



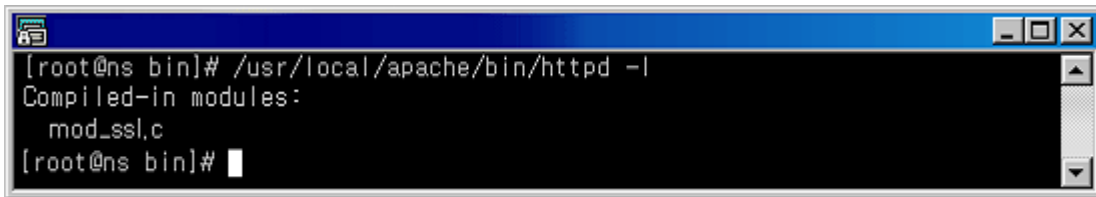
- ⑧ 이제 SSL 인증서의 설치가 완료되었습니다. VI장으로 이동하셔서 실제 웹페이지를 어떻게 수정해야 하는지 알아보겠습니다.

## 2.2 Apache 서버에서 보안서버 구축하기

### 가. Apache 서버에 OpenSSL과 mod\_ssl의 설치 방법

Apache 서버에서 SSL 통신을 가능하게 하기 위해서는 OpenSSL과 mod\_ssl이 필요합니다.

우선, 현재 서비스 중인 Apache 서버에 mod\_ssl이 설치되어 있는지를 `httpd -l` 옵션을 사용하여 `mod_ssl.c` 또는 `mod_ssl.so`가 있는지 확인하시기 바랍니다. 만일 설치되어 있다면 Apache 서버의 버전에 맞는 개인키 생성 및 CSR 생성 방법 과정으로 이동하시기 바랍니다.



```
[root@ns bin]# /usr/local/apache/bin/httpd -l
Compiled-in modules:
  mod_ssl.c
[root@ns bin]#
```

<그림 3-3> mod\_ssl 설치 확인 예

OpenSSL은 Apache 버전과 mod\_ssl의 버전을 확인한 후에 알맞은 OpenSSL을 설치해야 합니다. 예를 들어 Apache 1.3.3 버전에는 mod\_ssl 2.1.6 (또는 2.1.7)을 설치해야 하고, mod\_ssl 2.1.6은 OpenSSL 0.8.1b와 0.9.1c 버전 사이에서만 동작합니다. 버전을 확인하지 않고 OpenSSL과 mod\_ssl을 설치하면 Apache 컴파일 과정에서 오류가 발생합니다.

mod\_ssl은 반드시 Apache 버전에 맞는 것을 설치하셔야 하며 [www.modssl.org](http://www.modssl.org)에서 Apache 버전을 확인한 후 그에 맞는 mod\_ssl을 다운받아 설치하시기 바랍니다.

mod\_ssl에서 지원하는 apache 버전 및 OpenSSL의 버전은 mod\_ssl 소스의 README.Versions에서 확인할 수 있으며, [www.openssl.org](http://www.openssl.org)에서도 확인할 수 있습니다.

### ① OpenSSL의 설치([www.openssl.org](http://www.openssl.org))

#### 압축풀기

```
$ gzip -cd openssl-0.9.6.tar.gz | tar xvf -
```

```
$ ./config$ make$ make installconfig
```

☞ prefix를 주지 않았을 때에는 /usr/local/ssl 디렉토리에 설치가 됩니다.

다른 디렉토리에 설치를 하고자 한다면 다음과 같이 합니다.

```
$ ./config --prefix=/usr/local --openssldir=/usr/local/openssl
```

☞ OpenSSL의 실행파일은 /usr/local/ssl/bin에 설치되고

인증서비스를 위한 파일들은 /usr/local/openssl 아래의 디렉토리에 생성됩니다.

② mod\_ssl의 설치 (www.modssl.org)

압출풀기

```
$ gzip -cd apache_1.3.19.tar.gz | tar xvf
$ gzip -cd mod_ssl-2.8.1-1.3.19.tar.gz | tar xvf
```

파일의 다운로드와 압축풀기가 끝나면 mod\_ssl 설정을 합니다.

mod\_ssl 설정

```
$ cd mod_ssl-2.8.1-1.3.19
$ ./configure \
--with-apache=../apache_1.3.19 \
--with-ssl=../openssl-0.9.6 \
--prefix=/usr/local/apache
```

③ Apache 서버 설치(www.apache.org )

```
$ cd ../apache_1.3.x
$ SSL_BASE=../openssl-0.9.6 \
./configure \
--prefix=/usr/local/apache \
--enable-module=ssl \
$ make
$ make certificate
$ make install
```

나. Apache 1.3.X 버전에서 보안서버 구축하기

(1) 개인키 생성 및 CSR 생성 방법

## ① 랜덤 넘버 생성

```
$ openssl md5 * > rand.dat
```

## ② 키 쌍 생성

```
$ openssl genrsa -rand rand.dat -des3 -out 1024 > key.pem
```

☞ 개인키 비밀번호를 입력하며 반드시 기억해야 합니다. (암호를 분실할 경우 SSL 사용을 위한 apache를 구동할 수 없습니다)

## ③ 생성된 키 쌍을 이용하여 CSR 생성

```
$ openssl req -new -key key.pem > csr.pem
```

☞ 여기서 key.pem은 단계 ②에서 생성한 키 이름이며 csr.pem은 출력 CSR 파일의 이름입니다.

다음 정보를 입력하라는 메시지가 나타납니다. (모든 내용은 영문으로 작성해야 하며, 아래는 입력 예입니다)

```
Country(국가 코드) KR
State/province (시/도의 전체 이름) Seoul
Locality(시,구,군 등의 이름) : Songpa-gu
Organization(회사 이름) : Korea Information Security Agency
Organization Unit(부서명) : Policy Development Division
Common Name (host name + domain name) : www.kisa.or.kr
```

"추가 속성"을 입력하라는 메시지가 나타나면 그냥 넘어가셔도 무방합니다.

## ④ CSR 제출

생성된 CSR(예:csr.pem)의 내용은 다음과 같습니다.

```

-----BEGIN CERTIFICATE REQUEST-----
MIIBETCBvAIBADBXMqswCQYDVQQGEwJBVTETMBEGA1UECBMKU29tZS1TdGF0ZTEh
MB8GA1UEChMYSW50ZXJuZXQgV2lkZ2l0cyBQdHkgTHRkMRAwDgYJKoZIhvcNAQkB
FgFgMFwwDQYJKoZIhvcNAQEBBQADSwAwSAJBAL6nPTy3avNgbubx+ESmD4LV1LQG
fcSh8nehEOIxGwmCPIrhTP87PaA0XvGpvRQUjCGStrlQsd8lcYVVkOaytNUCAwEA
AaAAMA0GCSqGSIb3DQEBAUAA0EAXcMsa8eXgbG2ZhVyFkRVrI4vT8haN39/QJc9
BrRh2nOTKgfMcT9h+1Xx0wNRQ9/SIGV1y3+3abNiJmJBWnJ8Bg==
-----END CERTIFICATE REQUEST-----

```

CSR 내용을 인증기관에게 메일로 송부하거나 인증서 신청화면에 붙여 넣으신 후 인증서 신청을 진행하시면 됩니다.  
인증기관의 발급 절차에 따라서 인증서가 발급됩니다.

## (2) 인증서 설치 방법

- ① 메일로 받은 인증서를 복사하여 파일로 저장합니다.(예: Cert.pem)

```

-----BEGIN CERTIFICATE-----
MIIBETCBvAIBADBXMqswCQYDVQQGEwJBVTETMBEGA1UECBMKU29tZS1TdGF0ZTEh
TEhMB8GA1UEChMYSW50ZXJuZXQgV2lkZ2l0cyBQdHkgTHRkMRAwDgYJKoZIhvcNA
QkBFgFgMFwwDQYJKoZIhvcNAQEBBQADSwAwSAJBAL6nPTy3avNgbubx+ESmD4LV
1LQGfcSh8nehEOIxGwmCPIrhTP87PaA0XvGpvRQUjCGStrlQsd8lcYVVkOaytNUCAwE
AAaAAMA0GCSqGSIb3DQEBAUAA0EAXcMsa8eXgbG2ZhVyFkRVrI4vT8haN39/QJc9
BrRh2nOTKgfMcT9h+1Xx0wNRQ9/SIGV1y3+3abNiJmJBWnJ8Bg==
-----END CERTIFICATE-----

```

- ② [Apache 서버](#)의 적절한 위치에 저장합니다.

- ③ 환경설정 파일(httpd.conf 또는 ssl.conf)을 수정합니다.

```
<VirtualHost _default_:443>

# General setup for the virtual host
DocumentRoot /Apache/htdocs
ServerName www.kisa.or.kr:443
ServerAdmin admin@kisa.or.kr
ErrorLog logs/error_log
TransferLog logs/access_log
SSLCertificateFile /Apache/ssl/cert.pem → 인증서 파일 경로
SSLCertificateKeyFile /Apache/ssl/key.pem → 개인키 파일 경로
```

- ④ Apache 서버를 재구동합니다.

```
./apachectl startssl
```

Apache 서버에서 SSL을 사용하기 위한 시작 명령어인 startssl을 실행하면 개인키의 비밀번호를 묻는데, 이 비밀번호는 이전의 설치과정 '개인키 생성 및 CSR 생성 방법' 중 ② 키 쌍 생성시 입력한 개인키 비밀번호를 입력하시면 됩니다.

- ⑤ 이제 SSL 인증서의 설치가 완료되었습니다. VI장으로 이동하셔서 실제 웹페이지를 어떻게 수정해야 하는지 알아보겠습니다.

## 다. Apache 2.X 버전에서 보안서버 구축하기

### (1) 개인키 생성 및 CSR 생성 방법

#### ① 랜덤 넘버 생성

개인키 생성시 사용할 랜덤 정보를 생성합니다. 생성된 rand.dat 파일이 중요하지 않다고 판단될 때, 언제든지 이 파일을 삭제, 변경할 수 있습니다.

```
$ openssl sha1 * > rand.dat
또는
$ cat file1 file2 file3 > rand.dat
```

## ② 키 쌍 생성

```
$ openssl genrsa -rand rand.dat -des 1024 > key.pem
```

- ☞ 개인키 비밀번호를 입력하며 반드시 기억해야 합니다. (암호를 분실할 경우 SSL 사용을 위한 apache를 구동할 수 없습니다)
- ☞ 개인키를 분실하신 경우, 백업된 개인키를 사용해야 하므로, 생성한 개인키의 백업복사본은 별도의 저장매체에 보관하여 주시기 바랍니다.

## ③ 생성된 키 쌍을 이용하여 CSR 생성

```
$ openssl req -new -key key.pem -out csr.pem
```

- ☞ 여기서 key.pem은 단계 ②에서 생성한 키 이름이며 csr.pem은 출력 CSR 파일의 이름입니다.

다음 정보를 입력하라는 메시지가 나타납니다. (모든 내용은 영문으로 작성해야 하며, 아래는 입력 예입니다)

```
Country(국가 코드) : KR
State/province (시/도의 전체 이름) : Seoul
Locality(시,구,군 등의 이름) : Songpa-gu
Organization(회사 이름) : Korea Information Security Agency
Organization Unit(부서명) : Development Division
Common Name (host name + domain name) : www.kisa.or.kr
```

"추가 속성"을 입력하라는 메시지가 나타나면 그냥 넘어가셔도 무방합니다.

## ④ CSR 제출

생성된 CSR(예:csr.pem)의 내용은 다음과 같습니다.

```

-----BEGIN CERTIFICATE REQUEST-----
MIIBETCBvAIBADBXMqswCQYDVQQGEwJBVTETMBEGA1UECBMKU29tZS1TdGF0ZTEh
MB8GA1UEChMYSW50ZXJuZXQgV2lkZ2l0cyBQdHkgTHRkMRAwDgYJKoZIhvcNAQkB
FgFgMFwwDQYJKoZIhvcNAQEBBQADSwAwSAJBAL6nPTy3avNgbubx+ESmD4LV1LQG
fcSh8nehEOIxGwmCPIrhTP87PaA0XvGpvRQUjCGStrlQsd8lcYVVkOaytNUCAwEA
AaAAMA0GCSqGSIb3DQEBAUAA0EAXcMsa8eXgbG2ZhVyFkRVrI4vT8haN39/QJc9
BrRh2nOTKgfMcT9h+1Xx0wNRQ9/SIGV1y3+3abNiJmJBWnJ8Bg==
-----END CERTIFICATE REQUEST-----

```

CSR 내용을 인증기관에게 메일로 송부하거나 인증서 신청화면에 붙여 넣으신 후 인증서 신청을 진행하시면 됩니다.  
인증기관의 발급 절차에 따라서 인증서가 발급됩니다.

## (2) 인증서 설치 방법

### ① 메일로 받은 인증서를 복사하여 파일로 저장합니다.(예: Cert.pem)

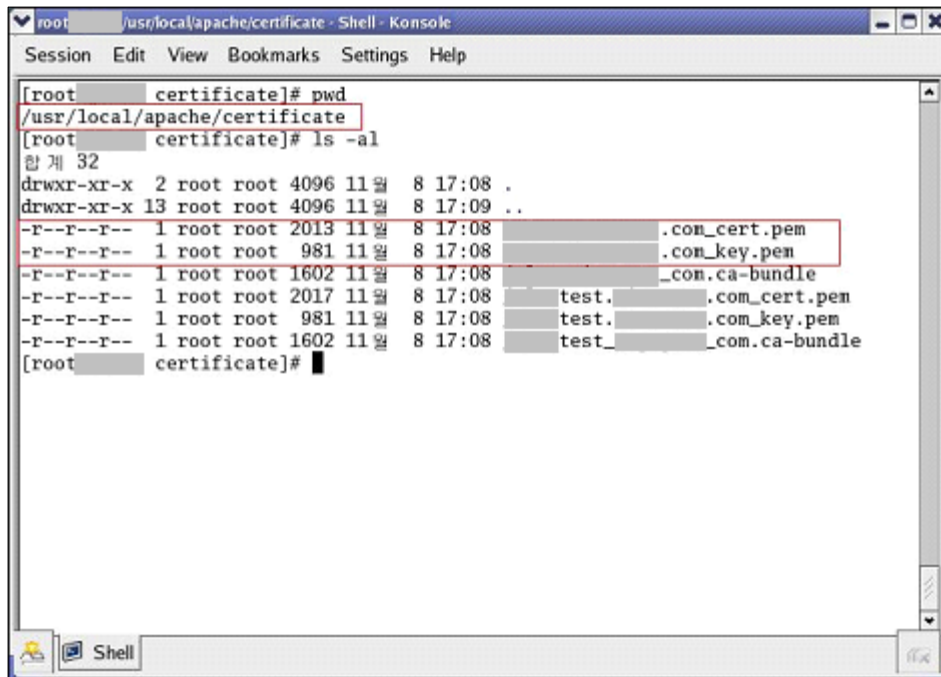
```

-----BEGIN CERTIFICATE-----
MIIBETCBvAIBADBXMqswCQYDVQQGEwJBVTETMBEGA1UECBMKU29tZS1TdGF0ZTEh
TEhMB8GA1UEChMYSW50ZXJuZXQgV2lkZ2l0cyBQdHkgTHRkMRAwDgYJKoZIhvcNA
QkBFgFgMFwwDQYJKoZIhvcNAQEBBQADSwAwSAJBAL6nPTy3avNgbubx+ESmD4LV
1LQGfcSh8nehEOIxGwmCPIrhTP87PaA0XvGpvRQUjCGStrlQsd8lcYVVkOaytNUCAwE
AAaAAMA0GCSqGSIb3DQEBAUAA0EAXcMsa8eXgbG2ZhVyFkRVrI4vT8haN39/QJc9
BrRh2nOTKgfMcT9h+1Xx0wNRQ9/SIGV1y3+3abNiJmJBWnJ8Bg==
-----END CERTIFICATE-----

```



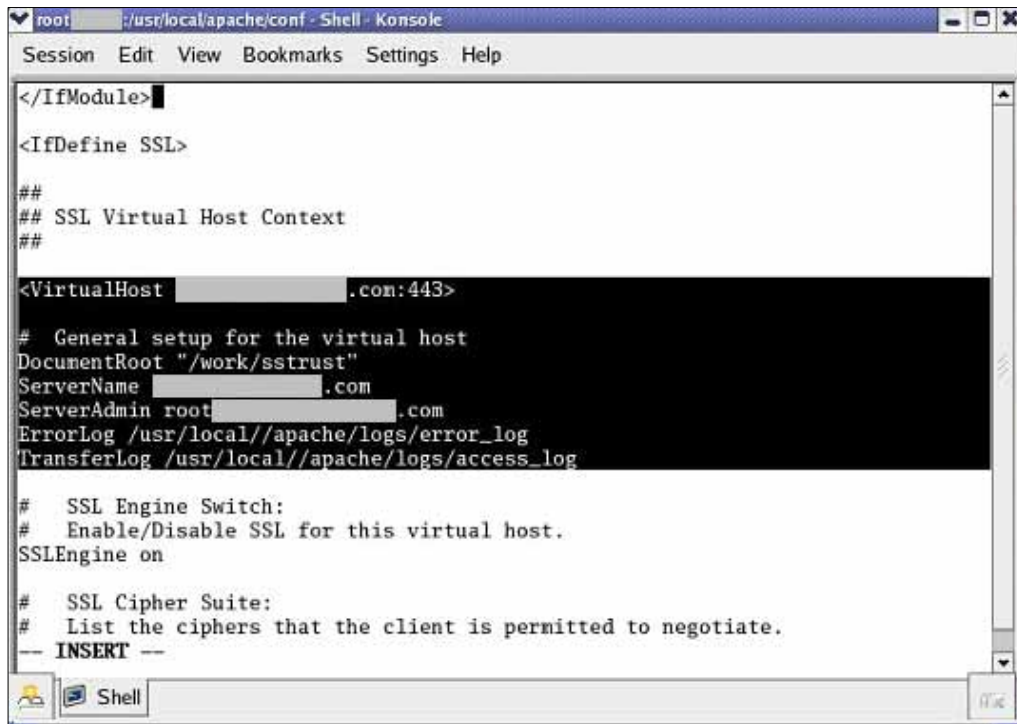
② Apache 서버의 적절한 위치에 저장합니다.



③ ssl.conf 수정 (virtual host 설정)

ssl.conf의 https(SSL)을 사용하기 위해 구성된 virtual host 부분을 http 설정 부분과 동일하게 수정합니다.

```
<VirtualHost (domain):443>
# General setup for the virtual host
DocumentRoot "/usr/local/apache/htdocs"
ServerName (domain):443
ServerAdmin root@(domain)
ErrorLog /usr/local/apache/logs/ssl_error_log
TransferLog /usr/local/apache/logs/ssl_access_log
```



```

root @ /usr/local/apache/conf - Shell - Konsole
Session Edit View Bookmarks Settings Help

</IfModule>

<IfDefine SSL>

##
## SSL Virtual Host Context
##

<VirtualHost [redacted].com:443>

# General setup for the virtual host
DocumentRoot "/work/ssstrust"
ServerName [redacted].com
ServerAdmin root@[redacted].com
ErrorLog /usr/local//apache/logs/error_log
TransferLog /usr/local//apache/logs/access_log

# SSL Engine Switch:
# Enable/Disable SSL for this virtual host.
SSLEngine on

# SSL Cipher Suite:
# List the ciphers that the client is permitted to negotiate.
-- INSERT --

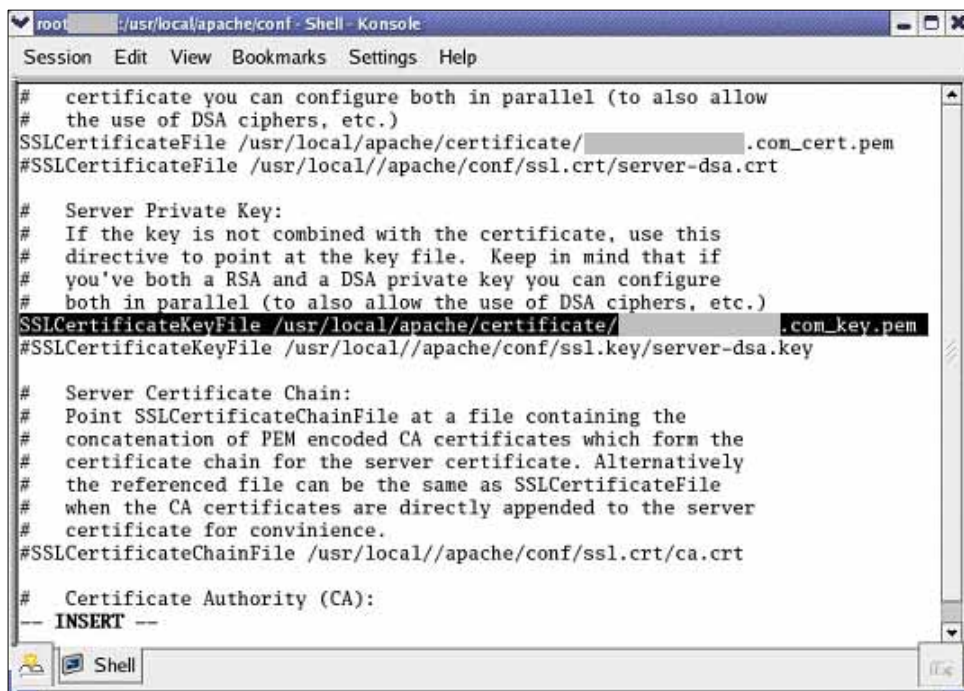
```

#### ④ ssl.conf 수정 (키 파일과 인증서 설정)

ssl.conf 파일에서 인증서 파일과 개인키 파일의 위치와 이름을 알맞게 수정합니다.

인증서 설정 : SSLCertificateFile /usr/local/apache/cert/(domain name)\_cert.pem

개인키 설정 : SSLCertificateKeyFile /usr/local/apache/certificate/(domain name)\_key.pem



```

root @ /usr/local/apache/conf - Shell - Konsole
Session Edit View Bookmarks Settings Help

# certificate you can configure both in parallel (to also allow
# the use of DSA ciphers, etc.)
SSLCertificateFile /usr/local/apache/certificate/[redacted].com_cert.pem
#SSLCertificateFile /usr/local//apache/conf/ssl.crt/server-dsa.crt

# Server Private Key:
# If the key is not combined with the certificate, use this
# directive to point at the key file. Keep in mind that if
# you've both a RSA and a DSA private key you can configure
# both in parallel (to also allow the use of DSA ciphers, etc.)
SSLCertificateKeyFile /usr/local/apache/certificate/[redacted].com_key.pem
#SSLCertificateKeyFile /usr/local//apache/conf/ssl.key/server-dsa.key

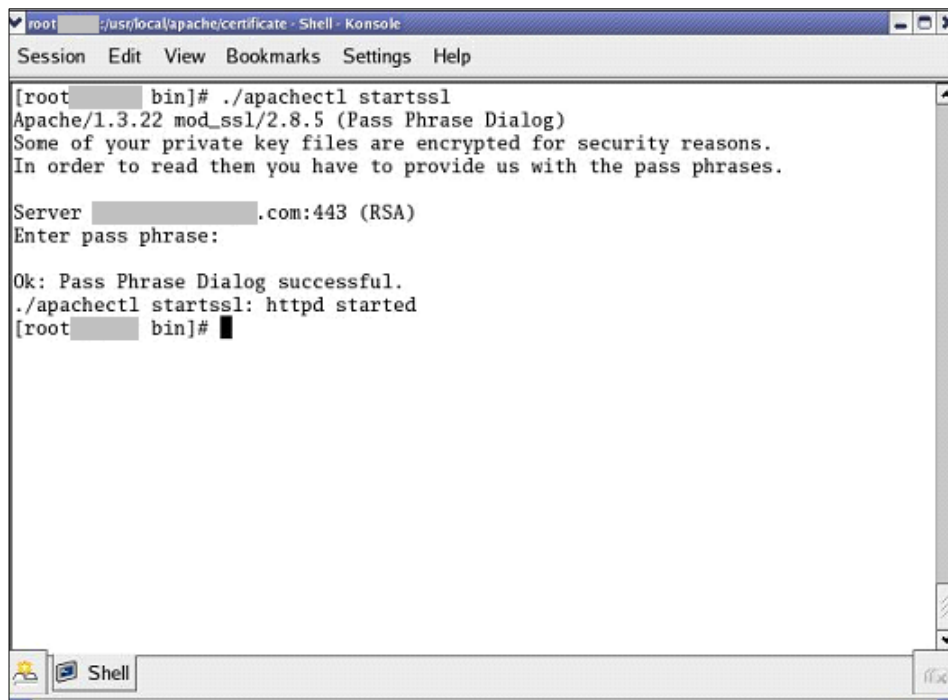
# Server Certificate Chain:
# Point SSLCertificateChainFile at a file containing the
# concatenation of PEM encoded CA certificates which form the
# certificate chain for the server certificate. Alternatively
# the referenced file can be the same as SSLCertificateFile
# when the CA certificates are directly appended to the server
# certificate for convenience.
#SSLCertificateChainFile /usr/local//apache/conf/ssl.crt/ca.crt

# Certificate Authority (CA):
-- INSERT --

```

## ⑤ 웹 서버 재구동

- httpd.conf 파일에 오류가 없는지 확인  
`/usr/local/apache/bin/apachectl -t`  
 Syntax OK 라고 나오면 오류가 없습니다.
- 기존 아파치 서비스 중단  
`/usr/local/apache/bin/apachectl stop`
- http, https 웹 서버를 구동  
`/usr/local/apache/bin/apachectl start` 실행 후 인증서 개인키  
 패스워드 입력하면 http(80), 과 https(443) 두 서비스가 실행



```

root@localhost:~# ./apachectl startssl
Apache/1.3.22 mod_ssl/2.8.5 (Pass Phrase Dialog)
Some of your private key files are encrypted for security reasons.
In order to read them you have to provide us with the pass phrases.

Server localhost.com:443 (RSA)
Enter pass phrase:
Ok: Pass Phrase Dialog successful.
./apachectl startssl: httpd started
root@localhost:~#
  
```

- ⑥ 이제 SSL 인증서의 설치가 완료되었습니다. VI장으로 이동하셔서 실제 웹페이지를 어떻게 수정해야 하는지 알아보겠습니다.

## 2.3 Web2B 서버에서 보안서버 구축하기

## 가. 개인키 생성 및 CSR 생성 방법