

PROCEDURE OPEN-SSH

Installation et Configuration d'un serveur SSH : serveur et client, d'une paire clé publique et privée.

FADIGA Bafode

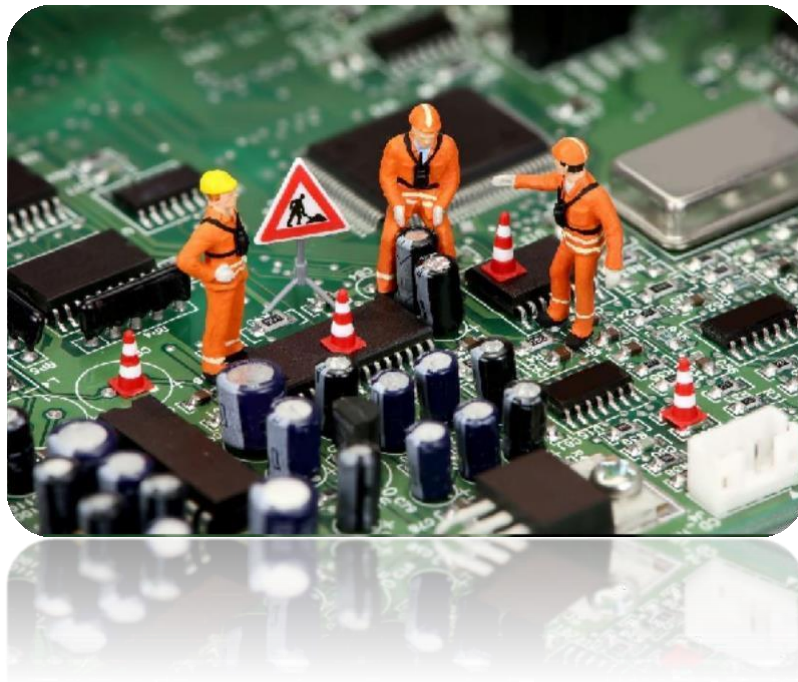
Mise en place d'un
serveur SSH linux

BTS Services Informatiques aux Organisations Option
Solutions d'Infrastructure, Systèmes et Réseaux

Epreuve E6 : Conception et Maintenance de solutions Informatiques.

Documentation technique

Projet 2 : Mise en place d'un service SSH pour une connexion client.



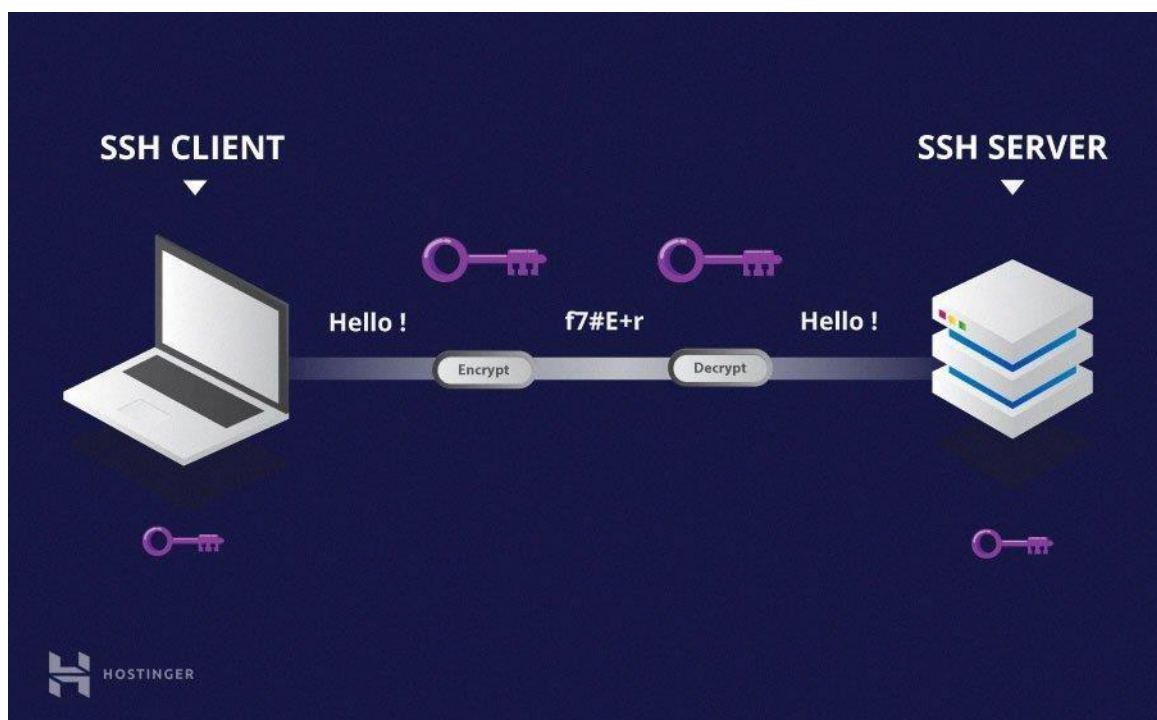
Sommaire

- Introductionp.4
- Installation du serveur Debian 12 sous linux.....p.5
- Configuration du serveur Debian 12 sous linux.....p.7
- Définir le nom et le mot de passe de la paire de clés privée et publique ainsi que le Passphrase.....p.9
- Configuration du fichier SSHD_CONFIG.....p.9
- Création de la banierre.....p.11
- Connexion avec une solution mobile.....p.12
- Double authentification google.....p.14
- Activer le client SSH intégré à Windows.....p.17
- Conclusion.....p.18

Introduction

OpenSSH

OpenSSH est un ensemble d'outils informatiques libres permettant des communications sécurisées sur un réseau informatique en utilisant le protocole SSH.



LVM Crypté

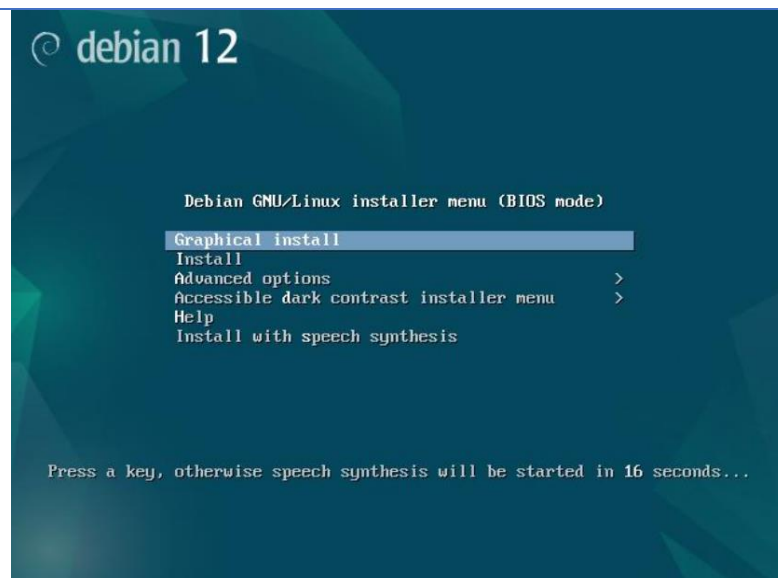
Lorsqu'une partition LVM cryptée est utilisée, la clé de cryptage est stockée dans la mémoire (RAM). ... Si cette partition n'est pas cryptée, le voleur peut accéder à la clé et l'utiliser pour décrypter les données des partitions cryptées. C'est pourquoi, lorsque vous utilisez des partitions chiffrées LVM, il est recommandé de chiffrer également la partition d'échange.

Diminuer les surfaces d'attaques sur nos serveurs et clients SSH :

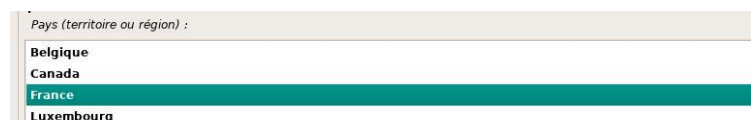
Nous allons sécuriser le disque dur et les menus de démarrage sur les machines virtuels : Mise en place de l'infrastructure

- Installer les machines virtuelles : 2 Serveurs sous Linux Debian 11, Windows 10, et Smartphone,
- Installation LVM chiffré (Disque dur).

Installation du serveur Debian 12 sous Linux



Choisir le pays d'emplacement « France »



Choisir la langue du paquet installer en français



Configurer la langue du clavier en français



Laisser les options s'installer

Définir le mot de passe administrateur (root)

Par sécurité, rien n'est affiché pendant la saisie.
Mot de passe du superutilisateur (« root ») :

●●●●●●●●

Veuillez entrer à nouveau le mot de passe du superutilisateur afin de vérifier qu'il a été saisi correctement.
Confirmation du mot de passe :

●●●●●●●●

Crée un utilisateur

Entrez le nom de l'utilisateur
FADIGA Définir puis confirmer le mot
de passe

Lancement du partitionnement de disque : utiliser le disque en LVM Chiffré

debian 12

Partitionner les disques

Le programme d'installation peut vous assister pour le partitionnement d'un disque (avec plusieurs choix d'organisation). Vous pouvez également effectuer ce partitionnement vous-même. Si vous choisissez le partitionnement assisté, vous aurez la possibilité de vérifier et personnaliser les choix effectués.

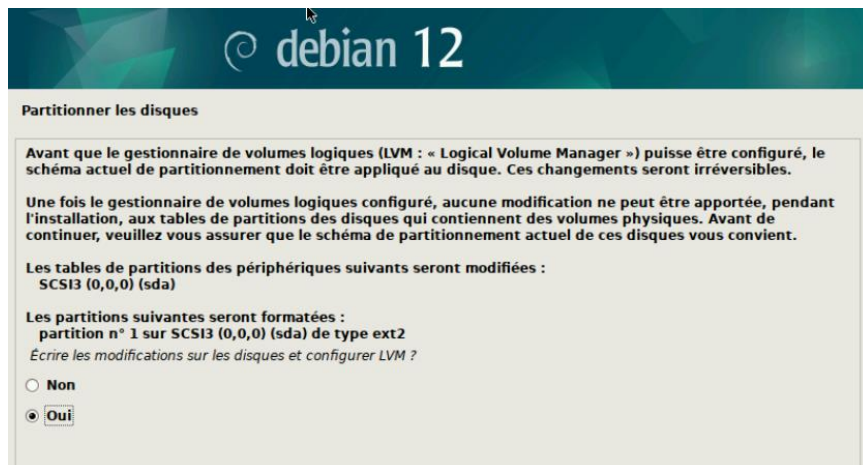
Si vous choisissez le partitionnement assisté pour un disque complet, vous devrez ensuite choisir le disque à partitionner.

Méthode de partitionnement :

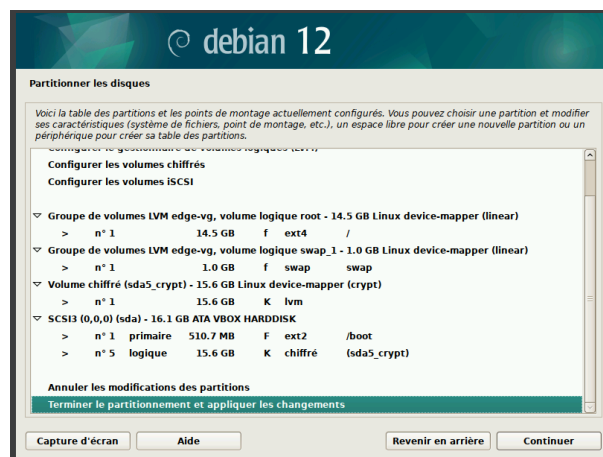
- Assisté - utiliser un disque entier
- Assisté - utiliser tout un disque avec LVM
- Assisté - utiliser tout un disque avec LVM chiffré**
- Manuel

Capture d'écran Revenir en arrière Continuer

Appliquer les modifications au disque

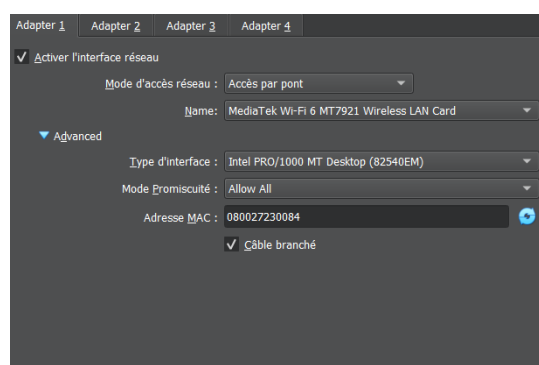


Les partitions primaires et logiques ont été créé sur le disque



Configuration du serveur Debian 12 sous linux

Configuration de la carte réseau du serveur ssh 1 de même pour Windows 10



Définir le nom du serveur

```

Debian GNU/Linux 12 edge tty1
edge login: root
Password:
Linux edge 6.1.0-35-amd64 #1 SMP PREEMPT_DYNAMIC Debian 6.1.137-1 (2025-05-07) x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
root@edge:~# hostnamectl set-hostname sshserver1
root@edge:~# _

```

Mise à jour du serveur

```

root@edge:~# apt update && apt upgrade
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances... Fait
Lecture des informations d'état... Fait
Tous les paquets sont à jour.
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances... Fait
Lecture des informations d'état... Fait
Calcul de la mise à jour... Fait
0 mis à jour, 0 nouvellement installés, 0 à enlever et 0 non mis à jour.
root@edge:~#

```

Installer le paquet openssh-server sur le serveur ssh 1

```
root@sshserver1:~# apt install openssh-server
```

Crée l'utilisateur FADIGA puis définir son mot de passe

```
root@sshserver1:~# adduser fadiga
```

Création d'une identité numérique de l'utilisateur fadiga

Se rendre dans le dossier .ssh de l'utilisateur fadiga

```

root@sshserver1:~# cd /home/fadiga/
root@sshserver1:/home/fadiga# mkdir .ssh

```

Installer le paquet openssh-server sur le serveur ssh 1

```
root@sshserver1:~# apt install openssh-server
```


Définir le nom et le mot de passe de la paire de clés privée et publique ainsi que le Passphrase

```
root@sshserver1:/home/fadiga# cd /home/fadiga/.ssh/
root@sshserver1:/home/fadiga/.ssh# ssh-keygen -t rsa -b 1024
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa): fadiga_rsa
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in fadiga_rsa
Your public key has been saved in fadiga_rsa.pub
The key fingerprint is:
SHA256:/YYb9NUh24CDWMoyWMNhh7MBXgJiRqWILz7GuB7kDic root@sshserver1
The key's randomart image is:
+---[RSA 1024]-----+
|. *0+0*0. . |
|= + *+=+ + . . |
|0. 0 0++ . 0 0 . |
|. . .0 . . =.. |
|... S 0 .... |
|* . + . |
|EB. 0 + |
|++0 + |
|00 . |
+----[SHA256]-----+
root@sshserver1:/home/fadiga/.ssh# _
```

Configuration du fichier SSHD_CONFIG

Afin d'éviter les attaques de force brute on change le port ssh par défaut

```
root@sshserver1:~# nano /etc/ssh/sshd_config|
```

Sur sshserver 1 on change le port 22 en 33

```

GNU nano 7.2 /etc/s
# This is the sshd server system-wide configuration file.  See
# sshd_config(5) for more information.

# This sshd was compiled with PATH=/usr/local/bin:/usr/bin:/bin:/usr/games

# The strategy used for options in the default sshd_config shipped with
# OpenSSH is to specify options with their default value where
# possible, but leave them commented.  Uncommented options override the
# default value.

Include /etc/ssh/sshd_config.d/*.conf

Port 33
#AddressFamily any
#ListenAddress 0.0.0.0
#ListenAddress ::

#HostKey /etc/ssh/ssh_host_rsa_key
#HostKey /etc/ssh/ssh_host_ecdsa_key
#HostKey /etc/ssh/ssh_host_ed25519_key

# Ciphers and keying
#RekeyLimit default none

# Logging
#SyslogFacility AUTH
#LogLevel INFO

# Authentication:
allowusers fadiga
LoginGraceTime 1m
PermitRootLogin yes

```

Ctrl + x : enregistrer le fichier sshd_config > oui Redémarrer le service sshd : service sshd restart

```

root@sshserver1:~# service sshd restart

```

```

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Sat May 31 13:38:42 CEST 2025 from 10.0.0.2 on pts/0
root@sshserver1:~# ssh root@10.0.0.3 -p 33

```

Activation de la clé de l'agent ssh : Se connecter avec l'utilisateur root

```
root@sshserver1:/home/fadiga/.ssh# exec ssh-agent bash
root@sshserver1:/home/fadiga/.ssh# ls
fadiga_rsa  fadiga_rsa.pub
root@sshserver1:/home/fadiga/.ssh# ssh-add fadiga_rsa
Enter passphrase for fadiga_rsa:
Identity added: fadiga_rsa (root@sshserver1)
```

Installer le programme figlet pour crée une bannière de connexion SSH

```
root@sshserver1:~# apt install figlet
```

Le fichier de la bannière a bien été créé dans le dossier .ssh de l'utilisateur fadiga

```
root@sshserver1:~# figlet Bienvenu, Connexion SSH Fadiga > /home/fadiga/.ssh/banner
```

nano /etc/ssh/sshd_config : Il faut aller chercher le fichier de configuration ssh
déclarer le chemin de la bannière enregistrer les modifications (ctrl+x) > oui

```
# no default banner path
Banner /home/fadiga/.ssh/banner
```

Après redémarrer le service ssh : **service ssh restart**

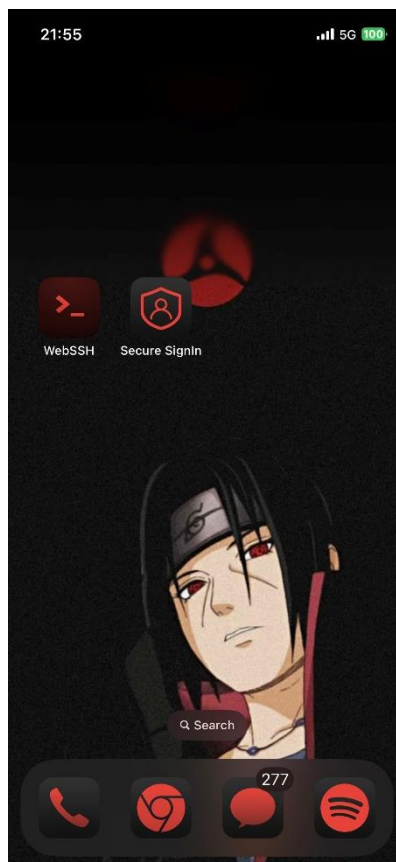
Vérification du contenu de la bannière

[illegible]

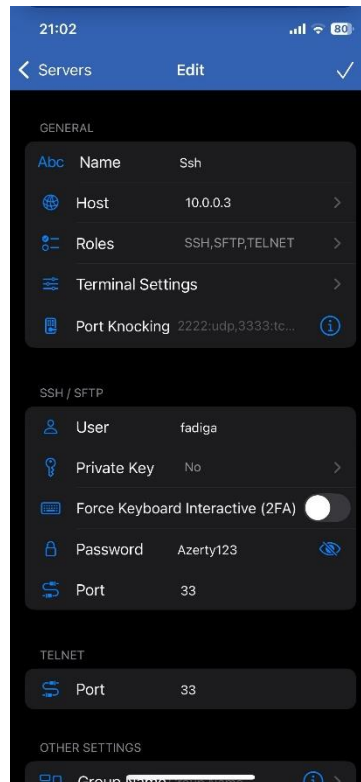
Connexion avec une solution mobile

Sur le serveur ssh1 avec l'utilisateur fadiga

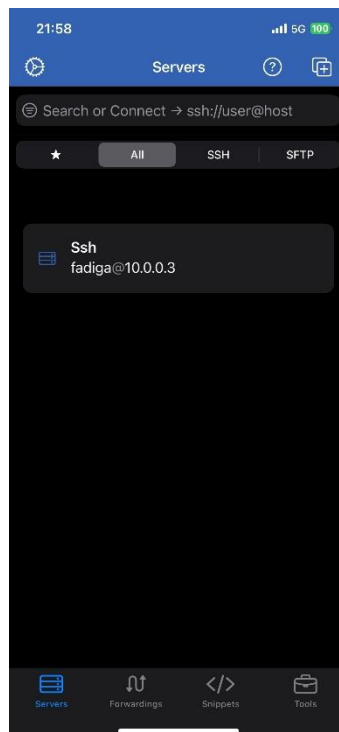
Vérification de l'emprunt numérique > enter le mot de passe > connexion réussie



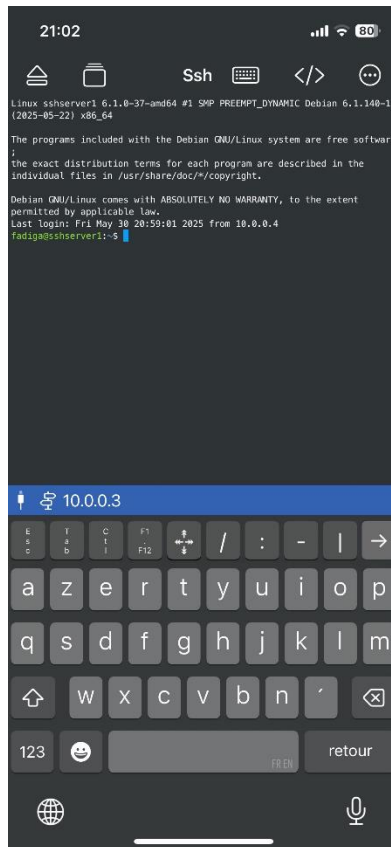
Télécharger l'app WebSSH et Secure Signin



Entrez l'adresse IP et le port du serveur
Avec le nom et le mot utilisateur.



Le serveur SSH devrait apparaitre.



Voilà ! vous êtes connectée sur votre serveur SSH.

Double authentification google

Installation du google authenticateur :

`apt install libpam-google-authenticator`

Se rendre pour modifier le fichier sshd de la double authentification

`nano /etc/pam.d/sshd`

Ajouter 2 lignes à la fin du fichier

```
# authentification google
auth required pam_google_authenticator.so
```

Ctrl X + yes

Rédemarre le service : `service sshd restart` ou `systemctl restart sshd`

Généré le QRCODE de la double authentification google

```
root@sshserver1:~# google-authenticator
```

```
root@10.0.0.3's password:
root@10.0.0.3: Permission denied (password).
root@sshserver1:~# google-authenticator
```

```
Do you want authentication tokens to be time-based (y/n) y
```

```
Warning: pasting the following URL into your browser exposes the OTP secret to Google:
```

```
https://www.google.com/chart?chs=200x200&chld=M|0&cht=qr&chl=otpauth://totp/root@sshserver1%3Fsecret%3DZ3SREXXWIXI27J4R2RJNILQIRI
```



```
Your new secret key is: Z3SREXXWIXI27J4R2RJNILQIRI
```

```
Enter code from app (-1 to skip): 793032
```

```
Code confirmed
```

```
Your emergency scratch codes are:
```

```
26656434
```

```
78515984
```

```
48692501
```

```
67091392
```

```
20817484
```

```
Do you want me to update your "/root/.google_authenticator" file? (y/n) y
```

```
Do you want to disallow multiple uses of the same authentication token? This restricts you to one login about every 30s, but it increases your chances to notice or even prevent man-in-the-middle attacks (y/n) y
```

```
By default, a new token is generated every 30 seconds by the mobile app. In order to compensate for possible time-skew between the client and the server, we allow an extra token before and after the current time. This allows for a time skew of up to 30 seconds between authentication server and client. If you experience problems with poor time synchronization, you can increase the window from its default size of 3 permitted codes (one previous code, the current code, the next code) to 17 permitted codes (the 8 previous codes, the current code, and the 8 next codes). This will permit for a time skew of up to 4 minutes between client and server.
```

```
Do you want to do so? (y/n) y
```

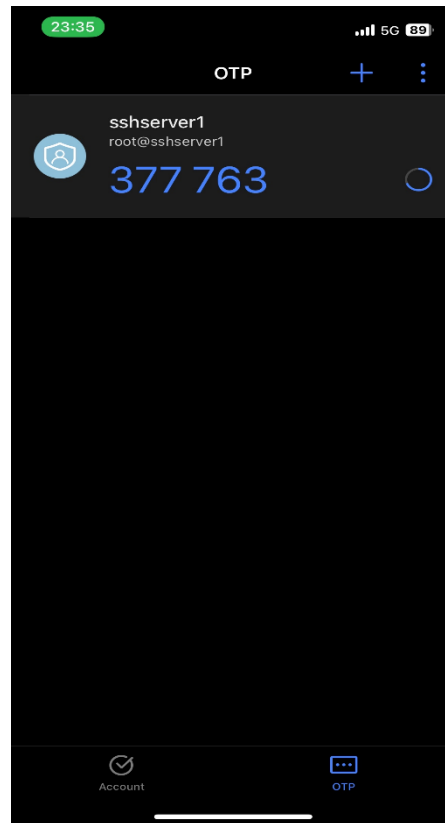
```
If the computer that you are logging into isn't hardened against brute-force login attempts, you can enable rate-limiting for the authentication module. By default, this limits attackers to no more than 3 login attempts every 30s.
```

```
Do you want to enable rate-limiting? (y/n) y
```

```
root@sshserver1:~#
```

Questions de renforcement de la sécurité authentifiée (répondre y)

Test de la double authentification en root avec secure signin



Et puis se connecter sur le serveur SSH en root, entrez le mot de passe root. Il va vous demander de vérification code de double authentification sur ton téléphone vous prenez le code à 6 chiffres.

[illegible]

Activer le client SSH intégré à Windows

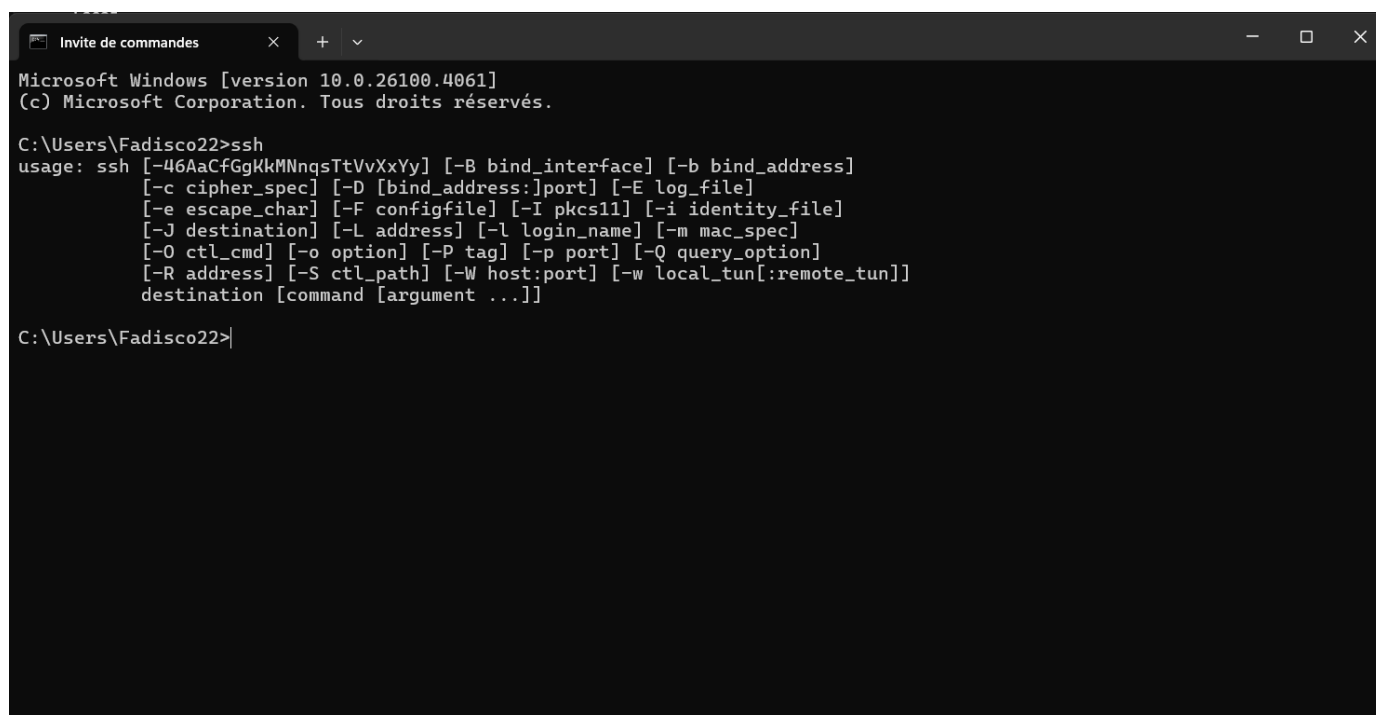
Depuis la mise à jour Fall Creators Update, Windows intègre un client OpenSSH vous permettant de vous connecter à un serveur Secure Shell. Plus besoin donc de passer par un utilitaire tiers comme Putty.

1. Le client SSH est disponible en tant qu'option et n'est pas installé par défaut. Pour l'installer, cliquez sur le bouton **Démarrer** puis sur **Paramètres>applications>fonctionnalités facultatives> Ajouter une fonctionnalité>OpenSSH Client**

 Client OpenSSH

Redémarrez votre ordinateur.

Vous pouvez désormais utiliser le client SSH en utilisant la commande **ssh** dans une fenêtre PowerShell ou d'Invite de commandes. Saisissez la commande et validez par **Entrée** pour connaître la syntaxe de la commande.



```
Microsoft Windows [version 10.0.26100.4061]
(c) Microsoft Corporation. Tous droits réservés.

C:\Users\Fadisco22>ssh
usage: ssh [-46AaCfGgKkMMNnqsTtVvXxYy] [-B bind_interface] [-b bind_address]
          [-c cipher_spec] [-D [bind_address:]port] [-E log_file]
          [-e escape_char] [-F configfile] [-I pkcs11] [-i identity_file]
          [-J destination] [-L address] [-l login_name] [-m mac_spec]
          [-O ctl_cmd] [-o option] [-P tag] [-p port] [-Q query_option]
          [-R address] [-S ctl_path] [-W host:port] [-w local_tun[:remote_tun]]
          destination [command [argument ...]]

C:\Users\Fadisco22>
```

MISSION ACOMPLIE !!!

Conclusion La mise en place d'un service SSH sécurisé avec authentification par clé et double authentification renforce considérablement la sécurité des accès distants aux serveurs. Grâce à l'utilisation de l'agent SSH, les connexions sont facilitées tout en maintenant un haut niveau de protection. Cette solution assure à la fois confidentialité, intégrité des échanges et simplification de l'administration système à distance.

