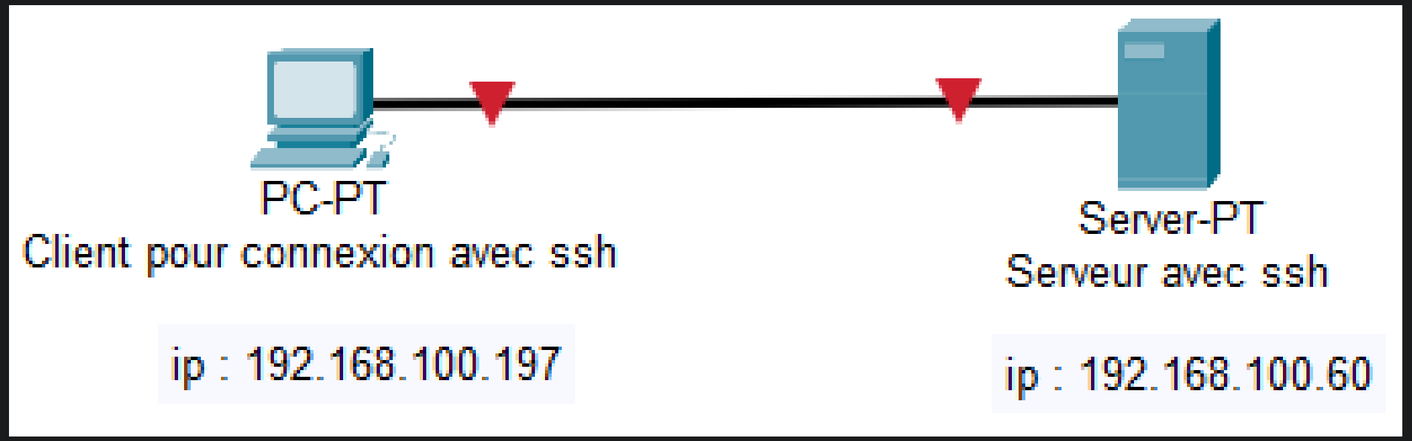
TP- SSH

DOMONT ALEXANDRE

SOMMAIRE

1. SCHÉMA DU RÉSEAU



2. INSTALLATION ET CRÉATION D'UN UTILISATEUR POUR SSH

```
root@debiansio:~# which ssh
/usr/bin/ssh
root@debiansio:~# _
```

De par cette commande nous verrifions si ssh est installer. Ici oui Le programme ssh est installé il est situer dans le répertoire /usr/bin/

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

Part précotion nous l'installons quand même.

```
root@debiansio:~# systemctl start ssh
root@debiansio:~# systemctl enable ssh
Synchronizing state of ssh.service with SysV service script with /lib/systemd/systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install enable ssh
root@debiansio:~# systemctl status ssh
● ssh.service - OpenBSD Secure Shell server
     Loaded: loaded (/lib/systemd/system/ssh.service; enabled; preset: enabled)
     Active: active (running) since Thu 2025-02-06 14:13:22 CET; 16min ago
       Docs: man:sshd(8)
             man:sshd_config(5)
   Main PID: 752 (sshd)
      Tasks: 1 (limit: 2306)
     Memory: 2.9M
        CPU: 62ms
     CGroup: /system.slice/ssh.service
             └─752 "sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups"

févr. 06 14:13:22 debiansio systemd[1]: Starting ssh.service - OpenBSD Secure Shell server...
févr. 06 14:13:22 debiansio sshd[752]: Server listening on 0.0.0.0 port 22.
févr. 06 14:13:22 debiansio sshd[752]: Server listening on :: port 22.
févr. 06 14:13:22 debiansio systemd[1]: Started ssh.service - OpenBSD Secure Shell server.
root@debiansio:~#
```

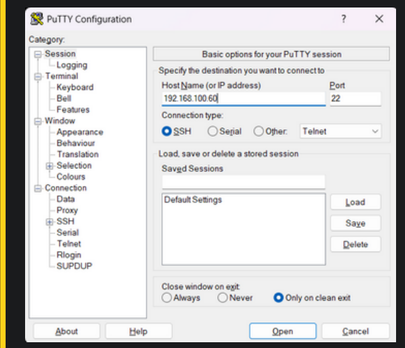
Ici nous vérifions que le service est bien actif

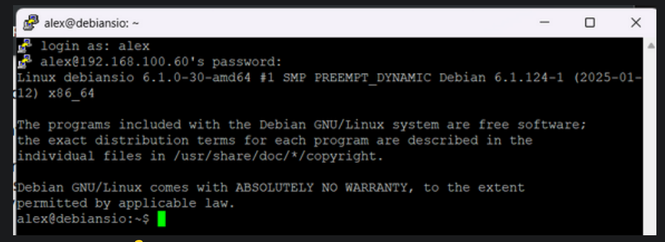
Pour utiliser ssh nous créons un user ici nommé alex avec comme mot de passe *Sio2O2O

```
root@debiansio:~# adduser alex
```

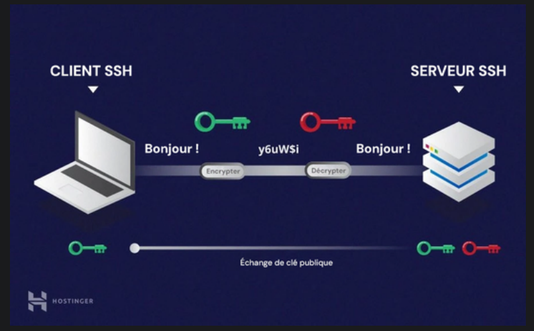
Et ici nous le mettons dans le groupe ssyh.

```
root@debiansio:~# usermod -aG _ssh alex
root@debiansio:~#
```





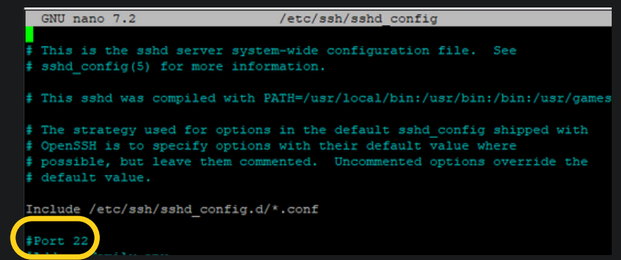
On peut voir que la connexion seffectue correctement.

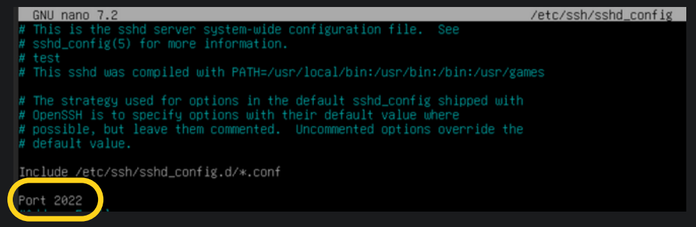


SSH signifie Secure Shell. C'est un protocole qui permet de se connecter à distance à un ordinateur de manière sécurisée. Il chiffre les données pour empêcher les interceptions, ce qui rend les communications entre deux ordinateurs confidentielles et protégées.
SSH emploie des paires de clés publique et privée pour l'authentification et l'établissement initial de la connexion sécurisée.

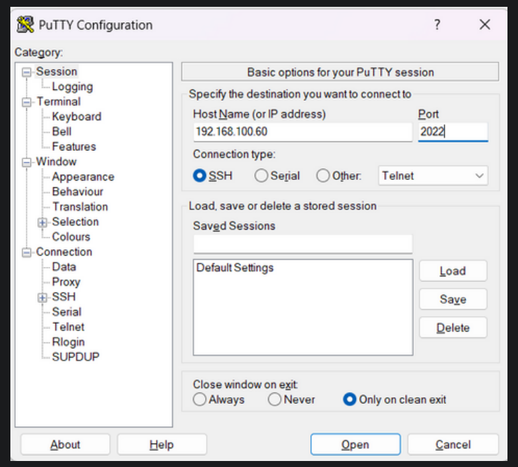
3. CHANGER LE PORT D'ÉCOUTE

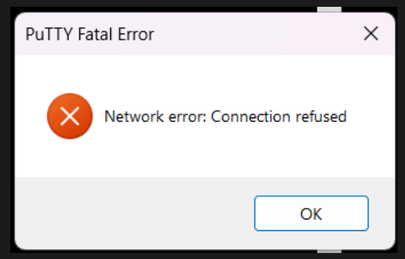
```
root@debiansio:/home/alex# nano /etc/ssh/sshd_config
```





Ici nous nous rendons la le fichier de configuration de ssh pour passer du port 22 au port 2022. Il est important de accèder au fichier temps qu'utilisateur root car le fichier est un fichier de configuration système critique qui contient des paramètres de sécurité importants pour le service SSH. Donc en tant qu'utilisateur non root on ne peut pas accéder en écriture sur le fichier.





Si nous changeons le port d'écoute sans redémarrer le service alors la connexion échouera car le client SSH essaiera toujours d'utiliser le port par défaut dont le 22.

Le changement de port SSH présente quelques avantages :

1. Réduction des attaques automatisées ciblent spécifiquement le port 22.
2. Ralentir les attaquants moins sophistiqués.
3. Détection plus rapide des tentatives d'intrusion : En combinaison avec d'autres mesures de sécurité comme un IDS, le changement de port peut forcer les attaquants à effectuer plus d'opérations, facilitant leur détection.

```
[ Le fichier « /etc/ssh/sshd_config » n'est pas accessible en écriture ]
```

Ceci est affiché en bas du fichier si l'on essaie de se connecter avec un utilisateur normal.

Ici, pour se connecter depuis un client Windows, nous utilisons PuTTY.

4. CRÉATION DES UTILISATEURS, GROUPES ET DOSSIERS .SSH

USER/GROUPE

```
root@debiansio:~# groupadd etudiant
root@debiansio:~# groupadd ssh
```

Ici nous créons les groupes.

```
root@debiansio:~# adduser user1
root@debiansio:~# adduser user2
root@debiansio:~# adduser user3
```

Ici nous créons les utilisateurs.

```
root@debiansio:~# usermod -aG etudiant,ssh user1
root@debiansio:~# usermod -aG ssh user2
root@debiansio:~# usermod -aG etudiant user3
```

Ici nous attribuons des groupes aux différents utilisateur

```
root@debiansio:~# groups user1
user1 : user1 users etudiant ssh
root@debiansio:~# groups user2
user2 : user2 users ssh
root@debiansio:~# groups user3
user3 : user3 users etudiant
```

ici nous vérifions dans quel groupe se situe les users

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

Grace à cette commande noous pouvons changer les mot de passe des utilisateur.

DOSSIER

```
alex@debiansio:/home$ pwd
/home
```

Nous nous rendons dans le répertoire home et nous veerrifions avec la commande pwd.

```
root@debiansio:/home# mkdir -p user1/.ssh user2/.ssh user3/.ssh
```

grace à la commade mkdir nous créons les répertoire .ssh dans chaque user

```
root@debiansio:/home/user1# ls -a
.  ..  .bash_logout  .bashrc  .profile  .shh  .ssh
root@debiansio:/home/user2# ls -a
.  ..  .bash_logout  .bashrc  .profile  .shh  .ssh
root@debiansio:/home/user3# ls -a
.  ..  .bash_logout  .bashrc  .profile  .ssh
```

Ici nous verrifions dans chaques dossier si le fichier .ssh est bien mis avec la commande ls -a. Le -a est la pour montrer les fichier cacher car les fichier en .quelque chose sont des fichier cacher.

```
root@debiansio:/# chmod 0770 /home/user1/.ssh
root@debiansio:/# chmod 0770 /home/user2/.ssh
root@debiansio:/# chmod 0770 /home/user3/.ssh
root@debiansio:/# chmod 0770 ~/.ssh
```

Pour finir nous mettons sur chaque dossier .ssh de chaque user des droits.

5. GÉRER LES CLÉS D'AUTHENTIFICATION SUR LE SERVEUR ET LE CLIENT

```
root@debiansio:/# ssh-keygen -t rsa -f root/.ssh/id_rsa
Generating public/private rsa key pair.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in root/.ssh/id_rsa
Your public key has been saved in root/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:8QzDiZ3mr7JPvmoadeIFsA4Mk3E8Dcq/MLnaIFxGJdM root@debiansio
The key's randomart image is:
+---[RSA 3072]----+
|++O=O.           |
|o*O =E + o       |
|..oo. o X        |
|  +o   + *       |
| + +. o S o      |
|. * .O + .       |
|oo .. . . .      |
|oo   .oo .       |
|. . .oo==.       |
+----[SHA256]-----+
root@debiansio:/#
root@debiansio:/# _
```

ici nous générons une clé asymétrique du type RSA.

```
root@debiansio:/# ssh-copy-id -i /root/.ssh/id_rsa.pub root@192.168.100.60
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/root/.ssh/id_rsa.pub"
The authenticity of host '192.168.100.60 (192.168.100.60)' can't be established.
ED25519 key fingerprint is SHA256:SrjItcg/eREY4kNYHnjx4DmEJH7lPl/sgDSslg7g6k8.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted now it is to install the new keys
root@192.168.100.60's password:

Number of key(s) added: 1

Now try logging into the machine, with:   "ssh 'root@192.168.100.60'"
and check to make sure that only the key(s) you wanted were added.

root@debiansio:/#
```

Avec cette commande on envoie la clé public au serveur ssh.

```
root@debiansio:~/.ssh# cat id_rsa
-----BEGIN OPENSSH PRIVATE KEY-----
b3BlbnNzaC1rZXktdjEAAAAABG5vbmUAAAAEbm9uZQAAAAAAAAABAAABlwAAAAdzc2gtcn
NhAAAAAwEAAQAAAYEAvcVkOojCRX4JMD3QevTWFJqV20o183IAOL8gA5XJx1Oa+B68tpqs
Fkgex2cA9udQoNv/LkN9vTrVl222bQcRDSskh/AGRuDpUoZ6orx+59w2BqX7QrGhQYuTZM
H6S4weh/1SF19eF7VI6RO80SavjvAXnEVRJeFk7Na1vc+17LKpJihRFZbz3bbjWd2VrGnO
XvHLZGHO2v/HAS+0naOJYSHDxbDCrf2RlZHGUQQ5oAgE/fgzRbbAwmrfHz2nmTGhaSggo5
HbzoK0tH3ZirbPSYBqzNZwBA89/4pg+vcrh1KHRkIuT5dxYyMB8RCpF2+vz6hAgVOqfQm2
+5EkLjA2jzTAVKwnAvgnFmm7/udgh7X24ju0/S1+kSS+86JKpqK2VRvwf1IE580jc0T3Evr
PrEKC2YRD65yteBGXPylV41+efameR96Y7uP4sCVKIN2ty9lB19Ncv/sX3U20ArokV7SxK
RlK/JEV6Jf+10hAzO99qDuM3fsYgtvmeJ5PFHvjrAAAFiOcbT5fnG0+XAAAAB3NzaC1yc2
EAAAGBAL3FSJqIwkV+CTA90Hr0ihSaldtKNfNyADi/IAOVycdTmvgevLaarBZIHsWXAPbn
UKDb/y5Dfb061Zdmdm0HEQ0rJIfwBkbg6VkGeqK8fufcGQal+0KxoUGLk2TB+kuMHof9Uh
dfXheiSOkTvNEmr47wF5xFUSXhSuzWtb3PteyyqSYoURWk892241nWVaxpzl7xy2Rhztr/
xwEvtJ2jiWEhw8WuwQ39kYmRxlEEOaAIBP34M0K2wMJq3x89p5kxoWkoIkOR286JDrR92Y
q2z0mAaszdsAQPPf+KyPr3K4dSh0ZCLk+XcWMjAfEQqRdvr8+oQIFTqn0JmfuRJC4wNo80
wFVsJwL4JxZpu/7ncoe12eI7tP0tfkkvvOoyqaitlUb0H9SBOfNI3NE9xL6z6xCgmWEQ+u
crXgRlz75VeNfn2pnkfemO7j+LAlSiDWbb/ZQdfTXL/7F91GdAK6JFe0sSk2Sv4xFeiX/
tdIQMzvfag7jN37GILb5nieTxR746wAAAAMBAAEAAAGASUnONvjl/iyEo+rbfwDbRK3Gvh
/U4iXezAuT845/lW1HChfkzaJCWQCVUb3Vk4UV8GnGDTnnb5MvNf9OMSEPhsRooU8jPXBS
65of9oVRhX3M9HJZrRIossHSsNDxNwfuKpi8gWZFUmait9D1lNiZ6Fv04lcq1bRCWHE6Yw
m6uqTtCli7oxjfKXseJm3OPH3IQ4aiBCx51L3/PwgLeu3Ten2tu8VeCg/pivWPX26ASruO
If8e3geMPglED16JcVhm0O6xqUvjezjrDond24WXQgA5FOwbrfY2K79acWa3FW8F69lcnv
PYXKcbOseuHe6m+LIzst4MQEL5wMC9+oMpIN1aRskKePnrUIjBh7X3UI9xj1OoIFGYBzQd
RAlg+zxsUkyPwk1ROIFEUufxP6G1UrcLgWQWk0g2JqLLiCJU3Q+K1ifFJJs/7FyDqNXuGn
lirkT6a2aPtQxgk8tYRQ3Q5qWHyw9LIQqCIdFsiYgIEprKL6NtTnhD90jaVKMRs0RAAAA
wH2VbP3dED2p5cA5MlqGLTMsoKgIzDwfmtP0OLX7ynneGoHzm8U8hTA5wMNc1O/IUIqRD
OS2m2iL8w2a22wq3xpwhY9S2dt4KbyxPg2erT0kgSRpWHNQzzRJQNL9AGIPDC+4C90okN
Py9Gy9FugLQ44Vqe506KrqP6LLA3i1hCOcZk2xfvazJix2flvi8eSitlkpbh7/n6yW6bXi
xd5OSWfTH0DykS5BYZy12h0mhNvi6e9uJYIRdJuSvRFffd1uAAAMEA+c/NZJqcbsbzPeRc
xWOPQNaalG/uE7ztRufPz3eP20uv2avum2Wx5MLg53z9iF1hPWSE3vMMZcvnVxwk2relnb
S/vbMiY78hrcROUdxTzo4DJMYXiPDvXMQ7N2Xx89UkB02DYSWRSJVfU9k5/F13I2l1bRk8
gjhguWMPF6M65yB0n5QQ9M0V5nbp7LgZGWuMl+bWWboHkHknyu7cQyOo0g9PkayP7Ir3aT
VkCPp0RtjC6zeiqJuv2+Kn0TPq2uhPAAAAwQDCeLu1nUfWvfpK/xQsCe6I4YijWRJ2uWma
f5bECOuk+D+Qrte+OSio9pg/vcIJjSEVIqd+1lwzh3kxr9u7uf2swaeVPc9MN/IvvamMro
+tdJt1+j4x9/f3Jp8d/iuNdx/W3anfuVXe4y6igrO/aj9fwuKT2RzhiFjXc8VqdIhuMdbp
yzfRdSZP0qmHgh8fS4NAsHWpVP/a6NeI/LYv0gl4/h2DrGop6ug+e+MpL6GYhPnhnYR+YB
sbZWsTQSYXYqUAAAAOcm9vdEBkZWJpYW5zaW8BAgMEBQ==
-----END OPENSSH PRIVATE KEY-----
```

```
root@debiansio:~/.ssh# ls
id_rsa  id_rsa.pub  known_hosts  known_hosts.old
root@debiansio:~/.ssh# _
```

Nous pouvons voir dans le fichier /root/.shh que deux fichier on étaient générer : une clé public à était générer et une clé priver.

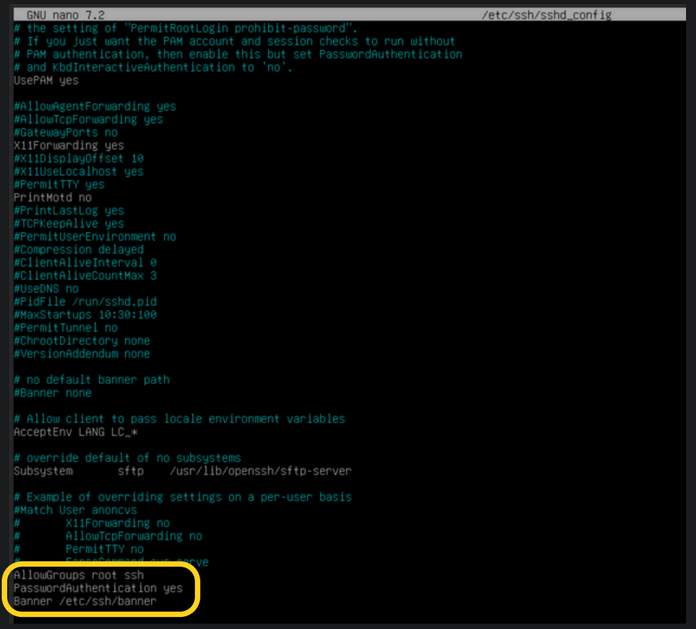
```
root@debiansio:~/.ssh# cat id_rsa.pub
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABgQC9xUo6iMJFfgkwPdB69NYUmpXbSjXzcgA4vyADlcnHU5r4Hry2mqwWSB7FlwD251Cg2/8uQ329OtWXZn2tBxENKySH8AZG4OlShnqivH7n3BkGpftCsaFBi5Nk
wfpLjB6H/VIXX14XtUjpE7zRJq+O8BecRVEl4Urs1rH9z7XssqkmKFEVlvPdtuNZ1lWsac5e8ctkYc7a/8cBL7Sdo4lhIcPFsMKt/ZGJkcZRBDmgCAT9+DNFtsDCat8fPaeZMaFpkCCjkdvOiQ60fdmKts9JgGrM
3bAEDz3/imD69yuHUodGQi5Pl3FjIwHxEKkXb6/PqECBU6p9C2n7kSQuMDaPNMBVbCcC+CcWabv+53KHtdniO7T9LX5JL7zqMqmor2VG/B/UgTnzSNzRPcS+s+sQoJlhEPrnK14E2c++VXjX559qZ5H3pju4/iwJ
Uog1m2/2UHX01y/+xfdRnQCuiRXtLEpGUr+MRXol/7XSEDM732oO4zd+xiC2+Z4nk8Ue+Os= root@debiansio
root@debiansio:~/.ssh# _
```

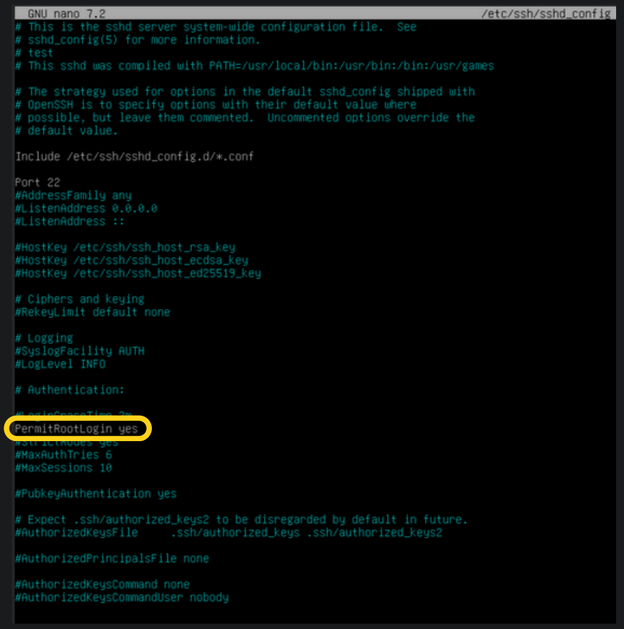
Le fichier id_rsa contient la clé privée. Cette clé doit rester confidentielle et ne jamais être partagée. Elle est utilisée pour déchiffrer les messages chiffrés avec la clé publique correspondante

Le fichier id_rsa.pub contient la clé publique. Cette clé peut être librement partagée et copiée sur d'autres systèmes. Elle est utilisée pour chiffrer les messages qui ne pourront être déchiffrés qu'avec la clé privée correspondante.
C'est cette clé qui se trouvera sur le serveur dans le fichier ~/.ssh/authorized_keys.

6. CONFIGURATION DU FICHIER SSH

Voici une parti du fichier ou on effectue des modifications :





`PermitRootLogin yes`

Cette commande nous permet de nous connecter en temps que utilisateur root, si nous ne voulons pas nous connecter en temps que root nous devons marquer no. La maîtrise de la permission pour root est importante pour la sécurité, car elle réduit les risques d'attaques par force brute et améliore la traçabilité.

`AllowGroups root ssh`

Autorise seulement certains membres de groupes à avoir accès via SSH à cette machine.

ALLOWUSERS NOM_UTILISATEUR1 ... NOM_UTILISATEURN
Autorise seulement certains utilisateurs à avoir accès via SSH à cette machine.

`PasswordAuthentication yes`

Cette commande signifie que l'authentification par mot de passe est autorisée pour les connexions SSH14. Cette option permet aux utilisateurs de se connecter au serveur SSH en utilisant leur nom d'utilisateur et leur mot de passe

`Banner /etc/ssh/banner`

Cette commande spécifie le chemin vers un fichier contenant un message d'accueil ou d'avertissement qui sera affiché aux utilisateurs avant qu'ils ne se connectent via SSH.

La différence entre PermitEmptyPasswords no et PermitRootLogin without-password est :
PermitEmptyPasswords no : Interdit les mots de passe vides pour tous les comptes.
PermitRootLogin without-password : Autorise la connexion en root uniquement par clé SSH, pas par mot de passe.

Pour la bannier nous nous rendons ensuite dans le fichier /etc/ssh/banner et nous écrivons se que nous voulons afficher.

7. CONNEXION

```
root@debiansio:~# ssh root@192.168.100.60
BIENVENUE SUR LE SERVEUR SSH
Linux serveurSSHalexandre 6.1.0-30-amd64 #1 SMP PREEMPT_DYNAMIC Debian 6.1.124-1 (2025-01-12) 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: Tue Mar  4 15:00:50 2025 from 192.168.100.197
root@serveurSSHalexandre:~#
```

```
root@debiansio:/# ssh user1@192.168.100.60
BIENVENUE SUR LE SERVEUR SSH
user1@192.168.100.60's password:
Linux serveurSSHalexandre 6.1.0-30-amd64 #1 SMP PREEMPT_DYNAMIC Debian 6.1.124-1 (2025-01-12) 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: Tue Mar  4 13:31:53 2025 from 192.168.100.197
user1@serveurSSHalexandre:~$ _
```

```
root@debiansio:~# ssh user2@192.168.100.60
BIENVENUE SUR LE SERVEUR SSH
user2@192.168.100.60's password:
Linux serveurSSHalexandre 6.1.0-30-amd64 #1 SMP PREEMPT_DYNAMIC Debian 6.1.124-1 (2025-01-12) 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.
user2@serveurSSHalexandre:~$ _
```

Ici, avec la commande ssh user1@192.168.100.60, nous arrivons à nous connecter car user1, root et user2 est dans le groupe d'accès SSH défini dans le fichier de configuration SSH sur le serveur.

```
root@debiansio:~# ssh user3@192.168.100.60
BIENVENUE SUR LE SERVEUR SSH
user3@192.168.100.60's password:
Permission denied, please try again.
user3@192.168.100.60's password:
```

Nous voyons ici qu'avec le user3, nous n'arrivons pas à nous connecter, et cela est normal vu que le user3 n'est pas dans un groupe autorisé.