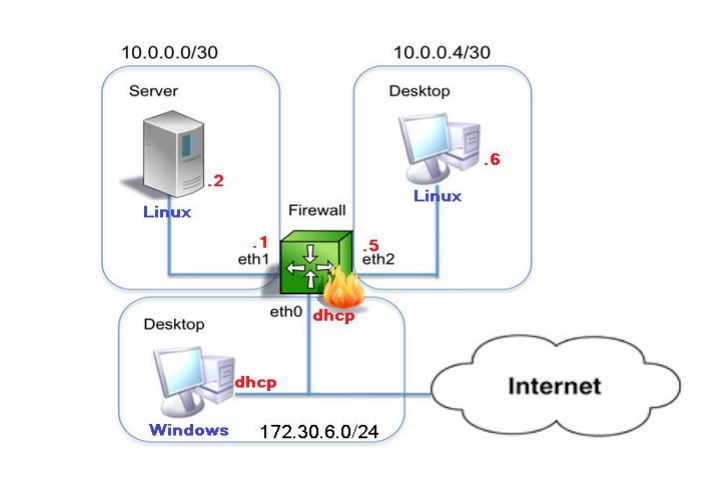
**Fiche de travail 1 : installation de l’environnement de travail**

***Introduction :***

Effectuez des tests de ping et d’accès d’internet sur toutes les machines virtuelles en respectant cette topologie ci-dessous.

***Travail à effectuer :***

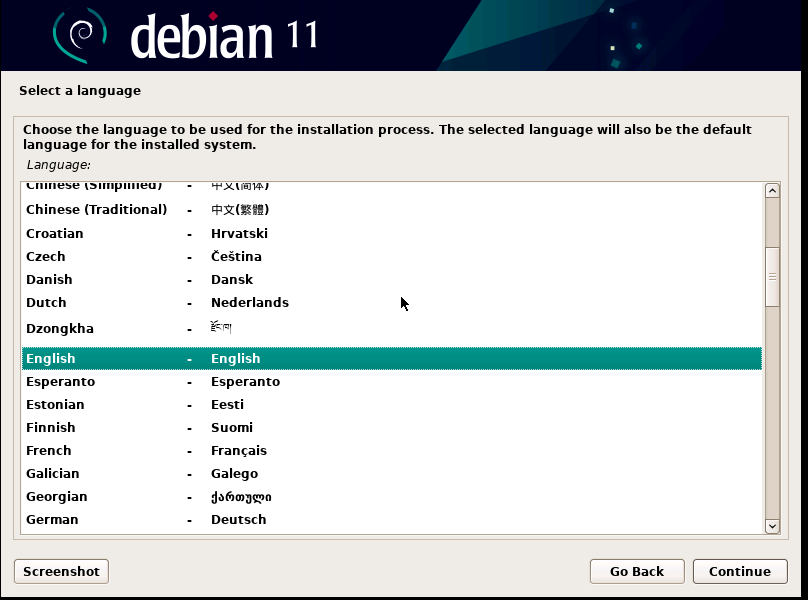


Installez 3 machines virtuelles avec Debian:

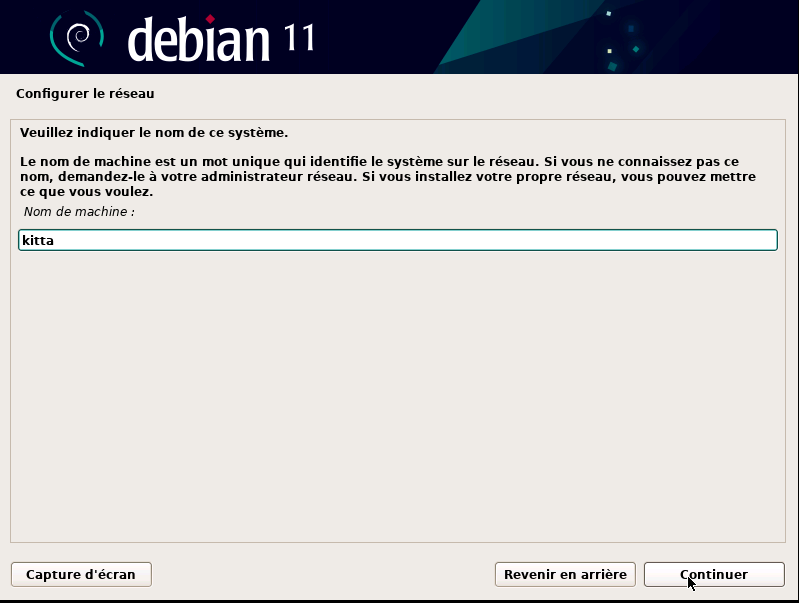
* Ici on devra choisir par une interface graphique ou soit interface serveur



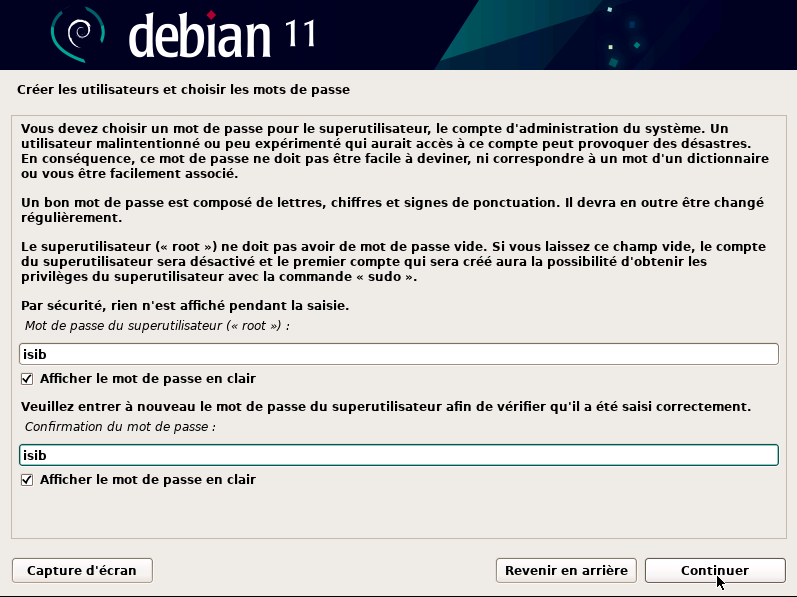
* Ici faudra choisir la langue



* Ici faut choisir le nom de la machine



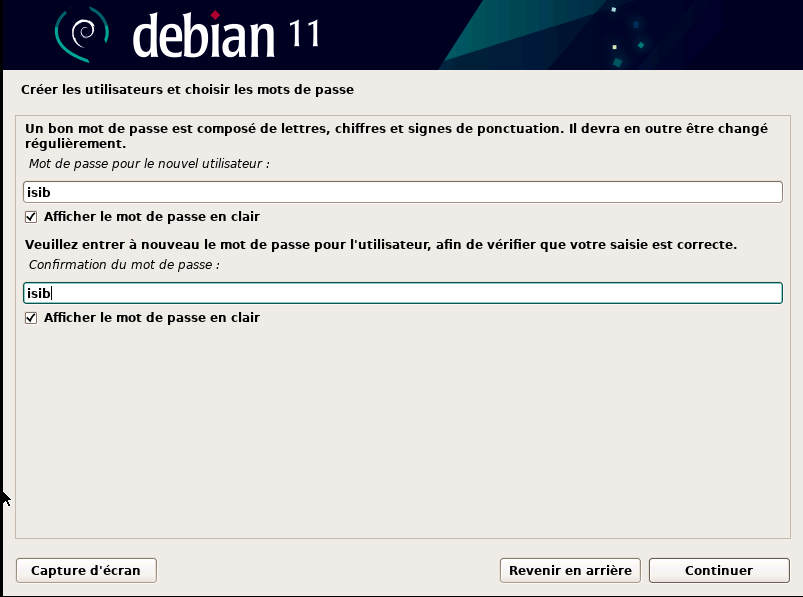
* Ici le mot de passe de la machine



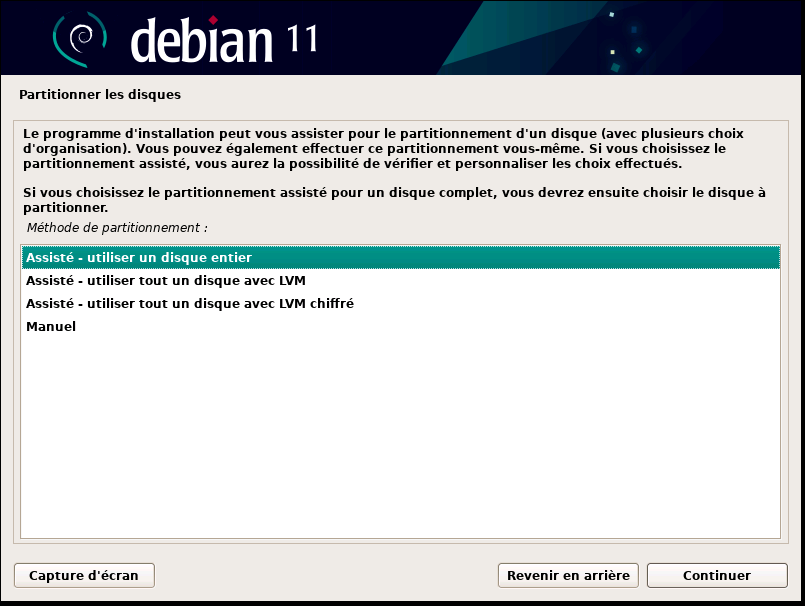
* Compte utilisateur



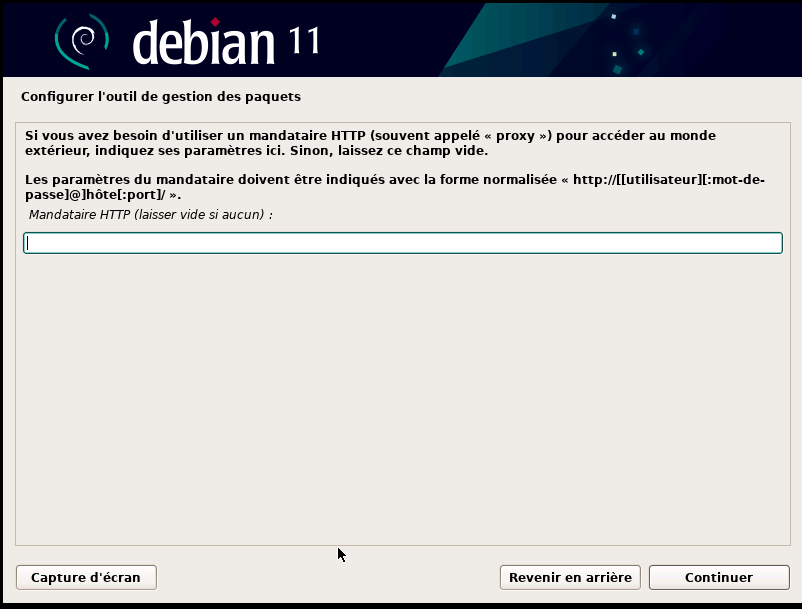
* Mot de passe du compte d’utilisateur



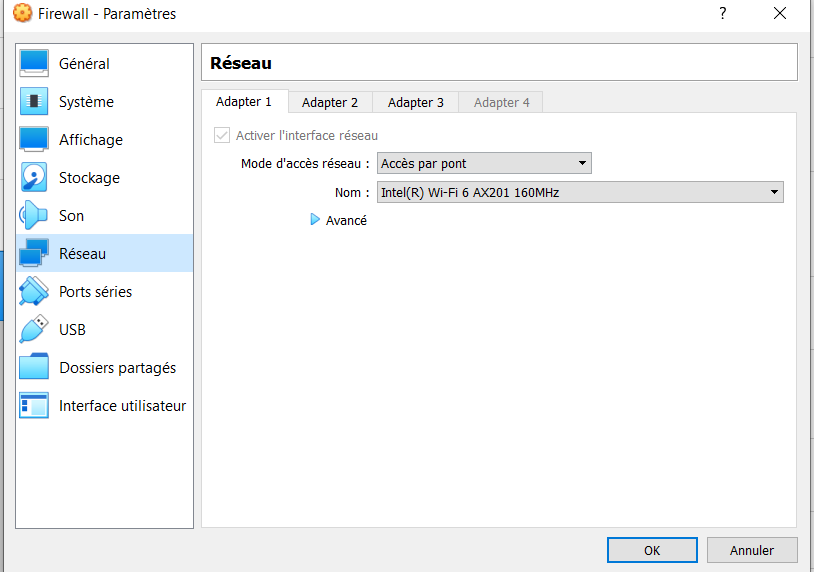
* Partitionnement du disque dur



* Ici on n'a pas besoin du service mandataire



Après faut mettre tous les cartes réseaux par **accès par pont**

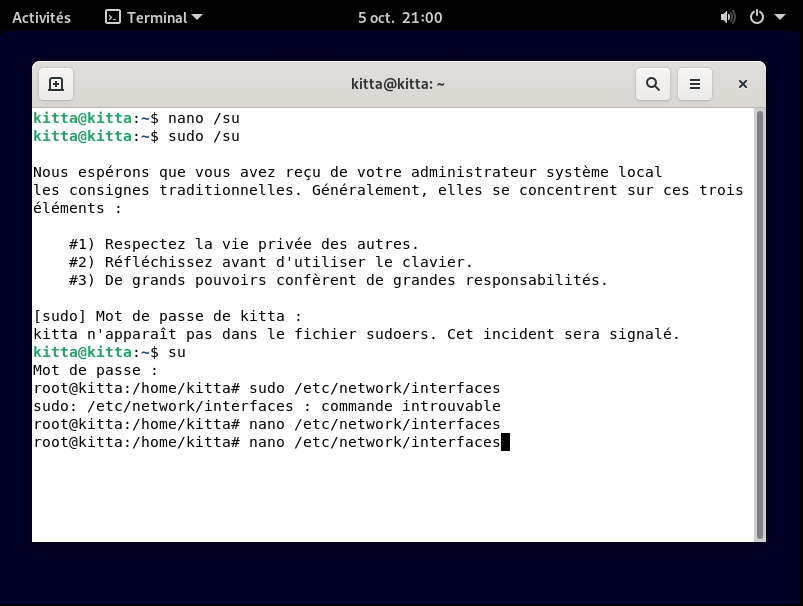


Après faut configurer les interfaces :

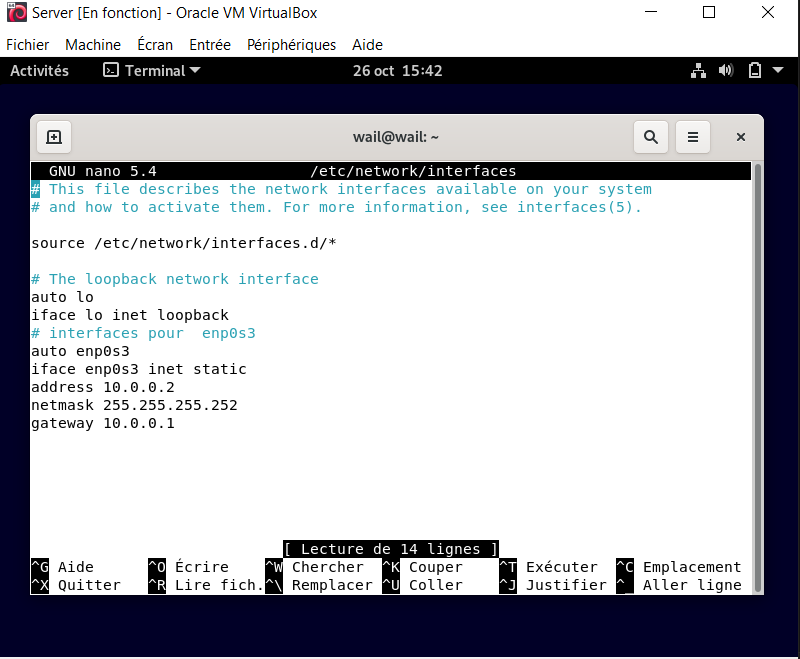
* On se dirige sur le terminal
* Puis on écrit la commande **su** ===>pour être en mode utilisateur.



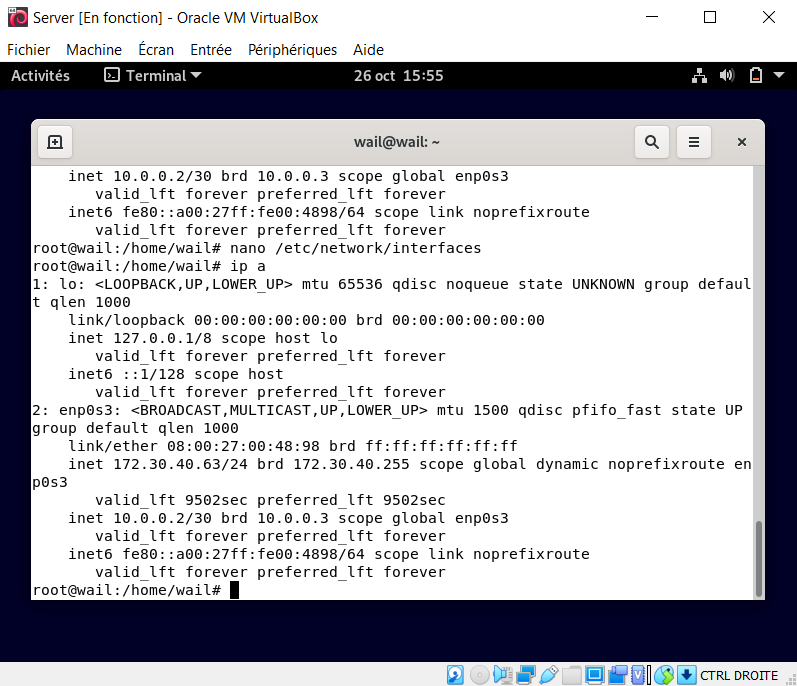
* Avec ceci on peut configurer les interfaces réseaux par la commande : **nano /etc/network/interfaces** (nano c’est pour éditer).



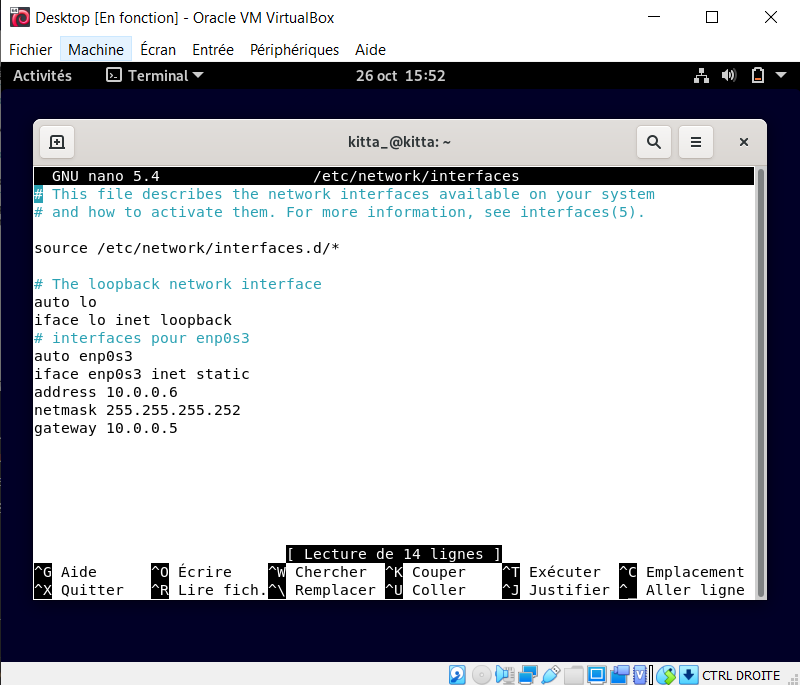
**Serveur ===> Configuration interface**



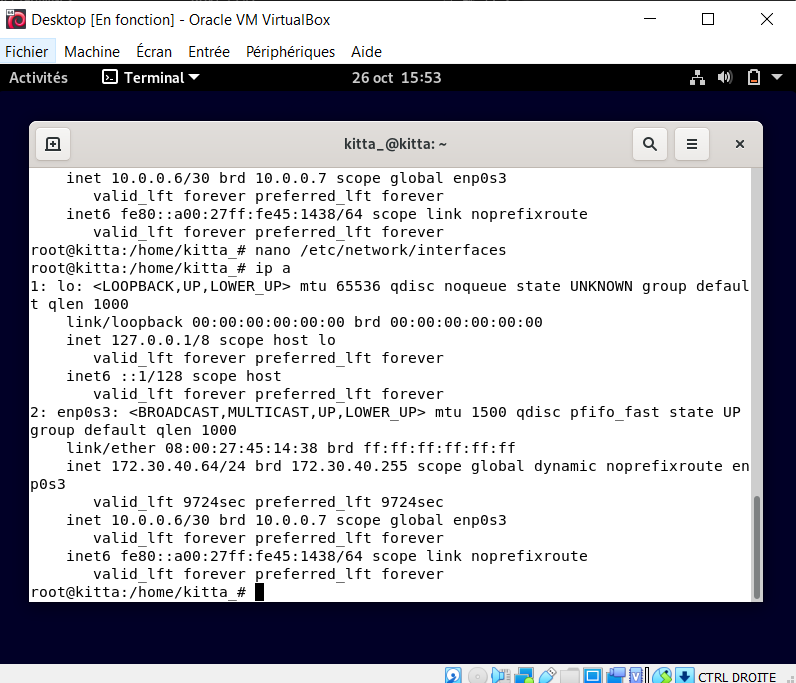
* Puis on utilise la commande : **/etc/init.d/networking restart**  ===> Pour redémarrer le service réseau.
* Puis on vérifie dans la commande **ip a** pour voir si l’adresse a été bien mis



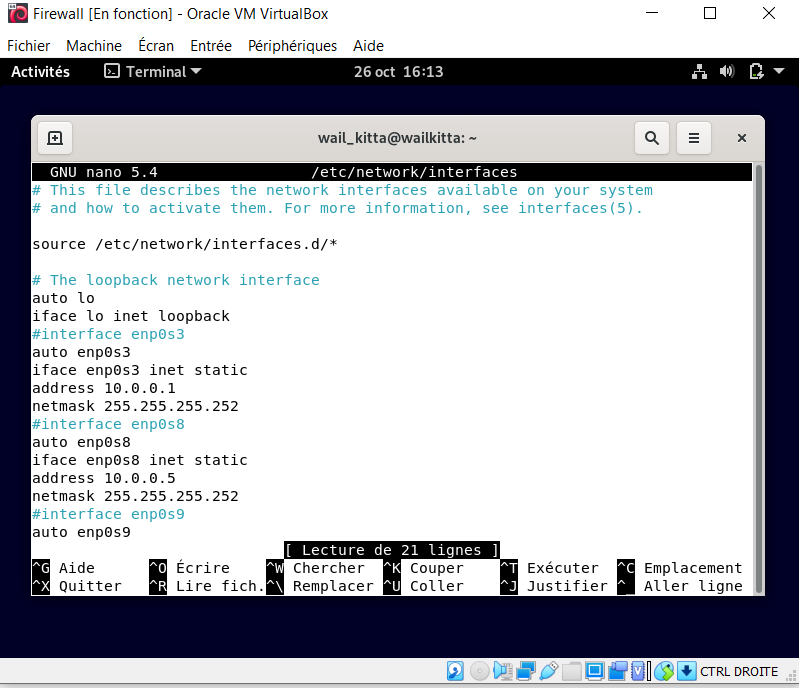
**Desktop ===> Configuration interface**



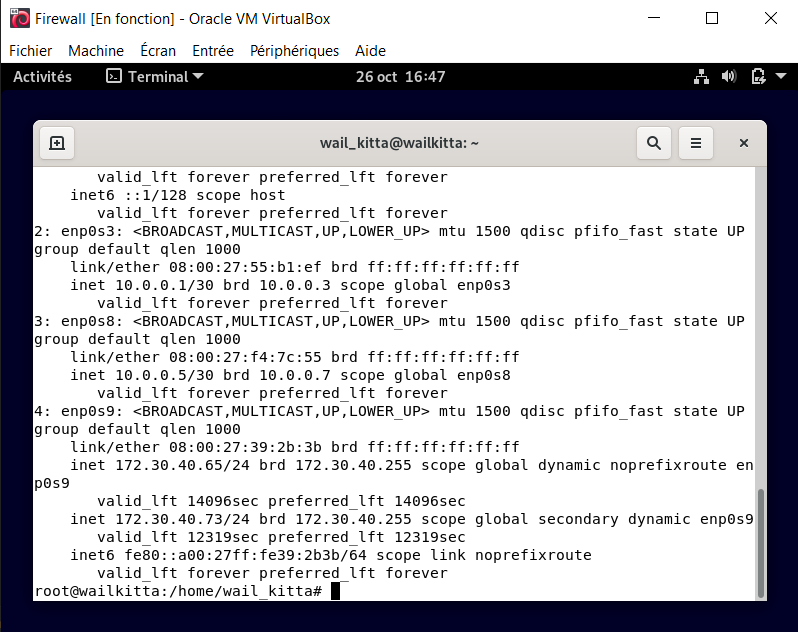
* Puis on utilise la commande : **/etc/init.d/networking restart**  ===> Pour redémarrer le service réseau.
* Puis on vérifie dans la commande **ip a** pour voir si l’adresse a été bien mis



**Firewall ===> Configuration interface**

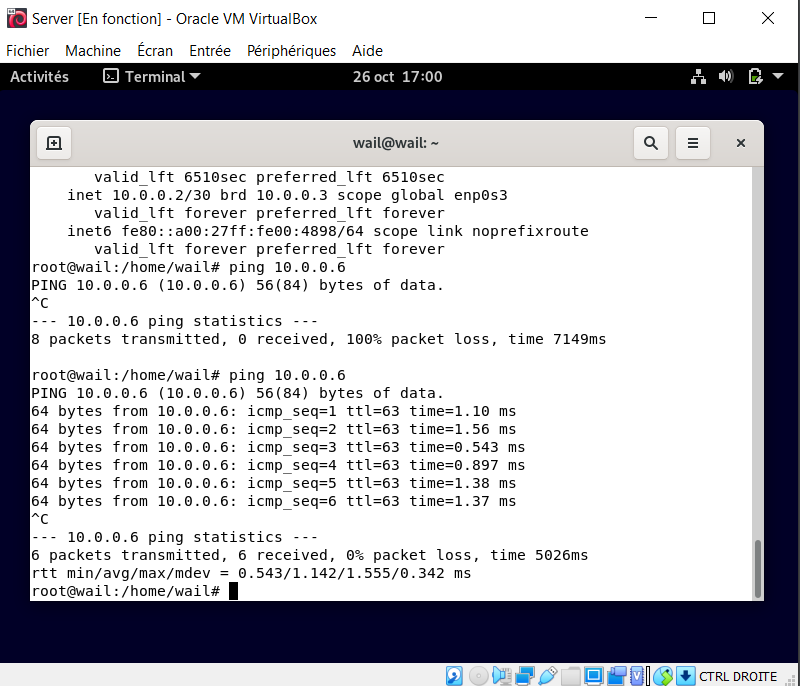


* Puis on utilise la commande : **/etc/init.d/networking restart**  ===> Pour redémarrer le service réseau.
* Puis on vérifie dans la commande **ip a** pour voir si l’adresse a été bien mis

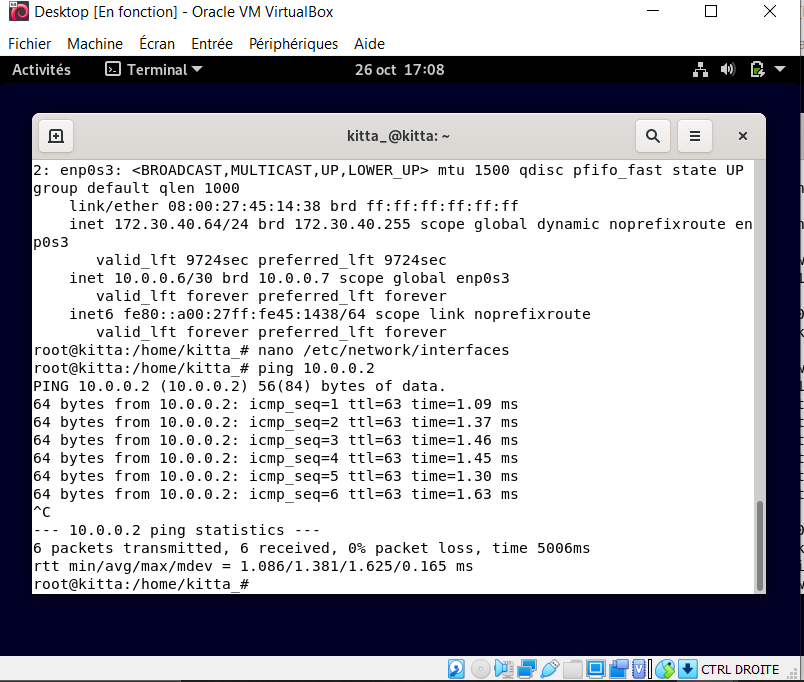


On va activer le routage dans le firewall par la commande : **nano /proc/sys/net/ipv4/ip\_forward** (en le changeant le 0 par 1).

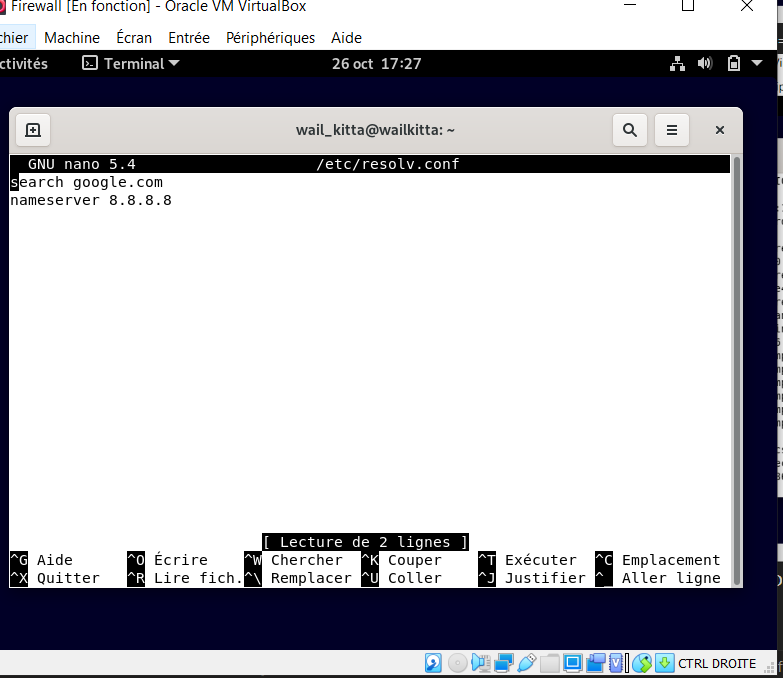
* On va ping serveur==>desktop



* On va ping desktop==>serveur

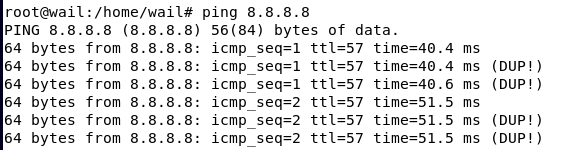


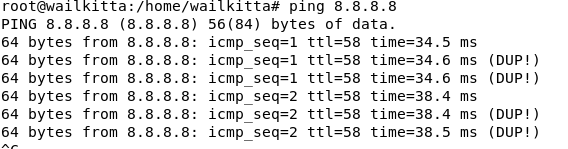
-On va mettre l’adresse IP DNS dans la passerelle (FireWall) grâce à la commande : **nano /etc/resolv.conf**

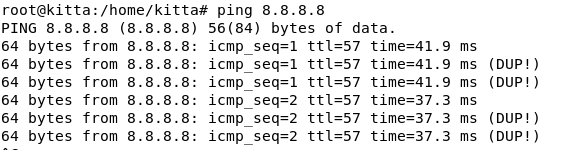


* On va activer le NAT par la commande : **sudo iptables -t nat -A POSTROUTING -o nom\_interface -j MASQUERADE**

On aura avec les 3 machines qui ping google.com





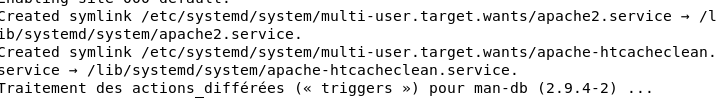


**Fiche de travail 2 : HTTP et SSH**

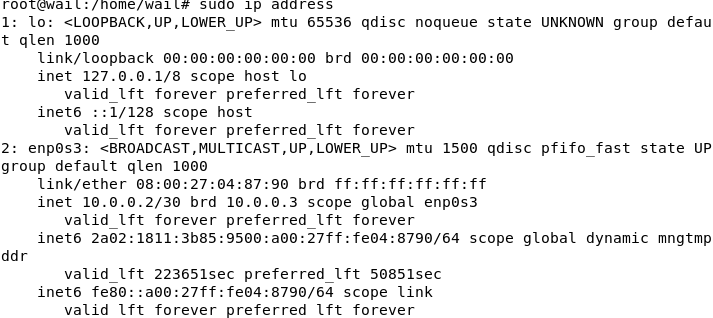
Le service http est l’ensemble de règles régissant le transfert de fichiers ( text,image,son,…) sur le Web.

* Installation du service http (avec apache2) dans le serveur

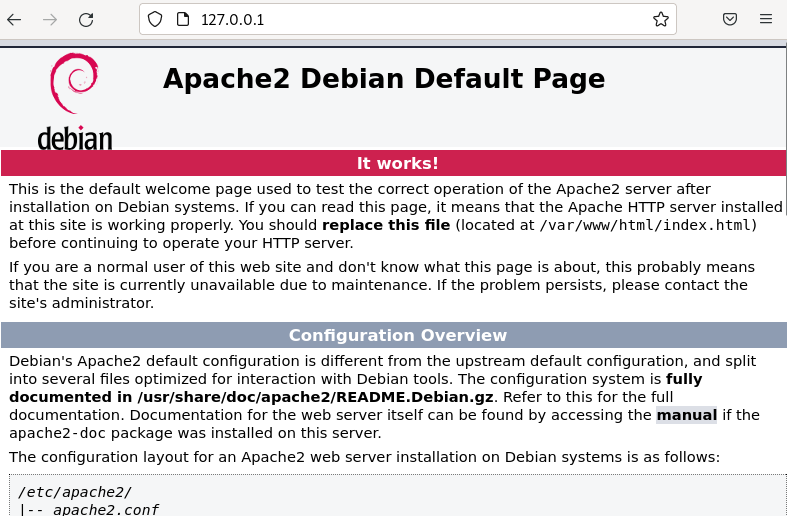
**Sudo apt-get install apache2**



* On fait **sudo ip address** on a l’adresse IP du serveur (127.0.0.1) donc le localhost



* On écrit l’adresse IP du serveur donc le localhost (127.0.0.1) dans la page WEB



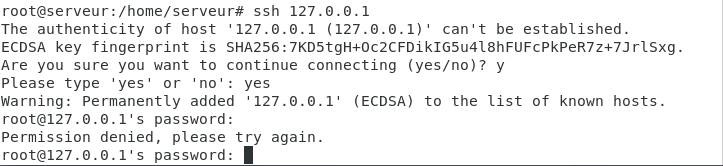
Le SSH (Secure Shell) est un protocole d’administration à distance qui va permettre aux utilisateurs de contrôler et de modifier leurs serveurs distants sur Internet.

* On installe le service ssh

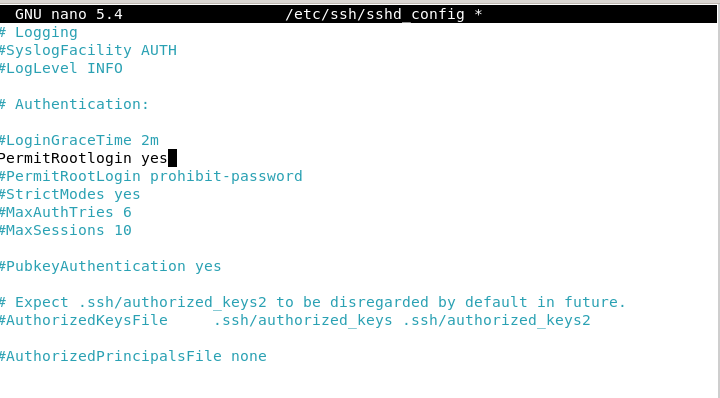




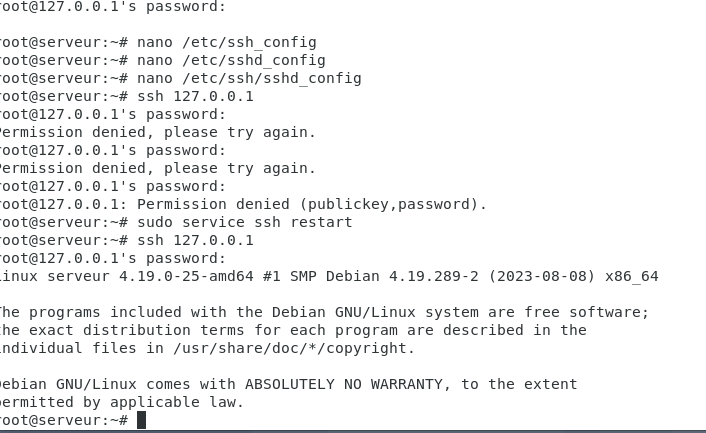
* L’installation a été bien lieu et on va se connecter dans le serveur à distance avec le service ssh (en utilisant le propre serveur pour voir si ça fonctionne bien)



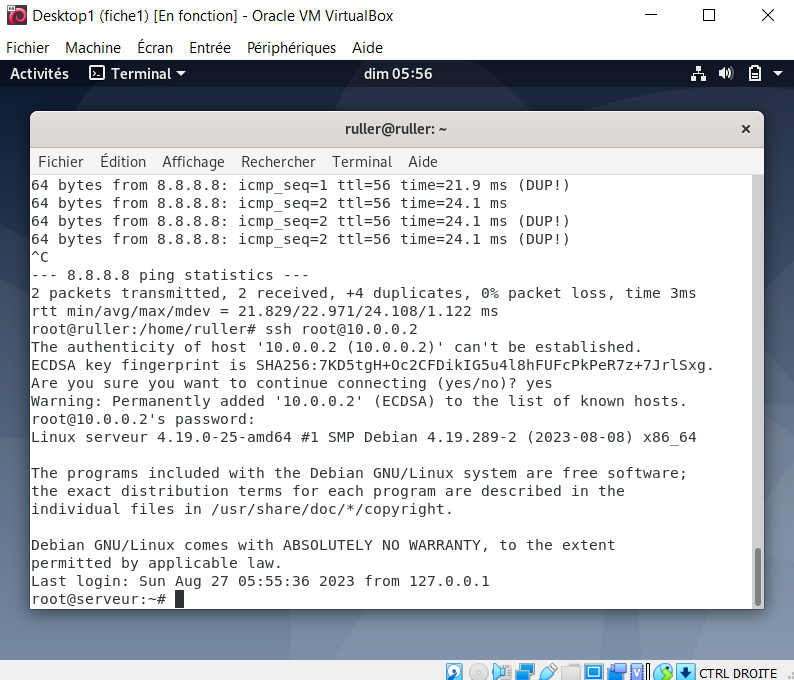
* On n’arrive pas à s’authentifier donc on va aller dans le fichier de configuration **nano /etc/ssh/sshd\_config** et on pourra modifier pour y avoir accès



* On remplace le “#PermitRootLogin prohibit-password" par “PermitRootLogin yes”



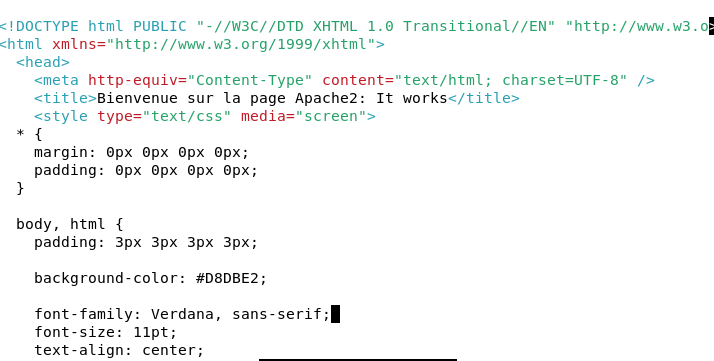
* Donc puis on a fait **sudo service ssh restart** pour faire un restart des services de ssh et on va s’authentifier et ça marche très bien
* Donc là on va essayer de s’aunthentier avec la machine client



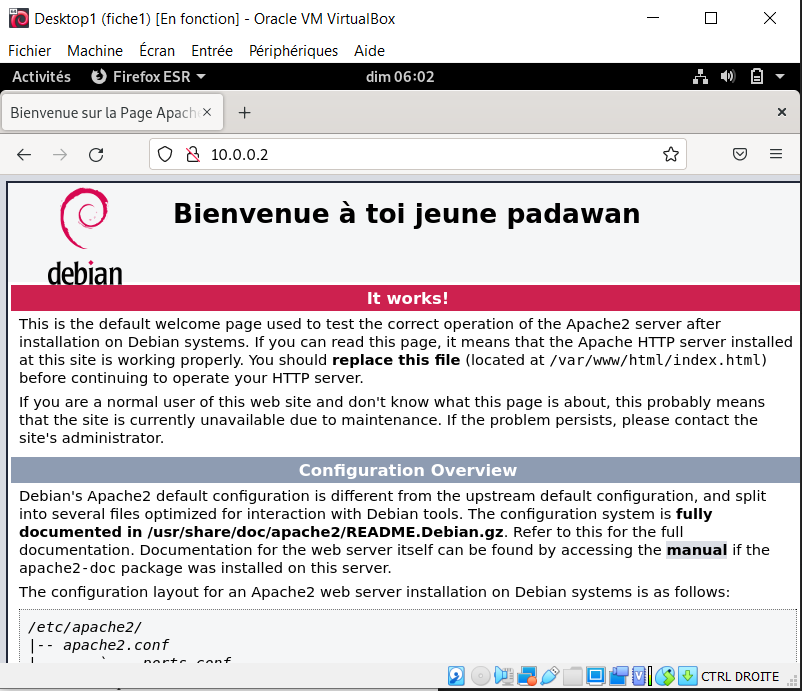
* Donc on est dans la machine client on s’authentifie avec **ssh** [**root@10.0.0.2**](mailto:root@10.0.0.2)(10.0.0.2 adresse IP de la machine serveur)
* On se connecte dans la machine serveur comme prévu
* On va personnalisez la page Web avec cette commande ci-dessous:

Nano /var/www/html/index.html

* Donc on aura ceci



* On va modifier le titre



* Comme prévu ça a très bien marché.

**Fiche de travail 3 : DNS**

Un service DNS permet de traduire un nom de domaine en une adresse IP.

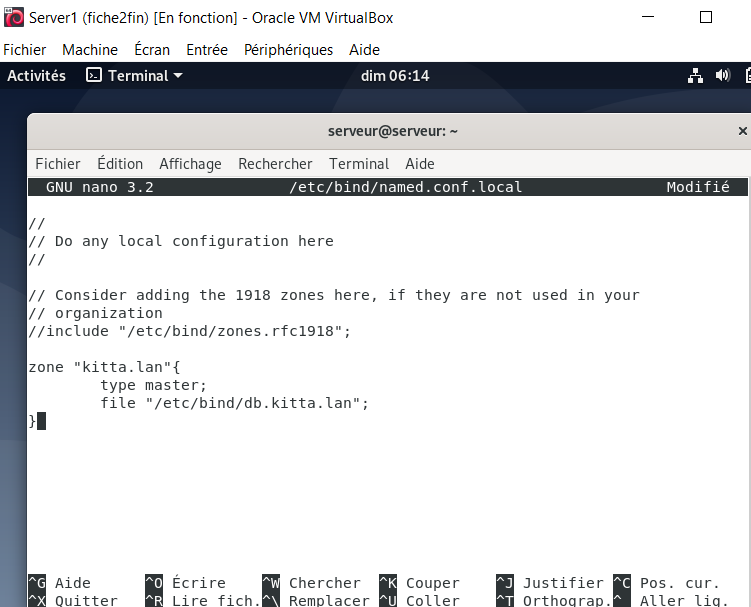
* On installe le service DNS (avec bind9):

Sudo apt-get install bind9

* Ici c’est bien active le service DNS



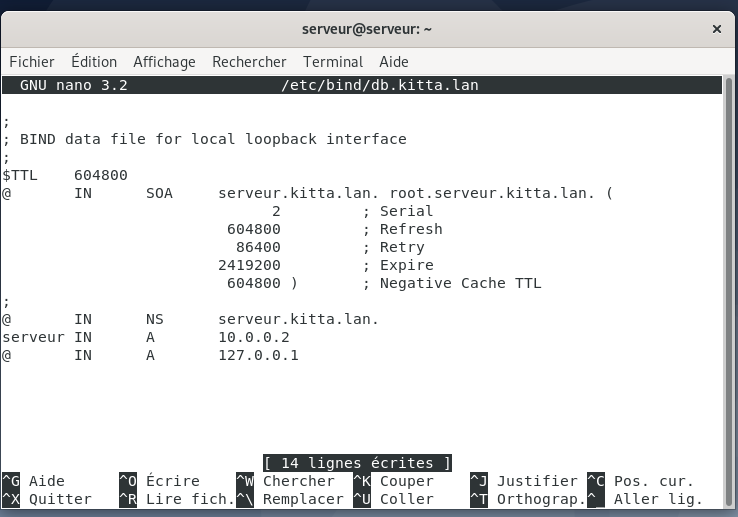
* On va configurer la zone directe dans la commande **nano /etc/bind/named.conf.local**



* On va créer un fichier de config pour chaque zone crée, donc ici on a qu’une. Donc on va copier le fichier “db.local” qui se trouve dans le dossier “/etc/bind/” et on va le renommer “db.kitta.lan” donc on va réaliser par la commande ci-dessous.



* Puis on va rentrer dans le fichier “**nano /etc/bind/db.kitta.lan”**



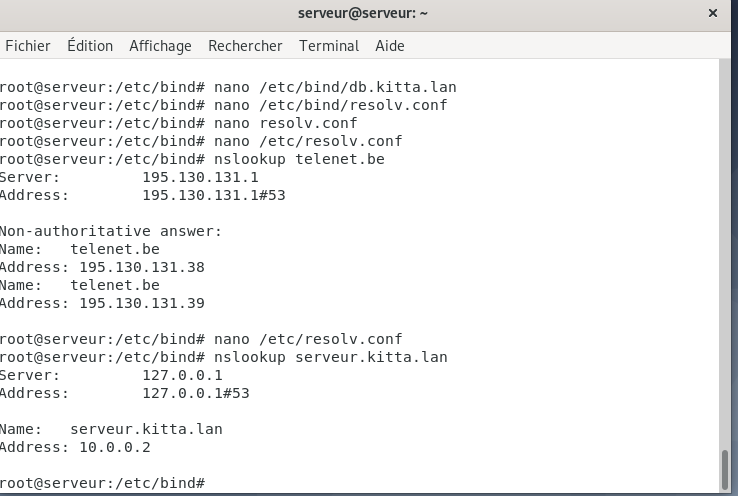
* Mais d’abord on verra le nom du serveur de cette machine par cette commande “**nano /etc/hostname” qui sera serveur**
* Donc on va fixer une adresse IP DNS dans ce serveur avec cette commande-ci “**nano /etc/resolv.conf”**



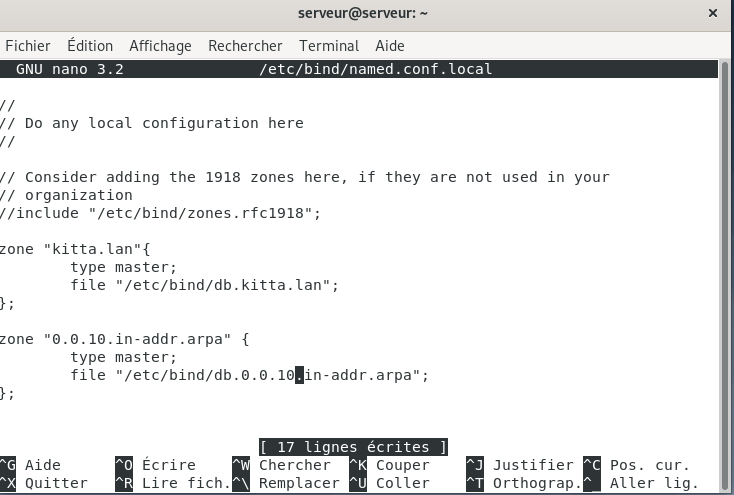
* Pour que les modifications soient prises on met cette commande ci-dessous:

Sudo service bind9 restart

* On va faire un test avec nslookup “**sudo nslookup serveur.kitta.lan”**



* Ca marche très bien
* Là on va essayer de faire la zone inverse



* On rajoute ici la zone inverse
* Donc la on va faire le même protocole que la zone directe avec la zone inverse





* La on va faire le nslookup



Voilà

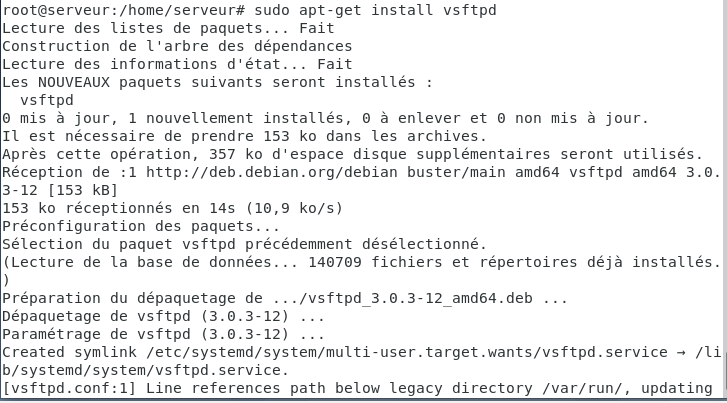
**Fiche de travail 4 : FTP**

Le serveur FTP (Protocole de transfert de fichier) est un type de serveur qui va permettre le transfert de fichiers entre les ordinateurs connectés à un réseau.

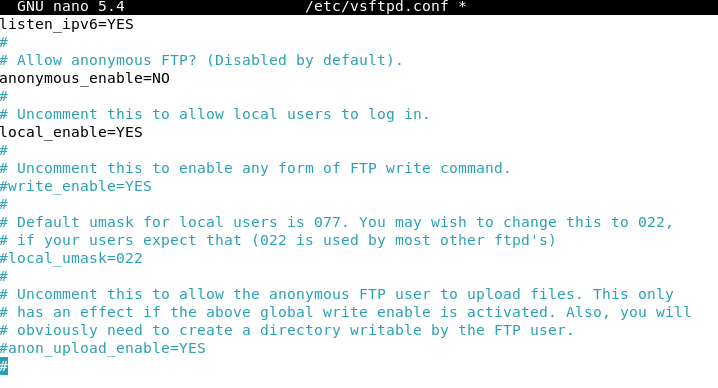
* On installe le serveur FTP par cette commande ci-dessous

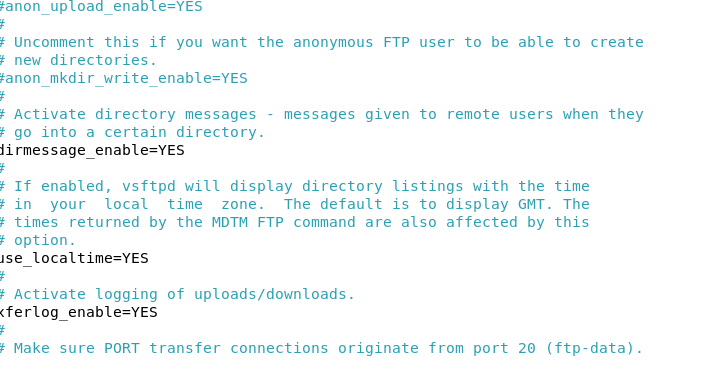
Sudo apt-get install vsftpd

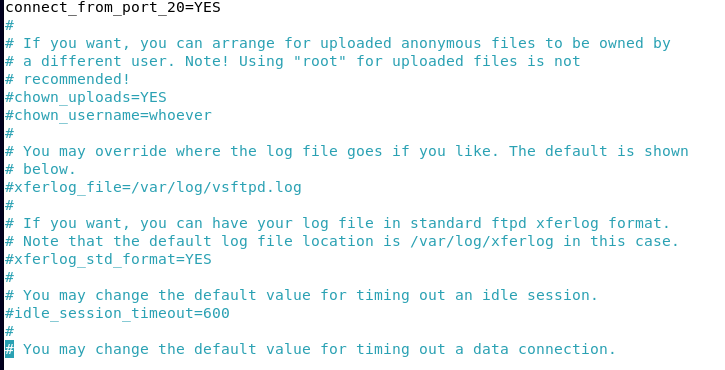
* L’installation à au bien lieu

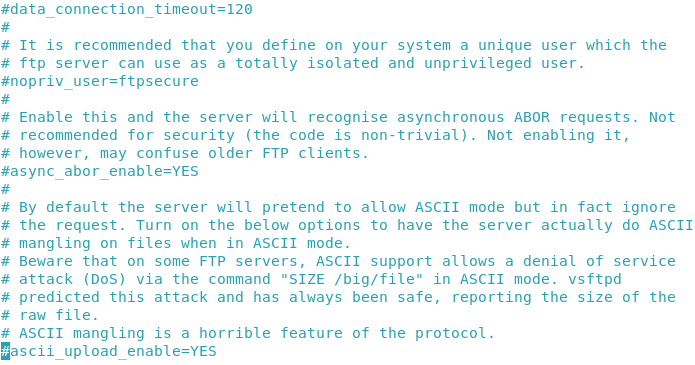


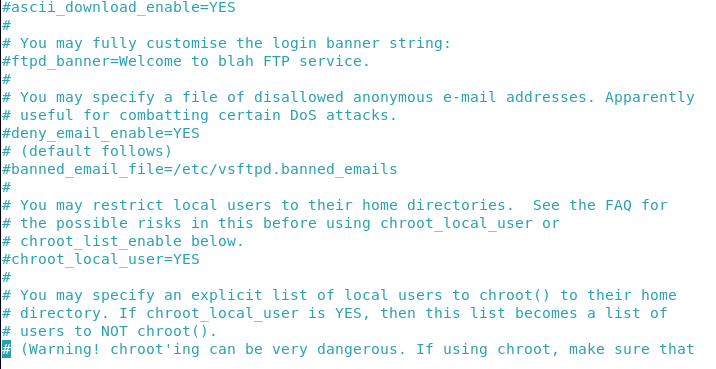
* On va faire des configurations sur le serveur ftp donc on va configurer le fichier “vsftpd.conf” se trouvant dans le dossier “/etc/” donc :
* On aura ceci :





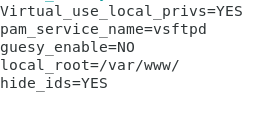








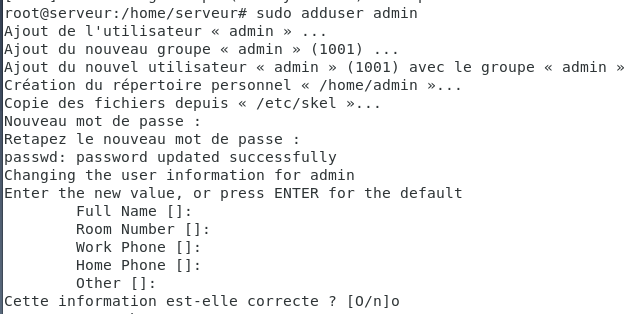
* On va modifier et on aura cette liste:
* anonymous\_enable=NO
* local\_enable=YES
* write\_enable=YES
* local\_umask=022
* anon\_upload\_enable=NO
* anon\_mkdir\_write\_enable=NO
* xferlog\_file=/var/log/vsftpd.log
* Idle\_session\_timeout=300
* data\_connection\_timeout=120
* ascii\_upload\_enable=NO
* ascii\_download\_enable=NO
* ftpd\_banner=Salut
* chroot\_local\_user=YES
* chroot\_list\_enable=NO
* Puis on rajoute ça :



* On redémarre vsftpd
* sudo /etc/init.d/vsftpd restart



* On rajoute un utilisateur : admin



* La on a deux utilisateur dans le pc serveur



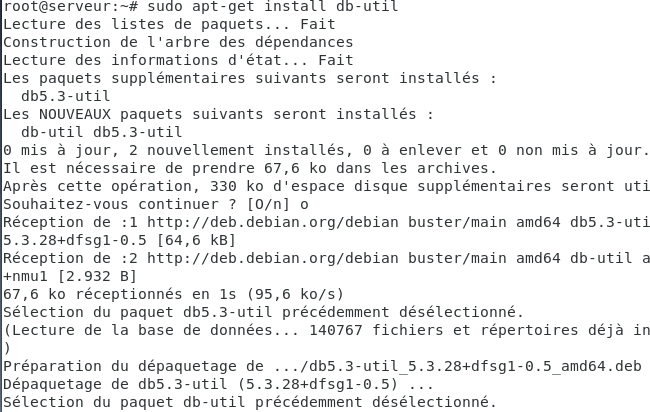
* On va créer un fichier pour la BD de vsftpd :





* On va installer db-util pour la base de donnée

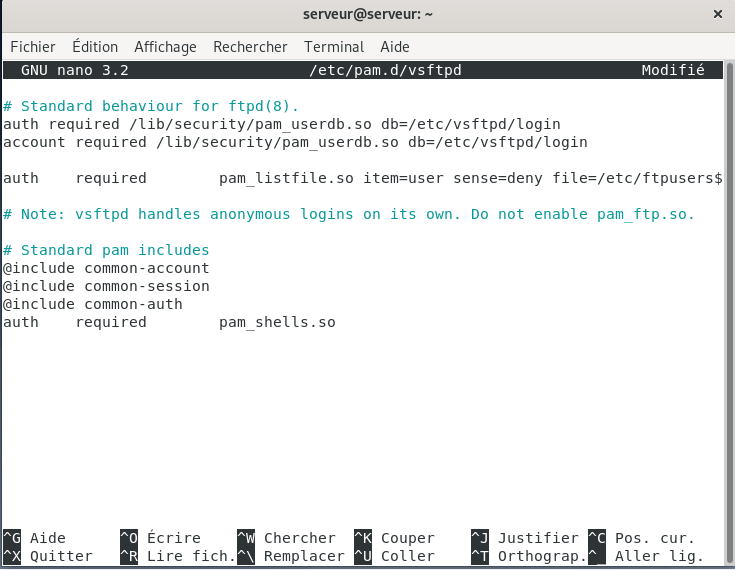




* Cette commande ci-dessous c’est pour créer la base de donnée

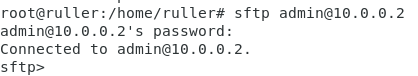


* On va ajouter des lignes de commandes dans ce dossier :

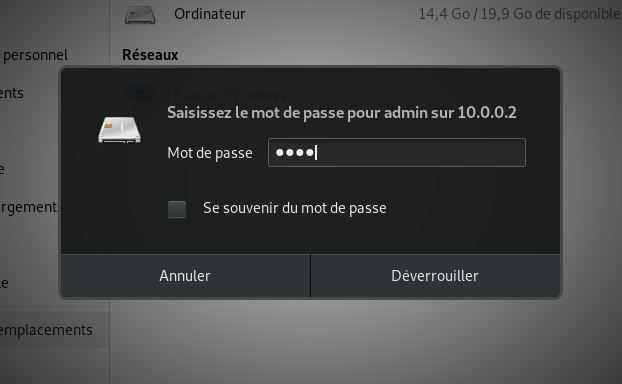


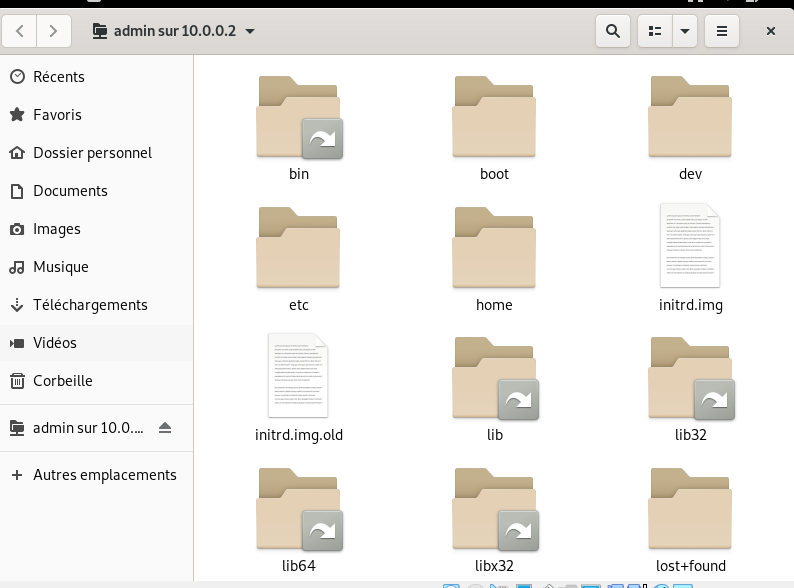


* Le service a été bien redémarré
* La connexion avec le ftp classique n’est plus possible dans le navigateur mozilla du coup on utilise sftp qui est utilisé dans un environnement du bureau gnome

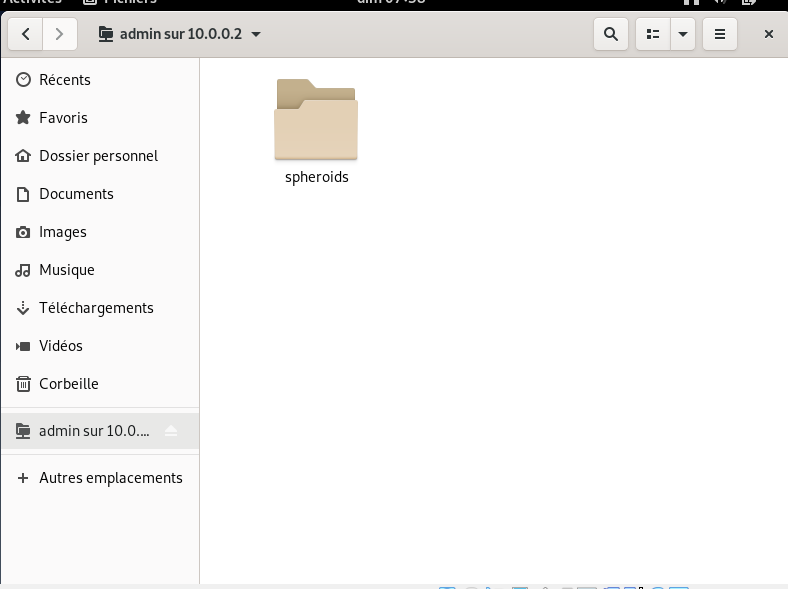








* On installe un dossier spheroids pour modifier la page web

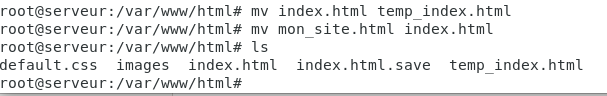


* Puis on va copier ce qu’il y a dans le dossier et mettre tout ça dans le dossier de la page web

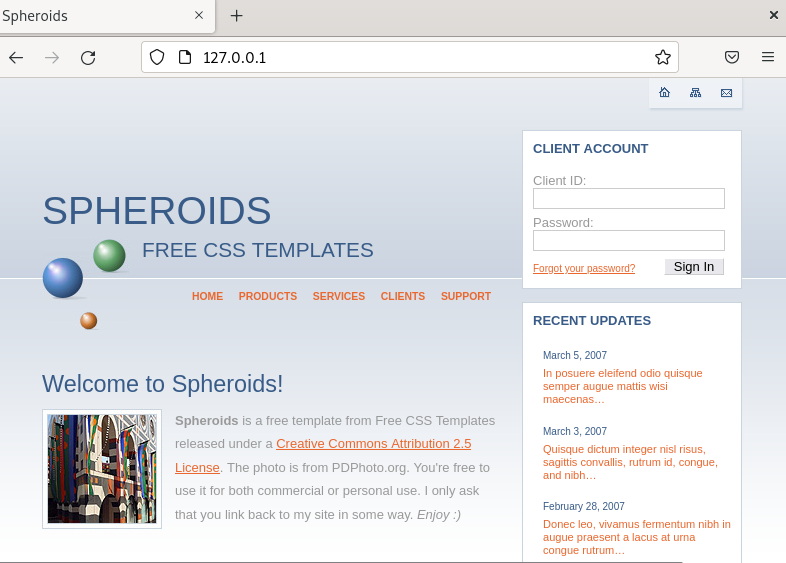




* On va faire autoriser ça



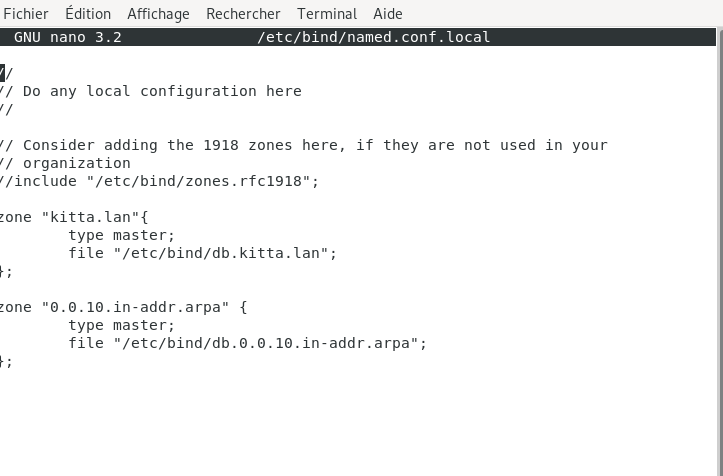
* Les fichiers sont bien conformes pour être appliqué



**Fiche de travail 5 : SMTP**

Le SMTP est un protocole de communication standard utilisé pour l'envoi de courriers électroniques (e-mails) à travers les réseaux informatiques.

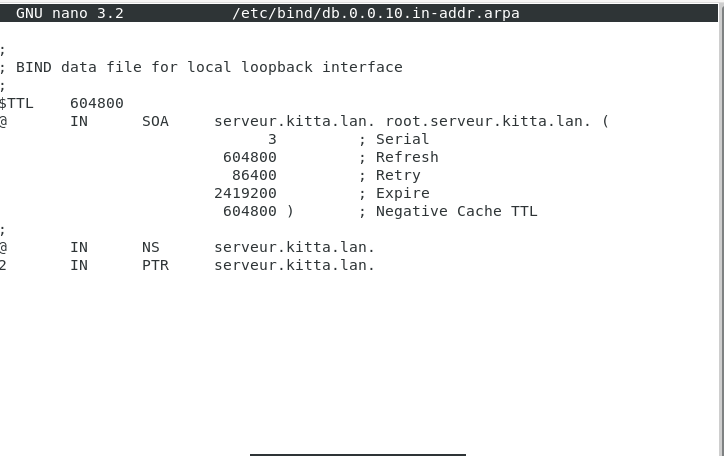
D’abord on va voir si le DNS est bon.





Ici on a rajouté une commande MX

Donc le MX est un type d’enregistrement DNS utilisé pour spécifier les serveurs de messagerie qui sont responsables de la réception des courriers électroniques pour un domaine spécifique.





L’entrée marche bien ici parfait.

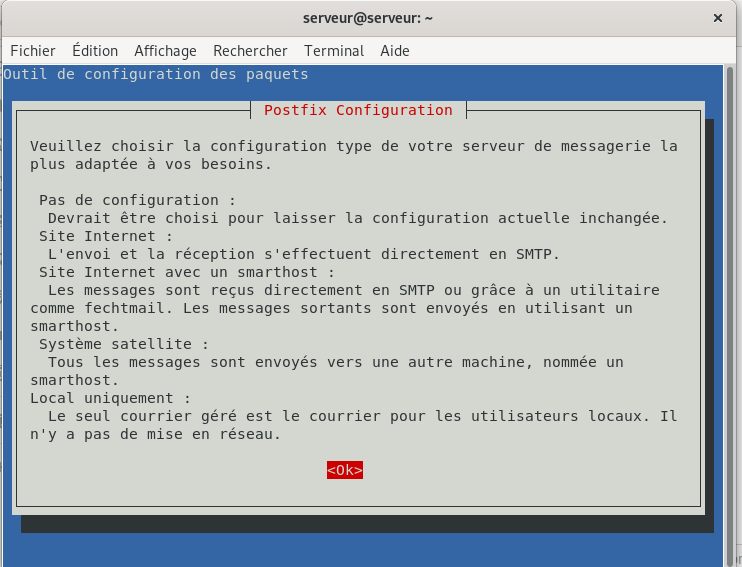
On va installe le postfix :

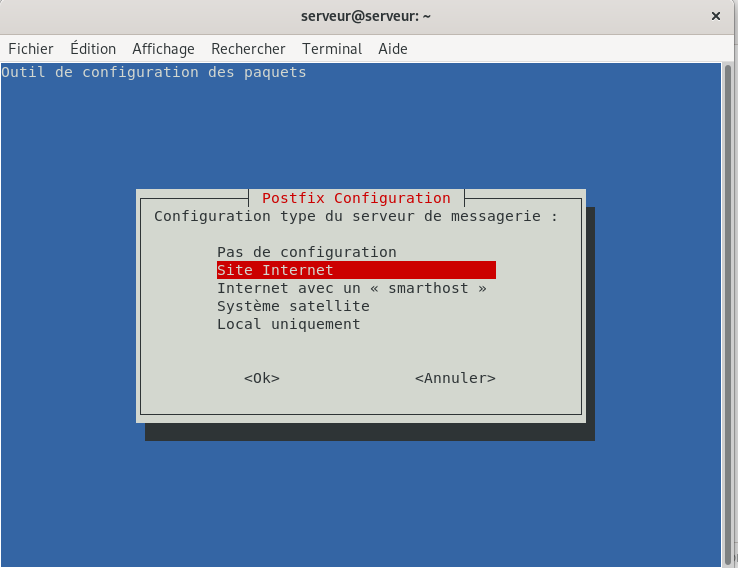


Postfix est un logiciel de serveur de messagerie électronique open-source et largement utilisé.

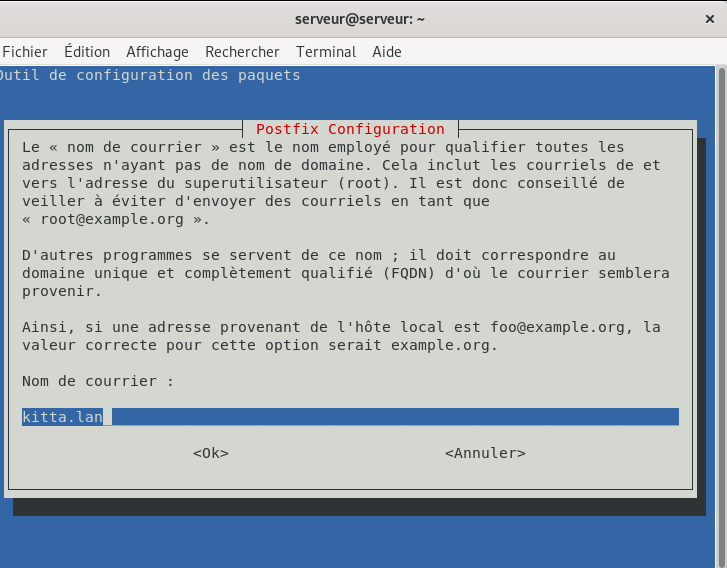
Postfix est principalement utilisé comme serveur de transfert d'e-mails (MTA - Mail Transfer Agent).

MTA signifie "Mail Transfer Agent" (Agent de transfert de courrier en français). C'est un logiciel ou un composant logiciel responsable de l'acheminement des courriers électroniques (e-mails) d'un serveur à un autre dans le cadre du processus de livraison des e-mails.

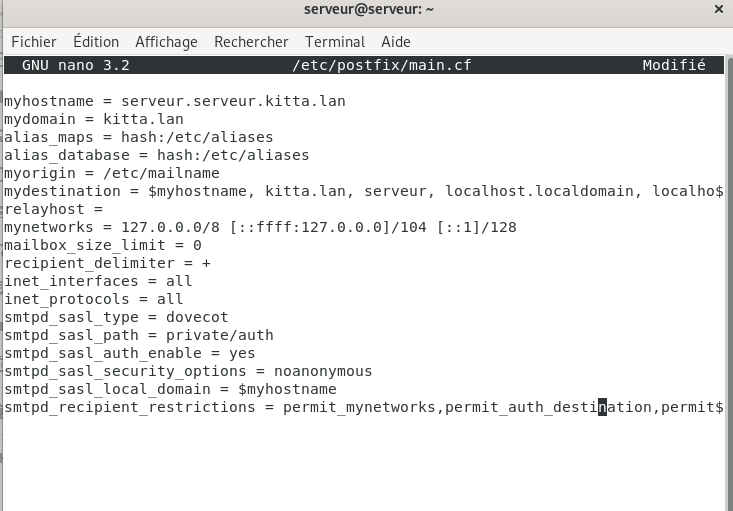




On choisi “Site Internet”



Ici on met le nom de notre domaine



On configure le main.cf



On redémarre le service

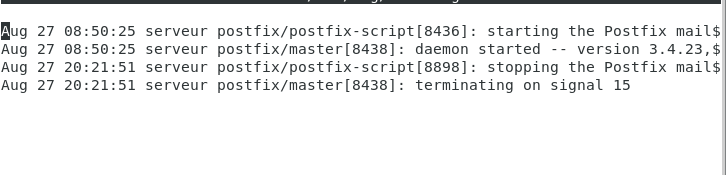
Puis on installe mailutils:



Puis on essaye de tester pour un envoie d’un mail



Vu que je n’ai pas l’accès à [eln@irisib.lan](mailto:eln@irisib.lan) alors j’ai testé avec mon mail en essayant le test j’ai vu que ça marchait pas car:



Donc ça vient de main.cf le problème et j’ai remarqué que je devais rajouter deux commandes et enlevez des commandes pas nécessaire.

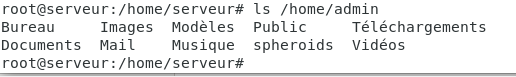
Mais même avec ça mon mail personelle est protégé par google c’est pour ça je reçois pas.



Ici j’ai envoyé un mail à admin et il recevait bien un courrier où je l’ai ajouté dans le main.cf



On va voir ce courrier dans la session de l’utilisateur:



Puis on va installer dovecot.

Donc devecot est un serveur IMAP et POP3 sécurisé. Le IMAP (Internet Message) permet de sauvegarder les messages et pièces jointes dans le serveur de messagerie, ce qui nous permet de les accéder peu Importe les appareils (smartphone, ordinateur, tablette). POP3 (Post Office Protocol) supprime les messages et pièces jointes du serveur une fois que l’appareil de destination les a obtenues.

Du coup on va l’installer

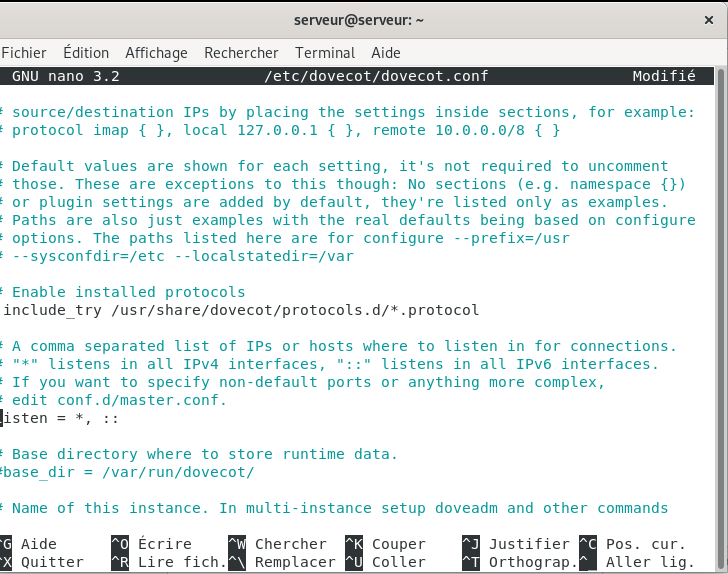


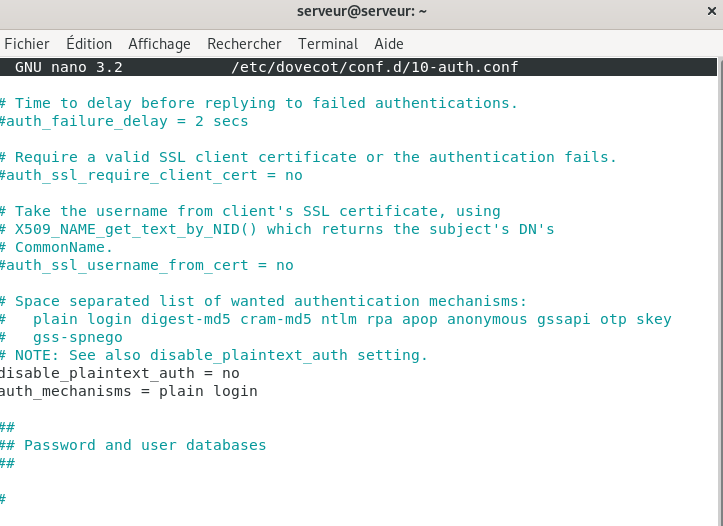
Ca installe dovecot-pop3



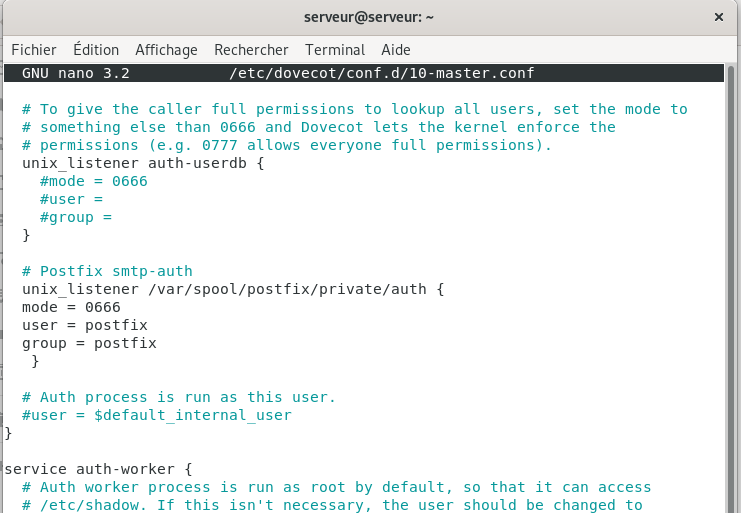
Ici ca installe dovecot-imap

Puis on configure

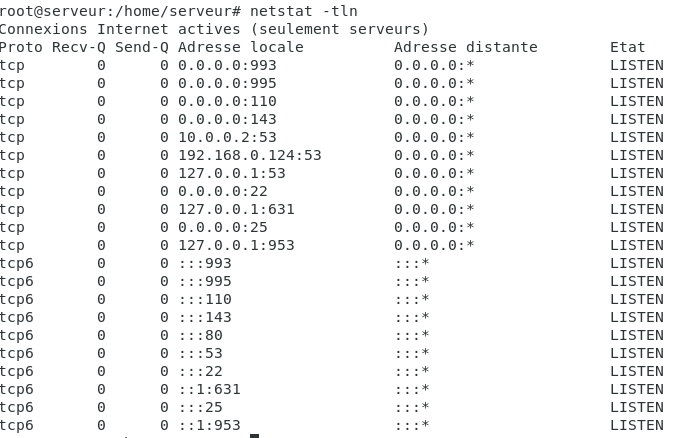








Puis là je vérifie si j’ai bien des connexions sur les ports 143 (POP3), 993 (IMAP) et SMTP (25) :

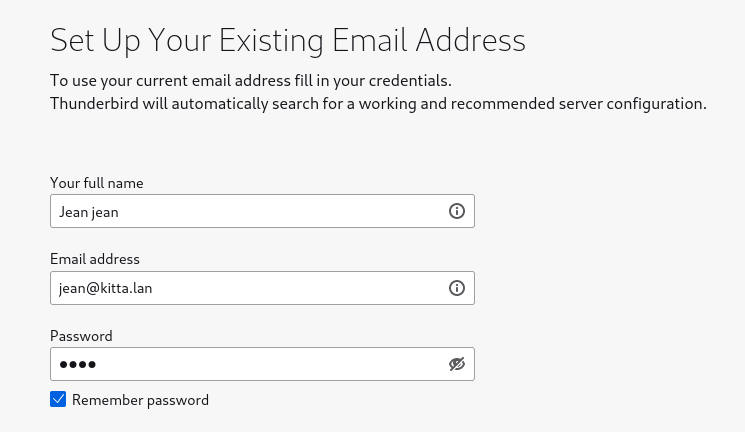


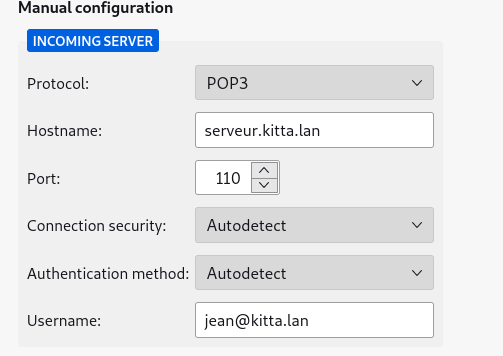
Tout est en écoute parfait.

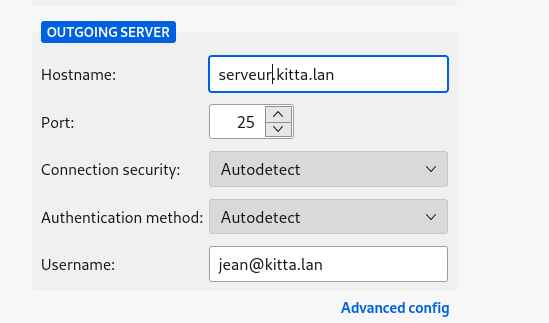
Avant d’envoyer un mail , on va installer thunderbird.

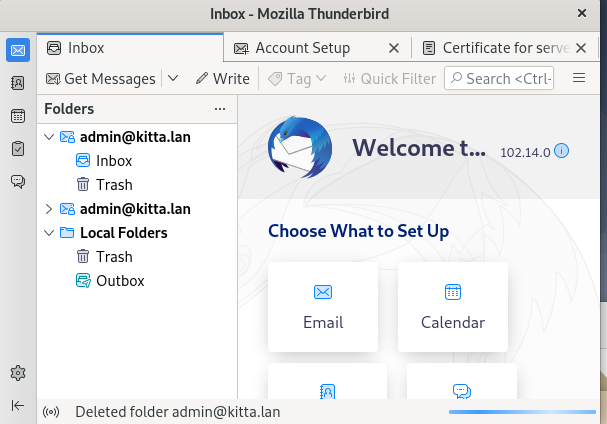


Thunderbird est un client de messagerie et me permettra de vérifier que le mail à bien été reçu.







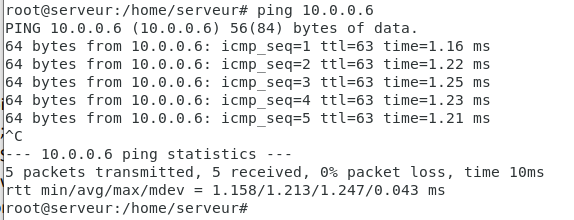


Ici je me suis connecté avec admin donc voila le mail de admin.

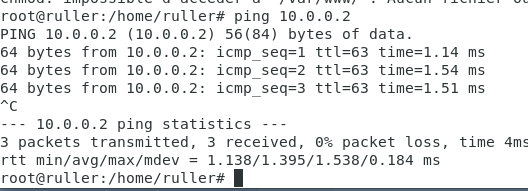
**Fiche de travail 6 : Netfilter**

La configuration des machines est déjà faite depuis la fiche 1 aussi l’activation de routage et de la mise en place de NAT.

On va déjà verifiér si le pc client ping avec le serveur.

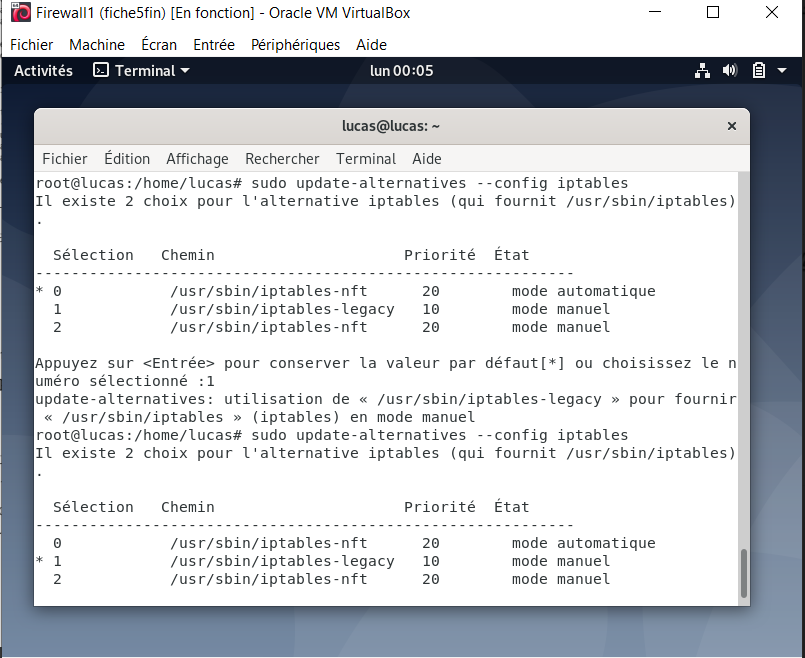


Serveur ok



Client ok

Du coup on va changer le mode de nftables à legacy.



Ici c’est le Firewall(lucas)

Les commandes iptables :

Ils permettent de créer des règles au sein de réseau donc ils sont utilisé pour configurer le pare-feu dans le système. Donc permettent de gérer des règles de filtrages de paquets, de translation d'adresses réseau (NAT) et d'autres aspects de la sécurité réseau.

Les premiers paramètres sont :

* A : ajouter une nouvelle CHAIN rule
* D : supprimer une chaine rule
* L : liste tout les règles

Nous allons nous intéressez au "CHAIN" de 2 type de tables :

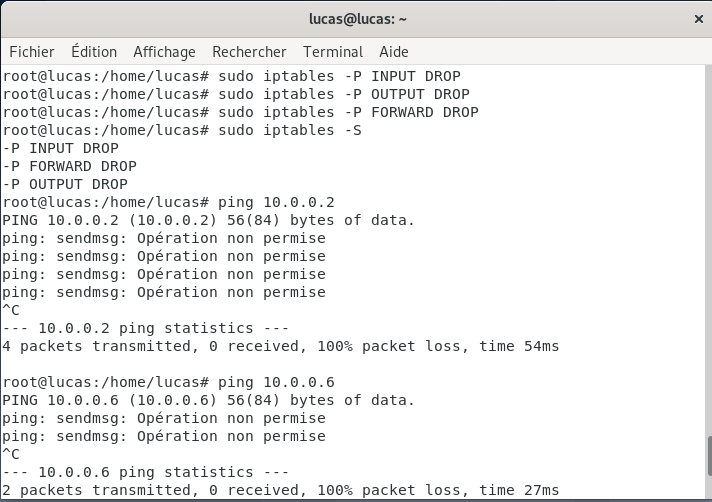
1. CHAIN de la table FILTER

* FORWARD : Filtre les paquets qui transitent par le pare feu
* INPUT : Filtre les paquets entrants qui sont destinés au pare-feu
* OUTPUT : Filtre les paquets sortants qui sont émis par le pare-feu

1. CHAIN de la table NAT

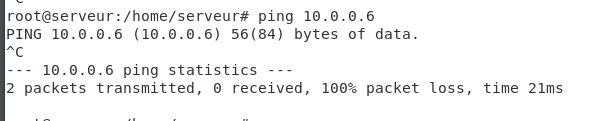
* PREROUTING : changement d’IP destination
* POSTROUTING : changement d’IP source
* OUTPUT : changement d’IP des paquets du pare-feu
* Blocage des trafics avec la police par défaut

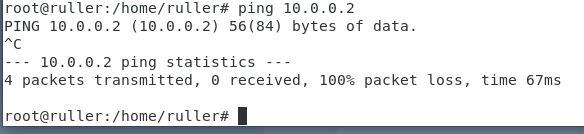
Avant de commencer avec ses règles on va bloquer tout le réseau on va utiliser l’état “DROP”



Donc ici on a utilisé ces commandes dans le firewall

On va voir s’ils ping les deux autres machines :





Ca ping pas donc la règle a été bien appliqué.

* La on va mettre en place les règles
  + Autorisation ping

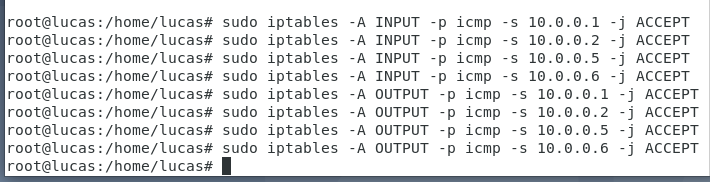
Pour autoriser par communication ping faut ajouter 2 règles pour chaque adresse:

1. sudo iptables -A INPUT -p icmp -s 10.0.0.2 -j ACCEPT

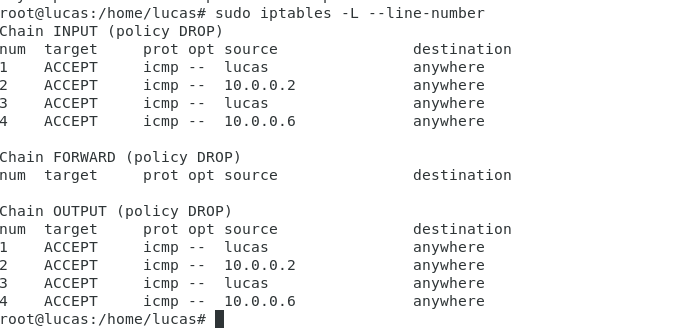
* p : protocole pour laquelle cette règle s’applique, ici c’est icmp
* s : adresse ip source
* j : permet de définir l’action à faire si le traffic coresspondant à la règle, ici c’est ACCEPT,

1. sudo iptables -A OUTPUT -p icmp -d 10.0.0.2 -j ACCEPT

* d : adresse destination



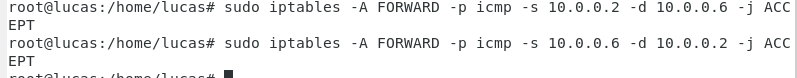
Ajout des règles



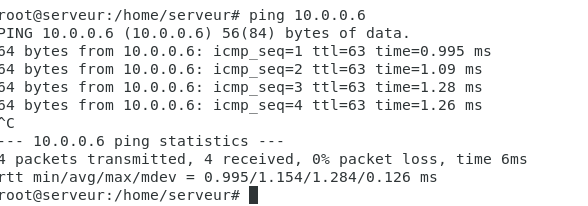
Ici les règles ont été bien appliqué mais il manque le CHAIN FORWARD qui va permettre au pare-feu de transiter les paquets :

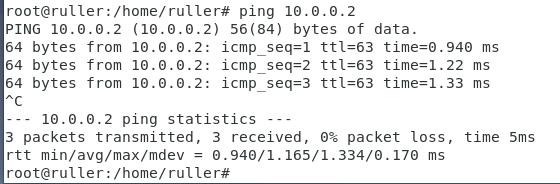
1. sudo iptables -A FORWARD -p icmp -s 10.0.0.2 -d 10.0.0.6 -j ACCEPT

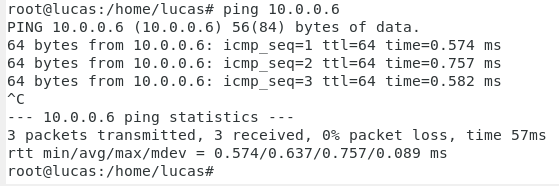
* s : adresse source
* d : adresse destination

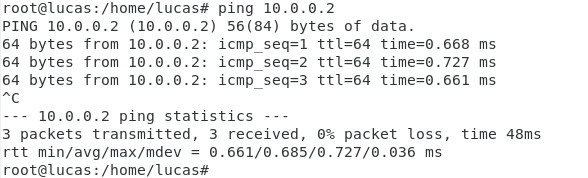


Puis on va voir si ca ping entre les machines :

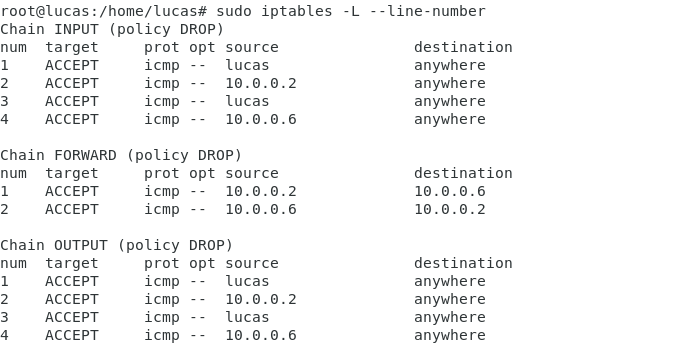




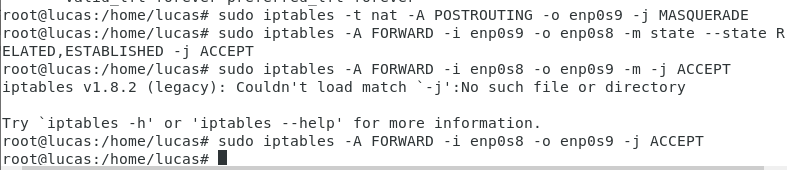




Donc on peut voir que ca ping et que la règle a été bien appliqué.



* + **Permettre uniquement à la machine Desktop la navigation web**



J’ai appliqué 3 commandes ici :

1. sudo iptables -A POSTROUTING -o enp0s9 -j MASQUERADE

* POSTROUTING : CHAIN sortante du réseau
* MASQUERADE : application du NAT Dynamique

1. sudo iptables -A FORWARD -i enp0s9 -o enp0s8 -m state –state RELATED,ESTABLISHED -j ACCEPT

* i : interface d’entrée
* o : interface de sortie
* m state : module state
* state RELATED,ESTABLISHED : indique que la règle s’appliquera au trafic qui est lié à une connexion établie ou déjà établie.

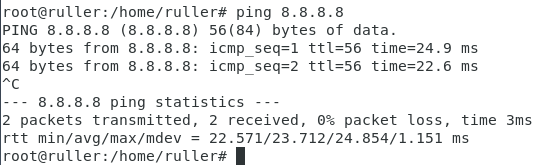
1. sudo iptables -A FORWARD -i enp0s8 -o enp0s9 -m -j ACCEPT



Ici c’est la machine firewall donc y a pas de ping avec google

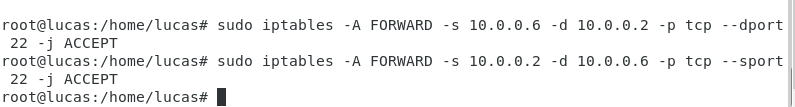


Même chose pour le serveur



Et là la machine client ping sur google donc la règle a été bien appliqué.

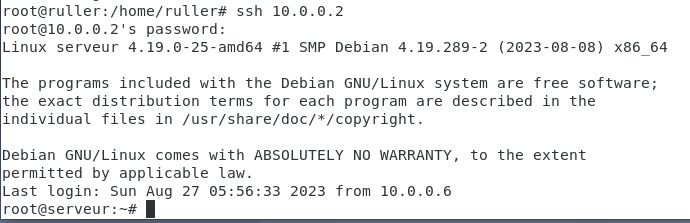
* + **Permettre uniquement à la machine Desktop de se connecter à Server via SSH**



- dport : le port de destination

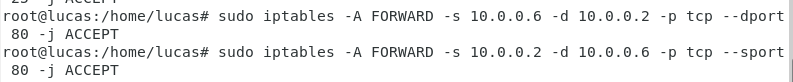
- sport : le port source

Donc ici on a utilisé 2 règles où le port de ssh c’est 22

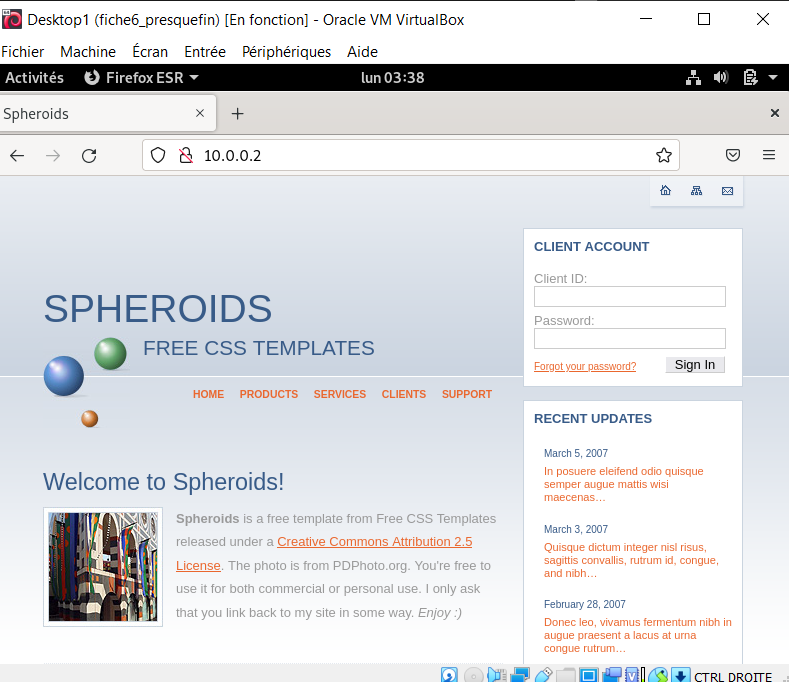


Les règles ont été bien appliqué

* + **Permettre uniquement à la machine Desktop d’ouvrir une page web sur Server**

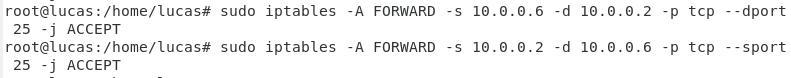


Même style que le ssh juste on a changé le port ssh par le port http qui est 80

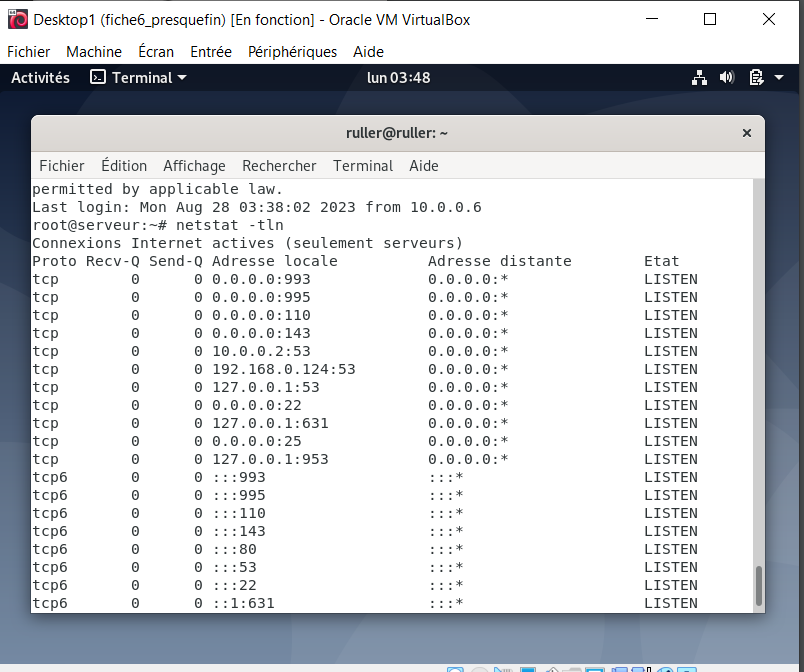


Les règles ont été bien appliqué

* + **Permettre uniquement à la machine Desktop d’effectuer une connexion smtp sur Server**



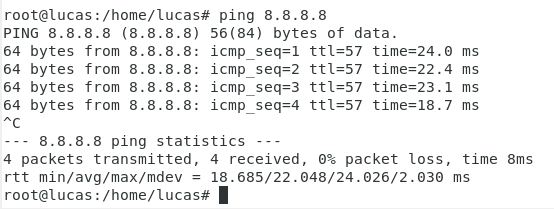
Ici même chose que les précédentes on change que le port par le port smtp qui est 25



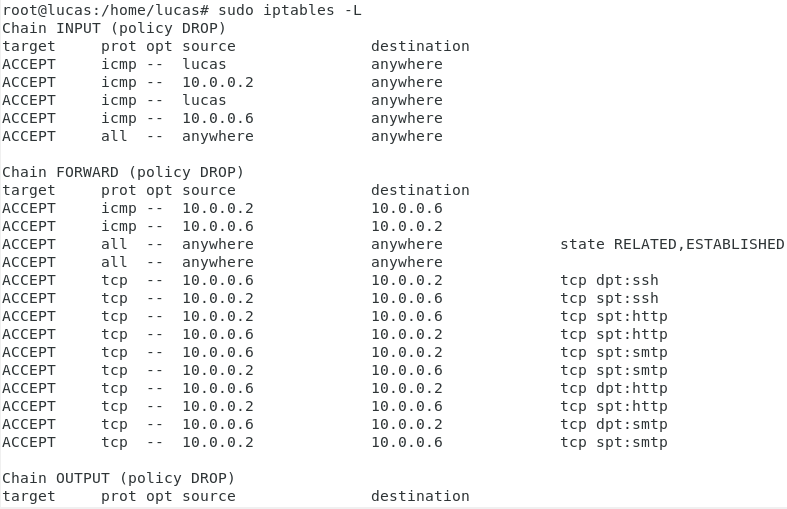
Ici les règles ont bien été appliqués

* + **Permettre au pare-feu l’accès à internet**





Ici aussi pour firewall il a une connexion internet donc les règles ont été bien appliqué



Voici toutes les règles qu’on a fait.

* Shorewall

Shorewall est un outil de configuration qui permet de réaliser la configuration des règles iptables plus simple.



Donc on va installer shorewall



Ici on va prendre un exemple pour le mettre en pratique.

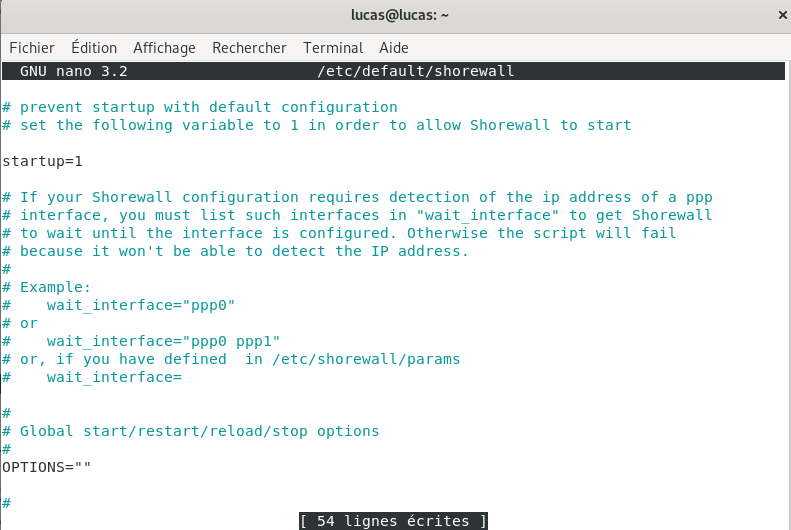
Donc on va le copier dans un dossier /etc /shorewall



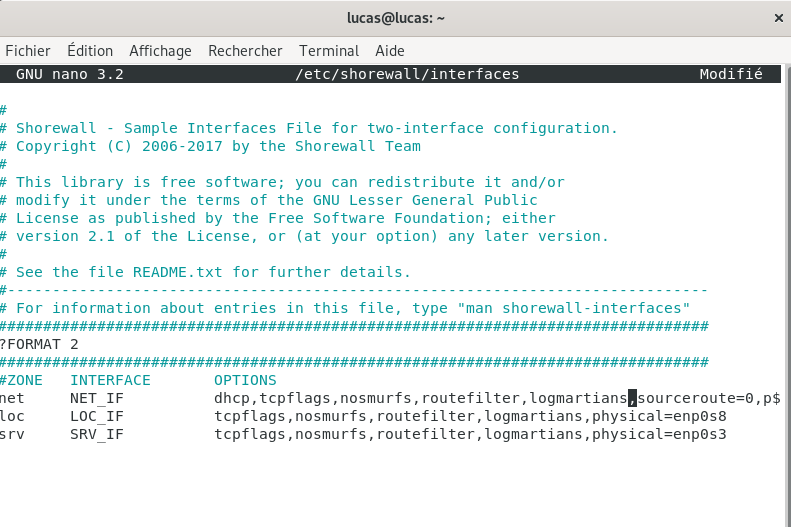
Ce qu’il y aura à l’intérieur

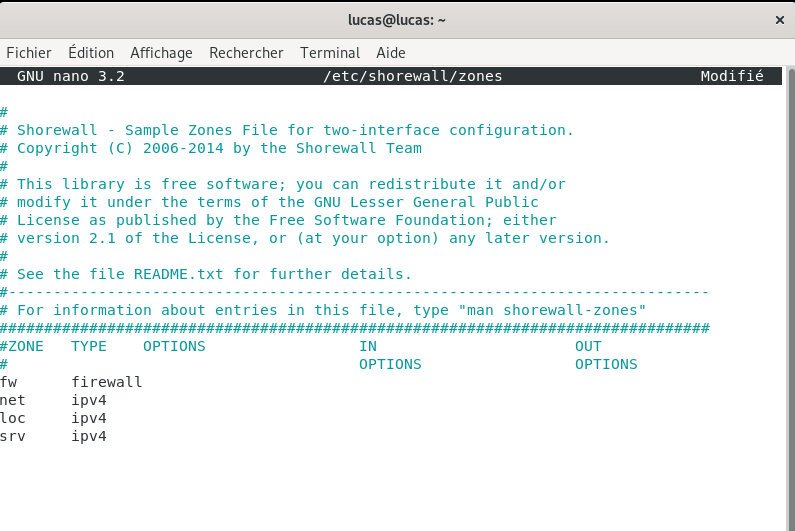


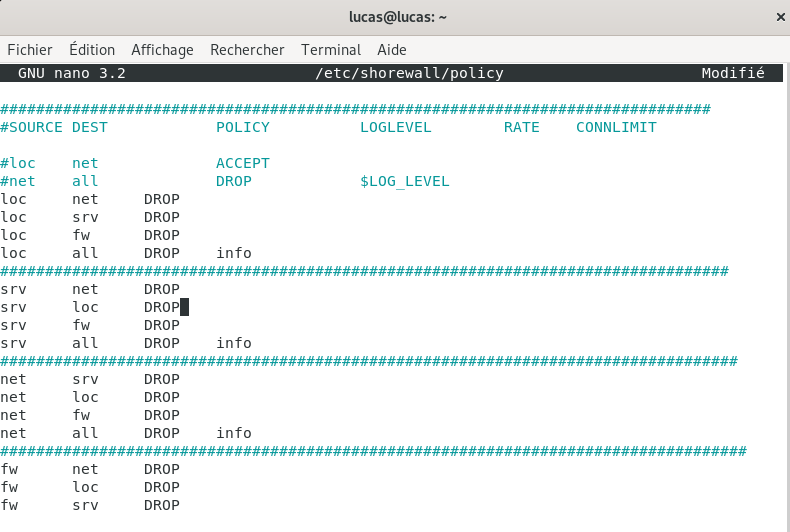
Puis j’autorise le lancement automatique de shorewall au démarrage de la machine en configurant ci-dessous:

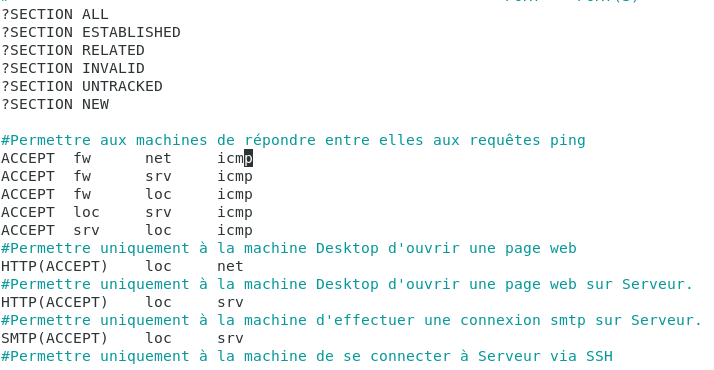


Puis on va configurer les fichiers :

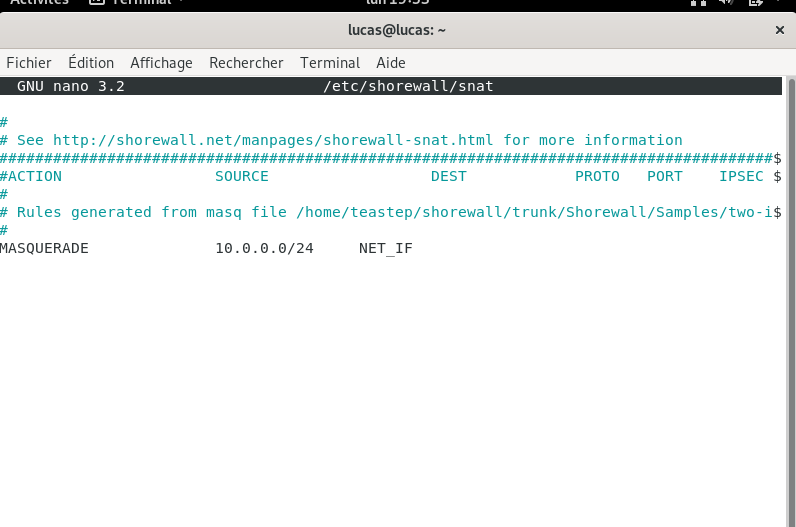








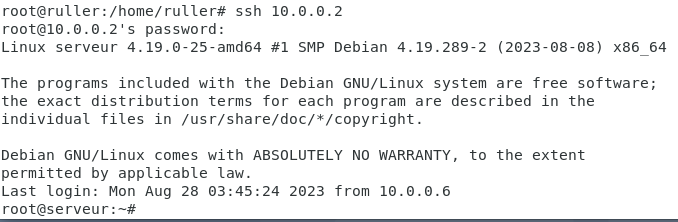
Ici ci-dessus c’est le fichier rules.

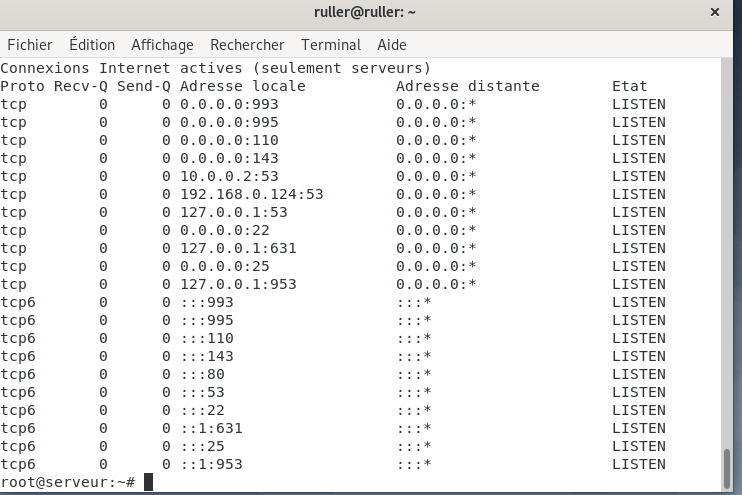


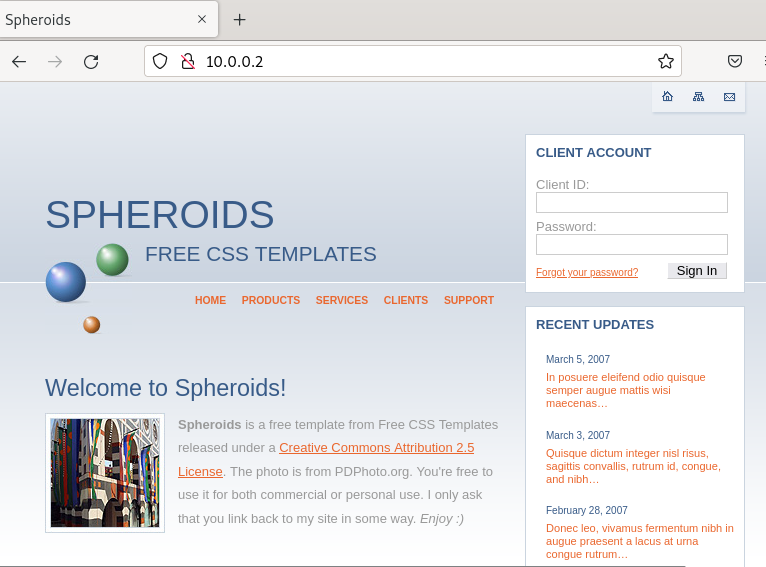
Puis on redémarre le service



On va effectuer les test :







Tout marche bien.

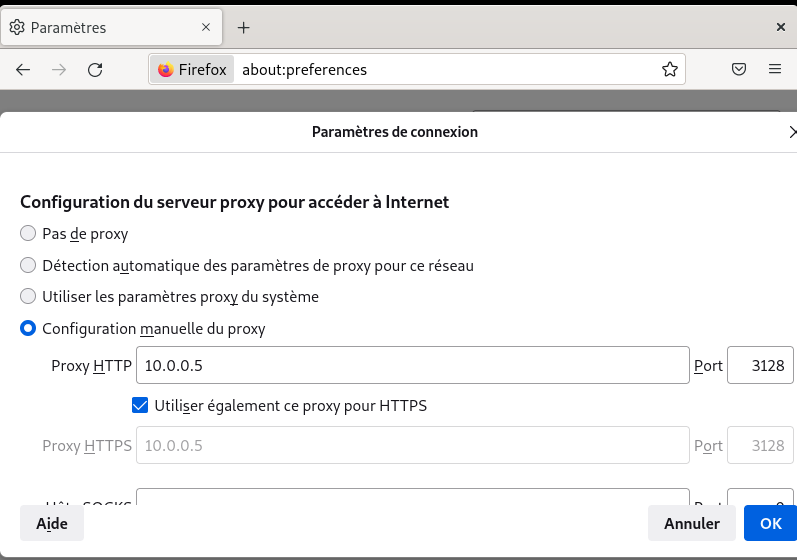
**Fiche de travail 7 : Proxy**

Un serveur proxy est un serveur intermédiaire qui agit en tant qu'intermédiaire entre un client et un autre serveur. Son rôle principal est de faciliter, contrôler et sécuriser les échanges de données entre ces deux parties.

Pour configurer un serveur proxy faut commencer par installer Squid:



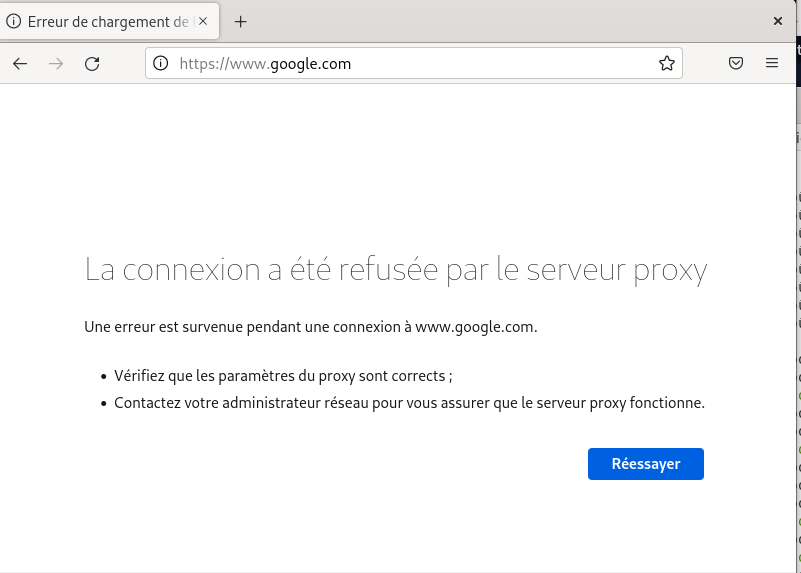
Puis on va dans la machine client (Desktop) pour configurer le proxy dans Firefox :



Pour que les modifications soient enregistré on redémarre le service Squid :



A présent le proxy est installé et n’autorise aucun trafic :



Pour être encore certain , je vais vérifier dans des logs en temps réel.

Avec cette commande :



On aura ceci :



Que ce site et si je teste même les autres sites sont bloqués.

On va configurer dans le fichier squid.conf:

