Lab Report: Securing Apache Web Server
Lab Task 5

Md Sadman Hafiz-2018831057

Introduction

This lab focuses on setting up a secure web server using Apache and digital certificates. The main objectives are to become a certificate authority, create digital certificates, and deploy HTTPS into the Apache web server. The tasks are divided into several checkpoints to ensure the proper setup and functioning of the secure web server.

1. Setting up the prerequisites:

This step ensures that OpenSSL is installed and sets up the necessary configuration files and directories for creating certificates.

```
          <p>
                Please use the <tt>ubuntu-bug</tt> tool to report bugs in the
                Apache2 package with Ubuntu. However, check <a
                href="https://bugs.launchpad.net/ubuntu/+source/apache2"
                rel="nofollow">existing bug reports</a> before reporting a new bug.
          </p>
          <p>
                Please report bugs specific to modules (such as PHP and others)
                to their respective packages, not to the web server itself.
          </p>
        </div>

      </div>
    </div>
    <div class="validator">
    </div>
  </body>
</html>
hafiz@hafiz-VirtualBox:/var/www/html$ cd
hafiz@hafiz-VirtualBox:~$ sudo mkdir -p /var/www/example.com/html
hafiz@hafiz-VirtualBox:~$ sudo chown -R $USER:$USER
chown: missing operand after ‘hafiz:hafiz’
Try 'chown --help' for more information.
hafiz@hafiz-VirtualBox:~$ sudo chown -R $hafiz:$hafiz
chown: missing operand after ‘:’
Try 'chown --help' for more information.
hafiz@hafiz-VirtualBox:~$ sudo chown -R $hafiz:$hafiz
chown: missing operand after ‘:’
Try 'chown --help' for more information.
hafiz@hafiz-VirtualBox:~$
```

2. Setting up usernames:

```
hafiz@hafiz-VirtualBox:~$ sudo chown -R $hafiz:$hafiz
chown: missing operand after ':'
Try 'chown --help' for more information.
hafiz@hafiz-VirtualBox:~$ sudo chown -R hafiz:hafiz
chown: missing operand after 'hafiz:hafiz'
Try 'chown --help' for more information.
hafiz@hafiz-VirtualBox:~$ sudo chown -R $USER:$USER
chown: missing operand after 'hafiz:hafiz'
Try 'chown --help' for more information.
hafiz@hafiz-VirtualBox:~$ sudo chown -R $USER:$USER /var/example.com/html
chown: cannot access '/var/example.com/html': No such file or directory
hafiz@hafiz-VirtualBox:~$ sudo chown -R $USER:$USER/var/example.com/html
chown: missing operand after 'hafiz:hafiz/var/example.com/html'
Try 'chown --help' for more information.
hafiz@hafiz-VirtualBox:~$ sudo chown -R $USER:$USER /var/example.com/html
chown: cannot access '/var/example.com/html': No such file or directory
hafiz@hafiz-VirtualBox:~$ ls
Desktop  Documents  Downloads  Music  Pictures  Public  snap  Templates  Videos
hafiz@hafiz-VirtualBox:~$ cd
hafiz@hafiz-VirtualBox:~$ cd ..
hafiz@hafiz-VirtualBox:/home$ cd ..
hafiz@hafiz-VirtualBox:/$ ls
bin                cdrom  home   lib.usr-is-merged  mnt   root  sbin.usr-is-merged  swap.img  usr
bin.usr-is-merged  dev    lib    lost+found         opt   run   snap                sys       var
boot               etc    lib64  media              proc  sbin  srv                 tmp
\hafiz@hafiz-VirtualBox:/$sudo chown -R $USER:$USER /var/www/example.com/html
hafiz@hafiz-VirtualBox:/$ sudo chmod -R 755 /var/www/example.com
hafiz@hafiz-VirtualBox:/$ cd /var/www/example.com
hafiz@hafiz-VirtualBox:/var/www/example.com$ sudo nano html/index.html
hafiz@hafiz-VirtualBox:/var/www/example.com$
```

3.Checking the server:

```
\hafiz@hafiz-VirtualBox:/$sudo chown -R $USER:$USER /var/www/example.com/html
hafiz@hafiz-VirtualBox:/$ sudo chmod -R 755 /var/www/example.com
hafiz@hafiz-VirtualBox:/$ cd /var/www/example.com
hafiz@hafiz-VirtualBox:/var/www/example.com$ sudo nano html/index.html
hafiz@hafiz-VirtualBox:/var/www/example.com$ cd

d: command not found
hafiz@hafiz-VirtualBox:/var/www/example.com$ ^[[200~sudo nano /etc/apache2/sites-available/example.com.conf
sudo: command not found
hafiz@hafiz-VirtualBox:/var/www/example.com$ sudo nano /etc/apache2/sites-available/example.com.conf
hafiz@hafiz-VirtualBox:/var/www/example.com$ sudo a2ensite example.com.conf
Enabling site example.com.
To activate the new configuration, you need to run:
  systemctl reload apache2
hafiz@hafiz-VirtualBox:/var/www/example.com$ sudo a2dissite 000-default.conf
Site 000-default disabled.
To activate the new configuration, you need to run:
  systemctl reload apache2
hafiz@hafiz-VirtualBox:/var/www/example.com$ sudo systemctl reload apache2
hafiz@hafiz-VirtualBox:/var/www/example.com$ sudo apache2ctl configtest
AH00558: apache2: Could not reliably determine the server's fully qualified domain name, using 127.0.1.1. Set the 'Serve
rName' directive globally to suppress this message
Syntax OK
hafiz@hafiz-VirtualBox:/var/www/example.com$ sudo systemctl restart apache2
hafiz@hafiz-VirtualBox:/var/www/example.com$ curl http://webserverlab.com
```

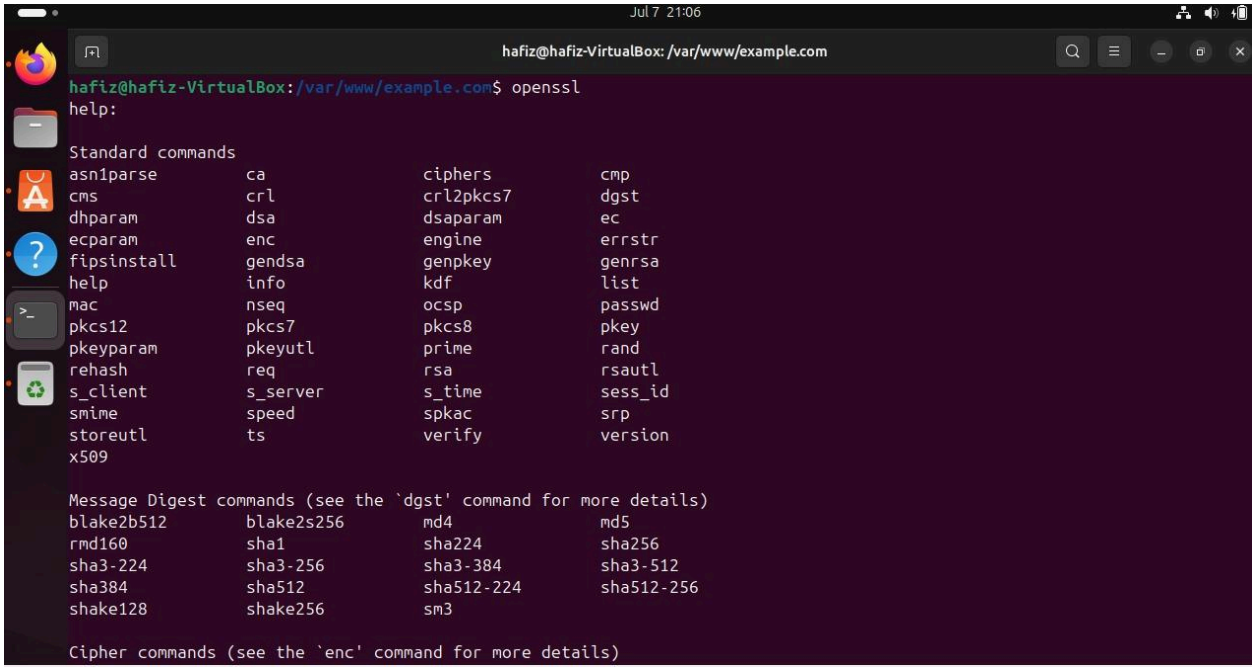
4.Not secure website: http:// example.com

5.Not secure website:http:// webserverlab.com:

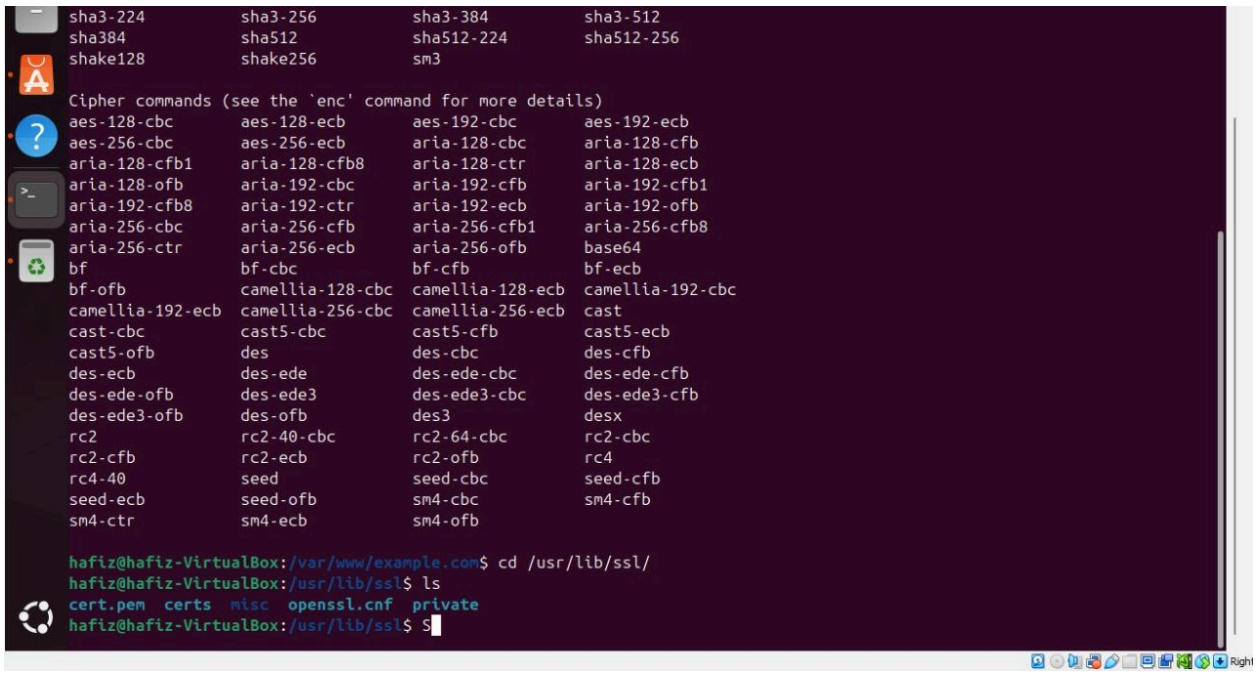
Task 1: Becoming a Certificate Authority

Step 1: Checking OpenSSL and Configuring the File

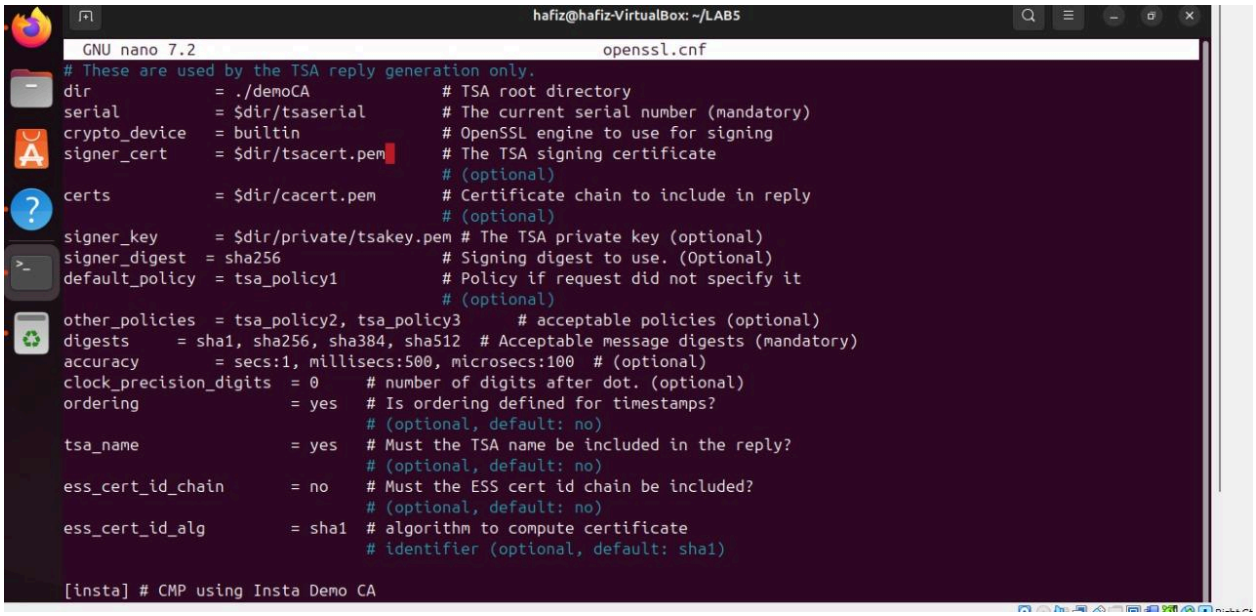
Checking openssl: This step ensures that OpenSSL is installed and sets up the necessary configuration files and directories for creating **certificates.**

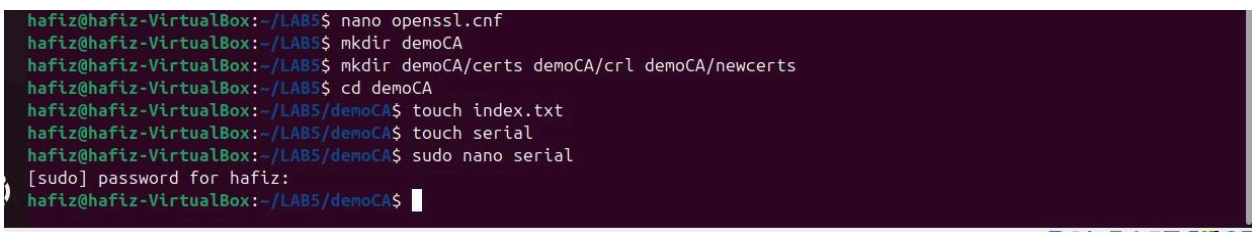


Checking if the ssl.configuaration file exists:



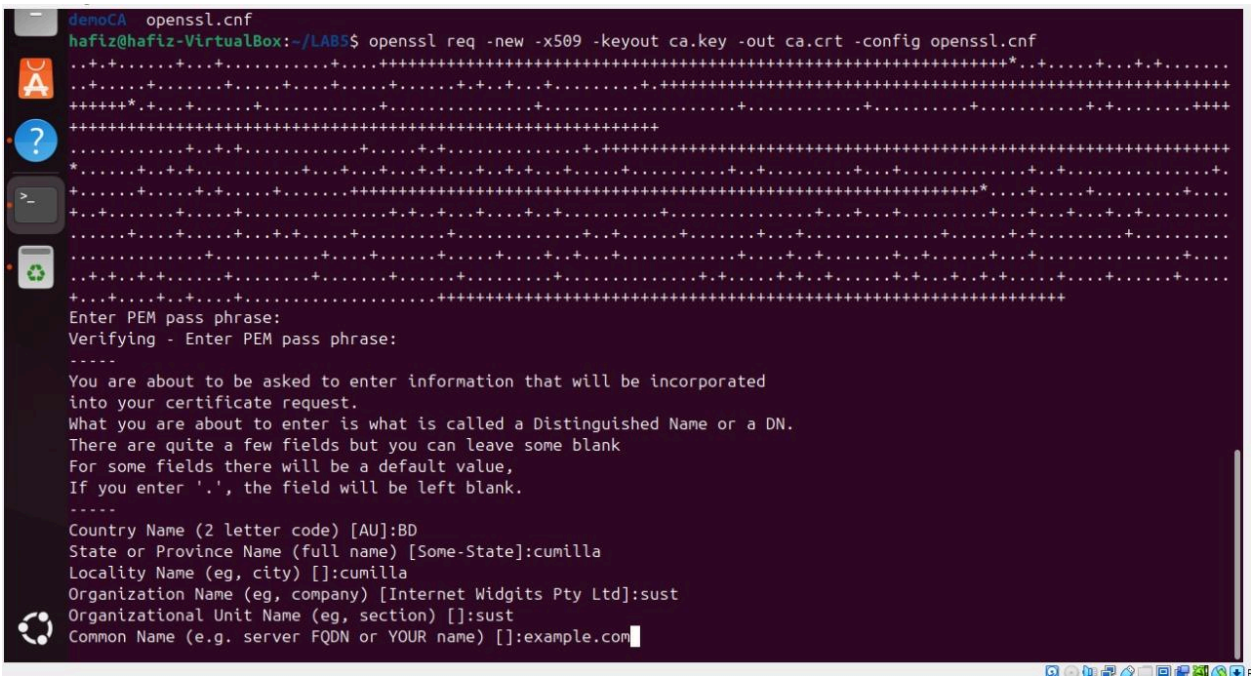
Configure file:





Step 2: Generating a Self-Signed Certificate for the CA

This command generates a self-signed certificate for the CA, which will be used to sign other certificates.



This process involves generating a public/private key pair, creating a Certificate Signing Request (CSR), and signing the CSR with the CA's certificate to generate a certificate for example.com.

```
Verifying - Enter PEM pass phrase:
hafiz@hafiz-VirtualBox:~/LAB5$ openssl req -new -key server.key -out server.csr -config openssl.cnf
Enter pass phrase for server.key:
Could not read private key from server.key
4057D58B68720000:error:1608010C:STORE routines:ossl_store_handle_load_result:unsupported:../crypto/store/store_result.c:
151:
4057D58B68720000:error:1C800064:Provider routines:ossl_cipher_unpadblock:bad decrypt:../providers/implementations/cipher
s/ciphercommon_block.c:124:
4057D58B68720000:error:11800074:PKCS12 routines:PKCS12_pbe_crypt_ex:pkcs12 cipherfinal error:../crypto/pkcs12/p12_decr.c
:86:maybe wrong password
hafiz@hafiz-VirtualBox:~/LAB5$ hafiz
hafiz: command not found
hafiz@hafiz-VirtualBox:~/LAB5$ openssl req -new -key server.key -out server.csr -config openssl.cnf
Enter pass phrase for server.key:
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]:BD
State or Province Name (full name) [Some-State]:cumilla
Locality Name (eg, city) []:cumilla
Organization Name (eg, company) [Internet Widgits Pty Ltd]:sust
Organizational Unit Name (eg, section) []:sust
Common Name (e.g. server FQDN or YOUR name) []:example.com
Email Address []:
```

```
Check that the request matches the signature
Signature ok
Certificate Details:
        Serial Number: 4096 (0x1000)
        Validity
            Not Before: Jul  7 15:29:09 2024 GMT
            Not After : Jul  7 15:29:09 2025 GMT
        Subject:
            countryName               = BD
            stateOrProvinceName       = cumilla
            organizationName          = sust
            organizationalUnitName    = sust
            commonName                = example.com
        X509v3 extensions:
            X509v3 Basic Constraints:
                CA:FALSE
            X509v3 Subject Key Identifier:
                E5:FA:B7:C2:CD:35:14:53:E4:CA:77:7D:8F:82:17:62:DA:13:88:21
            X509v3 Authority Key Identifier:
                3F:C4:51:8C:98:38:81:11:FF:D3:63:51:0C:FC:A6:44:81:01:81:C9
Certificate is to be certified until Jul  7 15:29:09 2025 GMT (365 days)
Sign the certificate? [y/n]:y


1 out of 1 certificate requests certified, commit? [y/n]y
Write out database with 1 new entries
Database updated
hafiz@hafiz-VirtualBox:~/LAB5$ cp server.key server.pem
hafiz@hafiz-VirtualBox:~/LAB5$ cat server.crt >> server.pem
hafiz@hafiz-VirtualBox:~/LAB5$
```

```
hafiz@hafiz-VirtualBox:~/LAB5$ openssl ca -in server.csr -out server.crt -cert ca.crt -keyfile ca.key -config openssl.cn
f
Using configuration from openssl.cnf
Enter pass phrase for ca.key:
Check that the request matches the signature
Signature ok
Certificate Details:
        Serial Number: 4096 (0x1000)
        Validity
            Not Before: Jul  7 15:29:09 2024 GMT
            Not After : Jul  7 15:29:09 2025 GMT
        Subject:
            countryName               = BD
            stateOrProvinceName       = cumilla
            organizationName          = sust
            organizationalUnitName    = sust
            commonName                = example.com
        X509v3 extensions:
            X509v3 Basic Constraints:
                CA:FALSE
            X509v3 Subject Key Identifier:
                E5:FA:B7:C2:CD:35:14:53:E4:CA:77:7D:8F:82:17:62:DA:13:88:21
            X509v3 Authority Key Identifier:
                3F:C4:51:8C:98:38:81:11:FF:D3:63:51:0C:FC:A6:44:81:01:81:C9
Certificate is to be certified until Jul  7 15:29:09 2025 GMT (365 days)
Sign the certificate? [y/n]:y
```

```
1 out of 1 certificate requests certified, commit? [y/n]y
Write out database with 1 new entries
Database updated
hafiz@hafiz-VirtualBox:~/LAB5$ cp server.key server.pem
hafiz@hafiz-VirtualBox:~/LAB5$ cat server.crt >> server.pem
hafiz@hafiz-VirtualBox:~/LAB5$ openssl s_server -cert server.pem -www
Enter pass phrase for server.pem:
Enter pass phrase for server.pem:
Using default temp DH parameters
ACCEPT
```

Step 4: Launching the Web Server with the Generated Certificate

This command launches a simple web server using the generated certificate. The server can be accessed via `https://example.com:4433`.

<u>Checkpoint-1:</u>

https://example.com:4433

```
s_server -cert server.pem -www
Secure Renegotiation IS NOT supported
Ciphers supported in s_server binary
TLSv1.3    :TLS_AES_256_GCM_SHA384    TLSv1.3    :TLS_CHACHA20_POLY1305_SHA256
TLSv1.3    :TLS_AES_128_GCM_SHA256    TLSv1.2    :ECDHE-ECDSA-AES256-GCM-SHA384
TLSv1.2    :ECDHE-RSA-AES256-GCM-SHA384 TLSv1.2    :DHE-RSA-AES256-GCM-SHA384
TLSv1.2    :ECDHE-ECDSA-CHACHA20-POLY1305 TLSv1.2    :ECDHE-RSA-CHACHA20-POLY1305
TLSv1.2    :DHE-RSA-CHACHA20-POLY1305 TLSv1.2    :ECDHE-ECDSA-AES128-GCM-SHA256
TLSv1.2    :ECDHE-RSA-AES128-GCM-SHA256 TLSv1.2    :DHE-RSA-AES128-GCM-SHA256
TLSv1.2    :ECDHE-ECDSA-AES256-SHA384 TLSv1.2    :ECDHE-RSA-AES256-SHA384
TLSv1.2    :DHE-RSA-AES256-SHA256     TLSv1.2    :ECDHE-ECDSA-AES128-SHA256
TLSv1.2    :ECDHE-RSA-AES128-SHA256   TLSv1.2    :DHE-RSA-AES128-SHA256
TLSv1.0    :ECDHE-ECDSA-AES256-SHA    TLSv1.0    :ECDHE-RSA-AES256-SHA
SSLv3      :DHE-RSA-AES256-SHA        TLSv1.0    :ECDHE-ECDSA-AES128-SHA
TLSv1.0    :ECDHE-RSA-AES128-SHA      SSLv3      :DHE-RSA-AES128-SHA
TLSv1.2    :RSA-PSK-AES256-GCM-SHA384 TLSv1.2    :DHE-PSK-AES256-GCM-SHA384
TLSv1.2    :RSA-PSK-CHACHA20-POLY1305 TLSv1.2    :DHE-PSK-CHACHA20-POLY1305
TLSv1.2    :ECDHE-PSK-CHACHA20-POLY1305 TLSv1.2    :AES256-GCM-SHA384
TLSv1.2    :PSK-AES256-GCM-SHA384     TLSv1.2    :PSK-CHACHA20-POLY1305
TLSv1.2    :RSA-PSK-AES128-GCM-SHA256 TLSv1.2    :DHE-PSK-AES128-GCM-SHA256
TLSv1.2    :AES128-GCM-SHA256         TLSv1.2    :PSK-AES128-GCM-SHA256
TLSv1.2    :AES256-SHA256             TLSv1.2    :AES128-SHA256
TLSv1.0    :ECDHE-PSK-AES256-CBC-SHA384 TLSv1.0    :ECDHE-PSK-AES256-CBC-SHA
SSLv3      :SRP-RSA-AES-256-CBC-SHA   SSLv3      :SRP-AES-256-CBC-SHA
TLSv1.0    :RSA-PSK-AES256-CBC-SHA384 TLSv1.0    :DHE-PSK-AES256-CBC-SHA384
SSLv3      :RSA-PSK-AES256-CBC-SHA    SSLv3      :DHE-PSK-AES256-CBC-SHA
SSLv3      :AES256-SHA                TLSv1.0    :PSK-AES256-CBC-SHA384
SSLv3      :PSK-AES256-CBC-SHA        TLSv1.0    :ECDHE-PSK-AES128-CBC-SHA256
TLSv1.0    :ECDHE-PSK-AES128-CBC-SHA  SSLv3      :SRP-RSA-AES-128-CBC-SHA
SSLv3      :SRP-AES-128-CBC-SHA       TLSv1.0    :RSA-PSK-AES128-CBC-SHA256
TLSv1.0    :DHE-PSK-AES128-CBC-SHA256 SSLv3      :RSA-PSK-AES128-CBC-SHA
SSLv3      :DHE-PSK-AES128-CBC-SHA    SSLv3      :AES128-SHA
TLSv1.0    :PSK-AES128-CBC-SHA256     SSLv3      :PSK-AES128-CBC-SHA
---
Ciphers common between both SSL end points:
TLS_AES_128_GCM_SHA256     TLS_CHACHA20_POLY1305_SHA256 TLS_AES_256_GCM_SHA384
```

Checkpoint-2:

```
s_server -cert server.pem -www
Secure Renegotiation IS NOT supported
Ciphers supported in s_server binary
TLSv1.3    :TLS_AES_256_GCM_SHA384    TLSv1.3    :TLS_CHACHA20_POLY1305_SHA256
TLSv1.3    :TLS_AES_128_GCM_SHA256    TLSv1.2    :ECDHE-ECDSA-AES256-GCM-SHA384
TLSv1.2    :ECDHE-RSA-AES256-GCM-SHA384 TLSv1.2    :DHE-RSA-AES256-GCM-SHA384
TLSv1.2    :ECDHE-ECDSA-CHACHA20-POLY1305 TLSv1.2    :ECDHE-RSA-CHACHA20-POLY1305
TLSv1.2    :DHE-RSA-CHACHA20-POLY1305 TLSv1.2    :ECDHE-ECDSA-AES128-GCM-SHA256
TLSv1.2    :ECDHE-RSA-AES128-GCM-SHA256 TLSv1.2    :DHE-RSA-AES128-GCM-SHA256
TLSv1.2    :ECDHE-ECDSA-AES256-SHA384 TLSv1.2    :ECDHE-RSA-AES256-SHA384
TLSv1.2    :DHE-RSA-AES256-SHA256     TLSv1.2    :ECDHE-ECDSA-AES128-SHA256
TLSv1.2    :ECDHE-RSA-AES128-SHA256   TLSv1.2    :DHE-RSA-AES128-SHA256
TLSv1.0    :ECDHE-ECDSA-AES256-SHA    TLSv1.0    :ECDHE-RSA-AES256-SHA
SSLv3      :DHE-RSA-AES256-SHA        TLSv1.0    :ECDHE-ECDSA-AES128-SHA
TLSv1.0    :ECDHE-RSA-AES128-SHA      SSLv3      :DHE-RSA-AES128-SHA
TLSv1.2    :RSA-PSK-AES256-GCM-SHA384 TLSv1.2    :DHE-PSK-AES256-GCM-SHA384
TLSv1.2    :RSA-PSK-CHACHA20-POLY1305 TLSv1.2    :DHE-PSK-CHACHA20-POLY1305
TLSv1.2    :ECDHE-PSK-CHACHA20-POLY1305 TLSv1.2    :AES256-GCM-SHA384
TLSv1.2    :PSK-AES256-GCM-SHA384     TLSv1.2    :PSK-CHACHA20-POLY1305
TLSv1.2    :RSA-PSK-AES128-GCM-SHA256 TLSv1.2    :DHE-PSK-AES128-GCM-SHA256
TLSv1.2    :AES128-GCM-SHA256         TLSv1.2    :PSK-AES128-GCM-SHA256
TLSv1.2    :AES256-SHA256             TLSv1.2    :AES128-SHA256
TLSv1.0    :ECDHE-PSK-AES256-CBC-SHA384 TLSv1.0    :ECDHE-PSK-AES256-CBC-SHA
SSLv3      :SRP-RSA-AES-256-CBC-SHA   SSLv3      :SRP-AES-256-CBC-SHA
TLSv1.0    :RSA-PSK-AES256-CBC-SHA384 TLSv1.0    :DHE-PSK-AES256-CBC-SHA384
SSLv3      :RSA-PSK-AES256-CBC-SHA    SSLv3      :DHE-PSK-AES256-CBC-SHA
SSLv3      :AES256-SHA                TLSv1.0    :PSK-AES256-CBC-SHA384
SSLv3      :PSK-AES256-CBC-SHA        TLSv1.0    :ECDHE-PSK-AES128-CBC-SHA256
TLSv1.0    :ECDHE-PSK-AES128-CBC-SHA  SSLv3      :SRP-RSA-AES-128-CBC-SHA
SSLv3      :SRP-AES-128-CBC-SHA       TLSv1.0    :RSA-PSK-AES128-CBC-SHA256
TLSv1.0    :DHE-PSK-AES128-CBC-SHA256 SSLv3      :RSA-PSK-AES128-CBC-SHA
SSLv3      :DHE-PSK-AES128-CBC-SHA    SSLv3      :AES128-SHA
TLSv1.0    :PSK-AES128-CBC-SHA256     SSLv3      :PSK-AES128-CBC-SHA
---
Ciphers common between both SSL end points:
TLS_AES_128_GCM_SHA256     TLS_CHACHA20_POLY1305_SHA256 TLS_AES_256_GCM_SHA384
ECDHE-ECDSA-AES128-GCM-SHA256 ECDHE-RSA-AES128-GCM-SHA256 ECDHE-ECDSA-CHACHA20-POLY1305
ECDHE-RSA-CHACHA20-POLY1305 ECDHE-ECDSA-AES256-GCM-SHA384 ECDHE-RSA-AES256-GCM-SHA384
ECDHE-ECDSA-AES256-SHA     ECDHE-ECDSA-AES128-SHA     ECDHE-RSA-AES128-SHA
ECDHE-RSA-AES256-SHA       AES128-GCM-SHA256          AES256-GCM-SHA384
AES128-SHA                 AES256-SHA
Signature Algorithms: ECDSA+SHA256:ECDSA+SHA384:ECDSA+SHA512:RSA-PSS+SHA256:RSA-PSS+SHA384:RSA-PSS+SHA512:RSA+SHA256:RSA+SHA384:RSA+SHA512:ECDSA+SHA1:RSA+SHA1
Shared Signature Algorithms: ECDSA+SHA256:ECDSA+SHA384:ECDSA+SHA512:RSA-PSS+SHA256:RSA-PSS+SHA384:RSA-PSS+SHA512:RSA+SHA256:RSA+SHA384:RSA+SHA512
Supported groups: x25519:secp256r1:secp384r1:secp521r1:ffdhe2048:ffdhe3072
```

Task 2: Deploy HTTPS into Apache

```
hafiz@hafiz-VirtualBox:~/certs$ openssl req -x509 -new -nodes -key myCA.key -sha256 -days 1825 -out myCA.pem
Enter pass phrase for myCA.key:
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]:BD
State or Province Name (full name) [Some-State]:cumilla
Locality Name (eg, city) []:cumilla
Organization Name (eg, company) [Internet Widgits Pty Ltd]:sust
Organizational Unit Name (eg, section) []:sust
Common Name (e.g. server FQDN or YOUR name) []:example.com
Email Address []:
hafiz@hafiz-VirtualBox:~/certs$ sudo apt-get install -y ca-certificates
[sudo] password for hafiz:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
ca-certificates is already the newest version (20240203).
ca-certificates set to manually installed.
0 upgraded, 0 newly installed, 0 to remove and 62 not upgraded.
hafiz@hafiz-VirtualBox:~/certs$
```

6.

```
Country Name (2 letter code) [AU]:BD
State or Province Name (full name) [Some-State]:cumilla
Locality Name (eg, city) []:cumilla
Organization Name (eg, company) [Internet Widgits Pty Ltd]:sust
Organizational Unit Name (eg, section) []:sust
Common Name (e.g. server FQDN or YOUR name) []:example.com
Email Address []:
hafiz@hafiz-VirtualBox:~/certs$ sudo apt-get install -y ca-certificates
[sudo] password for hafiz:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
ca-certificates is already the newest version (20240203).
ca-certificates set to manually installed.
0 upgraded, 0 newly installed, 0 to remove and 62 not upgraded.
hafiz@hafiz-VirtualBox:~/certs$ sudo cp ~/certs/myCA.pem /usr/local/share/ca-certificates/myCA.crt
hafiz@hafiz-VirtualBox:~/certs$ sudo update-ca-certificates
Updating certificates in /etc/ssl/certs...
rehash: warning: skipping ca-certificates.crt,it does not contain exactly one certificate or CRL
1 added, 0 removed; done.
Running hooks in /etc/ca-certificates/update.d...
done.
hafiz@hafiz-VirtualBox:~/certs$
```

Right Ctrl

7.

```
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:BD
State or Province Name (full name) [Some-State]:cumilla
Locality Name (eg, city) []:cumilla
Organization Name (eg, company) [Internet Widgits Pty Ltd]:sust
Organizational Unit Name (eg, section) []:sust
Common Name (e.g. server FQDN or YOUR name) []:example.com
Email Address []:
hafiz@hafiz-VirtualBox:~/certs$ sudo apt-get install -y ca-certificates
[sudo] password for hafiz:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
ca-certificates is already the newest version (20240203).
ca-certificates set to manually installed.
0 upgraded, 0 newly installed, 0 to remove and 62 not upgraded.
hafiz@hafiz-VirtualBox:~/certs$ sudo cp ~/certs/myCA.pem /usr/local/share/ca-certificates/myCA.crt
hafiz@hafiz-VirtualBox:~/certs$ sudo update-ca-certificates
Updating certificates in /etc/ssl/certs...
rehash: warning: skipping ca-certificates.crt,it does not contain exactly one certificate or CRL
1 added, 0 removed; done.
Running hooks in /etc/ca-certificates/update.d...
done.
hafiz@hafiz-VirtualBox:~/certs$ awk -v cmd='openssl x509 -noout -subject' '/BEGIN/{close(cmd)};{print | cmd}' < /etc/ssl
/certs/ca-certificates.crt | grep Hellfish

hafiz@hafiz-VirtualBox:~/certs$
hafiz@hafiz-VirtualBox:~/certs$
```

```
hafiz@hafiz-VirtualBox:~/certs$ openssl genrsa -out hellfish.test.key 2048
hafiz@hafiz-VirtualBox:~/certs$ openssl req -new -key hellfish.test.key -out hellfish.test.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]:BD
State or Province Name (full name) [Some-State]:cumilla
Locality Name (eg, city) []:cumilla
Organization Name (eg, company) [Internet Widgits Pty Ltd]:sust
Organizational Unit Name (eg, section) []:sust
Common Name (e.g. server FQDN or YOUR name) []:example.com
Email Address []:

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:hafiz
An optional company name []:
hafiz@hafiz-VirtualBox:~/certs$
```

```
Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:hafiz
An optional company name []:
hafiz@hafiz-VirtualBox:~/certs$ ls
hellfish.test.csr  hellfish.test.key  myCA.key  myCA.pem
hafiz@hafiz-VirtualBox:~/certs$ cd
hafiz@hafiz-VirtualBox:~$ cd ..
hafiz@hafiz-VirtualBox:/home$ cd /etc/apache2/sites-available
hafiz@hafiz-VirtualBox:/etc/apache2/sites-available$ ls
000-default.conf  default-ssl.conf  example.com.conf
hafiz@hafiz-VirtualBox:/etc/apache2/sites-available$
```

Task-3: Deploy HTTPS into Apache:

Step 1: Configuring Apache to Use the Created Certificates

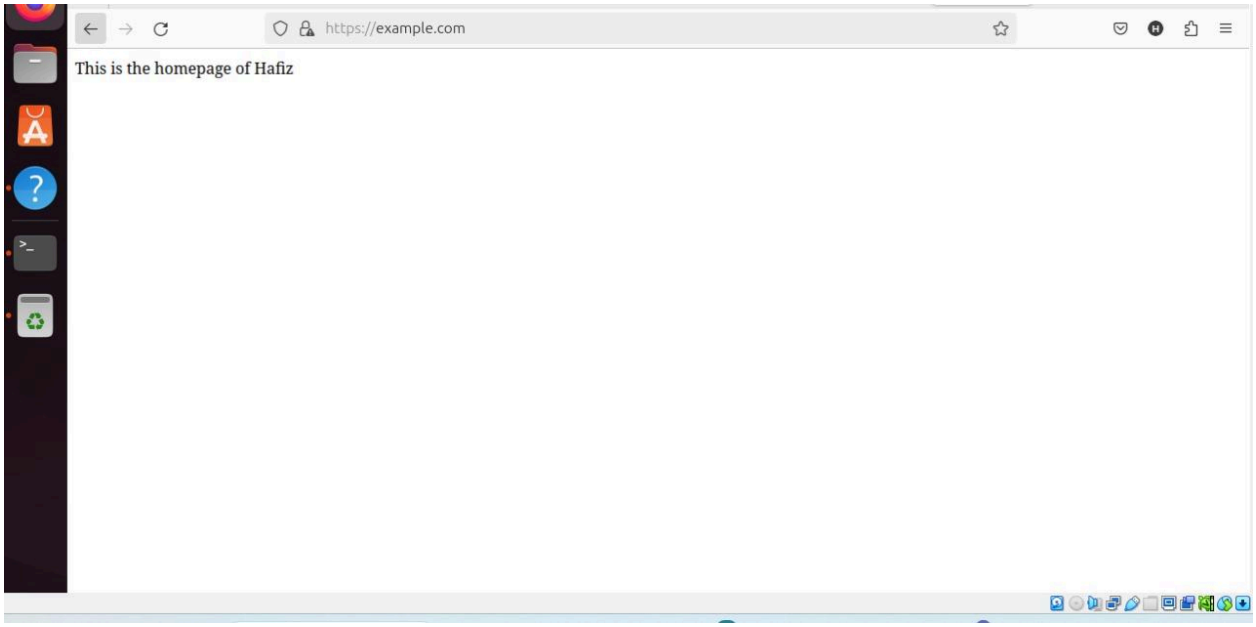
```
hafiz@hafiz-VirtualBox: /etc/apache2/sites-available
GNU nano 7.2                                   example.com.conf *

<VirtualHost *:80>
ServerAdmin admin@example.com
ServerName example.com
ServerAlias www.example.com
DocumentRoot /var/www/example.com/html
ErrorLog ${APACHE_LOG_DIR}/error.log
CustomLog ${APACHE_LOG_DIR}/access.log combined
</VirtualHost>
```

```
DocumentRoot /var/www/example.com/html
ErrorLog ${APACHE_LOG_DIR}/error.log
CustomLog ${APACHE_LOG_DIR}/access.log combined
</VirtualHost>

<VirtualHost _default_:443>
        ServerAdmin admin@example.com
        ServerName example.com

        DocumentRoot /var/www/example.com/html

        ErrorLog ${APACHE_LOG_DIR}/error.log
        CustomLog ${APACHE_LOG_DIR}/access.log combined

        SSLEngine on
        SSLCertificateFile    /etc/ssl/certs/apache-selfsigned.crt
        SSLCertificateKeyFile /etc/ssl/private/apache-selfsigned.key

        <FilesMatch "\.(cgi|shtml|phtml|php)$">
                        SSLOptions +StdEnvVars
        </FilesMatch>
        <Directory /usr/lib/cgi-bin>
                        SSLOptions +StdEnvVars
        </Directory>

    </VirtualHost>
```

9.

```
        CPU: 31ms

Jul 07 23:30:35 hafiz-VirtualBox systemd[1]: Starting apache2.service - The Apache HTTP Server...
Jul 07 23:30:35 hafiz-VirtualBox apachectl[13451]: AH00526: Syntax error on line 24 of /etc/apache2/sites-enabled/examp>
Jul 07 23:30:35 hafiz-VirtualBox apachectl[13451]: Invalid command 'SSLEngine', perhaps misspelled or defined by a modu>
Jul 07 23:30:35 hafiz-VirtualBox systemd[1]: apache2.service: Control process exited, code=exited, status=1/FAILURE
Jul 07 23:30:35 hafiz-VirtualBox systemd[1]: apache2.service: Failed with result 'exit-code'.
Jul 07 23:30:35 hafiz-VirtualBox systemd[1]: Failed to start apache2.service - The Apache HTTP Server.
lines 1-14/14 (END)
hafiz@hafiz-VirtualBox:/etc/apache2/sites-available$ sudo nano /etc/apache2/sites-enabled/example
hafiz@hafiz-VirtualBox:/etc/apache2/sites-available$ sudo nano /etc/apache2/sites-enabled/example.com.conf
hafiz@hafiz-VirtualBox:/etc/apache2/sites-available$ sudo apache2ctl configtest
AH00526: Syntax error on line 24 of /etc/apache2/sites-enabled/example.com.conf:
Invalid command 'SSLEngine', perhaps misspelled or defined by a module not included in the server configuration
hafiz@hafiz-VirtualBox:/etc/apache2/sites-available$ sudo a2enmod ssl
Considering dependency mime for ssl:
Module mime already enabled
Considering dependency socache_shmcb for ssl:
Enabling module socache_shmcb.
Enabling module ssl.
See /usr/share/doc/apache2/README.Debian.gz on how to configure SSL and create self-signed certificates.
To activate the new configuration, you need to run:
  systemctl restart apache2
hafiz@hafiz-VirtualBox:/etc/apache2/sites-available$ systemctl restart apache2
hafiz@hafiz-VirtualBox:/etc/apache2/sites-available$ sudo apache2ctl configtest
AH00558: apache2: Could not reliably determine the server's fully qualified domain name, using 127.0.1.1. Set the 'Serve
rName' directive globally to suppress this message
Syntax OK
hafiz@hafiz-VirtualBox:/etc/apache2/sites-available$ sudo nano /etc/apache2/sites-enabled/example.com.conf
hafiz@hafiz-VirtualBox:/etc/apache2/sites-available$
```

Checkpoint-3:



Checkpoint-4: