TP : Connexion SSH sous Windows avec OpenSSH

Objectif

Ce fascicule traite de l'utilisation de **SSH** (Secure Shell) qui est un protocole de communication qui permet une connexion cryptée. Nous ne traiterons ses utilisations que sur un poste Windows 11

Il se base sur un algorithme de cryptage asymétrique, c'est à dire que la clé de chiffrement n'est pas la même que la clé de déchiffrement. Chaque client génère une paire de clé et envoie sa clé publique au serveur distant pour pouvoir s'y connecter. La clé privée ne doit jamais être déplacée pour limiter tout risque de divulgation.

Il faut distinguer les termes :
- **SSH** : le protocole de communication
- **ssh** : le programme client permettant de se connecter au serveur
- **sshd** : le serveur (ssh daemon) écoutant sur le port 22 par défaut

Partie 1 : Connexion SSH avec mot de passe

Étape 1 : Activer le serveur SSH

1. Ouvrir Paramètres > Système (ou Applications) > Fonctionnalités facultatives.

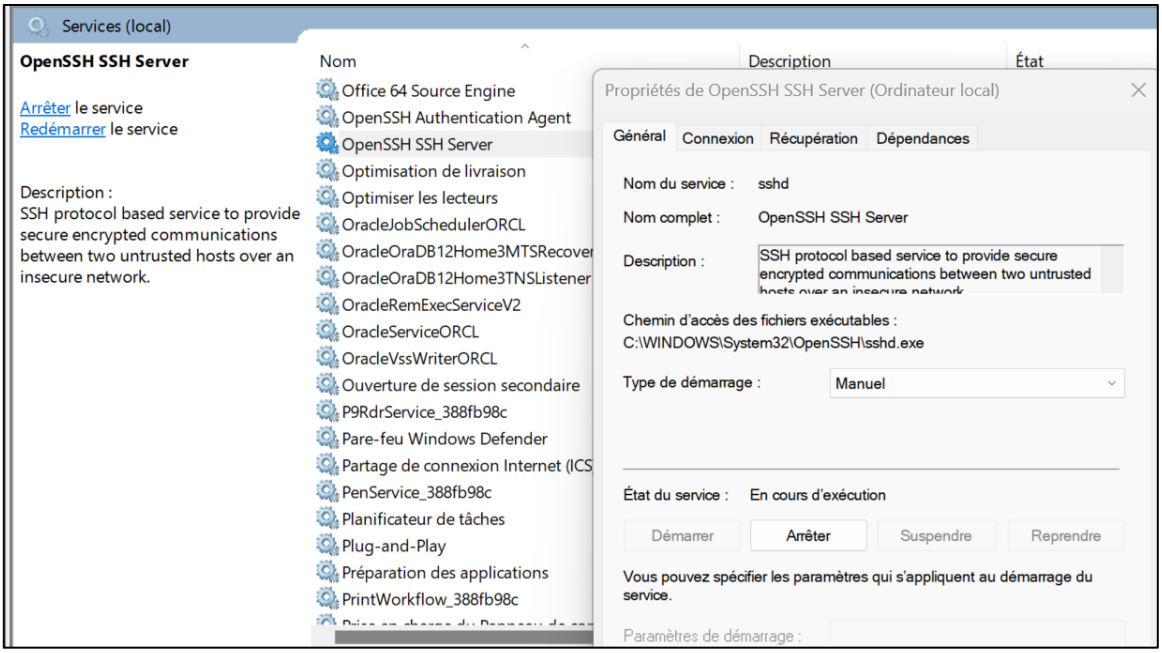
2. Installer OpenSSH Server s'il n'est pas encore installé.

3. Redémarrer le service SSH :

- Dans PowerSchell

  Start-Service sshd
  Set-Service -Name sshd -StartupType 'Automatic'

- Ou bien via « services »



Étape 2 : Autoriser l'authentification par mot de passe

1. Éditer le fichier C:\ProgramData\ssh\sshd_config :

Vérifier la ligne suivante :

PasswordAuthentication yes

2. Sauvegarder et redémarrer le service :

Restart-Service sshd

Étape 3 : Tester la connexion

Dans un terminal PowerShell :

ssh <nom_utilisateur>@localhost

Entrer le mot de passe Windows lorsque demandé.

Partie 2 : Connexion SSH avec clé publique

Étape 1 : Générer une paire de clés

La création de la paire de clé se fait avec la commande : **ssh-keygen**

Il existe 2 types de clés : **RSA et DSA** correspondent à deux algorithmes différents. On peut utiliser l'un ou l'autre. Chacune pouvant être de longueur différente : 1024, 2048, 4096 bits.

Commande pour une clé RSA de 1024 bits : **ssh-keygen -t rsa –P ''passpass''-b 1024**
Avec :

- -t : type de clé (DSA ou RSA)
- -b : nombre de bytes

Dans PowerShell ou Git Bash taper la commande suivante
(paramètres par défaut:

ssh-keygen

Valider l'emplacement par défaut : C:\Users\<Utilisateur>\.ssh\id_rsa

Deux fichiers ont été créés (dans le dossier ~/.ssh/) dans le poste du client :

- **id_rsa** : contient la clé privée.
- **id_rsa.pub** : contient la clé publique, c'est elle qui sera mise sur le serveur.

Étape 2 : Ajouter la clé publique sur le serveur

1. Copier le contenu du fichier id_rsa.pub dans :

C:\Users\<Utilisateur>\.ssh\authorized_keys

Sur le poste Client : Il faut envoyer la clé générée par le client sur le poste du serveur.

Pour ce faire, on tape la commande suivante :

scp ~/.ssh/id_rsa.pub user@serveur:C:\Users\user\.ssh\authorized_keys

Cette commande envoie le la clé publique dans le fichier **/home/serveur/.ssh/authorized-key** (qui sera modifié ou créé) afin que le serveur accepte le nouveau client.

Remarque : il vous sera demandé le mot de passe du compte serveur.

Sur le poste Serveur :

On interdit tout accès avec mot de passe pour des raisons de sécurité. Une connexion **ssh** ne pourra se faire qu'avec une paire de clé publique/privée.

Désactivation de l'authentification par mot de passe et activation de l'authentification par clé publique dans le fichier : /**etc/ssh/sshd_config**

PubkeyAuthentication yes

PasswordAuthentication no

2. S’assurer que le fichier authorized_keys a les bons droits :

Il ne doit pas être accessible en écriture pour d'autres utilisateurs.

Étape 3 : Éditer le fichier sshd_config

1. Commenter les lignes suivantes si elles existent :

#Match Group administrators
 # AuthorizedKeysFile
__PROGRAMDATA__/ssh/administrators_authorized_keys

2. Vérifier la ligne suivante :

AuthorizedKeysFile .ssh/authorized_keys

3. Redémarrer le service :

Restart-Service sshd

Étape 4 : Tester la connexion avec la clé

ssh <nom_utilisateur>@localhost

Il ne doit plus demander de mot de passe.

Etape 5 : SFTP

La commande **sftp** permet de transférer un fichier entre deux serveurs, de manière similaire à **ftp** mais sécurisée. Le chemin du serveur peut être indiqué en absolu ou relatif à partir du répertoire de base. Pour utiliser **sftp**, vous devez connaître l’arborescence exacte des répertoires de la machine distante.
Il faut que **SSH** soit installé sur les deux machines pour effectuer le transfert.
Syntaxe : ***sftp user@IP-serveur***
Cette commande donne accès à :
sftp >
Tapez **help** pour accéder à la liste des commandes internes à **sftp**.

Commandes utiles :

quit : pour quitter la session en cours

get : récupère un fichier présent sur le serveur FTP et le place sur votre machine

put : transfère un fichier de votre disque dur vers le serveur

ls : permet de lister le contenu du répertoire courant côté FTP

pwd : affiche le nom du répertoire courant sur le FTP

delete et **rm** : effacent un fichier sur le FTP

mkdir : créé un répertoire sur le FTP

Exemple :
Pour envoyer un fichier : sftp> put fichier
Pour télécharger un fichier : sftp> get fichier