Rousse Danais Kenan

Procédure OPEN-SSH

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

Epreuve E5 : Administration des systèmes et des réseaux

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

Sommaire

Configuration du serveur Debian 11 sous linux

Définir le nom du serveur

```
root@debian:~# hostnamectl set-hostname sshserver1
```

Mise à jour du serveur

```
root@debian:~# apt update && apt upgrade
Ign :1 cdrom://[Debian GNU/Linux 11.2.0 _Bullseye_ - Official
Err :2 cdrom://[Debian GNU/Linux 11.2.0 _Bullseye_ - Official
  Veuillez utiliser apt-cdrom afin de faire reconnaître ce céd
déroms
Réception de :3 http://deb.debian.org/debian bullseye InReleas
Réception de :4 http://security.debian.org/debian-security bul
Réception de :5 http://deb.debian.org/debian bullseye-updates
```

Installer le paquet openssh-server sur le serveur ssh 1

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

Crée l'utilisateur kaiser puis définir son mot de passe

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

Création d'une identité numérique de l'utilisateur kaiser

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

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

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

```
root@sshserver1:/home/kaiser/.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): kaiser_rsa
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in kaiser_rsa
Your public key has been saved in kaiser_rsa.pub
The key fingerprint is:
SHA256:j8ZTh3nVXGeWqd84H4q7AugRGStmY2iZrUDMN2otFKI root@sshserver1
The key's randomart image is:
+---[RSA 1024]----+
|+..              B|
|o= o .          Bo|
|E O . +        o o|
|.O O +     o o   |
|+ * o o S + o ...|
| .   o o + o  oo.|
|    . . * . . .o.|
|     . . o . .  .|
|         .oo     |
+----[SHA256]-----+
```

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

```
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 kaiser
LoginGraceTime 1m
PermitRootLogin yes
#StrictModes yes
```

Ctrl + x : enregistrer le fichier sshd_config > oui

Redémarrer le service SSHD : service SSHD restart

```
root@sshserver1:~# exit
déconnexion
Connection to 192.168.64.65 closed.
PS C:\Users\Tooba> ssh kaiser@192.168.64.65
ssh: connect to host 192.168.64.65 port 22: Connection refused
PS C:\Users\Tooba> ssh kaiser@192.168.64.65 -p 33
kaiser@192.168.64.65's password:
Linux sshserver1 5.10.0-13-amd64 #1 SMP Debian 5.10.106-1 (2022-03-17) 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.
Last login: Thu Apr 21 06:19:27 2022 from 192.168.64.108
kaiser@sshserver1:~$
```

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

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

Création de la bannière

Installer le programme figlet pour crée une bannière de connexion SSH

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

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

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

Après redémarrer le service SSH : **service SSH restart**

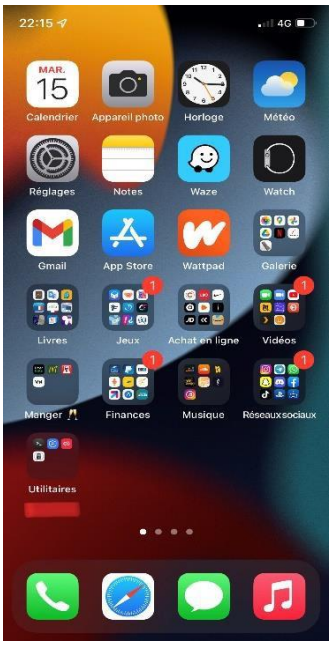
Vérification du contenu de la bannière

```
kaiser@sshserver1:~$ su
Mot de passe :
root@sshserver1:/home/kaiser# cd .ssh
root@sshserver1:/home/kaiser/.ssh# cat banner
 ____  _
| __ )(_) ___ _ ____   _____ _ __  _   _
|  _ \| |/ _ \ '_ \ \ / / _ \ '_ \| | | |
| |_) | |  __/ | | \ V /  __/ | | | |_| |_
|____/|_|\___|_| |_|\_/ \___|_| |_|\__,_( )
                                        |/
  ____                                  _                 ____ ____   _    _
 / ___|___  _ __  _ __   _____  _(_) ___  _ __   / ___/ ___|| | | |
| |   / _ \| '_ \| '_ \ / _ \ \/ / |/ _ \| '_ \  \___ \___ \| |_| |
| |__| (_) | | | | | | |  __/>  <| | (_) | | | |  ___) |__) |  _  |
 \____\___/|_| |_|_| |_|\___/_/\_\_|\___/|_| |_| |____/____/|_| |_|

 _  __      _
| |/ /__ _(_)___  ___ _ __
| ' // _` | / __|/ _ \ '__|
| . \ (_| | \__ \  __/ |
|_|\_\__,_|_|___/\___|_|
```

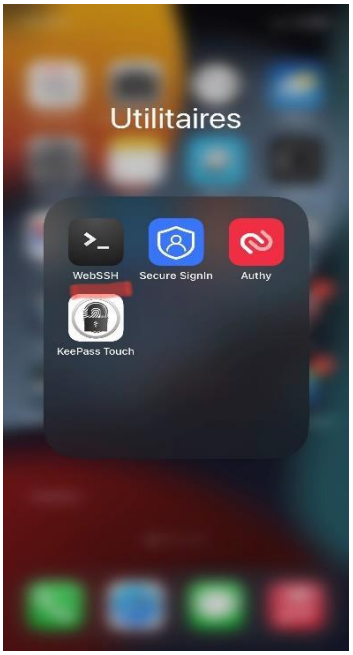
Connexion avec une solution mobile

Sur le serveurssh1 avec l'utilisateur kaiser

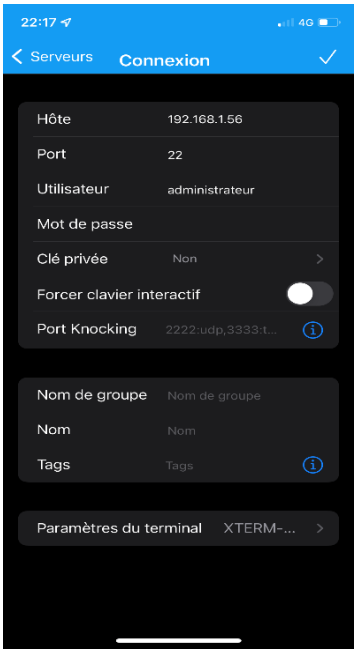
Vérification de l'emprunt numérique > enter le mot de passe > connexion réussie



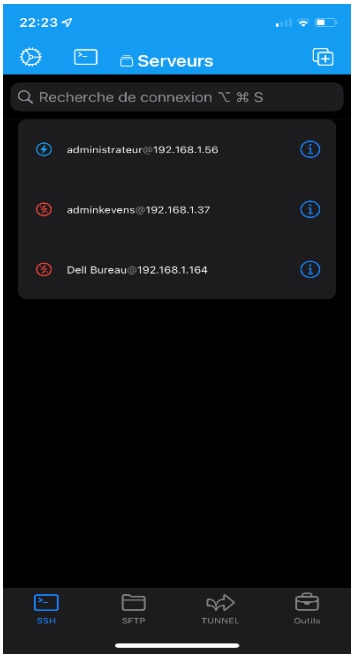
Sur mon téléphone



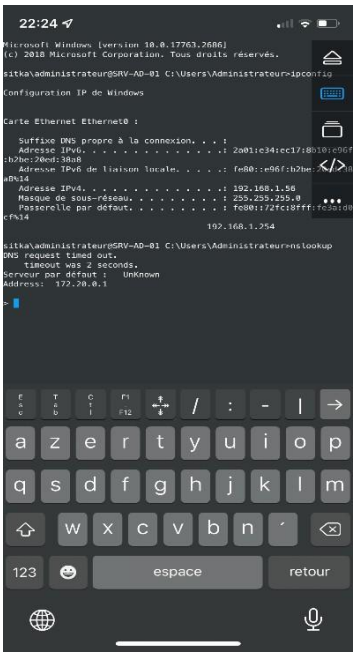
Télécharger apk WebSSH



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



Le serveur SSH apparaitre.



Voilà ! vous êtes connectée sur votre serveur SSH.

Double authentification Google

Installation du Google authentificateur :

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

Redémarrer le service : service SSHD restart ou systemctl restart SSHD

Généré le QRCODE de la double authentification Google

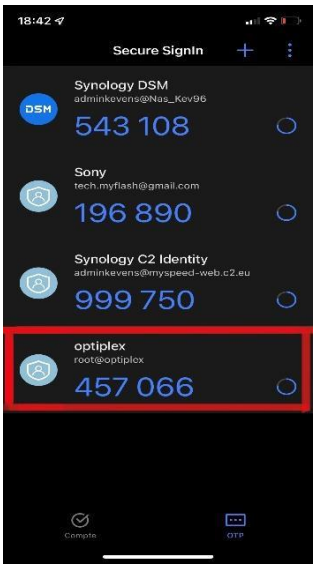
```
root@sshserver1:~# google-authenticator
```

Questions de renforcement de la sécurité authentifiée

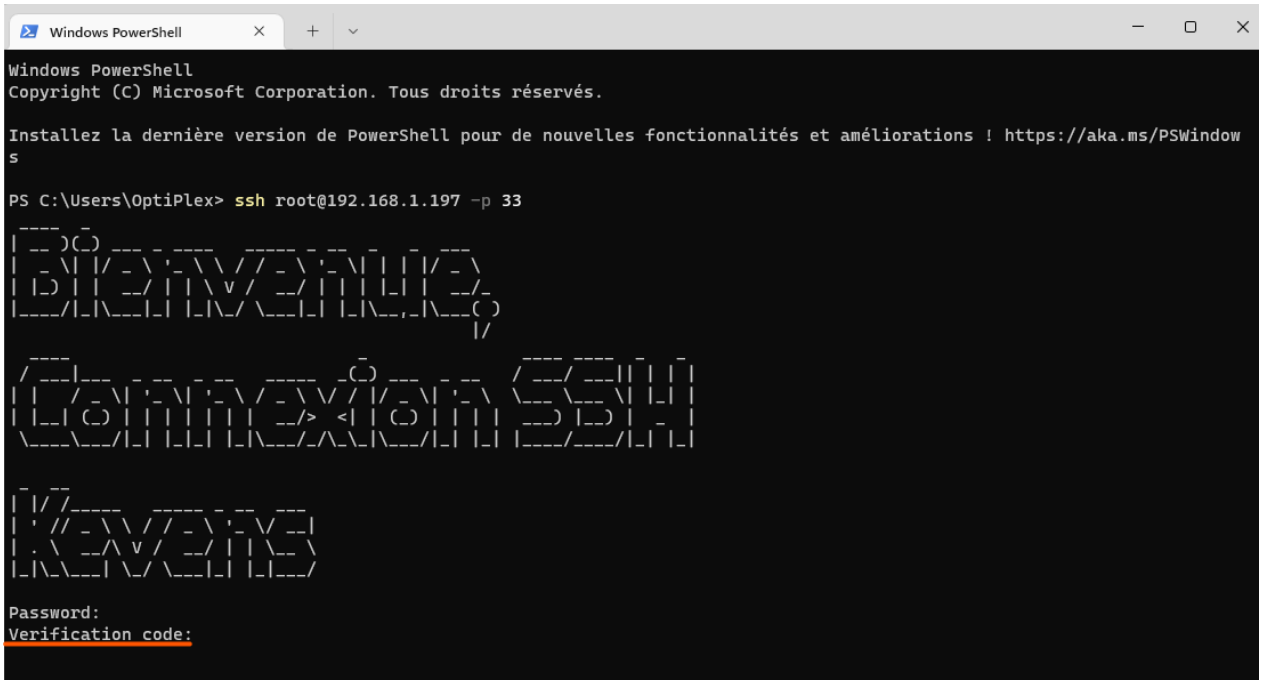
```
Your new secret key is: 4RCOU5A2IB37J6BWCBZRC5SEFA
Enter code from app (-1 to skip): 918038
Code confirmed
Your emergency scratch codes are:
  43263684
  38579593
  98493906
  92047573
  28480205

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
```

Test de la double authentification en root.

Dans un premier temps il faut installer Secure Signin sur votre téléphone.



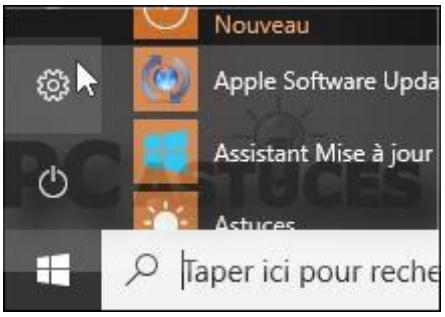
Et puis se connecter sur le serveur SSH en root, entrez le mot de passe root. Il va vous demander de vérification code de double authentification sur ton téléphone vous prenez le code à 6 chiffres.



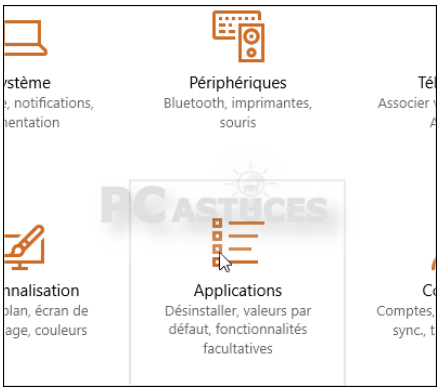
Activer le client SSH intégré à Windows

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.

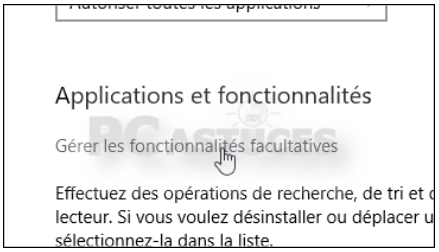
1. Le client SSH est disponible en tant qu'option et n'est pas installé par défaut. Pour l'installer, cliquez sur le bouton **Démarrer** puis sur **Paramètres**.



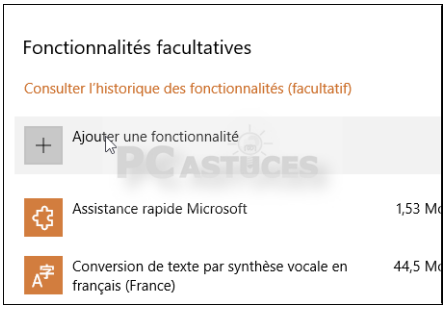
2. Cliquez sur **Applications**.



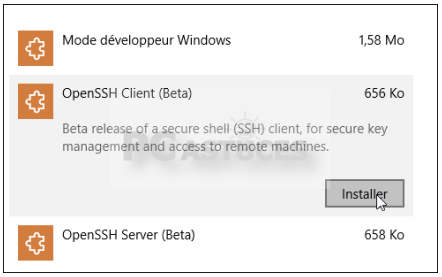
3. Cliquez sur **Gérer les fonctionnalités facultatives**.



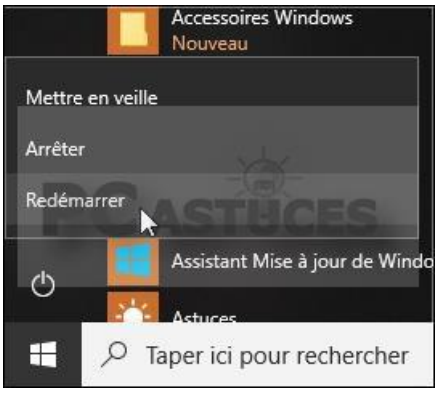
4. Cliquez sur **Ajouter une fonctionnalité**.



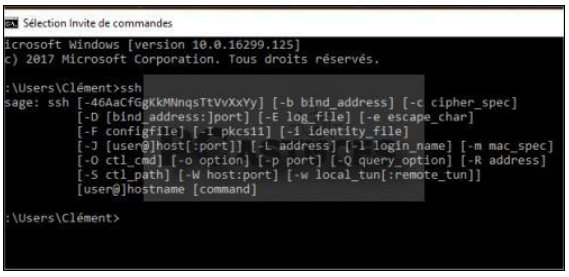
5. Cliquez sur **OpenSSH Client (Beta)** puis cliquez sur le bouton **Installer**.



6. Redémarrez votre ordinateur.



7. 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 connaitre la syntaxe de la commande.



8. Les options et la syntaxe sont les mêmes que la commande SSH sous Linux ou MacOS.