Lab 5: Securing Apache Web Server

Objectives:

1.The main objective of the lab was to set up a secure Apache web server using digital certificates to enable encrypted communication over HTTPS.

2.Acted as a local Certificate Authority (CA) by generating a root certificate and private key.

3.Used the CA to issue and sign server certificates for specific domains.

4.Verified HTTPS connections using the OpenSSL test server, ensuring authenticity and data integrity.

5.Deployed HTTPS in Apache by configuring SSL virtual hosts for multiple domains.

6.Enabled secure access to domains such as `example.com` and `webserverlab.com`.

7.Documented each step with screenshots under corresponding checkpoints to provide evidence of successful implementation.

Environmental setup

OS: Ubuntu (Desktop/Server)
Packages: apache2, openssl
Browser: Chrome (for CA import and HTTPS test)

Prerequisites (HTTP virtual hosts working)

For Install Apache

-sudo apt update

-sudo apt install apache2 -y

Enable the Apache SSL module

-sudo a2enmod ssl

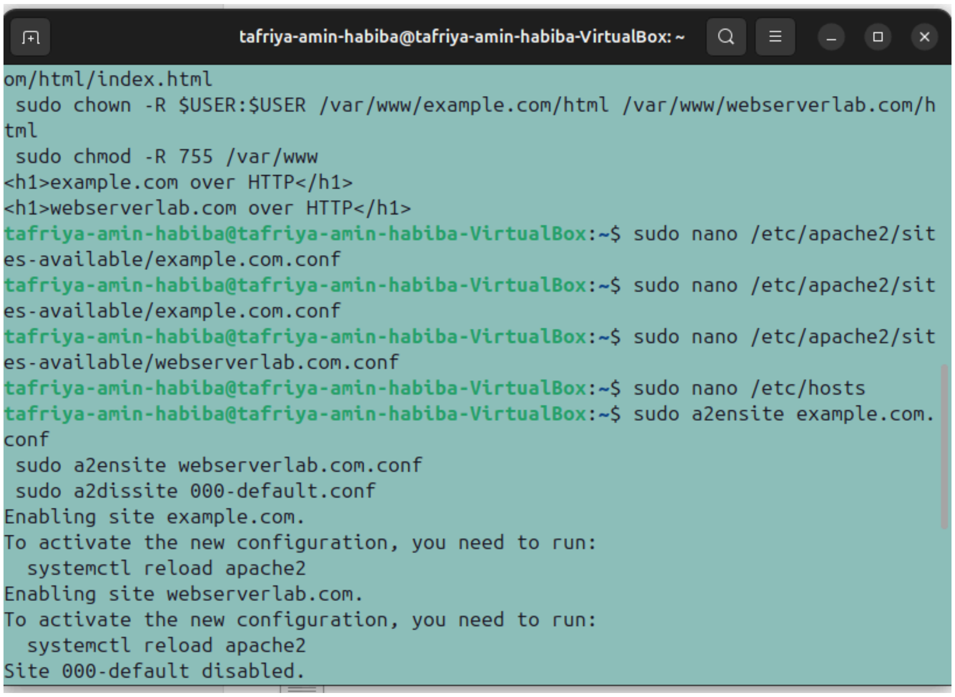
-sudo systemctl restart apache2

Create document roots and simple index pages

-sudo mkdir -p /var/www/example.com/html
-sudo mkdir -p /var/www/webserverlab.com/html
-echo "<h1>example.com over HTTP</h1>" | sudo tee /var/www/example.com/html/index.html
-echo "<h1>webserverlab.com over HTTP</h1>" | sudo tee /var/www/webserverlab.com/html/index.html
-sudo chown -R $USER:$USER /var/www/example.com/html /var/www/webserverlab.com/html
-sudo chmod -R 755 /var/www

Create HTTP virtual hosts:

-sudo nano /etc/apache2/sites-available/example.com.conf



```
om/html/index.html
 sudo chown -R $USER:$USER /var/www/example.com/html /var/www/webserverlab.com/h
tml
 sudo chmod -R 755 /var/www
<h1>example.com over HTTP</h1>
<h1>webserverlab.com over HTTP</h1>
tafriya-amin-habiba@tafriya-amin-habiba-VirtualBox:~$ sudo nano /etc/apache2/sit
es-available/example.com.conf
tafriya-amin-habiba@tafriya-amin-habiba-VirtualBox:~$ sudo nano /etc/apache2/sit
es-available/example.com.conf
tafriya-amin-habiba@tafriya-amin-habiba-VirtualBox:~$ sudo nano /etc/apache2/sit
es-available/webserverlab.com.conf
tafriya-amin-habiba@tafriya-amin-habiba-VirtualBox:~$ sudo nano /etc/hosts
tafriya-amin-habiba@tafriya-amin-habiba-VirtualBox:~$ sudo a2ensite example.com.
conf
 sudo a2ensite webserverlab.com.conf
 sudo a2dissite 000-default.conf
Enabling site example.com.
To activate the new configuration, you need to run:
  systemctl reload apache2
Enabling site webserverlab.com.
To activate the new configuration, you need to run:
  systemctl reload apache2
Site 000-default disabled.
```

Add following lines in that file

```
<VirtualHost *:80>
        ServerName example.com
        ServerAlias www.example.com
        DocumentRoot /var/www/example.com/html
        ErrorLog ${APACHE_LOG_DIR}/example_error.log
        CustomLog ${APACHE_LOG_DIR}/example_access.log combined
 </VirtualHost>
```

```
-sudo nano /etc/apache2/sites-available/webserverlab.com.conf
```

```
<VirtualHost *:80>
        ServerName webserverlab.com
        ServerAlias www.webserverlab.com
        DocumentRoot /var/www/webserverlab.com/html
        ErrorLog ${APACHE_LOG_DIR}/webserverlab_error.log
        CustomLog ${APACHE_LOG_DIR}/webserverlab_access.log combined
 </VirtualHost>
```

Map hostnames locally and enable sites:

Add hosts entries (local name resolution)

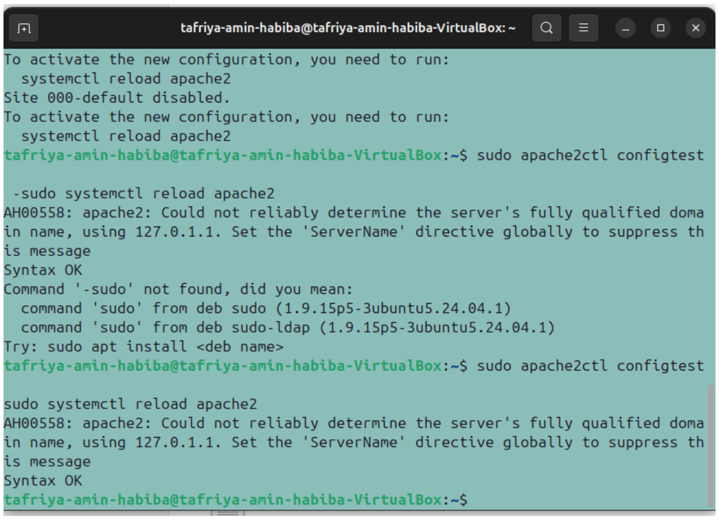
```
 -sudo nano /etc/hosts
 add:
        127.0.0.1   example.com www.example.com
        127.0.0.1   webserverlab.com www.webserverlab.com
```

Enable sites and reload

```
 - sudo a2ensite example.com.conf
 -sudo a2ensite webserverlab.com.conf
 -sudo a2dissite 000-default.conf
```

(Reload)

```
 -sudo apache2ctl configtest
 -sudo systemctl reload apache2
```

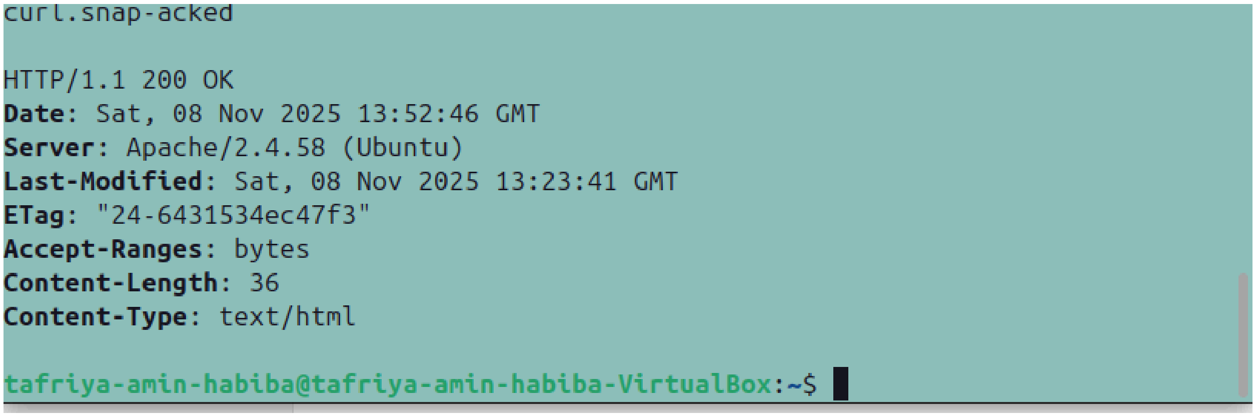


```
To activate the new configuration, you need to run:
  systemctl reload apache2
Site 000-default disabled.
To activate the new configuration, you need to run:
  systemctl reload apache2
tafriya-amin-habiba@tafriya-amin-habiba-VirtualBox:~$ sudo apache2ctl configtest

 -sudo systemctl reload apache2
AH00558: apache2: Could not reliably determine the server's fully qualified doma
in name, using 127.0.1.1. Set the 'ServerName' directive globally to suppress th
is message
Syntax OK
Command '-sudo' not found, did you mean:
  command 'sudo' from deb sudo (1.9.15p5-3ubuntu5.24.04.1)
  command 'sudo' from deb sudo-ldap (1.9.15p5-3ubuntu5.24.04.1)
Try: sudo apt install <deb name>
tafriya-amin-habiba@tafriya-amin-habiba-VirtualBox:~$ sudo apache2ctl configtest

sudo systemctl reload apache2
AH00558: apache2: Could not reliably determine the server's fully qualified doma
in name, using 127.0.1.1. Set the 'ServerName' directive globally to suppress th
is message
Syntax OK
tafriya-amin-habiba@tafriya-amin-habiba-VirtualBox:~$
```

Verify HTTP

-curl -I http://example.com

-curl -I http://webserverlab.com



```
curl.snap-acked

HTTP/1.1 200 OK
Date: Sat, 08 Nov 2025 13:52:46 GMT
Server: Apache/2.4.58 (Ubuntu)
Last-Modified: Sat, 08 Nov 2025 13:23:41 GMT
ETag: "24-6431534ec47f3"
Accept-Ranges: bytes
Content-Length: 36
Content-Type: text/html

tafriya-amin-habiba@tafriya-amin-habiba-VirtualBox:~$ 
```

Task-1: Become a Certificate Authority (CA)

Create a working folder and copy OpenSSL config:

-mkdir -p ~/lab5-ca && cd ~/lab5-ca

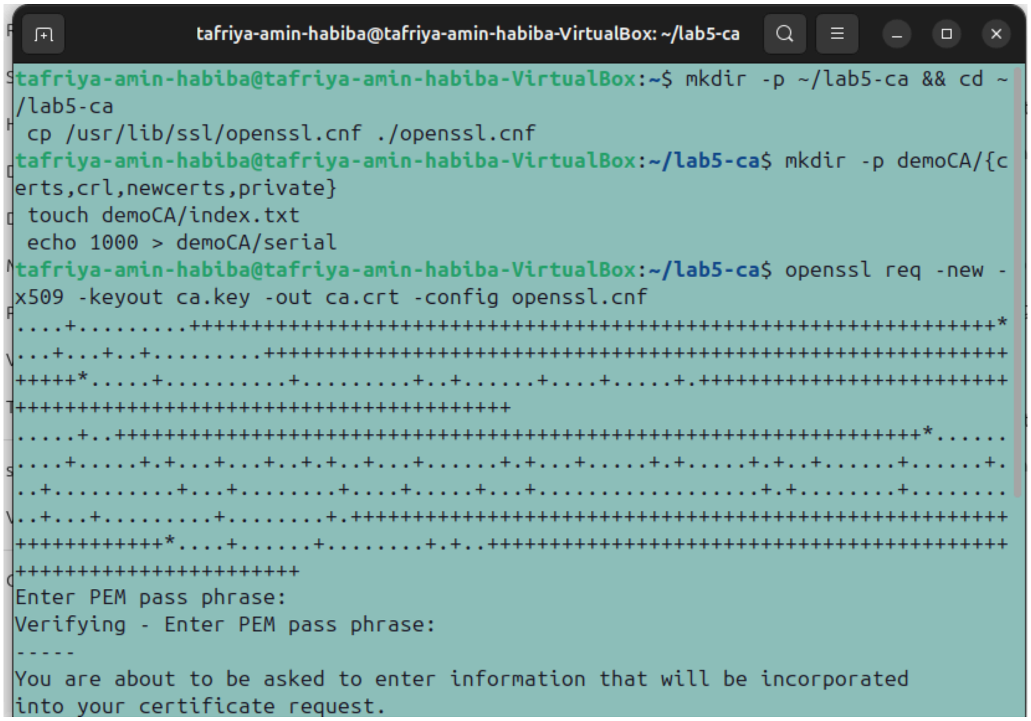
-cp /usr/lib/ssl/openssl.cnf ./openssl.cnf

Create the CA directory structure expected by openssl.cnf (default demoCA path):

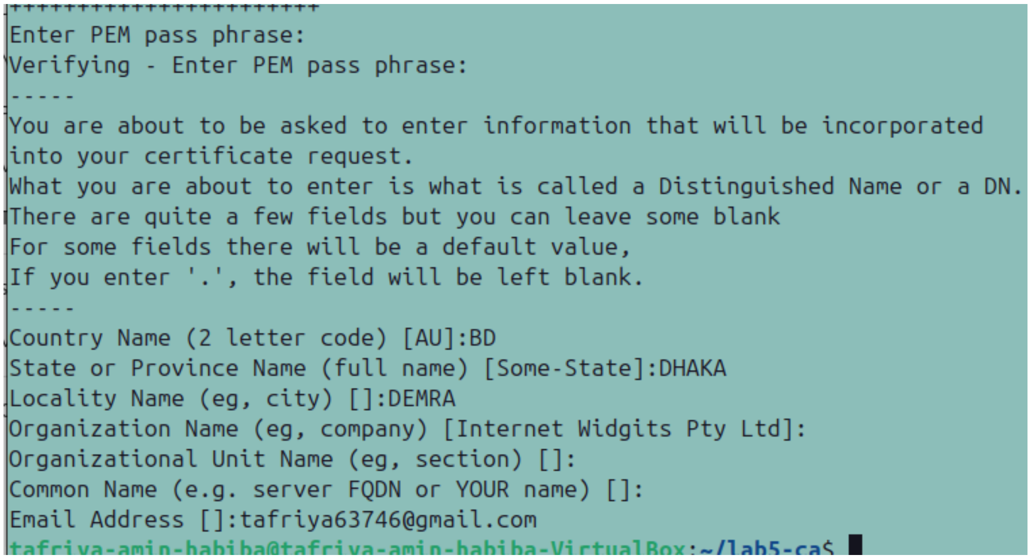
-mkdir -p demoCA/{certs,crl,newcerts,private}
-touch demoCA/index.txt
-echo 1000 > demoCA/serial

Generate the Root CA (self-signed certificate):

-openssl req -new -x509 -keyout ca.key -out ca.crt -config openssl.cnf



```
tafriya-amin-habiba@tafriya-amin-habiba-VirtualBox:~$ mkdir -p ~/lab5-ca && cd ~
/lab5-ca
 cp /usr/lib/ssl/openssl.cnf ./openssl.cnf
tafriya-amin-habiba@tafriya-amin-habiba-VirtualBox:~/lab5-ca$ mkdir -p demoCA/{c
erts,crl,newcerts,private}
 touch demoCA/index.txt
 echo 1000 > demoCA/serial
tafriya-amin-habiba@tafriya-amin-habiba-VirtualBox:~/lab5-ca$ openssl req -new -
x509 -keyout ca.key -out ca.crt -config openssl.cnf
....+........+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++*
...+...+..+........++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
+++++*....+.........+.........+..+......+....+.....+.++++++++++++++++++++++++++++
++++++++++++++++++++++++++++++++++++++++
.....+..++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++*......
....+.....+.+...+...+..+.+..+...+......+.+...+.....+.+.....+.+..+......+......+.
..+.........+...+........+....+.....+...+.................+.+........+........
..+...+.........+........+.++++++++++++++++++++++++++++++++++++++++++++++++++++++
++++++++++++*....+......+........+.+..++++++++++++++++++++++++++++++++++++++++++
+++++++++++++++++++++++
Enter PEM pass phrase:
Verifying - Enter PEM pass phrase:
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
```



```
++++++++++++++++++++++++
Enter PEM pass phrase:
Verifying - Enter PEM pass phrase:
-----
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]:DHAKA
Locality Name (eg, city) []:DEMRA
Organization Name (eg, company) [Internet Widgits Pty Ltd]:
Organizational Unit Name (eg, section) []:
Common Name (e.g. server FQDN or YOUR name) []:
Email Address []:tafriya63746@gmail.com
tafriya-amin-habiba@tafriya-amin-habiba-VirtualBox:~/lab5-ca$
```

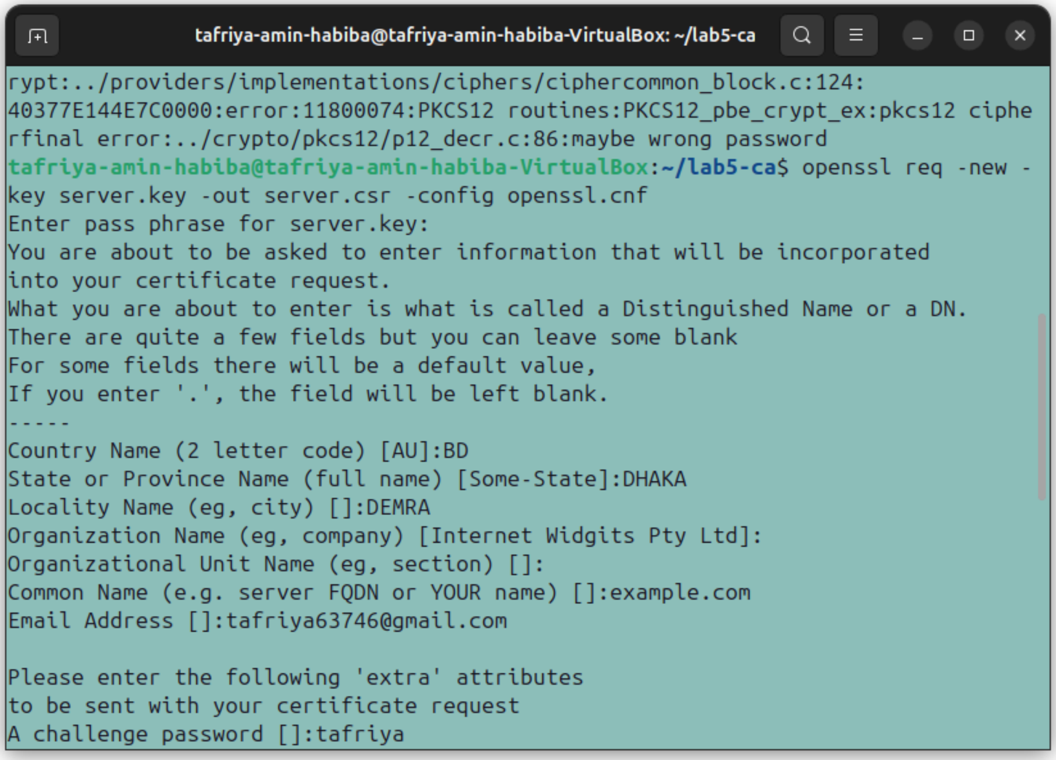
Task-2: Create and Sign Certificate for example.com

Generate a 2048-bit RSA server key:

-openssl genrsa -des3 -out server.key 2048
(creates a password-protected server private key)

Create a CSR (Common Name = example.com):

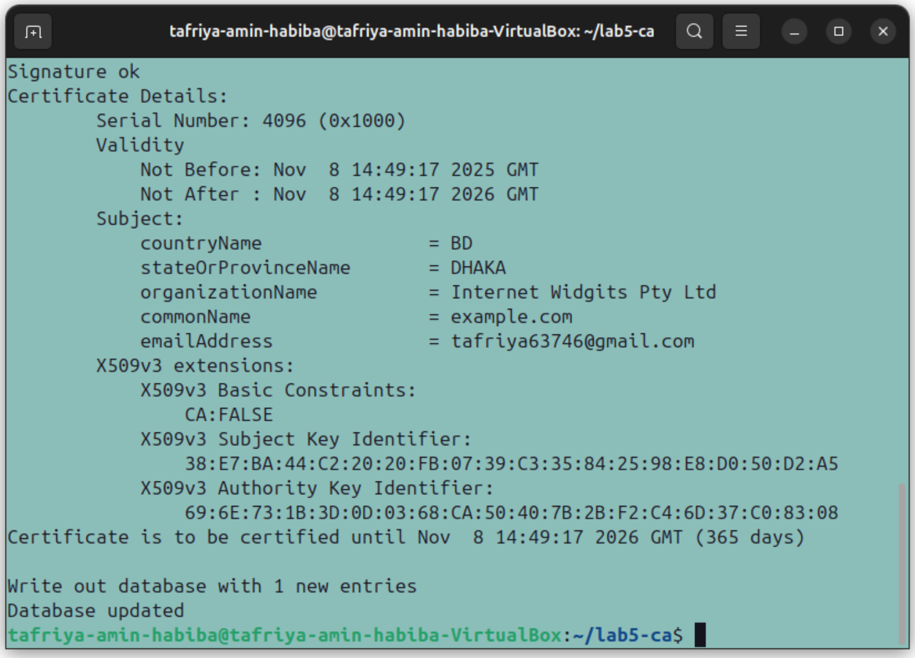
-openssl req -new -key server.key -out server.csr -config openssl.cnf
(prepares a certificate signing request to be signed by the CA)


tafriya-amin-habiba@tafriya-amin-habiba-VirtualBox: ~/lab5-ca

```
rypt:../providers/implementations/ciphers/ciphercommon_block.c:124:
40377E144E7C0000:error:11800074:PKCS12 routines:PKCS12_pbe_crypt_ex:pkcs12 ciphe
rfinal error:../crypto/pkcs12/p12_decr.c:86:maybe wrong password
tafriya-amin-habiba@tafriya-amin-habiba-VirtualBox:~/lab5-ca$ 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]:DHAKA
Locality Name (eg, city) []:DEMRA
Organization Name (eg, company) [Internet Widgits Pty Ltd]:
Organizational Unit Name (eg, section) []:
Common Name (e.g. server FQDN or YOUR name) []:example.com
Email Address []:tafriya63746@gmail.com

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:tafriya
```

Sign the CSR with your CA to produce the server certificate:

-openssl ca -in server.csr -out server.crt -cert ca.crt -keyfile ca.key -config openssl.cnf -batch



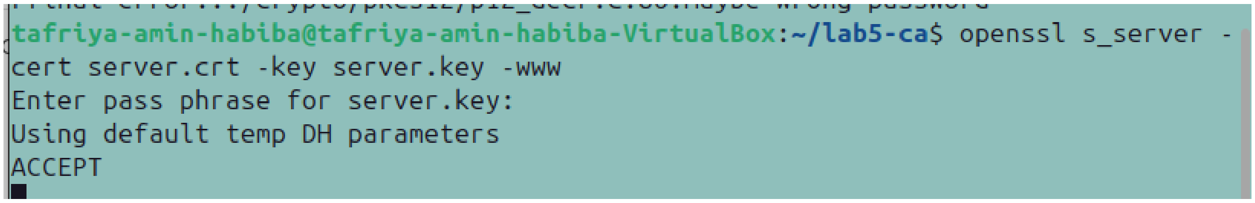
tafriya-amin-habiba@tafriya-amin-habiba-VirtualBox: ~/lab5-ca

```
Signature ok
Certificate Details:
        Serial Number: 4096 (0x1000)
        Validity
            Not Before: Nov  8 14:49:17 2025 GMT
            Not After : Nov  8 14:49:17 2026 GMT
        Subject:
            countryName               = BD
            stateOrProvinceName       = DHAKA
            organizationName          = Internet Widgits Pty Ltd
            commonName                = example.com
            emailAddress              = tafriya63746@gmail.com
        X509v3 extensions:
            X509v3 Basic Constraints:
                CA:FALSE
            X509v3 Subject Key Identifier:
                38:E7:BA:44:C2:20:20:FB:07:39:C3:35:84:25:98:E8:D0:50:D2:A5
            X509v3 Authority Key Identifier:
                69:6E:73:1B:3D:0D:03:68:CA:50:40:7B:2B:F2:C4:6D:37:C0:83:08
Certificate is to be certified until Nov  8 14:49:17 2026 GMT (365 days)

Write out database with 1 new entries
Database updated
tafriya-amin-habiba@tafriya-amin-habiba-VirtualBox:~/lab5-ca$
```

Quick HTTPS Smoke Test with OpenSSL s_server

Start the test server:

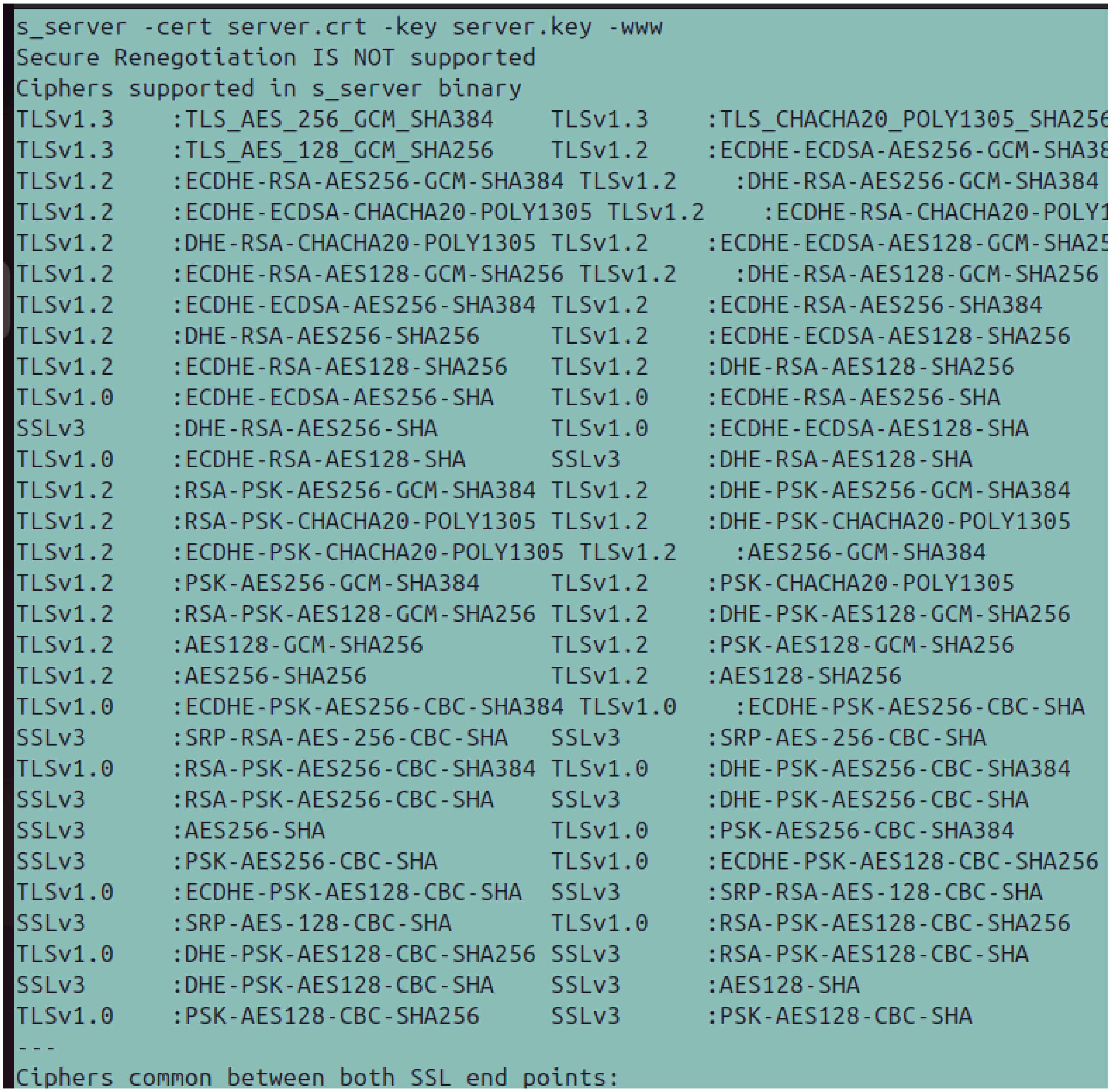
-openssl s_server -cert server.crt -key server.key -www
(runs a temporary HTTPS server on port 4433 serving a test page)



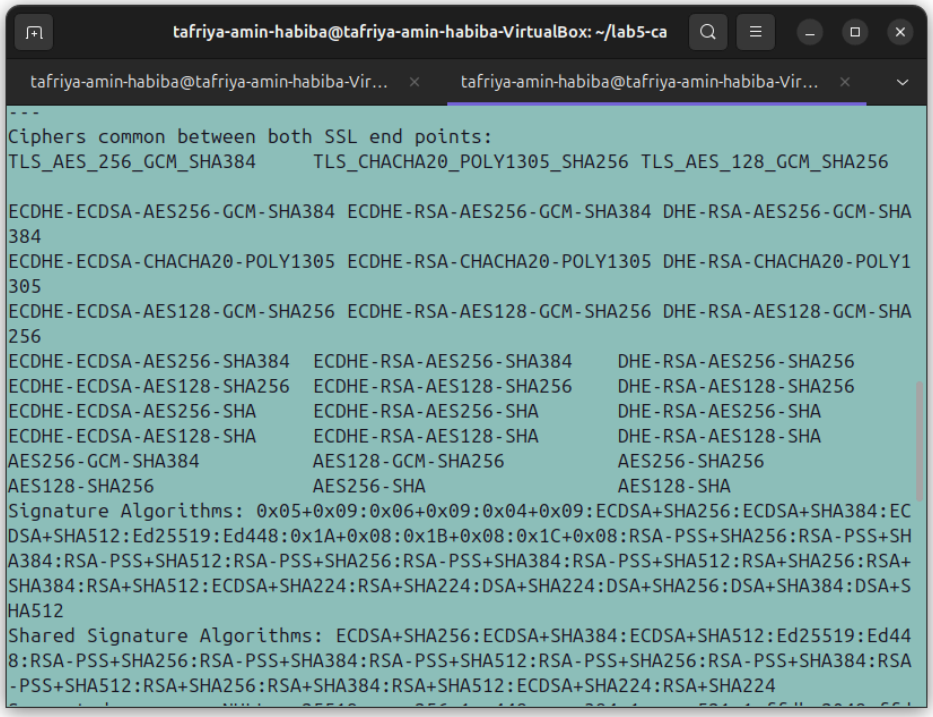
```
tafriya-amin-habiba@tafriya-amin-habiba-VirtualBox:~/lab5-ca$ openssl s_server -
cert server.crt -key server.key -www
Enter pass phrase for server.key:
Using default temp DH parameters
ACCEPT
```

In another terminal, verify the page loads:

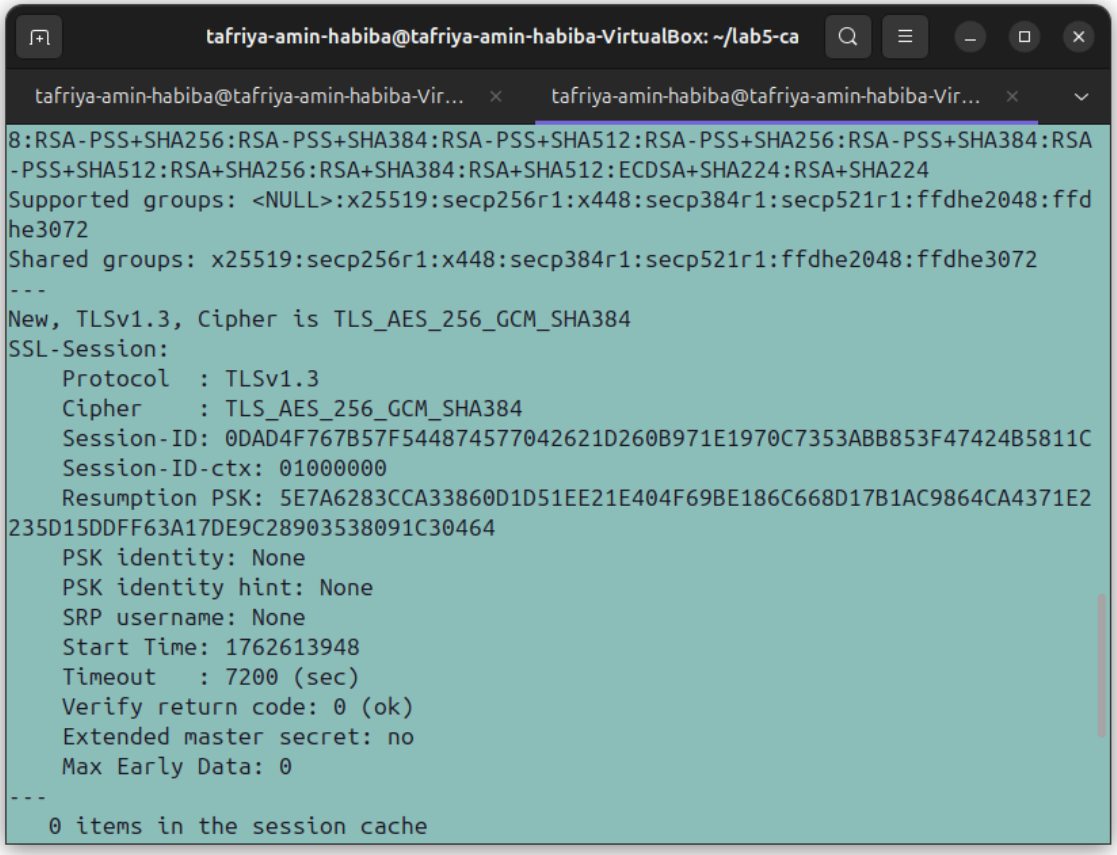
or quick (insecure) test:
curl -k https://localhost:4433/



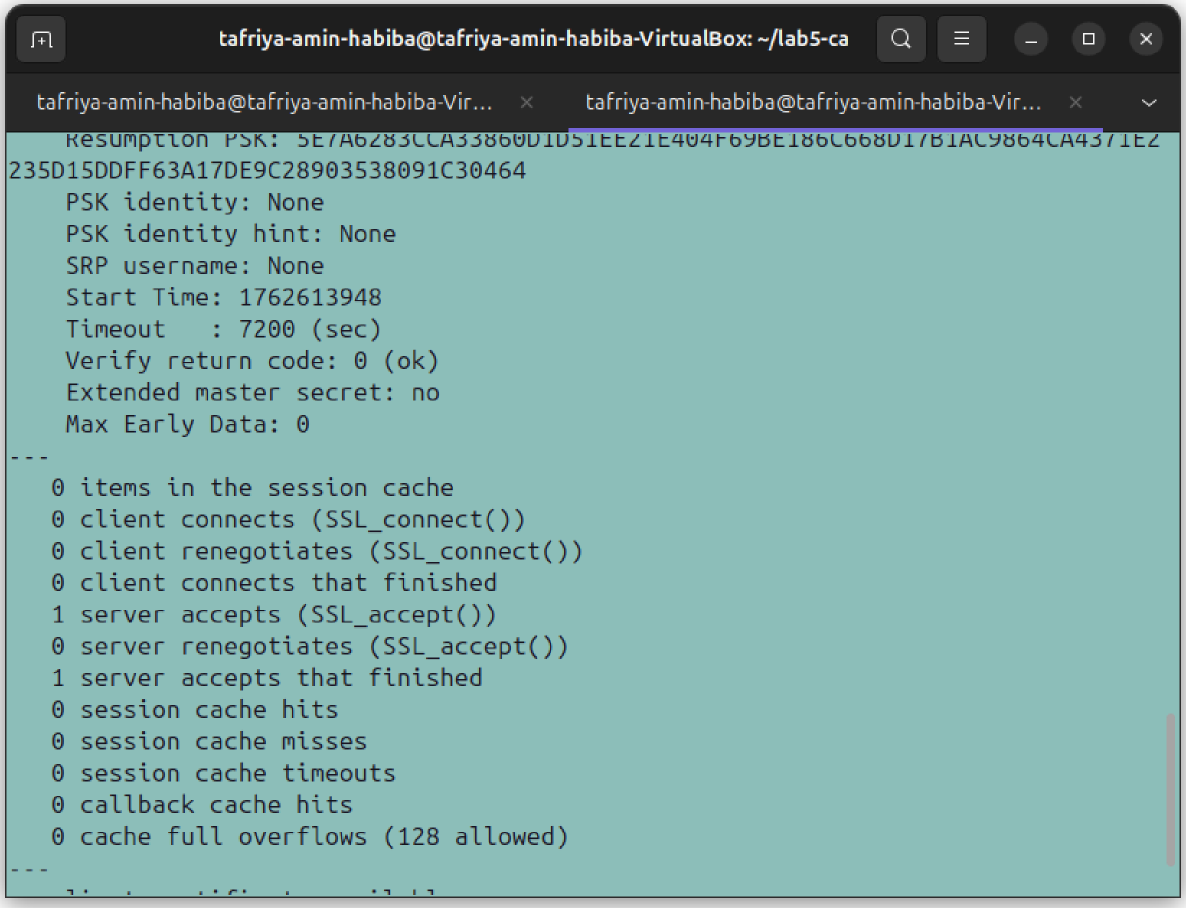
```
s_server -cert server.crt -key server.key -www
Secure Renegotiation IS NOT supported
Ciphers supported in s_server binary
TLSv1.3    :TLS_AES_256_GCM_SHA384    TLSv1.3    :TLS_CHACHA20_POLY1305_SHA256
TLSv1.3    :TLS_AES_128_GCM_SHA256    TLSv1.2    :ECDHE-ECDSA-AES256-GCM-SHA38
TLSv1.2    :ECDHE-RSA-AES256-GCM-SHA384 TLSv1.2    :DHE-RSA-AES256-GCM-SHA384
TLSv1.2    :ECDHE-ECDSA-CHACHA20-POLY1305 TLSv1.2    :ECDHE-RSA-CHACHA20-POLY1
TLSv1.2    :DHE-RSA-CHACHA20-POLY1305 TLSv1.2    :ECDHE-ECDSA-AES128-GCM-SHA25
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:
```


tafriya-amin-habiba@tafriya-amin-habiba-VirtualBox: ~/lab5-ca
tafriya-amin-habiba@tafriya-amin-habiba-Vir...
tafriya-amin-habiba@tafriya-amin-habiba-Vir...

```
---
Ciphers common between both SSL end points:
TLS_AES_256_GCM_SHA384     TLS_CHACHA20_POLY1305_SHA256 TLS_AES_128_GCM_SHA256

ECDHE-ECDSA-AES256-GCM-SHA384 ECDHE-RSA-AES256-GCM-SHA384 DHE-RSA-AES256-GCM-SHA
384
ECDHE-ECDSA-CHACHA20-POLY1305 ECDHE-RSA-CHACHA20-POLY1305 DHE-RSA-CHACHA20-POLY1
305
ECDHE-ECDSA-AES128-GCM-SHA256 ECDHE-RSA-AES128-GCM-SHA256 DHE-RSA-AES128-GCM-SHA
256
ECDHE-ECDSA-AES256-SHA384  ECDHE-RSA-AES256-SHA384    DHE-RSA-AES256-SHA256
ECDHE-ECDSA-AES128-SHA256  ECDHE-RSA-AES128-SHA256    DHE-RSA-AES128-SHA256
ECDHE-ECDSA-AES256-SHA     ECDHE-RSA-AES256-SHA       DHE-RSA-AES256-SHA
ECDHE-ECDSA-AES128-SHA     ECDHE-RSA-AES128-SHA       DHE-RSA-AES128-SHA
AES256-GCM-SHA384          AES128-GCM-SHA256          AES256-SHA256
AES128-SHA256              AES256-SHA                 AES128-SHA
Signature Algorithms: 0x05+0x09:0x06+0x09:0x04+0x09:ECDSA+SHA256:ECDSA+SHA384:EC
DSA+SHA512:Ed25519:Ed448:0x1A+0x08:0x1B+0x08:0x1C+0x08:RSA-PSS+SHA256:RSA-PSS+SH
A384:RSA-PSS+SHA512:RSA-PSS+SHA256:RSA-PSS+SHA384:RSA-PSS+SHA512:RSA+SHA256:RSA+
SHA384:RSA+SHA512:ECDSA+SHA224:RSA+SHA224:DSA+SHA224:DSA+SHA256:DSA+SHA384:DSA+S
HA512
Shared Signature Algorithms: ECDSA+SHA256:ECDSA+SHA384:ECDSA+SHA512:Ed25519:Ed44
8:RSA-PSS+SHA256:RSA-PSS+SHA384:RSA-PSS+SHA512:RSA-PSS+SHA256:RSA-PSS+SHA384:RSA
-PSS+SHA512:RSA+SHA256:RSA+SHA384:RSA+SHA512:ECDSA+SHA224:RSA+SHA224
```


tafriya-amin-habiba@tafriya-amin-habiba-VirtualBox: ~/lab5-ca
tafriya-amin-habiba@tafriya-amin-habiba-Vir...
tafriya-amin-habiba@tafriya-amin-habiba-Vir...

```
8:RSA-PSS+SHA256:RSA-PSS+SHA384:RSA-PSS+SHA512:RSA-PSS+SHA256:RSA-PSS+SHA384:RSA
-PSS+SHA512:RSA+SHA256:RSA+SHA384:RSA+SHA512:ECDSA+SHA224:RSA+SHA224
Supported groups: <NULL>:x25519:secp256r1:x448:secp384r1:secp521r1:ffdhe2048:ffd
he3072
Shared groups: x25519:secp256r1:x448:secp384r1:secp521r1:ffdhe2048:ffdhe3072
---
New, TLSv1.3, Cipher is TLS_AES_256_GCM_SHA384
SSL-Session:
    Protocol  : TLSv1.3
    Cipher    : TLS_AES_256_GCM_SHA384
    Session-ID: 0DAD4F767B57F544874577042621D260B971E1970C7353ABB853F47424B5811C
    Session-ID-ctx: 01000000
    Resumption PSK: 5E7A6283CCA33860D1D51EE21E404F69BE186C668D17B1AC9864CA4371E2
235D15DDFF63A17DE9C28903538091C30464
    PSK identity: None
    PSK identity hint: None
    SRP username: None
    Start Time: 1762613948
    Timeout   : 7200 (sec)
    Verify return code: 0 (ok)
    Extended master secret: no
    Max Early Data: 0
---
   0 items in the session cache
```


tafriya-amin-habiba@tafriya-amin-habiba-VirtualBox: ~/lab5-ca
tafriya-amin-habiba@tafriya-amin-habiba-Vir...
tafriya-amin-habiba@tafriya-amin-habiba-Vir...

```
    Resumption PSK: 5E7A6283CCA33860D1D51EE21E404F69BE186C668D17B1AC9864CA4371E2
235D15DDFF63A17DE9C28903538091C30464
    PSK identity: None
    PSK identity hint: None
    SRP username: None
    Start Time: 1762613948
    Timeout   : 7200 (sec)
    Verify return code: 0 (ok)
    Extended master secret: no
    Max Early Data: 0
---
   0 items in the session cache
   0 client connects (SSL_connect())
   0 client renegotiates (SSL_connect())
   0 client connects that finished
   1 server accepts (SSL_accept())
   0 server renegotiates (SSL_accept())
   1 server accepts that finished
   0 session cache hits
   0 session cache misses
   0 session cache timeouts
   0 callback cache hits
   0 cache full overflows (128 allowed)
---
```

Task-3: Deploy HTTPS in Apache

Create an HTTPS virtual host for example.com:

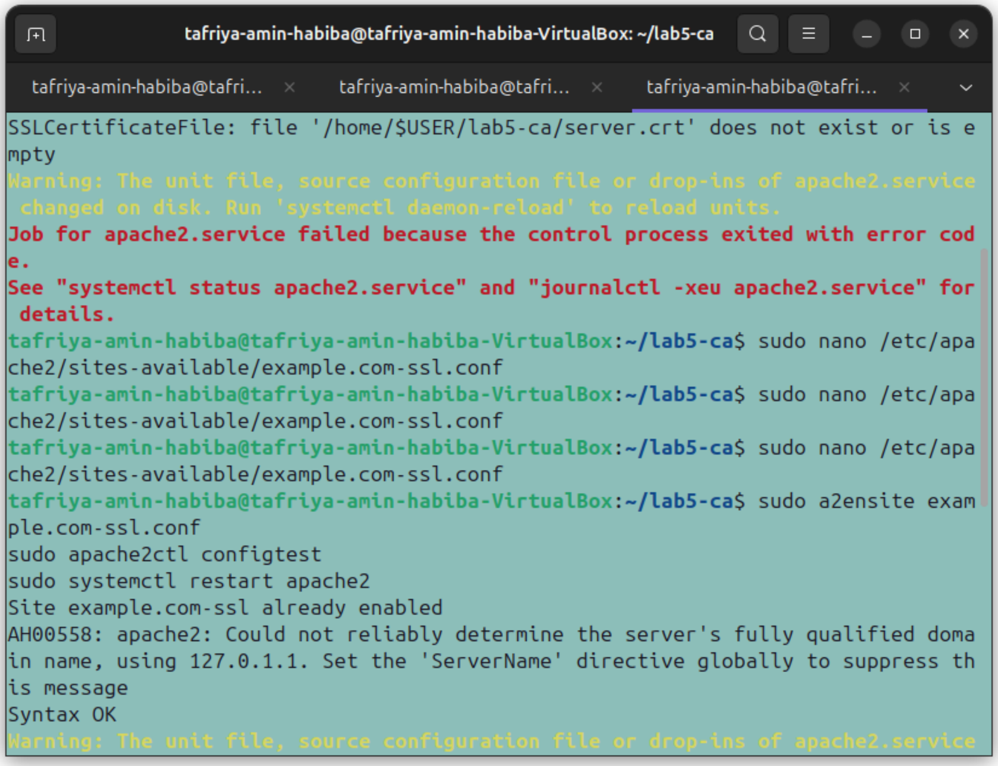
-sudo nano /etc/apache2/sites-available/example.com-ssl.conf

```
<IfModule mod_ssl.c>
 <VirtualHost *:443>
   ServerAdmin admin@example.com
   ServerName example.com
   ServerAlias www.example.com

   DocumentRoot /var/www/example.com/html
   ErrorLog ${APACHE_LOG_DIR}/example_ssl_error.log
   CustomLog ${APACHE_LOG_DIR}/example_ssl_access.log combined

   SSLEngine on
```

```
  SSLCertificateFile    /home/tafriya-amin-habiba/lab5-ca/server.crt
  SSLCertificateKeyFile /home/tafriya-amin-habiba/lab5-ca/server.key
</VirtualHost>
</IfModule>
```

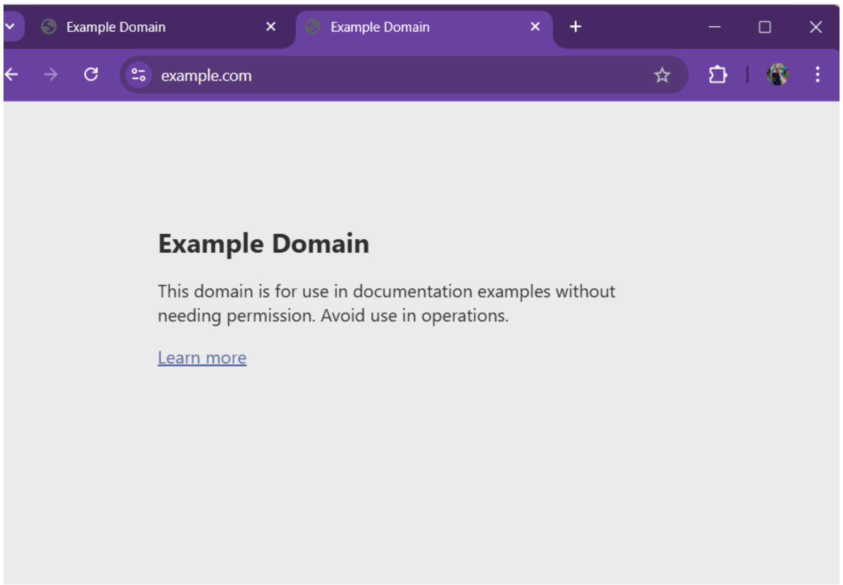
Enable the SSL site and restart Apache:

-sudo a2ensite example.com-ssl.conf
-sudo apache2ctl configtest
-sudo systemctl restart apache2


tafriya-amin-habiba@tafriya-amin-habiba-VirtualBox: ~/lab5-ca
tafriya-amin-habiba@tafri...
tafriya-amin-habiba@tafri...
tafriya-amin-habiba@tafri...

```
SSLCertificateFile: file '/home/$USER/lab5-ca/server.crt' does not exist or is e
mpty
Warning: The unit file, source configuration file or drop-ins of apache2.service
 changed on disk. Run 'systemctl daemon-reload' to reload units.
Job for apache2.service failed because the control process exited with error cod
e.
See "systemctl status apache2.service" and "journalctl -xeu apache2.service" for
 details.
tafriya-amin-habiba@tafriya-amin-habiba-VirtualBox:~/lab5-ca$ sudo nano /etc/apa
che2/sites-available/example.com-ssl.conf
tafriya-amin-habiba@tafriya-amin-habiba-VirtualBox:~/lab5-ca$ sudo nano /etc/apa
che2/sites-available/example.com-ssl.conf
tafriya-amin-habiba@tafriya-amin-habiba-VirtualBox:~/lab5-ca$ sudo nano /etc/apa
che2/sites-available/example.com-ssl.conf
tafriya-amin-habiba@tafriya-amin-habiba-VirtualBox:~/lab5-ca$ sudo a2ensite exam
ple.com-ssl.conf
sudo apache2ctl configtest
sudo systemctl restart apache2
Site example.com-ssl already enabled
AH00558: apache2: Could not reliably determine the server's fully qualified doma
in name, using 127.0.1.1. Set the 'ServerName' directive globally to suppress th
is message
Syntax OK
Warning: The unit file, source configuration file or drop-ins of apache2.service
```

Test in browser: https://example.com/

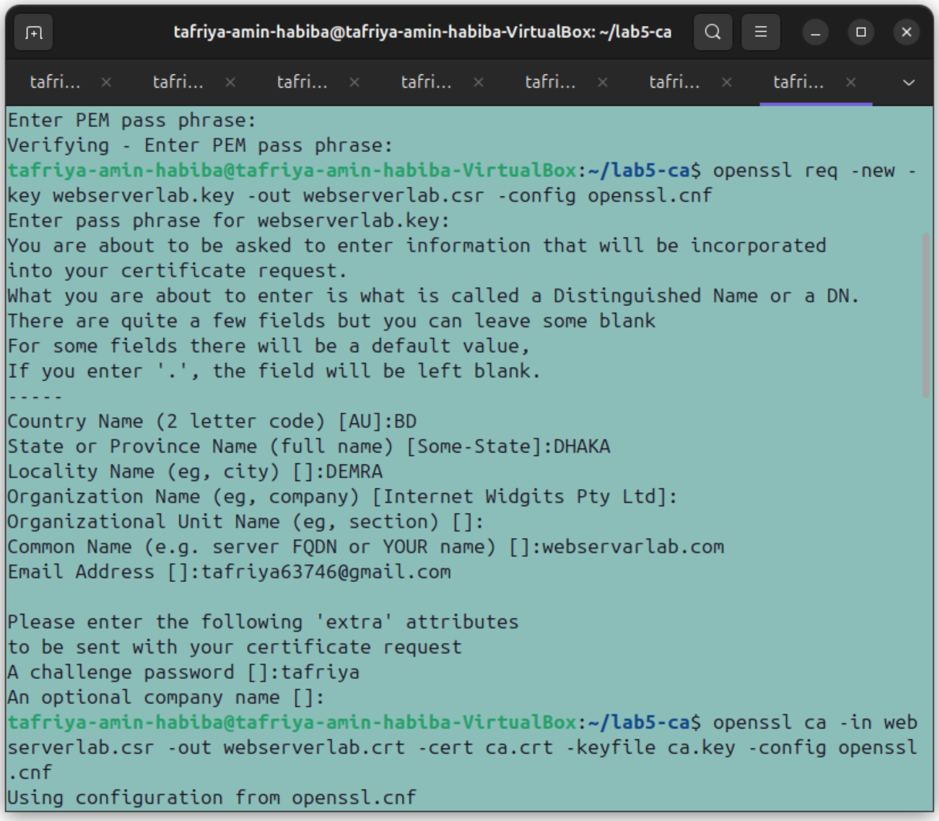


Example Domain

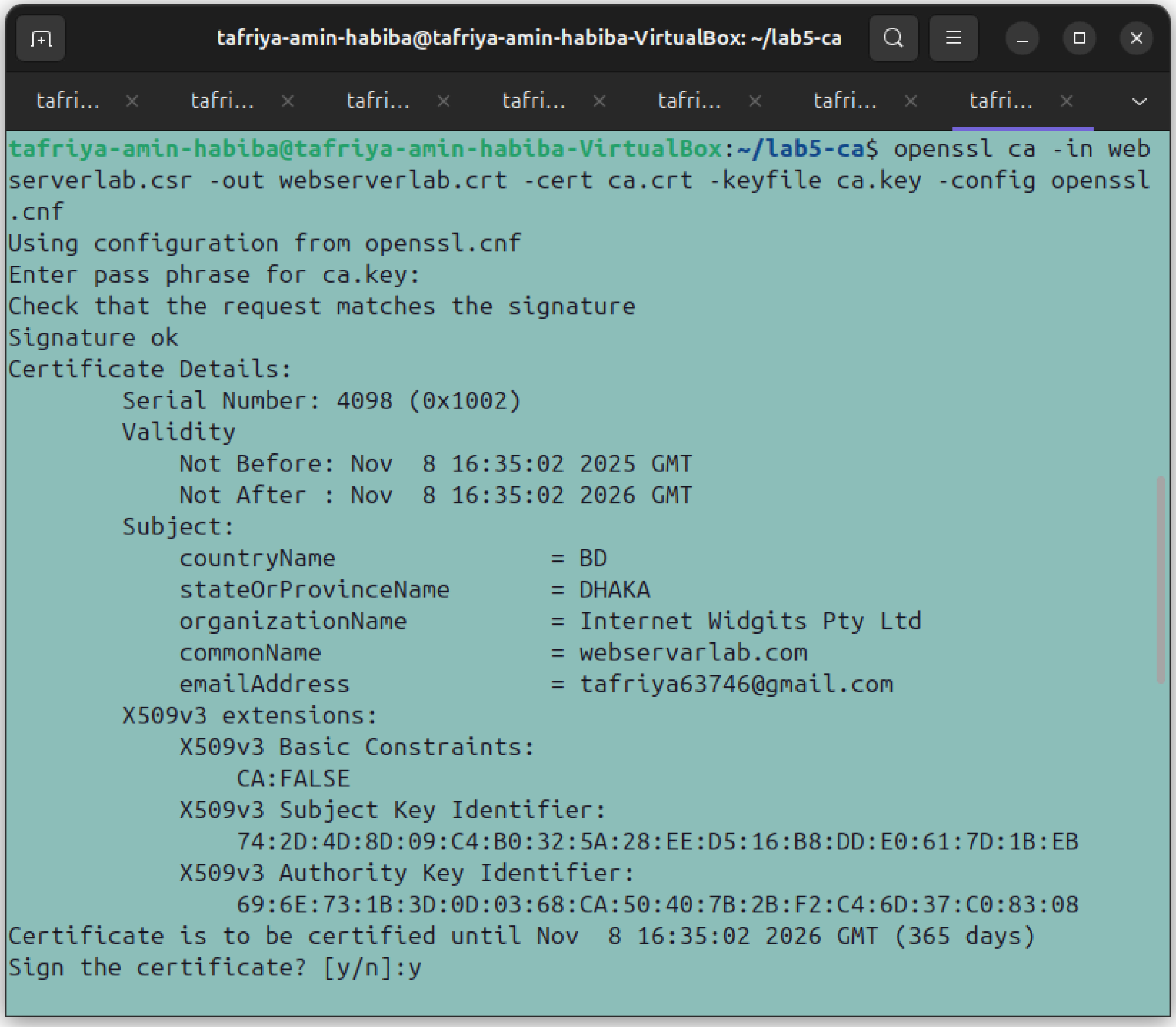
This domain is for use in documentation examples without needing permission. Avoid use in operations.

Learn more

Task 4 : Repeat for webserverlab.com


tafriya-amin-habiba@tafriya-amin-habiba-VirtualBox: ~/lab5-ca
tafri... tafri... tafri... tafri... tafri... tafri... tafri...

```
Enter PEM pass phrase:
Verifying - Enter PEM pass phrase:
tafriya-amin-habiba@tafriya-amin-habiba-VirtualBox:~/lab5-ca$ openssl req -new -
key webserverlab.key -out webserverlab.csr -config openssl.cnf
Enter pass phrase for webserverlab.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]:DHAKA
Locality Name (eg, city) []:DEMRA
Organization Name (eg, company) [Internet Widgits Pty Ltd]:
Organizational Unit Name (eg, section) []:
Common Name (e.g. server FQDN or YOUR name) []:webservarlab.com
Email Address []:tafriya63746@gmail.com

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:tafriya
An optional company name []:
tafriya-amin-habiba@tafriya-amin-habiba-VirtualBox:~/lab5-ca$ openssl ca -in web
serverlab.csr -out webserverlab.crt -cert ca.crt -keyfile ca.key -config openssl
.cnf
Using configuration from openssl.cnf
```


tafriya-amin-habiba@tafriya-amin-habiba-VirtualBox: ~/lab5-ca
tafri...
tafri...
tafri...
tafri...
tafri...
tafri...
tafri...

```
tafriya-amin-habiba@tafriya-amin-habiba-VirtualBox:~/lab5-ca$ openssl ca -in web
serverlab.csr -out webserverlab.crt -cert ca.crt -keyfile ca.key -config openssl
.cnf
Using configuration from openssl.cnf
Enter pass phrase for ca.key:
Check that the request matches the signature
Signature ok
Certificate Details:
        Serial Number: 4098 (0x1002)
        Validity
            Not Before: Nov  8 16:35:02 2025 GMT
            Not After : Nov  8 16:35:02 2026 GMT
        Subject:
            countryName               = BD
            stateOrProvinceName       = DHAKA
            organizationName          = Internet Widgits Pty Ltd
            commonName                = webservarlab.com
            emailAddress              = tafriya63746@gmail.com
        X509v3 extensions:
            X509v3 Basic Constraints:
                CA:FALSE
            X509v3 Subject Key Identifier:
                74:2D:4D:8D:09:C4:B0:32:5A:28:EE:D5:16:B8:DD:E0:61:7D:1B:EB
            X509v3 Authority Key Identifier:
                69:6E:73:1B:3D:0D:03:68:CA:50:40:7B:2B:F2:C4:6D:37:C0:83:08
Certificate is to be certified until Nov  8 16:35:02 2026 GMT (365 days)
Sign the certificate? [y/n]:y
```

In this lab, I secured an Apache web server using SSL/TLS certificates generated with OpenSSL. The process began by setting up two HTTP virtual hosts, example.com and webserverlab.com, and verifying their functionality over port 80. I then created my own Certificate Authority (CA) and used it to issue domain-specific certificates, ensuring complete control over the trust chain.

Using the OpenSSL **s_server** and **s_client** tools, we verified the validity of our certificates,confirming a proper TLS 1.3 handshake with “**Verify return code: 0 (ok)**.”