PROJECT 3: Public-Key Infrastructure (PKI) Lab

Objective: The objective of this project is to gain first-hand experience on PKI, how its used to protect the Web, and how Man-in-the-middle attacks can be defeated by PKI.

Task 1: Becoming a Certificate Authority (CA):

To get started with tasks, we did some configuration setup, as shown in below snapshots:

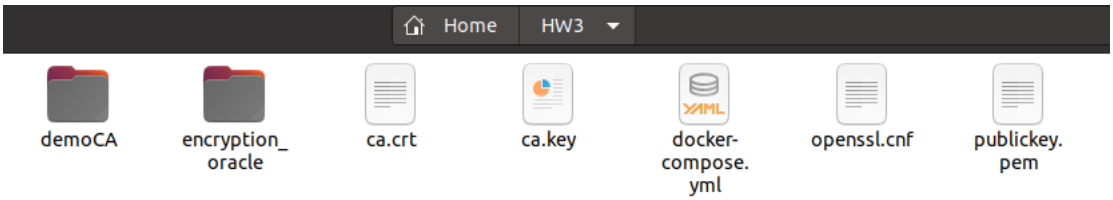
```
[04/08/23]seed@VM:~/HW3$ cp /usr/lib/ssl/openssl.cnf .
[04/08/23]seed@VM:~/HW3$ ll
total 20
-rwxrwxrwx 1 seed seed   425 Feb 16 18:22 docker-compose.yml
drwxrwxrwx 2 seed seed  4096 Feb 16 18:29 encryption_oracle
-rw-r--r-- 1 seed seed 10909 Apr  8 12:32 openssl.cnf
[04/08/23]seed@VM:~/HW3$
```

```
seed@VM: ~/HW3/demoCA
[04/08/23]seed@VM:~/HW3$ mkdir demoCA
[04/08/23]seed@VM:~/HW3$ cd demoCA
[04/08/23]seed@VM:~/HW3/demoCA$ mkdir certs crl newcerts
[04/08/23]seed@VM:~/HW3/demoCA$ touch index.txt serial
[04/08/23]seed@VM:~/HW3/demoCA$
```

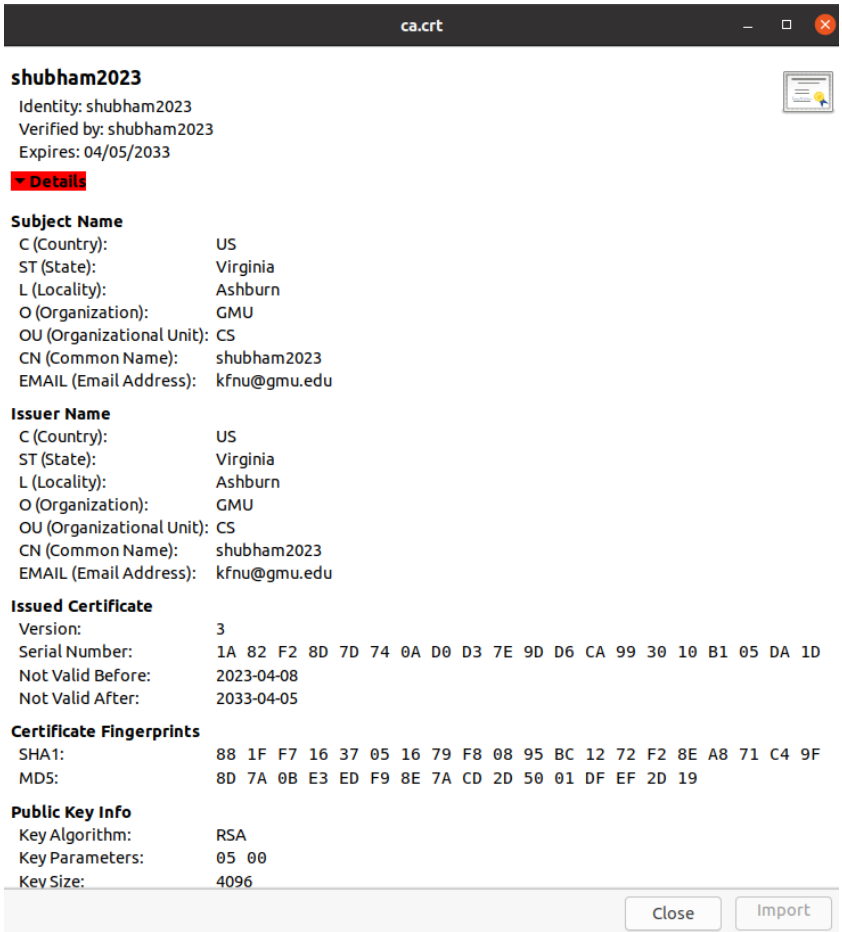
```
seed@VM: ~/HW3/demoCA
 GNU nano 4.8                    serial
1000
```

After configuring all the directories and sub-directories, we need to generate a self-signed certificate for our Certificate Authority (CA). So, we ran below command to create ca.crt (contains public key certificate) and ca.key (contains CA's private key).

```
seed@VM: ~/HW3
[04/08/23]seed@VM:~/HW3$ openssl req -x509 -newkey rsa:4096 -sha256 -days 3650 -keyout ca.key -out ca.crt
Generating a 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]:Virginia
Locality Name (eg, city) []:Ashburn
Organization Name (eg, company) [Internet Widgits Pty Ltd]:GMU
Organizational Unit Name (eg, section) []:CS
Common Name (e.g. server FQDN or YOUR name) []:shubham2023
Email Address []:kfnu@gmu.edu
[04/08/23]seed@VM:~/HW3$
```



The file ca.crt (public key certificate) looks like below:



Q.1: What part of the certificate indicates this is a CA's certificate?

==> First, I ran this command "openssl x509 -in ca.crt -text -noout". This command will display the certificate's details in a human-readable format. Now, check If the Basic Constraints extension has **"CA:TRUE"**, as shown below, it indicates that the certificate belongs to a Certificate Authority.

```
X509v3 Basic Constraints: critical
    CA:TRUE
```

Q.2: What part of the certificate indicates this is a self-signed certificate?

==> If the issuer and subject fields are identical, as shown below, the certificate is self-signed.

```
[04/08/23]seed@VM:~/HW3$ openssl x509 -in ca.crt -noout -issuer -subject
issuer=C = US, ST = Virginia, L = Ashburn, O = GMU, OU = CS, CN = shubham2023, emailAddress = kfnu@gmu.edu
subject=C = US, ST = Virginia, L = Ashburn, O = GMU, OU = CS, CN = shubham2023, emailAddress = kfnu@gmu.edu
[04/08/23]seed@VM:~/HW3$
```

Q.3: In the RSA algorithm, we have a public exponent e, a private exponent d, a modulus n, and two secret numbers p and q, such that n = pq. Please identify the values for these elements in your certificate and key files.

==> Using this command, "$ openssl rsa -in ca.key -text -noout", I was able to see the values of public exponent e, a private exponent d, a modulus n, and two secret numbers p and q, as shown in the below snapshots.

```
[04/08/23]seed@VM:~/HW3$ openssl rsa -in ca.key -text -noout
Enter pass phrase for ca.key:
RSA Private-Key: (4096 bit, 2 primes)
modulus:
    00:cd:2d:ff:dc:d2:23:bc:92:8b:b9:93:fd:7d:3c:
    1b:c0:2c:fe:ac:e4:5b:b5:59:20:93:7b:5f:2b:dc:
    7c:12:1c:33:c4:36:8c:ef:34:2c:88:5c:3c:a7:b6:
    4a:a7:85:af:2b:3d:04:3a:dd:c3:09:1f:0f:40:49:
    14:e9:34:e5:62:dd:46:02:58:d5:3a:9e:18:4d:0f:
    0a:75:58:30:8c:ac:8e:7c:fe:e2:c5:83:55:7c:3b:
    99:1d:b9:07:ff:ed:19:7c:41:25:57:57:9b:7f:c0:
    c7:26:98:77:f5:85:cd:fb:f8:d1:de:05:72:88:8d:
    a0:6b:d4:7b:f1:bb:f1:0a:7d:3a:18:97:5e:6e:61:
    a1:ff:dc:ae:74:d9:0d:b9:be:06:53:f2:74:f4:ff:
    21:9e:c7:74:77:4e:b8:f0:5d:67:8d:89:5b:ba:f6:
    98:07:28:ee:2a:0c:86:1c:5a:df:01:2a:e9:af:d1:
    d1:ce:fc:9d:80:20:47:b7:24:25:eb:5b:bb:05:cd:
    86:26:b8:cc:65:d0:a9:b0:94:77:e5:a2:4a:50:f2:
    e0:96:02:d2:f8:44:f8:e8:54:7a:dd:6e:ab:74:f8:
    56:c9:43:da:6d:48:26:a7:20:90:16:75:ba:61:c6:
    b8:8e:3c:4f:43:13:5a:1e:ac:e6:1d:f4:c7:56:b3:
    06:8f:7f:ed:e6:42:6f:54:fe:fb:6e:04:d8:af:d0:
    69:2f:40:6c:d3:ec:d6:48:fb:5a:e4:00:b4:28:20:
    67:c8:63:3b:d9:74:bc:14:b2:f1:cc:01:0a:34:39:
    dd:64:97:38:11:a2:3c:0b:52:1e:2b:15:25:86:f9:
    2d:86:e4:ae:7d:45:74:fe:fe:ff:e1:00:0d:1a:c7:
    fa:03:d7:4c:ef:9a:8c:76:f2:49:d5:a5:f1:18:52:
    5d:67:9d:84:1a:4e:8e:87:d9:dc:1e:f2:85:81:e4:
    24:a2:a8:90:bb:b2:ab:2d:6e:19:0c:f8:09:f8:f6:
    97:0c:95:70:cf:60:85:d9:27:6b:56:e0:45:d9:da:
    9a:79:bb:9b:d7:b2:60:f6:84:c6:d5:7c:91:3c:82:
    57:32:18:68:c7:39:1a:7f:56:1b:3e:96:53:2b:cf:
    f6:5f:68:eb:9f:d3:fa:01:10:cf:34:d2:cc:f0:2c:
    a7:f9:b8:0f:9e:31:f5:ee:4b:ad:aa:f8:dd:1a:87:
    08:39:9e:a2:57:97:e8:27:56:53:16:90:72:42:b3:
    60:d3:41:d9:33:26:fc:42:02:3c:7b:8f:70:03:b5:
    b6:22:e7:7a:86:39:8f:29:27:8d:82:cf:6e:fd:e7:
    8e:16:a1:ad:02:10:20:26:8c:c4:fb:dd:30:20:9e:
    fb:4d:85
publicExponent: 65537 (0x10001)
privateExponent:
    74:1a:8c:87:8b:4d:f7:9e:41:7c:c0:f6:97:50:55:
    2f:b9:06:60:15:54:a3:d9:0c:6b:08:4c:01:88:e1:
    98:69:e7:0d:28:05:36:32:a2:e4:82:b5:3b:fe:16:
    5e:97:72:59:18:4c:f5:76:99:af:e6:a8:7d:ab:1d:
    2f:1d:e4:93:be:3c:a8:85:56:1b:b6:6c:6f:e6:8b:
    f7:7c:f0:f4:19:8d:03:c4:43:d4:9a:8d:dc:1f:e9:
    73:fd:49:3e:94:0f:70:d5:78:68:e5:45:33:d4:85:
    9d:1c:77:fd:32:3b:2e:53:4d:86:c2:34:5e:7b:2e:
    04:f6:97:30:62:36:72:c6:0c:02:f0:5c:da:3f:5b:
    29:6a:da:39:7a:bd:9e:96:9e:04:10:d6:07:f0:fb:
    38:5b:0b:57:9b:ef:60:ec:a9:f5:58:7d:d4:dd:47:
    87:fd:ac:4f:95:08:aa:b0:ed:32:3e:f3:3f:af:8e:
    76:ff:51:ac:b8:17:7c:d5:f2:b3:2d:14:36:a3:67:
    3f:30:21:51:97:f2:fd:dd:f5:be:b6:c8:30:e8:4f:
    17:f3:ff:27:c7:f7:c5:98:50:72:4e:f6:ac:d8:fe:
    5b:23:8f:4d:79:9f:8d:c1:0a:67:fb:34:7d:b1:b2:
    ab:a2:0b:04:1c:c6:49:47:73:6c:0e:20:62:ba:29:
    82:33:be:7f:94:1e:34:eb:ab:d0:02:ef:88:b4:cb:
    aa:06:26:20:6f:cd:6b:d0:11:d7:b3:65:ed:df:0c:
```

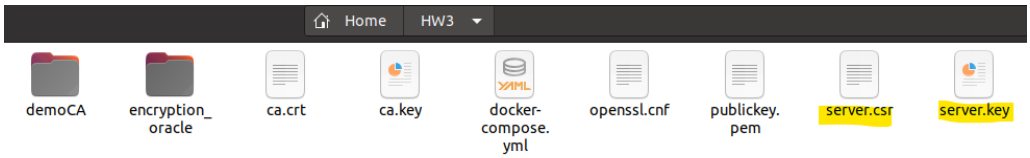
```
prime1:
    00:fc:67:9d:f3:2e:43:53:eb:89:db:8d:f2:8e:80:
    24:20:cf:e3:6d:76:df:c9:af:22:70:5b:22:03:ac:
    d0:f2:d9:bd:7d:73:4d:79:7b:f9:75:bc:b6:f3:5f:
    7d:1b:1c:a2:3b:80:96:c9:65:89:a3:0f:c8:69:ee:
    7b:33:a7:1a:42:10:c4:62:d5:83:ec:25:7b:a5:a0:
    a2:3f:eb:f5:ae:f4:f5:cf:b5:25:93:97:c9:85:df:
    ee:36:43:18:b9:68:e4:63:48:33:0b:7f:7b:1a:f3:
    8a:8b:f5:9c:c2:c1:d3:00:e2:34:93:b2:f6:b1:ef:
    6d:90:ef:1a:2c:59:81:97:0e:87:1b:81:97:ce:97:
    37:7f:65:e4:54:3f:c0:59:db:33:fd:c1:02:fd:a8:
    b3:9e:e1:3d:24:c4:93:b9:3d:9d:55:58:03:27:cf:
    57:8e:4e:22:40:d1:3a:16:3c:1c:7f:ae:65:07:e2:
    c9:58:e5:d6:00:12:29:ea:4d:ce:7d:fc:fb:8e:f5:
    2f:70:92:db:41:bb:e2:b6:5e:89:59:46:79:54:c0:
    62:ae:d3:32:46:7a:32:b4:67:b5:12:4d:59:f8:d8:
    9f:45:e6:11:8d:db:da:1f:0e:14:61:15:2a:5a:51:
    cf:4c:92:54:35:4b:89:f6:96:9c:cf:52:ef:09:dc:
    fc:69
prime2:
    00:d0:1a:2d:a5:4d:64:05:c6:eb:d3:4f:81:3f:33:
    a7:29:5b:ef:a9:ab:32:03:b4:b0:b4:75:d0:8e:f6:
    47:28:26:ec:d4:b4:b1:16:44:7b:cf:d8:15:3f:96:
    d8:d9:5c:96:5e:63:48:a7:c7:a8:34:17:74:b0:51:
    cd:60:d1:ff:c6:1d:7d:47:96:ae:2d:f0:dd:66:dd:
    e1:d6:1c:ea:87:ca:90:1c:bf:a9:cb:c4:2b:3b:72:
    20:f6:95:29:72:ab:de:6b:87:ae:2b:23:36:9d:07:
    46:93:14:16:a3:5a:e3:04:b9:d6:90:80:a0:a9:e5:
    11:82:f4:2e:a2:c9:83:b6:ce:db:a2:ec:94:2d:f4:
    71:91:67:2c:33:f4:3d:ff:09:3e:b5:a8:2e:93:93:
    e0:e5:66:34:f6:6f:37:74:bf:5a:77:6c:4e:b1:c2:
    a2:04:87:47:4f:b4:9f:b5:1f:e7:6a:4c:91:07:51:

coefficient:
    00:e7:44:39:d5:60:58:11:5b:52:f1:e3:00:94:de:
    4e:e4:49:f5:d2:75:61:2e:b1:5a:4b:2a:5b:9e:56:
    87:95:f8:91:89:72:63:b1:83:e7:7e:6b:99:53:a4:
    27:77:a5:e6:bf:fc:eb:b3:f3:11:3f:bc:c0:1e:62:
    a5:2d:a0:3e:bb:4c:70:00:0f:e2:da:6d:9f:e9:5e:
    c0:21:0b:e7:c4:69:8e:f8:c2:6f:b7:d4:41:4c:00:
    de:3f:fe:c8:4f:b7:b0:26:f2:eb:06:de:0a:d2:1e:
    ae:b7:e5:c4:35:7b:f0:84:f1:75:30:d9:d4:2e:05:
    c5:8e:c7:94:50:94:17:79:93:ad:70:f1:60:6d:06:
    27:e7:89:dc:54:dc:5e:69:f3:c9:3c:09:40:18:6f:
    40:87:4f:79:88:9a:03:59:6c:bd:58:5d:f1:b7:02:
    24:78:fd:dd:9d:9e:bc:72:51:d5:5a:e3:cf:74:5a:
    cc:2d:4d:e5:b3:29:aa:f0:b4:25:81:b6:d5:14:3f:
    92:8d:10:02:a7:0e:df:b5:91:9a:24:00:1a:aa:20:
    c3:4f:2c:3f:0f:23:60:ac:b8:82:9b:a2:f2:ba:06:
    76:cc:a8:c6:97:ab:b0:be:71:3c:ee:9f:19:41:f9:
    fc:2a:ef:ef:41:9f:30:50:74:84:65:a1:e6:ff:29:
    e2:07
[04/08/23]seed@VM:~/HW3$
```

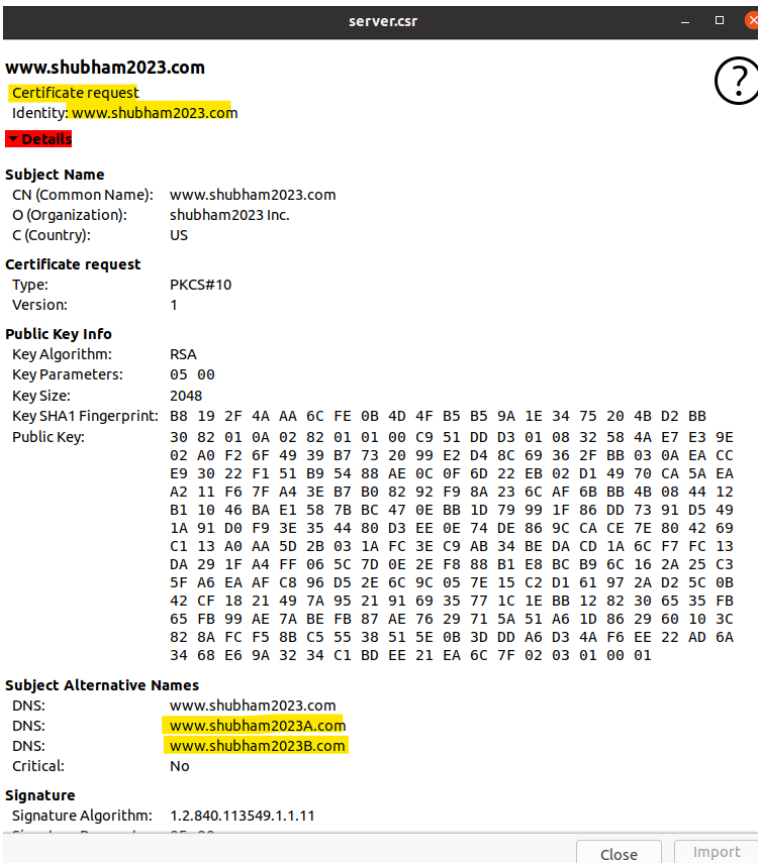
```
exponent1:
    50:3b:1f:9a:0d:3a:99:92:65:5f:c4:df:35:2c:c6:
    4f:27:c5:c3:25:e6:3c:d5:ce:bc:a8:3b:47:af:c7:
    3b:6a:bb:31:05:55:bb:28:0d:43:bf:98:e8:03:92:
    60:eb:d9:25:d0:da:1d:6a:89:35:ab:ee:a8:bd:85:
    ac:05:91:f3:2e:21:3e:c9:60:05:d0:64:5d:61:92:
    31:7f:6f:b9:0d:64:95:81:9f:50:2b:0f:83:de:8a:
    79:f5:1a:06:d5:b0:c3:5e:4f:db:77:ca:49:e6:e0:
    43:5c:ef:57:79:04:f9:62:ee:13:84:ef:a5:df:d5:
    94:1c:6c:d5:6c:1e:25:46:18:ec:45:65:5b:e5:f5:
    5e:29:a5:4a:a2:79:22:16:eb:8f:dd:aa:75:b7:0a:
    61:96:77:39:18:c3:5c:3e:99:a4:67:35:99:91:ee:
    e2:ed:33:36:27:4b:af:85:ac:09:6a:05:f3:5b:c3:
    64:de:aa:07:9b:be:77:1c:04:67:47:e3:2e:ae:01:
    46:67:68:0a:9d:0e:93:a9:b9:3f:07:87:c9:c8:dd:
    95:74:cc:a6:5a:71:83:a6:f3:87:bd:ce:bb:ab:1f:
    f5:b8:0c:1c:6b:a6:09:f9:0f:c1:4c:59:0a:1c:b7:
    28:3f:db:44:95:9e:65:23:ba:f4:ac:08:0f:ba:8f:
    01
exponent2:
    00:c8:13:33:29:87:83:8f:ee:55:53:1f:50:53:14:
    75:35:d5:77:51:78:0f:a8:f2:73:25:bd:53:db:ef:
    3a:30:87:64:11:88:fb:ed:3d:c0:8f:ee:df:56:f8:
    cf:da:03:23:6a:f2:27:ab:6f:d0:ee:ec:52:3b:27:
    1a:bf:68:87:71:d1:6e:5e:82:a7:49:f2:16:db:4c:
    9e:51:3e:11:da:a6:4c:f1:e6:ef:f7:c1:ce:12:e4:
    42:7e:f0:a0:1d:e8:d1:fa:2e:43:ac:a3:b9:61:35:
    da:8b:93:b7:c4:2c:a0:b1:2d:7e:f6:05:44:f2:b3:
    12:a8:05:de:44:92:5b:f9:49:5e:6f:80:e7:62:5b:
    6f:48:ed:99:d9:05:8d:09:83:b8:ef:ec:17:83:b9:
```

Task 2: Generating a Certificate Request for your Web Server:

In this step, we will create a Certificate Signing Request (CSR) for **shubham2023.com,** as shown below:

```
[04/08/23]seed@VM:~/HW3$ openssl req -newkey rsa:2048 -sha256 \
> -keyout server.key -out server.csr \
> -subj "/CN=www.shubham2023.com/O=shubham2023 Inc./C=US" \
> -passout pass:dees
Generating a RSA private key
.....................................+++++
.......+++++
writing new private key to 'server.key'
-----
[04/08/23]seed@VM:~/HW3$
```





Task 3: Generating a Certificate for your Server:

To form a certificate, the CSR file must have the CA's signature. Below command uses ca.crt and ca.key with some additional highlighted parameter to avoid any ambiguity, to sign the **CSR** (server.csr) into a **certificate** (X509) for **shubham2023.com**, as shown in below snapshot.

```
[04/08/23]seed@VM:~/HW3$ openssl ca -config myCA_openssl.cnf -policy policy_anything -md sha256
tch -cert ca.crt -keyfile ca.key
Using configuration from myCA_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 00:47:54 2023 GMT
            Not After : Apr  6 00:47:54 2033 GMT
        Subject:
            countryName               = US
            organizationName          = shubham2023 Inc.
            commonName                = www.shubham2023.com
        X509v3 extensions:
            X509v3 Basic Constraints:
                CA:FALSE
            Netscape Comment:
                OpenSSL Generated Certificate
            X509v3 Subject Key Identifier:
                DE:B0:97:45:29:98:F7:2F:74:CA:47:1D:1E:E8:18:08:F3:53:D2:8A
            X509v3 Authority Key Identifier:
                keyid:B2:B5:CA:95:72:D5:5E:18:37:6F:3B:EA:72:66:97:F9:98:4E:CD:2A

Certificate is to be certified until Apr  6 00:47:54 2033 GMT (3650 days)

Write out database with 1 new entries
Data Base Updated
[04/08/23]seed@VM:~/HW3$
```

Below is the decoded content of the certificate, and the alternative names are included.

```
[04/10/23]seed@VM:~/HW3$ openssl x509 -in server.crt -text -noout
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number: 4096 (0x1000)
        Signature Algorithm: sha256WithRSAEncryption
        Issuer: C = US, ST = Virginia, L = Ashburn, O = GMU, OU = CS, CN
        Validity
            Not Before: Apr 10 01:59:57 2023 GMT
            Not After : Apr  7 01:59:57 2033 GMT
        Subject: C = US, O = shubham2023 Inc., CN = www.shubham2023.com
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
                RSA Public-Key: (2048 bit)
                Modulus:
                    00:c6:dd:2a:ff:9c:0b:c8:8f:5b:d6:38:89:df:37:
                    88:09:72:8e:13:ef:bc:5d:b6:ce:36:dd:3c:c8:52:
                    96:4f:ce:31:47:d2:11:90:ac:1e:f2:dd:c7:fe:fe:
                    bd:66:99:55:48:30:83:64:07:f4:4f:8d:2a:64:77:
                    93:88:bb:21:cc:85:bf:0c:be:8a:54:5f:b9:1f:b5:
                    b7:3e:26:dd:4c:40:b4:c1:13:05:98:ce:8c:eb:95:
                    6b:01:b5:58:c7:08:1f:a4:30:9d:72:b6:6b:f6:3b:
                    3b:34:89:f1:12:55:0d:32:22:c7:3b:a1:5b:a4:25:
                    89:5b:df:af:30:64:8b:1a:70:08:84:60:78:a7:37:
                    df:61:95:ff:d1:c1:07:1b:ab:a2:28:8d:18:f9:5b:
                    a7:6c:b4:bb:88:61:53:57:3b:eb:cf:ac:61:4e:81:
                    42:f4:3f:ee:fa:e6:e0:3d:9c:4a:7b:bc:51:ff:80:
                    22:d3:e8:d5:48:98:46:6a:1c:66:05:a1:e6:38:fb:
                    f9:0c:82:52:87:35:5a:97:ae:01:5e:32:d7:b3:a4:
                    c9:1b:e9:c3:2e:37:17:9f:8a:e1:22:b2:bc:56:1c:
                    40:b0:9e:e3:14:6e:0d:b9:fc:13:0a:23:a8:fd:40:
                    50:3f:0a:0f:da:ae:48:e7:51:83:93:97:30:df:ff:
                    b9:a9
                Exponent: 65537 (0x10001)
        X509v3 extensions:
            X509v3 Basic Constraints:
                CA:FALSE
            Netscape Comment:
                OpenSSL Generated Certificate
            X509v3 Subject Key Identifier:
                F8:0B:8F:00:9A:CE:B5:53:BF:51:42:31:DA:84:4F:80:7B:5A:9A:03
            X509v3 Authority Key Identifier:
                keyid:EE:BC:AD:97:21:3A:E4:CE:91:16:F1:06:C3:FE:C0:67:05:CE:E4:0B

            X509v3 Subject Alternative Name:
                DNS:www.shubham2023.com, DNS:www.shubham2023A.com, DNS:www.shubham2023B.com
    Signature Algorithm: sha256WithRSAEncryption
         41:bf:3c:be:cf:13:bd:66:39:39:9b:c7:d1:c9:de:ad:43:b3:
         66:2a:29:0f:1b:b7:10:3c:24:08:81:da:32:8d:9b:4b:a1:c6:
         7a:1f:50:65:2c:58:13:36:57:38:a3:3b:0a:62:71:6e:68:73:
         27:90:0d:76:d0:76:59:25:da:75:df:94:ad:d4:85:4c:b2:cc:
         1b:f3:8d:33:77:25:d9:73:8a:a0:dd:10:eb:53:28:88:e0:cf:
         6a:42:66:22:92:0d:df:e8:a8:83:e0:a9:4b:10:1a:94:ec:e6:
         45:85:d1:eb:0d:cc:96:e4:38:34:ea:47:9c:d6:10:28:ab:64:
```

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

In this step, we followed below steps:

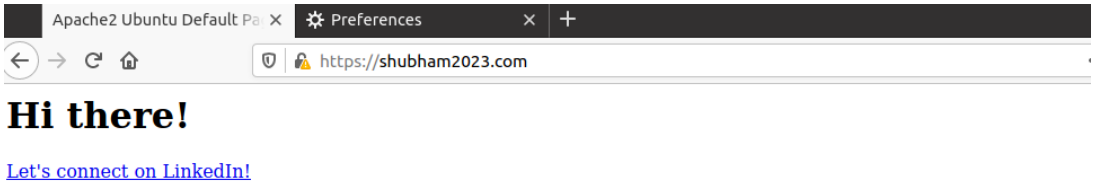
- First, we added DNS entry for **www.shubham2023.com** in the **/etc/hosts** file and added our own **index.html** file.
- Second, we added below **VirtualHost** entry in **/etc/apache2/sites-available/default-ssl.conf** file, as shown below:

```
# vim: syntax=apache ts=4 sw=4 sts=4 sr noet
<VirtualHost *:443>
        DocumentRoot /var/www/shubham2023
        ServerName www.shubham2023.com
        ServerAlias www.shubham2023A.com
        ServerAlias www.shubham2023B.com
        DirectoryIndex index.html
        SSLEngine On
        SSLCertificateFile /home/seed/HW3/shubham2023_cert.pem
        SSLCertificateKeyFile /home/seed/HW3/shubham2023_key.pem
</VirtualHost>
~
-- INSERT --
```

After making above changes, we ran few commands to enable SSL and to restart the Apache server, as shown in below snapshots.

```
seed@VM: /etc
[04/09/23]seed@VM:.../sites-available$ sudo apachectl configtest
AH00558: apache2: Could not reliably determine the server's fully qualified domain r
ally to suppress this message
Syntax OK
[04/09/23]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:
Module socache_shmcb already enabled
Module ssl already enabled
[04/09/23]seed@VM:.../sites-available$ sudo a2ensite default-ssl
Site default-ssl already enabled
[04/09/23]seed@VM:.../sites-available$ sudo service apache2 restart
Enter passphrase for SSL/TLS keys for www.shubham2023.com:443 (RSA): (press TAB ****
```

Now, we **imported** our **certificate** in **Firefox browser**, checked our server https://shubham2023.com and found that we can **successfully** browse the **HTTPS** site, as shown below:

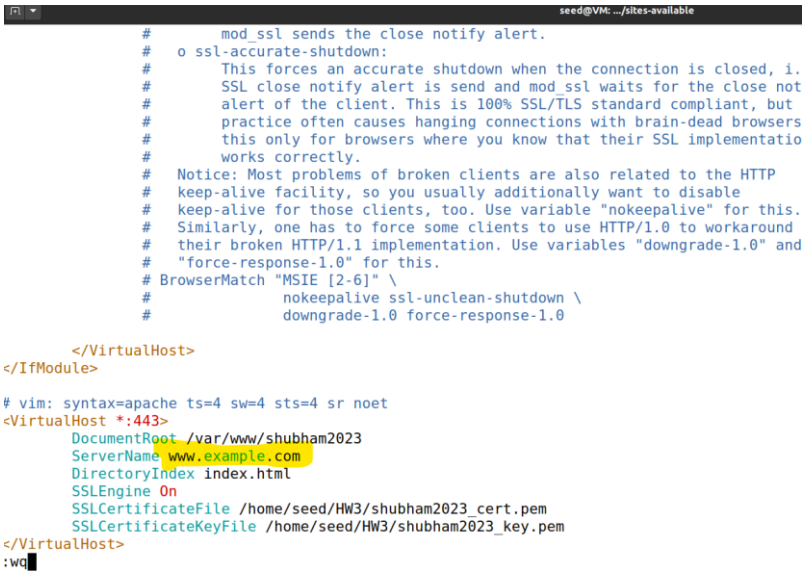


Task 5: Launching a Man-In-The-Middle Attack:

In this task, we will emulate an MITM attack, and see how exactly PKI can defeat it.

Step 1: Setting up the malicious website:

In this step, we will try to pretend to be www.example.com. To do this, we added a server name "www.example.com" in VirtualHost of /etc/apache2/sites-available/default-ssl.conf file, keeping the rest of the configurations same as before.



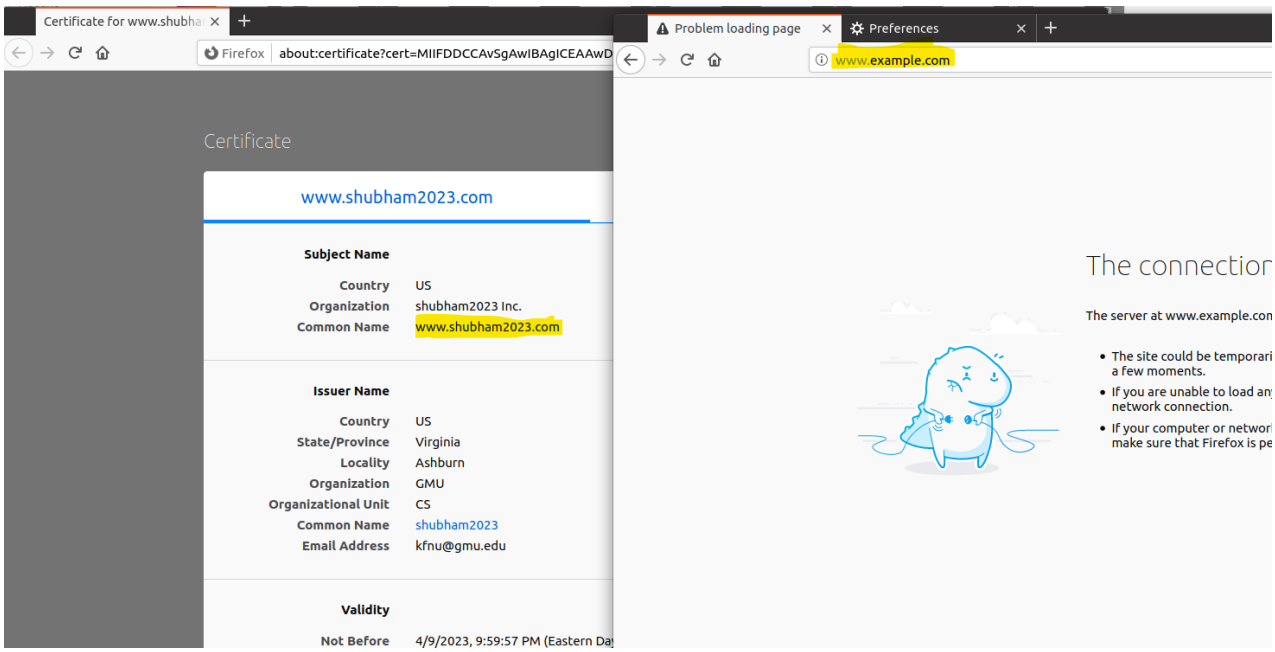
Step 2: Becoming the man in the middle:

In this task, we added the required DNS entry in the **/etc/hosts** file to perform MITM, as shown in below snapshot:

Step 3: Browse the target website:

Our Observation:

Finally, we tried to browse our target website: example.com, as you can see in the below snapshot but we get an error, because **the certificate doesn't match with the common name of the website** and hence it proves that the **MITM attack is not successful with the use of PKI**.



References:

[1] https://www.madboa.com/geek/openssl/
[2] https://docs.docker.com/