

Menu interactif pour gestion simplifiée des serveurs **Linux**

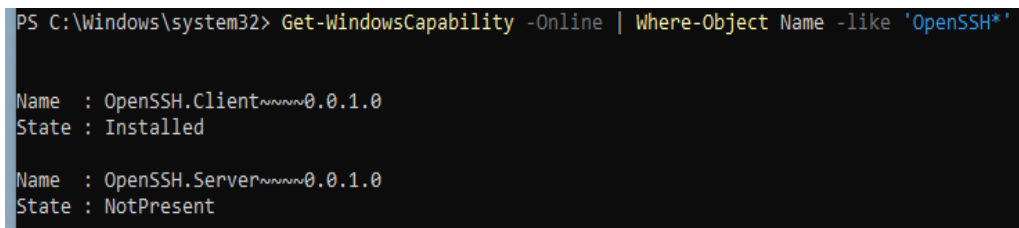
1. Connexion SSH sécurisée entre l'host (W11) et le guest (Debian12)

A- Sur l'host Windows 11(HOST), utilisez OpenSSH (intégré à Windows 10/11).

Pour installer **OpenSSH** avec PowerShell sur Windows 11, procédez comme suit :

1. Exécutez PowerShell en tant qu'administrateur.
2. Assurez-vous qu'OpenSSH est disponible.

« Get-WindowsCapability -Online | Where-Object Name -like 'OpenSSH*' »



```
PS C:\Windows\system32> Get-WindowsCapability -Online | Where-Object Name -like 'OpenSSH*'

Name : OpenSSH.Client~~~~0.0.1.0
State : Installed

Name : OpenSSH.Server~~~~0.0.1.0
State : NotPresent
```

3. Ensuite, installez les composants selon vos besoins.

Install the OpenSSH Client

« Add-WindowsCapability -Online -Name OpenSSH.Client~~~~0.0.1.0 »

Install the OpenSSH Server

« Add-WindowsCapability -Online -Name OpenSSH.Server~~~~0.0.1.0 »

Seul OpenSSH.Client est requis pour le host

4. Générez la clé SSH avec une longueur de 4096bits en exécutant :

« ssh-keygen -b 4096 »

5. Lorsque vous êtes invité à **"Enter file in which to save the key"**, appuyez sur Entrée pour accepter l'emplacement par défaut (**~/ssh/id_rsa**).

6. Entrez une passphrase sécurisée lorsque vous y êtes invité. Cette passphrase protégera votre clé privée. Choisissez une passphrase robuste, longue et unique.

7. Une fois la génération de clés terminée, le fichier **id_rsa.pub** contient votre clé publique. Uploadez ce fichier sur un service de partage sécurisé comme un serveur SFTP ou un service de stockage cloud chiffré.

B- Sur le guest Debian 12(GUEST)

1. Mettez à jour les paquets et installez le serveur **OpenSSH** :

```
« sudo apt update »  
« sudo apt install openssh-server »
```

2. Vérifiez le status du service **SSH** :

```
« sudo systemctl status ssh »
```

3. Si le service n'est pas démarré, démarrez-le et activez-le pour qu'il démarre automatiquement au redémarrage :

```
« sudo systemctl start ssh »  
  
« sudo systemctl enable ssh »
```

4. Créez un nouvel utilisateur non-root avec des privilèges sudo (remplacez 'votreprenom' par votre prénom) :

```
« sudo adduser votreprenomadm »  
  
« sudo usermod -aG sudo votreprenomadm »
```

5. Connectez-vous avec le nouvel utilisateur :

```
« su -votreprenomadm »
```

6. Installez wget pour télécharger le fichier de clé publique partagé :

```
« sudo apt install wget »
```

7. Téléchargez le fichier de clé publique partagé (remplacez <URL_DU_FICHER_PARTAGE> par l'URL réelle) :

```
« wget <URL_DU_FICHER_PARTAGE> »
```

8. Créez le répertoire .ssh et définissez les permissions appropriées :

```
« mkdir ~/.ssh »  
  
« chmod 700 ~/.ssh »
```

9. Ajoutez la clé publique au fichier `authorized_keys` et définissez les bonnes permissions :

```
« cat id_rsa.pub >> ~/.ssh/authorized_keys »
```

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

10. Modifiez le fichier de configuration

```
« sudo nano /etc/ssh/sshd_config »
```

11. Décommentez la ligne « **PubkeyAuthentication yes** » et remplacez « **#AuthorizedKeysFile** » par « **AuthorizedKeysFile .ssh/authorized_keys** ».

```
« PubkeyAuthentication yes »
```

12. Désactivez l'authentification root pour plus de sécurité en ajoutant « **PermitRootLogin no** ».

```
PubkeyAuthentication yes
PermitRootLogin no

# Expect .ssh/authorized_keys2 to be disregarded by default in future.
#AuthorizedKeysFile.ssh/authorized_keys
```

13. Redémarrez le service **SSH** pour prendre en compte les modifications :

```
« sudo systemctl restart sshd »
```

C- Connectez-vous depuis Windows (Host)

1. Trouvez l'adresse IP de la machine guest avec :

```
« ip addr show »
```

2. Sur PowerShell Windows :

```
« ssh root@ip_guest »
```

Assurez-vous d'utiliser un service de partage sécurisé et de supprimer le fichier partagé après l'avoir téléchargé sur le serveur

2. Réaliser l'outil à partir du fichier "menu_pas_fini.txt"

