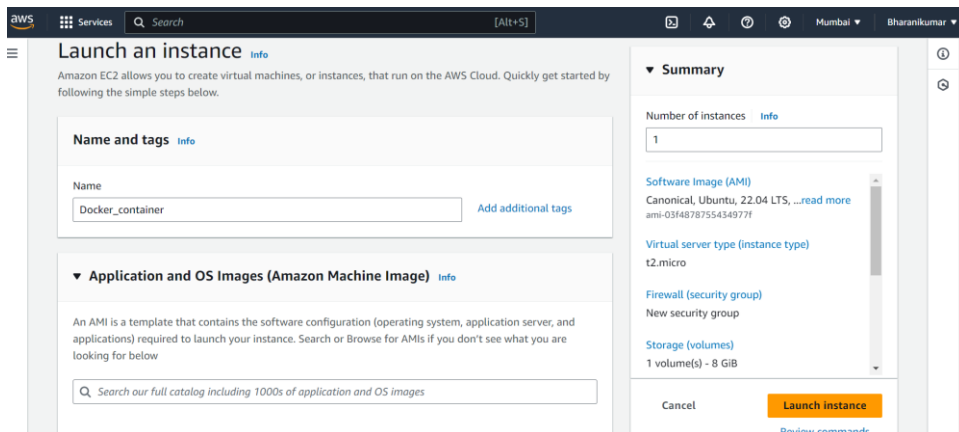


# DOCKER

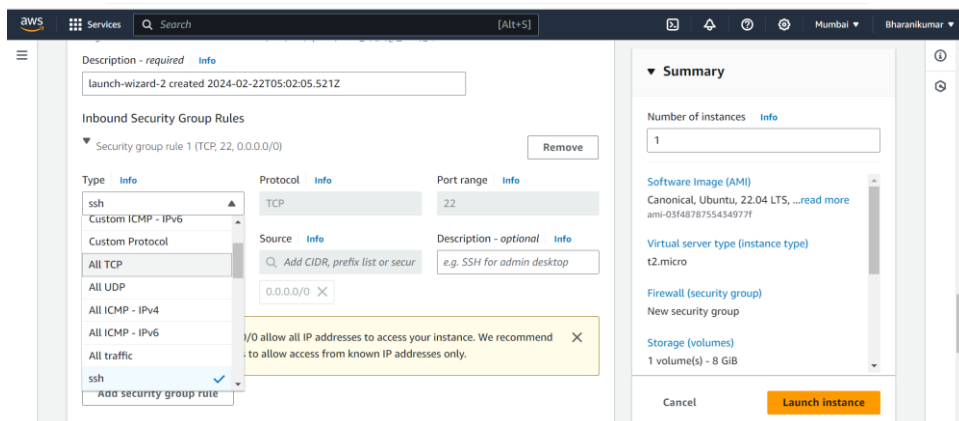
Bharani kumar.B

ACE11691

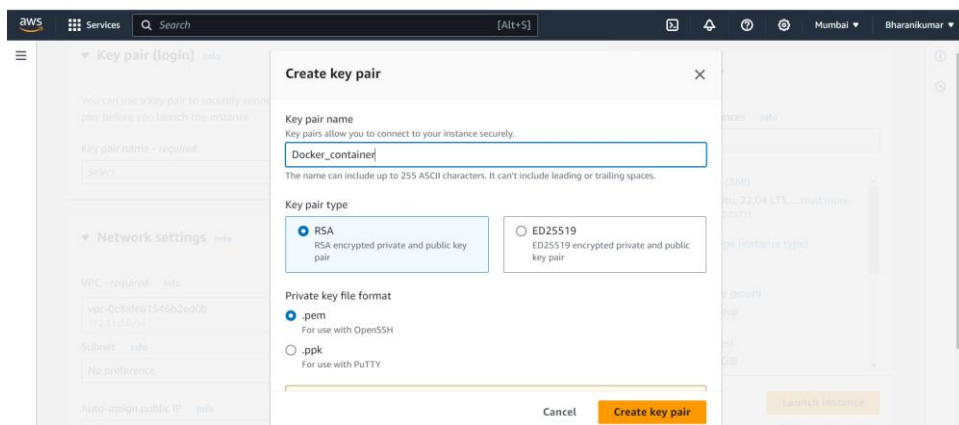
1.Open AWS > create one instance name as Docker\_container



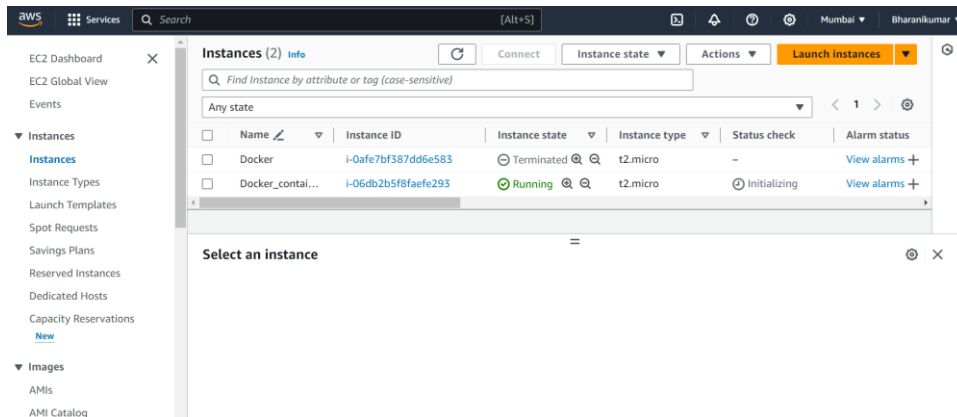
2.Change security groups



3.Create key pair > launch instance



#### 4. Click Docker\_container instance > connect



#### 5. Connect after > type the commands

Reference: [Install Docker Engine on Ubuntu | Docker Docs](#)

- `sudo apt-get update`
- `sudo apt-get install ca-certificates curl`
- `sudo install -m 0755 -d /etc/apt/keyrings`
- `sudo curl -fsSL https://download.docker.com/linux/ubuntu/gpg -o /etc/apt/keyrings/docker.asc`
- `sudo chmod a+r /etc/apt/keyrings/docker.asc`
- # Add the repository to Apt sources:  
`echo \`  
`"deb [arch=$(dpkg --print-architecture) signed-`  
`by=/etc/apt/keyrings/docker.asc]`  
`https://download.docker.com/linux/ubuntu \`  
`$(. /etc/os-release && echo "$VERSION_CODENAME") stable" | \`  
`sudo tee /etc/apt/sources.list.d/docker.list > /dev/null`
- `sudo apt-get update`
- Install the latest version
- `sudo apt-get install docker-ce docker-ce-cli containerd.io`  
`docker-buildx-plugin docker-compose-plugin`
- `sudo docker run hello-world`

```
aws Services Search [Alt+S] Mumbai Bharanikumar

Stable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

The list of available updates is more than a week old.
To check for new updates run: sudo apt update

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@ip-172-31-35-206:~$

i-06db2b5f8faefe293 (Docker_container)
PublicIPs: 13.201.122.90 PrivateIPs: 172.31.35.206
```

```
aws Services Search [Alt+S] Mumbai Bharanikumar

Get:4 https://download.docker.com/linux/ubuntu jammy InRelease [48.8 kB]
Hit:5 http://security.ubuntu.com/ubuntu jammy-security InRelease
Get:6 https://download.docker.com/linux/ubuntu jammy/stable amd64 Packages [26.7 kB]
Fetched 75.5 kB in 1s (94.6 kB/s)
Reading package lists... Done
ubuntu@ip-172-31-35-206:~$ sudo apt-get install docker-ce docker-ce-cli containerd.io docker-buildx-plugin docker-compose-plugin
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  docker-ce-rootless-extras libltdl7 libsllp0 pigz slirp4netns
Suggested packages:
  aufs-tools cgroups-mount | cgroup-lite
The following NEW packages will be installed:
  containerd.io docker-buildx-plugin docker-ce docker-ce-cli docker-ce-rootless-extras docker-compose-plugin libltdl7 libsllp0 pigz
  slirp4netns
0 upgraded, 10 newly installed, 0 to remove and 74 not upgraded.
Need to get 117 MB of archives.
After this operation, 420 MB of additional disk space will be used.
Do you want to continue? [Y/n] Y

i-06db2b5f8faefe293 (Docker_container)
PublicIPs: 13.201.122.90 PrivateIPs: 172.31.35.206
```

```
aws Services Search [Alt+S] Mumbai Bharanikumar

Setting up docker-ce-rootless-extras (5:25.0.3-1~ubuntu.22.04~jammy) ...
Setting up slirp4netns (1.0.1-2) ...
Setting up docker-ce (5:25.0.3-1~ubuntu.22.04~jammy) ...
Created symlink /etc/systemd/system/multi-user.target.wants/docker.service → /lib/systemd/system/docker.service.
Created symlink /etc/systemd/system/sockets.target.wants/docker.socket → /lib/systemd/system/docker.socket.
Processing triggers for man-db (2.10.2-1) ...
Processing triggers for libc-bin (2.35-0ubuntu3.4) ...
Scanning processes...
Scanning linux images...

Running kernel seems to be up-to-date.

No services need to be restarted.

No containers need to be restarted.

No user sessions are running outdated binaries.

No VM guests are running outdated hypervisor (qemu) binaries on this host.
ubuntu@ip-172-31-35-206:~$

i-06db2b5f8faefe293 (Docker_container)
PublicIPs: 13.201.122.90 PrivateIPs: 172.31.35.206
```

```
aws Services Search [Alt+S] Mumbai Bharanikumar

c1ec31eb5944: Pull complete
Digest: sha256:d000bc569937abbe195e20322a0bde6b2922d805332fd6d8a68b19f524b7d21d
Status: Downloaded newer image for hello-world:latest

Hello from Docker!
This message shows that your installation appears to be working correctly.

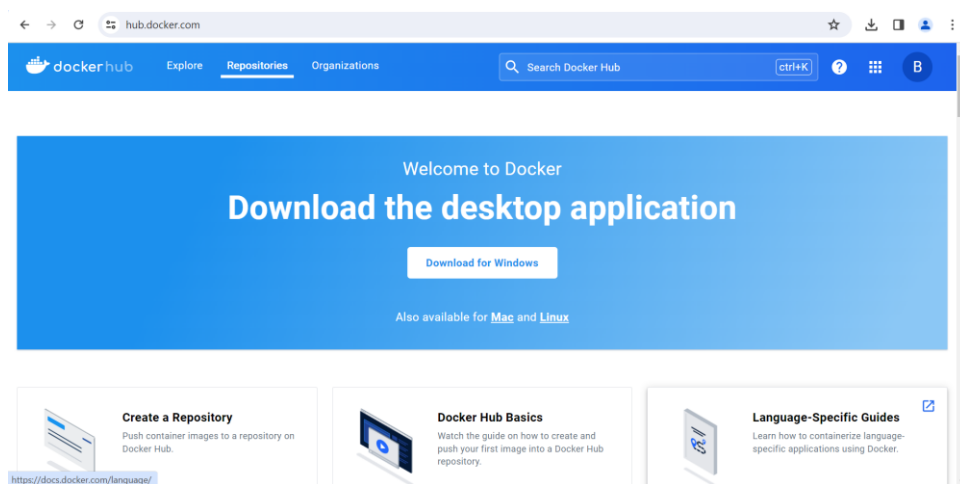
To generate this message, Docker took the following steps:
1. The Docker client contacted the Docker daemon.
2. The Docker daemon pulled the "hello-world" image from the Docker Hub.
   (amd64)
3. The Docker daemon created a new container from that image which runs the
   executable that produces the output you are currently reading.
4. The Docker daemon streamed that output to the Docker client, which sent it
   to your terminal.

To try something more ambitious, you can run an Ubuntu container with:
$ docker run -it ubuntu bash

Share images, automate workflows, and more with a free Docker ID:

i-06db2b5f8faefe293 (Docker_container)
PublicIPs: 13.201.122.90 PrivateIPs: 172.31.35.206
```

## 9. Go to > Docker hub

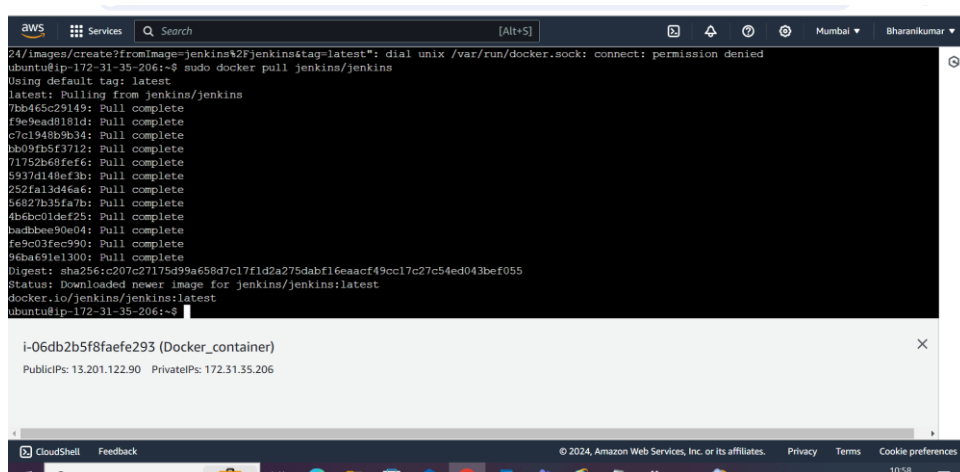


## 10. Type the commands

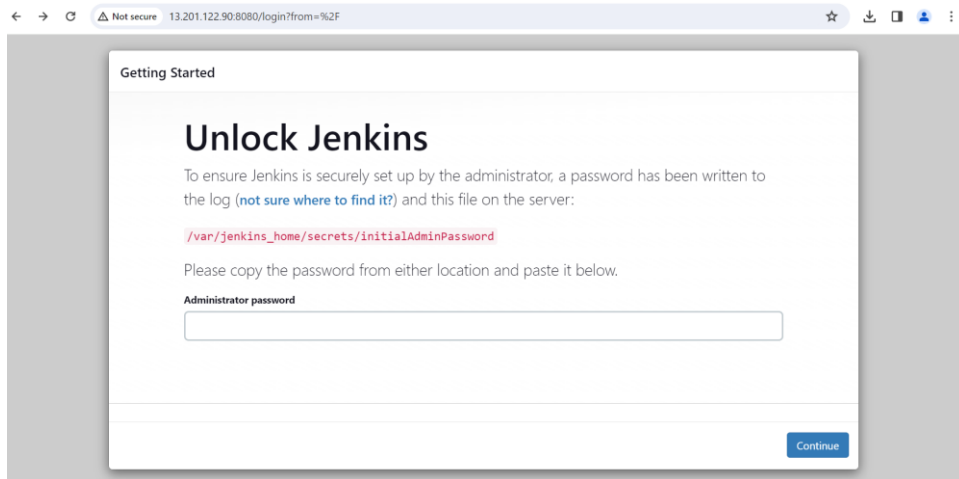
1.) `sudo docker pull jenkins/jenkins`

2.) `sudo docker run -d -p 8080:8080 -p 50000:50000 --name jenkins_container jenkins/jenkins`

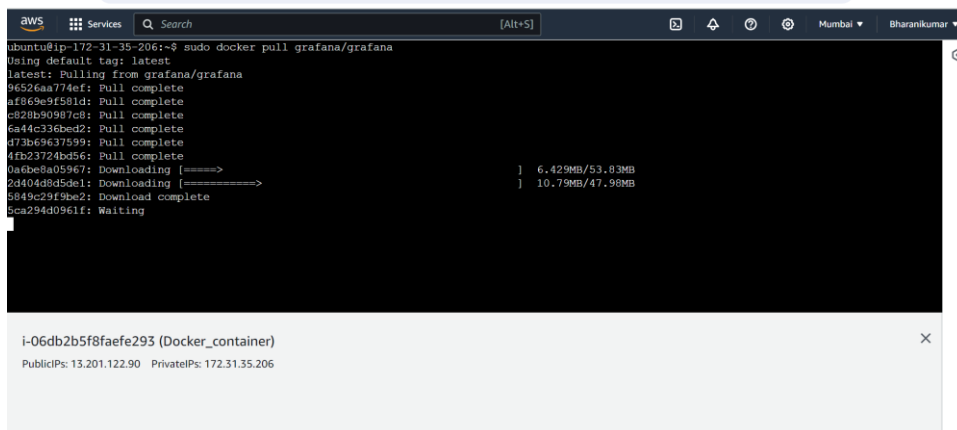
This command is for installing jenkins into docker



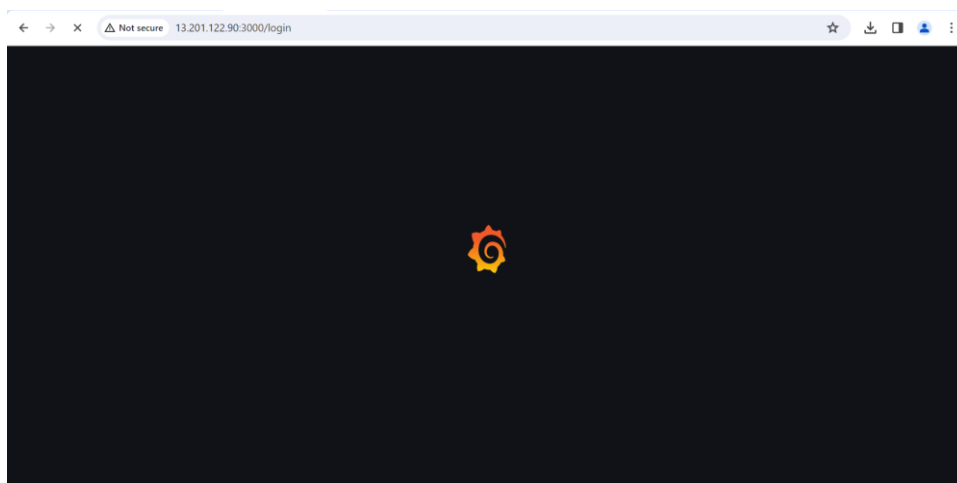
## 11. Now jenkins is opened below

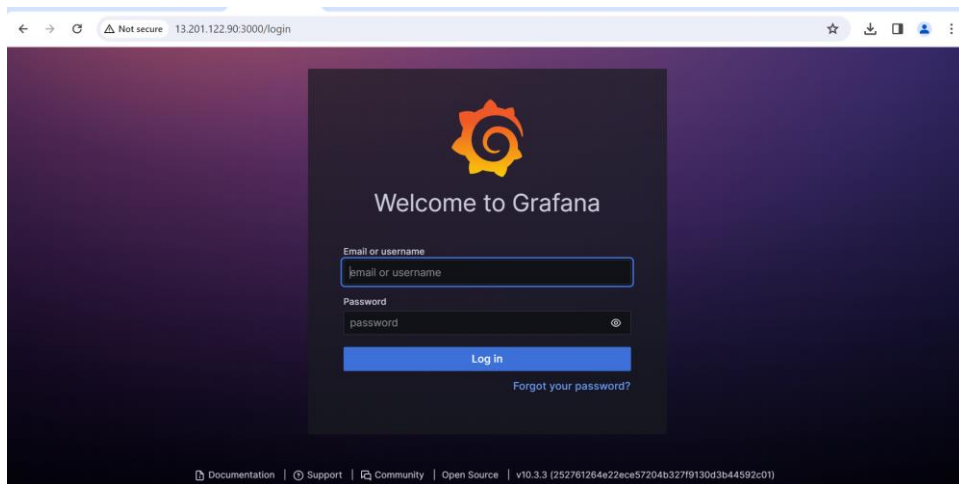


12. Now I am pulling Grafana from Docker Hub  
**command** = `docker pull grafana/grafana`



13. Now Grafana is opened





14.iam going to use httpd in docker by pull command

```
aws Services Search [Alt+S] Mumbai Bharanikumar
er.sock/v1.24/containers/json": dial unix /var/run/docker.sock: connect: permission denied
ubuntu@ip-172-31-35-206:~$ sudo docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED        STATUS        PORTS                               NAMES
9dccc60ec70b   httpd    "httpd-foreground"      5 minutes ago Up 5 minutes   80/tcp                               e8
e8a448800927   httpd    "httpd-foreground"      12 minutes ago Up 12 minutes   0.0.0.0:9090->80/tcp, :::9090->80/tcp cranky_s
aha
f143dc66d8ec   httpd    "httpd-foreground"      About an hour ago Up About an hour 80/tcp                               webserve
r
ubuntu@ip-172-31-35-206:~$ sudo docker exec -it e8 /bin/bas
OCI runtime exec failed: exec failed: unable to start container process: exec: "/bin/bas": stat /bin/bas: no such file or directory:
unknown
ubuntu@ip-172-31-35-206:~$ sudo docker run -d --e8 httpd
unknown flag: --e8
See 'docker run --help'.
ubuntu@ip-172-31-35-206:~$ sudo docker run -d --name webserver httpd
docker: Error response from daemon: Conflict. The container name "/webserver" is already in use by container "f143dc66d8ec5ef6915c110
4b012db487dedd2727ec88fel7d0f5fe77c7009f4". You have to remove (or rename) that container to be able to reuse that name.
See 'docker run --help'.
ubuntu@ip-172-31-35-206:~$ sudo docker run -d --name demo httpd
36fe388e86444dbd0ae193bf118c472c97363f022f6619bdaf9bfeb64f6dbf9
ubuntu@ip-172-31-35-206:~$ docker exec -it 36 /bin/bash
i-06db2b5f8faefe293 (Docker_container)
PublicIPs: 65.2.137.31 PrivateIPs: 172.31.35.206

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences
ubuntu@ip-172-31-35-206:~$ sudo docker exec -it e8 /bin/bas
OCI runtime exec failed: exec failed: unable to start container process: exec: "/bin/bas": stat /bin/bas: no such file or directory:
unknown
ubuntu@ip-172-31-35-206:~$ sudo docker run -d --e8 httpd
unknown flag: --e8
See 'docker run --help'.
ubuntu@ip-172-31-35-206:~$ sudo docker run -d --name webserver httpd
docker: Error response from daemon: Conflict. The container name "/webserver" is already in use by container "f143dc66d8ec5ef6915c110
4b012db487dedd2727ec88fel7d0f5fe77c7009f4". You have to remove (or rename) that container to be able to reuse that name.
See 'docker run --help'.
ubuntu@ip-172-31-35-206:~$ sudo docker run -d --name demo httpd
36fe388e86444dbd0ae193bf118c472c97363f022f6619bdaf9bfeb64f6dbf9
ubuntu@ip-172-31-35-206:~$ docker exec -it 36 /bin/bash
permission denied while trying to connect to the Docker daemon socket at unix:///var/run/docker.sock: Get "http://%2Fvar%2Frun%2Fdock
er.sock/v1.24/containers/36/json": dial unix /var/run/docker.sock: connect: permission denied
ubuntu@ip-172-31-35-206:~$ sudo docker exec -it 36 /bin/bash
root@36fe388e8644:/usr/local/apache2# ls -lrt
total 40
drwxr-xr-x 2 root root 4096 Feb 13 01:55 modules
drwxr-xr-x 3 root root 4096 Feb 13 01:55 icons
```

```
aws Services Search [Alt+S] Mumbai Bharanikumar
See 'docker run --help'.
ubuntu@ip-172-31-35-206:~$ sudo docker run -d --name demo httpd
36fe388e86444dbd0ae193b1f118c472c97363f022f6619bdaf9bfeb64f6dbf9
ubuntu@ip-172-31-35-206:~$ docker exec -it 36 /bin/bash
permission denied while trying to connect to the Docker daemon socket at unix:///var/run/docker.sock: Get "http://%2Fvar%2Frun%2Fdock
er.sock/v1.24/containers/36/json": dial unix /var/run/docker.sock: connect: permission denied
ubuntu@ip-172-31-35-206:~$ sudo docker exec -it 36 /bin/bash
root@36fe388e8644:/usr/local/apache2# ls -lrt
total 40
drwxr-xr-x 2 root root 4096 Feb 13 01:55 modules
drwxr-xr-x 3 root root 4096 Feb 13 01:55 icons
drwxr-xr-x 2 root root 4096 Feb 13 01:55 htdocs
drwxr-xr-x 3 root root 4096 Feb 13 01:55 error
drwxr-xr-x 2 root root 4096 Feb 13 01:55 cgi-bin
drwxr-xr-x 2 root root 4096 Feb 13 01:55 include
drwxr-xr-x 2 root root 4096 Feb 13 01:55 bin
drwxr-xr-x 4 root root 4096 Feb 13 01:55 conf
drwxr-xr-x 2 root root 4096 Feb 13 01:55 build
drwxr-xr-x 1 root root 4096 Feb 22 18:15 logs
root@36fe388e8644:/usr/local/apache2# cd htdocs
root@36fe388e8644:/usr/local/apache2/htdocs#
```

i-06db2b5f8fafe293 (Docker\_container)

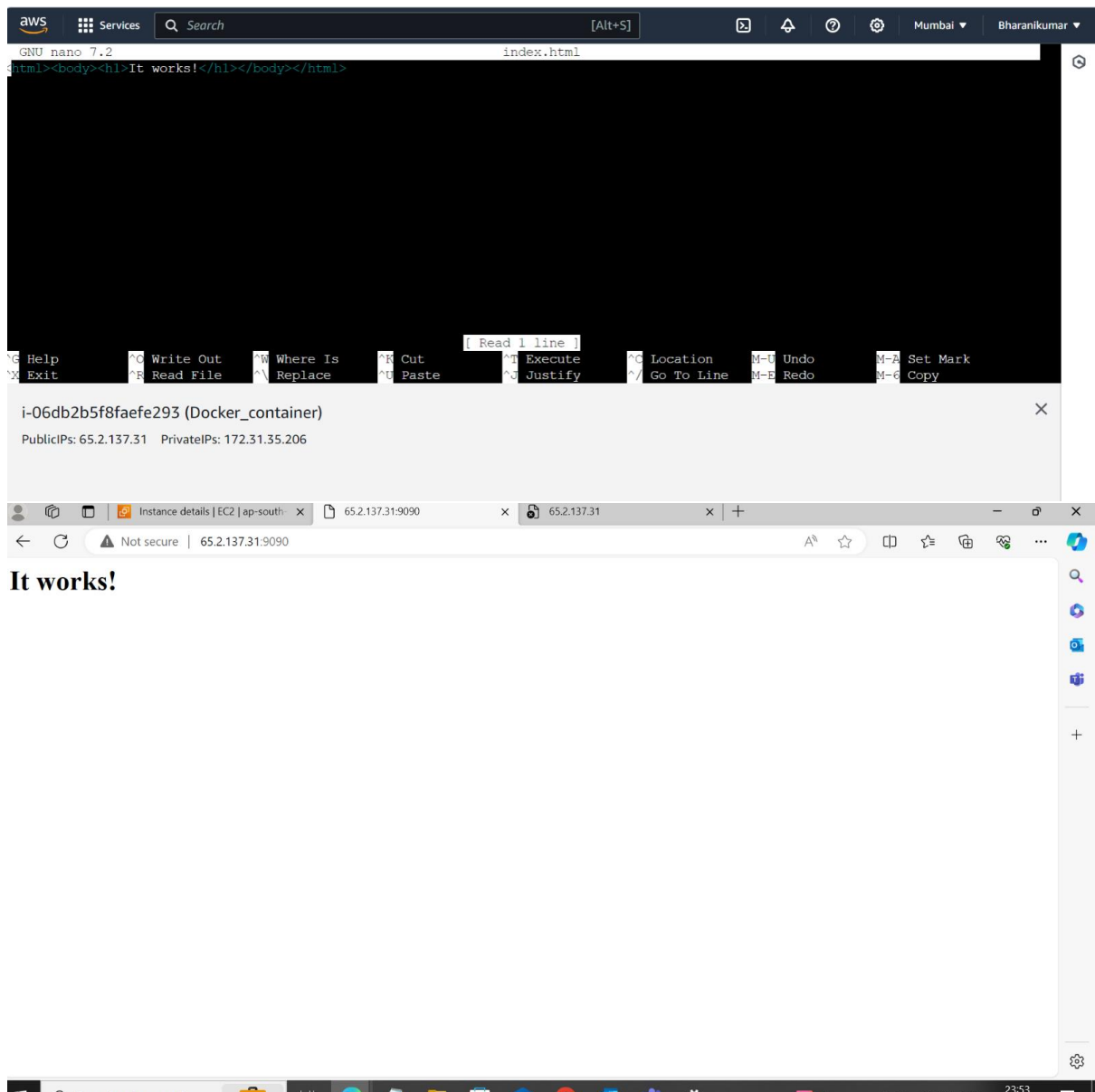
PublicIPs: 65.2.137.31 PrivateIPs: 172.31.35.206

```
aws Services Search [Alt+S] Mumbai Bharanikumar
36fe388e86444dbd0ae193b1f118c472c97363f022f6619bdaf9bfeb64f6dbf9
ubuntu@ip-172-31-35-206:~$ docker exec -it 36 /bin/bash
permission denied while trying to connect to the Docker daemon socket at unix:///var/run/docker.sock: Get "http://%2Fvar%2Frun%2Fdock
er.sock/v1.24/containers/36/json": dial unix /var/run/docker.sock: connect: permission denied
ubuntu@ip-172-31-35-206:~$ sudo docker exec -it 36 /bin/bash
root@36fe388e8644:/usr/local/apache2# ls -lrt
total 40
drwxr-xr-x 2 root root 4096 Feb 13 01:55 modules
drwxr-xr-x 3 root root 4096 Feb 13 01:55 icons
drwxr-xr-x 2 root root 4096 Feb 13 01:55 htdocs
drwxr-xr-x 3 root root 4096 Feb 13 01:55 error
drwxr-xr-x 2 root root 4096 Feb 13 01:55 cgi-bin
drwxr-xr-x 2 root root 4096 Feb 13 01:55 include
drwxr-xr-x 2 root root 4096 Feb 13 01:55 bin
drwxr-xr-x 4 root root 4096 Feb 13 01:55 conf
drwxr-xr-x 2 root root 4096 Feb 13 01:55 build
drwxr-xr-x 1 root root 4096 Feb 22 18:15 logs
root@36fe388e8644:/usr/local/apache2# cd htdocs
root@36fe388e8644:/usr/local/apache2/htdocs# ls
index.html
root@36fe388e8644:/usr/local/apache2/htdocs# nano index.html
```

i-06db2b5f8fafe293 (Docker\_container)

PublicIPs: 65.2.137.31 PrivateIPs: 172.31.35.206

```
root@36fe388e8644:/usr/local/apache2/htdocs# apt-get update
apt-get install nano
```



15.Iam going to customize my html content

To customize html like below

```
<html><body><h1>Hi bharani kumar</h1></body></html>
```

**1.control+0 =store html**

**2.Enter**

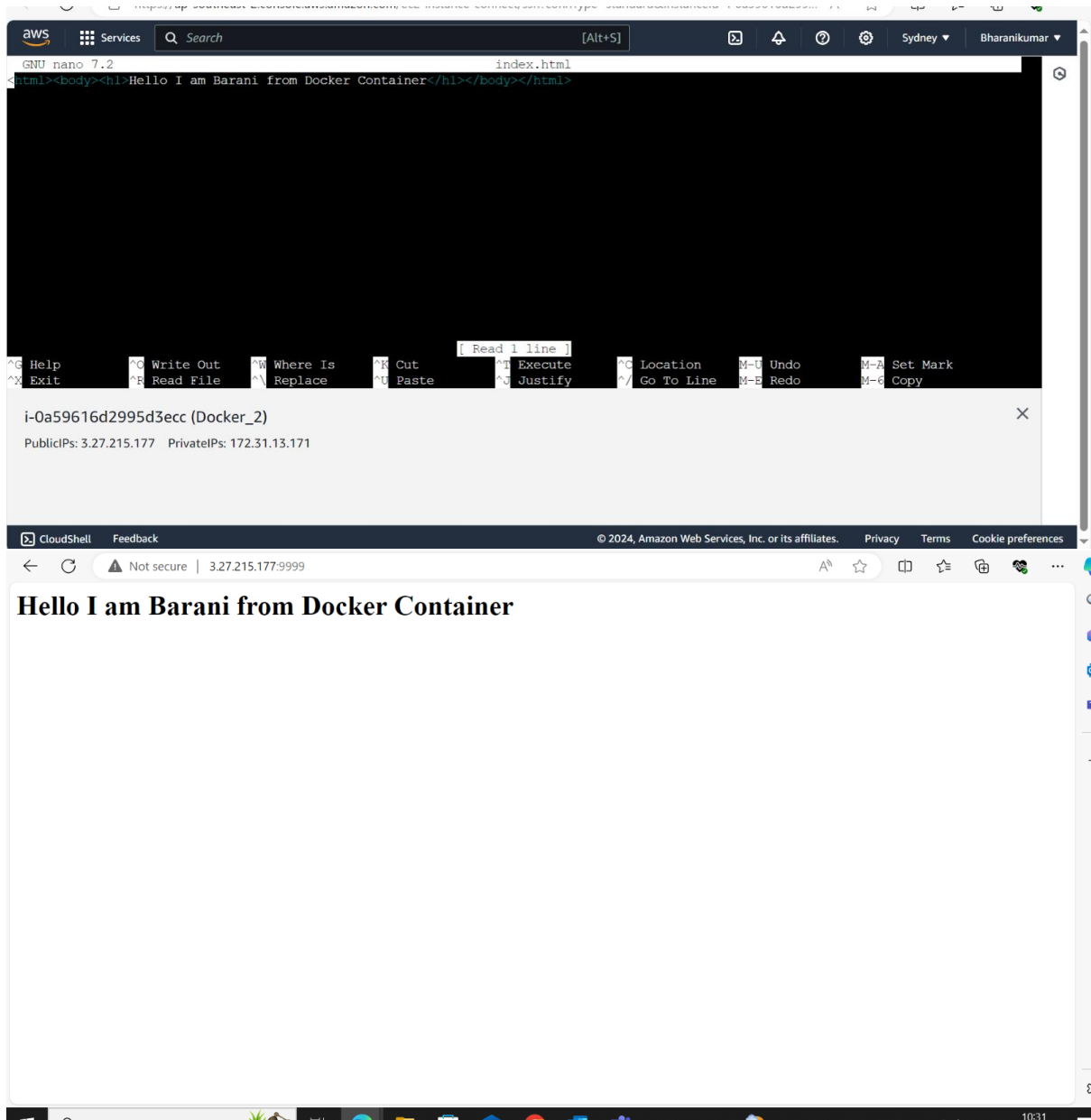
**3.control+x =exit from html**

**4.exit =in shell**

**5.sudo docker restart 9cb1f2c2d3e9 =restart the container**

**After enter into host and put port like Http//34.567090 :9090**





- I am using Jenkins in Docker
- I am using Grafana in Docker
- I am using httpd in Docker