

# Réseaux

## Couche Réseau

E. Jeandel

### Partiel

- Mercredi 10 novembre
- 14h
- Salle 001
- Tout le cours jusqu'aujourd'hui, compris
- Cours non autorisé

## 1 Un routeur

### Un routeur

- Un routeur relie plusieurs réseaux entre eux.
- Il décide pour chaque paquet entrant, sur quel port il va sortir
- Deux tampons (buffer) :
  - Sur le port entrant (avant la décision de routage)
  - Sur le port sortant

### Un routeur

### Table

- Chaque routeur possède une *table de routage*
- Le routeur relaie *forwarde* les paquets en utilisant uniquement la table
- Toute la subtilité est dans le remplissage (et la mise à jour !) de la table
- Entrée par “default” dans la table pour les destinations que le routeur ne connaît pas explicitement

### Taille du paquet

- Vaut-il mieux envoyer plein de petits paquets ou un gros ?

### **Petits messages**

- Chacune des liaisons physiques ne peut transporter qu'une quantité finie de données à la fois (MTU)
- Évite qu'un brin ne soit monopolisé par une seule personne
- Diminue les délais

### **Problématique de la fragmentation**

- Doit-on la faire sur chacun des routeurs ou à la source/destination ?
- IP choisit de la faire à la source/dest
- Champ dans un paquet IP indiquant le numéro du fragment

### **Récapitulatif**

Quand  $A$  veut envoyer un message à  $B$  :

- $A$  fragmente le message en plusieurs paquets IP contenant chacun  $A.ip$  et  $B.ip$
- Chacun des paquets est envoyé au routeur de  $A$  (c'est l'adresse MAC dans la trame Ethernet qui spécifie que le routeur va recevoir le message)
- Le routeur examine  $B.ip$  et décide où transmettre le paquet

Le paquet IP n'est modifié à aucune étape (en tout cas dans cette version simpliste)

## **2 Des routeurs**

### **Deux modes de routage**

Deux modes possibles

- Chacun des paquets de  $A$  à  $B$  est routé indépendamment (IP)
- On établit d'abord un *circuit virtuel* par lequel vont passer tous les paquets de  $A$  à  $B$  (ATM)
  - Permet de donner des garanties (débit, délai...)

Que contient la table dans chacun des cas ?

### **Tables de routage**

Comment construire la table de routage ?

- statique
- dynamique

### **Algorithmes**

Comment choisir un algorithme de routage ?

- Livraison *précise* des paquets
- Choix rapide du routage
- Overhead limité

Il doit s'adapter :

- Au changement de topologie du réseau
- A la densité du trafic
- A la congestion

## Algorithmes

Deux types d'algorithme

- Algorithme centralisé
- Algorithme distribué
  - Connaissance locale
  - Connaissance globale

## Algorithme

Algorithme *du plus court chemin*.

Choix de la métrique

- proportionnel au délai. Comment le calculer ?
- proportionnel à la congestion (trafic)
- inv. proportionnel à la capacité
- égale à 1.

## Bellman-Ford (Algorithme à vecteurs de distance)

Algorithme *distribué*

- Chaque routeur retient, pour chaque routeur destination, par où (routeur suivant) il doit passer, et le coût de la route
- Chaque routeur envoie cette information à ses voisins, qui mettent à jour leurs informations en conséquence

## Bellman-Ford (Démonstration)

### Bellman-Ford

Que se passe-t-il si un routeur disparaît ?

- Ajustement du protocole :  $\infty = 16$
- *horizon éclaté* : On ne transmet pas l'information de distance à un routeur s'il est sur la route

### Bellman-Ford (Inconvénients)

- Convergence lente
- Gros messages
- Problèmes de passage à l'échelle

### Dijkstra (Algorithme à état de lien)

Nécessite une connaissance globale du graphe

- On rajoute à chaque étape le nœud du graphe le plus près

### Dijkstra (Démonstration)

### **Algorithme par inondation**

- But : envoyer un message à tout le réseau
- Moyen comme un autre d'acheminer un paquet
- Chaque routeur envoie le paquet à tous les autres routeurs
- Utilisé dans des situations critiques (e.g. militaires)
- Problème ?
- Solution : Time to Live (TTL)

## **3 Beaucoup de routeurs**

### **La pratique**

La vision est idéaliste :

- Suppose la taille des tables des routeurs très grande
- Suppose que l'acheminement des paquets peut suivre n'importe quelle route

La réalité

- Taille des tables modeste
- Enjeux politiques et économiques

### **Internet**

Internet est constitué de systèmes autonomes (AS) :

- Entité sous le contrôle d'un seul organisme
- Chaque AS a un numéro (ASN) unique
- Organise le routage à l'intérieur comme il souhaite
- Politique de routage interne
- "Seulement" 35000 à l'heure actuelle
- ASN codé sur 16 bits, va passer sur 32 bits

### **AS**

Deux types d'AS

- AS de transit : relié à plusieurs AS, permet le transit des paquets
- AS *stub* ou *multihomed*. AS connecté à (resp.) un ou plusieurs autres AS, mais qui ne permet pas le transit

## A l'intérieur d'un AS

Protocole de routage entre les différents routeurs :

- RIP (Routing Information Protocol)
  - routage par vecteur de distance, avec nombre de saut comme métrique
  - 16 sauts = infini
  - UDP, port 520
- OSPF (Open Shortest Path First)
  - Algorithme par état de lien, obtenu par inondation
  - Fonctionne avec election d'un chef (Designated Router)
  - métrique :  $10^8$ /bande passante
  - Concept de zone (area) (backbone, stub, not-so-stubby, transit)
  - Over IP

Les deux protocoles comprennent CIDR.

## Entre les AS

Protocole BGP (Border Gateway Protocol)

- Pour chaque frontière entre deux AS, un routeur est désigné
- TCP, port 179
- Table de 300000 entrées
- Transfert d'information :
  - "L'AS 1664 permet de transférer des données vers 192.0.16.64/16 en utilisant le routeur 54 pour rentrer dans l'AS, avec la politique suivante"
- Aucune métrique sur le coût à l'intérieur d'un AS. Pourquoi ?
- Métrique simplifiée : nombre d'AS traversés.

# 4 Pour aller plus loin

## 4.1 ICMP

**ICMP (Internet Control Message Protocol)**

- Protocole central de l'Internet
- Erreurs, Diagnostics
- over IP

Exemples :

- Echo request/Echo reply (ping)
- Destination Unreachable
- TTL expired in transit
- Source Quench

Comment programmer l'utilitaire *traceroute* ?

## 4.2 Autres algorithmes

### Autres algorithmes de routage

- Routage par deflection (*hot potato*)
  - Voir dessin au tableau
  - But : éviter les buffers
- Routage de la source
  - La source envoie tout le chemin
  - Quand un routeur reçoit le paquet, il s'enlève du chemin...
  - ... ou alors on garde toute la route (Intérêt ?)
  - Théoriquement possible dans IPv4 et IPv6

### Mobile IP

- Pouvoir changer d'endroit sans changer d'IP ?

Principe :

- Deux adresses : adresse "home" permanente et une adresse temporaire CoA (care-of address)
- Deux intermédiaires : le *home agent* (pays) et le *foreign agent* (consulat)
- Le consulat *B* prévient le pays *A* qu'une machine de *A* vient d'arriver chez lui
- Lorsque le pays *A* reçoit un paquet pour la machine, il le forward (par un tunnel) à *B*.

Ne peut pas marcher sans sécurité.