**Guide de l’utilisateur.**

Projet LINUX

Équipe : Kevin Clovel, Bruno de Santana, Mario Camelo.

Prof : William Herrera.

Indice du projet :

1. Installez deux systèmes d’exploitation Linux sur OracleVM.
2. Communiquez les deux machines (Serveur et Client), à l’aide de deux adaptateurs réseau.
3. Créez l’utilisateur « projet ».
4. Générez une clé SSH sur le serveur pour l’utilisateur « projet ».
5. Copiez la clé publique sur la machine « client ».
6. Validez qu’il y a bien communication entre le serveur et le client en transmettant le fichier/home/projet/test.txt
7. Créez le menu suivant sous Linux :

Générer le fichier à copier.

Valider le type de fichier généré.

Générer le cksum à partir du fichier.

Transmettre le fichier du client au serveur.

\*Valider le cksum à partir du fichier du serveur.

Journal de transmission.

1. **Installez deux systèmes d’exploitation Linux sur OracleVM.**

**Prérequis :**

Télécharger et installé le « ORACLE VM VIRTUALBOX »

Lien - <https://www.virtualbox.org/wiki/Downloads>

Télécharger l’IMAGE ISO Ubuntu 20.04

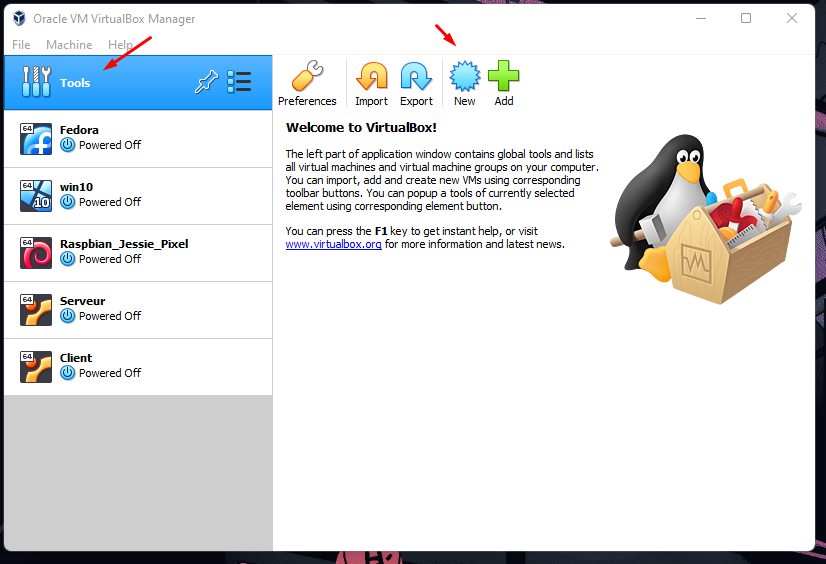
Lien <https://ubuntu.com/download/>

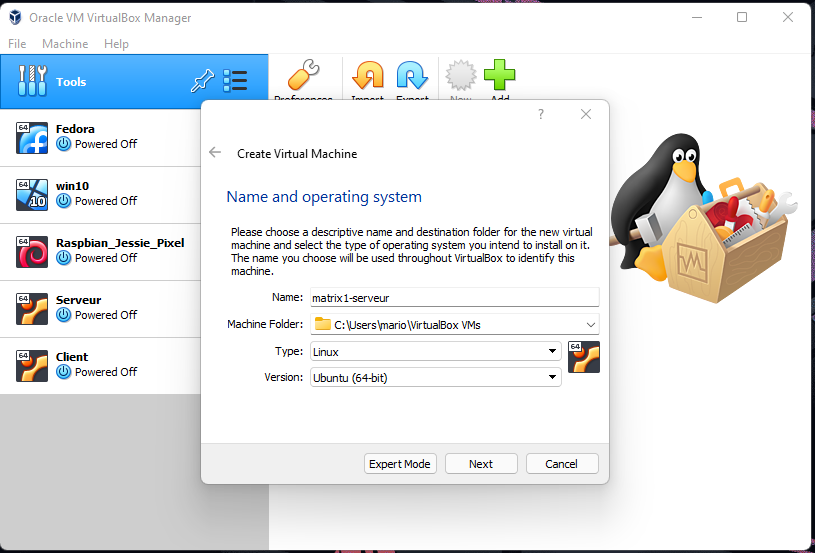
* Serveur Linux – Ubuntu
* Nom : matrix1-serveur
* Matériel : 3Gb RAM ; Disque 20GB
* Client Linux – Ubuntu
* Nom : matrix1-client
* Matériel : 1Gb RAM ; Disque 10GB

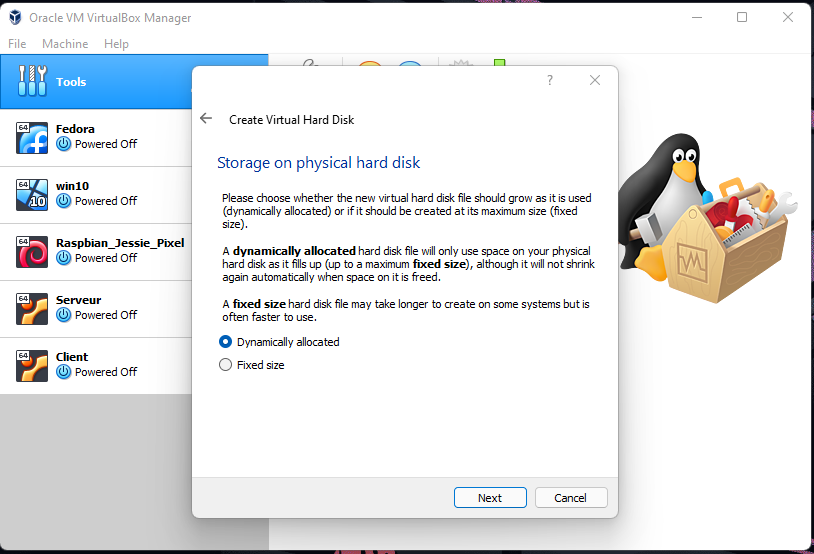
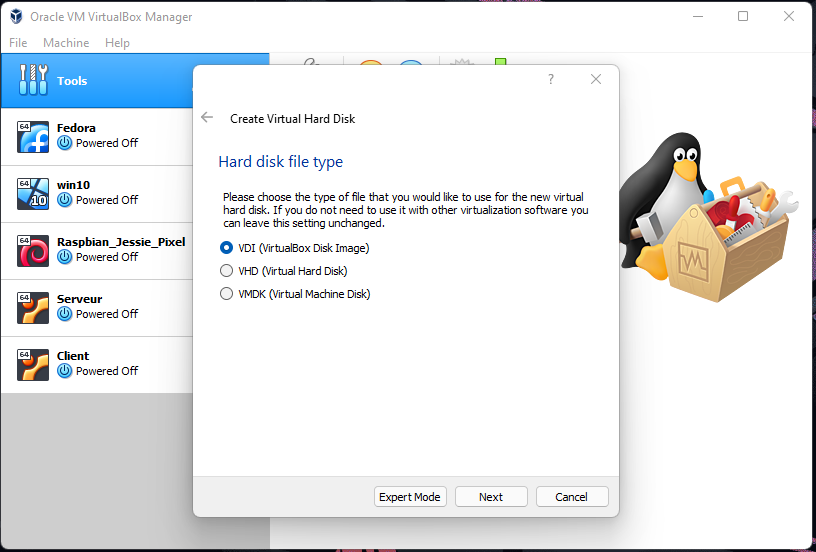
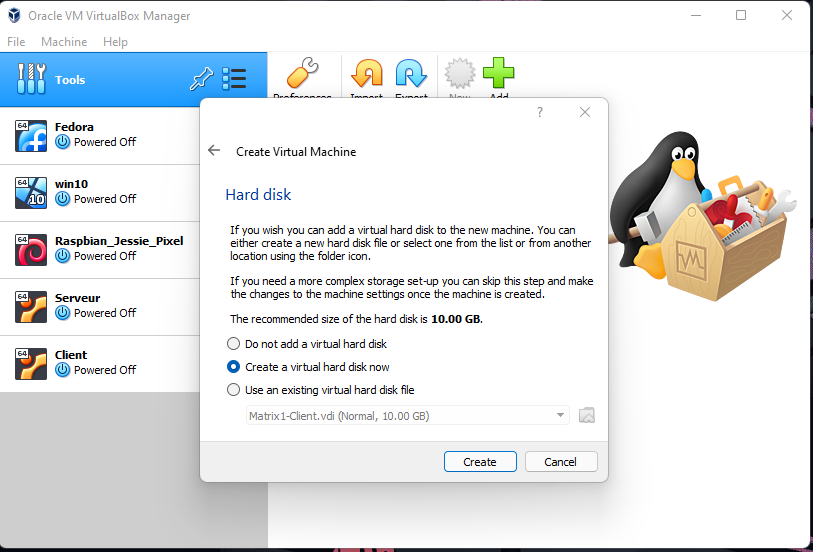
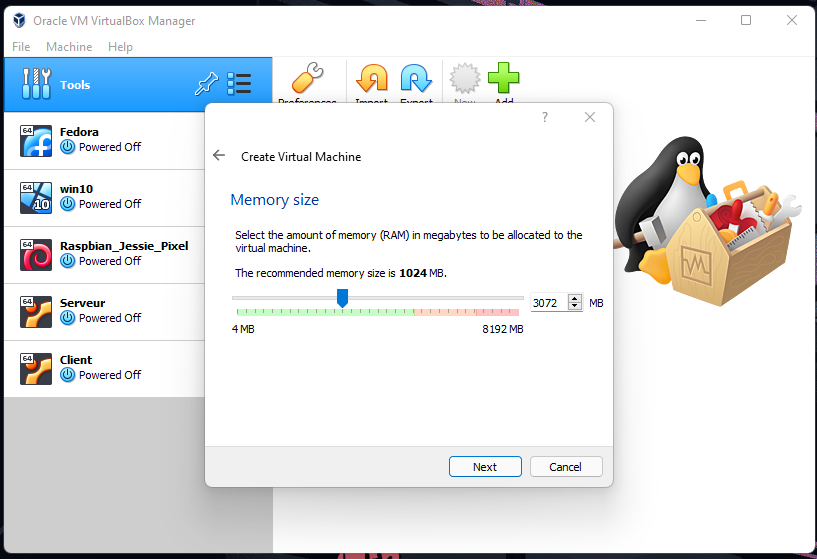
**Serveur :**

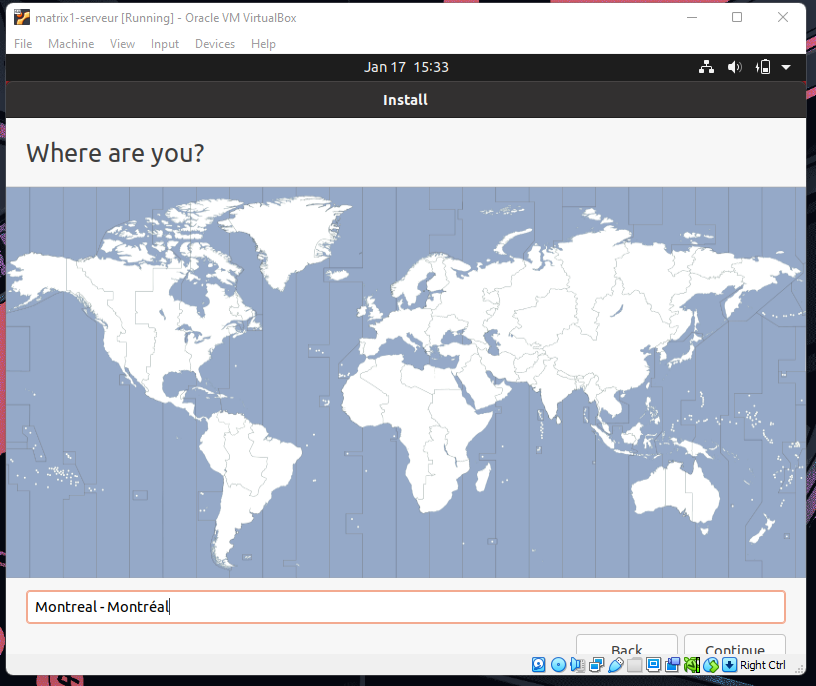
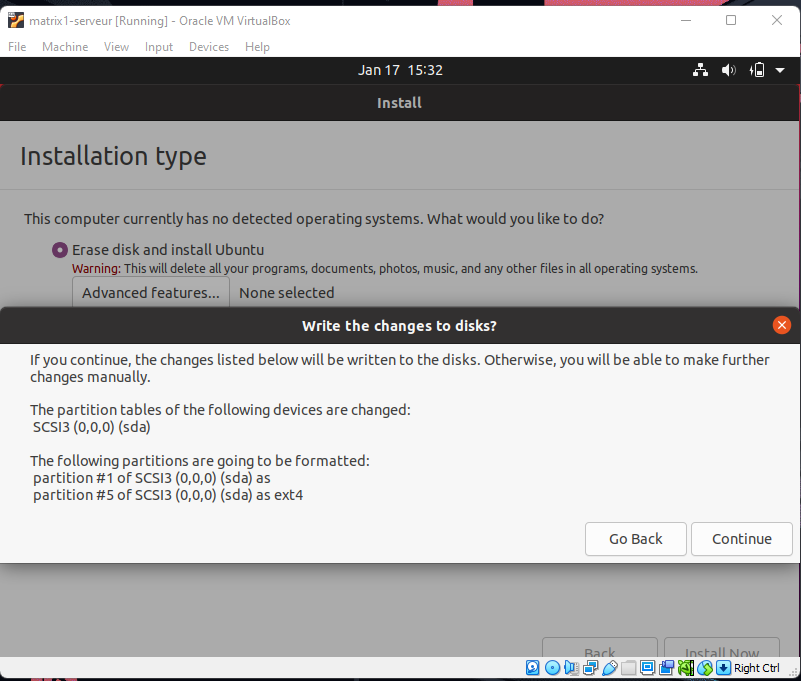
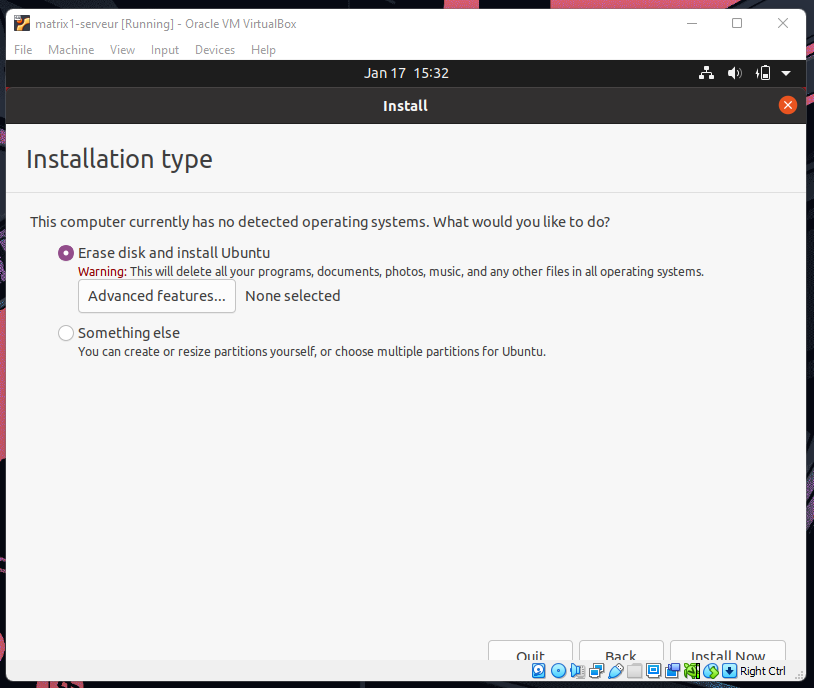
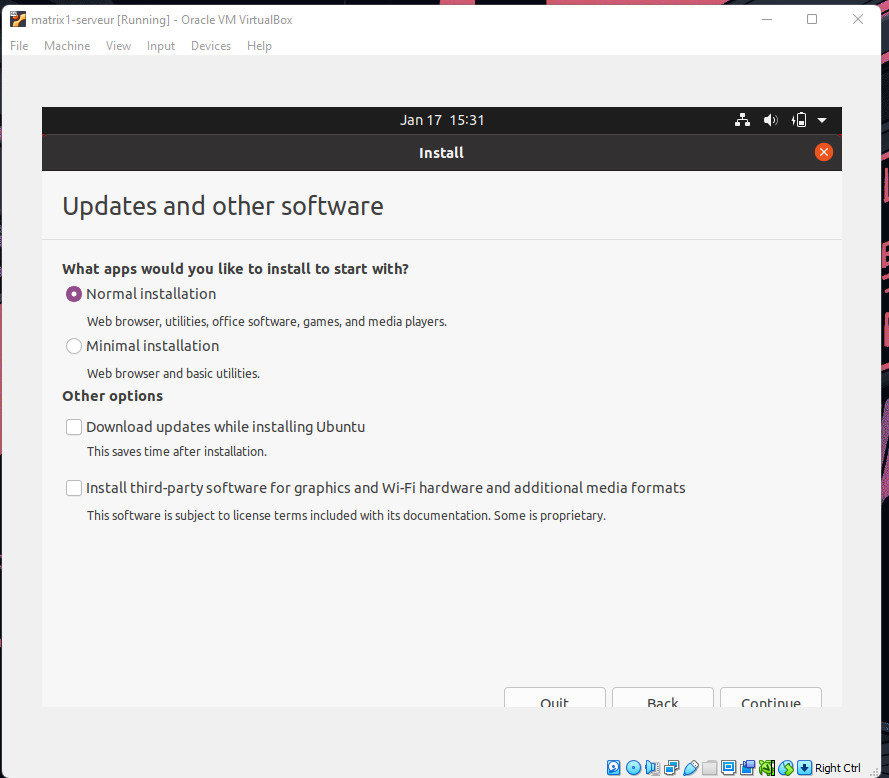
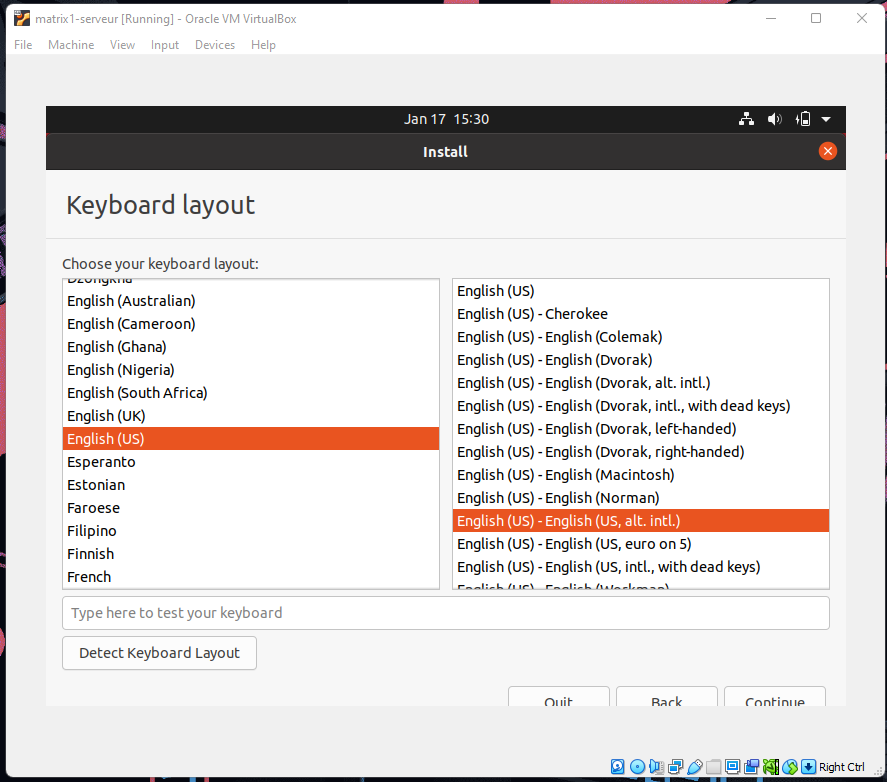
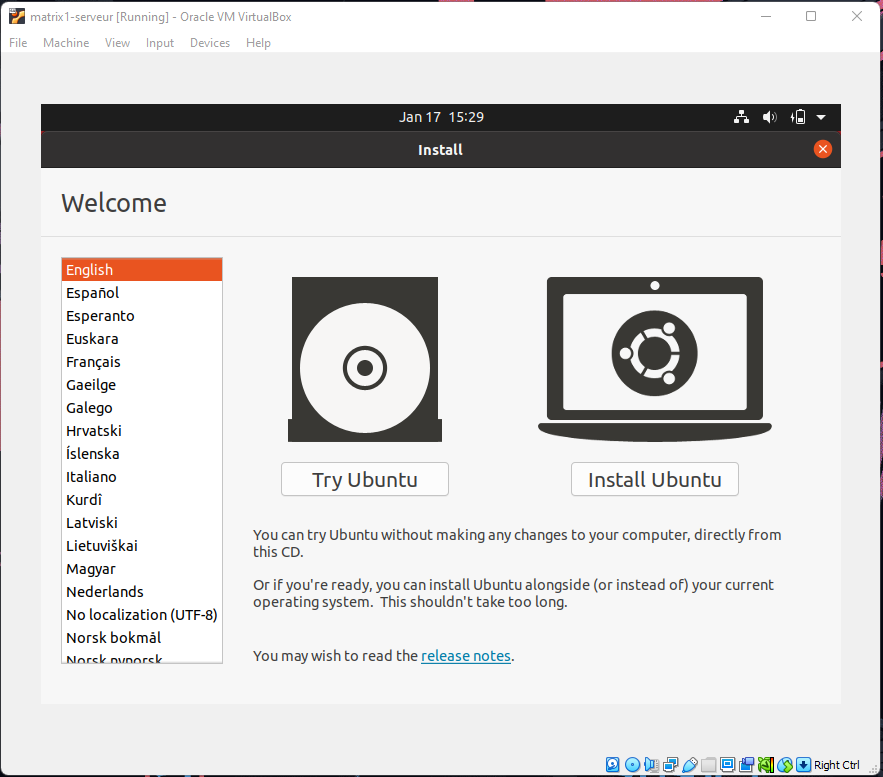
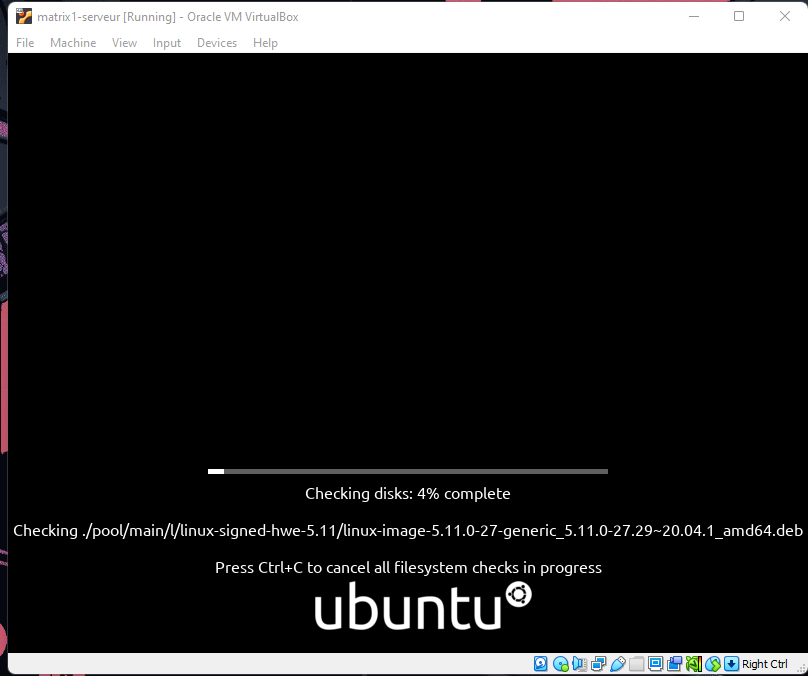
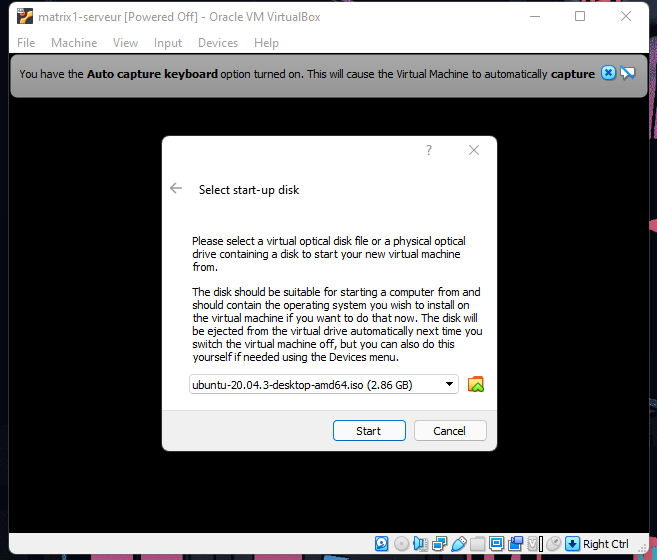
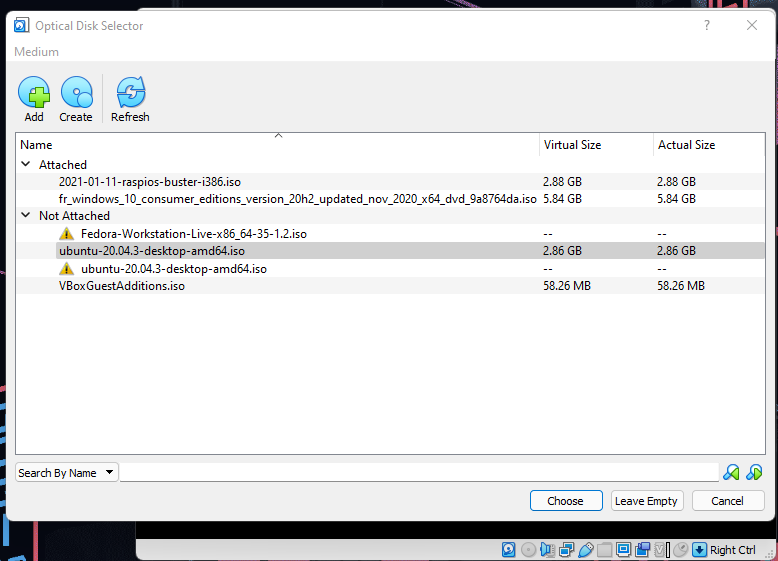
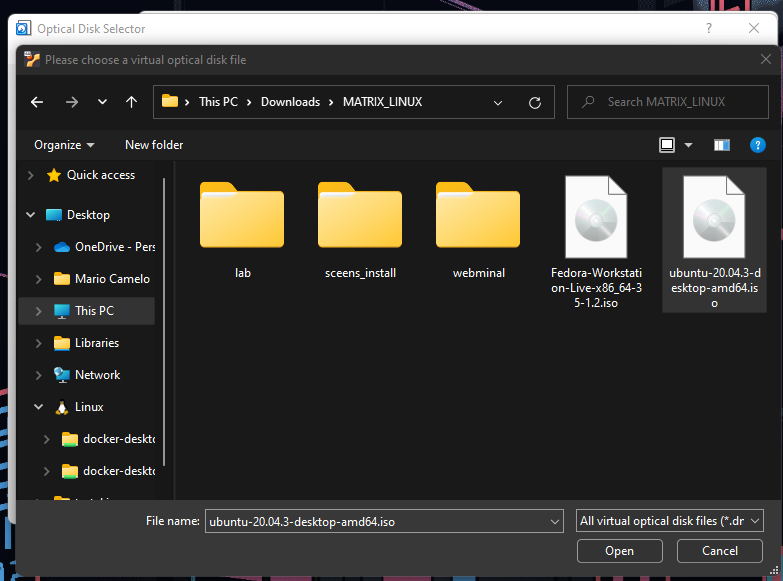
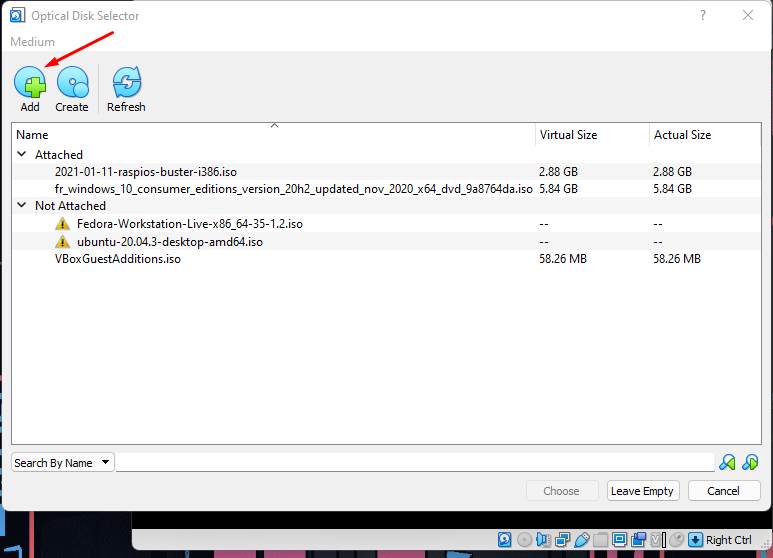
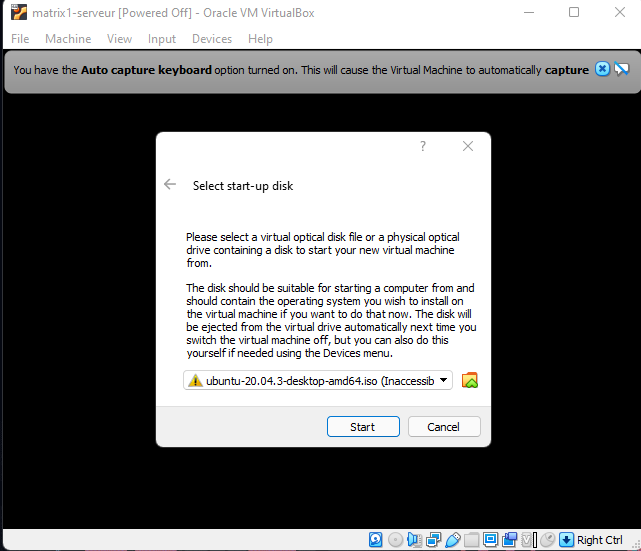
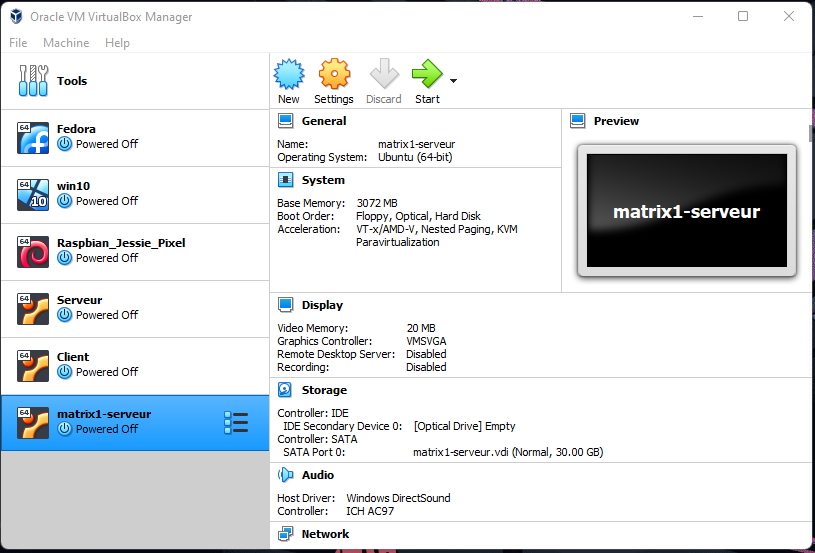
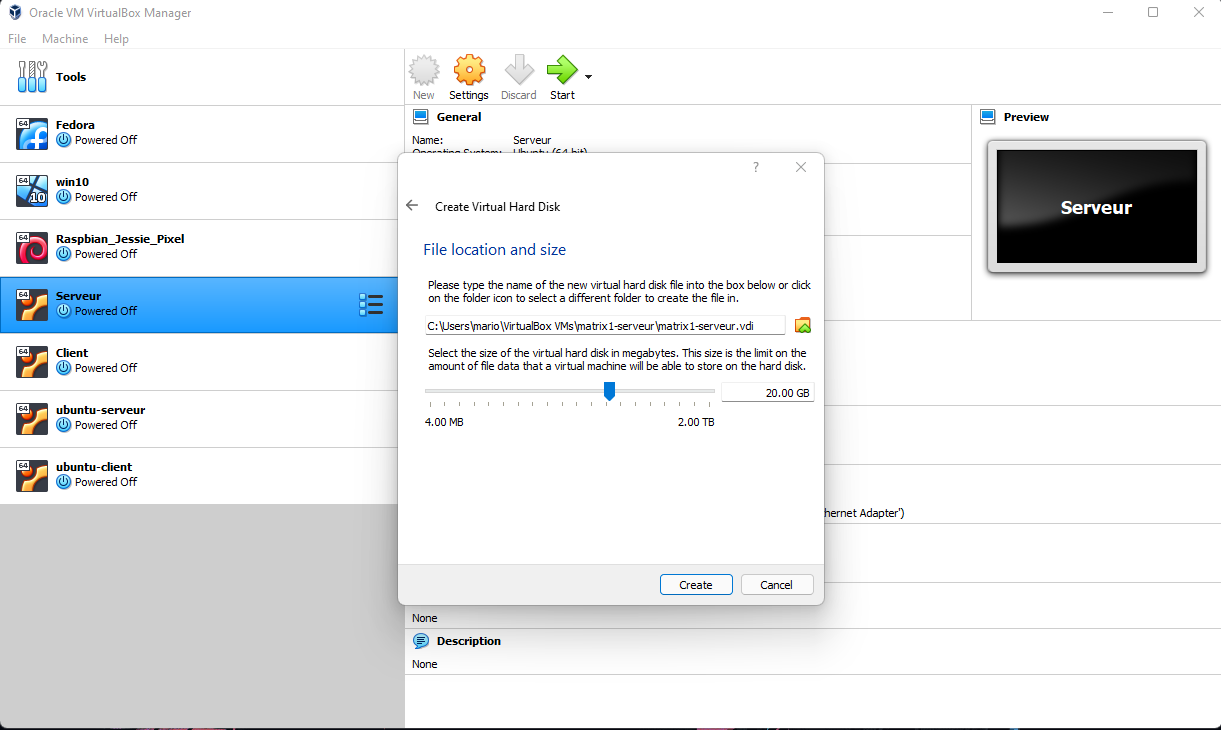
Ici on vas faire le même procediment du « serveur », choisissez les propriétés que nous avez été demandé, comme le nom de la machine, mémoire ram, mémoire du disque, le nom de l’utilisateur.

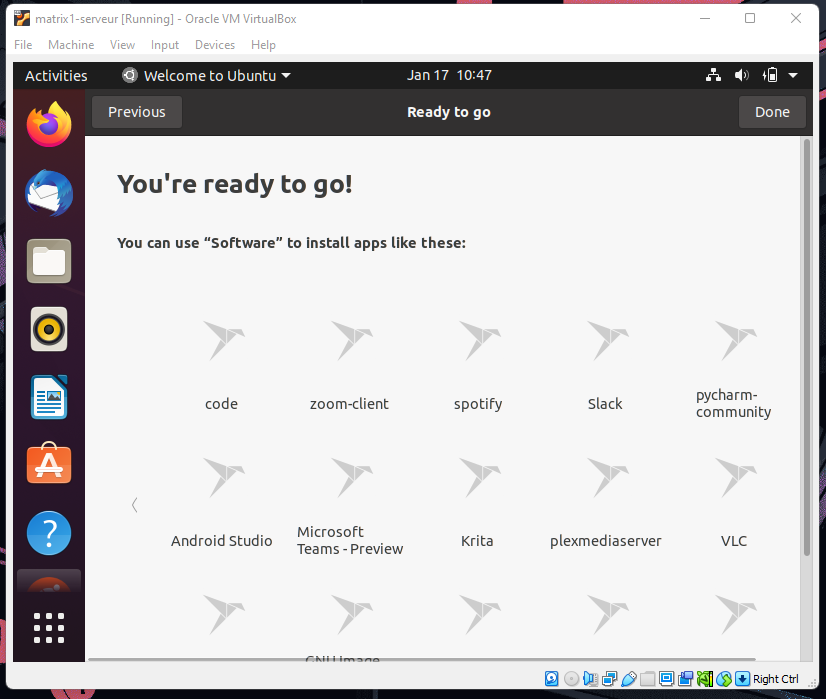
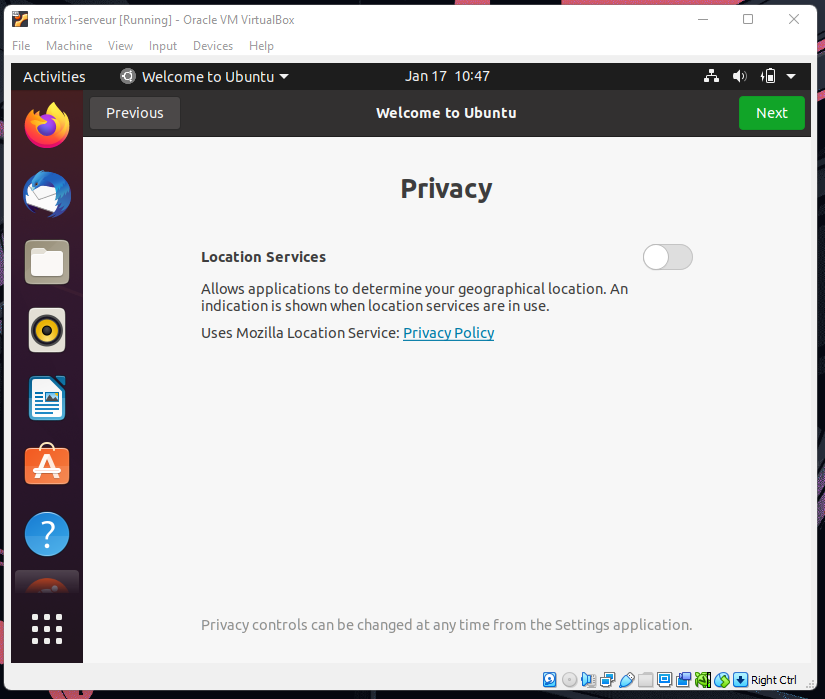
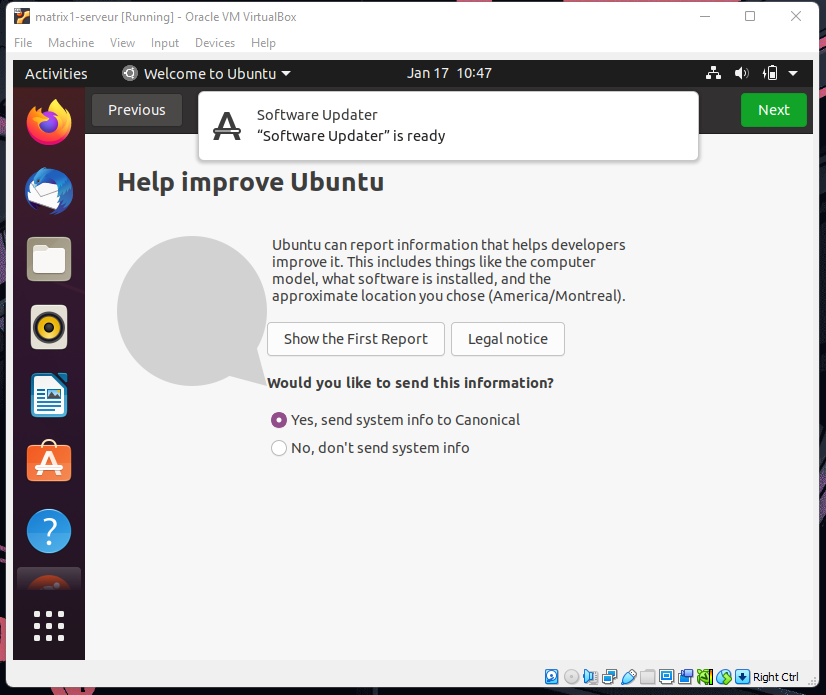
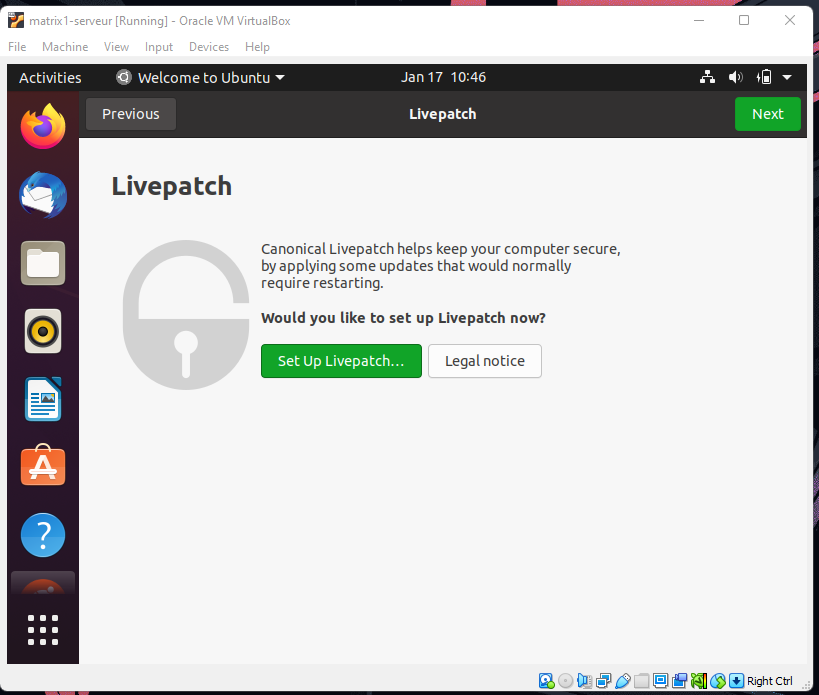
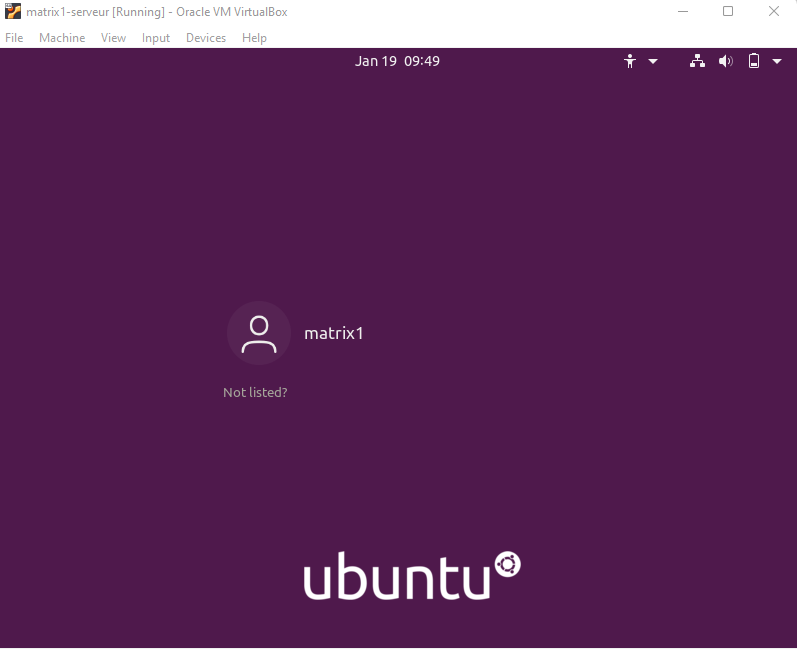
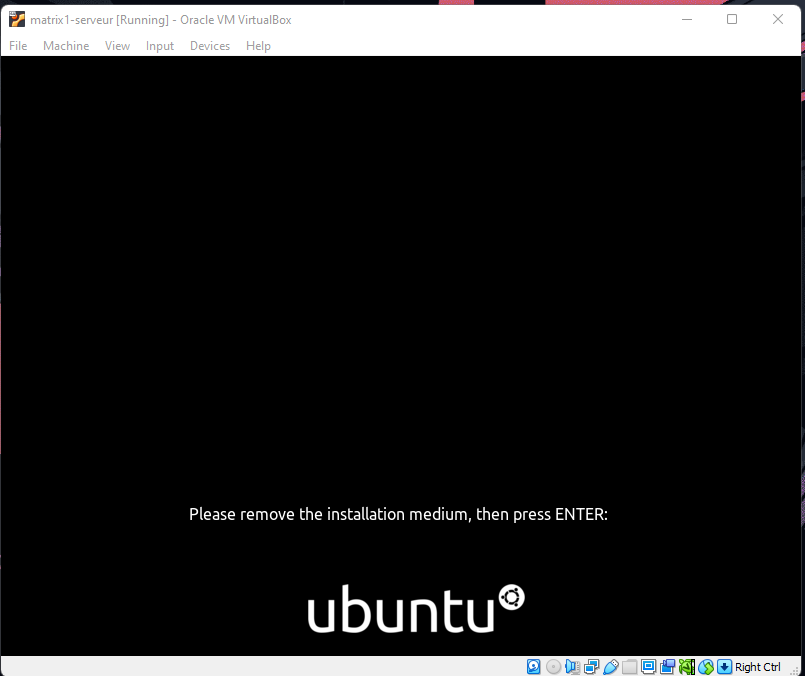
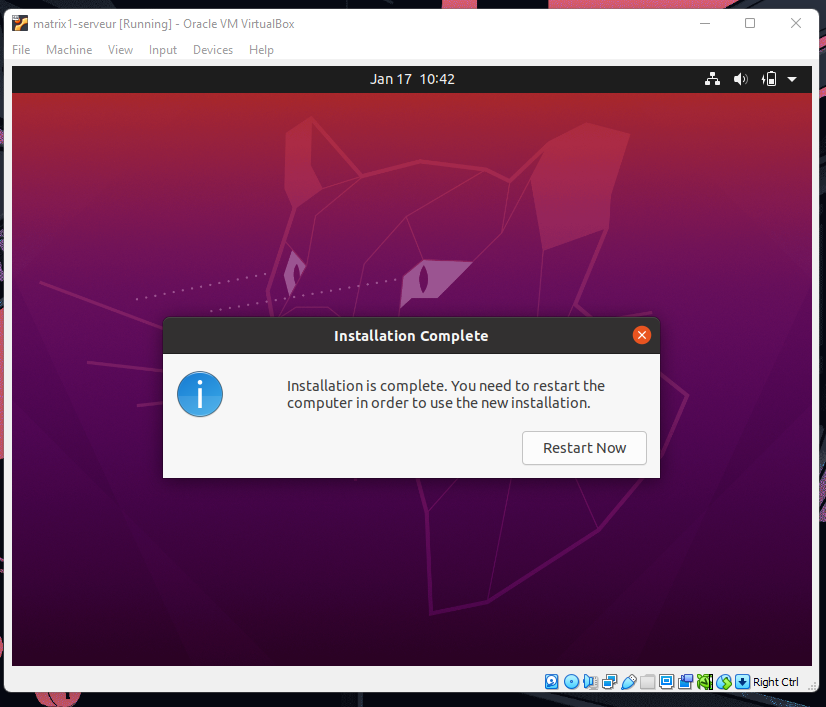
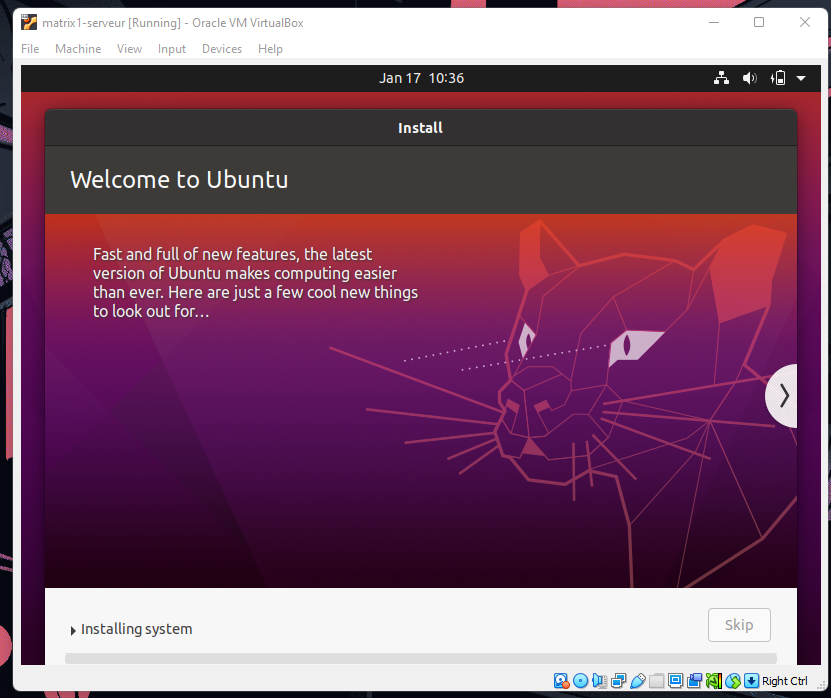
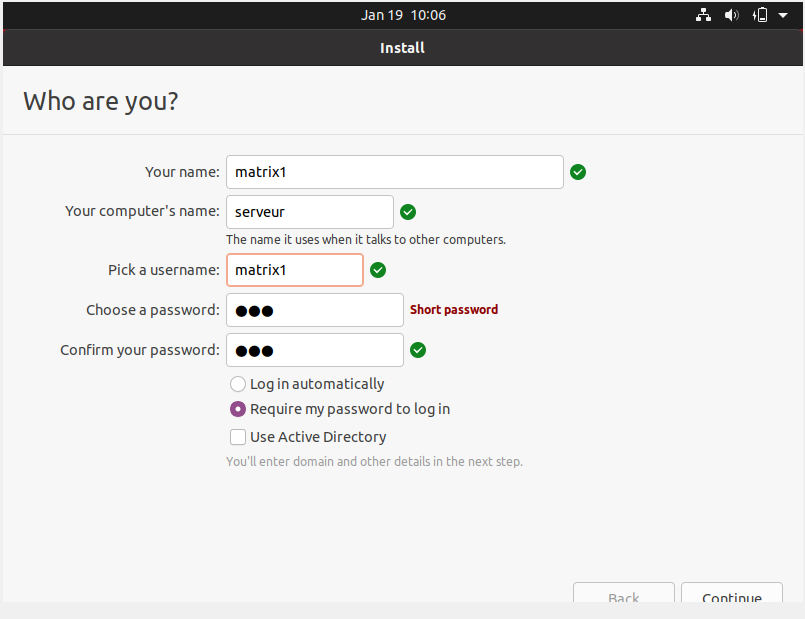
* Serveur Linux – Ubuntu
* Nom : matrix1-serveur
* Matériel : 3Gb RAM ; Disque 20GB





**Choisissez 3GB de mémoire RAM, pour le machine virtuelle « matrix1-serveur ».**

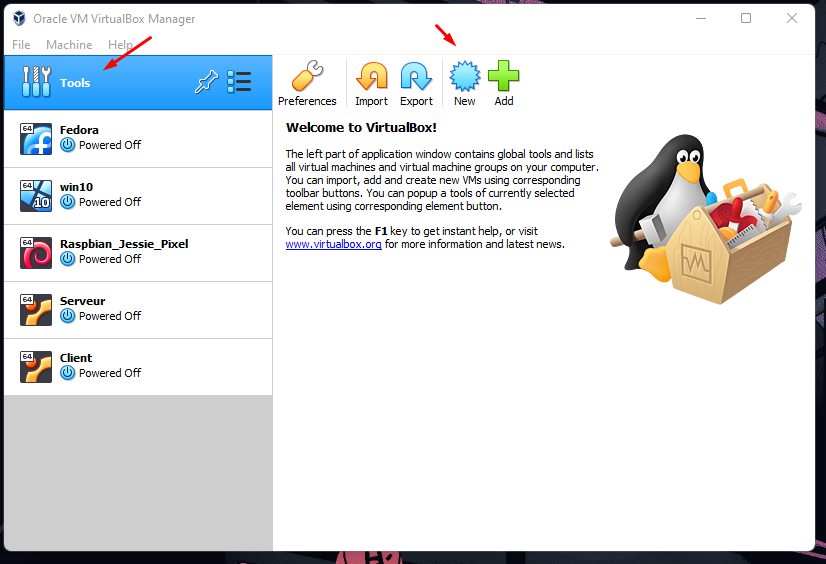
**Choisissez 20.GB de mémoire DISK, pour le machine virtuelle « matrix1-serveur ».**

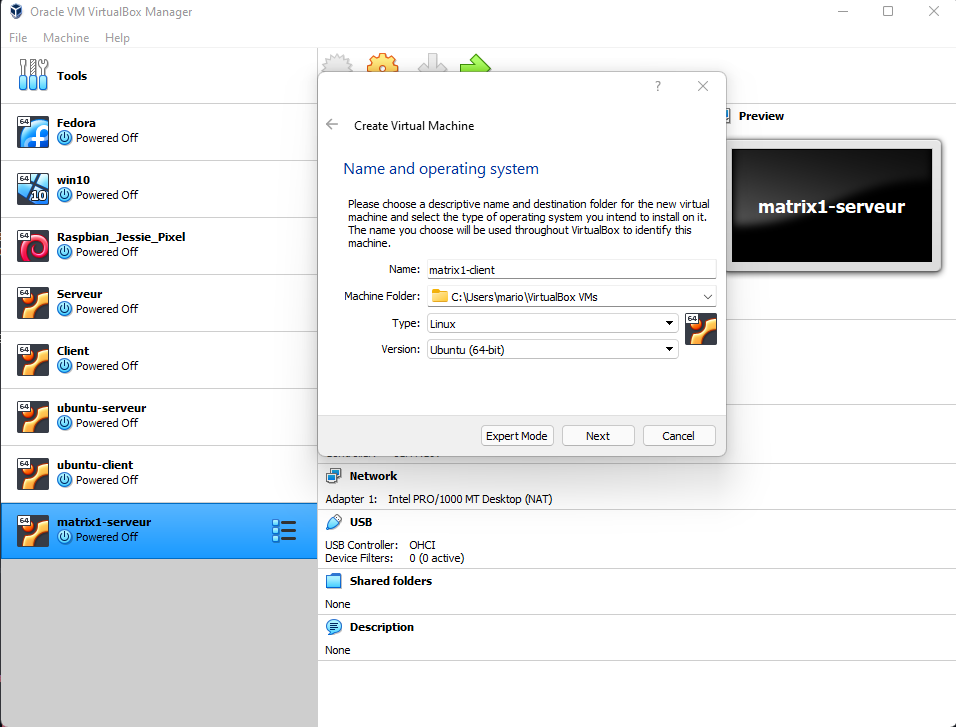


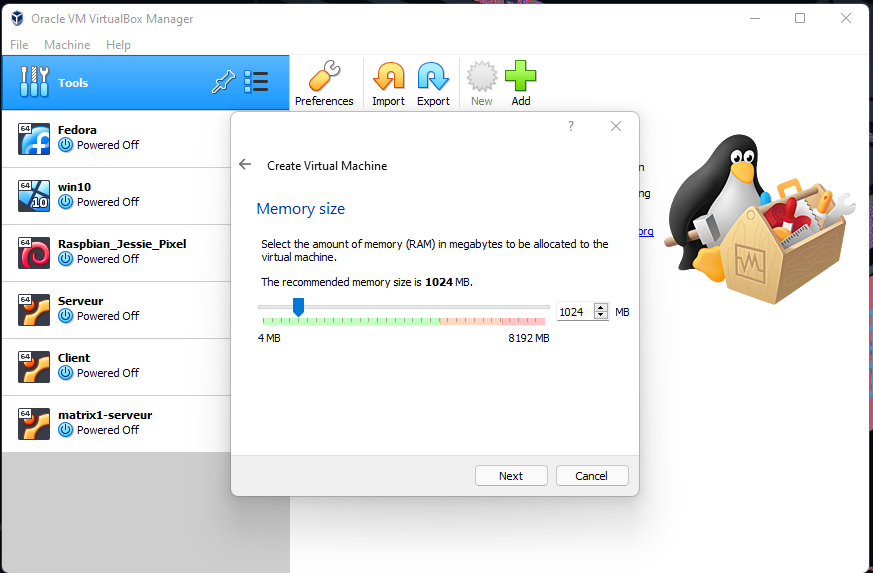
**Client :**

Ici on vas faire le même procediment du « serveur », mais maintenant choisissez les propriétés du « client » que nous avez été demandé, comme le nom de la machine, mémoire ram, mémoire du disque, le nom de l’utilisateur.

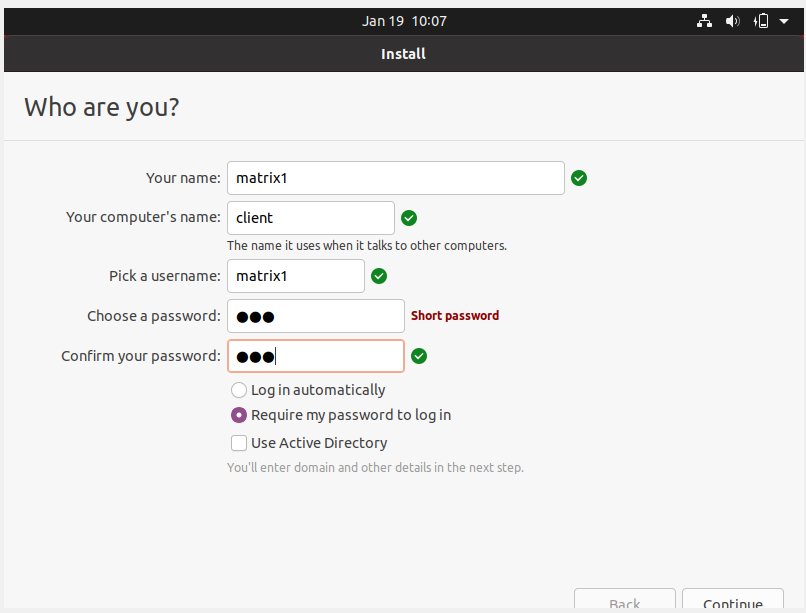
* Client Linux – Ubuntu
* Nom : matrix1-client
* Matériel : 1Gb RAM ; Disque 10GB



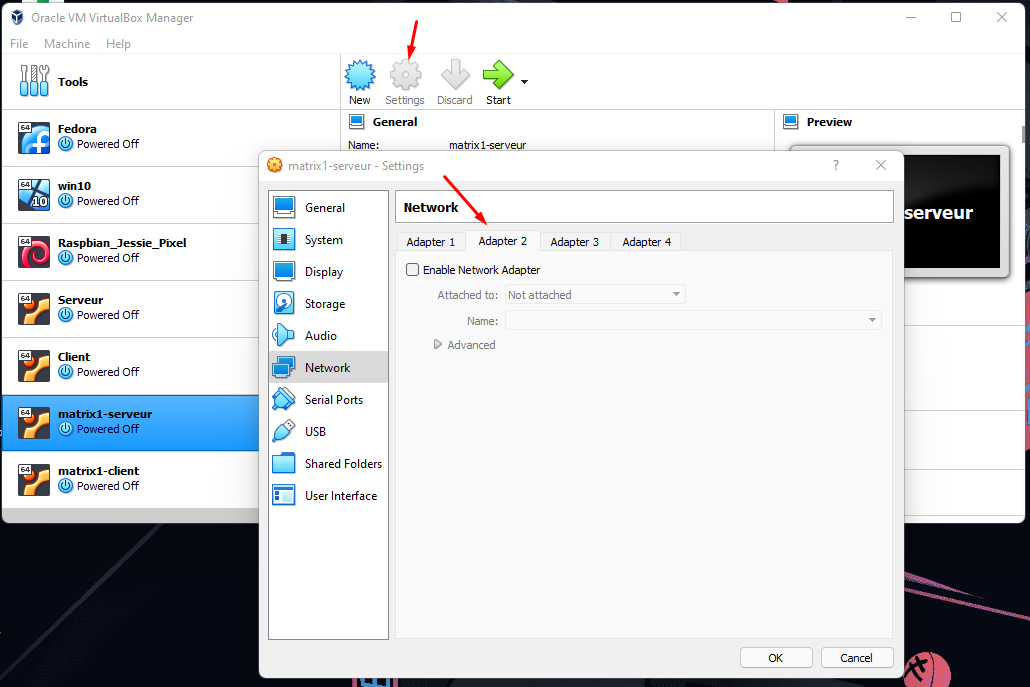
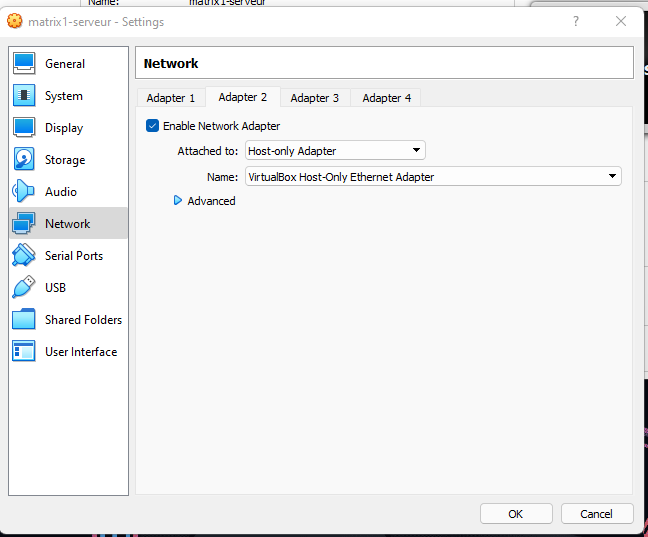


**Choisissez 1GB de mémoire RAM, pour le virtual machine matrix1-client.**

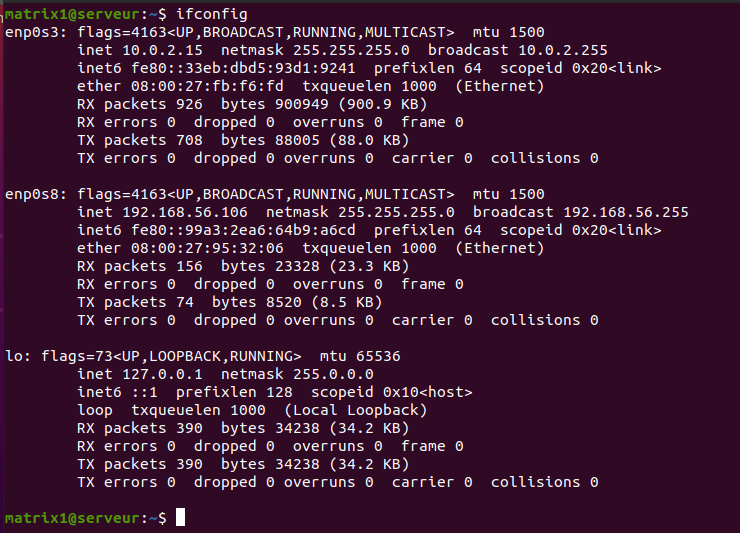
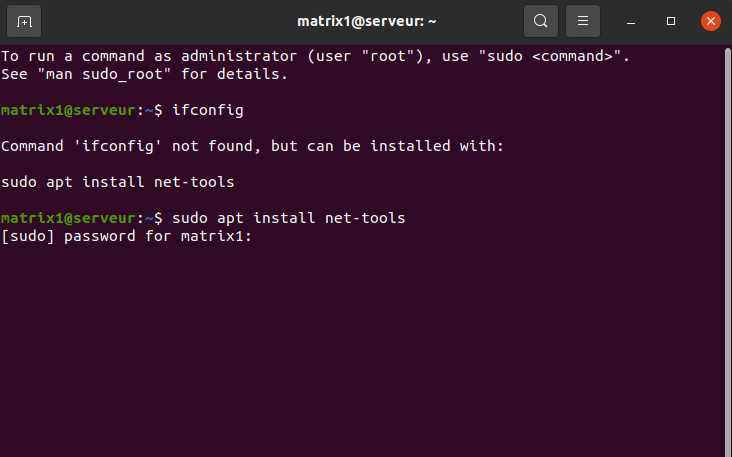
**Choisissez 10GB de mémoire DISK, pour le virtual machine matrix1-client.**

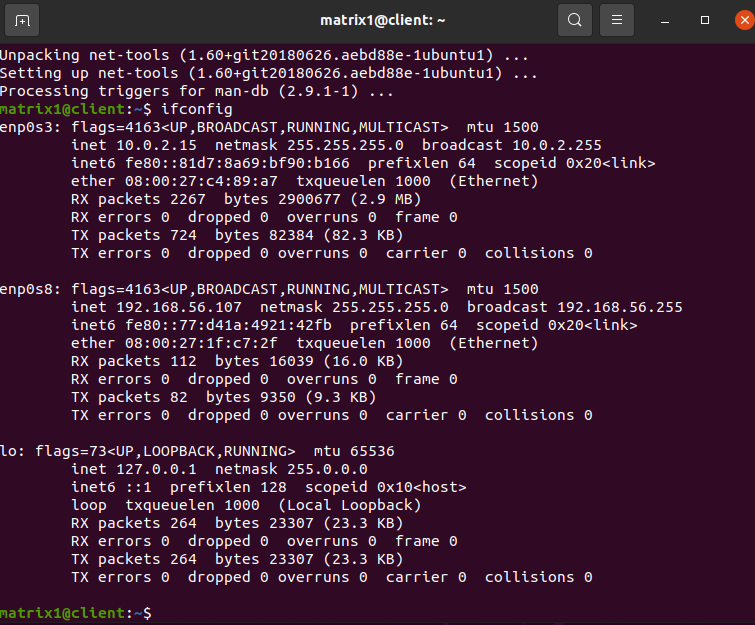


**2 – Communiquez les deux machines (Serveur et Client), à l’aide de deux adaptateurs réseau.**

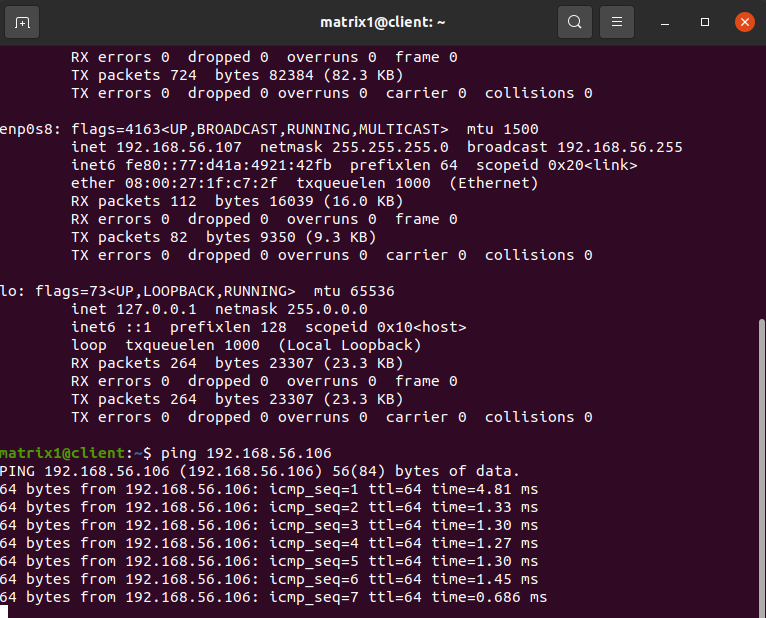
On va ajouter une carte de réseau sur les machines virtuelles, afin d’elles se communiquent. Cliquez sur « SETTINGS », après vous allez au « Network ».  Cliquez sur option « ADAPTER 2 » et marquez la boite « ENABLE NETWORK ADPTER ». Finalement, choisissez « HOST ONLY ADAPTER ».

Pour validez la bonne installation, on peut utiliser le command « ifconfig » dedans le TERMINAL SHELL.





Pour validez la bonne communication entre les deux, on peut utiliser le command « ping <IP> ».

****

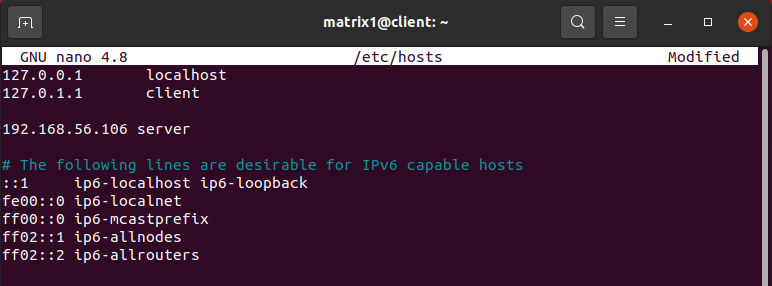
On va créer les variables de hosts, afin d’être plus simples se souvins de l’IP adress.

**#CLIENT**

**sudo pico /etc/hosts**

**#add <ip-server> <nickName>**

**192.168.56.106 server**

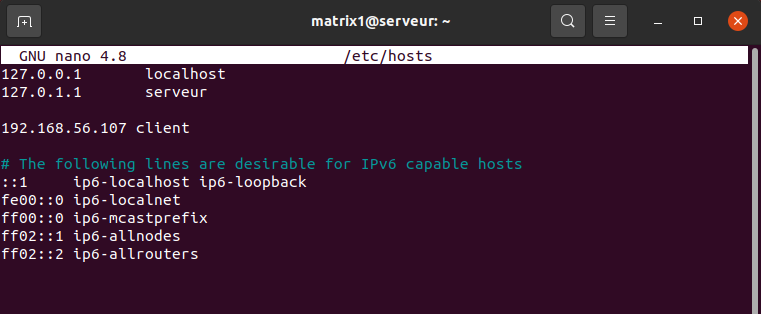
****

**#SERVER**

**sudo pico /etc/hosts**

**#add <ip-client> <nickName>**

**192.168.56.105 client**

****

**Confirmer avec le command: ping <nickName>**

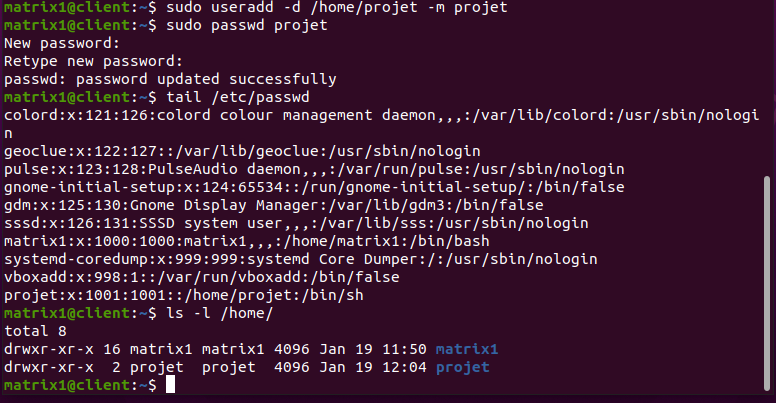
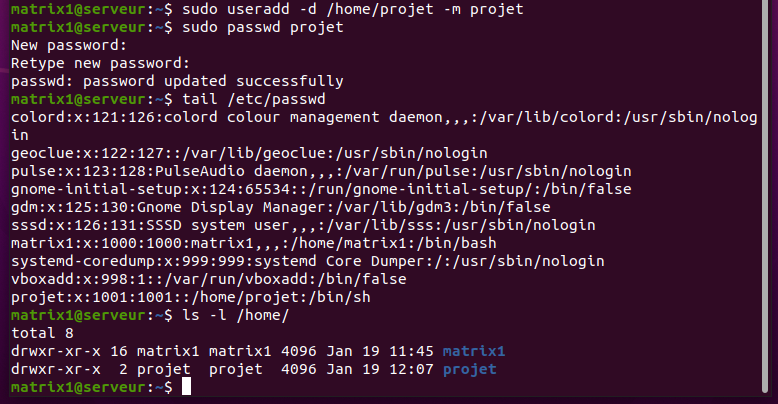
**Ex:**

**ping client**

**ping server**

**3. Créez l’utilisateur « projet ».**

On va une nouvelle l’utilisateur. Command « useradd ». Créer le mot de passe avec « passwd ».

****

**4. Générez une clé SSH sur le serveur pour l’utilisateur du projet.**

**#Prérequis:**

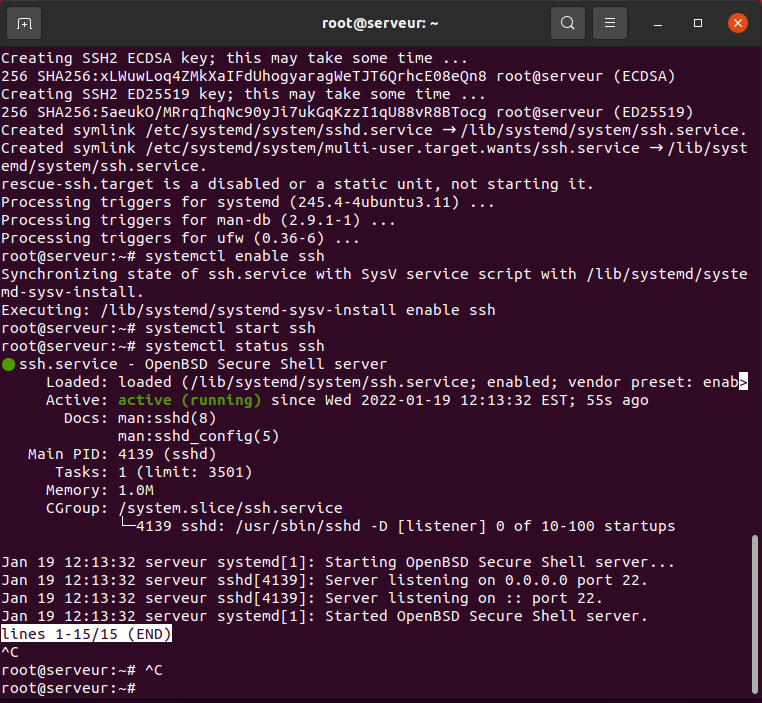
**sudo su –**

**apt install openssh-server**

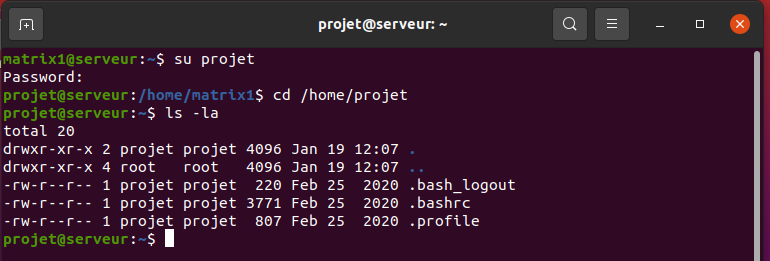
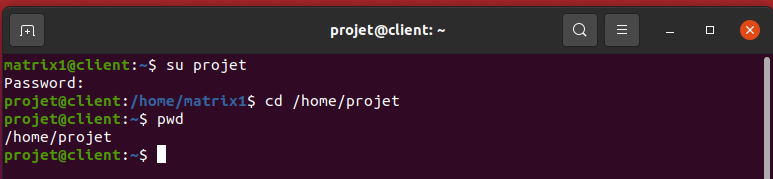
**systemctl enable ssh**

**systemctl start ssh**

**systemctl status ssh**

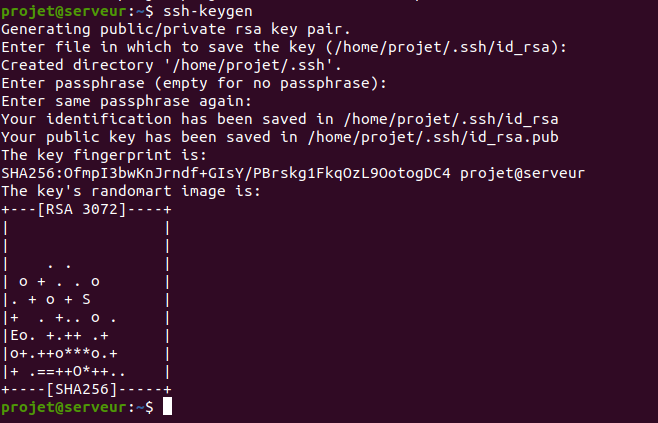


Utilisez le commando « exit », pour sortir de « root ». Après roule le command « su projet » pour utiliser user projet.



**ssh-keygen**

**#Entrez le mot de passe.**

****

**5. Copiez la clé publique sur le client.**

Après la création de la clé publique, sur le couté du serveur, on vas le copié vers le client. Après la transmission de la clé sur le client, on peut voir le fichier, « authorized\_keys »

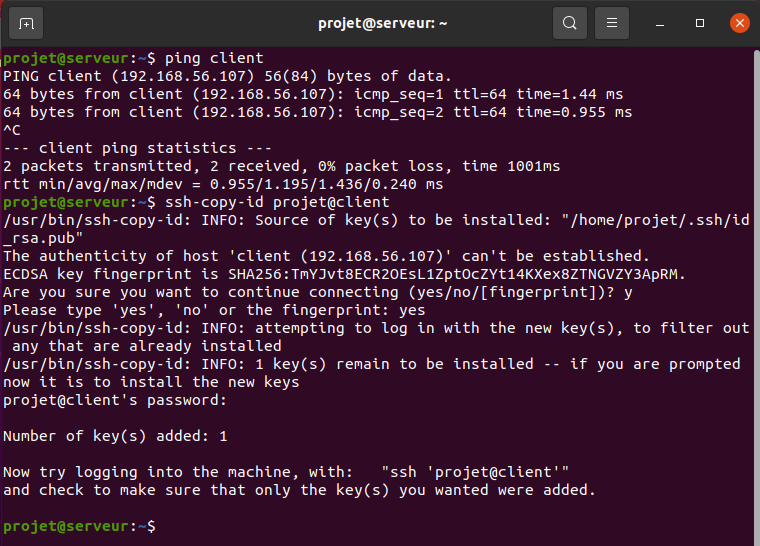
**#SERVEUR**

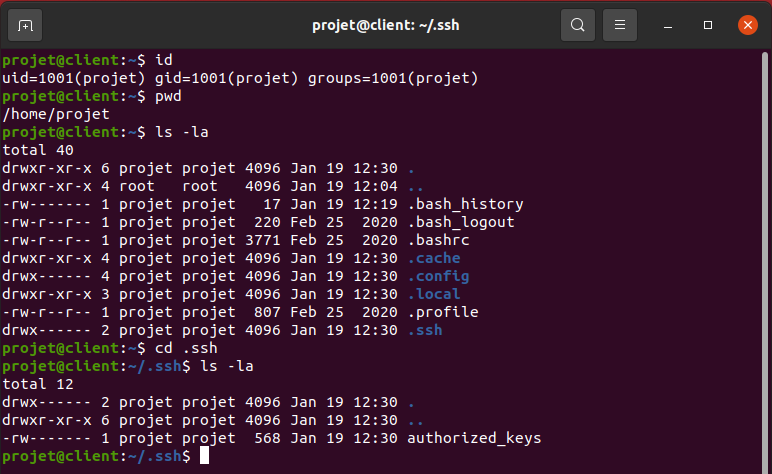
**#ssh-copy-id <user-client>@<ip-client>**

**ssh-copy-id projet@client**

**# On peut utilizer nom client, parce que on fait l’édition sur /etc/hosts , ou**

**ssh-copy-id projet@192.168.56.107**





**6. Validez qu`il y a bien communication entre le serveur et le client en transmettant le fichier /home/projet/test.txt du client au serveur.**

**Prérequis:**

**###SERVEUR**

**#Sortir d’utilisateur « projet », avec command « exit ».**

**sudo apt install vsftpd -y**

**systemctl start vsftpd**

**#Logger avec « projet » : su projet | cd /home/projet**

**###CLIENT**

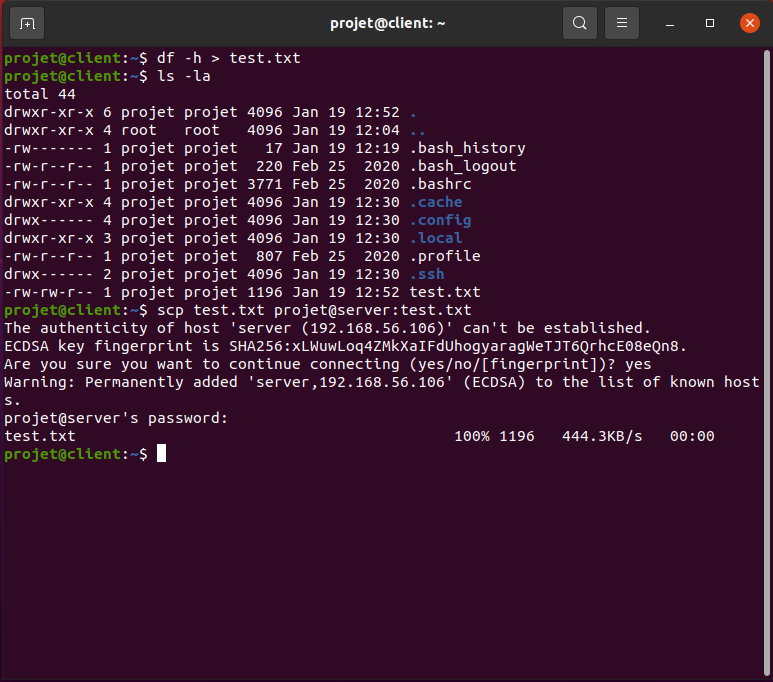
**#Creer un fichier.**

**df -h > test.txt**

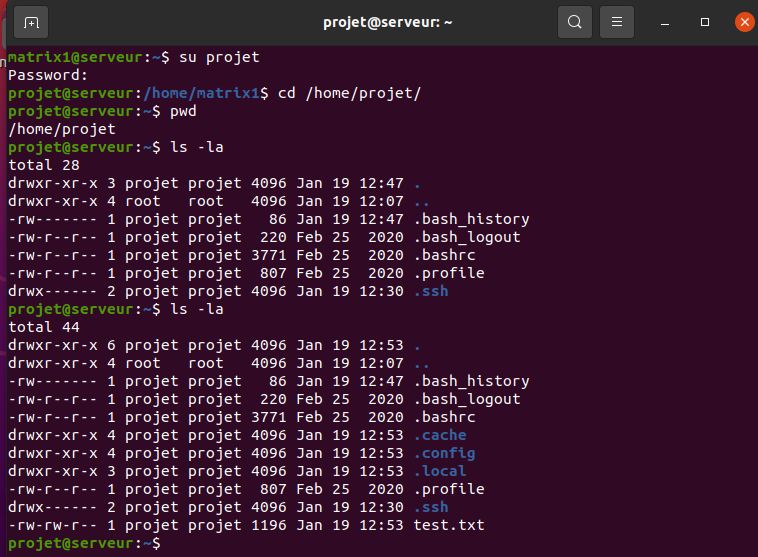
**#Envoiez le fichier avec le command « scp ».**

**#Ex: scp <nomFichier> <nomUtilisateur>@<ipServer> :<nouveuNomeFichier>**

**scp test.txt projet@server:test.txt**

****

On peut voir et validé le fichier qui été transféré.

****

**7. Script Main.sh**