

# Projet :

## Connexion à distance via SSH



### Table des matières

<b>I. Définition</b> .....	2
<b>II. Objectif</b> .....	2
<b>III. Outils</b> .....	2
<b>IV. Pratique</b> .....	3
Phase une : Préparation .....	3
Phase deux : connexion .....	4

# **I. Définition**

**SSH (Secure Shell)** est un protocole de réseau cryptographique utilisé pour sécuriser les communications entre deux machines, généralement sur un réseau non sécurisé comme l'Internet. SSH permet aux utilisateurs d'accéder à une machine distante et d'exécuter des commandes comme s'ils étaient en local

Protocoles intégrés comme SCP (Secure Copy) et SFTP (SSH File Transfer Protocol) pour copier des fichiers entre machines.

**Flexibilité** : Support de divers mécanismes d'authentification, y compris les mots de passe, les clés publiques, et les certificats.

**Sécurité** : Chiffrement fort pour protéger contre les attaques de type écoute clandestine (eavesdropping) et détournement de connexion.

# **II. Objectif**

- Repérer les appareils dans le réseau local.
- Ouvrir les ports.
- Établir une connexion SSH entre deux machines, que ce soit en local ou à distance.
- Avoir une interface graphique si besoin.

# **III. Outils**

Pour réaliser cela j'ai besoin de trois machines sous différentes OS



Windows 11 (Professionnel) ;



Linux

(Zorin) ;



IOS

(17.5.1)

Utilitaire

Windows : [Terminal PowerShell](#), [PuTTY](#), [OpenSSH](#), [Advanced IP Scanner](#) (et 2XGO)

Linux : [Terminal](#), [OpenSSH](#)

IOS : [WebSSH](#)

## IV. Pratique

### Phase une : Préparation

#### - OpenSSH :

Vérifier que Windows est bien équipé de **OpenSSH** et que ce dernier est bien activé. Pour ma part **OpenSSH** n'était point présent sur mon pc, donc j'ai dû l'installer, l'exécuter et vérifier son statut sur le terminal (exécuté en administrateur) avec la commande : *Get-Service -Name sshd*

**Pour l'exécuter** : *Start-Service -Name sshd*

```
PS C:\Users\dede_> Start-service -name sshd
PS C:\Users\dede_> Get-service -name sshd
```

Status	Name	DisplayName
-----	----	-----
Running	sshd	OpenSSH SSH Server

Si running est indiqué, **OpenSSH** est bien en activé et fonctionnel

**Pour arrêter le service** : *Stop-Service -Name sshd*

### Connexion Local

#### - **Advanced IP Scanner**

va m'aider à obtenir les IP de toutes les machine sur le réseau, pour bien repéré les trois machines que je vais utiliser pour le SSH en local.

Pour la partie connexion depuis un autre réseau le logiciel n'aura point d'utilité.

### Connexion depuis un autre réseau

Premièrement, trouver son **IP publique**.

Voici un moyen d'obtenir son IP : <https://www.whatismyip.com/fr/>

Deuxièmement, ouvrir les ports du routeur (distributeur réseau ; Free, orange ect) de la machine qui va être contrôlé et indiquer l'IP de la machine.

Des portails mis en place par les opérateurs sont disponible et accessible, simplement avec des identifiants & mot de passe transmit en amont.

Se connecter, Ouvrir les Ports (22 pour cette situation), mettre l'IP de la machine et enregistrer.

## Phase deux : connexion

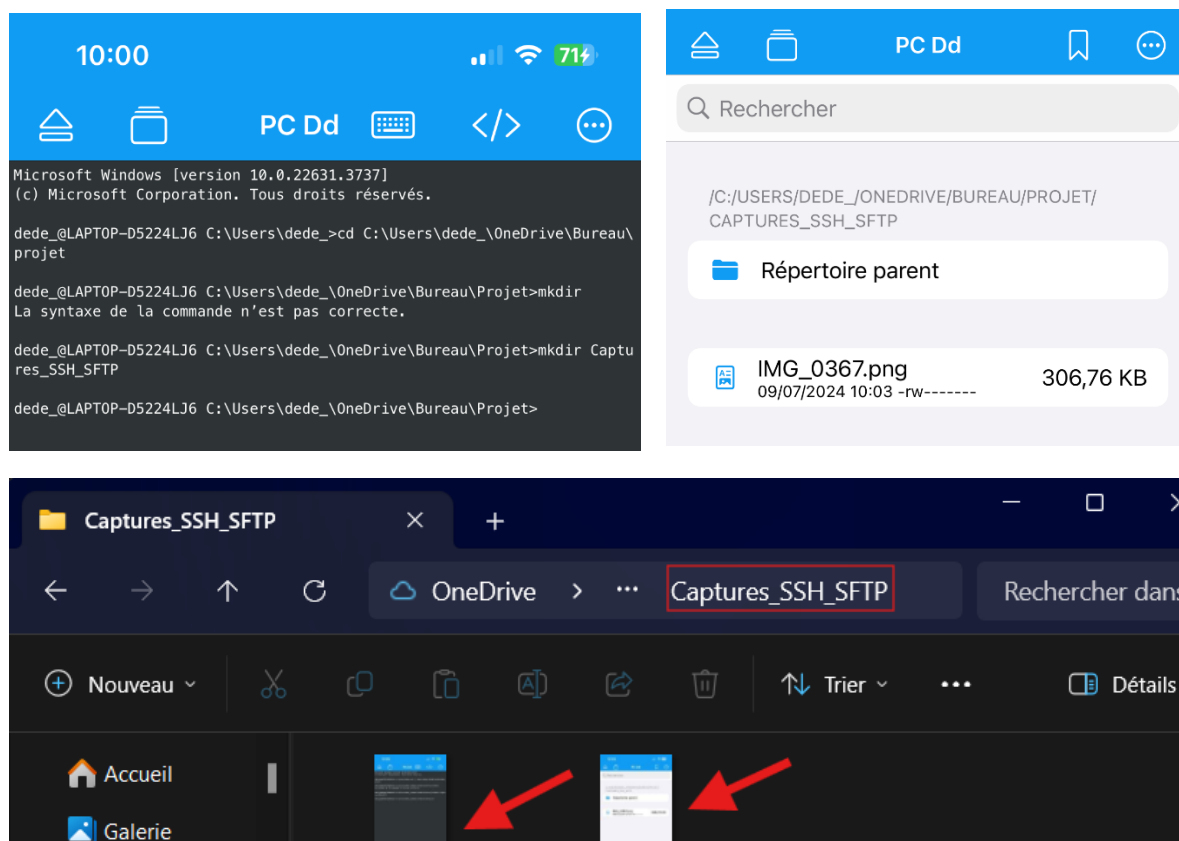
IOS 🍏 à Windows 🪄

Depuis mon iPhone, j'installe une application qui permettra d'effectuer une connexion SSH & SFTP. Dans mon cas j'utilise l'application **WebSSH**.

J'entre les informations tel que **L'adresse IP de l'hôte\***, **Le Port**, Mon compte utilisateur et Mot de passe de ma session Windows.

Une fois la connexion établie, l'application nous demande de choisir entre SSH ou SFTP.

Deux capture ci-dessus pour montrer la partie SSH ou j'ai créé un dossier, ensuite le dossier dans la partie SFTP de l'application et mon import de la première capture dans le dossier.

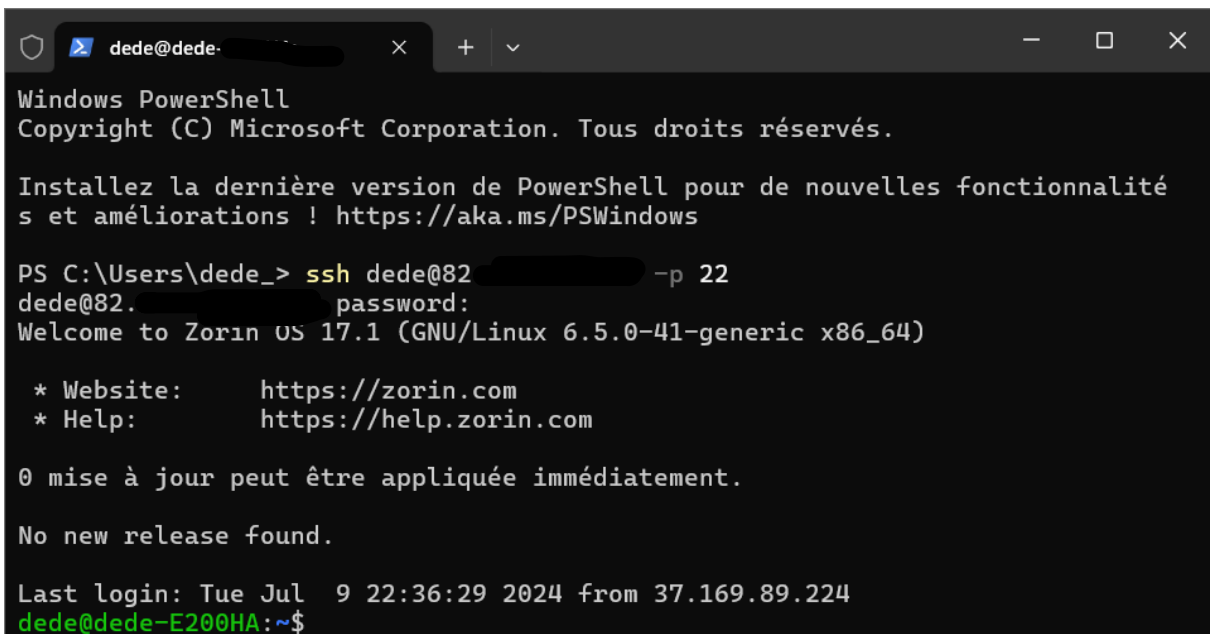


*\*l'IP de la machine si je suis sur le même réseau, sinon l'IP Publique*

## Windows à Linux (Non Local)

Depuis mon Windows, je vais me connecter Via SSH avec PowerShell et un logiciel nommé PuTTY, qui a parmi ses fonctionnalités, la connexion SSH. Lancé PowerShell en tant qu'administrateur et taper les commandes ci-dessous :

`ssh [Nom d'utilisateur de la session du pc distant] @ [Nom de domaine ou IP publique] -p 22 (par défaut)`



```
Windows PowerShell
Copyright (C) Microsoft Corporation. Tous droits réservés.

Installez la dernière version de PowerShell pour de nouvelles fonctionnalités et améliorations ! https://aka.ms/PSWindows

PS C:\Users\dede> ssh dede@82 [redacted] -p 22
dede@82. [redacted] password:
Welcome to Zorin OS 17.1 (GNU/Linux 6.5.0-41-generic x86_64)

* Website:      https://zorin.com
* Help:         https://help.zorin.com

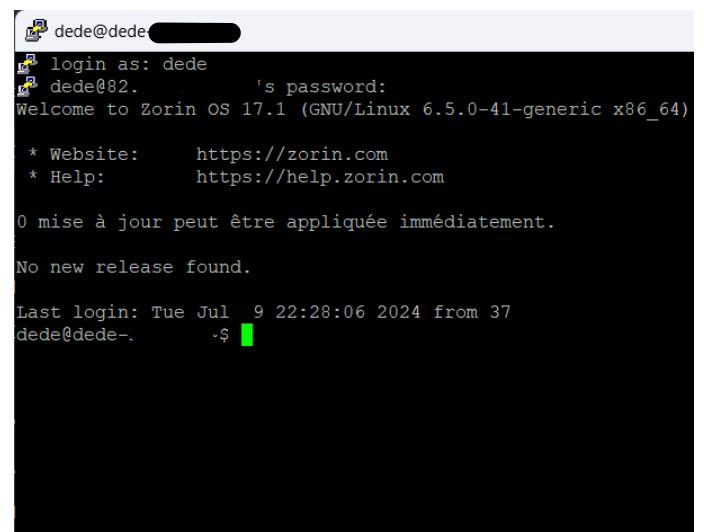
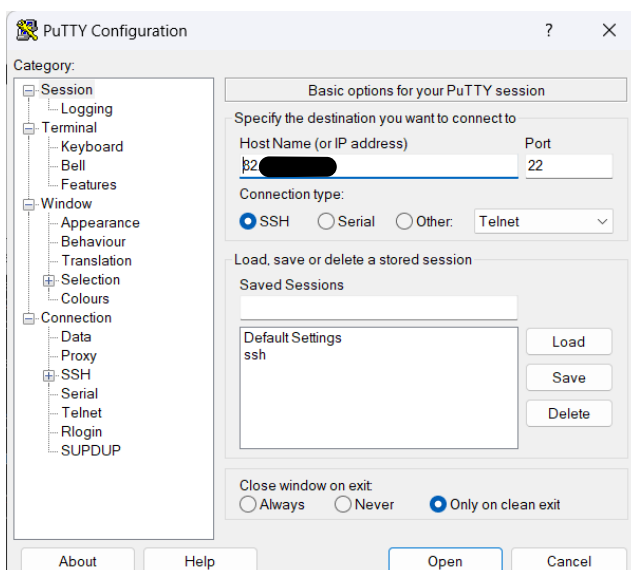
0 mise à jour peut être appliquée immédiatement.

No new release found.

Last login: Tue Jul  9 22:36:29 2024 from 37.169.89.224
dede@dede-E200HA:~$
```

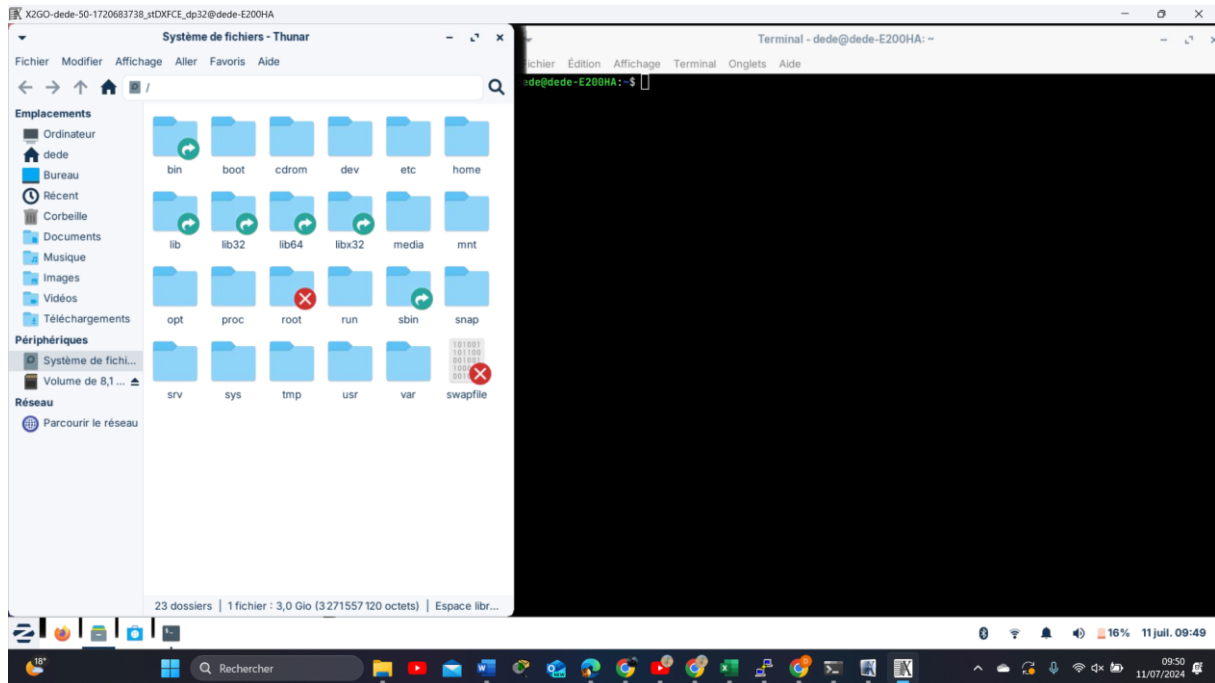
Après avoir entré le mot de passe me voilà connecter.

Sur PuTTY la démarche est similaire j'insère quelques captures ci-dessous :



Pour les personnes non familiarisées avec les lignes de commande le logiciel 2XGO est une connexion SSH, qui va nous donner accès à la machine, mais la particularité est au niveau graphique.

Une fois la connexion établie une fenêtre de l'interface de la machine s'affiche.



### L'installation sur linux :

*sudo apt-get install x2goserver x2goserver-xsession*

*pour la partie graphique :*

*sudo apt-get install xfce4 xfce4-goodies*

### Sur Windows :

Installer tout simplement le client


<https://xming.fr.softonic.com/telecharger>

*Sur le client entrer les informations demandées sur le serveur et pour conclure sélectionner « XFCE »*

Paramétrage de la session - Nouvelle session

Session Connexion Entrées/Sorties Media Dossiers partagés

Nom de la session : Nouvelle session

 << changer d'icône

Chemin : / ...

Serveur

Hôte : 82 ...

Identifiant : dede

Port SSH : 22

Utiliser une clé RSA/DSA pour la connexion SSH : ...

☐ Essayer l'authentification automatique (via l'agent SSH ou la clé SSH par défaut)

☐ Authentification Kerberos 5 (GSSAPI)

☐ Délégation de l'identification GSSAPI au serveur

☐ Utiliser un serveur mandataire pour la connexion SSH

Type de session

☐ Démarrer dans X2GoKDrive (expérimental)

XFCESession KDE GNOME LXDE LXQT XfceMATE UNITY CINNAMON TRINITY OPENBOX

Commande :

Ok Cancel Défauts