TP GnuPG : Sécurisation des échanges pour une équipe projet

Contexte :

Vous faites partie d'une équipe travaillant sur un projet sensible. Vous devez échanger des fichiers et des messages de manière sécurisée. Pour cela, chaque membre de l'équipe doit utiliser GnuPG pour chiffrer, signer, et vérifier les messages. Ce TP vous guide dans la mise en place de cette sécurité.

Objectifs spécifiques :

1. Créer et gérer des clés GPG pour protéger vos échanges.
2. Partager votre clé publique avec les membres de l'équipe.
3. Chiffrer un fichier ou un message avant de l'envoyer.
4. Vérifier l'identité des expéditeurs à l'aide des signatures numériques.

Étapes détaillées :

Étape 1 : Préparation de l'environnement

- Installez GnuPG si ce n'est pas déjà fait.
    - **Linux :** sudo apt install gnupg.
    - **macOS :** brew install gnupg.
    - **Windows :** Téléchargez GPG4Win.

Étape 2 : Création de votre clé GPG

1. **Lancez la commande suivante :**

gpg --full-generate-key

2. **Répondez aux questions posées :**
    - **Type de clé :** Appuyez sur **Entrée** pour choisir RSA et RSA.
    - **Taille de la clé :** Tapez 4096 pour une meilleure sécurité.
    - **Durée de validité :** Indiquez 0 (pas de limite).
    - **Identité :** Entrez votre **nom**, **adresse e-mail** et, éventuellement, un commentaire.
    - **Mot de passe :** Choisissez un mot de passe robuste.
3. **Confirmez la création de la clé.**
4. **Vérifiez que la clé a été générée :**

gpg --list-keys

Vous devriez voir votre clé publique dans la liste.

TP GnuPG : Sécurisation des échanges pour une équipe projet

Étape 3 : Exporter votre clé publique

1. Exportez votre clé publique pour la partager avec vos collègues :

gpg --armor --export [votre_email] > public_key.asc

Un fichier public_key.asc sera créé. Envoyez ce fichier à un membre de votre équipe.

2. **Importer une clé publique reçue :** Si un collègue vous partage sa clé publique, importez-la :

gpg --import colleague_public_key.asc

3. **Lister les clés publiques disponibles :**

gpg --list-keys

Étape 4 : Chiffrer un message ou un fichier

1. Créez un fichier texte contenant des informations sensibles, par exemple :

echo "Le mot de passe du serveur est: Projet2024!" > secret.txt

2. Chiffrez ce fichier avec la clé publique de votre collègue :

gpg --encrypt --recipient [email_de_collegue] secret.txt

Cela créera un fichier chiffré secret.txt.gpg.

3. **Envoyez le fichier chiffré (**secret.txt.gpg**) à votre collègue.**

Étape 5 : Déchiffrer un fichier reçu

1. Si vous recevez un fichier chiffré, utilisez la commande suivante pour le déchiffrer :

gpg --decrypt secret.txt.gpg

2. Entrez votre mot de passe lorsque demandé. Vous verrez le contenu déchiffré.

Étape 6 : Signer un fichier ou un message

1. **Signer un fichier :**

gpg --clear-sign message.txt

Cela génère un fichier signé message.txt.asc.

2. **Envoyez ce fichier signé à votre collègue.**

Étape 7 : Vérifier une signature

1. Si vous recevez un fichier signé, vérifiez la signature :

gpg --verify message.txt.asc

2. GnuPG confirmera si la signature est valide et associée à la clé publique de l'expéditeur.

Résumé des commandes clés

Action	Commande
Générer une paire de clés	gpg --full-generate-key
Exporter une clé publique	gpg --armor --export [email] > public_key.asc
Importer une clé publique	gpg --import colleague_public_key.asc
Lister les clés	gpg --list-keys
Chiffrer un fichier	gpg --encrypt --recipient [email] fichier
Déchiffrer un fichier	gpg --decrypt fichier.gpg
Signer un fichier	gpg --clear-sign fichier
Vérifier une signature	gpg --verify fichier.asc

Exercice final : Mise en situation

1. Créez une clé GPG si ce n'est pas déjà fait.
2. Envoyez votre clé publique à un collègue (ou simulez cet échange en créant une autre clé pour "lui").
3. Chiffrez un message pour votre collègue et envoyez-le.
4. Déchiffrez un fichier reçu.
5. Signez un message et demandez à votre collègue de vérifier votre signature.

En suivant ces étapes, vous maîtriserez les bases de GnuPG et serez prêt à sécuriser vos échanges dans un cadre professionnel !