How To Create a Self-Signed SSL Certificate for Nginx on debian

217 Manu Chacko August 21, 2024 Tweaks & Configuration 8403

How to create an SSL certificate on SSH jump server for Nginx on Debian

---

Overview: This article describes how to create a self-signed SSL certificate, configure Nginx to use SSL, and adjust the Nginx configuration to enable SSL for secure web server communication.

Step 1. Create a Self-signed 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/ssl/private/nginx-selfsigned.key -out /etc/ssl/certs/nginx-selfsigned.crt
```

With this command, we will be both creating the self-signed SSL certificate and the server key. This command will prompt the terminal to display a list of fields that need to be filled in. Fill out the prompts appropriately. The most important line is the one that requests the Common Name (ex. server FQDN or YOUR name). You need to enter the domain name associated with your server or, more likely, your server's public IP address.

```
Output:

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

State or Province Name (full name) [Some-State]:New York

Locality Name (eg, city) []:New York City

Organization Name (eg, company) [Internet Widgits Pty Ltd]:Bouncy Castles, Inc.

Organizational Unit Name (eg, section) []:Ministry of Water Slides

Common Name (e.g. server FQDN or YOUR name) []:server_IP_address
```

```
Email Address []:admin@your_domain.com
```

Both of the files you created will be placed in the appropriate subdirectories of the /etc/nginx directory.

Next, create a new Nginx SSL configuration snippet in the /etc/nginx/snippets directory.

```
:~# nano /etc/nginx/snippets/self-signed.conf
```

Within this file, we just need to set the ssl_certificate directive to our certificate file and the ssl_certificate_key to the associated key. In our case, this will look like this:

```
ssl_certificate /etc/ssl/certs/nginx-selfsigned.crt;

ssl_certificate_key /etc/ssl/private/nginx-selfsigned.key;
```

When you've added those lines, save and close the file.

Next, we will create another configuration file to tighten up the SSL.

Before we go any further, let's back up our current server block file:

```
:~# sudo cp /etc/nginx/sites-available/default /etc/nginx/sites-available/default.bak
```

Open the server block file to make adjustments.

```
:~# sudo nano /etc/nginx/sites-available/default
```

```
server {

listen 80 default_server;

listen [::]:80 default_server;

server_name server_domain_or_IP;

return 302 https://$server_name$request_uri;

}

server {

# SSL configuration

listen 443 ssl default_server;
```

```
listen [::]:443 ssl default_server;

include snippets/self-signed.conf;

. . .
```

Your virtual host is now all setup! Save and Exit Restart Apache.

```
:~# sudo systemctl restart nginx
```

Related Articles

Configure the Jump server to use SSL for Mysql.

Install SSL cert in Jump Server.

Install Master/Slave with Mysql ssl.

Online URL:
https://www.ezeelogin.com/kb/article/how-to-create-a-self-signed-ssl-certificate-for-nginx-on-debian-217.html