Routage

Yves Caniou (yves.caniou@ens-lyon.fr)

M1 IF11

2008-2009

Plan du module (rappel)

3 intervenants pour 5 parties

- Yves Caniou : partie Routage, Multicast, Pair-à-pair
- Florent Dupont : partie Codage
- Isabelle Guérin-Lassous : partie Adhoc et mobilité

Objectifs de cette partie

Cours

- Connaître
  - la terminologie
  - les problèmes inhérants aux réseaux
  - les mécanismes de routage

TDs

- Connaissance théorique des algorithmes de routage

TPs

- Configuration du matériel réseau
- Maîtrise pratique des mécanismes de routage dynamique avec
  - RIP
  - OSPF

Références pour cette partie

Ouvrages de référence

- Fondé sur le cours de F. Suter (`http://www.loria.fr/~suter/pages/enseign/index.html`)
- Fondé sur le cours de Jacques Bonneville (`http://www710.univ-lyon1.fr/~bonnev/`)
- *Les réseaux*
  Pujolle, Eyrolles.
- *Computer Networking: A top-Down Approach Featuring the Internet.*
  Jim Kurose and Keith Ross. Addison-Wesley.

Plan du cours

Première partie

Introduction

Objectifs

- Grandes lignes et terminologie
- Approche : exemple d'Internet

Aperçu

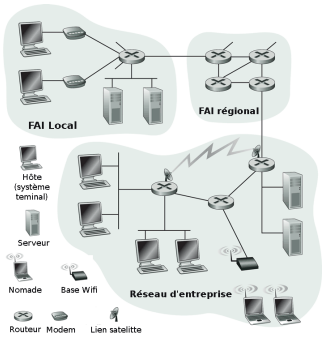
- Qu'est-ce qu'Internet ?
- Les *bords* du réseau
- Le cœur du réseau
- Performances : pertes, délais
- Couches de protocoles, modèles de services
- Modèle de réseau

Première partie

Introduction

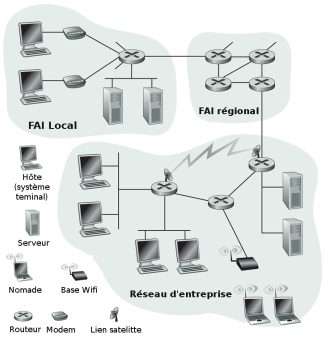
Internet : une vue d'ensemble - 1/2

- Des millions d'ordinateurs connectés
  - Hôtes ou systèmes terminaux
  - Exécutent des applications réseau
- Des liens de communication
  - Cuivre, fibre optique, radio, satellite
  - Taux de transfert = bande passante
- Des routeurs
  - Transmettent des paquets (morceaux d'information)



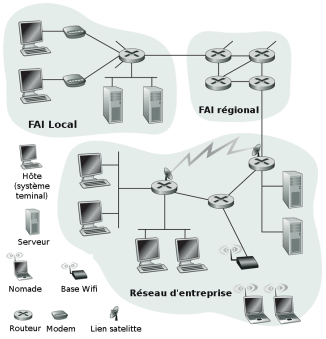
Internet : une vue d'ensemble - 2/2

- Des protocoles contrôlant l'émission et la réception de messages
  - Ex. : TCP, IP, HTTP, FTP, PPP
- Internet : le réseau des réseaux
  - Faiblement hiérarchique
  - Public vs. Intranet privé
- Standards Internet
  - RFC : Request for comments
  - IETF : Internet Engineering Task Force



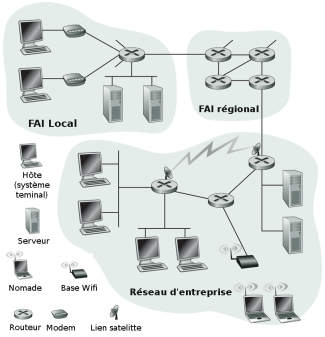
Internet : une vue orientée services

- Infrastructure de communication
  - → Applications distibuées
  - Web, email, jeux en réseau, partage de fichiers (ogg, mp3, DiVX)
- Services de communication offerts aux applications
  - Non-connecté non-fiable
  - Connecté fiable



Internet : objectifs

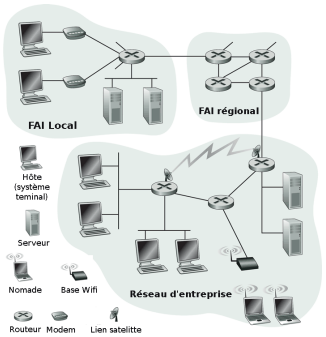
- Connectivité totale
  - Tout le monde communique avec tout le monde
- Coût acceptable
  - Réseau n'est pas un graphe complet
  - → Mutualiser les ressources
- Réseaux à commutation
- Qualité de service
  - Raisons techniques (voix, vidéo)



Plan

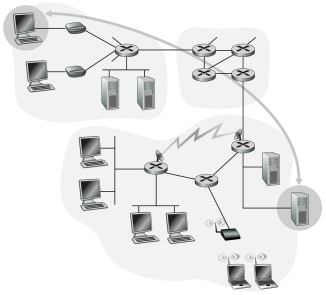
Zoom sur la structure du réseau

- Bords du réseau
  - Applications et hôtes
- Cœur du réseau
  - Routeurs
  - Réseau des réseaux
- Réseau d'accès
  - Liens de communication



Les bords du réseau

- Systèmes terminaux (hôtes)
  - Exécutent les programmes applicatifs
  - Ex. : Web, email
  - Au "bord du réseau"
- Modèle client/serveur
  - Les hôtes client émettent des requêtes et reçoivent des services de serveurs toujours actifs
  - Ex. : Navigateur/Serveur Web, Client/Serveur Mail
- Modèle pair-à-pair
  - Utilise peu (ou pas) de serveurs dédiés
  - Ex. : Gnutella, KaZaA, Skype



Service orienté connexion – 1/2

- Norme ISO 7498
- Objectif : transfert de données entre systèmes terminaux
  - Poignée de main : établissement du dialogue
    - Le « bonjour, bonjour » du protocole humain
    - Fixe l' « état » des hôtes communiquants
- Avantages : sécurisation du transport de l'information, QOS
- Inconvénients : lourdeurs, accès multipoints
- Exemples
  - HDLC (niveau 2) - High-level Data Link Control
  - X.25 ou ISO 8208 (niveau 3)
  - TCP - Transmission Control Protocol
    - Service orienté connexion d'Internet

Service orienté connexion – 2/2

- Service TCP [RFC 793]
- Transfert de flux d'octets fiable et ordonné
  - Si perte : accusés-réception et retransmissions
- Contrôle de flot
  - L'émetteur ne sature pas le récepteur
- Contôle de congestion
  - Réseau congestionné : l'émetteur "ralentit son taux d'émission"

Service non connecté

- Objectif : transfert de données entre systèmes terminaux
  - Comme avant
- Avantages : intéressant pour messages courts
- Inconvénients : précautions du gestionnaire réseau
- Exemples
  - LLC 1 (niveau 2) - Logical Link Control
  - Norme Internet ISO (niveau 3)
  - UDP - User Datagram Protocol [RFC 768]
    - Non connecté
    - Transfert non fiable
    - Pas de contrôle de flot
    - Pas de contrôle de congestion

Exemple d'application

Applications TCP

- HTTP (Web), FTP (transfert de fichiers), Telnet (connexion à distance), SMTP (email)

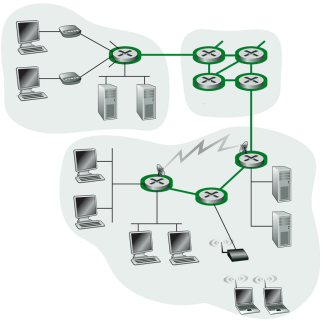
Applications UDP

- Flux multimédia, vidéo conférence, DNS, téléphonie sur Internet

Plan

Le cœur du réseau

- Interconnexion de routeurs
- Question fondamentale : Comment sont transférées les données sur le réseau ?
  - Commutation de circuits : un circuit dédié par appel (Téléphone)
  - Commutation de messages : historiquement, mail et news sur UUCP
    Tout est recopié à chaque saut
  - Commutation de paquets : données envoyées sur le réseau en morceaux
  - Commutation de cellules

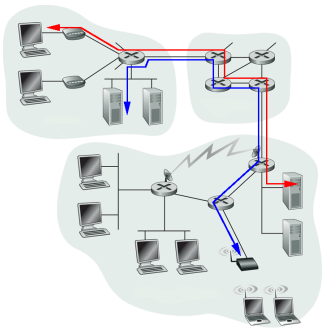


Commutation de circuits

- Ressources réservées de bout en bout pour un "appel" à l'aide de commutateur
- Établissement de l'appel nécessaire
- Chemin doit rester établit durant toute la communication

→ Pas de partage (ressources dédiées)

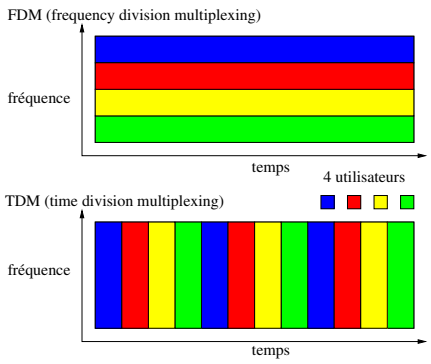
→ Performances garanties



Commutation de circuits

- Ressources réseau (ex. : bande passante) découpées en morceaux
  - Morceaux alloués aux appels
  - Ressource inactive si non utilisée par l'appel (pas de partage)
- Deux manières de découper
  - Par fréquence
  - Par temps (téléphonie)

Commutation de circuits : FDM et TDM



Application numérique

Question

- Combien de temps faut-il pour envoyer un fichier de 640 000 bits de l'hôte A vers l'hôte B sur un réseau à commutation de circuit ?
  - Tous les liens ont une bande passante de 1,536 Mbps
  - Chaque lien utilise TDM avec 24 slots/sec
  - Il faut 500 ms pour établir le circuit de bout en bout

Réponse (à compléter)

- Débit réel du circuit : . . .
- Temps pour transférer le fichier : . . .
- Temps total : . . .

Application numérique

Question

- Combien de temps faut-il pour envoyer un fichier de 640 000 bits de l'hôte A vers l'hôte B sur un réseau à commutation de circuit ?
  - Tous les liens ont une bande passante de 1,536 Mbps
  - Chaque lien utilise TDM avec 24 slots/sec
  - Il faut 500 ms pour établir le circuit de bout en bout

Réponse

- Débit réel du circuit : $1.536\text{Mbps}/24 = 64\text{kbps}$
- Temps pour transférer le fichier : $640000/64000 = 10$ secondes
- Temps total : $10 + 0.5 = 10.5$ secondes

Autre application numérique

Question

- Combien de temps faut-il pour envoyer un fichier de 640 000 bits de l'hôte A vers l'hôte B sur un réseau à commutation de circuit ?
  - Tous les liens ont une bande passante de 1,536 Mbps
  - Chaque lien utilise FDM avec 24 canaux/fréquence
  - Il faut 500 ms pour établir le circuit de bout en bout

Réponse

- D'après vous ? ?

Réponse

- C'est pareil que pour TDM !

Commutation de paquets : service niveau 3 orienté connexion

- Tous les paquets suivent le même chemin : un label (d'appartenance à un flux) suffit pour l'identification ; **notion de circuit virtuel** : X25, Frame Relay, MPLS, etc.
- Circuit virtuel permanent : ressource toujours disponible
- Circuit virtuel dynamique : paquet d'ouverture pour déterminer le chemin, paquet de confirmation qui alloue les ressources au retour (souvent prévu, par toujours mis en œuvre - FR)

Commutation de paquets : service niveau 3 non orienté connexion

- Chaque paquet d'une communication est routé sur la base de sa destination et de l'état du réseaux (table de routage) ;
- Principe fondateur de l'Internet : best effort !
- Adaptation rapide aux changement d'état du réseaux, mais risque d'instabilité

Commutation de cellules

- Dans un circuit virtuel, si tous les paquets sont de même taille (et généralement de petite taille), alors on parle de commutation de cellule comme pour ATM (53 octets).
- L'unicité des tailles permet des optimisations, et *a priori*, diminue les besoins de stockage intermédiaire.
- Les ressources doivent être réservées (ouverture d'un CV)

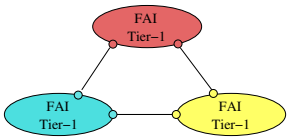
Problèmes

- Définition des critères : vitesse, débit, coût, sureté,...
- Définition de l'adressage : norme X121, IPv4, IPv6
- Qualité de service : que doit-on garantir ?
- Les réseaux sont des systèmes dynamiques, il faut s'adapter

Plan

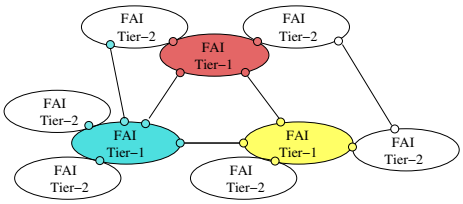
Structure d'Internet : Réseau des réseaux

- Vaguement hiérarchique
- Centre : FAI "tier-1" → couverture nationale ou internationale
  - Se parlent d'égal à égal



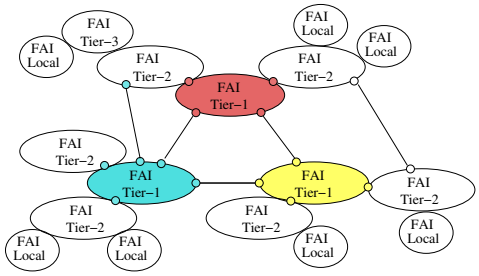
Structure d'Internet : Réseau des réseaux

- FAI "tier-2" : couverture régionale
  - Connectés à un ou plusieurs FAI tier-1
  - Et parfois à d'autres FAI tier-2



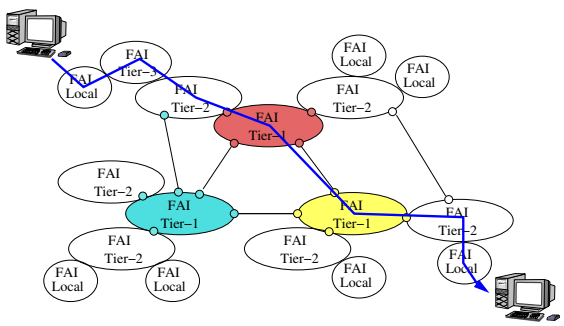
Structure d'Internet : Réseau des réseaux

- FAI "tier-3" locaux
  - Clients de FAI tier-2 qui les connectent à Internet
  - Plus proches des systèmes terminaux



Structure d'Internet : Réseau des réseaux

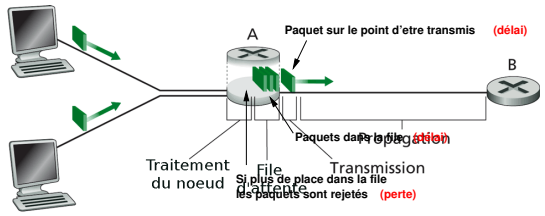
- Chemin d'un paquet
  - Traverse de nombreux réseaux



Plan

Comment perd-on des paquets et du temps ?

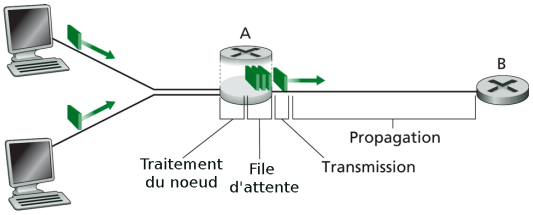
- Les paquets font la queue sur le routeur
- Taux d'arrivée >> capacité de sortie du lien
- Chacun attend son tour



Quatre sources de retard

1. Traitement du nœud

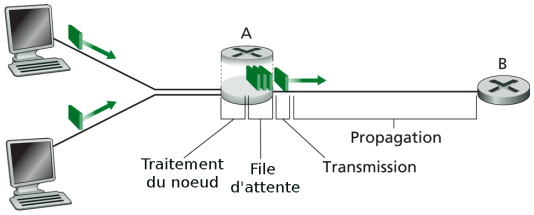
- Contrôle d'erreurs
- Détermination du lien de sortie



Quatre sources de retard

2. File d'attente

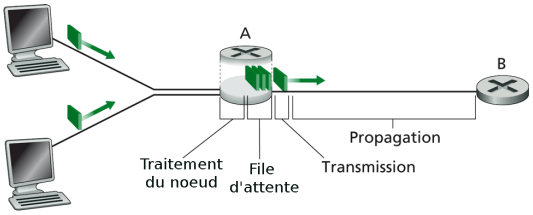
- Temps d'attente pour transmission sur le lien de sortie
- Dépend du niveau de congestion du routeur



Quatre sources de retard

3. Délai de transmission

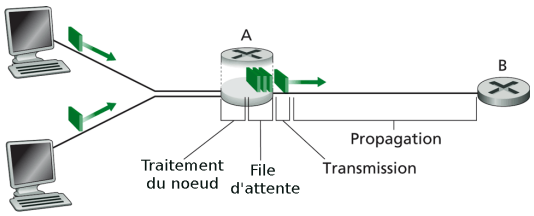
- R = bande passante (bps), L=Taille du paquet (bits)
- Temps d'envoi des bits sur le lien = L/R



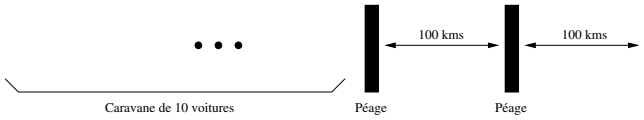
Quatre sources de retard

4. Délai de propagation

- d = longueur du lien, s=vitesse de propagation (2×10^8 m/sec)
- Délai de propagation = d/s



Analogie : caravane de voitures

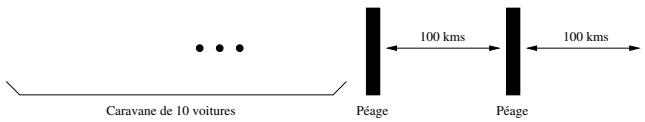


- voiture = bit et caravane = paquet
- Les voitures se "propagent" à 100 km/h
- Chaque péage prend 12 secondes pour faire passer une voiture (temps de transmission)

Question

Combien de temps faut-il pour que la caravane soit alignée devant le second péage ?

Analogie : caravane de voitures

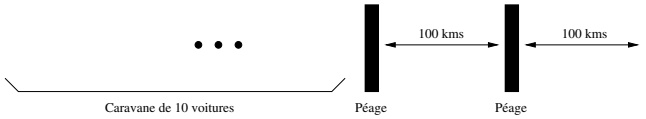


- voiture = bit et caravane = paquet
- Les voitures se "propagent" à 100 km/h
- Chaque péage prend 12 secondes pour faire passer une voiture (temps de transmission)

Réponse : 62 minutes

- Temps pour faire passer toute la caravane à travers le péage : $12 \times 10 = 120 sec$.
- Temps pour que la dernière voiture se propage jusqu'au second péage : 100 km / (100 km/h) = 1h

Analogie : caravane de voitures (suite)

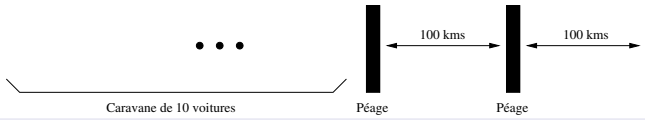


- Les voitures se "propagent" maintenant à 1000 km/h
- Chaque passage de péage prend maintenant 1 minute

Question

Des voitures arriveront-elles au deuxième péage avant que les toutes les voitures aient passées le premier ?

Analogie : caravane de voitures (suite)



- Les voitures se "propagent" maintenant à 1000 km/h
- Chaque passage de péage prend maintenant 1 minute

Réponse : Oui

- Après 7 mn la première voiture est au second péage et il en reste 3 au premier

Conclusion

Le premier bit d'un paquet peut arriver au second routeur avant que le paquet soit entièrement transmis par le premier routeur !

Délai d'un nœud

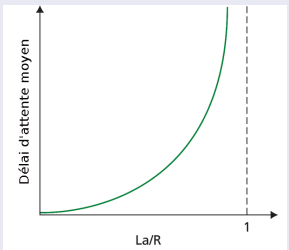
$$d_{noeud} = d_{trait} + d_{att} + d_{trans} + d_{prop}$$

- d_{trait} = délai de traitement
  - Généralement quelques microsecondes ou moins
- d_{att} = délai d'attente
  - Dépend de la congestion
- d_{trans} = délai de transmission
  - L/R, important pour les réseaux peu rapides
- d_{prop} = délai de propagation
  - de quelques microsecondes à plusieurs centaines de millisecondes

Délai d'attente

- R : bande passante (bps)
- L : taille du paquet (bits)
- a : taux moyen d'arrivée des paquets

→ Intensité du traffic = La/R



- La/R proche de 0 : délai court
- La/R → 1 : le délai augmente
- La/R > 1 : plus de travail arrive qu'il peut en être fait, le délai est infini !

Perte de paquets

- File d'attente (ou buffer) précédant le lien dans un routeur
  → Capacité finie
- Arrivée d'un paquet dans une file pleine
  → Paquet rejeté (ou perdu)
- Retransmission possible des paquets perdus
  - Par le nœud précédent
  - Par l'hôte source
  - Pas de retransmission

Vrais routes et délais sur Internet

- A quoi ressemblent de "vrais" délais et pertes sur Internet ?
- Programme `traceroute` → délais point-à-point (routeur à routeur) de la source à la destination. Pour tout i :
  - Envoie 3 paquets qui vont atteindre le routeur i sur le chemin vers la destination
  - Le routeur i renvoie ces paquets l'émetteur
  - qui mesure le temps entre envoi et réception

Vrais routes et délais sur Internet

`traceroute` : gaia.cs.umass.edu vers www.eurecom.fr

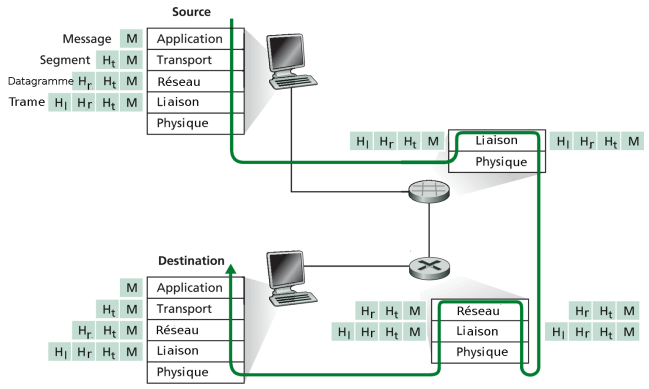
```
1 cs-gw (128.119.240.254) 1 ms 1 ms 2 ms
2 border1-rt-fa5-1-0.gw.umass.edu (128.119.3.145) 1 ms 1 ms 2 ms
3 cht-vbns.gw.umass.edu (128.119.3.130) 6 ms 5 ms 5 ms
4 jn1-at1-0-0-19.wor.vbns.net (204.147.132.129) 16 ms 11 ms 13 ms
5 jn1-so7-0-0-0.wae.vbns.net (204.147.136.136) 21 ms 18 ms 18 ms
6 abilene-vbns.abilene.ucaid.edu (198.32.11.9) 22 ms 18 ms 22 ms
7 nycm-wash.abilene.ucaid.edu (198.32.8.46) 22 ms 22 ms 22 ms
8 62.40.103.253 (62.40.103.253) 104 ms 109 ms 106 ms
9 de2-1.de1.de.geant.net (62.40.96.129) 109 ms 102 ms 104 ms
10 de.fr1.fr.geant.net (62.40.96.50) 113 ms 121 ms 114 ms
11 renater-gw.fr1.fr.geant.net (62.40.103.54) 112 ms 114 ms 112 ms
12 nio-n2.cssi.renater.fr (193.51.206.13) 111 ms 114 ms 116 ms
13 nice.cssi.renater.fr (195.220.98.102) 123 ms 125 ms 124 ms
14 r3t2-nice.cssi.renater.fr (195.220.98.110) 126 ms 126 ms 124 ms
15 eurecom-valbonne.r3t2.ft.net (193.48.50.54) 135 ms 128 ms 133 ms
16 194.214.211.25 (194.214.211.25) 126 ms 128 ms 126 ms
17 * * *
18 * * *

19 fantasia.eurecom.fr (193.55.113.142) 132 ms 128 ms 136 ms
```


Plan

Pile des protocoles Internet

- Application : implante des applications réseaux
  - FTP, SMTP, HTTP
- Transport : transfert de données hôte à hôte
  - TCP, UDP
- Réseaux : routage de datagrammes de la source à la destination
  - IP, protocoles de routages
- Liaison : Transfert de données entre deux éléments réseau voisins
  - PPP, Ethernet
- Physique : Signaux sur un cable

Encapsulation



Plan

Histoire d'Internet

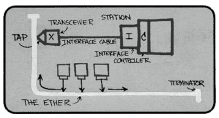
1961-1972 : premiers principes de la commutation de paquets

- 1961 : théorie de file d'attente ⇒ efficacité de la commutation de paquets (Kleinrock)
- 1964 : commutation de paquets dans les réseaux militaires (Baran)
- 1967 : ARPAnet (Advanced Research Projects Agency Network)
- 1969 : premier nœud opérationnel de ARPAnet
- 1972
  - ARPAnet montré au public
  - NCP (Network Control Protocol) premier protocole de machine à machine
  - Premier programme de courrier électronique
  - ARPAnet a 15 nœuds

Histoire d'Internet

1972-1980 : interconnexion, nouveaux réseaux et réseaux propriétaires

- 1970 : ALOHANet, réseau satellitaire à Hawaii (Kleinrock)
- 1973 : Metcalfe propose Ethernet dans sa thèse



Histoire d'Internet

1972-1980 : interconnexion, nouveaux réseaux et réseaux propriétaires

- 1970 : ALOHANet, réseau satellitaire à Hawaii (Kleinrock)
- 1973 : Metcalfe propose Ethernet dans sa thèse
- 1974 : architecture des réseaux d'interconnexion (Cerf et Kahn)

- Minimalité, autonomie : interconnexion sans changements internes
- Modèle de service *best effort*
- Routeurs sans états
- Contrôle décentralisé

→ définissent l'architecture actuelle d'Internet

Histoire d'Internet

1972-1980 : interconnexion, nouveaux réseaux et réseaux propriétaires

- 1970 : ALOHANet, réseau satellitaire à Hawaii (Kleinrock)
- 1973 : Metcalfe propose Ethernet dans sa thèse
- 1974 : architecture des réseaux d'interconnexion (Cerf et Kahn)
- fin 70's : architectures propriétaires : DECnet, SNA, XNA
- fin 70's : commutation de paquets à taille fixe (précurseur d'ATM)
- 1979 ARPANET a 200 nœuds

Histoire d'Internet

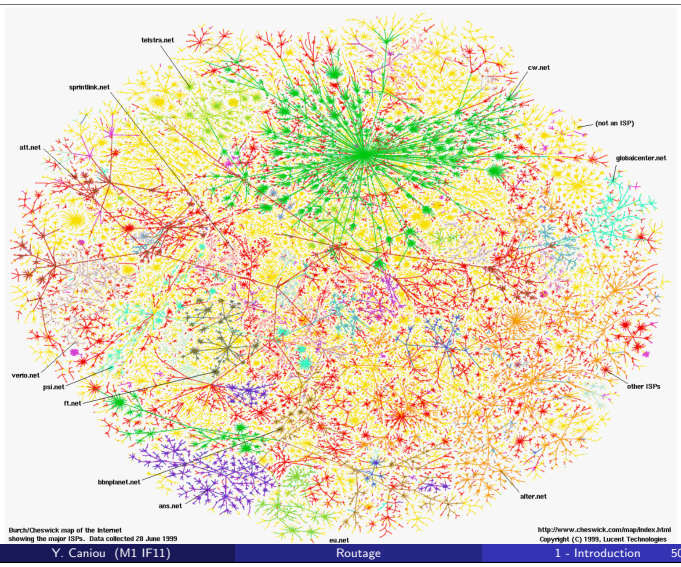
1980-1990 : nouveaux protocoles, prolifération des réseaux

- 1983 : déploiement de TCP/IP
- 1982 : définition du protocole de mail SMTP
- 1983 : définition de DNS pour la transformation nom / adresse IP
- 1985 : définition du protocole FTP
- 1988 : contrôle de congestion dans TCP
- Nouveaux réseaux nationaux : Csnet, BITnet, NSFnet, Minitel
- 100 000 hôtes connectés à une confédération de réseaux

Histoire d'Internet

1990-2000's : commercialisation, Web et applications

- début des 90's : ARPAnet mis de coté
- 1991 : la NSF lève les restrictions sur l'utilisation commerciale de NSFnet (mis de coté en 1995)
- début des 90's : le Web
  - Hypertext [Bush 1945, Nelson 1960's]
  - HTML, HTTP : Berners-Lee
  - 1994 : Mosaic (futur Netscape)
  - fin des 90's : commercialisation du Web
- Fin des 90's- 2000's :
  - Plus d'applications : messagerie instantanée, P2P (KaZaa)
  - Sécurité au premier rang
  - Env. 50 millions d'hôtes, 100 millions d'utilisateurs
  - Liens *backbone* au Gigabit par seconde



Résumé de l'introduction

Une « tonne » d'informations

- Apercu d'ensemble d'Internet
- Les bords, le cœur du réseau
  - Commutation de paquets vs. commutation de circuits
- Structure d'Internet
- Performances : pertes et délais
- Modèle en couches et de services
- Historique

Deuxième partie

Routage

Objectifs

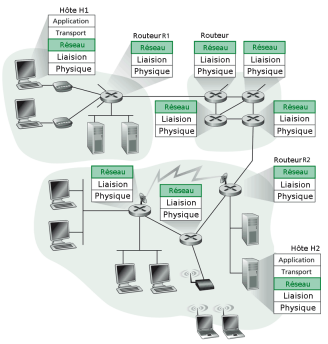
- Comprendre les principes derrière les services de la couche réseau
  - Routage (sélection de chemin)
  - Gestion de l'échelle
  - Comment marche un routeur
- Instantiation et implantation sur Internet

Deuxième partie

Routage

Couche réseau

- Transporte des segments d'un hôte émetteur à un hôte récepteur
- Coté émetteur : encapsulation des segments en datagrammes
- Coté récepteur : livre des segments à la couche transport
- Les protocoles de la couche réseau se situent sur tous les hôtes et tous les routeurs
- Un routeur examine les champs d'en-tête de tous les paquets qui passent par lui



Fonctions clés de la couche réseau

Retransmission

- Déplacer des paquets depuis une entrée d'un routeur vers la sortie appropriée du routeur

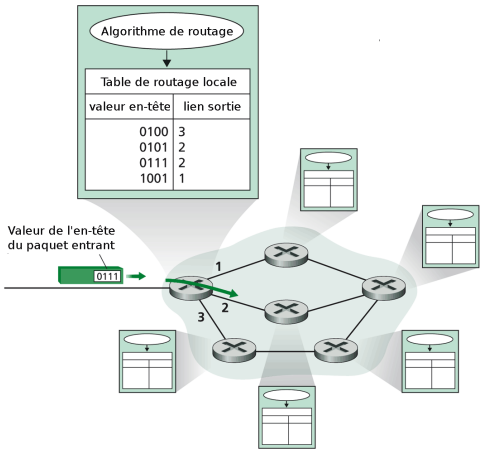
Routage

- Déterminer la route prise par les paquets de la source à la destination
  - Algorithmes de routage

Analogie

- Routage : préparer son itinéraire
- Retransmission : passer sur un échangeur autoroutier

Relations entre routage et retransmission



Établissement de la connexion

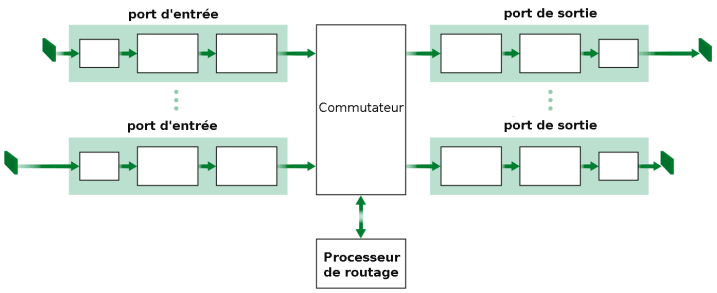
- Troisième fonction clé de certaines architectures réseaux
  - ATM, Frame-relay, X.25
- Avant d'envoyer des datagrammes, deux hôtes et les routeurs participant établissent une connexion virtuelle
  - Implication des routeurs
- Services de connexion des couches réseau et transport
  - Réseau : entre deux hôtes
  - Transport : entre deux processus

Plan

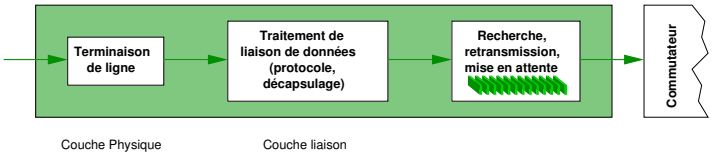
Aperçu de l'architecture d'un routeur

Deux fonctions clés

- Exécuter algorithmes et protocoles de routage (RIP, OSPF, BGP)
- Retransmettre les datagrammes du lien entrant vers le lien sortant



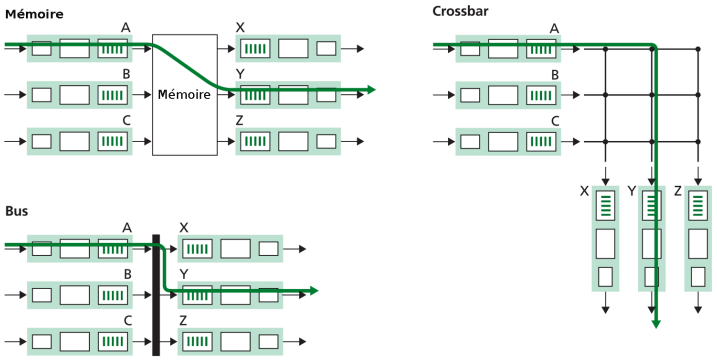
Fonctions du port d'entrée



Commutation décentralisée

- D'après la destination du datagramme, cherche le port de sortie en utilisant la table de routage dans la mémoire du port d'entrée
- Objectif : traiter à la "vitesse de la ligne"
- File d'attente : si les paquets arrivent plus vite que le débit de retransmission du commutateur

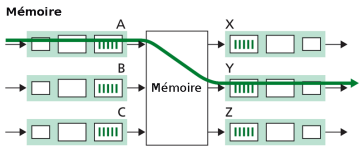
Trois types de commutateurs



Commutation en mémoire

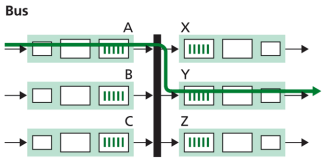
Commutateurs de première génération

- Ordinateurs traditionnels avec commutation sous contrôle direct du processeur
- Paquets copiés dans la mémoire du système
- Vitesse limitée par la bande passante de la mémoire (2 passages par le bus par datagramme)



Commutation via un bus

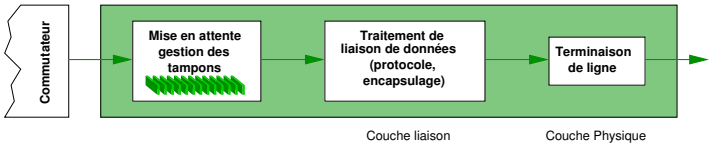
- Un datagramme va de la mémoire du port d'entrée à celle du port de sortie via un bus partagé
- Contention du bus : vitesse de commutation limitée par la bande passante du bus
- Ex : Bus 1Gbps Cisco 1900 → vitesse suffisante pour des routeurs d'accès (ADSL) ou d'entreprises (LAN) mais pas pour du régional ni sur un backbone



Commutation via un réseau d'interconnexion

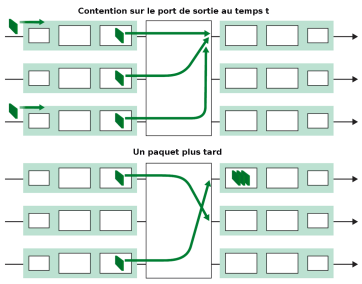
- Surmonte les limitations dues à la bande passante du bus
- Réseaux de Banyan (ou autres) → initialement conçus pour connecter les processeurs d'un multi-processeurs
- Conception avancée : fragmenter les datagrammes en cellules de taille fixe puis commuter les cellules
- Ex : Cisco 12000 : commute des Gbps via son réseau d'interconnexion

Ports de sortie



- Mise en attente nécessaire si les datagrammes arrivent du commutateur plus vite que le débit de transmission
- La politique d'ordonnancement choisit les datagrammes à transmettre parmi ceux mis en attente

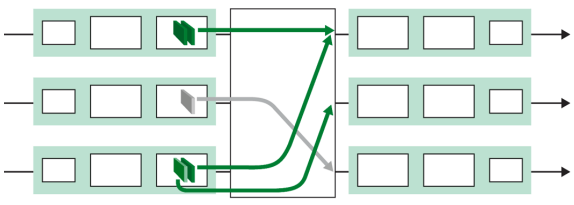
Mise en attente en sortie



- Quand le taux d'arrivée est supérieur à la vitesse de la ligne
- Délais d'attente et pertes dues au débordement du tampon du port de sortie

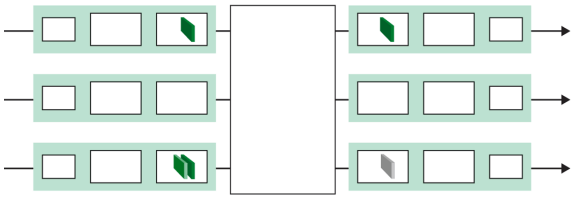
Mise en attente en entrée

- Commutateur plus lent que la combinaison des ports → mise en attente possible dans les files d'entrée
- Blocage de la tête de file (*HOL, Head of the line*) : le premier datagramme de la file empêche les autres de passer
- Délais d'attente et pertes dues au débordement du tampon du port de sortie



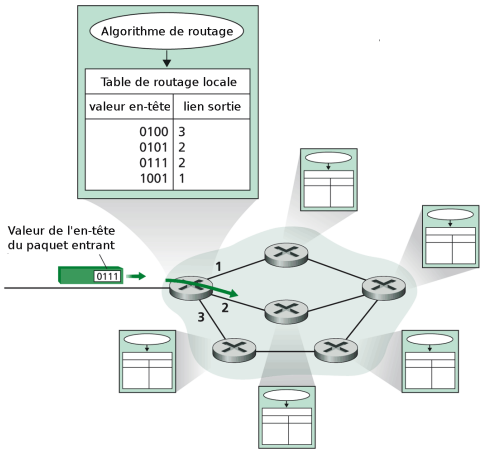
Mise en attente en entrée

- Commutateur plus lent que la combinaison des ports → mise en attente possible dans les files d'entrée
- Blocage de la tête de file (*HOL, Head of the line*) : le premier datagramme de la file empêche les autres de passer
- Délais d'attente et pertes dues au débordement du tampon du port de sortie



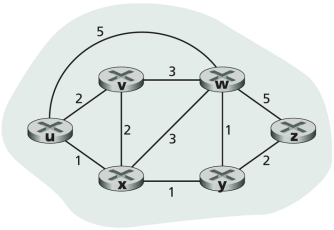
Plan

Relations entre routage et retransmission

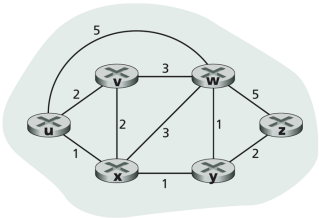


Abstraction par l'intermédiaire d'un graphe

- Graphe : $G = (N, E)$
- N = ensemble de routeurs $= u, v, w, x, y, z$
- E = ensemble de liens $= \{(u,v),(u,x),(v,x),(v,w),(x,w),(x,y),(w,y),(w,z),(y,z)\}$



Abstraction : coûts



- $c(x, x')$ = coût du lien (x, x'). Ex. : $c(w, z) = 5$
- Coûts pourrait être toujours 1, ou inversement proportionnel à la bande passante ou inversement proportionnel à la congestion

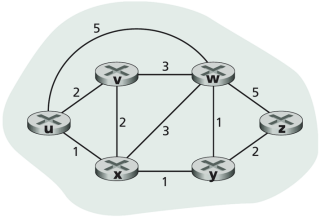
Coût du chemin $(x_1, x_2, x_3, \cdots, x_p) =$

$$c(x_1, x_2) + c(x_2, x_3) + \cdots + c(x_{p-1}, x_p)$$

Question

Quel est le chemin de coût minimum entre u et z ?

Abstraction : coûts



- $c(x, x')$ = coût du lien (x, x'). Ex. : $c(w, z) = 5$
- Coûts pourrait être toujours 1, ou inversement proportionnel à la bande passante ou inversement proportionnel à la congestion

Coût du chemin $(x_1, x_2, x_3, \cdots, x_p) =$

$$c(x_1, x_2) + c(x_2, x_3) + \cdots + c(x_{p-1}, x_p)$$

Réponse

Un algorithme de routage va trouver ce chemin de coût minimum

Classification des algorithmes de routage

Informations globales ou décentralisées

Décentralisées

- Un routeur connaît ses voisins et le coût des liens vers ses voisins
- Processus itératif de calcul et d'échange d'informations entre voisins
- Algorithmes à *vecteurs de distances*

Globales

- Tous les routeurs connaissent la topologie complète et les informations de coût de tous les liens
- Algorithmes à *état de lien*

Classification des algorithmes de routage

Statique ou dynamique ?

Statique

- Les routes changent lentement au cours du temps

Dynamique

- Les routes changent plus rapidement
  - Mise-à-jour périodique
  - En réponse aux changements des coûts des liens

Plan

Un algorithme de routage à état de lien

Algorithme de Dijkstra

- Topologie réseau et coûts des liens connus de tous les nœuds
  - Possible par une « diffusion de l'état des liens »
  - Tous les nœuds ont la même information
- Calcule les plus courts chemins depuis une source vers tous les autres nœuds
  - Fournit donc la table de routage pour ce nœud
- Itératif : après k itérations, on connaît le plus court chemin vers k destinations

Un algorithme de routage à état de lien

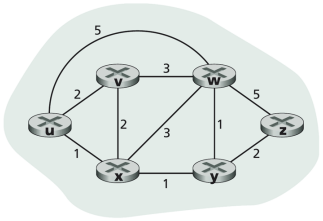
Notations

- $c(x, y)$: coût du lien du nœud x au nœud y. Égal à $+\infty$ si pas voisins directs
- $D(v)$: valeur courante du chemin de la source à la destination v
- $p(v)$: nœud précédant v dans le chemin de la source à v
- N' : ensemble de nœuds pour lesquels le plus court chemin est définitivement connu

Algorithme de Dijkstra

```
1 Initialisation :
2
3 N' = {u}
4 Pour tous les nœuds v
5       Si v voisin de u
6             alors D(v) = c(u, v)
7       sinon D(v) = ∞
8
9 Répéter
10      Trouver w ∉ N' tel que D(w) est minimal
11      Ajouter w à N'
12      Mettre à jour D(v) pour tout v voisin de w et ∉ N'
13      D(v) = min(D(v), D(w) + c(w, v))
14      /* Le nouveau coût pour v est soit l'ancien coût pour v, soit le
15      coût du plus court chemin vers w plus le coût de w à v */
16 jusqu'à ce que tous les nœuds soient dans N'
```

Algorithme de Dijkstra : exemple



	N'	$D(v), p(v)$	$D(w), p(w)$	$D(x), p(x)$	$D(y), p(y)$	$D(z), p(z)$
0	u	$2, u$	$5, u$	$1, u$	∞	∞
1	ux	$2, u$	$4, x$		$2, x$	∞
2	uxy	$2, u$	$3, y$			$4, y$
3	$uxyv$		$3, y$			$4, y$
4	$uxyvw$					$4, y$
5	$uxyvwz$					

Algorithme de Dijkstra : résultats

Arbre des plus courts chemins depuis u

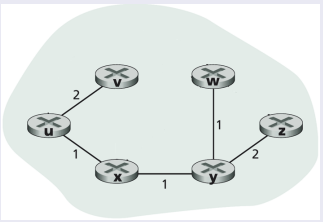


Table de routage pour u

Destination	Lien
v	(u, v)
x	(u, x)
y	(u, x)
w	(u, x)
z	(u, x)

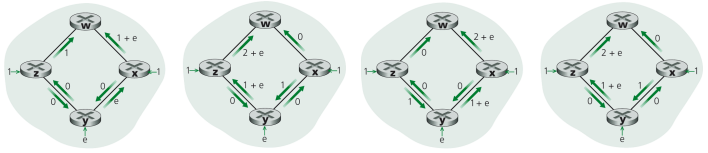
Algorithme de Dijkstra : discussion

Complexité de l'algorithme pour n nœuds

- Chaque itération : verification de tous les nœuds $w \notin N$
- $n(n+1)/2$ comparaisons : $\mathcal{O}(n^2)$
- Versions plus efficaces en $\mathcal{O}(n \log n)$

Oscillations possibles

- Ex. : coût du lien = trafic sur le lien
- x et z envoient 1 à w, y envoie e à w



Plan

Algorithme à vecteur de distances

Équation de Bellman-Ford

- Définit

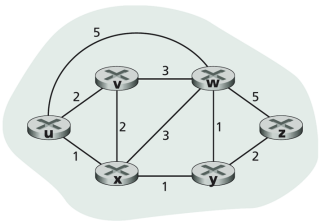
$$d_x(y) \leftarrow \text{coût du plus court chemin de } x \text{ à } y$$

- puis

$$d_x(y) = \min_v\{c(x, v) + d_v(y)\}$$

- où le minimum est trouvé parmi tous les voisins v de x

Bellman-Ford : exemple



- On peut voir que $d_v(z) = 5$, $d_x(z) = 3$, $d_w(z) = 3$
- L'équation de Bellman-Ford dit que :

$$\begin{aligned} d_u(z) = & \; min\{\, c(u, v) + d_v(z), \\ & \quad c(u, x) + d_x(z), \\ & \quad c(u, w) + d_w(z)\} \\ = & \; min\{\, 2 + 5, 1 + 3, 5 + 3\} \\ = & \; 4 \end{aligned}$$

Le nœud qui donne le minimum est le prochain saut dans le plus court chemin → table de routage

Algorithme à vecteur de distances

- $D_x(y)$ estimation du moindre coût de x à y
- Vecteur de distances : $D_x = [D_x(y) : y \in N]$
- Le nœud x connaît le coût pour chacun de ses voisins v : $c(x, v)$
- Le nœud x maintient $D_x = [D_x(y) : y \in N]$
- Le nœud x maintient aussi les vecteurs de distances de ses voisins
  - Pour chaque voisin v, x maintient $D_v = [D_v(y) : y \in N]$

Algorithme à vecteur de distances

Idée de base

- Chaque nœud envoie périodiquement sa propre estimation du vecteur de distances à ses voisins
- Lorsqu'un nœud x reçoit un nouveau vecteur d'un voisin, il met à jour son propre vecteur en utilisant l'équation de Bellman-Ford

$$D_x(y) \leftarrow \min_v\{c(x, v) + D_v(y)\}, \ \forall y \in N$$

- Sous quelques conditions réalistes, le $D_x(y)$ estimé converge vers le véritable moindre coût $d_x(y)$

Algorithme à vecteur de distances

Itératif, aynchrone

Chaque itération locale provoquée par

- Un changement de coût d'un lien local
- Un message de mise-à-jour de vecteur de la part d'un voisin

Distribué

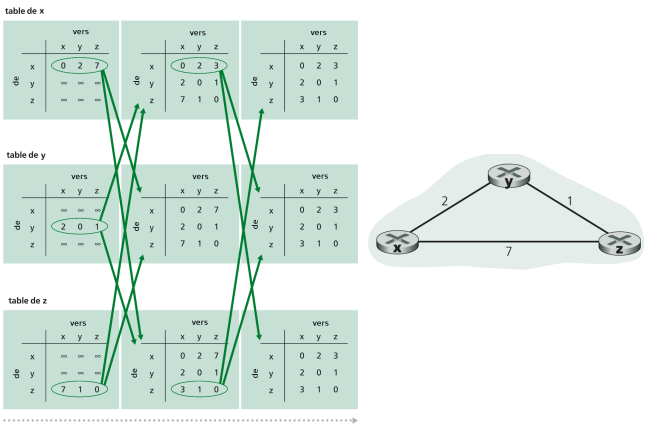
- Chaque nœud ne notifie ses voisins que lorsque son vecteur change
  - Les voisins notifient alors leurs voisins si besoin est

Algorithme à vecteur de distances

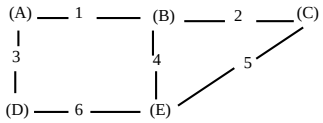
Comportement d'un nœud

1. Attend (changement de coût d'un lien local ou message d'un voisin)
2. Recalcule ses estimations
3. Si un vecteur change, nofitie ses voisins
1. Attend . . .

Algorithme à vecteur de distances



Vecteur de distances, autre exemple



- Objectif : construire les tables de routages des nœuds $N = \{A, B, C, D, E\}$
- Configuration initiale : pour chaque nœud $i \in N$,

de i à	liaison	coût
i	L (locale)	0

- Exemple repris de *le routage dans l'Internet* de C.Huitema, édition Eyrolles, ISBN 2-212-08902-3.

Traitement des messages (rappel)

- À la réception d'un message, le coût de la liaison par laquelle est reçu le message est ajouté à la distance annoncée
- **Pour** chaque destination annoncée
  - **Si** la destination est inconnue : mettre la nouvelle entrée dans la table de routage
  - **Sinon**
    - **Si** la distance est plus courte (relâchement), mettre à jour
    - **Si** la distance est plus grande et utilise la route déjà connue, mettre à jour
    - **Sinon** information ignorée

Traitement des messages (2)

A diffuse sur 1 et 3 $(A, 0)$

État avant			État après		
sur A					
A	L	0	A	L	0
sur B					
B	L	0	B	L	0
			A	1	1
sur D					
D	L	0	D	L	0
			A	3	1

Traitement des messages (3)

B diffuse sur 1, 2 et 4
D diffuse sur 3, 6

sur A						sur C					
A	L	0	A	L	0	C	L	0	C	L	0
			B	1	1				B	2	1
			D	3	1				A	2	2
sur B						sur E					
B	L	0	B	L	0	E	L	0	E	L	0
A	1	1	A	1	1				B	4	1
sur D									A	4	2
D	L	0	D	L	0				D	6	1
A	3	1	A	3	1						

Traitement des messages (4)

A diffuse sur 1, 3
C diffuse sur 2, 5
E diffuse sur 4, 5 et 6

sur B						sur E					
B	L	0	B	L	0	E	L	0	E	L	0
A	1	1	A	1	1	B	4	1	B	4	1
			D	1	2	A	4	2	A	4	2
			C	2	1	D	6	1	D	6	1
			E	4	1				C	5	1
sur D											
D	L	0	D	L	0						
A	3	1	A	3	1						
			B	3	2						
			E	6	1						

Traitement des messages (5)

B diffuse sur 1, 2 et 4
D diffuse sur 3, 6

sur A						sur C					
A	L	0	A	L	0	C	L	0	C	L	0
			B	1	1				B	2	1
			D	3	1				A	2	2
sur B						sur E					
B	L	0	B	L	0	E	L	0	E	L	0
A	1	1	A	1	1				B	4	1
sur D									A	4	2
D	L	0	D	L	0				D	6	1
A	3	1	A	3	1						

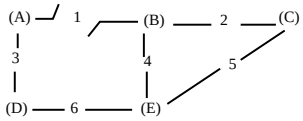
Traitement des messages (6)

B diffuse sur 1, 3 et 4
D diffuse sur 3, 6
E diffuse sur 4, 5 et 6

sur A						sur C					
A	L	0	A	L	0	C	L	0	C	L	0
B	1	1	B	1	1	B	2	1	B	2	1
D	3	1	D	3	1	A	2	2	A	2	2
			C	1	2				E	5	1
			E	1	2				D	5	2
sur D											
D	L	0	D	L	0						
A	3	1	A	3	1						
B	3	2	B	3	2						
E	6	1	E	6	1						
			C	6	2						

En cas de coupure

→ Découverte de la panne sur les noeuds A et B



⇒ Mise à jour des tables :(coût liaison $1 \leftarrow \infty$)

sur A		
A	L	0
B	1	∞
D	3	1
C	1	∞
E	1	∞

sur B		
B	L	0
A	1	∞
D	1	∞
C	2	1
E	4	1

⇒ Diffusion de la mise-à-jour

En cas de coupure (2)

A diffuse sur 3
B diffuse sur 2 et 4

sur D		
D	L	0
A	3	1
B	3	∞
E	6	1
C	6	2

sur C		
C	L	0
B	2	1
A	2	∞
E	5	1
D	5	2

sur E		
E	L	0
B	4	1
A	4	∞
D	6	1
C	5	1

En cas de coupure (3) : remarque

- D reçoit de A sur liaison 3 :
  $< (A,0)(B,\infty)(D,1)(C,\infty)(E,\infty) >$ qui après incrémentation (coût liaison 3) devient : $< (A,1)(B,\infty)(D,2)(C,\infty)(E,\infty) >$.
  Ces distances sont supérieures à celles dans la table, mais pour **B**, la liaison 3 était celle utilisée, c'est pourquoi il y a mise-à-jour.
- Règle : Si pour une destination, la nouvelle distance est supérieure à l'ancienne **et** que cette information arrive par la liaison utilisée comme route vers cette destination, alors la table est mise-à-jour.

En cas de coupure (4)

D diffuse sur 3 et 6
C diffuse sur 2 et 5
E diffuse sur 4, 5 et 6

sur A		
A	L	0
B	1	∞
D	3	1
C	3	3
E	3	2

sur B		
B	L	0
A	1	∞
D	4	2
C	2	1
E	4	1

sur D		
D	L	0
A	3	1
B	6	2
E	6	1
C	5	2

sur E		
E	L	0
B	4	1
A	6	2
D	6	1
C	5	1

En cas de coupure (5)

A diffuse sur 3
B diffuse sur 2 et 4
D diffuse sur 3 et 6
E diffuse sur 4, 5 et 6

sur A		
A	L	0
B	3	3
D	3	1
C	3	3
E	3	2

sur B		
B	L	0
A	4	3
D	4	2
C	2	1
E	4	1

sur C		
C	L	0
B	2	1
A	5	2
E	5	1
D	5	2

- Convergence
- Connectivité globale retrouvée

L'effet rebond

- Coûts différents : la liaison 5 a un coût de 10
- Les routes vers (C) seront :

	Liaison	Coût
$A \rightarrow C$	1	2
$B \rightarrow C$	2	1
$C \rightarrow C$	L	0
$D \rightarrow C$	3	3
$E \rightarrow C$	4	2

L'effet rebond - scénario

- Coupure de la liaison 2 ; (B) détecte la coupure, la table des routes vers (C) devient :

	Liaison	Coût
$A \to C$	1	2
$B \to C$	2	∞
$C \to C$	L	0
$D \to C$	3	3
$E \to C$	4	2

Mais A diffuse son vecteur à (B) et (D) avant que (B) annonce sa mise-à-jour. En effet, dans la plupart des implantations, une transmission régulière des vecteurs est rajoutée à la retransmission déclenchée

L'effet rebond - scénario (2)

- Pas d'effet sur (D), mais sur (B), la route vers (C) annoncée a un coût de $2+1 < \infty$. Dans la table de (B), se trouve l'entrée : $B \to C$, 1,3
- (B) annonce ses routes à (A) (liaison 1) et (E) (liaison 4), ce qui après mise-à-jour (les messages arrivent par les liaisons utilisées pour la route vers (C)) :

	Liaison	Coût
$A \to C$	1	4
$B \to C$	1	3
$C \to C$	L	0
$D \to C$	3	3
$E \to C$	4	4

▷ boucle dans les tables de routages entre (A) et (B) ; c'est **l'effet rebond** qui disparaîtra avec le temps

L'effet rebond - scénario (3)

C annonce (E ignore (10)) A et E annoncent (MàJ B et D)			B annoce (MàJ en A,C, 6) A annoce à B et D			nouveau cycle (distances vers C : + 2)		
$A \to C$	1	4	$A \to C$	1	6	$A \to C$	1	8
$B \to C$	1	5	$B \to C$	1	7	$B \to C$	1	9
$C \to C$	L	0	$C \to C$	L	0	$C \to C$	L	0
$D \to C$	3	5	$D \to C$	3	7	$D \to C$	3	9
$E \to C$	4	4	$E \to C$	4	6	$E \to C$	4	8

encore un cycle E ignore encore l'annonce de C			E prend en compte l'annonce de C (10¡11) et après quelques échanges		
$A \to C$	1	12	$A \to C$	1	12
$B \to C$	1	11	$B \to C$	4	11
$C \to C$	L	0	$C \to C$	L	0
$D \to C$	3	12	$D \to C$	6	11
$E \to C$	4	11	$E \to C$	5	10

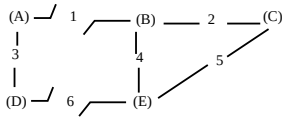
L'effet rebond - remarque (4)

- Attention : processus aléatoire, donc le résultat et la vitesse de convergence dépendent de l'ordre d'échange des messages (imprévisible)
- Effet rebond : congestion locale de paquets (dans la boucle) et donc perte (TTL=0), y compris des paquets de routages :)

→ Situation à éviter

Le comptage à "l'infini"

- Revenons au cas où tous les coûts sont de 1
- Supposons la coupure de la liaison 1 comme précédemment,
- après convergence, supposons une deuxième coupure de la liaison 6



- (A) et (D) isolés (perte de connexité du réseaux)

Le comptage à "l'infini" (2)

- (D) détecte la panne, sa table devient :

sur D		
D	L	0
A	3	1
B	6	∞
E	6	∞
E	6	∞

Le comptage à "l'infini" (3)

- Comportement de (A) et (D) :
  1. (D) détecte la panne et annonce à (A), convergence
  2. (A) annonce avant (D), la table de (D) devient :

sur D		
D	L	0
A	3	1
B	3	4
E	3	4
E	3	4

  3. Apparition d'une boucle ; à chaque cycle, les distances augmentent de 2, plus de convergence possible sauf quand "l'infini est atteint". c'est le **comptage à l'infini**

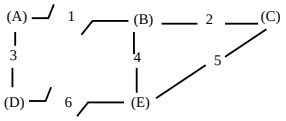
L'horizon partagé et retour empoisonné

- **Horizon partagé** :

  Principe (simple) : Si A passe par B pour atteindre X, alors B ne doit pas essayer d'atteindre X par A
  donc : dans l'annonce faite sur une liaison, sont enlevées toutes les destinations atteintes (apprises) via cette liaison. (cela suppose que les destinations inaccessibles ne sont jamais annoncées, ou bien qu'elles sont oubliées après un certain délai.)
- **Retour empoisonné** : Si, juste après avoir détecté une route coupée, un routeur reçoit une information d'accessibilité avec un coût "important" par rapport au coût initial, il ignore cette information (boucle)

Pas toujours suffisant

- Revenons à la situation précédente :



-

(E) découvre la panne routes vers (D)			(E) annonce sur 4 et 5 (B) reçoit (C) ne reçoit pas			(C) annonce (retour empoisonné) apparition du cycle		
$B \to D$	4	2	$B \to D$	4	∞	$B \to D$	2	2
$C \to D$	5	2	$C \to D$	5	2	$C \to D$	5	2
$E \to D$	6	∞	$E \to D$	6	∞	$E \to D$	4	4

La mise-à-jour déclenchée

- À quel moment un routeur doit-il annoncer ?
- Envois périodiques : souvent préconisée

  Surveillance voisin et correction erreurs
  Temporisation sur une route : au delà, route coupée...
  Par exemple : RIP recommande 6 fois la période de répétition des messages
  Ennuyeux si trop long (réactivité)
- **Mise-à-jour déclenchée** : *[RFC 1058]*, faire annonce dès que la table est modifiée
- **Mise-à-jour retardée**

État de lien (EL) vs. vecteur de distance (VD)

Complexité en nombre de messages

- EL : n nœuds et E liens $\rightarrow \mathcal{O}(nE)$ messages envoyés
- VD : échanges entre voisins uniquement $\rightarrow$ le temps de convergence varie

Vitesse de convergence

- EL : un algorithme en $\mathcal{O}(n^2)$ demande $\mathcal{O}(nE)$ messages (oscillations possibles)
- VD : Temps variable (problème du comptage à l'infini et boucles de routage)

État de lien (EL) vs. vecteur de distance (VD)

Robustesse : qu'arrive-t-il si le routeur est défectueux ?

- EL
  - Un nœud peut publier un coût de lien erroné
  - Chaque nœud calcule sa propre table
- VD
  - Un nœud peut publier un coût de chemin erroné
  - La table de chaque nœud est utilisé par les autres $\rightarrow$ propagation des erreurs

État de lien (EL) vs. vecteur de distance (VD)

Métrique

- Plus précise
- Éventuellement plusieurs métriques

Plan

Routage hiérarchique

Constat

Étude idéalisée du routage pour l'instant

- Tous les routeurs sont identiques
- Réseau « à plat »

. . . fausse en pratique

Routage hiérarchique

Constat

Étude idéalisée du routage pour l'instant . . . fausse en pratique

Échelle : 200 millions de destinations

- Impossible de les stocker toutes dans les tables de routage !
- Échanges de tables → gaspillage des liens !

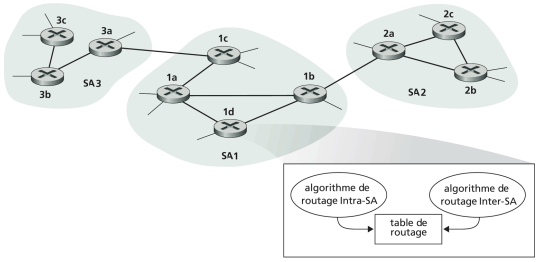
Autonomie administrative

- Internet : réseau de réseaux
- Chaque administrateur réseau peut vouloir contrôler le routage dans son propre réseau

Routage hiérarchique

- Aggréger les routeurs en régions → systèmes autonomes (SA)
- Les routeurs d'un même SA exécutent le même protocole de routage
  - Protocole de routage "intra-SA"
  - Des routeurs dans des SA différents peuvent exécuter des protocoles de routage intra-SA différents
- Routeur passerelle → liaison directe vers un routeur d'un autre SA

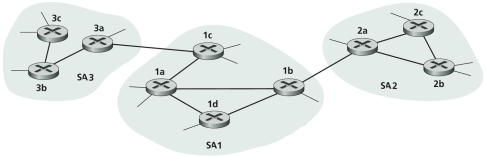
SA interconnectés



- La table de routage est configurée par les algorithmes de routage intra- et inter-SA
  - Intra-SA → entrées pour destinations internes
  - Inter-SA et Intra-SA → entrées pour destinations externes

Tâches inter-SA

- Supposons qu'un routeur dans SA1 reçoive un datagramme dont la destination est en dehors de SA1
  - Le routeur doit retransmettre le paquet vers un des routeurs passerelle, mais lequel ?
- SA1 doit
  1. apprendre quelles destinations peuvent être atteintes au travers de SA2 et au travers de SA3
  2. propager ces informations à tous les routeurs de SA1
- Travail du routage inter-SA



Fixer la table de routage du routeur 1d

Exemple

- Supposons que SA1 apprenne par le protocole inter-SA que le sous-réseau x est atteignable par SA3 (passerelle 1c) mais pas par SA2
- Le protocole inter-SA propage l'information à tous les routeurs internes
- Le routeur 1d détermine à partir du routage intra-SA que son interface I est sur le chemin de moindre coût vers 1c
- Ajoute l'entrée (x,I) dans la table de routage

Choisir parmi plusieurs SA

Exemple

- Supposons maintenant que SA1 apprenne du protocole inter-SA que le sous réseau x est atteignable par SA3 et par SA2
- Pour configurer la table de routage, le routeur 1d doit déterminer par quelle passerelle retransmettre les paquets destinés à x
- C'est aussi le travail du protocole de routage inter-SA !
- Routage en patate chaude : envoyer le paquet vers le plus proche routeur des deux

Plan

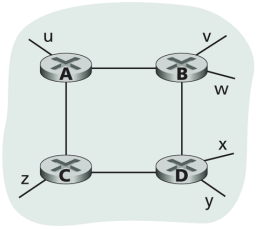
Routage intra-SA

- Également connu sous le nom de *Interior Gateway Protocol (IGP)*
- Protocoles de routage intra-SA les plus courants
  - RIP : *Routing Information Protocol*
  - OSPF : *Open Shortest Path First*
  - IGRP : *Interior Gateway Routing Protocol* (Cisco)

RIP *Routing Information Protocol*

RIP v1

- Algorithme à vecteur de distances
- Inclus dans la distribution BSD-Unix en 1982
- Mesure de distance : nombre de sauts (maximum = 15 sauts)



Destination	Sauts
u	1
v	2
w	2
x	3
y	3
z	2

Publications RIP

- Vecteurs de distances : échangés entre voisins toutes les 30 secondes *via* des messages de réponses (aussi appelés publications)
- Publication : liste composée d'un maximum de 25 réseaux destinations au sein du SA
- Mises-à-jour déclenchées avec retard entre 1 et 5 sec

RIP : exemple

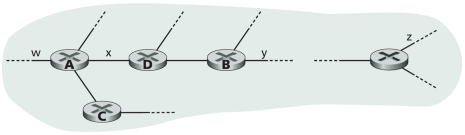
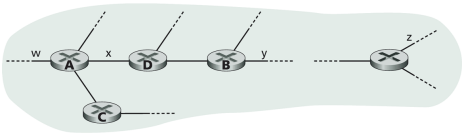


Table de routage de D

Réseau destination	Prochain routeur	Nombre de sauts
w	A	2
y	B	2
z	B	7
x	–	1
...	...	...

RIP : exemple



Publication de A à D

Dest.	Prochain	Sauts
w	–	1
y	–	1
z	C	4
...	...	...

Table de routage de D

Réseau destination	Prochain routeur	Nombre de sauts
w	A	2
y	B	2
z	**A**	**5**
x	–	1
...	...	...

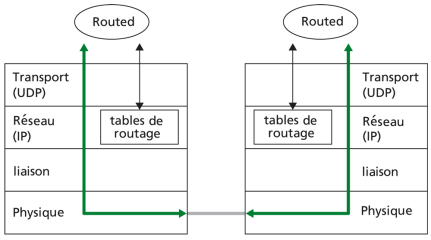
RIP : panne de lien et récupération

Si aucune publication reçue après 180s → voisin/lien déclaré mort

- Invalidation des routes passant par ce voisin
- Nouvelles publications envoyées aux voisins
- qui envoient à leur tour de nouvelles publications (si les tables ont changé)
- Propagation rapide de l'information de panne
- Utilisation de réponses empoisonnées pour éviter les boucles ping-pong (distance infinie = 16 sauts)

RIP : traitement des tables

- Gestion des tables de routage de RIP par un processus de **niveau application** appelé `routed` (démon)
- Publications envoyées dans des paquets UDP réémis périodiquement



RIP v1

- Origine : document 1988 (C.Hedrick) *[RFC 1058]*, suggère l'horizon partagé et mise-à-jour déclenchée
- Demeure : *Interior Gateway Protocol* pour les *autonomous system* de taille limitée
- Identification des destinations : adresse IP (32 bits)
  - Une ligne - une destinaion : réseau, sous-réseau, hôte
  - En fonction de la classe (A,B,C), *netmask* non transmis
  - Pour différencier un hôte d'un sous-réseau, il faut connaître le masque, celui-ci ne devrait pas être connu en dehors du réseau
  - Route par hôte optionnelle (et remplacée par des routes réseaux)
  - 0.0.0.0 : route par défaut (réseaux extérieurs).

RIP v1 (2)

- Taille max : 512 octets, soit 21 entrées (si + , plusieurs messages)
- Une entrée

commande	version=1	0
fam. d'adresse = 2 (IP)		0
adresse IP		
0 (historique BSD-ref port)		
0		
métrique (coût)		

- Commande

  REQUEST = 1
  RESPONSE = 2
  traceon, traceoff(3,4) obsolètes, (5) réservé SunMicrosystems, (6) réservé

RIP v2

- 1993 : Gary Malkins *[RFC 1388], [RFC 1721-1724]*
- Technologie obsolète (par rapport au protocoles à états de liaisons)
- **Mais** suffisante dans de nombreuses situations
- Implante le CIDR (transmission des masques de sous-réseaux)

RIP v2 (2)

commande	version=2	Domain
(optionnel)		
0xFFFF authentification		type authentification
Authentification		
Authentification		
Authentification		
Authentification		
(répété jusqu'à 24 fois)		
Address family		route tag
adresse IP		
masque sous réseau		
prochain saut		
métrique (coût)		

RIP v2 (2)

-Si la valeur de "type authentification" est égale à 0, mot de passe en clair ; un autre format plus complexe avec chiffrement des mots de passe existe aussi

-Le champs "Domain" permet de découper logiquement le réseau (un routeur ignore les messages des autres domaines)

-"Route Tag" permet de le marquage (EGP,...)

-"Next Hop" : routeur qui annonce (en général)

-En mode *broadcast* ou *multicast* (224.0.0.9)

OSPF (*Open Shortest Path First*)

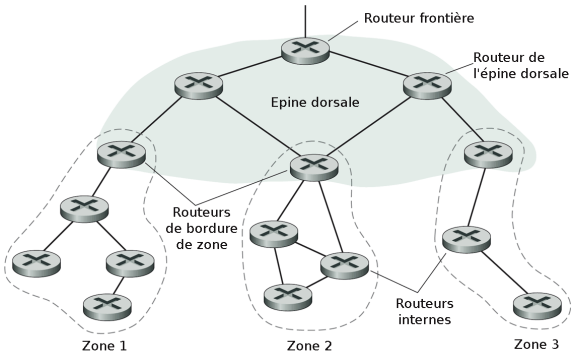
«

- Open » : disponible publiquement
- Utilise un algorithme à état de lien
  - Dissémination de paquets EL
  - Carte de la topologie sur chaque nœud
  - Calcul des routes par l'algorithme de Dijkstra
- Publication OPSF → une entrée par routeur voisin
- Publications disséminées dans tout le SA (par inondation)
  - Messages OPSF directement sur IP (plutôt que TCP ou UDP)

Caractéristiques « avancées » d'OSPF (pas dans RIP)

- Sécurité : authentification de tous les messages OSPF (pour éviter les intrusions malveillantes)
- Plusieurs chemins de même coût autorisés (un seul pour RIP)
- Pour chaque lien, plusieurs mesures de coût pour différents Temps de service (ex. : coût d'un lien satellite fixé à "faible" pour du *best effort* et à "élevé" pour du temps réel)
- OSPF hiérarchique dans les grands domaines

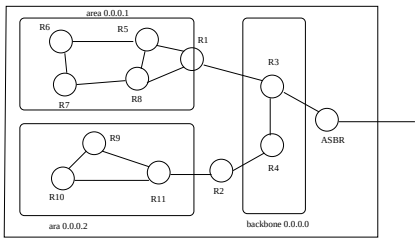
OSPF hiérarchique



OSPF hiérarchique

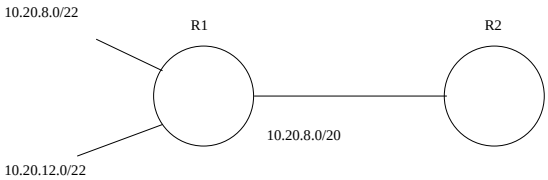
- Hiérarchie à deux niveaux : zone locale et épine dorsale
  - Publication de l'état des liens seulement dans la zone
  - Chacun des nœuds à une vision détaillée de la topologie de la zone
  - et ne connaît que la direction (plus court chemin) vers des réseaux situés dans d'autres zones
- Routeurs de bord de zone : "résument" les distances vers les réseaux de leurs propres zones et les diffusent vers les autres routeurs de bord de zone
- Routeurs de l'épine dorsale : exécutent un routage OSPF limité à l'épine dorsale
- Routeurs de frontière : connectent aux autres SA

Zones



- Internal Router : n'annonce que les routes internes de la zone (R5-R8 : area 0.0.0.1, R9-R11 : area 0.0.0.2, R2, R3, R4 : area 0.0.0.0)
- Area Boudary Router : connexion au backbone et annonce route externe (R1). R2 qui ne fait que la liaison entre la zone 2 et le backbone est considéré comme IR au backbone
- Autonomous System Boudary Router : échanges avec autres AS, routes exterieures apprises par protocoles autres (statique, BGP, ...)

Aggrégation de routes



Mise en œuvre de OSPF

- Directement sur IP : protocole 87
- Multicast sur réseaux à diffusion :

  224.0.0.5 : tous les routeurs de l'aire
  22.4.0.0.6 : le routeur désigné (écouter par le routeur backup aussi)
- 4 étapes :

  Élection du routeur désigné et backup
  Synchronisation des BD
  Mise-à-jour
  Calcul des chemins
- Réalisées par 3 protocoles

  Hello : découverte des routeurs voisins et élection
  Protocole d'échange : synchronisation (démarrage, rétablissement de connectivité)
  Innondation pour propager une modification

Entête OSPF

Entête commune à tous les messages OSPF (24 octets)

version=2	type	longueur (o)
IP source		
Identificateur de zone		
Checksum	type authentification	
Authentification		
Authentification		

Type des messages :

1	Hello	2	description de BD
3	requête état de la lisaison (LSR)	4	message MàJ (LSU)
5	ack de 4		

OSPF : limitation du trafic de mise à jour

- Notion de *Designated Router* (routeur désigné) et de *Backup Router*
- Utilise un algorithme d'élection reposant sur la priorité (entre 0 et 255) affectée par l'administrateur, et à priorité égale, sur la plus haute ID (adresse IP) du routeur
- Le routeur envoie sa mise-à-jour au routeur désigné qui rediffuse l'information

OSPF : démarrage, élection, HELLO

- À sa mise sous tension, un routeur écoute le trafic et découvre le routeur désigné et celui de secours ; il les accepte, même si sa priorité est plus grande. En l'absence de trafic, déclenchement d'une élection (détection aussi de la défaillance du DR)
- Message « HELLO » envoyé périodiquement par chaque routeur
- Si aucun message reçu pendant période $>$ **intervalle de mort**, liason déclarée coupée

OSPF : démarrage, élection, HELLO (2)

Entête OSPF type=1 (24o)		
masque sous-réseau		
intervalle Hello	xxxxxxET	priorité
intervalle de mort		
DR (ou 0)		
BR (ou 0)		
(autant que nécessaire)		
@IP routeur voisin		
...		

bit E : émet et reçoit les routes externes

bit T : prise en compte des champs TOS

OSPF : synchronisation, échange

- Initialisation BD topologique : liste des liens, routeurs responsables des MàJ
- Entre routeurs adjacents ou avec DR
- Celui qui prend l'initiative est le maître, l'autre l'esclave (en cas de conflit, @IP)
- Echange de LSA : description des enregistrements
- Construction de la liste des manquants ou des trop vieux
- Demande au voisin

Innondation sans routeur désigné

- Le routeur qui s'appercoit de la MàJ diffuse à ses voisins

↪ Traitement sur réception :

- Rechercher dans la base
- **Si** Pas trouvé

  Insérer l'enregistrement dans la base
  Diffuser le message (sauf sur voie entrante)
- **Sinon Si** Numéro message plus récent

  Remplacer l'enregistrement dans la base
  Diffuser le message (sauf sur voie entrante)
- **Sinon** NOP

Innondation avec routeur désigné

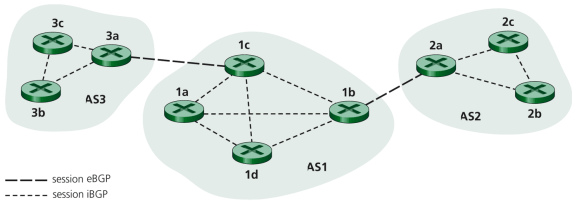
- Le routeur qui s'appercoit de la MàJ envoie au routeur désigné
- Ce dernier rediffuse aux autres routeurs

Routage inter-SA d'Internet : BGP

- BGP *Border Gateway Protocol* : le standard *de facto*
- BGP fournit à chaque SA un moyen de
  1. Savoir comment atteindre un sous-réseau d'un SA voisin
  2. Propager cette information à tous les routeurs du SA
  3. Déterminer les "bonnes" routes vers les sous réseaux en fonction de l'information d'atteignabilité et de la politique choisie
- Permet à un sous-réseau de publier son existence au reste d'Internet ; « Je suis là »

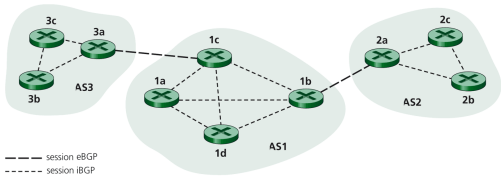
Fondements de BGP

- Des paires de routeurs échangent des informations de routage sur des connexions TCP semi-permanentes : sessions BGP
- Les sessions BGP ne correspondent pas aux liens physiques
- Lorque SA2 publie un préfixe à SA1, SA2 promet qu'il retransmettra tous les datagrammes destinés à ce préfixe par ce préfixe
  - SA2 peut agréger les préfixes dans sa publication



Distribution de l'information d'atteignabilité

- Avec la session eBGP entre 3a et 1c, SA3 envoie l'information d'atteignabilité d'un préfixe à SA1
- 1c peut alors utiliser la session iBGP pour distribuer l'information sur ce nouveau préfixe à tous les routeurs de SA1
- 1b peut ensuite re-publier la nouvelle information vers SA2 par la session eBGP 1b-2a
- Lorsqu'un routeur apprend un nouveau préfixe, il crée une entrée pour ce préfixe dans sa table de routage



Attributs de chemins et routes BGP

- Publication d'un préfixe → ajout d'attributs BGP
  - préfixe+attributs = "route"
- Deux attributs importants
  - AS-PATH : contient les SA par lesquels la publication est passée : SA 67 SA 17
  - NEXT-HOP : indique le routeur interne vers le prochain SA (il peut y avoir plusieurs liens du SA courant vers le prochain SA)
  - Lorsqu'un routeur passerelle reçoit une publication de route, il utilise une politique d'importation pour accepter/refuser

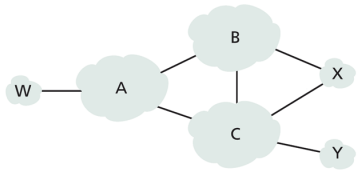
Sélection de route BGP

- Un routeur peut apprendre plus d'une route vers un préfixe
  - Il doit choisir une route
- Règles d'élimination
  1. Attribut de préférence locale : décision politique
  2. AS-PATH le plus court
  3. Routeur NEXT-HOP le plus court : routage en patate chaude
  4. Critère supplémentaire

Messages BGP

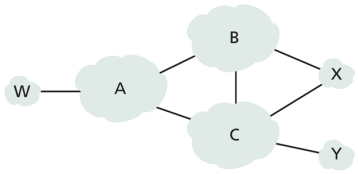
- Messages BGP échangés en utilisant TCP
- Messages BGP
  - OPEN : ouvre la connexion TCP avec le pair et authentifie l'émetteur
  - UPDATE : publie un nouveau chemin (ou supprime l'ancien)
  - KEEPALIVE : garde la connexion vivante en l'absence de mises à jour. Accuse réception d'une requête OPEN
  - NOTIFICATION : rapporte les erreurs des messages précédents. Également utilisé pour clore une connexion

Politique de routage BGP



- A, B et C sont des réseaux fournisseurs
- X, W et Y sont clients
- X est doublement hébergé : rattaché à deux réseaux
  - X ne veut pas router de B vers C en passant par X
  - ...X ne publiera pas de route vers C à l'intention de B

Politique de routage BGP



- A publie le chemin AW vers B
- B publie le chemin BAW vers X
- B doit-il publier le chemin BAW vers C
  - Pas question ! B n'a aucun intérêt à router CBAW puisque ni W ni C ne sont clients de B
  - B veut forcer C à router vers W en passant par A
  - B ne veut router que de et vers ses clients !

Pourquoi des routages inter- et intra-SA différents ?

Politique

- Inter-SA : l'administrateur veut contrôler comment son trafic est routé et qui route à travers son réseau
- Intra-SA : un seul administrateur → pas besoin de décisions politiques

Échelle

- Le routage hiérarchique réduit la taille des tables et le trafic de mise à jour

Performance

- Intra-SA : peut se focaliser sur les performances
- Inter-SA : la politique peut influer sur les performances

Résumé

- Principes derrière les services de la couche réseau
  - Routage (sélection de chemin)
  - Gestion de l'échelle
  - Comment marche un routeur
- Instantiation et implantation sur Internet
  - RIP
  - OSPF
  - BGP

Rappels pour les TPs

- Aspects réseau
  - `mii-tool`
  - `ifconfig`
  - `add route default gw`
- Éditeurs
  - `vim`
  - `emacs`
  - `jed`

Et la suite ?

Multicast

- Mécanismes locaux
- Routage avec DVMRP et PIM-SM

Pair-à-pair

- État de l'art
- Étude de quelques protocoles
  - Chord
  - Pastry