Public-Key Infrastructure (PKI) Lab

Task 1: Becoming a Certificate Authority (CA)

Having the following files under demoCA folder:

```
[04/09/20]seed@VM:~/pki$ ls demoCA/
certs  crl  index.txt  newcerts  serial
```

Copy the config file from /usr/lib/ssl/openssl.cnf to this directory.

```
[04/09/20]seed@VM:~/pki$ ls
ca.crt  ca.key  demoCA  openssl.cnf
```

Then run the following command to generate the self-signed certificate for the CA:

```
[04/09/20]seed@VM:~/pki$ openssl req -new -x509 -keyout ca.key -out ca.crt -conf
ig openssl.cnf
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]:New York
Locality Name (eg, city) []:Syracuse
Organization Name (eg, company) [Internet Widgits Pty Ltd]:SU
Organizational Unit Name (eg, section) []:internet security
```

Fill in the fields then we would have the key file.

Task 2: Creating a Certificate

Step 1:

Generate the public and private key.

```
[04/09/20]seed@VM:~/pki$ 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:
Verifying - Enter pass phrase for server.key:
[04/09/20]seed@VM:~/pki$ ls
ca.crt  ca.key  demoCA  openssl.cnf  server.key
[04/09/20]seed@VM:~/pki$
```

Check the content:

```
-----BEGIN RSA PRIVATE KEY-----
Proc-Type: 4,ENCRYPTED
DEK-Info: AES-128-CBC,5ECA8AF7E6939F5672F608F02A3CA0DA

OJczBCcvvTwBWnKQP5i0dlBpKyJyEU07/k38p/CEjUMKjd5xDqqz5j3bRBuwp8az
SE8mk3dKRstvb3bvA2kqOStNEvIj2J4Awmj69sTAMaD3Vugkt2R5mJ9fD6CoX8Lg
3sb/gNCYfPXoFEPqe7fjwyIJEkLVvxY4ii0dWWco0/Rj/HvcqPImFWULRziznvBl
vAwY5QKDyC2Bv7Fp8xGfrw7Cs6SmAvRtrgMc00ALPOmH6s5huGAHBf36pQZagmKT
Ou7/aL+D3hyM4DK59yJQK4gMxn1tnyWViCBp86nrd1bOSBfU50SN2T5KkUZa6PYC
XFZLE7AjiABdYe9fKxwr2MQhPqFhNKYPAHwUvfdzFdDpXN7Uw48D9GLj4swp7JuR
ytI4nOM2KoCIg1h0VITi/yU6S2OzgycX5IddqIEQiHxfw5foIaJ1P1cjbWrts18/
VyRv29X00szqZ28339l5oZ6B7gU/Ybo5sHXlLWIRQnOhojfDmhpAKwDOqquLuhYj
lxwNt1B41F6LsmnunAYPl+mYknlAz2LrgGWE94Fm3po4kFCRJYDwD26z8VupZSg
1RnR85xO+OojtM3nABamXq9x2JoFd/599AqebGDDpa3bd1F+xt0l+mcecFP8cadh
jsVNOBPa8bHbEcJqFi4vI9ZlaeyNpZW4aA4tPlbwb7rNprfbQhW9/IJZaDsdiLPu
Bd3225+qWgnNCLz8HsAaEOVCP4DQVusPGrtTTiiHfBmH6ZHRV/QEl0w52jafGBfL
O5D5icmWCAltQZnVkvCFfaxJArSugNRGpGsvQmyDdyQSElAD2y/SfIj0wTQVErph
-----END RSA PRIVATE KEY-----
~
~
~
"server.key" 18L, 986C                                        13,1
```

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

```
[04/09/20]seed@VM:~/pki$ openssl req -new -key server.key -out server.csr -confi
g 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]:New York
Locality Name (eg, city) []:Syracuse
Organization Name (eg, company) [Internet Widgits Pty Ltd]:SU
Organizational Unit Name (eg, section) []:SU
Common Name (e.g. server FQDN or YOUR name) []:zfang18.com
Email Address []:zfang18@syr.edu

Please enter the following 'extra' attributes
to be sent with your certificate request
```

Step 3: Generating Certificates.

```
[04/09/20]seed@VM:~/pki$ 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
Certificate Details:
        Serial Number: 4096 (0x1000)
        Validity
            Not Before: Apr  9 18:42:22 2020 GMT
            Not After : Apr  9 18:42:22 2021 GMT
        Subject:
            countryName               = US
            stateOrProvinceName       = New York
            organizationName          = SU
            organizationalUnitName    = SU
            commonName                = zfang18.com
            emailAddress              = zfang18@syr.edu
        X509v3 extensions:
```

Task 3: Deploying Certificate in an HTTPS Web Server

Step 1: Configuring DNS.

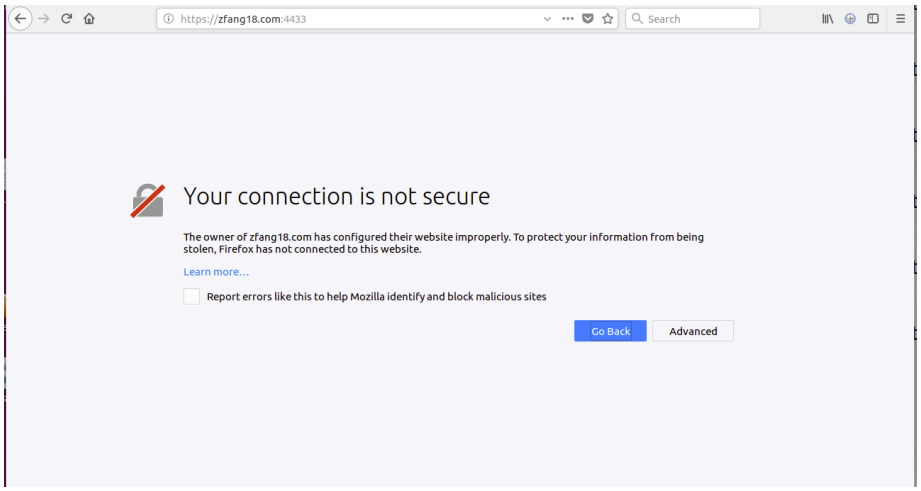
```
127.0.0.1       zfang18.com
```

Step 2: Configuring the web server.

Combine the secret key and certificate into one file and Launch the web server using server.pem

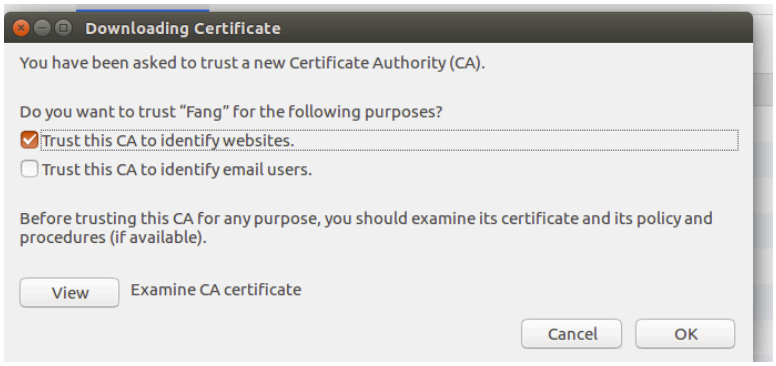
```
[04/09/20]seed@VM:~/pki$ cp server.key server.pem
[04/09/20]seed@VM:~/pki$ cat server.crt >> server.pem
[04/09/20]seed@VM:~/pki$ ls
ca.crt  demoCA       server.crt  server.key
ca.key  openssl.cnf  server.csr  server.pem
[04/09/20]seed@VM:~/pki$ openssl s_server -cert server.pem -www
Enter pass phrase for server.pem:
Using default temp DH parameters
ACCEPT

3070486208:error:1407609C:SSL routines:SSL23_GET_CLIENT_HELLO:http request:s23_s
rvr.c:394:
ACCEPT
3070486208:error:1407609C:SSL routines:SSL23_GET_CLIENT_HELLO:http request:s23_s
```

Browse this webpage with default port 4433, finding that this certificate is not trusted:

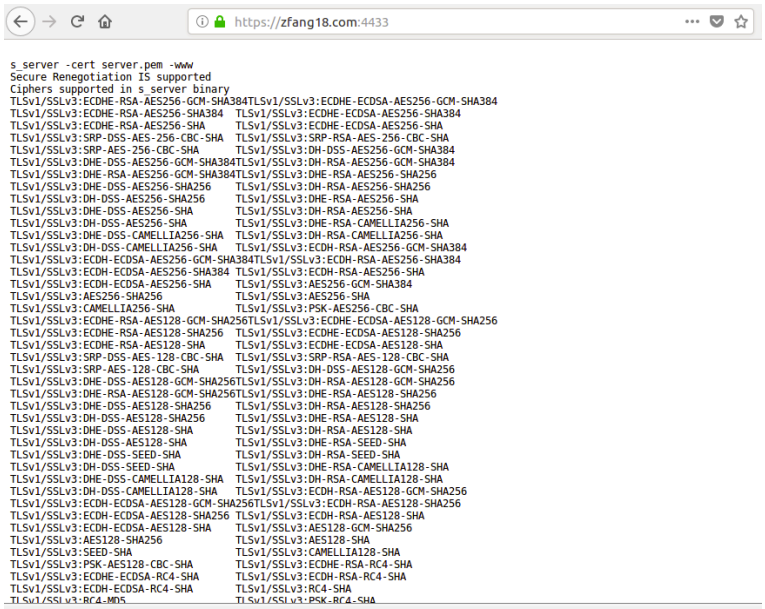


Step 3: Getting the browser to accept our CA certificate.

Load ca.crt into Firefox:

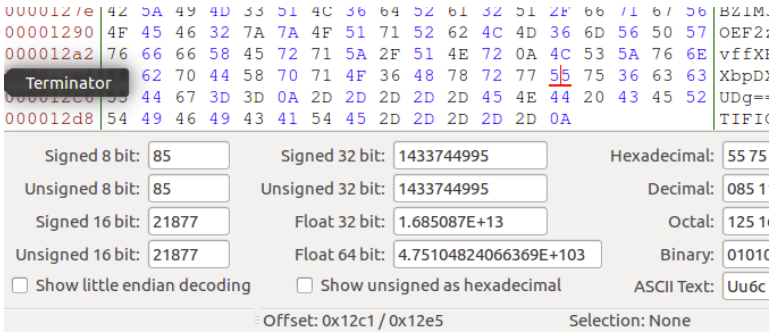


Then we are able to browse this webpage:



Step 4. Testing our HTTPS website.

Modify a single byte of server.pem, and restart the server, and reload the URL.

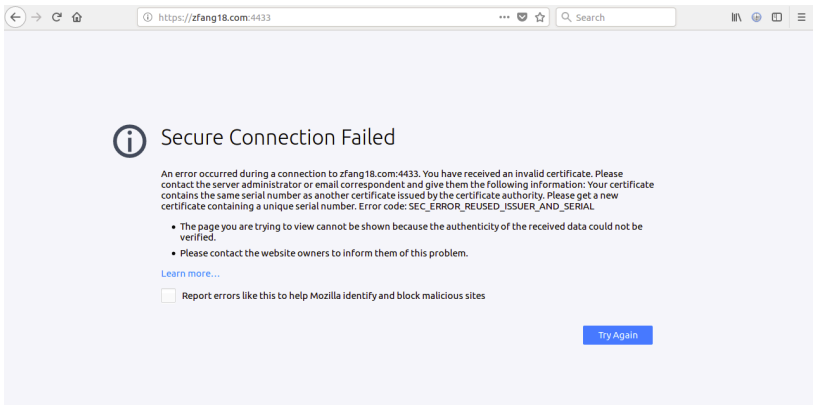


Simply modify one bit from 54 to 55:

```
[04/09/20]seed@VM:~/pki$ openssl s_server -cert server.pem -www
Enter pass phrase for server.pem:
Using default temp DH parameters
ACCEPT
3071211200:error:14094412:SSL routines:ssl3_read_bytes:sslv3 alert bad certifica
te:s3_pkt.c:1487:SSL alert number 42
3071211200:error:140780E5:SSL routines:ssl23_read:ssl handshake failure:s23_lib.
c:137:
ACCEPT
3071211200:error:14094412:SSL routines:ssl3_read_bytes:sslv3 alert bad certifica
te:s3_pkt.c:1487:SSL alert number 42
3071211200:error:140780E5:SSL routines:ssl23_read:ssl handshake failure:s23_lib.
c:137:
ACCEPT
```

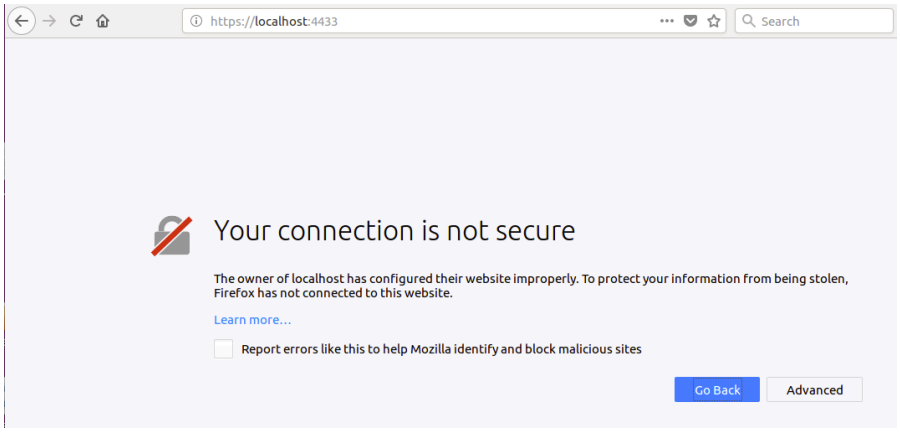
Then restart the server then browse:

We found that the connection failed, as we have modified the certificate one bit, it became invalid.



We restore the server.pem file and browse localhost:

I assume that the address on the certificate is diff from localhost so that it' s not validated.



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

To add an HTTPS website, we need to add a VirtualHost entry to the default-ssl.conf file

```
        </VirtualHost>
</IfModule>
<VirtualHost *:443>
ServerName zfang18.com
DocumentRoot /var/www/html
DirectoryIndex index.html
SSLEngine On
SSLCertificateFile /home/seed/pki/server.crt
SSLCertificateKeyFile  /home/seed/pki/server.key
</VirtualHost>
# vim: syntax=apache ts=4 sw=4 sts=4 sr noet
```

Test the Apache configuration file for errors

```
[04/09/20]seed@VM:.../sites-available$ sudo apachectl configtest
AH00112: Warning: DocumentRoot [/var/www/seedlabclickjacking] 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
```

Enable the SSL module

```
[04/09/20]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
```

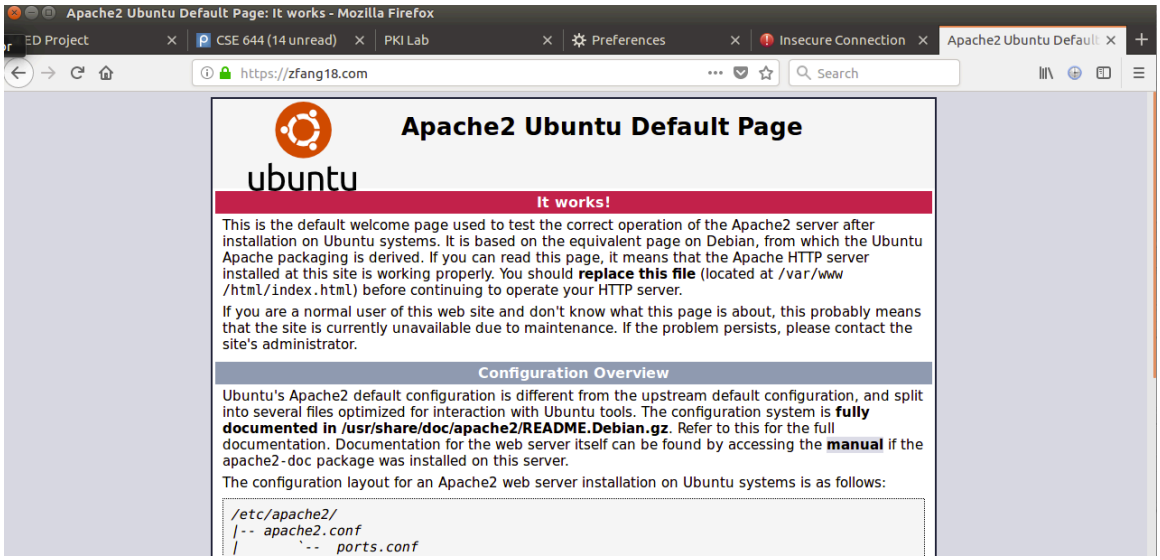
Enable the site we have just edited

```
[04/09/20]seed@VM:.../sites-available$ sudo a2ensite default-ssl
Enabling site default-ssl.
To activate the new configuration, you need to run:
  service apache2 reload
[04/09/20]seed@VM:.../sites-available$
```

Restart Apache

```
[04/09/20]seed@VM:.../sites-available$ sudo service apache2 restart
Enter passphrase for SSL/TLS keys for zfang18.com:443 (RSA): ****
```

Then we could see the apache2 default page on https://zfang18.com



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

Step 1

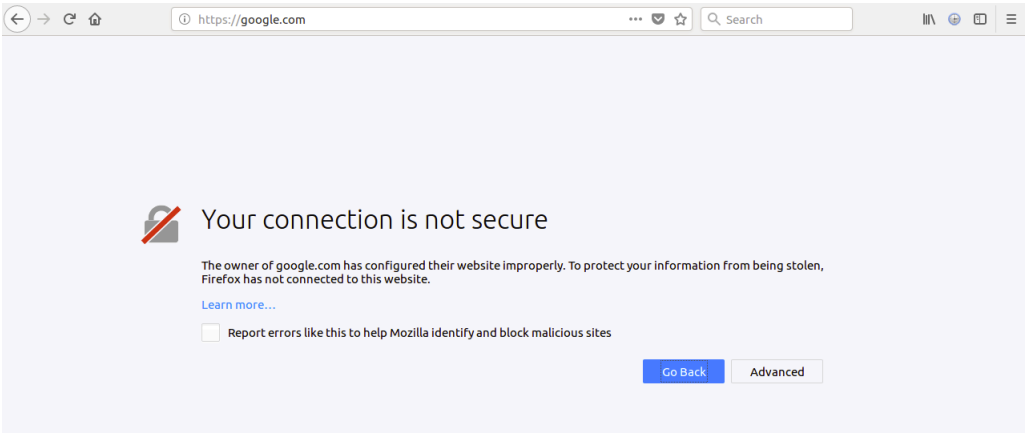
Modify the ServerName to google.com and then Restart the server.

```
</IfModule>
<VirtualHost *:443>
ServerName google.com
DocumentRoot /var/www/html
DirectoryIndex index.html
SSLEngine On
SSLCertificateFile /home/seed/pki/server.crt
SSLCertificateKeyFile  /home/seed/pki/server.key
</VirtualHost>
# vim: syntax=apache ts=4 sw=4 sts=4 sr noet
~
```

Modify the /etc/hosts file, tied it to local address

```
127.0.0.1       google.com
```

Of course we don’ t have google public key and private key, this connection will be treated as not secure.



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

As we have generated the public and private key before, to do this we start from the step2 that is to **Generate a Certificate Signing Request (CSR)**

```
[04/09/20]seed@VM:~/pki$ openssl req -new -key server.key -out server.csr -confi
g openssl.cnf
Enter pass phrase for server.key:
User interface error
unable to load Private Key
3070584512:error:0906A068:PEM routines:PEM_do_header:bad password read:pem_lib.c
:457:
[04/09/20]seed@VM:~/pki$ openssl req -new -key server.key -out google.csr -confi
g 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]:New York
Locality Name (eg, city) []:Syracuse
Organization Name (eg, company) [Internet Widgits Pty Ltd]:SU
```

Generating Certificates.

```
[04/09/20]seed@VM:~/pki$ openssl ca -in google.csr -out google.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: 4097 (0x1001)
        Validity
            Not Before: Apr  9 20:48:00 2020 GMT
            Not After : Apr  9 20:48:00 2021 GMT
        Subject:
            countryName               = US
            stateOrProvinceName       = New York
            organizationName          = SU
            organizationalUnitName    = su
            commonName                = google.com
            emailAddress              = zfang18@syr.edu
        X509v3 extensions:
```

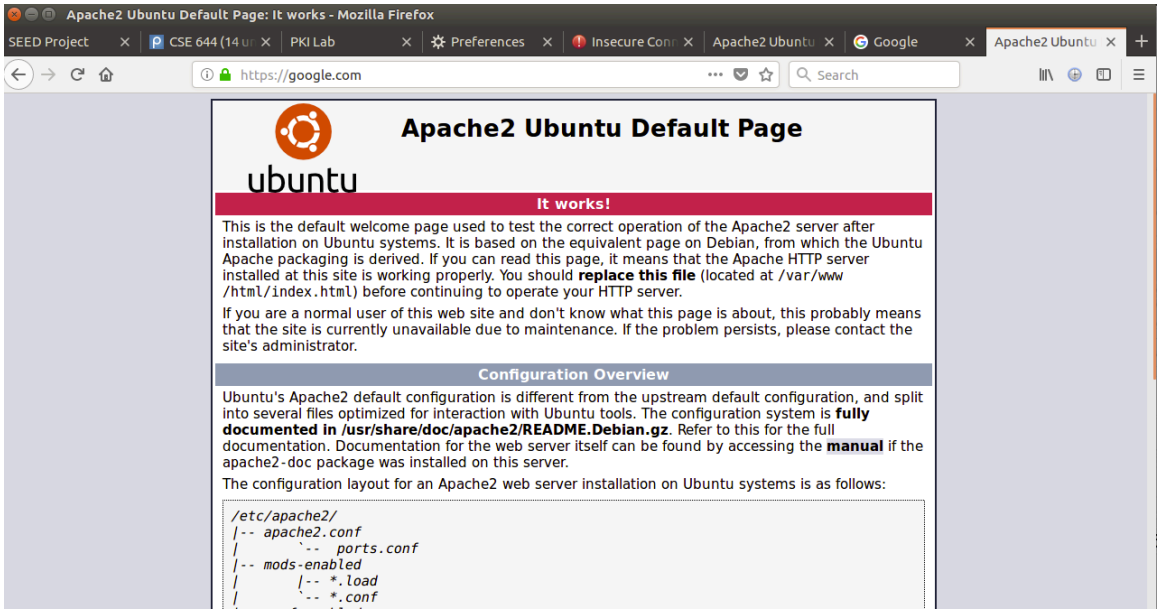
```
[04/09/20]seed@VM:~/pki$ ls
ca.crt  demoCA      google.csr   server.crt  server.key
ca.key  google.crt  openssl.cnf  server.csr  server.pem
[04/09/20]seed@VM:~/pki$
```

Then modify default-ssl.conf file as before:

```
<VirtualHost *:443>
ServerName google.com
DocumentRoot /var/www/html
DirectoryIndex index.html
SSLEngine On
SSLCertificateFile /home/seed/pki/google.crt
SSLCertificateKeyFile  /home/seed/pki/server.key
</VirtualHost>
# vim: syntax=apache ts=4 sw=4 sts=4 sr noet
~
```

Restart apache2 server:

```
[04/09/20]seed@VM:.../sites-available$ sudo service apache2 restart
Enter passphrase for SSL/TLS keys for google.com:443 (RSA): ****
[04/09/20]seed@VM:.../sites-available$
```



We can browse this default page with no warning