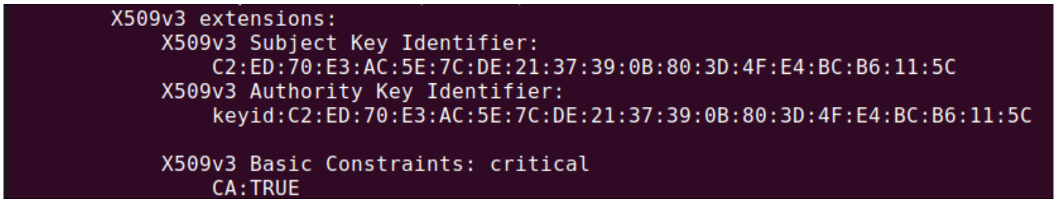
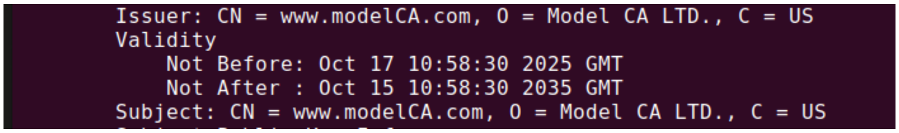
Task 1: Becoming a Certificate Authority (CA)

Looking at the output of `**openssl x509 -in ca.crt -text -noout**`:



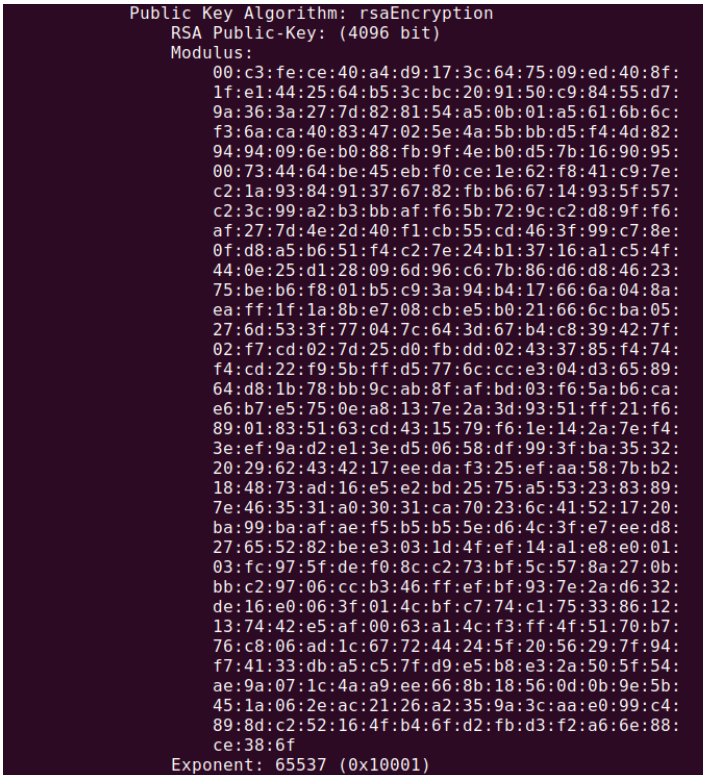
```
        X509v3 extensions:
            X509v3 Subject Key Identifier:
                C2:ED:70:E3:AC:5E:7C:DE:21:37:39:0B:80:3D:4F:E4:BC:B6:11:5C
            X509v3 Authority Key Identifier:
                keyid:C2:ED:70:E3:AC:5E:7C:DE:21:37:39:0B:80:3D:4F:E4:BC:B6:11:5C

            X509v3 Basic Constraints: critical
                CA:TRUE
```

The line "CA:TRUE" indicates that this is a CA's certificate.



```
        Issuer: CN = www.modelCA.com, O = Model CA LTD., C = US
        Validity
            Not Before: Oct 17 10:58:30 2025 GMT
            Not After : Oct 15 10:58:30 2035 GMT
        Subject: CN = www.modelCA.com, O = Model CA LTD., C = US
```

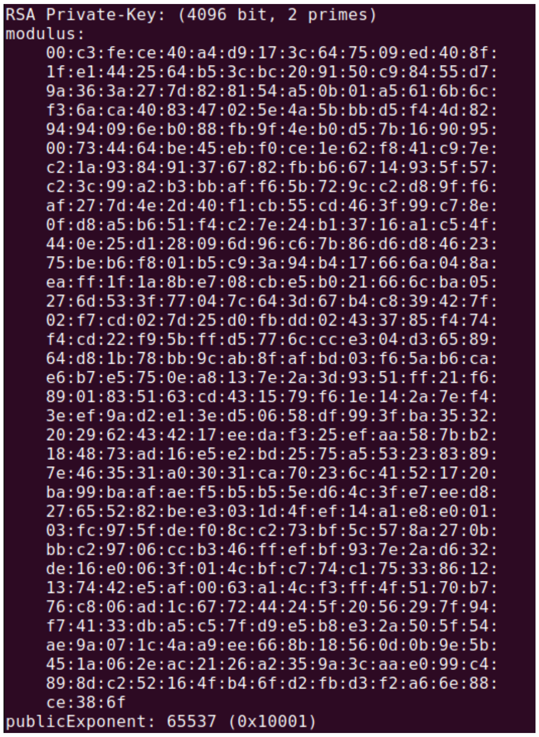
Looking at the "Issuer: CN = www.modelCA.com, O = Model CA LTD., C = US" and "Subject: CN = www.modelCA.com, O = Model CA LTD., C = US", since the issuer and the subject are exactly the same, it indicates that it is a self-signed certificate.



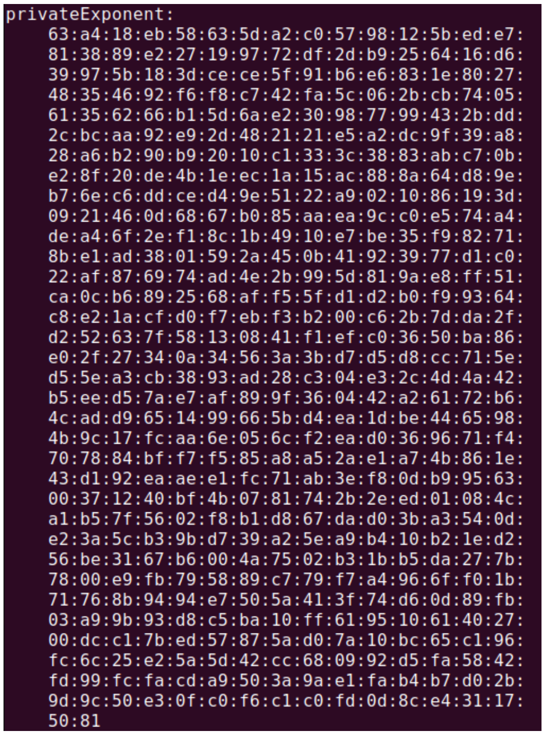
```
            Public Key Algorithm: rsaEncryption
                RSA Public-Key: (4096 bit)
                Modulus:
                    00:c3:fe:ce:40:a4:d9:17:3c:64:75:09:ed:40:8f:
                    1f:e1:44:25:64:b5:3c:bc:20:91:50:c9:84:55:d7:
                    9a:36:3a:27:7d:82:81:54:a5:0b:01:a5:61:6b:6c:
                    f3:6a:ca:40:83:47:02:5e:4a:5b:bb:d5:f4:4d:82:
                    94:94:09:6e:b0:88:fb:9f:4e:b0:d5:7b:16:90:95:
                    00:73:44:64:be:45:eb:f0:ce:1e:62:f8:41:c9:7e:
                    c2:1a:93:84:91:37:67:82:fb:b6:67:14:93:5f:57:
                    c2:3c:99:a2:b3:bb:af:f6:5b:72:9c:c2:d8:9f:f6:
                    af:27:7d:4e:2d:40:f1:cb:55:cd:46:3f:99:c7:8e:
                    0f:d8:a5:b6:51:f4:c2:7e:24:b1:37:16:a1:c5:4f:
                    44:0e:25:d1:28:09:6d:96:c6:7b:86:d6:d8:46:23:
                    75:be:b6:f8:01:b5:c9:3a:94:b4:17:66:6a:04:8a:
                    ea:ff:1f:1a:8b:e7:08:cb:e5:b0:21:66:6c:ba:05:
                    27:6d:53:3f:77:04:7c:64:3d:67:b4:c8:39:42:7f:
                    02:f7:cd:02:7d:25:d0:fb:dd:02:43:37:85:f4:74:
                    f4:cd:22:f9:5b:ff:d5:77:6c:cc:e3:04:d3:65:89:
                    64:d8:1b:78:bb:9c:ab:8f:af:bd:03:f6:5a:b6:ca:
                    e6:b7:e5:75:0e:a8:13:7e:2a:3d:93:51:ff:21:f6:
                    89:01:83:51:63:cd:43:15:79:f6:1e:14:2a:7e:f4:
                    3e:ef:9a:d2:e1:3e:d5:06:58:df:99:3f:ba:35:32:
                    20:29:62:43:42:17:ee:da:f3:25:ef:aa:58:7b:b2:
                    18:48:73:ad:16:e5:e2:bd:25:75:a5:53:23:83:89:
                    7e:46:35:31:a0:30:31:ca:70:23:6c:41:52:17:20:
                    ba:99:ba:af:ae:f5:b5:b5:5e:d6:4c:3f:e7:ee:d8:
                    27:65:52:82:be:e3:03:1d:4f:ef:14:a1:e8:e0:01:
                    03:fc:97:5f:de:f0:8c:c2:73:bf:5c:57:8a:27:0b:
                    bb:c2:97:06:cc:b3:46:ff:ef:bf:93:7e:2a:d6:32:
                    de:16:e0:06:3f:01:4c:bf:c7:74:c1:75:33:86:12:
                    13:74:42:e5:af:00:63:a1:4c:f3:ff:4f:51:70:b7:
                    76:c8:06:ad:1c:67:72:44:24:5f:20:56:29:7f:94:
                    f7:41:33:db:a5:c5:7f:d9:e5:b8:e3:2a:50:5f:54:
                    ae:9a:07:1c:4a:a9:ee:66:8b:18:56:0d:0b:9e:5b:
                    45:1a:06:2e:ac:21:26:a2:35:9a:3c:aa:e0:99:c4:
                    89:8d:c2:52:16:4f:b4:6f:d2:fb:d3:f2:a6:6e:88:
                    ce:38:6f
                Exponent: 65537 (0x10001)
```

ca.crt contains the modulus value, n, in hexadecimal format, and the public exponent e = 65537.

Looking at the output of `**openssl rsa -in ca.key -text -noout**`:

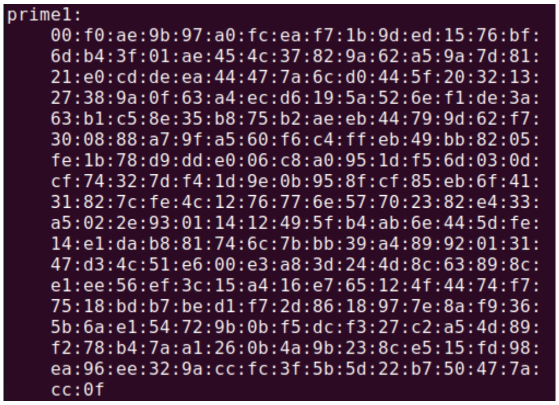


```
RSA Private-Key: (4096 bit, 2 primes)
modulus:
    00:c3:fe:ce:40:a4:d9:17:3c:64:75:09:ed:40:8f:
    1f:e1:44:25:64:b5:3c:bc:20:91:50:c9:84:55:d7:
    9a:36:3a:27:7d:82:81:54:a5:0b:01:a5:61:6b:6c:
    f3:6a:ca:40:83:47:02:5e:4a:5b:bb:d5:f4:4d:82:
    94:94:09:6e:b0:88:fb:9f:4e:b0:d5:7b:16:90:95:
    00:73:44:64:be:45:eb:f0:ce:1e:62:f8:41:c9:7e:
    c2:1a:93:84:91:37:67:82:fb:b6:67:14:93:5f:57:
    c2:3c:99:a2:b3:bb:af:f6:5b:72:9c:c2:d8:9f:f6:
    af:27:7d:4e:2d:40:f1:cb:55:cd:46:3f:99:c7:8e:
    0f:d8:a5:b6:51:f4:c2:7e:24:b1:37:16:a1:c5:4f:
    44:0e:25:d1:28:09:6d:96:c6:7b:86:d6:d8:46:23:
    75:be:b6:f8:01:b5:c9:3a:94:b4:17:66:6a:04:8a:
    ea:ff:1f:1a:8b:e7:08:cb:e5:b0:21:66:6c:ba:05:
    27:6d:53:3f:77:04:7c:64:3d:67:b4:c8:39:42:7f:
    02:f7:cd:02:7d:25:d0:fb:dd:02:43:37:85:f4:74:
    f4:cd:22:f9:5b:ff:d5:77:6c:cc:e3:04:d3:65:89:
    64:d8:1b:78:bb:9c:ab:8f:af:bd:03:f6:5a:b6:ca:
    e6:b7:e5:75:0e:a8:13:7e:2a:3d:93:51:ff:21:f6:
    89:01:83:51:63:cd:43:15:79:f6:1e:14:2a:7e:f4:
    3e:ef:9a:d2:e1:3e:d5:06:58:df:99:3f:ba:35:32:
    20:29:62:43:42:17:ee:da:f3:25:ef:aa:58:7b:b2:
    18:48:73:ad:16:e5:e2:bd:25:75:a5:53:23:83:89:
    7e:46:35:31:a0:30:31:ca:70:23:6c:41:52:17:20:
    ba:99:ba:af:ae:f5:b5:b5:5e:d6:4c:3f:e7:ee:d8:
    27:65:52:82:be:e3:03:1d:4f:ef:14:a1:e8:e0:01:
    03:fc:97:5f:de:f0:8c:c2:73:bf:5c:57:8a:27:0b:
    bb:c2:97:06:cc:b3:46:ff:ef:bf:93:7e:2a:d6:32:
    de:16:e0:06:3f:01:4c:bf:c7:74:c1:75:33:86:12:
    13:74:42:e5:af:00:63:a1:4c:f3:ff:4f:51:70:b7:
    76:c8:06:ad:1c:67:72:44:24:5f:20:56:29:7f:94:
    f7:41:33:db:a5:c5:7f:d9:e5:b8:e3:2a:50:5f:54:
    ae:9a:07:1c:4a:a9:ee:66:8b:18:56:0d:0b:9e:5b:
    45:1a:06:2e:ac:21:26:a2:35:9a:3c:aa:e0:99:c4:
    89:8d:c2:52:16:4f:b4:6f:d2:fb:d3:f2:a6:6e:88:
    ce:38:6f
publicExponent: 65537 (0x10001)
```

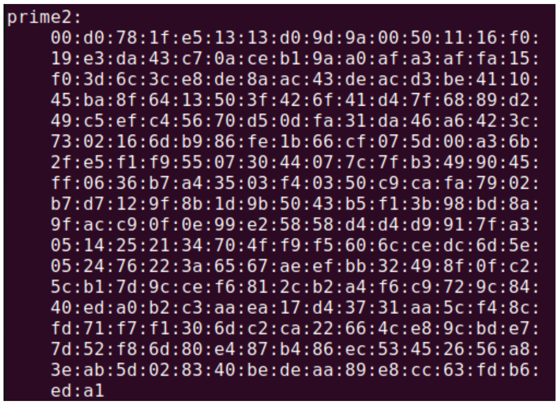


```
privateExponent:
    63:a4:18:eb:58:63:5d:a2:c0:57:98:12:5b:ed:e7:
    81:38:89:e2:27:19:97:72:df:2d:b9:25:64:16:d6:
    39:97:5b:18:3d:ce:ce:5f:91:b6:e6:83:1e:80:27:
    48:35:46:92:f6:f8:c7:42:fa:5c:06:2b:cb:74:05:
    61:35:62:66:b1:5d:6a:e2:30:98:77:99:43:2b:dd:
    2c:bc:aa:92:e9:2d:48:21:21:e5:a2:dc:9f:39:a8:
    28:a6:b2:90:b9:20:10:c1:33:3c:38:83:ab:c7:0b:
    e2:8f:20:de:4b:1e:ec:1a:15:ac:88:8a:64:d8:9e:
    b7:6e:c6:dd:ce:d4:9e:51:22:a9:02:10:86:19:3d:
    09:21:46:0d:68:67:b0:85:aa:ea:9c:c0:e5:74:a4:
    de:a4:6f:2e:f1:8c:1b:49:10:e7:be:35:f9:82:71:
    8b:e1:ad:38:01:59:2a:45:0b:41:92:39:77:d1:c0:
    22:af:87:69:74:ad:4e:2b:99:5d:81:9a:e8:ff:51:
    ca:0c:b6:89:25:68:af:f5:5f:d1:d2:b0:f9:93:64:
    c8:e2:1a:cf:d0:f7:eb:f3:b2:00:c6:2b:7d:da:2f:
    d2:52:63:7f:58:13:08:41:f1:ef:c0:36:50:ba:86:
    e0:2f:27:34:0a:34:56:3a:3b:d7:d5:d8:cc:71:5e:
    d5:5e:a3:cb:38:93:ad:28:c3:04:e3:2c:4d:4a:42:
    b5:ee:d5:7a:e7:af:89:9f:36:04:42:a2:61:72:b6:
    4c:ad:d9:65:14:99:66:5b:d4:ea:1d:be:44:65:98:
    4b:9c:17:fc:aa:6e:05:6c:f2:ea:d0:36:96:71:f4:
    70:78:84:bf:f7:f5:85:a8:a5:2a:e1:a7:4b:86:1e:
    43:d1:92:ea:ae:e1:fc:71:ab:3e:f8:0d:b9:95:63:
    00:37:12:40:bf:4b:07:81:74:2b:2e:ed:01:08:4c:
    a1:b5:7f:56:02:f8:b1:d8:67:da:d0:3b:a3:54:0d:
    e2:3a:5c:b3:9b:d7:39:a2:5e:a9:b4:10:b2:1e:d2:
    56:be:31:67:b6:00:4a:75:02:b3:1b:b5:da:27:7b:
    78:00:e9:fb:79:58:89:c7:79:f7:a4:96:6f:f0:1b:
    71:76:8b:94:94:e7:50:5a:41:3f:74:d6:0d:89:fb:
    03:a9:9b:93:d8:c5:ba:10:ff:61:95:10:61:40:27:
    00:dc:c1:7b:ed:57:87:5a:d0:7a:10:bc:65:c1:96:
    fc:6c:25:e2:5a:5d:42:cc:68:09:92:d5:fa:58:42:
    fd:99:fc:fa:cd:a9:50:3a:9a:e1:fa:b4:b7:d0:2b:
    9d:9c:50:e3:0f:c0:f6:c1:c0:fd:0d:8c:e4:31:17:
    50:81
```

ca.key also contains the modulus value, n, in hexadecimal format, the public exponent e = 65537, and the private exponent value, d, in hexadecimal format on the right.



```
prime1:
    00:f0:ae:9b:97:a0:fc:ea:f7:1b:9d:ed:15:76:bf:
    6d:b4:3f:01:ae:45:4c:37:82:9a:62:a5:9a:7d:81:
    21:e0:cd:de:ea:44:47:7a:6c:d0:44:5f:20:32:13:
    27:38:9a:0f:63:a4:ec:d6:19:5a:52:6e:f1:de:3a:
    63:b1:c5:8e:35:b8:75:b2:ae:eb:44:79:9d:62:f7:
    30:08:88:a7:9f:a5:60:f6:c4:ff:eb:49:bb:82:05:
    fe:1b:78:d9:dd:e0:06:c8:a0:95:1d:f5:6d:03:0d:
    cf:74:32:7d:f4:1d:9e:0b:95:8f:cf:85:eb:6f:41:
    31:82:7c:fe:4c:12:76:77:6e:57:70:23:82:e4:33:
    a5:02:2e:93:01:14:12:49:5f:b4:ab:6e:44:5d:fe:
    14:e1:da:b8:81:74:6c:7b:bb:39:a4:89:92:01:31:
    47:d3:4c:51:e6:00:e3:a8:3d:24:4d:8c:63:89:8c:
    e1:ee:56:ef:3c:15:a4:16:e7:65:12:4f:44:74:f7:
    75:18:bd:b7:be:d1:f7:2d:86:18:97:7e:8a:f9:36:
    5b:6a:e1:54:72:9b:0b:f5:dc:f3:27:c2:a5:4d:89:
    f2:78:b4:7a:a1:26:0b:4a:9b:23:8c:e5:15:fd:98:
    ea:96:ee:32:9a:cc:fc:3f:5b:5d:22:b7:50:47:7a:
    cc:0f
```

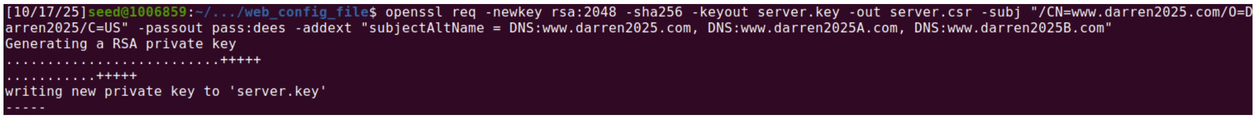


```
prime2:
    00:d0:78:1f:e5:13:13:d0:9d:9a:00:50:11:16:f0:
    19:e3:da:43:c7:0a:ce:b1:9a:a0:af:a3:af:fa:15:
    f0:3d:6c:3c:e8:de:8a:ac:43:de:ac:d3:be:41:10:
    45:ba:8f:64:13:50:3f:42:6f:41:d4:7f:68:89:d2:
    49:c5:ef:c4:56:70:d5:0d:fa:31:da:46:a6:42:3c:
    73:02:16:6d:b9:86:fe:1b:66:cf:07:5d:00:a3:6b:
    2f:e5:f1:f9:55:07:30:44:07:7c:7f:b3:49:90:45:
    ff:06:36:b7:a4:35:03:f4:03:50:c9:ca:fa:79:02:
    b7:d7:12:9f:8b:1d:9b:50:43:b5:f1:3b:98:bd:8a:
    9f:ac:c9:0f:0e:99:e2:58:58:d4:d4:d9:91:7f:a3:
    05:14:25:21:34:70:4f:f9:f5:60:6c:ce:dc:6d:5e:
    05:24:76:22:3a:65:67:ae:ef:bb:32:49:8f:0f:c2:
    5c:b1:7d:9c:ce:f6:81:2c:b2:a4:f6:c9:72:9c:84:
    40:ed:a0:b2:c3:aa:ea:17:d4:37:31:aa:5c:f4:8c:
    fd:71:f7:f1:30:6d:c2:ca:22:66:4c:e8:9c:bd:e7:
    7d:52:f8:6d:80:e4:87:b4:86:ec:53:45:26:56:a8:
    3e:ab:5d:02:83:40:be:de:aa:89:e8:cc:63:fd:b6:
    ed:a1
```

ca.key also contains the 2 secret numbers p and q as shown above, prime1 and prime2, in hexadecimal format.

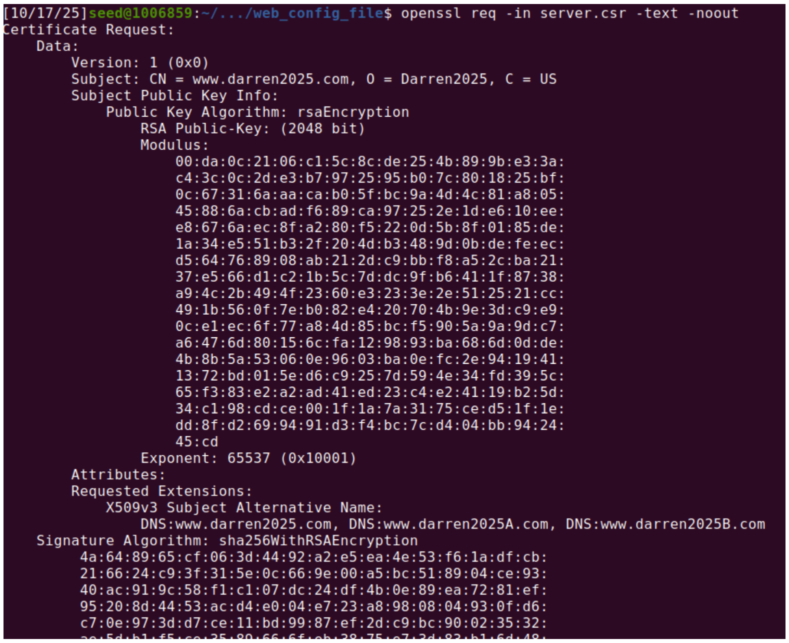
Task 2: Generating a Certificate Request for Your Web Server

In /etc/hosts, I defined my server name to be www.darren2025.com



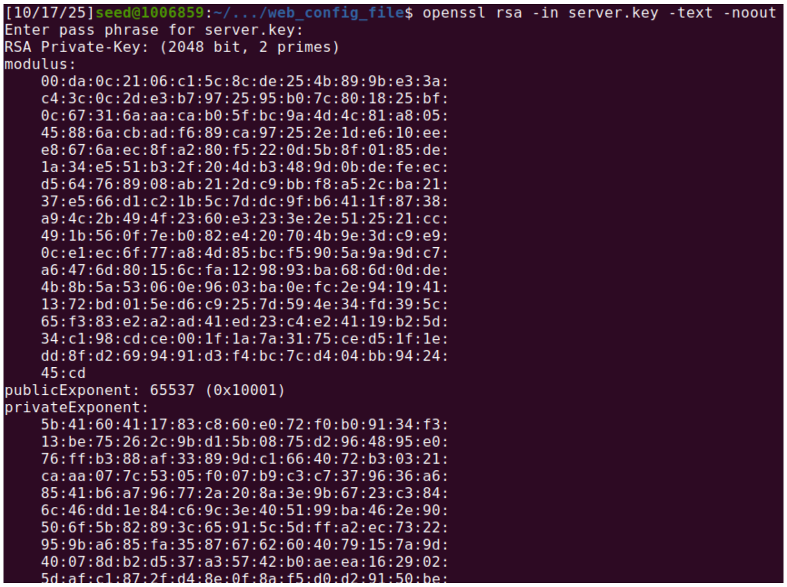
```
[10/17/25]seed@1006859:~/.../web_config_file$ openssl req -newkey rsa:2048 -sha256 -keyout server.key -out server.csr -subj "/CN=www.darren2025.com/O=D
arren2025/C=US" -passout pass:dees -addext "subjectAltName = DNS:www.darren2025.com, DNS:www.darren2025A.com, DNS:www.darren2025B.com"
Generating a RSA private key
.........................+++++
...........+++++
writing new private key to 'server.key'
-----
```

With this, I generated a certificate request for my web server with 2 alternative names to my certificate signing request, www.darren2025A.com and www.darren2025B.com. I obtained server.csr and server.key as shown below:



```
[10/17/25]seed@1006859:~/.../web_config_file$ openssl req -in server.csr -text -noout
Certificate Request:
    Data:
        Version: 1 (0x0)
        Subject: CN = www.darren2025.com, O = Darren2025, C = US
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
                RSA Public-Key: (2048 bit)
                Modulus:
                    00:da:0c:21:06:c1:5c:8c:de:25:4b:89:9b:e3:3a:
                    c4:3c:0c:2d:e3:b7:97:25:95:b0:7c:80:18:25:bf:
                    0c:67:31:6a:aa:ca:b0:5f:bc:9a:4d:4c:81:a8:05:
                    45:88:6a:cb:ad:f6:89:ca:97:25:2e:1d:e6:10:ee:
                    e8:67:6a:ec:8f:a2:80:f5:22:0d:5b:8f:01:85:de:
                    1a:34:e5:51:b3:2f:20:4d:b3:48:9d:0b:de:fe:ec:
                    d5:64:76:89:08:ab:21:2d:c9:bb:f8:a5:2c:ba:21:
                    37:e5:66:d1:c2:1b:5c:7d:dc:9f:b6:41:1f:87:38:
                    a9:4c:2b:49:4f:23:60:e3:23:3e:2e:51:25:21:cc:
                    49:1b:56:0f:7e:b0:82:e4:20:70:4b:9e:3d:c9:e9:
                    0c:e1:ec:6f:77:a8:4d:85:bc:f5:90:5a:9a:9d:c7:
                    a6:47:6d:80:15:6c:fa:12:98:93:ba:68:6d:0d:de:
                    4b:8b:5a:53:06:0e:96:03:ba:0e:fc:2e:94:19:41:
                    13:72:bd:01:5e:d6:c9:25:7d:59:4e:34:fd:39:5c:
                    65:f3:83:e2:a2:ad:41:ed:23:c4:e2:41:19:b2:5d:
                    34:c1:98:cd:ce:00:1f:1a:7a:31:75:ce:d5:1f:1e:
                    dd:8f:d2:69:94:91:d3:f4:bc:7c:d4:04:bb:94:24:
                    45:cd
                Exponent: 65537 (0x10001)
        Attributes:
        Requested Extensions:
            X509v3 Subject Alternative Name:
                DNS:www.darren2025.com, DNS:www.darren2025A.com, DNS:www.darren2025B.com
    Signature Algorithm: sha256WithRSAEncryption
         4a:64:89:65:cf:06:3d:44:92:a2:e5:ea:4e:53:f6:1a:df:cb:
         21:66:24:c9:3f:31:5e:0c:66:9e:00:a5:bc:51:89:04:ce:93:
         40:ac:91:9c:58:f1:c1:07:dc:24:df:4b:0e:89:ea:72:81:ef:
         95:20:8d:44:53:ac:d4:e0:04:e7:23:a8:98:08:04:93:0f:d6:
         c7:0e:97:3d:d7:ce:11:bd:99:87:ef:2d:c9:bc:90:02:35:32:
         ae:5d:b1:f5:ce:35:89:66:6f:eb:38:75:e7:3d:83:b1:6d:48:
```

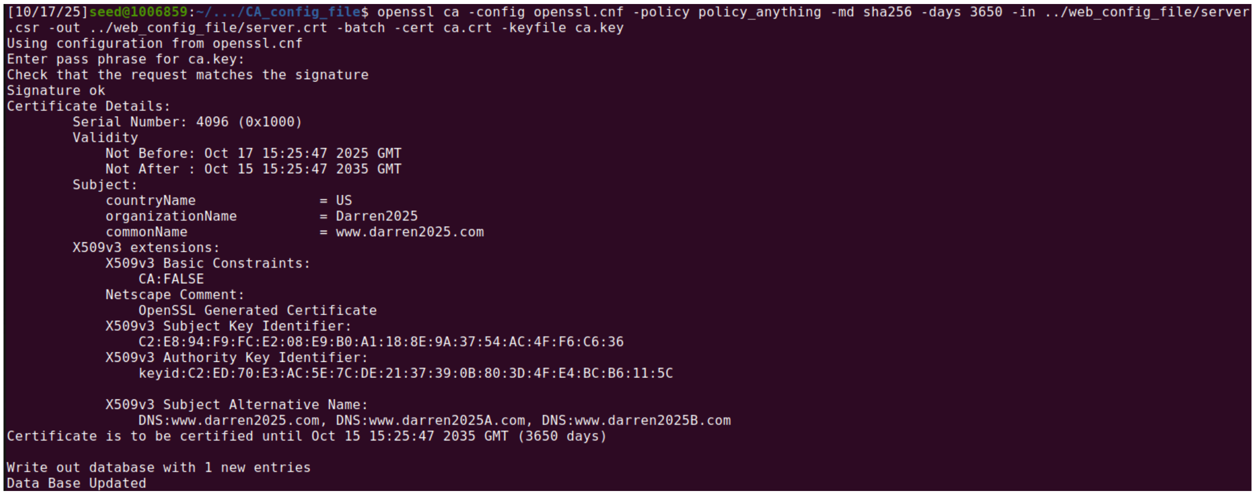
We can observe the SAN that we defined above.



```
[10/17/25]seed@1006859:~/.../web_config_file$ openssl rsa -in server.key -text -noout
Enter pass phrase for server.key:
RSA Private-Key: (2048 bit, 2 primes)
modulus:
    00:da:0c:21:06:c1:5c:8c:de:25:4b:89:9b:e3:3a:
    c4:3c:0c:2d:e3:b7:97:25:95:b0:7c:80:18:25:bf:
    0c:67:31:6a:aa:ca:b0:5f:bc:9a:4d:4c:81:a8:05:
    45:88:6a:cb:ad:f6:89:ca:97:25:2e:1d:e6:10:ee:
    e8:67:6a:ec:8f:a2:80:f5:22:0d:5b:8f:01:85:de:
    1a:34:e5:51:b3:2f:20:4d:b3:48:9d:0b:de:fe:ec:
    d5:64:76:89:08:ab:21:2d:c9:bb:f8:a5:2c:ba:21:
    37:e5:66:d1:c2:1b:5c:7d:dc:9f:b6:41:1f:87:38:
    a9:4c:2b:49:4f:23:60:e3:23:3e:2e:51:25:21:cc:
    49:1b:56:0f:7e:b0:82:e4:20:70:4b:9e:3d:c9:e9:
    0c:e1:ec:6f:77:a8:4d:85:bc:f5:90:5a:9a:9d:c7:
    a6:47:6d:80:15:6c:fa:12:98:93:ba:68:6d:0d:de:
    4b:8b:5a:53:06:0e:96:03:ba:0e:fc:2e:94:19:41:
    13:72:bd:01:5e:d6:c9:25:7d:59:4e:34:fd:39:5c:
    65:f3:83:e2:a2:ad:41:ed:23:c4:e2:41:19:b2:5d:
    34:c1:98:cd:ce:00:1f:1a:7a:31:75:ce:d5:1f:1e:
    dd:8f:d2:69:94:91:d3:f4:bc:7c:d4:04:bb:94:24:
    45:cd
publicExponent: 65537 (0x10001)
privateExponent:
    5b:41:60:41:17:83:c8:60:e0:72:f0:b0:91:34:f3:
    13:be:75:26:2c:9b:d1:5b:08:75:d2:96:48:95:e0:
    76:ff:b3:88:af:33:89:9d:c1:66:40:72:b3:03:21:
    ca:aa:07:7c:53:05:f0:07:b9:c3:c7:37:96:36:a6:
    85:41:b6:a7:96:77:2a:20:8a:3e:9b:67:23:c3:84:
    6c:46:dd:1e:84:c6:9c:3e:40:51:99:ba:46:2e:90:
    50:6f:5b:82:89:3c:65:91:5c:5d:ff:a2:ec:73:22:
    95:9b:a6:85:fa:35:87:67:62:60:40:79:15:7a:9d:
    40:07:8d:b2:d5:37:a3:57:42:b0:ae:ea:16:29:02:
    5d:af:c1:87:2f:d4:8e:0f:8a:f5:d0:d2:91:50:be:
```

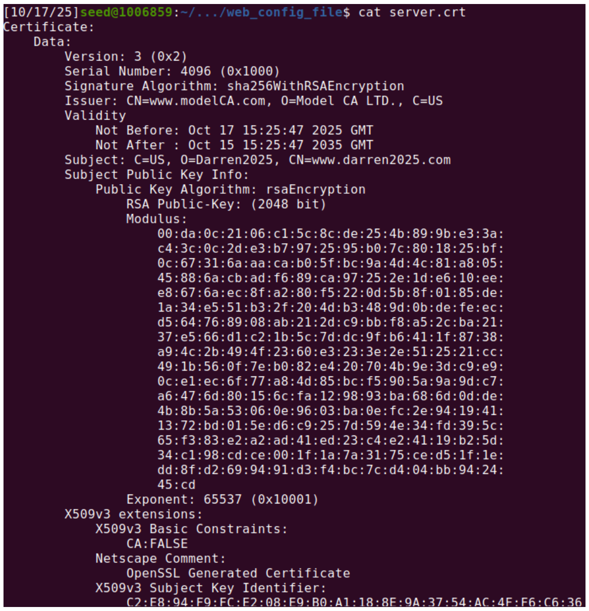
Task 3: Generating a Certificate for your server

Using the openssl.cnf file that I edited after copying from /usr/lib/ssl/openssl.cnf, I signed the web server’s certificate with my own trusted CA.

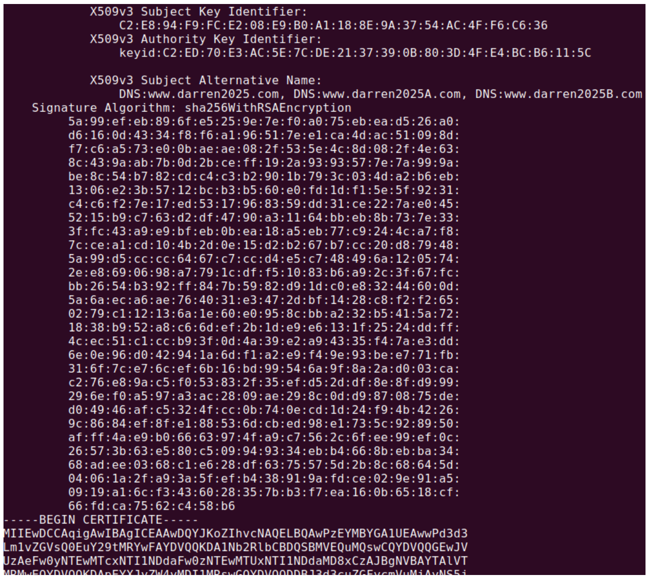


```
[10/17/25]seed@1006859:~/.../CA_config_file$ openssl ca -config openssl.cnf -policy policy_anything -md sha256 -days 3650 -in ../web_config_file/server
.csr -out ../web_config_file/server.crt -batch -cert ca.crt -keyfile ca.key
Using configuration from openssl.cnf
Enter pass phrase for ca.key:
Check that the request matches the signature
Signature ok
Certificate Details:
        Serial Number: 4096 (0x1000)
        Validity
            Not Before: Oct 17 15:25:47 2025 GMT
            Not After : Oct 15 15:25:47 2035 GMT
        Subject:
            countryName               = US
            organizationName          = Darren2025
            commonName                = www.darren2025.com
        X509v3 extensions:
            X509v3 Basic Constraints:
                CA:FALSE
            Netscape Comment:
                OpenSSL Generated Certificate
            X509v3 Subject Key Identifier:
                C2:E8:94:F9:FC:E2:08:E9:B0:A1:18:8E:9A:37:54:AC:4F:F6:C6:36
            X509v3 Authority Key Identifier:
                keyid:C2:ED:70:E3:AC:5E:7C:DE:21:37:39:0B:80:3D:4F:E4:BC:B6:11:5C

            X509v3 Subject Alternative Name:
                DNS:www.darren2025.com, DNS:www.darren2025A.com, DNS:www.darren2025B.com
Certificate is to be certified until Oct 15 15:25:47 2035 GMT (3650 days)

Write out database with 1 new entries
Data Base Updated
```

The following pictures show some of the contents of the signed web server’s certificate:



```
[10/17/25]seed@1006859:~/.../web_config_file$ cat server.crt
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number: 4096 (0x1000)
        Signature Algorithm: sha256WithRSAEncryption
        Issuer: CN=www.modelCA.com, O=Model CA LTD., C=US
        Validity
            Not Before: Oct 17 15:25:47 2025 GMT
            Not After : Oct 15 15:25:47 2035 GMT
        Subject: C=US, O=Darren2025, CN=www.darren2025.com
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
                RSA Public-Key: (2048 bit)
                Modulus:
                    00:da:0c:21:06:c1:5c:8c:de:25:4b:89:9b:e3:3a:
                    c4:3c:0c:2d:e3:b7:97:25:95:b0:7c:80:18:25:bf:
                    0c:67:31:6a:aa:ca:b0:5f:bc:9a:4d:4c:81:a8:05:
                    45:88:6a:cb:ad:f6:89:ca:97:25:2e:1d:e6:10:ee:
                    e8:67:6a:ec:8f:a2:80:f5:22:0d:5b:8f:01:85:de:
                    1a:34:e5:51:b3:2f:20:4d:b3:48:9d:0b:de:fe:ec:
                    d5:64:76:89:08:ab:21:2d:c9:bb:f8:a5:2c:ba:21:
                    37:e5:66:d1:c2:1b:5c:7d:dc:9f:b6:41:1f:87:38:
                    a9:4c:2b:49:4f:23:60:e3:23:3e:2e:51:25:21:cc:
                    49:1b:56:0f:7e:b0:82:e4:20:70:4b:9e:3d:c9:e9:
                    0c:e1:ec:6f:77:a8:4d:85:bc:f5:90:5a:9a:9d:c7:
                    a6:47:6d:80:15:6c:fa:12:98:93:ba:68:6d:0d:de:
                    4b:8b:5a:53:06:0e:96:03:ba:0e:fc:2e:94:19:41:
                    13:72:bd:01:5e:d6:c9:25:7d:59:4e:34:fd:39:5c:
                    65:f3:83:e2:a2:ad:41:ed:23:c4:e2:41:19:b2:5d:
                    34:c1:98:cd:ce:00:1f:1a:7a:31:75:ce:d5:1f:1e:
                    dd:8f:d2:69:94:91:d3:f4:bc:7c:d4:04:bb:94:24:
                    45:cd
                Exponent: 65537 (0x10001)
        X509v3 extensions:
            X509v3 Basic Constraints:
                CA:FALSE
            Netscape Comment:
                OpenSSL Generated Certificate
            X509v3 Subject Key Identifier:
                C2:E8:94:F9:FC:E2:08:E9:B0:A1:18:8E:9A:37:54:AC:4F:F6:C6:36
```

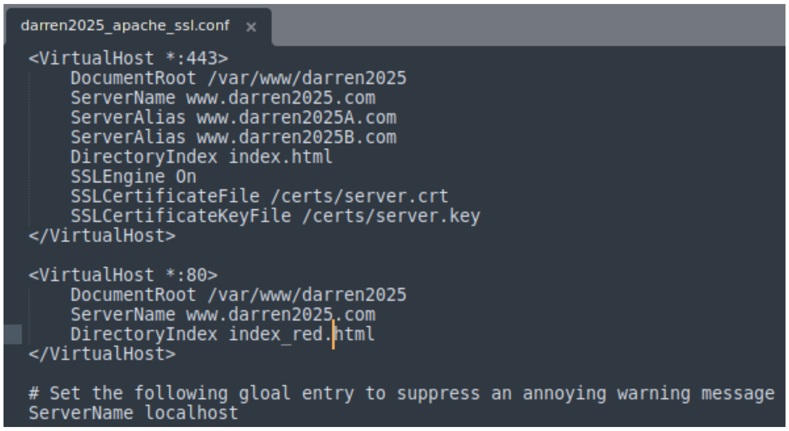


```
            X509v3 Subject Key Identifier:
                C2:E8:94:F9:FC:E2:08:E9:B0:A1:18:8E:9A:37:54:AC:4F:F6:C6:36
            X509v3 Authority Key Identifier:
                keyid:C2:ED:70:E3:AC:5E:7C:DE:21:37:39:0B:80:3D:4F:E4:BC:B6:11:5C

            X509v3 Subject Alternative Name:
                DNS:www.darren2025.com, DNS:www.darren2025A.com, DNS:www.darren2025B.com
    Signature Algorithm: sha256WithRSAEncryption
         5a:99:ef:eb:89:6f:e5:25:9e:7e:f0:a0:75:eb:ea:d5:26:a0:
         d6:16:0d:43:34:f8:f6:a1:96:51:7e:e1:ca:4d:ac:51:09:8d:
         f7:c6:a5:73:e0:0b:ae:ae:08:2f:53:5e:4c:8d:08:2f:4e:63:
         8c:43:9a:ab:7b:0d:2b:ce:ff:19:2a:93:93:57:7e:7a:99:9a:
         be:8c:54:b7:82:cd:c4:c3:b2:90:1b:79:3c:03:4d:a2:b6:eb:
         13:06:e2:3b:57:12:bc:b3:b5:60:e0:fd:1d:f1:5e:5f:92:31:
         c4:c6:f2:7e:17:ed:53:17:96:83:59:dd:31:ce:22:7a:e0:45:
         52:15:b9:c7:63:d2:df:47:90:a3:11:64:bb:eb:8b:73:7e:33:
         3f:fc:43:a9:e9:bf:eb:0b:ea:18:a5:eb:77:c9:24:4c:a7:f8:
         7c:ce:a1:cd:10:4b:2d:0e:15:d2:b2:67:b7:cc:20:d8:79:48:
         5a:99:d5:cc:cc:64:67:c7:cc:d4:e5:c7:48:49:6a:12:05:74:
         2e:e8:69:06:98:a7:79:1c:df:f5:10:83:b6:a9:2c:3f:67:fc:
         bb:26:54:b3:92:ff:84:7b:59:82:d9:1d:c0:e8:32:44:60:0d:
         5a:6a:ec:a6:ae:76:40:31:e3:47:2d:bf:14:28:c8:f2:f2:65:
         02:79:c1:12:13:6a:1e:60:e0:95:8c:bb:a2:32:b5:41:5a:72:
         18:38:b9:52:a8:c6:6d:ef:2b:1d:e9:e6:13:1f:25:24:dd:ff:
         4c:ec:51:c1:cc:b9:3f:0d:4a:39:e2:a9:43:35:f4:7a:e3:dd:
         6e:0e:96:d0:42:94:1a:6d:f1:a2:e9:f4:9e:93:be:e7:71:fb:
         31:6f:7c:e7:6c:ef:6b:16:bd:99:54:6a:9f:8a:2a:d0:03:ca:
         c2:76:e8:9a:c5:f0:53:83:2f:35:ef:d5:2d:df:8e:8f:d9:99:
         29:6e:f0:a5:97:a3:ac:28:09:ae:29:8c:0d:d9:87:08:75:de:
         d0:49:46:af:c5:32:4f:cc:0b:74:0e:cd:1d:24:f9:4b:42:26:
         9c:86:84:ef:8f:e1:88:53:6d:cb:ed:98:e1:73:5c:92:89:50:
         af:ff:4a:e9:b0:66:63:97:4f:a9:c7:56:2c:6f:ee:99:ef:0c:
         26:57:3b:63:e5:80:c5:09:94:93:34:eb:b4:66:8b:eb:ba:34:
         68:ad:ee:03:68:c1:e6:28:df:63:75:57:5d:2b:8c:68:64:5d:
         04:06:1a:2f:a9:3a:5f:ef:b4:38:91:9a:fd:ce:02:9e:91:a5:
         09:19:a1:6c:f3:43:60:28:35:7b:b3:f7:ea:16:0b:65:18:cf:
         66:fd:ca:75:62:c4:58:b6
-----BEGIN CERTIFICATE-----
MIIEwDCCAqigAwIBAgICEAAwDQYJKoZIhvcNAQELBQAwPzEYMBYGA1UEAwwPd3d3
Lm1vZGVsQ0EuY29tMRYwFAYDVQQKDA1Nb2RlbCBDQSBMVEQuMQswCQYDVQQGEwJV
UzAeFw0yNTEwMTcxNTI1NDdaFw0zNTEwMTUxNTI1NDdaMD8xCzAJBgNVBAYTAlVT
```

From the picture on the right, we can see under “X509v3 Subject Alternative Name:” that our alternative names are included in the certificate.

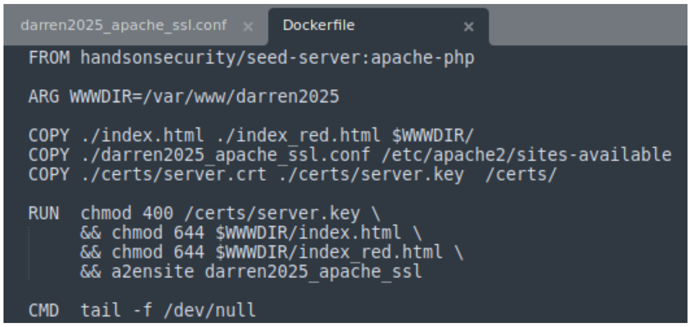
Task 4: Deploying Certificate in an Apache-Based HTTPS Website

First, I created my own apache_ssl.conf, with my own server name, www.darren2025.com and its aliases, www.darren2025A.com and www.darren2025B.com. I configured it to use my server.crt and server.key after transferring them to the /certs folder in the image-www folder.



```
<VirtualHost *:443>
    DocumentRoot /var/www/darren2025
    ServerName www.darren2025.com
    ServerAlias www.darren2025A.com
    ServerAlias www.darren2025B.com
    DirectoryIndex index.html
    SSLEngine On
    SSLCertificateFile /certs/server.crt
    SSLCertificateKeyFile /certs/server.key
</VirtualHost>

<VirtualHost *:80>
    DocumentRoot /var/www/darren2025
    ServerName www.darren2025.com
    DirectoryIndex index_red.html
</VirtualHost>

# Set the following gloal entry to suppress an annoying warning message
ServerName localhost
```

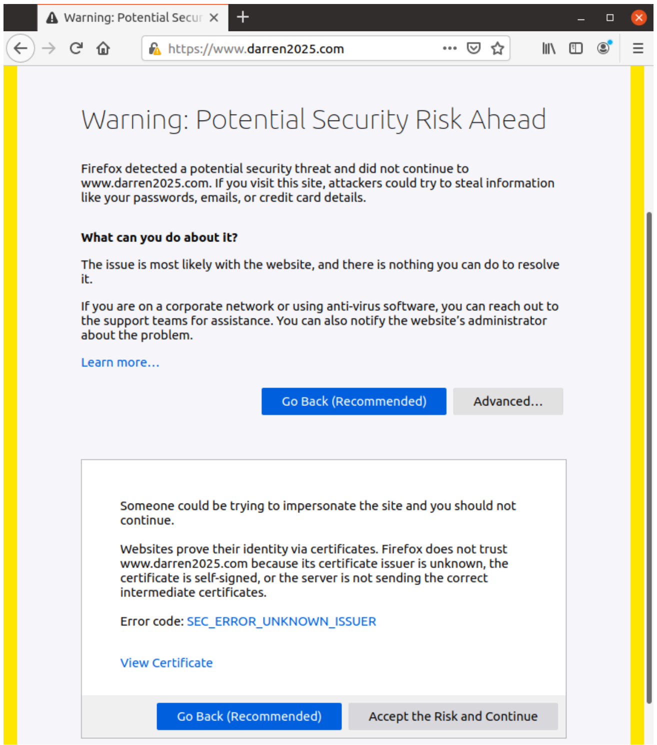
Then, I edited the Dockerfile to use the configurations I have made for my own web server.



```
FROM handsonsecurity/seed-server:apache-php

ARG WWWDIR=/var/www/darren2025

COPY ./index.html ./index_red.html $WWWDIR/
COPY ./darren2025_apache_ssl.conf /etc/apache2/sites-available
COPY ./certs/server.crt ./certs/server.key  /certs/

RUN  chmod 400 /certs/server.key \
     && chmod 644 $WWWDIR/index.html \
     && chmod 644 $WWWDIR/index_red.html \
     && a2ensite darren2025_apache_ssl

CMD  tail -f /dev/null
```

Then, I `dcbuild` again to load my configurations onto the www-10.9.0.80 container and enabled the site using “a2enmod ssl” and “a2ensite darren2025_apache_ssl”. After that, I started the Apache server from the container with “service apache2 start”.

Upon visiting https://www.darren2025.com, I am faced with the following error, stating that there is a potential security risk ahead. Looking into the reasons, I noticed that we are not able to succeed because Firefox does not trust the certificate issuer of my web server. This is because the web server's certificate is self-signed by our CA, so it needs our CA to verify its identity.



Warning: Potential Security Risk Ahead

Firefox detected a potential security threat and did not continue to www.darren2025.com. If you visit this site, attackers could try to steal information like your passwords, emails, or credit card details.

What can you do about it?

The issue is most likely with the website, and there is nothing you can do to resolve it.

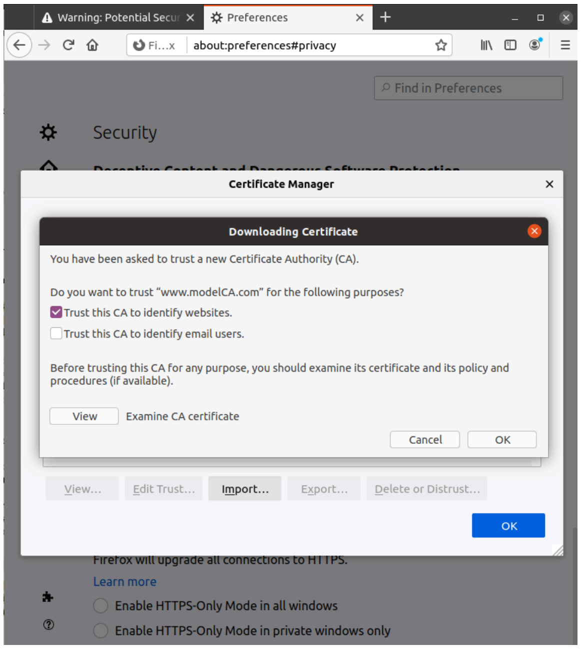
If you are on a corporate network or using anti-virus software, you can reach out to the support teams for assistance. You can also notify the website's administrator about the problem.

Someone could be trying to impersonate the site and you should not continue.

Websites prove their identity via certificates. Firefox does not trust www.darren2025.com because its certificate issuer is unknown, the certificate is self-signed, or the server is not sending the correct intermediate certificates.

Error code: SEC_ERROR_UNKNOWN_ISSUER

To fix the above problem, I imported my certificate, ca.crt, into Firefox so that our CA can be trusted to verify the identity of the website www.darren2025.com.



You have been asked to trust a new Certificate Authority (CA).

Do you want to trust "www.modelCA.com" for the following purposes?

☑ Trust this CA to identify websites.

□ Trust this CA to identify email users.

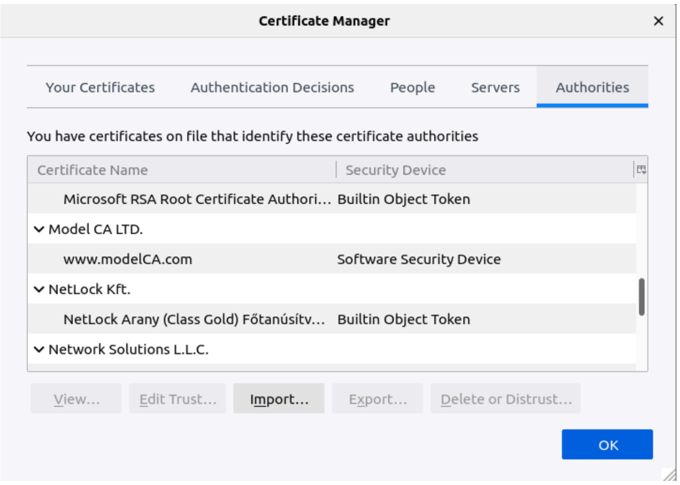
Before trusting this CA for any purpose, you should examine its certificate and its policy and procedures (if available).

Firefox will upgrade all connections to HTTPS.

○ Enable HTTPS-Only Mode in all windows

○ Enable HTTPS-Only Mode in private windows only

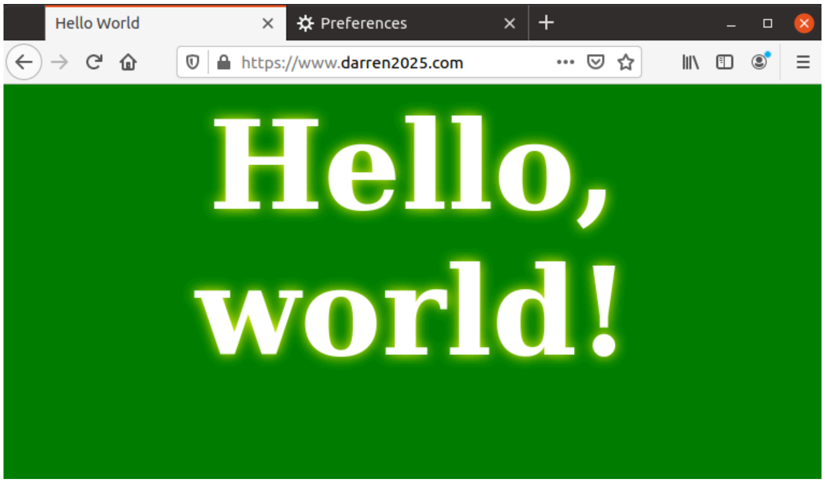
We can see that our CA’s ca.crt has been successfully imported into Firefox.



You have certificates on file that identify these certificate authorities

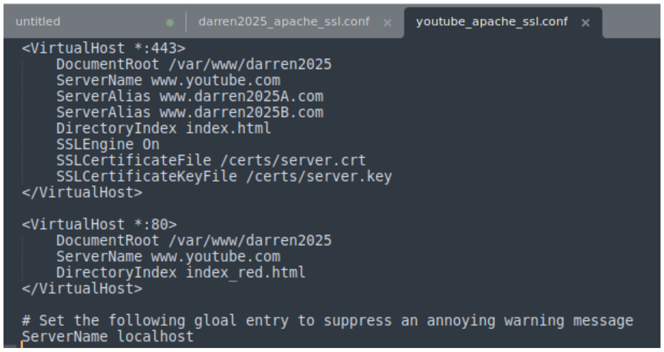
Certificate Name	Security Device
Microsoft RSA Root Certificate Authori…	Builtin Object Token
Model CA LTD.	
www.modelCA.com	Software Security Device
NetLock Kft.	
NetLock Arany (Class Gold) Főtanúsítv…	Builtin Object Token
Network Solutions L.L.C.	

Upon reloading www.darren2025.com, the error did not occur and we successfully accessed the website as our CA verified the identity of darren2025.com.



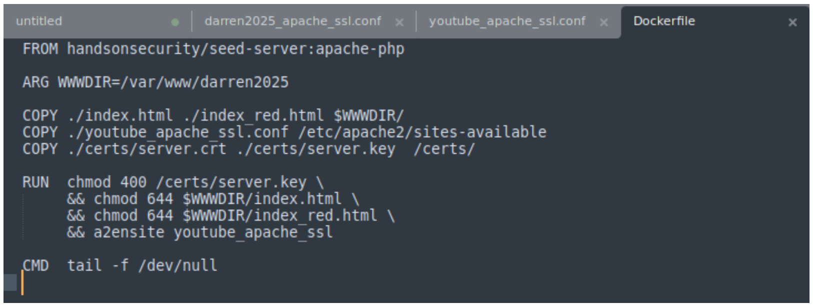
Task 5: Launching a Man-In-The-Middle Attack

My target website is www.youtube.com. I will be using the same Apache server to impersonate www.youtube.com with the previous configurations from Task 4, except the ServerName is www.youtube.com.



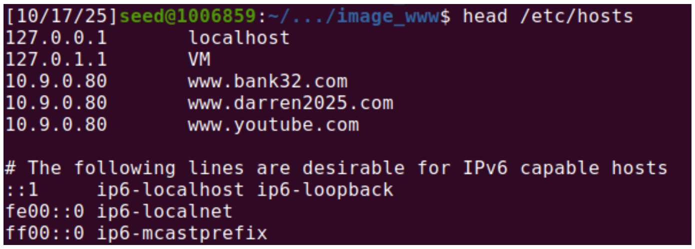
```
<VirtualHost *:443>
    DocumentRoot /var/www/darren2025
    ServerName www.youtube.com
    ServerAlias www.darren2025A.com
    ServerAlias www.darren2025B.com
    DirectoryIndex index.html
    SSLEngine On
    SSLCertificateFile /certs/server.crt
    SSLCertificateKeyFile /certs/server.key
</VirtualHost>

<VirtualHost *:80>
    DocumentRoot /var/www/darren2025
    ServerName www.youtube.com
    DirectoryIndex index_red.html
</VirtualHost>

# Set the following gloal entry to suppress an annoying warning message
ServerName localhost
```

I also edited the Dockerfile to use the configurations for www.youtube.com. Then, I “dcbuild” again to load my configurations on the container.



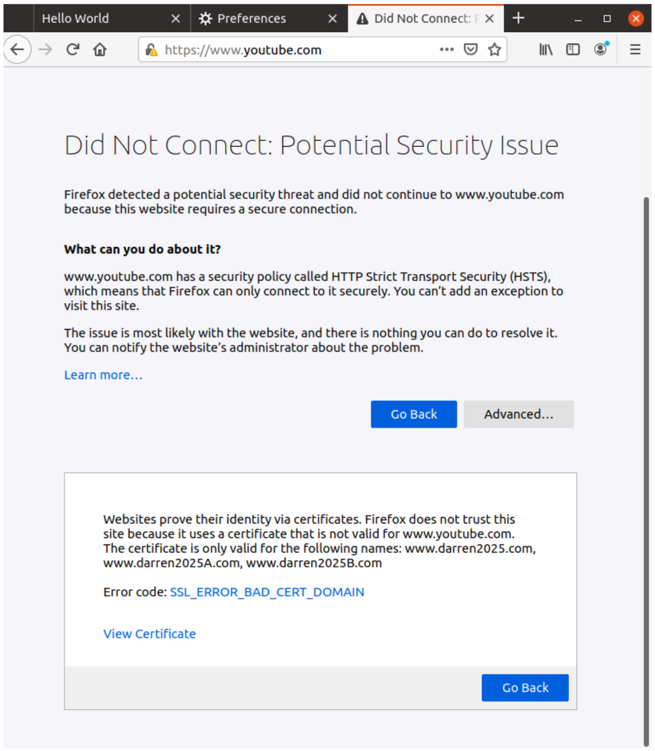
```
FROM handsonsecurity/seed-server:apache-php

ARG WWWDIR=/var/www/darren2025

COPY ./index.html ./index_red.html $WWWDIR/
COPY ./youtube_apache_ssl.conf /etc/apache2/sites-available
COPY ./certs/server.crt ./certs/server.key  /certs/

RUN  chmod 400 /certs/server.key \
     && chmod 644 $WWWDIR/index.html \
     && chmod 644 $WWWDIR/index_red.html \
     && a2ensite youtube_apache_ssl

CMD  tail -f /dev/null
```

After that, I started Apache and included the mapping of www.youtube.com to 10.9.0.80 in /etc/hosts to emulate the result of a DNS cache poisoning attack.



```
[10/17/25]seed@1006859:~/.../image_www$ head /etc/hosts
127.0.0.1       localhost
127.0.1.1       VM
10.9.0.80       www.bank32.com
10.9.0.80       www.darren2025.com
10.9.0.80       www.youtube.com

# The following lines are desirable for IPv6 capable hosts
::1     ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
```

With everything set up, we would face a potential security issue when accessing www.youtube.com. This is because according to server.crt, the CA only identifies www.darren2025.com, www.darren2025A.com, and www.darren2025B.com. As such, www.youtube.com is unable to be verified by the CA when we send the same server.crt to the CA for verification, hence facing the issue below.



Did Not Connect: Potential Security Issue

Firefox detected a potential security threat and did not continue to www.youtube.com because this website requires a secure connection.

What can you do about it?

www.youtube.com has a security policy called HTTP Strict Transport Security (HSTS), which means that Firefox can only connect to it securely. You can’t add an exception to visit this site.

The issue is most likely with the website, and there is nothing you can do to resolve it. You can notify the website’s administrator about the problem.

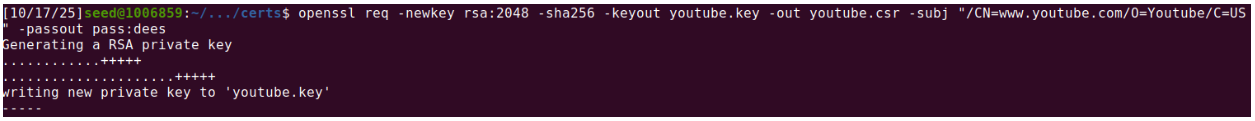
Learn more…

Websites prove their identity via certificates. Firefox does not trust this site because it uses a certificate that is not valid for www.youtube.com. The certificate is only valid for the following names: www.darren2025.com, www.darren2025A.com, www.darren2025B.com

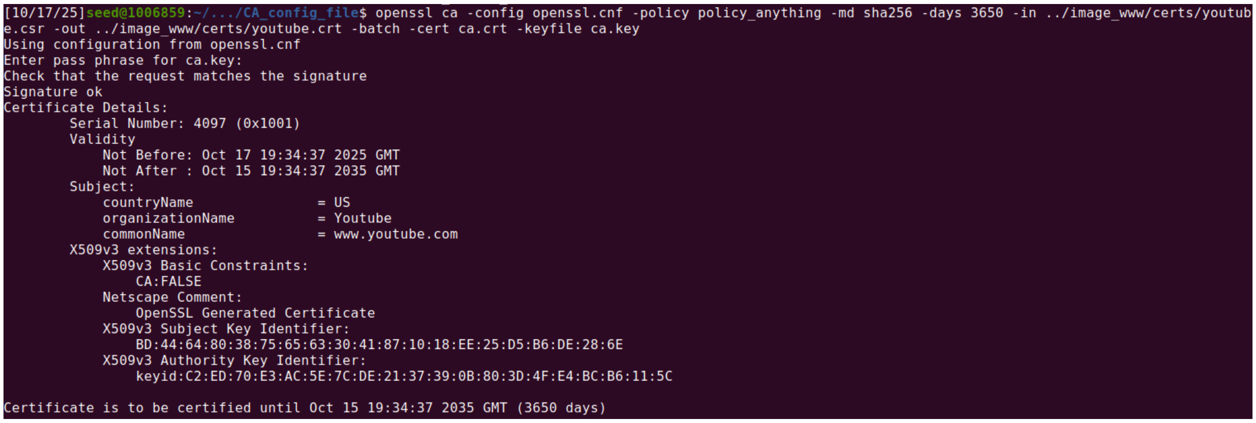
Error code: SSL_ERROR_BAD_CERT_DOMAIN

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

With a compromised CA, an attacker can generate any arbitrary certificate using the CA's private key. I will be creating a certificate for www.youtube.com as the attacker.

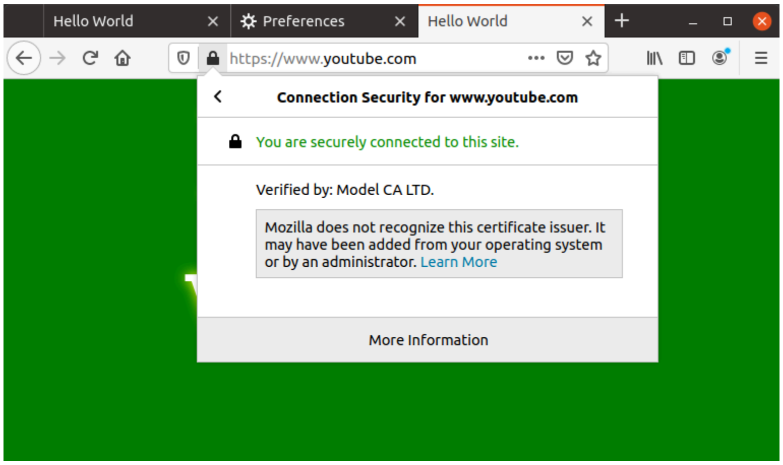


```
[10/17/25]seed@1006859:~/.../certs$ openssl req -newkey rsa:2048 -sha256 -keyout youtube.key -out youtube.csr -subj "/CN=www.youtube.com/O=Youtube/C=US
" -passout pass:dees
Generating a RSA private key
...........+++++
.....................+++++
writing new private key to 'youtube.key'
-----
```



```
[10/17/25]seed@1006859:~/.../CA_config_file$ openssl ca -config openssl.cnf -policy policy_anything -md sha256 -days 3650 -in ../image_www/certs/youtub
e.csr -out ../image_www/certs/youtube.crt -batch -cert ca.crt -keyfile ca.key
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: Oct 17 19:34:37 2025 GMT
            Not After : Oct 15 19:34:37 2035 GMT
        Subject:
            countryName               = US
            organizationName          = Youtube
            commonName                = www.youtube.com
        X509v3 extensions:
            X509v3 Basic Constraints: 
                CA:FALSE
            Netscape Comment: 
                OpenSSL Generated Certificate
            X509v3 Subject Key Identifier: 
                BD:44:64:80:38:75:65:63:30:41:87:10:18:EE:25:D5:B6:DE:28:6E
            X509v3 Authority Key Identifier: 
                keyid:C2:ED:70:E3:AC:5E:7C:DE:21:37:39:0B:80:3D:4F:E4:BC:B6:11:5C

Certificate is to be certified until Oct 15 19:34:37 2035 GMT (3650 days)
```

After that, I tried accessing www.youtube.com and there were no errors. This also shows that the root CA can verify the certificate of www.youtube.com, increasing the legitimacy of the website.



Connection Security for www.youtube.com

You are securely connected to this site.

Verified by: Model CA LTD.

Mozilla does not recognize this certificate issuer. It may have been added from your operating system or by an administrator. Learn More

More Information

This mimics a successful launch of the MITM attack.

The attacker would poison the victim's DNS cache, which will lead him to visit the attacker's malicious website on 10.9.0.80. With a compromised CA, the attacker can construct an arbitrary certificate that allows the malicious website to be verified by the CA. This will increase the legitimacy of the malicious website as the browser will state that we are securely connected to the site and we would not face any suspicious errors, just like the picture above.