

**CSE489 – Cyber Security, Law, and Ethics**

[Summer 2022]

Section: 01

**Project Title:** Configuration of Certification Authority and Implementation of Transport Layer Security over HTTP

**Instructor Name:** Rashedul Amin Tuhin

Senior Lecturer, Department of Computer Science and Engineering.

**Mini Project Report - 01**

**Submitted by:**

|  |  |
| --- | --- |
| **Student ID** | **Student Name** |
| 2019-1-60-255 | Marwa khanom Nurtaj |
| 2019-1-60-208 | Aysha Siddeka |
| 2019-1-60-264 | Md. Shamsur Rahman Talukdar |

**Date of Submission: 25-08-2022**

# Preliminary Work:

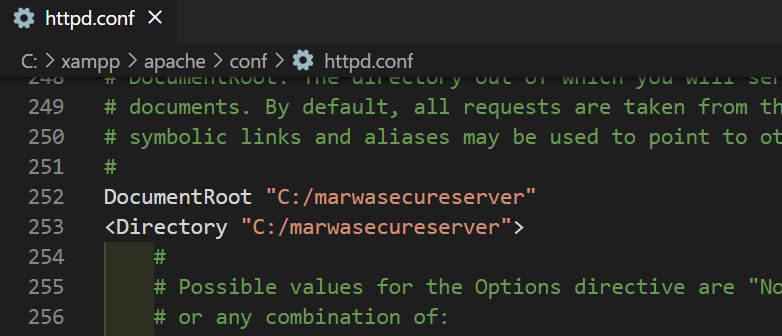
### Firstly, we have to download and install Xampp in our local machine. We can easily download and install Xampp Control Panel easily from here.

* We are creating a folder based on our website domain name ***‘marwasecureserver’*** in c drive and in the directory, we created a file named ***‘formfillup.html’*** where we have written a basic html code.

# Configure DNS File:

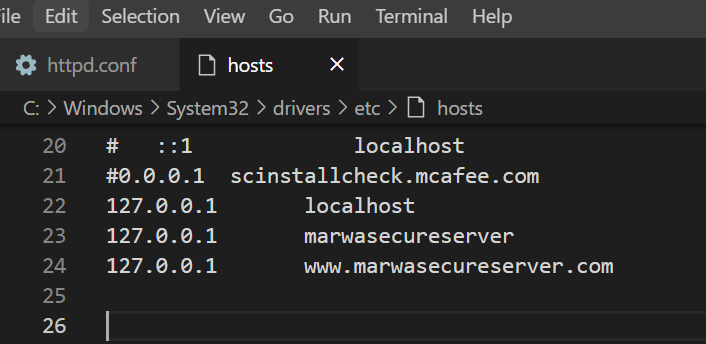
* Go to ***‘C:\xampp\apache\conf\ httpd.conf’*** file and open it in an editor then go to **252**

### number line and edit it.



***Fig: Configure httpd.conf file***

* Go to ***‘C:\Windows\System32\drivers\etc\hosts’*** file and open it in an editor and go to the last line add these two lines below.



***Fig: Configure hosts file***

# Configure OpenSSL environment path:

* Open windows command prompt with ***‘Run as Administrator’*** and run this command

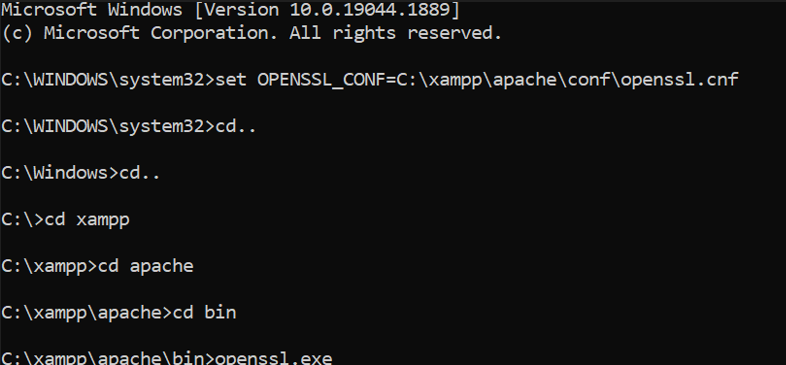
set OPENSSL\_CONF=C:\xampp\apache\conf\openssl.cnf

# Creating server certificate:

### In the previous command prompt run these following commands:

1. cd ../../

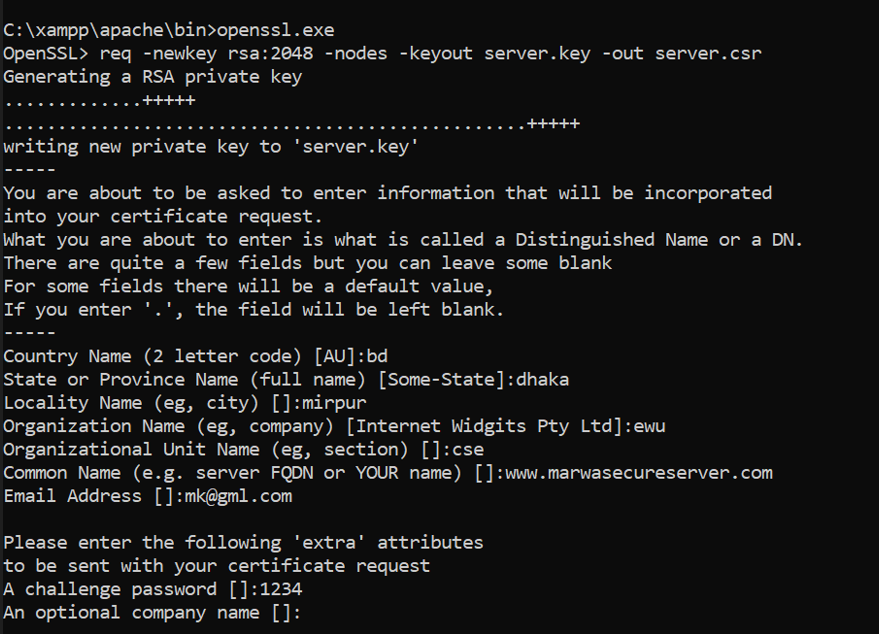
* 1. cd xampp/apache/bin
  2. openssl.exe
  3. You will see an interface like this.



* 1. req -newkey rsa:2048 -nodes -keyout server.key -out server.csr
  2. Then We have to provide Country Name, State Name, Locality Name, Organization Name, Unit Name, Common Name ([www.](http://www/) marwasecureserver.com), email address.
  3. For checking:

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

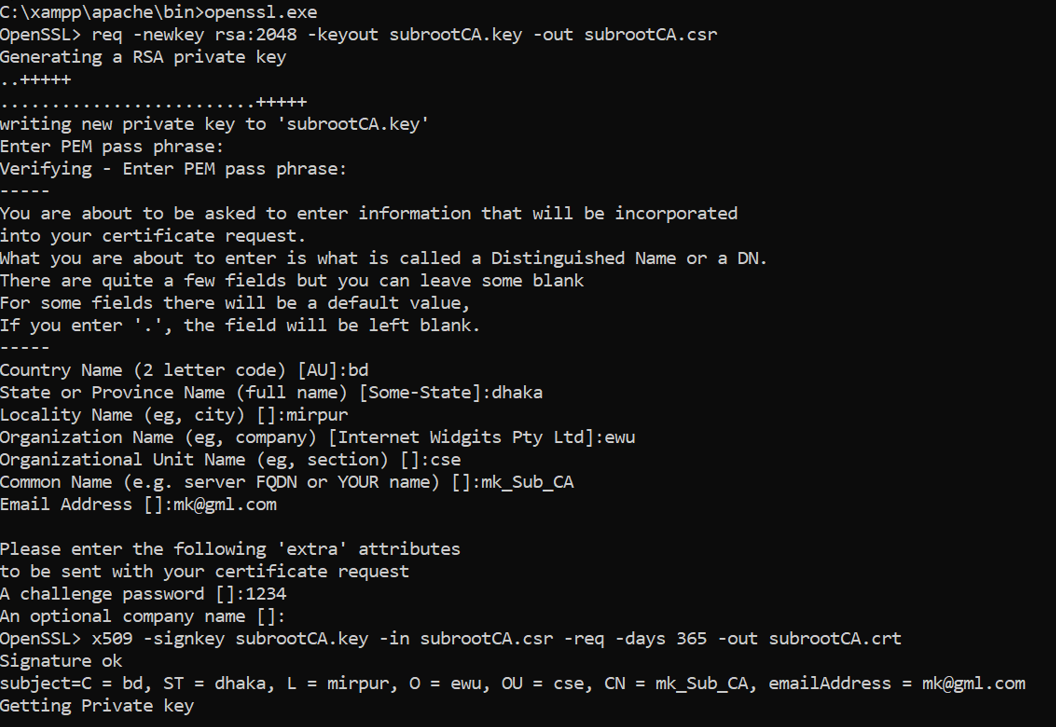
* 1. We will see an interface like this:

 Text

Description automatically generated

# Creating sub root CA certificate:

1. Ctrl + C and then type openssl.exe again.
2. req -newkey rsa:2048 -keyout subrootCA.key -out subrootCA.csr
3. Then We have to provide Country Name, State Name, Locality Name, Organization Name, Unit Name, Common Name (mk\_Sub\_CA), email address.
4. For checking:
5. x509 -signkey subrootCA.key -in subrootCA.csr -req -days 365 - out subrootCA.crt
6. We will see an interface like this:



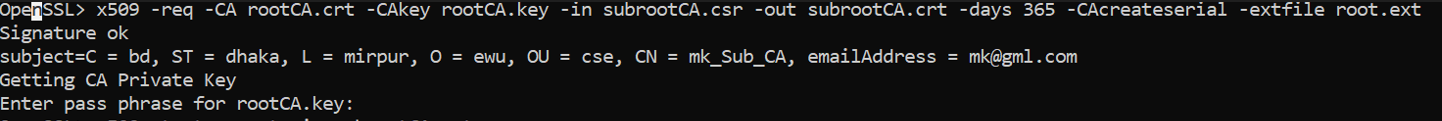
# Creating root CA certificate:

1. Ctrl + C and then type openssl.exe again.
2. req -x509 -sha256 -days 1825 -newkey rsa:2048 -keyout rootCA.key

-out rootCA.crt

1. Then We have to provide Country Name, State Name, Locality Name, Organization Name, Unit Name, Common Name (MK\_ROOT\_CA), email address.
2. We will see an interface like this:

Text

Description automatically generated 

# Configure those certificates:

* Go to ***‘C:\xampp\apache\bin’*** and create a file name ***‘domain.ext’*** and paste the following code:

authorityKeyIdentifier=keyid,issuer

basicConstraints=CA:FALSE

subjectAltName = @alt\_names

[alt\_names]

DNS.1 =www.marwasecureserver.com

DNS.2 =127.0.0.1

* Go to ***‘C:\xampp\apache\bin’*** and create a file name ***‘root.ext’*** and paste the following code:

authorityKeyIdentifier=keyid,issuer

basicConstraints=CA:TRUE

subjectAltName = @alt\_names

[alt\_names]

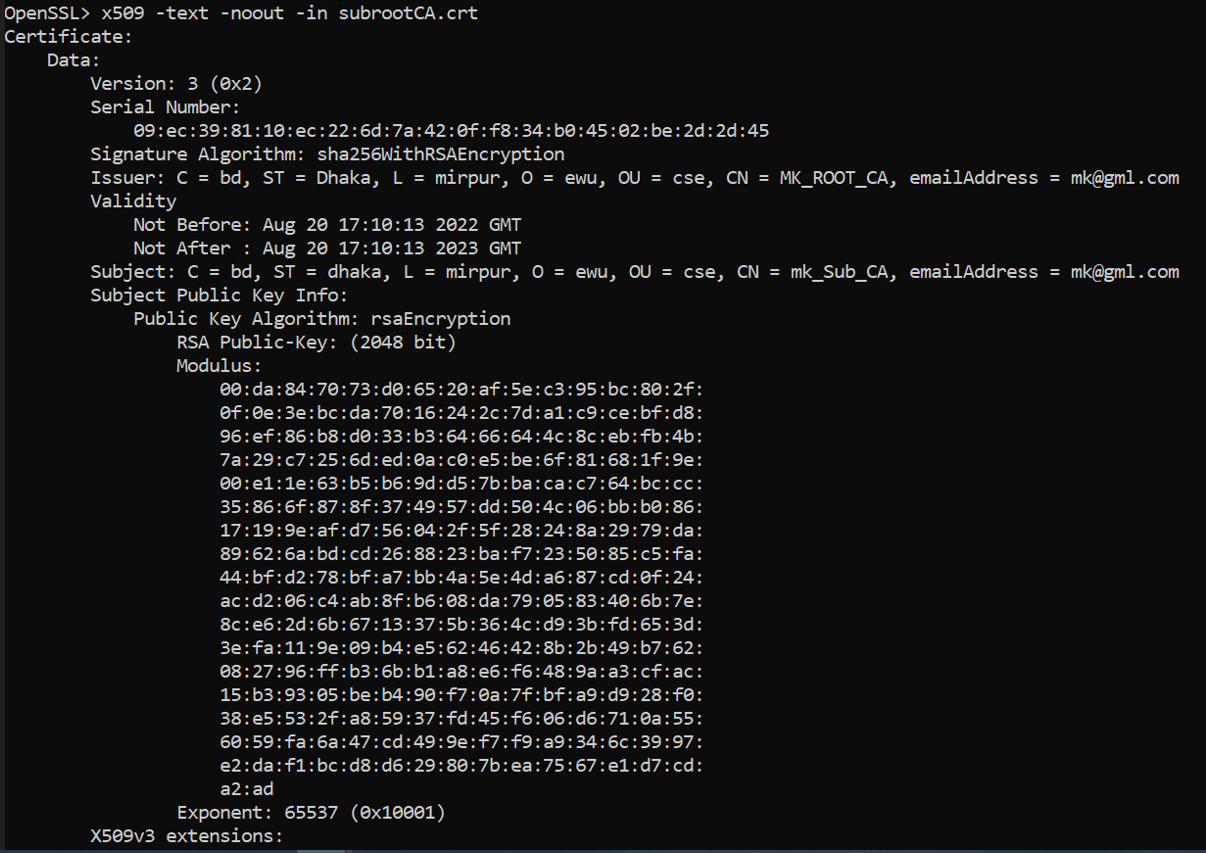
DNS.1 =www.marwasecureserver.com

DNS.2 =127.0.0.1

# Signing sub root CA certificate with root CA certificate:

* x509 -req -CA rootCA.crt -CAkey rootCA.key -in subrootCA.csr -out subrootCA.crt -days 365 -CAcreateserial -extfile root.ext
* For Checking:

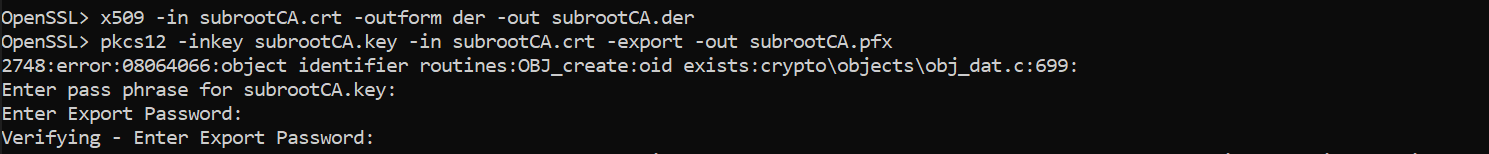
x509 -text -noout -in subrootCA.crt

 Text

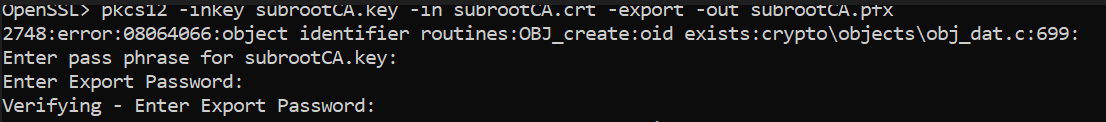
Description automatically generated

# Exporting the sub root CA key file in sub root CA pfx file:

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

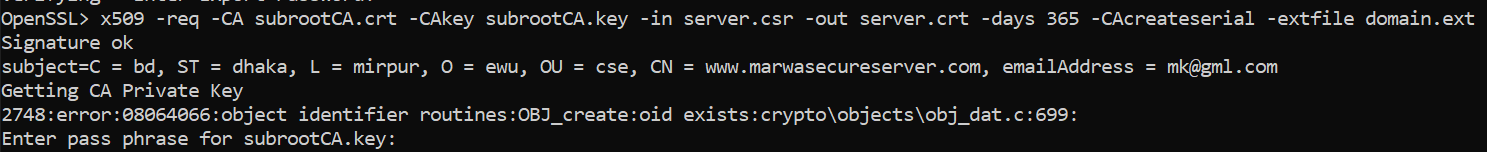


* pkcs12 -inkey subrootCA.key -in subrootCA.crt -export -out subrootCA.pfx

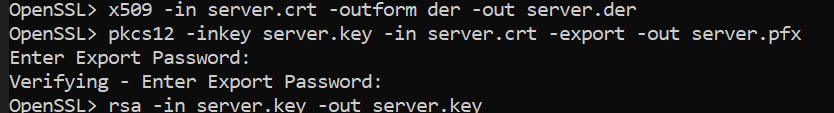


# Signing server certificate with sub root CA certificate:

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

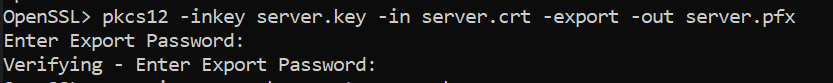


* x509 -in server.crt -outform der -out server.der



# Exporting the server key file in the server .pfx file:

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

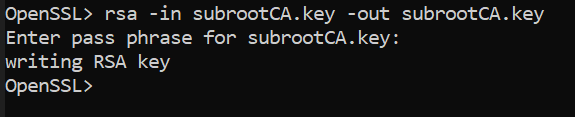


# Replacing the RSA encryption from the server and sub root CA key for setting the validity:

* rsa -in server.key -out server.key



* rsa -in subrootCA.key -out subrootCA.key



Now we have to install those certificates. Go to ***‘C:\xampp\apache\bin’*** and install ***rootCA.crt***

and ***subrootCA.pfx***.

* Copy from that location ***‘server.crt’*** and replace with

## ‘C:\xampp\apache\conf\ssl.crt\server.crt’

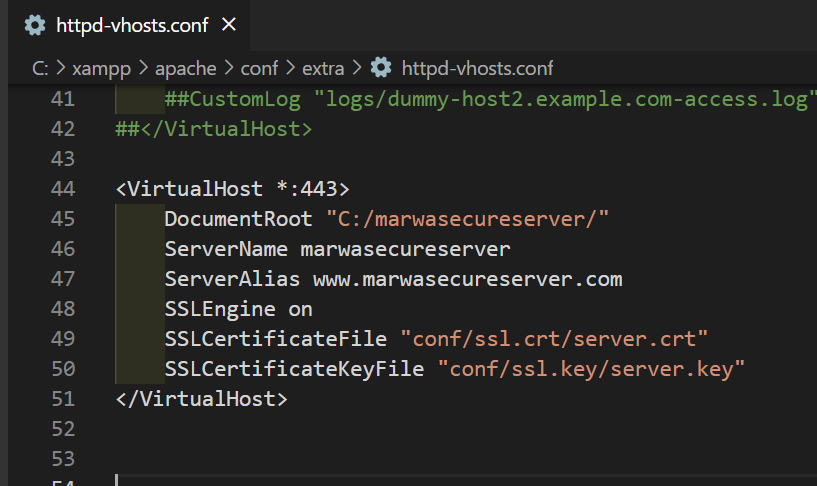
* Copy from that location ***‘server.csr’*** and replace with

## ‘C:\xampp\apache\conf\ssl.csr\server.csr’

* Copy from that location ***‘server.key’*** and replace with

## ‘C:\xampp\apache\conf\ssl.key\server.key’

Lastly, go to ***‘C:\xampp\apache\conf\extra\httpd-vhosts.conf’*** and open in an editor and add these lines of code at the last for configuring ***httpd-vhosts***:



# Finally we can have our certified server like below:

# Analysis of secure and non-secure server using Wireshark:

# 

# Configure DNS with Bind 9:

# 

# DOS Attack by using kali Linux:

# 

# Graphical user interface, application Description automatically generated

# Intrusion Detection using Snort:

# 

# Firewall configuration to allow necessary ports:

# Graphical user interface, text, application, Teams Description automatically generated

# Revocation of certificate:

* Go to ***‘C:\xampp\apache\bin’*** location and create file named ***‘subrootCA.conf’*** where this code will be written:

[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 = ewu

[ 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.marwasecureserver.com

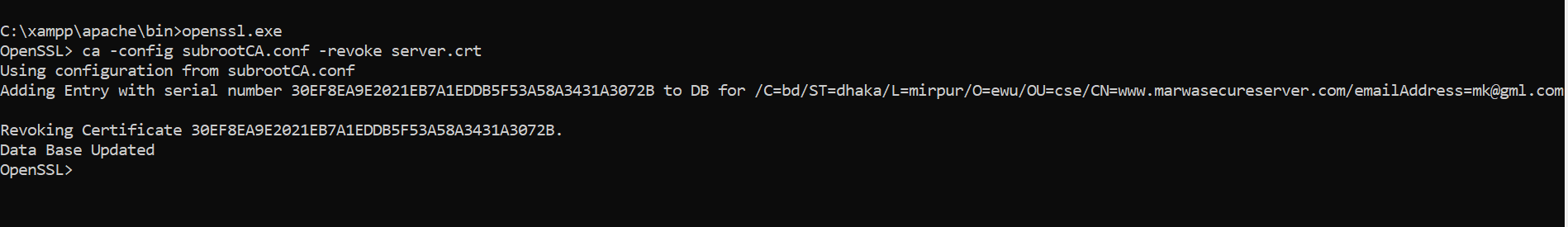
DNS.2 = 127.0.0.1

* Now we have to create some files in the same directory named ***index.txt***, ***serial.txt*** and

## crlnumber.txt

### Open openssl.exe to revoke the certificate issued to verysecureserver.com from the Acme CA

ca -config subrootCA.conf -revoke server.crt



Then we need to copy the revoking certification number and paste it into crlnumber.txt fileGraphical user interface, text, application

Description automatically generated

### To generate revocation crl file

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

Text

Description automatically generated

### To see the revocation file in the form of text

crl -in rev.crl -noout -text

Text

Description automatically generated