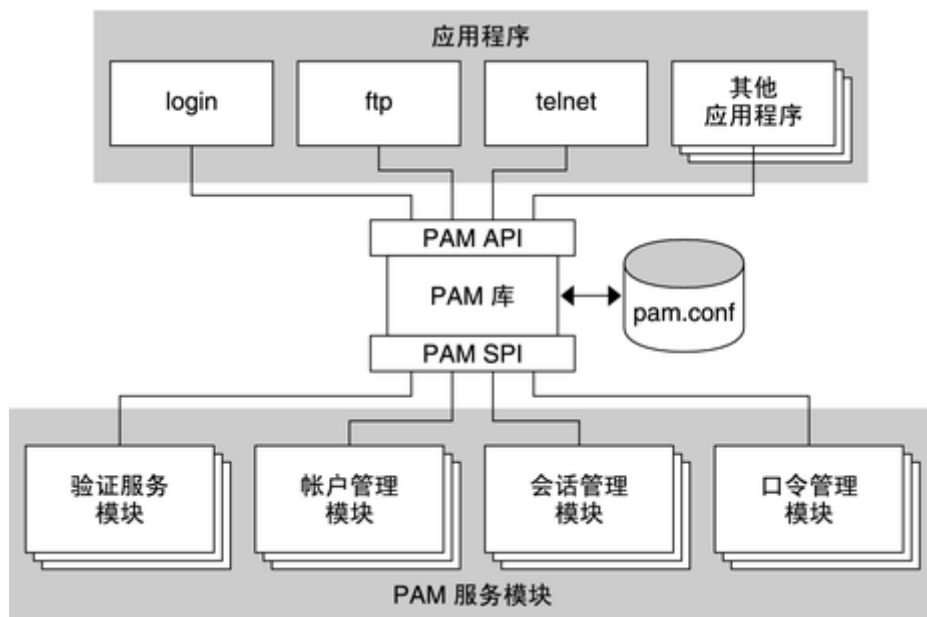


自建CA配置HTTPS



OpenSSL

OpenSSL: Secure Socket Layer ([安全套接层协议](#))。是一个开放源代码的软件库包，应用程序可以使用这个包来进行安全通信，避免窃听，同时确认另一端连接者的身份。这个包广泛被应用在互联网的网页服务器上。是SSL和TLS协议的实现。SSL协议处于应用层与传输之间。

OpenSSL由三部分组成：

1. libcrypto：通用加密库
2. libssl：用于实现TLS/SSL的功能
3. openssl：多功能命令行工具。可以实现私有证书颁发机构。功能：生成密钥、创建数字证书、手动加密解密数据

OenSSL应用

```
1 #使用DES算法加密文件
2 openssl enc -des3 -salt -a -in Sou.file -out
  Pur.file.des3
3 #解密
4 openssl enc -des3 -d -salt -a -in Pur.file.des3 -out
  Sou.file
```

```
1 #计算文件特征码
2 openssl dgst [-md5|-md4|-md2|-sha1|-sha|-mdc2|-
  ripemd160|-dss1] [-out filename] Sou.file
```

```
1 #生成随机数
2 openssl rand -base64|-hex 长度
```

OpenSSL配置文件

/etc/pki/tls/openssl.cnf

```
1 [ CA_default ]
2 dir                = /etc/pki/CA                # CA工作目录
3 certs              = $dir/certs                 # 证书目录
4 crl_dir            = $dir/crl                   # 吊销列表
5 database           = $dir/index.txt             # 索引文件数
   数据库
6 #unique_subject    = no                         #
7 new_certs_dir      = $dir/newcerts              # 刚签署的证
   书放置目录
8 certificate        = $dir/cacert.pem            # CA自签署证
   书放置目录
9 serial             = $dir/serial                 # 证书序列
   号，下一个证书的序列号
10 crlnumber          = $dir/crlnumber             # 吊销证书序
   列号
11 crl                = $dir/crl.pem              # 当前正在使
   用吊销列表文件
12 private_key        = $dir/private/cakey.pem     # CA的私钥
13 RANDFILE           = $dir/private/.rand        # 随机数文件
```

CA

数字证书认证机构CA (Certificate Authority) 。

关于HTTPS及CA参考: <http://netsmell.com/post/maybe-it-is-easy-to-understand-https-thus.html>

基于OpenSSL自建CA

1.创建必要文件

```
1 cd /etc/pki/CA/
2
3 #创建索引文件数据库index.txt
4 #创建证书序列号，下一个证书的序列号serial
5 touch {index.txt,serial}
6
7 #写入其实序列号
8 echo 01 > serial
```

2.创建CA

```
1 #生成私钥
2 (umask 066; openssl genrsa -out
   /etc/pki/CA/private/cakey.pem 2048)
3 #生成自签署证书
4 openssl req -new -x509 -key
   /etc/pki/CA/private/cakey.pem -out cacert.pem -days
   365
5 #查看生成的证书
6 openssl x509 -text -in /etc/pki/CA/cacert.pem
```

Apache配置自签署证书

1.安装ssl_mod模块

```
1 yum install mod_ssl
```

ssl_mod默认配置文件是/etc/httpd/conf.d/ssl.conf

2.申请证书

```
1 #创建ssl目录
2 mkdir /etc/httpd/ssl
3
```

```
4 #生成私钥。将私钥保存至使用此证书的应用服务的配置文件目录下
5 (umask 077;openssl genrsa -out
  /etc/httpd/ssl/httpd.key 2048)
6
7 #生成证书申请文件
8 openssl req -new -key /etc/httpd/ssl/httpd.key -out
  /etc/httpd/ssl/httpd.csr
9
10 #客户将申请文件发给认证中心。由于是私有CA所有，这一步不需要
    发送
11
12 #认证中心签署证书
13 openssl ca -in /etc/httpd/ssl/httpd.csr -out
  /etc/pki/CA/certs/httpd.crt -days 365
14
15 #认证中心将证书httpd.crt发给申请的客户，由于是私有CA，所有
    可以将证书复制到应用的目录下
16 cp -p /etc/pki/CA/certs/httpd.crt /etc/httpd/ssl
```

3.Apache配置

```
1 Listen 443
2 <VirtualHost *:443>
3     DocumentRoot "/var/www/html"
4     ServerName www.xuejinwei.cc
5     #SSL引擎操作开关
6     SSLEngine on
7     #指定服务器证书位置
8     SSLCertificateFile "/etc/httpd/ssl/httpd.crt"
9     #服务器私钥文件
10    SSLCertificateKeyFile "/etc/httpd/ssl/httpd.key"
11 </VirtualHost>
```

4.实现自动跳转https

```
1 <VirtualHost *:80>
2     DocumentRoot "/var/www/html"
3     ServerName www.xuejinwei.cc
4     RewriteEngine on
5     RewriteCond %{SERVER_PORT} !^443$
6     RewriteRule ^(.*)?$ https://%{SERVER_NAME}/$1
    [L,R]
7 </VirtualHost>
```

5.只针对某个目录跳转

```
1 RewriteEngine on
2 RewriteBase /yourfolder
3 RewriteCond %{SERVER_PORT} !^443$
4 #RewriteRule ^(.*)?$ https://%{SERVER_NAME}/$1 [L,R]
5 RewriteRule ^.*$ https://%{SERVER_NAME}%{REQUEST_URI}
    [L,R]
```