Docker Setup

How to install Docker in Ubuntu

sudo apt update

sudo apt install apt-transport-https ca-certificates curl software-properties-common

curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo apt-key add –

sudo add-apt-repository "deb [arch=amd64] https://download.docker.com/linux/ubuntu $(lsb\_release -cs) stable"

**Verify Docker Installation**

sudo docker run hello-world

**Add Your User to the Docker Group (Optional)**: If you want to avoid using **sudo** each time you run Docker commands, you can add your user to the **docker** group:

sudo usermod -aG docker $USER

how to check docker images run in background

docker ps

how to check previously created images and running images

docker ps –a

how to run docker in interactive mode

docker run -it nginx

how to run docker in background mode

docker run -dt nginx

2) what is the difference between docker rmi and docker rm command?

Ans:

The docker rmi and docker rm commands in Docker serve different purposes:

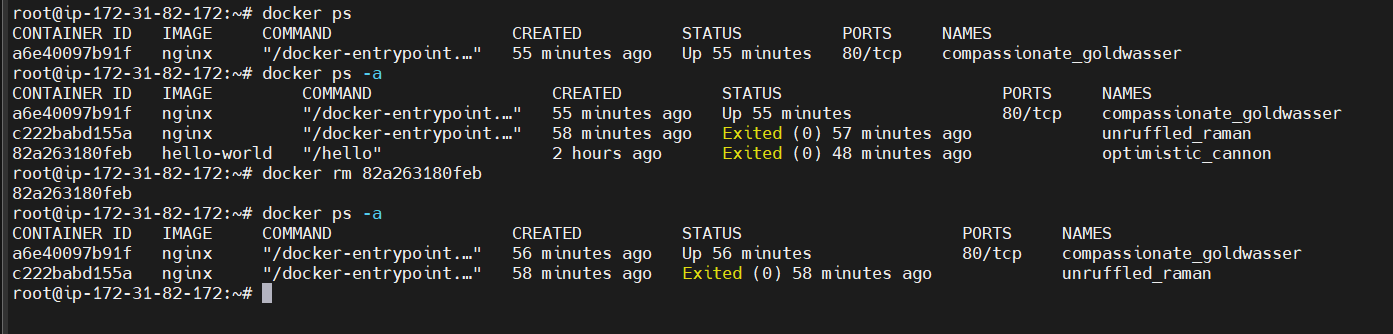
docker rmi: This command is used to remove Docker images from your local system

$ docker rmi image\_name\_or\_id

docker rm: This command is used to remove Docker containers from your system. It stands for "remove container".

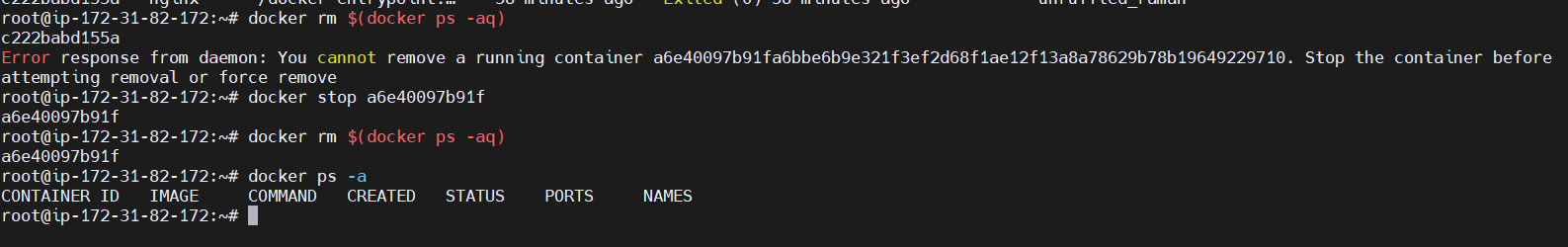
docker rm container\_name\_or\_id

O/p



3)How to remove all the docker container which is not running

docker rm $(docker ps -aq)

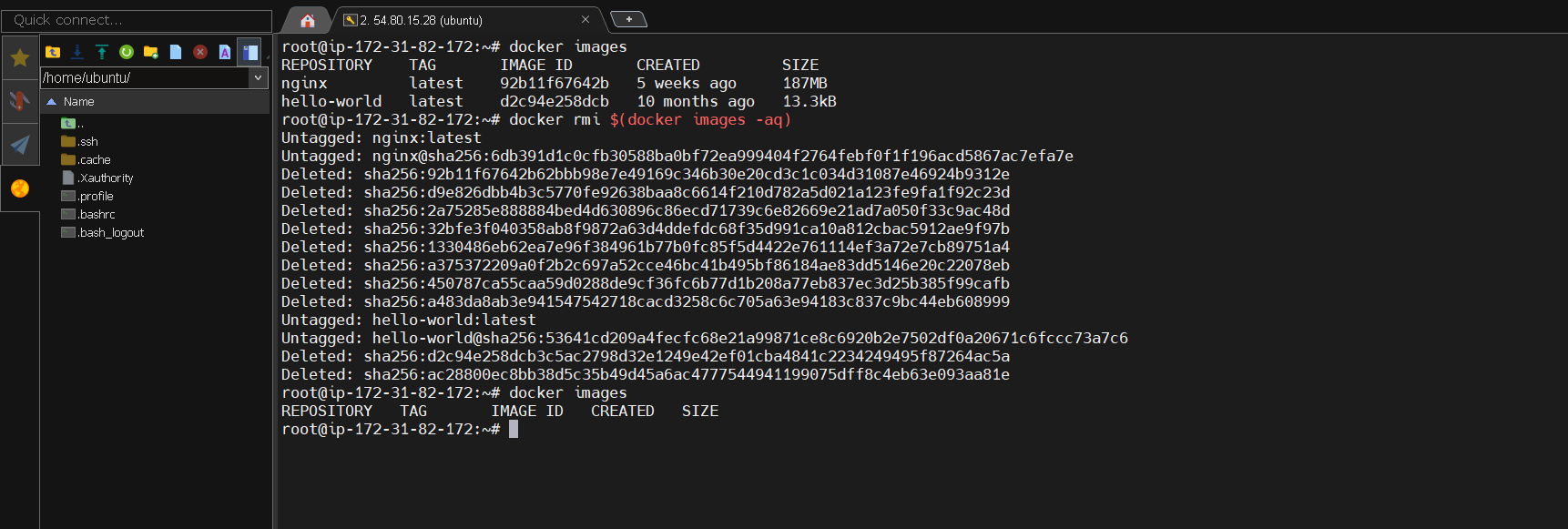


But still there is docker images

4) how to remove all docker images

Docker rmi $(docker images -aq)

o/p



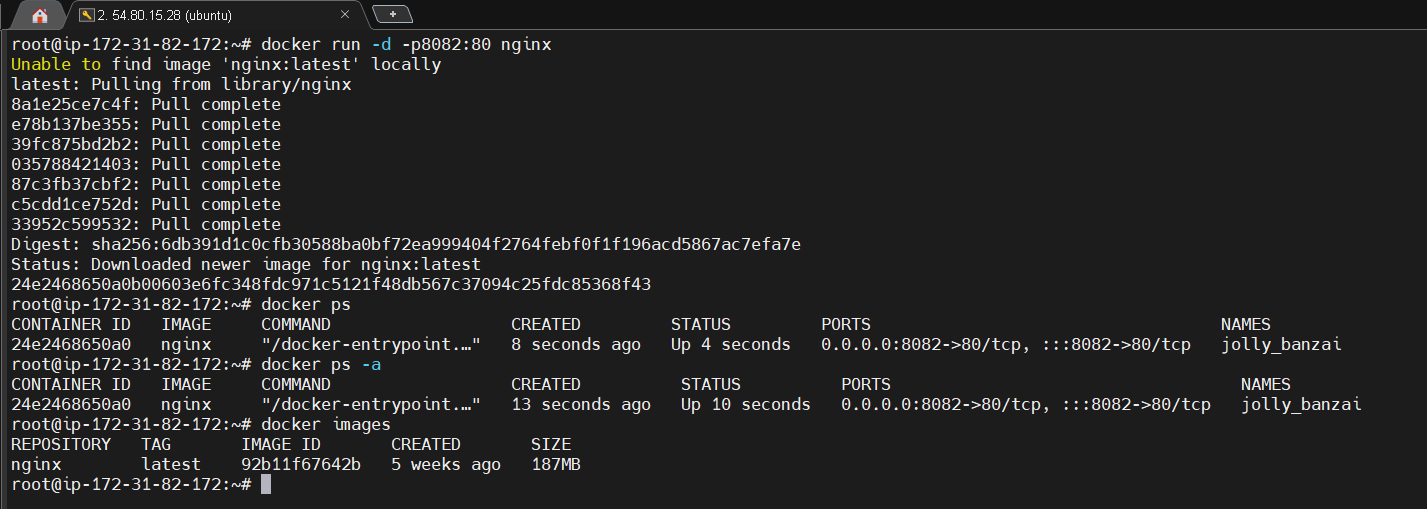
How to do port mapping in Docker

There are two ways of port mapping

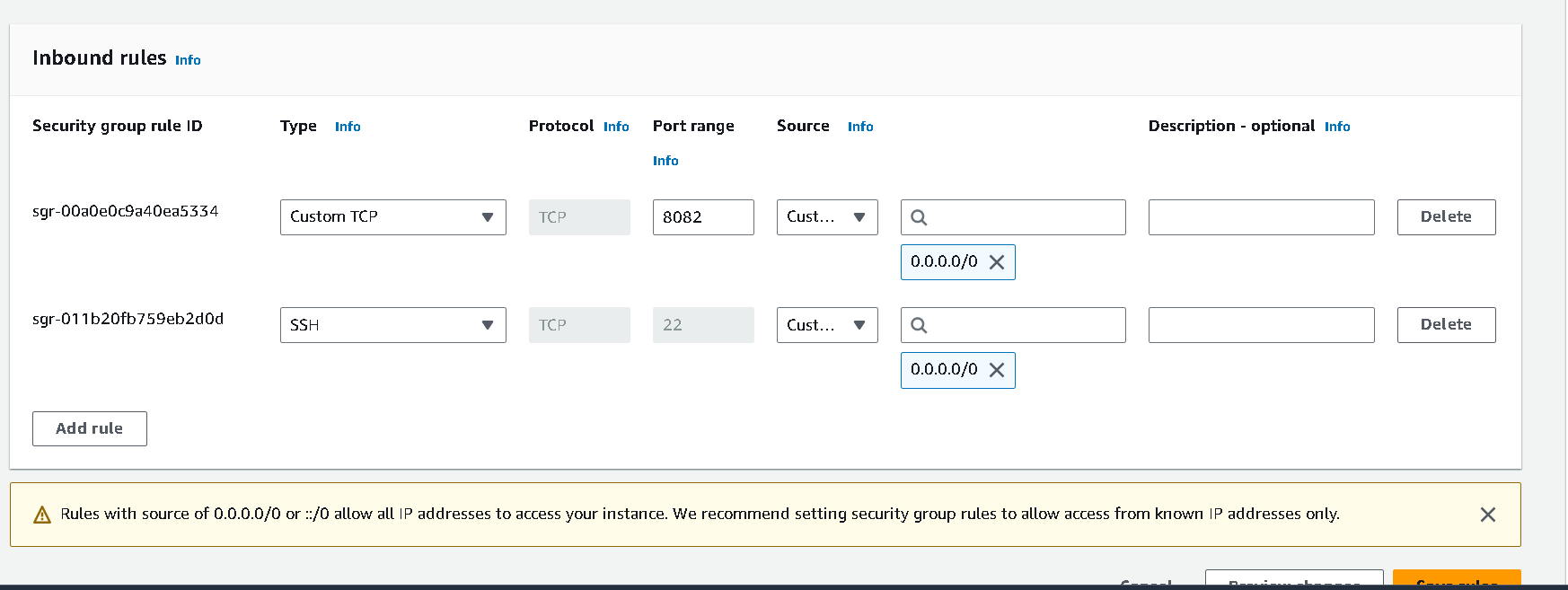
1)Custom Port Mapping

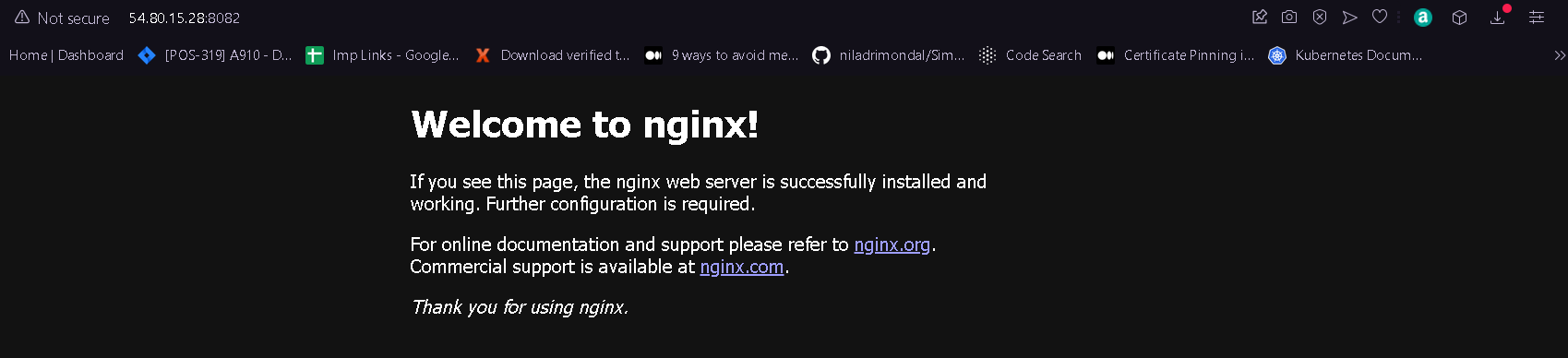
docker run -d -p8082:80 nginx

o/p



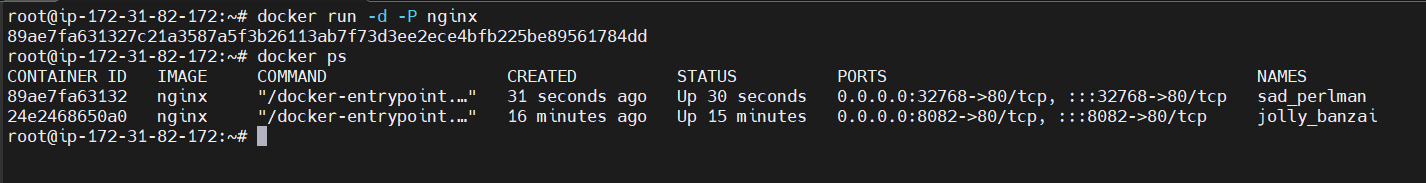
To check nginx is working opem ip and its port in web browser but before that add security group



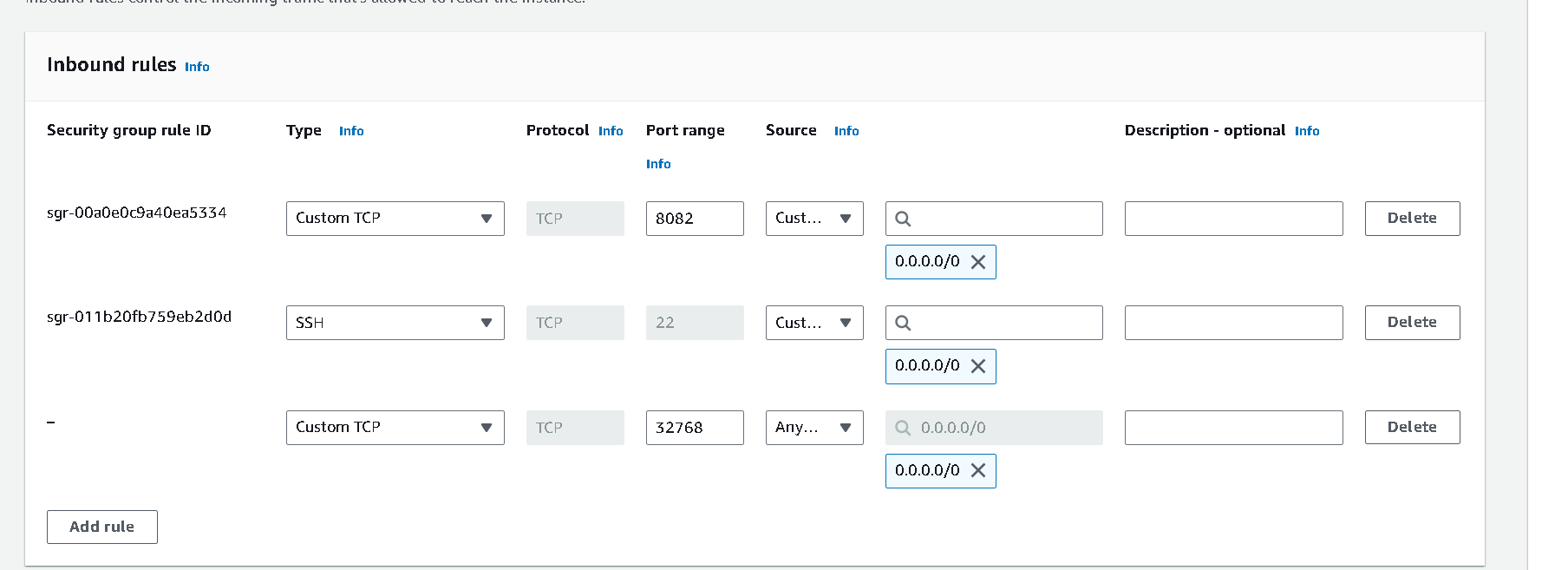


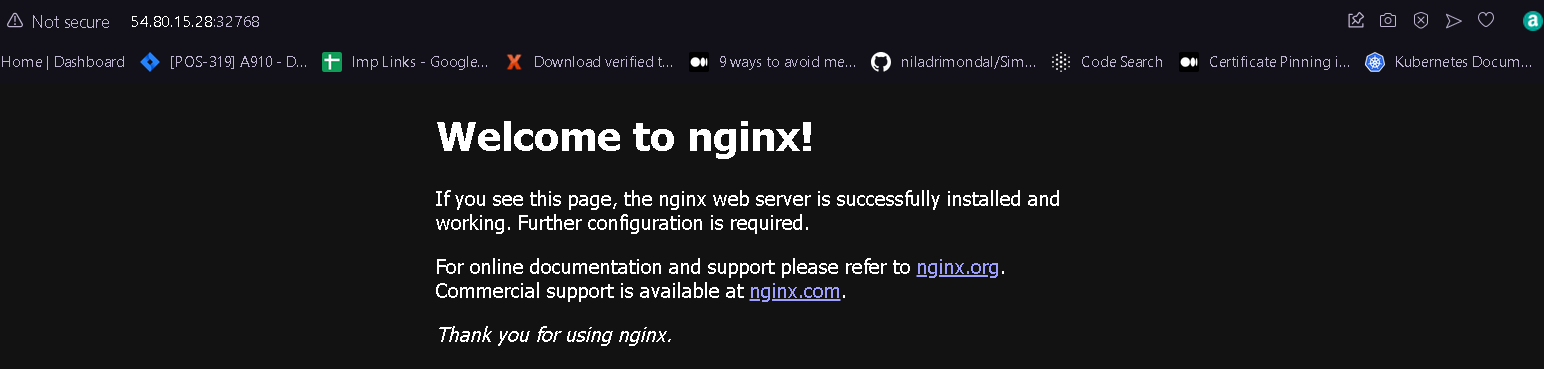
2) Random port Mapping

docker run -d -P nginx



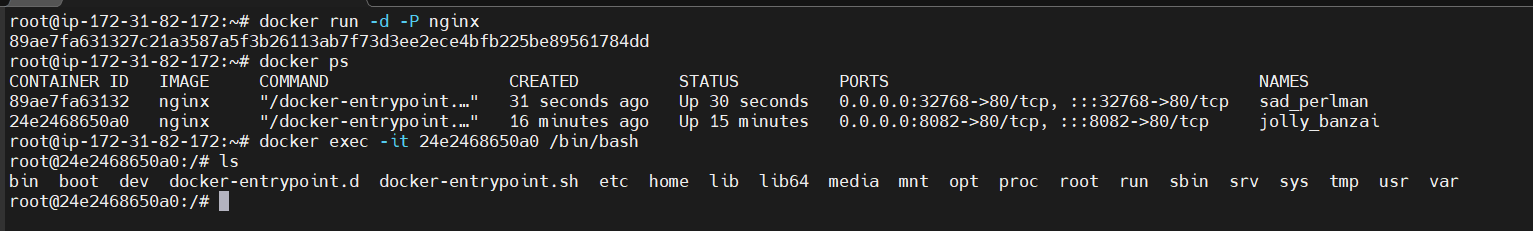
Add this random port 32768 insecurity group





How to get in nginx ? and how to host own html file on nginx

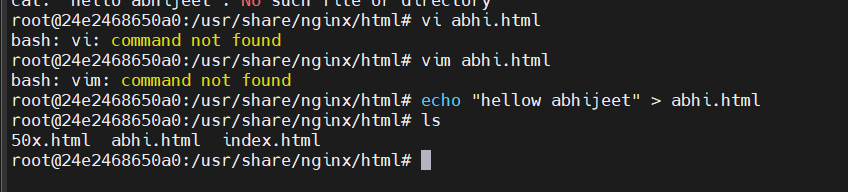
docker exec -it 24e2468650a0 /bin/bash



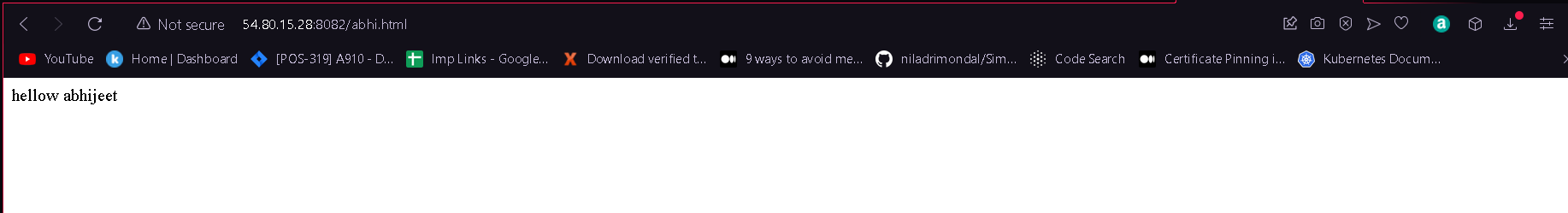
Navigate to usr/share/nginx/html folder

cd usr/share/nginx/html

vim and vi is not working so using echo create abhi.html file



To test this ty ip port number and file name on browser to see is file hosted or not

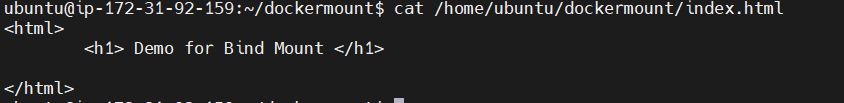


**How to create Docker with port and volume**

First make directory dockermount then navigate to that directory

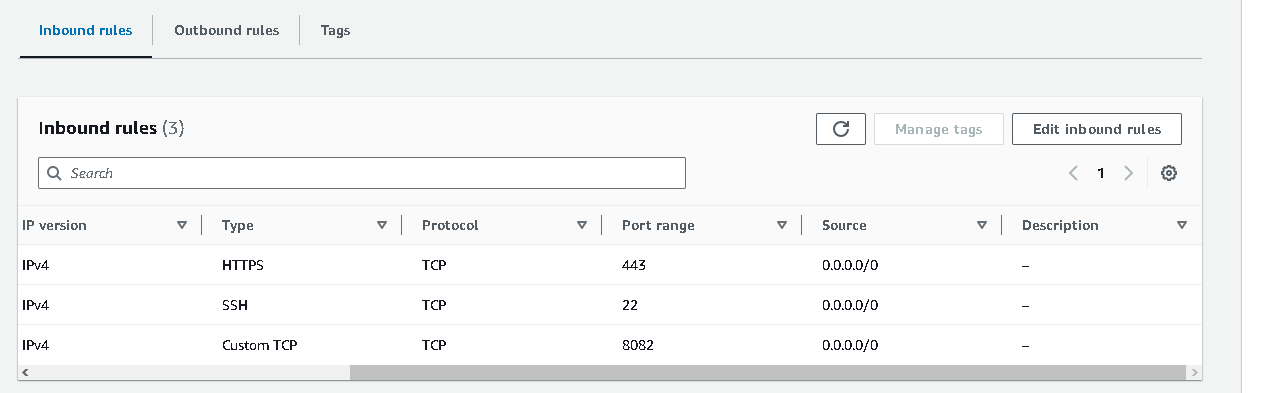
cd /home/ubuntu/dockermount

Inside dockermount directory create index.html page

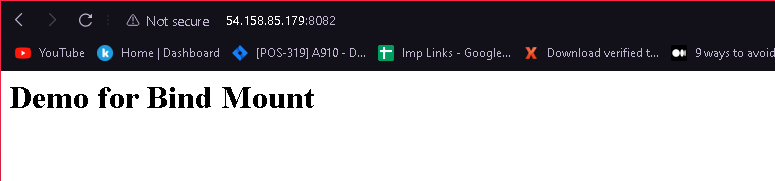


Basically what we did created a container in detach mode with port 8082:80 and give local file system path then run nginx.

To see output on browser first add security groups



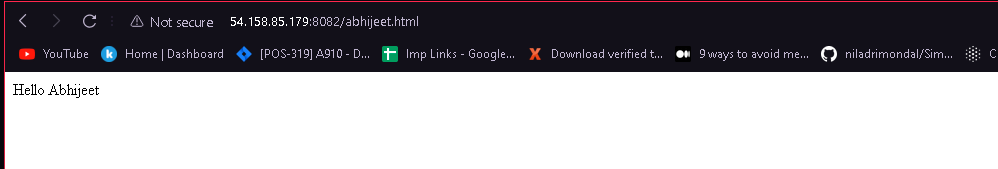
O/P



Now I have created 1 more html file in dockermount folder abhijeet.html

And saved in that folder now I am also able to get that file on browser like this

o/p



Now there is one drawback any one can easily edit html file or create new html file and host this so

to prevent this we have to restrict to normal person and on devops team can do these

**docker volume commands**

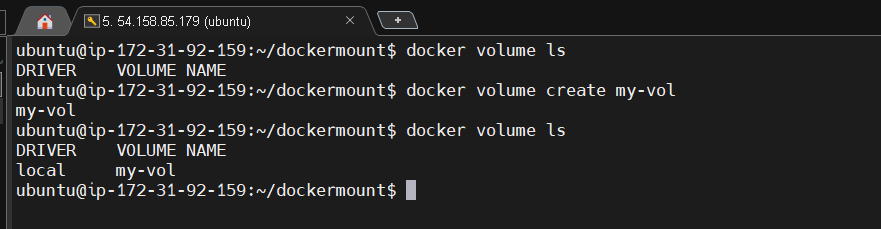
check docker volumes

docker volume ls

create volume

docker volume create my-vol

O/P



So in above example we see any one can go to dockermount folder and edit the html file to prevent this we used below commands

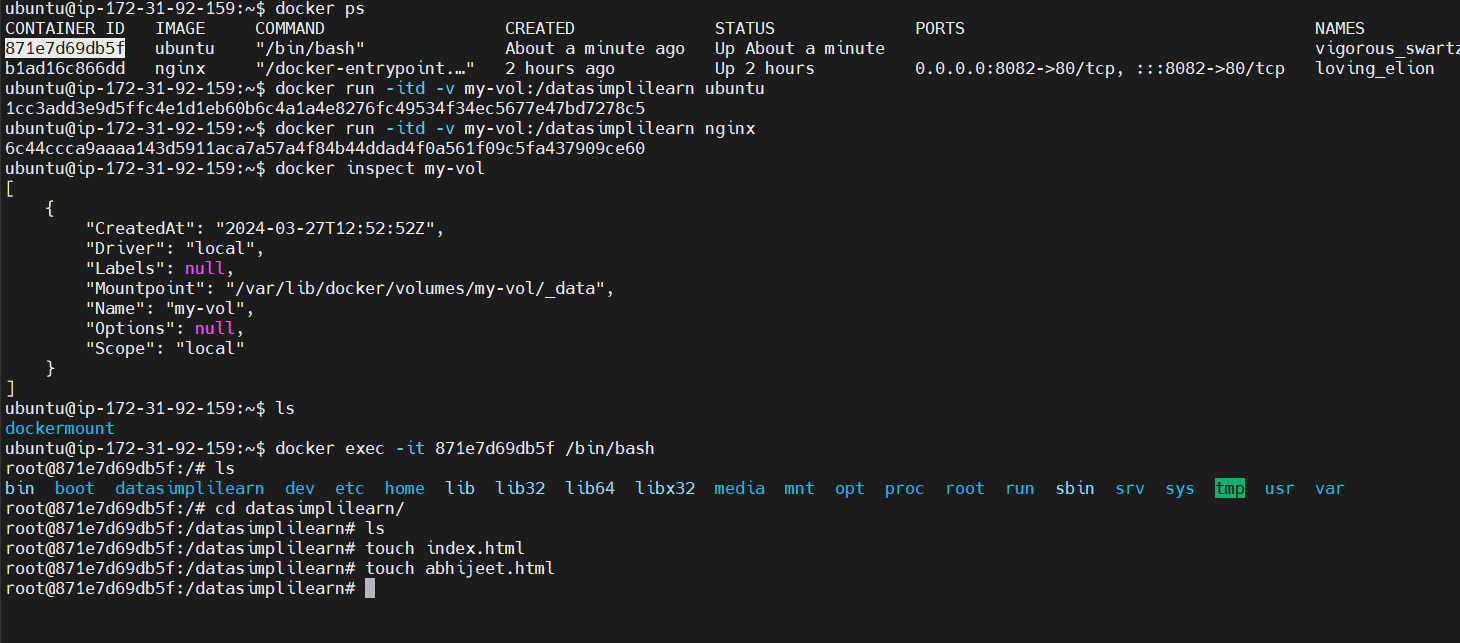
docker run –itd –v my-vol:/datasimplilearn Ubuntu

after this command only docker process is able to control that html file not other process

to check this execute nginx in interactive mode

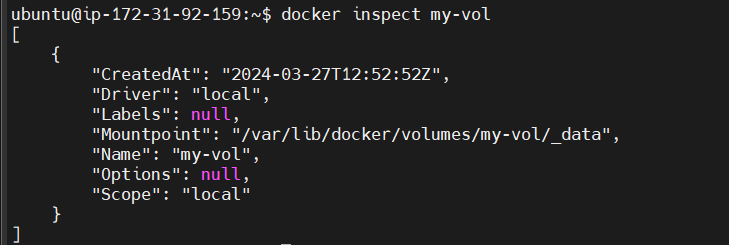
docker exec –it docker\_id /bin/bash

here we can see not able to modify or to read index.html file and abhijeet.html

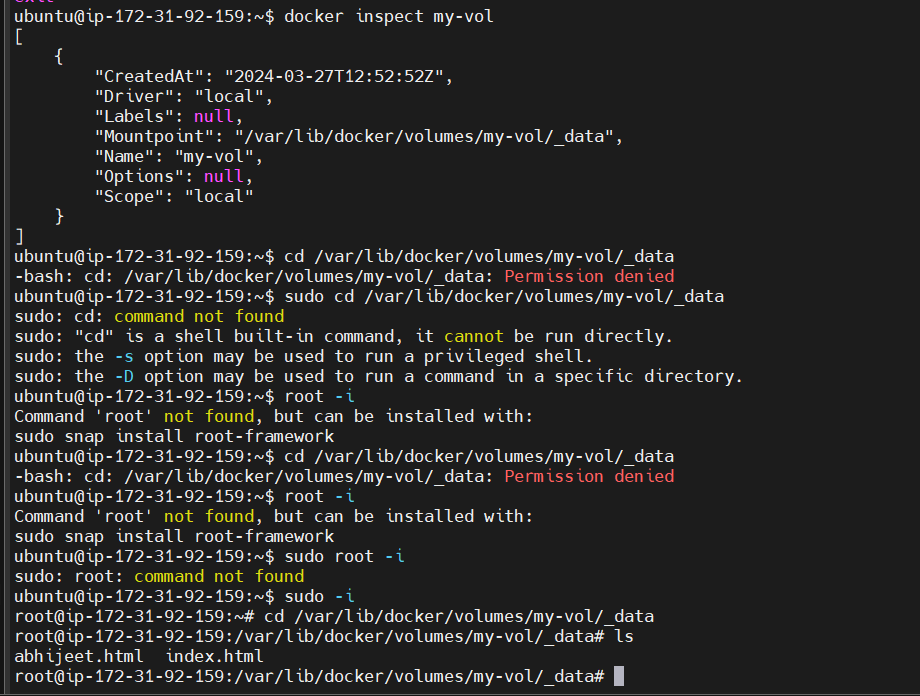


To insepct docker using following command

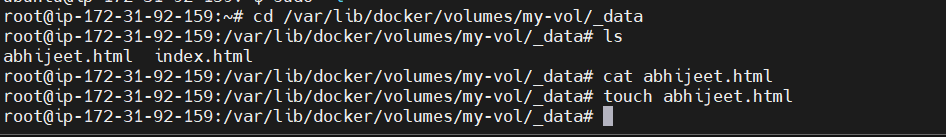
docker inspect my-vol



Here our file is securely stored



See won’t be able to read or write file using cat and touch command



So insed of bind mount always prefer with volume way to mound docker , volume way is much safer and only be controlled by docker prcess not any other process

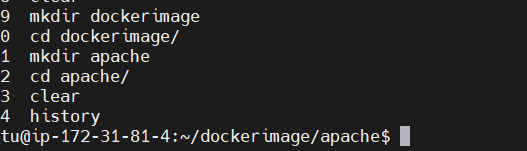
---------------------- Docker Base Image example-----------------------------------

What is Docker Images?

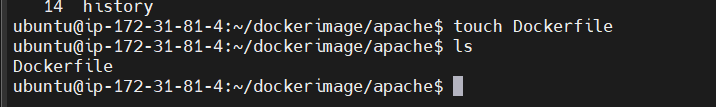
Ans: An image holds instruction that are required to run an application.

Host .html file on Apache2 server

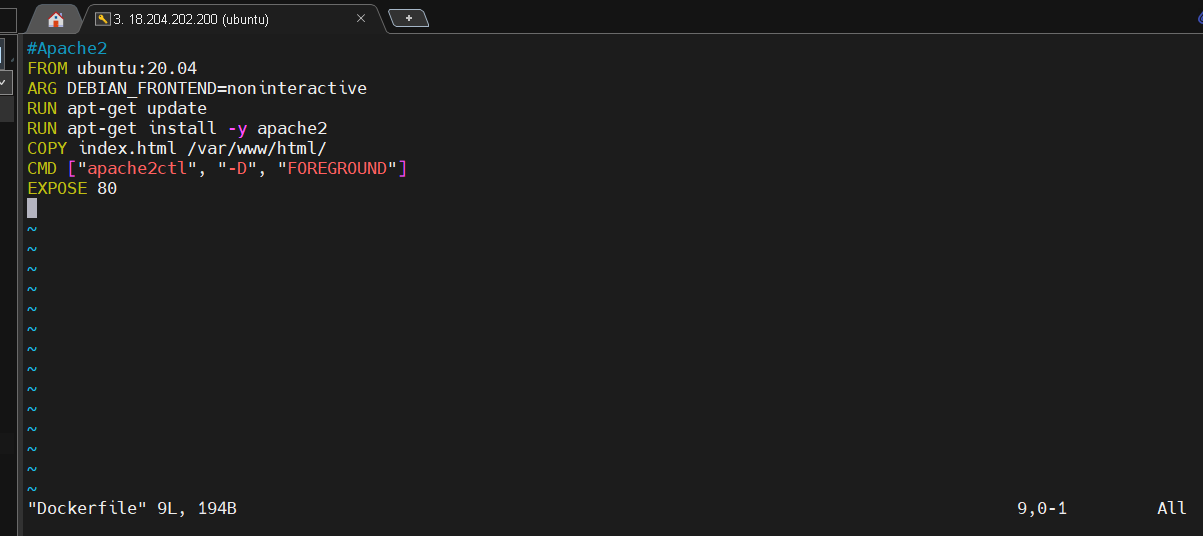
Step1 : First create directory dockerimages and inside that create another one apache directory



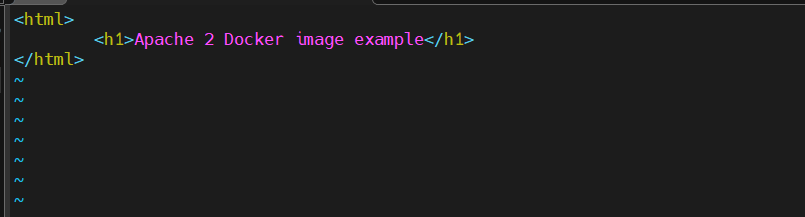
Step2: Inside apache directory create Dockerfile using touch command



Step3: write down steps and arguments for creating apache image



Now create index.html file that we need to store in apache directory ie /var/www/html



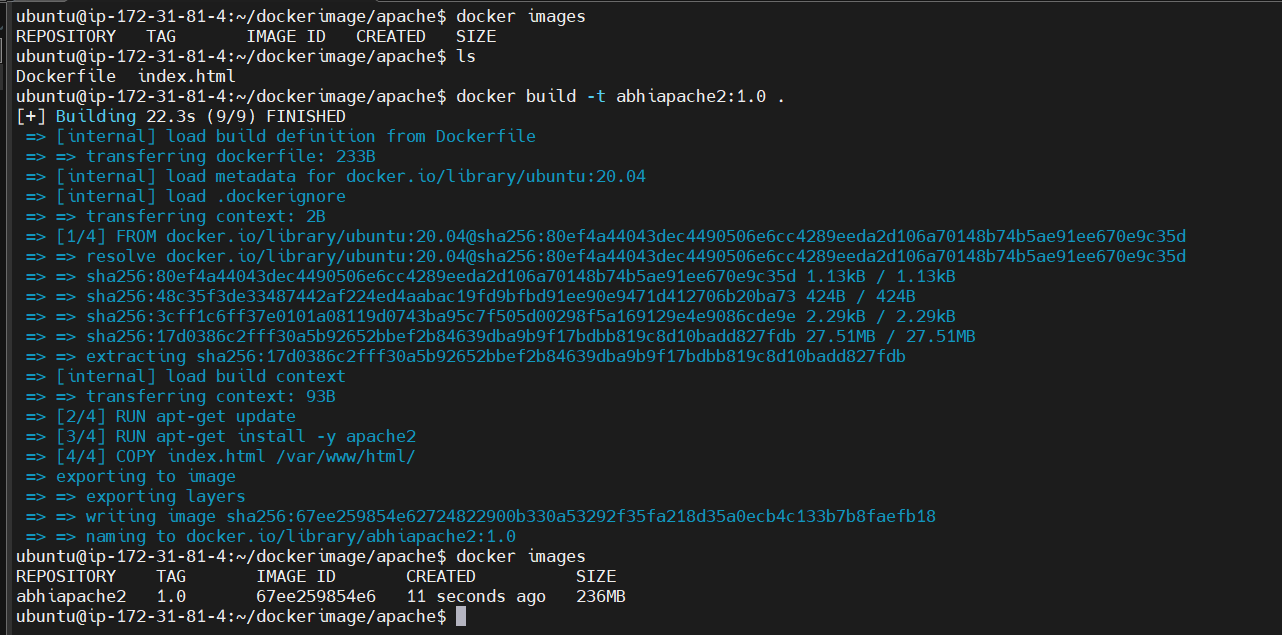
Now create docker image using following command

docker build -t abhiapache2:1.0 <path of docker file>

but I am in directory of apache2 which contain dockerfile so I will add only . like this

docker build -t abhiapache2:1.0 .

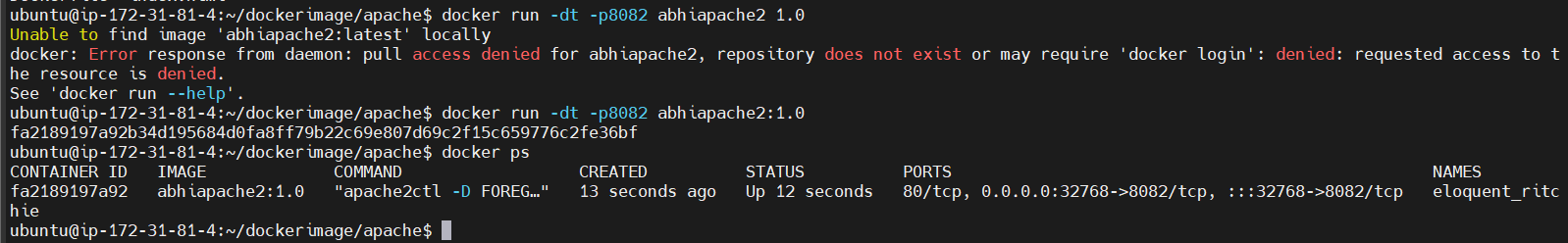
O/P



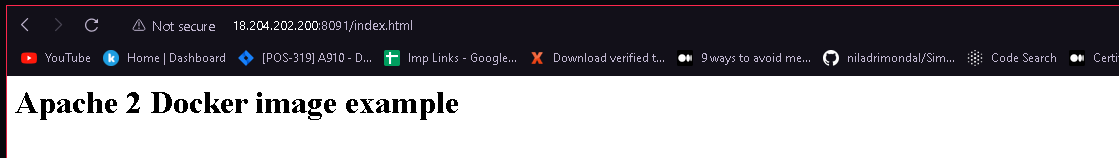
It should show 2 images Ubuntu and abhiapache2 but showing only abhiapache2

Ubuntu is our base image

Now run created docker image in detach mode



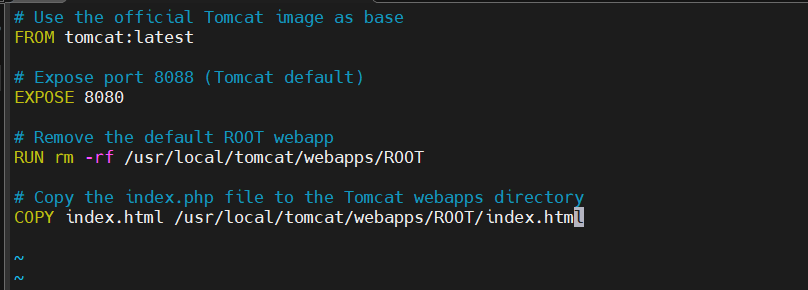
To see the output on browser add port on security group first then put ip and port/index.html in browser



-------------------- **Host HTML file on TOM CAT SERVER** --------------------------------------

Step1: make directory of tomcat

Step2: vi Dockerfile



In this Dockerfile:

We start with the official Tomcat base image.

We expose the default Tomcat port 8080.

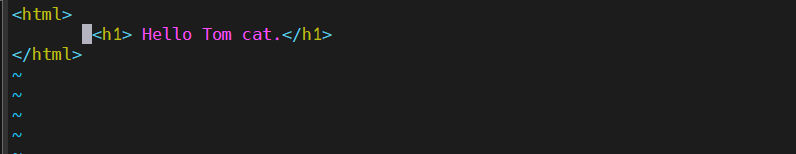
We remove the default ROOT web application that comes with Tomcat.

We copy your index.html file into the Tomcat webapps directory under the name ROOT/index.html. This will serve as the default web application.

Make sure you have your index.html file in the same directory as your Dockerfile

Step3: Create here index.html file

vi index.html



Step 4: Now create tomcat image using build command along with version number

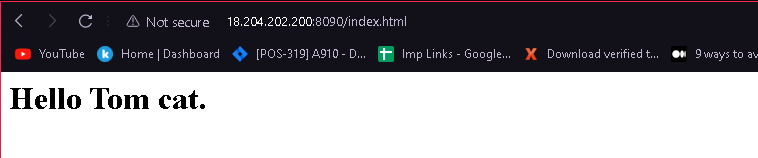
docker build -t abhi\_tomcat:1.1 .

note index.html are in same directory of our Dockerfile hence used . other wise define the .html file path

Step5 Now run Docker command

docker run -d -p 8090:8080 abhi\_tomcat:1.1

FinalStep: to see the output on browser first add 8090 port in security group

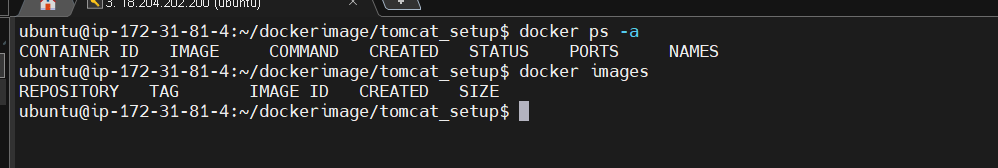


Now to remove all docker container from local use below command

docker rm -f $(docker ps -aq)

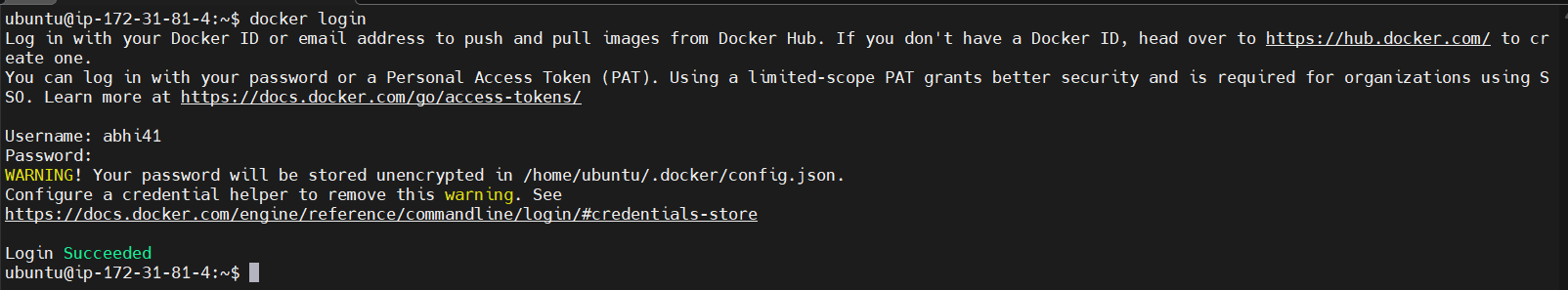
To remove all docker images from local use this command

docker rmi -f $(docker images -aq)



**--------- How to push Tomcat based image on docker hub --------------**

Step1: first login to Docker hub account



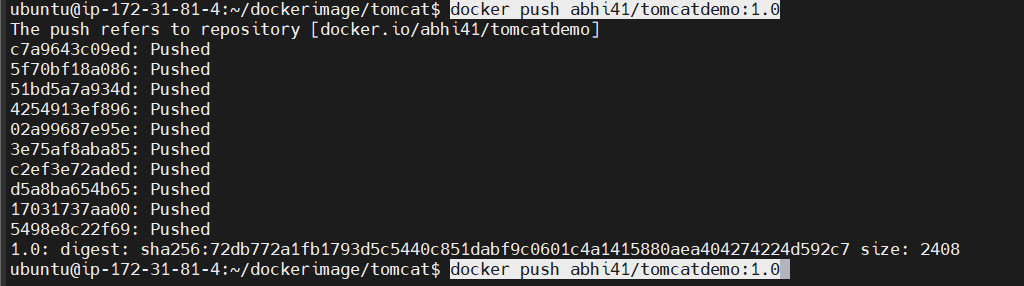
Step2: create docker build but make sure its name should be dockerhub\_username/tomcatdemo like this



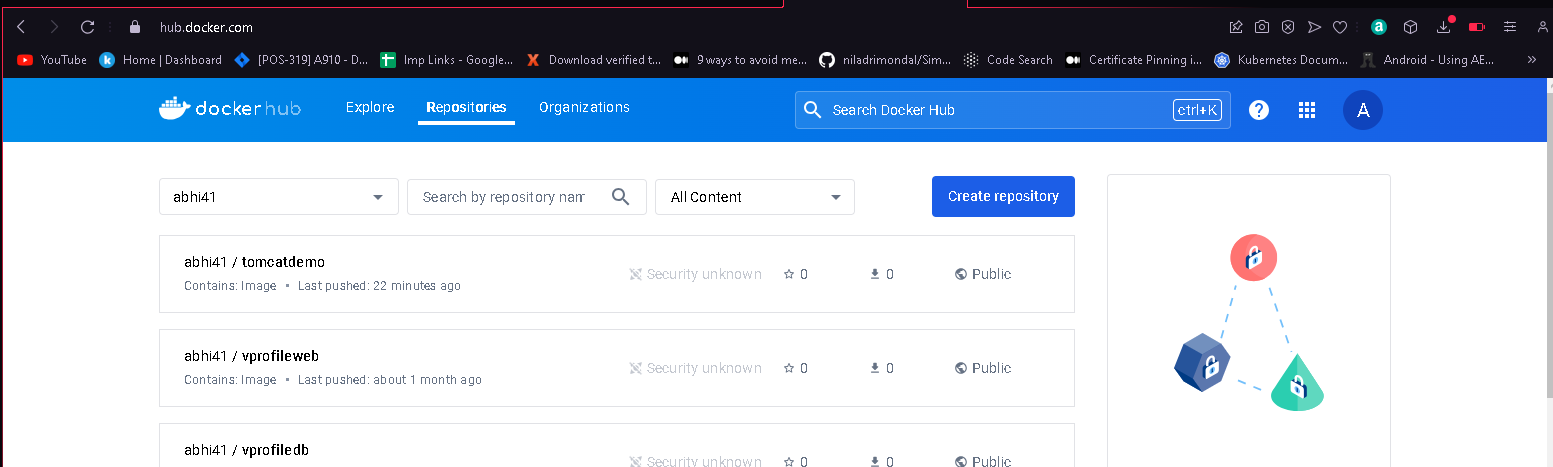
Step3: push image in docker hub using push command and don’t forget to define tag

docker push abhi41/tomcatdemo:1.0

o/p



Finally check on docker hub repository weather it is uploaded or not.



**-----------------------------------------Docker Compose-----------------------------------------**

1) What is Docker Compose ?

ans: \* Compose is a tool for defining and running multi-container Docker application.

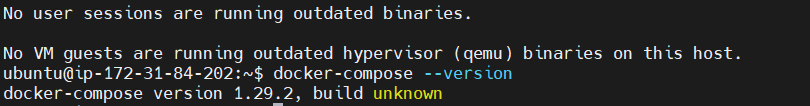
\* with Compose, you use a YAML file to configure your application's services.

\* then with single command , you create and start all services from your configuration

\* Compose works in all enviroments: production, staging, development, testing, as well as CI workflows

Installation of Docker Compose in Ubuntu

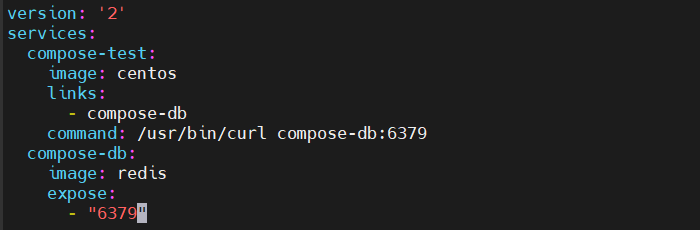
sudo apt install docker-compose



Create docker compose file before that create directory

mkdir dockercompose

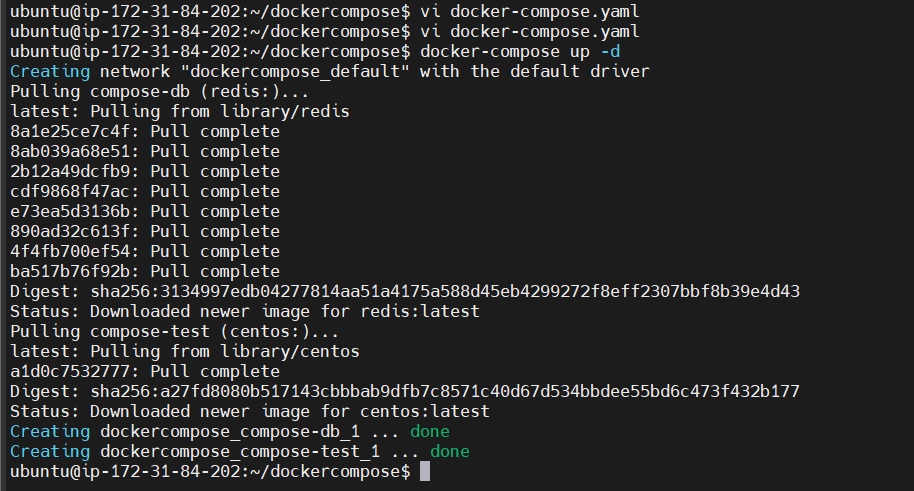
create docker-compose file

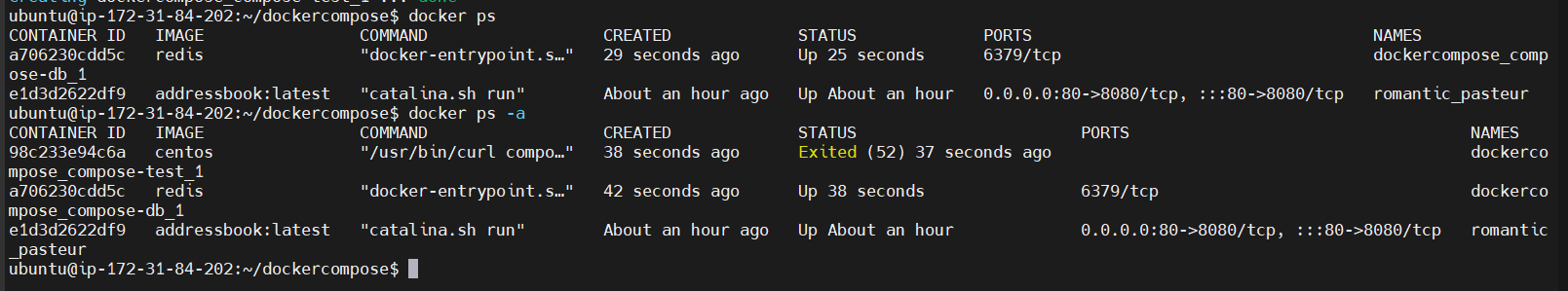


Now run Docker Compose file

docker-compose up -d

O/P





Basically centos container is depend on redis that’s why redis first get created