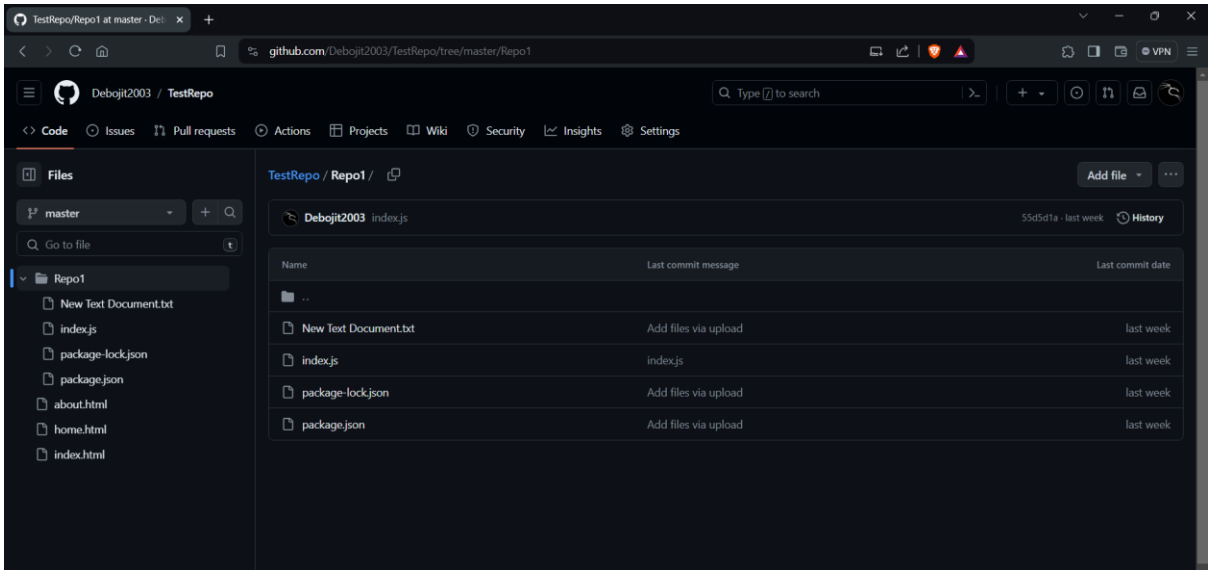
ASSIGNMENT No: 9

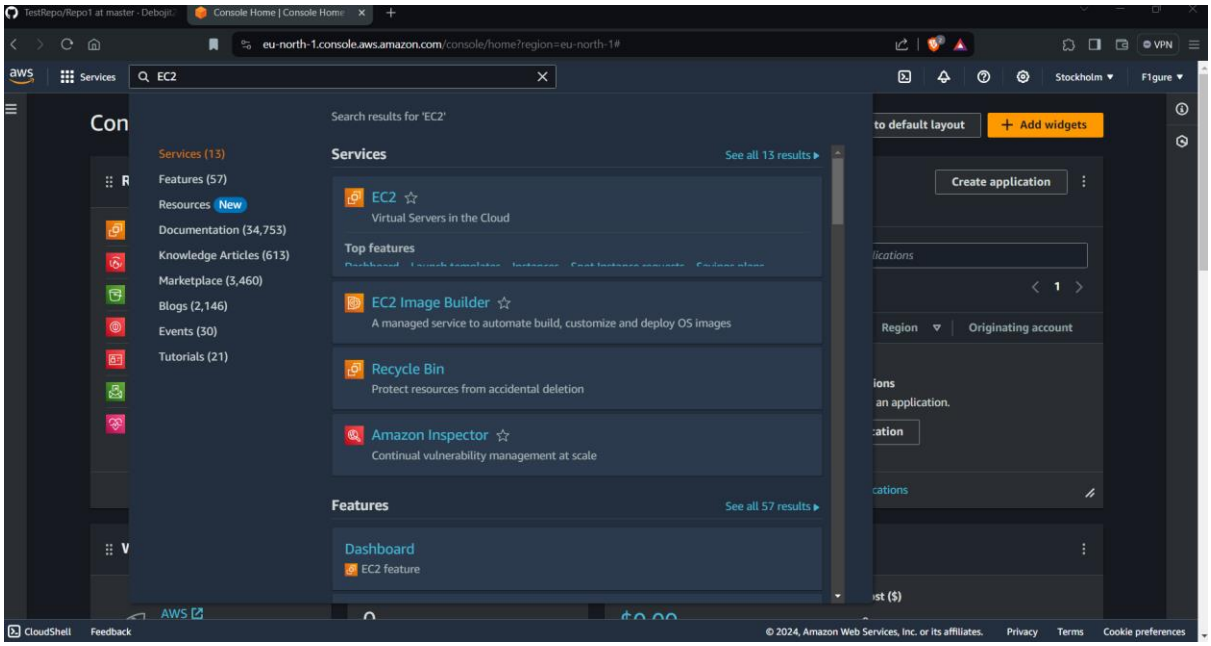
Problem Statement: Deploy a project from GitHub to EC2.

The steps are as follows: -

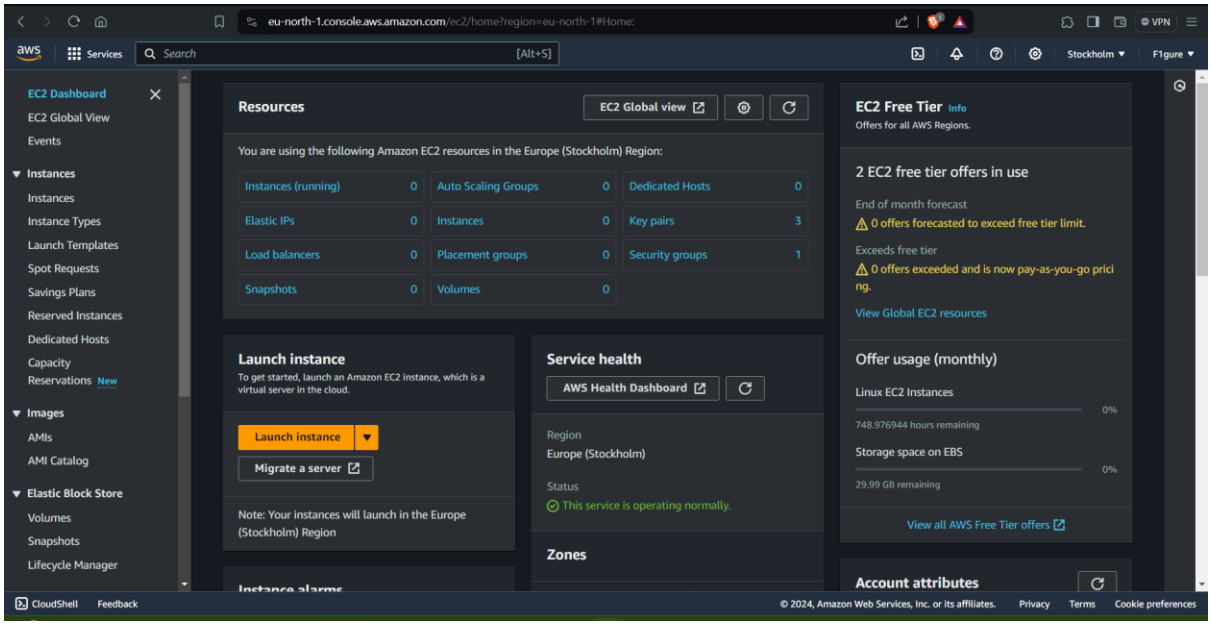
1. Please confirm whether the `**index.js**` file exists in your GitHub repository. If it's not present, then clone it into a repository if the file already exists on your local machine using the command `**git clone <repository_path>**`.



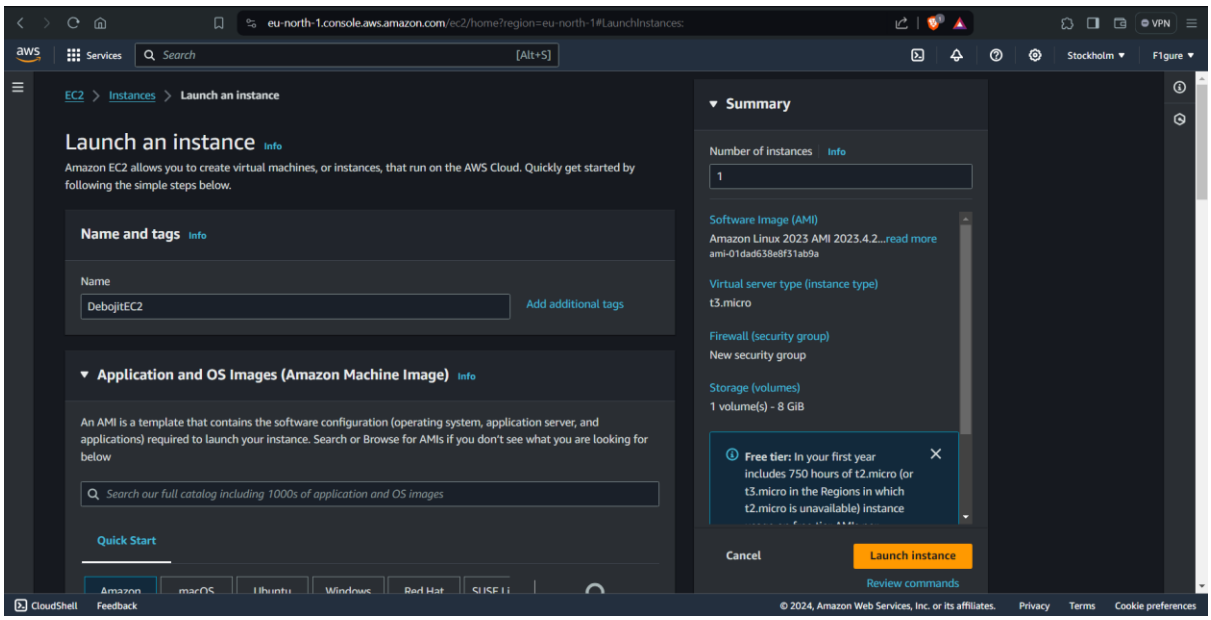
2. Access your AWS account and utilize the search box to look for EC2, then proceed to click on the first option displayed.



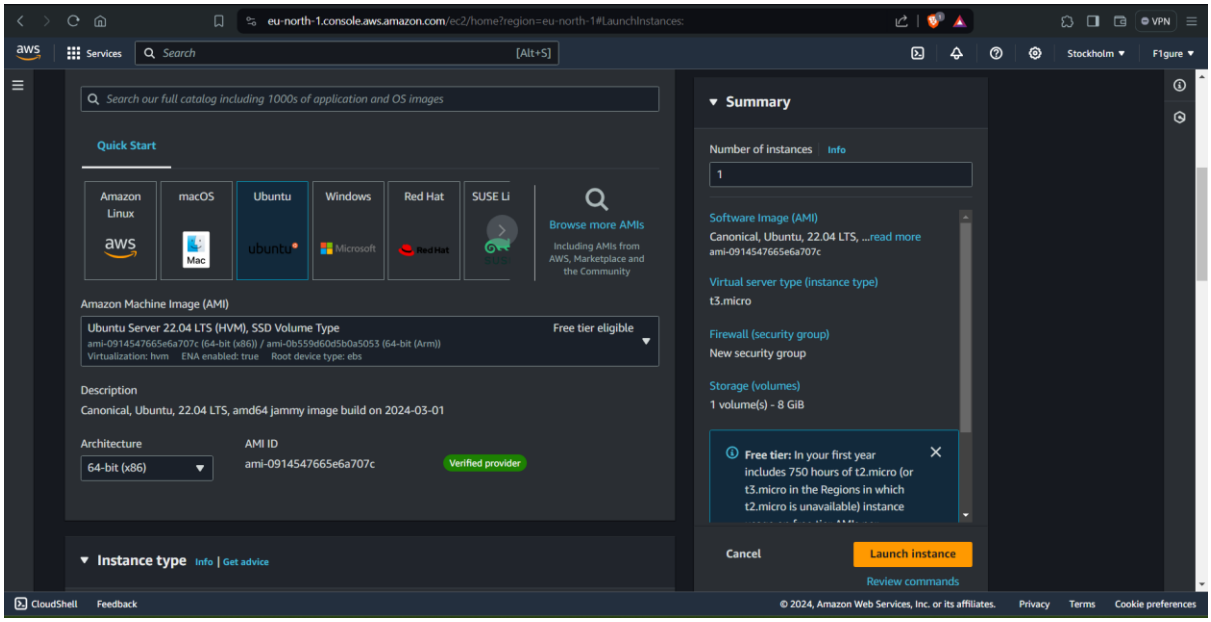
3. Subsequently, select "**Launch Instance**".



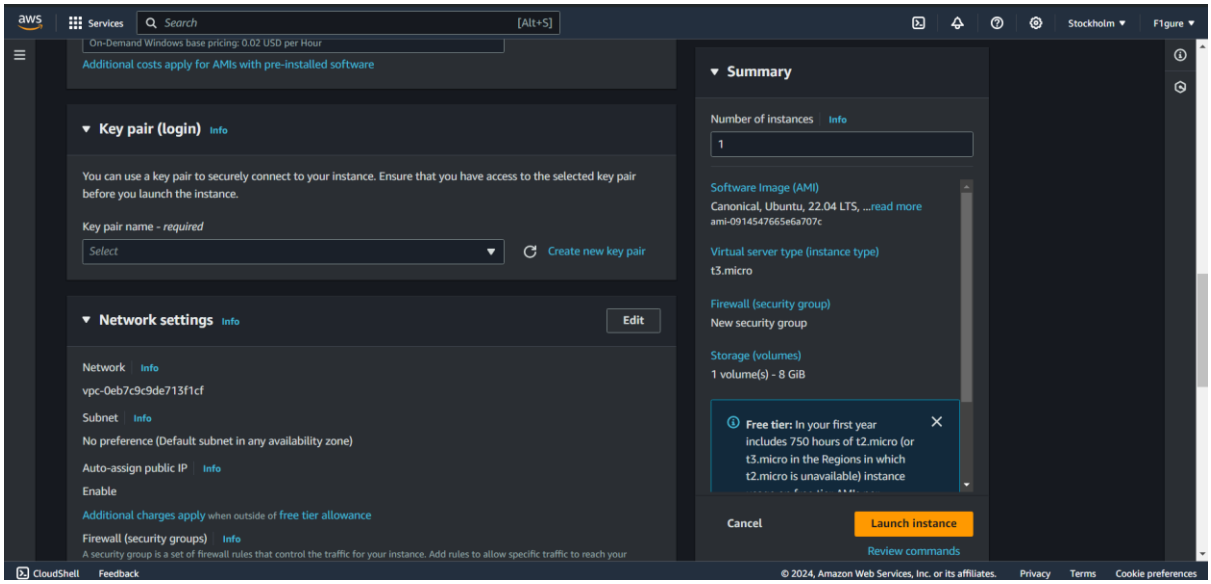
4. For the subsequent step, ensure to assign an appropriate name to the EC2 instance. (For instance, here it is : DebojitEC2)



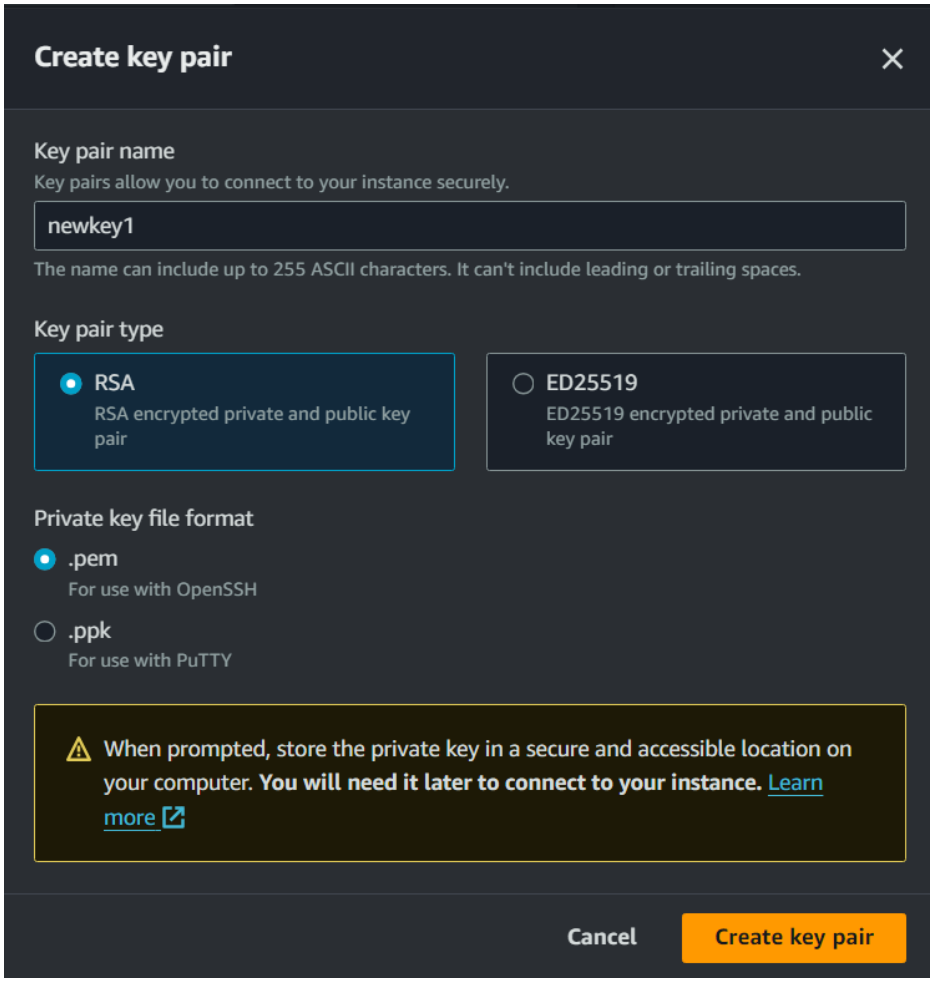
5. Choose **Ubuntu** from the available AMIs.



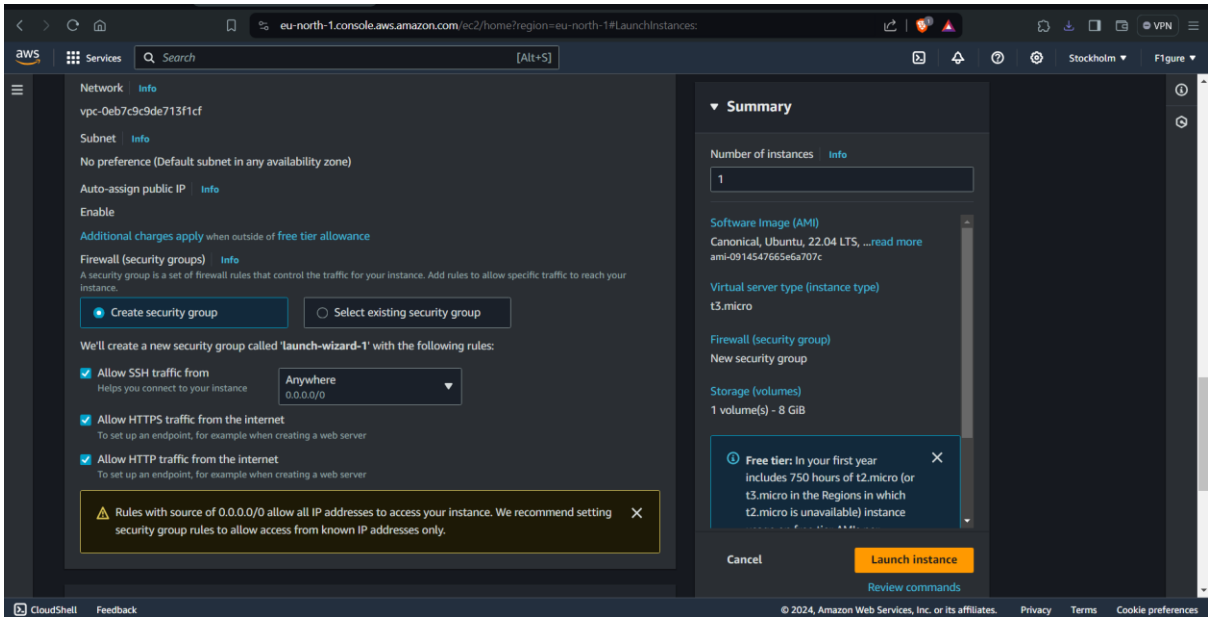
6. Generate a new key pair or alternatively, utilize an existing one. A new key pair is generated by selecting "**Create new key pair**".



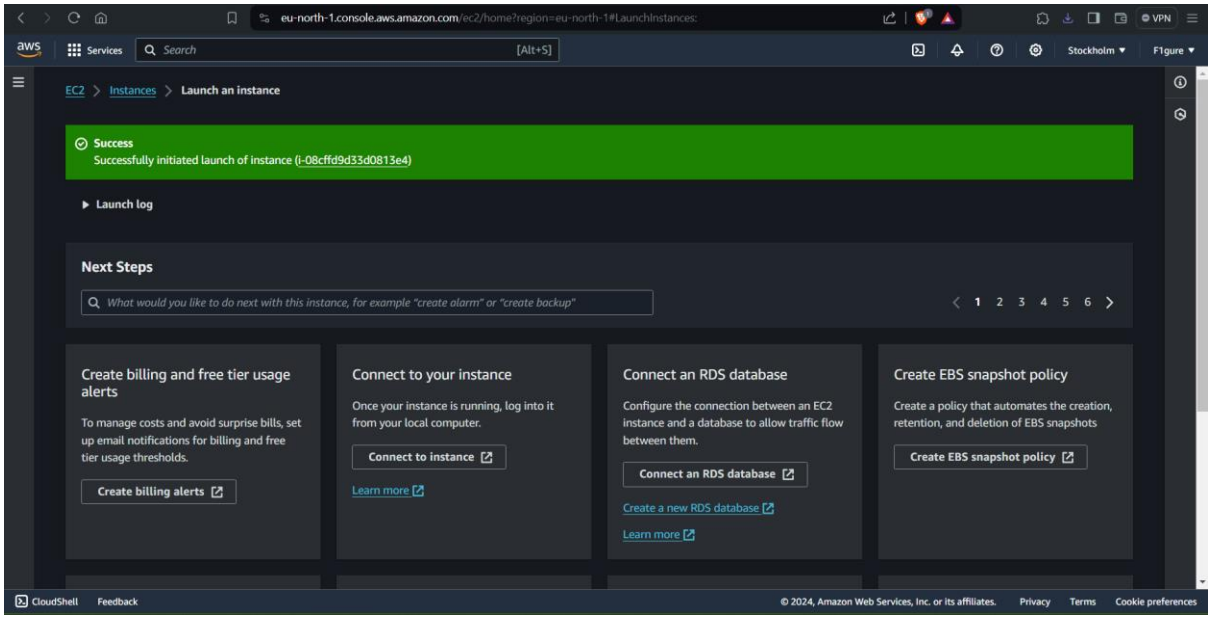
7. In the following window, provide a name for the key pair(here it is- newkey1) and opt for the **.pem** format. Then click on "**create key pair**".



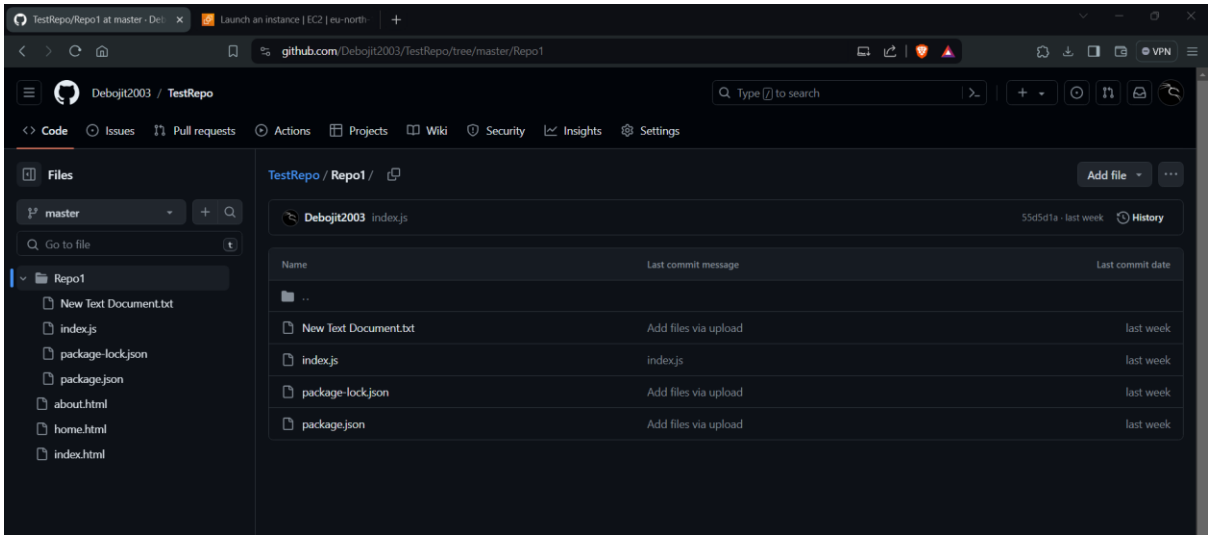
8. Select to create a server and ensure to enable all three protocols: SSH, HTTP, and HTTPS.



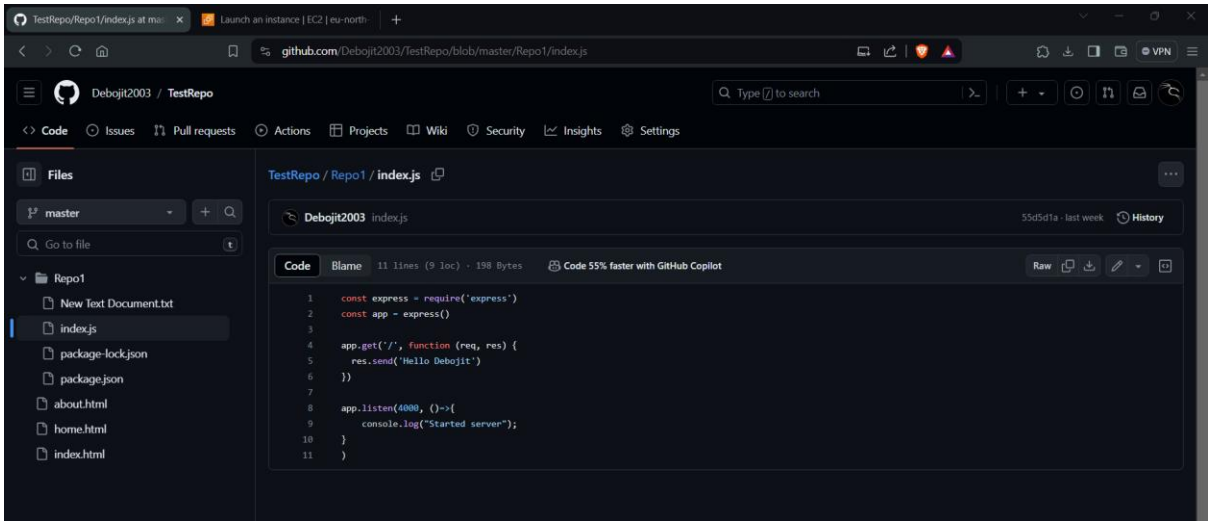
9. Finally, proceed by clicking on "**Launch Instance**".



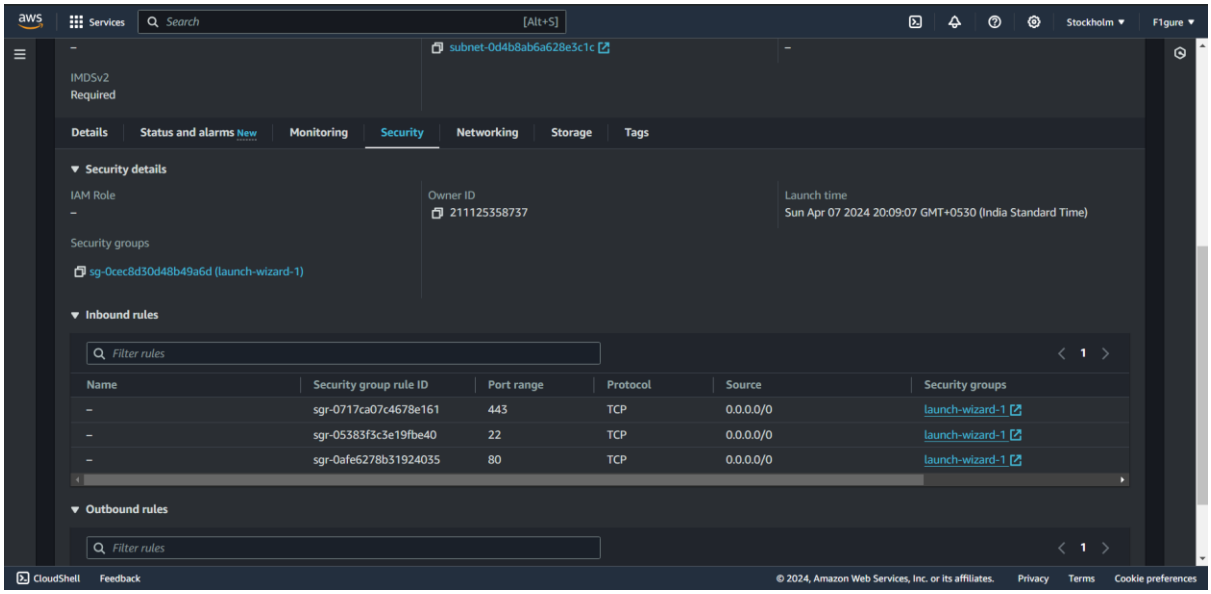
10. Access GitHub and navigate to your repository that is currently being worked on, containing the `**index.js**` file.



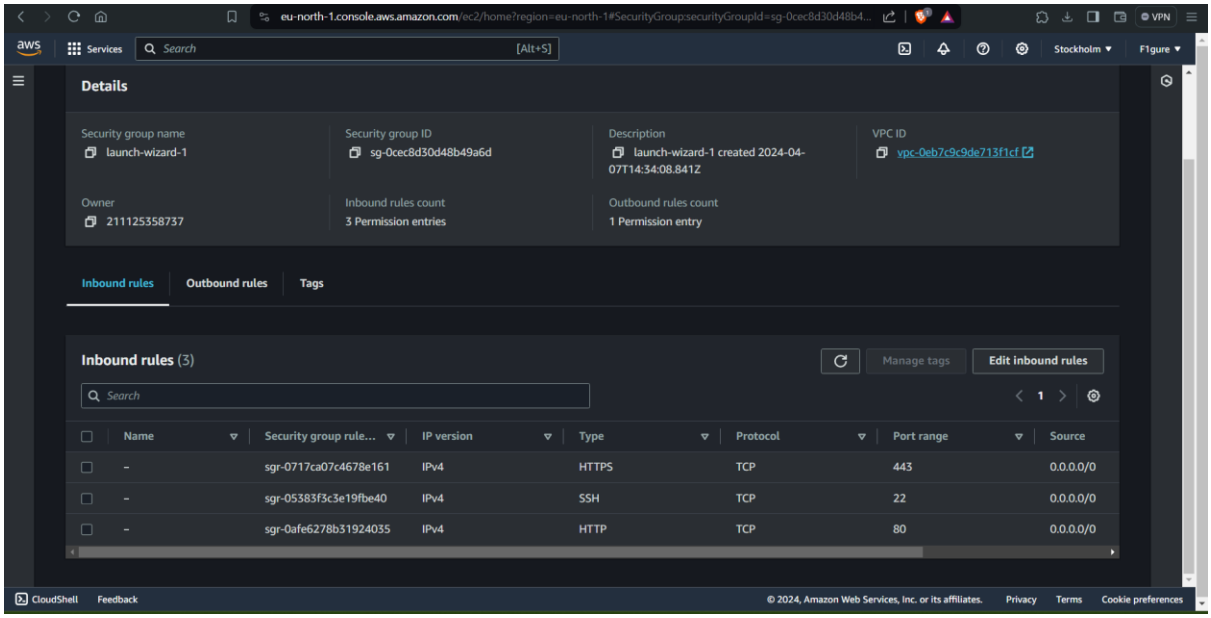
11. Access the "**index.js**" file and modify the code by replacing "**hello students**" with "**hello <your_name>".**



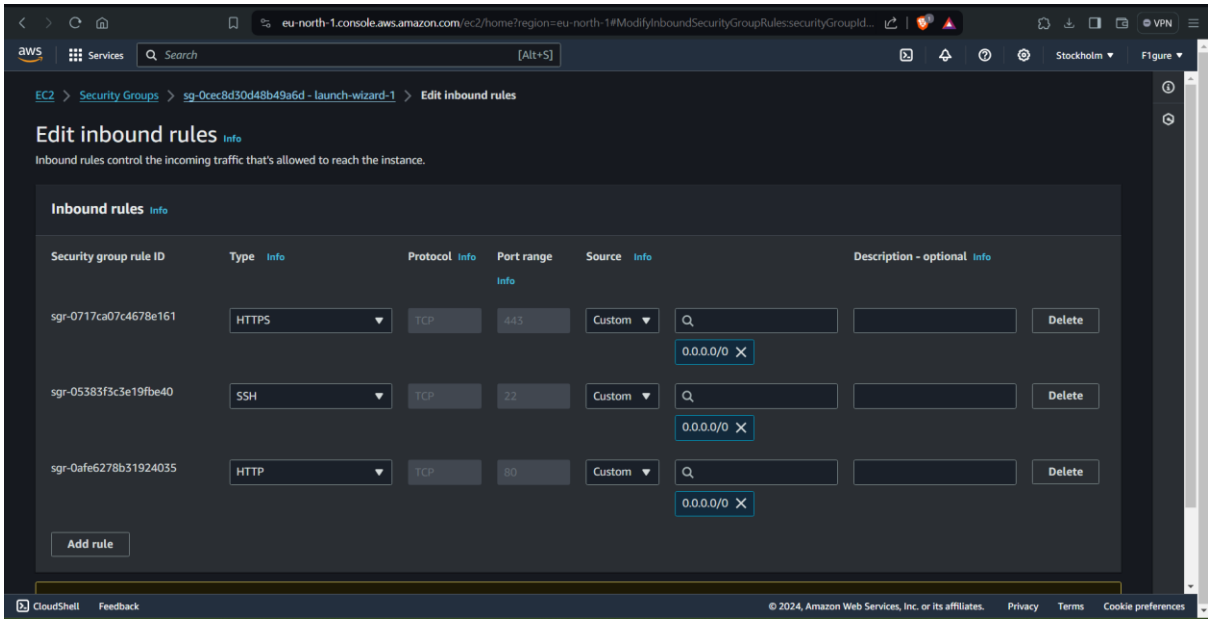
12. Return to the AWS tab currently open in your system, then navigate to **instances** and select **security.**



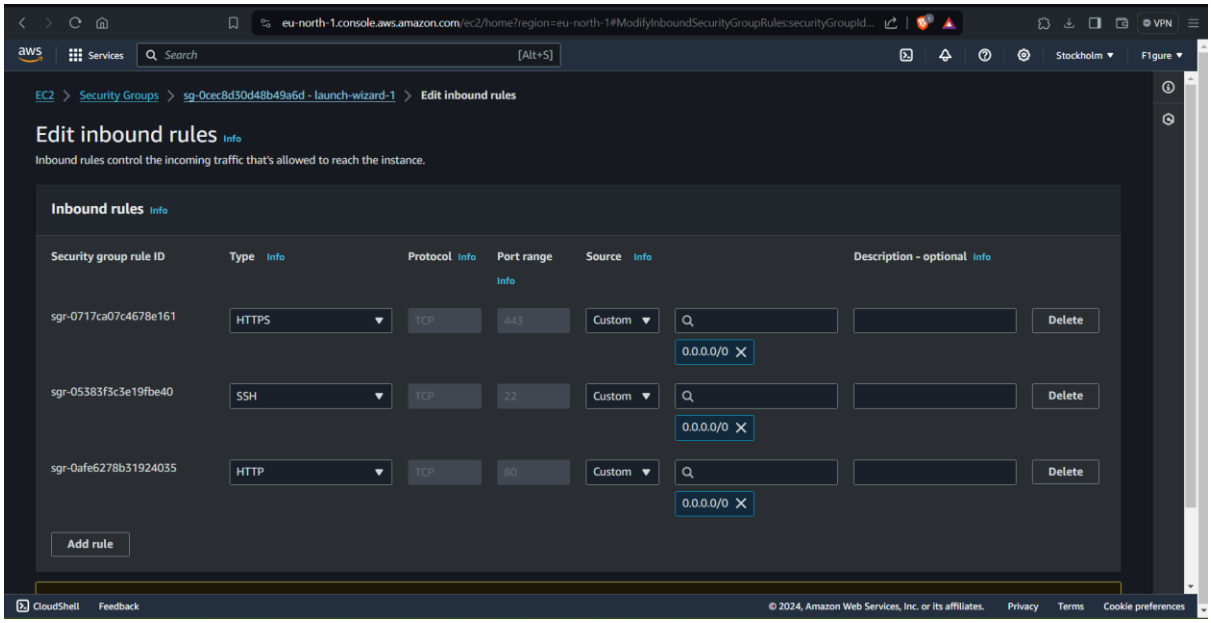
13. Select the security group and observe that three protocols have been chosen.



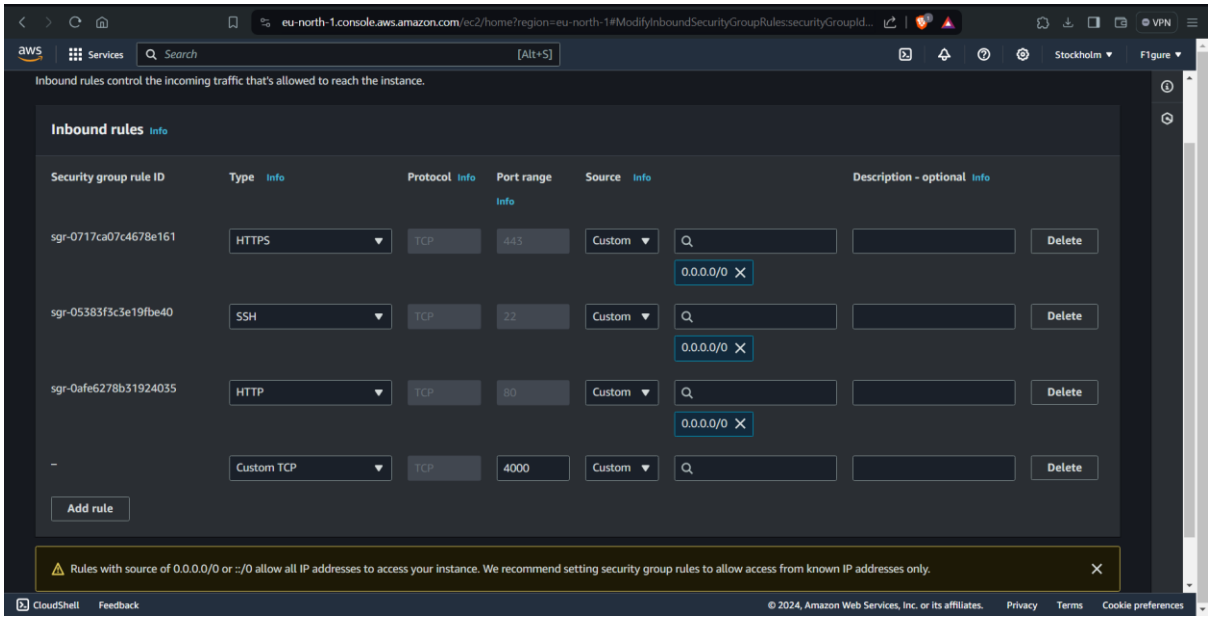
14. Select "**Edit inbound rules.**"



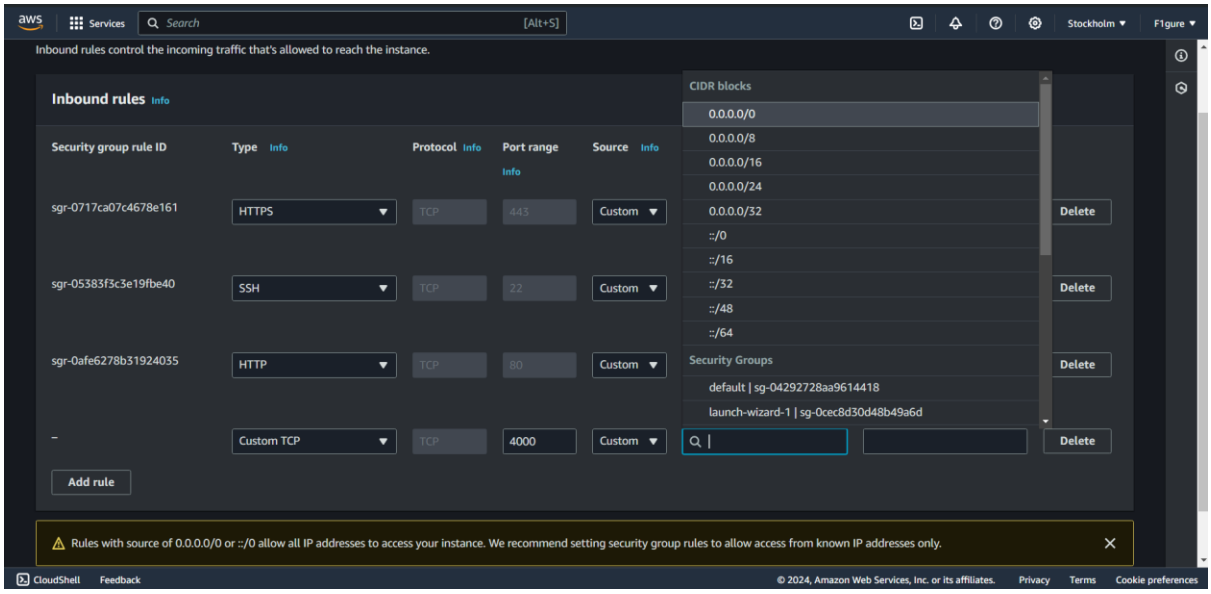
15. Scroll down and select "**Add rule**" to add Custom protocol.



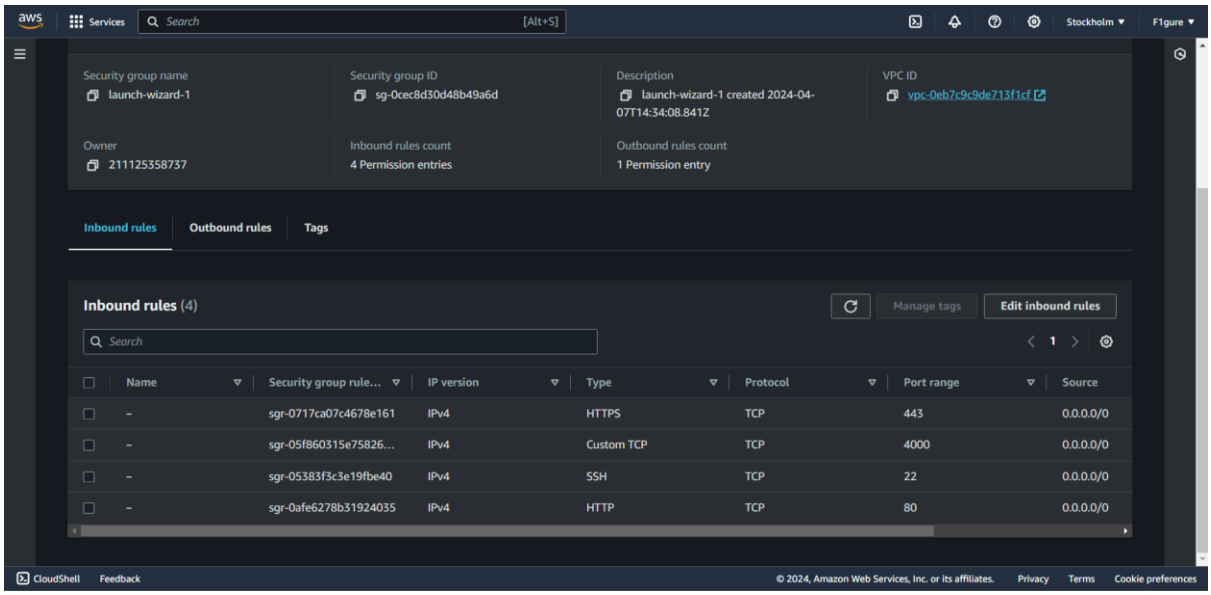
16. Enter 4000 in the port number field, as specified in the `**index.js**` code.



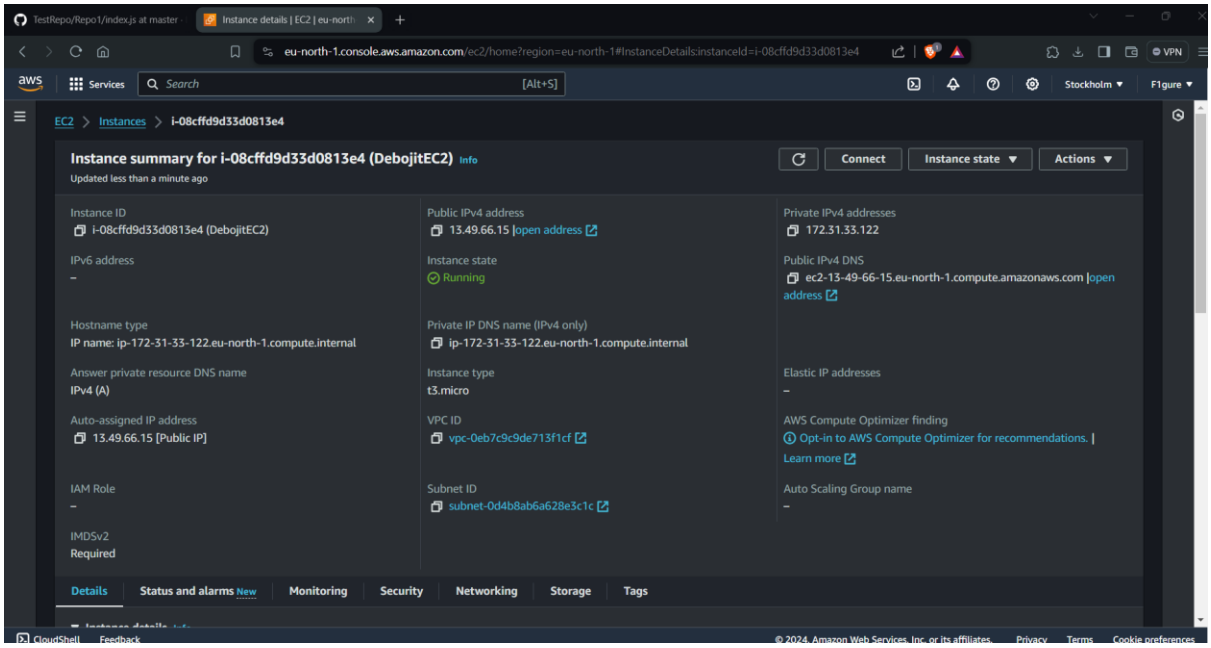
17. Choose the first option available beside it.



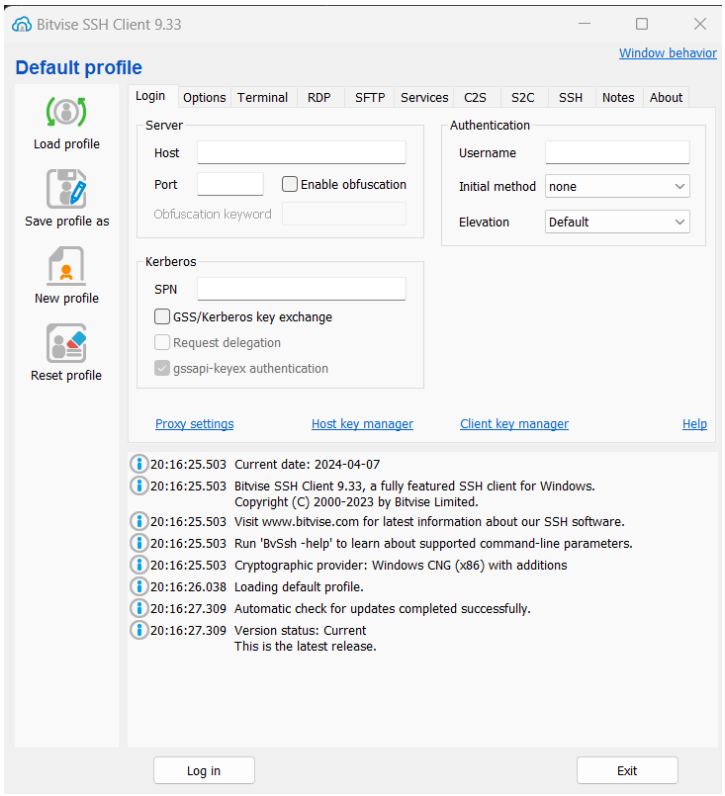
18.Finally click on **Save rules**. Then you will see the following window appears.



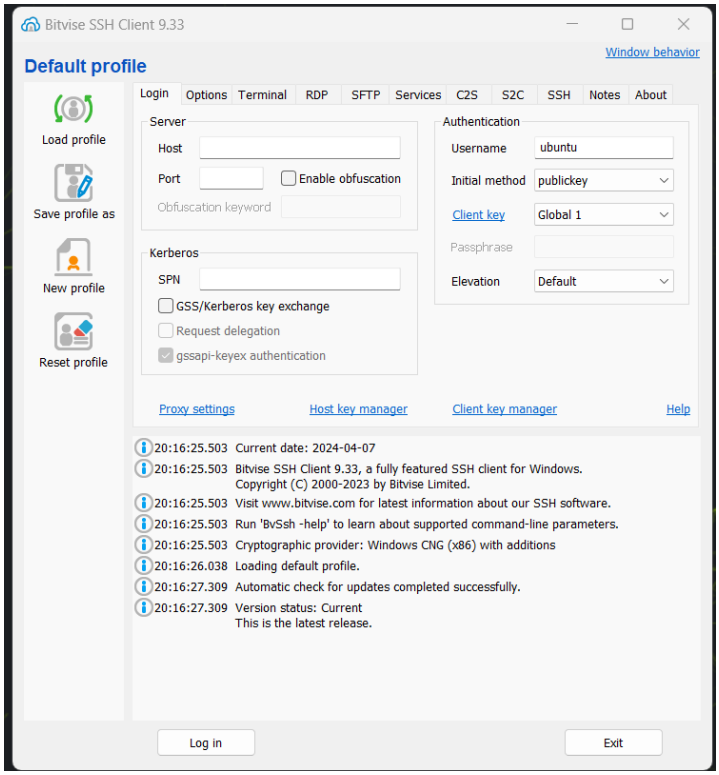
19. Navigate to instances and copy the Public IPv4 address.



20. Launch Bitvise SSH Client and proceed to click on "**Login**".

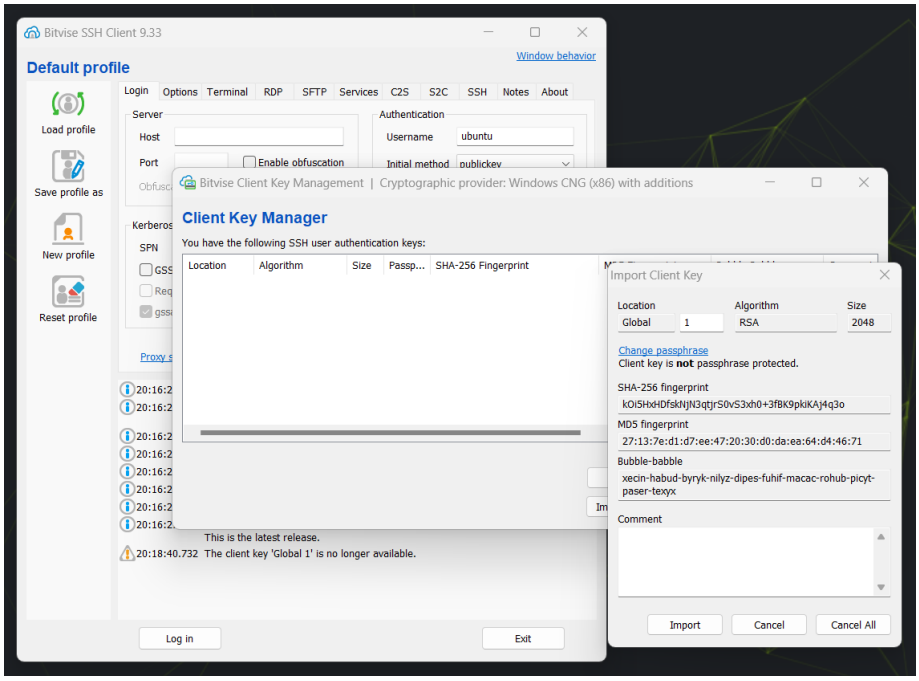


21.Paste the IPv4 address into the host field and enter "ubuntu" as the username.

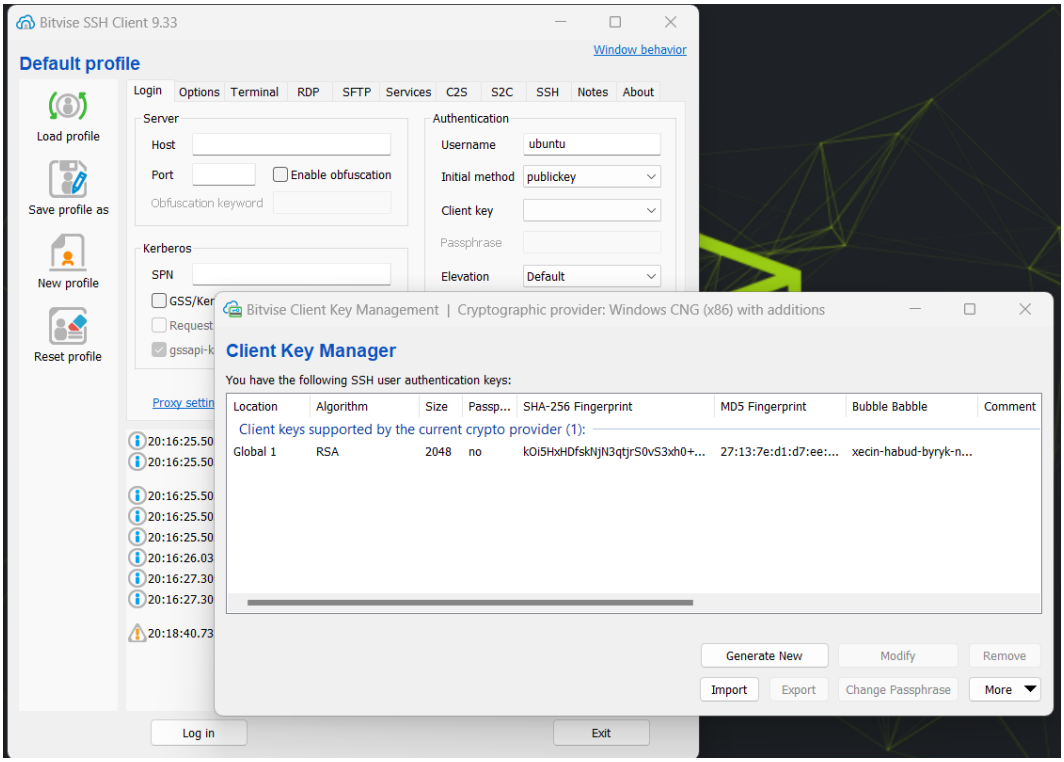


22. Navigate to the client key manager to verify if the correct key is being used. If not, remove the existing key and import the correct one. The steps are: -

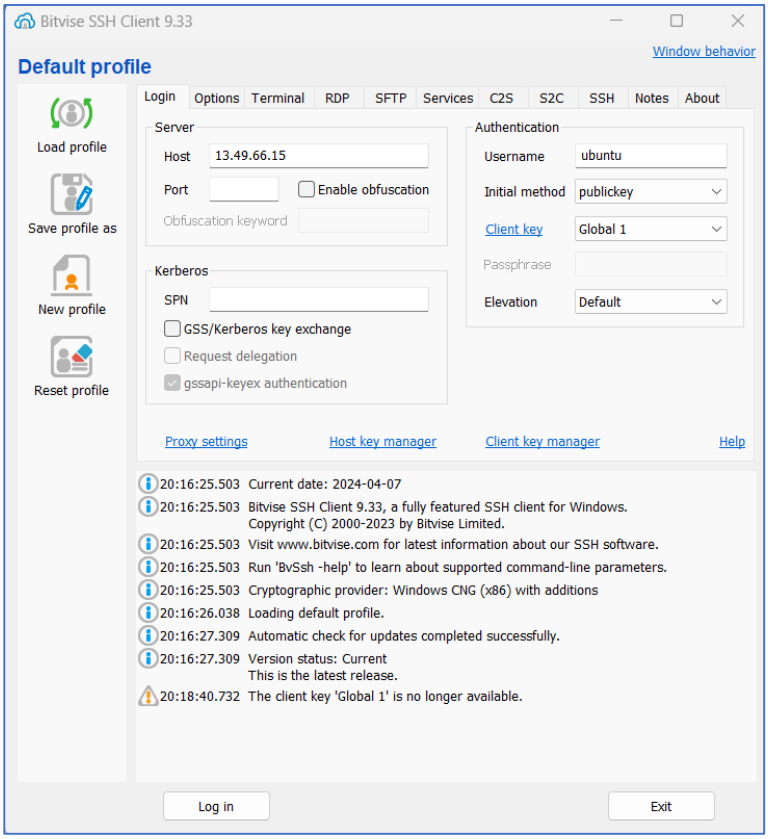
i)

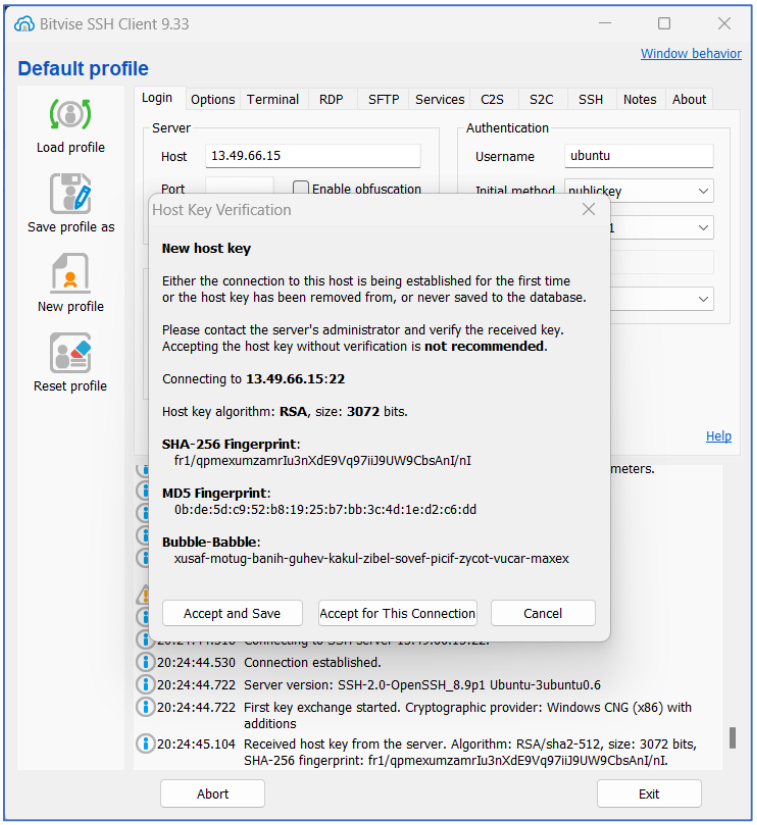


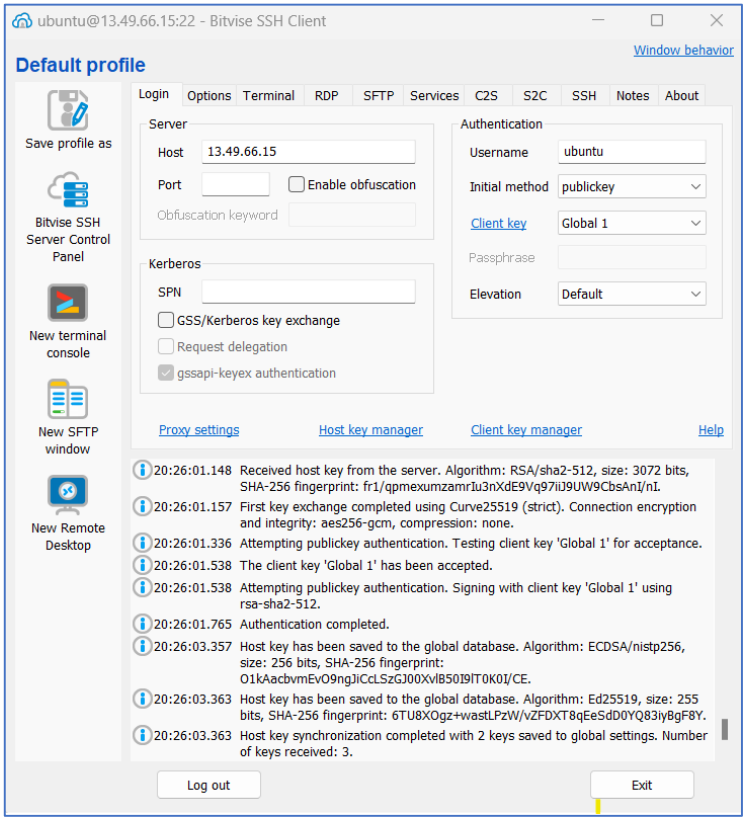
ii)



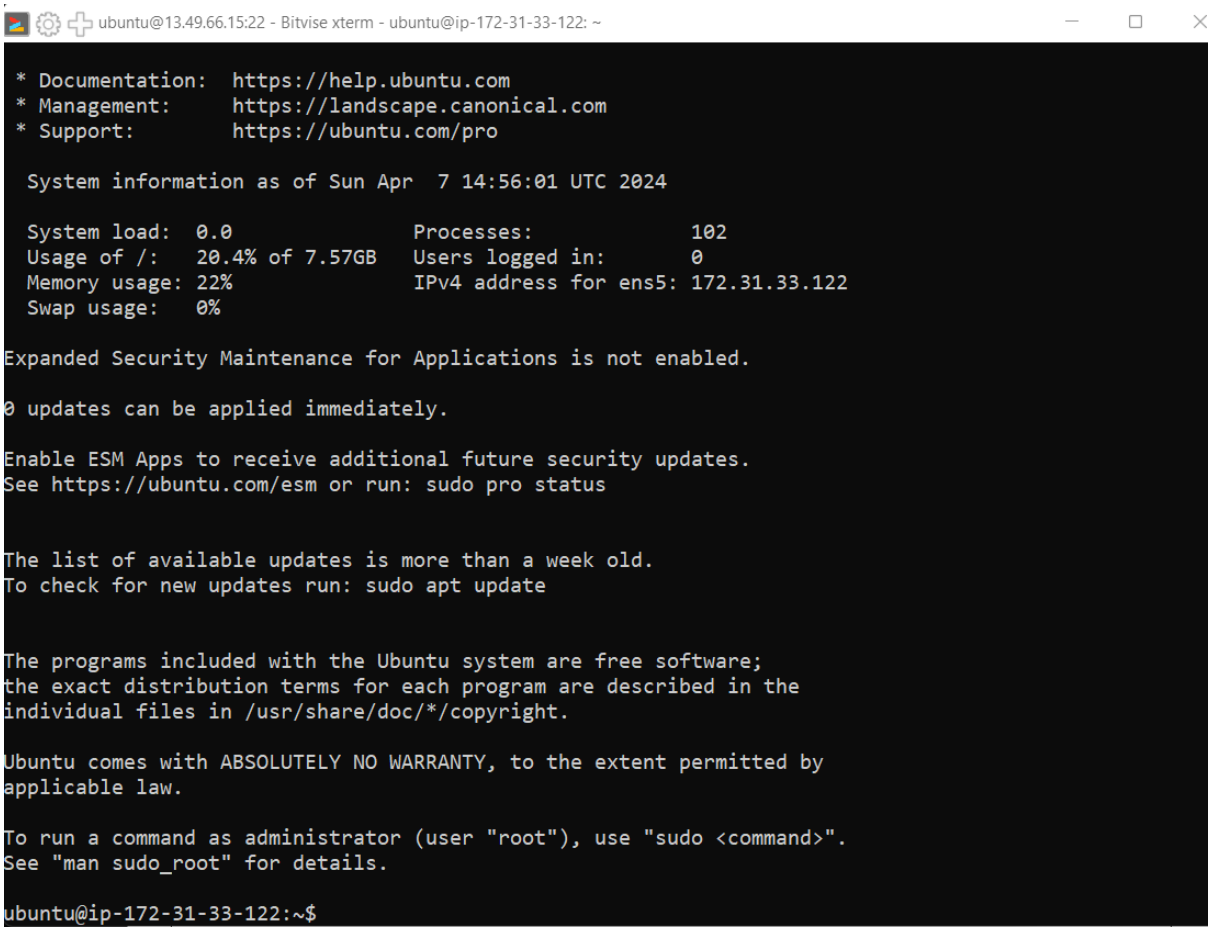
23. Enter the HOST as the public IP of your instance then click on Log in. After logging in, click on "**Accept and Save**". Subsequently, the following window will appear.





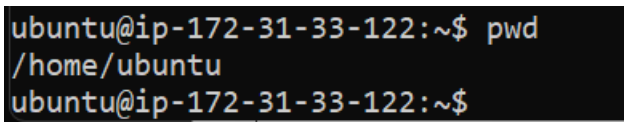


24. Access the terminal console from the provided left pane.



25. Proceed by entering the commands as listed below.

->***pwd***
(When executed in a terminal, it displays the current directory or the full path of the current working directory in the file system.)



->***sudo apt-get update***
(The `sudo apt-get update` command is used on Ubuntu and other Debian-based systems to update the local package index.)

```
ubuntu@ip-172-31-33-122:~$ sudo apt-get update
Hit:1 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu jammy InRelease
Get:2 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu jammy-updates InRelease [119 kB]
Get:3 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu jammy-backports InRelease [109 kB]
Get:4 http://security.ubuntu.com/ubuntu jammy-security InRelease [110 kB]
Get:5 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu jammy/universe amd64 Packages [14.1 MB]
Get:6 http://security.ubuntu.com/ubuntu jammy-security/main amd64 Packages [1303 kB]
Get:7 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu jammy/universe Translation-en [5652 kB]
Get:8 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu jammy/universe amd64 c-n-f Metadata [286 kB]
Get:9 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu jammy/multiverse amd64 Packages [217 kB]
Get:10 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu jammy/multiverse Translation-en [112 kB]
Get:11 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu jammy/multiverse amd64 c-n-f Metadata [8372 B
]
Get:12 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 Packages [1519 kB]
Get:13 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main Translation-en [293 kB]
Get:14 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/restricted amd64 Packages [1648
 kB]
Get:15 http://security.ubuntu.com/ubuntu jammy-security/main Translation-en [233 kB]
Get:16 http://security.ubuntu.com/ubuntu jammy-security/restricted amd64 Packages [1616 kB]
Get:17 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/restricted Translation-en [275
```

->***sudo apt-get upgrade***

(The `sudo apt-get upgrade` command, when executed on Ubuntu or other Debian-based systems, upgrades all installed packages to their latest available versions)

```
ubuntu@ip-172-31-33-122:~$ sudo apt-get upgrade
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Calculating upgrade... Done
The following packages have been kept back:
  linux-aws linux-headers-aws linux-image-aws ubuntu-advantage-tools ubuntu-pro-client-l10n
The following packages will be upgraded:
  apt apt-utils bash bsdextrautils bsdutils coreutils curl dpkg eject ethtool fdisk libapt-pkg6.0
  libblkid1 libcurl3-gnutls libcurl4 libexpat1 libfdisk1 libgpgme11 libldap-2.5-0 libldap-common
  libmount1 libsmartcols1 libuuid1 mount python3-cryptography python3-update-manager snapd
  update-manager-core update-notifier-common util-linux uuid-runtime vim vim-common vim-runtime
  vim-tiny xxd
36 upgraded, 0 newly installed, 0 to remove and 5 not upgraded.
```

->***sudo apt-get install nginx***

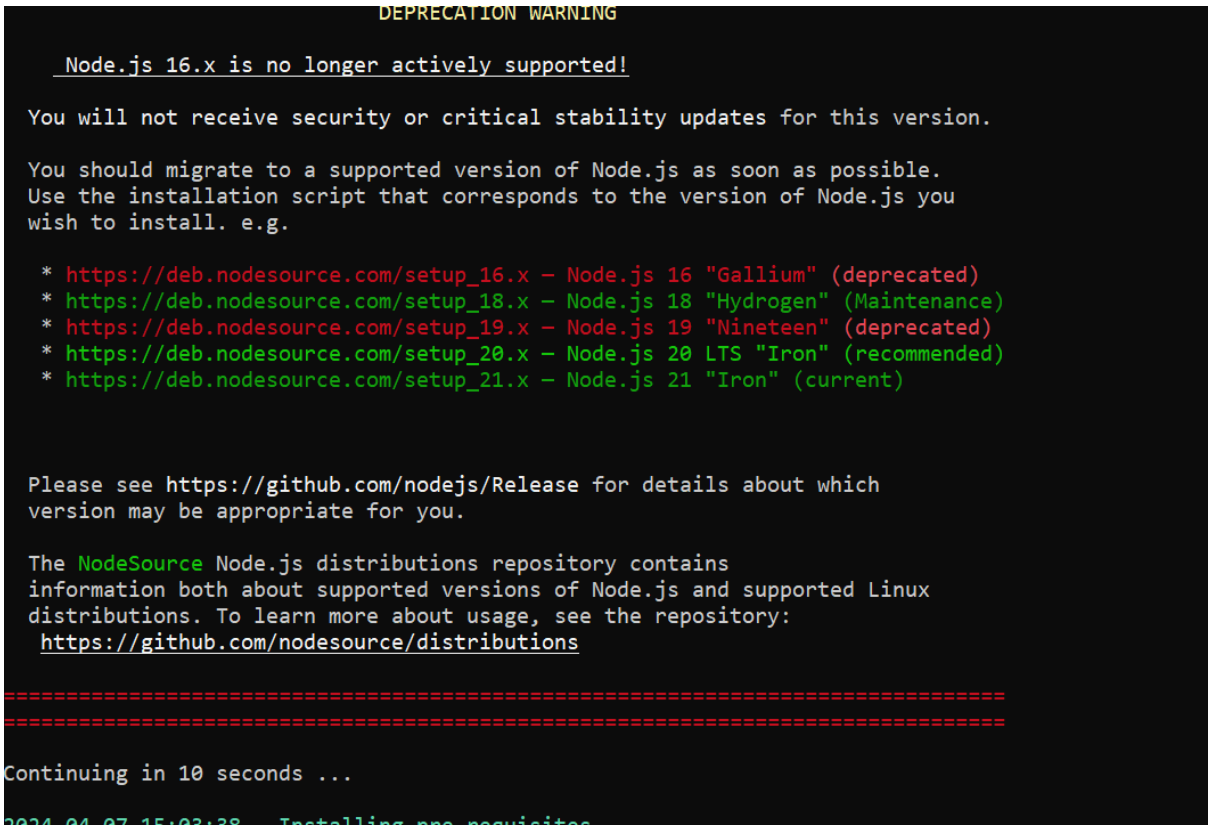
(The `sudo apt-get install nginx` command, when executed on Ubuntu or other Debian-based systems, installs the Nginx web server software.)

```
ubuntu@13.49.66.15:22 - Bitvise xterm - ubuntu@ip-172-31-33-122: ~
ubuntu@ip-172-31-33-122:~$ sudo apt-get install nginx
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  fontconfig-config fonts-dejavu-core libdeflate0 libfontconfig1 libgd3 libjbig0 libjpeg-turbo8
  libjpeg8 libnginx-mod-http-geoip2 libnginx-mod-http-image-filter libnginx-mod-http-xslt-filter
  libnginx-mod-mail libnginx-mod-stream libnginx-mod-stream-geoip2 libtiff5 libwebp7 libxpm4
  nginx-common nginx-core
```

->***curl -SL https://deb.nodesource.com/setup_16.x|sudo -E bash***

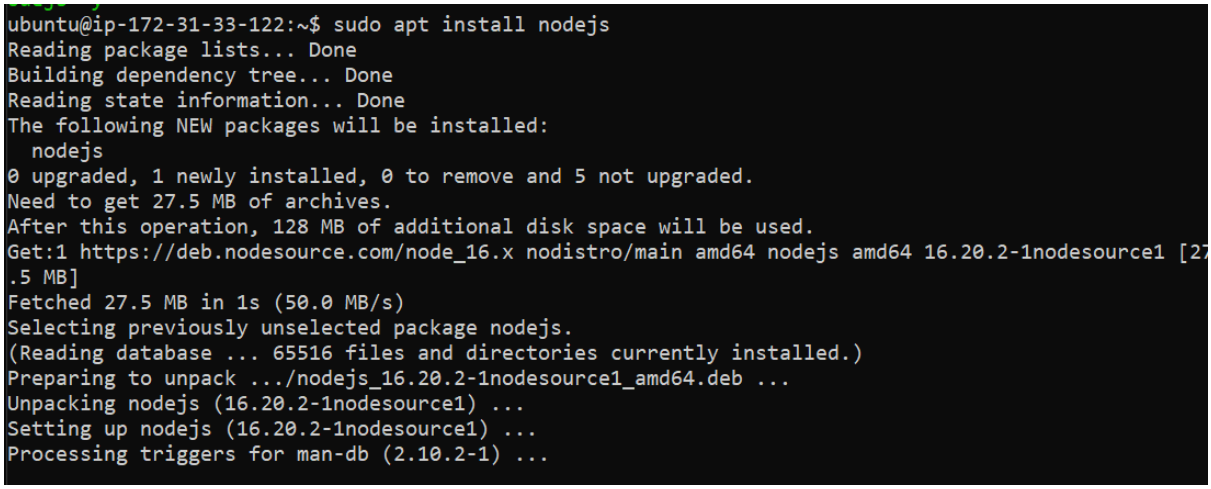
```
ubuntu@ip-172-31-33-122:~$ curl -SL https://deb.nodesource.com/setup_16.x|sudo -E bash
```

(After execution, your system will be set up to install Node.js version 16.x from the Node Source repository.)



->***sudo apt install nodejs***

(The `sudo apt install NodeJS command, when executed on Ubuntu or other Debian-based systems, installs the Node.js runtime environment along with the npm package manager.)



->***node --version***

(Returns the version of nodejs installed.)

```
ubuntu@ip-172-31-33-122:~$ node --version
v16.20.2
```

->***git clone <your repository path>***

(The `git clone` command is used to create a copy of a Git repository on your local machine.)

```
ubuntu@ip-172-31-33-122:~$ git clone https://github.com/Debojit2003/TestRepo.git
Cloning into 'TestRepo'...
remote: Enumerating objects: 16, done.
remote: Counting objects: 100% (16/16), done.
remote: Compressing objects: 100% (14/14), done.
remote: Total 16 (delta 5), reused 4 (delta 2), pack-reused 0
Receiving objects: 100% (16/16), 50.48 KiB | 1.17 MiB/s, done.
Resolving deltas: 100% (5/5), done.
ubuntu@ip-172-31-33-122:~$
```

>Now go inside the repository using cd and ls.

```
ubuntu@ip-172-31-33-122:~$ ls
TestRepo
ubuntu@ip-172-31-33-122:~$ cd TestRepo
ubuntu@ip-172-31-33-122:~/TestRepo$ ls
Repo1  about.html  home.html  index.html
ubuntu@ip-172-31-33-122:~/TestRepo$ cd Repo1
ubuntu@ip-172-31-33-122:~/TestRepo/Repo1$ ls
'New Text Document.txt'   index.js   package-lock.json   package.json
ubuntu@ip-172-31-33-122:~/TestRepo/Repo1$
```

->***npm install***

(The `npm install` command is used in a Node.js project to install dependencies listed in the `package.json` file)

```
ubuntu@ip-172-31-33-122:~/TestRepo/Repo1$ npm install
npm WARN deprecated uuid@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 8s

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 major version of npm available! 8.19.4 -> 10.5.1
npm notice Changelog: https://github.com/npm/cli/releases/tag/v10.5.1
npm notice Run npm install -g npm@10.5.1 to update!
npm notice
ubuntu@ip-172-31-33-122:~/TestRepo/Repo1$
```

->***node index.js***

(The command `node index.js` is used to execute a JavaScript file named `index.js` using the Node.js runtime environment.)

```
ubuntu@ip-172-31-33-122:~/TestRepo/Repo1$ node index.js
Started server
```

26. Open a new browser window and enter the IPv4 address followed by ":4000", like "IPv4_address:4000".