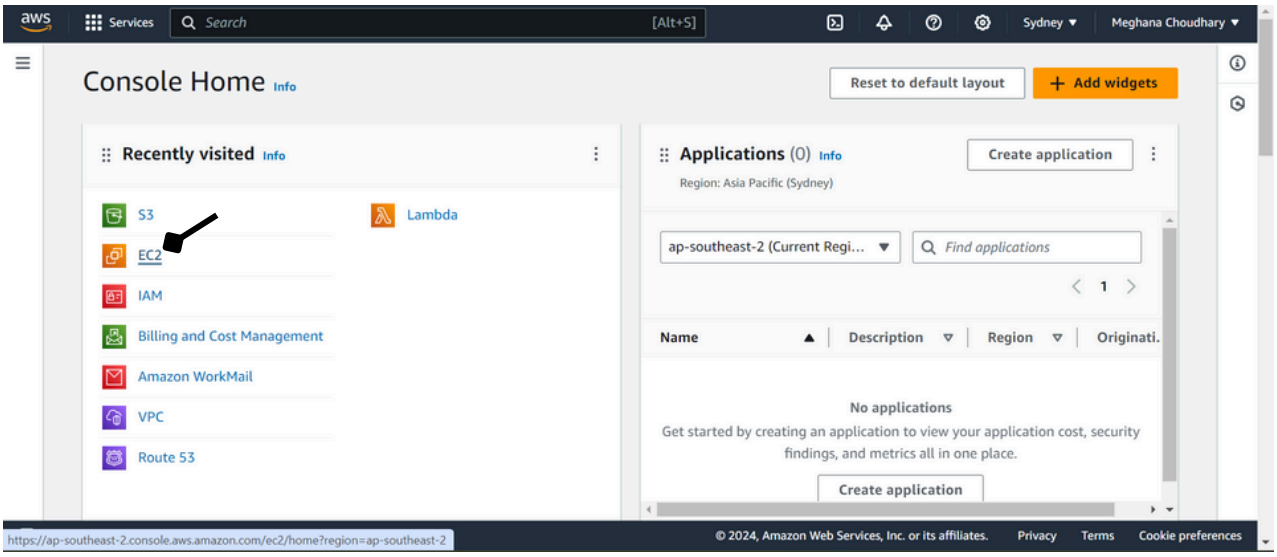
9. PROBLEM STATEMENT:

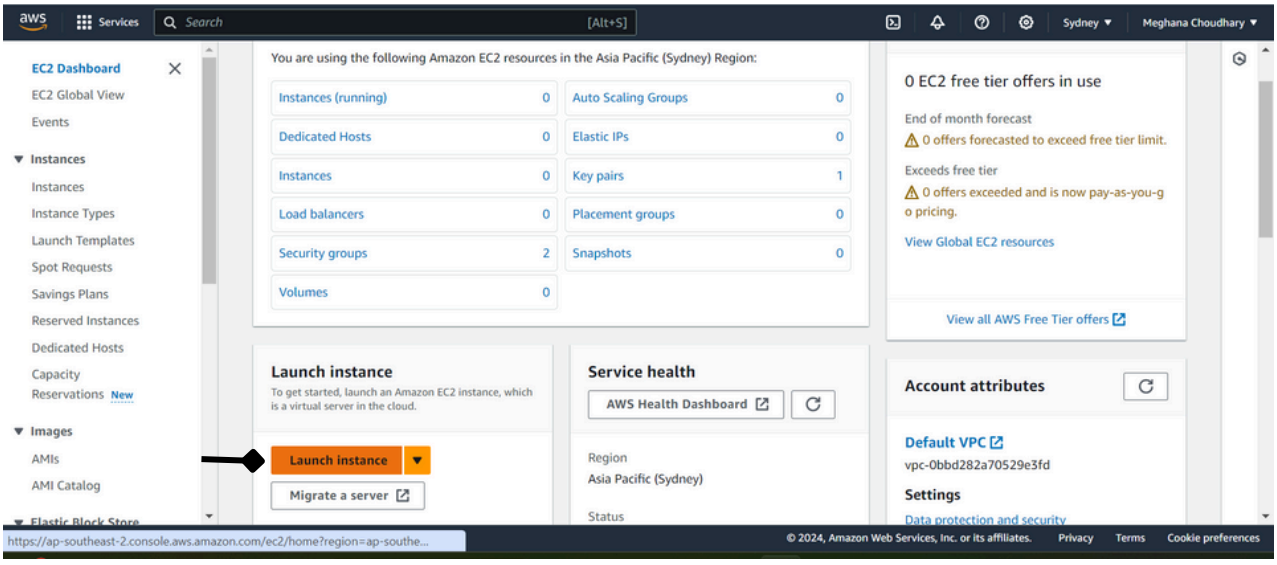
Deploy a project from GitHub to EC2

Creating a EC2 Instance with port 4000 open:-

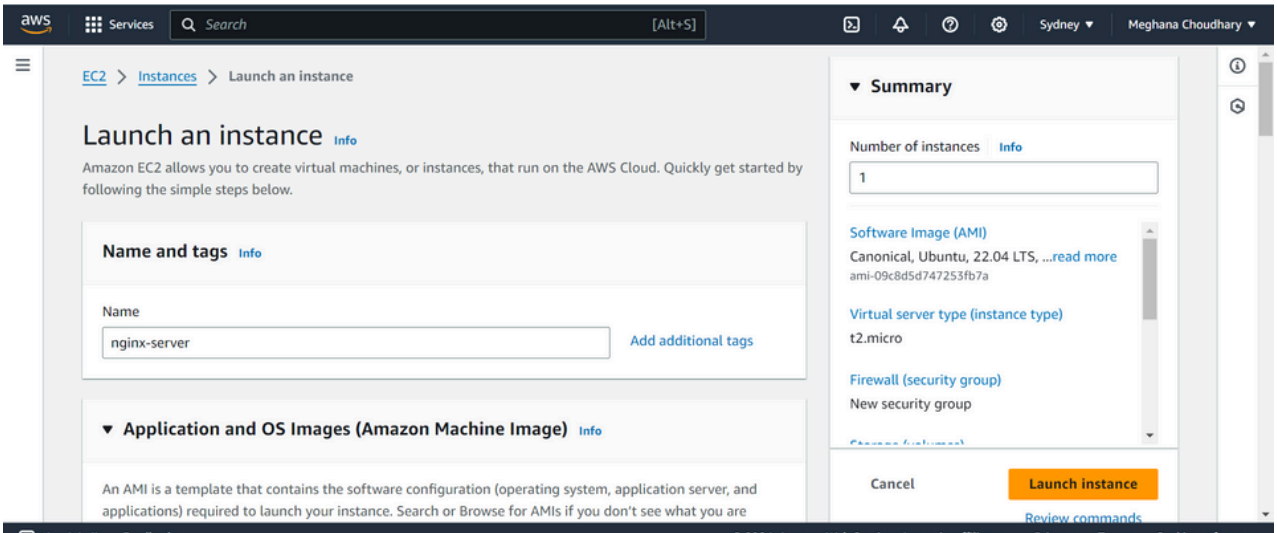
1. Log into AWS console and click on “EC2”.

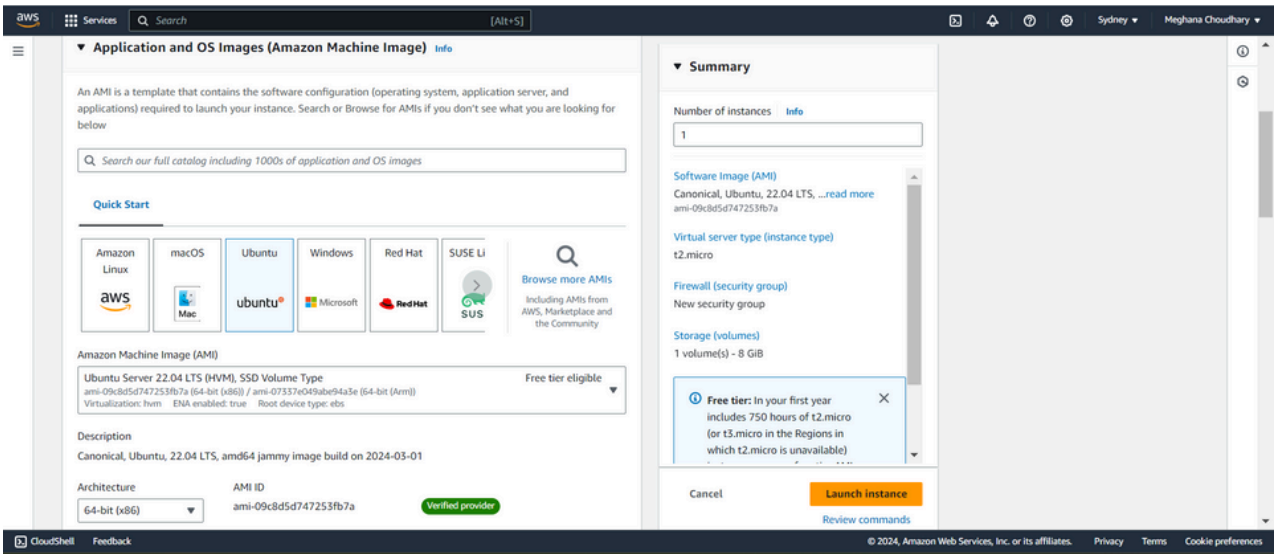


2. Click on ‘Launch instance’.

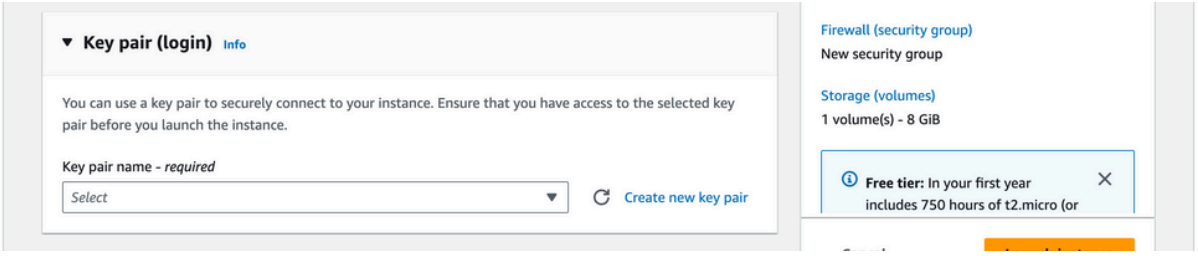


3. Fill the required details.
Give a name to your server and choose ubuntu Server as the OS Image.

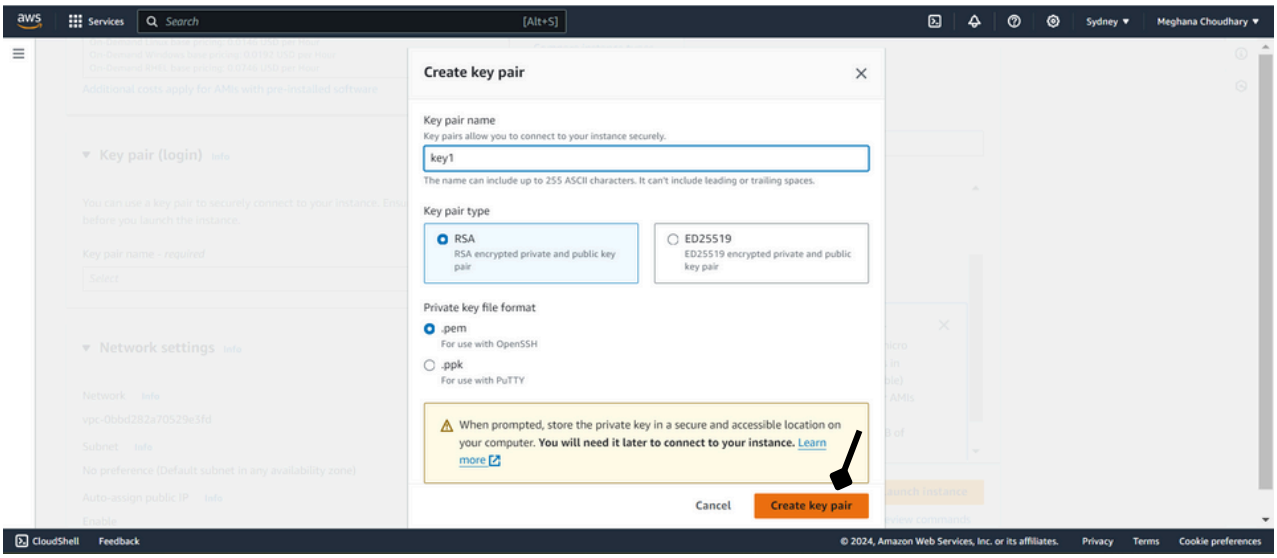




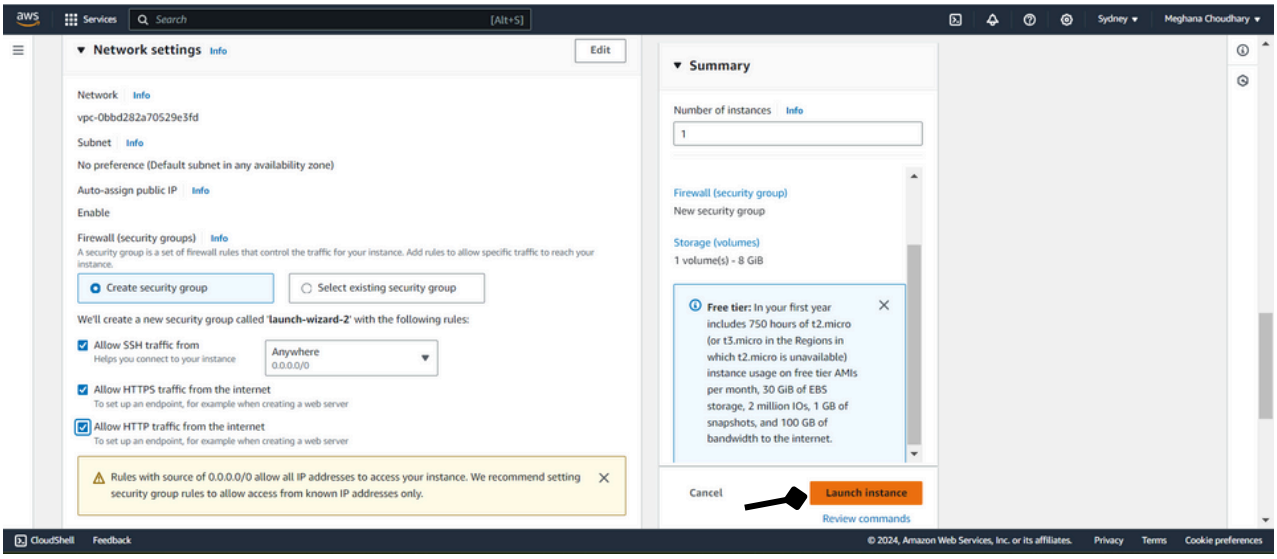
4. For Key pair, if you have a exixting key pair select that else ‘Create e new key pair’.



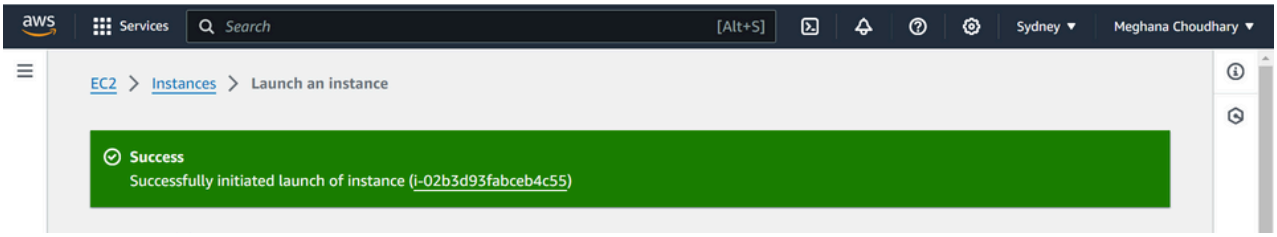
5. Give a name to the key pair. Make sure the key pair type is ‘RSA’ and the file extention is ‘.pem’.



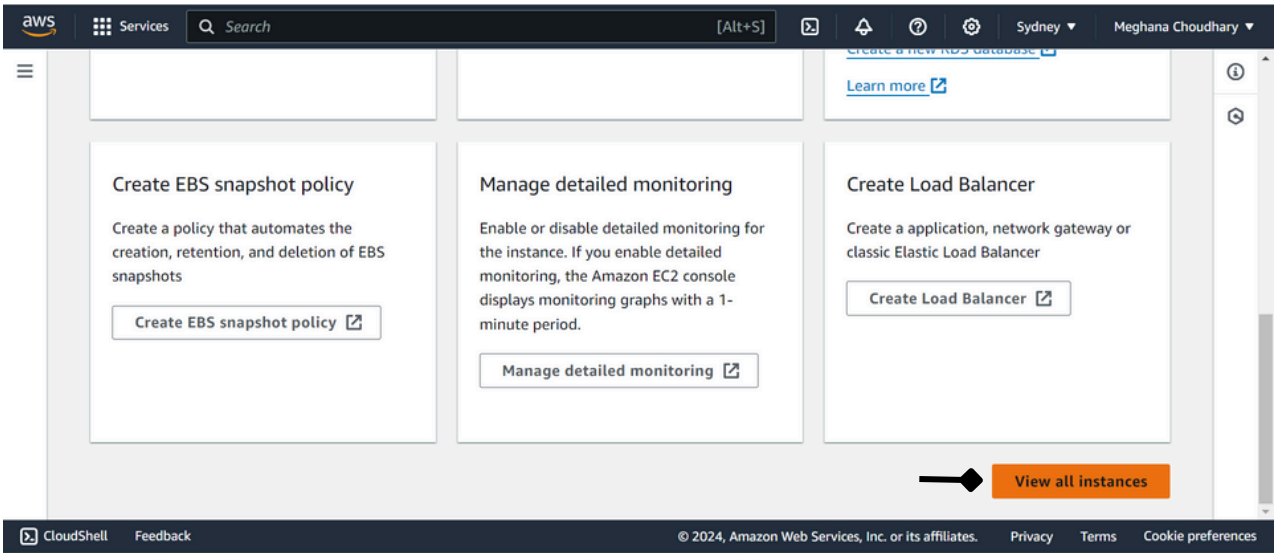
6. For 'Network Settings’ Select all the check boxes(SSH, HTTPS,HTTP). Then Click on ‘Launch instance’.



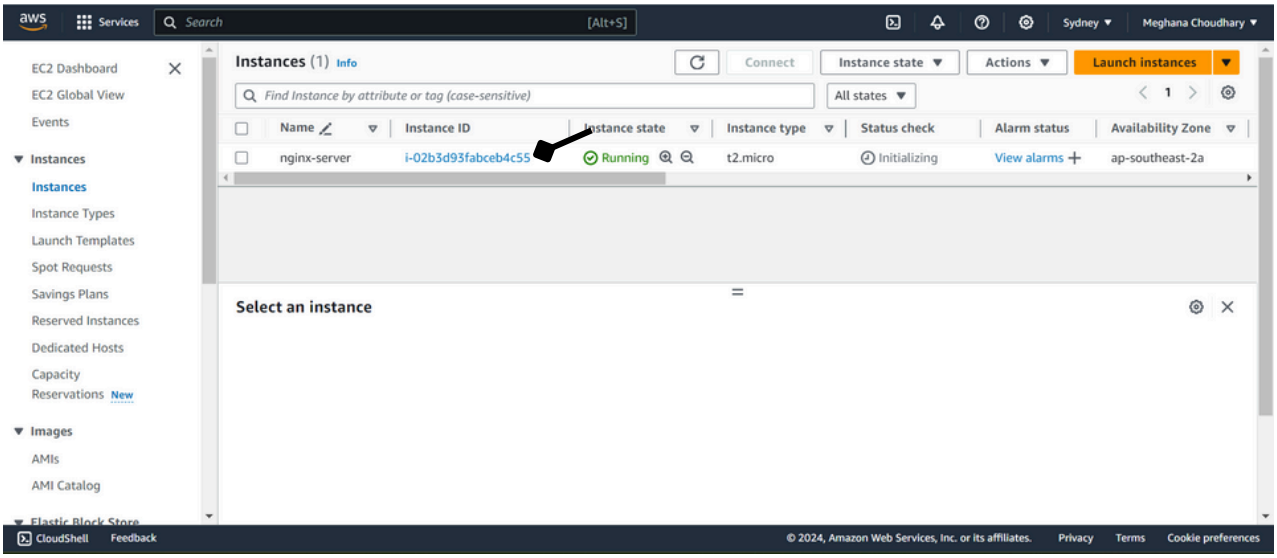
7. Your instance is created successfully.



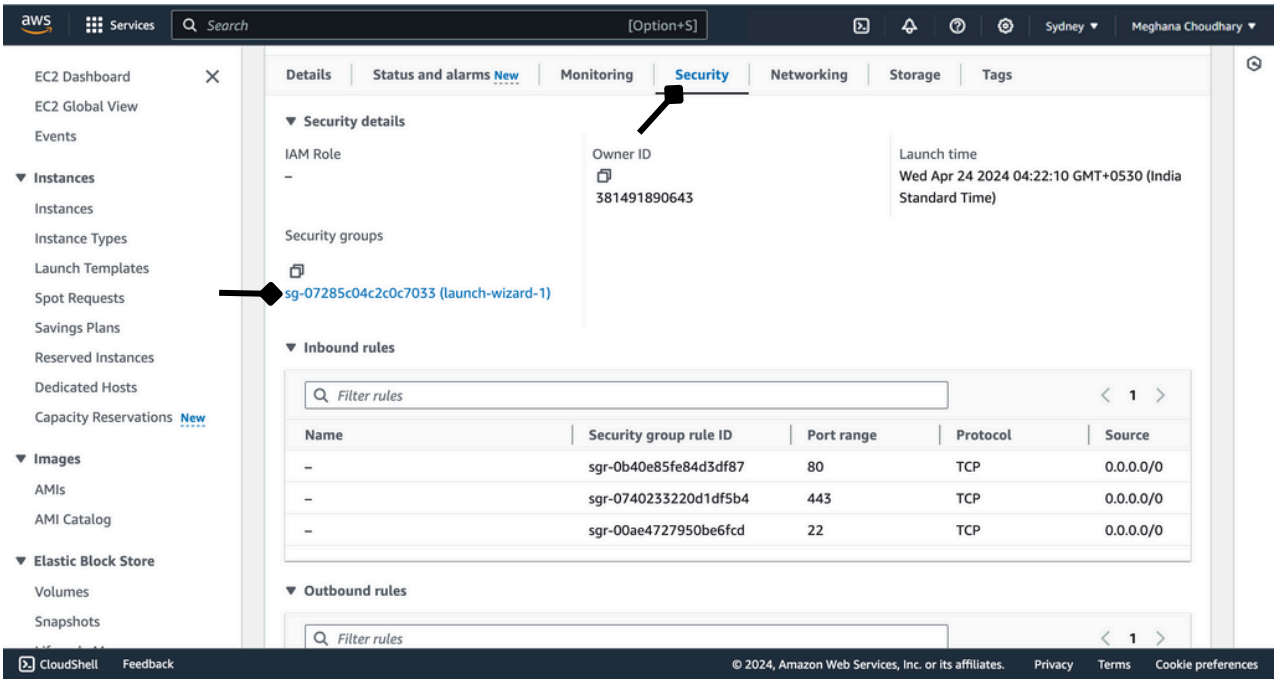
8. Then Scroll Down and Click on 'View all instances'.



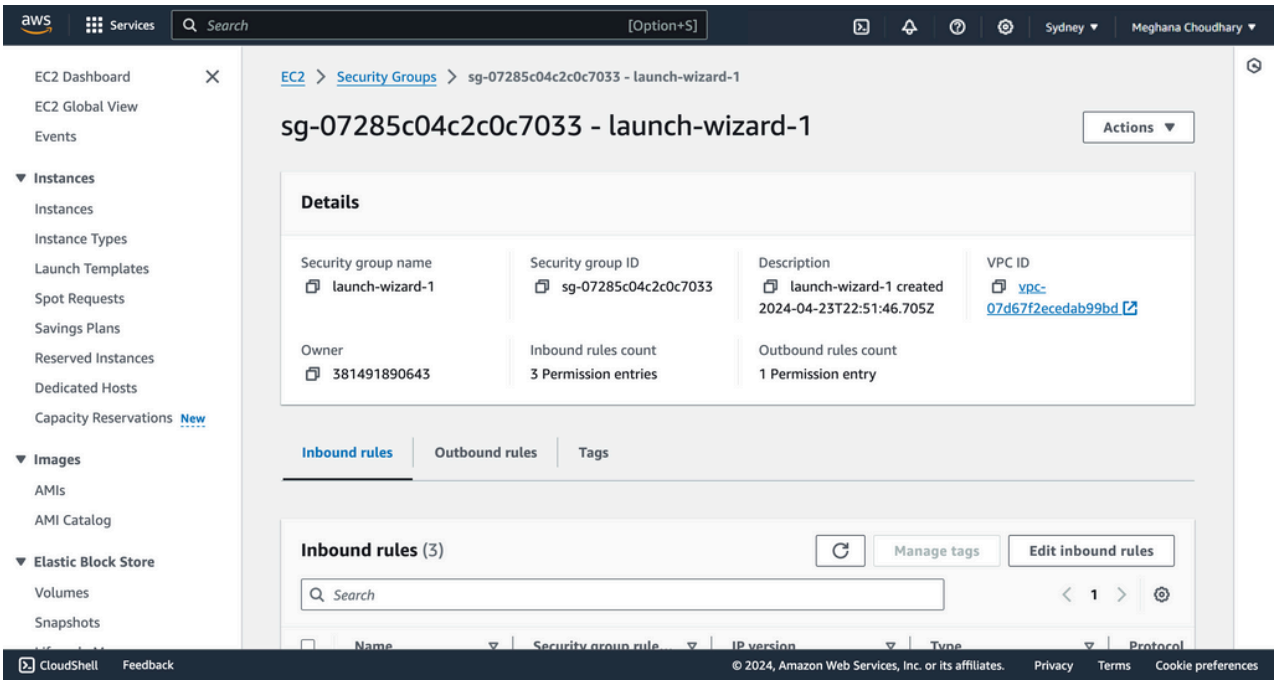
9. You can see your instance is created and running. Click on the server Instance ID



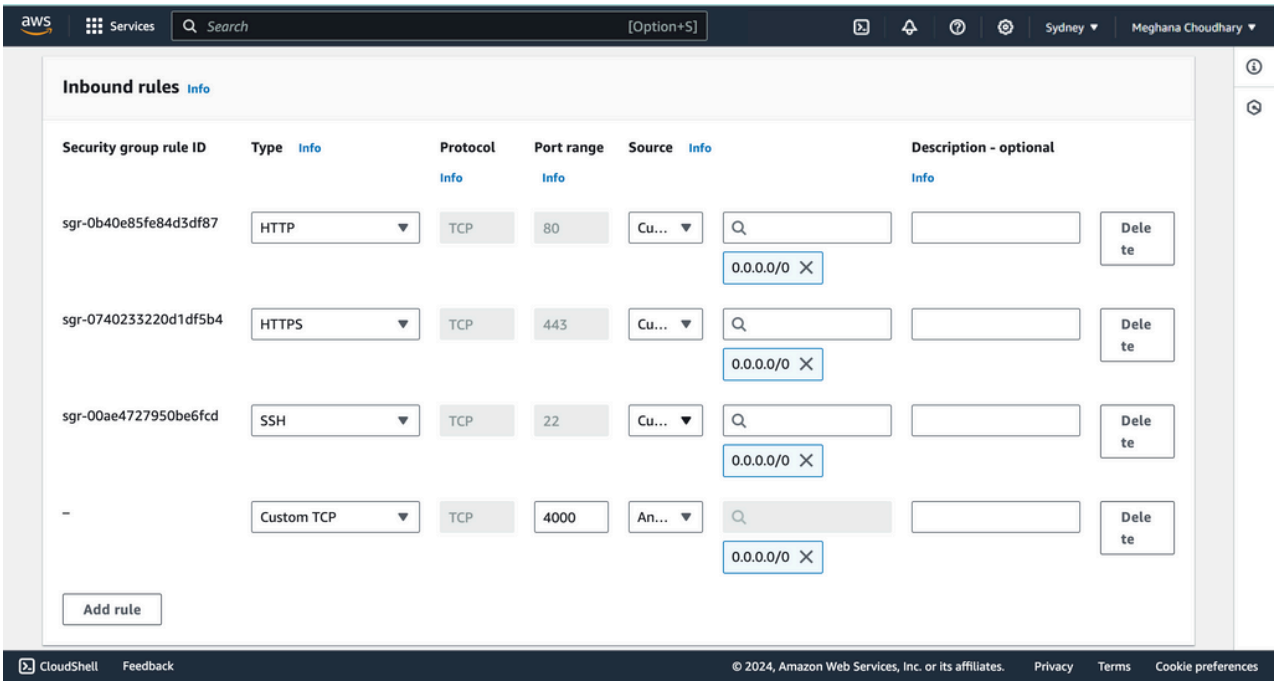
10. Scroll Down, Go to ‘Security’ then ‘Security groups’



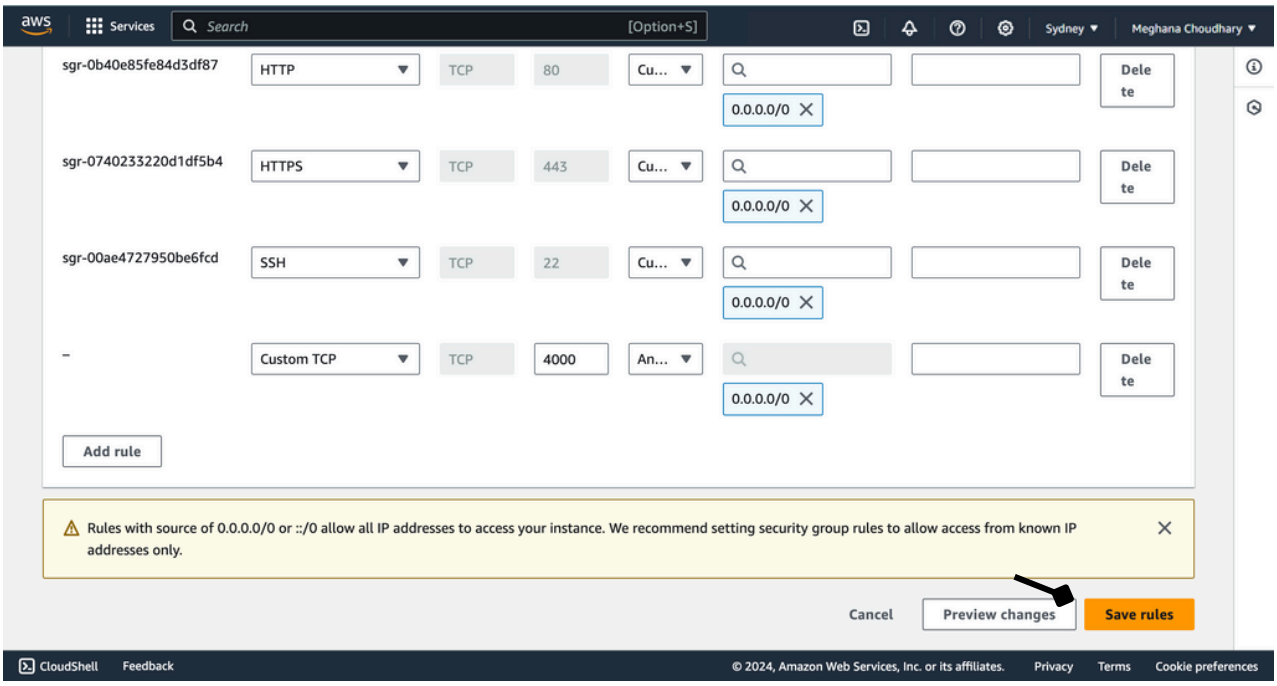
11. Now ‘Edit inbound rules’



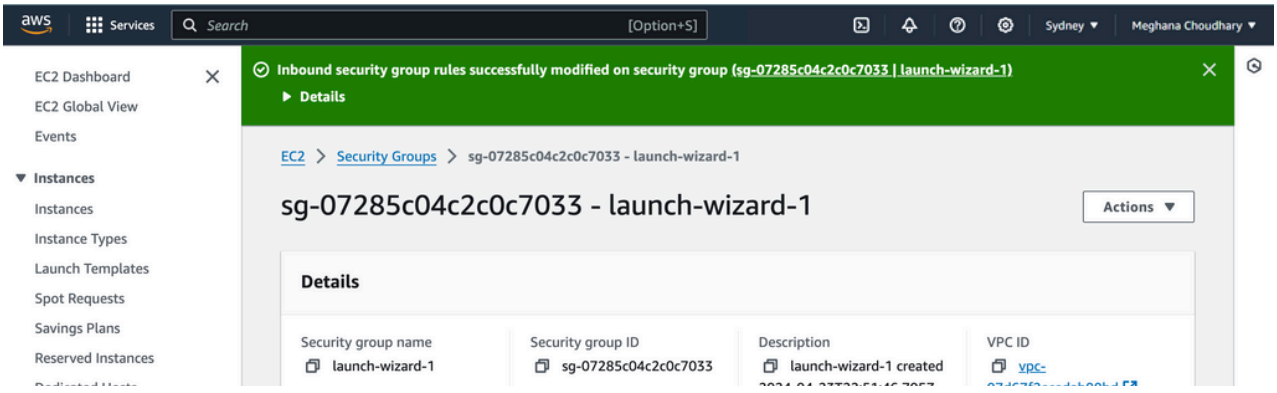
12. Add a Rule allowing traffic on port 4000 form anywhere.



13. Then 'Save rules'.

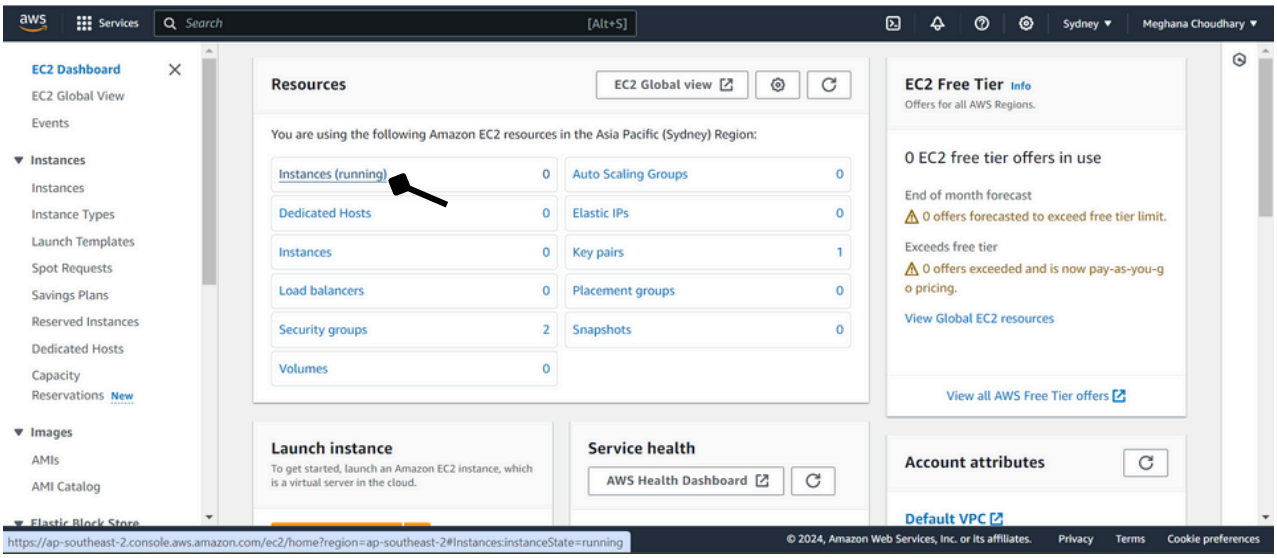


You'll Get a confirmation message.

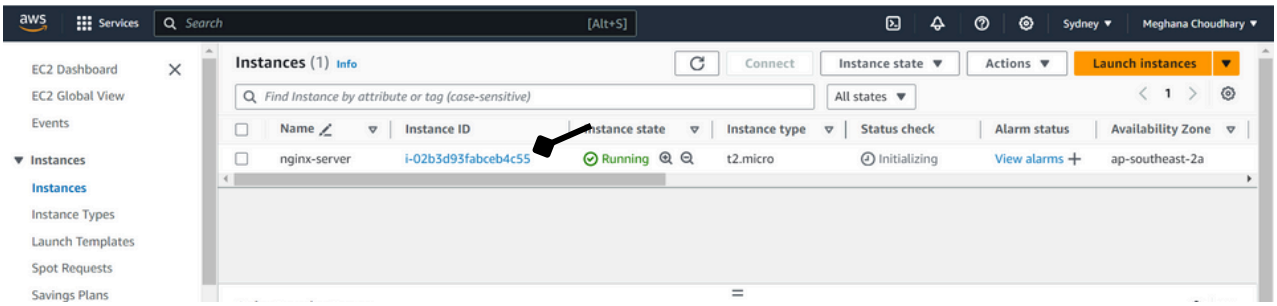


Connecting to the Instance with Bitvise SSH and deploying a project from Github:-

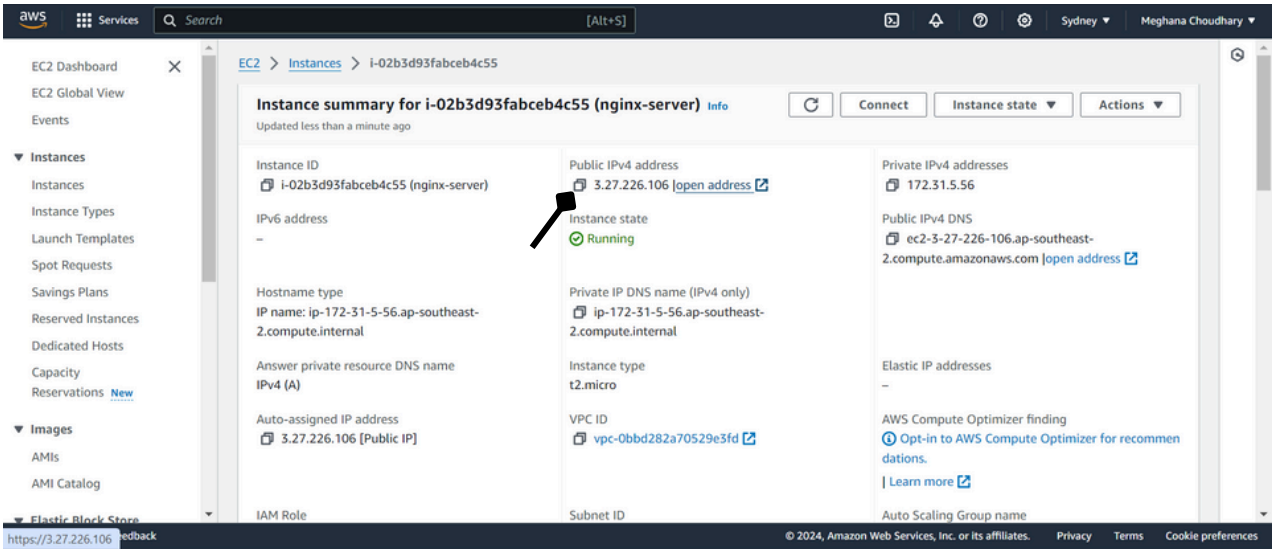
1. Log into AWS console, search for EC2 and open your EC2 dashboard and click on ‘Instances’.



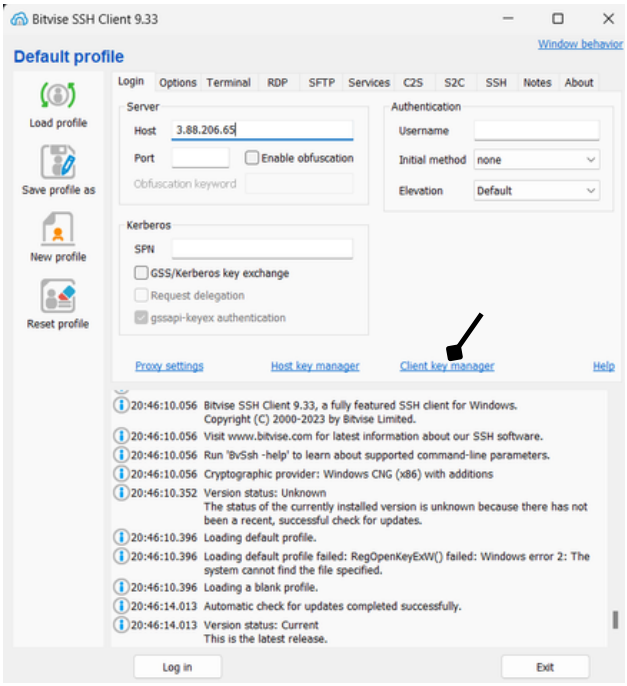
2. Click on the ‘Instance ID’ of your instance.



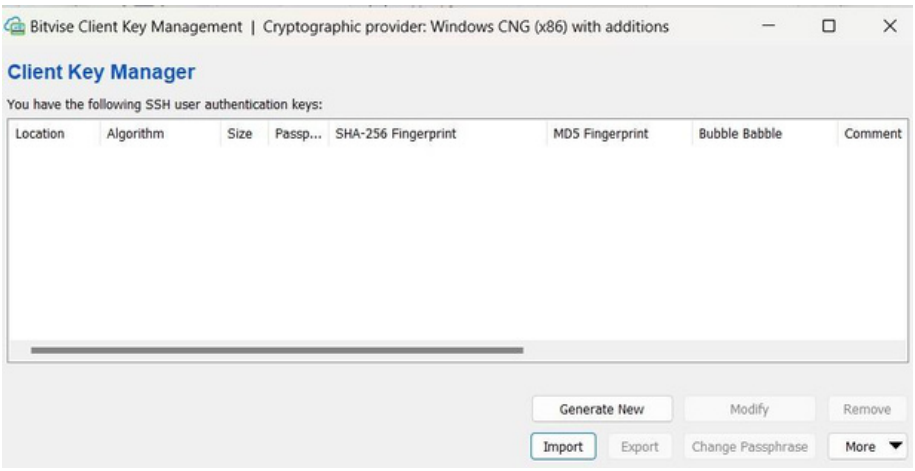
3. Copy the Public IPV4 address of your instance.

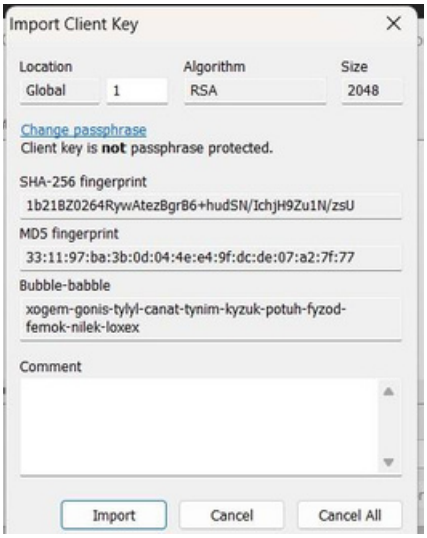


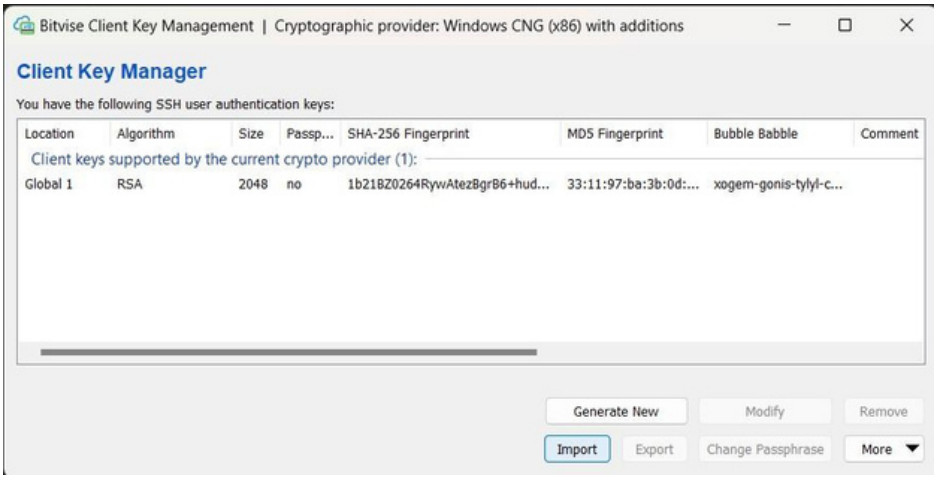
4. In Bitvise SSH Client, paste the ‘Public IPV4 address’ and click on ‘Client key Manager’.



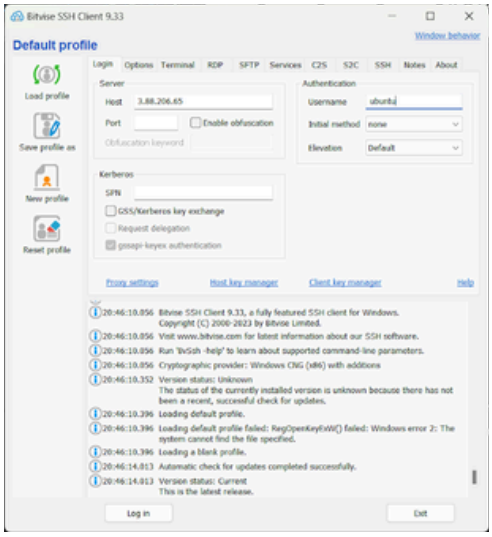
5. Under ‘Client key manager’, if there is any existing key then remove it and click on ‘Import’, select the same key that you used while creating the instance, from your file directory and then again click on ‘Import’ and then the key is successfully imported.

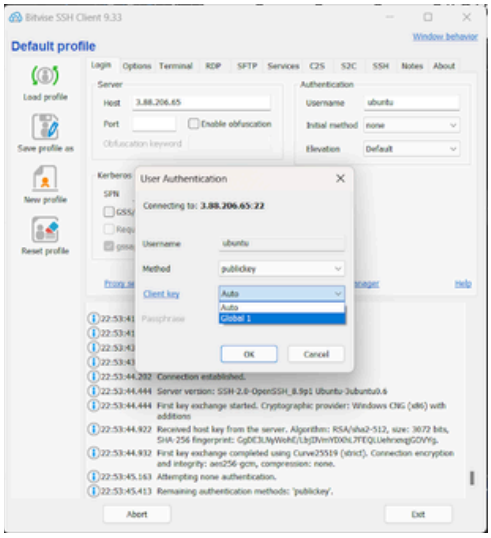


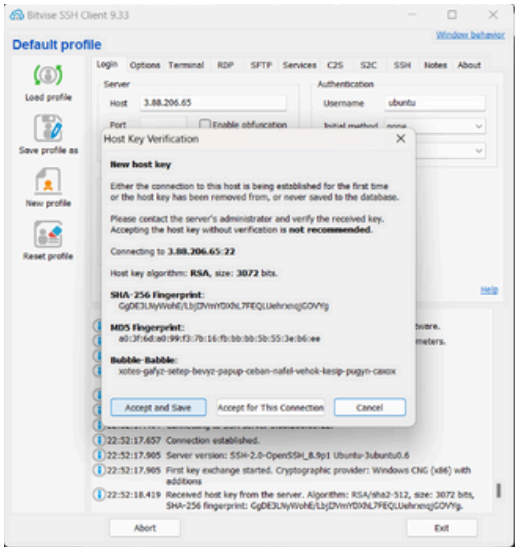




6. In ‘Bitvise SSH Client’, under ‘Default profile’ give the ‘Username’ as ‘ubuntu’ then select ‘Global1’ from ‘Client Key Manager’ then click on ‘Log in’ & ‘Accept and save’.







If the ‘Log In’ button changes to ‘Log Out’, then you are logged in successfully.

7. Click on ‘New terminal consol’, to access the terminal of your instance.

8. Once you are inside the teminal of your instance run

To update the system packages and upgrade them to the latest version.

```
sudo apt update -y && sudo apt upgrade -y
```

```
ubuntu@ip-172-31-10-86:~$ sudo apt update -y && sudo apt upgrade -y
```

Download Node.js from the source

```
curl -fsSL https://deb.nodesource.com/setup_current.x | sudo bash -
```

```
ubuntu@ip-172-31-2-183:~$ curl -fsSL https://deb.nodesource.com/setup_current.x | sudo bash -
```

Install Nginx, Git, Node.js with

```
sudo apt install git nginx nodejs -y
```

```
ubuntu@ip-172-31-2-183:~$ sudo apt install git nginx nodejs -y
```

Start and Enable the Nginx Service

```
sudo systemctl start nginx && sudo systemctl enable nginx
```

```
ubuntu@ip-172-31-10-86:~$ sudo systemctl start nginx && sudo systemctl enable nginx
```

Clone the Github Repository

```
git clone https://github.com/meghana-choudhary/aws_proj.git
```

```
ubuntu@ip-172-31-4-9:~$ git clone https://github.com/meghana-choudhary/aws_proj.git
```

Go to the repository folder.

```
cd aws_proj
```

```
ubuntu@ip-172-31-4-9:~$ cd aws_proj
```

Install Dependencies described in the package.json

```
npm install
```

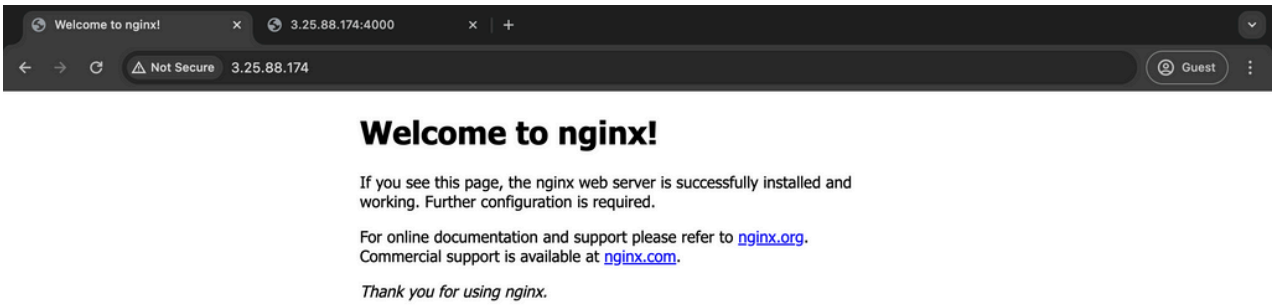
```
ubuntu@ip-172-31-4-9:~/aws_proj$ npm install
```

Start the Node server

```
node index.js
```

```
ubuntu@ip-172-31-4-9:~/aws_proj$ node index.js
```

9. Now pasting the Public IPV4 address of your EC2 instance in a web browser gets us to the Nginx default welcome page, through the default HTTP port, port 80.



Adding :4000 after the IPV4 address, we get to our Node.js server through our specified port 4000.