

République Algérienne Démocratique et Populaire  
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

---

CENTRE UNIVERSITAIRE DE MILA  
INSTITUT DES SCIENCES ET DE LA TECHNOLOGIE

Mémoire de fin d'étude  
Présenté pour l'obtention du diplôme de

---

## LICENCE ACADEMIQUE

---

Domaine : **Mathématiques et Informatique**  
Filière : **Informatique**  
Spécialité : **INFORMATIQUE**

### THÈME

---

## **Simulation d'un protocole de routage à vecteur de distance à l'aide des IPCs sous Linux**

---

Présenté par :

- Imène Kadri
- Besma Flifla

Sous la direction de  
**M. Bilal DOUAS**

Année universitaires 2010 / 2011

# TABLE DES MATIÈRES

<b>Table des matières</b>	<b>ii</b>
<b>Table des figures</b>	<b>v</b>
<b>INTRODUCTION GÉNÉRALE</b>	<b>1</b>
<b>Chapitre 01</b>	<b>2</b>
<b>I. RESEAUX INFORMATIQUES ET INTERNET</b>	<b>2</b>
<b>I.1. Réseaux informatiques</b>	<b>3</b>
<b>I.2. Classification des réseaux informatiques</b>	<b>3</b>
I.2.1. Classification selon la taille du réseau	3
I.2.1.1 Réseaux personnels (PAN : Personal Area Network)	3
I.2.1.2 Réseaux locaux (LAN : Local Area Network)	3
I.2.1.3 Réseaux métropolitains (MAN : Metropolitan Area Networks)	3
I.2.1.4 Réseaux étendus (WAN : Wide Area Networks)	3
I.2.2. Classification selon la topologie	4
I.2.2.1 Réseaux en étoile	4
I.2.2.2 Réseaux en anneau	4
I.2.2.3 Réseaux en bus	4
<b>I.3. Applications des réseaux</b>	<b>4</b>
<b>I.4. Architecture des réseaux</b>	<b>5</b>
I.4.1. modèle OSI	5
I.4.1.1 La couche physique	6
I.4.1.2 La couche liaison	6
I.4.1.3 La couche réseau	6
I.4.1.4 La couche Transport	6
I.4.1.5 La couche Session	6
I.4.1.6 La couche présentation	6
I.4.1.7 La couche application	7
I.4.2. Le modèle TCP/IP	7
I.4.2.1 La couche d'accès au réseau	7
I.4.2.2 La couche internet	7
I.4.2.3 La couche transport	7
I.4.2.4 La couche application	8
I.4.3. Comparaison entre OSI et TCP/IP	8
I.4.4. Terminologies	9
I.4.4.1 Adressage logique (Adressage IP)	9
I.4.4.2 adresse IP	9
I.4.4.3 sous réseau	9
I.4.4.4 le masque de réseau et de sous réseau:	9
I.4.4.5 CIDR (Classless InterDomain Routing)	10
I.4.4.6 VLSM (Variable Length Subnet Mask)	10
<b>I.5. Equipements d'interconnexion de réseaux</b>	<b>10</b>
I.5.1. Répéteur	10

I.5.2.	Concentrateur (hub)	10
I.5.3.	Pont (bridge)	11
I.5.4.	Commutateur(Switch)	11
I.5.5.	Routeur	11
<b>I.6.</b>	<b>internet et Internet :</b>	<b>11</b>
I.6.1.	Histoire d'Internet :	11
I.6.2.	Structure d'Internet	13
<b>Chapitre 02</b>		<b>15</b>
<b>II. Le Routage ip</b>		<b>15</b>
<b>II.1. Le routage dans l'Internet</b>		<b>15</b>
II.1.1.	Le routeur	16
II.1.2.	La table de routage	17
<b>II.2. Classification des techniques de routage</b>		<b>18</b>
II.2.1.	Selon la distribution	18
II.2.1.1	Routage depuis la source	18
II.2.1.2	Routage distribué	18
II.2.2.	Selon le mode d'apprentissage	18
II.2.2.1	Routage Statique	18
II.2.2.2	Routage Dynamique	18
II.2.3.	Selon l'algorithme de base	18
<b>II.3. Algorithmes de routage à vecteur de distance</b>		<b>18</b>
II.3.1.	Caractéristique de l'algorithme	19
II.3.2.	exemple	19
II.3.3.	Inconvénients de l'algorithme	20
<b>II.4. Algorithmes de routage à états de liens</b>		<b>20</b>
<b>II.5. Protocoles de routages</b>		<b>20</b>
II.5.1.	Domaine ou système autonome	20
II.5.1.1	IGP	21
II.5.1.2	EGP	21
II.5.2.	Protocoles de routages et algorithmes à vecteur de distance	21
II.5.2.1	RIP	23
II.5.2.2	IGRP	23
II.5.2.3	EIGRP	23
II.5.3.	Convergence d'un protocole	24
<b>II.6. RIP</b>		<b>24</b>
II.6.1.	Déroulement du protocole	24
II.6.2.	format de message RIP	25
II.6.3.	Le problème de convergence lente	26
II.6.4.	RIP et le problème du bouclage à l'infini 'Counting to infinity'	27
II.6.5.	Périodes RIP	27
II.6.6.	La table de routage de RIP	27
II.6.7.	Limites du protocole	28
II.6.7.1	Réseau de taille limitée	28
II.6.7.2	Pas de support pour CIDR / VLSM	28

II.6.7.3	Métrieque faible	28
II.6.7.4	Consommation des ressources du réseau	28
II.6.8.	Améliorations de RIPv2 par rapport à RIPv1	28
II.6.8.1	format de message RIPv2	29
II.6.8.2	Compatibilité avec RIP	30
<b>II.7.</b>	<b>Critères d'évaluation des protocoles de routage</b>	<b>30</b>
<b>Chapitre 03</b>		<b>32</b>
<b>III.</b>	<b>Simulation d'un protocole de routage à vecteur de distance</b>	<b>32</b>
<b>III.1.</b>	<b>Environnement de travail</b>	<b>32</b>
III.1.1.	Plateforme	32
III.1.1.1	Linux c'est quoi ?	32
III.1.1.2	Les distributions	33
III.1.2.	Programmation Orienté Objet en C++	34
III.1.3.	Les IPCs	34
<b>III.2.</b>	<b>Détails technique de notre application</b>	<b>35</b>
III.2.1.	Les Timers	36
III.2.2.	Les principales structures	36
III.2.3.	Dictionnaire de donnée :	37
<b>III.3.</b>	<b>Fonctionnement générale</b>	<b>37</b>
III.3.1.	Le manager	37
III.3.2.	Les processus routeurs	38
<b>III.4.</b>	<b>Exemple d'illustration n° 01</b>	<b>38</b>
<b>III.5.</b>	<b>Exemple d'illustration n° 02</b>	<b>41</b>
<b>III.6.</b>	<b>Phénomène du comptage à l'infinie : Exemple</b>	<b>48</b>
<b>III.7.</b>	<b>Conclusion</b>	<b>51</b>
	<b>CONCLUSION GÉNÉRALE</b>	<b>52</b>
	<b>RÉFÉRENCES BIBLIOGRAPHIQUES</b>	<b>53</b>

# TABLE DES FIGURES

Figure 1: Le modèle OSI .....	5
Figure 2: Comparaison entre OSI et TCP/IP. ....	8
Figure 3: Internet, un réseau de systèmes autonomes .....	13
Figure 4: Structure hiérarchique d'Internet.....	14
Figure 5: Le routeur.....	16
Figure 6: Routage des données – Exemple .....	17
Figure 7: Equation de Bellman-Ford .....	18
Figure 8: Algorithme de Bellman-Ford.....	19
Figure 9: Algorithme de Bellman-Ford – Exemple .....	19
Figure 10: IGP VS. EGP.....	21
Figure 11: Message RIP .....	25
Figure 12: Message RIPv2. ....	29
Figure 13: Récapitulatif des protocoles de routage .....	30
Figure 14: Comparaison entre les protocoles de routages existants.....	31
Figure 15: Diagramme de classes.....	35

# INTRODUCTION GÉNÉRALE

Les réseaux de données que nous utilisons quotidiennement pour apprendre, jouer et travailler sont aussi bien des petits réseaux locaux que des grands inter réseaux mondiaux. Peut-être avez-vous chez vous un routeur et deux ordinateurs, voire plus. L'entreprise qui vous emploie utilise peut-être plusieurs routeurs et commutateurs pour répondre aux besoins de communication de données de centaines, voire de milliers d'ordinateurs.

Si le réseau est d'une taille importante alors la tâche d'acheminement des paquets devient plus compliquée et doit être automatisé. C'est le rôle des protocoles de routage dynamique. Ils sont généralement utilisés dans des réseaux plus importants pour réduire la surcharge administrative et fonctionnelle liée à l'utilisation exclusive de routes statiques. Un réseau utilise généralement à la fois un protocole de routage dynamique et des routes statiques. Dans la plupart des réseaux, un seul protocole de routage dynamique est utilisé. Toutefois, différentes parties du réseau peuvent utiliser des protocoles de routage différents.

Dans ce mémoire, nous allons commencer par la présentation des concepts liés aux réseaux informatiques. Puis nous discutons le problème du routage avant d'aborder dans le détail les protocoles de routage à vecteur de distance.

Les protocoles de routages à vecteur de distance constituent la solution idéale pour les domaines de petite taille mais ils ont également des inconvénients. Nous présentons dans ce mémoire une étude sur les techniques utilisées pour le routage dans les réseaux fixes notamment ceux à base de vecteur de distance en précisant les lacunes dans ces protocoles là comme RIP. Nous avons réalisé une simulation à l'aide des IPCs sous Linux pour constater le fonctionnement de ce type de protocole. Elle est présentée dans le dernier chapitre.

---

# Chapitre 01

## I. RESEAUX INFORMATIQUES ET INTERNET

---

Au début de l'ère informatique les ordinateurs étaient d'une taille importante mais avec le temps, ils sont devenus de petites machines pouvant tenir sur un bureau ou même dans une poche. Nous avons passé d'une période où seulement les entreprises et les administrations peuvent acquérir un ordinateur, vers une ère où tout le monde peut avoir son ordinateur personnel ou même plusieurs ordinateurs de différents types (Laptops, Smartphones ...). Cette omniprésence de l'équipement informatique a favorisé plusieurs formes d'utilisation de ces équipements particulièrement en les reliant ensemble.

Parmi les éléments essentiels à l'existence humaine, le besoin de communiquer arrive juste après le besoin de survie. Le besoin de communiquer est aussi important pour nous que l'air, l'eau, la nourriture et le gîte. Nos besoins de communication et de partage de ressources ont incité les gens à connecter les ordinateurs entre eux. Cela a donné naissance aux réseaux informatiques. Les premiers réseaux de données se limitaient à échanger des informations reposant sur des caractères entre des systèmes informatiques connectés, les réseaux modernes comme l'internet ont évolué pour prendre en charge le transfert audio, des flux vidéo, du texte et des graphismes entre des périphériques de types très différents et des applications diverses.

Ce chapitre vise à vous introduire aux concepts liés aux réseaux informatiques et à Internet particulièrement.

## *1.1. Réseaux informatiques*

Un réseau informatique est un ensemble d'ordinateurs (et autres équipements autonomes) connectés entre eux à l'aide d'un système de communication. Le but visé peut être résumé dans les trois points suivants :

- Permettre l'échanger des données.
- Permettre aux entités dispersées sur le réseau toute forme de coopération.
- Partager les ressources.

Deux ordinateurs d'un réseau doivent donc être capables d'échanger des informations entre eux.

## *1.2. Classification des réseaux informatiques*

On peut classifier les réseaux selon différents critères :

### *1.2.1. Classification selon la taille du réseau*

#### **1.2.1.1 Réseaux personnels (PAN : Personal Area Network)**

Les réseaux personnels constituent une forme très récente qui permet de créer des micro-réseaux ad hoc à partir d'un ensemble d'équipements mobiles (téléphones portables, Smartphones, assistants personnels, ...) en utilisant le canal radio comme support de transmission.

#### **1.2.1.2 Réseaux locaux (LAN : Local Area Network)**

Un réseau local s'étend généralement sur une zone géographique réduite s'étend de moins d'un mètre à 1Km avec un nombre limité de machines. Ce type de réseaux fournit des services aux utilisateurs au sein d'une structure organisationnelle commune. Telle que la maison, les établissements de toutes sorte ou les petites entreprises. Les LAN ont la caractéristique d'être très faciles à monter et ils sont très performants (débit de 10Mbit/s à 1Gbits/s).

#### **1.2.1.3 Réseaux métropolitains (MAN : Metropolitan Area Networks)**

Les MANs sont des réseaux qui relient des LANs ou des ordinateurs dispersés dans une surface de l'ordre de quelques kilomètres (une ville ou une zone urbaine). A cause de la distance plus importante des équipements spéciaux d'interconnexion sont requis. Ce type de réseaux est utilisé par les entreprises qui disposent de plusieurs sièges dans la même ville par exemple.

#### **1.2.1.4 Réseaux étendus (WAN : Wide Area Networks)**

Le WAN relie des ordinateurs ou des LANs séparés par de grandes distances (sites d'un pays). Cela est réalisé généralement en utilisant des connexions via un réseau séparé de fournisseurs de services de télécommunications.

## *1.2.2. Classification selon la topologie*

La topologie constitue la façon dans laquelle les nœuds sont liés. On distingue principalement trois topologies:

### **1.2.2.1 Réseaux en étoile**

Les équipements de réseaux (les nœuds) sont reliés à un nœud centrale. Cette topologie a deux inconvénients importants:

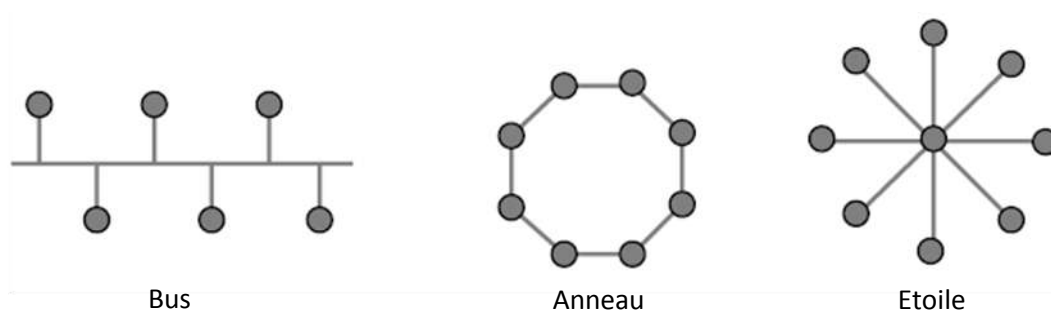
- Le nœud central subit une charge très importante par rapport aux autres.
- Si le nœud central tombe en panne aucune communication n'est possible.

### **1.2.2.2 Réseaux en anneau**

Avec cette topologie toutes les stations sont reliées entre elles dans une boucle fermée (anneau). Un message envoyé sur le réseau traverse éventuellement plusieurs nœuds intermédiaires avant d'arriver à destination. Comme pour la topologie en étoile, celle-ci aussi n'est pas tolérante aux pannes car le départ d'un nœud provoque un dysfonctionnement du réseau.

### **1.2.2.3 Réseaux en bus**

Avec cette topologie un segment linéaire est utilisé pour connecter tous les nœuds. Un message envoyé dans ce type de réseaux peut être intercepté par tous les nœuds liés au bus mais une panne d'un nœud n'affecte que ce dernier. Le réseau n'est pas affecté.



## *1.3. Applications des réseaux*

- Communications entre utilisateurs
  - o Messagerie électronique
  - o Messagerie instantanée
- Transfert de fichiers
- Accès distant
- Clustering
- Systèmes informatiques et traitements distribués

## *1.4. Architecture des réseaux*

Il existe deux types de modèles de réseau de base : les modèles de protocole et les modèles de référence. Un modèle de protocole fournit un modèle qui correspond étroitement à la structure d'une suite de protocoles particulière. L'ensemble hiérarchique des protocoles associés dans une suite représente généralement toutes les fonctionnalités requises à l'interface entre le réseau humain et le réseau de données. Le modèle TCP/IP est un modèle de protocole, car il décrit les fonctions qui interviennent à chaque couche de protocoles au sein de la suite TCP/IP.

Un modèle de référence fournit une référence commune pour maintenir la cohérence dans tous les types de protocoles et de services réseau. Un modèle de référence n'est pas destiné à être une spécification d'implémentation, ni à fournir un niveau de détail suffisant pour définir précisément les services de l'architecture réseau. Le principal objectif d'un modèle de référence est d'aider à obtenir une compréhension plus claire des fonctions et du processus impliqués.

Le modèle OSI (Open Systems Interconnection) constitue le modèle de référence interréseau le plus répandu. Il est utilisé pour la conception de réseaux de données, pour les spécifications de fonctionnement et pour le dépannage.

Bien que les modèles TCP/IP et OSI soient les principaux modèles utilisés lorsqu'il s'agit de fonctionnalités réseau, les concepteurs de protocoles, de services ou de périphériques réseau peuvent créer leurs propres modèles pour représenter leurs produits. Enfin, les concepteurs doivent communiquer avec l'industrie en associant leurs produits ou leurs services aux modèles OSI ou TCP/IP ou aux deux.

### *1.4.1. modèle OSI*



Figure 1: Le modèle OSI

Le modèle de référence OSI (Open System Interconnection) publié en 1984 est le modèle descriptif de réseau créé par l'ISO, il est considéré comme le meilleur outil disponible pour décrire l'envoi et la réception de données sur un réseau. Le modèle de référence OSI comporte sept couches numérotées, chacune illustrant une fonction réseau bien précise.

#### 1.4.1.1 **La couche physique**

Cette couche s'occupe de la transmission des données brutes sur les supports physiques de communication. Elle définit les spécifications électriques, mécaniques et fonctionnelles pour activer, maintenir ou désactiver le lien physique. De telles caractéristiques comme le voltage, la synchronisation, les fréquences, la distance maximale, les connecteurs, ... tout ça est défini par les spécifications de cette couche.

#### 1.4.1.2 **La couche liaison**

La couche liaison de données assure une transmission fiable des données à travers la couche physique. Les techniques utilisées sont souvent : la correction d'erreurs, les retransmissions, les acquittements, le contrôle de l'ordre de livraison et le contrôle de flux.

#### 1.4.1.3 **La couche réseau**

C'est une couche complexe qui assure la connectivité, le routage et la sélection de chemins entre deux ordinateurs qui ne sont pas forcément sur le même réseau. Cette couche s'occupe de l'adressage logique à l'aide de protocoles comme IP, IPX ou AppleTalk.

#### 1.4.1.4 **La couche Transport**

La couche transport du modèle OSI, sert souvent à décrire des services ou des fonctions générales qui gèrent des conversations individuelles entre des hôtes source et de destination. Ces fonctions incluent le reçu, la reprise sur erreur et le séquençement.

#### 1.4.1.5 **La couche Session**

Comme l'implique le nom de la couche session, les fonctions s'exécutant au niveau de cette couche permettent un dialogue entre les applications source et de destination. La couche session traite l'échange des informations pour initier et maintenir un dialogue et pour redémarrer les sessions interrompues ou inactives pendant une longue période.

#### 1.4.1.6 **La couche présentation**

La couche présentation assure que les données envoyées par la couche application peuvent être lues à la même couche du récepteur. La couche présentation remplit trois fonctions principales :

- Codage et conversion des données de la couche application afin que les données issues du périphérique source puissent être interprétées par l'application appropriée sur le périphérique de destination ;
- Compression des données de sorte que celles-ci puissent être décompressées par le périphérique de destination ;
- Chiffrement des données en vue de leur transmission et déchiffrement des données reçues par le périphérique de destination.

#### 1.4.1.7 **La couche application**

Cette couche comporte toutes les applications qui utilisent les services du réseau.

### 1.4.2. *Le modèle TCP/IP*

Le premier modèle de protocole en couches pour les communications inter réseau fut créé au début des années 70 et est appelé modèle Internet. Il définit quatre catégories de fonctions qui doivent s'exécuter pour que les communications réussissent. L'architecture de la suite de protocoles TCP/IP suit la structure de ce modèle. Pour cette raison, le modèle Internet est généralement appelé modèle TCP/IP. La plupart des modèles de protocole décrivent une pile de protocoles spécifique au fournisseur. Cependant, puisque le modèle TCP/IP est une norme ouverte, aucune entreprise ne contrôle la définition du modèle. Les définitions de la norme et des protocoles TCP/IP sont traitées dans un forum public et définies dans un ensemble de documents disponible au public. Ces documents sont appelés documents RFC (Request For Comments). Ils contiennent les spécifications formelles des protocoles de données ainsi que des ressources qui décrivent l'utilisation des protocoles.

#### 1.4.2.1 **La couche d'accès au réseau**

Constitue l'interface avec le réseau et est constituée d'un driver du système d'exploitation et d'une carte d'interface de l'ordinateur avec le réseau.

#### 1.4.2.2 **La couche internet**

Gère l'acheminement des paquets à travers le réseau. Elle comprend aussi les protocoles ICMP (Internet Control Message Protocol) et IGMP (Internet Group Management Protocol)

#### 1.4.2.3 **La couche transport**

assure tout d'abord une communication de bout en bout en faisant abstraction des machines intermédiaires entre l'émetteur et le destinataire. Elle s'occupe de réguler le flux de données et assure un transport fiable (données transmises sans erreur et reçues dans l'ordre de leur émission) dans le cas de TCP (Transmission Control Protocol) ou non fiable dans le cas de UDP (User Datagram

Protocol). Pour UDP, il n'est pas garanti qu'un paquet (appelé dans ce cas datagramme) arrive à bon port, c'est à la couche application de s'en assurer.

#### 1.4.2.4 *La couche application*

Constituée des programmes utilisateurs comme Telnet (connexion à un ordinateur distant), FTP (File Transfert Protocol ), SMTP (Simple Mail Transfert Protocol ), etc...

### 1.4.3. *Comparaison entre OSI et TCP/IP*

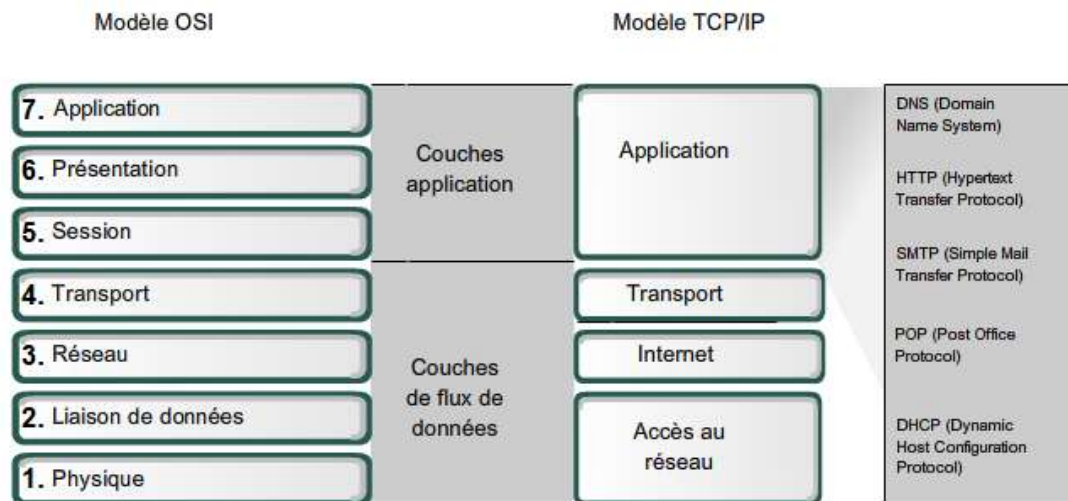


Figure 2: Comparaison entre OSI et TCP/IP.

Les protocoles qui constituent la suite de protocoles TCP/IP peuvent être décrits selon les termes du modèle de référence OSI. Dans le modèle OSI, la couche d'accès réseau et la couche application du modèle TCP/IP sont encore divisées pour décrire des fonctions discrètes qui doivent intervenir au niveau de ces couches.

Au niveau de la couche d'accès au réseau, la suite de protocoles TCP/IP ne spécifie pas quels protocoles utiliser lors de la transmission à travers un support physique ; elle décrit uniquement la remise depuis la couche Internet aux protocoles réseau physiques. Les couches OSI 1 et 2 traitent des procédures nécessaires à l'accès aux supports et des moyens physiques pour envoyer des données à travers un réseau.

Les principaux parallèles entre les deux modèles de réseau se situent aux couches 3 et 4 du modèle OSI. La couche 3 du modèle OSI, la couche réseau, est utilisée presque partout dans le monde afin de traiter et de documenter l'éventail des processus qui interviennent dans tous les réseaux de données pour adresser et acheminer des messages à travers un interréseau. Le protocole IP est le protocole de la suite TCP/IP qui contient la fonctionnalité décrite à la couche 3.

La couche 4, la couche transport du modèle OSI, sert souvent à décrire des services ou des fonctions générales qui gèrent des conversations individuelles entre des hôtes source et de destination. Ces fonctions incluent le reçu, la reprise sur erreur et le séquençement. Au niveau de cette couche, les protocoles TCP/IP Transmission Control Protocol (TCP) et User Datagram Protocol (protocole UDP) fournissent les fonctionnalités nécessaires.

La couche application TCP/IP inclut plusieurs protocoles qui fournissent des fonctionnalités spécifiques à plusieurs applications d'utilisateur final. Les couches 5, 6 et 7 du modèle OSI sont utilisées en tant que références pour les développeurs et les éditeurs de logiciels d'application, afin de créer des produits qui doivent accéder aux réseaux pour des communications.

#### *1.4.4. Terminologies*

##### **1.4.4.1 Adressage logique (Adressage IP)**

L'adressage est l'une des premières fonctions des protocoles de la couche réseau. Il permet de mettre en œuvre la transmission de données entre des hôtes situés sur un même réseau ou sur des réseaux différents. Tout équipement sur un réseau TCP/IP doit disposer d'une adresse IP unique, cette adresse permet à un ordinateur de localiser un autre ordinateur sur le réseau.

##### **1.4.4.2 adresse IP**

Adresse 32 bits assignée à des hôtes utilisant TCP/IP. Une adresse IP appartient à l'une des cinq classes (A, B, C, D ou E) et est rédigée sous forme de quatre octets séparés par des points. Chaque adresse comprend un numéro de réseau, un numéro de sous-réseau optionnel et un numéro de machine. Les numéros de réseau et de sous-réseau utilisés ensemble servent au routage, tandis que le numéro de machine est utilisé pour s'adresser à un hôte particulier du réseau ou du sous-réseau.

##### **1.4.4.3 sous réseau**

La création de sous réseau est un moyen de diviser le réseau et de fractionner un réseau important ou très important en segments plus petit ; plus efficaces et plus faciles à gérer.

##### **1.4.4.4 le masque de réseau et de sous réseau:**

Combinaison de bits utilisée pour décrire la portion d'une adresse qui désigne le réseau ou le sous-réseau et la portion qui désigne l'hôte. Le masque d'un réseau est une adresse IP dont les bits de la partie réseau sont à 1 et ceux de la partie hôte à 0.

#### 1.4.4.5 **CIDR (Classless InterDomain Routing)**

Au début des années 90, suite à l'afflux des nouveaux utilisateurs d'internet, surtout des entreprises, le système d'attribution des réseaux IP basé sur le système des classes commença à montrer ses limites car la taille des tables de routage se mit à gonfler exponentiellement. Un nouveau système de répartition des adresses en dehors des classes fut mis en place: le CIDR .

CIDR est une technique prise en charge par les protocoles de routage et basée sur le regroupement des routes. Le routage CIDR permet aux routeurs de regrouper des routes afin de réduire la quantité d'informations de routage transportée par les routeurs de cœur de réseau. Avec le routage CIDR, plusieurs réseaux IP sont perçus comme une seule grande entité par les réseaux à l'extérieur du groupe. CIDR permet donc de définir des "masques" réseau plus grands que celui de la "classe naturelle" des préfixes réseaux.

La notion de classe d'adresses n'a aucune signification pour les routeurs compatibles CIDR. En effet, la partie réseau de l'adresse est déterminée par le masque de sous-réseau du réseau, également connu sous le nom de « préfixe de réseau » ou de « longueur de préfixe » (noté /8, /19, etc.). Depuis le CIDR, la partie réseau de l'adresse n'est plus déterminée par la classe de l'adresse.

#### 1.4.4.6 **VLSM (Variable Length Subnet Mask)**

Masque de sous-réseau de longueur variable. Capacité de spécifier un masque de sous-réseau différent pour le même numéro de réseau sur des sous-réseaux différents. Le masque de sous-réseau de longueur variable peut contribuer à optimiser l'espace adresse disponible. VLSM découpe simplement un sous-réseau en sous-réseaux. Le VLSM peut être considéré comme un découpage en sous-sous-réseaux.

## ***1.5. Equipements d'interconnexion de réseaux***

### ***1.5.1. Répéteur***

Le répéteur (appelé répéteur-régénérateur) est un équipement intermédiaire qui sert à amplifier le signal numérique et ainsi étendre la distance maximale entre deux nœuds d'un réseau. Il fonctionne au de la couche physique.

### ***1.5.2. Concentrateur (hub)***

Les concentrateurs peuvent être actifs (lorsqu'ils répètent les signaux acheminés sur eux), passifs (lorsqu'ils ne répètent pas les signaux acheminés sur eux, mais qu'ils les divisent simplement) ou intelligent (fonctionne de la même façon que les concentrateurs actifs. Plus efficaces dans les situations de dépannage).

### *1.5.3. Pont (bridge)*

Dispositif qui réunit deux segments de réseaux qui utilisent le même protocole de communication et permet la circulation de paquets entre eux. Les ponts agissent au niveau de la couche liaison de données (couche 2) du modèle de référence OSI. En général, les ponts filtrent, réacheminent ou diffusent une trame en fonction de l'adresse MAC.

### *1.5.4. Commutateur(Switch)*

Il est parfois qualifié de «pont multiports», les commutateurs recherchent des informations sur les trames de données qu'ils reçoivent. Ils se servent ensuite de ces informations pour créer des tables et déterminer la destination. Ils sont plus sophistiqués qu'un pont (il permet d'atténuer la congestion dans les LAN Ethernet en réduisant le trafic et en augmentant la bande passante).

### *1.5.5. Routeur*

Un routeur est un équipement intermédiaire de couche réseau qui utilise des métriques (des valeurs qui permettent de définir le meilleur chemin) pour déterminer le chemin optimal pour le trafic réseau. Il possède les deux fonctions principales suivantes:

- Le routeur gère les tables de routage et s'assure que les autres routeurs ont connaissance des modifications apportées à la topologie du réseau (Il se sert des protocoles de routage pour échanger les informations).
- Le routeur détermine la destination des paquets à l'aide de la table de routage lorsque ceux-ci arrivent à l'une de ses interfaces. Il les transfère vers la bonne interface. Il ajoute les informations de trame de cette interface, puis transmet la trame.

## *1.6. internet et Internet :*

Le mot **internet** vient de la contraction des mots anglais « Interconnected Networks» c'est-à-dire l'interconnexion de nombreux networks ou un réseau de réseaux. Ces réseaux sont interconnectés par des dispositifs appelés '**Routeurs**'. Quand on utilise le mot **Internet** (Notez le I en majuscule) aujourd'hui on veut le réseau mondial, cependant tout réseau de réseaux est un internet.

### *1.6.1. Histoire d'Internet :*

Internet n'est pas venu bouleverser le monde informatique aussi soudainement qu'on pourrait le croire. Il est le résultat d'une longue évolution et était destiné à l'origine à un usage bien plus modeste qu'il ne l'est aujourd'hui.

L'histoire d'Internet commence en plein cœur de la guerre froide. Les Russes viennent de lancer Spoutnik, et le gouvernement américain, tiraillé entre la menace des armes nucléaires cubaines et le début du conflit armé au Vietnam, aiguillonne la guerre froide vers le front technologique. Alimentée par les subventions fédérales et par l'hystérie publique, la recherche avance à grands pas, particulièrement dans le domaine de l'informatique.

1964: Projet ARPANET: Le début du projet pour créer le premier réseau de communication distribué. L'avantage de cette technologie est qu'elle est décentralisée. Le système n'est pas dépendant d'un ordinateur principal pour coordonner les communications. Si un, deux, même plusieurs, ordinateurs tombent en panne, sont surchargés ou ne sont pas disponibles, le système peut contourner ces problèmes sans aucune difficulté. Ceci est parfait pour les besoins militaires qui créèrent le réseau.

1969: Mise en opération du projet ARPANET qui utilise un protocole de communication appelé NCP (Network Control Protocol). C'est la première fois que des ordinateurs communiquent entre eux malgré de longues distances. Le tout composait un réseau d'ordinateurs superpuissants pour la période. Le réseau survivrait même aux ravages d'une guerre atomique puisque chaque site est connecté physiquement à quatre autres. La communication et la coordination des forces armées survivantes à une attaque nucléaire sont assurées. Pour des raisons de sécurité nationale, l'armée refuse aux scientifiques l'accès au réseau militaire et l'utilisation du protocole de communication entre les sites.

1973: Les scientifiques prennent leur revanche en développant leur propre réseau à travers les universités et en utilisant un protocole de communication amélioré nommé TCP/IP (Transfert Control Protocol/ Internet Protocol). Le protocole IP s'assure que chaque ordinateur relié au réseau a une adresse unique et que chacun soit capable de communiquer avec les autres. Ce protocole est utilisé sur plusieurs plates-formes pour permettre une communication entre ceux-ci. L'accès est devenu avec le temps international. Les fondations pour l'Internet d'aujourd'hui sont placées. La recherche a permis d'augmenter le potentiel de l'Internet avec plusieurs autres services.

1989: Tim Berners-Lee conçoit la base de ce qui allait devenir le World Wide Web.

En 1992, le langage HTML qui permet de créer des documents multimédias et le protocole HTTP (Hyper Text Transfer Protocol) qui permet de véhiculer ces documents sur Internet, sont présentés au Centre Européen de Recherche Nucléaire (CERN). Le World Wide Web est né . Ce nouvel outil de diffusion

d'informations engendre la formidable expansion d'Internet dont nous sommes actuellement témoins.

1995: C'est cette année que le terme "Internet" est entré dans le vocabulaire de la plupart du monde. C'est surtout à cause d'un "nouveau" service: le World Wide Web (WWW ou W3). Plusieurs personnes confondent encore aujourd'hui le terme WWW avec l'Internet.

A cette époque, malgré sa popularité au sein de la communauté scientifique et universitaire, Internet n'a pas encore séduit le grand public. Il est vrai que les outils dont on dispose sont encore rudimentaires et le type d'information disponible relève toujours du domaine de la recherche.

On dispose alors du courrier électronique pour communiquer et d'outils comme FTP (File Transfer Protocol) et TelNet pour accéder à des banques de d'informations où l'utilisateur doit faire preuve d'intuition et compter sur la chance pour trouver ce qu'il cherche. L'arrivée de Gopher marque une étape importante dans le développement des outils d'accès à l'information disponible sur Internet. Au début des années 1990, le service informatique de l'université du Minnesota met au point une interface de recherche par thème, facile à utiliser, pour permettre à ses étudiants d'accéder aux documents informatisés répartis sur les serveurs des différentes facultés. Ce nouvel outil est baptisé Gopher en l'honneur de la mascotte de l'université. La technologie Gopher est rapidement adoptée par la communauté universitaire qui trouve enfin une façon simple et efficace d'organiser et gérer ses banques d'information électronique.

### *1.6.2. Structure d'Internet*

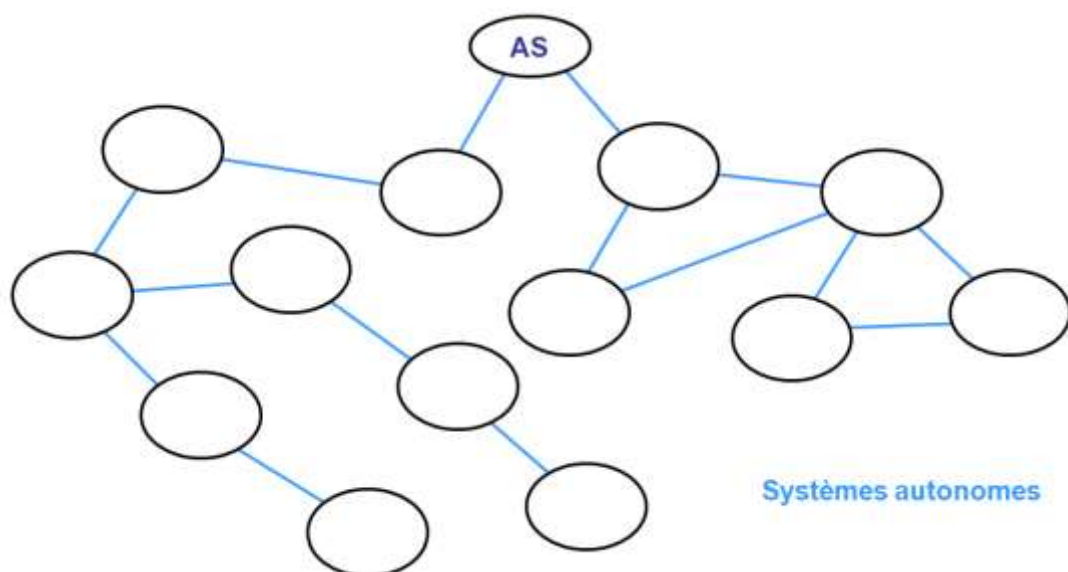


Figure 3: Internet, un réseau de systèmes autonomes

L'Internet est un réseau de systèmes autonomes, chaque système autonome est un réseau administré par une seule entité administrative et a une certaine autonomie comme son nom l'indique. Seulement, ces systèmes autonomes ne sont pas équivalents, ils sont organisés d'une manière hiérarchique. Un ISP par exemple peut relier plusieurs petit AS (universités, société, ...).

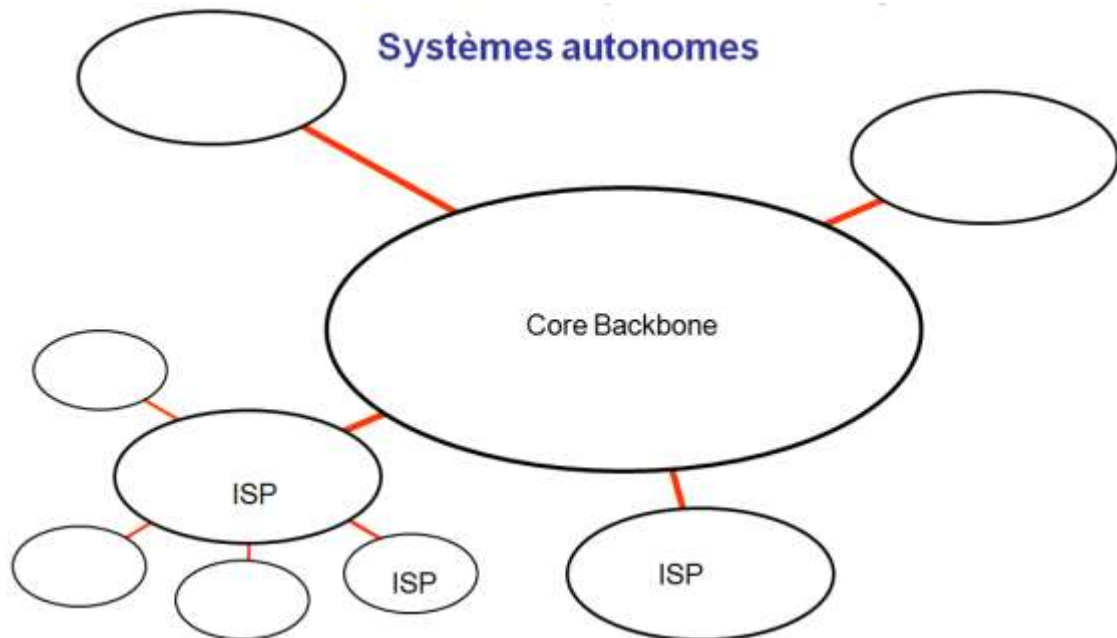


Figure 4: Structure hiérarchique d'Internet.

---

# Chapitre 02

## II. LE ROUTAGE IP

---

Les réseaux informatiques et, dans une plus grande mesure, Internet permettent aux personnes de communiquer, de collaborer et d'interagir comme jamais auparavant. Nous utilisons les réseaux de diverses manières, notamment pour les applications Web, la téléphonie IP, la vidéoconférence, les jeux interactifs, le commerce électronique et le domaine éducatif. Mais un Internet n'est pas un seul réseau logique. C'est une interconnexion de plusieurs réseaux et d'une façon très complexe.

Cela fait que la tâche de relier chaque couple d'ordinateur de cet internet devient très délicate. C'est grâce aux routeurs qu'on peut avoir cette liaison quelque en soit la complexité du réseau. Le but d'un routeur est de relier un réseau à d'autres. Le routeur est donc responsable de la transmission de paquets à travers différents réseaux. La destination du paquet IP peut être un ordinateur se trouvant dans un autre pays ou situé sur le réseau local. Dans tout les cas, les routeurs doivent transmettre ces paquets rapidement. L'efficacité des communications interréseaux dépend, en grande partie, de la capacité des routeurs à transférer des paquets de la manière la plus efficace possible.

Ce chapitre est consacré au problème du routage dans les internets ou plus précisément le routage dans l'Internet mondial.

### *II.1. Le routage dans l'Internet*

On peut définir le routage par l'opération d'acheminement des paquets de la source vers la destination à travers un ou plusieurs réseaux. Pour le faire il faut se disposer du chemin complet ou de la première partie du chemin qui doit être complétée par les stations intermédiaires. C'est une tâche de la couche réseau

(couche 3 du modèle OSI) qui consiste à acheminer les paquets sur une des interfaces du routeur. Cette tâche doit être répétée pour chaque paquet entrant, d'où la nécessité d'une prise de décision rapide. Une table maintenue par chaque routeur d'un réseau, la table de routage, permet d'établir une correspondance entre le réseau de destination (auquel appartient le destinataire du paquet), et l'adresse du prochain routeur (prochain saut) permettant d'atteindre la destination finale. Cette table de routage peut être constituée de deux manières :

- De manière statique : chaque routeur possède un fichier lu au moment de son initialisation, et qui établit, de manière fixe, la correspondance réseau de destination - prochain saut. Dans ce cas les mises à jour se font d'une façon manuelle. Cependant cette méthode se révèle fastidieuse dans le cas de gros réseaux, puisqu'il faut configurer manuellement et individuellement chaque routeur. De plus, les routeurs se révèlent incapables de s'adapter automatiquement à une modification de la topologie de réseau, et encore moins aux variations de l'état de congestion des liens.
- Les algorithmes de routage dynamiques ne nécessitent aucune configuration initiale de la table de routage : celle-ci est construite automatiquement en se basant sur des messages que s'échangent les routeurs. Les réseaux informatiques modernes emploient généralement plutôt des algorithmes de routage dynamiques. Dans la suite de ce Mémoire, nous nous intéresserons à cette deuxième catégorie.

### *II.1.1. Le routeur*

Le rôle principal d'un routeur est de relier plusieurs réseaux et de transférer des paquets destinés à ses propres réseaux ou à d'autres réseaux. Le routeur est considéré comme relevant de la couche 3 car sa décision principale de transfert est basée sur les informations du paquet IP de la couche 3, en particulier l'adresse IP de destination. C'est le processus appelé routage.

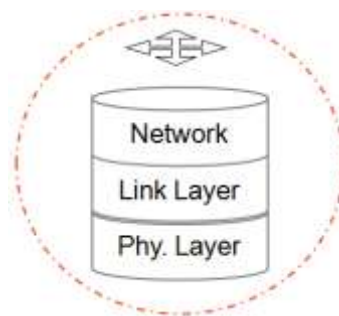


Figure 5: Le routeur.

Lorsqu'un routeur reçoit un paquet, il examine l'adresse IP de destination de celui-ci. Si celle-ci n'appartient à aucun des réseaux directement connectés au routeur, ce dernier doit transférer le paquet à un autre routeur. Chaque routeur qui reçoit un paquet consulte sa table de routage pour trouver la meilleure correspondance entre l'adresse IP de destination du paquet et l'une des adresses réseau figurant dans la table de routage. Une fois qu'une correspondance est trouvée, le paquet est encapsulé dans la trame liaison de données de couche 2 pour cette interface sortante. Le type d'encapsulation de la liaison de données dépend du type d'interface, Ethernet ou HDLC par exemple.

Au final, le paquet atteint un routeur faisant partie d'un réseau qui correspond à l'adresse IP de destination du paquet.

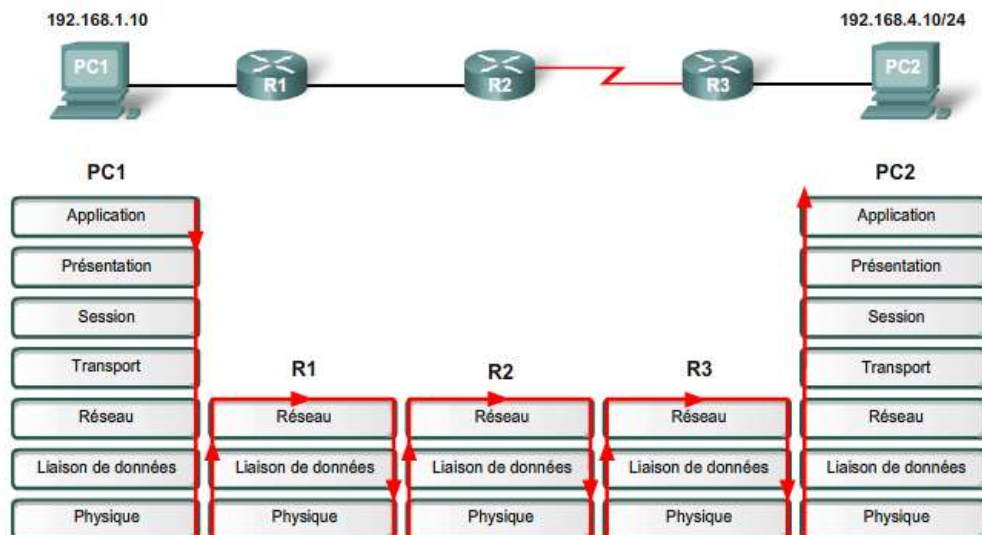


Figure 6: Routage des données – Exemple

### II.1.2. La table de routage

La table de routage est une table stocké dans chaque routeur et qui contient l'information à jour des destinations possibles. Elle est utilisée pour prendre les décisions de routage. Cette table comprend les champs suivant :

- Destination : adresse d'une machine ou d'un réseau à atteindre.
- Masque : masque réseau associé à l'adresse destination.
- Passerelle : adresse du saut suivant pour atteindre la destination correspondante.
- Interface : désigne le périphérique physique sur lequel est connecté le saut suivant.
- Métrique : distance ou autre coût de la route. Les protocoles de routage utilisent cette mesure pour déterminer le meilleur chemin d'accès à un

réseau. La mesure utilisée par le protocole de routage RIP est le nombre de sauts, autrement dit le nombre de routeurs qu'un paquet doit traverser pour atteindre un autre réseau. Le protocole OSPF utilise la bande passante pour déterminer le chemin le plus court...

## *II.2. Classification des techniques de routage*

On peut classifier les techniques de routage connus aujourd'hui selon plusieurs critères:

### *II.2.1. Selon la distribution*

#### **II.2.1.1 Routage depuis la source**

Dans ce type de routage la source doit disposer du chemin total vers la destination. La manière la plus évidente pour se faire est de disposer de la carte de topologie complète sur chaque nœud.

#### **II.2.1.2 Routage distribué**

Dans ce cas un nœud ne connaît pas l'état global de son environnement mais dispose de quelques informations qui lui permettent de décider du pas suivant pour atteindre une destination.

### *II.2.2. Selon le mode d'apprentissage*

#### **II.2.2.1 Routage Statique**

Le routeur ne connaît que les routes manuellement configurées par l'administrateur.

#### **II.2.2.2 Routage Dynamique**

Le routeur apprend les routes d'une manière automatique à l'aide de l'exécution d'un protocole spécifique.

### *II.2.3. Selon l'algorithme de base*

Les protocoles de routages existants sont basés essentiellement sur deux algorithmes : Algorithmes à vecteur de distance et les Algorithmes à états de liens.

## *II.3. Algorithmes de routage à vecteur de distance*

Cet algorithme est basé sur l'équation de Bellman-Ford qui est à la base de l'algorithme du plus court chemin dans la théorie des graphes nommé Bellman-Ford lui aussi:

$$D(x, y) = \min \{c(x, v) + D(v, y)\} \forall v \text{ voisin de } x$$

Figure 7: Equation de Bellman-Ford

Le principe est de cet algorithme est de découvrir les routes optimales après un nombre fini d'itérations, où à chaque itération chaque nœud met à jour sa distance par rapport à la source à l'aide des distances entre lui et ses voisins et aussi les distances entre ces voisins et la source.



Figure 8: Algorithme de Bellman-Ford

### II.3.1. Caractéristique de l'algorithme

- Avec cet algorithme les routeurs ne peuvent pas avoir la topologie complète du réseau donc le routage est distribué inévitablement, les seules informations qu'un routeur peut avoir sont les distances vers toutes les destinations possibles avec le saut suivant correspondant.
- La complexité de cet algorithme est polynomiale et dépend de la complexité de la topologie.
- L'algorithme ne nécessite aucune synchronisation pour converger.
- L'implémentation de cet algorithme devrait être très facile.
- La distance utilisée est initialement le nombre de pas entre les routeurs, cependant il est tout à fait possible d'envisager des métriques plus significatives.

### II.3.2. exemple

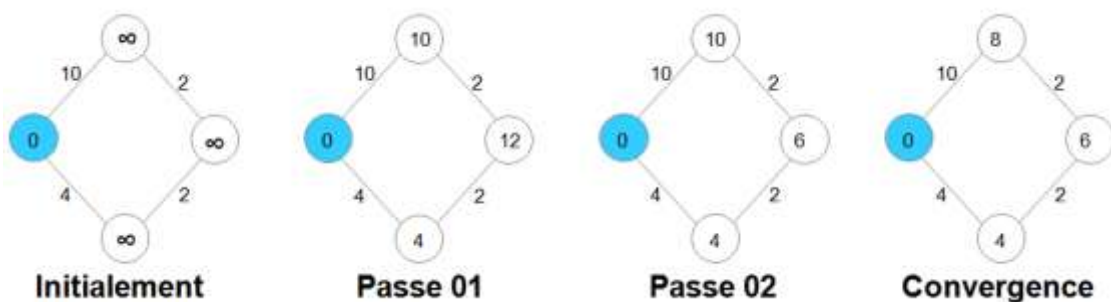


Figure 9: Algorithme de Bellman-Ford – Exemple

### *II.3.3. Inconvénients de l'algorithme*

- Temps de convergence élevé si la topologie est complexe ou très grande.
- Les messages échangés entre les routeurs concernent toute la topologie et donc sont de taille considérable.
- Si la distance a été redéfinie elle doit être choisie avec soin pour éviter le phénomène des oscillations, ce phénomène se produit si la distance choisie change très rapidement pour la même destination, dans ce cas le routeur se retrouve entraîné de basculer entre deux ou plusieurs routes.

### *II.4. Algorithmes de routage à états de liens*

Les algorithmes à état de liaisons permettent à chaque routeur d'avoir une carte complète de la topologie du réseau, les routeurs écoutent les états des liens avec leurs voisins, à chaque lien est associé un coût et un indicateur UP/DOWN, ces états sont diffusés dans tout le réseau ce qui fournira un état global du réseau à chaque nœud, On utilise l'algorithme de Dijkstra pour calculer l'ensemble des plus courts chemins.

Cet algorithme utilise l'inondation pour diffuser les états des liens dans tout le réseau, l'inondation doit atteindre tous les nœuds, éviter les boucles et terminer en un coût raisonnable. La complexité de l'algorithme est  $O(n^2)$  et peut être réduite à  $[n \cdot \log(n)]$ . Comme celui vu ci-avant, il souffre lui aussi du problème des oscillations, le coût du lien doit être bien choisi.

### *II.5. Protocoles de routages*

Un protocole est un ensemble de règles et un certain ordre d'actions à suivre pour réaliser une tâche précise. Les protocoles de routage calcul et maintiennent les tables de routage afin de permettre une tâche de routage adéquate à l'intérieur d'un domaine ou entre les domaines.

#### *II.5.1. Domaine ou système autonome*

Un système autonome (SA), également appelé domaine de routage, est un ensemble de routeurs dont l'administration est commune. Le réseau interne d'une société et le réseau d'un fournisseur de services Internet en sont des exemples. Un système autonome se compose généralement de nombreux réseaux individuels appartenant à des entreprises, des établissements scolaires et autres institutions. Dans la mesure où Internet repose sur le concept de système autonome, deux types de protocoles de routage sont nécessaires :

### II.5.1.1 IGP

Les IGP (Interior Gateway Protocol) assurent le routage à l'intérieur des domaines. Les protocoles IGP sont utilisés à des fins de routage dans un domaine de routage (les réseaux sous le contrôle d'une même organisation). Un protocole IGP permet d'assurer le routage au sein du système autonome et au sein des réseaux individuels eux-mêmes. Les protocoles IGP pour IP sont :

- RIP
- IGRP
- EIGRP
- OSPF
- IS-IS.

### II.5.1.2 EGP

Les protocoles EGP (Exterior Gateway Protocol) sont utilisés pour le routage entre systèmes autonomes. En revanche, les protocoles EGP sont conçus pour être utilisés entre différents systèmes autonomes contrôlés par des administrations distinctes. BGP est le seul protocole de routage EGP actuellement viable utilisé par Internet. BGP est un protocole à vecteur de chemin qui peut utiliser de nombreux attributs différents pour mesurer des routes. Les FAI, quant à eux, sont confrontés à des problèmes souvent plus importants que le simple choix du chemin le plus rapide.

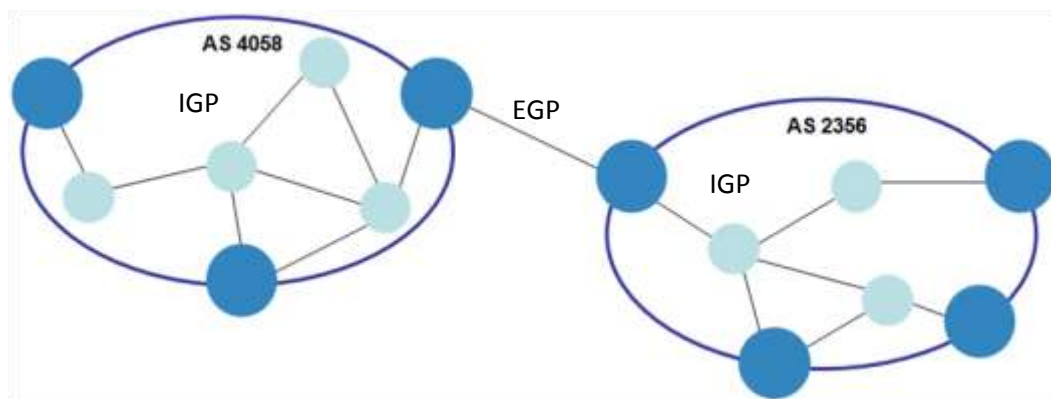


Figure 10: IGP VS. EGP

## II.5.2. Protocoles de routages et algorithmes à vecteur de distance

Les protocoles de routage sont basés sur un des deux algorithmes mentionnés précédemment. Dans le cas des protocoles à vecteur de distances, les routes sont annoncées en tant que vecteurs de distance et de direction. En envoyant régulièrement le contenu de la table de routage, et en effectuant une mise à jour des entrées uniquement lorsqu'un nouveau coût est inférieur à l'ancien, cet algorithme permet de découvrir la distance la plus courte pour atteindre un

réseau. Cependant certains défauts doivent être mis en évidence, et notamment le manque de réactivité lors du changement de topologie du réseau (panne d'un routeur par exemple), comme nous allons le voir lors de l'étude du protocole RIP. La distance est définie en termes de mesure, comme le nombre de sauts, et la direction est simplement le routeur de tronçon suivant ou l'interface de sortie. Un routeur utilisant un protocole de routage à vecteur de distance ne connaît pas le chemin complet vers un réseau de destination. Le routeur ne connaît que les éléments suivants :

- La direction ou l'interface dans laquelle les paquets doivent être transmis ;
- La distance le séparant du réseau de destination.

Certains protocoles de routage à vecteur de distance appellent le routeur pour qu'il diffuse régulièrement la totalité de la table de routage à chacun de ses voisins. Cette méthode est inefficace car les mises à jour sollicitent non seulement de la bande passante, mais également les ressources processeur du routeur pour traiter ces mises à jour. Les caractéristiques communes des protocoles de routage à vecteur de distance sont:

- Des mises à jour régulières sont envoyées à intervalles fixes (30 secondes pour le protocole RIP et 90 secondes pour le protocole IGRP). Même si la topologie est inchangée depuis plusieurs jours, des mises à jour régulières continuent d'être envoyées à tous les voisins.
- Les voisins sont des routeurs qui partagent une liaison et qui sont configurés de manière à utiliser le même protocole de routage. Le routeur ne connaît que les adresses réseau de ses propres interfaces et les adresses des réseaux distants qu'il peut atteindre par le biais de ses voisins. Il n'en sait pas davantage sur la topologie du réseau. Les routeurs utilisant un routage à vecteur de distance ne connaissent pas la topologie du réseau.
- Des mises à jour de diffusion sont envoyées à 255.255.255.255. Les routeurs voisins configurés avec le même protocole de routage traiteront ces mises à jour. Tous les autres périphériques traiteront également la mise à jour jusqu'à la couche 3, avant de la supprimer. Certains protocoles de routage à vecteur de distance utilisent des adresses de multidiffusion à la place d'adresses de diffusion.
- Des mises à jour de toute la table de routage sont envoyées régulièrement à tous les voisins, à quelques exceptions près qui seront examinées plus loin. Les voisins qui reçoivent ces mises à jour traitent l'ensemble de la mise

à jour pour trouver les informations pertinentes et suppriment le reste. Certains protocoles de routage à vecteur de distance comme EIGRP n'envoient pas de mises à jour régulières de la table de routage.

Les protocoles de routage à vecteur de distance comprennent : RIP, IGRP et EIGRP.

#### II.5.2.1 **RIP**

Le protocole RIP (Routing Information Protocol) a été initialement défini dans le document RFC 1058. Ses principales caractéristiques sont les suivantes :

- Il utilise le nombre de sauts comme mesure de sélection d'un chemin.
- Si le nombre de sauts pour un réseau est supérieur à 15, le protocole RIP ne peut pas fournir de route à ce réseau.
- Par défaut, les mises à jour de routage sont diffusées ou multidiffusées toutes les 30 secondes.

#### II.5.2.2 **IGRP**

Le protocole IGRP (Interior Gateway Routing Protocol) est un protocole propriétaire développé par Cisco. Les principales caractéristiques conceptuelles du protocole IGRP sont les suivantes :

- La bande passante, le délai, la charge et la fiabilité sont utilisés pour créer une mesure composite.
- Par défaut, les mises à jour de routage sont diffusées toutes les 90 secondes.
- Prédécesseur du protocole EIGRP, le protocole IGRP est désormais obsolète.

#### II.5.2.3 **EIGRP**

Le protocole EIGRP (Enhanced IGRP) est un protocole de routage à vecteur de distance propriétaire développé par Cisco. Ses principales caractéristiques sont les suivantes :

- Il peut effectuer un équilibrage de charge à coût inégal.
- Il utilise l'algorithme DUAL (Diffused Update Algorithm) pour calculer le chemin le plus court.
- Contrairement aux protocoles RIP et IGRP, il n'y a pas de mises à jour régulières. Des mises à jour de routage sont envoyées uniquement en cas de modification de la topologie.

### *II.5.3. Convergence d'un protocole*

On parle de convergence lorsque les tables de routage de tous les routeurs ont atteint un état de cohérence. Le réseau a convergé lorsque tous les routeurs disposent d'informations complètes et précises sur le réseau. Le temps de convergence est le temps nécessaire aux routeurs pour partager des informations, calculer les meilleurs chemins et mettre à jour leurs tables de routage. Un réseau n'est pas complètement opérationnel tant qu'il n'a pas convergé. La plupart des réseaux nécessitent un bref temps de convergence.

La convergence est à la fois collaborative et indépendante. Les routeurs partagent des informations les uns avec les autres, mais doivent calculer chacun de leur côté l'impact des modifications de la topologie sur leurs propres routes. Comme ils développent un accord avec la nouvelle topologie de manière indépendante, il est dit qu'ils convergent sur ce consensus.

Les propriétés de convergence incluent la vitesse de propagation des informations de routage et le calcul des chemins optimaux. Les protocoles de routage peuvent être classés en fonction de leur vitesse de convergence : une convergence rapide améliore un protocole de routage. Généralement, les protocoles RIP et IGRP mettent du temps à converger, alors que les protocoles EIGRP et OSPF sont plus rapides.

## *II.6. RIP*

RIP est considéré le premier IGP dans la famille TCP/IP, il a été largement utilisé dans les années 80 et plusieurs implémentations existaient mais la plus célèbre était celle implémenté dans UNIX BSD de Berkeley : le démon routed. Cette version basée sur les travaux de XEROX et leur protocole XNS RIP. RIP a été officiellement documenté en 1988 via la RFC1058. RIP est un IGP à vecteur de distance, c'est l'application directe de l'algorithme tel que nous l'avons présenté ci-avant.

### *II.6.1. Déroulement du protocole*

- Chaque routeur envoie périodiquement son vecteur encapsulé dans un datagramme UDP:520 à ces voisins, au début le vecteur de chaque routeur ne contient qu'une seule composante associée à lui-même avec la distance 0.
- Les routeurs écoutent en permanence les messages envoyés par ces voisins pour mettre à jour son vecteur.
- Les messages périodiques jouent deux rôles, la diffusion des mises à jour et le rafraîchissement.

- Si un routeur ne reçoit aucun rafraîchissement de son voisin pendant une période déterminée, il considère qu'il est déconnecté et met à jour son vecteur par des distances infinies pour les réseaux correspondants.
- A la réception d'un message de mise à jour le routeur procède par les étapes suivantes pour chaque entrée :
  - o Si la destination est inconnue elle sera ajoutée à la table de routage avec la distance correspondante.
  - o Sinon si la nouvelle route est plus courte que celle qui est installée alors elle prendra place de l'ancienne.
  - o Sinon si le saut suivant pour la destination de la route existante est identique à celui de la route de mise à jour alors la route sera mise à jour sinon cette entrée du message sera ignoré.

### II.6.2. format de message RIP

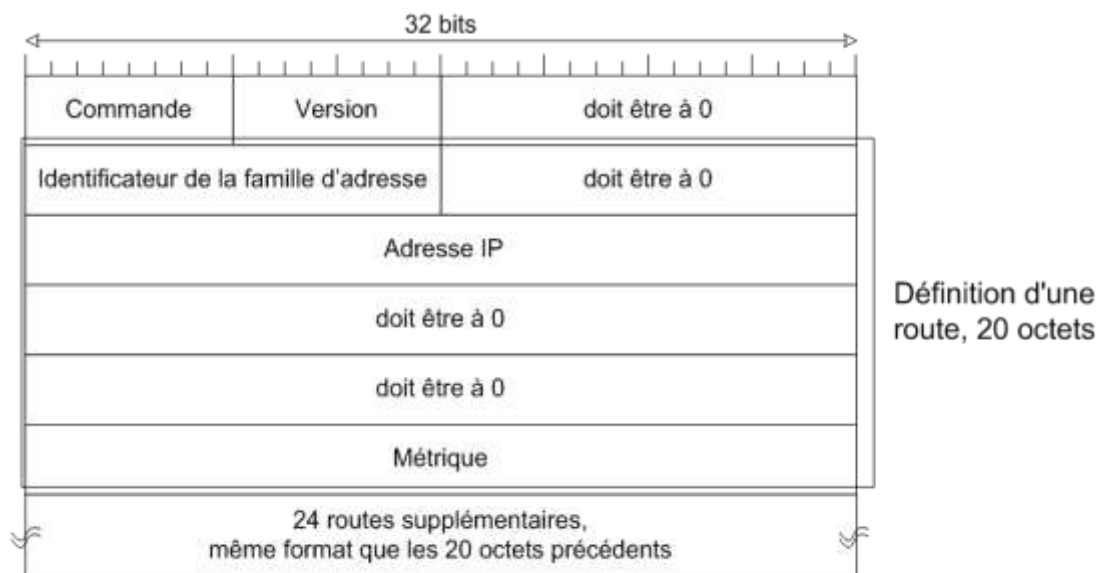


Figure 11: Message RIP

#### - *Commande*

Ce champ sert à préciser le contexte du message 1 – Request 2 – Response, comme vous pouvez remarquer RIP permet d'envoyer une requête concernant une ou plusieurs routes à l'aide de la commande 'Request', par contre la commande 'Response' est utilisée pour répondre à une requête comme pour les messages de mise à jour périodiques.

#### - *Version*

Indique la version du protocole et il doit être supérieur à 0, celle-ci est la version 1 de RIP, ce champ est utile pour des extensions futures.

- *Champs mis à 0*

Doit être à 0 si la version de RIP est 1 et ils seront ignorés si la version est supérieure à 1 donc ils peuvent contenir des informations nécessaires pour d'éventuelles améliorations.

- *AFI*

Indique la famille du protocole, 0x0002 est attribué à IP.

- *Le champ adresse*

Sert à identifier un réseau, une station ou un sous réseau selon le type de l'adresse, la conclusion est basée sur la classification standard des adresses IP.

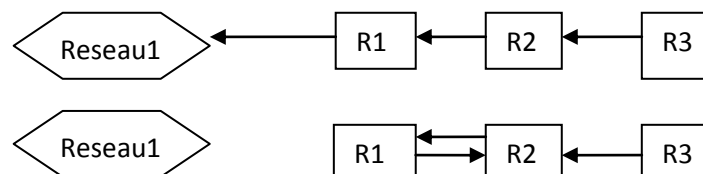
- *Métrique*

Le nombre de saut tout simplement, la taille du champ est 4 octets malgré que la valeur maximale permise est 16.

Une entrée dans le message de mise à jour de RIP est de taille 20 octets, le nombre total d'entrées est 25 ce qui donnera 500 octets en ajoutant les 4 octets de l'entête RIP plus les 8 octets de l'entête UDP nous aurons la taille maximale du datagramme UDP = 512.

### II.6.3. Le problème de convergence lente

Considérons l'ensemble de routeurs, qui illustre les routes jusqu'au réseau 1 existant dans l'internet.



Le routeur R1 accède directement au réseau 1, sa table comprend donc une route de distance de 1, incluse dans les messages régulièrement diffusés. R2 apprend la route de R1, l'installe dans sa propre table de routage, et l'annonce avec une distance de 3.

Supposons maintenant que la connexion de R1 au réseau 1 tombe en panne. R1 met immédiatement à jour sa table de routage et enregistre une distance de 16 (infinie). À l'occasion de diffusion suivante, R1 annonce la route de coût plus élevé. Cependant, un autre routeur peut diffuser ses routes avant R1, sauf si le protocole inclut des mécanismes supplémentaires permettant de l'empêcher. Ainsi supposons que R2 annonce des routes juste après la panne de la connexion avec R1. Dans ce cas, R1 reçoit le message de R2 et respecte l'algorithme à vecteur de distance habituel : il remarque que R2 annonce une route vers le réseau 1 dont le

cout est plus faible, il calcule qu'il lui faut désormais 3 sauts pour atteindre le réseau 1 (2 sauts de R2 au réseau 1, plus 1 saut de R1 à R2) ; et il installe une nouvelle route dont R2 est le saut suivant. A partir de cet instant, si l'un des routeurs R1 ou R2 reçoit un datagramme destiné au réseau 1, il le transmet vers l'autre routeur, qui lui renvoie, et ainsi de suite, jusqu'à expiration de la durée de vie du datagramme.

#### II.6.4. RIP et le problème du bouclage à l'infini 'Counting to infinity'

RIP souffre d'un problème majeure qui est du à son algorithme de base, ce problème est connu sous le nom du bouclage à l'infini. Ce problème résulte de la circulation de mises à jours qui ne sont plus valides et qui tournent en boucle avec l'accroissement de la distance jusqu'à l'infini.

#### II.6.5. Périodes RIP

##### - Update timer

Compteur des mises à jour périodiques égales à 30s.

##### - Time out

Compteur d'invalidation de routes, une route qui n'est pas rafraîchi pendant 180s sera déclarée invalide et donc une distance infinie est attribuée à la destination.

##### - Hold Down – Cisco –

Ce compteur est utilisée pour geler la réactivité aux mise à jours comme vu dans le Hold Down associé aux triggered updates, ce compteur est initialisé si une route est basculée vers la valeurs infinie.

##### - Garbage collector timer

Ou flush timer comme nommé par Cisco est un compteur initialisé à chaque fois qu'une route est invalidée, avant l'expiration de ce compteur la route continu à être annoncée avec une distance infinie jusqu'à l'expiration ou la route sera supprimé du vecteur par un programme nommé le ramasse miettes. Sa valeur est 120s mais Cisco utilise 240s.

#### II.6.6. La table de routage de RIP

DESTINATION	PASSERELLE	INTERFACE	MÉTRIQUE	TIMERS	FLAG

**Timers** contient les différents compteurs déjà vus et **flag** est un indicateur qui mentionne si une route a été récemment modifiée.

## *II.6.7. Limites du protocole*

### **II.6.7.1 Réseau de taille limitée**

Le problème majeur dont souffre RIP est le temps de convergence relativement élevé, ce temps dépend totalement de la complexité de la topologie ainsi que de la taille du réseau, si le réseau est très grand l'utilisation de cet algorithme est impossible et Cela revient aux caractéristiques de l'algorithme de base de ce protocole qu'est l'algorithme à vecteur de distance. A cause de ça, la spécification de RIP impose une limite sur le nombre maximal de sauts dans le réseau qu'est 15 et suggère que la valeur  $V=16$ .

### **II.6.7.2 Pas de support pour CIDR / VLSM**

Si vous regarder le format des messages de mise à jour de RIP vous trouverez qu'il ne comporte aucun champ pour le masque réseau, cela limite considérablement les capacités d'adressage et d'agrégation des préfixes réseaux. Les sous réseaux à masques variables ne sont pas supportés 'VLSM' et l'adressage CIDR aussi. Cela conduit à moins de flexibilité dans l'architecture du réseau et aussi à l'explosion de la taille de la table de routage si le nombre de préfixes sur le réseau est élevé.

### **II.6.7.3 Métrique faible**

La métrique utilisée dans RIP est simplement le nombre de saut en terme de routeurs entre la source et la destination, cette métrique ne reflète pas le coût réel du chemin car un chemin de longueur 2 peut être (c'est le cas souvent) beaucoup moins rapide qu'un chemin de 5 sauts.

### **II.6.7.4 Consommation des ressources du réseau**

Nous avons vu que RIP diffuse périodiquement (chaque 30s) le vecteur de tous les distances qu'il a appris et qui peut être de taille considérable, cette opération enregistre une consommation élevée de la bande passante dans les AS de taille moyenne ou élevée.

**Malgré tous ces inconvénients RIP constitue la solution idéale dans plusieurs situations. RIP est un protocole très facile à implémenter et si l'AS est de petite taille tous les inconvénients cités ci-dessus ne seront plus d'importance.**

## *II.6.8. Améliorations de RIPv2 par rapport à RIPv1*

Même si les principes évoqués ci-dessus sont valables quelle que soit la version de RIP, les différences restent intéressantes à relever. Les améliorations de RIPv2 sont :

- Diffusion des masques de sous-réseaux associés aux adresses réseaux (RIPv1 n'utilisait que les masques réseau par défaut).

- Utilisation d'une adresse de multicast pour diffuser les vecteurs de distance au lieu de l'adresses de broadcast ; ce qui réduit l'encombrement sur le réseau.
- Interopérabilité entre protocoles de routage en diffusant des routes apprises à partir d'autres protocoles.

L'ensemble de ces raisons rendent RIPv1 obsolète bien qu'il soit encore supporté par la plupart des routeurs logiciels ou matériels.

#### II.6.8.1 *format de message RIPv2*

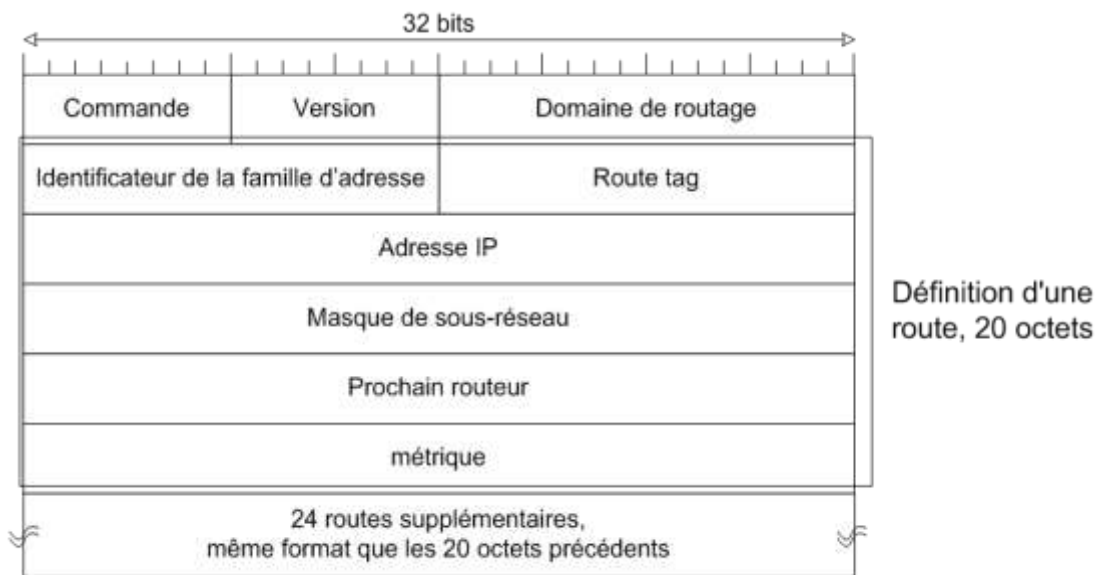


Figure 12: Message RIPv2.

RIPv2 est défini dans le document RFC 1723. À l'instar de la version 1, RIPv2 est encapsulé dans un segment UDP via le port 520 et peut transporter jusqu'à 25 routes. Bien que RIPv2 possède le même format de message de base que RIPv1, deux modifications majeures ont été apportées.

- La première modification apportée au format de message RIPv2 se situe au niveau du champ du masque de sous-réseau, qui permet d'inclure un masque 32 bits dans l'entrée de route RIP. En conséquence, le routeur récepteur ne dépend plus du masque de sous-réseau de l'interface entrante ou du masque par classe lors de la détermination du masque de sous-réseau d'une route.
- La deuxième modification significative apportée au format du message RIPv2 concerne l'ajout de l'adresse de tronçon suivant. L'adresse de tronçon suivant permet, le cas échéant, d'identifier une adresse de tronçon

suivant mieux adaptée que l'adresse du routeur émetteur. Si le champ contient uniquement des zéros (0.0.0.0), l'adresse du routeur émetteur constitue la meilleure adresse de tronçon suivant.

### II.6.8.2 Compatibilité avec RIP

RIPv2 est complètement compatible avec RIP en théorie, seulement les implémentations existantes de RIP ne sont pas conformes à la spécification de ce protocole qui est arrivé trop en retard par rapport aux implémentations existantes : le 'routed' de Berkeley distribué avec UNIX BSD depuis 1984. Normalement RIPv1 doit traiter tous les messages avec des versions supérieures de la même façon sauf qu'il doit ignorer les champs 'All os', de même RIPv2 supporte les messages de RIPv1 et prend seulement les champs que ce dernier peut fournir.

Enfin, RIPv2 n'est qu'un RIP et sauf les améliorations apporté et que nous avons cité, RIPv2 souffre des lui aussi des mêmes problèmes dont RIPv1 souffre et dus à leur algorithme de base.

	Protocoles de routage à vecteur de distance		Protocoles de routage à état de liaisons		Protocole BGP
Par classe	RIP	IGRP			EGP
Sans classe	RIPv2	EIGRP	OSPFv2	IS-IS	BGPv4
IPv6	RIPng	EIGRP for IPv6	OSPFv3	IS-IS for IPv6	BGPv4 for IPv6

Figure 13: Récapitulatif des protocoles de routage

## II.7. Critères d'évaluation des protocoles de routage

Plusieurs caractéristiques permettent de différencier les protocoles de routage :

### - Temps de convergence

Le temps de convergence définit la rapidité à laquelle les routeurs dans la topologie du réseau parviennent à partager les informations de routage et à disposer d'une base de connaissances cohérente. Plus la convergence est rapide, plus le protocole est recommandé. Des boucles de routage peuvent apparaître lorsque des tables de routage incohérentes ne sont pas mises à jour en raison d'une convergence plus lente dans un environnement réseau changeant.

### - *Évolutivité*

L'évolutivité définit la taille maximale d'un réseau en fonction du protocole de routage qui est déployé. Plus le réseau est grand, plus le protocole de routage doit être évolutif.

### - *Sans classe (utilisation d'un masque VLSM) ou par classe*

Les protocoles de routage sans classe incluent le masque de sous-réseau dans les mises à jour. Cette fonctionnalité prend en charge l'utilisation de masques de sous-réseau de longueur variable (VLSM) et permet un meilleur résumé des routes. Les protocoles de routage par classe n'incluent pas le masque de sous-réseau et ne peuvent pas prendre en charge les masques VLSM.

### - *Utilisation des ressources*

Inclut les exigences d'un protocole de routage telles que l'espace mémoire, l'utilisation du processeur et l'utilisation de la bande passante de liaison. Pour des besoins en ressources plus élevés, un matériel plus puissant est nécessaire pour prendre en charge le fonctionnement du protocole de routage en plus des processus de transfert de paquets.

### - *Implémentation et maintenance*

L'implémentation et la maintenance font référence aux connaissances qu'un administrateur réseau doit posséder pour implémenter et gérer le réseau en fonction du protocole de routage déployé.

	Vecteur de distance				État des liaisons	
	RIPv1	RIPv2	IGRP	EIGRP	OSPF	IS-IS
Vitesse de convergence	Lent	Lent	Lent	Rapide	Rapide	Rapide
Évolutivité : taille du réseau	Petit	Petit	Petit	Grand	Grand	Grand
Utilisation de VLSM	Non	Oui	Non	Oui	Oui	Oui
Utilisation des ressources	Bas	Bas	Bas	Moyen	Haut	Haut
Implémentation et maintenance	Simple	Simple	Simple	Complexe	Complexe	Complexe

Figure 14: Comparaison entre les protocoles de routages existants.

---

# Chapitre 03

## III. SIMULATION D'UN PROTOCOLE DE ROUTAGE À VECTEUR DE DISTANCE

---

Dans cette partie nous présentons une simulation que nous avons réalisé à l'aide des IPCs sous linux pour un protocole de routage à vecteur de distance. Cette simulation est construite sur deux composants : un gestionnaire du réseau qui permet de définir ou de modifier l'état du réseau. La deuxième composante est l'ensemble des routeurs. Un processus correspond à chaque routeur.

### *III.1. Environnement de travail*

#### *III.1.1. Plateforme*

Nous avons choisi comme système d'exploitation le système Gnu Linux. On utilise Linux au lieu d'un autre système d'exploitation parce que :

- C'est un système très bien documenté.
- Linux est Open source.
- Système multi tache et multi utilisateurs puissant et robuste.
- Il offre des mécanismes de communication entre processus appropriés.
- Il offre plusieurs bibliothèques gratuitement pour les programmeurs.

##### **III.1.1.1 Linux c'est quoi ?**

Linux est un système d'exploitation gratuit qui a été développé la première fois par un jeune étudiant Finlandais (Linus Torvalds). Lorsqu'il étudiait à l'université d'Helsinki en 1991, Torvalds a commencé le développement du noyau d'un clone d'Unix du zéro malgré qu'il travaille déjà avec le petit noyau MINIX développé par Andrew Tanenbaum (Un professeur Hollandais très connu dans le domaine) car ce dernier ne permettait pas la réutilisation de son code.

Après Torvalds a diffusé gratuitement le code source du noyau qu'il a écrit pour la communauté des développeurs sur internet pour qu'ils puissent contribuer. La sortie de ce noyau a pleinement profité d'un événement qui a précédé sa mise en place, car avant et en 1983, Richard Stallman (un développeur adepte du logiciel libre et un hacker) a fondé le Projet GNU qui visait de développer un système libre clone d'Unix mais complètement réécrit. Ils ont construit pas mal d'outil indispensable pour un vrai système d'exploitation jusqu'à 1990 (compilateurs, éditeurs...) mais le développement du noyau qui a démarré en 1990 et avant que Linus commence le sien a rencontré plusieurs difficultés c'est pour cela que les outils déjà développés ont été intégrés avec le noyau de Linus Torvalds. Cependant, le noyau de GNU survit toujours mais il n'est pas intégré dans aucune distribution. Quant au noyau linux il a connu un très grand progrès, il est stable et figure actuellement dans beaucoup de distributions commerciales ou gratuites.

Linux aujourd'hui est un système d'exploitation multi utilisateurs et multi tâches complet et incontournable, il est disponible pour la majorité des plateformes matérielles et offre une performance et une fiabilité très grande. Linux est de plus en plus utilisé par les professionnels mais on doit attendre un peu plus pour voir linux distribué comme système par défaut avec les PCs aux particuliers comme Windows.

### III.1.1.2 *Les distributions*

Linux est donc le noyau du système d'exploitation. Un système doit offrir aussi des services aux utilisateurs. Les distributions de Linux sont formées à partir du noyau et d'autres outils open source (principalement les outils GNU : compilateurs, éditeurs de textes, lecteurs multimédia, ...).

Les distributions les plus connues sont :



Nous avons travaillé avec la distribution Fedora et plus exactement la version Fedora 14.

### ***III.1.2. Programmation Orienté Objet en C++***

Pour la réalisation de notre projet, nous avons choisit comme langage de programmation le langage C++. Nous avons choisi ce langage pour deux raisons :

- Ce langage permet une vraie programmation orientée objet.
- En utilisant ce langage nous aurons l'accès à tous les bibliothèques écrites en C sous Linux et plus spécialement les bibliothèques de programmation système et de communication entre processus.

### ***III.1.3. Les IPCs***

Les IPCs, ou Inter Process Communication, apparus avec SystemV d'UNIX et repris dans linux sont des mécanismes systèmes de communication entre processus. Ils permettent d'échanger des données, de partager des données ou de synchroniser entre processus. ils sont constitués de 3 mécanisme :

- Les files de messages.
- Les segments de mémoire partagés.
- Les sémaphores.

#### ***- Les files de messages***

Ce mécanisme permet l'échange de messages par des processus. Chaque message possède un corps de longueur variable, et un type (entier strictement positif) qui peut servir à préciser la nature des informations contenues dans le corps. Au moment de la réception, on peut choisir de sélectionner les messages d'un type donné.

#### ***- Les mémoires partagés***

Ce mécanisme permet à plusieurs programmes de partager des segments mémoire. Chaque segment mémoire est identifié, au niveau du système, par une clé à laquelle correspond un identifiant. Lorsqu'un segment est attaché à un programme, les données qu'il contient sont accessibles en mémoire par l'intermédiaire d'un pointeur.

#### ***- Les sémaphores***

Un sémaphore est une variable protégée et constitue la méthode utilisée couramment pour restreindre l'accès à des ressources partagées dans un environnement de programmation concurrente bien qu'ils ne permettent pas d'éviter tous les interblocages (ou deadlocks). Sous GNU/Linux et les systèmes

d'exploitation compatibles SystemV, pour créer un sémaphore, l'utilisateur doit lui associer une clé. Le système lui renvoie un identifiant de sémaphore auquel sont attachés les  $n$  sémaphores numérotés de 0 à  $n - 1$ . Pour spécifier un sémaphore parmi les  $n$ , l'utilisateur indique l'identifiant et le numéro du sémaphore à chaque utilisateur de sémaphore sont associés des droits d'accès pour effectuer des opérations sur les sémaphores, mais seul le propriétaire peut réaliser les opérations de destruction d'un identifiant de sémaphore ou la modification des droits d'accès.

### III.2. Détails technique de notre application

Nous avons développé une application concurrente qui comprend :

- Un processus pour gérer le réseau et qui a pour tâche de définir le réseau simulé et de le modifier.
- Plusieurs processus pour simuler les routeurs, un processus pour chacun.

Le code est entièrement en Orienté objet et le diagramme suivant décrit les classes utilisées :

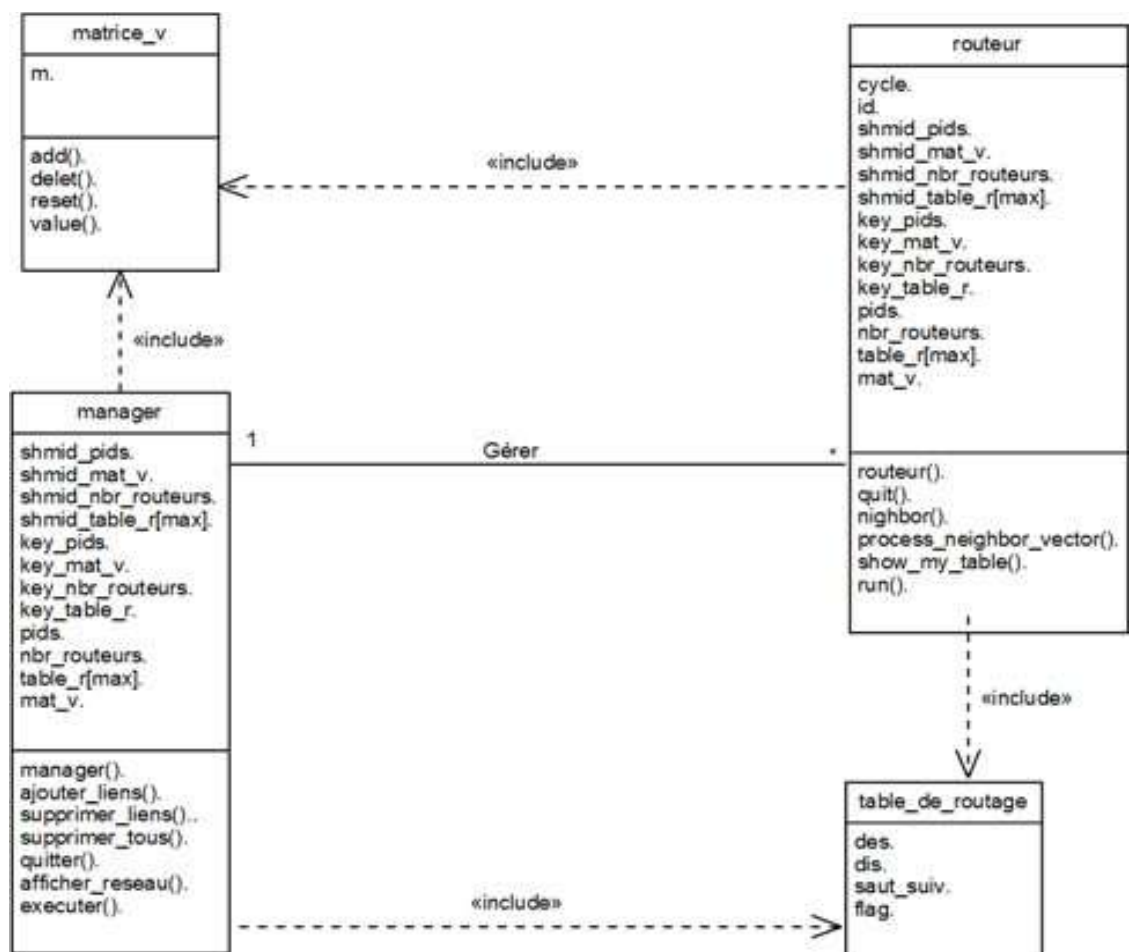


Figure 15: Diagramme de classes

### III.2.1. Les Timers

Pour bien observer le fonctionnement du protocole simulé nous avons utilisés des intervalles plus court dans l'implémentation :

- Le timeout : 9 secondes (au lieu de 180 secondes dans RIP).
- Le Update Timer : 3 secondes (au lieu de 30 secondes dans RIP).

### III.2.2. Les principales structures

```
class manager
{
public:
    int shmid_pids;
    int shmid_mat_v;
    int shmid_nbr_routeurs;
    int shmid_table_r[max];
    int key_pids;
    int key_mat_v;
    int key_nbr_routeurs;
    int key_table_r;
    int *pids;
    int *nbr_routeurs;
    table_de_routage *table_r[max];
    matrice_v *mat_v;
public:
    manager (int, int, int, int);
    void ajouter_liens(int,int);
    void supprimer_liens(int,int);
    void supprimer_tous();
    void quitter();
    void afficher_reseau();
    void afficher_tables_routage();
    void executer();
};
```

```
class routeur
{
public:
    int cycle;
    int id;
    int shmid_pids;
    int shmid_mat_v;
    int shmid_nbr_routeurs;
    int shmid_table_r[max];
    int key_pids;
    int key_mat_v;
    int key_nbr_routeurs;
    int key_table_r;
    int *pids;
    int *nbr_routeurs;
    table_de_routage *table_r[max];
    matrice_v *mat_v;
public:
    routeur (int, int, int, int);
    void quit ();
    bool neighbor (int);
    void process_neighbor_vector (int);
    void show_my_table ();
    void run ();
    void send(int);
    void receive(int);
};
```

### III.2.3. Dictionnaire de donnée :

Référence	Signification
<b>Id</b>	l'identificateur de routeur
<b>Shmid_pids</b>	la mémoire partagé réservé pour les pids .
<b>Shmid_mat_v</b>	la mémoire partagé réservé pour la matrice de voisinage.
<b>Shmid_nb_routeurs</b>	la mémoire partagé réservé pour le nombre de routeurs.
<b>Shmid_table_r[max]</b>	la mémoire partagé réservé pour la table de routage.
<b>Key_pids</b>	La clé de la mémoire partagé réservé pour les pids .
<b>Key_mat_v</b>	La clé de la mémoire partagé réservé pour la matrice de voisinage.
<b>Key_nbr_routeurs</b>	La clé de la mémoire partagé réservé pour le nombre de routeurs.
<b>Key_table_r</b>	La clé de la mémoire partagé réservé pour la table de routage.
<b>Pids</b>	Les Pids des routeurs.
<b>Nbr_routeurs</b>	Nombre de routeurs.
<b>Table_r[max]</b>	Table de routage.
<b>Mat_v</b>	Matrice de voisinage.
<b>Des</b>	Destination.
<b>Dis</b>	Distance.
<b>Saut_suiv</b>	Saut suivant.
<b>Flag</b>	Flag.
<b>m</b>	Matrice de voisinage.

## III.3. Fonctionnement générale

### III.3.1. Le manager

Ce processus prépare les structures de données nécessaires pour l'échange de messages entre les routeurs. Après il offre une interface à l'utilisateur qui lui permet de contrôler le réseau :

- Ajouter des liens.

- Supprimer des liens.
- Réinitialiser le réseau.
- Terminer la simulation.

### III.3.2. Les processus routeurs

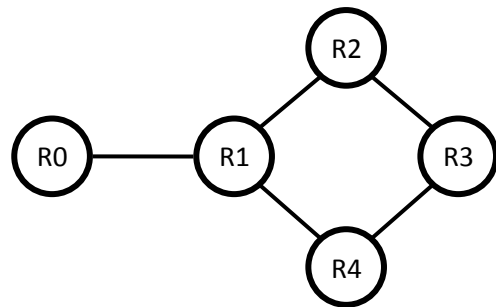
Un routeur commence par récupérer les structures de données qui lui seront utiles pour communiquer avec les autres processus routeurs avant de commencer sa vraie tâche. Cette dernière consiste à :

- Envoyer les vecteurs des distances aux voisins à l'aide des IPCs.
- Recevoir aussi les messages également de chez ses voisins.
- Traiter les vecteurs de distance reçus et mettre à jour sa table de routage.
- Afficher constamment sa table de routage.

### III.4. Exemple d'illustration n° 01

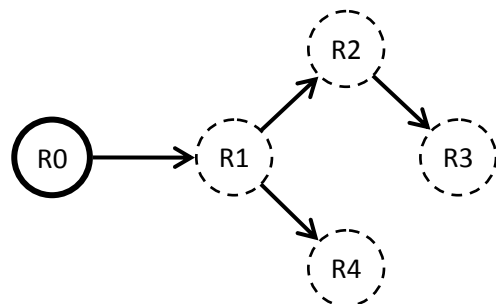
A l'aide du manager on définit le réseau suivant :

	R0	R1	R2	R3	R4
R0	0	1	0	0	0
R1	-	0	1	0	1
R2	-	-	0	1	0
R3	-	-	-	0	1
R4	-	-	-	-	0



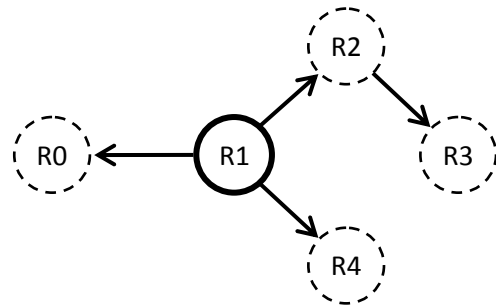
- *Routeur R0*

destination	distance	saut suiv
1	1	directe
2	2	1
3	3	1
4	2	1



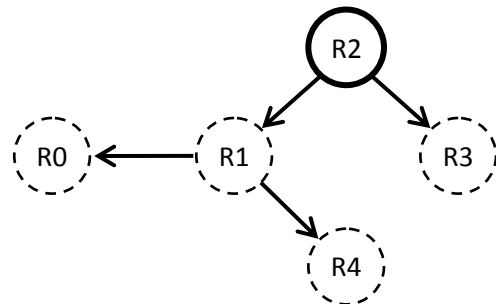
- *Routeur R1*

destination	distance	saut suiv
0	1	directe
2	1	directe
3	2	2
4	1	directe



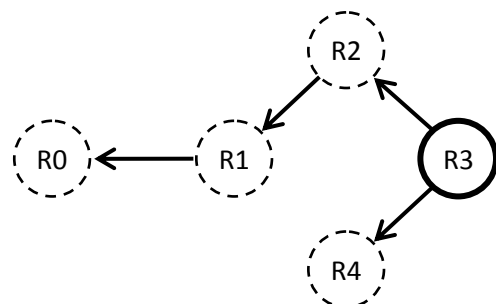
- *Routeur R2*

destination	distance	saut suiv
0	2	1
1	1	directe
3	1	directe
4	2	1



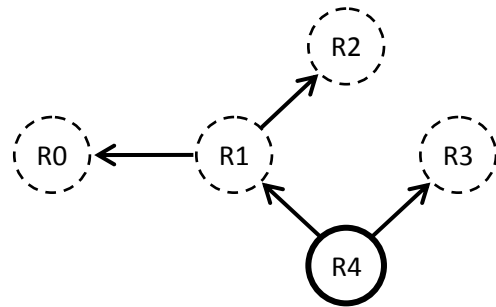
- *Routeur R3*

destination	distance	saut suiv
0	3	2
1	2	2
2	1	directe
4	1	directe



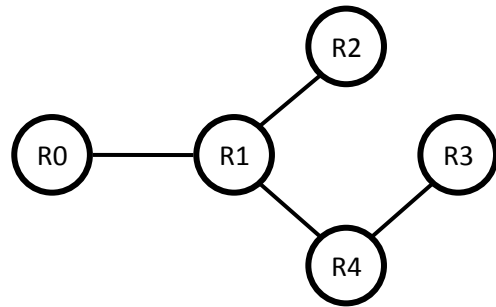
- *Routeur R4*

destination	distance	saut suiv
0	2	1
1	1	directe
2	2	1
3	1	directe



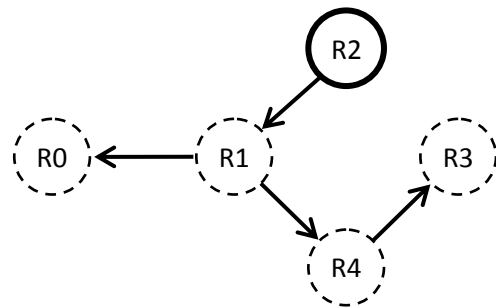
- *Suppression du lien entre R2 et R3*

	R0	R1	R2	R3	R4
R0	0	1	0	0	0
R1	-	0	1	0	1
R2	-	-	0	0	0
R3	-	-	-	0	1
R4	-	-	-	-	0



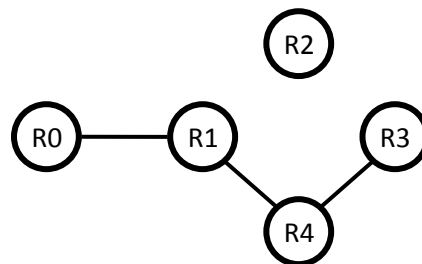
- *Nouvelles tables de routages pour R2*

destination	distance	saut suiv
0	2	1
1	1	directe
3	3	1
4	2	1

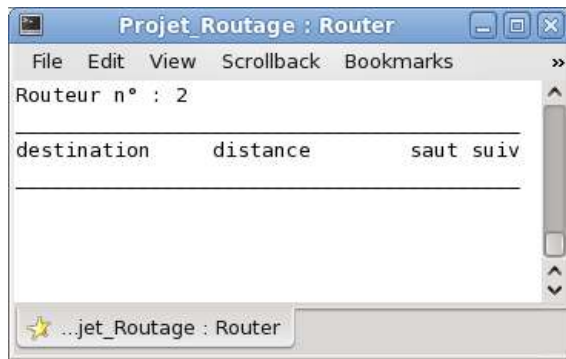


- *Suppression du lien entre R2 et R1*

	R0	R1	R2	R3	R4
R0	0	1	0	0	0
R1	-	0	0	0	1
R2	-	-	0	0	0
R3	-	-	-	0	1
R4	-	-	-	-	0



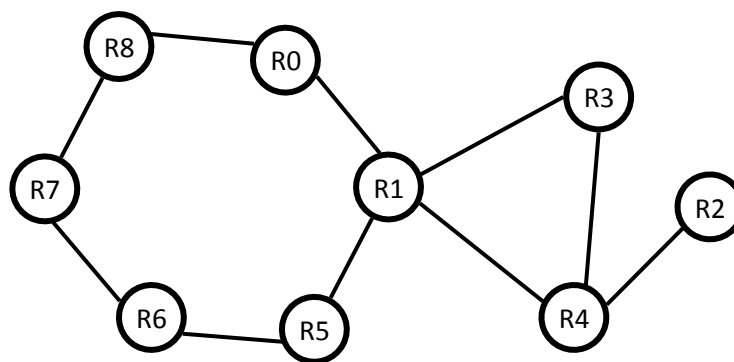
- *Routeur R2 isolé*



### III.5. Exemple d'illustration n° 02

On définit le nouveau réseau suivant à l'aide du manager:

	R0	R1	R2	R3	R4	R5	R6	R7	R8
R0	0	1	0	0	0	0	0	0	1
R1	-	0	0	1	1	1	0	0	0
R2	-	-	0	0	1	0	0	0	0
R3	-	-	-	0	1	0	0	0	0
R4	-	-	-	-	0	0	0	0	0
R5	-	-	-	-	-	0	1	0	0
R6	-	-	-	-	-	-	0	1	0
R7	-	-	-	-	-	-	-	0	1
R8	-	-	-	-	-	-	-	-	0



Projet\_Routage : Router

File Edit View Scrollback Bookmarks >>

Routeur n° : 0

destination	distance	saut suiv
1	1	directe
2	3	1
3	2	1
4	2	1
5	2	1
6	3	1
7	2	8
8	1	directe

☆ ...jet\_Routage : Router

Projet\_Routage : Router

File Edit View Scrollback Bookmarks >>

Routeur n° : 1

destination	distance	saut suiv
0	1	directe
2	2	4
3	1	directe
4	1	directe
5	1	directe
6	2	5
7	3	5
8	2	0

☆ ...jet\_Routage : Router

Projet\_Routage : Router

File Edit View Scrollback Bookmarks >>

Routeur n° : 2

destination	distance	saut suiv
0	3	4
1	2	4
3	2	4
4	1	directe
5	3	4
6	4	4
7	5	4
8	4	4

☆ ...jet\_Routage : Router

Projét\_Routage : Router

File Edit View Scrollback Bookmarks >>

Routeur n° : 3

destination	distance	saut suiv
0	2	1
1	1	directe
2	2	4
4	1	directe
5	2	1
6	3	1
7	4	1
8	3	1

☆ ...jet\_Routage : Router

Projét\_Routage : Router

File Edit View Scrollback Bookmarks >>

Routeur n° : 4

destination	distance	saut suiv
0	2	1
1	1	directe
2	1	directe
3	1	directe
5	2	1
6	3	1
7	4	1
8	3	1

☆ ...jet\_Routage : Router

Projét\_Routage : Router

File Edit View Scrollback Bookmarks >>

Routeur n° : 5

destination	distance	saut suiv
0	2	1
1	1	directe
2	3	1
3	2	1
4	2	1
6	1	directe
7	2	6
8	3	1

☆ ...jet\_Routage : Router

Projet\_Routage : Router

File Edit View Scrollback Bookmarks >>

Routeur n° : 6

destination	distance	saut suiv
0	3	5
1	2	5
2	4	5
3	3	5
4	3	5
5	1	directe
7	1	directe
8	2	7

☆ ...jet\_Routage : Router

Projet\_Routage : Router

File Edit View Scrollback Bookmarks >>

Routeur n° : 7

destination	distance	saut suiv
0	2	8
1	3	6
2	5	6
3	4	6
4	4	6
5	2	6
6	1	directe
8	1	directe

☆ ...jet\_Routage : Router

Projet\_Routage : Router

File Edit View Scrollback Bookmarks >>

Routeur n° : 8

destination	distance	saut suiv
0	1	directe
1	2	0
2	4	0
3	3	0
4	3	0
5	3	0
6	2	7
7	1	directe

☆ ...jet\_Routage : Router

- *Nouvelles tables de routage après la suppression du lien entre R0 et R1*

Projet\_Routage : Router

File Edit View Scrollback Bookmarks >>

Routeur n° : 0

destination	distance	saut suiv
1	5	8
2	7	8
3	6	8
4	6	8
5	4	8
6	3	8
7	2	8
8	1	directe

☆ ...jet\_Routage : Router

Projet\_Routage : Router

File Edit View Scrollback Bookmarks >>

Routeur n° : 1

destination	distance	saut suiv
0	5	5
2	2	4
3	1	directe
4	1	directe
5	1	directe
6	2	5
7	3	5
8	4	5

☆ ...jet\_Routage : Router

Projet\_Routage : Router

File Edit View Scrollback Bookmarks >>

Routeur n° : 2

destination	distance	saut suiv
0	7	4
1	2	4
3	2	4
4	1	directe
5	3	4
6	4	4
7	5	4
8	6	4

☆ ...jet\_Routage : Router

Projet\_Routage : Router

File Edit View Scrollback Bookmarks >>

Routeur n° : 3

destination	distance	saut suiv
0	6	1
1	1	directe
2	2	4
4	1	directe
5	2	1
6	3	1
7	4	1
8	5	1

...jet\_Routage : Router

Projet\_Routage : Router

File Edit View Scrollback Bookmarks >>

Routeur n° : 4

destination	distance	saut suiv
0	6	1
1	1	directe
2	1	directe
3	1	directe
5	2	1
6	3	1
7	4	1
8	5	1

...jet\_Routage : Router

Projet\_Routage : Router

File Edit View Scrollback Bookmarks >>

Routeur n° : 5

destination	distance	saut suiv
0	4	6
1	1	directe
2	3	1
3	2	1
4	2	1
6	1	directe
7	2	6
8	3	6

...jet\_Routage : Router

Projet\_Routage : Router

File Edit View Scrollback Bookmarks

Routeur n° : 6

destination	distance	saut suiv
0	3	7
1	2	5
2	4	5
3	3	5
4	3	5
5	1	directe
7	1	directe
8	2	7

Projet\_Routage : Router

Projet\_Routage : Router

File Edit View Scrollback Bookmarks

Routeur n° : 7

destination	distance	saut suiv
0	2	8
1	3	6
2	5	6
3	4	6
4	4	6
5	2	6
6	1	directe
8	1	directe

Projet\_Routage : Router

Projet\_Routage : Router

File Edit View Scrollback Bookmarks

Routeur n° : 8

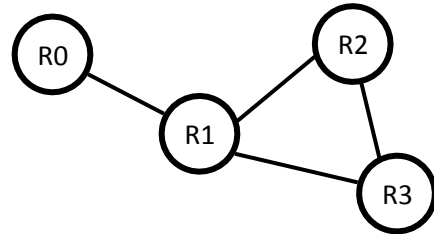
destination	distance	saut suiv
0	1	directe
1	4	7
2	6	7
3	5	7
4	5	7
5	3	7
6	2	7
7	1	directe

Projet\_Routage : Router

### III.6. Phénomène du comptage à l'infinie : Exemple

Soit maintenant le réseau suivant :

	R0	R1	R2	R3
R0	0	1	0	0
R1	-	0	1	1
R2	-	-	0	1
R3	-	-	-	0



Les tables de routage des quatre routeurs donnent :

Routeur n° : 0

destination	distance	saut	suiv
1	1		directe
2	2	1	
3	2	1	

Routeur n° : 1

destination	distance	saut	suiv
0	1		directe
2	1		directe
3	1		directe

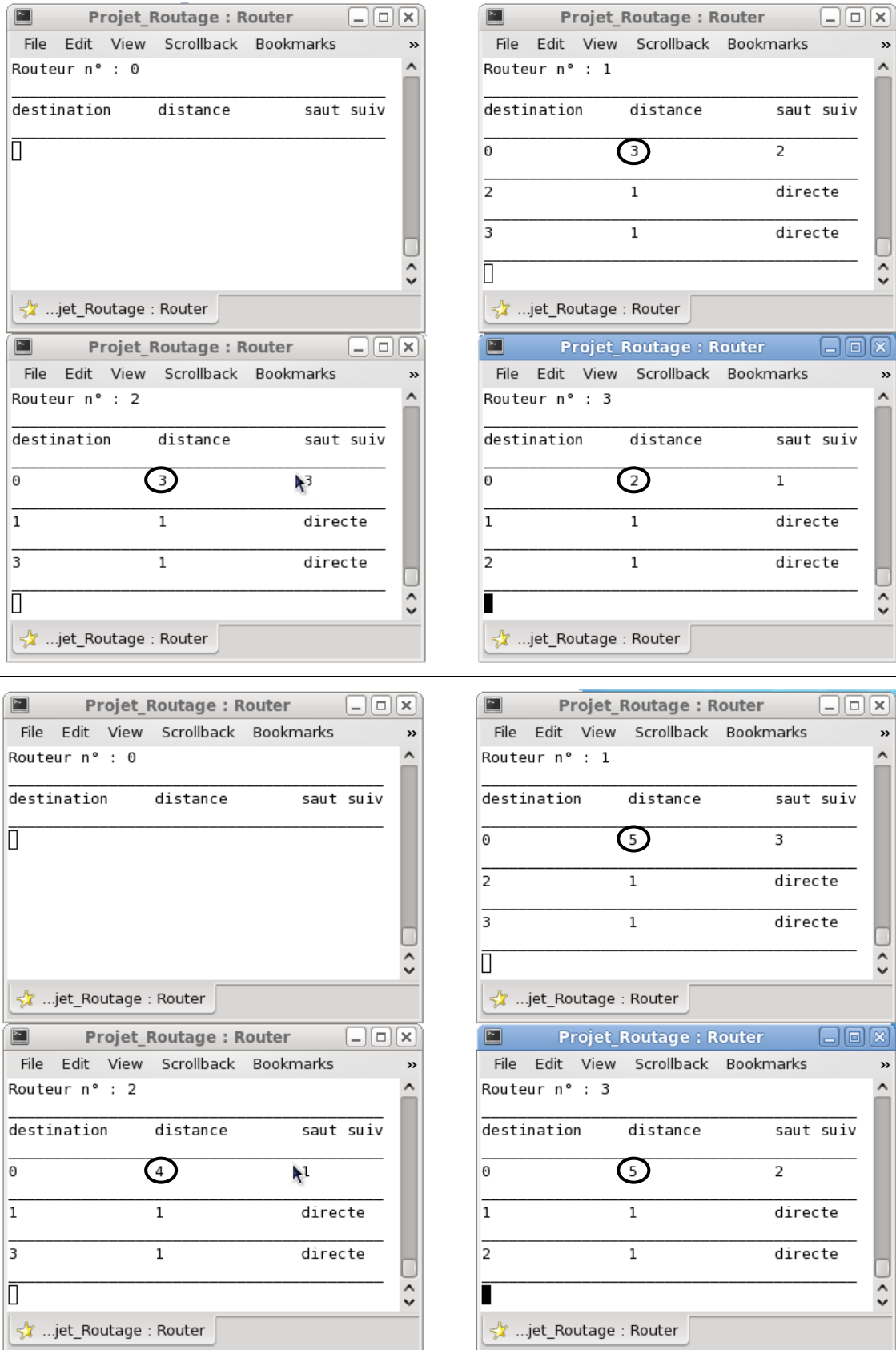
Routeur n° : 2

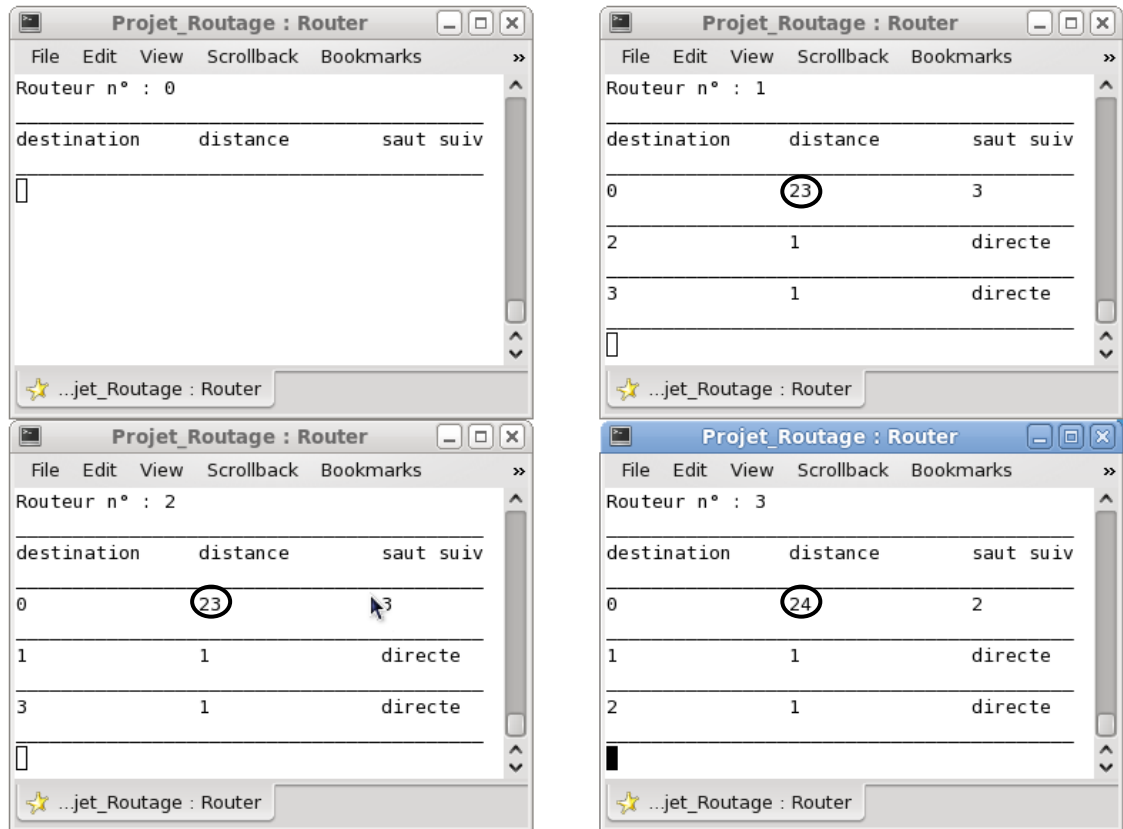
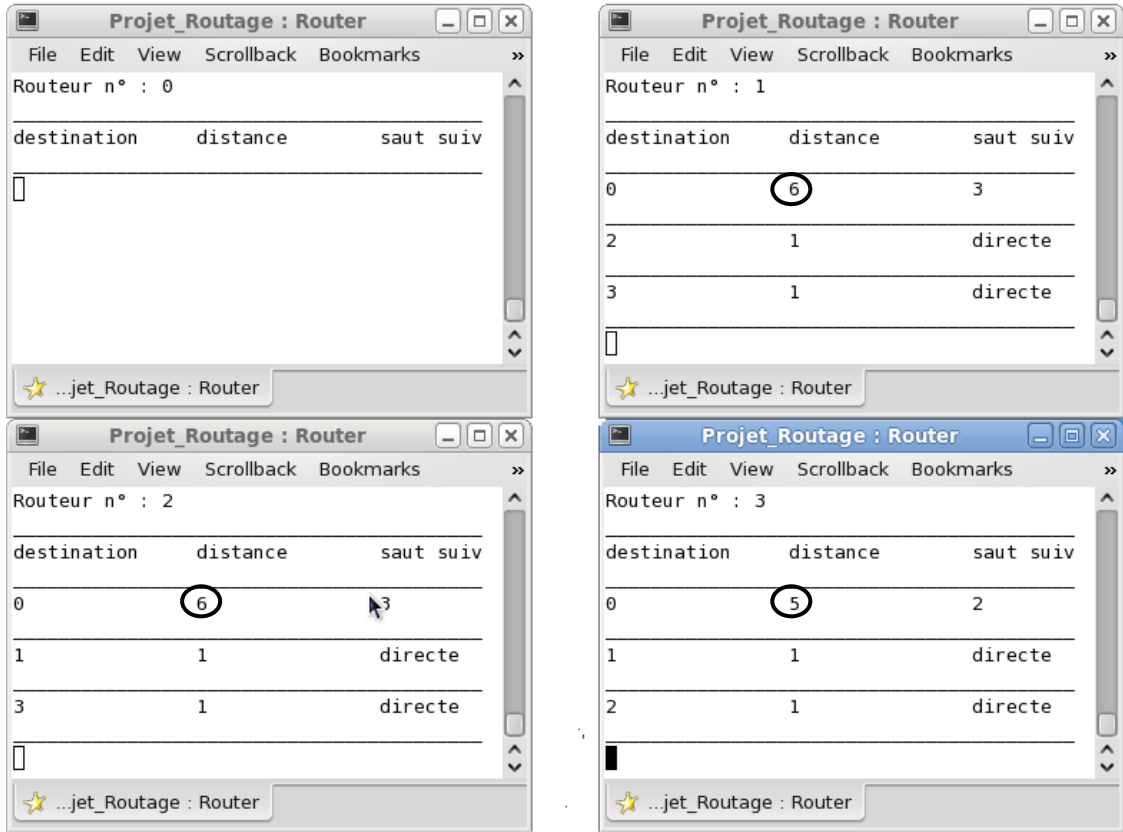
destination	distance	saut	suiv
0	2	1	
1	1		directe
3	1		directe

Routeur n° : 3

destination	distance	saut	suiv
0	2	1	
1	1		directe
2	1		directe

Si on supprime le lien entre R0 et R1 on distingue un comptage à l'infinie comme expliqué dans les inconvénients de RIP :





Le comptage à l'infini est une situation qui se produit lorsque des mises à jour de routage inexactes augmentent la valeur de la mesure jusqu'à l'infini pour un réseau qui n'est plus accessible. Les images ci-avant illustre comment les tables de

routage sont modifiées lorsque les trois routeurs continuent à s'envoyer des mises à jour inexactes.

Pour arrêter l'incrémentation d'une mesure, l'« infini » est défini par l'attribution d'une valeur maximale à la mesure. Par exemple, le protocole RIP considère que 16 sauts représentent l'infini, ce qui correspond à une mesure inaccessible. Une fois que les routeurs ont « compté jusqu'à l'infini », ils marquent la route comme étant inaccessible. Cette solution n'est pas la meilleure évidemment.

### *III.7. Conclusion*

Dans cette partie de notre mémoire nous avons montré une simulation du protocole RIP. Cette simulation montre que le protocole fonctionne correctement mais il peut être facilement exposé à une situation de comptage à l'infini.

# CONCLUSION GÉNÉRALE

A travers ce projet nous avons remarqué que les protocoles de routage à vecteur de distance sont relativement simples et pratiques pour des réseaux de basse complexité. Mais ils comportent aussi des inconvénients.

Par ce travail nous avons simulé un vrai protocole de routage avec tous ces éléments. Aussi nous avons simulé un travail de recherche. C'est-à-dire nous nous sommes mis dans une place qui nous a permis d'observer les inconvénients et d'en chercher les solutions possible. On espérait atteindre un meilleur stade mais malheureusement la durée du projet a été très courte. On voulait principalement chercher et tester une solution pour le phénomène du comptage à l'infinie. Aussi on espérait ajouter des interfaces graphiques plus conviviales à l'aide de la bibliothèque multi plateformes QT ou même de refaire toute l'implémentation sous Java...

Malgré ces limites nous retiendrons que nous avons fait une étude approfondie sur les réseaux et leur fonctionnement surtout concernant le routage. Nous avons développé une application concurrente très correcte et entièrement en orienté objet à l'aide du C++. Nous avons utilisé les mécanismes d'IPCs du système Linux pour réaliser les communications dans notre application et tout ça nous a été très bénéfique.

# RÉFÉRENCES BIBLIOGRAPHIQUES

- Bilal DOUAS, Les protocoles de routage Unicast dans l'Internet, ReSyD3, 2006.
- Andrew Tanenbaum, Réseau informatiques, 4e édition, 2007 .
- Daugles Caumer , TCP/IP.
- par Jean\_Michel Lébry, Algorithmique en C++, 2006 .
- C.Delannoy , Programmer en langage C++.
- Cisco Systems, Cisco Network Academy, CCNA1 , CCNA2.
- C. Hedrick, RFC1058, Routing Information Protocol, 1988.
- G. Malkin, RFC1723, RIPv2, 1994.
- F. Baker, RFC1812, Requirement of IP routers, 1995.
- RFC2328, OSPFv2.
- RFC3630, OSPF extension for traffic engineering – 2003.
- Carles Seguí, Evolution of unicast routing protocols in data networks.
- G. Griffin, Interdomain Routing Streams.
- Ravi Malhotra, IP Routing, 2002.
- Sam Halabi, Internet Routing Architectures, 2nd Edition, 2000.
- Jeff Doyle, Routing TCP/IP, Volume I, 1998.
- IBM, TCP / IP tutorial & technical overview, IBM Red Books – 2001.
- Chris Lewis, Cisco TCP/IP Routing Professional Reference, 1999.
- Cisco Systems, IP Routing fundamentals, 1998.