Installing a certificate on Nginx

This guide will assist you in the installation of your SSL certificate on Nginx. We have used a Comodo PositiveSSL as an example below. However, the steps remain the same for all SSLs.

1. Upload certificates on the server where your website is hosted.

In case of Comodo certificates, you should receive the zip archive with *.crt and *.ca-bundle files.

You can download a completed Bundle file for each certificate we provide here.

Another way is downloading the certificate from your Namecheap account according to this guide. The zip folder will contain the .crt file for your certificate and .ca-bundle file for the CA Bundle.

For most Comodo Domain Validated certificates (such as PositiveSSL, for example) the files will appear like the ones below:

youdomainname.crt
youdomainname.ca-bundle

or you may receive the CA bundle in separate files as provided below:

ComodoRSADomainValidationSecureServerCA.crt
COMODORSAAddTrustCA.crt
AddTrustExternalCARoot.crt

2. Combine all the certificates into a single file.

For Nginx it is required to have all the certificates (one for your domain name and CA ones) combined in a single file. The certificate for your

domain should be listed first in the file, followed by the chain of CA certificates.

To combine the certificates in case of PositiveSSL, run the following command in terminal:

$ cat *yourdomainname*.crt ComodoRSADomainValidationSecureServerCA.crt COMODORSAAddTrustCA.crt AddTrustExternalCARoot.crt >> cert_chain.crt

Note! If you have downloaded a complete CABundle file for your certificate, replace chain files' names with the name of your downloaded file. It will look like:

$ cat *yourdomainname*.crt COMODO_DV_SHA-256_bundle.crt >> cert_chain.crt
or
$ cat *yourdomainname*.crt *yourdomainname*.ca-bundle >> cert_chain.crt

3. Edit your Nginx VirtualHost file.

By default, the configuration file is named nginx.conf and placed in the directory /usr/local/nginx/conf, /etc/nginx, or /usr/local/etc/nginx.

If you do not have a record for port 443 in your VirtualHost, you should add it manually.

To simplify the process, you can duplicate the record for port 80 (should be in your VirtualHost file by default) and change port 80 to port 443. Simply add it below the non-secure module. In addition to port changes you will need to add the special lines in the record:

ssl on;

 # ssl_certificate should be pointed to the file with combined certificates (file you created in step 2)
ssl_certificate /etc/ssl/cert_chain.crt;
 # ssl_certificate_key should be pointed to the Private Key that has been generated with the CSR code that you have used for activation of the certificate.
ssl_certificate_key /etc/ssl/*your_private_key*.key;

Completed VirtualHost record for port 443 may look like the one below:

server {
listen 443;
ssl on;
ssl_certificate /etc/ssl/cert_chain.crt;
ssl_certificate_key /etc/ssl/yourdomainnamekey.key;

server_name **yourdomainname_com;**
access_log /var/log/nginx/nginx.vhost.access.log;
error_log /var/log/nginx/nginx.vhost.error.log;
location / {

root /var/www/;
index index.html;
}

}

Note! If you are using a multi-domain or wildcard certificate, it is necessary to modify the configuration files for each domain/subdomain included in the certificate. You would need to specify the domain/subdomain you need to secure and refer to the same certificate files in the VirtualHost record the way described above.

Once you have modified the VirtualHost file, it is required to restart Nginx

in order to apply the changes. You can restart Nginx with this command:
nginx -s reload