

Documentation – Serveur DHCP/DNS



Classe : BTS SIO 25.1A

Nom : Rayan Bellahouel

Table des matières

1- Définition

2- Prérequis

3- Installation

4- Fonctionnement

5- Fonctionnalités

1- Définition :

Définition du TP : Gérer les adresses IP des clients avec un serveur DHCP et DNS en les limitant entre 10.0.2.100 et 10.0.2.200 et les connecter au nom de domaine de l'entreprise.

DHCP : Le DHCP ou Dynamic Host Configuration Protocol est un protocole de communication ou protocole réseau parfaitement indispensable. On le traduit généralement par protocole de configuration dynamique des adresses IP.

DNS : Un système de noms de domaine, ou DNS, traduit les noms de domaine lisibles par l'homme (par exemple www.amazon.com) en adresses IP lisibles par une machine (par exemple, 192.0.2.44).

Debian : Debian est une distribution GNU/Linux non commerciale, elle a pour principal but de fournir un système d'exploitation composé uniquement de logiciels libres.

Netmask : Un netmask joue un rôle crucial dans les communications réseau puisqu'il détermine la façon dont les adresses IP sont divisées en réseaux et en hôtes. En appliquant le masque, vous pouvez identifier les appareils se trouvant sur le même réseau et communiquer directement entre eux sans avoir besoin d'un routage.

Gateway : Un gateway : La Gateway est le dispositif par lequel deux réseaux informatiques ou deux réseaux de télécommunication de nature différente sont reliés. Le dispositif permet de vérifier la sécurité du réseau qui cherche à se connecter à l'autre. La Gateway est aussi appelée passerelle applicative.

NAT : La NAT (Network Address Translation) permet à des serveurs, des hôtes et des consoles se trouvant sur différents réseaux de communiquer entre eux via un réseau interne commun. Une solution NAT met en correspondance un domaine d'adresses locales privées avec un domaine d'adresses publiques.

IP : Internet Protocol (IP) est un protocole, ou un ensemble de règles, appliqué au routage et à l'adressage des paquets de données afin qu'ils puissent traverser les réseaux et arriver à la bonne destination. Les données traversant Internet sont divisées en morceaux plus petits, appelés paquets. Des informations IP sont attachées à chaque paquet, et ces informations permettent aux routeurs d'envoyer des paquets au bon endroit. Chaque appareil ou domaine qui se connecte à Internet se voit attribuer une adresse IP.

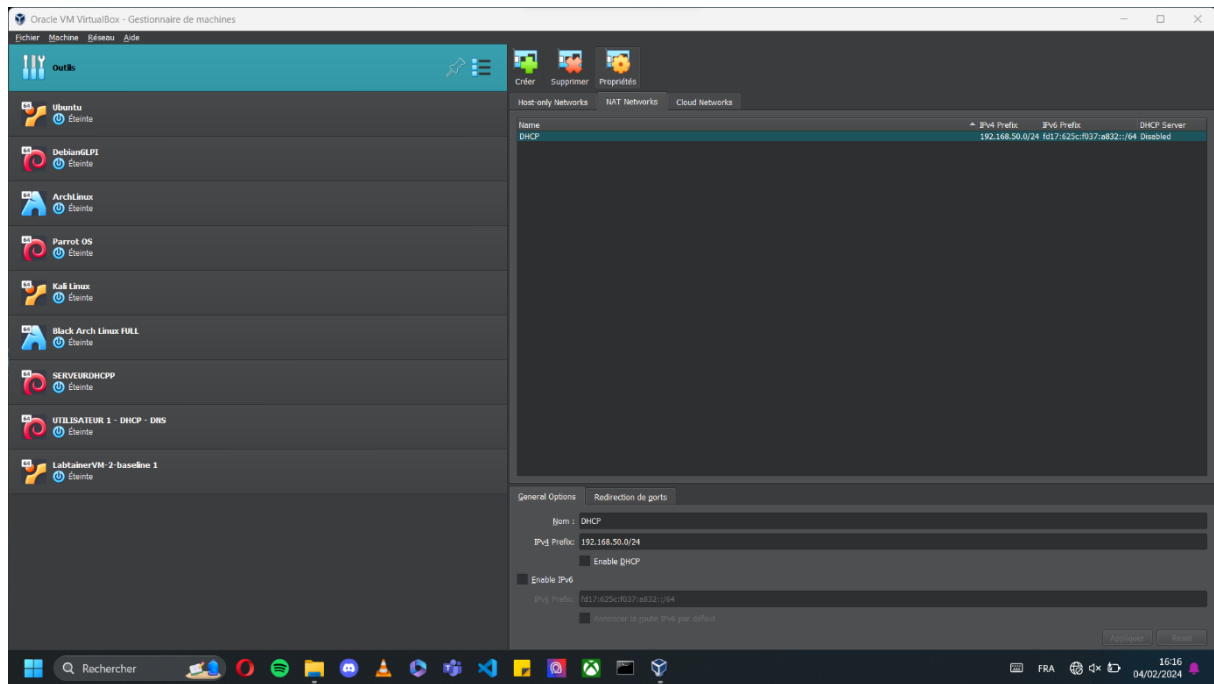
2- Prérequis :

- Avoir un PC, avec un logiciel de virtualisation hyperviseur de type 2. (Ici VirtualBox est utilisé)
- Deux machine virtuelle (un hôte et une cliente)
- Avoir un ISO d'une distribution Linux. (Ici Debian 12)
- Créer une machine virtuelle en utilisant l'ISO Debian ici une interface non graphique donc without GUI. Avec 4096mo en ram, 2 cœurs et 20 go de stockage.
- Connaissance en administration basique d'un serveur Linux.

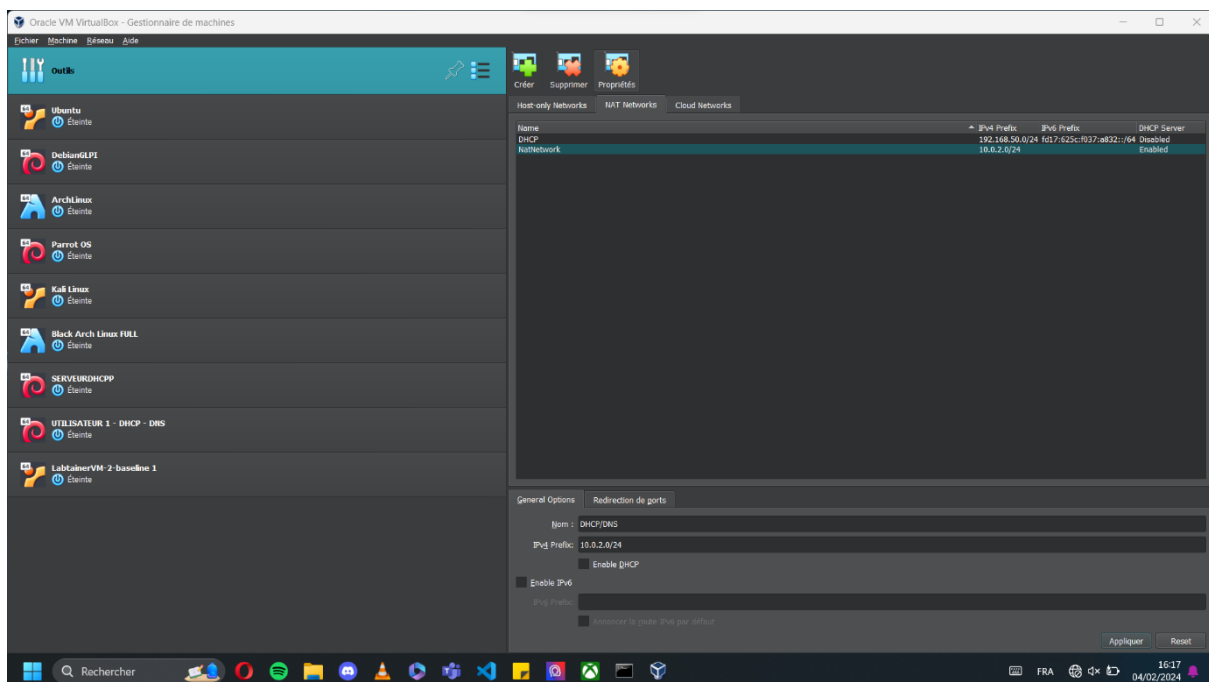
3- Installation :

Avant de commencer le TP

Une fois sur VirtualBox, cliquer sur l'onglet outils -> Nat Networks et cliquer sur créer

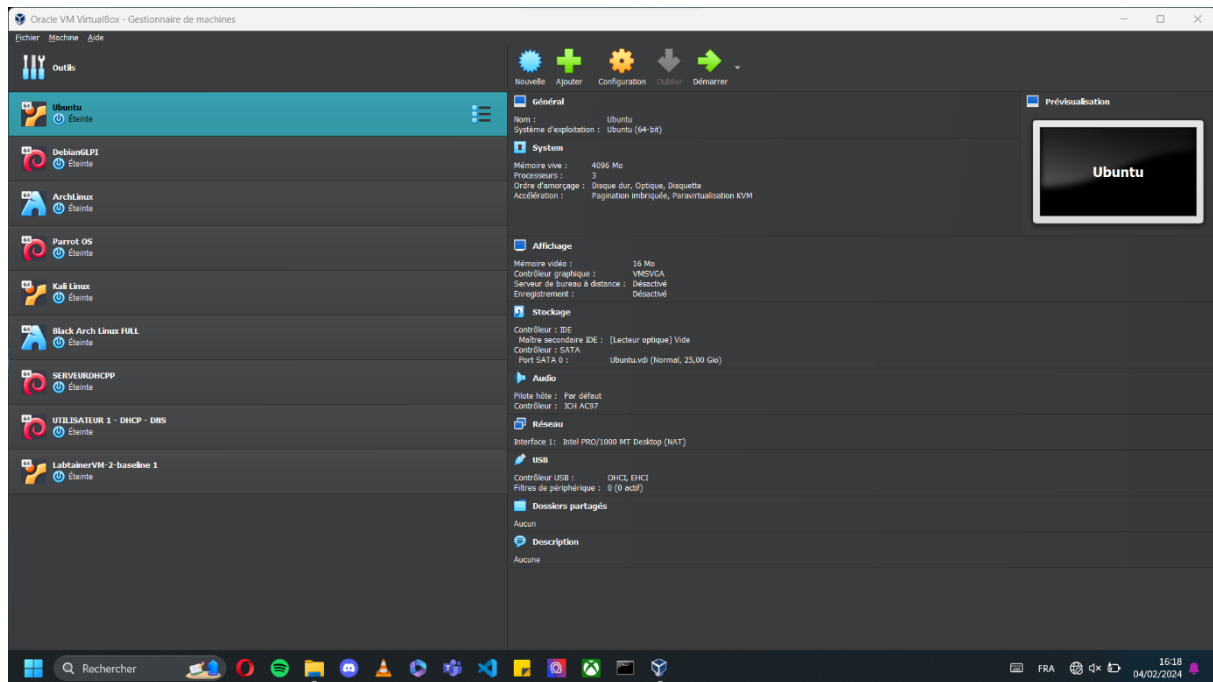


Vous pouvez modifier le nom du NAT Par exemple « DHCP/DNS » et pour l'IPv4 attribuer lui une IP qui commence par 10, 192 ou 172 Ex : 10.0.2.0. Décoché Enable DHCP

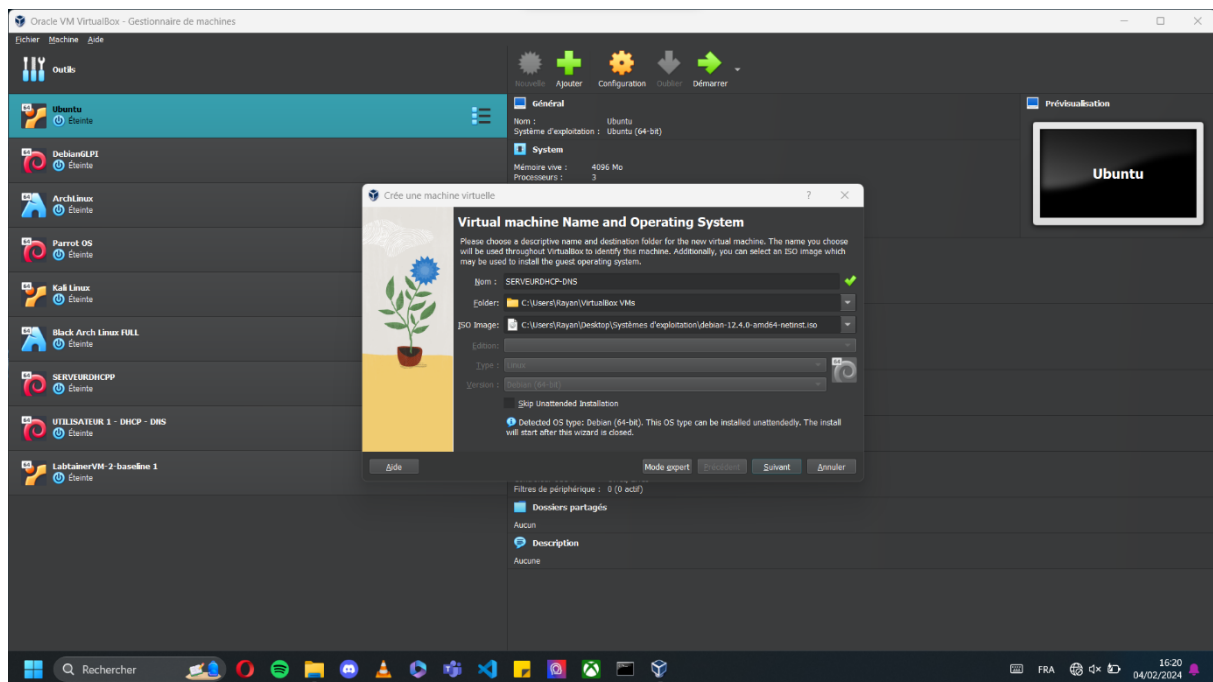


Passons maintenant à l'installation de l'iso debian 12

(Disponible au téléchargement ici : <https://www.debian.org/download>)

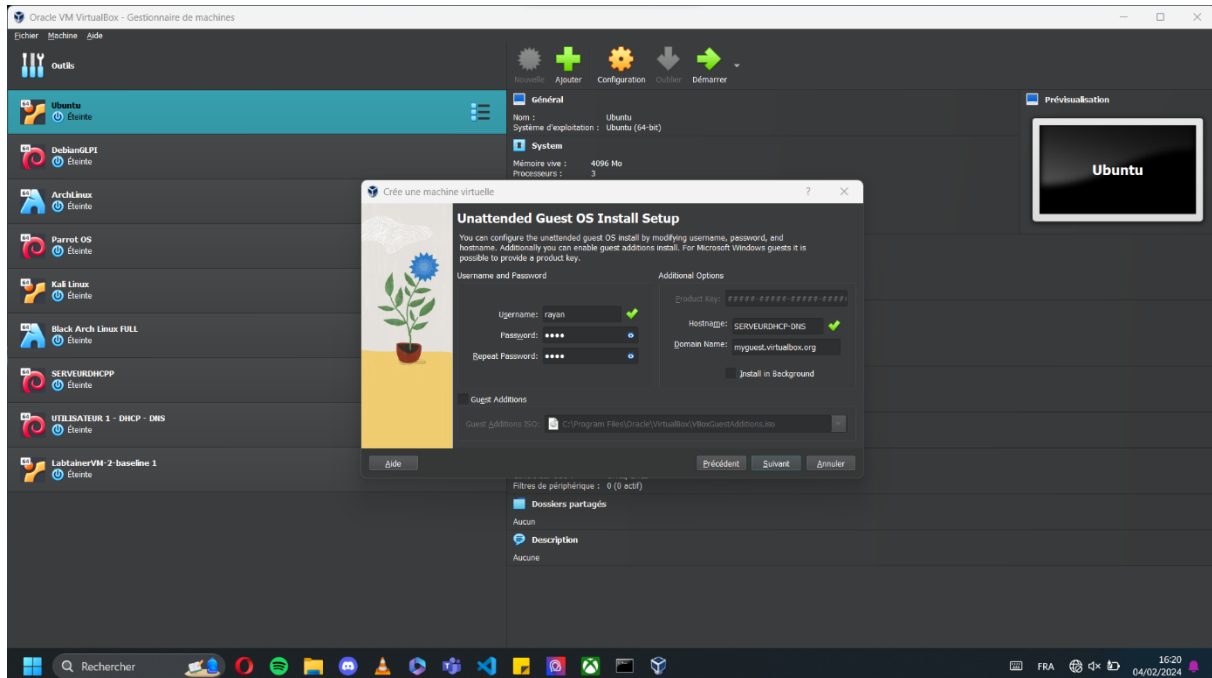


Cliquer sur Nouvelle, configurer un nom pour votre machine, et sélectionner le répertoire de l'iso, pour finir cliquer sur suivant.

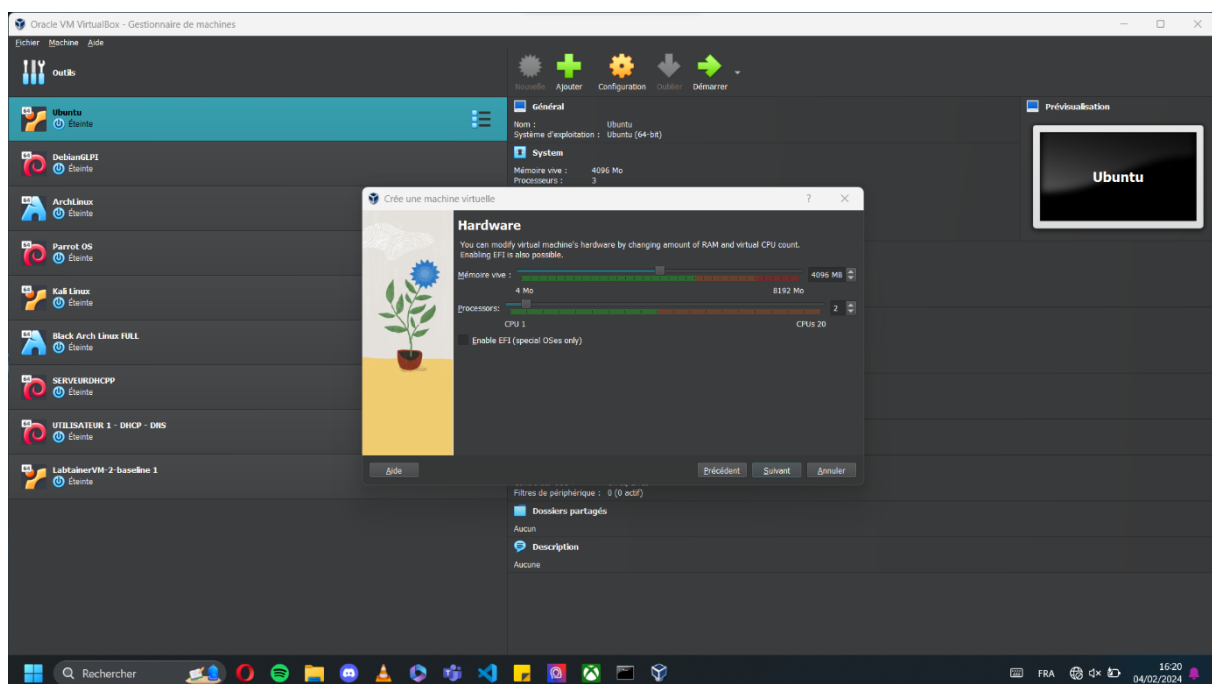


Configurer un UserName et un mot de passe pour votre machine et cliquer sur suivant.

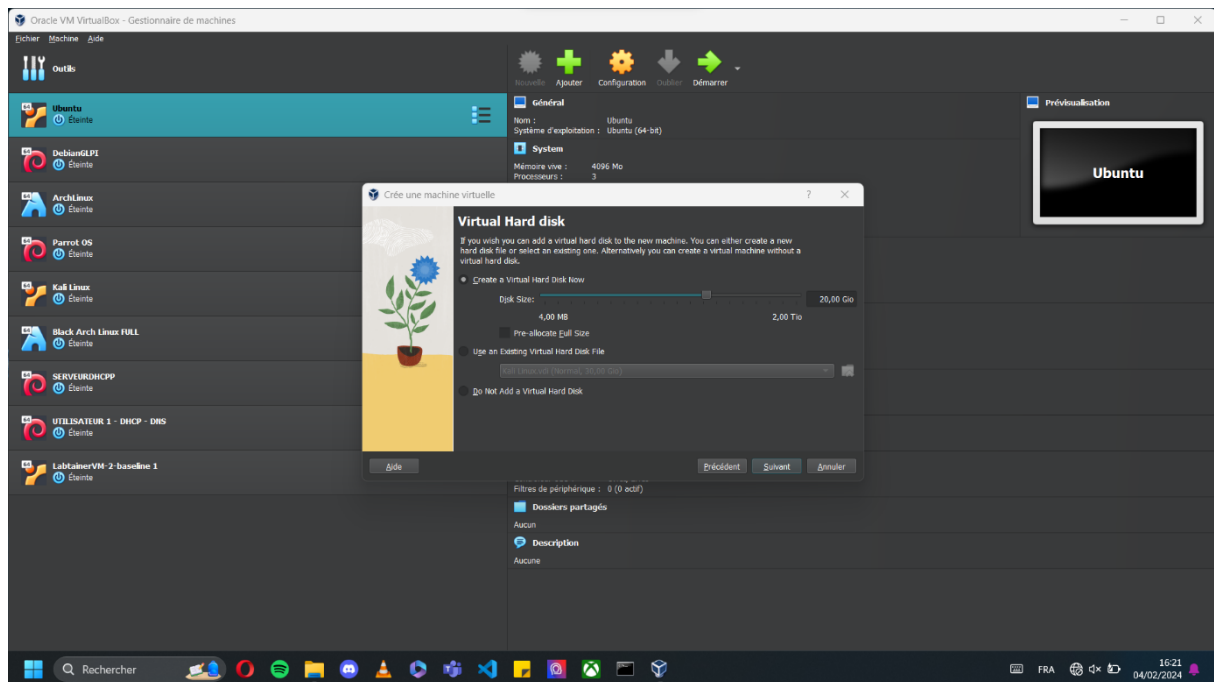
ATTENTION : n'oubliez pas votre mot de passe.



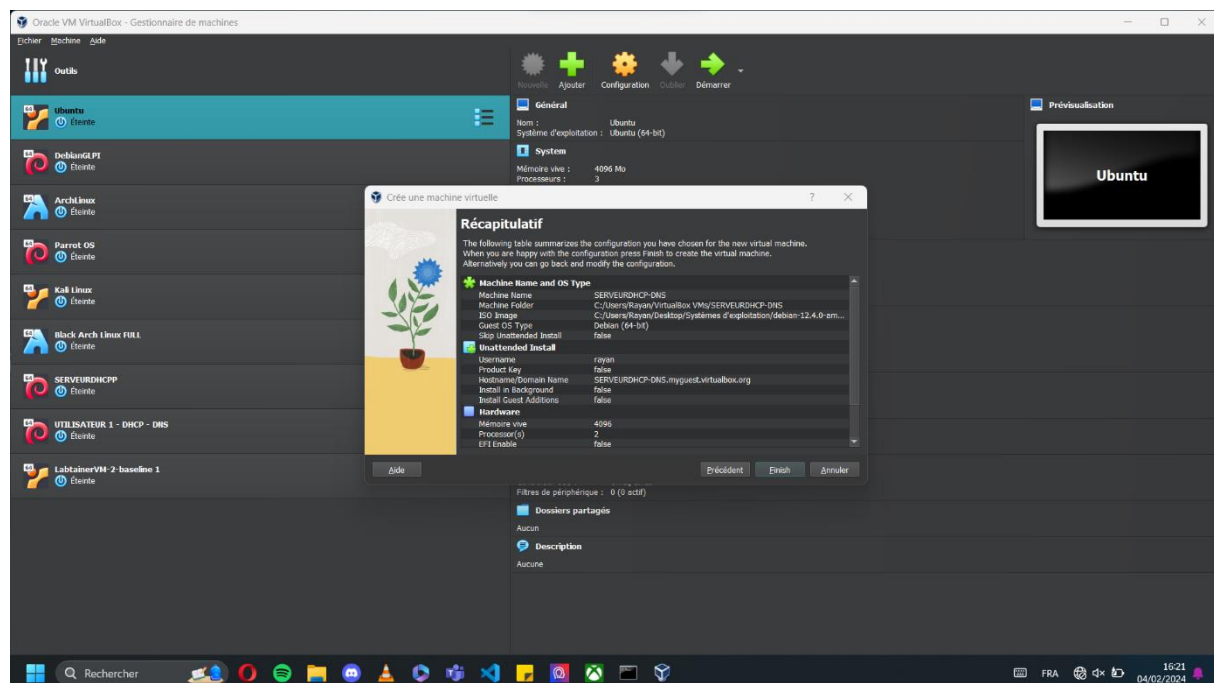
Passons à la configuration matérielle de la machine virtuelle, pour la mémoire vive mettez 4096 MB et pour le processeur 2. Cliquer sur suivant.



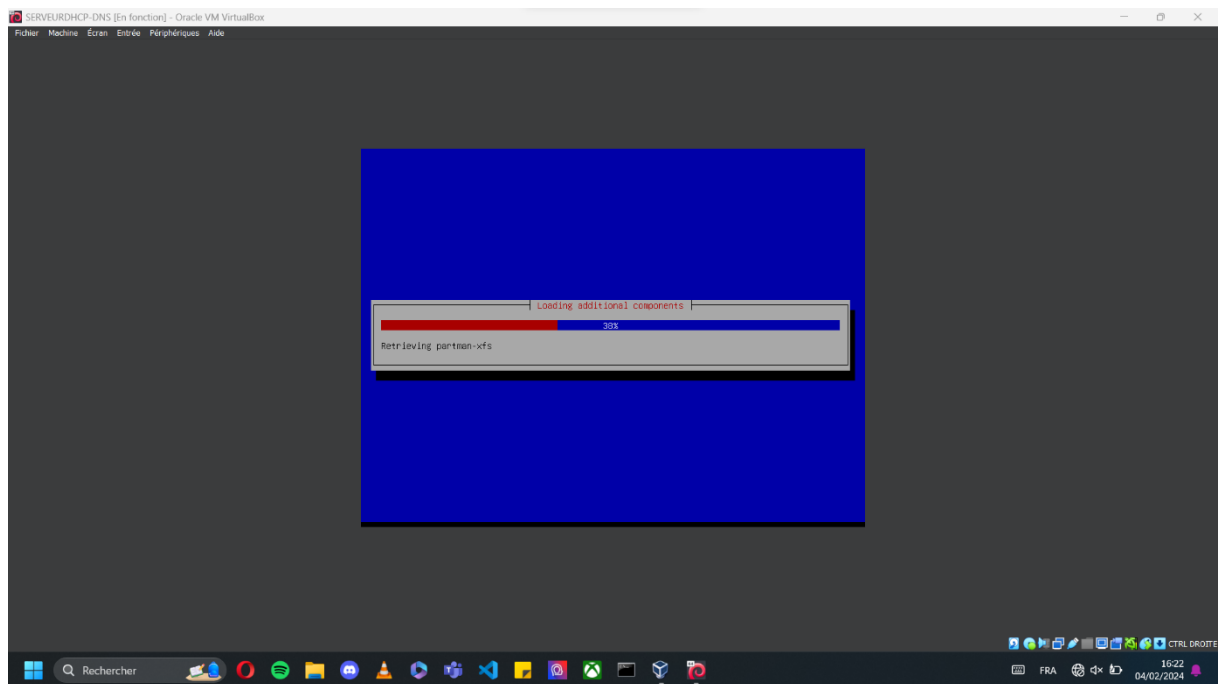
Pour le stockage de la machine virtuel, laisser les configurations de base et cliquer sur suivant.



Voici le récapitulatif de votre machine, si tout est bon, appuyer sur finish.

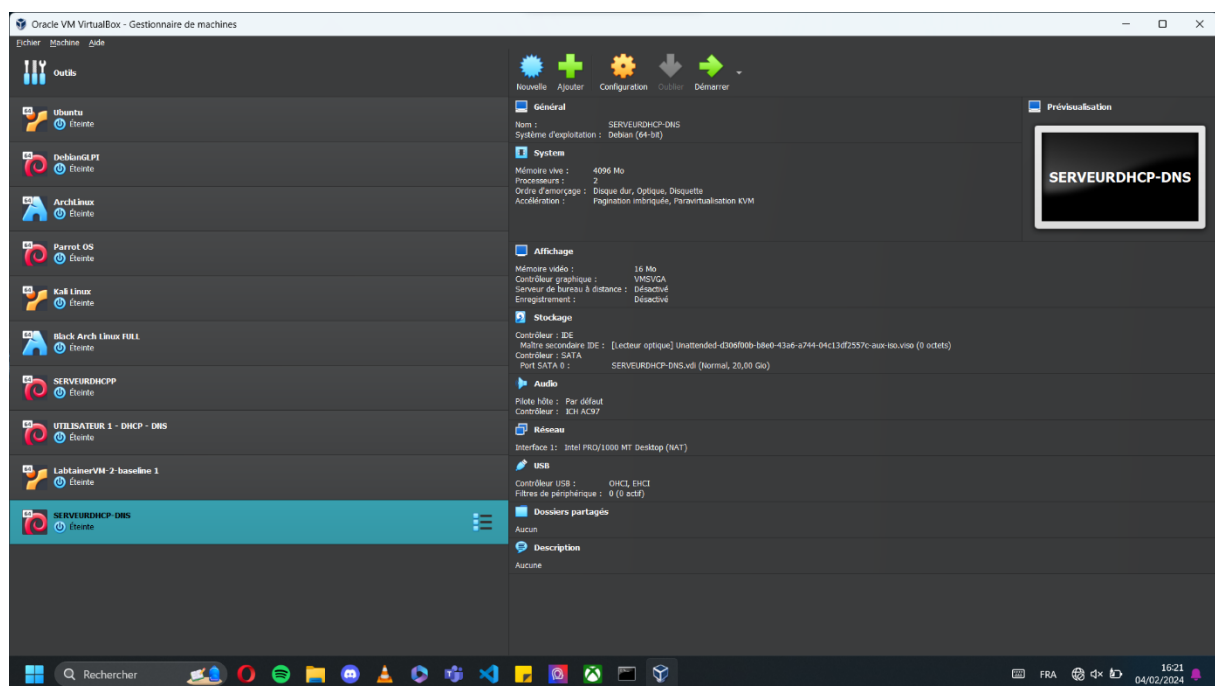


Vous n'avez plus qu'à attendre la fin de l'installation.

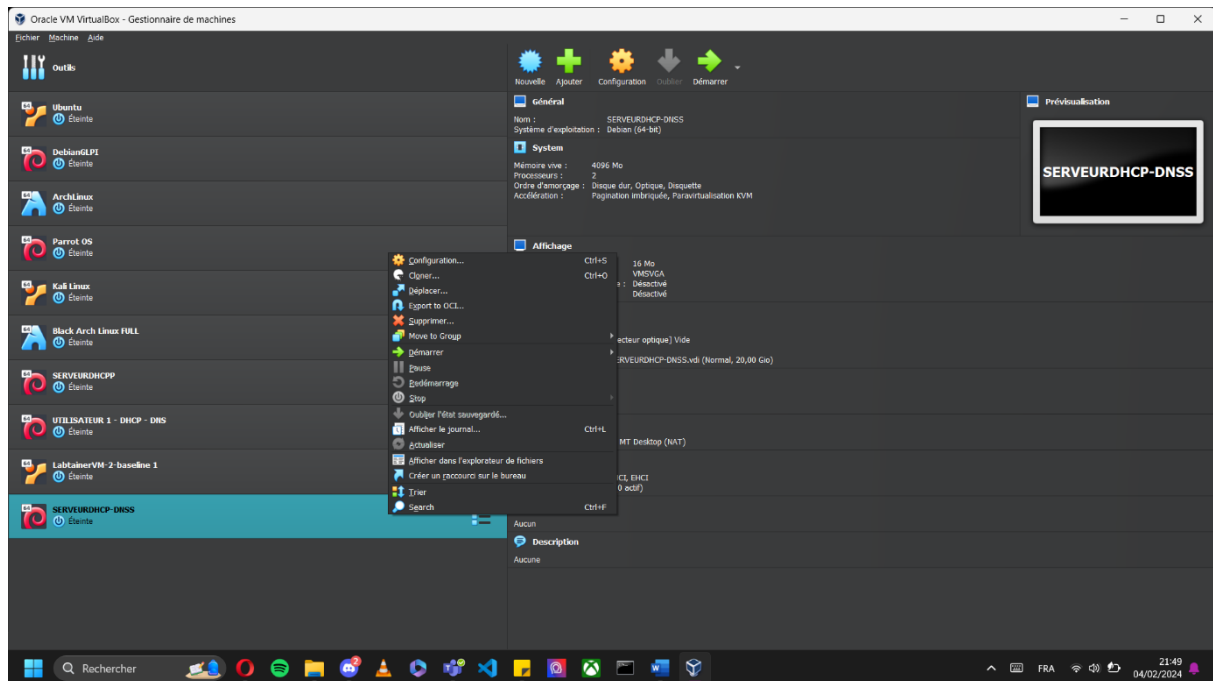


N'oubliez pas de lier le serveur DHCP a la machine virtuelle

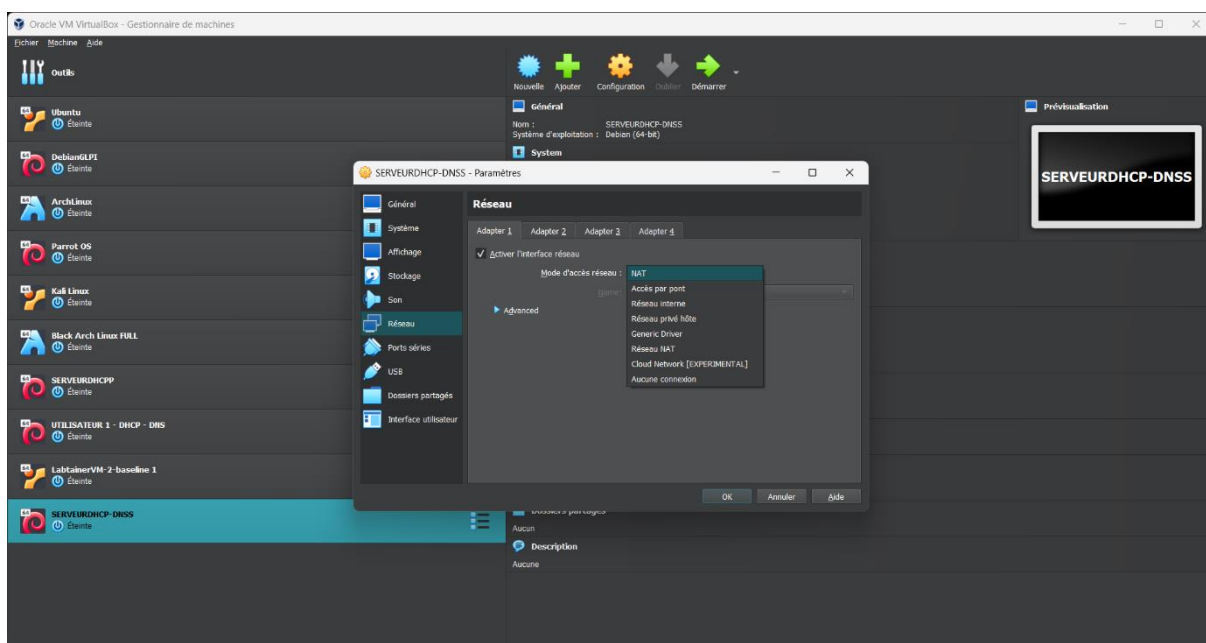
Note : N'oublier pas de faire la même chose pour la deuxième machine virtuelle.



Faites un clic droit, sélectionner configuration



Cliquer sur réseau, pour le mode d'accès réseau sélectionner "réseau NAT" et dans Name" le réseau que vous avez créé en l'occurrence DHCP/DNS si vous avez mis le même nom.



Vous pouvez commencer le TP

4- Fonctionnement :

Partie 1 :

1- Mettre une adresse IP statique :

Pour pouvoir modifier notre adresse ip il nous faut d'abord savoir quel est le nom du réseau avec la commande :

IP A

On obtient alors ceci :

```
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 noprefixroute
        valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:5e:a5:c3 brd ff:ff:ff:ff:ff:ff
    inet 10.0.2.10/24 brd 10.0.2.255 scope global enp0s3
        valid_lft forever preferred_lft forever
    inet6 fe80::a00:27ff:fe5e:a5c3/64 scope link
        valid_lft forever preferred_lft forever
```

Pour pouvoir changer l'adresse IP en statique nous devons passer par la commande nous renvoyant l'interface de connexion :

nano /etc/network/interfaces

Ce qui nous envoie au document que l'on modifie tel que l'on obtienne :

```
# 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

auto enp0s3
iface enp0s3 inet static
address 10.0.2.10/24
gateway 10.0.2.2
netmask 255.255.255.0
```

Après cette manipulation nous pouvons alors redémarrer le tout avec :

systemctl restart networking

Si vous n'avez pas de message d'erreur, alors, tout est bon :

```
rayan@SERVEURDHCPP:~$ sudo systemctl restart networking
[sudo] password for rayan:
rayan@SERVEURDHCPP:~$ _
```

Commande pour vérifier si tout fonctionne bien vous pouvez utiliser cette commande :

Systemctl status networking

S'il n'y a pas de message d'erreur, alors, tout est bon :

```
rayan@SERVEURDHCPP:~$ sudo systemctl status networking
• networking.service - Raise network interfaces
  Loaded: loaded (/lib/systemd/system/networking.service; enabled; preset: enabled)
  Active: active (exited) since Sun 2024-02-04 22:35:13 CET; 3min 30s ago
  Docs: man:interfaces(5)
  Process: 954 ExecStart=/sbin/ifup -a --read-environment (code=exited, status=0/SUCCESS)
  Process: 1003 ExecStart=/bin/sh -c if [ -f /run/network/restart-hotplug ]; then /sbin/ifup -a ->
  Main PID: 1003 (code=exited, status=0/SUCCESS)
  Tasks: 0 (limit: 4623)
  Memory: 12.0K
  CPU: 117ms
  CGroup: /system.slice/networking.service

Feb 04 22:35:13 SERVEURDHCPP systemd[1]: Starting networking.service - Raise network interfaces...
Feb 04 22:35:13 SERVEURDHCPP systemd[1]: Finished networking.service - Raise network interfaces.
lines 1-14/14 (END)
```

2- Installation des paquets et adressage réseau.

Installation du paquet isc-dhcp-server :

apt install isc-dhcp-server

L'installation du paquet nous donne accès a deux fichiers importants,

Le premier : etc/dhcp/dhcpd.conf

Le deuxième : etc/default/isc-dhcp-server

Une fois le paquet **isc-dhcp-server** installer récupérer l'adresse MAC avec :

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 noprefixroute
        valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:5e:a5:c3 brd ff:ff:ff:ff:ff:ff
    inet 10.0.2.15/24 brd 10.0.2.255 scope global enp0s3
        valid_lft forever preferred_lft forever
    inet6 fe80::a00:27ff:fe5e:a5c3/64 scope link
        valid_lft forever preferred_lft forever
```

Note : n'oubliez pas : **Sudo apt update** et **Sudo apt upgrade** Après chaque installation de paquet

Il vous faut taper la commande suivante pour modifier le fichier :

nano /etc/dhcp/dhcpd.conf

Modifier le contenu du fichier pour empêcher notre serveur de se donner une adresse IP a lui-même et limiter l'adressage réseau entre l'adresse .100 et .200 :

```
GNU nano 7.2 /etc/dhcp/dhcpd.conf
# dhcpd.conf
#
# Sample configuration file for ISC dhcpd
#
# option definitions common to all supported networks...
option domain-name "example.org";
option domain-name-servers ns1.example.org, ns2.example.org;

subnet 10.0.2.0 netmask 255.255.255.0
{
    range 10.0.2.100 10.0.2.200;
    option domain-name-servers 1.1.1.1;
    option routers 10.0.2.2;
}
host block_host_1{
hardware ethernet 08:00:27:5e/a5:c3;
deny booting;
}

default-lease-time 600;
max-lease-time 7200;

# The ddns-updates-style parameter controls whether or not the server will
# attempt to do a DNS update when a lease is confirmed. We default to the
# behavior of the version 2 packages ('none', since DHCP v2 didn't
# have support for DDNS.)
ddns-update-style none;
```

Il y'a un autre fichier à modifier :

nano /etc/default/isc-dhcp-server

Modifier le contenu du fichier comme ci-dessous, c'est simplement un fichier de configuration, rien de plus :

```
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="enp0s3"
#INTERFACESv6=""

[ Read 18 lines ]
^G Help      ^O Write Out ^W Where Is  ^K Cut       ^T Execute   ^C Location  M-U Undo
^X Exit      ^R Read File ^\ Replace   ^U Paste     ^J Justify   ^_ Go To Line M-E Redo
```

Enfin il nous reste plus qu'à relancer le service avec :

systemctl restart isc-dhcp-server

S'il n'y a pas de message d'erreur, alors, tout est bon :

```
rayan@SERVEURDHCPP:~$ sudo systemctl restart isc-dhcp-server
[sudo] password for rayan:
rayan@SERVEURDHCPP:~$
```


Commande pour vérifier si le serveur DHCP fonctionne bien vous pouvez utiliser cette commande :

Systemctl status isc-dhcp-server

S'il n'y a pas de message d'erreur, alors, tout est bon :

```
mayan@SERVEURDHCPP:~$ sudo systemctl status isc-dhcp-server
• isc-dhcp-server.service - LSB: DHCP server
   Loaded: loaded (/etc/init.d/isc-dhcp-server; generated)
   Active: active (running) since Sun 2024-02-04 23:01:42 CET; 40s ago
     Docs: man:systemd-sysv-generator(8)
  Process: 1113 ExecStart=/etc/init.d/isc-dhcp-server start (code=exited, status=0/SUCCESS)
    Tasks: 1 (limit: 4623)
   Memory: 6.8M
      CPU: 37ms
   CGroup: /system.slice/isc-dhcp-server.service
           └─1126 /usr/sbin/dhcpd -4 -q -cf /etc/dhcp/dhcpd.conf enp0s3

Feb 04 23:01:40 SERVEURDHCPP systemd[1]: Starting isc-dhcp-server.service - LSB: DHCP server...
Feb 04 23:01:40 SERVEURDHCPP isc-dhcp-server[1113]: Launching IPv4 server only.
Feb 04 23:01:40 SERVEURDHCPP dhcpd[1126]: Wrote 0 deleted host decls to leases file.
Feb 04 23:01:40 SERVEURDHCPP dhcpd[1126]: Wrote 0 new dynamic host decls to leases file.
Feb 04 23:01:40 SERVEURDHCPP dhcpd[1126]: Wrote 1 leases to leases file.
Feb 04 23:01:40 SERVEURDHCPP dhcpd[1126]: Server starting service.
Feb 04 23:01:42 SERVEURDHCPP isc-dhcp-server[1113]: Starting ISC DHCPv4 server: dhcpd.
Feb 04 23:01:42 SERVEURDHCPP systemd[1]: Started isc-dhcp-server.service - LSB: DHCP server.
```

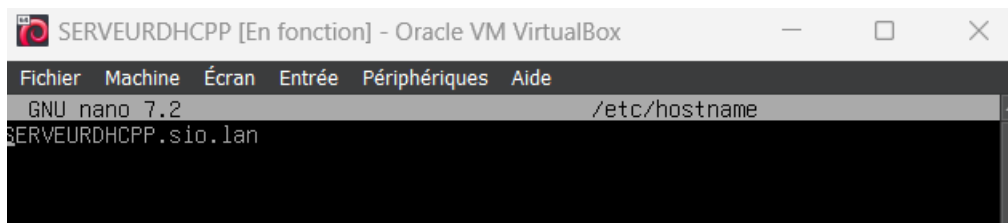
Partie 2 :

Pour commencer la partie 2 nous allons d'abord installer le paquet bind9 avec la commande suivante :

Apt install bind9

Maintenant nous devons modifier plusieurs dossiers. Nous commençons par modifier le hostname avec :

nano /etc/hostname

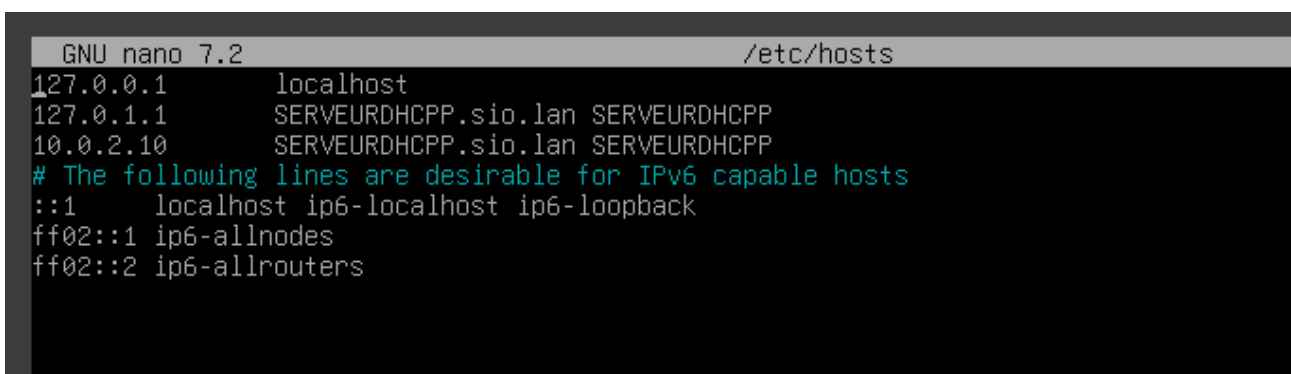


```
GNU nano 7.2 /etc/hostname
SERVEURDHCPP.sio.lan
```

Ici nous avons gardé en compte que celui-ci est le nom donné lors de la création et ajouter .sio.lan (Vous pouvez lui donné le nom que vous voulez)

Puis nous pouvons modifier le dossier hosts avec :

nano /etc/hosts



```
GNU nano 7.2 /etc/hosts
127.0.0.1    localhost
127.0.1.1    SERVEURDHCPP.sio.lan SERVEURDHCPP
10.0.2.10    SERVEURDHCPP.sio.lan SERVEURDHCPP
# The following lines are desirable for IPv6 capable hosts
::1         localhost ip6-localhost ip6-loopback
ff02::1     ip6-allnodes
ff02::2     ip6-allrouters
```

On marque à la ligne 2 et 3 de la colonne de droite le hostname.

Et on marque à la ligne 3 dans la colonne de gauche l'adresse IP de votre machine.

Maintenant retournons dans le fichier :

nano /etc/network/interfaces

Pour configurer le DNS.

```
GNU nano 7.2 /etc/network/interfaces
# 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

auto enp0s3
iface enp0s3 inet static
address 10.0.2.10/24
gateway 10.0.2.2
netmask 255.255.255.0
dns-domain sio.lan
dns-search sio.lan
dns-nameservers 10.0.2.10
```

Maintenant modifiez le fichier de configuration avec :

nano /etc/bind/named.conf.local

```
//  
// Do any local configuration here  
//  
  
// Consider adding the 1918 zones here, if they are not used in your  
// organization  
//include "/etc/bind/zones.rfc1918";  
  
zone "sio.lan" {  
    type master;  
    file "/etc/bind/forward.sio.lan";  
};  
  
zone "2.0.10.in-addr.arpa" {  
    type master;  
    file "/etc/bind/reverse.sio.lan";  
};
```

On crée deux zones une direct et un inverse.

Nous allons pouvoir créer le fichier pour la zone direct :

Mkdir /etc/bind/forward.sio.lan

Et

mkdir /etc/bind/reverse.sio.lan

Une fois les fichiers créés, nous pouvons y accéder :

nano /etc/bind/forward.sio.lan

```
GNU nano 7.2 /etc/bind/forward.sio.lan
$TTL      10800
$ORIGIN    sio.lan.
@          IN      SOA      SERVEURDHCPP.sio.lan. root.sio.lan. (
                                3; Serial
                                3h; Refresh
                                1h; Retry
                                1w; Expire
                                1h); Negative Cache TTL
;
@          IN      NS       SERVEURDHCPP.sio.lan.
SERVEURDHCPP IN      A       10.0.2.10
localhost  IN      A       127.0.0.1
USER1      IN      A       10.0.2.100
```

nano /etc/bind/reverse.sio.lan

```
GNU nano 7.2 /etc/bind/reverse.sio.lan
$TTL      10800
@          IN      SOA      SERVEURDHCPP.sio.lan. root.sio.lan. (
                                3; Serial
                                3h; Refresh
                                1h; Retry
                                1w; Expire
                                1h); Negative Cache TT;
;
@          IN      NS       SERVEURDHCPP.sio.lan.
10         IN      PTR      SERVEURDHCPP.sio.lan.
100        IN      PTR      USER1.sio.lan.
```

Modifier les fichiers comme indiqué ci-dessus.

Pour vérifier la syntaxe de tous les fichiers bind saisissez la commande suivante :

Named-checkconf - z

```
rayan@SERVEURDHCPP:~$ named-checkconf -z
zone sio.lan/IN: loaded serial 3
zone 2.0.10.in-addr.arpa/IN: loaded serial 3
zone localhost/IN: loaded serial 2
zone 127.in-addr.arpa/IN: loaded serial 1
zone 0.in-addr.arpa/IN: loaded serial 1
zone 255.in-addr.arpa/IN: loaded serial 1
rayan@SERVEURDHCPP:~$
```

Pour vérifier que tout est bon, saisissez la commande suivante :

Systemctl restart bind9

```
rayan@SERVEURDHCPP:~$ systemctl restart bind9
==== AUTHENTICATING FOR org.freedesktop.systemd1.manage-units ====
Authentication is required to restart 'named.service'.
Authenticating as: root
Password:
==== AUTHENTICATION COMPLETE ====
rayan@SERVEURDHCPP:~$
```

S'il n'y a pas de message d'erreur, alors, tout est bon :

Vérification du status du service bind9 :

Systemctl status bind9

```
rayan@SERVEURDHCPP:~$ systemctl status bind9
• named.service - BIND Domain Name Server
   Loaded: loaded (/lib/systemd/system/named.service; enabled; preset: enabled)
   Active: active (running) since Mon 2024-02-05 02:38:56 CET; 1min 51s ago
     Docs: man:named(8)
  Main PID: 1138 (named)
    Status: "running"
     Tasks: 6 (limit: 4623)
    Memory: 37.2M
       CPU: 116ms
    CGroup: /system.slice/named.service
            └─1138 /usr/sbin/named -f -u bind
rayan@SERVEURDHCPP:~$ _
```

S'il n'y a pas de message d'erreur, alors, tout est bon :

Vous avez terminé le TP, passons à la phase de test !

5- Fonctionnalités

Avec les commandes :

Systemctl restart

Et

Systemctl status

Nous avons vérifié que chaque service que nous avons configuré était théoriquement correct.

Pour le service networking :

L'objectif était d'attribuer une adresse IP à notre machine hôte.

Voici le résultat avec la commande :

IP A

```
1136 /usr/sbin/named -f -o bind
rayan@SERVEURDHCPP:~$ 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 noprefixroute
        valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:32:91:89 brd ff:ff:ff:ff:ff:ff
    inet 10.0.2.10/24 brd 10.0.2.255 scope global enp0s3
        valid_lft forever preferred_lft forever
    inet6 fe80::a00:27ff:fe32:9189/64 scope link
        valid_lft forever preferred_lft forever
rayan@SERVEURDHCPP:~$ _
```


Pour le service isc-dhcp-server :

L'objectif était d'attribuer une adresse IP à notre machine utilisateur via notre machine hôte

Voici le résultat avec la commande :

IP A (effectuer sur la machine utilisateur)

```
rayan@USER1:~$ 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 noprefixroute
        valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:eb:3b:b3 brd ff:ff:ff:ff:ff:ff
    inet 10.0.2.100/24 brd 10.0.2.255 scope global dynamic noprefixroute enp0s3
        valid_lft 464sec preferred_lft 464sec
    inet6 fe80::a00:27ff:feeb:3bb3/64 scope link noprefixroute
        valid_lft forever preferred_lft forever
rayan@USER1:~$
```

Pour le service bind9 :

L'objectif était d'attribuer un nom de domaine (DNS) à notre machine hôte et notre machine utilisateur

Voici le résultat avec la commande :

Ping USER1.sio.lan

```
rayan@SERVEURDHCPP:~$ ping USER1.sio.lan
PING USER1.sio.lan (10.0.2.100) 56(84) bytes of data:
64 bytes from USER1.sio.lan (10.0.2.100): icmp_seq=1 ttl=64 time=1.23 ms
64 bytes from USER1.sio.lan (10.0.2.100): icmp_seq=2 ttl=64 time=3.11 ms
64 bytes from USER1.sio.lan (10.0.2.100): icmp_seq=3 ttl=64 time=1.15 ms
64 bytes from USER1.sio.lan (10.0.2.100): icmp_seq=4 ttl=64 time=2.22 ms
64 bytes from USER1.sio.lan (10.0.2.100): icmp_seq=5 ttl=64 time=1.36 ms
64 bytes from USER1.sio.lan (10.0.2.100): icmp_seq=6 ttl=64 time=2.48 ms
64 bytes from USER1.sio.lan (10.0.2.100): icmp_seq=7 ttl=64 time=1.76 ms
64 bytes from USER1.sio.lan (10.0.2.100): icmp_seq=8 ttl=64 time=1.75 ms
64 bytes from USER1.sio.lan (10.0.2.100): icmp_seq=9 ttl=64 time=2.03 ms
64 bytes from USER1.sio.lan (10.0.2.100): icmp_seq=10 ttl=64 time=1.66 ms
64 bytes from USER1.sio.lan (10.0.2.100): icmp_seq=11 ttl=64 time=1.76 ms
64 bytes from USER1.sio.lan (10.0.2.100): icmp_seq=12 ttl=64 time=0.941 ms
```

dig USER1.sio.lan

```
rayan@SERVEURDHCPP:~$ dig USER1.sio.lan

; <<>> DiG 9.18.19-1~deb12u1-Debian <<>> USER1.sio.lan
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 36977
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:: udp: 1232
; COOKIE: 7176f3d61897217f0100000065c044e3af9089ef9c66cbcc (good)
;; QUESTION SECTION:
;USER1.sio.lan.                IN      A

;; ANSWER SECTION:
USER1.sio.lan.                10800    IN      A      10.0.2.100

;; Query time: 0 msec
;; SERVER: 10.0.2.10#53(10.0.2.10) (UDP)
;; WHEN: Mon Feb 05 03:16:03 CET 2024
;; MSG SIZE rcvd: 86
```

Vérification de la recherche inversée :

Dig -x 10.0.2.10

```
rayan@SERVEURDHCPP:~$ dig -x 10.0.2.10

; <<>> DiG 9.18.19-1~deb12u1-Debian <<>> -x 10.0.2.10
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 3023
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:: udp: 1232
; COOKIE: 994790ab36e724d20100000065c04549f0a6545ddb2227e2 (good)
;; QUESTION SECTION:
;10.2.0.10.in-addr.arpa.       IN      PTR

;; ANSWER SECTION:
10.2.0.10.in-addr.arpa. 10800    IN      PTR      SERVEURDHCPP.sio.lan.

;; Query time: 0 msec
;; SERVER: 10.0.2.10#53(10.0.2.10) (UDP)
;; WHEN: Mon Feb 05 03:17:45 CET 2024
;; MSG SIZE rcvd: 113
```

**Si tout fonctionne, alors, vous avez réussi à configurer
un serveur DHCP/DNS**