Institut National Universitaire Champollion	L2 Informatique UE Réseau 1 TP2 : GPG

Objectifs :

Le but de ce TP est de vous familiariser avec la notion de signature électronique et de chiffrement clé publique / clé privée en utilisant le logiciel GPG. GPG est une version libre du logiciel PGP (Pretty Good Privacy) crée par Philip Zimmermann. Bien qu'il existe des clients graphique (GPA par exemple), nous allons utilisez l'outil le plus basique (mais aussi le plus puissant) : le programme gpg en mode texte.

Introduction :

Le format OpenPGP est le standard ouvert issu de PGP (*Pretty Good Privacy*, basé sur l'algorithme de chiffrement asymétrique d'ElGamal) et largement déployé, en particulier dans les distributions Linux ,à travers l'outil GnuPG. Deux principaux logiciels ont adopté le format OpenPGP : GPG qui est gratuit et PGP qui est payant.
Notez que GnuPG est également disponible sous Windows même si dans le cadre de ce TP nous ne l'utiliserons qu'avec Linux. Toutes les manipulations qui seront présentées dans la suite sont donc reproductibles sous Windows.

Dans ce TP vous pouvez utiliser le manuel en ligne de GPG :

https://www.gnupg.org/gph/fr/manual.html

Sous linux :

Dans un terminal vérifiez que le logiciel est bien installé avec la commande

$ gpg –version

Vous accéderez à la documentation par la commande # man gpg
et un aperçu des commandes disponibles peut être généré avec # gpg –help

Génération d'un couple de clés

Avant toute utilisation, il est fortement recommandé de générer son couple de clé pour pouvoir chiffrer et authentifier ses messages. Utilisez :
gpg - -full -generate-key
et implémentez le modèle suivant :
• Algorithme : RSA et RSA
• Taille de clé : 2048

• Date d’expiration de la clé : jamais

Durant la procédure, une passphrase vous sera demandée. Cette même passphrase deviendra un élément incontournable pour votre utilisation de GPG donc ne tapez pas n’importe quoi !

➢ Quel est le rôle de cette passphrase ?
➢ A quoi correspondent les champs pub sub et uid ?

Vous pouvez lister les clés publiques de votre trousseau : gpg - - list-key
trouvez la commande pour lister vos clés privées.

Exportez ensuite vos deux clés dans des fichiers de la forme :nomPubKey.gpg et nomPrivKey.gpg où vous remplacerez « nom » par votre nom.
La commande :

gpg --output nomPubKey.gpg --armor --export "Identifiant"

permet d’exporter votre clé publique.

➢ Que signifie l'option --armor ?

Trouvez la commande pour exporter votre clé privée et faites-le.

Une fois les deux clés stockées dans des fichiers, échangez les _____ au sein du binôme (par USB, par mail...).

➢ Qu’avez-vous échangé ? Pourquoi ?

A la réception de cette clé, stockez-la dans votre trousseau de clés grâce à l'option - -import

➢ Produisez une capture d’écran de l’import.
➢ A priori, quels types de clés détenez-vous ?
➢ La structure stockant les clés est un trousseau de clés. Quelles sont les options permettant de lister les différents types de clés ?
➢ Comparez les résultats produits par ces deux commandes.

Chiffrement d’un texte.

Commencez dans un premier temps par saisir un petit texte de votre choix dans un éditeur de texte et sauvegardez-le dans le fichier message.txt
Chiffrez ensuite le message. La sélection de la clé se fait à l'aide de l'option --recipient "Identifiant" et le chiffrement grâce à l'option –encrypt. Si tout s'est bien déroulé, un fichier message.txt.gpg a été crée.
Vérifiez que le contenu du message est illisible.
➢ Supprimez le fichier d'origine, puis essayez de déchiffrer son contenu : quelle commande utilisez-vous ?
Une fois que tout fonctionne, répétez l'opération en chiffrant cette fois-ci le message avec la clé publique de votre collègue et envoyez-lui le résultat pour déchiffrement.
➢ Présentez graphiquement le processus en précisant les commandes employées dans chaque étape (chiffrement, déchiffrement) et les valeurs de paramètres.

Signature électronique.

L'objectif est de signer un message numériquement puis de le faire valider par votre collègue.

Pour générer une signature, vous pouvez utiliser l'instruction suivante :
gpg --default-key "Identifiant" --armor --detach-sign message.txt

➢ Quelle est l’extension du fichier de signature généré ? Le fichier message.txt a-t-il été modifié par la commande ?

Transmettez le message en clair et la signature à votre collègue pour qu'il en vérifie l'intégrité.

➢ Indiquez sur votre CR la commande permettant de vérifier la signature.
➢ Utilisez l’option –clearsign à présent pour signer votre fichier. Décrivez son résultat.

Comparez les signatures de votre message et du même document où vous avez passé une lettre de minuscule à majuscule.

1. Vérifiez la signature d'un message que votre voisin vous enverra.
2. Vérifiez la signature d'un message qui a été modifié après signature. Que se passe-t'il ?