|  |  |
| --- | --- |
|  | **Sécuriser un serveur Apache2 (HTTPS)** |

# Création du certificat

**Prérequis**Pour cette configuration, on doit disposer d'un hôte Linux avec un serveur web Apache et un accès root.   
On peut utiliser une machine virtuelle en locale ou un hôte hébergé dans le nuage.

**Installation d’Apache**

Si tel n’est pas le cas, pour installer Apache2 …  
**>> sudo apt install apache2**

**Vérification de la bonne installation**

Pour vérifier si Apache2 est installé et démarré …  
**>> sudo apachectl -v  
Server version: Apache/2.4.51 (Debian)à  
Server built: 2021-10-07T17:49:44**et **>> sudo systemctl status apache2.service  
● apache2.service - The Apache HTTP Server  
 Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor preset: enabled)  
 Active: active (running) since Fri 2021-11-05 17:37:16 EDT; 14h ago  
 Docs: https://httpd.apache.org/docs/2.4/  
…**

# Création de la paire de clés et du certificat

On utilise le paquet openssl pour la génération du certificat.

**Remarque** …  
Le paquet peut ne pas être déjà installé sur le serveur.  
On peut le vérifier …  
**>> openssl version  
OpenSSL 1.1.1k 25 Mar 2021**

Sinon, il faut l’installer, pour ce faire …  
**>> sudo apt install openssl**

On crée et on se déplace dans le répertoire dans lequel on retrouvera le certificat et clés …  
**>> sudo mkdir --verbose /etc/apache2/ssl/ && cd /etc/apache2/ssl**

La commande pour la création du certificat est la suivant …  
**>> sudo openssl req -x509 -nodes \  
 -days 365 -newkey rsa:2048 \  
 -keyout /etc/apache2/ssl/serveur.key \  
 -out /etc/apache2/ssl/serveur.crt  
Generating a RSA private key  
......................................+++++  
.....................................+++++  
…  
-----  
Country Name (2 letter code) [AU]: CA  
State or Province Name (full name) [Some-State]: Quebec  
Locality Name (eg, city) []: Montreal  
Organization Name (eg, company) [Internet Widgits Pty Ltd]: squidly.info  
Organizational Unit Name (eg, section) []: Formation  
Common Name (e.g. server FQDN or YOUR name) []: www.squidly.info  
Email Address []: squidly@squidly.info  
-----BEGIN CERTIFICATE-----  
MIID9TCCAt2gAwIBAgIUV2hA6eDYQQ6jlr1tSWa6yoySRfkwDQYJKoZIhvcNAQEL  
BQAwgYkxCzAJBgNVBAYTAmNhMRIwEAYDVQQIDAlRdcODwqliZWMxEzARBgNVBAcM  
Ck1vbnRyw4PCqWwxFTATBgNVBAoMDHNxdWlkbHkuaW5mbzEZMBcGA1UEAwwQd3d3  
…  
DomAxpICeLkc  
-----END CERTIFICATE-----**

Quelques explications sur la commande …

* **openssl req -x509**  
  Demande à OpenSSL de créer un certificat de type x509 autosigné ;
* **nodes**  
  Option pour ne pas chiffrer la clé privée avec un mot de passe, pour qu'Apache puisse utiliser la clé privée sans intervention utilisateur ;
* **days 365**Durée de validité du certificat, d'un an ;
* **newkey rsa:2048**Option pour générer une nouvelle paire de clés pour le certificat, en utilisant le système RSA avec une clé de   
  2 048 bits ;
* **keyout /etc/apache2/ssl/serveur.key**Emplacement où sera stockée la clé privée générée ;
* **out /etc/apache2/ssl/serveur.crt**Emplacement où sera stocké le certificat, qui contient la clé publique générée.

On peut laisser les informations par défaut demandées par OpenSSL lors de la génération du certificat en appuyant sur Entrée.

Pour vérifier le certificat en mode lecture … **>> sudo openssl x509 -noout -text -in /etc/apache2/ssl/serveur.crt**

### Modification du fichier de configuration du site sécurisé

Le certificat autosigné est créé.   
Il faut maintenant configurer Apache pour l'utiliser.   
On entre la commande suivante pour ouvrir le fichier de configuration du Virtual Host TLS par défaut d'Apache …   
**>> sudo nano /etc/apache2/sites-available/default-ssl.conf**

En remplace les lignes suivantes …

* **SSLCertificateFile /etc/ssl/certs/apache-autosigne.crt**  
  par **SSLCertificateFile /etc/apache2/ssl/serveur.crt**
* **SSLCertificateKeyFile /etc/ssl/private/apache-autosigne.key**par   
  **SSLCertificateKeyFile /etc/apache2/ssl/serveur.key**

Il s’agit du certificat et de la clé privée qui viennent d’être créées.  
On enregistre et on ferme le fichier.

Pour vérifier la syntaxe des fichiers de configuration …  
**>> sudo apachectl configtest**

Il faut ensuite activer le module ssl d'Apache avec la commande …  
**>> sudo a2enmod ssl**

On active le Virtual Host TLS d'Apache …  
**>> sudo a2ensite default-ssl**

Comme dernière étape, on redémarre le service Apache …  
**>> sudo systemctl restart apache2.service**

Il est maintenant possible de tester le bon fonctionnement en ouvrant votre navigateur, par exemple Firefox, et en visitant **https://localhost**.

# Chaînes de certificats SSL

Il est maintenant possible de visiter https://localhost grâce au certificat autosigné.

Le **fureteur affiche un message d'erreur**.   
Cela est normal… Le fureteur est configuré pour ne pas faire confiance aux certificats autosignés.   
Il est néanmoins possible d’**ajouter** **une exception pour ce site** en cliquant sur Avancé puis Ajouter une exception et Confirmer l'exception.

Certains fureteurs (*browsers*) peuvent se plaindre d'un certificat signé par une autorité de certification bien connue, tandis que d'autres fureteurs peuvent accepter le certificat sans problème. Cela se produit parce que l'autorité émettrice a signé le certificat de serveur à l'aide d'un certificat intermédiaire qui n'est pas présent dans la base d'autorités de certification de confiance bien connues qui est distribué dans un fureteur particulier.

Utiliser un **certificat autosigné** (ou expiré) **ne diminue pas la force du chiffrement des données** **HTTP** par rapport à un certificat valide signé par une autorité de certification. En revanche, un certificat autosigné ne permet pas d'authentifier le serveur web, ce qui signifie qu'un attaquant exécutant une attaque MITM peut intercepter et remplacer le certificat du serveur par le sien, puis déchiffrer et modifier toutes les informations échangées entre le client et le serveur, réduisant à néant la sécurité de TLS.

Un **certificat autosigné est généralement utilisé en environnement local** de développement, de test ou de préproduction.

Par ailleurs, le certificat racine d'une autorité de certification racine est autosigné, puisque c'est elle-même qui est chargée de signer les certificats. Cependant, ce certificat fait partie des certificats de confiance des machines clientes, et est donc reconnu comme valide.

### Redirection d’un site HTTP vers HTTPS

Afin de rediriger les clients HTTP vers un site sécurisé, il faut modifier le site en clair vers le site le site sécurisé.

Pour ce faire, on doit modifier le fichier de configuration du site non sécurisé …  
**>> sudo nano /etc/apache2/sites-available/000-default.conf**

On ajout à la fin du fichier la ligne suivante …  
**Redirect permanent / <Site HTTPS>  
Redirect permanent / https://192.168.1.100**

**Remarque** …  
Dans la ligne précédente, il y a un espace entre / et <Site HTTP>