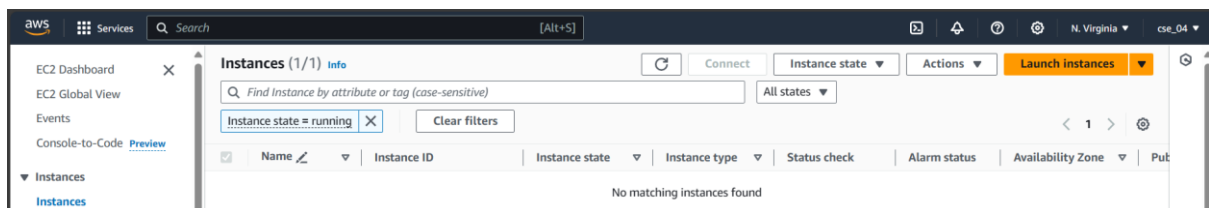


Assignment No: 12

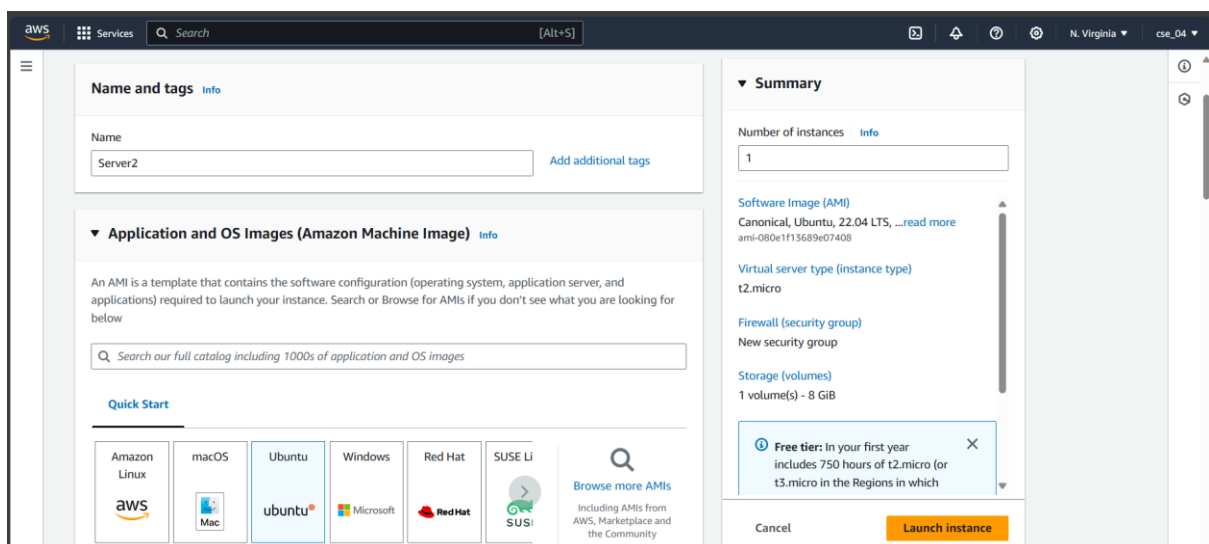
Problem Statement: Deploy and run the project in AWS without using port.

Steps:

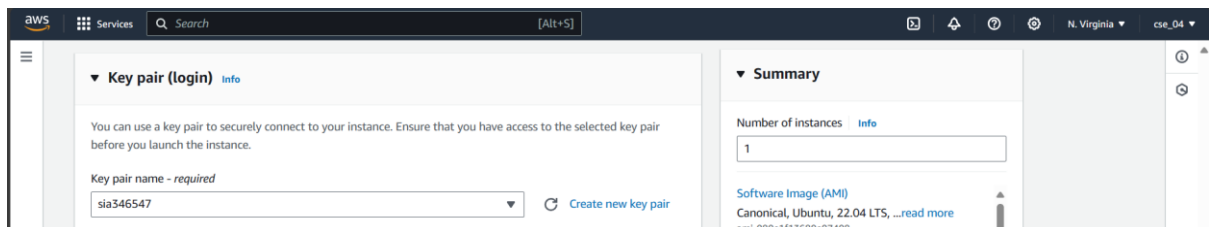
1. At first go to EC2 and then click on **Security groups**. And create a Security Group or take previously created Security group.
2. Now, go to EC2 dashboard and click on **Launch instance**.



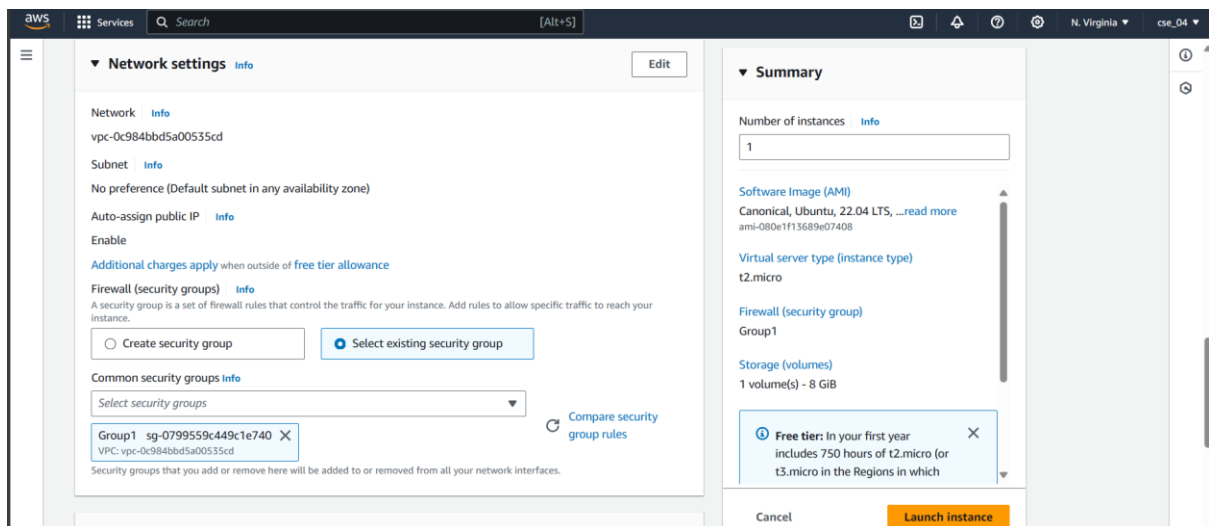
3. Fill up the **instance name** and select **Ubuntu** as the **AMI**.



4. Select an **existing key pair** or create a new one.



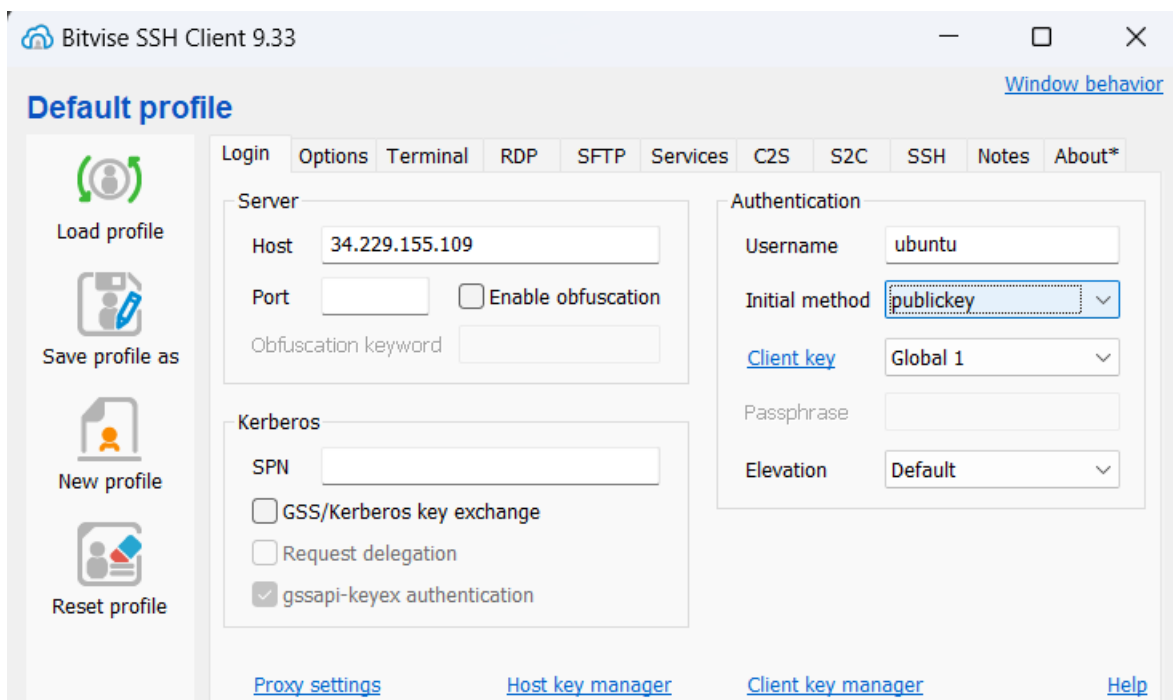
5. Now, select **Existing security group** and select the newly created security group, and click on **Launch Instance**.



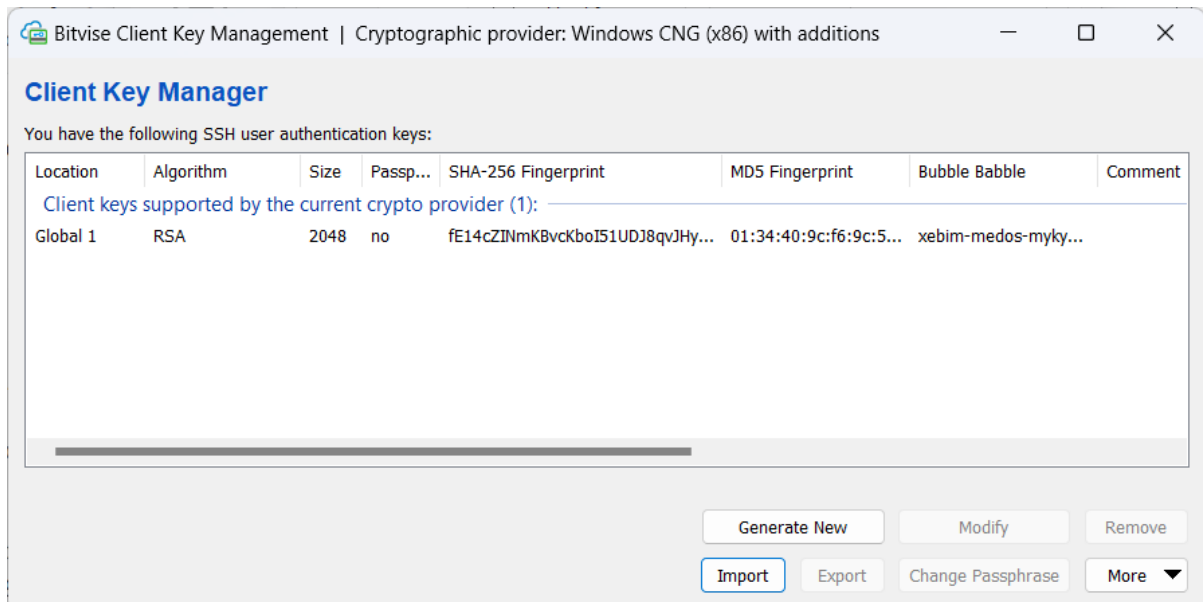
6. Click on **Instance** and copy **public IPv4 address**.



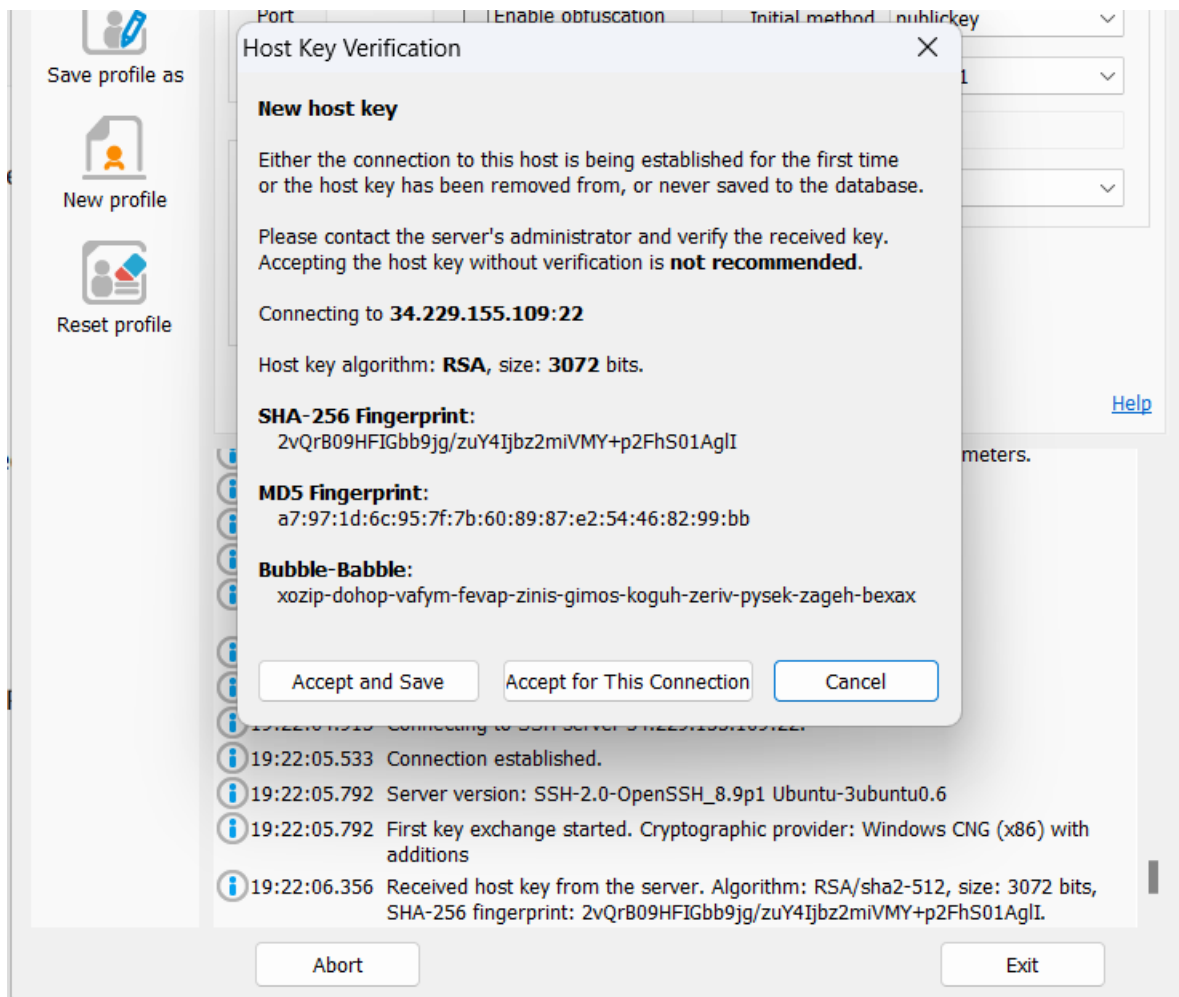
7. Paste the IPv4 address into the **host field** and enter Ubuntu as the **username** and **Client key** as global1 in Bit vise SSH Client.



8. Go to **Client key manager** and import that downloaded key.



9. Now click on **log in** and also click on **accept and save**.



10. Now open **terminal console** after login and then write all commands:

- pwd

```
ubuntu@ip-172-31-23-162:~$ pwd
/home/ubuntu
```

- sudo apt-get update

```
ubuntu@ip-172-31-23-162:~$ sudo apt-get update
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy InRelease
Hit:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates InRelease
Hit:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-backports InRelease
Hit:4 http://security.ubuntu.com/ubuntu jammy-security InRelease
Reading package lists... Done
```

- sudo apt-get upgrade

```
ubuntu@ip-172-31-23-162:~$ 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 python3-update-manager ubuntu-advantage-tools
  ubuntu-pro-client-l10n update-manager-core
0 upgraded, 0 newly installed, 0 to remove and 7 not upgraded.
```

- sudo apt-get install nginx.
- curl -sL https://deb.nodesource.com/setup_18.x | sudo -E bash -

```
ubuntu@ip-172-31-23-162:~$ sudo apt-get install nginx
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
nginx is already the newest version (1.18.0-6ubuntu14.4).
0 upgraded, 0 newly installed, 0 to remove and 7 not upgraded.
ubuntu@ip-172-31-23-162:~$ curl -sL https://deb.nodesource.com/setup_18.x | sudo -E bash -
2024-04-21 14:32:38 - Installing pre-requisites
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy InRelease
Hit:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates InRelease
curl is already the newest version (7.81.0-1ubuntu1.16).
gnupg is already the newest version (2.2.27-3ubuntu2.1).
apt-transport-https is already the newest version (2.4.12).
0 upgraded, 0 newly installed, 0 to remove and 7 not upgraded.
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy InRelease
Hit:2 http://security.ubuntu.com/ubuntu jammy-security InRelease
Hit:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates InRelease
Hit:4 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-backports InRelease
Get:5 https://deb.nodesource.com/node_18.x nodistro InRelease [12.1 kB]
Get:6 https://deb.nodesource.com/node_18.x nodistro/main amd64 Packages [8387 B]
Fetched 20.5 kB in 1s (23.7 kB/s)
Reading package lists... Done
2024-04-21 14:32:43 - Repository configured successfully. To install Node.js, run: apt-get install nodejs -y
```

- Sudo apt install nodejs

```
ubuntu@ip-172-31-23-162:~$ sudo apt install nodejs
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following NEW packages will be installed:
  nodejs
0 upgraded, 1 newly installed, 0 to remove and 7 not upgraded.
Need to get 29.6 MB of archives.
After this operation, 187 MB of additional disk space will be used.
Get:1 https://deb.nodesource.com/node_18.x nodistro/main amd64 nodejs amd64 18.20.2-1nodesource1 [29.6 MB]
Fetched 29.6 MB in 1s (53.2 MB/s)
Selecting previously unselected package nodejs.
(Reading database ... 65516 files and directories currently installed.)
Preparing to unpack .../nodejs_18.20.2-1nodesource1_amd64.deb ...
Unpacking nodejs (18.20.2-1nodesource1) ...
Setting up nodejs (18.20.2-1nodesource1) ...
Processing triggers for man-db (2.10.2-1)
```

- git clone <https://github.com/mampi6das/Repo1.git>

```
ubuntu@ip-172-31-23-162:~$ git clone https://github.com/mampi6das/Repo1.git
Cloning into 'Repo1'...
remote: Enumerating objects: 6, done.
remote: Counting objects: 100% (6/6), done.
remote: Compressing objects: 100% (6/6), done.
remote: Total 6 (delta 0), reused 6 (delta 0), pack-reused 0
Receiving objects: 100% (6/6), 48.00 KiB | 8.00 MiB/s, done.
```

- cd Repo1
- ls

```
ubuntu@ip-172-31-23-162:~$ cd Repo1
ubuntu@ip-172-31-23-162:~/Repo1$ ls
'New Text Document.txt'  index.js  package-lock.json  package.json
```

- npm install

```
ubuntu@ip-172-31-23-162:~/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 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 patch version of npm available! 10.5.0 -> 10.5.2
npm notice Changelog: https://github.com/npm/cli/releases/tag/v10.5.2
npm notice Run npm install -g npm@10.5.2 to update!
npm notice
```

- node index.js

```
ubuntu@ip-172-31-23-162:~/Repo1$ node index.js
Started server
```

11. Now server has started. If we paste public IPv4 in Browser we can see nginx has started.

Welcome to nginx!

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to nginx.org.
Commercial support is available at nginx.com.

Thank you for using nginx.

12. Now stop the server using ctrl + c on keyboard.

```
ubuntu@ip-172-31-23-162:~/Repo1$ node index.js
Started server
^C
ubuntu@ip-172-31-23-162:~/Repo1$
```

13. Now write these all commands:

- cd /
- Pwd
- cd etc/nginx/sites-available/
- sudo nano default

```
ubuntu@ip-172-31-23-162:~/Repo1$ cd /
ubuntu@ip-172-31-23-162:/$ pwd
/
ubuntu@ip-172-31-23-162:/$ cd etc/nginx/sites-available/
ubuntu@ip-172-31-23-162:/etc/nginx/sites-available$ sudo nano default
ubuntu@ip-172-31-23-162:/etc/nginx/sites-available$
```

```
GNU nano 6.2                                default
#
# You should look at the following URL's in order to grasp a solid understanding
# of Nginx configuration files in order to fully unleash the power of Nginx.
# https://www.nginx.com/resources/wiki/start/
# https://www.nginx.com/resources/wiki/start/topics/tutorials/config_pitfalls/
# https://wiki.debian.org/Nginx/DirectoryStructure
#
# In most cases, administrators will remove this file from sites-enabled/ and
# leave it as reference inside of sites-available where it will continue to be
# updated by the nginx packaging team.
#
# This file will automatically load configuration files provided by other
# applications, such as Drupal or Wordpress. These applications will be made
# available underneath a path with that package name, such as /drupal8.
#
# Please see /usr/share/doc/nginx-doc/examples/ for more detailed examples.
##

# Default server configuration
#
server {
    listen 80 default_server;
    listen [::]:80 default_server;
```

14. A new window will be opened. There at first go to location area and comment all codes and the write:

```
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;}
```

```
# Add index.php to the list if you are using PHP
index index.html index.htm index.nginx-debian.html;

server_name _;

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;
}
```

15. After it click ctrl + x, then y then press enter.

```
server_name _;

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;
}

# First attempt to serve request as file, then
# as directory, then fall back to displaying a 404.
```

Save modified buffer?

Y	Yes
N	No
^C	Cancel

16. Now open new server terminal and write **cd Repo1**.

```
ubuntu@ip-172-31-23-162:~$ cd Repo1
ubuntu@ip-172-31-23-162:~/Repo1$
```

17. Write **sudo systemctl restart nginx**.

```
ubuntu@ip-172-31-1-148:~/repo$ sudo systemctl restart nginx
ubuntu@ip-172-31-1-148:~/repo$
```

18. Started the server by execute **node index.js**.

```
ubuntu@ip-172-31-23-162:~/Repo1$ node index.js
Started server
```

19. Now copy that public IPv4 address again and paste it in URL and there you can see that without giving port (:4000) with URL we have hosted the website.

```
< ↻ ⓘ https://34.229.155.109
Hello Students...
```