

# 1 TP1

**Installation vm** Début de ce tp création d'une vm Debian. Une fois crée plusieurs chose a installer mais déjà faut passer en root pour pour pouvoir les utiliser Si on est pas root on utilise la commande suivante :

```
su root
```

installation de logiciel en utilisant cette commande :

```
aptitude install lynx sudo tcpdump vim
```

Une fois fait on doit mettre a jour le fichier de configuration pour l'utilisateur dans mon cas anthony puisse utiliser sudo. on fait :

```
visudo
```

Cela ouvre le fichier de conf /etc/sudoers. Et on peut rajouter la ligne :

```
anthony ALL=(ALL) ALL
```

On utilise visudo et no vim /etc/sudoers car visudo est un utilitaire de vérification du fichier. Il effectue une vérification de l'intégrité du fichier après modification avant de l'enregistrer. En cas d'erreur lors de la modification, le nouveau fichier n'est pas enregistré, ce qui vous évite de vous retrouver dans l'impossibilité de corriger votre erreur. Enfin, il s'assure que ce fichier conserve ses droits Unix originaux, ce qui garantit le bon fonctionnement de sudo. À la fermeture du fichier /etc/sudoers ouvert par son outil d'édition visudo, la nouvelle configuration est automatiquement chargée. Cela fonctionne en revenant en utilisateur normale et tester le sudo.

Ensuite on clone notre vm dans virtual box on peut cloner notre vm et on les clone en leur donnant gateway, client et serveur web.

Dans virtualbox il y a un bouton instantanés qui permet de faire une image de la vm. la fonction rm -rf / permet de supprimer la vm et donc en revant a notre instantanés on retrouve notre vm au départ.

On met la vm a avoir un accès par pont. Ensuite la commande ip addr montre la configuration réseau. La commande lynx permet d'ouvrir un navigateur en mode texte. Une fois ouvert on peut se connecter au réseau.

utilisation de lynx. Pour aller a une page particulière on fait :

```
lynx <url>
```

ou bien maj+ Gdans lynx. On peut ainsi se connecter a n'importe quelle site mais toute les pages seront en version ligne de commandes startpage.com est une page de research et ainsi on peut faire des recherches internet

## 2 Tp2

Dans la configuration de la gateway dans virtualbox on active 3 carte réseau. Apres on installe openssh via la commande

```
sudo aptitude install ssh
```

Il faut que la carte réseau soit configuré en accès par pont. On récupere l'adresse de la vm qui est dans mon cas 192.168.99.81 et on essaye de se connecter a la machine en faisant :

```
ssh utilisateur@192.168.99.81
```

En tant qu'utilisateur courant on peut se connecter mais si on est root cela ne fonctionne pas

Pour modifier le port d'écoute de openssh on fait man sshd\_config sur la vm et on arrive sur la page d'explication. on fait :

```
sudo nano /etc/ssh/sshd_config
```

Il faut modifier le fichier /etc/ssh/sshd\_config ou on le port 22 qui est configuré par défaut et on le change en 26.

Pour l'échange de clé ssh on fait d'abord ssh-keygen pour créer une clé si on en a pas. Une fois créer on fait la commande :

```
ssh-copy-id -i ~/.ssh/id_rsa.pub anthony@192.168.99.81
```

Cette commande copie la clé ssh vers la vm. Ensuite pour tester si cela fonctionne on fait :

```
ssh anthony@192.168.99.81 -p 26
```

On ajoute -p 26 car le port d'écoute de la vm est 26

Pour installer un serveur dhcp on utilise la commande :

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

Il faut aussi bien configurer la vm. Dans les configurations de la gateway on a 3 carte réseau une en accès par pont la deuxième en réseau nat et nat 1 et la troisième en réseau nat et nat2.

Il faut ensuite modifier le fichier dhcp.conf se trouvant etcdhcpd.conf. Ce fichier est celui de la gateway. On le modifie en récupérant la mac adress de la gateway et on y ajoute ceci :

```

# Le fichier contient (aide dans les commentaires)
ddns-update-style none;
# DNS de Google
# Si la machine sert de DNS
default-lease-time 600;
max-lease-time 7200;
authoritative;
log-facility local7;
subnet 192.168.1.0 netmask 255.255.255.0 {
    range 192.168.1.100 192.168.1.120;
    option subnet-mask 255.255.255.0;
    option broadcast-address 192.168.1.255;
    option routers 192.168.1.254;
    interface eth1;
}

subnet 192.168.2.0 netmask 255.255.255.0 {
    range 192.168.2.100 192.168.2.120;
    option subnet-mask 255.255.255.0;
    option broadcast-address 192.168.1.255;
    option routers 192.168.1.254;
    interface eth1;
}

subnet 192.168.2.0 netmask 255.255.255.0 {
    range 192.168.2.100 192.168.2.120;
    option subnet-mask 255.255.255.0;
    option broadcast-address 192.168.2.255;
    option routers 192.168.2.254;
    interface eth2;
}

host serverweb {
    hardware ethernet 08:00:27:7b:f1:a6;
    fixed-address 192.168.2.10;
}

```

Il faut modifier le fichier intreface se trouvant dans etcnetworkinterfaces

pour mettre en relation le dhcp et les interface.

```
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).
```

```
source /etc/network/interfaces.d/*
```

```
# The loopback network interface
```

```
auto lo
```

```
iface lo inet loopback
```

```
# The primary network interface
```

```
allow-hotplug eth0
```

```
iface eth0 inet dhcp
```

```
allow-hotplug eth1
```

```
iface eth1 inet static
```

```
    address 192.168.1.254
```

```
    netmask 255.255.255.0
```

```
allow-hotplug eth2
```

```
iface eth2 inet static
```

```
    address 192.168.2.254
```

```
    netmask 255.255.255.0
```

Une fois cela fait on redemarre le service networking de la gateway et du serveur web. La commande est :

```
service networking restart
```

On doit le faire plusieurs fois car le changement n'est pas pris en compte directement. une fois cela fait on verifie que l'adresse ip de la gateway et du serveur web soit les bonnes. Pour cela on utilise la commande :

```
ifconfig
```