

## Rapport : Configuration du Serveur DHCP sous Linux

Fait par: Fatima ezzahraa Qanoufi

### I. Introduction:

C'est un rapport qui se penche sur la mise en place d'un serveur DHCP sous Linux. Ce dernier exploite le protocole DHCP (Dynamic Host Configuration Protocol) afin d'automatiser l'attribution d'adresses IP au sein d'un réseau. L'objectif est de simplifier la gestion des adresses IP, de minimiser les erreurs humaines et d'optimiser l'efficacité du réseau.

### II. Fondements Théoriques:

#### **1. Dynamic Host Configuration Protocol (DHCP):**

Le DHCP est un protocole réseau qui automatise la configuration des adresses IP et autres paramètres réseau pour les appareils connectés. Il assigne dynamiquement les adresses IP, la passerelle par défaut et les serveurs DNS.

#### **2. Avantages du DHCP:**

-- **Réduction des conflits d'adresses IP** : En attribuant dynamiquement les adresses IP, le serveur DHCP minimise les risques de conflits d'adresses IP sur le réseau.

-- **Flexibilité et évolutivité** : Le serveur DHCP permet de facilement ajuster la configuration du réseau en fonction des besoins changeants, ce qui le rend adaptable à des environnements en croissance ou en évolution.

### III. Objectifs de Configuration:

Le but de cette configuration est d'établir un serveur DHCP afin d'assigner automatiquement des adresses IP aux clients du réseau. Les objectifs spécifiques impliquent la délimitation d'une plage d'adresses IP, la précision des paramètres réseau, et garantissent une gestion efficace des adresses allouées.

#### **IV. Logiciels Utilisés:**

Le choix du logiciel **ISC DHCP Server** s'est basé sur sa réputation de fiabilité et de flexibilité. Il est largement adopté dans les environnements Linux et propose une configuration robuste pour **le serveur DHCP**.

#### **V. Configuration du Serveur DHCP**

##### **1. Installation du Serveur DHCP**

Le processus d'installation du serveur DHCP sous Linux a été effectué en utilisant les commandes standards du gestionnaire de paquets.

-- La commande: **sudo apt-get install isc-dhcp-server:** pour installer mon serveur dhcp.

-- Pour vérifier la version installée du serveur dhcp: La commande **dhcpcd --version**.

```
root@FatimaEzzahraaQanoufi:/home/fati# dhcpcd --version
isc-dhcpd-4.4.1
root@FatimaEzzahraaQanoufi:/home/fati#
```

##### **2. Configuration du Fichier dhcpcd.conf**

Le fichier de configuration principal, situé à /etc/dhcp/dhcpd.conf, a été ajusté pour Incorporez des instructions de configuration afin de délimiter votre réseau, définir la plage d'adresses IP à allouer et spécifier d'autres options nécessaires comme vous voyez ci-dessous :

```
root@FatimaEzzahraaQanoufi: /home/fati
GNU nano 4.8 /etc/dhcp/dhcpd.conf
# dhcpd.conf
#
# Sample configuration file for ISC dhcpd
#
# Attention: If /etc/ltsp/dhcpd.conf exists, that will be used as
# configuration file instead of this file.
#
# option definitions common to all supported networks...
subnet 192.168.1.0 netmask 255.255.255.0 {
    range 192.168.1.100 192.168.1.200;
    option routers 192.168.1.1;
    option domain-name-servers 8.8.8.8, 8.8.4.4;
    default-lease-time 600;
    max-lease-time 7200;
}

# The ddns-updates-style parameter controls whether or not the server wi
# attempt to do a DNS update when a lease is confirmed. We default to th
# behavior of the version 2 packages ('none', since DHCP v2 didn't
[ Read 113 lines ]
^G Get Help ^O Write Out ^W Where Is ^K Cut Text ^J Justify ^C Cur
^X Exit ^R Read File ^\ Replace ^U Paste Text ^T To Spell ^_ Go T
```

-- **subnet 192.168.1.0 netmask 255.255.255.0**: Définit le sous-réseau avec l'adresse IP de réseau 192.168.1.0 et un masque de sous-réseau de 255.255.255.0.

-- **range 192.168.1.100 192.168.1.200**: Spécifie la plage d'adresses IP que le serveur DHCP peut attribuer aux clients, de 192.168.1.100 à 192.168.1.200.

-- **option routers 192.168.1.1**: Indique la passerelle par défaut à utiliser par les clients du réseau, définie sur l'adresse IP 192.168.1.1.

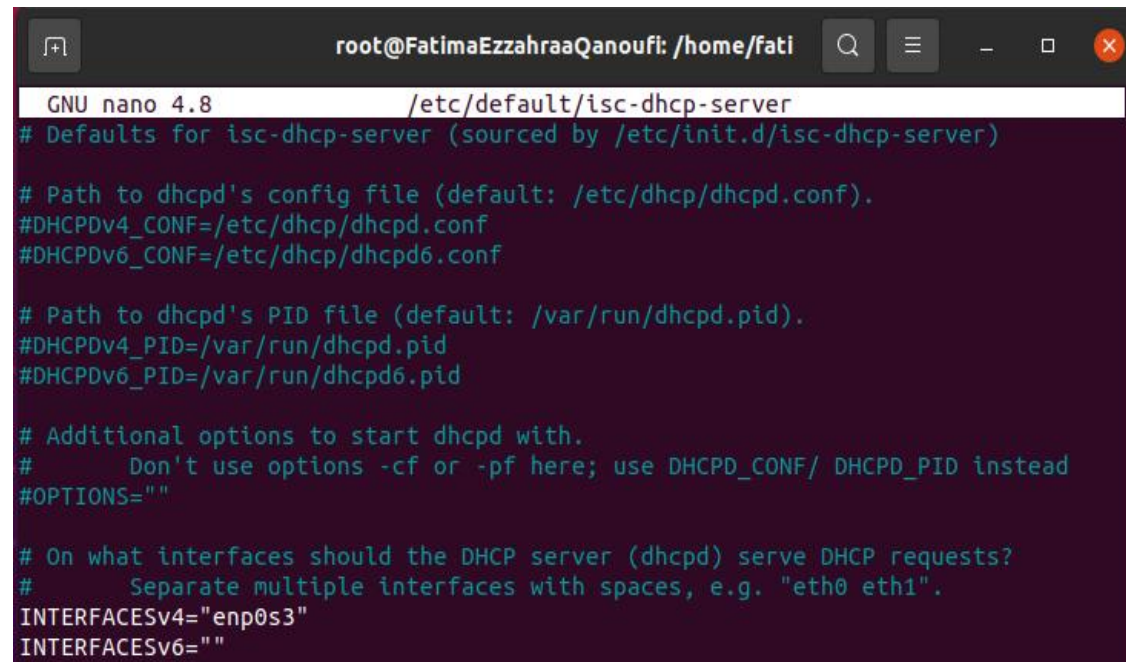
-- **option domain-name-servers 8.8.8.8, 8.8.4.4**: Spécifie les serveurs DNS que les clients du réseau utiliseront pour la résolution des noms de domaine, dans cet exemple, les serveurs DNS Google (8.8.8.8 et 8.8.4.4).

-- **default-lease-time 600**: Définit le temps de bail par défaut, soit la durée pendant laquelle une adresse IP attribuée est réservée pour un client, ici 600 secondes (10 minutes).

-- **max-lease-time 7200**: Spécifie la durée maximale du bail, soit la période maximale pendant laquelle une adresse IP peut être attribuée à un client, ici 7200 secondes (2 heures).

### 3. Configurer l'interface réseau :

On a ajouté le nom de notre interface qui est enp0s3 dans l'interface v4:



```
root@FatimaEzzahraaQanoufi: /home/fati
GNU nano 4.8 /etc/default/isc-dhcp-server
# Defaults for isc-dhcp-server (sourced by /etc/init.d/isc-dhcp-server)

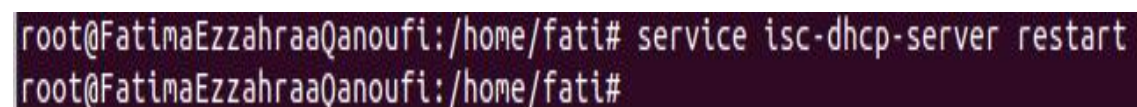
# Path to dhcpd's config file (default: /etc/dhcp/dhcpd.conf).
#DHCPDv4_CONF=/etc/dhcp/dhcpd.conf
#DHCPDv6_CONF=/etc/dhcp/dhcpd6.conf

# 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=""
```

### 4. Redémarrage de serveur dhcp:



```
root@FatimaEzzahraaQanoufi:/home/fati# service isc-dhcp-server restart
root@FatimaEzzahraaQanoufi:/home/fati#
```

## VI. Tests de Validation

### 1. Vérification de l'état actuel de notre serveur est ce que c'est actif ou non :



```

root@FatimaEzzahraaQanoufi:/home/fati# systemctl status isc-dhcp-server
● isc-dhcp-server.service - ISC DHCP IPv4 server
   Loaded: loaded (/lib/systemd/system/isc-dhcp-server.service; enabled; vend
   Active: active (running) since Wed 2024-03-06 09:56:20 +01; 29s ago
     Docs: man:dhcpd(8)
    Main PID: 1581 (dhcpd)
      Tasks: 4 (limit: 2415)
     Memory: 4.4M
    CGroup: /system.slice/isc-dhcp-server.service
            └─1581 dhcpd -user dhcpd -group dhcpd -f -4 -pf /run/dhcp-server/d>

09:56:20 06 مليس FatimaEzzahraaQanoufi sh[1581]: Wrote 0 leases to leases file.
09:56:20 06 مليس FatimaEzzahraaQanoufi dhcpd[1581]: PID file: /run/dhcp-server/>
09:56:20 06 مليس FatimaEzzahraaQanoufi dhcpd[1581]: Wrote 0 leases to leases fi>
09:56:20 06 مليس FatimaEzzahraaQanoufi dhcpd[1581]: Listening on LPF/enp0s3/08:>
09:56:20 06 مليس FatimaEzzahraaQanoufi sh[1581]: Listening on LPF/enp0s3/08:00:>
09:56:20 06 مليس FatimaEzzahraaQanoufi sh[1581]: Sending on LPF/enp0s3/08:00:>
09:56:20 06 مليس FatimaEzzahraaQanoufi sh[1581]: Sending on Socket/fallback/f>
09:56:20 06 مليس FatimaEzzahraaQanoufi dhcpd[1581]: Sending on LPF/enp0s3/08:>
09:56:20 06 مليس FatimaEzzahraaQanoufi dhcpd[1581]: Sending on Socket/fallbac>
09:56:20 06 مليس FatimaEzzahraaQanoufi dhcpd[1581]: Server starting service.
lines 1-20/20 (END)

```

## 2. Vérification de l'attribution de l'adresse IP d'abord pour la machine host où le serveur est installé par la commande ifconfig:

```

root@FatimaEzzahraaQanoufi:/home/fati# ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.1.1 netmask 255.255.255.0 broadcast 192.168.1.255
    ether 08:00:27:46:c2:d2 txqueuelen 1000 (Ethernet)
    RX packets 9 bytes 1409 (1.4 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 124 bytes 19356 (19.3 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 1834 bytes 133472 (133.4 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 1834 bytes 133472 (133.4 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

```

## 3. Vérification de l'attribution de l'adresse IP pour la machine cliente, dans mon cas c'est kali:

```

(kali@FatimaEzzahraaQanoufi)-[~]
$ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.1.100 netmask 255.255.255.0 broadcast 192.168.1.255
    inet6 fe80::a00:27ff:fe2f:fd2d prefixlen 64 scopeid 0<link>
    ether 08:00:27:2f:fd:2d txqueuelen 1000 (Ethernet)
    RX packets 7 bytes 986 (986.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 25 bytes 3809 (3.7 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

```

#### 4. Faire un test avec ping pour assurer la connectivité entre les deux machine :

```

(root@FatimaEzzahraaQanoufi)-[/home/kali]
# ping 192.168.1.1
PING 192.168.1.1 (192.168.1.1) 56(84) bytes of data.
64 bytes from 192.168.1.1: icmp_seq=1 ttl=64 time=1.19 ms
64 bytes from 192.168.1.1: icmp_seq=2 ttl=64 time=1.15 ms
64 bytes from 192.168.1.1: icmp_seq=3 ttl=64 time=0.814 ms
64 bytes from 192.168.1.1: icmp_seq=4 ttl=64 time=1.70 ms
64 bytes from 192.168.1.1: icmp_seq=5 ttl=64 time=1.22 ms
64 bytes from 192.168.1.1: icmp_seq=6 ttl=64 time=1.10 ms
64 bytes from 192.168.1.1: icmp_seq=7 ttl=64 time=1.37 ms
64 bytes from 192.168.1.1: icmp_seq=8 ttl=64 time=2.70 ms
64 bytes from 192.168.1.1: icmp_seq=9 ttl=64 time=2.36 ms
64 bytes from 192.168.1.1: icmp_seq=10 ttl=64 time=0.876 ms
64 bytes from 192.168.1.1: icmp_seq=11 ttl=64 time=1.28 ms
64 bytes from 192.168.1.1: icmp_seq=12 ttl=64 time=2.16 ms
64 bytes from 192.168.1.1: icmp_seq=13 ttl=64 time=1.64 ms
64 bytes from 192.168.1.1: icmp_seq=14 ttl=64 time=1.07 ms
64 bytes from 192.168.1.1: icmp_seq=15 ttl=64 time=1.13 ms
64 bytes from 192.168.1.1: icmp_seq=16 ttl=64 time=1.03 ms
64 bytes from 192.168.1.1: icmp_seq=17 ttl=64 time=1.11 ms
64 bytes from 192.168.1.1: icmp_seq=18 ttl=64 time=2.51 ms
64 bytes from 192.168.1.1: icmp_seq=19 ttl=64 time=0.925 ms
64 bytes from 192.168.1.1: icmp_seq=20 ttl=64 time=1.05 ms
^C
— 192.168.1.1 ping statistics —
20 packets transmitted, 20 received, 0% packet loss, time 19107ms
rtt min/avg/max/mdev = 0.814/1.418/2.695/0.555 ms

```



## VII. Problèmes Rencontrés et Solutions

Le problème rencontré était au niveau de l'attribution de l'adresse ip pour la machine host. Même si après la configuration du serveur et la vérification que notre serveur est bien actif, notre machine host n'a pas toujours d'adresse IP comme vous voyez dans la photo si-dessous:

```
root@FatimaEzzahraaQanoufi:/home/fati# ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
    ether 08:00:27:46:c2:d2  txqueuelen 1000  (Ethernet)
    RX packets 0  bytes 0 (0.0 B)
    RX errors 0  dropped 0  overruns 0  frame 0
    TX packets 101  bytes 16768 (16.7 KB)
    TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
    inet 127.0.0.1  netmask 255.0.0.0
    inet6 ::1  prefixlen 128  scopeid 0x10<host>
    loop txqueuelen 1000  (Local Loopback)
    RX packets 782  bytes 58648 (58.6 KB)
    RX errors 0  dropped 0  overruns 0  frame 0
    TX packets 782  bytes 58648 (58.6 KB)
    TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

Pour résoudre ce problème on a exécuté la commande ci-dessous pour attribuer a la main une adresse IP a notre machine HOST Ubuntu:

```
root@FatimaEzzahraaQanoufi:/home/fati# ifconfig enp0s3 192.168.1.1
```

Cette commande a résolu le problème avec succès et maintenant on a une adresse IP pour notre UBUNTU:

```
root@FatimaEzzahraaQanoufi:/home/fati# ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.1.1 netmask 255.255.255.0 broadcast 192.168.1.255
    ether 08:00:27:46:c2:d2 txqueuelen 1000 (Ethernet)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 109 bytes 17626 (17.6 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 782 bytes 58648 (58.6 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 782 bytes 58648 (58.6 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

## VIII. Conclusion

La configuration du serveur DHCP a été réalisée avec succès, mettant en évidence l'efficacité du protocole DHCP dans la gestion des adresses IP. Cette approche offre une solution automatisée et évolutive pour les réseaux, améliorant ainsi la flexibilité et la gestion simplifiée.