Alexandre VIALE

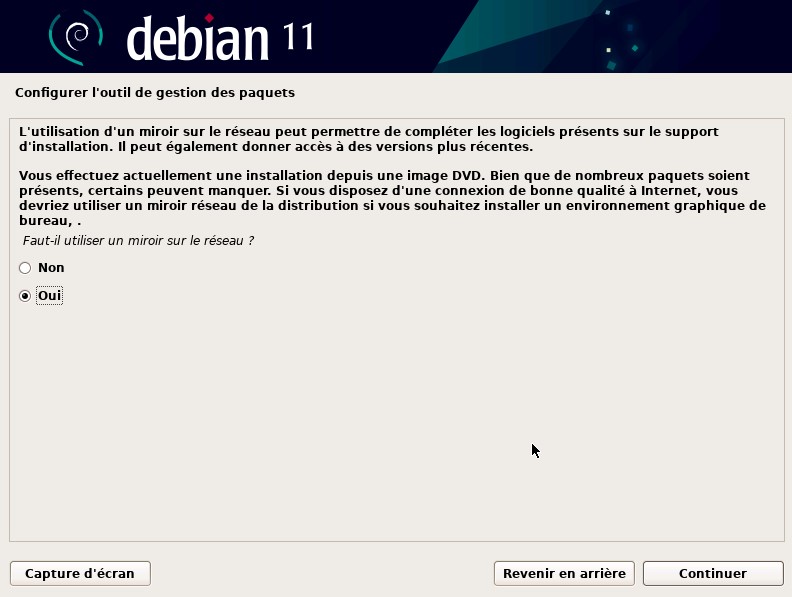
JOB 1 :

Pour le JOB 1 il suffit d’installer une VM debian mais il nous sera demandé de faire un serveur et pour cela quelques réglages sont à faire lors de l’installation de debian.

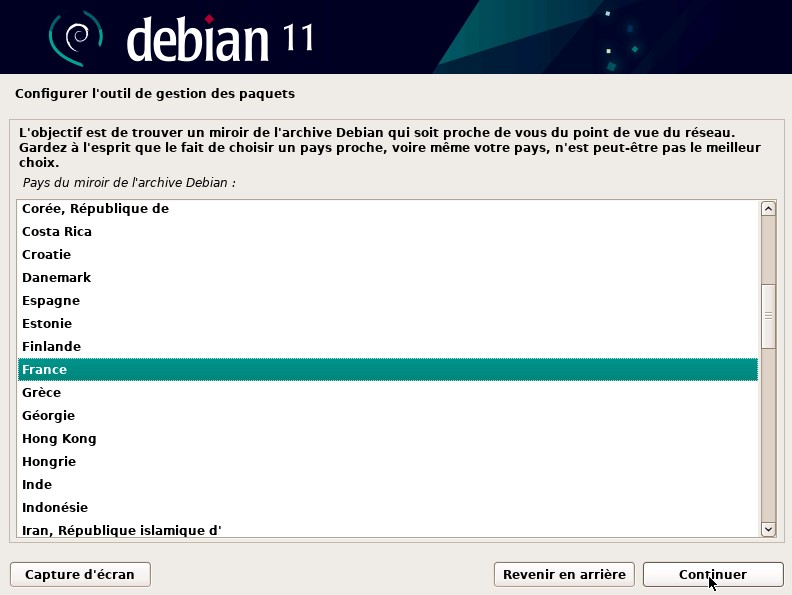
(Les étapes non spécifiés ci-dessous se font de manieres classique)

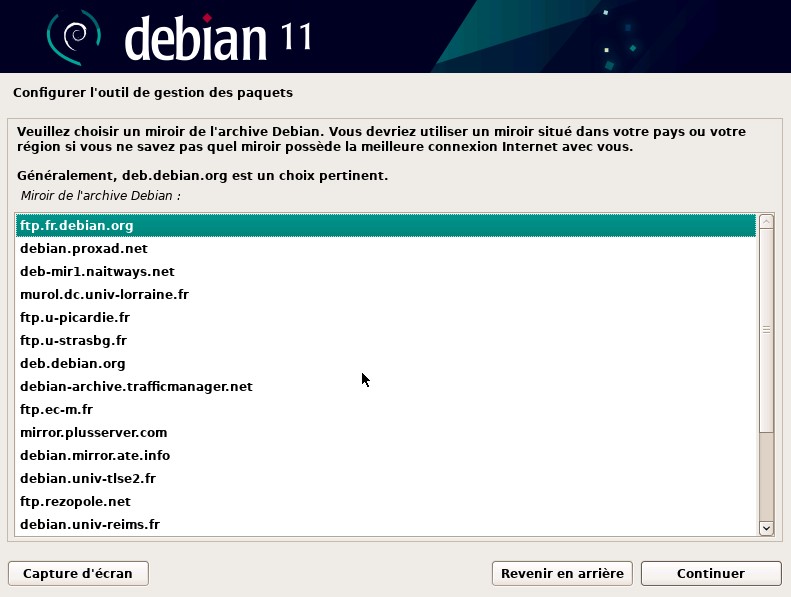
Voici les étapes importantes :

Choisir un miroir réseau permet de faciliter l’accés aux paquets et aux réseaux

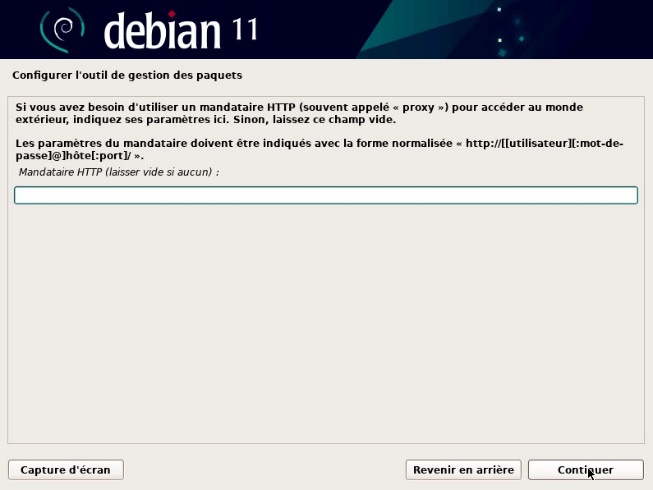


On choisir le miroir le plus proche de soit

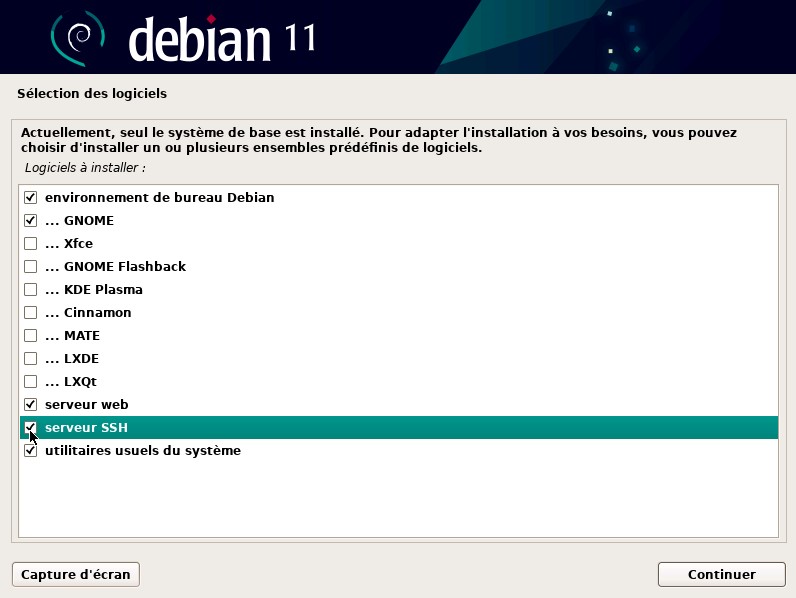


On choisit le serveur français

Ne pas mettre de mandataire

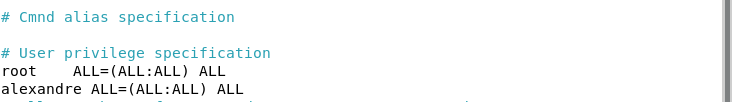


Laisser **environnement de bureau Debian, GNOME, utilitaires usuels du système.** Cocher **serveur web** et **serveur SSH** pour faciliter l’installation du serveur.



JOB 2 :

Pour le JOB 2 il faut installer Apache2 mais avant ça on va ajouté son utilisateur aux sudo pour cela on se me en root, on rentre dans le fichier visudo via la commande *nano sudo visudo* puis on ajoute son utilisateur ici



Ensuite on installe le paquet d’apache2 :

Dans le terminal :

*sudo apt-get update*

*sudo apt-get upgrade*

*sudo apt-get install apache2*

JOB 3

Il existe plusieurs programmes permettant d’utiliser un serveur web, la plupart utilise linux comme (Apache, Nginx, etc.). D’autre repose sur windows comme Microsoft IIS qui exclusif à Windows

**Serveur Apache**: le serveur Apache est un programme open source et qui donc évolue avec les utilisateurs.Il est le serveur Web le plus utilisé. Cependant dernièrement il a été rattrapé par la concurrence qui arrive à des temps de transfert très court

**Apache Tomcat**: Il s’agit ici aussi d’un programme open source. Il utilise sur un environnement Java. Il est donc très approprié aux personnes utilisant Java. De plus il peut se compléter avec d’autre serveur web comme apache via un connecteur AJP pour Apache JServ Protocol

**Nginx**: En plus des fonctions de serveurs web classique Nginx propose un proxy inverse qui donne l’accès à des serveurs internes, ce qui permet de travailler plus rapidement.

**Microsoft IIS**: Microsoft Internet Information Service (IIS) est un ensemble de logiciels dans lequel on trouve notamment un serveur Web. Il fonctionne de la même manière que les autres serveurs web mais est réservé aux utilisateurs windows.

**Serveur Web LiteSpeed**: Le serveur Web Litespeed se différencie surtout pour sa rapidité de transmission de données.

JOB 4 :

On va ici installer bind9 qui le logiciel pour serveur DNS le plus utilisé pour cela on va dans le terminal

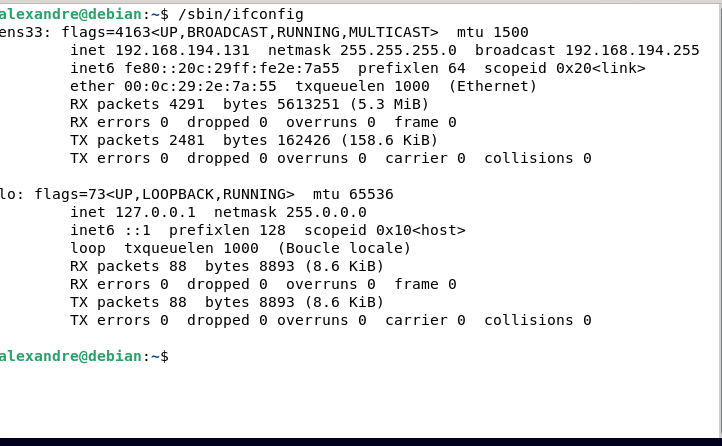
En ligne de commande :

*sudo apt-get update*

*sudo apt-get upgrade*

*sudo apt-get install -y bind9 bind9utils bind9-doc dnsutils*

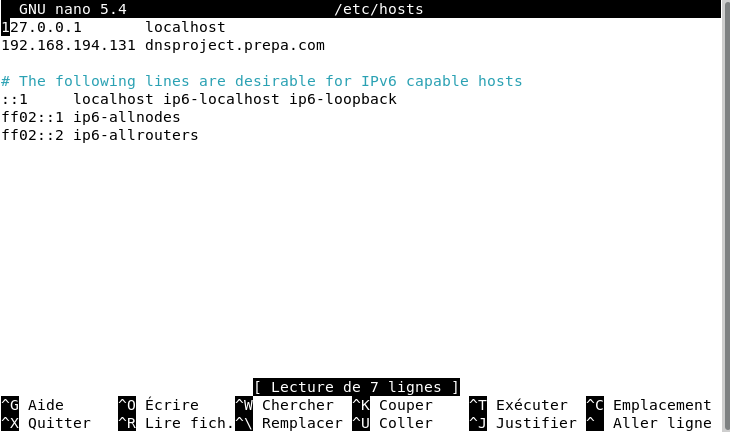
Ensuite on regarde son ip avec ifconfig,



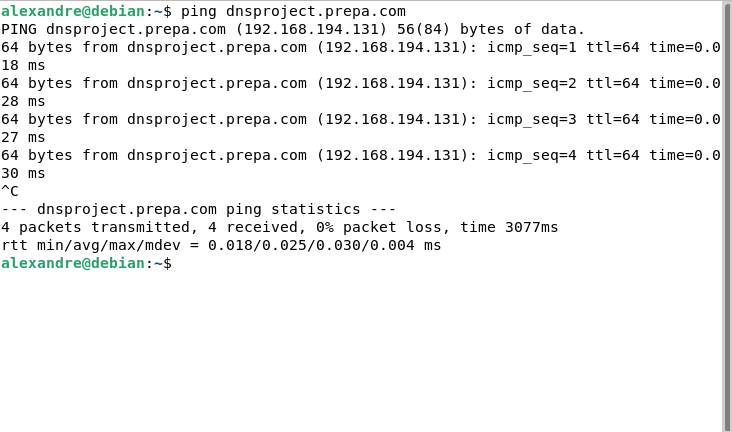
Dans mon cas c’est 192.168.194.131

Puis on rentre dans le fichier hosts via la commande sudo nano /etc/hosts

On remplace la deuxième ligne du fichier pour son ip et par le nom de domaine demandé



Voici le résultat d’un ping du nom de domaine



JOB 5

Premièrement il y a plusieurs types de nom de domaines, il y a le domaine géographique (.fr .de .eu).

Il y a des domaines génériques .com (pour des activités commerciales) .net (pour les entreprises) et .org (pour les associations) .

Pour obtenir un nom de domaine il faut s'adresser à l'organisme gestionnaire qui en a la charge.

Pour un .fr il faut s’adresse à Afnic pour un .org .com ou .net c’est Icann qu’il faut se tourner.

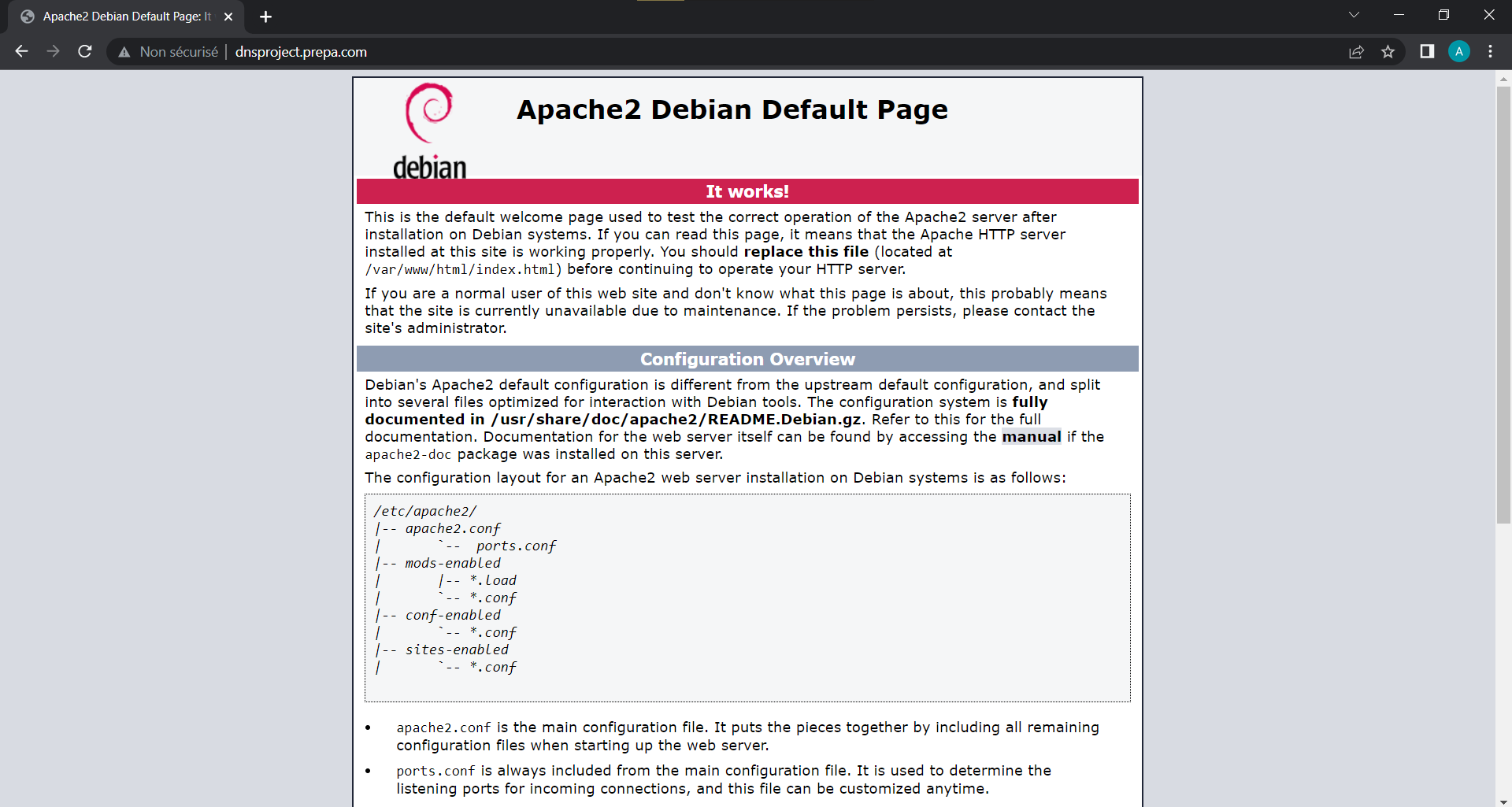
Le prix est d’environ une vingtaine d’euros par ans.

JOB 6 :

Pour ce JOB nous devons accéder à notre serveur via l’hôte pour cela on rajoute de la même manière dans le fichier hosts de l’hôte



On peut maintenant accéder à la page apache depuis l’hôte via l’ip ou le nom de domaine



Job 7 :

Pour le JOB 7 on va faire un serveur dhcp pour cela dans le terminal on va faire ceci

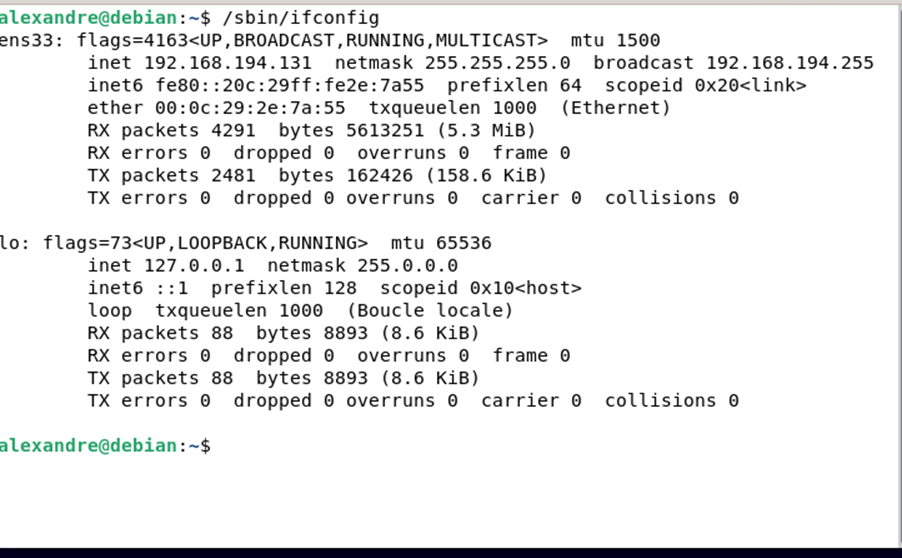
En ligne de commande :

*sudo apt-get udpade*

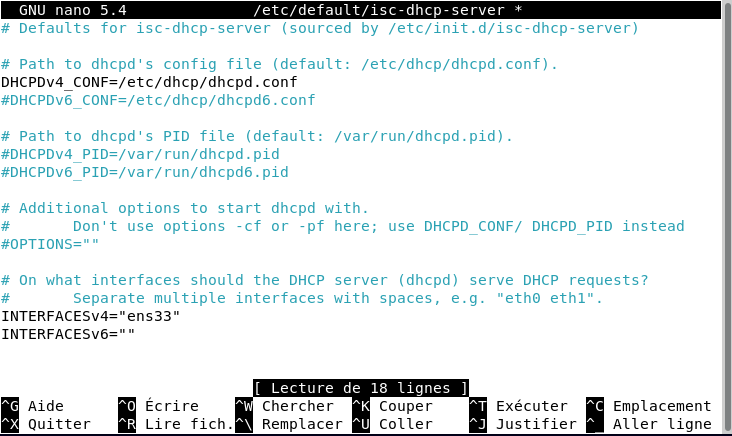
*sudo apt-get upgrade*

*sudo apt-get install isc-dhcp-server*

Ensuite on fait un ifconfig pour connaître l’interface que l’on utilise, ici c’est ens33



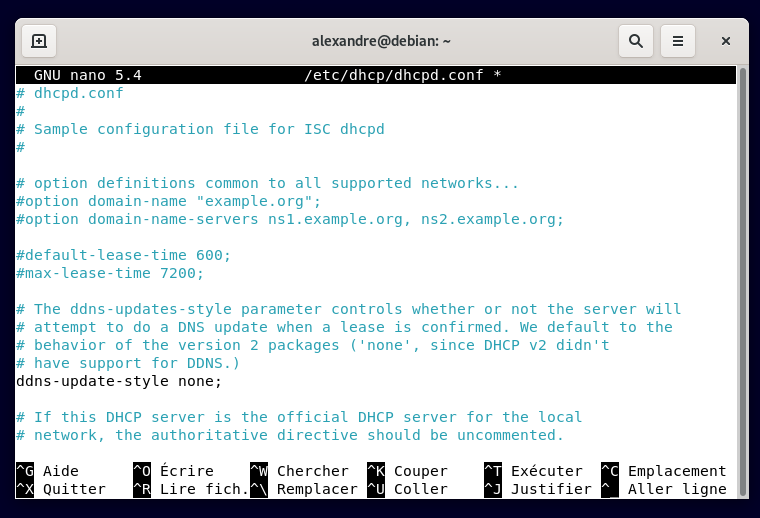
Puis on va dans le /etc/default/isc-dhcp-server pour préciser l’interface utilisé

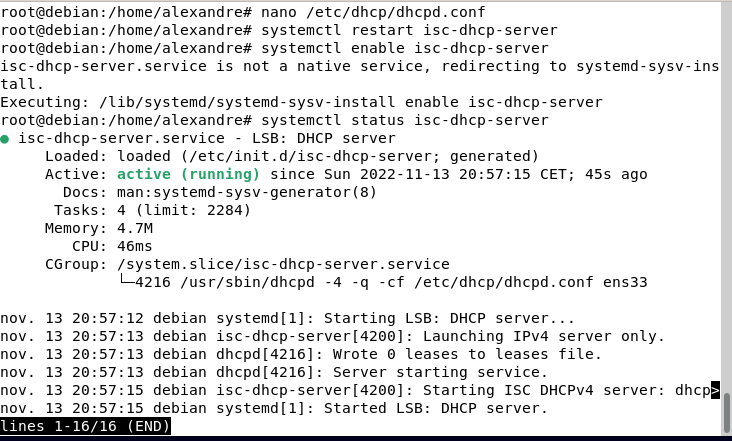
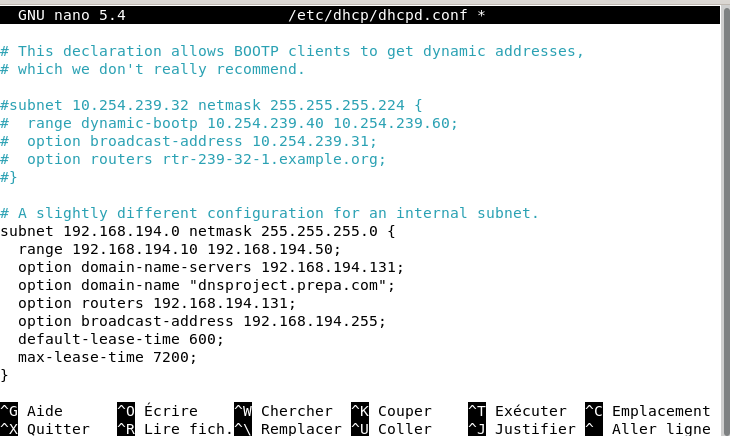


On va maintenant dans le fichier de configuration du dhcp via la commande :

*sudo nano /etc/dhcp/dhcpd.conf*

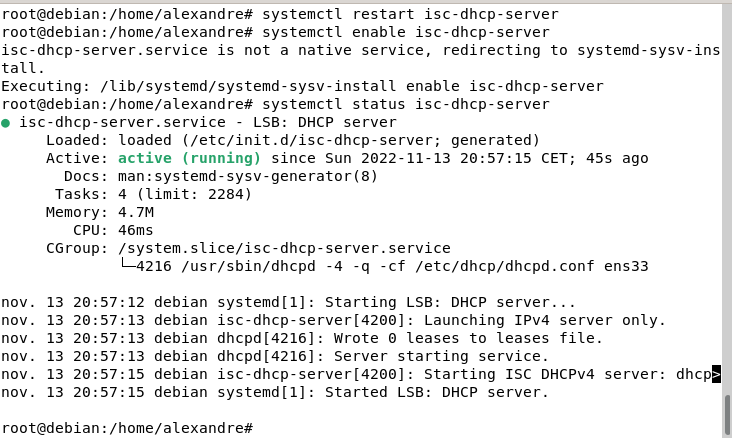
On commente les 4 premières lignes que ne l’étaient pas au départ car se sont des paramètres que nous allons préciser en dessous



On décommente ses lignes que l’on complète en fonction de l’ip de la machine.

On va maintenant relance le serveur pour qu’il prenne en compte les modifications avec la commande *systemctl restart isc-dhcp-server* puis on fait la commande *systemctl enable isc-dhcp-server* pour q’il se lance au démarrage

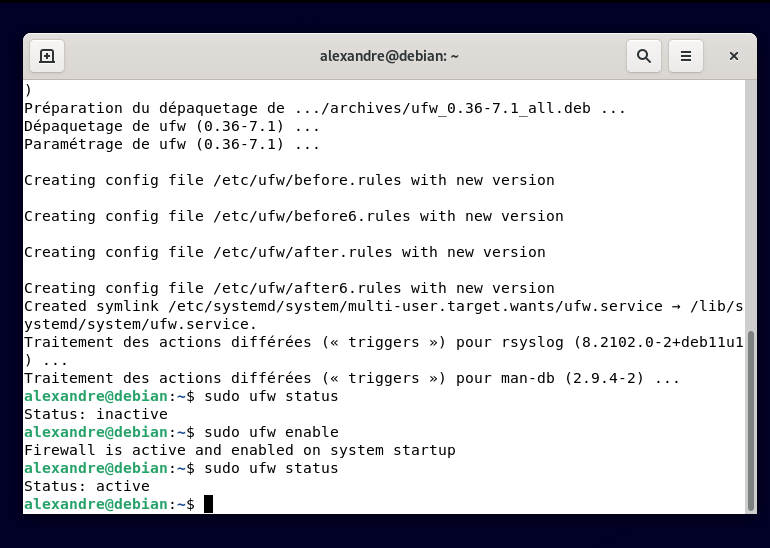
Enfin on fait *systemctl status isc-dhcp-server* pour vérifier que le serveur est bien fonctionnel

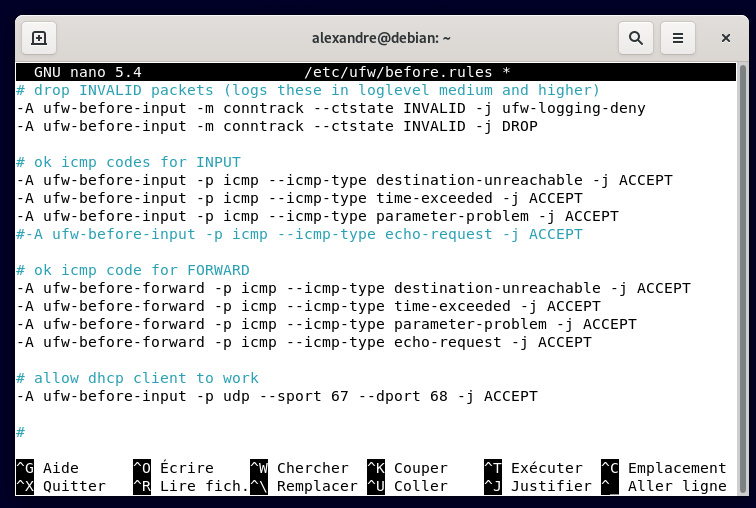


JOB 8

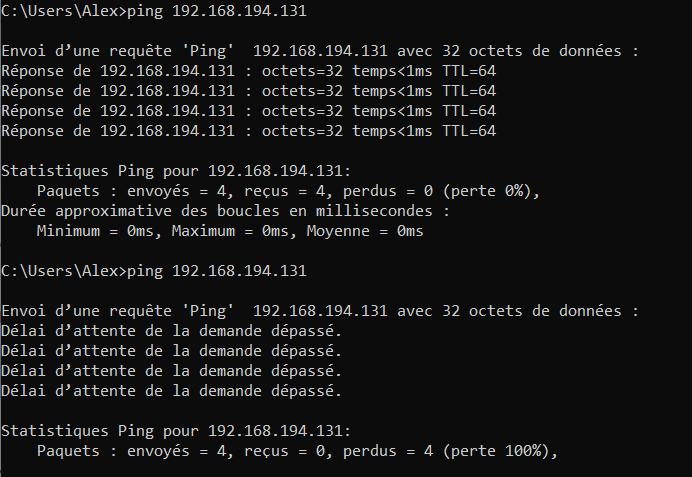
JOB 9

Pour le JOB 9 il faut mettre en place un pare feu pour cela dans le terminal de commande on tape *sudo ufw enable* pour activer le pare feu qui est désactiver par défaut pour vérifier qu’ils sont bien activés on fait *sudo ufw status*



Une fois le pare feu actif on va interdire le ping pour cela on va dans le fichier /etc/ufw/before.rules afin de commenter la ligne -A uf-before-input -p icmp –icmp-type echo-request -j ACCEPT

Voici deux pings du serveur le premier avant de commenter la ligne le second après que la ligne soit commentée et le pare feu relancer



JOB 10