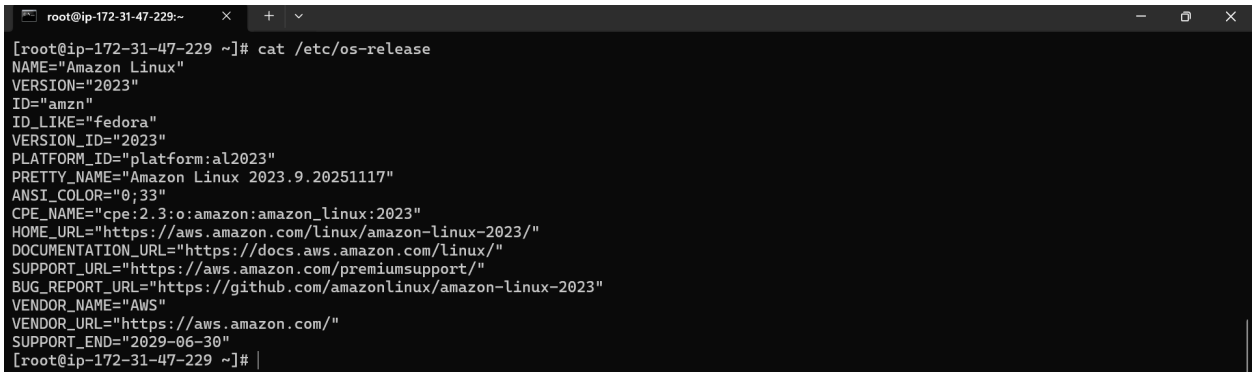
🌐 Web Server Project-1

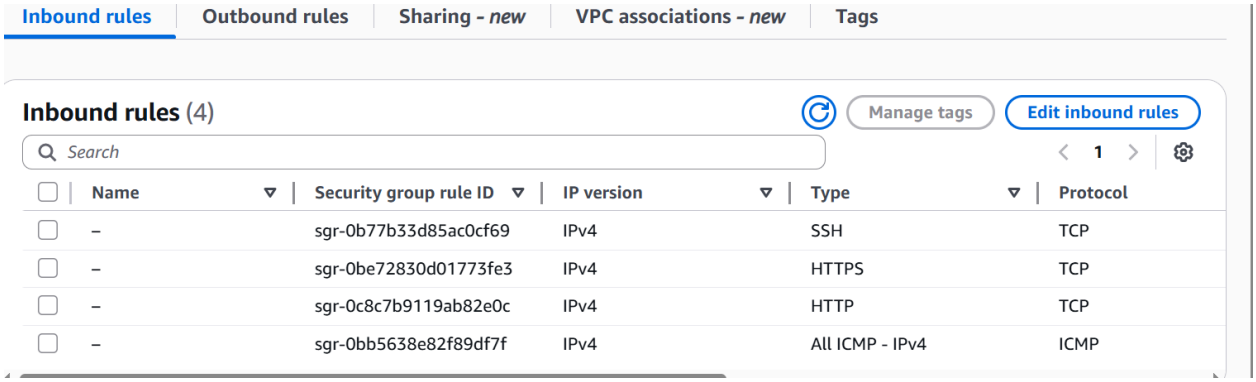
🌍 Project	🧠 Description
Apache Web Hosting	⚙️ Install and configure Apache web server to host a website site.
Virtual Host & SSL Setup	🔒 Configure virtual hosts and enable HTTPS using Open SSL certificates & Redirect all client traffic on https protocol only.

Step1 . Launch Instance

- **AMI → Amazon Linux 2023**
- **Instance → t2.micro (Free Tier)**
- **Key pair → create or use existing**



- **Security Group:**
    - **Allow 22 (SSH) from your IP**
    - **Allow 80 (HTTP) from anywhere**
    - **Allow 443 (HTTPS) from anywhere**



Step2 . SSH into EC2

ssh -i ec2key_key.pem ec2-user@EC2_PUBLIC_IP

Step 3. Install Apache Web Server

Install Apache

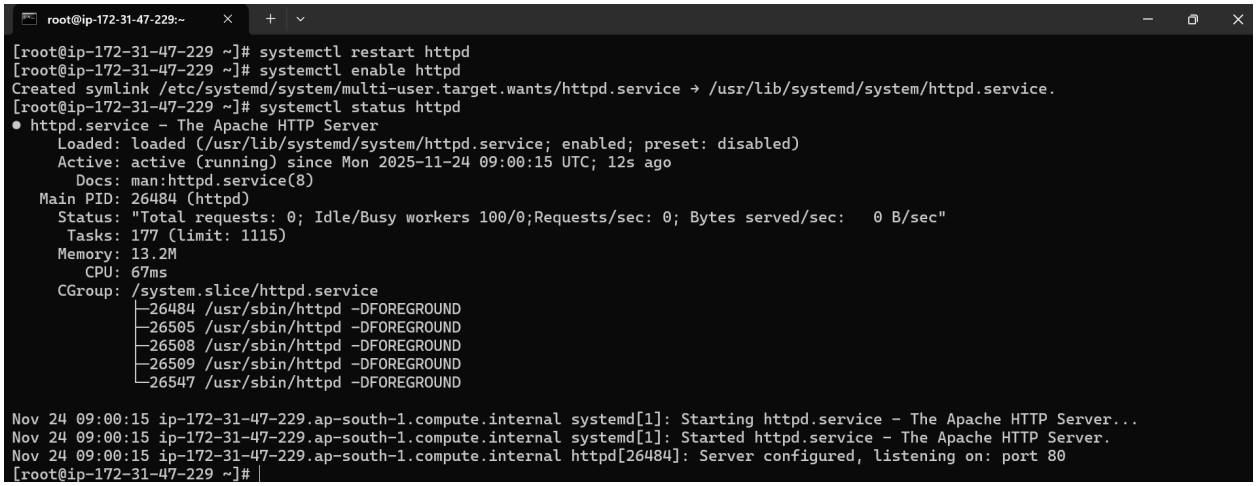
```
sudo dnf install httpd -y
```

Enable and start service

```
sudo systemctl enable httpd
sudo systemctl start httpd
```

Check status

```
sudo systemctl status httpd
```



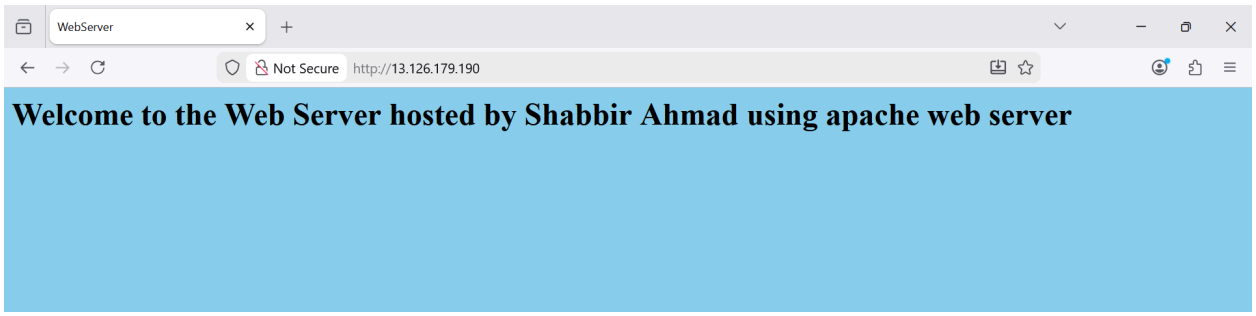
Step4 . create test webpage

#cd /var/www/html/
#vim index.html
<html>
<head><title>WebServer</title></head>
<body bgcolor=skyblue>
<h1>Welcome to the Web Server hosted by Shabbir Ahmad using apache web server</h1>
</body>
</html>
:wq

Access Web Server

Visit public IP in browser:

👉 http://13.126.179.190



Step 5 .Configure Virtual web hosting

Edit virtual host config

```
vim   /etc/httpd/conf.d/vhosts.conf
```

<VirtualHost *:80>
ServerName 13.126.179.190 **(# Use EC2 Pub-IP Because currently we do not have DNS)**
ServerAdmin root@example.com
DocumentRoot /var/www/html/
</VirtualHost>
<Directory /var/www/html/>
<RequireAll>
Require all granted
</RequireAll>
</Directory>

:wq

Restart Apache

```
sudo systemctl restart httpd
```

Step 6 .Configure/enable Apache Web hosting with HTTPS using Open SSL certificates .

Install Require package for https

#yum install httpd mod_ssl openssl -y

A- create self-signed Certificate and Key for https site

#openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout /home/server.key -out /home/server.crt

B- Modify ssl config file

#vim /etc/httpd/conf.d/ssl.conf

SSLCertificateFile /etc/pki/tls/certs/server.crt
SSLCertificateKeyFile /etc/pki/tls/private/server.key

:wq

C- Copy generated crt and key on required location

#cp -rvf /home/server.crt /etc/pki/tls/certs/
#cp -rvf /home/server.key /etc/pki/tls/private/

D- Update configuration file of pune and mumbai

#vim /etc/httpd/conf.d/vhosts.conf

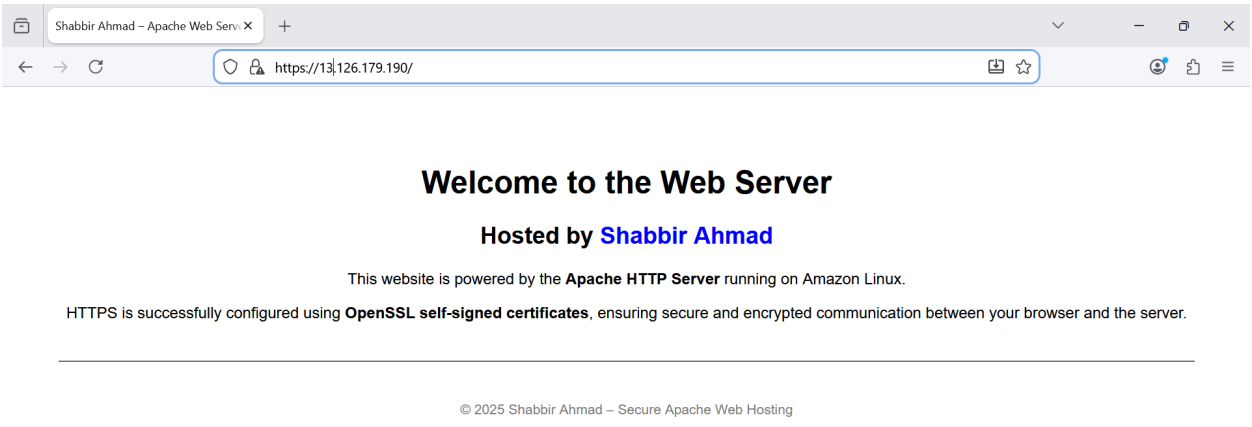
<VirtualHost *:443>
SSLEngine on
SSLCertificateFile /etc/pki/tls/certs/server.crt
SSLCertificateKeyFile /etc/pki/tls/private/server.key
ServerName example.com
ServerAdmin root@example.com
DocumentRoot /var/www/html/
</VirtualHost>
<Directory /var/www/html/>
<RequireAll>
Require all granted
</RequireAll>
</Directory>
:wq

systemctl restart httpd

Access Your Web Server

Visit your public IP in browser:

👉 **https://13.126.179.190**



Step7 . Redirect all client traffic on https protocol

Edit configuration files

#vim /etc/httpd/conf.d/vhosts.conf

<VirtualHost *:443>
SSLEngine on
SSLCertificateFile /etc/pki/tls/certs/server.crt
SSLCertificateKeyFile /etc/pki/tls/private/server.key
ServerName 13.126.179.190
ServerAdmin root@example.com
DocumentRoot /var/www/html/
</VirtualHost>
<Directory /var/www/html/>
<RequireAll>
Require all granted
</RequireAll>
</Directory>
<VirtualHost *:80>
ServerName test.com
Redirect / https://13.126.179.190
</VirtualHost>

:wq

systemctl restart httpd

Visit your public IP in browser:

👉 http://13.126.179.190

Note: if we type http:// 13.126.179.190 only site by default redirect us on https

Troubleshooting (quick)

- Blank page / cannot connect → re-check Security Group inbound rules and EC2 public IP.
- Service failed → check logs: `sudo journalctl -u httpd --since "1 hour ago"` and `/var/log/httpd/error_log`.

✅ Project Completed

Apache Web Server successfully deployed on EC2 as part of DevOps/Linux practice by Shabbir Ahmad……..Thnak You !!