***Objectif :***

Comprendre le fonctionnement des clés asymétriques.

Savoir comment utiliser le protocole et la connexion SSH de manière plus sécurisée en utilisant les méthodes de chiffrement.

***Matériel nécessaire :***

Une machine Linux et une machine cliente Windows (ça peut être votre machine physique).

Avant de commencer ce TP, vérifier que vos 2 machines peuvent communiquer entre elles et que la machine linux a accès à internet.

Travail à effectuer :

Précédemment, vous avez installé le protocole SSH sur une machine Linux, et donc fait de votre machine un serveur SSH.

Vous vous êtes connecté en direct sur la machine en utilisant le nom d’un utilisateur et un mot de passe depuis votre client Windows.

Maintenant, vous allez configurer la connexion pour se connecter avec un SSH par clés asymétriques, un système de chiffrement.

# Rappel : Faites Attention !

Ne confondez pas SSH, ssh et sshd !

SSH est le protocole de communication.

ssh est le programme client permettant de se connecter au serveur

sshd est le daemon ssh du serveur ssh qui communique par défaut sur le port 22.

Pourquoi faire la distinction entre tout ça ?

Principalement pour les fichiers de configuration.

Le fichier /etc/ssh/ssh\_config est le fichier de configuration du client ssh. On a rarement besoin d’y toucher.

Tandis que le fichier /etc/ssh/sshd\_config est le fichier de configuration du serveur ssh. Et vous allez le modifier dès à présent.

# Changement du numéro de port du serveur ssh

Manipulations sur la machine Linux « principale ».

sudo nano /etc/ssh/sshd\_config.

La ligne ‘Port 22’, signifie que votre serveur SSH écoute sur le port 22. Il s’agit du port par défaut. Changez-le. Mettez le port 22320.

Vous avez changé le fichier de configuration, vous devez le relancer le service ssh pour que les modifications soient prises en compte.

systemctl restart sshd

De manière simple, tenter de vous connecter d’une machine cliente (Windows ou Linux) sur la machine Linux principale, en ssh. Pour que la machine cliente sache sur quel port elle doit communiquer avec le serveur, il faut lui préciser :

ssh [login]@[ip] -p 22320

# Authentification par clés avec Putty (Linux / Windows)

Manipulation sur la machine Windows et sur la Linux principale.

Au lieu de s’authentifier par mot de passe, c’est-à-dire de manière « classique », les utilisateurs peuvent s’authentifier grâce au chiffrement asymétrique et son couple de clefs privées/publiques.

Pour paramétrer cette authentification par clé asymétrique, c’est la machine dite ‘cliente’ qui va générer les clés, et qui ensuite enverra la clé publique au serveur.

## Générer les clés

Il est possible de se connecter avec Putty, et le système de clés.

Cela se fait grâce à l’utilitaire ‘Puttygen’ que vous avez installé à même temps que Putty. Lancez-le.

Cliquer sur le bouton ‘Generate’.

Le programme va générer la clé privée et la clé publique. Mais il a besoin de votre aide pour cela. Il faut le ‘réveiller’, pour cela, bougez votre souris rapidement sur la fenêtre de Putty pour faire avancer la barre de ‘chargement’.

Une fois fait, vous voyez votre clé publique apparaitre, elle est en clair.

Pour plus de sécurité, il est recommandé de saisir une ‘passphrase’ qui sera le ‘mot de passe’ pour créer et **chiffrer** votre clé publique. Donc elle ne sera plus ‘en clair’.

Saisissez une phrase que vous retiendrez. Attention, vous aurez besoin de vous rappeler de cette passphrase, car elle sera demandée à chaque connexion.

Ensuite, enregistrez en cliquant sur ‘Save **public** key’.   
Enregistrez là ou bon vous semble, sous le nom que vous souhaitez. Choisissez quand même un nom parlant !   
Ensuite, enregistrez la clé privée ‘Save **private** key’. Nommez là comme bon vous semble, mais donnez-lui l’extension .ppk .

## Envoi de la clé publique au serveur Linux

À présent, envoyons la clé publique au serveur Linux sur lequel on voudrait se connecter.

Ouvrez Putty, attention, sans fermer Puttygen. Connectez-vous au serveur Linux en connexion ssh simple ou classique, c’est-à-dire avec le mot de passe. **Déplacez-vous** dans le dossier **~/.ssh**

Si ce dossier n’existe pas (ce qui est probablement le cas), **vous devrez le créer**.

Saisissez ensuite la commande suivante, mais sans l’envoyer ! :

echo votre\_cle >> authorized\_keys

Sur Windows, retournez sur Puttygen et copiez votre clé publique.

Ensuite, sur le prompt de votre serveur dans la fenêtre de Putty, vous allez copier votre sélection en faisant **‘shift’+’insert’** à la place de « votre\_cle ».

Faites ‘Entrée’ à la suite de votre commande.

Pour vérifier, faite : nano authorized\_keys et vous devriez voir dans le fichier votre clé publique. La copie s’est donc bien passée.

Déconnectez-vous de la session (logout). **Fermez** Putty et Puttygen.

## Connexion avec les clés

Ouvrez une nouvelle fois Putty. À gauche dans les catégories, cliquez sur ‘Connexion’, puis sur ‘SSH’, et sur ‘Auth’, et sur ‘Credentials’. Grâce au bouton ‘Browse’ au niveau de ‘Private key file for authentication’ recherchez sur votre machine le fichier contenant votre clé privée, celui que vous avez nommé terminant par .ppk.

À gauche, retournez sur la catégorie ‘Session’. Entrez l’adresse IP de votre serveur, ainsi que le port ssh. Avant de faire ‘Open’, donnez un nom à votre connexion dans le champ ‘Saved Sessions’, et cliquez sur ‘Save’. Cela enregistrera tous les paramètres que vous venez de saisir pour cette connexion. À présent, vous pouvez cliquer sur ‘Open’.

Comme d’habitude, on vous demande le nom de votre utilisateur, et ensuite la passphrase que vous avez paramétré à partir de Putty. Vous voilà connectés de manière sécurisée grâce aux clés et à la passphrase !!