Les réseaux informatiques, les modèles en couches avec leurs protocoles associés

Les parties surlignées sont à consulter en priorité

Sommaire

1. Qu'est-ce qu'un réseau ? 3

2. Modèles en couches OSI et TCP/IP 3

2.1 Modèle théorique de fonctionnement OSI 3

2.2 Modèle TCP/IP ou DOD 3

2.3 Les protocoles utilisés dans le modèle TCP/IP (et OSI) 4

3. L'architecture réseau 4

3.1 Client/Serveur 4

3.2 Peer to peer (P2P) 5

Les Protocoles de communication 5

4. Protocole Ethernet (OSI → couche n°2, TCP/IP → couche n°1) 6

4.1 Fonction du protocole Ethernet 6

4.2 Rôle du Switch (commutateur) dans un réseau LAN 6

5. Protocole IP, adressage logique et routage (OSI → couche n°3, TCP/IP → couche n°2) 7

5.1 Fonction et utilité du Protocole IP (Internet Protocol) 7

5.2 L'adresse IP 7

5.2.1 Autopsie des adresses IP (v4), masques de sous réseau et classes 7

5.2.2 En-tête d'encapsulation IP 9

5.3 Qu'est-ce que le routage IP 9

5.3.1 Types de routage 10

Routage statique 10

Routage dynamique 10

5.4 Rôle des routeurs au sein des réseaux 11

6. Le Protocole DHCP (Couche Application : OSI → couche 7, TCP/IP → couche 4) 12

6.1 Introduction 12

6.2 Rôle du Protocole DHCP 12

6.3 Fonctionnement du Protocole DHCP 12

7. Le protocole ARP (OSI → couche n°3, TCP/IP → couche n°2) 13

7.1 Objectif du protocole ARP (Address Resolution Protocol). 13

7.2 Principe de fonctionnement du protocole ARP. 14

8. Le Protocole ICMP (OSI → couche n°3, TCP/IP → couche n°2) 14

8.1 Introduction 14

8.2 ECHO / ECHO RESPONSE : La commande PING ! 15

9. Les protocoles de transport TCP et UDP (OSI → couche n°4, TCP/IP → couche n°3) 16

9.1 Introduction 16

9.2 Notion de PORTS logiciels, et de Socket 16

9.2.1 Ports logiciels 16

9.2.2 Socket 16

9.3 Modes de connexion et de transport 17

9.3.1 Le mode non-connecté (Protocole UDP) 17

9.3.2 Le mode Connecté (Protocole TCP) 17

10. Le protocole DNS (Couche Application : OSI → couche 7, TCP/IP → couche 4) 20

10.1 Principe de recherche d'une adresse IP à partir d'un nom de domaine 20

10.2 Comment la box connait-elle l'adresse d'un serveur DNS ? 22

10.3 Le cache DNS de l'ordinateur 22

11. Le Protocole HTTP (Couche Application : OSI → couche 7, TCP/IP → couche 4) 23

11.1 Rôle et utilité 23

11.2 Chronologie d'une requête HTTP 23

11.3 Les méthodes de commande au serveur 23

12. Le protocole FTP (Couche Application : OSI → couche 7, TCP/IP → couche 4) 24

12.1 Introduction 24

12.2 Accès à un serveur FTP 24

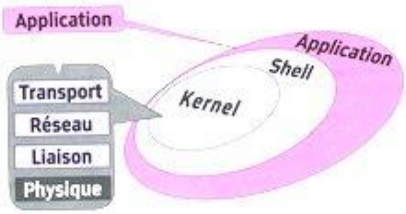
12.3 Mode de fonctionnement 24

1. Qu'est-ce qu'un réseau ?

Définition : Un réseau informatique est un ensemble de machines (ordinateurs, imprimantes, etc…) connectés ensemble. Il permet l'échange et la communication de données entre machines.

Les systèmes d'exploitation (windows, Linux, MACOSX,…) doivent pouvoir gérer la transmission des données au sein du réseau. Mais rien n'oblige les plateformes client et serveur à fonctionner avec le même système d'exploitation.

Exemple : Des serveurs sous Linux doivent pouvoir dialoguer avec des smartphones (IOS, Android), des PC sous Windows, ...



Pour cela, deux modèles théoriques (**OSI**, ou bien **TCP/IP**) décrivent comment le transfert des informations à travers le réseau doit être construit.

Ces modèles OSI et TCP/IP utilisent des **protocoles** bien définis (HTTP, FTP, TCP, UDP, IP, ICMP, ARP, …) afin de communiquer entre les machines du réseau.

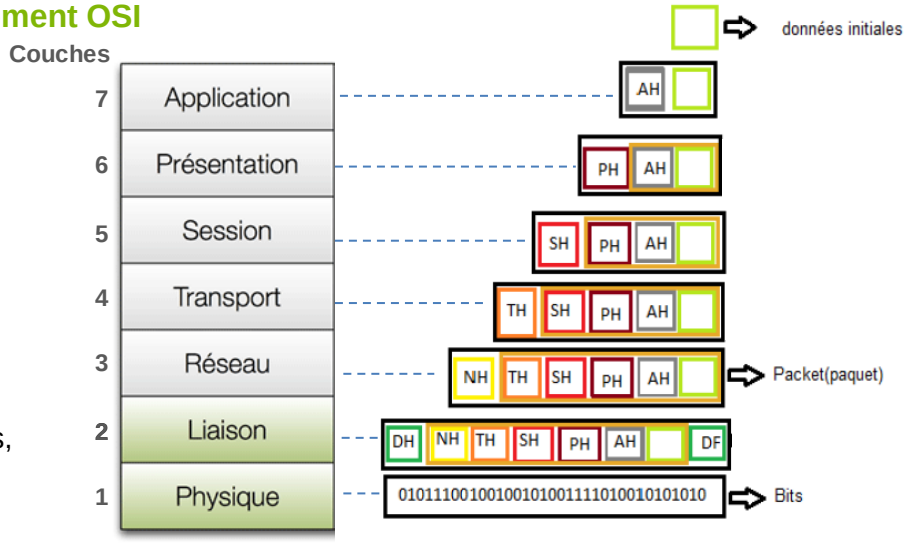
2. Modèles en couches OSI et TCP/IP

2.1 Modèle théorique de fonctionnement OSI

Afin de normaliser les protocoles, l'International Standard Organisation a développé le modèle OSI (**O**pen **S**ystem **I**nterconnections). Il permet de bien distinguer les fonctions d'un système de communication.

Ce modèle divise en **7 couches** les fonctions d'un système de communication.

Cependant il n'est pas indispensable de disposer de toutes les couches dans un système : selon les fonctionnalités requises, certaines couches intermédiaires sont inutiles.



2.2 Modèle TCP/IP ou DOD

Le modèle TCP/IP (DOD) ne comporte que **4 couches**, en cohérence avec le modèle OSI.

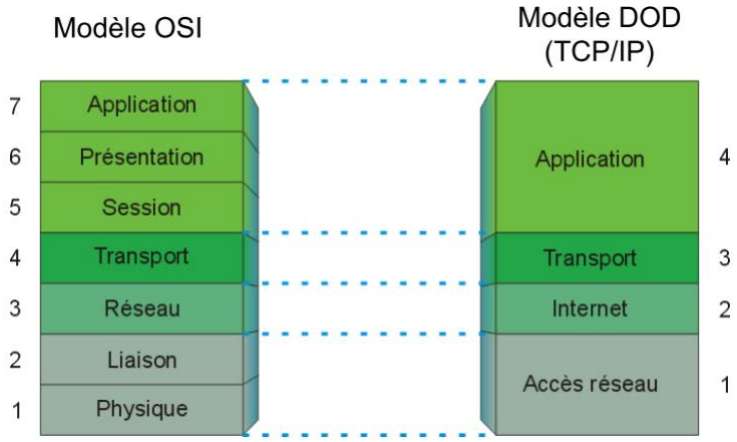
Rôle des 4 couches du modèle TCP/IP :

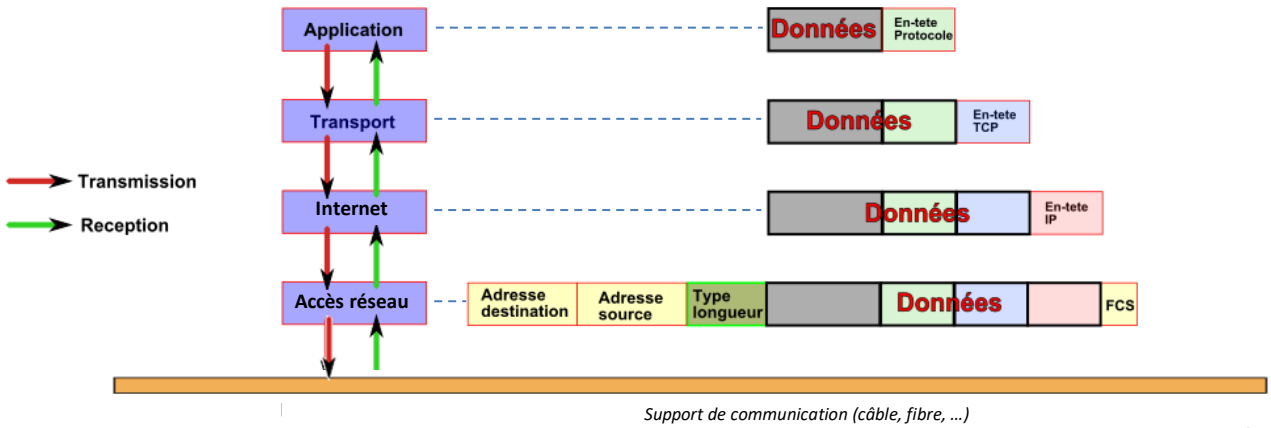
Application : Interface utilisateur/réseau

Transport : Assure le transport des données et la gestion des erreurs.

Internet : Gère les adresses et le routage des données.

Accès réseau : Interface avec la carte réseau.



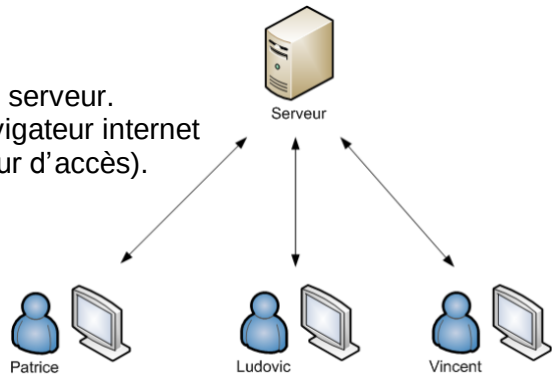


2.3 Les protocoles utilisés dans le modèle TCP/IP (et OSI)

<table>
<tr><th>Couche</th><th>Rôle</th><th colspan="2">Les protocoles et les services Internet</th></tr>
<tr><td rowspan="4">Application</td><td rowspan="4"><u>Couche application</u>
Elle regroupe tous les services qui font le succès d'Internet</td><td colspan="2">FTP
(File Transport Protocol)
échange direct de fichiers</td></tr>
<tr><td colspan="2">HTTP
(HyperText Transfer Protocol)
échange de pages Web au format HTML</td></tr>
<tr><td colspan="2">Telnet
connexion d'un client (affichage + clavier) à un ordinateur distant</td></tr>
<tr><td colspan="2">SMTP, POP
(Simple Network Management Protocole ; Post Office Protocol)
courrier électronique</td></tr>
<tr><td>Transport</td><td><u>Couche transport</u>
Elle gère les échanges entre applications indépendamment du réseau utilisé</td><td>TCP
(Transport Control Protocol)
transmission fiable par flot* en mode connecté à la machine distante.</td><td>UDP
(User Datagram Protocol)
transmission par messages, sans connexion donc sans fiabilité.</td></tr>
<tr><td rowspan="3">Internet</td><td rowspan="3"><u>Couche Internet</u>
Elle envoie les informations d'une machine à une autre toutes les deux identifiées par une adresse Internet (adresse IP)</td><td colspan="2">IP (Internet Protocol)
▪ Il constitue les paquets de données appelés datagrammes.
▪ Il effectue leur fragmentation à l'émission pour que leur taille soit compatible avec les trames du réseau physique et leur assemblage à la réception.
▪ Il gère les adresses IP qui identifient une machine dans le réseau mondial.</td></tr>
<tr><td colspan="2">ICMP (Internet Control Message Protocol)
détecte les erreurs de transmission dues aux machines.
Protocole utilisé par les routeurs qui l'utilisent pour signaler une erreur</td></tr>
<tr><td colspan="2">ARP (Address Resolution Protocol)
assure la correspondance entre l'adresse physique unique d'une carte réseau et une adresse logique universelle (adresse IP) utilisée par Internet pour identifier les machines du réseau</td></tr>
<tr><td>Accès au réseau</td><td><u>Couche d'accès au réseau</u></td><td colspan="2">Permet d'accéder à un réseau physique quel qu'il soit (Ethernet, Token Ring, connexion à une ligne téléphonique …)</td></tr>
</table>

3. L'architecture réseau

3.1 Client/Serveur



Chaque machine sur le réseau est considérée comme un client ou un serveur. Chaque logiciel client peut envoyer des requêtes à un serveur (ex : navigateur internet à un serveur web ; logiciel de messagerie au serveur mail du fournisseur d'accès).

Un serveur peut être spécialisé en serveur d'applications, de fichiers, ou encore de messagerie électronique.

<u>En résumé :</u> Le client pose une question (ou donne un ordre)… et le serveur répond à la question (ou obéit).

- **<u>Le client</u>**

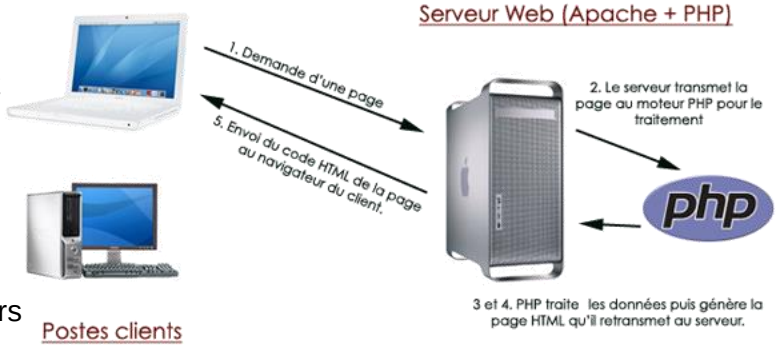
  Les caractéristiques d'un client sont les suivantes : il est d'abord actif (ou maître), il envoie des requêtes au serveur, il attend et reçoit les réponses du serveur.

- **<u>Le serveur</u>**

  Un serveur est initialement passif, il attend, il est à l'écoute, prêt à répondre aux requêtes envoyées par des clients. Dès qu'une requête lui parvient, il la traite et envoie une réponse.

- **<u>le dialogue</u>**

  Le client et le serveur doivent bien sûr utiliser le même protocole de communication. Un serveur est généralement capable de servir plusieurs clients simultanément.

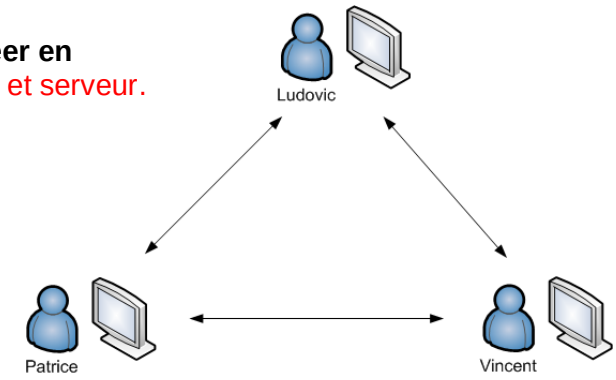


<u>Remarques :</u>

Une fois le client traité, le serveur peut en traiter un autre. Il existe des serveurs multi clients comme les serveurs Web/http qui sont capables de traiter plusieurs clients en même temps.

3.2 Peer to peer (P2P)

Un autre type d'architecture réseau est le **pair-à-pair (peer to peer en anglais ou P2P)**, dans lequel chaque ordinateur est à la fois client et serveur.



Les réseaux P2P permettent à plusieurs ordinateurs de communiquer et partager simplement des fichiers le plus souvent, mais également des flux multimédia comme par exemple la téléphonie avec Skype.

L'utilisation d'un réseau P2P nécessite pour chaque ordinateur l'utilisation d'un logiciel particulier. Ce logiciel, qui remplit alors à la fois les fonctions de **client et de serveur**, est parfois appelé **servent** (de la contraction de « serveur » et de « client »).

Exemples d'applications P2P : eDonkey, eMule, FastTrack (utilisé par KaZaA) ou BitTorrent...

Avantages *(serveur centralisé)*

- Toutes les données sont centralisées sur un seul serveur, on a donc « un contrôle de sécurité simplifié ».
- Les technologies supportant l'architecture client/serveur sont plus anciennes et matures que les autres.
- L'administration se porte au niveau serveur. Toute la complexité/puissance peut être déportée sur le(s) serveur(s), les utilisateurs utilisant simplement un client léger.
- Les serveurs étant centralisés, cette architecture est particulièrement adaptée et suffisamment véloce pour retrouver et comparer de vastes quantités d'informations (moteur de recherche sur le web).

Inconvénients *(serveur centralisé)*

- Si trop de clients veulent communiquer avec le serveur en même temps, ce dernier ne supporte pas la charge.
- Si le serveur n'est plus disponible, plus aucun des clients ne fonctionne (le réseau pair à pair continue à fonctionner, même si plusieurs participants quittent le réseau).
- Les coûts de mise en place et de maintenance sont élevés.
- Les clients ne peuvent communiquer entre eux, entraînant une asymétrie de l'information au profit des serveurs.

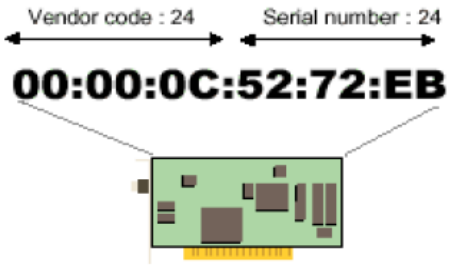
Les Protocoles de communication

C'est quoi un protocole ?

C'est un mode opératoire de communication connu et commun aux machines sur le réseau. Il n'y a pas de communication possible sans avoir recours aux différents protocoles (ARP, IP, TCP, FTP, HTTP, IP, …) des modèles en couche OSI et TCP/IP.

L'adresse MAC

Chaque carte réseau est identifiée par une **adresse physique nommée adresse MAC** : Media Access Control.



Codée sur 48 bits, elle est **UNIQUE** au monde et désigne un poste de travail de manière absolue (en théorie) contrairement à l'adresse IP.

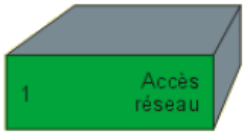
Les 3 premiers octets sont réservés à l'identification du fabricant de carte, les 3 derniers à l'identificateur de la carte.

Exemples d'adresses MAC :

00-00-0C-xxx CISCO	00-00-0E-xxx Fujitsu	00-00-0F-xxx NEXT
00-00-10-xxx SYTEK	00-00-15-xxx DataPoint Co.	00-00-1B-xxx Novell

4. Protocole Ethernet (OSI → couche n°2, TCP/IP → couche n°1)

4.1 Fonction du protocole Ethernet



Ethernet CSMA/CD

C'est le protocole de plus bas niveau sur le réseau, il correspond aux deux premières couches du modèle OSI. Il assure la bonne gestion du médium (détection de collisions) et permet l'acheminement des informations entre émetteur et destinataire au niveau des adresses MAC.

Les données à transmettre sont ainsi encapsulées dans une série d'information contenant notamment :

- ✓ l'adresse MAC de destination
- ✓ l'adresse MAC de l'émetteur
- ✓ Le type de protocole de la couche supérieure (exemples : 0x0800 → IP ; 0x0806 → ARP)
- ✓ FCS (Frame control sequence) qui contient un CRC (Cyclic Redundancy Check) : somme de contrôle. Ce CRC appliqué à la trame permet de garantir son intégrité.

En octets

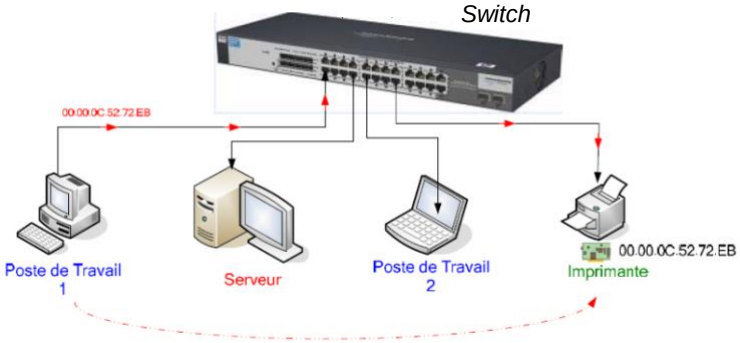
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14 … 1513	1514	1515	1516	1517
Adresse MAC destination						Adresse MAC source						Type de protocole		Données	FCS/CRC			

Trame Ethernet

4.2 Rôle du Switch (commutateur) dans un réseau LAN

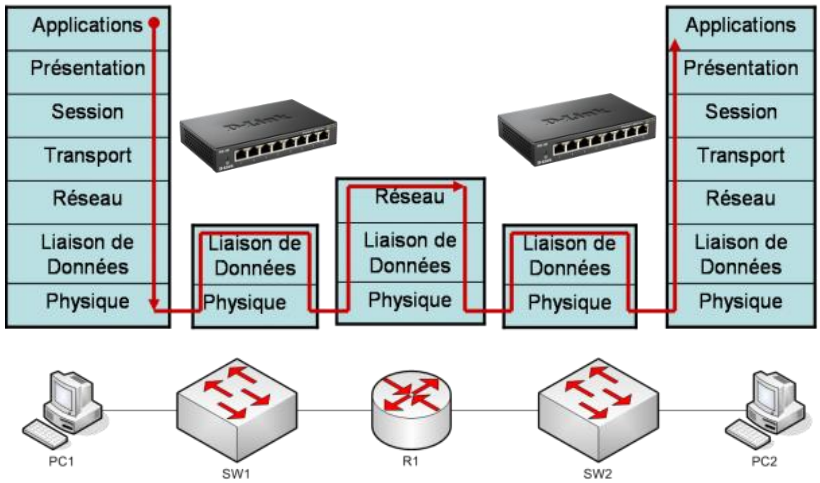
Symbole

Switch



Les Switch (commutateurs) dirigent les informations à transmettre directement sur le port où est connecté le poste destinataire concerné. Cela est possible grâce aux adresses MAC de destination et de l'émetteur contenues dans **la trame Ethernet**.

Mais, avant cela, le switch doit «apprendre» progressivement où se trouvent les postes du réseau en **remplissant une table de correspondance MAC/Port**.



En effet, quand un message lui parvient, **le switch associe le port par lequel arrive le message, à l'adresse MAC de l'émetteur du message**, s'il ne la connaît pas encore.

Lorsqu'un PC ne connait pas l'adresse MAC de destination, il émet sur le réseau une **requête ARP** (*voir §6*) **en broadcast** afin que le poste visé lui réponde. Au passage le switch met à jour sa table Mac/Port.

Ainsi, après un certain nombre de messages, le commutateur connaît «l'emplacement» des postes sur le réseau et peut les mettre en relation directement.

Il existe une adresse MAC permettant au Switch de communiquer l'information à l'ensemble de ses ports donc à toutes les stations : c'est l'adresse de broadcast (FF:FF:FF:FF:FF:FF).

Certains modèles permettent également de séparer plusieurs réseaux, en créant virtuellement une barrière entre eux. On appelle cela **un VLAN** (Virtual Local Area Network) ou Réseau Local Privé.

5. Protocole IP, adressage logique et routage (OSI → couche n°3, TCP/IP → couche n°2)

5.1 Fonction et utilité du Protocole IP (Internet Protocol)

Le protocole IP est l'un des plus importants sur internet, il assure les fonctions d'**adressage** (**routage**) et éventuellement de la **fragmentation** des données à transmettre sur le réseau.

Modèle en couches TCP/IP

Il achemine des paquets de données à travers un ensemble de réseaux en les transférant d'un routeur à l'autre jusqu'à atteindre une adresse de destination appelée **adresse IP.**

Le protocole IP est considéré comme « non fiable ». Il n'offre aucune garantie sur :

- La corruption de données.
- L'ordre d'arrivée des paquets (un paquet A peut être envoyé avant un paquet B, mais arriver après).
- La perte, la destruction ou la duplication de paquets.

5.2 L'adresse IP

L'ensemble des postes informatiques du monde entier ne peut être contenu dans une table MAC/port d'un switch. Cela représenterait des dizaines de millions d'adresses !!

De plus l'adresse MAC d'une carte réseau correspond à l'adresse d'un poste et d'un seul. Or les postes sont généralement regroupés en réseau. Comment identifier le réseau auquel appartient le poste, et comment regrouper facilement un ensemble de postes dans un réseau ?

On utilise alors l'adresse IP qui est un autre type d'adressage, l'adressage logique.

5.2.1 Autopsie des adresses IP (v4), masques de sous réseau et classes

Codage d'une adresse IP:

Une adresse IP est un nombre de 32 bits codé sur 4 octets séparés par des points (1 octet = 8 bits). Cette adresse IP est bien souvent affichée/utilisée sous forme décimale, c'est-à-dire une succession de quatre nombres compris entre 0 et 255.

Exemple : 212.217.0.1

Parfois il est indispensable d'écrire cette adresse IP sous forme binaire.

Exemple : notation décimale 212 . 217 . 0 . 1

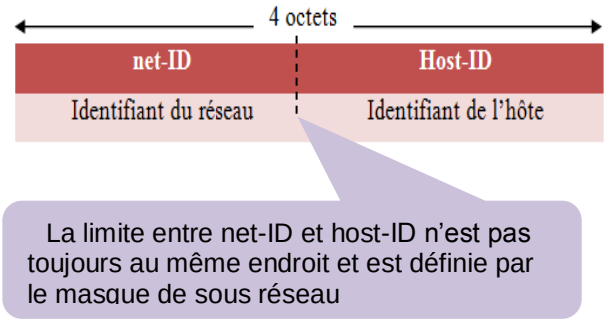
Notation binaire 1101 0100 . 1101 1001 . 0000 0000 . 0000 0001

Composition d'une adresse IP:

Une adresse IP est composée de deux parties bien distinctes :

- Identificateur réseau : net-ID située à gauche, elle désigne le réseau contenant les ordinateurs.
- Identificateur de l'hôte : host-ID située à droite et désignant les ordinateurs de ce réseau.

Au sein d'un même sous réseau, seule la partie host-ID peut changer.



Le masque de sous-réseau :

Un masque de sous réseau utilise la même notation qu'une adresse IP. Il est donc composé de 4 nombres compris entre 0 et 255 séparés par des points. Toutefois il est obligatoirement composé en binaire d'une suite ininterrompue de 1 suivie d'une suite ininterrompue de 0.

- La partie composée de 1 du masque de sous réseau définie la partie net-ID des adresses IP du réseau.
- La partie composée de 0 du masque de sous réseau définie la partie host-ID des adresses IP du réseau.

Exemple :

Le masque de sous réseau 255.255.0.0 s'écrit en binaire :

1111 1111 . 1111 1111 . 0000 0000 . 0000 0000

Les bits à 1 représentent la partie réseau de l'adresse IP

Les bits à 0 représentent la partie « host » de l'adresse

Autre exemple :

Soit l'adresse IP : 192.168.1.23 en binaire : 1100 0000 . 1010 1000 . 0000 0001 . 0001 0111

et son masque associé : 255.255.255.0 en binaire : 1111 1111 . 1111 1111 . 1111 1111 . 0000 0000

Séparation entre la partie ID réseau et la partie « host » de l'adresse IP

Deux adresses IP particulières pour ce sous réseau :

- La partie host entièrement à 0 : en binaire : 1100 0000 . 1010 1000 . 0000 0001 . 0000 0000
  soit en décimale : 192 . 168 . 1 . 0

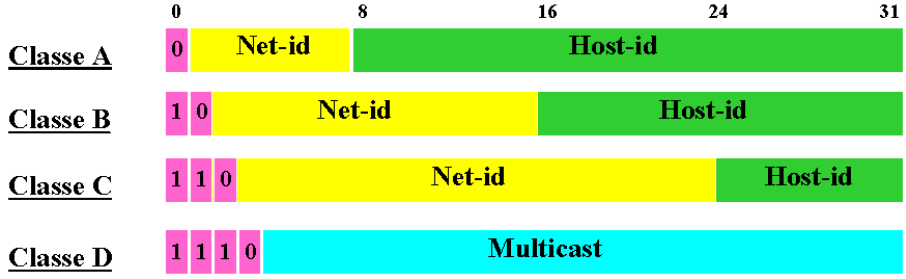
  Cette adresse est appelée adresse réseau, elle ne peut être attribuée à un hôte.

- La partie host entièrement à 1 : en binaire : 1100 0000 . 1010 1000 . 0000 0001 . 1111 1111
  soit en décimale : 192 . 168 . 1 . 255

  Cette adresse est appelée adresse de broadcast, elle est utilisée pour envoyer un message à toutes les machines du réseau et ne peut être attribuée à un hôte.

Les classes d'adresses IP :

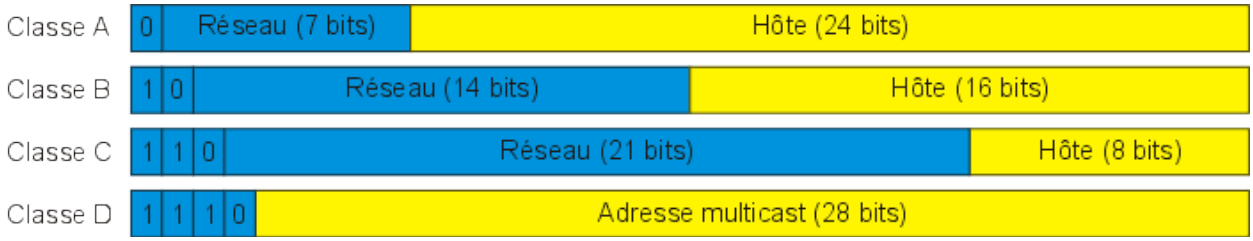
Hormis la classe D multicast, destinée à faire de la diffusion d'information pour plusieurs hôtes simultanément, il existe trois classes d'adresses IP:



La classe A permet de créer peu de réseaux, mais avec beaucoup d'hôtes (de machines) dans chaque réseau, La classe C faisant l'inverse.

Voici la règle pour identifier l'étendue de chaque classe :

- La classe est définie par les bits les plus lourds (les plus à gauche)
- La classe A est signalée par un seul bit, un 0
- La classe B par deux bits, donc 1 0
- La classe C par trois bits, donc 1 1 0
- La classe D (multicast) par 4 bits donc 1 1 1 0



Ce qui nous donne en valeur d'adressage IP et en masque de sous réseau par défaut :

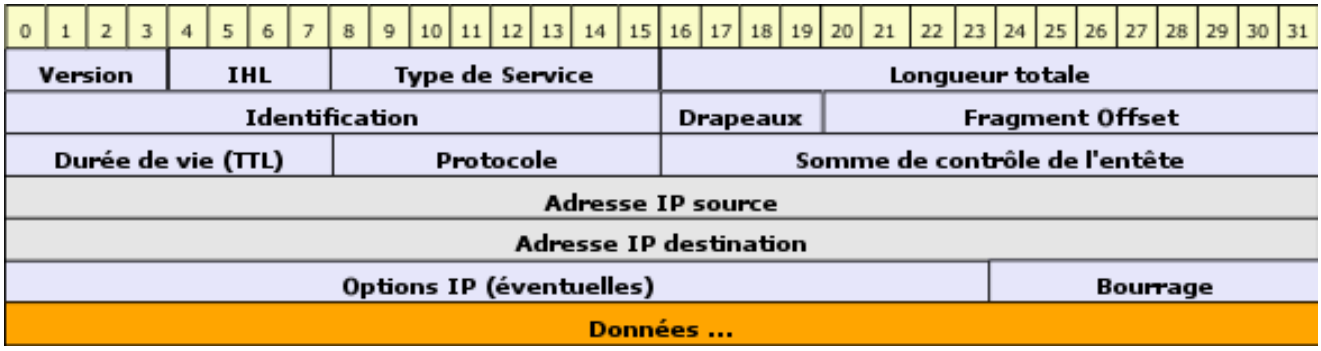
Classe	Première adresse	Dernière adresse
A	0.0.0.1	127.255.255.254
B	128.0.0.1	191.255.255.254
C	192.0.0.1	223.255.255.254
D	224.0.0.1	239.255.255.254

Classe	Masque par défaut	Octets	Nb Machines
A	255.0.0.0	3	16 777 214
B	255.255.0.0	2	65 534
C	255.255.255.0	1	254

5.2.2 En-tête d'encapsulation IP

Les paquets de données circulant sur le net sont encapsulés par les différents protocoles, c'est-à-dire que l'on a ajouté des en-têtes aux données initiales à transmettre. Ces en-têtes contiennent dans le cas du protocole IP des informations sur le transport.

Structure de l'en-tête d'encapsulation



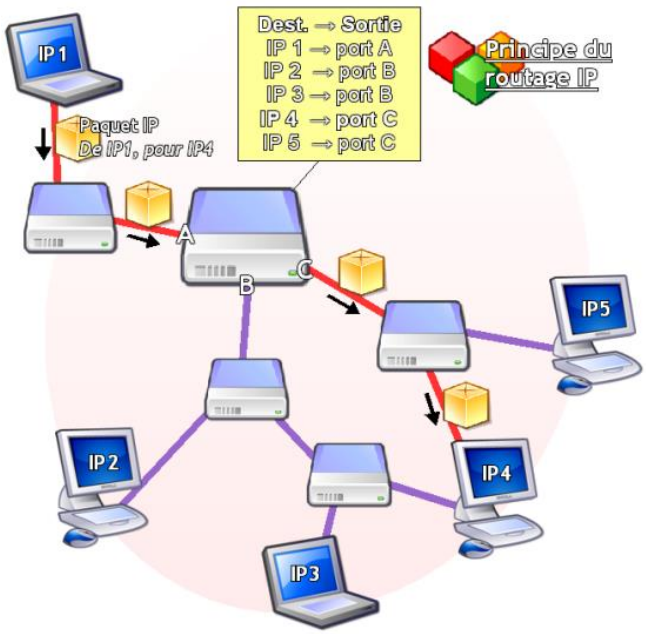
Signification des différents champs :

- **Version** (4 bits) : version du protocole IP que l'on utilise (IPv4 ou IPv6)
- **Longueur d'en-tête**, ou *IHL* pour *Internet Header Length* (4 bits) : nombre de mots de 32 bits constituant l'en-tête (nota : la valeur minimale est 5).
- **Type de service** (8 bits) : indique la façon selon laquelle le datagramme doit être traité.
- **Longueur totale** (16 bits): indique la taille totale du datagramme en octets. La taille de ce champ étant de 2 octets, la taille totale du datagramme ne peut dépasser 65536 octets. Utilisé conjointement avec la taille de l'en-tête, ce champ permet de déterminer où sont situées les données.
- **Identification, drapeaux (flags) et déplacement de fragment** sont des champs qui permettent le réassemblage des datagrammes lorsqu'ils ont été fragmentés pour s'adapter au MTU du réseau.
- **Durée de vie** appelée aussi **TTL**, pour *Time To Live* (8 bits) : ce champ indique le nombre maximal de routeurs à travers lesquels le datagramme peut passer. Ainsi ce champ est décrémenté à chaque passage dans un routeur, lorsque celui-ci atteint la valeur critique de 0, le routeur détruit le datagramme. Cela évite l'encombrement du réseau par les datagrammes perdus.
- **Protocole** (8 bits) : ce champ, en notation décimale, permet de savoir de quel protocole est issu le datagramme (ICMP : 1 ou IGMP : 2 ou TCP : 6 ou UDP : 17)
- **Somme de contrôle de l'en-tête, ou en anglais *header checksum* (16 bits)** : ce champ contient une valeur codée sur 16 bits qui permet de contrôler l'intégrité de l'en-tête afin de déterminer si celui-ci n'a pas été altéré pendant la transmission.
- **Adresse IP source** (32 bits) : adresse IP de la machine émettrice, il permet au destinataire de répondre.
- **Adresse IP destination** (32 bits) : adresse IP du destinataire du message.

5.3 Qu'est-ce que le routage IP

Lorsqu'un ordinateur émet un message vers un ordinateur situé sur un réseau différent, il transmet le message à "son" routeur (dit passerelle par défaut), qui à son tour le fait suivre à un autre routeur et ainsi de suite jusqu'à atteindre l'hôte de destination.

Que ce soit l'ordinateur émetteur ou tous les routeurs intermédiaires, tous consultent leur **table de routage** pour savoir à qui transmettre le paquet d'informations.



Pour accéder à la table de routage d'un poste :

- Sous Windows (dans cmd) : ***route print***
- Sous Linux (console) : ***route -n***

Pour accéder à la table de routage d'un routeur :

- Voir notice constructeur

Exemple : Table de routage d'un ordinateur sous Windows 7

Chemin par défaut si l'adresse de destination souhaitée n'est pas connue

```
C:\Users\sen>route print
===========================================================================
Liste d'Interfaces
 11...74 f0 6d 5d da 16 ......Atheros AR2427 Wireless Network Adapter
 10...20 cf 30 6d 50 c1 ......Atheros AR8132 PCI-E Fast Ethernet Controller (N
S 6.20)
  1...........................Software Loopback Interface 1
 14...00 00 00 00 00 00 00 e0 Carte Microsoft ISATAP
 13...00 00 00 00 00 00 00 e0 Carte Microsoft ISATAP #2
 12...00 00 00 00 00 00 00 e0 Teredo Tunneling Pseudo-Interface
===========================================================================

IPv4 Table de routage
===========================================================================
Itinéraires actifs :
Destination réseau    Masque réseau  Adr. passerelle   Adr. interface Métrique
          0.0.0.0          0.0.0.0   192.168.8.254     192.168.8.16     20
        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
      192.168.8.0    255.255.255.0         On-link       192.168.8.16    276
     192.168.8.16  255.255.255.255         On-link       192.168.8.16    276
    192.168.8.255  255.255.255.255         On-link       192.168.8.16    276
        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       192.168.8.16    276
  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       192.168.8.16    276
===========================================================================
Itinéraires persistants :
  Aucun
```

Voici les rubriques que l'on peut trouver :

Destination et masque réseau : *Réseaux à atteindre*

Adresse passerelle : *Adresse du routeur pour accéder au réseau spécifié sur la même ligne par destination et masque*

Adresse interface : *Carte réseau à utiliser*

Métrique : *Coût relatif de l'itinéraire pour atteindre la destination*

Voir aussi : http://openclassrooms.com/courses/apprenez-le-fonctionnement-des-reseaux-tcp-ip/le-routage-1

5.3.1 Types de routage

Routage statique

La table de routage est établie au démarrage de la machine, l'administrateur peut ajouter des routes manuellement. Ce type de routage est à utiliser dans les petits réseaux. L'inconvénient principal est que si une route est défectueuse, le routeur continue à vouloir l'utiliser.

Routage dynamique

Dès que le réseau atteint une certaine taille (avec plusieurs routeurs), il est nécessaire de mettre en œuvre un routage dynamique. Les protocoles de routage dynamique sont utilisés par les routeurs pour partager des informations sur l'accessibilité et l'état des réseaux distants.

Les protocoles de routage dynamique effectuent plusieurs tâches, notamment :

- Détection de réseaux.
- Mise à jour et maintenance des tables de routage.

Détection automatique de réseaux

Concrètement, les routeurs s'échangent leurs tables de routage et établissent un « meilleur chemin » s'il en existe plusieurs. Ce meilleur chemin dépend du protocole utilisé (voir plus loin).

Maintenance des tables de routage

Après la découverte initiale des réseaux, les protocoles de routage dynamique les mettent à jour et les gèrent dans leurs tables de routage. Les protocoles de routage dynamique déterminent également un nouveau meilleur chemin si le chemin initial devient inutilisable (ou si la topologie change).

Protocoles de routage IP

Il existe plusieurs protocoles de routage dynamique IP : (RIP, IGRP, EIGRP, OSPF, IS-IS, BGP)

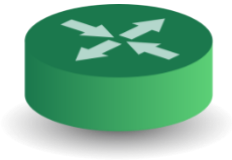
Meilleur chemin (métrique).

La détermination du meilleur chemin d'un routeur implique d'évaluer plusieurs chemins menant au même réseau de destination et de choisir le chemin optimal pour atteindre ce réseau.
Le meilleur chemin est sélectionné par le protocole de routage, qui utilise une métrique pour déterminer la distance à parcourir pour atteindre un réseau.

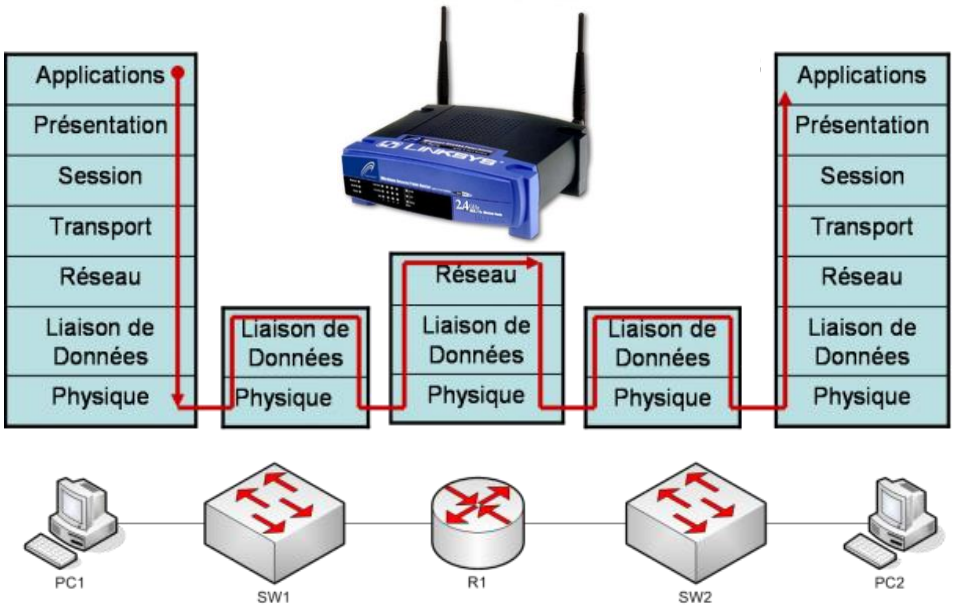
La métrique est une mesure de la « distance » qui sépare un routeur d'un réseau de destination, elle peut correspondre :

- Au nombre de sauts IP nécessaires pour atteindre le réseau destination (RIP)
- A un coût numérique qui dépend de la bande passante des liens franchis (OSPF)
- Ou au résultat d'un calcul plus complexe, qui tient compte de la charge, du délai, du MTU, ...

5.4 Rôle des routeurs au sein des réseaux

Symbole : 

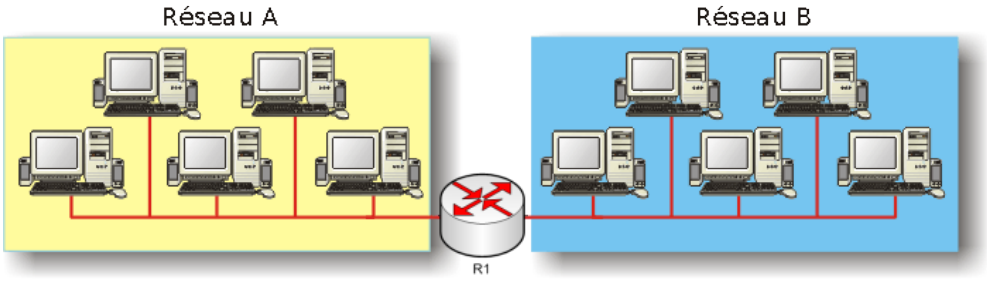
Les routeurs opèrent au niveau de la couche 3 du modèle OSI (couche 2 modèle TCP/IP).



Un routeur est un équipement d'interconnexion de réseaux permettant d'assurer le routage des informations entre deux réseaux ou plus afin de déterminer le chemin qu'un paquet va emprunter.

Pour y parvenir, les routeurs tiennent à jour **leurs tables de routage**, véritable cartographie des itinéraires à suivre en fonction de l'adresse IP visée. Un routeur possède plusieurs cartes réseaux, chacune connectée sur un réseau différent. Il possède ainsi autant d'adresses IP que de réseaux différents sur lesquels il est connecté.

Il peut être identifié comme **une passerelle** entre deux réseaux :



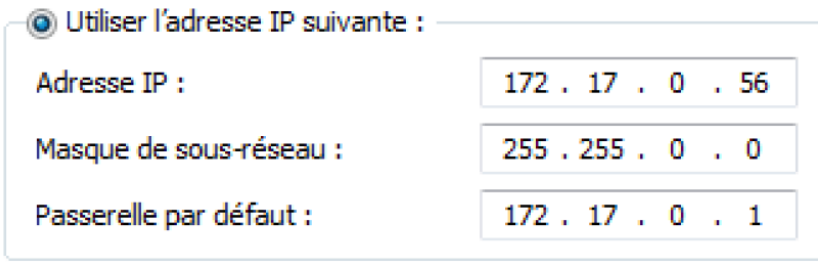
Une machine qui utilise le protocole IP ne peut communiquer directement qu'avec les postes qui sont sur le même réseau, c'est-à-dire les postes ayant la même plage d'adressage que le sien.

Exemple : *PC1 : IP : 192.168.24.3 Masque : 255.255.255.0*
PC2 : IP : 192.168.24.2 Masque : 255.255.0.0
PC3 : IP : 192.168.24.250 Masque : 255.255.255.0

Seul PC3 et PC1 peuvent communiquer ensemble sur le réseau 192.168.24.0

Pour communiquer avec des machines qui ne sont pas dans le même réseau, il est nécessaire d'indiquer l'adresse IP du routeur qui permettra de communiquer ces données vers d'autres réseaux.

Cette adresse IP doit être définie dans les paramètres du protocole IP de la carte réseau à la rubrique : "**passerelle par défaut** ".



Lorsqu'une machine voudra communiquer avec une adresse IP en dehors de son réseau, les informations transiteront directement par le routeur « passerelle par défaut » afin de sortir du réseau.

6. Le Protocole DHCP (Couche Application : OSI → couche 7, TCP/IP → couche 4)

6.1 Introduction

Sur les réseaux locaux, il n'est pas rare que les utilisateurs changent fréquemment. Les nouveaux utilisateurs arrivent avec des ordinateurs portables et ont besoin d'une connexion. D'autres disposent de nouvelles stations de travail ou d'autres périphériques réseau, tels que des smartphones, qui doivent être connectés. Plutôt que de demander à chaque fois à l'administrateur réseau d'attribuer des adresses IP à chaque station de travail, il est plus facile d'attribuer ces adresses automatiquement. Cette opération est réalisée à l'aide du protocole DHCP (**D**ynamic **H**ost **C**onfiguration **P**rotocol).

Lorsque vous connectez une machine à un réseau Ethernet TCP/IP, cette machine, pour fonctionner correctement, doit disposer :

- D'une adresse IP unique dans votre réseau et appartenant au même réseau logique que toutes les autres machines du réseau en question.
- Un masque de sous réseau, le même pour tous les hôtes du réseau.
- L'adresse de la passerelle qui vous permet justement d'accéder au Net.
- Une adresse de DNS, pour pouvoir naviguer sur internet.

6.2 Rôle du Protocole DHCP

Le protocole DHCP permet donc à un ordinateur qui se connecte sur un réseau local d'obtenir et de configurer dynamiquement et automatiquement :

- Son adresse IP
- Son masque de sous-réseau
- La passerelle par défaut
- Une adresse IP du serveur DNS
- Le nom de son domaine

Le protocole DHCP est généralement la méthode d'attribution d'adresses IPv4 privilégiée pour les réseaux de grande taille, car le personnel de support du réseau est dégagé de cette tâche et le risque d'erreur de saisie est éliminé.

La configuration du serveur DHCP (souvent le routeur, votre box) nécessite qu'une plage d'adresses, appelée ***pool d'adresses***, soit définie pour l'attribution aux clients DHCP d'un réseau.

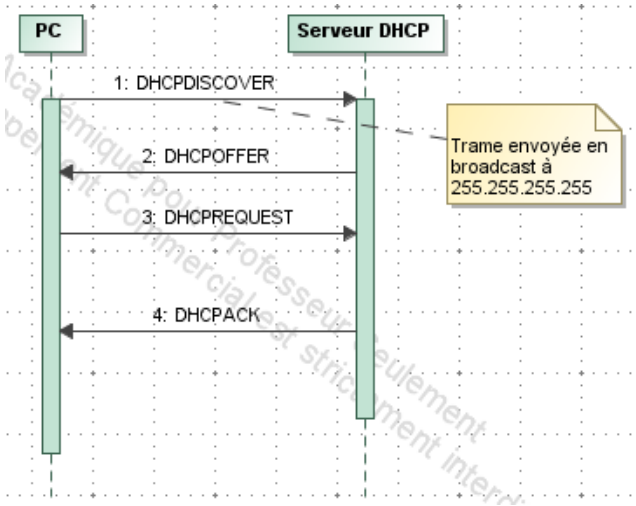
L'autre avantage de l'attribution dynamique réside dans le fait que les adresses ne sont pas permanentes pour les hôtes, elles sont uniquement « louées » pour une certaine durée (bail). Si l'hôte est mis hors tension ou retiré du réseau, l'adresse pourra être réutilisée à expiration du bail. Cela est particulièrement intéressant pour les utilisateurs mobiles qui se connectent et se déconnectent d'un réseau.

6.3 Fonctionnement du Protocole DHCP

Si un PC ne possède pas d'adresse IP fixe, il possède néanmoins :

- Une "MAC Address" qui n'est jamais perdue puisqu'elle est écrite "en dur" dans la carte réseau.
- La possibilité de "Broadcaster" c'est à dire d'envoyer des trames à toutes les machines du réseau.

Au démarrage le PC et le serveur DHCP dialoguent donc de la manière suivante:



1. Lorsque le PC démarre, il n'a aucune connaissance du réseau. Il envoie donc une trame "DHCPDISCOVER", destinée à trouver un serveur DHCP. Cette trame est un "broadcast", envoyée à l'adresse 255.255.255.255. N'ayant pas encore d'adresse IP, il adopte provisoirement l'adresse 0.0.0.0. Comme ce n'est pas avec cette adresse que le DHCP va l'identifier, il fournit aussi sa "MAC Address".
2. Le serveur DHCP du réseau reçoit cette trame et répond par un "DHCPOFFER". Cette trame contient l'adresse IP affectée au client, un bail (durée de vie de l'adresse IP). Elle contient aussi l'adresse IP du serveur.
3. Le client répond alors par un DHCPREQUEST pour indiquer qu'il accepte l'offre.
4. Le serveur DHCP répond définitivement par un DHCPACK qui constitue une confirmation du bail. L'adresse du client est alors marquée comme utilisée et ne sera plus proposée à un autre client pour toute la durée du bail.

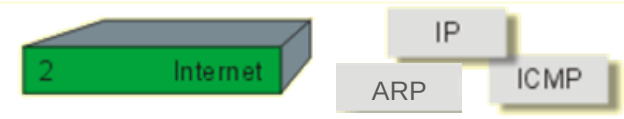
Dans un souci d'efficacité, le DHCP utilise le protocole UDP pour communiquer (voir §9.3.1).

Il est à tout moment possible de casser le bail, au moyen de la commande *ipconfig /release*, ou de renouveler le bail au moyen de la commande *ipconfig /renew*.

7. Le protocole ARP (OSI → couche n°3, TCP/IP → couche n°2)

7.1 Objectif du protocole ARP (*Address Resolution Protocol*).

Pour s'envoyer des messages dans un réseau Ethernet, les ordinateurs s'envoient des trames Ethernet. Le premier champ de la trame Ethernet est l'adresse Mac de l'ordinateur de destination (voir §4):



Modèle en couches TCP/IP

En octets

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14 … 1513	1514	1515	1516	1517
Adresse MAC destination						Adresse MAC source						Type de protocole		Données	FCS/CRC			

Trame Ethernet

Les ordinateurs doivent donc connaitre l'adresse MAC de leur destinataire !

Ils ont en mémoire un ***Cache ARP***, qui fait la correspondance **entre les adresses IP du réseau et les adresses Mac associées.**

Pour accéder cache ARP d'un poste :

- Sous windows (dans cmd) : arp *–a*
- Sous Linux (console) : arp -an

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

C:\Users\M&S>arp -a

Interface : 192.168.1.12 --- 0xb
  Adresse Internet      Adresse physique      Type
  192.168.1.1           00-26-91-80-8b-5b     dynamique
  192.168.1.10          00-21-63-f4-cc-5d     dynamique
  192.168.1.11          a0-21-b7-84-ba-76     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
  224.0.0.253           01-00-5e-00-00-fd     statique
  239.255.255.250       01-00-5e-7f-ff-fa     statique
  255.255.255.255       ff-ff-ff-ff-ff-ff     statique
```

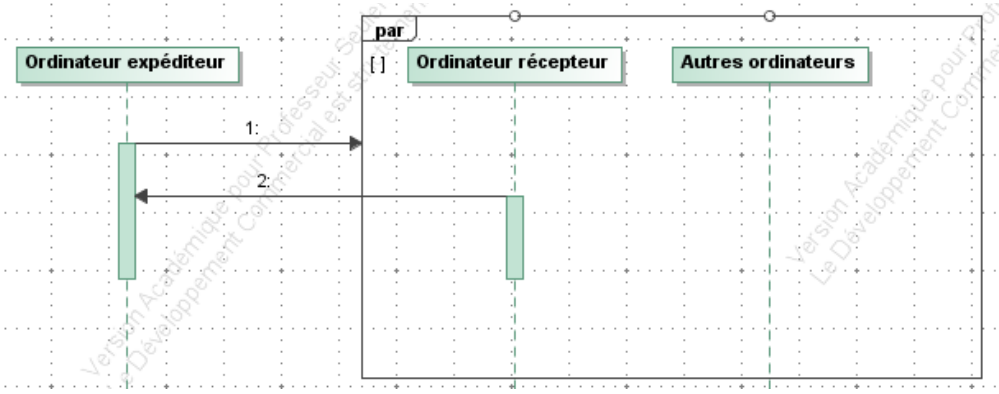
Les ordinateurs peuvent donc envoyer des trames Ethernet en ne connaissant que l'adresse IP du destinataire.

Il est souhaitable de mettre à jour ce *Cache ARP*. En effet, il faut vérifier à intervalle régulier si l'ordinateur auquel on croit envoyer des informations est le bon.

Le protocole ARP a donc pour objectif de permettre la mise à jour du cache ARP.

7.2 Principe de fonctionnement du protocole ARP.

Afin d'envoyer une trame Ethernet, un ordinateur *expéditeur* souhaite connaitre l'adresse Mac d'un ordinateur *récepteur*. Or, l'ordinateur *expéditeur* ne connait que l'adresse IP de l'ordinateur *récepteur*.



(1) L'ordinateur expéditeur envoie une requête ARP en Broadcast à tous les ordinateurs du réseau. Ce message peut se décoder ainsi : *je suis IP 192.168.1.10 d'adresse MAC Inventec_4f :06 :2d et je souhaite contacter le poste d'adresse IP 192.168.1.1 et d'adresse MAC inconnue par moi.*

```
Sender MAC address: Inventec_4f:06:2d (00:1e:33:4f:06:2d)
Sender IP address: 192.168.1.10 (192.168.1.10)
Target MAC address: 00:00:00_00:00:00 (00:00:00:00:00:00)
Target IP address: 192.168.1.1 (192.168.1.1)
```

(2) L'ordinateur récepteur qui se reconnait par son adresse IP répond à la requête ARP directement à l'expéditeur en lui indiquant son adresse MAC.

```
Sender MAC address: SagemCom_80:8b:5b (00:26:91:80:8b:5b)
Sender IP address: 192.168.1.1 (192.168.1.1)
Target MAC address: Inventec_4f:06:2d (00:1e:33:4f:06:2d)
Target IP address: 192.168.1.10 (192.168.1.10)
```

A la fin de l'échange, les deux ordinateurs ont mis à jour leur table ARP, et l'expéditeur peut envoyer sa trame Ethernet à l'ordinateur récepteur.

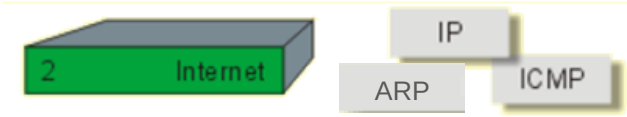
Voir aussi : http://openclassrooms.com/courses/apprenez-le-fonctionnement-des-reseaux-tcp-ip/les-autres-protocoles

8. Le Protocole ICMP (OSI → couche n°3, TCP/IP → couche n°2)

8.1 Introduction

ICMP (**I**nternet **C**ontrol **M**essage **P**rotocol) est un module obligatoire d'IP qui assure deux fonctions principales :

- Rendre compte d'un problème réseau.
- Tester l'accessibilité d'une machine.

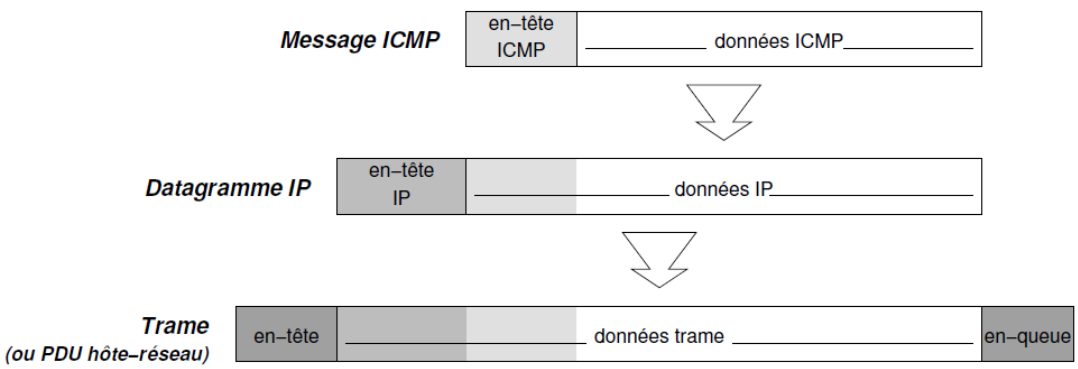


Modèle en couches TCP/IP

Les messages ICMP sont de deux natures :

- Les messages d'erreurs : suite à une erreur constatée sur un paquet (qui entraînera le plus souvent sa destruction).
- Les messages d'interrogation/information : messages divers contribuant au (ou informant sur le) bon fonctionnement des équipements.

Les messages ICMP sont encapsulés dans des paquets IP (champ Protocole vaut 1) :



Principaux paquets ICMP utilisés :

Type numérique	Nom Symbolique	Description
0	echo-reply	*Response à un ping*
3	destination-unreachable	*Message d'erreur général : l'hôte de destination n'est pas en mesure de délivrer le paquet au port de destination ou un routeur sur le chemin vers la destination n'est pas en mesure de remettre le paquet à la destination suivante. Utilisé par traceroute.*
4	source-quench	*Permet à un équipement de réseau (généralement une passerelle) de signifier à un émetteur une congestion du réseau, afin de solliciter un ralentissement de l'émission*
5	redirect	*Un message de routage retourné par l'expéditeur quand un routeur détermine qu'un chemin plus court existe.*
8	echo-request	*Requête ping.*
11	time-exceeded	*Message d'erreur typiquement retourné lorsque le TTL est dépassé ; le routeur à l'origine détruit le paquet (pour éviter les boucles dans le réseau). Utilisé par traceroute.*
12	parameter-problem	*Valeur inattendue dans l'en-tête d'un paquet IP.*

8.2 ECHO / ECHO RESPONSE : La commande PING !

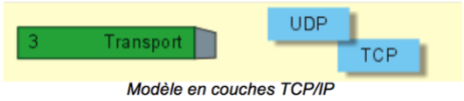
La commande PING utilise le protocole ICMP, elle permet de tester l'accessibilité d'un équipement IP.

Lorsque vous entrez la commande PING <@IP> :

1. L'émetteur créé un paquet ICMP_Echo avec quelques octets de données aléatoires. Le paquet ICMP est placé dans un paquet IP, d'adresse destination l'@IP du ping.
2. A l'émission du paquet, votre applicatif "pingueur" enclenche un timer puis il émet un nouveau paquet ICMP_Echo (en général 4 à la suite sous windows, cela peut se paramétrer).
3. Quand le récepteur reçoit les paquets ICMP_Echo, il les transfère à son programme ICMP. Celui-ci formate un paquet ICMP_Echo_Response. Le paquet ICMP est ensuite placé dans un paquet IP d'adresse destination égale à l'adresse de l'émetteur du ICMP_Echo.
4. Le paquet ICMP_Echo_Response, encapsulé dans un paquet IP, est véhiculé à travers le réseau jusqu'à l'emetteur du ping.
5. L'emetteur du ping reçoit le paquet ICMP_Echo_Response. L'applicatif "pingueur" retourne une ligne vous indiquant que le paquet a été bien reçu et la valeur du timer à la réception du paquet. Vous obtenez ainsi le délai de transit aller-retour du paquet.
6. La même opération est répétée pour les 4 paquets émis par l'expéditeur (applicatif "pingueur").

9. Les protocoles de transport TCP et UDP (OSI → couche n°4, TCP/IP → couche n°3)

9.1 Introduction



Modèle en couches TCP/IP

Nous savons désormais comment les couches 1 à 3 permettent d'acheminer correctement les paquets d'une machine à l'autre sur un réseau.

Intéressons-nous maintenant aux protocoles utilisés pour gérer l'échange de données entre applications sur deux machines d'un réseau.

Pour que deux machines communiquent entre elles au niveau logiciel (couche Applicative) il est nécessaire d'identifier un **PORT**, une adresse **IP** et un modèle de connexion (**TCP** ou **UDP**). Les informations à transmettre sont ensuite segmentées (morcelées) en paquets de données puis encapsulées et transmises soit en UDP soit en TCP.

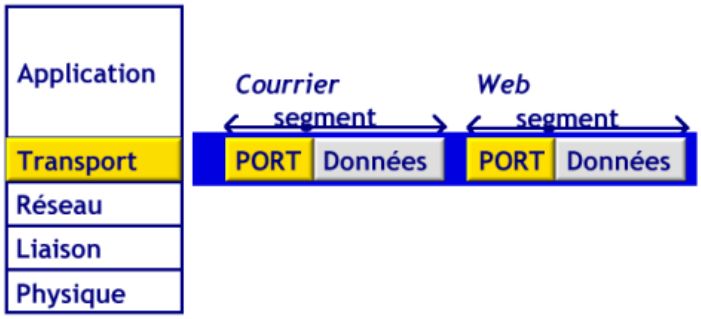
9.2 Notion de PORTS logiciels, et de Socket

9.2.1 Ports logiciels

En couches 2 et 3, nous avons vu qu'il fallait une adresse pour identifier les éléments nécessaires à l'identification des moyens de communication. L'adresse MAC identifie la carte réseau en couche 2, et l'adresse IP identifie l'adresse de notre machine au sein d'un réseau, en couche 3.

Eh bien en couche 4, **l'adresse utilisée est le port**. Le port est l'**adresse d'une application sur une machine**.

On peut aussi considérer les ports comme des portes donnant accès au système d'exploitation. Pour fonctionner, un programme (Firefox, Adobe reader, …) ouvre des portes pour entrer dans le système d'exploitation et pouvoir communiquer sur le net. Lorsque l'on quitte le programme, la porte n'a plus besoin d'être ouverte.



Un Port est un numéro codé sur 16 bits (0 à 65535). Certains ports sont fixés par défaut, exemple :

Port	Description	Port	Description	Port	Description
20	FTP – File Transfert Protocol	80	HTTP – HyperText Transfer Protocol	5900	VNC
25	SMTP - Simple Mail Transfer Protocol	110	POP3 - Post Office Protocol	25565	Minecraft
53	DNS – Domain Name Service	443	HTTPS - HyperText Transfer Protocol Secure	49300	PRONOTE

Les ports **en dessous de 1024** sont réservés et appelés "*well known ports*" (ports bien connus), les autres sont libres d'utilisation même si certains sont, par habitude, utilisés pour des applications connues comme MySQL (Port 3306).

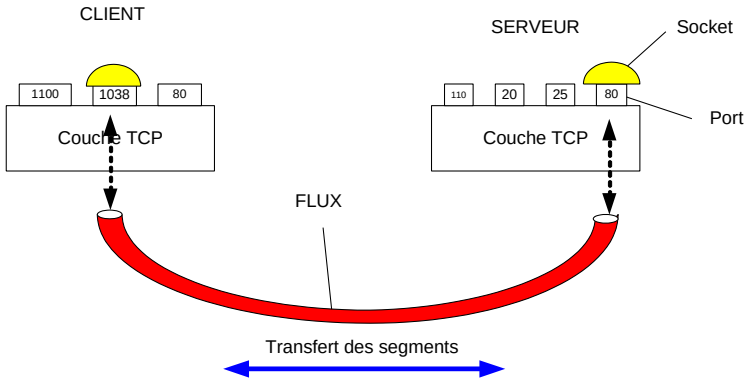
Liste d'utilisation des ports : http://www.frameip.com/liste-des-ports-tcp-udp/

9.2.2 Socket

Une socket est simplement un moyen de désigner l'extrémité d'une connexion, côté client ou serveur.

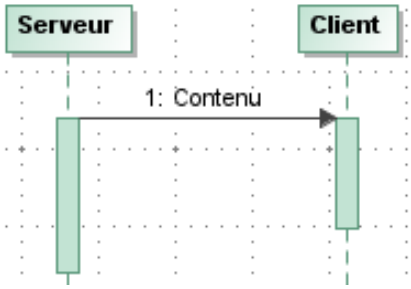
C'est une association d'une **adresse IP** et d'un **numéro de port**.

Lorsque la connexion est effectuée, la transmission des données devient un flux (tuyau entre le client et le serveur)



9.3 Modes de connexion et de transport

9.3.1 *Le mode non-connecté (Protocole UDP)*



Le protocole UDP (**U**ser **D**atagram **P**rotocol) a pour objectif de fournir une communication sans contrôle entre deux machines du réseau.

Ce protocole permet à une application d'envoyer un message à une autre application selon un mécanisme minimaliste. Il ne garantit pas la délivrance du message.

Ceci a pour inconvénient la perte possible de données. Il ne serait pas possible d'utiliser ce protocole pour transférer des fichiers texte ou exécutables par exemple.

Par contre, l'avantage est que le protocole ne surcharge pas le réseau par des mécanismes de connexion ni par des accusés de réception.

Les applications de streaming et la téléphonie par internet utilisent ce protocole car :

- Elles recherchent une bande passante maximale.
- Il est inutile de chercher à récupérer à postériori un paquet de données perdues.

Si un paquet se perd, la communication sera de moindre qualité, mais ne sera pas interrompu : c'est l'objectif recherché !

Structure de l'en-tête d'encapsulation

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	← bit
Port source																Port destination																
Longueur																Somme de contrôle de l'entête																
Données ...																																

9.3.2 *Le mode Connecté (Protocole TCP)*

Les applications nécessitant une transmission fiabilisée des données utiliseront donc de préférence le protocole TCP (**T**ransmission **C**ontrol **P**rotocol).

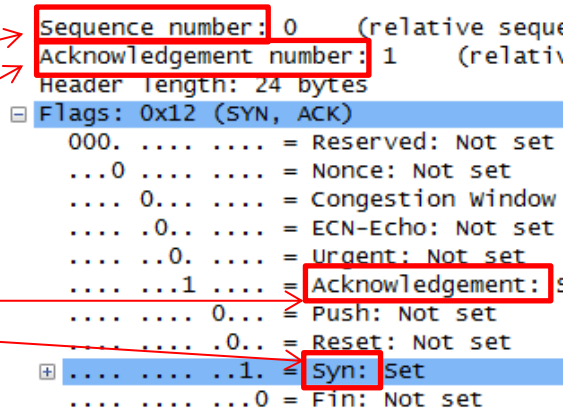
La lettre recommandée avec accusé de réception est un bon exemple du mode connecté. Si l'émetteur reçoit l'accusé réception, alors il est certain que sa lettre est arrivée à destination.

9.3.2.1 *Initialisation de la communication (Principe du « three ways handshake »)*

A l'initialisation de la communication, il se met en place un processus de « poignée de main en 3 temps » entre le **Client** et le **Serveur**. Ce processus leur permet de se synchroniser et d'établir un dialogue à propos du transfert des données.

Pour cela, deux types d'objets :

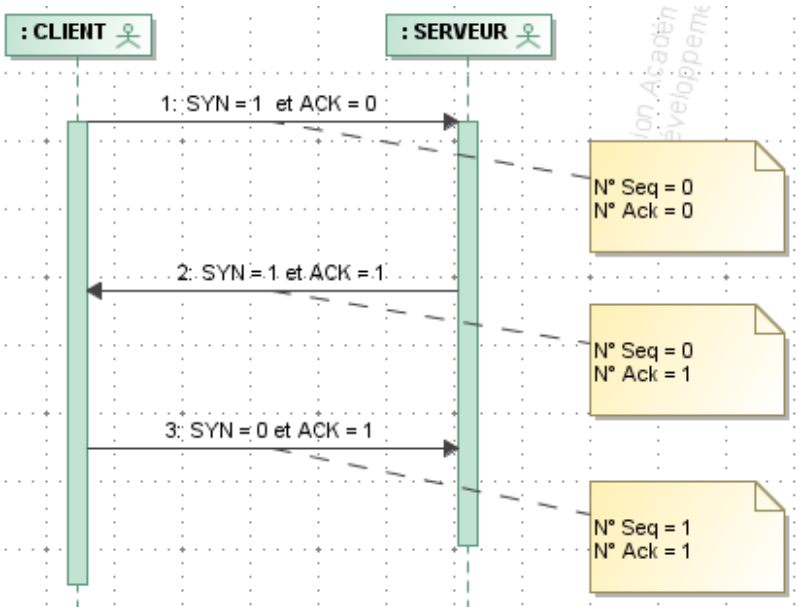
- Les numéros :
  - numéro de séquence : c'est le numéro de la trame.
  - numéro d'accusé de réception.
- Les drapeaux, qui peuvent être à l'état 0 ou 1 :
  - ACK : accusé de réception ;
  - SYN : demande de synchronisation.



Lorsque le drapeau SYN est à 1, il s'agit d'une demande de synchronisation.
Lorsque le drapeau ACK est à 1, il s'agit d'un accusé de réception.

Le «***three ways handshake*** » qui permet d'initier la communication en 3 temps se déroule comme suit :

1. Le client transmet une trame dont :
   - le drapeau SYN est à 1 (il s'agit d'un segment de synchronisation) et le drapeau ACK est à 0
   - les numéros de séquence et d'acquittement sont à 0 car rien n'a encore été échangé
2. Le serveur reçoit le message provenant du client, puis lui envoie un accusé de réception, c'est-à-dire une trame dont :
   - le drapeau ACK est à 1 (accusé de réception de la demande) et le drapeau SYN est à 1 (car on est encore dans la synchronisation)
   - le numéro d'ordre est 0 et le numéro d'accusé de réception est 1 : le serveur accuse la réception de la trame précédente



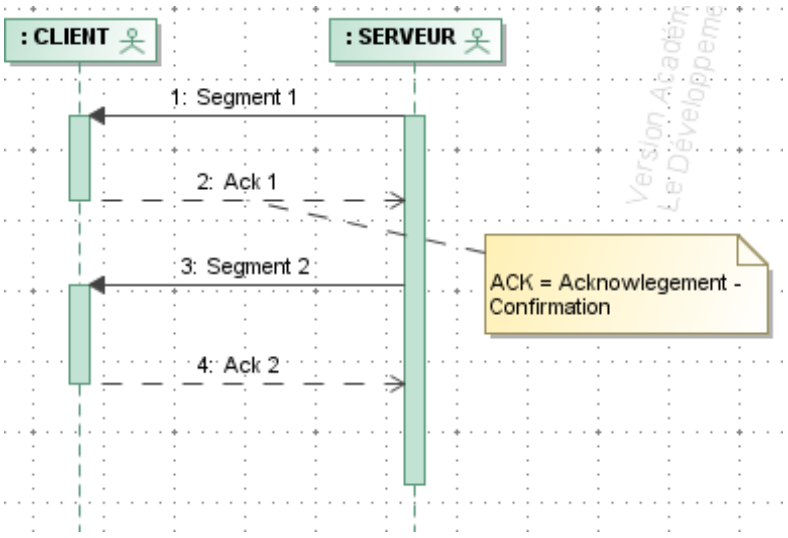
3. Le client transmet au serveur un « accusé de réception de l'accusé de réception », c'est-à-dire une trame dont :
   - le drapeau ACK est à 1, dont le drapeau SYN est à zéro (il ne s'agit plus d'un segment de synchronisation)
   - le numéro d'ordre et le numéro d'accusé de réception sont égaux à 1

9.3.2.2 Envoi de trames en mode « une par une » (en théorie)

Une fois que la communication a été initialisée, les trames peuvent être envoyées. Une première solution peut être d'envoyer les trames une par une.

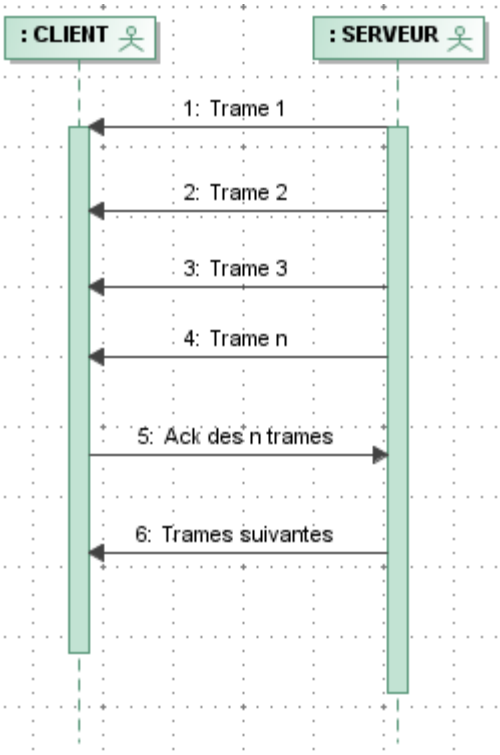
Lors de l'émission d'une trame, cette trame porte en elle son numéro, appelé *numéro de séquence*.

A réception d'une trame de données, le client retourne une trame dont le drapeau ACK est à 1 (afin de signaler qu'il s'agit d'un accusé de réception) accompagné d'un numéro d'accusé de réception égal au numéro de la trame reçue.



En pratique, ce mode n'est jamais utilisé, car il serait beaucoup trop long d'accuser réception de chaque trame.

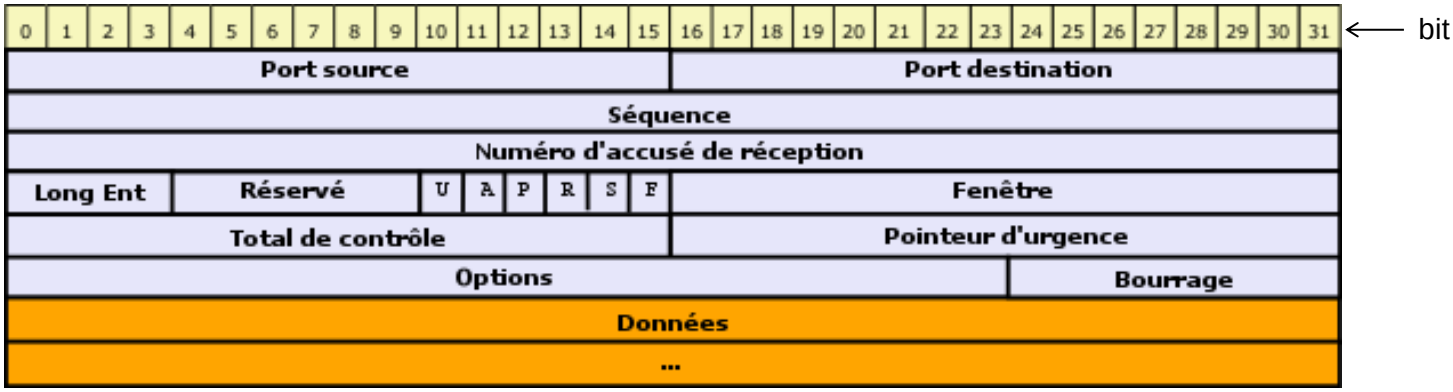
9.3.2.3 *Envoi de plusieurs trames (dans la réalité)*



En pratique, le protocole TCP utilise la technique du Sliding Windows (fenêtre coulissante).

La fenêtre définit le volume de données en octets susceptibles d'être passées via une connexion TCP, avant que le récepteur n'envoie un accusé de réception.

9.3.2.4 *Structure de l'en-tête d'encapsulation des données*



Signification des différents champs :

- **Port source (16 bits).** C'est le port utilisé pour les données à émettre.
- **Port destination (16 bits).** C'est le port où les données sont envoyées.
- **Numéro de séquence (32 bits).** Il donne la position du segment dans le flux de l'émetteur. Deux cas sont à considérer :
  - Le bit SYN est positionné à 1, alors le numéro de séquence a pour valeur *ISN* [*Initial Sequence Number*] + 1.
  - Le bit SYN est positionné à 0, alors le numéro de séquence a pour valeur le numéro du premier octet de données relativement au début de la transmission.
- **Numéro d'accusé de réception (32 bits).** Il indique le numéro du prochain octet attendu par le récepteur.
- **Longueur en-tête (4 bits).** Il indique la longueur de l'en-tête d'un segment TCP et est exprimé comme un multiple de 32 bits. Ce champ est rendu indispensable dans la mesure où la longueur du champ option est variable (selon les options choisies).
- **Réservé (6 bits).** Comme son nom l'indique, il est réservé à un usage futur. Il est donc positionné à zéro.
- **Bits de code TCP :** (U→Urgent ; A→Acknowledgment ; P→Push ; R→Restart ; S→ Synchronisation ; F→Fin)
- **Fenêtre (16 bits).** Indique le nombre d'octet que le récepteur peut admettre (à partir de la position contenu dans l'accusé de réception) sans qu'un accusé de réception soit nécessaire.
- **Total de contrôle (16 bits).** Ce champ permet de vérifier l'intégrité de l'en-tête TCP et des données. C'est le complément à 1 (sur 16 bits) de la somme des compléments à 1 des octets de l'en-tête et des données (par mots de 32 bits). A noter que le champ de 16 bits le représentant est positionné à 0 lors du calcul.
- **Pointeur d'urgence (16 bits).** Ce champ utilisé si le bit *URG* est positionné (à 1) indique dans la fenêtre la position où les données urgentes s'arrêtent.
- **Options (variable).** Ce champ contient les différentes options TCP. Par exemple, le *MSS* ([*Maximum Segment Size*], taille maximale des segments), le *window scale option*, le *timestamp option* ...
- **Bourrage (variable).** Permet de parvenir à un en-tête d'une taille multiple de 32 bits. Il complète par des 0 la fin du champ *options*.
- **Données (variable).** Il s'agit des données à transmettre.

10. Le protocole DNS (Couche Application : OSI → couche 7, TCP/IP → couche 4)

Dans le monde de l'Internet, les machines du réseau sont identifiées par des adresses IP. Néanmoins, ces adresses ne sont pas très agréables à manipuler.

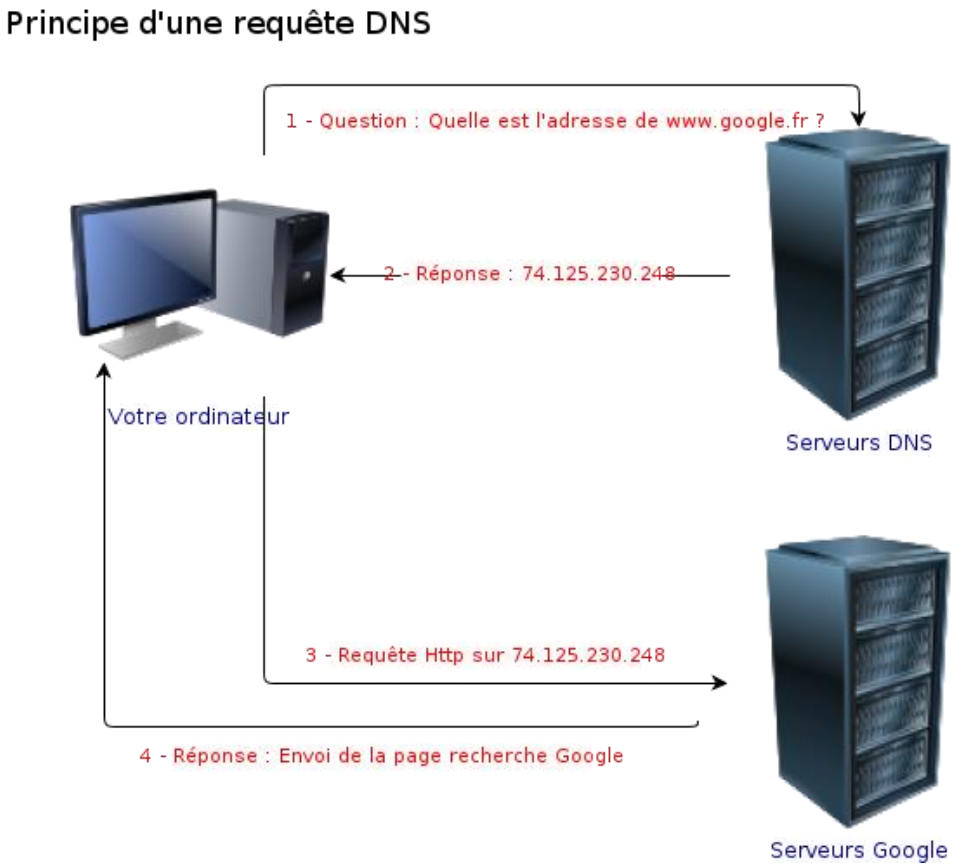
Par exemple, il ne serait pas pratique de devoir retenir l'adresse 74.125.24.94 pour effectuer une recherche sur le net avec Google.

C'est pourquoi, à la place de l'adresse IP d'un site, on utilise son nom de domaine (dans notre exemple : www.google.fr).

Ainsi, il est nécessaire d'associer des noms en langage courant aux adresses IP. Le protocole DNS (**D**omain **N**ame **S**ystem) se charge de faire cette correspondance.

On appelle résolution de noms de domaines (ou résolution d'adresses) le lien entre les adresses IP et le nom de domaine associé.

10.1 Principe de recherche d'une adresse IP à partir d'un nom de domaine



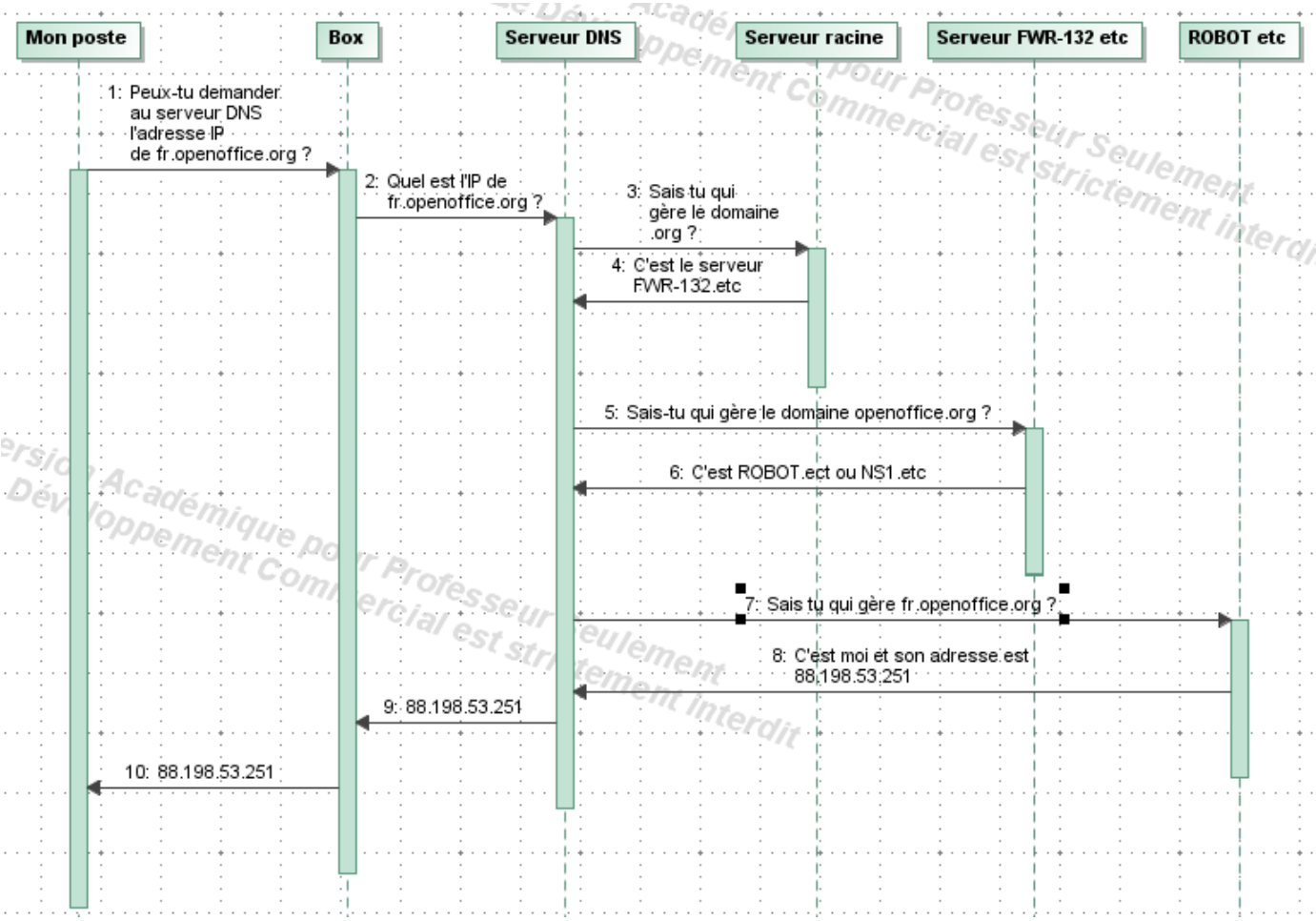
Pour rechercher l'adresse IP d'un nom de domaine, votre ordinateur émet une requête DNS auprès d'un serveur DNS. Il existe plusieurs types de serveurs DNS :

- Les serveurs DNS *récursifs* : un serveur récursif est le serveur que va interroger votre ordinateur pour rechercher l'adresse IP correspondant à un nom de domaine. Ce serveur récursif va lui-même se charger de trouver l'adresse IP du site, et répondre à votre ordinateur.
- Les serveurs DNS *racine* : ce sont les serveurs DNS à la racine d'internet. Ils connaissent tous les sous domaines d'internet. Ce sont eux que le serveur DNS récursif va interroger en premier pour la recherche d'une adresse IP.
  Ces serveurs racine sont *physiquement* au nombre de 13, notés de A à M. Ils sont dupliqués *logiquement* à travers le monde. La localisation des serveurs est donnée par le site : http://www.root-servers.org/

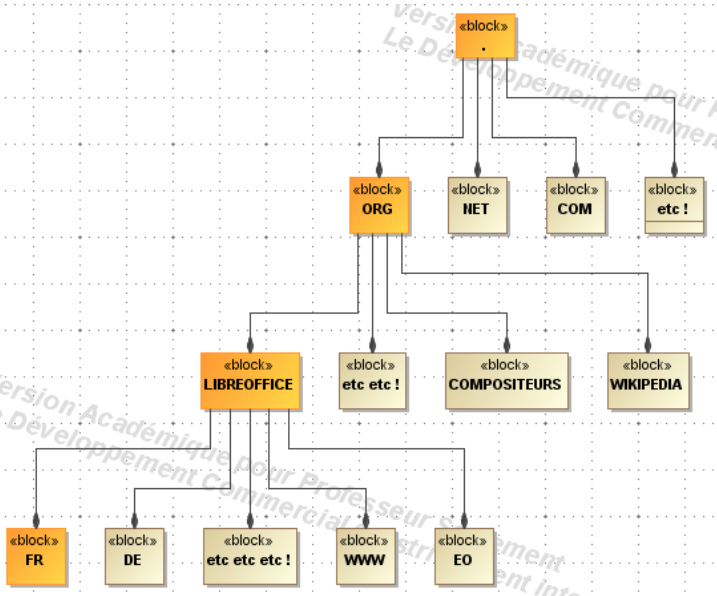
Sans ces serveurs, l'internet ne pourrait plus fonctionner : attaque des serveurs DYN

Chaque domaine et sous domaine possède son ou ses serveurs de DNS.

Exemple : Requête DNS de votre ordinateur pour obtenir l'adresse IP associée à http://fr.openoffice.org :



Organisation des domaines et sous-domaines sur le net :

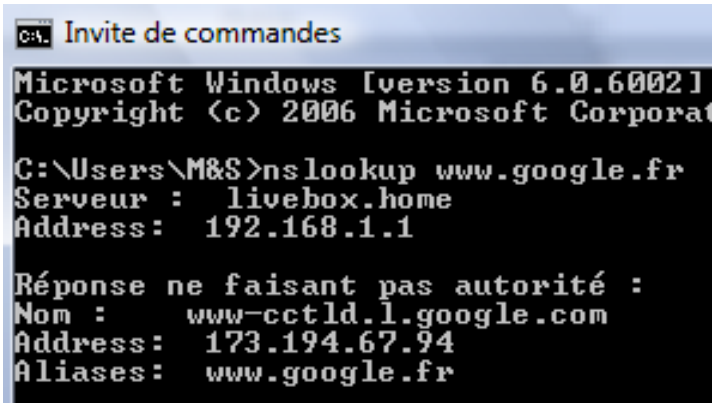


Dans un souci d'efficacité, le Protocole DNS utilise le protocole UDP pour communiquer.

10.2 Comment la box connait-elle l'adresse d'un serveur DNS ?

Des adresses IP de serveurs DNS récursifs sont paramétrés dans la box (serveur DNS primaire, secondaire,…). Lors d'une requête DNS, la box interroge donc d'abord le DNS primaire, puis, si celui-ci est indisponible, elle interroge le DNS secondaire.

Il est possible de faire une requête DNS manuellement, en utilisant la commande ***nslookup*.**



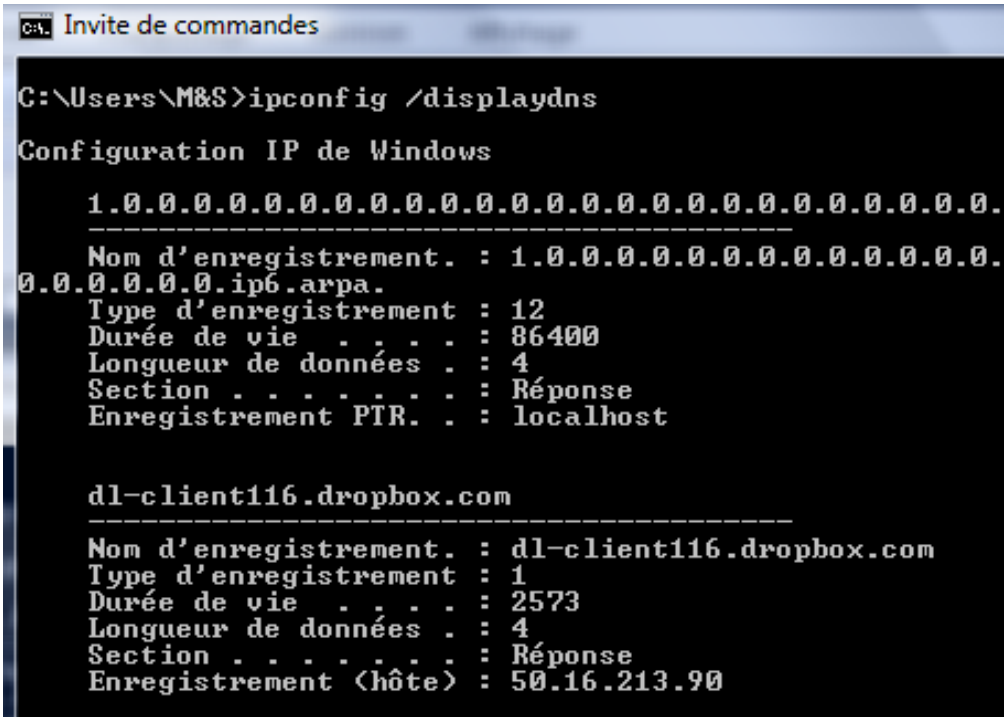
L'adresse IP de www.google.fr renvoyée est donc 173.194.67.94

10.3 Le cache DNS de l'ordinateur

Votre ordinateur ne réalise pas de requête DNS à chaque fois que vous cherchez www.google.fr. Il garde l'adresse dans une mémoire, appelée le Cache DNS afin d'économiser de la ressource.

Chaque adresse possède son *Bail*, c'est-à-dire le temps (en secondes) avant lequel une nouvelle requête DNS devra être réalisée.

Il est possible de visualiser son cache DNS au moyen de la commande ***ipconfig / displaydns*** :



11. Le Protocole HTTP (Couche Application : OSI → couche 7, TCP/IP → couche 4)

11.1 Rôle et utilité

Le protocole **HTTP** (Hyper Text Transfert Protocol) est un protocole destiné à transférer des pages web. Il permet de transférer, depuis un serveur vers un client (navigateur internet) : du texte, des illustrations, ou des fichiers quelconques.

Rappel sur l'URL (*Uniform Resource Locator*):

Une page web est définie par son URL c'est-à-dire une adresse web, par exemple:

http://www.google.fr/index.html

Vous pouvez distinguer 3 parties dans cet URL :

- **http** : indique quel protocole vous allez employer.
- **//www.google.fr** : représente le serveur web
- **/index.html** : représente le chemin de la page demandée dans l'arborescence du serveur concerné.

Pourtant, si vous indiquez comme URL http://www.google.fr , il manque la partie correspondant au nom de la page. Cependant, nous arrivons bien à visualiser la page google... Tout simplement, parce que le serveur est paramétré pour ajouter " /index.html " s'il n'y a pas le nom de la page.

Ainsi, http://www.google.fr/index.html est équivalent à http://www.google.fr.

Attention, ne pas confondre HTTP et HTML. HTML (Hyper Text Markup Language) est le langage permettant d'élaborer le contenu des pages web. Même si HTTP et HTML sont intimement liés, il s'agit bien de deux choses différentes.

Les clients (navigateurs Firefox, IE, Chrome …) se connectent à des serveurs http tels qu'**Apache http Server** ou **IIS** (**Internet Information Services**).

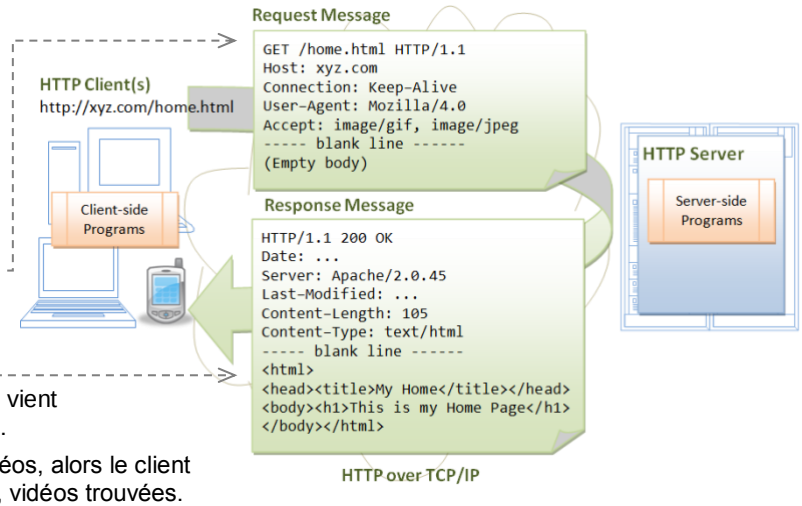
Remarque :

HTTPS (avec S pour Secured ou Sécurisé) est la variante de http sécurisée par l'usage du protocole **SSL** (**S**ecure **S**ockets **L**ayer) ou nouvellement par son successeur le protocole **TLS** (**T**ransport **L**ayer **S**ecurity**).**

Ces protocoles sécurisent les échanges sur internet.

11.2 Chronologie d'une requête HTTP

Que se passe-t-il lors d'une requête client pour afficher une page web (ici la page web http://xyz.com/home.html):



1. Le serveur possède un port en écoute (par défaut le port 80).
2. Le client http se connecte et de synchronise au port en écoute du serveur (synchronisation TCP).
3. Le client transmet une requête à l'aide d'une commande GET par exemple.
4. Le serveur transmet au client la réponse à sa requête avec le contenu html de la page web.
5. Le navigateur internet du client scanne le contenu html qui vient de lui être envoyé à la recherche d'URL d'images, vidéos...
6. Si le scan du contenu html contient des URL d'images, vidéos, alors le client transmet autant de commandes GET que d'URL d'images, vidéos trouvées.
7. Le serveur transmet au client autant de réponses que de requêtes client, avec comme contenu dans chacune de ses réponses une image ou une vidéo.

11.3 Les méthodes de commande au serveur

Dans le protocole http, une méthode est un type de requête, c'est-à-dire qu'elle demande au serveur d'effectuer une action. Les méthodes les plus utilisées sont GET et POST :

- GET : c'est la méthode la plus courante pour demander une ressource. Une requête GET est sans effet sur la ressource.
- POST : cette méthode doit être utilisée pour ajouter une nouvelle ressource, comme un message sur un forum, un article dans un site ou encore un login et un mot de passe.

Lien vers les codes HTTP : http://www.codeshttp.com/

12. Le protocole FTP (Couche Application : OSI → couche 7, TCP/IP → couche 4)

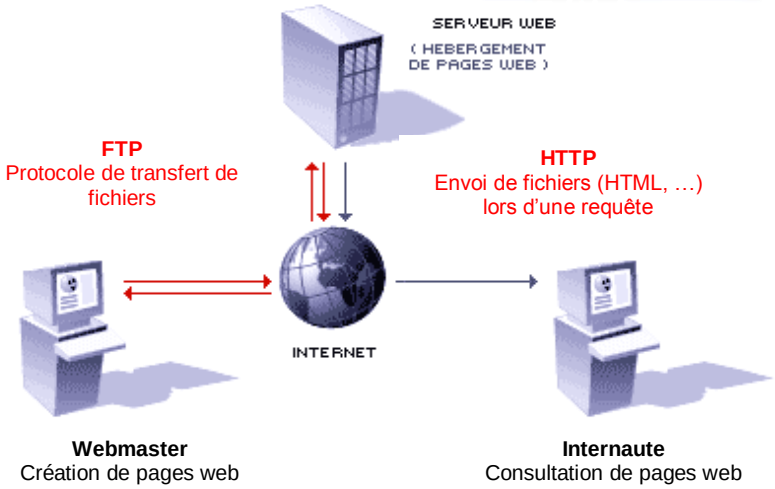
12.1 Introduction

Le protocole de transfert de fichiers **FTP** (**F**ile **T**ransfer **P**rotocol) est un protocole de communication destiné à l'échange de fichiers sur un réseau. Il permet, depuis un serveur, de copier des fichiers vers un ordinateur client du réseau, d'alimenter un site web sur un serveur, …

FTP obéit à un modèle client/serveur, c'est-à-dire qu'une des deux parties, le client, envoie des requêtes et le serveur répond.

En pratique, le serveur FTP est un ordinateur sur lequel fonctionne un logiciel serveur FTP.

La variante de FTP protégée par les protocoles SSL ou TLS (TLS étant le successeur de SSL) s'appelle FTPS.



12.2 Accès à un serveur FTP

Lors d'une connexion sur un serveur FTP vous ne verrez qu'une liste de fichiers hébergés sur le serveur auquel vous vous connectez, comme si vous ouvriez un explorateur de fichiers. Les fichiers sont accessibles directement en brut sur le serveur.

Un serveur FTP requiert une identification du client. Il existe souvent un compte "anonyme", qui donne accès en lecture seule dans la partie publique du serveur, mais il existe également des parties privées où les clients disposant d'un compte peuvent accéder en écriture sur certains répertoires de l'arborescence.

Pour accéder à un serveur FTP, on peut soit :

- Utiliser un logiciel client FTP (Filezilla par exemple : http://filezilla-project.org/) possédant une interface graphique et conviviale.
- Utiliser un navigateur web (Firefox, Internet explorer, Google Chrome…)
- Taper des lignes de commande DOS.

12.3 Mode de fonctionnement

Durant une connexion FTP, deux canaux de transmission sont ouverts :

- Un canal de contrôle initialisé par le client, vers le serveur (port 21 en général), pour transmettre les commandes de fichiers (transférer, supprimer, renommer, lister des fichiers…).
- Un canal de données initialisé par le client ou le serveur (port 20 en général) pour transférer les données (fichiers).