

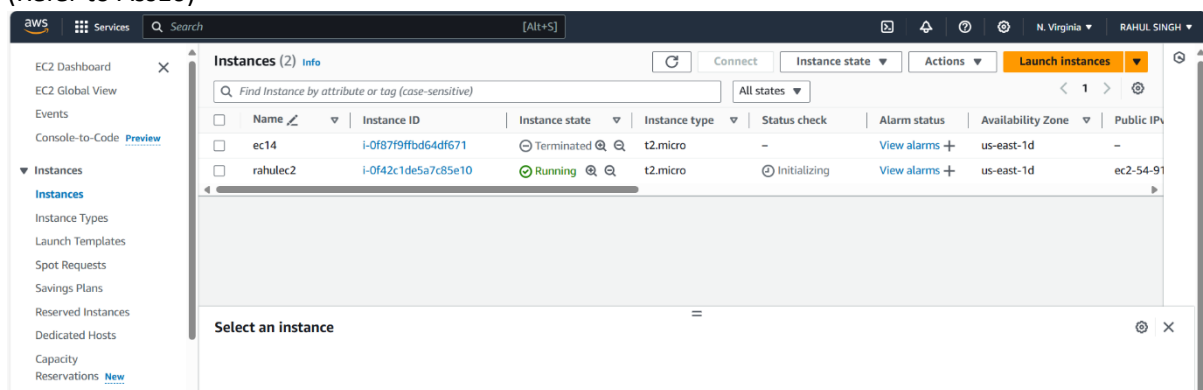
ASSIGNMENT-12

Problem Statement: Deploy a project from GitHub to EC2 without using Port.

Procedures:

1. Sign-in to AWS console.
2. Go to the EC2 dashboard. Now go to the instances page.
3. Click on the create new instance button.
4. Now create an EC2 server using the Security Group created earlier and enter the user data

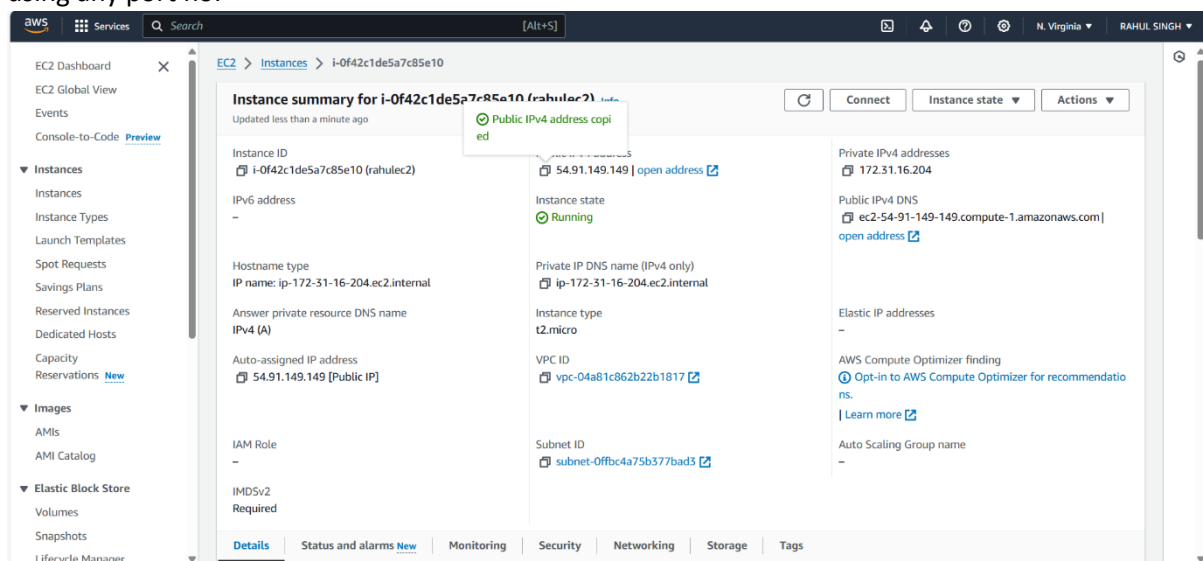
(Refer to Ass10)



5. Create the instance and click on the instance after creation.
6. Copy the public IPv4 address and paste it in another browse. The nginx homepage will show up.

Our server is working perfectly. Note, in previous assignments we used to connect to our project webpages using port no. However, in this exercise we are going to access our project webpage without

using any port no.

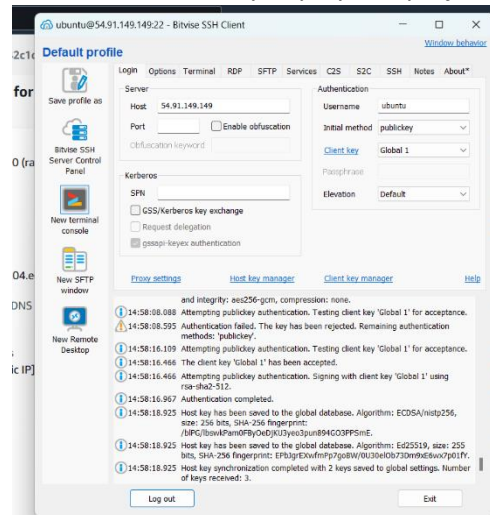


7. Copy the Public IPv4 address of the server instance and use this to connect it to the server using Bitwise

SSH client. (Refer Ass7)

8. Now open the terminal in Bitwise without entering our port no.

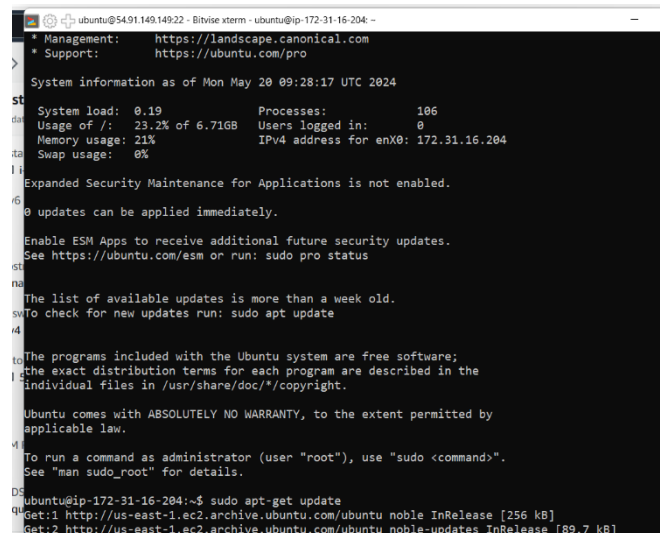
We have successfully deployed a project from GitHub to EC2 without using port.



9. Enter the following commands in it.

Sudo apt - get update

Sudo apt -get upgrade



10. Write the command `nginx -v` to check the version

```
Service restarts being deferred:
/etc/needrestart/restart.d/dbus.service
systemctl restart getty@tty1.service
systemctl restart networkd-dispatcher.service
systemctl restart serial-getty@ttyS0.service
systemctl restart systemd-logind.service
systemctl restart unattended-upgrades.service

No containers need to be restarted.

User sessions running outdated binaries:
ubuntu @ session #1: sshd[989,1113]
ubuntu @ user manager service: systemd[1008]

No VM guests are running outdated hypervisor (qemu) binaries on this host.
ubuntu@ip-172-31-16-204:~$ nginx -v
nginx version: nginx/1.24.0 (Ubuntu)
ubuntu@ip-172-31-16-204:~$
```

11. Write `curl -SL https://deb.nodesource.com/setup_18.x | sudo -E bash -`

```
systemctl restart unattended-upgrades.service

No containers need to be restarted.

User sessions running outdated binaries:
ubuntu @ session #1: sshd[989,1113]
ubuntu @ user manager service: systemd[1008]

No VM guests are running outdated hypervisor (qemu) binaries on this host.
ubuntu@ip-172-31-16-204:~$ nginx -v
nginx version: nginx/1.24.0 (Ubuntu)
ubuntu@ip-172-31-16-204:~$ curl -SL https://deb.nodesource.com/setup_18.x | sudo -E bash -
```

12. Then we have to install nodejs for installing nodejs we write the command `sudo apt install nodejs`.

```
/etc/needrestart/restart.d/dbus.service
systemctl restart getty@tty1.service
systemctl restart networkd-dispatcher.service
systemctl restart serial-getty@ttyS0.service
systemctl restart systemd-logind.service
systemctl restart unattended-upgrades.service

No containers need to be restarted.

User sessions running outdated binaries:
ubuntu @ session #1: sshd[989,1113]
ubuntu @ user manager service: systemd[1008]

No VM guests are running outdated hypervisor (qemu) binaries on this host.
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble InRelease
Hit:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates InRelease
Hit:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease
Get:4 https://deb.nodesource.com/node_18.x nodistro InRelease [12.1 kB]
Hit:5 http://security.ubuntu.com/ubuntu noble-security InRelease
Get:6 https://deb.nodesource.com/node_18.x nodistro/main amd64 Packages [8669 B]
Fetched 20.8 kB in 1s (29.0 kB/s)
Reading package lists... Done
2024-05-20 09:37:09 - Repository configured successfully. To install Node.js, run: apt-get install nodejs
ubuntu@ip-172-31-16-204:~$ sudo apt install nodejs
```

13. To check the version we write `nodejs -v`

```
systemctl restart systemd-logind.service
systemctl restart unattended-upgrades.service

No containers need to be restarted.

User sessions running outdated binaries:
ubuntu @ session #1: sshd[989,1113]
ubuntu@ip-172-31-16-204:~$ nodejs -v
v18.20.2
ubuntu@ip-172-31-16-204:~$
```

14 Enter the following commands in it.

`pwd`

(To check current directory)

`cd /`

(To go to root folder)

`pwd`

write `ls`.

```
No containers need to be restarted.

User sessions running outdated binaries:
ubuntu @ session #1: sshd[989,1113]
ubuntu@ip-172-31-16-204:~$ nodejs -v
v18.20.2
ubuntu@ip-172-31-16-204:~$ cd /
ubuntu@ip-172-31-16-204:/$ pwd
/
ubuntu@ip-172-31-16-204:/$ sudo git clone https://github.com/RahulSinghrajpt/ass8.git
```

```
ubuntu@ip-172-31-16-204:/$ sudo git clone https://github.com/RahulSinghrajpt/ass8.git
Cloning into 'ass8'...
remote: Enumerating objects: 10, done.
remote: Counting objects: 100% (10/10), done.
remote: Compressing objects: 100% (10/10), done.
remote: Total 10 (delta 2), reused 5 (delta 0), pack-reused 0
Receiving objects: 100% (10/10), 49.12 KiB | 7.02 MiB/s, done.
Resolving deltas: 100% (2/2), done.
ubuntu@ip-172-31-16-204:/$ ls
ass8  boot  home  lib64  mnt  root  sbin  usr-is-merged  sys  var
bin    dev  lib  lost+found  opt  run  snap  tmp  usr
bin usr-is-merged  etc  lib usr-is-merged  media  proc  sbin  srv
ubuntu@ip-172-31-16-204:/$ cd ass8/
ubuntu@ip-172-31-16-204:/ass8$ ls
'New Text Document.txt'  index.js  package-lock.json  package.json
ubuntu@ip-172-31-16-204:/ass8$
```

15. For npm installing we write `sudo npm i` means install.

```
ubuntu@ip-172-31-16-204:/$ cd ass8/
ubuntu@ip-172-31-16-204:/ass8$ ls
'New Text Document.txt'  index.js  package-lock.json  package.json
ubuntu@ip-172-31-16-204:/ass8$ sudo npm i
npm WARN deprecated uid@3.4.0: Please upgrade to version 7 or higher. Older versions may use Math.random() in certain circumstances, which is known to be problematic. See https://v8.dev/blog/math-random for details.

added 258 packages, and audited 259 packages in 9s

18 packages are looking for funding
  run npm fund for details

12 vulnerabilities (10 moderate, 2 critical)

To address all issues, run:
  npm audit fix

Run 'npm audit' for details.
npm notice
npm notice New minor version of npm available! 10.5.0 -> 10.8.0
npm notice Changelog: https://github.com/npm/cli/releases/tag/v10.8.0
```

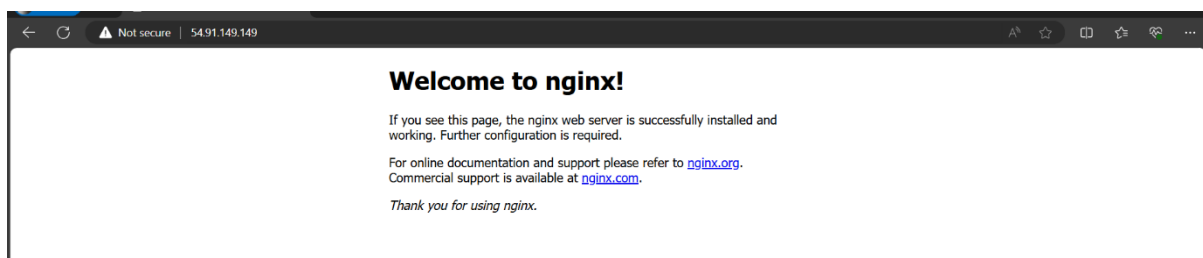
16. After installing npm we have to check the version of npm for checking the version of npm we write `npm -v`.

```
npm notice
npm notice New minor version of npm available! 10.5.0 -> 10.8.0
npm notice Changelog: https://github.com/npm/cli/releases/tag/v10.8.0
npm notice Run npm install -g npm@10.8.0 to update!
npm notice
ubuntu@ip-172-31-16-204:/ass8$ npm -v
10.5.0
ubuntu@ip-172-31-16-204:/ass8$
```

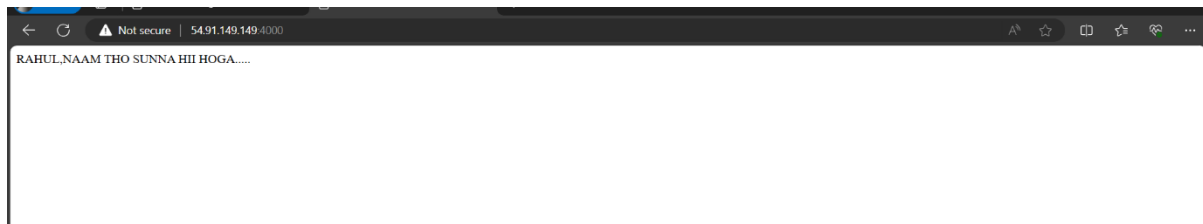
17. Then we shall start the server for stating the server we write `node index.js`

```
npm notice
ubuntu@ip-172-31-16-204:/ass8$ npm -v
10.5.0
ubuntu@ip-172-31-16-204:/ass8$ node index.js
Started server
```

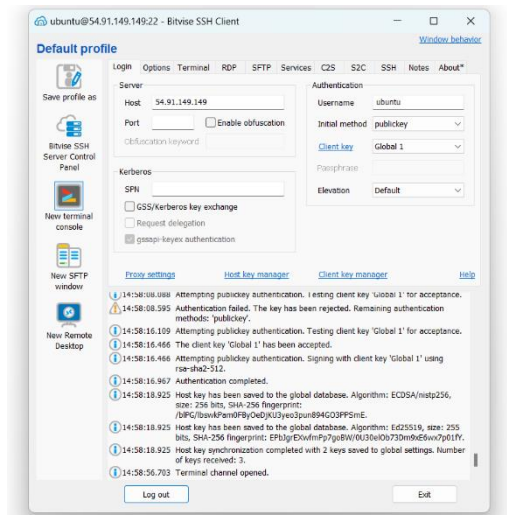
18. Copy the ipv4 public address and paste them into new window the nginx will be running.



19. Then we write the 4000 with the port to connect the index.js file.



20.Again we will open new terminal console.



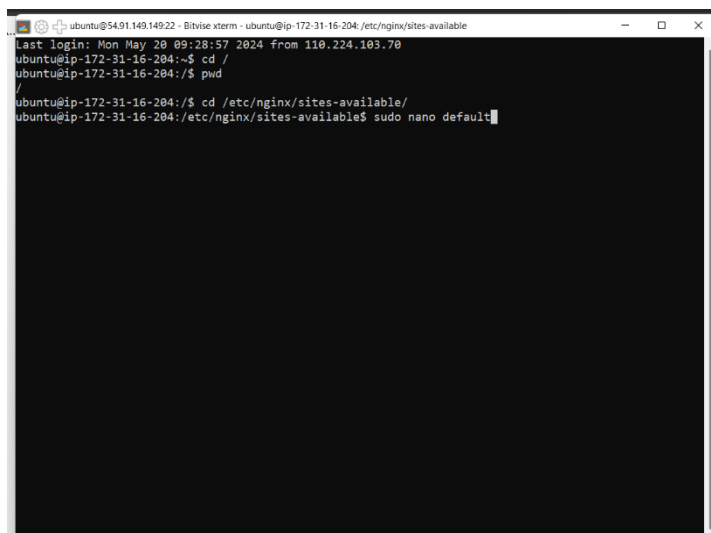
21.Here we write the following command

Cd /

Pwd

Cd /etc/nginx/sites-available/

Sudo nano default



22.The we write the new location and previous location we will comment out then write the new location.

The after we press the ctrl x for exit

Then press Y.

```
server_name _;

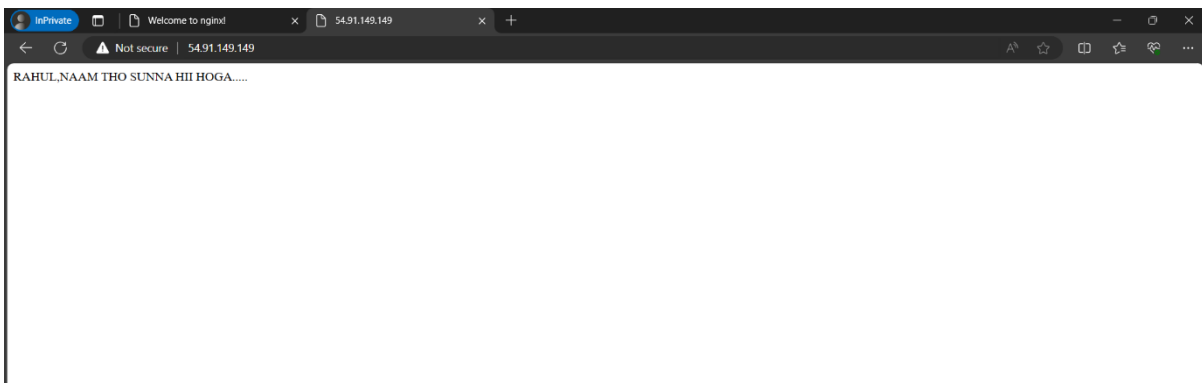
#
location / {
    # First attempt to serve request as file, then
    # as directory, then fall back to displaying a 404.
    try_files $uri $uri/ =404;
}

location / {
    proxy_pass http://localhost:4000;
    proxy_http_version 1.1;
    proxy_set_header Upgrade $http_upgrade;
    proxy_set_header Connection 'upgrade';
    proxy_set_header Host $host;
    proxy_cache_bypass $http_upgrade;
}
```

23. Again we back to our terminal.

```
ubuntu@54.91.149.149:22 - Bitvise xterm - ubuntu@ip-172-31-16-204:/etc/nginx/sites-available
Last login: Mon May 20 09:28:57 2024 from 110.224.103.70
ubuntu@ip-172-31-16-204:~$ cd /
ubuntu@ip-172-31-16-204:/$ pwd
/
ubuntu@ip-172-31-16-204:/$ cd /etc/nginx/sites-available/
ubuntu@ip-172-31-16-204:/etc/nginx/sites-available$ sudo nano default
ubuntu@ip-172-31-16-204:/etc/nginx/sites-available$ sudo systemctl restart nginx
ubuntu@ip-172-31-16-204:/etc/nginx/sites-available$
```

24. And now we open the new windows copy the ipv4 public address here we not write the :4000 port number then when we paste the port the tap the enter our project will deploy without port.



without entering our port no.

We have successfully deployed a project from GitHub to EC2 without using port.