



# Le routage IGP-réseau d'un ISP ou FAI

20.02.2023

*L'ancêtre du GPS☹****Non, l'autre droite....***

Auteur : Pascal Fougeray

source : <http://www.dudelire.com/nimportekeoi/circulation-anglaise,37.html>

## 1 Introduction

Le routage est aux réseaux ce que les processus sont au processeur... Bon ça ne veut rien dire.  
Le routage c'est se poser la question suivante :

***Par où je passe pour aller là ?***

Bon ce n'est pas encore terrible mais ça s'améliore, non ?

Comme nous l'avons vu dans le cours intitulé Internet, les réseaux contiennent des adresses et à ces adresses se trouvent des entités.

Le souci c'est qu'il existe des dizaines, que dis-je des centaines, que dis-je des milliers, que dis-je beaucoup de routes possibles pour y aller.

Pas loin de 1 million actuellement!!!

Alors laquelle prendre ?

**réponse** : ça dépend...

Elle est nulle l'introduction du prof, on a rien compris.

**Rassurez** vous le prof non plus et comment introduire le routage ?

**Internet ce sont des milliers de réseaux (Backbone) et chaque réseau contient plein de routes et de routeurs !**

**Entre 2 routes se trouve un routeur et entre 2 routeurs se trouve un réseau**



**Les réseaux sont comme des boîtes qui contiennent des boîtes, un réseau contient des réseaux, un réseau contient des routeurs et ce que vous n'avez pas encore vu, c'est qu'un routeur peut contenir des routeurs... virtuels...**

## 2 IGP et/ou EGP

Dans le routage il existe 2 catégories de protocoles, les IGP (**Interior Gateway Protocol**) pour et les EGP pour (**Exterior Gateway Protocol**)

Dans ce cours, nous allons voir que les IGP.

En EGP il n'en existe qu'un seul, c'est BGP, voir cours suivant ☺.

Les protocoles de routage interne utilisent les adresses **multicast** de lien local ! **224.0.0.0/24** et non 239.0.0.0/24

Ces @IP de type multicast ne peuvent traverser les routeurs, d'où le terme lien local.

- **224.0.0.4** et ff02 : :5 (SPF) et ff02 : :6 (DR) pour **OSPF** (on va le voir)
- **224.0.0.9** et ff02 : :9 pour **RIP** (on va le voir)
- **224.0.0.18** pour **VRRP** (Ce n'est pas un protocole de routage à proprement parlé mais on va le voir)
- **224.0.0.19-21** et ff02 : :8 pour **IS-IS** (on ne va pas le voir, pour 2 raisons, je ne le "domine pas et les routeurs Mikrotik non plus ☺)

## 3 Le routage

Le routage est le mécanisme par lequel des **chemins** sont sélectionnés dans un réseau pour acheminer les données d'un expéditeur jusqu'à un ou plusieurs destinataires.

C'est une tâche (processus ) exécutée dans les réseaux, tels que

- les réseaux téléphoniques,
- le réseau Internet,
- et les réseaux de transports.

**Chaque entité, Ordinateur et routeur, dans le monde du réseau possède une table de routage !**

Table de routage que l'on peut voir à l'aide des commandes suivantes :

— Sous **Linux** : **ip route ls**

```
$ ip route ls
default via 10.38.16.1 dev eth0
10.38.16.0/22 dev eth0 proto kernel scope link src 10.38.19.111
169.254.0.0/16 dev eth0 scope link metric 1000
172.17.0.0/16 dev docker0 proto kernel scope link src 172.17.0.1 linkdown
192.168.56.0/24 dev vboxnet0 proto kernel scope link src 192.168.56.1 linkdown
fougeray@C304L-159C00:~$
```

— Sous windows : **route PRINT -4 ou netstat -r**

```
C:\WINDOWS\system32>route PRINT -4
=====
Liste d'Interfaces
25...c8 d3 ff 1f 67 40 .....Intel(R) Ethernet Connection (2) I219-LM
7...0a 00 27 00 00 07 .....Mpcap Loopback Adapter
17...0a 00 27 00 00 11 .....VirtualBox Host-Only Ethernet Adapter #3
16...e4 a7 a0 6c 02 56 .....Microsoft Wi-Fi Direct Virtual Adapter #4
14...e6 a7 a0 6c 02 55 .....Microsoft Wi-Fi Direct Virtual Adapter #5
20...00 50 56 c0 00 01 .....VMware Virtual Ethernet Adapter for VMnet1
23...00 50 56 c0 00 08 .....VMware Virtual Ethernet Adapter for VMnet8
26...e4 a7 a0 6c 02 55 .....Intel(R) Dual Band Wireless-AC 8260
1.....Software Loopback Interface 1
=====

IPv4 Table de routage
=====
Itinéraires actifs :
Destination réseau    Masque réseau    Adr. passerelle    Adr. interface    Métrique
0.0.0.0              0.0.0.0          192.168.1.254      192.168.1.18      25
127.0.0.0            255.0.0.0        On-link            127.0.0.1         331
127.0.0.1            255.255.255.255  On-link            127.0.0.1         331
127.255.255.255      255.255.255.255  On-link            127.0.0.1         331
192.168.1.0          255.255.255.0    On-link            192.168.1.18      281
192.168.1.18         255.255.255.255  On-link            192.168.1.18      281
192.168.1.255        255.255.255.255  On-link            192.168.1.18      281
192.168.56.0         255.255.255.0    On-link            192.168.56.1      281
192.168.56.1         255.255.255.255  On-link            192.168.56.1      281
192.168.56.255       255.255.255.255  On-link            192.168.56.1      281
192.168.83.0         255.255.255.0    On-link            192.168.83.1      291
192.168.83.1         255.255.255.255  On-link            192.168.83.1      291
192.168.83.255       255.255.255.255  On-link            192.168.83.1      291
192.168.109.0        255.255.255.0    On-link            192.168.109.1     281
192.168.109.1        255.255.255.255  On-link            192.168.109.1     281
192.168.109.255      255.255.255.255  On-link            192.168.109.1     281
192.168.203.0        255.255.255.0    On-link            192.168.203.1     291
192.168.203.1        255.255.255.255  On-link            192.168.203.1     291
192.168.203.255      255.255.255.255  On-link            192.168.203.1     291
224.0.0.0            240.0.0.0        On-link            127.0.0.1         331
224.0.0.0            240.0.0.0        On-link            192.168.109.1     281
224.0.0.0            240.0.0.0        On-link            192.168.56.1      281
224.0.0.0            240.0.0.0        On-link            192.168.1.18      281
224.0.0.0            240.0.0.0        On-link            192.168.203.1     291
224.0.0.0            240.0.0.0        On-link            192.168.83.1      291
255.255.255.255      255.255.255.255  On-link            127.0.0.1         331
255.255.255.255      255.255.255.255  On-link            192.168.109.1     281
```

### — Routeurs Mikrotik : *ip route print*

```
[admin@Caen] > ip route print
Flags: X - disabled, A - active, D - dynamic,
C - connect, S - static, r - rip, b - bgp, o - ospf, m - mme,
B - blackhole, U - unreachable, P - prohibit
# DST-ADDRESS      PREF-SRC  GATEWAY      DISTANCE
0 ADC 1.1.1.1/32        1.1.1.1     10           0
1 ADo 2.2.2.2/32        10.0.0.2    110          0
2 ADo 3.3.3.3/32        10.0.0.2    110          0
3 ADC 10.0.0.0/30        10.0.0.1     ether1        0
4 ADC 10.0.0.4/30        10.0.0.5     ether2        0
5 ADC 10.0.0.8/30        10.0.0.9     ether3        0
6 ADo 10.0.0.12/30       10.0.0.2     110          0
7 ADo 10.0.0.16/30       10.0.0.2     110          0
8 ADo 10.0.0.20/30       10.0.0.2     110          0
9 ADo 10.0.0.24/30       10.0.0.2     110          0
10 ADo 10.0.0.28/30      10.0.0.2     110          0
```

### — Routeurs Cisco : *show ip route*

```
R1#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area

Gateway of last resort is not set

10.0.0.0/8 is variably subnetted, 13 subnets, 2 masks
C    10.12.0.2/31 is directly connected, GigabitEthernet4/0
O    10.2.2.2/32 [110/2] via 10.12.0.2, 00:04:26, GigabitEthernet4/0
O    10.3.3.3/32 [110/3] via 10.12.0.2, 00:04:26, GigabitEthernet4/0
C    10.1.1.1/32 is directly connected, Loopback0
O    10.6.6.6/32 [110/3] via 10.12.0.2, 00:04:26, GigabitEthernet4/0
O    10.4.4.4/32 [110/4] via 10.12.0.2, 00:04:26, GigabitEthernet4/0
O    10.5.5.5/32 [110/5] via 10.12.0.2, 00:04:26, GigabitEthernet4/0
O    10.26.0.2/31 [110/2] via 10.12.0.2, 00:04:26, GigabitEthernet4/0
O    10.23.0.2/31 [110/2] via 10.12.0.2, 00:04:26, GigabitEthernet4/0
O    10.45.0.2/31 [110/4] via 10.12.0.2, 00:04:26, GigabitEthernet4/0
O    10.46.0.2/31 [110/3] via 10.12.0.2, 00:04:26, GigabitEthernet4/0
O    10.34.0.2/31 [110/3] via 10.12.0.2, 00:04:26, GigabitEthernet4/0
O    10.36.0.2/31 [110/3] via 10.12.0.2, 00:04:27, GigabitEthernet4/0
...
```

### — Routeurs Juniper : *show route table*



```

user@router> show route

inet.0: 9 destinations, 9 routes (9 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

10.10.10.91/32      *[Direct/0] 00:09:40
                   > via lo0.0
10.10.10.92/32      *[OSPF/10] 00:01:50, metric 1
                   > to 172.16.1.2 via ge-0/0/2.0
100.100.1.0/24      *[Static/5] 00:01:50
                   Reject
172.16.1.0/24       *[Direct/0] 00:06:09
                   > via ge-0/0/2.0
172.16.1.1/32       *[Local/0] 00:06:09
                   Local via ge-0/0/2.0
192.168.0.0/16      *[Aggregate/130] 00:00:06
                   Reject
192.168.0.0/17      *[Aggregate/130] 00:00:06
                   > to 172.16.1.2 via ge-0/0/2.0
192.168.50.0/24     *[Static/5] 00:00:06
                   > to 172.16.1.2 via ge-0/0/2.0
192.168.51.0/24     *[Static/5] 00:00:06
                   > to 172.16.1.2 via ge-0/0/2.0

```

— etc...

La table de routage contient :

- les adresses du routeur ou de l'ordinateur,
- les adresses des **sous-réseaux** auxquels le routeur est directement connecté,
- les routes **statiques**, celles configurées explicitement,
- les routes **dynamiques**, apprises par des protocoles de routage dynamique tels BGP, OSPF, IS-IS, RIP, etc...
- une route par **défaut**.!!!

### 3.1 Objectifs des protocoles de routage

1. Le rôle principal est de **découvrir dynamiquement** les routes vers les réseaux d'un **inter réseau** et les inscrire dans la **table de routage sur routeur**. **Je parle ici de routage en IGP** (Voir plus loin)
2. S'il existe plus d'une route vers un réseau, inscrire **la meilleure route** dans la table de routage.
3. **Détecter** les routes **invalides** et les **supprimer** de la table de routage.
4. **Ajouter** le plus rapidement possible les **nouvelles** routes ou **remplacer** le plus rapidement les routes **perdues** par **la meilleure route actuellement disponible**.

### 3.2 Fonctionnement des protocoles de routage

- Un protocole de routage transporte des informations sur les différentes routes dans l'inter réseau mais aussi **des messages de maintien de relations de voisinage**.
- Chaque routeur reçoit et envoie des informations de routage à ses voisins et **qu'à ses voisins!!!**  
Mais qu'est-ce qu'un voisin !
  - Pour OSPF et autre c'est le routeur d'à côté physiquement !
  - Pour BGP ce n'est pas vrai !
- Il applique un algorithme, par exemple Dijkstra, qui optimise ces informations en **chemins** cohérents.

**Un chemin est un ensemble de routes !**

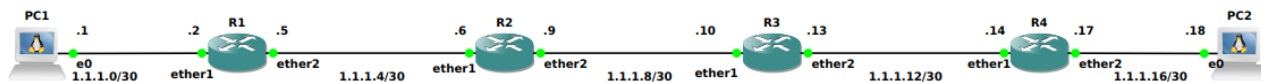
### 3.3 Routage statique VS routage dynamique

- Avantages du routage statique :
  - Il peut servir de mécanisme de **backup**.
  - Il est "facile" à configurer.
  - Aucune ressource supplémentaire nécessaire. (Ne consomme pas de CPU !)
  - Il est plus sécurisé.
- Désavantages du routage statique :
  - Chaque changement dans la topologie nécessite une intervention manuelle sur les routeurs de la topologie.

- Cette méthode ne convient pas pour des réseaux d'opérateurs qui utilisent obligatoirement le routage dynamique OSPF ou IS-IS.

#### Exemple de routage statique :

Voir le projet : **ping-et-routage-statique**



Voici les configurations, **la partie routage**, des routeurs R1 à R4

```
/ip route
add distance=1 gateway=1.1.1.6
add distance=1 dst-address=1.1.1.8/30 gateway=1.1.1.6
add distance=1 dst-address=1.1.1.12/30 gateway=1.1.1.6
add distance=1 dst-address=1.1.1.16/30 gateway=1.1.1.6
add distance=1 dst-address=22.22.22.22/32 gateway=1.1.1.6
add distance=1 dst-address=33.33.33.33/32 gateway=1.1.1.6
add distance=1 dst-address=44.44.44.44/32 gateway=1.1.1.6

/ip route
add distance=1 dst-address=1.1.1.0/30 gateway=1.1.1.9
add distance=1 dst-address=1.1.1.4/30 gateway=1.1.1.9
add distance=1 dst-address=1.1.1.16/30 gateway=1.1.1.14
add distance=1 dst-address=11.11.11.11/32 gateway=1.1.1.9
add distance=1 dst-address=22.22.22.22/32 gateway=1.1.1.9
add distance=1 dst-address=44.44.44.44/32 gateway=1.1.1.14

/ip route
add distance=1 dst-address=1.1.1.0/30 gateway=1.1.1.13
add distance=1 dst-address=1.1.1.4/30 gateway=1.1.1.13
add distance=1 dst-address=1.1.1.8/30 gateway=1.1.1.13
add distance=1 dst-address=11.11.11.11/32 gateway=1.1.1.13
add distance=1 dst-address=22.22.22.22/32 gateway=1.1.1.13
add distance=1 dst-address=33.33.33.33/32 gateway=1.1.1.13

/ip route
add distance=1 dst-address=1.1.1.0/30 gateway=1.1.1.5
add distance=1 dst-address=1.1.1.12/30 gateway=1.1.1.10
add distance=1 dst-address=1.1.1.16/30 gateway=1.1.1.10
add distance=1 dst-address=11.11.11.11/32 gateway=1.1.1.5
add distance=1 dst-address=33.33.33.33/32 gateway=1.1.1.10
add distance=1 dst-address=44.44.44.44/32 gateway=1.1.1.10
```

#### Que peut-on en conclure ?

1. Que c'est fastidieux ! ça l'est tellement que j'ai la fainéantise d'ajouter un 5ième routeur appelé raccourci, vous vous souvenez ?
2. Que pour R1 et R4 on pouvait ne mettre qu'une seule ligne **/ip route add distance=1 gateway=1.1.1.6** (pour R1) ou **1.1.1.13** (pour R4)  
Mais voilà un routeur avec une seule route c'est comme un rond point avec une seule route, sauf que sur un rond point on peut y voir des lapins et de jolis fleurs ☺
3. Que si on ajoute un routeur et bien il faut faire des modifications sur chaque oui oui chaque routeur pour leur indiquer les nouvelles routes de ce nouveau routeur ! Nous pourrions le faire en TP mais ce serait du temps perdu ?
4. Que cette structure est au réseau ce que Print ("Hello World") est au développement ☺. Elle n'existe pas dans le monde réel !!!  
En effet un routeur de FAI ne possède jamais qu'une seule route !
5. Qu'en utilisant un protocole de routage dynamique ça fonctionne sans problème, si la personne qui le configure connaît le principe !!!
6. Bref le routage statique c'est pour les machines en bout de ligne seulement, les CE par exemple.

### 3.4 Protocoles de routage à vecteur de distance

- Un protocole de routage à vecteur de distance utilise un **algorithme de routage additionnant les distances** pour trouver les meilleures routes (**Bellman-Ford**).
- Les routeurs envoient la totalité de leur table de routage aux voisins.
- Ils sont sensibles aux boucles de routage.
- Avec ce type de protocole, aucun routeur ne remplit de fonction particulière. On parle de connaissance "plate" de l'inter réseau ou de routage non-hiérarchique.
- Ils convergent lentement.
- **RIP** et **EIGRP** (Cisco) sont des protocoles à vecteur de distance
- Ils ne sont pas utilisés dans les backbones car MPLS a besoin d'un protocole de routage à état de liens

### 3.5 Protocoles de routage à état de liens

- Un protocole de routage à état de liens utilise un algorithme plus efficace (**Dijkstra** ou **Shortest Path First**) mais demande plus de ressources CPU/RAM.



- Les routeurs collectent l'ensemble des coûts des liens d'un inter réseau et construisent de leur point de vue l'arbre de tous les chemins possibles. Les meilleures routes sont alors intégrées à la table de routage.
- On parle de routage hiérarchique.
- Ils convergent très rapidement.
- OSPF et IS-IS sont des protocoles de routage à état de liens.
- **Les routeurs entretiennent des relations de voisinage maintenues, ils n'arrêtent pas de dialoguer !**

### 3.6 Quelques définitions

**Convergence** : c'est le temps nécessaire pour qu'un ensemble de routeurs puissent disposer d'une vision homogène, complète et efficace de l'ensemble des routes d'un inter réseau.

Le temps de convergence est particulièrement éprouvé lorsqu'il y a des modifications topologiques dans l'inter réseau.

**Métrique** : La métrique d'une route est la valeur d'une route en comparaison à d'autres routes apprises par le même protocole de routage.

Plus sa valeur est faible, meilleure est la route.

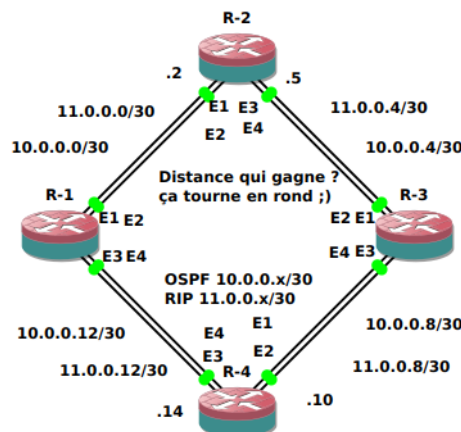
Chaque protocole dispose de sa méthode de valorisation.

On peut trouver toute une série de composantes de métrique parmi lesquelles :

- le nombre de sauts **RIP**
- la bande passante **EIGRP**
- le délai **EIGRP**
- la charge **EIGRP**
- la fiabilité **EIGRP**
- le coût **OSPF** et IS-IS

Soit la structure suivante qui n'a rien de "professionnelle" mais juste pour voir l'intérêt de la métrique.

La métrique



Les routeurs R1 à R4 ont chacun une interface de loopback avec les @IP 1.1.1.1 à 4.4.4.4

Imaginons que l'on fasse tomber les 2 interfaces E3 et E4 de R4.

Par quelle route le Routeur R1 passera pour atteindre l'interface de loopback de R4 à savoir 4.4.4.4 ?

Réponse : **tool traceroute 4.4.4.4** renvoie 10.0.0.2, 10.0.0.6 et 4.4.4.4, il a donc utilisé le chemin des routes en OSPF car la distance administrative d'OSPF est de 110 et celle de RIP est de 120...

Si on active les 2 interfaces E3 et E4 de R4, par où va-t'il passer si on fait un **tool traceroute 3.3.3.3** ?

La métrique étant la même, par R2 ou par R4 ?

Réponse : par R2 car les @IP sont plus petites, il fallait bien choisir, pour une fois que les petits gagnent

☺

**Distance administrative d'une route** : indique la préférence dans une table de routage pour des destinations apprises par un protocole de routage par rapport aux mêmes destinations apprises par un autre protocole de routage. Cette valeur est codée sur 8 bits et va de 0 à 255

Plus la valeur est petite et plus le protocole est préféré.





Par exemple, une route EIGRP sera préférée à une route RIP ;

**Une route statique sera préférée à toute autre route dynamique.**

Le tableau ci dessous donne des **distances administratives**

Méthode de routage	Distance administrative
<b>Réseau connecté</b>	<b>0</b>
<b>Route statique</b>	<b>1</b>
Ext-BGP	20
Int-EIGRP	90
<b>OSPF</b>	<b>110</b>
IS-IS	115
RIP	120
Int-BGP	200
Inconnu	255

### 3.7 Dans un réseau d'opérateur

Dans un réseau d'opérateur on utilise le routage dynamique !

Il existe plusieurs protocoles de routage dynamique qui ont chacun leur intérêt.

En TP nous en verrons qu'un seul !

Pourquoi ? Parce que pas le temps et que le but est de comprendre les principes et l'intérêt du routage dynamique !

Ce sera **OSPF**, pourquoi ? Parce que c'est celui qui est le plus utilisé.

Pas le meilleur, le meilleur c'est... aucune idée...

Il existe aussi RIPv2 qui est généralement utilisé entre le CE (**Customer Edge**) et le PE (**Provider Edge**)

### 3.8 Les réseaux d'opérateurs

Nous avons vu que les opérateurs possèdent chacun un Backbone.

Si on relie ces Backbones entre eux cela forme Internet !

Le protocole de routage utilisé par les opérateurs pour s'échanger les routes d'un backbone à l'autre se nomme BGP

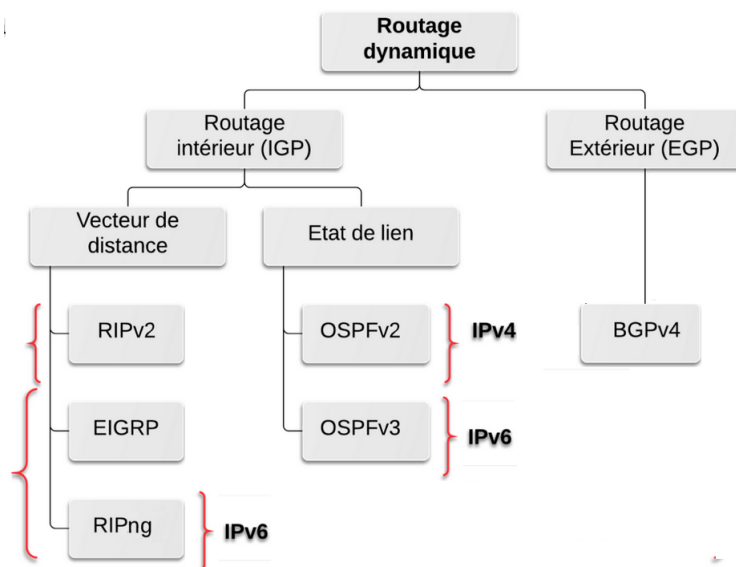
Nous verrons ce protocole plus en détails lors du cours sur BGP !

### 3.9 Synthèse des IGP

Vecteur de distance	Etat de lien
Algorithme Bellman-Ford (RIP)	Algorithme Dijkstra (OSPF)
Facile à configurer	Compétences requises
Partage des tables de routage	Partage des liaisons
Réseaux plats	Réseaux organisés en aires
Convergence plus lente	Convergence rapide, répartition de charge
Topologies limitées	Topologies complexes et larges
Gourmand en bande passante	Relativement discret
Peu consommateur en RAM et CPU	Grand consommateur en RAM et CPU
Mises à jour régulière en Broadcast/Multicast	Mises à jour immédiate
Pas de signalisation	Signalisation fiable et en mode connecté (BGP)
RIP v1, v2 et NG (IPv6)	OSPFv2/v3
EIGRP	ISIS



### 3.10 Synthèse de protocoles de routage



## 4 Un peu de tout ☺

### 4.1 L'interface de Loopback

Juste un petit rappel !

Cette interface est virtuelle, elle ne permet pas aux routeurs de "router" des données, non non !!!

**Elle sert globalement d'identifiant pour un routeur.**

Elle a comme toute interface une @IP mais en /32

Si cette interface ne répond pas c'est que le routeur est très mal en point...

Soit éteint soit en panne d'un point de vue kernel !!!

Sur les routeurs de type Mikrotik, ceux que l'on va utiliser en TP, elle est créée à l'aide de la commande :

**/interface bridge add name=loopback0**

On peut en créer plusieurs, il suffit de changer le nom.

### 4.2 Système autonome (AS)

- Un système autonome (AS) est un ensemble de réseaux sous la même **autorité administrative** dans le sens administratif !.
- Au sein d'un système autonome, les routes sont générées par des protocoles de routage intérieurs les **IGP** comme **RIP**, **EIGRP**, **OSPF** ou **ISIS**.
- **Le protocole** (il n'y en a qu'un !) de routage qui permet de connecter les AS entre eux est le protocole de routage extérieur **EGP** comme **BGP**.
- Dans le contexte de l'interconnexion mondiale des réseaux, l'**IANA** (par délégation aux organismes régionaux) attribue les numéros de système autonome (16/32 bits).

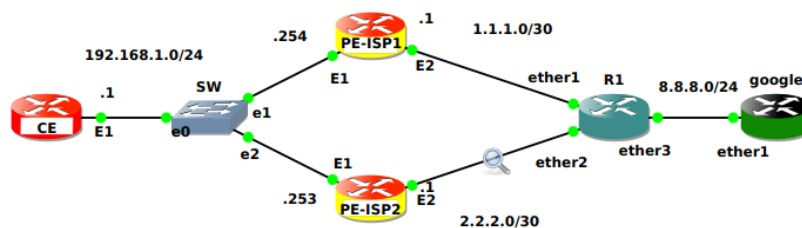
Nous verrons cela plus en détails dans le cours sur BGP et au fait on l'a vu au cours précédent sur Internet !

### 4.3 Plusieurs routes par défaut ?

**Remarque :** La distance administrative est définie administrativement et permet de positionner plusieurs routes statiques avec des distances administratives de valeurs différentes pour la même route !!!

Soit la structure suivante :





Une entreprise a accès à Internet en passant par un ISP.

Certaines grosses entreprises ont les moyens de prendre 2 ISP.

1. Un ISP principal, ici représenté par le routeur PE-ISP1
2. Un ISP secondaire, ici représenté par le routeur PE-ISP2

La problématique est la suivante, comment faire pour que la route par défaut initiale qui est de passer par PE-ISP1 change si PE-ISP1 tombe en panne ?

La solution tient en cette configuration

```

/ip route
add check-gateway=ping distance=1 gateway=192.168.1.254
add distance=2 gateway=192.168.1.253

```

On voit qu'il a 2 passerelle (gateway) une avec la distance administrative de 1 donc route par défaut et une avec la distance administrative de 2.

Expliquons ce que cela engendre en lançant la commande **/ip route print** sur le CE

```
[admin@CE] > /ip route print
```

#	DST-ADDRESS	PREF-SRC	GATEWAY	DISTANCE
0 A S	0.0.0.0/0		192.168.1.254	1
1 S	0.0.0.0/0		192.168.1.253	2
2 ADC	192.168.1.0/24	192.168.1.1	ether1	0

```
[admin@CE] >
```

On voit qu'il a 2 routes par **défaut 0.0.0.0/0** **mais qu'une seule est active**.

Ici celle passant par la passerelle 192.168.1.254.

Si le routeur PE-ISP1 tombe en panne, le CE le verra de suite, car il fait un ping 192.168.1.254 en permanence. Une non réponse entraîne la décision de ne plus activer cette première route par défaut mais la seconde, soit 192.168.1.253 donc le routeur PE-ISP2

En lançant à nouveau la commande **/ip route print** sur le CE

```
[admin@CE] > /ip route print
```

#	DST-ADDRESS	PREF-SRC	GATEWAY	DISTANCE
0 S	0.0.0.0/0		192.168.1.254	1
1 A S	0.0.0.0/0		192.168.1.253	2
2 ADC	192.168.1.0/24	192.168.1.1	ether1	0

```
[admin@CE] >
```

La route par **défaut 0.0.0.0/0 active** est celle par 192.168.1.253

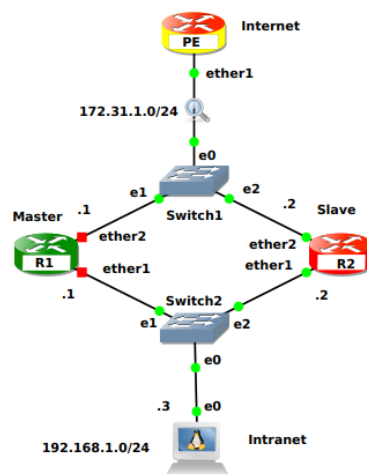
Bien sur si le routeur PE-ISP1 redevient actif il redevient la passerelle active !

**Conclusion** : À un temps **t** il n'y a qu'une **seule route défaut 0.0.0.0/0 active** !

## 4.4 VRRP

On vient de voir qu'un CE pouvait avoir 2 routes par défaut de manières que si un ISP a une panne l'autre ISP prenne le relai.

Il y a une autre solution possible, mettre 2 routeurs du même ISP et utiliser le protocole VRRP comme le montre la figure suivante : Le design VRRP



Comment cela fonctionne ?

On utilise le protocole de redondance de routeur virtuel VRRP ( **Virtual Router Redundancy Protocol** ).

C'est un protocole standard dont le but est d'augmenter **la disponibilité de la passerelle par défaut** des hôtes d'un même réseau.

**Le principe est de définir la passerelle par défaut** pour les hôtes du réseau comme étant **une adresse IP virtuelle référençant un groupe de routeurs**.

Oui un groupe, car on peut mettre plus que 2 routeurs !

Les 2 routeurs ont coté client donc intranet pour @IP 192.168.1.1 et .2 /24 sur l'interface ether1.

On leur ajoute à tous les 2 une interface virtuelle associée à cette interface ether1. Cette interface virtuelle aura pour @IP 192.168.1.254 et sera la passerelle du client intranet.

Voici la conf des 2 routeurs R1 et R2

```

/interface vrrp
add interface=ether1 name=vrrp1 priority=254 vrid=49
/interface wireless security-profiles
set [ find default=yes ] supplicant-identity=MikroTik
/ip address
add address=192.168.1.1/24 interface=ether1 network=192.168.1.0
add address=172.31.1.1/24 interface=ether2 network=172.31.1.0
add address=192.168.1.254 interface=vrrp1 network=192.168.1.254
/system identity
set name=R1

/interface vrrp
add interface=ether1 name=vrrp1 vrid=49
/interface wireless security-profiles
set [ find default=yes ] supplicant-identity=MikroTik
/ip address
add address=192.168.1.2/24 interface=ether1 network=192.168.1.0
add address=172.31.1.2/24 interface=ether2 network=172.31.1.0
add address=192.168.1.254 interface=vrrp1 network=192.168.1.254
/system identity
set name=R2

```

La ligne `/interface vrrp add interface=ether1 name=vrrp1 vrid=49` permet de "créer" une interface virtuelle de type VRRP un peut comme les interfaces virtuelles de type VLAN ou loopback.

On voit un seul paramètre qui change dans la "création" de l'interface virtuelle nommée vrrp1 c'est la **priority** qui vaut **254** pour R1 et par défaut elle vaut 100. **Plus elle est élevée plus elle est prioritaire.**

Les 2 routeurs s'informent mutuellement de leur état via le protocole VRRP en utilisant une **@IP de type multicast 224.0.0.18**

Comme le montre la capture wireshark suivante.

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.1.1	224.0.0.18	VRRP	46	Announcement (v3)
2	1.002675	192.168.1.1	224.0.0.18	VRRP	46	Announcement (v3)
3	2.004730	192.168.1.1	224.0.0.18	VRRP	46	Announcement (v3)

```

Frame 1: 46 bytes on wire (368 bits), 46 bytes captured (368 bits) on interface -, id 0
Ethernet II, Src: IETF-VRRP-VRID_31 (00:00:5e:00:01:31), Dst: IPv4mcast_12 (01:00:5e:00:00:12)
Internet Protocol Version 4, Src: 192.168.1.1, Dst: 224.0.0.18
Virtual Router Redundancy Protocol
  Version 3, Packet type 1 (Advertisement)
    Virtual Rtr ID: 49
    Priority: 254 (Non-default backup priority)
    Addr Count: 1
    0000 .... = Reserved: 0
    .... 0000 0110 0100 = Adver Int: 100
    Checksum: 0x6b89 [correct]
    [Checksum Status: Good]
    IP Address: 192.168.1.254

```



**Remarques :**

- Cisco possède son propre protocole de redondance nommé HSRP **Hot Standby Router Protocol**
- Une alternative open source existe aussi : CARP **Common Address Redundancy Protocol**
- Le principe n'existe pas que pour les routeurs. Il existe aussi pour les serveurs Web par exemple. Si vous voulez en lire d'avantage je vous conseille cet article très récent : <https://geekflare.com/fr/open-source-load-balancer/>

## 4.5 Répartition de charge

Ce que l'on appelle communément "**Load Balancing**" est la capacité pour un routeur de supporter plusieurs chemins à **coût égaux** vers une destination.

Il en résulte que les paquets vers une même destination sont répartis sur plusieurs interfaces.

Dans l'exemple précédent, le routeur R2 est complètement inactif pendant l'état de sauvegarde. C'est un gaspillage de ressources précieuses.

Il serait bien que le routeur R2 soit défini comme passerelle pour certains clients.

L'avantage évident de cette configuration est la mise en place **d'un schéma de partage de charge**.

Mais ce faisant, le routeur R2 n'est pas protégé par la configuration VRRP actuelle.

Pour que la configuration précédente fonctionne, **il faut 2 routeurs virtuels**.

Une partie du parc informatique aura comme passerelle 192.168.1.254 et l'autre partie aura la passerelle 192.168.1.253

Si un des 2 routeurs tombe en panne et bien momentanément l'autre sera la passerelle de tous.

Voici la conf des 2 routeurs

Pour R1

```
/ip address add address=192.168.1.1/24 interface=ether1
/interface vrrp add interface=ether1 vrid=49 priority=254
/interface vrrp add interface=ether1 vrid=77
/ip address add address=192.168.1.253/32 interface=vrrp1
/ip address add address=192.168.1.254/32 interface=vrrp2
```

Pour R2

```
/ip address add address=192.168.1.2/24 interface=ether1
/interface vrrp add interface=ether1 vrid=49
/interface vrrp add interface=ether1 vrid=77 priority=254
/ip address add address=192.168.1.253/32 interface=vrrp1
/ip address add address=192.168.1.254/32 interface=vrrp2
```

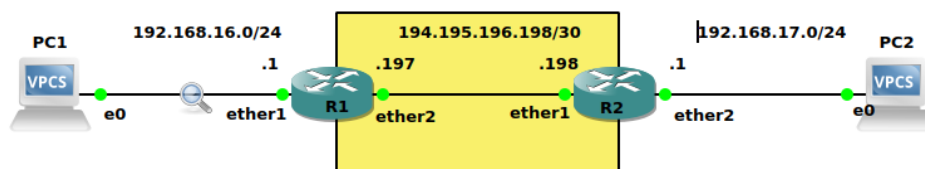
## 4.6 Les préfixes bogons

Nous avons vu qu'ils existaient des adresses privées et des adresses publiques et aussi d'autres adresses spécifiques telles les adresses **multicast**, les adresses APIPA etc...

Toutes ces adresses ne doivent pas circuler sur Internet !

Un préfixe bogon ne doit pas être visible dans la table de routage générale, car il est **destiné à un usage interne** (RFC 1918), aux réseaux de test (RFC 4737), à la **multidiffusion** et à d'autres usages toujours en interne. Ces préfixes ne doivent donc jamais être acceptés ou annoncés dans la zone DFZ (**Default-Free Zone**).

Soit la structure simple suivante. (Voir le design **bogons**)



Normalement, j'écris bien normalement, les 2 routeurs R1 et R2 doivent faire du NAT si on veut que PC1 et PC2 puissent aller sur Internet (Partie en jaune).

Imaginons que l'administrateur aie oublié de le faire et bien c'est "dangereux" !!!

Rien n'empêche nativement un routeur de router les @IP privées ou autres adresses spéciales.

Une @IP c'est une @IP !

Pour palier à cela, on filtre les adresses avec la configuration suivante

```
/ip firewall address-list
add list="BOGONS" address=0.0.0.0/8
add list="BOGONS" address=10.0.0.0/8
```



```
add list="BOGONS" address=100.64.0.0/10
add list="BOGONS" address=127.0.0.0/8
add list="BOGONS" address=169.254.0.0/16
add list="BOGONS" address=172.16.0.0/12
add list="BOGONS" address=192.0.0.0/24
add list="BOGONS" address=192.0.2.0/24
add list="BOGONS" address=192.168.0.0/16
add list="BOGONS" address=198.18.0.0/15
add list="BOGONS" address=198.51.100.0/24
add list="BOGONS" address=203.0.113.0/24
add list="BOGONS" address=224.0.0.0/3
```

```
/ip firewall filter
add action=drop chain=forward comment="Block Bogon IP Addresses"
in-interface=NOM-Interface src-address-list=BOGONS
```

Si on le met que sur R1 en remplaçant NOM-Interface par ether1 et que PC2 l'**echo reply** du ping PC1 le ping ira jusqu'à PC1 mais l'**echo request** au retour lui ne pourra pas traverser R1 !

Il faut mettre la même conf sur R2 !!!

**Toutes les interfaces externes des routeurs sur Internet doivent bloquer ces préfixes appelés Bogons**

## 4.7 Sous Linux

Quand on travaille sur un serveur ou un ordinateur quelconque, ce dernier peut posséder plusieurs interfaces.

Imaginons le scénario suivant :

- Vous avez un ordinateur avec 2 interfaces filaires Eth0 et Eth1, plus une interface Wifi plus une interface 4G, pourquoi pas.
- Vous faites un ping vers une IP très célèbre telle 8.8.8.8, que se passe-t-il ?
- Oui ça répond, mais par où est passée votre packet ICMP Echo Request ?
- Non non ce n'est pas au pif, c'est toujours pareil

Pour le savoir, il suffit de lancer la commande **ip route ls** qui doit vous renvoyer quelque chose comme :

```
$ ip route ls
default via 10.38.16.1 dev eth0
10.38.16.0/22 dev eth0 proto kernel scope link src 10.38.19.111
169.254.0.0/16 dev eth0 scope link metric 1000
172.17.0.0/16 dev docker0 proto kernel scope link src 172.17.0.1 linkdown
192.168.56.0/24 dev vboxnet0 proto kernel scope link src 192.168.56.1 linkdown
fougeray@C304L-159C00:~$
```

\$ □

À la FAC pas chez vous !

Expliquons ce résultat !

Ou faisons un schéma ?

Combien d'interfaces ?

Je crois que tout cela on l'a déjà fait et compris dans la partir LAN ☺

## 4.8 Qu'est-ce que le réseau 169.254.0.0/16 ?

C'est l'**APIPA (Automatic Private Internet Protocol Addressing)** ou **IPv4LL**.

Un processus qui permet à un OS de s'attribuer automatiquement une adresse IP, lorsque le serveur DHCP est hors service ou injoignable.

APIPA utilise la plage d'adresses IP 169.254.0.0/16, c'est-à-dire la plage dont les adresses vont de 169.254.0.0 à 169.254.255.255.

Cette plage est réservée à cet usage auprès de l'IANA.

Et dire que l'on manque d'@IPv4 ... Bref

## 5 OSPF

### OSPF est un protocole de routage dynamique à état de liens !

OSPF est l'acronyme de **Open Shortest Path First**

Je ne vais pas vous faire un cours complet sur OSPF non juste comprendre les principes !

Si vous voulez devenir expert en OSPF vous pouvez mais pas en 1h...

[https://fr.wikipedia.org/wiki/Open\\_Shortest\\_Path\\_First](https://fr.wikipedia.org/wiki/Open_Shortest_Path_First)

Ou chez Cisco

[https://www.cisco.com/c/fr\\_ca/support/docs/ip/open-shortest-path-first-ospf/7039-1.html](https://www.cisco.com/c/fr_ca/support/docs/ip/open-shortest-path-first-ospf/7039-1.html)

### 5.1 Fonctionnement général

- Dans OSPF, chaque routeur établit des **relations d'adjacence** avec ses voisins immédiats en envoyant des **messages hello à intervalles réguliers**.
- Chaque routeur communique ensuite la liste des réseaux auxquels il est connecté et qui sont dans sa configuration OSPF par des messages **Link-state advertisements (LSA)** propagés de proche en proche à tous les routeurs du réseau. L'ensemble des LSA forme une **BDD de l'état des liens Link-State Database (LSDB)** pour chaque aire, qui est identique pour tous les routeurs participants dans cette aire. Chaque routeur utilise ensuite l'algorithme de Dijkstra, Shortest Path First (SPF) pour déterminer la route la plus rapide vers chacun des réseaux connus dans la LSDB.
- Le bon fonctionnement d'OSPF requiert donc une complète cohérence dans le calcul **SPF**, il n'est donc par exemple pas possible de **filtrer des routes** ou de les **résumer à l'intérieur d'une aire**.
- **En cas de changement de topologie, de nouveaux LSA sont propagés de proche en proche, et l'algorithme SPF est exécuté à nouveau sur chaque routeur.**
- Cela va très vite!!!!

### 5.2 Les aires

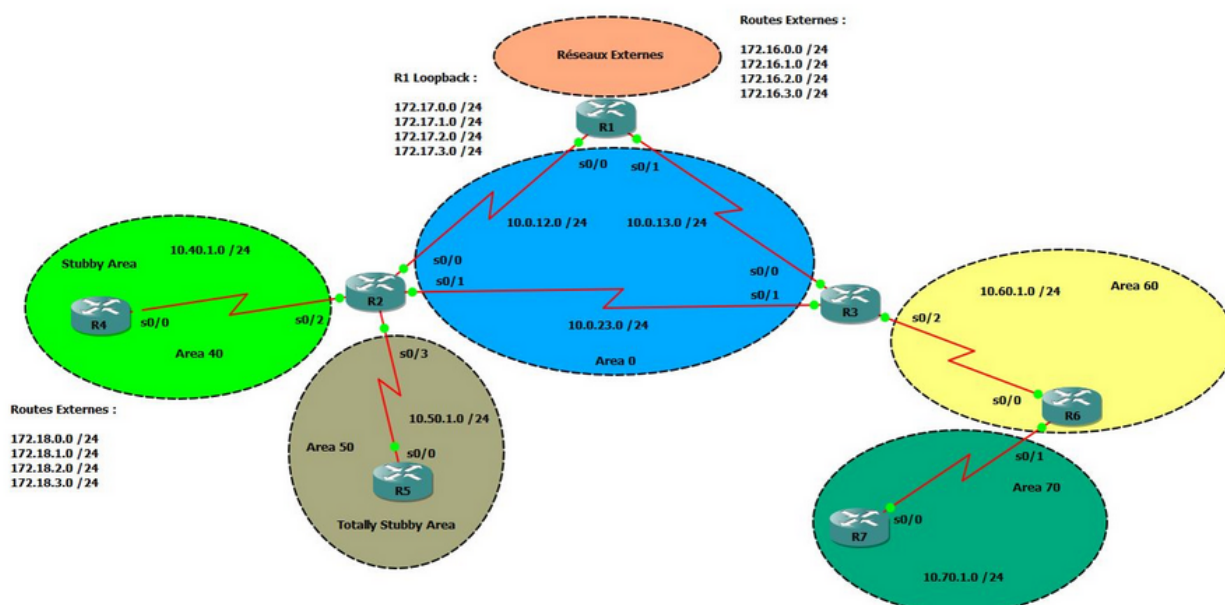
Afin de simplifier les calculs etc..., il est possible même nécessaire de partager ce backbone en aires.

Comme je n'ai pas pour but de faire de vous des experts en OSPF, mais des étudiants qui ont compris le principe, je ne vais pas développer.

**Sachez** seulement

- qu'il existe toujours une **aire principale** appelée **aire 0** et nous verrons cela en TP.
- Des **stubs** (cul de sac) donc des aires qui n'ont qu'une entrée/sortie et donc tous les routeurs **ont tous la même route par défaut** !

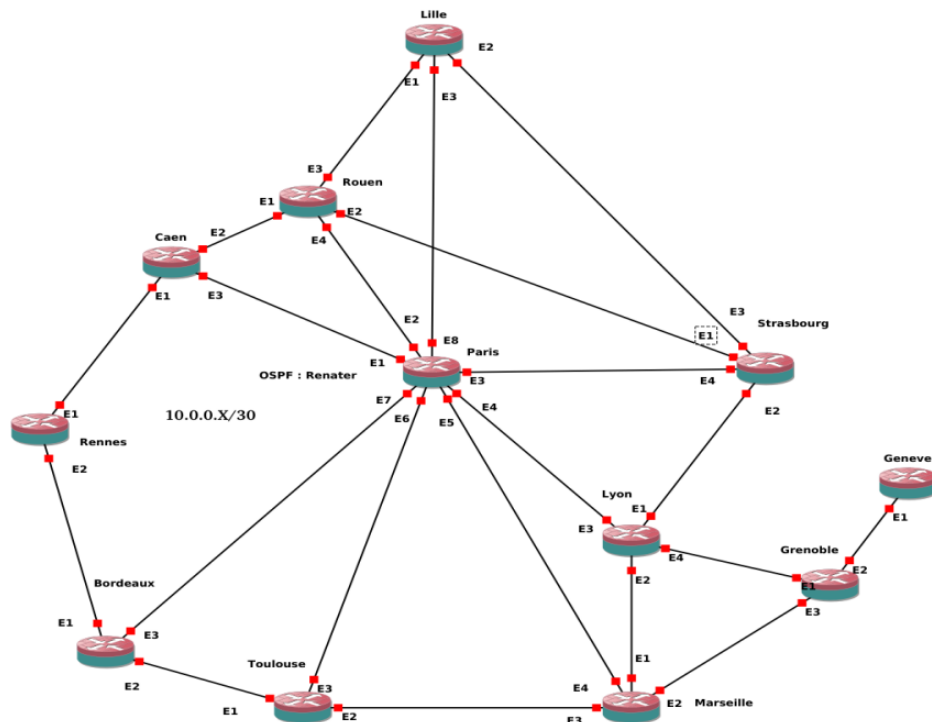
Exemple sur une image prise ici : <https://www.networklab.fr/les-differents-types-de-zones-ospf/>



### 5.3 OSPF en pratique

En TP nous travaillerons sur la structure suivante.

Elle représente le réseau d'un opérateur tel RENATER, vous savez le FAI qui vous permet d'aller sur Internet de la FAC ☺



Cette structure ou disons ce backbone contient 12 routeurs placés dans des grandes villes dont Caen...

Vu de l'extérieur c'est un seul réseau, vu de l'intérieur c'est 33 réseaux !

Pourquoi 33 réseaux ?

Et bien si on compte toutes les liaisons entre chaque routeur cela en fait 21 et il y a aussi le réseau de loopbacks de chaque routeur :  $21 + 12 = 33$ ...

#### 5.3.1 La configuration

Il n'y a rien de bien compliqué. Rien n'est compliqué quand on sait ☺

1. On déclare une instance OSPF ici la ligne :

***/routing ospf instance set 0 router-id=1.1.1.1***

**Remarque :** une instance peut être vue comme un processus.

2. On déclare une zone

***/routing ospf network add area=backbone***

**Remarque :** une zone est un ensemble de routeurs qui s'échangent leurs routes.

3. On déclare les réseaux que l'on désire propager à ses voisins ici pour Caen les lignes

***/routing ospf network***

***add area=backbone network=1.1.1.1/32***

***add area=backbone network=10.0.0.0/30***

***add area=backbone network=10.0.0.4/30***

***add area=backbone network=10.0.0.8/30***

Vous remarquerez qu'ils sont tous dans la même zone.

ici **Caen** et **Rouen** !

- 1 Caen
- 2 /interface bridge add name=l0
- 3 /system identity set name=Caen
- 4 /ip address



```
5 add address=1.1.1.1 interface=l0 network=1.1.1.1
6 add address=10.0.0.1/30 interface=ether1 network=10.0.0.0
7 add address=10.0.0.5/30 interface=ether2 network=10.0.0.4
8 add address=10.0.0.9/30 interface=ether3 network=10.0.0.8
9 /routing ospf instance set 0 router-id=1.1.1.1
10 /routing ospf network
11 add area=backbone network=1.1.1.1/32
12 add area=backbone network=10.0.0.0/30
13 add area=backbone network=10.0.0.4/30
14 add area=backbone network=10.0.0.8/30
15
16 Rouen
17 /interface bridge add name=l0
18 /system identity set name=Rouen
19 /ip address
20 add address=2.2.2.2 interface=l0 network=2.2.2.2
21 add address=10.0.0.2/30 interface=ether1 network=10.0.0.0
22 add address=10.0.0.17/30 interface=ether2 network=10.0.0.16
23 add address=10.0.0.21/30 interface=ether3 network=10.0.0.20
24 add address=10.0.0.13/30 interface=ether4 network=10.0.0.12
25 /routing ospf instance set 0 router-id=2.2.2.2
26 /routing ospf network
27 add area=backbone network=2.2.2.2/32
28 add area=backbone network=10.0.0.0/30
29 add area=backbone network=10.0.0.12/30
30 add area=backbone network=10.0.0.16/30
31 add area=backbone network=10.0.0.20/30
```

Si on doit ajouter un routeur relié à Rouen et Caen, par exemple Le Havre avec les réseaux

- 10.0.0.100/30 entre Caen et Le Havre
- 10.0.0.104/30 entre Rouen et Le Havre

Il suffit d'ajouter une ligne

- sur Caen : ***/routing ospf network add area=backbone network=10.0.0.100/30***
- sur Rouen : ***/routing ospf network add area=backbone network=10.0.0.104/30***

Et tous les autres routeurs seront avertis que pour aller au Havre il faut passer par Caen pour les bretons et par Rouen pour les Ch'tis

Voilà l'avantage du routage dynamique par rapport au routage statique vu précédemment !

### 5.3.2 Le protocole OSPF

OSPF est un protocole de routage, donc ça parle ?

Voyons ce qui se passe quand on allume seulement 3 routeurs (ça marche aussi avec plus mais ça parle trop ☺) **Caen**, **Rouen** et **Lille** puis on place une sonde wireshark entre Caen et Rouen.

Analysons rapidement tout cela !

C'est ce que vous ferez en TP !



No.	Time	Source	Destination	Protocol	Leng	Info
23.367285	10.0.0.2	224.0.0.5	OSPF	82	82	Hello Packet
30.805090	10.0.0.1	224.0.0.5	OSPF	82	82	Hello Packet
33.540729	10.0.0.2	224.0.0.5	OSPF	82	82	Hello Packet
40.760664	10.0.0.1	10.0.0.2	OSPF	66	66	DB Description
40.808723	10.0.0.1	224.0.0.5	OSPF	82	82	Hello Packet
43.538402	10.0.0.2	224.0.0.5	OSPF	82	82	Hello Packet
45.756751	10.0.0.1	10.0.0.2	OSPF	66	66	DB Description
50.760368	10.0.0.1	10.0.0.2	OSPF	66	66	DB Description
50.806559	10.0.0.1	224.0.0.5	OSPF	82	82	Hello Packet
53.322768	10.0.0.2	10.0.0.1	OSPF	66	66	DB Description
53.359267	10.0.0.1	10.0.0.2	OSPF	86	86	DB Description
53.361308	10.0.0.2	10.0.0.1	OSPF	70	70	LS Request
53.362041	10.0.0.2	10.0.0.1	OSPF	86	86	DB Description
53.384060	10.0.0.1	10.0.0.2	OSPF	134	134	LS Update
53.385775	10.0.0.1	10.0.0.2	OSPF	70	70	LS Request
53.385895	10.0.0.1	10.0.0.2	OSPF	66	66	DB Description
53.388093	10.0.0.2	10.0.0.1	OSPF	66	66	DB Description
53.389354	10.0.0.2	10.0.0.1	OSPF	146	146	LS Update
53.404708	10.0.0.1	10.0.0.2	OSPF	66	66	DB Description
53.430388	10.0.0.1	224.0.0.6	OSPF	134	134	LS Update
53.432984	10.0.0.2	224.0.0.5	OSPF	178	178	LS Update
53.544295	10.0.0.2	224.0.0.5	OSPF	82	82	Hello Packet
54.387352	10.0.0.2	224.0.0.5	OSPF	78	78	LS Acknowledge
54.422747	10.0.0.1	224.0.0.5	OSPF	98	98	LS Acknowledge
60.810540	10.0.0.1	224.0.0.5	OSPF	82	82	Hello Packet
63.407499	10.0.0.2	10.0.0.1	OSPF	146	146	LS Update
63.431142	10.0.0.1	10.0.0.2	OSPF	134	134	LS Update
63.545320	10.0.0.2	224.0.0.5	OSPF	82	82	Hello Packet
64.410129	10.0.0.1	224.0.0.5	OSPF	78	78	LS Acknowledge
64.433608	10.0.0.2	224.0.0.5	OSPF	78	78	LS Acknowledge
70.814375	10.0.0.1	224.0.0.5	OSPF	82	82	Hello Packet
73.541264	10.0.0.2	224.0.0.5	OSPF	82	82	Hello Packet

Bordure !

```

Frame 26: 82 bytes on wire (656 bits), 82 bytes captured (656 bits) on interface 0
Ethernet II, Src: 0c:ca:a6:b0:41:00 (0c:ca:a6:b0:41:00), Dst: IPv4mcast_05 (01:00:5e:00:00:05)
Internet Protocol Version 4, Src: 10.0.0.2, Dst: 224.0.0.5
Open Shortest Path First
  OSPF Header
    Version: 2
    Message Type: Hello Packet (1)
    Packet Length: 48
    Source OSPF Router: 2.2.2.2
    Area ID: 0.0.0.0 (Backbone)
    Checksum: 0xf598 [correct]
    Auth Type: Null (0)
    Auth Data (none): 0000000000000000
  OSPF Hello Packet

```

```

0000  01 00 5e 00 00 05 0c ca a6 b0 41 00 08 00 45 c0
0010  00 44 72 57 00 00 01 59 5c 43 0a 00 00 02 e0 00
0020  00 05 02 01 00 30 02 02 02 02 00 00 00 00 f5 98
0030  00 00 00 00 00 00 00 00 00 00 ff ff ff fc 00 0a
0040  02 01 00 00 00 28 00 00 00 00 00 00 00 01 01
0050  01 01

```

```

--A-----A--E-
-Drw---Y \C-----
-...-...-...-...-
-...-...-...-...-
-...-...-...-...-

```

Bordure !

### 5.3.3 Adresses sources et destinations

On peut remarquer que les 2 routeurs utilisent les **adresses destination** !

- De leur port Ethernet respectif soient 10.0.0.1/24 et 2 car ils sont sur le réseau 10.0.0.0/30 !  
Pour tout ce qui est **DB Description** et **LS Update** (pas toujours !)
- De type **multicast 224.0.0.5** pour les paquets de type **Hello Packet** , **LS-Acknowledge** et **LS Update** (pas toujours !)

2 adresses **multicast** sont utilisées

- 224.0.0.5 et FF02 : :5 (Ipv6) pour tous les routeurs OSPF
- **224.0.0.6** et FF02 : :6 (Ipv6) pour tous les routeurs OSPF de **bordure** DR/BDR OSPF

C'est le cas du routeur de Caen quand on n'allume que 3 routeurs Caen, Rouen et Lille.

Si vous regardez bien le schéma, dans ce cas Caen est un routeur de Bordure et Lille aussi alors que Rouen non.

- Rouen utilise 224.0.0.5 et Caen 224.0.0.6 dans ce cas.
- Si on allume les bretons, le routeur Rennes alors Caen ne serait plus un routeur de Bordure !

### 5.3.4 Les messages

Il y a 5 types de messages OSPF !

1. Les **hello packet** : ça semble logique, les routeurs signalent leur présence à leurs voisins.

Il disent salut en n'utilisant pas une adresse IP de type Unicast, mais de type multicast (224.0.0.5 et .6), ça forme un bon groupe un ensemble de routeur qui parle le même protocole !



```

▶ Frame 394: 82 bytes on wire (656 bits), 82 bytes captured (656 bits) on interface 0
▶ Ethernet II, Src: 0c:ca:a6:51:cb:00 (0c:ca:a6:51:cb:00), Dst: IPv4mcast_05 (01:00:5e:00:00:05)
▶ Internet Protocol Version 4, Src: 10.0.0.1, Dst: 224.0.0.5
▼ Open Shortest Path First
  ▼ OSPF Header
    Version: 2
    Message Type: Hello Packet (1)
    Packet Length: 48
    Source OSPF Router: 1.1.1.1
    Area ID: 0.0.0.0 (Backbone)
    Checksum: 0xe195 [correct]
    Auth Type: Null (0)
    Auth Data (none): 0000000000000000
  ▼ OSPF Hello Packet
    Network Mask: 255.255.255.252
    Hello Interval [sec]: 10
    ▶ Options: 0x02, (E) External Routing
    Router Priority: 1
    Router Dead Interval [sec]: 40
    Designated Router: 10.0.0.2
    Backup Designated Router: 10.0.0.1
    Active Neighbor: 2.2.2.2

```

2. Les **DB description** : Décrit le contenu des bases de données d'état de liens (**link-state database**) des routeurs OSPF.

```

▶ Frame 32: 66 bytes on wire (528 bits), 66 bytes captured (528 bits) on interface 0
▶ Ethernet II, Src: 0c:ca:a6:51:cb:00 (0c:ca:a6:51:cb:00), Dst: 0c:ca:a6:b0:41:00 (0c:ca:a6:b0:41:00)
▶ Internet Protocol Version 4, Src: 10.0.0.1, Dst: 10.0.0.2
▼ Open Shortest Path First
  ▼ OSPF Header
    Version: 2
    Message Type: DB Description (2)
    Packet Length: 32
    Source OSPF Router: 1.1.1.1
    Area ID: 0.0.0.0 (Backbone)
    Checksum: 0xb3f8 [correct]
    Auth Type: Null (0)
    Auth Data (none): 0000000000000000
  ▼ OSPF DB Description
    Interface MTU: 1500
    ▶ Options: 0x42, 0, (E) External Routing
    ▼ DB Description: 0x07, (I) Init, (M) More, (MS) Master
      .... 0... = (R) OOBResync: Not set
      .... .1.. = (I) Init: Set
      .... ..1. = (M) More: Set
      .... ...1 = (MS) Master: Yes
    DD Sequence: 0

```

```

0000  0c ca a6 b0 41 00 0c ca a6 51 cb 00 08 00 45 c0  ....A...-Q...E-
0010  00 34 72 51 00 00 01 59 32 5e 0a 00 00 01 0a 00  -4rQ...Y 2^.....
0020  00 02 02 02 00 20 01 01 01 01 00 00 00 00 b3 f8  ....B...
0030  00 00 00 00 00 00 00 00 00 00 05 dc 42 07 00 00  ....
0040  00 00

```

3. Les **LS-Update** (LSU) : Transporte les **link-sate advertisements**, les LSA, aux routeurs voisins. En français donne les réseaux qu'il connaît!

```

▶ Frame 412: 226 bytes on wire (1808 bits), 226 bytes captured (1808 bits) on interface 0
▶ Ethernet II, Src: 0c:ca:a6:b0:41:00 (0c:ca:a6:b0:41:00), Dst: IPv4mcast_05 (01:00:5e:00:00:05)
▶ Internet Protocol Version 4, Src: 10.0.0.2, Dst: 224.0.0.5
▼ Open Shortest Path First
  ▼ OSPF Header
    Version: 2
    Message Type: LS Update (4)
    Packet Length: 192
    Source OSPF Router: 2.2.2.2
    Area ID: 0.0.0.0 (Backbone)
    Checksum: 0x44e2 [correct]
    Auth Type: Null (0)
    Auth Data (none): 0000000000000000
  ▼ LS Update Packet
    Number of LSAs: 3
    ▼ LSA-type 1 (Router-LSA), len 84
      .000 0000 0000 0001 = LS Age (seconds): 1
      0... .. Do Not Age Flag: 0
      ▶ Options: 0x02, (E) External Routing
      LS Type: Router-LSA (1)
      Link State ID: 2.2.2.2
      Advertising Router: 2.2.2.2
      Sequence Number: 0x80000004
      Checksum: 0xd373
      Length: 84
      ▶ Flags: 0x00
      Number of Links: 5
      ▶ Type: Stub ID: 10.0.0.12 Data: 255.255.255.252 Metric: 10
      ▶ Type: Stub ID: 10.0.0.16 Data: 255.255.255.252 Metric: 10
      ▶ Type: Stub ID: 10.0.0.2 Data: 10.0.0.2 Metric: 10
      ▶ Type: Stub ID: 2.2.2.2 Data: 255.255.255.252 Metric: 10
    ▼ LSA-type 2 (Network-LSA), len 32
      .000 0000 0000 0001 = LS Age (seconds): 1
      0... .. Do Not Age Flag: 0
      ▶ Options: 0x02, (E) External Routing
      LS Type: Network-LSA (2)
      Link State ID: 10.0.0.21
      Advertising Router: 2.2.2.2
      Sequence Number: 0x80000001
      Checksum: 0xb869
      Length: 32
      Netmask: 255.255.255.252
      Attached Router: 2.2.2.2
      Attached Router: 3.3.3.3
    ▼ LSA-type 1 (Router-LSA), len 48
      .000 0000 0000 0101 = LS Age (seconds): 5
      0... .. Do Not Age Flag: 0
      ▶ Options: 0x02, (E) External Routing
      LS Type: Router-LSA (1)
      Link State ID: 3.3.3.3
      Advertising Router: 3.3.3.3
      Sequence Number: 0x80000002
      Checksum: 0x11dd
      Length: 48
      ▶ Flags: 0x00
      Number of Links: 2
      ▶ Type: Stub ID: 10.0.0.20 Data: 255.255.255.252 Metric: 10

```

Stub ?  
Transit ?

Lille est dénoncé à  
Caen par Rouen ;)

4. Les **LS-Acknowledge** (LSAk) : Accusés de réception des LSA des voisins.



```

▶ Frame 53: 78 bytes on wire (624 bits), 78 bytes captured (624 bits) on interface 0
▶ Ethernet II, Src: 0c:ca:a6:b0:41:00 (0c:ca:a6:b0:41:00), Dst: IPv4mcast_05 (01:00:5e:00:00:05)
▶ Internet Protocol Version 4, Src: 10.0.0.2, Dst: 224.0.0.5
▼ Open Shortest Path First
  ▼ OSPF Header
    Version: 2
    Message Type: LS Acknowledge (5)
    Packet Length: 44
    Source OSPF Router: 2.2.2.2
    Area ID: 0.0.0.0 (Backbone)
    Checksum: 0xea03 [correct]
    Auth Type: Null (0)
    Auth Data (none): 0000000000000000
  ▼ LSA-type 1 (Router-LSA), len 72
    0000 0000 0011 0100 = LS Age (seconds): 52
    0... .. = Do Not Age Flag: 0
    ▶ Options: 0x02 (E) External Routing
    LS Type: Router-LSA (1)
    Link State ID: 1.1.1.1
    Advertising Router: 1.1.1.1
    Sequence Number: 0x80000002
    Checksum: 0x8943
    Length: 72

0000 01 00 5e 00 00 05 0c ca a6 b0 41 00 08 00 45 c0 ..A...A..E
0010 00 40 72 68 00 00 01 59 5c 36 0a 00 00 02 e0 00 ..@rh...Y \6...
0020 00 05 02 05 00 2c 02 02 02 02 00 00 00 00 ea 03 .....4...
0030 00 00 00 00 00 00 00 00 00 00 00 34 02 01 01 01 .....4...
0040 01 01 01 01 01 01 80 00 00 02 89 43 00 48 .....C.H

```

### 5.3.5 La table de routage avec 3 routeurs allumés !

On voit que les réseaux connus par Lille, par exemple 10.0.0.28 (la ligne 10) entre Lille et Strasbourg est connu de Caen.

Lille l'a dit à Rouen qui l'a dit à Caen ...

Pour preuve, voyons la table de routage de Caen : **ip route print**...(tiens c'était **ip route ls** sous Linux ^^) renvoie

```

[admin@Caen] > ip route print
Flags: X - disabled, A - active, D - dynamic,
C - connect, S - static, r - rip, b - bgp, o - ospf, m - mme,
B - blackhole, U - unreachable, P - prohibit
#    DST-ADDRESS    PREF-SRC    GATEWAY    DISTANCE
0   ADC  1.1.1.1/32      1.1.1.1      lo          0
1   ADo  2.2.2.2/32      10.0.0.2     110
2   ADo  3.3.3.3/32      10.0.0.2     110
3   ADC  10.0.0.0/30      10.0.0.1     ether1      0
4   ADC  10.0.0.4/30      10.0.0.5     ether2      0
5   ADC  10.0.0.8/30      10.0.0.9     ether3      0
6   ADo  10.0.0.12/30     10.0.0.2     110
7   ADo  10.0.0.16/30     10.0.0.2     110
8   ADo  10.0.0.20/30     10.0.0.2     110
9   ADo  10.0.0.24/30     10.0.0.2     110
10  ADo  10.0.0.28/30     10.0.0.2     110

```

### 5.3.6 La table de routage avec tous les routeurs allumés !

Voici la table de routage complète sur le routeur de Caen quand tous les routeurs du Backbone sont allumés !

```

[admin@Caen] > ip route print
Flags: X - disabled, A - active, D - dynamic,
C - connect, S - static, r - rip, b - bgp, o - ospf, m - mme,
B - blackhole, U - unreachable, P - prohibit
#    DST-ADDRESS    PREF-SRC    GATEWAY    DISTANCE
0   ADC  1.1.1.1/32      1.1.1.1      lo          0
1   ADo  2.2.2.2/32      10.0.0.2     110
2   ADo  3.3.3.3/32      10.0.0.2     110
3   ADo  4.4.4.4/32      10.0.0.10    110
4   ADo  5.5.5.5/32      10.0.0.10    110
5   ADo  6.6.6.6/32      10.0.0.10    110
6   ADo  7.7.7.7/32      10.0.0.10    110
7   ADo  8.8.8.8/32      10.0.0.6     110
8   ADo  9.9.9.9/32      10.0.0.6     110
9   ADC  10.0.0.0/30      10.0.0.1     ether1      0
10  ADC  10.0.0.4/30      10.0.0.5     ether2      0
11  ADC  10.0.0.8/30      10.0.0.9     ether3      0
12  ADo  10.0.0.12/30     10.0.0.2     110
13  ADo  10.0.0.16/30     10.0.0.2     110
14  ADo  10.0.0.20/30     10.0.0.2     110
15  ADo  10.0.0.24/30     10.0.0.10    110
16  ADo  10.0.0.28/30     10.0.0.2     110
17  ADo  10.0.0.32/30     10.0.0.10    110
18  ADo  10.0.0.36/30     10.0.0.2     110
19  ADo  10.0.0.40/30     10.0.0.10    110
20  ADo  10.0.0.44/30     10.0.0.10    110
21  ADo  10.0.0.48/30     10.0.0.10    110
22  ADo  10.0.0.52/30     10.0.0.10    110
23  ADo  10.0.0.56/30     10.0.0.10    110
24  ADo  10.0.0.60/30     10.0.0.10    110
25  ADo  10.0.0.64/30     10.0.0.10    110
26  ADo  10.0.0.68/30     10.0.0.6     110
27  ADo  10.0.0.72/30     10.0.0.10    110
28  ADo  10.0.0.76/30     10.0.0.6     110
29  ADo  10.0.0.80/30     10.0.0.10    110
30  ADo  11.11.11.11/32   10.0.0.10    110

```

On voit bien les 31 réseaux eux non 33, Ah oui c'est vrai Genève n'est pas configuré...

Allez en TP vous annexe la Suisse et vous y mettez vos sous euh non la bonne configuration en vous inspirant de la configuration d'un autre routeur !☺



### 5.3.7 Le redistribution

Il est possible de redistribuer des routes apprises par un protocole de routage dans un autre protocole de routage.

On peut redistribuer par exemple ce qui est connecté dans OSPF.

Avec les routeurs Mikrotik c'est la commande :

**/routing ospf instance set 0 router-id=9.9.9.9 redistribute-connected=as-type-1**

Imaginons que l'on désire ajouter des clients à ce **backbone**.

C'est simple, il suffit de relier à une interface des routeurs

Voici ce que cela donnerait pour Rennes et Marseille

```
/routing ospf instance
set [ find default=yes ] redistribute-connected=as-type-1 router-id=6.6.6.6
/routing ospf network
add area=backbone network=6.6.6.6/32
add area=backbone network=10.0.0.48/30
add area=backbone network=10.0.0.52/30
add area=backbone network=10.0.0.56/30
add area=backbone network=10.0.0.60/30
```

Rennes

/ip address

add address=192.168.1.254/24 interface=ether3

/routing ospf instance set 0 router-id=9.9.9.9 redistribute-connected=as-type-1 router-id=9.9.9.9

Marseille

/ip address

add address=192.168.2.254/24 interface=ether3

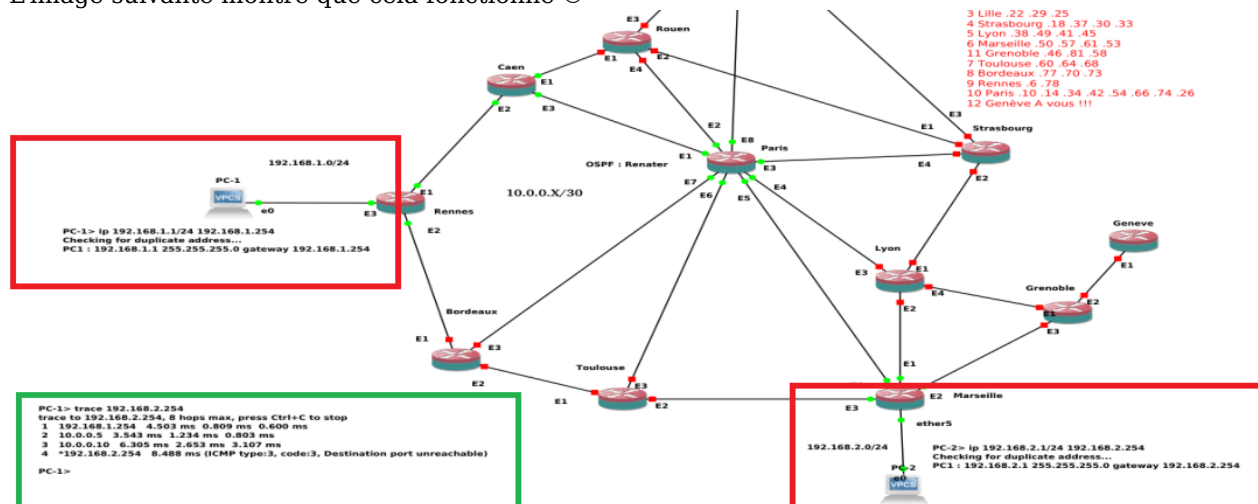
/routing ospf instance set 0 router-id=6.6.6.6 redistribute-connected=as-type-1 router-id=6.6.6.6

On redistribue via OSPF les réseaux connus par connecté !

La table de routage des routeurs associés à cette aire est alors complétée

On y ajoute les réseaux 192.168.1.0/24 et 192.168.2.0/24

L'image suivante montre que cela fonctionne ☺



**Remarque :** Il est possible de faire plein de types de redistribution mais cela dépasse ce cours

Plus d'informations ici : <https://wiki.mikrotik.com/wiki/Manual:Routing/OSPF>

```
[admin@Marseille] /routing> /ip route print
Flags: X - disabled, A - active, D - dynamic,
C - connect, S - static, r - rip, b - bgp, o - ospf, m - mme,
B - blackhole, U - unreachable, P - prohibit
```

#	DST-ADDRESS	PREF-SRC	GATEWAY	DISTANCE
0	ADo 1.1.1.1/32		10.0.0.54	110
1	ADC 6.6.6.6/32	6.6.6.6	lo	0
2	ADo 9.9.9.9/32		10.0.0.54	110
3	ADo 10.0.0.0/30		10.0.0.54	110
4	ADo 10.0.0.4/30		10.0.0.54	110
5	ADo 10.0.0.8/30		10.0.0.54	110
6	ADo 10.0.0.12/30		10.0.0.54	110
7	ADo 10.0.0.24/30		10.0.0.54	110
8	ADo 10.0.0.32/30		10.0.0.54	110
9	ADo 10.0.0.40/30		10.0.0.54	110
10	ADC 10.0.0.48/30	10.0.0.50	ether1	0
11	ADC 10.0.0.52/30	10.0.0.53	ether4	0
12	ADC 10.0.0.56/30	10.0.0.57	ether2	0
13	ADC 10.0.0.60/30	10.0.0.61	ether3	0
14	ADo 10.0.0.64/30		10.0.0.54	110
15	ADo 10.0.0.72/30		10.0.0.54	110
16	ADo 10.0.0.76/30		10.0.0.54	110
17	ADo 192.168.1.0/24		10.0.0.54	110
18	ADC 192.168.2.0/24	192.168.2.254	ether5	0

```
[- [Q quit|D dump]
```

## 6 Conclusion

Le routage statique c'est pour les CE (**Customer Edge**) et les PC en LAN, jamais pour les PE (**Provider Edge**).

Le routage dynamique n'a rien de difficile, ça fonctionne sans problème !

Nous venons de voir ce qui existe dans tous les ISP ou FAI.

Mais heureusement que l'on utilise pas que cela sur Internet sinon on ne ferait pas le dixième de ce que l'on y fait.

Le routage est un mécanisme du 20<sup>ième</sup> siècle au millénaire dernier.

Depuis on a installé des mécanismes s'appuyant sur le routage et qui permettent qu'il n'y ai pas de bouchons !

Et oui, c'est bien de tous prendre la route la plus rapide, mais cette route devient la plus lente... surtout aux heures de travail...

Le pont de Calix ☺

Cela se nomme MPLS pour ceux qui voudraient approfondir.

Il n'y aura pas de cours dessus !