**Linux/Shell**

1. Shell program to find second largest number in the list.

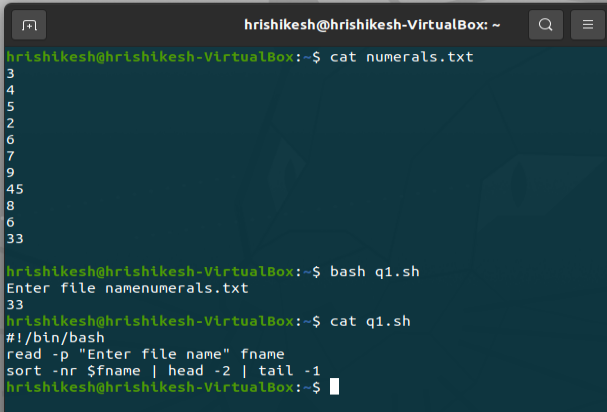
**#!/bin/bash**

**read -p “Enter filename: “ fname**

#Reads the file name and stores in fname

**sort -nr $fname | head -2 | tail -1**

#To sort the file using numerical values and then reverse the sorted list and then print the second highest value using head and tail



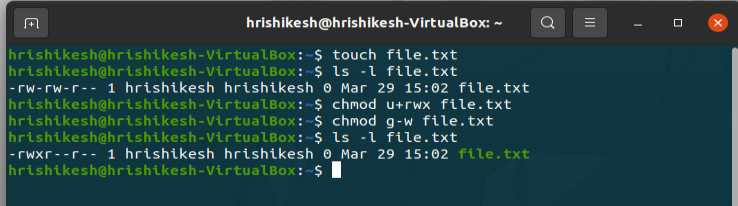
1. chmod

touch file.txt #created file

ls -l file.txt

chmod u+rwx #give owner read write execute

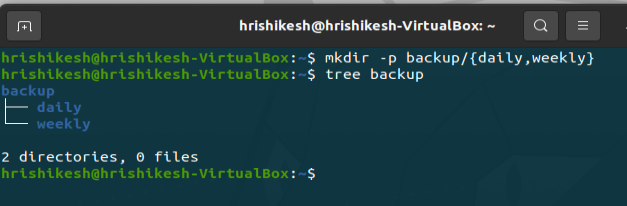
chmod g-w #removed write permissions for group

ls -l file.txt #check file permissions

1. Directory

mkdir -p backup/{daily,weekly} #created directory backup with daily, weekly as subdirectories

tree backup



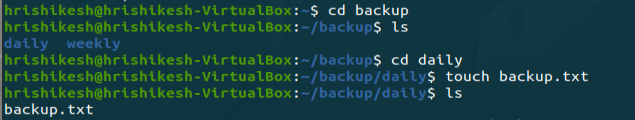
cd backup

ls

cd daily

touch backup.txt #created backup.txt file under daily

ls

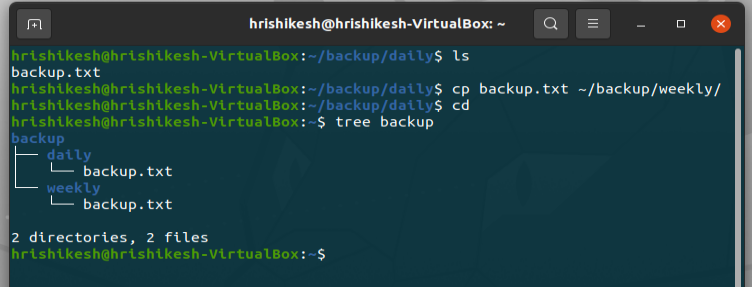


ls

cp backup.txt ~/backup/weekly #copied backup.txt to backup/weekly

cd

tree backup



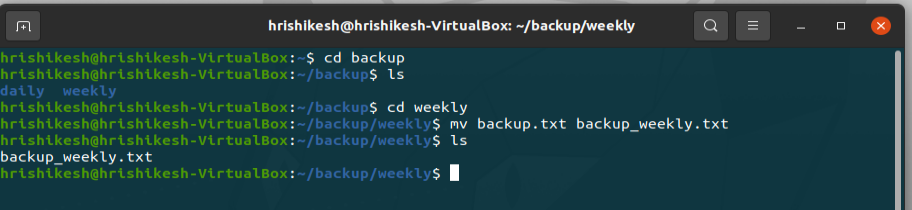
cd backup

ls

cd weekly

mv backup.txt backup\_weekly.txt #renamed backup.txt to backup\_weekly.txt which is under weekly

ls



ls

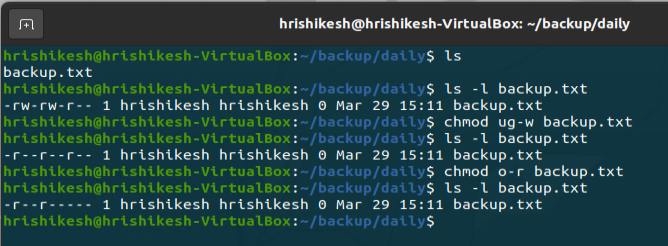
ls -l backup.txt #checked for existing permissions of file backup.txt

chmod ug-w backup.txt #removed write permissions for user and group

ls -l backup.txt

chmod o-r backup.txt #removed all permissions for others

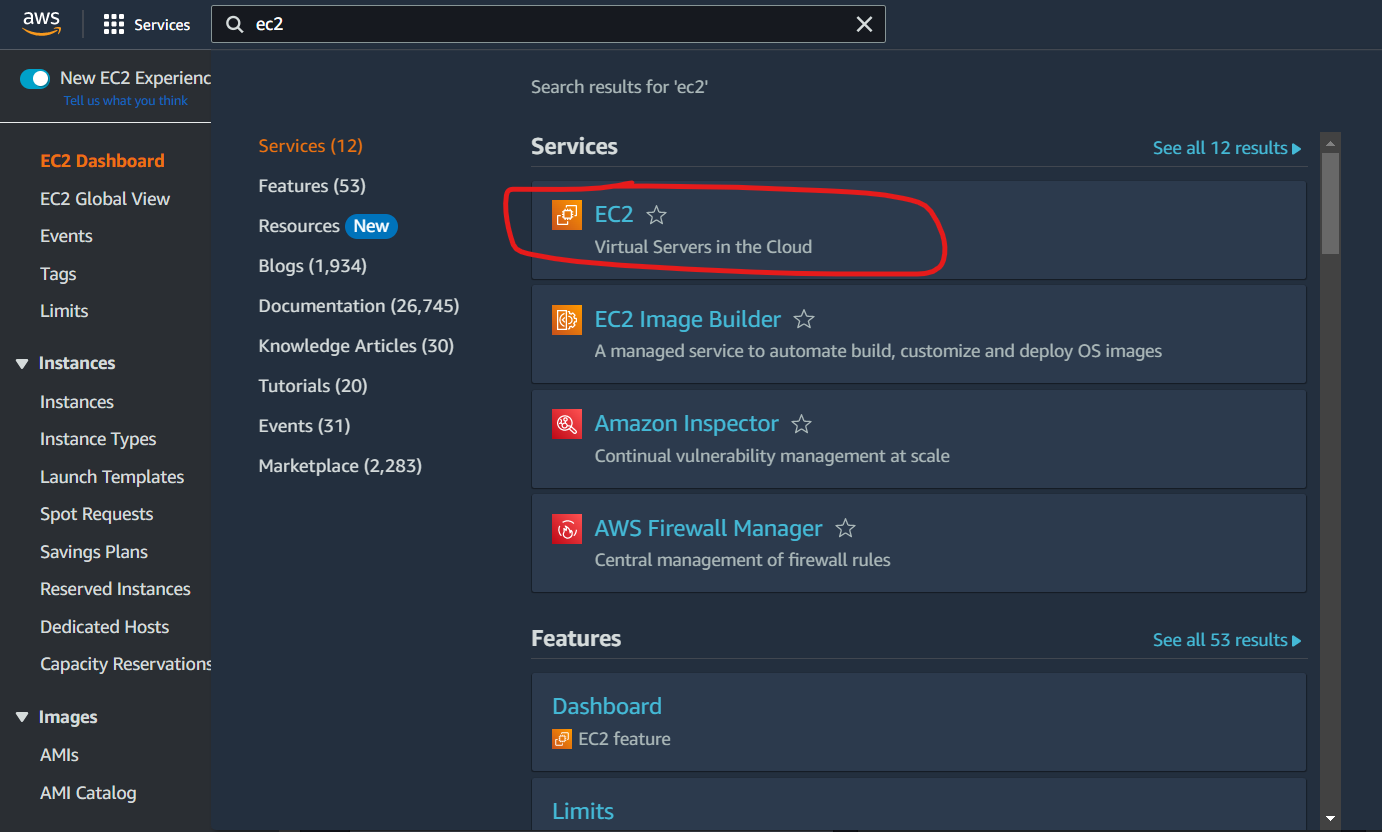
ls -l backup.txt



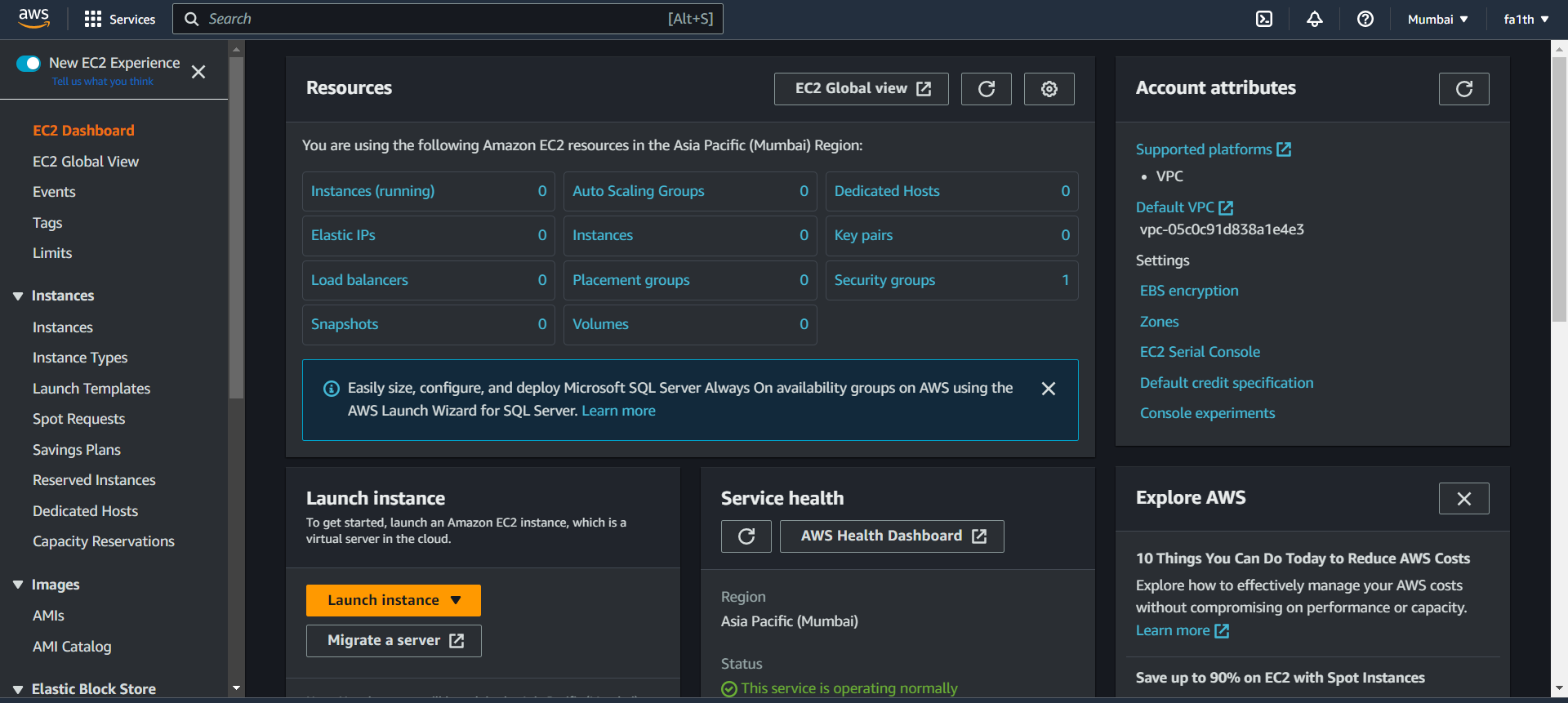
**Cloud**

1. Create a Linux EC2

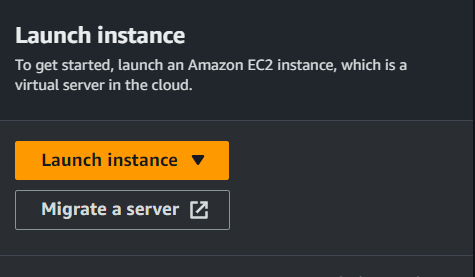
Search for ec2 instance in services



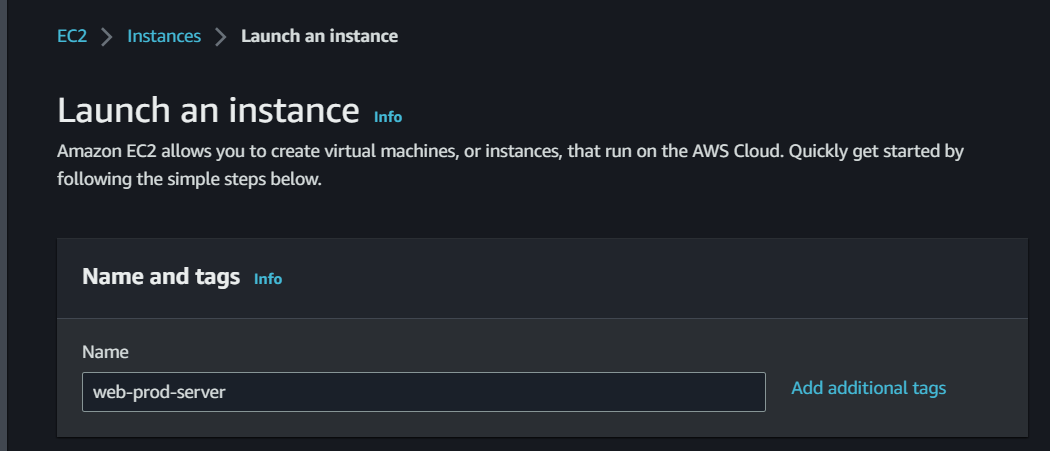
EC2 Dashboard



Launch Instance

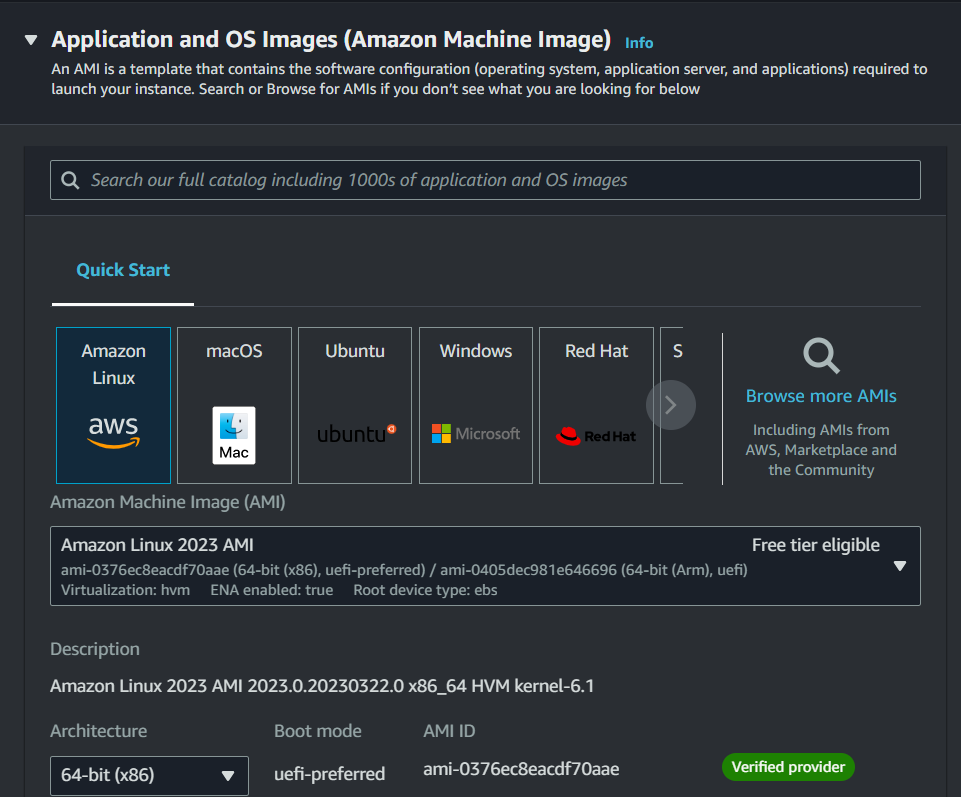


Suitable name for the instance



Selecting the OS and architecture.

Here we are selecting **Amazon Linux 64 bit architecture**



Instance type as per the requirement

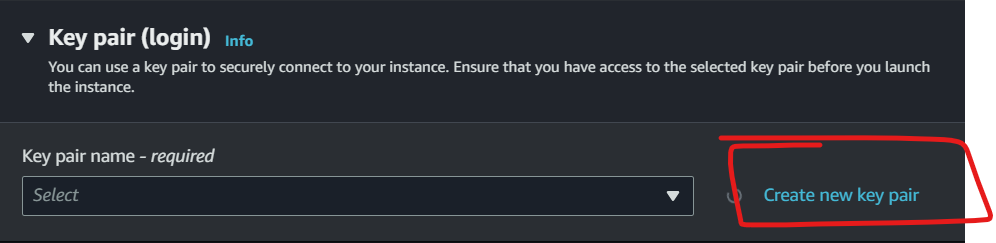
Here t2.micro

**1 vCPU and GiB Memory**

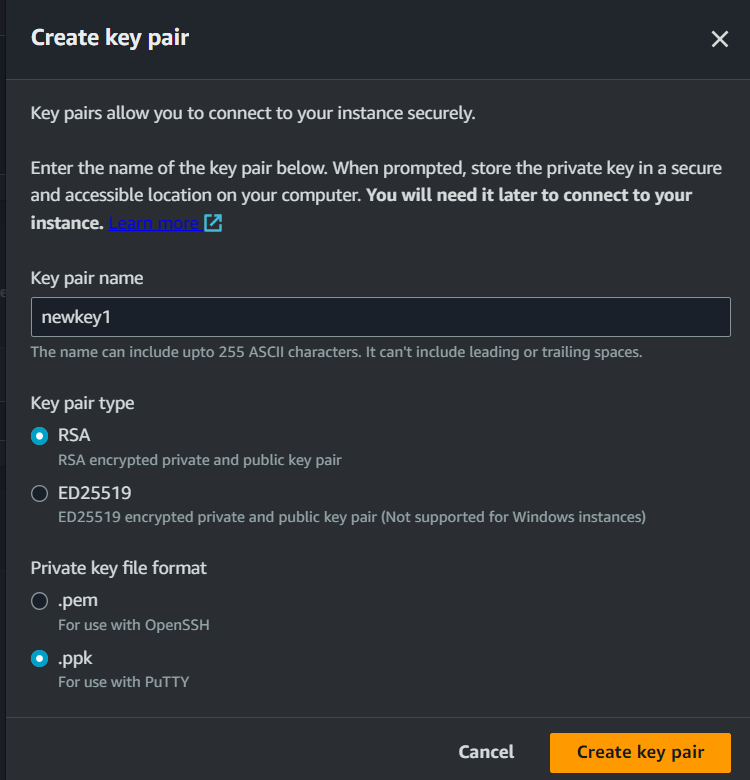


Now we need key pair for authentication

So we create new key

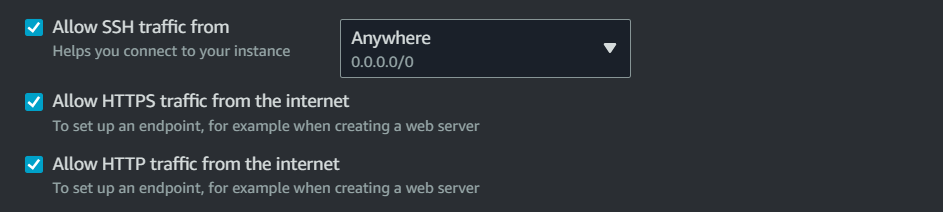


New key name and type **.ppk** since we have to connect using putty.

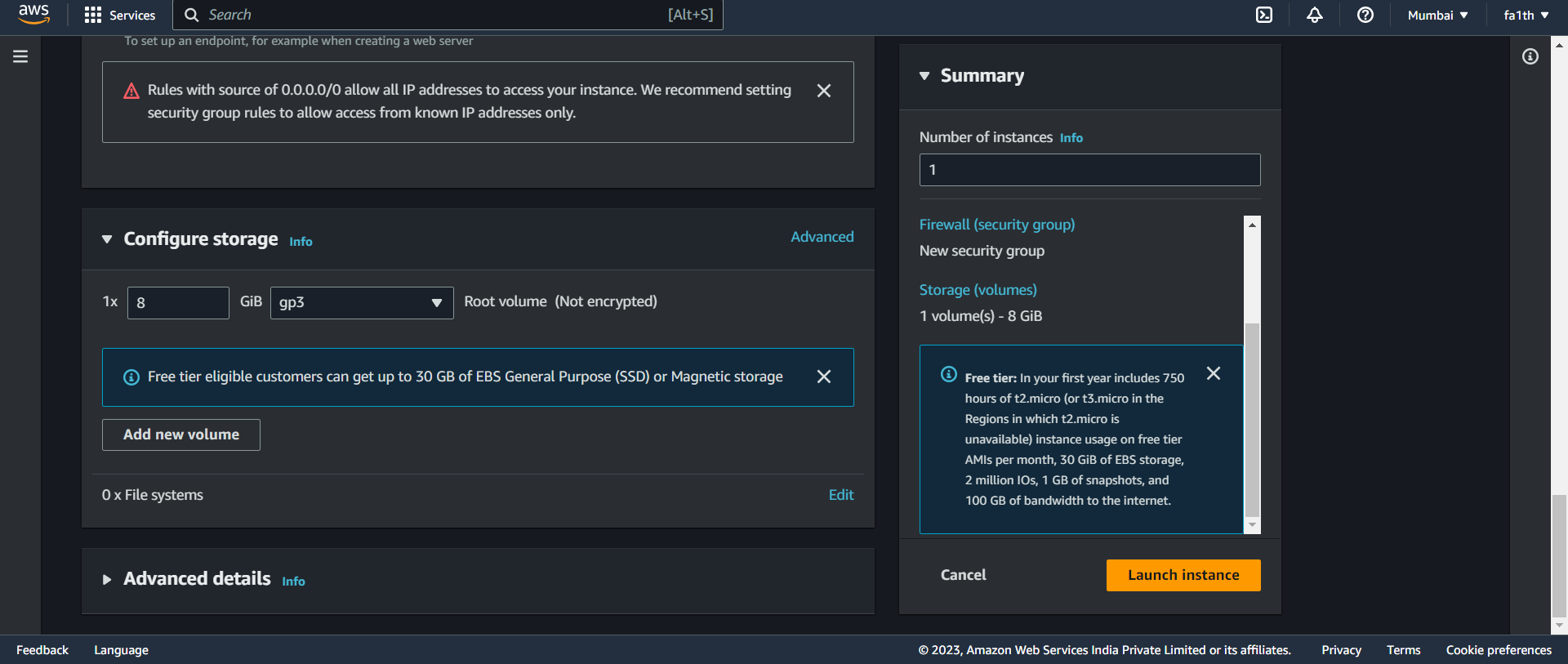


Key will be downloaded.

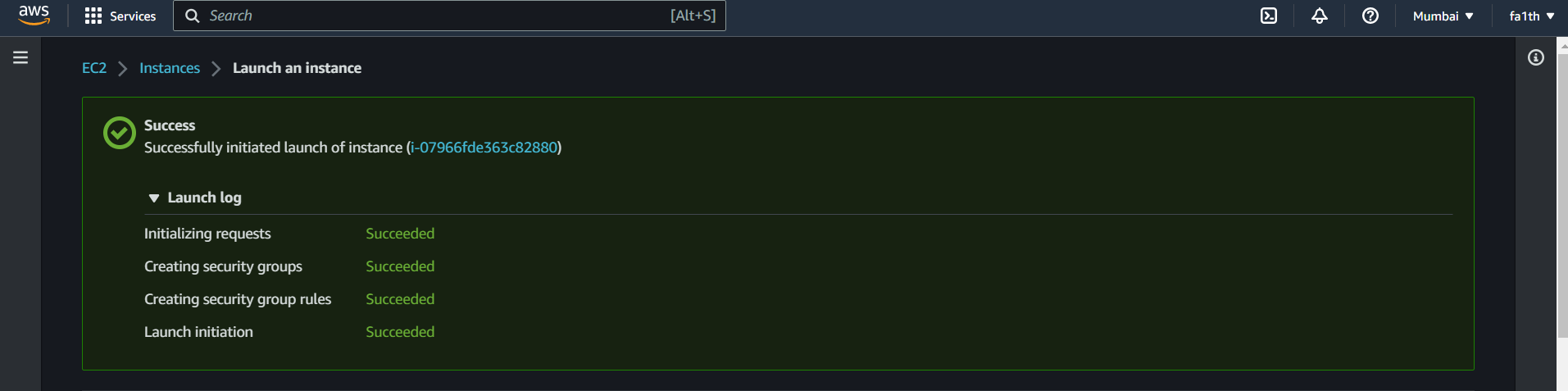
Now we need to allow traffic from anywhere



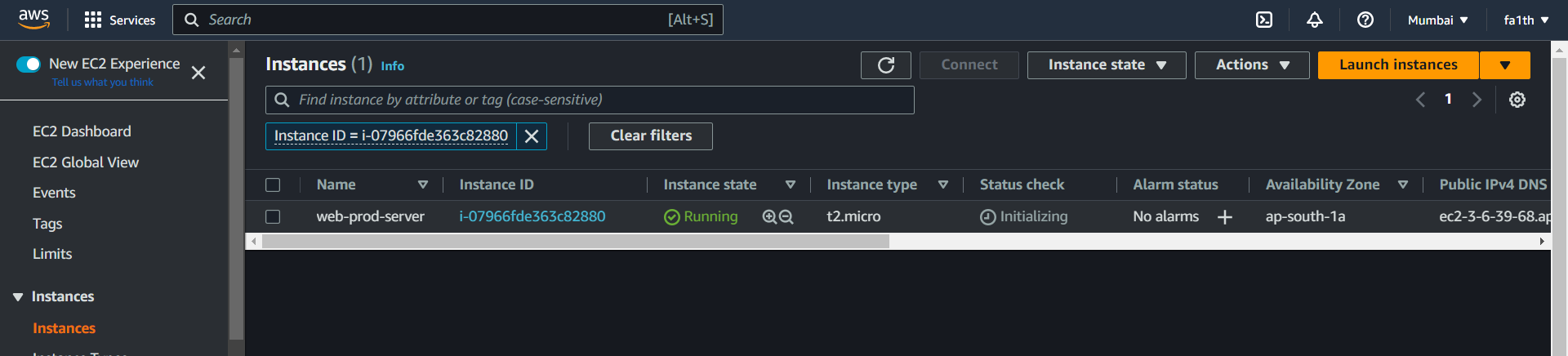
Now Launch instance

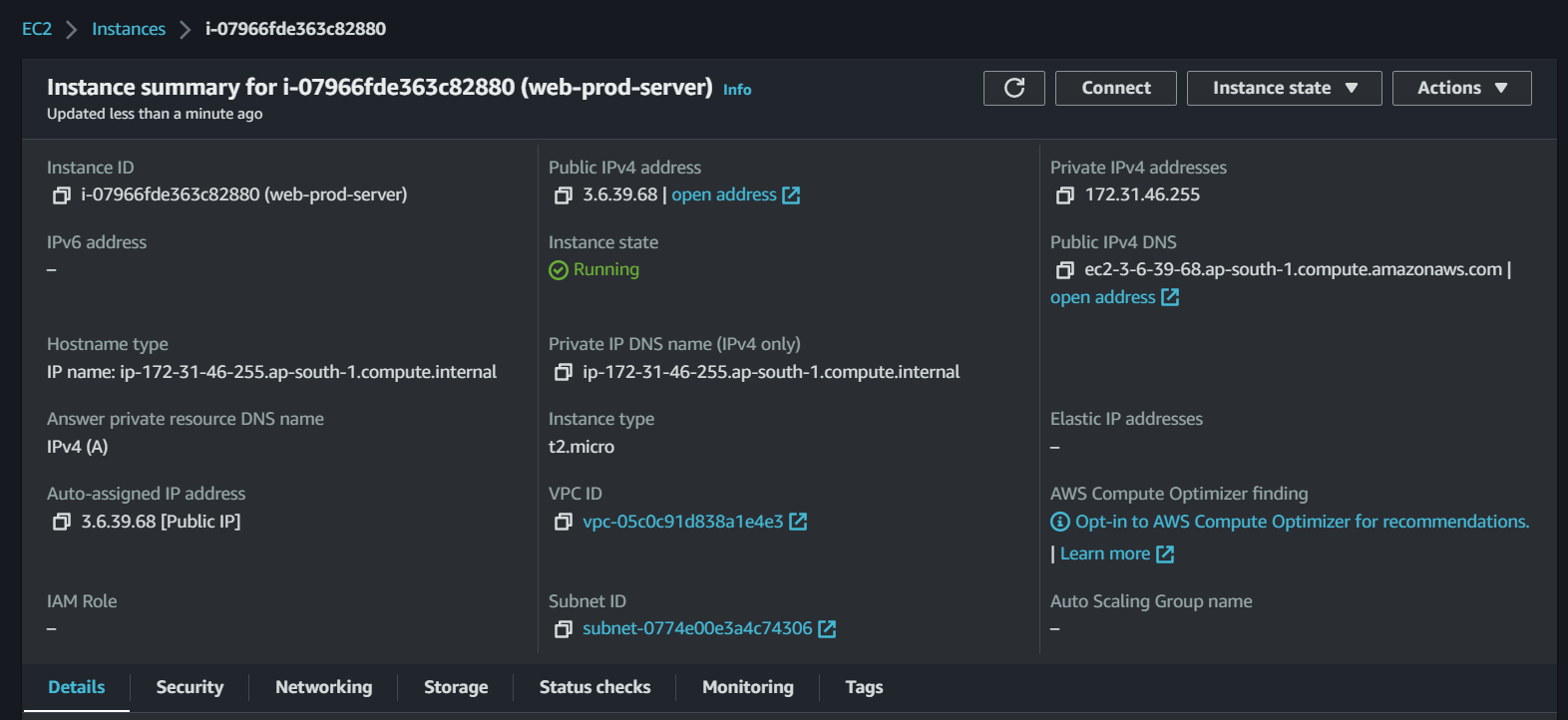


Instance successfully created



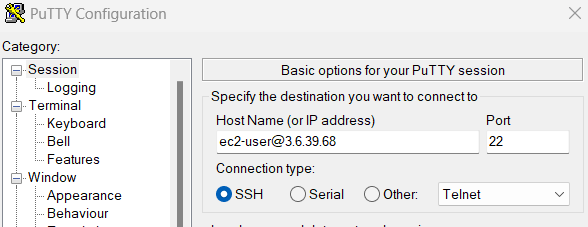
Now we click on instance id and it will redirect to console



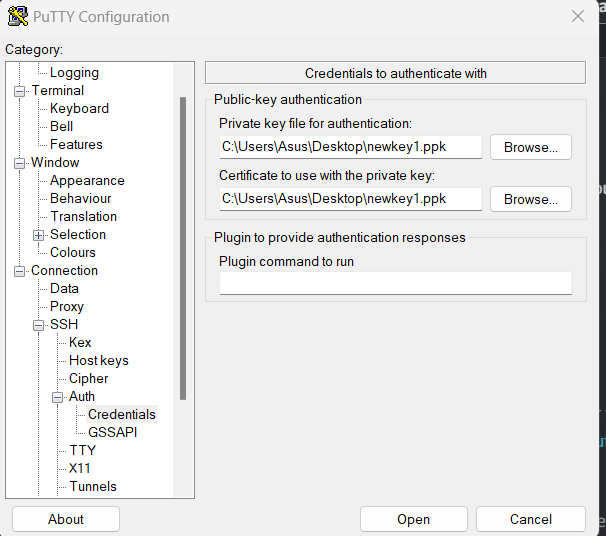


Now we open Putty on our local machine

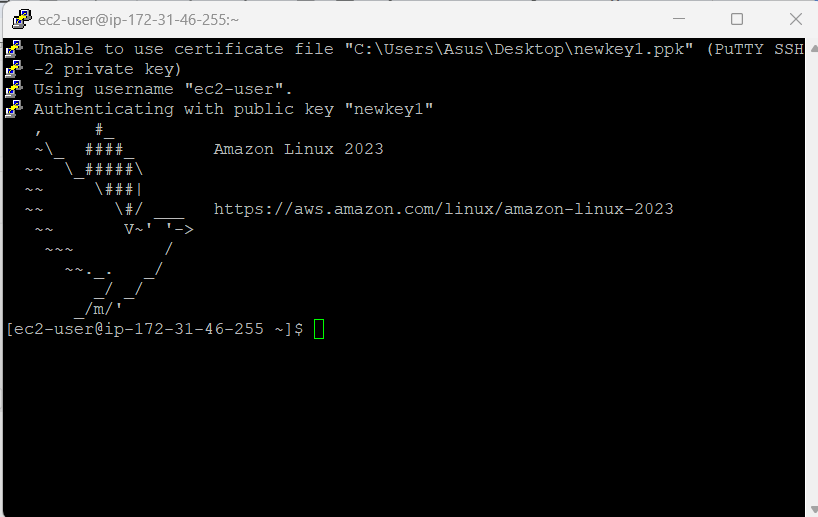
We enter the user name and public ip of the instance and select ssh



SSH > Auth > Credentials

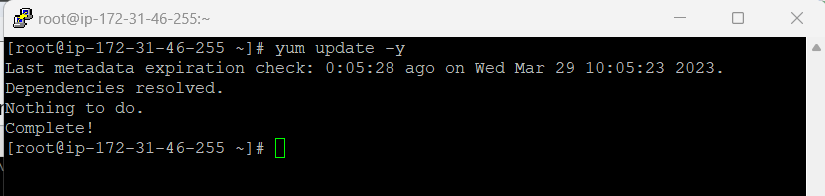


Click open and we will be inside our instance (Amazon Linux)



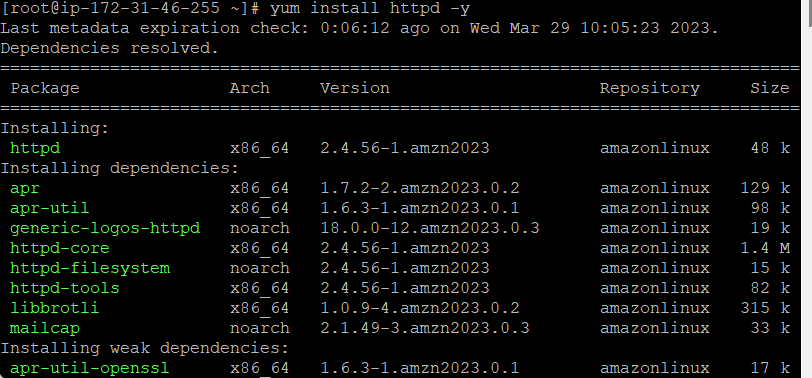
Check for any updates

yum update -y



Now we need the apache http server to host our webpage

yum install httpd -y



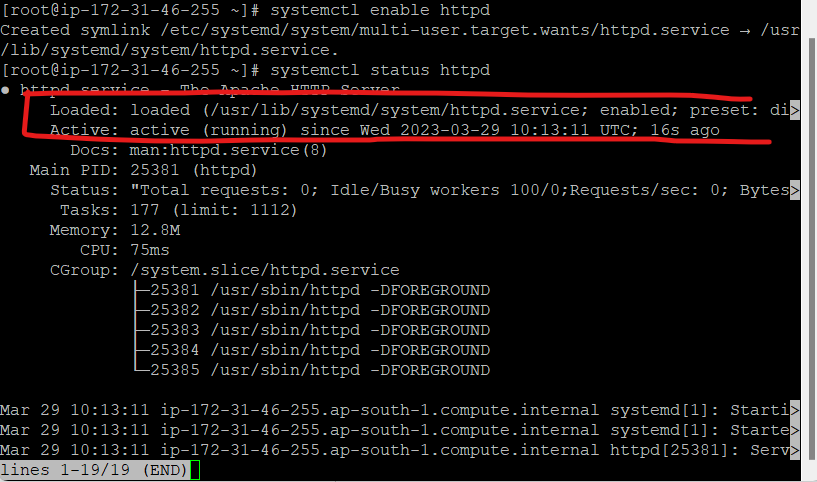
After installing the service, we need to start and enable the httpd service

systemctl start httpd

systemctl enable httpd

systemctl status httpd

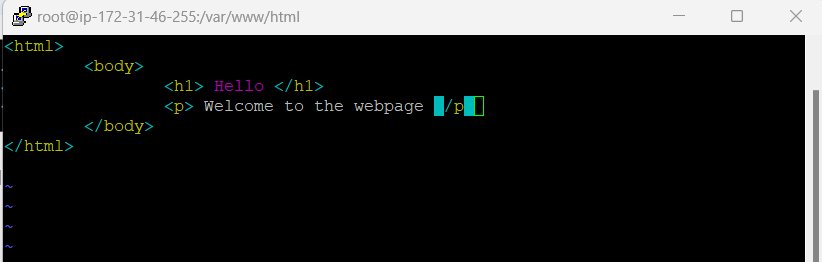
Thus the status of httpd is active and running



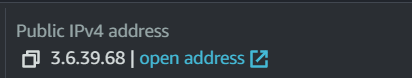
Now we need to create a html page. For that we need to go into /var/www/html

cd /var/www/html

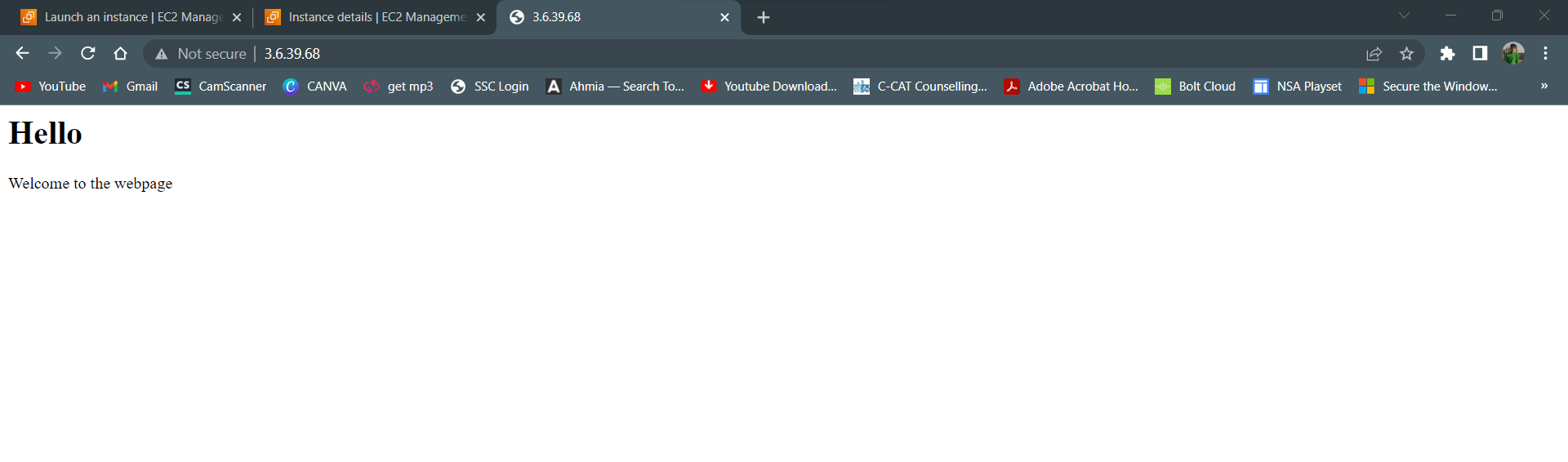
vim index.html



Copy the public IP

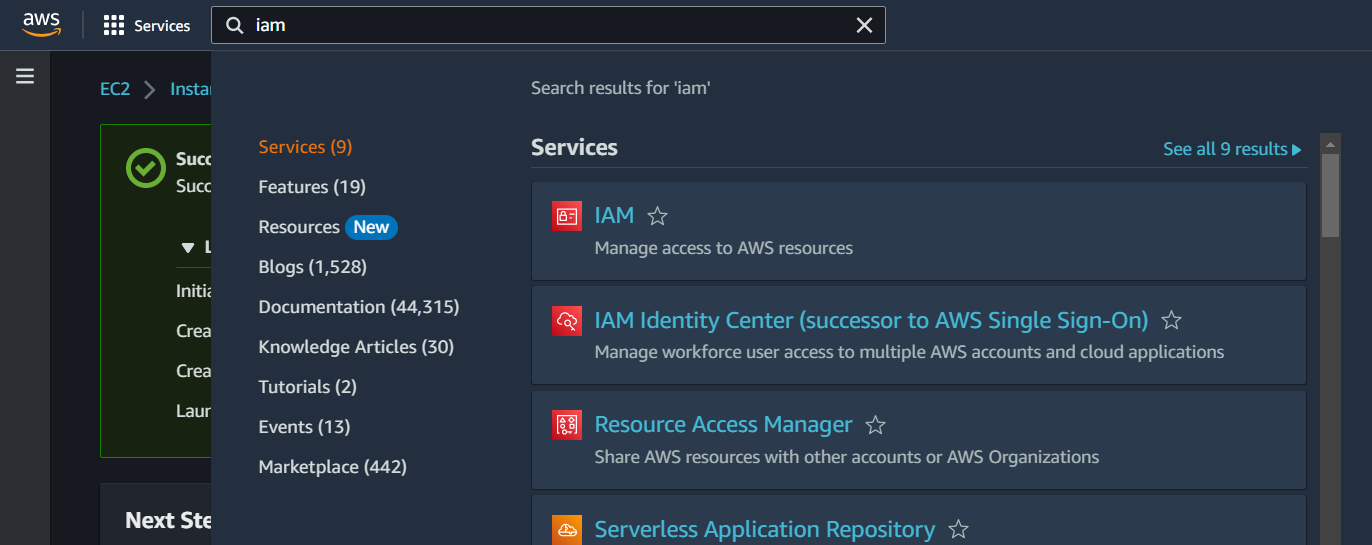


Paste in another tab and it will show the contents of html file

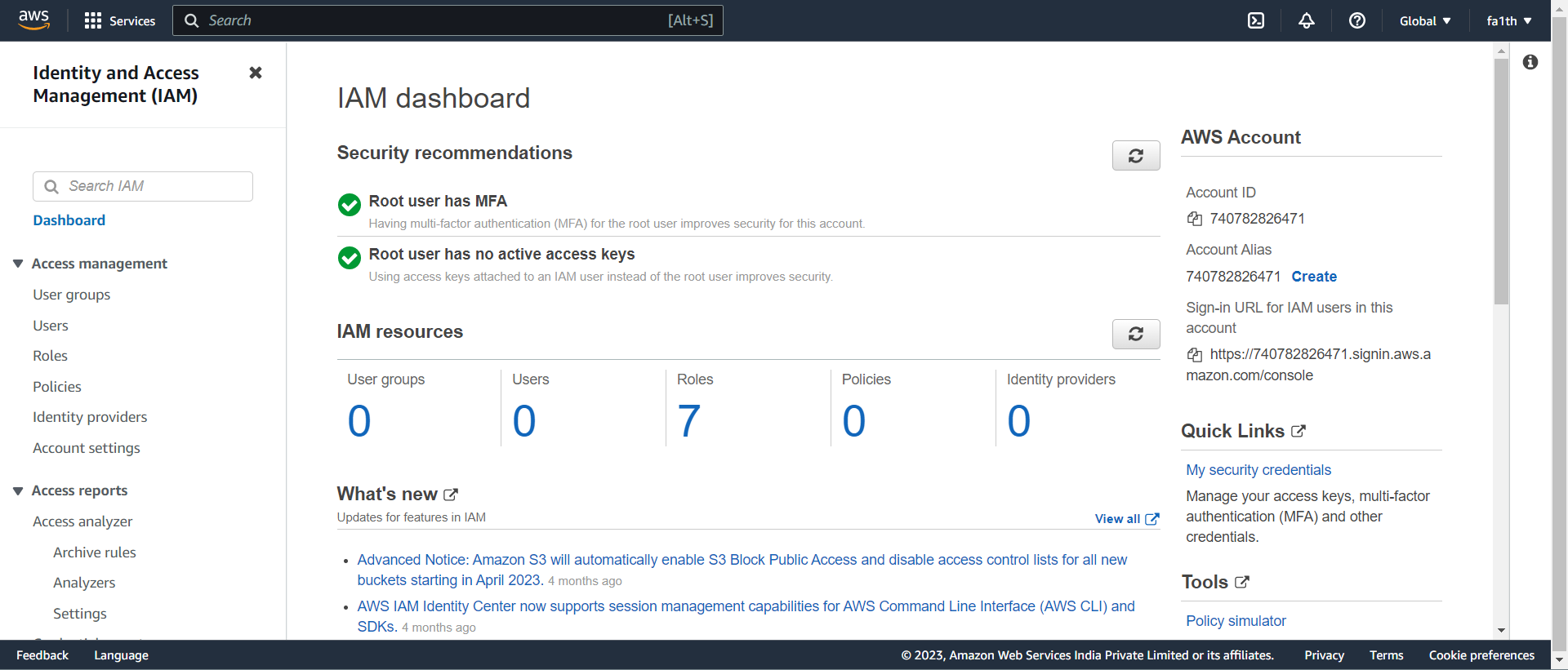


1. **IAM**

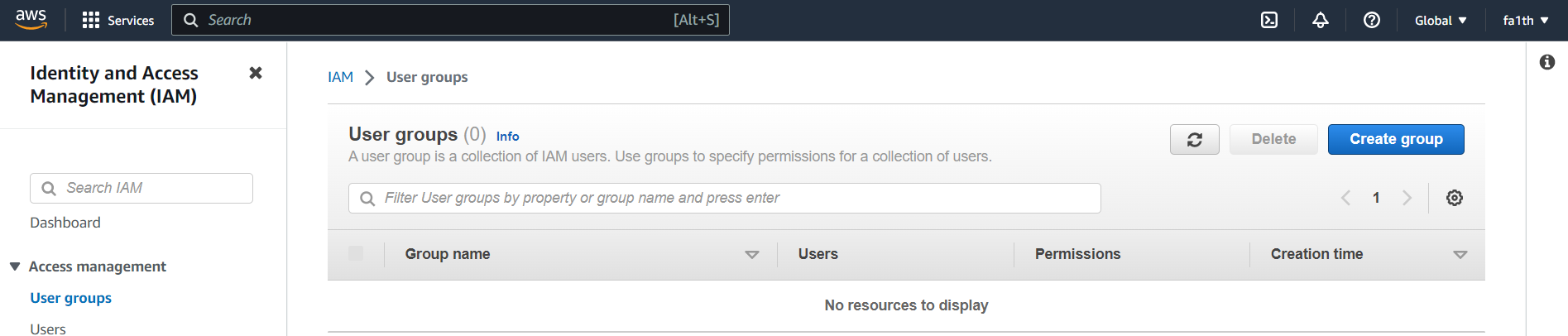
Search IAM in services

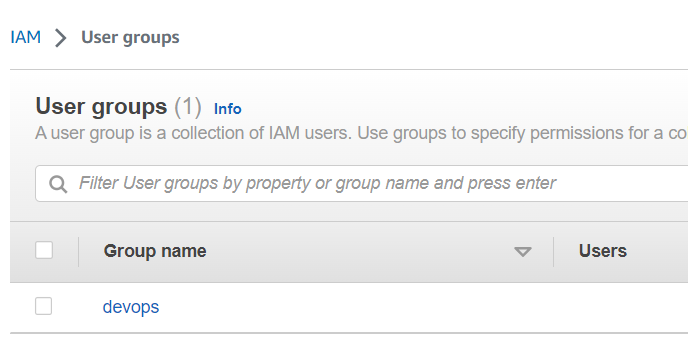


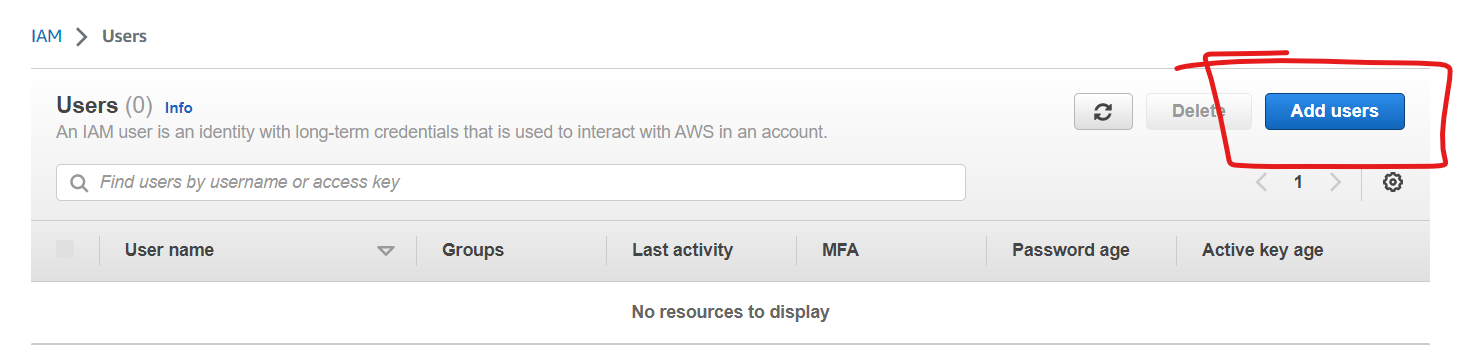
IAM Dashboard will open up



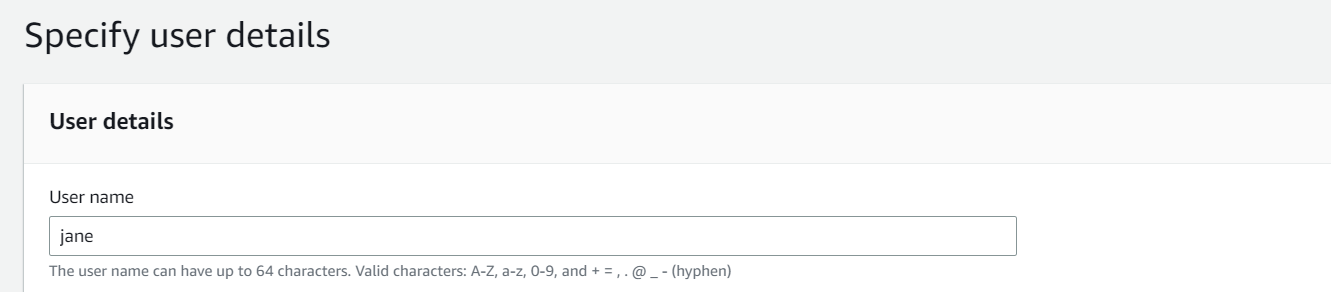
Then we click on groups and create new group - devops



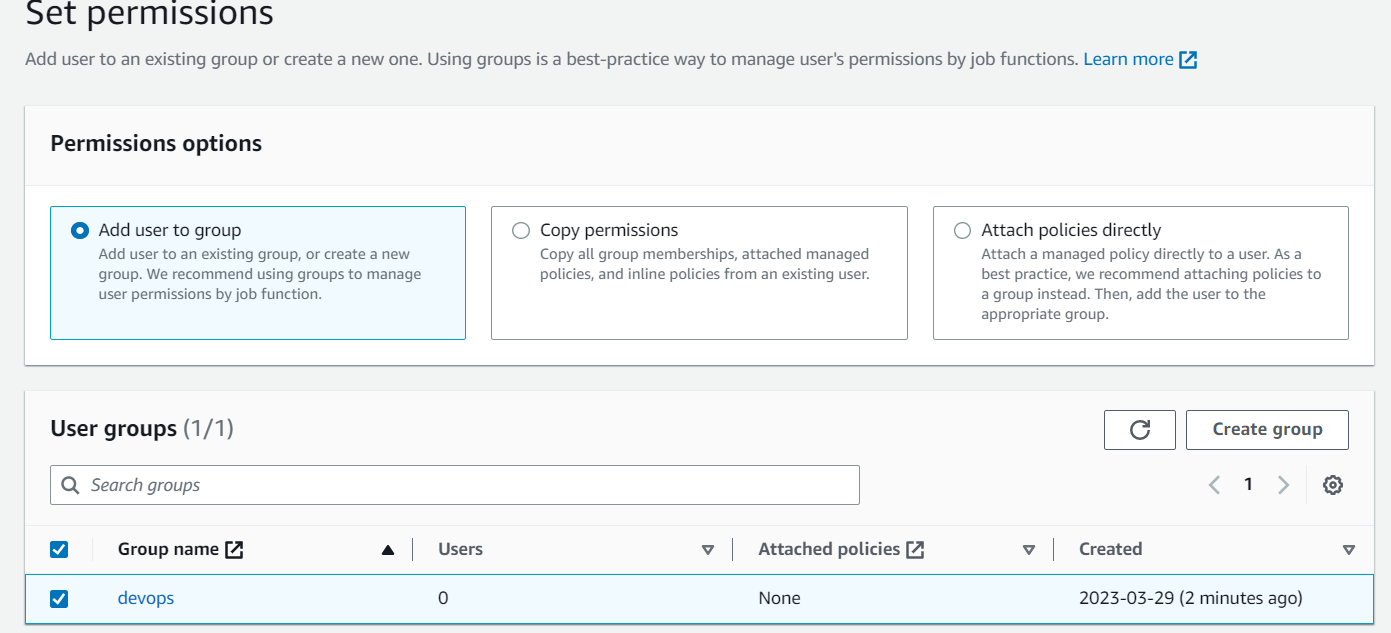


Create user 

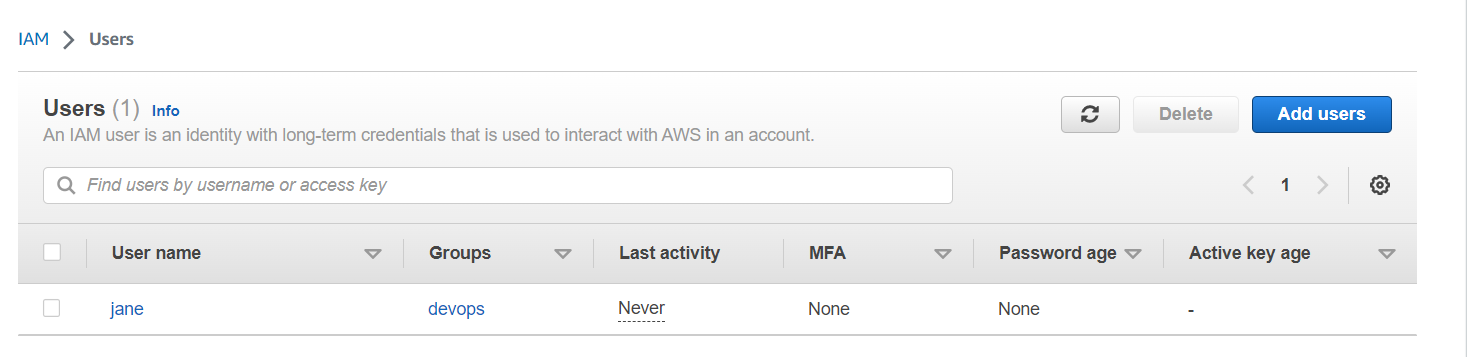
Specify the user details



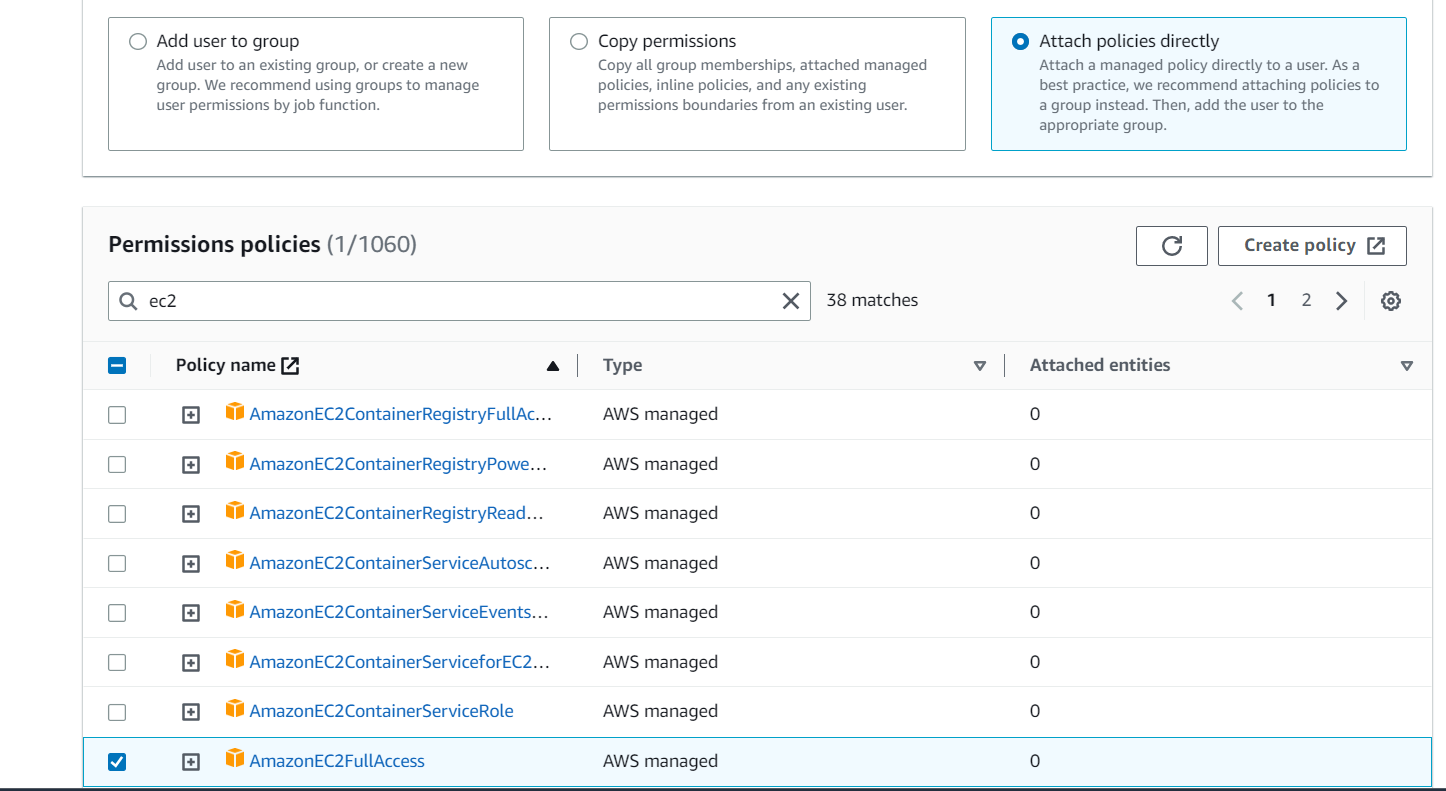
Add the user to the group as well



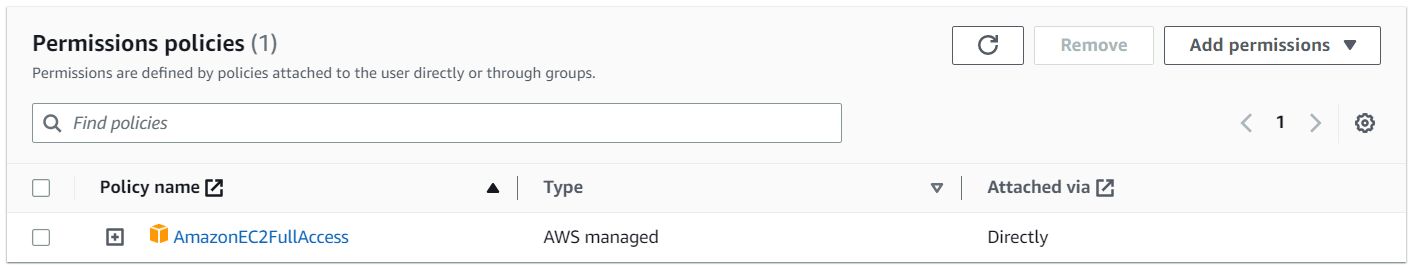
User is created and added to the group



Click on the username and we attach policies giving full access to EC2



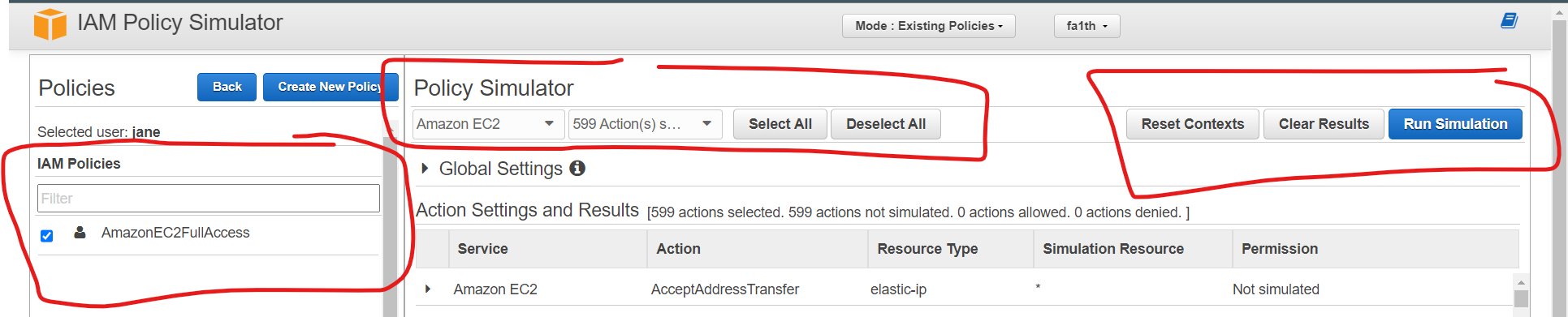
Thus, policy has been added



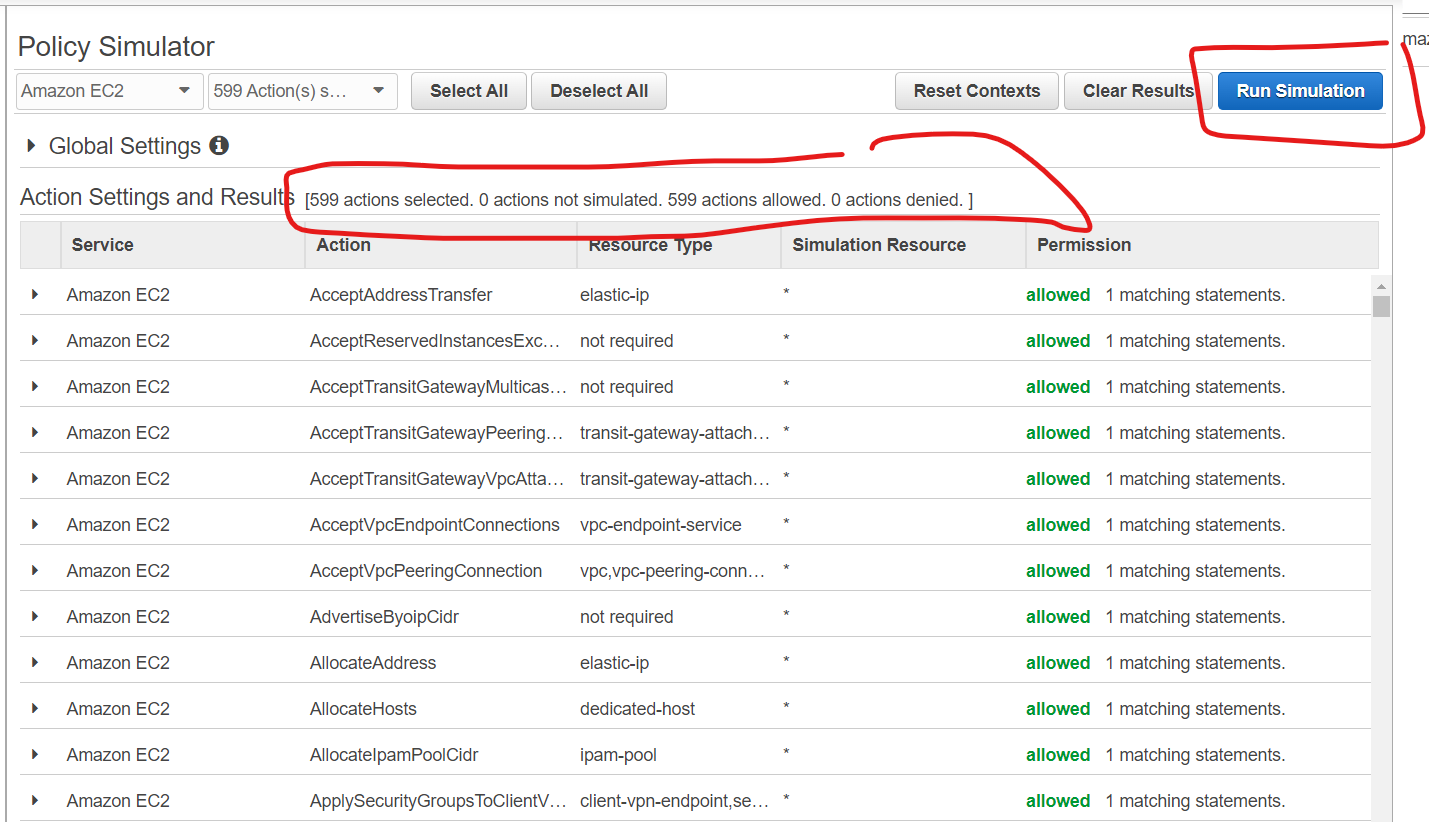
Now we go to IAM Policy Simulator

And click on the user we created - jane

Here we can see the assigned policy to the user.



Once we click simulate. All the selected actions will be simulated and allowed to the user.



Thus we have verified user permissions, all actions allowed because we have given EC2 full access.