CA服务器的搭建以及证书签署、dropbear的编译安装

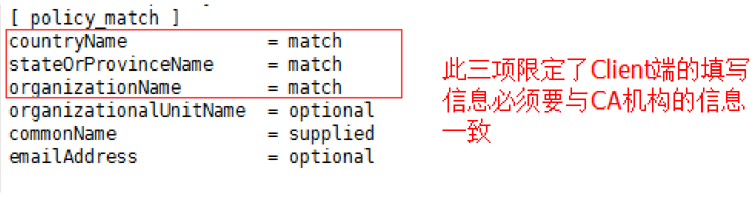
CA服务器的搭建以及证书签署、dropbear的编译安装

一、CA Server和Client:

1、CA server：创建私钥CA

（1） openssl的配置文件：/etc/pki/tls/openssl.conf

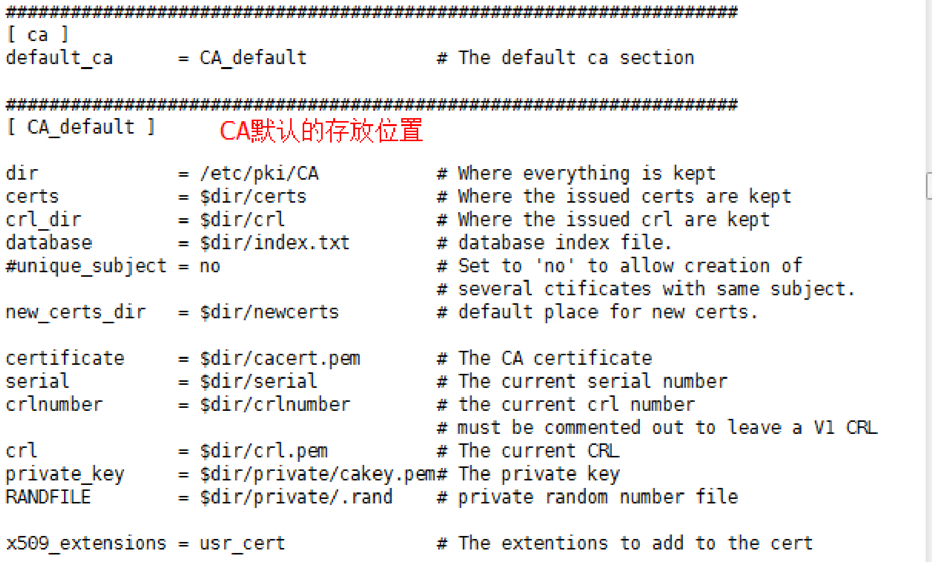
如果Client端的申请是来自不同的国家，则需要将下图中红色框内的三项，由“match”改为“optional”



```
[ policy_match ]
countryName             = match
stateOrProvinceName     = match
organizationName        = match
organizationalUnitName  = optional
commonName              = supplied
emailAddress            = optional
```

此三项限定了Client端的填写信息必须要与CA机构的信息一致

blob.png



```
####################################################################
[ ca ]
default_ca      = CA_default            # The default ca section

####################################################################
[ CA_default ]

dir             = /etc/pki/CA           # Where everything is kept
certs           = $dir/certs            # Where the issued certs are kept
crl_dir         = $dir/crl              # Where the issued crl are kept
database        = $dir/index.txt        # database index file.
#unique_subject = no                    # Set to 'no' to allow creation of
                                        # several ctificates with same subject.
new_certs_dir   = $dir/newcerts         # default place for new certs.

certificate     = $dir/cacert.pem       # The CA certificate
serial          = $dir/serial           # The current serial number
crlnumber       = $dir/crlnumber        # the current crl number
                                        # must be commented out to leave a V1 CRL
crl             = $dir/crl.pem          # The current CRL
private_key     = $dir/private/cakey.pem# The private key
RANDFILE        = $dir/private/.rand    # private random number file

x509_extensions = usr_cert              # The extentions to add to the cert
```

CA默认的存放位置

blob.png

由上图可以得知CA的存放位置为/etc/pki/CA目录下，该目录下有index.txt数据库文件、crl吊销证书存放目录、certs证书存放目录、newscerts默认新证书存放位置、serial证书签署的序列号、crlnumber吊销证书的序列号等等文件。

（2） 创建所需要的文件

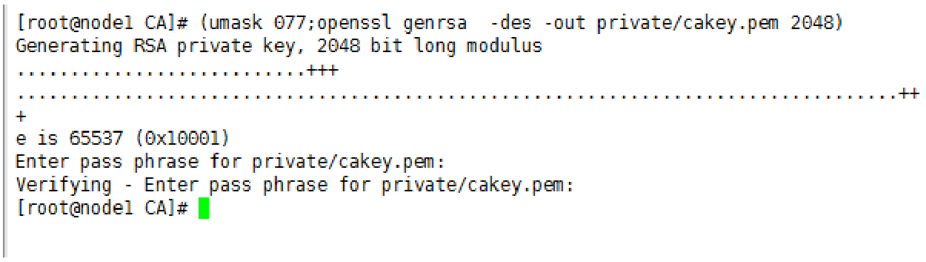
#touch /etc/pki/CA/index.txt

#echo 01 > /etc/pki/CA/serial 说明第一个证书的颁发编号从01开始

（3） CA自签证书：

#cd /etc/pki/CA

#(umask 077;openssl genrsa -des -out private/cakey.pem 2048)

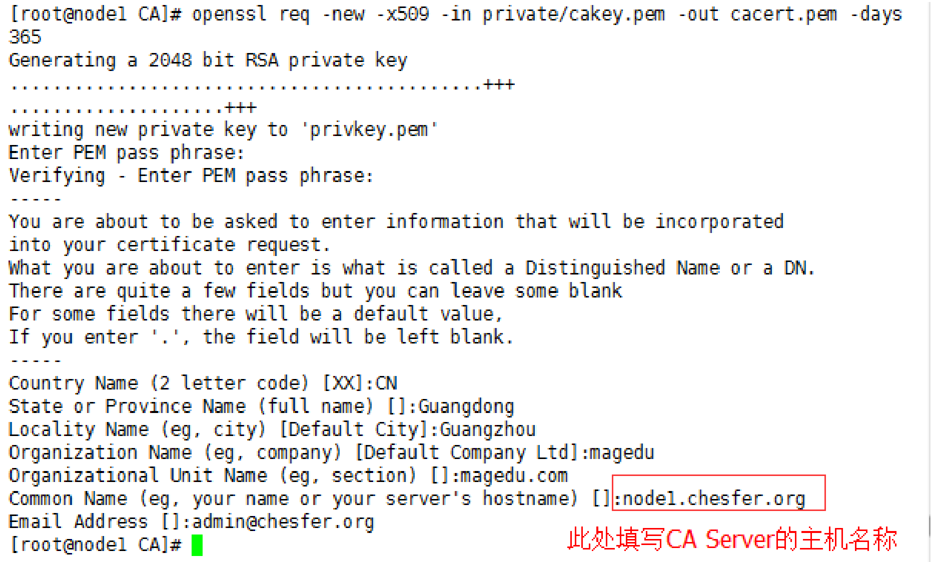


```
[root@node1 CA]# (umask 077;openssl genrsa  -des -out private/cakey.pem 2048)
Generating RSA private key, 2048 bit long modulus
...........................+++
.................................................................................++
+
e is 65537 (0x10001)
Enter pass phrase for private/cakey.pem:
Verifying - Enter pass phrase for private/cakey.pem:
[root@node1 CA]# 
```

blob.png

（4） 生成自签名证书：

# openssl req -new -x509 -in private/cakey.pem -out cacert.pem -days 365



```
[root@node1 CA]# openssl req -new -x509 -in private/cakey.pem -out cacert.pem -days
365
Generating a 2048 bit RSA private key
...........................................+++
...................+++
writing new private key to 'privkey.pem'
Enter PEM pass phrase:
Verifying - Enter PEM pass phrase:
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [XX]:CN
State or Province Name (full name) []:Guangdong
Locality Name (eg, city) [Default City]:Guangzhou
Organization Name (eg, company) [Default Company Ltd]:magedu
Organizational Unit Name (eg, section) []:magedu.com
Common Name (eg, your name or your server's hostname) []:node1.chesfer.org
Email Address []:admin@chesfer.org
[root@node1 CA]#
```

此处填写CA Server的主机名称

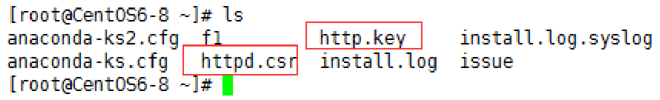
blob.png

（注：-new: 生成新证书签署请求；-x509: 专用于CA生成自签证书；-key: 生成请求时用到的私钥文件；-days n：证书的有效期限；-out /PATH/TO/SOMECERTFILE: 证书的保存路径）

（5） Client的证书请求和CA对Client的证书颁发：

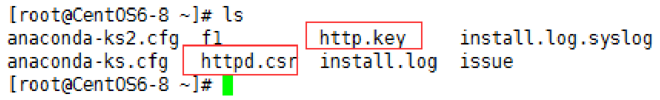
(umask 066;openssl genrsa -out http.key 2048)

openssl req -new -key http.key -days 365 -out httpd.csr



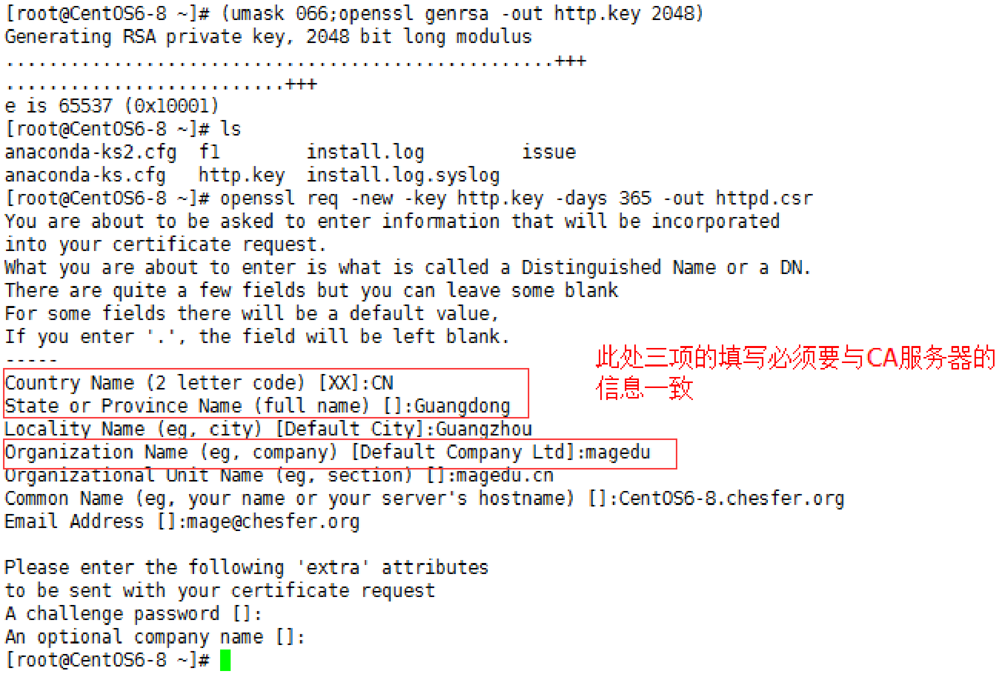
```
[root@CentOS6-8 ~]# ls
anaconda-ks2.cfg  f1         http.key     install.log.syslog
anaconda-ks.cfg   httpd.csr  install.log  issue
[root@CentOS6-8 ~]#
```

blob.png



```
[root@CentOS6-8 ~]# ls
anaconda-ks2.cfg  f1         http.key     install.log.syslog
anaconda-ks.cfg   httpd.csr  install.log  issue
[root@CentOS6-8 ~]#
```

blob.png



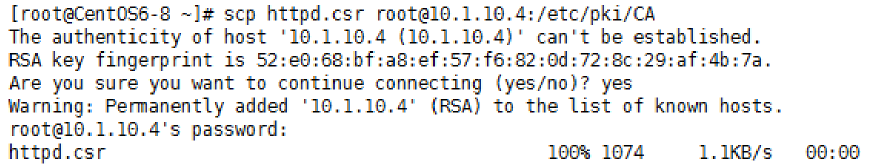
```
[root@CentOS6-8 ~]# (umask 066;openssl genrsa -out http.key 2048)
Generating RSA private key, 2048 bit long modulus
.................................................+++
..........................+++
e is 65537 (0x10001)
[root@CentOS6-8 ~]# ls
anaconda-ks2.cfg  f1        install.log         issue
anaconda-ks.cfg   http.key  install.log.syslog
[root@CentOS6-8 ~]# openssl req -new -key http.key -days 365 -out httpd.csr
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [XX]:CN
State or Province Name (full name) []:Guangdong
Locality Name (eg, city) [Default City]:Guangzhou
Organization Name (eg, company) [Default Company Ltd]:magedu
Organizational Unit Name (eg, section) []:magedu.cn
Common Name (eg, your name or your server's hostname) []:CentOS6-8.chesfer.org
Email Address []:mage@chesfer.org

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:
An optional company name []:
[root@CentOS6-8 ~]#
```

此处三项的填写必须要与CA服务器的信息一致

blob.png

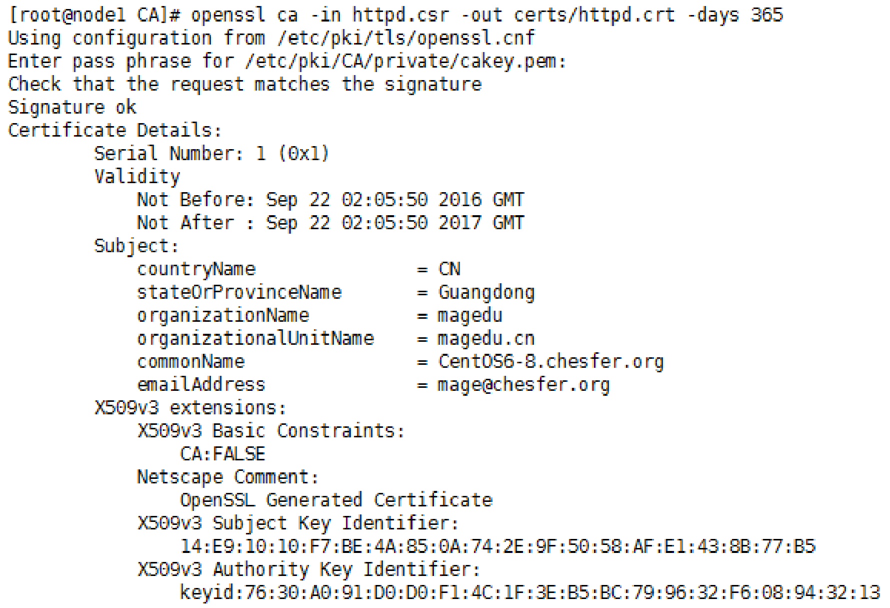
发送证书请求：



```
[root@CentOS6-8 ~]# scp httpd.csr root@10.1.10.4:/etc/pki/CA
The authenticity of host '10.1.10.4 (10.1.10.4)' can't be established.
RSA key fingerprint is 52:e0:68:bf:a8:ef:57:f6:82:0d:72:8c:29:af:4b:7a.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '10.1.10.4' (RSA) to the list of known hosts.
root@10.1.10.4's password:
httpd.csr                                     100% 1074     1.1KB/s   00:00
```

blob.png

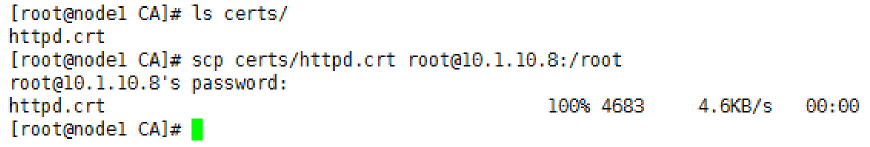
CA对Client端的证书请求进行签署：



```
[root@node1 CA]# openssl ca -in httpd.csr -out certs/httpd.crt -days 365
Using configuration from /etc/pki/tls/openssl.cnf
Enter pass phrase for /etc/pki/CA/private/cakey.pem:
Check that the request matches the signature
Signature ok
Certificate Details:
        Serial Number: 1 (0x1)
        Validity
            Not Before: Sep 22 02:05:50 2016 GMT
            Not After : Sep 22 02:05:50 2017 GMT
        Subject:
            countryName               = CN
            stateOrProvinceName       = Guangdong
            organizationName          = magedu
            organizationalUnitName    = magedu.cn
            commonName                = CentOS6-8.chesfer.org
            emailAddress              = mage@chesfer.org
        X509v3 extensions:
            X509v3 Basic Constraints:
                CA:FALSE
            Netscape Comment:
                OpenSSL Generated Certificate
            X509v3 Subject Key Identifier:
                14:E9:10:10:F7:BE:4A:85:0A:74:2E:9F:50:58:AF:E1:43:8B:77:B5
            X509v3 Authority Key Identifier:
                keyid:76:30:A0:91:D0:D0:F1:4C:1F:3E:B5:BC:79:96:32:F6:08:94:32:13
```

blob.png

将签署完后的证书返回给Client端：



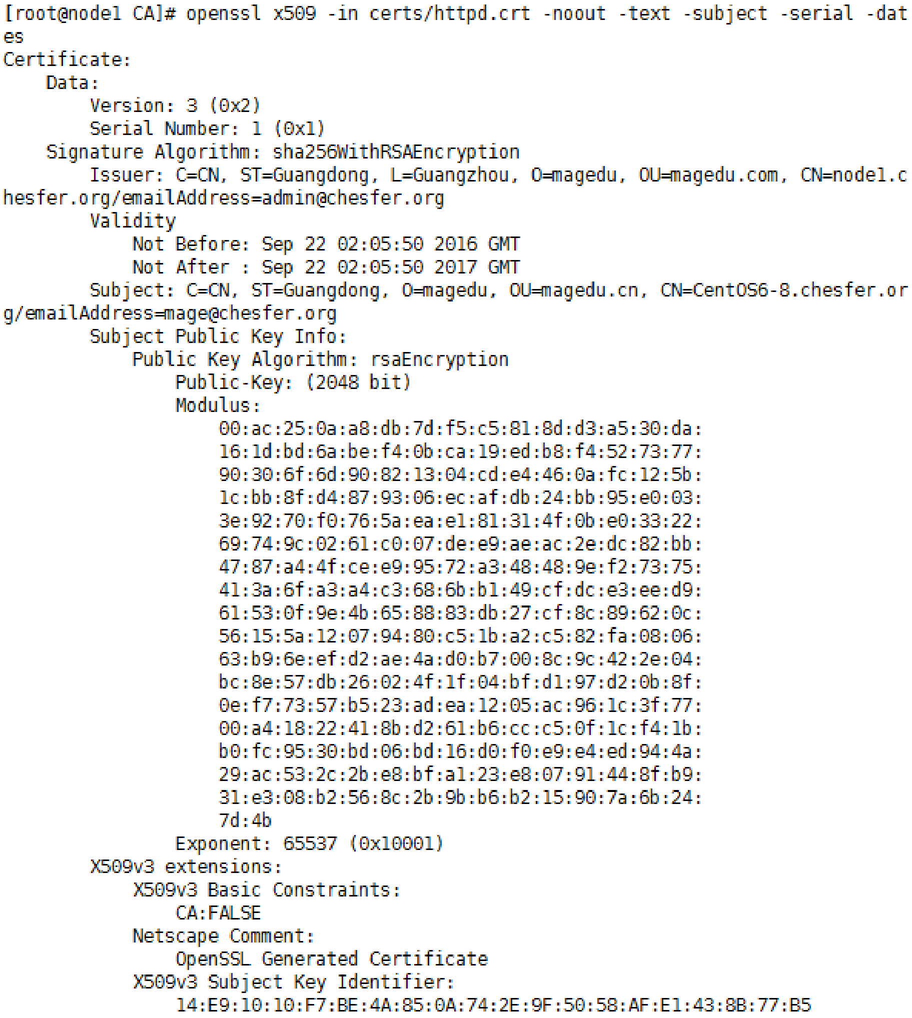
```
[root@node1 CA]# ls certs/
httpd.crt
[root@node1 CA]# scp certs/httpd.crt root@10.1.10.8:/root
root@10.1.10.8's password:
httpd.crt                                     100% 4683     4.6KB/s   00:00
[root@node1 CA]#
```

blob.png

2、证书的管理：

（1）证书的查看：

# openssl x509 -in certs/httpd.crt -noout -text -subject -serial -dates



```
[root@node1 CA]# openssl x509 -in certs/httpd.crt -noout -text -subject -serial -dat
es
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number: 1 (0x1)
    Signature Algorithm: sha256WithRSAEncryption
        Issuer: C=CN, ST=Guangdong, L=Guangzhou, O=magedu, OU=magedu.com, CN=node1.c
hesfer.org/emailAddress=admin@chesfer.org
        Validity
            Not Before: Sep 22 02:05:50 2016 GMT
            Not After : Sep 22 02:05:50 2017 GMT
        Subject: C=CN, ST=Guangdong, O=magedu, OU=magedu.cn, CN=CentOS6-8.chesfer.or
g/emailAddress=mage@chesfer.org
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
                Public-Key: (2048 bit)
                Modulus:
                    00:ac:25:0a:a8:db:7d:f5:c5:81:8d:d3:a5:30:da:
                    16:1d:bd:6a:be:f4:0b:ca:19:ed:b8:f4:52:73:77:
                    90:30:6f:6d:90:82:13:04:cd:e4:46:0a:fc:12:5b:
                    1c:bb:8f:d4:87:93:06:ec:af:db:24:bb:95:e0:03:
                    3e:92:70:f0:76:5a:ea:e1:81:31:4f:0b:e0:33:22:
                    69:74:9c:02:61:c0:07:de:e9:ae:ac:2e:dc:82:bb:
                    47:87:a4:4f:ce:e9:95:72:a3:48:48:9e:f2:73:75:
                    41:3a:6f:a3:a4:c3:68:6b:b1:49:cf:dc:e3:ee:d9:
                    61:53:0f:9e:4b:65:88:83:db:27:cf:8c:89:62:0c:
                    56:15:5a:12:07:94:80:c5:1b:a2:c5:82:fa:08:06:
                    63:b9:6e:ef:d2:ae:4a:d0:b7:00:8c:9c:42:2e:04:
                    bc:8e:57:db:26:02:4f:1f:04:bf:d1:97:d2:0b:8f:
                    0e:f7:73:57:b5:23:ad:ea:12:05:ac:96:1c:3f:77:
                    00:a4:18:22:41:8b:d2:61:b6:cc:c5:0f:1c:f4:1b:
                    b0:fc:95:30:bd:06:bd:16:d0:f0:e9:e4:ed:94:4a:
                    29:ac:53:2c:2b:e8:bf:a1:23:e8:07:91:44:8f:b9:
                    31:e3:08:b2:56:8c:2b:9b:b6:b2:15:90:7a:6b:24:
                    7d:4b
                Exponent: 65537 (0x10001)
        X509v3 extensions:
            X509v3 Basic Constraints:
                CA:FALSE
            Netscape Comment:
                OpenSSL Generated Certificate
            X509v3 Subject Key Identifier:
                14:E9:10:10:F7:BE:4A:85:0A:74:2E:9F:50:58:AF:E1:43:8B:77:B5
```

blob.png

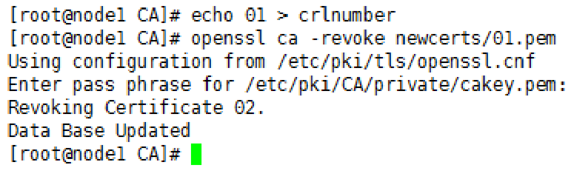
（2）证书的吊销：

openssl ca -revoke newcerts/01.pem

（注：01.pem为签署序列号为01的证书文件）

生成吊销证书的编号(第一次吊销一个证书时才需要执行)

echo 01 > /etc/pki/CA/crlnumber

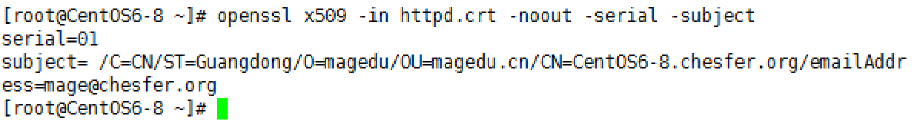


```
[root@node1 CA]# echo 01 > crlnumber
[root@node1 CA]# openssl ca -revoke newcerts/01.pem
Using configuration from /etc/pki/tls/openssl.cnf
Enter pass phrase for /etc/pki/CA/private/cakey.pem:
Revoking Certificate 02.
Data Base Updated
[root@node1 CA]#
```

blob.png

（3）Client端获取要吊销证书的serial:

#openssl x509 -in httpd.crt -noout -serial -subject

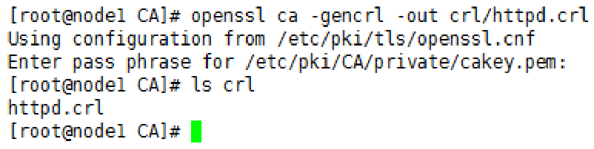


```
[root@CentOS6-8 ~]# openssl x509 -in httpd.crt -noout -serial -subject
serial=01
subject= /C=CN/ST=Guangdong/O=magedu/OU=magedu.cn/CN=CentOS6-8.chesfer.org/emailAddr
ess=mage@chesfer.org
[root@CentOS6-8 ~]#
```

blob.png

（4）更新证书的吊销列表：

openssl ca -gencrl -out crl/httpd.crl

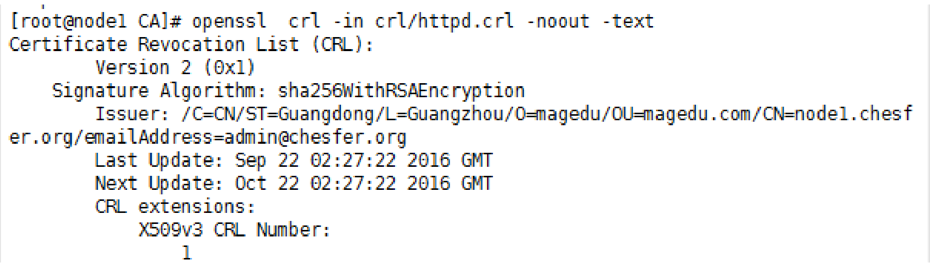


```
[root@node1 CA]# openssl ca -gencrl -out crl/httpd.crl
Using configuration from /etc/pki/tls/openssl.cnf
Enter pass phrase for /etc/pki/CA/private/cakey.pem:
[root@node1 CA]# ls crl
httpd.crl
[root@node1 CA]#
```

blob.png

（5）　查看CRL文件：

openssl crl -in crl/httpd.crl -noout -text



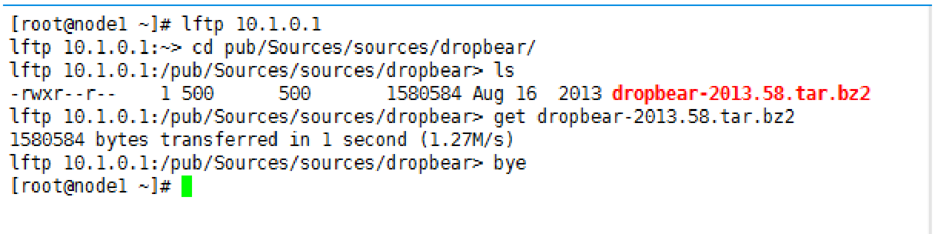
```
[root@node1 CA]# openssl  crl -in crl/httpd.crl -noout -text
Certificate Revocation List (CRL):
        Version 2 (0x1)
    Signature Algorithm: sha256WithRSAEncryption
        Issuer: /C=CN/ST=Guangdong/L=Guangzhou/O=magedu/OU=magedu.com/CN=node1.chesf
er.org/emailAddress=admin@chesfer.org
        Last Update: Sep 22 02:27:22 2016 GMT
        Next Update: Oct 22 02:27:22 2016 GMT
        CRL extensions:
            X509v3 CRL Number:
                1
```

blob.png

二、编译安装dropbear

（1）安装准备：1、安装开发包组:yum groupinstall “Development Tools”

2、ftp://172.16.0.1/pub/Sources/sources/dropbear/dropbear-2013.58.tar.bz2



```
[root@node1 ~]# lftp 10.1.0.1
lftp 10.1.0.1:~> cd pub/Sources/sources/dropbear/
lftp 10.1.0.1:/pub/Sources/sources/dropbear> ls
-rwxr--r--    1 500      500       1580584 Aug 16  2013 dropbear-2013.58.tar.bz2
lftp 10.1.0.1:/pub/Sources/sources/dropbear> get dropbear-2013.58.tar.bz2
1580584 bytes transferred in 1 second (1.27M/s)
lftp 10.1.0.1:/pub/Sources/sources/dropbear> bye
[root@node1 ~]# 
```

blob.png

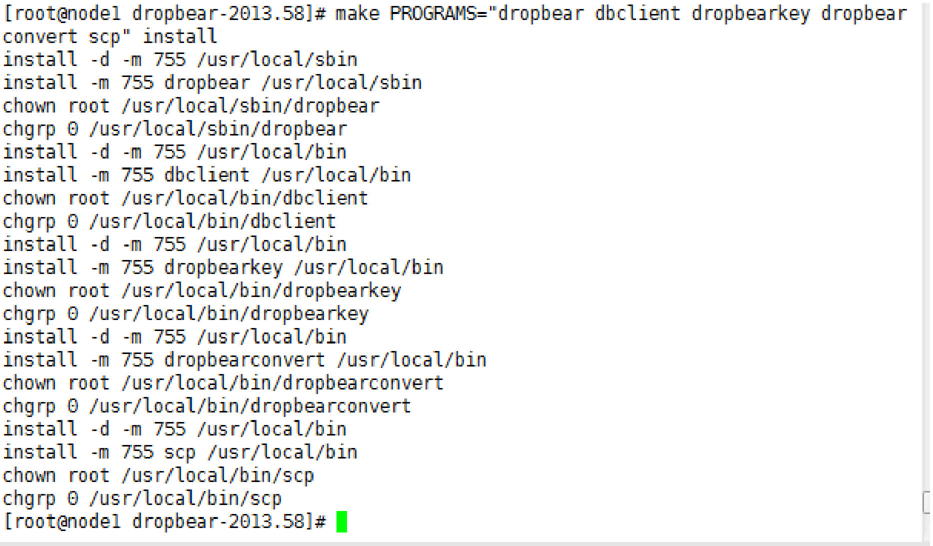
（2）安装：1、tar xf dropbear-2013.58.tar.bz2,

2、less INSTALL

3、./configure

4、make PROGRAMS="dropbear dbclient dropbearkey dropbearconvert scp"

5、make PROGRAMS="dropbear dbclient dropbearkey dropbearconvert scp" install



```
[root@node1 dropbear-2013.58]# make PROGRAMS="dropbear dbclient dropbearkey dropbear
convert scp" install
install -d -m 755 /usr/local/sbin
install -m 755 dropbear /usr/local/sbin
chown root /usr/local/sbin/dropbear
chgrp 0 /usr/local/sbin/dropbear
install -d -m 755 /usr/local/bin
install -m 755 dbclient /usr/local/bin
chown root /usr/local/bin/dbclient
chgrp 0 /usr/local/bin/dbclient
install -d -m 755 /usr/local/bin
install -m 755 dropbearkey /usr/local/bin
chown root /usr/local/bin/dropbearkey
chgrp 0 /usr/local/bin/dropbearkey
install -d -m 755 /usr/local/bin
install -m 755 dropbearconvert /usr/local/bin
chown root /usr/local/bin/dropbearconvert
chgrp 0 /usr/local/bin/dropbearconvert
install -d -m 755 /usr/local/bin
install -m 755 scp /usr/local/bin
chown root /usr/local/bin/scp
chgrp 0 /usr/local/bin/scp
[root@node1 dropbear-2013.58]# 
```

blob.png

（3）启动ssh服务：

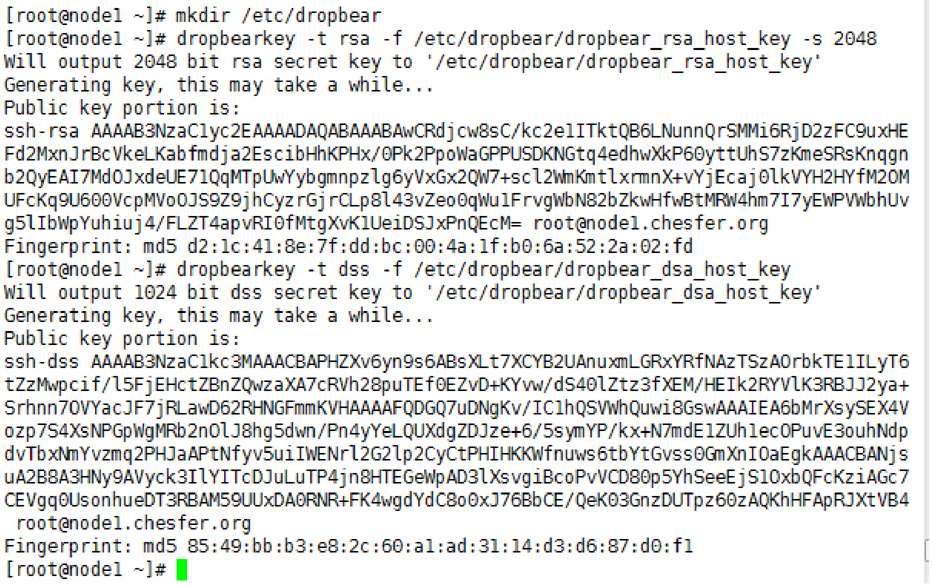
#ls /usr/local/sbin/ /usr/local/bin/

#/usr/local/sbin/dropbear -h

#mkdi r/etc/dropbear

#dropbearkey -t rsa -f /etc/dropbear/dropbear_rsa_host_key-s 2048

#dropbearkey -t dss -f /etc/dropbear/dropbear_dsa_host_key

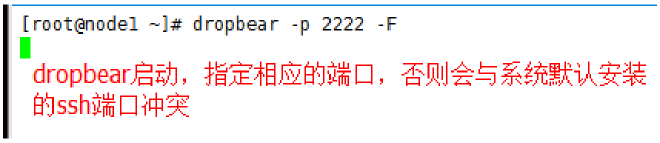


```
[root@node1 ~]# mkdir /etc/dropbear
[root@node1 ~]# dropbearkey -t rsa -f /etc/dropbear/dropbear_rsa_host_key -s 2048
Will output 2048 bit rsa secret key to '/etc/dropbear/dropbear_rsa_host_key'
Generating key, this may take a while...
Public key portion is:
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABAwCRdjcw8sC/kc2elITktQB6LNunnQrSMMi6RjD2zFC9uxHE
Fd2MxnJrBcVkeLKabfmdja2EscibHhKPHx/0Pk2PpoWaGPPUSDKNGtq4edhwXkP60yttUhS7zKmeSRsKnqgn
b2QyEAI7MdOJxdeUE71QqMTpUwYybgmnpzlg6yVxGx2QW7+scl2WmKmtlxrmnX+vYjEcaj0lkVYH2HYfM2OM
UFcKq9U600VcpMVoOJS9Z9jhCyzrGjrCLp8l43vZeo0qWu1FrvgWbN82bZkwHfwBtMRW4hm7I7yEWPVWbhUv
g5lIbWpYuhiuj4/FLZT4apvRI0fMtgXvK1UeiDSJxPnQEcM= root@node1.chesfer.org
Fingerprint: md5 d2:1c:41:8e:7f:dd:bc:00:4a:1f:b0:6a:52:2a:02:fd
[root@node1 ~]# dropbearkey -t dss -f /etc/dropbear/dropbear_dsa_host_key
Will output 1024 bit dss secret key to '/etc/dropbear/dropbear_dsa_host_key'
Generating key, this may take a while...
Public key portion is:
ssh-dss AAAAB3NzaC1kc3MAAACBAPHZXv6yn9s6ABsXLt7XCYB2UAnuxmLGRxYRfNAzTSzAOrbkTE1ILyT6
tZzMwpcif/l5FjEHctZBnZQwzaXA7cRVh28puTEf0EZvD+KYvw/dS40lZtz3fXEM/HEIk2RYVlK3RBJJ2ya+
Srhnn7OVYacJF7jRLawD62RHNGFmmKVHAAAAFQDGQ7uDNgKv/IC1hQSVWhQuwi8GswAAAIEA6bMrXsySEX4V
ozp7S4XsNPGpWgMRb2nOlJ8hg5dwn/Pn4yYeLQUXdgZDJze+6/5symYP/kx+N7mdE1ZUh1ecOPuvE3ouhNdp
dvTbxNmYvzmq2PHJaAPtNfyv5uiIWENrl2G2lp2CyCtPHIHKKWfnuws6tbYtGvss0GmXnIOaEgkAAACBANjs
uA2B8A3HNy9AVyck3IlYITcDJuLuTP4jn8HTEGeWpAD3lXsvgiBcoPvVCD80p5YhSeeEjS1OxbQFcKziAGc7
CEVgq0UsonhueDT3RBAM59UUxDA0RNR+FK4wgdYdC8o0xJ76BbCE/QeK03GnzDUTpz60zAQKhHFApRJXtVB4
 root@node1.chesfer.org
Fingerprint: md5 85:49:bb:b3:e8:2c:60:a1:ad:31:14:d3:d6:87:d0:f1
[root@node1 ~]# 
```

blob.png

#dropbear -p 2222 -F –E #前台运行

#dropbear -p 2222 #后台运行



```
[root@node1 ~]# dropbear -p 2222 -F
```

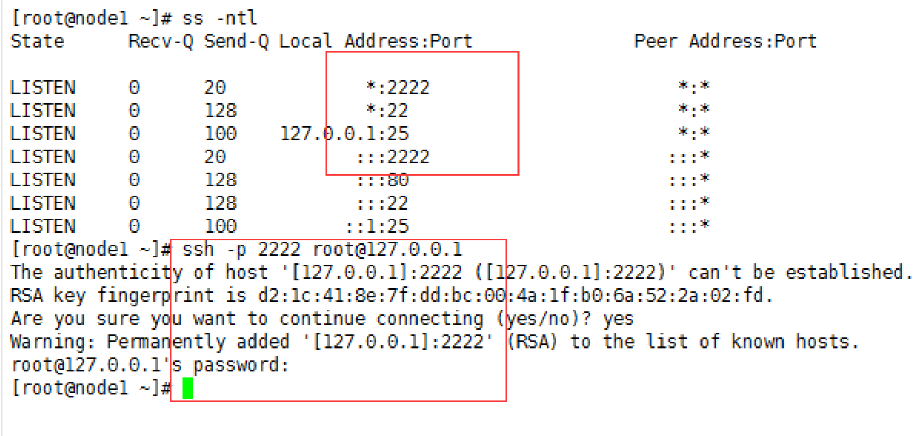
dropbear启动，指定相应的端口，否则会与系统默认安装的ssh端口冲突

blob.png

（4）客户端访问：

#ssh -p 2222 root@127.0.0.1

#dbclient -p 2222 root@127.0.0.1



```
[root@node1 ~]# ss -ntl
State      Recv-Q Send-Q Local Address:Port               Peer Address:Port

LISTEN     0      20             *:2222                       *:*
LISTEN     0      128            *:22                         *:*
LISTEN     0      100    127.0.0.1:25                         *:*
LISTEN     0      20            :::2222                      :::*
LISTEN     0      128           :::80                        :::*
LISTEN     0      128           :::22                        :::*
LISTEN     0      100          ::1:25                        :::*
[root@node1 ~]# ssh -p 2222 root@127.0.0.1
The authenticity of host '[127.0.0.1]:2222 ([127.0.0.1]:2222)' can't be established.
RSA key fingerprint is d2:1c:41:8e:7f:dd:bc:00:4a:1f:b0:6a:52:2a:02:fd.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '[127.0.0.1]:2222' (RSA) to the list of known hosts.
root@127.0.0.1's password:
[root@node1 ~]# 
```

blob.png