**How to create a self-signed SSL Certificate**

*The self-signed certificate is guaranteed to work for encryption but not identification,*

*while the digital identification certificate from the CA is guaranteed to work for encryption and identification.*

**Step 1: Generate a Private Key**

The first step is to create your RSA Private Key.

This key is a 1024-bit RSA key which is encrypted using Triple-DES and stored in a **PEM format** so that it is readable as ASCII text.

# openssl genrsa -des3 -out server.key 1024

Generating RSA private key, 1024 bit long modulus

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

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

e is 65537 (0x10001)

Enter PEM pass phrase:

Verifying password - Enter PEM pass phrase:

**Step 2: Generate a CSR (Certificate Signing Request)**

Once the private key is generated a Certificate Signing Request can be generated. The CSR is then used in one of two ways:

1. Ideally, the CSR will be sent to a Certificate Authority.
2. The second option is to self-sign the CSR, which will be demonstrated in the next section.

During the **generation of the CSR**, you will be prompted for several pieces of information. These are the **X.509 attributes** of the certificate. One of the prompts will be for "Common Name (e.g., YOUR name)". It is important that this field be filled in with the FQDN of the server to be protected by SSL.

openssl req -new -key server.key -out server.csr

Country Name (2 letter code) [AU]:AU

State or Province Name (full name) [Queensland]: Queensland

Locality Name (eg, city) [Brisbane]: Brisbane

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

Organizational Unit Name (eg, section) []: Information Technology

Common Name (eg, your name or your server's hostname) []: **server1.local.vm**

Email Address []: some.email@local.vm

Please enter the following 'extra' attributes

to be sent with your certificate request

A challenge password []:

An optional company name []:

**Step 3: Remove Passphrase from Key**

One unfortunate side-effect of the pass-phrased private key is that Apache will ask for the passphrase each time the web server is started. Obviously, this is not necessarily convenient as someone will not always be around to type in the passphrase, such as after a reboot or crash.

mod\_ssl includes the ability to use an external program in place of the built-in pass-phrase dialog, however, this is not necessarily the most secure option either. It is possible to remove the Triple-DES encryption from the key, thereby no longer needing to type in a passphrase. If the private key is no longer encrypted, it is critical that this file only be readable by the root user! If your system is ever compromised and a third party obtains your unencrypted private key, the corresponding certificate will need to be revoked.

Use the following command to remove the passphrase from the key:

cp server.key server.key.org

openssl rsa -in server.key.org -out server.key

The newly created server.key file has no more passphrase in it.

-rw-r--r-- 1 root root 745 Jun 29 12:19 server.csr

-rw-r--r-- 1 root root 891 Jun 29 13:22 server.key

-rw-r--r-- 1 root root 963 Jun 29 13:22 server.key.org

**Step 4: Generating a Self-Signed Certificate**

At this point you will need to generate a self-signed certificate because you either don't plan on having your certificate signed by a CA, or you wish to test your new SSL implementation while the CA is signing your certificate. This temporary certificate will generate an error in the client browser to the effect that the signing certificate authority is unknown and not trusted.

To generate a temporary certificate which is good for 365 days, issue the following command:

openssl x509 -req -days 365 -in server.csr -signkey server.key -out server.crt

Signature ok

subject=/C=CH/ST=Bern/L=Oberdiessbach/O=Akadia AG/OU=Information

Technology/CN=public.akadia.com/Email=martin dot zahn at akadia dot ch

Getting Private key

**Step 5: Installing the Private Key and Certificate**

When Apache with mod\_ssl is installed, it creates several directories in the Apache config directory. The location of this directory will differ depending on how Apache was compiled.

cp server.crt /usr/local/apache/conf/ssl.crt

cp server.key /usr/local/apache/conf/ssl.key

**Step 6: Configuring SSL Enabled Virtual Hosts**

SSLEngine on

SSLCertificateFile /usr/local/apache/conf/ssl.crt/server.crt

SSLCertificateKeyFile /usr/local/apache/conf/ssl.key/server.key

SetEnvIf User-Agent ".\*MSIE.\*" nokeepalive ssl-unclean-shutdown

CustomLog logs/ssl\_request\_log \

"%t %h %{SSL\_PROTOCOL}x %{SSL\_CIPHER}x \"%r\" %b"

**Step 7: Restart Apache and Test**

/etc/init.d/httpd stop

/etc/init.d/httpd stop