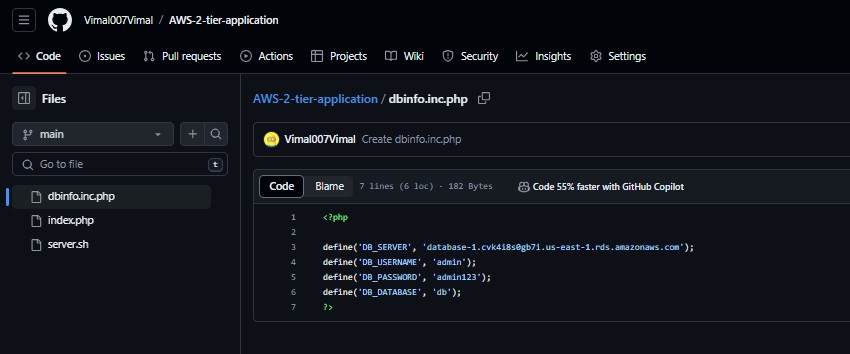
define('DB\_SERVER', 'database-1.cvk4i8s0gb7i.us-east-1.rds.amazonaws.com');

define('DB\_USERNAME', 'admin');

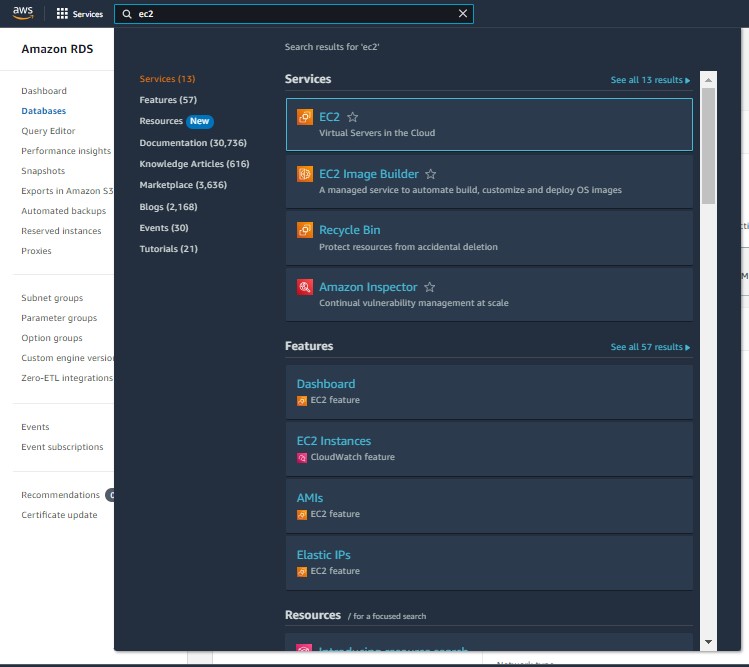
define('DB\_PASSWORD', 'admin123');

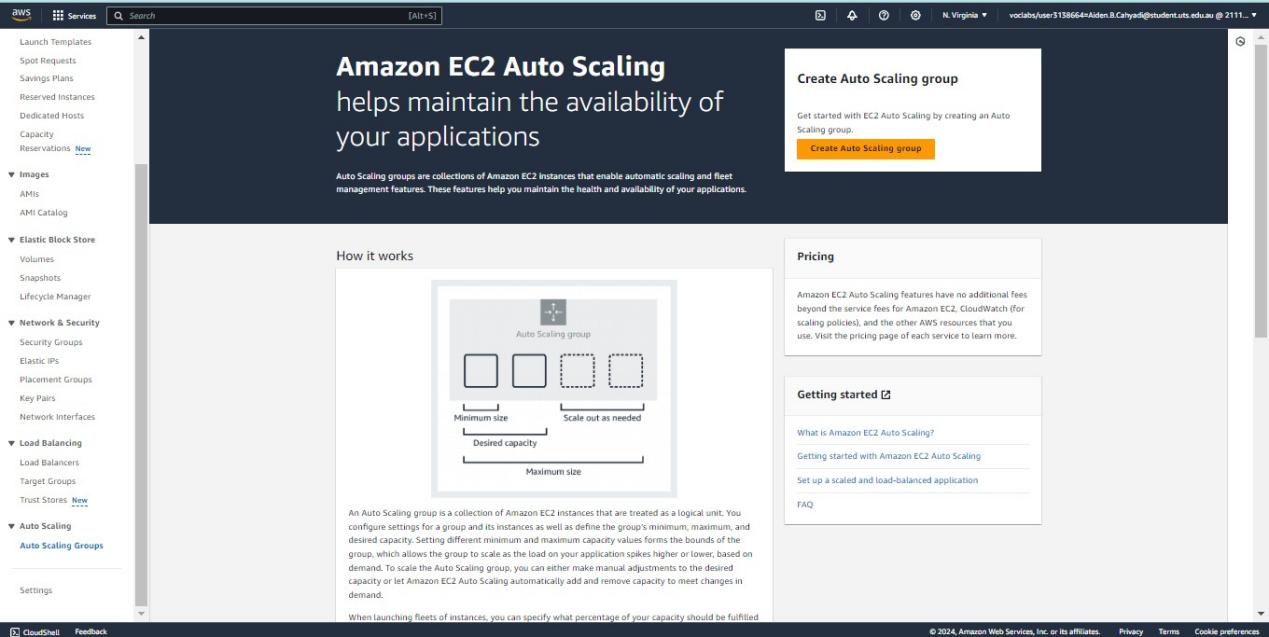
define('DB\_DATABASE', 'db');

?>

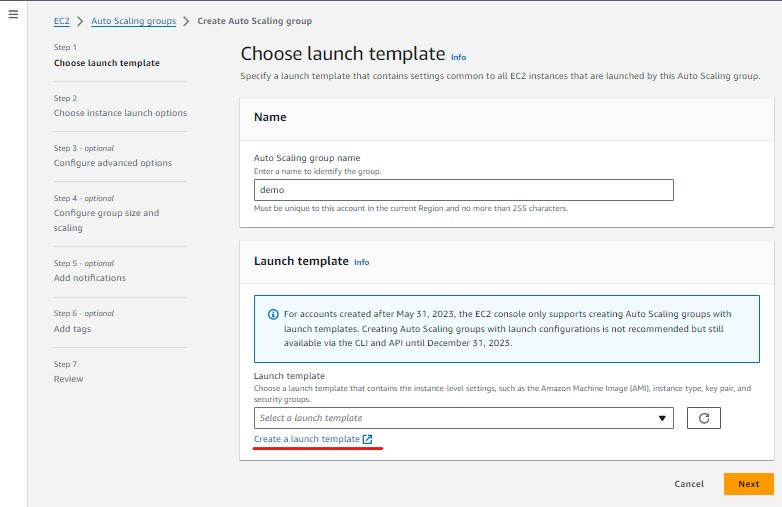


Now go to EC2

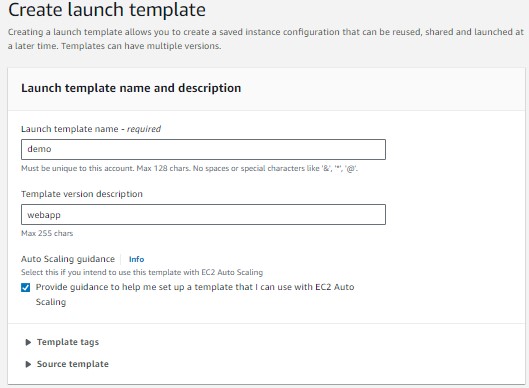


Go to Auto Scaling group 

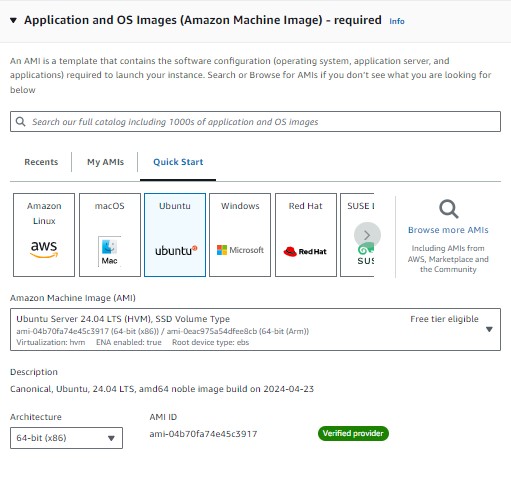
Give the ACG name and click on launch templete

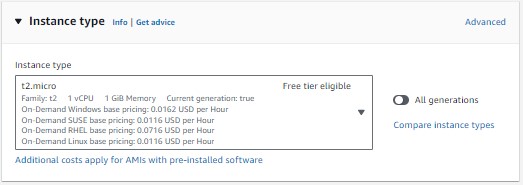


Now give it name

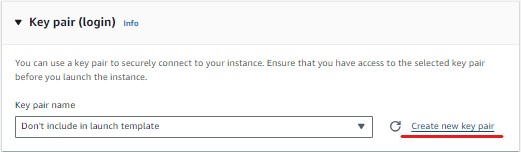


Select the os type - we are selecting ubantu

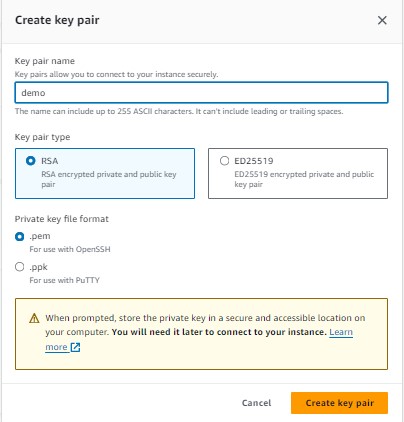


Now select instance type 

Now create nw key pair

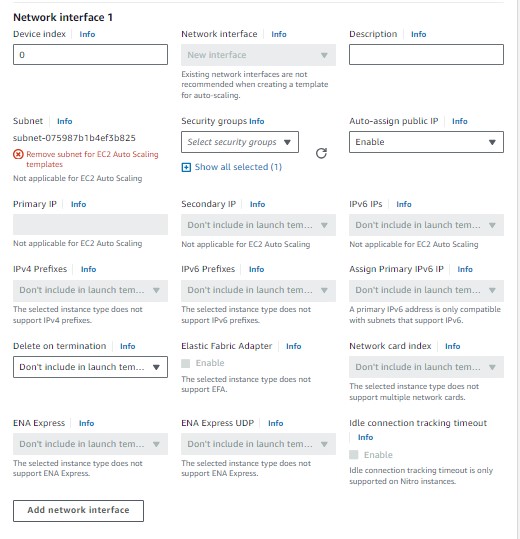


Give it a name and download it

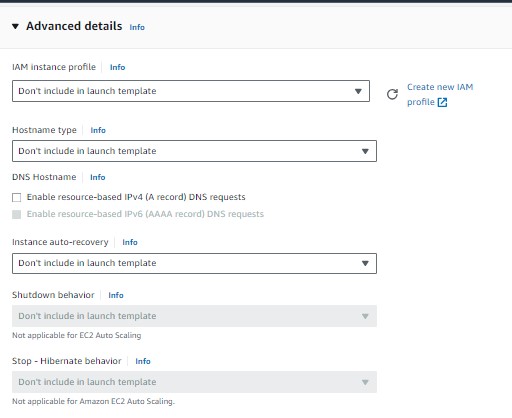


Now give I the network details of our vpc and subnet 

Give the additional details



Go to advanced details



Here we need to give script so all the instance which is created by the ASG - The server will be configure automatically

#!/bin/bash

sudo su

# Update package list

apt update -y

# Install Apache web server

apt install apache2 -y

# Install PHP and necessary PHP modules

apt install php libapache2-mod-php php-mysql -y

# Install MySQL client

apt install mysql-client -y

# Install archive management tools

apt install rar unrar zip unzip -y

# Install Git

apt install git -y

# Navigate to the web server's root directory

cd /var/www/html/

# Clone the GitHub repository

git clone https://github.com/Vimal007Vimal/AWS-2-tier-application.git

# Remove the default Apache index file

rm -f index.html

# Move the contents of the cloned repository to the web server's root directory

cd AWS-2-tier-application

mv \* /var/www/html/

# Navigate back and remove the empty directory

cd ..

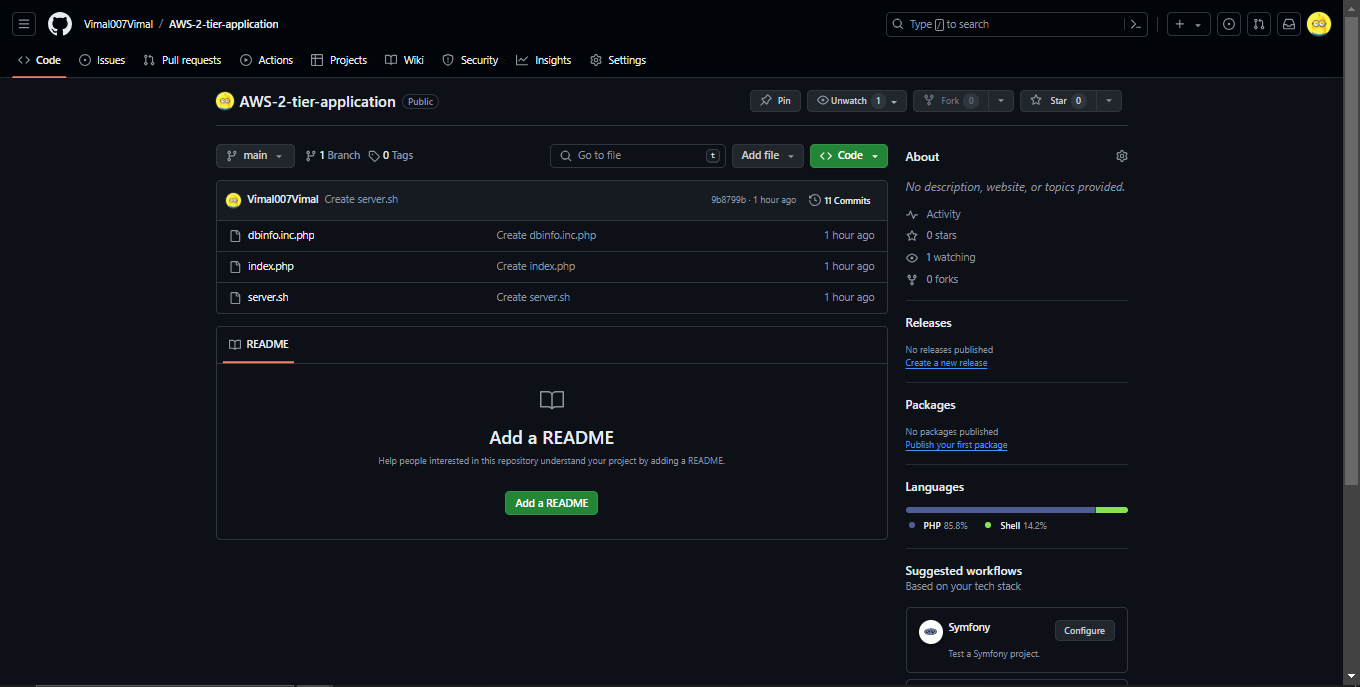
rmdir AWS-2-tier-application

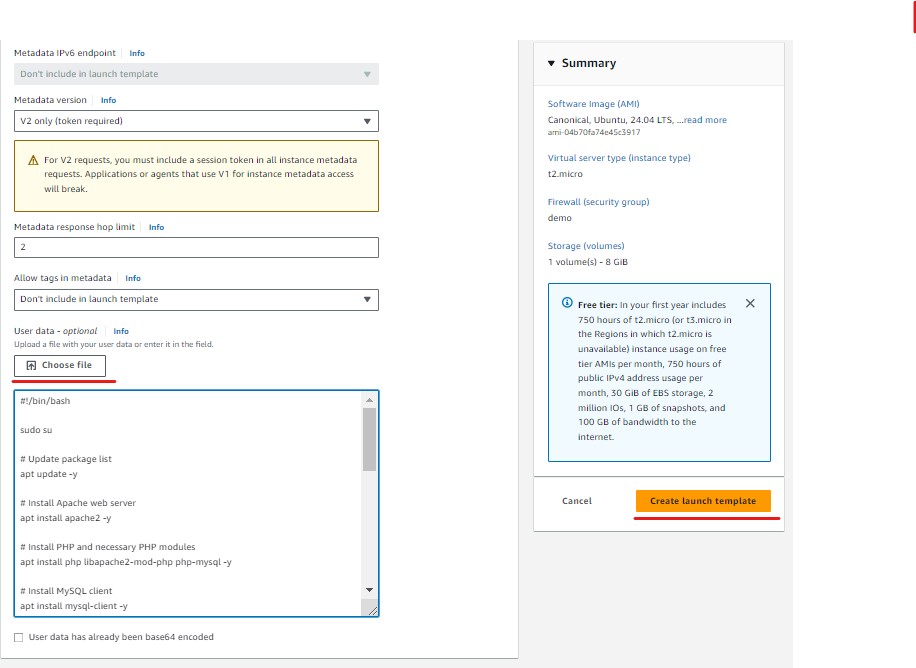
# Restart and enable Apache web server

systemctl restart apache2

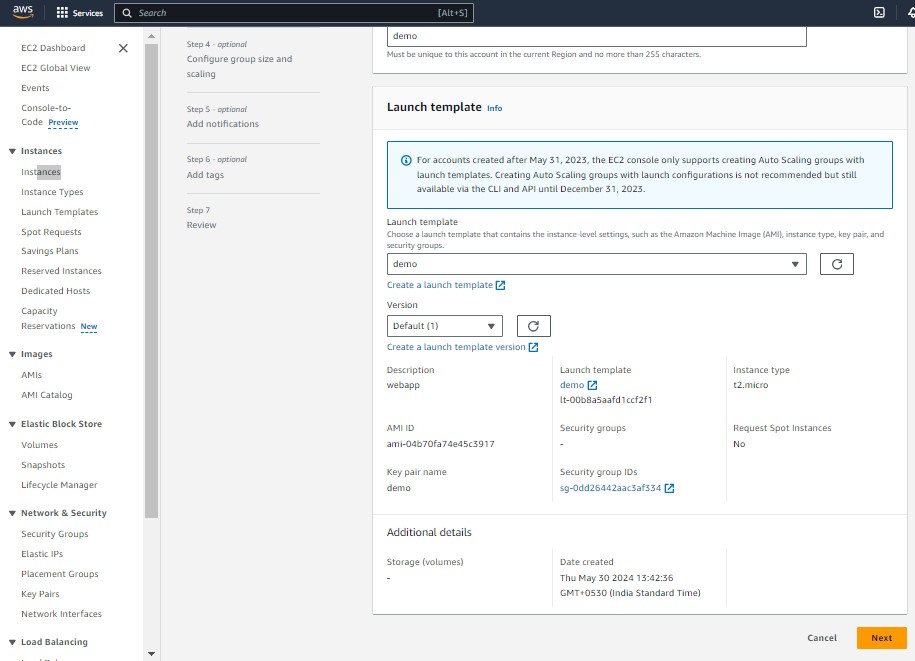
systemctl enable apache2

Here I have the code git link of our application

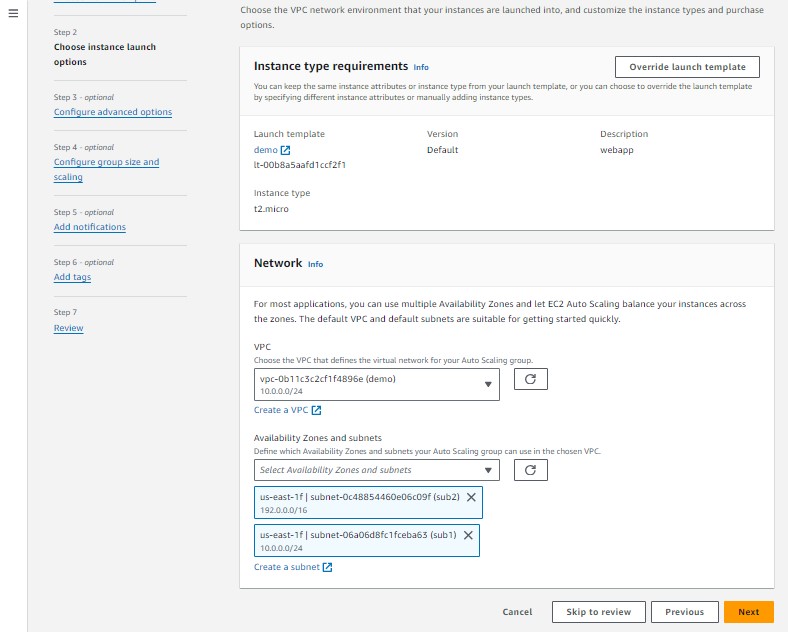




Now create it



Here we can see the templete details - give next

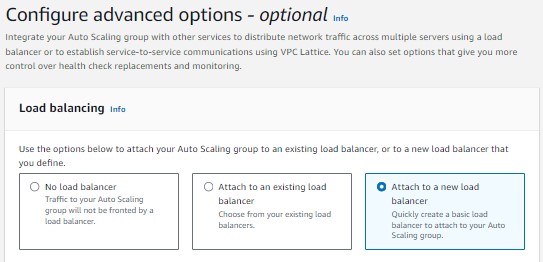


Here we need to give the network details - our vpc and subnet

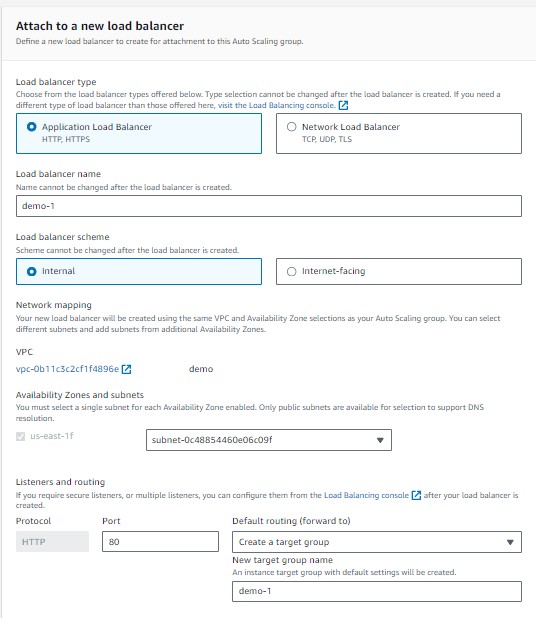
And give next

Here we need to LB(load balancer) details

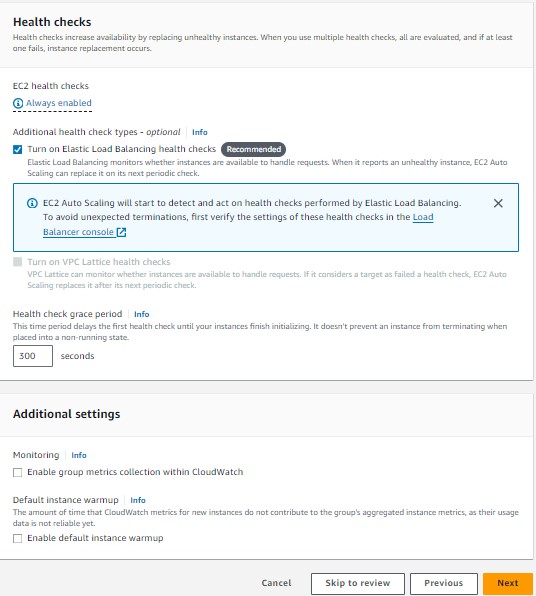
Give new load attach a new load balancer



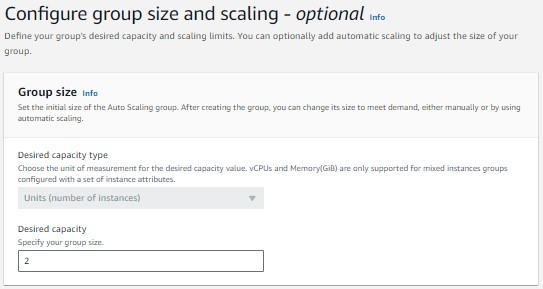
Select the type its network details



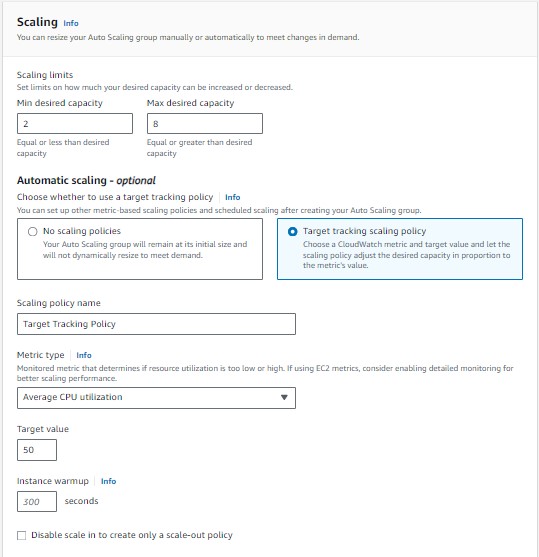
Click on health checks and give next

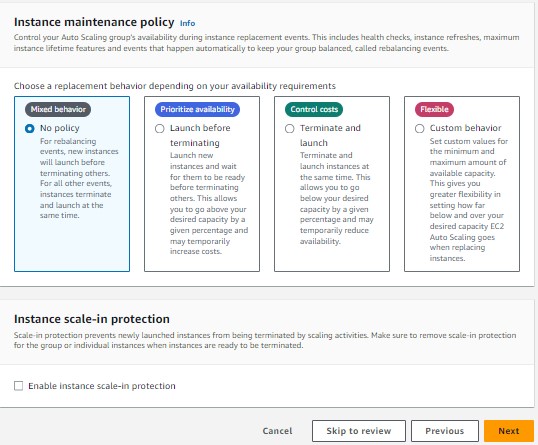


Now give the group size

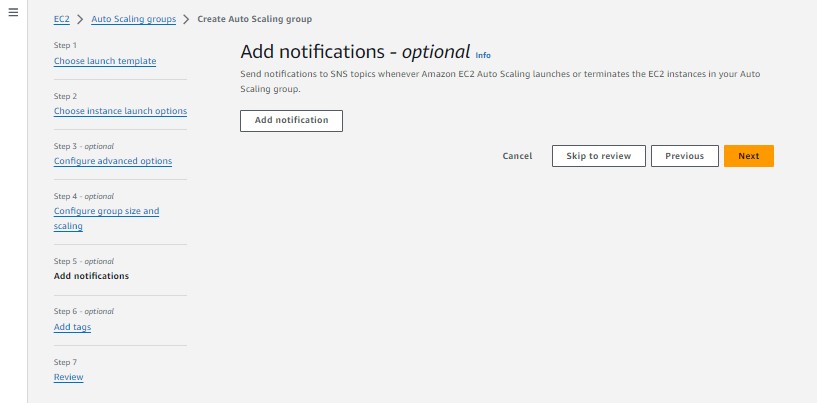


Now give the scaling details and give automatic scaling policy





give next here



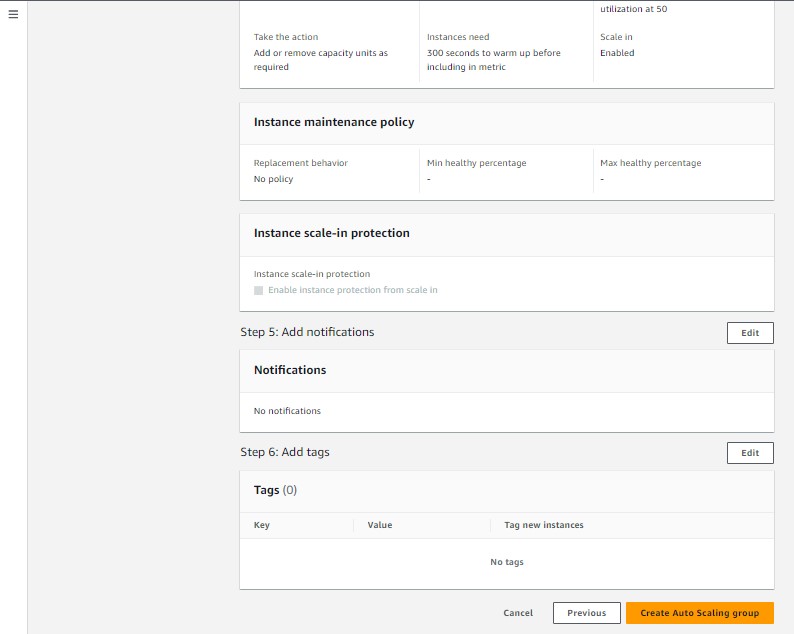
Give next here



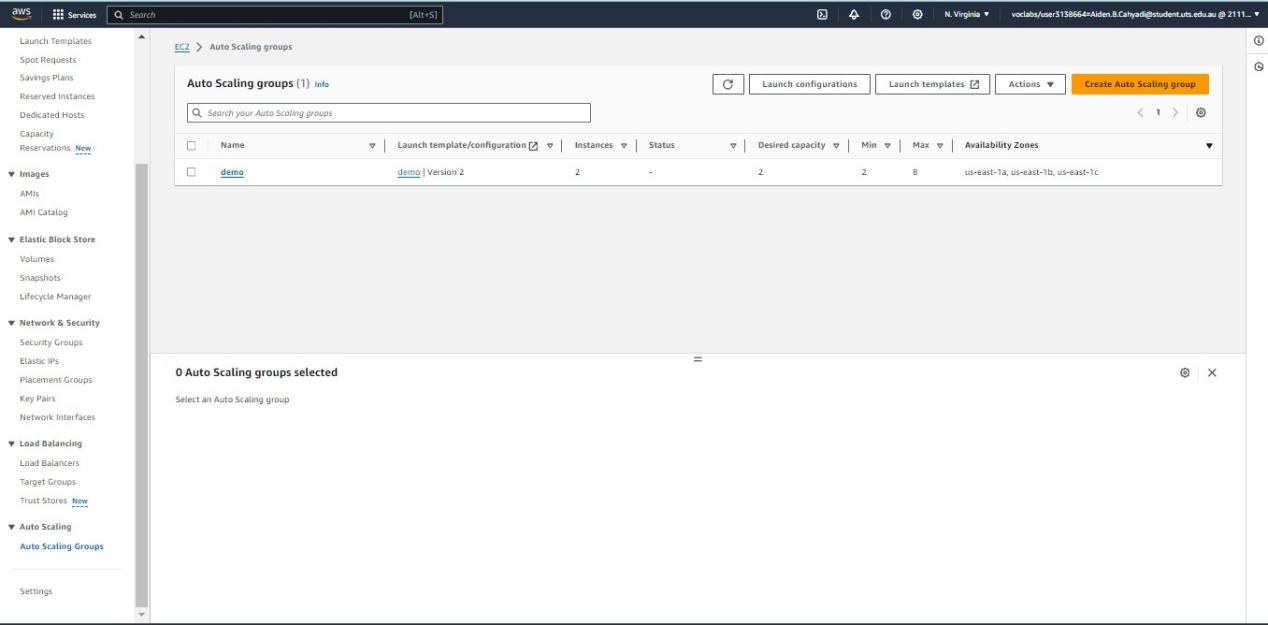
Now review the details



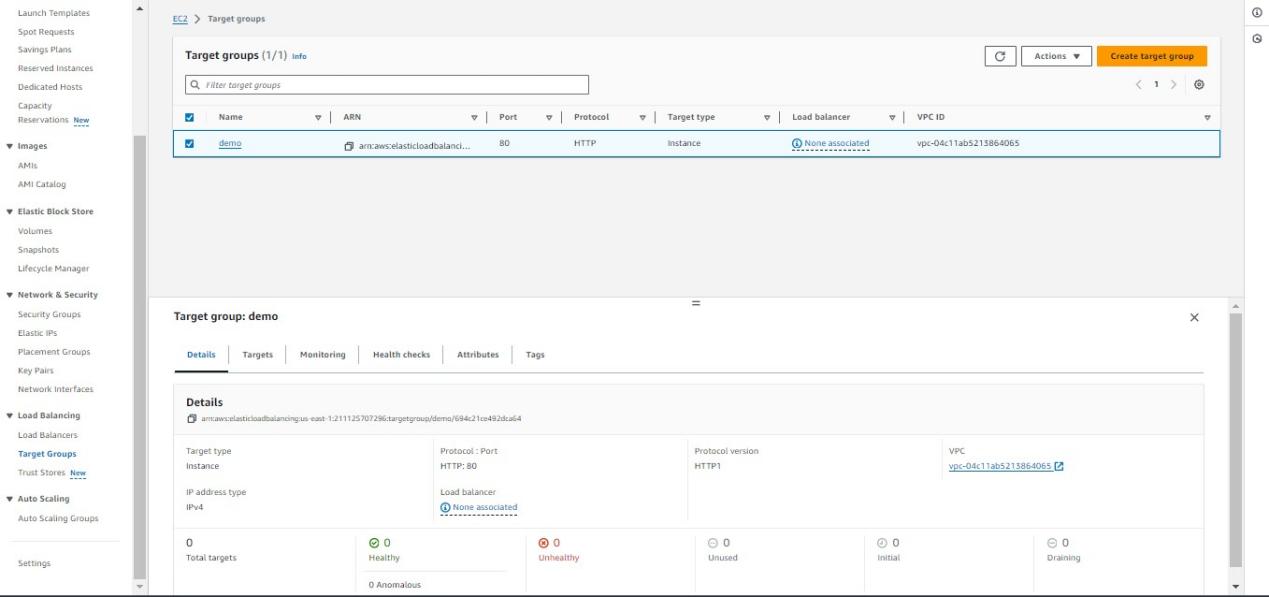
Now create it



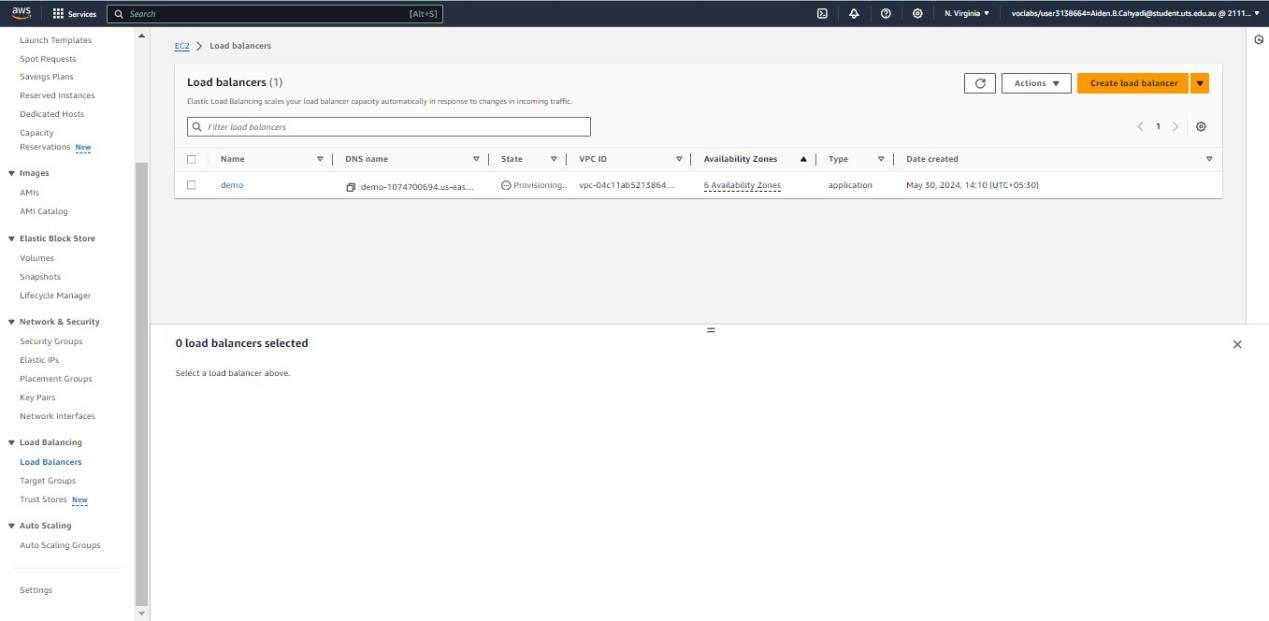
Here we can see the ASG(auto scaling group)



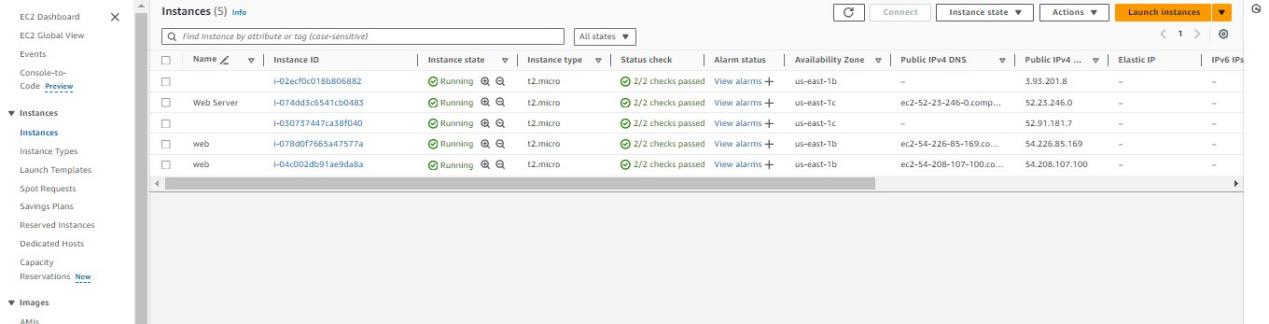
Here we can see our Target groups



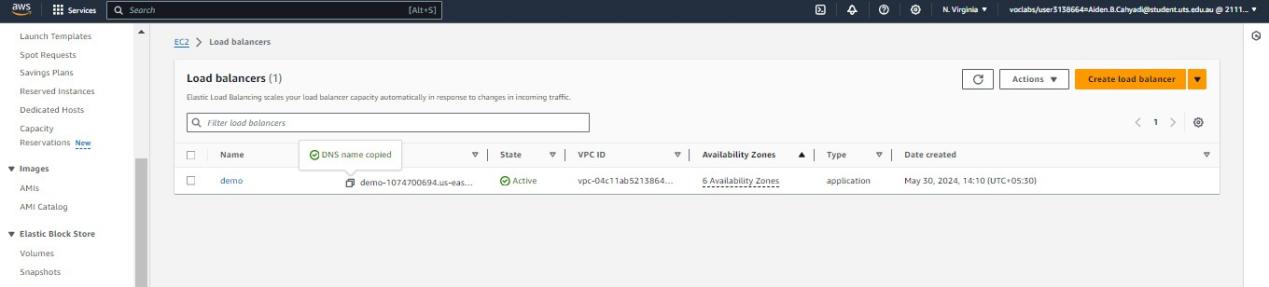
Here we can see our load balances



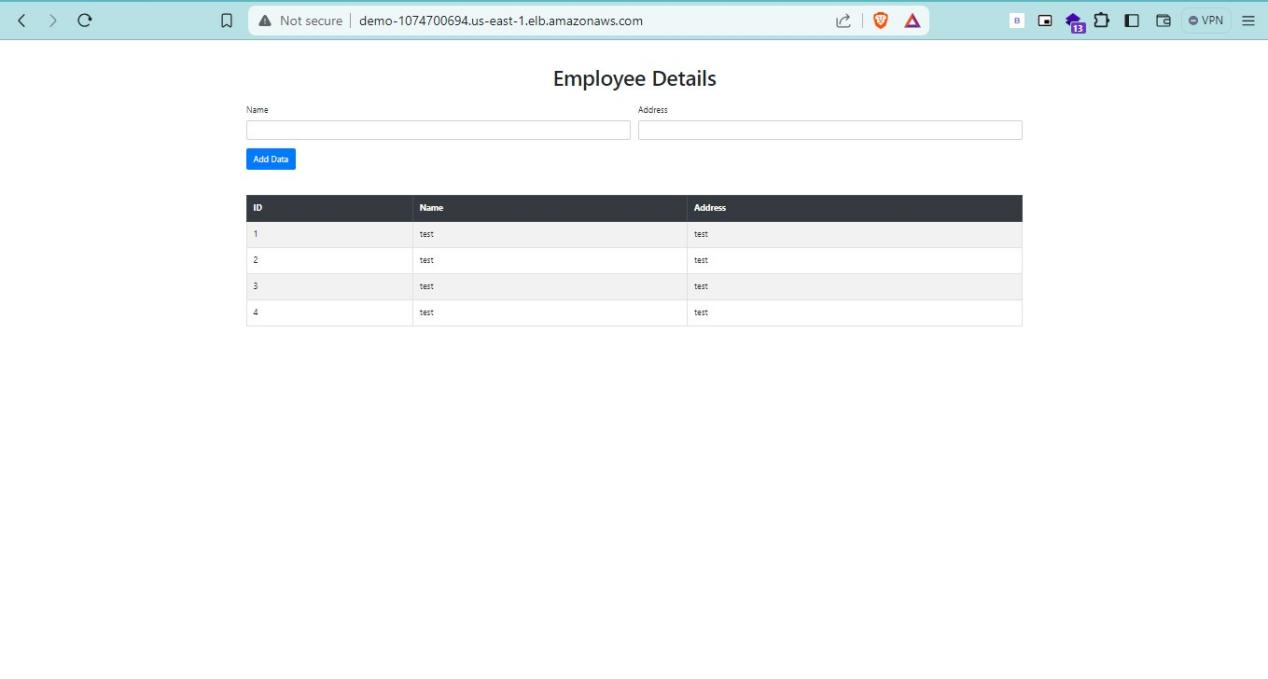
Here we can see our instances



Now copy the DNS of load balances

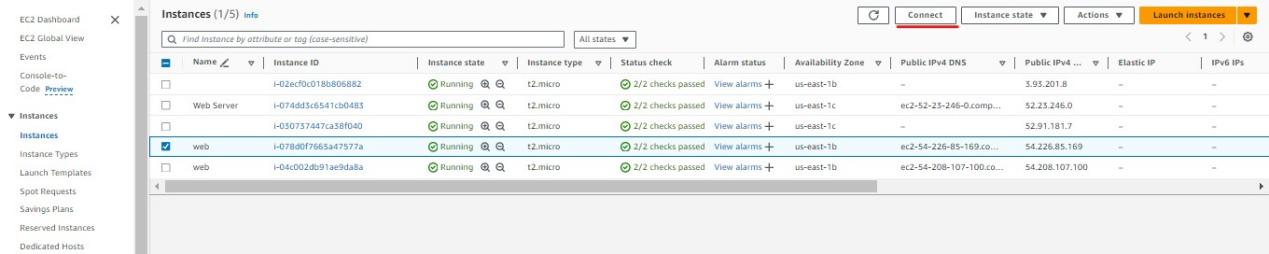


Here check it in the webpage

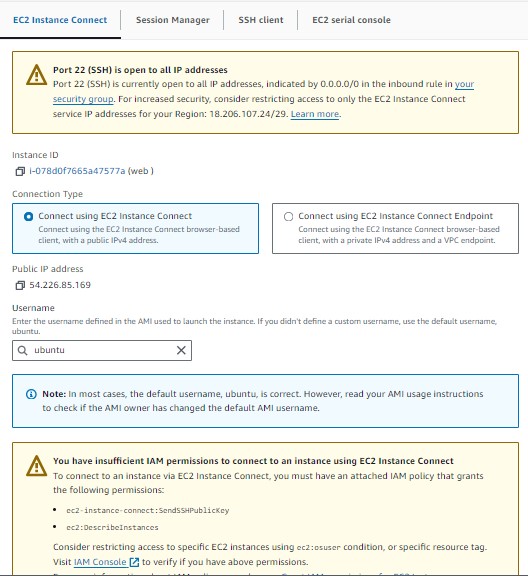


Here we can see the application from LB

Now go to one of the instances and click connect



Select the ec2 instance connect option

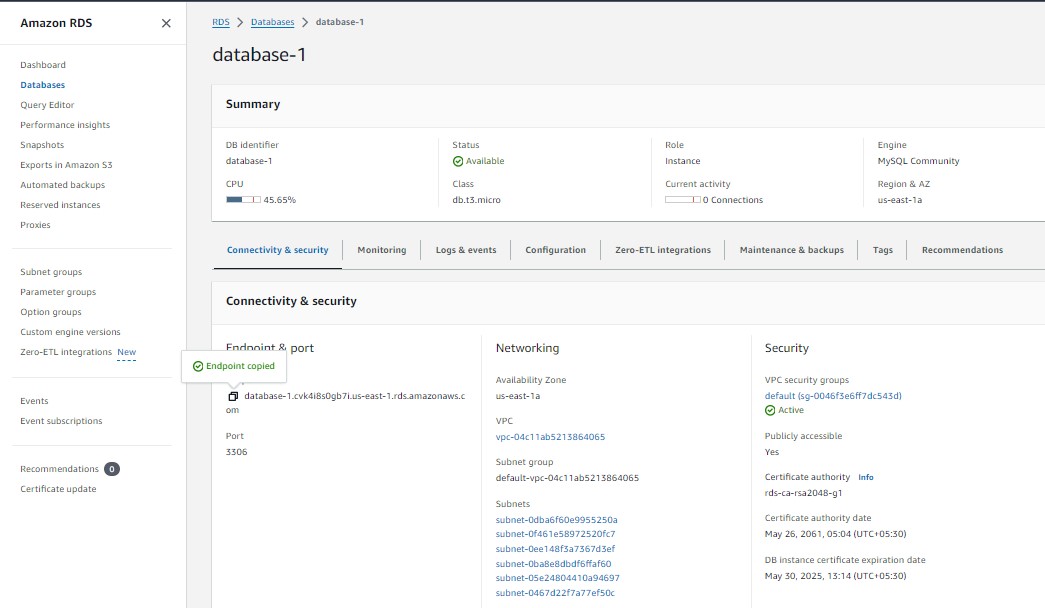




Here we have connected to the ubantu instance go to root user

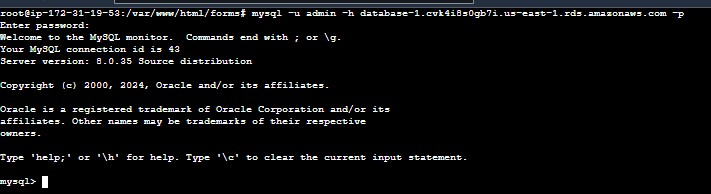
Sudo su

Now go rds and copy the endpoint link



Mysql -u username -h <endpointlink> -p

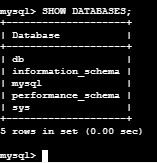
Using this command connect to the DB from the ubantu instance



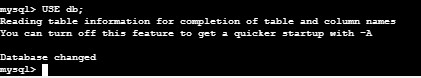
Now we have access to the db

SHOW DATABASES;

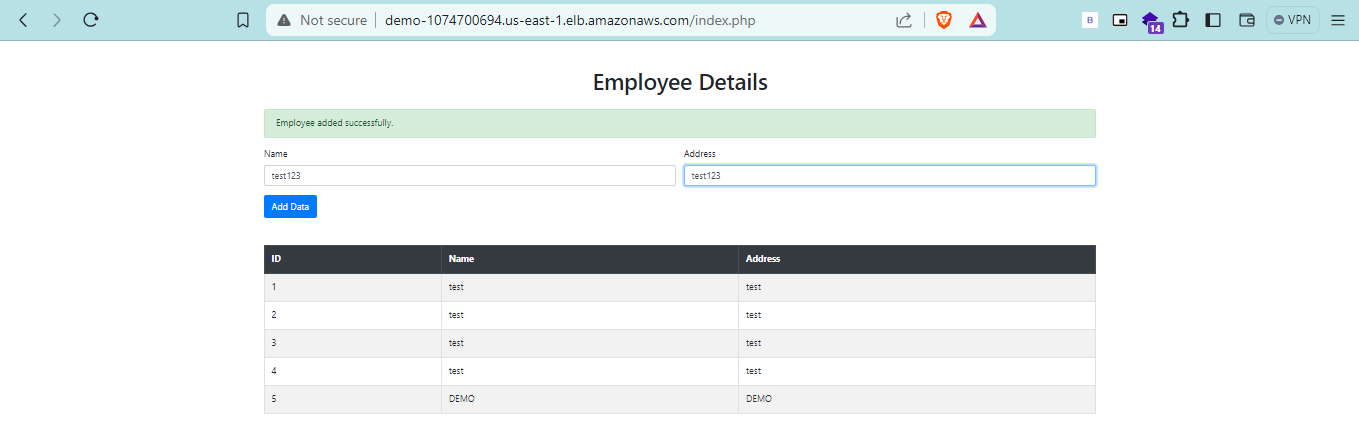
Here w can see our DB



USE databaename;



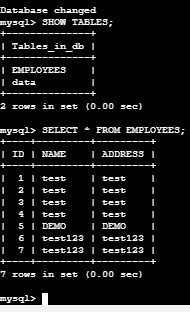
Now go to the LB load balances link and enter some data



Now lets check the data have entered or not

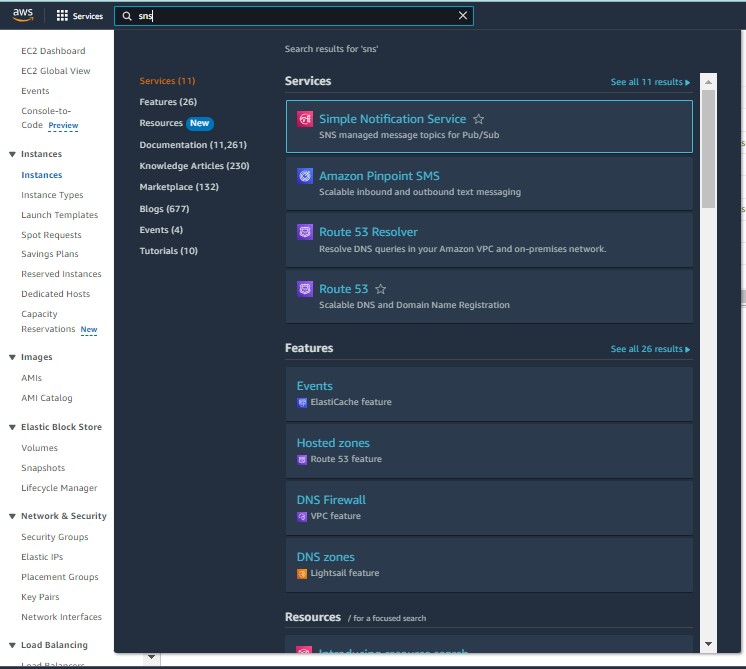
SHOW TABLES;

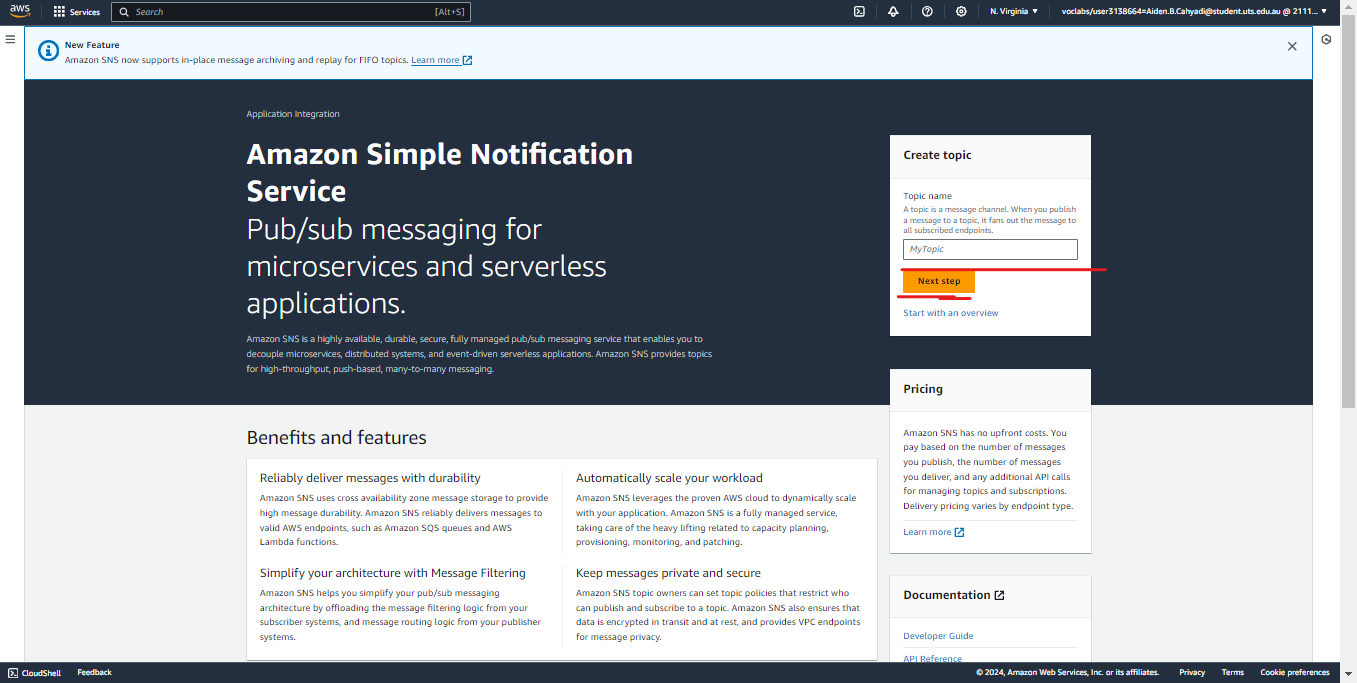
SELECT \* FROM tablename;

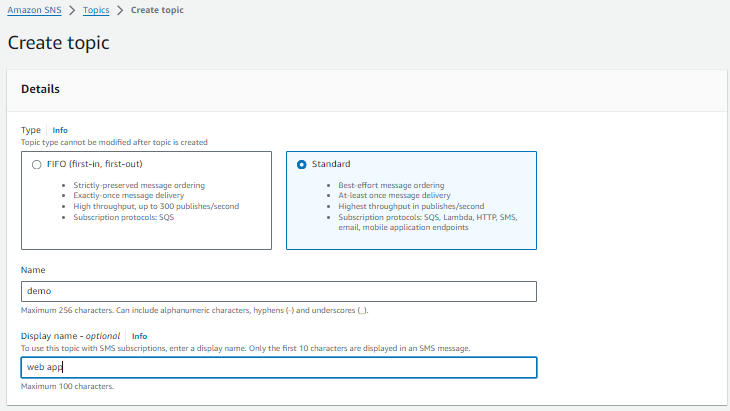


Now our data have entered in our DB

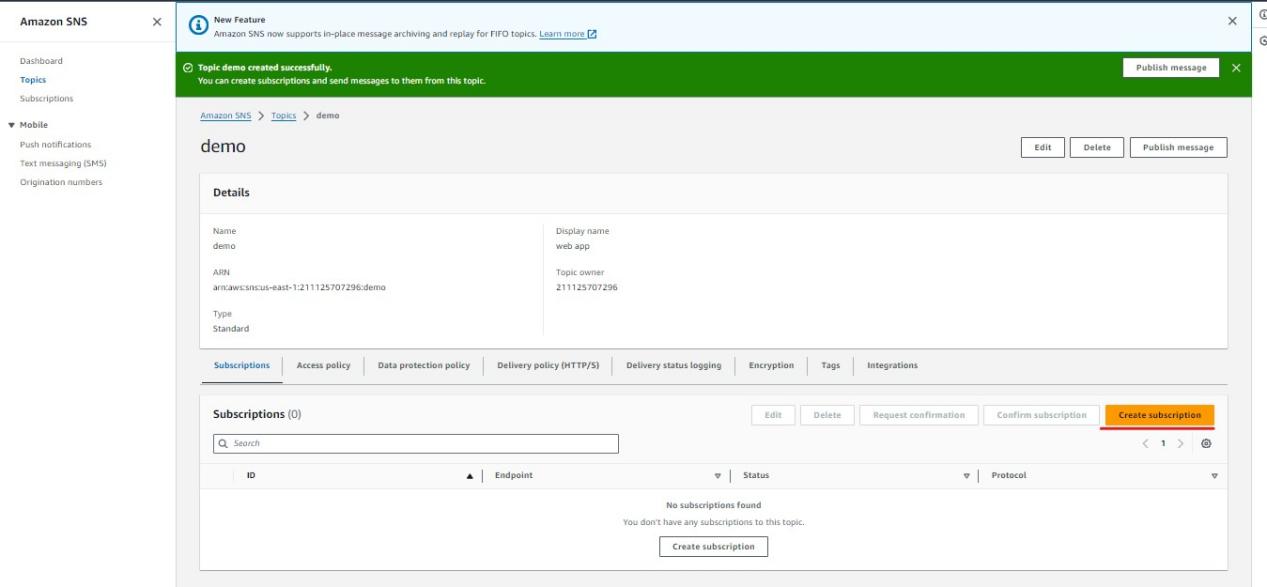
Now go to SNS service



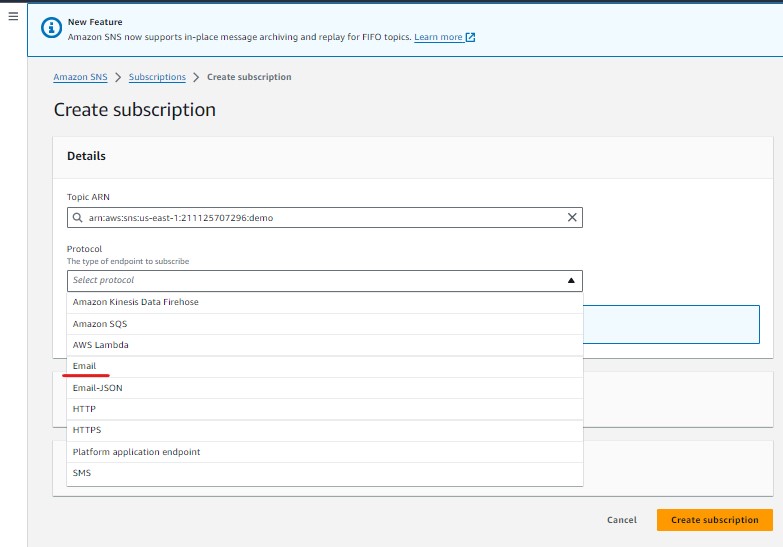
Give the name of topic   
  




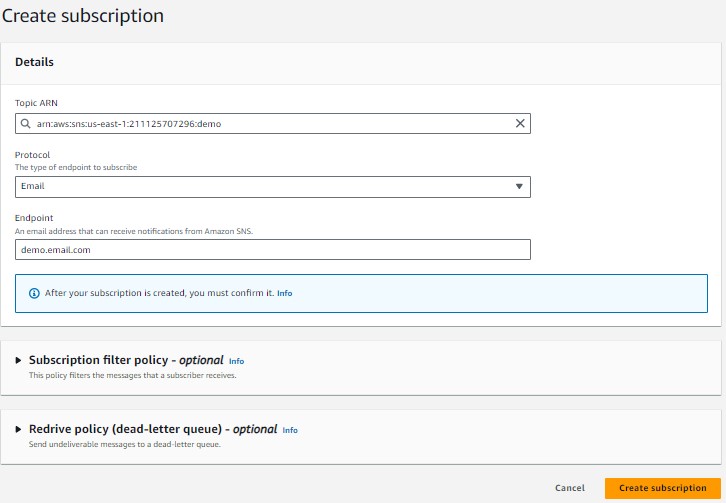
Select topic type and give name and give create topic



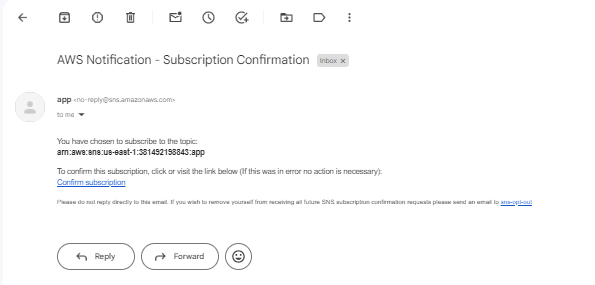
Here select the type of the service

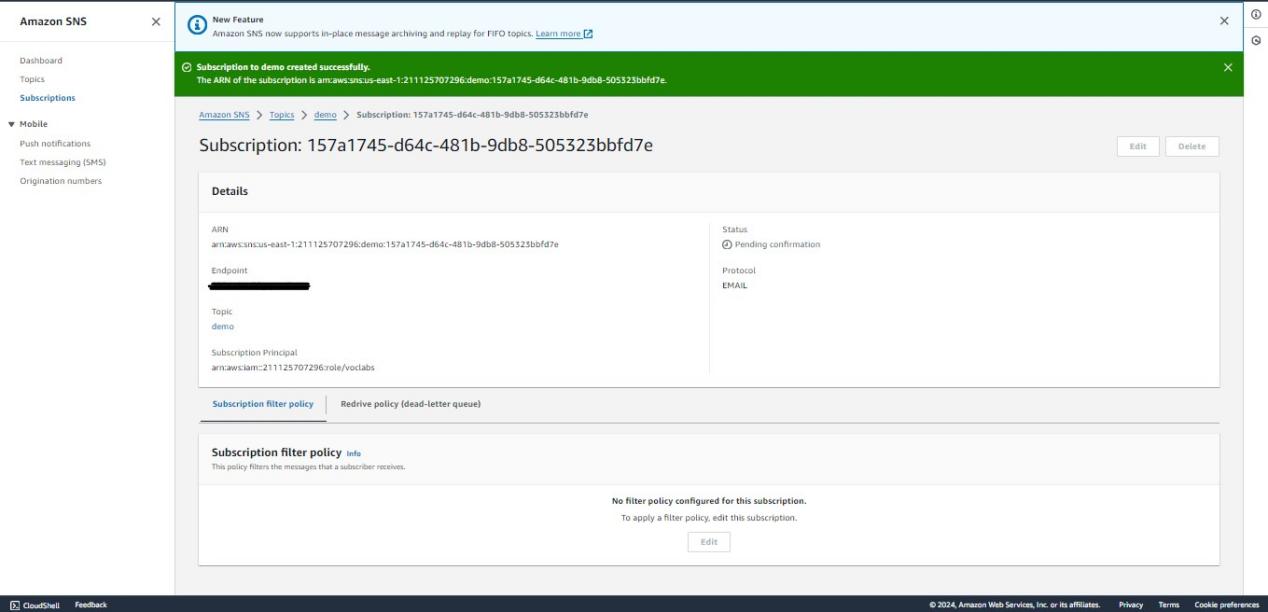


Give the email to it and click create subscription

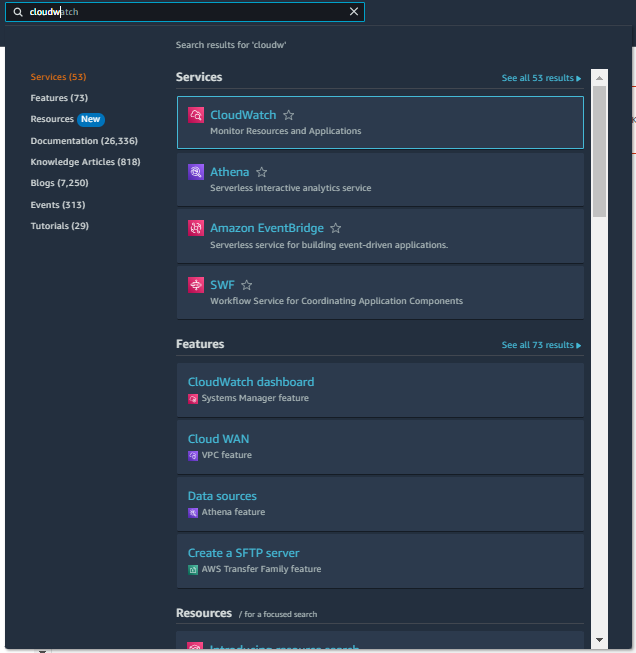


Now you will receive a mail to conform

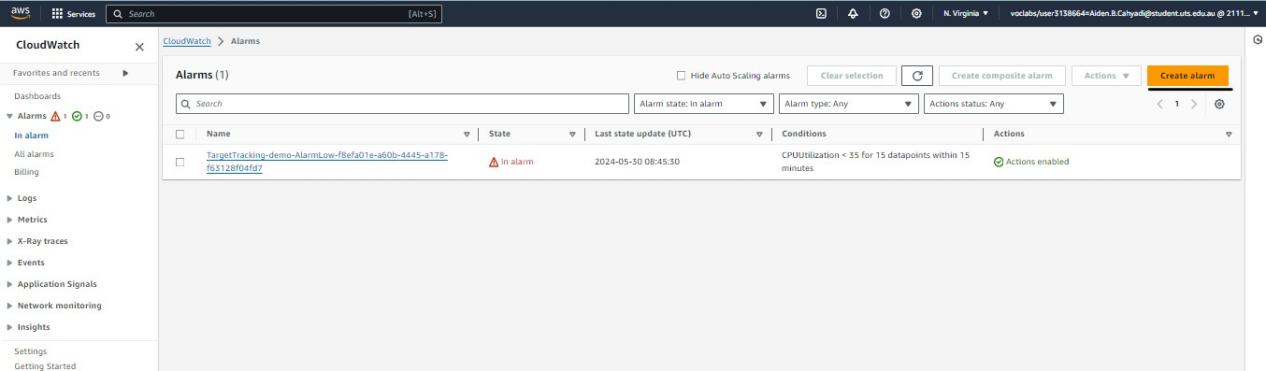


Here we can see our subscription 

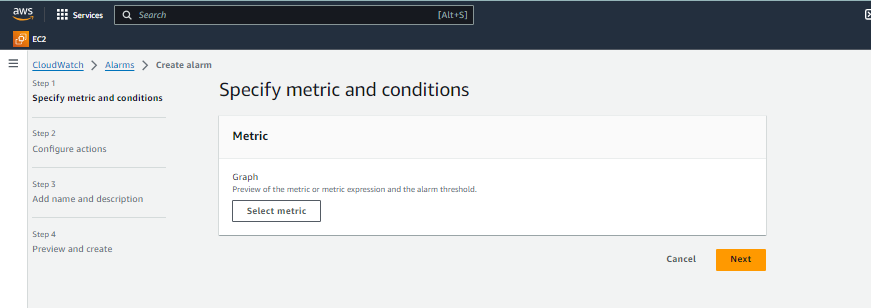
Now we need to set the alarm for that go to cloudwatch



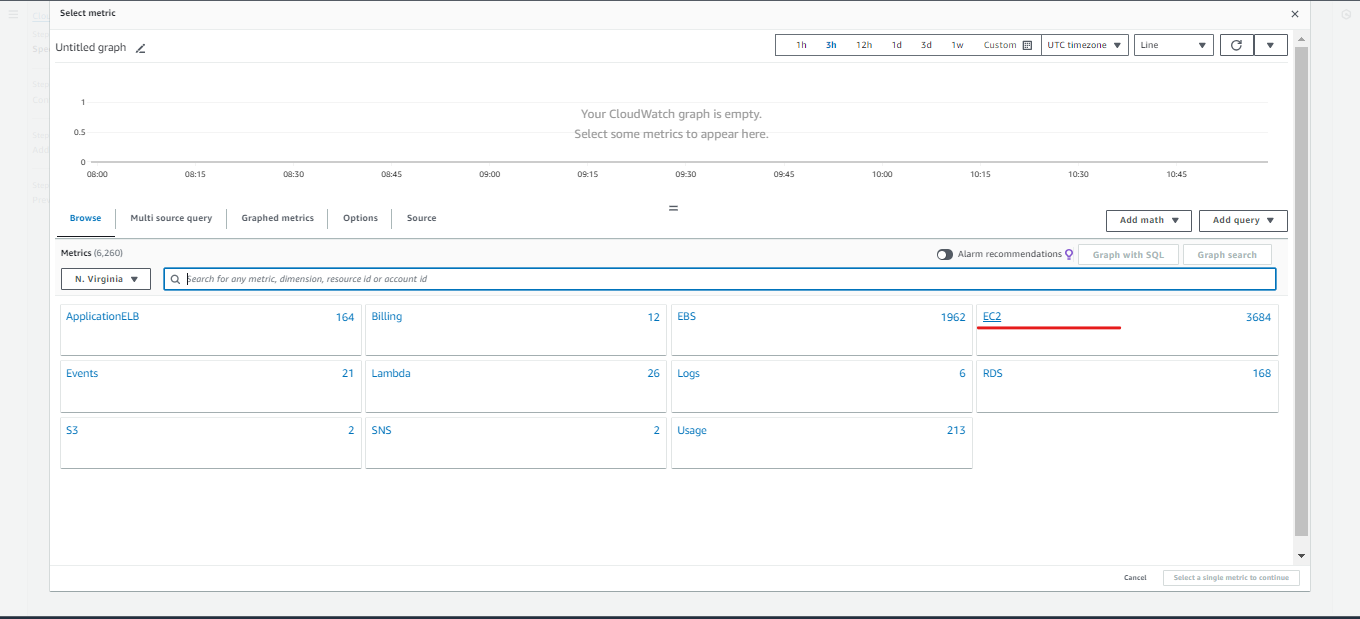
Select create alarm



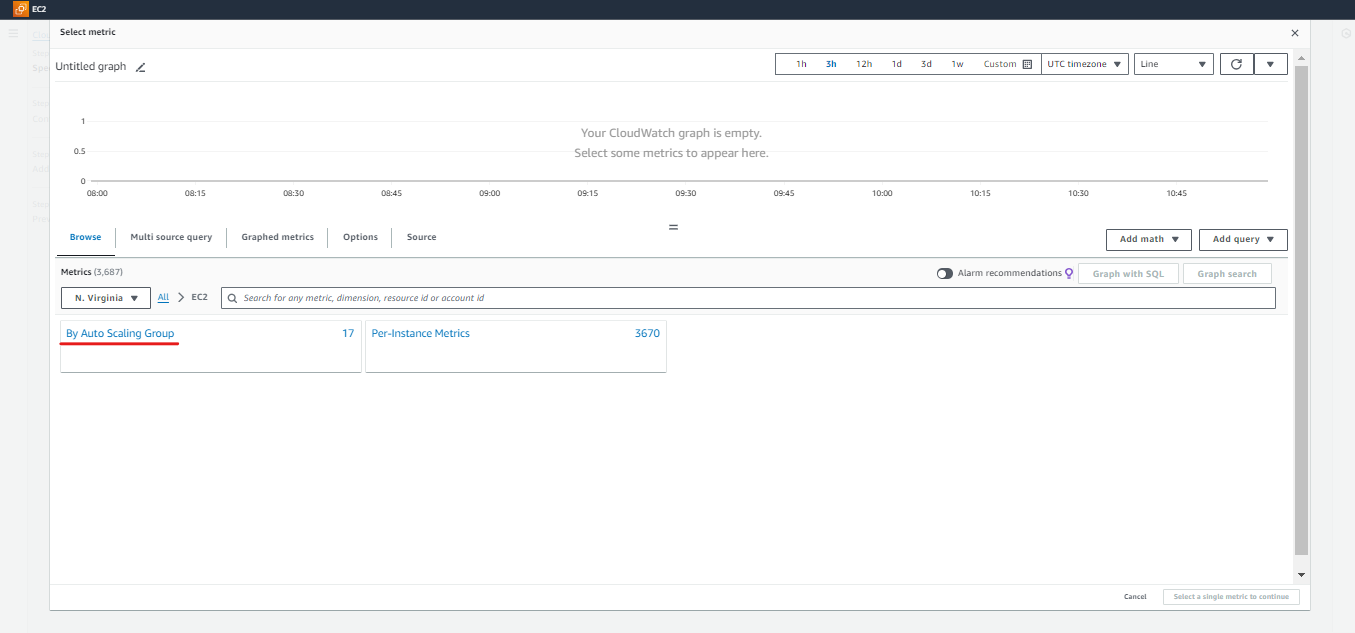
Click on select metric



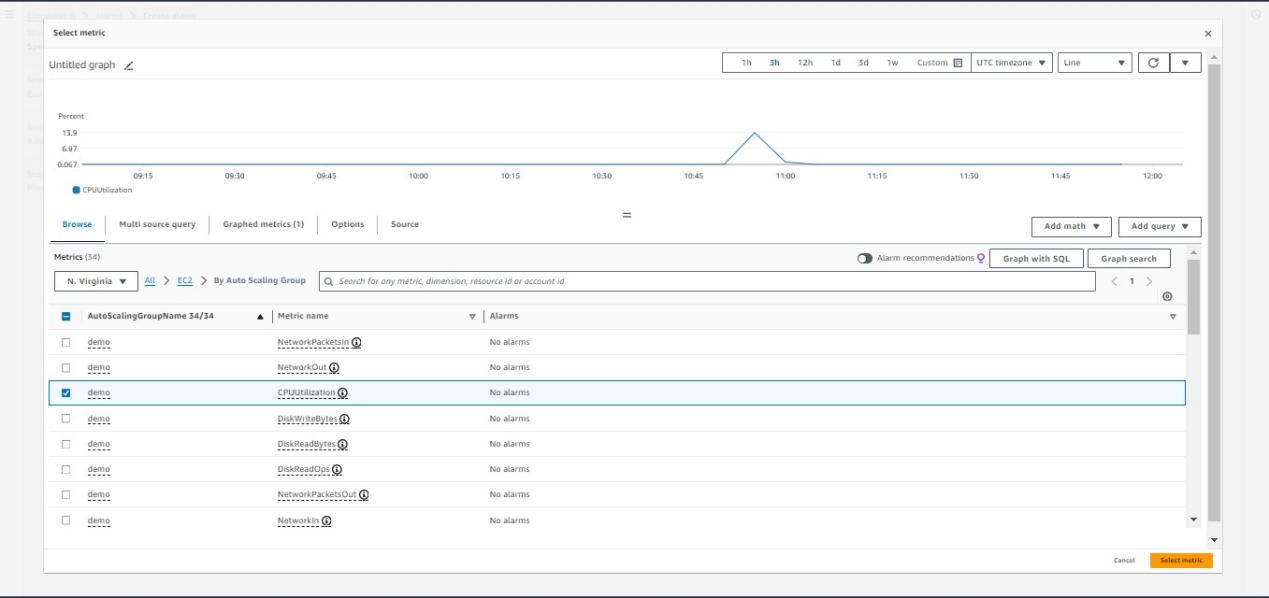
Selcet EC2



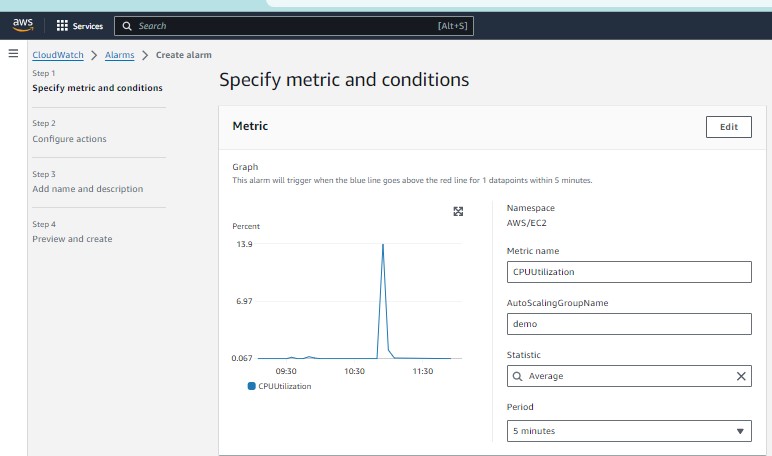
Select auto scaling ( because our instance is running from the template of auto scaling group )



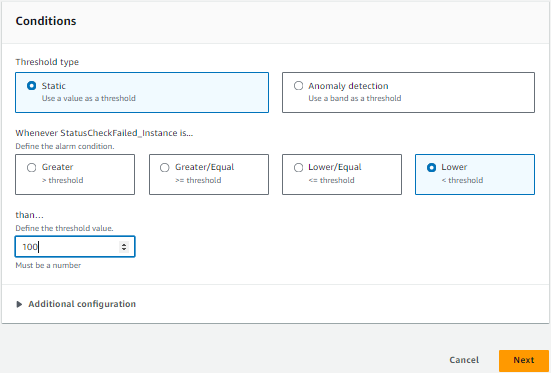
Select the metric you need ( here I have selected status check failed instances)



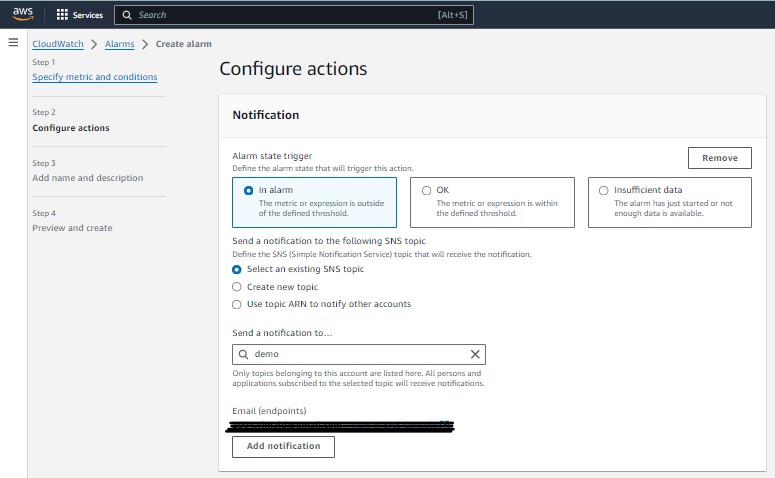
Now give next



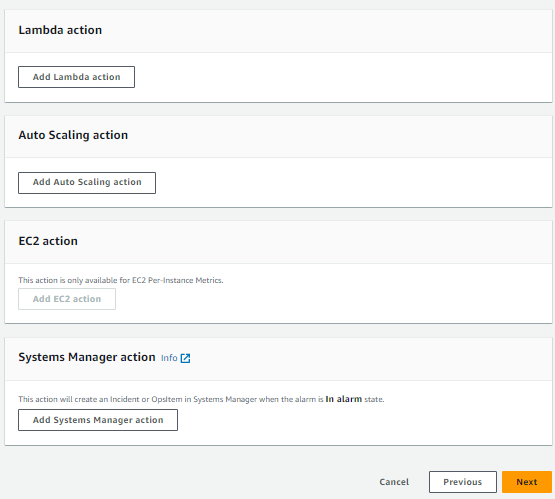
Select the conditions



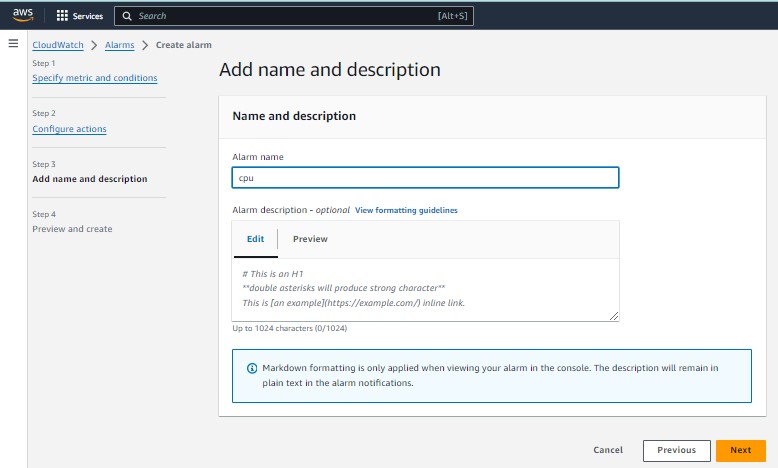
Here select our topic we created earlier

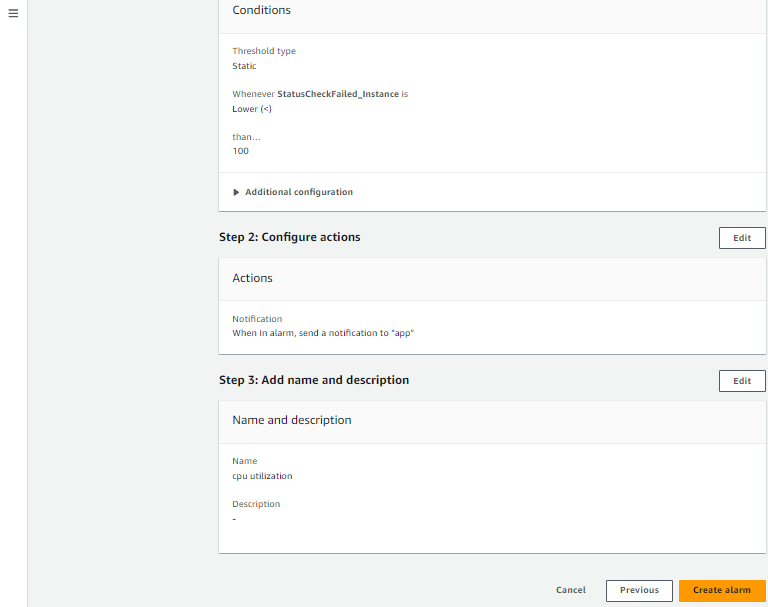


Now we can see our mail

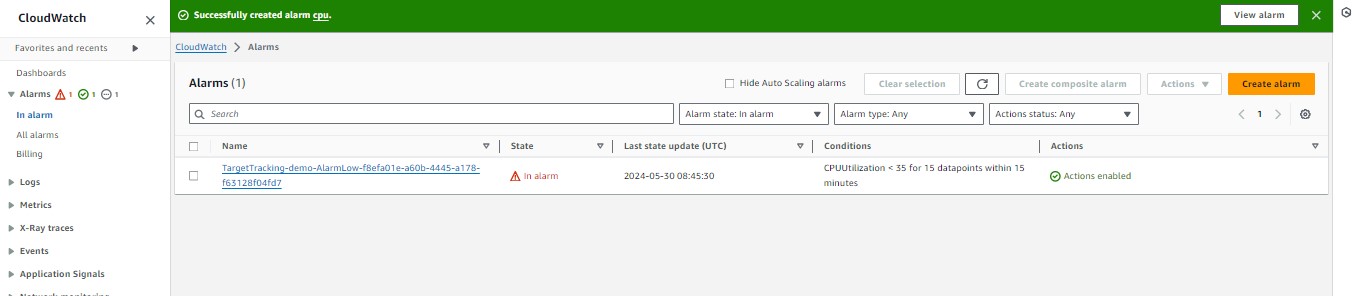


Give a alarm name



Give a review and create it 

Here we can see our alarm



In our case if a instance is running with no usage it will be mailed like this 