

Procédure TP DHCP Relay

Objet :

Configuration d'un serveur Debian avec le rôle DHCP (Serveur) dans le LAN SRV (192.168.10.0/24) pour qu'il attribue à un client Debian dans le LAN Client (192.168.1.0/24) une adresse IP. Les deux LAN sont reliés par un routeur Debian qui a le rôle DHCP Relay.

Table des matières :

Routeur DEBIAN 11 :	2
Serveur Debian 11 (DHCP Serveur) :	8
Client Debian 11 :	11

Configuration du routeur Debian 11 :

Nous allons commencer par la création de la machine virtuelle (VM) que l'on nommera « Debian-routeur » avec la composition proposée par VMware Workstation, sauf avec la mémoire à 2GB. On laissera la VM en NAT dans un 1^{er} temps afin d'avoir accès à internet pour l'installation des différents services.

Une fois sur la VM, pour ma part le login et mot de passe sera « **root** ».

```
Debian GNU/Linux 11 debian tty1
debian login: root
Password: _
```

On commence par une mise à jour des paquets du système :

apt-get update

```
root@debian:~# apt-get update
Atteint :1 http://deb.debian.org/debian bullseye InRelease
Réception de :2 http://security.debian.org/debian-security bullseye-security InRelease [48,4 kB]
Réception de :3 http://deb.debian.org/debian bullseye-updates InRelease [44,1 kB]
Réception de :4 http://security.debian.org/debian-security bullseye-security/main Sources [170 kB]
Réception de :5 http://security.debian.org/debian-security bullseye-security/main amd64 Packages [271 kB]
Réception de :6 http://security.debian.org/debian-security bullseye-security/main Translation-en [175 kB]
708 ko réceptionnés en 0s (2 422 ko/s)
Lecture des listes de paquets... Fait
root@debian:~# _
```

Nous allons maintenant installer le service DHCP Relay avec la commande suivante :

apt-get install isc-dhcp-relay

```
root@debian:~# apt-get install isc-dhcp-relay
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances... Fait
Lecture des informations d'état... Fait
Les paquets supplémentaires suivants seront installés :
  libirs-export161 libiscfg-export163
Les NOUVEAUX paquets suivants seront installés :
  isc-dhcp-relay libirs-export161 libiscfg-export163
0 mis à jour, 3 nouvellement installés, 0 à enlever et 103 non mis à jour.
Il est nécessaire de prendre 809 ko dans les archives.
Après cette opération, 1 324 ko d'espace disque supplémentaires seront utilisés.
Souhaitez-vous continuer ? [O/n]
```

Puis faire « O » pour confirmer l'installation

Suite à cette installation, une information importante est à renseigner :

L'adresse de notre serveur DHCP (que nous allons paramétrer plus tard) :

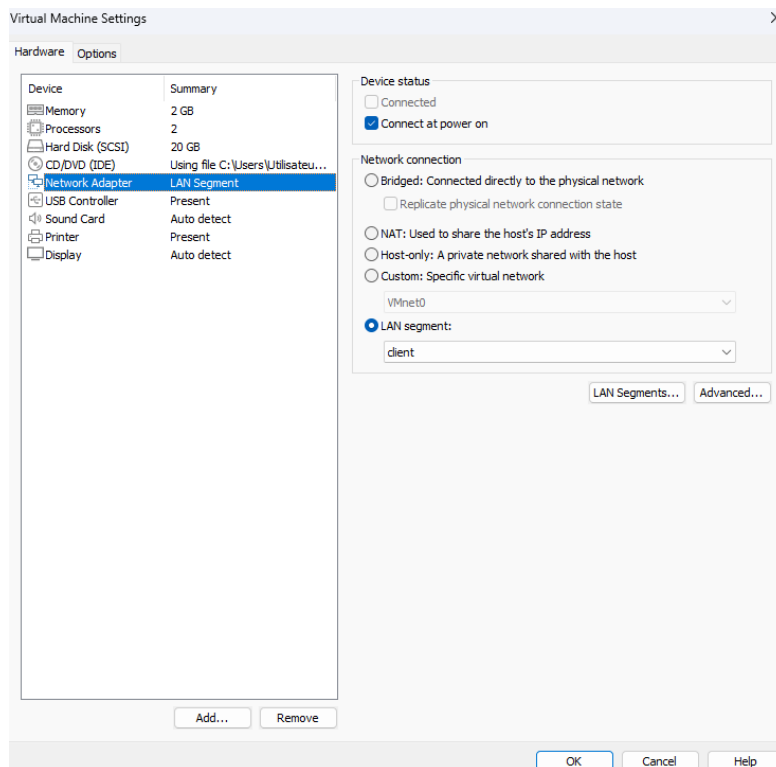
192.168.10.250

DHCP Relay	
Veuillez indiquer le nom ou l'adresse IP d'au moins un serveur DHCP auquel faire suivre les requêtes DHCP et BOOTP.	
Vous pouvez indiquer plus d'un serveur. Séparez les noms (ou les adresses IP) des serveurs par un espace.	
Serveurs DHCP auxquels faire suivre les requêtes de relais DHCP :	
192.168.10.250	
<Ok>	

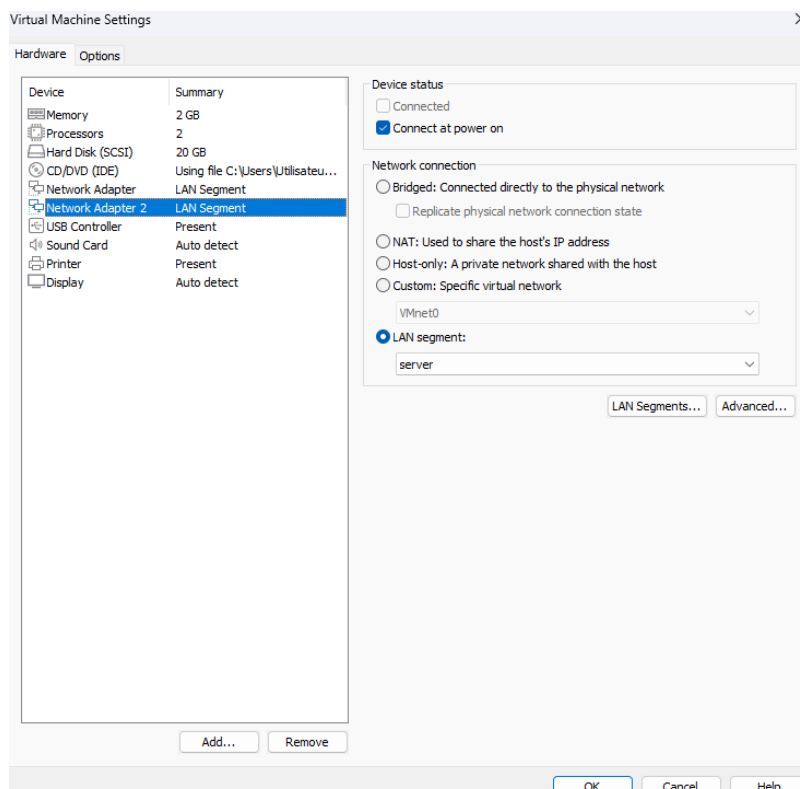
Pour les étapes suivantes, il suffira juste de les passer sans rien renseigner dedans.

A partir de ce moment, nous pouvons momentanément éteindre la VM afin de faire des modifications sur les adaptateurs réseaux.

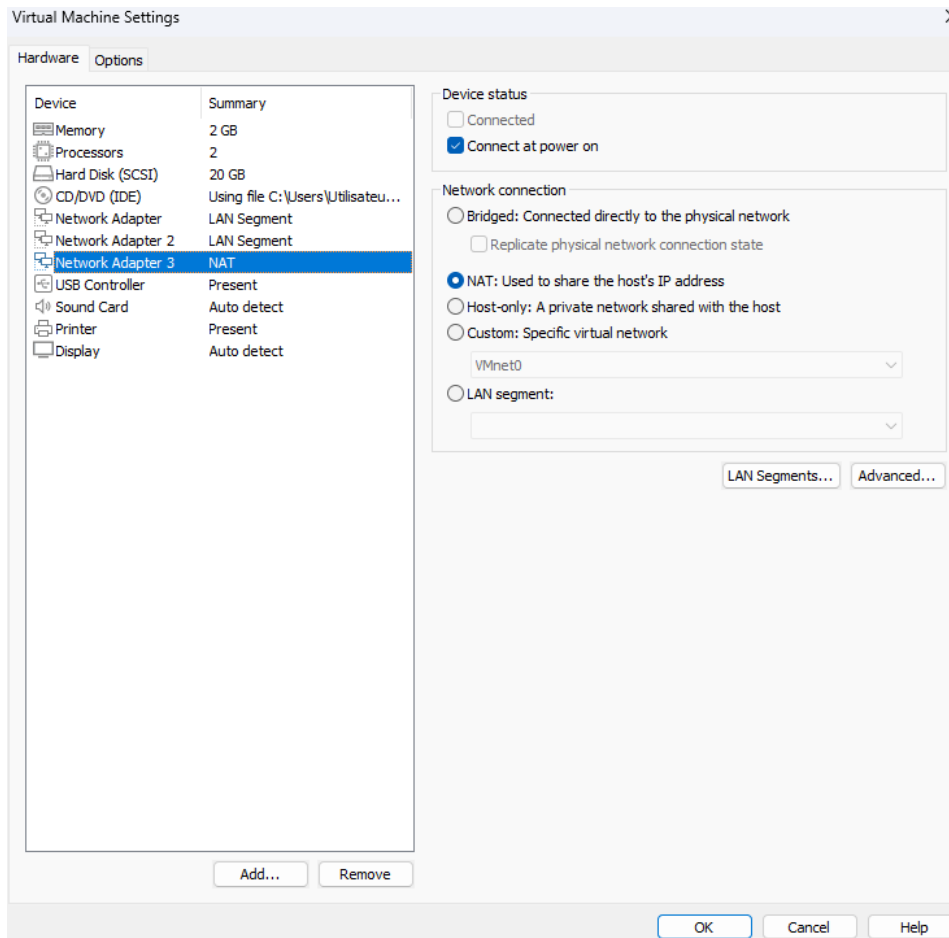
Pour la configuration réseau du routeur dans VMware WorkStation, nous allons sélectionner le LAN segment « **CLIENT** » pour la première carte réseau.



Nous allons ensuite ajouter une seconde carte réseau au routeur et créer le LAN segment « **SERVER** ».



Et en ajouter une troisième en la connectant au réseau NAT WorkStation.



Nous pouvons redémarrer la VM.

Nous allons maintenant configurer les interfaces réseaux ainsi que la mise en place du routage :

`nano /etc/network/interfaces`

Dans ce fichier, nous allons renseigner nos interfaces réseau :

```
#CLIENT
allow-hotplug ens33
iface ens33 inet static
address 192.168.1.254
netmask 255.255.255.0

#SERVER
allow-hotplug ens36
iface ens36 inet static
address 192.168.10.254
netmask 255.255.255.0

#NAT
allow-hotplug ens37
iface ens37 inet static
address 192.168.50.254
netmask 255.255.255.0
gateway 192.168.50.2
```

On quitte le fichier sans oublier de sauvegarder et sans changer le nom.

Ensuite pour mettre à jour les interfaces réseau, il suffit de faire la commande « ifdown ens33 » et « ifup ens33 » pour ma part et ainsi de suite avec toutes les interfaces réseau changées.

Une fois cette commande effectuée, nous pouvons voir le changement.

```
root@debian:~# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: ens33: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 00:0c:29:98:0c:4b brd ff:ff:ff:ff:ff:ff
    altname enp2s1
    inet 192.168.1.254/24 brd 192.168.1.255 scope global ens33
        valid_lft forever preferred_lft forever
    inet6 fe80::20c:29ff:fe98:c4b/64 scope link
        valid_lft forever preferred_lft forever
3: ens36: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 00:0c:29:98:0c:55 brd ff:ff:ff:ff:ff:ff
    altname enp2s4
    inet 192.168.10.254/24 brd 192.168.10.255 scope global ens36
        valid_lft forever preferred_lft forever
    inet6 fe80::20c:29ff:fe98:c55/64 scope link
        valid_lft forever preferred_lft forever
4: ens37: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 00:0c:29:98:0c:5f brd ff:ff:ff:ff:ff:ff
    altname enp2s5
    inet 192.168.50.254/24 brd 192.168.50.255 scope global ens37
        valid_lft forever preferred_lft forever
    inet6 fe80::20c:29ff:fe98:c5f/64 scope link
        valid_lft forever preferred_lft forever
root@debian:~#
```

Nous allons mettre en place le routage à l'aide de la commande suivante :

```
nano /etc/sysctl.conf
```

```
GNU nano 5.4 /etc/sysctl.conf
#
# /etc/sysctl.conf - Configuration file for setting system variables
# See /etc/sysctl.d/ for additional system variables.
# See sysctl.conf (5) for information.
#
#kernel.domainname = example.com
# Uncomment the following to stop low-level messages on console
#kernel.printk = 3 4 1 3
#####
# Functions previously found in netbase
#
# Uncomment the next two lines to enable Spoof protection (reverse-path filter)
# Turn on Source Address Verification in all interfaces to
# prevent some spoofing attacks
#net.ipv4.conf.default.rp_filter=1
#net.ipv4.conf.all.rp_filter=1
#
# Uncomment the next line to enable TCP/IP SYN cookies
# See http://lwn.net/Articles/277146/
# Note: This may impact IPv6 TCP sessions too
#net.ipv4.tcp_syncookies=1
# Uncomment the next line to enable packet forwarding for IPv4
net.ipv4.ip_forward=1
```

Sur cette ligne nous allons supprimer le # et quitter le fichier en enregistrant.

```
net.ipv4.tcp_syncookies=1
# Uncomment the next line to enable packet forwarding for IPv4
net.ipv4.ip_forward=1
```

Afin d'appliquer les changements, nous allons exécuter la commande suivante :

```
sysctl-p /etc/sysctl.conf
```

Pour finir, nous allons activer le **POSTROUTING**, ce qui va permettre au poste client de pouvoir communiquer avec Internet.

```
apt install iptables-y
```

```
iptables-t nat -A POSTROUTING -o ens37 -j MASQUERADE
```

Le paramétrage de notre routeur est terminé, nous allons quand même laisser la VM allumée.

Configuration du Serveur Debian 11 :

On commence donc par la création d'une première VM que l'on nommera « **Server-DHCP** » avec la composition par défaut proposée par VMware Workstation sauf la mémoire à 2 GB.

On laissera notre VM en NAT dans un 1er temps afin d'avoir l'accès à internet et d'installer les différents services notamment les mises à jour et isc-dhcp-server.

1) Paramétrage du serveur DHCP :

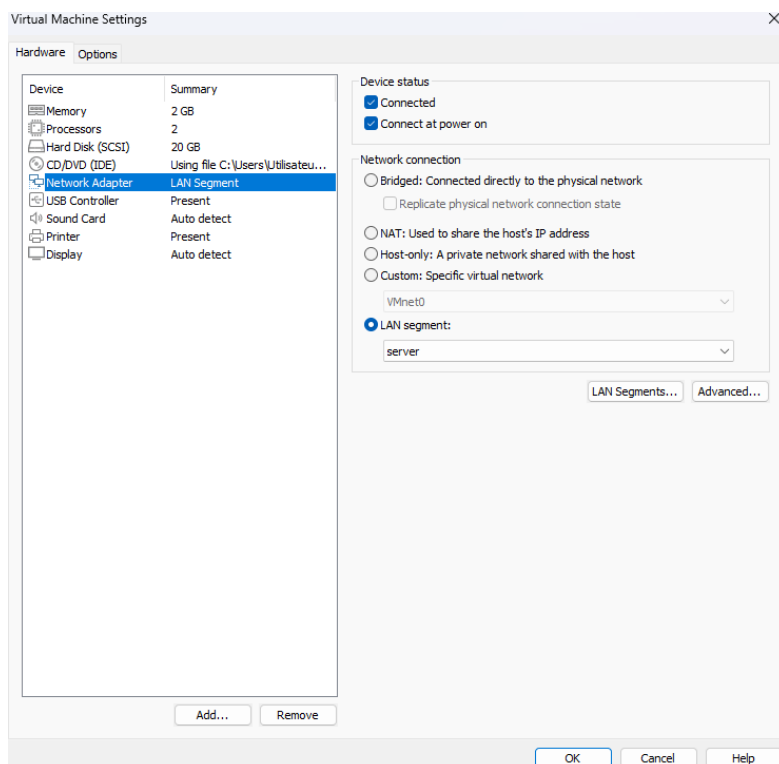
Après l'installation de Debian, on commence par une mise à jour des paquets du système :

```
apt-get update
```

Nous allons maintenant installer le service DHCP Serveur avec la commande suivante :

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

Nous allons ensuite éteindre notre VM afin de modifier l'adaptateur réseau qui est actuellement en NAT, pour le remplacer par un LAN Segment (server) que nous avons créé précédemment.



Nous allons ensuite démarrer notre VM et fixer l'adresse IP de notre serveur DHCP :

```
nano /etc/network/interfaces
```

Dans ce fichier vous allez faire les modifications comme ci-dessous (pensez à vérifier votre interface réseau, pour ma part elle se nomme **ens33**)

```
# The primary network interface
allow-hotplug ens33
iface ens33 inet static
address 192.168.10.254
netmask 255.255.255.0
gateway 192.168.10.254
```

On quitte le fichier en l'enregistrant ainsi que sans modifier le nom.

Ensuite pour mettre à jour les interfaces réseau, il suffit de faire la commande « **ifdown ens33** » et « **ifup ens33** » pour ma part et ainsi de suite avec toutes les interfaces réseau changées.

Nous allons maintenant finir de configurer le serveur DHCP :

```
nano /etc/default/isc-dhcp-server
```

Pour cela, il suffit de rajouter entre guillemets le nom de l'interface réseau (« **ens33** »)

```
GNU nano 5.4 /etc/default/isc-dhcp-server
# Defaults for isc-dhcp-server (sourced by /etc/init.d/isc-dhcp-server)

# Path to dhcpd's config file (default: /etc/dhcp/dhcpd.conf).
#DHCPDv4_CONF=/etc/dhcp/dhcpd.conf
#DHCPDv6_CONF=/etc/dhcp/dhcpd6.conf

# Path to dhcpd's PID file (default: /var/run/dhcpd.pid).
#DHCPDv4_PID=/var/run/dhcpd.pid
#DHCPDv6_PID=/var/run/dhcpd6.pid

# Additional options to start dhcpd with.
# Don't use options -cf or -pf here; use DHCPD_CONF/ DHCPD_PID instead
#OPTIONS=""

# On what interfaces should the DHCP server (dhcpd) serve DHCP requests?
# Separate multiple interfaces with spaces, e.g. "eth0 eth1".
INTERFACESv4=""
INTERFACESv6=""
```

```
# On what interfaces sh
# Separate multi
INTERFACESv4="ens33"
#INTERFACESv6=""
```

Nous allons commenter « **INTERFACESv6** » étant donné qu'elle ne nous est pas utile car il n'y a pas d'IPv6.

On quitte le fichier en l'enregistrant.

C'est dans le fichier qui va suivre que l'on indiquera les configurations IP à fournir :

nano /etc/dhcp/dhcpd.conf

```
# This is a very basic subnet declaration.

subnet 192.168.1.0 netmask 255.255.255.0 {
    range 192.168.1.100 192.168.1.200;
    option routers 192.168.1.254;
}
subnet 192.168.10.0 netmask 255.255.255.0 {
}
```

On quitte le fichier en l'enregistrant et sans modifier le nom du fichier.

Nous allons redémarrer le service DHCP à l'aide de la commande suivante :

service isc-dhcp-server restart

Nous avons terminé avec notre serveur DHCP, nous allons maintenant tester sur notre client Debian bien évidemment laisser notre VM **Server-DHCP** allumé.

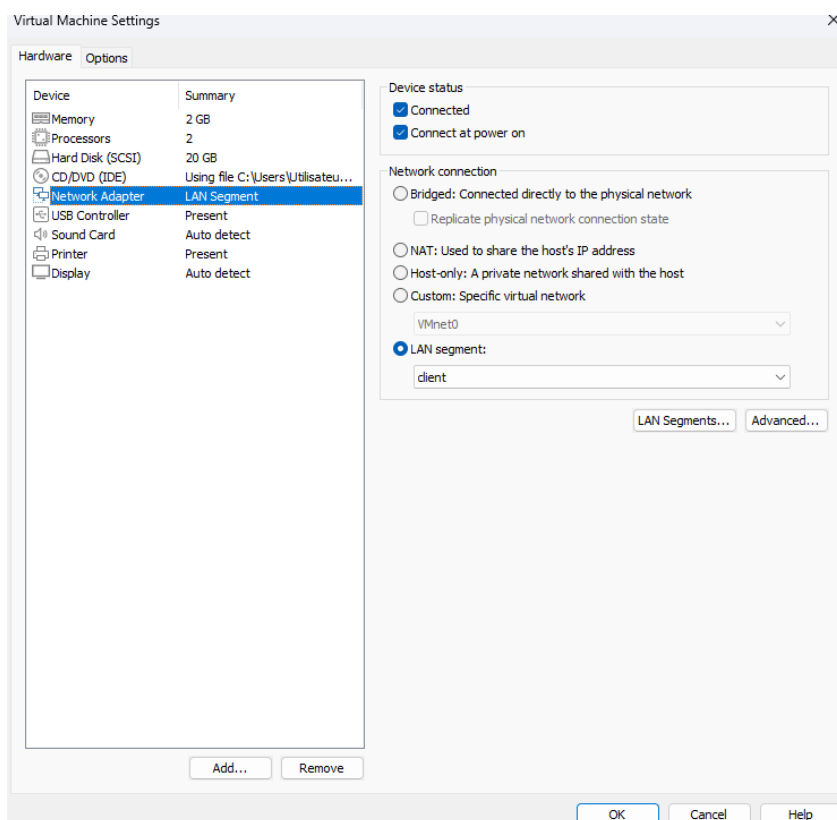
Configuration du Client Debian 11 :

On commence donc par la création d'une première VM que l'on nommera « **Debian-client** » avec la composition par défaut proposée par VMware Workstation sauf la mémoire à 2 GB. On laissera notre VM en NAT dans un 1er temps afin d'avoir l'accès à internet et de pouvoir installer les mises à jour.

Après l'installation de Debian, on commence par une mise à jour des paquets du système :

apt-get-update

Nous allons ensuite éteindre notre VM afin de modifier l'adaptateur réseau qui est actuellement en NAT par un LAN Segment (client) que nous avons créé précédemment.



Pour vérifier la bonne attribution de l'adresse IP par le serveur DHCP au client, je vais effectuer un **ip a**

L'adresse obtenue doit se situer entre 192.168.1.100 et 192.168.1.200.

Pour finir, je vais voir si la VM client peut bien communiquer avec l'extérieur. Pour cela, je vais faire un ping de google.fr (8.8.8.8).