

# 实验一：私有 CA 证书签发的简单实现

专业：密码科学与技术 姓名：柳致远 学号：2113683

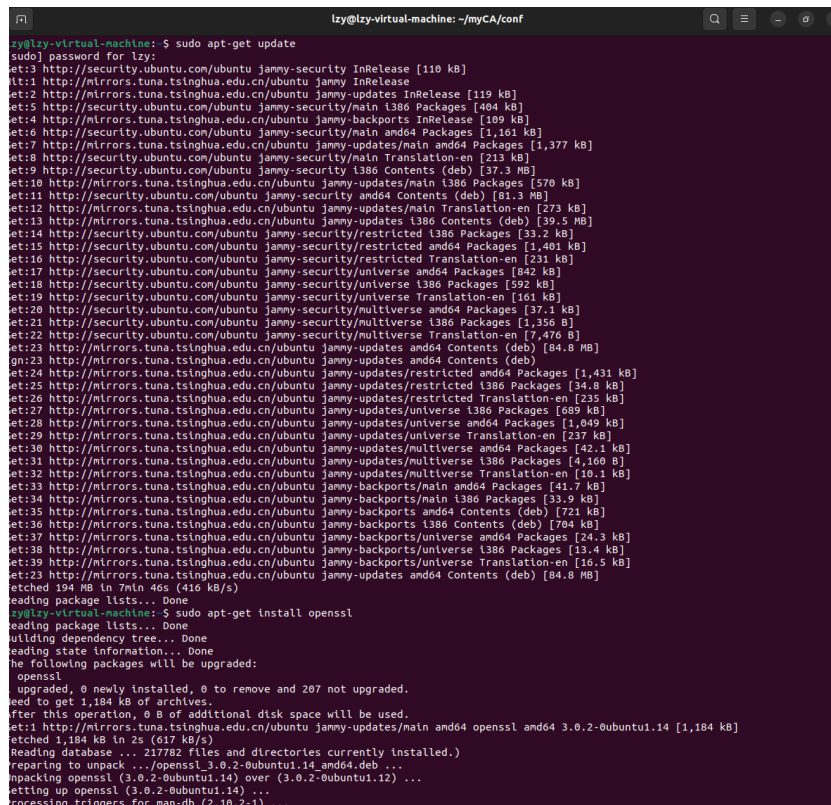
实验过程：

## 一、搭建私有 CA

### 1、安装 OpenSSL：

```
sudo apt-get update
```

```
sudo apt-get install openssl
```



```
lzy@lzy-virtual-machine: ~/myCA/conf
lzy@lzy-virtual-machine:~$ sudo apt-get update
sudo: password for lzy:
Get:1 http://security.ubuntu.com/ubuntu jammy-security InRelease [110 kB]
Get:2 http://mirrors.tuna.tsinghua.edu.cn/ubuntu jammy InRelease
Get:3 http://mirrors.tuna.tsinghua.edu.cn/ubuntu jammy-updates InRelease [119 kB]
Get:4 http://security.ubuntu.com/ubuntu jammy-security/main i386 Packages [404 kB]
Get:5 http://mirrors.tuna.tsinghua.edu.cn/ubuntu jammy-backports InRelease [109 kB]
Get:6 https://security.ubuntu.com/ubuntu jammy-security/main amd64 Packages [1,161 kB]
Get:7 http://mirrors.tuna.tsinghua.edu.cn/ubuntu jammy-updates/main amd64 Packages [1,377 kB]
Get:8 http://security.ubuntu.com/ubuntu jammy-security/main Translation-en [213 kB]
Get:9 http://security.ubuntu.com/ubuntu jammy-security i386 Contents (deb) [37.3 MB]
Get:10 http://mirrors.tuna.tsinghua.edu.cn/ubuntu jammy-updates/main i386 Packages [570 kB]
Get:11 http://mirrors.tuna.tsinghua.edu.cn/ubuntu jammy-updates/main Translation-en [273 kB]
Get:12 http://mirrors.tuna.tsinghua.edu.cn/ubuntu jammy-updates i386 Contents (deb) [39.5 MB]
Get:13 http://security.ubuntu.com/ubuntu jammy-security/restricted i386 Packages [33.2 kB]
Get:14 http://security.ubuntu.com/ubuntu jammy-security/restricted amd64 Packages [1,401 kB]
Get:15 http://security.ubuntu.com/ubuntu jammy-security/restricted Translation-en [231 kB]
Get:16 http://security.ubuntu.com/ubuntu jammy-security/universe amd64 Packages [842 kB]
Get:17 http://security.ubuntu.com/ubuntu jammy-security/universe i386 Packages [592 kB]
Get:18 http://security.ubuntu.com/ubuntu jammy-security/universe Translation-en [161 kB]
Get:19 http://security.ubuntu.com/ubuntu jammy-security/multiverse amd64 Packages [37.1 kB]
Get:20 http://security.ubuntu.com/ubuntu jammy-security/multiverse i386 Packages [1,356 B]
Get:21 http://security.ubuntu.com/ubuntu jammy-security/multiverse Translation-en [7,476 B]
Get:22 http://mirrors.tuna.tsinghua.edu.cn/ubuntu jammy-updates amd64 Contents (deb) [84.8 MB]
Get:23 http://mirrors.tuna.tsinghua.edu.cn/ubuntu jammy-updates/restricted amd64 Packages [1,431 kB]
Get:24 http://mirrors.tuna.tsinghua.edu.cn/ubuntu jammy-updates/restricted i386 Packages [34.8 kB]
Get:25 http://mirrors.tuna.tsinghua.edu.cn/ubuntu jammy-updates/restricted Translation-en [235 kB]
Get:26 http://mirrors.tuna.tsinghua.edu.cn/ubuntu jammy-updates/universe i386 Packages [689 kB]
Get:27 http://mirrors.tuna.tsinghua.edu.cn/ubuntu jammy-updates/universe amd64 Packages [1,049 kB]
Get:28 http://mirrors.tuna.tsinghua.edu.cn/ubuntu jammy-updates/universe Translation-en [227 kB]
Get:29 http://mirrors.tuna.tsinghua.edu.cn/ubuntu jammy-updates/multiverse amd64 Packages [42.1 kB]
Get:30 http://mirrors.tuna.tsinghua.edu.cn/ubuntu jammy-updates/multiverse i386 Packages [4,160 B]
Get:31 http://mirrors.tuna.tsinghua.edu.cn/ubuntu jammy-updates/multiverse Translation-en [10.1 kB]
Get:32 http://mirrors.tuna.tsinghua.edu.cn/ubuntu jammy-backports/main amd64 Packages [41.7 kB]
Get:33 http://mirrors.tuna.tsinghua.edu.cn/ubuntu jammy-backports/main i386 Packages [33.9 kB]
Get:34 http://mirrors.tuna.tsinghua.edu.cn/ubuntu jammy-backports amd64 Contents (deb) [721 kB]
Get:35 http://mirrors.tuna.tsinghua.edu.cn/ubuntu jammy-backports i386 Contents (deb) [704 kB]
Get:36 http://mirrors.tuna.tsinghua.edu.cn/ubuntu jammy-backports/universe amd64 Packages [24.3 kB]
Get:37 http://mirrors.tuna.tsinghua.edu.cn/ubuntu jammy-backports/universe i386 Packages [13.4 kB]
Get:38 http://mirrors.tuna.tsinghua.edu.cn/ubuntu jammy-backports/universe Translation-en [16.5 kB]
Get:39 http://mirrors.tuna.tsinghua.edu.cn/ubuntu jammy-updates amd64 Contents (deb) [84.8 MB]
Fetched 194 MB in 7min 46s (416 kB/s)
Reading package lists... Done
lzy@lzy-virtual-machine:~$ sudo apt-get install openssl
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following packages will be upgraded:
  openssl
1 upgraded, 0 newly installed, 0 to remove and 207 not upgraded.
Need to get 1,184 kB of archives.
After this operation, 0 B of additional disk space will be used.
Get:1 http://mirrors.tuna.tsinghua.edu.cn/ubuntu jammy-updates/main amd64 openssl amd64 3.0.2-0ubuntu1.14 [1,184 kB]
Fetched 1,184 kB in 2s (617 kB/s)
Reading database ... 217782 files and directories currently installed.)
Preparing to unpack .../openssl_3.0.2-0ubuntu1.14.amd64.deb ...
Unpacking openssl (3.0.2-0ubuntu1.14) over (3.0.2-0ubuntu1.12) ...
Setting up openssl (3.0.2-0ubuntu1.14) ...
Processing triggers for man-db (2.10.2-1) ...
```

### 2、创建私有 CA 所需要的文件目录，保存 CA 的相关信息

相关指令如下：

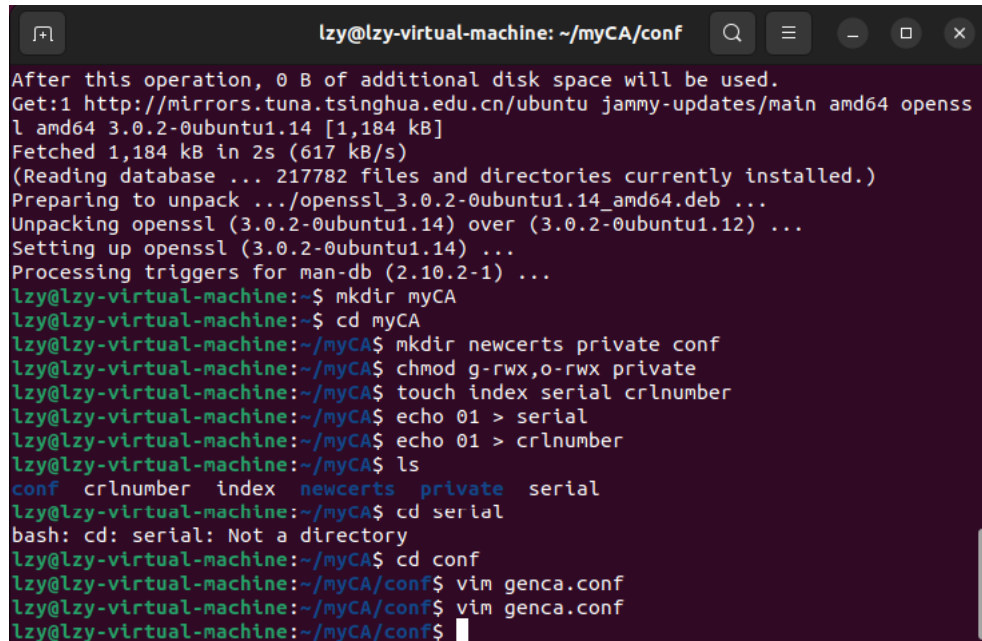
```
mkdir myCA //创建 CA 根文件夹
```

```
cd myCA //进入 CA 根文件夹
```

```
mkdir newcerts private conf //创建三个文件夹，用来存放新发  
放证书、私钥和配置文件
```

```
chmod g-rwx,o-rwx private //设置 private 文件夹的操作权限
touch index serial crlnumber //创建证书信息数据库、证书序号
文件、crl 序号文件
echo 01 > serial //初始化证书的序号
echo 01 > crlnumber //初始化吊销证书序号
```

实验过程截图：



```
lzy@lzy-virtual-machine: ~/myCA/conf
After this operation, 0 B of additional disk space will be used.
Get:1 http://mirrors.tuna.tsinghua.edu.cn/ubuntu jammy-updates/main amd64 openssl
amd64 3.0.2-0ubuntu1.14 [1,184 kB]
Fetched 1,184 kB in 2s (617 kB/s)
(Reading database ... 217782 files and directories currently installed.)
Preparing to unpack .../openssl_3.0.2-0ubuntu1.14_amd64.deb ...
Unpacking openssl (3.0.2-0ubuntu1.14) over (3.0.2-0ubuntu1.12) ...
Setting up openssl (3.0.2-0ubuntu1.14) ...
Processing triggers for man-db (2.10.2-1) ...
lzy@lzy-virtual-machine:~$ mkdir myCA
lzy@lzy-virtual-machine:~$ cd myCA
lzy@lzy-virtual-machine:~/myCA$ mkdir newcerts private conf
lzy@lzy-virtual-machine:~/myCA$ chmod g-rwx,o-rwx private
lzy@lzy-virtual-machine:~/myCA$ touch index serial crlnumber
lzy@lzy-virtual-machine:~/myCA$ echo 01 > serial
lzy@lzy-virtual-machine:~/myCA$ echo 01 > crlnumber
lzy@lzy-virtual-machine:~/myCA$ ls
conf crlnumber index newcerts private serial
lzy@lzy-virtual-machine:~/myCA$ cd serial
bash: cd: serial: Not a directory
lzy@lzy-virtual-machine:~/myCA$ cd conf
lzy@lzy-virtual-machine:~/myCA/conf$ vim genca.conf
lzy@lzy-virtual-machine:~/myCA/conf$ vim genca.conf
lzy@lzy-virtual-machine:~/myCA/conf$
```

### 3、 创建生成 CA 自签名证书的配置文件

相关指令如下：

```
cd conf
//进入配置文件夹
vim genca.conf
//创建用来生成自签名证书的配置文件
```

配置文件配置如下：

```
lzy@lzy-virtual-machine: ~/myCA/conf
[ req ]
default_keyfile = /root/myCA/private/cakey.pem
default_md = md5
prompt = no
distinguished_name = ca_distinguished_name
x509_extensions = ca_extensions

[ ca_distinguished_name ]
organizationName = DCYorg
organizationalUnitName = DCYunit
commonName = DCY
emailAddress = dcy@nankai.edu.cn

[ ca_extensions ]
basicConstraints = CA:true
~
~
~
~
~
~
-- INSERT --
```

#### 4、 生成私有 CA 的私钥和自签名证书（根证书）

指令如下：

```
openssl req -x509 -newkey rsa:2048 -out cacert.pem -
outform PEM -days 2190 -config /root/myCA/conf/genca.conf
//回到 myCA 根目录下，生成 x509 自签名证书，过程中需要输入
CA 私钥的保护密码，请牢记。
//CA 会按照 gentestca.conf 文件中配置的规则自签名生成证书
```

CA 自签名证书截图如下：

```

root@lzy-virtual-machine: /home/lzy/myCA
+++++
Enter PEM pass phrase:
Verifying - Enter PEM pass phrase:
-----
root@lzy-virtual-machine: /home/lzy/myCA# openssl x509 -in cacert.pem -text -noout
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number:
            28:6a:bf:d4:0e:87:9f:0e:45:f3:3e:d5:1e:bc:61:86:59:41:fe:5f
        Signature Algorithm: md5WithRSAEncryption
        Issuer: O = DCYorg, OU = DCYunit, CN = DCY, emailAddress = dcy@nankai.edu.cn
        Validity
            Not Before: Feb 20 07:39:01 2024 GMT
            Not After : Feb 18 07:39:01 2030 GMT
        Subject: O = DCYorg, OU = DCYunit, CN = DCY, emailAddress = dcy@nankai.edu.cn
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
            Public-Key: (2048 bit)
            Modulus:
                00:e8:2c:49:c0:30:3f:9c:9f:28:0b:e4:7e:69:bd:
                8c:c2:06:dd:57:73:95:f7:4f:c9:2f:e8:61:60:d9:
                d8:10:61:ca:1b:ad:36:e2:60:02:5f:e0:2e:0e:3c:
                21:c2:88:99:a1:f4:1c:5d:0a:5d:9d:02:36:e4:02:
                1f:80:bd:23:ce:d2:07:3f:7b:96:a2:7d:ef:48:2d:
                d5:85:93:15:30:1c:ef:77:b9:2f:08:71:09:ee:f8:
                a5:9e:ba:c5:e9:3a:40:1f:7a:c0:b9:2d:67:d4:c3:
                92:ab:6d:ae:0a:e0:be:9d:ce:1a:e9:fe:5d:d7:e2:
                ee:f5:3a:52:2e:a5:18:71:65:96:78:ab:e1:93:a7:
                32:90:80:c0:a7:4d:5b:2c:b2:3d:b7:c9:99:30:53:
                36:7f:6c:04:8f:42:b0:83:f7:52:82:b9:0c:7d:7c:
                5e:66:45:62:92:c6:ed:4d:6f:13:5a:ed:dd:5f:d2:
                1a:28:95:90:9b:21:ab:80:7c:ca:3c:f0:01:33:ce:
                00:5f:23:b4:47:c5:71:57:55:3d:9b:de:c5:80:03:
                d1:0b:0d:80:01:0a:4b:0c:57:73:15:3e:ec:31:99:
                f4:0e:51:3f:c5:09:f3:8a:42:21:97:d9:a5:15:78:
                09:6d:34:5d:71:d6:fd:97:e1:b3:fb:d6:cd:d8:3e:
                4e:2d
            Exponent: 65537 (0x10001)
        X509v3 extensions:
            X509v3 Basic Constraints:
                CA:TRUE
            X509v3 Subject Key Identifier:
                B6:3E:3E:76:AB:D9:17:87:BB:4E:CE:09:78:2B:90:FC:31:B1:92:9E
        Signature Algorithm: md5WithRSAEncryption
        Signature Value:
            a0:05:e3:e6:cf:e7:a6:ab:70:fe:e2:4c:14:b6:e6:eb:c6:bd:
            c8:6d:e2:db:10:a1:39:66:3d:b3:ea:41:d8:e6:c7:6b:ba:70:
            5e:60:39:9c:b9:50:f4:24:0b:e7:40:8b:6d:d7:c9:17:e5:70:
            0a:ef:b2:77:d2:36:59:05:13:11:ea:d4:31:24:d6:9d:47:4f:
            e2:63:a0:59:07:d9:c3:5c:76:4a:a2:99:52:eb:c9:6b:79:ba:
            4b:21:cf:51:6f:6d:be:91:bc:ba:4d:e5:09:57:ce:6f:fb:90:
            be:81:2e:24:55:61:3e:47:16:4b:39:48:a4:78:9e:06:09:34:
            45:48:46:67:c7:f6:3d:aa:e2:ff:6b:91:aa:0c:55:45:6a:b0:
            28:b4:a3:c3:92:28:fc:85:5c:0c:ce:44:07:22:77:e5:aa:4a:
            cb:e0:5e:98:44:30:dd:29:65:c1:0c:81:49:c8:1a:05:2a:bd:
            d1:98:03:5c:7d:30:f9:a4:d6:02:82:4b:27:1f:ea:62:ff:68:
            52:9f:6d:bd:7f:44:e7:13:77:30:46:01:db:7f:57:95:1f:2b:
            e3:9a:2a:2d:d7:01:9f:ae:09:3b:29:44:cc:4f:06:f9:34:74:
            b7:20:4c:db:8b:18:91:b3:70:91:76:1a:88:74:b7:08:30:68:
            2a:50:49:02
root@lzy-virtual-machine: /home/lzy/myCA#

```

## 二、私有 CA 为服务器签发证书

### 1、 创建用来为其他请求签发证书的配置文件

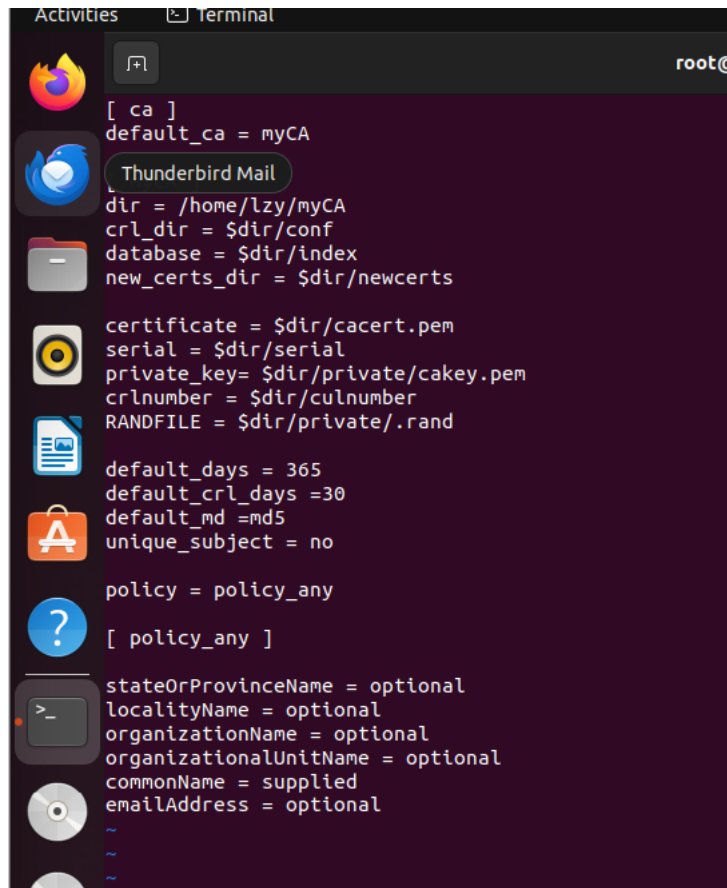
```
cd conf
```

```
//再次进入配置文件夹
```

```
vim ca.conf
```

```
//创建用来为其他请求签发证书的配置文件
```

配置文件配置如下：

A terminal window with a dark purple background and light blue text. The prompt is 'root@'. The text shows the configuration for a Certificate Authority (CA) named 'ca'. It sets 'default\_ca = myCA' and defines several variables for directories and files: 'dir = /home/lzy/myCA', 'crl\_dir = \$dir/conf', 'database = \$dir/index', 'new\_certs\_dir = \$dir/newcerts', 'certificate = \$dir/cacert.pem', 'serial = \$dir/serial', 'private\_key = \$dir/private/cakey.pem', 'crlnumber = \$dir/culnumber', and 'RANDFILE = \$dir/private/.rand'. It also sets 'default\_days = 365', 'default\_crl\_days = 30', 'default\_md = md5', and 'unique\_subject = no'. The policy is set to 'policy\_any', and the [policy\_any] section lists optional fields: 'stateOrProvinceName', 'localityName', 'organizationName', 'organizationalUnitName', 'commonName' (supplied), and 'emailAddress' (optional).

```
Activities Terminal root@
[ ca ]
default_ca = myCA
Thunderbird Mail
dir = /home/lzy/myCA
crl_dir = $dir/conf
database = $dir/index
new_certs_dir = $dir/newcerts
certificate = $dir/cacert.pem
serial = $dir/serial
private_key = $dir/private/cakey.pem
crlnumber = $dir/culnumber
RANDFILE = $dir/private/.rand
default_days = 365
default_crl_days = 30
default_md = md5
unique_subject = no
policy = policy_any
[ policy_any ]
stateOrProvinceName = optional
localityName = optional
organizationName = optional
organizationalUnitName = optional
commonName = supplied
emailAddress = optional
~
~
```

## 2、模拟服务器，生成私钥与证书申请的请求文件

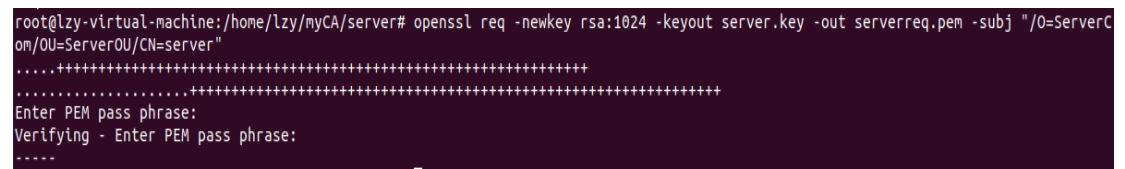
mudar server

//在任意路径下创建服务器文件夹 server

```
openssl req -newkey rsa:1024 -keyout server.key -out serverreq.pem -subj "/O=ServerCom/OU=ServerOU/CN=server"
```

//生成 server 的 1024 位私钥 server.key 和证书申请的请求文件 serverreq.pem，此时需要设置服务器的私钥保护密码，请牢记

结果截图如下：

A terminal window showing the execution of the 'openssl req' command. The prompt is 'root@lzy-virtual-machine: /home/lzy/myCA/server#'. The command is 'openssl req -newkey rsa:1024 -keyout server.key -out serverreq.pem -subj "/O=ServerCom/OU=ServerOU/CN=server"'. The output shows a series of dots, followed by 'Enter PEM pass phrase:' and 'Verifying - Enter PEM pass phrase:'.

```
root@lzy-virtual-machine: /home/lzy/myCA/server# openssl req -newkey rsa:1024 -keyout server.key -out serverreq.pem -subj "/O=ServerCom/OU=ServerOU/CN=server"
.....
Enter PEM pass phrase:
Verifying - Enter PEM pass phrase:
.....
```

## 3、CA 根据服务器的证书请求文件生成证书并将其返回给服务器

```
openssl ca -in serverreq.pem -out server.crt -config /root/myCA/conf/ca.conf
```

//向私有 CA 提交证书请求文件 serverreq.pem，CA 生成并返回证书 server.crt

//生成证书的规则是参照之前为 CA 定义的 ca.conf 配置文件执行的

实验过程截图如下：

```
root@lzy-virtual-machine:/home/lzy/myCA/server# openssl ca -in serverreq.pem -out server.crt -config /home/lzy/myCA/conf/ca.conf
Using configuration from /home/lzy/myCA/conf/ca.conf
Enter pass phrase for /home/lzy/myCA/private/cakey.pem:
Check that the request matches the signature
Signature ok
The Subject's Distinguished Name is as follows
organizationName      :ASN.1 12:'ServerCom'
organizationalUnitName:ASN.1 12:'ServerOU'
commonName            :ASN.1 12:'server'
Certificate is to be certified until Feb 19 11:20:31 2025 GMT (365 days)
Sign the certificate? [y/n]:y

1 out of 1 certificate requests certified, commit? [y/n]y
Write out database with 1 new entries
Data Base Updated
```

- 4、 在 myCA 下的 index（证书信息数据库）中可以看到证书信息数据库的更新数据

```
V          250219112031Z          01          unknown /O=ServerCom/OU=ServerOU/CN=server
```

以及，在 newcerts 目录下可以看到 CA 发放给服务器的证书文件备份。

```
root@lzy-virtual-machine:/home/lzy/myCA/newcerts# ls
01.pem
```

### 三、私有 CA 为客户端签发证书

- 1、即仿照为服务端签发证书的过程为客户端签发证书即可

首先在 myCA 目录下创建文件夹 client：

```
mkdir client
```

接着生成 client 的 1024 位私钥 client.key 和证书申请的请求文件 client.pem，此时需要设置服务器的私钥保护密码，代码修改如下：

```
openssl req -newkey rsa:1024 -keyout client.key -out clientreq.pem -subj "/O=ClientCom/OU=ClientOU/CN=client"
```

实验过程截图如下：

```

root@lzy-virtual-machine:/home/lzy/myCA/client# ls
root@lzy-virtual-machine:/home/lzy/myCA/client# openssl req -newkey rsa:1024 -keyout client.key
-out clientreq.pem -subj "/O=ClientCom/OU=ClientOU/CN=client"
.....+++++
+++++
Enter PEM pass phrase:
Verifying - Enter PEM pass phrase:
-----

```

2、同样使用之前的配置文件 ca.conf 为客户端生成并返回证书

client.crt

```

root@lzy-virtual-machine:/home/lzy/myCA/client# openssl ca -in clientreq.pem -out client.crt
-config /home/lzy/myCA/conf/ca.conf
Using configuration from /home/lzy/myCA/conf/ca.conf
Enter pass phrase for /home/lzy/myCA/private/akey.pem:
Check that the request matches the signature
Signature ok
The Subject's Distinguished Name is as follows
organizationName      :ASN.1 12:'ClientCom'
organizationalUnitName:ASN.1 12:'ClientOU'
commonName            :ASN.1 12:'client'
Certificate is to be certified until Feb 20 12:42:49 2025 GMT (365 days)
Sign the certificate? [y/n]:y

1 out of 1 certificate requests certified, commit? [y/n]y
Write out database with 1 new entries
Data Base Updated

```

查看证书 client.crt

```

Certificate:
  Data:
    Version: 1 (0x0)
    Serial Number: 2 (0x2)
    Signature Algorithm: md5WithRSAEncryption
    Issuer: O=DCYorg, OU=DCYunit, CN=DCY/emailAddress=dcy@nankai.edu.cn
    Validity
      Not Before: Feb 21 12:42:49 2024 GMT
      Not After : Feb 20 12:42:49 2025 GMT
    Subject: O=ClientCom, OU=ClientOU, CN=client
    Subject Public Key Info:
      Public Key Algorithm: rsaEncryption
      Public-Key: (1024 bit)
      Modulus:
        00:c3:a8:76:15:93:4d:92:c9:3c:b6:13:42:b2:04:
        e9:ae:5c:83:68:a8:ea:06:3b:6a:4d:0c:82:df:4e:
        48:2f:76:5a:07:bd:39:a6:a9:c2:2e:80:c4:3a:4d:
        57:aa:41:24:8f:b5:cf:80:9c:63:48:dd:69:20:63:
        57:1b:81:0f:c9:27:8c:55:c7:c2:e0:68:ad:66:cc:
        2f:2d:28:1a:db:09:bc:4b:20:d8:56:c5:81:86:aa:
        c4:52:6a:f2:ce:69:fd:f0:c6:37:11:94:08:7b:d0:
        21:d7:12:a6:88:0e:82:73:d7:18:55:1a:71:aa:08:
        5d:75:1d:a8:31:68:cf:28:59
      Exponent: 65537 (0x10001)
    Signature Algorithm: md5WithRSAEncryption
    Signature Value:
      14:33:4a:f3:4f:77:bb:c4:bc:ab:48:6a:bf:f8:d4:9c:ba:9f:
      de:eb:b6:fc:74:92:4e:5e:ff:6c:19:20:ae:5d:c9:d7:f9:cd:
      ff:08:57:40:4b:72:ec:25:1d:5a:b4:d5:20:a1:5f:86:2e:48:
      f8:55:ca:40:f3:2b:f9:08:80:fc:c3:c0:1d:3e:35:6c:2f:5b:
      42:f6:26:56:aa:6f:d2:6a:f3:64:ba:7f:64:d5:e7:27:fc:da:
      94:13:ec:cd:c4:b0:30:c8:3a:e0:67:21:95:3f:a5:01:8e:02:
      2c:49:9d:5c:12:22:fc:40:95:23:87:c6:ea:39:5e:18:58:25:
      dc:c4:99:b3:12:70:7a:58:17:c3:6b:c3:47:c3:f9:b6:ce:9d:
      5b:cc:64:ff:4e:bb:ad:99:86:ae:72:1d:9b:de:36:bd:d3:b1:
      74:cb:de:da:3d:dd:e0:c5:70:79:eb:fb:15:2e:ea:7a:d4:b2:
      2f:c9:4e:97:8a:59:d0:7f:61:d6:a8:b0:a4:7c:71:93:5f:c6:

```

5,1

Top

在 newcerts 文件夹内保存了该证书的备份



```
root@lzy-virtual-machine:/home/lzy/myCA/newcerts# ls
01.pem 02.pem
```

并在 myCA 下的 index（证书信息数据库）中可以看到证书信息数据库的更新数据

```

V      250219112031Z      01      unknown /O=ServerCom/OU=ServerOU/CN=server
V      250220124249Z      02      unknown /O=ClientCom/OU=ClientOU/CN=client
```

#### 四、CA 吊销用户证书

1、 在之前的配置文件 ca.conf 的基础上生成证书吊销列表

ca.crl:

```
openssl ca -config /home/lzy/myCA/conf/ca.conf -gencrl
-out ca.crl -crl days 30
```

实验结果截图如下:

```
root@lzy-virtual-machine:/home/lzy/myCA# openssl ca -config /home
/lzy/myCA/conf/ca.conf -gencrl -out ca.crl -crl days 30
Using configuration from /home/lzy/myCA/conf/ca.conf
Enter pass phrase for /home/lzy/myCA/private/cakey.pem:
root@lzy-virtual-machine:/home/lzy/myCA#
```

2、 吊销 01 证书, 使用命令

```
openssl ca -revoke /home/lzy/myCA/newcerts/01.pem
-config "/home/lzy/myCA/conf/ca.conf" 进行吊销
```

实验结果截图如下:

```
root@lzy-virtual-machine:/home/lzy/myCA# openssl ca -revoke /home
/lzy/myCA/newcerts/01.pem -config "/home/lzy/myCA/conf/ca.conf"
Using configuration from /home/lzy/myCA/conf/ca.conf
Enter pass phrase for /home/lzy/myCA/private/cakey.pem:
Revoking Certificate 01.
Data Base Updated
root@lzy-virtual-machine:/home/lzy/myCA#
```

3 、使用命令 `openssl ca -gencrl -out /home/lzy/myCA/ca.crl -config`



/home/lzy/myCA/conf/ca.conf -crl days 30 更新证书吊销列表

实验过程截图如下

```
root@lzy-virtual-machine:/home/lzy/myCA# openssl ca -gencrl -out  
/home/lzy/myCA/ca.crl -config /home/lzy/myCA/conf/ca.conf -crlday  
s 30  
Using configuration from /home/lzy/myCA/conf/ca.conf  
Enter pass phrase for /home/lzy/myCA/private/cakey.pem:  
root@lzy-virtual-machine:/home/lzy/myCA#
```

使用命令 `openssl crl -in ca.crl -noout -text` 可以查看 `crl` 文件

```
root@lzy-virtual-machine:/home/lzy/myCA# openssl crl -in ca.crl -noout -text  
Certificate Revocation List (CRL):  
  Version 2 (0x1)  
  Signature Algorithm: md5WithRSAEncryption  
  Issuer: O = DCYorg, OU = DCYunit, CN = DCY, emailAddress = dcy@nankai.edu.cn  
  Last Update: Mar  3 20:01:01 2024 GMT  
  Next Update: Apr  2 20:01:01 2024 GMT  
  CRL extensions:  
    X509v3 CRL Number:  
      4  
Revoked Certificates:  
  Serial Number: 01  
    Revocation Date: Mar  3 19:58:18 2024 GMT  
  Signature Algorithm: md5WithRSAEncryption  
  Signature Value:  
    14:41:67:13:09:15:47:b8:f8:53:3c:9d:a3:fc:b7:67:62:8b:  
    b0:0c:d5:6d:e5:24:0a:21:2a:e9:ea:b7:9e:66:b2:9e:42:3b:  
    7e:d6:7c:4c:af:6c:fd:1c:9d:49:0f:15:60:52:d8:b5:d1:de:  
    2b:0d:02:cb:18:ab:09:75:c9:de:9d:bc:9f:3e:d7:c4:53:df:  
    c1:09:fe:fc:3c:02:83:bd:fd:94:12:8b:6e:7c:b5:4e:d7:02:  
    b1:ef:84:aa:f4:61:21:c9:9d:23:be:f8:e8:73:90:56:9c:f2:  
    fa:f4:79:3b:7e:1d:04:fe:61:41:1d:44:f8:4a:1c:15:a1:6a:  
    1d:66:64:b0:c0:2e:81:72:3d:e8:22:a8:8b:00:6f:2f:47:f0:  
    47:67:61:a6:ac:14:9c:b8:4e:0c:c9:00:6f:0c:22:5e:e3:4b:  
    d8:0d:2d:90:1d:e0:98:aa:a5:15:8d:75:79:e5:56:7e:3a:30:  
    6d:5c:40:e0:77:2a:c0:90:62:e9:49:03:f1:46:b6:28:68:0b:  
    50:42:c6:01:cd:ca:a7:f3:5f:61:98:d4:05:39:a5:d6:1f:48:  
    b9:ea:3d:72:a8:83:36:5e:9a:b9:73:51:5a:bd:a9:3b:cd:35:  
    db:38:2c:be:88:31:6c:e3:f2:20:69:d7:5d:ba:9f:55:90:e7:  
    7c:37:d0:dd  
root@lzy-virtual-machine:/home/lzy/myCA#
```