Créer un certificat signé par une CA avec openSSH

Présentation

Avec OpenSSH vous pouvez créer une autorité de certification et signer les certificats.

Pour en savoir plus : https://sysadmin.cyklodev.com/creer-une-autorite-de-certification-ca-avec-openssl/

Démarche

La création d'un certificat auto-signé pour le CA

- Création de la **clé privée** de la CA ;
- Création de la **requête de certification** ;
- Création du **certificat auto-signé** de la CA.

Dans ce document est présenté la création d'un certificat signé pour un utilisateur

- Création de la **clé privée** de l'utilisateur ;
- Création de la **requête de certification** ;
- Création du **certificat** de l'utilisateur signé par la CA.

Création de la clé privée de l'autorité de certification (CA)

Dans l'invite de commandes lancez la commande suivante :

```
$ openssl genrsa -aes256 -out ca-key.pem 2048
```

Indiquez une pass-phrase pour votre nouvelle clé, pass-phrase qui sera demandé lors de la création du certificat.

Création de la requête de demande de signature du certificat de la CA

Pour générer la requête de demande de signature, il faut utiliser le fichier contenant la clé privée.

Lors de la création de la requête vous avez à saisir vos informations d'identification du CA. Il est important d'indiquer un **Common Name** unique.

- Saisissez la commande suivante :

```
$ openssl req -new -key ca-key.pem -out ca.csr
```

Génération du certificat auto-signé du CA

Lors de la génération d'un certificat autosigné, vous avez à préciser le format du certificat (x509) et la durée de validation. En général un certificat d'une autorité de certification a une durée plutôt longue (ici 365000 jours)

```
$ openssl x509 -req -days 365000 -in ca.csr -signkey ca-key.pem -out ca.crt
```

Le certificat généré est le fichier **.crt** et la clé privée associée, le fichier **.pem**.

Configuration du CA

Le fichier de configuration du CA permet notamment de définir :

- le fichier serial qui incrémentera le nombre de certificats signés ;
- le fichier index.txt qui va garder la trace de chaque certificat émis.

Cela ne sera pas abordé dans ce document.

Signature d'un certificat

Deux cas de figure sont possibles :

- la CA gère à la fois les clés privées et publiques, génère le certificat, le signe et le transmet à l'utilisateur.
- l'utilisateur génère sa bi-clé privée/publique, crée sa demande de certificat (fichier csr) puis demande au CA de signer son certificat. Le CA vérifie alors la validée de la demande (les informations d'identités), signe le certificat et le transmet à l'utilisateur.

La démarche de création d'un certificat signé est la suivante :

- Création de la bi-clé privée et publique (KEY) ;
- Création de la demande de signature (CSR) auprès du CA ;
- Signature du certificat (CRT)

Création de la clé privée de l'utilisateur ou du serveur

Dans l'invite de commandes lancez la commande suivante :

```
$ openssl genrsa -aes256 -out user-private.pem 2048
```

Indiquez une pass-phrase pour votre nouvelle clé, pass-phrase qui sera demandé lors de la création du certificat.

Création de la requête de demande de signature du certificat

Pour générer la requête de demande de signature, il faut utiliser le fichier contenant la clé privée.

Lors de la création de la requête vous avez à saisir vos informations d'identification du CA. Il est important d'indiquer un **Common Name** unique.

- Saisissez la commande suivante :

```
$ openssl req -new -key user-private.pem -out user.csr
```

Génération du certificat avec signature par le CA

Lors de la génération du certificat signé par le CA, vous avez à préciser le format du certificat (x509) et la durée de validation (365 jours). Il est nécessaire de préciser el numéro de série du certificat (set*serial). <code shell> $ openssl x509 -req -days 365 -CA ca.crt -CAkey ca-key.pem -set*serial 01 -in user.csr -out user.crt </code> Le certificat généré est le fichier **.crt** et la clé privée associée, le fichier **.pem**.

Pour vérifier que le certificat de l'utilisateur est correct, utilisez la commande suivante :

```
$ openssl verify -CAfile ca.crt user.crt

user.crt: OK
```

Visualisation des informations du certificat

- sous Linux, vous pouvez constater que le certificat est signé par la CA qui a vérifié l'identité de l'utilisateur:

Extraire la clé publique du certificat

```
$ openssl x509 -in user.crt -pubkey -out user-public.pem
```

Pour visualiser le contenu de cette clé privée :

```
$ openssl rsa -in user-public.pem -pubin -text
```

Utilisation de la clé publique pour une connexion distance en SSH

Génération de des informations qui seront ensuite copiée dans le fichier **~/.ssh/authorized*keys du serveur distant: <code shell> # ssh-keygen -i -m PKCS8 -f user-public.pem » nomfichier </code> Le contenu du fichier nom_fichier doit être ajouté au contenu du fichier ~/.ssh/authorizedkeys. Vous pouvez ensuite vous connecter au serveur distant en indiquant votre clé privée : <code shell> $ ssh -i user-private.pem user@adresseIPserverur </code> ===== Transmettre le certificat ===== Le certificat user.crt est maintenant à transmettre à l'utilisateur : * si l'utilisateur possède déjà sa clé privée, la transmission du certificat peut se faire sans utiliser un canal sécurisé car la clé privée n'est pas transmise. * si vous avez à transmettre la clé privée en plus du certificat c'est à dire une identité numérique complète, il faudra sécuriser la transmission pour protéger la clé privée.***

Cette CA n'est pas reconnu. Il faudra également transmettre le certificat de la CA (ca.crt) pour que l'autorité soit reconnue en l'important dans la magasin de certificat de son ordinateur en tant que certificat approuvée par l'organisation.

===== Placer dans le magasin des certificats confiance les certificats ===== Pour reconnaître un certificat comme de confiance, il faut l'importer dans le magasin des certificats approuvés. Cela concerne le certificat de la CA. ==== Linux-Debian ==== Pour les distributions Debian/Ubuntu, tous les certificats racine sont installés dans le répertoire **/etc/ssl/certs** mais pour mettre à jour ce répertoire, il faut suivre la démarche suivante : * placez le certificat de la CA dans le dossier **/usr/local/share/ca-certificates/** * puis ensuite il faut effectuer une mise à jour : <code shell> $ sudo cp ca.crt /usr/local/share/ca-certificates/ca.crt $ sudo update-ca-certificates </code> ==== Windows ==== Utilisez la console **certmgr.msc** pour importer le certificat dans le dossier **Autorités de certification racines de confiance**.

CAcert.org est une association qui fourni gratuitement des certificats ssl (signature, chiffrement, connexion, certificats https).

Pour que les certificats signés par **CAcert.org** soit reconnus comme de confiance, il est nécessaire d'installer les certificats racine de CAcert.org.

https://www.cacert.org/