Advanced EC2 Web App Project

Project Name: EC2 + Nginx + Node.js Web Application Deployment

1 Introduction

This project demonstrates deploying a Node.js application on an AWS EC2 (Ubuntu) instance and setting up Nginx as a reverse proxy. The goal is to host a Node.js app on EC2 and make it accessible via the EC2 Public IP.

2 EC2 Instance Launch

- Launched an EC2 t2.micro instance using Ubuntu 24.04 LTS AMI.

- Created a Key Pair for SSH access.

- Configured the Security Group with the following rules:

  - SSH (22) from My IP
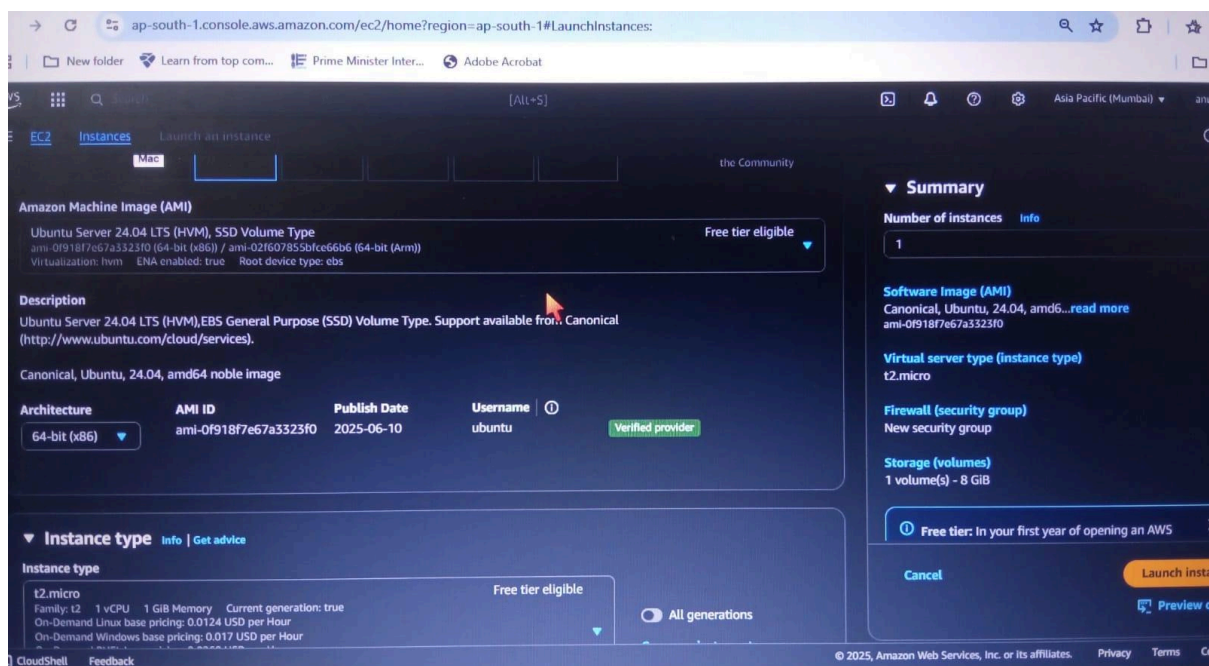
  - HTTP (80) from Anywhere
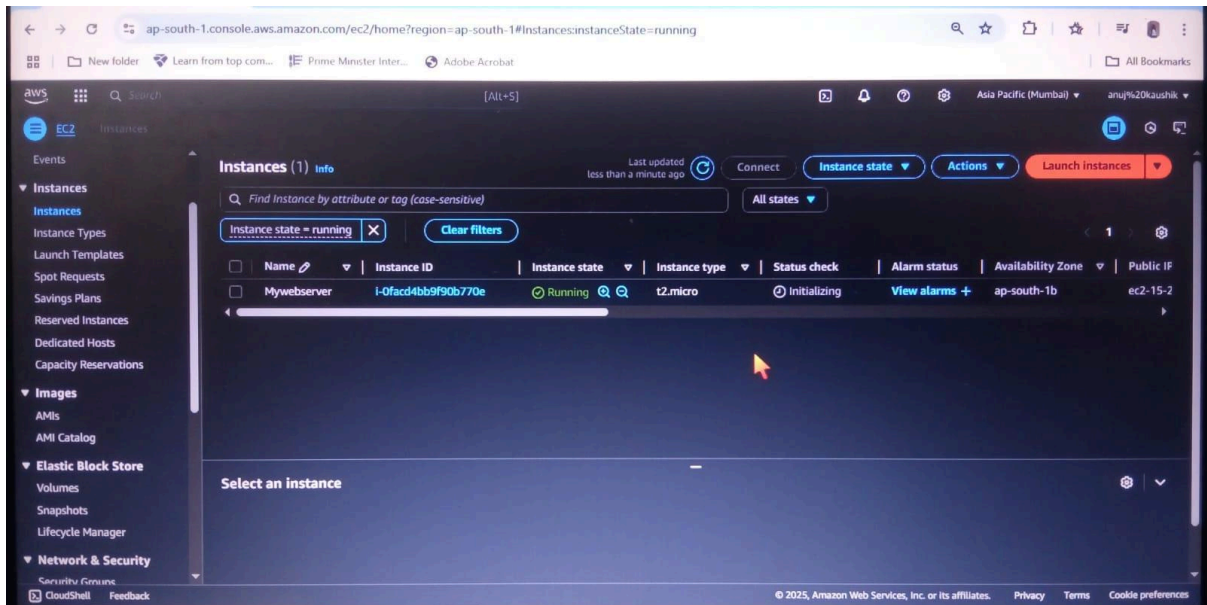

Figure 2.1: EC2 Instance Creation Page

Figure 2.2: EC2 Instance Running Status

3 SSH Connection

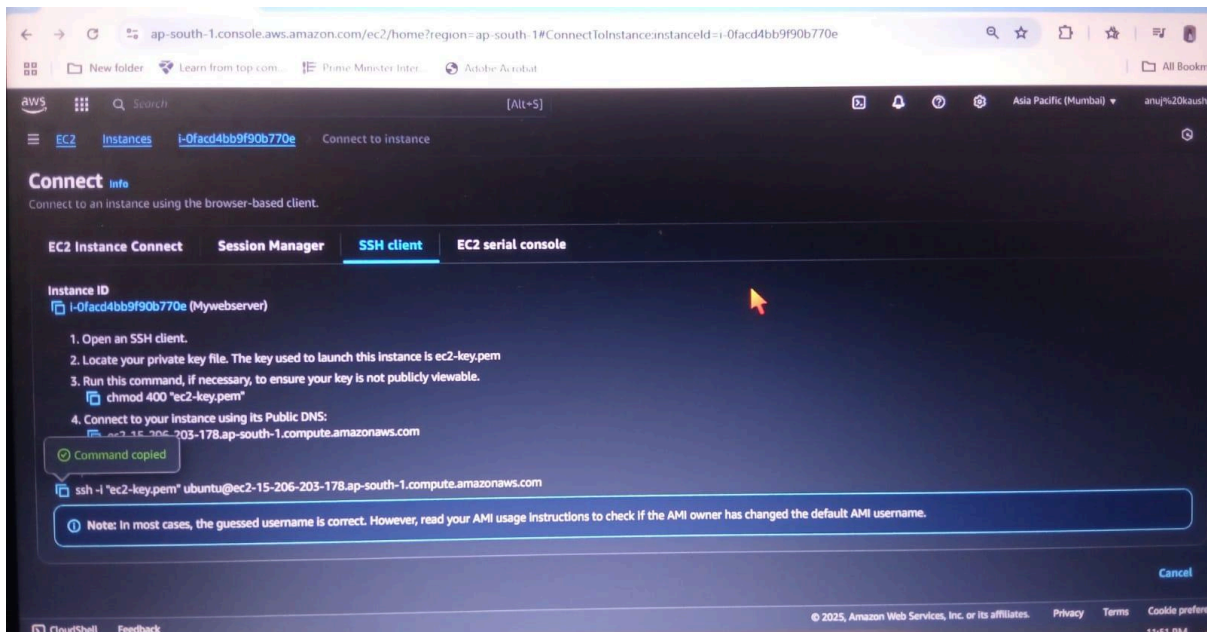Connect to EC2 using SSH:

ssh -i "yourkey.pem" ubuntu@<EC2-Public-IP

Figure 3.1: SSH Connection Terminal Page

Figure 3.2: SSH Connection Terminal Output

4 System Update

Updated the system packages:

sudo apt update && sudo apt upgrade -y

Figure 4: System Update Output

5 Nginx Install & Configuration
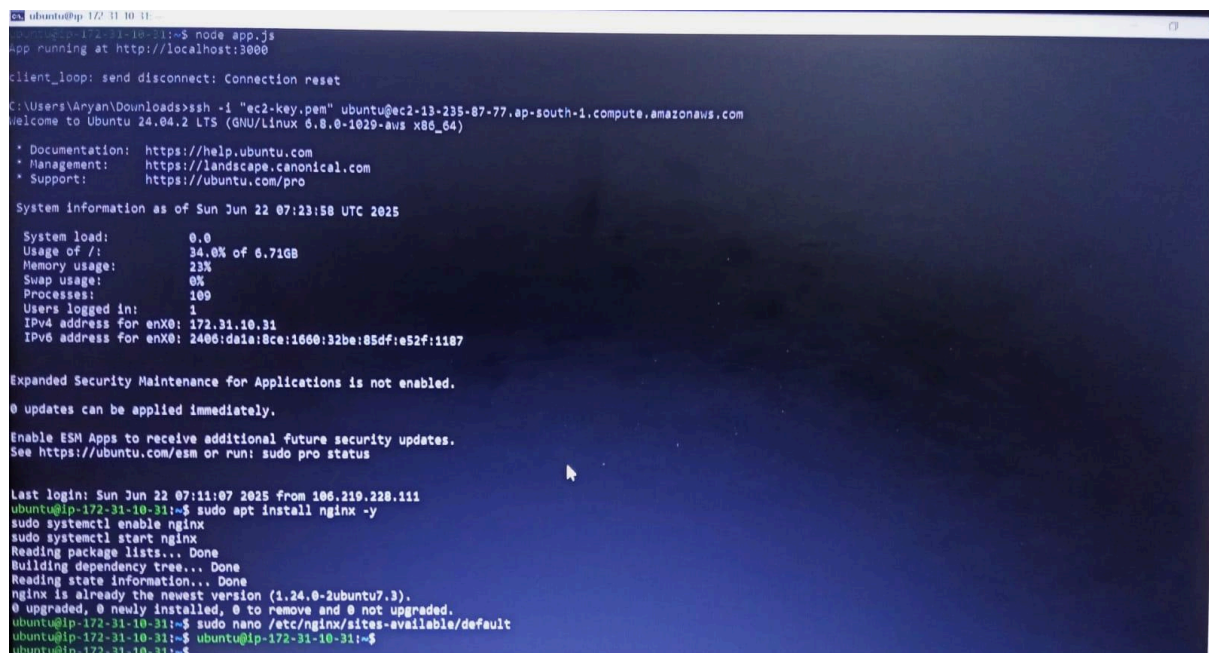
Installed Nginx:

```
sudo apt install nginx -y
sudo systemctl enable nginx
sudo systemctl start nginx
```

Updated Nginx configuration (`/etc/nginx/sites-available/default`):

```
server {
    listen 80;
    location / {
        proxy_pass http://localhost:3000;
    }
}
```

Restarted Nginx:

```
sudo systemctl restart nginx
```


Figure 5.1: Terminal Output for sudo apt install nginx

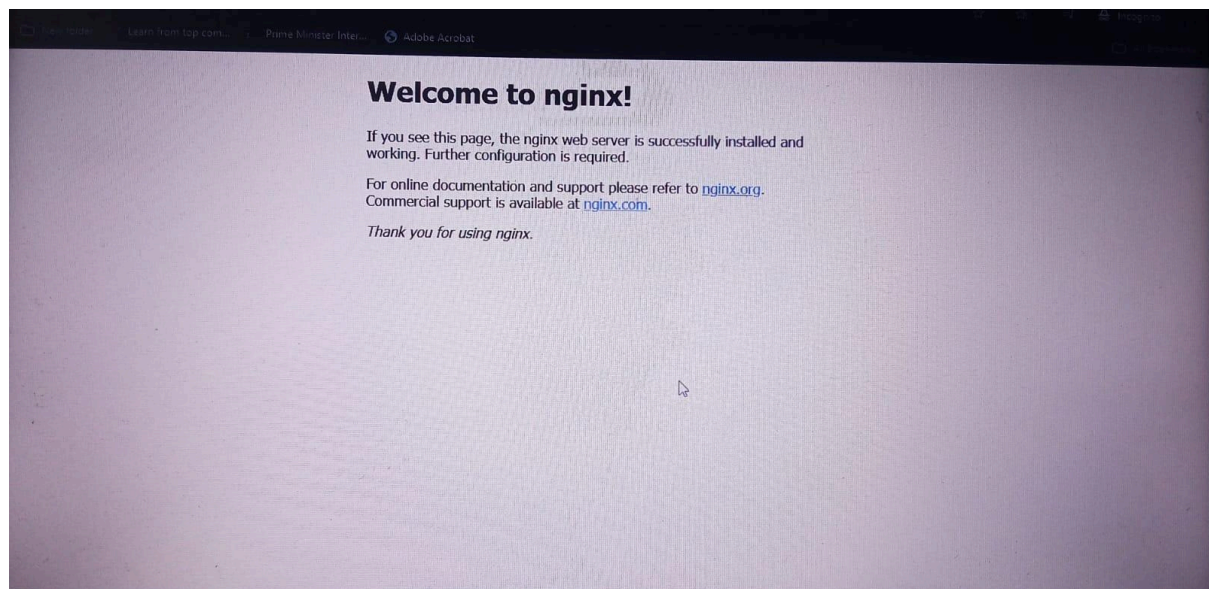Figure 5.2: Terminal Output for sudo apt install nginx

Figure 5.3: nginx output in Browser

6 Node.js Install

Installed Node.js:

```
curl -fsSL https://deb.nodesource.com/setup_18.x | sudo -E bash -
sudo apt install -y nodejs
node -v
npm -v
```

Figure 6: Terminal Output for Node.js (node -v & npm -v)

7 Created Node.js App

App (`app.js`) code:

```
const http = require("http");
const port = 3000;
const requestHandler = (req, res) => {
  res.end("Hello Aryan! Node.js App is running on EC2!");
};
const server = http.createServer(requestHandler);
server.listen(port, () => {
  console.log(`App running at http://localhost:${port}`);
});
```

Run the app:

node app.js

Figure 7: Terminal Output (App running at http://localhost:3000)

8 Final Output

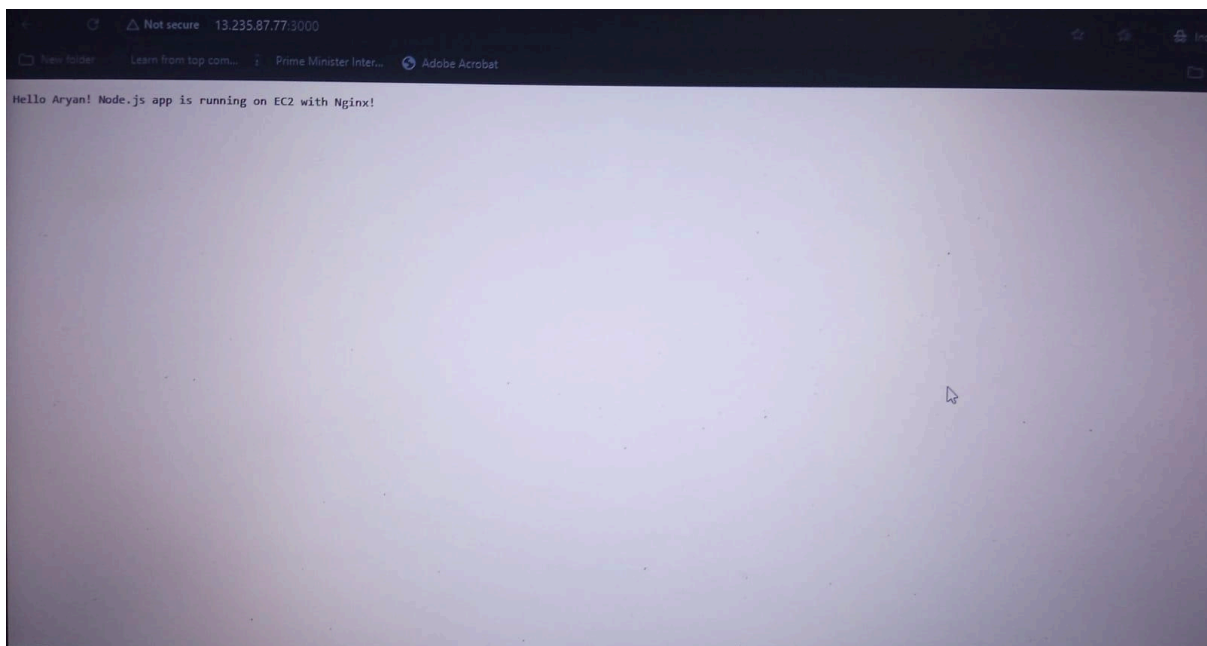App is now available at:

http://<EC2-Public-IP>

Figure 8: Final Output in Browser (Hello Aryan! Node.js App is running on EC2!)

✅ Conclusion

- Deployed a Node.js app on EC2

- Configured Nginx as a reverse proxy

- Created a fully working environment for a Node.js app on AWS EC2