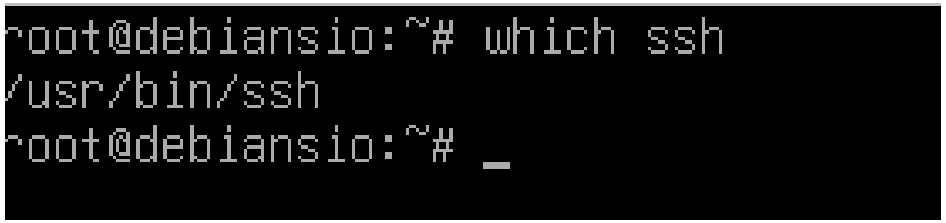
TP ADMIN A DISTANCE_SSH

ROMAIN LHEUREUX

La commande "which ssh" est utilisée pour afficher le chemin absolu de l'exécutable SSH sur votre système.



LA COMMANDE "WHICH SSH"

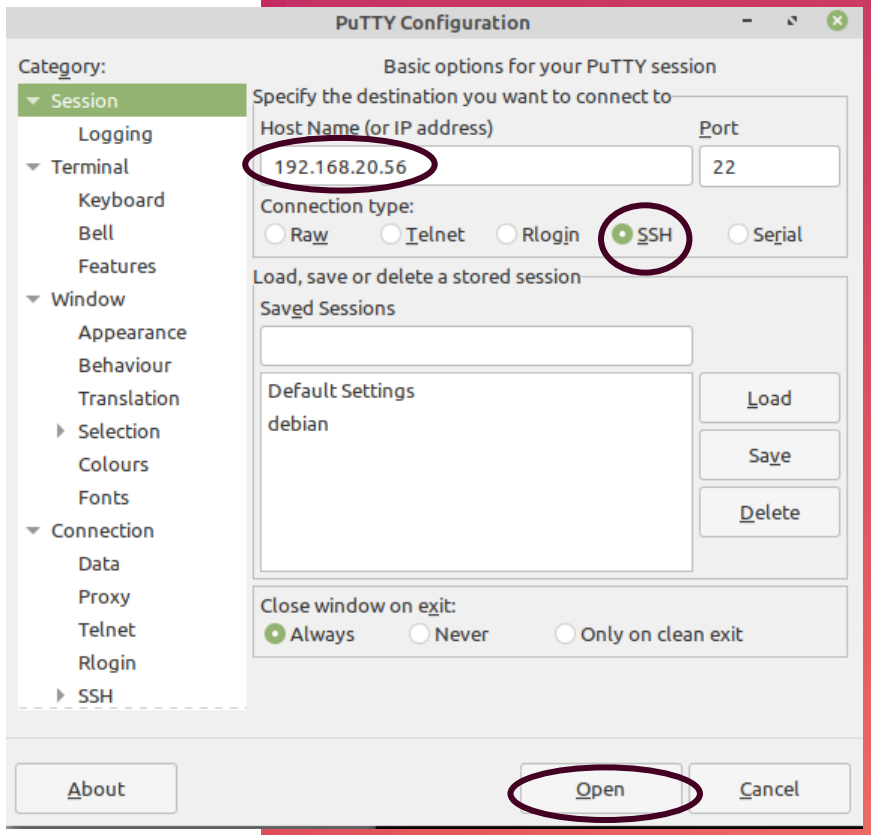
INSTALLER SSH

Commande:

« apt install openssh-server »

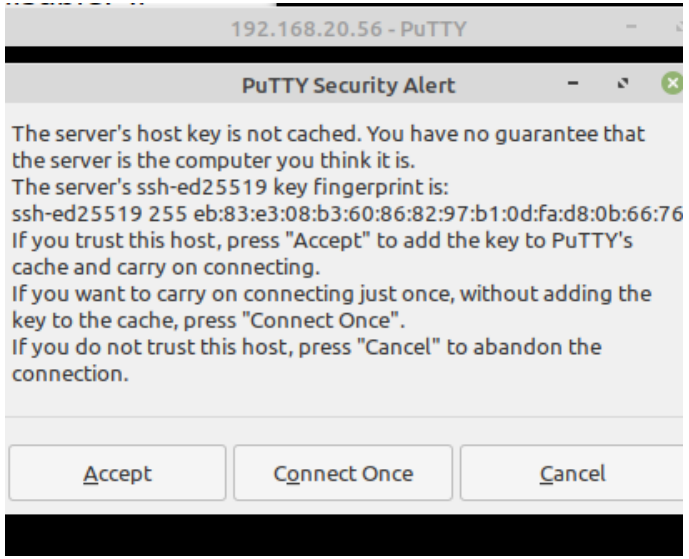
CONNECTION AU SSH

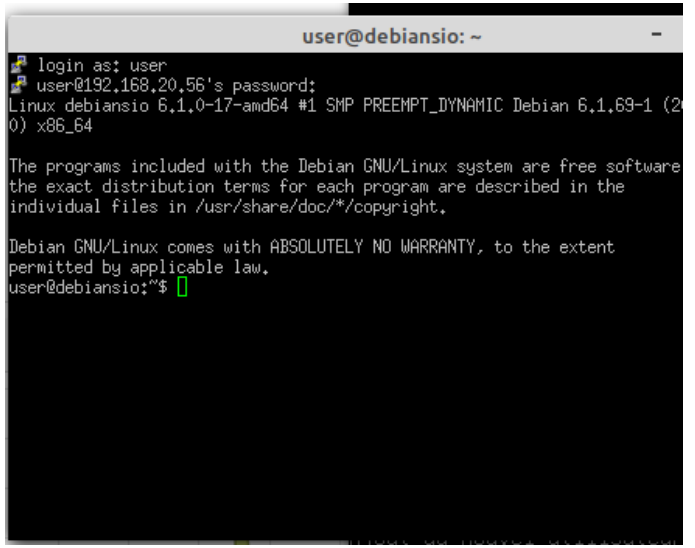
- Pour se connecter au ssh on peut utiliser PuTTY, il nous suffit de saisir l'adresse IP du serveur.



CONNECTION SSH

- Comme nous pouvons le voir, nous somme connecter au serveur sous l'utilisateur "user".

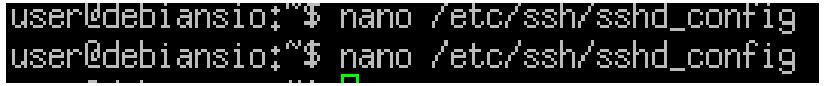


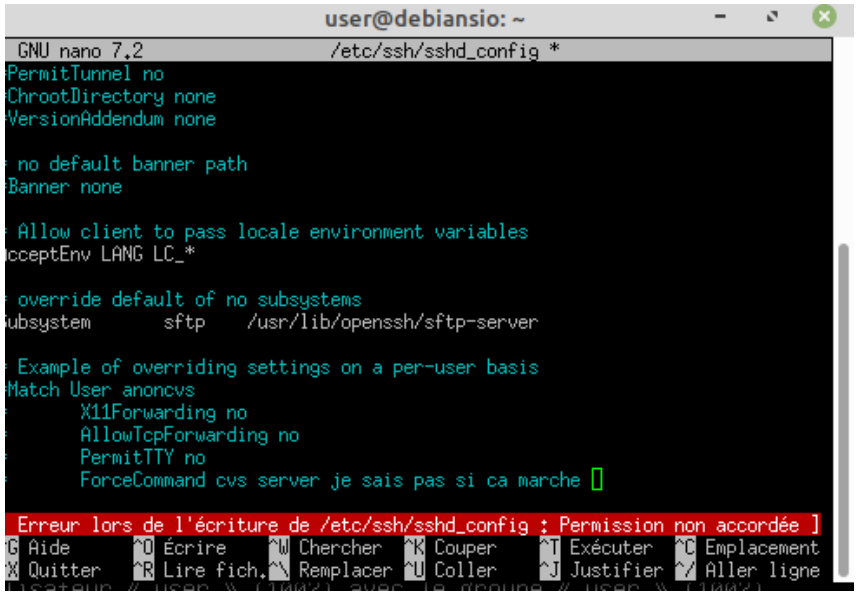


VÉRIFICATION DES DROITS

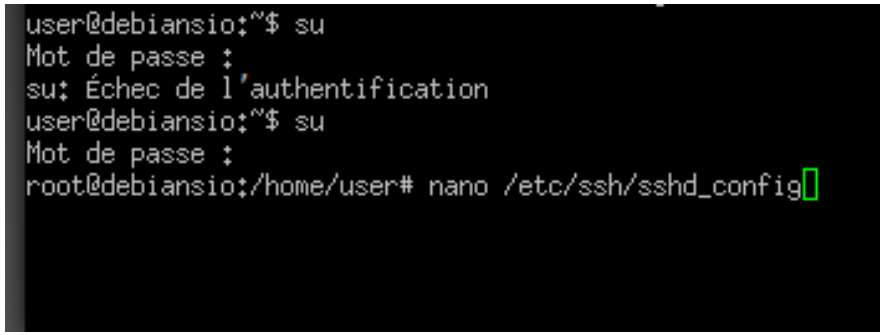
Commande : nano /etc/ssh/sshd_config
Avec cette commande nous pouvons accèdes a la configuration du ssh.

Comme vue plus tôt avec nous somme connecter sous l'utilisateur user. Et comme le montre le screen. Avec l'utilisateur user il n'est pas possible de modifier la configuration du ssh.

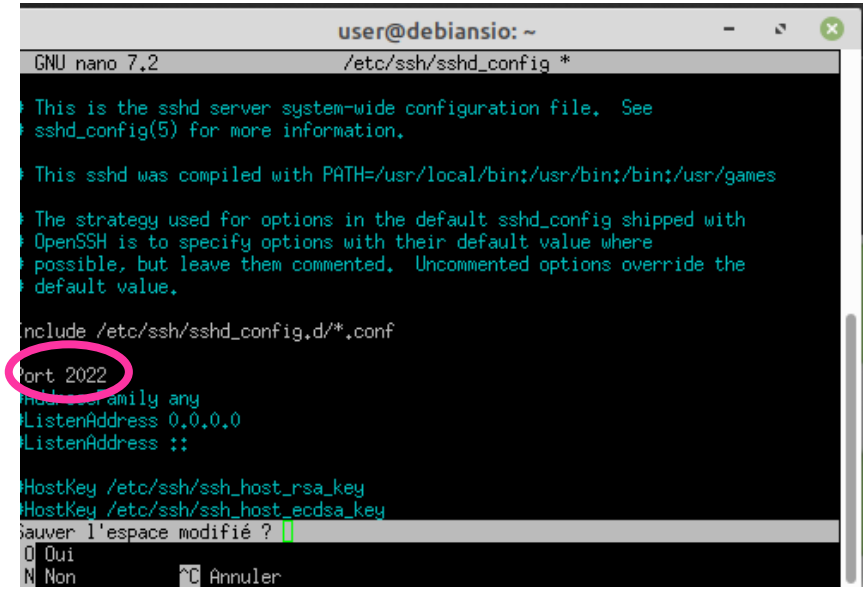




CONFIGURATION SSH



- Pour pouvoir modifier la configurations du ssh il nous faut passer en admin. Pour se faire nous allons utiliser la commande “SU”



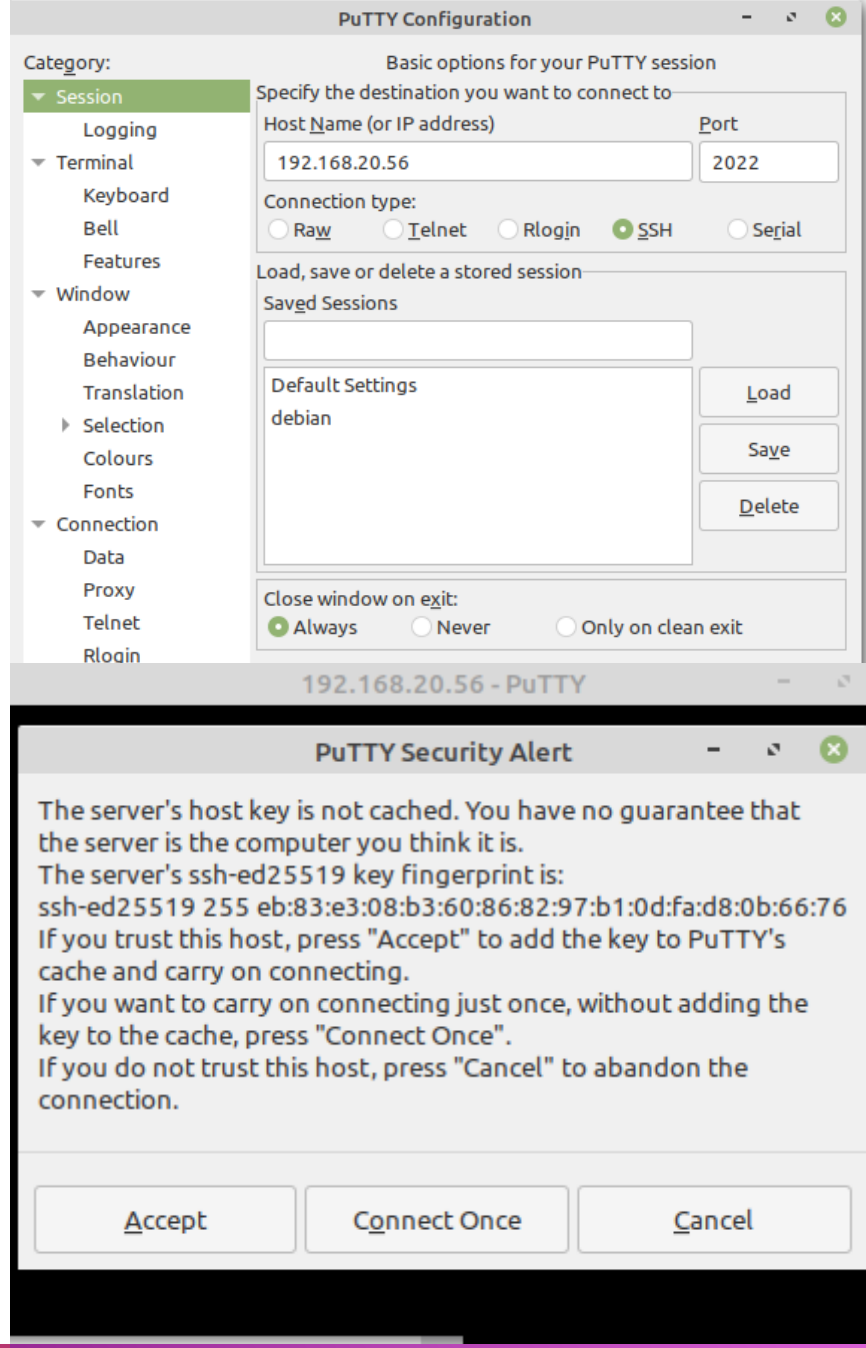
- Ensuite une fois dans le nano nous pouvons changer le port du serveur ce qui peut augmenter la sécurité en rendant notre serveur moins visible pour les attaquants qui ciblent généralement les ports standard.
- Pour se faire trouver dans le nano « Port » et enlever le # pour que la modifications soit pris en compte

CONFIGURATION SSH

- Pour appliquer les modifications il faut effectuer la commande « service ssh stop » puis « service ssh start ».

CONNECTION AU SERVEUR SSH

- Pour verifier que nos modifications sont bien pris en compte nous allons une nouvelle fois nous connecter avec le port modifier,
- Pour revenir au port 22 il nous suffit de retourner dans le nano et mettre un # devant port

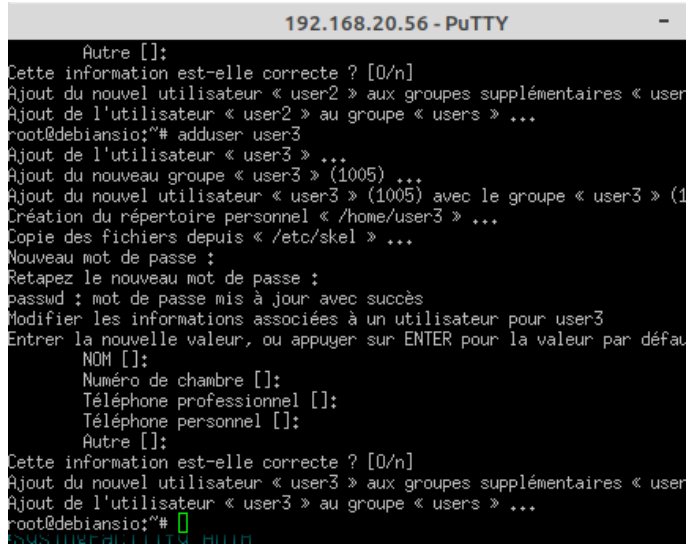


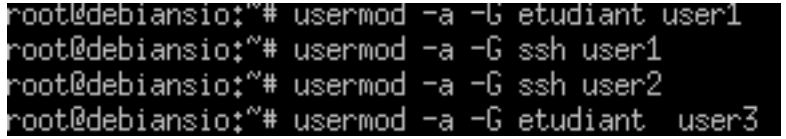
CONFIGURATION SSH

- Maintenant nous allons retouner dans le “nano” de notre serveur pour permettre de se connecter directement en root.
- Pour se faire il nous faut trouver “PermitRootLogin” , Une fois trouver il faut retirer le # et mettre yes.

UTILISATEUR SSH

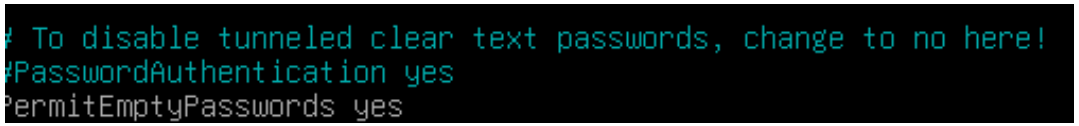
- Nous allons creer des groupes et des utilisateurs pour notre serveur ssh
- Commande:
- Creation d'utisateur "adduser"
- Creation de groupe "groupadd"
- Mettre des utilisatueur dans des groupes " usermod –a –G "nom du groupe" "nom de l'utilisateur".





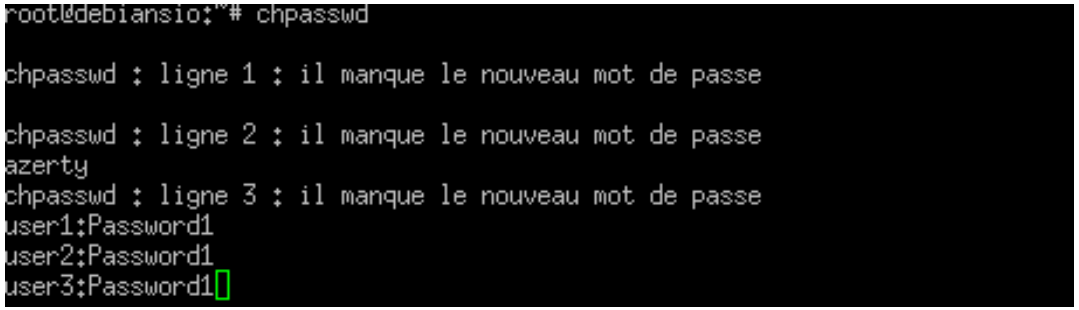
CONFIGURATION SSH

- Pour mettre les mots de passe vide il nous faut trouver “permitEmptyPasswords” puis changer le “no” par “yes”.



SERVEUR SSH

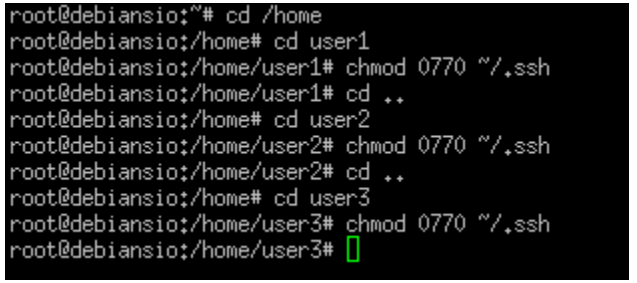
- la commande chpasswd permet de mettre à jour rapidement et efficacement les mots de passe des utilisateurs.



SERVEUR SSH

Pour chaque utilisateur nous allons créer un dossier .ssh grâce a la commande mkdir. Il va principalement stocker les clés d'authentification SSH et les configurations spécifiques à SSH pour cet utilisateur,

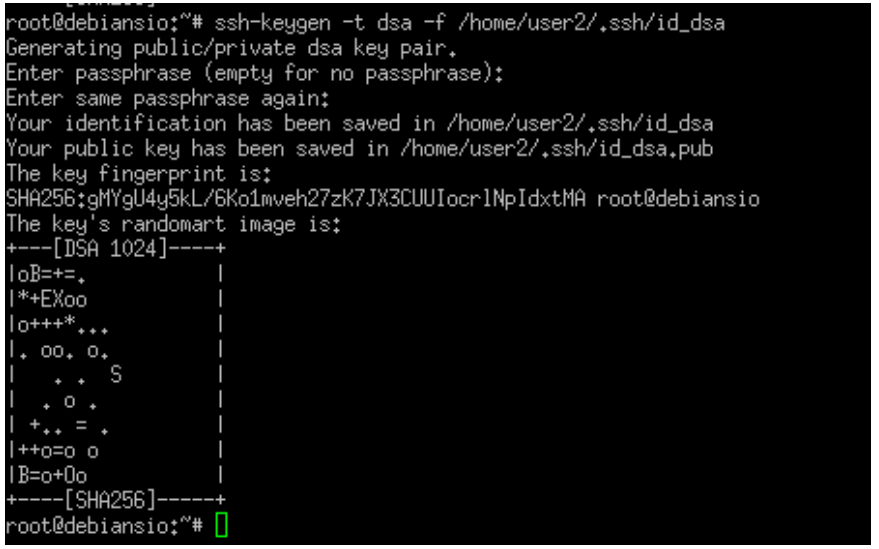
Pour renforcer la sécurité nous allons utiliser la commande « chmod 0770 ~/.ssh » ce qui a pour effet de limiter l'accès aux seuls propriétaires et membres du groupe, renforçant ainsi la protection des clés d'authentification SSH stockées à l'intérieur.

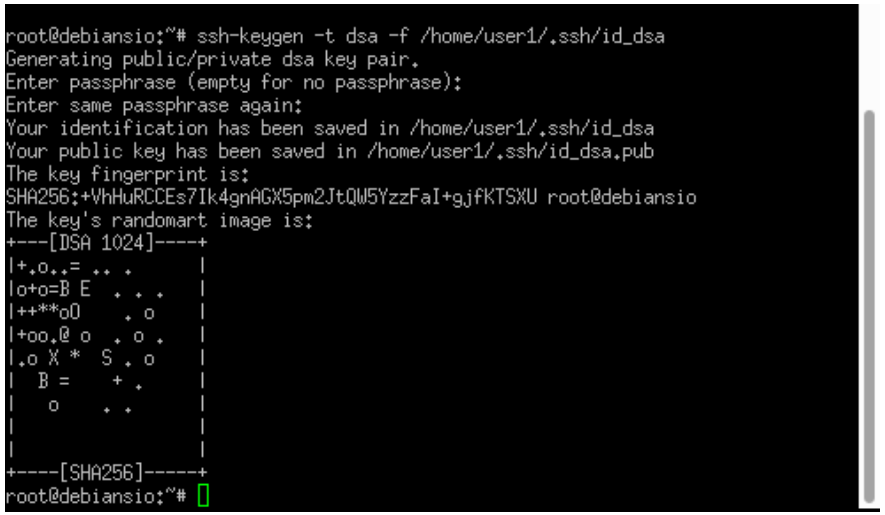


SERVEUR SSH

- Commande: ssh-keygen -t dsa -f ~/.ssh/id_dsa.
- Cette commande nous permet de générer des clés. Ce qui est crucial pour sécuriser cela permet une authentifications sécurisée

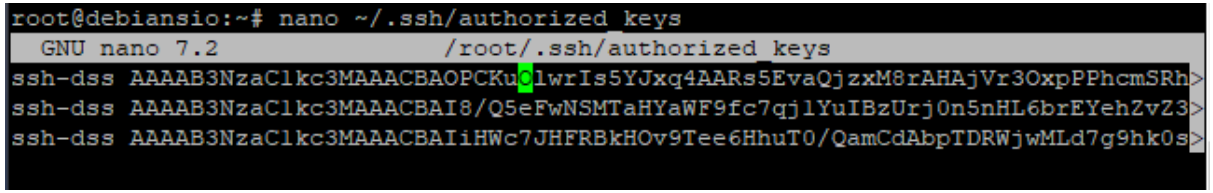
- Nous allons faire cette commande pour chaque utilisateur.
- Il est normal que ces deux fichiers soient différents, car l'un est une clé privée et l'autre est une clé publique. La clé privée est gardée secrète et n'est pas partagée, tandis que la clé publique est partagée avec les serveurs distants. même si la clé publique est divulguée, la clé privée reste secrète, ce qui permet de vérifier l'identité de l'utilisateur.



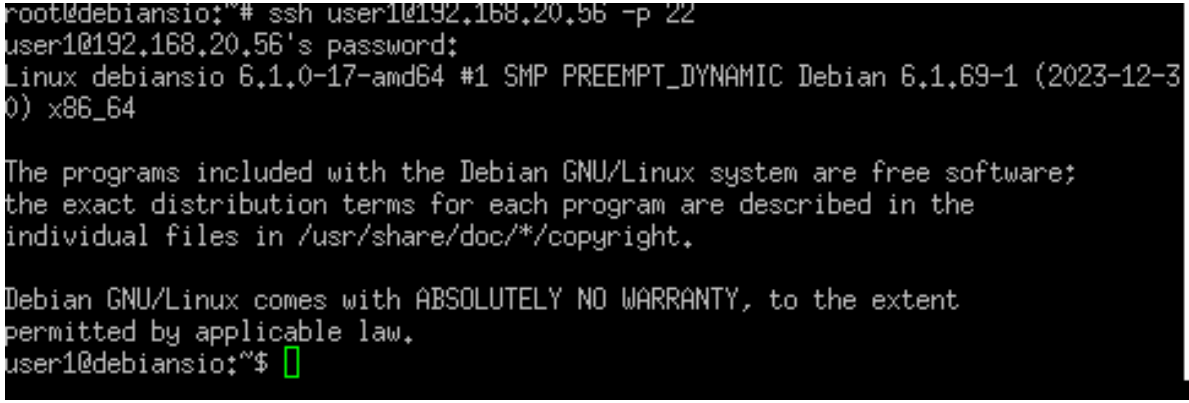


SERVEUR SSH

Avec la commande nano ~/.ssh/authorized_keys sert a voir que les clés publiques car Les clés privées ne sont pas comprises car leur présence compromettrait la sécurité.



SERVEUR SSH



- Maintenant nous allons voir si nous avons une connection en se connectant sur un utilisateur pour se faire nous allons utilizer la commande ssh user_name@adresseIPserveurSSH –p port

SERVEUR SSH

Maintenant nous allons configurer les accès au serveur ssh

pour se faire nous avons les commandes:

AllowUsers nom_utilisateur1 ect

AllowGroups nom du groupe1 ect

```
# Authentication:
AllowUsers user1 user2 user3
AllowGroups root ssh
```

TEST

```
sisr-6@sisr-6:~$ ssh user1@192.168.20.56
user1@192.168.20.56's password:
Linux debiansio 6.1.0-17-amd64 #1 SMP PREEMPT_DYNAMIC Deb
0) x86_64

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

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to th
permitted by applicable law.
Last login: Mon Feb  5 14:52:26 2024 from 192.168.20.56
user1@debiansio:~$
```

```
6@sisr-6:~$ ssh user3@192.168.20.56
@192.168.20.56's password:
 debiansio 6.1.0-17-amd64 #1 SMP PREEMPT_DYNAMI
6_64

rograms included with the Debian GNU/Linux syst
xact distribution terms for each program are de
idual files in /usr/share/doc/*/copyright.

n GNU/Linux comes with ABSOLUTELY NO WARRANTY,
tted by applicable law.
```