Introduction à la Sécurité

TP2

Chiffrement – Outil GPG :

**BINOMES**

1. **PRENOM NOM : Cheikh Tidiane GUEYE**
2. **PRENOM NOM : Alioune Badara FAM**

**Compte de rendu de : Cheikh Tidiane GUEYE**

IMPORTANT :

* Choisir qui sera Binôme 1 et qui sera Binôme 2
* Le TP est un travail personnel, chaque binôme rendra un rapport personnel de son travail

**Objectifs du TP**

⇒Se familiariser avec les fonctionnalités et la bibliothèque GPG

⇒Savoir utiliser les différents mécanismes de chiffrement symétrique et asymétrique

**Notez bien** : lorsque les informations qui vous sont fournies sont insuffisantes, faites une recherche dans l’aide qui vous est proposée par Linux avec la commande

« man *gpg* ».

1. I – L’outils GPG

* GNU Privacy Guard (GPG) est un logiciel permettant de chiffrer et/ou signer des documents (fichiers, emails, commits git...), fonctionnant principalement à partir d'un système de paires de clés publique et privée.

1. Exemple d’utilisation de gpg Pour un chiffrement symétrique :
   * Pour chiffrer un fichier avec un chiffrement symétrique, utiliser la commande **gpg** avec l’option **--symmetric** ou **-c** : *gpg --symmetric <Fichier-à-chiffrer>*  
     Cela va créer un fichier binaire d’extension .gpg. Durant le chiffrement, il vous sera demandé une passephrase à ne pas oublier et qui sera utiliser pour déchiffrer le fichier
   * Pour chiffrer un fichier et générer un fichier chiffré au format texte ASCII, ajouter l’option –armor : *gpg --symmetric --armor <Fichier-à-chiffrer>*  
     Cela va créer un fichier texte d’extension .asc.
   * Il est également possible de changer l’algorithme de chiffrement utilisé avec l’option --cipher-algo et en précisant un algorithme proposé par GnuPG :   
     *gpg --symmetric --cipher-algo AES256 <Fichier-à-chiffrer >*  
     La liste des algorithmes peut être obtenue en utilisant la commande ***gpg –version***
   * Pour déchiffrer le fichier, il suffit d’utiliser la commande **gpg** avec l’option **--decrypt** ou -d :  
     *gpg --decrypt < Fichier-chiffré >*Il vous sera demandé la passephrase utilisée pendant le chiffrement.
   * Par défaut le fichier déchiffré est écrit sur la sortie standard, utiliser l’option --output pour indiquer le fichier de sortie :  
     *gpg -- decrypt --output <Fichier\_Claire> < Fichier-chiffré >*

***Tâche 1 : Chiffrement symétrique***

1. *Créer un dossier* ***symetrique***
2. *Placer vous dans le dossier et Créer un fichier* ***Prenom\_Nom.txt*** *contenant votre Prénom, nom, numéro carte étudiant, ville de naissance, telephone,*
3. *Chaque Binôme utilisera un algorithme de* ***chiffrement symétrique différent de son choix*** *pour chiffrer le fichier en* ***mode ascii***
4. *Noter la* ***passephrase*** *dans un fichier appelé* ***motdepasse.txt*** *pour ne pas l’oublier*
5. *Envoyer le fichier chiffré à votre deuxième Binôme ainsi que la passephrase pour le déchiffrement*
6. *Le Deuxième Binôme devra placer le fichier chiffré reçu dans le répertoire* ***symetrique*** *avant de le déchiffré*
7. Exemple d’utilisation de gpg Pour un chiffrement asymétrique :
   * Génération de la paire de clé public privé : gpg --full-generate-key

Génére une paire de clés en utilisant moins de valeurs par défaut, et demande de manière interactive les valeurs correspondantes (type de clé, votre nom, votre adresse email, et une passephrase pour utiliser la clé privée)

* + Pour envoyer une clé publique à quelqu’un, il faudra d’abord l’exporter avec la commande : gpg --output nom.pub --export --armour **uid**

Où uid est l’identité de la clé (adresse email par exemple)

* + Pour importer la clé publique reçue et contenue dans un ficher nom.pub, il faut utiliser la commande : gpg --import nom.pub
  + Pour Exporter votre la clé privé, utiliser la commande : gpg --output nom.priv --export --armour **uid**

La clé privée ne doit jamais être envoyer à votre Binome

***Tâche 2 : Gestion des paires de clés***

1. *Créer un dossier* ***asymetrique***
2. *Générer une paire de clé de votre choix parmi la liste des clés en prenant le soin de donner votre nom, votre adresse email universitaire et une passephrase)*
3. *Noter la* ***passephrase*** *dans un fichier appelé* ***motdepasse.txt*** *pour ne pas l’oublier*
4. *Vérifier que la clé est bien créée*
5. *Exporter votre clé publique en le nommant en* ***nomdefamille.pub***
6. *Exporter votre clé privée en le nommant en* ***nomdefamille.priv (***La clé privée ne doit jamais être envoyer à votre Binome ***)***
7. *Envoyer votre clé publique à votre Binôme*
8. *Placer la clé publique reçu de votre Binôme dans le dossier* ***asymetrique***
9. *Importer la clé publique de votre Binôme*
10. *Vérifier que la clé publique de votre Binôme est bien importée*
    * Chiffrer un document avec un algorithme asymétrique nécessite de connaitre la clé publique du destinataire du document.   
      *gpg --encrypt --armor -r uid <Fichier-à-chiffrer >*L’option -r ou --recipient permet de préciser le uid du destinataire afin de chiffrer le document avec la clé publique de ce dernier
    * Déchiffrer un document nécessite uniquement la clé privée associée à la clé publique ayant permis de le chiffrer :  
      gpg --decrypt *--output <Fichier\_Claire> < Fichier-chiffré >*

***Tâche 3 : Chiffrement asymetrique***

1. *Copier le fichier* ***Prenom\_Nom.txt*** *du dossier* ***symetrique*** *dans le dossier* ***asymetrique***
2. *Placez vous dans le dossier* ***asymetrique*** *et chiffrez le fichier* ***Prenom\_Nom.txt*** *en utilisant la clé publique de votre binôme importée durant la tache 2*
3. *Envoyer le fichier chiffré d’extension* ***.asc*** *à votre Binôme*
4. *Placer le fichier chiffré reçu de votre binôme dans le dossier asymetrique*
5. *Déchiffrer le fichier chiffré reçu de votre binôme*

Documents à déposer sur la plateforme FAD :

* Compresser le dossier **symetrique** après avoir vérifier que tous les fichiers de la Tâche 1 s’y trouve
* Compresser le dossier **asymetrique** après avoir vérifié que tous les fichiers des Tâches 2 et 3 s’y trouve
* Déposer les deux dossiers compressés sur la plateforme FAD