

Informatique Générale, Systèmes d'exploitation, TP2

Le but de ce TP est de se familiariser avec l'outil Gnu Privacy Guard afin de pouvoir encrypter ou signer des données. Il est possible d'utiliser l'outil directement en ligne de commande ou à travers une interface graphique. Pour ce TP, nous utiliserons *kgpg* qui est installé en standard sur les machines de l'université.

Exercice 1 : Configuration de GPG

1. Démarrer *kgpg* nécessite de taper son nom en ligne de commande. Pour cela, il faut ouvrir un terminal. Allez dans la barre en haut de votre écran, choisissez *Applications* puis *Outils Système*, et finalement *terminal*. Une fenêtre devrait apparaître, dans laquelle vous allez pouvoir taper des commandes.
2. Tapez *kgpg*. Lors du premier lancement, le logiciel pose plusieurs questions, validez les choix par défaut.
3. La première étape consiste à créer un couple de clefs publique/privée. Dans le menu *keys*, choisissez *Generate Key Pair*. Entrez votre nom/prénom et votre adresse e-mail de l'université, et validez.
4. Choisissez un mot de passe, il protégera votre clef au cas où quelqu'un aurait accès à votre machine. La génération des clefs peut être longue, soyez patient. Si tout se passe bien, une nouvelle ligne apparaîtra dans l'interface du logiciel, avec votre adresse e-mail.
5. Nous devons maintenant rendre la clef publique accessible à d'autres personnes. Il y a plusieurs possibilités, quelles sont-elles ?
6. Exportez votre clef vers un serveur, celui proposé par défaut. S'agit-il de la clef publique ou privée ?

Exercice 2 : Envoi de données cryptées

Nous appellerons Bob l'étudiant qui veut envoyer des données secrètes, Alice celui qui doit les recevoir. Il est bien évident que personne d'autre ne doit connaître ces données. Cet exercice doit être fait deux fois, chaque étudiant prenant le rôle de Bob et Alice

1. Si Bob veut envoyer des données secrètes à Alice, quelle clef doit-il utiliser ? S'agit-il de sa clef publique, privée, ou la clef publique ou privée d'Alice ?
2. La première étape pour Bob consiste à récupérer la clef publique d'Alice. Heureusement, elle l'a placée sur un serveur. En cliquant sur *key server dialog* dans *kgpg*, importez la clef d'Alice.
3. Dans le terminal ouvert précédemment, tapez la commande ***konqueror***. (le point est important !). Si tout va bien, vous devriez voir apparaître un explorateur de fichier.
4. Grâce à cet explorateur, sélectionnez un fichier dont Alice ne connaît pas le contenu. Par exemple un fichier texte avec un message dedans, ou une image. Cliquez dessus avec le bouton droite et choisissez *actions*. Dans le sous-menu, choisissez *Encrypt File*.
5. Une fenêtre devrait apparaître, demandant quelle clef utiliser. Choisissez celle d'Alice, validez. Un nouveau fichier apparaît, quel est son nom ? Le fichier original est-il toujours là ?
6. Que se passe-t-il si vous cliquez sur ce fichier ? Bob peut-il décrypter ce fichier ?
7. Bob peut maintenant envoyer le fichier encrypté à Alice.
8. Une fois le fichier reçu et sauvegardé sur le disque, Alice peut le décrypter. Comment doit-elle faire ? Quel est le contenu de ce fichier ?

Exercice 3 : Signature de clef publique

1. Dans *kgpg* il est possible de signer une clef publique avec votre clef privée. Comment faire ?
2. Quel est l'intérêt ?

Exercice 4 : Hachage md5 et sha1

Nous allons nous intéresser aux outils permettant de calculer une valeur de hachage d'un fichier, afin, par exemple, de détecter les fichiers identiques.

1. Ouvrez un terminal et tapez la commande *md5sum* suivie de la touche ENTREE. Que se passe-t-il ? pour débloquer votre terminal, appuyez simultanément sur CTRL et c. Cela tue la commande *md5sum*.
2. Pour fonctionner *md5sum* a besoin d'un nom de fichier. Tapez la commande *md5sum /bin/ls* (en une seule ligne). Expliquez à quoi correspond l'affichage.
3. Avec un éditeur de texte, créez un fichier *test.txt* contenant la phrase *Test de md5sum*. Calculez la valeur de hachage md5 de ce fichier.
4. Faites une copie (*test2.txt*) du fichier *test.txt* à l'aide de la commande *cp*.
5. Vérifiez que les deux fichiers ont la même valeur de hachage.
6. Éditez le fichier *test2.txt* pour en modifier une unique lettre. Comment a changée sa valeur de hachage ?
7. Quelle est la taille (en bits) de la valeur de hachage calculée par *md5sum* ?
8. Calculez la valeur de hachage de vos fichiers de test avec la commande *sha1sum*. Quelle est la différence avec la commande *md5sum* ?
9. Sur combien de bits la valeur de hachage de *sha1* est-elle calculée ? Quel est l'intérêt par rapport à *md5* ?
10. Téléchargez l'archive <http://deptinfo.unice.fr/~huet/L1/donnees.zip>, décompressez-le, et trouvez les fichiers identiques.