EAST WEST UNIVERSITY

CSE487, Section: 01

Cybersecurity, Law and Ethics

Spring 2022

Project Report

Securing a networked system with Public Key
(Infrastructure Implementing Transport Layer Security on HTTP for https:// connection)

Submitted to:

Rashedul Amin Tuhin

Senior Lecturer,

Department of CSE

East West University

Submitted by:

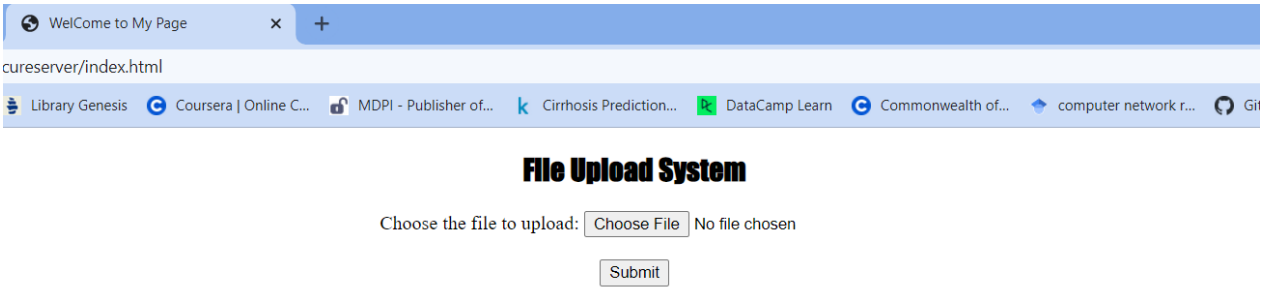
Student ID	Student Name
2019-1-60-034	Sadia Huq
2018-2-60-071	Fabiha Tasneem
2018-2-60-129	Sanjida Kater

Requirements:

- Configuration of Certification Authority AcmeCA with AcmeRootCA as the RootCA.
- Configuration of the Web Server with Apache2 on a Linux Host.
- DNS configuration for www.verysecureserver.com
- Firewall configuration to allow necessary ports (53, 80, 443) only
- CSR Configuration and Generation for the www.verysecureserver.com
- Transferring the CSR to AcmeCA.
- Certification process (Verification and Certificate Generation from CSR)
- Transferring the certificate from AcmeCA to www.verysecureserver.com
- Installation of the signed the SSL certificate in the server of www.verysecureserver.com
- Making the system trust Acme-RootCA
- Implementation of a simple file uploading page in the server.
- Verifying the security of the connection by inspection (the padlock icon), and with wireshark from another computer.
- Revoke the certificate issued to www.verysecureserver.com from the CA and distribute the first CRL. [bonus]
- Verifying the revocation of previous certificate from the CRL (no padlock icon).
- Configuring IDS [bonus]

Step1)

we created a folder on the C drive named 'verysecureserver' which is the web server name for this project and on that folder, we have created a simple html page.



Step2)

Configuring DNS:

We configured DNS (Domain Name System) for our public IP.

Local disk (C:)→Windows→System32→drivers→etc→hosts (open with notepad or any text editor)

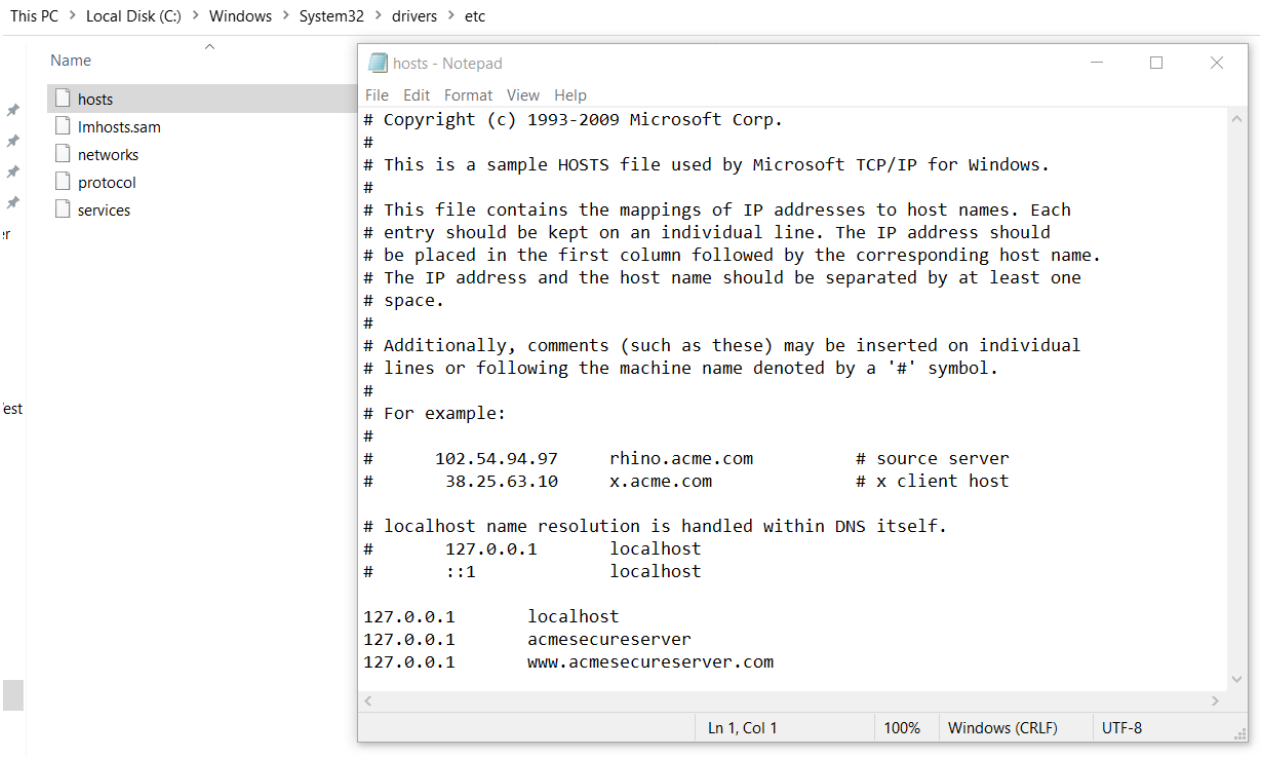
And then write the DNS configurations for our server and write the localhost Ip address and the server's name www.verysecureserver.com at the end of the file.

hosts:

127.0.0.1 localhost

127.0.0.1 acmesecureserver

127.0.0.1 www.verysecureserver.com



Step3)

Configuring of another DNS file in the XAMPP:

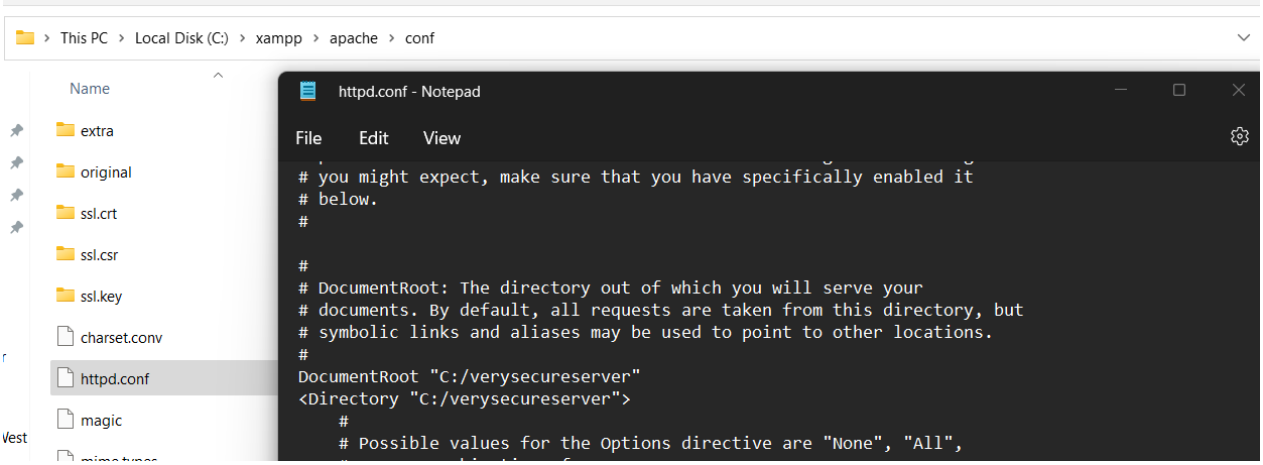
Now, configure another DNS file in the XAMPP.

xampp→apache→conf→

httpd.conf:

DocumentRoot "C:/verysecureserver"

<Directory "C:/verysecureserver">



Step4)

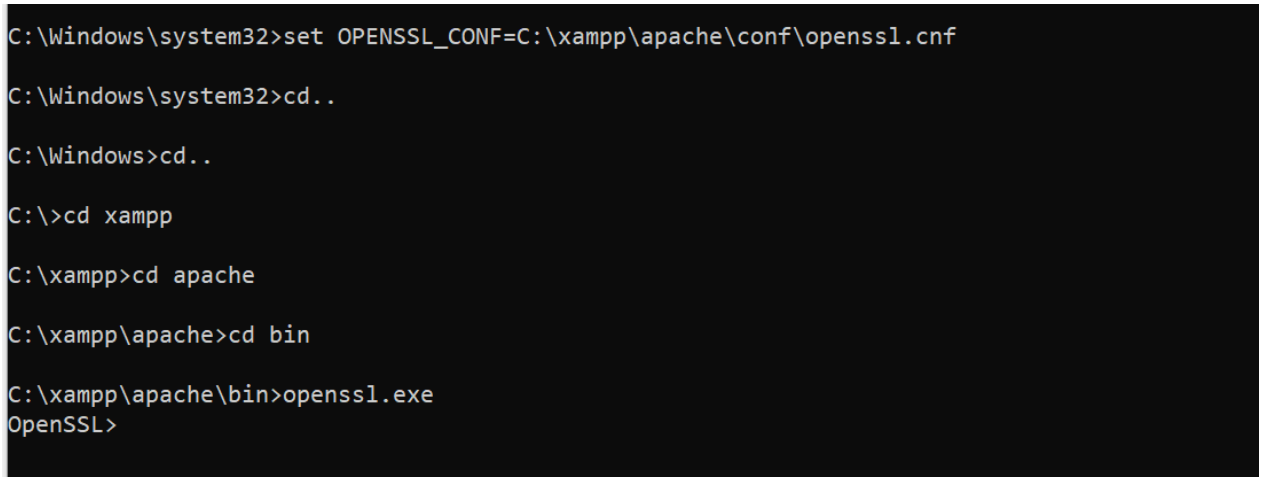
For openssl environment path configuration:

For secured connection we signed the server certificates using openssl.

- First, installed OPENSSL in our Windows.

And for openssl environment path run the command prompt as administrator and write the below command:

set OPENSSL_CONF=C:\xampp\apache\conf\openssl.cnf



- To generate a pair of private key and public Certificate Signing Request (CSR) for a web server, “server”, use the following command for creating a server certificate→

~ req -newkey rsa:2048 -nodes -keyout server.key -out server.csr

Now enter following details. Common name: www.verysecureserver.com

~ x509 -signkey server.key -in server.csr -req -days 365 -out server.crt

control c

- For creating a sub root CA certificate go to openssl and write→

~ req -newkey rsa:2048 -keyout subrootCA.key -out subrootCA.csr

Now enter following details. Common name: AcmeCA

~ x509 -signkey subrootCA.key -in subrootCA.csr -req -days 365 -out subrootCA.crt

control c

- For creating a sub root CA certificate go to openssl and write→

~ req -x509 -sha256 -days 1825 -newkey rsa:2048 -keyout rootCA.key -out rootCA.crt

Now enter following details. Common name: AcmeRootCA

- Now open two .ext file as we created the three certificates and now it's the time for signing those certificates:

We need to go to xampp→apache→bin

domain.ext:
authorityKeyIdentifier=keyid,issuer
basicConstraints=CA: FALSE
subjectAltName = @alt_names
[alt_names]
DNS.1 =www.verysecureserver.com
DNS.2 =127.0.0.1

root.ext:
authorityKeyIdentifier=keyid,issuer
basicConstraints=CA: TRUE
subjectAltName = @alt_names

[alt_names]
DNS.1 =www.verysecureserver.com
DNS.2 =127.0.0.1

- In the command prompt:

Signing subrootCA certificate with rootCA certificate→
~ x509 -req -CA rootCA.crt -CAkey rootCA.key -in subrootCA.csr -out subrootCA.crt -days 365 -CAcreateserial -extfile root.ext

- For checking the subrootCa certificate→

~ x509 -text -noout -in subrootCA.crt

~ x509 -in subrootCA.crt -outform der -out subrootCA.der

- Now, Exporting the subrootCA key file in subrootCA pfx file→

~ pkcs12 -inkey subrootCA.key -in subrootCA.crt -export -out subrootCA.pfx

- Signing server certificate with subrootCA certificate→

~ x509 -req -CA subrootCA.crt -CAkey subrootCA.key -in server.csr -out server.crt -days 365 -CAcreateserial -extfile domain.ext

~ x509 -in server.crt -outform der -out server.der

- Exporting the server key file in the server .pfx file→

~ pkcs12 -inkey server.key -in server.crt -export -out server.pfx

- Replacing the RSA encryption from the server and subrootCA key for setting the validity→

~ rsa -in server.key -out server.key
~ rsa -in subrootCA.key -out subrootCA.key

Step5)

Installing Certificate:

Open C drive then to the xampp folder then apache→bin then we will see our certificates then we need to install it first the rootCA.cert in local machine→trusted root certificate authorities→finish then our installation will be completed in this way we need to install server.crt, subrootCA.pfx and the subrootCA.cert will be already installed by installing the rootCA.cert.

After that now we need to copy the server.crt, server.csr and the server.key file and paste it to the apache→conf file into the ssl.crt, ssl.csr, ssl.key.

Step5)

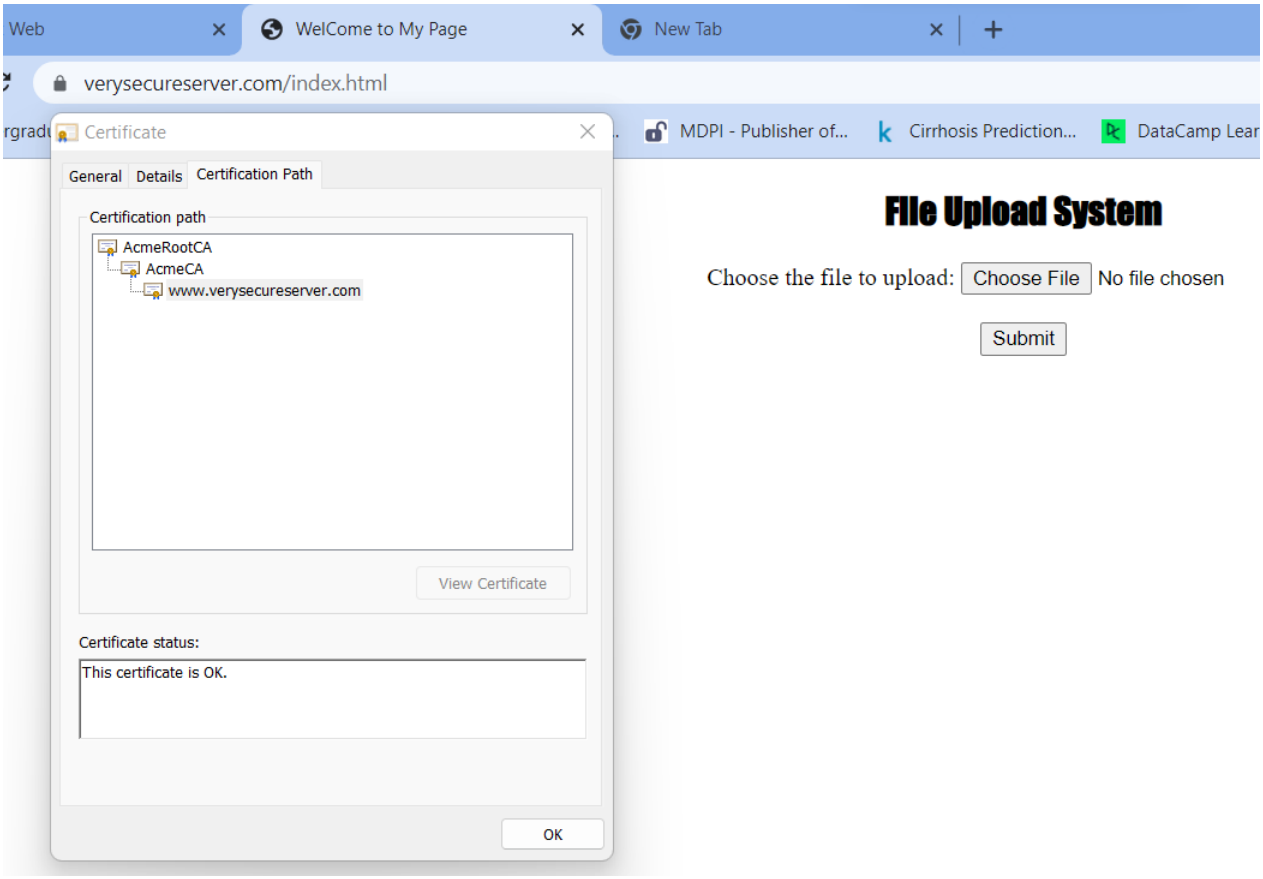
Creating certificate:

For that we need to go to the xampp→apache→conf→extra→httpd-vhosts.conf

Configuring httpd-vhosts:

```
<VirtualHost *:443>
    DocumentRoot "C:/verysecureserver/"
    ServerName verysecureserver
    ServerAlias www. verysecureserver.com
    SSLEngine on
    SSLCertificateFile "conf/ssl.crt/server.crt"
    SSLCertificateKeyFile "conf/ssl.key/server.key"
</VirtualHost>
```

Finally, open xampp and start Apache and go to our website which is running with SSL Certificate.



Step 6)

Revocation of certificate:

Open three files in the bin folder which is index.txt, serial.txt, crlnumber.txt where the serial number will be stored.

- Open openssl.exe to revoke the certificate issued to acmesecureserver.com from the AcmeCA →

ca -config subrootCA.conf -revoke server.crt

- To generate revocation crl file →

ca -config subrootCA.conf -gencrl -out rev.crl

- To see the revocation file in the form of text →

crl -in rev.crl -noout -text

subrootCA.conf:

```
[ca]
default_ca = CA_default
[CA_default]
dir =C:/xampp/apache/bin
certs = $dir
crl_dir = $dir
new_certs_dir = $dir
database = $dir/index.txt
serial = $dir/serial.txt
RANDFILE = $dir/private/. rand
private_key = $dir/subrootCA.key
certificate = $dir/subrootCA.crt
crlnumber = $dir/crlnumber.txt
crl = $dir/crl/ca.crl
default_crl_days = 30
default_md = sha256
name_opt = ca_default
cert_opt = ca_default
default_days = 365
preserve = no
policy = policy_loose
[ policy_strict ]
countryName = supplied
stateOrProvinceName = supplied
```

```
organizationName = supplied
organizationalUnitName = optional
commonName = supplied
emailAddress = optional
[ policy_loose ]
countryName = optional
stateOrProvinceName = optional
localityName = optional
organizationName = optional
organizationalUnitName = optional
commonName = supplied
emailAddress = optional
[ req ]
# Options for the req tool, man req.
default_bits = 2048
distinguished_name = req_distinguished_name
string_mask = utf8only
default_md = sha256
# Extension to add when the -x509 option is used.
x509_extensions = v3_ca
[ req_distinguished_name ]
countryName = Country Name (2 letter code)
stateOrProvinceName = State or Province Name
localityName = Locality Name
0.organizationName = Organization Name
organizationalUnitName = Organizational Unit Name
commonName = Common Name
emailAddress = Email Address
countryName_default = BD
stateOrProvinceName_default = Dhaka
0.organizationName_default = Acme
[ v3_ca ]
# Extensions to apply when createing root ca
# Extensions for a typical CA, man x509v3_config
subjectKeyIdentifier = hash
authorityKeyIdentifier = keyid:always,issuer
basicConstraints = critical, CA:true
keyUsage = critical, digitalSignature, cRLSign, keyCertSign
[ v3_intermediate_ca ]
# Extensions to apply when creating intermediate or sub-ca
# Extensions for a typical intermediate CA, same man as above
subjectKeyIdentifier = hash
authorityKeyIdentifier = keyid:always,issuer
```

```
#pathlen:0 ensures no more sub-ca can be created below an intermediate
basicConstraints = critical, CA:true, pathlen:0
keyUsage = critical, digitalSignature, cRLSign, keyCertSign
crlDistributionPoints = @crl_dist_points
[ server_cert ]
# Extensions for server certificates
basicConstraints = CA:FALSE
nsComment = "OpenSSL Generated Server Certificate"
subjectKeyIdentifier = hash
authorityKeyIdentifier = keyid,issuer:always
keyUsage = nonRepudiation, digitalSignature, keyEncipherment
extendedKeyUsage = serverAuth
subjectAltName = @alt_names
[alt_names]
DNS.1 = www.verysecureserver.com
DNS.2 = 127.0.0.1
```

By this the revocation will be done and we will see the time and date of the revocation in the terminal.