Autorité de certification privée avec Easy RSA

Contributeurs : Démarche reprise du travail de A. Leroy et T. Ledoux (SISR2022)

Source :
https://www.digitalocean.com/community/tutorials/how-to-set-up-and-configure-a-certificate-authority-ca-on-debian-10-fr

RSA (Rivest–Shamir–Adleman du nom des créateurs) est un système cryptographique de gestion de clés asymétriques.

L'**outil easy-rsa** est son implémentation, notamment sous Linux.

Cet outil permet de mettre en place une **infrastructure à clé publique** ou autorité de certification.

Installation et configuration

Procédure

1. installer le paquetage easy-rsa
2. [préparation du dossier de gestion de l'infrastructure]
3. paramétrage du fichier de configuration
4. création de l'autorité de certification

Installation du paquetage

```
 apt update
apt install easy-rsa
```

Préparation du dossier (optionnel)

L'installation ajoute des fichiers dans ***/usr/share/easy-rsa***. Pour simplifier l'accès, on peut créer un dossier virtuel lié par un lien symbolique au dossier /usr/share/easy-rsa/.

```
mkdir ~/easy-rsa
ln -s /usr/share/easy-rsa/* ~/easy-rsa/
chmod 700 ~/easy-rsa
cd ~/easy-rsa
./easyrsa init-pki
```

Paramétrage du fichier

Par défaut, l'outil exploite des valeurs prédéfinies. On doit donc les personnaliser en fonction du contexte en adaptant le fichier exemple disponible lors de l'installation (qu'on va copier).

```
cd ~/easy-rsa
cp vars.example vars
nano vars
```

On ira modifier les paramétrages, notamment ceux d'identification de l'entreprise pour l'autorité (les numéros de ligne sont donnés à titre indicatif). On peut laisser les valeurs par défaut sur les algorithmes pour plus de compatibilité.

```
95 - set_var EASYRSA_REQ_COUNTRY "VOTREPAYS"
96 - set_var EASYRSA_REQ_PROVINCE "VOTREREGION"
97 - set_var EASYRSA_REQ_CITY  "VOTREVILLE"
98 - set_var EASYRSA_REQ_ORG  "VOTREENTREPRISE"
99 - set_var EASYRSA_REQ_EMAIL  "EMAILADMIN"
100 - set_var EASYRSA_REQ_OU  "NOMDUSERVICE"
#valeurs à adpater ou laisser par défaut
117 - set_var EASYRSA_ALGO rsa
214 - set_var EASYRSA_DIGEST
```

Création d'une autorité de certification

Lors de la création de l'autorité, le système demande une **passphrase** (phrase secrète) qui permet de sécuriser les actions futures (certification des demandes).

On se verra aussi demander le nom de l'autorité (qui sera reconnue par les navigateurs).

```
./easyrsa build-ca
```

Deux fichiers ont été créés :

- ca.crt c'est le certificat public de l'AC. C'est ce fichier qu'il faut distribuer aux différents clients. Il se trouve dans le dossier ***pki***
- ca.key qui est la clé privée de l'AC, présente dans ***pki/private***

Ajout du certificat de l'autorité sur le client

L'autorité ainsi créée n'est pas reconnu par les navigateurs.

Il est donc nécessaire d'ajouter le certificat de l'autorité au navigateur pour qu'il puisse l'interroger pour faire valider les certificats des serveurs qu'elle a signé.

Procédure

- Récupérer le certificat ca.crt de l'AC

- Aller dans les ***paramètres*** du navigateur (la démarche est décrite pour Firefox),
  - dans la partie ***vie privée et sécurité***
  - ***afficher les certificats***
  - **importer le certificat ===== *Gestion des demandes de certificat authentique* =====**

> Procédure
>
> - Sur la machine à sécuriser (avec SSL)
>   - créer la clé privée
>   - créer la demande CSR
>   - envoyer le fichier CSR vers l'AC
> - sur l'AC
>   - importer le fichier CSR
>   - signer le fichier CSR
>   - retourner le certificat généré vers le serveur à sécuriser
> - sur le serveur à sécuriser
>   - intégrer le certificat authentique dans la configuration du service à sécuriser (Web, FTP ou autre)

==== Création et signature d'une demande de certificat sur le serveur à sécuriser ==== Si ce n'est pas déjà le cas, on crée la clé privée.

```
//on suppose qu'on part de rien. Si l'environnement est déjà installé, on adaptera les commandes ci-dessous.
apt update
apt install openssl
mkdir /etc/ssl/certs
cd /etc/ssl/certs
//création de la clé privée
openssl genrsa -out <nomcle.key>
```

=== création d'une demande CSR === <code lscript>openssl req -new -key <nomcle.key> -out <demandeCertif.csr> </code> **Après cela il faut copier le fichier .csr sur l'Autorité** <code lscript>scp <demandeCertif.csr> <compte>@<machineAutorite>:/<dossierDestination> </code> ==== Signature d'une CSR sur l'AC ==== On agit à présent sur l'autorité pour signer la demande et produire le certificat

> On suppose que le fichier de requête est présent dans le dossier /tmp sur l'AC.

<code lscript>cd ~/easy-rsa *importe la demande sous un nom qui gardera la trace sur l'AC ./easyrsa import-req /tmp/<demandeCertif.csr> <certif_nomserver>* signe la demande importée à destination d'un 'server' (pourrait être 'client' ou 'ca') ./easyrsa sign-req server <certif_nomserver></code> Il faut récupérer le fichier .crt **au chemin suivant :** ~/easy-rsa/pki/issued/certif_nomserver.crt ***et le copier sur le serveur à sécuriser dans le dossier qui contient ses certificats. <code lscript>scp ~/easy-rsa/pki/issued/<certif_nomserver>.crt <comptedistant>@<serveraSecuriser>:/<dossier></code> Voir ensuite SSL/TLS pour le passage de HTTP vers HTTPS d'un site en prenant en compte ce certificat authentique. ===== Révocation de certificat authentique ===== A la suite d'un événement inattendu (compromission, sortie d'un réseau ou de l'entreprise, etc), les certificats garantis par l'AC ne doivent plus être considérés comme valide. Il faut donc révoquer les certificats concernés, puis*** propager ***l'information auprès des machines qui ont enregistré la validité d'un certificat (normalement les clients, mais aujourd'hui les serveurs OCSP utilisent un mécanisme de propagation automatisé). Sur l'AC <code lscript>cd ~/easy-rsa révocation ./easyrsa revoke <certif_server-web> génération de la liste des révocation de certificat (Certificates Revocation List - CRL) ./easyrsa gen-crl </code> La commande gen-crl crée un fichier*** ~/easy-rsa/pki/crl.pem***. Celui-ci doit être envoyé à toutes les machines (postes utilisateurs, serveur qui hébergeait le certificat, y compris l'AC) qui faisaient confiance aux certificats révoqués (en*** *scp* ***par exemple). sur les machines à mettre à jour On devra importer la liste sur les serveurs devant prendre en compte la nouvelle liste des révocations <code lscript>openssl crl -in <chemin>/crl.pem -noout -text</code>***