**Installation d’un réseau avec Tiny Core**

**Ojectif :** créer un réseaux composer d’une machine Debian buster 10, d’une machine TinyCore CorePlus et de les faire communiquer par l’intermédiaire d’un routeur.

Sommaire

[Schéma du réseau 2](#_Toc115440985)

[Installation de l’environnement 2](#_Toc115440986)

[Création et configuration du TC-Routeur 2](#_Toc115440987)

[Création et configuration de la machine Debian czimage 3](#_Toc115440988)

[Rappel sur Telnet 4](#_Toc115440989)

[Présentation 4](#_Toc115440990)

[Utilisation 4](#_Toc115440991)

[SSH 5](#_Toc115440992)

[Création de la vm tc master, installation de CorePlus et de TinyCore. 6](#_Toc115440993)

[Installation de CorePlus 7](#_Toc115440994)

[Installation de TinyCore 8](#_Toc115440995)

[Installation d’une machine cliente avec interface graphique LXDE 11](#_Toc115440996)

[Samba 13](#_Toc115440997)

[Installation de Samba coté serveur (czimage) 14](#_Toc115440998)

[Installation de Samba côté client (LXDE) 14](#_Toc115440999)

[Coté client Windows 14](#_Toc115441000)

[NFS (Network File System) 15](#_Toc115441001)

[Côté serveur (czimage) 15](#_Toc115441002)

[Côté client (LXDE) 16](#_Toc115441003)

[Clonage avec Clonezilla 17](#_Toc115441004)

[Qu’est-ce que Clonezilla 17](#_Toc115441005)

[Clonage de l’image de tc-master pour l’envoyer sur le serveur czimage par Samba 17](#_Toc115441006)

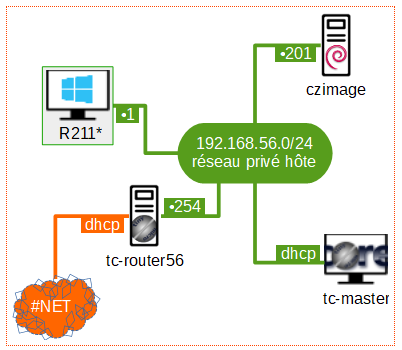
[Création de tc-clone 27](#_Toc115441007)

[Clonage avec SSH 32](#_Toc115441008)

[SSH avec échange de clés 33](#_Toc115441009)

[Debian vers Debian 33](#_Toc115441010)

# Schéma du réseau

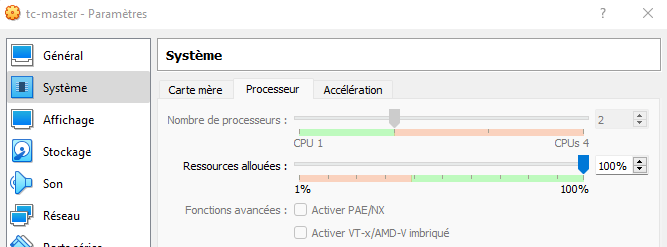


# Installation de l’environnement

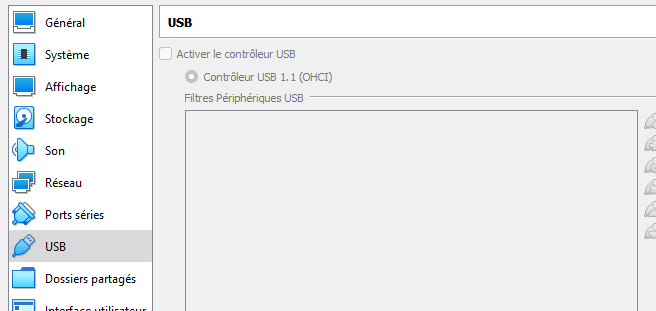
## Création et configuration du TC-Routeur

On créer tc-routeur en important le fichier VDIdans VirtualBox sans oublier :

* De lui ajouter un cœur



* De désactiver le contrôler usb

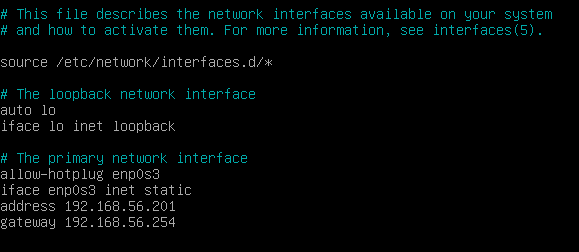


* De prendre un instantané de la machine permettant de ne pas affecter directement la machine avec les manipulations diverses.

## Création et configuration de la machine Debian czimage

On créer la vm czimage, sous une distribution linux Debian Buster 10, et l’on y configure l’interface réseaux en éditant le dossier ***/etc/network/interfaces*** avec la commande ***nano.***

1. Mettre l’adresse IP en « static »
2. Entrer l’adresse ip de la machine avec **address *ip\_utilisée***
3. Entrer la passerelle du TC routeur (si utilisé) avec **gateway *ip\_de\_la\_passerelle***



1.

2.

3.

# Rappel sur Telnet

## Présentation

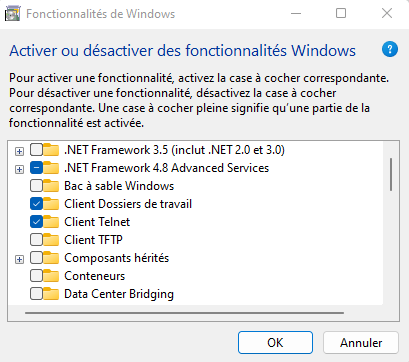
* Terminal network (port 23 par défaut)
* Protocole utilisé sur tout réseaux TCP/IP, permettant de communiquer avec un serveur distant en échangeant des lignes de texte et en recevant des réponses également sous forme de texte
* Tombé en désuétude par défaut de sécurisation
* Toutes communication est transmise en clair sur le réseaux, mot de passe compris
* Des sniffers comme tcpdump ou wireshark permettent d’intercepter les communications

## Utilisation

* Si la connexion est réussie, un écran vierge sera affiché, ce qui signifie que le port de l’ordinateur est ouvert
* Une connexion ratée sera accompagnée d’un message d’erreur

Installation et utilisation

Activer la fonction Telnet en allant dans ***activer ou désactiver des fonctionnalités Windows***



Sélectionner le Client Telnet et cliquer sur OK pour l’activer

Sur la machine Debian, effectuer un **apt update** pour mettre à jour les paquets avant toute installation.

Installer telnetd en utilisant ***apt install telnetd***

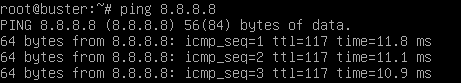
Reboot la machine Debian

Se connecter à la machine Debian depuis Powershell avec la commande ***Telnet 192.168.56.201***

# SSH

* Secure Shell est un protocole sécurisé
* Utilise le port 22
* Fournit une connexion de gestion sécurisée à un appareil distant
* Remplace Telnet, FTP
* Permet : accès ligne de commandes, transferts de fichier, accès ressources distantes, transport d’autres protocoles via tunnel
* Principe technique : openssh-server sur la machine distante et un client ssh sur votre poste (ssh sur debian ou autre client ssh comme Putt, WinSCP)

Tester la connexion en faisant un ping du serveur DNS de Google avec ***ping 8.8.8.8***



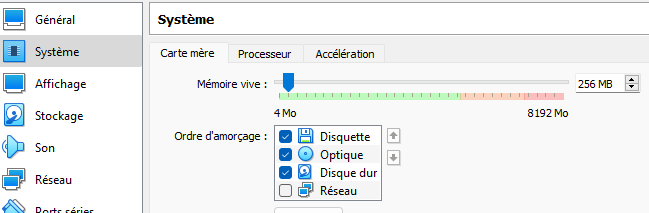
La connexion fonctionne.

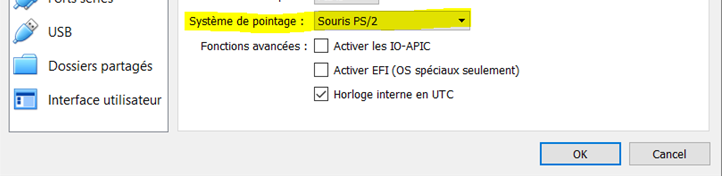
Renommer le nom de la machine et créer les répertoires suivants :



## Création de la vm tc master, installation de CorePlus et de TinyCore.

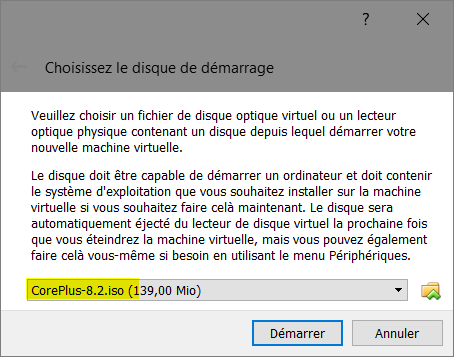
On peut modifier l’ordre d’amorçage (le matériel sur lequel l’OS va booter) pour empêcher que la machine tc-master reboot sur l’iso après suppression de ce dernier.

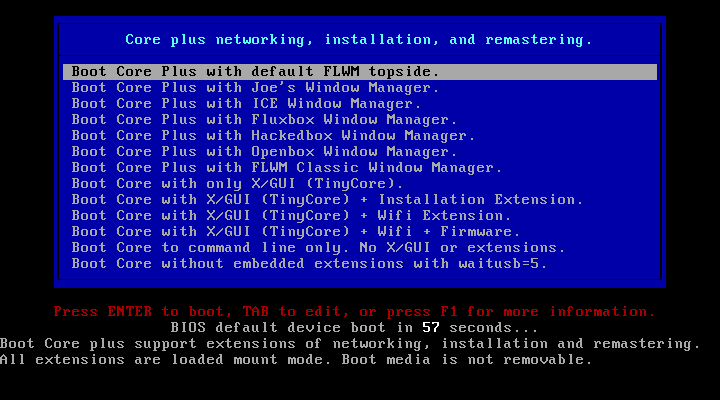




### Installation de CorePlus

Ajouter le fichier ISO CorePlus8.2 depuis le site internet de TinyCore et le sélectionner pour le démarrage de la vm tc-master



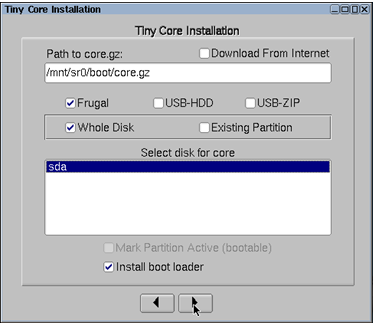


### Installation de TinyCore

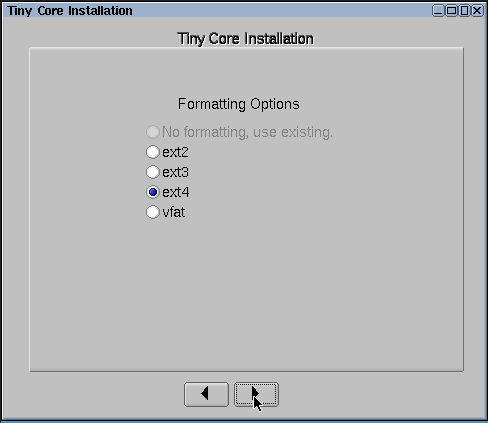
Cliquer sur l’icône encadré pour lancer l’installation de tinycore

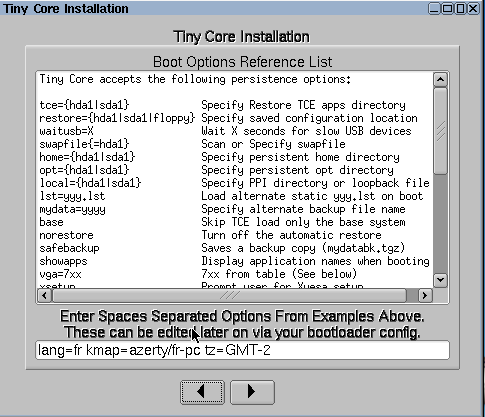


Sélectionner le disque sda (nom du disque principal sous linux)



Sélectionner ext4 (système de fichier proposé le plus récent)





1.

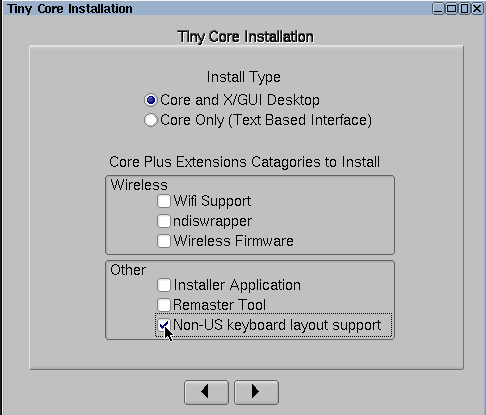
2.

3.

1. définie la langue de la machine en français

2. définie le mappage de touche du clavier en azerty

3. définie le fuseau horaire de la machine (tz pour timezone)



1.

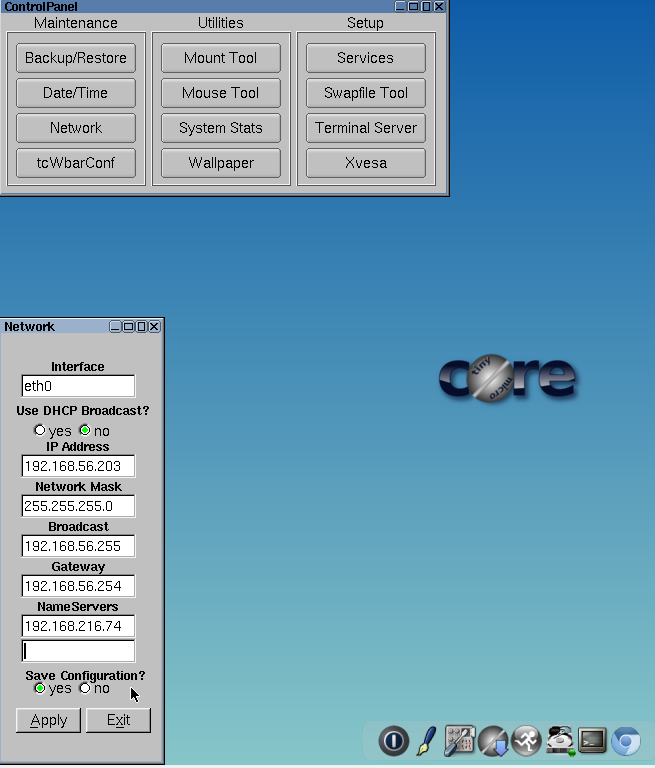
2.

1. choisit l’interface graphique de TinyCore

2. le clavier US (qwerty) n’est pas pris en charge

Valider les paramètres dans la review et une fois TinyCore installé **shuttdown la machine** et sur Virtual box **retirer le fichier ISO de TinyCore** depuis l’onglet stockage de la configuration de la machine tc-master. **Car si ce n’est pas fait, il faudra recommencer l’installation de TinyCore.**

Configuration de l’interface réseau de tc-master



## Installation d’une machine cliente avec interface graphique LXDE

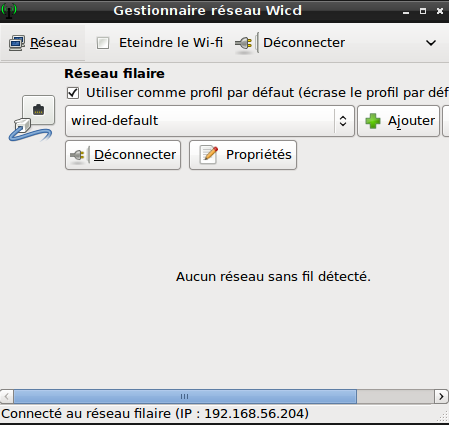
Créer une VM avec l’image disque 

Configurer l’interface réseaux cette fois-ci via l’interface graphique :

* On clique sur l’icône réseau filaire connecté

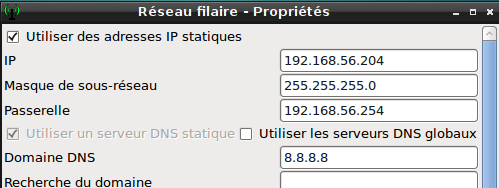


* On se rend dans « propriété »



* On coche « utiliser des adresses IP statiques » et on entre :

1. L’ip du poste
2. L’ip du tc-routeur
3. L’ip du serveur DNS de Google



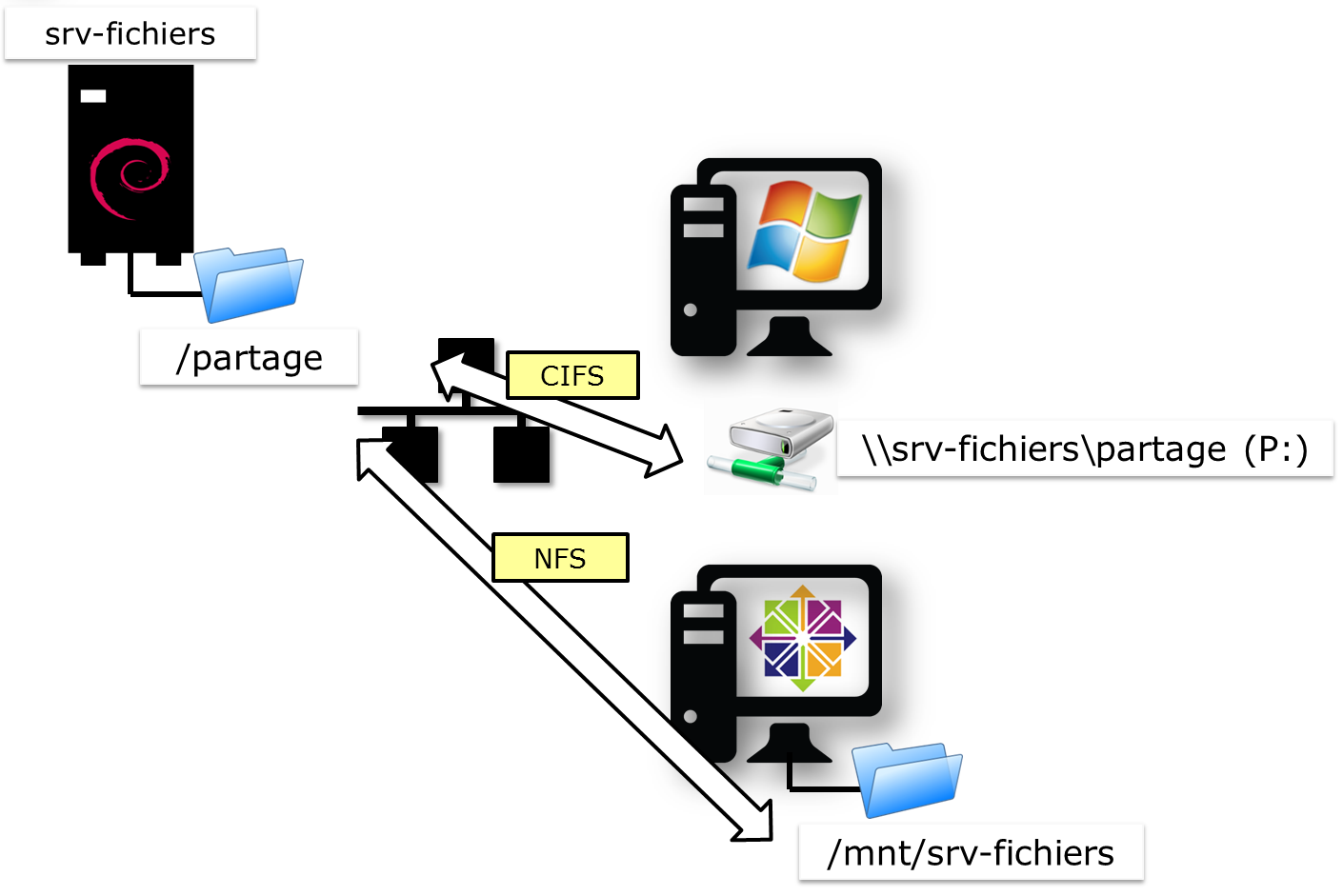
1.

3.

2.

# Samba

* Serveur de fichier permettant l’interopérabilité entre divers systèmes
* Permet à des systèmes linux de créer des partages utilisables par des machines Windows et vice-versa
* Utilise le protocole SMB et CIFS



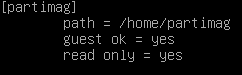
* SMB (Server Message Block) est un protocole serveur-client réglant l’accès à des fichiers, à des répertoires complets et d’autres ressources du réseau comme les imprimantes, les routeurs ou les interfaces partagé dans le réseau.

## Installation de Samba coté serveur (czimage)

* Installer le serveur samba avec 
* Ajouter le dossier partagé à la fin du fichier de configuration ***/etc/samba/smb.conf*** avec nano et ajouter tout en bas du fichier le texte suivant :

1. Le nom du dossier partagé
2. Le chemin du dossier partagé
3. Autorise les invités sur le dossier
4. Donne accès au dossier seulement en lecture

1.



2.

3.

4.

* Pour valider la configuration entrer  dans l’invite de commande après avoir enregistré l’édition du fichier smb.conf. Un texte s’affiche reprenant les modifications et indiquant si les commandes rentrées son intègre, que la syntaxe est correcte (elles seront ignorées dans le cas contraire).
* Utiliser ***service smbd restart*** pour redémarrer le service Samba

## Installation de Samba côté client (LXDE)

* Mettre à jour les paquets avec 
* Installer le client samba avec 
* Se connecter à la ressource distante avec 

1.

2.

3.

4.

5.

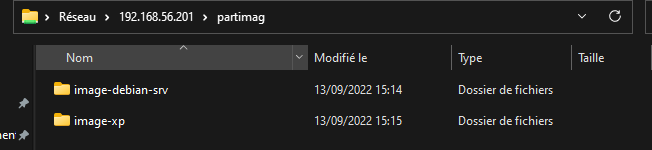
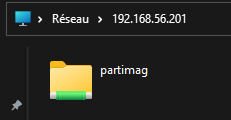
1. Commande de samba pour se connecter
2. Ip du serveur
3. Nom du dossier que l’on veut consulter
4. Indique le profil de connexion User
5. Le nom du profil User

## Coté client Windows

Tenter de consulter le dossier partimag dans l’explorateur de fichier dans la section réseau, activer le partage de fichier si ce n’est pas déjà fait et entrer l’ip du serveur samba

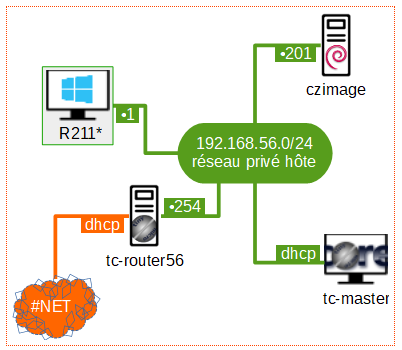


On a bien accès au dossier partimag



# NFS (Network File System)

* Protocole qui permet d’accéder à des fichiers via le réseau.
* Les clients montent la partition de la machine distante comme si c’était un disque local

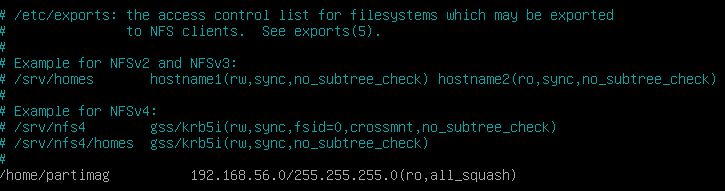


.202

Partage du répertoire /home/partimag du serveur dans le répertoire /media/NFS du client

## Côté serveur (czimage)

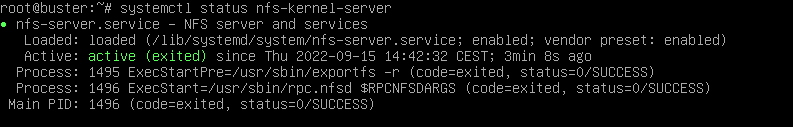
* Installation du serveur NFS avec 
* Ajouter le dossier partagé au fichier de configuration ***/etc/exports***



(read only) donne accès au dossier en lecture seule

Permet de rendre actif le partage pour tous les utilisateurs

* Relancer le service avec 
* Vérifier qu’il soit bien relancer avec ***systemctl status nfs-kernel-server***



## Côté client (LXDE)

* Montage de la ressource distante en root avec 

Permet d’afficher les ressources que partage le serveur NFS sur le réseau et de crée le liens NFS avec le serveur

**-----------------------------------------------------------------------------------------------------------------**

Petite parenthèse :

Il y a une possibilité que le terminal renvoi le message « commande introuvable » cela signifie que la machine n’a pas d’installé le paquet permettant d’utiliser la commande showmount

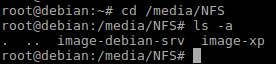
Pour ce faire, il faut utiliser la commande ***apt-cache search showmount*** elle permet de trouver où se trouve le paquet contenant showmount (apt-cache permet de trouver une chaîne de caractère dans la liste des paquets)

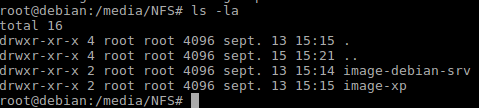
* Installation du paquet nfs-common avec 
* Le répertoire /home/partimag du serveur est bien monté dans le répertoire /media/NFS du client.





* Vérification





# Clonage avec Clonezilla

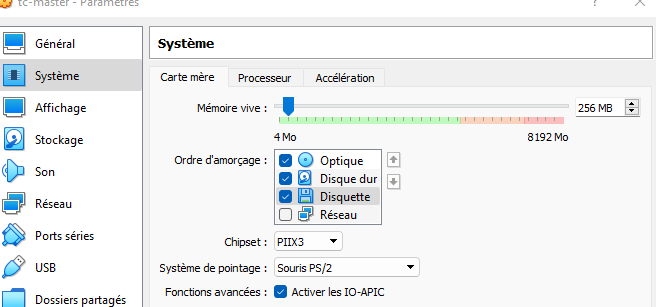
## Qu’est-ce que Clonezilla

 Un logiciel libre de restauration de données, de clonage de disque, et de création d'image de disque. Créer par Steven Shiau. Alternative libre aux logiciels proprio payants (Norton Ghost, Acronis True Image, Rembo, ou même RIS/WDS des Windows Server)

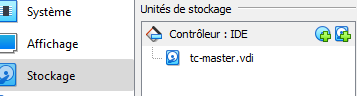
## Clonage de l’image de tc-master pour l’envoyer sur le serveur czimage par Samba

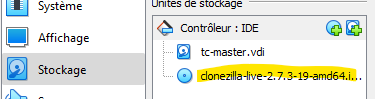
On créer une vm tc-clone 1 que l’on laisse vierge

Sur la vm tc-master, changer l’ordre d’amorçage (de boot) pour que la vm boot sur l’iso de Clonezilla.



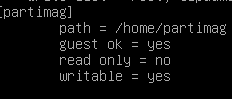
On ajoute l’iso de Clonezilla sur tc-master



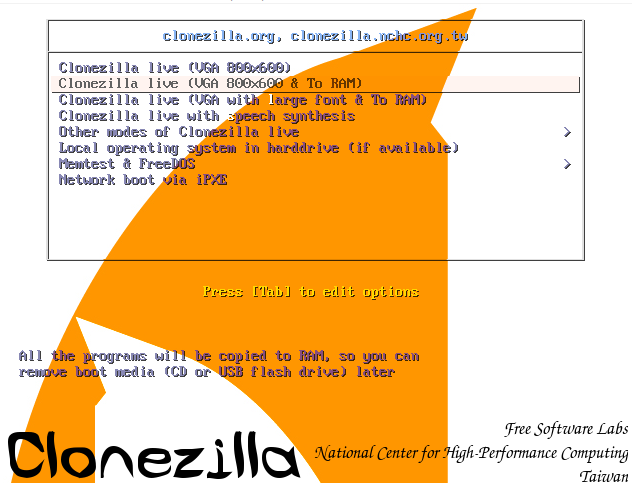


On démarre la vm tc-master

Avant l’installation de Clonezilla modifier le fichier /etc/samba/smb.conf+



On lance Clonezilla live (VGA 800x600)

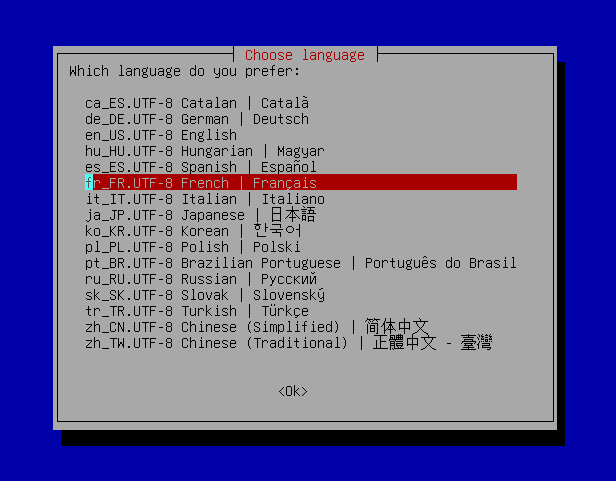


Mais une erreur survient le message suivant s’affiche « Kernel requires an x86-64 cpu but only detected an i686 cpu ».

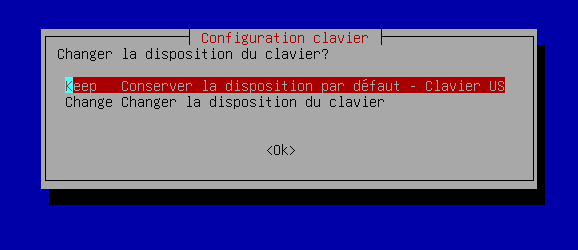
Or, il s’agit d’une erreur de version de l’os de tc-master lors de la création de la vm, il faut donc faire 3 choses :

* Changer la version de tc-master « other linux 32bits » en « debian 64bits »

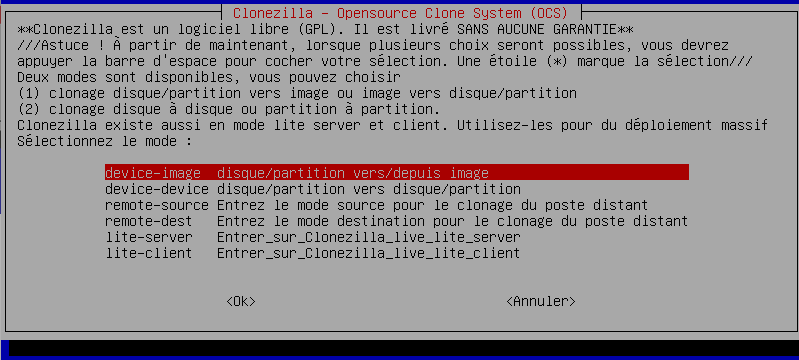
On définit la langue en français



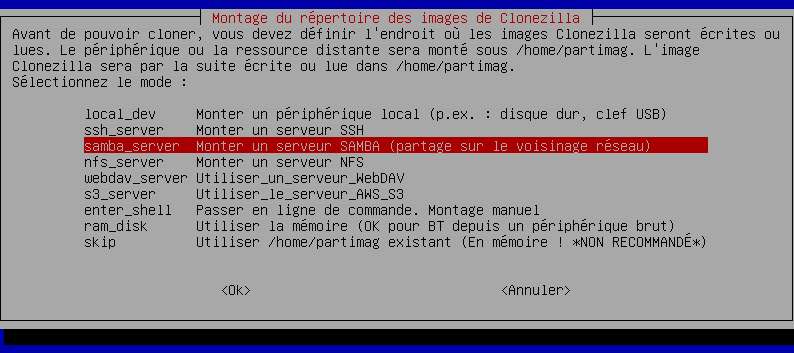
On laisse le clavier US



On démarre Clonezilla



On souhaite créer un serveur Samba pour envoyer l’image à czimage



On choisit « static » et on entre les données suivantes

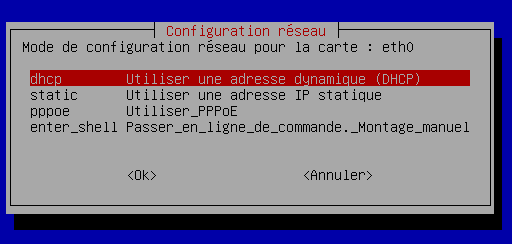
L’ip de la machine : 192.168.56.250 (on choisis l’ip que l’on veut)

Le masque : 255.255.255.0

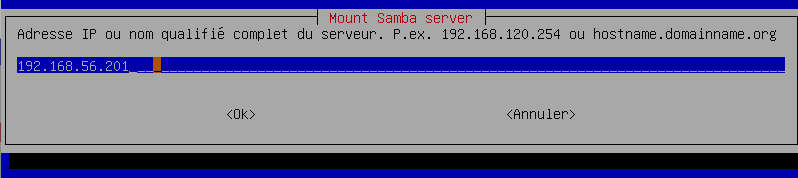
La passerelle : 192.168.56.254

Le dns : 8.8.8.8 (DNS de Google)

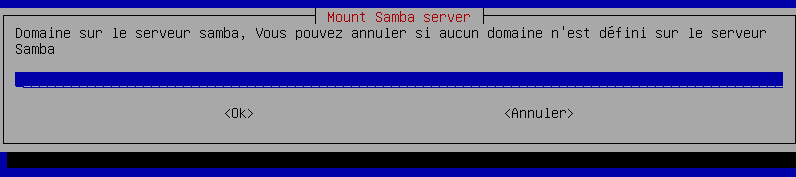
L’ip du serveur samba : 192.168.56.201 (l’ip de czimage dans notre cas)



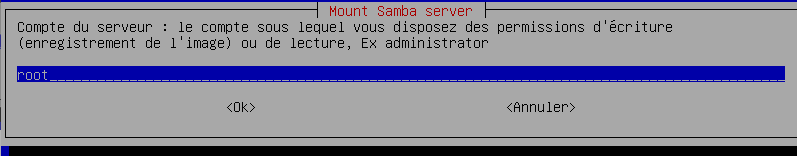
On entre l’IP du serveur auquel on souhaite se connecter



Le réseau n’a pas de domaine donc cliquer sur « annuler »

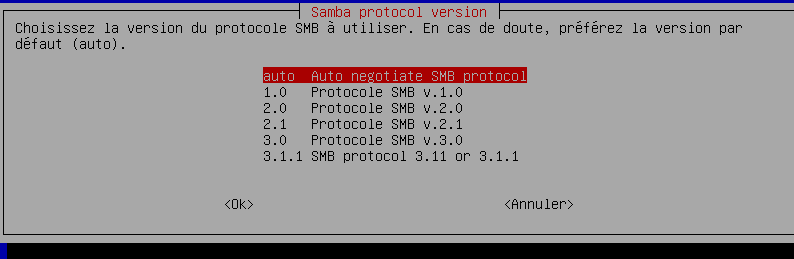


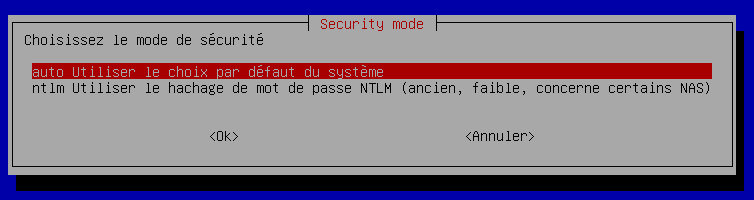
On choisit ensuite le nom d’utilisateur que l’on souhaite. Les droits sont ouverts pour « root » et « sio »



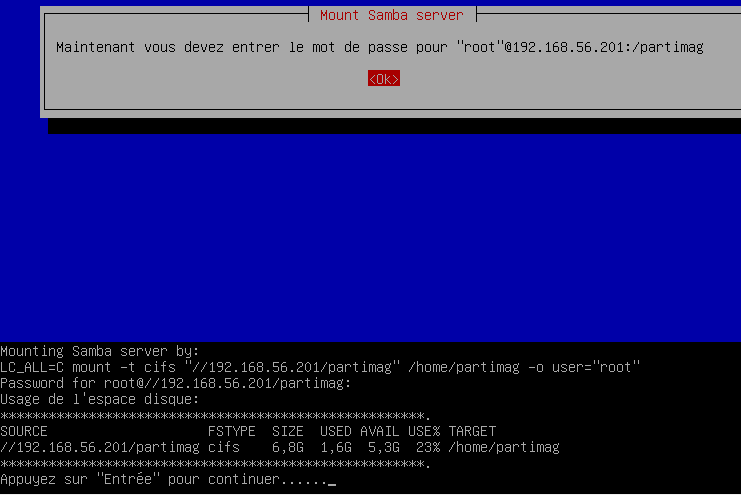
On entre l’arborescence du dossier partager, dans notre cas /partimag

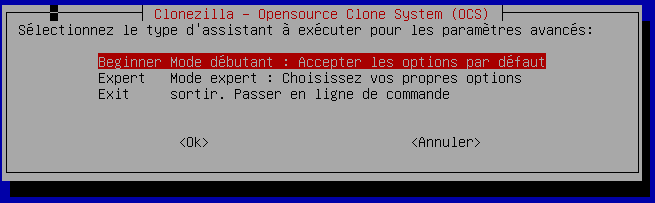
On laisse le choix du protocole SMB en auto



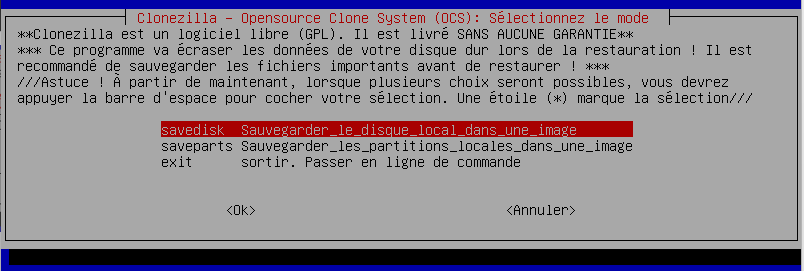


On entre le mdp du compte root dans le Shell

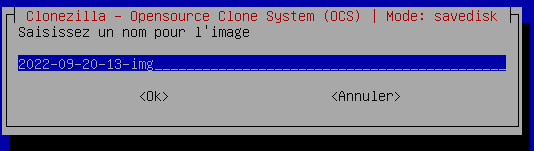




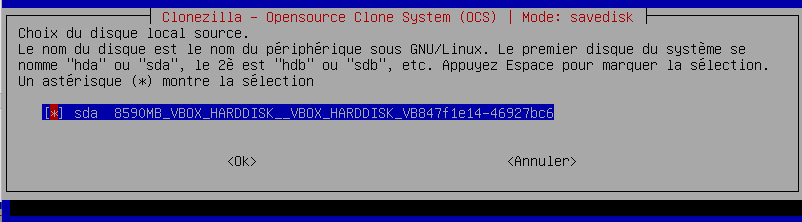
On choisis de sauvegarder le disque dans l’image



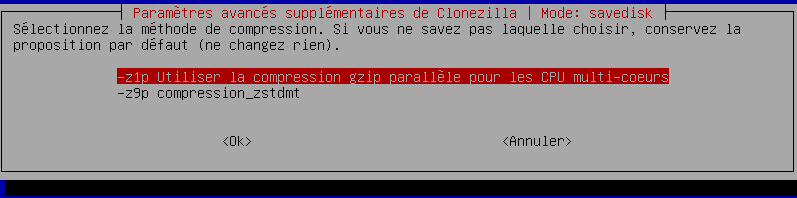
On laisse le nom donné par défaut



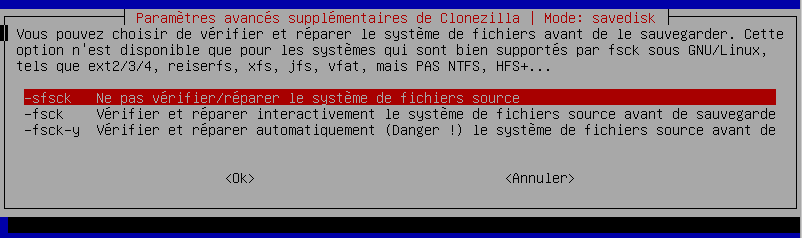
Sélectionner le disque en surbriance



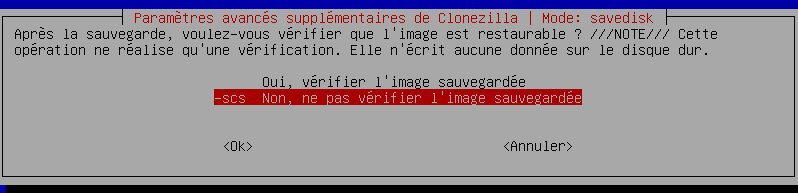
Utiliser la compression gzip

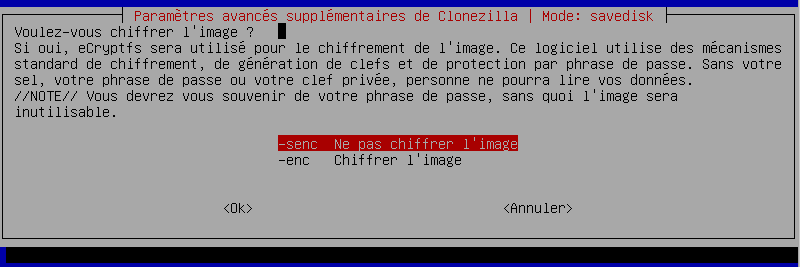


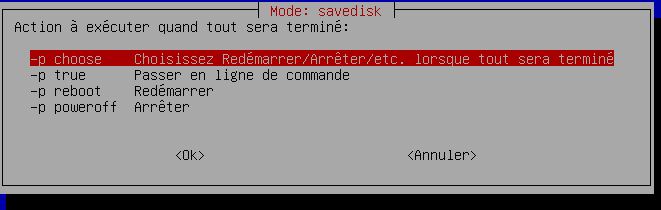
Choisir sfsck

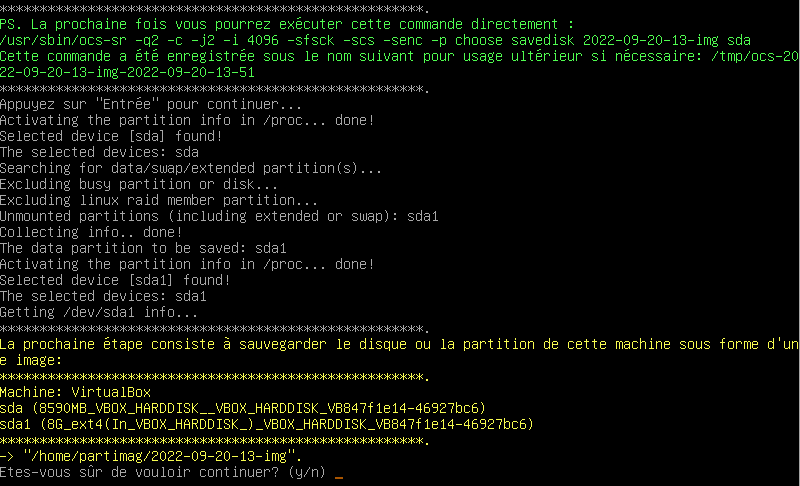


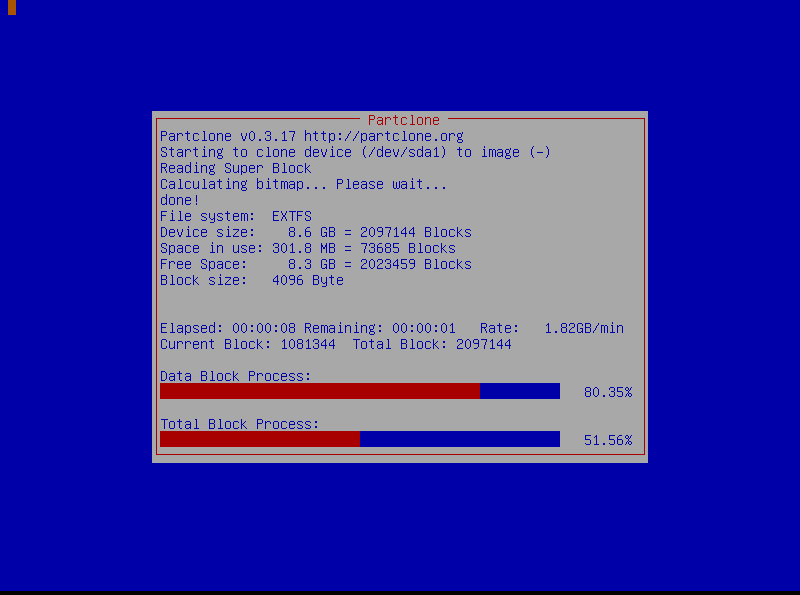
On peut choisir de vérifier ou pas l’intégrité de l’image (on pourrait aussi ne pas la vérifier si l’OS est volumineux pour gagner du temps)

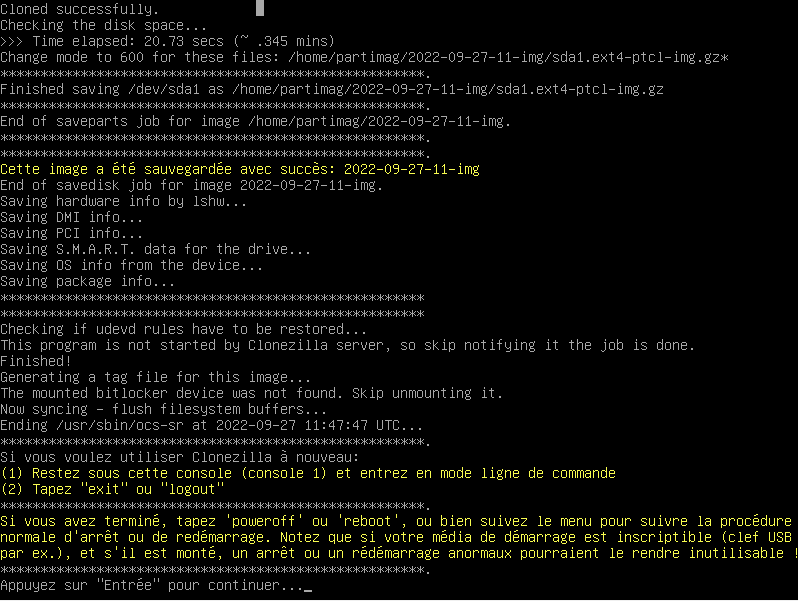










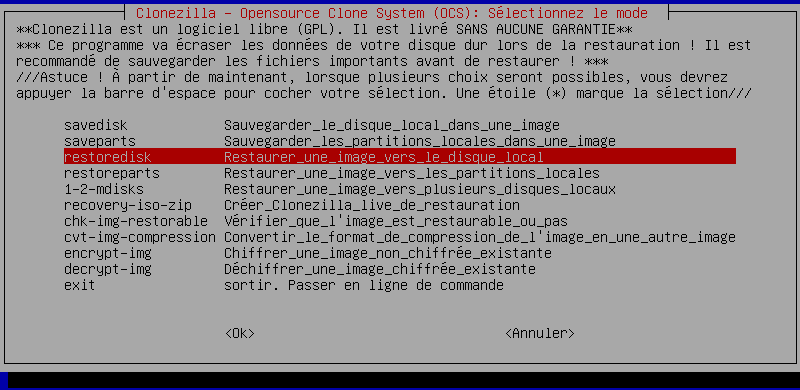


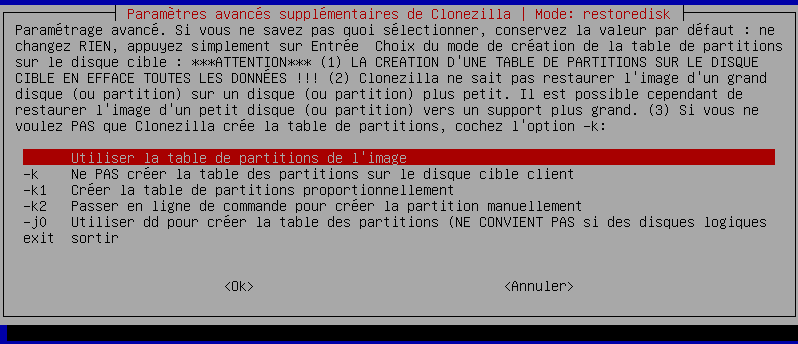
## Création de tc-clone

Créer une VM vide (sans images disque) et incorporer l’iso de Clonezilla dans l’onglet « stockage » de configuration de la VM

Démarrer tc-clone sur l’iso de Clonezilla

Suivre le même protocole de configuration que pour la vm tc-master jusqu’à cette partie et bien choisir de restaurer une image

  
on utilise la table de partitions de l’image

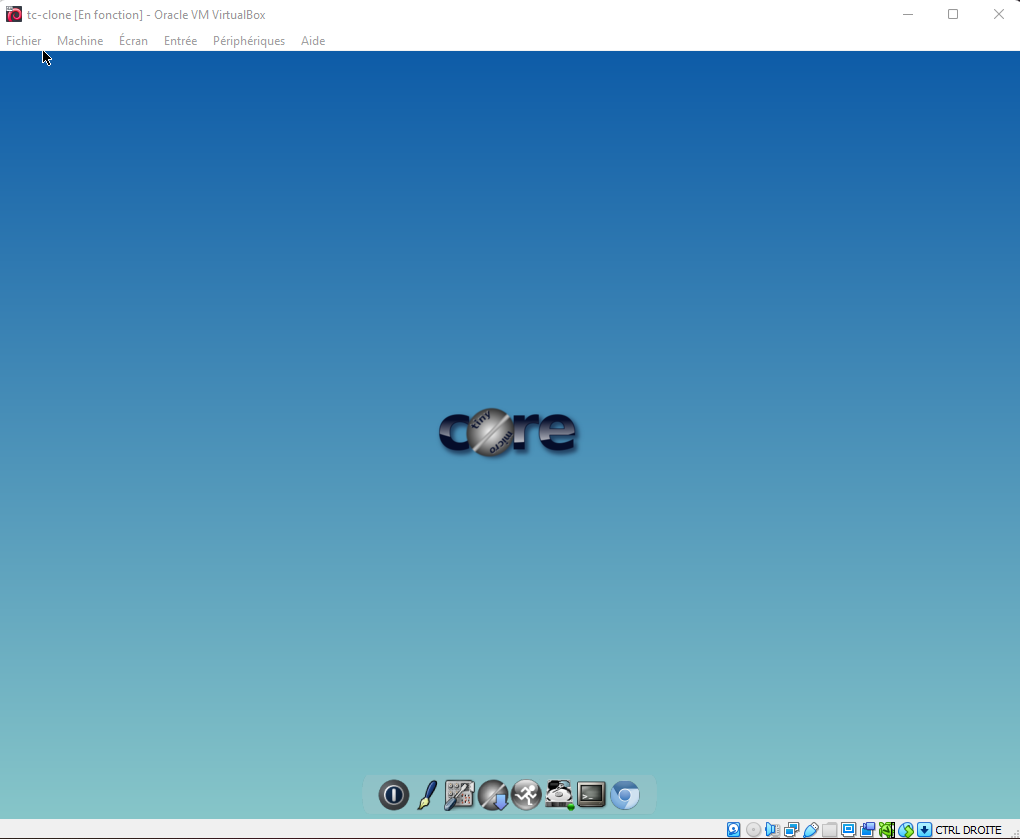


Vérifier l’image avant la restauration

On éteint la vm à la fin de la configuration et on éjecte l’iso Clonezilla (si cela n’a pas été fait tout seul).

On la redémarre pour vérifier que l’image est fonctionnelle, que nous avons bien l’image de tc-master sur la VM tc-clone.

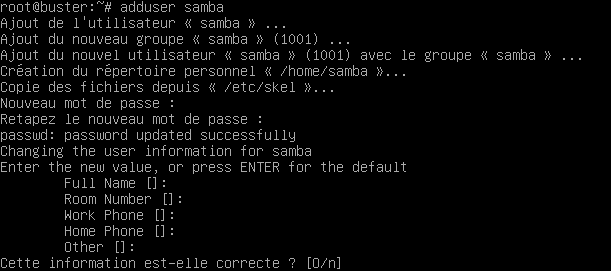
C’est le cas !



Création de l’utilisateur samba

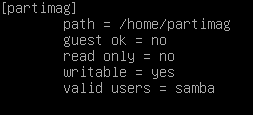
On souhaite créer un utilisateur à utiliser uniquement pour le clonage de façon à ne plus à avoir besoin d’utiliser les profils root/sio, il serait par adéquation le seul utilisateur à accéder à partimag

Depuis czimage, on créer l’utilisateur « samba » avec la commande ***adduser samba***



Configurer la fichier smb.conf

Il faut maintenant configurer le fichier smb.conf de manière à ce que le user « samba » ai lui seul un accès en écriture à partimag



On met « gest ok » en « no » pour limiter l’accès au répertoire

On définit l’utilisateur « samba » comme seul utilisateur valide pour consulter le répertoire

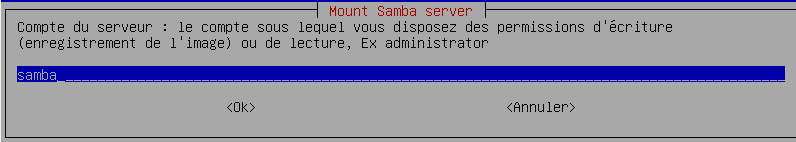
Ne pas oublier de faire la commande ***testparm*** juste après avoir modifié le fichier pour vérifier l’intégrité du fichier smb.conf, redémarrer Samba avec la commande ***service smbd restart*** et la commande ***chown –R $USER :samba /home/partimag***

Création de tc-clone 2 pour tester l’utilisateur samba

Lancer la VM avec Clonezilla en insérant l’iso dans l’onglet stockage de la configuration de tc-clone 2

Suivre les étapes comme pour les installations précédentes et choisir l’utilisateur « samba » précéde

mment créé.



Lorsqu’on se connecte avec le profil samba on a l’erreur suivante

Mount error (13) access denied

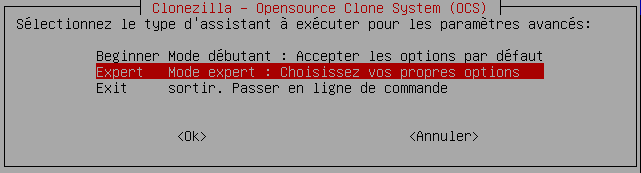
L’explication est que l’utilisateur samba est un simple utilisateur Debian, il n’est donc pas relié et reconnu par samba

Il faut faire la commande ***smbpasswd –a NomduUser*** (ici, « samba ») sur la machine serveur (czimage)

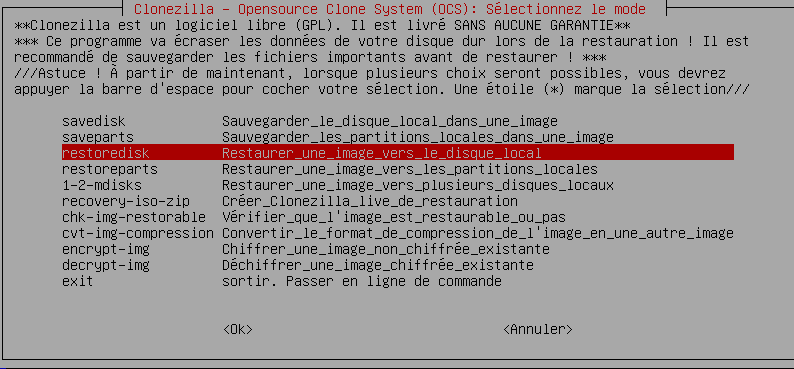
Pour donner accès au dossier partimag à l’utilisateur, il faut entrer la commande ***pdbedit –a NomDuUser*** (ici, toujours « samba »)

La connexion fonctionne lors de l’installation du disque

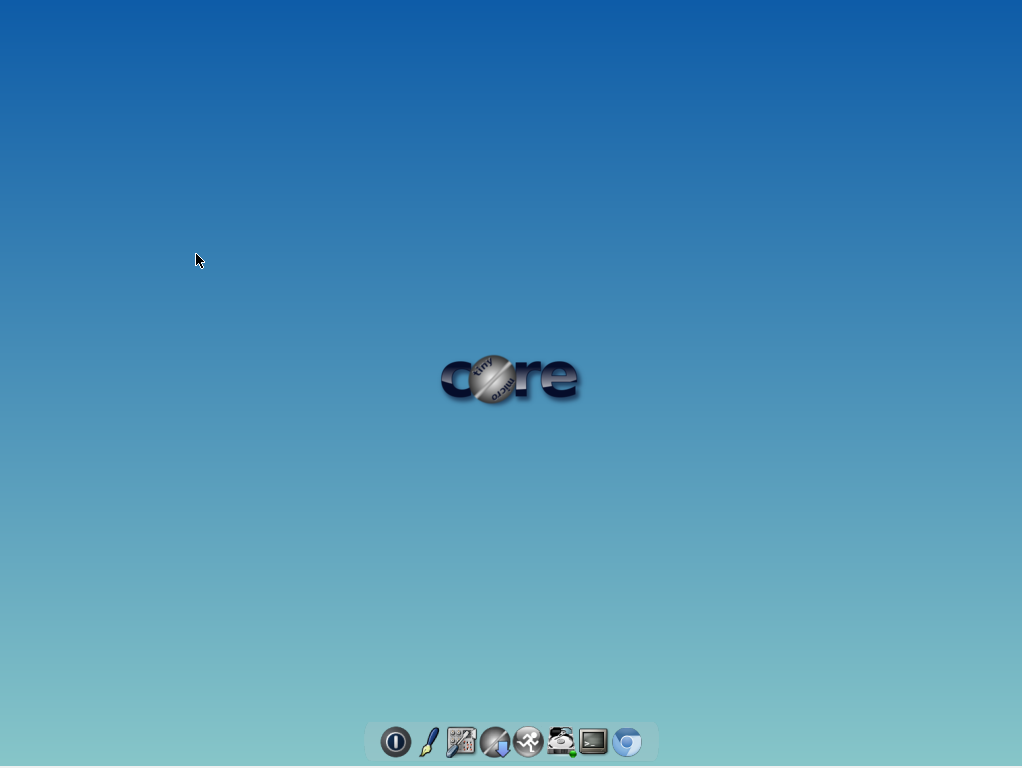
Après cela choisir le mode expert



Ensuite, on choisit de restaurer le disque pour descendre l’image sur la VM tc-clone2



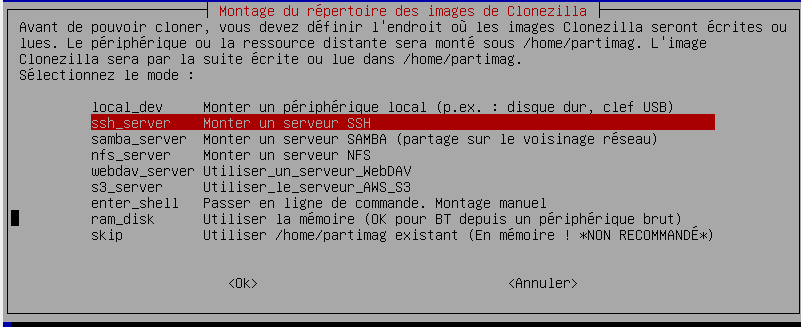
Après redémarrage, l’image est bien installée



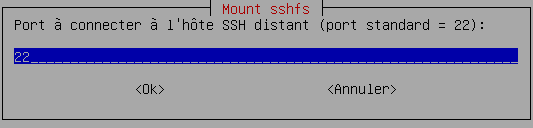
## Clonage avec SSH

Créer la VM tc-clone3, la mettre en réseaux privé hôte et la démarrer avec clonezilla

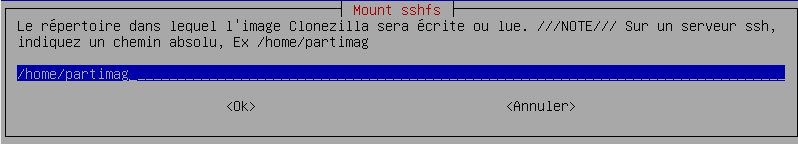
Suivre le même protocole que pour le clonage précédent à l’exception de cette étape (sélectionner ssh\_server)



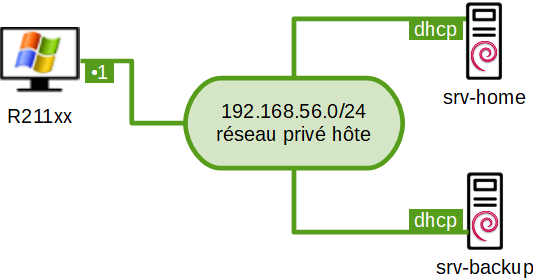
Choisir le port 22 (port par défaut de SSH)



Et bien entrer le chemin **absolu** de partimag



# SSH avec échange de clés



Créer les 2 VM Debian buster 10, les configurer en réseaux privé hôte sans oublier de rajouter un cœur, configurer l’interface réseaux et installer SSH sur chacune avec ***apt install ssh***

## Debian vers Debian

* Objectifs : accéder à SRV-HOME depuis SRV-BACKUP
* Générer la paire de clés privées/publiques et vérifier l’empreinte (fingerprint) sur srv-backup
* Pas de passphrase
* Se connecter en sio
* Bien renommer les serveurs pour plus de lisibilité

Sur la machine SRV-BACKUP, se connecter avec le profile « sio » et générer la paire de clés asymétrique avec la commande ***ssh-keygen***

Appuyer sur entrer pour valider le chemin



Appuyer sur entrer pour ne pas donner de passphrase



Voici le fingerprint



Copier le fingerprint sur SRV-HOME pour qu’il se souvienne de SRV-BACKUP avec la commande

***ssh-copy-id –i*** [***sio@192.168.56.101***](mailto:sio@192.168.56.101)

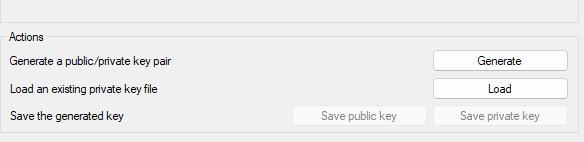
Entrer le mdp du profil « sio »  la clé à bien été ajoutée.

Se connecter en ssh à SRV-HOME depuis SRV-BACKUP pour tester la connexion avec la commande ***ssh*** [***sio@192.168.56.101***](mailto:sio@192.168.56.101)

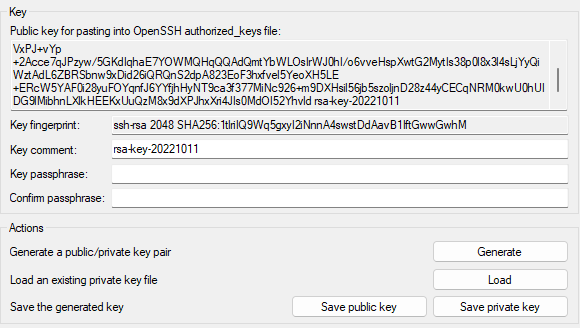
## Windows vers Debian

Objectif : instaurer une connexion

Depuis PuTTYGen

Générer une clé publique sur le poste Windows

Sauvegarder cette clé



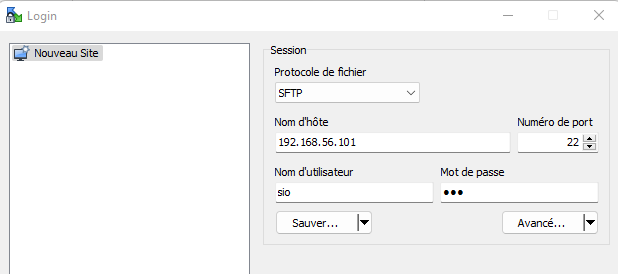
Le fichier généré **doit** être enregistré en .PUB

Se connecter en sio sur SRV-HOME

Entrer la commande ssh-keygen

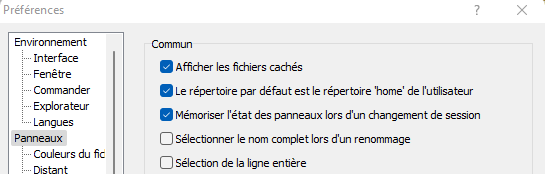


Se connecter au compte sio de SRV-HOME avec WinSCP



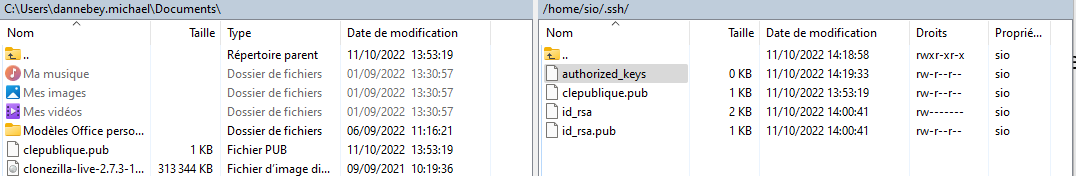
On se rend dans le répertoire **/home/sio/.ssh**

Pour afficher les fichiers cacher se rendre dans **options/préférences**

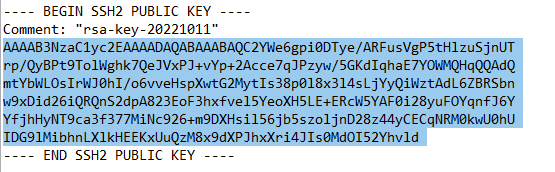


Sur la machine SRV-HOME, créer le fichier authorized\_keys dans **/home/sio/.ssh** qui permettra de stocker les clés publiques avec la commande **touch authorized\_keys**

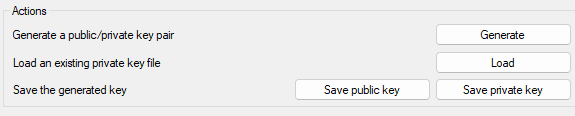
On transfert la clé publique .pub de la machine Windows vers ce répertoire



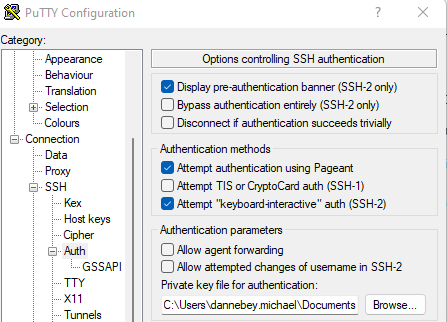
On ouvre le fichier .pub et on copie la clé dans le fichier authorized\_keys



On sauvegarde la clé privée en .ppk



On la renseigne dans PuTTY



On configure la session

