Partage de fichiers, SSH échange de clés, Création d'image SE

Systèmes Linux et Windows

Synthèse du document:

Présentation de la distribution Linux, TinyCore.

Le partage de fichier sera centré sur l'utilisation du service Samba avec son protocole SMB et du protocole NFS. L'objectif sera de mettre en place ces services sur un serveur et de s'y connecter grâce à une machine cliente.

La méthode de connexion SSH avec échange de clés sera centré sur l'installation du service, la mise en place de l'échange de clés. De plus, on testera la connectivité de ce protocole entre les systèmes d'exploitation Windows et Linux.

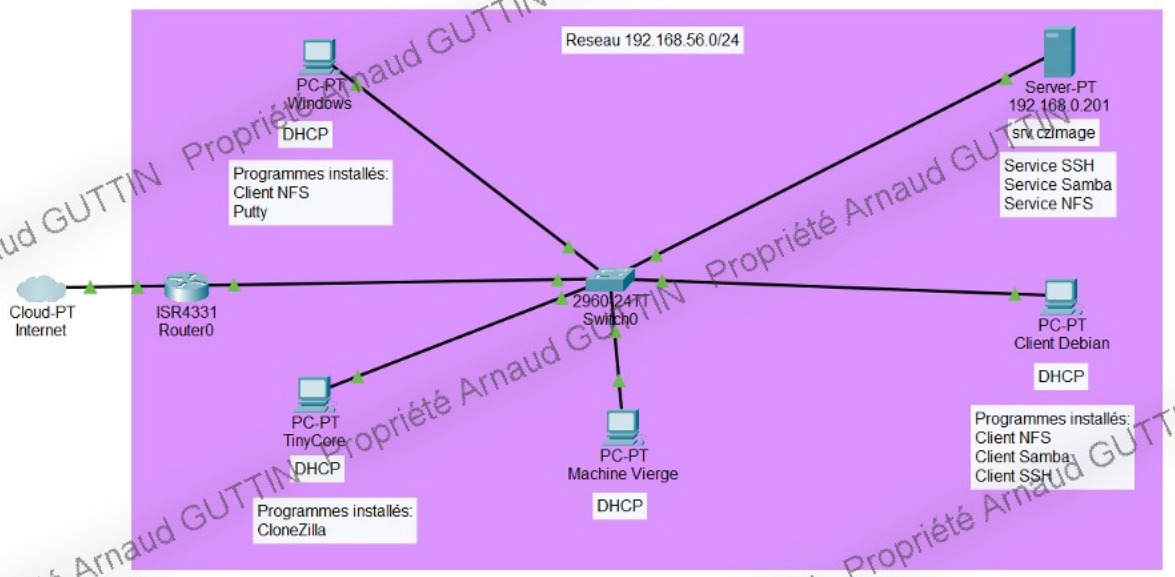
La création d'image de systèmes d'exploitation va porter sur l'exportation du système d'exploitation Linux. L'objectif est de créer une image d'une machine existante, de sauvegarder celle-ci sur un serveur de partage de fichier et de l'exporter sur une machine vierge.

Propriété Arnaud GUTTIN

Table des matières

Propriété Arnaud GUTTIN

Schéma réseau du laboratoire de test



Présentation de l'OS Tiny Core

Le système d'exploitation TinyCore est une distribution Linux. Il dispose d'une interface graphique simple. Ce système est très léger (environ 10Mo), et très stable.

Propriété Arnaud GUTTIN

Création d'un répertoire de partage

Pour créer un système de partage de fichier, il faut créer un répertoire où les utilisateurs distant pourront s'y connecter. Pour ce faire, il faut créer un répertoire (ici partimag), en utilisant la commande, `mkdir /home/partimag`, (voir ci-dessous). Cela signifie que le répertoire sera créé dans le répertoire home lui-même situé à la racine du système d'exploitation.

Il faudra ensuite gérer les droits de ce répertoire, afin de définir quelle personne a droit sur celui- ci. On utilise la commande chmod pour attribuer les droits à un fichier ou un répertoire. Pour notre activité on effectue la commande, `chmod 777 /home/partimag -R`.

Les trois chiffres suivant la commande chmod permettent de définir les droits. Ces droits sont séparés en trois groupes, le premier chiffre est le propriétaire du fichier/répertoire, le deuxième est le groupe propriétaire et le troisième est pour les autres utilisateurs.
Pour un droit de lecture on utilise le nombre 4, pour l'écriture le nombre 2 et pour l'exécution le nombre 1, on fera ensuite la somme de nos différents droits.
Le paramètres -R permet de donner la condition de droits récursif, c'est-à-dire que ces droits vont s'appliquer sur les éléments filles du répertoire.

Attention, la commande effectuée ci-dessous est dangereuse, elle signifie que tout le monde a un total contrôle sur le fichier, dans notre cas, il sera sécurisé directement depuis la configuration du service de partage de fichier.

```
root@czimage:~# mkdir /home/partimag
root@czimage:~# chmod 777 /home/partimag -R
```

Pour visualiser les droits des fichiers/répertoire sous le système Linux, on utilise la commande ls suivi du chemin avec le paramètre -la, (voir exemple ci-dessous).

```
root@czimage:~# ls /home/partimag/ -la
total 8
drwxrwxrwx 2 root root 4096 sept. 15 14:34 .
drwxr-xr-x 4 root root 4096 sept. 15 14:34 ..
root@czimage:~# ls /home/partimag -la
total 8
drwxrwxrwx 2 root root 4096 sept. 15 14:34 .
drwxr-xr-x 4 root root 4096 sept. 15 14:34 ..
root@czimage:~# _
```

Lors de notre activité nous aurons aussi besoin du répertoire image-xp et image-debian-srv, (voir ci-dessous).

```
root@czimage:~# mkdir /home/partimag/image-xp
root@czimage:~# mkdir /home/partimag/image-debian-srv
```

Mise en place de SSH avec échange de clés

Qu'est ce que le protocole SSH ?

Le protocole SSH, (Secure Shell) est un protocole de communication utilisé pour sécuriser les communications à distance sur des réseaux.
SSH certifie la confidentialité des données transmises car il utilise le chiffrement asymétrique grâce à des clés de chiffrement.
Il certifie aussi l'exactitude et l'intégrité des données car la connexion SSH est basée sur le protocole TCP qui s'assurera que les données sont bien transmises.
Il est utilisé pour l'accès à distance à des systèmes, la gestion de serveurs, le transfert de fichiers sécurisé, et d'autres.

Ce protocole est basé sur l'architecture client-serveur. Il faudra donc que la machine à distance soit équipée du service SSH et que la machine se connectant possède un client SSH.

Pourquoi utiliser SSH avec l'échange de clés?

La mise en place du service SSH avec échange de clé va permettre à la machine cliente de se connecter au serveur sans authentification.

Fonctionnement de la communication entre client et serveur

La communication entre le client et le serveur va se dérouler par les étapes suivantes:

- Le client envoie une demande de connexion SSH au serveur.
- Le serveur répond en indiquant les méthodes d'authentification prises en charge et les paramètres de cryptographie qu'il accepte.
- Le client et le serveur conviennent des paramètres, notamment les algorithmes de chiffrement, les méthodes d'authentification, les clés, etc.
- Le client aura précédemment généré une paire de clés : une clé privée et une clé publique. La clé privée est conservée en secret, tandis que la clé publique sera envoyée au serveur.
- Le serveur vérifie si la clé publique est autorisée à se connecter vérifiant sa configuration. Si la clé publique est autorisée, le serveur envoie une requête au client.
- Le client utilise sa clé privée pour signer la requête du serveur.
- Le serveur va vérifier la signature à l'aide de la clé publique du client. Si la vérification est réussie, le client est authentifié.

Lorsque le client est authentifié avec succès, une session SSH sécurisée est établie. Toutes les données échangées entre le client et le serveur seront chiffrées.

Installation du service SSH

Lors de cette activité nous allons installer le service SSH sur la machine nommée “czimage”.
Pour installer le service SSH sur une machine Debian, il faut entrer la commande `apt install openssh-server -y`, le paramètre -y permettra de valider par défaut les autorisations demandées, (voir ci-dessous).
Attention, vous devrez auparavant avoir realisé la mise à jour de la liste des paquets grâce à la commande `apt update -y`.

```
root@czimage:~# apt install openssh-server -y
```

Pour vérifier l'état du service ssh vous pouvez entrer la commande systemctl status ssh, (voir ci-dessous).

```
root@czimage:~# systemctl status ssh
● ssh.service - OpenBSD Secure Shell server
     Loaded: loaded (/lib/systemd/system/ssh.ser
     Active: active (running) since Mon 2023-10-
       Docs: man:sshd(8)
             man:sshd_config(5)
    Process: 415 ExecStartPre=/usr/sbin/sshd -t
   Main PID: 433 (sshd)
      Tasks: 1 (limit: 2359)
     Memory: 3.4M
     CGroup: /system.slice/ssh.service
             └─433 /usr/sbin/sshd -D
```

Génération de clés public et privé sous Linux

Nous allons générer une paire de clés sur notre machine cliente, dans notre activité la machine “buster”.

Pour ce faire, il faut entrer la commande `ssh-keygen`. Il vous sera ensuite demandé de valider le chemin d’enregistrement de vos clés en tapant la touche Entrer.
Il n’est pas nécessaire de mettre une phrase de passe pour la génération de clé , vous pouvez donc faire la touche Entrer jusqu'à la fin du formulaire, (voir ci-dessous).

Par défaut, l’algorithme de chiffrement sera le SHA256, pour utiliser un meilleur algorithme de chiffrement tel que le SHA512, vous pouvez entrer la commande `ssh-keygen -t rsa-sha2-256 -b 2048`.

```
root@buster:~# ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa):
Created directory '/root/.ssh'.
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:
SHA256:1fTVC9tA7lBVy2taI31y1Y2mepV9dEy6eX/y44bU2Js root@buster
The key's randomart image is:
+---[RSA 2048]----+
|             oo..*|
|            oo+.*=|
|           ....XoO|
|          .  o+.O+|
|         S  .oX*o*|
|            . +=*=|
|           . o.o =|
|            . . E.|
|               o.o|
+----[SHA256]-----+
```

Pour visionner votre clé de chiffrement et sa longueur, vous pouvez entrer la commande, `ssh-keygen -lf .ssh/id_rsa`, (voir ci-dessous).

```
root@buster:~# ssh-keygen -lf .ssh/id_rsa
2048 SHA256:1fTVC9tA7lBVy2taI31y1Y2mepV9dEy6eX/y44bU2Js root@buster (RSA)
```

Nous allons maintenant copier la clé publique de la machine cliente sur le serveur SSH en utilisant la commande ssh-copy-id -i sio@192.168.56.201. Il sera nécessaire de préciser un utilisateur local du serveur (czimage) comme sio, puis d'ajouter l'adresse IP du serveur, (voir ci-dessous).

```
root@buster:~# ssh-copy-id -i sio@192.168.56.201
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/root/.ssh/id_rsa.pub"
The authenticity of host '192.168.56.201 (192.168.56.201)' can't be established.
ECDSA key fingerprint is SHA256:q12VvzAD7XjDEz4c8MktSFV5zYO7fBXrk1OazxW8AJ4.
Are you sure you want to continue connecting (yes/no)? yes
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out an
eady installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted now
all the new keys
sio@192.168.56.201's password:

Number of key(s) added: 1

Now try logging into the machine, with:   "ssh 'sio@192.168.56.201'"
and check to make sure that only the key(s) you wanted were added.
```

Connection avec un client SSH sous Linux

Vous pouvez maintenant vous connecter au serveur distant (czimage), grâce à la commande `ssh sio@192.168.56.201`, (voir ci-dessous).
Vous n'avez pas besoin d'entrer le mot de passe car le serveur SSH à reconnu la signature avec la clé public de la machine cliente.

```
root@buster:~# ssh sio@192.168.56.201
Linux czimage 4.19.0-6-amd64 #1 SMP Debian 4.19.67-2+deb10u2 (2019-11-11) 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 Jan  7 14:13:42 2020
sio@czimage:~$ _
```

Pour sortir de la connexion SSH vous pouvez entrer la commande `exit`.

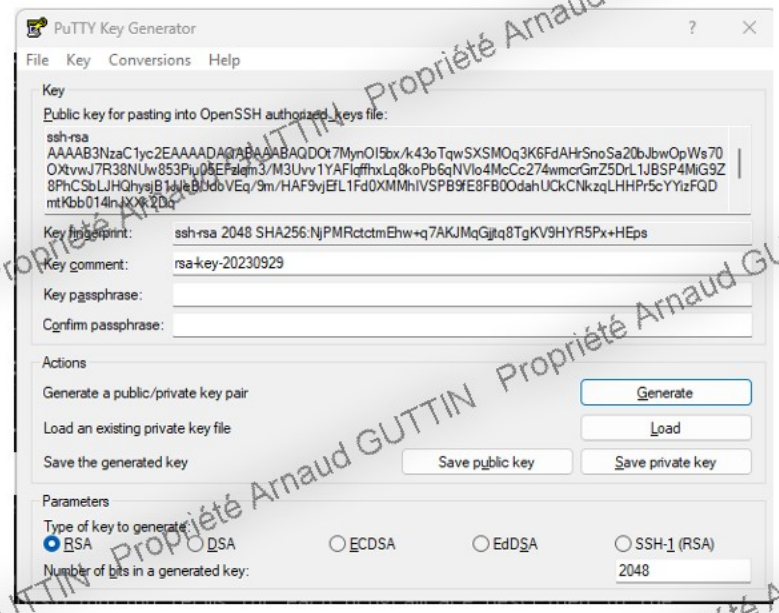
Génération de clés public et privé sous Windows

Pour vous connecter à un serveur SSH sur une machine Windows sans mot de passe, vous aurez besoin de générer une paire de clés. Pour ce faire, vous pouvez utiliser un générateur de clé SSH tel que Putty Key Generator.

Une fois le logiciel ouvert vous pouvez cliquer sur Generate, il sera nécessaire de faire bouger votre souris durant un certain temps, ce sont les coordonnées de votre souris qui permettront de générer une clé totalement aléatoire et unique, (voir ci-dessous).

Vous pouvez sélectionner la taille et le type de chiffrement de vos clés pour une meilleur sécurité dans l'onglets paramètres.

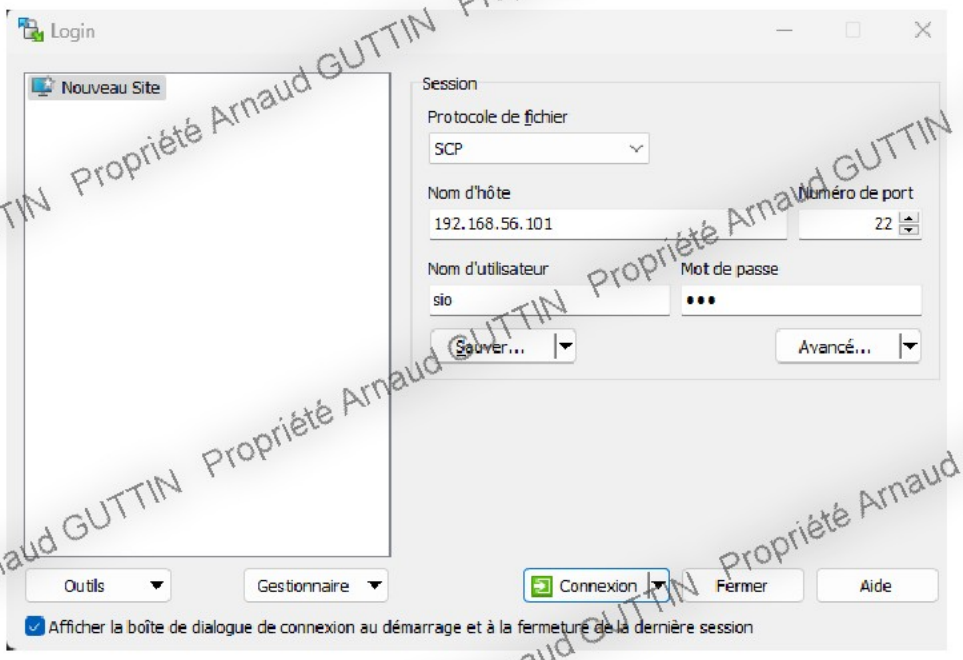
Propriété Arnaud GUTTIN



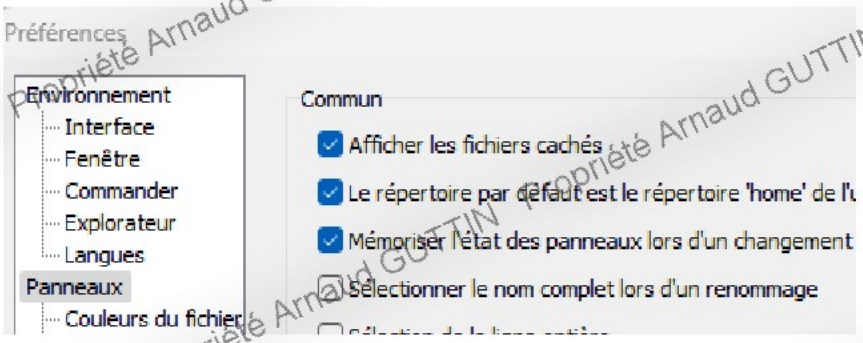
Transfère de la clé publique au serveur SSH avec Windows

Pour transférer la clé publique de la machine Windows sur le serveur SSH (czimage), vous aurez besoin du logiciel WinSCP afin de modifier le fichier de configuration /home/sio/.ssh/authorized_keys.

Pour ce faire, il faudra ouvrir WinSCP. Ensuite, vous devrez entrer l'adresse IP du serveur SSH (czimage), préciser le port (22 pour SCP, basé sur SSH), puis les identifiants et mot de passe d'un utilisateur local du serveur, (voir ci-dessous).

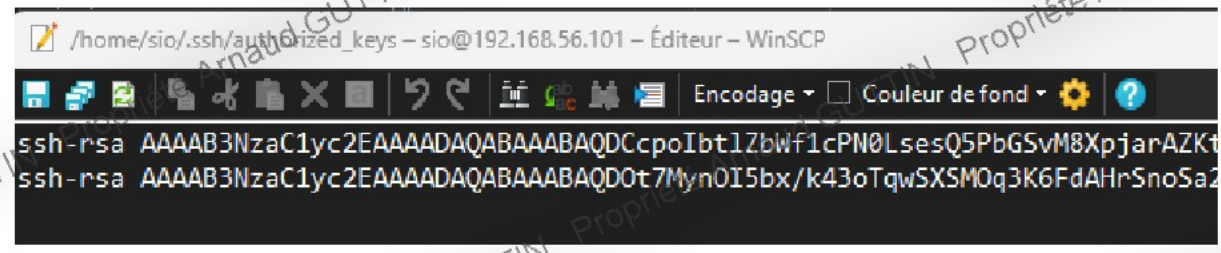


La clé publique étant un fichier caché (nommé avec un point devant) il sera nécessaire d'aller dans Préférences → Panneaux, et de cocher la case Afficher les fichiers cachés, (voir ci-dessous). Vous pourrez donc voir votre fichier dans le système de fichier.



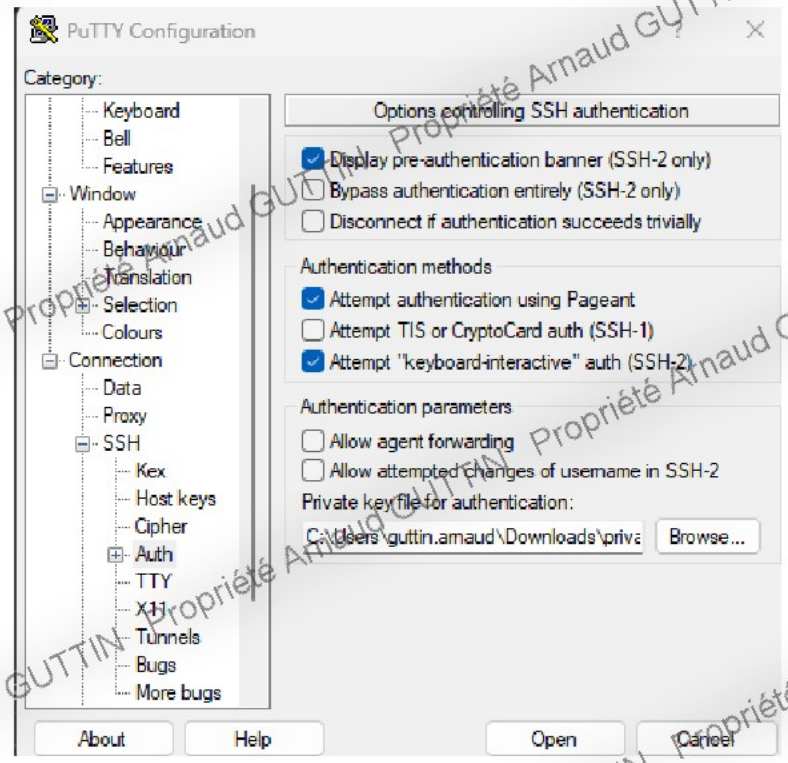
Une fois la connexion établie, vous devrez vous rendre dans le répertoire /home/sio/.ssh/. Puis modifiez le fichier "authorized_keys" et ajouter votre clé public en ajoutant la mention "ssh-rsa" devant, (veillez à ce que votre clé public soit sur une seule ligne), (voir ci-dessous).

Attention, si vous souhaitez vous connecter avec un autre utilisateur, il sera nécessaire de mettre aussi la clé SSH dans son répertoire personnel.

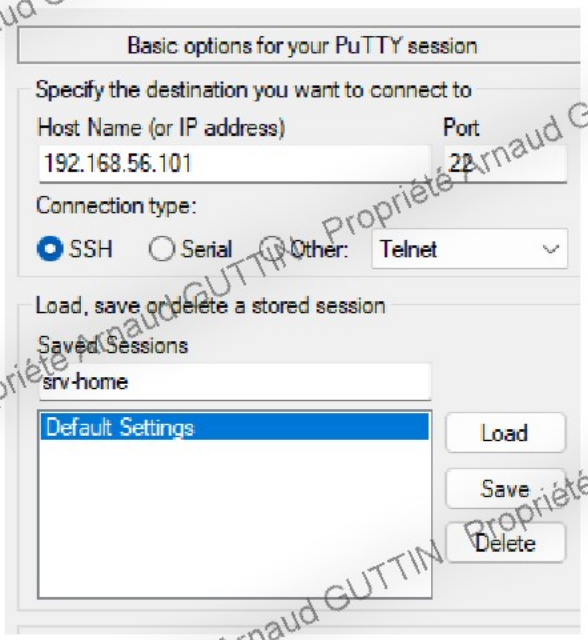


Connexion avec le client Windows

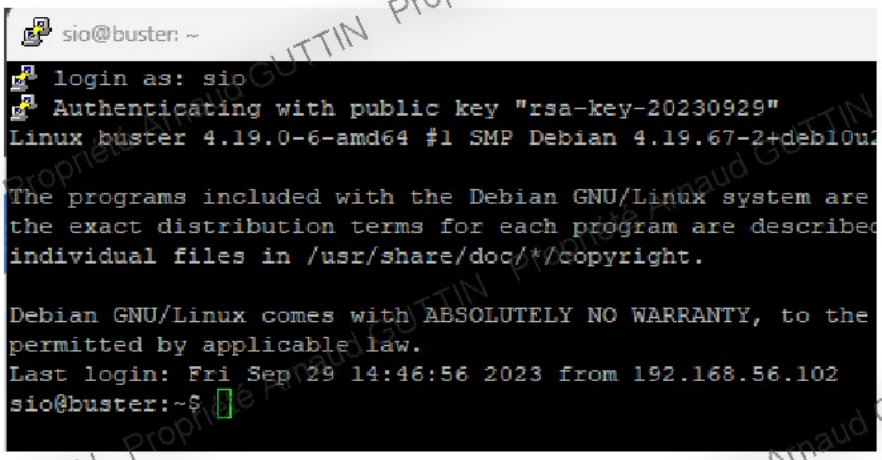
Pour vous connecter au serveur SSH , vous aurez besoin d'un client SSH tel que Putty.
Une fois le programme ouvert, dans le menu déroulant à gauche, allez dans Connection→SSH→Auth, puis déposez votre clé privée.
Putty pourra l'utiliser pour chiffrer la connexion.



Une fois de retour sur l'onglet par défaut de Putty, vous devrez entrer l'adresse IP du serveur SSH (czimage), puis préciser le port utilisé par le protocole SSH (port 22), (voir ci-dessous).
Pour des questions de confort, vous avez la possibilité de sauvegarder votre configuration en la nommant et en cliquant sur Save.



Une fois cliquer sur Connection, une fenêtre de Terminal s'ouvrira. Il faudra préciser l'utilisateur sur lequel vous souhaitez vous connecter, il ne sera pas nécessaire de mettre le mot de passe car le serveur SSH reconnaîtra votre signature grâce à la clé publique transmise précédemment.



Service Samba

Qu'est ce que Samba ?

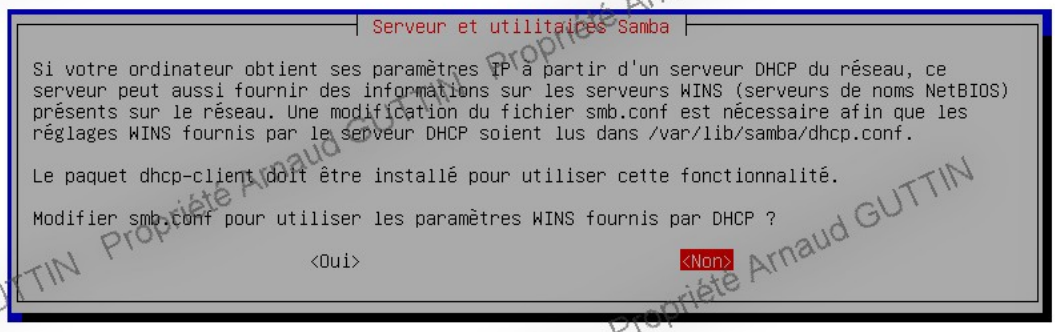
Samba est un service. Il permet de partager des fichiers, des répertoires et d'autres ressources réseau (imprimantes, routeurs…). Il fonctionne avec le protocole SMB (Server Message Block) propriétaire de Microsoft, grâce à celui-ci, nous pouvons permettre l'interopérabilité entre les systèmes. C'est à dire qu'on pourra partager le répertoire Linux et les machines Windows ou MacOS pourront s'y connecter même si ce n'est pas le même formatage.

Installation de Samba

Pour cette activité, nous effectuons l'installation sur la machine czimage.

Pour installer le service de partage sous Linux, vous aurez besoin d'entrer la commande `apt update -y`, afin de mettre à jour les paquets. Vous pouvez ensuite entrer la commande `apt install samba -y`.

Une fois l'installation débutée, il vous sera demandé si vous souhaitez ajouter les paramètres pour Wins, cliquez sur Non, (voir ci-dessous).



Configuration de Samba

Lors de l'installation terminée, vous devrez modifier le fichier de configuration grâce à la commande `nano /etc/samba/smb.conf`. C'est dans ce fichier qu'il faudra ajouter un répertoire de partage et configurer la sécurité d'accès à celui-ci.

Pour des droits d'accès au partage nommé partimag, au chemin /home/partimage, avec un accès invité et une lecture seul, il faudra entrer la configuration ci-dessous.

```
[partimag]
        path = /home/partimag
        guest ok = yes
        read only = yes
```

Lorsque la modification du fichier est terminée, il faudra valider la configuration effectuée. Pour ce faire, effectuez la commande `testparm`.

Dans notre activité, il sera nécessaire d'avoir des droits d'accès en écriture sur le répertoire /home/partimag. Vous devrez donc effectuer la configuration ci-dessous pour avoir un accès en écriture.

- Le paramètre "read only" devra être modifié à la valeur "no".
- Le paramètre "writable" devra être à "yes"
- Le paramètre "valid users" permet d'autoriser uniquement certains utilisateurs locaux, ici nous mettrons l'utilisateur smbuser

```
[partimag]
        path = /home/partimag
        guest ok = yes
        read only = no
        writable = yes
        valid users = smbuser
```

Vous pouvez ensuite valider la configuration avec `testparm`.

Connection au serveur Samba sous Linux

Pour vous connecter au partage réseau Linux, il faudra télécharger le client Samba grâce à la commande `apt install smbclient- y`, (voir ci-dessous), (cette section est effectuée sur la machine Debian Buster).

```
root@buster:~# apt update -y && apt install smbclient
```

Une fois l'installation terminée, vous pouvez entrer la commande `smbclient //192.168.56.201/partimag -U smbuser`. Enfin, entrez le mot de passe pour vous authentifier. Une fois sur la console vous pouvez effectuer différentes commandes, comme `ls` pour visualiser les fichiers du répertoire, (voir ci-dessous). Pour sortir de la console, entrez la commande `exit`.

Attention, vous devrez ajuster les paramètres d'adresse IP, de nom de partage et le nom d'utilisateur en fonction de votre configuration.

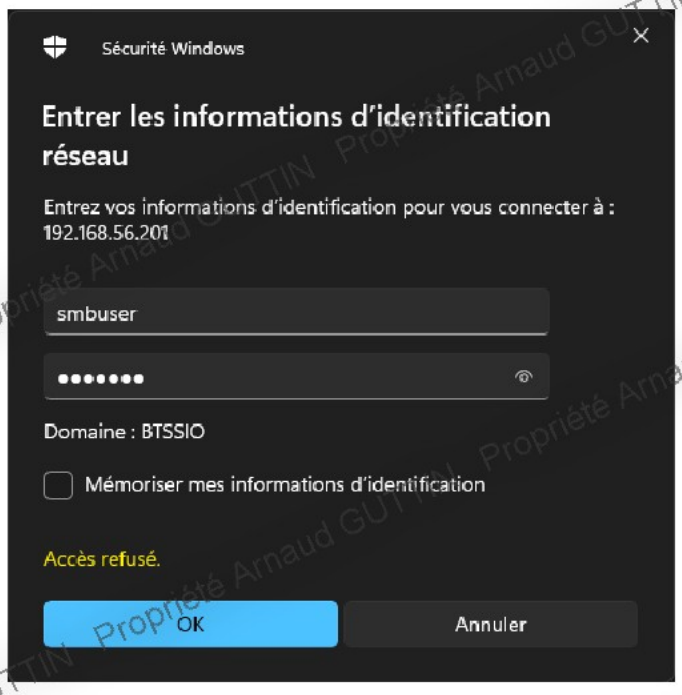
```
root@buster:~# smbclient //192.168.56.201/partimag -U smbuser
Enter WORKGROUP\smbuser's password:
Try "help" to get a list of possible commands.
smb: \> ls
  .                                   D        0  Fri Sep 15 15:20:03 2023
  ..                                  D        0  Fri Sep 22 14:13:00 2023
  TEST.txt                            N        0  Fri Sep 15 15:20:03 2023
  image-debian-srv                    D        0  Fri Sep 15 14:37:19 2023
  image-xp                            D        0  Fri Sep 15 14:36:59 2023

                7158264 blocks of size 1024. 5523568 blocks available
```

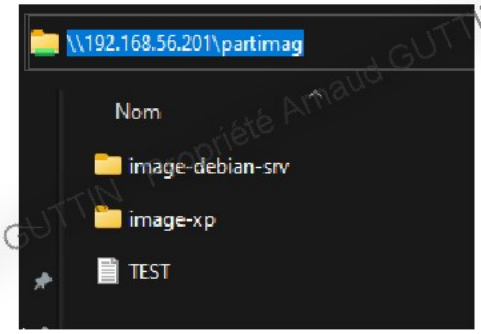
Connection au serveur Samba sous Windows

Pour la connexion au partage réseau Samba sur une machine Windows vous devrez vous rendre la l'Explorateur de fichier. Dans la barre de chemin veuillez mettre \\192.168.56.201\partimag, vous pourrez ensuite entrer le nom d'utilisateur et le mot de passe, (voir ci-dessous).

Propriété Arnaud GUTTIN



Une fois connecté, vous pouvez visualiser vos fichiers et répertoires.



Protocole NFS

Qu'est ce que le protocole NFS ?

Le protocole NFS, (Network File System) est un protocole de partage de fichiers et répertoires. Grâce au service NFS, il sera possible de monter un volume sur une machine comme si celui-ci était réellement connecté à la machine.

Installation et configuration du service NFS

Pour installer le service NFS (installation sur la machine czimage), il faudra entrer la commande, `apt install nfs-kernel-server -y`, si vous n'avez pas mis à jour la liste des paquets, il faudra auparavant entrez la commande `apt update -y` (voir ci-dessous).

```
root@czimage:~# apt install nfs-kernel-server -y
```

Configuration du service NFS

Vous devez modifier la configuration du service NFS pour ajouter le répertoire /home/partimag. Pour ce faire, vous devrez modifier le fichier de configuration grâce à la commande, `nano /etc/exports`.

Vous devrez ensuite ajouter la ligne (non commenté) ci-dessous.

```
# /etc/exports: the access control list for filesystems whi
#               to NFS clients.  See exports(5).
#
# Example for NFSv2 and NFSv3:
# /srv/homes       hostname1(rw,sync,no_subtree_check) host
#
# Example for NFSv4:
# /srv/nfs4        gss/krb5i(rw,sync,fsid=0,crossmnt,no_sub
# /srv/nfs4/homes  gss/krb5i(rw,sync,no_subtree_check)
#
/home/partimag  192.168.56.0/255.255.255.0(ro,all_squash)
```

La ligne ci-dessous va permettre de rendre visible le partage sur toute la plage réseau 192.168.50.0/24.

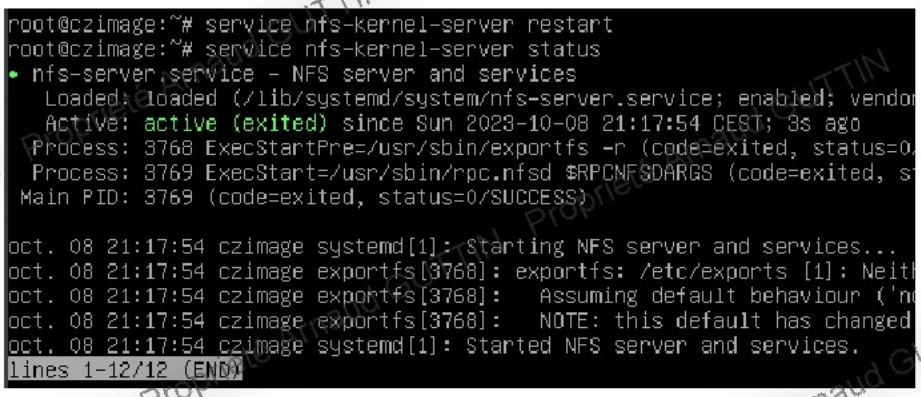
Le paramètre ro signifie que le système de fichiers exporté est accessible en lecture seule. Les hôtes distants ne peuvent pas modifier les données partagées sur le système de fichiers. Pour autoriser des hôtes à effectuer des modifications sur le système de fichiers (par exemple, lecture/écriture), veuillez spécifier l'option rw.

Le paramètre sync signifie que le serveur NFS ne répondra pas aux requêtes effectuées avant que les changements demandés par les requêtes précédentes soient écrits sur disque.

Le paramètre all_squash signifie que le serveur va forcer le mapping (le montage du disque) de tous les utilisateurs vers un utilisateur anonyme

Attention, par défaut, le serveur NFS retardera l'écriture sur disque s'il “suspecte” qu'une autre requête d'écriture va arriver. Cela peut améliorer les performances. Pour désactiver cette option il faudra entrer le paramètre no delay.

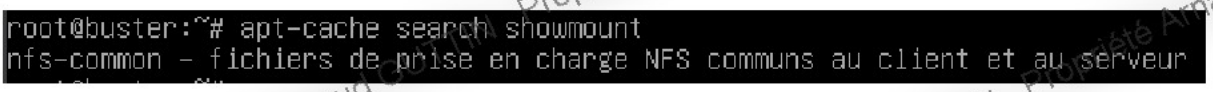
Vous valider la configuration du serveur vous pouvez entrer la commande, `systemctl restart nfs-kernel-server`. Pour vérifier si le service fonctionne correctement, vous pouvez entrer la commande `systemctl status nfs-kernel-server` (voir ci-dessous).



Montage du volume sous Linux

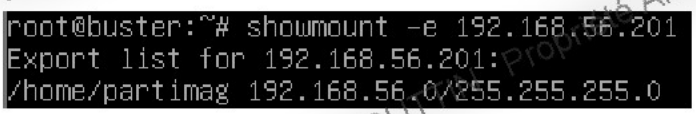
Pour monter le volume sur une machine Linux (configuration effectuée sur la machine Debian Buster), il faudra utiliser la commande showmount.

Attention, cette commande n'est pas installée par défaut, vous ne pouvez entrer directement la commande d'installation de showmount. Il est nécessaire de localiser dans quel paquet elle se situe. Pour ce faire, entrez la commande `apt-cache search showmount`, (voir ci-dessous).



La commande étant située dans le paquet nfs-common, vous pouvez maintenant entrer la commande, `apt install nfs-common -y`.

Il faudra ensuite vérifier si le montage de volume est bien disponible en entrant la commande, showmount -e192.168.56.201, (voir ci-dessous).



Pour monter le volume de partage, il faudra créer un répertoire ou vous pourrez retrouver ce volume, ici nous allons créer le répertoire /media/NFS grâce à la commande `mkdir /media/NFS`.
Nous pouvons ensuite monter le volume de partage grace a la commande, `mount 192.168.56.201:/home/partimag /media/NFS`, (voir ci-dessous).

```
root@buster:~# mkdir /media/NFS
root@buster:~# mount 192.168.56.201:/home/partimag /media/NFS
root@buster:~# ls -la /media/NFS
total 16
drwxrwxrwx 4 root root 4096 oct.   8 20:39 .
drwxr-xr-x 4 root root 4096 oct.   8 21:24 ..
drwxr-xr-x 2 root root 4096 oct.   8 20:39 image-debian-srv
drwxr-xr-x 2 root root 4096 oct.   8 20:38 image-xp
```

Système CloneZilla

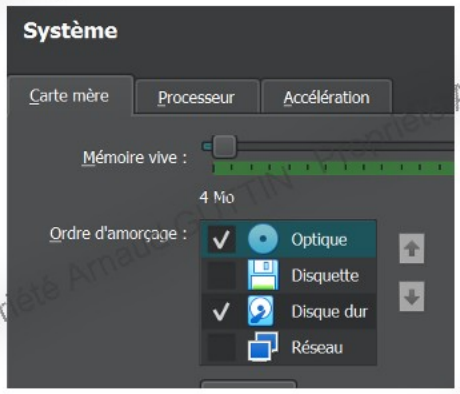
Qu'est ce que CloneZilla ?

CloneZilla est un logiciel de restauration de sauvegarde, et de clonage de système d'exploitation.
Il sera utile lorsqu'on souhaite configurer un système d'exploitation et que nous souhaitons diffuser cette configuration sur plusieurs machines, (ce qui évite la réciprocité de la tâche, et donc d'automatiser celle-ci).

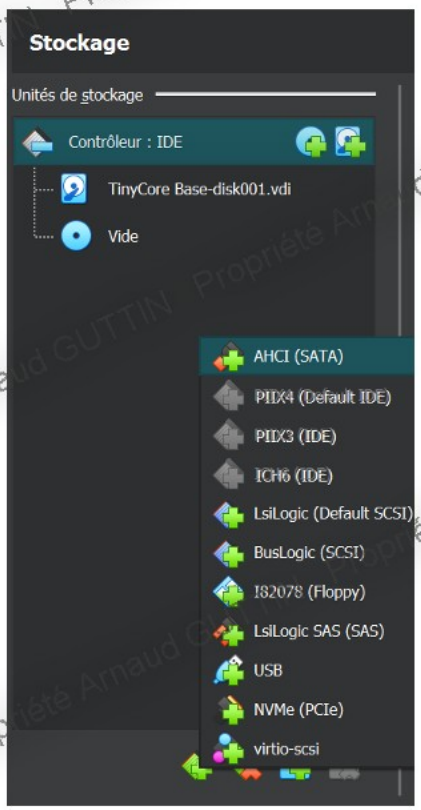
Mise en place du système CloneZilla

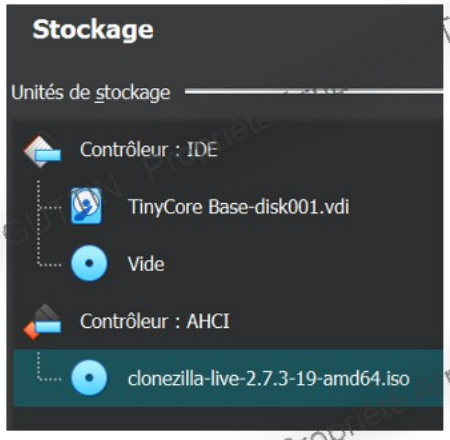
CloneZilla est un système d'exploitation sous forme d'image iso, il faudra faire booter la machine sur cette image.
Dans notre activité, nous utilisons l'hyperviseur VirtualBox, il est tout à fait possible de réaliser ces opérations avec de vraies machines.

Pour que la machine boot sur l'image iso de CloneZilla, il faudra modifier l'ordre de boot dans VirtualBox, (voir ci-dessous).

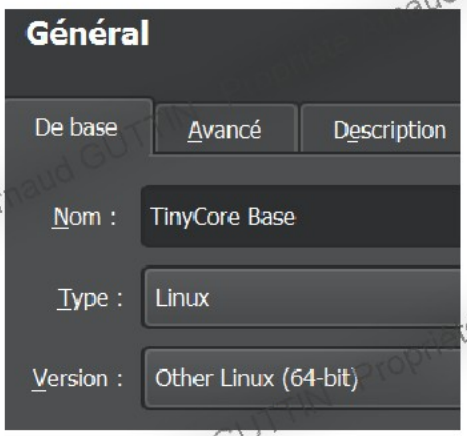


Dans les configurations de stockage, pour VirtualBox, il faudra ajouter un contrôleur SATA (non présent par défaut), cela permettra de d'ajouter l'image, (voir ci-dessous).





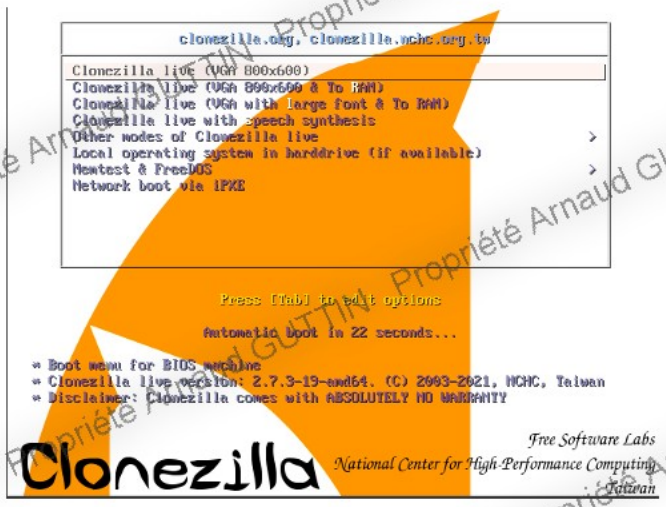
Attention, veillez à ce que votre machine soit capable de gérer les systèmes 64 bits car CloneZilla est un système 64 bits. Si vous êtes sur un environnement de virtualisation vous pouvez modifier ce paramètre, (voir ci-dessous).



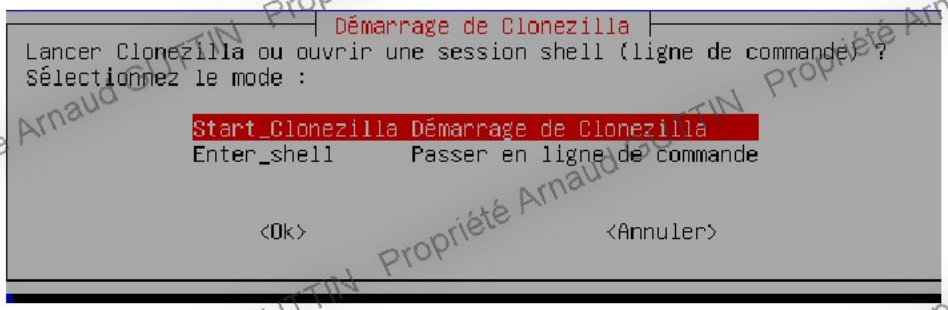
Création de l'image de la machine hôte

Une fois votre machine démarrer et CloneZilla démarrer vous ferez face à l'interface d'accueil CloneZilla, vous pouvez taper la touche Entrer.

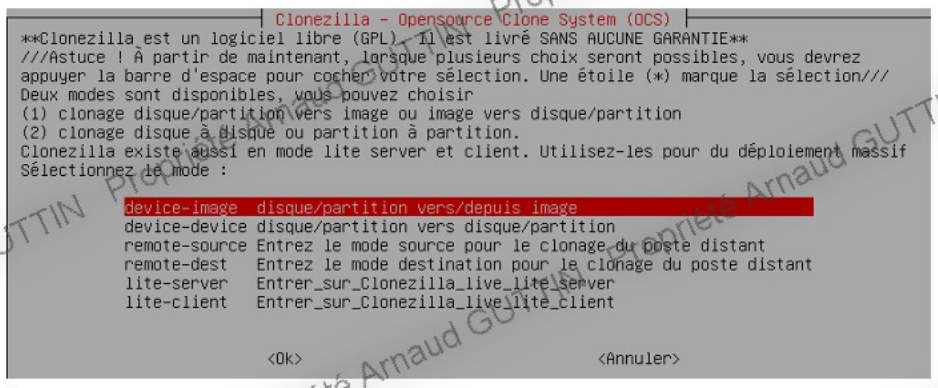
Propriété Arnaud GUTTIN



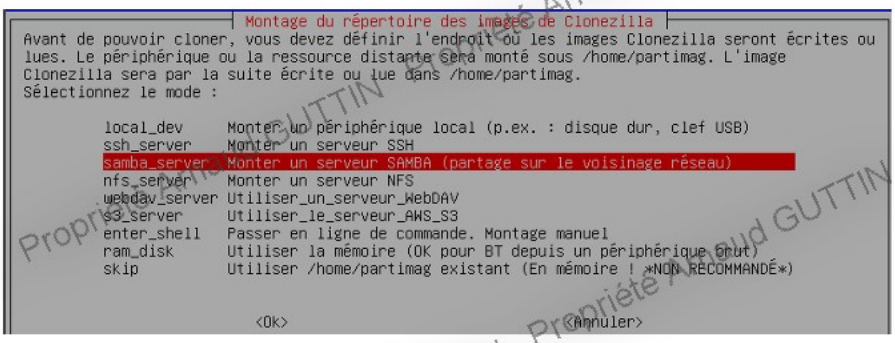
Après diverses questions basiques de configuration tels que la langue du clavier et du système, il vous sera demandé de démarrer la configuration de CloneZilla.



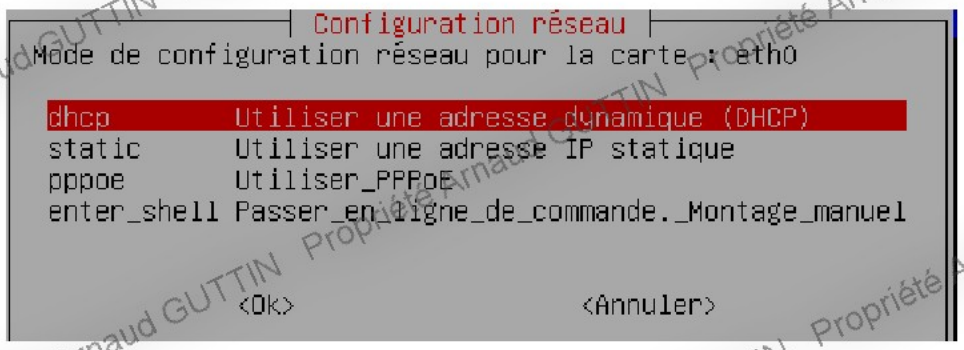
Pour exporter une image il y a deux possibilités, d'exporter celle-ci sur une image iso, la deuxième est de directement positionner l'image sur un disque dur ou une partition présente sur la machine. Pour notre activité, nous allons sélectionner device-image.



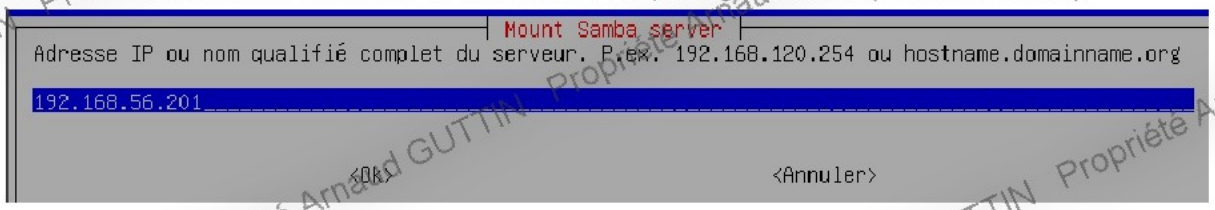
Vous pouvez choisir le mode d'exportation de l'image créé, nous allons choisir le mode samba-server, (voir ci-dessous).



CloneZilla aura besoin d'une configuration IP durant la communication avec le serveur de partage Samba, vous pouvez sélectionner DHCP pour avoir une adresse IP automatiquement, ou static pour configurer la vôtre.

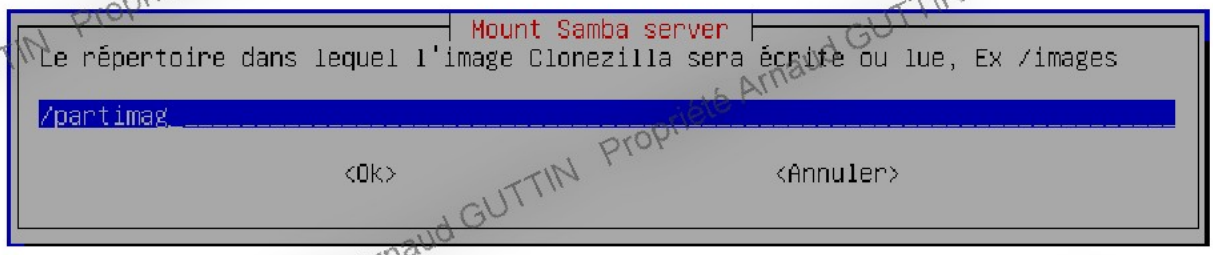


Il faudra ensuite entrer l'adresse IP du serveur de partage Samba, (voir ci-dessous).

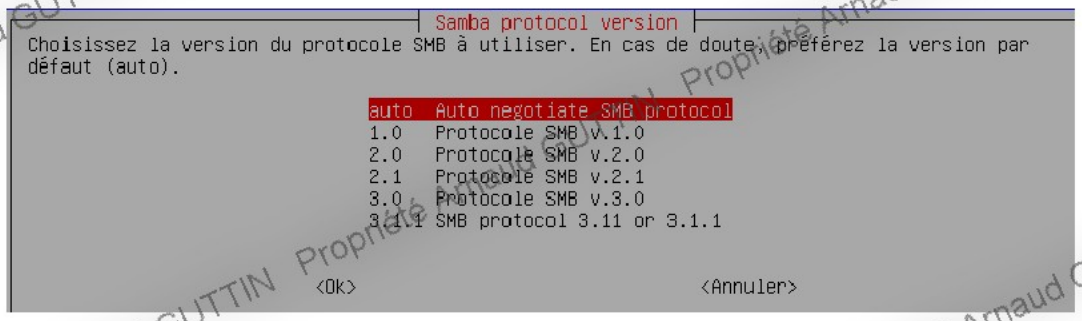


Il faut ensuite entrer le nom d'utilisateur avec lequel CloneZilla se connecte au serveur de partage.

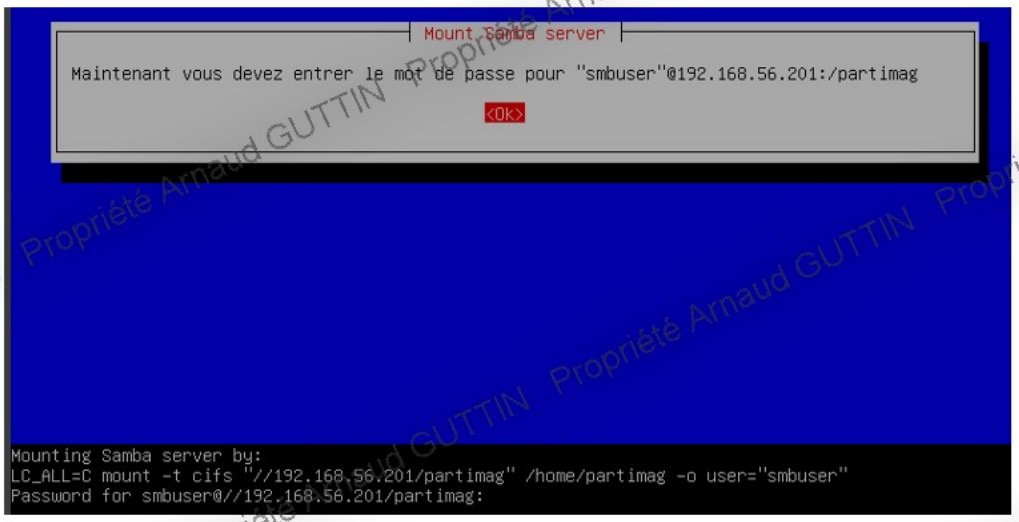
Veuillez préciser le nom du partage, ici, /partimag.



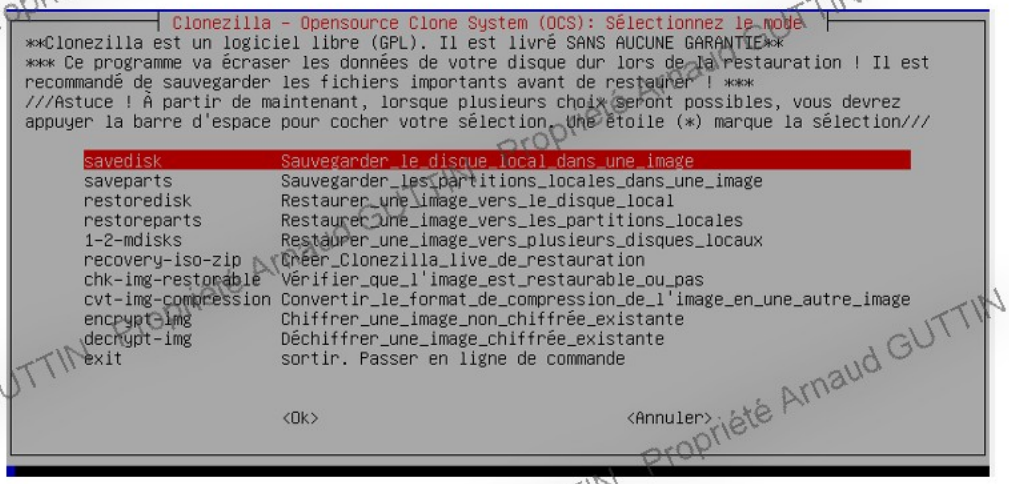
Laissez la version de protocole en auto.



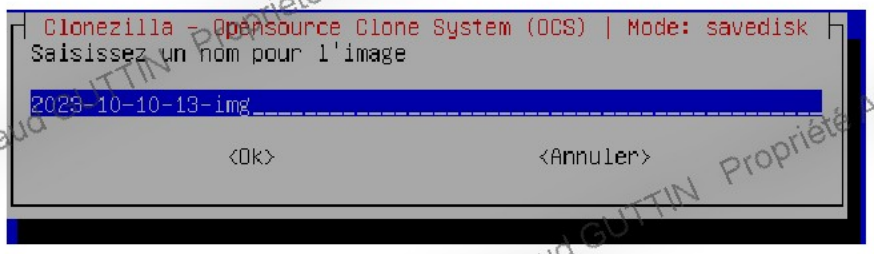
Il est maintenant nécessaire d'entrer le mot de passe de l'utilisateur smbuser pour autoriser la connexion au serveur Samba.



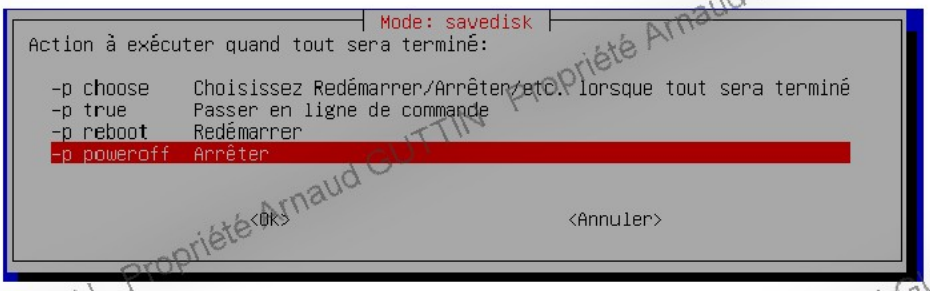
Il vous sera ensuite demandé si vous souhaitez sauvegarder le disque local dans une image, dans une partition, ou si vous souhaitez restaurer une image. Choisissez savedisk, (voir ci-dessous).



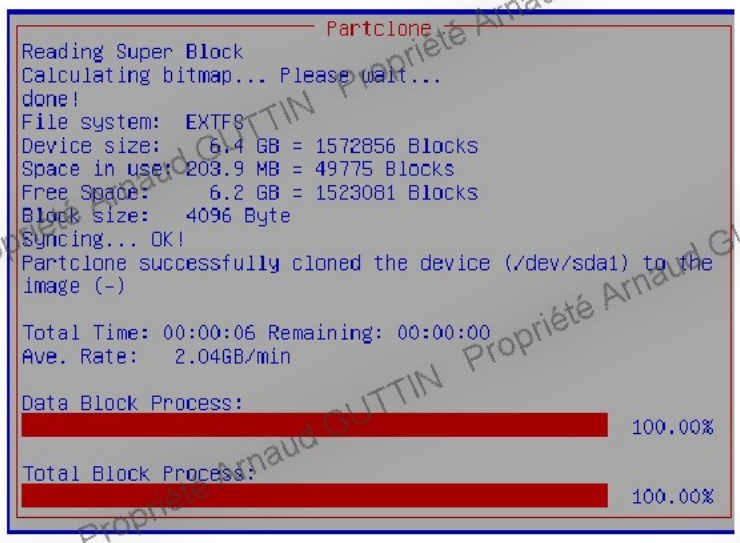
Vous pouvez ensuite renommer le nom de votre image.



CloneZilla demandera quel action il effectuera a la fin de l'exportation, sélectionnez power off. Si vous mettez reboot et que vous n'avez pas enlevé l'image CloneZilla, la machine va rebooter dessus.



Vous avez ensuite une échelle de progression de l'exportation de l'image.
Veuillez patienter !

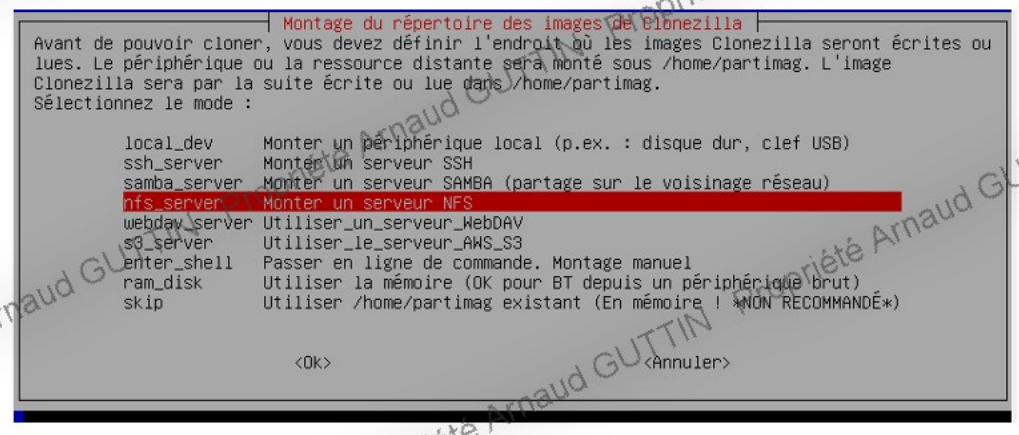


Sur notre serveur Samba (czimage) nous pouvons effectuer la commande `ls /home/partimag` et voir que l'image se situe bien dans le répertoire.

Importation d'une image sur une machine avec CloneZilla

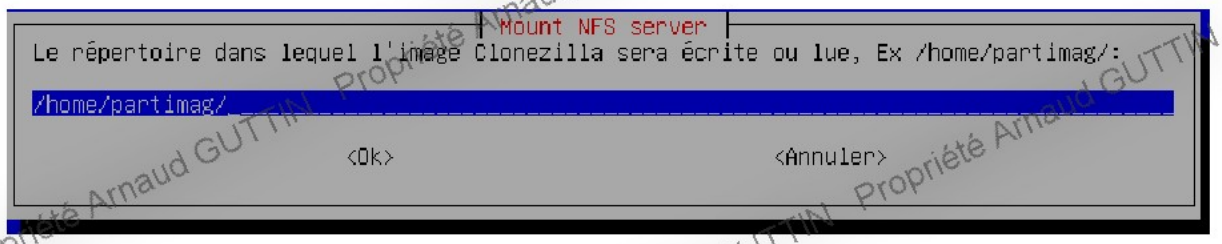
Pour importer une image avec CloneZilla il faudra recommencer l'étape d'installation de CloneZilla,

Pour notre activité, nous allons faire descendre l'image sur la nouvelle machine grâce au service NFS.
Lorsque CloneZilla vous demande quelle méthode de transfert vous souhaitez utiliser, choisissez nfs-server, (voir ci-dessous).

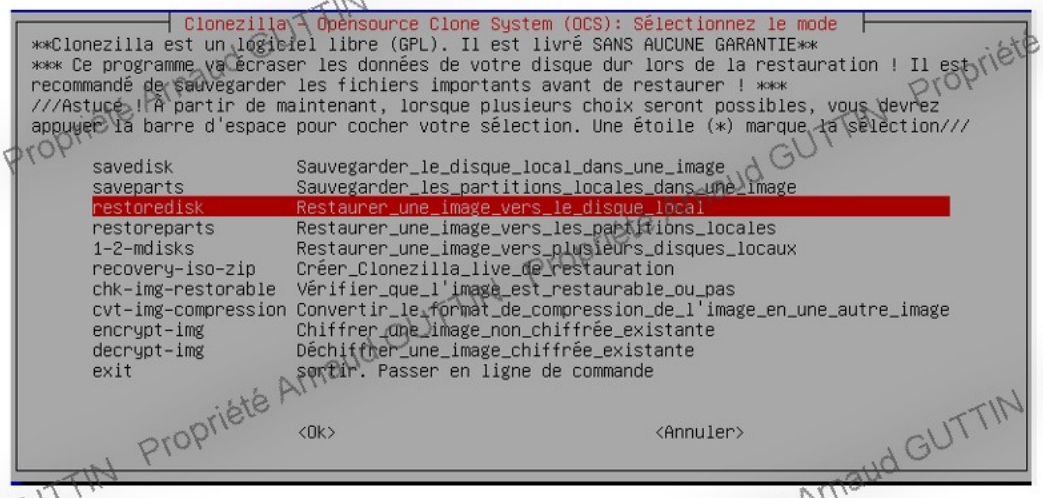


Il sera ensuite nécessaire de donner l'adresse IP du serveur NFS (czimage).

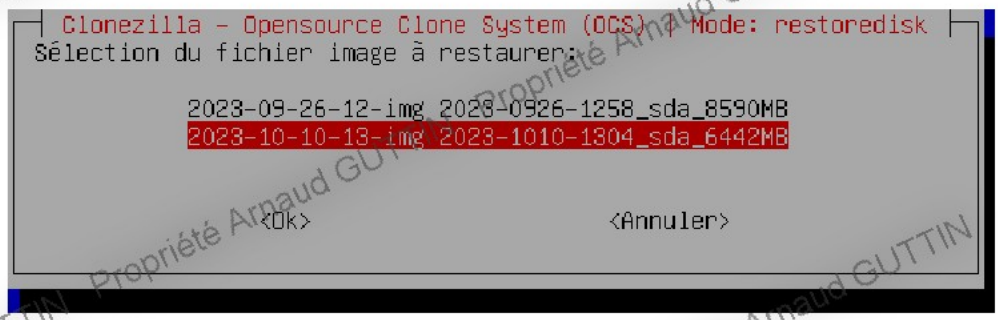
Vous devez ensuite donner le chemin du répertoire de partage, (voir ci-dessous).
Attention, précédemment nous avons utilisé /partimag pour le protocole Samba, ici il faut veillez à utiliser le chemin total /home/partimag pour le protocole NFS.



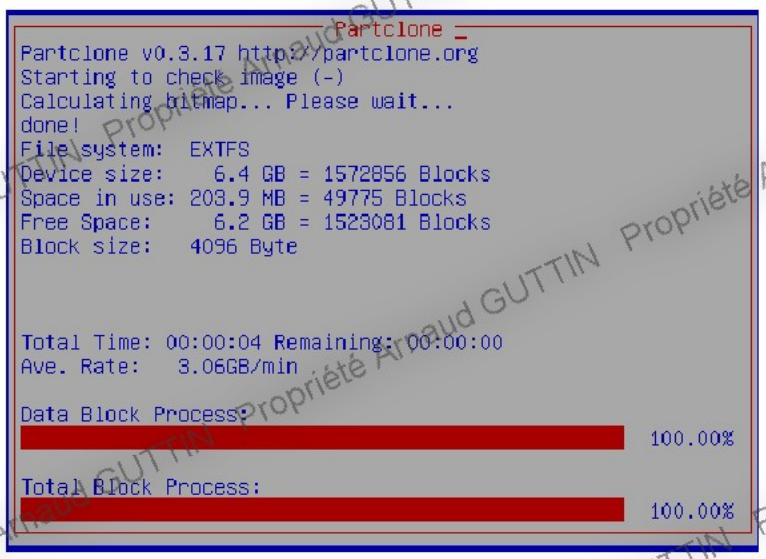
Vous pouvez ensuite choisir l'option "restoredisk", pour restaurer l'image précédemment réalisée.



Lorsque CloneZilla aura analysé le serveur NFS, il vous proposera les différentes images que vous aurez réalisé, il faudra choisir celle que vous souhaitez, (voir ci-dessous).

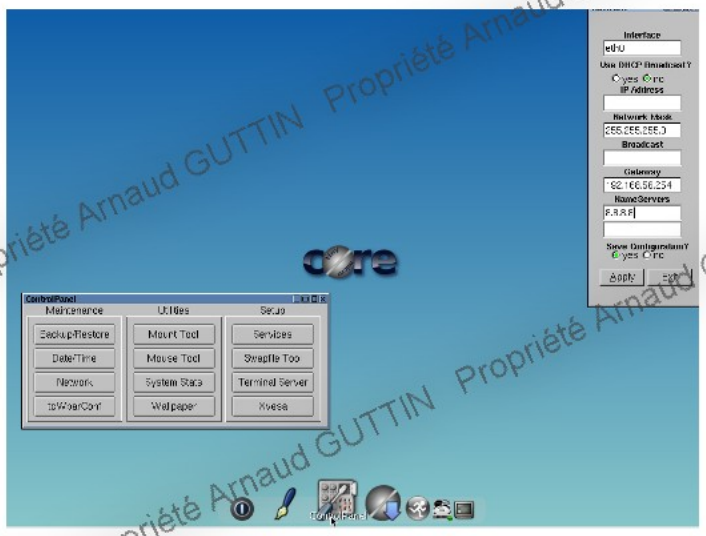


Ensuite, le transfert d'image entre la machine czimage et la machine vierge débutera.



Attention, lorsque vous effectuez cette action, CloneZilla vous préviens que l'intégralité des données présentes sur le disque vont être supprimées, (voir ci-dessous).

Lors du redémarrage de votre machine, vous retrouvez exactement le système d'exploitation avec la configuration de la précédente machine.



Propriété Arnaud GUTTIN