TP n°1
Cryptographie

Chiffrement et signature de messages

TP de 2h.
Démarrez les machines sous Linux

Objectifs de ce TP

Apprendre à utiliser GnuPG, gérer ses clés, chiffrer et signer.

L'application de la cryptographie au chiffrement des courriels est expliquée ici : http://www.pgpi.org/doc/pgpintro/
Voir aussi : http://www.francoz.net/doc/gpg/ ou http://www.pgpi.org

Exercice 1 : Envoi de spam *(à refaire rapidement ou à passer…)*

Dans cet exercice, nous allons voir que le SMTP (*Simple* Mail Transport Protocol), c'est-à-dire le protocole d'envoi de courriel, n'a pas été conçu de manière à être sûr.
Un serveur smtp tourne sur la machine `smtp.xxx.fr`

1. Connectez-vous au port 25 de ce serveur
2. Présentez-vous au serveur par un `helo <domaine>`
3. Déclarez l'émetteur du courriel avec `mail from: <emetteur>`
4. Déclarez le récepteur du courriel avec `rcpt to: <recepteur>`
5. Donnez le corps du mail avec `data`
6. Quittez avec `quit`

Le déroulement typique d'un envoi de courriel est le suivant :

```
> telnet smtp.xxx.fr 25
Trying 139.zzz.yyy.xxx…
Connected to smtp.xxx.fr (139.zzz.yyy.xxx).
Escape character is '^]'.
220 xxx.fr ESMTP version 0.12a (Linux 2.4)
helo domaine.fr
250 copyleft.xxx.xxx.fr
mail from: toto@titi.com
250 Ok
rcpt to: truc@machin.fr
250 Ok
data
354 End data with <CR><LF>.<CR><LF>
Ceci est un test
.
250 Ok: queued as B30A14BC00
```

```
quit
221 Bye
Connection closed by foreign host.
```

Exercice 2 : Mise en pratique de la cryptographie avec GnuPG

Dans cet exercice, vous apprendrez à utiliser les fonctionnalités fondamentales de GnuPG (GNU Privacy Guard) : générer des clés cryptographiques et les utiliser pour chiffrer ou signer numériquement des fichiers.

Attention : l'ordre des arguments sur la ligne de commande a un sens.

1. Commencez par générer votre paire de clés. Pour savoir comment faire cette opération, reportez-vous à la page de manuel du programme GnuPG (`man gpg`) ou consultez les pages Web indiquées en introduction. Pensez à protéger vos clés par une passphrase correcte (éventuellement, se reporter à `man passwd`, section "*hints for user passwords*").
   Note : pour aider le générateur de nombres aléatoires, le logiciel vous conseille d'utiliser le disque dur, le réseau, la souris et le clavier pendant la génération de clés. Pour avoir suffisamment d'entropie, il faut bouger (très) énergiquement la souris.
2. Profitez-en pour générer aussi le certificat de révocation (voir la documentation de l'option `–gen-revoke`). Pourquoi faut-il le générer maintenant et à quoi sert-il ? Dans quels répertoires et fichiers se trouvent vos clés ?
3. Demandez à votre voisin de droite de vous envoyer sa clé **publique** par courriel (voir la documentation des options `--export` et `--armor`).
   Remarque : cette procédure est risquée, puisqu'un attaquant pourrait facilement intercepter votre courriel et mettre sa clé à la place (attaque de type "*man in the middle*"). Nous verrons une autre façon de transmettre la clé dans l'exercice suivant.
4. Importez sa clé dans votre trousseau (option `--import`).
5. Chiffrez un fichier avec cette clé (voir la documentation des options `--output`, `--encrypt`, `--recipient` et `--armor`). Envoyez le fichier chiffré à son destinataire par courriel.
   Note : vous n'êtes plus en mesure de déchiffrer le fichier vous-même (puisque seule la clé privée associée à la clé publique utilisée pour le chiffrement peut le faire), à moins que vous ne vous ajoutiez comme destinataire du message.
6. Déchiffrez le courriel que votre voisin de gauche vous a envoyé (option `--decrypt`).
7. Recommencez les opérations 3 à 6, avec votre voisin de gauche, cette fois-ci en ***signant*** le message au lieu de le chiffrer (voir les options `--clearsign` et `--verify`).

Afin de mieux comprendre le fonctionnement des différentes commandes du logiciel GnuPG, nous l'avons utilisé en ligne de commande. Cependant, les opérations décrites ci-dessus peuvent être facilitées par l'utilisation d'une interface graphique telle que le greffon *Enigmail* pour Mozzila Thunderbird, GPGOE pour Outlook Express, EudoraGPG pour Eudora ou le script 'Encrypt This!" pour le webmail de GMail (attention ce dernier n'est pas très sûr…). Voir aussi KGpg.

Exercice 3 : Utilisation d'un serveur de clés publiques

Afin d'échanger des clés entre personnes physiquement éloignées, on passe généralement par un serveur de clés publiques. Un tel serveur maintient une base de données de clés et dispose d'une interface (généralement web) permettant de rechercher / récupérer la clé publique d'une personne donnée. Recherchez un serveur de clefs sur le Net ou, si vous travaillez sur votre propre machine, essayer d'installer votre serveur de clefs (PKI).

1. Ajoutez votre clé **publique** sur le serveur de clés. Pour ce faire, utilisez GnuPG avec les options `--keyserver` et `--send-keys`.
2. Recherchez / choisissez une clé sur le serveur, puis importez-la dans votre trousseau (options `--keyserver` et `--recv-keys`).

Les informations que vous trouvez sur le serveur sont en fait ce qu'on appelle des "*certificats*". Ce terme désigne simplement l'association d'une identité (nom de personne, adresse de courriel, …) et d'une clé publique.

Pour assurer un certain niveau de confiance dans les clés que l'on récupère sur les serveurs de clés publiques, on s'appuie sur un réseau de confiance (TrustNet). Le principe est le suivant : si *A* signe le certificat de *B* et si *B* signe celui de *C*, *A* peut par transitivité avoir une certaine *confiance* dans le fait que l'identité présente dans le certificat concernant *C* est correcte.

Dans le logiciel GnuPG, le découpage est en fait plus fin : il est possible d'associer un niveau de confiance (parmi : `full`, `marginal`, `don't trust` et `unknown`) au possesseur d'une clé, pour sa capacité à signer d'autres certificats. Ensuite, le logiciel GnuPG est capable de déterminer automatiquement qu'une clé est de confiance si :
- Elle est signée par suffisamment de clés :
  - Vous l'avez signée vous-même
  - Elle a été signée par *une* personne à laquelle vous faites confiance totalement (ultimate/full)
  - Elle a été signée par *trois* personnes auxquelles vous faites confiance marginalement.
- Le chemin de signatures retour (de la clé utilisée jusqu'à vous) est inférieur à cinq étapes.

1. Si vous avez pu vérifier la provenance de la dernière clé que vous avez ajoutée à votre trousseau (par exemple, en utilisant le "*fingerprint*", option `--edit-key` puis `fpr`), alors créez un certificat en signant cette clé (option `--sign-key`).
2. Ajoutez une **autre** clé (choisissez parmi vos camarades), **ne la signez pas** et affectez lui un niveau de confiance (option `--edit-key` puis `trust`). Répétez l'opération avec différentes clés et différents niveaux de confiance.
3. Chiffrez un document avec chacune des clés et observez le comportement du logiciel.

Exercice 4 : Quelles sont les primitives cryptographiques utilisées par GPG ? Rédigez un rapport sur les services et le fonctionnement de GPG. Rapport à rendre avant la fin du TP.