Les certificats

Mise en place d'une autorité de certification

Imaginez une entreprise qui met en service un serveur web. Ce service peut faire l'objet d'une stratégie de sécurité. On souhaite donc mettre des conditions à la consultation du serveur :

- fournir la garantie aux clients qui se connectent que le serveur qu'ils consultent est bien celui de cette entreprise
- permettre au serveur de vérifier l'identité des clients qui se connectent
- en cas de problème, pouvoir interdire l'accès à certains clients

Pour arriver à cela, il vous est proposé de suivre les étapes suivantes :

Etape 1. Voir le certificat d'un serveur https public, voir les paramètres relatifs aux certificats dans le navigateur
Etape 2. Installer le serveur http de l'entreprise www.entreprise.com
Etape 3. Installer un serveur https sur votre machine, laisser toutes les valeurs par défaut, et le consulter
Etape 4. Maîtrise du contenu affiché par le serveur https
Etape 5. Création d'une autorité de certification, utilisation du certificat auto-signé pour le serveur
Etape 6. Création d'une autorité de certification pour l'entreprise, utilisation d'un certificat dérivé pour le serveur, reconnaissance du certificat racine de l'entreprise par le navigateur du client
Etape 7. Création de certificat pour les clients, authentification mutuelle client-serveur
Etape 8. Utilisation des listes de révocation

Pour aller plus loin :

- Vous pouvez créer une autorité de certification qui possède sa clé privée et son certificat auto-signé. Cette autorité primaire doit être absolument protégée, et le meilleur moyen est de la laisser hors-ligne, dans un coffre fort. Dans ces conditions, il est difficile de créer des certificats dérivés. En réalité, on protège effectivement l'autorité racine, mais avant de la mettre au coffre, on en dérive quelques autorités dérivées. Ce sont ces autorités dérivées qui seront approuvées par les navigateurs, et qui vont signer les certificats.

Quelques conventions de nommage à utiliser pendant tout le TP :

- nom de l'entreprise : entreprise
- nom du domaine Internet de l'entreprise : entreprise.com
- nom du serveur http de l'entreprise : www.entreprise.com
- nom du serveur https de l'entreprise : serv1.entreprise.com

Etape 1

Voir le certificat d'un serveur, voir les paramètres relatifs aux certificats dans le navigateur

Depuis un navigateur, il est possible d'avoir des informations relatives à l'utilisation des certificats :

- concernant le serveur consulté : certificat du serveur
- concernant le navigateur lui-même : liste des autorités reconnues comme digne de confiance par ce navigateur

1. Allez sur un site serveur https pour récupérer son certificat :
   - allez sur Gmail (par exemple), double cliquer sur le petit cadenas
   - interprétez toutes les informations contenues dans le certificat

2. Observez la liste des certificats que votre navigateur accepte comme dignes de confiance :
   - Edition/préférences/avancé/chiffrement…
   - Voyez la liste des autorités et des serveurs autorisés de façon permanente…

Remarque : il est possible que vous ayez à mettre votre navigateur à jour pour avoir accès à toutes ces informations.

Etape 2

Installer un serveur http sur votre machine : www.entreprise.com

1. Pour effectuer les opérations suivantes, vous devez avoir les pouvoir administrateur :
   - **su –**
   - **Password : xxxx**

2. Mettez à jour le serveur appache (ou installez-le si il ne l'est pas…), lancez le serveur
   - **yum update httpd**
   - **yum install httpd**
   - **service httpd start**

3. Lancez le navigateur firefox, adressez http://127.0.0.1.

   A ce stade, le serveur répond, mais uniquement sollicité par l'adresse de boucle locale, et vous ne maîtrisez pas la page affichée. Le fichier de configuration du serveur est /etc/httpd/conf/httpd.conf. Il contient beaucoup de paramètres et en particulier :
   - ligne 275 environ : ServerName le nom sous lequel le serveur va répondre
   - ligne 291 environ : DocumentRoot le répertoire racine du site

4. Editez le fichier /etc/httpd/conf/httpd.conf :
   - dé-commentez le paramètre ServerName, et placez-le à la valeur www.entreprise.com
   - notez la valeur du paramètre DocumentRoot : /var/www/html

5. Ajoutez www.entreprise.com dans le fichier de résolution des noms /etc/hosts :
   - 127.0.0.1 www.entreprise.com

6. Redémarrez le serveur httpd :
   - **service httpd restart**

7. Editez un fichier index.html, dans lequel vous mettrez le nom du répertoire racine du serveur :
   - **echo « /var/www/html/index.html » > /var/www/html/index.html**

8. Relancez le navigateur firefox, adressez http://127.0.0.1 ou http://www.entreprise.com

Etape 3

Installer un serveur https sur votre machine, laisser toutes les valeurs par défaut, et le consulter

1. Mettez à jour le serveur appache (ou installez-le si il ne l'est pas…) et installez le module ssl pour ce serveur
    - **yum update httpd**
    - **yum install mod_ssl**

2. Observez les modifications provoquées par l'installation du module ssl :
    - apparition du fichier /etc/httpd/conf.d/ssl.conf (ouvrez-le, et observez son contenu)
    - création d'une clef privée dans /etc/pki/tls/private
    - création d'un certificat autorité (auto signé) dans /etc/pki/tls/certs

3. Redémarrez le service appache serveur
    - **service httpd restart**

4. Lancez le navigateur firefox vers https://127.0.0.1
    - Constatez les messages d'avertissement : pas le bon nom de serveur, et certificat auto-signé

5. Lancez firefox vers https://localhost.localdomain
    - Constatez qu'il n'y a plus qu'un warning : *certificat auto-signé*

A l'issue de cette expérience, vous avez :

- installé le serveur https ;
- observé et interprété le message d'avertissement que votre navigateur vous donne à la consultation de ce serveur ;

mais :

- vous n'avez rien maîtrisé concernant le contenu du serveur ;
- vous n'avez rien maîtrisé concernant le certificat.

Etape 4

Maîtriser le contenu affiché par le serveur https

1. Créez le répertoire /var/www/ssl/serv1, y créer un fichier index.html
    - **echo « /var/www/ssl/serv1/serv1.html » > /var/www/ssl/serv1/index.html**

2. Ajoutez serv1.entreprise.com dans le fichier de résolution /etc/hosts

Au démarrage d'appache, plusieurs fichiers de configuration sont lus. Le premier, c'est /etc/httpd/conf/httpd.conf. Puis, tous les fichiers dont le nom se termine par .conf, et qui sont présents dans le dossier /etc/httpd/conf.d sont lus.
Parmi ces fichiers, il y a /etc/httpd/conf.d/ssl.conf, c'est celui-là qu'il faut modifier.

3. Faites une sauvegarde du fichier ssl.conf en ssl.back, afin de pouvoir retrouver la situation initiale...
4. Modifiez ssl.conf de manière à ce que le serveur https
    - affiche la page index.html située en /var/www/ssl/serv1 (DocumentRoot)
    - puisse être adressé sous le nom serv1.entreprise.com (ServerName)

5. Pensez à redémarrer le serveur pour que les changements de configuration soient pris en compte
6. Faites un essai : adressez https://serv1.entreprise.com depuis votre navigateur

Etape 5

Création d'une autorité de certification, utilisation du certificat auto-signé pour le serveur

Le but est de maîtriser le contenu du certificat que le serveur va utiliser. Dans un premier temps, on choisit de faire en sorte que le serveur utilise un certificat auto-signé. Il faut donc :

- créer une autorité de certification racine pour cette entreprise
- créer un certificat auto-signé
- faire en sorte que le serveur utilise ce certificat pour s'authentifier

1. Choisissez un nom de domaine : « entreprise »
2. Créez l'arborescence

   mkdir –p /etc/pki/entreprise/{req,certs,crl,newcerts,private}

   La structure des sous-dossiers et des fichiers utiles doit être créée dans le répertoire /etc/pki/entreprise/

   Il faut créer 4 sous répertoires :

   - req : pour mettre les requêtes de certificats
   - certs : pour mettre le certificat racine
   - crl : pour les listes de révocation
   - newcerts : pour mettre les certificats dérivés du certificat racine
   - private : pour mettre les clés privées

3. Créez les fichiers nécessaires

touch /etc/pki/entreprise/index.txt	crée un fichier index.txt vide
echo « 01 » >/etc/pki/entreprise/serial	crée et initialise le fichier serial
echo « 01 » >/etc/pki/entreprise/crlnumber	crée et initialise le fichier crlnumber

4. Editez le fichier /etc/pki/tls/openssl.cnf

   Rechercher la ligne dir = etc/pki/CA et la remplacer par dir = /etc/pki/entreprise

5. Créez la clé privée de l'autorité racine, la stocker dans le fichier entreprise.key, et stocker ce fichier dans le répertoire « private »

   openssl genrsa –out /etc/pki/entreprise/private/entreprise.key 1024

6. Créez la requête du certificat, stockée dans /etc/pki/entreprise/req/entreprise.csr. Pour cela, on a besoin de la clé privée, et de renseigner certaines infos comme : pays, ville, etc… Ne pas mettre de mot de passe sur cette requête de certificat.

   openssl req –new –key /etc/pki/entreprise/private/entreprise.key –out /etc/pki/entreprise/req/entreprise.csr

7. Créez le certificat X509 autosigné de l'entreprise, stocké dans le fichier /etc/pki/entreprise/certs/entreprise.crt.

   Pour cela, on a besoin :
   - de choisir une durée de validité (choisissez 365 jours par exemple)
   - de spécifier le nom du fichier qui contient la requête
   - de la mention de la clé privée avec laquelle ce certificat sera signé

   Choisissez de signer avec la clé que vous avez créée au 5. : il s'agit donc d'un certificat auto-signé.

   openssl x509 –req –days 365 –in /etc/pki/entreprise/req/entreprise.csr –out /etc/pki/entreprise/certs/entreprise.crt –signkey /etc/pki/entreprise/private/entreprise.key

   A ce stade, vous avez créé la structure mère de l'entreprise :
   - son certificat auto-signé
   - signé par une clé privée créée localement

   Il faut maintenant associer le certificat au serveur web.

8. Faites en sorte que le serveur appache utilise le certificat auto-signé pour le fournir à un client qui se connecte
   - Modifiez ssl.conf de manière à ce que le serveur https :
     - utilise le certificat auto-signé /etc/pki/entreprise/certs/entreprise.crt (SSLCertificateFile)
     - utilise la clé privée /etc/pki/entreprise/private/entreprise.key (SSLCertificateKeyFile)
   - Pensez à redémarrer le serveur pour que les changements de configuration soient pris en compte
9. Testez en adressant votre serveur https://serv1.entreprise.com
   - Observez et interpréter l'avertissement de sécurité, accepter le certificat (pas de façon définitive)
   - Vérifiez que la page est la bonne
   - Vérifiez que le serveur utilise le bon certificat : le certificat racine de votre entreprise…

Etape 6

Création d'une autorité de certification pour l'entreprise, utilisation d'un certificat dérivé pour le serveur, reconnaissance du certificat racine de l'entreprise par le navigateur du client

Il faut :

- créer une autorité de certification, et donc créer une clé privée et un certificat auto-signé pour cette CA ;
- créer une clé privée et un certificat pour le serveur (signé par la CA) ;
- attribuer son certificat au serveur ;
- démarrer un client https qui va consulter le serveur ;
- récupérer le certificat du serveur avec le navigateur ;
- faire en sorte que le navigateur approuve le certificat racine de l'autorité de certification de l'entreprise ;
- observer que le client n'affiche plus d'alerte de sécurité à la consultation du serveur.

1. Reprenez l'étape 5 si vous voulez recréer une autorité de certification. Passez directement à la suite si vous voulez utiliser celle déjà créée…
2. Créez l'arborescence pour le certificat du serveur dans /etc/pki/serv1. Créez et initialisez les fichiers nécessaires.

   mkdir –p /etc/pki/serv1/{req,certs,crl,newcerts,private}

   touch /etc/pki/serv1/index.txt

   echo « 01 » >/etc/pki/serv1/serial

   echo « 01 » >/etc/pki/serv1/crlnumber

3. Créez la clé privée du serveur

   openssl genrsa –out /etc/pki/serv1/private/serv1.key 1024

4. Créez la requête de certificat pour le serveur. Pensez à renseigner correctement le FQDN de votre serveur serv1.entreprise.com.

   openssl req –new –key /etc/pki/serv1/private/serv1.key –out /etc/pki/serv1/req/serv1.csr

5. Créez le certificat X509 du serveur, stocké dans le fichier /etc/pki/serv1/certs/serv1.crt. Attention, ce certificat ne sera pas auto-signé. Il faut donc :
   - faire référence au certificat racine
   - préciser avec quelle clé privée vous allez signer ce certificat : c'est bien celle de l'autorité de certification. C'est l'autorité racine qui signe les certificats dérivés.

openssl ca –in /etc/pki/serv1/req/serv1.csr –cert /etc/pki/entreprise/certs/entreprise.crt –keyfile /etc/pki/entreprise/private/entreprise.key –out /etc/pki/serv1/certs/serv1.crt

6. Observez le contenu du répertoire /etc/pki/entreprise/newcerts
7. Faites en sorte que le serveur appache utilise le certificat que vous avez créé pour lui.
    - Modifiez ssl.conf de manière à ce que le serveur https :
        - utilise le certificat auto-signé /etc/pki/serv1/certs/serv1.crt (SSLCertificateFile)
        - utilise sa clé privée /etc/pki/serv1/private/serv1.key (SSLCertificateKeyFile)
        - connaisse le certificat de l'autorité qui a créé son certificat à lui /etc/pki/entreprise/certs/entreprise.crt (SSLCACertificateFile à dé-commenter)
    - Pensez à redémarrer le serveur pour que les changements de configuration soient pris en compte

8. Testez en adressant votre serveur https://serv1.entreprise.com
    - observez et interpréter l'avertissement de sécurité
    - vérifiez que le certificat utilisé est bien celui contenu dans serv1.crt

A ce stade, le serveur possède un certificat valide, qui lui vient d'une autorité de certification. C'est maintenant le problème du navigateur de savoir s'il fait confiance à cette autorité. Dans notre exemple, on voudrait que le navigateur reconnaisse l'autorité de certification de l'entreprise.

9. Dans le navigateur, supprimez l'éventuelle exception pour le serveur serv1
    - Allez sur la page du serveur https ;
    - Allez dans Edit/Preferences, Advanced, Encryption, View Certificates, Servers :
        - Si, à l'occasion des consultations prcédentes, vous n'avez pas accepté l'exception pour le serveur, le certificat de serv1 n'est pas dans la liste, et le navigateur affiche l'avertissement de sécurité ;
        - si vous avez déjà accepté l'exception pour ce serveur, vous voyez le certificat de serv1 dans la liste, et le navigateur vous a affiché la page sans avertissement. Dans ce cas, supprimez le certificat de serv1 de la liste afin de supprimer cette exception, rechargez la page, vous obtenez à nouveau l'avertissement.

10. Dans le navigateur, ajoutez l'autorité de certification de l'entreprise dans la liste des autorités de confiance
    - Allez sur la page du serveur https, vérifiez qu'il affiche bien l'avertissement de sécurité ;
    - Allez dans Edit/Preferences, Advanced, Encryption, View Certificates, Authorities :
        - vous voyez la liste des autorités reconnues par votre navigateur, l'autorité « Entreprise » n'y est pas bien-sûr.
        - cliquez sur Import, déplacez-vous dans l'arborescence, et sélectionnez le certificat entreprise.crt
        - choisissez « Trust this CA to identify web sites »

11. Testez à nouveau : vous n'avez plus de message d'avertissement.

Etape 7

Création de certificat pour les clients, authentification mutuelle client-serveur

On souhaite maintenant restreindre l'accès au service. Seuls les clients que le serveur sera capable de reconnaître doivent pouvoir gagner l'accès au service.

La stratégie est la même, mais dans l'autre sens cette fois : on va créer des certificats pour les clients (par groupe, par machine, par utilisateur, etc..), et c'est le serveur qui va décider s'il fait confiance au certificat que lui propose le client.

Dans le serveur, cette décision aura un caractère automatique, un certificat non valide sera refusé, et le serveur ne fournira pas le service. Souvenez-vous, dans la situation inverse, quand le navigateur vérifie le certificat du serveur, la décision n'est pas automatique, et c'est l'utilisateur qui, au final, décide.

Une autre différence réside dans le format de codage du certificat client. En effet, puisque le certificat est attaché au client, et que le détenteur du certificat doit aussi posséder la clé privée associée au certificat, il va falloir trouver le moyen de fournir l'ensemble « clé privée + certificat » au navigateur. C'est à cela que sert le format pkcs12 : il code l'ensemble certificat plus clé privée associée dans un seul fichier, protégé par un mot de passe. C'est le fichier pkcs12 qui sera importé dans le navigateur.

Il faut :

- Dans le serveur activer la fonction de reconnaissance du client par certificat, cela va se faire pour l'accès à un répertoire particulier par exemple. Dans notre exemple, en reprenant la configuration établie à l'étape 6, on va laisser l'accès public à la page serv1.entreprise.com, mais restreindre l'accès à la page serv1.entreprise.com/prive aux seuls clients qui possèdent un certificat valide.
- Créer un couple clé privée, certificat pour le client.
- Formater ce couple au format pkcs12, et l'enregistrer dans le navigateur.

1. Paramétrer le serveur.

   Dans la section <VirtualHost> du fichier de configuration d'appache (/etc/httpd/conf.d/ssl.conf), il faut ajouter une section :

   <Location « /prive »>
   Options Indexes

   SSLRequireSSL

   SSLVerifyClient require

   SSLVerifyDepth 10

   </Location>

« prive » est le nom du répertoire qui contient les pages auxquelles on souhaite restreindre l'accès. Attention, il s'agit d'un chemin relatif au DocumentRoot : /prive doit être dans le répertoire mentionné en DocumentRoot (pour cet exemple /var/www/ssl/serv1/prive)

2. Créez le repertoire /var/www/ssl/serv1/prive et placez y un fichier index.html explicite :

   echo « /var/www/ssl/serv1/prive/index.html » > /var/www/ssl/serv1/prive/index.html

3. Essayez d'y accéder avec le navigateur : https://serv1.entreprise.com/prive, l'accès vous est refusé (Secure Connection Failed). Attention : méfiez vous du cache de votre navigateur si vous faites plusieurs essais !
4. Créez l'arborescence pour ranger les fichiers liés au certificat du client
5. Créez une clé pour votre client et générez la requête de certificat (on peut faire tout dans la même commande)

   openssl req –newkey rsa:1024 -keyout /etc/pki/client/private/client.key –out /etc/pki/client/req/client.csr

6. Générez le certificat, et signez-le avec la clé de votre autorité de certification

   openssl ca –in /etc/pki/client/req/client.csr –cert /etc/pki/entreprise/certs/enterprise.crt –keyfile /etc/pki/enterprise/private/enterprise.key –out /etc/pki/client/certs/client.crt

7. Observez le contenu du répertoire /etc/pki/entreprise/newcerts

8. Intégrez la clé privée du client et son certificat dans un fichier de type pkcs12

   openssl pkcs12 –export –in /etc/pki/client/certs/client.crt –inkey /etc/pki/client/private/client.key –out /etc/pki/client/client.p12

9. Importez le certificat client dans votre navigateur et vérifiez :
   - Allez sur la page du serveur https://serv1.entreprise.com/prive, vérifiez que l'accès est bien refusé : Secure Connection Failed
   - Allez dans Edit/Preferences, Advanced, Encryption, View Certificates, YourCertificates :
     - o normalement, la liste est vide ;
     - o cliquez sur Import, déplacez-vous dans l'arborescence et choisissez le fichier client.p12
     - o vous comprenez pourquoi le système vous a demandé un mot de passe au moment de la création du fichier pkcs12…
     - o une fois importé, le certificat client apparaît dans la liste
   - Rechargez la page https://serv1.entreprise.com/prive, observez…
   - C'est cool, non ?

Etape 8

Utilisation des listes de révocation

Un des employés de l'entreprise s'est fait voler son ordinateur. Le voleur possède donc le certificat client, et compte bien s'en servir pour accéder aux données du répertoire privé…

Là les gamins, il y a du suspens…

…il faut créer et maintenir une liste des certificats révoqués.

1. Pour révoquer un certificat, il faut connaître son numéro de série. Les numéros des certificats générés par votre autorité sont rangés dans le fichier /etc/pki/entreprise.index.txt :

   tail /etc/pki/entreprise/index.txt

   Repérez le numéro d'index du certificat que vous voulez révoquer : **02** par exemple. Ne confondez pas avec le numéro de série qui lui est plus long (un truc du genre 1306020027117Z)

2. Copiez la clé privée de l'autorité de certification dans /etc/pki/entreprise/private/cakey.pem
3. Copiez le certificat racine de l'autorité dans /etc/pki/entreprise/cacert.pem
4. Révoquez le certificat que vous voulez révoquer :

   openssl ca –revoke /etc/pki/entreprise/newcerts/02.pem

5. Générez le fichier qui contient la liste des certificats révoqués :

   openssl ca –gencrl –out /etc/pki/entreprise/crl/entrepriseCRL.pem

6. Modifiez la configuration d'appache (ssl.conf) pour qu'il tienne compte de votre fichier liste de révocation :

   SSLCARevocationFile /etc/pki/entreprise/crl/entrepriseCRL.pem

7. Testez depuis le navigateur : vous n'avez plus accès à la page « prive », et normalement, votre navigateur vous dit pourquoi.