

# Installation et Configuration du service DHCP sous Linux

Après la version Windows, voici la version **serveur DHCP sous Linux Debian** !

Le DHCP souvenez-vous, c'est ce petit **service magique** qui fait gagner du temps aux **admins** et qu'on adore tous, surtout quand il y a beaucoup de postes dans l'infra ! 🤖

## Installer et configurer un serveur DHCP sous Debian

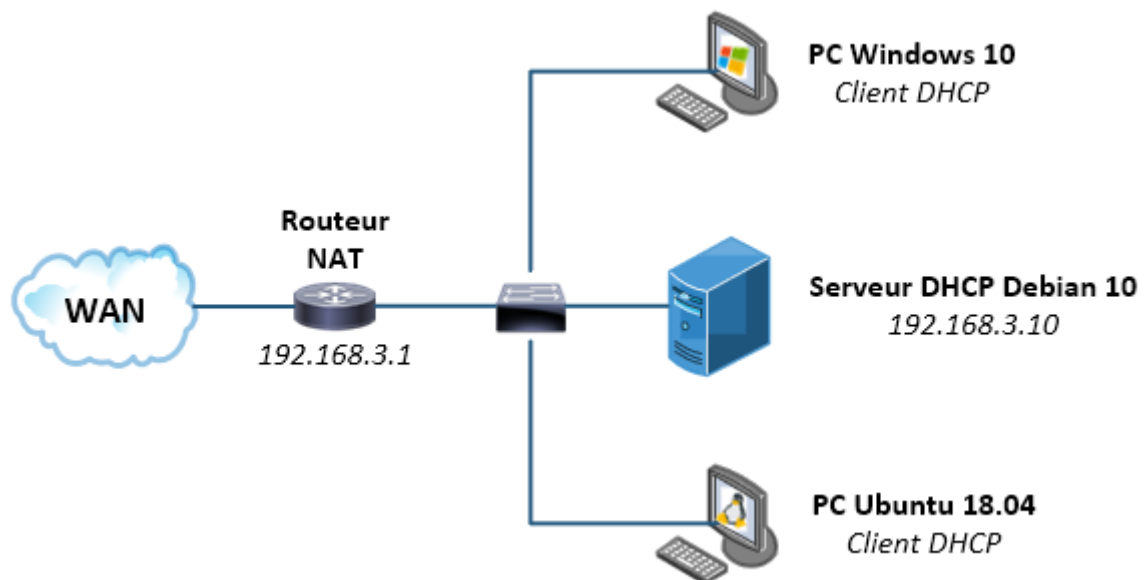
Pour rappel, le DHCP (*Dynamic Host Configuration Protocol*) est un **protocole réseau** qui permet d'assurer la configuration automatique des paramètres IP des machines qui en font la demande. Un serveur DHCP dans un réseau permet de ne pas avoir à configurer chaque poste à la main... (très pratique !)

**Il faut bien penser à passer sa machine en « réseau interne »**

Info + : Pour en savoir plus sur le protocole DHCP, consultez cet article : [Introduction au DHCP](#)

Et pour la version Windows, c'est par ici : [DHCP sous Windows](#)

Voici la topologie de mon labo du jour :



Comme d'habitude, pas besoin de beaucoup de machines pour réaliser ce tuto :

- 
- Une machine sous **Debian 10**
- Un client **Ubuntu**
- Un client **Windows**

- Un **routeur NAT** pour sortir du réseau local
- Un **switch** pour relier tout ce petit monde évidemment

Mon réseau est le **192.168.3.0/24**.

Le futur serveur DHCP doit disposer d'un adressage IP fixe et pouvoir accéder à Internet. Le serveur **DNS** du réseau sera celui de Google en **8.8.8.8**.

```
# The primary network interface
allow-hotplug ens33
auto ens33
iface ens33 inet static
    address 192.168.3.10/24
    gateway 192.168.3.1
```

```
domain localdomain
search localdomain
nameserver 8.8.8.8
```

Ensuite, on définit *“sur papier”* la configuration que nous voulons effectuer sur notre serveur DHCP. Voici ce que j’ai choisi de faire :

Étendue du sous réseau N°: <b>192.168.3.0/24</b>		Réservations	
		Adresse Mac	Adresse IP
Adresse début	<b>192.168.3.100</b>	<b>00:0C:29:EF:00:B1</b>	<b>192.168.3.150</b>
Adresse fin	<b>192.168.3.200</b>		
Masque	<b>255.255.255.0</b>		
Durée du bail	<b>3600 secondes</b>		
Options DHCP d'étendue			
Nom	Valeur		
<b>Routeur</b>	<b>192.168.3.1</b>		
Options DHCP de serveur			
Nom	Valeur		
<b>Serveur DNS</b>	<b>8.8.8.8</b>		

Si tout est prêt, vous pouvez poursuivre !

Pour mettre en place le service DHCP dans notre réseau local, nous allons utiliser le **paquet isc-dhcp-server**. Commençons par l'installer :

```
apt install isc-dhcp-server -y
```

Ensuite, nous devons **préciser sur quel interface du serveur, le “démon” (le “service”) va écouter** et donc attendre les requêtes des clients. Modifiez le fichier nécessaire avec la commande suivante :

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

```
# On what interfaces should the DHCP server (dhcpd) serve DHCP requests?
#       Separate multiple interfaces with spaces, e.g. "eth0 eth1".
INTERFACESv4="ens33"
#INTERFACESv6=""
```

Nous n'utiliserons **pas d'IPv6**, la ligne peut donc être **commentée**. En revanche, pour l'interface en IPv4, il vous faudra **ajouter le nom de l'interface réseau de votre serveur entre les guillemets**.

Ensuite, il faut éditer le fichier *dhcpd.conf* pour **configurer le service DHCP** :

```
nano /etc/dhcp/dhcpd.conf
```

**Soyez vigilants à la syntaxe employée dans ce fichier**, une erreur toute bête peut empêcher le service de démarrer et faire perdre du temps. **Chaque ligne de paramètre doit se terminer par le symbole “;”**. Veillez également à bien **ouvrir/fermer les blocs de paramètres** pour les étendues avec les symboles “{ }”.

Voici comment je configure la 1ère partie de ce fichier :

```
# dhcpd.conf
#
# Sample configuration file for ISC dhcpd
#
# option definitions common to all supported networks...
#option domain-name "example.org";
option domain-name-servers 8.8.8.8;

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;

# If this DHCP server is the official DHCP server for the local
# network, the authoritative directive should be uncommented.
authoritative;

# Use this to send dhcp log messages to a different log file (you also
# have to hack syslog.conf to complete the redirection).
#log-facility local7;
```

Regardons rapidement les options présentes.

Le premier paramètre est le “**domain-name**”. Comme son nom l’indique, on déclare ici **le nom de notre domaine**. Nous n’en avons pas, *ce paramètre est donc à commenter*. Ensuite nous avons l’option “**domain-name-servers**”. Si les serveurs DNS sont les mêmes pour toutes les étendues, on peut déclarer cette option à partir de ce point. Pour le DNS, j’ai choisi tout bêtement celui de Google, le 8.8.8.8. Peut importe les étendues, ce sera le même DNS partout, sinon, l’option peut être précisée dans chaque déclaration d’étendue. C’est donc ici une option dite “*de serveur*” et non pas “*d’étendue*”.

Les paramètres “**default-lease-time**” et “**max-lease-time**” sont les **durées des baux pour les adresses attribuées avant une libération ou un renouvellement**. Par défaut ces durées sont comprise entre **3600 secondes (1 heure) et 7200 secondes (2 heures)**. L’option “**ddns-update-style**” sert à **définir le type de mise à jour du DNS**. Nous n’en n’avons *pas besoin dans notre contexte*. Par défaut, ce paramètre est défini sur “*none*”.

Le paramètre **“authoritative”** peut être décommenté si ce serveur DHCP est le **serveur officiel du réseau local**, *ce qui sera le cas ici*.

L'option **“log-facility”** est le **niveau de log à conserver**. Par défaut, vous trouverez les logs du DHCP dans `/var/log/syslog`. Vous pouvez décommentez cette ligne pour déplacer les logs dans un autre endroit. Attention, des paramétrages supplémentaires non couverts par cet article seront nécessaires.

Voilà pour les configurations présentes dans ce fichier.

Maintenant, on va **déclarer l'étendue du réseau que devra desservir le DHCP**. Ajoutez ces lignes à la fin du fichier :

```
# Etendue LAN
subnet 192.168.3.0 netmask 255.255.255.0 {
option routers 192.168.3.1;
range 192.168.3.100 192.168.3.200;
}
```

*Info + : Sur isc-dhcp-server, il n'existe pas de paramètre permettant de créer des plages d'exclusions. Pour palier à ça, il vous mettre plusieurs lignes de “range” les unes à la suite des autres, toujours sans oublier*

```
# Etendue LAN
subnet 192.168.3.0 netmask 255.255.255.0 { # Adresse reseau et masque + ouverture bloc
option routers 192.168.3.1;                # Passerelle de l'etendue
range 192.168.3.100 192.168.3.200;         # Plage d'adresses a distribuer
}                                             # Fermeture bloc de l'etendue
```

On **redémarre le service** pour prendre en compte les modifications :

```
service isc-dhcp-server restart
```

**Vérifions sur le PC Ubuntu** s'il a récupéré une configuration IP automatiquement :

```
tech@Ubuntu-GUI:~$ ip a s ens33
2: ens33: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 00:0c:29:ef:00:b1 brd ff:ff:ff:ff:ff:ff
    inet 192.168.3.101/24 brd 192.168.3.255 scope global dynamic noprefixroute ens33
        valid_lft 454sec preferred_lft 454sec
    inet6 fe80::5bf6:abfe:55e3:715b/64 scope link noprefixroute
        valid_lft forever preferred_lft forever

tech@Ubuntu-GUI:~$ ping google.fr
PING google.fr (172.217.16.131) 56(84) bytes of data.
64 bytes from zrh04s06-in-f131.1e100.net (172.217.16.131): icmp_seq=1 ttl=127 time=20.6 ms
64 bytes from zrh04s06-in-f131.1e100.net (172.217.16.131): icmp_seq=2 ttl=127 time=21.8 ms
64 bytes from zrh04s06-in-f131.1e100.net (172.217.16.131): icmp_seq=3 ttl=127 time=21.0 ms
64 bytes from zrh04s06-in-f131.1e100.net (172.217.16.131): icmp_seq=4 ttl=127 time=21.3 ms
^C
--- google.fr ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3005ms
rtt min/avg/max/mdev = 20.647/21.233/21.843/0.452 ms
```

**C'est bien le cas ! Le PC Linux a l'adresse IP 192.168.3.101 et communique bien avec l'extérieur.**

Regardons avec un logiciel comme [Wireshark](#) ce qu'il se passe entre les **clients DHCP** et le **serveur DCHP** :

61	34.743817	0.0.0.0	255.255.255.255	DHCP	344	DHCP Discover
64	35.745695	192.168.3.10	192.168.3.100	DHCP	342	DHCP Offer
65	35.746240	0.0.0.0	255.255.255.255	DHCP	370	DHCP Request
66	35.747346	192.168.3.10	192.168.3.100	DHCP	342	DHCP ACK

Pour capturer ces trames, j'ai démarré le PC sous Windows. On voit bien les différentes requêtes effectuées entre le client DHCP et le serveur DHCP.

Si je vérifie la configuration de mon PC Windows :

```
C:\Users\tech>ipconfig /all

Configuration IP de Windows

    Nom de l'hôte . . . . . : DESKTOP-HJ25I7C
    Suffixe DNS principal . . . . . :
    Type de noeud . . . . . : Hybride
    Routage IP activé . . . . . : Non
    Proxy WINS activé . . . . . : Non

Carte Ethernet Ethernet0 :

    Suffixe DNS propre à la connexion. . . :
    Description. . . . . : Intel(R) 82574L Gigabit Network Connection
    Adresse physique . . . . . : 00-0C-29-B2-7C-97
    DHCP activé. . . . . : Oui
    Configuration automatique activée. . . : Oui
    Adresse IPv4. . . . . : 192.168.3.100(préfér  )
    Masque de sous-r  seau. . . . . : 255.255.255.0
    Bail obtenu. . . . . : samedi 28 d  cembre 2019 16:18:50
    Bail expirant. . . . . : samedi 28 d  cembre 2019 16:28:49
    Passerelle par d  faut. . . . . : 192.168.3.1
    Serveur DHCP . . . . . : 192.168.3.10
    Serveurs DNS. . . . . : 8.8.8.8
    NetBIOS sur Tcpip. . . . . : Activ  

C:\Users\tech>ping google.fr

Envoi d'une requ  te 'ping' sur google.fr [172.217.16.131] avec 32 octets de donn  es :
R  ponse de 172.217.16.131 : octets=32 temps=20 ms TTL=127
R  ponse de 172.217.16.131 : octets=32 temps=20 ms TTL=127
R  ponse de 172.217.16.131 : octets=32 temps=19 ms TTL=127
R  ponse de 172.217.16.131 : octets=32 temps=20 ms TTL=127

Statistiques Ping pour 172.217.16.131:
    Paquets : envoy  s = 4, re  us = 4, perdus = 0 (perte 0%),
    Dur  e approximative des boucles en millisecondes :
        Minimum = 19ms, Maximum = 20ms, Moyenne = 19ms
```

**Il a bien obtenu une adresse IP dans la plage d  clar  e dans le service DHCP !**

Pour voir les baux en cours d'utilisation sur le serveur DHCP, vous pouvez lire le contenu du fichier `/var/lib/dhcp/dhcpd.leases` avec `cat`.

```

root@SRV-DHCP:/# cat /var/lib/dhcp/dhcpd.leases
# The format of this file is documented in the dhcpd.leases(5) manual page.
# This lease file was written by isc-dhcp-4.4.1

# authoring-byte-order entry is generated, DO NOT DELETE
authoring-byte-order little-endian;

lease 192.168.3.101 {
  starts 6 2019/12/28 15:33:24;
  ends 6 2019/12/28 15:43:24;
  tstp 6 2019/12/28 15:43:24;
  cltt 6 2019/12/28 15:33:24;
  binding state free;
  hardware ethernet 00:0c:29:ef:00:b1;
}
lease 192.168.3.100 {
  starts 6 2019/12/28 15:35:25;
  ends 6 2019/12/28 15:45:25;
  tstp 6 2019/12/28 15:45:25;
  cltt 6 2019/12/28 15:35:25;
  binding state free;
  hardware ethernet 00:0c:29:b2:7c:97;
  uid "\001\000\014)\262|\227";
  set vendor-class-identifier = "MSFT 5.0";
}
server-uid "\000\001\000\001%\232.\275\000\014)\263\2705";

```

Les baux en cours correspondent bien aux adresses que l'on a vu sur les deux PCs clients.

Nous avons également à **mettre en place une réservation pour un PC** bien spécifique. Pour cela, il faut commencer par **recupérer son adresse physique (MAC)**. Depuis le PC Linux pour lequel nous voulons réserver une adresse IP, lancer la commande :

```
ip a s ens33 | grep "ether"
```

```

tech@PC-BOSS-Ubuntu:~$ ip a s ens33 | grep "ether"
link/ether 00:0c:29:ef:00:b1 brd ff:ff:ff:ff:ff:ff

```

Notez son adresse MAC (*partie située entre "ether" et "brd"*) et retournez sur le serveur DHCP.

Modifiez de nouveau le fichier *dhcpd.conf* et dans l'étendue précédemment créée, juste avant de refermer le bloc de paramètres (sur la ligne précédent le symbole "}"), ajoutez les lignes suivantes en adaptant l'adresse MAC par celle que vous avez noté :

```

# Reservation
host pc-boss {
  hardware ethernet 00:0C:29:EF:00:B1;
  fixed-address 192.168.3.150;
}

```

Ce qui vous donnera finalement une configuration de ce genre :

```
# Etendue LAN
subnet 192.168.3.0 netmask 255.255.255.0 {
option routers 192.168.3.1;
range 192.168.3.100 192.168.3.200;
    #Reservation
    host pc-boss {
        hardware ethernet 00:0c:29:EF:00:B1;
        fixed-address 192.168.3.150;
    }
}
```

Vérifions sur le client concerné qu'il reçoit bien l'adresse IP qui lui est réservée :

```
tech@PC-BOSS-Ubuntu:~$ ip a s ens33
2: ens33: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 00:0c:29:ef:00:b1 brd ff:ff:ff:ff:ff:ff
    inet 192.168.3.150/24 brd 192.168.3.255 scope global dynamic noprefixroute ens33
        valid_lft 452sec preferred_lft 452sec
    inet6 fe80::5bf6:abfe:55e3:715b/64 scope link noprefixroute
        valid_lft forever preferred_lft forever
```

C'est bien le cas, nous voyons l'adresse MAC que nous avons déclaré dans le service DHCP et l'adresse IP 192.168.3.150 qui lui est attribuée ; **la réservation est opérationnelle.**

**Voilà pour cet article sur la configuration d'un serveur DHCP sous Linux !**

Rien de bien compliqué, une fois encore, il suffit d'être vigilant sur la syntaxe que l'on utilise

Installer FTP : [Partie 7. Comment installer un serveur FTP sur Debian 11? | \(lucidar.me\)](#)