How To Create a SSL Certificate on Apache for Ubuntu 12.04

Authored by: ***ASPHostServer Administrator*** *[asphostserver@gmail.com]*
Saved From: *http://faq.asphosthelpdesk.com/article.php?id=162*

About SSL Certificates

A SSL certificate is a way to encrypt a site's information and create a more secure connection. Additionally, the certificate can show the virtual private server's identification information to site visitors. Certificate Authorities can issue SSL certificates that verify the server's details while a self-signed certificate has no 3rd party corroboration.

Set Up

The steps in this tutorial require the user to have root privileges on the server. You can see how to set that up here in steps 3 and 4.

Additionally, you need to have apache already installed and running on your virtual server.
If this is not the case, you can download it with this command:

sudo apt-get install apache2

Step One—Activate the SSL Module

The next step is to enable SSL on the control panel.

sudo a2enmod ssl

Follow up by restarting Apache.

sudo service apache2 restart

Step Two—Create a New Directory

We need to create a new directory where we will store the server key and certificate

sudo mkdir /etc/apache2/ssl

Step Three—Create a Self Signed SSL Certificate

When we request a new certificate, we can specify how long the certificate should remain valid by changing the 365 to the number of days we prefer. As it stands this certificate will expire after one year.

sudo openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout /etc/apache2/ssl/apache.key -out /etc/apache2/ssl/apache.crt

With this command, we will be both creating the self-signed SSL certificate and the server key that protects it, and placing both of them into the new directory.

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@awesomeinc.com

Step Four—Set Up the Certificate

Now we have all of the required components of the finished certificate.The next thing to do is to set up the virtual hosts to display the new certificate.

Open up the SSL config file:

nano /etc/apache2/sites-available/default-ssl

Within the section that begins with <VirtualHost _default_:443>, quickly make the following changes.

Add a line with your server name right below the Server Admin email:

ServerName example.com:443

Replace example.com with your DNS approved domain name or server IP address (it should be the same as the common name on the certificate).

Find the following three lines, and make sure that they match the extensions below:

SSLEngine on

SSLCertificateFile /etc/apache2/ssl/apache.crt

SSLCertificateKeyFile /etc/apache2/ssl/apache.key

Save and Exit out of the file.

Step Five—Activate the New Virtual Host

Before the website that will come on the 443 port can be activated, we need to enable that Virtual Host:

sudo a2ensite default-ssl

You are all set. Restarting your Apache server will reload it with all of your changes in place.

sudo service apache2 reload

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