

Partage serveur



Présentation

OpenSSH est une suite d'outils open source utilisée pour la communication sécurisée sur un réseau. Voici une présentation de ses principales caractéristiques :

1. **Sécurité des communications** : OpenSSH assure la sécurité des communications en chiffrant les données échangées entre les systèmes. Il utilise des algorithmes de chiffrement robustes pour protéger les informations sensibles, empêchant ainsi les attaquants d'intercepter ou de modifier les données transitant sur le réseau.
2. **Connexions sécurisées** : OpenSSH prend en charge les connexions sécurisées entre des machines distantes via le protocole SSH (Secure Shell). Ce protocole permet aux utilisateurs de se connecter à des serveurs distants de manière sécurisée et d'exécuter des commandes à distance de manière fiable, même sur des réseaux non sécurisés.
3. **Authentification sécurisée** : OpenSSH propose plusieurs méthodes d'authentification sécurisée, y compris l'authentification par mot de passe, l'authentification par clé publique et l'authentification à deux facteurs, offrant ainsi une flexibilité pour répondre aux besoins de sécurité spécifiques de chaque environnement.
4. **Transfert de fichiers sécurisé** : OpenSSH inclut également des outils tels que scp (Secure Copy) et sftp (Secure File Transfer Protocol) pour transférer des fichiers de manière sécurisée entre des machines distantes. Ces outils utilisent SSH pour chiffrer les données lors du transfert, assurant ainsi la confidentialité et l'intégrité des fichiers échangés.
5. **Tunnels SSH** : OpenSSH permet de créer des tunnels SSH, également appelés tunnels SSH ou tunnels port forwarding, qui permettent de sécuriser et d'acheminer le trafic réseau à travers un tunnel chiffré. Cela peut être utilisé pour sécuriser les communications sur des réseaux non fiables ou pour accéder à des services internes à

distance de manière sécurisée.

6. **Gestion à distance sécurisée** : OpenSSH est largement utilisé pour la gestion à distance sécurisée des serveurs et des équipements réseau. Il permet aux administrateurs système d'accéder aux machines distantes de manière sécurisée, de configurer et de maintenir les systèmes à distance sans compromettre la sécurité.
7. **Open Source et documentation abondante** : OpenSSH est un logiciel open source largement utilisé dans l'industrie informatique. Il est bien documenté, avec une communauté active d'utilisateurs et de développeurs qui fournissent un support technique et des ressources en ligne pour aider les utilisateurs à tirer le meilleur parti de cette suite d'outils.

Site officiel [OpenSSH](#).

Installation

```
apt update && apt upgrade
apt install sshfs
```

Création des clés

Pour que le montage Serveur se fasse, j'utilise le protocole sshfs :

Lors de l'ouverture de la session utilisateur, la machine envoie une requetes de montage réseau SSHFS avec un utilisateur et une clé privée. Cette clé est comparée avec la clé publique, si les clés sont bien compatibles, le montage se fait.

Pour faire tout ceci, on se connecte avec un utilisateur lambda. On lance un terminal Ctrl+Alt+T et on génère les clés avec la commande suivante :

```
ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa): /home/mnteleve/.ssh/Montage-Eleves
Enter passphrase (empty for no passphrase): (laisser vide)
Enter same passphrase again: (laisser vide)
Your identification has been saved in /home/mnteleve/.ssh/Montage-Eleves
Your public key has been saved in /home/mnteleve/.ssh/Montage-Eleves.pub
The key fingerprint is:
SHA256:UL9NLEimhn2SU99SXVPn0WWWqlo0WgfPrqR9TK0gJgc
```

The key's randomart image is:

```
+---[RSA 3072]-----+
|           = .. oX|
|    o B + = .==|
|   . 0 o = B .o.|
|   . E  X = .|
|           S = * . |
|   . = = o . |
|           + B = . |
|           o o + |
|           . |
+-----[SHA256]-----+
```

```
ssh-keygen -t rsa
```

Generating public/private rsa key pair.

Enter file in which to save the key (/root/.ssh/id_rsa): /home/mtenseignants/.ssh/Montage-Profes

Enter passphrase (empty for no passphrase): (laisser vide)

Enter same passphrase again: (laisser vide)

Your identification has been saved in /home/mtenseignants/.ssh/Montage-Profes

Your public key has been saved in /home/mtenseignants/.ssh/Montage-Profes.pub

The key fingerprint is:

SHA256:IjpYR0Vm0trkiFSPJEgSxsNoR9K4kI8HY3wtbFj20bY

The key's randomart image is:

```
+---[RSA 3072]-----+
|0B0==+*          |
|XX+X.0+o         |
|+*+.+Bo .        |
|..+.o oE         |
| .. o . S        |
| o o . .         |
|. o              |
| .               |
|                 |
+-----[SHA256]-----+
```

Une fois la clé générée, on copie le contenu de la clé "Nom-Montage.pub" dans le dossier .ssh/authorized_keys de l'utilisateur de montage associé.

```
cp /home/mnteleve/.ssh/Montage-Eleves.pub /home/mnteleve/.ssh/authorized_keys
cp /home/mntenseignants/.ssh/Montage-Profes.pub /home/mntenseignants/.ssh/authorized_keys
```

Il ne faut pas oublier de placer la clé privé "Nom-Montage" dans le dossier .ssh du skel, ou changer le chemin d'accès sur le fichier /etc/profile.

```
cp /home/mnteleve/.ssh/Montage-Eleves* /etc/skel/.ssh/
cp /home/mntenseignants/.ssh/Montage-Profes* /etc/skel/.ssh/
touch /etc/skel/.ssh/authorized_keys
echo /home/mnteleve/.ssh/authorized_keys >> /etc/skel/.ssh/authorized_keys
echo /home/mntenseignants/.ssh/authorized_keys >> /etc/skel/.ssh/authorized_keys
```

Il faut ensuite que les droits d'accès soient correctent :

```
chmod 700 /home/mnteleve/.ssh
chmod 700 /home/mntenseignants/.ssh
chmod 644 /home/mnteleve/.ssh/*.pub
chmod 644 /home/mntenseignants/.ssh/*.pub
chmod 600 /home/mnteleve/.ssh/Montage-Eleves
chmod 600 /home/mntenseignants/.ssh/Montage-Profes
chmod 700 /etc/skel/.ssh
chmod 644 /etc/skel/.ssh/*.pub
chmod 600 /etc/skel/.ssh/Montage-Eleves
chmod 600 /etc/skel/.ssh/Montage-Profes
```

Montage automatique

Pour que le montage soit automatique, mais qu'il respecte en plus les droits différenciés entre Enseignants et Elèves, nous devons modifier le fichier /etc/profile :

```
groupe=$(groups)

if [[ $groupe = "Enseignants" ]];
then
    sshfs mntenseignants@192.168.1.1:/Serveur ~/Serveur -o IdentityFile=~/.ssh/Montage-
Profes -o StrictHostKeyChecking=accept-new
else
```

```
sshfs mnteleve@192.168.1.1:/Serveur ~/Serveur -o IdentityFile=~/.ssh/Montage-Eleves -o  
StrictHostKeyChecking=accept-new  
fi
```

Ce script permet de définir quel utilisateur et quelle clé SSH utiliser pour faire le montage réseau en fonction du groupe de l'utilisateur. Cette section est à ajouter à la suite de l'existant dans le fichier.

Il est impératif que le dossier `.ssh` contenant les deux clés soit présent dans le dossier `skel`, sans ça, le montage est impossible.

Revision #1

Created 2025-10-29 12:36:04 UTC

Updated 2025-10-29 12:36:07 UTC