Lab 1
Public-Key Infrastructure (PKI) Lab

Author：57119108 吴桐

Date：2022.8.9

Lab Tasks

Task 1: Becoming a Certificate Authority (CA)

证书颁发机构（CA）是发布数字证书的受信任实体。数字证书通过证书的命名主体来验证公钥的所有权。

在此任务中，我们需要创建一个根 CA，并为这个 CA 生成一个自签名证书。根 CA 的证书通常预装在操作系统、Web 浏览器和其他依赖于 PKI 的软件中。根 CA 的证书是无条件受信任的。

为了使用 OpenSSL 创建证书，我们必须拥有一个配置文件。针对配置文件的操作我们会在 Task 3 中涉及。

我们需要为 CA 生成一个自签名证书，将其作为完全可信的根证书。如图 1 所示，运行命令为 CA 生成自签名证书，该命令的输出存储在两个文件中：ca.key 和 ca.crt。其中，ca.key 包含 CA 的私钥，而 ca.crt 包含 CA 的公钥证书，加密口令为 dees。

```
root@495530af16d5:/# openssl req -x509 -newkey rsa:4096 -sha256 -days 3650 -keyout ca.key
-out ca.crt -subj "/CN=www.modelCA.com/O=Model CA LTD./C=CN" -passout pass:dees
Generating a RSA private key
.....................................................................................++++
.....................................................................++++
writing new private key to 'ca.key'
-----
```

图 1

使用 openssl 命令将 ca.key 和 ca.crt 文件转成可读形式并输出，结果如图 2-7 所示。

```
root@495530af16d5:/# openssl x509 -in ca.crt -text -noout
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number:
            35:f7:f3:e3:23:14:37:fe:cc:9a:e4:7c:c1:2a:cf:75:16:9d:c8:c3
        Signature Algorithm: sha256WithRSAEncryption
        Issuer: CN = www.modelCA.com, O = Model CA LTD., C = CN
        Validity
            Not Before: Aug  8 11:09:56 2022 GMT
            Not After : Aug  5 11:09:56 2032 GMT
        Subject: CN = www.modelCA.com, O = Model CA LTD., C = CN
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
                RSA Public-Key: (4096 bit)
3               Modulus:
                    00:fa:b2:bd:91:46:26:4c:a3:38:3d:19:71:cf:99:
                    36:87:22:dd:ed:47:96:a1:25:b8:34:64:18:c7:3f:
                    dc:41:cd:c6:ce:aa:ec:92:b2:49:df:74:fa:33:91:
                    6d:06:4a:31:f1:1a:13:05:0f:48:d4:ad:ef:34:17:
                    47:b6:02:f5:2d:5c:1e:bf:2b:e8:1c:11:68:46:07:
                    23:82:ed:d8:a0:4e:83:66:ed:7d:70:e5:a4:59:83:
                    52:8a:5c:b7:ac:3b:be:39:39:97:b4:f1:ce:b5:77:
                    54:3d:a1:2d:2a:a9:87:ba:26:48:60:ed:67:d8:21:
                    66:67:7f:50:af:35:93:1e:39:66:51:bc:d2:11:e9:
                    ca:81:ce:88:84:a9:aa:3b:2f:52:22:cc:43:a1:80:
                    07:31:61:f3:de:53:2b:b8:28:9c:94:0b:93:be:89:
                    8f:e7:ed:0b:66:d6:b1:c1:30:cd:b8:81:5a:36:15:
                    6a:b1:90:ed:d3:e4:c6:fb:bb:fa:64:19:79:9f:e6:
                    9a:a6:47:6c:31:7c:c6:11:77:df:d9:4a:8e:3d:c1:
                    0a:28:1f:f7:1f:c7:f5:1e:66:00:e5:0a:6f:dd:a0:
                    74:b5:50:66:cc:b8:b0:b0:5c:54:ae:92:b7:30:70:
                    25:f1:44:df:d3:2b:a2:58:89:9a:00:6b:28:f8:e7:
                    a1:66:43:36:54:54:61:33:b5:4b:35:a2:99:fc:10:
                    ab:67:bc:7f:6d:65:33:e2:93:56:6e:1b:43:fd:08:
                    24:76:f3:60:3b:36:12:f8:87:4a:53:06:0b:fc:68:
                    91:b9:b9:f9:06:a5:c2:ce:af:6e:13:0b:30:08:cc:
                    da:d7:13:7e:f3:87:51:8e:e3:20:56:68:2e:50:26:
                    2d:16:0f:94:b4:8e:db:ca:01:c8:2f:9f:f7:e2:29:
                    7d:9e:d7:da:ce:1c:7b:a2:f5:e1:8b:a3:c7:80:23:
                    ec:33:27:b6:f8:61:a4:47:63:83:8b:02:e3:aa:c4:
                    7b:70:4a:22:b7:1a:7e:d6:6b:e8:39:48:f1:6f:55:
                    b5:78:ef:b3:66:38:4e:4a:f8:6d:df:80:7d:5e:a9:
                    8b:63:38:cc:2e:05:4b:38:b9:88:2d:fa:39:59:95:
                    34:d8:15:bc:a5:83:9d:bd:82:36:9d:9b:29:78:da:
                    92:11:7e:eb:68:b2:a9:5a:7f:f8:f0:62:fe:8a:d7:
                    7b:5a:ae:7b:4d:af:4f:af:76:4c:15:3f:1f:78:6a:
                    53:c3:2f:12:68:a7:a1:fc:52:3d:c9:56:a6:8b:b2:
                    56:7d:28:fa:e1:dd:e1:82:ac:3d:7c:76:27:d3:4d:
                    ee:f6:43:ad:2c:ba:34:29:75:3a:61:28:96:cd:e3:
                    4b:5e:15
```

图 2

```
1               Exponent: 65537 (0x10001)
        X509v3 extensions:
            X509v3 Subject Key Identifier:
                CC:0D:1D:BD:BC:7C:A1:25:0A:F5:C1:78:F5:1C:25:7E:45:1F:61:A9
            X509v3 Authority Key Identifier:
                keyid:CC:0D:1D:BD:BC:7C:A1:25:0A:F5:C1:78:F5:1C:25:7E:45:1F:61:A9

            X509v3 Basic Constraints: critical
                CA:TRUE
    Signature Algorithm: sha256WithRSAEncryption
         c7:9f:66:a4:31:d9:74:d7:27:5c:e2:68:8f:78:4d:dc:8c:ba:
         56:c3:f1:93:4c:3e:aa:3b:dc:21:17:a9:a9:12:0c:af:a0:32:
         1d:32:37:04:40:8d:85:3c:26:0c:f1:7e:1b:09:f5:38:63:7c:
         0f:d0:a3:ae:b7:27:a4:52:33:cc:d3:ed:f8:a6:b3:2c:b3:95:
         0b:f1:d5:bc:dc:88:30:4d:98:0d:95:8e:d2:21:1b:4d:fe:d2:
         29:09:0e:f8:43:a0:00:17:56:1d:d0:58:1d:b4:6d:07:9c:e9:
         df:3b:31:6c:a4:ea:6a:5a:62:e0:13:06:c6:2a:b2:e7:2e:25:
         c7:2b:b4:ff:03:f0:7c:d6:e8:14:19:2d:15:75:fa:10:d8:ee:
         03:bf:12:70:22:43:bd:68:8f:c9:0c:0e:c9:a8:44:e0:b0:c9:
         61:4c:aa:00:cc:0d:70:1b:48:60:74:df:f0:0f:25:14:b0:4e:
         85:11:5c:59:53:ff:30:16:26:24:4e:b5:0d:b3:7f:dc:5e:10:
         9a:d2:37:fa:44:78:82:4a:80:53:c6:1f:e5:50:8e:6b:dc:4b:
         1d:e0:69:98:5c:6b:ce:83:dd:eb:da:b2:45:41:91:9a:32:70:
         ea:63:79:ed:6c:3a:d7:8b:19:85:f7:2d:39:de:6b:a6:4e:ed:
         3c:6c:aa:17:5a:ba:75:94:ce:3c:d1:4c:41:98:43:ca:9b:08:
         b1:5d:1e:15:f4:a6:7c:1f:86:09:24:f1:79:a7:23:ed:07:2c:
         4a:11:b1:c4:92:90:49:79:6d:11:9f:9b:5c:e4:fa:73:60:8d:
         be:ec:da:6e:76:8b:43:8c:75:c1:5d:79:73:cf:6e:9f:9c:7b:
         22:89:88:13:4b:94:5d:96:0f:04:48:a3:f0:63:68:e8:62:c7:
         e2:1c:63:c0:74:00:d5:d2:7b:a7:3d:02:73:2a:24:f1:ff:06:
         a7:0b:b8:4d:2b:a6:f4:b2:4a:e7:22:74:1f:92:1d:57:00:43:
         a6:5f:63:8d:65:e1:00:f6:ca:8d:03:1b:c0:fb:50:13:35:03:
         31:db:05:18:19:ac:84:ad:fa:04:6d:91:79:12:c0:0d:22:57:
         b0:bc:ac:ba:94:06:a6:ec:60:99:fa:51:ba:33:d4:9f:68:95:
         cb:70:7b:e0:45:08:48:ec:ab:0f:cf:cb:4c:6c:55:64:17:ad:
         47:1d:1e:e8:62:eb:16:70:b3:48:1e:6f:5e:9f:33:12:d5:b9:
         8f:f1:a6:e2:2b:04:75:79:19:01:23:da:e3:08:bb:fb:2e:f8:
         58:53:2b:79:5c:f5:38:f3:59:92:8e:e7:7f:64:a8:94:c3:82:
         27:64:67:45:1b:ae:a7:b1
```

图 3

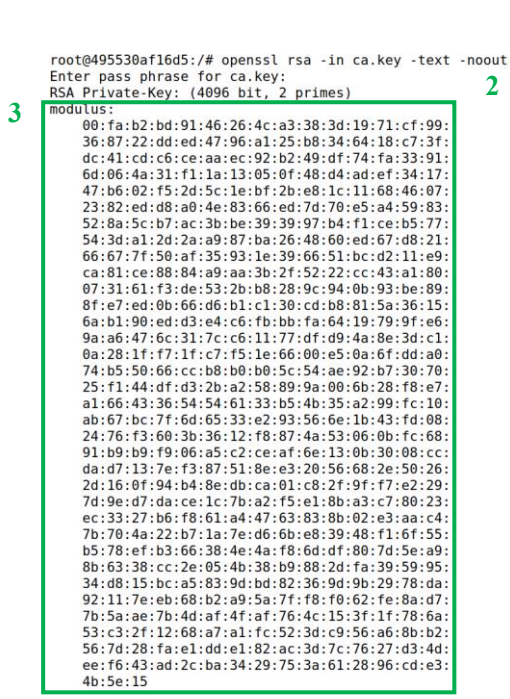
root@495530af16d5:/# openssl rsa -in ca.key -text -noout
Enter pass phrase for ca.key:
RSA Private-Key: (4096 bit, 2 primes)
modulus:
00:fa:b2:bd:91:46:26:4c:a3:38:3d:19:71:cf:99:
36:87:22:dd:ed:47:96:a1:25:b8:34:64:18:c7:3f:
dc:41:cd:c6:ce:aa:ec:92:b2:49:df:74:fa:33:91:
6d:06:4a:31:f1:1a:13:05:0f:48:d4:ad:ef:34:17:
47:b6:02:f5:2d:5c:1e:bf:2b:e8:1c:11:68:46:07:
23:82:ed:d8:a0:4e:83:66:ed:7d:70:e5:a4:59:83:
52:8a:5c:b7:ac:3b:be:39:39:97:b4:f1:ce:b5:77:
54:3d:a1:2d:2a:a9:87:ba:26:48:60:ed:67:d8:21:
66:67:7f:50:af:35:93:1e:39:66:51:bc:d2:11:e9:
ca:81:ce:88:84:a9:aa:3b:2f:52:22:cc:43:a1:80:
07:31:61:f3:de:53:2b:b8:28:9c:94:0b:93:be:89:
8f:e7:ed:0b:66:d6:b1:c1:30:cd:b8:81:5a:36:15:
6a:b1:90:ed:d3:e4:c6:fb:bb:fa:64:19:79:9f:e6:
9a:a6:47:6c:31:7c:c6:11:77:df:d9:4a:8e:3d:c1:
0a:28:1f:f7:1f:c7:f5:1e:66:00:e5:0a:6f:dd:a0:
74:b5:50:66:cc:b8:b0:b0:5c:54:ae:92:b7:30:70:
25:f1:44:df:d3:2b:a2:58:89:9a:00:6b:28:f8:e7:
a1:66:43:36:54:54:61:33:b5:4b:35:a2:99:fc:10:
ab:67:bc:7f:6d:65:33:e2:93:56:6e:1b:43:fd:08:
24:76:f3:60:3b:36:12:f8:87:4a:53:06:0b:fc:68:
91:b9:b9:f9:06:a5:c2:ce:af:6e:13:0b:30:08:cc:
da:d7:13:7e:f3:87:51:8e:e3:20:56:68:2e:50:26:
2d:16:0f:94:b4:8e:db:ca:01:c8:2f:9f:f7:e2:29:
7d:9e:d7:da:ce:1c:7b:a2:f5:e1:8b:a3:c7:80:23:
ec:33:27:b6:f8:61:a4:47:63:83:8b:02:e3:aa:c4:
7b:70:4a:22:b7:1a:7e:d6:6b:e8:39:48:f1:6f:55:
b5:78:ef:b3:66:38:4e:4a:f8:6d:df:80:7d:5e:a9:
8b:63:38:cc:2e:05:4b:38:b9:88:2d:fa:39:59:95:
34:d8:15:bc:a5:83:9d:bd:82:36:9d:9b:29:78:da:
92:11:7e:eb:68:b2:a9:5a:7f:f8:f0:62:fe:8a:d7:
7b:5a:ae:7b:4d:af:4f:af:76:4c:15:3f:1f:78:6a:
53:c3:2f:12:68:a7:a1:fc:52:3d:c9:56:a6:8b:b2:
56:7d:28:fa:e1:dd:e1:82:ac:3d:7c:76:27:d3:4d:
ee:f6:43:ad:2c:ba:34:29:75:3a:61:28:96:cd:e3:
4b:5e:15
publicExponent: 65537 (0x10001)
privateExponent:
00:b8:66:ed:45:84:81:fb:71:3b:96:c5:b6:8f:6b:
35:4e:a5:78:f4:65:ea:08:52:c4:71:d5:06:dc:d8:
be:5c:b0:ce:01:8a:7c:45:53:5b:3f:40:be:21:5c:
61:03:75:63:d2:e2:c9:06:b9:bd:6a:a3:1e:06:04:
f1:bc:91:e4:62:1f:16:86:ab:f7:29:5e:2c:83:06:
11:3a:e7:54:a6:65:da:05:c5:be:b5:62:55:47:7a:
6f:26:53:18:cf:59:8a:8c:71:2e:83:ec:31:8e:0d:
5e:2b:8e:f8:b8:1e:c7:35:47:9f:89:9c:56:fc:05:
65:5b:cf:ac:3b:94:7e:c8:86:94:1a:9f:77:2d:b1:
f8:41:0e:f8:20:0c:e4:a8:70:20:4c:74:4f:72:e5:
c7:64:ad:96:41:fc:8f:76:00:79:5b:f4:40:12:b1:
00:e7:29:d7:6f:27:90:cc:77:e7:7a:45:bb:63:78:
d1:68:17:20:b8:f1:d8:e3:a5:db:81:f1:b3:22:c8:
ef:0c:4c:10:eb:3d:ca:3b:63:1f:86:a7:6e:85:42:
0f:04:80:ae:b7:f3:ce:fb:33:30:89:48:ea:0b:ed:
50:ab:65:9e:98:0b:69:fb:ef:8e:62:72:8c:c8:d6:
55:32:68:e4:f2:61:70:23:0b:e3:f9:56:d9:66:6c:
8a:ac:47:27:98:c7:87:ea:39:17:16:dc:94:9a:d3:
22:ac:12:35:cd:cf:9d:30:8a:2f:92:98:b8:58:16:
e6:0b:13:91:ac:cb:fd:04:1a:11:5e:a8:0c:2a:e3:
62:4d:f8:60:d1:ff:17:1b:ac:e6:b8:cc:57:9d:e4:
de:0b:9e:bb:46:80:ac:c1:2f:de:4f:70:6b:30:66:
c0:88:e6:70:ce:8d:b3:4b:f1:43:61:66:c3:45:1a:
ae:58:ef:73:ad:54:4d:ef:22:05:b1:2e:c0:09:bd:
a1:e0:2b:bd:f9:c0:b8:98:be:4e:44:3d:a0:37:b5:
18:08:3b:20:e9:db:7f:a1:00:d8:43:02:5b:1c:dd:
6e:da:44:38:28:00:12:77:1c:ab:c1:d1:6a:f4:c6:
7b:b9:ea:0a:47:26:8f:ec:8b:1d:f1:a6:21:ed:8d:
d5:08:3a:73:30:d6:1a:0b:89:15:28:ae:8c:4d:d9:
0b:86:76:03:60:38:f5:ad:81:42:27:94:bf:fe:45:
a6:3f:3f:6c:f8:7b:f0:fb:4b:af:d1:5b:2f:56:29:
36:dd:76:80:c8:36:7b:94:c0:d2:7a:76:ba:8b:85:
2d:22:f8:9a:65:af:1d:a2:5e:02:ac:45:b2:1e:66:
d0:8e:02:67:bf:98:13:12:1a:b2:5c:35:49:11:e0:
a0:0c:41
1
2
3

图 4

图 5

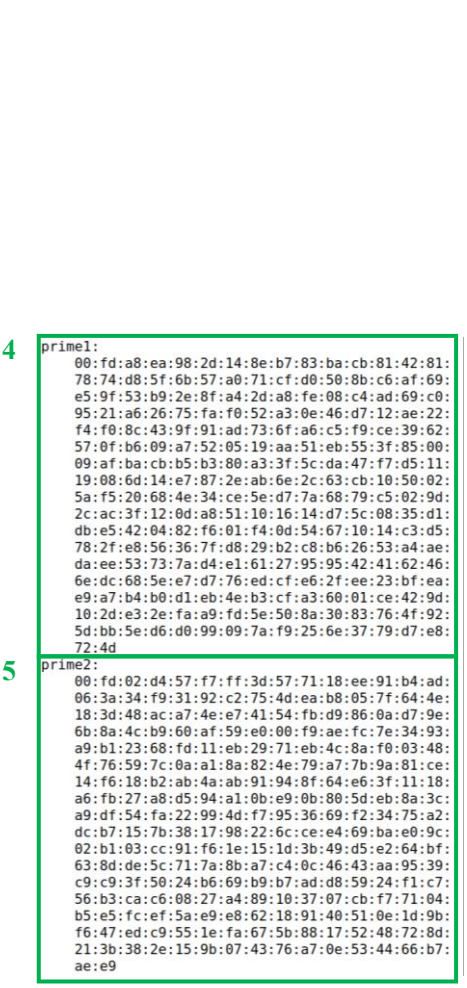
prime1:
00:fd:a8:ea:98:2d:14:8e:b7:83:ba:cb:81:42:81:
78:74:d8:5f:6b:57:a0:71:cf:d0:50:8b:c6:af:69:
e5:9f:53:b9:2e:8f:a4:2d:a8:fe:08:c4:ad:69:c0:
95:21:a6:26:75:fa:f0:52:a3:0e:46:d7:12:ae:22:
f4:f0:8c:43:9f:91:ad:73:6f:a6:c5:f9:ce:39:62:
57:0f:b6:09:a7:52:05:19:aa:51:eb:55:3f:85:00:
09:af:ba:cb:b5:b3:80:a3:3f:5c:da:47:f7:d5:11:
19:08:6d:14:e7:87:2e:ab:6e:2c:63:cb:10:50:02:
5a:f5:20:68:4e:34:ce:5e:d7:7a:68:79:c5:02:9d:
2c:ac:3f:12:0d:a8:51:10:16:14:d7:5c:08:35:d1:
db:e5:42:04:82:f6:01:f4:0d:54:67:10:14:c3:d5:
78:2f:e8:56:36:7f:d8:29:b2:c8:b6:26:53:a4:ae:
da:ee:53:73:7a:d4:e1:61:27:95:95:42:41:62:46:
6e:dc:68:5e:e7:d7:76:ed:cf:e6:2f:ee:23:bf:ea:
e9:a7:b4:b0:d1:eb:4e:b3:cf:a3:60:01:ce:42:9d:
10:2d:e3:2e:fa:a9:fd:5e:50:8a:30:83:76:4f:92:
5d:bb:5e:d6:d0:99:09:7a:f9:25:6e:37:79:d7:e8:
72:4d
prime2:
00:fd:02:d4:57:f7:ff:3d:57:71:18:ee:91:b4:ad:
06:3a:34:f9:31:92:c2:75:4d:ea:b8:05:7f:64:4e:
18:3d:48:ac:a7:4e:e7:41:54:fb:d9:86:0a:d7:9e:
6b:8a:4c:b9:60:af:59:e0:00:f9:ae:fc:7e:34:93:
a9:b1:23:68:fd:11:eb:29:71:eb:4c:8a:f0:03:48:
4f:76:59:7c:0a:a1:8a:82:4e:79:a7:7b:9a:81:ce:
14:f6:18:b2:ab:4a:ab:91:94:8f:64:e6:3f:11:18:
a6:fb:27:a8:d5:94:a1:0b:e9:0b:80:5d:eb:8a:3c:
a9:df:54:fa:22:99:4d:f7:95:36:69:f2:34:75:a2:
dc:b7:15:7b:38:17:98:22:6c:ce:e4:69:ba:e0:9c:
02:b1:03:cc:91:f6:1e:15:1d:3b:49:d5:e2:64:bf:
63:8d:de:5c:71:7a:8b:a7:c4:0c:46:43:aa:95:39:
c9:c9:3f:50:24:b6:69:b9:b7:ad:d8:59:24:f1:c7:
56:b3:ca:c6:08:27:a4:89:10:37:07:cb:f7:71:04:
b5:e5:fc:ef:5a:e9:e8:62:18:91:40:51:0e:1d:9b:
f6:47:ed:c9:55:1e:fa:67:5b:88:17:52:48:72:8d:
21:3b:38:2e:15:9b:07:43:76:a7:0e:53:44:66:b7:
ae:e9
4
5

图 6

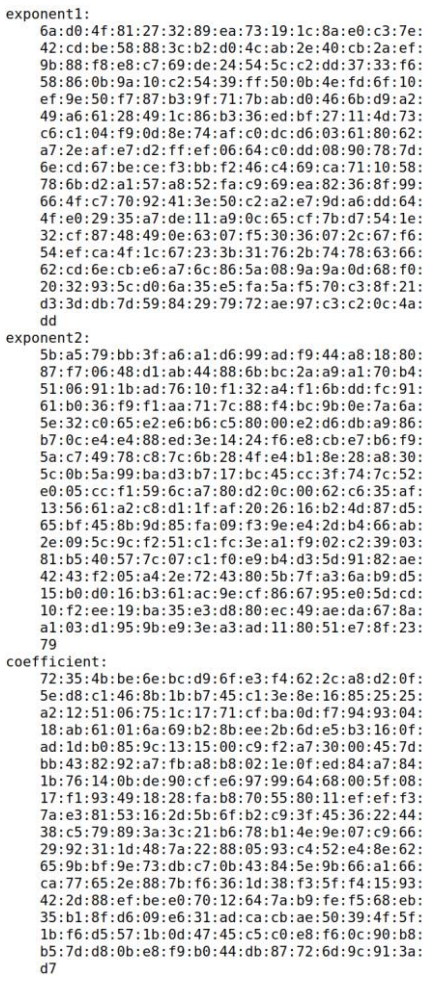
exponent1:
6a:d0:4f:81:27:32:89:ea:73:19:1c:8a:e0:c3:7e:
42:cd:be:58:88:3c:b2:d0:4c:ab:2e:40:cb:2a:ef:
9b:88:f8:e8:c7:69:de:24:54:5c:c2:dd:37:33:f6:
58:86:0b:9a:10:c2:54:39:ff:50:0b:4e:fd:6f:10:
ef:9e:50:f7:87:b3:9f:71:7b:ab:d0:46:6b:d9:a2:
49:a6:61:28:49:1c:86:b3:36:ed:bf:27:11:4d:73:
c6:c1:04:f9:0d:8e:74:af:c0:dc:d6:03:61:80:62:
a7:2e:af:e7:d2:ff:ef:06:64:c0:dd:08:90:78:7d:
6e:cd:67:be:ce:f3:bb:f2:46:c4:69:ca:71:10:58:
78:6b:d2:a1:57:a8:52:fa:c9:69:ea:82:36:8f:99:
66:4f:c7:70:92:41:3e:50:c2:a2:e7:9d:a6:dd:64:
4f:e0:29:35:a7:de:11:a9:0c:65:cf:7b:d7:54:1e:
32:cf:87:48:49:0e:63:07:f5:30:36:07:2c:67:f6:
54:ef:ca:4f:1c:67:23:3b:31:76:2b:74:78:63:66:
62:cd:6e:cb:e6:a7:6c:86:5a:08:9a:9a:0d:68:f0:
20:32:93:5c:d0:6a:35:e5:fa:5a:f5:70:c3:8f:21:
d3:3d:db:7d:59:84:29:79:72:ae:97:c3:c2:0c:4a:
dd
exponent2:
5b:a5:79:bb:3f:a6:a1:d6:99:ad:f9:44:a8:18:80:
87:f7:06:48:d1:ab:44:88:6b:bc:2a:a9:a1:70:b4:
51:06:91:1b:ad:76:10:f1:32:a4:f1:6b:dd:fc:91:
61:b0:36:f9:f1:aa:71:7c:88:f4:bc:9b:0e:7a:6a:
5e:32:c0:65:e2:e6:b6:c5:80:00:e2:d6:db:a9:86:
b7:0c:e4:e4:88:ed:3e:14:24:f6:e8:cb:e7:b6:f9:
5a:c7:49:78:c8:7c:6b:28:4f:e4:b1:8e:28:a8:30:
5c:0b:5a:99:ba:d3:b7:17:bc:45:cc:3f:74:7c:52:
e0:05:cc:f1:59:6c:a7:80:d2:0c:00:62:c6:35:af:
13:56:61:a2:c8:d1:1f:af:20:26:16:b2:4d:87:d5:
65:bf:45:8b:9d:85:fa:09:f3:9e:e4:2d:b4:66:ab:
2e:09:5c:9c:f2:51:c1:fc:3e:a1:f9:02:c2:39:03:
81:b5:40:57:7c:07:c1:f0:e9:b4:d3:5d:91:82:ae:
42:43:f2:05:a4:2e:72:43:80:5b:7f:a3:6a:b9:d5:
15:b0:d0:16:b3:61:ac:9e:cf:86:67:95:e0:5d:cd:
10:f2:ee:19:ba:35:e3:d8:80:ec:49:ae:da:67:8a:
a1:03:d1:95:9b:e9:3e:a3:ad:11:80:51:e7:8f:23:
79
coefficient:
72:35:4b:be:6e:bc:d9:6f:e3:f4:62:2c:a8:d2:0f:
5e:d8:c1:46:8b:1b:b7:45:c1:3e:8e:16:85:25:25:
a2:12:51:06:75:1c:17:71:cf:ba:0d:f7:94:93:04:
18:ab:61:01:6a:69:b2:8b:ee:2b:6d:e5:b3:16:0f:
ad:1d:b0:85:9c:13:15:00:c9:f2:a7:30:00:45:7d:
bb:43:82:92:a7:fb:a8:b8:02:1e:0f:ed:84:a7:84:
1b:76:14:0b:de:90:cf:e6:97:99:64:68:00:5f:08:
17:f1:93:49:18:28:fa:b8:70:55:80:11:ef:ef:f3:
7a:e3:81:53:16:2d:5b:6f:b2:c9:3f:45:36:22:44:
38:c5:79:89:3a:3c:21:b6:78:b1:4e:9e:07:c9:66:
29:92:31:1d:48:7a:22:88:05:93:c4:52:e4:8e:62:
65:9b:bf:9e:73:db:c7:0b:43:84:5e:9b:66:a1:66:
ca:77:65:2e:88:7b:f6:36:1d:38:f3:5f:f4:15:93:
42:2d:88:ef:be:e0:70:12:64:7a:b9:fe:f5:68:eb:
35:b1:8f:d6:09:e6:31:ad:ca:cb:ae:50:39:4f:5f:
1b:f6:d5:57:1b:0d:47:45:c5:c0:e8:f6:0c:90:b8:
b5:7d:d8:0b:e8:f9:b0:44:db:87:72:6d:9c:91:3a:
d7

图 7

Questions:

1、证书的哪一部分表明这是 CA 的证书？

答：如图 2 中红色方框所示，数字证书中的 Subject 域提供该证书拥有者的信息。

2、证书的哪一部分表明这是自签名证书？

答：如图 2 中蓝色方框所示，数字证书中的 Issuer 域提供签发该证书的认证机构的信息。可以看出该证书由 Modal CA LTD.签发，即自签名证书。

3、在 RSA 算法中，有一个公共指数 e，一个私有指数 d，一个模数 n 和两个秘密数 p 和 q，其中有 n＝pq。请在证书和密钥文件中标识这些元素的值。

答：如图 2-6 中绿色方框所示，左上角的数字标识分别对应：1→公共指数 e，2→私有指数 d，3→模数 n，4→秘密数 p，5→秘密数 q。

Task 2: Generating a Certificate Request for Your Web Server

在 Task 2 和 Task 3 中，我们将利用 Task 1 中创建的 CA 为自己的 Web 服务器（www.cocot2022.com）签发一个公钥证书。首先，我们需要生成证书签名请求（CSR），其中包含服务器的公钥和身份信息。CSR 将发送给 CA，CA 验证请求中的身份信息后生成证书。

注意，生成 CSR 的命令中不包含-x509 选项，否则该命令将直接生成一个自签名证书。此外，我们使用-addext 选项在 SubjectAltName 扩展域中为证书添加多个名称。

```
root@495530af16d5:/# openssl req -newkey rsa:2048 -sha256 -keyout server.key -out
server.csr -subj "/CN=www.cocot2022.com/O=Cocot2022 Inc./C=CN" -passout pass:dees
-addext "subjectAltName = DNS:www.cocot2022.com,DNS:www.cocot2022A.com,DNS:www.coc
ot2022B.com"
Generating a RSA private key
.........................+++++
.....................................+++++
writing new private key to 'server.key'
-----
```

图 8

使用 openssl 命令将 server.key 和 server.csr 文件转成可读形式并输出，结果如图 9-11 所示。

```
root@495530af16d5:/# openssl req -in server.csr -text -noout
Certificate Request:
    Data:
        Version: 1 (0x0)
        Subject: CN = www.cocot2022.com, O = Cocot2022 Inc., C = CN
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
                RSA Public-Key: (2048 bit)
                Modulus:
                    00:bd:1a:01:6b:c8:69:b5:b8:57:81:b1:54:f4:2d:
                    20:c4:fc:c5:ce:d5:8a:a5:fe:ad:e1:0d:63:f4:f2:
                    61:e3:25:a3:a2:a5:f0:5c:55:cf:b5:d0:18:35:e6:
                    4d:d4:2d:d6:8d:c2:63:81:12:48:f2:2d:7b:34:ff:
                    e7:e1:6d:e6:21:45:94:be:13:99:9e:d7:d0:3c:4c:
                    cd:c8:ca:f1:1a:0a:22:32:3f:6c:79:2e:68:b2:79:
                    fc:85:fe:8e:9b:51:44:ee:7e:cd:4e:6a:ab:90:bf:
                    0e:2b:74:31:8d:4d:d2:f3:e8:7c:80:00:ba:ba:4f:
                    89:a4:76:cb:8b:d7:06:68:ae:94:2c:4a:72:e3:21:
                    a1:f2:29:29:1c:d3:8a:03:9c:07:4f:6c:1b:b4:74:
                    b9:86:a8:e7:60:6b:91:06:d2:9b:d3:d0:9a:13:b3:
                    13:24:3d:ce:f9:08:cd:45:7b:20:cc:df:0e:8a:30:
                    3c:2e:5c:28:6e:fc:92:95:22:65:a2:1f:d3:ce:b6:
                    b0:e4:02:42:6a:8e:d1:b7:38:8e:80:c4:29:d9:c1:
                    df:56:af:96:9b:41:dc:5e:59:68:69:28:52:3d:35:
                    f1:9f:d0:f9:50:80:20:8d:e5:c0:00:68:e2:a2:ed:
                    09:08:87:70:c1:9d:4c:9a:4f:b6:90:1f:0c:52:cf:
                    c1:35
                Exponent: 65537 (0x10001)
        Attributes:
        Requested Extensions:
            X509v3 Subject Alternative Name:
                DNS:www.cocot2022.com, DNS:www.cocot2022A.com, DNS:www.cocot2022B.com
    Signature Algorithm: sha256WithRSAEncryption
         b1:ef:25:56:ac:1c:bf:46:a0:ad:84:d3:a2:8f:8a:77:a3:61:
         c8:56:a5:8d:b3:bc:3f:d2:d8:df:72:3d:dc:fd:25:7a:b7:eb:
         5f:da:36:45:8e:77:53:e1:55:80:28:1c:e3:dd:18:e6:05:f6:
         67:02:8d:72:83:16:fa:93:1f:93:90:2d:4e:56:fd:1f:30:46:
         49:5d:52:fb:a1:5b:a2:99:cb:f6:b8:b3:d4:8b:c6:bf:fa:6f:
         08:ae:ae:51:66:85:e4:21:53:94:4e:7b:d4:66:ce:f8:81:cc:
         01:a2:2b:ab:d2:8b:c9:2c:26:8a:25:59:eb:24:59:2b:60:7a:
         1d:8a:38:0a:5a:30:07:9e:cf:19:45:4c:54:80:34:4e:fa:ba:
         79:7d:f4:a1:c9:69:83:2f:f4:1c:78:9f:04:aa:c9:a2:32:35:
         d1:b9:a3:28:90:b1:1d:90:70:c4:b2:38:5b:41:ce:56:96:95:
         59:64:5d:05:81:72:1b:2c:24:e4:02:49:54:89:cf:f3:0b:b3:
         5d:7e:cc:0a:0a:a5:bf:66:9c:66:98:e6:b7:f6:f5:3e:bf:93:
         78:ea:a2:3e:13:0a:64:5d:b6:ee:07:92:7c:d6:03:a4:78:ed:
         f5:bc:b7:dd:46:e0:84:5f:62:ee:4f:aa:55:50:4e:55:58:a8:
         a4:5d:b3:8a
```

图 9

```
root@495530af16d5:/# openssl rsa -in server.key -text -noout
Enter pass phrase for server.key:
RSA Private-Key: (2048 bit, 2 primes)
modulus:
    00:bd:1a:01:6b:c8:69:b5:b8:57:81:b1:54:f4:2d:
    20:c4:fc:c5:ce:d5:8a:a5:fe:ad:e1:0d:63:f4:f2:
    61:e3:25:a3:a2:a5:f0:5c:55:cf:b5:d0:18:35:e6:
    4d:d4:2d:d6:8d:c2:63:81:12:48:f2:2d:7b:34:ff:
    e7:e1:6d:e6:21:45:94:be:13:99:9e:d7:d0:3c:4c:
    cd:c8:ca:f1:1a:0a:22:32:3f:6c:79:2e:68:b2:79:
    fc:85:fe:8e:9b:51:44:ee:7e:cd:4e:6a:ab:90:bf:
    0e:2b:74:31:8d:4d:d2:f3:e8:7c:80:00:ba:ba:4f:
    89:a4:76:cb:8b:d7:06:68:ae:94:2c:4a:72:e3:21:
    a1:f2:29:29:1c:d3:8a:03:9c:07:4f:6c:1b:b4:74:
    b9:86:a8:e7:60:6b:91:06:d2:9b:d3:d0:9a:13:b3:
    13:24:3d:ce:f9:08:cd:45:7b:20:cc:df:0e:8a:30:
    3c:2e:5c:28:6e:fc:92:95:22:65:a2:1f:d3:ce:b6:
    b0:e4:02:42:6a:8e:d1:b7:38:8e:80:c4:29:d9:c1:
    df:56:af:96:9b:41:dc:5e:59:68:69:28:52:3d:35:
    f1:9f:d0:f9:50:80:20:8d:e5:c0:00:68:e2:a2:ed:
    09:08:87:70:c1:9d:4c:9a:4f:b6:90:1f:0c:52:cf:
    c1:35
publicExponent: 65537 (0x10001)
privateExponent:
    32:bd:74:9e:28:23:7d:38:1e:7a:d5:4b:57:4d:78:
    9a:82:bc:cc:f7:ed:81:06:ea:3f:15:e9:b9:43:04:
    52:ff:b9:7d:9e:0a:f5:ab:ac:d5:3b:26:13:e8:5c:
    ff:ce:e0:e0:1e:c5:b6:4f:62:b5:60:d0:cb:72:d3:
    14:60:f9:bb:f9:c0:a4:93:fb:c7:6c:5f:1a:9e:fd:
    24:a0:97:bb:05:f7:11:06:85:a3:aa:79:3b:52:92:
    18:9e:2a:43:b0:b5:cf:4c:c4:e8:02:a5:50:6c:83:
    6f:d3:78:ac:52:b2:84:f1:df:de:34:d3:e5:c4:0b:
    c9:ce:7a:65:66:67:f8:d4:50:7b:fd:a6:a0:92:c9:
    d8:18:7a:4d:45:fd:a7:27:61:16:5d:98:97:85:dd:
    1a:7e:a7:a4:18:62:b9:0b:2d:24:24:69:48:01:55:
    15:52:1d:e6:40:be:ad:13:06:e8:99:71:98:ad:0a:
    47:b6:d4:29:65:0b:e0:47:7e:50:87:8c:bd:d6:6f:
    6b:ca:79:dd:73:ec:42:73:2a:75:f4:75:dc:20:97:
    01:17:f4:95:28:15:3f:7b:20:1c:99:60:88:06:33:
    0c:6f:03:f9:2a:52:57:a0:2d:75:88:bf:50:ca:18:
    23:e0:90:9d:5b:d0:70:15:6c:b1:e0:61:fa:c0:79:
    a1
```

图 10

```
prime1:
    00:f4:d2:9a:8d:7a:3a:86:a7:1e:c3:79:66:60:f8:
    b2:1c:b4:24:71:b6:50:5b:ac:cf:6a:ec:3f:e9:0d:
    58:0e:01:0c:7a:92:b1:c0:04:12:8e:1d:77:d0:ea:
    bc:80:0a:2d:da:a1:b9:65:ba:8a:79:3f:bd:2b:f2:
    c4:e0:d3:5e:38:14:de:06:23:b0:f5:2a:f9:21:8c:
    6f:33:47:7b:57:81:ae:52:4b:4a:3c:f9:df:6d:7b:
    34:e3:5d:dd:ba:8d:d0:99:cc:e5:41:49:eb:32:14:
    c7:e7:3d:5f:84:30:78:6f:b9:51:11:f9:6f:2f:c1:
    58:40:1e:68:e8:a4:1a:b7:7d
prime2:
    00:c5:bc:27:8d:81:92:81:3a:9d:5a:54:9e:d5:33:
    a9:25:1e:4a:6e:3e:dc:13:e4:4e:3c:fc:32:e4:e0:
    dc:e3:c0:f4:2c:22:53:93:49:86:61:e2:53:8d:78:
    49:6f:fb:89:17:cf:45:14:5e:de:65:27:92:98:5b:
    f1:8c:8d:55:d1:65:29:23:9b:79:1f:5c:ef:bb:93:
    37:ec:b2:df:74:5d:4f:87:05:ac:dc:f0:d1:2b:3e:
    44:f9:a0:a2:24:40:00:c6:bb:25:98:50:e8:f0:04:
    5a:df:3e:8a:e0:bb:6d:95:3f:89:b1:39:67:a9:b2:
    42:61:72:86:8d:44:c7:0e:19
exponent1:
    3a:4d:59:1e:ee:07:b7:ff:5e:75:20:98:ff:e8:d9:
    c9:ba:20:9f:af:d2:0d:32:e7:26:48:62:a3:e6:58:
    9c:e5:25:0a:9f:9a:92:e6:a5:60:90:a5:f2:eb:a3:
    be:3e:2e:53:4d:86:30:32:af:3d:56:af:7f:22:ce:
    3d:d8:38:2c:d4:d3:56:d4:f3:14:3f:8a:9a:b1:ad:
    dd:a1:5d:3f:26:93:d3:e7:38:23:b3:41:c0:f9:c1:
    4b:90:13:f8:94:43:24:0d:46:5f:38:f2:38:b7:f8:
    2f:1a:4f:7f:d4:67:29:fc:10:d1:5d:fc:5b:7c:08:
    bf:24:3e:c9:80:23:18:a5
exponent2:
    3d:c2:d5:3f:d7:b6:e0:f2:63:ef:b0:fa:a3:71:2d:
    65:d0:9e:42:ed:13:64:8b:2c:fc:d3:71:3a:18:1a:
    26:71:40:53:00:ad:c6:15:73:09:e3:dd:61:14:af:
    2b:71:0f:93:06:44:77:66:62:64:8a:05:b9:dd:0e:
    07:6e:a5:dd:6f:91:77:f3:b9:d3:57:fd:f1:42:bc:
    77:0c:2f:cb:72:d1:c2:44:bd:87:8d:18:68:3f:5c:
    df:f3:92:71:6c:24:51:ee:66:81:ba:d3:e7:14:15:
    26:21:ac:d3:20:f7:cb:64:3b:27:c5:dc:61:1e:b2:
    a9:29:52:69:91:8f:f3:11
coefficient:
    22:31:3e:4b:13:3e:06:86:75:7f:b4:cd:4d:c0:74:
    db:7a:02:1c:18:f6:8f:6b:f1:42:3f:22:a4:0e:09:
    e8:4d:c1:29:6e:ba:a1:7c:81:7b:43:3b:c0:44:5c:
    a0:60:c6:ed:13:1d:e8:2e:54:6a:31:a8:3f:58:a2:
    04:b8:32:c1:a6:b3:ce:3c:70:b2:76:2d:13:ac:c0:
    67:34:a8:43:13:b6:99:be:5b:42:7a:68:b6:3e:0a:
    c1:e5:16:23:76:77:93:e8:1e:5f:d4:91:cf:34:44:
    60:97:d0:b3:f9:2f:ac:a6:be:34:d3:a0:1d:c2:2e:
    a0:6f:6b:c8:f0:be:59:1f
```

图 11

Task 3: Generating a Certificate for your server

在此任务中，我们将使用 Task 1 中创建的根 CA 为 CSR 文件签名，生成证书。

首先，我们需要从/usr/lib/ssl 文件夹中拷贝 openssl.cnf 文件为 myCA_openssl.cnf 文件，如图 12 所示。

```
root@495530af16d5:/# cd /usr/lib/ssl
root@495530af16d5:/usr/lib/ssl# ls
certs  misc  openssl.cnf  private
root@495530af16d5:/usr/lib/ssl# cp openssl.cnf  /myCA_openssl.cnf
```

图 12

我们需要使用该配置文件中定义的 policy_anything 策略。该策略并不是默认策略。默认策略具有更多限制，强制要求请求中的部分域信息与 CA 证书中的域信息匹配。而 policy_anything 策略则不强制任何匹配规则。

出于安全考虑，openssl.cnf 中的默认设置不允许 openssl ca 命令将扩展域从请求拷贝到最终证书。为了允许拷贝，我们需要取消以下内容的注释：

```
# Extension copying option: use with caution.
copy_extensions = copy
```

图 13

由于 docker 中无法使用 vim 直接修改文件，我们使用 cat 命令输出 myCA_openssl.cnf 文件的内容，将其复制到主机上建立的同名文件中，在主机上修改文件。之后使用 docker cp 命令在本地和容器之间传输数据，如图 14 所示。

```
[08/08/22]seed@VM:~/.../Labsetup$ ls
docker-compose.yml  image_www  myCA_openssl.cnf  volumes
[08/08/22]seed@VM:~/.../Labsetup$ docker cp myCA_openssl.cnf 495530af16d5:/myCA_openssl.cnf
```

图 14

完成 OpenSSL 配置并使用 openssl ca 命令对 CSR 生成 CA 签名的数字证书，如图 15 所示。

```
root@495530af16d5:/# mkdir -p ./demoCA/newcerts
root@495530af16d5:/# touch ./demoCA/index.txt
root@495530af16d5:/# touch ./demoCA/serial
root@495530af16d5:/# echo '01' > ./demoCA/serial
root@495530af16d5:/# openssl ca -config myCA_openssl.cnf -policy policy_anything -md sha256
 -days 3650 -in server.csr -out server.crt -batch -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: 1 (0x1)
        Validity
            Not Before: Aug  8 11:38:35 2022 GMT
            Not After : Aug  5 11:38:35 2032 GMT
        Subject:
            countryName               = CN
            organizationName          = Cocot2022 Inc.
            commonName                = www.cocot2022.com
        X509v3 extensions:
            X509v3 Basic Constraints:
                CA:FALSE
            Netscape Comment:
                OpenSSL Generated Certificate
            X509v3 Subject Key Identifier:
                6D:3F:AE:AB:D8:B3:75:F5:B9:E3:3E:F2:34:2E:D6:E7:5C:87:39:AD
            X509v3 Authority Key Identifier:
                keyid:CC:0D:1D:BD:BC:7C:A1:25:0A:F5:C1:78:F5:1C:25:7E:45:1F:61:A9

            X509v3 Subject Alternative Name:
                DNS:www.cocot2022.com, DNS:www.cocot2022A.com, DNS:www.cocot2022B.com
Certificate is to be certified until Aug  5 11:38:35 2032 GMT (3650 days)

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

图 15

使用 openssl 命令将 server.crt 文件转成可读形式并输出，结果如图 16-17 所示。可以看到，证书中包含了主体的别称。

```
root@495530af16d5:/# openssl x509 -in server.crt -text -noout
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number: 1 (0x1)
        Signature Algorithm: sha256WithRSAEncryption
        Issuer: CN = www.modelCA.com, O = Model CA LTD., C = CN
        Validity
            Not Before: Aug  8 11:38:35 2022 GMT
            Not After : Aug  5 11:38:35 2032 GMT
        Subject: C = CN, O = Cocot2022 Inc., CN = www.cocot2022.com
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
                RSA Public-Key: (2048 bit)
                Modulus:
                    00:bd:1a:01:6b:c8:69:b5:b8:57:81:b1:54:f4:2d:
                    20:c4:fc:c5:ce:d5:8a:a5:fe:ad:e1:0d:63:f4:f2:
                    61:e3:25:a3:a2:a5:f0:5c:55:cf:b5:d0:18:35:e6:
                    4d:d4:2d:d6:8d:c2:63:81:12:48:f2:2d:7b:34:ff:
                    e7:e1:6d:e6:21:45:94:be:13:99:9e:d7:d0:3c:4c:
                    cd:c8:ca:f1:1a:0a:22:32:3f:6c:79:2e:68:b2:79:
                    fc:85:fe:8e:9b:51:44:ee:7e:cd:4e:6a:ab:90:bf:
                    0e:2b:74:31:8d:4d:d2:f3:e8:7c:80:00:ba:ba:4f:
                    89:a4:76:cb:8b:d7:06:68:ae:94:2c:4a:72:e3:21:
                    a1:f2:29:29:1c:d3:8a:03:9c:07:4f:6c:1b:b4:74:
                    b9:86:a8:e7:60:6b:91:06:d2:9b:d3:d0:9a:13:b3:
                    13:24:3d:ce:f9:08:cd:45:7b:20:cc:df:0e:8a:30:
                    3c:2e:5c:28:6e:fc:92:95:22:65:a2:1f:d3:ce:b6:
                    b0:e4:02:42:6a:8e:d1:b7:38:8e:80:c4:29:d9:c1:
                    df:56:af:96:9b:41:dc:5e:59:68:69:28:52:3d:35:
                    f1:9f:d0:f9:50:80:20:8d:e5:c0:00:68:e2:a2:ed:
                    09:08:87:70:c1:9d:4c:9a:4f:b6:90:1f:0c:52:cf:
                    c1:35
                Exponent: 65537 (0x10001)
        X509v3 extensions:
            X509v3 Basic Constraints:
                CA:FALSE
            Netscape Comment:
                OpenSSL Generated Certificate
            X509v3 Subject Key Identifier:
                6D:3F:AE:AB:D8:B3:75:F5:B9:E3:3E:F2:34:2E:D6:E7:5C:87:39:AD
            X509v3 Authority Key Identifier:
                keyid:CC:0D:1D:BD:BC:7C:A1:25:0A:F5:C1:78:F5:1C:25:7E:45:1F:61:A9

            X509v3 Subject Alternative Name:
                DNS:www.cocot2022.com, DNS:www.cocot2022A.com, DNS:www.cocot2022B.com
```

图 16

```
    Signature Algorithm: sha256WithRSAEncryption
         c3:a3:ec:73:44:31:e5:59:12:03:c4:ed:16:c7:33:3e:f0:5e:
         35:6e:fb:76:50:cc:06:b7:48:e0:45:72:c0:b5:b1:8b:00:69:
         cd:8f:83:ad:ff:15:62:00:6b:7d:6f:ed:59:24:ac:56:da:4c:
         c1:a6:27:94:ec:d9:8f:d0:e0:86:24:bb:68:6f:29:e6:1d:04:
         9a:ba:cc:04:39:fd:ac:7c:c5:63:fb:f5:f5:b2:7b:24:33:c0:
         26:f8:38:61:72:e4:2a:e4:d2:26:8f:71:c6:4f:da:b9:aa:b6:
         25:c7:b0:eb:2c:1b:8d:c2:c9:86:f1:70:fd:2c:e1:08:10:21:
         01:fb:80:55:72:65:2c:65:74:e4:9d:c7:57:87:6c:b9:03:0c:
         de:86:fb:3b:93:e7:d2:43:d9:19:db:cb:6f:08:af:0e:85:f5:
         1e:59:87:d5:cd:a7:1a:54:ab:ca:d5:37:de:1a:df:7d:3f:ae:
         bd:78:d7:b4:02:29:19:29:8c:fd:ee:7c:1d:87:5d:a2:a6:20:
         96:12:f7:68:22:92:d3:47:f4:94:21:0d:f0:cb:d8:d4:8c:c1:
         75:4b:7f:f2:75:8f:d2:84:bb:2c:be:bc:5c:8c:1b:2c:f7:a6:
         4c:15:cb:9b:77:d3:d3:a4:f9:66:38:8b:cc:5c:76:b2:65:9e:
         67:04:ea:f4:7a:e1:3c:c0:55:a2:2c:38:f9:21:22:cc:53:f8:
         05:e1:20:3b:e2:f1:66:fd:a1:0e:d0:29:0c:d7:dc:32:b7:e1:
         b4:0d:12:a4:76:b9:ce:16:dd:36:ef:7e:0d:cc:45:f0:7e:38:
         04:6a:7a:14:f4:f1:bb:12:be:8c:3f:f7:15:80:b8:5b:bb:18:
         b3:10:99:ed:7e:0f:e6:66:51:bb:be:14:fb:b7:9c:8e:94:87:
         6a:81:2f:43:e0:9a:5c:36:b5:e6:91:87:f9:c2:f9:5f:bd:aa:
         e0:32:4a:90:01:50:6b:a5:b8:25:20:79:b9:1a:42:5a:1e:99:
         90:b3:b3:a9:8d:f3:b1:a6:6e:d4:0a:a6:c3:94:c1:02:1f:2f:
         4d:aa:ad:63:39:76:09:16:55:dd:fe:42:19:94:c7:9c:55:bd:
         b1:67:3b:44:f2:c0:b7:e3:af:ae:0e:e5:74:20:58:91:2c:30:
         0e:38:dd:4a:44:a9:32:88:da:57:32:39:0c:04:1a:22:ec:56:
         c3:f9:9f:a6:eb:75:1e:70:e9:a3:13:60:e4:15:e0:48:ba:a9:
         a7:4f:87:a3:5d:63:6d:e8:09:8b:10:70:28:8a:c7:c4:4b:f3:
         5b:52:13:00:b1:5a:d7:8b:78:8f:f3:d8:9e:b7:68:4b:8c:38:
         c4:36:83:a1:11:f1:15:ee
```

图 17

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

在此任务中，我们将建立一个基于 Apache 的 HTTPS 网站，并利用公钥证书来保护浏览过程。要创建 HTTPS 网站，我们只需要配置 Apache 服务器，给出获取数字证书和私钥的

位置。图 18 展示的是 www.cocot2022.com 的 VirtualHost 文件。其中 DocumentRoot 指定网站文件的保存位置，ServerName 指定网站的 URL，公钥证书和私钥的文件名分别为 server.crt 和 server.key。

```
<VirtualHost *:443>
    DocumentRoot /var/www/html
    ServerName www.cocot2022.com
    ServerAlias www.cocot2022A.com
    ServerAlias www.cocot2022B.com
    DirectoryIndex index.html
    SSLEngine On
    SSLCertificateFile /server.crt
    SSLCertificateKeyFile /server.key
</VirtualHost>
```

图 18

如图 19 所示，将包含 VirtualHost 条目的 cocot2022_apache_ssl.conf 文件复制到 /etc/apache2/sites-available 目录下。

```
root@495530af16d5:/# cd volumes/
root@495530af16d5:/volumes# ls
README.md  cocot2022_apache_ssl.conf
root@495530af16d5:/volumes# cp cocot2022_apache_ssl.conf /etc/apache2/sites-available
root@495530af16d5:/volumes# cd /etc/apache2/sites-available
root@495530af16d5:/etc/apache2/sites-available# ls
000-default.conf  bank32_apache_ssl.conf  cocot2022_apache_ssl.conf  default-ssl.conf
```

图 19

如图 20 所示，在 docker 中输入相关命令启动服务器。在 Apache 启动时，需要为每个 HTTPS 站点加载私钥。私钥在创建时被一个简单的口令加密，所以 Apache 会提示我们输入口令进行私钥解密。这里我们输入生成 server.key 时规定的口令：dees。

```
root@495530af16d5:/etc/apache2/sites-available# apachectl configtest
Syntax OK
root@495530af16d5:/etc/apache2/sites-available# 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
root@495530af16d5:/etc/apache2/sites-available# a2ensite cocot2022_apache_ssl
Site cocot2022_apache_ssl already enabled
root@495530af16d5:/etc/apache2/sites-available# service apache2 restart
 * Restarting Apache httpd web server apache2
Enter passphrase for SSL/TLS keys for www.cocot2022.com:443 (RSA):
                                                                    [ OK ]
```

图 20

配置完成后，在主机上使用火狐浏览器访问 HTTPS 网站（https://www.cocot2022.com），然而很遗憾，我们会发现此时无法访问网站。

为了修复这个问题，我们需要修改主机上的 hosts 文件（在/etc 目录下），将主机名 www.cocot2022.com 映射到网站服务器的 IP 地址。

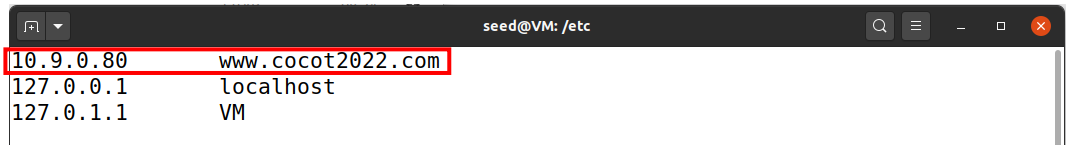

图 21

在主机上使用火狐浏览器重新访问网站，显示如下界面。

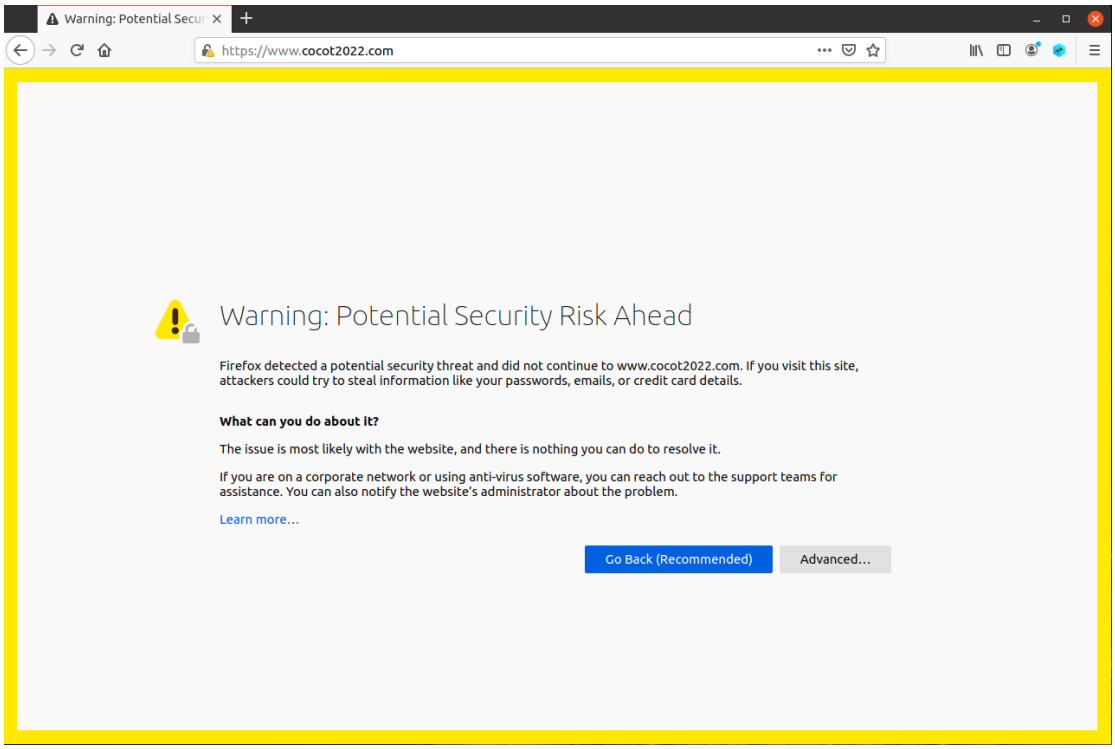

图 22

这说明这个连接是不安全的。这是因为浏览器没有 ModelCA 的公钥，因此它不能验证证书中的签名。我们需要向火狐浏览器的信任列表中手动添加一个 CA 证书。如图 23 所示，按照实验手册上的步骤，导入 Task 1 中创建的可信自签名证书。

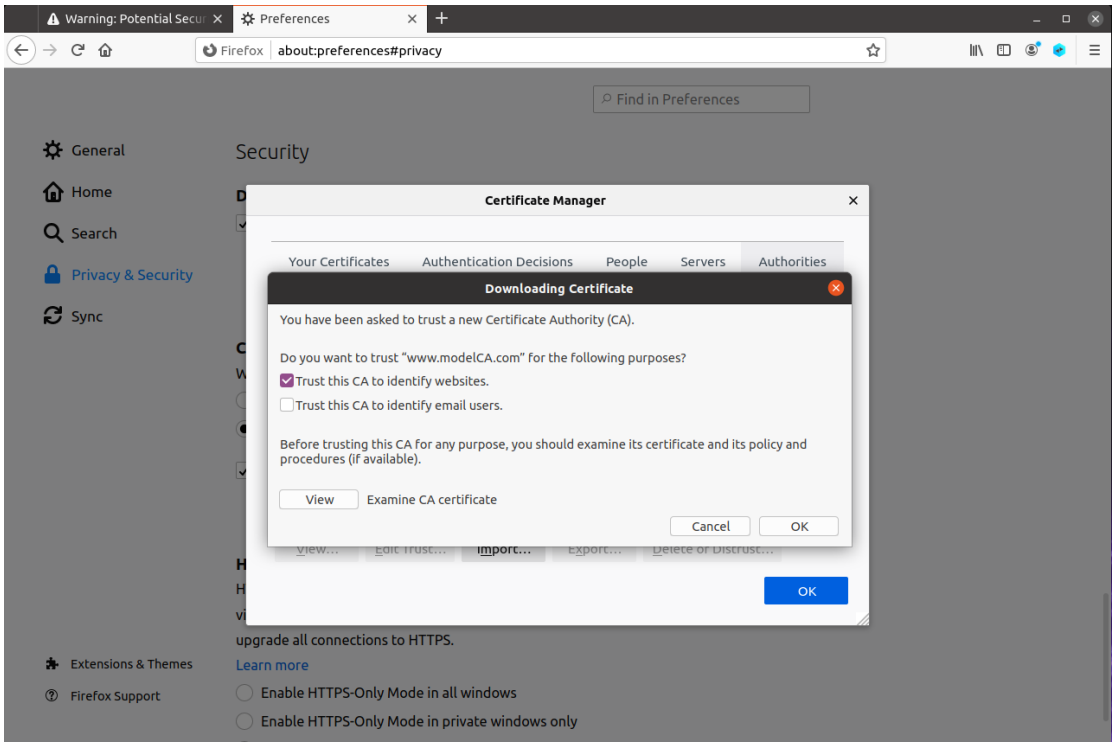

图 23

导入成功后，就可以在火狐浏览器的信任列表中看到 ModelCA 的证书。

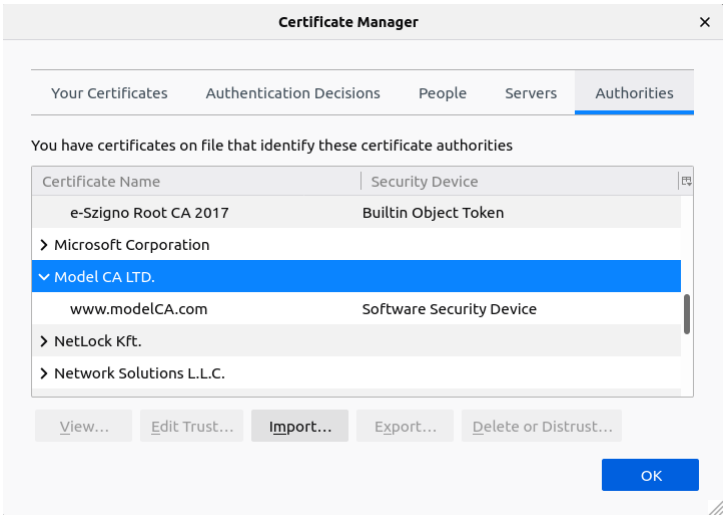

图 24

在主机上使用火狐浏览器重新访问网站，显示如下界面。

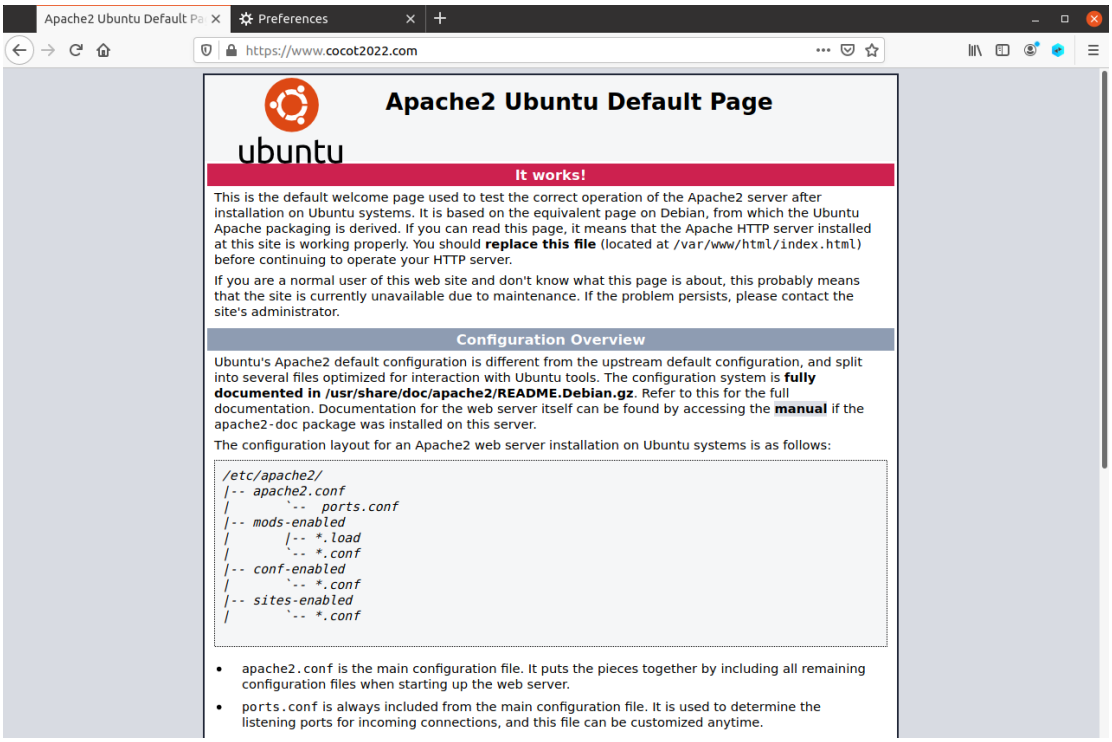

图 25

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

在此任务中，我们将展示 PKI 如何抵御中间人（MITM）攻击。

首先，设置一个恶意网站。在 Task 4 中，我们已经搭建了一个基于 Apache 的 HTTPS 网站。在这里，我们将使用相同的 Apache 服务器实现 www.example.com。类似图 18 展示的 VirtualHost 条目，我们将服务器名称修改为 www.example.com，其余内容保持一致，如图 26 所示。

这里我们模拟的攻击过程是，攻击者在劫持了用户的请求后，将自己的合法证书发送给用户。一旦证书到达用户的浏览器，浏览器将用已经预装的可信证书去验证它，这个验证将通过，因为攻击者的证书是有效的。

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

图 26

如图 27 所示，在 docker 中输入相关命令启动服务器。在 Apache 启动时，需要为每个 HTTPS 站点加载私钥。私钥在创建时被一个简单的口令加密，所以 Apache 会提示我们输入

口令进行私钥解密。这里我们输入生成 server.key 时规定的口令：dees。

```
root@495530af16d5:/volumes# cp example_apache_ssl.conf /etc/apache2/sites-available
root@495530af16d5:/volumes# cd /etc/apache2/sites-available
root@495530af16d5:/etc/apache2/sites-available# ls
000-default.conf        cocot2022_apache_ssl.conf  example_apache_ssl.conf
bank32_apache_ssl.conf  default-ssl.conf
root@495530af16d5:/etc/apache2/sites-available# apachectl configtest
Syntax OK
root@495530af16d5:/etc/apache2/sites-available# 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
root@495530af16d5:/etc/apache2/sites-available# a2ensite example_apache_ssl
Enabling site example_apache_ssl.
To activate the new configuration, you need to run:
  service apache2 reload
root@495530af16d5:/etc/apache2/sites-available# service apache2 restart
 * Restarting Apache httpd web server apache2
Enter passphrase for SSL/TLS keys for www.example.com:443 (RSA):
                                                                    [ OK ]
```

图 27

我们希望能够实现：用户试图访问 www.example.com 时，攻击者使用户登录自己的服务器，而攻击者的服务器上设置了一个伪造的 www.example.com 网站。

为了实现这个目标，攻击者可以攻击路由，将用户的 HTTPS 请求重定向到攻击者的 Web 服务器。另一种方法是攻击 DNS，当用户的主机在查找目标网站的 IP 地址时，会从 DNS 中得到攻击者的 Web 服务器的 IP 地址。

在此任务中，我们模拟了攻击 DNS 方法。我们只需修改主机的/etc/hosts 文件，将域名 www.example.com 映射到恶意的 Web 服务器上。

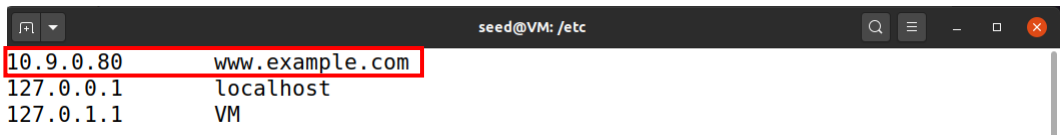

图 28

在主机上使用火狐浏览器访问网站 www.example.com，显示如下界面。浏览器注意到用户在 URL 中输入的是 https://www.example.com，因此浏览器知道用户的目的是访问 www.example.com，但是证书的 Subject 域是 www.cocot2022.com。这个不匹配导致浏览器立刻中断握手协议。

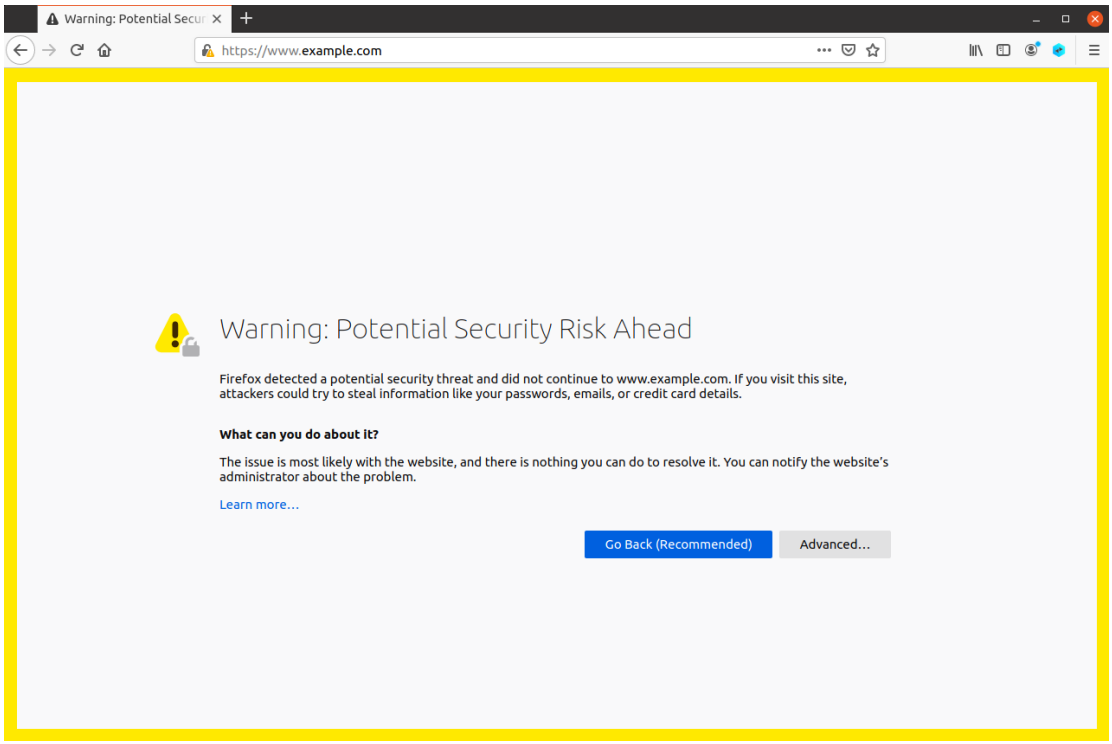

图 29

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

在此任务中，我们假设 Task 1 中创建的根 CA 被攻击者劫持，攻击者可以使用此 CA 的私钥生成任意证书。我们将设计一个实验来证明在这种情况下，攻击者可以成功地对任何目标发起 MITM 攻击。

我们维持 Task 5 中图 26-27 的所有设置。如图 30 所示，我们重新生成一对属于攻击者的私钥（attacker.key）和 CSR 请求（attacker.csr），加密口令为 attack。

```
root@495530af16d5:/# openssl req -newkey rsa:2048 -sha256 -keyout attacker.key -out
attacker.csr -subj "/CN=www.example.com/O=Example Inc./C=US" -passout pass:attack
Generating a RSA private key
................+++++
.............+++++
writing new private key to 'attacker.key'
-----
```

图 30

之后我们使用被劫持的 CA 对 CSR（attacker.csr）生成数字证书（example.crt），如图 31 所示。

```
root@495530af16d5:/# openssl ca -config myCA_openssl.cnf -policy policy_anything -md sha256
 -days 3650 -in attacker.csr -out example.crt -batch -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: 3 (0x3)
        Validity
            Not Before: Aug  8 16:49:23 2022 GMT
            Not After : Aug  5 16:49:23 2032 GMT
        Subject:
            countryName               = US
            organizationName          = Example Inc.
            commonName                = www.example.com
        X509v3 extensions:
            X509v3 Basic Constraints:
                CA:FALSE
            Netscape Comment:
                OpenSSL Generated Certificate
            X509v3 Subject Key Identifier:
                7B:38:A6:95:50:3F:B9:19:71:CA:E0:CA:E4:19:80:72:9E:08:10:73
            X509v3 Authority Key Identifier:
                keyid:CC:0D:1D:BD:BC:7C:A1:25:0A:F5:C1:78:F5:1C:25:7E:45:1F:61:A9

Certificate is to be certified until Aug  5 16:49:23 2032 GMT (3650 days)

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

图 31

移除原有的 server.crt 和 server.key 文件。将 attacker.key 重命名成 server.key，包含攻击者的私钥。将 example.crt（被劫持 CA 对 attacker.csr 签名生成 www.example.com 的证书）重命名为 server.crt，该证书里存放的是攻击者的公钥。由于在创建 HTTPS 网站时，我们只需要配置 Apache 服务器，给出获取数字证书和私钥的位置，所以当我们重新访问 www.example.com 时，Apache 服务器从原先的目录中找到了我们伪造的证书和对应的私钥，从而成功通过所有验证，MITM 攻击成功。对 www.example.com 的访问结果如图 32 所示。

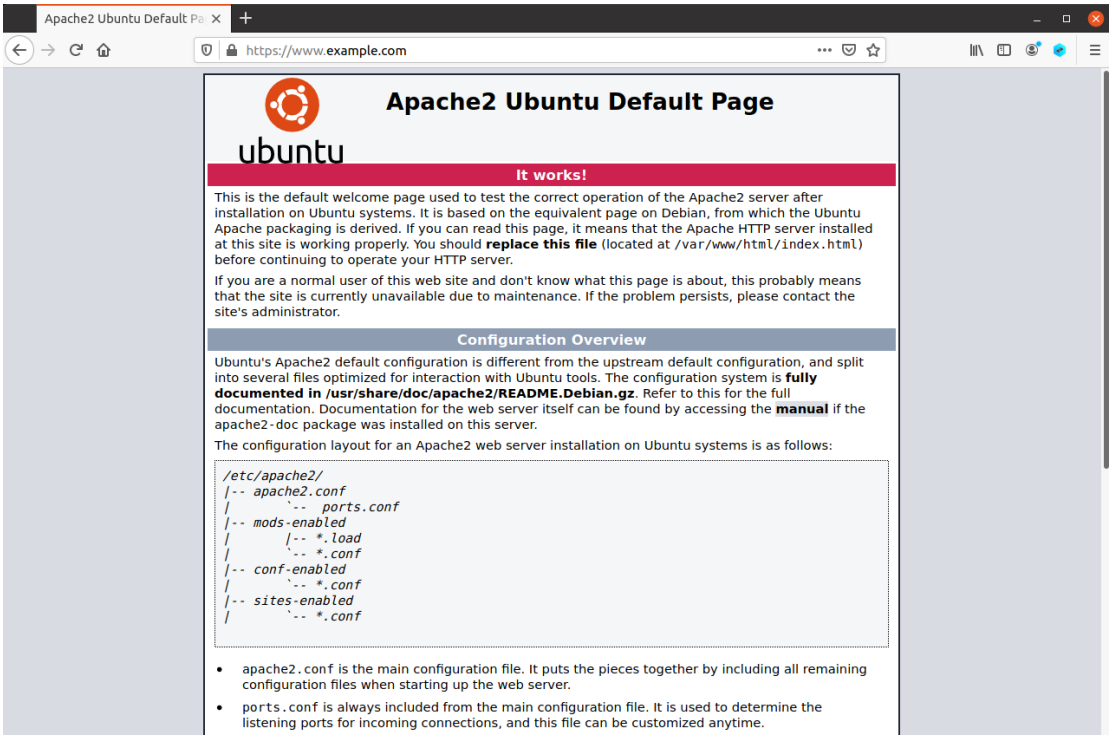

图 32

Summary

公钥密码是当今安全通信的基础，但当通信的一方将其公钥发送给另一方时，会受到 MITM 攻击的威胁。其根本问题在于，没有简单的方法来验证公钥的所有权，而公钥基础设施（PKI）是解决这一问题的有效方法。

通过本次实验，我深入理解了 PKI 的工作机制，并自主创建了可信任的根 CA，利用该 CA 对 Web 服务器签名；此外，我还初步模拟了 MITM 攻击的过程，了解其基本原理，并验证了当 CA 被劫持时，会对整个网络信息系统造成巨大的危害。

在实验过程中，我通过查找资料，与老师和同学讨论等方法，克服了遇到的种种困难，例如：docker 中无法直接使用 vim 修改文件，主机和 docker 之间的通信问题等等。