Apache HTTP Server (Part 2)

Apache2 Virtual Host

1) IP-based vhosts
   where each vhost has its own unique IP address
2) Name-based vhosts
   where more than one vhost runs on the same IP address(on the same host), but with different names accessible. This is also the most common way to run multiple Web sites on the same Apache server.

Configure Name-based vhosts

1) store the Web sites on the file system i.e /var/www/example.com (if Web sites example.com will be served)
2) allow this Web sites to be available on Apache server by setting up a Name-based virtual host in /etc/apache2/sites-available/
3) enable this configured virtual host in /etc/apache2/sites-enabled/ by creating a soft link to its configuration file in /etc/apache2/sites-available/ (more in Apache HTTP server: Part1)
4) reload the configuration file of Apache server, if needed, restart apache server again

> *Adding name-based virtual hosts to your Apache configuration will not let the the DNS server resolve this name "example.com" to correct web host. Adding records to the DNS Server is also necessary, so that the name "example.com" resolve to the IP address of the Apache Web Server. When users type the URL "http://example.com" into any browser location bars, the browser will contact a DNS Server to look up that name "example.com" and resolve it to the exact IP address. If there is no DNS record, then their browser can't find the web pages on the apache server.*

Step by Step to set up of a self-signed SSL certificate with Apache2

> Idea of setting up a self-signed SSL
> *- install apache and openSSL (1)*
> *- using openssl to generate necessary files of Certifications (2)(3)(4)(5)*
> *- integrate them into apache web server (6)*
> *- configure Apache to use the self-signed Certificate (7)(8)*

1) install apache2 and openssl

```
$ sudo aptitude install apache2 libapache-mod-ssl
$sudo aptitude install openssl
```

2) create a private key for SSL private/public key encryption system with name "server.key"

```
$ sudo su
$ cd /root
$ mkdir certificate && cd certificate

root@think-pad:~/certificate# openssl genrsa -des3 -out server.key 1024
Generating RSA private key, 4096 bit long modulus
.....++
.....++
e is 65537 (0x10001)
Enter pass phrase for server.key:
Verifying - Enter pass phrase for server.key:
root@think-pad:~/certificate# ls -l
total 4
-rw-r--r-- 1 root root 3311 2010-06-19 17:05 server.key
```

3) using "server.key" to create a certificate signing requestion "server.csr"

```
root@think-pad:~/certificate# openssl req -new -key server.key -out server.csr
Enter pass phrase for server.key:                    #if you have protected the key with passwd
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) [AU]:IQ                 #IQ: for Iraq
State or Province Name (full name) [Some-State]:Babylon
Locality Name (eg, city) []:Hilla
Organization Name (eg, company) [Internet Widgits Pty Ltd]:Net. Dept.
```

```
Organizational Unit Name (eg, section) []:Linux Lab
Common Name (eg, YOUR name) []:uobabylon.org
Email Address []:webmaster@uobabylon.org

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:
An optional company name []:

root@think-pad:~/certificate# cat server.csr
-----BEGIN CERTIFICATE REQUEST-----
MIIE2zCCAsMCAQAwgZUxCzAJBgNVBAYTAkRFMQ8wDQYDVQQIEwZCZXJsaW4xDzAN … …
gTiFrx63rTn9XJn/NIoHhFABfFr80o99cd5SZphJpj02w+JnfOi6qIbUG2AudC8=
-----END CERTIFICATE REQUEST-----
```

4.a) using private key to sign the generated Certificate Request server.csr and this CRT lasts 365 days

```
root@think-pad:~/certificate# openssl x509 -req -days 365 -in server.csr -signkey server.key -out server.crt
Signature ok
subject=/C=DE/ST=Berlin/L=Berlin/O=ziik TU
Berlin/OU=ziik/CN=www.uobabylon.org/emailAddress=george.ejaam@gmail.com
Getting Private key
Enter pass phrase for server.key:
root@think-pad:~/certificate# ls -l
total 12
-rw-r--r-- 1 root root 2021 2010-06-19 18:01 server.crt    # security certificate
-rw-r--r-- 1 root root 1760 2010-06-19 17:37 server.csr    # certificate signing request
-rw-r--r-- 1 root root 3311 2010-06-19 17:05 server.key    # key
```

5) remove passwd from the server.key and left a copy of server.key with passwd protection

```
root@think-pad:~/certificate# openssl rsa -in server.key -out server.key.insecure
Enter pass phrase for server.key:
writing RSA key
root@think-pad:~/certificate# mv server.key server.key.secure
root@think-pad:~/certificate# mv server.key.insecure server.key
root@think-pad:~/certificate# ls -l
total 16
-rw-r--r-- 1 root root 2021 2010-06-19 18:01 server.crt          # self-signed server certificate
-rw-r--r-- 1 root root 1760 2010-06-19 17:37 server.csr          # certificate signing request
-rw-r--r-- 1 root root 3243 2010-06-19 18:10 server.key          # without passwd, start apache without entering passwd
-rw-r--r-- 1 root root 3311 2010-06-19 17:05 server.key.secure   # with passwd, start apache with entering passwd
```

6) copy certification related files in apache configuration directory (suggests to create /etc/apache2/ssl)

```
root@think-pad:~# mkdir /etc/apache2/ssl
root@think-pad:~# cp /root/certificate/server.key /etc/apache2/ssl
root@think-pad:~# cp /root/certificate/server.crt /etc/apache2/ssl
root@think-pad:~# ls -l /etc/apache2/ssl /
total 8
-rw-r--r-- 1 root root 2021 2010-06-19 19:16 server.crt
-rw-r--r-- 1 root root 3243 2010-06-19 19:16 server.key
```

7) turn on SSL engine and enable /load SSL module on apache web server

```
root@think-pad:~# ls /etc/apache2/mods-enabled/
alias.conf      authz_default.load   autoindex.conf  deflate.load  mime.conf      php5.conf     status.conf
alias.load      authz_groupfile.load autoindex.load  dir.conf      mime.load      php5.load     status.load
root@think-pad:~# a2enmod ssl
Enabling module ssl.
See /usr/share/doc/apache2.2-common/README.Debian.gz on how to configure SSL and create self-signed certificates.
Run '/etc/init.d/apache2 restart' to activate new configuration!
root@think-pad:~# ls /etc/apache2/mods-enabled/
alias.conf      authz_default.load   autoindex.conf  deflate.load  mime.conf      php5.conf     ssl.conf
alias.load      authz_groupfile.load autoindex.load  dir.conf      mime.load      php5.load     ssl.load
root@think-pad:~# vi /etc/apache2/ports.conf
put this line into ports.conf
NameVirtualHost *:443
```

8) references .crt and .key in the SSL configuration for your web on apache server

root@think-pad:~# ***cp /etc/apache2/sites-available/default-ssl /etc/apache2/sites-available/uobabylon.org.ssl***
root@think-pad:~# ***a2ensite uobabylon.org.ssl***
Enabling site uobabylon.org.ssl.
Run '/etc/init.d/apache2 reload' to activate new configuration!
root@think-pad:~# ***ls /etc/apache2/sites-enabled/***
000-default uobabylon.org ***uobabylon.org.ssl*** *unikufa.org*
root@think-pad:~# ***vi /etc/apache2/sites-available/uobabylon.org.ssl***

update these directives to references the certificate-related file on apache server

```
ServerName www.uobabylon.org
DocumentRoot /var/www/uobabylon.org

<Directory /var/www/uobabylon.org/>
            Options Indexes FollowSymLinks MultiViews
             AllowOverride All
             Order allow,deny
             allow from all
 </Directory>

SSLEngine On

SSLCertificateFile /etc/apache2/ssl/server.crt
SSLCertificateKeyFile /etc/apache2/ssl/server.key
```

9) restart apache server and test your web with HTTPS

be sure you have added the domain name into /etc/hosts, which you used in certificate signing request (server.csr)
$root@think-pad:~# ***/etc/init.d/apache2 restart***
** Restarting web server apache2 ... waiting*
[OK]

4.b) [optional solution] sign the CSR with self-created CA

root@think-pad:~/certificate# ***openssl genrsa -des3 -out CA.key 1024***
Generating RSA private key, 1024 bit long modulus
.....++++++
.....++++++
e is 65537 (0x10001)
Enter pass phrase for CA.key:
Verifying - Enter pass phrase for CA.key:

root@think-pad:~/certificate# ***openssl req -new -x509 -days 365 -key CA.key -out CA.crt***
Enter pass phrase for CA.key:
You are about to be asked to enter information that will be incorporated
into your certificate request. ……..
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
*Country Name (2 letter code) [AU]:***IQ**
*State or Province Name (full name) [Some-State]:***Kufa province**
*Locality Name (eg, city) []:***Kufa city**
*Organization Name (eg, company) [Internet Widgits Pty Ltd]:***University of Kufa**
*Organizational Unit Name (eg, section) []:***ITCK**
Common Name (eg, YOUR name) []: ***Information Technology Center of Kufa University (ITCK)***
*Email Address []:***webmaster@itck.uni-kufa.iq**

*root@think-pad:~/certificate#****openssl x509 -req -days 365 -in server.csr -CA CA.crt -CAkey CA.key -set_serial 01 -out server.crt***
Signature ok
subject=/C=DE/ST=Berlin/L=Berlin/O=ziik TU
Berlin/OU=ziik/CN=www.uobabylon.org/emailAddress=george.ejaam@gmail.com
Getting CA Private Key
Enter pass phrase for CA.key:

!!! *send CA.crt to user and import CRT in used browser.(for local testingm* ***root@think-pad:~/certificate# cp CA.crt /home/toledot/tmp/*** *)*

Used References:

Apache 2.2

http://httpd.apache.org/docs/2.2/

openssl

http://www.openssl.org/docs/HOWTO/certificates.txt

http://www.openssl.org/docs/HOWTO/keys.txt

http://www.openssl.org/docs/HOWTO/proxy_certificates.txt

http://www.modssl.org/docs/

[AC96]
Bruce Schneier, *Applied Cryptography*, 2nd Edition, Wiley, 1996. Seehttp://www.counterpane.com/ for various other materials by Bruce Schneier.

[X208]
ITU-T Recommendation X.208, *Specification of Abstract Syntax Notation One (ASN.1)*, 1988. See for instance http://www.itu.int/rec/recommendation.asp?type=items&lang=e&parent=T-REC-X.208-198811-I.

[X509]
ITU-T Recommendation X.509, *The Directory - Authentication Framework*. See for instancehttp://www.itu.int/rec/recommendation.asp?type=folders&lang=e&parent=T-REC-X.509.

[PKCS]
Public Key Cryptography Standards (PKCS), RSA Laboratories Technical Notes, Seehttp://www.rsasecurity.com/rsalabs/pkcs/.

[MIME]
N. Freed, N. Borenstein, *Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies*, RFC2045. See for instance http://ietf.org/rfc/rfc2045.txt.

[SSL2]
Kipp E.B. Hickman, *The SSL Protocol*, 1995. Seehttp://www.netscape.com/eng/security/SSL_2.html.

[SSL3]
Alan O. Freier, Philip Karlton, Paul C. Kocher, *The SSL Protocol Version 3.0*, 1996. Seehttp://www.netscape.com/eng/ssl3/draft302.txt.

[TLS1]
Tim Dierks, Christopher Allen, *The TLS Protocol Version 1.0*, 1999. Seehttp://ietf.org/rfc/rfc2246.txt.