PKI公钥认证体系结构

PKI ---- Public Key Infrastructure

计算机网络提供数据传输的功能

保证数据安全性：

1、安全性

加密算法

对称加密算法

加密、解密数据时使用的密钥是一样的

DES, 3DES, AES

加密数据：

[root@node\_01 ~]# openssl enc -e -des -in /etc/passwd -out /tmp/passwd\_1

enter des-cbc encryption password:

Verifying - enter des-cbc encryption password:

解密数据：

[root@node\_01 ~]# openssl enc -d -des -in /tmp/passwd\_1

enter des-cbc decryption password:

非对称加密算法/公钥加密

加密、解密数据时使用的密钥是不一样的

密钥对：公钥、私钥

公钥加密、私钥解密数据

代表性算法：

DSA, RSA

gpg非对称加密

# useradd user1

# useradd user2

使用user1登录系统

# gpg --gen-key >>>生成密钥对

查看公钥、私钥

[user1@node1 ~]$ gpg --list-keys

/home/user1/.gnupg/pubring.gpg

------------------------------

pub 2048R/D339A512 2017-07-12

uid user1 <user1@localhost>

sub 2048R/A9273739 2017-07-12

[user1@node1 ~]$ gpg --list-secret-keys

/home/user1/.gnupg/secring.gpg

------------------------------

sec 2048R/D339A512 2017-07-12

uid user1 <user1@localhost>

ssb 2048R/A9273739 2017-07-12

[user1@node1 ~]$

将user1公钥导出

[user1@node1 ~]$ gpg --export --armor user1 > /tmp/user1.key

user2导入user1的公钥

[user2@node1 ~]$ gpg --import /tmp/user1.key

[user2@node1 ~]$ gpg --list-keys

user2使用该公钥加密数据

[user2@node1 ~]$ gpg --encrypt --armor --recipient user1 文件名称

[user2@node1 ~]$ cp 加密文件， user1

user1解密数据

[user1@node1 ~]$ gpg 加密文件名称

数据加密算法的应用：

使用对称加密算法加密真实的数据，使用非对称加密算法加密对称加密算法的密钥

2、数据完整性

通过哈希算法计算数据的校验码

哈希hash算法：

MD5

SHA

密钥对的作用：

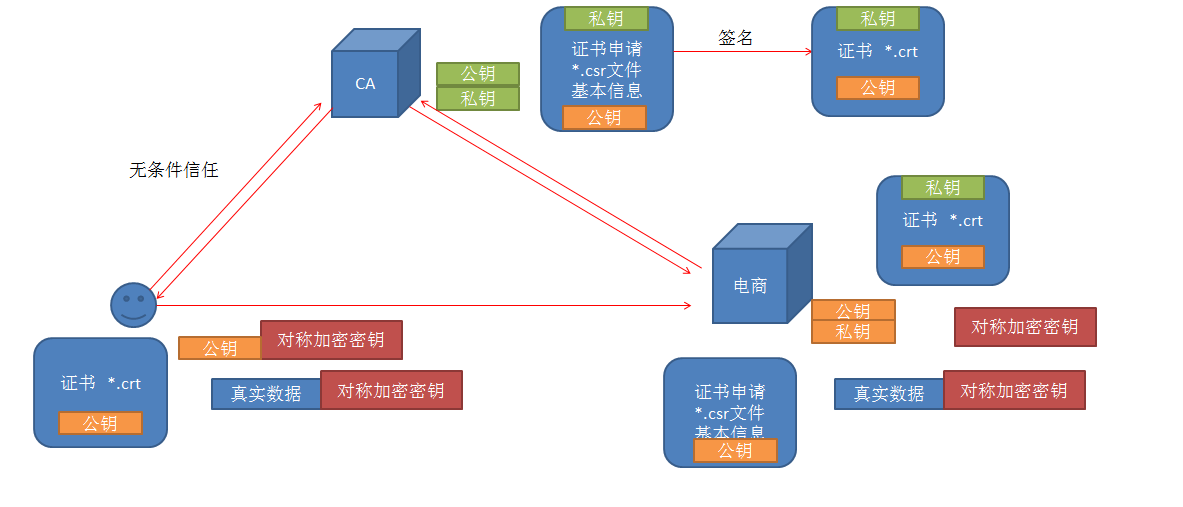
1、加密

公钥加密、私钥解密

2、签名

私钥签名、公钥验证签名

PKI体系结构：



说明：

1、电商有证书申请，放入自己的公钥，向CA提交申请

2、CA使用自己的私钥签名，签发证书

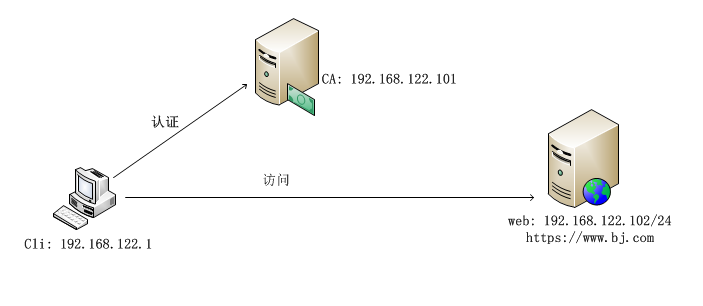
3、客户端访问电商时，电商出示自己的证书，客户端向CA验证证书的真实性；CA通过公钥验证签名

4、客户端生成一个对称加密算法的密钥，使用电商的公钥加密密钥，向电商发送密钥

5、电商收到加密密钥后，使用私钥解密，获取了客户端的对称加密算法的密钥

6、使用对称加密算法的密钥加、解密数据

**部署私有CA实现https加密虚拟主机**



部署私有CA ------ openssl   
1、生成密钥对(公钥, 私钥)  
2、签发自签证书  
  
部署https网站   
  
1、生成证书申请、向CA发送证书申请  
2、CA签发证书  
3、mod\_ssl模块

**一、部署私有CA**

**1、**创建索引文件及证书编号文件

**[root@ca\_server ~]# touch /etc/pki/CA/index.txt**

**[root@ca\_server ~]# echo 01 > /etc/pki/CA/serial**

**2、创建密钥对**

**[root@ca\_server ~]# cd /etc/pki/CA/**

**[root@ca\_server CA]# openssl genrsa -out private/cakey.pem 2048**

**Generating RSA private key, 2048 bit long modulus**

**............................................+++**

**...........................................................................................+++**

**e is 65537 (0x10001)**

**[root@ca\_server CA]#**

**[root@ca\_server CA]# chown 600 /etc/pki/CA/private/cakey.pem**

**3、生成CA的自签证书**

**[root@ca\_server ~]# openssl req -new -x509 -key /etc/pki/CA/private/cakey.pem -out /etc/pki/CA/cacert.pem -days 3650**

**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) []:cn**

**Locality Name (eg, city) [Default City]:bj**

**Organization Name (eg, company) [Default Company Ltd]:uplooking**

**Organizational Unit Name (eg, section) []:uplooking**

**Common Name (eg, your name or your server's hostname) []:ca\_server**

**Email Address []:uplooking@qq.com**

**二、配置web服务器申请证书，实现https网站**

**1、安装http、mod\_ssl模块**

**[root@web\_server ~]# yum install -y httpd mod\_ssl**

2、生成web服务器的密钥对

[root@web\_server ~]# mkdir /etc/httpd/ssl

[root@web\_server ~]# openssl genrsa -out /etc/httpd/ssl/httpd.key 2048

Generating RSA private key, 2048 bit long modulus

............................................................................+++

............................+++

e is 65537 (0x10001)

[root@web\_server ~]# chmod 600 /etc/httpd/ssl/httpd.key

3、生成证书申请，将申请发送给CA

[root@web\_server ~]# openssl req -new -key /etc/httpd/ssl/httpd.key -out /etc/httpd/ssl/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) []:cn

Locality Name (eg, city) [Default City]:bj

Organization Name (eg, company) [Default Company Ltd]:uplooking

Organizational Unit Name (eg, section) []:uplooking

Common Name (eg, your name or your server's hostname) []:www.bj.com

Email Address []:uplooking@qq.com

Please enter the following 'extra' attributes

to be sent with your certificate request

A challenge password []:

An optional company name []:

[root@web\_server ~]# rsync -av /etc/httpd/ssl/httpd.csr 192.168.122.101:/tmp/

4、在CA上颁发证书

[root@ca\_server ~]# openssl ca -in /tmp/httpd.csr -out /etc/pki/CA/certs/httpd.crt -days 3650

Using configuration from /etc/pki/tls/openssl.cnf

Check that the request matches the signature

Signature ok

Certificate Details:

Serial Number: 1 (0x1)

Validity

Not Before: Sep 14 04:31:21 2018 GMT

Not After : Sep 11 04:31:21 2028 GMT

Subject:

countryName = cn

stateOrProvinceName = cn

organizationName = uplooking

organizationalUnitName = uplooking

commonName = www.bj.com

emailAddress = uplooking@qq.com

X509v3 extensions:

X509v3 Basic Constraints:

CA:FALSE

Netscape Comment:

OpenSSL Generated Certificate

X509v3 Subject Key Identifier:

2E:50:B6:05:CF:7D:20:30:EF:92:7A:09:50:58:6B:98:FD:64:C9:C0

X509v3 Authority Key Identifier:

keyid:62:8F:4B:21:F4:46:C0:C2:6C:19:FF:4B:EB:EB:8C:1B:19:82:82:63

Certificate is to be certified until Sep 11 04:31:21 2028 GMT (3650 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

[root@ca\_server ~]# ls /etc/pki/CA/certs/

httpd.crt

[root@ca\_server ~]# cd /etc/pki/CA/

[root@ca\_server CA]# ls

cacert.pem certs crl index.txt index.txt.attr index.txt.old newcerts private serial serial.old

[root@ca\_server CA]# cat serial

02

[root@ca\_server CA]# cat index.txt

V 280911043121Z 01 unknown /C=cn/ST=cn/O=uplooking/OU=uplooking/CN=www.bj.com/emailAddress=uplooking@qq.com

[root@ca\_server CA]# rsync -av /etc/pki/CA/certs/httpd.crt 192.168.122.102:/etc/httpd/ssl

5、配置ssl的虚拟主机

[root@web\_server conf.d]# vim /etc/httpd/conf.d/ssl.conf

56 <VirtualHost \_default\_:443>

57

59 DocumentRoot "/bj"

60 ServerName www.bj.com:443

100 SSLCertificateFile /etc/httpd/ssl/httpd.crt

107 SSLCertificateKeyFile /etc/httpd/ssl/httpd.key

[root@web\_server conf.d]# netstat -antp | grep httpd

tcp6 0 0 :::80 :::\* LISTEN 1301/httpd

tcp6 0 0 :::443 :::\* LISTEN 1301/httpd

在物理机测试访问

<https://www.bj.com/>

