Administration et Sécurité des Réseaux

Ressource R102
Principes et architectures des réseaux

Plan du cours

- Architecture réseau
  - Définitions & généralités
  - Modèles de référence : OSI, TCP-IP
  - Encapsulation
- Analyse de protocoles
  - ARP
  - IP / ICMP
  - TCP / UDP
- Topologies de réseaux
  - Physique / Logique
- Normalisation des technologies
  - RFC / normes

Chapitre 1

- Architecture Réseau
  - Généralités & définitions
  - Modèle en couches
  - Les standards
    - OSI
    - TCP – IP

Généralités & Définitions

- A quoi sert un réseau ?
  - Transporter de l'information
  - Constitué d'un support et d'un message
  - Analogie :
    - Rail + Train : Support
    - Marchandises : Mess
  - On peut changer ?
    - Support : oui
    - Message : heureusement
    - Indépendance

Généralités

- Fonctionnalités du réseau :

Acheminement local

Fiabilité

Sécurité

Multi-utilisateurs

Acheminement distant

Compatibilité matérielle

Adaptabilité

Disponibilité du destinataire

Gestion du flux

Reprise sur erreur

Hétérogénéité réseau

etc ...

Identification

Compatibilité données

Simplicité

Généralités

- Semble compliqué
  - Comment gérer tout ces problèmes ?
- Historique
  - 1961 : Leonard Kleinrock du MIT propose un papier sur la commutation de paquets
  - 1962 : J.C.R. Licklider du MIT écrit des mémos sur le « Galactic Network » et embarque des personnes du DARPA et du MIT dans l'histoire.

DARPA : Defense Advanced Research Projects Agency

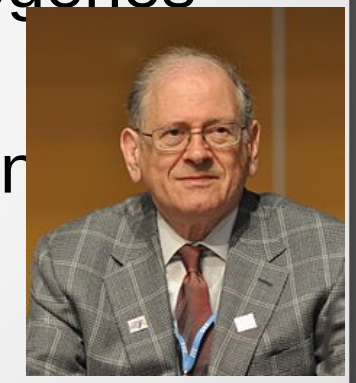
Suite de l'histoire

- 1967 : Roberts rejoint le DARPA et publie sur l'ARPANET. D'autre travaux similaires existent
  - Donald Davis, labo NPL ((National Physical Laboratory Royaume Uni)
  - Paul Baran, RAND militaires transmission de la voix

Sources : http://aeris.11vm-serv.net/cours/internet/histoire.html
http://www.internetsociety.org/internet/what-internet/history-internet/brief-history-internet

Suite ...

- 1969 ARPANET opérationnel relie 4 instituts universitaires
  - Communication entre ordinateurs et calcul partagé
- 1970 les laboratoires de recherche rejoignent le réseau Arpanet.
  - Avantage : interconnexion de machines hétérogènes (Unix, IBM,...)
  - Seule condition même mode de communication **protocole** standard universel
  - NCP (Network Control Protocol).



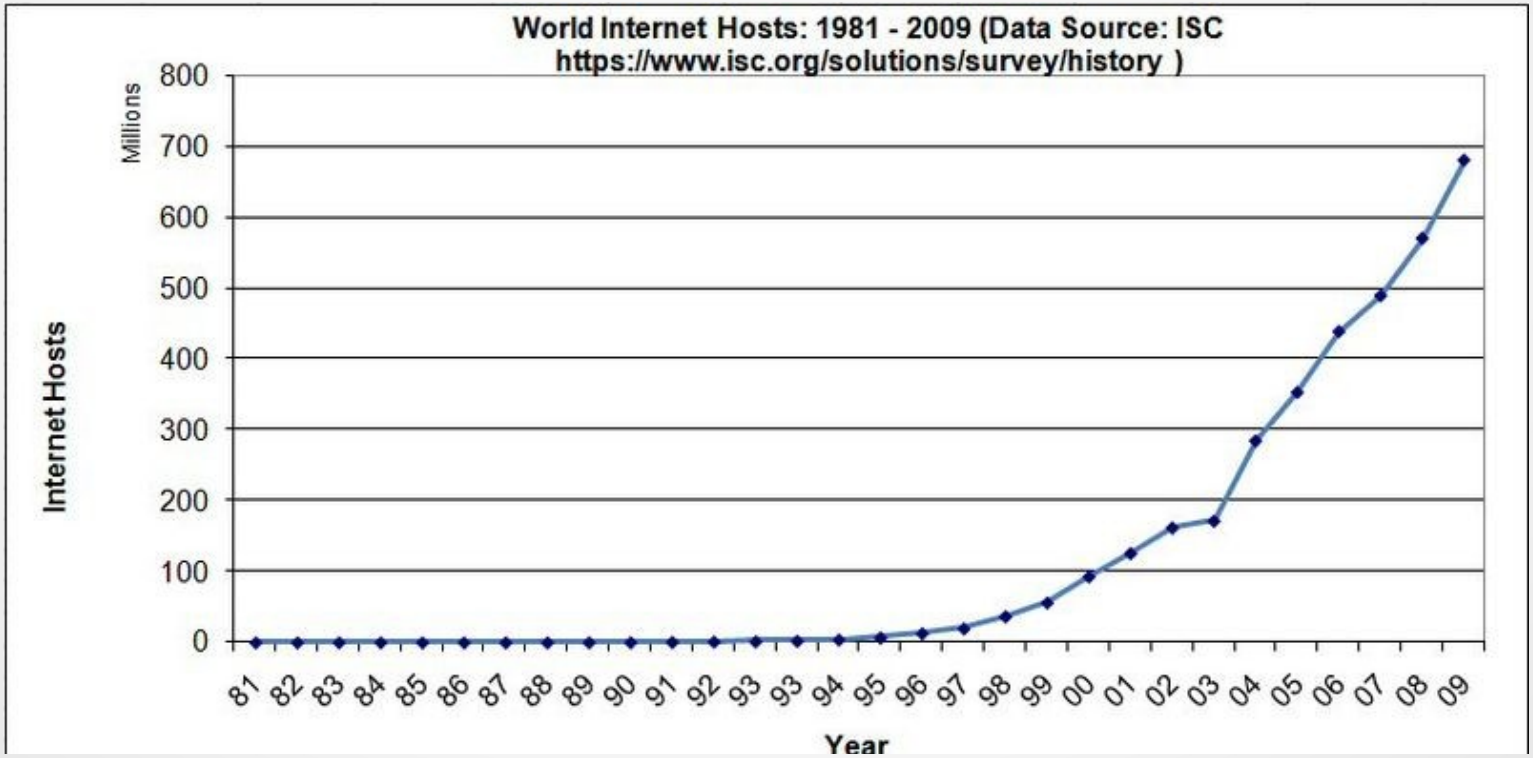
- 1972 NCP limité : Bob Kahn travaille sur une évolution
  - Internetting
- 1983 NCP disparaît pour TCP/IP

Règles pour établissement de TCP/IP

- 4 Règles de base ennoncées par Bob Kahn :
  - Each distinct network would have to stand on its own and no internal changes could be required to any such network to connect it to the Internet.
  - Communications would be on a best effort basis. If a packet didn't make it to the final destination, it would shortly be retransmitted from the source.
  - Black boxes would be used to connect the networks; these would later be called gateways and routers. There would be no information retained by the gateways about the individual flows of packets passing through them, thereby keeping them simple and avoiding complicated adaptation and recovery from various failure modes.
  - There would be no global control at the operations level.
- A partir de là … succès

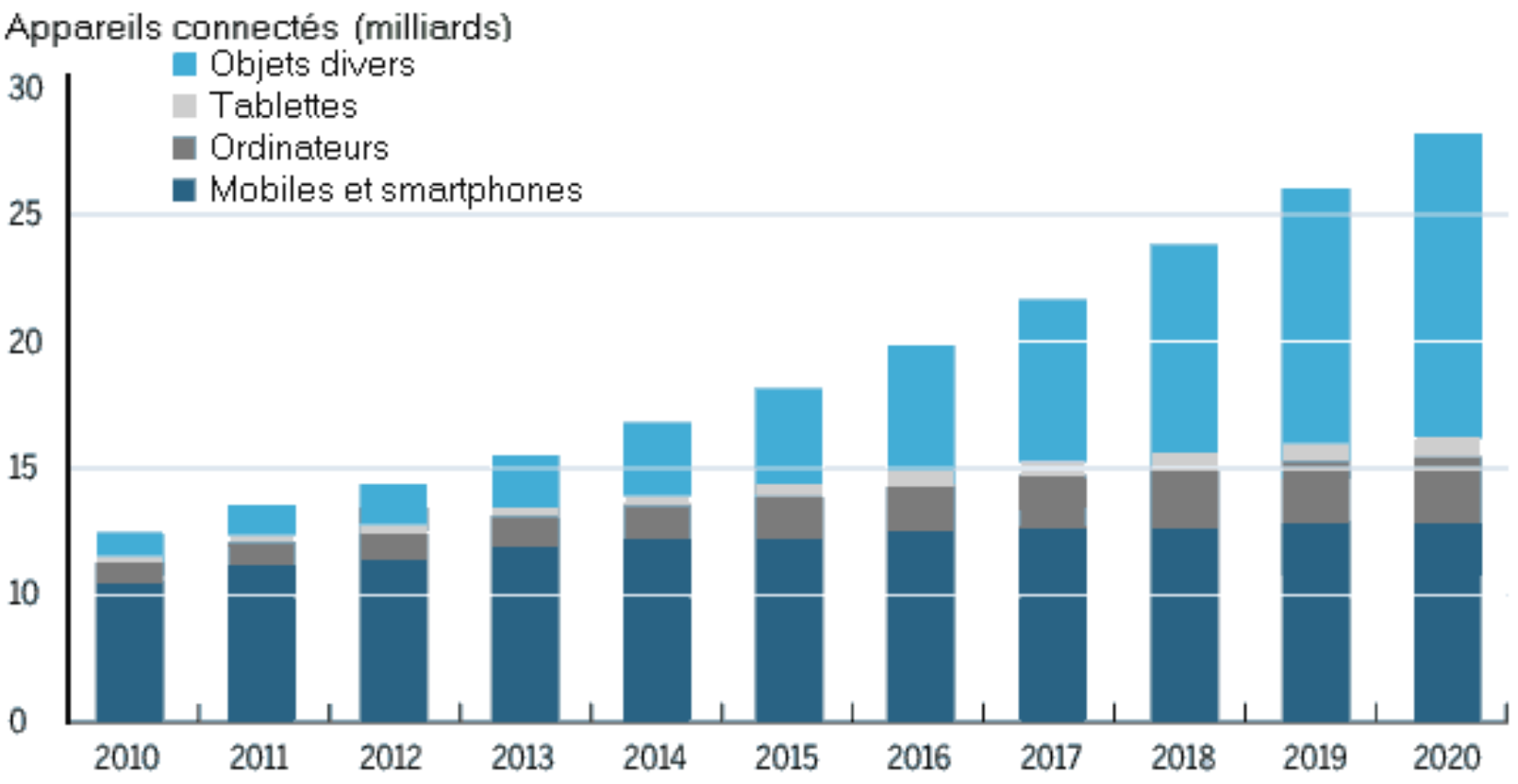
Connectés sur Internet

- Au début :



Connectés sur Internet

- Maintenant :



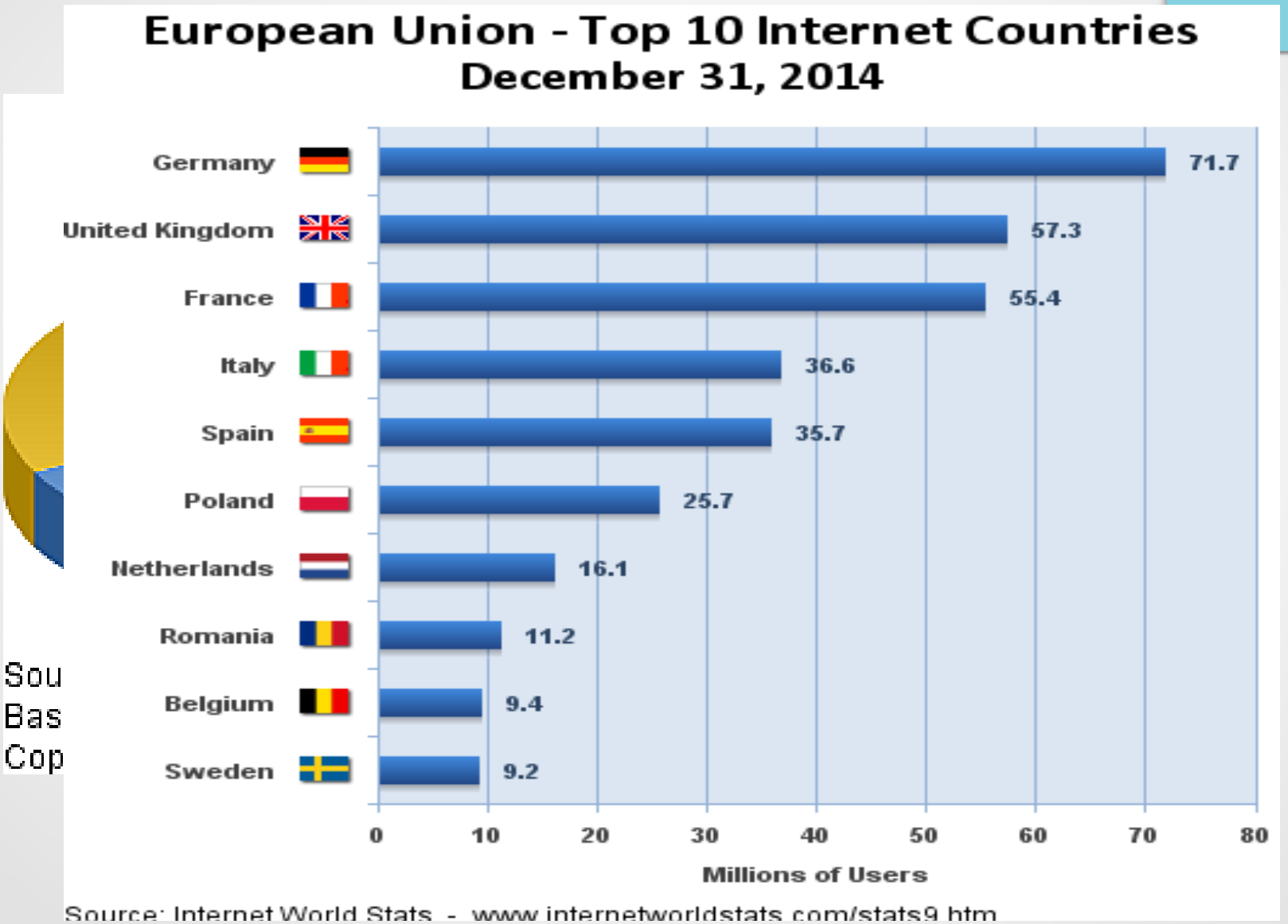
En France

- Jusqu'en 2010

Année	Internautes	Population	% population	Source
2000	8 500 000	58 879 000	14.4 %	ITU
2004	24 848 009	60 293 927	41.2 %	Nielsen Net
2006	30 837 595	61 350 009	50.3 %	Nielsen Net
2007	32 925 953	61 350 009	53.7 %	Nielsen Net
2008	36 153 327	62 177 676	58.1 %	Nielsen Net
2010	44 625 300	64 768 389	68.9 %	ITU

- Maintenant :
  - http://www.arcep.fr/index.php?id=10292

Position de la France dans le monde



Source : http://www.internetworldstats.com/stats4.htm#europe

Mais alors c'est quoi l'Architecture du réseau ?

- Organiser les fonctionnalités
  - Qui fait quoi et comment ?
- Organiser la structure matérielle
  - Comment on relie tout le monde ?
  - Topologie physique (Chap 3)
- Organiser la structure fonctionnelle
  - Comment est-ce que tout ça fonctionne ?
  - Topologie logique (Chap 3)

Organisation des fonctionnalités

- Objectifs :
  - Système qui peut évoluer sans avoir tout à refaire
  - Indépendance des fonctionnalités
  - Indépendance vis-à-vis du matériel
- Moyens :
  - Définir des règles de communication : **Protocoles**
  - Structurer les fonctionnalités : modèle en **Couches**
  - Structurer les échanges entre couches : **Services**
- Exemple ...

Analogie commerciale

- Vente d'un produit
- Si le patron s'occupe de tout → Danger
  - Lois à respecter
  - Devises
  - Format des échanges
  - Langue
  - ...

Sources : Sumo http://anatomias.mediasmile.net/pretoebranco.htm
Super dupond :Goetlib

Analogie commerciale

- On s

Cherche :
- le meilleur rapport qualité prix
- la meilleure disponibilité
- le marché
- distributeurs

Vérifie :
- les réglementations du pays
- normes à respecter

- Définit une langue commune
- Assure la traduction

Se charge :
- de la mise en forme
- de l'envoi de la correspondance

PATRON

SERVICE JURIDIQUE

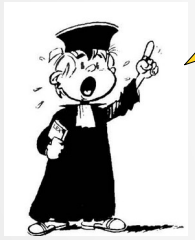
LOIS

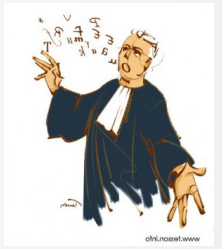
PATRON

SERVICE JURIDIQUE

TRADUCTEUR

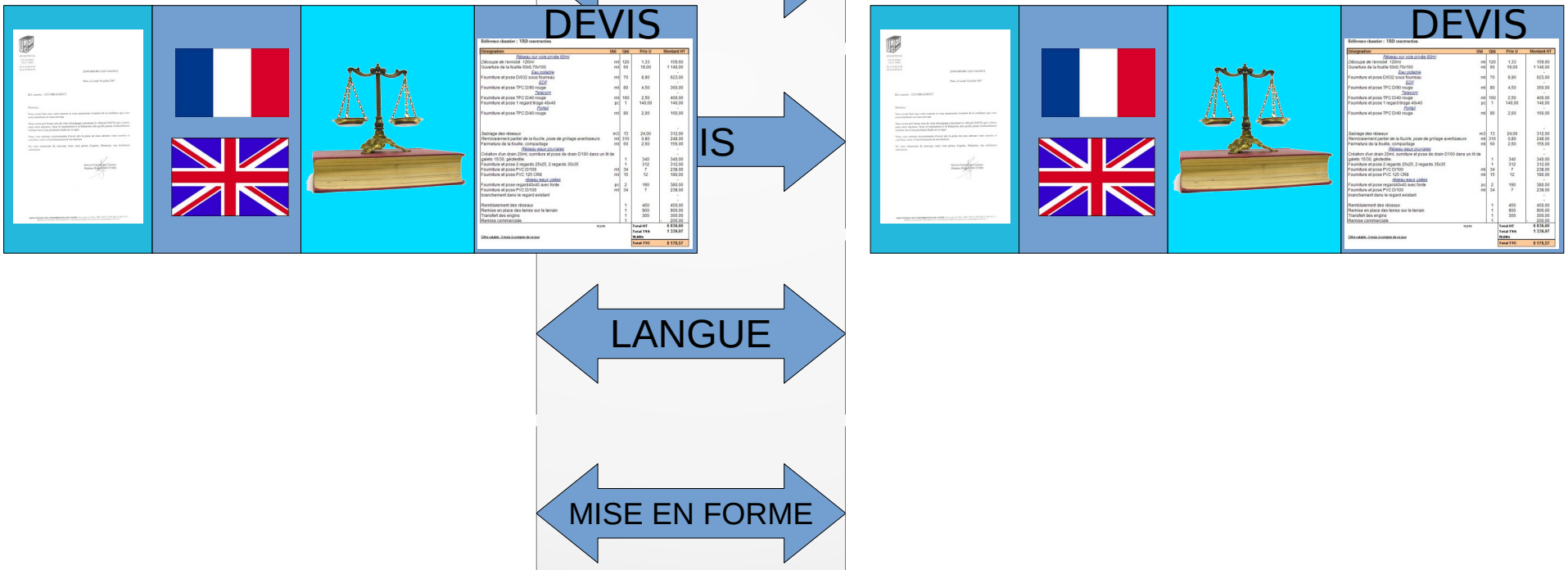
SECRETARIAT





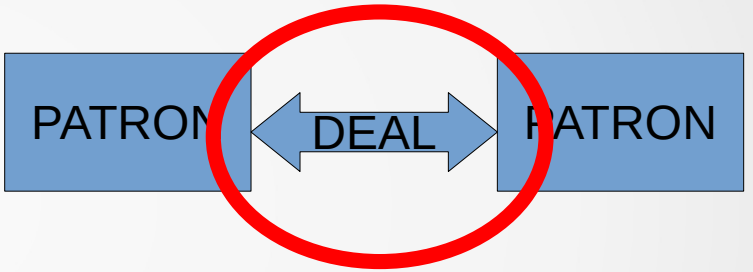
Analogie commerciale

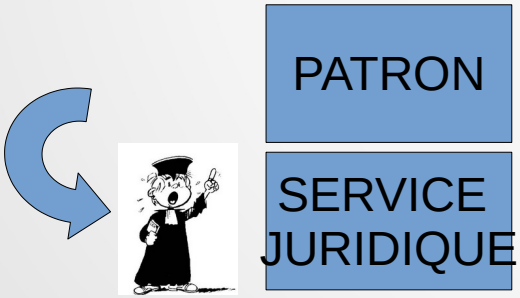
- Comment ça marche ?



Termes associés

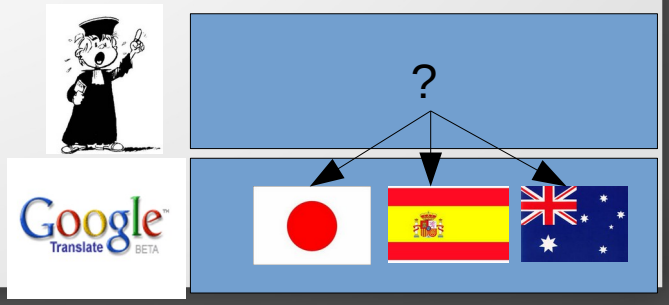
- Architecture en **Couches (layer)**
- Dialogue entre couches paires
  - Dialogue "virtuel"
  - Définit un **Protocole**





- Liaison entre couches : **Services**
  - La couche N l'utilise
  - La couche N-1 le fournit

- Quel service utilise-t-on ?
  - **Point d'accès** au service

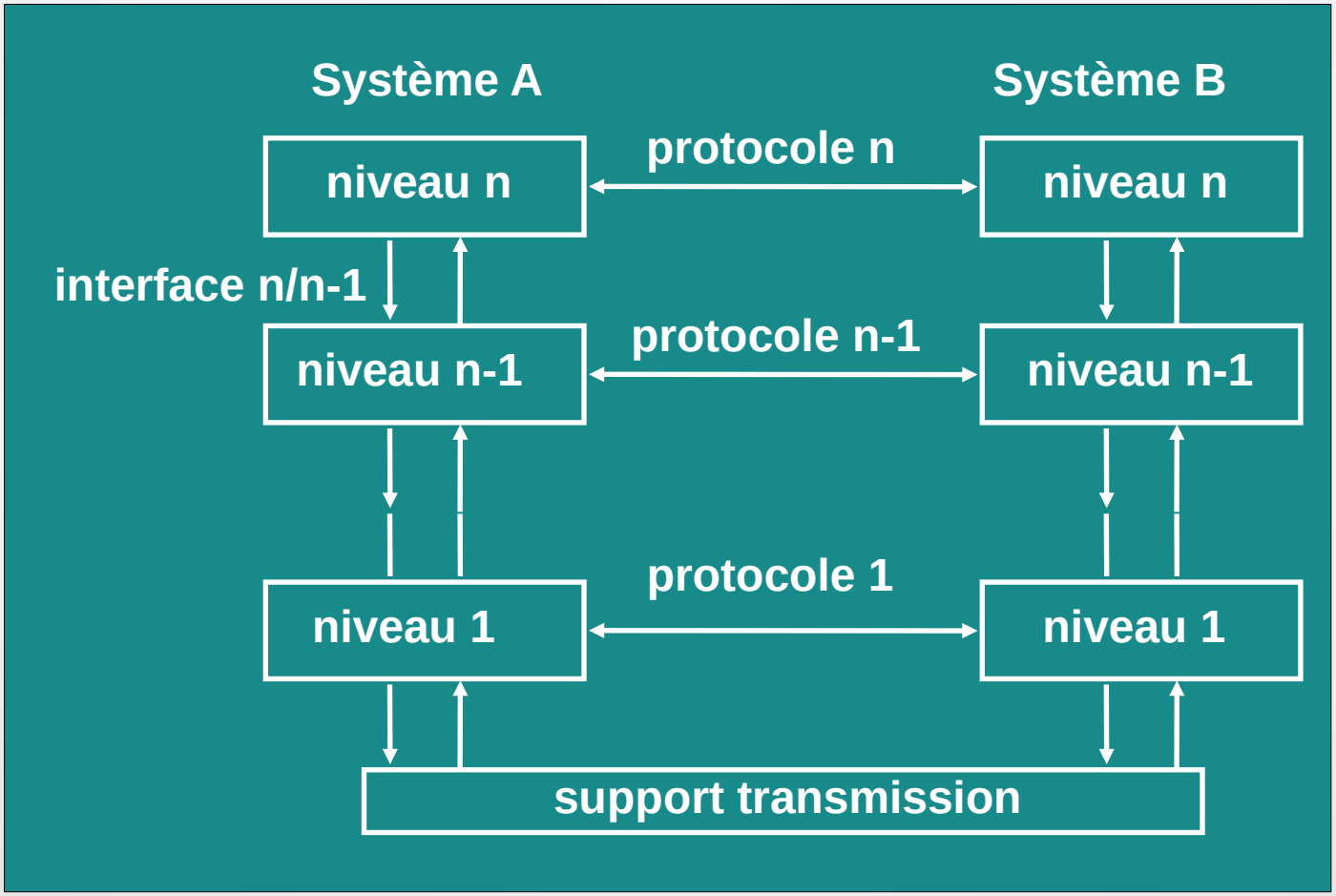


Termes associés

- Protocole
  - Ensemble de règles/informations spécifiant :
    - Format des unités de données (trames, paquets, etc.)
    - Séquencement normal des messages (ordre des messages lors de connexions)
- Service
  - Ensemble de fonctionnalités proposées par une couche.
  - On les utilise par le biais de **primitives** simples : Demande, Indication, Réponse, Confirmation.
  - Mnémonique : téléphone
- Interface
  - Moyen concret d'accéder au service (**point d'accès**)

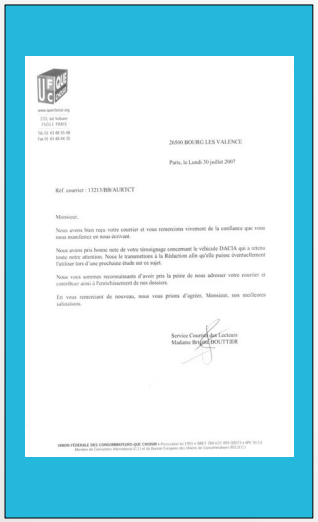
Schéma général

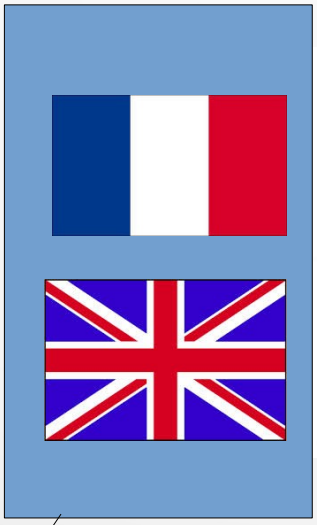
- Interaction des couches



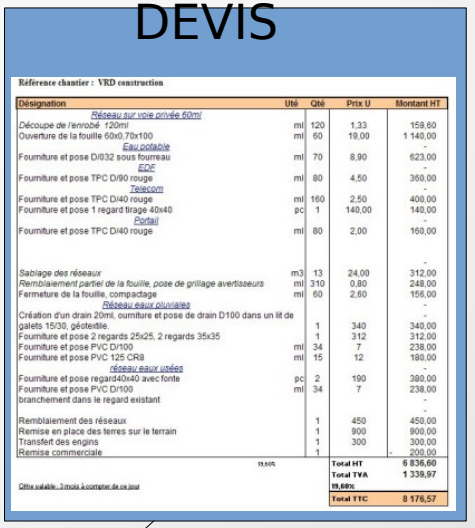
Encapsulation

- Chaque couche rajoute des informations (protocole) / aux couches précédentes.
- Phénomène poupée russes ou **d'encapsulation**





DEVIS



Référence chantier : VRD construction

Désignation	Uté	Qté	Prix U	Montant HT
Réseau sur voie privée 60ml				
Découpe de l'enrobé 120ml	ml	120	1,33	159,60
Ouverture de la fouille 60x0,70x100	ml	60	19,00	1 140,00
Eau potable				-
Fourniture et pose D/032 sous fourreau	ml	70	8,90	623,00
EDF				-
Fourniture et pose TPC D/90 rouge	ml	80	4,50	360,00
Telecom				-
Fourniture et pose TPC D/40 rouge	ml	160	2,50	400,00
Fourniture et pose 1 regard tirage 40x40	pc	1	140,00	140,00
Portail				-
Fourniture et pose TPC D/40 rouge	ml	80	2,00	160,00
				-
Sablage des réseaux	m3	13	24,00	312,00
Remblaiement partiel de la fouille, pose de grillage avertisseurs	ml	310	0,80	248,00
Fermeture de la fouille, compactage	ml	60	2,60	156,00
réseau eaux pluviales				-
Création d'un drain 20ml, ourniture et pose de drain D100 dans un lit de galets 15/30, géotextile.		1	340	340,00
Fourniture et pose 2 regards 25x25, 2 regards 35x35		1	312	312,00
Fourniture et pose PVC D/100	ml	34	7	238,00
Fourniture et pose PVC 125 CR8	ml	15	12	180,00
réseau eaux usées				-
Fourniture et pose regard40x40 avec fonte	pc	2	190	380,00
Fourniture et pose PVC D/100 branchement dans le regard existant	ml	34	7	238,00
				-
Remblaiement des réseaux		1	450	450,00
Remise en place des terres sur le terrain		1	900	900,00
Transfert des engins		1	300	300,00
Remise commerciale		1		-200,00
	19,60%		Total HT	6 836,60
			Total TVA	1 339,97
			19,60%	
			Total TTC	8 176,57

Offre valable : 2 mois à compter de ce jour

Données présentation Mise en forme

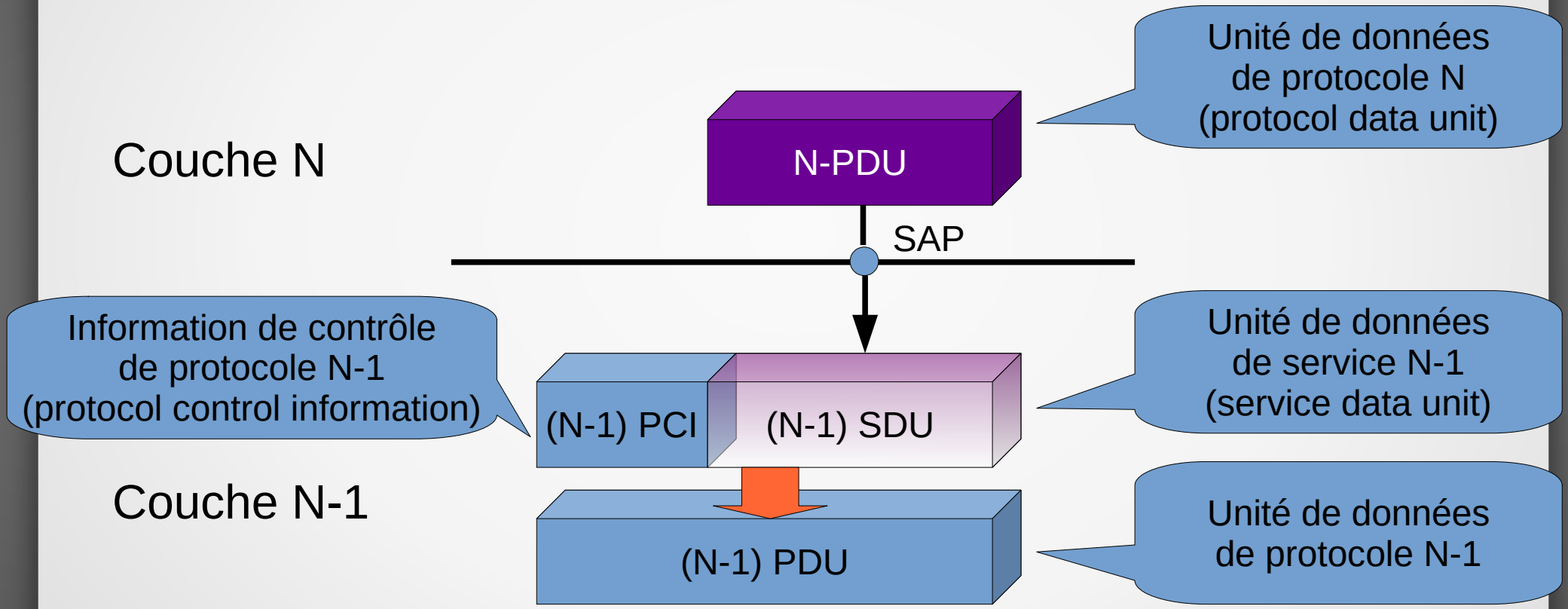
Données langues

Données juridiques

Données utiles : DEAL

Terminologie de l'encapsulation

- PDU, SDU, PCI ??

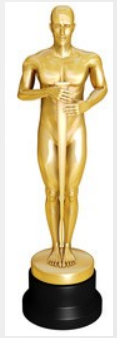


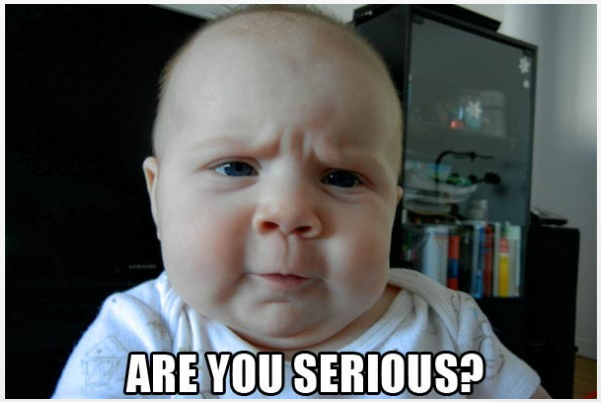
Mais alors ...

- On envoie autre chose que des données **utiles** ?
- Notion de débit utile
  - Quantité du débit utilisé pour envoyer les données (pas les infos de protocole)
- Ça implique quoi ?
  - Non prolifération des couches

Modèles d'architecture de référence

- Existe-t-il des modèles standards ? Oui !
  - Modèle OSI
  - Modèle TCP-IP (Internet)
- And the oscar goes to …
  - TCP-IP
- MAIS …
  - OSI fait référence





Modèle d'architecture de référence

- Modèle OSI
  - Open System Interconnection
  - Référence ISO 7498
  - Modèles à **7 couches**
  - Dernière révision date de 1994
- Fait toujours référence … mais peu implémenté en pratique

Modèle OSI

- 7 Application
- 6 Présentation
- 5 Session
- 4 Transport
- 3 Réseau
- 2 Liaison
- 1 Physique

Milieu : support de transmission

Type de données	Fonction
Données	Point d'accès aux services réseaux
Données	Conversion données manipulées au niveau applicatif et chaînes d'octets effectivement transmises.
Données	Gère session entre applications
Segment	Connexion de bout en bout (processus) / Contrôle de flux
Paquet/Datagramme	Parcours des données et adressage logique
Trame	Adressage physique / Acheminement local
Bit	Transmission des signaux binaires

Modèle OSI

- 4 couches Hautes (7,6,5,4)
  - 7 : Applications réseaux
  - 6 : Présentation gère le codage, la représentation des données applicatives (sémantique)
  - 5 : Session gère la synchronisation des communications (n'importe qui peut parler) et reprise sur erreur
  - 4 : Transport gère la communication de bout en bout entre processus
- Exemple de protocoles
  - FTP, DNS, DHCP
  - ASCII, Unicode
  - AppleTalk, NetBios
  - TCP, UDP

Modèle OSI

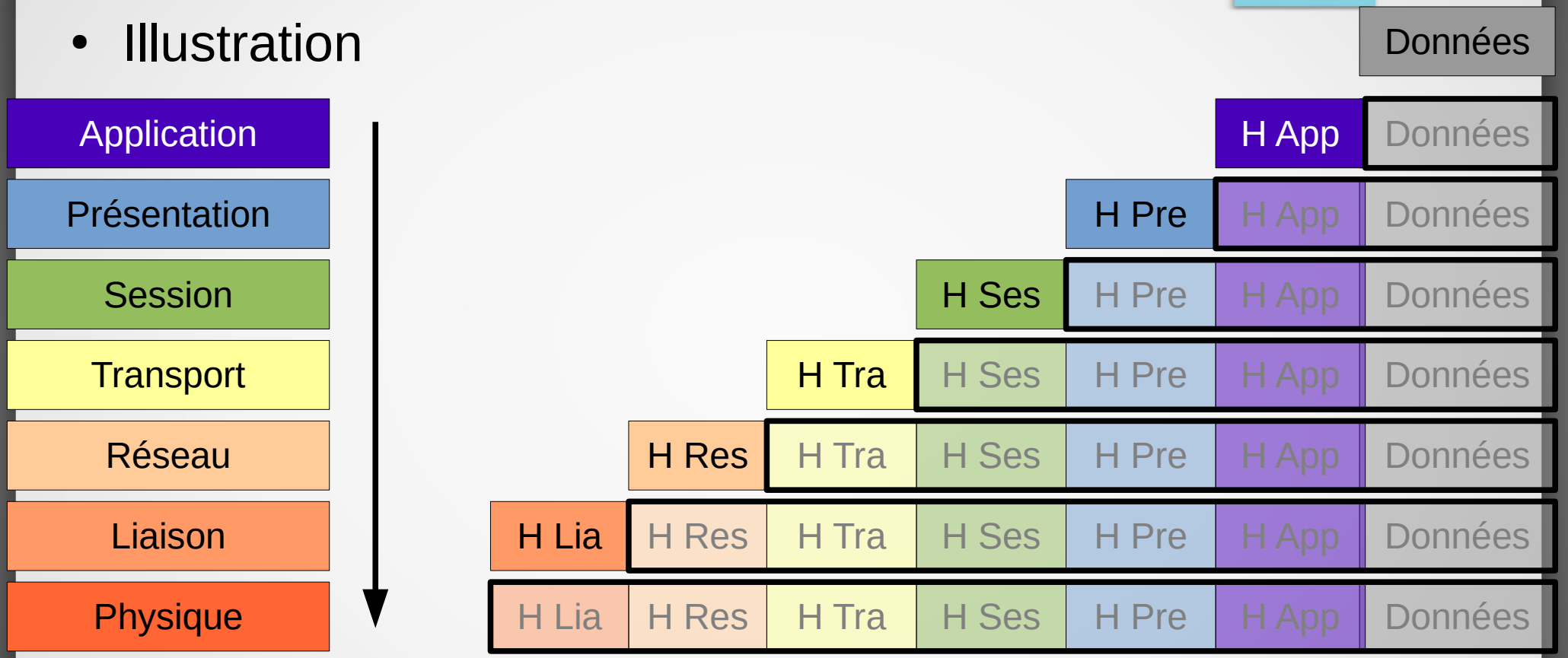
- 3 couches matérielles (3, 2, 1)
  - Réseau : gère les communications de proche en proche, adressage, routage
  - Liaison : gère les communications entre machine reliées par un support physique
  - Physique : transmission de signaux entre les machines.
- Exemple de protocoles
  - IP, ICMP, IGMP, NetBEUI, RIP, OSPF, EIGRP
  - Ethernet, PPP, MPLS, FDDI, FrameRelay
  - Codage NRZ, Manchester, Miller, RS232, 100 baseTx

Application de l'encapsulation

- Seules les couches physiques communiquent directement
- Comment discutent les entités de la couche liaison ? (envoyer des trames)
  - Demande de service à la couche physique
- Comment discutent les entités de la couche réseau ?
  - Demande de service à la couche liaison
- Et ainsi de suite ...

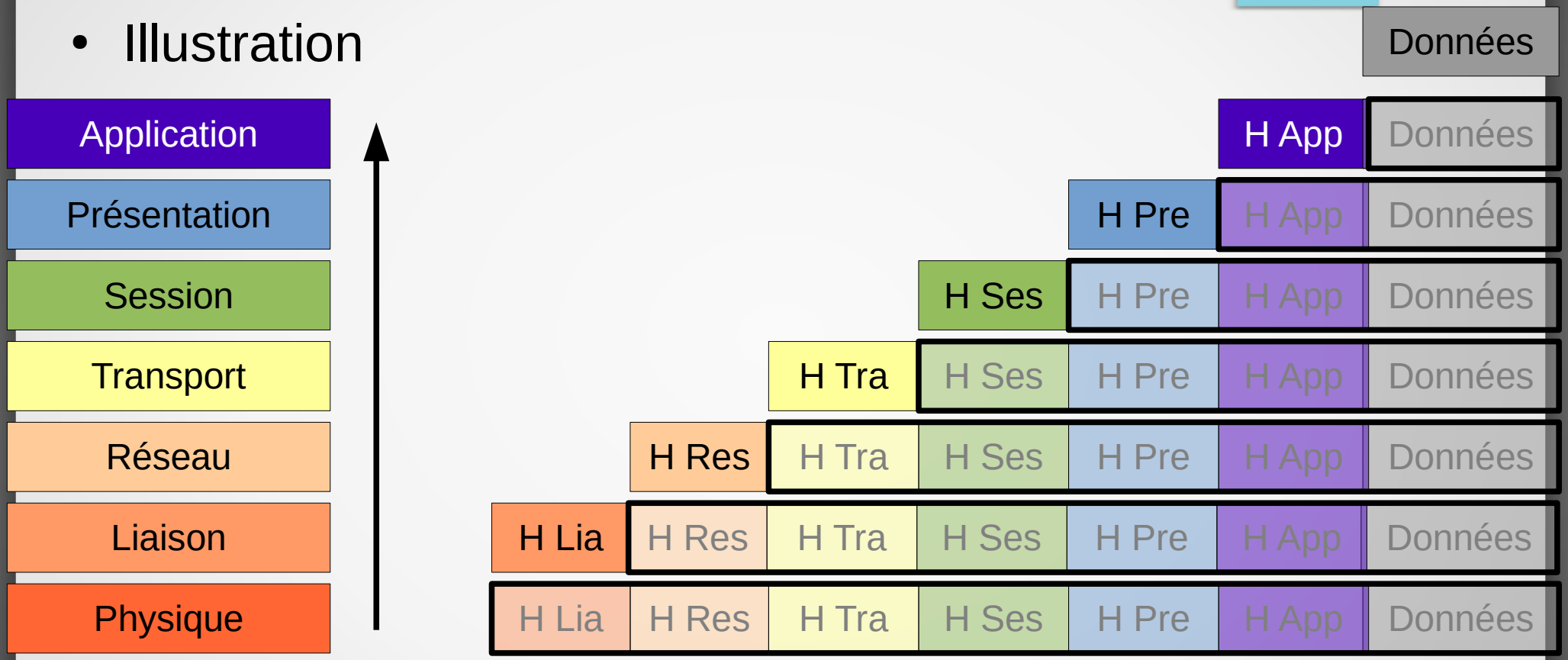
Application de l'encapsulation

- Illustration



Application de la "désencapsulation"

- Illustration



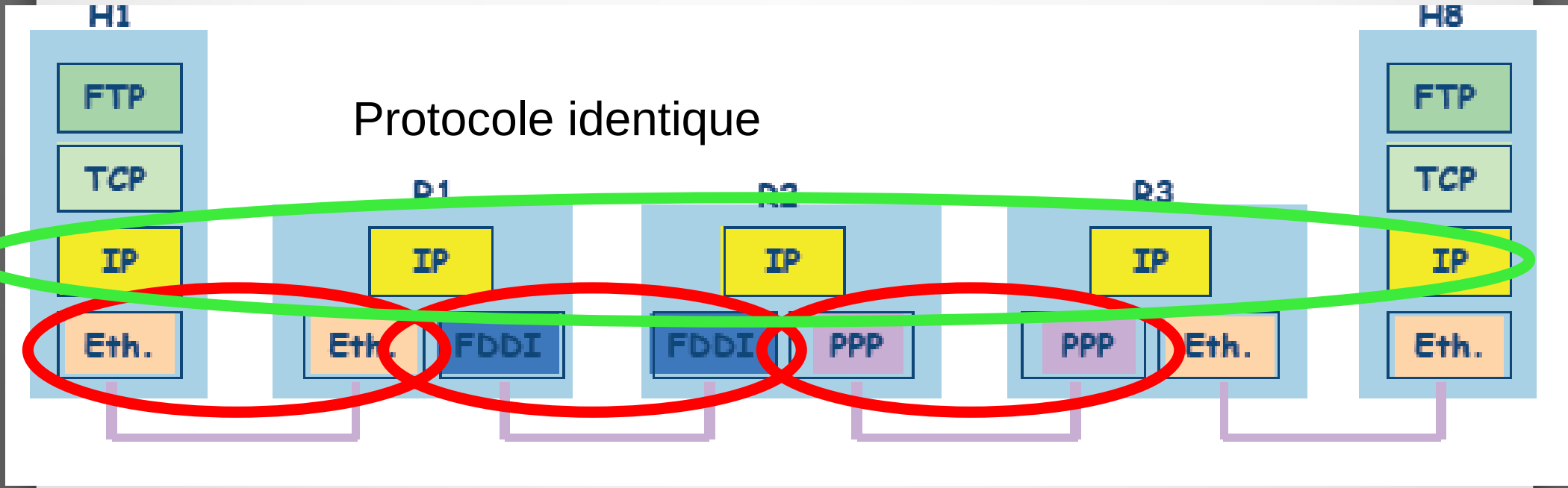
SDU d'une couche

Modèle Internet (TCP-IP)

- Modèle **Internet**
- Basé sur le couple de protocoles TCP-IP
- Simplification du modèle OSI (moins de couches)
  - Regroupement des couches hautes 5, 6 et 7
- Utilisé car répandu et développé dans le monde universitaire
- Bénéficie des développement du monde Unix et de la croissance du réseau Internet

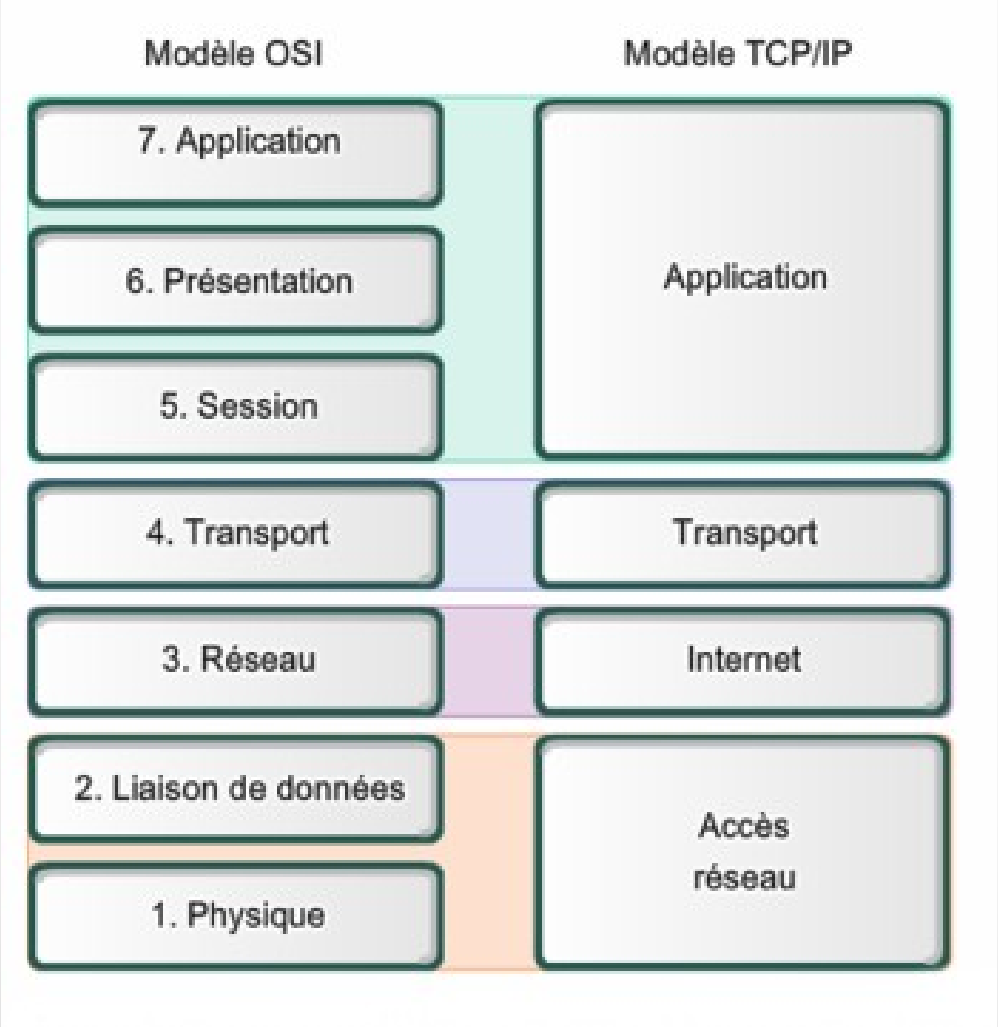
Modèle Internet (TCP-IP)

- Les couches 1 et 2 ne sont pas définies
- Objectif : assurer une homogénéité dans le réseau



Modèle Internet (TCP-IP)

- Modèle 4 ou 5 couches

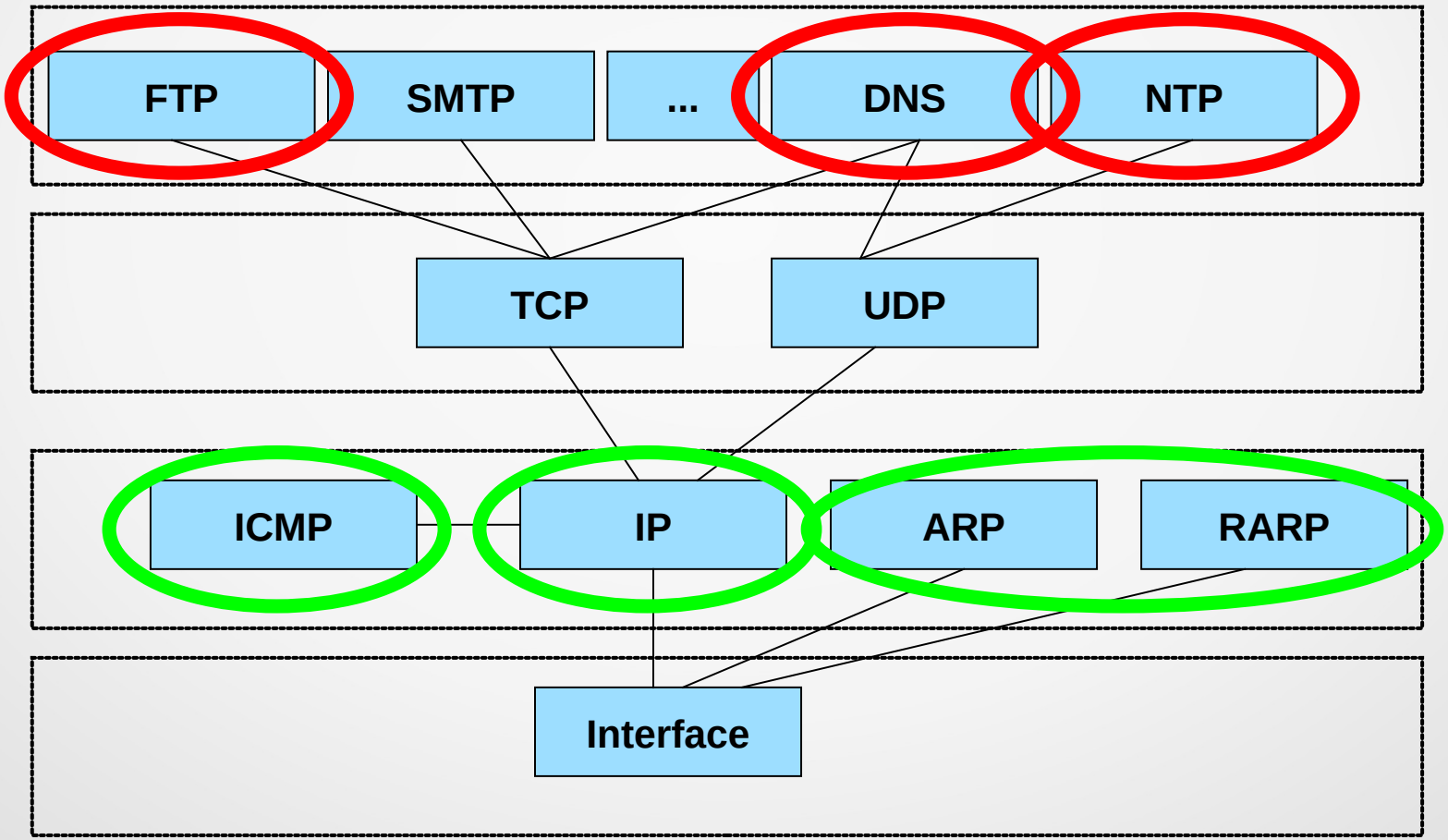


Liaison de données

Physique

Modèle Internet (TCP-IP)

- Principaux protocoles du modèle Internet



Exemple de capture de trame

- Outils :
  - Wireshark (graphique)
    - Décryptage des trames
    - Très agréable
  - Tcp-dump (mode texte)
    - Capture en hexadécimal
    - Plus austère mais pas nécessairement d'interface graphique (serveurs)

Exemple de capture de trame

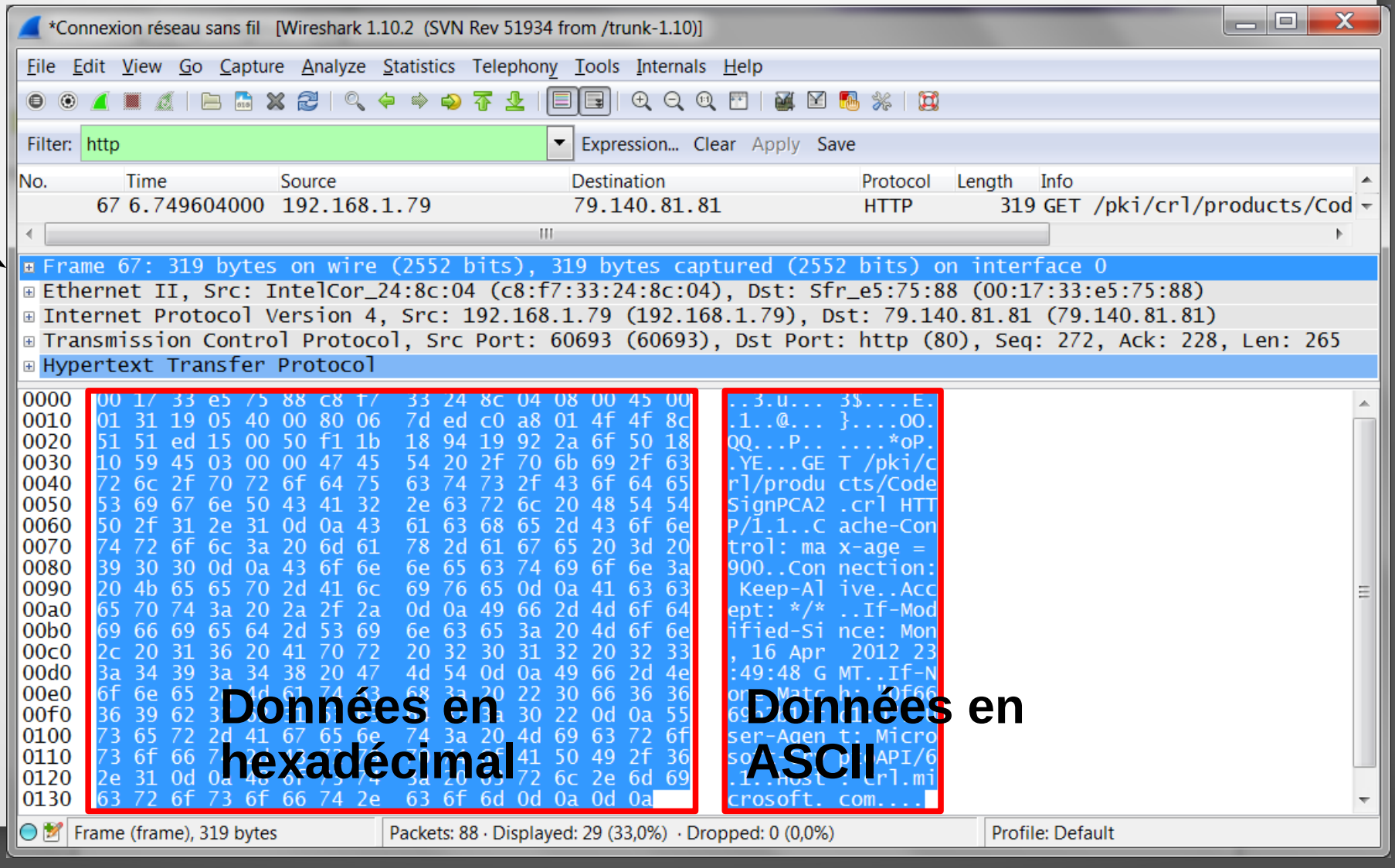
```
root@DebianFred_serv:/home/test# tcpdump
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 65535 bytes
15:16:20.980457 IP DebianFredserv.local.45040 > neufbox.domain: 43083+ A? sb-ssl.google.com. (35)
15:16:20.980814 IP DebianFredserv.local.60506 > neufbox.domain: 10059+ PTR? 1.1.168.192.in-addr.arpa. (42)
15:16:20.980879 IP DebianFredserv.local.45040 > neufbox.domain: 65204+ AAAA? sb-ssl.google.com. (35)
15:16:21.090770 IP neufbox.domain > DebianFredserv.local.60506: 10059* 1/0/0 PTR neufbox. (63)
15:16:21.091191 IP DebianFredserv.local.47340 > neufbox.domain: 20168+ PTR? 15.2.0.10.in-addr.arpa. (40)
15:16:21.095299 IP neufbox.domain > DebianFredserv.local.47340: 20168 NXDomain* 0/0/0 (40)
15:16:21.130804 IP neufbox.domain > DebianFredserv.local.45040: 43083 12/0/0 CNAME sb-ssl.l.google.com., A 173.194.45.69, A 173.194.45.70, A 173.194.45.71, A 173.194.45.66, A 173.194.45.73, A 173.194.45.67, A 173.194.45.78, A 173.194.45.65, A 173.194.45.72, A 173.194.45.64, A 173.194.45.68 (234)
15:16:21.134943 IP neufbox.domain > DebianFredserv.local.45040: 65204 2/0/0 CNAME sb-ssl.l.google.com., AAAA 2a00:1450:400c:c05::5b (86)
...
```

Exemple de capture de trame

```
root@DebianFred_serv:/home/test# tcpdump -vv -i eth0 -X port http -c 10
tcpdump: listening on eth0, link-type EN10MB (Ethernet), capture size 65535 bytes
15:29:53.327981 IP (tos 0x0, ttl 64, id 8680, offset 0, flags [DF], proto TCP (6), length
687)
    DebianFredserv.local.33580 > we-in-f94.1e100.net.www: Flags [P.], cksum 0xfed0
(incorrect -> 0xaf16), seq 3305361276:3305361923, ack 82594152, win 63900,
length 647
        0x0000:  4500 02af 21e8 4000 4006 1a32 0a00 020f  E...!.@.@..2....
        0x0010:  adc2 425e 832c 0050 c503 cf7c 04ec 4968  ..B^.,.P...|..Ih
        0x0020:  5018 f99c fed0 0000 4745 5420 2f20 4854  P.......GET./.HT
        0x0030:  5450 2f31 2e31 0d0a 486f 7374 3a20 7777  TP/1.1..Host:.ww
        0x0040:  772e 676f 6f67 6c65 2e66 720d 0a55 7365  w.google.fr..Use
        0x0050:  722d 4167 656e 743a 204d 6f7a 696c 6c61  r-Agent:.Mozilla
        0x0060:  2f35 2e30 2028 5831 313b 2055 3b20 4c69  /5.0.(X11;.U;.Li
        0x0070:  6e75 7820 7838 365f 3634 3b20 6672 3b20  nux.x86_64;.fr;.
        0x0080:  7276 3a31 2e39 2e31 2e31 3629 2047 6563  rv:1.9.1.16).Gec
        0x0090:  6b6f 2f32 3031 3231 3230 3720 4963 6577  ko/20121207.Icew
        0x00a0:  6561 7365 6c2f 332e 352e 3136 2028 6c69  easel/3.5.16.(li
        0x00b0:  6b65 2046 6972 6566 6f78 2f33 2e35 2e31  ke.Firefox/3.5.1
        0x00c0:  3629 0d0a 4163 6365 7074 3a20 7465 7874  6)..Accept:.text
        ...
```

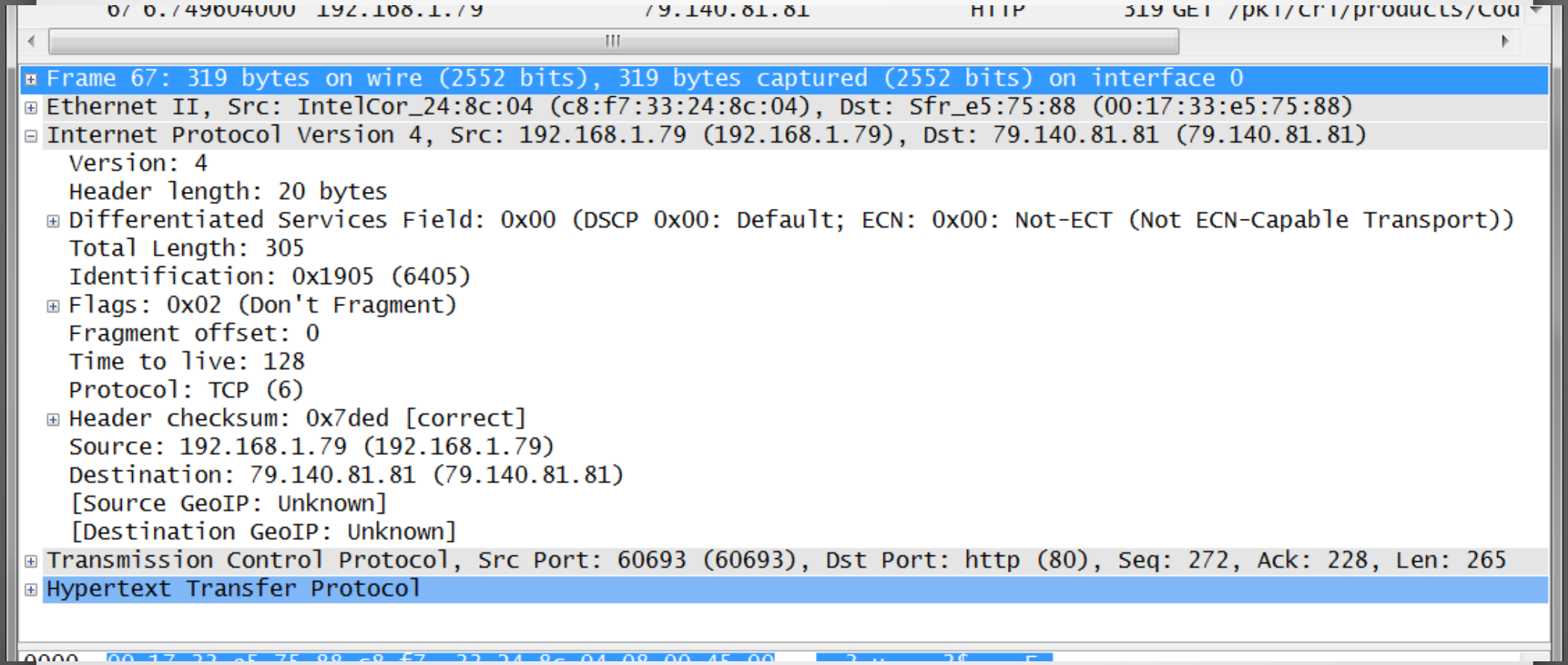
Exemple de capture de trame

- Avec wireshark



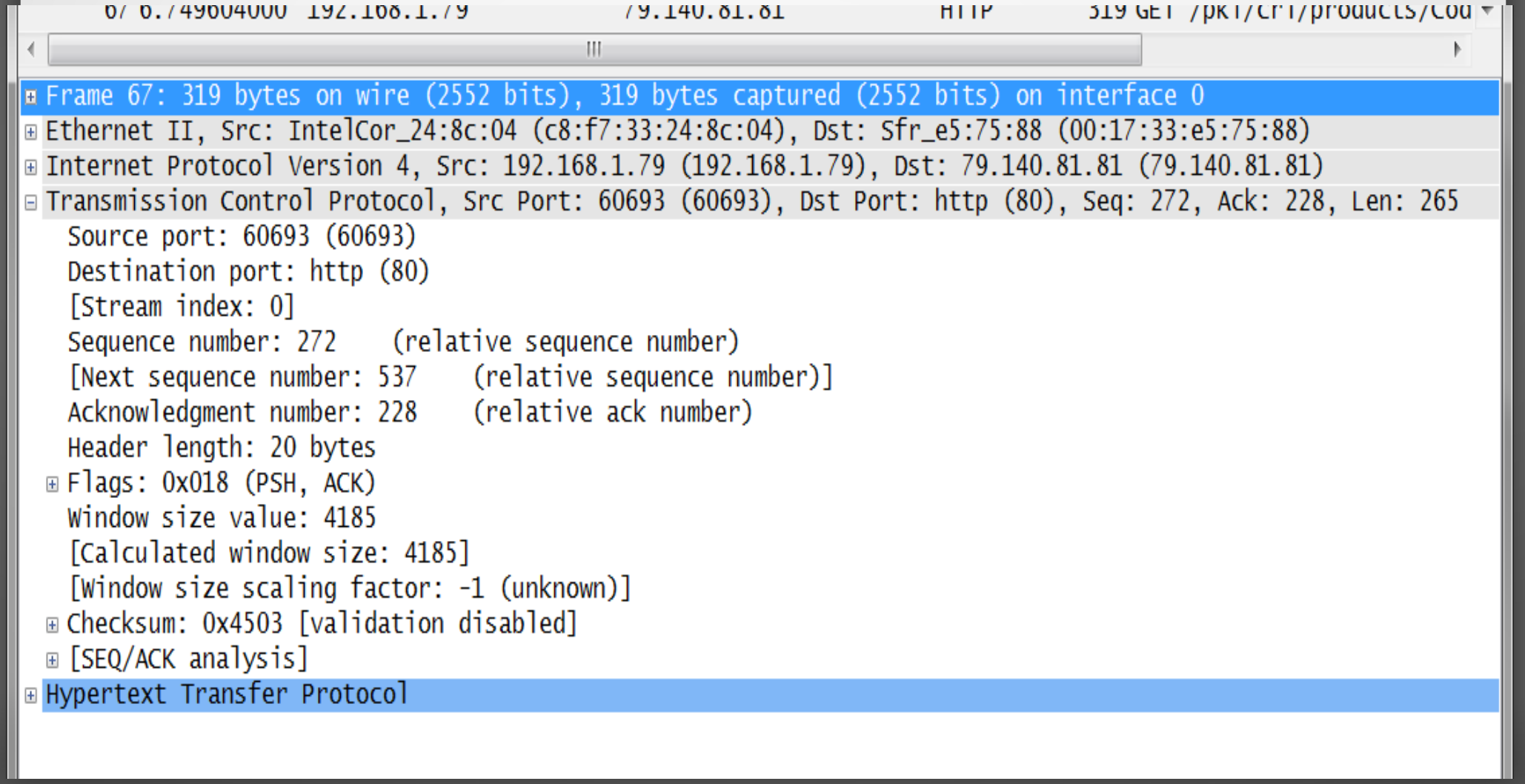
Exemple de capture de trame

- Détail des protocoles : analyse par couche
  - Détail du PCI de la couche 3 : IP



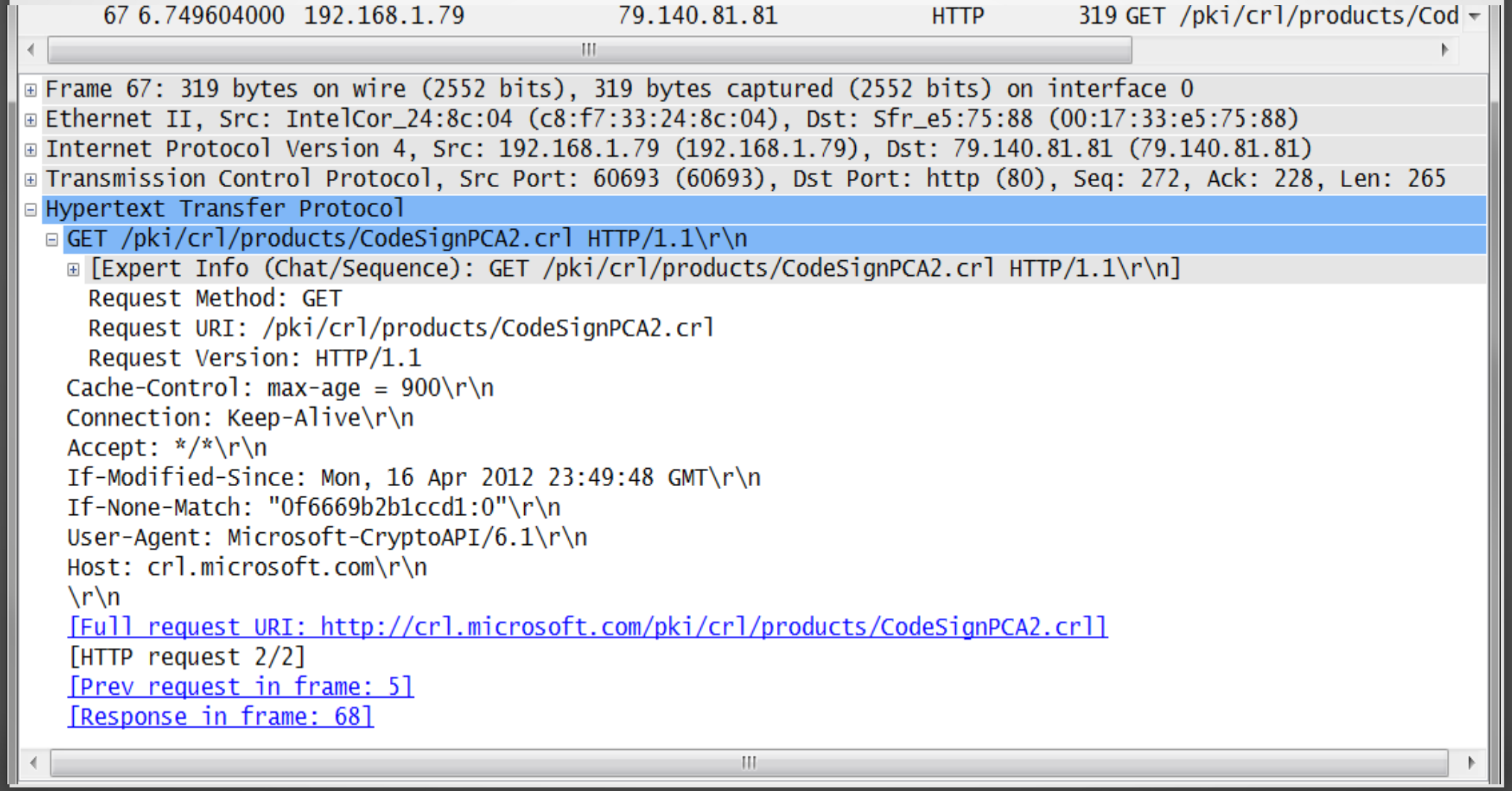
Exemple de capture de trame

- Détail du PCI de la couche 4 (TCP)



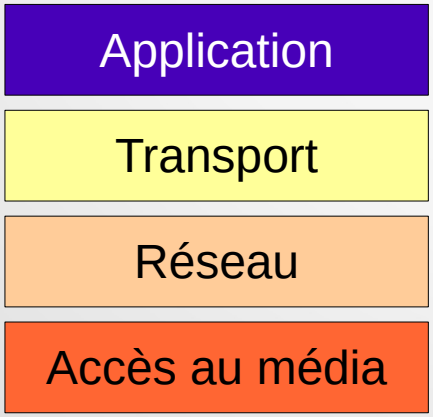
Exemple de capture de trame

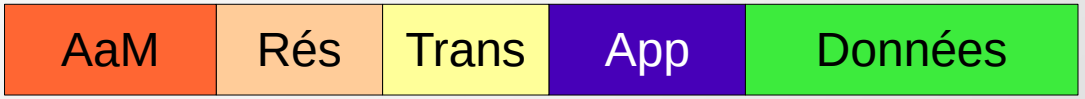
- Détail du PCI de la couche 5 : HTTP



Exemple de capture de trame

- On retrouve bien la notion d'encapsulation
- Les en-tête (PCI) des protocoles sont bien imbriqués les uns à la suite des autres
- On ne retrouve pas de d'information de la couche N dans la couche N+1 et vice-versa
- Attention à la représentation





D'autre modèles d'architecture ...

- Il en existe d'autre
  - Ethernet pour les couches 1 et 2 (développé plus tard dans l'année)
  - ATM pour les couches 1 et 2 (très utilisé dans le transfert de la voix)
- Il complètent le manque dans le modèle TCP-IP
  - Couches non traitées

Chapitre 2

- Analyse de protocoles
  - Protocoles de couche 3
    - ARP
    - IP
    - ICMP
  - Protocole de couche 4
    - TCP
    - UDP

Protocole vous avez dit protocole ?

- Que faut il savoir ?
  - Quelques principes de base de fonctionnement
  - Où trouver les infos utiles
  - Le reste viendra avec la pratique

Protocoles de couche 3

- Le réseau

Protocole 1

- ARP : Address Resolution Protocol

Protocole ARP

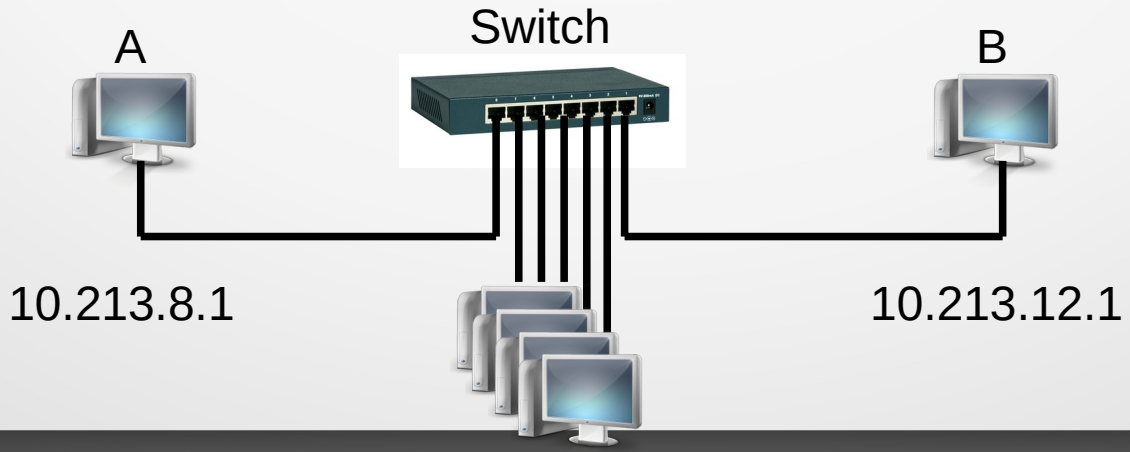
- Son rôle ?
  - Déterminer l'adresse physique de la machine si on connaît son adresse logique
- A quoi ça sert ?
  - Scénario : la machine A veut envoyer un paquet IP à la machine B
- Plusieurs cas
  - Réseau point à point
  - Réseau multi points

Cas réseau point à point

- Liaison PPP / HDLC / Frame Relay
- Liaison série... pas de suspens
- Le paquet IP est encapsulé dans une trame PPP et envoyé à l'autre extrémité du réseau

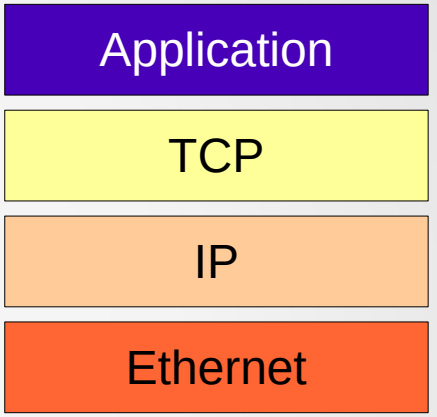
Cas réseau multipoints

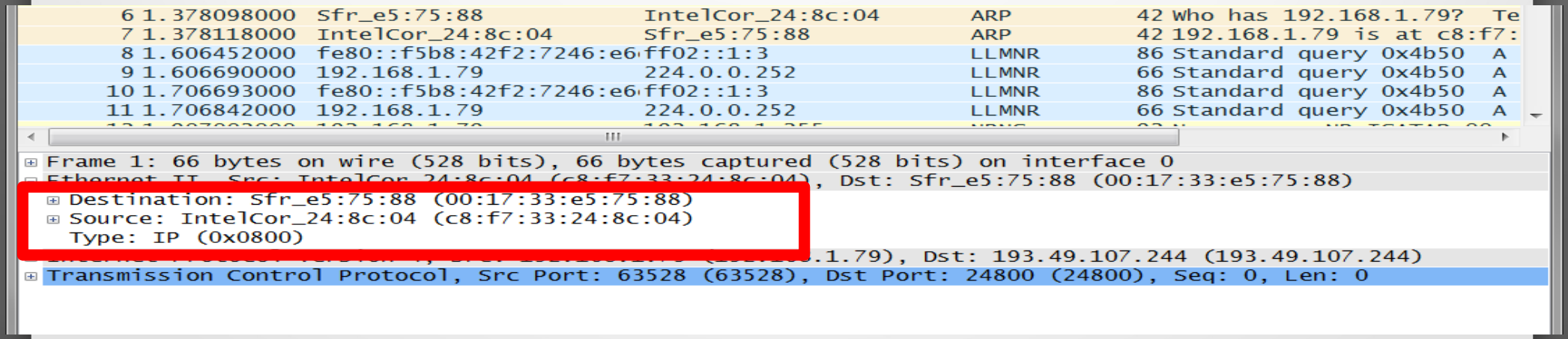
- Cas Ethernet
  - On doit encapsuler le paquet dans une trame Ethernet
  - Plusieurs machines sont reliées sur un même brin du réseau : il faut **identifier** la bonne → **adressage**
  - Il faut déterminer l'adresse MAC du destinataire
  - Rôle du protocole ARP



Construction du message

- Rappel sur les couches
- Leurs besoins
  - IP : les adresses IP
  - Ethernet : les adresses MAC





Construction du message

- Si on ne sait pas remplir les PCI …
  - On ne peut pas envoyer le message
- Il faut **résoudre** l'adresse MAC de la destination
  - Trouver une @ MAC qui correspond à une @ IP

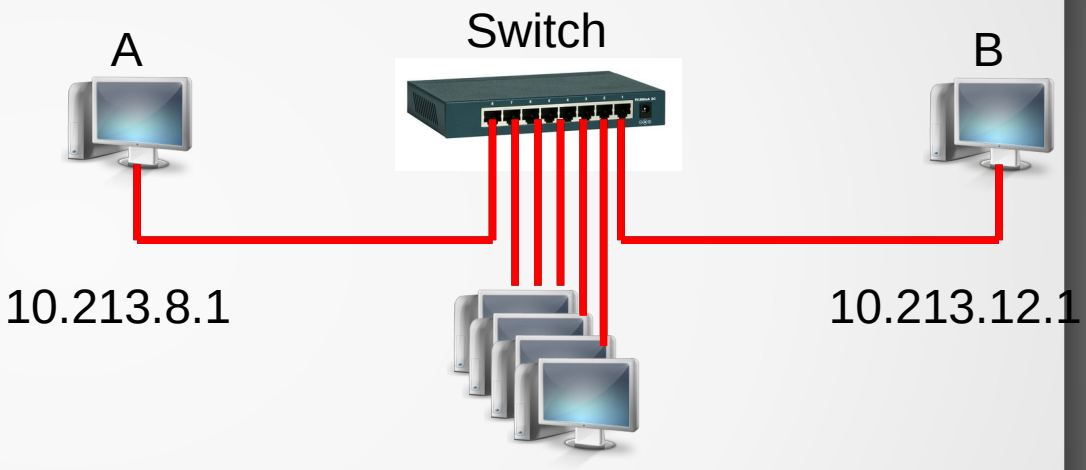
Remarque :
Seulement dans IPv4 en IPv6 ARP disparaît
(remplacé par NDP neighbor Discovery Protocol)

Généralités sur ARP

- Protocole de niveau 3
  - Se situe à l'interface entre les couches 2 et 3
- Définit par la RFC 826
- Comment récupérer l'@ MAC d'une machine si on ne peut pas émettre de trame sans @ MAC ?
- Utilisation de la diffusion :
  - Adresse MAC diffusion FF:FF:FF:FF:FF:FF
  - Cible toutes les machines sur un même brin du réseau

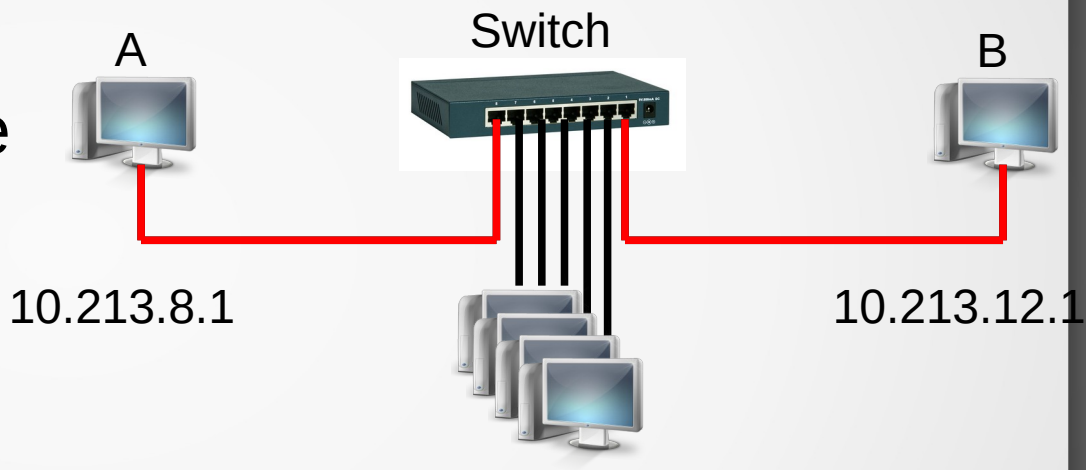
Fonctionnement d'ARP

- A envoie une trame Ethernet en diffusion
  - MAC source : MAC_A
  - MAC destination : **FF:FF:FF:FF:FF:FF**
  - Contenant une requête ARP
  - IP source : IP_A
  - IP destination : IP_B



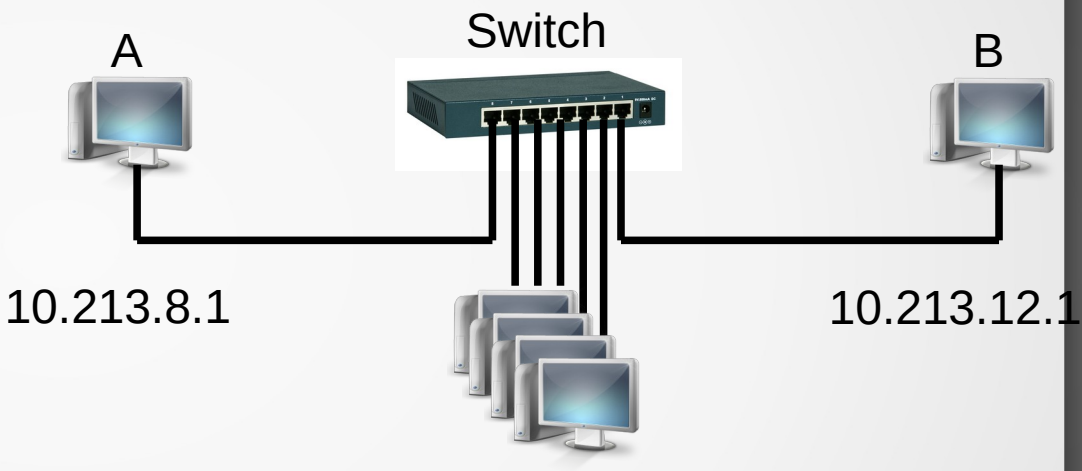
Fonctionnement d'ARP

- Optionnel : mise à jour du cache ARP en B (MAC de A)
- La machine B renvoie une trame Ethernet
  - MAC source : MAC_B
  - MAC dest : MAC_A
  - Contenant une réponse ARP
  - IP source : IP_B
  - IP dest : IP_A



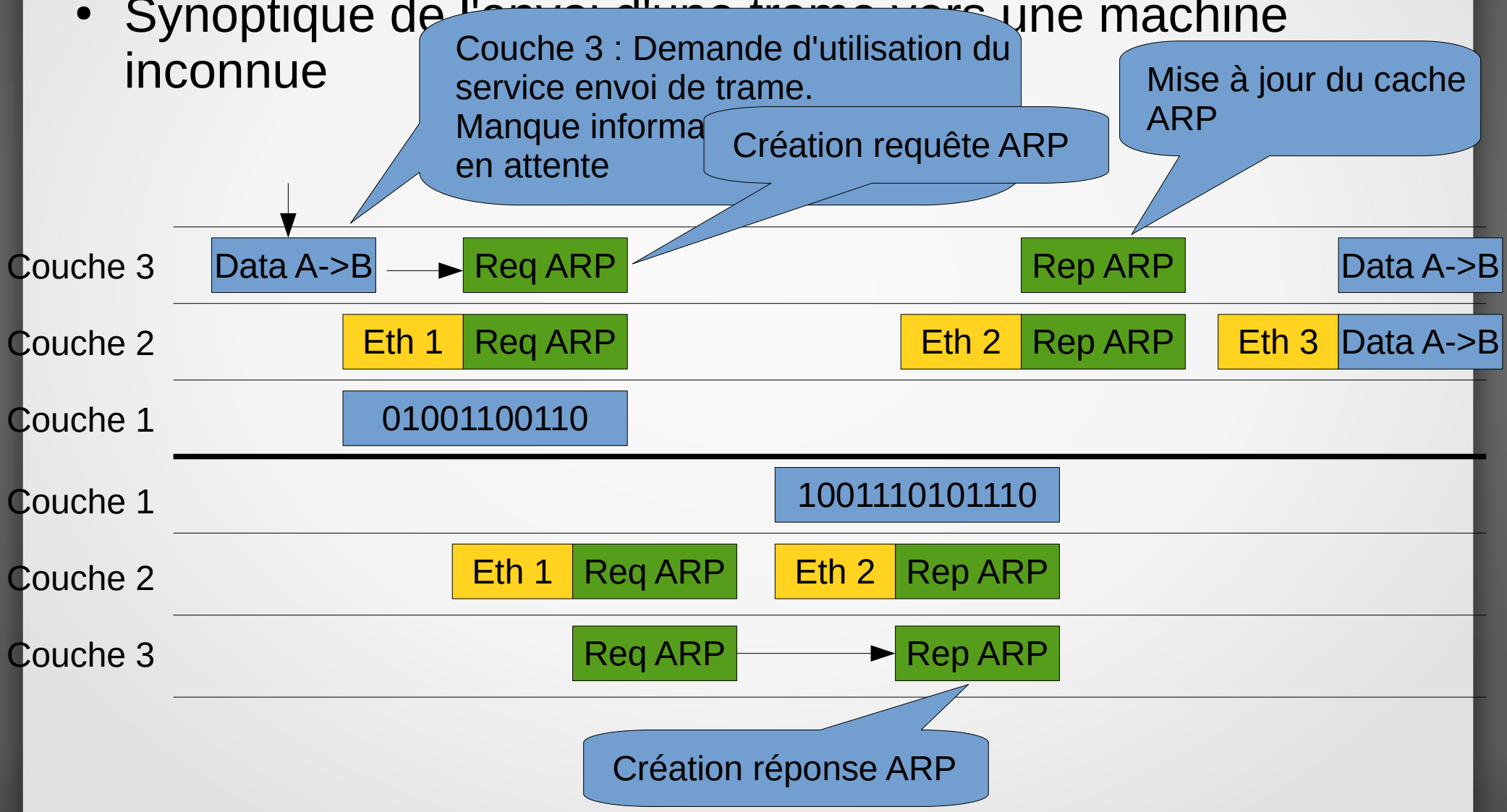
Fonctionnement d'ARP

- Mise à jour du cache ARP en A (MAC de B)
- Les machines A et B sont prêtes pour discuter



Généralités sur ARP

- Synoptique de l'envoi d'une trame vers une machine inconnue



Analyse protocole ARP

- Capture de trame ARP
  - Requête ARP

Filter: Expression... Clear Apply Save

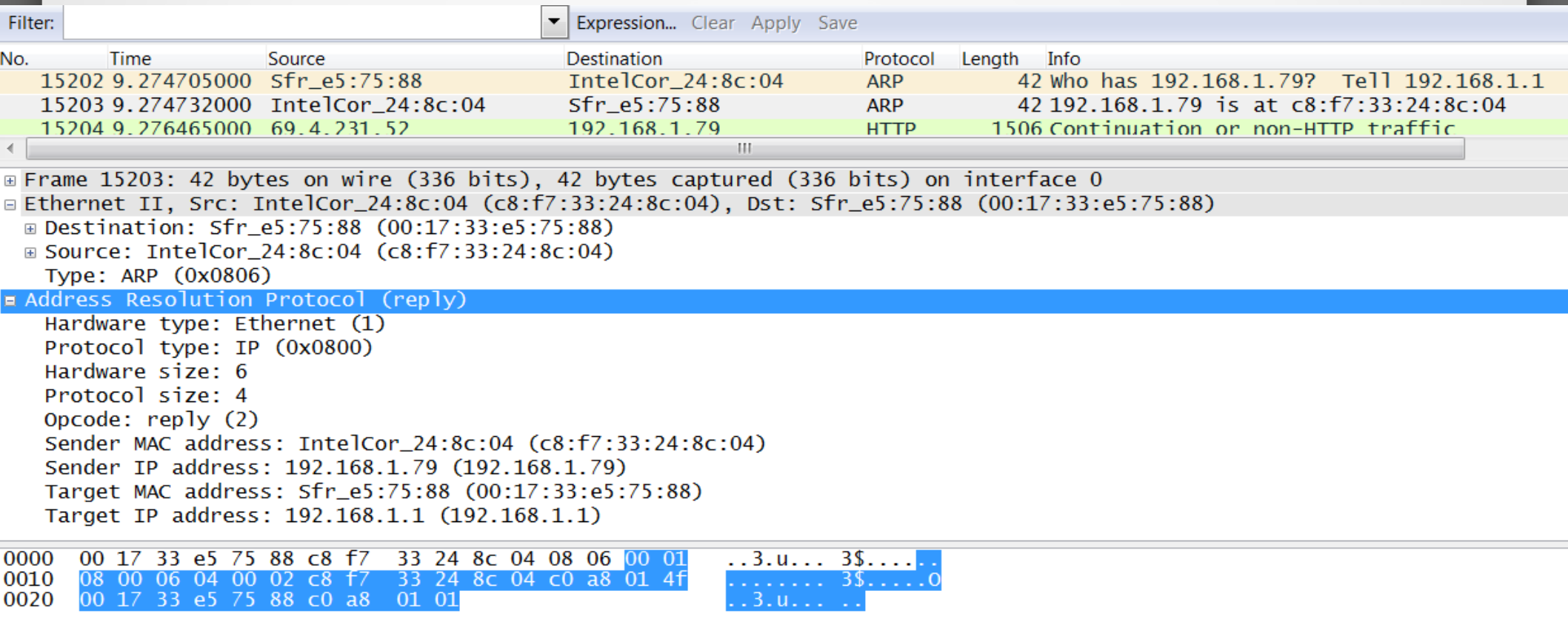
No.	Time	Source	Destination	Protocol	Length	Info
15202	9.274705000	Sfr_e5:75:88	IntelCor_24:8c:04	ARP	42	Who has 192.168.1.79? Tell 192.168.1.1
15203	9.274732000	IntelCor_24:8c:04	Sfr_e5:75:88	ARP	42	192.168.1.79 is at c8:f7:33:24:8c:04
15204	9.276465000	69.4.231.52	192.168.1.79	HTTP	1506	Continuation or non-HTTP traffic

```
Frame 15202: 42 bytes on wire (336 bits), 42 bytes captured (336 bits) on interface 0
Ethernet II, Src: Sfr_e5:75:88 (00:17:33:e5:75:88), Dst: IntelCor_24:8c:04 (c8:f7:33:24:8c:04)
  Destination: IntelCor_24:8c:04 (c8:f7:33:24:8c:04)
  Source: Sfr_e5:75:88 (00:17:33:e5:75:88)
  Type: ARP (0x0806)
Address Resolution Protocol (request)
  Hardware type: Ethernet (1)
  Protocol type: IP (0x0800)
  Hardware size: 6
  Protocol size: 4
  Opcode: request (1)
  Sender MAC address: Sfr_e5:75:88 (00:17:33:e5:75:88)
  Sender IP address: 192.168.1.1 (192.168.1.1)
  Target MAC address: 00:00:00_00:00:00 (00:00:00:00:00:00)
  Target IP address: 192.168.1.79 (192.168.1.79)
```

```
0000  c8 f7 33 24 8c 04 00 17  33 e5 75 88 08 06 00 01   ..3$.... 3.u.....
0010  08 00 06 04 00 01 00 17  33 e5 75 88 c0 a8 01 01   ........ 3.u.....
0020  00 00 00 00 00 00 c0 a8  01 4f                     ........ .O
```

Analyse protocole ARP

- Capture de trame
  - Réponse ARP



Analyse protocole ARP

- Encapsulé dans une trame (Ethernet)

- Format du PCI ARP
  - Cas général
  - Taille inconnue car dépend des adresses

	Bits 0-7	Bits 8-15	Bits 16-31
0	Hardware type	Hardware type	Protocol type
32	Hardware address length	Protocol address length	Operation
64	Sender	Hardware	Address
?	Sender	Protocol	Address
?	Target	Hardware	Address
?	Target	Protocol	Address

Analyse protocole ARP (PCI)

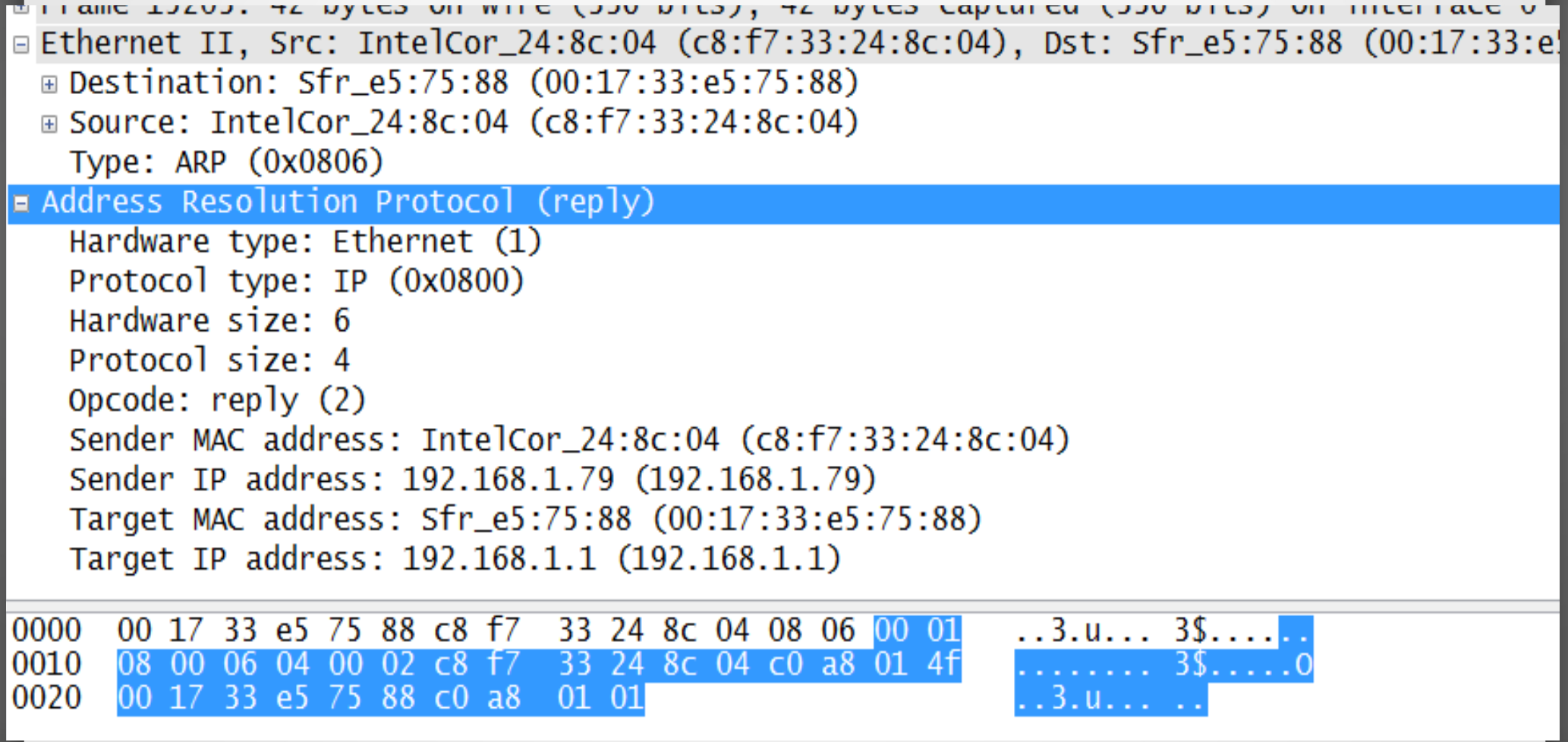
- Hardware type (2 octets)
  - Caractérise le protocole de la couche liaison
  - Chez nous : 0001 = Ethernet (il en existe d'autres)
- Protocole type (2 octets)
  - Caractérise le protocole de la couche réseau
  - Chez nous : 0800 = IP
- Hardware Address Length (1 octet)
  - Taille des adresses de niveau 2 : 6 octets (Ethernet)
- Protocol Address Length (1 octet)
  - Taille des adresses de niveau 3 : 4 octets (IP)

Analyse protocole ARP (PCI)

- Operation (2 octets)
  - Type de message : 01 requête, 02 réponse
- Sender Hardware Address
  - Adresse de niveau 2 de l'émetteur (adresse MAC)
- Sender Protocol Address
  - Adresse de niveau 3 de l'émetteur (adresse IP)
- Target Hardware Address
  - Adresse de niveau 2 de l'émetteur (adresse MAC)
- Target Protocol Address
  - Adresse de niveau 3 de l'émetteur (adresse IP)

Analyse protocole ARP (PCI)

- Exemple :



Analyse protocole ARP

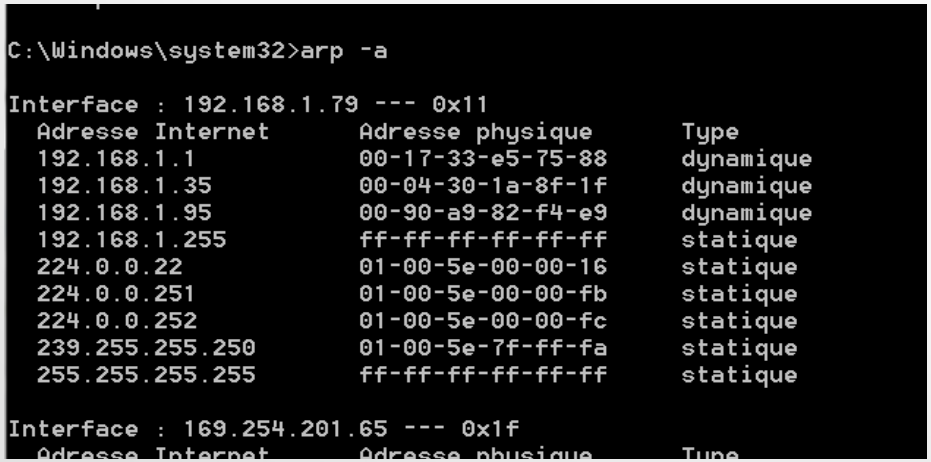
- Cache ARP ?
  - Fichier contenant les paires @MAC / @IP
- Comment le visualise-t-on ?
  - arp -a sous windows
  - ip neigh sous linux (ou cat /proc/net/arp)
- Différentes entrées dans le cache
  - Statique
  - Dynamique

Analyse protocole ARP

- Sous linux

```
test@DebianFred_serv:~$ ip neigh
10.0.2.2 dev eth0 lladdr 52:54:00:12:35:02 STALE
test@DebianFred_serv:~$ cat /proc/net/arp
IP address       HW type     Flags       HW address            Mask     Device
10.0.2.2         0x1         0x2         52:54:00:12:35:02     *        eth0
test@DebianFred_serv:~$
```

- Sous windows



Cache ARP

- Pourquoi un cache ?
  - Pour éviter les résolutions à chaque paquet
- Pourquoi des entrées statiques ?
  - Évite les requêtes en diffusion pour des machines souvent usitées
- Pourquoi entrées dynamiques ?
  - Ne pas garder toutes les entrées inutilisées
  - Permettre la mise à jour des adresses MAC (changement carte)
- Décrit dans la RFC 826

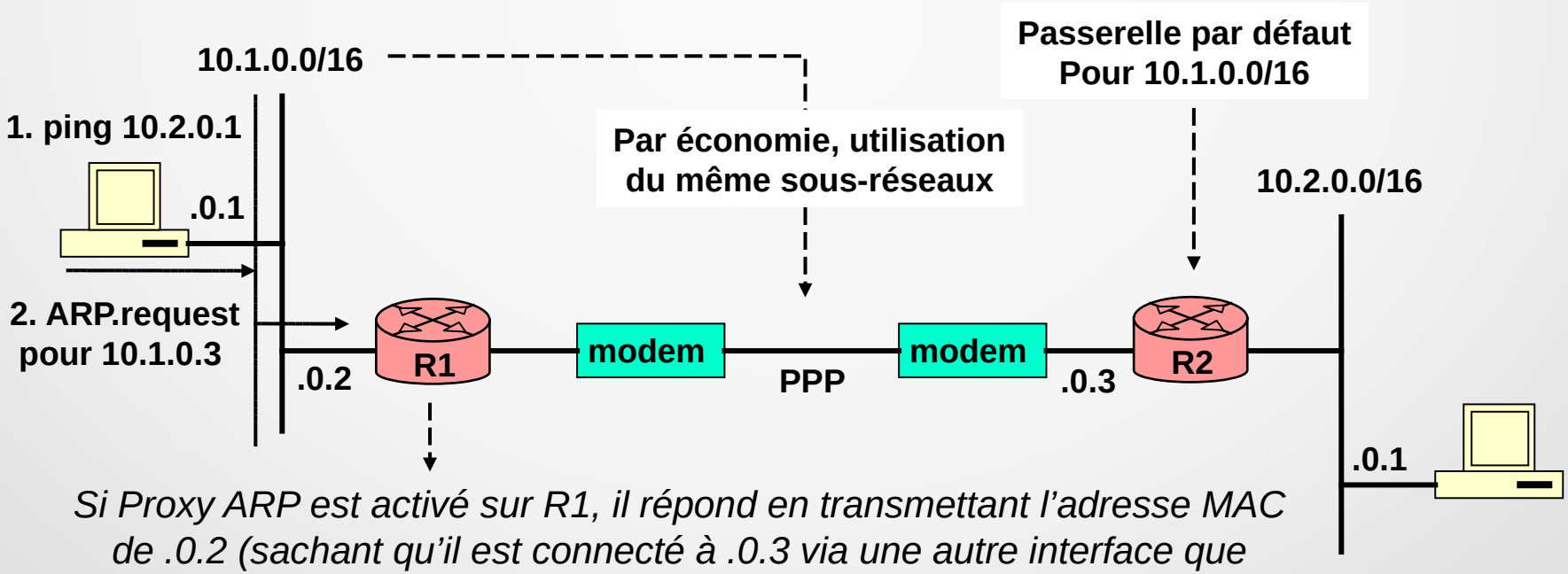
Cache ARP

- Timeout
  - Après 3 tentatives ratées on arrête
- Les états possibles d'une entrée

cache state	**meaning**	**action if used**
permanent	never expires; never verified	reset use counter
noarp	normal expiration; never verified	reset use counter
reachable	normal expiration	reset use counter
stale	still usable; needs verification	reset use counter; change state to delay
delay	schedule ARP request; needs verification	reset use counter
probe	sending ARP request	reset use counter
incomplete	first ARP request sent	send ARP request
failed	no response received	send ARP request

Proxy ARP

- Portée d'une requête ARP ?
  - Réseau local : jusqu'au routeur
- Que se passe t-il si le réseau est divisé ?



Si Proxy ARP est activé sur R1, il répond en transmettant l'adresse MAC de .0.2 (sachant qu'il est connecté à .0.3 via une autre interface que l'interface par laquelle est arrivée la requête ARP)

ARP gratuit

- Question envoyée sans attendre de réponse
- Permet d'annoncer qu'une adresse IP est associée avec une adresse mac
- Utile pour savoir si une adresse IP est déjà utilisée (dans ce cas réponse)
  - DHCP
- Permet la mise à jour des caches ARP

Limites du protocole ARP

- Il existe des failles …
  - Usurpation d'adresse MAC (MAC Spoofing)
    - On corrompt le cache ARP d'un switch (message forgé)
    - Obsolète et aléatoire (nécessite dénis de service)
  - Usurpation d'identité ARP (ARP spoofing)
    - Répondre plus vite que le destinataire au requêtes ARP
    - Obsolète et aléatoire (nécessite dénis de service)
  - Cache poisoning
    - Outil arp-sk
    - Envoi de requête ARP unicast avec des valeurs choisies entraîne mise à jour du cache faussé

Les parades

- Les systèmes de détection d'intrusions
  - IDS : surveille les modification ARP (ArpWatch)
- Le cache ARP Statique
  - Attention chez microsoft statique signifie qui n'expire pas … mais est modifiable
- Filtrage au niveau ARP
  - Filtrage MAC / IP
- Sources :
  - http://sid.rstack.org/arp-sk/article/arp.html
  - http://tools.ietf.org/html/rfc826
  - http://linux-ip.net/html/ether-arp.html

Protocole 2

- Protocole IP : Internet Protocol

Protocole IP

- Protocole de couche 3
- Définit par la RFC 791
  - Modifié par la RFC 1349
- Son rôle ?
  - Acheminer "au mieux" les paquets sur le réseau
  - Best effort : il peut perdre des paquets mais pas volontairement

Liste des RFC : http://fr.wikipedia.org/wiki/Liste_de_RFC

Protocole IP

- Protocole non fiable car :
  - Possible corruption des données
  - Ordre des paquets pas garanti
  - Duplication possible de paquets
- Seule fiabilité
  - Le PCI (en-tête) du protocole est vérifié
  - Si erreur alors destruction du paquet + envoi message d'erreur
- Objectif simplifier la tâche des routeurs pour rapidité

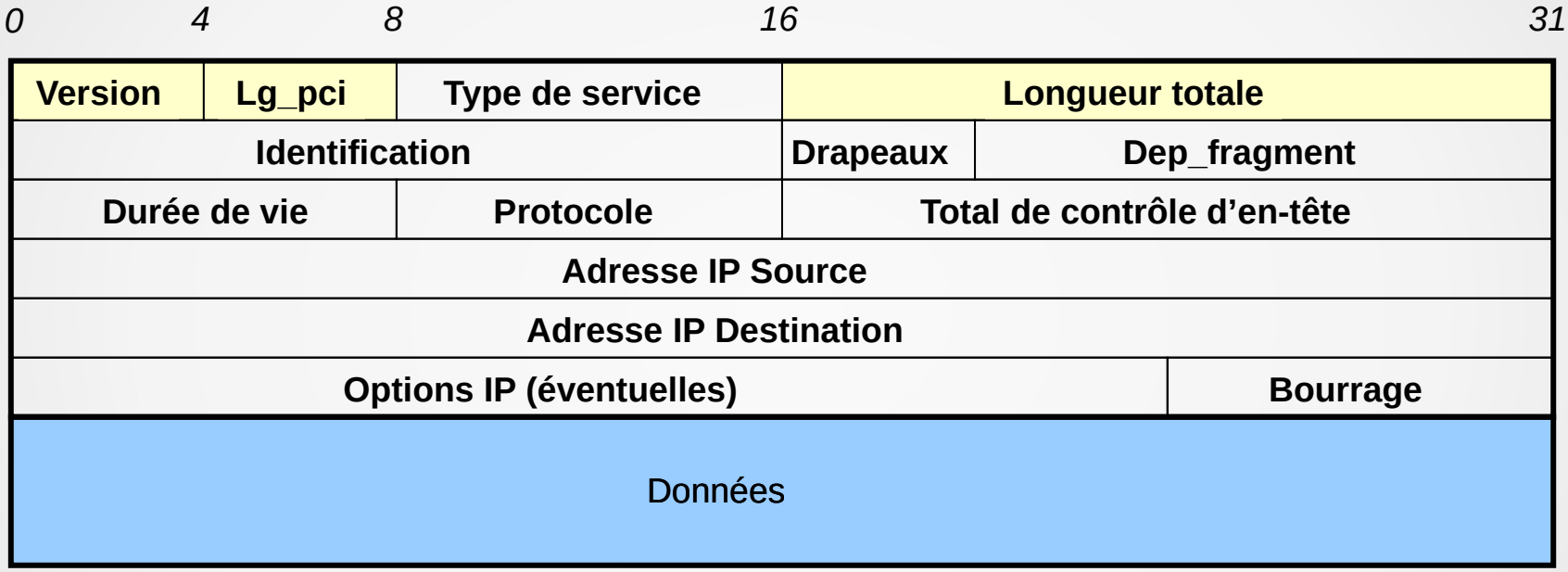
Protocole IP

- Gère l'adressage des machines
  - Adresses IP (déjà vu en M1101)
- Assure une couche d'homogénéité au niveau du réseau
- Tend s'imposer comme protocole standard dans

  l'acheminement des messages : routage ip remplace la commutation de trames (même pour la voix)
- On va vers un monde tout IP

Analyse du protocole IP

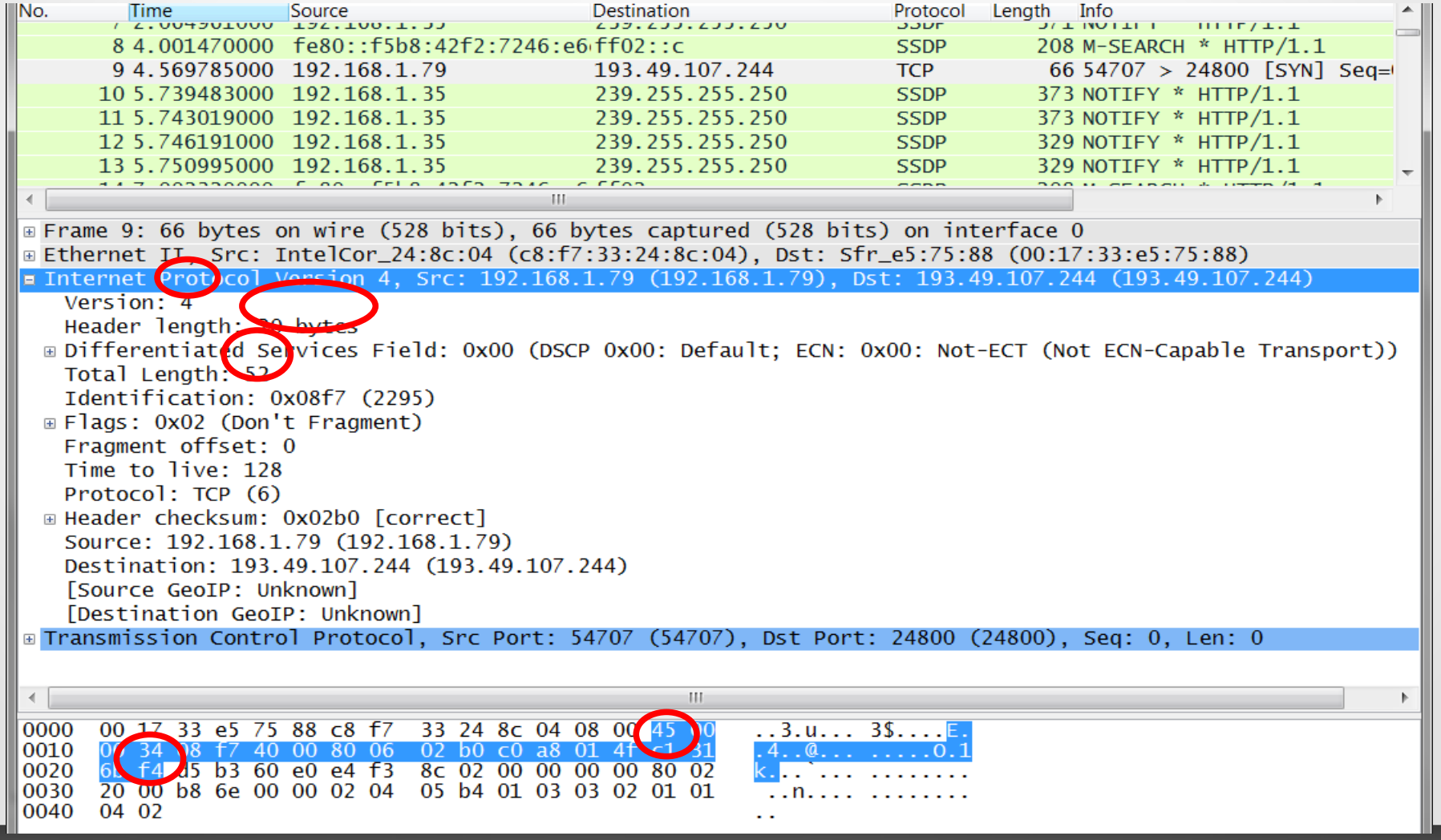
- Analyse du PCI IP



- Version : version du protocole (4 généralement)
- Lg_pci : longueur de l'en tête en mot de 32 bits
- Longueur totale : longueur totale du paquet en octets

Analyse du protocole IP

- Un exemple de paquet IP



Analyse du protocole IP

- Limitation de la couche Liaison
  - Impossible d'encapsuler de très longs paquets

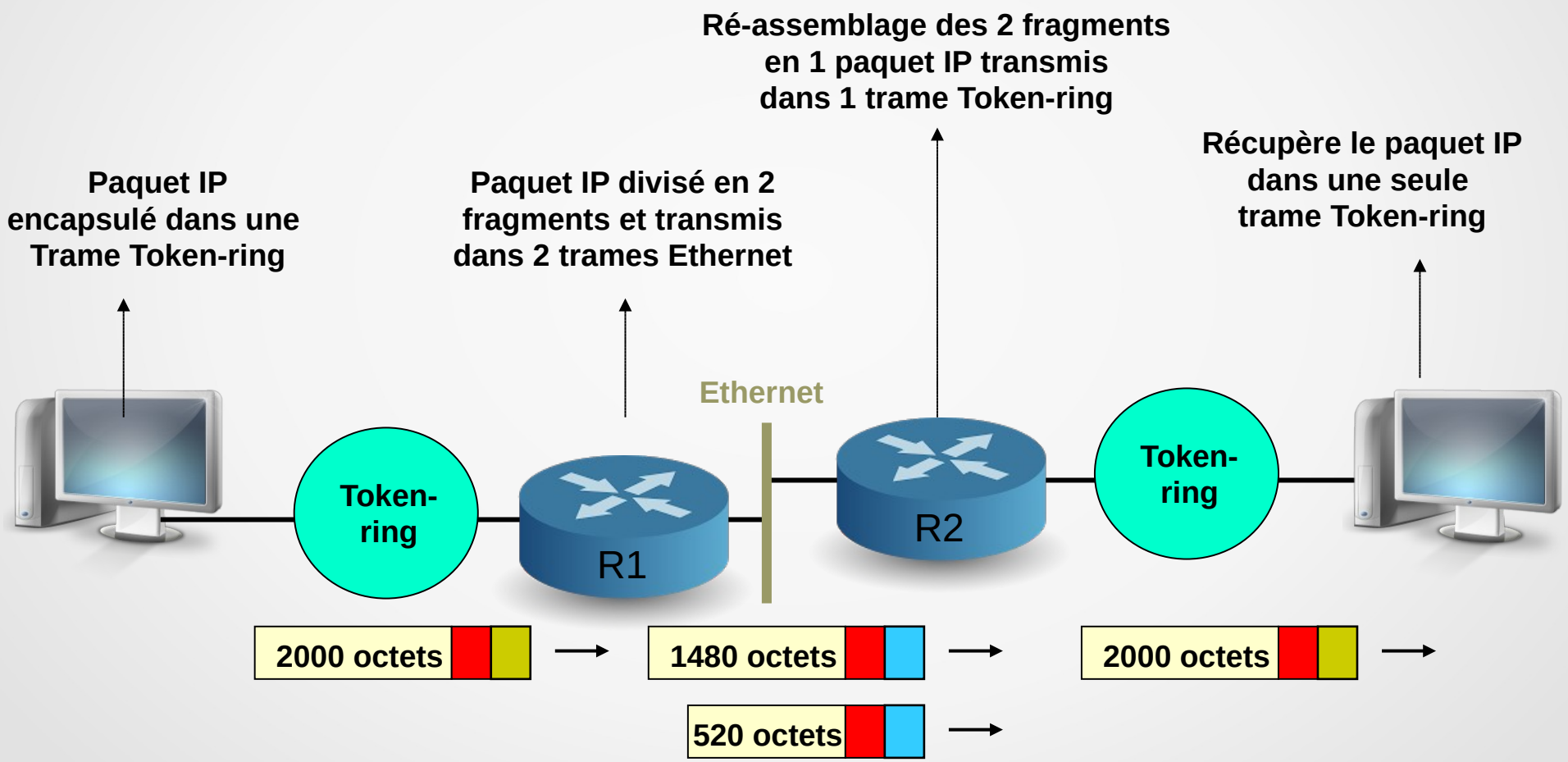
Exemple : 1500 octets Ethernet ou PPP, 53 octets trames ATM, 4500 octets trames token-ring.

- Solution proposée par IP :
  - Découper les longs paquets sur chaque liaison de données
  - **Fragmentation**
  - **Ré-assemblage**

Analyse du protocole IP

- Exemple de fragmentation : cas pratique
  - Problème posé :
  - Transmission d'un paquet de 2000 octets de données (en-tête non compris)
  - Doit traverser des réseaux "Token-ring" et "Ethernet"
  - Contraintes :
    - Au plus 4500 octets dans les trames « Token-ring »
    - Au plus 1500 octets dans les trames « Ethernet »

Analyse du protocole IP



Analyse du protocole IP

- Comment fait on pour retrouver les morceaux du paquets et les "re-coller" ensemble ?
  - Arrivée dans le mauvais ordre possible
  - Arrivée non consécutive
- Tout est compris dans l'en-tête IP
- Fragmentation / Ré-assemblage : plusieurs solutions ...

Analyse du protocole IP

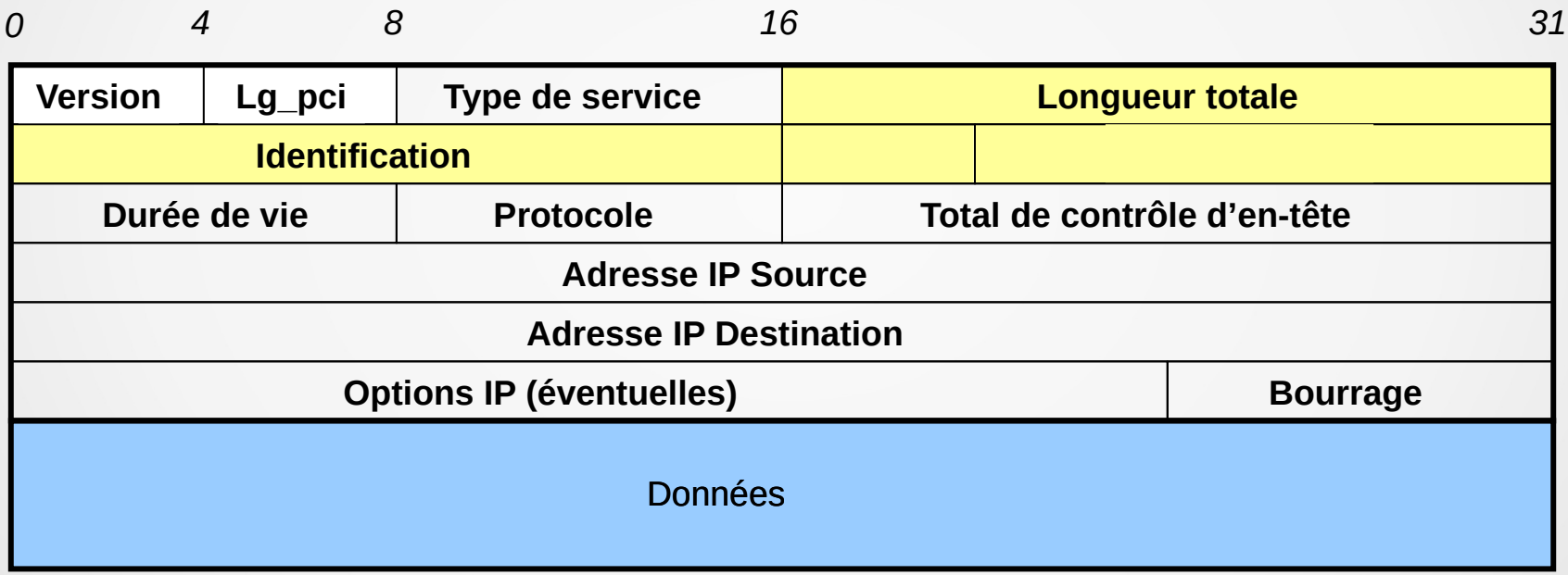
- Tout équipement (station ou routeur) peut fragmenter ou ré-assembler
  - ***Un paquet peut être fragmenté et ré-assemblé plusieurs fois***
  - **Avantages** : utilisation au mieux de toutes les liaisons
  - **Inconvénients** : coûteux en terme de CPU (routeurs) et des délais de bout-en-bout
- Seule les stations peuvent ré-assembler
  - ***Pas de ré-assemblage au niveau des routeurs, seulement de la fragmentation***
  - **Avantages** : moins coûteux en CPU au niveau des routeurs et réduction des délais de bout-en-bout
  - **Inconvénients** : utilisation non optimale de toutes les liaisons

Analyse du protocole IP

- Seule les stations peuvent réassembler et fragmenter
  - ***Pas de réassemblage et fragmentation au niveau des routeurs Choix fait au niveau de IPv6***
  - **Avantages** : faible coût CPU aux routeurs + réduction des délais de bout-en-bout
  - **Inconvénients** : utilisation non optimale de toutes les liaisons
- Si un routeur doit transmettre un paquet trop gros et qu'il ne peut fragmenter ?
  - Impossible … donc ...
  - Il en informe la source en utilisant **ICMP**
  - La source fragmente

Analyse du protocole IP

- Champs utilisés :



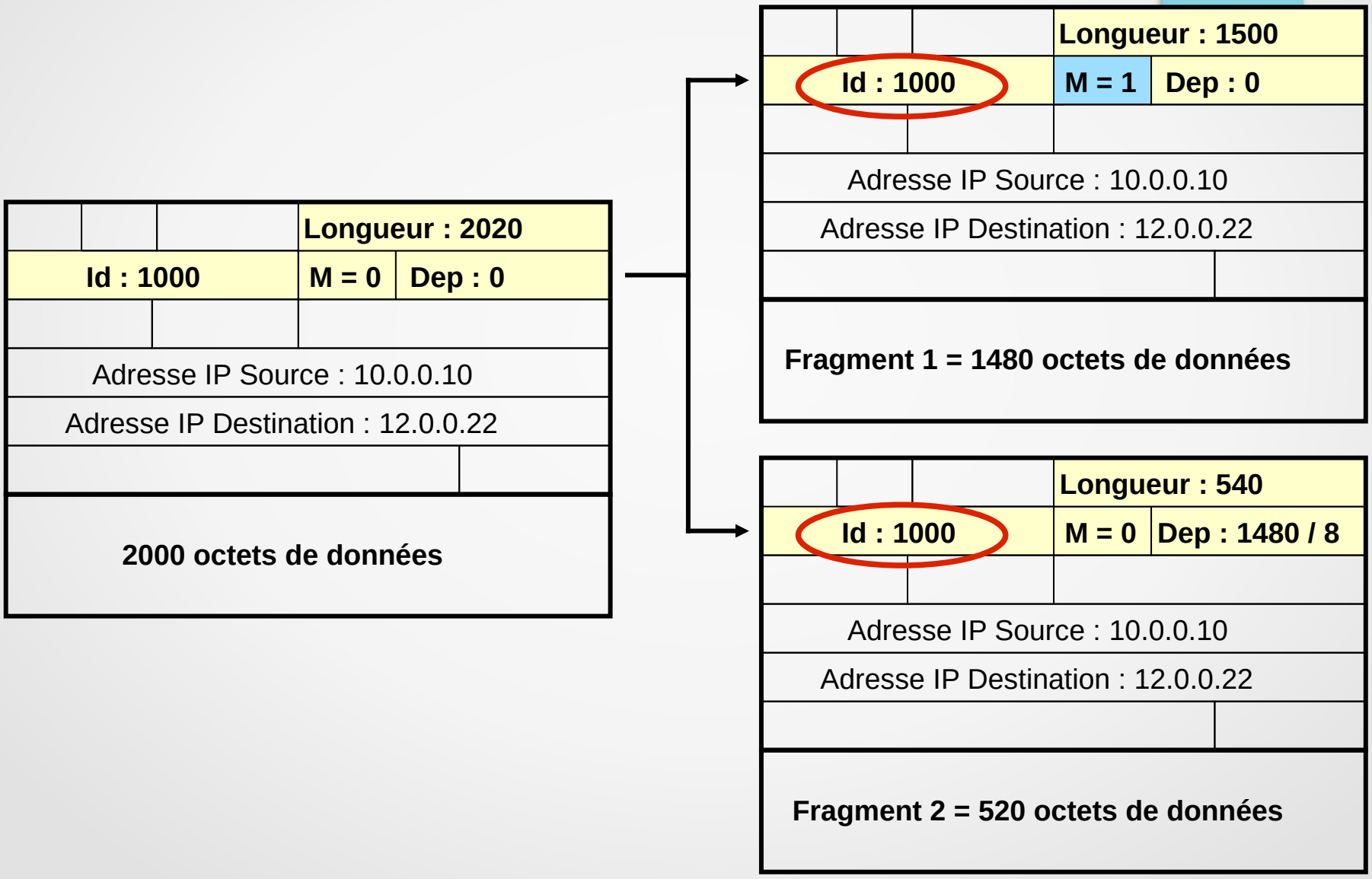
Analyse du protocole IP

- **Longueur totale** : indication de la taille du fragment
- **Identification** : identifiant commun à tous les fragments d’un même paquet
- **Dep_fragment** : position dans le datagramme du premier octet du fragment (en multiple de 8 octets)
- **Drapeaux** (3 bits)

Analyse du protocole IP

- **Bit M « à suivre »** : indique si un fragment est le dernier fragment d'un datagramme
- **Bit de « non-fragmentation »** : permet d'interdire la fragmentation d'un datagramme (détruit si impossible)

Analyse du protocole IP

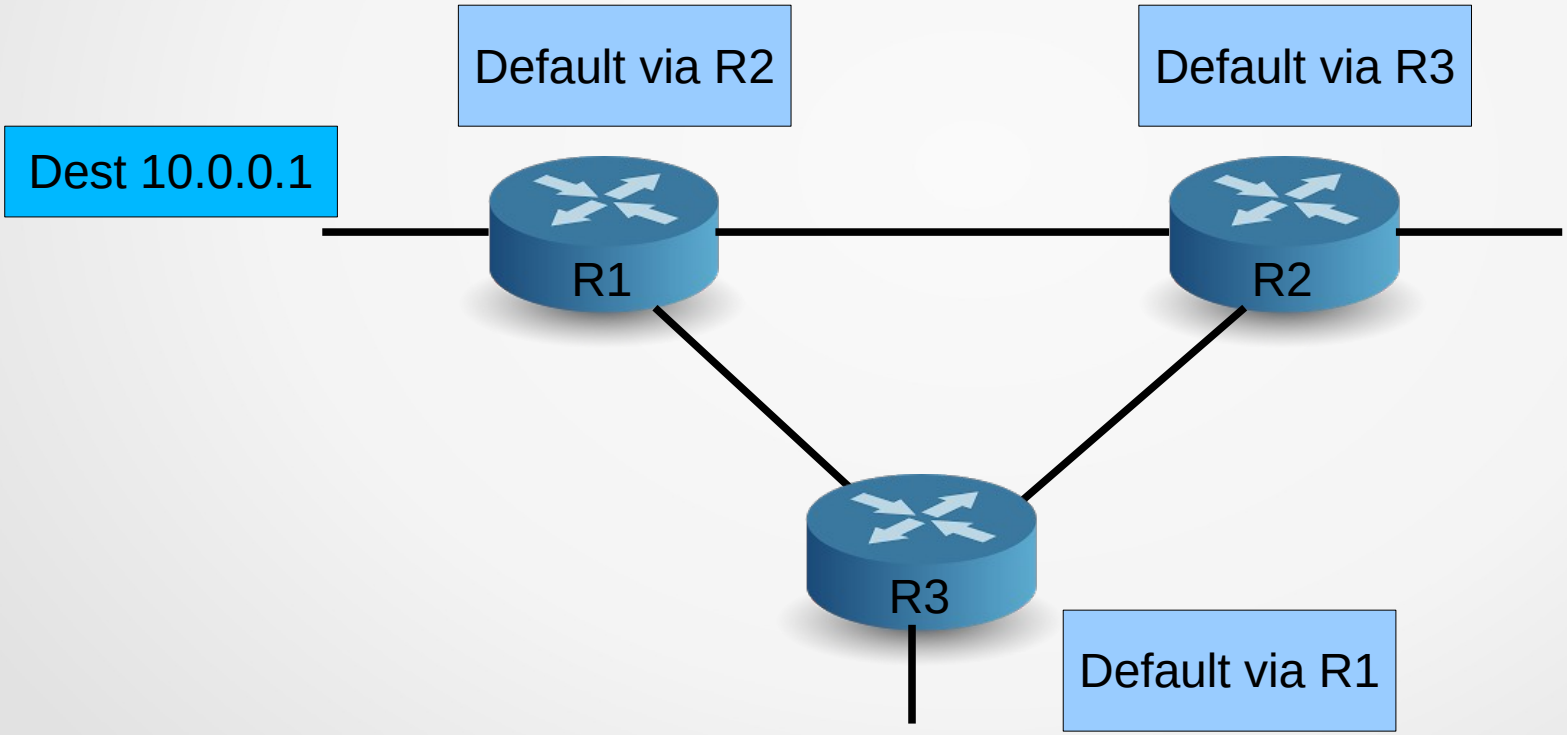


Analyse du protocole IP

- Quand a-t-on reçu tous les fragments ?
  - Le dernier fragment contient un bit M = 0
- **Attention** : ordre des fragments pas garanti
  - Tous les fragments ont été reçus ?
  - grâce aux champs « longueur totale » et « dep_fragments »
  - Souvent, dernier fragment transmis en premier → on peut réserver l'espace mémoire pour le ré-assemblage
- Que fait-on si des fragments sont perdus ou erronés ?
  - Le paquet complet ne peut pas être reconstitué
  - Une perte d'un fragment induit donc une perte d'un paquet

Analyse du protocole IP

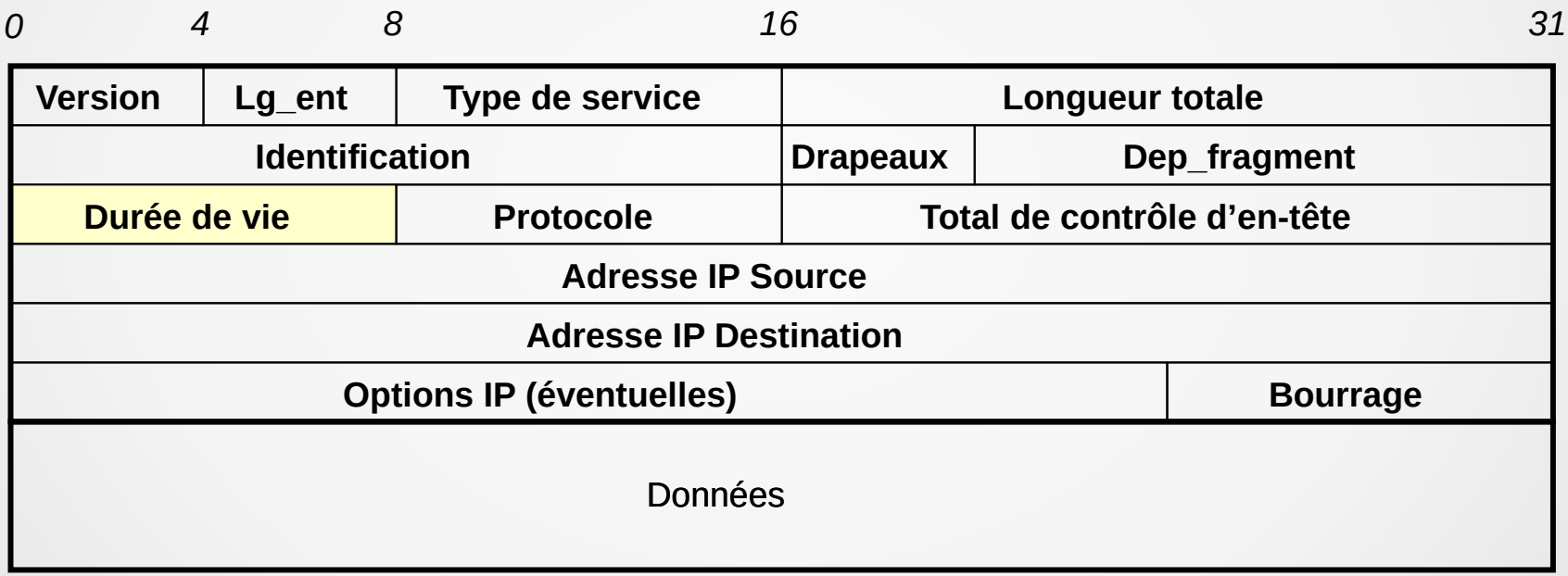
- Autre problème :
  - Mauvaise configuration des routeurs + bouclages



  - Occupation de la bande passante inutilement

Analyse du protocole IP

- TTL : Time to Live (durée de vie)



Analyse du protocole IP

- Rôle et fonctionnement du TTL
  - Valeur décrémentée à chaque traversée d'1 routeur
  - Si TTL = 0 : message détruit
  - Et envoi d'un message **ICMP** (time to live excedeed)

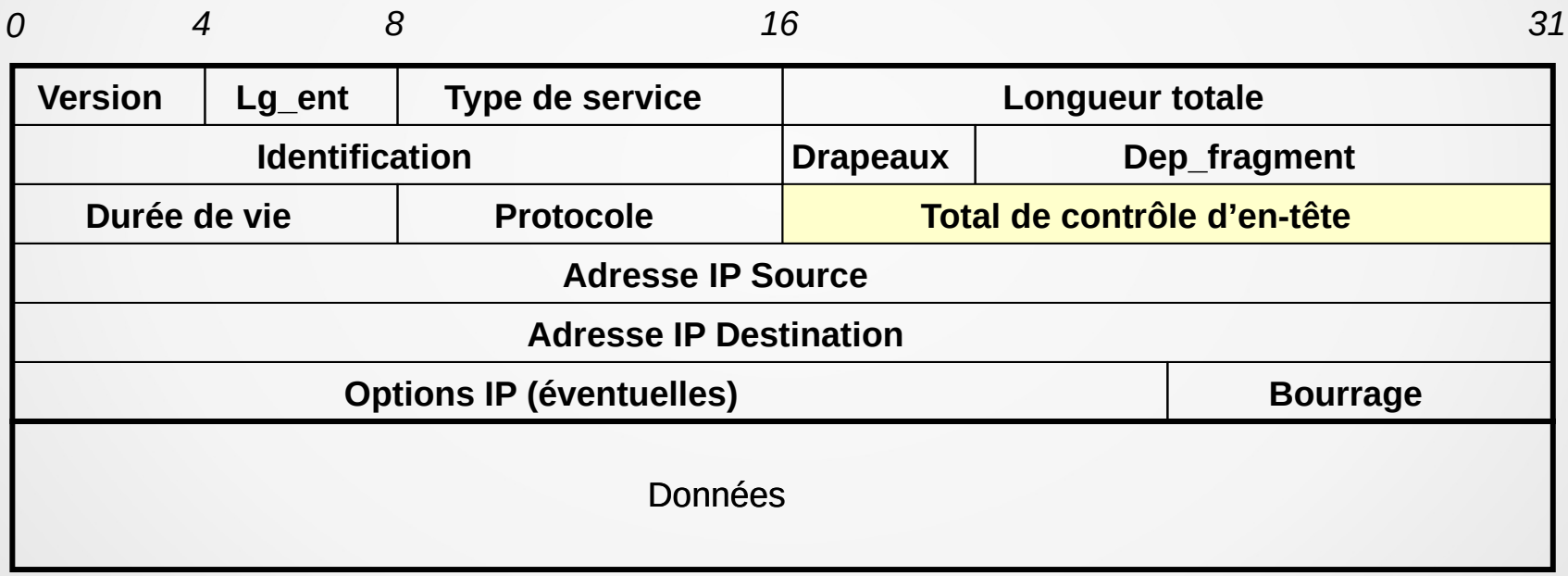
Remarque :
Valeur initiale du TTL dépend de l'OS
- Windows = 128
- Linux = 64
- http://noahdavids.org/self_published/TTL_values.html

Analyse du protocole IP

- IP = Service sans connexion et non fiable
  - Pas de détection des erreurs sur le contenu
  - Détection des erreurs sur l'en-tête
- Permet éviter d'acheminer des paquets vers des destinations erronées
- Utilisation d'une somme de contrôle (checksum) au niveau de l'en-tête (somme bit à bit des mots de 16 bits)
- Problème posé :
  - Le checksum doit être recalculé chaque fois que l'en-tête
    est modifié
  - Par exemple, lors des modifications du champ TTL

Analyse du protocole IP

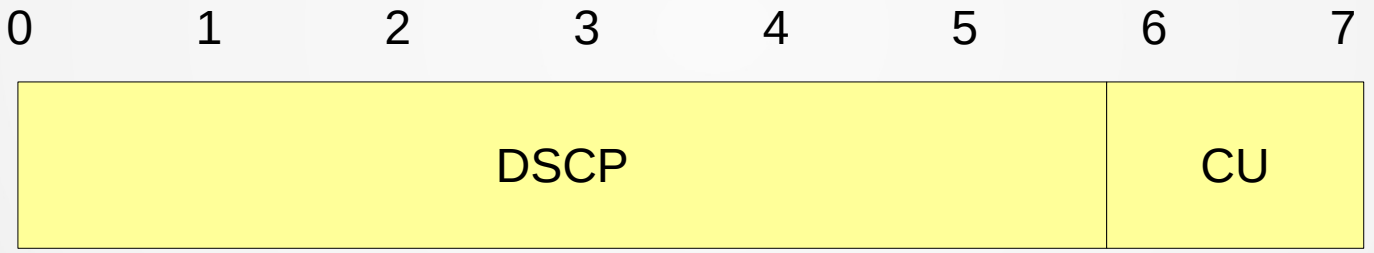
- Checksum
  - Basé sur la division polynomiale



Analyse du protocole IP

- Autres champs :
- Differentiated Services Field
  - RFC 2474

DSCP : Differntiated Service Code Point
CU : Currently Unused



  - Gère la qualité de service proposé
    - Network control (30), Guaranteed (28), Control load (18), autre trafic 0
  - Tous les équipements ne le gèrent pas ...

Analyse du protocole IP

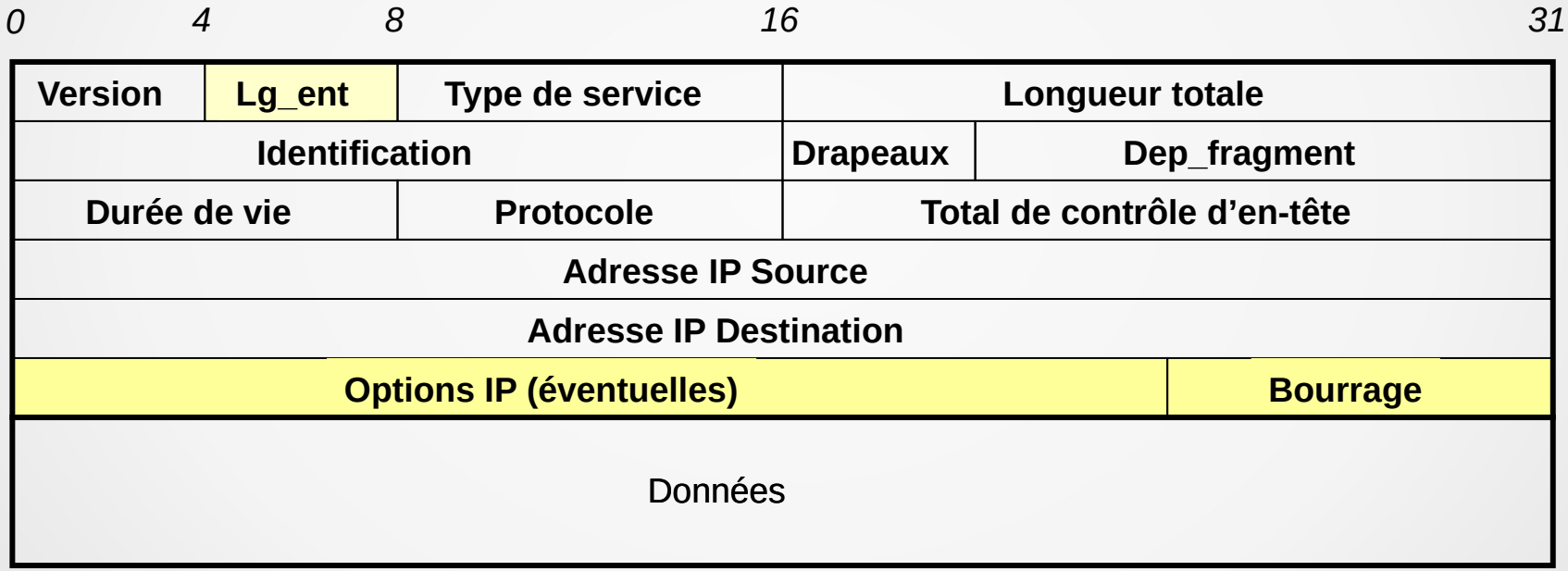
- Protocole
  - Permet d'identifier l'utilisateur du service IP : Qui a placé les données dans le paquet constitué
  - Si le paquet est arrivé à destination : On transmet au bon utilisateur (processus) les données encapsulées dans le paquet
  - Valeurs principales du champ « Protocole »
    - 1 = ICMP
    - 2 = IGMP
    - 4 = IP (IP dans IP)
    - 6 = TCP
    - 17 = UDP, etc.

Internet Assigned Numbers Authority

Voir www.iana.org/assignments/protocol-numbers

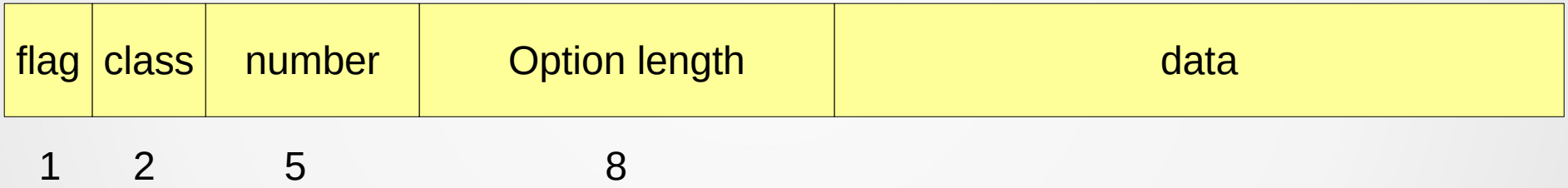
Analyse du protocole IP

- Champs optionnels
  - Détectés à l'aide de la longueur de l'en-tête



Analyse du protocole IP

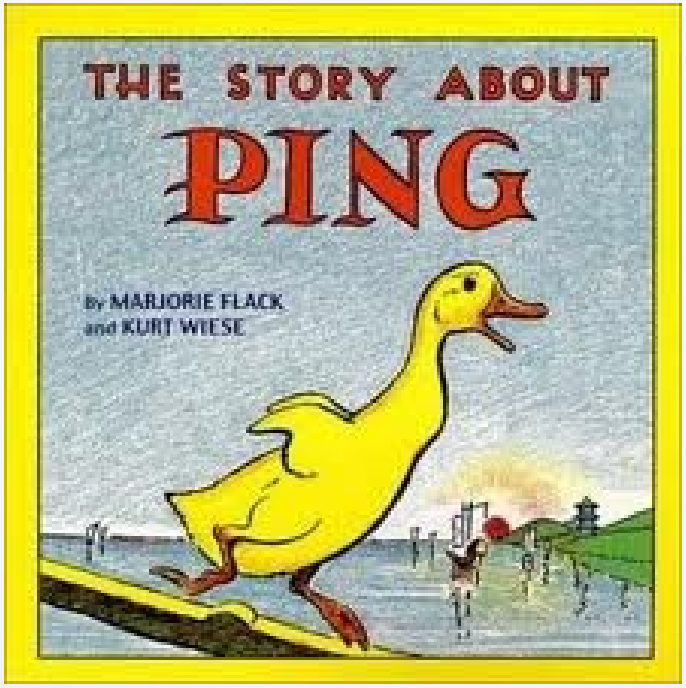
- Champs optionnels
  - 1 octet type d'option
  - 1 octet longueur de l'option
  - Des données



  - Flag 0 option pas recopiée dans tous les paquets (1 sinon)
  - Option class : 0 pour le contrôle (classique) et 2 pour débuggage et mesure
  - Option number : 0 – Cas spécial : fin de liste d'option, 1 – Pas d'opération, 2 – Sécurité (RFC 791), etc

Protocole 3

- Protocole ICMP : Internet Control Message Protocol



Protocole ICMP

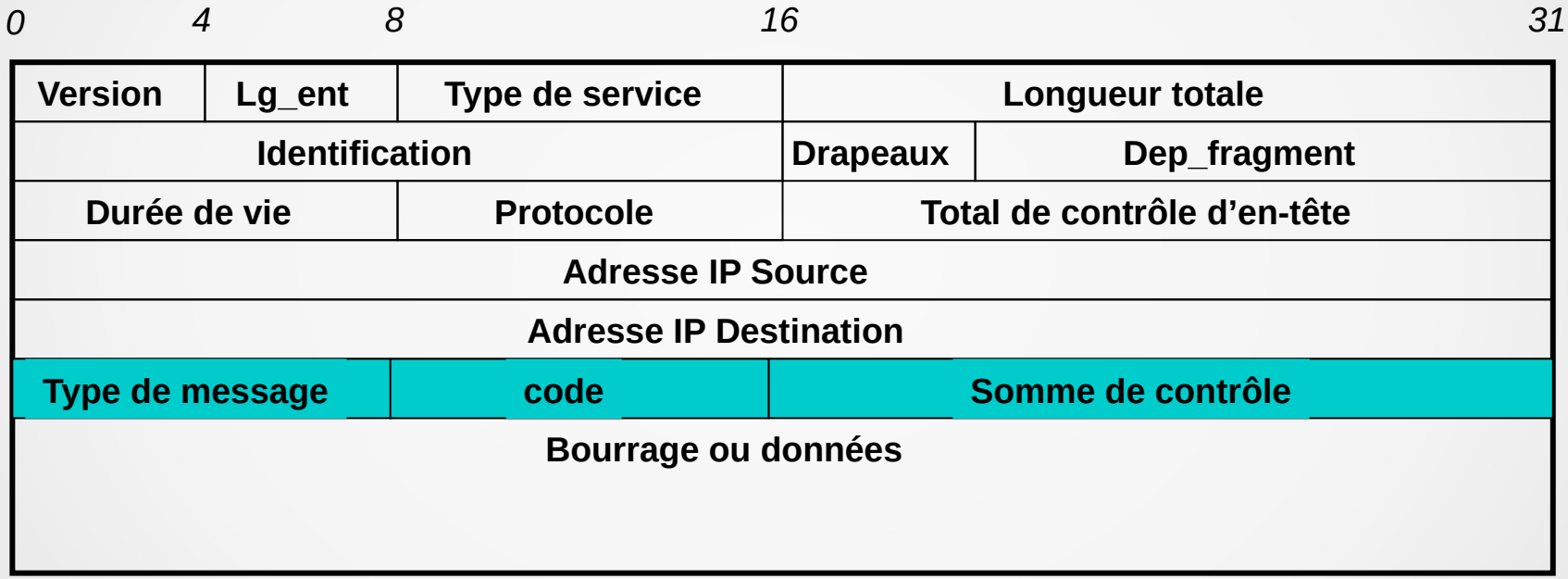
- Protocole de niveau 3
- RFC 792
  - Ajout RFC 1122 (ajoute quelques messages)
  - Ajout RFC 1812
- Rôle : permet d'avertir qu'un problème a eu lieu sur le réseau.
  - Le protocole IP ne le permet pas
  - Sur-couche d'IP

Protocol ICMP

- Encapsulé dans un entête IP
  - Champ protocol = 1 (ICMP)
- Ne peut pas générer de nouveau message d'erreur
  - Erreur sur un message d'erreur → boucles dans le réseau.

Protocole ICMP

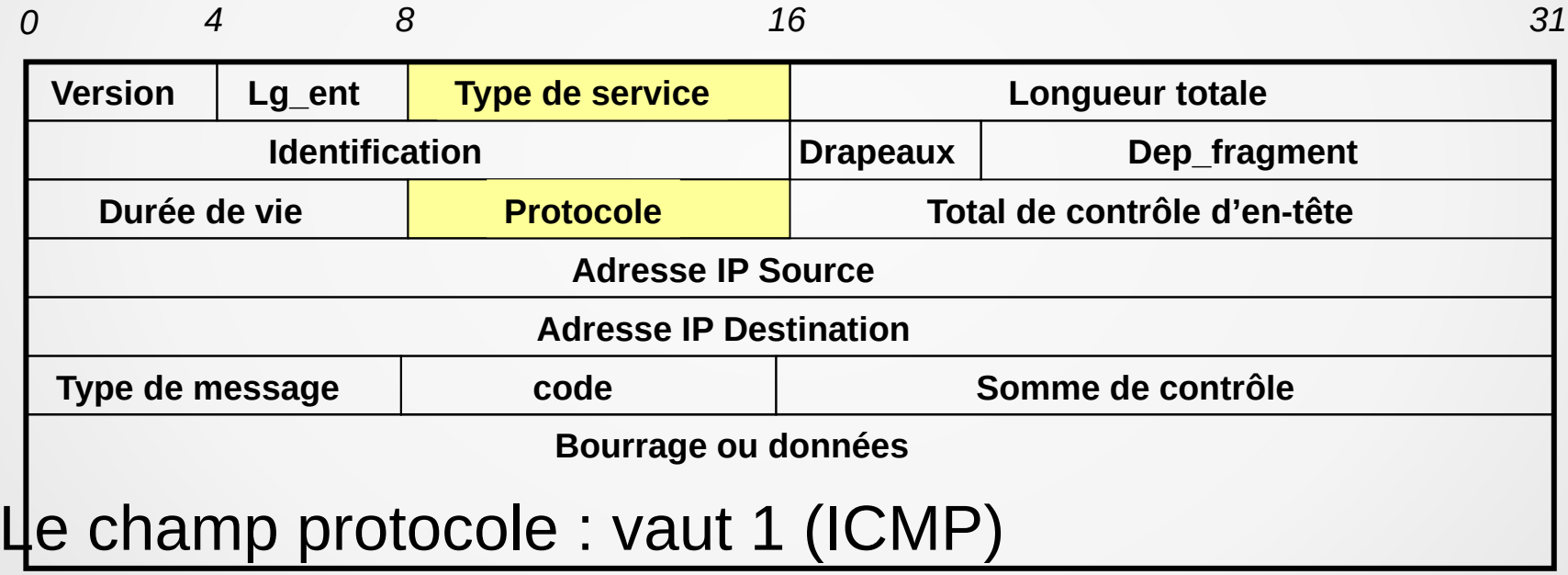
- Format de l'en-tête ICMP



- Très semblable à l'en-tête IP
  - Différence en bleu

Protocole ICMP

- Protocole de niveau 3 contenant … un protocole de niveau 3



- Le champ protocole : vaut 1 (ICMP)
- Le champ type de service vaut 0

Protocole ICMP

- Le champ somme de contrôle est calculé uniquement sur la partie ICMP
- Les type et codes permettent de gérer les messages d'erreur
- Les données peuvent contenir des informations relatives à la panne détectée

Protocole ICMP

- Type et code

Type	Code	description
0	0	Réponse écho
8	0	Echo
3	0	Réseau inaccessible
	1	Hôte innaccessible
	2	Protocole non disponible
	3	Port non accessible
	4	Fragmentation nécessaire mais interdite
	5	Echec du routage
	6	Réseau inconnu

Protocole ICMP

- Type et code (suite)

Type	Code	description
3	7	Machine inconnue
	8	Machine non connectée
	9	Communication avec le réseau interdite
	10	Communication avec la machine interdite
	11	Réseau inaccessible pour le service
	12	Machine inaccessible pour le service
5	0	Redir. pour un hôte (Envoi de nouvelle route + optimale)
	1	Redir. pour un hôte et un service
	2	Redir. pour un réseau
	3	Redir. pour un réseau et un service
9	0	Annonce de routeur
	16	Ne route pas le trafic conventionnel

Protocole ICMP

- Type et code (suite)

Type	Code	description
10	0	Sélection de routeur
11	0	Temps dépassé (indique datagramme détruit)
	1	Temps de réassemblage d'un fragment trop long
12	0	Erreur dans l'en tête
13	0	Demande de l'horloge de la machine (Time stamp)
14	0	Réponse horloge
15	0	Demande d'adresse IP au réseau (obsolète)
16	0	Réponse à 15 (obsolète)
17	0	Demande un masque de sous réseau (obsolète)
18	0	Réponse à 17 (obsolète)

- Attention certains sont obsolètes
  - Site de l'IANA

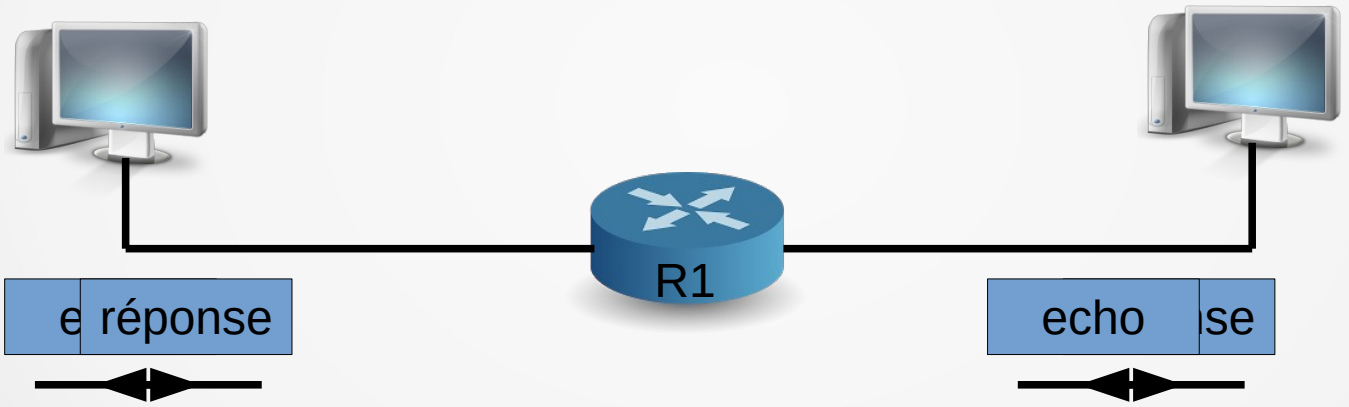
Utilisation du protocole ICMP

- La plus connue : PING
  - Type 8 code 0 : Ping echo (requête)
  - Type 0 code 0 : Ping echo response
- Permet de tester si une machine est joignable

```
C:\Windows\system32>ping www.google.fr

Envoi d'une requête 'ping' sur www.google.fr [173.194.66.94] avec 32 octets de d
onnées :
Réponse de 173.194.66.94 : octets=32 temps=52 ms TTL=46
Réponse de 173.194.66.94 : octets=32 temps=52 ms TTL=46
Réponse de 173.194.66.94 : octets=32 temps=54 ms TTL=46
Réponse de 173.194.66.94 : octets=32 temps=56 ms TTL=46

Statistiques Ping pour 173.194.66.94:
    Paquets : envoyés = 4, reçus = 4, perdus = 0 (perte 0%),
Durée approximative des boucles en millisecondes :
    Minimum = 52ms, Maximum = 56ms, Moyenne = 53ms

C:\Windows\system32>_
```

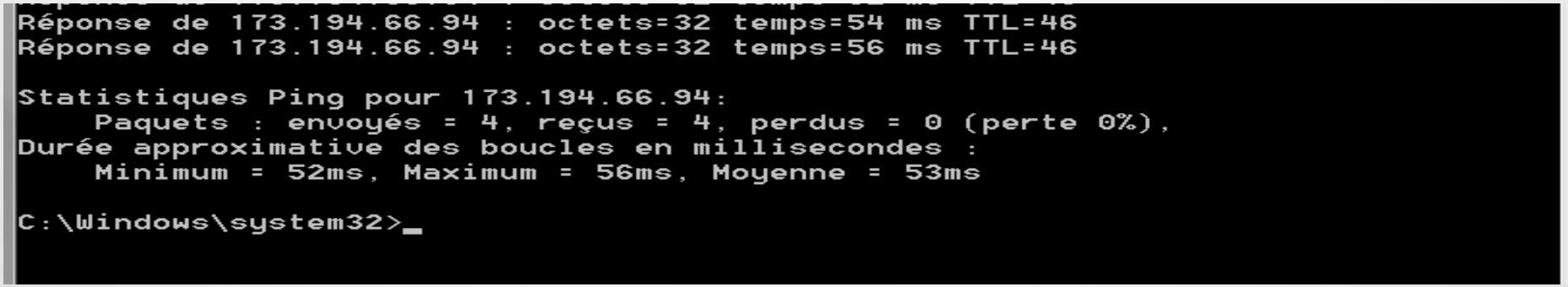
Ping

- Fonctionnement :



Utilisation du protocole ICMP

- Comportement différent sous linux et windows
  - Windows : 4 tentatives par défaut
  - Linux : nombre de tentatives illimité par defaut
    - Attention de ne pas laisser tourner un ping …
- Fournit une statistique de l'état du réseau



Utilisation du protocole ICMP

- Comment connaître le bon couple requête/réponse ?
  - Champ dans la partie "données"
  - Identifier (16 bits)
  - Sequence number (16 bits)
  - Valeurs identiques entre la requête et la réponse

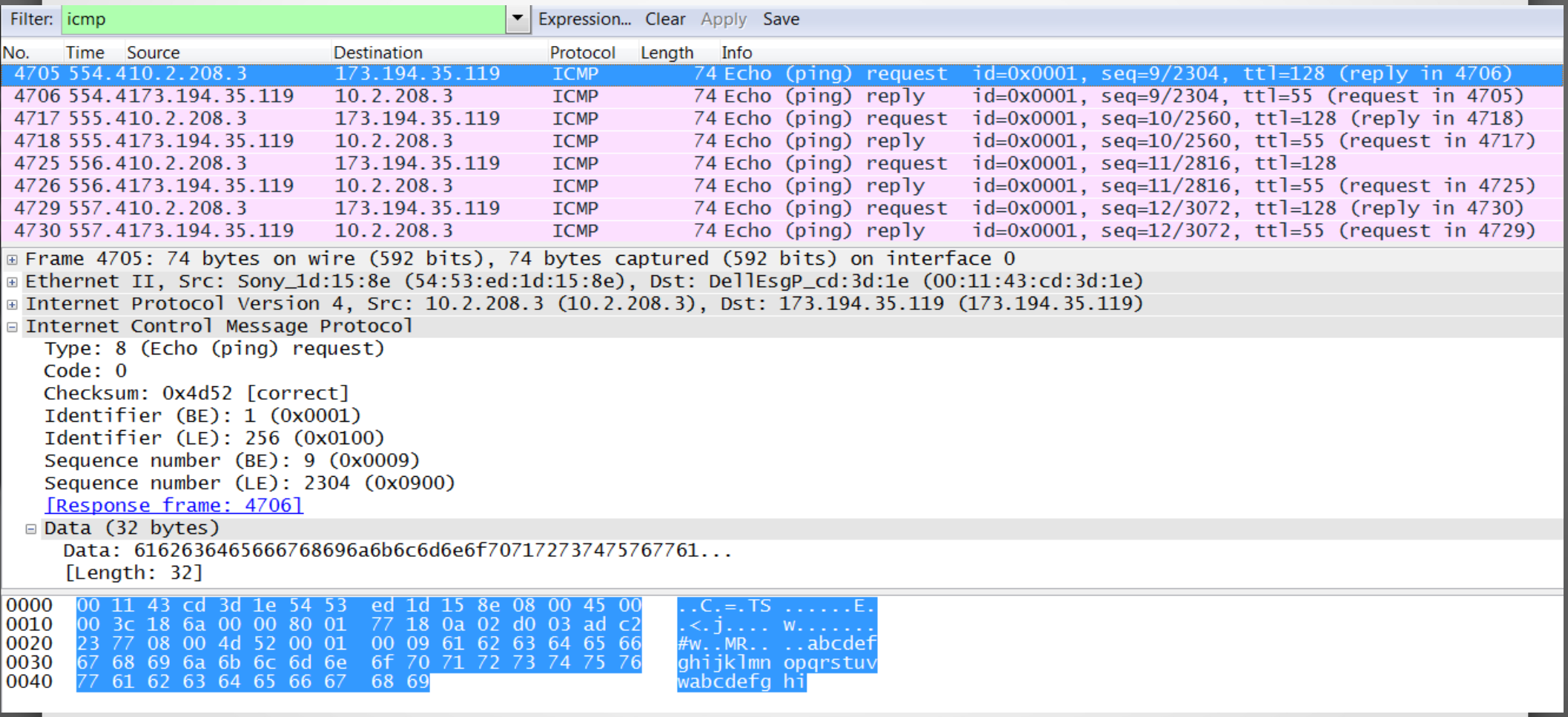
Type de message	**code**	**Somme de contrôle**
identifier		**Sequence number**
Bourrage ou données		

Utilisation du protocole ICMP

- Identifier / Sequence number : 2 pratiques
  - Linux :
    - Identifier unique pour chaque process ping
    - Numéro de séquence incrémenté à chaque ping
  - Windows :
    - Identifier fixe dépendant de la version de windows
    - Numéro de séquence remis à 0 au reboot de la machine

Utilisation du protocole ICMP

- Capture de datagramme ICMP



Utilisation du protocole ICMP

- PING suite et fin
  - La partie donnée est juste du bourrage pour atteindre la taille par défaut 64 octets
  - Suite de caractères

```
Checksum: 0x4d52 [correct]
Identifier (BE): 1 (0x0001)
Identifier (LE): 256 (0x0100)
Sequence number (BE): 9 (0x0009)
Sequence number (LE): 2304 (0x0900)
[Response frame: 4706]
Data (32 bytes)
  Data: 6162636465666768696a6b6c6d6e6f7071727374757677 61...
  [Length: 32]
```

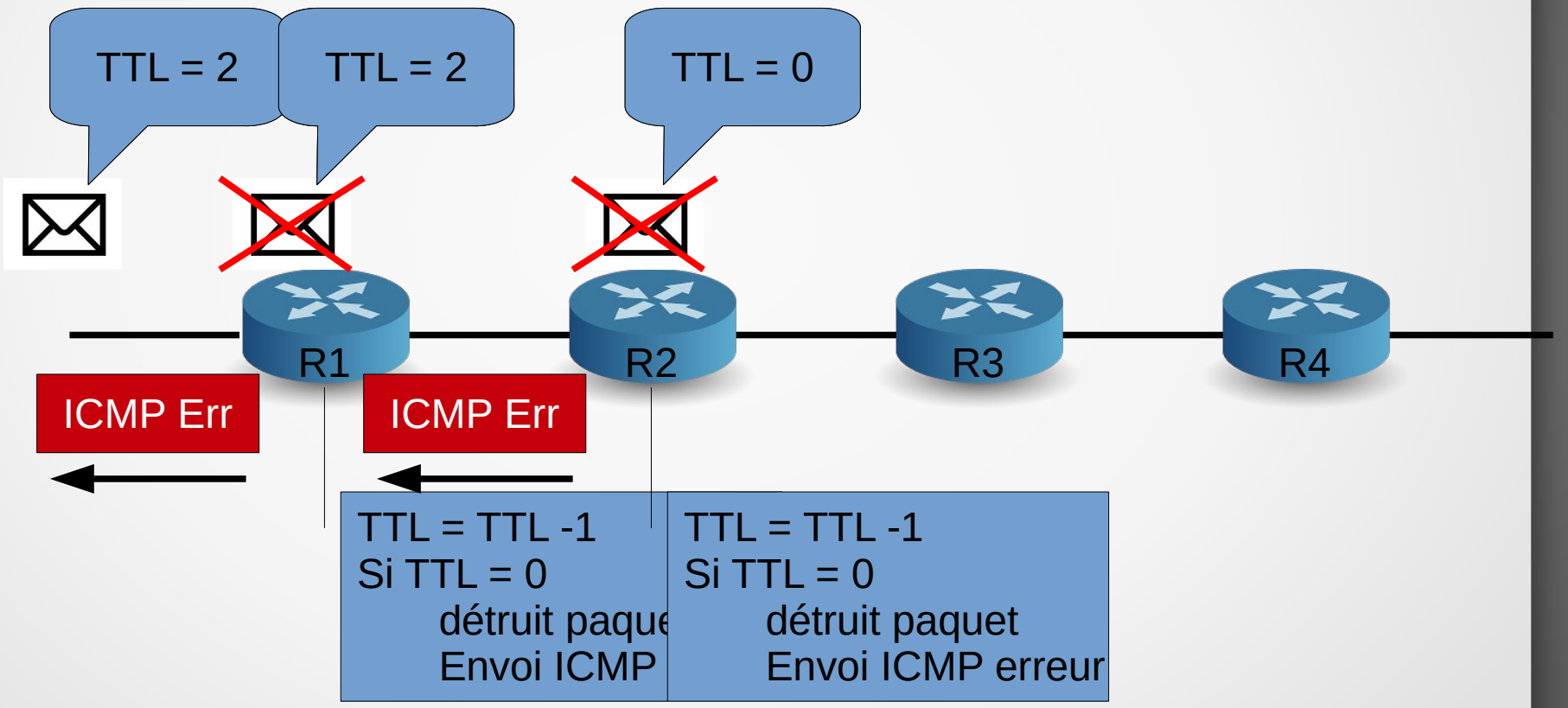
```
0000  00 11 43 cd 3d 1e 54 53  ed 1d 15 8e 08 00 45 00   ..C.=.TS ......E.
0010  00 3c 18 6a 00 00 80 01  77 18 0a 02 d0 03 ad c2   .<.j.... w.......
0020  23 77 08 00 4d 52 00 01  00 09 61 62 63 64 65 66   #w..MR.. ..abcdef
0030  67 68 69 6a 6b 6c 6d 6e  6f 70 71 72 73 74 75 76   ghijklmn opqrstuv
0040  77 61 62 63 64 65 66 67  68 69                    wabcdefg hi
```

Utilisation du protocole ICMP

- Découverte du réseau …
  - Problématique : comment connaître le chemin employé par les paquets sur le réseau ?
  - Objectif recevoir un message de chaque routeur traversé …
  - Commande traceroute / tracert / mtr
  - Utilise ICMP (pas forcément pour les requêtes)

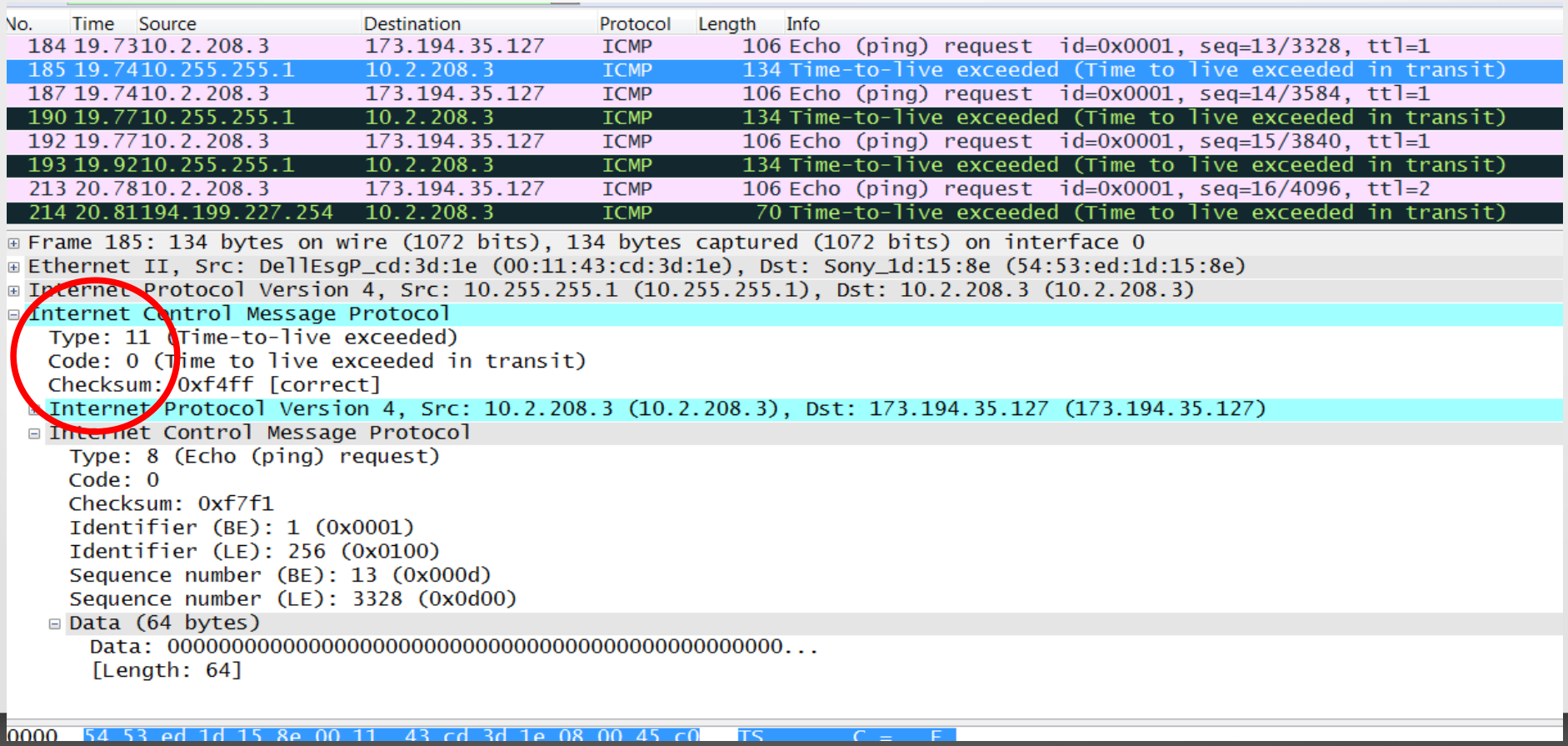
Les messages d'erreur ICMP

- Principe du traceroute



Les messages d'erreur ICMP

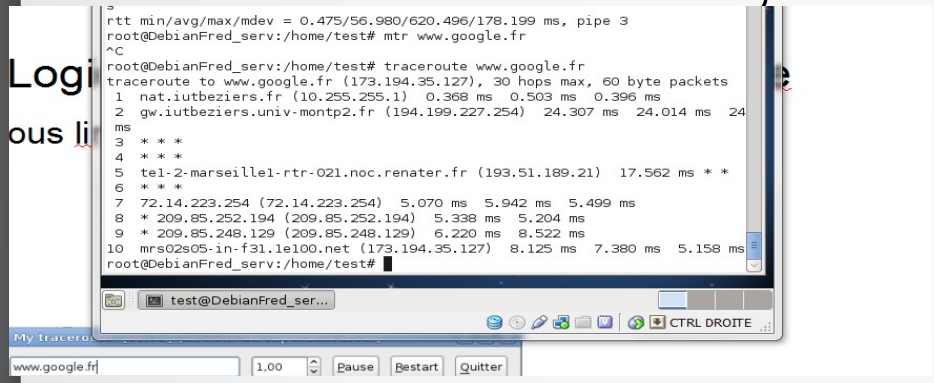
- Traceroute
  - On reçoit des message ICMP : type 11 code 0 (durée de vie dépassée)

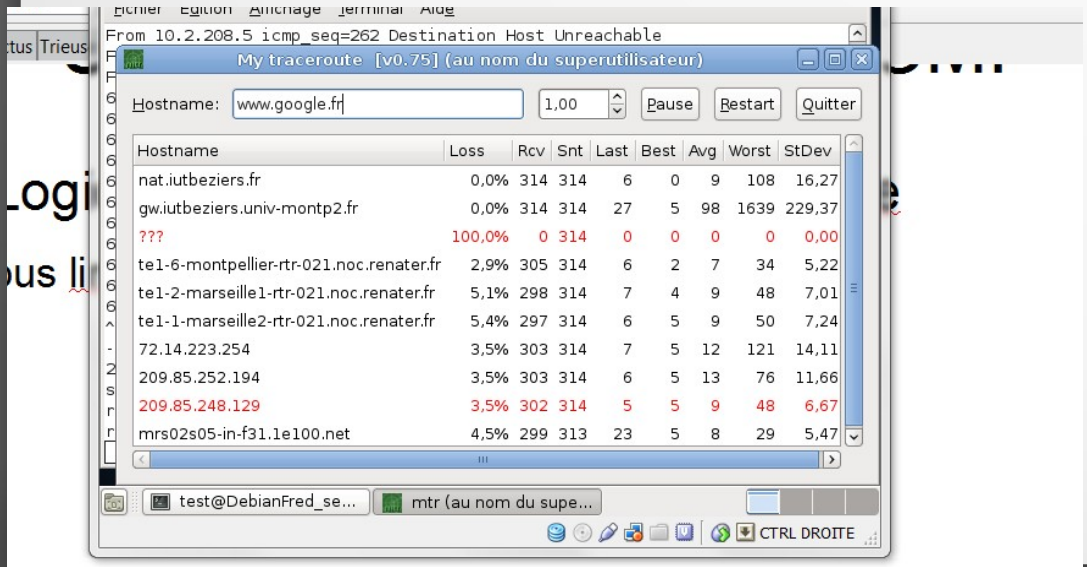


Les messages d'erreur ICMP

- Logiciels pour effectuer un traceroute

- Sous linux : traceroute, mtr





- Sous windows : tracert

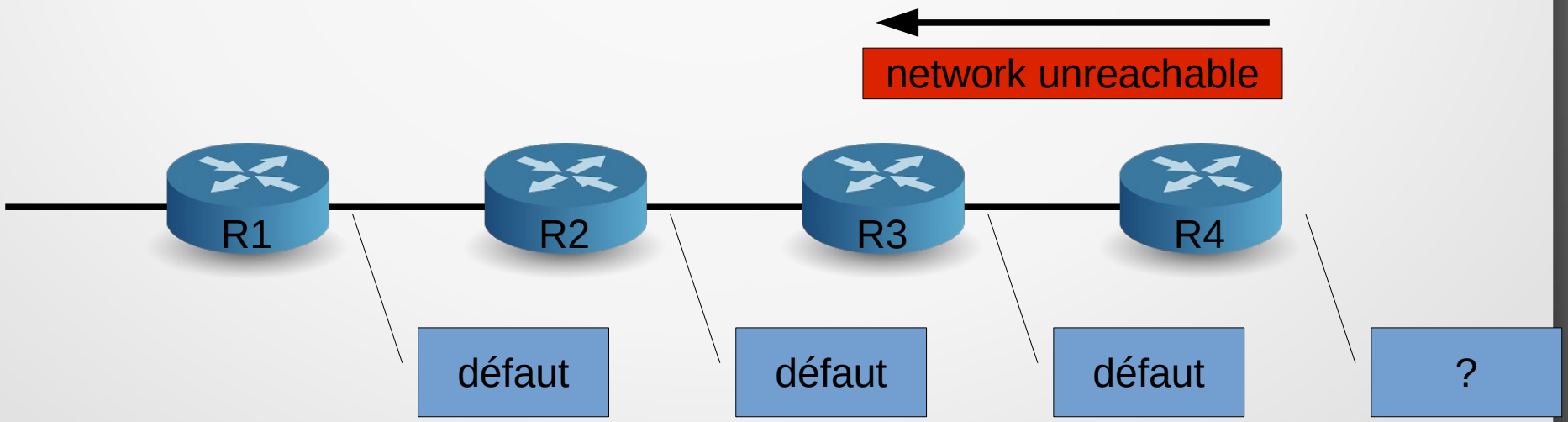
```
^C
C:\Windows\system32>tracert www.google.fr

Détermination de l'itinéraire vers www.google.fr [173.194.35.119]
avec un maximum de 30 sauts :

  1    <1 ms    <1 ms    <1 ms  nat.iutbeziers.fr [10.255.255.1]
  2    28 ms    29 ms    29 ms  gw.iutbeziers.univ-montp2.fr [194.199.227.254]
  3     *        *        *     Délai d'attente de la demande dépassé.
  4     1 ms     2 ms     1 ms  te1-6-montpellier-rtr-021.noc.renater.fr [193.51
.182.190]
  5     5 ms     4 ms     4 ms  te1-2-marseille1-rtr-021.noc.renater.fr [193.51.
189.21]
  6     *        5 ms     5 ms  te1-1-marseille2-rtr-021.noc.renater.fr [193.51.
179.158]
  7     5 ms     5 ms     5 ms  72.14.223.254
  8     5 ms     5 ms     4 ms  209.85.252.194
  9     5 ms     5 ms     5 ms  209.85.248.129
 10     5 ms     5 ms     4 ms  mrs02s05-in-f23.1e100.net [173.194.35.119]

Itinéraire déterminé.
```

Les messages d'erreur ICMP

- Traceroute (suite et fin de la route)
  - Les messages émis ne sont pas toujours en **ICMP echo request**
  - Simple paquet segment udp encapsulé dans IP
  - Permet de passer les routeurs qui ne répondent pas au ping
  - Les réponses sont toujours des **type 11 code 0** (temps dépassé)

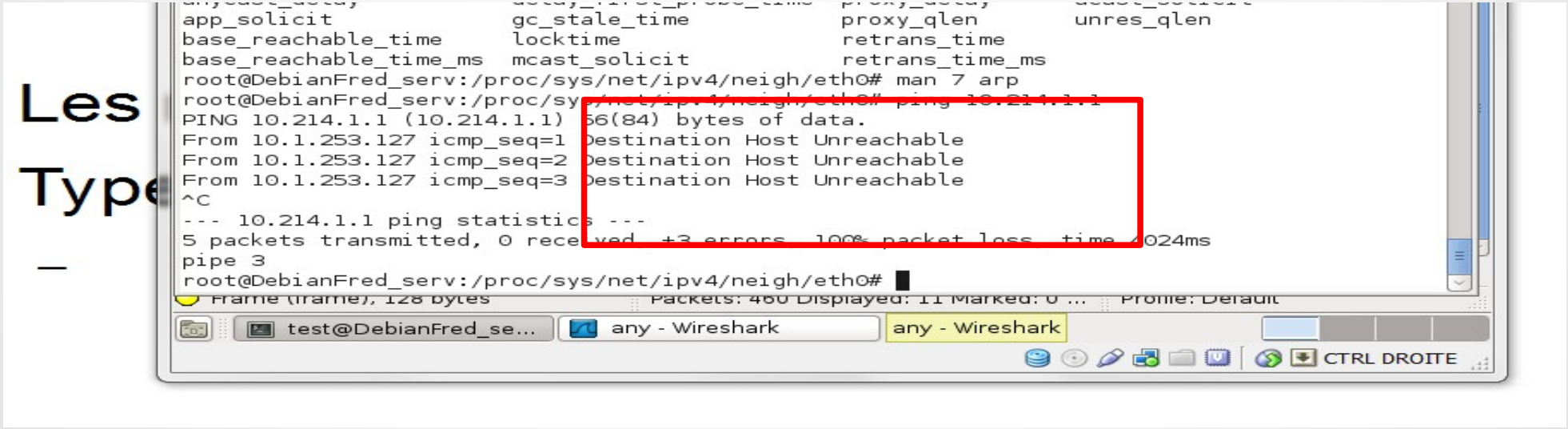
Les messages d'erreur ICMP

- Type 3 code 0 : network unreachable
  - Message envoyé par un **routeur** si un réseau est inaccessible
  - Message envoyé de route par défaut en route par défaut jusqu'au dernier routeur qui ne sait pas.



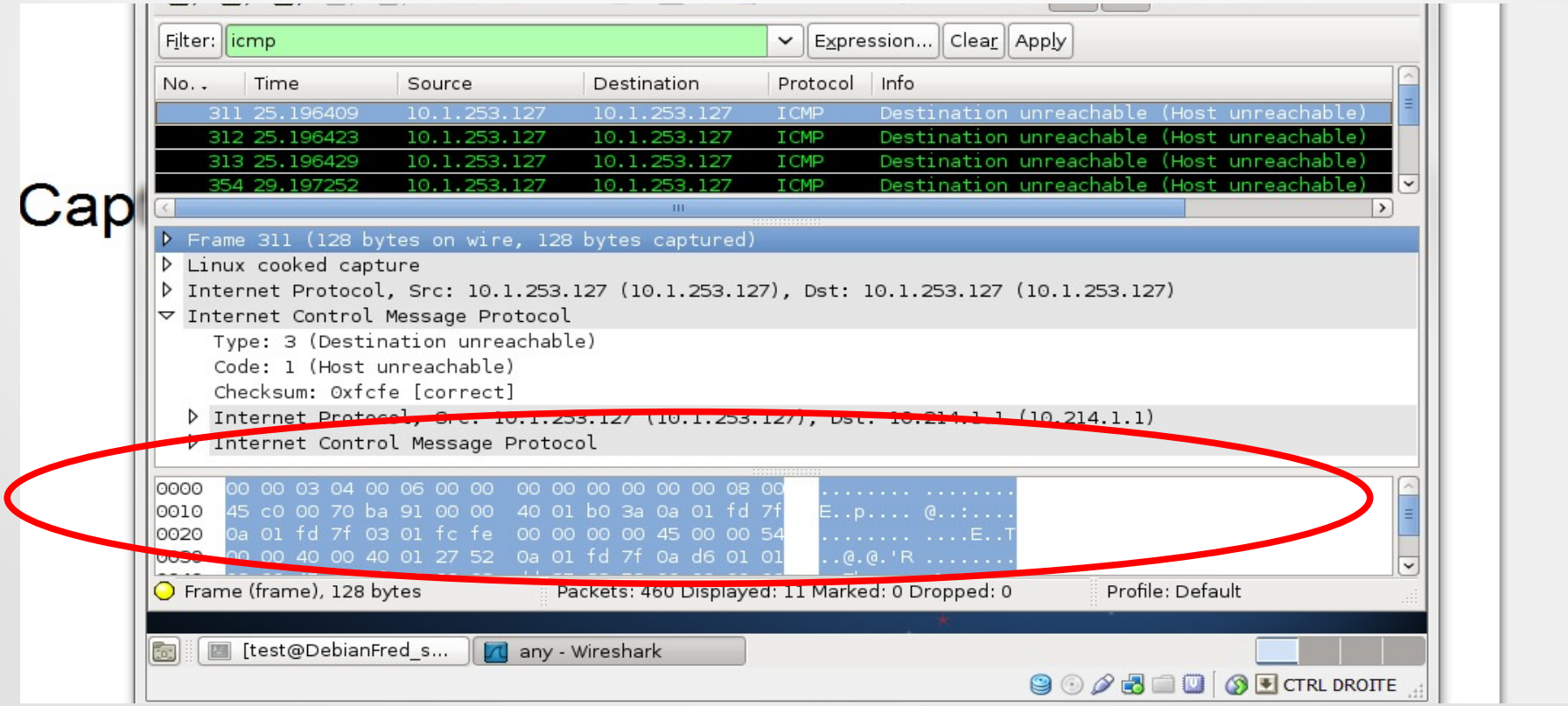
Les messages d'erreur ICMP

- Type 3 code de 1
  - Host unreachable (machine éteinte)



Les messages d'erreur ICMP

- Capture de trame
  - Le message d'erreur contient l'en tête du message incriminé



Les messages d'erreur ICMP

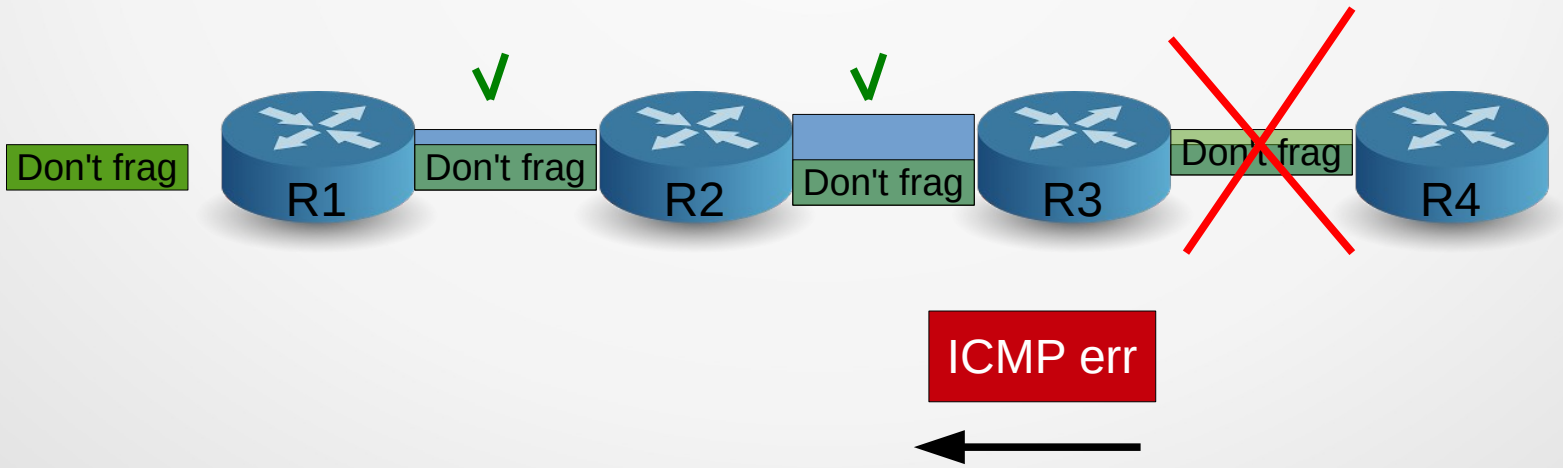
- Type 3 code 2 : protocol unreachable
  - Envoyé par un hôte
  - Lorsqu'un protocole n'est pas disponible sur une machine
  - Protocole non implémenté (par exemple protocole propriétaire)

Les messages d'erreur ICMP

- Type 3 code 3 : port unreachable
  - Message envoyé par un hôte
  - Quand un port n'est pas disponible
  - Exemple SNMP
    - Serveur interroge des machines
    - Il n'y a pas de client snmp sur 1 des machines
    - Elle retourne un message "port unreachable"

Les messages d'erreur ICMP

- Type 3 code 4
  - Message envoyé par un routeur
  - Fragmentation nécessaire et bit Don't Fragment positionné à 1
  - Destruction du message et envoir message erreur

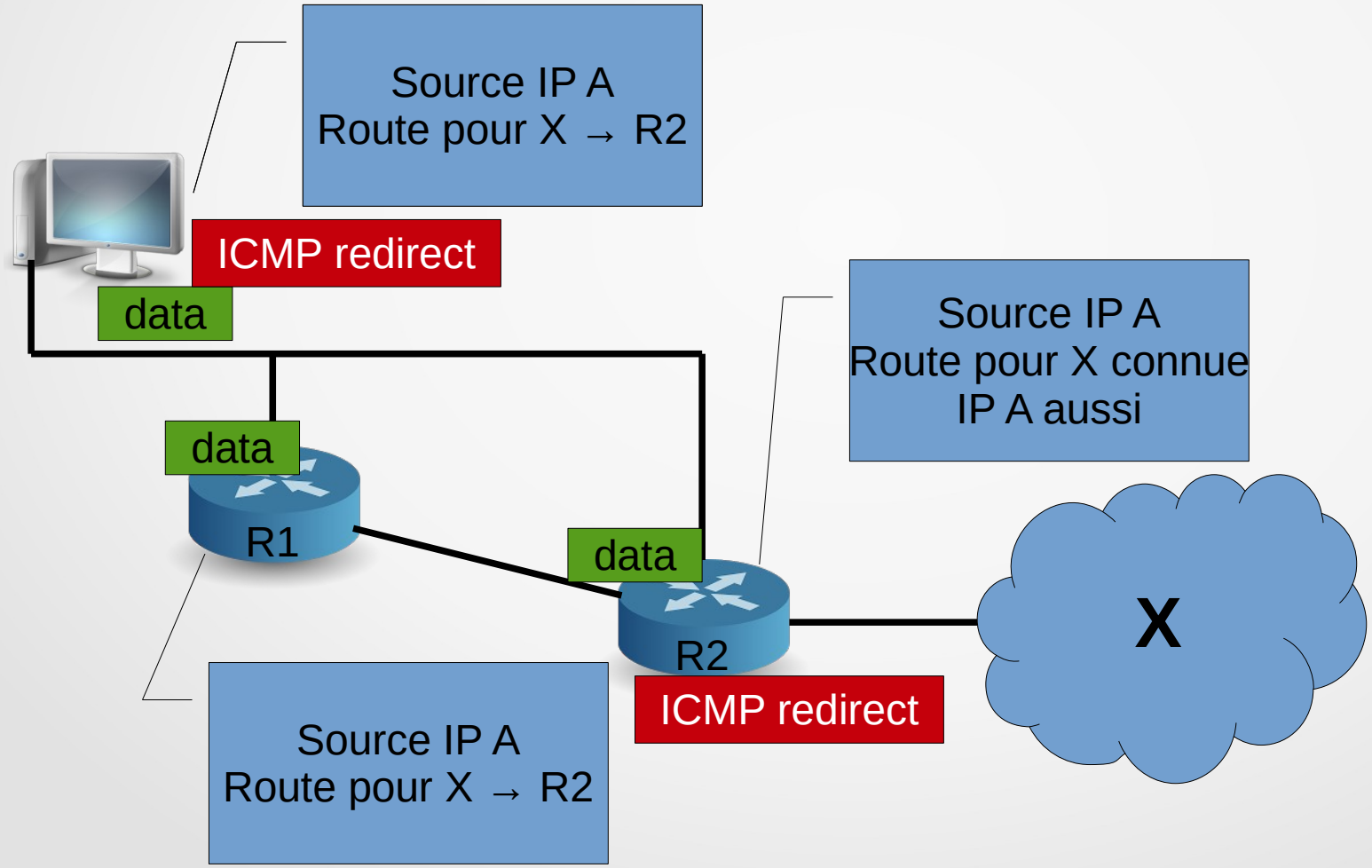


Les messages d'erreur ICMP

- Type 5
  - Message de « redirect »
  - Permet de spécifier une meilleure route à un équipement (par exemple dans le dialogue entre routeurs)
  - Plusieurs code : 0 → 3
    - 0 = Redirect datagrams for the Network.
    - 1 = Redirect datagrams for the Host.
    - 2 = Redirect datagrams for the Type of Service and Network.
    - 3 = Redirect datagrams for the Type of Service and Host.

Les messages d'erreur ICMP

- Exemple



Les failles d'ICMP

- Failles de **Fernando Gont** : professeur, administrateur système et chercheur en réseau, août 2004 plusieurs failles de sécurité dans le protocole ICMP.

  Contacte Microsoft, dev. de Linux, Cisco Systems mais tous n'ont pas répondu, et Cisco a même tenté de breveter ces failles !!

- Exemple de failles
  - Les messages ICMP de type 3 pour les codes 2 ou 3 (voire 4) peuvent clore une connexion TCP
  - Un envoi répété de message ICMP de type 4 (code 0) ralentit grandement le débit d'une connexion
  - Le message ICMP de type 3 pour le code 4 ralentit une connexion en passant le MTU au minimum (68 octets) puis en l'augmentant progressivement

Les failles d'ICMP

- Les attaques classiques … mais obsolètes
  - Ping of Death (POD)
    - Envoi d'un ping de taille non conventionnelle (>64 octets)
    - Faisait crasher les piles IP des machines
    - Obsolète car maintenant la pile ip est mieux conçue

Les failles d'ICMP

- Unreachable Host
  - Envoi d'un message de type 3 code 1
  - Provoque la déconnexion des sessions et bloque l'utilisateur
  - Attaque simple car les envois de datagramme ICMP peuvent être envoyés sans grand débit
    - Octobre 1998, l'attaque Smack inondait de paquets ICMP aléatoires de type UnReachable. Les systèmes d'exploitations visés étaient BSD, RedHAT, Slackware.
  - Obsolète

Les failles d'ICMP

- ICMP redirect
  - Envoi de messages ICMP de type "*Redirect*" à une cible pouvant être aussi bien un serveur comme un routeur
  - Le datagramme informe la victime qu'il faut passer par une autre chemin et peut provoquer une indisponibilité Wan

Les failles d'ICMP

- Ping flood
  - Inonder une machine de paquets ICMP pour la rendre indisponible
  - L'objectif est de saturer la bande passante
  - La plus répandu des attaques par déni de service (DoS)
- Différent de Ping of Death car ici l’objectif est de saturer la bande passante
  - Le 23 octobre 2002, attaque de type Ping Flood en mode distribuée sur les serveurs racines DNS de l'Internet. Seulement 4 serveurs sur 13 sont restés disponibles.

Les failles d'ICMP

- SMURF attack
  - Envoi d'un ping en broadcast avec comme IP source l'adresse de la cible
  - Ceux qui répondent vont ralentir la cible
  - Le nom smurf vient du fichier "smurf.c", code source du programme réalisé en 1997 par Tfreak.

    (sources wikipédia et http://www.phreak.org/archives/exploits/)

Protocole de couche 3

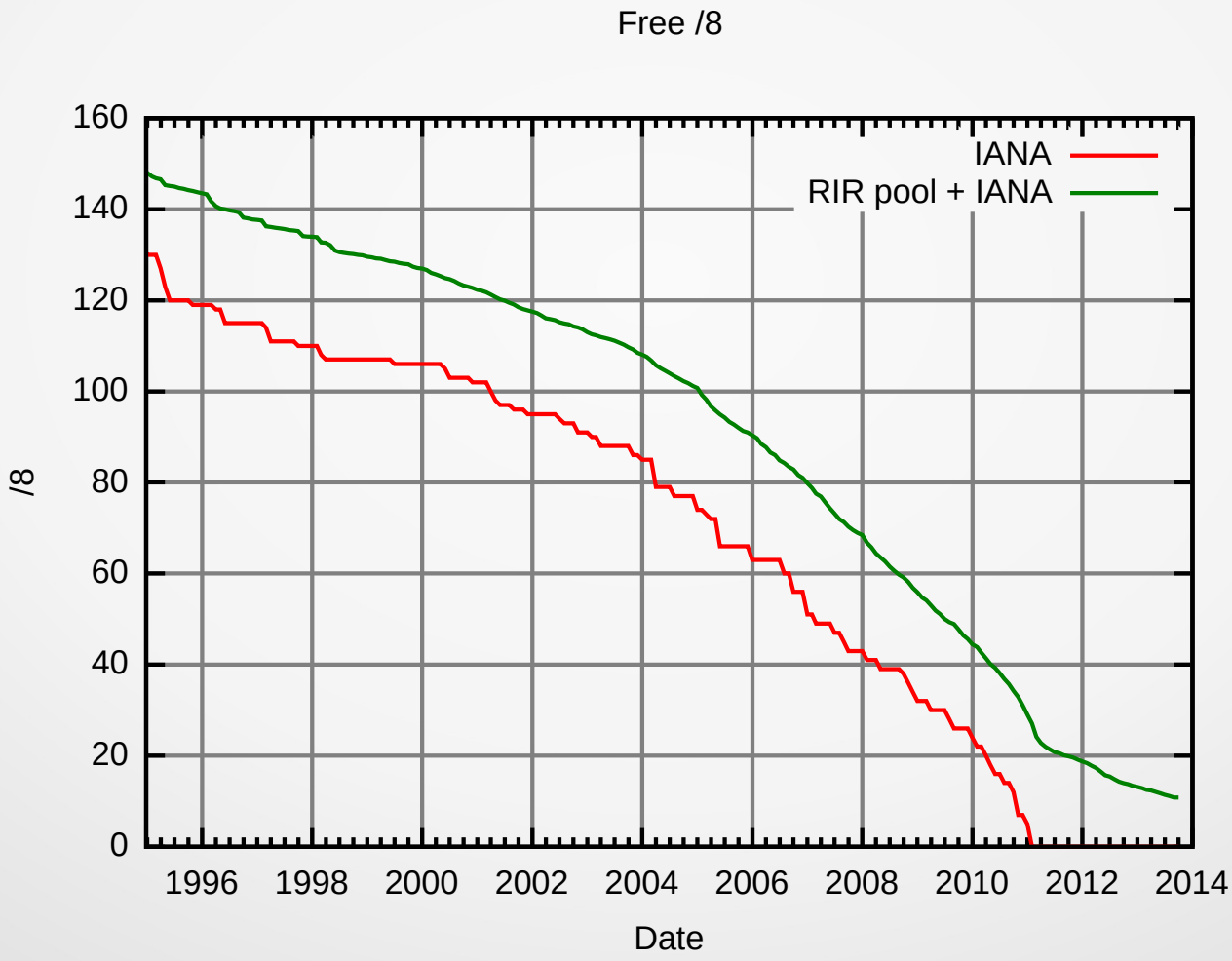
- Il en existe d'autre
  - RIP / OSPF / BGP : protocoles de routage abordés plus tard
  - IGMP : protocole lié à la mutlidiffusion (on parle à un groupe d'utilisateurs)
  - BOOTP : découverte d'une @IP pour machine sans disque dur

Petit + couche 3

- Quelques bases sur IPv6
  - Protocole de niveau 3
  - Définit par la RFC 2460
  - Pourquoi IPV6 ?

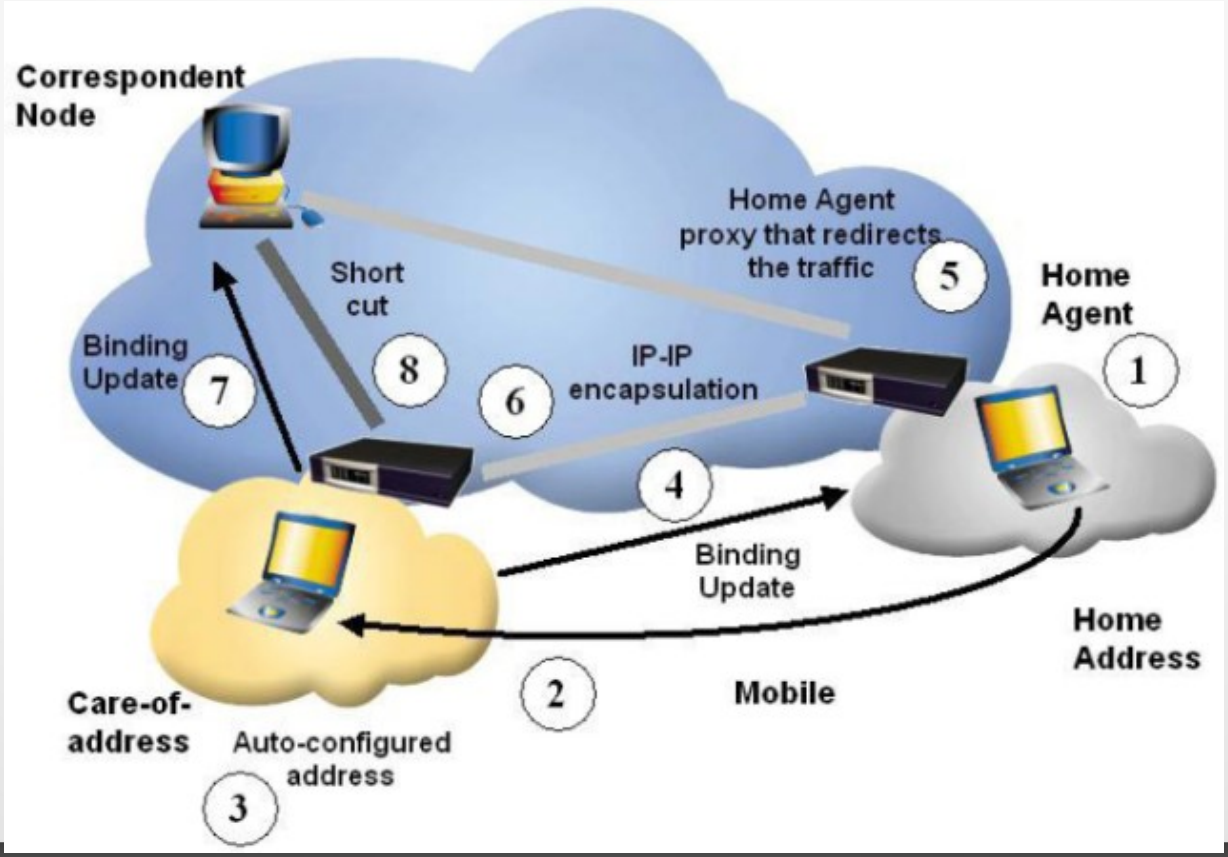
Pourquoi IPv6 ?

- 1 – épuisement des adresses IPv4



Pourquoi IPv6 ?

- Apparition de nouvelles problématiques
  - Exemple la **mobilité** : un terminal ne reste plus au même endroit tout le temps



Apports d'IPv6

- Changement d'adresse
  - Adresses sur 128 bits
  - On passe de 2^{32} ($4{,}3\times10^{9}$) adresses à 2^{128} ($3{,}4\times10^{38}$)
  - 667 millions de milliards d'adresses IP disponibles par mm2 de la surface de la Terre

Format des adresses IPv6

- 16 octets (au lieu de 4)
- Hexadécimal (au lieu de décimal pointé)
- 8 groupes de 2 octets séparés par des :
  - Ex : 2001:0db8:0000:85a3:0000:0000:ac1f:8001
- On peut simplifier
  - 1 et 1 seul groupe de 0 peut être remplacé par ::
    - Ex : 2001:0db8:0000:85a3::ac1f:8001
  - On peut supprimer les 0 inutiles
    - Ex : 2001:db8:0:85a3::ac1f:8001

Format des adresses IPv6

- Pas de classe d'addresse
  - Notation CIDR (Classless Inter Domain Routing)
  - Masque de sous réseau défini le préfixe (partie commune de l'adresse : nom de famille)
  - Le préfixe 2001:db8:1f89::/48 représente l'ensemble des adresses qui commence à 2001:db8:1f89:0:0:0:0:0 et finit à 2001:db8:1f89:ffff:ffff:ffff:ffff:ffff

Format des adresses IPv6

- Types d'adresses spéciaux

Préfixe	Description
::/8	Adresses réservées
2000 ::/3	Adresses unicast routables sur internet
Fc00 ::/7	Adresses locales uniques
Fe80 ::/10	Adresses lien local
Ff00 ::/8	Adresses multicast

Catégories d'adresses IPv6

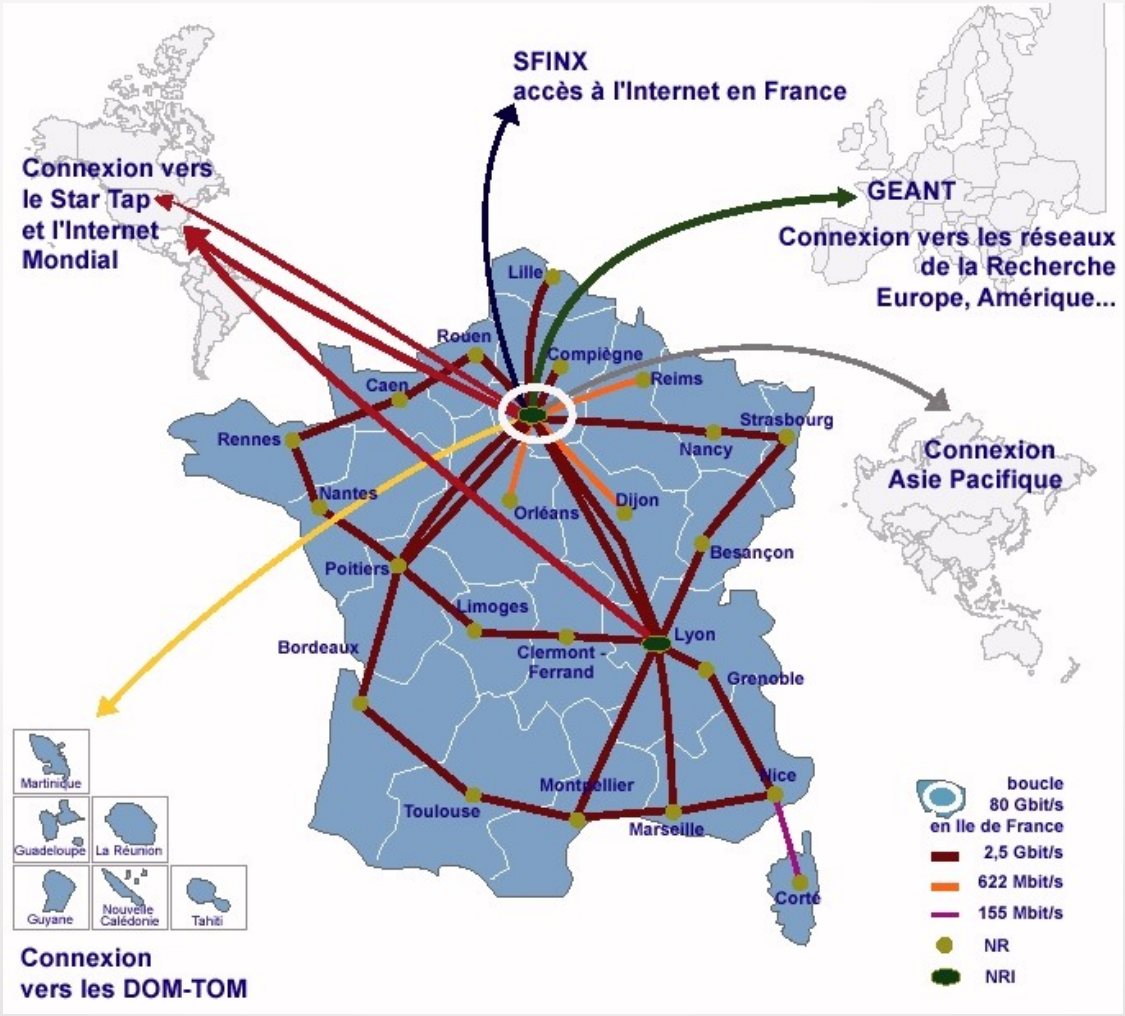
- Adresses réservées
  - ::/128 adresse indéterminée
  - ::1/128 adresse de loopback (eq ip127.0.0.1)
  - ::FFFF :0 :0/96 adresses IPv4 dans IPv6
    - 174.162.12.1 → ::FFFF:174.162.12.1
      ou ::FFFF :EA2:C01
  - ...

Catégories d'adresses IPv6

- Définies par leur portée
  - Adresses globales unicast
    - Adresses routables
  - Adresses locales uniques
    - Equivalent adresses privées (peuvent être routables)
  - Adresses de lien local
    - Adresse uniques sur un lien local (couche 2)
  - Adresses multicast
    - Pour les groupes

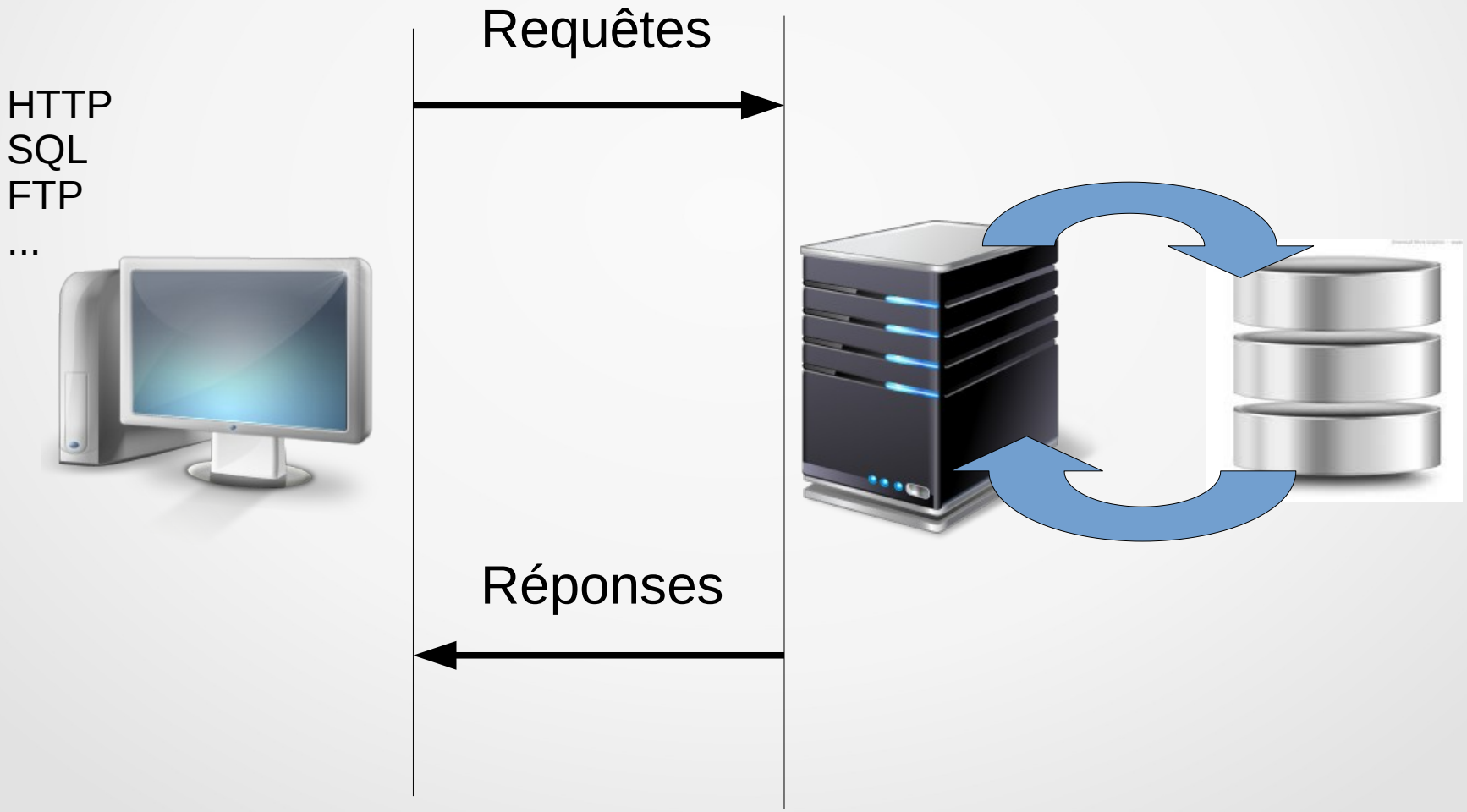
Protocoles de couche 4

- Le transport



Analyse du besoin

- Applications en mode client serveur



Fonctionnement Client/Serveur

- Architecture 2 tiers
- Architecture du serveur
  - Le serveur ouvre un point de connexion sur le réseau
  - Il attend (infini) les requêtes de la part de clients
  - S'il reçoit une requête il la traite et renvoie le résultat au client
- Est ce que la couche réseau permet de traiter tout ça ?

Utilité de la couche transport

- La couche réseau permet :
  - L'envoi de paquets n'importe où dans le réseau
  - D'envoyer des données de n'importe quelle taille
  - D'adresser les machines de façon unique
- Elle ne permet pas :
  - De garantir l'arrivée d'un paquet
  - De respecter l'ordre des paquets
  - D'utiliser plusieurs applications simultanément
  - De gérer la sécurité

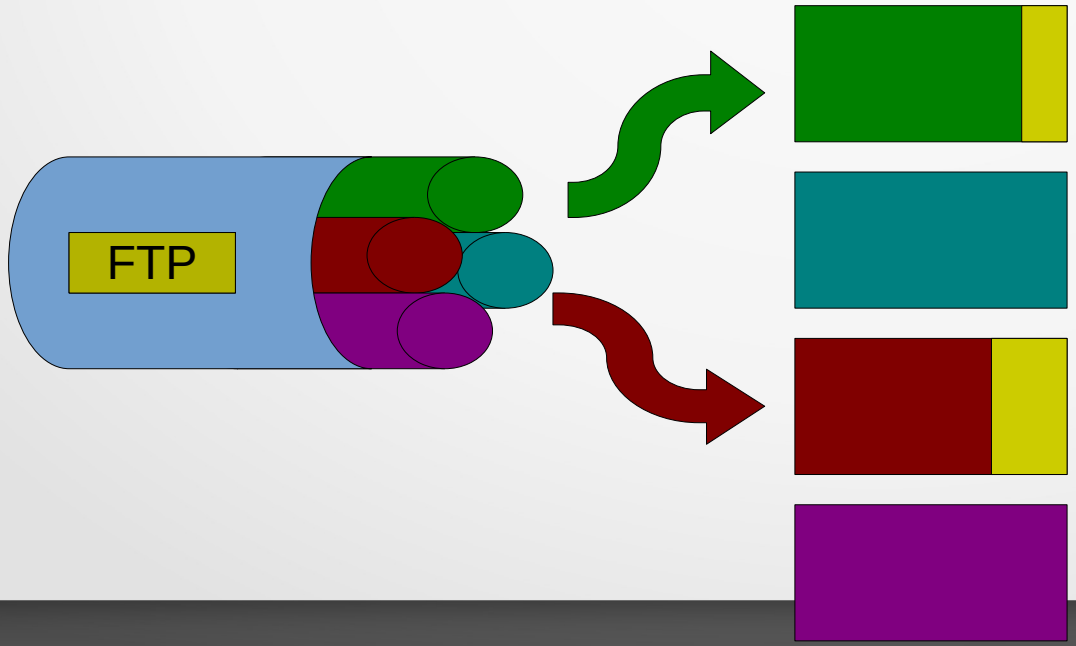
2 grandes familles de transport

- En fonction des besoins
  - Fiabilité
  - Rapidité
- 2 solutions
  - Mode connecté (TCP)
    - Plus lourd
  - Mode non connecté (UDP)
    - Plus léger

- Objectif partagé : permettre à plusieurs processus de partager une liaison

Notion commune

- La notion de **port**
- Point d'accès à un service (une application)
- Peut être considéré comme une file d'attente de messages



Les applications vident la file d'attente

Les ports

- Comment on les représente ?
  - Numéro sur 16 bits
  - De 0 → 65535
- Différents types de ports (RFC 6335)
  - 0 → 1023 ports système (assignés par l'IANA)
    - Ex : 80 HTTP, 22 SSH, 23 telnet, …
  - 1024 → 49151 ports utilisateur
    - Associés à des logiciels
  - 49151 → 65535 ports dynamiques et/ou privés
    - Utilisés à l'autre bout d'une connexion avec un port système

Exemple d'utilisation des ports

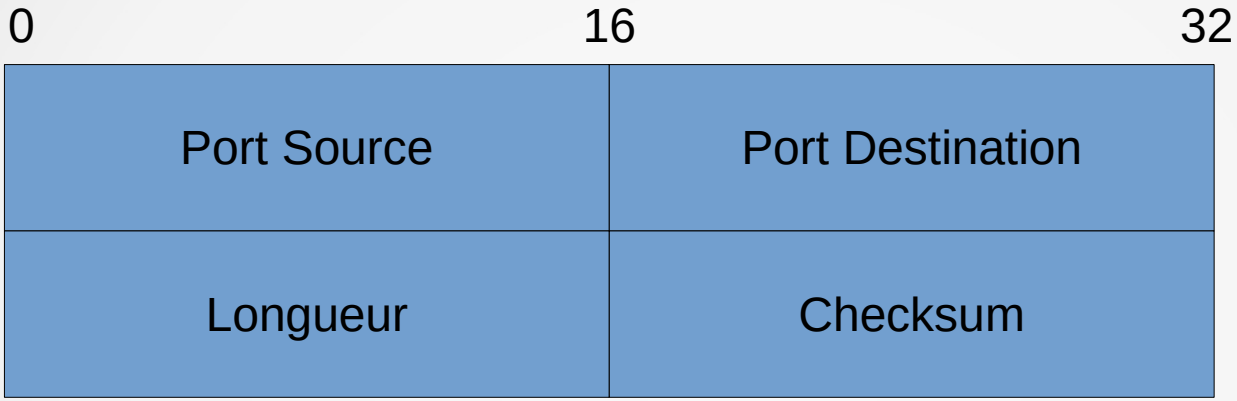
- Demande de page HTTP
- Le client demande un port → 55382
- Le client envoie sa requête :
  - IP client:55382 → IP serveur:80
- Le serveur répond en envoyant la page web
  - IP serveur:80 → IP client:55382
  - Le port 55382 est libéré (soit de suite, soit à la fin de la connexion)
- Pour lister les ports outil **NMAP**

Mode de transport non connecté

- Le plus simple
- Protocole **UDP**
- User Datagram Protocol
- RFC 768
- Caractérisé par :
  - Une machine source / destination
  - Un port source / destination
- Le port client est rendu après utilisation
- Le port serveur attend un autre client

Protocole UDP

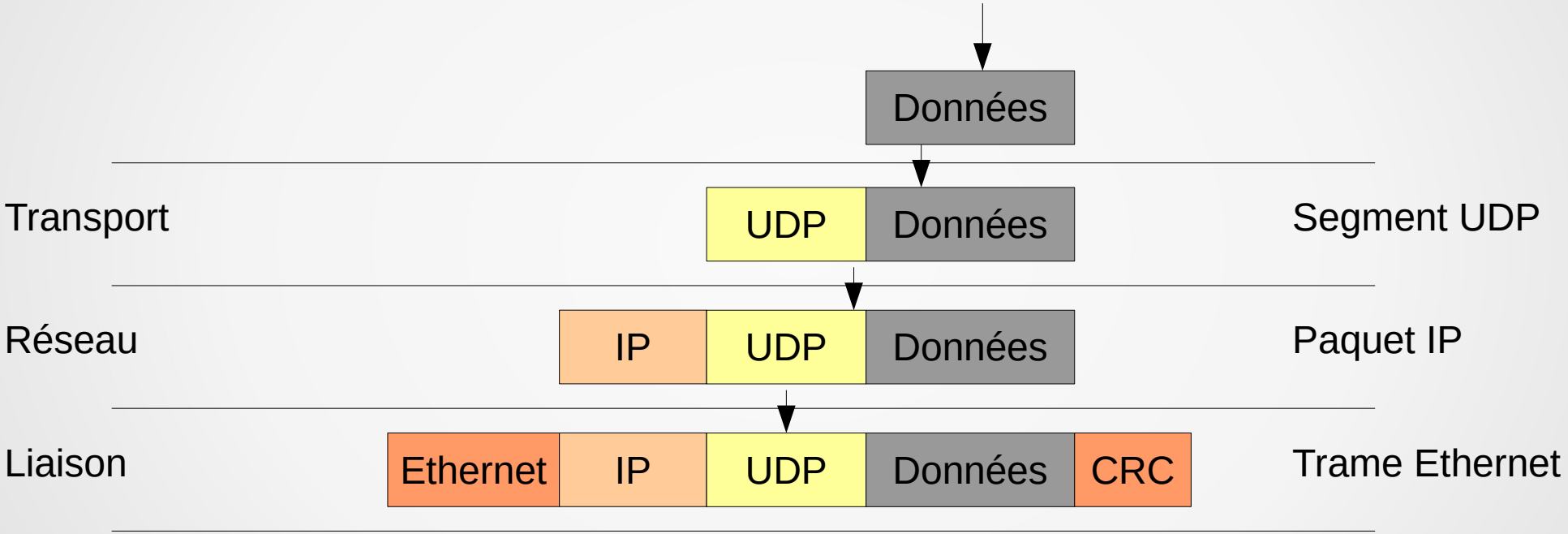
- L'en tête (PCI)



- Longueur : longueur du segment UDP (en-tête + données)
- Checksum : somme de contrôle sur le PDU de niveau 4

Protocole UDP

- Encapsulation



Utilisation d'UDP

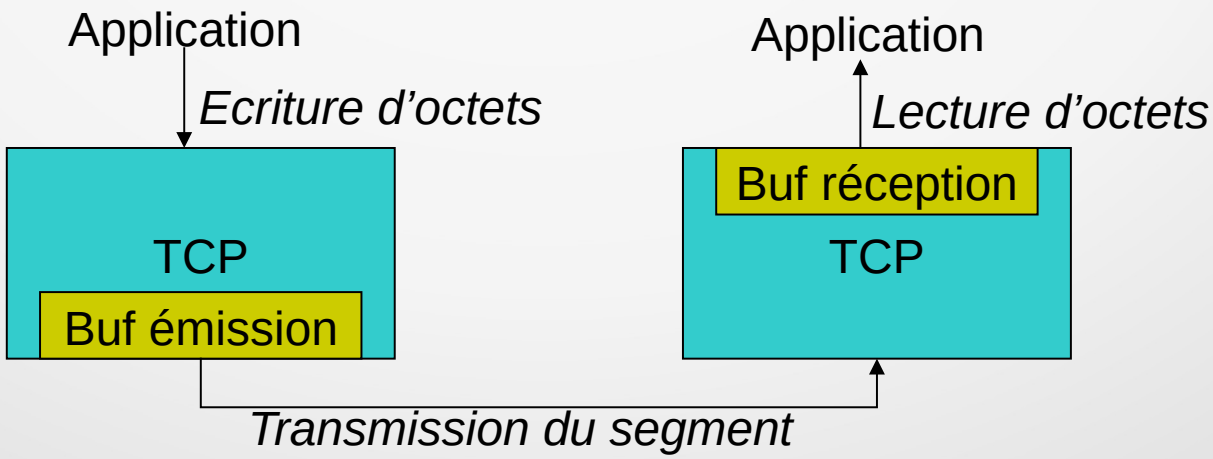
- Dans les applications unicast ou multicast
  - Architectures client serveur : DNS, RPC, SNMP, etc.
  - Application multimédia : voix sur IP, téléphonie sur IP
- Limites :
  - Non fiable (perte de segments, duplication, …)
  - Possibilité de corriger au niveau application ou ajout de spécificités à UDP (protocole RTP : real time protocol)

Mode de transport connecté

- Plus complexe
- Protocole **TCP (**Transmission Control Protocol)
- RFC 793
- Service orienté connexion : fiable
- Communication bi-directionnelle simultanée
- Remise garantie des segments dans le bon ordre, sans perte, par contre non structuré

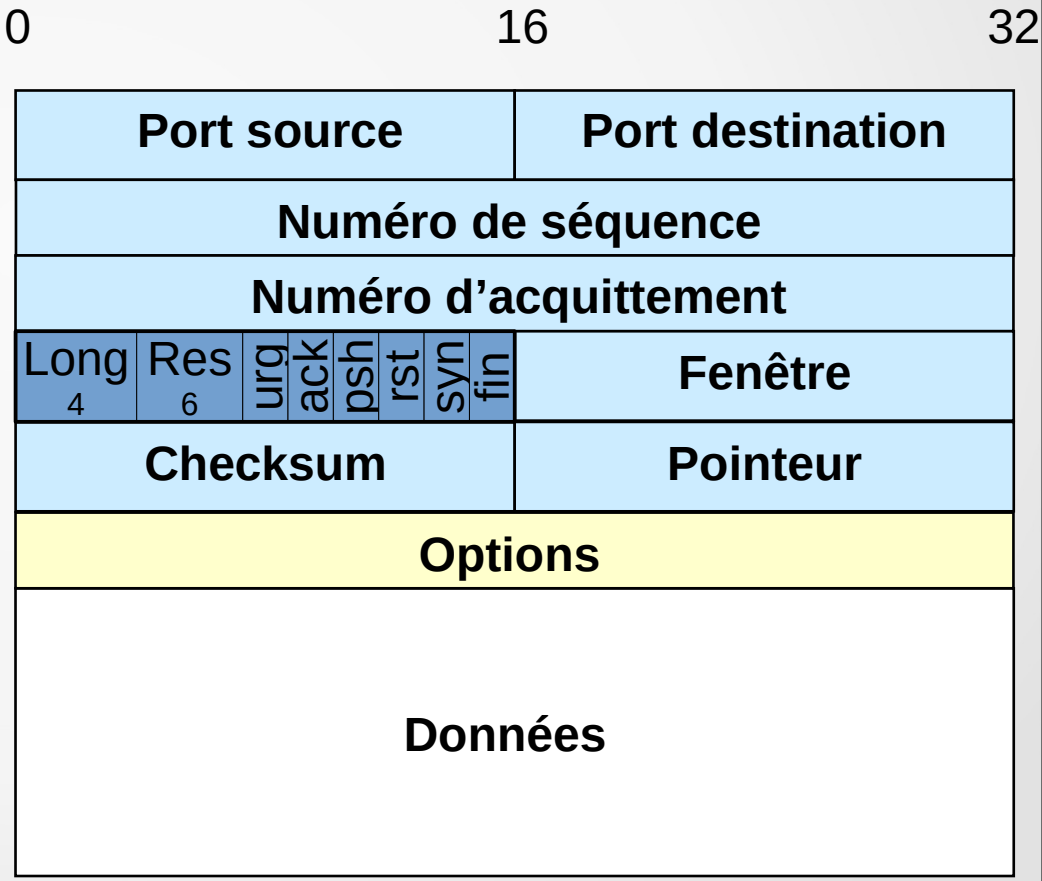
Protocole TCP

- Flot de données
  - Pas de transmission individuelle d’octets
  - A l’émission TCP bufférise les octets quand leur nombre est suffisant TCP envoie un segment
  - A la réception TCP place le contenu des segments dans le buffer de réception et l’application vient lire un nombre d’octets



Protocole TCP

- En-tête TCP
  - Long : longueur de l'en-tête (en mots de 32 bits)
  - Res : réservé
  - Ports : identique UDP
  - Fenêtre : largeur de la fenêtre de réception
  - Pointeur : pointeur pour les données urgentes

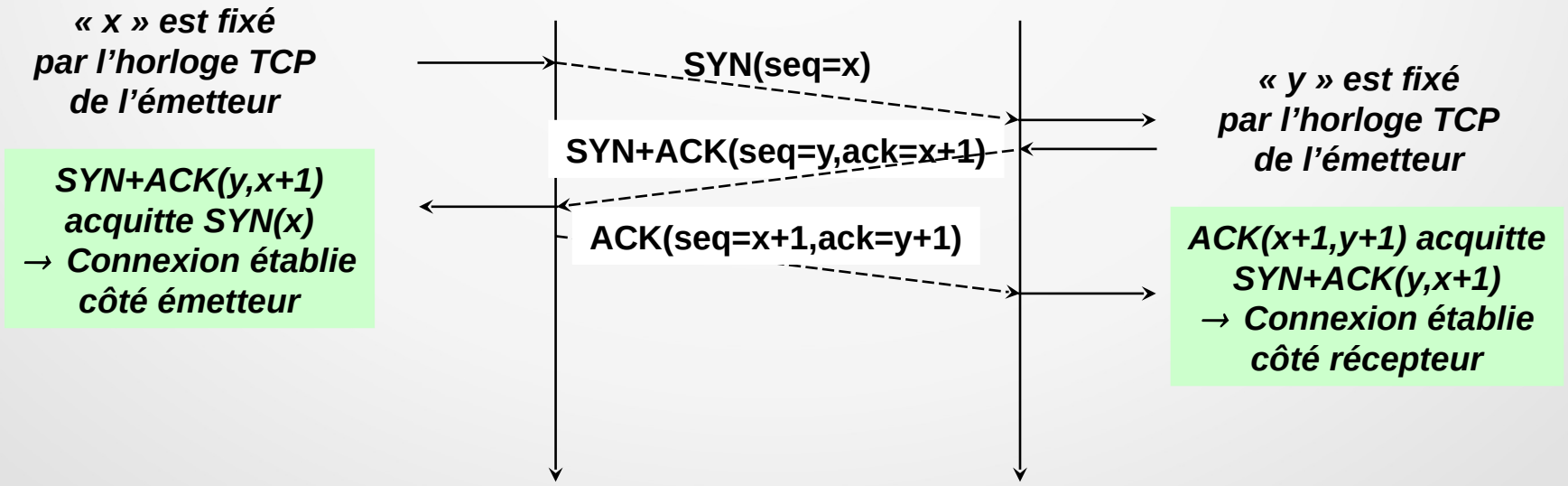


Protocole TCP

- Les drapeaux
  - URG : le pointeur de données urgentes est utilisé
  - ACK : indique un acquittement des données
  - PSH : demande d'envoi des données à la couche application sans attendre de remplir le tampon mémoire
  - RST : demande réinitialisation de la connexion
  - SYN : demande la synchronisation des numéros de séquence (ouverture de connexion)
  - FIN : indique la fin de transmission

Protocole TCP

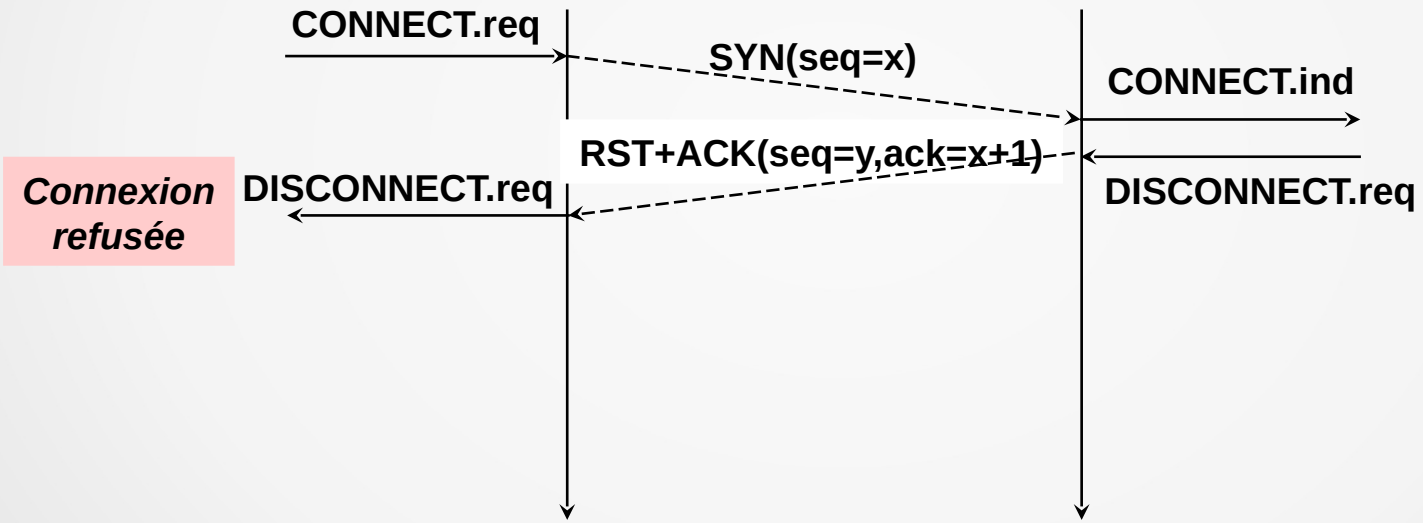
- Ouverture de connexion (3 étapes)
  - Avec **SYN**=1, « consomme » un numéro de séquence
  - Avec **ACK**=1, indique un numéro d'acquittement valide



Les horloges TCP sont incrémentées de 1 toutes les 4 micro-secondes et à chaque établissement de connexions

Protocole TCP

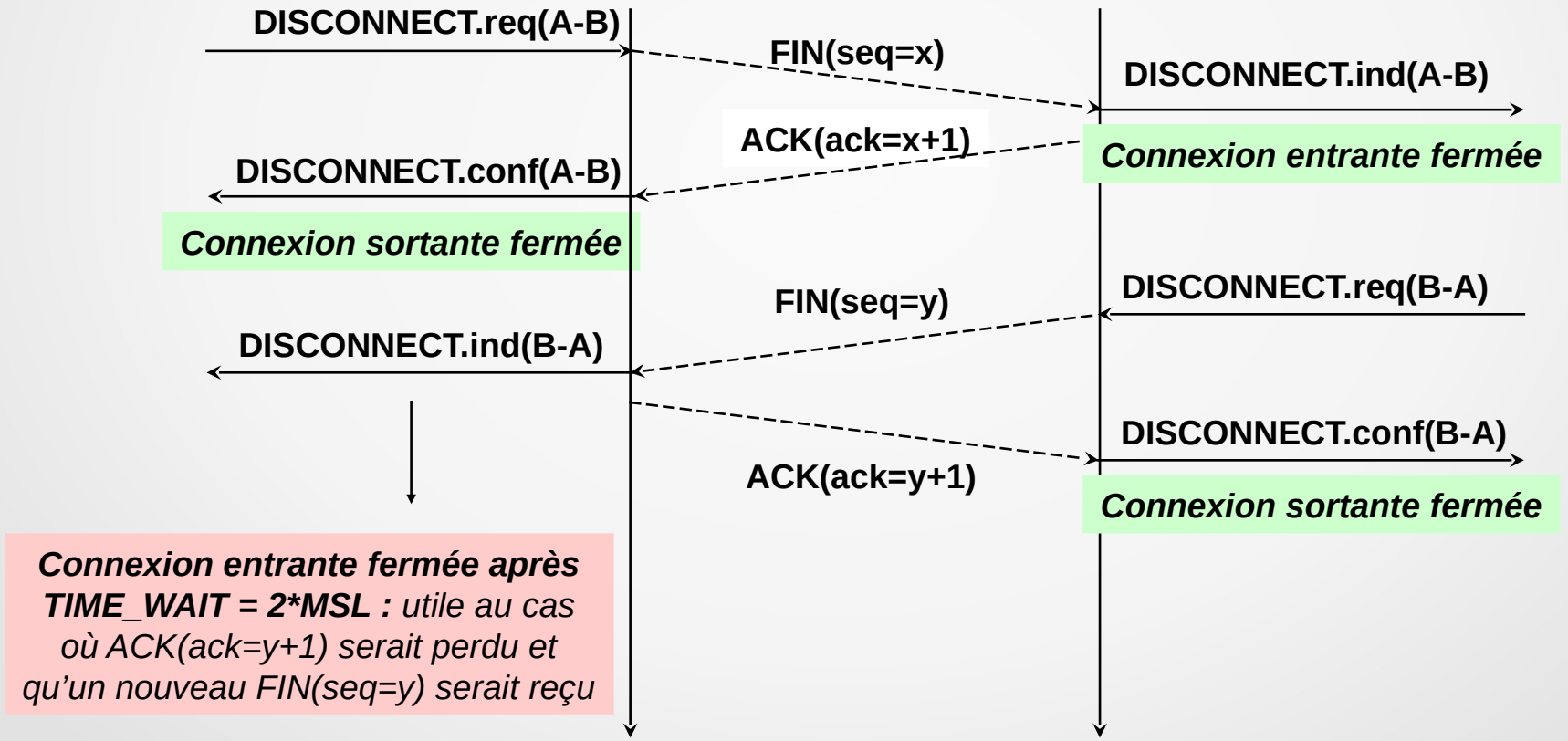
- Utilisation du drapeau RST



Un segment TCP avec le drapeau RST=1 → fermeture de la connexion TCP

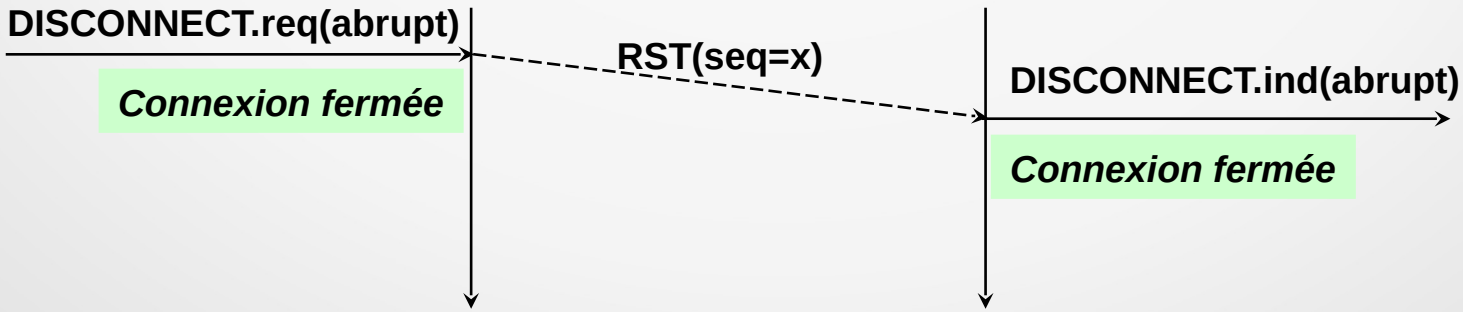
Protocole TCP

- Fermeture de connexion



Protocole TCP

- Autre fermeture plus brutale
  - Des données peuvent être perdues
  - Pas d'attente
  - Si nouveau segment arrive → on renvoie un RST

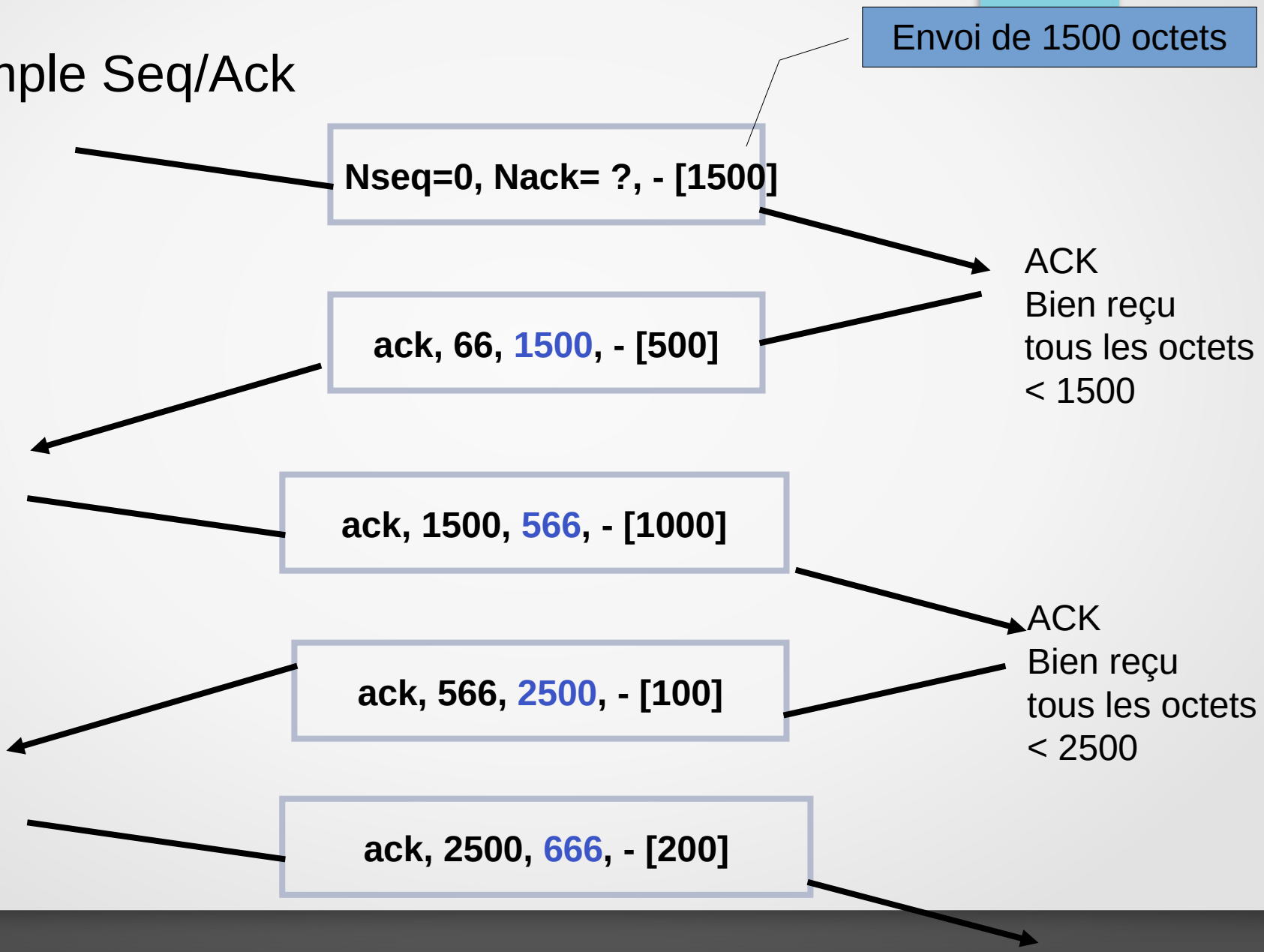


Protocole TCP

- Gestion des erreurs
  - Numéros de séquence et d'acquittement.
  - Numéro de séquence :
    - Chaque octet transmis consomme un numéro de séquence
  - Numéro d'acquittement :
    - SI ACK=1 il indique le numéro du prochain octet attendu
    - Sinon pas de sens
    - Tous les octets de numéro < au numéro d'ack sont validés

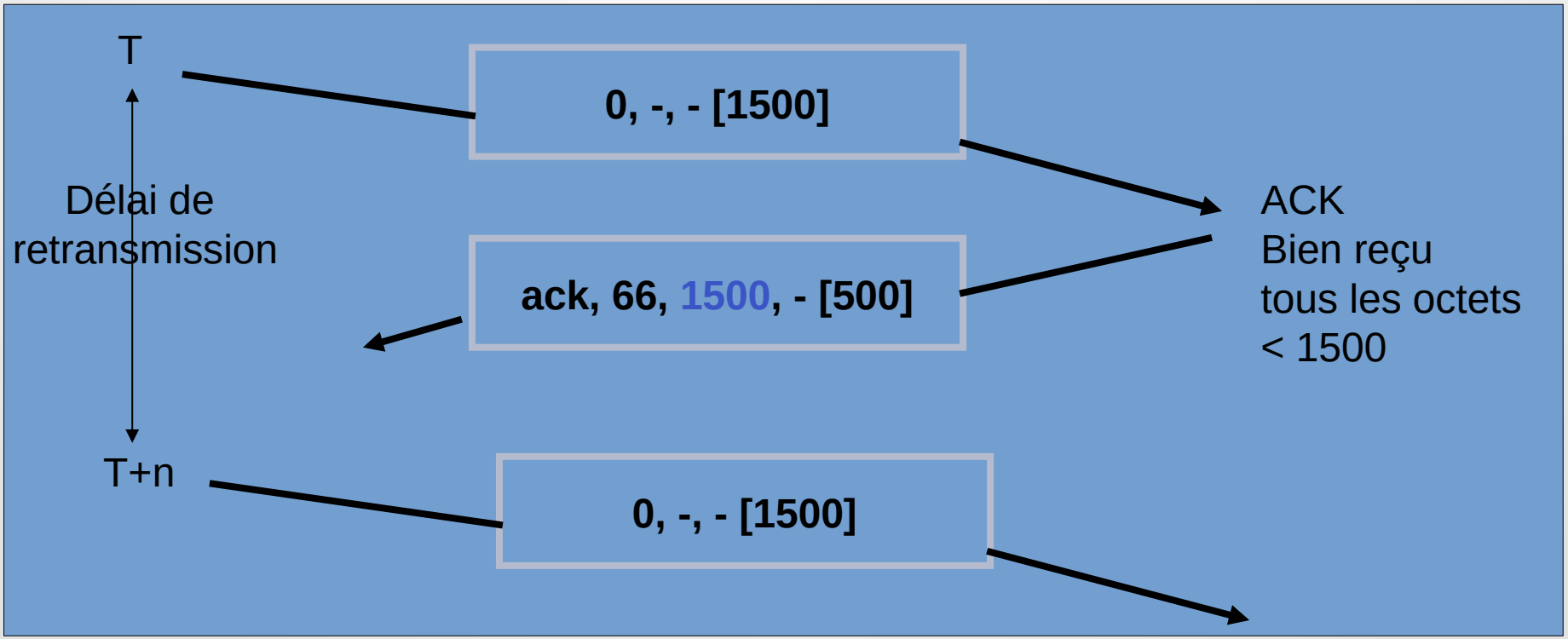
Protocole TCP

- Exemple Seq/Ack



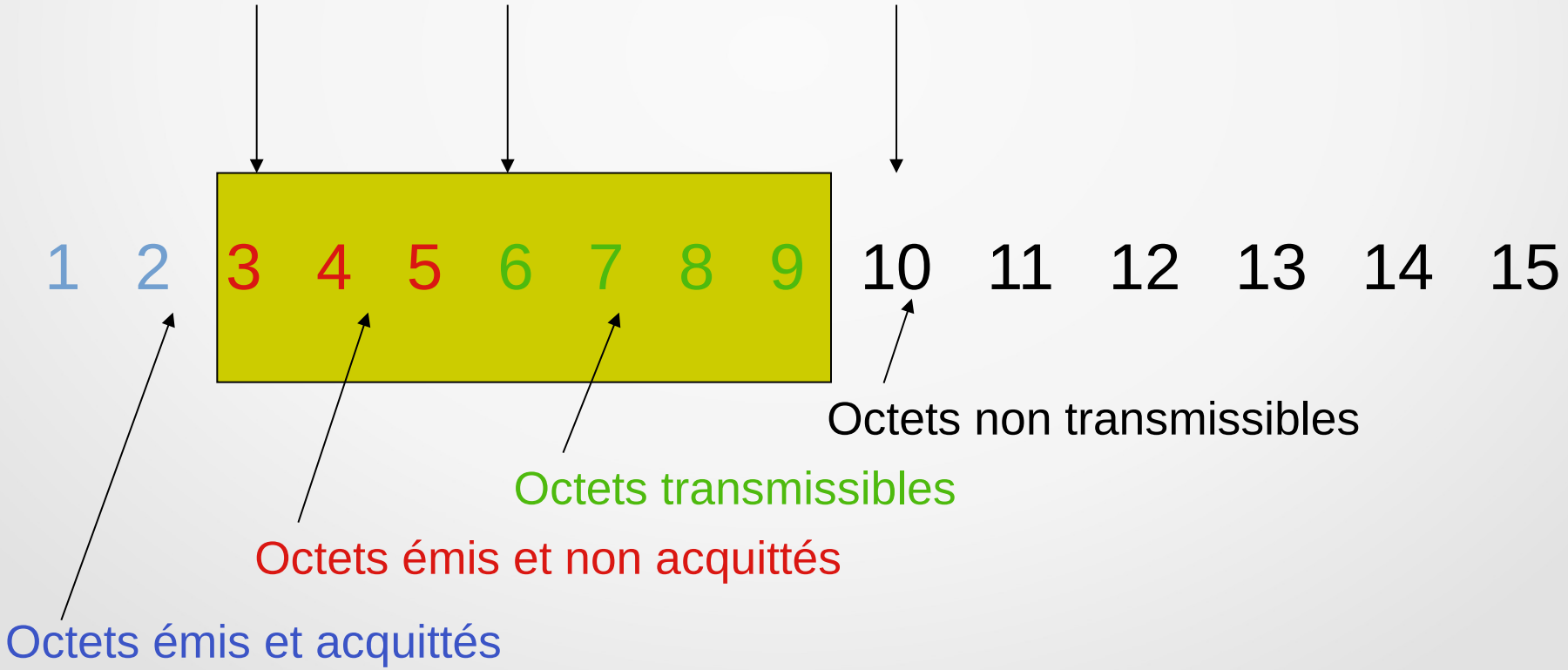
Protocole TCP

- Risques :
  - Perdre un segment ACK
  - Doit on tout renvoyer ?



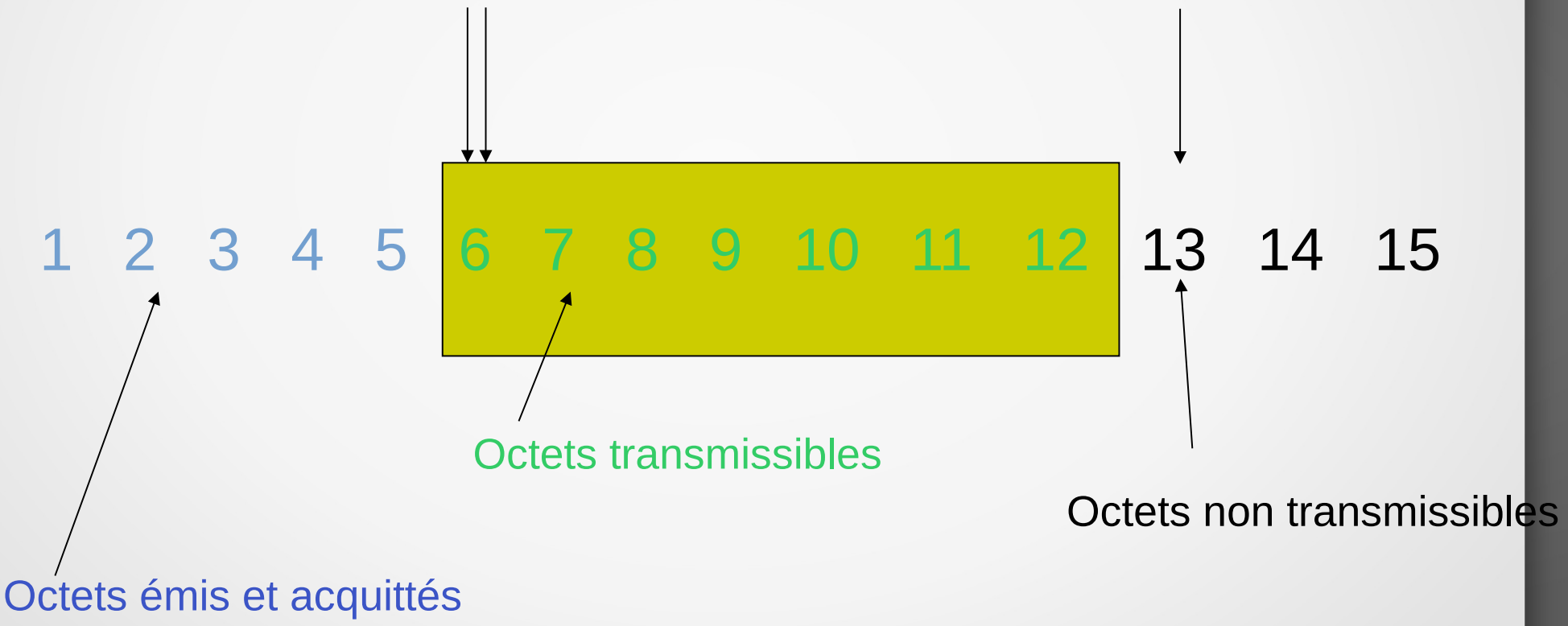
Protocole TCP

- Fenêtre d'anticipation (sliding window) :
  - Stocker dans une mémoire tampon tous les segments reçus et pas encore acquittés



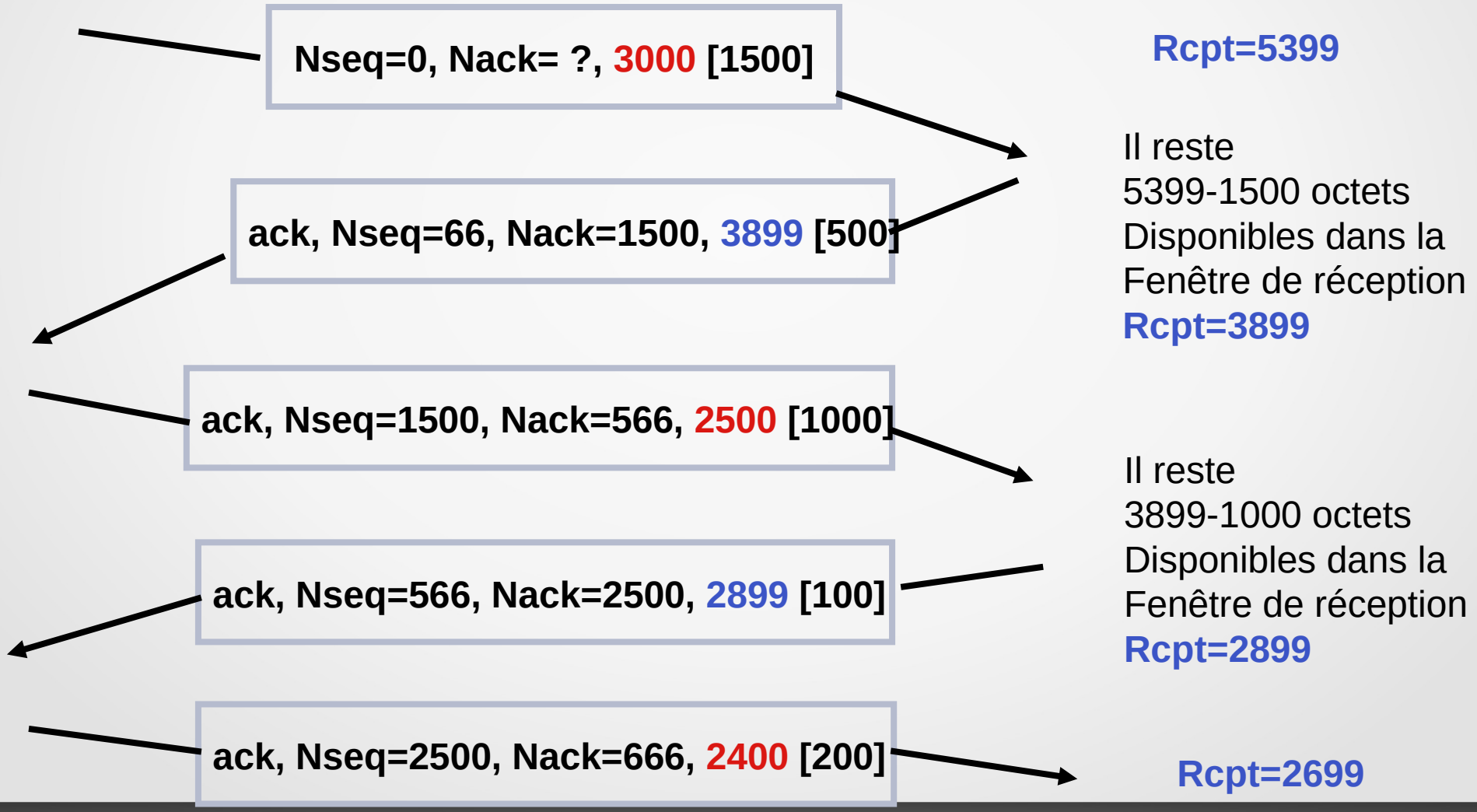
Protocole TCP

- Fenêtre glissante



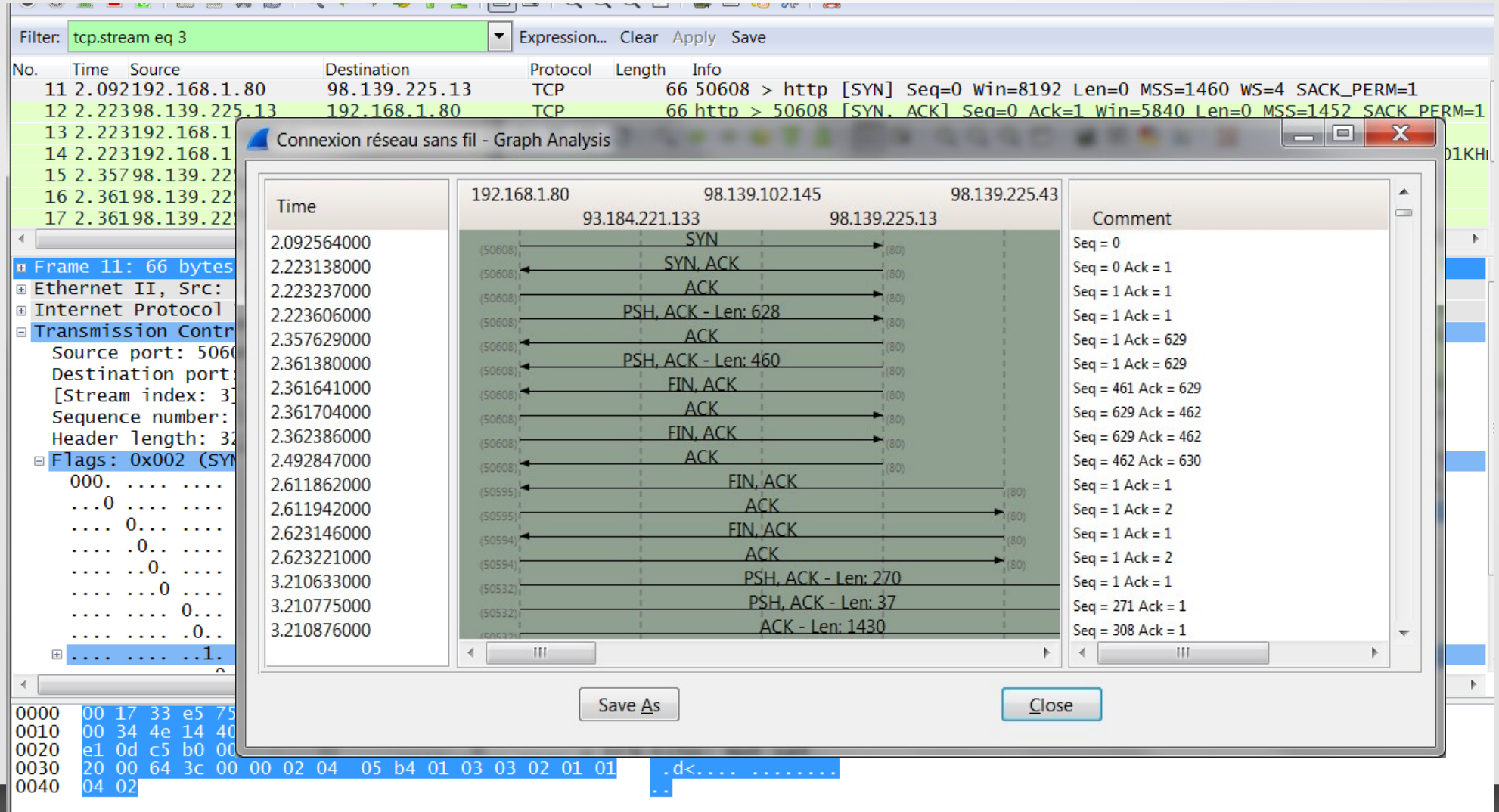
Protocole TCP

- Fenêtre de réception



Protocole TCP

- Capture de trame



Chapitre 3

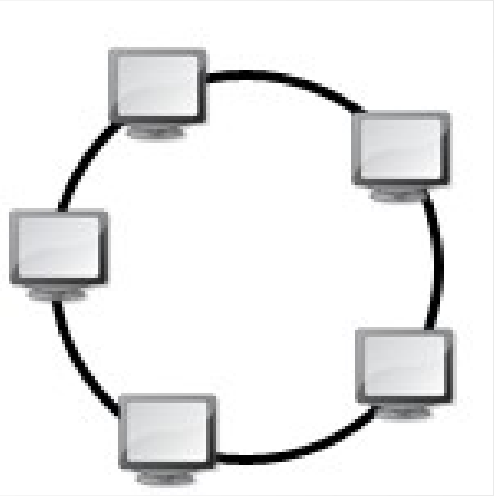
- Topologies de réseaux

Topologie

- 2 types de topologies
  - Physique
    - Façon dont sont reliés les ordinateurs
  - Logique
    - Façon dont ils se comportent sur le réseau
  - Les 2 peuvent être différentes
    - Ex : hub
      - Topologie physique en étoile
      - Topologie logique en bus

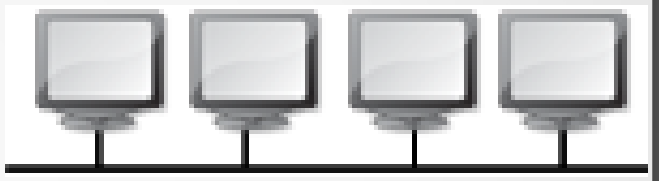
Différentes topologies réseau

- Anneau
  - Stations chainées entre elles par une liaison bi-directionnelle
  - Liaison type diffusion
  - Pas nécessairement en boucle
    - MAU (muti-station unit)
  - Chaque station traite le message et le ré-émet si elle n'est pas destinataire
  - 1 station défaillante et le réseau est bloqué
  - Avantage faible coût



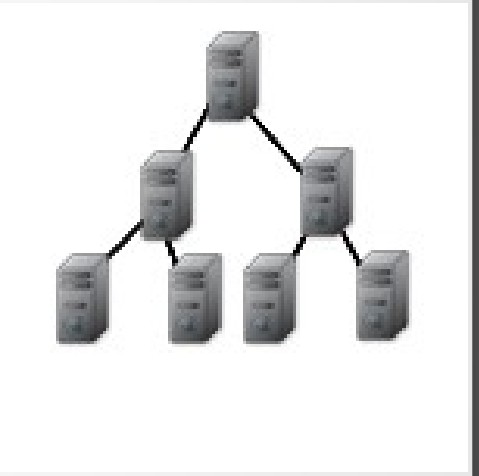
Différentes topologies réseau

- Topologie en bus
  - Stations reliées entre elles par un support unique
  - Liaison type difffusion
  - Avantage faible coût
  - Une station en panne ne compromet pas le réseau
  - Un lien défaillant paralyse le réseau



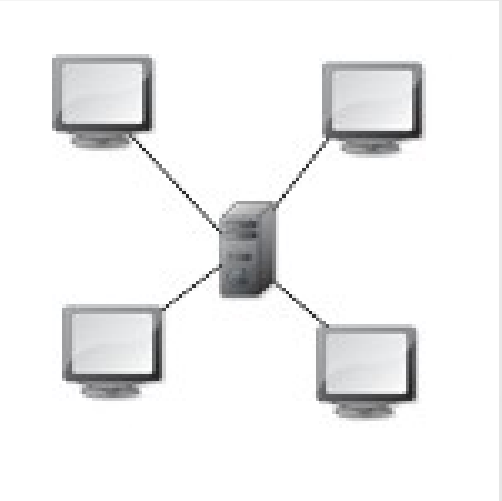
Différentes topologies réseau

- Topologie en arbre
  - Hiérarchie dans les nœuds du réseau
    - Division en niveaux
  - Faiblesse sur le nœud au sommet
    - Isole le réseau en 2 parties
  - Liaison type point à point
  - Exemple à l'IUT
    - Switch de salle
    - Switch d'étage
    - Switch salle serveurs



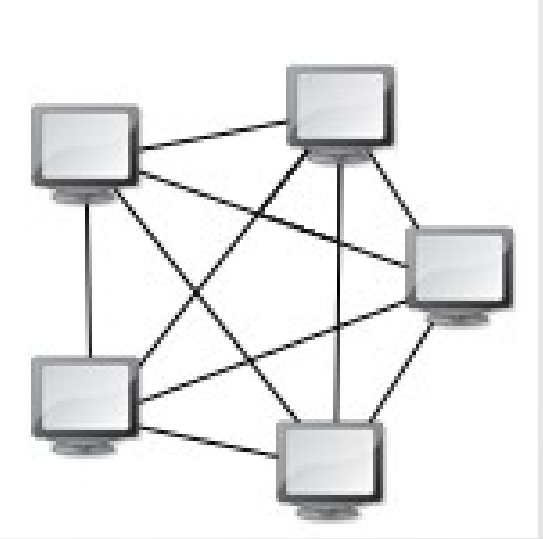
Différentes topologies réseau

- Topologie en étoile
  - Structure la plus courante maintenant
  - Utilise concentrateur
    - Exemple : switch
  - Liaison type point à point
  - Point faible le nœud



Différentes topologies réseau

- Topologie maillée
  - Très bonne tolérance aux pannes
  - Coût élevé
  - Liaison type point à point
  - Présent dans les réseaux de distribution (internet) et les interconnexion de routeurs wifi.
  - On parle de « mesh »



Module M1104

FIN