Lab4

Name: Zhiyuan Ma, Wenting Zheng

Task1:

Preparing files:

```
[11/21/22]seed@VM:~/.../lab4$ cp /usr/lib/ssl/openssl.cnf .
[11/21/22]seed@VM:~/.../lab4$ ls
openssl.cnf
[11/21/22]seed@VM:~/.../lab4$ mkdir demoCA
[11/21/22]seed@VM:~/.../lab4$ cd demoCA
[11/21/22]seed@VM:~/.../demoCA$ mkdir certs crl newcerts
[11/21/22]seed@VM:~/.../demoCA$ touch index.txt serial
[11/21/22]seed@VM:~/.../demoCA$ ls
certs  crl  index.txt  newcerts  serial
[11/21/22]seed@VM:~/.../demoCA$ vim serial
[11/21/22]seed@VM:~/.../demoCA$ ls
certs  crl  index.txt  newcerts  serial
[11/21/22]seed@VM:~/.../demoCA$
```

Generate self-signed certificate:

```
Generating a 2048 bit RSA private key
.....................................................................................
.....................................+++
.......................................................................+++
writing new private key to 'ca.key'
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]:US
State or Province Name (full name) [Some-State]:CA
Locality Name (eg, city) []:IRVINE
Organization Name (eg, company) [Internet Widgits Pty Ltd]:TEST
Organizational Unit Name (eg, section) []:TEST
Common Name (e.g. server FQDN or YOUR name) []:TEST
Email Address []:zhiyuam3@uci.edu
[11/21/22]seed@VM:~/.../lab4$
```

```
[11/21/22]seed@VM:~/.../lab4$ ls
ca.crt  ca.key  demoCA  openssl.cnf
[11/21/22]seed@VM:~/.../lab4$
```

Task2:

Step 1: Generate public/private key pair.

```
[11/21/22]seed@VM:~/.../lab4$ openssl genrsa -aes128 -out server.key 1024
Generating RSA private key, 1024 bit long modulus
..++++++
.................++++++
e is 65537 (0x10001)
Enter pass phrase for server.key:
3070953152:error:28069065:lib(40):UI_set_result:result too small:ui_lib.c:823:Yo
u must type in 4 to 1023 characters
Enter pass phrase for server.key:
Verifying - Enter pass phrase for server.key:
3070953152:error:28069065:lib(40):UI_set_result:result too small:ui_lib.c:823:Yo
u must type in 4 to 1023 characters
Enter pass phrase for server.key:
Verifying - Enter pass phrase for server.key:
[11/21/22]seed@VM:~/.../lab4$ ls
ca.crt  ca.key  demoCA  openssl.cnf  server.key
[11/21/22]seed@VM:~/.../lab4$
```

View content:

```
[11/21/22]seed@VM:~/.../lab4$ openssl rsa -in server.key -text
Enter pass phrase for server.key:
Private-Key: (1024 bit)
modulus:
    00:cf:f8:43:75:5d:88:55:db:6a:9b:d6:5d:65:15:
    e9:5b:e7:6c:e9:e9:15:7e:cf:98:91:4f:48:77:dd:
    49:26:2b:69:a7:cc:74:0d:14:9e:ba:1a:6c:85:26:
    59:91:91:31:56:48:37:ee:ff:b8:f6:85:46:ac:b4:
    54:0e:af:7c:0d:e9:f1:cf:fc:1c:86:87:93:b7:48:
    55:b7:ef:df:44:4c:f8:5e:2c:e3:b3:35:4d:6f:65:
    00:b2:da:fb:f8:60:00:37:d7:e9:46:ed:30:52:ed:
    39:ba:50:70:f2:4d:1c:7c:3e:2b:cc:bd:a8:fc:f4:
    14:93:38:fb:a6:17:6f:33:d3
publicExponent: 65537 (0x10001)
privateExponent:
    0a:78:d1:cc:c3:1c:ee:55:d2:14:6e:c2:dd:28:31:
    4b:f3:67:44:8f:fc:09:1a:a2:f0:0e:31:f3:f7:2c:
    3:7b:da:7b:2d:0c:cb:91:02:85:fa:18:ea:49:
    70:4f:03:50:f8:e9:f8:1a:e0:51:69:7c:99:8d:aa:
    fd:e2:50:38:bb:d9:48:ed:b1:41:fd:4b:e4:28:be:
```

Terminal

Step 2: Generate a Certificate Signing Request (CSR).

```
[11/21/22]seed@VM:~/.../lab4$ 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]:US
State or Province Name (full name) [Some-State]:CA
Locality Name (eg, city) []:IRVINE
Organization Name (eg, company) [Internet Widgits Pty Ltd]:SEEDPKILab2020.com
Organizational Unit Name (eg, section) []:SEEDPKILab2020.com
Common Name (e.g. server FQDN or YOUR name) []:SEEDPKILab2020.com
Email Address []:zhiyuam3@uci.edu

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:805805
An optional company name []:SEEDPKILab2020.com
```

```
[11/21/22]seed@VM:~/.../lab4$ ls
ca.crt  ca.key  demoCA  openssl.cnf  server.csr  server.key
[11/21/22]seed@VM:~/.../lab4$ cat server.csr
-----BEGIN CERTIFICATE REQUEST-----
MIICHjCCAYcCAQAwgaMxCzAJBgNVBAYTAlVTMQswCQYDVQQIDAJDQTEPMA0GA1UE
BwwGSVJWSU5FMRswGQYDVQQKDBJTRUVEUEtJTGFiMjAyMC5jb20xGzAZBgNVBAsM
ElNFRURQS0lMYWIyMDIwLmNvbTEbMBkGA1UEAwwSU0VFRFBLSUxhYjIwMjAuY29t
MR8wHQYJKoZIhvcNAQkBFhB6aGl5dWFtM0B1Y2kuZWR1MIGfMA0GCSqGSIb3DQEB
AQUAA4GNADCBiQKBgQDP+EN1XYhV22qb1l1lFelb52zp6RV+z5iRT0h33UkmK2mn
zHQNFJ66GmyFJlmRkTFWSDfu/7j2hUastFQOr3wN6fHP/ByGh5O3SFW3799ETPhe
LOOzNU1vZQCy2vv4YAA31+lG7TBS7Tm6UHDyTRx8PivMvaj89BSTOPumF28z0wID
AQABoDowFQYJKoZIhvcNAQkHMQgMBjgwNTgwNTAhBgkqhkiG9w0BCQIxFAwSU0VF
RFBLSUxhYjIwMjAuY29tMA0GCSqGSIb3DQEBCwUAA4GBALFTGLCwQQ7NAtCeehZx
eC9GQHTUTootGizQVn1OYAj3HJRGgc4r/WbtjFvdzgn25jIDBzXQq8QQ8n004Tmw
09G7V3GioaFMJBupO0EkTpcAiFl/0oZzyy7Vpph6JnDG2PhuCyvwN1TEBqh4pDn4
vErkypp5O4krNlqMVKp2dwZn
-----END CERTIFICATE REQUEST-----
```

Step 3: Generating Certificates.

Different org name:

```
[11/21/22]seed@VM:~/.../lab4$ openssl ca -in server.csr -out server.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
The organizationName field needed to be the same in the
CA certificate (TEST) and the request (SEEDPKILab2020.com)
[11/21/22]seed@VM:~/.../lab4$
```

So we have to change the policy:

```
default_days    = 365                   # how long to certify for
default_crl_days= 30                    # how long before next CRL
default_md      = default               # use public key default MD
preserve        = no                    # keep passed DN ordering

# A few difference way of specifying how similar the request should look
# For type CA, the listed attributes must be the same, and the optional
# and supplied fields are just that :-)
policy          = policy_anything

# For the CA policy
[ policy_match ]
countryName             = match
stateOrProvinceName     = match
organizationName        = match
organizationalUnitName  = optional
commonName              = supplied
emailAddress            = optional

# For the 'anything' policy
# At this point in time, you must list all acceptable 'object'
# types.
[ policy_anything ]
-- INSERT --                                                  81,26-34      22%
```

Then we succeed:

```
            organizationName          = SEEDPKILab2020.com
            organizationalUnitName    = SEEDPKILab2020.com
            commonName                = SEEDPKILab2020.com
            emailAddress              = zhiyuam3@uci.edu
        X509v3 extensions:
            X509v3 Basic Constraints:
                CA:FALSE
            Netscape Comment:
                OpenSSL Generated Certificate
            X509v3 Subject Key Identifier:
                F3:D0:30:4E:CD:44:C9:51:10:B3:78:B0:A1:CC:E4:E0:18:D8:D8:10
            X509v3 Authority Key Identifier:
                keyid:C7:35:EA:7A:F3:B7:33:01:A9:C7:C6:78:A8:5D:D6:DA:10:F1:D4:7
C

Certificate is to be certified until Nov 21 22:28:20 2023 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
Data Base Updated
[11/21/22]seed@VM:~/.../lab4$
```

Task 3:

Step 1: Configuring DNS.

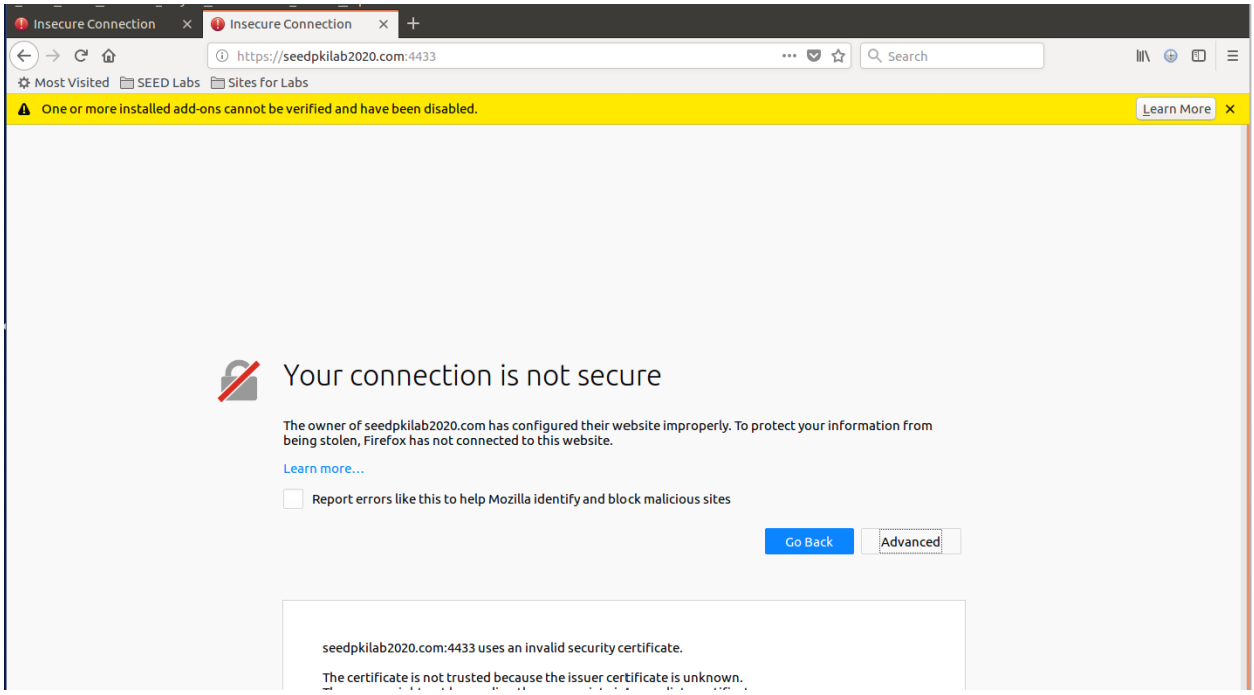
```
127.0.0.1       localhost
127.0.1.1       VM

# The following lines are desirable for IPv6 capable hosts
::1     ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
127.0.0.1       User
127.0.0.1       Attacker
127.0.0.1       Server
127.0.0.1       www.SeedLabSQLInjection.com
127.0.0.1       www.xsslabelgg.com
127.0.0.1       www.csrflabelgg.com
127.0.0.1       www.csrflabattacker.com
127.0.0.1       www.repackagingattacklab.com
127.0.0.1       www.seedlabclickjacking.com
127.0.0.1    SEEDPKILab2020.com
~
~
~
~
-- INSERT --                                          19,32         All
```

Step 2: Configuring the web server.
Combine the secret key and certificate into one file

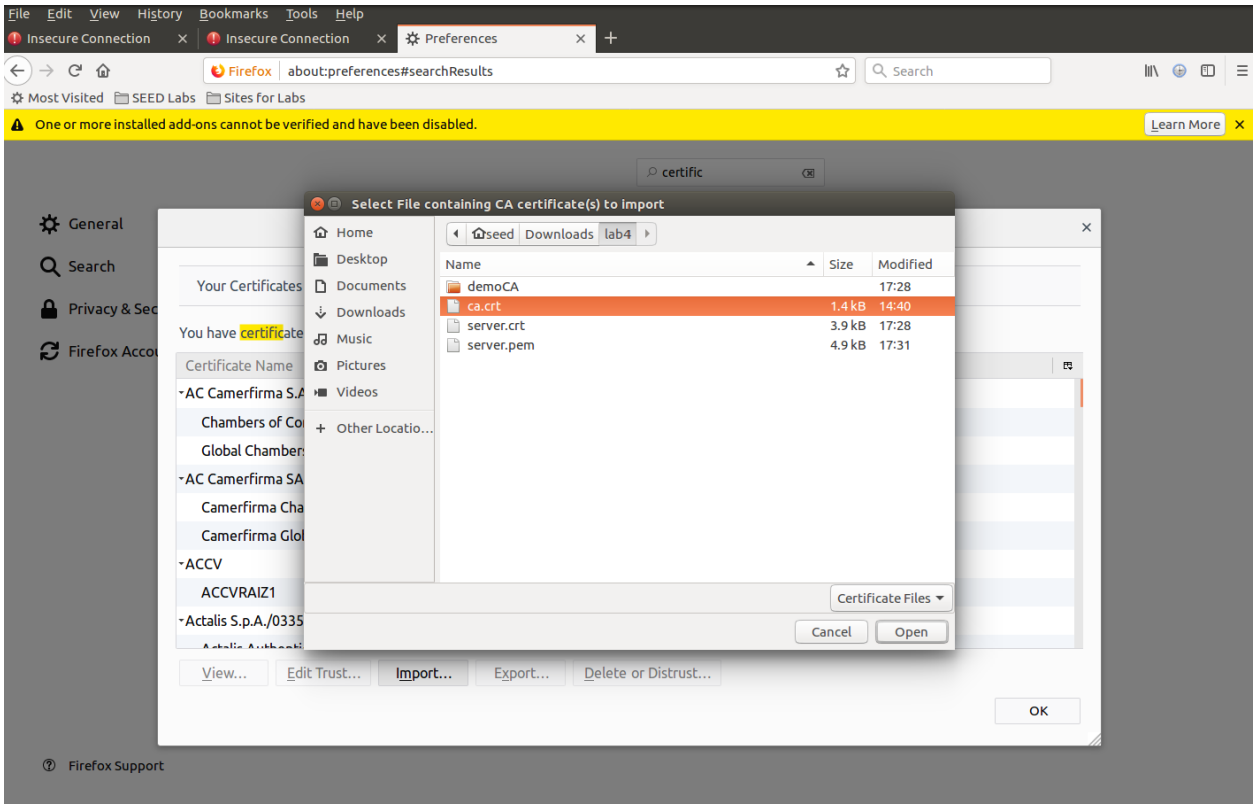
```
[11/21/22]seed@VM:~/.../lab4$ sudo vim /etc/hosts
[11/21/22]seed@VM:~/.../lab4$ cp server.key server.pem
[11/21/22]seed@VM:~/.../lab4$ cat server.crt >> server.pem
[11/21/22]seed@VM:~/.../lab4$ ls
ca.crt  demoCA       server.crt  server.key
ca.key  openssl.cnf  server.csr  server.pem
[11/21/22]seed@VM:~/.../lab4$
```

Create a server using the pem:

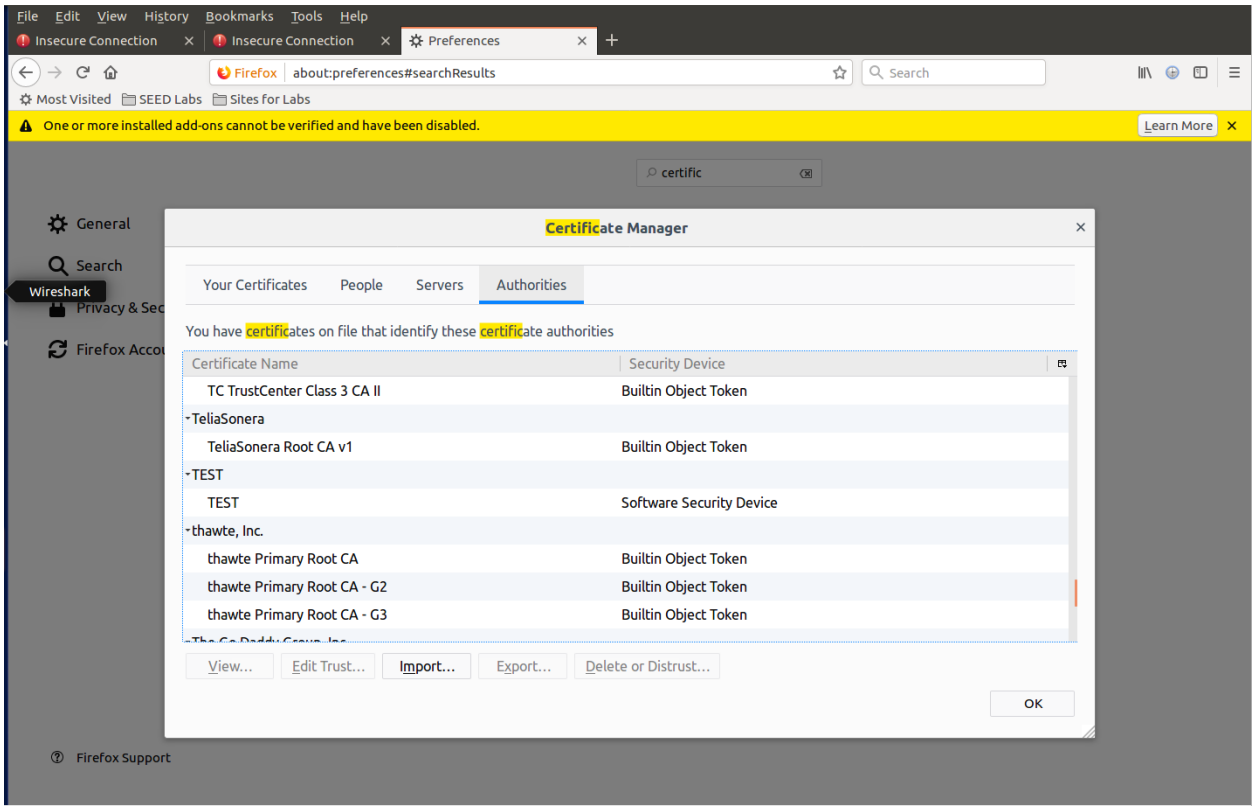
```
[11/21/22]seed@VM:~/.../lab4$ openssl s_server -cert server.pem -www
Enter pass phrase for server.pem:
Using default temp DH parameters
ACCEPT
```



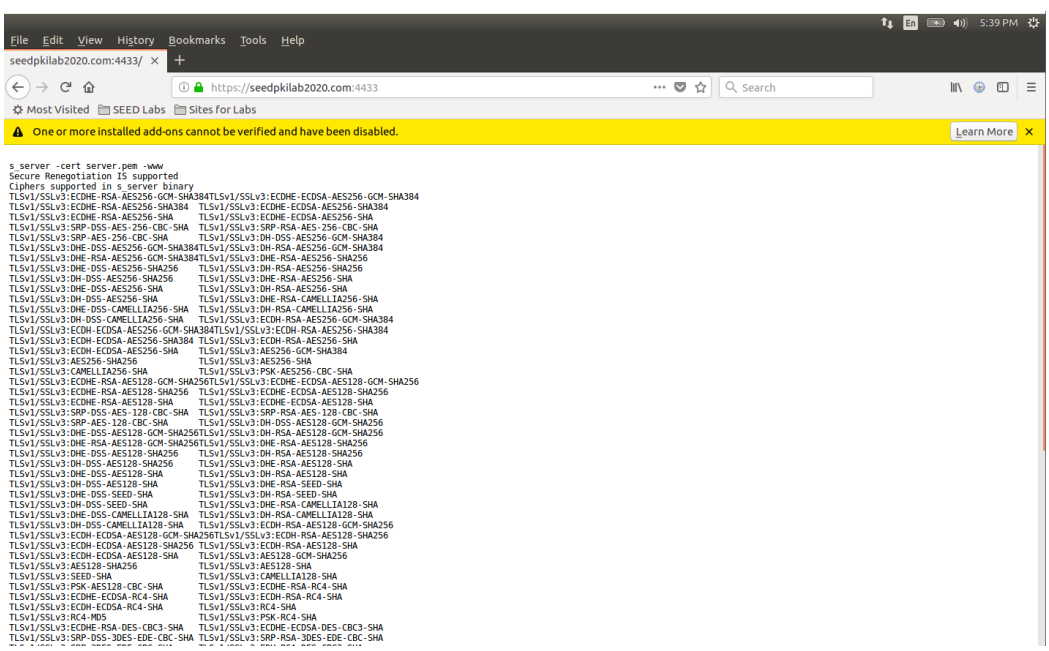
Step 3: Getting the browser to accept our CA certificate.



The result show that “TEST” is added ro accepted certificates



Step 4. Testing our HTTPS website.
We could access the website now:



Add “TEST” to the pem file:

```
[11/21/22]seed@VM:~/.../lab4$ cat server.pem
-----BEGIN RSA PRIVATE KEY-----
Proc-Type: 4,ENCRYPTED
DEK-Info: AES-128-CBC,075492BF7AC93C54EDC37C39DACEEC73

TESTxv0VRJPpth5+ylRo0Xf2LR1kPqGg32ZlgS2GY7eQwoJUc5ejGuk1KEfNblCO/WUW
Nqn12ok3Mij6sCu0c5khFCCAtAovORvZnvwuAgNoYJPp4Om2ui5BWEDPBOMJneq2
+JGmdQu/NHxFR4Z8VdD4yV2BJ16+uo+f2Cm1rFYYHtrUbT4bT3beSznWvSClxtwe
30oIUMfxV61m+udbWFQQzLde+kjU/OgrCv5cY1uvn8qGRaPRZYZyFVYMdaaYTcTo
swL96NVcviG3Z5AQGIahhwqEfYGHPE6u8sim3H7JvVkZWVAQDvErpkuVojdxCS8r
nsRRvjEw/l9IW/MVXctj2/HXfIRVxW5HDINmVZdHrz5ZNXnO5JfSA9apIr07SP7R
aBI4b1GNkMkO9Po2hsfLy++QMSpdekWjsQkHgPDAxZ6AO2Oz0Ks8FA9rZYfO/U6h
SsBbbsvEq4DsMkO6s5EUjAJsPVsQl4b+Q4F7c50VNnXldejhFMFIzovEhCb1SDST
+9m0pXPynLAIbuWmNu/xtw5tZuTE80eJKYSlCb0y4wrZ1rM1tkkitrGVxd+l0OBA
EuLrlck1IcQUajPMTZJHRQttKr9ikhKc6SWEFwdfJAUZdAWvk4efYFmndcZOdFFh
qPv8ScwUZjBFZY3hP6+LpCfDaBR+WDho2XMLNTwl/2iStj9VJA+X7ztWqUUbAlMB
```

Unable to start the server:

```
[11/21/22]seed@VM:~/.../lab4$ openssl s_server -cert server.pem -www
unable to load server certificate private key file
3071133376:error:0906D066:PEM routines:PEM_read_bio:bad end line:pem_lib.c:809:
[11/21/22]seed@VM:~/.../lab4$
```

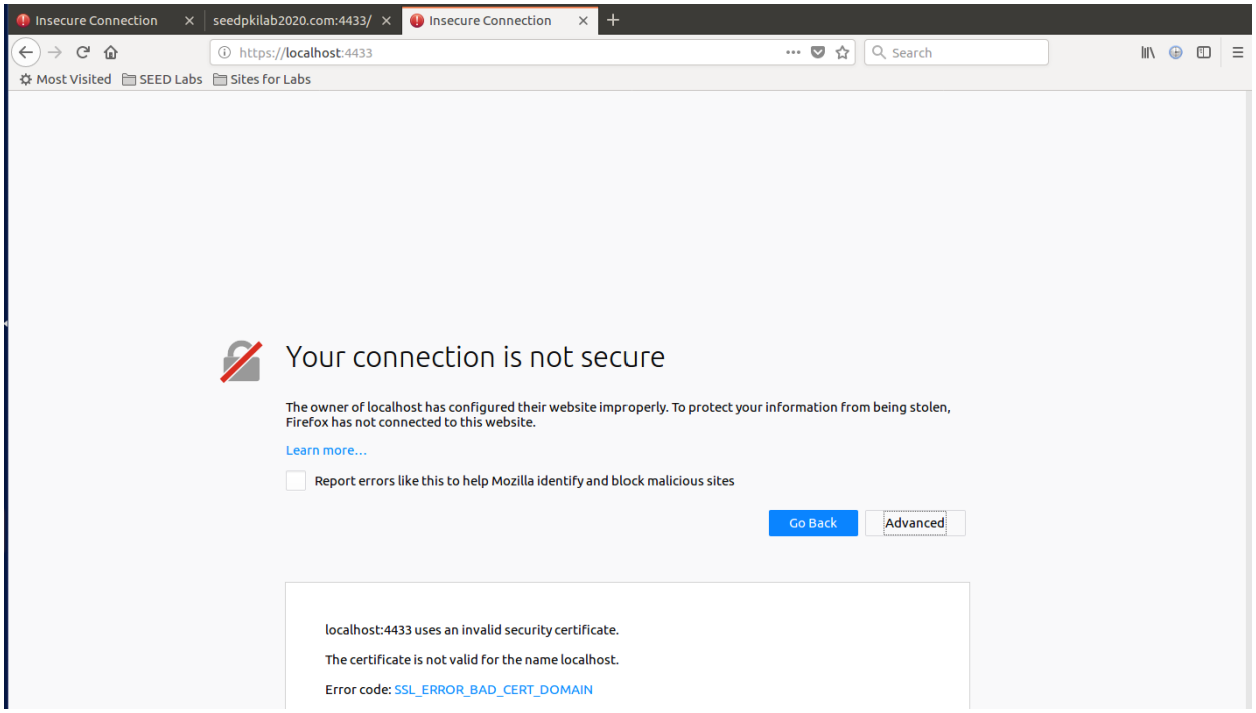
Add something to the end of pem file:

```
YW0zQHVjaS5lZHUwHhcNMjIxMTIxMjIyODIwWhcNMjMxMTIxMjIyODIwWjCBozEL
MAkGA1UEBhMCVVMxCzAJBgNVBAgMAkNBMQ8wDQYDVQQHDAZJUlZJTkUxGzAZBgNV
BAoMElNFRURQS0lMYWIyMDIwLmNvbTEbMBkGA1UECwwSU0VFRFBLSUxhYjIwMjAu
Y29tMRswGQYDVQQDDBJTRUVEUEtJTGFiMjAyMC5jb20xHzAdBgkqhkiG9w0BCQEW
EHpoaXl1YW0zQHVjaS5lZHUwgZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBAM/4
Q3VdiFXbapvWXWUV6VvnbOnpFX7PmJFPSHfdSSYraafMdA0UnroabIUmWZGRMVZI
N+7/uPaFRqy0VA6vfA3p8c/8HIaHk7dIVbfv30RM+F4s47M1TW9lALLa+/hgADfX
6UbtMFLtObpQcPJNHHw+K8y9qPz0FJM4+6YXbzPTAgMBAAGjezB5MAkGA1UdEwQC
MAAwLAYJYIZIAYb4QgENBB8WHU9wZW5TU0wgR2VuZXJhdGVkIENlcnRpZmljYXRl
MB0GA1UdDgQWBBTz0DBOzUTJURCzeLChzOTgGNjYEDAfBgNVHSMEGDAWgBTHNep6
87czAanHxnioXdbaEPHUfDANBgkqhkiG9w0BAQsFAAOCAQEAEsryIqKQOdlmQ67Z
YRWt1WZZz9q0j0MNdfC1mxvio3KZPuyrFV00rJCbPLNnkDT8ujFFF+GRiGaj2cdq
u35k0Eb5WBNXBHcfvwQUwCnU4j5WlGJLN+u1h0xTSwxz5QExWzja6tGGKWa+sXQ7
IYuZQgO/73uFVy9WeQzphBoQ9tFNEKBk0nSq0XbYltNJ8vX6UEixVuHlsS4UBgBW
tYXWhR4laYuHAC3yT+FRI4p5sY22ucCjsRIORhM3QwVtk+9SiUePsXvf4cFYAh2a
aXFrv9PgxHIA/xHeC1gTshpC1+IQ+v/BYRP0PlWIECBc72FdhJ+pRdpHJVrUsXfU
WsfoYg==
-----END CERTIFICATE-----
whatwhatwaht
[11/21/22]seed@VM:~/.../lab4$
```

We could start the server as long as it is not between start and end

```
Q3VdiFXbapvWXWUV6VvnbOnpFX7PmJFPSHfdSSYraafMdA0UnroabIUmWZGRMVZI
N+7/uPaFRqy0VA6vfA3p8c/8HIaHk7dIVbfv30RM+F4s47M1TW9lALLa+/hgADfX
6UbtMFLtObpQcPJNHHw+K8y9qPz0FJM4+6YXbzPTAgMBAAGjezB5MAkGA1UdEwQC
MAAwLAYJYIZIAYb4QgENBB8WHU9wZW5TU0wgR2VuZXJhdGVkIENlcnRpZmljYXRl
MB0GA1UdDgQWBBTz0DBOzUTJURCzeLChzOTgGNjYEDAfBgNVHSMEGDAWgBTHNep6
87czAanHxnioXdbaEPHUfDANBgkqhkiG9w0BAQsFAAOCAQEAEsryIqKQOdlmQ67Z
YRWt1WZZz9q0j0MNdfC1mxvio3KZPuyrFV00rJCbPLNnkDT8ujFFF+GRiGaj2cdq
u35k0Eb5WBNXBHcfvwQUwCnU4j5WlGJLN+u1h0xTSwxz5QExWzja6tGGKWa+sXQ7
IYuZQgO/73uFVy9WeQzphBoQ9tFNEKBk0nSq0XbYltNJ8vX6UEixVuHlsS4UBgBW
tYXWhR4laYuHAC3yT+FRI4p5sY22ucCjsRIORhM3QwVtk+9SiUePsXvf4cFYAh2a
aXFrv9PgxHIA/xHeC1gTshpC1+IQ+v/BYRP0PlWIECBc72FdhJ+pRdpHJVrUsXfU
WsfoYg==
-----END CERTIFICATE-----
whatwhatwaht
[11/21/22]seed@VM:~/.../lab4$ openssl s_server -cert server.pem -w
Enter pass phrase for server.pem:
Using default temp DH parameters
ACCEPT
```

And we can't visit the website on localhost, because the comma name of certificate is set to be http://seedpkilab2020.com/



Task 4: Deploying Certificate in an Apache-Based HTTPS Website

First we modify the config file /etc/apache2/sites-available/000-default.conf:

```
Terminal
        ServerName http://www.csrflabelgg.com
        DocumentRoot /var/www/CSRF/Elgg
</VirtualHost>
<VirtualHost *:80>
        ServerName http://www.csrflabattacker.com
        DocumentRoot /var/www/CSRF/Attacker
</VirtualHost>
<VirtualHost *:80>
        ServerName http://www.repackagingattacklab.com
        DocumentRoot /var/www/RepackagingAttack
</VirtualHost>
<VirtualHost *:80>
        ServerName http://www.seedlabclickjacking.com
        DocumentRoot /var/www/seedlabclickjacking
</VirtualHost>
<VirtualHost *:443>
        ServerName SEEDPKILab2020.com
        DocumentRoot /var/wwww/SEEDPKILab2020
        DirectoryIndex index.html
        SSLEngine On
        SSLCertificateFile /home/seed/Downloads/lab4/server.crt
        SSLCertificateKeyFile  /home/seed/Downloads/lab4/server.key
</VirtualHost>
-- INSERT --                                                63,15         Bot
```

Then we enable ssl and test syntax

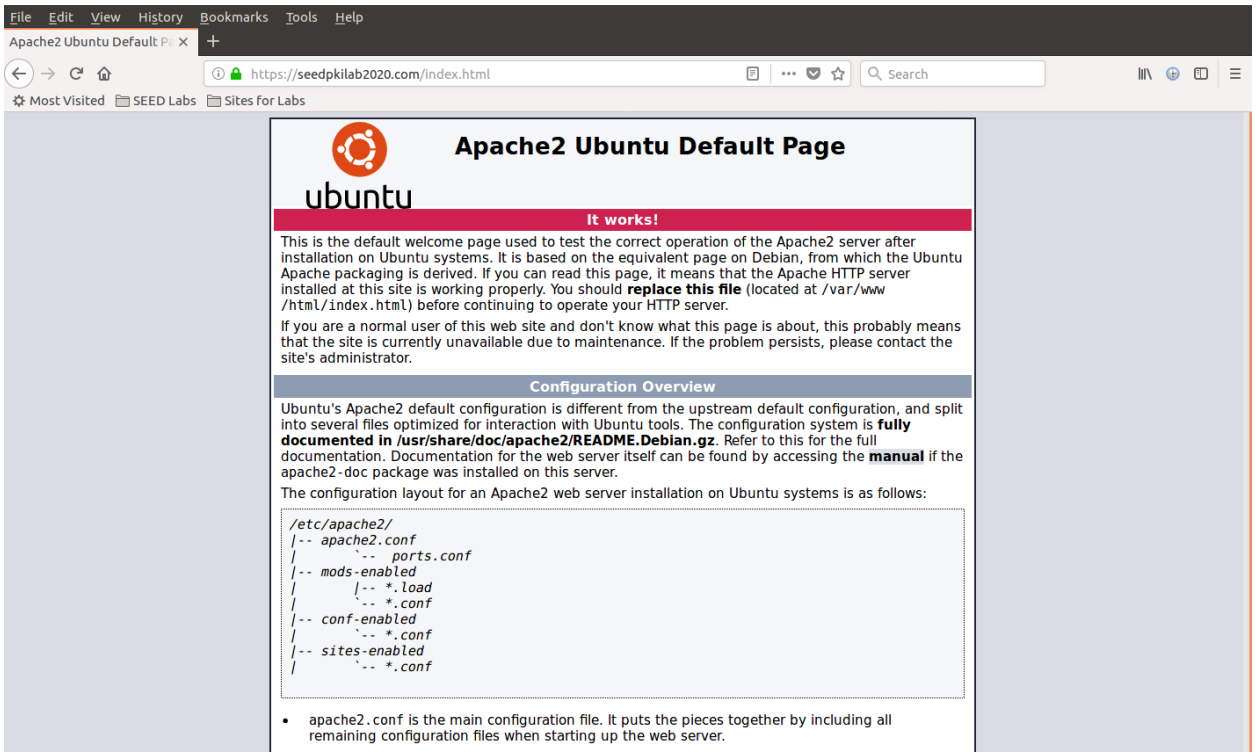
```
[11/21/22]seed@VM:.../sites-available$ sudo a2enmod ssl
Considering dependency setenvif for ssl:
Module setenvif already enabled
Considering dependency mime for ssl:
Module mime already enabled
Considering dependency socache_shmcb for ssl:
Enabling module socache_shmcb.
Enabling module ssl.
See /usr/share/doc/apache2/README.Debian.gz on how to configure SSL and create s
elf-signed certificates.
To activate the new configuration, you need to run:
  service apache2 restart
[11/21/22]seed@VM:.../sites-available$ sudo apachectl configtest
AH00112: Warning: DocumentRoot [/var/www/seedlabclickjacking] does not exist
AH00112: Warning: DocumentRoot [/var/wwww/SEEDPKILab2020] does not exist
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
[11/21/22]seed@VM:.../sites-available$
```

Then we enable the site and restart the service:

```
[11/21/22]seed@VM:.../sites-available$ sudo a2ensite default-ssl
Enabling site default-ssl.
To activate the new configuration, you need to run:
  service apache2 reload
[11/21/22]seed@VM:.../sites-available$ sudo service apache2 restart
Enter passphrase for SSL/TLS keys for SEEDPKILab2020.com:443 (RSA): ******
```

Setup the page for the site:

```
[11/21/22]seed@VM:.../sites-available$ sudo mkdir /var/www/SEEDPKILab2020
[11/21/22]seed@VM:.../sites-available$ sudo cp /var/www/html/index.html /var/www
/SEEDPKILab2020
```



Task 5: Launching a Man-In-The-Middle Attack

Target: google.com

Step 1: Setting up the malicious website

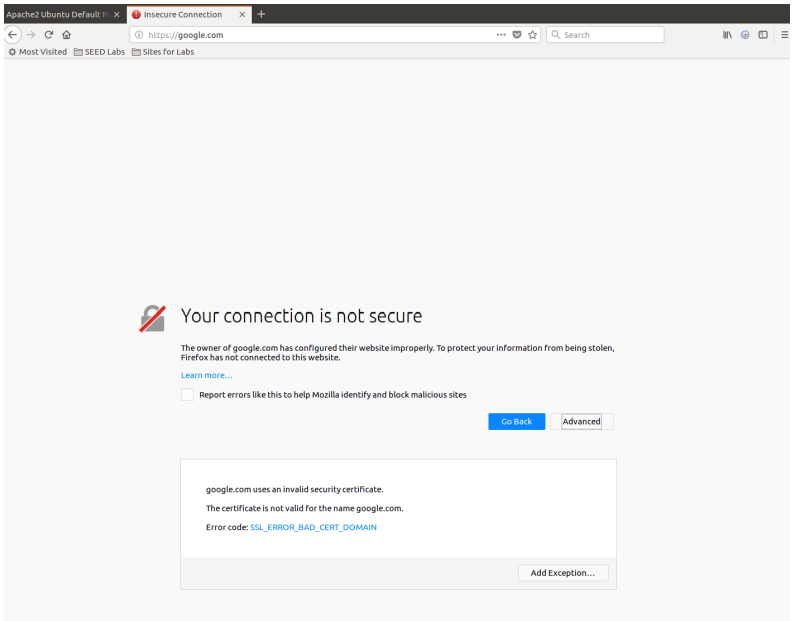
Add google to etc/hosts and points to 127.0.0.1

```
# The following lines are desirable for IPv6 capable hosts
::1     ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
127.0.0.1       User
127.0.0.1       Attacker
127.0.0.1       Server
127.0.0.1       www.SeedLabSQLInjection.com
127.0.0.1       www.xsslabelgg.com
127.0.0.1       www.csrflabelgg.com
127.0.0.1       www.csrflabattacker.com
127.0.0.1       www.repackagingattacklab.com
127.0.0.1       www.seedlabclickjacking.com
127.0.0.1    SEEDPKILab2020.com
127.0.0.1    google.com
[11/21/22]seed@VM:.../sites-available$
```

Step 2: Becoming the man in the middle
Add virtualhost entry the same as last step

```
<VirtualHost *:443>
        ServerName google.com
        DocumentRoot /var/www/SEEDPKILab2020
        DirectoryIndex index.html
        SSLEngine On
        SSLCertificateFile /home/seed/Downloads/lab4/server.crt
        SSLCertificateKeyFile  /home/seed/Downloads/lab4/server.key
</VirtualHost>
```

Step 3: Browse the target website.
Result:



The certificate used is for SEEDPKILab2020.com, but the domain we are trying to access is google.com

Task 6: Launching a Man-In-The-Middle Attack with a Compromised CA

the attacker can generate and sign a certificate that has a common name that is the same as the target site.

To generate and sign a certificate, we can run the following commands in the console:

openssl genrsa -aes128 -out gg-server.key 1024

openssl req -new -key gg-server.key -out gg-server.csr -config openssl.cnf

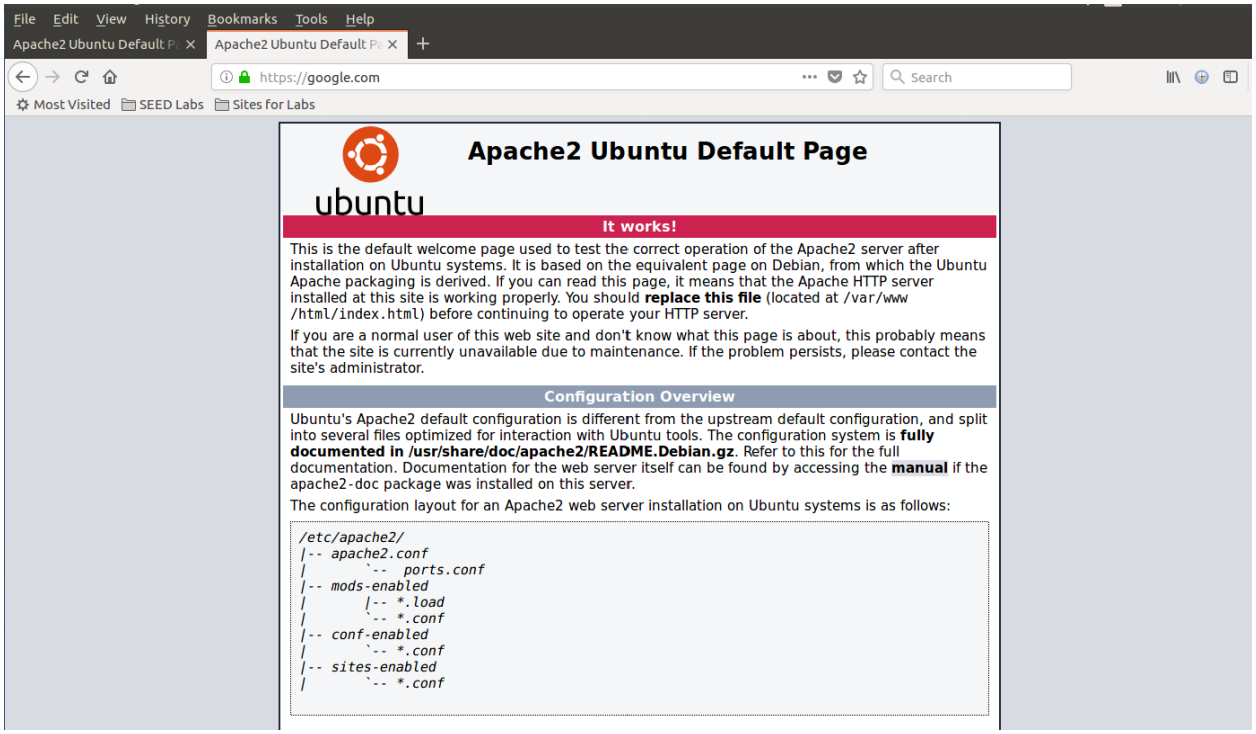
openssl ca -in gg-server.csr -out gg-server.crt -cert ca.crt -keyfile ca.key -config openssl.cnf

```
Terminal
            commonName                = google.com
            emailAddress              = zhiyuam3@uci.edu
        X509v3 extensions:
            X509v3 Basic Constraints:
                CA:FALSE
            Netscape Comment:
                OpenSSL Generated Certificate
            X509v3 Subject Key Identifier:
                3B:1A:42:3F:CE:20:77:6A:04:CB:DB:C2:73:84:89:6E:57:E0:92:80
            X509v3 Authority Key Identifier:
                keyid:C7:35:EA:7A:F3:B7:33:01:A9:C7:C6:78:A8:5D:D6:DA:10:F1:D4:7
C

Certificate is to be certified until Nov 22 04:08:52 2023 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
Data Base Updated
[11/21/22]seed@VM:~/.../lab4$ ls
ca.crt  demoCA         gg-server.csr  openssl.cnf  server.csr  server.pem
ca.key  gg-server.crt  gg-server.key  server.crt   server.key
[11/21/22]seed@VM:~/.../lab4$
```

Then create the entry in apache server:

```
        DocumentRoot /var/www/seedlabclickjacking
</VirtualHost>
<VirtualHost *:443>
        ServerName google.com
        DocumentRoot /var/www/SEEDPKILab2020
        DirectoryIndex index.html
        SSLEngine On
        SSLCertificateFile /home/seed/Downloads/lab4/gg-server.crt
        SSLCertificateKeyFile  /home/seed/Downloads/lab4/gg-server.key
</VirtualHost>
-- INSERT --                                          62,54-61      Bot
```

Result:


File Edit View History Bookmarks Tools Help
Apache2 Ubuntu Default Pa
Apache2 Ubuntu Default Pa
https://google.com
Search
Most Visited
SEED Labs
Sites for Labs
ubuntu
Apache2 Ubuntu Default Page
It works!
This is the default welcome page used to test the correct operation of the Apache2 server after installation on Ubuntu systems. It is based on the equivalent page on Debian, from which the Ubuntu Apache packaging is derived. If you can read this page, it means that the Apache HTTP server installed at this site is working properly. You should replace this file (located at /var/www/html/index.html) before continuing to operate your HTTP server.
If you are a normal user of this web site and don't know what this page is about, this probably means that the site is currently unavailable due to maintenance. If the problem persists, please contact the site's administrator.
Configuration Overview
Ubuntu's Apache2 default configuration is different from the upstream default configuration, and split into several files optimized for interaction with Ubuntu tools. The configuration system is fully documented in /usr/share/doc/apache2/README.Debian.gz. Refer to this for the full documentation. Documentation for the web server itself can be found by accessing the manual if the apache2-doc package was installed on this server.
The configuration layout for an Apache2 web server installation on Ubuntu systems is as follows:
/etc/apache2/
|-- apache2.conf
| `-- ports.conf
|-- mods-enabled
| |-- *.load
| `-- *.conf
|-- conf-enabled
| `-- *.conf
|-- sites-enabled
| `-- *.conf