Ecole Centrale de Lyon – Mastère MDSI
René CHALON

Réseaux informatiques
~
Couche Réseau - IP - routage

1- Couche Réseau

2- Adressage IP

3- Protocole IP

4- Routage

5- IPv6

Version du 9/10/2012

Couche Réseau

- Fonctions de la couche réseau:
  - communication entre systèmes à travers un réseau:
    - ➔ transfert de paquets de bout en bout
  - adressage:
    - ➔ identifier de manière non-ambiguë les équipements
  - routage et acheminement:
    - ➔ transmettre les paquets de données à travers le réseau au destinataire en calculant au préalable une route (de préférence optimale)
  - indépendance par rapport aux supports de transmission
- Exemples:
  - IP (Internet Protocol) : version 4 (IPv4) et version 6 (IPv6)
  - X25 (norme ISO)
  - ATM peut aussi être vu comme un protocole de niveau 3

7	*Application*
6	*Présentation*
5	*Session*
4	*Transport*
3	**Réseau**
2	*Liaison de données*
1	*Physique*

Interconnexion au niveau réseau

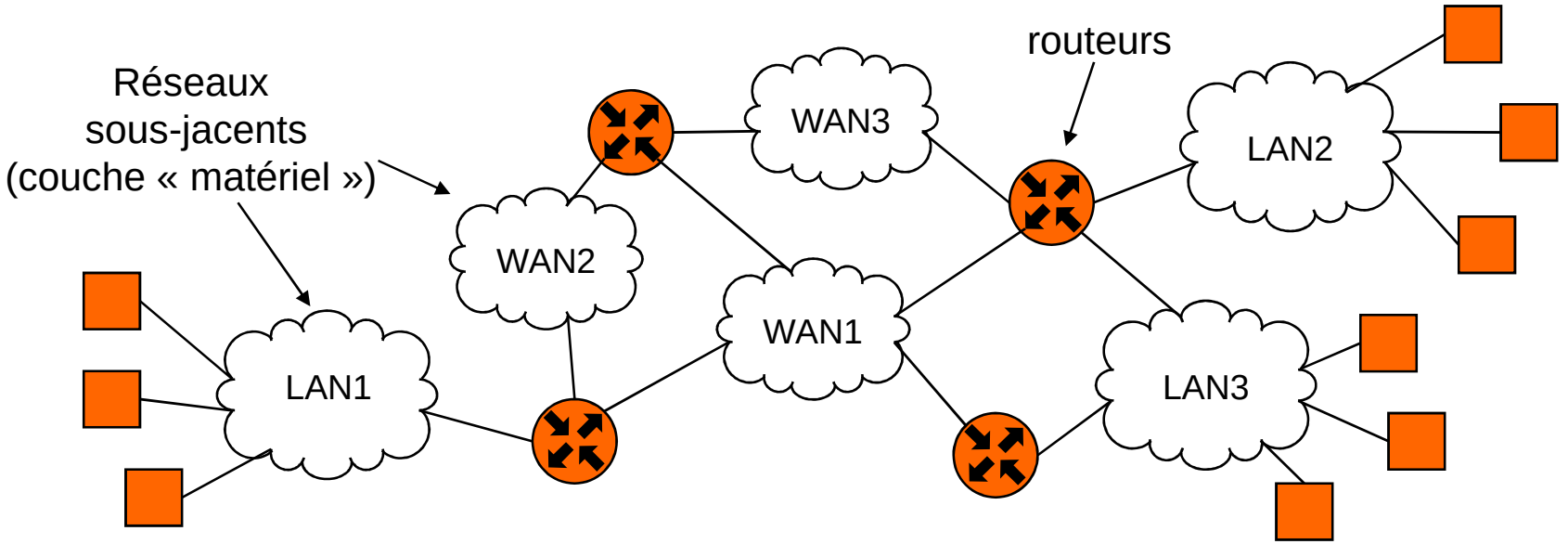
- Deux considérations :
  - Une seule technologie de réseau ne peut satisfaire les besoins de tous les utilisateurs:
    - LAN rapides mais distances faibles,
    - WAN plus lents,
    - Satellites pour la multi-diffusion,
    - Réseaux sans-fils...
  - Les utilisateurs souhaitent un moyen d'interconnexion universel:
    - ne pas être limité aux bornes physiques du réseau
    - ne pas utiliser des applications spécifiques au réseau utilisé
- Solution d'un internet:
  - interconnecter les réseaux hétérogènes de manière transparente aux applications en masquant les détails des réseaux
  - Système de communication abstrait

Propriétés d'un internet

- Cacher l'architecture sous-jacente :
  - ➔ les applications ne doivent pas connaître les détails des connexions physiques
- Indépendance vis-à-vis des réseaux sous-jacents :
  - ➔ Les actions pour établir une communication restent indépendantes des réseaux sous-jacents et du type d'ordinateur destinataire
- Ne pas imposer de topologie particulière de réseau :
  - ➔ l'ajout d'un nouveau réseau ne doit pas impliquer sa connexion à un ordinateur central ou sa connexion à tous les réseaux existants
- Possibilité d'envoyer des informations à travers des réseaux intermédiaires
  - ➔ notion de réseaux relais
- Tous les ordinateurs doivent partager un ensemble d'identificateurs qui est universel:
  - ➔ notions d'adresses et/ou de noms

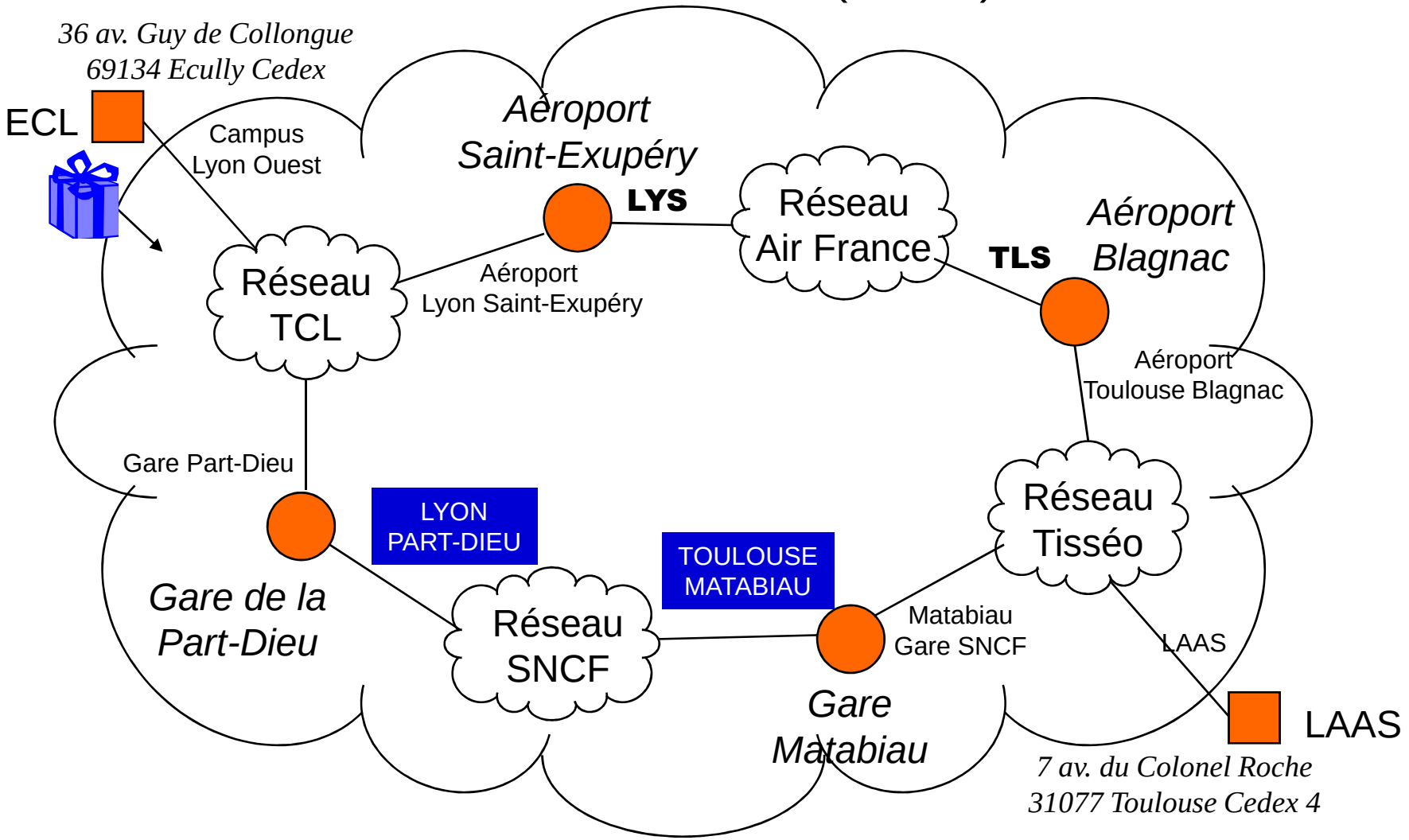
Architecture d'un internet

- Interconnexion de plusieurs réseaux différents
  - par un routeur
    - Routeur IP [Internet router]
    - ancienne appellation : passerelle IP [Internet gateway]
  - Connecté à chaque réseau qu'il relie
  - Mode de fonctionnement « store and forward »



Métaphore de transport (1/3)

- Emmener un colis de l'ECL au LAAS (CNRS) à Toulouse :



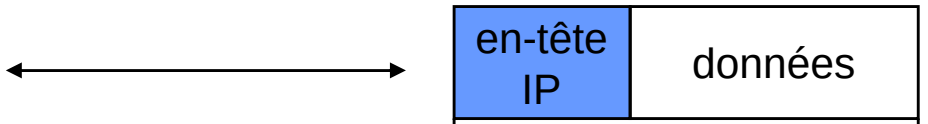
Métaphore de transport (2/3)

Domaine source:		Domaine cible:
		Au niveau 3:
ECL, LAAS	⟷	équipement terminal
gare, aéroport	⟷	équipement intermédiaire, routeur
colis, paquet	⟷	paquet
		Au niveau 2:
réseaux TCL et Tisséo	⟷	réseaux locaux
réseaux SNCF et Air France	⟷	réseaux longue distance
nom arrêt bus/métro, nom de gare, code d'aéroport	⟷	adresses de niveau 2
bus, trains, avions	⟷	trames
correspondance (entre 2 trains, 2 bus, ou 2 avions)	⟷	pont ou commutateur

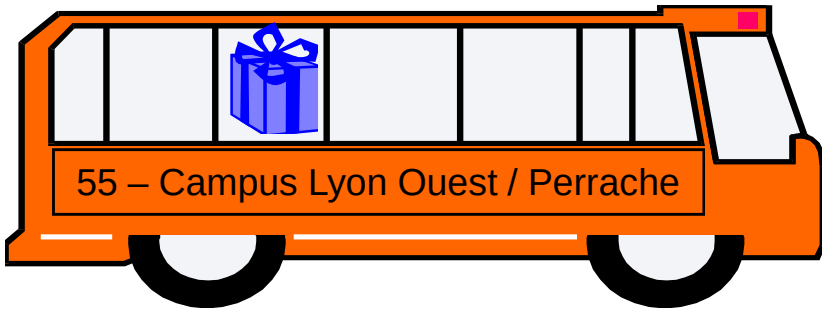
Métaphore de transport (3/3)

- Colis :

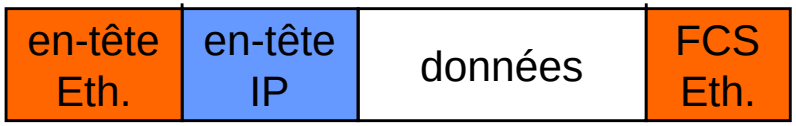
- Paquet IP :



- Transport du colis dans un bus :

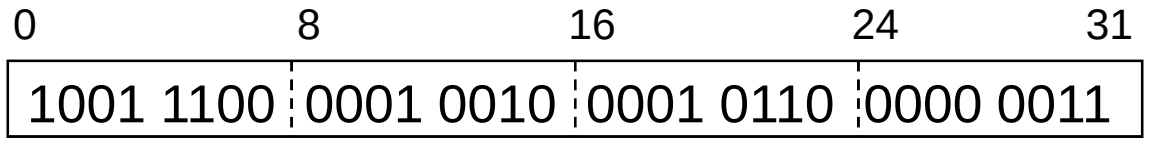


- encapsulation du paquet IP dans une trame Ethernet :



Adressage IPv4

- Adresses de 32 bits (4 octets)



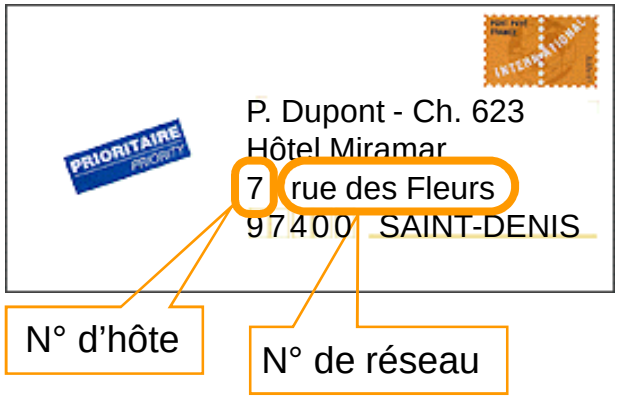
  - Notation « décimale pointée »
    - Représentation en base 10 des 4 octets : **`www.xxx.yyy.zzz`**
    - Exemple : **`156.18.22.3 =`** 1001 1100 0001 0010 0001 0110 0000 0011
- Adresses unicast (adresses entre 1.0.0.0 et 223.255.255.255)
  - Attribuées de manière unique à un équipement dans l'Internet
  - Historiquement 3 classes d'adresses : A, B, C (obsolète)
- Adresses multicast et broadcast
  - Permettent d'envoyer un paquet
    - Soit à un groupe de machines (multicast) = 224.0.0.0 à 239.255.255.255
    - Soit à toutes les machines d'un réseau (broadcast) = 255.255.255.255
  - Adresses utilisables que comme adresse de destination

Structure des adresses IP

- 2 parties dans chaque adresse unicast :
  - Numéro du réseau
  - Numéro d'hôte dans le réseau local
  - Limite entre les 2 zones variable
    → donnée par le masque

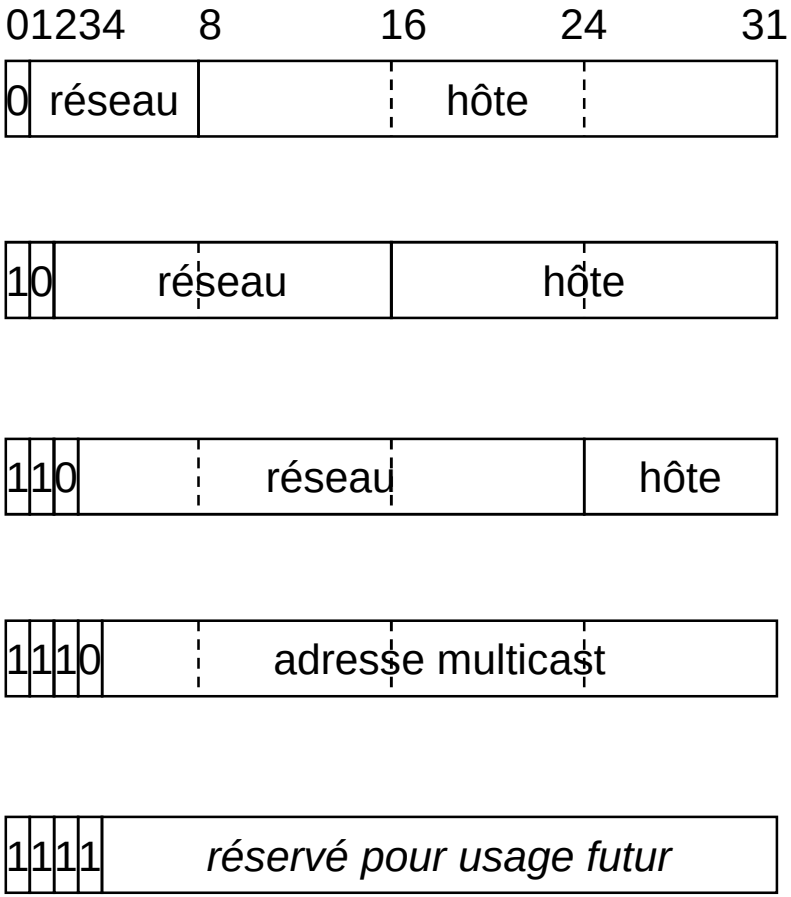
Adresse:	n° réseau	n° hôte
Masque:	111 … … 111	000…000

Métaphore postale



  - Autant de bits à 1 que de bits codant le n° de réseau dans l'adresse
  - Notations du masque :
    - Ancienne : 255.255.255.0 pour 1111 1111 1111 1111 1111 1111 0000 0000
    - Moderne : « /n » (n étant le nb de bits à 1)
- Exemples :
  - 156.18.22.3/24 signifie :
    - N° réseau : 1001 1100 0001 0010 0001 0110
    - N° machine : 0000 0011
  - 156.18.0.0/16 : réseau d'adresses 156.18.0.0 à 156.18.255.255

Historique : classes d'adressage



- Classe A
  - réseaux 1.0.0.0 à 126.0.0.0
  - 126 réseaux de 16 millions d'hôtes
- Classe B
  - réseaux 128.0.0.0 à 191.255.0.0
  - 16384 réseaux de 65534 hôtes
- Classe C
  - réseaux 192.0.0.0 à 223.255.255.0
  - 2097152 réseaux de 254 hôtes
- Classe D (adresses multicast)
  - Adresses de 224.0.0.0 à 239.255.255.255
  - Plus de 268 millions de groupes
- Classe E (adresses réservées)
  - 268 millions d'adresses réservées

- Les classes A, B, C sont déclarées obsolètes depuis 1993 (RFC 1519)

Adresses particulières

- Adresse d'un réseau
  - Tous les bits de la partie hôte à 0
    - ➔ Exemple: 156.18.0.0 (réseau de l'Ecole Centrale)

N° réseau	Tout à 0

- Adresse de diffusion dirigée
  - Tous les bits de la partie hôte à 1
    - ➔ Exemple: 156.18.255.255 (tous les hôtes de l'ECL)
  - permet de s'adresser à tous les hôtes d'un réseau
  - Obsolète depuis 1999 (RFC 2644) mais adresse réservée

N° réseau	Tout à 1

- Adresse de diffusion "locale"
  - 255.255.255.255
  - Ne doit pas sortir du réseau local

Tout à 1

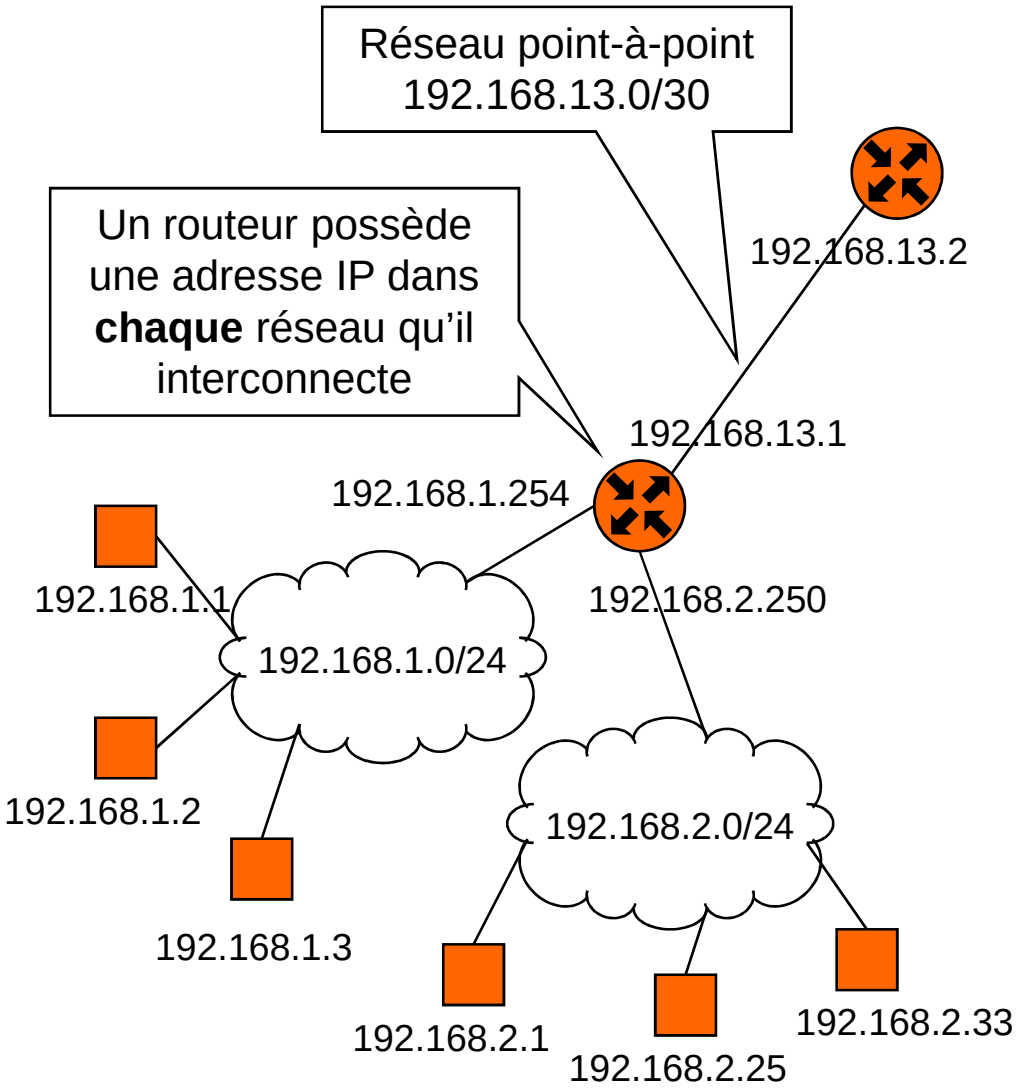
- Rebouclage
  - Toutes les adresses du bloc 127.0.0.0/8 (ex classe A)
  - Usage local à la machine : ne doit jamais apparaître sur un réseau

127	Nb quelconque

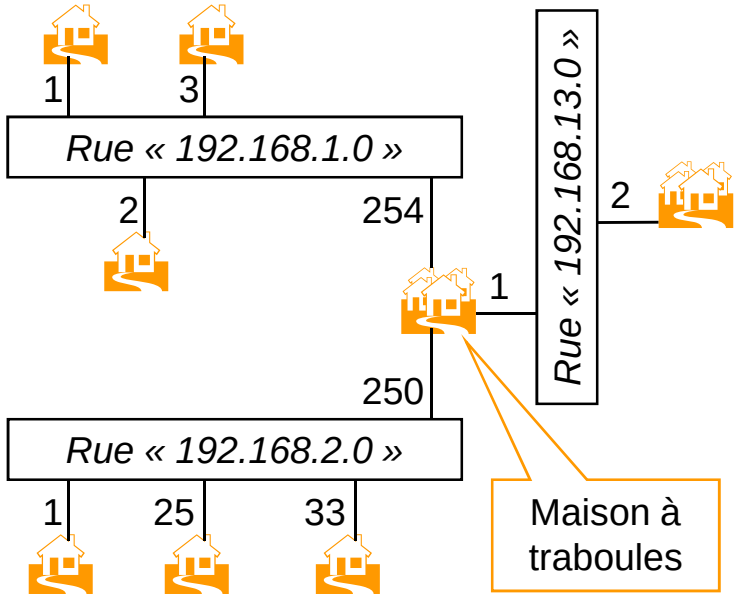
- NB: le bloc 0.0.0.0/8 est réservée
  - Adresses provisoires (protocole RARP)

0	Nb quelconque

Exemple



*Métaphore **lyonnaise***



Ville « spéciale » sans carrefour dont les rues ne sont interconnectés que par des « traboules » (maisons qui ont plusieurs adresses)

Attribution des adresses (1/2)

- ICANN [Internet Corporation for Assigned Names and Numbers] est l'organisme chargé :
  - ➔ de l'allocation des adresses IP
  - ➔ de la gestion des noms de domaine
  - ➔ de l'homologation des protocoles et de leurs paramètres
  - ➔ de la gestion des serveurs racines
  - Organisme international indépendant et géré par la communauté Internet (membres nommés et membres élus)
  - ➔ plus d'info sur: **`www.icann.org`**
- Autrefois, les adresses étaient gérées par l'IANA [Internet Assigned Number Authority] : **`www.iana.org`**
  - organisme dépendant du gouvernement américain
  - Règle d'attribution fixée par le RFC 2050

Attribution des adresses (2/2)

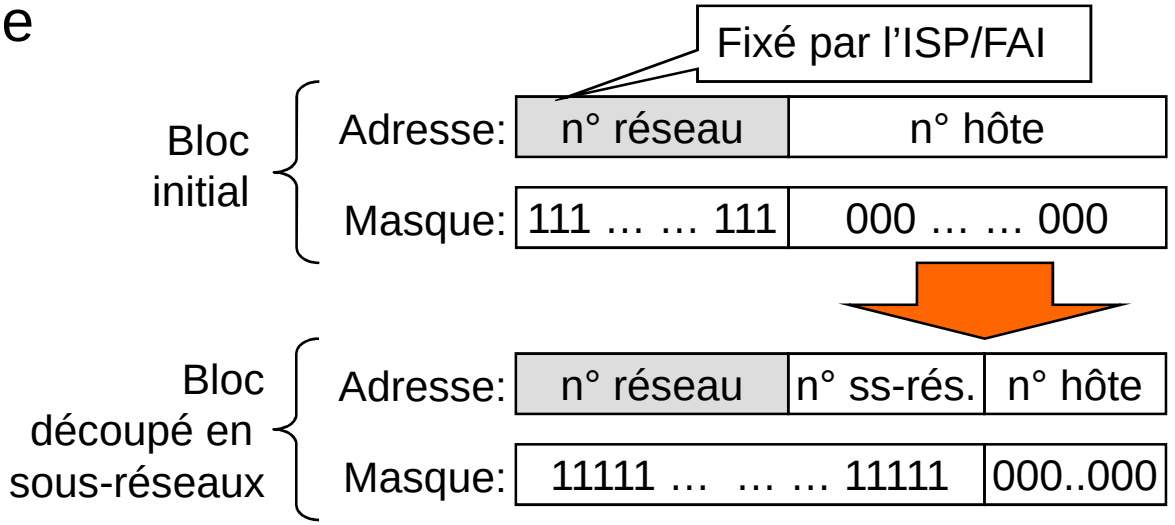
- L'attribution est déléguée à un RIR [Regional Internet Registry] :
  - APNIC [Asia Pacific Network Information Center] : **`www.apnic.net`**
  - ARIN [American Registry for Internet Numbers] : **`www.arin.net`**
  - LACNIC (Regional Latin-American and Caribbean IP Address Registry) : **`lacnic.net`**
  - RIPE NCC [Réseaux IP européens] : **`www.ripe.net`**
  - AfriNIC (Afrique/océan indien) : **`www.afrinic.net`**

Depuis avril 2005 !
Avant : - Nord : RIPE
- Sud : ARIN

- Délégation en cascade de blocs d'adresses:
  - National Internet Registry (NIR) (région Asie-Pacifique seulement)
  - Local Internet Registry (LIR) ou Fournisseurs d'accès (FAI) [Internet Service Provider – ISP]
  - Utilisateurs finaux: entreprises, individus
- Attribution des adresses des machines dans le réseau :
  - par l'administrateur du réseau

Sous-réseaux IP

- Découpage d'un réseau en entités plus petites :
  - création de sous-réseaux par l'administrateur du site
  - les sous-réseaux ne sont pas visibles à l'extérieur du site
- Utilisation du masque
  - On « vole » des bits sur la partie « n° hôte » pour numéroter les sous-réseaux



  - Exemple : Ecole Centrale = bloc 156.18.0.0/16
    - 156.18.22.0/24 : sous-réseau du CRI
    - 156.18.37.0/24 : sous-réseau du département MI
    - etc…

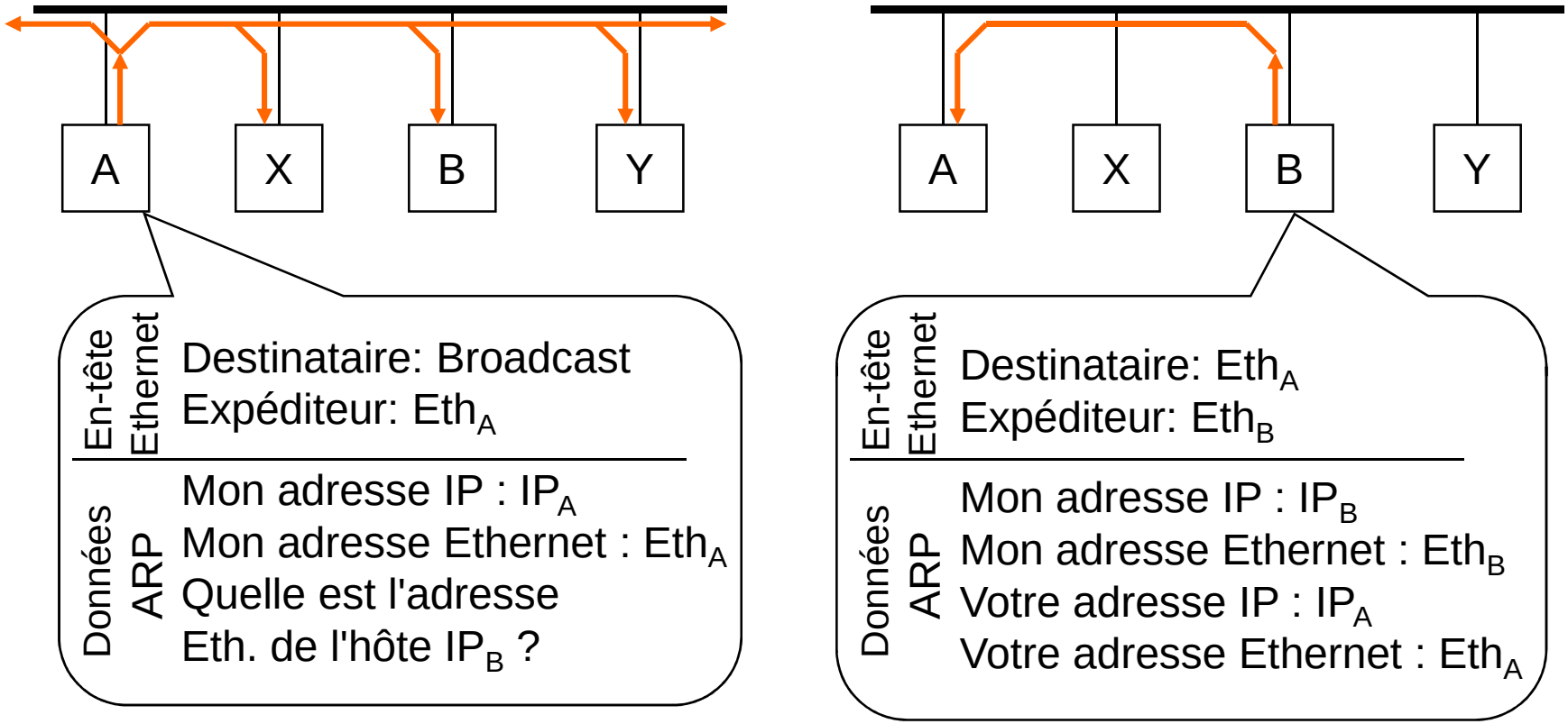
Correspondance adresse IP - adresse réseau

- Les adresses IP sont indépendantes des adresses des réseaux sous-jacents
- Mise en relation directe :
  - table statique de correspondance
    - Par exemple avec réseaux X25 :
      table de correspondance adresse IP⇔ adresse X121 (RFC 877)
  - la partie "hôte" est l'adresse dans le réseau sous-jacent
    - Impossible avec IPv4 et Ethernet car 6 octets ne peuvent entrer dans 1 2 ou 3 octets max !!!
    - Utilisable avec IPv6 où 8 octets sont réservés pour l'adresse locale sur 16 octets
- Mise en relation dynamique :
  - protocole ARP (RFC 826)
  - découverte automatique des adresses IP des machines dans le même réseau local

Protocole ARP sur Ethernet (1/2)

- Principe de ARP sur réseau Ethernet :

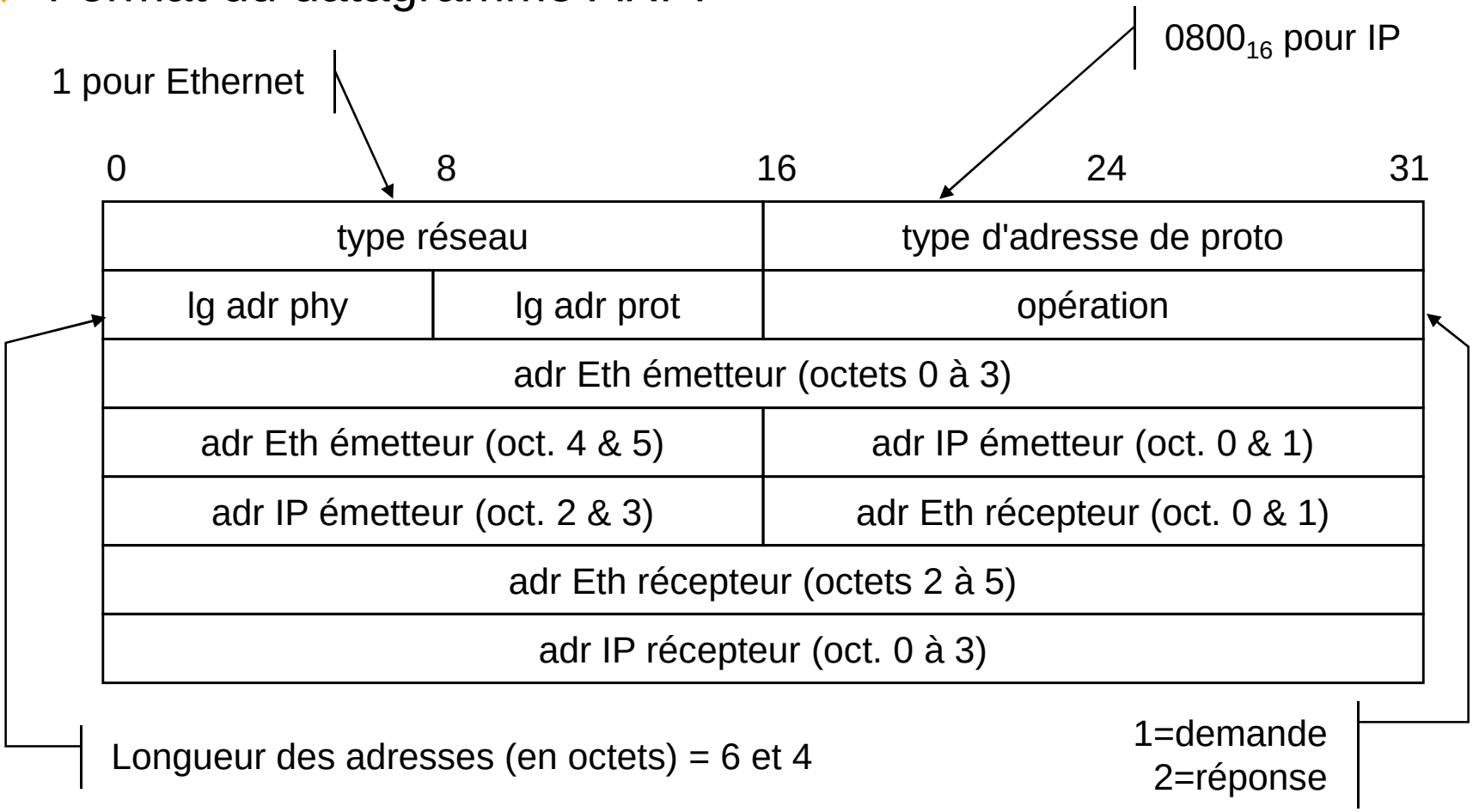
Question diffusée (broadcast) :



Réponse directe :

Protocole ARP sur Ethernet (2/2)

- Utilise des trames Ethernet II, "type"=0806_{16}
- Format du datagramme ARP:



Adressage des équipements

- Les adresses font références à des points d'accès au réseau:
  - Un ordinateur connecté à plusieurs réseaux possède plusieurs adresses, une par réseau :
    - Obligatoire et essentiel pour un routeur
    - Présente des inconvénients pour les autres machines (serveurs, …)

- L'adresse dépend du réseau de connexion:
  - Un ordinateur doit changer d'adresse s'il change de réseau
  - Administration lourde pour les ordinateurs mobiles
  - Solution: DHCP [Dynamic Host Configuration Protocol]

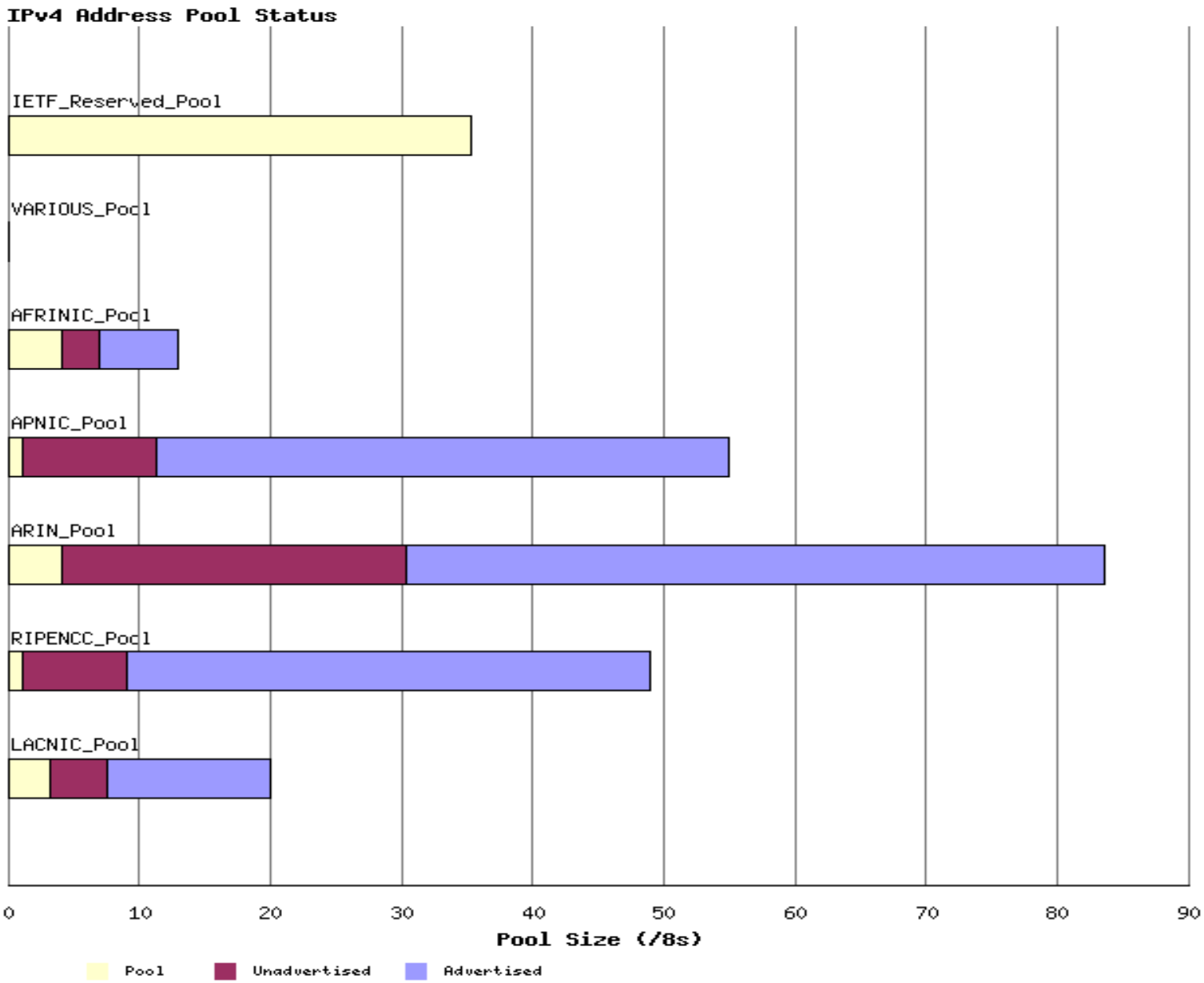
DHCP

- DHCP = Dynamic Host Configuration Protocol
  - remplace RARP [Reverse ARP]
  - sur-ensemble de BOOTP [Bootstrap Protocol]
- But: donner à une machine ses paramètres réseau au démarrage:
  - attribution automatique de son adresse IP et du masque de réseau
  - adresse du routeur par défaut
  - nom de la machine, adresses des serveurs DNS, etc…
- Fonctionnement:
  - Un serveur DHCP gère des listes d'adresses IP : il loue les adresses pour un certain temps (bail) aux machines clientes
  - les clients doivent renouveler leur bail avant l'expiration
  - par ce mécanisme il peut y avoir plus de machines que d'adresses disponibles (en supposant qu'elles ne sont pas toutes présentes)

Pénurie d'adresses IPv4

- Pas assez d'adresses disponibles:
  - Pourtant 4 octets ➔ 4,43 milliards d'adresses
  - Mais seul 86 % de l'espace d'adressage est réellement utilisable
    - ➔ Réseaux 0.0.0.0/8 et 127.0.0.0/8 perdus et classe E réservée
    - ➔ Gaspillage dû à un découpage en classes mal conçu à l'origine : classes A et B trop grandes, classes C trop petites
- Demande de plus en plus forte :
  - Le nombre de machines qui se connectent à l'Internet croit exponentiellement : ordinateurs fixes, ordinateurs portables, PDA, téléphones mobiles et bientôt réfrigérateur, voitures, etc.
- La fin est là :
  - L'ICANN a distribué tous les blocs disponibles aux RIR (fév. 2011)
  - Le 14/09/2012, le RIRE RIPE-NCC alloue le dernier bloc /8 disponible (allocation en blocs de /22 !)
  - Plus d'adresses vers 2012-2014 !!!

Plus d'infos :
http://www.potaroo.net/tools/ipv4



Source: http://www.potaroo.net/tools/ipv4/ (octobre 2012)

Solutions pour gérer la pénurie

- Limiter le gaspillage des adresses :
  - Depuis1993 : suppression de la notion de classe et utilisation de CIDR [Classless Inter-Domain Routing] (RFC 1519)
- Allouer moins d'adresses que de machines dans un réseau et partager ce lot d'adresses dynamiquement :
  - Solution: utilisation d'adresses privées en interne et ré-écriture avec NAT [Network Address Translation] pour sortir sur l'Internet
- Solution à long terme : IPv6
  - augmenter la taille des adresses : 16 octets, soit 340.10^{36} adresses possibles !!!

CIDR

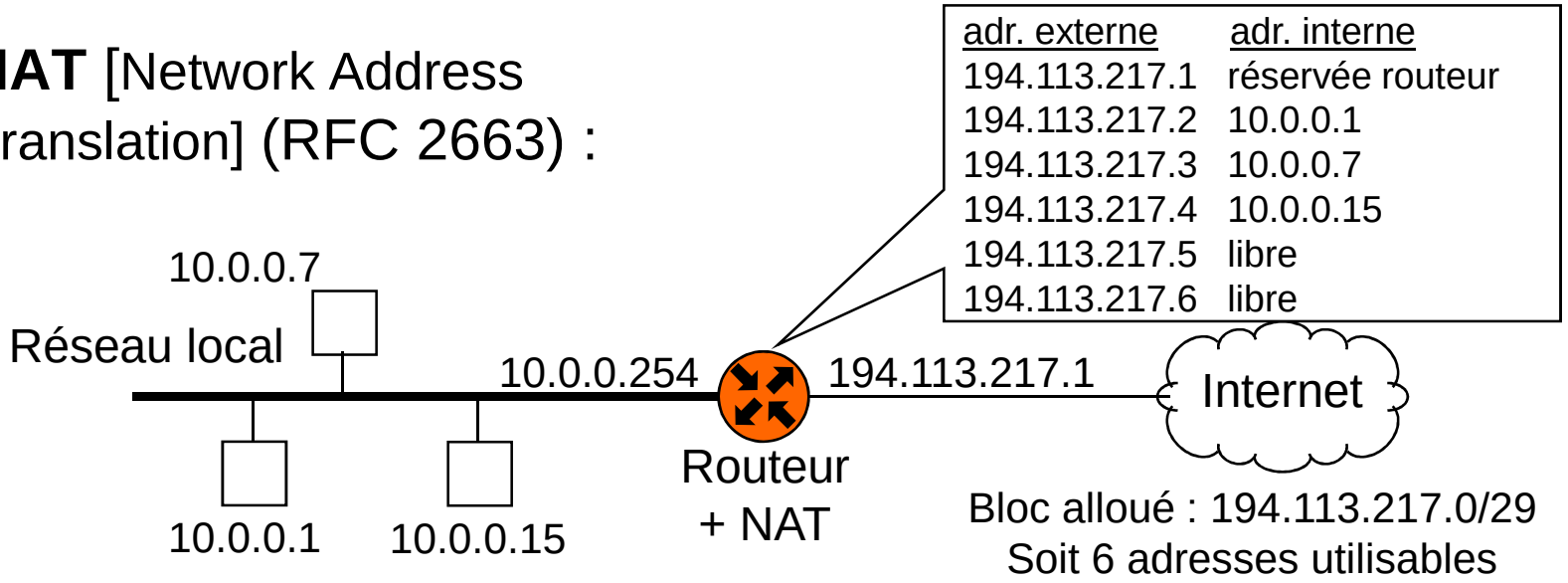
- CIDR = Classless Inter-Domain Routing (RFC 1519 / RFC 4632) :
  - Depuis 1993 : le découpage en classe A, B, C est supprimé
  - Toute réseau est exprimée par un numéro et un masque
    - Exemple : 156.18.0.0/16
- Avantage :
  - Meilleure utilisation des adresses :
    - les blocs d'adresses sont alloués avec une taille adaptée au nombre de machines du réseau connecté
  - Simplification des tables de routage :
    - Politique d'allocation des blocs consécutifs par un même RIR
    - Exemple : 194.0.0.0/8 est alloué par RIPE
      - Un routeur en Amérique place une seule ligne dans sa table de routage
- Inconvénient :
  - Complique la tâche des routeurs

Adresses privées – NAT – NAPT (1/2)

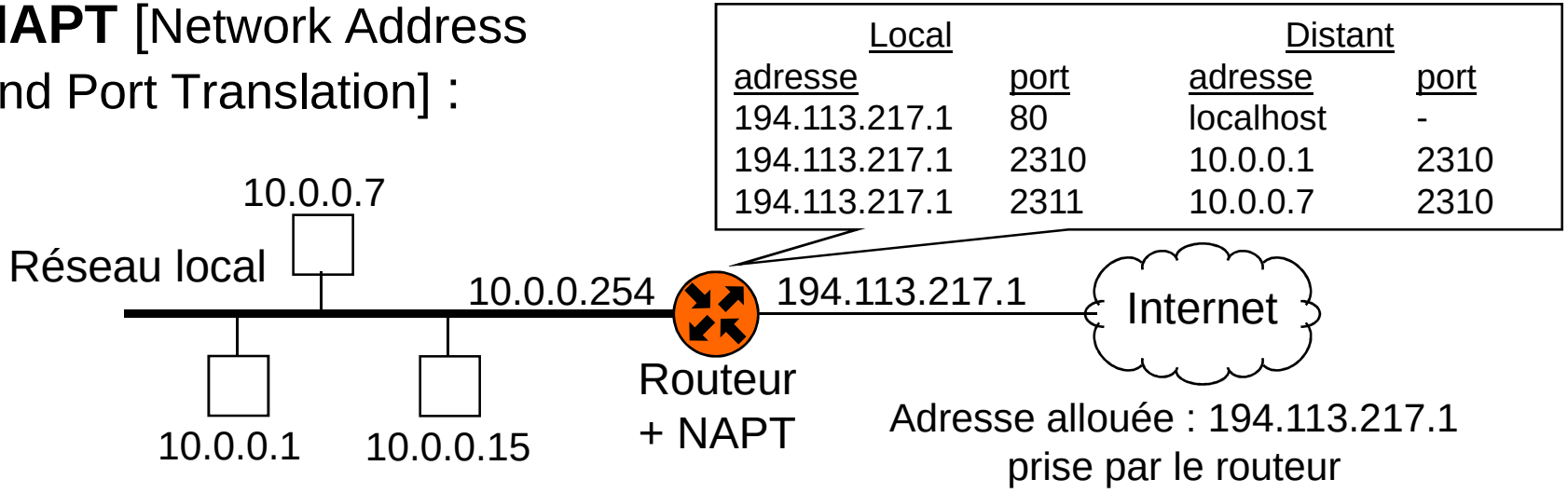
- Adresses IP réservées à un usage privé (RFC 1918):
  - 3 blocs définis :
    - 10.0.0.0/8 (10.0.0.0 à 10.255.255.255), soit 1 ex-classe A
    - 172.16.0.0/12 (172.16.0.0 à 172.31.255.255) soit 16 ex-classes B
    - 192.168.0.0/16 (192.168.0.0 à 192.168.255.255) soit 255 ex-classes C
- Ces adresses ne doivent pas apparaître sur l'Internet :
  - Traduction « à la volée » des adresses privées dans chaque paquet en une adresse publique avant de sortir sur l'Internet :
  - 2 possibilités :
    - Une adresse publique pour chaque adresse privée : NAT « pur » [Network Address Translation] (RFC 2663)
    - Une seule adresse publique partagée par toutes les adresses privées: NAPT [Network Address and Port Translation] ; il faut aussi gérer la traduction des numéros de port TCP et UDP pour éviter d'éventuels conflits entre machines (pas de RFC !!!). Appelé aussi « NAT44 ».

Adresses privées – NAT – NAPT (2/2)

- **NAT** [Network Address Translation] (RFC 2663) :



- **NAPT** [Network Address and Port Translation] :

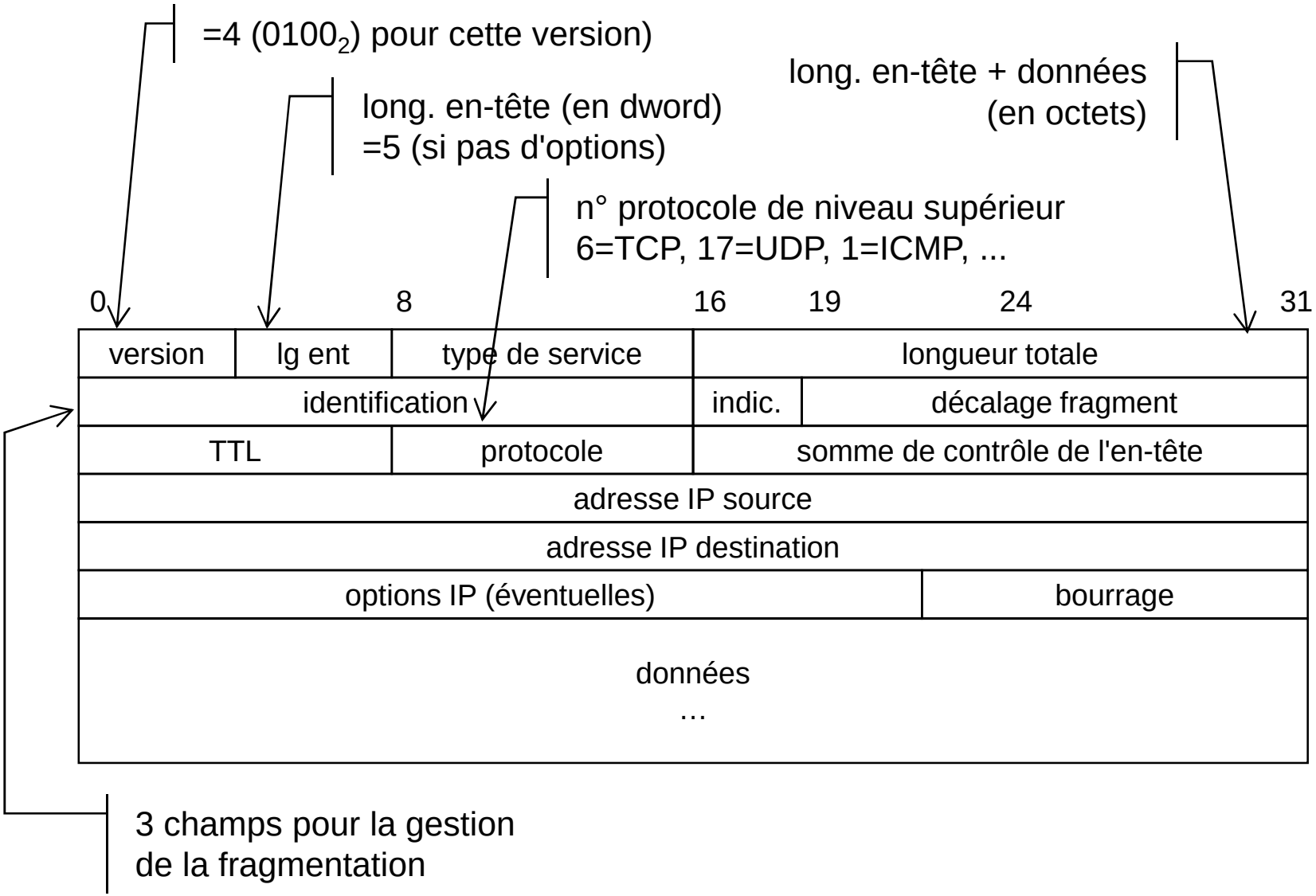


3

Protocole IP

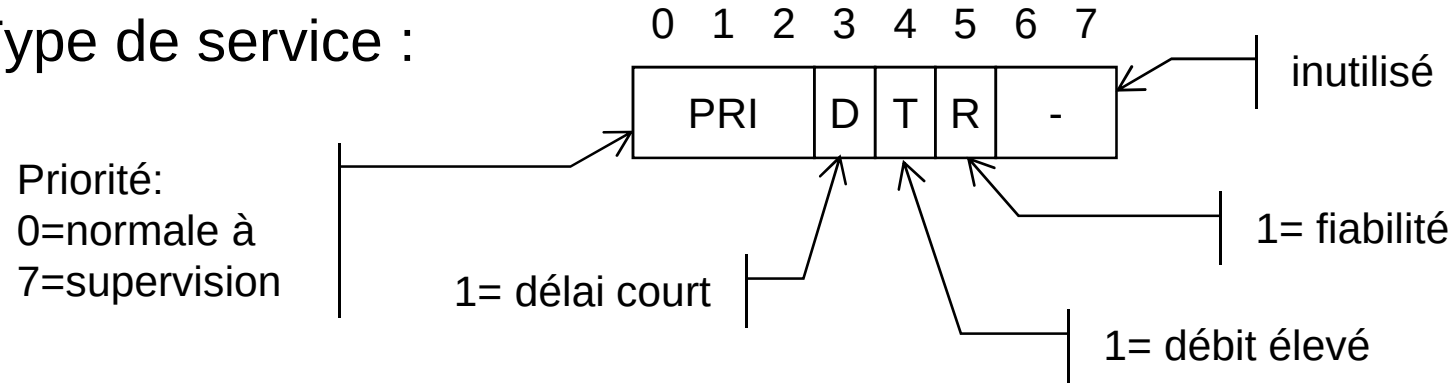
- IP assure les fonctionnalités de la couche 3 de l'OSI:
  - Il permet le transfert de paquets (appelés datagrammes) entre deux ordinateurs (éventuellement en traversant plusieurs réseaux)
- Caractéristiques :
  - Mode **non-connecté**
  - Transfert **non fiable** des données (la fiabilité des réseaux traversés) IP fait au mieux (*best effort protocol*)
  - Taille maximale d'un datagramme= 64 Ko; il peuvent être fragmentés en fonction de la nature des réseaux traversés
- Services assurés :
  - adressage : tout équipement (ordinateur et routeur) dispose d'une ou plusieurs adresses
  - routage : acheminement des paquets entre deux sous-réseaux
- Version actuelle: version 4 (appelée IPv4) - RFC 791 (STD 5)

Datagramme IP (1/3)



Datagramme IP (2/3)

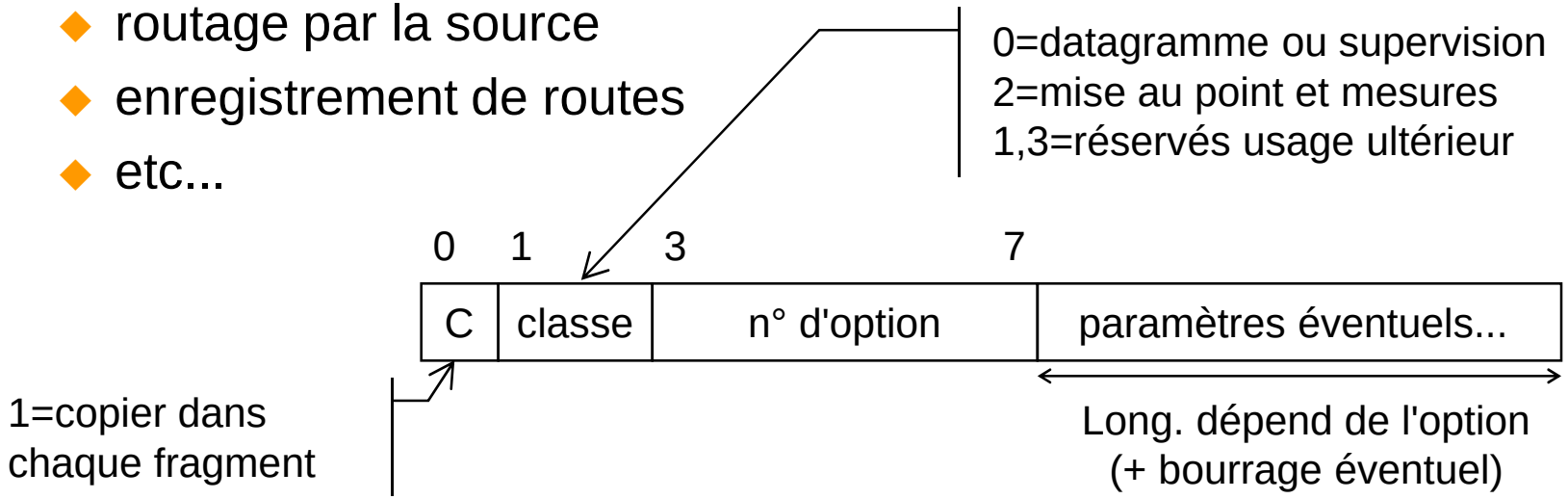
- Type de service :



  - la priorité permet, en cas de congestion, que les informations de contrôle du réseau puissent circuler (sans être victimes elles-mêmes de la congestion !)
  - D, T et R permettent aux routeurs de choisir le chemin qui correspond à ces critères (s'ils ont le choix !)
- Somme de contrôle:
  - complément à un de la somme de tous les mots de l'en-tête
  - ne vérifie que l'en-tête pas les données ! C'est aux couches supérieures de vérifier l'intégrité des données

Datagramme IP (3/3)

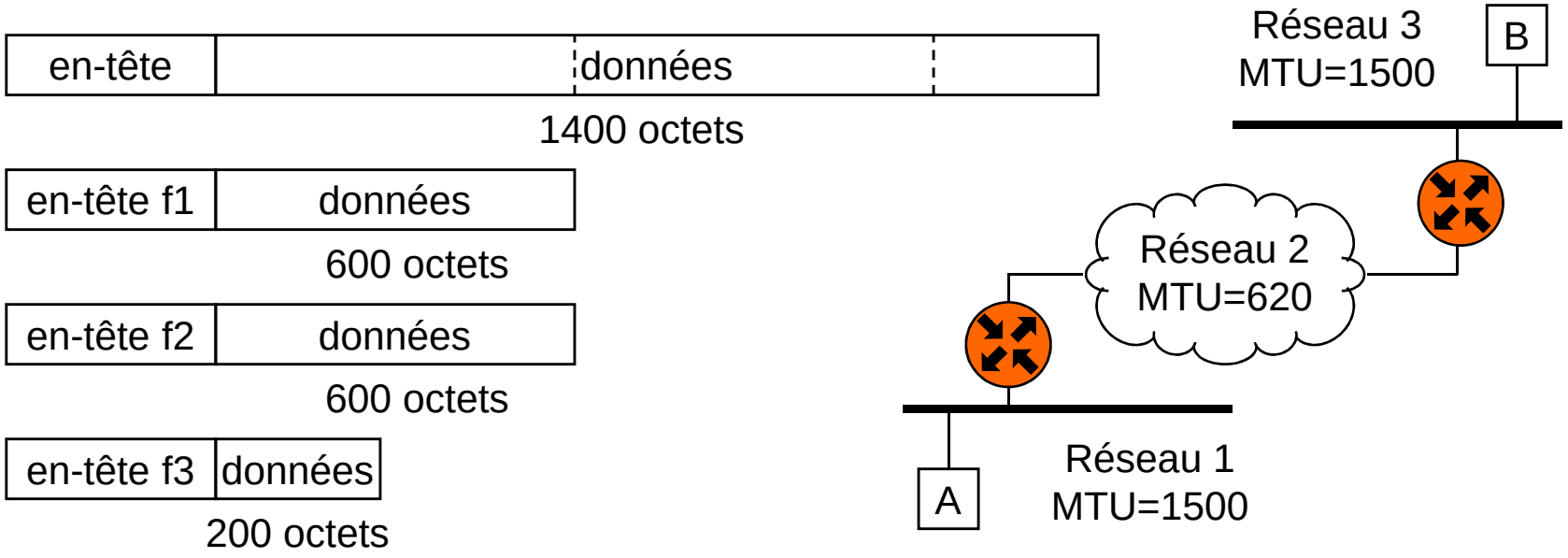
- TTL [Time to Live] = durée de vie
  - exprime en secondes, la durée maximale de transit d'un paquet
  - chaque routeur doit décrémenter la valeur (en pratique, le TTL compte donc le nombre de routeurs traversés)
  - le datagramme est détruit quand TTL=0
    - ➔ évite au datagramme de circuler indéfiniment en cas de boucle
- Options:
  - routage par la source
  - enregistrement de routes
  - etc...



Fragmentation des datagrammes IP (1/2)

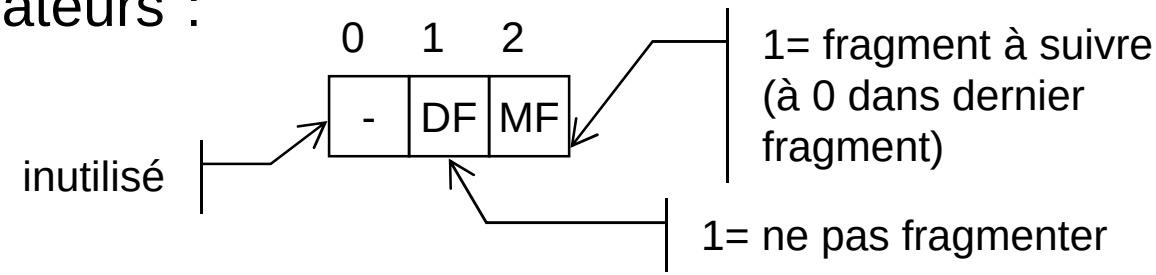
- **Problème**: un datagramme IP peut être plus grand que la taille maximale de trame admise par le réseau sous-jacent
  - Ethernet = 1500 octets
  - FDDI = 4470 octets

  } MTU [Maximum Transfer Unit] (exprimé en données utiles !)
- **Solution**: un routeur peut fragmenter un datagramme pour qu'il respecte le MTU du réseau sous-jacent



Fragmentation des datagrammes IP (2/2)

- Le champ "identification" contient un n° unique de datagramme
- Le champ "décalage fragment" contient la localisation du fragment par rapport au début du bloc initial de données
  - exprimé sous la forme d'un multiple de 8 octets
  - donc les fragments doivent être des multiples de huit !
- Champ "indicateurs":



- Ré-assemblage du datagramme initial :
  - utilisation du champs "identification"
  - les fragments sont ré-assemblés par l'ordinateur destinataire et non par les routeurs

Fonctions non assurées par IP

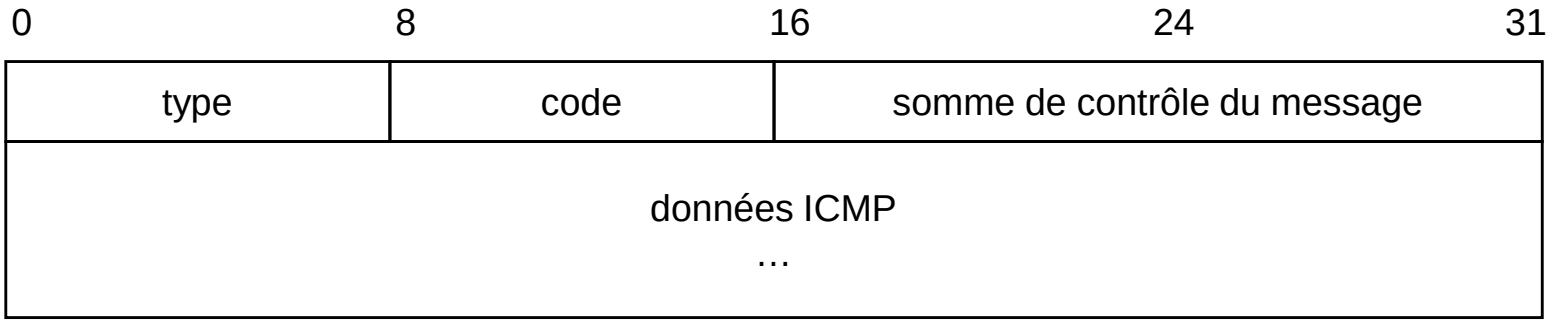
- IP n'assure pas :
  - le multiplexage:
    - ➔ plusieurs flux de données simultanés (assuré par couche 4)
  - la vérification du séquencement :
    - ➔ les paquets peuvent arriver en désordre, voire dupliqués
  - la détection de perte:
    - ➔ doit être assurée par les couches supérieures
  - la retransmission en cas d'erreur:
    - ➔ doit être assurée par les couches supérieures
  - le contrôle de flux:
    - ➔ assuré en partie par ICMP

Protocole ICMP (RFC 792 – STD 5)

- ICMP [Internet Control Message Protocol] gère les messages d'erreurs et de contrôle entre les différents systèmes
- Caractéristiques:
  - Utilise IP (champ protocole=1)
  - Permet de palier au manque de services d'IP
  - Protocole obligatoire sur tous les équipements IP !
  - Il ne demande pas de réponse: un message ICMP ne doit pas engendrer un autre message ICMP
- Message renvoyé à l'expéditeur par l'équipement destinataire ou le routeur intermédiaire :
  - Quand il s'aperçoit d'un problème dans le datagramme:
    - ➔ par exemple: TTL expiré
  - Pour avertir l'émetteur afin qu'il modifie son comportement
    - ➔ par exemple: demande de ralentir l'émission

Message ICMP

- Format du message:



- Chaque type de message a un format particulier:
  - 22 types définis
- Les messages ICMP qui rendent compte d'erreurs renvoient toujours l'en-tête du datagramme IP + 64 premiers bits de données :
  - ➔ permet de localiser facilement les problèmes provenant de protocoles de plus haut niveau

Exemples de messages ICMP

- Messages d'echo (Cf. commande ping):
  - permet de tester la connectivité entre 2 machines
  - demandes (type=8, code=0), réponses (type=0, code=0)
- Compte-rendu de pb de paramètres (type=12, code=0 ou 1):
  - permet de renvoyer des erreurs sur l'en-tête IP ou les options
- Destination inaccessible (type=3):
  - code=0: réseau inaccessible
  - code=1: ordinateur inaccessible
  - code=3: port TCP ou UDP inaccessible
  - etc… (13 codes en tout)
- Contrôle de flux:
  - Demande de ralentissement de l'émission [source quench] (4,0)

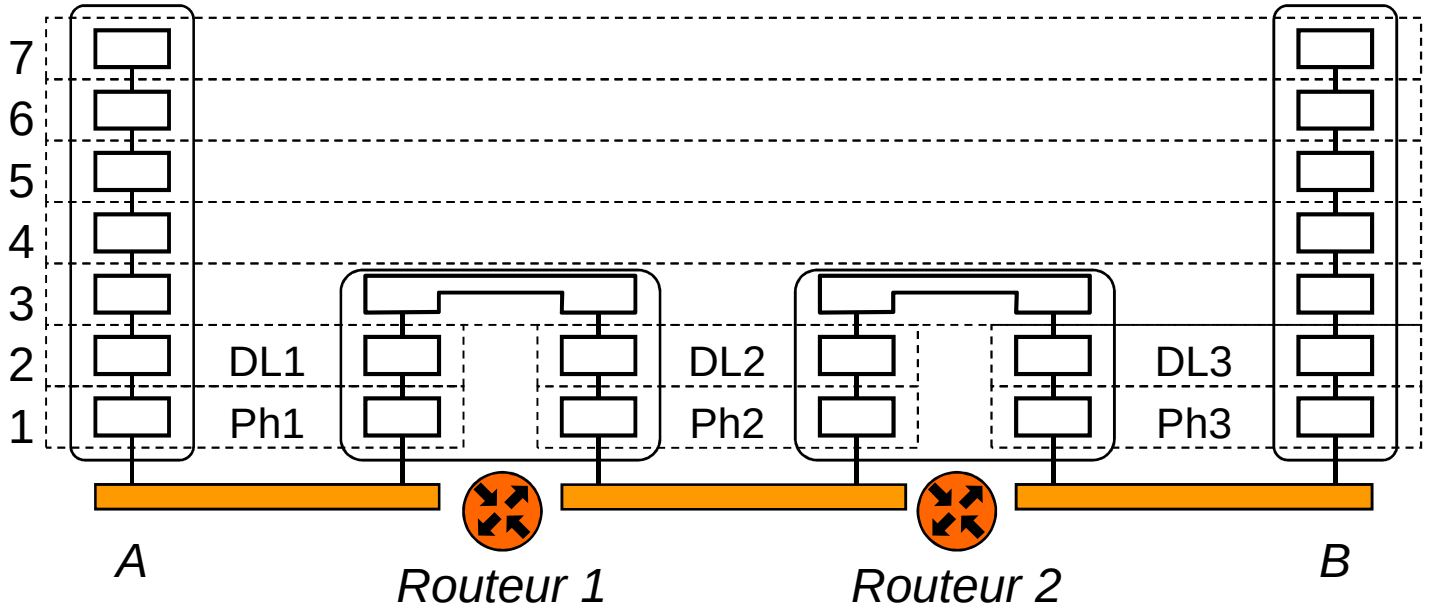
  (Obsolète depuis RFC 6633, mai 2012)
- Durée de vie dépassée (TTL expiré) (type=11, code=0)
- Demande de modifications de routes (type=5, code=0 à 3)

Routage

- Recherche d'un chemin (le meilleur) pour acheminer les paquets d'un point à un autre d'un réseau.
- Situé au niveau 3 du modèle OSI
- Réalisé par des routeurs

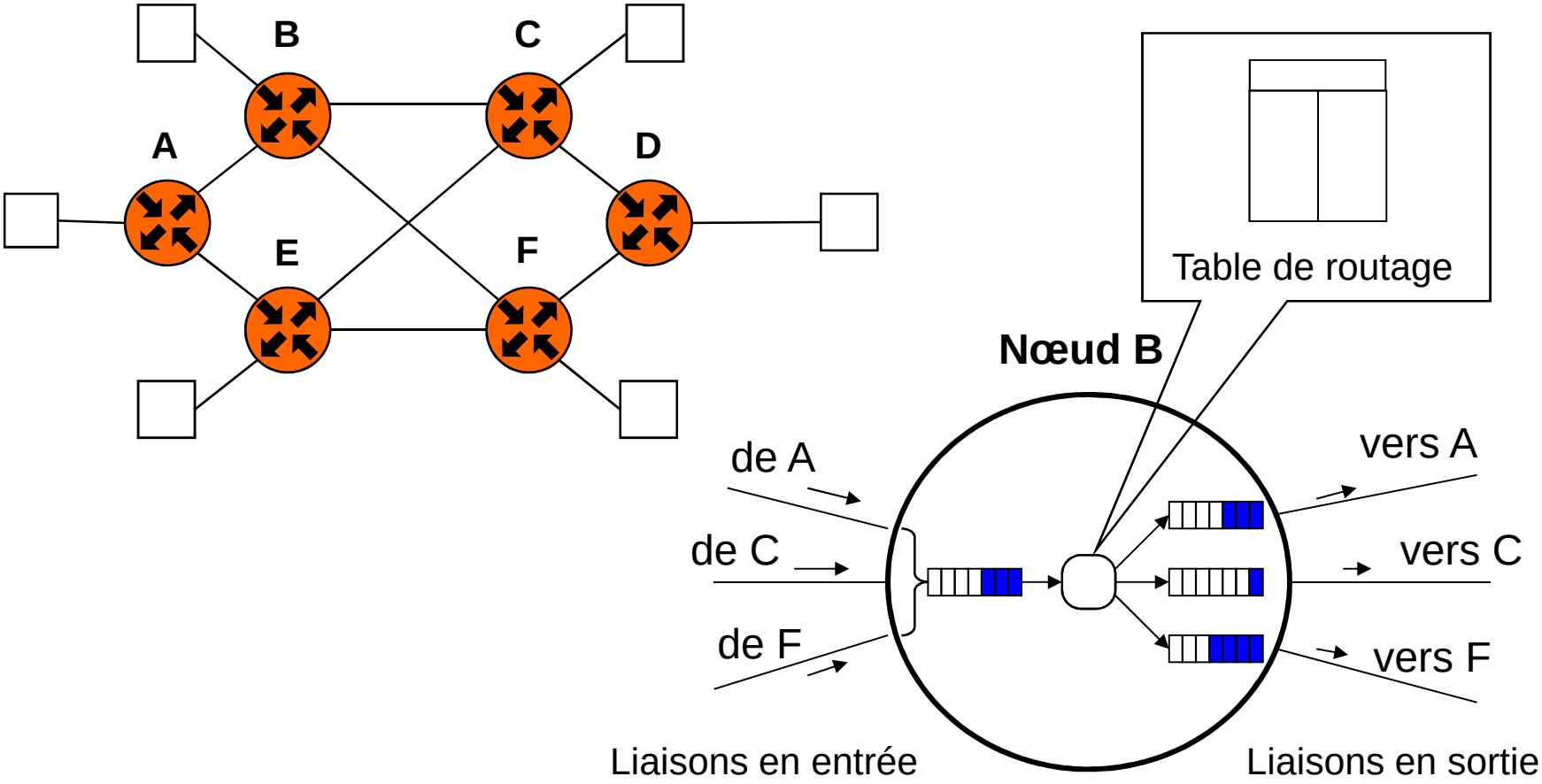
Métaphore postale

Centre de tri postal



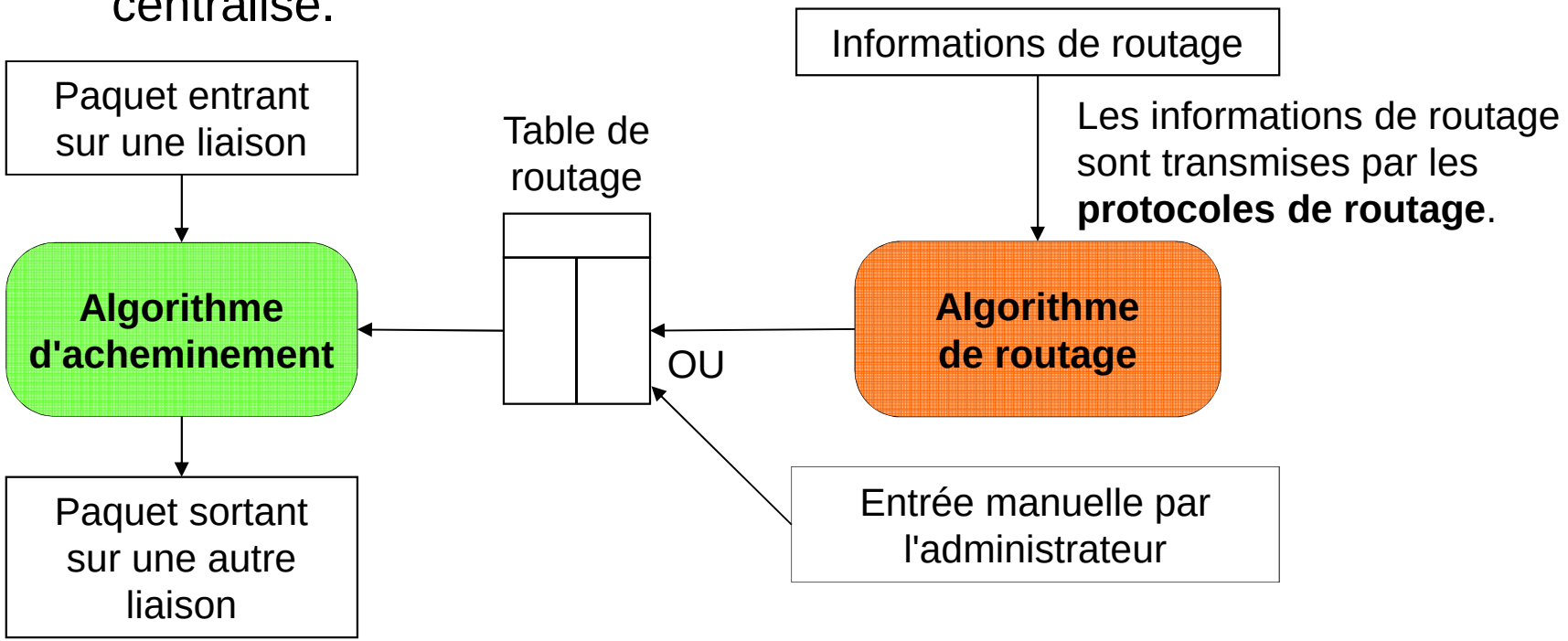
Equipement de routage

- Un équipement de routage possède des files d'attente et une table de routage



Acheminement et routage

- Dans un routeur, Il faut distinguer 2 algorithmes :
  - l'algorithme d'acheminement, exécuté à chaque paquet entrant, par l'équipement de routage, en se basant sur une table
  - l'algorithme de routage qui calcule les tables, exécuté de manière périodique, par l'équipement de routage ou par un système centralisé.



Principe d'une table de routage

- Chaque nœud d'un réseau possède une table de routage:
  - Pour chaque destination possible la table dit vers quelle liaison il faut envoyer le paquet
  - il est possible d'avoir une route par défaut qui est utilisée si la destination n'est pas trouvée dans la table

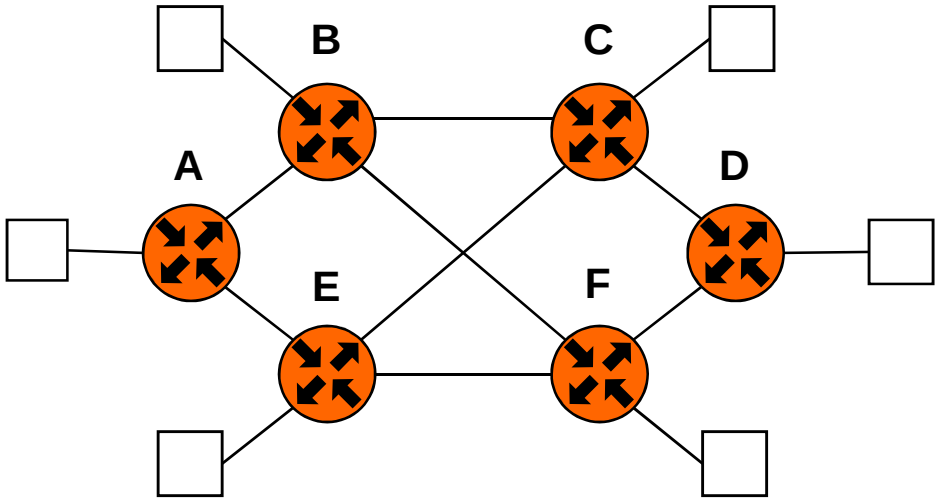
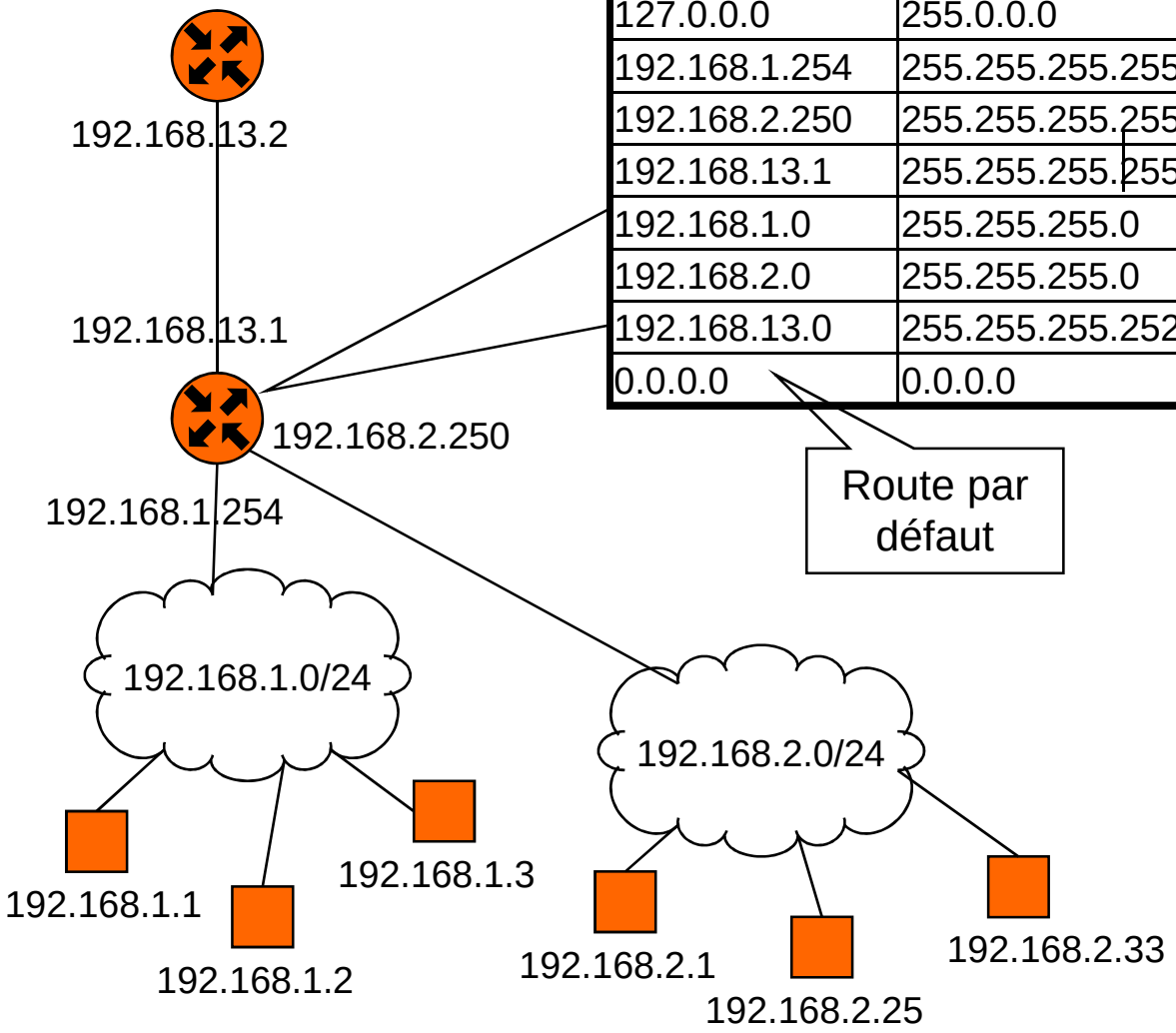


Table de routage de A

Destination	Liaison vers
B	B
C	B
D	B
E	E
F	E

Ici, un paquet ayant pour destination le nœud D va être acheminé sur la liaison vers le nœud B.

Exemple de table de routage IP



Network	Mask	Gateway	Interface
127.0.0.0	255.0.0.0	127.0.0.1	127.0.0.1
192.168.1.254	255.255.255.255	127.0.0.1	127.0.0.1
192.168.2.250	255.255.255.255	127.0.0.1	127.0.0.1
192.168.13.1	255.255.255.255	127.0.0.1	127.0.0.1
192.168.1.0	255.255.255.0	192.168.1.254	192.168.1.254
192.168.2.0	255.255.255.0	192.168.2.250	192.168.2.250
192.168.13.0	255.255.255.252	192.168.13.1	192.168.13.1
0.0.0.0	0.0.0.0	192.168.13.2	192.168.13.1

Route par défaut

Si *gateway* = *interface*
Alors remise directe dans le réseau local

Utilisation de la table de routage IP

- Soit un paquet à router dont l'adresse de destination est 192.168.2.25
  - Pour chaque ligne faire un ET logique avec le masque (colonne 2) et regarder si il y a égalité avec l'adresse réseau (colonne 1)
  - Si oui, alors utiliser l'interface (colonne 4) pour expédier le paquet et
    - ➔ si colonne 4 ≠ colonne3 : transmettre ce paquet au routeur suivant (colonne 3)
    - ➔ Sinon faire une remise directe dans le réseau local
  - Exemple
    - ➔ 1ère ligne : 192.168.2.25 ET 255.0.0.0 = 192.0.0.0 ≠ 127.0.0.0 ➔ NON
    - ➔ 6ème ligne : 192.168.2.25 ET 255.255.255.0 = 192.168.2.0 ➔ OUI

Ligne à utiliser

Remarque: si plusieurs lignes de la table de routage conviennent alors choisir celle qui a le masque le plus long

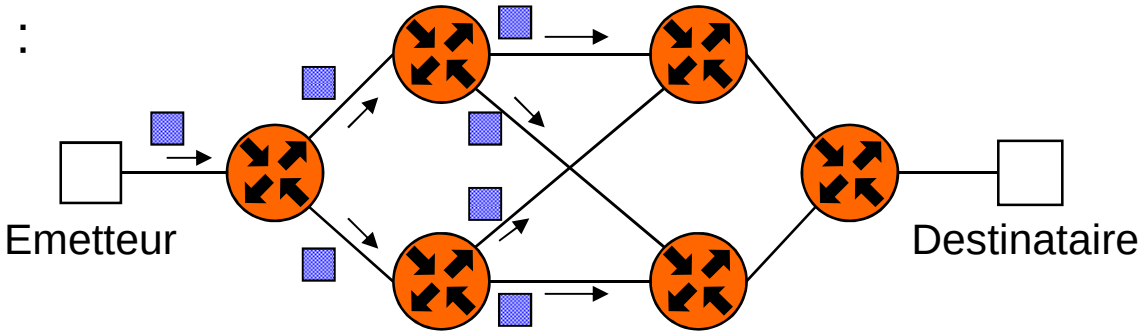
Network	Mask	Gateway	Interface
127.0.0.0	255.0.0.0	127.0.0.1	127.0.0.1
192.168.1.254	255.255.255.255	127.0.0.1	127.0.0.1
192.168.2.250	255.255.255.255	127.0.0.1	127.0.0.1
192.168.13.1	255.255.255.255	127.0.0.1	127.0.0.1
192.168.1.0	255.255.255.0	192.168.1.254	192.168.1.254
192.168.2.0	255.255.255.0	192.168.2.250	192.168.2.250
192.168.13.0	255.255.255.252	192.168.13.1	192.168.13.1
0.0.0.0	0.0.0.0	192.168.13.2	192.168.13.1

Types de routage

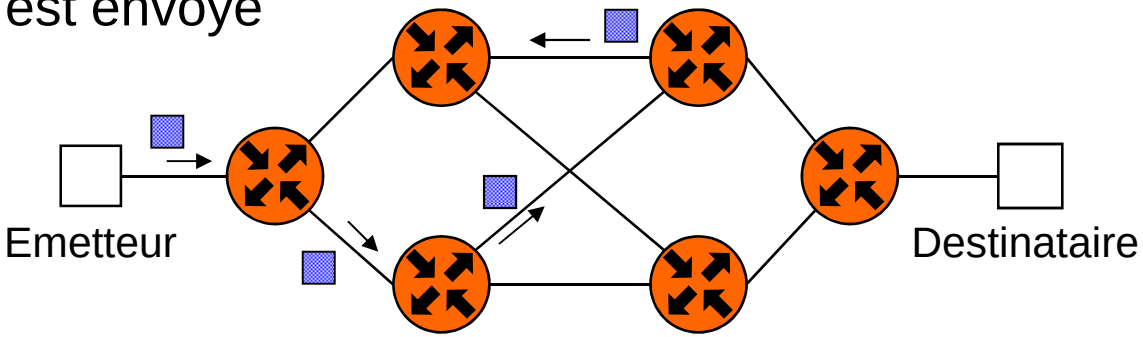
- Adaptabilité aux changements dans le réseau :
  - routage statique: le chemin calculé est fixe
  - routage dynamique: le chemin peut varier en fonction
    - ➔ de l'état du réseau: panne, nouvelle architecture
    - ➔ du trafic sur le réseau: encombrements, etc…
- Distribution du calcul sur le réseau:
  - routage isolé: chaque routeur calcule sa table (ou pas de table !)
  - routage centralisé: une seule machine centrale fait les calculs et transmet les résultat aux routeurs
  - routage distribué: chaque routeur fait une partie du calcul et communique ses résultats aux routeurs voisins
- But: trouver le « meilleur » chemin possible à travers le réseau
- En pratique: très grand nombre de nœuds et de liaisons
  - on utilise des algorithmes de routage distribués et dynamiques

Routage isolé (1/2)

- Inondation [flooding]
  - Chaque paquet reçu est réexpédié sur toutes les liaisons
  - Il faut une condition d'arrêt pour contrôler l'inondation :
    - ➔ par exemple : un nombre de sauts maximum

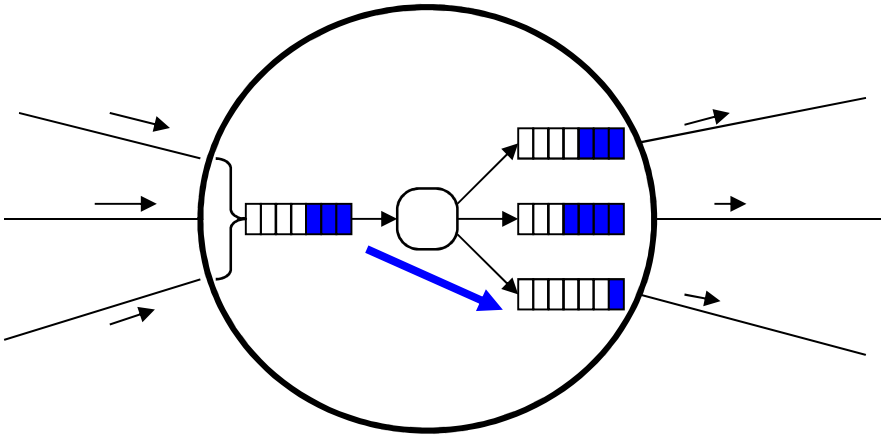


- Marche aléatoire [random routing]
  - Un paquet qui arrive est envoyé sur une liaison de manière aléatoire



Routage isolé (2/2)

- Patate chaude [hot potatoe]
  - Chaque paquet reçu est réexpédié sur la liaison ayant la file d'attente la plus courte

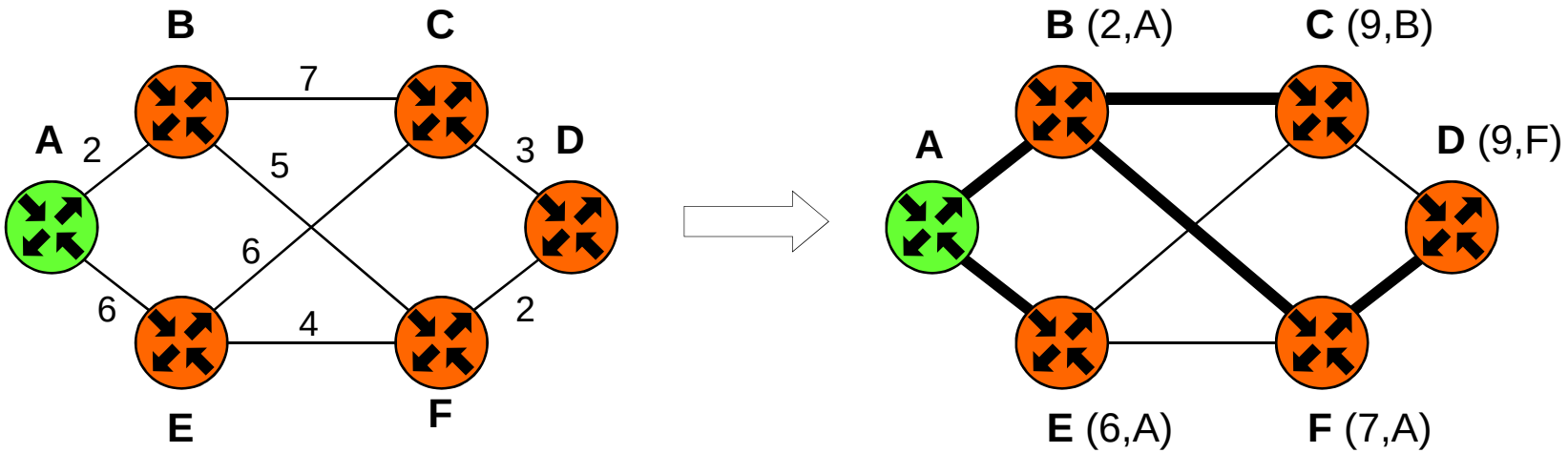


- Rumeur
  - Chaque paquet contient :
    - le nœud destinataire
    - le nœud émetteur
    - et le nombre de nœuds parcourus
  - Un nœud qui reçoit un paquet en déduit une distance approximative / nœud émetteur
  - Chaque nœud se construit ainsi une représentation approximative du réseau

Routage centralisé

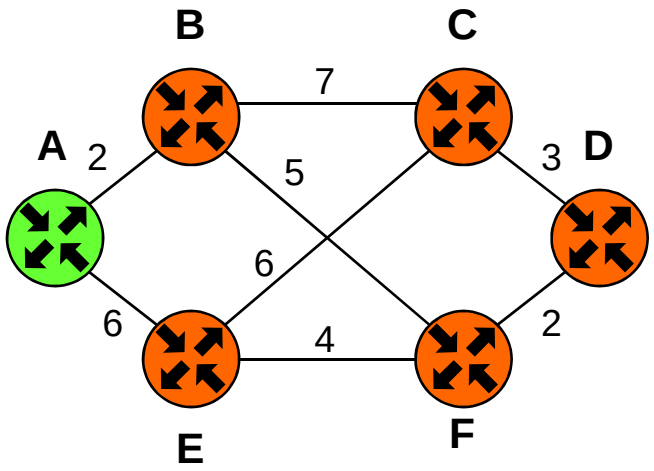
- Un nœud central calcule les tables pour tous les autres nœuds

- Algorithme du plus court chemin [Shortest path] ou Dijkstra
  - Chaque liaison possède un poids qui représente le critère à minimiser (délai, coût, etc.)
  - A la fin du calcul, à chaque nœud est associé :
    - La longueur minimale depuis le nœud de départ
    - Le prédécesseur ce qui permet de reconstruire un arbre recouvrant



Routage distribué (1/3)

- Algorithme de Vecteur de distance (Bellman-Ford, Ford-Fulkerson)
  - Chaque nœud transmet périodiquement un vecteur de distance à ses voisins : destination, ligne et distance associée.
  - Etapes pour chaque nœud:
    - Réception des vecteurs de distance des voisins
    - Calcul d'une nouvelle table
    - Transmission d'un vecteur de distance aux voisins



Le nœud A reçoit les vecteurs de ses 2 voisins :

de B :

Dest.	Dist.
A	2
B	0
C	7
D	7
E	8
F	5

d(A,B)=2

de E :

Dest.	Dist.
A	6
B	8
C	6
D	6
E	0
F	4

d(A,E)=6

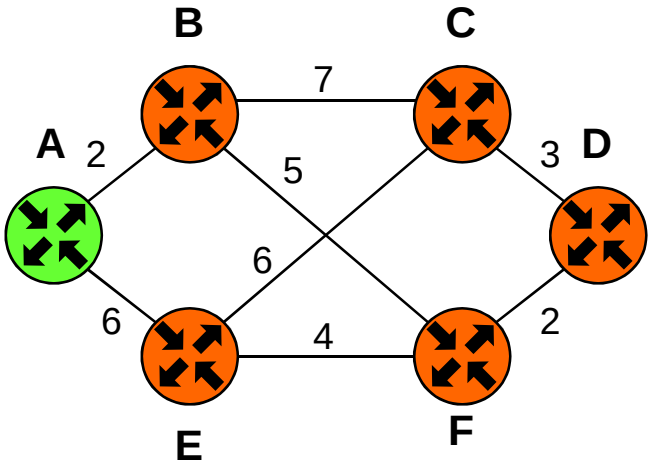
Nouvelle table de A

Dest.	Dist.	Lien
A	0	-
B	2	B
C	9	B
D	9	B
E	6	E
F	7	B

Routage distribué (2/3)

- Etat des liaisons [link state routing] :
  - Chaque nœud a une connaissance complète de la topologie du réseau
  - Etapes pour chaque nœud :
    - Découvrir ses voisins
    - Mesurer le temps d'acheminement aux voisins

    } Peut être configuré a priori
    - Construire un paquet d'information d'état de lien
    - Envoyer ce paquet à tous les routeurs du réseau
    - Calculer le plus court chemin vers tous les routeurs (Dijkstra par ex.)



Paquets d'état de liaison de chaque nœud :

A	
Séq.	
Âge	
B	2
E	6

B	
Séq.	
Âge	
A	2
C	7
F	5

C	
Séq.	
Âge	
B	7
D	3
E	6

D	
Séq.	
Âge	
C	3
F	7

E	
Séq.	
Âge	
A	6
C	6
F	4

F	
Séq.	
Âge	
B	5
D	2
E	4

Routage distribué (3/3)

- Avantages de ces algorithmes :
  - En cas de problème sur une liaison le réseau peut se reconfigurer
- Inconvénient principal des algos à vecteur de distance:
  - Convergence trop lente en cas de panne :
    - ➔ Établissement de boucles transitoires
    - ➔ Pb de la valeur infinie : il faut définir une valeur max qui est l'infini :
      - Faible pour converger rapidemment
      - Plus grande que la plus longue route (sinon des routeurs se retrouvent derrière « l'horizon » !)
- Inconvénient principal des algos à état des liaisons:
  - Difficulté de distribuer l'état des liaisons à TOUS les routeurs :
    - ➔ Certains routeurs peuvent se faire une représentation incorrecte de la topologie ce qui conduit à des boucles
    - ➔ Utilisation des champs « séquence » et « âge » et maintient d'un tampon de tous les échanges dans chaque routeur ➔ lourd !

Protocoles de routage IP

- Basés sur algorithme de Vecteur de distance (ou Bellman-Ford, Ford-Fulkerson):
  - Hello [RFC 891]:
    - inutilisé
  - RIP (Routing information Protocol) [RFC 1058] et RIP2 [RFC 2453 – STD 56] :
    - Métrique = nombre de nœuds (15 max. ; 16 = ∞)
    - Un message toutes les 30 secondes
  - IGRP et EIGRP (Cisco)
    - Proche de RIP + détection des boucles

} Pour converger vers un état stable, il faut plusieurs minutes !

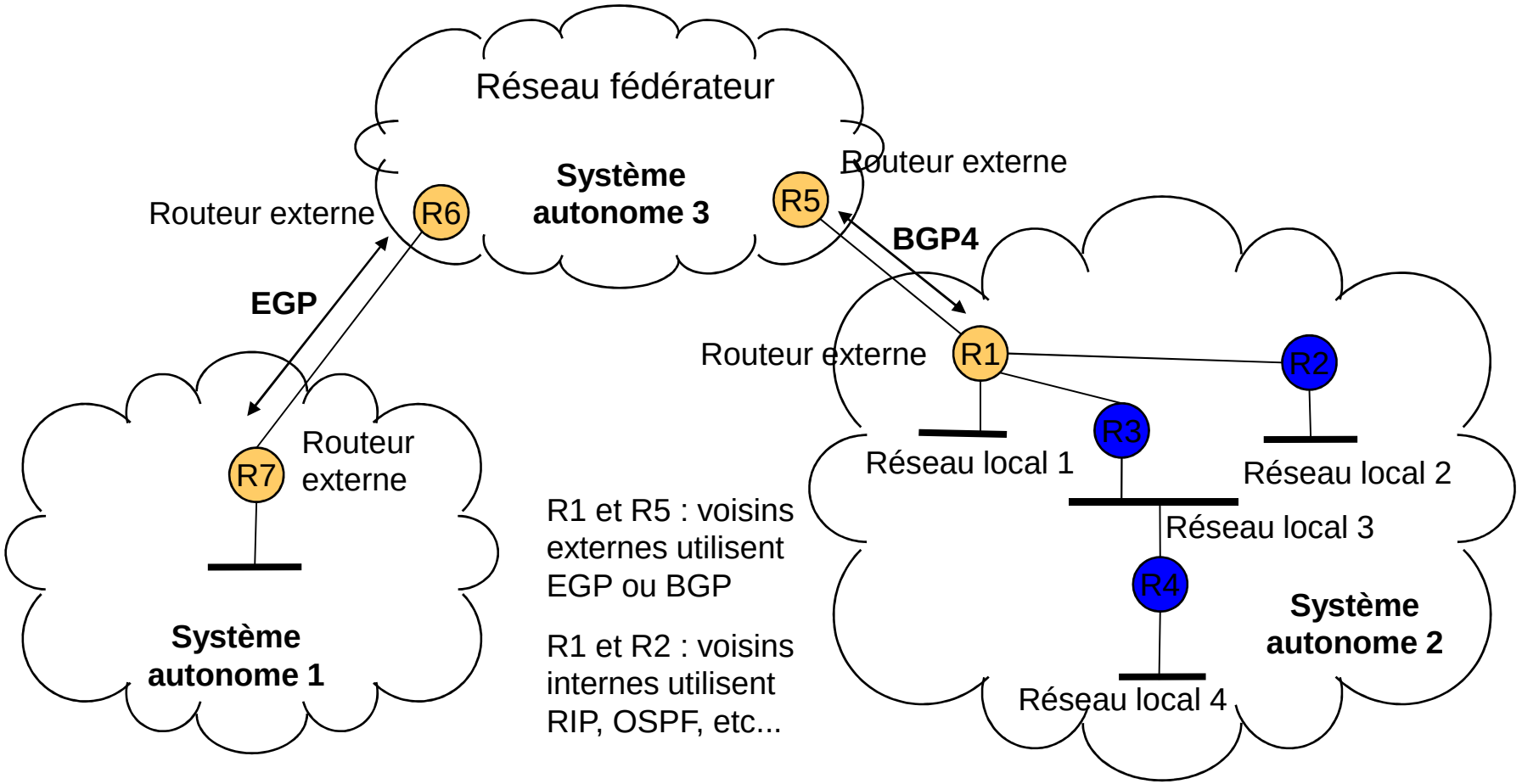
- Basés sur algorithme d'état des liaisons :
  - OSPF (Open Shortest Past First) version 2 [RFC 2328 – STD 54]:
    - performant: équilibrage de charge, sous-zones, routage par type de service, authentification des routeurs
    - complexe à mettre en œuvre (trop complexe ?)
  - EIGRP (Cisco)

Routage dans Internet (1/2)

- Utilisation du routage hiérarchique:
  - perfectionnement du routage distribué permettant de réduire la taille des tables
  - Le réseau est découpé en systèmes autonomes : ensemble de réseaux et de routeurs sous une même responsabilité administrative
- Système Autonome (AS=Autonomous System) :
  - on est libre de choisir le routage à l'intérieur (RIP, OSPF, ...):
    - ➔ les routeurs internes ne connaissent que les routes à l'intérieur de l'AS
    - ➔ ils ont une route par défaut vers le (ou les) routeur(s) externe(s)
  - les routeurs externes permettent de passer d'un AS à un autre:
    - ➔ ils échanges des informations d'accessibilité à leurs réseaux internes entre eux
  - 3 catégories de réseaux : stub (sans issue), réseaux multiconnectés et réseaux de transit

Routage dans Internet (2/2)

- Echange de routes entre AS:
  - EGP (Exterior Gateway Protocol) [RFC 827]
  - BGP (Border Gateway Protocol): v3 [RFC 1267] et v4 [RFC 1771]



IP version 6

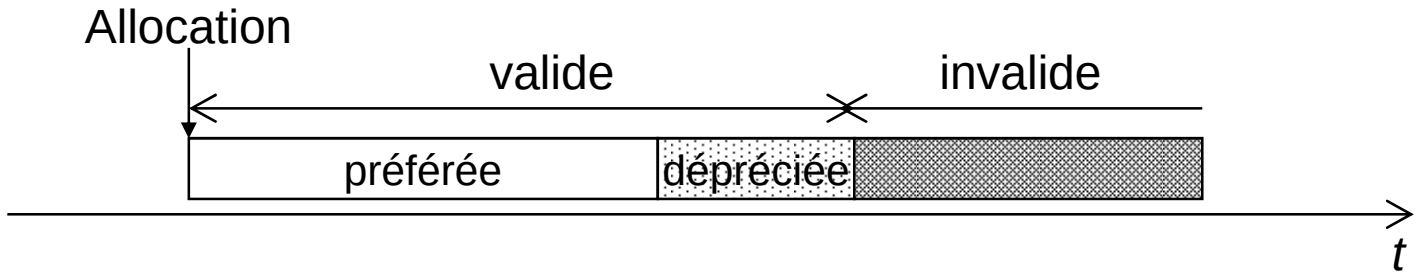
- Objectifs de ce nouveau protocole:
  - résoudre le problème de l'épuisement des adresses IPv4 (prévu vers 2008 à +/- 3 ans au début des années 90)
  - réduire la taille des tables de routage grâce à l'organisation hiérarchique des adresses
  - simplifier le protocole ==> rapidité de traitement
  - fournir une meilleure sécurité: authentification et chiffrement
  - gérer la qualité de service en particulier pour le temps réel
  - gérer la diffusion multicast en standard
  - gérer la mobilité des ordinateurs
  - permettre l'évolution future du protocole
- Mais :
  - IPv4 et IPv6 sont incompatibles entre eux
  - il faut assurer la coexistence des 2 versions pendant la période transitoire (au minimum 10 ans, peut-être toujours !)

Adresses IPv6

- Adresses de 128 bits (16 octets):
  - 340.10^{36} équipements adressables (soit 10^{23} adresses par m^2 !)
- Notation:
  - 8 groupes de 4 chiffres hexadécimaux séparés par ":"
    - Exemple : 8000:0000:0000:0000:0123:4567:89AB:CDEF
  - Forme textuelle canonique [Cf. RFC 4191]
    - Pas de zéro non significatif ; Une séquence de zéro s'abrège avec ::
    - L'adresse du dessus se note : 8000::123:4567:89ab:cdef
- 3 catégories:
  - Unicast
  - Multicast (remarque: le broadcast est supprimé, utilise le multicast)
  - Anycast : adressage au "plus près" = adresses unicast partagées par plusieurs équipements redondants

Attribution des adresses

- Les adresses ne sont plus attribuées « à vie »:
  - Les adresses sont attribuées temporairement à une interface
  - Plusieurs adresses peuvent être attribuées simultanément
  - Cycle de vie d'une adresse :



- Sites multi-connectés :
  - Chaque opérateur attribut ses adresses dans son bloc
    - → Permet de préserver l'adressage hiérarchique
    - → Une machine d'un site muti-connecté a donc plusieurs adresses

Types et allocations des adresses

- Allocation des adresses (RFC 4291) :

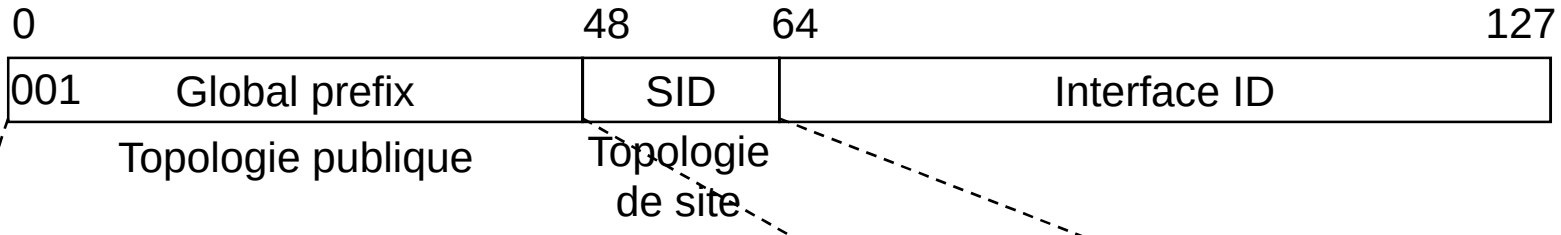
Adresse	Préfixe binaire	Allocation
0::/8	0000 0000	Adresses spéciales et IPv4-embedded IPv4-mapped ::ffff:xxx.xxx.xxx.xxx
2000::/3	001	**adresses globales unicast**
FC00::/7	1111 110	adresses uniques locales (ULA)
FE80::/10	1111 1110 10	adresses de liaisons locales
FEC0::/10	1111 1110 11	adresses locales de sites (obsolète)
FF00::/8	1111 1111	adresses multicast

Toutes les autres adresses sont non-allouées (85%)

- Adresses particulières:
  - Adresse non-spécifiée ::/128
  - Adresse de bouclage ::1/128
  - adresses IPv4 transformées en adresses IPv6 :
    exemple: 156.18.22.3 devient ::FFFF:156.18.22.3

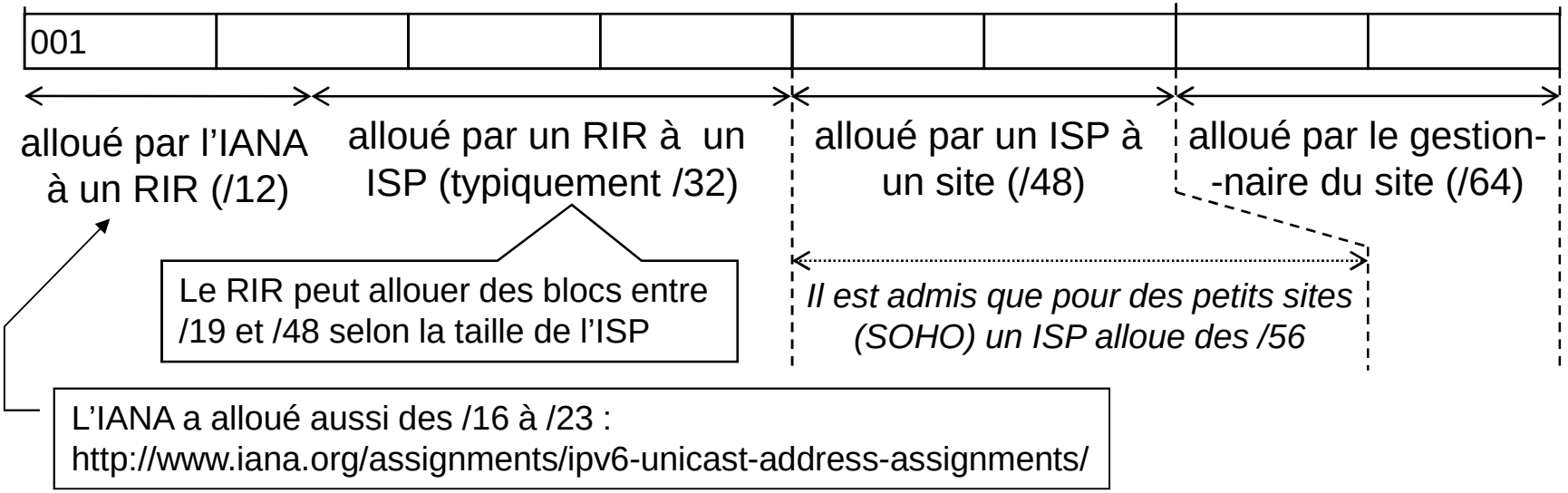
Adresses Unicast globales IPv6 (1/2)

- 2 parties : numéro de réseau | numéro d'hôte



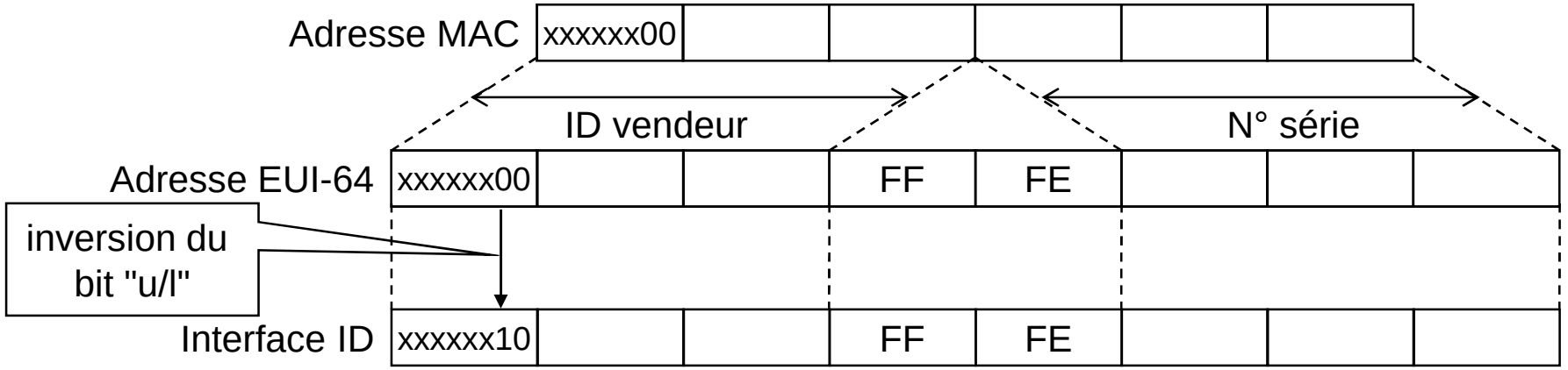
  - Cette partition n'est valable que pour 2000::/3 (Cf. RFC 3587)

- Topologie publique (adresses globales unicast) :



Adresses Unicast globales IPv6 (2/2)

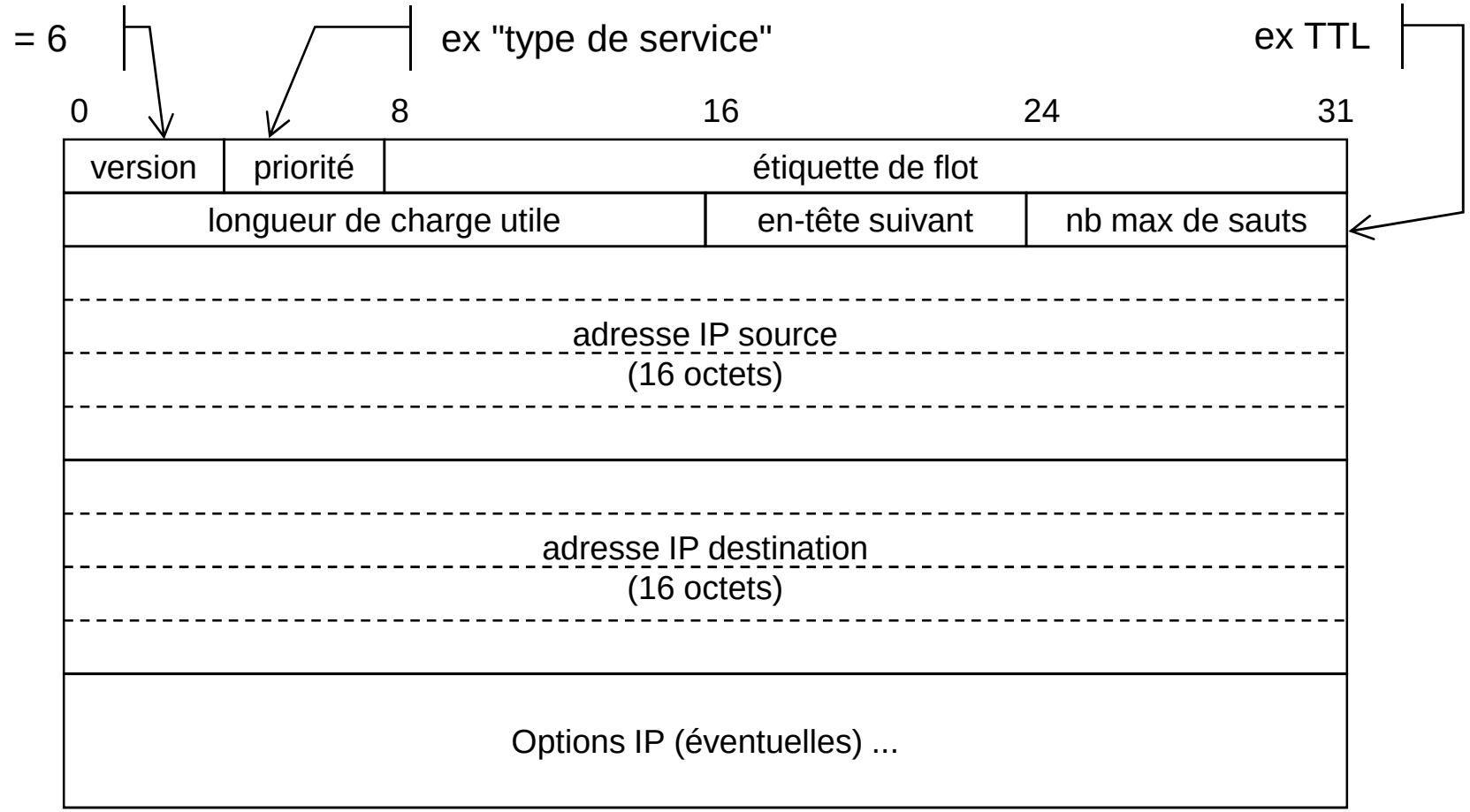
- ID d'interface :
  - A partir de l'adresse de la carte d'interface
    - → adresses EUI-64 (définies pour IEEE 1394 et utilisées par 802.15.4)
    - → méthode pour obtenir l'EUI-64 à partir de l'adresse MAC [RFC 2464] :



  - Numéroter « à la main »
    - → Utile pour des serveurs
  - Générer aléatoirement
  - Dériver l'ID à partir d'une clef publique
    - → facilite l'authentification de l'équipement

ID=0 est réservé comme adresse anycast des routeurs du ss-réseau

En-tête IPv6 (1/3)



En-tête IPv6 (2/3)

- Priorité:
  - permet de classer la nature des flux
  - utile aux routeurs en cas de congestion

0	sans priorité
1	trafic de base (news)
2	transf. de données simple (e-mail)
3	réservé
4	transf. par bloc avec attente du récepteur (ftp)
5	réservé
6	trafic interactif (telnet, rlogin)
7	contrôle de flux et routage

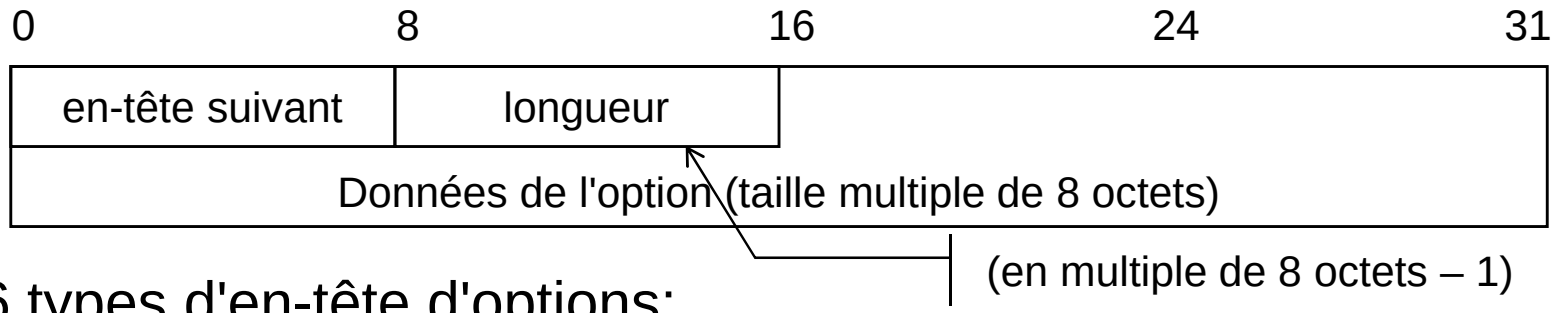
- Etiquette de flot:
  - permet de définir des pseudo-circuits virtuels
  - utile pour le transport de données à fortes contraintes temporelles
- Plus de champ "somme de contrôle":
  - calcul trop "coûteux"
  - les réseaux sont de plus en plus fiables
  - possibilité de la calculer au niveau 2 (liaison de données)

En-tête IPv6 (3/3)

- Plus de champs pour la fragmentation:
  - datagrammes de 576 octets doivent être supportés
  - si paquet trop grand:
    - le routeur renvoie un code d'erreur ICMP
    - et possibilité d'utiliser l'option de fragmentation à la source
- En-tête suivant:
  - contient:
    - soit le type d'en-tête d'option qui suit,
    - soit le type de protocole de niveau supérieur (TCP, UDP, …)

0	Hop-by-hop Option Header
4	IPv4
6	TCP
17	UDP
41	IPv6
43	Routing Header
44	Fragment Header
45	Interdomain Routing Protocol
46	Resource Reservation Header
50	Encapsulating Security Payload
51	Authentification Header
58	ICMPv6
59	No next Header
60	Destination Options Header

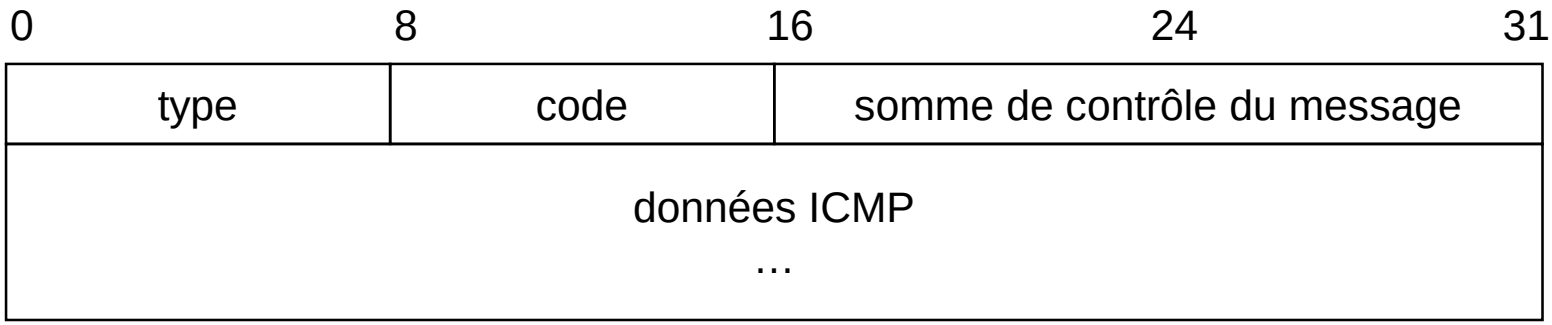
Options d'en-tête IPv6



- 6 types d'en-tête d'options:
  - pas-à-pas: option traitée par chaque routeur,
    - ➔ permet en particulier d'avoir des jumbogrammes (datagrammes > 64 Ko) ; dans ce cas le champ longueur = 0
  - routage: strict ou lâche (seuls quelques routeurs sont précisés)
  - fragmentation: similaire à IPv4 mais seul l'ordinateur source fragmente. En général, on recherche le MTU pour l'éviter
  - authentification: signature des datagrammes (par défaut MD5)
  - charge utile chiffrée: permet la confidentialité, seul le destinataire peut lire les données (par défaut utilise chiffrement DES)
  - option de destination: non utilisée

ICMPv6

- Protocole de contrôle similaire à ICMPv4



- Nouveaux types attribués plus logiquement :
  - <127 : messages d'erreurs
  - >128 : messages d'information
- Nouvelles fonctionnalités :
  - Configuration automatique
  - Découverte des voisins (Neighbor discovery)

Neighbor discovery

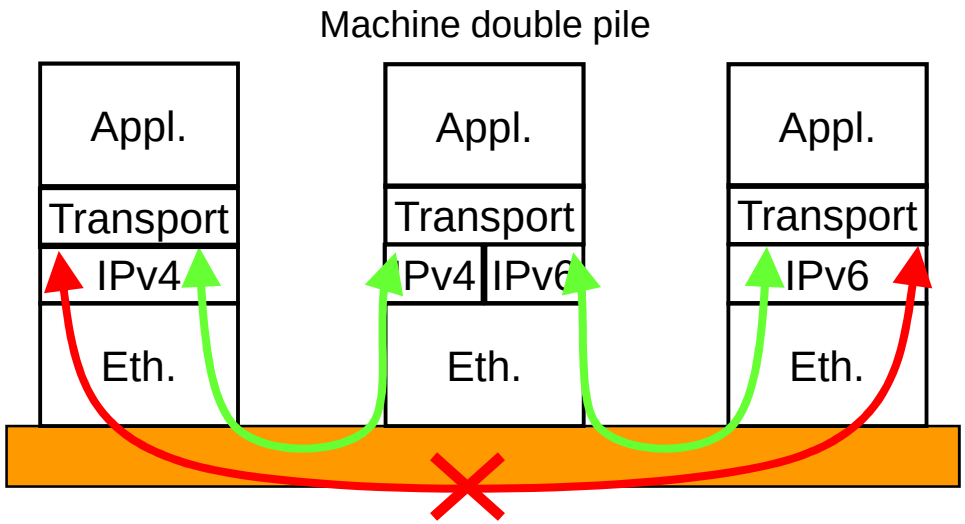
- Permet la « découverte des voisins » et la communication avec ceux-ci dans le réseau local
- Utilise des messages ICMPv6 multicast avec nb sauts = 255
- Fonctions:
  - Résolution des adresses : idem ARP pour IPv4
  - NUD (Neighbor Unreachability Detection)
  - Configuration :
    - découverte du préfixe réseau,
    - découverte des routeurs,
    - découverte des adresses dupliquées,
    - découverte des paramètres (taille MTU, …)
  - Indication de redirection : reconfiguration de routes
- Il existe aussi DHCPv6 pour une configuration « avec état » de l'interface

Transition IPv4 – IPv6

- Très nombreuses solutions pour « faciliter » la transition ou la coexistence IPv4 – IPv6 :
  - Solutions locales à la machine :
    - Dual-Stack + Adresses mappées IPv4
  - Solutions de Tunneling :
    - 6to4
    - 6rd
    - Teredo
    - ISATAP
  - Solutions de traduction de paquets :
    - NAT-PT, NAPT-PT (RFC 2766 abandonné)
    - SIIT
    - NAT64 + DNS64

Solutions locales à la machine

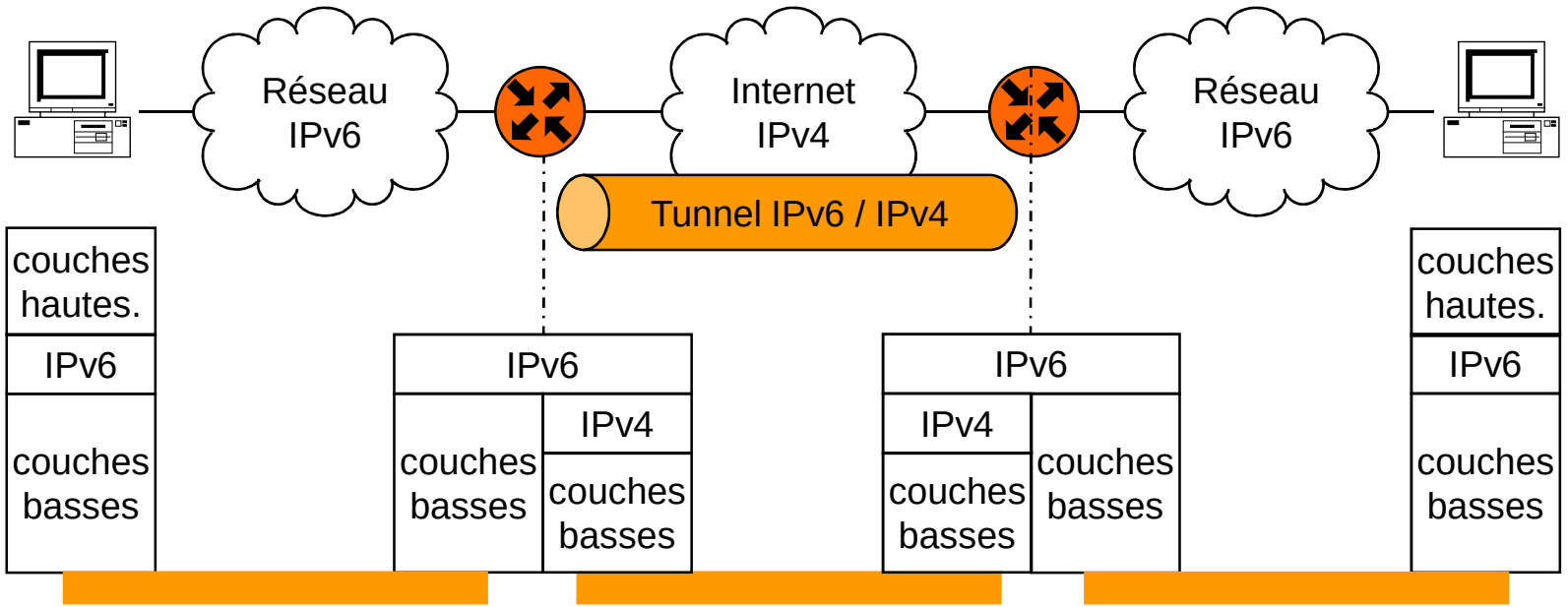
- Double pile [dual stack]
  - Des machines ont les 2 protocoles IPv4 et IPv6
  - Pb: travail doublé pour les administrateurs car les réseaux locaux doivent supporter les 2 technologies



- Adresses IPv4 mappées
  - Adresses IPv6 de la forme **`::ffff:a.b.c.d`**
  - Permet aux applications d'une machine double pile de voir toutes les adresses sous la forme d'adresses IPv6

Solutions de tunneling (1/5)

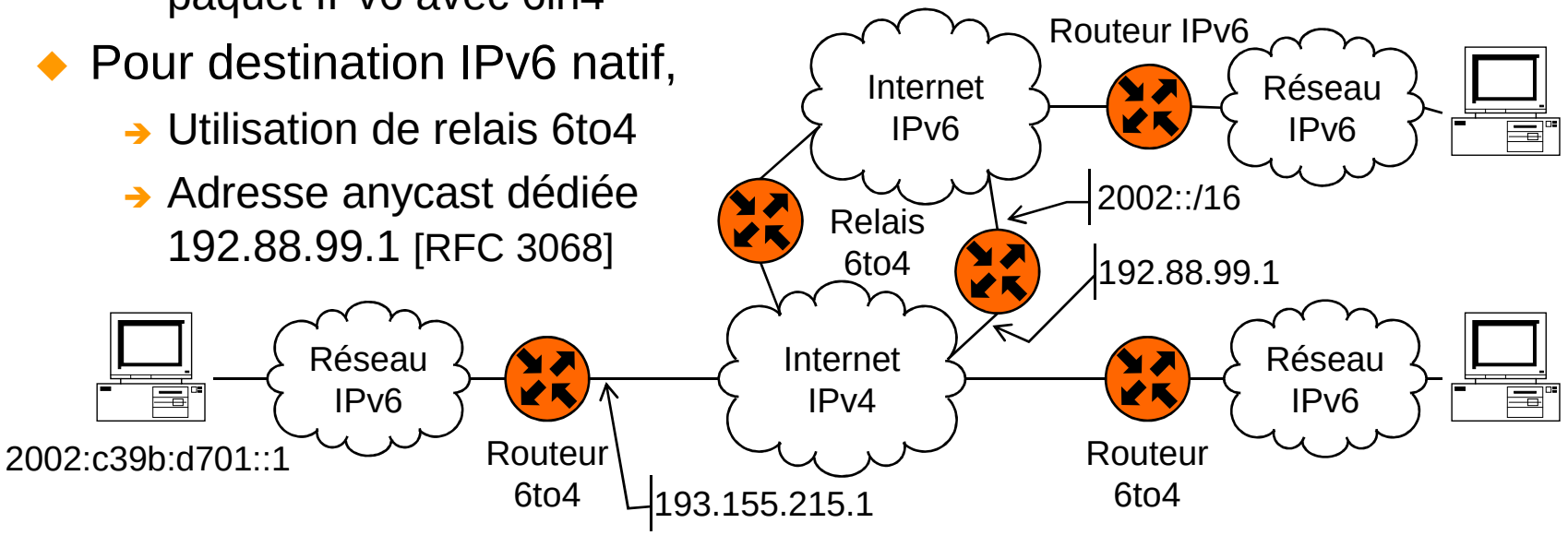
- Tunneling classique
  - 6in4 [RFC 4213] : permet d'interconnecter 2 réseaux IPv6 à travers un ancien réseau uniquement IPv4
  - Encapsulation IPv6 dans IPv4 (n° protocole = 41)
  - Inconvénient : configuration manuelle du tunnel



  - Sur le même principe on peut faire l'inverse avec 4in6 [RFC 2473]

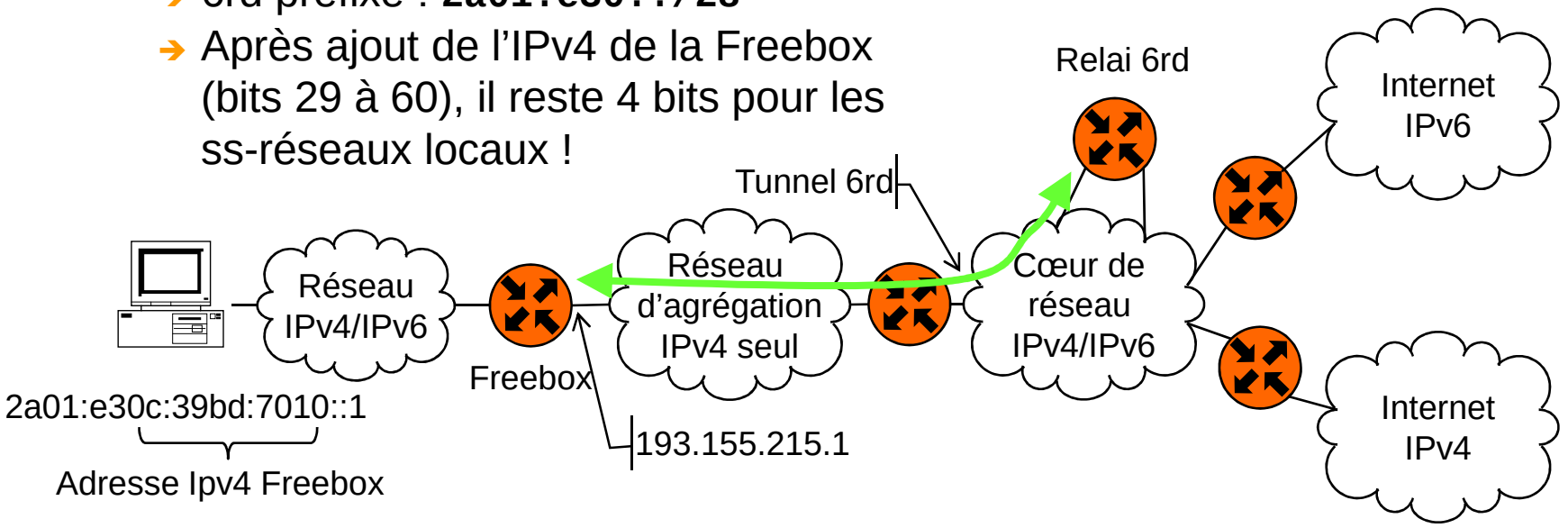
Solutions de tunneling (2/5)

- 6to4 [RFC 3056]:
  - Tunnel automatique à partir de l'adresse IPv4 des routeurs
  - Préfixe réseau local : **`2002:xxxx:xxxx::/48`**
    - xxxx:xxxx est l'adresse IPv4 du routeur en hexadécimal
    - Exemple : routeur = 193.155.215.1 ➔ préfixe = **`2002:c39b:d701::/48`**
  - Pour destination en 2002::/16
    - le routeur 6to4 extraie l'adresse IPv4 destination avant d'encapsuler le paquet IPv6 avec 6in4
  - Pour destination IPv6 natif,
    - Utilisation de relais 6to4
    - Adresse anycast dédiée 192.88.99.1 [RFC 3068]



Solutions de tunneling (3/5)

- 6rd (rapid deployement) [RFC 5569/RFC 5969] :
  - Palie à un inconvénient de 6to4
    - L'usage du préfix 2002:/16 limite la qualité de service et peut poser des problèmes de fractionnement en cas de panne du relai 6to4
  - Fonctionnement similaire à 6to4 mais le préfixe réseau est spécifique au fournisseur d'accès
  - Exemple de Free :
    - Préfixe IPv6 obtenu de RIPE : **`2a01:e00::/26`**
    - 6rd préfixe : **`2a01:e30::/28`**
    - Après ajout de l'IPv4 de la Freebox (bits 29 à 60), il reste 4 bits pour les ss-réseaux locaux !



Solutions de tunneling (4/5)

- Teredo [RFC 4380+5991+6081]
  - Permettre une connectivité IPv6 à des machines sur des réseaux IPv4
  - Marche sur des réseaux à adresses privées (derrière des NAT44)
  - Encapsule les paquets IPv6 dans des segments UDP/IPv4
  - 4 types de machines :
    - Clients Teredo : hôte ayant une connectivité IPv4 derrière un NAT44
    - Serveur Teredo : machine bien connue permettant la configuration initiale du tunnel
    - Relai Teredo : machine à l'autre bout du tunnel et qui relaie le trafic
    - Relai spécifique à un hôte : relai intégré à une machine pour échanger avec des clients Teredo mais qui utilise IPv6 natif pour le reste de sa connectivité IPv6
  - Adresses IPv6 des clients sont construites ainsi :

Solutions de tunneling (5/5)

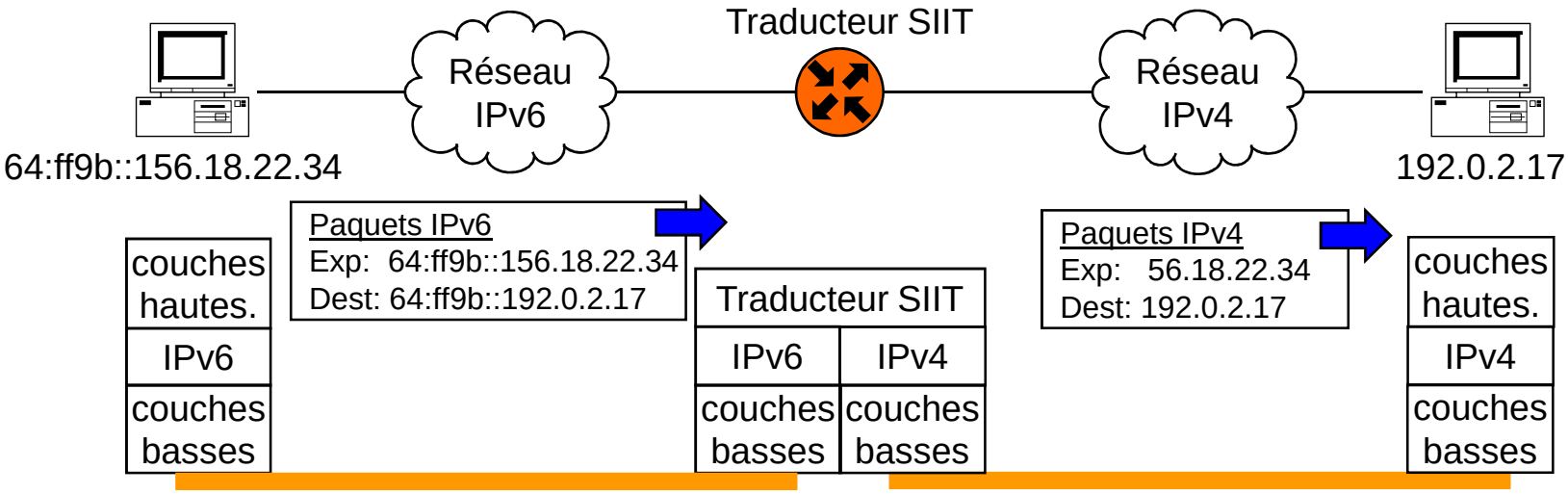
- ISATAP (Intra-Site Automatic Tunnel Addressing Protocol) [RFC 4214]
  - Permettre une inter-connectivité locale IPv6 entre des machines double pile sur un réseau local IPv4
  - Spécifie comment générer une adresse local de lien à partir de l'adresse IPv4 sous la forme fe80::200:5efe:a.b.c.d
  - Inconvénient : comme le réseau IPv4 ne supporte pas forcément le multicast, Neighbor Discovey ne peut pas fonctionner !
    - Nécessité de définir une liste de routeurs potentiels (PRL=potential routers list)
    - Chaque routeur est périodiquement interrogé par des messages ICMPv6 *router discovery* pour obtenir le préfixe réseau
    - La PRL est typiquement obtenue en interrogeant le DNS sur le domaine: isatap.*mondomaine.tld*

Solutions de traduction de paquets (1/3)

- NAT-PT et NAPT-PT [RFC 2766 rendu **obsolète** par RFC 4966]
  - NAT-PT (Network Address Translation/Protocol Translation)
  - NAPT-PT (Network Address Port Translation + Protocol Translation)
  - Principes similaires à NAT et NAPT avec IPv4
    - Tables de traduction d'adresses (et de ports) IPv4 ←→IPv6
- Aujourd'hui :
  - RFC 6144: cadre général de traduction des paquets IPv4 – IPv6
  - RFC 6052: adresses « IPv4 embarquées » (IPv4-embedded)
    - Cadre général pour inclure une adresse IPv4 dans une adresse IPv6
    - Plusieurs types de préfixes /32, /40, /48, /56, /64 et /96
    - Well known prefix : **`64:ff9b::/96`** (choisi car le checksum est nul) d'où Format des adresses : **`64:ff9b::a.b.c.d`**
  - 2 versions de traductions définies :
    - NAT64 (Stateful Network Address and Protocol Translation)
    - SIIT (Stateless IP/ICMP Translation)

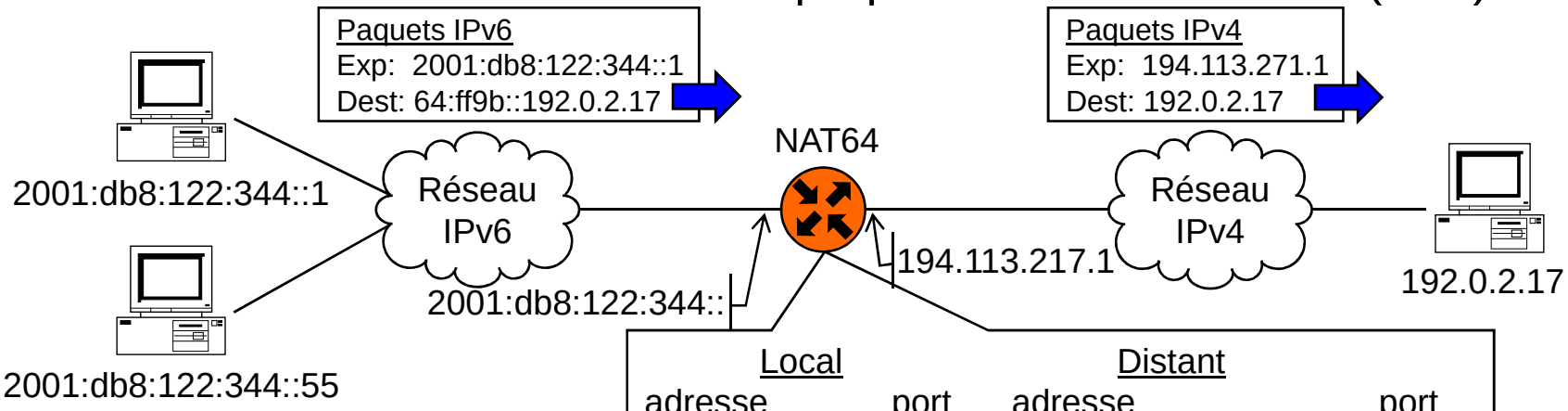
Solutions de traduction de paquets (2/3)

- SIIT (Stateless IP/ICMP Translation) [RFC 6145]
  - Spécifie la traduction des en-têtes IPv4 en en-tête IPv6 et vice-versa
  - Traite la gestion des paquets fragmentés
  - Spécifie la traduction des paquets ICMPv4 en ICMPv6 et vice-versa
  - Traite la modification des en-têtes TCP/UDP induite par la traduction
- Remarque :
  - en pratique chaque nœud IPv6 doit avoir une adresse IPv4 valide (éventuellement temporaire) pour communiquer avec les nœuds IPv4



Solutions de traduction de paquets (3/3)

- NAT64 (Stateful Network Addr. and Protocol Translation) [RFC 6146]
  - Similaire à NAT44
    - Non symétrique : seuls les nœuds IPv6 peuvent initier une session
    - Une **seule adresse IPv4 partagée** par plusieurs nœuds IPv6
    - Supporte ICMP, TCP et UDP seulement (pour l'instant)
  - La traduction des en-têtes des paquets suit le RFC 6145 (SIIT)



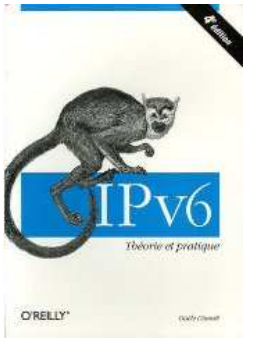
Local		Distant	
adresse	port	adresse	port
194.113.217.1	80	::1 (localhost)	-
194.113.217.1	2310	2001:db8:122:344::1	2310
194.113.217.1	2311	2001:db8:122:344::55	2310

  - Couplé en pratique avec DNS64 :
    - Construction dynamique d'enregistrements AAAA à partir des A

Bibliographie

- Bibliographie:
  - D. Comer *et al.*, "TCP/IP Architecture, protocoles, applications", 4ème édition, Dunod, 2003. ISBN 2-10-008181-0. (~ 57 €)

  - G. Cizault, "IPv6 - Théorie et pratique« , O'Reilly, 4ème édition, 2005. ISBN 978-2841773374. Epuisé, disponible sur : http://livre.g6.asso.fr/



- Sites Internet:
  - RFC: www.ietf.org, www.faqs.org
  - Cisco: www.cisco.com
  - Wikipedia : fr.wikipedia.org *ou* en.wikipedia.org