

# Serveur DHCP rédigé pour AlmaLinux 8.5

Hainaut Patrick 2022

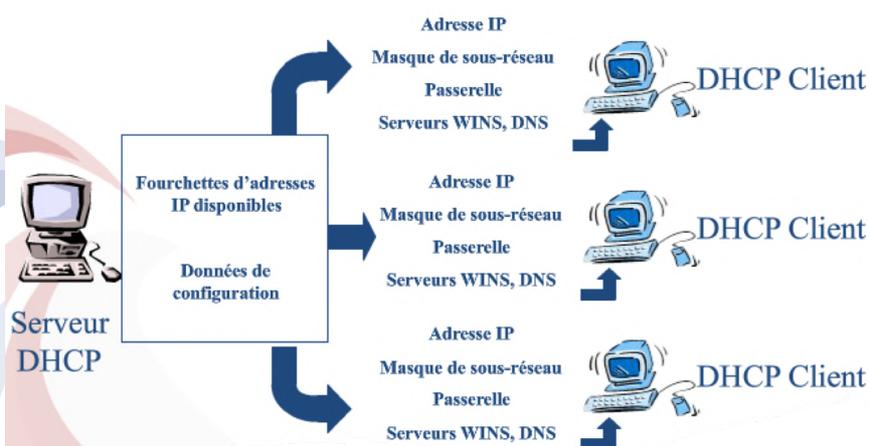
## But de cette présentation

- Vous permettre de comprendre et de configurer le service DHCP sur un serveur Linux via l'invite de commande
- Voir comment configurer les cartes réseaux via l'invite de commande

## Le protocole DHCP

- DHCP est un protocole client/serveur qui permet de centraliser et d'automatiser la configuration des données TCP/IP et d'affecter dynamiquement les adresses IP et paramètres associés
- En plus de l'adresse IP, il est possible de télécharger sur le client DHCP plus de 50 paramètres supplémentaires, en particulier:
  - Le masque de sous-réseau
  - La passerelle par défaut
  - Le(s) serveur(s) DNS

## Le protocole DHCP



## Le protocole DHCP

- On parle d'adresse dynamique pour un client DHCP et d'adresse statique pour une configuration manuelle. Les deux peuvent coexister
- Les avantages DHCP sont:
  - Le gain de productivité par l'absence de configuration manuelle
  - Une modification éventuelle de la plage d'adresse est grandement simplifiée
  - Les erreurs de configuration sont impossibles en production
  - Il est possible de réserver une adresse pour un client afin qu'il utilise toujours la même

## Processus d'acquisition d'une adresse IPv4

- Lorsque l'hôte se connecte sur un réseau, il envoie un message en broadcast (adresse IP source 0.0.0.0 et adresse IP destination 255.255.255.255) appelé DHCPDISCOVER du port UDP 68 vers le port UDP 67 pour demander une IP
- Tout serveur DHCP recevant le message DHCPDISCOVER doit traiter cette requête
- S'ils sont plusieurs, il répondent tous mais ce sera le plus rapide qui sera pris en compte

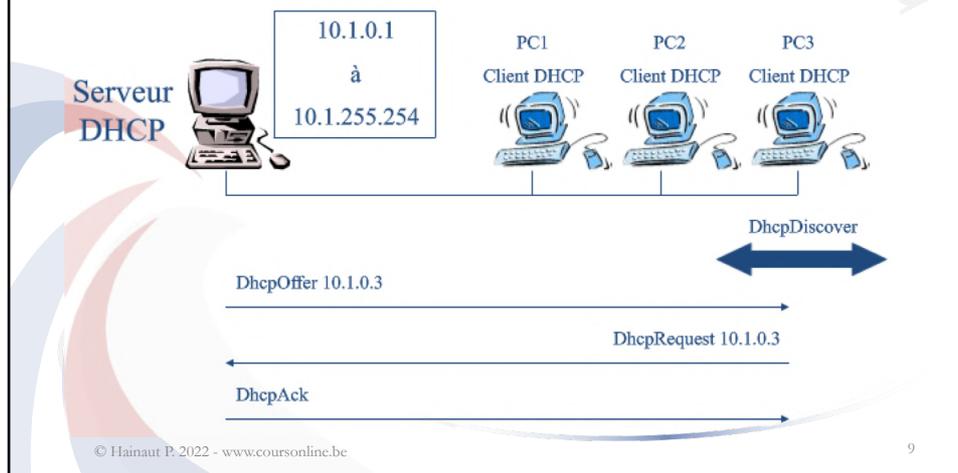
## Processus d'acquisition d'une adresse IPv4

- Le serveur propose une adresse IP au client via un message DHCP OFFER depuis le port UDP 67 vers le port UDP 68 en indiquant aussi la durée du bail proposé et son adresse IP
- Ce message est soit en broadcast, soit en unicast, suivant la position de certains bits dans le message DHCP (pas important à notre niveau d'étude)

## Processus d'acquisition d'une adresse IPv4

- Le client retourne un message de diffusion DHCPREQUEST comportant l'identificateur du serveur DHCP choisi afin de dire au serveur qu'il veut utiliser cette adresse
- Le message en diffusion indique aux autres serveurs DHCP éventuels qu'ils ne sont pas choisis.
- Le serveur répond par un message DHCPACK unicast, ce qui permet au client d'utiliser l'adresse IP pendant la durée du bail

## Processus d'acquisition d'une adresse IPv4



## Processus d'acquisition d'une adresse IPv4

- Le bail définit la durée d'utilisation de l'adresse IP par l'ordinateur client
  - La valeur par défaut est de 3 jours
- A 50% de la durée du bail, le client DHCP essaie automatiquement de renouveler le bail
  - Par un paquet DhcpRequest
- Si ce n'est pas possible, il réessaie à 87,5% de la durée du bail, et si cela ne fonctionne pas, l'adresse IP est libérée à l'expiration et le client DHCP doit recommencer le processus complet
- Le renouvellement du bail (DHCPREQUEST et DHCPACK) se fait par message unicast

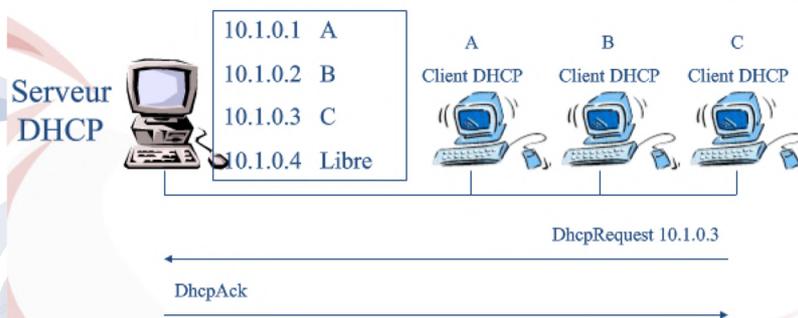
© Hainaut P. 2022 - www.coursonline.be

10

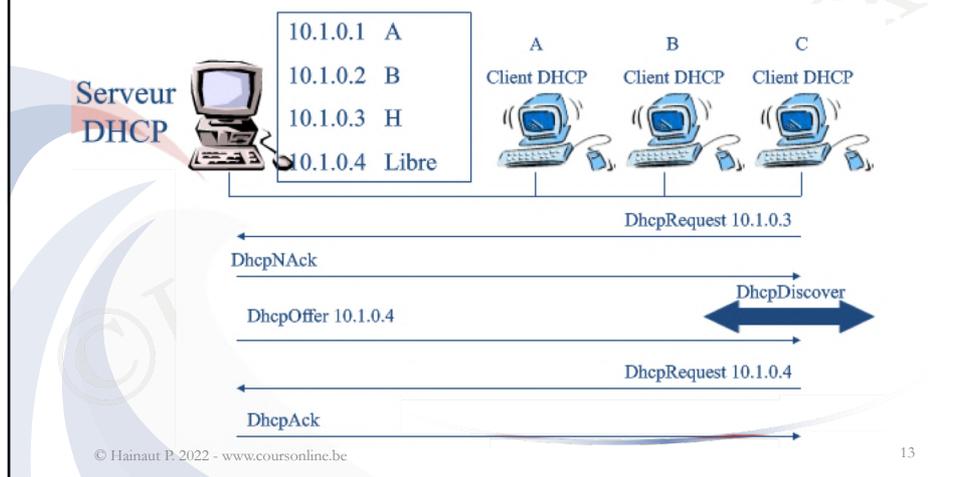
## Processus d'acquisition d'une adresse IPv4

- Lors d'un redémarrage, le client envoie directement un message DHCPREQUEST
- Si l'adresse est toujours disponible, le serveur DHCP répond par un message DHCPACK
- Si pas, il répond par un message DHCPNACK et le client doit recommencer entièrement le processus d'acquisition

## Processus d'acquisition d'une adresse IPv4



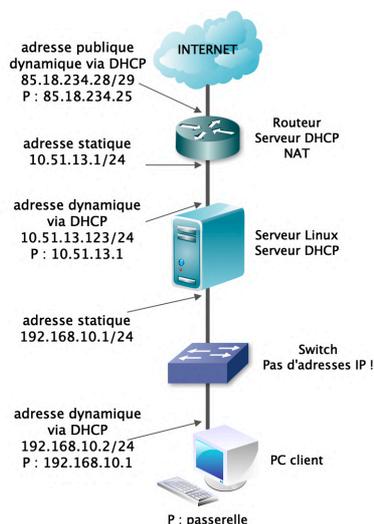
## Processus d'acquisition d'une adresse IPv4



## Prérequis pour l'installation

- Le serveur doit disposer d'une adresse IP sur l'interface qui écoutera les requêtes DHCP des clients
- Cette adresse doit être statique pour que les clients puissent renouveler leur bail

## Schéma de principe



© Hainaut P. 2022 - www.coursonline.be

15

Le serveur Linux est client DHCP sur l'interface réseau reliée au routeur (enp0s3 par exemple) et serveur DHCP sur l'autre (enp0s8 par exemple)

Sur cette dernière carte, l'adresse IP sera configurée en statique

Ces deux cartes réseaux sont obligatoirement dans des réseaux différents ! (le serveur joue le rôle de routeur)

Le PC client reçoit ses paramètres IP, dynamiquement, du serveur Linux

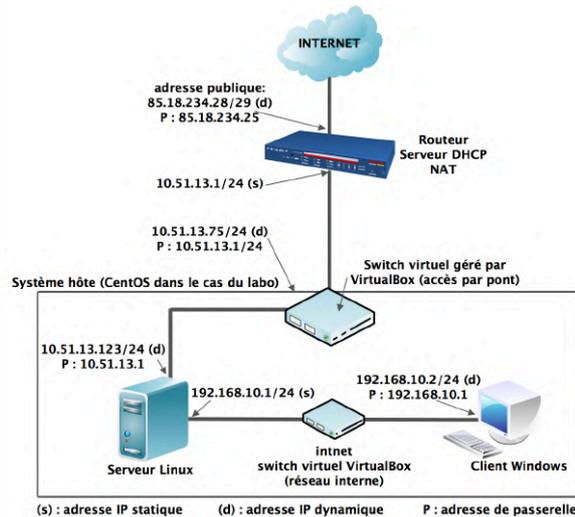
## Cas des machines réelles

- On prend une machine pour constituer notre serveur, avec deux cartes réseaux
- Une carte sera reliée à la connexion Internet (via routeur, serveur, modem,...) -> réseau WAN (Wide Area Network, réseau étendu)
- L'autre sera reliée à un switch auquel se connecteront les PC clients -> réseau LAN (Local Area Network, réseau local)
- Dans notre exemple, on utilise un seul PC client, mais dans la réalité, ils seront, bien sûr, plusieurs...

© Hainaut P. 2022 - www.coursonline.be

16

## Cas des machines virtuelles



Si nous utilisons VirtualBox, nous configurerons, pour le serveur Linux, Carte1 pour qu'elle ait accès à Internet (accès par pont) et Carte2 comme carte interne (réseau interne)

Les paramètres IP forment un exemple cohérent, mais construisez votre propre plan d'adressage

## Cas des machines virtuelles

- Dans VirtualBox, quand on choisit « accès par pont », on rajoute une nouvelle machine dans le réseau du PC hôte

Réseau LAN de l'hôte (Ex.:10.51.13.0/24)



- Dans le cas du labo, notre serveur Linux recevra ses paramètres IP en DHCP sur la carte réseau configurée en accès par pont

## Cas des machines virtuelles

- Quand on choisit « réseau interne », on crée un réseau privé entre les machines 'affiliées' à ce réseau interne
- Le regroupement se fait par le nom du réseau interne qui est aussi le nom du switch virtuel mis en place par VirtualBox (ici intnet)  
-> Plusieurs réseaux internes peuvent coexister



© Hainaut P. 2022 - www.coursonline.be

19

## Noms des cartes réseau sous Linux

- Depuis la sortie du noyau Linux 4.4, les cartes réseau sont nommées en fonction de leur nature et de leur emplacement dans l'ordinateur:
  - D'abord un préfixe en deux lettres: - en pour Ethernet
  - Une lettre suivant le type de bus: - p pour un slot PCI
  - Un numéro de port
  - Pour les cartes enfichées, on peut trouver en plus, s<slot>
- Exemple: *enp5s3*
- Dans le reste de la présentation, nous utiliserons les cartes réseau *enp0s3* (équivalent *eth0*) et *enp0s8* (équivalent *eth1*), adaptez les commandes en fonction de vos cartes réseau ;-)

© Hainaut P. 2022 - www.coursonline.be

20

## Noms des cartes réseau sous Linux

- Pourquoi ce système de noms prévisibles (Predictable Network Interface Names) ?
  - Parce qu'avec l'ancien système de nommage eth0, eth1, ..., les interfaces étaient nommées suivant l'ordre de détection par le système
  - Qu'après un redémarrage ou un changement de carte réseau, le nom de la carte pouvait donc avoir changé, ce qui posait problème pour tous les services réseaux basés sur le nom de la carte
- Cependant, ce système étant relativement nouveau, vous trouverez encore beaucoup de littérature technique employant les noms ethX
- Rien ne change, il suffit d'adapter le nom

## Configuration de la carte réseau

- **ip addr** (remplace **ifconfig**)
  - c'est l'équivalent de **ipconfig** sous Windows
  - **ip addr** permet de voir les cartes présentes
  - **ip addr show dev enp0s3** permet d'avoir les détails de cette carte

```
root@srvLX01:~# ip addr show enp0s3
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP
group default qlen 1000
    link/ether 08:00:27:ab:c4:43 brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.123/24 brd 192.168.1.255 scope global enp0s3
        valid_lft forever preferred_lft forever
    inet6 fe80::a00:27ff:feab:c443/64 scope link
        valid_lft forever preferred_lft forever
root@srvLX01:~# _
```

## Configuration de la carte réseau

- **ip addr add** *ip\_address/netmask dev interface* assigne des paramètres IP (n'est effectif que durant la session, disparaît au redémarrage) comme **ifconfig** *interface ip\_address netmask*

Exemple: `ip addr add 10.0.20.1/255.255.255.0 dev enp0s8`  
ou `ip addr add 10.0.20.1/24 dev enp0s8`

## Configuration de la carte réseau

- Contrairement à `ifconfig`, on peut assigner plusieurs adresses IP à une interface
- **ip addr del** *ip\_address/netmask dev interface* enlève des paramètres IP

Exemple: `ip addr del 10.0.20.1/255.255.255.0 dev enp0s8`  
ou `ip addr del 10.0.20.1/24 dev enp0s8`

- **ip addr flush dev interface** (`ip a f interface`) enlève tous les paramètres IP de l'interface

Exemple: **ip -4 a f enp0s8** enlèvera tous les paramètres IPv4

## Configuration de la carte réseau

- **ip link set dev *interface* up** permet d'activer une interface

Exemple: `ip l s enp0s8 up`

- **ip link set dev *interface* down** permet de désactiver une interface

Exemple: `ip l s enp0s8 down`

## Configuration de la carte réseau

- Pour garder les réglages après redémarrage, il faut éditer le fichier `/etc/sysconfig/network-script/ifcfg-*`:  
(avec \* le nom de l'interface réseau)

Exemple: `vi /etc/sysconfig/network-script/ifcfg-enp0s8`

vi est un éditeur de texte parmi tant d'autres, son avantage, c'est qu'il est présent par défaut dans toutes les distributions Linux et, si on connaît quelques raccourcis clavier, facile à utiliser (voir Manip5)

## Configuration de la carte réseau

- Une fois le fichier **ifcfg-\*** sauvé, celui-ci n'étant qu'un fichier texte, la modification n'est prise en compte que si on redémarre le service réseau
- Pour cela:  
**systemctl restart NetworkManager**  
  
ou  
**nmcli networking off**  
**nmcli networking on**

## Configuration du réseau pour notre manipulation

- Nous avons deux cartes réseaux sur notre serveur
- Avec la configuration VirtualBox, nous avons la carte enp0s3 qui est en accès par pont et qui doit donc recevoir une adresse IP du serveur DHCP du labo
- Nous avons aussi une carte enp0s8 en réseau interne, qui sera connecté au client Windows, et à qui on doit attribuer des paramètres IP statiques

## Configuration du réseau pour notre manipulation

- Il y a lieu d'aller vérifier dans le fichier **ifcfg-enp0s3** que l'interface est bien en configuration par DHCP et activée

```
...  
DEVICE=enp0s3  
BOOTPROTO=dhcp  
ONBOOT=yes  
...
```

## Configuration du réseau pour notre manipulation

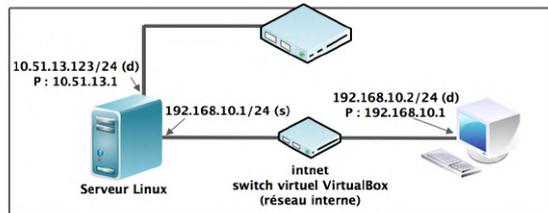
- Il faut ensuite modifier et compléter le fichier **ifcfg-enp0s8** en renseignant une configuration statique et en activant l'interface

```
...  
DEVICE=enp0s8  
BOOTPROTO=static  
IPADDR=192.168.10.1  
NETMASK=255.255.255.0  
ONBOOT=yes  
...
```

- Veuillez noter que des renseignements complémentaires sont présents dans ces fichiers

## /etc/sysconfig/network-script/ifcfg-enp0s8

- BOOTPROTO=static  
IPADDR=192.168.10.1  
NETMASK=255.255.255.0
- Cette adresse IP constituera donc l'adresse de passerelle pour les PC du réseau local (le Windows dans le cas de notre manip sous VirtualBox)
- Les PC du réseau local devront utiliser le même masque de sous-réseau



© Hainaut P. 2022 - www.coursonline.be

31

## Configuration du réseau pour notre manipulation

- Un redémarrage du réseau s'impose
- **ip addr** nous permettra de vérifier si *enp0s3* reçoit bien une IP et si *enp0s8* est configurée correctement
- **ping** [www.google.be](http://www.google.be) permet de vérifier qu'on a bien accès à Internet
- Si ce n'est pas le cas, revoyez la configuration des deux fichiers *ifcfg-\**

© Hainaut P. 2022 - www.coursonline.be

32

## Configuration du réseau pour notre manipulation

- Vous devriez obtenir un résultat similaire à ce qu'on trouve sur la photo écran suivante:

```
[root@srv01 ~]# ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN qlen 1
    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: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP qlen 1000
    link/ether 08:00:27:26:f4:66 brd ff:ff:ff:ff:ff:ff
    inet 10.51.13.84/24 brd 10.51.13.255 scope global dynamic enp0s3
        valid_lft 7198sec preferred_lft 7198sec
    inet6 fe80::6254:d6c8:6657:f998/64 scope link
        valid_lft forever preferred_lft forever
3: enp0s8: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP qlen 1000
    link/ether 08:00:27:8c:20:41 brd ff:ff:ff:ff:ff:ff
    inet 192.168.10.1/24 brd 192.168.10.255 scope global enp0s8
        valid_lft forever preferred_lft forever
    inet6 fe80::2671:feff:c5eb:e28e/64 scope link
        valid_lft forever preferred_lft forever
[root@srv01 ~]# _
```

© Hainaut P. 2022 - www.coursonline.be

33

## Configuration du réseau pour notre manipulation

- Il peut arriver aussi que l'interface cliente DHCP ne reçoive pas d'adresse du premier coup ...
- En cours de manipulation, si vous avez un problème réseau, pensez à vérifier que le serveur a toujours accès à Internet (**ping**) et redémarrer le réseau si nécessaire (**systemctl restart NetworkManager**)

© Hainaut P. 2022 - www.coursonline.be

34

## Mise en place du serveur DHCP

- Nous allons installer la fonction serveur DHCP sur notre serveur Linux, pour permettre au PC client d'obtenir ses paramètres IP, automatiquement, de notre serveur
- Notre serveur 'écouterà' les demandes sur la carte dont les paramètres sont définis statiquement (enp0s8 dans notre exemple)

## Installation d'une application sous AlmaLinux

- **yum install** ou mieux, son évolution **dnf install**

## Installation du serveur DHCP

- Elle consiste à installer le paquet dhcp via l'utilitaire yum:

```
dnf -y install dhcp-server
```

- L'argument -y (optionnel) permet de répondre automatiquement à toutes les questions

## Configuration de l'interface d'écoute

- Pour indiquer sur quelle interface 'écoute' notre serveur DHCP, jusque CentOS6, il fallait éditer le fichier `/etc/sysconfig/dhcpd` et remplir la ligne **DHCPARGS=** avec le nom de la carte réseau qui est connectée au réseau local
- → Plus besoin à partir CentOS7 et les dérivés de Red Hat Entreprise comme AlmaLinux, le serveur écoute sur la carte correspondant au range défini dans la configuration du serveur DHCP

## Configuration du serveur DHCP

- Le fichier de configuration se trouve dans **/etc/dhcp/dhcpd.conf**
- Ce fichier est quasi-vide
- Pour éditer un fichier exemple, copiez le fichier `/usr/share/doc/dhcp-server/dhcpd.conf.example` vers `/etc/dhcp/dhcpd.conf`
- Pour retrouver le fichier `dhcpd.conf.example` facilement, utilisez la commande `find` (voir Manip5)

## Configuration type

```
##### Options générales #####  
ddns-update-style none;  
option domain-name "atc.lan";  
option domain-name-servers 8.8.8.8;  
default-lease-time 3600;  
max-lease-time 7200;  
authoritative;  
##### RESEAUX #####  
subnet 192.168.10.0 netmask 255.255.255.0 {  
    option routers 192.168.10.1;  
    range 192.168.10.10 192.168.10.254;  
}
```

## Examen des directives

- `ddns-update-style none;`

*indique que le DNS ne doit pas se mettre à jour*

- option `domain-name "atc.lan";`

*permet de renseigner le nom de domaine du serveur DNS*

- option `domain-name-servers 8.8.8.8;`

*permet de renseigner l'adresse IP du serveur DNS*

## Examen des directives

- `default-lease-time 36000;`  
`max-lease-time 72000;`

*temps par défaut et maximum pour lesquels le bail est accordé (en secondes)*

- `authoritative;`

*Serveur DHCP en mode autoritaire. C'est lui qui répond aux demandes des clients.*

## Examen des directives

- subnet 192.168.10.0 netmask 255.255.255.0 {  
option routers 192.168.10.1;  
range 192.168.10.10 192.168.10.254;  
}

subnet 192.168.10.0: *définition du sous-réseau  
(attention ! Adresse de réseau !)*

netmask 255.255.255.0: *définition du masque de sous-réseau  
(valable pour tout le LAN)*

option routers 192.168.10.1: *définition de la passerelle  
(adresse du serveur du côté LAN)*

## Examen des directives

- subnet 192.168.10.0 netmask 255.255.255.0 {  
option routers 192.168.10.1;  
range 192.168.10.10 192.168.10.254;  
}

range 192.168.10.10 192.168.10.254 : *définition de l'intervalle d'adresses  
IP que le serveur DHCP peut distribuer*

*Ne peut pas comporter les adresses de réseau, de diffusion et du serveur !*

*Dans cet exemple, on aurait pu commencer avec l'adresse 192.168.10.2, mais  
on garde quelques adresses disponibles pour adresser des périphériques en statique  
comme une imprimante IP*

## Démarrage du serveur DHCP

- Cela se fait via la commande

```
systemctl start dhcpd.service
```

- Si vous voulez le stopper, remplacer **start** par **stop**
- Si vous voulez le redémarrer (pour qu'il recharge la nouvelle config.), remplacez **start** par **restart**
- Attention, sous AlmaLinux, le serveur DHCP n'est pas réactivé au démarrage de l'OS. Pour que ce soit le cas, il faut taper:

```
systemctl enable dhcpd.service
```

## Test au niveau du PC client

- Démarrez le PC client, et logez-vous dans son domaine (pas le domaine géré par Samba)
- dans une invite de commande, un **ipconfig** devrait vous renseigner sur le bon fonctionnement de votre serveur DHCP
- **ipconfig /all** permettra de voir les paramètres DNS donnés par votre serveur DHCP
- Si vous avez une adresse IP commençant par 169.254. ..., elle est donnée par Windows et il faut rafraichir l'adresse IP par **ipconfig /release** suivi de **ipconfig /renew**
- si tout est ok, passez au ppt suivant ...

## Conclusion

- Vous voilà maintenant avec un serveur Linux qui distribue les paramètres IP aux clients de votre réseau local
- Si vous suivez Manip8, vous pourrez activer l'accès Internet pour ces clients
- Merci de votre attention