

Guide d'installation de SSH sur Ubuntu en WSL

Introduction

L'installation de SSH sur Ubuntu via WSL (Windows Subsystem for Linux) permet d'accéder à votre système Ubuntu à distance, ce qui peut être pratique pour la gestion de votre environnement de développement. Ce guide vous expliquera étape par étape comment installer et configurer SSH sur Ubuntu en WSL.

L'installation est importante pour laisser l'accès à mon tuteur d'alternance.

Prérequis

Avant de commencer, assurez-vous que vous avez les éléments suivants :

- Windows 10 version 2004 ou supérieure, ou Windows 11, avec WSL activé.
- Ubuntu installé via WSL.

Étapes de l'installation

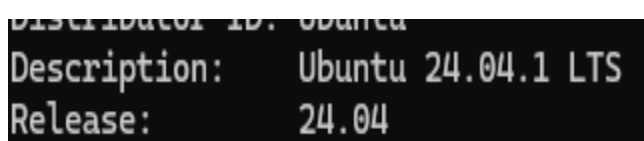
1. Ouvrir votre distribution Ubuntu dans WSL

- Ouvrez Ubuntu depuis le menu Démarrer ou via la commande `wsl` dans PowerShell.
- Vous accédez à un terminal avec une invite en ligne de commande sous Ubuntu.

2. Mettre à jour les packages Ubuntu

Avant d'installer SSH, il est important de mettre à jour votre système :

- `bash`
- `sudo apt update`
- `sudo apt upgrade`



```
Distribution ID: ubuntu
Description:    Ubuntu 24.04.1 LTS
Release:        24.04
```

Cela garantit que vous disposez des dernières mises à jour pour tous vos paquets Ubuntu.

3. Installer OpenSSH Server

Pour installer le serveur SSH, exécutez la commande suivante dans le terminal :

bash

- `sudo apt install openssh-server`

b

```
adem@almighty:~$ sudo apt install openssh-server
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  libwrap0 ncurses-term openssh-sftp-server ssh-import-id
Suggested packages:
  molly-guard monkeysphere ssh-askpass ufw
The following NEW packages will be installed:
  libwrap0 ncurses-term openssh-server openssh-sftp-server ssh-import-id
0 upgraded, 5 newly installed, 0 to remove and 0 not upgraded.
```

Cette commande va installer le serveur SSH (OpenSSH) qui permet d'accepter les connexions à distance via SSH.

4. Vérifier que SSH fonctionne

Après l'installation, il est important de vérifier que le service SSH fonctionne correctement.

- Vérifiez l'état du service SSH :

bash

- `sudo service ssh status`

```
adem@almighty:~$ sudo service ssh status
○ ssh.service - OpenBSD Secure Shell server
   Loaded: loaded (/usr/lib/systemd/system/ssh.service; disabled; preset: enabled)
   Active: inactive (dead)
 TriggeredBy: ● ssh.socket
    Docs: man:sshd(8)
          man:sshd_config(5)
```

- Si le service n'est pas démarré, lancez-le avec cette commande :

bash

- `sudo service ssh start`

5. Vérifier que SSH écoute sur le bon port

Par défaut, SSH écoute sur le port 22. Vérifiez que ce port est en écoute avec la commande suivante :

bash

- `sudo netstat -tuln | grep :22`

```
adem@almighty:~$ sudo netstat -tuln | grep :22
tcp6      0      0 :::22          :::*           LISTEN
```

si la commande ne fonctionne pas faite la commande suivante :

```
adem@almighty:~$ sudo apt install net-tools
```

Si vous voyez une ligne indiquant 0.0.0.0:22, cela signifie que SSH fonctionne correctement et écoute les connexions entrantes sur le port 22.

6. Configurer SSH (facultatif)

Si vous souhaitez personnaliser le comportement de SSH, vous pouvez modifier son fichier de configuration. Par exemple, vous pourriez vouloir changer le port d'écoute ou activer des options de sécurité supplémentaires.

- Ouvrez le fichier de configuration SSH :

bash

Copier le code

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

- Apportez les modifications nécessaires. Pour changer le port, trouvez la ligne Port 22 et remplacez 22 par un autre numéro de port si vous le souhaitez.
- Enregistrez et fermez le fichier (Ctrl + X, puis Y pour confirmer).
- Redémarrez le service SSH pour appliquer les modifications :

bash

- `sudo service ssh restart`

7. Accéder à Ubuntu via SSH depuis Windows

1. Récupérez l'adresse IP de votre système Ubuntu avec la commande suivante :

Bash

- `hostname -I`

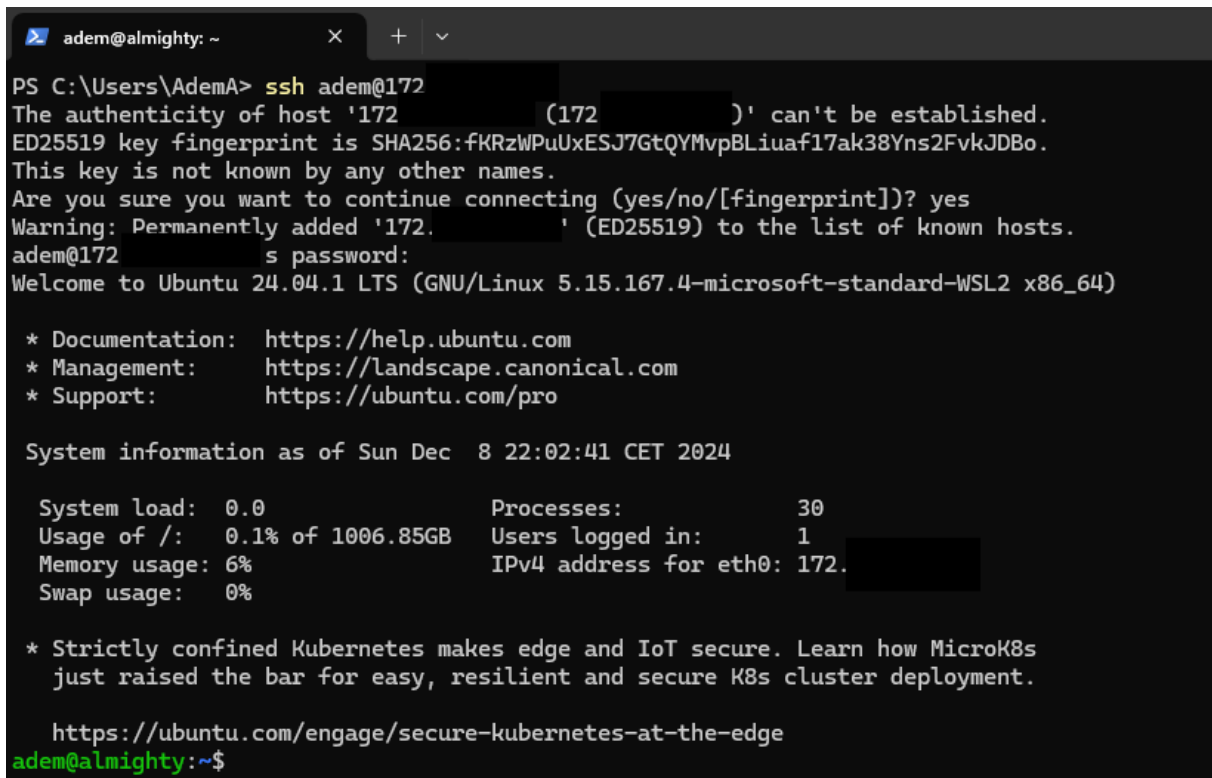
2. Copiez l'adresse IP affichée et ouvrez PowerShell sur votre machine Windows.

3. Exécutez la commande SSH pour vous connecter à votre machine Ubuntu :

bash

`ssh <votre_utilisateur>@<adresse_ip>`

Remplacez `<votre_utilisateur>` par votre nom d'utilisateur Ubuntu et `<adresse_ip>` par l'adresse IP obtenue précédemment.



```
PS C:\Users\AdemA> ssh adem@172.17.0.1
The authenticity of host '172.17.0.1 (172.17.0.1)' can't be established.
ED25519 key fingerprint is SHA256:FKRzWPuUxESJ7GtQYMvpBLiuaf17ak38Yns2FvkJDBo.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '172.17.0.1' (ED25519) to the list of known hosts.
adem@172.17.0.1's password:
Welcome to Ubuntu 24.04.1 LTS (GNU/Linux 5.15.167.4-microsoft-standard-WSL2 x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/pro

System information as of Sun Dec  8 22:02:41 CET 2024

System load:  0.0               Processes:            30
Usage of /:   0.1% of 1006.85GB  Users logged in:     1
Memory usage: 6%               IPv4 address for eth0: 172.17.0.1
Swap usage:   0%

 * Strictly confined Kubernetes makes edge and IoT secure. Learn how MicroK8s
   just raised the bar for easy, resilient and secure K8s cluster deployment.

https://ubuntu.com/engage/secure-kubernetes-at-the-edge
adem@almighty:~$
```

4. Vous serez invité à entrer votre mot de passe Ubuntu. Si tout est configuré correctement, vous serez connecté à votre système Ubuntu.
-

8. Activer le démarrage automatique de SSH (facultatif)

Si vous souhaitez que SSH démarre automatiquement à chaque démarrage de WSL, vous pouvez ajouter la commande de démarrage dans votre fichier `.bashrc`.

1. Ouvrez le fichier `.bashrc` :

```
bash
```

```
nano ~/.bashrc
```

2. Ajoutez la ligne suivante à la fin du fichier pour démarrer SSH automatiquement :

```
bash
```

```
sudo service ssh start
```

3. Enregistrez et fermez le fichier (Ctrl + X, puis Y et Entrée).