

# **FTP - DNS - DHCP - SSH**



## **SOMMAIRE :**

**Page n°2 : Installation Debian sans interface graphique**

**Page n°5 : Mise à jour des systèmes**

**Page n°6 : Installation et configuration du serveur DHCP**

**Page n°10 : Installation et configuration du serveur FTP et SSH**

**Page n°13 : Installation et configuration du serveur DNS**

**Page n°15 : Test de la connexion au serveur SFTP**

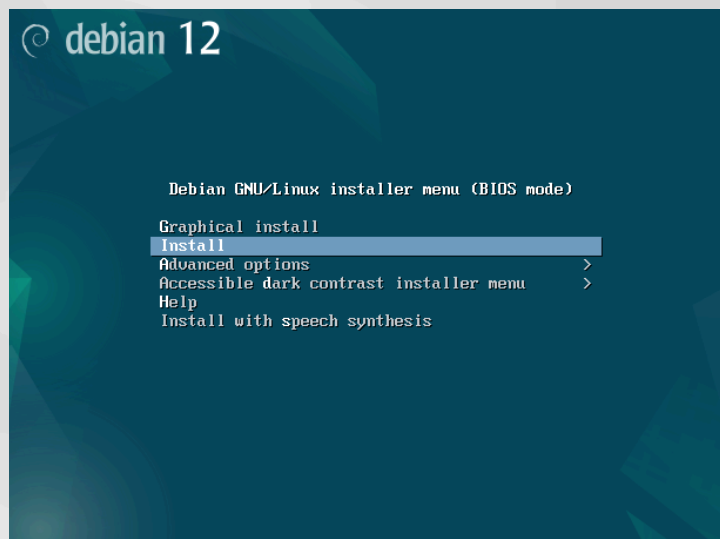
**Page n°16 : Paramètres de sécurités additionnelles**

# INSTALLATION DEBIAN SANS INTERFACE GRAPHIQUE

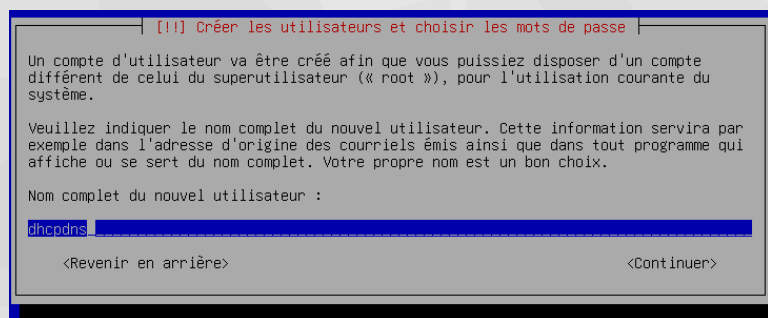
Pour commencer, nous téléchargeons l'ISO de Debian depuis leur site pour obtenir la dernière version disponible.

Ensuite, après avoir mis cela en place, nous utilisons un hyperviseur de notre choix, dans ce cas VMWare, pour créer deux machines virtuelles.

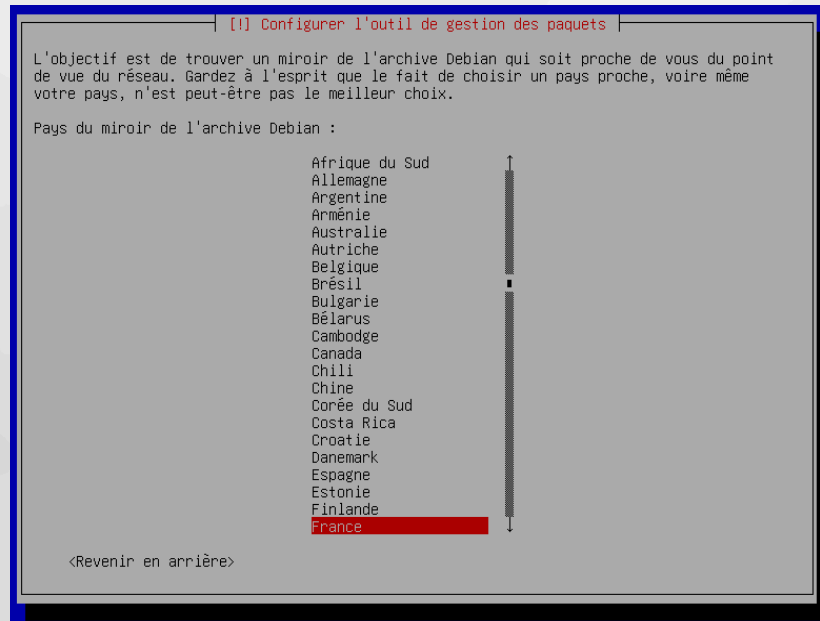
La première sera nommée "Machine 1" et héberge le serveur [DHCP/DNS](#), tandis que la deuxième sera nommée "Machine 2" et héberge le serveur [FTP/SSH](#). L'installation des deux machines virtuelles est similaire, vous pouvez donc suivre les étapes suivantes pour les deux.



Après avoir lancé notre machine virtuelle, nous tombons sur ce menu et nous choisissons d'installer Debian sans l'interface graphique.



Après avoir choisi la langue de l'installation et les autres paramètres, nous créons un utilisateur qui aura pour nom "[dhcpcdns](#)" afin de le distinguer facilement de l'utilisateur de la machine 2.

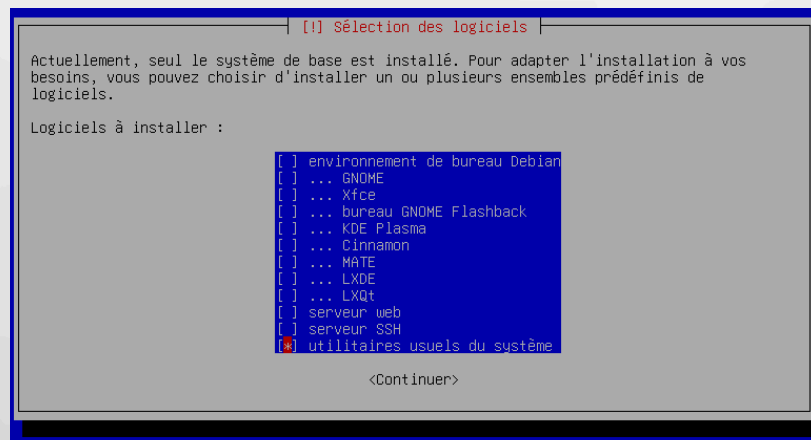


Nous continuons l'installation en configurant le réseau et en choisissant le miroir de l'archive Debian. Nous sélectionnons le pays "France" pour obtenir une liste de miroirs locaux.



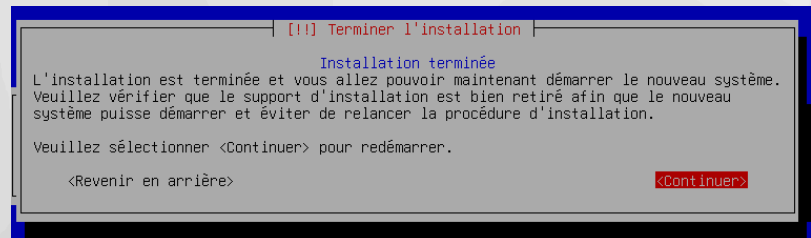
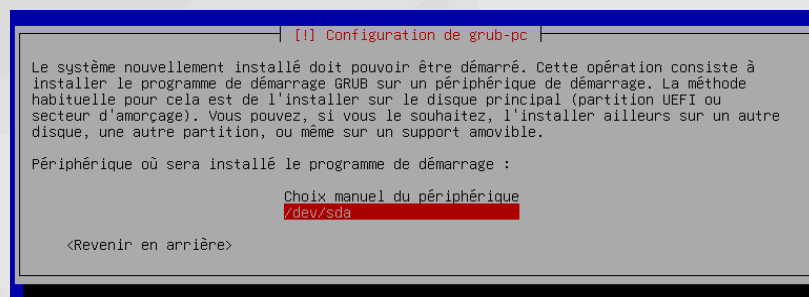
Ensuite, nous choisissons "deb.debian.org" comme miroir par défaut car il est recommandé pour les utilisateurs français et il est connu pour sa fiabilité et sa rapidité.

En choisissant un miroir proche de notre emplacement géographique, nous réduisons le temps de téléchargement des paquets et améliorons la vitesse d'installation de Debian.



Ensuite, nous nous assurons de ne pas installer d'interface graphique comme la consigne nous le demande.

Pour ce faire, nous décochons les options "GNOME" et "environnement de bureau Debian" à l'aide de la touche "Espace". Cela permet de ne pas installer de bureau graphique et de conserver une installation minimale de Debian.



Enfin, nous terminons l'installation en installant GRUB sur le disque principal (sur "/dev/sda" pour être précis). GRUB est un chargeur d'amorçage qui permet de démarrer le système d'exploitation Debian. Une fois GRUB installé, nous démarrons la machine virtuelle et nous pouvons maintenant utiliser notre machine Debian sans interface graphique, prête à être configurée pour héberger les services **DHCP/DNS ou FTP/SSH**.



## MISE À JOUR DES SYSTÈMES

```
dhcpcdns@dhcpcdns:~$ su
Mot de passe :
root@dhcpcdns:/home/dhcpcdns# apt update
Atteint :1 http://deb.debian.org/debian bookworm InRelease
Atteint :2 http://security.debian.org/debian-security bookworm-security InRelease
Atteint :3 http://deb.debian.org/debian bookworm-updates InRelease
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances... Fait
Lecture des informations d'état... Fait
Tous les paquets sont à jour.
root@dhcpcdns:/home/dhcpcdns# apt upgrade -y
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances... Fait
Lecture des informations d'état... Fait
Calcul de la mise à jour... Fait
0 mis à jour, 0 nouvellement installés, 0 à enlever et 0 non mis à jour.
root@dhcpcdns:/home/dhcpcdns# _
```

Comme indiqué sur le screenshot ci-dessous, nous avons mis à jour la machine virtuelle "Machine 1" (qui héberge le serveur DHCP/DNS) à l'aide des commandes

**"apt update" et "apt upgrade -y".**

Avant de lancer ces commandes, nous sommes passés en root grâce à la commande **"su"** pour éviter tout problème de permission.

Le screenshot montre que la commande **"apt update"** a été exécutée avec succès et que la liste des paquets disponibles a été mise à jour.

Ensuite, la commande **"apt upgrade -y"** a été exécutée pour mettre à jour tous les paquets installés sur la machine virtuelle vers leurs dernières versions disponibles.

L'option **"-y"** a été utilisée pour répondre automatiquement **"oui"** à tout les invites de confirmation pendant le processus de mise à jour.

Il est important de maintenir les machines virtuelles à jour pour garantir la sécurité et la stabilité des services qu'elles hébergent.

Nous suivrons le même processus de mise à jour pour la machine virtuelle "Machine 2" (qui héberge le serveur FTP/SSH).

# INSTALLATION ET CONFIGURATION DU SERVEUR DHCP

Nous commençons par la première machine virtuelle, où nous allons installer le serveur DHCP à l'aide de la commande ci-dessous.

```
root@dhcpdns:/home/dhcpdns# apt install isc-dhcp-server -y_
```

Cette commande permet d'installer le paquet "isc-dhcp-server" qui contient le serveur DHCP que nous allons configurer pour attribuer automatiquement des adresses IP aux clients du réseau.

Ensuite, nous avons utilisé la commande `ip a` pour trouver le nom de notre interface réseau, notre adresse IP et d'autres informations essentielles pour le reste de l'installation.

```
2: ens160: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 00:0c:29:eb:8f:fd brd ff:ff:ff:ff:ff:ff
    altname enp3s0
    inet 172.16.0.3/16 brd 172.16.255.255 scope global ens160
        valid_lft forever preferred_lft forever
    inet6 fe80::20c:29ff:feeb:8ffd/64 scope link
        valid_lft forever preferred_lft forever
```

Nous avons constaté que le nom de notre interface réseau était "ens33". Nous avons donc utilisé la commande `nano /etc/default/isc-dhcp-server` pour accéder au fichier et ajouter "ens33" dans la section INTERFACESv4.

```
GNU nano 7.2 /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="ens33"
INTERFACESv6=""
```

Maintenant, nous allons configurer le serveur DHCP pour attribuer des adresses IP.  
Pour ce faire, nous utilisons la commande **nano /etc/dhcp/dhcpd.conf**.

Grâce aux informations fournies par la commande **ip a**, nous pouvons ajouter les lignes suivantes dans le fichier

```
subnet 172.16.0.0 netmask 255.255.0.0 {  
    range 172.16.0.2 172.16.254.254;  
    option routers 172.16.0.2;  
    option subnet-mask 255.255.0.0;  
    option broadcast-address 172.16.255.255;  
}
```

Ces lignes définissent un sous-réseau avec une plage d'adresses IP disponibles pour les clients DHCP, ainsi que des options pour les routeurs et les serveurs DNS.

Ensuite, pour éviter les problèmes de réseau, nous allons attribuer une adresse IP fixe à la machine hébergeant le serveur DHCP.

Une fois que nous avons identifié l'interface réseau, nous devons ouvrir le fichier `/etc/network/interfaces` en utilisant un éditeur de texte. Nous recherchons ensuite la section correspondant à l'interface réseau que nous avons identifiée précédemment.

Dans cette section, nous devons modifier la ligne `iface ens33 inet dhcp` pour attribuer une adresse IP fixe à l'interface réseau.

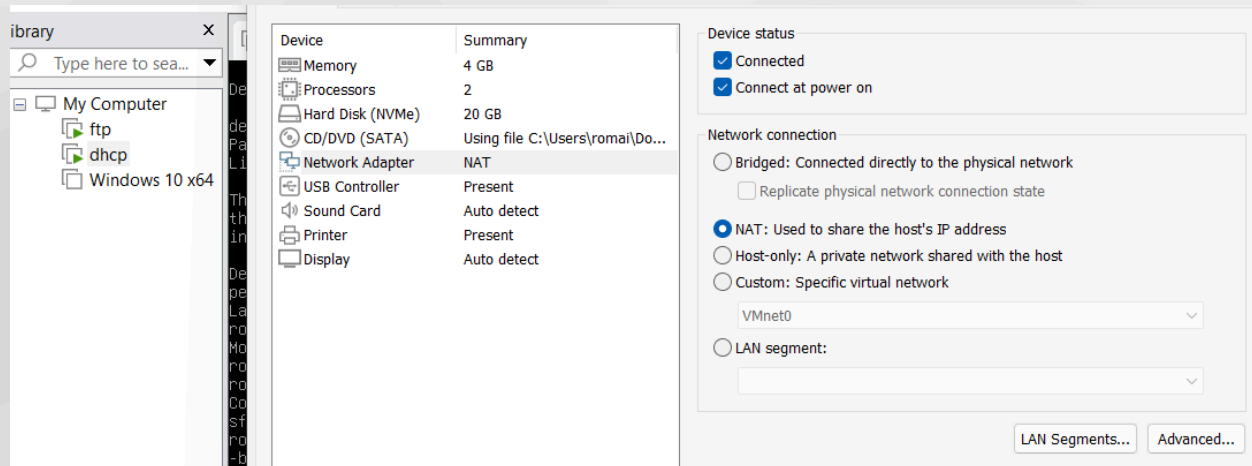
```
allow-hotplug ens160  
iface ens160 inet static  
    address 172.16.0.3  
    netmask 255.255.0.0  
    gateway 172.16.0.2  
    broadcast 172.16.255.255
```

Il est important de remplacer `"ens33"` par le nom de l'interface réseau correspondante et de modifier les valeurs de `"address"`, `"netmask"`, `"gateway"` et `"dns-nameservers"` en fonction de la configuration réseau.

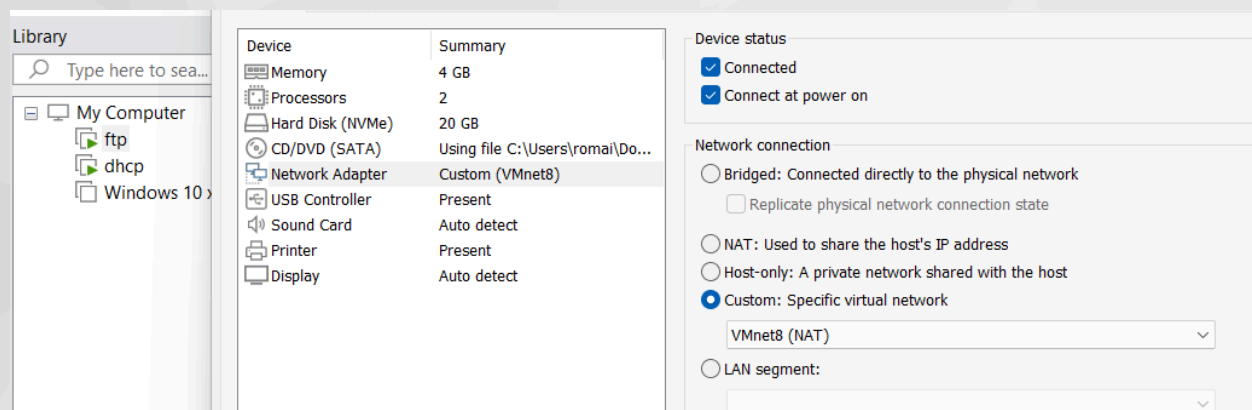
Enfin, nous enregistrons le fichier et redémarrons le service réseau pour appliquer les modifications. Cette étape est cruciale pour s'assurer que la machine hébergeant le serveur DHCP a une adresse IP fixe et stable, ce qui évitera les problèmes de réseau ultérieurs.



Il nous reste plus qu'à aller sur notre application vmware et modifier le network adapter de la vm ayant le serveur DHCP en sélectionnant " NAT : used to share the host's IP address



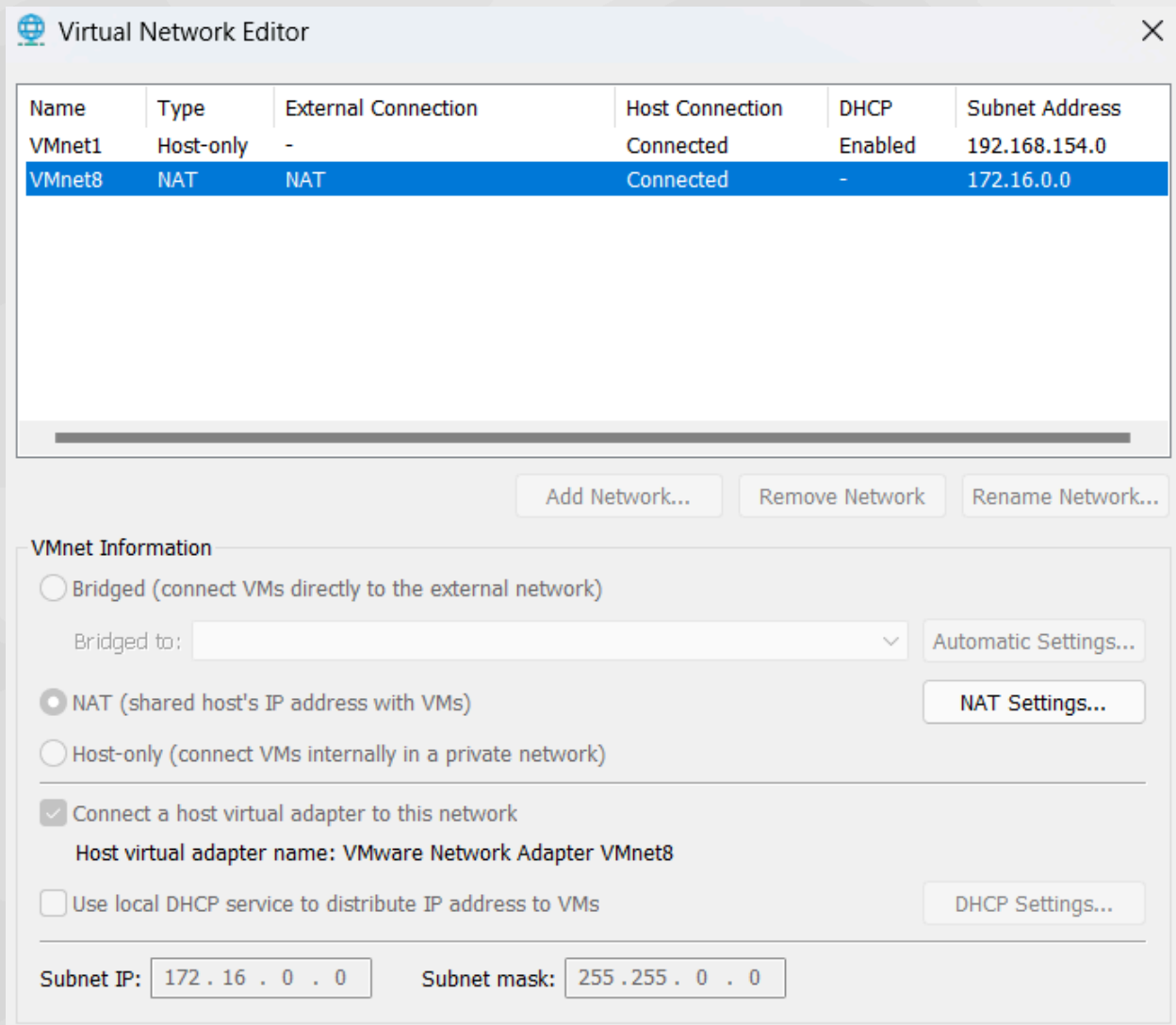
Pour la seconde VM ayant le serveur FTP, toujours dans le network adapter on sélectionne " Custom specific virtual network " avec le VMnet8 qui dans notre cas correspond à la VM ayant le serveur DHCP





Puis pour finir on ouvre le virtual network edition dans l'onglet edit et on applique les changements suivants :

- adresse de notre réseau de classe b
- on sélectionne NAT



The screenshot shows the 'Virtual Network Editor' window. At the top, there is a table listing virtual networks. The second row, 'VMnet8', is selected and highlighted in blue. Below the table are three buttons: 'Add Network...', 'Remove Network', and 'Rename Network...'. Under the 'VMnet Information' section, the 'NAT (shared host's IP address with VMs)' radio button is selected. The 'Bridged to:' dropdown is empty, and the 'Automatic Settings...' button is visible. The 'Host-only (connect VMs internally in a private network)' option is unselected. A checked checkbox 'Connect a host virtual adapter to this network' is present, with the 'Host virtual adapter name' set to 'VMware Network Adapter VMnet8'. The 'Use local DHCP service to distribute IP address to VMs' checkbox is unselected, and the 'DHCP Settings...' button is visible. At the bottom, the 'Subnet IP' is set to '172.16.0.0' and the 'Subnet mask' is set to '255.255.0.0'.

Name	Type	External Connection	Host Connection	DHCP	Subnet Address
VMnet1	Host-only	-	Connected	Enabled	192.168.154.0
VMnet8	NAT	NAT	Connected	-	172.16.0.0

Buttons: Add Network..., Remove Network, Rename Network...

**VMnet Information**

☐ Bridged (connect VMs directly to the external network)  
Bridged to:  Automatic Settings...

☒ NAT (shared host's IP address with VMs) NAT Settings...

☐ Host-only (connect VMs internally in a private network)

☒ Connect a host virtual adapter to this network  
Host virtual adapter name: VMware Network Adapter VMnet8

☐ Use local DHCP service to distribute IP address to VMs DHCP Settings...

Subnet IP:  Subnet mask:

Félicitation notre seconde machine virtuel est sur le réseau DHCP

# INSTALLATION ET CONFIGURATION DU SERVEUR FTP ET SSH

Pour installer un serveur ProFTPD et SSH, nous utilisons les commandes suivantes :

**apt install proftpd apt install openssh-server**

Ces commandes permettent d'installer les paquets nécessaires pour le fonctionnement des serveurs ProFTPD et SSH sur notre machine.

```
root@debian:/etc/proftpd# apt install proftpd
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances... Fait
Lecture des informations d'état... Fait
```

Pour configurer le serveur FTP avec une seule session de connexion possible, nous devons modifier le fichier de configuration de ProFTPD. Pour ce faire, nous ouvrons le fichier de configuration en utilisant la commande suivante :

**nano /etc/proftpd/proftpd.conf**

Dans ce fichier, nous devons rechercher la ligne suivante :

**MaxClients 10**

Nous devons remplacer "10" par "1" pour limiter le nombre de connexions simultanées à une seule session. Cela signifie qu'une fois qu'un utilisateur est connecté au serveur FTP, toute tentative de connexion supplémentaire sera refusée jusqu'à ce que la première session se termine.

```
#  # Limit the maximum number of anonymous logins
MaxClients 1
#
```

On nous demande ensuite de créer un utilisateur, nous utilisons la commande **adduser** suivie du nom d'utilisateur souhaité.

Dans notre cas, nous souhaitons créer un utilisateur nommé "laplateforme".

```
root@debian:/etc/proftpd# adduser laplateforme
Ajout de l'utilisateur « laplateforme » ...
Ajout du nouveau groupe « laplateforme » (1001) ...
Ajout du nouvel utilisateur « laplateforme » (1001) avec le groupe « laplateforme » (1001) ...
Création du répertoire personnel « /home/laplateforme » ...
Copie des fichiers depuis « /etc/skel » ...
Nouveau mot de passe :
Retapez le nouveau mot de passe :
passwd : mot de passe mis à jour avec succès
```

Cette commande nous demandera de saisir un mot de passe pour l'utilisateur, puis de confirmer le mot de passe.

```
AuthUserFile /etc/proftpd/ftpd.passwd
AuthPam off_
Group sftpusers

<Limit LOGIN>
    AllowUser laplateforme
    DenyAll
</Limit>
```

Pour configurer le serveur FTP avec une authentification spécifique, nous devons ajouter certaines lignes à la fin du fichier de configuration de ProFTPD.

Tout d'abord, nous ajoutons la ligne suivante pour indiquer le chemin vers le fichier contenant les informations d'authentification des utilisateurs :

**AuthUserFile /etc/proftpd/ftpd.passwd**

Ensuite, nous désactivons l'authentification PAM en ajoutant la ligne suivante :

**AuthPAM off**

Cela permet de centraliser la configuration d'authentification dans le fichier proftpd.conf plutôt que de répartir la configuration entre plusieurs fichiers de configuration PAM. De plus, désactiver PAM peut réduire la surface d'attaque potentielle en réduisant le nombre de points d'entrée possibles pour les attaquants.

Enfin, nous ajoutons les lignes suivantes pour limiter les connexions au serveur FTP à un seul utilisateur :

**<Limit LOGIN> AllowUser laplateforme DenyAll </Limit>**

Cette configuration permet uniquement à l'utilisateur "laplateforme" de se connecter au serveur FTP. Toutes les autres tentatives de connexion seront refusées.

Pour créer le fichier de mots de passe FTP et y ajouter les informations d'identification de l'utilisateur, nous utilisons la commande suivante :

```
sh -c 'echo "laplateforme:${openssl passwd -1 Marseille13!}" >>  
/etc/proftpd/ftpd.passwd'
```

Cette commande crée un mot de passe crypté pour l'utilisateur "laplateforme" avec le mot de passe "Marseille13!" en utilisant la commande "openssl passwd -1". Le mot de passe crypté est ensuite ajouté au fichier "/etc/proftpd/ftpd.passwd" en utilisant la commande "echo". Le symbole ">>" est utilisé pour ajouter le mot de passe au fichier plutôt que de remplacer son contenu existant.

```
laplateforme:$1$j9YvI0xf$eCBcMRAHWyEyFMW37qqfc1
```

Ensuite, j'ai créé un groupe en utilisant la commande "groupadd" suivie du nom du groupe. J'ai ajouté mon utilisateur à ce groupe en utilisant la commande "usermod -aG" suivie du nom du groupe et du nom de l'utilisateur.

Pour donner les droits au groupe créé précédemment, j'ai utilisé les commandes suivantes pour les fichiers ftpd.passwd et proftpd.conf :

```
chown : "nom_groupe" chemin_repertoire
```

Cette commande permet de changer le propriétaire du fichier en le groupe spécifié. Ensuite, pour donner les accès appropriés, j'ai utilisé la commande chmod suivie des permissions souhaitées et du chemin du répertoire. Dans notre cas, j'ai utilisé la commande suivante :

```
chmod 640 chemin_repertoire
```

Le 6 signifie que le créateur du fichier peut le lire et écrire, le groupe utilisateur peut uniquement lire le fichier et 0 que les autres n'ont aucune permission.

```
root@debian:~# chown :sftpusers /etc/proftpd/ftpd.passwd  
root@debian:~# chmod 640 /etc/proftpd/ftpd.passwd
```

On peut ensuite s'assurer que les droits ont bien été mis avec la commande suivante:

```
root@debian:~# ls -l /etc/proftpd/ftpd.passwd  
-rw-r----- 1 root sftpusers 48 25 mars 12:24 /etc/proftpd/ftpd.passwd
```



# INSTALLATION ET CONFIGURATION DU SERVEUR DNS

Pour mettre en place un serveur DNS, nous allons utiliser Bind9. Nous commençons par installer Bind9 en utilisant la commande

**"apt install bind9".**

Ensuite, nous allons créer un fichier de zone pour notre domaine. Dans notre cas, nous allons créer un fichier pour "ftp.com". Nous utilisons la commande

**"nano /etc/bind/db.ftp.com"**

```
$TTL 604800
@      IN      SOA      ftp.com. admin.ftp.com. (
                        1          ; Serial
                        604800     ; Refresh
                        86400      ; Retry
                        241920     ; Expire
                        604800 )   ; Negative Cache TTL
;
@      IN      NS       ns
@      IN      A        172.16.0.3
ns     IN      A        172.16.0.3
dns    IN      A        172.16.222.41
```

Dans ce fichier, nous allons définir les enregistrements DNS pour notre domaine.

Nous allons créer un enregistrement A pour notre serveur FTP, qui pointe vers l'adresse IP 172.16.222.41. Nous allons également créer un enregistrement NS pour notre domaine, qui pointe vers notre serveur DNS.

Ensuite, nous allons configurer Bind9 pour utiliser notre fichier de zone. Pour ce faire, nous allons modifier le fichier de configuration de Bind9 en utilisant la commande

**"nano /etc/bind/named.conf.local".**

Dans ce fichier, nous allons ajouter une section pour notre domaine, en spécifiant le chemin vers notre fichier de zone et en autorisant le transfert de zone vers notre adresse IP 172.16.222.41.

```
Zone "ftp.com" {
    type master;
    file "/etc/bind/db.ftp.com";
    allow-transfer {172.16.222.41; };
};
```

Enfin, nous allons redémarrer Bind9 en utilisant la commande "systemctl restart bind9" pour prendre en compte les modifications.

Nous allons également modifier le fichier de configuration du résolveur en utilisant la commande "**nano /etc/resolv.conf**" pour ajouter notre serveur DNS en tant que premier serveur de noms.

```
domain ftp.com  
search ftp.com  
nameserver 172.16.0.3
```

En configurant notre serveur DNS de cette manière, nous avons maintenant la possibilité de résoudre les noms de domaine de notre choix, y compris celui de notre serveur FTP

**"db.ftp.com".**

Cela signifie que lorsqu'un utilisateur tentera de se connecter à notre serveur FTP en utilisant l'adresse "db.ftp.com", notre système interrogera notre serveur DNS pour obtenir l'adresse IP correspondante, à savoir 172.16.222.41.

## Test de Connexion au Serveur SFTP

Pour se connecter au serveur depuis la première machine, nous devons tout d'abord récupérer l'adresse IP de la machine virtuelle hébergeant le serveur Proftpd. Pour ce faire, nous nous connectons à cette machine virtuelle et exécutons la commande

**"ip a"**

Dans notre cas, l'adresse IP de la machine virtuelle est "172.16.222.41". Nous pouvons maintenant utiliser cette adresse IP pour nous connecter au serveur Proftpd depuis la première machine.

```
valid_lft forever preferred_lft forever
2: ens160: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state
    link/ether 00:0c:29:00:af:8d brd ff:ff:ff:ff:ff:ff
    altname enp3s0
    inet 172.16.222.41/16 brd 172.16.255.255 scope global dynamic ens160
        valid_lft 158sec preferred_lft 158sec
    inet6 fe80::20c:29ff:fe00:af8d/64 scope link
        valid_lft forever preferred_lft forever
```

Une fois que nous avons récupéré l'adresse IP de la machine hébergeant le serveur Proftpd, nous pouvons retourner sur la première machine pour nous connecter au serveur via SFTP. Pour ce faire, nous utilisons la commande suivante :

**sftp "user"@"hôte"**

où "user" correspond à l'utilisateur avec lequel nous souhaitons nous connecter au serveur, et "hôte" correspond à l'adresse IP de la machine hébergeant le serveur Proftpd.

Après avoir exécuté cette commande, nous sommes invités à entrer le mot de passe de l'utilisateur spécifié

```
root@debian:/etc/bind# sftp -P6500 laplateforme@dns.ftp.com
laplateforme@dns.ftp.com's password:
Connected to dns.ftp.com.
sftp> _
```

## Paramètres de Sécurité Additionnels

En utilisant la commande **nano /etc/ssh/sshd\_config**, nous pouvons accéder au fichier de configuration SSH. Ce fichier nous permet de contrôler divers paramètres, tels que le filtrage d'adresses IP ou l'accès via un certain port.

Dans notre cas, nous souhaitons modifier le port par défaut utilisé pour les connexions SSH. En effet, le port par défaut est souvent la cible d'attaques par force brute, il est donc recommandé de le changer pour un port moins commun. Nous avons choisi le port 6500.

```
# This is the sshd server system-wide configuration file. See
# sshd_config(5) for more information.

# This sshd was compiled with PATH=/usr/local/bin:/usr/bin:/bin:/usr/

# The strategy used for options in the default sshd_config shipped with
# OpenSSH is to specify options with their default value where
# possible, but leave them commented. Uncommented options override the
# default value.

Include /etc/ssh/sshd_config.d/*.conf

Port 6500
#AddressFamily any
#ListenAddress 0.0.0.0
#ListenAddress ::

#HostKey /etc/ssh/ssh_host_rsa_key
#HostKey /etc/ssh/ssh_host_ecdsa_key
#HostKey /etc/ssh/ssh_host_ed25519_key

# Ciphers and keying
#RekeyLimit default none

# Logging
#SyslogFacility AUTH
#LogLevel INFO

# Authentication:
AllowUsers laplateforme
UsePam no
PermitRootLogin no
#LoginGraceTime 2m
#PermitRootLogin prohibit-password
#StrictModes yes
#MaxAuthTries 6
#MaxSessions 10

#PubkeyAuthentication yes

# Expect .ssh/authorized_keys2 to be disregarded by default in future
#AuthorizedKeysFile .ssh/authorized_keys .ssh/authorized_keys2
```



Pour limiter l'accès au serveur SSH uniquement à l'utilisateur "laplateforme", nous pouvons ajouter la ligne suivante au fichier de configuration SSH :

**"AllowUsers laplateforme"**. Cela empêchera tout autre utilisateur de se connecter au serveur via SSH.

En outre, nous pouvons également désactiver l'utilisation de PAM en ajoutant la ligne **"UsePam no"** au fichier de configuration SSH. Cela peut aider à renforcer la sécurité du serveur en empêchant l'utilisation de modules d'authentification externes.

Enfin, pour empêcher la connexion directe en tant que root, nous pouvons ajouter la ligne **"PermitRootLogin no"** au fichier de configuration SSH. Cela peut aider à réduire les risques liés à l'accès non autorisé au serveur. Il est recommandé de se connecter en tant qu'utilisateur régulier et d'utiliser la commande "sudo" pour exécuter des tâches nécessitant des privilèges d'administrateur.

```
AllowUsers laplateforme
UsePam no
PermitRootLogin no
```

Pour redémarrer le service SSH, nous pouvons utiliser la commande suivante :

**systemctl restart ssh.service**

Cela redémarrera le service SSH et prendra en compte les modifications apportées au fichier de configuration.

Pour se connecter en SFTP, nous pouvons utiliser la commande suivante :

**sftp -P 6500 laplateforme@"hôte"**

Cette commande nous permet de nous connecter au service FTP de manière sécurisée via le port 6500. Il est important de ne pas oublier l'option "-P" pour spécifier le numéro de port que nous avons choisi pour la connexion SFTP. Dans cet exemple, nous nous connectons en tant qu'utilisateur "laplateforme" à l'hôte spécifié.

```
root@debian:/etc/bind# sftp -P6500 laplateforme@dns.ftp.com
laplateforme@dns.ftp.com's password:
Connected to dns.ftp.com.
sftp> _
```