

# Protocoles de routage



## ■ Introduction

Un réseau informatique permet de relier les différentes machines afin qu'elles puissent communiquer.

Un message échangé entre deux machines est découpée en paquets, et chacun de ces paquets transite indépendamment des autres sur le réseau de la machine émettrice à la machine destinataire qui reconstitue le message à partir des différents paquets.

Un réseau est en réalité un ensemble de sous-réseaux interconnectés par des machines particulières appelées **routeurs**. Les interconnexions peuvent être de natures diverses : Ethernet, Wi-Fi, fibre optique, câble téléphonique, liaison par satellite, etc.



Un routeur CISCO



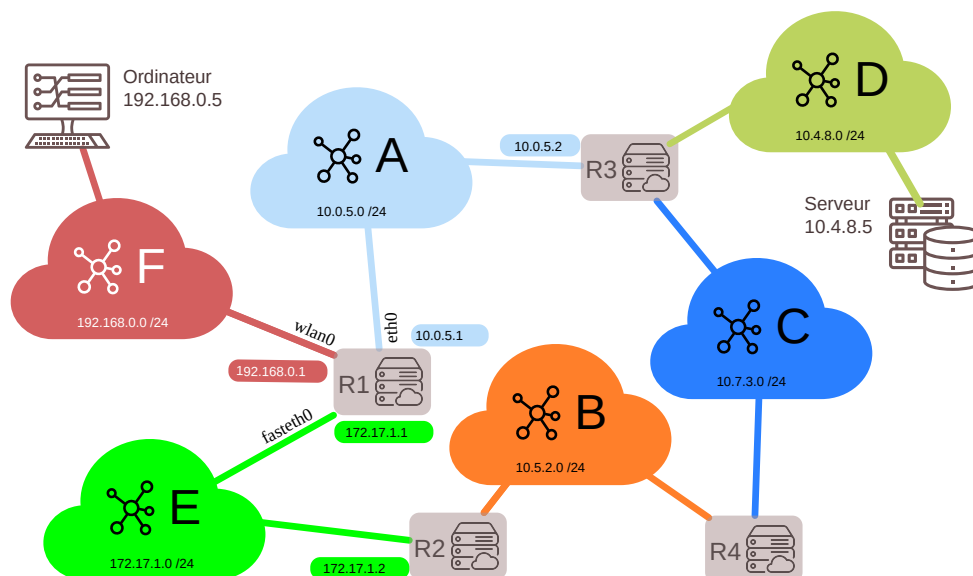
Un routeur Wi-Fi

Ce sont ces routeurs qui jouent un rôle essentiel dans la transmission des paquets sur Internet puisque ce sont eux qui déterminent la *meilleure* route que doit emprunter un paquet pour aller jusqu'à destination. Pour cela, chaque routeur dispose d'une **table de routage** qui peut être définie manuellement (par un administrateur, souvent pour des petits réseaux comme celui d'une entreprise, d'une école, etc.) ou dynamiquement grâce à des **algorithmes de routage** spécifiques. Nous étudierons dans ce chapitre deux protocoles de routage dynamique : RIP et OSPF.

## ■ Topologie d'un réseau

L'interconnexion des routeurs forme ce que l'on appelle la *topologie* d'un réseau.

Dans la suite, on considérera le réseau suivant (adapté du cours de Gilles Lassus, lien dans les références à la fin) :



Dans ce réseau, il y a :

- 4 routeurs : R1, R2, R3 et R4 ;
- 6 sous-réseaux : A, B, C, D, E et F.

Chaque routeur possède au moins deux cartes réseaux qui lui permet de relier au moins deux sous-réseaux (sinon, il ne sert à rien). Ainsi, un routeur appartient à chacun des sous-réseaux qu'il relie et possède donc plusieurs adresses IP : une par sous-réseaux auquel il appartient.

Par exemple, le routeur R1 :

- interconnecte trois réseaux : `192.168.0.0/24` (F), `10.0.5.0/24` (A) et `172.17.1.0/24` (E)
- possède trois adresses : `192.168.0.1` (son adresse dans le sous-réseau F), `10.0.5.1` (son adresse dans le sous-réseau A) et `172.17.1.1` (son adresse dans le sous-réseau E).

Par soucis de simplification, on a donné des noms (A, B, ...) à chaque sous-réseau mais il faut savoir qu'en réalité c'est l'adresse IP de chaque sous-réseau qui est utilisée. Par exemple, le sous-réseau A possède l'adresse IP `10.0.5.0` et le masque de ce sous-réseau est `24`.

## ■ Tables de routage

Imaginons que l'ordinateur (client) d'adresse `192.168.0.5` veut interroger le serveur `10.4.8.5`.

Comme l'adresse IP du serveur n'est pas dans le sous-réseau F, alors l'ordinateur transmet le message au routeur R1 de son réseau local. Ce dernier regarde alors dans sa table de routage pour déterminer vers quel routeur voisin il doit transmettre le message. Chaque routeur réceptionnant le message procède ensuite de la même manière jusqu'à atteindre le dernier routeur appartenant au réseau local de l'adresse recherchée.

La table de routage d'un routeur contient plusieurs colonnes :

- une *destination* sous la forme d'une adresse sous-réseaux/masque ;
- une *passerelle* qui donne l'adresse IP du prochain routeur voisin pour atteindre cette destination ;
- une *interface* qui indique la carte réseau (ou interface réseau) à utiliser pour atteindre la passerelle ;
- une *métrique* indiquant le "coût" pour atteindre la destination.

Par exemple, la table de routage de R1 pourrait être :

| Destination (@IP ss-rés) | Passerelle (@IP routeur) | Interface | Métrique |
|--------------------------|--------------------------|-----------|----------|
| 192.168.0.0 /24 (F)      |                          | wlan0     |          |
| 172.17.1.0 /24 (E)       |                          | fasteth0  |          |
| 10.0.5.0 /24 (A)         |                          | eth0      |          |
| 10.5.2.0 /24 (B)         | 172.17.1.2 (R2)          | fasteth0  |          |
| 10.7.3.0 /24 (C)         | 10.0.5.2 (R3)            | eth0      |          |
| 10.4.8.0 /24 (D)         | 10.0.5.2 (R3)            | eth0      |          |

L'interface wlan0 indique que le routeur R1 est connecté au réseau 192.168.0.0 /24 via une interface sans fil (wlan pour *Wireless Local Area Network*). Les interfaces fasteth0 et eth0 indiquent que le routeur R1 est connecté respectivement à chacun des réseaux 172.17.1.0 /24 et 10.0.5.0 /24 par une carte FastEthernet (fasteth) et Ethernet (eth), celles de numéro 0.

On a laissé la colonne *Métrique* vide car cela va dépendre du protocole utilisé, comme on le verra par la suite.

## Routage statique vs routage dynamique

Les tables de routage peuvent être construites de manière *statique* ou de manière *dynamique*.

Dans le cas d'un routage statique, les tables sont définies manuellement par l'administrateur. C'est simple à mettre en oeuvre et le contrôle est facile ce qui amène de la sécurité. En revanche, ce n'est adapté qu'à de petits réseaux (routage interne) comme le réseau d'une (petite) entreprise ou d'une école car il faut mettre à jour (manuellement) les tables de routage à chaque nouvel équipement installé, à chaque panne ou équipement défectueux.

Dans le cas d'un routage dynamique, les routeurs diffusent et partagent des informations pour indiquer quels réseaux sont atteignables par eux. En fonction des informations reçues chaque routeur peut alors mettre à jour dynamiquement sa table de routage grâce à des algorithmes de routage spécifiques. Ce sont des algorithmes de meilleur chemin qui sont utilisés, et qui permettent de définir les différentes métriques.

Nous allons maintenant voir deux routages dynamiques très connus mis en oeuvre par les **protocoles RIP et OSPF**.

## ■ Le protocole RIP

Le **protocole RIP**, pour *Routing Information Protocol* (ou "protocole d'information de routage" en français), a été conçu dès l'origine d'Internet pour fonctionner sur des réseaux de taille modérée.

Il utilise l'algorithme de Bellman-Ford qui prévoit que chaque routeur diffuse régulièrement (toutes les 30 secondes) à ses voisins les routes qu'il connaît, c'est-à-dire la liste des réseaux qu'il peut atteindre et la distance, en **nombre de sauts**, pour les atteindre.

Ainsi, chaque routeur du réseau reçoit périodiquement (30 sec.) les routes des routeurs voisins. Il les utilise pour effectuer les traitements suivants :

- il met alors à jour sa propre table de routage :
  - si une route reçue comprend un chemin plus court que celui qu'il connaît : il modifie sa table avec ce plus court chemin
  - si une route n'est pas connue, il l'ajoute à sa table en ajoutant 1 au nombre de sauts du voisin qui lui a transmis la route
  - sinon, il n'y a pas de changement
- il émet sa table de routage à tous ses routeurs voisins

Si un routeur n'a aucune nouvelle d'un voisin pendant 3 minutes (180 sec., c'est le *premier timeout*), il considère que ce voisin est en panne, toutes les routes passant par lui sont affectées dans la distance infinie (égale à 16, voir en-dessous) et l'information est diffusée à tous ses voisins qui diffuseront aussi l'information à leurs voisins, etc. Au bout d'un certain temps supplémentaire (*deuxième timeout*), si un routeur n'a pas de nouvelle d'un voisin à distance 16, la route est complètement supprimée de sa table.

L'émission périodique des tables permet ainsi de prendre en compte :

- les évolutions du réseaux (apparition/disparition d'un routeur) ;
- la coupure des liaisons ;
- la panne d'un ou plusieurs routeurs.

Si tout se passe bien sur le réseau, les tables de routages *convergent* rapidement. La convergence est plus lente si des pannes ont lieu ou si de nouveaux routeurs apparaissent.

Par exemple, la table de routage du routeur R1 du réseau précédent peut être la suivante (après convergence !) :

| Destination (@IP ss-rés) | Passerelle (@IP routeur) | Interface | Nombre de sauts (métrique) |
|--------------------------|--------------------------|-----------|----------------------------|
| 192.168.0.0 /24 (F)      |                          | wlan0     | 0                          |
| 172.17.1.0 /24 (E)       |                          | fasteth0  | 0                          |
| 10.0.5.0 /24 (A)         |                          | eth0      | 0                          |
| 10.5.2.0 /24 (B)         | 172.17.1.2 (R2)          | fasteth0  | 1                          |
| 10.7.3.0 /24 (C)         | 10.0.5.2 (R3)            | eth0      | 1                          |
| 10.4.8.0 /24 (D)         | 10.0.5.2 (R3)            | eth0      | 1                          |

Ainsi, avec le protocole RIP, si l'ordinateur d'adresse 192.168.0.5 veut interroger le serveur 10.4.8.5, le chemin utilisé sera R1-R3, pour un coût égal à 1 (qui est meilleur que le chemin R1-R2-R4-R3 de coût égal à 3).



Dans certains exercices, on compte aussi le routeur de départ pour la métrique, on obtient alors un décalage de 1 :

| Destination (@IP ss-rés) | Passerelle (@IP routeur) | Interface | Nombre de sauts (métrique) |
|--------------------------|--------------------------|-----------|----------------------------|
| 192.168.0.0 /24 (F)      |                          | wlan0     | 1                          |
| 172.17.1.0 /24 (E)       |                          | fasteth0  | 1                          |
| 10.0.5.0 /24 (A)         |                          | eth0      | 1                          |
| 10.5.2.0 /24 (B)         | 172.17.1.2 (R2)          | fasteth0  | 2                          |
| 10.7.3.0 /24 (C)         | 10.0.5.2 (R3)            | eth0      | 2                          |
| 10.4.8.0 /24 (D)         | 10.0.5.2 (R3)            | eth0      | 2                          |

C'est une convention sur laquelle il suffit de se mettre d'accord au départ. Cela ne change rien sur la route empruntée puisque toutes les métriques sont augmentées de 1.

### Remarques et inconvénients

- La *métrique* utilisée par le protocole RIP est donc la distance en *nombre de sauts*, autrement dit le nombre de routeurs traversés pour atteindre la destination (en comptant le routeur de départ).

👉 *Inconvénient* : le protocole RIP ne tient alors pas compte de la qualité des liaisons entre les routeurs : pourtant, il se peut très bien qu'une route plus *longue* (qui traverse davantage de routeurs) soit en réalité plus *rapide*

- La distance maximale autorisée par le protocole RIP est égale à 15, et ainsi 16 correspond à la distance infinie.

👉 *Inconvénient* : cela limite l'utilisation du protocole RIP à des réseaux de petite taille

- Les informations ne sont échangées qu'entre voisins directs

👉 *Inconvénient* : Un routeur n'a pas de vision au-delà de ses propres voisins et n'a donc jamais connaissance de la topologie du réseau tout entier.

Les inconvénients évoqués ci-dessus (et ce ne sont pas les seuls), et en particulier la non-prise en compte de la bande passante reliant les routeurs (donc la qualité des liaisons) font que d'autres protocoles ont vu le jour. En particulier, le protocole OSPF a vu le jour dans l'idée de remplacer le protocole RIP.

## ■ Le protocole OSPF

Le **protocole OSPF**, pour *Open Shortest Path First*, a été développé à partir de 1997 avec l'objectif de remplacer le protocole RIP dont on vient de citer quelques inconvénients.

Dans le protocole OSPF, et cela constitue deux différences fondamentales avec RIP, chaque routeur :

- possède une **vision globale** de la topologie du réseau
- possède une table de routage qui tient compte de la **vitesse de communication** entre les routeurs, c'est-à-dire de la qualité des liaisons (bande passante)



En particulier, les tables de routages du protocole OSPF sont identiques pour tous les routeurs.

Le protocole OSPF se décompose en deux phases importantes :

- 1ère phase : les routeurs s'échangent des messages afin de **connaître la topologie complète du réseau** (= toutes les liaisons ainsi que leurs débits)
- 2ème phase : chaque routeur applique ensuite un **algorithme afin de calculer les chemins les plus rapides** vers les différents réseaux

Plus précisément, chaque routeur :

- maintient une carte complète du réseau (topologie)
- calcule les meilleurs chemins localement
- teste périodiquement l'état des liaisons qui le relie à ses voisins et diffuse cet état à tous les routeurs du domaine.
- à la réception d'un message :
  - met à jour la carte des liaisons
  - recalcule, pour chaque liaison modifiée, la nouvelle meilleure route

## Les coûts des liaisons selon leur type

Voici un tableau regroupant les différents types de liaison ainsi que leurs débits théoriques :

| Technologie   | BP descendante         | BP montante            |
|---------------|------------------------|------------------------|
| Modem         | 56 kbit/s              | 48 kbit/s              |
| Bluetooth     | 3 Mbit/s               | 3 Mbit/s               |
| Ethernet      | 10 Mbit/s              | 10 Mbit/s              |
| Wi-Fi         | 10 Mbit/s ~ 10 Gbits/s | 10 Mbit/s ~ 10 Gbits/s |
| ADSL          | 13 Mbit/s              | 1 Mbit/s               |
| 4G            | 100 Mbit/s             | 50 Mbit/s              |
| Satellite     | 50 Mbit/s              | 1 Mbit/s               |
| Fast Ethernet | 100 Mbit/s             | 100 Mbit/s             |
| FFTH (fibre)  | 10 Gbit/s              | 10 Gbit/s              |
| 5G            | 20 Gbit/s              | 10 Gbit/s              |

Le protocole OSPF associe un **coût** à chaque type de liaison. Ce coût est inversement proportionnel au débit de transfert et s'obtient généralement par la formule :

$$\text{coût} = \frac{10^8}{\text{débit}}$$

où **débit** est exprimé en bit/s.

Ainsi, plus le débit d'une liaison est important, plus le coût associé est faible. Les liaisons les plus rapides sont donc celles qui auront le coût le plus faible et donc celles privilégiées dans la recherche du meilleur chemin.



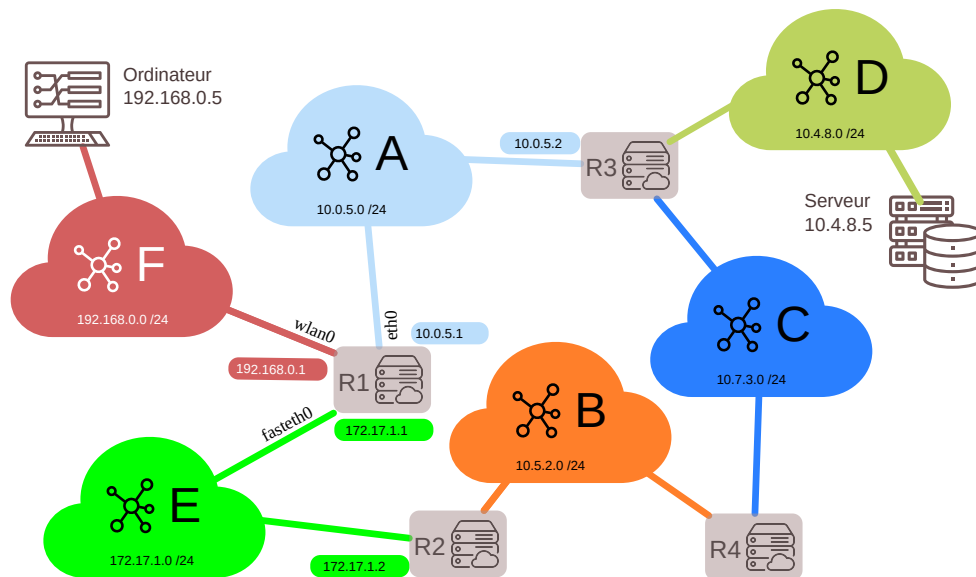
Cette formule peut varier suivant les exercices, celle à utiliser sera indiquée dans l'énoncé (c'est le numérateur qui peut varier).

Avec cette formule :

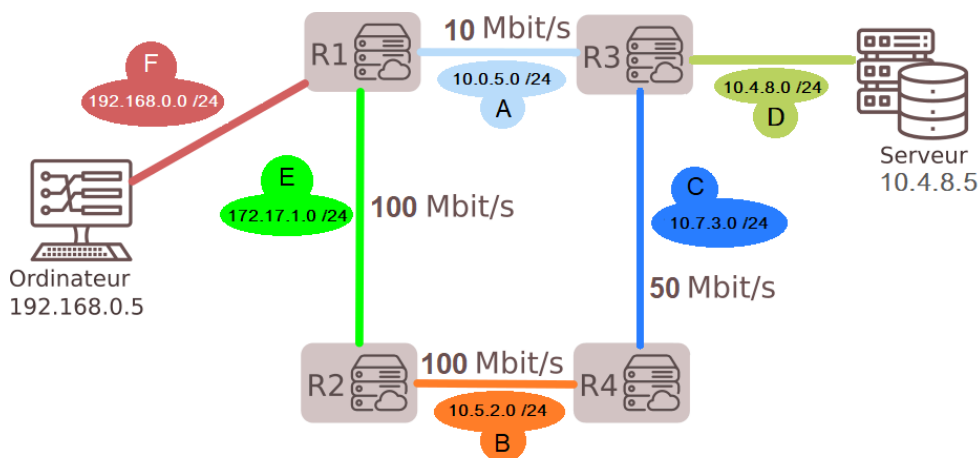
- une liaison *Fast Ethernet* de débit 100 Mbits/s, c'est-à-dire  $100 \times 10^6 = 10^8$  bit/s, aura un coût égal à  $\frac{10^8}{10^8} = 1$ .
- une liaison *Ethernet* de débit 10 Mbits/s, c'est-à-dire  $10 \times 10^6 = 10^7$  bits/s, aura un coût égal à  $\frac{10^8}{10^7} = 10$ .
- etc.

## Exemple

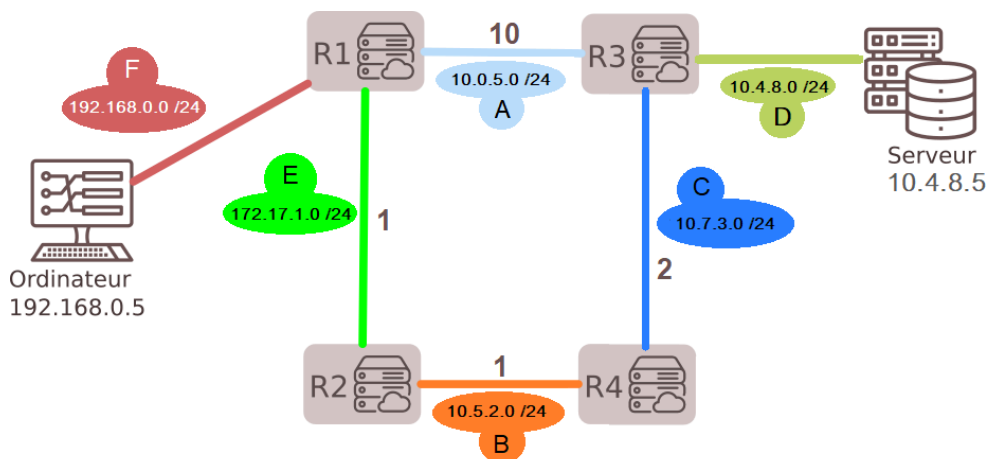
Reprenons le réseau initial, en indiquant les débits entre les différents routeurs.



On simplifie sa représentation en dessinant uniquement les réseaux ainsi que les routeurs et les *débits* entre eux.



On peut alors calculer le coût de chaque liaison pour obtenir le *graphe pondéré* suivant :



## Construction de la topologie du réseau

En pratique, avec OSPF les routeurs communiquent entre eux pour déterminer la topologie complète du réseau :

- au démarrage, un routeur envoie des messages (que l'on appelle *messages HELLO*) à tous les routeurs auxquels il est connecté.
- Ces derniers lui répondent et permettent ainsi au routeur de départ de prendre connaissance de ses voisins. Comme il connaît les bandes passantes vers chacun de ces voisins, il peut initialiser sa vision de la topologie du réseau dans sa table de routage avec les liaisons vers ses voisins.
- Après la phase d'initialisation, les routeurs s'échangent des paquets (que l'on appelle *paquets LSA*, pour *Link State Advertisement*) contenant leurs connaissances du réseau. Ainsi, au bout d'un certain nombre d'échanges, tous les routeurs du réseau ont la même vision du réseau (celui correspondant à leur zone).

Dans le cas du réseau précédent, lorsque les échanges sont terminés, la topologie du réseau vue par R1 (ou tout autre routeur de la zone) est la suivante :

| Lien  | Sous-réseau    | Coût |
|-------|----------------|------|
| R1-R2 | 172.17.1.0 /24 | 1    |
| R1-R3 | 10.0.5.0 /24   | 10   |
| R2-R4 | 10.5.2.0 /24   | 1    |
| R3-R4 | 10.7.3.0 /24   | 2    |

## Calculs des plus courts chemins

La topologie du réseau étant connue, on peut passer à la dernière phase du protocole OSPF : chaque routeur applique un algorithme pour déterminer les plus courts chemins vers chaque sous-réseau et construit sa table de routage avec les meilleures routes.

| Destination (@IP ss-rés) | Passerelle (@IP routeur) | Interface | Coût (métrique) |
|--------------------------|--------------------------|-----------|-----------------|
| 192.168.0.0 /24 (F)      |                          | wlan0     | 0               |
| 172.17.1.0 /24 (E)       |                          | fasteth0  | 0               |
| 10.0.5.0 /24 (A)         |                          | eth0      | 0               |
| 10.5.2.0 /24 (B)         | 172.17.1.2 (R2)          | fasteth0  | 1               |
| 10.7.3.0 /24 (C)         | 172.17.1.2 (R2)          | fasteth0  | 2               |
| 10.4.8.0 /24 (D)         | 172.17.1.2 (R2)          | fasteth0  | 4               |

Dans notre exemple, on se rend compte que pour aller de l'ordinateur au serveur, le chemin le plus rapide n'est plus R1-R3 (avec le protocole RIP), mais R1-R2-R4-R3 avec un coût de 4 (contre 10 pour R1-R3).



Dans l'exemple précédent les coûts d'accès aux différents réseaux n'ont pas été indiqués. Ils sont généralement communs aux différentes routes potentielles et n'entrent pas en compte pour calculer les meilleures routes. Par contre, si un réseau est accessible via deux routeurs avec des débits différents, il faudrait en tenir compte.

Ici, le réseau est *petit* pour l'exemple, et un humain peut rapidement déterminer le chemin le plus court en un coup d'oeil. En réalité, c'est un algorithme qui calcule le chemin le plus court dans le graphe pondéré obtenu, qui peut être bien plus complexe.

## Algorithme de Dijkstra

Le protocole OSPF est basé sur l'**algorithme de Dijkstra** pour trouver le chemin le plus court (= le plus rapide !). C'est un algorithme découvert en 1959 par le mathématicien et informaticien néerlandais [Edsger Dijkstra](#).

Cet algorithme n'est *a priori* pas au programme de Terminale NSI, mais son principe est très simple à comprendre et à appliquer. Voici une [vidéo](#) présentant son fonctionnement :

Source vidéo : <https://youtu.be/rHyICtXdNs>

## ■ Routage mondial

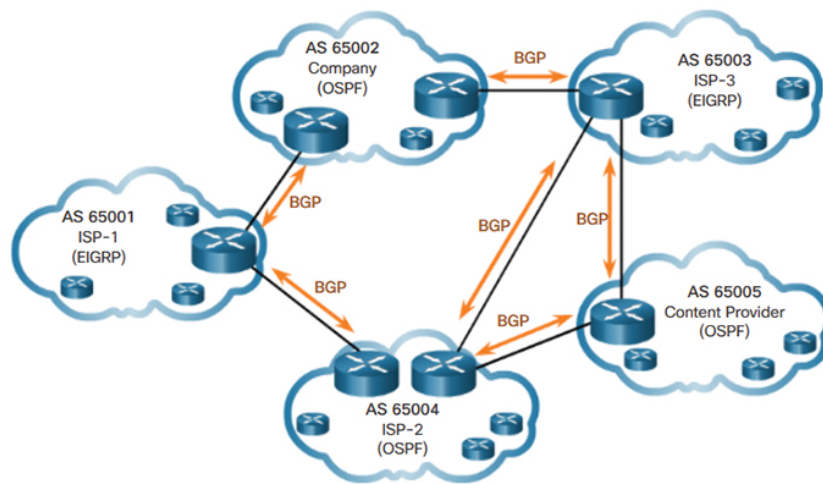
Le monde IP est divisé en *domaines administratifs* (ou *Administrative System* en anglais) qui sont des zones dans lesquelles n'exerce qu'une seule autorité (ex: FAI ou ISP en anglais, pour *Internet Service Provider*) ayant différentes tâches comme l'administration des adresses, la facturation des coûts, la sécurisation et l'organisation des domaines de routage.

Chaque domaine administratif est lui-même divisé en différents **Systèmes Autonomes** (ou *Autonomous System* en anglais, souvent abrégé **AS**). Un Système Autonome est :

- un ensemble de réseaux interconnectés partageant la même stratégie de routage (ex : entreprise, campus, coeur d'un réseau national, FAI, ...)
- identifié par un numéro unique (ASN, pour *Autonomous System Number*)
- dans lequel tous les routeurs internes à ce système obéissent au même protocole de routage (comme RIP et OSPF, mais il en existe d'autres : IGRP, EIGRP, IS-IS).

Les AS sont reliés entre eux par des routeurs externes chargés de transférer les informations de routage entre les différents AS (ces routeurs externes utilisent d'autres protocoles comme BGP pour *Border Gateway Protocol*, mais ce n'est pas le seul).





Le routage mondial.

Source : [ciscopress.com](http://ciscopress.com)



Ce que nous avons raconté dans ce cours sur RIP et OSPF s'applique donc à des réseaux qui sont des AS.

En 2019, il y avait plus de 91 000 AS contre 35 000 en 2010 et 5 000 en 1999. Pour plus d'informations, voir cet article Wikipédia : [Autonomous System](https://fr.wikipedia.org/wiki/Autonomous_System).

## ■ Bilan

- Les machines communiquent entre elles grâce à des **routeurs**.
- Les routeurs utilisent leur **table de routage** pour déterminer la *meilleure* route pour acheminer les paquets jusqu'à destination. Ces tables peuvent être définies manuellement, ce qui reste adapté sur un domaine restreint (routage interne) mais qui est inadapté pour le réseau mondial où les tables doivent évoluer dynamiquement.
- **RIP** et **OSPF** sont deux protocoles de *routage dynamique* dans lesquels les routeurs s'échangent périodiquement des messages pour indiquer quels réseaux on peut atteindre par eux, permettant de mettre à jour dynamiquement les tables de routage grâce à des **algorithmes de routage** spécifiques. Ces algorithmes utilisent une **métrique** pour mesurer la *distance* entre un routeur et le réseau de destination, et permettent notamment une reconfiguration rapide en cas de **changement de topologie** du réseau.
- Dans le protocole RIP, les routeurs connaissent seulement leurs voisins et n'ont pas connaissance de la topologie complète du réseau. L'algorithme utilisé (Bellman-Ford) tient compte du **nombre de sauts** (= métrique) pour trouver la meilleure route : celle dont le nombre de sauts est minimum, autrement dit la plus *courte*. C'est l'algorithme de routage dynamique le plus ancien, le plus simple et il est encore très utilisé. En revanche, il ne tient pas compte de la qualité des liens entre les routeurs (débit ou bande passante) et n'est pas très adapté pour des réseaux de grande taille.
- Dans le protocole OSPF, créé pour remplacer RIP, la qualité des liens est prise en compte et les routeurs ont une vision complète de la topologie du réseau. L'algorithme utilisé (Dijkstra) tient compte du **coût** (= métrique), qui dépend de la bande passante des liens franchis, pour trouver la meilleure route : celle dont le coût est minimal, autrement dit la plus *rapide*. OSPF est très complet, très performant mais plus compliqué et consommateur de ressources dans les routeurs, on le réserve généralement à de grands réseaux.

### Références :

- Equipe éducative DIU EIL, cours d'Introduction aux réseaux de P. PASSARD et S. HAMMA, Université de Nantes.
- Cours de Gilles Lassus sur les [protocoles de routage](#) notamment pour les schémas
- Livre *Numérique et Sciences Informatiques, 24 leçons, Terminale*, T. BALABONSKI, S. CONCHON, J.-C. FILLIATRE, K. NGUYEN, éditions ELLIPSES.
- Cours d'Olivier Lecluse sur les [protocoles de routage](#)
- Cours de Denis Quenton sur les protocoles [RIP](#) et [OSPF](#)

Germain BECKER, Lycée Mounier, ANGERS

Ressource éducative libre distribuée sous [Licence Creative Commons Attribution - Pas d'Utilisation Commerciale - Partage dans les Mêmes Conditions 4.0 International](https://creativecommons.org/licenses/by-nc-sa/4.0/)

