How To Create a SSL Certificate on Apache on Arch Linux

Authored by: ***ASPHostServer Administrator*** *[asphostserver@gmail.com]*
Saved From: *http://faq.asphosthelpdesk.com/article.php?id=217*

Step One—Switch Into the Apache Config Directory

The first step is move into the main apache configuration directory. All of the subsequent steps will take place within the directory:

```
cd /etc/httpd/conf
```

Step Two—Create a Self Signed SSL Certificate

Start off by creating the 1024 rsa private key. The "-des3â€• option designates the need for a passphrase. Although having the passphrase in place does provide heightened security, the issue starts when one tries to reload apache. In the event that apache crashes or needs to reboot, you will always have to re-enter your passphrase to get your entire web server back online.

```
sudo openssl genrsa -des3 -out server.key 1024\
```

Now it"s time to create a certificate-signing request. If you set up a passphrase in the previous step, you will be prompted to enter it in this step as well:

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

This command will prompt terminal to display a lists of fields that need to be filled in.

The most important line is "Common Name". Enter your official domain name here or, if you don't have one yet, your site's IP address.

```
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]:US
State or Province Name (full name) [Some-State]:New York
Locality Name (eg, city) []:NYC
Organization Name (eg, company) [Internet Widgits Pty Ltd]:Awesome Inc
Organizational Unit Name (eg, section) []:Dept of Merriment
Common Name (e.g. server FQDN or YOUR name) []:example.com
Email Address []:webmaster@yourdomain.com
```

Finally, remove the passphrase:

```
sudo cp server.key server.key.org
sudo openssl rsa -in server.key.org -out server.key
```

Finish by specifying how long the certificate should remain valid by changing the 365 to the number of days you prefer. As it stands this certificate will expire after one year.

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

Step Three—Finish Up

Your certificate has been created and signed. You only have to be sure that apache includes it in its configuration. Go ahead and open up the main apache config file:

```
sudo nano /etc/httpd/conf/httpd.conf
```

Once there, uncomment the following line:

```
Include conf/extra/httpd-ssl.conf
```

Restart apache to put your changes into effect:

```
sudo systemctl restart httpd
```

In your browser, type https://youraddress, and you will be able to see the new self-signed certificate.