

TASK:

Configuring Apache2 to serve content over HTTPS using self-signed SSL Certificate

Step 1: Configuring Firewall to open both port 80 and 443 and allow incoming HTTP/HTTPS:

- UFW (Uncomplicated Firewall) is a user-friendly command-line tool for managing firewall rules on Linux systems. It provides a simplified interface to configure and manage firewall settings, making it easier for users to secure their systems and control network traffic.

Installing UFW package:

-> `sudo apt install`

List all UFW application:

-> `sudo ufw app list`

Allow Apache Full:

- Allows both HTTP (port 80) and HTTPS (port 443)

-> `sudo ufw allow 'Apache Full'`

Enable UFW Rules:

-> `sudo ufw enable`

```
aditya@aditya-ubuntu:~$ sudo ufw status
Status: active

To Action From
--
Apache Secure ALLOW Anywhere
Apache Full ALLOW Anywhere
Apache Secure (v6) ALLOW Anywhere (v6)
Apache Full (v6) ALLOW Anywhere (v6)
```

Step 2: Generating a self-signed SSL Certificate:

- SSL provides a secure and encrypted connection between a client and a server. It encrypts the data exchanged between the client and server, making it unreadable to anyone who may intercept it. Encryption prevents unauthorized access to sensitive information, such as personal data, login credentials, or financial details.

To generate SSL an OpenSSL package is needed:

-> `sudo apt install openssl`

To generate certificate.csr and private.key:

-> `openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout /var/www/private.key -out /var/www/certificate.crt`

```
aditya@aditya-ubuntu:/var/www/html$ sudo openssl req -new -key private.key -out
certificate.csr
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:IN
State or Province Name (full name) [Some-State]:Karnataka
Locality Name (eg, city) []:Bangalore
Organization Name (eg, company) [Internet Widgits Pty Ltd]:Aditya test
Organizational Unit Name (eg, section) []:it
Common Name (e.g. server FQDN or YOUR name) []:aditya
Email Address []:adityar947@gmail.com
```

Step 3: Configure Virtual Hosts for HTTPS:

- Open virtual host configuration, add the following directives to enable HTTPS and specify the paths to the SSL certificate and private key files

-> `sudo nano /etc/apache2/sites-available/greet.conf`

<VirtualHost *:443>

ServerName greet1.com

DocumentRoot /var/www/html/greet

SSLEngine On

SSLCertificateFile /var/www/ssl/certificate.crt

SSLCertificateKeyFile /var/www/ssl/private.key

</VirtualHost>

Step 4: Redirect HTTP to HTTPS:

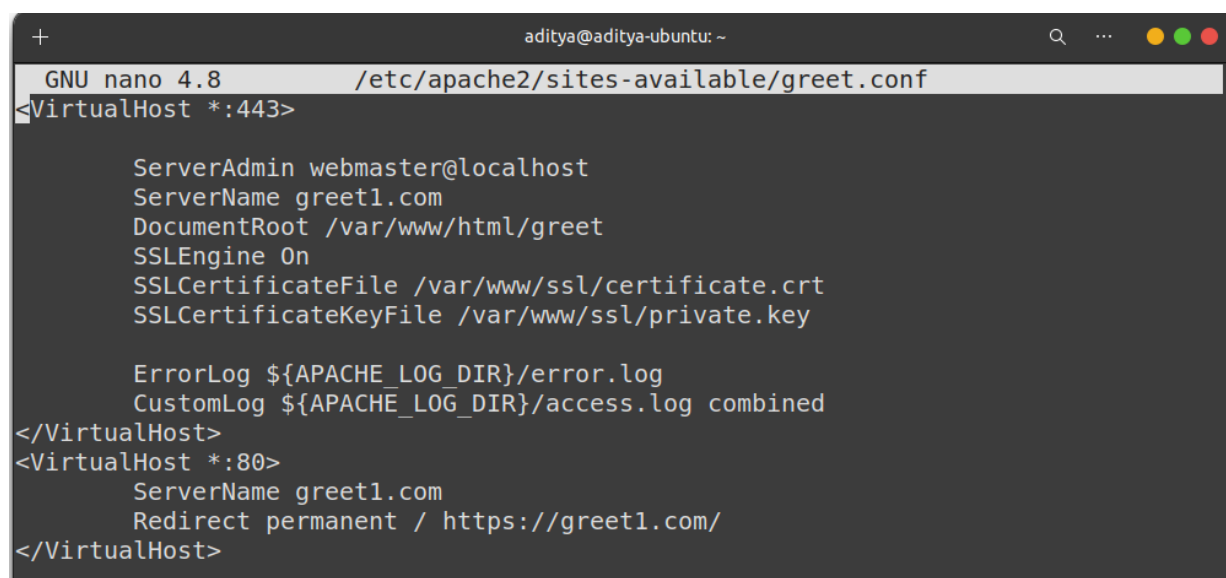
- **Add a separate virtual host configuration for port 80 that redirects to the corresponding HTTPS URL.**

<VirtualHost *:80>

ServerName greet1.com

Redirect permanent / https://greet1.com/

</VirtualHost>



```
GNU nano 4.8 /etc/apache2/sites-available/greet.conf
<VirtualHost *:443>

    ServerAdmin webmaster@localhost
    ServerName greet1.com
    DocumentRoot /var/www/html/greet
    SSLEngine On
    SSLCertificateFile /var/www/ssl/certificate.crt
    SSLCertificateKeyFile /var/www/ssl/private.key

    ErrorLog ${APACHE_LOG_DIR}/error.log
    CustomLog ${APACHE_LOG_DIR}/access.log combined
</VirtualHost>
<VirtualHost *:80>
    ServerName greet1.com
    Redirect permanent / https://greet1.com/
</VirtualHost>
```

Step 5: Enabling virtualhost configuration file:

-> `sudo a2ensite greet.conf`

Adding ip and ServerName:

-> `sudo nano /etc/hosts`

```
192.168.244.44 greet1.com
```

Disabling 000-default.conf:

-> `sudo a2dissite 000-default.conf`

Step 6: Enabling SSL Module:

-> `sudo a2enmod ssl`

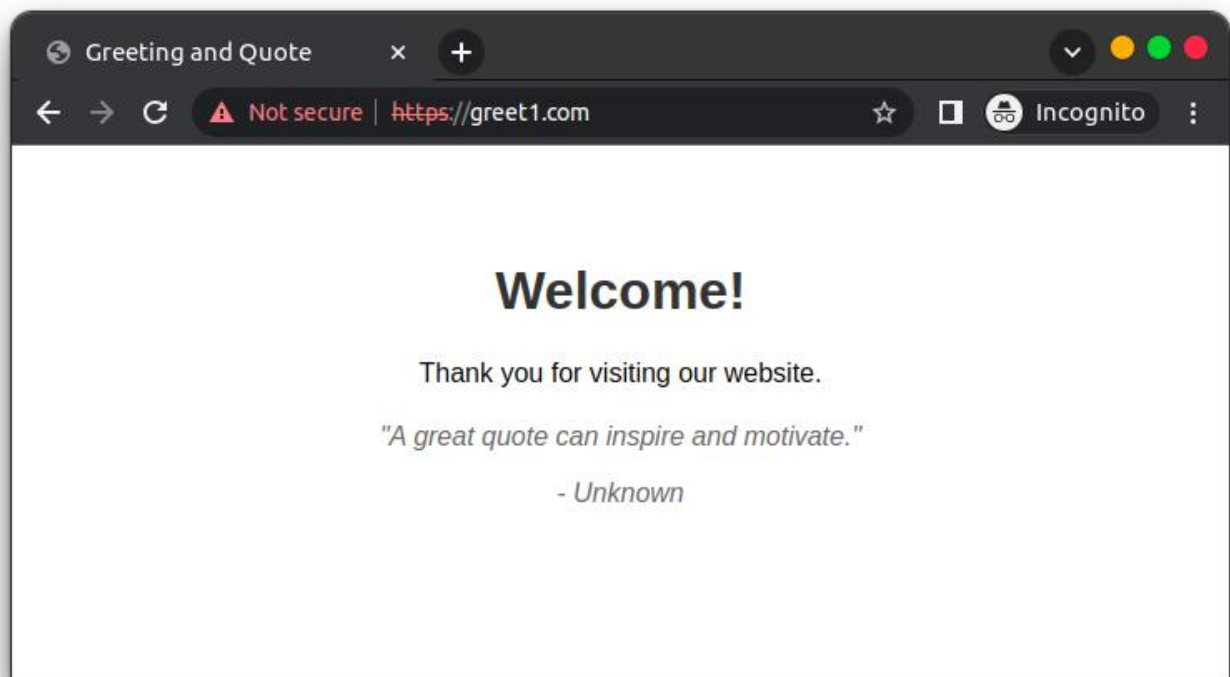
```
aditya@aditya-ubuntu:~$ sudo a2enmod ssl
Considering dependency setenvif for ssl:
Module setenvif already enabled
Considering dependency mime for ssl:
Module mime already enabled
Considering dependency socache_shmcb for ssl:
Module socache_shmcb already enabled
Module ssl already enabled
```

Restarting the apache2:

-> `sudo systemctl restart apache2`

Step 7: Test HTTPS Connections:

- Open a web browser and navigate to <http://greet1.com>
- Ignore any browser warnings about the self-signed certificate.
- Verify that the website is successfully served over HTTPS and that the SSL certificate is working correctly.



Welcome!

Thank you for visiting our website.

"A great quote can inspire and motivate."

- Unknown