# Couche réseau

# 1 Introduction

C'est la couche n° 3 du modèle OSI. Elle fournit :

- Le **choix du meilleur chemin** entre plusieurs solutions. => ROUTAGE ;
- La **remise en ordre** des trames ;
- La possibilité de **segmenter** des trames ;
- le **contrôle d'erreurs.**

Protocoles les plus connus :

- Internet Protocol,
- X25

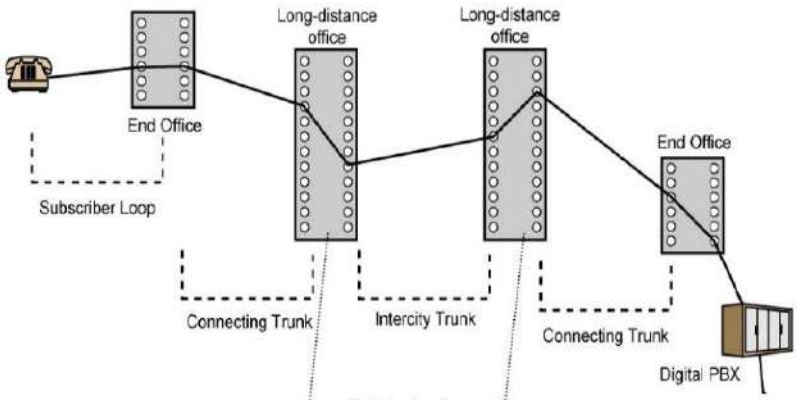
## 1.1 La commutation de circuits

### 1.1.1 Commutation de circuits réels

l'Ancêtre analogique.

Aujourd'hui la technique a évolué pour donner naissance à la commutation de circuits numériques.

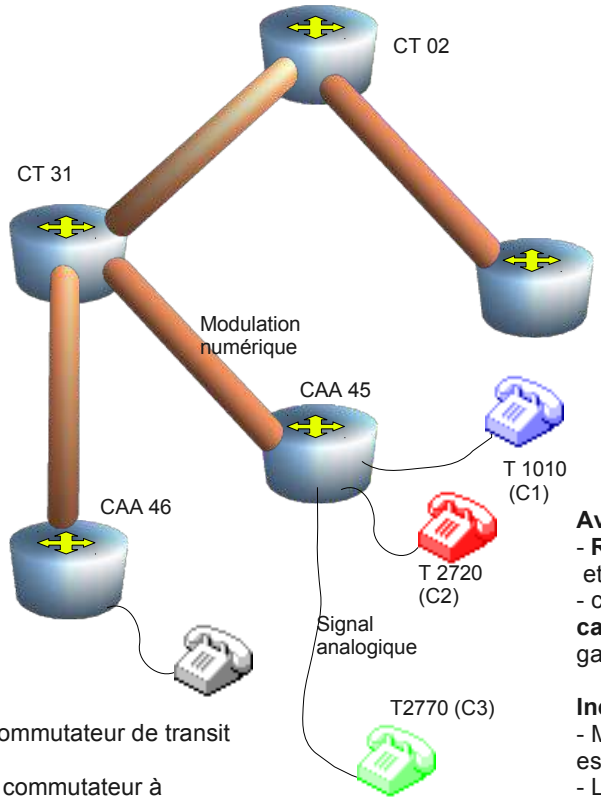
Cette technique a elle aussi évolué et donné naissance à la commutation de circuits virtuels.




### 1.1.2 Commutation de circuits virtuels

Utilisé dans ATM (Asynchronous Transfer Mode), X25 ou Frame Relay.

Par analogie à la commutation de circuits électriques que nous connaissons pour le téléphone, **l'adresse n'est communiquée qu'au début de la connexion**, et le paquet reçoit un identifiant sur chaque tronçon entre 2 commutateurs. L'identifiant est plus court et les commutateurs ont moins de travail à fournir car l'identifiant peut avoir un lien direct avec le numéro de port de sortie du commutateur. Dans le cas de circuits réels, la liaison est réservée de bout en bout et ne peut être utilisée par d'autres. Dans le cas de circuits virtuels, il y a **surréservation** du réseau et le réseau s'adapte en fonction de la demande de chacun. Ainsi, si 2 correspondants ne dialoguent plus ensemble, mais n'ont pas encore rompus la communication, la bande passante qui leur était allouée est redistribuée sur le réseau.



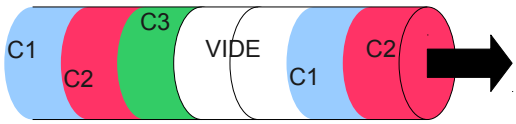

CT : commutateur de transit

CAA : commutateur à autonomie d'acheminement.

**Transmission Synchrone** : multiplexage temporel dans le medium (fibre optique en SDH par exemple) entre un CAA et un CT.

Le signal émis par le téléphone ou le modem est échantilloné (8bits/8kHz) par le CAA puis transmis à un débit supérieur vers le CT.

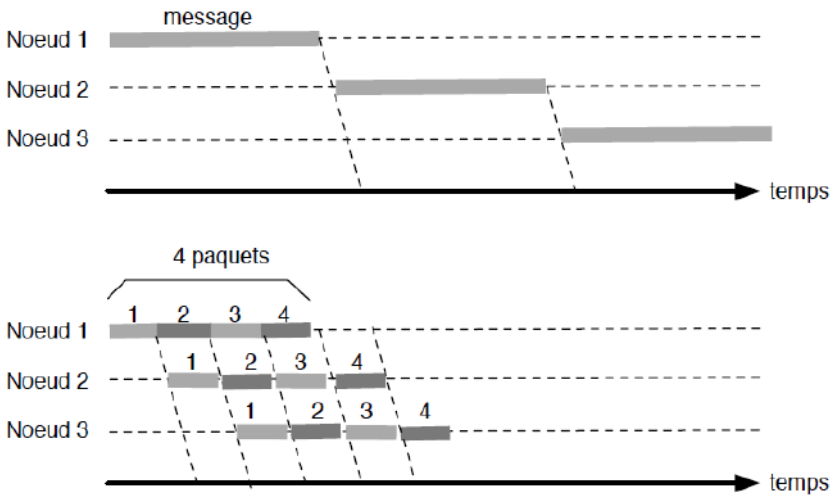
Pour l'ADSL, une bande de fréquence au dessus des 20 kHz est utilisée par le modem afin de ne pas perturber la voix. Dans ce cas le CAA est épaulé par un DSLAM.




**Avantages :**
- **Routage simplifié.** Le CAA ne connaît que ses abonnés et son CT.
- chaque téléphone, une fois la connexion établie, a son **canal réservé** : délai d'acheminement et débit constants garantis.

**Inconvénients :**
- Même s'il n'y a pas d'information échangée, le canal est réservé : **pas de surbooking** faisable.
- Le chemin entre 2 abonnés est toujours le même : pas de délestage possible.
- Connexion obligatoire pour échanger de l'information : perte de temps.

## Avantage des petits messages

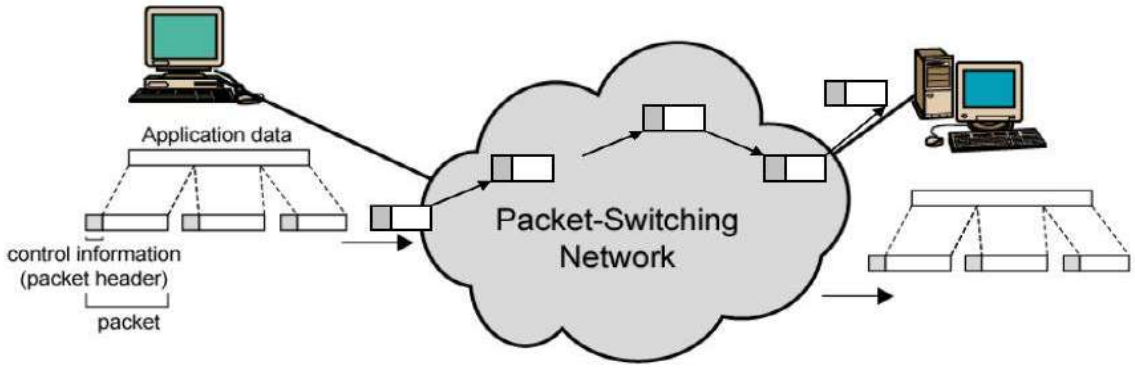



*Les petits paquets arrivent avant les gros paquets.*

Mais dans ce cas, il faut que l'overhead soit peu important. Dans ATM, une cellule fait 53 octets dont 5 octets pour l'entête et 48 pour les données. Chaque cellule ATM ne transporte que les informations liées à l'identification du circuit.

## 1.2 La commutation de paquets

Pour acheminer un paquet à travers un réseau de routeurs (ou commutateurs) un autre solution a vu le jour : L'**acheminement par adresse** de datagrammes comme dans IP.

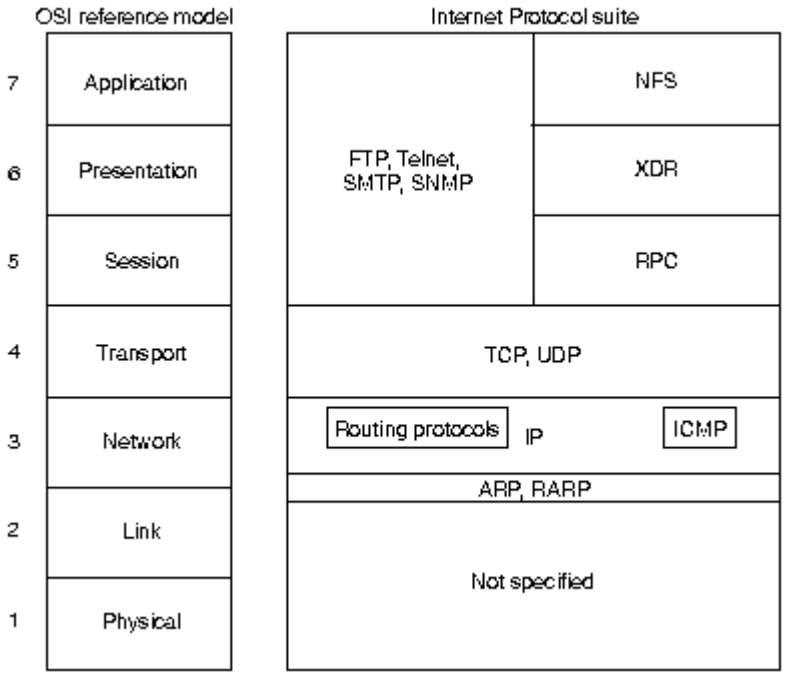



Le paquet est trié dans chaque routeur en fonction de son adresse de destination ; cela demande plus de puissance de calcul que dans le cas de la commutation de circuit, mais c'est simple à mettre en oeuvre.

Les différentes trames :

- les trames de données ;
- les trames de contrôle des données (contrôle de flux, acquittement, reprise en cas d'erreurs...) ;
- les trames de supervision du réseau (gestion du routage, maintenance afin de prévenir la congestion...).

# 2 - Le protocole Internet




Dans le milieu des années 70, le **DARPA** (Defence Advanced Research Projects Agency) fut intéressé par la réalisation d'un réseau à commutation de paquets pour fournir un moyen de communication aux centres de recherches américains. Ce fut le début de l'engouement pour la commutation de paquets, mais aussi le début des problèmes que tout le monde connaît actuellement concernant l'interconnexion de systèmes hétérogènes.

Le résultat du DARPA et de l'université de Stanford fut le développement de la suite de protocoles Internet, les plus connus étant TCP (Transmission Control Protocol) et IP (Internet Protocol).

Les protocoles Internet peuvent être utilisés pour communiquer sur un ensemble de réseaux interconnectés. Ils sont conçus « aussi bien » pour les LAN que les WAN. Ils offrent en outre non seulement des moyens de contrôle

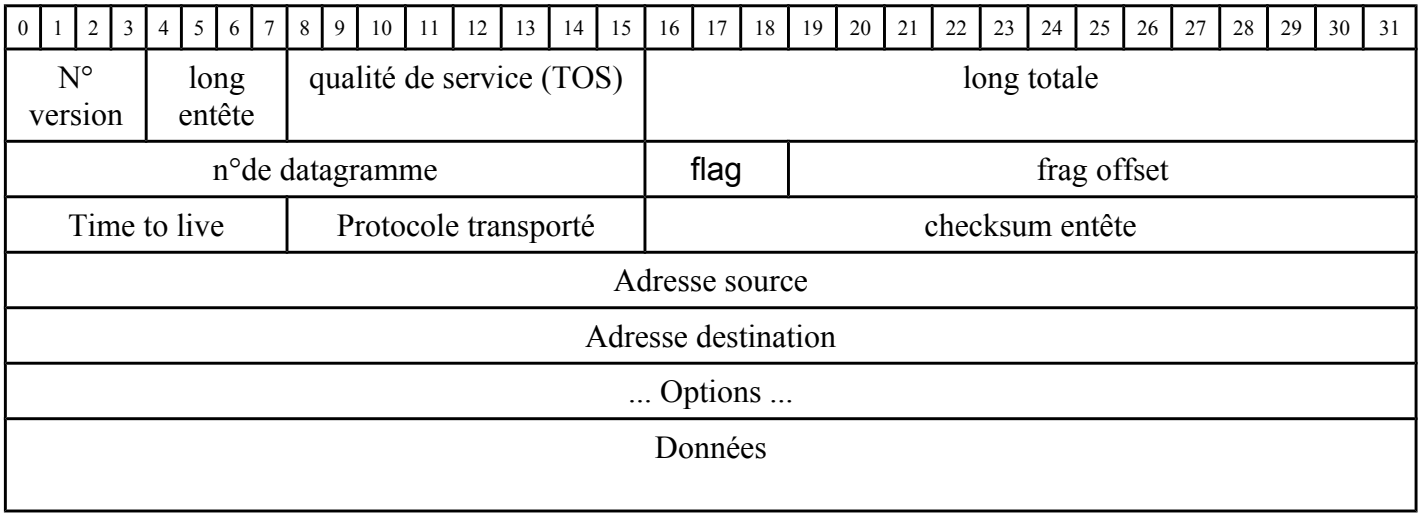
de la transmission des paquets, mais aussi des applications tels que le courrier électronique, l'émulation de terminal à distance, et le transfert de fichiers. La figure ci-dessous donne quelques exemples et la relation avec le modèle OSI.

Les spécifications du protocole Internet se font par des RFC (Request for Comments). Ces RFC sont publiés, puis revues par la communauté Internet, et republiées. Ces travaux sont toujours en cours afin d'améliorer la suite de protocoles IP. Par exemple la RFC 1340 référence tous les codes hexadécimaux utilisés par IP, mais aussi Ethernet.

# 2.1 - La trame de données

## 2.1.1 - Description de la trame

IP est la couche 3 des protocoles Internet. En plus de l'adressage et du routage, IP fournit les services de fragmentation et réassemblage des datagrammes et la notification d'erreurs sur l'entête seulement. IP et TCP sont les fondations de cette suite de protocoles. Le format des paquets est représenté ci-dessous. C'est la norme Big Endian qui est utilisée pour le format des nombres.




- **N° version** : Numéro de version d' IP utilisé (actuellement 4).
- **long entête** : longueur de l'entête en mots de 32 bits.
- **qualité de service** : niveau d'importance du datagramme transporté (souvent non utilisé)
- **long totale** : longueur en octet du paquet.
- **n°de datagramme** : Identifiant de datagramme. Un ensemble de paquets fragmentés issus du même datagramme ont le même identifiant.
- **flag** : 3 bits dont le MSB signifie que le paquet ne doit pas être fragmenté et le LSB qu'il reste des paquets fragmentés à venir.
- **frag offset** : position du paquet dans le paquet fragmenté, en multiple de 8 octets. Un paquet est fragmenté quand il passe sur un réseau de MTU (Maximum Transmit Unit) plus petit.
- **Time to live :** décrémenté à chaque passage par un routeur afin d'éviter un bouclage sans fin des paquets perdus. Décrémenté de 1 ou du temps en seconde de traversé d'un routeur.
- **Protocole transporté** : par exemple TCP = 6 et UDP =17
- **checksum entête** : Pour le contrôle d'erreur. Somme sur 16 bits en complément à 1.
- **Adresse source** : adresse sur 4 octets de l'émetteur de départ.
- **Adresse destination** : adresse sur 4 octets du récepteur final.

## 2.1.2 Adressage

Une adresse IP est représentée sur 32 bits. Elle est divisée en 2 parties de longueur variable : l'adresse réseau et l'adresse de la machine. Il appartient à l'administrateur du réseau de diviser ou non son réseau en sous réseau.
Les adresses réseaux sont réparties en 5 catégories, suivant l'étendue du codage de la partie réseau de l'adresse :
**Classe A** : de 0.0.0.0 à 126.255.255.255, soit 7 bits utiles pour l'adresse réseau (8 bits mais premier bit = 0), pour les grands réseau.
**Classe B** : de 128.0.0.0 à 191.255.255.255, soit 14 bits utiles pour l'adresse réseau (16 bits mais les deux premiers bits = 10) ; pratiquement plus d'adresses de ce type disponibles.
**Classe C** : de 192.0.0.0 à 223.255.255.255, soit 21 bits utiles pour l'adresse réseau (24 bits mais les 3 premiers bits = 110)
**Classe D :** de 224.0.0.0 à 239.255.255.255. réservée pour le multi-adressage, afin qu'une trame devant aller sur plusieurs machines ne soit dupliquée que le plus tard possible. Ceci nécessite un protocole particulier où les machines s'abonnent à un groupe de diffusion.
**Classe E** : usage future.

Exemple d'adresse de classe C en notation décimale pointée : 194.199.103.7

Les classes d'adresses sont distribuées par l' IANA (Internet Assigned Numbers Authority) qui délègue à chaque pays cette distribution.

Afin de faire face à la pénurie d'adresse de classe B, on met en oeuvre une technique d'agrégation des adresses de classe C, appelée CIDR (classless interdomain routing). Les adresses de classes C doivent être contiguës et doivent pouvoir être masquables.

La technique du masque permet de reconnaître la partie réseau de la partie machine. Le masque ou « netmask » est un nombre de 32 bits dont les bits correspondant à la partie adresse machine sont à 0 et les autres à 1. Lorsque l'on fait le ET logique netmask & adresse IP, on doit trouver l'adresse réseau, c'est à dire la première adresse disponible sur un réseau.

exemple :

Soit le réseau 194.64.3.0 contenant 4 sous-réseaux.

en binaire cette adresse s'écrit :

```
11000010 01000000 00000011 00000000
```

Si l'on considère les 4 sous-réseaux ensembles, le masque de ce réseau global est celui des réseaux de classe C :

```
11111111 11111111 11111111 00000000 soit 255.255.255.0
```

Par contre pour distinguer les 4 sous réseaux, on a besoin du masque suivant :

```
11111111 11111111 11111111 11000000 soit 255.255.255.192
```

Remarque : lorsque l'on attribue des adresses à des machines la première adresse et la dernière d'un sous-réseau sont réservées à la diffusion sur tout le réseau. On n'attribuera donc pas ces adresses.

Dans l'exemple précédent, pour le premier sous-réseau, les adresses 194.64.3.0 et 194.64.3.63 sont réservées. Un manière plus concise de donner le netmask associé à l'adresse réseau et de coller à la suite de l'adresse réseau le nombre de 1 du netmask. Exemple 194.64.3.0/26.

Enfin, il existe un certain nombre d'adresses réservées :

- Toute machine se voit localement comme faisant aussi partie d'un réseau local d'adresse 127.0.0.0 et s'attribue l'adresse 127.0.0.1. Cette adresse permet de simuler ou tester des applications IP localement en faisant communiquer des processus localement.

- Il existe 3 ensembles d'adresses qui ne seront jamais distribuées par l'IANA. Ce sont les adresses 10.0.0.0 à 10.255.255.255 (10.0.0.0/8), 172.16.0.0 à 172.31.255.255 (172.16.0.0/12) et 192.168.0.0 à 192.168.255.255 (192.168.0.0/16 ). Elles peuvent donc servir pour adresser des machines cachées (car invisible de l'extérieur du réseau de l'entreprise) qui n'ont pas besoin d'accéder au reste de la communauté Internet. Ces adresses ne sont pas routées par les routeurs à moins que cela ne soit explicitement marqué dans la table de routage.

## 2.1.3 - Attribution d'adresses à une machine :

Pour un réseau local, ceci se fait soit de manière **statique** (l'adresse IP est entrée à la main à la configuration de la carte réseau) ou par le biais d'un serveur **DHCP** (Dynamic Host Configuration Protocol). La machine envoie une trame de **diffusion** à tout le réseau local pour trouver le serveur DHCP et celui-ci lui renvoie une adresse **IP libre** (parmi une ensemble d'adresses qu'il gère). L'adresse est louée pour une durée au delà de laquelle le serveur DHCP reprend l'adresse en avertissant la station. Pendant la **durée de validité**, le serveur DHCP «ping» la station de temps à autre pour savoir si cette machine est toujours connectée.
En plus de l'adresse IP, le serveur DHCP peut aussi fournir l'adresse de la passerelle et du serveur DNS.

Évidemment le serveur DHCP doit être sur le même réseau local que la station demandeuse.

Pour un accès à distance (RTC, ADSL...) le serveur d'accès est aussi serveur DHCP.

## 2.1.4 - Résolution de Nom : DNS

Lorsqu'une application veut envoyer un message à une autre elle utilise en général le nom de domaine. Par exemple la machine cybele de l'école possède le nom suivant : *cybele.ecole.ensicaen.fr*. Cependant, le protocole IP n'utilise pas les noms de machines pour qu'un paquet arrive à la bonne destination, mais les adresses IP, par exemple 193.49.200.148.
La machine doit donc pouvoir faire la correspondance entre les noms et les adresses IP.
Plusieurs techniques existent :

- La machine possède un fichier ***/etc/hosts*** (Unix) ou *lmhosts* (Windows) dans lequel on a une correspondance entre des noms et des adresses IP ;
- La machine est connecté à un serveur **NIS** (Network Information Service de Unix) ou WINS (de Windows) et ce serveur met à disposition son propre fichier */etc/hosts ;*
- Netbios Name Server, un service propore à Windows ;
- Domain Name serveur ;
- Multicast DNS, un service de la série Zeroconf.

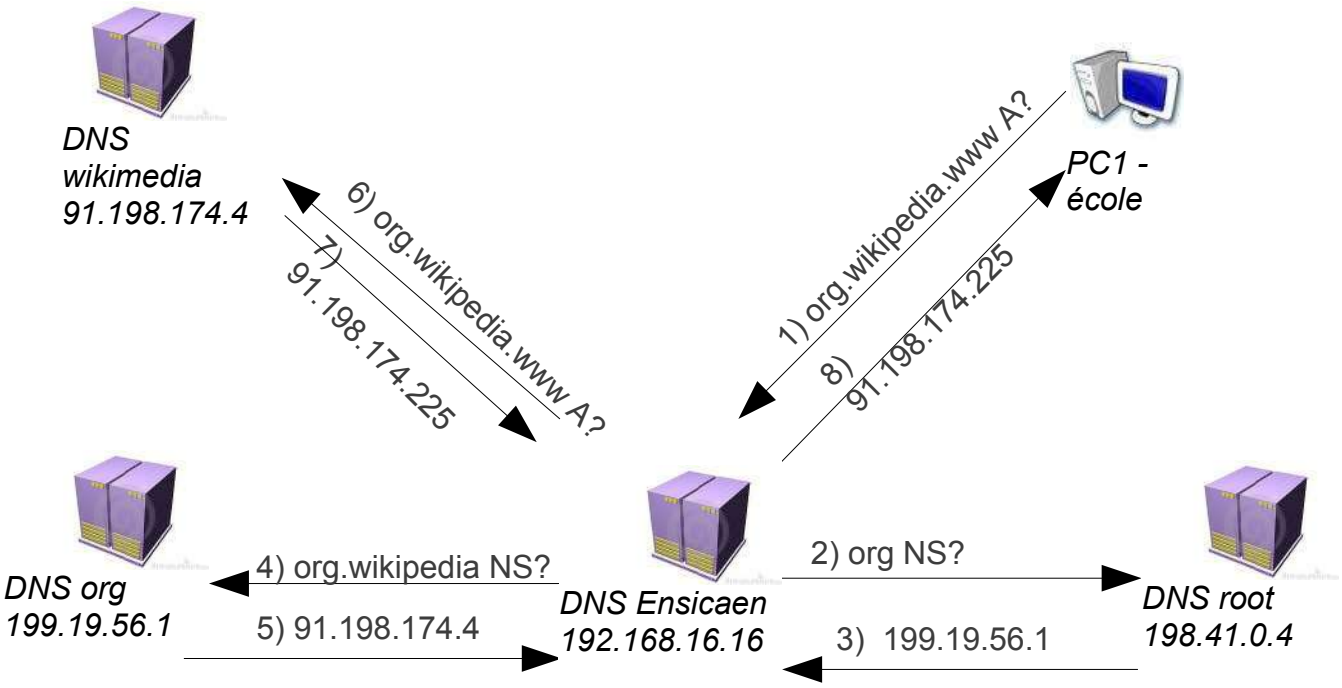
### interrogation d'un DNS (Domain name System)

Un serveur DNS est une machine qui répond à des requêtes de noms de domaines. Ainsi chaque domaine ou sous-domaine est géré par une machine qui fait autorité. C'est elle qui connaît toutes les adresses IP de toutes les machines du domaine. Cette machine est capable aussi d'aller interroger le bon serveur DNS quand elle ne connaît pas la réponse. Souvent, elle stocke en cache la réponse et devient capable de donner la réponse la prochaine fois. Cette réponse sera dite « non authoritative ».
Les serveurs DNS permettent aussi de fournir l'adresse IP du serveur de mail (serveur SMTP ) d'un domaine.

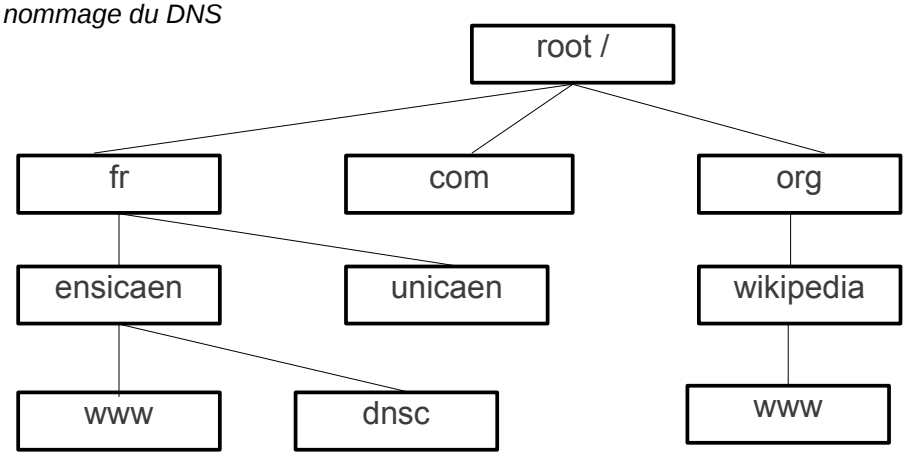
Les requêtes DNS sont de plusieurs types parmi les plus fréquentes :

- A : adresse Ipv4
- AAAA : adresse Ipv6
- NS : nom du DNS
- MX : nom du mail exchanger (serveur SMTP)

*Exemple d'interrogation DNS : Un serveur de l'école demande l'adresse IP de www.wikipedia.org.*




*Arbre de nommage du DNS*




## 2.1.5 - ARP (Address Resolution Protocol)

Les adresses de couches 2 sont celles qui permettent la liaison point à point entre 2 machines. Il faut donc faire la correspondance entre les adresses physiques de couche 2 et les adresses réseaux de couche 3. Cette correspondance peut se faire de manière dynamique. C'est le cas des réseaux IEEE 802 tel qu' Ethernet. Le protocole ARP permet de faire cette correspondance. L'émetteur connaît l'adresse IP du destinataire, mais pas son adresse MAC (Media Access Control). Il envoie donc sur touT le réseau local une trame de diffusion de manière à demander au destinataire de communiquer son adresse physique (MAC).
RARP (Reverse Address Resolution Protocol) permet l'opération inverse. Cela peut être utile pour des stations sans disque demandant alors leur adresse IP à un serveur lors de leur démarrage.

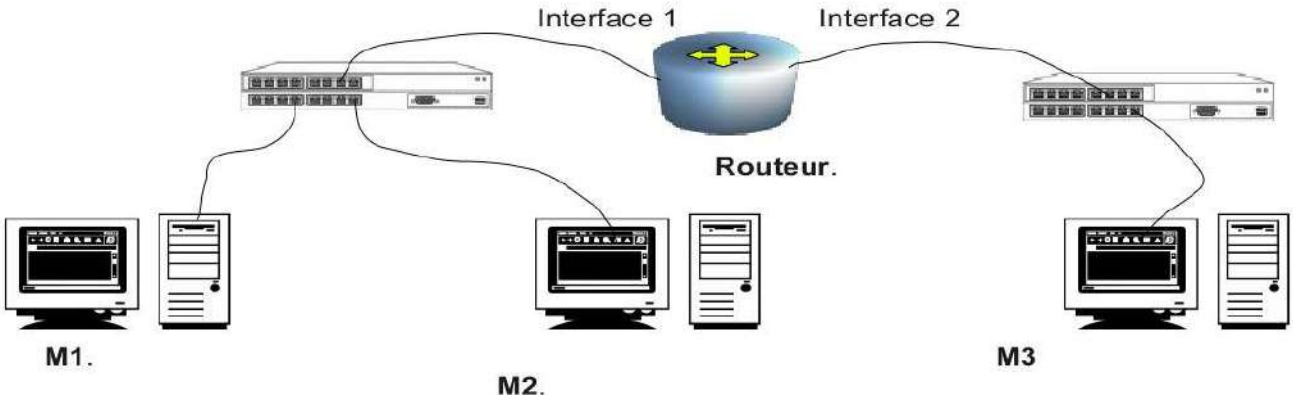
### 2.1.6 – Zeroconf

- Série de protocoles utilisés pour auto-configurer une machine
- Ipv4 Local Link : permet à une machine de s'attribuer automatiquement une adresse sur le réseau 169.254.0.0/16
- Multicast DNS : permet à toute machine de répondre à une requête DNS. Les requêtes MDNS utilisent l'adresse multicast 224.0.0.251
- Pas de route par défaut fournie.

## 2.2 - Routage IP

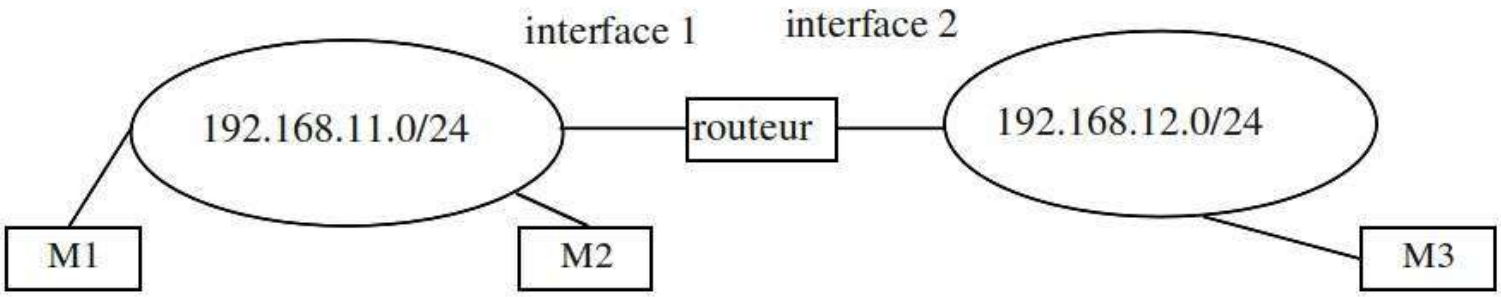
### 2.2.1 - Principe

- Il existe 2 types de routage : le routage **direct**, de machine à machine et le routage **indirect**, en passant par un ou plusieurs routeurs. A chaque fois qu'un routeur reçoit une trame (l'adresse de couche 2 de la trame est celle du routeur), il doit renvoyer la trame sur un de ses ports, **sans changer le contenu du datagramme ni l'adresse de destination.** Il change alors l'adresse physique de destination afin de redistribuer la trame soit à un autre routeur, soit à la machine finale.
- L'exemple suivant montre le **schéma physique** d'un réseau de 3 machines reliées par 2 switches, eux-mêmes interconnectés par un routeur.

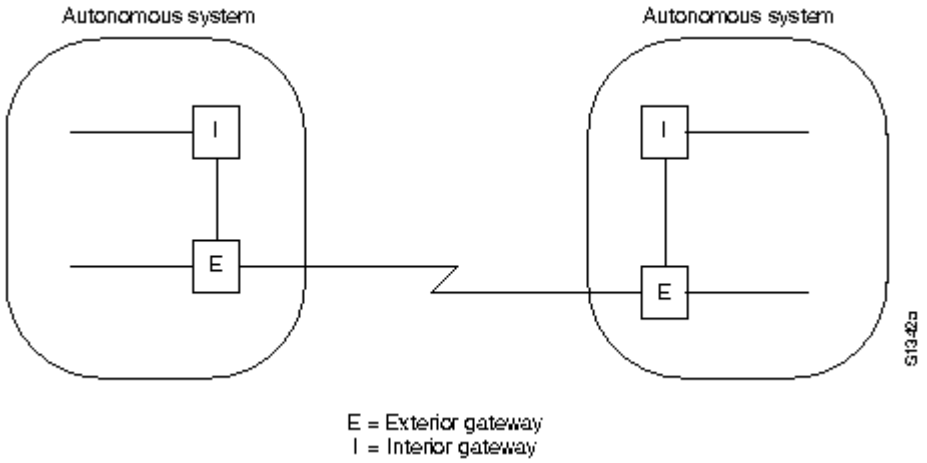



**M1 vers M2 :** La trame qui part de M1 contient les adresses MAC de M1 et M2 pour la partie ethernet et les adresses IP de M1 et M2 pour la partie IP.
**M1 vers M3 :** La trame qui part de M1 contient les adresses MAC de M1 et de l'interface 1 du routeur pour la partie ethernet et les adresses IP de M1 et M2 pour la partie IP.
Une fois arrivée au routeur, la trame en repart avec les adresses MAC de l'interface 2 du routeur et de M2 pour la partie ethernet et les adresses IP de M1 et M2 pour la partie IP.

- D'un point de vue IP, on préfère substituer au schéma physique, le **schéma logique**, uniquement constitué des équipements de couche 3 : routeurs et PC. Ce schéma permet de bien visualiser les différents réseaux IP.




- Le routage est fait selon l'algorithme de la **patate chaude** ou encore de manière "best-effort". Une trame est analysée en fonction de l'adresse de destination. Si le routeur ne sait pas il redistribue la trame à un routeur par défaut.
- Lorsqu'une machine désire atteindre une machine autre que celles qui sont sur son réseau local (même adresse réseau) la machine envoie la trame au routeur qui se trouve sur son réseau local.
- Afin de limiter la taille des tables de routage, **seules les adresses de réseaux sont stockées**. Si l'adresse de destination correspond à un couple adresse réseau / netmask connu, alors elle est soit envoyée soit sur un routeur soit directement à la machine destination si cette dernière est connectée à un réseau local auquel le routeur est lui-même connecté. Dans le cas contraire, la trame est envoyée à un autre routeur par défaut qui sait peut-être la router.
- Afin d'éviter des bouclages infinis, **le champ TTL est décrémenté** à chaque passage dans un routeur et la trame est détruite lorsque celui-ci atteint la valeur 0. Le routeur qui détruit la trame envoie un message ICMP à l'émetteur pour le prévenir que la trame a été détruite.




- Les passerelles (gateways) ou routeurs désignent les machines capable d'effectuer les opérations de routage. Ils sont souvent organisés de manière hiérarchique. Certains sont utilisés pour transmettre l'information à un site particulier de sous-réseaux sous la même autorité administrative. Les routeurs utilisés pour l'échange d'information à l'intérieur de ce site utiliseront des protocoles IGP (interior gateway protocols) pour accomplir leur travail. Les routeurs de bordure permettant la communication entre sites autonomes utiliseront quant à eux des protocoles EGP (exterior gateway protocol) comme BGP.

- Le routage IP est dynamique. Les routes les plus courtes ( coût le plus faible en terme de noeuds ou de temps) sont calculées à intervalles réguliers. Mais l'administrateur d'un site peut imposer un routage statique.

- Une table d'adressage IP contenue dans un routeur est une correspondance entre l'adresse réseau à atteindre, l'adresse du prochain routeur par lequel il faut passer pour l'atteindre, et parfois le coût pour l'atteindre.
- A chaque fois qu'un datagramme IP arrive à un routeur, ce dernier le réexpédie vers une autre branche du réseau en fonction de l'adresse de destination contenue dans ce datagramme. La route totale suivie par le paquet n'est connue ni de l'émetteur, ni du récepteur. S'il arrive un problème de routage, le routeur le remarquant « peut » notifier l'erreur par le biais du protocole ICMP.

### 2.2.2 - un exemple : RIP

Un des premiers protocoles de routage interne (IGP) à être apparu en masse dans les réseaux IP est **RIP** (Routing Information Protocol), et sa version améliorée RIPv2. RIP fait partie de la catégorie des protocoles de routages dits **"distance-vector",** à opposer à la catégorie plus moderne des **"link-state"** dont fait partie notamment **OSPF**. Ces derniers sont plus modernes et plus efficaces puisqu'ils tiennent comptent non pas du nombre de routeurs traversés, mais de l'état des liens empruntés (débit, congestion, temps de latence...).
Avec RIP, chaque routeur **annonce les préfixes IP qu'il est capable de joindre**, avec une métrique égale au nombre de routeurs traversés. Si un message est reçu d'un routeur annonçant une route plus courte pour joindre un préfixe dont on a déjà trace dans la table de routage, on remplace cette entrée par la nouvelle, symbolisant le nouveau plus court chemin pour joindre la destination.
A intervalles réguliers, chaque routeur envoie à ses voisins sa table de routage complète afin de s'assurer que les informations sont cohérentes dans le réseau. Pour ne pas perturber inutilement le réseau avec des messages de broadcast, les échanges sont faits par l'adresse multicast, 224.0.0.9. Seules les machines intéressées par ces paquets les décoderont.
Lorsqu'un évènement particulier dans le réseau provoque un c**hangement dans la table de routage d'un routeur**, celui-ci envoie un **message de mise à jour à ses voisins** concernant le préfixe impacté (injoignable, changement de métrique, etc.). Si le comportement de routage de ceux-ci est également impacté, **ils devront eux-même propager l'information**. Afin d'éviter de trop fréquentes mises à jour, une temporisation est nécessaire afin de stabiliser le réseau.
Cisco IOS software envoie des informations de mise à jour toutes les 30 secondes. Si un routeur ne reçoit pas de publicité concernant une route pendant 180 s il marque la route inutilisable. Si au bout de 240s il n'a toujours pas d'information, il l'enlève de la table de routage.

Le "Split-Horizon" consiste à **ne pas ré-annoncer à un routeur une route que l'on a apprise par lui**. Car même si la route ainsi renvoyée ne peut être immédiatement celle de plus court chemin (+2 à la métrique à cause de l'aller-retour sur le lien), il se peut qu'en cas de rupture d'un lien amont sur le routeur d'origine, celui-ci puisse croire que la route optimale passe par le routeur qui lui a ré-annoncé sa propre route.

## 2.3 - Le plan de contrôle : ICMP (Internet Control Message Protocol)

Ce protocole permet de contrôler les paquets IP. Il permet notamment l'envoie de messages tels que :

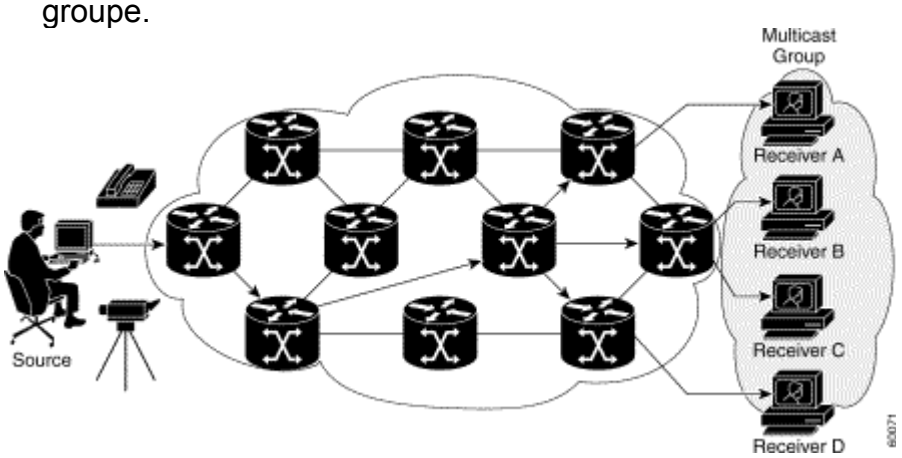
- Echo/Reply pour tester si un hôte est atteignable ou non (utilisé par la commande *ping*) ;
- Informer d'un dépassement de durée de vie (TTL) d'un datagramme perdu sur le réseau, et donc détruit ;
- « Router advertisement » et « router solicitation » pour découvrir l'adresse d'un routeur interne ;
- Un routeur peut avertir une machine que son paquet n'a pas pu être délivré.
- ICMP redirect : Un routeur peut avertir une machine qu'il existe un chemin direct en passant par un autre routeur. Il fournit alors l'adresse IP de cet autre routeur.

| 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 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| type | | | | | | | | Code | | | | | | | | checksum | | | | | | | | | | | | | | | |
| Données complémentaires dépendant de Type | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Entête Internet et les 64 premiers octets du datagramme ayant provoqué l'émission du paquet ICMP | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

Quelques types et codes :
type = 3 : Le message ne peut atteindre sa destination
code = 0 : réseau ne peut être atteint
1 : station ne peut être atteint
type = 8 : demande d'écho
type = 0 : réponse à l'écho
type = 11 : durée de vie expirée.

## 2.4 - IGMP ( Internet Group Management Protocol)

- Protocole permettant d'envoyer des messages d'une machine vers plusieurs en **évitant le plus possible la réplication** des messages.
- Transporté par IP. IP s'occupe de l'adressage multicast avec la classe D : **une seule @IP de classe D est donnée à un groupe de machines**.
- IGMP s'occupe de la gestion des utilisateurs qui désirent **s'abonner ou se désabonner** au groupe.




- La machine s'abonne au groupe en dialoguant avec le routeur multicast le plus proche.
- Une fois abonnée, le routeur multicast demande de façon aléatoire des **comptes rendus pour savoir si la machine désire toujours faire partie du groupe**. Si elle ne répond pas, elle est désabonnée.
- D'autres protocoles sont utilisés afin de résoudre les nombreux problèmes que pose la multi-diffusion :
  - Trouver le **meilleur arbre pour joindre tous les membres** ?
  - Comment gérer la QOS : 2 utilisateurs d'un même groupe demandent des **QOS différentes, laquelle choisir** ?
  - **Peut on dégrader le message** (ex : video) pour un membre du groupe si celui-ci n'a pas les ressources physiques pour accepter cette QOS ?
  - comment organiser des tunnels entre routeurs multicast, séparés par des routeurs non multicast ?...

# Couche transport

Les fonctionnalités de couche 4, à savoir le **transport fiable d'informations de bout en bout**, sont assurées dans le monde Internet par 2 protocoles, TCP ou UDP. TCP offre un transport à la fois **connecté et plus fiable** que UDP. Ce dernier étant plus **efficace** dans un réseau générant très peu d'erreurs.
Ces 2 protocoles dialoguent avec les couches supérieures par l'intermédiaire du **numéro de port**. Le port destination est en général connu, c'est celui qui **référence l'application** sur le serveur (21 ftp, 22 ssh, 23 telnet, 25 smtp, 53 dns, 80 http,, 110 pop3, 161 snmp...). Quant au port source du client, il est attribué par le système, permettant ainsi à **plusieurs tâches** de se connecter. Ceci crée un circuit virtuel entre le client et le serveur. Il est virtuel, puisque la couche du dessous (IP) peut à tout moment, changer le chemin physique pris par la trame. On dit aussi que TCP(UDP)/IP est un protocole à **commutation de paquets**, car tous les paquets sont indépendants et peuvent suivre des **chemins différents**. Ceci est à différencier de la commutation de circuits réels telle qu'on la connaît pour le téléphone, où le circuit est réservé et fixé de bout en bout, a tel point d'ailleurs que même si personne ne parle, personne d'autre ne peut néanmoins utiliser votre ligne et évidemment cela vous est facturé.

## 1 TCP (Transmission Control Protocol)

### 1.1 Caractéristiques

- TCP fournit une communication **full-duplex**, avec accusé de réception et contrôle de flux.

- **Les paquets TCP sont numérotés**, et TCP peut supporter plusieurs connexions simultanées. Il offre une connexion sur un circuit virtuel identifié de manière unique par les adresses et les numéros de port de l'émetteur et du récepteur.

- **TCP est bufferisé** : l'application utilisant TCP rempli des tampons et TCP les envoie lorsqu'ils sont remplis suffisamment de manière à assurer une transmission la plus efficace possible. L'utilisateur peut néanmoins commander le vidage des tampons.

- TCP conserve en mémoire tout paquet envoyé non acquitté pour pouvoir le retransmettre. L'accusé de réception se fait en transmettant le numéro du prochain octet attendu, un peu à la manière de HDLC. Si l'horloge arrive à expiration, le paquet non acquitté est réémis. Une trame d'acquittement peut acquitter plusieurs paquets. Mais il est recommandé que les systèmes acquittent un paquet au plus tard 0,5s après sa réception.

- Un mécanisme de fenêtre glissante est utilisé de manière à autoriser l'émetteur à envoyer plusieurs trames avant d'attendre l'acquittement de la première. La taille de la fenêtre est dynamique. Le récepteur indique le numéro de l'octet maximum qu'il peut recevoir, informant ainsi de la capacité de ses tampons de réception. On oppose ce mécanisme au vieil acquittement positif qui consiste à attendre tant que l'on n'a pas reçu l'acquittement. Au démarrage de la connexion (slow start), la taille de la fenêtre est fixée à la taille maximum d'un paquet. Puis cette taille augmente de 1 (ou est doublée selon les implémentations) à chaque fois qu'un volume de paquet correspondant à la taille de la fenêtre a été acquitté. Dès qu'une congestion est détectée, la taille de la fenêtre est divisée par 2.

- Lorsqu'un paquet se perd, il empêche tous les paquets suivants d'être acquittés : c'est un problème car il risque d'y avoir duplication.

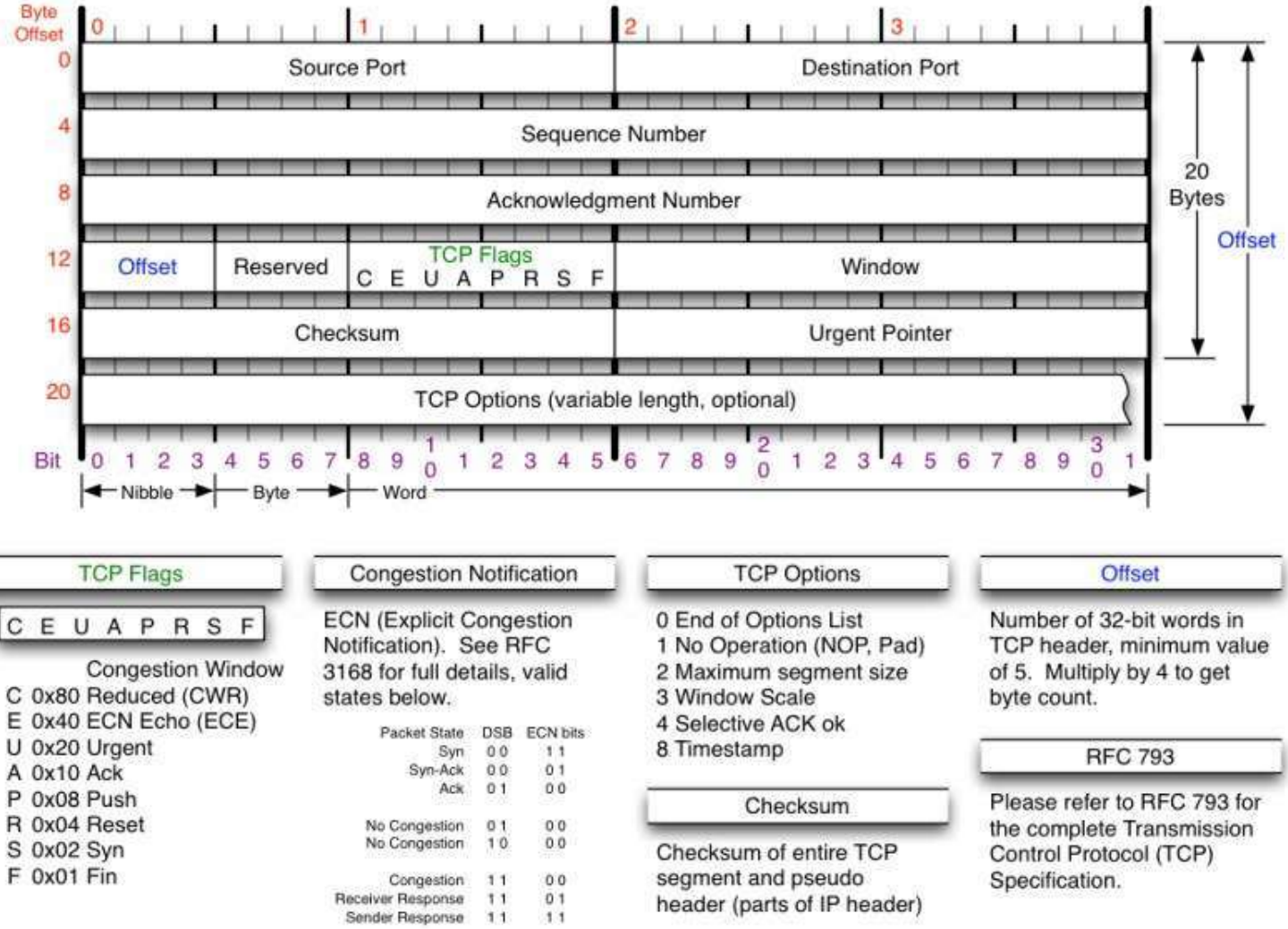
- Le réglage du temps de retransmission (RTT) pose aussi un problème. Trop court, il y a risque de retransmission inutile, trop long ce n'est pas efficace. En fait, le temps de boucle, temps moyen entre un paquet-aller et un acquittement-retour est mesuré. Un algorithme adaptatif sert à estimer RTT par rapport au temps de boucle. Mais attention si un paquet est supposé perdu, comment savoir si l'acquittement vient du premier paquet ou du paquet retransmis ? Si on suppose qu'il vient du premier, le RTT risque d'augmenter sans cesse si il y a beaucoup de paquets perdus. Si on suppose qu'il vient du second alors qu'en fait le réseau est simplement lent, RTT va tendre vers 0 ! Des algorithmes sophistiqués tentent de régler ce problème. La norme TCP ne précise rien à ce sujet.

- RTT est initialisé au temps d'aller retour de la demande de connexion.

- Lorsqu'un paquet traverse différents réseaux, les MTU de ces réseaux peuvent être plus petits que la taille du paquet initial. Dans ce cas le paquet est fragmenté ce qui ralenti la communication puisque cela nécessite plus de traitements. Pour éviter cela la taille maximum d'un paquet TCP (Maximum Segment Size) est négociée au départ de la connexion. Le MSS est initialisé au MTU du réseau local et peut donc être différent pour les 2 sens de la communication.

Format de trame :

*source nmap.org*




– **Source port et destination port :** Spécifie avec quel port (telnet, FTP...) la communication se fait

avec les couches supérieurs. C'est le Service Access Point de TCP.
**– Sequence number**: désigne normalement un numéro assigné au premier octet du message transporté.
**– Acknowledgment number :** Contient le « Sequence number » du prochaine octet que l'émetteur s'attend à recevoir. (ce mécanisme sert d'acquittement).
**– Data offset :** nombre de mot de 32 bits de l'entête TCP.
**– Reserved** : pour usage futur.
**– Flags :** SYN (demande de connexion), ACK (acquittement), FIN (demande de libération de connexion), PSH (le système qui reçoit ce paquet doit le transmettre aussitôt à l'application. Le comportement par défaut est de bufferiser et de ne transmettre à l'application que des paquets suffisamment volumineux)
**– Window :** Spécifie la taille maximale des données que peut recevoir l'émetteur.
**– Checksum :** somme de contrôle sur l'entête.
**– Urgent pointer :** point sur le premier octet urgent dans le paquet TCP, pour les données "hors bande"
– **Options :** différentes valeurs dont le MSS lors des trames SYN.

Diagramme d'état :

Exemple de connexion :

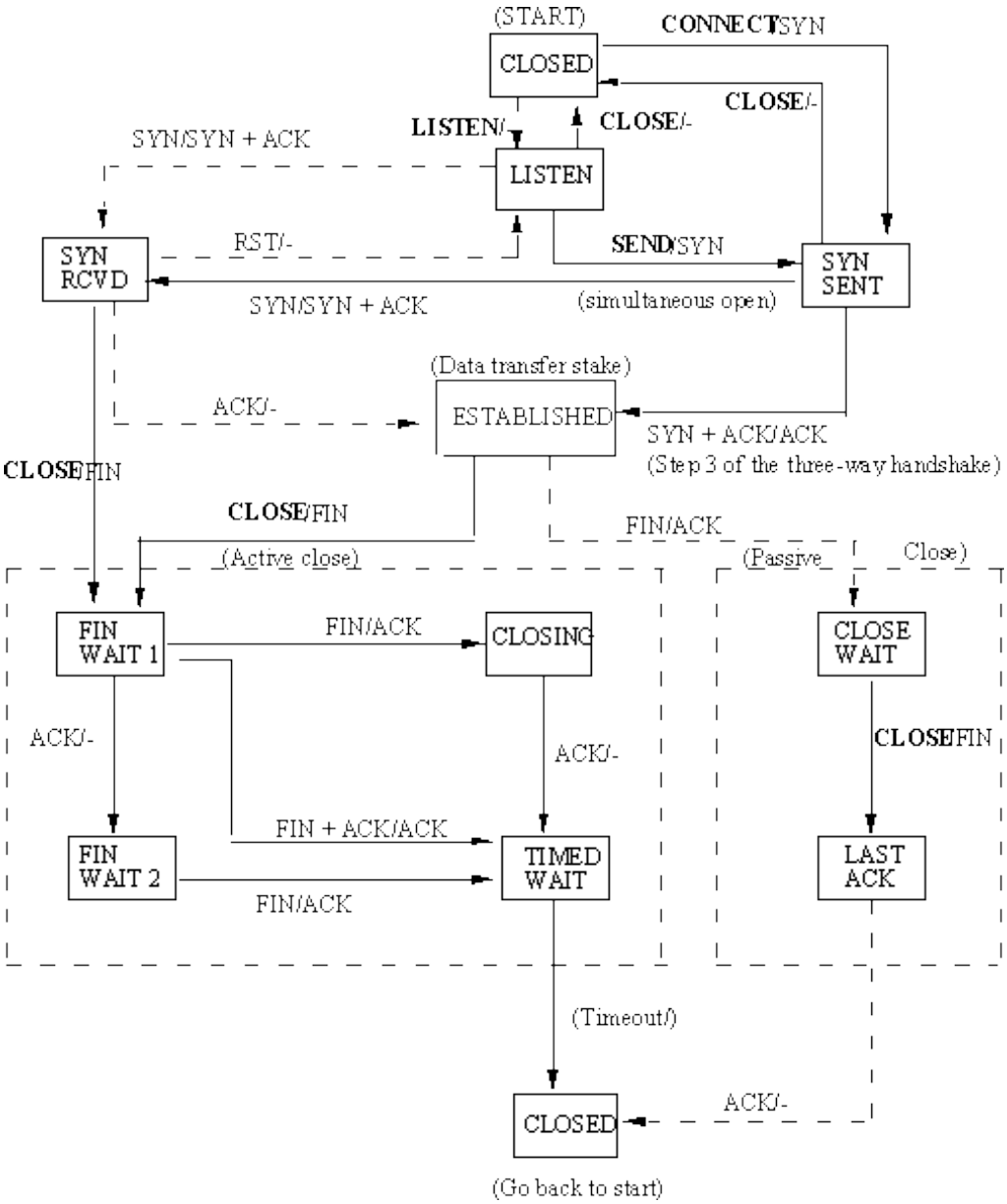
| Partie appelante | Partie appelée |
|---|---|
| SYN -------> | |
| | <---------- ACK + SYN |
| ACK -------> | |

Exemple de déconnexion :

| Partie interrompante | Partie interrompue |
|---|---|
| FIN----------> | |
| | <---------- ACK |
| | <---------- FIN |
| | ou bien <---------- FIN + ACK |
| ACK ----------> | |

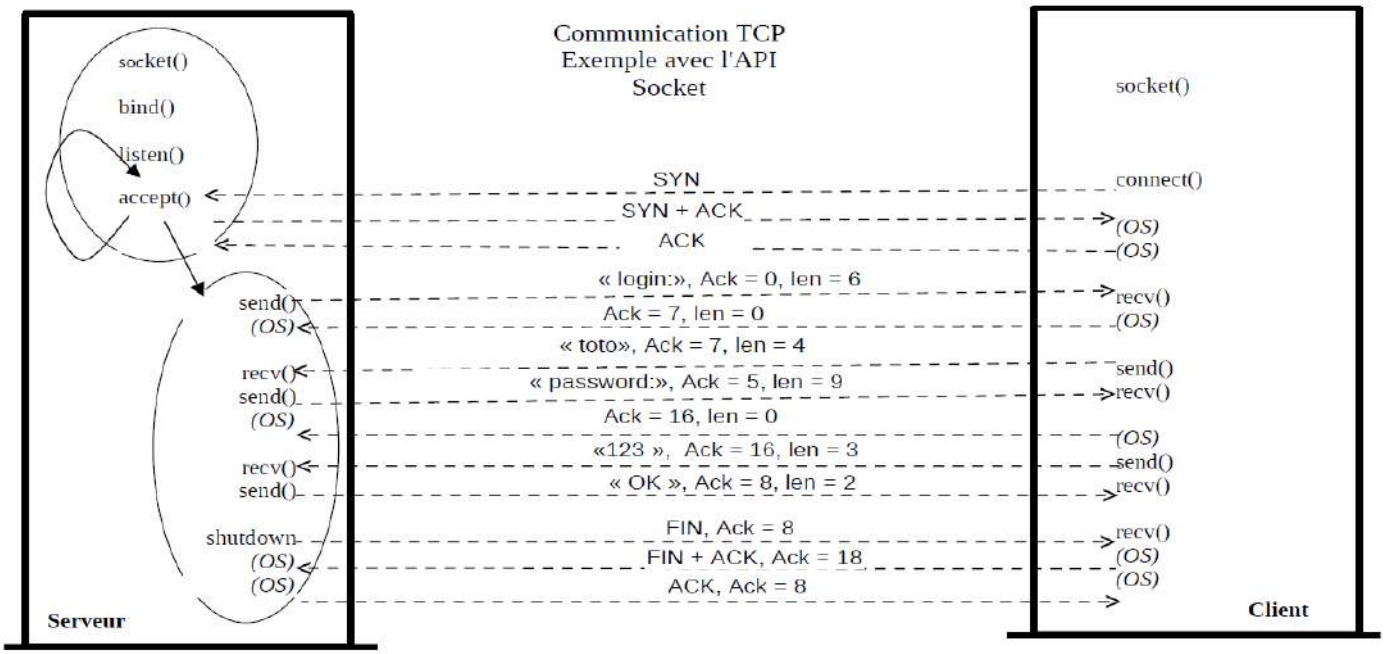
**Automate** :

C'est un diagramme qui décrit les transitions entre les états de TCP. Il se lit avec les conventions suivantes :




- en gras, action utilisateur vers TCP ;
- en non gras, paquet, émis ou reçu ;
- barre oblique pour indiquer : réception / émission.

# Diagramme de séquence d'une communication TCP.




*(OS) : quand la trame n'est pas explicitement reçue par l'application mais seulement gérée par le système d'exploitation.*

## 3.2 - UDP (User Datagram Protocol)

Puisque UDP offre moins de services que TCP, son entête est plus courte, ce qui en fait un protocole plus efficace.

L'entête possède 4 champs :

- port source (2 octets),
- port destination (2 octets),
- longueur totale (2 octets)
- checksum sur l'entête (2 octets).

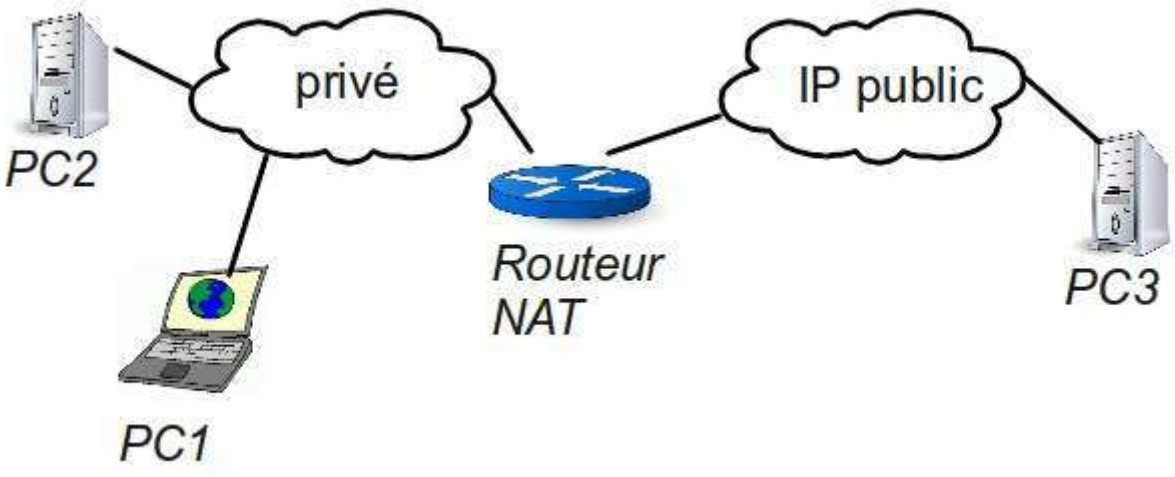
UDP ne peut pas transporter des trames plus grandes que le MTU de IP. Ce qui fait que sur un réseau ethernet/IP, la taille maximale d'un paquet est de 1500-20- 8 = 1472 octets. Si l'application veut envoyer un paquet plus grand, celui-ci est en général tronqué par la couche UDP.

UDP est utilisé lorsque :

- on ne veut pas surcharger le réseau (requêtes DNS) : Normalement le réseau est fiable ; pourquoi alors échanger 8 trames TCP pour une simple requête/réponse qui demande 2 trames en UDP. Autant prendre le risque !
- la reprise sur erreur ne sert à rien : En téléphonie, si le mot manquant d'une discussion arrive après... quelle pagaille !

## 3.3 – Network Address Translation / Port Address Translation

- Pour faire face à la **pénurie** d'adresses IP et partager une connexion internet (c'est ce que fait une box ASDL)
- Mais aussi pour **contrôler** ce qui sort sur l'internet public (Toutes les connexions data sur réseau GSM et dérivés utilisent ce principe).




- Quand PC1 envoie une trame à PC3, le routeur **remplace l'@ IP source de PC1** par son adresse IP (le routeur possède une adresse IP publique puisqu'il est relié à l'internet public).
- Le routeur mémorise le numéro de port source utilisé par PC1.
- Si le routeur (ou PC2) tilise déjà ce port source à destination de PC3 et sur le même port de

destination, le routeur remplace également le port source.
- PC3 répond donc au routeur qui renvoie à PC1 en changeant l'adresse destination et éventuellement le port destination.
- Comment PC3 peut-il ouvrir une connexion sur PC1 ? Il faut que cela soit prévu au niveau du routeur. (par exemple redirection de port d'une box ADSL).

## 3.4 - PPPOE, ou comment surcharger un protocole qui n'en avait pas besoin

- Connexion via un Fournisseur d'accès à internet → facturation → Identification

- TCP ou UDP ne savent pas identifier → utilisation d'un protocole qui sait le faire : Point to Point Protocol.

- PPP inventé pour les liaisons séries et basé sur High-Level Data Link Control. PPP rajoute à HDLC des protocoles d'autentification tels que Microsoft Challenge-Handshake Authentication Protocol.

- HDLC est un protocole de niveau 2 comme ethernet qui assure les services de connexion/déconnexion, contrôle et reprise sur erreur.

- Comme la plus part des box utilisent ethernet, PPP est inséré entre IP et ethernet. On l'appelle dans ce cas PPP over Ethernet. Il occupe 8 octets et diminue le MTU d'autant.

# 4 - Qualité de service dans les réseaux

**Plusieurs techniques** intervenant en couche 2 ou 3 existent :

## 4.1 - ethernet 802.1Q/p

Extension de norme associée aux VLAN : extension de la
trame Ethernet : passe de 1518 octets à 1522 octets
4 octets ajoutés devant le champ type (VLAN tag):
- Tag Protocol (16 b): Identifier (8100 pour Ethernet)
- User Priority ( 3 b)
- CFI (1 b )
- VLAN ID (12 b)

- 8 niveaux de priorité, qui permettent à un commutateur d'écouler un trafic prioritairement à un autre. Mais aucun délai d'acheminement ni de bande passante réellement garantie, contrairement à ATM ou MPLS.
- Peut servir à mapper le champ priorité d'un datagramme IP (Type of Service)
- Gestion de la priorité (comme la gestion des VLAN) est propriétaire et peut varier d'un fournisseur de commutateur à l'autre

## 4.2 - MPLS

Multi- Protocol Label Switching permet de marquer des paquets transitant à travers un même réseau local étendu. Les paquets d'un même flux recevront le même label, seront traités avec la même priorité et emprunteront le même chemin. Ceci simplifie le traitement des paquets d'un même flux car

celui-ci est fait en couche 2 et non en couche 3. MPLS rajoute 4 à 8 octets entre l'entête ethernet et l'entête IP.

## 4.3 - Diffserv :

- se sert du champs TOS d'IPv4 ;
- Le champs Type Of service sert à indiquer la priorité du datagramme et le type de routage nécessaire ;
- Les 3 premiers bits définissent la priorité : 000 : normal, 001 : prioritaire, 010 : immédiat, 011 : urgent, 1XX gestion réseau ;
- Les 4 derniers appelés D,T,R,C indiquent si le datagramme doit être routé au meilleur Délai, débit , fiabilité ou Coût... ou une combinaison des 4 ;
- L'algorithme de routage QosPF (QoS Path First) permet de calculer la meilleur route pour tenir compte des contraintes de QoS des paquets.

## 4.4 - RTP / RTCP (IPv4)

Ces deux protocoles sont transportés par des paquets UDP. Le protocole Real Time Protocole transporte les données multimédia. Il permet l'horodatage des paquets pour reconstiruer une base de temps du flux. Il permet aussi de détecter une rupture de flux.

Le protocole Real Time Control Protocol contrôle des données transportées par RTP. Il fournit périodiquement des rapports à l'émetteur ou aux récepteurs (cas de la multi-difusion) tels que variation de délai, nombres de paquets perdus...

## 4.5 - RSVP (Ipv4)

Resource Reservation Protocol a été conçu à l'origine pour adapter les flux aux différents récepteurs d'une multi-diffusion. En effet les caractéristiques des liaisons peuvent être différentes selon les récepteurs et RSVP est capable de dégrader (pour un flux video par exemple) les transferts vers un récepteur à faible débit.

C'est le récepteur qui fait la demande de réservation de bande passante. Le chemin emprunté sur le réseau est alors unique. Le récepteur est obligé d'envoyer périodiquement des confirmations de réservations aux routeurs pour conserver la QoS demandée.

# 5 - IP v6

## 5.1 - Présentation

Ipv6  est née pour palier à deux problèmes cruciaux d'IP v4 : La **pénurie d'adresses** (75 % des adresses sont réservées par les USA), et le **manque de qualité de service**. La longueur des adresses a été portée a 128 bits et des mécanismes de réservation de bande passante ainsi que de contrôle du délai d'acheminement ont été rajoutés. Néanmoins, ce protocole a du mal a percé car des rustines ont été apportées à Ipv4 : NAT et CIDR pour les adresses et DiffServ, RTP/RTCP/RSVP pour la QoS.

## 5.2 - La trame

<table>
<tr><td>No de version<br>(4 b)</td><td>Champ differentiate service ou priorite datagramme (RFC 1883) (8 b)</td><td colspan="2">Indentificateur de flot<br>(20b)</td></tr>
<tr><td colspan="2">Longueur des données (16 b)</td><td>Entête prochain (8 b)</td><td>nombre de sauts (8b)</td></tr>
<tr><td colspan="4">Adresse émetteur (128 b)</td></tr>
<tr><td colspan="4">Adresse récepteur (128 b)</td></tr>
<tr><td colspan="4">Entête suivant s'il existe</td></tr>
<tr><td colspan="4">Données</td></tr>
</table>

- **L'entête est simplifiée** par rapport à Ipv4 de manière d'alléger le traitement dans le routeur.
- Le type de l'entête suivant est précisé par le champs «entête prochain». Ce champ existe dans les entêtes supplémentaires pour chaîner d'autres entêtes ou vaut le **numéro du protocole transporté pour le dernier entête** (06 TCP). Les entêtes servent à transmettre des informations de routage, QoS...
- **L'identificateur de flot** sert à indiquer au routeur que le paquet fait partie d'un flux qui doit avoir un traitement spécial (QoS particulière).
- **nombre de sauts** : remplace le TTL ipv4
- En Ipv6 on utilise la **plus petite taille de la taille maximale des paquets** (MTU) transportables par tous les réseaux traversés. En effet, si on considère que le paquet doit être intégralement reçu pour être renvoyé, un petit paquet mettra moins de temps à traverser un routeur qu'un gros et le temps d'acheminement d'un ensemble de petits paquet sera moins long que celui d'un gros (c'est aussi pour cela que les cellules d'ATM sont si petites).
- Le champ differentiate service indique, comme le TOS d'IPv4, une classe de service et une priorité.
- Ipv6 intègre de façon native des possibilités d'authentification ainsi que du chiffrement.

## 5.3 - L'adressage

Longueur : 128 bits

- Les 64 bits de poids fort pour la partie réseau et les 64 de poids faibles pour l'identifiant machine (interface).
- L'identifiant machine :

        - soit l'adresse MAC avec une petite transformation (au format EUI64) : le deuxième bit du premier octet de l'adresse MAC est mis à 1 et on insère FFFE au milieu de l'adresse MAC. Exemple : 01:02:03:04:05:06 devient : 2102:03FF:FE04:0506 ;
        - soit l'identifiant est fixé ;
        - soit il est choisit aléatoirement par la machine au boot (en évitant les adresses du 1er type).
    - La partie réseau sur 64 bits :
        - Dans le cas d'adresses normales, les 3 premiers bits (appelés Format Prefix) sont 001, les 13 suivants concernent le Top Level Aggregator TLA ID dont la valeur est 0001, les 13 suvants le sub TLA ID (c'est à dire le n° du FAI du FAI) et les 19 derniers le Next Level Aggregator NLA ID (le n° du FAI). Une adresse normale commence donc par 2001.
        - Enfin les 16 bits de SLAID viennent compléter les 48 premiers bits.
        - D'autres types d'adresses commencent par :
            - 3FFE : préfixe réservés aux travaux de recherche sur IPv6 ;
            - 2002 : préfixe de d'adresses mappées sur Ipv4 : 6to4
            - FFxx : multicast
            - FF01 : poste
            - FF02 : LAN
            - FF03 : site
            - FF05 : global
            - FE80 : locale au LAN ne traverse aucun routeur
            - FEC0 : locale à un site (plusieurs LAN), à définir par l'admin.
            - ::0:@ipv4 : compatible IPv4 (tunnel)
            - ::FFFF:@ipv4 : @ mappée sur IPv4

Comme les adresses sont longues, il existe des règles d'écriture :

- on sépare chaque groupe de 2 octets par “:”
- on peut supprimer le ou les 0 en début de chaque groupe
- on peut supprimer un ensemble de zéros consécutifs par “::”

exemple : 2001:0102:0000:0102:0304:0506 devient 2001:102::102:304:506

Selon les besoins, les FAI peuvent attribuer un préfixe de moins de 64 bits, laissant l'administrateur du site gérer les derniers bits de l'identifiant réseau (à des fins de segmentation du réseau du site). Afin de connaître la taille du préfixe donné, on fait alors suivre (comme en ipv4) l'adresse IP de la longueur du préfixe. Exemple : 2001:102::102:304:506/64

D'autres adresses :

- ::1 loopback
- :: non spécifiée

## 5.4 - Autoconfiguration.

Une des particularités de Ipv6 est la possibilité qui est donnée à une machine de s'autoconfigurer. 3 scenarii :

- La machine récupère sur le réseau un préfixe. Pour cela elle doit envoyer une trame “Router Sollicitation”. Le routeur répond par “Router advertisement” en donnant le préfixe et l'adresse du routeur par défaut. Enfin elle construit son adresse soit à partir de l'@ MAC, soit aléatoirement.
- La machine fait une requête DHCPv6 (comme en ipv4).
- La machine n'a rien trouvé et s'invente une adresse LAN locale à partir de l'@ MAC.

## 5.5 - ICMPv6

- Détection d'erreurs ;
- tests de liaison (ping) ;
- Configuration automatique des équipements ;
- Gestion des groupes de multicast ;
- ICMPv6 reprend les fonctions du protocole  ARP.

## 5.6 - RIPv6 ou RIPng

Reprend les fonctionnalités de RIPv2 mais adaptées à ipv6.

## 5.7 - Transition ipv4 vers ipv6

Plusieurs mécanismes existent. Le plus fréquent est 6to4 et plus récemment 6to4rd pour « rapid deployement ». 6to4rd est un service inventé par Rémi Després pour aider l'opérateur Free à déployer Ipv6 sur son backbone. Cependant, 6to4 ne permet pas de faire le lien entre des hotes ipv6 pures et des hotes ipv4 pures.

6to4 permet de relier des hotes ipv6 isolés à des réseaux Ipv6 en utilisant au milieu le réseau ipv4.

- L'hôte isolé reçoit un préfixe ipv6 particulier puisqu'il n'est pas connecté au « monde ipv6 ». Ce préfixe est envoyé par le routeur 6to4 et construit en accolant le préfixe 2002 :: à l'adresse ipv4 publique du routeur. Par exemple si le routeur a comme adresse IP 193.49.200.204, l'hote reçoit 2002 :C131:C8CC ::/48 comme préfixe et construit normalement son adresse ipv6 avec son adresse MAC.
- L'hôte envoie normalement ses paquets ipv6 au routeur 6to4.
- Celui-ci encapsule les paquets ipv6 dans des paquets ipv4 dont le champ protocole transporté sera égale à 41.
- 2 cas se présentent alors :
  1) La destination est un hote isolé. Dans ce cas le routeur 6to4 déduit l'@ ipv4 du routeur 6to4 du correspondant par le préfixe ipv6 de l'@ destination.
  2) la destination est le réseau Ipv6 natif. Dans ce cas, le routeur 6to4 envoie son paquet à un relai v4/v6 dont il connait l'@ ipv4. Cependant, pour simplifier les configurations, si le réseau du FAI héberge un tel relai, le FAI a configuré son relai avec l'@ 192.88.99.1. qui est donc le routeur par défaut du routeur 6to4.

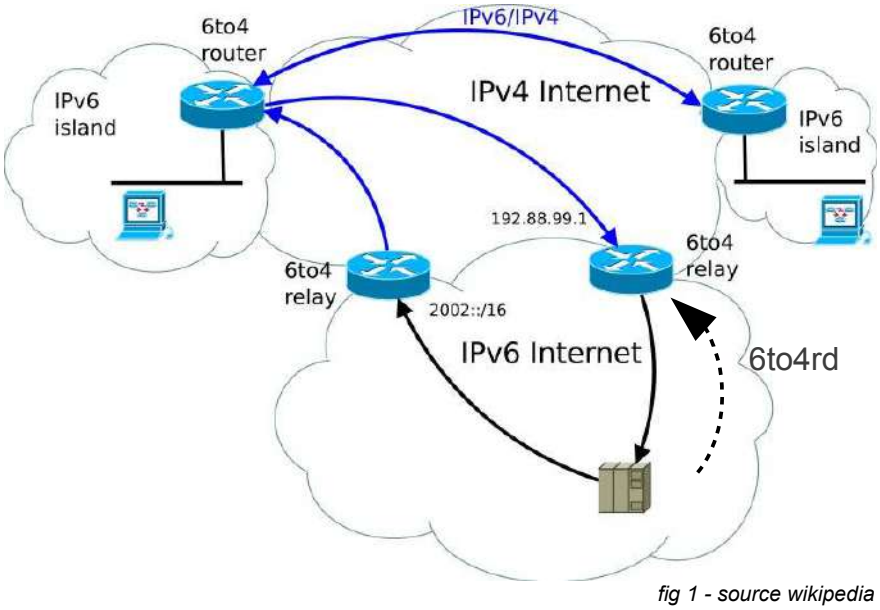


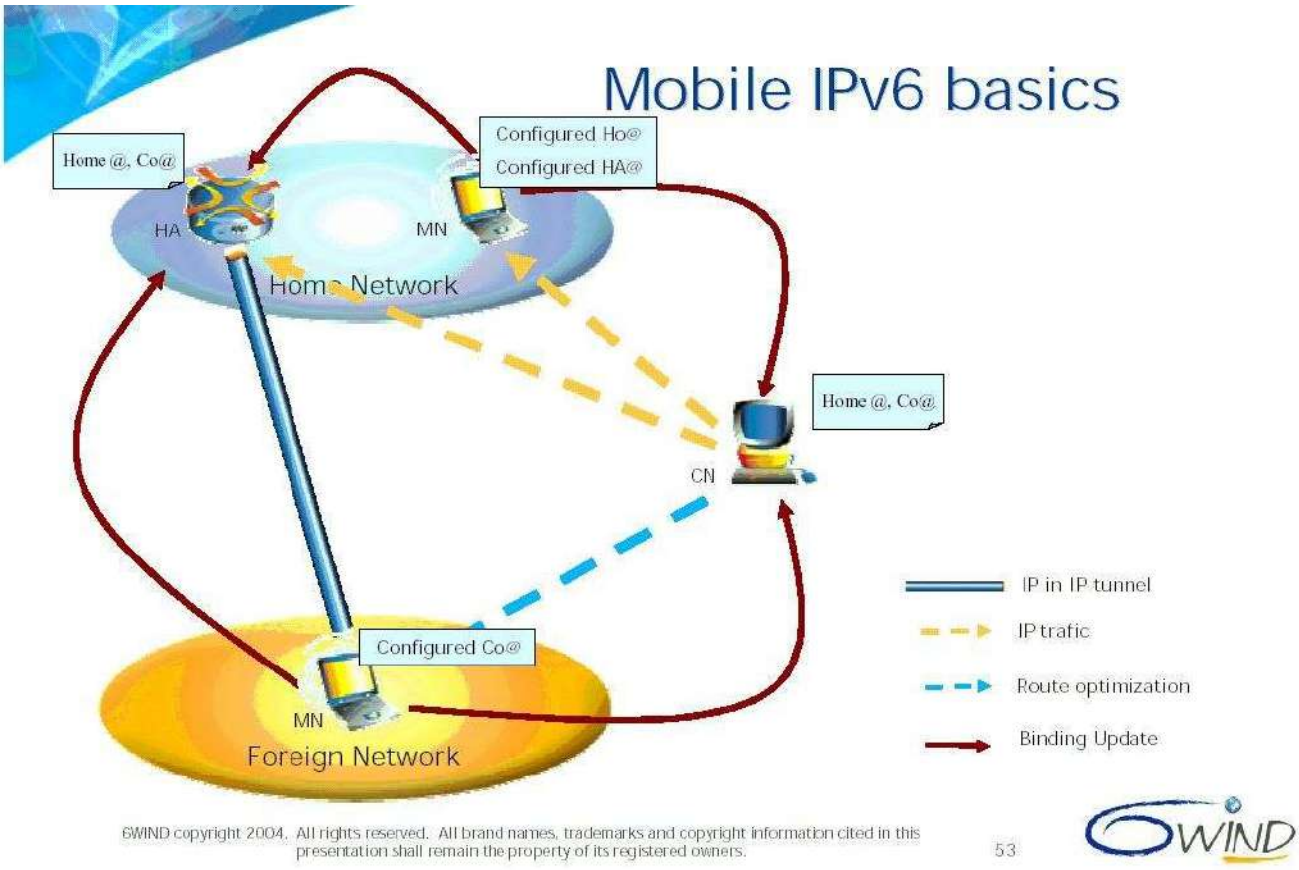

*fig 1 - source wikipedia*

Cependant, en voyageant dans le monde ipv6, le paquet sort du réseau du FAI et la réponse peut revenir par le réseau Ipv4 d'un autre FAI n'ayant pas ou mal implémenté 6to4. Il faut donc obliger le paquet à revenir dans le monde par le relai 6to4 d'origine. Pour cela il suffit au FAI non pas d'utiliser

les adresses en 2002 :: mais d'utiliser son propre préfixe Ipv6 et d'y accoler l'adresse IP du routeur 6to4. Ainsi le paquet reviendra au relai ipv6 du FAI.

## 5.8 - Mobilité :

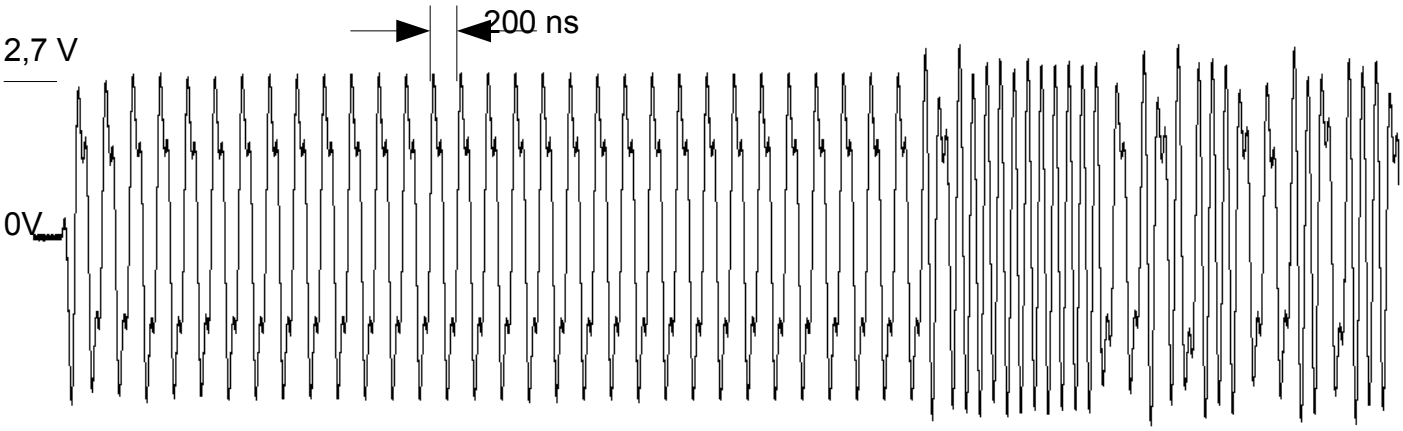
HN : Home network
FN : Foreign network
HA : Home agent
Ho@ : Home address
Co@ : Care of addresse (l'@ du mobile dans le réseau FN)
MN : Mobile Node
CN : Correspondant




Dans Ipv4 MN informait son HO qu'il avait bougé. Le HA retransmettait à CN tout le traffic de MN. Dans Ipv6, MN informe son HO qu'il a bougé mais aussi CN en lui envoyant sa Co@. Ceci permet un traffic direct entre CN et MN. Cela est rendu possible grâce à l'autoconfiguration.

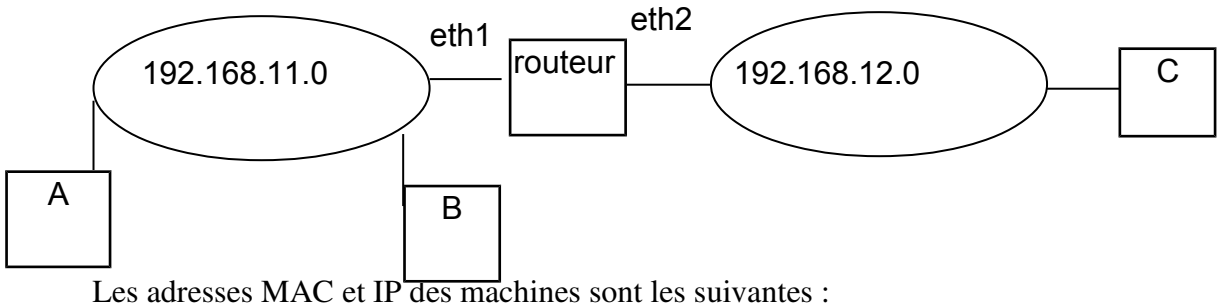
# Exercice 1

Voici une trame capturée par un oscilloscope. **Interpréter** la trame sachant que 0 est représenté par un front descendant, 1 est représenté par un front montant.
Les octets sont représentés au format LSb d'abord et les champs au format MSB d'abord.




# Exercice 2

Soit deux réseaux interconnectés par un routeur comme indiqué sur le schéma suivant




Les adresses MAC et IP des machines sont les suivantes :

Machine A : 00AA00 0000AA - 192.168.11.2
Machine B : 00BB00 0000BB - 192.168.11.3
Machine C : 00CC00 0000CC - 192.168.12.2
Interface eth1 du routeur : 001100 000011 - 192.168.11.1
Interface eth2 du routeur : 002200 000022 - 192.168.12.1

1) **Complétez** le tableau suivant. Vous indiquerez dans ce tableau quelles sont les adresses transportées dans les trames lorsque les machines s'envoient des informations.

| | Adresse MAC source | Adresse MAC destination | Adresse IP source | Adresse IP destination |
|---|---|---|---|---|
| Trame de A vers B vue de B | | | | |
| Trame de A vers C vue de A | | | | |
| Trame de A vers C vue de C | | | | |

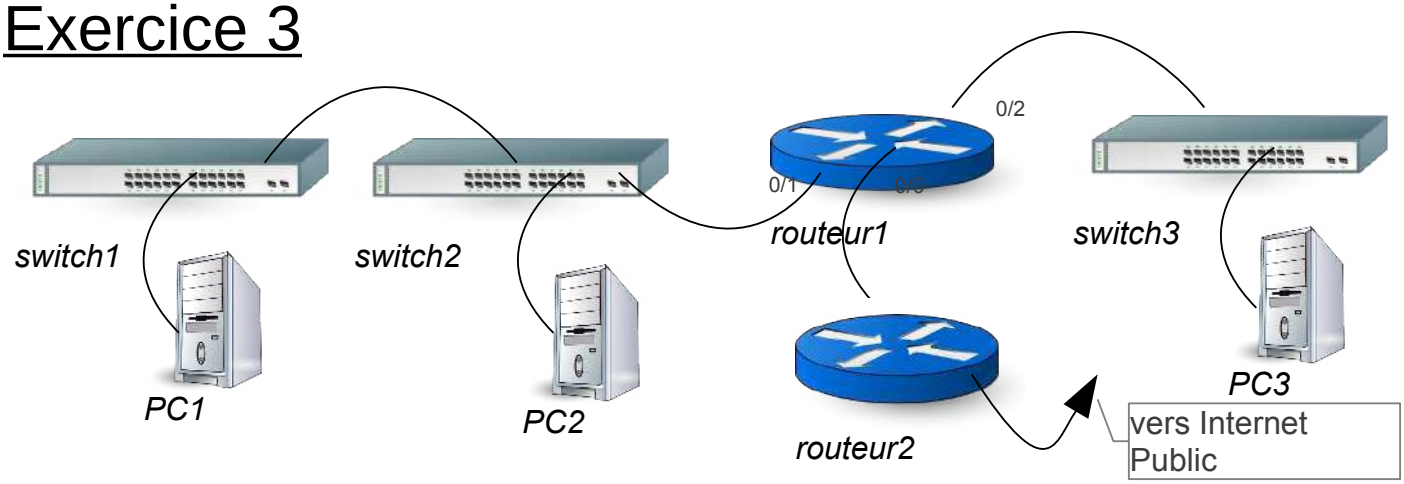
**2) Écrire** la table de routage de la machine C

3) **Reprendre** l'exercice en supposant maintenant que le routeur fait du NAT. A et B sont dans le

réseau interne, et C dans le réseau externe.
On suppose que A et B utilisent le même port source et qu'elles interrogent un serveur web sur la machine C. B a commencé à communiquer avec C avant A.

| | Adresse MAC source | Adresse MAC destination | Adresse IP source | Adresse IP destination | port source | port destination |
|---|---|---|---|---|---|---|
| Trame de A vers C vue de A | | | | | | |
| Trame de B vers C vue de C | | | | | | |
| Trame de A vers C vue de C | | | | | | |
| Trame de C vers A vue de C | | | | | | |
| Trame de C vers A vue de A | | | | | | |

# Exercice 3




Pour chacun des scénarios :

- Dessinez les réseaux logiques ;
- Donnez des adresses IP aux machines ;
- Donnez les netmask et les adresses de début et de fin de chaque réseau ;
- Donnez la table de routage de routeur1 et de PC3 ;

**scenario 1 :**

Le routeur1 appartient à l'entreprise et le routeur2 au fournisseur d'accès. L'entreprise dispose des adresses 82.12.12.64/26 pour que les PC puissent avoir accès à internet. Le routeur 1 a accès à internet par 82.12.12.1/26. Quelle pourrait être l'adresse donnée à l'interface 0/0 ?

**scenario 2 :**

Idem scénario 1, mais PC1 est sur un réseau privé alors que PC2 et PC3 ont un accès direct à internet. PC1 doit pouvoir dialoguer avec PC2 et PC3. Donnez une solution en incluant un routeur de plus. (Pensez aux messages ICMP redirect).

**scenario 3 :**

Idem scénario *1, mais PC1 est sur un réseau privé alors que PC2 et PC3 ont un accès direct à internet. PC1 doit pouvoir dialoguer avec PC2 et PC3. Donnez une solution à base d'IP alias. Cette solution est-elle plus efficace que la solution précédente ?

# Exercice 4

Voici un extrait d'échange d'une connexion SSH. Wireshark a été lancé sur la machine d'adresse 192.168.0.6 où le client SSH tournait. Au moment de la capture, l'utilisateur avait le doigt enfoncé sur une touche du clavier et le clavier fonctionnait donc en mode de répétition automatique. Pendant cet échange, un problème est survenu sur le réseau. On rappelle qu'en ssh, l'envoi d'un caractère est codé sur 48 octets.

1) Décrire ce qu'il s'est passé en commentant brièvement chaque trame.

2) Commentez également l'augmentation du temps entre deux envois consécutifs à partir de la trame 328.

| N° | Time | IP source | IP dest | Infos TCP |
|---|---|---|---|---|
| 324 | 151.8 | 192.168.0.6 | 193.49.200.204 | 48424 > ssh [ACK] Seq=4808 Ack=9157 Len=48 |
| 325 | 151.882185 | 193.49.200.204 | 192.168.0.6 | ssh > 48424 [ACK] Seq=9157 Ack=4856 Len=48 |
| 326 | 151.882285 | 192.168.0.6 | 193.49.200.204 | 48423 > ssh [ACK] Seq=4856 Ack=9205 Len=0 |
| 327 | 152.067592 | 192.168.0.6 | 193.49.200.204 | 48423 > ssh [ACK] Seq=4856 Ack=9205 Len=48 |
| 328 | 152.287641 | 192.168.0.6 | 193.49.200.204 | 48423 > ssh [ACK] Seq=4904 Ack=9205 Len=48 |
| 329 | 152.295564 | 192.168.0.6 | 193.49.200.204 | 48423 > ssh [ACK] Seq=4856 Ack=9205 Len=96 [Retransmission] |
| 330 | 152.523691 | 192.168.0.6 | 193.49.200.204 | 48423 > ssh [ACK] Seq=4952 Ack=9205 Len=48 |
| 331 | 152.751709 | 192.168.0.6 | 193.49.200.204 | 48423 > ssh [ACK] Seq=4856 Ack=9205 Len=144 [Retransmission] |
| 332 | 153.003688 | 192.168.0.6 | 193.49.200.204 | 48423 > ssh [ACK] Seq=5000 Ack=9205 Len=96 |
| 333 | 153.664074 | 192.168.0.6 | 193.49.200.204 | 48423 > ssh [ACK] Seq=4856 Ack=9205 Len=240 [Retransmission] |
| 334 | 153.775611 | 192.168.0.6 | 193.49.200.204 | 48423 > ssh [ACK] Seq=5096 Ack=9205 Len=144 |
| 335 | 155.487738 | 192.168.0.6 | 193.49.200.204 | 48423 > ssh [ACK] Seq=4856 Ack=9205 Len=384 [Retransmission] |
| 336 | 155.764429 | 192.168.0.6 | 193.49.200.204 | 48423 > ssh [ACK] Seq=5240 Ack=9205 Len=384 |
| 337 | 159.135589 | 192.168.0.6 | 193.49.200.204 | 48423 > ssh [ACK] Seq=4856 Ack=9205 Len=768 |
| 338 | 159.174048 | 193.49.200.204 | 192.168.0.6 | ssh > 48423 [ACK] Seq=9205 Ack=5624 Len=48 |
| 339 | 159.174107 | 192.168.0.6 | 193.49.200.204 | 48423 > ssh [ACK] Seq=5624 Ack=9253 Len=576 |
| 340 | 159.176190 | 193.49.200.204 | 192.168.0.6 | 48423 > ssh [ACK] Seq=9253 Ack=5624 Len=48 |
| 341 | 159.177424 | 193.49.200.204 | 192.168.0.6 | 48423 > ssh [ACK] Seq=9301 Ack=5624 Len=48 |
| 342 | 159.177496 | 192.168.0.6 | 193.49.200.204 | 48423 > ssh [ACK] Seq=6200 Ack=9349 Len=0 |
| 343 | 159.212267 | 193.49.200.204 | 192.168.0.6 | 48423 > ssh [ACK] Seq=9349 Ack=6200 Len=624 |
| 344 | 159.213468 | 193.49.200.204 | 192.168.0.6 | 48423 > ssh [ACK] Seq=9973 Ack=6200 Len=48 |
| 345 | 159.213539 | 192.168.0.6 | 193.49.200.204 | 48423 > ssh [ACK] Seq=6200 Ack=10021 Len=0 |

# Exercice 5

Donnez un exemple d'échange avec un démarrage de connexion TCP en cold start. Intéressez-vous au mécanisme de fenêtre.