Documentation Technique

Configuration SSH pour Machine Virtuelle Debian

Connexion depuis Windows Host

Table des matières

1. Vue d'ensemble

Ce document décrit la procédure complète pour configurer une connexion SSH sécurisée entre une machine Windows hôte et une machine virtuelle Debian. L'authentification sera réalisée par clé publique/privée, offrant une sécurité renforcée par rapport à l'authentification par mot de passe.

1.1. Architecture

Machine hôte : Windows (génération et stockage de la clé privée) Machine virtuelle : Debian (serveur SSH avec clé publique autorisée)

2. Prérequis

2.1. Côté Windows

- Windows 10 (version 1809+) ou Windows 11 avec client OpenSSH installé
- Accès administrateur pour certaines configurations
- Logiciel de virtualisation (VirtualBox, VMware, Hyper-V, etc.)

2.2. Côté Debian VM

- Debian 11 ou supérieur installé
- Accès root ou utilisateur avec privilèges sudo
- Connexion réseau configurée (NAT ou Bridge)
- Accès internet pour installer les paquets

3. Configuration de la machine virtuelle Debian

3.1. Mise à jour du système

Avant toute installation, mettez à jour le système :

```
sudo apt update
sudo apt upgrade -y
```

3.2. Installation du serveur SSH

Installez le paquet openssh-server :

```
sudo apt install openssh-server -y
```

3.3. Vérification du service SSH

Vérifiez que le service SSH est actif et démarré :

```
sudo systemctl status ssh
```

Si le service n'est pas actif, démarrez-le :

```
sudo systemctl start ssh
```

Activez le démarrage automatique au boot :

```
sudo systemctl enable ssh
```

3.4. Configuration du pare-feu (si UFW est activé)

Si le pare-feu UFW est actif, autorisez le trafic SSH :

```
sudo ufw allow ssh
```

Vérifiez le statut :

```
sudo ufw status
```

3.5. Récupération de l'adresse IP

Identifiez l'adresse IP de votre VM :

```
ip addr show
```

Ou de manière plus concise :

```
hostname -I
```

Note : Notez cette adresse IP, elle sera nécessaire pour la connexion depuis Windows.

3.6. Création du répertoire .ssh et configuration des permissions

Créez le répertoire .ssh dans le répertoire home de l'utilisateur. Ce dossier se situera à `/home/username/.ssh` (remplacez *username* par votre nom d'utilisateur).

```
mkdir -p ~/.ssh
```

Note : Le symbole ~ représente le répertoire home de l'utilisateur courant (/home/username).

Définissez les permissions appropriées (très important pour la sécurité SSH) :

```
chmod 700 ~/.ssh
```

Créez le fichier authorized_keys avec les bonnes permissions :

```
touch ~/.ssh/authorized_keys
chmod 600 ~/.ssh/authorized_keys
```

3.7. Configuration du fichier sshd_config

Pour autoriser les connexions SSH, vous devez vérifier et configurer le fichier `/etc/ssh/sshd_config`. Ce fichier contient tous les paramètres du serveur SSH.

Éditez le fichier de configuration :

```
sudo nano /etc/ssh/sshd_config
```

Vérifiez ou ajoutez/modifiez les lignes suivantes pour permettre les connexions SSH :

```
# Port d'écoute SSH (22 par défaut)
Port 22

# Permettre l'authentification par clé publique
PubkeyAuthentication yes

# Permettre temporairement l'authentification par mot de passe
# (sera désactivé plus tard pour plus de sécurité)
PasswordAuthentication yes
```

```
# Fichier contenant les clés publiques autorisées
AuthorizedKeysFile .ssh/authorized_keys
```

Important : Certaines lignes peuvent être commentées (précédées de #). Décommentez-les en supprimant le # au début de la ligne.

Après modification, sauvegardez le fichier (Ctrl+O, Entrée, Ctrl+X) et redémarrez le service SSH :

```
sudo systemctl restart ssh
```

4. Configuration sur Windows

4.1. Vérification du client OpenSSH

Vérifiez que le client OpenSSH est installé :

```
ssh -V
```

Si la commande retourne une erreur, installez OpenSSH via *Paramètres* > *Applications* > *Fonctionnalités facultatives* > *Ajouter une fonctionnalité* et recherchez "Client OpenSSH".

4.2. Génération de la paire de clés SSH

Ouvrez PowerShell ou CMD et générez la paire de clés avec l'algorithme Ed25519 (recommandé) :

```
ssh-keygen -t ed25519
```

Remarques :

- Appuyez sur Entrée pour accepter l'emplacement par défaut : `C:\Users\username\.ssh\id_ed25519`
- Le dossier `.ssh` sera créé automatiquement dans votre profil utilisateur Windows s'il n'existe pas déjà
- Vous pouvez définir une passphrase pour sécuriser davantage la clé privée (recommandé)
- Deux fichiers seront créés : `id_ed25519` (clé privée) et `id_ed25519.pub` (clé publique)

4.3. Alternative : Génération avec RSA

Si vous devez utiliser RSA (compatibilité avec des systèmes anciens) :

```
ssh-keygen -t rsa -b 4096
```

5. Transfert de la clé publique

5.1. Méthode 1 : Transfert automatique (recommandé)

Depuis Windows, utilisez la commande suivante pour transférer automatiquement la clé publique :

```
type "C:\Users\username\.ssh\id_ed25519.pub" | ssh username@hostip "cat >> ~/.ssh/authorized_keys"
```

Important : Remplacez *username* par votre nom d'utilisateur Debian et *hostip* par l'adresse IP de votre VM.

Exemple concret :

```
type "C:\Users\john\.ssh\id_ed25519.pub" | ssh john@192.168.1.100 "cat >> ~/.ssh/authorized_keys"
```

5.2. Méthode 2 : Transfert manuel

Si la méthode automatique ne fonctionne pas, vous pouvez copier manuellement la clé :

Étape 1 : Affichez le contenu de la clé publique sur Windows :

```
type "C:\Users\username\.ssh\id_ed25519.pub"
```

Étape 2 : Copiez le contenu affiché (la ligne complète commençant par ssh-ed25519)

Étape 3 : Sur la VM Debian, ajoutez la clé au fichier authorized_keys :

```
nano ~/.ssh/authorized_keys
```

Collez la clé publique, puis sauvegardez (Ctrl+O, Entrée, Ctrl+X).

Étape 4 : Vérifiez les permissions du fichier :

```
chmod 600 ~/.ssh/authorized_keys
```

6. Test de connexion SSH

6.1. Première connexion

Depuis Windows, testez la connexion SSH :

```
ssh username@hostip
```

Note : Lors de la première connexion, vous verrez un avertissement concernant l'authenticité de l'hôte. Tapez "yes" pour continuer.

6.2. Connexion avec clé privée spécifique

Si vous avez plusieurs clés ou que la clé n'est pas à l'emplacement par défaut :

```
ssh -i "C:\Users\username\.ssh\id_ed25519" username@hostip
```

6.3. Configuration du fichier config SSH (optionnel)

Pour simplifier les connexions futures, créez un fichier config :

Créez le fichier `C:\Users\username\.ssh\config` avec le contenu suivant :

```
Host debian-vm
    HostName 192.168.1.100
    User username
```

```
    IdentityFile ~/.ssh/id_ed25519
    Port 22
```

Après configuration, vous pourrez vous connecter simplement avec :

```
ssh debian-vm
```

7. Sécurisation avancée

7.1. Désactivation de l'authentification par mot de passe

Pour renforcer la sécurité, désactivez l'authentification par mot de passe sur Debian :

Éditez le fichier de configuration SSH :

```
sudo nano /etc/ssh/sshd_config
```

Modifiez ou ajoutez les lignes suivantes :

```
PasswordAuthentication no
PubkeyAuthentication yes
PermitRootLogin no
```

Redémarrez le service SSH :

```
sudo systemctl restart ssh
```

7.2. Modification du port SSH (optionnel)

Pour réduire les tentatives d'intrusion automatisées, changez le port par défaut :

Dans /etc/ssh/sshd_config, modifiez :

```
Port 2222
```

Puis redémarrez SSH et adaptez le pare-feu :

```
sudo ufw allow 2222/tcp
sudo ufw delete allow ssh
sudo systemctl restart ssh
```

7.3. Installation de Fail2Ban (protection contre les attaques par force brute)

```
sudo apt install fail2ban -y
sudo systemctl enable fail2ban
```

8. Dépannage

8.1. Erreur : Permission denied (publickey)

Causes possibles :

- Permissions incorrectes sur ~/.ssh ou ~/.ssh/authorized_keys
- Clé publique mal formatée dans authorized_keys

- PubkeyAuthentication désactivée dans sshd_config

Solutions :

```
chmod 700 ~/.ssh
chmod 600 ~/.ssh/authorized_keys
sudo systemctl restart ssh
```

8.2. Erreur : Connection refused

Vérifiez que le service SSH est actif :

```
sudo systemctl status ssh
```

Vérifiez que le pare-feu autorise le trafic SSH :

```
sudo ufw status
```

Vérifiez la connectivité réseau :

```
ping hostip
```

8.3. Erreur : Connection timed out

Causes possibles :

- Configuration réseau de la VM incorrecte (vérifiez le mode NAT/Bridge)
- Pare-feu Windows bloquant la connexion sortante
- Redirection de port manquante si la VM est en NAT

8.4. Mode verbose pour diagnostic

Pour obtenir des informations détaillées sur la connexion :

```
ssh -vvv username@hostip
```

9. Annexes

9.1. Tableau des permissions SSH

Élément	Permission	Description
~/.ssh	700 (drwx------)	Lecture/écriture/exécution pour le propriétaire uniquement
~/.ssh/authorized_keys	600 (-rw-------)	Lecture/écriture pour le propriétaire uniquement
~/.ssh/id_ed25519 (privée)	600 (-rw-------)	Lecture/écriture pour le propriétaire uniquement
~/.ssh/id_ed25519.pub (publique)	644 (-rw-r--r--)	Lecture pour tous, écriture pour le propriétaire

9.2. Commandes utiles récapitulatives

Action	Commande
Vérifier statut SSH	`sudo systemctl status ssh`
Redémarrer SSH	`sudo systemctl restart ssh`
Voir IP de la VM	`hostname -I`
Tester connexion SSH	`ssh username@hostip`
Vérifier logs SSH	`sudo tail -f /var/log/auth.log`
Copier fichier via SCP	`scp file.txt user@host:/path`

9.3. Ressources supplémentaires

- Documentation OpenSSH : https://www.openssh.com/
- Guide de sécurisation SSH : https://wiki.debian.org/SSH
- Fail2Ban documentation : https://www.fail2ban.org/

Fin du document