Connexion en SSH par échange de clés

Windows - Linux Server

Tutoriel **SSH**

Yanis Dahman

Table des matières

Introduction

SSH (Secure Shell) est un protocole de communication sécurisé qui permet de se connecter à un ordinateur distant de façon sécurisée. Par défaut, la connexion **SSH** s'effectue avec un mot de passe. Cette méthode d'authentification n'est pas la plus sûre car votre mot de passe peut être dérobé ou deviné par des pirates qui pourraient dès lors accéder à votre serveur et voler vos données personnelles.

Pour plus de sécurité, il est conseillé de se connecter à un ordinateur distant en utilisant l'authentification par échange de clés **SSH**.

1.Les clés SSH ?

L'authentification par échange de clés SSH fonctionne en plaçant une clé publique sur l'ordinateur distant et en utilisant une clé privée depuis son ordinateur.

Ces deux clés (publique et privée) sont liées l'une à l'autre. C'est seulement en présentant la clé privée à la clé publique qu'il est possible se connecter.

Chaque clé se présente sous la forme d'une longue chaine de caractères enregistrée dans un fichier.

Pour plus de sécurité, on peut également protéger la clé privée avec une phrase secrète. Autrement dit, pour pouvoir utiliser la clé privée, il faudra saisir un mot de passe, ce qui renforce encore davantage la sécurité.

Les clés SSH peuvent être créées avec différents algorithmes de chiffrement :

- **DSA** : dangereux, il n'est plus pris en charge depuis OpenSSH 7.0
- **RSA** : acceptable si la longueur de la clé est de 3072 ou 4096 bits
- **Ed25519** : le plus sûr, c'est l'algorithme à privilégier aujourd'hui

2. Prérequis

On a besoin des différents matériels et logiciels pour la création d'une connexion SSH par certificat.

- Un PC client sous Windows
- Deux VM (Virtual Machines) linux → srv-home → srv-backup
- Le logiciel **Putty** pour se connecter en SSH au serveur
- Connaitre le logiciel **PowerShell** de Windows et sa ligne de commandes

3.Configuration des VM

Nous créons 2 VM, sur Debian, une nommé :

'srv-home' en 192.168.56.101

et l'autre nommé

'srv-backup' en 192.168.56.102

Premièrement, on installe SSH sur la machine "srv-home", on utilise la commande " apt install open-ssh server"
Ensuiteil faut changer le nom des 2 VM, grâce à la commande "nano /etc/hostname"
On redémarre la VM avec la commande "reboot"

4.Echange Debian vers Debian

On se rend sur srv-backup qui fait office de « **client** »

- Il faut se connecter en sio/sio et générer une paire de clé ssh avec la commande : ssh-keygen
- La commande ssh-keygen est un outil qui va générer une paire de clé privée et publique ssh

```
root@srv-backup:~# ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa): /root/.ssh/id_rsa
Created directory '/root/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /root/.ssh/id_rsa
Your public key has been saved in /root/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:Y6DZ/ooKAhk9KmIIYizFE+K4zbr1sYYiZ6J4gZ12pYU root@srv-backup
The key's randomart image is:
+---[RSA 3072]----+
|..o.             |
|++o              |
|*+o. ..          |
|*B .E+o.         |
|O+o.o+. S        |
|*.= o. . .       |
|+ooo  .          |
|B==o . .         |
|B*+.. ...        |
+----[SHA256]-----+
```

On peut afficher le **fingerprint** avec la commande suivante : ssh-keygen -lf .ssh/id_rsa

```
sio@srv-backup:~$ ssh-keygen -lf .ssh/id_rsa
2048 SHA256:XR1+KO+e8M1r6QLQJR6xWprMmerMyEt6zfiFwt6WauE sio@srv-backup (RSA)
```

Le **fingerprinting** ou « **prise d'empreinte** » est une technique probabiliste visant à identifier un utilisateur de façon unique sur un site web ou une application mobile en utilisant les caractéristiques techniques de son navigateur.

- On va donc copier notre empreinte sur srv-home pour qu'il nous reconnaisse en lui envoyant via la commande : ssh-copy-id –i sio@192.168.56.101

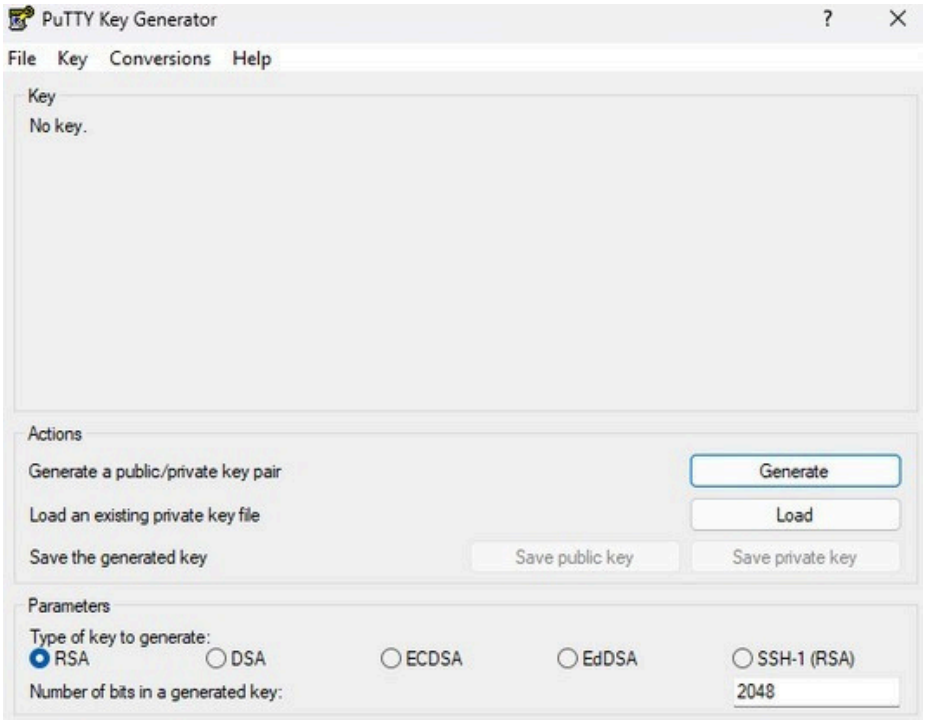
```
sio@srv-backup:~$ ssh-keygen -lf .ssh/id_rsa
2048 SHA256:XR1+KO+e8M1r6QLQJR6xWprMmerMyEt6zfiFwt6WauE sio@srv-backup (RSA)
sio@srv-backup:~$ ssh-copy-id -i sio@192.168.56.101
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/home/sio/.ssh/id_rsa.pub"
The authenticity of host '192.168.56.101 (192.168.56.101)' can't be established.
ECDSA key fingerprint is SHA256:3nxnJUBGrt/S4cvjpFMXeEPtzxn0Wu5LIyRhHxPnpJM.
Are you sure you want to continue connecting (yes/no)? yes
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are al
eady installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted now it is to ins
all the new keys
sio@192.168.56.101's password:

Number of key(s) added: 1

Now try logging into the machine, with:   "ssh 'sio@192.168.56.101'"
and check to make sure that only the key(s) you wanted were added.
```

Maintenant que srv-backup est connu par srv-home on peut se connecter en ssh

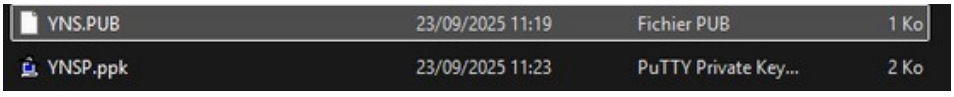
Nous sommes bien connecté à srv-home en ssh, on peut constater qu'il n'y a pas de mot de passe à rentrer car notre machine est connue par srv-home

5.Echange Windows vers Debian

Grâce à PuttyKeyGenerator, nous allons Générer une clé publique sur le poste Windows, il faudra ensuite sauvegarder cette clé



Cliquer sur « generate » puis il faudra bouger la souris pour générer le **fingerprint**
Il faudra sauvegarder la clé publique au format **xxx.pub** et la placer dans un répertoire



Il faut maintenant se connecter sur la machine Debian srv-home en utilisant le profil ‘sio’ Une fois connecté en sio il faut générer la paire de clé avec la commande ssh-keygen

```
sio@srv-home:~$ ssh-keygen
```

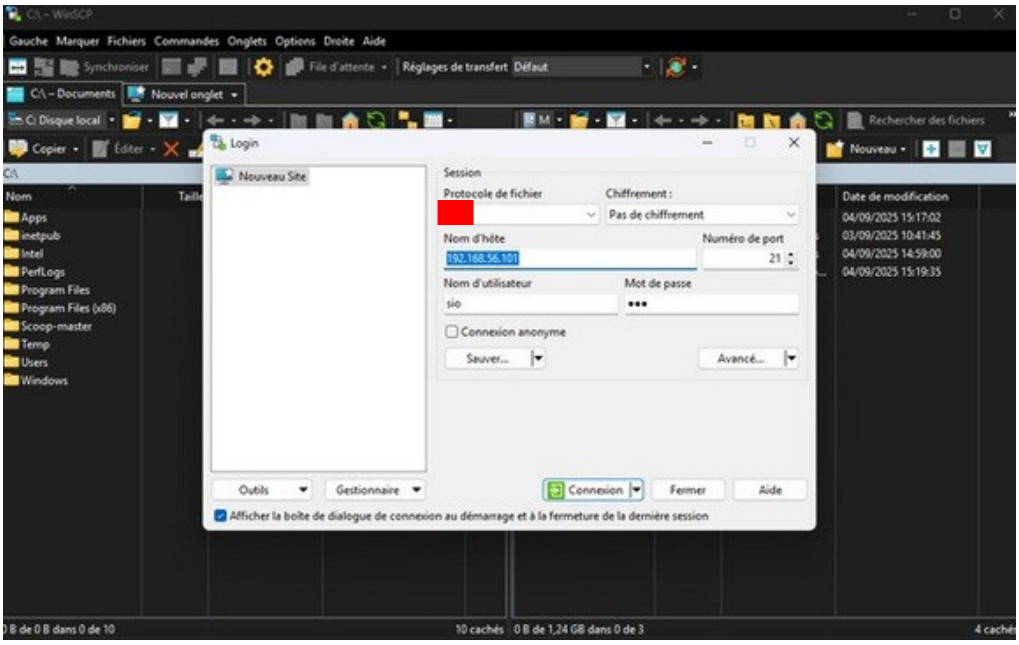
La paire de clé a bien été généré

```
root@srv-backup:~# ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa): /root/.ssh/id_rsa
Created directory '/root/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /root/.ssh/id_rsa
Your public key has been saved in /root/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:Y6DZ/ooKAhk9KmIIYizFE+K4zbrlsYYiZ6J4gZ12pYU root@srv-backup
The key's randomart image is:
+---[RSA 3072]----+
|..o.             |
|++o              |
|*+o. ..          |
|*B .E+o.         |
|O+o.o+. S        |
|*.= o. . .       |
|+ooo  .          |
|B==o . .         |
|B*+.. ...        |
+----[SHA256]-----+
```

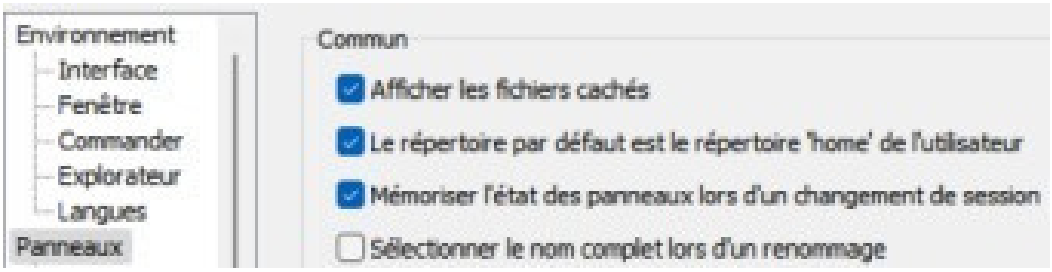
Il faut maintenant lancer WinSCP et pour cela, il faut se connecter à srv-home en **FTP**

Donc installer FTP car il y’a 1 paquet non installé pour la connexion

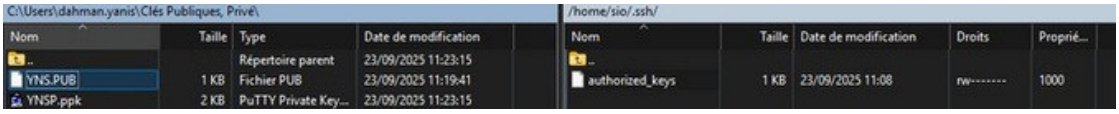
```
root@srv-home:~# apt install vsftpd_
```



Une fois connecté, on se rend dans les paramètres, panneaux, et on clique sur **afficher les fichiers cachés**, car sans sa nous ne trouvons pas le répertoires SSH.



Il faut maintenant copier la clé publique de notre machine hôte vers srv-home dans le répertoire /home/sio/.ssh

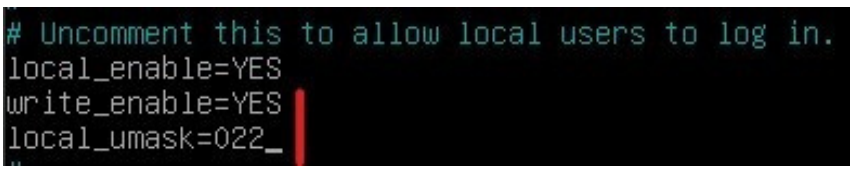


On observe une erreur lorsqu'on essaye de transférer la clé publique vers ce répertoire

Le problème se situe au niveau des permissions donc afin de le résoudre il faut
Il faut se rendre dans le le fichier /etc/vsftpd.conf

Le but va être de modifier le fichier pour pouvoir écrire sur le fichier

On ajoute ces 2 lignes de commande ce qui va autoriser les utilisateurs à se connecter et on leurs donne également les droits :



Il faut également faire des commandes qui vont donner la permission de transférer la clé publique

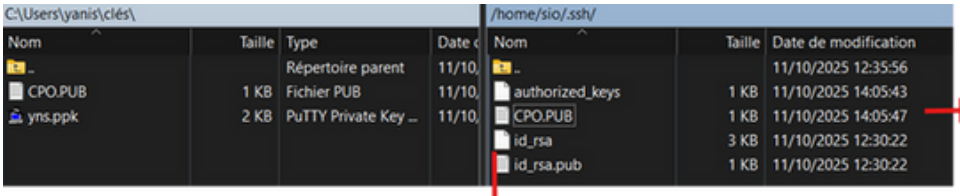
```
srv-home:~$ chmod 700 /home/sio/.ssh
```

Répertoire appliquant la commande

Nom de la commande
"**CH**ange **MOD**e"

Accorde le **droit** au propriétaire (**sio**)
→ **Lire, Ecrire, Exécuter**

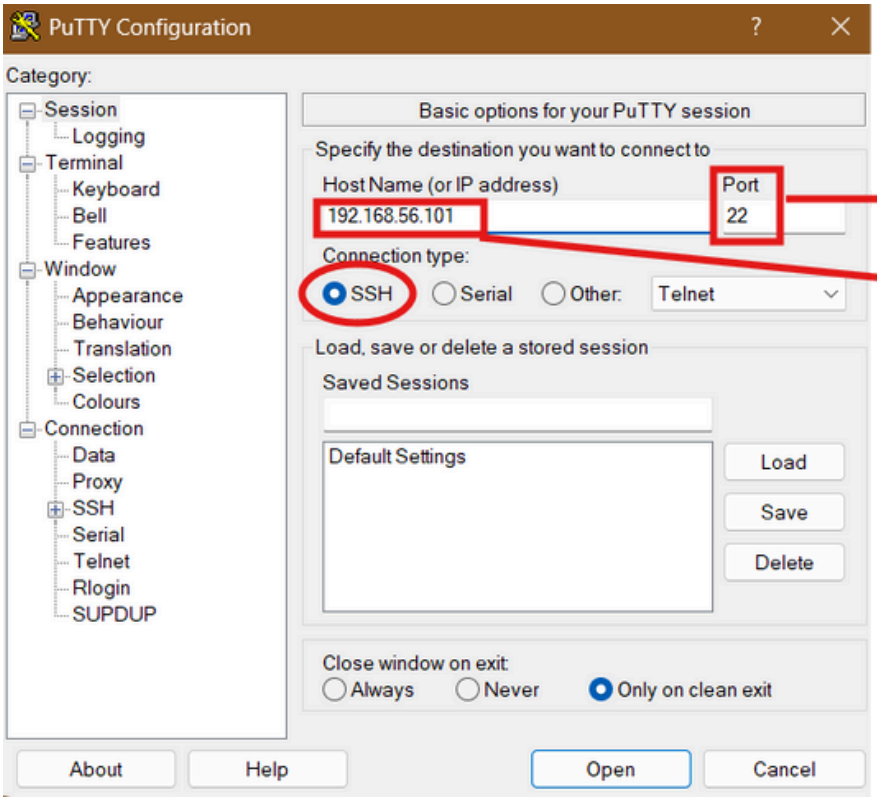
La clé publique à bien été transférer



Clé publique générer
via PuttyKeyGenerator

Clés publique/privé générer
via srv-home avec la
commande "ssh-keygen"

Il faut maintenant lancer Putty et mètres les paramètres suivants :

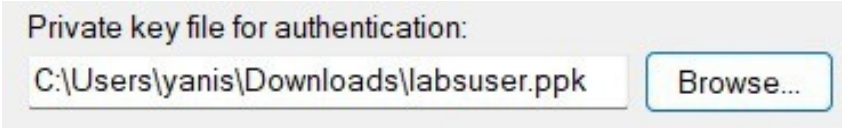


Port utilisée lors de la
connexion en SSH

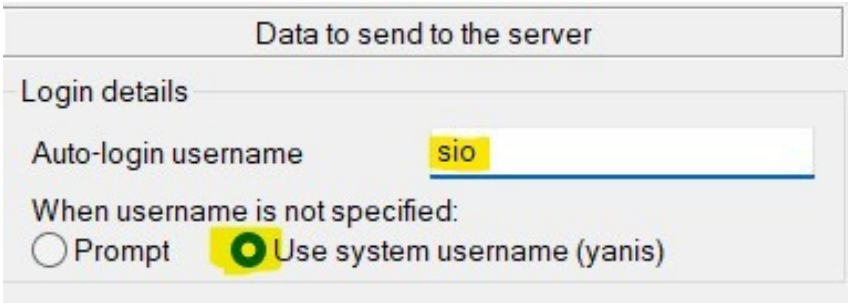
Adresse IP de la
machine srv-home

Maintenant se rendre dans Catégorie → Connexion → SSH → Auth → Credentials

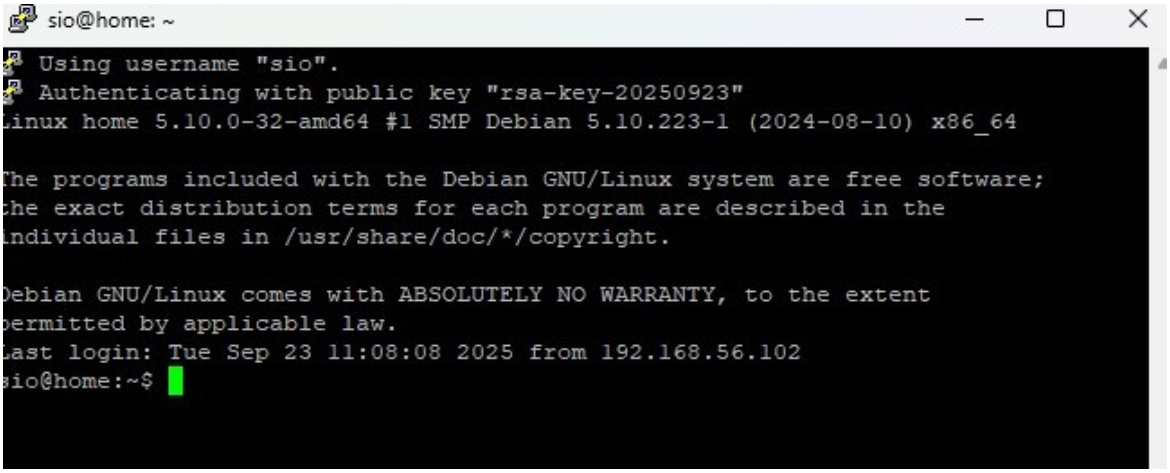
On peux maintenant mettre la clé privé enregistrer précédemment via PuttyKeyGenerator



Ensuite se rendre dans Catégorie → Connexion → Data et saisir "**sio**" et cocher "Use system username (.....)"



Saisir "**Open**" et on observe que nous sommes bien connecté en ssh (Le mot de passe n'est pas demandée lors de la connexion)



6. Conclusion

Les clés SSH privées et publiques sont bien installées et configurées avec succès sur le **serveur** et le **client**. On peut désormais se connecter en SSH par échange de clés SSH.