

# Adressage IP

## Deuxième partie

Hainaut Patrick 2024

# But de cette présentation

- Maintenant qu'on sait ce qu'est une adresse IP, un masque de sous-réseau, une adresse de réseau, de diffusion, de passerelle, un DNS, le NAT, ... on peut poursuivre notre exploration
- On se focalise toujours sur IPv4

# Masque de sous-réseau

- Nous avons vu précédemment qu'un masque de sous-réseau pouvait prendre 3 valeurs qui dépendent du nombre d'octets fixés dans l'adresse IP: /8, /16 et /24
- Les octets fixés représentent le Net-ID, la partie réseau qui doit être identique pour toutes les machines d'un même réseau
- Les octets non-fixés représentent le Host-ID, la partie hôte (machine) qui doit être différente pour chaque machine

# Masque de sous-réseau

- Ca manque un peu de nuances ;-)
- On aimerait parfois avoir trois réseaux de 62 machines sans prendre pour cela 3 réseaux de 254 machines (/24)
- Ou avoir un réseau de 510 machines sans devoir prendre pour cela un réseau de 65534 machines (/16)

# Masque de sous-réseau

- Pour cela, on va utiliser des masques de sous-réseaux différents
- Par exemple, pour les réseaux de 62 machines, on prendra des /26
- Et pour le réseau de 510 machines, on prendra un /23
- Comment peut-on déterminer cela ?
  - > En descendant au niveau du bit

# Notions de binaire

- On a vu précédemment qu'un octet (byte en anglais) est composé de 8 bits (contraction de binary digit) qui peuvent prendre comme valeur 0 ou 1
- Exemples: 01001011, 00000001, 11111111, 11110000, 00000000, ...
- Chaque bit dans le byte a un poids différent selon son emplacement, comme en décimal (le 1 dans 1276 a un poids différent du 1 dans 2761)
- En binaire, c'est simple; le bit le plus à droite vaut 1 en décimal, le suivant vaut 2, celui après vaut 4, le suivant vaut 8, ...

# Notions de binaire

- Sur 8 bits, on a donc:

128	64	32	16	8	4	2	1
0	0	0	1	0	0	1	0

- On peut en déduire que le nombre binaire indiqué vaut 18 en décimal ( $16+2=18$ )
- 8 bits à 0 = 0 en décimal
- 8 bits à 1 = 255 en décimal ( $128+64+32+16+8+4+2+1$ )
- Si le bit de poids 1 est à 0 (le plus à droite), le nombre sera pair
- Avec 8 bits, on a 256 valeurs différentes (de 0 à 255)

# Notions de binaire

- Sur 16 bits, on a:

32768	16384	8192	4096	2048	1024	512	256	128	64	32	16	8	4	2	1
0	0	0	1	0	0	0	0	1	0	0	1	0	0	1	0

- Le nombre binaire représenté vaut 4242 ( $4096+128+16+2$ )
- 16 bits à 0 = 0 en décimal
- 16 bits à 1 = 65535 en décimal
- Avec 16 bits, on 65536 valeurs différentes (de 0 à 65535)



# Notions de binaire

- Faisons un peu d'arithmétique binaire:

$$\begin{array}{r} 0 \\ + 0 \\ \hline 0 \end{array} \quad \begin{array}{r} 0 \\ + 1 \\ \hline 1 \end{array} \quad \begin{array}{r} 1 \\ + 1 \\ \hline 10 \end{array}$$

- 00111111 -> 63 en décimal (1+2+4+8+16+32)

$$\begin{array}{r} + \quad 1 \\ \hline 01000000 \end{array} \rightarrow 64 \text{ en décimal}$$

-> Nous en déduisons un fait très pratique pour la suite du cours:  
une suite de bits tous à 1 vaut (en décimal) le bit suivant -1

# Notions de binaire

- Une autre relation importante entre décimal et binaire est l'exposant de 2
- Exemple:  $2^3 = 8$
- A notre niveau, ça veut dire qu'avec 3 bits, on obtient 8 combinaisons, et donc 8 valeurs décimales  
->  $2^{\text{nbr\_de\_bits}} = \text{nbr de valeurs (nbr d'adresses ou nbr de réseaux)}$
- Retenez  $2^5 = 32$  et  $2^{10} = 1024$ , vous déduirez les autres valeurs à partir de celles-là

# Plage IP à partir d'un couple IP/masque

- Si on reprend l'exemple des réseaux de 62 machines:
  - D'une part, on sait que chaque réseau contient une adresse de réseau et une adresse de diffusion qui ne sont pas attribuables  
Il faudra donc avoir des réseaux de 64 adresses (62 hôtes +2)
  - D'autre part, notre cours de math nous dit que  $64=2^6$   
Suivant la formule de la page précédente, on en déduit qu'il faut donc 6 bits pour pouvoir adresser nos machines  
On les appellera des bits machines, ils constituent le Host-ID et sont donc à droite dans le masque de sous-réseau

# Plage IP à partir d'un couple IP/masque

- Une adresse IP comportant 4 bytes, c'est à dire  $4 \times 8 \text{ bits} = 32 \text{ bits}$ , le Net-ID comportera donc  $32 - 6 \text{ (Host-ID)} = 26 \text{ bits}$
- Par conséquent le masque de sous-réseau comportera 26 bits  $\rightarrow /26$
- Cela donne en binaire: 11111111.11111111.11111111.11000000  
Et en décimal: 255.255.255.192  
 $192 = 128 + 64$  (poids des deux bits à 1 dans le dernier byte)
- Rappel: un masque de sous-réseau est toujours une suite continue de bits à 1

# Plage IP à partir d'un couple IP/masque

- Si on reprend l'exemple du réseau de 510 machines:
  - Il faudra un réseau de 512 adresses (510 hôtes +2)
  - On sait que  $512=2^9$  -> Host-ID = 9 bits -> Net-ID =  $32 - 9 = 23$  bits
  - -> Masque de sous-réseau: /23 ou 255.255.254.0

(11111111.11111111.11111110.00000000)

Net-ID

Host-ID

# Plage IP à partir d'un couple IP/masque

- Si on prend maintenant un couple adresse IP/masque, on peut déterminer l'étendue de ce réseau et le nombre d'adresses qui le compose
- Exemple 1: 192.168.21.178/27

Déterminons premièrement la limite Net-ID Host-ID

Le nombre de bits réseaux étant supérieur à 24, cette limite se situe dans le dernier byte

11100000 -> valeur binaire du masque pour le dernier byte

10110010 -> 178 (valeur du dernier byte de l'adresse IP) en binaire

101xxxxx -> valeur des bits fixés par le masque, les autres variant

# Plage IP à partir d'un couple IP/masque

101xxxxx -> valeur des bits fixés par le masque, les autres variant

Dans la présentation précédente, quand un octet du masque était à 0, l'octet correspondant au niveau de l'adresse IP variait de 0 à 255

Ici, quand un bit du masque est à 0, le bit correspondant au niveau de l'adresse IP varie de 0 à 1

Si tous les x sont à 0 (début de la plage -> adresse de réseau), le byte vaut 160 (128+32)

Si tous les x sont à 1 (fin de la plage -> adresse de diffusion), le byte vaut 191 (128+32+31)

# Plage IP à partir d'un couple IP/masque

Donc pour l'adresse 192.168.21.178/27

La plage IP sera 192.168.21.160/27 à 192.168.21.191/27

L'adresse de réseau sera 192.168.21.160/27

L'adresse du premier PC sera 192.168.21.161/27

L'adresse du dernier PC sera 192.168.21.190/27

L'adresse de diffusion sera 192.168.21.191/27

Le nombre d'adresses dans le réseau sera:

$32 \text{ bits au total} - 27 \text{ bits réseaux} = 5 \text{ bits machines} \rightarrow 2^5 = 32 \text{ adresses}$

Le nombre de machines maximum dans le réseau sera  $32 - 2 = 30$  hôtes



# Plage IP à partir d'un couple IP/masque

- Exemple 2: 192.168.21.178/22

Le nombre de bits réseaux étant situé entre 16 et 24, la limite Net-ID Host-ID se situe dans le 3<sup>ème</sup> byte

11111100 -> valeur binaire du masque pour le 3<sup>ème</sup> byte

00010101 -> 21 (valeur du 3<sup>ème</sup> byte de l'adresse IP) en binaire

000101xx -> valeur des bits fixés par le masque, les autres variant

Si x=0, le 3<sup>ème</sup> byte vaut 20 - si x=1, le 3<sup>ème</sup> byte vaut 23

Au niveau du 4<sup>ème</sup> byte, les 8 bits sont à 0 au niveau du masque (le masque étant une suite continue de bits à 1 qui s'arrête ici au niveau du 3<sup>ème</sup> byte)

# Plage IP à partir d'un couple IP/masque

00000000 -> valeur binaire du masque pour le 4<sup>ème</sup> byte  
10110010 -> 178 (valeur du 4<sup>ème</sup> byte de l'adresse IP) en binaire  
xxxxxxx -> valeur des bits fixés par le masque, les autres variant

Si  $x=0$ , le 4<sup>ème</sup> byte vaut donc 0 - si  $x=1$ , le 4<sup>ème</sup> byte vaut donc 255

La plage IP va de 192.168.20.0/22 à 192.168.23.255/22

L'adresse de réseau sera 192.168.20.0/22

L'adresse du premier PC sera 192.168.20.1/22

L'adresse du dernier PC sera 192.168.23.254/22

L'adresse de diffusion sera 192.168.23.255/22

Le nombre d'adresses dans le réseau sera:

32 bits au total – 22 bits réseaux = 10 bits machines ->  $2^{10} = 1024$  adr.

Le nombre de machines maximum dans le réseau sera  $1024 - 2 = 1022$  hôtes

# Exercices

- Soit à créer un réseau de:
  - 126 machines
  - 1022 machines
  - 2 machines

Trouver le Net-ID, le Host-ID et le masque de sous-réseaux

# Exercices

- Pour les adresses suivantes, trouvez l'adresse de réseau, l'adresse de diffusion, l'adresse du premier PC, l'adresse du dernier PC, le nombre d'adresses dans le sous-réseau ainsi que le nombre de machines

10.0.20.123/25

193.234.160.121/21

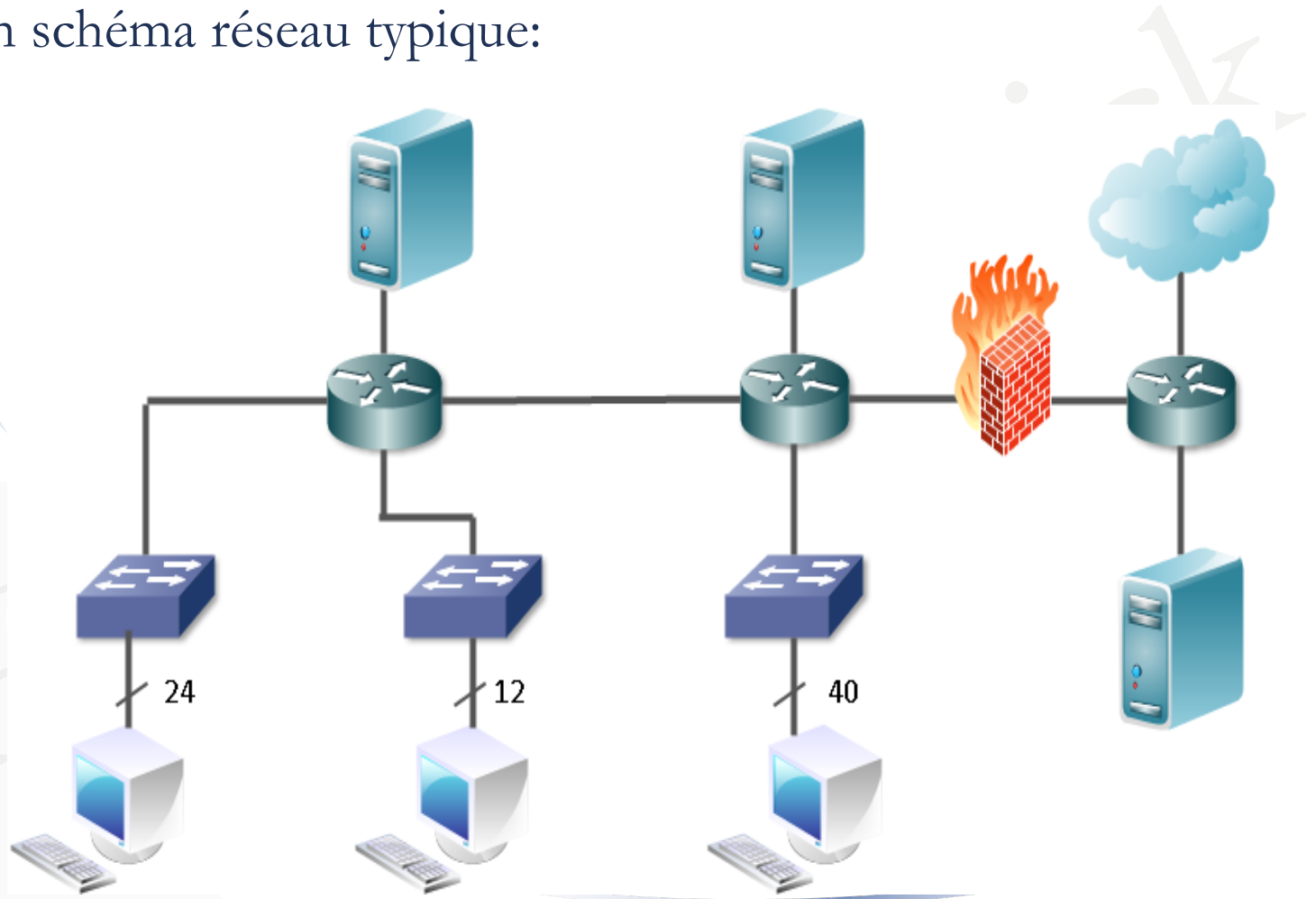
172.10.0.187/30

# Schémas réseaux

- Dessiner un schéma réseau clair et complet est essentiel pour avoir une vue d'ensemble de notre réseau
- Le schéma préétabli permettra de câbler et de configurer chaque machine sur le réseau facilement et de le dépanner par la suite
- Le schéma fait après coup permettra d'avoir une vue globale du réseau et de faciliter sa maintenance

# Schémas réseaux

- Voici un schéma réseau typique:



# Schémas réseaux

- On y distingue:



poste de travail (pdt)  
workstation



serveur  
server



commutateur  
switch



routeur  
router

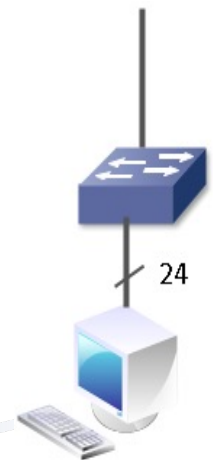


pare-feu  
firewal



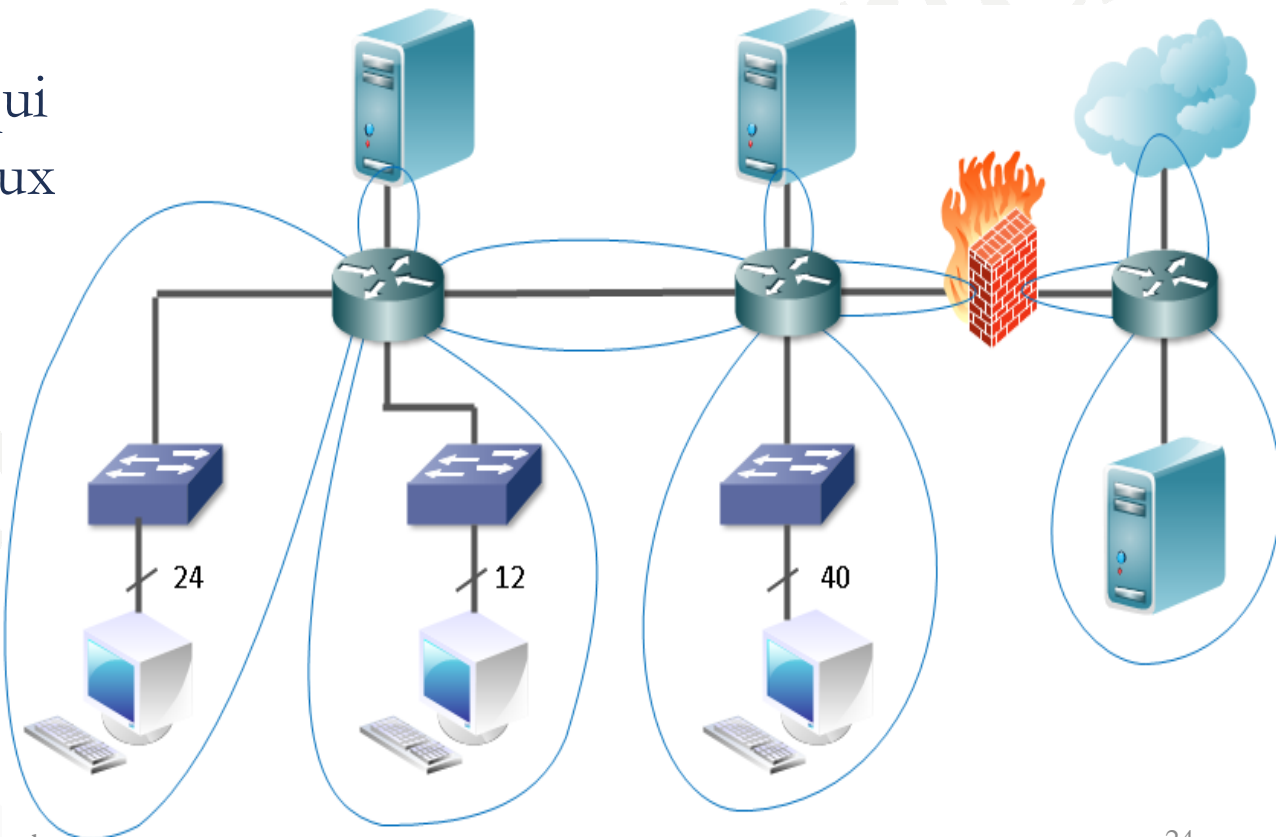
Internet  
Internet

- Au niveau des pdt, on ne va pas représenter sur le schéma le nombre exact de PC, ça rendrait le schéma illisible  
Par contre, on voit que du switch partent 24 câbles vers les PC



# Schémas réseaux

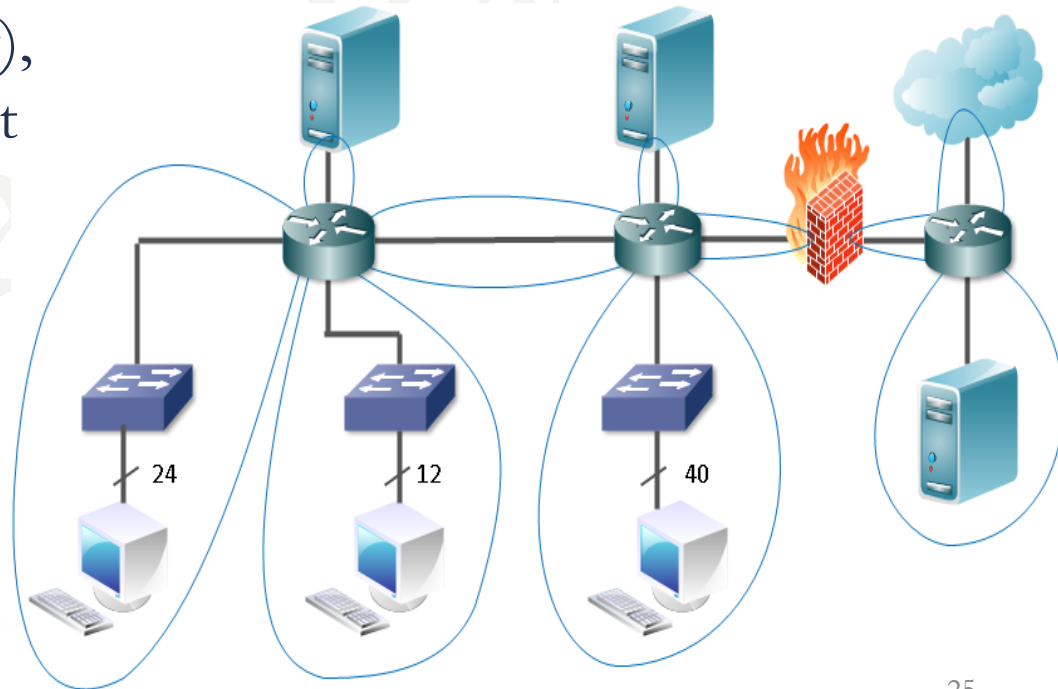
- Une fois qu'on a repéré les différents éléments, il faut distinguer le nombre de réseaux différents
- C'est le routeur qui délimite les réseaux
- Le firewall est ici considéré comme un routeur





# Schémas réseaux

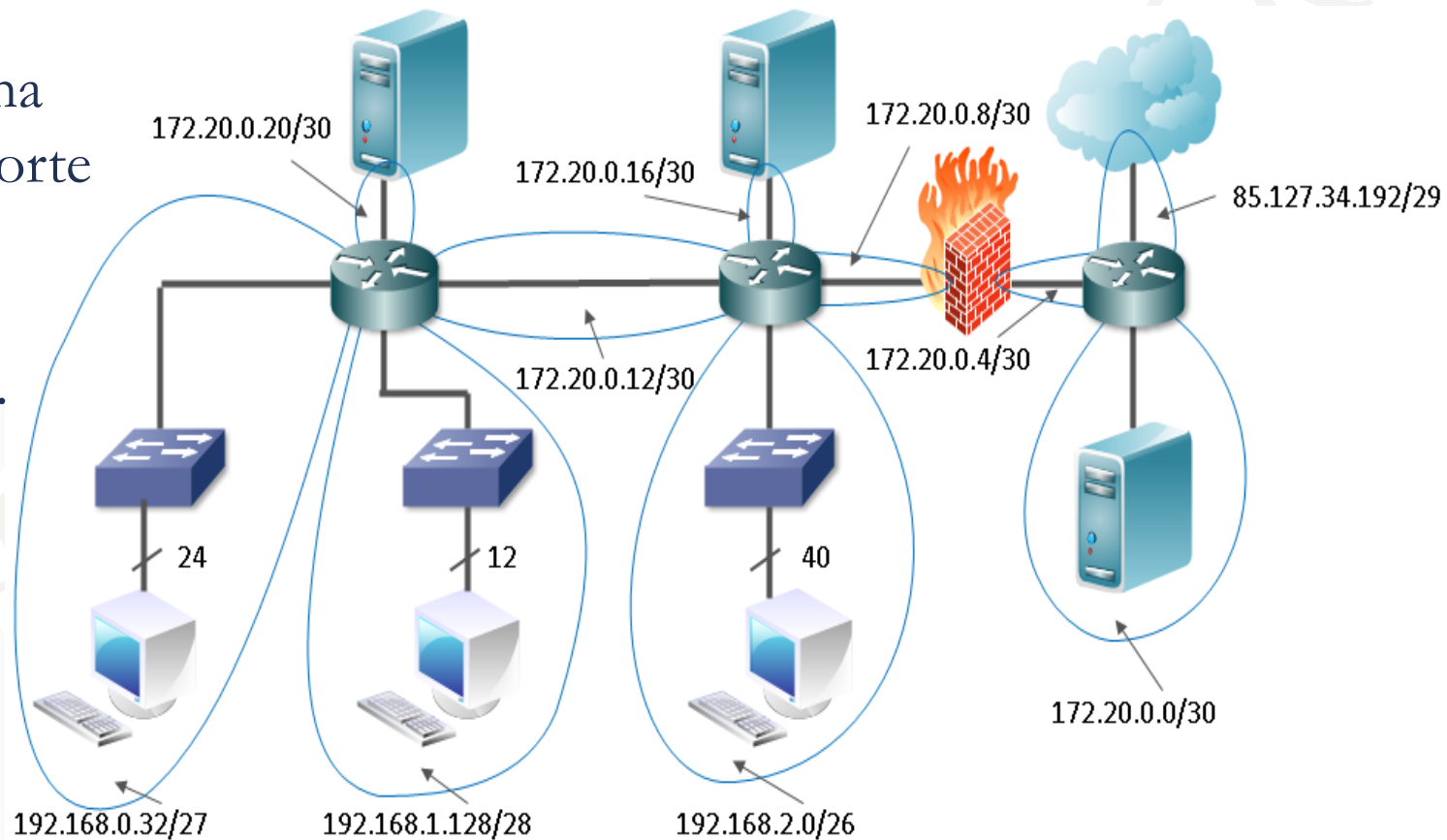
- Chaque réseau sera dans une étendue (range) IP différente (sinon, ce ne serait pas des réseaux différents ...)
- Dans un réseau d'entreprise classique (pas un provider), seul le réseau vers Internet disposera d'adresses IP publiques



# Schémas réseaux

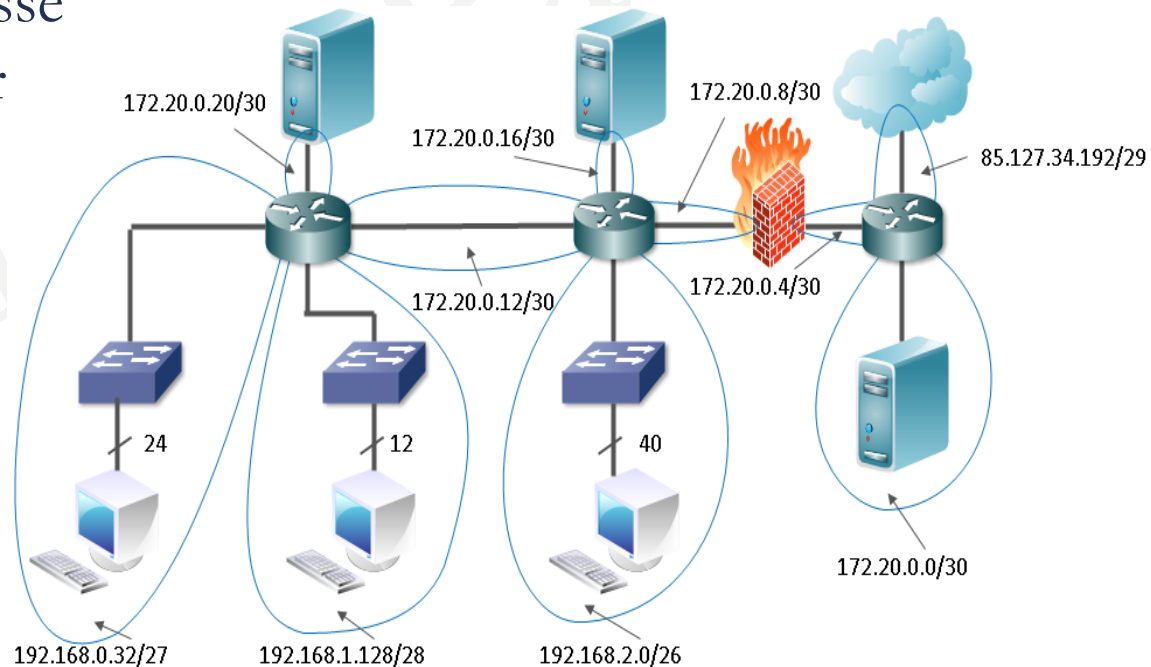
- On pourrait donc obtenir ceci, par exemple:

- Ce schéma ne comporte que des adresses réseau ...



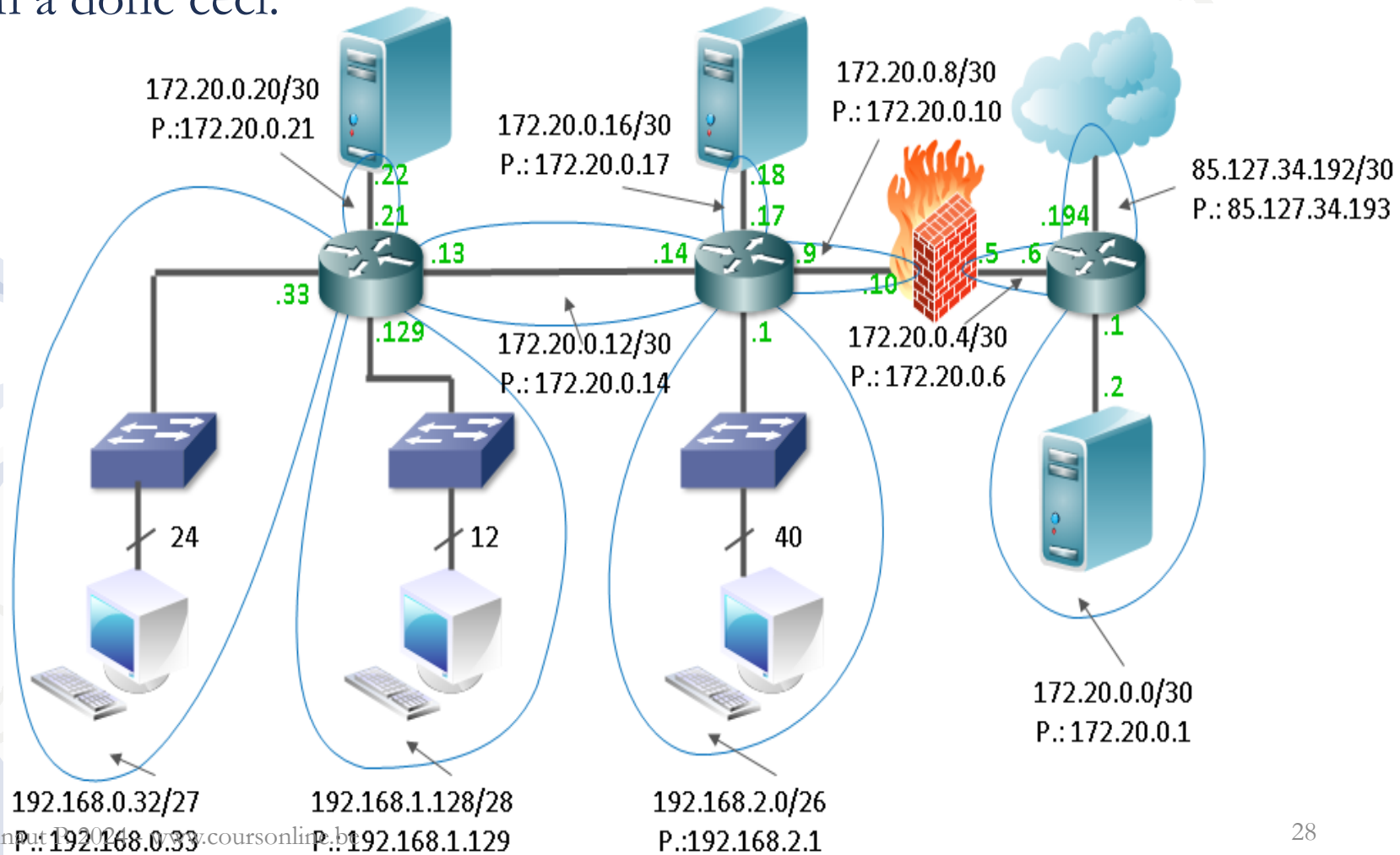
# Schémas réseaux

- On peut imposer une règle d'attribution des IP en notifiant par exemple que le routeur doit prendre la première adresse disponible (ou l'inverse) et dans le cas de deux routeurs, que le routeur à gauche sur le schéma prend la première adresse disponible et le routeur à droite, la dernière



# Schémas réseaux

- On a donc ceci:



# Schémas réseaux

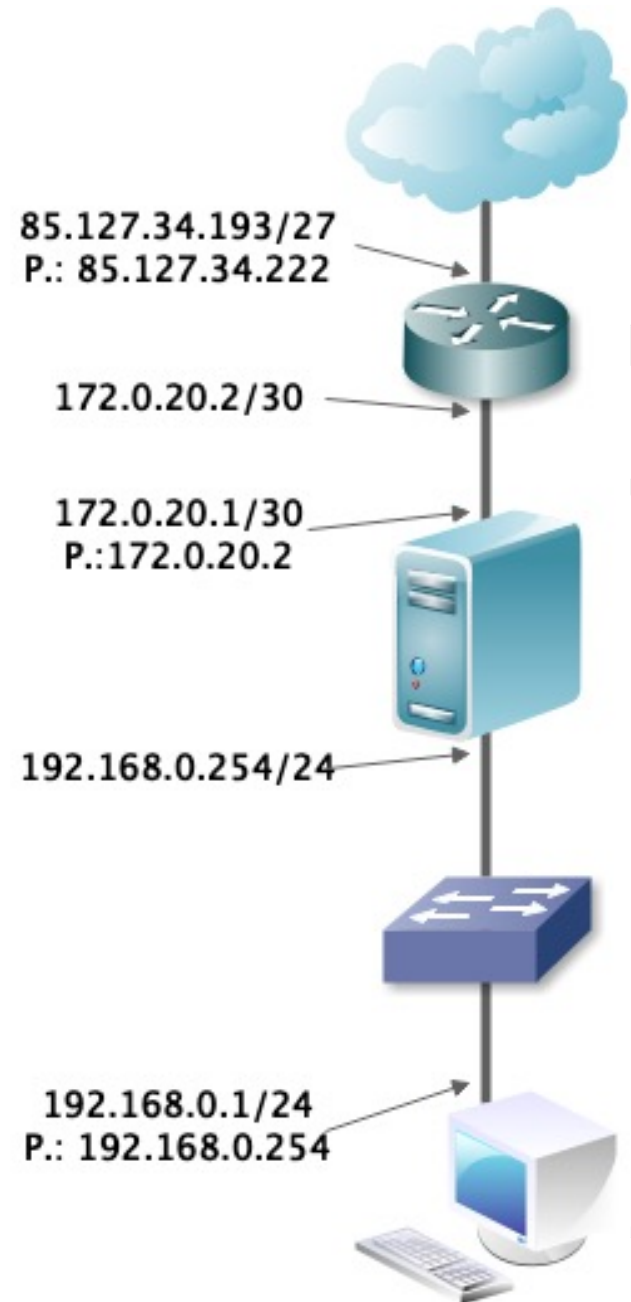
- On voit que chaque interface a une adresse IP (sauf celles des switch qui n'en n'ont pas)
- Elle a aussi un masque de sous-réseau qui n'est pas noté par soucis de visibilité mais qui est indentique à celui de l'adresse réseau
- On remarque l'adresse réseau n'est jamais attribuée à une interface

# Schémas réseaux

- Certaines interfaces ont aussi une adresse de passerelle pour sortir du réseau local (sauf les interfaces de passerelle)
- Par soucis de visibilité, les adresses de passerelle ont été notées au niveau des adresses réseau alors qu'elles devraient être notée au niveau de l'interface correspondante
- Attention, une seule passerelle par équipement  
-> si un équipement a 4 interfaces, une seule d'entre elles se verra attribuer une passerelle

# Schémas réseaux

- Voici un schéma plus simple comportant tous les renseignements IP:



# Schémas réseaux

- Il faut aussi pouvoir traduire un descriptif du réseau en schéma

Exemple: Soit un petit réseau d'entreprise comportant une connexion Internet via modem VDSL, un bureau comptable comprenant 4 postes, un secrétariat avec 2 postes et le bureau du directeur avec 1 poste.

Un serveur de fichiers et une grosse imprimante réseau servent pour les 3 bureaux et tout le monde doit avoir accès à Internet.

Néanmoins, les 3 bureaux doivent être dans des réseaux différents.

Dessinez le schéma réseau et indiquez les paramètres IP.

Vous connaissez l'adresse IP publique attribuée: 212.18.15.37/29 avec 212.18.15.33 comme adresse de passerelle

Ne passez pas à la dia suivante avant d'avoir fait l'exercice ;-)

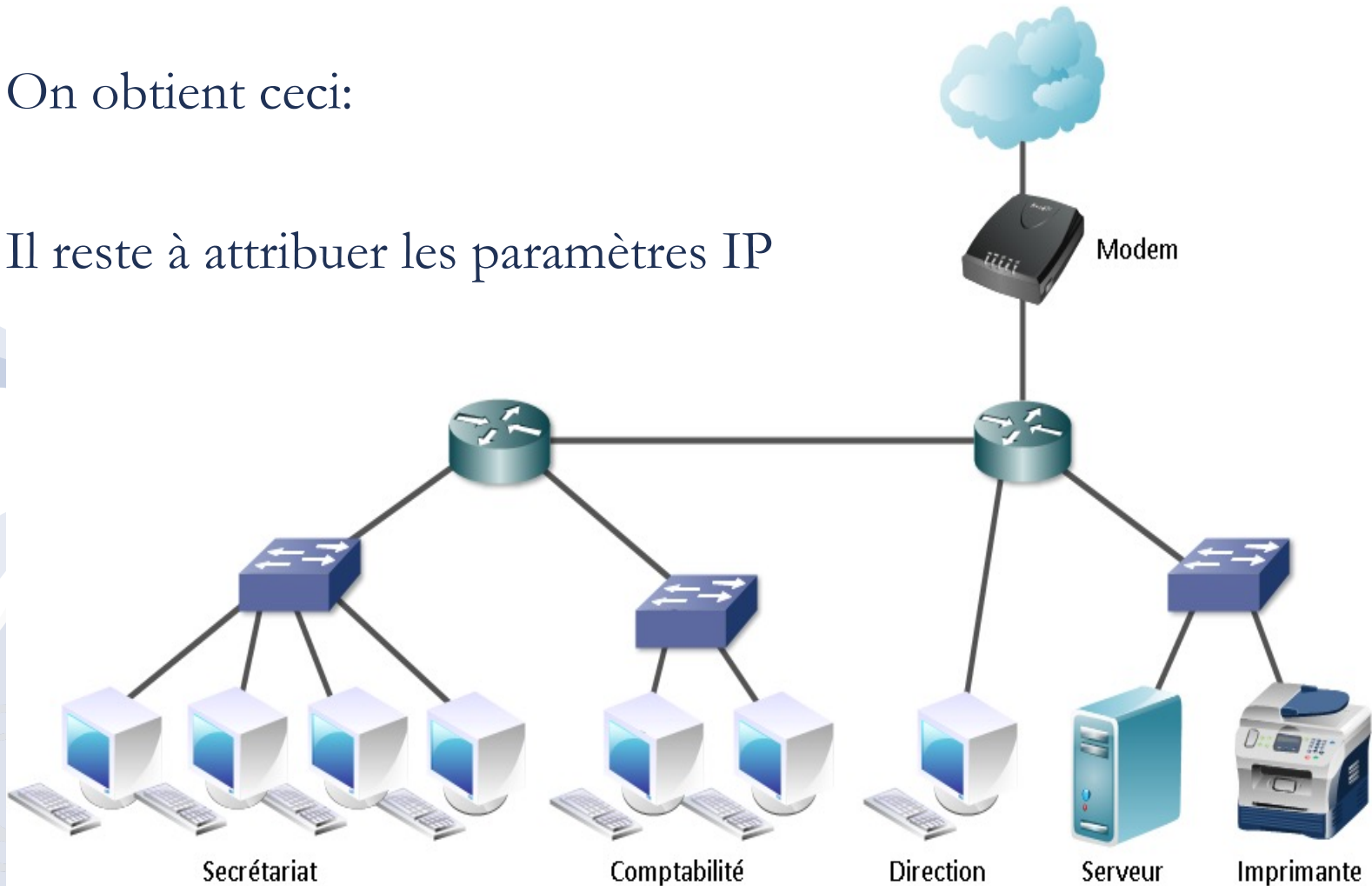


# Shémas réseaux

- Etablissons la liste du matériel nécessaire:
  - Un modem
  - Un serveur de fichiers
  - Une imprimante réseau
  - Sept postes de travail
  - 3 switch (secrétariat, comptabilité, serveur+imprimante)
  - Le nombre de routeurs dépend du modèle. On considère ici des routeurs avec 4 connections RJ-45 (vers 4 réseaux différents). On a 5 réseaux différents (secrétariat, comptabilité, direction, serveur+imprimante, Internet), on aura donc besoin de 2 routeurs

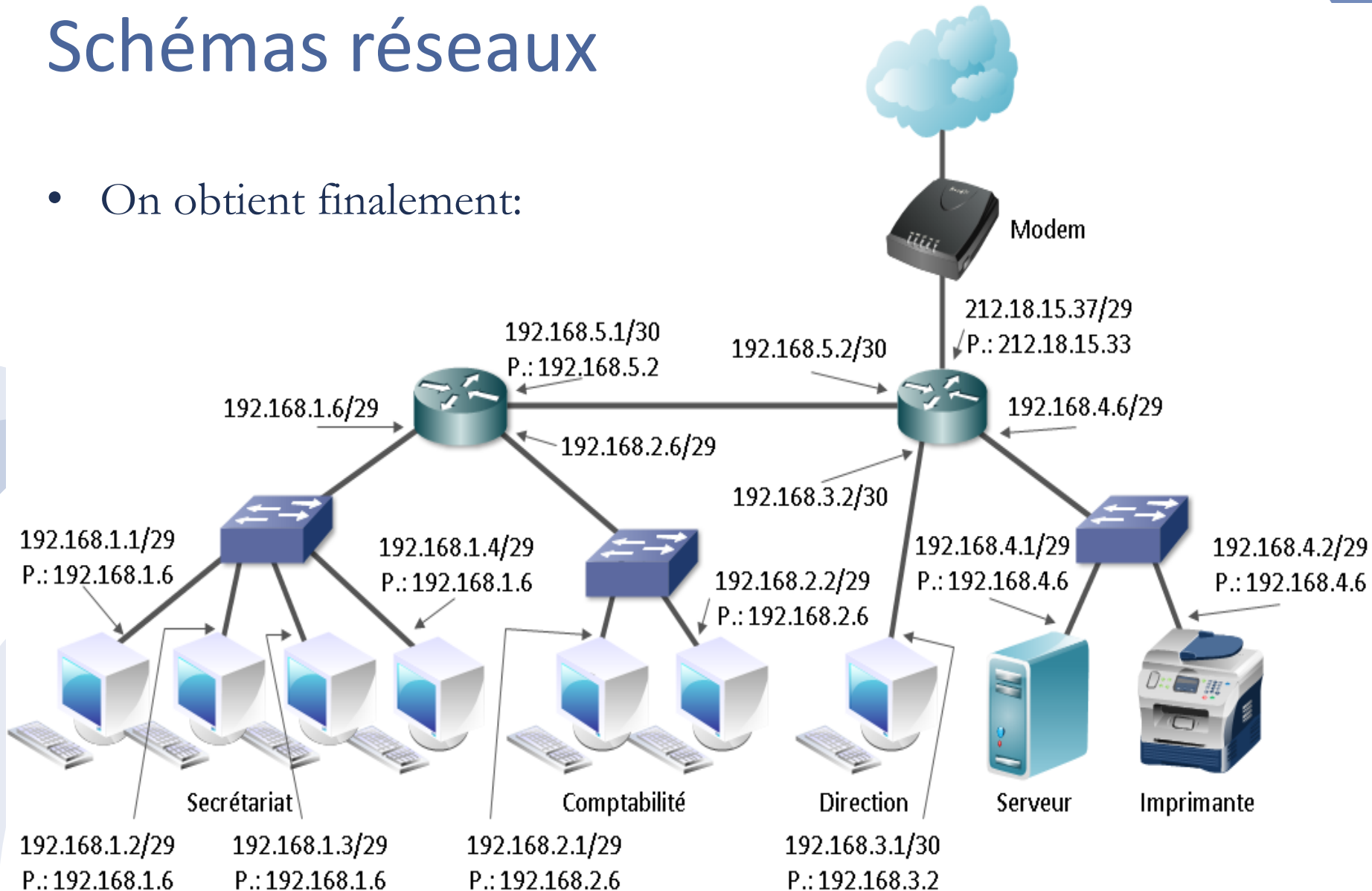
# Schémas réseaux

- On obtient ceci:
- Il reste à attribuer les paramètres IP



# Schémas réseaux

- On obtient finalement:



# Schémas réseaux

- Remarques:
  - seule l'adresse IP publique est définie par avance, mais si elle avait été dynamique, ça fonctionne aussi
  - Pour les adresses des réseaux locaux, on organise son plan d'adressage comme on veut, à condition:
    - d'utiliser des adresses privées
    - de bien séparer les réseaux
    - de ne jamais attribuer une adresse de réseau ou de diffusion
  - L'accès à l'imprimante et au serveur à partir des 3 autres réseaux locaux, se fera en configurant des règles de routage (voir Manip6, Exo2 et Exo3)

# Outils réseaux

- Nous avons déjà vu dans la présentation précédente ipconfig et ping qui sont essentiels
- D'autres outils existent et nous verrons sommairement dans cette présentation, les versions Windows de:
  - traceroute
  - route
  - arp
  - nslookup

Les versions Linux et plus détaillées seront vues dans Manip6

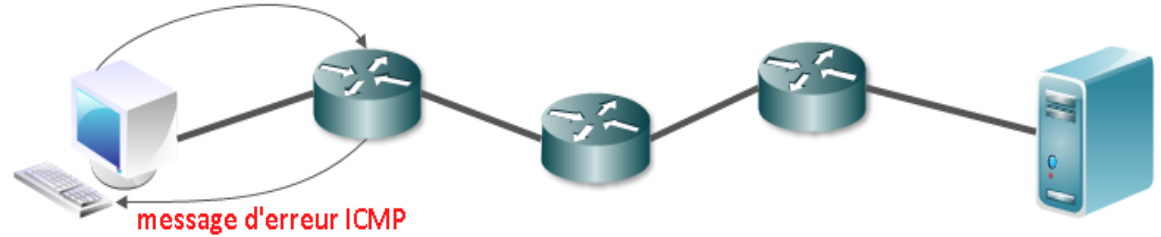
# traceroute/tracert

- Cet utilitaire réseau permet de déterminer la route empruntée par les paquets IP jusqu'à une destination donnée
- Chaque routeur traversé va envoyer un message ICMP avec son adresse IP
- Sous windows, cet utilitaire se nomme **tracert** et est disponible via l'invite de commande

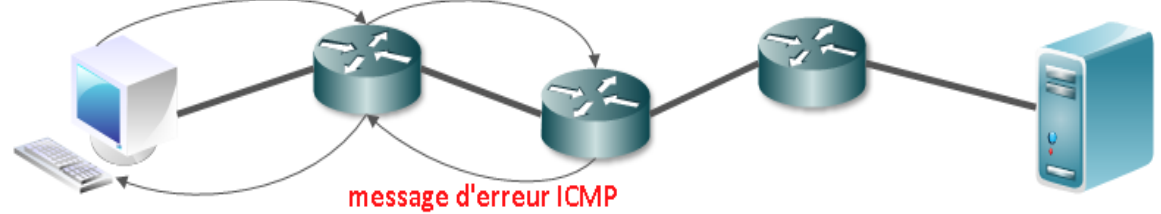
# traceroute

- Principe de fonctionnement:

message ICMP (echo request) dans un paquet avec un TTL=1



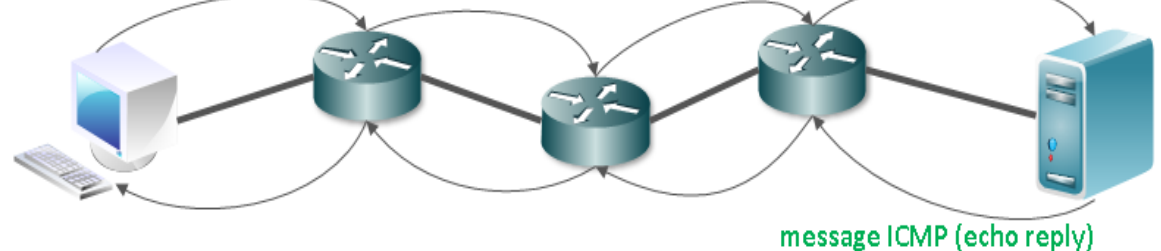
message ICMP (echo request) dans un paquet avec un TTL=2



message ICMP (echo request) dans un paquet avec un TTL=3



message ICMP (echo request) dans un paquet avec un TTL=4



# traceroute/tracert

- Principe de fonctionnement:
  - Chaque paquet IP sur le réseau a une durée de vie (time to live: TTL), qui est diminuée de 1 à chaque passage dans un nœud réseau (PC, serveur, routeur, ...)
  - Si le TTL=0, le paquet est éliminé et un message d'erreur est envoyé à l'émetteur du paquet
  - Dans le cas de tracert, le PC demandeur envoie un ping, avec au départ une durée de vie (time to live: TTL) de paquet IP égale à 1
  - Le prochain nœud reçoit le paquet, décrémente le TTL, constate qu'il vaut 0, jette le paquet, et envoie un message d'erreur au PC en indiquant son adresse IP



# traceroute/tracert

- Principe de fonctionnement:
  - Le PC demandeur renvoie un ping, avec un TTL=2
  - Le prochain nœud reçoit le paquet, décrémente le TTL, consulte sa table de routage et envoie le paquet au nœud suivant
  - Le nœud suivant reçoit le paquet, décrémente le TTL, constate qu'il vaut 0, ...
  - Ainsi de suite, jusqu'à atteindre la destination
  - Attention, parfois certains noeuds ne donnent pas leur adresse IP, elle est remplacée par une \*

# traceroute/tracert

- Utilité:
  - Permet de voir son adresse publique au niveau du modem  
Attention que certains providers comme voo, rassemblent un certain nombre d'abonnés dans un réseau privé pour gagner des adresses publiques  
Dans ce cas, l'adresse délivrée par le modem sera encore privée, ce n'est qu'à la passerelle suivante, que l'adresse sera publique
  - dans le cas où vous n'avez pas accès à Internet, permet de voir si le problème se situe en interne, au niveau du provider ou après

# traceroute/tracert

- Utilité:
  - dans le cas où vous n'avez pas accès à un hôte permet de voir si le problème se situe au niveau du réseau de destination ou avant (par exemple [www.skynet.be](http://www.skynet.be) ne répond pas au ping parce que le trafic ICMP est bloqué, alors que le site est bien accessible via un navigateur)
  - permet (dans une certaine mesure) de voir dans quel pays se situe un hôte (les étendues IP sont différentes selon les pays et les providers mais le résultat peut être faussé dans le cas de multinationales achetant une étendue IP pour tout leur réseau)

# traceroute/tracert

- Exemple:

```
C:\Windows\system32\cmd.exe

C:\Users\Patrick>tracert www.microsoft.com

Détermination de l'itinéraire vers e2847.dsph.akamaiedge.net [2.17.234.96]
avec un maximum de 30 sauts :

  1  <1 ms    <1 ms    <1 ms    192.168.13.1
  2  <1 ms    <1 ms    <1 ms    192.168.7.1
  3  <1 ms    <1 ms    <1 ms    172.20.0.1
  4   6 ms     7 ms     7 ms     10.150.0.1
  5   6 ms     7 ms     7 ms     host-78-129-127-225.dynamic.voo.be [78.129.127.2
25]
  6   8 ms     8 ms     9 ms     host-212-68-211-130.dynamic.voo.be [212.68.211.1
30]
  7  20 ms    10 ms    9 ms     212.3.232.1
  8  14 ms    13 ms    13 ms    ae-228-3604.edge3.London1.Level3.net [4.69.166.1
58]
  9  15 ms    23 ms    15 ms    4.68.63.150
 10  18 ms    16 ms    17 ms    kaia.londra32.lon.seabone.net [89.221.43.65]
 11  22 ms    21 ms    21 ms    ae0-113.cr1.ams2.nl.ipv4.kaiaglobal.com [195.13.
60.57]
 12  22 ms    23 ms    22 ms    akamai.cr1.ams2.nl.ipv4.kaiaglobal.com [193.34.4
8.130]
 13  21 ms    21 ms    21 ms    2.17.234.96

Itinéraire déterminé.

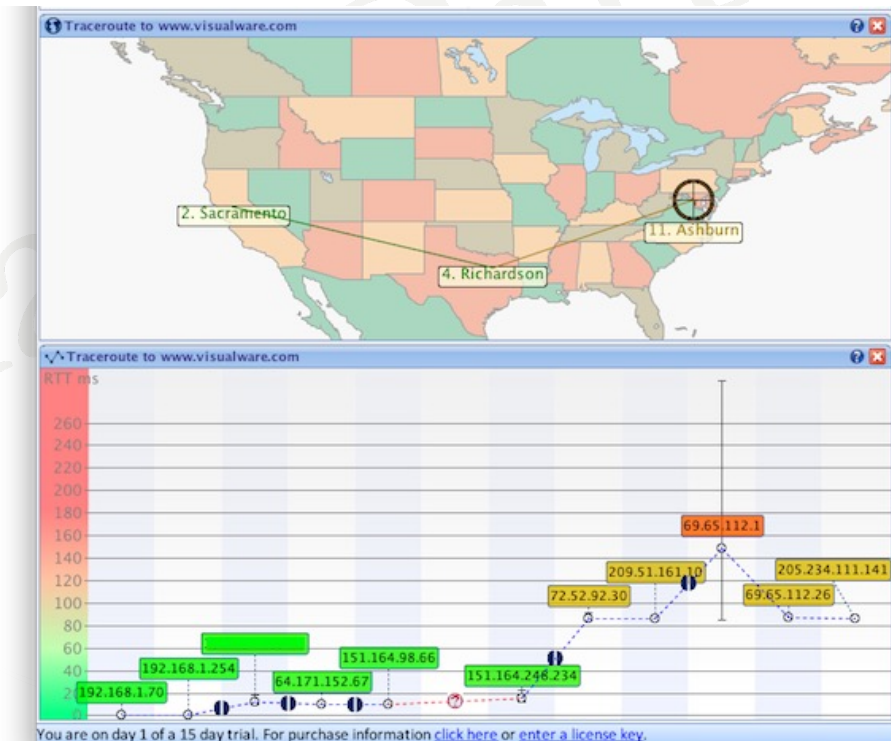
C:\Users\Patrick>_
```

# traceroute/tracert

- Exercices:
  - Faites un **tracert** de [www.helha.be](http://www.helha.be)
  - Faites un **tracert** de [www.cityofperth.wa.gov.au](http://www.cityofperth.wa.gov.au)
  - Faites un **tracert** de [www.nestle.be](http://www.nestle.be)
- Tirez quelques conclusions de vos observations

# VisualRoute

- Des versions graphiques de traceroute/tracert existent
- Malheureusement, elles sont bien souvent payantes
- Exemple: VisualRoute



# route

- La commande **route print** permet de consulter la table de routage de l'ordinateur
- Chaque équipement fonctionnant au niveau de la couche IP (PC, serveurs, routeurs, firewall, ...) possède une table de routage
- Celle-ci permet de voir quel est l'environnement IP de la machine, qui elle va pouvoir contacter et comment

# route

- Exemple:
- Beaucoup d'informations mais pas toutes indispensables

```
C:\Windows\system32\cmd.exe

C:\Users\Patrick>route print

=====
Liste d'Interfaces
16...bc 9a 78 56 34 12 .....Périphérique Bluetooth (réseau personnel)
13...00 1c 42 54 f6 d4 .....Parallels Ethernet Adapter
1 .....Software Loopback Interface 1
17...00 00 00 00 00 00 00 e0 Carte Microsoft ISATAP
11...00 00 00 00 00 00 00 e0 Teredo Tunneling Pseudo-Interface
18...00 00 00 00 00 00 00 e0 Carte Microsoft ISATAP #2
19...00 00 00 00 00 00 00 e0 Carte Microsoft ISATAP #3
=====

IPv4 Table de routage
=====
Itinéraires actifs :
Destination réseau      Masque réseau      Adr. passerelle    Adr. interface  Métrique
0.0.0.0                 0.0.0.0            192.168.13.1       192.168.13.154  10
127.0.0.0               255.0.0.0          On-link            127.0.0.1       306
127.0.0.1               255.255.255.255   On-link            127.0.0.1       306
127.255.255.255         255.255.255.255   On-link            127.0.0.1       306
192.168.13.0            255.255.255.0     On-link            192.168.13.154  266
192.168.13.154         255.255.255.255   On-link            192.168.13.154  266
192.168.13.255         255.255.255.255   On-link            192.168.13.154  266
224.0.0.0               240.0.0.0          On-link            127.0.0.1       306
224.0.0.0               240.0.0.0          On-link            192.168.13.154  266
255.255.255.255         255.255.255.255   On-link            127.0.0.1       306
255.255.255.255         255.255.255.255   On-link            192.168.13.154  266
=====
Itinéraires persistants :
Aucun

IPv6 Table de routage
=====
Itinéraires actifs :
If Metric Network Destination      Gateway
1 306 ::1/128 On-link
1 306 ff00::/8 On-link
=====
Itinéraires persistants :
Aucun

C:\Users\Patrick>
```



# route

- Voyons les informations utiles:

```
IPv4 Table de routage
=====
Itinéraires actifs :
Destination réseau      Masque réseau  Adr. passerelle  Adr. interface  Métrique
1 → 0.0.0.0             0.0.0.0        192.168.13.1     192.168.13.154  10
2 → 127.0.0.0           255.0.0.0      On-link          127.0.0.1       306
   127.0.0.1           255.255.255.255 On-link          127.0.0.1       306
   127.255.255.255     255.255.255.255 On-link          127.0.0.1       306
3 → 192.168.13.0       255.255.255.0  On-link          192.168.13.154  266
   192.168.13.154     255.255.255.255 On-link          192.168.13.154  266
   192.168.13.255     255.255.255.255 On-link          192.168.13.154  266
   224.0.0.0           240.0.0.0      On-link          127.0.0.1       306
   224.0.0.0           240.0.0.0      On-link          192.168.13.154  266
   255.255.255.255     255.255.255.255 On-link          127.0.0.1       306
   255.255.255.255     255.255.255.255 On-link          192.168.13.154  266
=====
```

1: route 8 fois 0 -> tout paquet IP avec une adresse de destination inconnue est dirigé vers la passerelle par défaut 192.168.13.1

2: boucle locale -> adresse de réseau, adresse attribuée, adresse de diffusion

3: réseau local -> adresse de réseau, adresse attribuée, adresse de diffusion

# route

- Remarque:
  - Nous avons ici un PC connecté à un réseau, donc une table de routage très simple
  - Nous aurons plus de lignes et plus de destinations sur un routeur (voir Manip6, Exo2 et Exo3)
- Utilité:
  - Vérifier que la route pour atteindre une destination existe (ou à défaut, la route par défaut 0.0.0.0 0.0.0.0 avec une passerelle par défaut)
- Exercice: vérifiez la table de routage de votre ordinateur

# arp

- Les PC du réseau local discutent entre-eux au niveau Ethernet, c'est à dire grace à leurs adresses MAC
- Mais les utilisateurs, eux, utilisent des adresses IP et généralement ne se préoccupent pas de l'adresse MAC
- ARP (Addres Resolution Protocol) est le protocole qui permet de connaître une adresse MAC en fonction de l'adresse IP et qui est utilisé par un équipement pour connaître l'adresse MAC d'un autre équipement

# arp

- Cet utilitaire permet d'afficher et de modifier les tables de correspondance MAC – IP qui servent de cache pour la résolution ARP
- Faites un **arp -d** pour effacer le cache ARP
- Faites un **ping** vers deux PC du même réseau
- Faites un **arp -a** pour voir le cache ARP
- Observer les résultats sur les PC pingués
- Effacer le cache ARP
- Faites un **ping** vers le 8.8.8.8
- Faites un **arp -a** pour voir le cache ARP
- Tirez quelques conclusions de vos observations

# arp

- Exemple:

```
C:\Users\Patrick>arp -a

Interface : 192.168.13.13 --- 0xd
  Adresse Internet      Adresse physique      Type
  192.168.13.1          00-22-6b-74-e8-ed     dynamique
  192.168.13.247       b8-e8-56-4a-a4-dc     dynamique
  192.168.13.255       ff-ff-ff-ff-ff-ff     statique
  224.0.0.22           01-00-5e-00-00-16     statique
  224.0.0.251          01-00-5e-00-00-fb     statique
  224.0.0.252          01-00-5e-00-00-fc     statique
  239.255.255.250     01-00-5e-7f-ff-fa     statique
  255.255.255.255     ff-ff-ff-ff-ff-ff     statique

C:\Users\Patrick>
```

- Utilité: connaître l'adresse MAC d'un équipement en utilisant ping puis arp, pour bloquer/autoriser cet équipement, effectuer une réservation DHCP, ...

# arp

- La table arp permet de ne pas envoyer un message arp à chaque requête
- Elle est dynamique et ne conserve ses informations qu'un laps de temps limité
- Il est possible d'ajouter une entrée statique valable jusqu'à réinitialisation de la pile TCP/IP en utilisant le commutateur `–s`  
Exemple: **arp -s** 192.168.13.234 00-AC-12-09-F2-5B

Utilité: permet d'attribuer une adresse IP à un équipement basique comme une caméra IP

# nslookup

- Cet utilitaire permet d'interroger un serveur DNS et de détecter d'éventuels problèmes au niveau du DNS
- **nslookup** sans argument indique le serveur DNS primaire configuré sur la machine et ouvre un mode interactif avec une invite de commande pour l'interrogation du serveur DNS

```
C:\Users\Patrick>nslookup
Serveur par défaut : google-public-dns-a.google.com
Address: 8.8.8.8
> _
```

# nslookup

- On peut alors interroger le serveur DNS et de trouver la correspondance entre un nom de domaine et les adresses IP (IPv4 et IPv6 quand elle est présente)
- Exemple:

```
C:\Users\Patrick>nslookup
Serveur par défaut : google-public-dns-a.google.com
Address: 8.8.8.8

> www.coursonline.be
Serveur : google-public-dns-a.google.com
Address: 8.8.8.8

Réponse ne faisant pas autorité :
Nom : www.coursonline.be
Addresses: 2001:41d0:1:1b00:213:186:33:151
           213.186.33.151

> -
```



# nslookup

- On peut faire la recherche dans l'autre sens
- Exemple:

```
C:\Users\Patrick>nslookup
Serveur par défaut : google-public-dns-a.google.com
Address: 8.8.8.8

> 195.238.2.21
Serveur : google-public-dns-a.google.com
Address: 8.8.8.8

Nom : dnspool041.isp.belgacom.be
Address: 195.238.2.21

>
```

# nslookup

- On peut aussi faire la recherche en précisant un autre serveur DNS pour cette recherche
- Exemple avec le serveur DNS de Voo

```
C:\Users\Patrick>nslookup www.coursonline.be 62.197.111.140
Serveur : 62-197-111-140.teledisnet.be
Address: 62.197.111.140

Réponse ne faisant pas autorité :
Nom : www.coursonline.be
Addresses: 2001:41d0:1:1b00:213:186:33:151
           213.186.33.151

C:\Users\Patrick>_
```

# nslookup

- Dans le mode interactif, on peut demander des précisions sur:
  - le serveur DNS associé au domaine: set type=ns
  - le serveur mail associé au domaine: set type=mx
  - l'hôte du réseau (mode par défaut): set type=a

- Exemple:

```
C:\Users\Patrick>nslookup
Serveur par défaut : google-public-dns-a.google.com
Address: 8.8.8.8

> set type=ns
> www.helha.be
Serveur : google-public-dns-a.google.com
Address: 8.8.8.8

helha.be
primary name server = ns1.herb.be
responsible mail addr = ns1.herb.be
serial = 1090
refresh = 10800 (3 hours)
retry = 3600 (1 hour)
expire = 604800 (7 days)
default TTL = 600 (10 mins)
```

# Conclusion

- Dans cette présentation, on a pu augmenter nos savoirs sur:
  - les masques de sous-réseaux
  - les schémas réseaux
  - les utilitaires réseaux Windows
- Des points essentiels pour un professionnel des réseaux ...