



# LES BASES DES RESEAUX TCP/IP

# L'INSTITUT DES RESSOURCES INDUSTRIELLES

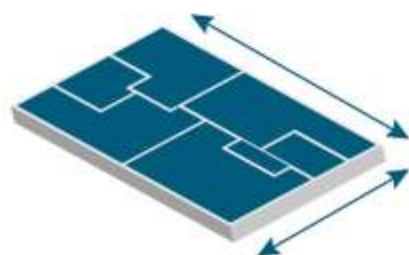
**2** STRUCTURES  
JURIDIQUES



AFPI LYON

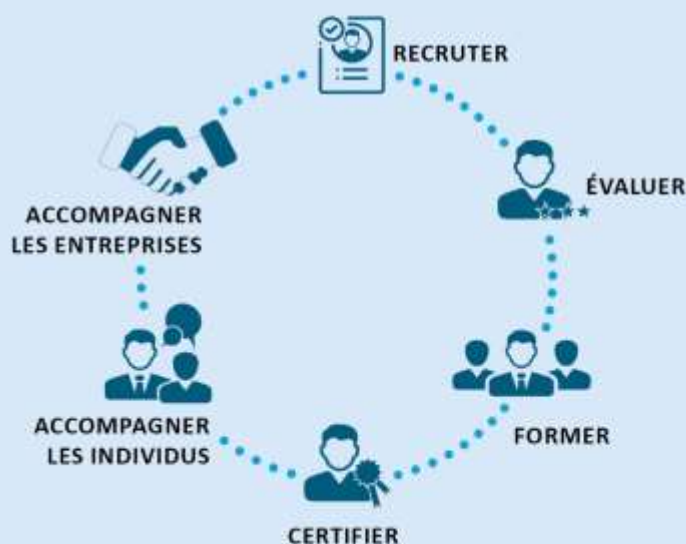


CFAI LYON



**30'000** M<sup>2</sup>  
DE MOYENS ET D'ÉQUIPEMENTS

**6** SOLUTIONS



**180**

SPÉCIALISTES

FORMATEURS  
INGÉNIEURS  
CONSULTANTS  
COLLABORATEURS



**11**

DOMAINES  
D'EXPERTISE



MAINTENANCE  
INDUSTRIELLE



ÉLECTROTECHNIQUE  
ÉLECTRONIQUE  
AUTOMATISMES



CHAUDRONNERIE  
TUYAUTERIE  
SOUDAGE



MECANIQUE  
PRODUCTIVE



RÉSEAUX  
NUMÉRIQUES



GENIE  
ÉNERGETIQUE



ORGANISATION  
ET PERFORMANCE  
INDUSTRIELLE



MANAGEMENT  
RESSOURCES  
HUMAINES



QUALITÉ+ HYGIÈNE  
SÉCURITÉ  
ENVIRONNEMENT



PILOTAGE  
D'ÉQUIPEMENTS  
INDUSTRIELS



ROBOTIQUE  
MECATRONIQUE

## NOTES PERSONNELLES

---



### Modèles OSI et TCP-IP

La communication en réseau fonctionne sur le même principe que celui évoqué dans l'analogie.

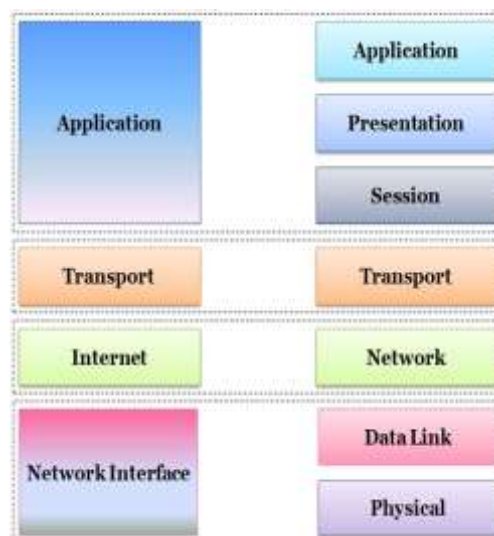
Afin de rendre les logiciels indépendants du matériel, l'ensemble du processus de communication est découpé en couches, chacune :

- ☐ assurant une fonction précise,
- ☐ utilisant un protocole de communication parfaitement codifié.

Entre deux appareils reliés, les couches doivent être les mêmes et pouvoir communiquer en utilisant le même protocole.



© Thierry Schanen - 2013



*TCP/IP*

*OSI*

## Modèles OSI et TCP-IP

L'ISO (International Standards Organisation) a développé le **modèle OSI** (Open Systems Interconnection), modèle théorique qui doit permettre l'interconnexion avec des systèmes hétérogènes.

Il se décompose en **7 couches**, chacune en charge d'un aspect de la communication. C'est un modèle purement théorique qui présente l'intérêt de fixer des directives pour garantir le bon acheminement des données en toute sécurité.

**TCP-IP** est un **modèle fonctionnel à 5 couches** en mesure de communiquer sur Internet. Il s'appuie partiellement sur le modèle OSI. C'est lui que nous allons développer en partie dans ce qui suit.



© Thierry Schanen - 2013

### Un peu de littérature...

Le modèle OSI définit 7 couches au lieu de 4 pour le modèle TCP/IP (les couches physique et liaison de données étant réunifiées en une couche identifiée comme couche physique).

Internet utilise le modèle TCP/IP, le modèle OSI étant un modèle théorique.

Pourquoi a-t-on recours à des notions de couches pour décrire ou implémenter les communications entre machines ?

- Interopérabilité
- Evolutivité (services)



## Modèles OSI et TCP-IP

Le modèle TCP-IP, adapté à la communication entre ordinateurs et sur Internet, utilise 5 couches distinctes :



### Couche application

La **couche application** assure l'interface entre l'utilisateur et le réseau :

- ☐ courrier électronique,
- ☐ transfert de fichiers,
- ☐ appel de procédures distantes,
- ☐ affichage de pages web,
- ☐ ...

Les données sont converties, cryptées, compressées... préparées pour le voyage sur le réseau.

## Présentation du protocole TCP/IP

On appelle protocole TCP/IP l'ensemble des règles qui permet à des machines (modem, ordinateur, routeur,) de communiquer entre elles sur un réseau informatique. Ces règles (RFC Requests For Comments) sont définies par des organismes internationaux, comme IETF, IEEE... et tous les constructeurs doivent les respecter pour que leurs matériels soient interopérables.

Certains protocoles sont spécialisés dans le transfert des fichiers (FTP par exemple), d'autres dans la consultation de pages web (http) ou pour gérer l'état des transmissions et des erreurs (ICMP).

Sur Internet, l'ensemble des protocoles utilisés porte le nom de suite TCP/IP, elle contient entre-autres les protocoles suivants : http, FTP, ARP, ICMP, IP, TCP, UDP, SMTP, IMAP...

## Principe de fonctionnement de TCP/IP

TCP/IP est à la fois une architecture réseau, mais aussi l'acronyme de 2 protocoles réseau liés :

- [TCP \(Transmission Control Protocol\)](#) : protocole de transport
- [IP \(Internet Protocol\)](#) : protocole réseau (adressage)

L'architecture réseau TCP/IP se décompose en 4 couches dans laquelle les protocoles TCP et IP jouent un rôle important.

### Couche 4 : Application

C'est ici que l'on trouve les protocoles de communication entre les clients et les serveurs. ([HTTP](#), [FTP](#), [POP](#) et [SMTP](#))

### Couche 3 : Transport

On retrouve ici les protocoles de transport des données. Les plus utilisés sont : [Protocole TCP](#) - [Protocole UDP](#)

### Couche 2 : Réseau

Dans cette couche on trouve principalement deux protocoles. Le [protocole IP](#) qui permet le routage des informations entre réseaux (Utilisation de l'[adresse IP](#)) et le [protocole ICMP](#) qui permet le contrôle d'erreur et de signalisation.

### Couche 1 : Accès réseau

C'est la couche de plus bas niveau sur le réseau. Cette couche contient des protocoles qui gèrent l'acheminement des informations entre émetteur et destinataire. On retrouve dans cette couche [les adresses MAC](#) ainsi que le [protocole Ethernet](#) et le protocole [WiFi \(802.11\)](#)



## TCP UDP Quelle différence ?

- **TCP protocole orienté connexion** : opère un contrôle des transmissions. La machine réceptrice envoie un accusé de réception pour chaque donnée reçue avec une vérification de son intégrité. La machine émettrice a la garantie de la validité des données qu'elle envoie, c'est l'équivalent postal de la lettre recommandée avec accusé de réception.
- **UDP protocole non orienté connexion** : la machine émettrice envoie des données sans prévenir la machine réceptrice, et cette dernière ne communique pas à l'émettrice la réception ou la validité des données éventuellement reçues.

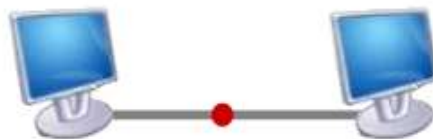
### En résumé

- TCP est plus fiable que UDP car le protocole garantit que les paquets sont bien arrivés ;
- TCP est plus courant que UDP, ce qui lui permet donc de fonctionner dans la plupart des situations, y compris à travers des firewalls, qui laissent par défaut un certain nombre de ports TCP ouverts (80, 443.etc...).
- UDP est plus rapide que TCP, puisque le protocole ne nécessite pas d'aller-retour pour vérifier la bonne livraison des paquets. Ce protocole est privilégié quand un flux peut supporter une dégradation temporaire du service (téléphonie sur IP, Vidéo streaming).

**Les protocoles orientés connexion** : opérant un contrôle de transmission des données pendant une communication établie entre deux machines. La machine réceptrice envoie des accusés de réception lors de la communication.



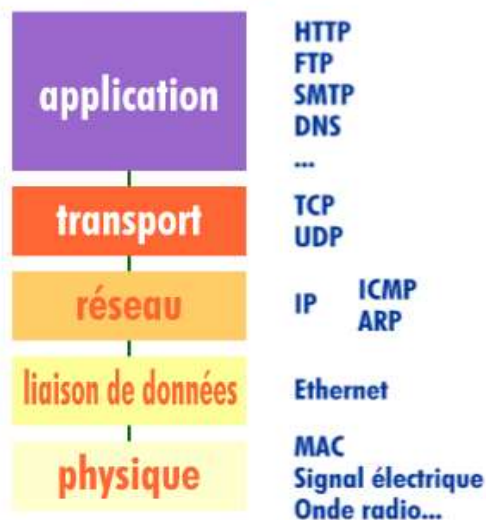
Les **protocoles non orientés connexion** : mode de communication dans lequel la machine émettrice envoie des données sans prévenir la machine réceptrice et la machine réceptrice reçoit les données sans envoyer d'avis de réception à la première.



### Protocoles

Sur Internet, les protocoles utilisés font partie d'une **suite de protocoles**, c'est-à-dire un ensemble de protocoles reliés entre-eux.

Cette suite de protocole s'appelle **TCP-IP**.



*Quelques protocoles...*

## Numérotation hexadécimale

Équivalents décimaux et binaires des caractères hexadécimaux 0 à F

Décimal	Binaire	Hexadécimal
0	0000	0
1	0001	1
2	0010	2
3	0011	3
4	0100	4
5	0101	5
6	0110	6
7	0111	7
8	1000	8
9	1001	9
10	1010	A
11	1011	B
12	1100	C
13	1101	D
14	1110	E
15	1111	F

### Adressage physique des stations

L'adressage physique permet l'identification fiable de chaque poste connecté à un réseau mais pose des difficultés pour localiser l'appareil car la distribution des adresses MAC est anarchique.



*Aucune méthode fiable et systématique ne permet de retrouver une adresse MAC donnée à l'intérieur d'un vaste réseau.*

Il convient de mettre en place un système d'identification permettant de localiser un poste avec efficacité. C'est le rôle de l'**adressage logique IP**.

### Adressage physique des stations

L'information à transmettre est mise en forme, les trames sont constituées en respectant les protocoles définis, les bits circulent dans les fils ou par les ondes... Et nous venons de voir que chaque trame doit contenir l'**adresse du destinataire** afin d'être convenablement acheminée...

Mais comment trouver le destinataire ?

Chaque appareil connecté au réseau est identifié par un code ou une **adresse unique**. Cet **identifiant unique** est déterminé à la fabrication de la carte réseau.

Sur un réseau de type Ethernet, cet identifiant s'appelle l'**adresse MAC** (Media Access Control). Il est affecté par le fabricant de la carte réseau et se présente sous forme d'une suite de 6 octets.

Exemple d'adresse physique (notée en hexadécimal) :





## Adresse IP

Cette adresse logique est nommée adresse **IP** (Internet Protocol). Actuellement, la majorité des systèmes utilisent encore une ancienne version : **IP v4**. Une nouvelle version est en cours de déploiement : **IP v6**.

Une adresse IP v4 est un nombre codé sur 32 bits présenté sous forme d'un groupement de 4 octets :

a.b.c.d

*Chaque lettre représente un octet (un nombre entre 0 et 255 ou \$00 et \$FF).*



Dans un même réseau, il ne peut y avoir deux postes ayant la même adresse IP.

### Décodage d'une adresse IP - exemple

Les 4 octets de l'adresse IP permettent de désigner le réseau et l'ordinateur ou le périphérique à l'intérieur de ce réseau.

192.168.20.2



*Les octets les plus à gauche  
désignent le réseau :  
il s'agit du **net-ID**.*

*Les octets les plus à droite  
désignent l'appareil connecté :  
il s'agit de l'**host-ID**.*

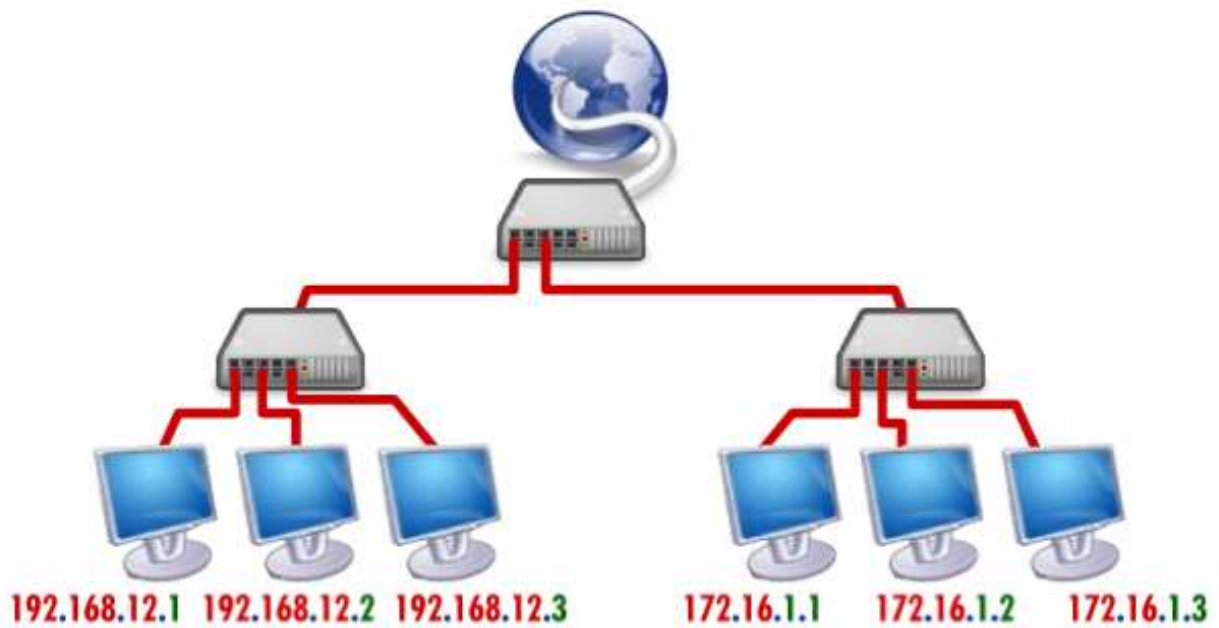
Le net-ID peut être constitué de 1, 2 ou 3 octets selon l'étendue du réseau.

Selon la taille du net-ID, l'host-ID sera constitué de 3, 2 ou 1 octets.



## Décodage d'une adresse IP - exemple

Deux petits réseaux sont connectés à Internet. Chacun a son propre net-ID.



## Classes de réseaux

L'ensemble des adresses IP est réparti en **classes**, selon le nombre d'octets qui représentent le réseau (taille du net-ID).

### 9.1- Classe A

- ☐ Le net-ID ne comporte qu'un **seul octet**.
- ☐ Le bit de poids fort du premier octet est à zéro, ce qui signifie qu'il y a  $2^7$  possibilités de réseaux (00000000 à 01111111).
- ☐ Le réseau 0 (00000000) n'existe pas et le nombre 127 est réservé pour désigner la machine elle-même.

Les réseaux disponibles en classe A sont donc les réseaux allant de **1** à **126**.

Les trois octets de droite représentent les ordinateurs du réseaux. Le réseau peut donc contenir :

$$2^{24}-2 = 16\,777\,214 \text{ ordinateurs}$$

Une adresse IP de classe A, en binaire, ressemble à ceci :

0xxxxxxx.xxxxxxxx.xxxxxxxx.xxxxxxxx

Réseau

Ordinateur

## Classes de réseaux

### 9.2- Classe B

- ❑ Le net-ID est écrit sur deux octets.
- ❑ Les deux premiers bits du premier octet valent 1 et 0, ce qui signifie qu'il y a  $2^{14}$  possibilités de réseaux (10000000.00000000 à 10111111.11111111) soit 16 384 réseaux.

Les réseaux disponibles en classe B sont donc les réseaux allant de 128.0 à 191.255.

Les deux octets de droite représentent les ordinateurs du réseaux. Le réseau peut donc contenir :

$$2^{16}-2 = 65\,534 \text{ ordinateurs}$$

Une adresse IP de classe B, en binaire, ressemble à ceci :

10xxxxxx.xxxxxxxx.xxxxxxxx.xxxxxxxx

Réseau

Ordinateur

## Classes de réseaux

### 9.3- Classe C

- ❑ Le net-ID est écrit sur **trois octets**.
- ❑ Les trois premiers bits du premier octet valent 1, 1 et 0, ce qui signifie qu'il y a  $2^{21}$  possibilités de réseaux (**110**00000.00000000.00000000 à **110**11111.11111111.11111111) soit 2 097 152 réseaux.

Les réseaux disponibles en classe C sont donc les réseaux allant de **192.0.0** à **223.255.255**.

Les deux octets de droite représentent les ordinateurs du réseaux. Le réseau peut donc contenir :

$$2^8 - 2 = 254 \text{ ordinateurs}$$

Une adresse IP de classe C, en binaire, ressemble à ceci :

**110**xxxxx.xxxxxxxx.xxxxxxxx.xxxxxxxx

Réseau

Ordinateur



## Adresses IP réservées

Deux appareils connectés à un même réseau ne peuvent avoir la même adresse IP. Si un réseau est connecté à Internet, l'attribution des adresses IP des machines reliées à l'extérieur ne peut pas se faire sans prendre en compte toutes les adresses déjà occupées par des postes reliés à Internet.

Il est possible d'obtenir auprès de l'**ICANN** une adresse libre fixe.

Tous les autres ordinateurs du réseau ayant cependant besoin d'une adresse IP, il a été défini une série d'adresses IP à utiliser dans les réseaux locaux qui n'interféreront pas avec les adresses réservées au WEB.

Adresses disponibles pour les réseaux privés :

**!** *Aucune autre adresse ne doit être utilisée dès lors que l'appareil est connecté à Internet.*

### Classe A

- ☐ net-ID : **10**
- ☐ host-ID de 0.0.1 à 255.255.254

### Classe B

- ☐ net-ID : **172.16 à 172.31**
- ☐ host-ID de 0.1 à 255.254

### Classe C

- ☐ net-ID : **192.168.0 à 192.168.255**
- ☐ host-ID de 1 à 254

© Thierry Schanen - 2011

## Classes de réseaux

### 9.4- Classes D et E

Les adresses de classes D et E sont réservées à des applications de maintenance et d'expérimentation.

Les réseaux disponibles en classe D sont donc les réseaux allant de **224.0.0** à **239.255.255**.

Les réseaux disponibles en classe E sont donc les réseaux allant de **240.0.0** à **254.255.255**.

## Les plages des adresses privées

Classe	Début de la plage	Fin de la plage	Nombre de réseaux
A	10.0.0.0	10.255.255.254	1
B	172.16.0.0	172.31.255.254	16
C	192.168.0.0	192.168.255.0	255

Les adresses privées constituent la substantifique moelle des réseaux locaux d'entreprises et domestiques et c'est à l'une ou l'autre de ces plages d'adresses que vous aurez affaire dans vos tâches de configuration d'équipements, ou que vous aurez à faire dans vos tâches d'administration. Ces adresses ont la portée d'un réseau local et ne devraient pas circuler sur le réseau Internet.

Mais...Hey ! Et l'adresse abracadabrante **127.0.0.1**, elle se situe où ? hein !

**127.0.0.1** est l'adresse de la machine locale (mon PC). Quoi ? Comment ? Pour me joindre les autres machines doivent-elles utiliser cette adresse ? **NON.**

Je suppose que tous les autres PC disposent de cette adresse également ?

**VOUS AVEZ VU JUSTE.**

A quoi elle peut bien servir, nom de Dieu ?

C'est l'adresse de rebouclage ou Loop back (pour faire savant), voilà qui est bien claire. D'ailleurs, pour éviter de la saisir, il suffit d'écrire **localhost**. Voyons :

```
C:\Users\tonde>localhost
'localhost' n'est pas reconnu en tant que commande interne
ou externe, un programme exécutable ou un fichier de commandes.

C:\Users\tonde>ping localhost

Envoi d'une requête 'ping' sur porttonde19.afpm.fr [127.0.0.1] avec 32 octets de données :
Réponse de 127.0.0.1 : octets=32 temps<1ms TTL=128
Réponse de 127.0.0.1 : octets=32 temps<1ms TTL=128
Réponse de 127.0.0.1 : octets=32 temps<1ms TTL=128
Réponse de 127.0.0.1 : octets=32 temps<1ms TTL=128

Statistiques Ping pour 127.0.0.1:
    Paquets : envoyés = 4, reçus = 4, perdus = 0 (perte 0%),
Durée approximative des boucles en millisecondes :
    Minimum = 0ms, Maximum = 0ms, Moyenne = 0ms
```

Waa...ouh ! **0 milliseconde** pour interroger 127.0.0.1, c'est plus rapide que la vitesse de la lumière. Ce n'est plus du réseau, ...c'est de la métaphysique.

C'est possible ça Monsieur ?

**Qui ne dit mot consent.**



Monsieur ne se mouille pas !

Et ce **Ping**, qu'a-t-il à voir là-dedans?

**On le décortiquera un peu plus loin.**

En dehors des classes d'adresses privées, ces notions de classe sont obsolètes et font désormais partie du passé de l'ancien régime! Elles sont désuètes ! Elles ont contribué à la gabegie, à une utilisation non optimale des adresses IP.

## **Les classes sont mortes, vive le CIDR :**

***CIDR = Classless Inter Domain Routing « routage sans classes entre domaines ».***

***On ne peut pas plus claire !***

***N'est-ce pas ?***

***Si non, attendre 4 pages plus bas pour en discuter.***

## Masque de sous réseau

Lors du **roulage des données** (leur acheminement vers le bon ordinateur à travers les ramifications du réseau), il est nécessaire d'identifier le net-ID à l'intérieur de l'adresse IP.

A cet effet, on applique un **masque de sous réseau** qui se présente comme une adresse IP. Il comprend, dans sa notation binaire, des 0 au niveau des bits de l'host-ID et des 1 au niveau de ceux du net-ID.

Par application d'un **ET logique** entre l'adresse IP et le masque, on obtient le net-ID :

*Exemple :*

adresse IP : 10.208.123.12 (il s'agit d'une adresse de classe A)

soit en binaire : 00001010.11010000.01111011.00001100

masque : 11111111.00000000.00000000.00000000

résultat après masquage en ET : 00001010.00000000.00000000.00000000

### Classe A

☐ masque : 255.0.0.0

### Classe B

☐ masque : 255.255.0.0

### Classe C

☐ masque : 255.255.255.0

## Deux adresses pour le prix d'une : le masque de sous-réseau

Une adresse IP est en fait l'adresse **du réseau ET de la machine**.

Plus exactement, une partie de l'adresse représentera l'adresse du réseau, et l'autre partie l'adresse de la machine.

Mais d'abord, comment elle s'écrit cette adresse ?

Une adresse IP est codée sur 32 bits (soit 4 octets, car un octet vaut 8 bits).

On en déduit au passage que la plus petite adresse IP est: 0.0.0.0 (quand tous les bits de l'adresse sont à 0) alors que la plus grande vaut : 255.255.255.255 (quand tous les bits sont à 1).

Et ces deux informations, adresse IP et masque, seront **inséparables**.

C'est le masque qui va indiquer quelle est la partie réseau de l'adresse, et quelle est la partie machine.

**Définition : Les bits à 1 dans le masque représentent la partie réseau de l'adresse IP.**

On en déduit que les bits à 0 représentent la partie machine de l'adresse.

Prenons un exemple : on associe l'adresse IP 192.168.0.1 au masque 255.255.0.0. Écrivons maintenant ces deux adresses en binaire pour y voir plus clair :

255.255.0.0                      =>                      11111111.11111111.00000000.00000000  
192.168.0.1 => 11000000.10101000.00000000.00000001

Le résultat ET bit à bit est :

11000000.10101000.**00000000.00000000**

**192.168 = adresse réseau et 0.0 = adresses-machines**

Donc la partie réseau de l'adresse est 192.168, et la partie machine peut varier de 0.0 à 255.255.

L'association **192.168.0.1 /255.255.0.0** représente la machine « 01 » sur ce réseau qui s'étende de **192.168.0.0 à 192.168.255.255**

## La contiguïté des bits

Dans un masque en binaire, il doit y avoir les 1 à gauche et les 0 à droite. On ne peut pas mélanger les 1 et les 0.

Par exemple, ce masque est **correct** : 11111111.11111000.00000000.00000000

Mais celui-ci est **incorrect** : 11111111.11100011.00000000.00000000.

Ainsi, on retrouvera toujours les mêmes valeurs pour les octets d'un masque, qui sont les suivantes :

00000000	> 0
10000000	> 128
11000000	> 192
11100000	> 224
11110000	> 240
11111000	> 248
11111100	> 252
11111110	> 254
11111111	> 255

Donc ce masque est **correct**: 255.255.128.0.

Et ce masque est **incorrect**: 255.255.173.0.

Et ce masque est encore **incorrect**: 255.128.255.0 (car il mélange des 0 et des 1).

## Exemple

Soit l'association adresse 192.168.0.1 et masque 255.255.240.0.

255.255.240.0	> 11111111.11111111.11110000.00000000
192.168.0.1	> 11000000.10101000.00000000.00000001

On vérifie que la partie réseau est égale à 11000000.10101000.0000

Sur ce réseau, les adresses vont être les suivantes :

11000000.10101000.00000000.00000000	->	192.168.0.0
11000000.10101000.00000000.00000001	->	192.168.0.1
11000000.10101000.00000000.00000010	->	192.168.0.2

11000000.10101000.00000000.00000011	->	192.168.0.3
11000000.10101000.00000000.00000100	->	192.168.0.4
11000000.10101000.00000000.00000101	->	192.168.0.5
...		
11000000.10101000.00001111.11111110	->	192.168.15.254
11000000.10101000.00001111.11111111	->	192.168.15.255

## Ecriture CIDR

Le masque 255.255.255.0 contient 24 bits à 1 on le notera /24

Au lieu d'écrire 192.168.0.1/255.255.255.0, on écrira 192.168.0.1/24. /24 se nomme le préfixe

On écrira donc /20 au lieu de 255.255.240.0

## Adresse de réseau, adresse de broadcast

Parmi la plage d'adresses définie par une adresse IP et un masque, deux adresses sont particulières, la première et la dernière.

La première adresse du réseau est celle dont tous les bits de la partie machine sont à 0. La dernière adresse du réseau est celle dont tous les bits de la partie machine sont à 1.

Dans l'exemple ci-dessus, 192.168.0.0 est l'adresse du réseau local (vu des autres sous réseaux). Cette adresse ne sera jamais utilisée par une application.



La dernière adresse 192.168.15.255 (tous les bits du host-id à 1) désigne toutes les « machines » du sous-réseau, ~~on lit et on entend souvent dire que c'est l'adresse de diffusion générale et que tout message envoyé à cette adresse serait reçu par tous les hôtes du sous-réseau!~~ Cela est vrai pour certains OS, pas pour MS Windows.

Heureusement que c'est faux, sinon un hacker débutant envoyant une avalanche de « ping 192.168.15.255 » à intervalles réguliers saturerait le réseau local le rendant inapte à assurer sa fonction.

```

for( i=0 ; i < 1000000 ; i++ ) do
    Ping 192.168.15.255 ;
    Attendre 1ms ;
End for

```

L'adresse de diffusion générale ou de broadcast est symbolique, elle n'a pas vocation à circuler sur le réseau dans les trames Ethernet et si tel est le cas, elle est tout simplement ignorée.

### Exercice

- ✓ Déterminez l'adresse broadcast de votre PC.
- ✓ Faites un ping à destination de cette adresse.
- ✓ Avez-vous fait mouche ?

MS Windows n'a pas été conçu comme un terrain d'expérimentation tel que Linux ou openBSD...

*Remarquez par la même occasion que dans un réseau ayant 16 adresses disponibles, seules 14 adresses seront utilisables par les machines du réseau, car la première et la dernière seront réservées pour le réseau et le broadcast. Et cela est vrai pour tout réseau. Pour chaque réseau, il y a deux adresses non utilisables pour les machines !*

Nous savons donc maintenant, à partir d'une adresse et du masque associé :

- Déterminer la première et la dernière adresse de la plage ;
- Connaître le nombre d'adresses de cette plage.

## EXERCICES

**Exercice1** : Pour chaque ligne ci-dessous, dire s'il s'agit d'une adresse machine, de réseau ou de diffusion

- a) 192.168.0.15/255.255.255.0
- b) 192.168.1.0/255.255.255.0
- c) 192.168.1.0/255.255.254.0
- d) 10.8.65.29/255.255.255.224
- e) 10.8.65.31/255.255.255.224
- f) 10.0.0.255/255.255.254.0

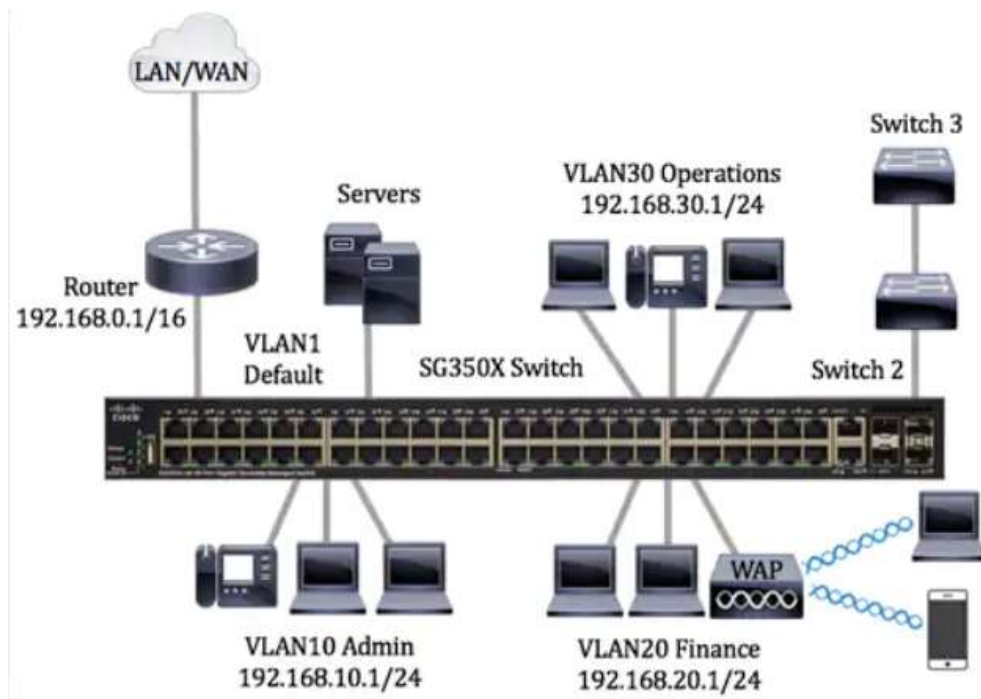


- g) Combien d'adresses peut rassembler le réseau IPv4 associé à l'adresse 10.1.2.0/31 ?
- h) Les machines 10.255.2.250/25 et 10.255.2.120/25 appartiennent au même réseau ?
- i) Une machine possède l'adresse 10.255.2.140/25. Quel est son masque de sous réseau
- j) Quelle taille de préfixe doit-on demander à son opérateur si l'on souhaite pouvoir avoir au moins 847 machines dans son réseau ?
- k) Pour créer un réseau pouvant contenir 63 machines, j'ai besoin du masque (entourer la bonne réponse)  
255.255.255.192  
  
255.255.255.254  
  
255.255.255.128  
  
255.255.255.224
- l) Quelle précaution faut-il prendre si on doit configurer un équipement-IP à l'aide d'un PC en P2P (pair à pair ou peer-to-peer) ?

<b>Mais en quoi le CIDR contribue à mieux utiliser les adresses IP ?</b>
--

La question sera discutée dans le paragraphe consacré au NAT et Port forwarding

## Exercice2 : VLAN ...What's up ?



### VLAN : Virtual Local Area Network

Qu'est-ce qu'un Vlan ? En quoi contribue-t-il à la sécurité et la flexibilité d'un réseau local ?

Une entreprise possède la plage 10.0.0.0/16. Elle compte 1000 techniciens, 200 commerciaux et 20 directeurs.

Votre mission est de découper cette plage énorme de 65536 adresses en 3 sous réseaux qui ne se chevauchent pas. Les hôtes d'un groupe ne peuvent pas communiquer directement avec un autre, ils doivent restés groupés groupés.

**1<sup>ère</sup> solution :** commencez par les techniciens, puis les commerciaux et enfin les directeurs. Pour chaque groupe, écrivez la première et la dernière adresse de sa plage d'adresses et exprimez-la en notation CIDR.

**2ème solution : *Pourquoi faire simple quand on peut faire compliqué !***

Au commencement seuls les directeurs avaient des postes connectés au réseau local ; Désormais on décide de doter les deux autres populations de postes connectés à des deux sous réseaux différents. Commencez le découpage par les directeurs, puis les commerciaux et enfin les techniciens avec les mêmes critères de réponse que précédemment.



Après la discussion qui vient de s'achever, cochez les 'bonnes' cases ci-dessous

- ☐ *En VLAN, les machines sont organisées de façon logique et non physique*
- ☐ *Les VLANs permettent de s'affranchir de contraintes géographiques*
- ☐ *Le VLAN est la capacité d'affecter les ports d'un switch dans des LAN différents*

*Types de VLANs existants*

- ☐ *VLAN par port*
- ☐ *VLAN-MAC*
- ☐ *VLAN par protocole*

*VLANs très sécurisés avec 0x8100 comme Ethertype*

- ☐ *VLAN dot1q ou IEEE802.1Q ou VLAN tagué*
- ☐ *VLAN sans switch*

Préambule	Début du délimiteur de trame (frame delimiter)	MAC destination	MAC source	tag 802.1Q (optionnel)	Ethertype (Ethernet II) ou longueur (IEEE 802.3)	LLC/SNAP (si 802.3) + Payload	Frame check sequence (32-bit CRC)
7 octets	1 octet	6 octets	6 octets	(4 octets)	2 octets	46-1 500 octets	4 octets

## Protocole ARP

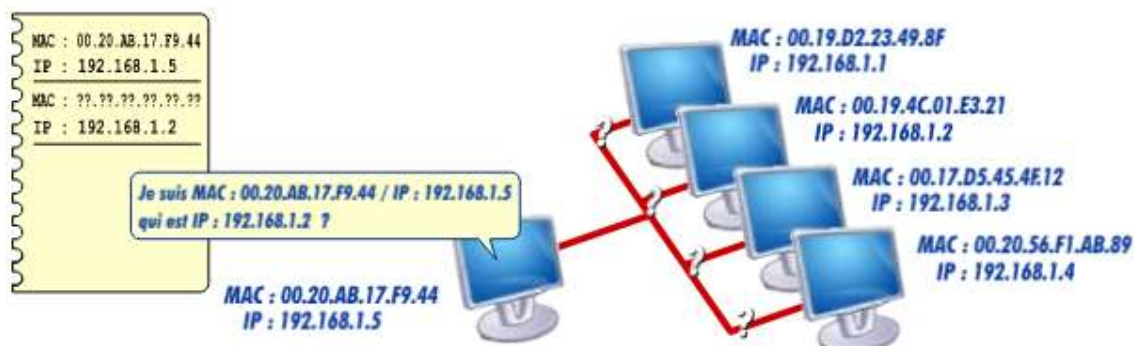
Le protocole **ARP** de la couche réseau permet de connaître l'adresse physique d'une carte réseau correspondant à une adresse IP connue.

ARP interroge toutes les machines du réseau. S'il obtient une réponse, il met à jour une table de correspondance entre les adresses logiques et les adresses physiques.

**ARP = Address Resolution Protocol (Protocole de résolution d'adresse)**

Lorsqu'une machine doit communiquer avec une autre, elle le fait à partir de l'adresse IP (seule adresse connue par les couches supérieures).

Si l'adresse IP demandée n'est pas encore connue par l'émetteur, le protocole ARP émet une requête sur le réseau.





## Protocole ARP

Le protocole **ARP** de la couche réseau permet de connaître l'adresse physique d'une carte réseau correspondant à une adresse IP connue.

ARP interroge toutes les machines du réseau. S'il obtient une réponse, il met à jour une table de correspondance entre les adresses logiques et les adresses physiques.

**ARP = Address Resolution Protocol (Protocole de résolution d'adresse)**

Lorsqu'une machine doit communiquer avec une autre, elle le fait à partir de l'adresse IP (seule adresse connue par les couches supérieures).

Si l'adresse IP demandée n'est pas encore connue par l'émetteur, le protocole ARP émet une requête sur le réseau.

Les machines du réseau vont comparer l'adresse IP cherchée avec la leur.

Celle qui se reconnaît va répondre à ARP.



*ARP est un protocole indispensable au fonctionnement des réseaux locaux, mais il présente une faille de sécurité aisément exploitable par un intrus pour détourner les communications vers une station et pire si cette attaque porte sur la passerelle ou sur le serveur DNS.*

*Mais pourquoi doit-on avoir 2 adresses MAC + IP pour une seule interface (machine) ?*

*Une adresse IP est dite logique et une adresse MAC est dite physique (donc pas logique ?)*

## Connaitre les adresses IP et MAC de mon ordinateur

Lancer, sous Windows, la fenêtre de la ligne de commande, également connue sous Shell ou fenêtre DOS. Une fois le prompt affiché, saisir la commande suivante :

***Ipconfig /all***      *meilleurs rendu que ipconfig*

Scruter le résultat affiché pour retrouver vos interfaces de connexion au réseau (cartes Ethernet et Wifi)

```
Carte réseau sans fil Wi-Fi :
    Suffixe DNS propre à la connexion. . . : lan
    Description. . . . . : Qualcomm Atheros AR956x Wireless Net
work Adapter
    Adresse physique . . . . . : 40-E2-30-FF-11-61
    DHCP activé. . . . . : Oui
    Configuration automatique activée. . . : Oui
    Adresse IPv6 de liaison locale. . . . . : fe80::fc20:f4ad:70d5:ea75%4(préfééré)

    Adresse IPv4. . . . . : 192.168.1.50(préfééré)
    Masque de sous-réseau. . . . . : 255.255.255.0
    Bail obtenu. . . . . : jeudi 31 octobre 2019 19:30:47
    Bail expirant. . . . . : lundi 4 novembre 2019 22:46:59
    Passerelle par défaut. . . . . : 192.168.1.254
    Serveur DHCP . . . . . : 192.168.1.254
    IAID DHCPv6 . . . . . : 88138288
    DUID de client DHCPv6. . . . . : 00-01-00-01-1C-B2-AD-3E-1C-B7-2C-15-D7
-BB
    Serveurs DNS. . . . . : 192.168.1.254
    NetBIOS sur Tcpip. . . . . : Activé
```

*Exemple de résultat d'ipconfig sur un PC relié à un réseau par WiFi*

Décortiquons ensemble quelques points remarquables du résultat de ipconfig, à vos plumes :

➤ DHCP

➤ Deux BAUX

➤ **Passerelle par défaut**

➤ **Serveurs DNS** (s'il y en a qu'un seul, c'est la passerelle par défaut qui joue ce rôle...en règle générale).

## Protocole ARP

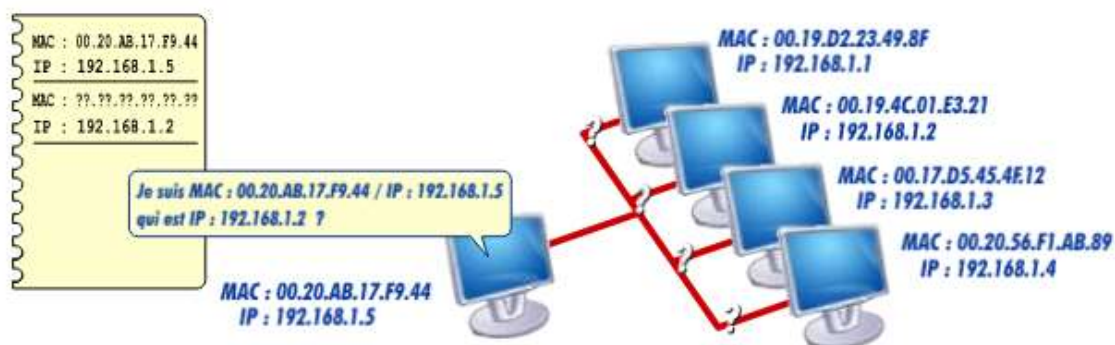
Le protocole **ARP** de la couche réseau permet de connaître l'adresse physique d'une carte réseau correspondant à une adresse IP connue.

ARP interroge toutes les machines du réseau. S'il obtient une réponse, il met à jour une table de correspondance entre les adresses logiques et les adresses physiques.

**ARP = Address Resolution Protocol (Protocole de résolution d'adresse)**

Lorsqu'une machine doit communiquer avec une autre, elle le fait à partir de l'adresse IP (seule adresse connue par les couches supérieures).

Si l'adresse IP demandée n'est pas encore connue par l'émetteur, le protocole ARP émet une requête sur le réseau.





## Protocole ARP

Le protocole **ARP** de la couche réseau permet de connaître l'adresse physique d'une carte réseau correspondant à une adresse IP connue.

ARP interroge toutes les machines du réseau. S'il obtient une réponse, il met à jour une table de correspondance entre les adresses logiques et les adresses physiques.

**ARP = Address Resolution Protocol (Protocole de résolution d'adresse)**

Lorsqu'une machine doit communiquer avec une autre, elle le fait à partir de l'adresse IP (seule adresse connue par les couches supérieures).

Si l'adresse IP demandée n'est pas encore connue par l'émetteur, le protocole ARP émet une requête sur le réseau.

Les machines du réseau vont comparer l'adresse IP cherchée avec la leur.

Celle qui se reconnaît va répondre à ARP.



© Thierry Schanen - 2012

*Un nouvel allié, après **ping** et **ipconfig**, voici  
**arp -a***

Pour retrouver les adresses **IP** et **MAC** des « machines » de notre sous-réseau, la commande **arp -a** est la plus simple. Cependant, elle n'affiche que les adresses de certaines machines de votre sous-réseau. Nous verrons par la suite en travaux pratiques comment retrouver les adresses de toutes les machines « connectées » de notre sous-réseau.

A noter que toute machine, de notre sous-réseau, ayant échangé avec notre PC figurera dans la liste affichée par la commande **arp -a (table ARP)**. Nous vérifierons cette assertion en constatant que cette liste s'accroît au fur et à mesure que nous interrogerons des machines initialement absentes de la liste.

On pourrait également enregistrer une entrée statique dans la table ARP (association adresses IP – MAC) grâce à la commande **arp -s** :

**Exemple arp -s 192.168.1.233 0A-0B-0C-01-02-03**

**arp -s** est un bon remède contre l'usurpation d'arp (*arp spoofing*).

Il est également possible de supprimer une entrée grâce à la commande **arp -d**

**Exemple arp -d 192.168.1.17**

ET vider la table ARP grâce à la commande **arp -d**

*Exemple de résultat de arp -a*

```
Interface : 192.168.1.50 --- 0x4
Adresse Internet      Adresse physique      Type
192.168.1.6           b8-78-26-93-58-0d     dynamique
192.168.1.8           c0-c9-76-cb-74-ab     dynamique
192.168.1.30          9c-80-df-e8-5e-09     dynamique
192.168.1.35          fc-18-3c-9f-8e-96     dynamique
192.168.1.45          d0-05-2a-c2-ba-98     dynamique
192.168.1.65          6c-19-c0-b4-07-c4     dynamique
192.168.1.75          b8-27-eb-fa-00-fe     dynamique
192.168.1.92          b8-27-eb-af-55-ab     dynamique
192.168.1.97          9c-8c-6e-64-0a-34     dynamique
192.168.1.254         d0-6e-de-c1-40-8c     dynamique
192.168.1.255         ff-ff-ff-ff-ff-ff     statique
224.0.0.2             01-00-5e-00-00-02     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
224.0.1.187           01-00-5e-00-01-bb     statique
239.255.255.250       01-00-5e-7f-ff-fa     statique
255.255.255.255       ff-ff-ff-ff-ff-ff     statique

Interface : 169.254.18.35 --- 0x14
Adresse Internet      Adresse physique      Type
169.254.255.255       ff-ff-ff-ff-ff-ff     statique
224.0.0.2             01-00-5e-00-00-02     statique
224.0.0.22            01-00-5e-00-00-16     statique
224.0.0.252           01-00-5e-00-00-fc     statique
224.0.1.187           01-00-5e-00-01-bb     statique
255.255.255.255       ff-ff-ff-ff-ff-ff     statique

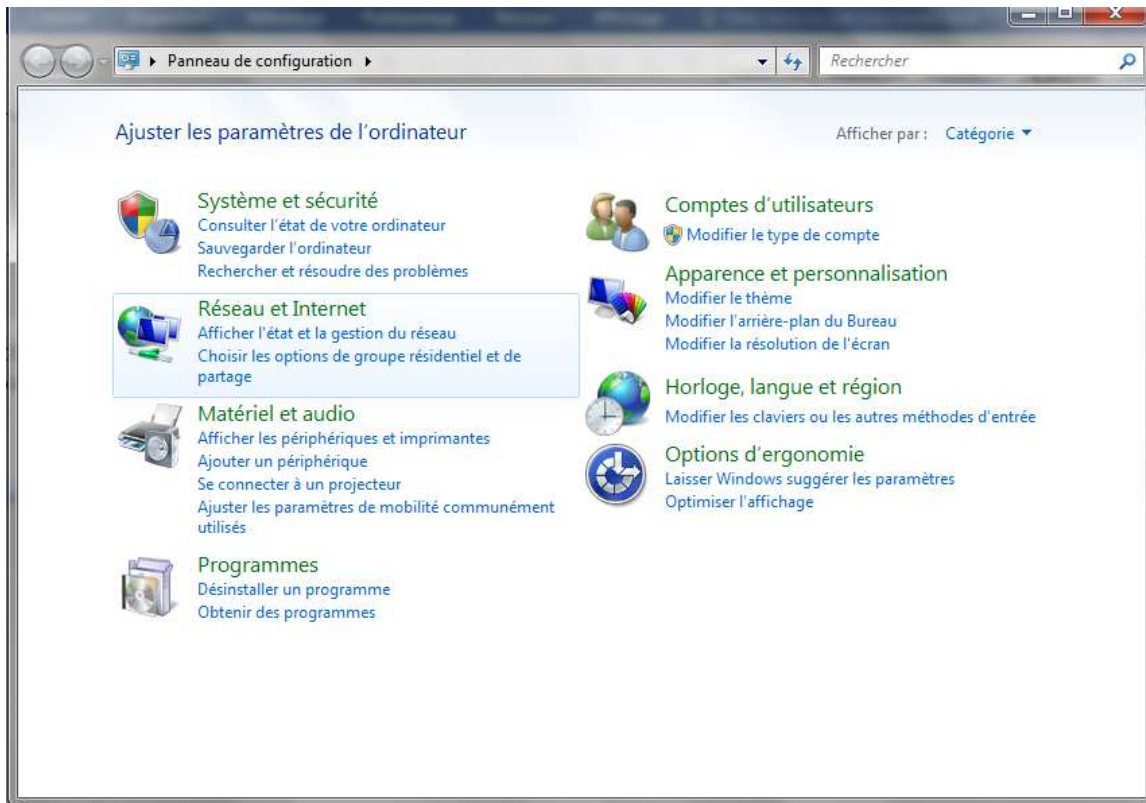
Interface : 169.254.125.161 --- 0x15
Adresse Internet      Adresse physique      Type
```



Sur la figure ci-dessus, on voit les table ARP de différentes interfaces, pour ne consulter que la table ARP d'une seule interface, par exemple l'interface 192.168.1.56 on procède ainsi : **arp -a -N 192.168.1.56**

```
C:\Users\ [redacted] >arp -a -N 192.168.1.50
Interface : 192.168.1.50 --- 0x4
Adresse Internet      Adresse physique      Type
192.168.1.6           b8-78-26-93-58-0d     dynamique
192.168.1.8           c0-c9-76-cb-74-ab     dynamique
192.168.1.30          9c-80-df-e8-5e-09     dynamique
192.168.1.45          d0-05-2a-c2-ba-98     dynamique
192.168.1.92          b8-27-eb-af-55-ab     dynamique
192.168.1.97          9c-8c-6e-64-0a-34     dynamique
192.168.1.254         d0-6e-de-c1-40-8c     dynamique
192.168.1.255         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
224.0.1.187           01-00-5e-00-01-bb     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\ [redacted]
```

# Configurer manuellement une adresse IP sous MS Windows



Afficher l'état et la gestion du réseau

Modifier les paramètres de la carte

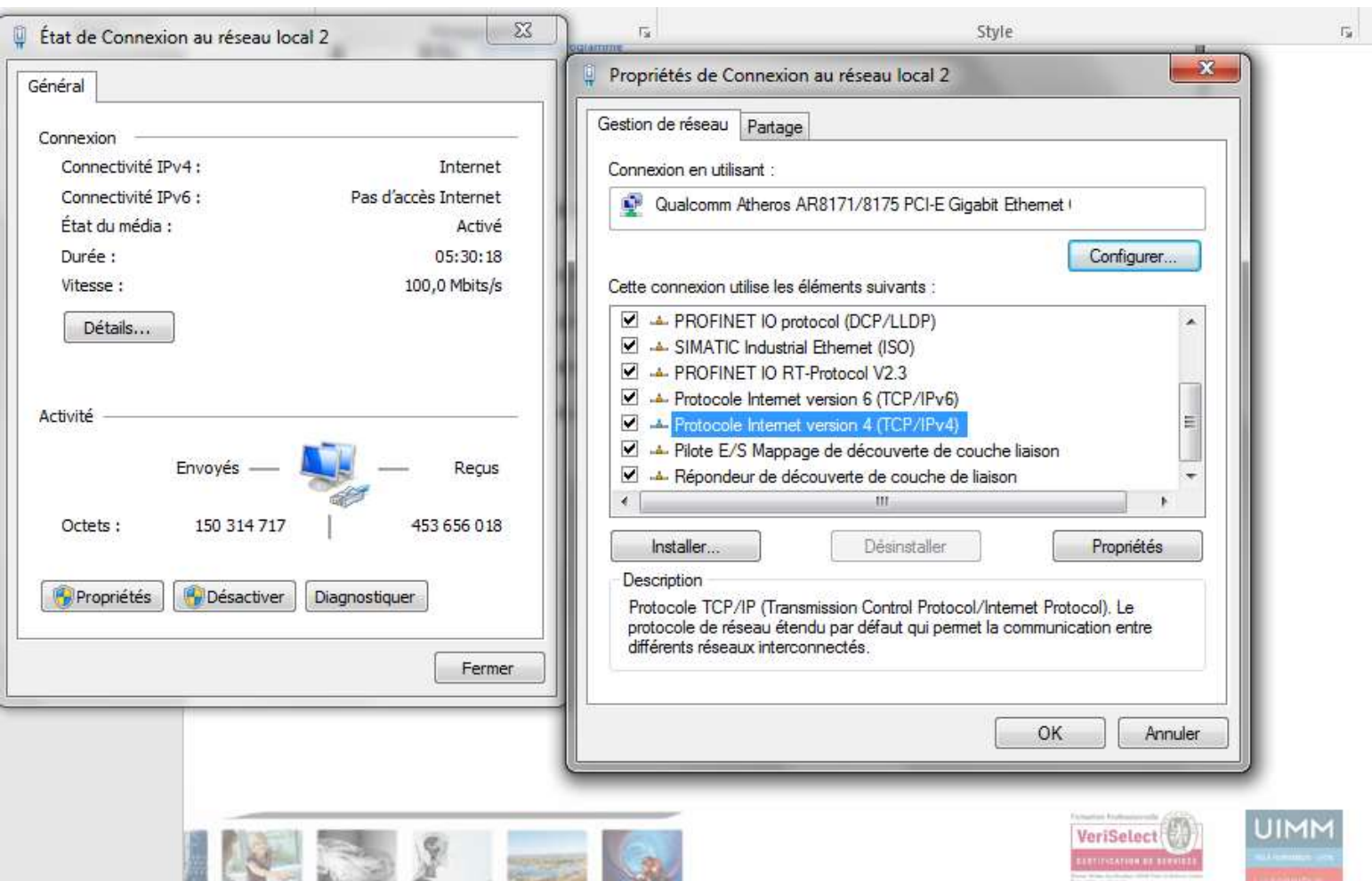
Double-Cliquer sur « connexion au réseau local 2 (afpm.fr)

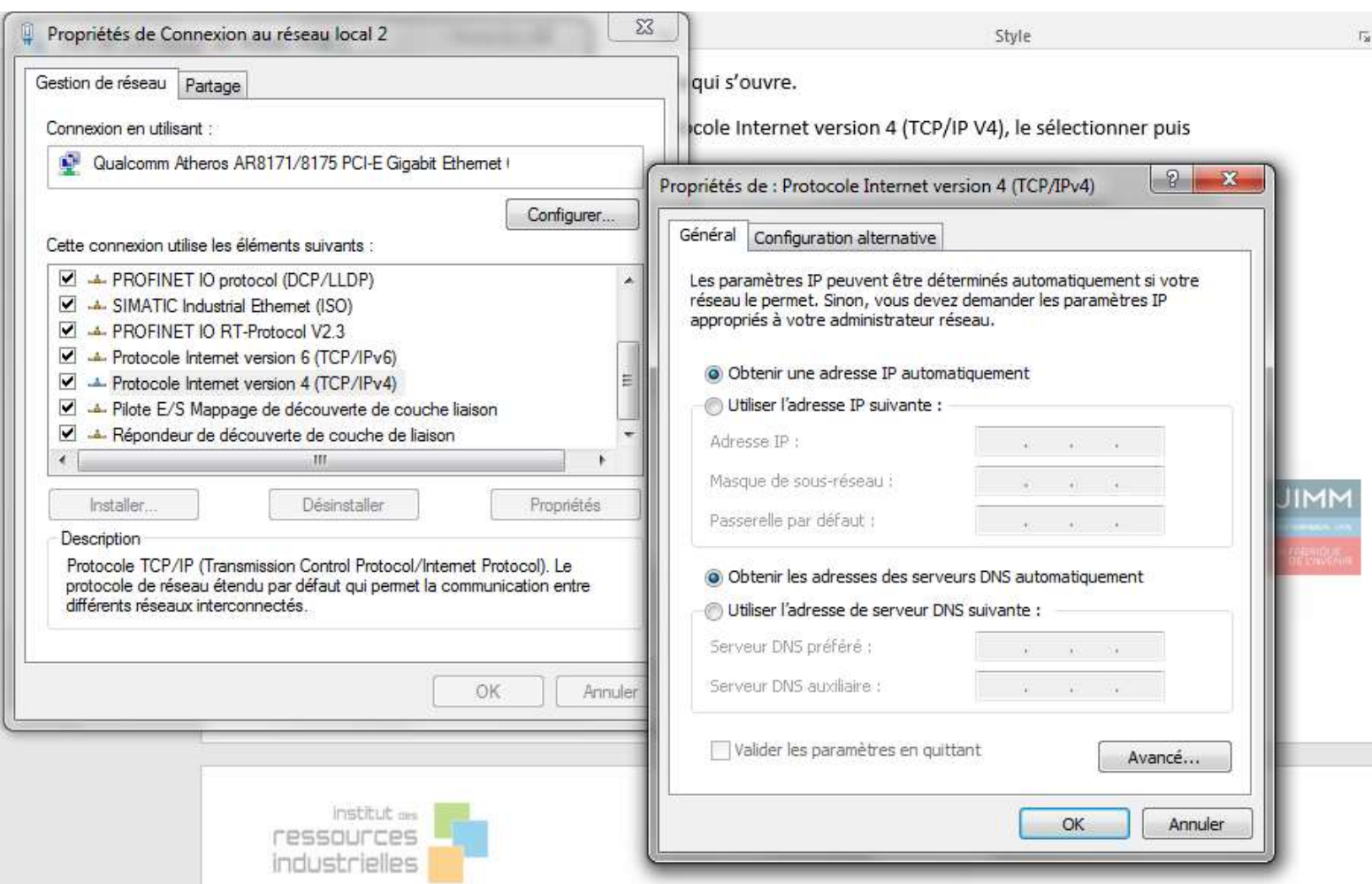
Puis sur « propriétés » dans la fenêtre qui s'ouvre.

Aller chercher la ligne suivante « **Protocole Internet version 4 (TCP/IP V4)** », le sélectionner puis :

Cliquer sur propriétés

Procéder aux affectations manuelles d'adresses IP, DNS...etc...(Des informations que l'on peut trouver grâce à la commande *ipconfig /all*)





## Connaître l'adresse IP d'une machine située hors de mon réseau local : la commande ping

La commande « ping » permet d'interroger une machine par son adresse IP ou son nom de domaine :

ping 192.168.1.50

ping google.fr

ping nasa.org

ping google.jp

Analyser et tirer des conclusions des statistiques affichées par ces commandes, durées moyennes, minimum et maximum.

Puis-je connaître l'adresse MAC du serveur de la NASA ?

Non. C'est pourquoi faire ?

Puis-je connaître l'adresse MAC du serveur de impots.gouv.fr?

Jamais. Ce n'est pas permis d'y penser.

Il faudra se contenter de leurs adresses IP. Un point c'est tout.

## Qui attribue les adresses IP publiques ?

Chaque fournisseur d'accès internet FAI possède une plage d'adresses IP fournie par l'AFNIC qui lui-même l'obtient auprès de l'ICANN et qu'il redistribue à ses clients pour que ces derniers se connectent à Internet.

Aller sur le site [www.mon-ip.com](http://www.mon-ip.com) pour afficher l'adresse IP publique du réseau local de votre entreprise ou de votre domicile (IPv4).

Cette adresse ne peut servir à communiquer avec un équipement de votre réseau local que depuis l'extérieur de votre réseau.

Pour les communications entre équipements d'un même réseau local, seules les adresses privées sont utilisées.

## Ports TCP/IP

On dit d'une machine qu'elle est un serveur, dès lors qu'elle fournit un service.

Le client est simplement un programme qui se connecte à un service pour l'utiliser.

Un navigateur web Mozilla FireFox, IE, MS Edge, Chrome, Opera ...sont des clients web permettant de se connecter à un serveur.

C'est bien une **connexion client/serveur** qui est établie entre le navigateur et un serveur web distant ou local.

### C'est quoi un port ?



Non ce n'est pas ça ! Là il s'agit de ports de switch (commutateur ou multiplexeur logique)

**Un port TCP/IP** est une adresse ! C'est même l'adresse d'une application sur une machine. Un nombre entier sur 16 bits (de 0 à 65535) servant à identifier une application parmi tant d'autres sur une même machine. C'est bien donc « l'adresse » ou l'identifiant d'une application s'exécutant sur une machine et ayant une connexion « réseau » avec une autre.

Un serveur écoute sur un port dédié, et un client utilise son propre port pour recevoir les données du serveur. Le couple adresse IP + Port se nomme **Socket**.



Plusieurs programmes **TCP/IP** peuvent être exécutés simultanément sur le même ordinateur, chacun de ces programmes travaille avec un protocole, toutefois l'ordinateur doit pouvoir distinguer les différentes sources et destination des données.

- Les ports **0 à 1023** sont les « **ports reconnus** » ou réservés «**Well Known Ports**». Ils sont, de manière générale, réservés à des applications standardisées.
- Les ports **1024 à 49151** sont appelés « **ports enregistrés** » (« **Registered Ports** »).
- Les ports **49152 à 65535** sont les « **ports dynamiques et/ou privés** » «**Dynamic and/or Private Ports**».

Voici certains des ports reconnus les plus couramment utilisés :

Port	Service ou Application
21	<a href="#">FTP</a>
23	<a href="#">Telnet</a>
25	<a href="#">SMTP</a>
53	<a href="#">Domain Name System</a>
63	Whois
70	Gopher
79	Finger
80	<a href="#">HTTP</a>
110	<a href="#">POP3</a>

SSH → **22**    HTTPS → **443**

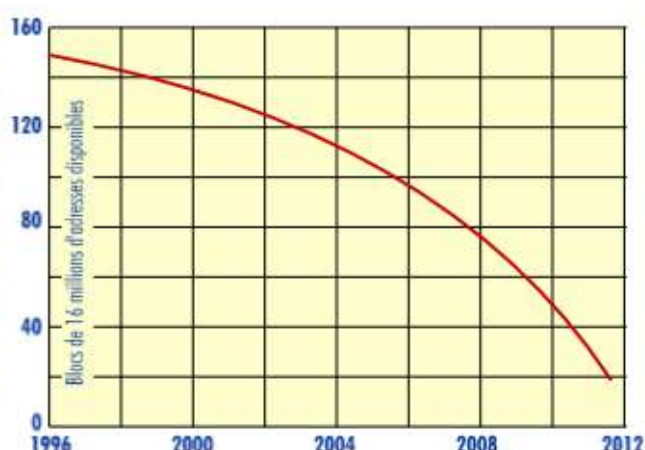
**Mais où sont les numéros de port pour TCP ? UDP ?**

Si vous réitérez une question de cet acabit, je demanderai à ce que vous soyez réinscrit pour une nouvelle session de formation.

## IPv6

Le nombre d'adresses IP disponibles avec le protocole actuel IPv4 (environ 4 milliards) ne suffit plus pour répondre à la demande croissante d'adresses fixes (téléphonie par IP, réseaux embarqués dans l'automobile, Internet dans les pays en voie de développement...)

L'épuisement des adresses publiques est inévitable (la technique d'attribution par classes ainsi que des pans entiers d'adresses bloquées limitent d'autant la réserve d'adresses).



En avril 2011, les derniers blocs libres ont commencé à être assignés et les prévisionnistes estiment l'épuisement complet pour le **début de l'année 2015**.

C'est la raison principale du développement d'un nouveau protocole Internet. :

## IPv6

## IPv6

La nouvelle norme **IPv6** (publiée dès 1995) est encore en cours de développement et de déploiement.

A terme, cette nouvelle norme doit remplacer l'actuelle norme IPv4 mais des contraintes matérielles et économiques en freinent le déploiement.

Outre l'augmentation du nombre d'adresses, la norme IPv6 améliore le routage des données en simplifiant les entêtes des paquets manipulés et offre des mécanismes de configuration et de renumérotation automatique.

L'adressage se fera sur 8 doubles octets, soit  $2^{128}$  adresses possibles.

**aaaa.aaaa.aaaa.aaaa.aaaa.aaaa.aaaa.aaaa**

La notation hexadécimale pointée utilisée avec IPv4 est abandonnée au profit d'une notation hexadécimale avec séparation des doubles octets par ":"

**2011:10B5:0000:46E7:0000:0000:AA28:2A26**

Les groupes consécutifs de 16 bits nuls peuvent être omis en conservant les séparateurs :

**2011:10B5::46E7:::AA28:2A26**

## La commande netsat

**netstat** est un outil utile pour vérifier les connexions réseau et Internet.

### Options de la commande Netstat

Commutateur	Description
-a	Affiche toutes les connexions et les ports en écoute
-n	Affiche les adresses et les numéros de ports au format numérique

Les connexions TCP et UDP ainsi que leurs adresses IP et port peuvent être obtenues en entrant la commande :

**netstat -an**

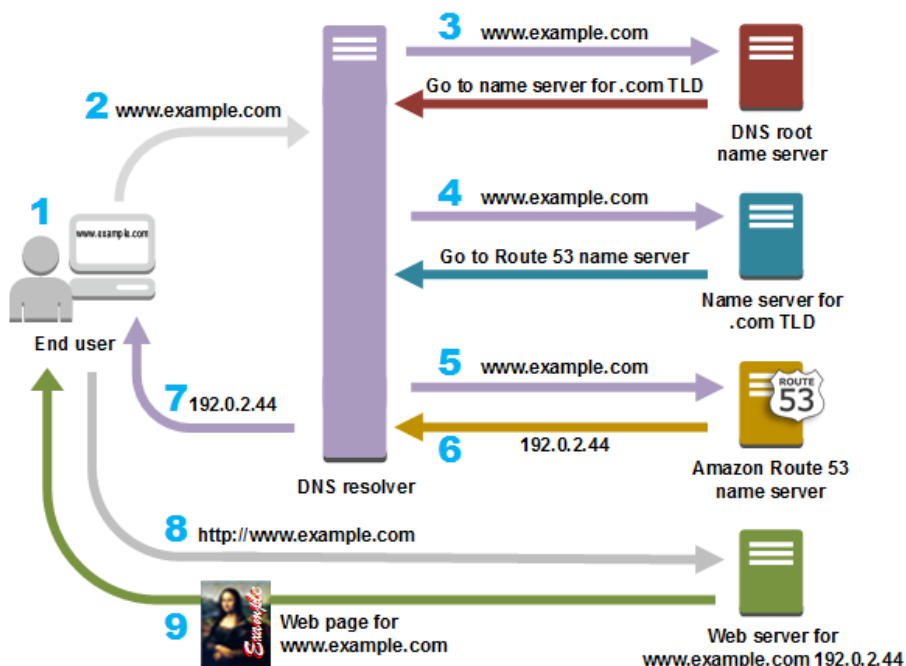
### Description des différents statuts de connexion

État	Description
CLOSED	Indique que le serveur a reçu un signal ACK envoyé par le client et que la connexion est fermée.
CLOSE_WAIT	Indique que le serveur a reçu le premier signal FIN envoyé par le client et que la connexion est en train d'être fermée.
ESTABLISHED	Indique que le serveur a reçu le signal SYN envoyé par le client et que la session est établie.
FIN_WAIT_1	Indique que la connexion est toujours active mais n'est pas utilisée actuellement.
FIN_WAIT_2	Indique que le client vient juste de recevoir l'accusé de réception du premier signal FIN envoyé par le serveur.
LAST_ACK	Indique que le serveur est en train d'envoyer son propre signal FIN.
LISTENING	Indique que le serveur est prêt à accepter une connexion.
SYN_RECEIVED	Indique que le serveur vient de recevoir un signal SYN envoyé par le client.
SYN_SEND	Indique que la connexion est ouverte et active.
TIME_WAIT	Indique que le client reconnaît la connexion comme encore activé mais non utilisée actuellement.

## Exemple d'exécution de netstat -an

```
TCP        127.0.0.1:14031      127.0.0.1:52001      ESTABLISHED
TCP        127.0.0.1:14032      127.0.0.1:52001      ESTABLISHED
TCP        127.0.0.1:14033      127.0.0.1:52001      ESTABLISHED
TCP        127.0.0.1:14034      127.0.0.1:52001      ESTABLISHED
TCP        127.0.0.1:21320      0.0.0.0:0            LISTENING
TCP        127.0.0.1:21321      0.0.0.0:0            LISTENING
TCP        127.0.0.1:21322      0.0.0.0:0            LISTENING
TCP        127.0.0.1:21323      0.0.0.0:0            LISTENING
TCP        127.0.0.1:21327      0.0.0.0:0            LISTENING
TCP        127.0.0.1:23165      127.0.0.1:23166      ESTABLISHED
TCP        127.0.0.1:23166      127.0.0.1:23165      ESTABLISHED
TCP        127.0.0.1:27015      0.0.0.0:0            LISTENING
TCP        127.0.0.1:27015      127.0.0.1:1121       ESTABLISHED
TCP        127.0.0.1:52001      0.0.0.0:0            LISTENING
TCP        127.0.0.1:52001      127.0.0.1:14031      ESTABLISHED
TCP        127.0.0.1:52001      127.0.0.1:14032      ESTABLISHED
TCP        127.0.0.1:52001      127.0.0.1:14033      ESTABLISHED
TCP        127.0.0.1:52001      127.0.0.1:14034      ESTABLISHED
TCP        169.254.18.35:139      0.0.0.0:0            LISTENING
TCP        169.254.125.161:139  0.0.0.0:0            LISTENING
TCP        192.168.1.50:139        0.0.0.0:0            LISTENING
TCP        192.168.1.50:32871      52.36.210.126:443     ESTABLISHED
TCP        192.168.1.50:32899      52.142.84.61:443      ESTABLISHED
TCP        192.168.1.50:34832      92.42.73.150:80        CLOSE_WAIT
TCP        192.168.56.1:139      0.0.0.0:0            LISTENING
TCP        [::]:135                  [::]:0                LISTENING
TCP        [::]:1445                  [::]:0                LISTENING
```

# DNS Domain Name Server



d'Amazon.

Le Domain Name System, généralement abrégé **DNS**, qu'on peut traduire en « système de noms de domaine », est le service informatique distribué utilisé pour traduire les noms de domaine Internet en adresse IP.

Amazon Route 53 est un serveur de nom de domaine privé bien connu sur le réseau Internet et destiné aux clients (entreprise)

Les datagrammes TCP/IP (trames) ne manipulent que des adresses IP, la conversion nom de domaine => adresse IP est un des principes fondamentaux du fonctionnement d'Internet.

C'est aussi des failles de sécurité dans les implémentations du DNS que des personnes malveillantes exploitent pour « attaquer » des réseaux d'entreprise et des sites Internet.

Le mDNS, s'il est exposé à Internet, présente une grave faille de sécurité pour un réseau local surtout pour les réseaux locaux abrités derrière un routeur de FAI.

Sans s'en rendre compte, nous avons utilisé le service DNS au cours des manipulations précédentes. Vous l'utilisez systématiquement quand vous sollicitez votre navigateur Internet.

Voyons ça de près....



## outil **tracert**

La commande **tracert** (également connu sous le vocable trace route) est un utilitaire de ligne de commande qui permet de suivre le chemin emprunté par un paquet IP (Internet Protocol) pour arriver à sa destination.

L'outil **tracert** détermine l'itinéraire vers une destination en envoyant des paquets d'écho ICMP (Internet Control Message Protocol) à la destination. Dans ces paquets, **tracert** tire profit des valeurs de durée de vie (Time-To-Live, TTL) pour identifier le chemin parcouru par les datagrammes pour atteindre une cible.

```
C:\> Administrateur : Invite de commandes

H:\>tracert google.fr

Détermination de l'itinéraire vers google.fr [172.217.22.131]
avec un maximum de 30 sauts :

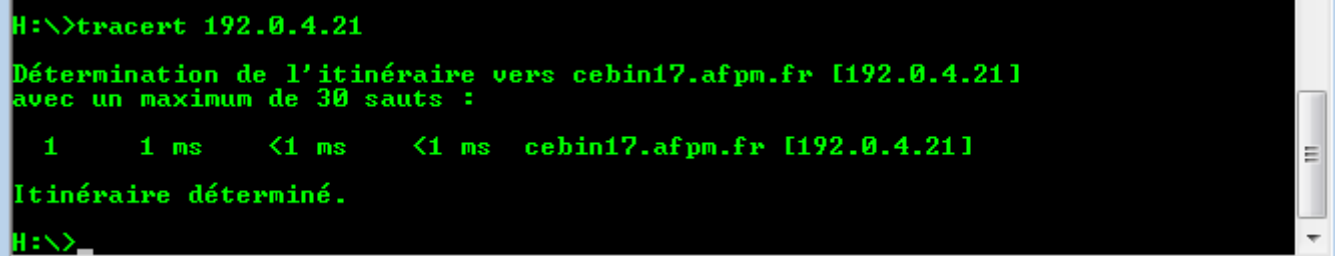
 1      <1 ms    <1 ms    <1 ms  firewall.iri-lyon.com [192.0.255.254]
 2      1 ms     1 ms     1 ms  host.233.77.23.62.rev.coltfrance.com [62.23.77.2
33]
 3      7 ms     7 ms     7 ms  62.23.112.77
 4      8 ms     8 ms     8 ms  212.36.135.175
 5      8 ms     8 ms     8 ms  212.36.135.175
 6      8 ms     8 ms     8 ms  72.14.219.202
 7      8 ms     8 ms     8 ms  108.170.244.193
 8      8 ms     8 ms     8 ms  66.249.95.103
 9      8 ms     8 ms     8 ms  par21s12-in-f3.1e100.net [172.217.22.131]

Itinéraire déterminé.
H:\>_
```

Exemple d'exécution de **tracert** vers google.fr

Étant donné que chaque routeur sur l'itinéraire doit diminuer la durée de vie d'un paquet d'au moins 1 avant de le transférer au routeur suivant, la TTL représente effectivement le nombre de sauts. Lorsque la TTL d'un paquet atteint zéro (0), le

routeur renvoie un message ICMP Temps dépassé à l'ordinateur émetteur de la requête.



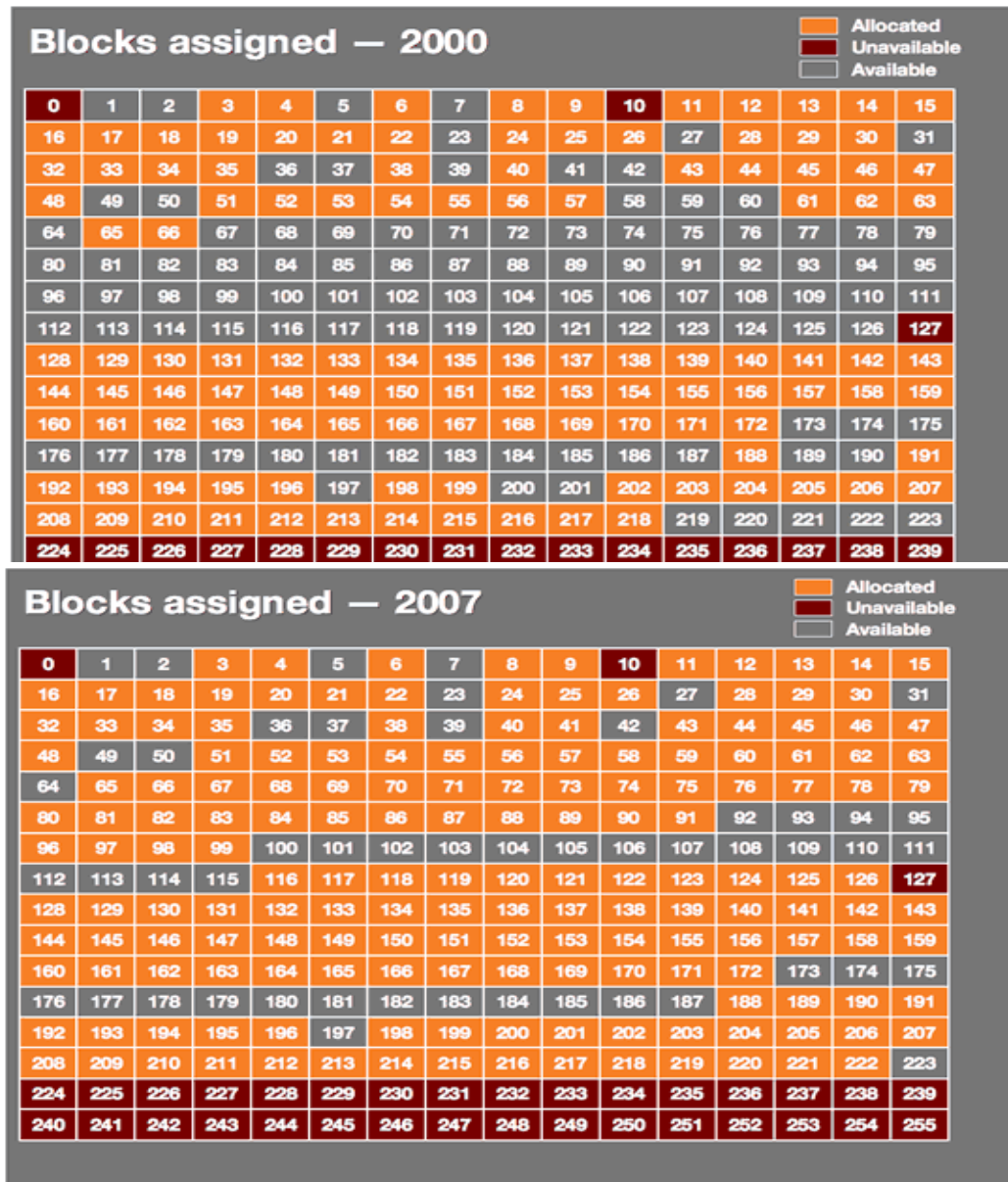
```
H:\>tracert 192.0.4.21
Détermination de l'itinéraire vers cebin17.afpm.fr [192.0.4.21]
avec un maximum de 30 sauts :

  1      1 ms    <1 ms    <1 ms  cebin17.afpm.fr [192.0.4.21]
Itinéraire déterminé.
H:\>
```

Exemple d'exécution de **tracert** vers une adresse de mon sous réseau, on y voit que l'identité du propriétaire de ce ordinateur est révélée ! On note également que la trame envoyée par **tracert** n'a traversé aucun routeur même pas celui de mon sous-réseau !

**Ouu..ff.** Ce réseau respecte (**vraiment ?**) les préceptes du TCP/IP.

# Le NAT et Port Forwarding



Utilisation des blocs d'adresses IP en 2007

## NAT : Network Address Translation

Le CIDR vu précédemment ne suffit pas à limiter le gaspillage des adresses IP V4, il a fallu faire appel au NAT et au Port forwarding pour y parvenir.

Le NAT Quèsaco ? Nous avons déjà constaté que le site <http://www.mon-ip.com/> donnait la même adresse IP pour toutes les machines de notre sous réseaux qui s'y connectent.

Sans le savoir, nous étions en pleine illustration de la NAT. Explications ?

Le Portforwarding Quèsaco ? c'est grâce à lui que plusieurs machines de mon sous réseaux peuvent se connecter sur Google pour consulter des documents sans que les données de Google à destination d'une machine ne se mélangent avec celles destinées à une autre.

Nous pouvons même utiliser deux navigateurs différents ou deux onglets différents du même navigateur sans que les données émanant d'internet ne se mélangent.

Sur une machine, on ne peut utiliser simultanément (que !) 65535 applications communicant avec d'autres machines, alors qu'un sous réseau peut comprendre des centaines de machines, dont certaines peuvent consulter simultanément les mêmes services web situés en dehors de mon sous réseau, alors comment fait le routeur de mon sous réseau pour ne pas perdre la tête et pour ne pas mélanger cet embrouillamini de trames de données ? C'est grâce au Portforwarding.

On pourrait voir grâce à un serveur embarqué sur microcontrôleur programmé en conséquence, comment évolue l'attribution des numéros de ports alloués à une application par le système d'exploitation Windows.

*Grâce aux routeurs adsl ou fibre et le portforwarding, finit donc le temps où les virus se baladaient dans les réseaux locaux comme dans du beur, les modem 56K de l'ancien régime ne pouvant que gérer une seule connexion internet.*

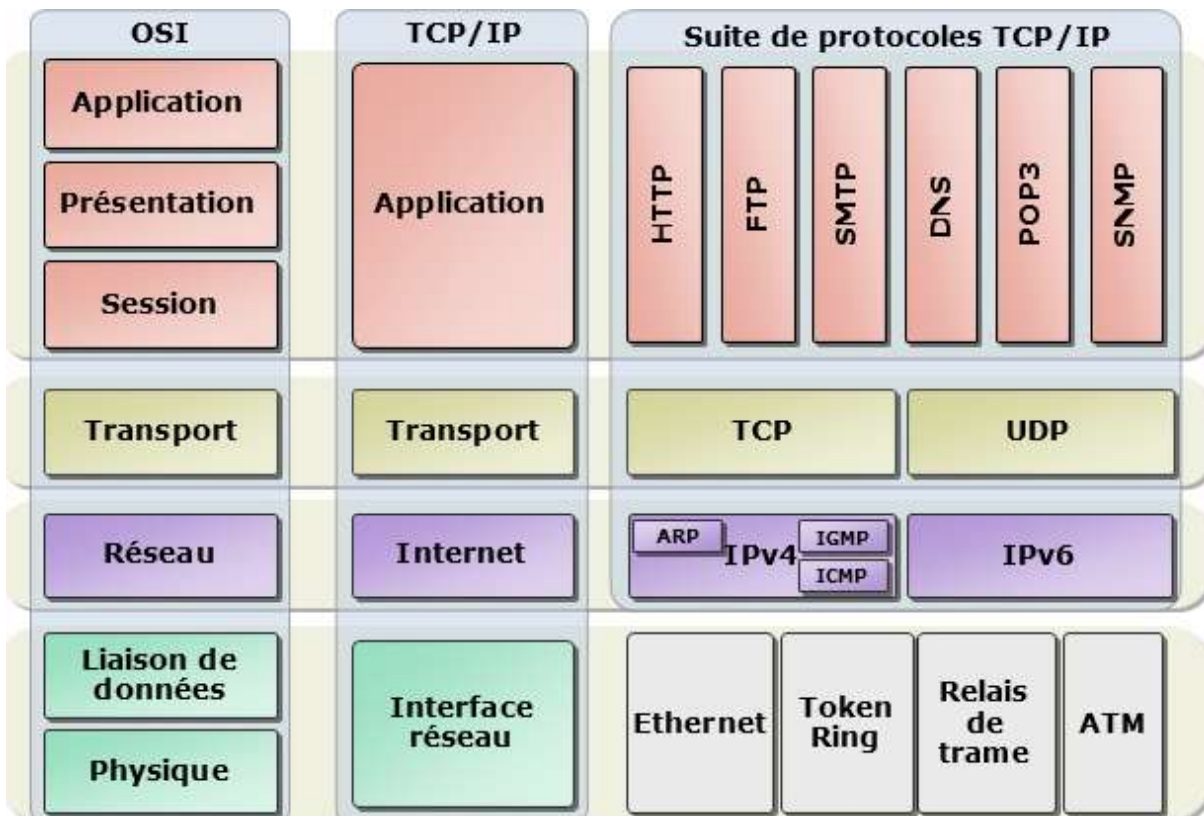
*En plus de la flexibilité qu'ils offrent, le NAT et le portforwarding ont contribué à la sécurité des réseaux locaux, en ne rendant accessible depuis l'extérieur, que ce qui est nécessaire.*

*En contrepartie, un réseau local, ne peut abriter qu'un SEUL serveur « web » d'un type donné (HTTP, SSH, DNS...) ou alors, il faut 'obliger' les clients à spécifier un numéro de port dans leurs requêtes.*

*Cette dernière limitation est conjurée par des solutions applicatives comme l'usage des serveurs web (Apache (62%), Microsoft-IIS (16%), Nginx (15%)...Google Servers(2%))*

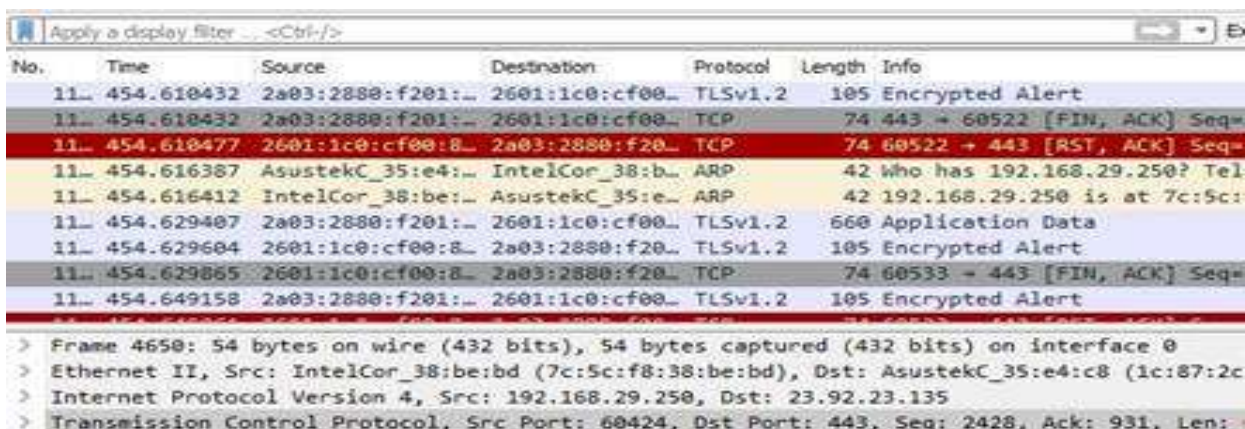


## Comment Utiliser Wireshark Pour Capturer, Filtrer Et Inspecter Les Paquets?



Wireshark, un outil d'analyse du réseau anciennement connu sous le nom d'Ethereal, capture les paquets en temps réel et les affiche dans un format lisible par un humain.

Wireshark inclut des filtres, un codage couleur et d'autres fonctionnalités qui vous permettent de creuser profondément dans le trafic réseau et d'inspecter les paquets individuels.





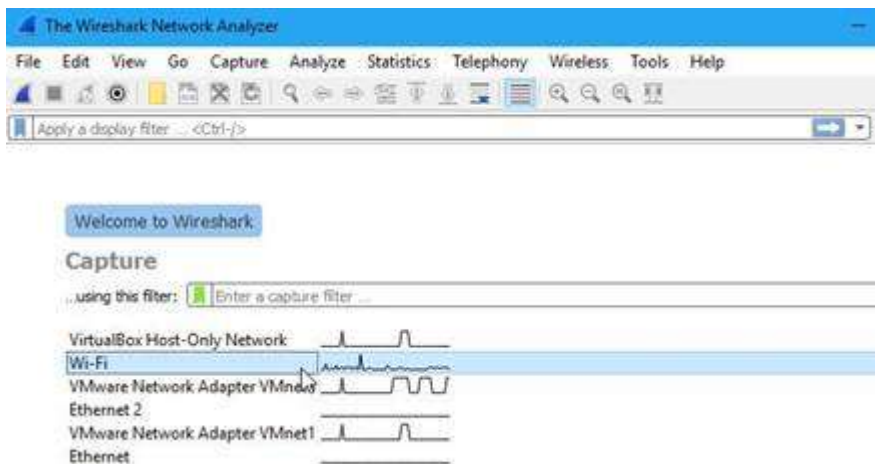
## UTILISER Wireshark

*Wireshark est en téléchargement libre sur son site officiel. Il suffit de taper son nom dans votre navigateur préféré pour y accéder.*

**Avertissement** Beaucoup d'organisations n'autorisent pas Wireshark et les outils similaires sur leurs réseaux. N'utilisez pas cet outil au travail sans autorisation.

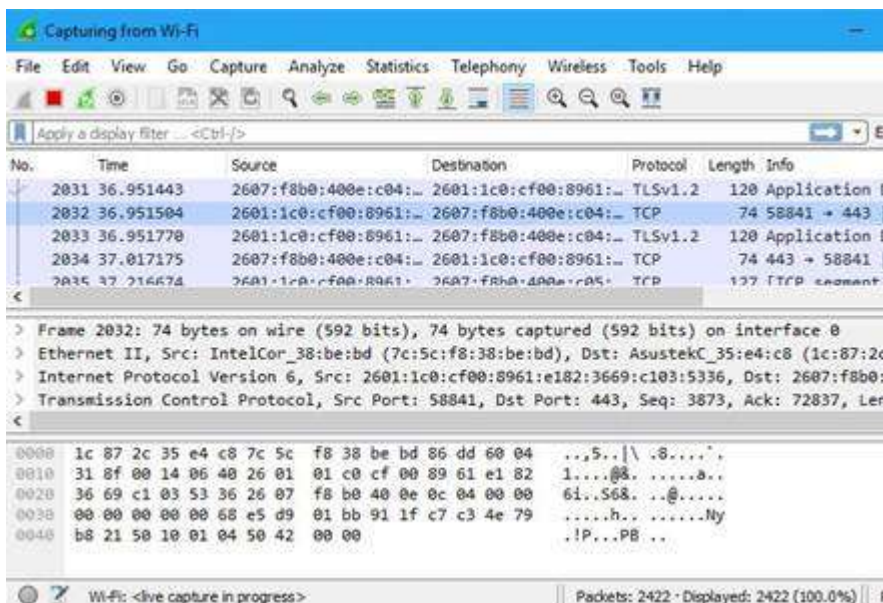
### Capter Les Paquets

Après avoir téléchargé et installé Wireshark, vous pouvez le lancer et double-cliquer sur le nom de l'interface réseau (*network interface*) sous **Capture** pour commencer à capturer les paquets sur cette interface. Par exemple, si vous souhaitez capturer du trafic sur votre réseau sans fil, cliquez sur votre interface sans fil (*Wi-Fi*).

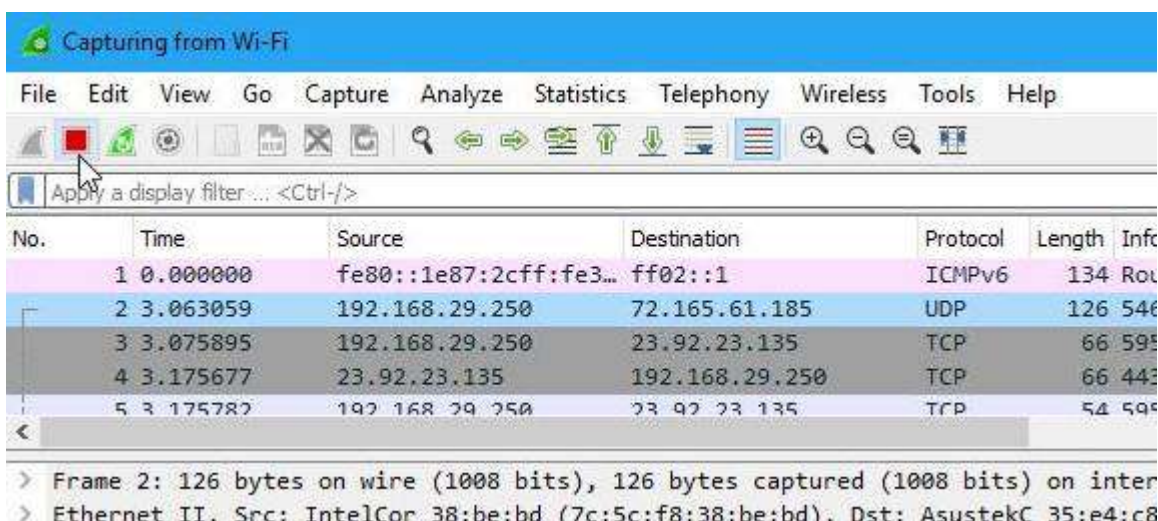


Dès que vous cliquez sur le nom de l'interface, les paquets commencent à apparaître en temps réel. Wireshark capture chaque paquet envoyé vers ou depuis le système hôte.

Si le mode Promiscuous est activé (*promiscuous mode*) —il est activé par défaut— tous les autres paquets du réseau sont également affichés et non pas uniquement les paquets destinés à votre carte réseau. Pour vérifier si le mode promiscuous (*promiscuous mode*) est activé, cliquez sur **Capture > Options** et vérifiez que la case à cocher « Activer le mode promiscuous sur toutes les interfaces » (*Enable promiscuous mode on all interfaces*) est activée au bas de cette fenêtre.



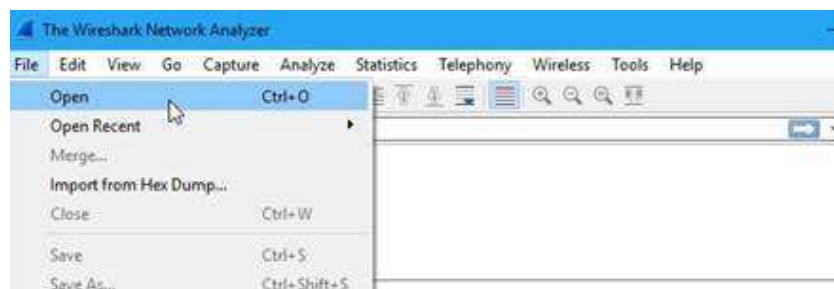
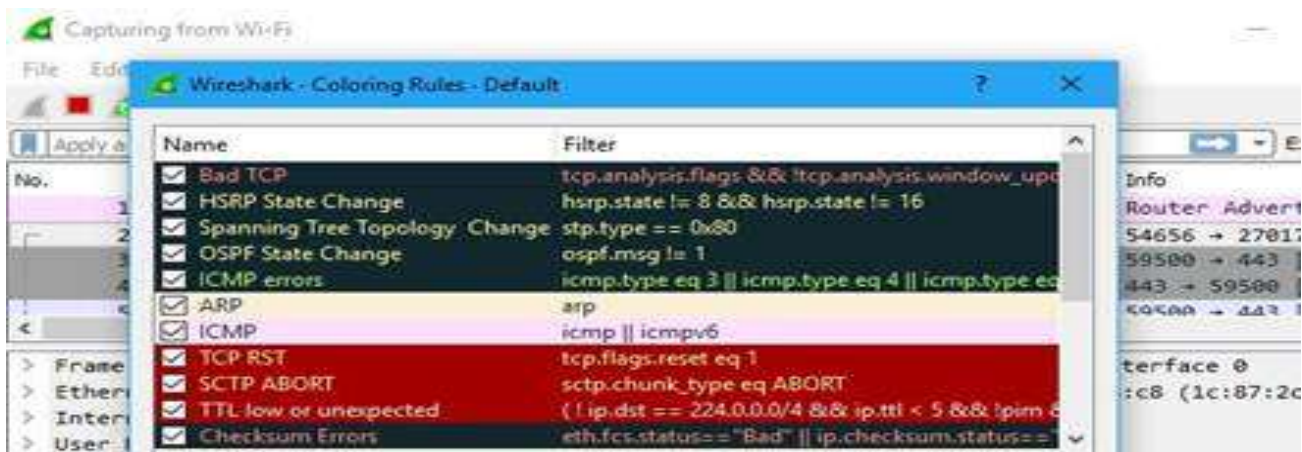
Cliquez sur le bouton rouge « *Stop* » près du coin supérieur gauche de la fenêtre lorsque vous souhaitez arrêter la capture du trafic.



## Codes de couleurs

Vous verrez probablement les paquets mis en évidence dans une variété de couleurs différentes. **Wireshark** utilise des couleurs pour vous aider à identifier les types de trafic en un coup d'œil. Par défaut, violet clair est le trafic TCP, bleu clair est le trafic UDP et noir identifie les paquets avec des erreurs, par exemple, ils pourraient avoir été livrés dans le désordre.

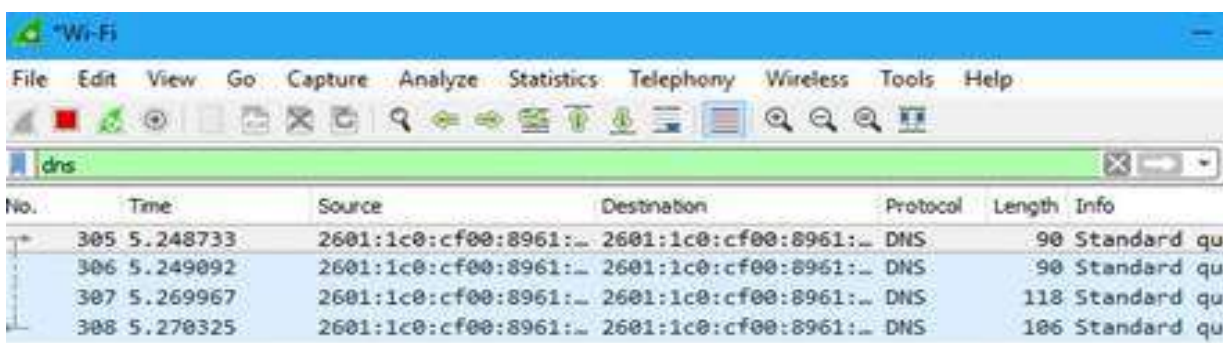
Pour voir exactement ce que signifient les codes de couleur, cliquez sur Affichage (ou vue) --> Règles de coloration (**View > Coloring Rules**). Ces règles sont modifiables.



## Filtrage des paquets

La manière la plus simple d'appliquer un filtre consiste à le saisir dans la zone de filtre en haut de la fenêtre et à cliquer sur Appliquer **Apply** (ou sur Entrée).

Par exemple, taper « **dns** » et on verra uniquement les paquets DNS.

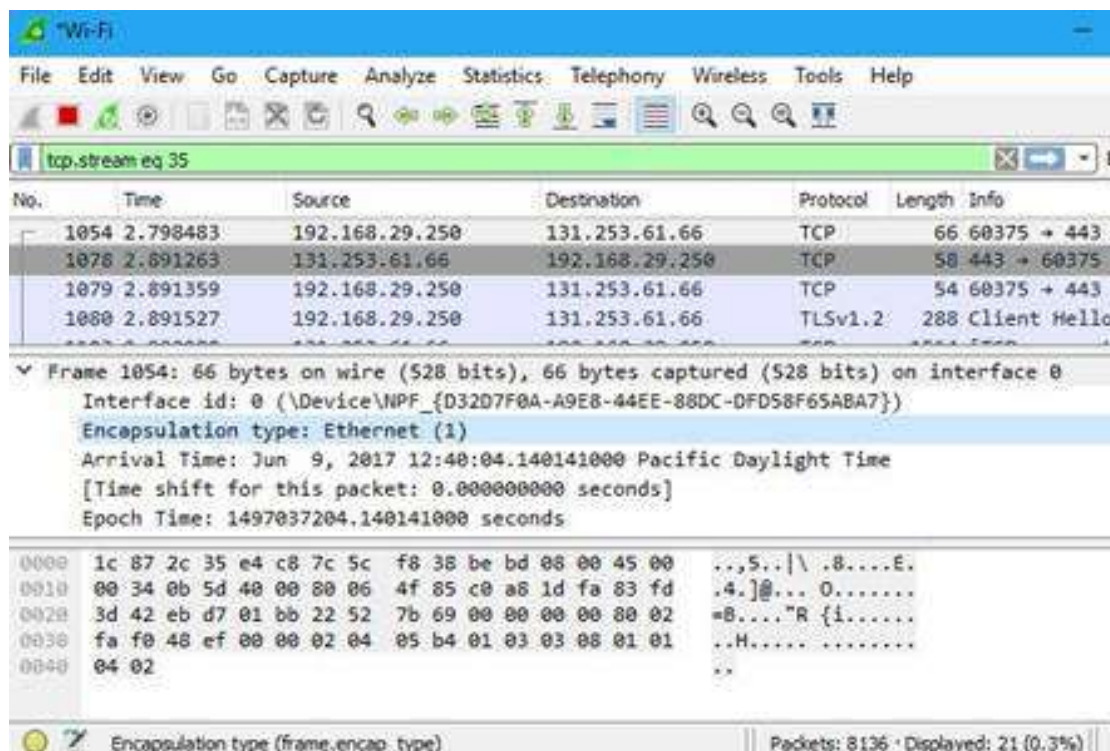
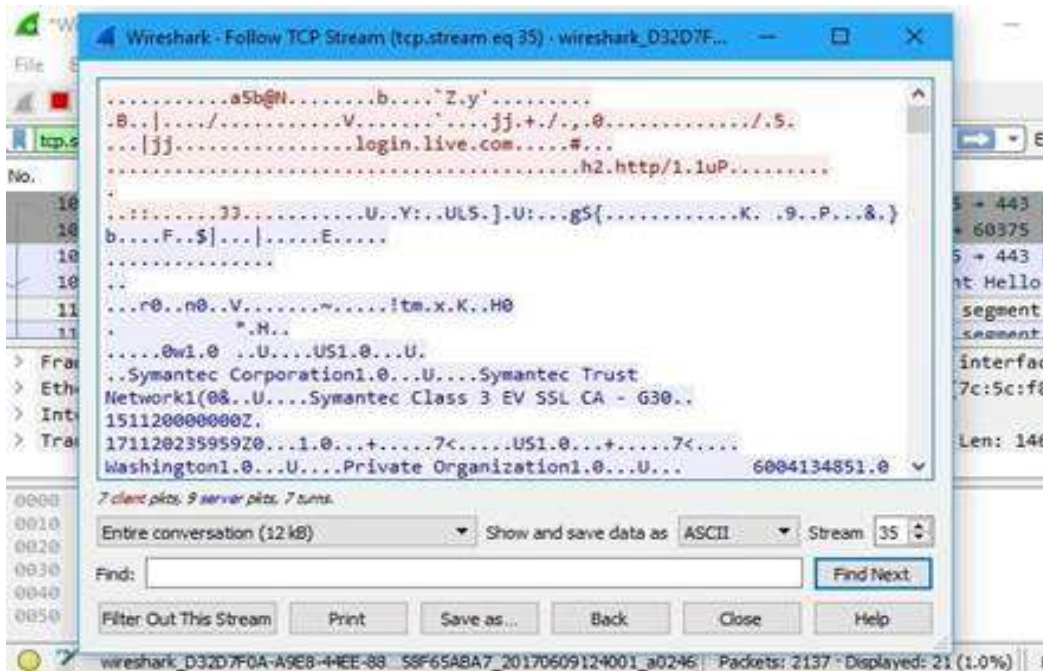


On pourrait également cliquer sur Analyser > Afficher les filtres (**Analyze > Display Filters**) pour choisir un filtre parmi les filtres par défaut inclus dans Wireshark.





Une autre particularité intéressante à explorer est de cliquer avec le bouton droit sur un paquet et de sélectionner Suivre> Flux TCP (*Follow > TCP Stream*).



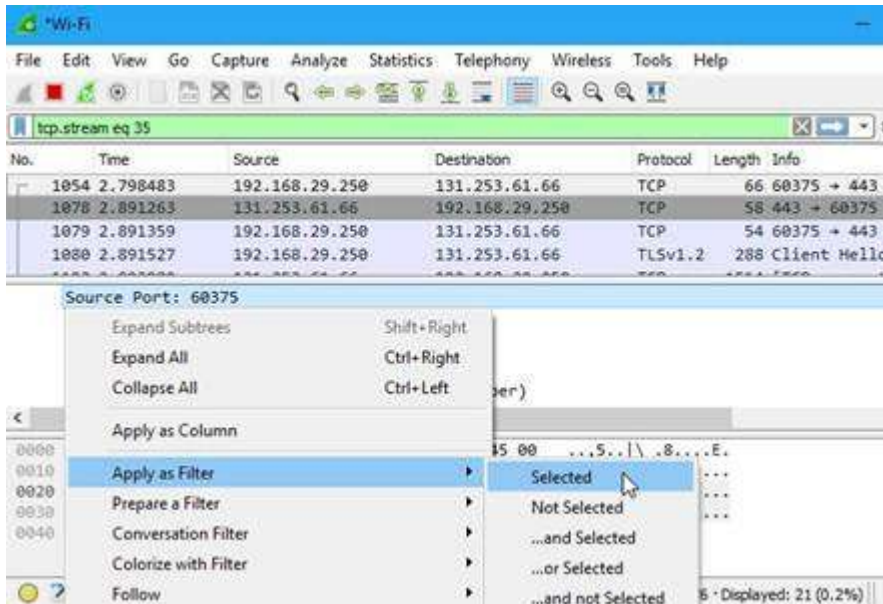
*Quand bien même il capture beaucoup de trames réseaux par unité de temps, Wireshark en rate plein d'autres surtout si aucun filtre n'est appliqué. Comment font donc les switches, qui eux doivent aiguiller toutes les trames sans en perdre une?*

*Même avec un Intel i9 64Go de RAM, difficile d'imaginer un OS avec une aussi-faible latence !*

## Inspection des paquets

Cliquez sur un paquet pour le sélectionner et vous pouvez creuser pour voir ses détails.

Vous pouvez également créer des filtres à partir d'ici – cliquez avec le bouton droit sur l'un des détails et utilisez le sous-menu Appliquer (*Apply*) en tant que filtre pour créer un filtre basé sur celui-ci.



Wireshark est un outil extrêmement puissant. Les professionnels l'utilisent pour déboguer les implémentations de protocoles réseaux, examiner les problèmes de sécurité et inspecter les composants internes d'un protocole réseau donné.

### Exemple de filtres récurrents

**ftp || tcp** - Paquets dont le type est : FTP, TCP.

**ip.addr == 10.20.144.150** - Paquets dont l'adresse IP source ou destination est 10.20.144.150

**ip.src == 10.20.144.150** - Paquets dont l'adresse IP source 10.20.144.150

**ip.dst == 10.20.144.151** - Paquets dont l'adresse IP source 10.20.144.151

**tcp.port == 35974** - Paquets dont le port source ou destination est 35974.

**tcp.srcport == 21** - Paquets dont le port source est 21 (port FTP).

**tcp.dstport == 21** - Paquets dont le port destination est 21.

Opérateurs de comparaison :

==	Est égal à
!=	N'est pas égal à
>	Plus grand que
<	Plus petit que
>=	Plus grand ou égal que
<=	Plus petit ou égal que

Opérateurs logiques :

	Ou
&&	Et
^^	Ou exclusif
!	Négation

## Exemple de filtres plus complexes :

`ip.dst == 10.20.144.151 && (tcp.dstport == 35974 || tcp.dstport == 21)`

Recherche les paquets à destination de 10.20.144.151 sur les ports TCP 35974 ou 21

**Wireshark** ne fait pas de miracle, il est possible de l'aveugler moyennant l'envoi de trames Ethernet qu'il ne saurait déchiffrer comme des trames Ethernet IEEE802.3.

# Notions de communications sans Fil WiFi



- Année 1999 : la genèse du WiFi à 2Mbit/s !!! (Les modems 56K étaient encore courants, même les PC ne pouvaient franchir le seuil des 115 200 Bauds en liaison série).

- Septembre 1999 deux standards WiFi s'affrontent : WiFi a (54 Mbit/s) vs Apple WiFi b (11 Mbit/s)

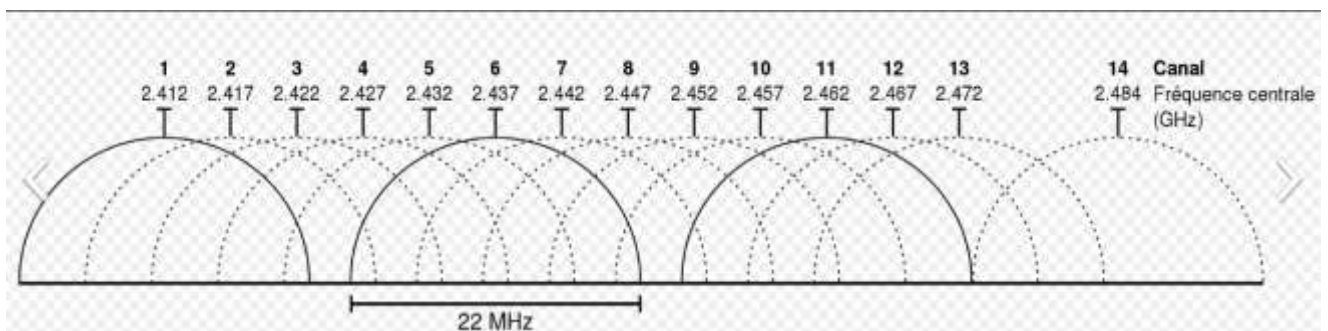
- Marketing efficace de Apple pour promouvoir le 'b' grâce à son iBook => Apple gagne la bataille de la portée grâce à l'usage de l'ISM.

- 2000/2001 WiFi g compatible avec le 'b' (2.4GHz) mais supportant le débit du 'a' (54Mbit/s), il sonne le glas du b tout en étant compatible avec lui avec un débit plus limité, mais surtout il marque la mise à mort du 'a' !

- Arrivée du n à 150 Mbit/s à 2.4GHz et 5GHz, compatible avec b, g et a ! Mais ne ressuscite pas le 'a' car ce dernier est limité à 2 Mbits/s et ne supporte pas les techniques de modulation OFDM ou MIMO.

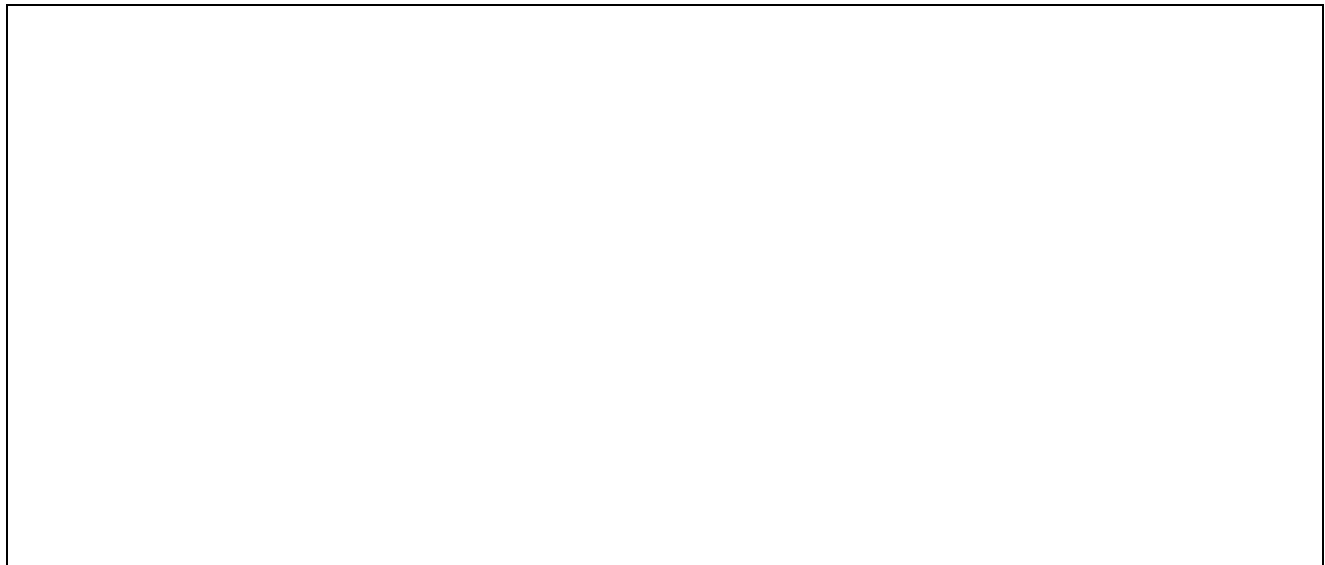
Aujourd'hui les équipements WiFi ac compatible 2.4GHz et 5GHz annoncent des débits théoriques de 300 Mbit/s avec une portée de près de 100m (Outdoor).

Les routeurs et Point d'Accès WiFi récents n'autorisent plus le mode de sécurité WEP, seul le WPA et mieux le WPA2 sont autorisés et pour cause l'usage du WEP ne se justifie plus, même avec un codage sur 256 bits.



Les Box (Orange, SFR, Bbox, Free..) et les équipements WiFi (b, g, n) en général, offrent la possibilité de choisir un canal. Que signifient ces canaux WiFi, et en quoi peuvent-ils m'être utiles ?

Ils sont espacés de 5MHz, ont une bande passante de +/-11 dB et s'étalent sur 11 canaux dans la majorité des pays (jusqu'à 14 au Japon).



## Le WiFi et la santé

**2.4GHz ou 5GHz quelle différence ? Que choisir ?**

**Intérêts du 5 GHz par rapports au 2.4 GHz (interférences, interaction rayonnement matière biologique, portée des liaisons, débit).**

**NB : les fours micro-onde opèrent sur 2.45GHz et atteignent plusieurs centaines de Watts.**

**L'OFDM et le MIMO (WiFi n et ac) sont des techniques implémentées par défaut dans la bande des 5GHz, limitant les effets de chevauchement des canaux et permettant d'accroître les débits.**

**Les téléphones portables sont en fait des récepteur/émetteur radios qui échangent des données via des ondes avec les antennes-relais installées à travers le territoire. Ces ondes ont une fréquence qui varie en fonction de la technologie utilisée :**

- **900 ou 1800 MHz pour le réseau GSM (c'est-à-dire la 2G) ;**
- **2100 MHz pour le réseau UMTS (c'est-à-dire la 3G) ;**
- **800 MHz ou 2,6 GHz pour le réseau LTE (c'est-à-dire la 4G)**

**Les puissances d'émission autorisées s'en déduisent :**

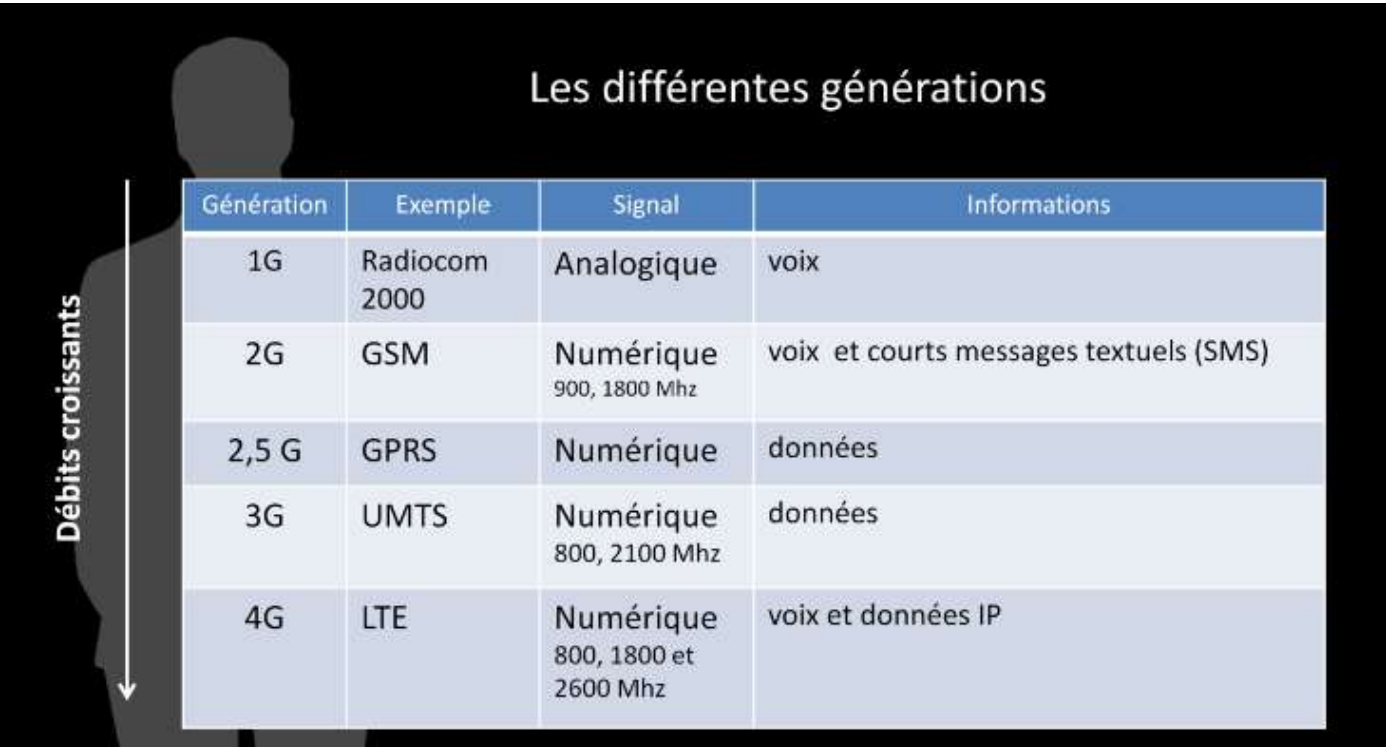
- **2 W pour le réseau GSM 900 MHz ;**
- **1 W pour le réseau GSM 1800 MHz ;**
- **0,125 W pour le réseau UMTS 2100 MHz ;**

Les téléphones portables adaptent automatiquement leur puissance d'émission en fonction des conditions de réception. Pour le réseau GSM et UMTS, le téléphone émet d'abord au maximum, puis régule fréquemment sa puissance en fonction des niveaux de réception.

Pourquoi un smart phone pourrait émettre entre 10 et 20 fois plus de puissance qu'une Box Internet domestique (plus éloignée de la tête que le smart phone) ?

Réponse : la question est mal formulée, faut être plus précis !!

## Evolution des communications sans fil



Les différentes générations

Génération	Exemple	Signal	Informations
1G	Radiocom 2000	Analogique	voix
2G	GSM	Numérique 900, 1800 Mhz	voix et courts messages textuels (SMS)
2,5 G	GPRS	Numérique	données
3G	UMTS	Numérique 800, 2100 Mhz	données
4G	LTE	Numérique 800, 1800 et 2600 Mhz	voix et données IP

Si les premiers réseaux mobiles ont permis de transporter la parole, ils ont rapidement évolué pour transporter des données. Les opérateurs ont alors interconnecté leur réseau voix GSM à commutation de circuits à leur réseau de données.

Mais ce sont les réseaux de troisième génération 3G, comme l'UMTS, qui vont offrir de nouveaux services comme l'accès à Internet, la lecture de vidéos, la TV en ligne, ou encore la visiophonie.

Enfin, le réseau de 4<sup>ème</sup> génération 4G, le LTE (Long Terme Evolution) diffère des générations précédentes du fait qu'il est "tout IP" et présente une architecture nouvelle.

Les opérateurs le déploient depuis 2010 environ, d'abord pour les données et accès Internet. Ce réseau haut débit permet des connexions théoriques jusqu'à 100 Mb/s.

**5G → le GiGa bits/s voire plus ?**

