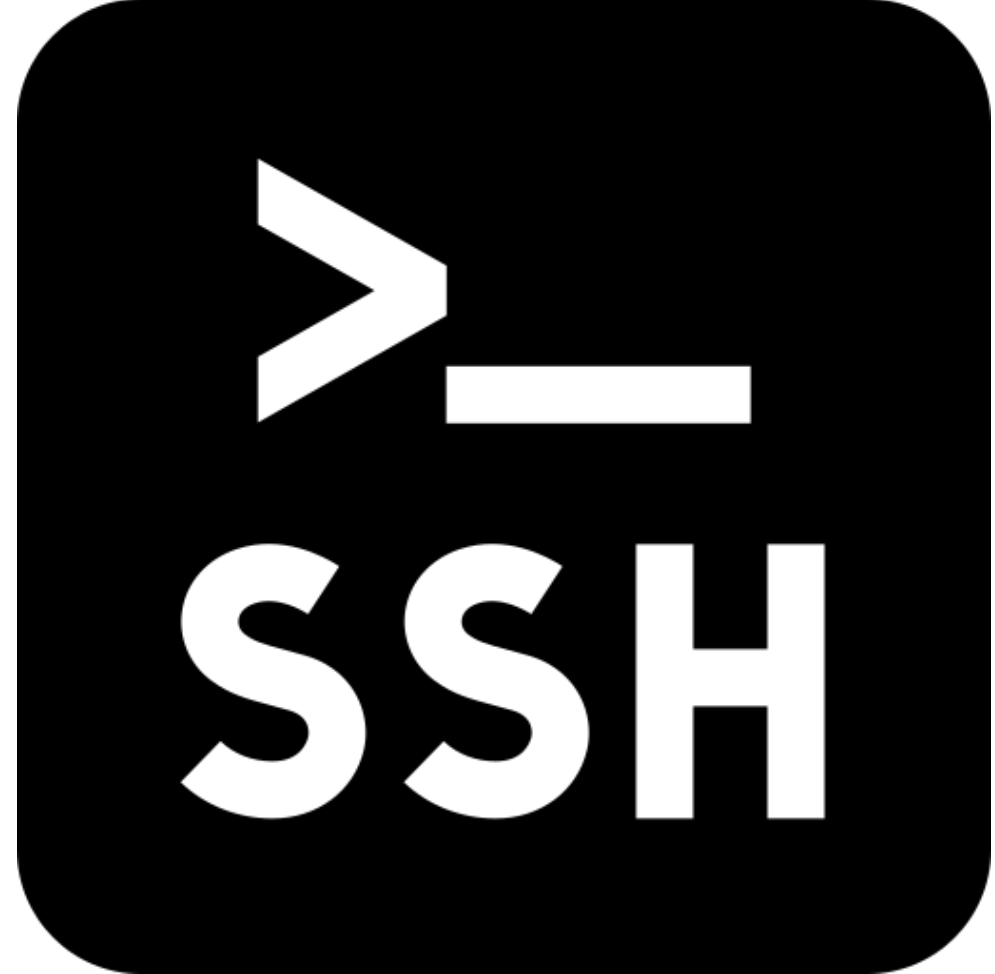
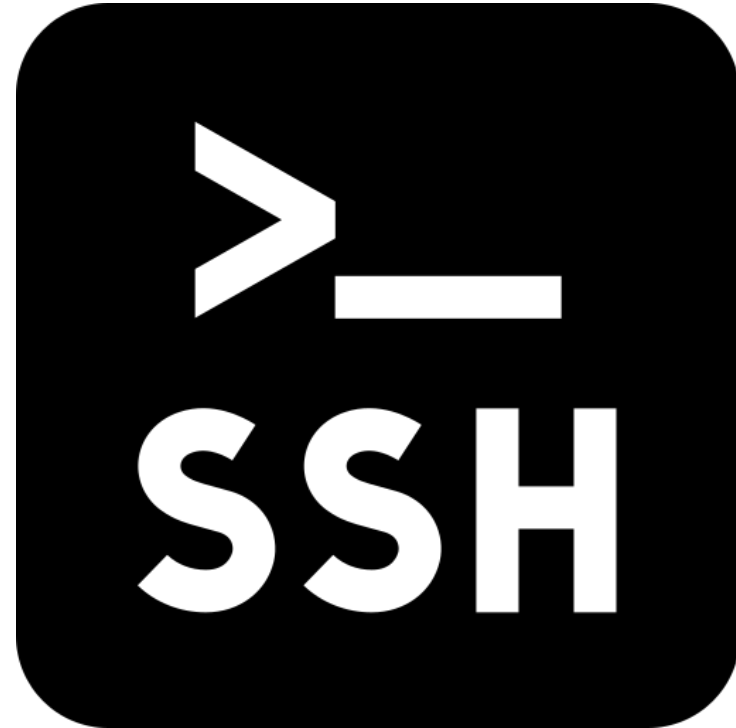


TP – Admin à
distance:
SSH



Le SSH

- Le SSH est un protocole de communication et un programme informatique qui permet notamment de se connecter à un ordinateur à distance.



Installation du SSH

- Pour installer SSH sur la machine il faudra utiliser la commande *apt install openssh-server*.

```
les paquets supplémentaires suivants seront installés :
libwrap0 openssh-sftp-server runit-helper
paquets suggérés :
molly-guard monkeysphere ssh-askpass ufw
les NOUVEAUX paquets suivants seront installés :
libwrap0 openssh-server openssh-sftp-server runit-helper
mis à jour, 4 nouvellement installés, 0 à enlever et 0 non mis à jour.
Il est nécessaire de prendre 584 ko dans les archives.
Après cette opération, 2 327 ko d'espace disque supplémentaires seront utilisés.
Souhaitez-vous continuer ? [O/n] O
Réception de :1 http://deb.debian.org/debian bookworm/main amd64 runit-helper all 2.15.2 [6 520 B]
Réception de :2 http://deb.debian.org/debian bookworm/main amd64 libwrap0 amd64 7.6.q-32 [54,9 kB]
Réception de :3 http://security.debian.org/debian-security bookworm-security/main amd64 openssh-sftp-se
Réception de :4 http://security.debian.org/debian-security bookworm-security/main amd64 openssh-server
584 ko réceptionnés en 5s (112 ko/s)
Réconfiguration des paquets...
Sélection du paquet openssh-sftp-server précédemment désélectionné.
Lecture de la base de données... 37157 fichiers et répertoires déjà installés.)
Réparation du dépaquetage de ../openssh-sftp-server_1%3a9.2p1-2+deb12u2_amd64.deb ...
Dépaquetage de openssh-sftp-server (1:9.2p1-2+deb12u2) ...
Sélection du paquet runit-helper précédemment désélectionné.
Réparation du dépaquetage de ../runit-helper_2.15.2_all.deb ...
Dépaquetage de runit-helper (2.15.2) ...
Sélection du paquet libwrap0:amd64 précédemment désélectionné.
Réparation du dépaquetage de ../libwrap0_7.6.q-32_amd64.deb ...
Dépaquetage de libwrap0:amd64 (7.6.q-32) ...
Sélection du paquet openssh-server précédemment désélectionné.
Réparation du dépaquetage de ../openssh-server_1%3a9.2p1-2+deb12u2_amd64.deb ...
Dépaquetage de openssh-server (1:9.2p1-2+deb12u2) ...
Paramétrage de runit-helper (2.15.2) ...
Paramétrage de openssh-sftp-server (1:9.2p1-2+deb12u2) ...
Paramétrage de libwrap0:amd64 (7.6.q-32) ...
Paramétrage de openssh-server (1:9.2p1-2+deb12u2) ...

Creating config file /etc/ssh/sshd_config with new version
Generating SSH2 RSA key; this may take some time ...
072 SHA256:3vl/hWj1sXbGBCikH++0rI/kIJYRgqucLkR88m/ELRo root@debiansio (RSA)
Generating SSH2 ECDSA key; this may take some time ...
56 SHA256:uo38PHamMffJYakTyAGoKN/X/c10f9cUel6SYjqTHHw root@debiansio (ECDSA)
Generating SSH2 ED25519 key; this may take some time ...
56 SHA256:h/CMk6k/UWXehVmqBdzs9eKU3f/9RmbP2SajRJKV6o root@debiansio (ED25519)
Created symlink /etc/systemd/system/ssh.service → /lib/systemd/system/ssh.service.
Created symlink /etc/systemd/system/multi-user.target.wants/ssh.service → /lib/systemd/system/ssh.servi
rescue-ssh.target is a disabled or a static unit, not starting it.
ssh.socket is a disabled or a static unit, not starting it.
Arraînement des actions différées (« triggers ») pour man-db (2.11.2-2) ...
Arraînement des actions différées (« triggers ») pour libc-bin (2.36-9+deb12u3) ...
root@debiansio:/# _
```

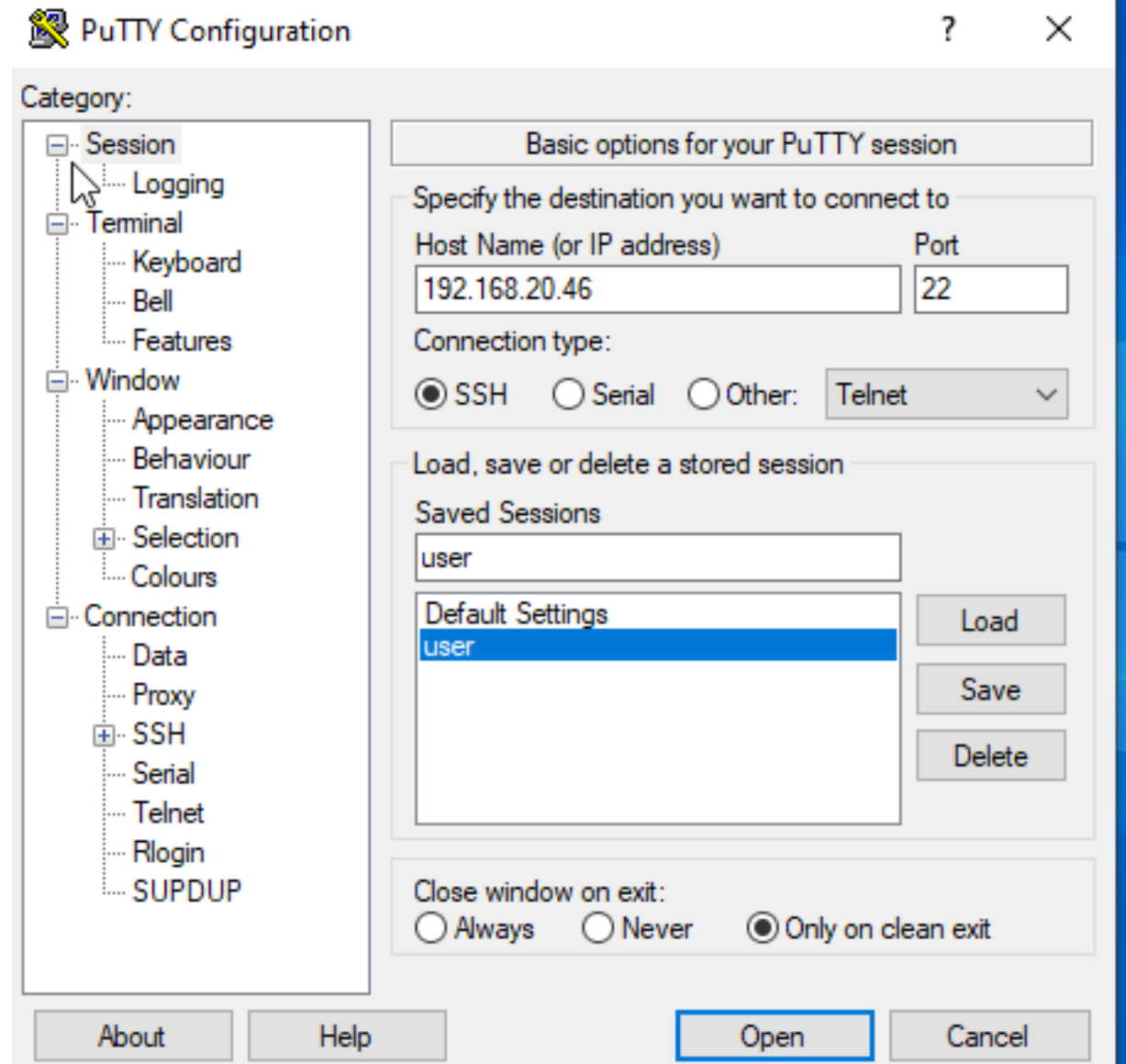
Création du User SSH

- Pour se connecter en SSH à distance j'utiliserais l'utilisateur user que je vais créer grâce à la commande *adduser user*.

```
root@debiansio:/# adduser user
Ajout de l'utilisateur « user » ...
Ajout du nouveau groupe « user » (1011) ...
Ajout du nouvel utilisateur « user » (1011) avec le groupe « user » (1011) ...
Création du répertoire personnel « /home/user » ...
Copie des fichiers depuis « /etc/skel » ...
Nouveau mot de passe :
Retapez le nouveau mot de passe :
passwd : mot de passe mis à jour avec succès
Modifier les informations associées à un utilisateur pour user
Entrer la nouvelle valeur, ou appuyer sur ENTER pour la valeur par défaut
  NOM []:
  Numéro de chambre []:
  Téléphone professionnel []:
  Téléphone personnel []:
  Autre []:
Cette information est-elle correcte ? [0/n]0
Ajout du nouvel utilisateur « user » aux groupes supplémentaires « users » ...
Ajout de l'utilisateur « user » au groupe « users » ...
```

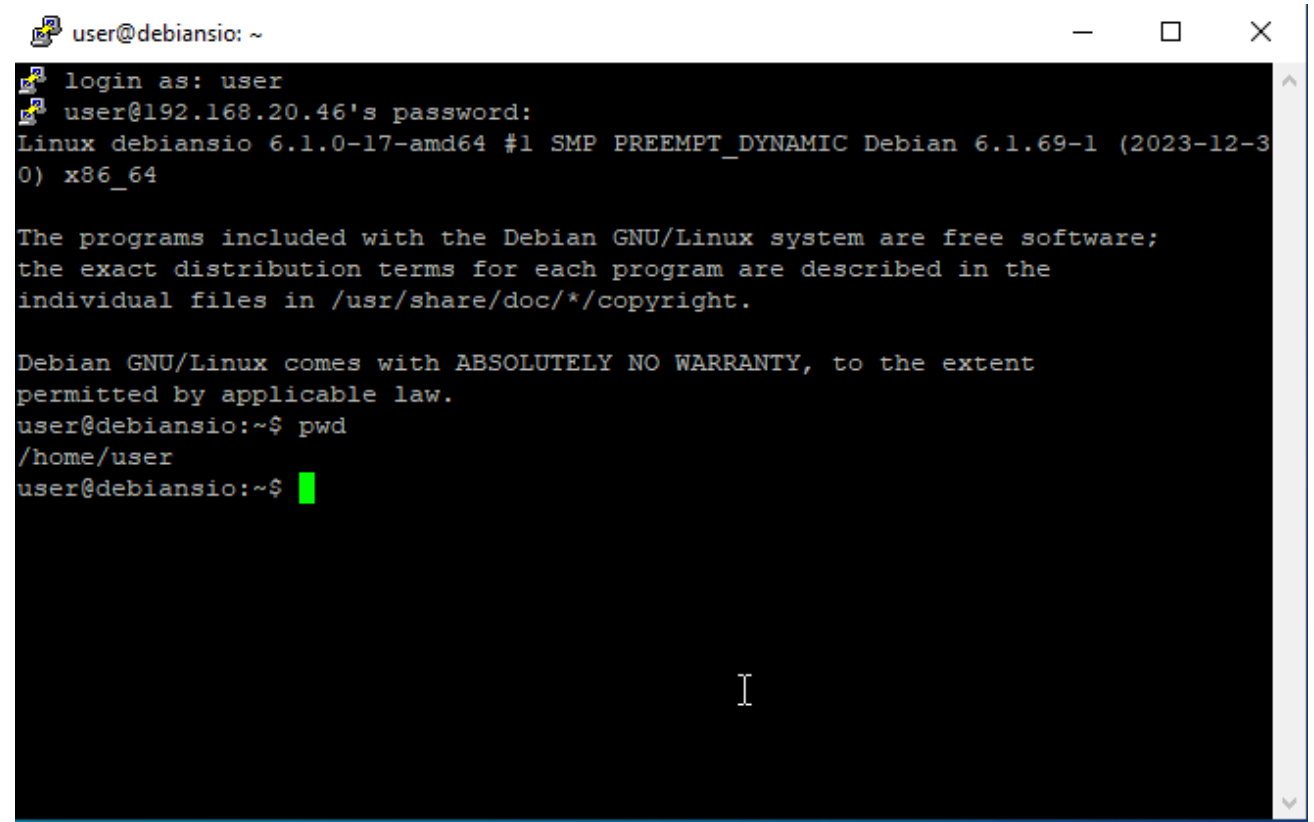
Connexion en SSH sur putty

- Il y'a plusieurs façons de se connecter en SSH dont le terminal que j'aborderais plus tard où le logiciel PuTTY.
- Pour se connecter en SSH il faudra que le client et le serveur soient sur le même réseau.
- Ici dans Host Name j'ai rentré l'IP du serveur, le port par défaut pour le SSH est le port 22, je rentre le nom de session user et je clique sur Open.



Connexion en SSH sur PuTTY

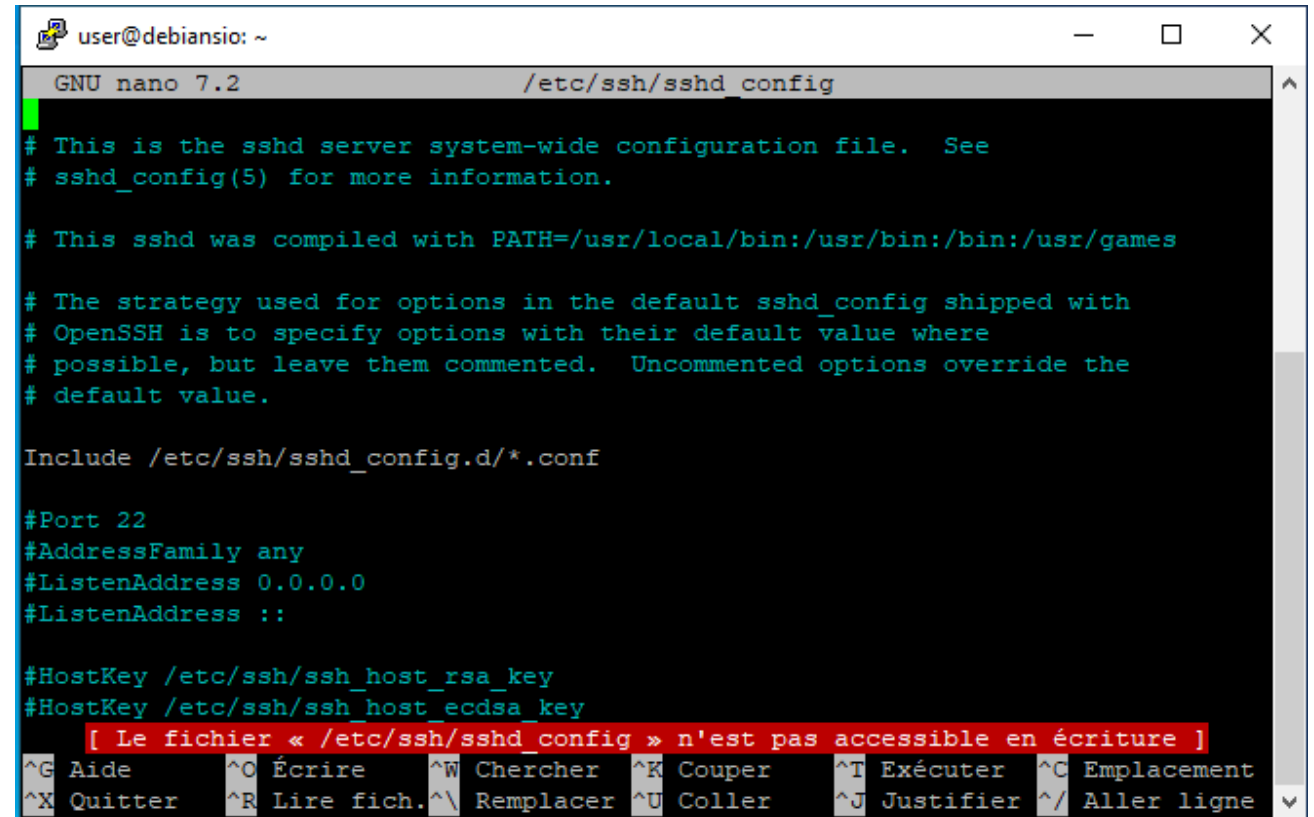
- Ensuite il faudra indiquer le nom d'utilisateur sur lequel se connecter, ici ce sera user et rentrer son mot de passe. Ensuite nous sommes connectés en SSH et nous pouvons accéder à la machine à distance.



```
user@debiansio: ~  
login as: user  
user@192.168.20.46's password:  
Linux debiansio 6.1.0-17-amd64 #1 SMP PREEMPT_DYNAMIC Debian 6.1.69-1 (2023-12-30) 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.  
user@debiansio:~$ pwd  
/home/user  
user@debiansio:~$
```

Droits d'accès en lecture/écriture

- L'utilisateur user a le droit d'afficher le fichier /etc/ssh/sshd_config mais pas de le modifier, c'est dû au fait qu'il n'a pas les permissions en écriture, il n'y'a que l'utilisateur root qui les a.



```
user@debiansio: ~
GNU nano 7.2 /etc/ssh/sshd_config
# 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 22
#AddressFamily any
#ListenAddress 0.0.0.0
#ListenAddress ::

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

[ Le fichier « /etc/ssh/sshd_config » n'est pas accessible en écriture ]
^G Aide      ^O Écrire    ^W Chercher  ^K Couper    ^T Exécuter  ^C Emplacement
^X Quitter   ^R Lire fich.^_ Remplacer  ^U Coller    ^J Justifier ^/ Aller ligne
```

Configuration du serveur SSH

- La configuration du serveur SSH se trouve dans le fichier `/etc/ssh/sshd_config` il faudra donc utiliser la commande `nano /etc/ssh/sshd_config`, pour modifier le fichier de config il faut être en root.
- Par défaut le port du SSH est 22, ici j'ai modifié le port pour mettre 2022.
- Pour prendre en compte les modifications il faut utiliser la commande `service ssh start`.

```
GNU nano 7.2 /etc/ssh/sshd_config *
# 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 2022
#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:

#LoginGraceTime 2m
#PermitRootLogin prohibit-password
#StrictModes yes
#MaxAuthTries 6
#MaxSessions 10

#PubkeyAuthentication yes

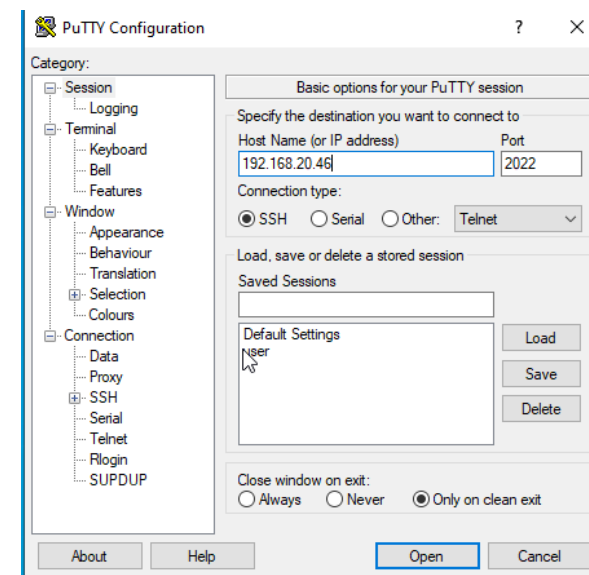
# Expect .ssh/authorized_keys2 to be disregarded by default in future.
#AuthorizedKeysFile .ssh/authorized_keys .ssh/authorized_keys2

#AuthorizedPrincipalsFile none

#AuthorizedKeysCommand none
#AuthorizedKeysCommandUser nobody
```


Connexion au port 2022

- Lorsque le port sur putty n'est pas changé, que le port 22 est toujours indiqué, cela ne fonctionne pas car le port ne correspond pas mais lorsque le port est changé et que le port 2022 est renseigné, la connexion SSH est possible.



```
user@debiansio: ~  
login as: user  
user@192.168.20.46's password:  
Linux debiansio 6.1.0-17-amd64 #1 SMP PREEMPT_DYNAMIC Debian 6.1.69-1 (2023-12-30) 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: Fri Feb  2 17:00:10 2024 from 192.168.20.49  
user@debiansio:~$
```

Intérêt du changement de port

- Nous pouvons voir avec la commande `nmap 192.168.20.46` que le port 2022 est ouvert.
- L'intérêt de ce changement de poste est qu'un potentiel attaquant testera d'abord le port 22 car c'est le port de base du SSH, donc pour pouvoir trouver le port sur lequel il devra se connecter il devra soit tester tous les ports 1 par 1, soit réussir à pénétrer le réseau pour faire un nmap.

```
root@sirs-6:/home/sirs-6# nmap 192.168.20.46
Starting Nmap 7.80 ( https://nmap.org ) at 2024-02-02 17:01 CET
Nmap scan report for 192.168.20.46
Host is up (0.00034s latency).
Not shown: 997 closed ports
PORT      STATE SERVICE
139/tcp    open  netbios-ssn
445/tcp    open  microsoft-ds
2022/tcp   open  down
MAC Address: 66:11:C0:C0:A9:85 (Unknown)

Nmap done: 1 IP address (1 host up) scanned in 0.29 seconds
root@sirs-6:/home/sirs-6#
```

Autorisation login root

- Dans le fichier de configuration `/etc/ssh/sshd_config`, la ligne `PermitRootLogin` accepte ou non les connexions en root en SSH, il y'a 3 valeurs possibles : `no`, `yes`, `without-password` (authentification par clé).
- Par défaut la valeur est à `no`, ce qui est une bonne chose question de sécurité car la permission donnée ou non est importante car l'utilisateur root à toutes les permissions de modification. J'aurais à l'autoriser plus tard mais de manière générale niveau sécurité il vaut mieux la désactiver (possible brute force).

```
GNU nano 7.2 /etc/ssh/sshd_config
# Logging
#SyslogFacility AUTH
#LogLevel INFO

# Authentication:

#LoginGraceTime 2m
PermitRootLogin no
```

PermitEmptyPasswords

- La ligne `PermitEmptyPasswords` autorise les mots de passe vides ce qui est niveau sécurité très dangereux, la valeur par défaut est `no`.
- Contrairement à `Without password` qui autorise les connexions par clé et qui requiert donc une authentification et `PermitEmptyPassword` autorise la connexion sans mot de passe.

```
GNU nano 7.2 /etc/ssh/sshd_config *
# Logging
#SyslogFacility AUTH
#LogLevel INFO

# Authentication:

#LoginGraceTime 2m
PermitRootLogin yes
#StrictModes yes
#MaxAuthTries 6
#MaxSessions 10

#PubkeyAuthentication yes

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

#AuthorizedPrincipalsFile none

#AuthorizedKeysCommand none
#AuthorizedKeysCommandUser nobody

# For this to work you will also need host keys in /etc/ssh/ssh_known_hosts
#HostbasedAuthentication no
# Change to yes if you don't trust ~/.ssh/known_hosts for
# HostbasedAuthentication
#IgnoreUserKnownHosts no
# Don't read the user's ~/.rhosts and ~/.shosts files
#IgnoreRhosts yes

# To disable tunneled clear text passwords, change to no here!
#PasswordAuthentication yes
PermitEmptyPasswords no
```

Ajout de user et groupe

- Pour la suite il faudra créer des utilisateurs qui feront partis de groupes différents, user1 dans les groupes étudiant et ssh, user2 dans ssh et user3 dans étudiant.

Pour créer les utilisateurs on utilise la commande `useradd user1` et pour les affecter à un groupe on utilise la commande `gpasswd -a user1 étudiant` (exemple pour l'user1).

- La vérification des groupes/utilisateurs se fait avec la commande `getent group`.

```
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
root@debian:~# useradd user1
root@debian:~# useradd user2
root@debian:~# useradd user3
root@debian:~# gpasswd -a user1 étudiant
gpasswd: group 'étudiant' does not exist in /etc/group
root@debian:~# groupadd étudiant
root@debian:~# groupadd ssh
groupadd: group 'ssh' already exists
root@debian:~# gpasswd -a user1 ssh
Adding user user1 to group ssh
root@debian:~# gpasswd -a user2 ssh
Adding user user2 to group ssh
root@debian:~# gpasswd -a user3 étudiant
Adding user user3 to group étudiant
root@debian:~#
```

```
ssh:x:111:user1,user2
bluetooth:x:112:debian
avahi-autoipd:x:113:
debian:x:1000:
systemd-coredump:x:999:
user1:x:1001:
user2:x:1002:
user3:x:1003:
étudiant:x:1004:user3,user1
root@debian:~#
```

```
ssh:x:1010:user1,user2
user:x:1011:
user1:x:1012:
user2:x:1013:
user3:x:1014:
étudiant:x:1015:user1,user3
```

Attribution des mots de passe

```
root@debiansio:/# chpasswd  
user1:Password1  
user2:Password1  
user3:Password1  
root@debiansio:/# _
```

- Pour attribuer les mots de passe il faut utiliser la commande *chpasswd* et renseigner le nom d'utilisateur suivi de « : » et du mot de passe.
- Pour sortir de la configuration il suffit d'appuyer sur CTRL + D.

Gérer l'échange des clés publiques - Sur le client

- Il faudra créer un repertoire pour les utilisateurs dans le /home et utiliser la commande `mkdir /user1` (exemple pour le user1). Ensuite il faudra créer un repertoire .ssh dans leur repertoire crée précédemment, pour prendre l'exemple du user1, depuis le /home il faudra utiliser la commande `cd user1/` pour se rendre dans le repertoire personnel et utiliser la commande `mkdir .ssh/`

```
root@debiansio:/home# cd user1/
root@debiansio:/home/user1# mkdir .ssh/
mkdir: impossible de créer le répertoire « .ssh/ »: Le fichier existe
root@debiansio:/home/user1# cd ..
root@debiansio:/home# cd user2/
root@debiansio:/home/user2# mkdir .ssh/
root@debiansio:/home/user2# cd ..
root@debiansio:/home# cd user3/
root@debiansio:/home/user3# mkdir .ssh/
root@debiansio:/home/user3#
```

```
root@debiansio:/# cd /home
root@debiansio:/home# mkdir user1
root@debiansio:/home# mkdir user2
root@debiansio:/home# mkdir user3
```

Changement des droits

- Pour changer les droits des users, on utilisera la commande `chown user1 .ssh` (exemple qui change la propriété du repertoire `.ssh` dans `user1` à l'utilisateur `user1`).

```
root@debiansio:/home/user1# chown user1 .ssh
root@debiansio:/home/user1# cd ..
root@debiansio:/home# cd user2/
root@debiansio:/home/user2# cd user2 .ssh
-bash: cd: trop d'arguments
root@debiansio:/home/user2# chown user2 .ssh
root@debiansio:/home/user2# cd ..
root@debiansio:/home# cd user3/
root@debiansio:/home/user3# chown user3 .ssh
root@debiansio:/home/user3#
```


Génération de clé DSA pour l'utilisateur

- Pour générer une clé DSA, il faut utiliser depuis la machine cliente la commande `ssh-keygen`, puis rentrer le mot de passe.
- Les fichiers générés sont `id_rsa` et `id_rsa.pub`, ce dernier représentant la clé publique et l'autre la clé privée

```
sisr-6@sisr-6:~$ ssh user1@192.168.20.46
user1@192.168.20.46's password:
Linux debiansio 6.1.0-17-amd64 #1 SMP PREEMPT_DYNAMIC Debian 6.1.69-1 (2023-12-30) 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: Wed Feb  7 15:43:07 2024 from 192.168.20.111
```

```
$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/home/user1/.ssh/id_rsa):
/home/user1/.ssh/id_rsa already exists.
Overwrite (y/n)? y
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/user1/.ssh/id_rsa
Your public key has been saved in /home/user1/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:1mnZucF8KP5P1hMtNPXzrynGSA62ZSdQEinBc06W5s user1@debiansio
The key's randomart image is:
+---[RSA 3072]-----+
|      .+B+      |
|      .++      |
|      .o  o..   |
|    +..* + .+   |
|    S=*oB o.o   |
|    .+oE..+ oo  |
|    . B.=. o.o  |
|    . o.+o o.   |
|    ..o+        |
+---[SHA256]-----+
```

```
root@debiansio:/home/user1# cd .ssh/
root@debiansio:/home/user1/.ssh# ls
id_rsa  id_rsa.pub
root@debiansio:/home/user1/.ssh#
```

Génération des clés publiques à destination du serveur

- La clé publique est le moyen pour le serveur de nous identifier, pour générer la clé publique il faudra utiliser la commande `ssh-copy-id -i ~/.ssh/id_dsa` et rentrer une nouvelle fois un mot de passe.



```
debian vierge [En fonction] - Oracle VM VirtualBox
Fichier  Machine  Écran  Entrée  Périphériques  Aide

+----[SHA256]-----+
root@debian:~# ssh user1@192.168.20.46
The authenticity of host '192.168.20.46 (192.168.20.46)' can't be established
ECDSA key fingerprint is SHA256:uo38PHamMffJYakTyAGoKN/X/c10f9cUe16SYjqtH
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.20.46' (ECDSA) to the list of known hosts
user1@192.168.20.46's password:
Linux debiansio 6.1.0-17-amd64 #1 SMP PREEMPT_DYNAMIC Debian 6.1.69-1 (20240816)

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: Wed Feb  7 15:52:30 2024 from 192.168.20.111
$ ssh-keygen -t dsa -f ~/.ssh/id_dsa
Generating public/private dsa key pair.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/user1/.ssh/id_dsa
Your public key has been saved in /home/user1/.ssh/id_dsa.pub
The key fingerprint is:
SHA256:npKI41xxU10VkyAJaYiCSP4CS1NJ4650w8y1A1GJEK4 user1@debiansio
The key's randomart image is:
+---[DSA 1024]-----+
|B+==+.0..00.0      |
|*0+++.0 .0  +      |
|0* ... 0   .        |
|+.0 . 0 .          |
|Eo @ + S           |
|. +.=..0 .         |
|.....0 0          |
|0..               |
| 0                 |
+----[SHA256]-----+
$
```

Envoi de la clé publique au serveur

- Pour pouvoir envoyer la clé publique au serveur, il faut utiliser la *commande ssh-copy-id -i ~/.ssh/id_dsa.pub root@192.168.20.46*
- Pour que cela soit bien envoyé il faudra que la valeur PermitRootLogin soit à yes dans le fichier de configuration SSH car il faudra s'identifier en root pour l'envoyer.

```
root@debian:~# ssh user1@192.168.20.46
user1@192.168.20.46's password:
Linux debiansio 6.1.0-17-amd64 #1 SMP PREEMPT_DYNAMIC Debian 6.1.69-1 (2023-12-30) 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: Wed Feb  7 16:33:35 2024 from 192.168.20.111
$ ssh-copy-id -i ~/.ssh/id_dsa.pub root@192.168.20.46
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/home/user1/.ssh/id_dsa.pub"
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are alr
eady installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted now it is to inst
all the new keys
root@192.168.20.46's password:
/root/.bashrc: ligne 19: setxkbmap : commande introuvable

Number of key(s) added: 1

Now try logging into the machine, with:  "ssh 'root@192.168.20.46'"
and check to make sure that only the key(s) you wanted were added.
```

Clés dans les différents fichiers

- Les clés se retrouvent dans deux fichiers différents, le fichier `~/.ssh/knownhosts` et `~/.ssh/authorized_keys`
- Les clés situées dans `~/.ssh/knownhosts` sont les clés publiques et celles situées dans `~/.ssh/authorized_keys` sont les clés privées.

```
GNU nano 7.2 known_hosts
#1 |kWFUdwdtp0uISV39Ig7LUcTdpc=|b/hI19fWss4/UKeu9sAG3HteAkW= ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAIAbCLodE6hGGqW+R47oEcLMvcY0Y6mNzChzctMy6Sgu9
#1 |5m97C56+V4jLVbeo2bXoHop2HbY=|Cy6KS61lmRskCHQME+uFPzoCjc= ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQGDkJah8IvTn7YVwtz8RGq8/p6fsS600kJ3wgmLbU3jjoFJgrHE1oPMVST79ot>
#1 |8jGq8UVyfrfbvFOYiFAWreKfiJI=|dv1tsfDkX2uS0nP7A527h3uBzJc= ecdsa-sha2-nistp256 AAAAE2VjZHNhLXNoYTItbmlzdHAyNTYAAAAIbmlzdHAyNTYAAABBBJ7zX71xcuKB51tNTN3L1ejFk1>
```

```
GNU nano 7.2 authorized_keys
ssh-dss AAAAB3NzaC1kc3MAAACBAFe41cJCUmY/Iq2rvZV9gMG8+4YX3wnNm1q5bMcgFr0svWmxILa70AYagSLJ6HqPmda0hJnILhXhh7kCK1g40H9jADcMpf/O1RPNshC1tHumY2H6Yct/NvQW7RH8t/HdnuM>
ssh-dss AAAAB3NzaC1kc3MAAACBAIoCb77TQckAc794A72D7HzvydGjmeSA9Z2Nbi6L175xTw64v1I5M6pu8Gbkyc215MSdRXIoLFDrK8BX0naE29WHyr46D6vrQbRrT6T9x8mg2kRHLERiohmSw5I3MiT7XMn>
ssh-dss AAAAB3NzaC1kc3MAAACBAKoFe1LBYTs1CKQboWU4D+X0n8XfNJNakWPhUsr7L7u3mPIM39tev8SnrICBuTP42S1o1UGsm/1PfQ14Vx0GqYhV6a9AB2XbT5VYwBk/CsHVgmu+KumNfQ1m8p7ccCxGwuT>
```

Test de la connexion

- Ensuite je teste la connexion avec la commande `ssh 192.168.20.46` depuis la machine cliente
- Et depuis le serveur j'utilise la *commande* `ssh user1@192.168.20.46 -p 22`
- La connexion est bien établie.

```
root@debian:~# su user1
$ ssh 192.168.20.46
The authenticity of host '192.168.20.46 (192.168.20.46)' can't be established.
ECDSA key fingerprint is SHA256:uo38PHamMffJYakTyAGoKN/X/c10f9cUel6SYjqtHHw.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Could not create directory '/home/user1/.ssh' (No such file or directory).
Failed to add the host to the list of known hosts (/home/user1/.ssh/known_hosts).
user1@192.168.20.46's password:
Linux debiansio 6.1.0-17-amd64 #1 SMP PREEMPT_DYNAMIC Debian 6.1.69-1 (2023-12-30) 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: Wed Feb  7 17:09:12 2024 from 192.168.20.46
```

```
root@debiansio:~# ssh user1@192.168.20.46 -p 22
user1@192.168.20.46's password:
Linux debiansio 6.1.0-17-amd64 #1 SMP PREEMPT_DYNAMIC Debian 6.1.69-1 (2023-12-30) 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: Wed Feb  7 17:10:17 2024 from 192.168.20.111
$
```

Autorisation des utilisateurs des groupes root et ssh

- Dans le fichier de configuration, AllowGroups permet d'autoriser ou non la connexion de certains groupes d'utilisateurs, ici j'autorise uniquement les groupes ssh et root.

```
GNU nano 7.2 /etc/ssh/sshd_config *
# 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 22
#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:
AllowGroups ssh root
#LoginGraceTime 2m
PermitRootLogin yes
```

Test connexion

- Pour tester que la configuration a bien été prise en compte il faut tester l'utilisateur user2 membre du groupe ssh qui devrait pouvoir se connecter et user3 uniquement membre du groupe etudiant qui n'est pas autorisé.
- Donc les permissions ont donc bien été prises en compte.

```
sisr-6@sisr-6:~$ ssh user2@192.168.20.46
user2@192.168.20.46's password:
Linux debiansio 6.1.0-17-amd64 #1 SMP PREEMPT_DYNAMIC Debian 6.1.69-1 (2023-12-30) 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: Wed Feb  7 16:40:51 2024 from 192.168.20.111
$
```

```
sisr-6@sisr-6:~$ ssh user3@192.168.20.46
user3@192.168.20.46's password:
Permission denied, please try again.
user3@192.168.20.46's password:
Permission denied, please try again.
user3@192.168.20.46's password:
```