EAST WEST UNIVERSITY

CSE487: Cybersecurity, Law and Ethics

[Summer 2022]

Section: 03

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

Project Report

Submitted to:

Rashedul Amin Tuhin

Senior Lecturer

Department of Computer Science & Engineering

East West University

Submitted by:

Student ID	Student Name
2019-1-60-068	Md. Shahadat Anik Sheikh
2019-1-60-036	Hasib Ar Rafiul Fahim
2018-3-60-088	Rashik Buksh Rafsan

Project Explanation:

Requirements:

1. Configuration of Certification Authority AcmeCA
2. Configuration of the Web Server
3. Firewall configuration to allow necessary ports (53, 80, 443) only
4. DNS configuration for www.oursecureserver.com
5. CSR Configuration and Generation for the www. oursecureserver.com
6. Certification process (Verification and Certificate Generation from CSR)
7. Installation of the signed SSL certificate in the server
8. Making the system trust Acme-RootCA
9. Implementation of a simple file uploading page in the server.
10. Verifying the security of the connection by inspection (the padlock icon), and with Wireshark from another client.
11. Revoke the certificate issued to acmesecureserver.com from the AcmeCA and distribute the first CRL.
12. Verifying the revocation of previous certificate from the CRL (no padlock icon).

Creating the Website:

First, we created a simple website which uploads file. The files are static not stored anywhere.

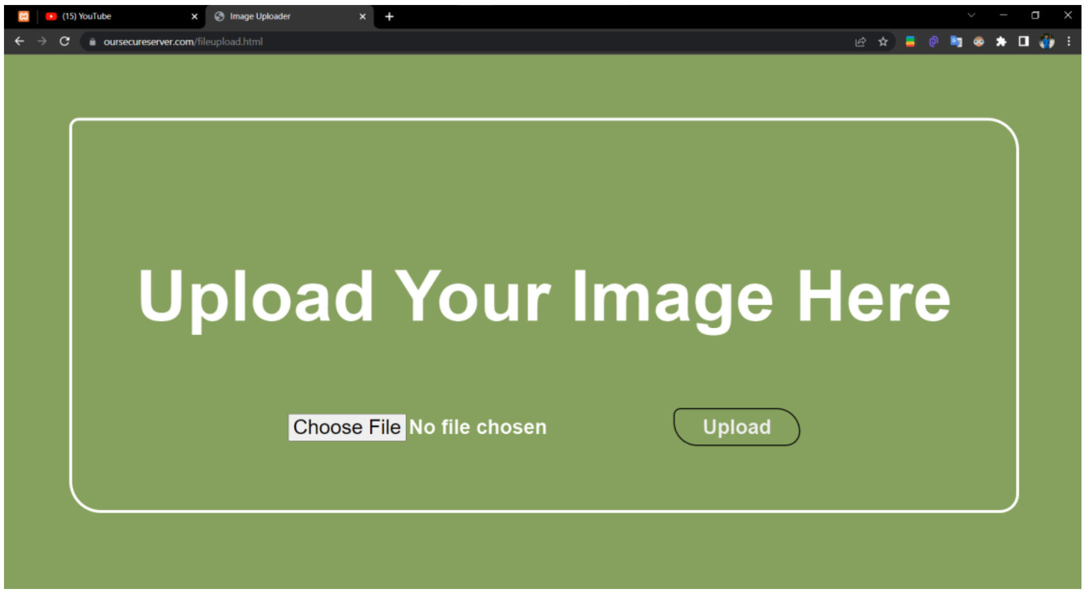


Figure 1: Created HTML website

For configuring of certification authority and implementing the transport layer security over http, we must follow some steps. All the steps are given below:

Step-1: Configuring DNS:

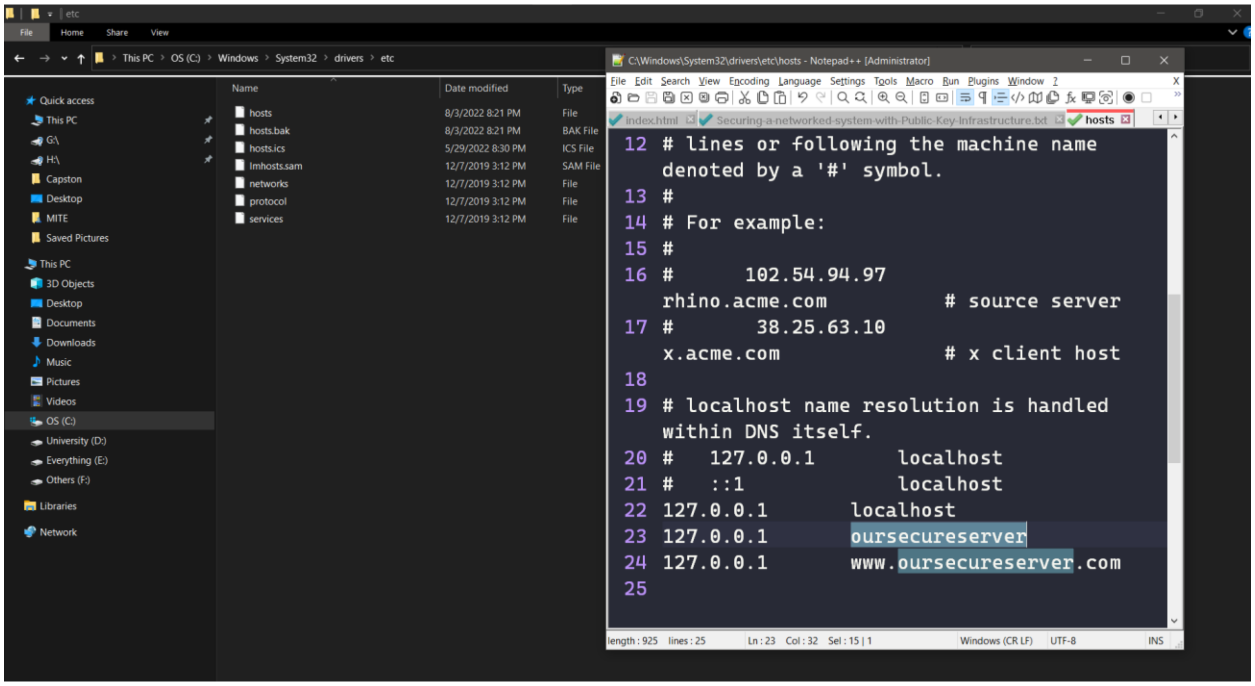
We configured DNS (Domain Name System) for our public IP. So, at first, we need to go to

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

And after going there we need to write the DNS configurations for our server and write the localhost Ip address and the server's name www.oursecureserver.com

hosts:

```
127.0.0.1      localhost
127.0.0.1      oursecureserver
127.0.0.1      www.oursecureserver.com
```



Name	Date modified	Type
hosts	8/3/2022 8:21 PM	File
hosts.bak	8/3/2022 8:21 PM	BAK File
hosts.ics	5/29/2022 8:30 PM	ICS File
lmhosts.sam	12/7/2019 3:12 PM	SAM File
networks	12/7/2019 3:12 PM	File
protocol	12/7/2019 3:12 PM	File
services	12/7/2019 3:12 PM	File

```
# lines or following the machine name denoted by a '#' symbol.
#
# For example:
#
#      102.54.94.97     rhino.acme.com          # source server
#       38.25.63.10     x.acme.com              # x client host

# localhost name resolution is handled within DNS itself.
#    127.0.0.1       localhost
#    ::1             localhost
127.0.0.1       localhost
127.0.0.1       oursecureserver
127.0.0.1       www.oursecureserver.com

```

Figure 2: DNS for our public IP

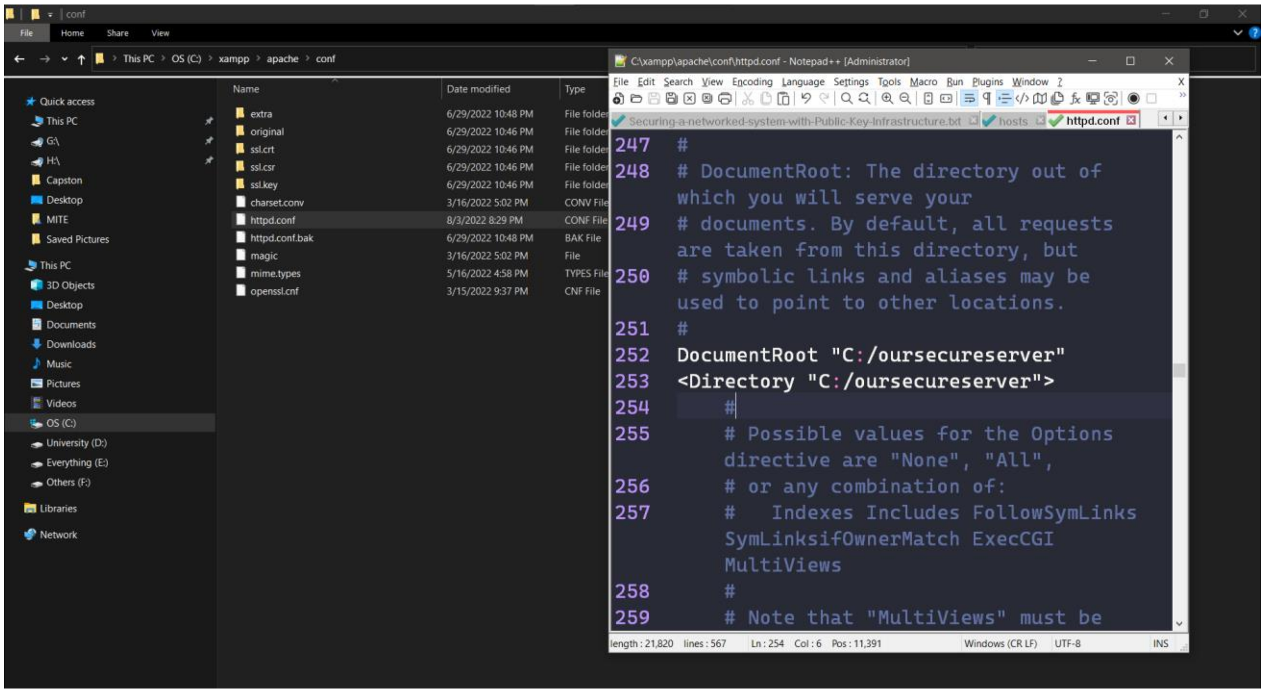
Step-2: Configuring of another DNS file in the xampp:

Now, we need to configure another DNS file in the xampp for that we need to go to the document root for that: xampp→ apache→ conf→

<u>httpd.conf</u>:

DocumentRoot "C:/oursecureserver"

<Directory "C:/oursecureserver">



Name	Date modified	Type
extra	6/29/2022 10:48 PM	File folder
original	6/29/2022 10:46 PM	File folder
ssl.crt	6/29/2022 10:46 PM	File folder
ssl.csr	6/29/2022 10:46 PM	File folder
ssl.key	6/29/2022 10:46 PM	File folder
charset.conv	3/16/2022 5:02 PM	CONV File
httpd.conf	8/3/2022 8:29 PM	CONF File
httpd.conf.bak	6/29/2022 10:48 PM	BAK File
magic	3/16/2022 5:02 PM	File
mime.types	5/16/2022 4:58 PM	TYPES File
openssl.cnf	3/15/2022 9:37 PM	CNF File

```
#
# DocumentRoot: The directory out of which you will serve your
# documents. By default, all requests are taken from this directory, but
# symbolic links and aliases may be used to point to other locations.
#
DocumentRoot "C:/oursecureserver"
<Directory "C:/oursecureserver">
    #
    # Possible values for the Options directive are "None", "All",
    # or any combination of:
    #   Indexes Includes FollowSymLinks SymLinksifOwnerMatch ExecCGI MultiViews
    #
    # Note that "MultiViews" must be
```

conf
File
Home
Share
View
This PC > OS (C:) > xampp > apache > conf
Quick access
This PC
G:\
H:\
Capston
Desktop
MITE
Saved Pictures
This PC
3D Objects
Desktop
Documents
Downloads
Music
Pictures
Videos
OS (C:)
University (D:)
Everything (E:)
Others (F:)
Libraries
Network
C:\xampp\apache\conf\httpd.conf - Notepad++ [Administrator]
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
Securing-a-networked-system-with-Public-Key-Infrastructure.txt
hosts
httpd.conf
length : 21,820 lines : 567 Ln : 254 Col : 6 Pos : 11,391 Windows (CR LF) UTF-8 INS

Figure 3: xampp dns setup

Step-3: For openssl environment path configuration:

Now, our website is up and running but it is not secured because there is no SSL certificate. In this part, we will create a signed certificate for our server. We used OpenSSL tool to generate private key and certificate request for our server. We used our DNS as the name for signing the certificate. Then, we used OpenSSL to generate a signed certificate for our corresponding DNS. The certificate request CSR (Certificate Signing Request) must be created with a key length of 2048 as the hash algorithm. It checks if DNS is owned by us or not. After successful check, it gives us the signed certificate.

First, we need to install OPENSSL in our Windows. For that, we need to run cmd as administrator. Then we used the following command, **choco install openssl**

And for openssl environment path we need to go to:

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

Now we need to go to C: by using cd command in the terminal after that we will go to

cd xampp → cd apache→ cd bin → openssl.exe

Then openssl will be on.

To generate a pair of private key and public Certificate Signing Request (CSR) for a web server, "server", use the following command:

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

This creates two files. The file myserver.key contains a private key. The private key is used as input in the command to generate a Certificate Signing Request (CSR). We will now be asked to enter details to be entered into the CSR. What we are about to enter is what is called a Distinguished Name or a DN. **Common name:** www.oursecureserver.com

Password for creating certificates: anik123

```
Country Name (2 letter code) [AU]: BD
State or Province Name (full name) [Some-State]: Dhaka
Locality Name (eg, city) []: Dhaka
Organization Name (eg, company) []: EWU
Organizational Unit Name (eg, section) []: CSE
Common Name (server name) []: www.oursecureserver.com
E-mail Address []: shahadatanik777@gmail.com
Please enter the following 'extra' attributes to be sent with your certificate request.
A challenge password []: (anik123)
```

Our CSR will now have been created.

To verify the certificate:

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

Now for creating the subroot CA certificate at first, we need to control c and again start the openssl otherwise we can get error because we are going to create another certificate for the subroot. **Common Name: AcmeCA** and it will ask for a **PEM password** which is **anik123**

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

```
Country Name (2 letter code) [AU]: BD
State or Province Name (full name) [Some-State]: Dhaka
```

```
Locality Name (eg, city) []: Dhaka
Organization Name (eg, company) []: EWU
Organizational Unit Name (eg, section) []: CSE
Common Name (server name) []: AcmeCA
E-mail Address []: shahadatanik777@gmail.com
Please enter the following ‘extra’ attributes to be sent with your certificate request.
A challenge password []: (anik123)
```

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

Now for creating the root CA certificate at first, we need to control c and again start the openssl otherwise we can get error because we are going to create another certificate for the root. **Common Name: Acme-RootCA** and it will ask for a **PEM password which is anik123**

```
Country Name (2 letter code) [AU]: BD
State or Province Name (full name) [Some-State]: Dhaka
Locality Name (eg, city) []: Dhaka
Organization Name (eg, company) []: EWU
Organizational Unit Name (eg, section) []: CSE
Common Name (server name) []: Acme-RootCA
E-mail Address []: shahadatanik777@gmail.com
Please enter the following ‘extra’ attributes to be sent with your certificate request.
A challenge password []: (anik123)
```

Now we must open two **.ext** file because as we created the three certificates and now it’s the time for signing those certificates so the files are:

We need to go: **xampp→ apache→ bin**

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

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

DNS.1 =www.oursecureserver.com
DNS.2 =127.0.0.1

Now, 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
And it will ask for the **PEM pass which is anik123**

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

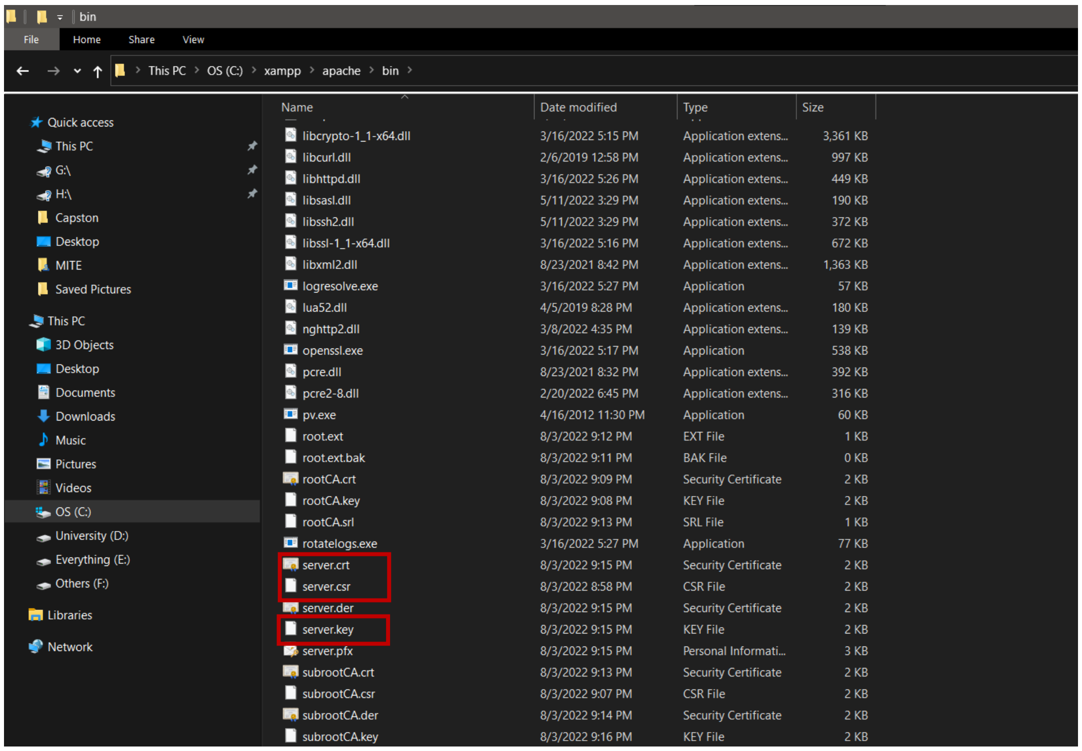
~ x509 -in server.crt -outform der -out server.der

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

Step-4: Installing Certificate:

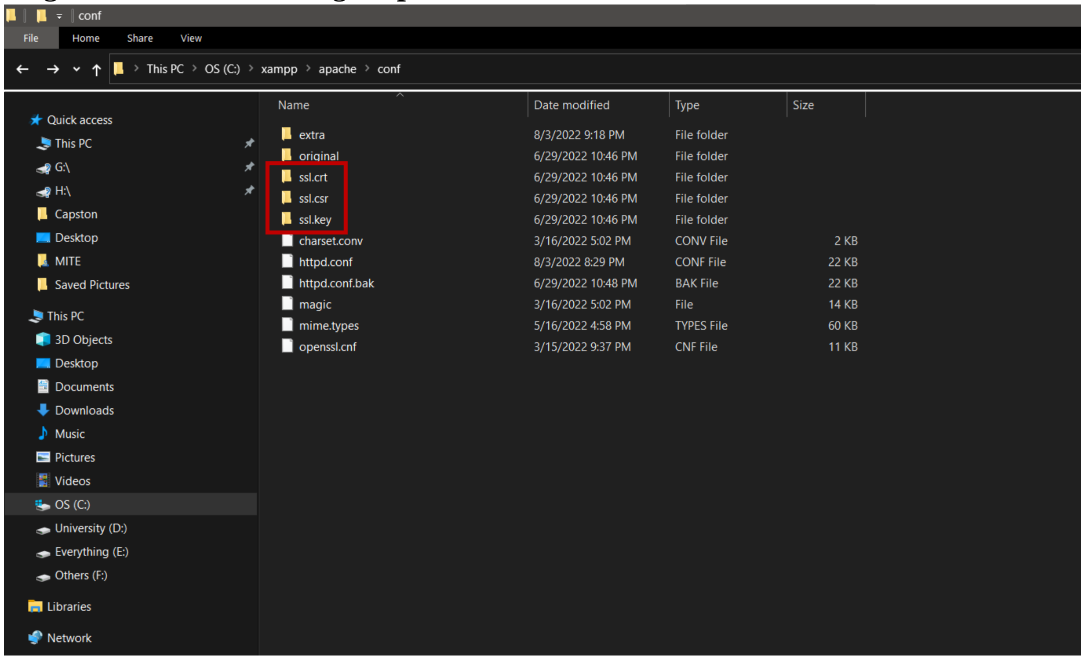
We opened **C drive → xampp → apache → bin** then we will see our certificates then we need to install it first the rootCA.cert in local **machine→trusted** root certificate **authorities→finish** then our installation will be completed in this way we need to install server.crt,subrootCA.pfx and the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



Name	Date modified	Type	Size
libcrypto-1_1-x64.dll	3/16/2022 5:15 PM	Application extens...	3,361 KB
libcurl.dll	2/6/2019 12:58 PM	Application extens...	997 KB
libhttpd.dll	3/16/2022 5:26 PM	Application extens...	449 KB
libsasl.dll	5/11/2022 3:29 PM	Application extens...	190 KB
libssh2.dll	5/11/2022 3:29 PM	Application extens...	372 KB
libssl-1_1-x64.dll	3/16/2022 5:16 PM	Application extens...	672 KB
libxml2.dll	8/23/2021 8:42 PM	Application extens...	1,363 KB
logresolve.exe	3/16/2022 5:27 PM	Application	57 KB
lua52.dll	4/5/2019 8:28 PM	Application extens...	180 KB
nghttp2.dll	3/8/2022 4:35 PM	Application extens...	139 KB
openssl.exe	3/16/2022 5:17 PM	Application	538 KB
pcre.dll	8/23/2021 8:32 PM	Application extens...	392 KB
pcre2-8.dll	2/20/2022 6:45 PM	Application extens...	316 KB
pv.exe	4/16/2012 11:30 PM	Application	60 KB
root.ext	8/3/2022 9:12 PM	EXT File	1 KB
root.ext.bak	8/3/2022 9:11 PM	BAK File	0 KB
rootCA.crt	8/3/2022 9:09 PM	Security Certificate	2 KB
rootCA.key	8/3/2022 9:08 PM	KEY File	2 KB
rootCA.srl	8/3/2022 9:13 PM	SRL File	1 KB
rotatelogs.exe	3/16/2022 5:27 PM	Application	77 KB
server.crt	8/3/2022 9:15 PM	Security Certificate	2 KB
server.csr	8/3/2022 8:58 PM	CSR File	2 KB
server.der	8/3/2022 9:15 PM	Security Certificate	2 KB
server.key	8/3/2022 9:15 PM	KEY File	2 KB
server.pfx	8/3/2022 9:15 PM	Personal Informati...	3 KB
subrootCA.crt	8/3/2022 9:13 PM	Security Certificate	2 KB
subrootCA.csr	8/3/2022 9:07 PM	CSR File	2 KB
subrootCA.der	8/3/2022 9:14 PM	Security Certificate	2 KB
subrootCA.key	8/3/2022 9:16 PM	KEY File	2 KB

Figure 4: Certificate Install

For pasting the files, we need to go: **apache → conf**



Name	Date modified	Type	Size
extra	8/3/2022 9:18 PM	File folder	
original	6/29/2022 10:46 PM	File folder	
ssl.crt	6/29/2022 10:46 PM	File folder	
ssl.csr	6/29/2022 10:46 PM	File folder	
ssl.key	6/29/2022 10:46 PM	File folder	
charset.conv	3/16/2022 5:02 PM	CONV File	2 KB
httpd.conf	8/3/2022 8:29 PM	CONF File	22 KB
httpd.conf.bak	6/29/2022 10:48 PM	BAK File	22 KB
magic	3/16/2022 5:02 PM	File	14 KB
mime.types	5/16/2022 4:58 PM	TYPES File	60 KB
openssl.cnf	3/15/2022 9:37 PM	CNF File	11 KB

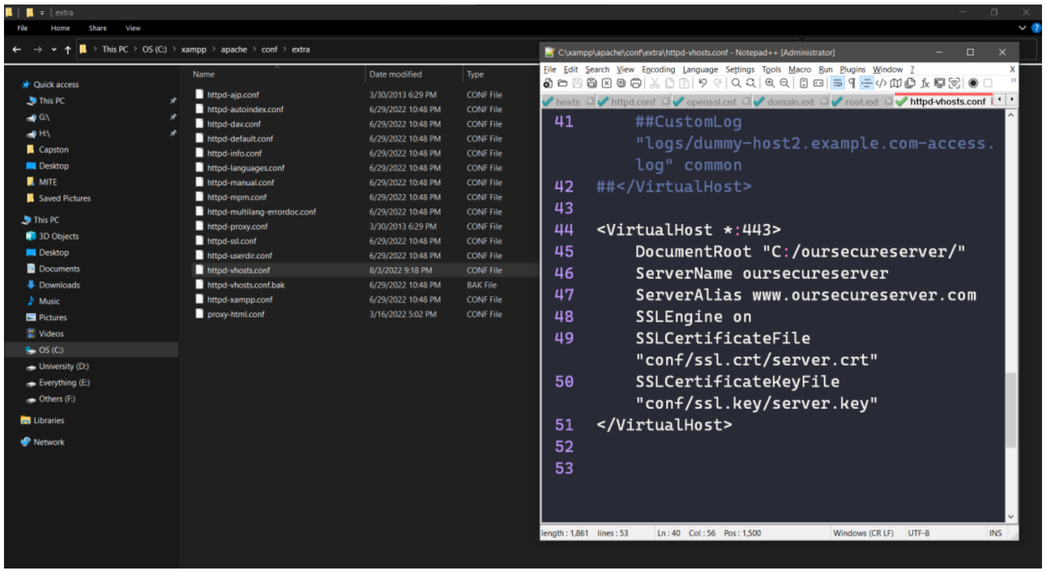
Figure 5: Pasting The Files

Step5: Creating certificate:

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

<u>Configuring httpd-vhosts</u>:

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


This PC › OS (C:) › xampp › apache › conf › extra

Name	Date modified	Type
httpd-ajp.conf	3/30/2013 6:29 PM	CONF File
httpd-autoindex.conf	6/29/2022 10:48 PM	CONF File
httpd-dav.conf	6/29/2022 10:48 PM	CONF File
httpd-default.conf	6/29/2022 10:48 PM	CONF File
httpd-info.conf	6/29/2022 10:48 PM	CONF File
httpd-languages.conf	6/29/2022 10:48 PM	CONF File
httpd-manual.conf	6/29/2022 10:48 PM	CONF File
httpd-mpm.conf	6/29/2022 10:48 PM	CONF File
httpd-multilang-errordoc.conf	6/29/2022 10:48 PM	CONF File
httpd-proxy.conf	3/30/2013 6:29 PM	CONF File
httpd-ssl.conf	6/29/2022 10:48 PM	CONF File
httpd-userdir.conf	6/29/2022 10:48 PM	CONF File
httpd-vhosts.conf	8/3/2022 9:18 PM	CONF File
httpd-vhosts.conf.bak	6/29/2022 10:48 PM	BAK File
httpd-xampp.conf	6/29/2022 10:48 PM	CONF File
proxy-html.conf	3/16/2022 5:02 PM	CONF File

C:\xampp\apache\conf\extra\httpd-vhosts.conf - Notepad++ [Administrator]

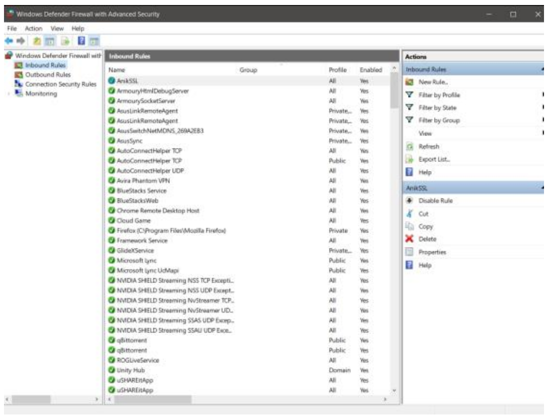
```
    ##CustomLog "logs/dummy-host2.example.com-access.log" common
##</VirtualHost>

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

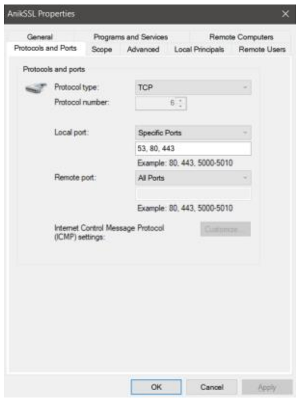
length : 1,861   lines : 53   Ln : 40   Col : 56   Pos : 1,500   Windows (CR LF)   UTF-8   INS

Figure 6: Configuring httpd-vhosts

Firewall configuration to allow necessary ports (53, 80, 443) only necessary screenshots are given:


Windows Defender Firewall with Advanced Security — Inbound Rules

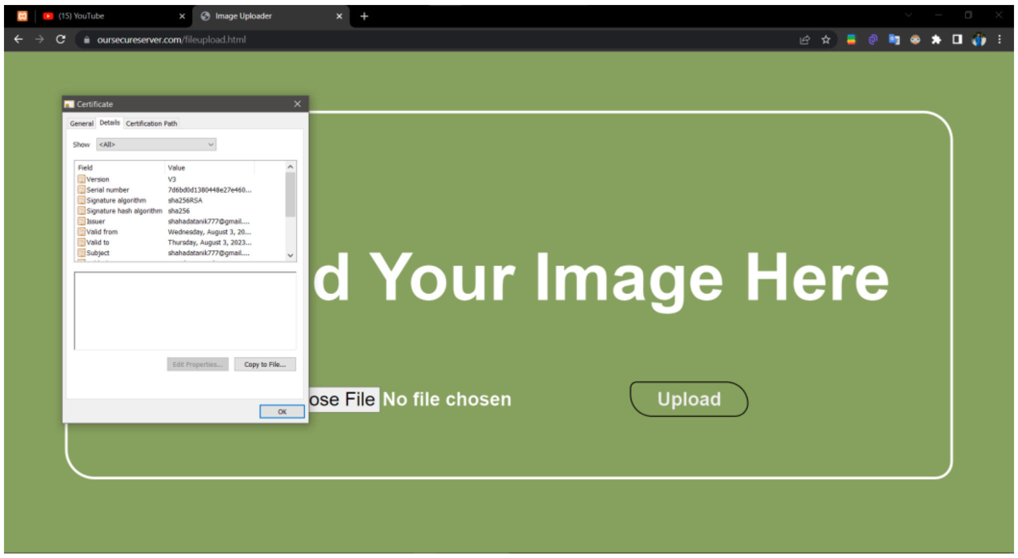
Name	Group	Profile	Enabled
AnikSSL		All	Yes
ArmouryHtmlDebugServer		All	Yes
ArmourySocketServer		All	Yes
AsusLinkRemoteAgent		Private,...	Yes
AsusLinkRemoteAgent		Private,...	Yes
AsusSwitchNetMDNS_269A2EB3		Private,...	Yes
AsusSync		Private,...	Yes
AutoConnectHelper TCP		All	Yes
AutoConnectHelper TCP		Public	Yes
AutoConnectHelper UDP		All	Yes
Avira Phantom VPN		All	Yes
BlueStacks Service		All	Yes
BlueStacksWeb		All	Yes
Chrome Remote Desktop Host		All	Yes
Cloud Game		All	Yes
Firefox (C:\Program Files\Mozilla Firefox)		Private	Yes
Framework Service		All	Yes
GlideXService		Private,...	Yes
Microsoft Lync		Public	Yes
Microsoft Lync UcMapi		Public	Yes
NVIDIA SHIELD Streaming NSS TCP Excepti...		All	Yes
NVIDIA SHIELD Streaming NSS UDP Except...		All	Yes
NVIDIA SHIELD Streaming NvStreamer TCP...		All	Yes
NVIDIA SHIELD Streaming NvStreamer UD...		All	Yes
NVIDIA SHIELD Streaming SSAS UDP Excep...		All	Yes
NVIDIA SHIELD Streaming SSAU UDP Exce...		All	Yes
qBittorrent		Public	Yes
qBittorrent		Public	Yes
ROGLiveService		All	Yes
Unity Hub		Domain	Yes
uSHAREitApp		All	Yes
uSHAREitApp		All	Yes


AnikSSL Properties — Protocols and Ports

Protocols and ports
Protocol type: TCP
Protocol number: 6
Local port: Specific Ports
53, 80, 443
Example: 80, 443, 5000-5010
Remote port: All Ports
Example: 80, 443, 5000-5010
Internet Control Message Protocol (ICMP) settings:

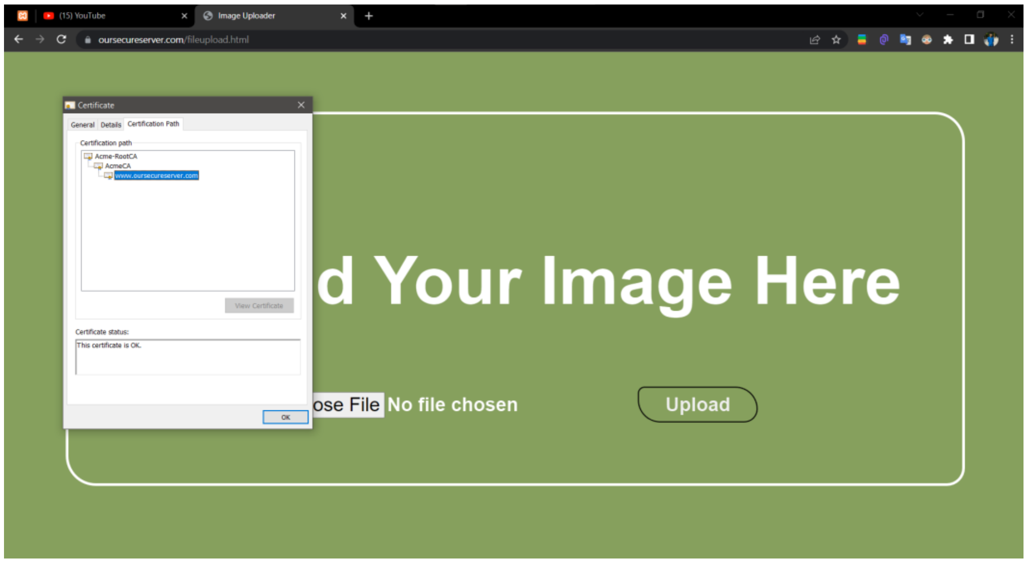
Figure 7: Firewall configuration

Finally, we will open our xampp and turn on Apache and go to our website which is running with SSL Certificate. We have also showed our certification path.



Field	Value
Version	V3
Serial number	7d6bd0d1380448e27e460...
Signature algorithm	sha256RSA
Signature hash algorithm	sha256
Issuer	shahadatanik777@gmail....
Valid from	Wednesday, August 3, 20...
Valid to	Thursday, August 3, 2023...
Subject	shahadatanik777@gmail....

Figure 8: Certificate Info



Certification path

```
Acme-RootCA
  AcmeCA
    www.oursecureserver.com
```

Certificate status:

This certificate is OK.

Figure 9: Certificate Path

Step 6: Revocation of certificate:

We must 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:
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 = 2
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.oursecureserver.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.