# La couche 3 du modèle OSI

# La couche Réseau

# Le rôle de la couche Réseau

- Son rôle est de transporter les paquets/datagramme issus de la couche transport d'une extrémité d'un réseau à une autre. De proche en proche cette couche permettra la remise des données de l'utilisateur.

# Fonctionnalités

- Gestion des connexions réseau (pas le cas de IP mais le cas pour X25)
- Gérer l'acheminement des paquets/datagrammes
- Multiplexer les connexions
- Découpage et groupage des paquets/datagrammes
- Gestion des pertes
- Qualité de service

# Services proposés

- Transfert des données
- Adressage
- Routage
- Qualité de service
  - Taux de transfert
  - Gestion des pertes

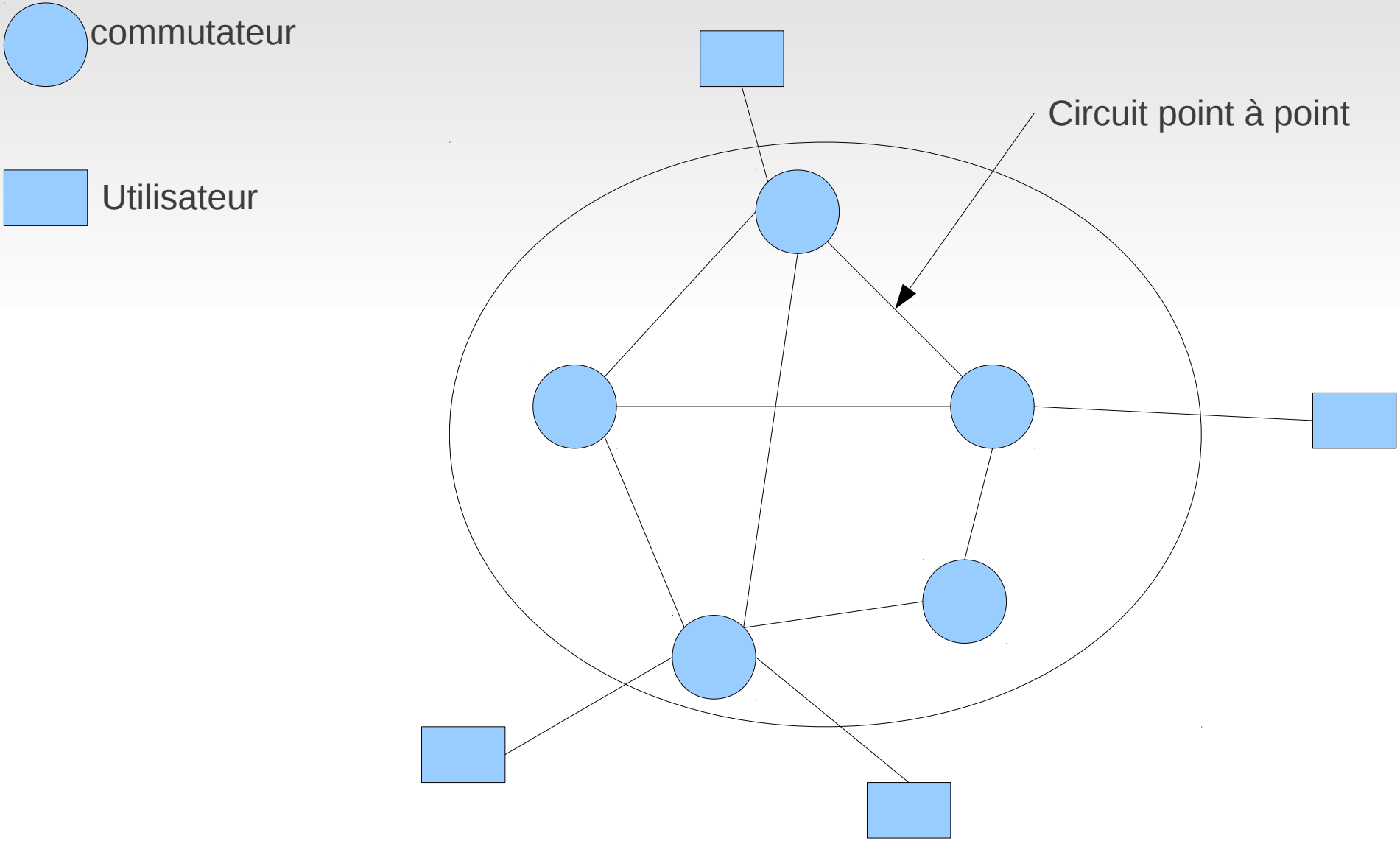
# Avec ou sans connexion

- Avec :
  - Établir et maintenir la connexion
    - Sécurité, séquencement, négociation des paramètres
    - Couteux en communication, pas forcément rentable
    - En général ce mode utilise la commutation et un circuit virtuel
- Sans :
  - Envoi des paquets sans contraintes (donc plus simple mais la connexion est géré par la couche session)
    - Simple, échange rapide
    - Pas de sécurité, difficile de gérer les paramètres
    - Ce mode utilise en général le routage de paquet

# Commutation

- Dans un grand réseau, il n'est pas possible de mettre en relation chaque utilisateurs directement avec les autres
- Pour se faire on utilise des tronçons de communication de manière à relier virtuellement 2 utilisateurs

# Commutation (2)



# Commutation de circuits

- Chaque commutateur dispose de plusieurs supports de transmission
- 2 utilisateurs désirant échanger des informations doivent être reliés par un circuit virtuel au travers du réseau physique :
  - Établissement du circuit
  - Transfert des données
  - Libération du circuit
- Ce circuit virtuel sera monopolisé durant l'échange
- Ex : la téléphonie

# Commutation de messages

- Un message est une suite de données relatives à une application (fichier texte, fichier audio, etc.)
- Le message passe de commutateur en commutateur à travers le réseau
- La mémoire des commutateurs doit être importante pour stocker plusieurs messages en même temps
- Plus le message est important plus les risques de perte sont élevés
- Besoin d'un protocole de liaison de données

# Commutation de paquets

- Un message est d'abord découpé en plusieurs paquets avant d'être expédié de commutateur en commutateur
- Besoin d'information de contrôle sous forme d'en-tête pour l'acheminement et le réassemblage
- Acheminement rapide et moins de risque de perte ou d'erreur
- Besoin de créer un circuit virtuel avant l'envoi des paquets de manière à ce que le flot de données emprunte le même chemin

# Commutation de cellules

- Apparu avec la technologie (Asynchronous Transfert Mode)
- Petit paquet de taille fixe
- Utilise la commutation de paquet et la commutation de circuit

# Routage de paquets

- Un paquet a une taille de 1500 octets maximum
- Un message est découpé en plusieurs paquets
- Les paquets sont ensuite expédiés indépendamment les uns des autres au travers du réseau suivant la meilleure route (qui peut varier suivant le temps)

# Multiplexage

- Lors de l'installation d'une liaison, le cout du support n'est pas le plus important, d'où l'idée d'utiliser des supports ayant de hauts débits et de les utiliser au maximum de leurs capacités
- Lorsque la largeur du signal a émettre est plus faible que la bande passante du support, il est intéressant d'utiliser le support pour plusieurs communications : le multiplexage

# Routage (sens général)

- Détermine le chemin que prendront les données lors de leur transit dans le réseau
- Routage statique : le chemin est toujours le même
- Routage dynamique, le chemin change en fonction de l'état du réseau
- Routage centralisé : un équipement est spécialement utilisé pour gérer le routage
- Routage adaptatif : chaque équipement doit connaître l'état du réseau

# Adressage

- Permet d'identifier de manière unique tous les utilisateurs d'un réseau
- Physique : adresse de l'équipement physique relié au support (ex:MAC)
- Logique : utilisée pour permettre le routage à travers le réseau. Déterminée par l'administrateur du réseau (ex : IP)
- Symbolique : plus facile à mémoriser et à utiliser que les autres ( ex : www.google.fr)

# Quelques protocoles "couche 3"

- X25 (avec connexion, commutation)
- IP (sans connexion, routage)

# X25 (1976)

- Gestions des circuits virtuels
- Gestion des erreurs
- Découpage et assemblage des paquets
- Adressage
- Multiplexage
- Abandon par Orange FT en 2011 … :(

# IP

- Mode non connecté, permet d'interconnecter des réseaux n'ayant pas les mêmes protocoles au niveau transport.
- Version IP4
  - Adressage sur 4 octets
  - Simple remise des paquets
  - Fragmentation des paquets
  - 20 octets d'en-tête

# Format IP4

Format en-tête IP

| mots \ bits | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | Version | | | | Long. en-tête | | | | service | | | | | | | | longueur totale du paquet | | | | | | | | | | | | | | | | |
| 2 | identification | | | | | | | | | | | | | | | | | DF | MF | position du fragment | | | | | | | | | | | | |
| 3 | TTL | | | | | | | | protocole | | | | | | | | checksum | | | | | | | | | | | | | | | |
| 4 | adresse IP source | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 5 | adresse IP destination | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 6 | option(s) + bits à 0 de remplissage | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

version de IP ici 4
longueur de l'entête en terme de mot de 32 bits (8 octets), allant de 5 à 15 mots
Service , détermine la qualité de service demandé
longueur paquet en octets
identification du paquet
DF indique de ne pas fragmenter le paquet
MF indique que le paquet est un fragment et que d'autre paquets vont suivre sinon le paquet est seul ou il est le dernier fragment
TTL est la durée de vie (time to live)
protocole du niveau/couche 4 (6 pour TCP, 17 pour UDP et 1 pour ICMP)

checksum, contrôle de l'intégrité du paquet, complément à 1 du complément à 1 de la somme des éléments de l'en tête

40 octets maximum d'option
bits de remplissage de manière à être multiple de 32 bits (ou 4 octets)
version de IP ici 4

# IP6

- Adressage 128 bits
- Pas de fragmentation
- Gestion des flux
- Options pour ajouter des fonctionnalités
- 40 octets pour l'entete

# Format IPv6

Format en-tête Ipv6

| | bits | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| mots | 1 | version | | | | priorité | | | | | | | | référence | | | | | | | | | | | | | | | | | | | |
| | 2 | longueur en octet sans l'en-tête | | | | | | | | | | | | | | | | protocole | | | | | | | | nombre de saut | | | | | | | |
| | 3 | adresse source | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | 4 | adresse destination | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | 5 | options | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

Version 6
niveau de priorité du paquet
référence : permet d'indiquer à quel flot de données appartient le paquet et de permettre une qualité de service
nombre de saut : indique combien de nœud seront traversé avant la destruction du paquet
options : plus nombreuses et ouvertes, elles permettent d'instaurer la sécurité.
Les options se suivent dans l'en-tête dans un ordre défini. Un nœud peut réagir différemment avec les options

# Adressage IPv4

- Identifiant unique allouée à l'interface d'un élément sur un réseau IP
- Codée sur 32 bits, nomenclature (ecriture) : 4 nombres décimaux (0 à 255) représentant 8 bits séparés par un .
- Ex 134.206.90.4
- On parle d'adresse logique (à la différence d'adresse physique)

# Adresse IP (@IP)

- Elle contient une partie « réseau » et une partie « hôte » qui sont variables en fonction du préfixe de l'adresse qui correspond au masque de l'adresse (le masque s'écrit comme une adresse)
- En binaire, dans le masque, chaque bit à 1 définit un bit appartenant à la partie réseau de l'@ip et chaque bit à 0 définit un bit appartenant à la partie hôte de l'@ip.
- Ex : masque : 255.255.0.0 et @IP : 134.206.90.3, 134.206 est la partie réseau et 90.3 la partie hote

# Explication

- 255 en décimal s'ecrit 11111111 en binaire
- Donc 255.255.0.0 s'ecrit :
  - 11111111.11111111.00000000.00000000 en binaire
  - Les 1 donne la partie réseau et les 0 la partie hote
- Donc 134.206 partie réseau et 90.3 partie hote
- 134.206.90.3 s'écrit en binaire
  - 10000110.11001110.01011010.00000011
- On fait un ET logique entre les 2, on obtient
  - 10000110.11001110.00000000.00000000
- Ce qui donne 134.206.0.0 comme identifiant réseau

# Autre représentation

- Adresse/longueur du préfixe
- Ainsi on écrit l'exemple précédent 134.206.90.3/16
- Autre exemple 134.206.90.3/19 est une adresse IP 134.206.90.3 avec un masque d'adresse : 255.255.224.0

# Bits/adresses réservés

- Les numéros d'hôtes 0 et 255 sont réservés
- Tous les bits hôte à 1 => adresse de diffusion
- Tous les bits hôte à 0 => adresse du réseau
- 127.0.0.0 adresse de rebouclage loopback
- 0.0.0.0 route par défaut
- 127.0.0.1 adresse localhost (utile pour tester le fonctionnement de TCP/IP sur la machine hôte)

# Classes d'adresse (masque par défaut)

- Classe A, 8 bits réseau 24 bits hôte
  - IP commence toujours par 0, 7 bits pour la partie réseau
- Classe B, 16 bits réseaux, 16 bits hôte
  - IP commence toujours par 10
- Classe C, 24 bits réseaux, 8 bits hôte
  - IP commence toujours par 110

# Classes d'adresse (suite)

- Classe D, @ de diffusion de groupe (multicast/ multipoint), commence par 1110
- Classe E, commence par 11110 et sont réservées pour une utilisation future (heu et IP6 ?)

# Adresses privées

- De classe A : 10.xx.xx.xx
- De classe B : 172.xx.yy.yy 16<=xx<=31
- De classe C : 192.168.xx.xx

# Sous réseaux

- Certains bits de la partie hôte peuvent servir à étendre la partie réseau. Ils sont associés au masque de l'adresse pour constituer le masque de sous réseau, on parle de bloc CIDR (Classless Inter-Domain Routing) qui rend obsolète la notion de classe
- Ceci permet de créer de nouveaux réseaux mais réduit le nombre de hôtes possibles
- Ex : classe C : 172.16.1.0/24 offre $2^8 = 256$ hôtes Si on utilise 2 bits pour créer 4 sous réseaux, il reste donc $2^6 = 62$ (64-2) adresses pour chaque sous réseau.

# Sous réseaux (suite)

- Les 4 sous réseaux sont :
  - 176.16.1.0 (172.16.1.00000000)
  - 172.16.1.64 (172.16.1.01000000)
  - 172.16.1.128 (172.16.1.10000000)
  - 172.16.1.192 (172.16.1.11000000)
- On peut utiliser le sous réseau (notation binaire) 00, 01, 11 et 10 (avant la RFC 1878 de 1995, on ne pouvait utiliser que 10 et 01 … )

# Dynamic Host Configuration Protocol

- Permet d'allouer dynamiquement des adresses d'une plage définie à des machines lors de leur connexion au réseau.

# Network Address Translation

- Permet de connecter des machines d'un réseau privé (ayant une adresse privée) à l'Internet adresse publique)
- De manière Statique (sécurité)
- De manière dynamique (résout le problème de manque d'adresse), c'est la passerelle ou le routeur qui gère la translation. L'adressage des hôtes locaux se fait en DHCP avec des adresses privées.

# Fonctionnement de NAT

- Un hote (derrière le routeur/passerelle) fait la demande suivante à son routeur/passerelle
  - (@Ipprivée K, port X) demande à IPpublique I sur le protocole Y sur le port Z.
  - Le routeur passerelle memorise cette demande et la translate comme si elle était pour lui :
    - (@Ippublique J, port P) demande à Ippublique I le protocole Y sur le port Z
  - Il recoit la reponse R de Ippublique I sur son port P
  - Il la translate et l'envoi à @Ipprivée K sur son port X, comme suit :
    - Reponse R de Ippublique I sur le port X

# Adresse MAC (niveau couche 2)

- Adresse Media Access Control
- 48 bits (6 octets) représentation en hexadécimale
  - AA:11:11:AA:11:AA
  - 1 bit : 0 adresse individuelle, ou 1 de groupe
  - 1 bit : 0 adresse universelle ou 1 adresse locale
  - 22 bits réservés pour l'adresse du constructeur sinon tous à 0
  - 24 bits : adresse unique ($2^{24}$ adresses par fabricant)
- Couche 2 du modèle OSI

# Address Resolution Protocol

- Permet à un équipement de connaitre l'adresse physique appelée @MAC (couche 2 du modèle OSI) correspondante à l'adresse logique (@IP) d'un autre équipement
- Fonctionnement
  - Émetteur envoi un message de diffusion (@MAC FF:FF:...:FF, et l'IP destinataire)
  - La machine IP destinataire répond avec sa MAC.
- Existe Reverse ARP ...

# Service IP

- Création et envoi du paquet
- Réception et livraison des données du paquet
- Routage de paquet
  - Routeur interconnecte au moins 2 réseaux IP
  - Existence d'une table de routage
    - Destinataire – routeur – interface
- Contrôle d'erreur Internal Control Message Protocol

# Envoi d'un paquet

- La couche IP reçoit un segment ou un datagramme de la couche transport
  - Elle l'encapsule dans un paquet (segment/datagramme + en-tête)
  - Elle l'envoie à la couche Liaison de donnée en indiquant
    - Soit le destinataire s'il est sur le même réseau
    - Soit le routeur si le destinataire est extérieur au réseau

# Réception d'un paquet

- La couche 3 reçoit un paquet de la couche 2
- Elle le décapsule (enlève l'en-tête) et remet à la couche 4 le segment ou le,datagramme que contenait le paquet

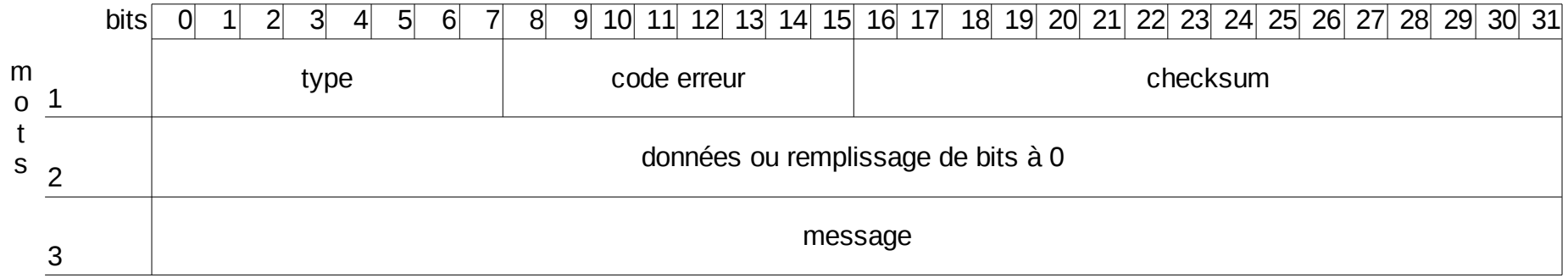
# ICMP

- Messages de contrôle au format paquet, les données ICMP sont insérées en place du segment/datagramme
- Initialement destiné au routeur pour informer des erreurs de transmission
- Utilisé pour d'autre application
- Contient un champ précisant l'erreur

# Exemple d'utilisation de ICMP

- Traceroute (type 11)
- Ping (type 8) la réponse est de type 0
- Non délivrance de paquet, envoie d'un message ICMP de type 3

Format ICMP




type de message ICMP
code erreur, précise certains types de message ICMP
ex : type 3 : destinataire inaccessible
ex de code : 2/3, protocole/port non joignable
données : utilisé pour les messages de type réponse
message : utilisé par exemple dans le cas de redirection de paquet

# ping

```
ludovic@ludovic-IGI:~$ ping 83.198.80.1
PING 83.198.80.1 (83.198.80.1) 56(84) bytes of data.
64 bytes from 83.198.80.1: icmp_req=1 ttl=255 time=25.8 ms
64 bytes from 83.198.80.1: icmp_req=2 ttl=255 time=27.2 ms
^C
--- 83.198.80.1 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 25.824/26.555/27.287/0.749 ms
ludovic@ludovic-IGI:~$ ping www.google.fr
PING www.l.google.com (209.85.227.104) 56(84) bytes of data.
64 bytes from wy-in-f104.1e100.net (209.85.227.104): icmp_req=1 ttl=53 time=54.8 ms
64 bytes from wy-in-f104.1e100.net (209.85.227.104): icmp_req=2 ttl=52 time=41.7 ms
^C
--- www.l.google.com ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 41.791/48.340/54.889/6.549 ms
ludovic@ludovic-IGI:~$
```

# ping

```
ludovic@ludovic-IGI:~$ ping 193.253.89.189
PING 193.253.89.189 (193.253.89.189) 56(84) bytes of data.
From 80.10.122.77 icmp_seq=1 Packet filtered
From 80.10.122.77 icmp_seq=2 Packet filtered
```

# traceroute

```
ludovic@ludovic-IGI:~$ traceroute www.google.fr
traceroute to www.google.fr (209.85.227.103), 30 hops max, 60 byte packets
 1  ALille-156-1-9-1.w83-198.abo.wanadoo.fr (83.198.80.1)  26.200 ms  28.843 ms  34.228 ms
 2  10.125.76.146 (10.125.76.146)  36.943 ms  38.389 ms  40.556 ms
 3  ae20-0.nclil102.Lille.francetelecom.net (193.253.89.193)  41.998 ms  43.737 ms  45.206 ms
 4  xe-3-3-0-0.nilil102.Lille.francetelecom.net (193.252.100.238)  46.892 ms  48.627 ms  50.096 ms
 5  193.252.100.226 (193.252.100.226)  59.680 ms  59.673 ms  69.470 ms
 6  google-17.GW.opentransit.net (193.251.254.202)  61.784 ms  30.341 ms  30.828 ms
 7  66.249.94.78 (66.249.94.78)  112.531 ms 66.249.94.76 (66.249.94.76)  35.114 ms  37.062 ms
 8  66.249.95.173 (66.249.95.173)  48.746 ms 72.14.232.134 (72.14.232.134)  50.631 ms 66.249.95.173
(66.249.95.173)  52.379 ms
 9  209.85.251.231 (209.85.251.231)  52.800 ms  55.204 ms 72.14.236.191 (72.14.236.191)  56.424 ms
10  209.85.243.93 (209.85.243.93)  70.476 ms 209.85.243.97 (209.85.243.97)  68.977 ms 209.85.243.93
(209.85.243.93)  70.642 ms
11  wy-in-f103.1e100.net (209.85.227.103)  65.261 ms  40.151 ms  43.322 ms
ludovic@ludovic-IGI:~$
```

# Internet Group Management Protocol

- Un utilisateur souhaitant appartenir ou quitter un groupe (groupe de diffusion participant à une même activité) utilise (en fait sa machine hôte) le protocole IGMP. Ce protocole traverse les nœuds du (des) réseau(x).

# Protocoles de signalisation

- Permettent au niveau de la couche 3 de mettre en oeuvre un quasi circuit virtuel donc un fonctionnement en mode commuté, ce qui offre des service jusque là inexistants (VoiceIP).

- Exemple de ces protocoles : Ressource reSerVation Protocol, qui averti les nœuds d'une route de l'arrivée d'un flot de données.

# Protocoles de signalisation

- Real Time Protocol, qui permet les applications temps réel. Il utilise des mixeurs qui regroupe plusieurs flots de plusieurs applications en un seul, ainsi que des translateur qui encode les données dans un format plus « transportable ».

- Real Time Control Protocol, associé à RTP pour la partie controle

# Sécurité

- Il existe des mécanismes de filtrage sur les équipements de routage sous forme de Access Control List.
- Il existe des travaux permettant un chiffrement des échanges.

# Access Control List

Gère les listes des paquets autorisés et interdites par un routeur, un pare feu, une passerelle. Il existe 2 catégories :

- standard, contrôle @IP source et une partie de l'@ IP source
- étendue, contrôle quasiment tous les champs des en-tête TCP, IP et UDP

# ACL notion de masque

Les masques permettent de traiter plusieurs @IP en une seule règle. Le masque définit la partie de l'@ à vérifier. Ex :

Le masque 0.0.255.255 indique de ne vérifier que les 2 premiers octets d'une adresse

Permit 134.206.5.0 masque 0.0.0.255 , accepte toutes les @IP commençant par 134.206.5

# Exemple d'ACL standard

Access-list 1 Permit 134.206.3.1 0.0.0.0

Autorise tous les paquets de cette @

Access-list 2 Permit 0.0.0.0 255.255.255.255

Accepte toutes les @IP

Permit host 134.206.90.1

Accepte juste cette @ip 134.206.90.1

# Exemple d'ACL étendue

access-list number { deny | permit } protocol source @ destination @

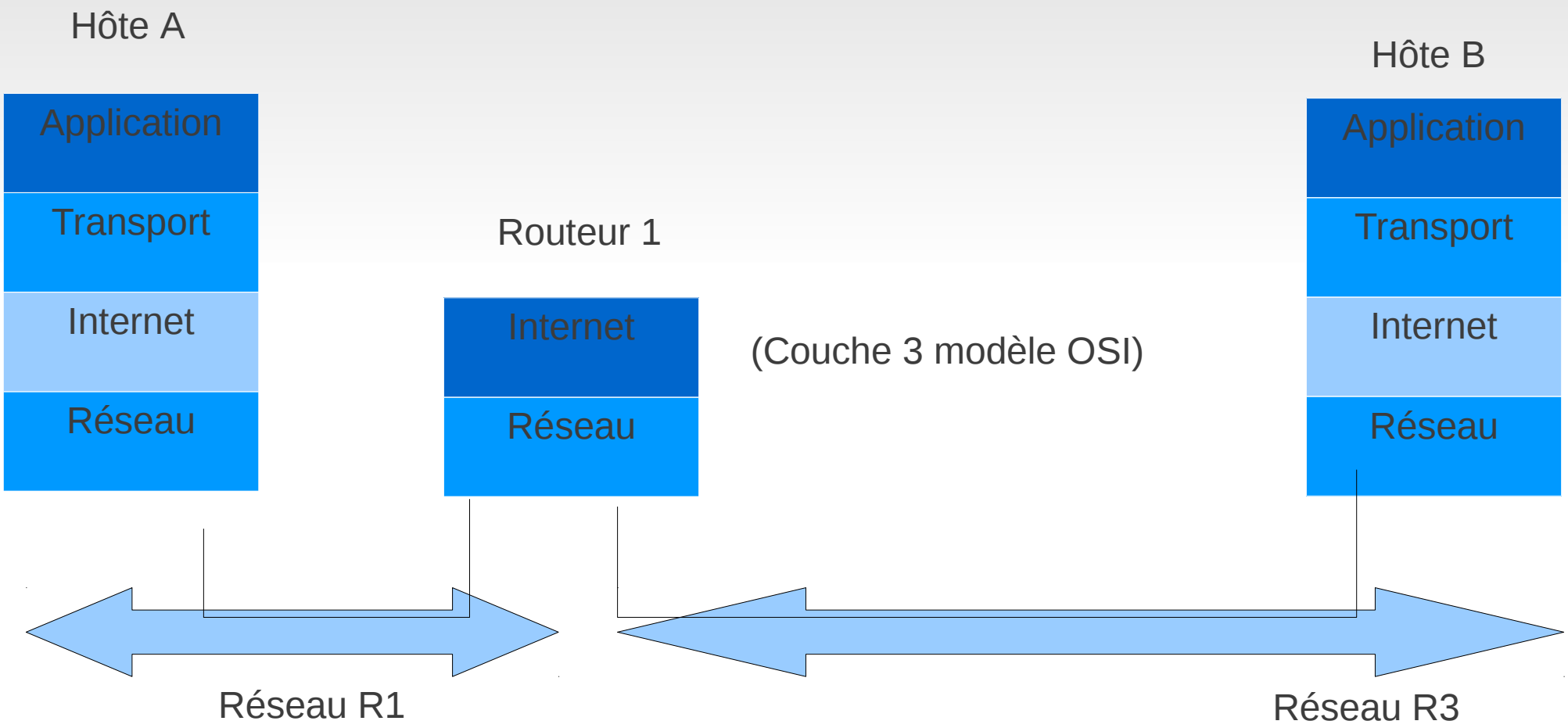
access-list 1 deny udp any gt 23 host 10.1.1.1 eq 21

Refuse tous les paquets udp émis depuis un port 23 de quelques machines que ce soit vers le port 21 de la machine 10.1.1.1

# Routage IP

- Intervient à 2 niveaux :
  - De la machine hôte
  - Des routeurs

# Routage des paquets



# But du routage

- Déterminer un chemin pour les paquets, entre l'émetteur et le récepteur, à travers les réseaux.
- Pour se faire les routeurs doivent connaître constamment l'état du réseau, ces informations sont mémorisées dans la table de routage
- Les routeurs échangent donc des informations afin de maintenir leur table de routage à jour
- Ces échangent d'informations sont possibles grâce à des protocoles de routage

# Table de routage

- Permet à l'équipement/nœud de déterminer pour chaque destination, la route (le prochain équipement/nœud) à qui transmettre le paquet
- Pour un ETTD, la table de routage contient la route de son réseau (routage direct), la route de son routeur (pour le routage indirect) ainsi qu'une route par défaut (0.0.0.0)

```
ludovic@ludovic-IGI:~$ route -n
Table de routage IP du noyau
Destination     Passerelle      Genmask         Indic Metric Ref    Use Iface
92.131.43.1     0.0.0.0         255.255.255.255 UH    0      0        0 ppp0
169.254.0.0     0.0.0.0         255.255.0.0     U     1000   0        0 ppp0
0.0.0.0         92.131.43.1     0.0.0.0         UG    0      0        0 ppp0
```

# Cache de routage

```
ludovic@igi-XPS-M1330:~$ route -Cn
cache de routage IP du noyau
Source          Destination     Passerelle      Indic Metric Ref    Use Iface
90.47.253.202   74.125.45.16    74.125.45.16            0      2        9 ppp0
74.125.157.16   90.47.253.202   90.47.253.202   il      0      0      358 lo
65.55.71.95     90.47.253.202   90.47.253.202   il      0      0      797 lo
74.125.45.16    90.47.253.202   90.47.253.202   il      0      0     1144 lo
194.254.131.165 90.47.253.202   90.47.253.202   il      0      0        5 lo
90.47.253.202   193.251.214.116 193.251.214.116         0      2      151 ppp0
193.251.214.116 90.47.253.202   90.47.253.202   il      0      0     1975 lo
90.47.253.202   193.50.192.7    193.50.192.7            0      0        0 ppp0
90.47.253.202   194.254.131.165 194.254.131.165         0      2       13 ppp0
90.47.253.202   65.55.71.95     65.55.71.95             0      1        3 ppp0
90.47.253.202   74.125.157.16   74.125.157.16           0      1        1 ppp0
```

# Table de routage (2)

- Pour un ETCD, elle contient :
  - Les routes directes
  - Les routes indirectes issues d'un protocole de routage
  - Les routes de sous-réseaux
  - Une route par défaut (sauf les routeurs de la dorsale d'Internet)
- Il est difficile de maintenir une table de routage à jour !!

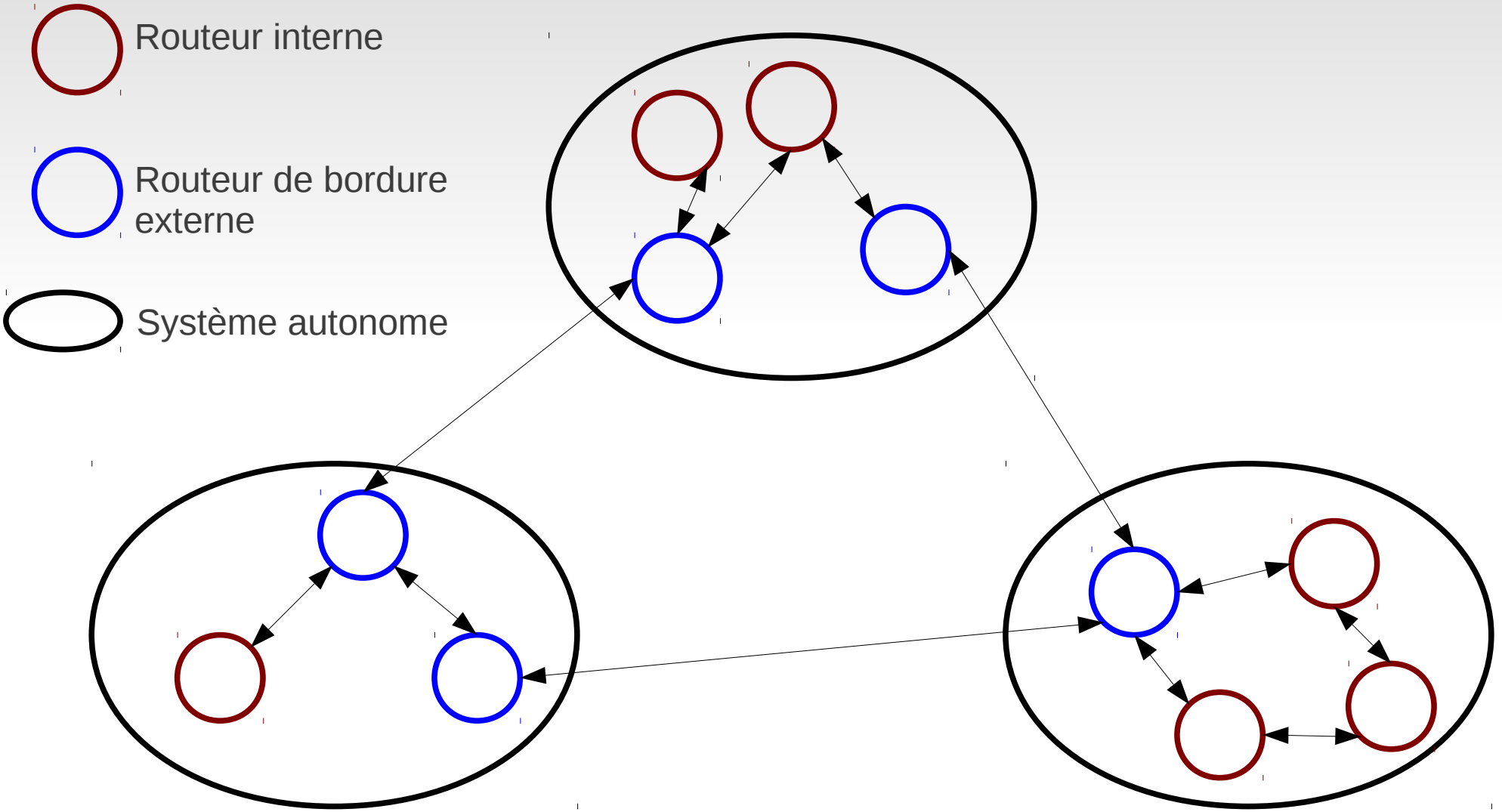
# Complément

Une destination n'est pas toujours une adresse IP complète mais parfois un préfixe.
Pour une même destination il peut y avoir plusieurs routes, le choix par défaut est la plus directe.
S'il n'y a pas de route par défaut, soit il y a destruction du paquet ou le routeur possède une table complète. Dans ce cas il connait toutes les destinations possibles, il fait alors partie de la dorsale d'Internet.

# Systèmes autonomes

Vu la taille croissante d'Internet, il a été décidé de le scinder en un ensembles de systèmes autonomes reliés entre eux.
Un AS est composé de réseaux et de routeurs reliés entre eux partagent le même protocole de routage.
Ce découpage permet de limiter la taille des tables de routage.

# Exemple d'AS


Routeur interne
Routeur de bordure externe
Système autonome

# Algorithme de routage

- Permet de calculer la table de routage en évitant les boucles.
- On peut utiliser 3 types d'algorithme de routage :
  - à vecteur de distance
  - à état de liens
  - à vecteur de chemin
- Ces différents algorithmes sont utilisés dans différents protocole de routage

# Vecteur de distance

- Chaque nœud échange des informations avec ses voisins auxquels il est directement connecté.
- Ces échangent permettent au nœud de mettre à jour sa table de routage
- Les informations transmises sont les reseaux que peut atteindre le noeud ainsi que la métrique associé (dans ce cas le nombre de saut)
  - Lourd dans le cas de grand réseau
  - Métrique restreinte

# États de liens

- Chaque nœud diffuse à travers l'AS la métrique de toutes les connexions directes qu'il a.
- Chaque nœud a donc une connaissance globale du réseau de l'AS
- Le nœud peut donc construire sa table de routage et/ou la mettre à jour
  - Méthode de diffusion couteuse
  - Possibilité de métriques différentes sur différents AS

# Vecteur de chemin

- Construction de la table de routage en tenant compte des réseaux ou AS à traverser pour atteindre une destination.
- Le nœud connait donc le chemin vers chaque destination
- Pas de notion de métrique

# Protocole de routage

- Permet de fournir les informations nécessaires à l'établissement de la table de routage et donc permettre le routage des informations.
- On distingue les protocoles utilisés par les routeurs à l'intérieur des AS et les protocoles utilisés par les routeurs aux bordures de plusieurs AS (qui peut aussi utiliser un protocole interne)

# IGP

- Différents types de protocoles :
  - À vecteur de distance, le routage se fait de proche en proche, chaque routeur transmet ses meilleures routes
  - A états de lien, les routeurs s'échangent la totalité des informations de routage
  - hybride des 2 premiers

# Routing Information Protocol

Protocole IP de type « à vecteur de distance ». Chaque routeur communique la distance (en terme de nombre de sauts) qui le sépare d'un(des) réseau(x) aux autres routeurs. Un routeur qui reçoit ce message met à jour sa table de routage (en ajoutant + 1 aux distances recues) en choisissant la route la plus courte (en ne dépassant pas le nombre se saut limité à 15). S'il y a modification de sa table de routage, il envoie un message à ses voisins en respectant un delai de mise à jour.

Un routeur envoie un message toutes les 30 sec à ses voisins.

Si au bout de 180 sec un voisins n'émet plus, on le considère inactif

Si au bout de 240sec un voisin n'émet plus on détruit son entrée dans la table de routage

# Interior Gateway Routing Protocol

Protocole développé par CISCO basé sur les vecteurs de distance. Permet de dépasser les limites de RIP en autorisant pour une même route plusieurs métriques : bande passante, la charge, le MTU, le délai et la fiabilité. De plus le nombre de saut est limité à 255, le délai passe de 30 à 90 secondes. Il détecte les bouclages.

# Enhanced Interior Gateway Routing Protocol

Successeur de IGRP, EIGRP est hybride. Il rend plus stable le routage face au changement de topologie, à la capacité de la bande passante et à la capacité du processeur du routeur.
Permet le load balancing sur des métriques différentes.

# Open Short Path First

Protocole à états de liens. Chaque routeur maintien des relations avec ses voisins (hello message). A intervalle régulier il transmet cette liste de relations à tout le réseau de proche en proche par le biais des Links State Advertisements. Ces LSA forment la Link State DataBase. Chaque routeur du réseau pouvant ainsi établir une route la plus courte vers tous les réseaux de la LSDB grâce à un algorithme de Short Path First. Les LSA sont renvoyées s'il y a changement de topologie.

# OSPF

Il utilise une métrique directement liée à la bande passante. Il permet de faire du load balancing sur des métriques égales.
Afin de limiter la diffusion des LSA, on peut diviser le réseau en différentes aires mais en maintenant toutefois une arête dorsale à laquelle toutes les aires doivent être reliées directement.

# OSPF

Dans le cas d'un réseau à diffusion (comme Ethernet) ou tout le monde envoie à tout le monde, afin de limiter la diffusion des LSA, il est déterminer des routeurs désignés (DR) qui reçoit les LSA (et un autre de secours BDR) et les retransmet aux autres routeurs. L'élection du routeur désigné se fait en fonction de sa priorité et de son IP. En cas d'ajout d'un routeur, il n'y a pas de réélection Une réélection est due à la panne d'un DR ou BDR.

# Protocole de routage externe

Ils permettent d'échanger les informations de routage entre les systèmes autonomes. Les routeurs frontaliers ou externes transmettent/échangent les informations d'accessibilité qui sont la liste des sous-réseaux au sein de leur réseau autonome

# Exterior Gateway Protocol

Ancien protocole utilisé pour dénommer ce type de protocole. Le fonctionnement est le suivant :
- échange de message hello et i heard you
- suite à une découverte d'un voisin, un routeur fait un poll
- le voisin répond par un update avec les informations d'accessibilité
- lors de la réception d'un update, le routeur met à jour sa table de routage
- au bout de 3 poll, un voisin est déclaré HS

EGP ne donne pas la meilleure route !!!

# EGP

La gestion de la meilleure route est laissé au systèmes autonome suivant ces propres critères.

EGP repose sur un ensemble de routeur de cœur qui maintiennent les routes de tous les systèmes autonome. Cependant avec l'explosion de l'Internet, ces routeurs n'ont plus été capables de traiter ces informations.
Pas de détection des boucles (exemple de boucle triangulaire à 3).

Protocole de type vecteur de distance !

# Border Gateway Protocol

Utilisé sur Internet, repose sur un mécanisme de connexion par TCP (port 179), il permet un routage politique (raison non techniques).
Protocole à base de vecteur de chemin !

Fonctionnement :
- connexion des voisins (peer)
- échange de message OPEN
- échange de message update (un chemin par message) seulement en cas de besoin (BGP optimise la bande passante)

# IP Mobile

Le principe est de permettre à des utilisateurs mobile de se détacher de leur adresse. On parle de handover.
Pour cela chaque utilisateur mobile est rattaché à un sous réseau avec une adresse fixe qui dispose d'un Home Agent (le routeur de son réseau fixe) qui va assurer le relai vers un Foreign Agent (le routeur du réseau ou est situé l'utilisateur mobile.
On suppose que l'adressage de message vers un utilisateur est toujours fait sur son adresse fixe.

# Fonctionnement

- Dès que la station est connectée à un sous réseau :
  - Elle détecte l'agent (FA ou HA) et elle s'y connecte.
  - S'il s'agit de son HA c'est terminé, sinon
  - Le FA assure le rôle d'agent (ou peut dédier la tâche à un routeur de son sous-réseau)
  - Le FA ou le routeur dédié établit la communication avec le HA et crée un tunnel
  - Le handover de niveau IP est établit
  - La station reçoit ses messages de manière transparente par le tunnel, l'envoi de message se fait par le FA (ou le routeur dédié)

# Détection des Agents (HA ou FA)

- Se fait par un message ICMP Router Discovery Protocol envoyés par les agents.
- Envoi fait sur une adresse particulière (pour que la station puisse les recevoir)
- Une station peut initier un message IRDP

# Tunnelling

- Mécanisme d'encapsulation qui permet de connecter 2 réseaux distants
- L'émetteur encapsule un paquet IP dans un autre paquet IP et l'envoie à destination du récepteur
- Le récepteur quand à lui décapsule le paquet initial et le traite