AWS Project Report – Secure VPC Architecture with EC2 Web Server

Project Overview

This project demonstrates the creation of a secure and scalable network architecture using Amazon VPC, followed by deploying an Apache web server on an EC2 instance. The goal of this project is to build and configure a fully functional AWS environment with security best practices, custom networking, and successful web server hosting.

1. Custom VPC Creation

A custom Virtual Private Cloud (VPC) was created with the following components:

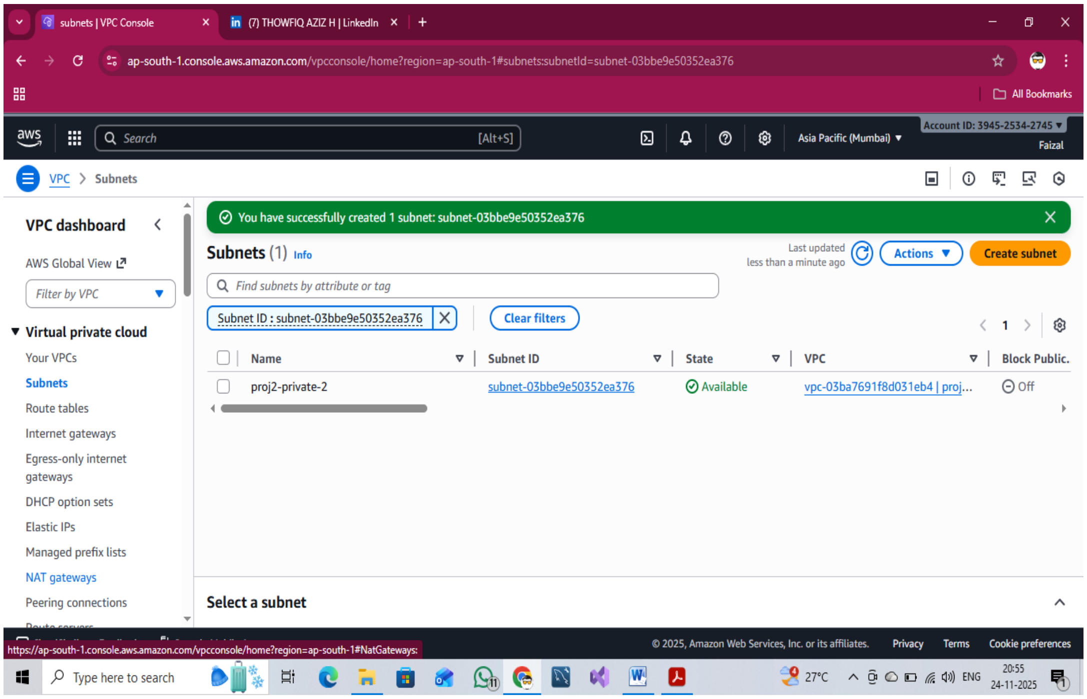
- CIDR Block: 10.0.0.0/16

- Two public subnets in different Availability Zones

- An Internet Gateway (IGW) attached to the VPC

- A public route table configured with route 0.0.0.0/0 pointing to the IGW

- Subnet associations mapped to ensure proper routing



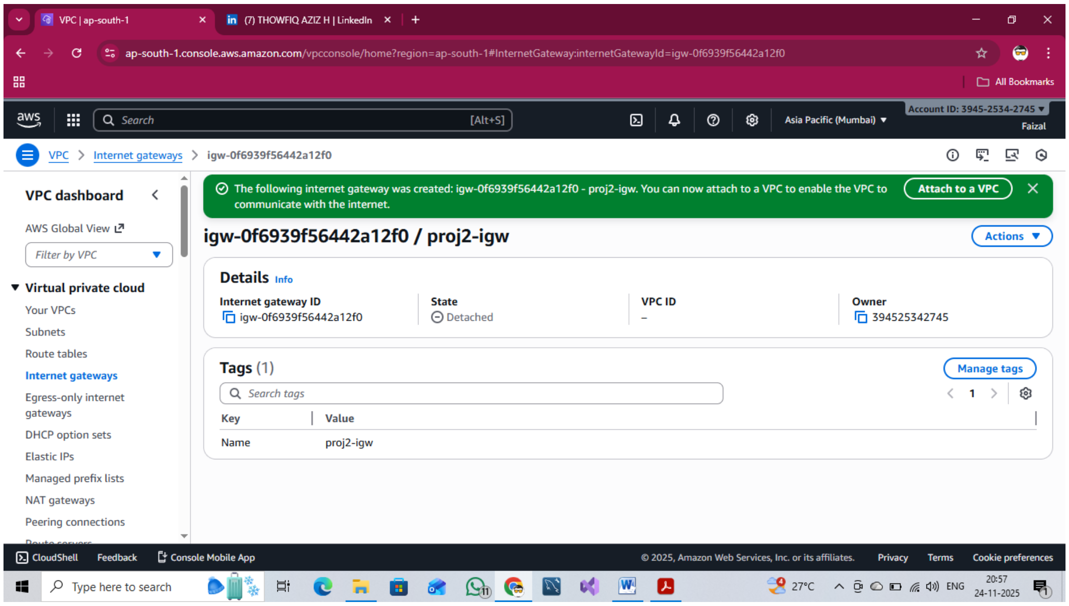
ap-south-1.console.aws.amazon.com/vpcconsole/home?region=ap-south-1#subnets:subnetId=subnet-03bbe9e50352ea376

You have successfully created 1 subnet: subnet-03bbe9e50352ea376

Subnets (1)

Subnet ID : subnet-03bbe9e50352ea376

Name	Subnet ID	State	VPC	Block Public.
proj2-private-2	subnet-03bbe9e50352ea376	Available	vpc-03ba7691f8d031eb4 | proj...	Off



ap-south-1.console.aws.amazon.com/vpcconsole/home?region=ap-south-1#InternetGateway:internetGatewayId=igw-0f6939f56442a12f0

VPC > Internet gateways > igw-0f6939f56442a12f0

The following internet gateway was created: igw-0f6939f56442a12f0 - proj2-igw. You can now attach to a VPC to enable the VPC to communicate with the internet.

igw-0f6939f56442a12f0 / proj2-igw

Details

Internet gateway ID	State	VPC ID	Owner
igw-0f6939f56442a12f0	Detached	–	394525342745

Tags (1)

Key	Value
Name	proj2-igw

2. EC2 Instance Deployment

An Amazon Linux EC2 instance was deployed inside one of the public subnets. The following configurations were applied:

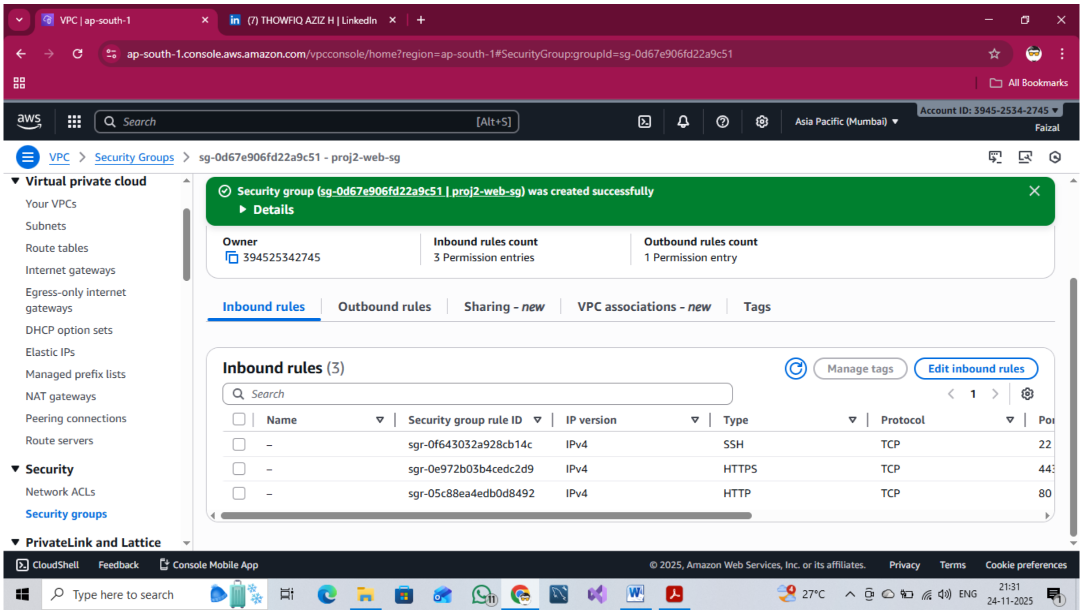
- Instance Type: t2.micro / t3.micro (Free-tier eligible)

- SSH Key Pair (.pem) generated for secure access

- Security Group configured with inbound rules: SSH (22), HTTP (80), HTTPS (443)

- Public IP enabled to allow internet access

- Connected to the instance using PuTTY after converting PEM to PPK



ap-south-1.console.aws.amazon.com/vpcconsole/home?region=ap-south-1#SecurityGroup:groupId=sg-0d67e906fd22a9c51

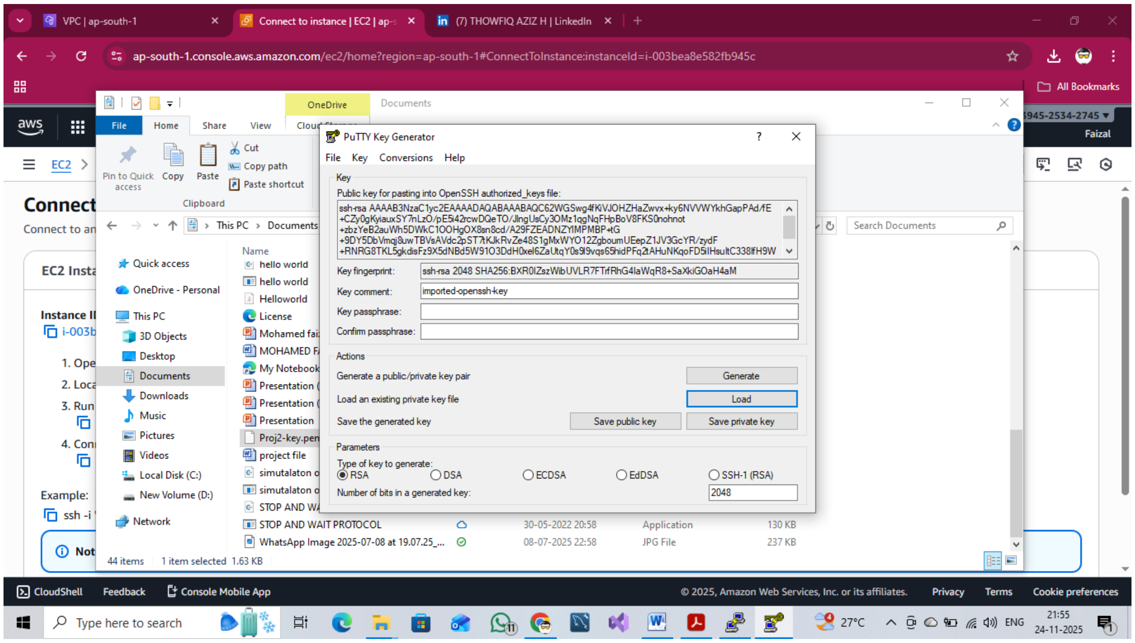
VPC > Security Groups > sg-0d67e906fd22a9c51 - proj2-web-sg

Security group (sg-0d67e906fd22a9c51 | proj2-web-sg) was created successfully

Owner	Inbound rules count	Outbound rules count
394525342745	3 Permission entries	1 Permission entry

Inbound rules (3)

Name	Security group rule ID	IP version	Type	Protocol	Po
–	sgr-0f643032a928cb14c	IPv4	SSH	TCP	22
–	sgr-0e972b03b4cedc2d9	IPv4	HTTPS	TCP	44
–	sgr-05c88ea4edb0d8492	IPv4	HTTP	TCP	80



ap-south-1.console.aws.amazon.com/ec2/home?region=ap-south-1#ConnectToInstance:instanceId=i-003bea8e582fb945c

PuTTY Key Generator

Key

Public key for pasting into OpenSSH authorized_keys file:

```
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABAQC62WGSwg4fKiVJOHZHaZwvx+ky6NVVWYkhGapPAd/fE
+CZy0gKyiauxSY7nLzO/pE5i42rcwDQeTO/JlngUsCy3OMz1qgNqFHpBoV8FKS0nohnot
+zbzYeB2auWh5DWkC1OOHgOX8sn8cd/A29FZEADNZYIMPMBP+tG
+9DY5DbVmqj8uwTBVsAVdc2pST7tKJkRvZe48S1gMxWYO12ZgboumUEepZ1JV3GcYR/zydF
+RNRG8TKL5gkdisFz9X5dNBd5W91O3DdH0xel6ZaUtqY0s9l9vqs65hidPFq2tAHuNKqoFD5iIHsultC338IfH9W
```

Key fingerprint: ssh-rsa 2048 SHA256:BXR0IZszWibUVLR7FTrfRhG4laWqR8+SaXkiGOaH4aM

Key comment: imported-openssh-key

Key passphrase:

Confirm passphrase:

Actions

Generate a public/private key pair — Generate

Load an existing private key file — Load

Save the generated key — Save public key — Save private key

Parameters

Type of key to generate:
● RSA ○ DSA ○ ECDSA ○ EdDSA ○ SSH-1 (RSA)

Number of bits in a generated key: 2048

3. Apache Web Server Configuration

After establishing SSH access to the EC2 instance, Apache HTTP Server was installed using the Yum package manager. The service was enabled to start automatically on system boot. A custom HTML file was deployed inside /var/www/html, replacing the default Apache test page. The web server was successfully accessed through the EC2 Public IP, confirming that the instance, routing, and security configurations were functioning correctly.

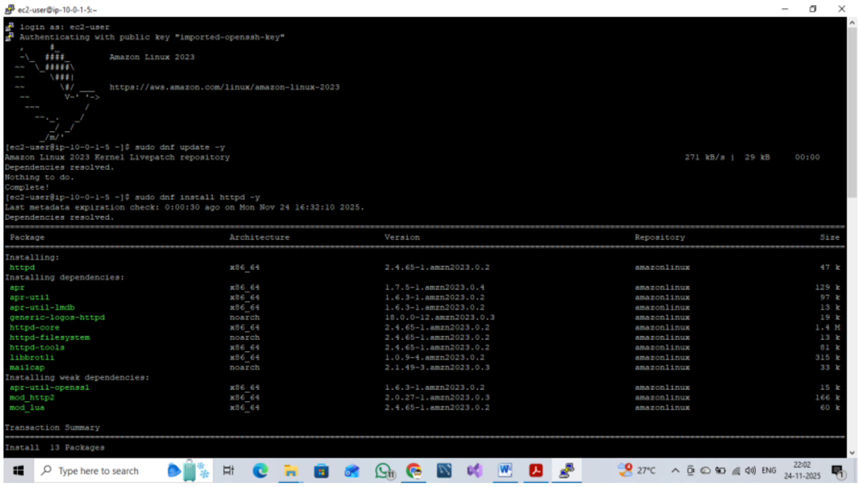
Commands used:

sudo yum update -y

sudo yum install httpd -y

sudo systemctl start httpd

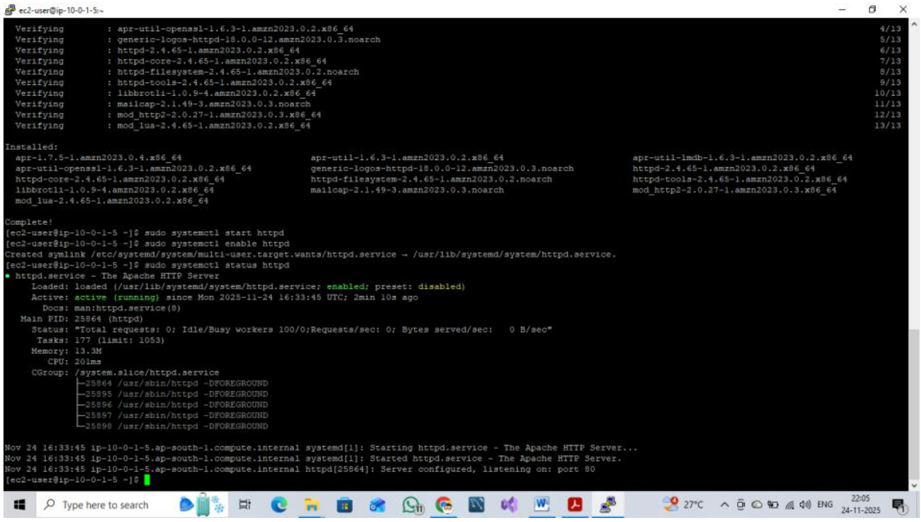
sudo systemctl enable httpd



ec2-user@ip-10-0-1-5:~

```
login as: ec2-user
Authenticating with public key "imported-openssh-key"
   ,     #_
   ~\_  ####_        Amazon Linux 2023
  ~~  \_#####\
  ~~     \###|
  ~~       \#/ ___   https://aws.amazon.com/linux/amazon-linux-2023
   ~~       V~' '->
    ~~~         /
      ~~._.   _/
         _/ _/
       _/m/'
[ec2-user@ip-10-0-1-5 ~]$ sudo dnf update -y
Amazon Linux 2023 Kernel Livepatch repository                          271 kB/s |  29 kB     00:00
Dependencies resolved.
Nothing to do.
Complete!
[ec2-user@ip-10-0-1-5 ~]$ sudo dnf install httpd -y
Last metadata expiration check: 0:00:30 ago on Mon Nov 24 16:32:10 2025.
Dependencies resolved.
==========================================================================================================
 Package                   Architecture     Version                        Repository              Size
==========================================================================================================
Installing:
 httpd                     x86_64           2.4.65-1.amzn2023.0.2          amazonlinux             47 k
Installing dependencies:
 apr                       x86_64           1.7.5-1.amzn2023.0.4           amazonlinux            129 k
 apr-util                  x86_64           1.6.3-1.amzn2023.0.2           amazonlinux             97 k
 apr-util-lmdb             x86_64           1.6.3-1.amzn2023.0.2           amazonlinux             13 k
 generic-logos-httpd       noarch           18.0.0-12.amzn2023.0.3         amazonlinux             19 k
 httpd-core                x86_64           2.4.65-1.amzn2023.0.2          amazonlinux            1.4 M
 httpd-filesystem          noarch           2.4.65-1.amzn2023.0.2          amazonlinux             13 k
 httpd-tools               x86_64           2.4.65-1.amzn2023.0.2          amazonlinux             81 k
 libbrotli                 x86_64           1.0.9-4.amzn2023.0.2           amazonlinux            315 k
 mailcap                   noarch           2.1.49-3.amzn2023.0.3          amazonlinux             33 k
Installing weak dependencies:
 apr-util-openssl          x86_64           1.6.3-1.amzn2023.0.2           amazonlinux             15 k
 mod_http2                 x86_64           2.0.27-1.amzn2023.0.3          amazonlinux            166 k
 mod_lua                   x86_64           2.4.65-1.amzn2023.0.2          amazonlinux             60 k

Transaction Summary
==========================================================================================================
Install  13 Packages
```



```
  Verifying        : apr-util-openssl-1.6.3-1.amzn2023.0.2.x86_64                                                                        4/13
  Verifying        : generic-logos-httpd-18.0.0-12.amzn2023.0.3.noarch                                                                   5/13
  Verifying        : httpd-2.4.65-1.amzn2023.0.2.x86_64                                                                                  6/13
  Verifying        : httpd-core-2.4.65-1.amzn2023.0.2.x86_64                                                                             7/13
  Verifying        : httpd-filesystem-2.4.65-1.amzn2023.0.2.noarch                                                                       8/13
  Verifying        : httpd-tools-2.4.65-1.amzn2023.0.2.x86_64                                                                            9/13
  Verifying        : libbrotli-1.0.9-4.amzn2023.0.2.x86_64                                                                              10/13
  Verifying        : mailcap-2.1.49-3.amzn2023.0.3.noarch                                                                               11/13
  Verifying        : mod_http2-2.0.27-1.amzn2023.0.3.x86_64                                                                             12/13
  Verifying        : mod_lua-2.4.65-1.amzn2023.0.2.x86_64                                                                               13/13

Installed:
  apr-1.7.5-1.amzn2023.0.4.x86_64                         apr-util-1.6.3-1.amzn2023.0.2.x86_64                          apr-util-lmdb-1.6.3-1.amzn2023.0.2.x86_64
  apr-util-openssl-1.6.3-1.amzn2023.0.2.x86_64            generic-logos-httpd-18.0.0-12.amzn2023.0.3.noarch             httpd-2.4.65-1.amzn2023.0.2.x86_64
  httpd-core-2.4.65-1.amzn2023.0.2.x86_64                 httpd-filesystem-2.4.65-1.amzn2023.0.2.noarch                 httpd-tools-2.4.65-1.amzn2023.0.2.x86_64
  libbrotli-1.0.9-4.amzn2023.0.2.x86_64                   mailcap-2.1.49-3.amzn2023.0.3.noarch                          mod_http2-2.0.27-1.amzn2023.0.3.x86_64
  mod_lua-2.4.65-1.amzn2023.0.2.x86_64

Complete!
[ec2-user@ip-10-0-1-5 ~]$ sudo systemctl start httpd
[ec2-user@ip-10-0-1-5 ~]$ sudo systemctl enable httpd
Created symlink /etc/systemd/system/multi-user.target.wants/httpd.service → /usr/lib/systemd/system/httpd.service.
[ec2-user@ip-10-0-1-5 ~]$ sudo systemctl status httpd
● httpd.service - The Apache HTTP Server
     Loaded: loaded (/usr/lib/systemd/system/httpd.service; enabled; preset: disabled)
     Active: active (running) since Mon 2025-11-24 16:33:45 UTC; 2min 10s ago
       Docs: man:httpd.service(8)
   Main PID: 25864 (httpd)
     Status: "Total requests: 0; Idle/Busy workers 100/0;Requests/sec: 0; Bytes served/sec:   0 B/sec"
      Tasks: 177 (limit: 1053)
     Memory: 13.3M
        CPU: 201ms
     CGroup: /system.slice/httpd.service
             ├─25864 /usr/sbin/httpd -DFOREGROUND
             ├─25895 /usr/sbin/httpd -DFOREGROUND
             ├─25896 /usr/sbin/httpd -DFOREGROUND
             ├─25897 /usr/sbin/httpd -DFOREGROUND
             └─25898 /usr/sbin/httpd -DFOREGROUND

Nov 24 16:33:45 ip-10-0-1-5.ap-south-1.compute.internal systemd[1]: Starting httpd.service - The Apache HTTP Server...
Nov 24 16:33:45 ip-10-0-1-5.ap-south-1.compute.internal systemd[1]: Started httpd.service - The Apache HTTP Server.
Nov 24 16:33:45 ip-10-0-1-5.ap-south-1.compute.internal httpd[25864]: Server configured, listening on: port 80
[ec2-user@ip-10-0-1-5 ~]$ 
```

4. Security Best Practices Applied

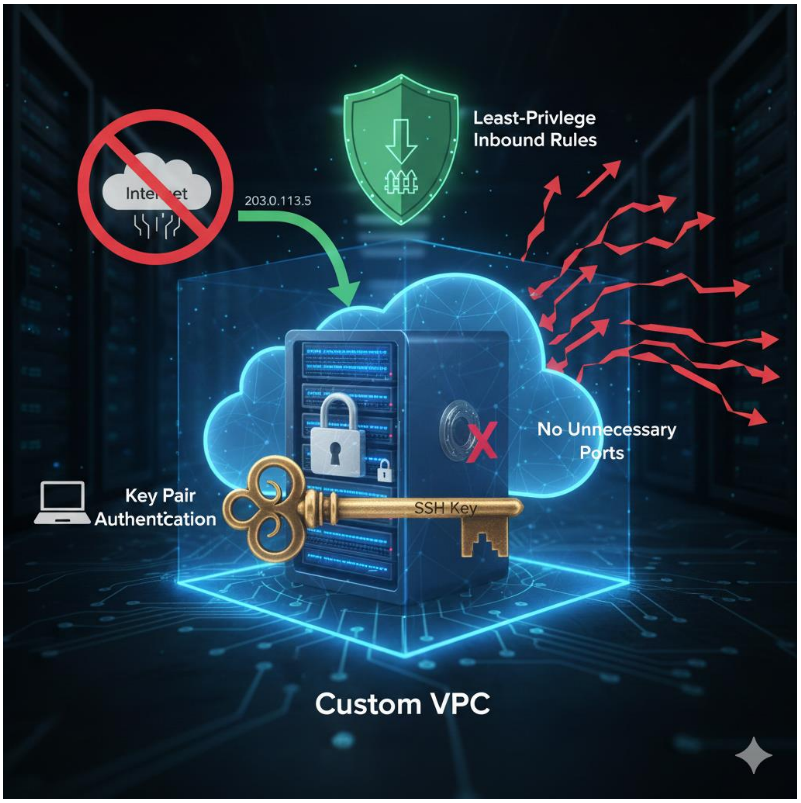
- Implemented least-privilege inbound rules in the Security Group

- Restricted SSH access to only the user's IP

- Used Key Pair authentication instead of password login

- Ensured no unnecessary ports were open

- Kept the architecture isolated inside a custom VPC



5. Final Output

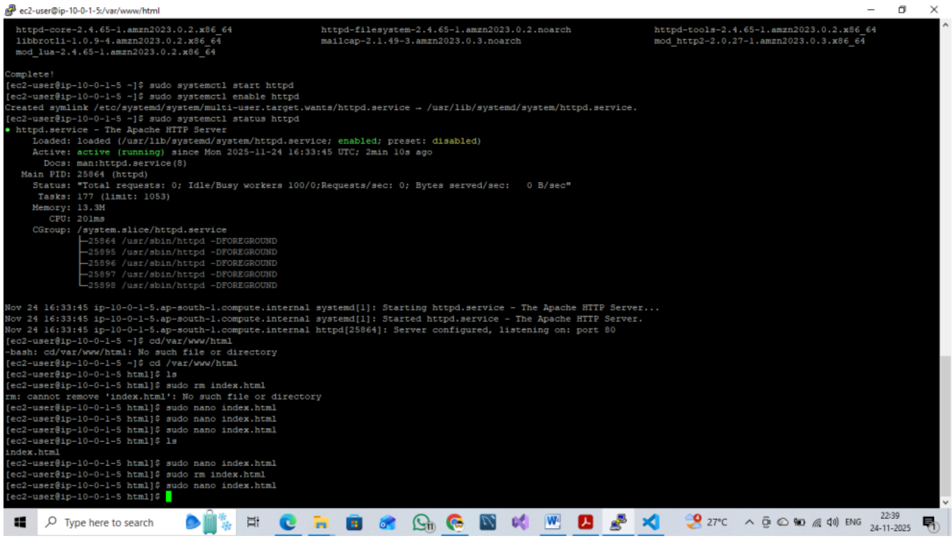
The project resulted in a fully functional and secure AWS environment. The EC2 web server displayed the hosted HTML webpage successfully, confirming correct configuration of:

- VPC and subnets

- Route tables and IGW

- Security Groups

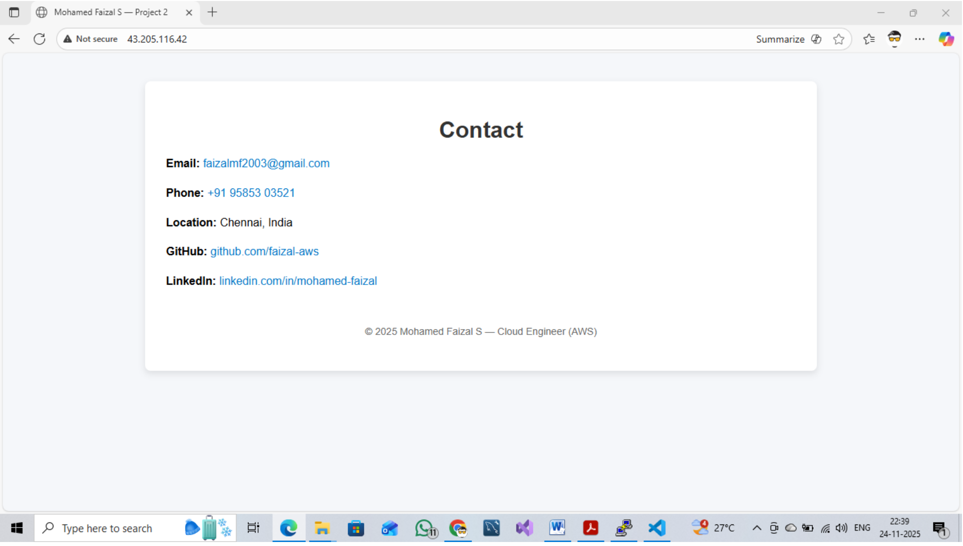
- Apache server setup



```
  httpd-core-2.4.65-1.amzn2023.0.2.x86_64                   httpd-filesystem-2.4.65-1.amzn2023.0.2.noarch             httpd-tools-2.4.65-1.amzn2023.0.2.x86_64
  libbrotli-1.0.9-4.amzn2023.0.2.x86_64                     mailcap-2.1.49-3.amzn2023.0.3.noarch                      mod_http2-2.0.27-1.amzn2023.0.3.x86_64
  mod_lua-2.4.65-1.amzn2023.0.2.x86_64

Complete!
[ec2-user@ip-10-0-1-5 ~]$ sudo systemctl start httpd
[ec2-user@ip-10-0-1-5 ~]$ sudo systemctl enable httpd
Created symlink /etc/systemd/system/multi-user.target.wants/httpd.service → /usr/lib/systemd/system/httpd.service.
[ec2-user@ip-10-0-1-5 ~]$ sudo systemctl status httpd
● httpd.service - The Apache HTTP Server
     Loaded: loaded (/usr/lib/systemd/system/httpd.service; enabled; preset: disabled)
     Active: active (running) since Mon 2025-11-24 16:33:45 UTC; 2min 10s ago
       Docs: man:httpd.service(8)
   Main PID: 25864 (httpd)
     Status: "Total requests: 0; Idle/Busy workers 100/0;Requests/sec: 0; Bytes served/sec:   0 B/sec"
      Tasks: 177 (limit: 1053)
     Memory: 13.3M
        CPU: 201ms
     CGroup: /system.slice/httpd.service
             ├─25864 /usr/sbin/httpd -DFOREGROUND
             ├─25895 /usr/sbin/httpd -DFOREGROUND
             ├─25896 /usr/sbin/httpd -DFOREGROUND
             ├─25897 /usr/sbin/httpd -DFOREGROUND
             └─25898 /usr/sbin/httpd -DFOREGROUND

Nov 24 16:33:45 ip-10-0-1-5.ap-south-1.compute.internal systemd[1]: Starting httpd.service - The Apache HTTP Server...
Nov 24 16:33:45 ip-10-0-1-5.ap-south-1.compute.internal systemd[1]: Started httpd.service - The Apache HTTP Server.
Nov 24 16:33:45 ip-10-0-1-5.ap-south-1.compute.internal httpd[25864]: Server configured, listening on: port 80
[ec2-user@ip-10-0-1-5 ~]$ cd/var/www/html
-bash: cd/var/www/html: No such file or directory
[ec2-user@ip-10-0-1-5 ~]$ cd /var/www/html
[ec2-user@ip-10-0-1-5 html]$ ls
[ec2-user@ip-10-0-1-5 html]$ sudo rm index.html
rm: cannot remove 'index.html': No such file or directory
[ec2-user@ip-10-0-1-5 html]$ sudo nano index.html
[ec2-user@ip-10-0-1-5 html]$ sudo nano index.html
[ec2-user@ip-10-0-1-5 html]$ sudo nano index.html
[ec2-user@ip-10-0-1-5 html]$ ls
index.html
[ec2-user@ip-10-0-1-5 html]$ sudo nano index.html
[ec2-user@ip-10-0-1-5 html]$ sudo rm index.html
[ec2-user@ip-10-0-1-5 html]$ sudo nano index.html
[ec2-user@ip-10-0-1-5 html]$ 
```



Contact

Email: faizalmf2003@gmail.com

Phone: +91 95853 03521

Location: Chennai, India

GitHub: github.com/faizal-aws

LinkedIn: linkedin.com/in/mohamed-faizal

© 2025 Mohamed Faizal S — Cloud Engineer (AWS)

This setup can serve as a foundation for more advanced architectures such as load balancers, private subnets, databases, auto scaling groups, and monitoring in future enhancements.