N° d'ordre : 07 / L3 / TCO Année Universitaire : 2013 / 2014

UNIVERSITE D'ANTANANARIVO

ECOLE SUPERIEURE POLYTECHNIQUE

DEPARTEMENT TELECOMMUNICATION

MEMOIRE DE FIN D'ETUDES

en vue de l'obtention

du DIPLOME de LICENCE

Mention : *Télécommunication*

Parcours : *Réseau et Système*

par : **RABEMIAFARA MAMIHARIMALALA Nantenaina**

MISE EN ŒUVRE DES TECHNOLOGIES DE TRANSITION IPv4 VERS IPv6

Soutenu le mercredi 06 Mars 2015 devant la Commission d'Examen composée de :

Président :

M. RATSIHOARANA Constant

Examinateurs :

M. RATSIMBAZAFY Andriamanga

Mme. RAMAFIARISONA Malalatiana

Mme. ANDRIANTSILAVO Haja Samiarivonjy

Directeur de mémoire :

M. RANDRIAMANAMPY Samuel

REMERCIEMENTS

Je remercie en premier lieu notre Seigneur, qui par sa grâce, a permis la réalisation de ce mémoire. Je tiens à exprimer toute ma reconnaissance à ceux, qui, de près ou de loin ont contribué à son élaboration. Aussi, je remercie respectueusement :

Monsieur ANDRIANARY Philippe Antoine, Professeur Titulaire, Directeur de l'Ecole Supérieure Polytechnique d'Antananarivo de m'avoir accueilli au sein de son établissement.

Monsieur RAKOTOMALALA Mamy Alain, Maître de conférences, Chef de Département Télécommunication, pour ma formation au sein de son département.

Monsieur RANDRIAMANAMPY Samuel, Enseignant au sein du Département Télécommunication, mon directeur de mémoire pour son aide et ses encouragements afin que ce travail soit en mesure des exigences.

Monsieur RATSIHOARANA Constant, Maître de conférences, Enseignant Chercheur à l'ESPA, d'avoir fait l'honneur de présider le Jury de ce mémoire.

Tous les membres du Jury, à savoir :

Monsieur RATSIMBAZAFY Andriamanga, Maître de Conférences, Enseignant Chercheur à l'ESPA ;

Madame RAMAFIARISONA Malalatiana, Maître de Conférences, Enseignant Chercheur à l'ESPA;

Madame ANDRIANTSILAVO Haja Samiarivonjy, Enseignant Chercheur à l'ESPA.

Qui ont eu l'amabilité d'examiner ce mémoire malgré leurs nombreuses occupations.

Mes vifs remerciements à tous les enseignants et les personnels administratifs de l'ESPA.

Ma profonde gratitude s'adresse également à :

La société ORANGE Madagascar, au département technique et informatique, pour leur accueil et leur conseil pendant la réalisation de ce mémoire.

Je n'oublierai pas ma famille et mes amis pour leurs soutiens bienveillants et leurs encouragements, durant l'élaboration de ce mémoire, comme en toutes circonstances.

TABLE DES MATIERES

LISTE DES ABREVIATIONS

AfriNIC	African Network Information Center
APNIC	Asia Pacific Network Information Center
ARIN	American Registry for Internet Numbers
ARP	Address Resolution Protocol
ARPANET	Advanced Research Projects Agency Network
AS	Autonomous System
BGP	Border Gateway Protocol
CE	Customer Edge router
DF	Don’t Fragment
DHCP	Dynamic Host Configuration Protocol
DHCPv6	Dynamic Host Configuration Protocol Version 6
DNS	Domain Name System
EGP	Exterior Gateway Protocols
EIGRP	Enhanced Interior Gateway Routing Protocol
ESPA	Ecole Supérieure Polytechnique d’Antananarivo
FAI	Fournisseur d’Accès Internet
FP	Format Prefix
FTP	File Transfert Protocol
FTPd	File Transfert Protocol demon
GNS3	Graphical Network Simulator 3rd version
HLen	Header length
HTTP	Hyper Text Transport Protocol.
IAB	Internet Architecture Board
IANA	Internet Assigned Numbers Authority
ICANN	Internet Corporation Assigned Names and Numbers
ICMP	Internet Control Message Protocol.
IEEE	Institute of Electrical and Electronic Engineers
IETF	Internet Engineering Task Force
IGMP	Internet Group Management Protocol.
IGP	Interior Gateway Protocols

IGRP	Interior Gateway Routing Protocol
IOS	Interface Operating System
IP	Internet Protocol
IPng	Internet Protocol Next Generation
IPsec	Internet Protocol Security
IPv4	Internet Protocol Version 4
IPv6	Internet Protocol Version 6
ISATAP	Intra-Site Automatic Tunnel Addressing Protocol
IS-IS	Intermediate System-to-Intermediate System
ISO	International Standard Organization
ISOC	Internet Society
LACNIC	Latin American and Caribbean IP address Regional Registry
LAN	Local Area Network
LDP	Label Distribution Protocol
LIR	Local Internet Registries
LSP	Label Switched Path
MAC	Media Access Control.
MAN	Metropolitan Area Network
MBGP	Multicast BGP
MF	More Fragment
MLD	Multicast Listener Discovery
MP-BGP	Multi Protocol Border Gateway Protocol
MPLS	Multi Protocol Label Switching
MTU	Maximum Transmission Unit
NAT	Network Address Translation
NAT	Network Address Translation
NAT-PT	Network Address Translation – Protocol Translation
ND	Neighbor Discovery
NLA	Next-Level Aggregation
NRO	Number Resources Organization
OS	Operating System
OSI	Open System Interconnection.

OSPF	Open Shortest Path First
OSPFv2	Open Shortest Path First Version 2
OSPFv3	Open Shortest Path First Version 3
P2P	Peer to peer
PC	Personal Computer
PE	Provider Edge router
POP	Post Office Protocol
PPP	Point to Point Protocol
PTR	Pointer Records
RFC	Request For Command
RIP	Routing Information Protocol
RIPE	Réseaux IP Européen
RIPng	Routing Information Protocol Next Generation
RIR	Regional Internet Registries
RSVP	Resources ReSerVation Protocol
SMTP	Simple Mail Transfert Protocol
TCP	Transmission Control Protocol
TDP	Tag Distribution Protocol
TELNET	TeleNetwork
TFTP	Trivial File Transfert Protocol
TLA	Top Level Aggregation
TTL	Time to live
UDP	User Datagram Protocol.
UIT-T	Union Internationale des Télécommunications – Télécommunications
VOIP	Voice Over IP
VPN	Virtual Private Network

INTRODUCTION GENERALE

Le réseau Internet actuel repose sur le protocole IP, précisément pour Internet Protocol, dans sa version 4 ; conçu au temps où l'informatique communicante n'en était qu'à ses débuts, ses quatre milliards d'adresses disponibles paraissaient alors plus qu'amplement suffisants. Or, nous sommes passés du temps de l'abondance à celui de la disette, et, avec la multiplication des FAI (Fournisseur d'Accès Internet) et des ordinateurs personnels, il devient de plus en plus difficile de se faire allouer, par les organismes compétents, des segments conséquents d'adresses contiguës. Les allocations se faisant désormais au coup par coup, il s'en suit une importante fragmentation de l'espace d'adressage, ce qui complique énormément la tâche des routeurs, incapables de s'appuyer sur un mécanisme simple d'aiguillage des paquets.

Pour pallier cette pénurie, les ingénieurs ont développé un ensemble de subterfuges, dont le fameux NAT pour Network Address Translation, qui permet à un routeur frontal de découpler des réseaux. Ainsi, même si le réseau interne compte un millier de machines, une seule adresse suffit à les atteindre toutes. L'autre palliatif utilisé par les FAI consiste à recourir à l'adressage dynamique qui revient à allouer une adresse uniquement sur demande.

Cependant, cela ne suffit pas. D'une part, de plus en plus de FAI proposent à leurs clients une adresse IP fixe ; d'autre part, le nombre de terminaux susceptibles de se connecter simultanément au réseau est en augmentation exponentielle. La véritable parade à l'épuisement des adresses se trouve dans l'adoption de la version plus évoluée d'IP, baptisée IPv6, qui propose un nombre exorbitant d'adresses. IPv6 est conçu pour s'affranchir des limitations d'IPv4, mais aussi pour prendre en compte les avancées issues des recherches sur les réseaux, comme l'autoconfiguration, le multicast ou encore la sécurité. L'objectif à terme est de remplacer tous les systèmes IPv4 par des systèmes IPv6 mais une phase de transition sera indispensable pour les faire coexister et pour migrer finalement en une infrastructure tout IPv6.

Ce sujet de mémoire trouve pleinement sa place au cœur de cette problématique. Il explorera en profondeur le protocole IPv6 et mettra en œuvre les technologies de transition IPv4 vers IPv6, d'où l'intitulé « Mise en œuvre des technologies de transition IPv4 vers IPv6 ». Pour ce faire, cet ouvrage sera divisé en quatre chapitres :

Le premier chapitre abordera les réseaux IP, plus particulièrement, l'architecture TCP/IP et le protocole IPv4.

Le deuxième chapitre traitera en profondeur le protocole IPv6.

Le troisième chapitre sera consacré à la présentation des mécanismes de transition en IPv6.

Et enfin, en dernier chapitre, la mise en œuvre des technologies de transition IPv4 vers IPv6 sous GNS3.

CHAPITRE 1
RESEAUX IP

1.1 Introduction

TCP/IP désigne communément une architecture réseau, mais cet acronyme désigne en fait 2 protocoles étroitement liés : un protocole de transport, TCP (Transmission Control Protocol) qu'on utilise "par-dessus" un protocole réseau, IP (Internet Protocol).

Ce qu'on entend par "modèle TCP/IP", c'est en fait une architecture réseau en 4 couches dans laquelle les protocoles TCP et IP jouent un rôle prédominant, car ils en constituent l'implémentation la plus courante.

1.2 Description du modèle

1.2.1 Modèle TCP/IP

Le modèle TCP/IP peut en effet être décrit comme une architecture réseau à 4 couches :

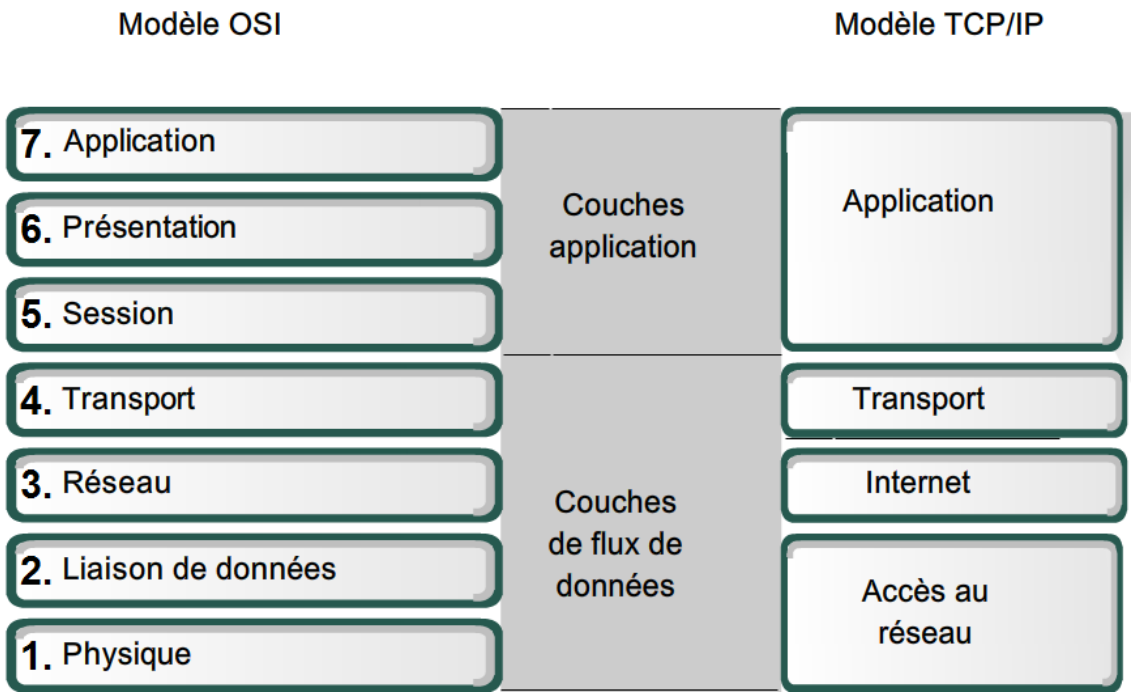

Figure 1.01 : *Le modèle TCP/IP et le modèle OSI*

Le modèle OSI a été mis à côté pour faciliter la comparaison entre les deux modèles. Il y a 4 couches principales dans l'environnement TCP/IP :

1.2.1.1 Couche application

Les applications interagissent avec les protocoles de la couche transport pour envoyer ou recevoir des données. Elle représente donc des données pour l'utilisateur ainsi que du codage et un contrôle de dialogue. Comme application on peut trouver HTTP, POP, FTP, SMTP, etc. [2]

1.2.1.2 Couche Transport

Elle est chargée de fournir un moyen de communication de bout en bout entre 2 programmes d'application. Elle agit en mode connecté et en mode non connecté. Elle divise le flux de données venant des applications en paquets, transmis avec l'adresse destination IP au niveau IP. Comme protocole de la couche transport on peut trouver UDP et TCP. [2]

1.2.1.3 Couche Internet

Elle encapsule les données reçues de la couche transport dans des datagrammes IP et elle détermine le chemin le meilleur chemin à travers le réseau. Mode non connecté et non fiable. [2]

1.2.1.4 Couche Accès réseau

Cette couche assure la transmission d'un datagramme venant de la couche IP en l'encapsulant dans une trame physique et en transmettant cette dernière sur un réseau physique. [2]

1.2.2 Protocoles de la couche application

Les protocoles d'application sont des protocoles de haut niveau, adaptés aux besoins d'applications spécifiques. Ils s'appuient sur UDP et TCP pour permettre le transfert d'informations entre une application serveur et ses applications clientes. [1]

1.2.2.1 Protocole HTTP (Hyper Text Transfert Protocol)

Ce protocole est utilisé pour la navigation web entre un serveur HTTP et un PC client. Le protocole HTTP est utilisé à travers le Web pour le transfert des données et constitue l'un des protocoles d'application les plus utilisés. Il constitue un protocole de requête/réponse. Lorsqu'un client (généralement un navigateur Web) envoie une requête à un serveur, le protocole HTTP définit les types de messages que le client utilise pour demander la page Web, ainsi que les types de messages que le serveur utilise pour répondre. Les trois types de messages courants sont GET, POST et PUT.

1.2.2.2 Protocole FTP (File Transfert Protocol)

Le protocole FTP (File Transfer Protocol) est un autre protocole de couche application couramment utilisé. Il a été développé pour permettre le transfert de fichiers entre un client et un serveur. Un client FTP est une application s'exécutant sur un ordinateur et utilisée pour extraire des fichiers d'un serveur exécutant le démon FTP (FTPd). C'est un protocole qui permet d'assurer le transfert de fichiers de façon indépendante des spécificités des OS (Operating System). Ainsi, un client FTP sous Windows peut télécharger un fichier depuis un serveur UNIX. [1] [2]

1.2.2.3 Protocole SMTP (Simple Mail Transfert Protocol)

Le protocole qui permet d'acheminer le courrier depuis le serveur SMTP de l'émetteur, jusqu'au serveur SMTP du destinataire, qui le classe dans les boîtes aux lettres de ses clients.

1.2.2.4 Protocole POP (Post Office Protocol)

Les protocoles POP et POP3 (Post Office Protocol, version 3) sont des protocoles de remise du courrier entrant et constituent des protocoles client/serveur standards. Ils transmettent le courriel du serveur de messagerie au client de messagerie. C'est le protocole qui permet au client de relever à distance le courrier classé dans sa boîte aux lettres. [2]

1.2.2.5 Protocole TELNET (Tele Network)

Telnet offre une méthode standard permettant d'émuler les périphériques terminaux texte via le réseau de données. Logiquement, une connexion qui utilise Telnet est nommée connexion ou session VTY (Virtual Terminal). Telnet est un protocole client/serveur qui spécifie la manière dont une session VTY s'établit et prend fin. Il fournit également la syntaxe et l'ordre des commandes qui permettent d'ouvrir une session Telnet, ainsi que les commandes de contrôle exécutables pendant une session [2]. En fait, un client TELNET est une console en mode texte, capable de se connecter sur la plupart des serveurs, comme POP3 ou SMTP. Il devient alors possible d'envoyer et de lire des messages, si l'on connaît les commandes inhérentes aux protocoles SMTP et POP3. Un serveur TELNET permet cependant des choses bien plus puissantes et "dangereuses" puisqu'il devient possible de prendre à distance le contrôle d'un hôte. C'est un outil qui permet l'administration distante d'une machine, du moment que l'on est capable d'ouvrir une session et d'acquérir les droits de "super utilisateur".

1.2.3 Protocoles de la couche transport

1.2.3.1 Protocole UDP (User Datagram Protocol)

Le protocole UDP est un protocole simple offrant des fonctions de couche transport de base. Il crée beaucoup moins de surcharge que le protocole TCP car il est sans connexion et ne propose pas de mécanismes sophistiqués de retransmission, de séquençage et de contrôle de flux. Il n'ouvre pas de session et n'effectue pas de contrôle d'erreur. Il est alors appelé "mode non connecté". Il est donc peu fiable, cependant, il permet aux applications d'accéder directement à un service de transmission de datagrammes rapide. [1] [2]

Mais cela ne signifie pas que les applications utilisant le protocole UDP manquent toujours de fiabilité. Cela signifie simplement que ces fonctions ne sont pas fournies par le protocole de la couche transport et qu'elles doivent être implémentées à un autre niveau, le cas échéant.

Bien que le volume total de trafic UDP trouvé sur un réseau typique soit relativement faible, des protocoles importants de la couche application utilisent le protocole UDP, notamment :

- TFTP (Trivial File Transfer Protocol)
- Lecture vidéo en continu
- Voix sur IP (VoIP)
- Jeux en ligne, etc.

Une application est donc identifiée sur le réseau par :

- L'adresse IP de la station sur laquelle elle se trouve.
- Le protocole TCP ou UDP.
- Le port number auquel elle s'est raccordée.

Cette connexion logique entre deux ports est appelée : Socket. UDP est un protocole de transport utilisant directement IP ce qui entraîne qu'il offre un service de transport :

- Non fiable (sans acquittement).
- Sans connexion.
- Sans contrôle de flux.

C'est aux applications de prendre en charge l'acquittement, la connexion et la remise dans l'ordre des messages.

Figure 1.02 : *Datagramme UDP*

1.2.3.2 Protocole TCP (Transfert Control Protocol)

La fiabilité est le principal élément différentiateur entre les protocoles TCP et UDP.

Le protocole TCP est un protocole avec connexion décrit dans le document RFC 793. Le protocole TCP impose une surcharge pour accroître les fonctionnalités. Le protocole TCP spécifie d'autres fonctions, à savoir la livraison dans l'ordre, l'acheminement fiable et le contrôle du flux. Chaque segment du protocole TCP utilise 20 octets de surcharge dans l'en-tête pour encapsuler les données de la couche application alors que chaque segment du protocole UDP n'ajoute que 8 octets de surcharge. Il ouvre une session et effectue lui-même le contrôle d'erreurs. Il est alors appelé "mode connecté". [1] [2]

TCP fournit un service :

- Fiable (canal sans erreurs).
- Avec contrôle de flux.
- Ordonné.
- En mode full duplex.
- En mode connecté.

Le protocole TCP est utilisé par des applications de :

- Navigateurs Web
- Courriel
- Transfert de fichiers

TCP tout comme UDP utilise la notion de port excepté que TCP utilise la connexion comme abstraction de port. Une connexion est identifiée par une paire de « End points » : Host (adresse IP d'une station) et Port (port TCP).

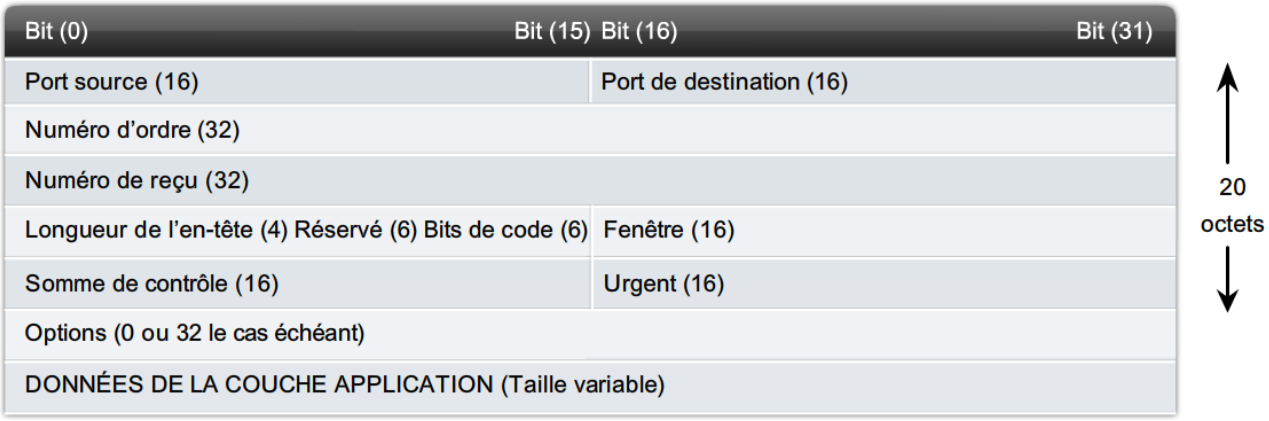

Figure 1.03 : *Segment TCP*

1.2.4 Protocoles de la couche Internet

1.2.4.1 Le protocole IP

IP signifie "Internet Protocol", protocole Internet. Il représente le protocole réseau le plus répandu et la version actuelle la plus utilisée est la version 4 (IPv4). Dans ce premier chapitre, le protocole IP désigne communément le protocole IPv4. Il permet de découper l'information à transmettre en paquets, de les adresser, de les transporter indépendamment les uns des autres et de recomposer le message initial à l'arrivée. Ce protocole utilise ainsi une technique dite de commutation de paquets.

Au niveau IP, les données des utilisateurs ou des applications sont encapsulées à l'intérieur d'unités de transfert appelées datagrammes IP. Le protocole IP fournit un service d'acheminement des datagrammes IP sans connexion et non fiable. [2]

Un datagramme se compose d'un en-tête et de données. Avant transmission sur un réseau physique, le datagramme IP est encapsulé dans une trame physique.

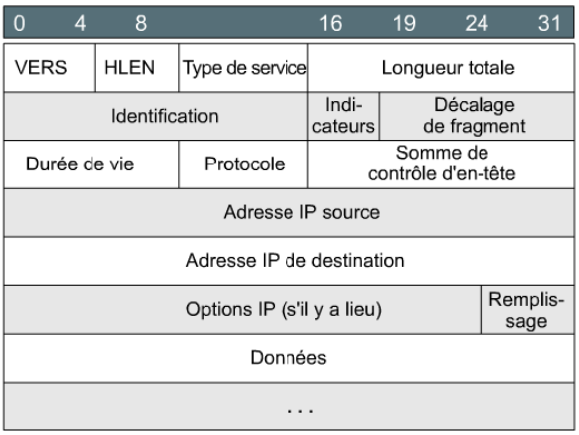

Figure 1.04 : *L'en-tête IPv4*

– *Vers*

Le champ version est codé sur 4 bits. Il représente le numéro de version du protocole IP. [1]

– *HLen*

HLen signifie "Internet header lenght". Ce champ est codé sur 4 bits et représente la longueur en mots de 32 bits de l'entête IP. Par défaut, il est égal à 5 (20 octets), cependant, avec les options de l'entête IP, il peut être compris entre 6 et 15. Le fait que le codage soit sur 4 bits, la taille maximum de l'entête IP est donc de 15*32bits = 60 octets. [1]

– *Service*

Le champ service "Type Of Service" est codé sur 8 bits, il permet la gestion d'une qualité de service traitée directement en couche 3 du modèle OSI. [1]

– *Longueur totale*

Le champ Longueur totale est codé sur 16 bits et représente la longueur du paquet incluant l'en-tête IP et les data associées. [1]

– *Identification*

Le champ Identification est codé sur 16 bits et constitue l'identification utilisée pour reconstituer les différents fragments. Chaque fragment possède le même numéro d'identification, les entêtes IP des fragments sont identiques à l'exception des champs Longueur totale, Checksum et Position fragment. [1]

– *Indicateurs*

Le champ *Indicateurs* est codé sur 3 bits et représente des indicateurs de contrôle, comme DF et MF. [1]

DF ou Don’t Fragment (1 bit) : Si ce flag est à 1, cela signifie que le datagramme ne doit pas être fragmenté. [2]

MF ou More Fragment (1 bit) : Si ce flag est à 1, le destinataire est informé que d’autre fragment vont suivre. Quand MF est à 0, cela indique qu’il s’agit du dernier fragment. [2]

– *Décalage de fragment*

Un routeur peut devoir fragmenter un paquet lors de sa transmission d’un média à un autre de MTU inférieure. Lorsqu’une fragmentation se produit, le paquet IPv4 utilise le champ de décalage du fragment et l’indicateur MF de l’en-tête IP pour reconstruire le paquet à son arrivée sur l’hôte de

destination. Le champ de décalage du fragment identifie l'ordre dans lequel placer le fragment de paquet dans la reconstruction. [1]

– *TTL*

Le champ TTL (Time To Live) est codé sur 8 bits et indique la durée de vie maximale du paquet. Il représente la durée de vie en seconde du paquet. Si le TTL arrive à 0, alors l'équipement qui possède le paquet, le détruira. À chaque passage d'un routeur le paquet se verra décrémenté d'une seconde. De plus, si le paquet reste en file d'attente d'un routeur plus d'une seconde, alors la décrémentation sera plus élevée. Elle sera égale au nombre de seconde passé dans cette même file d'attente. Par défaut, si les temps de réponse sont corrects, alors on peut, entre guillemet, en conclure que le Time To Live représente le nombre de saut maximum du niveau. Le but du champ TTL est d'éviter de faire circuler des trames en boucle infinie. [2]

– *Protocole*

Le champ Protocole est codé sur 8 bits et représente le type de Data qui se trouve derrière l'entête.

– *Checksum*

Le champ Checksum est codé sur 16 bits et représente la validité du paquet de la couche 3. Pour pouvoir calculer le Checksum, il faut positionner le champ du checksum à 0 et ne considérer que l'entête IP. Donc par exemple, si deux trames ont la même entête IP (y compris le champ length) et deux entêtes ICMP et Data différentes (mais de même longueur), le checksum IP sera alors le même.

– *Adresse IP source*

Le champ IP source est codé sur 32 bits et représente l'adresse IP source ou de réponse. Il est codé sur 4 octets qui forment l'adresse A.B.C.D.

– *Adresse IP destination*

Le champ IP destination est codé sur 32 bits et représente l'adresse IP destination. Il est codé sur 4 octets qui forment l'adresse A.B.C.D.

1.2.4.2 Le protocole ARP (Address Resolution Protocol)

Le protocole ARP, signifiant Address Resolution Protocol, fonctionne en couche Internet du modèle TCP/IP correspondant à la couche 3 du modèle OSI. L'objectif de ARP est de permettre de résoudre une adresse physique par l'intermédiaire de l'adresse IP correspondante d'un host distant. Le protocole ARP apporte un mécanisme de « translation » pour résoudre ce besoin. Il permet d'obtenir l'adresse physique (MAC, niveau 2) d'une machine connaissant son adresse IP (logique, niveau 3).

1.2.4.3 Le protocole ICMP (Internet Control Message Protocol)

Le protocole ICMP (Internet Control Message Protocol) permet de gérer les informations relatives aux erreurs du protocole IP. Il ne permet pas de corriger ces erreurs, mais d'en informer les différents émetteurs des Datagrammes en erreurs. [2]

Chaque pile IP, que ce soit des routeurs ou des stations de travail, gère ICMP par défaut. Les messages d'erreur ICMP sont transportés sur le réseau sous forme de datagramme, comme n'importe quelle donnée. [1] [2]

Ainsi, les messages d'erreurs peuvent eux-mêmes être sujets aux erreurs. Toutefois, en cas d'erreur sur un message ICMP, aucune trame d'erreur n'est délivrée pour éviter un effet "boule de neige". ICMP permet aux routeurs IP d'envoyer des messages d'erreurs et de contrôle à des hôtes ou à d'autres routeurs IP.

1.2.4.4 Le protocole IGMP (Internet Group Message Protocol)

Le protocole IGMP (Internet Group Management Protocol) permet de gérer les déclarations d'appartenance à un ou plusieurs groupes auprès des routeurs Multicast. Les inscriptions sont soit spontanées soit après requête du routeur. Pour cela, l'hôte envoi une trame IGMP destinées à ce ou ces groupes. [1]

1.3 Différentes classes d'adresses IP

L'Internet est donc un réseau basé sur un ensemble de protocoles : les protocoles de la famille TCP/IP. Pour localiser les machines, on fait usage d'adresses. Ces dernières sont utilisées à de nombreux niveaux dans les paquets qui transitent sur le réseau. Les adresses IP peuvent donc être représentées sur 32 bits. Ces 32 bits sont séparés en deux zones de bits contiguës [2] :

- Network ID : une partie décrit le numéro du réseau local auquel est rattachée la station.
- Host ID : une partie correspond au numéro de la station dans le réseau local lui-même, appelée numéro d'hôte.

Selon l'adresse IP on définit différentes classes d'adresses. Il existe cinq classes d'adresses avec la version 4 (version courante) des protocoles TCP/IP, car les parties réseau et hôte n'ont pas toujours la même taille.

1.3.1 Adresses de classe A

Un bloc d'adresses de classe A a été créé pour prendre en charge les réseaux de très grande taille, comportant plus de 16 millions d'adresses d'hôte. Les adresses IPv4 de classe A utilisaient un préfixe /8 invariable, le premier octet indiquant l'adresse réseau. Les trois octets restants correspondaient aux adresses d'hôte. [1] [3]

1.3.2 Adresses de classe B

L'espace d'adressage de classe B a été créé pour répondre aux besoins des réseaux de taille moyenne ou de grande taille, comportant plus de 65 000 hôtes. Les adresses IP de classe B utilisaient les deux premiers octets pour indiquer l'adresse réseau. Les deux octets suivants correspondaient aux adresses d'hôte. Comme avec la classe A, l'espace d'adressage pour les classes d'adresses restantes devait être réservé. [1] [3]

1.3.3 Adresses de classe C

L'espace d'adressage de la classe C était le plus disponible des anciennes classes d'adresses. Cet espace d'adressage était réservé aux réseaux de petite taille, comportant 254 hôtes au maximum. Les blocs d'adresses de classe C utilisaient le préfixe /24. Ainsi, un réseau de classe C ne pouvait utiliser que le dernier octet pour les adresses d'hôte, les trois premiers octets correspondant à l'adresse réseau. [1] [3]

1.3.4 Adresses de classe D

Ces adresses sont utilisées pour la multidiffusion

1.3.5 Adresses de classe E

Ces adresses sont des adresses expérimentales

Classe d'adresse	Plage du premier octet (décimale)	Bits du premier octet (les bits verts ne changent pas)	Parties réseau(N) et hôte (H) de l'adresse	Masque de sous-réseau par défaut (décimal et binaire)	Nombre de réseaux et d'hôtes possibles par réseau
A	1-127**	00000000-01111111	N.H.H.H	255.0.0.0	128 réseaux (2^7) 16 777 214 hôtes par réseau (2^24-2)
B	128-191	10000000-10111111	N.N.H.H	255.255.0.0	16 384 réseaux (2^14) 65 534 hôtes par réseau (2^16-2)
C	192-223	11000000-11011111	N.N.N.H	255.255.255.0	2 097 150 réseaux (2^21) 254 hôtes par réseau (2^8-2)
D	224-239	11100000-11101111	S.O. (multidiffusion)		
E	240-255	11110000-11111111	S.O. (expérimental)		

** Les adresses d'hôtes contenant uniquement des zéros (0) et des uns (1) ne sont pas valides.

Tableau 1.01: *Classe d'adresse IP*

1.4 Adresses privées et adresses publiques

L'organisme IANA offre un plan d'attribution d'adresse pour les réseaux connectés à Internet (réseau publique). Or, tous les réseaux n'ont pas nécessairement un besoin d'interconnexion via un réseau public, dans ce cas l'unicité d'adresse au plan mondial est inutile. Certaines entreprises disposent de leur propre réseau (réseau privé) et n'ont aucun besoin d'interconnexion vers l'extérieur, il est alors possible d'utiliser n'importe quelle adresse IP. Toutefois, afin d'éviter l'anarchie dans l'utilisation des adresses, l'IANA a défini dans la RFC 1918 des plages d'adresses réservées pour ces réseaux privés [2]. Ces adresses sont dites privées et donc non routables sur Internet. Le tableau suivant indique ces plages d'adresses.

Classe d'adresses	Plages d'adresses privées	Masque réseau	Nombre de machines adressables	Nombre de réseaux adressables
A	10.0.0.0 à 10.255.255.255	255.0.0.0	Sur 24 bits, soit 16777216 machines	1
B	172.16.0.0 à 172.31.255.255	255.240.0.0	Sur 20 bits, soit 1048576 machines	16
C	192.168.0.0 à 192.168.255.255	255.255.0.0	Sur 16 bits, soit 65536 machines	256

Tableau 1.02: *Plages d'adresses privées*

Dès que ces réseaux privés ont des besoins de se connecter à un réseau public comme Internet, il faut convertir ces adresses privées en des adresses publiques. Pour ce faire on doit renuméroter tous les terminaux avec des adresses publiques ou bien réaliser une translation d'adresses au moyen d'un translateur d'adresse NAT. [5]

Le processus de NAT fait intervenir une entité entre un terminal, ayant une adresse IP privée, et tout autre ayant une adresse IP public. Ce mécanisme consiste à insérer un boîtier, appelé passerelle NAT, entre le réseau Internet et le réseau privé (LAN, intranet d'une entreprise,…). Ce boîtier se charge de la translation des adresses IP privées en des adresses IP publiques, et effectue aussi l'opération inverse [1]. Actuellement, la plupart des boîtiers (Internet Box) des FAI proposent à leurs abonnés cette fonctionnalité. La figure suivante illustre ce principe de NAT.

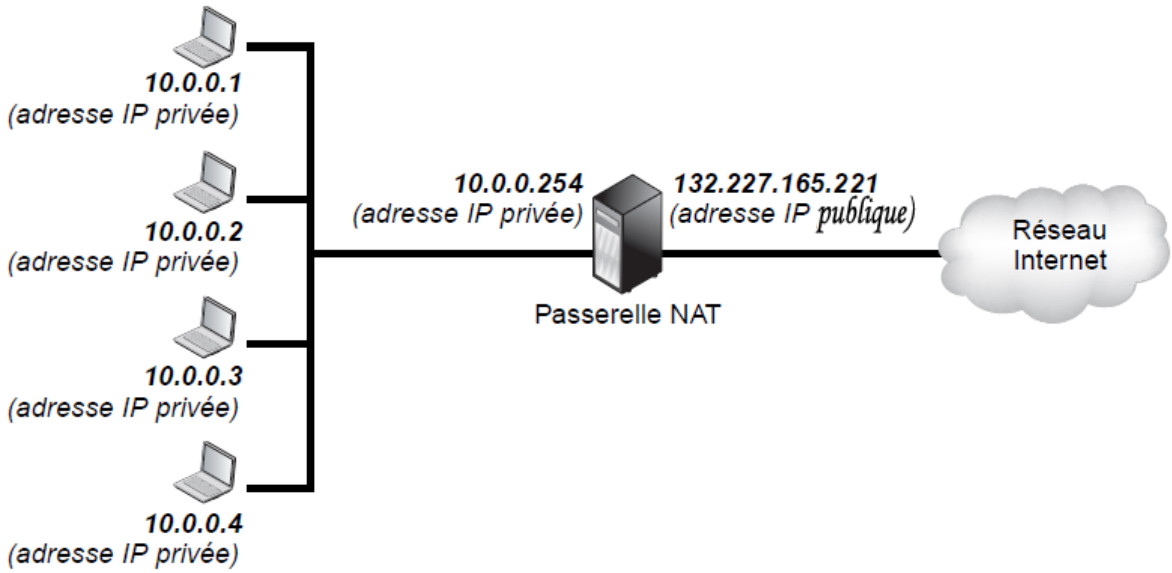

Figure 1.05 : *Translation d'adresses*

La traduction d'adresse par la passerelle NAT peut être statique ou dynamique. Dans le cas du NAT statique, une adresse IP public fixe est associée à chaque adresse IP privée. Une table NAT renseignée par l'administrateur du réseau contient cette association d'adresse. Pour le NAT dynamique, une plage d'adresses publiques est disponible et partagée par tous les utilisateurs du réseau privé. Chaque fois qu'une demande d'un utilisateur du réseau privé arrive à la passerelle NAT, celui-ci attribut dynamiquement une adresse IP publique. [1] [3]

1.5 Routage des Datagrammes IP

Le routage est la méthode par laquelle les périphériques réseaux dirigent les messages à travers les réseaux, afin qu'ils parviennent à leur destination. Ces routes peuvent être attribuées de façon statique au routeur par un administrateur ou être indiquées de façon dynamique au routeur par un autre routeur via un programme appelé protocole de routage.

Au niveau de la couche Internet de l'ensemble de protocoles de la pile TCP/IP, un routeur peut utiliser un protocole de routage IP pour réaliser le routage par la mise en œuvre d'un algorithme de routage particulier.

1.5.1 Routage

C'est un processus qui permet d'acheminer un datagramme IP de son hôte émetteur jusqu'à son hôte destinataire. Chaque datagramme est routé indépendamment des autres. Pour ce faire, le routeur doit rechercher les informations de routage stockées dans sa table de routage [1]. Une table de routage est un fichier de données dans la mémoire vive servant à stocker les informations sur la route à emprunter sur les réseaux directement connectés et les réseaux distants. [5]

1.5.2 Différents types de routes

1.5.2.1 Routes directement connectées

Lorsque l'hôte émetteur et l'hôte destinataire sont sur un réseau commun. L'hôte émetteur peut donc envoyer directement le datagramme sans passer par un ou plusieurs routeurs.

1.5.2.2 Routes statiques

Les routes statiques sont communément utilisées lors du routage d'un réseau vers un réseau d'extrémité. Un réseau d'extrémité est un réseau accessible par une seule route. Les routes statiques sont donc configurées pour la connectivité avec les réseaux distants qui ne sont pas connectés directement à un routeur.

1.5.2.3 Routes dynamiques

Les réseaux distants peuvent également être ajoutés à la table de routage à l'aide d'un protocole de routage dynamique. Les protocoles de routage dynamique sont utilisés par les routeurs pour partager des informations sur l'accessibilité et l'état des réseaux distants [1]. Les protocoles de routage dynamique effectuent plusieurs tâches, notamment :

- la détection de réseaux

– la mise à jour des tables de routage.

1.5.3 Protocoles de routage

Il existe plusieurs protocoles de routage dynamique IP. Voici quelques-uns des protocoles de routage dynamiques les plus répandus en matière de routage des paquets IP [1] [3] :

– RIP (Routing Information Protocol)
– IGRP (Interior Gateway Routing Protocol)
– EIGRP (Enhanced Interior Gateway Routing Protocol)
– OSPF (Open Shortest Path First)
– IS-IS (Intermediate System-to-Intermediate System)
– BGP (Border Gateway Protocol)

Les protocoles de routage dynamique sont utilisés dans les réseaux depuis le début des années 80.

La première version du protocole RIP a vu le jour en 1982, mais certains de ses algorithmes de base étaient déjà utilisés dans ARPANET depuis 1969.

De nouveaux protocoles de routage ont émergé à mesure que les réseaux ont évolué et se sont complexifiés. La figure ci-dessus illustre la classification des protocoles de routage.

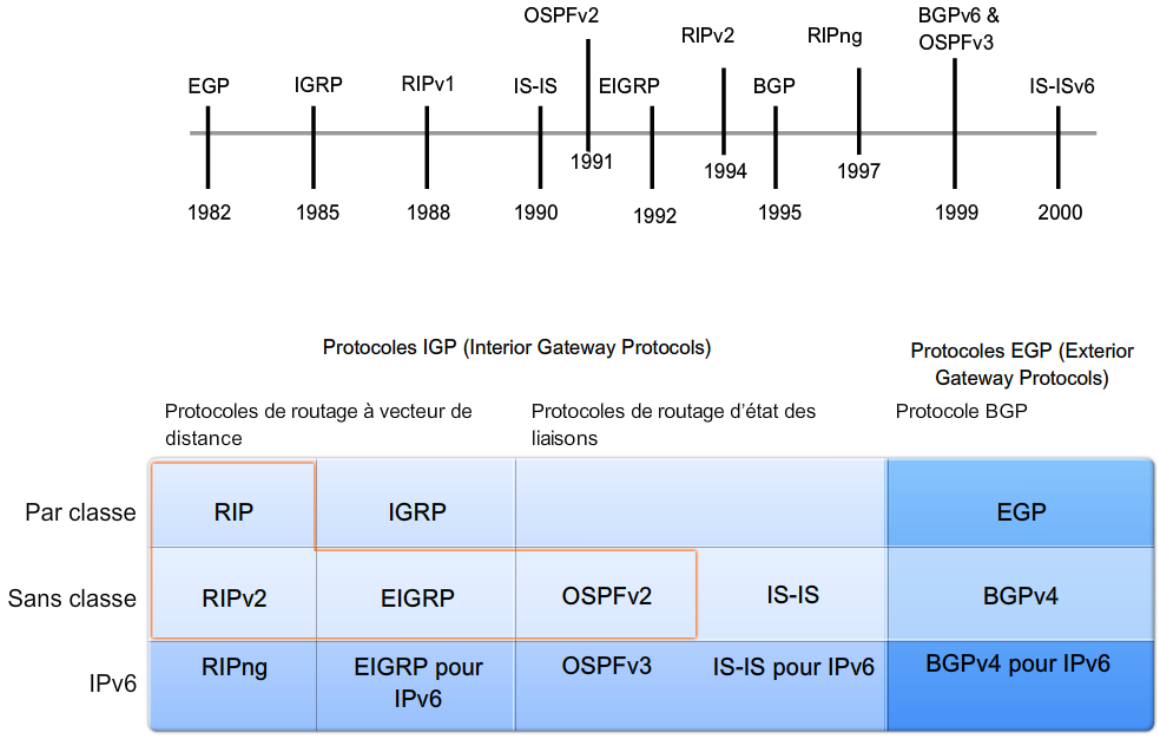

Figure 1.06 : *Evolution des protocoles de routages*

1.5.3.1 Le protocole RIP (Routing Information Protocol)

RIP est le protocole le plus utilisé dans l'environnement TCP/IP pour router les paquets entre les passerelles du réseau Internet. C'est un protocole IGP (Interior Gateway Protocol), qui utilise un algorithme permettant de trouver le chemin le plus court.

Par chemin, on entend le nombre de nœuds traversés, qui doit être compris entre 1 et 15. La valeur 16 indique une impossibilité. En d'autres termes, si le chemin pour aller d'un point à un autre du réseau Internet est supérieur à 15, la connexion ne peut être mise en place. Les messages RIP permettant de dresser les tables de routage sont envoyés approximativement toutes les 30 secondes. Si un message RIP n'est pas parvenu à son voisin au bout de trois minutes, ce dernier considère que le lien n'est plus valide, le nombre de liens étant supérieur à 15. Le protocole RIP se fonde sur une diffusion périodique des états du réseau d'un routeur vers ses voisins. La version RIP2 comporte un routage par sous-réseau, l'authentification des messages, la transmission multipoint, etc. [1] [3]

1.5.3.2 Le protocole OSPF (Open Shortest Path First)

OSPF fait partie de la deuxième génération de protocoles de routage. Beaucoup plus complexe que RIP, mais au prix de performances supérieures, il utilise une base de données distribuée, qui garde en mémoire l'état des liens. Ces informations forment une description de la topologie du réseau et de l'état des nœuds, qui permet de définir l'algorithme de routage par un calcul des chemins les plus courts.

L'algorithme OSPF permet, à partir d'un nœud, de calculer le chemin le plus court, avec les contraintes indiquées dans les contenus associés à chaque lien. Les routeurs OSPF communiquent entre eux par l'intermédiaire du protocole OSPF, placé au-dessus d'IP. [3]

L'hypothèse de départ pour les protocoles à état des liens est que chaque nœud est capable de détecter l'état du lien avec ses voisins (marche ou arrêt) et le coût de ce lien. Il faut donner à chaque nœud suffisamment d'informations pour lui permettre de trouver la route la moins chère vers toutes les destinations. Chaque nœud doit donc avoir la connaissance de ses voisins. Si chaque nœud à la connaissance des autres nœuds, une carte complète du réseau peut être dressée. Un algorithme se fondant sur l'état des voisins nécessite deux mécanismes : la dissémination fiable des informations sur l'état des liens et le calcul des routes par sommation des connaissances accumulées sur l'état des liens. [1] [3]

1.5.3.3 IS-IS

L'algorithme IS-IS a été principalement développé par l'ISO (ISO 10589). Il décrit un routage hiérarchique fondé sur la décomposition des réseaux de communication en domaines. Dans un domaine, les différents nœuds indiquent leur état aux routeurs IS-IS afférents [1]. Les communications interdomaines sont effectuées par un routage vers un point d'accès au domaine déterminé par les routeurs chargés des communications externes au domaine. [3]

1.5.3.4 IGRP

Version améliorée de RIP, IGRP a été conçu par Cisco Systems pour ses propres routeurs. Il intègre le routage multichemin, la gestion des routes par défaut, la diffusion de l'information toutes les 90 secondes au lieu de toutes les 30 secondes, la détection des bouclages, c'est-à-dire des retours à un point par lequel le paquet est déjà passé, etc [1]. Ce protocole a lui-même été étendu pour une meilleure protection contre les boucles par le protocole EIGRP (Extended IGRP).

1.5.4 Routage des datagrammes

Lorsque le routeur reçoit une trame physique, il en extrait le datagramme qu'elle contient. Puis il met en œuvre un algorithme de routage utilisant une table de routage pour déterminer vers quel réseau physique il va propager le datagramme.

Il détermine vers quel réseau physique il devra envoyer la trame en fonction de l'adresse de destination IP contenue dans l'en-tête du datagramme. Il encapsule le datagramme dans une nouvelle trame et l'émet vers le réseau physique voulu.

L'en-tête d'un datagramme en transit dans l'inter-réseau a toujours comme adresse IP source celle de l'hôte émetteur du datagramme et comme adresse destination celle de l'hôte destinataire. [1]

1.5.5 Tables de routage

Toute décision de routage est prise en fonction de l'adresse du réseau de destination du datagramme. Les tables de routage ne contiennent donc que des adresses réseau. Une entrée d'une table de routage contient 3 champs :

- Network Address : contient l'adresse IP d'un réseau.
- Subnet Mask : contient le SubnetMask associé à Network Address.

- IP Address : contient soit l'adresse IP du prochain routeur dans la direction du réseau à atteindre, soit la mention Deliver Directly alors la machine est connectée sur le même réseau physique que la machine destination.

1.6 DNS (Domain Name System)

Ce système consiste à identifier une machine par un nom plutôt que par son adresse IP [1]. Cependant pour qu'il n'y ait pas deux machines avec le même nom, il convient d'établir une hiérarchisation. Le mécanisme qui implémente l'adressage hiérarchique nominatif s'appelle DNS (Domain Name Service).

Un Domain Name est une suite de sous-noms appelés labels, séparés par des points. Le réseau Internet propose deux systèmes de hiérarchie :

- Organisationnel : basé sur la nature de l'activité de la société, exemple : COM : commercial organizations, EDU : education institutions, GOV : government institutions.
- Géographique, exemple : DZ : DJAZAIR. FR : FRANCE. UK : UNITED KINGDOM.

1.7 DHCP (Dynamic Host Configuration Protocol)

Protocole de service TCP/IP qui offre une configuration louée dynamique d'adresses IP hôte et qui distribue d'autres paramètres de configuration aux clients réseaux admissibles. DHCP fournit une configuration de réseau TCP/IP sûre, fiable et simple, qui évite les conflits d'adresse et permet de continuer à utiliser des adresses IP par clients sur le réseau. DHCP utilise un modèle client/serveur dans lequel le serveur DHCP assure la gestion centralisée des adresses IP utilisées sur le réseau. Les clients qui prennent en charge DHCP peuvent ensuite demander et obtenir la location d'une adresse IP auprès d'un serveur DHCP dans le cadre de leur procédure d'amorçage réseau. [1] [4]

1.8 Conclusion

Ce chapitre nous a montré une vue plus ou moins générale de l'architecture du modèle TCP/IP et les protocoles ainsi que les services qui vont avec. Certes, d'autres modèles peuvent être étudiés, mais on a jugé utile de souligner en particulier l'architecture TCP/IP. Quand on parle du protocole IP, on a tendance à oublier qu'il y a le protocole IPv6 successeur de l'IPv4. Le protocole IPv4 est fort connu du fait que c'est le protocole IP utilisé actuellement et ce premier chapitre a mis l'accent sur les traits caractéristiques de ce protocole. Le deuxième chapitre abordera en détail l'évolution du protocole Internet qui n'est tout autre que le protocole Internet version 6.

CHAPITRE 2
PROTOCOLE INTERNET VERSION 6

2.1 Introduction

En raison des récents événements suivants, l'importance du protocole Internet version 6 (IPv6) pour l'avenir de l'Internet est maintenant indéniable :

- Le 3 Février 2011, l'ICANN (Internet Corporation Assigned Names and Numbers) a rejoint le NRO (Number Resources Organization), l'IAB (Internet Architecture Board) et l'Internet Society (ISOC) pour annoncer que le dernier lot d'adresses IPv4 publiques a été entièrement alloué. L'espace d'adressage IPv4 publique existe encore pour être affecté à des organisations par les autorités régionales d'adresses, mais il n'y a plus d'espace d'adresses IPv4 publiques en réserve.
- Le 8 Juin 2011, Microsoft et d'autres membres de l'ISOC ont participé à la Journée mondiale de l'IPv6 pour un test temporaire de connectivité et de performance du Dual Stack (IPv4 et IPv6) sur Internet.
- En Avril 2012, l'Internet Engineering Task Force (IETF) a publié le RFC 6540, "IPv6 Support Required for All IP-Capable Nodes". Ce RFC signale que le support IPv6 est nécessaire pour tous les nœuds du réseau Internet, en plus d'IPv4.
- Le 6 Juin 2012, Microsoft et d'autres membres de l'ISOC ont participé à World IPv6 Launch pour activer en permanence le Dual Stack sur Internet.

"The time has come to embrace, learn, and understand IPv6."

2.2 Limites de l'IPv4

La version actuelle du protocole Internet (connu comme la version 4 ou IPv4) n'a pas beaucoup changé depuis RFC 791, qui a été publiée en 1981. IPv4 s'est avéré être robuste, facile à mettre en œuvre, et interopérable. Il a résisté à l'épreuve de l'élargissement de l'inter-réseau à un service mondial de la taille de l'Internet d'aujourd'hui.

Cependant, la conception initiale de l'IPv4 n'avait pas prévu ce qui suit :

- **La croissance exponentielle récente de l'Internet et l'épuisement imminent de l'espace d'adressage IPv4.** Bien que l'espace d'adressage 32 bits d'IPv4 permette 4,294,967,296 d'adresses, les pratiques d'attribution antérieures et actuelles limitent le nombre d'adresses IPv4 publiques à quelques centaines de millions [7]. En conséquence, les adresses IPv4 publiques sont devenues relativement rares, forçant de nombreux utilisateurs et certaines organisations d'utiliser un NAT. Bien que NAT favorise la réutilisation de l'espace d'adressage privé, il viole le principe fondamental de conception de l'Internet d'origine que tous les nœuds ont une adresse unique, accessible à l'échelle mondiale, empêchant ainsi une véritable connectivité de bout en bout pour tous les types d'applications du réseau. En outre, l'importante croissance des dispositifs et appareils connectés à Internet assure que l'espace d'adressage IPv4 publique va finir par baisser.

- **La nécessité d'une configuration plus simple.** La plupart des implémentations IPv4 actuelles doivent soit être configurées manuellement soit utiliser le protocole Dynamic Host Configuration (DHCP). Avec plusieurs ordinateurs et appareils utilisant le protocole IP, il existe un besoin pour une configuration plus simple et plus automatique d'adresses qui ne s'appuie pas sur l'administration d'une infrastructure DHCP. [7]

- **L'exigence de la sécurité à la couche Internet.** La communication privée sur un support public comme l'Internet nécessite des services cryptographiques qui protègent les données envoyées d'être consultées ou modifiées en transit. Bien que la norme existe désormais pour assurer la sécurité des paquets IPv4 (connus sous le nom de *« Internet Protocol Security »,* ou IPsec), cette norme est facultative pour IPv4, et des solutions de sécurité supplémentaires, dont certaines sont exclusives, sont répandues. [7]

Pour répondre à ces préoccupations et d'autres, l'IETF a développé une suite de protocoles et de normes connues sous le nom de IP version 6 (IPv6). Cette nouvelle version, précédemment appelée IP The Next Generation (IPng), intègre les concepts de nombreuses méthodes proposées pour la mise à jour du protocole IPv4.

2.3 Structure d'un paquet IPv6

Le protocole IPv6 représente la nouvelle génération du protocole IP, d'où le nom d'IPng (*Next generation*) qu'on lui donne également. C'est un protocole entièrement repensé par rapport à IPv4 et donc réellement nouveau. Le format du paquet IPv6 est illustré à la figure 2.01.

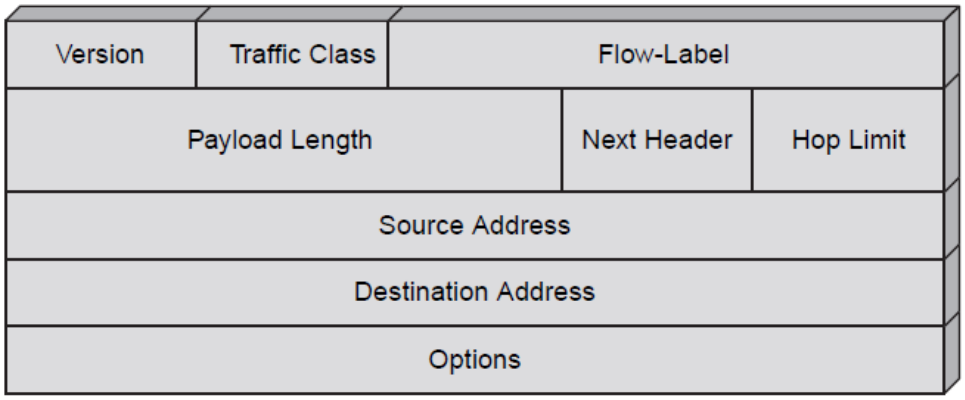

Figure 2.01 : *Format du paquet IPv6*

Ce paquet se présente de la façon suivante. La version porte le numéro 6. Le champ qui suit indique un niveau de priorité, qui permet de traiter les paquets plus ou moins rapidement dans les nœuds du réseau. Les principales valeurs de ce champ sont les suivantes [1] :

- 0 : pas de priorité particulière ;
- 1 : trafic de base (nouveau);
- 2 : transfert de données sans contrainte temporelle (e-mail) ;
- 3 : réservé pour des développements futurs ;
- 4 : transfert en bloc avec attente du récepteur (transfert de fichiers) ;
- 5 : réservé pour des développements futurs ;
- 6 : trafic interactif (*terminal virtuel* ou *rlogin*)
- 7 : trafic pour le contrôle (routage, contrôle de flux).

Le champ Référence de flot, ou *Flow-Label* est également nouveau. Il permet de transporter une référence (*label*), capable de préciser le flot auquel le paquet appartient et donc d'indiquer la qualité de service demandée par les informations transportées. Cette référence permet aux routeurs de prendre les informations transportées. Grâce à ce nouveau champ, le routeur peut traiter de façon personnalisée les paquets IPv6, autorisant ainsi la prise en compte des contraintes diverses. [1] [7]

Le champ Longueur, ou *Payload Lenght*, indique la longueur totale du datagramme en octet (sans tenir compte de l'en-tête). Ce champ étant de 2 octets, la longueur maximale du datagramme est de 64 Ko. [1]

Le champ En-tête suivant, ou *Next Header*, indique le protocole encapsulé dans la zone de données du paquet. Ce processus est illustré à la figure 2.02. Les options les plus classiques pour la valeur de ce champ sont 0 pour *Hop-by-Hop Option Header*, 4 pour IP, 6 pour TCP, 17 pour UDP et 58 pour ICMP. [1] [7]

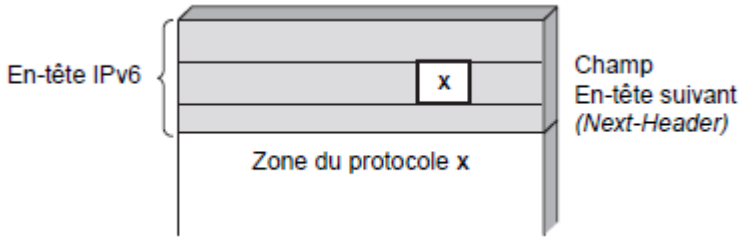

Figure 2.02 : *Champ En-tête suivant*

Le champ Nombre de nœuds traversés (*Hop Limit*) indique après combien de nœuds le paquet est détruit.

Le champ d'adresse est souvent présenté comme la raison d'être de la nouvelle version d'IP. En fait, c'est seulement une raison parmi d'autres. L'adresse IPv6 tient sur 16 octets. La difficulté réside dans la représentation et l'utilisation rationnelle de ces 128 bits. Le nombre d'adresses potentielles dépasse 1023 pour chaque mètre carré de la surface terrestre. De plus amples informations seront présentées dans la partie consacrée à l'adressage en IPv6.

L'en-tête du paquet IPv6 se termine par un champ d'options qui permet l'ajout de nouvelles fonctionnalités, en particulier concernant la sécurité. La figure 2.03 illustre le fonctionnement de ce champ d'options. Chaque zone d'extension commence par un champ portant un numéro correspondant au type d'extension. Les possibilités, qui ont déjà pu être utilisées dans la partie *En-tête suivant* sont les suivantes [1] :

- 0 : *Hop-by-Hop Option Header* ;
- 43 : *Routing Header* ;
- 44 : *Fragment Header* ;
- 51 : *Authentication Header* ;

- 59 : *No Next Header* ;
- 60 : *Destination Options Header.*

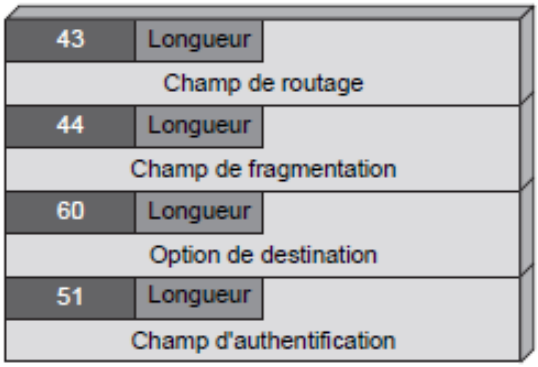

Figure 2.03 : *Champ d'extension avec quatre options*

Dans ce champ d'options, les différentes zones se suivent dans un ordre prédéterminé, qui est dicté par leur utilisation potentielle dans les nœuds intermédiaires. Si un nœud intermédiaire ne peut prendre en charge une option, plusieurs cas de figure se présentent : destruction du paquet, émission sans traitement, émission d'une signalisation ou attente d'une réponse pour prendre une décision. La figure 2.04 donne une idée de l'ordre de traitement.

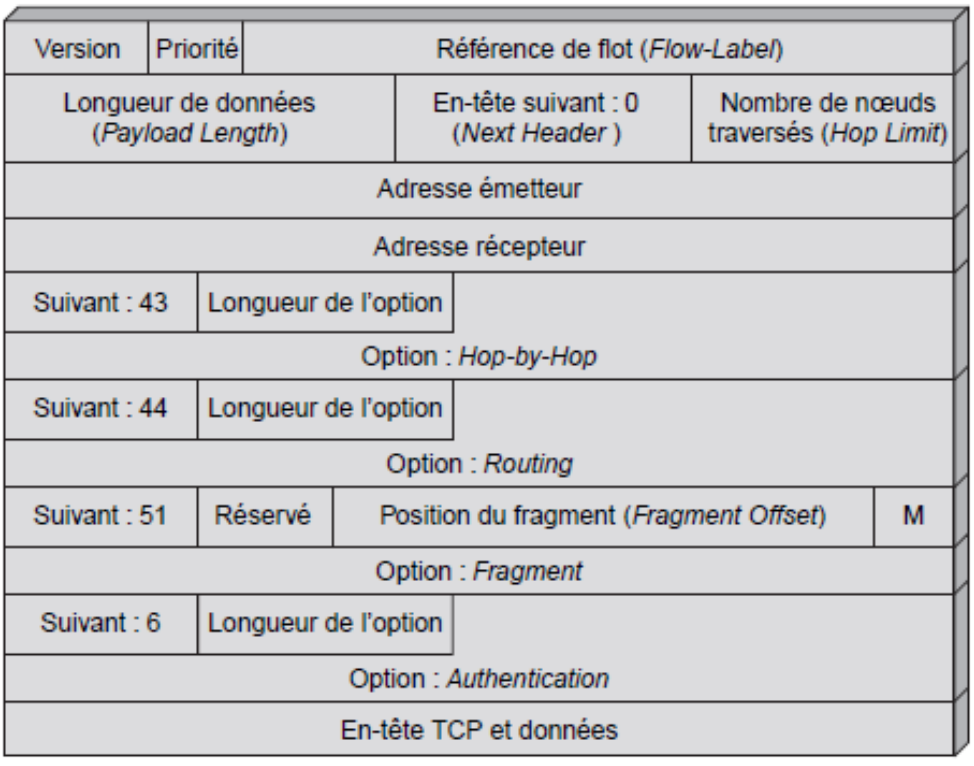

Figure 2.04 : *Traitement des options d'extension*

2.4 Comparaison des en–têtes IPv4 et IPv6

On a vu dans le chapitre consacré aux réseaux TCP/IP l'en-tête de l'IPv4, le tableau 2.01 illustre les différences entre cet en-tête et celui de l'IPv6.

IPv4	IPv6
Version	Même champ mais avec numéro de version différent.
Internet Header Length	Supprimé en IPv6. Ipv6 n'inclut pas le Header Length parce que l'en-tête IPv6 est toujours fixe de 40 octets.
Type of service	Remplacé par le champ Traffic Class Field.
Total Length	Remplacé par le champ Payload Lenght qui indique seulement la taille du datagramme (sans tenir compte de l'en-tête).
Identification **Flags** **Fragment Offset**	Supprimé en IPv6. L'information de fragmentation n'est pas inclue dans l'en-tête IPv6. C'est contenu dans l'en-tête d'extension "Fragment Header".
Time-To-Live	Remplacé par le champ Hop Limit.
Protocol	Remplacé par le champ Next Header.
Header Checksum	Supprimé en IPv6.
Source Address	Le champ est le même excepté que les adresses IPv6 sont de 128 bits.
Destination Address	Le champ est le même excepté que les adresses IPv6 sont de 128 bits.
Options	Supprimé en IPv6. Les en-têtes d'extension remplacent les options en IPv4.

Tableau 2.01 : *Comparaison des en-têtes IPv4 et IPv6*

2.5 Adresses IPv6

2.5.1 Représentation des adresses

Les adresses IPv6 sont codées sur 128 bits soit 16 octets.

La notation décimale pointée employée pour les adresses IPv4 (par exemple 172.31.128.1) est abandonnée au profit d'une écriture hexadécimale, où les 8 groupes de 2 octets (soit 16 bits par groupe) sont séparés par un signe deux-points « : ».

La notion historique de classes a totalement disparu, au profit de l'utilisation exclusive des préfixes et de la notation CIDR avec le slash et le masque, déjà utilisés en IPv4. Les masques par défaut disparaissent donc. [7]

L'identifiant réseau de l'adresse est nommé préfixe. La longueur du préfixe, sous la forme de /x, indique le nombre de bits dans l'identifiant réseau de l'adresse.

Exemple : 2001 : 0AB8 : 3409 : C0AB : 0001 : AEFF : FE00 : C801 /64

Une unique suite de un ou plusieurs groupes consécutifs de 16 bits tous nuls peut être omise, en conservant toutefois les signes deux-points de chaque côté de la suite de chiffres omise, c'est-à-dire une paire de deux points (::). [7]

Règle n°1 : *Les groupes complets de 0 consécutifs peuvent être remplacés par « :: », une seule fois dans l'adresse.*

Il est permis d'omettre de 1 à 3 chiffres zéros non significatifs dans chaque groupe de 4 chiffres hexadécimaux. [7]

Règle n°2 : *Les 0 non significatifs ne doivent pas être écrits.*

Exemple :
Adresse complète : 2001 : ABCD : 0000 : 0000 : 0ADE : 0000 : 0123 : C891
Règle n°1 : 2001 : ABCD : : 0ADE : 0000 : 0123 : C891
Règle n°2 : 2001 : ABCD : : ADE : 0 : 123 : C891

Remarque : Pour les cas où le ' : ' a un sens (par exemple dans une URL), on met l'adresse IPv6 entre [] pour éviter toute confusion. Exemple : http://[::1]/

2.5.2 Types d'adresses

IPv6 reconnaît trois types d'adresses : Unicast, Multicast et Anycast. [11]

Le premier de ces types, le type Unicast, est le plus simple. Une adresse de ce type désigne une interface unique. Un paquet envoyé à une telle adresse, sera donc remis à l'interface ainsi identifiée.

Une adresse de type Multicast désigne un groupe d'interfaces qui en général appartiennent à des nœuds différents pouvant être situés n'importe où dans l'Internet. Lorsqu'un paquet a pour destination une adresse de type multicast, il est acheminé par le réseau à toutes les interfaces membres de ce groupe.

Le dernier type, Anycast, est une officialisation de propositions faites pour IPv4 RFC 1546. Comme dans le cas du multicast, une adresse de type anycast désigne un groupe d'interfaces, la différence étant que lorsqu'un paquet a pour destination une telle adresse, il est acheminé à un des éléments du groupe et non pas à tous. C'est, par exemple, le plus proche au sens de la métrique des protocoles de routage.

2.5.2.1 Adresses Unicast

Il s'agit du même type d'adresse qu'en IPv4. Un paquet envoyé à une adresse unicast sera émis seulement à l'interface identifiée par cette adresse.

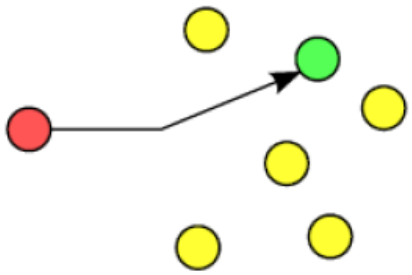

Figure 2.05 : *Principe de l'Unicast*

On distingue trois types d'adresses unicast :

- **Global Unicast Address** : adresse globale unicast. Il s'agit des adresses qui sont uniques dans le monde, et qui sont par conséquent routables sur Internet. Elles se composent d'un préfixe de routage global (/48), suivi du préfixe de sous-réseau (/16) et de l'identifiant d'interface (/64) (*Figure 2.06*). L'IANA fournit une liste des préfixes affectés aux différents RIR qui permettent donc de classer les IP rencontrées sur le réseau mondial par région du monde. [7] [11]

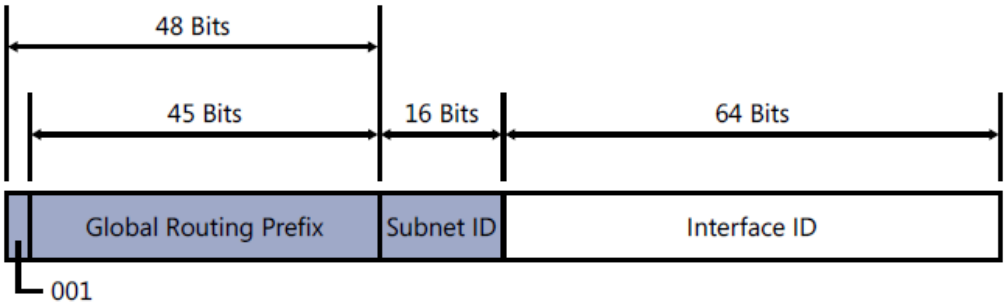

Figure 2.06 : *Structure d'une adresse globale unicast*

La RFC 2374 structure les adresses globales unicast (c'est-à-dire les adresses publiques) de façon hiérarchique :

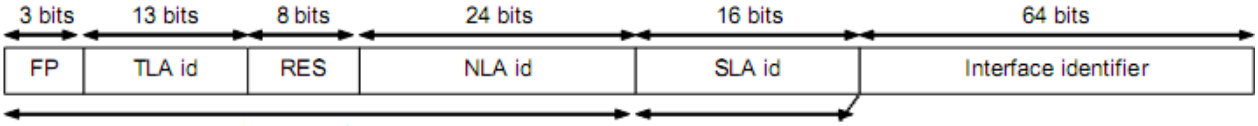

Figure 2.07 : *Hiérarchie d'une adresse globale unicast*

Champ	Description
FP	*Format Prefix* (=001) : identifie le type d'adresse global unicast
TLA	*Top Level Aggregation identifier* : identifie un organisme gérant un réseau public tel que l'IANA pour l'Internet
RES	Réservé pour étendre les champs TLA et NLA (notamment à la désignation régionale de l'IANA)
NLA	*Next-Level Aggregation identifier* : identifie un opérateur réseau tel un ISP
SLA	*Site-Level Aggregation identifier* : identifie 65 535 sous-réseaux au sein d'une organisation (équivalent aux subnets IPv4)
Interface Identifier	Adresse de l'interface au format IEEE EUI-64

Tableau 2.02 : *Description des champs*

Le format de l'adresse unicast globale retenu pour l'Internet est le suivant :

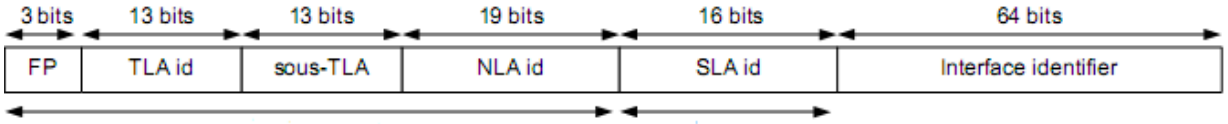

Figure 2.08 : *Adresse globale unicast*

Le champ sous-TLA correspond aux 8 bits du champ RES, plus 5 bits prélevés sur le champ NLA.

– **Link-local Unicast** : adresses de lien local, non-routables en local comme sur Internet. Elles sont systématiquement générées lors de l'utilisation de l'autoconfiguration sans état (stateless).

  Une interface pour laquelle on active IPv6 se génère automatiquement une adresse link-local, soit en générant les 64 bits hôtes aléatoirement, soit en utilisant la méthode EUI-64. Les adresses Link-local commencent toujours par FE80. Avec l'identifiant d'interface de 64 bits, le préfixe de cette adresse est FE80 :: /64. [7] [11]

Figure 2.09 : *Structure d'une adresse link-local*

Remarque :
La construction automatique de l'adresse IP lors de l'activation d'une interface suit le principe suivant :

- Ajouter les octets **FFFE** au milieu de l'adresse MAC de l'interface.
- Positionner le septième bit de l'adresse MAC modifiée en partant de la gauche à 1 si l'adresse est unique (ce qui est le cas pour toutes les adresses MAC par défaut) sinon 0.
- Récupération du préfixe si c'est une adresse globale, sinon utilisation du préfixe d'adresse de lien local.
- Concaténation du préfixe avec l'adresse MAC ainsi modifiée.

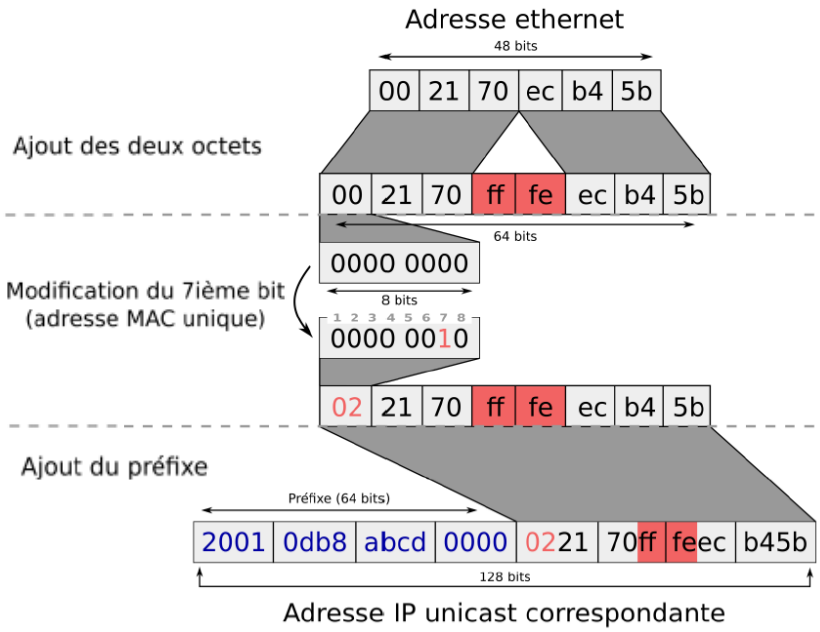

Figure 2.10 : *EUI-64 formé depuis l'EUI-48 correspondant à l'adresse MAC*

– **Unique Local Unicast** : adresses de site local, non-routables sur Internet. C'est la catégorie qui se rapproche le plus des adresses privées IPv4. Elles utilisent toutes le préfixe FC00::/7, mais avec le huitième bit en partant de la gauche (le champ L sur la figure 2.11) positionné à 1 si le préfixe est défini localement. Puisque la valeur 0 n'est pas possible actuellement (usage futur), elles sont reconnaissables par leur premier bloc qui commence systématiquement par FD. [7] [11]

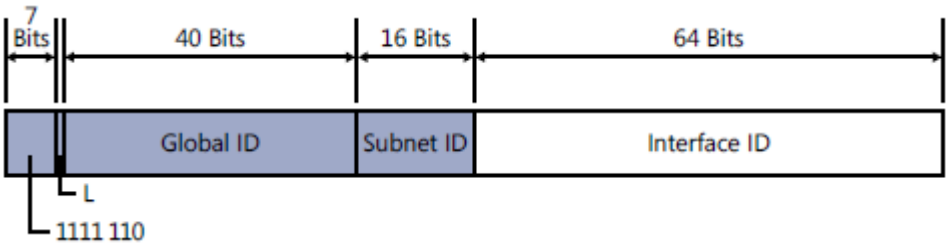

Figure 2.11 : *Structure d'une adresse de site local*

2.5.2.2 Autres types d'adresses

– **Adresse indéterminée**

L'adresse indéterminée (unspecified address) est utilisée comme adresse source par un nœud du réseau pendant son initialisation, avant d'acquérir une adresse. Sa valeur est 0:0:0:0:0:0:0:0 (en abrégé ::).

Cette adresse est utilisée uniquement par des protocoles d'initialisation, elle ne doit jamais être attribuée à un nœud et ne doit jamais apparaître comme adresse destination d'un paquet IPv6. [7]

– **Adresse de bouclage**

L'adresse de bouclage (loopback address) vaut 0:0:0:0:0:0:0:1 (en abrégé ::1). C'est l'équivalent de l'adresse 127.0.0.1 d'IPv4. Elle est utilisée par un nœud pour s'envoyer à lui-même des paquets IPv6. Un paquet IPv6 transitant sur le réseau ne peut avoir l'adresse de bouclage comme adresse source ni comme adresse destination. [7]

– **Adresses IPv4 mappées**

Elles sont représentées sous la forme ::FFFF:a.b.c.d où a.b.c.d est une adresse IPv4. On peut bien entendu aussi les écrire sous la forme ::FFFF:XXXX:YYYY où XXXXYYYY est la représentation hexadécimale de l'adresse IPv4 a.b.c.d. [7]

– **Adresses IPv4 compatibles**

Elles sont représentées sous la forme ::a.b.c.d où a.b.c.d est une adresse IPv4. Comme précédemment, on peut aussi les écrire sous la forme ::XXXX:YYYY où XXXXYYYY est la représentation hexadécimale de l'adresse IPv4 a.b.c.d. [7]

2.5.2.3 Adresses Muticast

Ce type d'adresse (RFC 4291) correspond précisément à son homologue IPv4. Il s'agit d'adresses virtuelles qui distribuent des paquets à tous les membres inscrits dans un groupe [11]. C'est une adresse utilisable comme adresse destination.

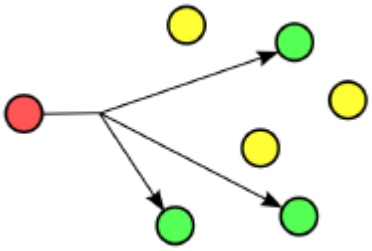

Figure 2.12 : *Principe du Multicast*

Elles utilisent toutes le préfixe FF00::/8. Elles sont donc reconnaissables par leur premier bloc qui commence systématiquement par FF.

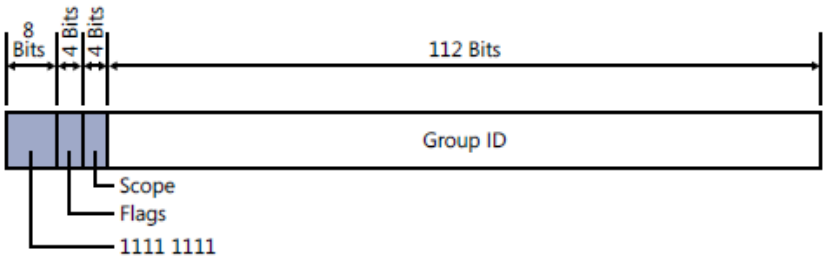

Figure 2.13 : *Structure d'une adresse Multicast*

Les premiers 8 bits à 1 (1111 1111) identifient l'adresse en tant que Multicast.

Le champ *Flags* est un mot de 4 bits. Les 3 premiers bits sont réservés et doivent être initialisés à zéro. Le dernier bit, nommé T, s'il est positionné à zéro indique que l'adresse est *permanente* (auquel cas, elle a été nécessairement attribuée par une autorité compétente de l'Internet), sinon l'adresse est *temporaire.* [11]

Le champ **Scope** délimite le niveau de diffusion du paquet. Il remplace le contenu time-to-live en IPv4. Les valeurs possibles sont présentées dans le tableau 2.03.

Valeur	Description
0	Réservé
1	Nœud (node-local scope)
2	Lien (link-local scope)
3	Réservé
4	Administration
5	Site (site-local scope)
6,7	Non assignés
8	Organisation (organization-local scope)
9, A, B, C, D	Non assignés
E	Global (global scope)
F	Réservé

Tableau 2.03 : *Valeurs du champ Scope*

On distingue également les adresses multicast particulières :

Adresses	Descriptions
FF02 :: /16	Adresses multicast de portée locale.
FF02 :: 1	Toutes les machines du réseau local (remplaçant le broadcast). Toute interface fonctionnant en IPv6 rejoint ce groupe.
FF02 :: 2	Tous les routeurs du réseau local.
FF02 :: 5	Tous les routeurs OSPFv3 du réseau local.
FF02 :: 6	Tous les routeurs OSPFv3 DR/BDR du réseau local.
FF02 :: 9	Tous les routeurs RIPng du réseau local.
FF02 :: A	Tous les routeurs EIGRP du réseau local.
FF02 :: 1 : FF00 :: /104	« Solicited Node multicast address » Adresse multicast dérivée d'une adresse configurée sur l'interface concernée.

Tableau 2.04 : *Adresses multicast particulières*

2.5.2.4 Adresses Anycast

Une adresse Anycast est un nouveau type d'adresse qui est assignée à un ensemble d'interfaces. Le principe sous-jacent est simple : un paquet anycast ayant comme destination un groupe anycast n'est pas envoyé à toutes les interfaces de ce groupe, mais uniquement à une de celle-ci. En général, c'est l'interface la plus proche (par exemple au sens de la métrique des protocoles de routage : RIPng, OSPF). Si le concept anycast est simple dans son principe, son implémentation est autrement délicate. En outre, ce concept n'est encore qu'un sujet de recherche. [7] [11]

Figure 2.14 : *Structure d'une adresse Anycast*

2.5.3 Adresses IPv4 et leurs équivalents IPv6

Le tableau 2.05 liste les adresses IPv4 et leurs équivalents IPv6.

Adresses IPv4	Adresses IPv6
Classes d'adresses	Pas de classes d'adresses en IPv6
Adresses Multicast (224.0.0.0/4)	Adresses Multicast (FF00::/8)
Adresses Broadcast	Pas d'adresses Broadcast en IPv6
Adresse indéterminée (0.0.0.0)	Adresse indéterminée (::)
Adresse loopback (127.0.0.1)	Adresse loopback (::1)
Adresses IP publiques	Adresses globales unicast
Adresses IP privées (10.0.0.0/8, 172.16.0.0/12, 192.168.0.0/16)	Adresses « Unique Local » (FD00::/8)

Tableau 2.05 : *Adresses IPv4 et leurs équivalents IPv6*

2.6 ICMPv6

Le protocole de contrôle d'IP a été revu. Dans IPv4, ICMP (Internet Message Control Protocol) sert à la détection d'erreurs (par exemple : équipement inaccessible, durée de vie expirée,...), au test (par exemple ping), à la configuration automatique des équipements (redirection ICMP, découverte des routeurs). Ces trois fonctions ont été mieux définies dans IPv6. De plus ICMPv6 (RFC 2463) intègre les fonctions de gestion des groupes de multicast (MLD : Multicast Listener Discovery) qui sont effectuées par le protocole IGMP (Internet Group Message Protocol) dans IPv4. ICMPv6 reprend aussi les fonctions du protocole ARP utilisé par IPv4.

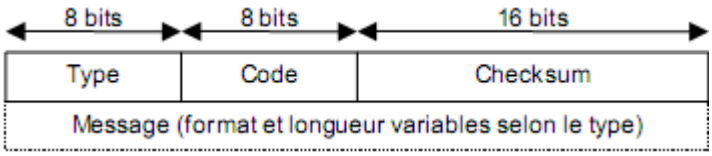

Figure 2.15 : *Format d'un message ICMPv6*

Le protocole se voit attribuer le numéro 58. Le format générique des paquets ICMPv6 est donné par la figure 2.15 :

- Le champ type code la nature du message ICMPv6. Contrairement à IPv4 où la numérotation ne suivait aucune logique, les valeurs inférieures à 127 sont réservées aux messages d'erreur. Les autres valeurs réservées aux messages d'information, parmi lesquels se trouvent ceux utilisés par le protocole découverte des voisins (neighbor discovery) pour la configuration automatique des équipements. [11]
- Le champ code précise la cause du message ICMPv6. [11]
- Le champ checksum permet de vérifier l'intégrité du paquet ICMP. [11]

Les messages ICMPv6 se classifient en 2 catégories :

- Les messages d'erreurs (notés de 0 à 127) : ils rapportent les erreurs rencontrés lors de la transmission d'un paquet IPv6.
- Les messages d'informations (notés de 128 à 255) : ils donnent des informations de diagnostic et des fonctionnalités additionnels de l'hôte.

Type	Message	Code	Description
1	Destination unreachable	0	Le routeur ne connaît pas la route
		1	La communication est interdite par un pare-feu
		2	L'adresse destination dépasse le domaine de routage de l'adresse source
		3	La destination n'est pas joignable
		4	Aucun programme ne répond sur le port TCP ou UDP
2	Packet too big	0	La taille du paquet dépasse le MTU de la couche liaison
3	Time exceeded	0	Le paquet a été reçu avec une limite de saut à 0
		1	Tous les fragments d'un paquet n'ont pas été reçus dans le temps imparti
4	Parameter problem	0	Un code d'extension d'en-tête inconnu a été trouvé
		1	Une option inconnue a été trouvée

Tableau 2.06 : *Messages d'erreurs types en ICMPv6*

Type	Message	Code	Description
128	Echo Request	0	Demande au récépteur de renvoyer un Echo reply (un identifiant et un numéro de séquence identifient le message)
129	Echo Reply	0	Réponse à un Echo request. Les données contenues dans le message Echo request doivent être reportées dans ce message
130	Multicast listener query	0	Demande de rapport envoyée par le routeur
131	Multicast listener report	0	Rapport de participation au groupe envoyé par le membre
132	Multicast listener done	0	Le membre indique au routeur qu'il quitte le groupe
133	Router solicitation	0	Emis par une station pour découvrir les routeurs sur son lien
134	Router advertisement	0	Envoyé par un routeur périodiquement ou en réponse à un message de solicitation pour indiquer les paramètres du lien
135	Neighbor sollicitation	0	Demande de résolution de l'adresse IPv6 en adresse MAC

Tableau 2.07 : *Messages d'informations types en ICMPv6*

Tout comme dans le cas des adresses IPv4 et de leur équivalent IPv6, il existe également une correspondance entre les messages ICMPv4 et les messages ICMPv6. Le tableau 2.08 illustre cette correspondance.

Message ICMPv4	Equivalent ICMPv6
Destination Unreachable – Network Unreachable (Type 3, Code 0)	Destination Unreachable – No Route to Destination (Type 1, Code 0)
Destination Unreachable – Host Unreachable (Type 3, Code 1)	Destination Unreachable – Address Unreachable (Type 1, Code 3)
Destination Unreachable – Protocol Unreachable (Type 3, Code 3)	Parameter Problem – Unrecognized Next Header Type Encountered (Type 4, Code 1)
Destination Unreachable – Fragmentation Needed an DF Set (Type 3, Code 4)	Packet too big (Type 2, Code 0)
Destination Unreachable – Communication with Destination Host Administratively Prohibited (Type 3, Code 10)	Destination Unreachable – Communication with Destination Administratively Prohibited (Type 1, code 1)
Source Quench (Type 4, Code 0)	Ce message n'est pas présent en IPv6
Redirect (Type 5, Code 0)	Neighbor Discovery Redirect message (Type 137, Code 0)
Time Exceeded – TTL Exceeded in transit (Type 11, Code 0)	Time Exceeded – Hop Limit Exceeded in transit (Type 3, Code 0)
Time Exceeded –Fragment Reassembly Time Exceeded (Type 11, Code 1)	Time Exceeded – Fragment Reassembly Time Exceeded (Type 3, Code 1)
Parameter Problem (Type 12, Code 0)	Parameter Problem (Type 4, Code 1 ou Code 0)

Tableau 2.08 : *Messages ICMPv4 et leur équivalent ICMPv6*

2.7 Neighbor Discovery

Neighbor Discovery (ND) est un ensemble de messages et de processus qui déterminent le rapport entre les nœuds voisins. Il remplace ARP, l'ICMP Router Discovery et l'ICMP Redirect Message utilisé dans l'ICMPv4. [11]

Les voisins sont les nœuds situés sur le même domaine de broadcast Ethernet ou sur le même lien série (PPP, Frame Relay ou ATM).

Derrière le terme ND (Neighbor Discovery), se cache une série de procédures qui sont :

- utilisées par les stations pour la découverte des routeurs, des préfixes réseaux et des paramètres utilisés (MTU, nombre maximal de sauts,…) ;
- utilisées par les routeurs pour la fonction de redirection vers une meilleure route ;
- utilisées par les nœuds en général pour la résolution d'adresses, la vérification de l'état des voisins et la recherche du prochain saut pour aller vers un réseau donné.

Toutes ces procédures reposent sur 5 messages ICMP, contenant un en-tête de message ND et éventuellement des options.

Le protocole Neighbor Discovery utilise la même structure de message que ICMPv6 (figure 2.16) et les 5 types de messages sont :

- Router Sollicitation : envoyé par une machine pour découvrir le routeur du lien ;
- Router Advertisement : réponse à un Router Sollicitation ;
- Neighbor Sollicitation : envoyé par une machine pour la résolution d'adresse ;
- Neighbor Advertisement : réponse à un Neighbor Sollicitation ;
- Redirect : utilisé par un routeur pour informer un meilleur chemin.

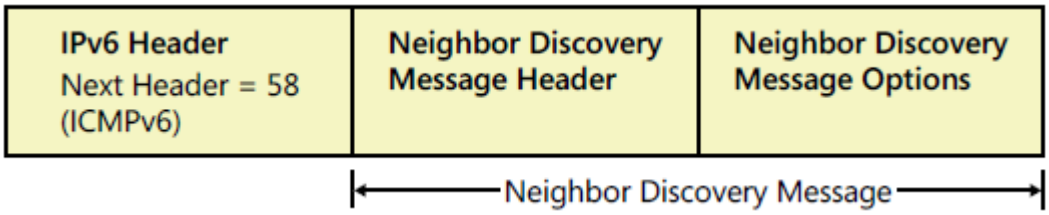

Figure 2.16 : *Format d'un message Neighbor Discovery*

Il existe également une correspondance entre ARP et ND comme nous le montre le tableau 2.09.

IPv4	IPv6
ARP Request Message	Neighbor Sollicitation Message
ARP Reply Message	Neighbor Advertisement
ARP Cache	Neighbor Cache
Gratuitous ARP	Duplicate Address Detection
Router Sollicitation Message (optionnel)	Router Sollicitation Message (requis)
Router Advertisement Message (optionnel)	Router Advertisement Message (requis)
Redirect Message	Redirect message

Tableau 2.09 : *Messages ARP et leur équivalent ND*

2.7.1 Résolution d'adresse

La procédure est relativement simple. Un nœud désirant connaître l'adresse MAC à partir de son adresse IP, émet une requête multicast dans le groupe de sollicitation de nœuds (Solicited Node) correspondant à cette même adresse IP.

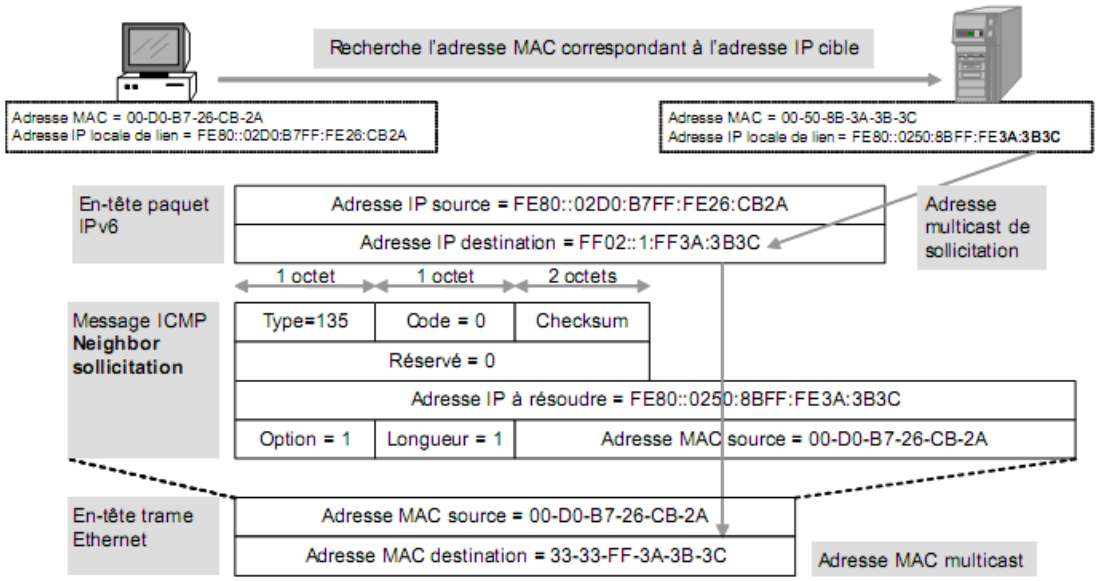

Figure 2.17 : *Recherche d'adresse MAC à l'adresse cible*

Seuls les nœuds dont l'adresse MAC se termine par les 24 derniers bits de l'adresse MAC de la trame multicast vérifient si l'adresse IP demandée correspond à la leur, et seul le nœud qui répond par l'affirmative renvoie une annonce à l'émetteur.

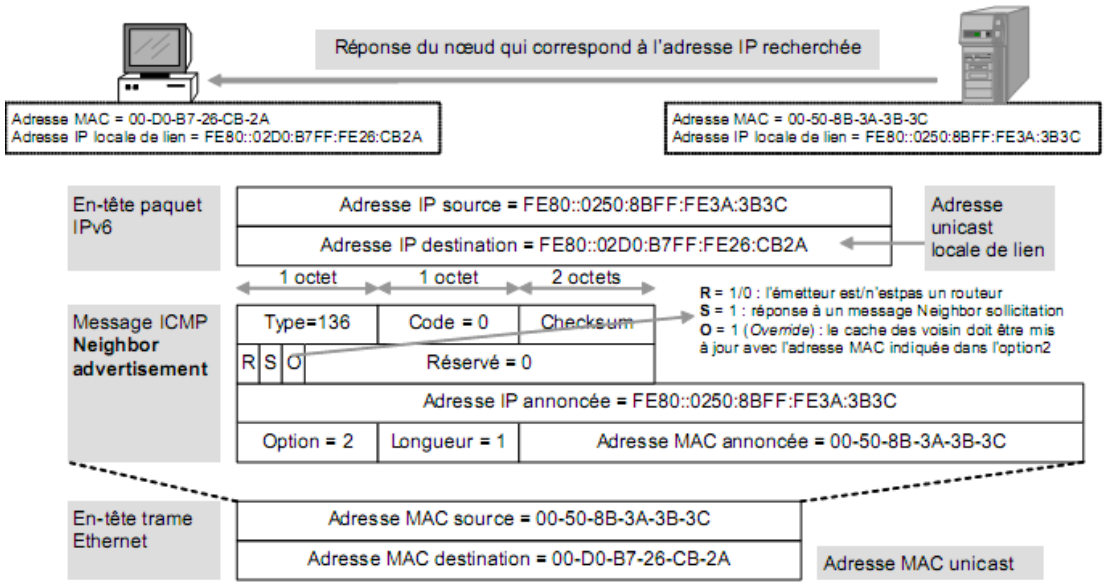

Figure 2.18 : *Recherche d'adresse MAC à l'adresse cible*

Les informations ainsi collectées permettent de mettre à jour le cache des voisins.

2.7.2 Découverte des routeurs

À son initialisation, tout nœud IPv6 recherche l'existence de routeurs sur son ou ses liens afin de déterminer s'il faut récupérer des préfixes et des paramètres.

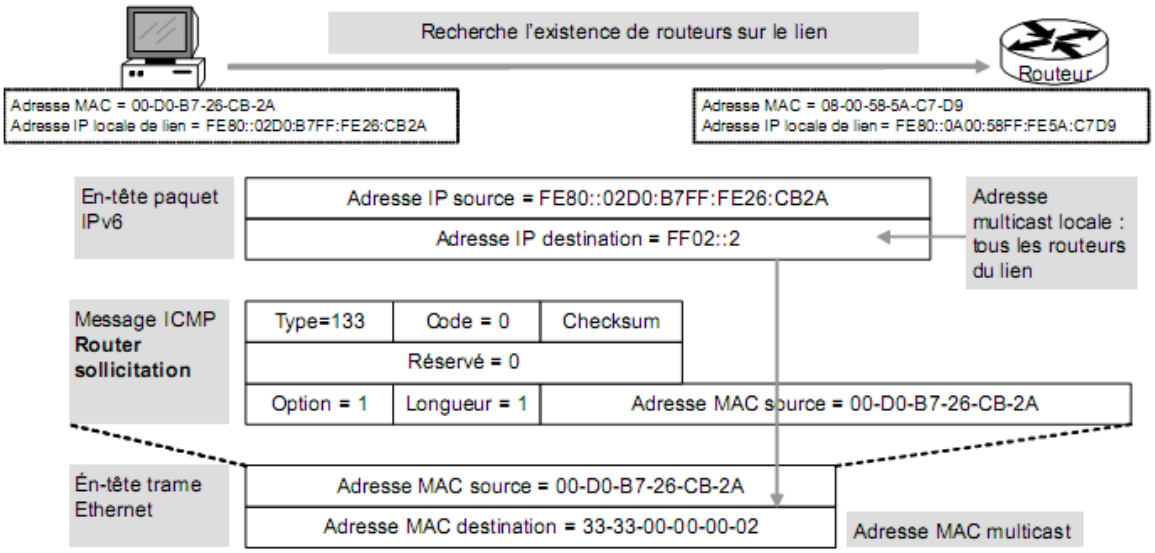

Figure 2.19 : *Recherche de l'existence de routeurs sur le lien*

Les routeurs envoient périodiquement des annonces et répondent également aux sollicitations comme celle montrée sur la figure 2.20.

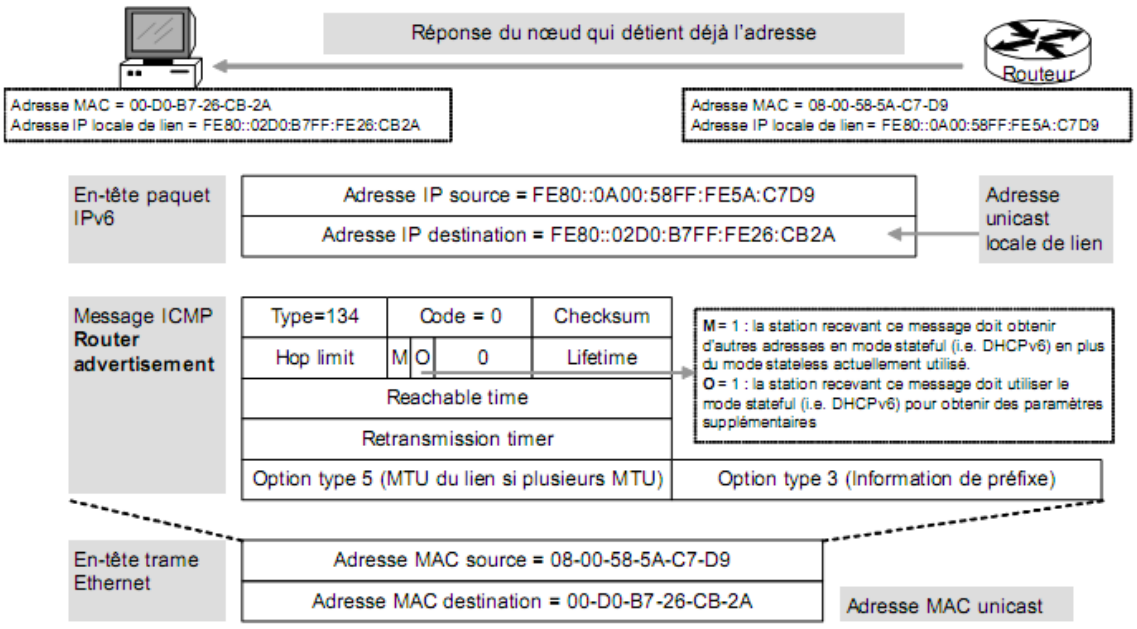

Figure 2.20 : *Réponse du nœud qui détient l'adresse*

Dans ce message, le routeur communique à la station une série de paramètres précisés dans le tableau 2.10.

Paramètre	Signification
Hop limit	Nombre limite de sauts : valeur que la station doit utiliser dans les paquets qu'elle émet.
Lifetime	Durée, en ms, pendant laquelle le routeur peut être considéré comme routeur par défaut. Une durée de 0 ms indique que le routeur ne peut pas être considéré comme routeur par défaut.
Reachable time	Durée, en ms, pendant laquelle un nœud voisin peut être considéré comme joignable après avoir reçu une confirmation.
Retransmission timer	Délai, en ms, entre les envois de message de sollicitation de voisins.
MTU	Dans le cas de réseaux commutés avec différents supports (Ethernet v2 et 802.3, FDDI, Token-Ring…).
Information de préfixe	Préfixe annoncé par le routeur.

Tableau 2.10 : *Paramètres envoyés par le routeur*

2.7.3 Vérification de l'état des voisins

Une fois les voisins découverts, un nœud vérifie périodiquement leur état et en garde une trace dans le cache des voisins.

État	Signification
Incomplete	La résolution d'adresse est en cours.
Reachable	Le nœud a été récemment confirmé comme joignable.
Stale	Le nœud n'est apparemment pas joignable, mais son état ne sera pas vérifié avant que des données doivent lui être envoyées.
Delay	Le nœud n'est apparemment pas joignable alors que des données lui ont été envoyées. Attend un certain délai avant de lancer une vérification.
Probe	Le nœud n'est apparemment pas joignable, et une vérification est en cours.

Tableau 2.11 : *Etat des voisins*

Si le voisin est le routeur par défaut et vient à ne plus être joignable, le nœud recherche un autre routeur dans sa liste de routeurs.

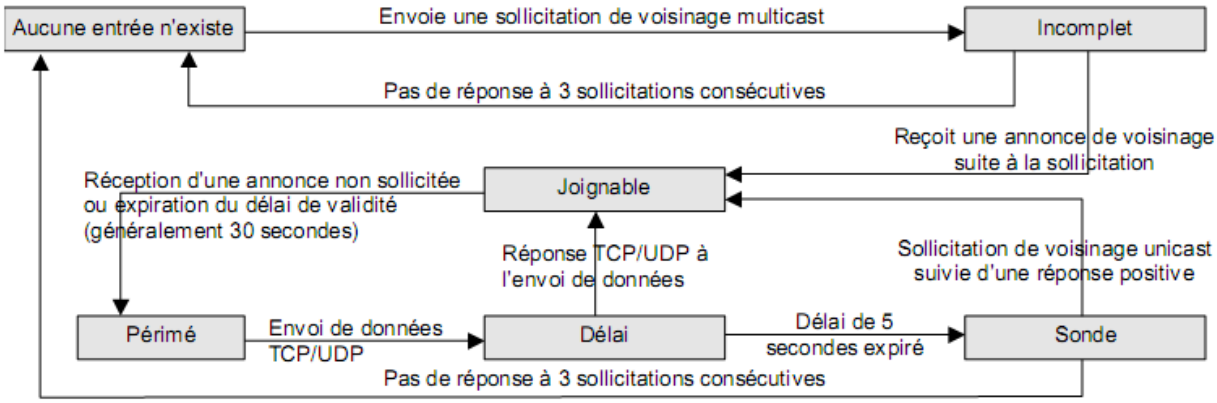

Figure 2.21 : *Vérification de l'état des voisins*

2.7.4 Redirection (Redirect)

Un routeur a la possibilité d'informer une station qu'il existe un routeur voisin disposant d'une meilleure route vers la destination spécifiée dans les paquets émis par la station.

De la même manière, il peut informer une station que son destinataire est directement joignable sans passer par un quelconque routeur.

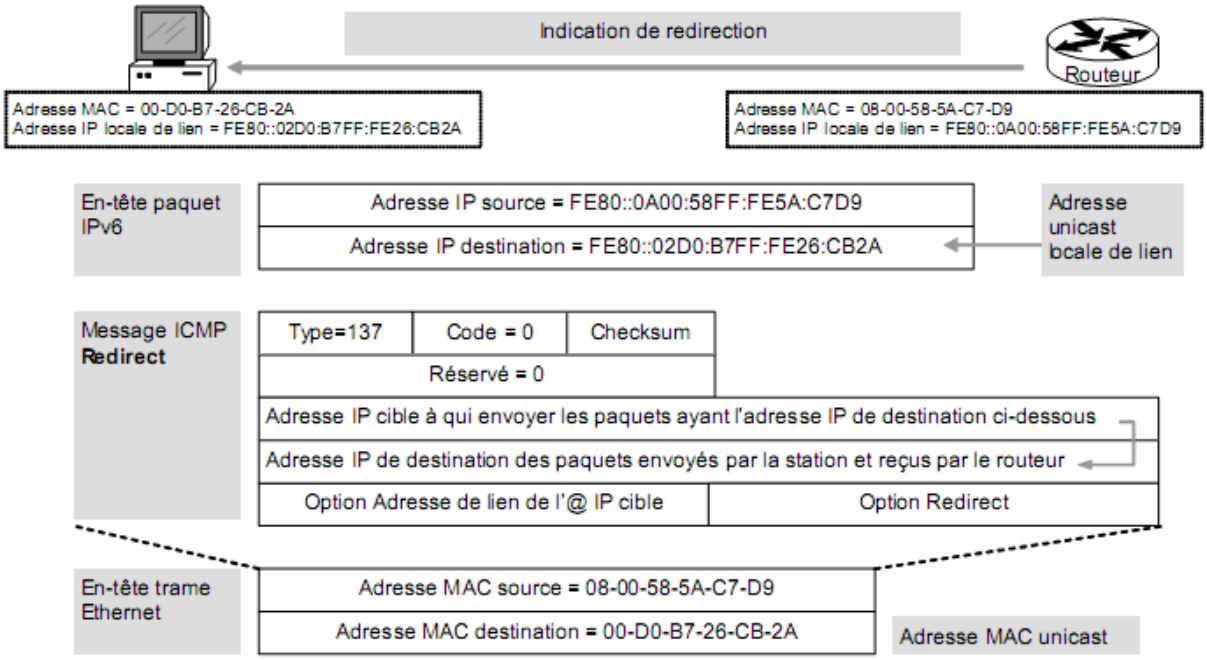

Figure 2.22 : *Indication de redirection envoyée*

À la réception de ce message, la station envoie les paquets à l'adresse MAC qui lui est indiquée.

2.8 Autoconfiguration des adresses

Lors de son initialisation, une interface IPv6 s'attribue une adresse de lien sur la base du préfixe FE80::/64 et de son adresse MAC dans le cas d'Ethernet. Des adresses supplémentaires sont créées si des routeurs lui envoient des annonces de préfixe. [11]

Cette procédure d'affectation automatique des adresses est connue sous le nom de *Stateless* par opposition au mode *Stateful* qui fait intervenir un serveur DHCP.

2.8.1 Affectation automatique des adresses

Le nœud fait appel à la procédure de détection des adresses dupliquées. En cas de détection positive, l'interface ne peut être activée. Dans le cas contraire, l'adresse est enregistrée.

Dans le cas d'une station, la procédure se poursuit par la recherche des routeurs voisins, et en cas de succès, la station sauvegarde les paramètres « Hop limit », « Reachable time », « Retransmission timer », et le MTU. [11]

Pour chaque préfixe indiqué dans l'annonce du routeur, la procédure d'autoconfiguration de la station se poursuit comme suit :

- Si l'indicateur « On-Link » de l'option d'information de préfixe est positionné à « 1 », le préfixe est ajouté à la liste des préfixes. [7]
- Si l'indicateur « Autonomous » de l'option d'information de préfixe est positionné à « 1 », le préfixe et l'adresse MAC sont concaténés pour former une adresse, dont l'unicité est vérifiée à l'aide de la procédure de détection des adresses dupliquées. [7] [11]
- Si l'indicateur « Managed address » est positionné à « 1 » dans le message d'annonce du routeur, un protocole tel que DHCP doit être utilisé pour obtenir d'autres adresses. [7]
- Si l'indicateur « Other stateful » est positionné à « 1 » dans le message d'annonce du routeur, un protocole tel que DHCP doit être utilisé pour obtenir d'autres paramètres. [7]

Les adresses IPv6 sont donc allouées dynamiquement, une de lien local, et d'autres, si des routeurs voisins sont détectés.

2.8.2 Détection d'adresse dupliquée

Il s'agit du cas hautement improbable d'une duplication d'adresse, sans doute suite à une mauvaise configuration lorsque, par exemple, l'administrateur a voulu fixer l'adresse IP indépendamment de l'adresse MAC de la carte réseau.

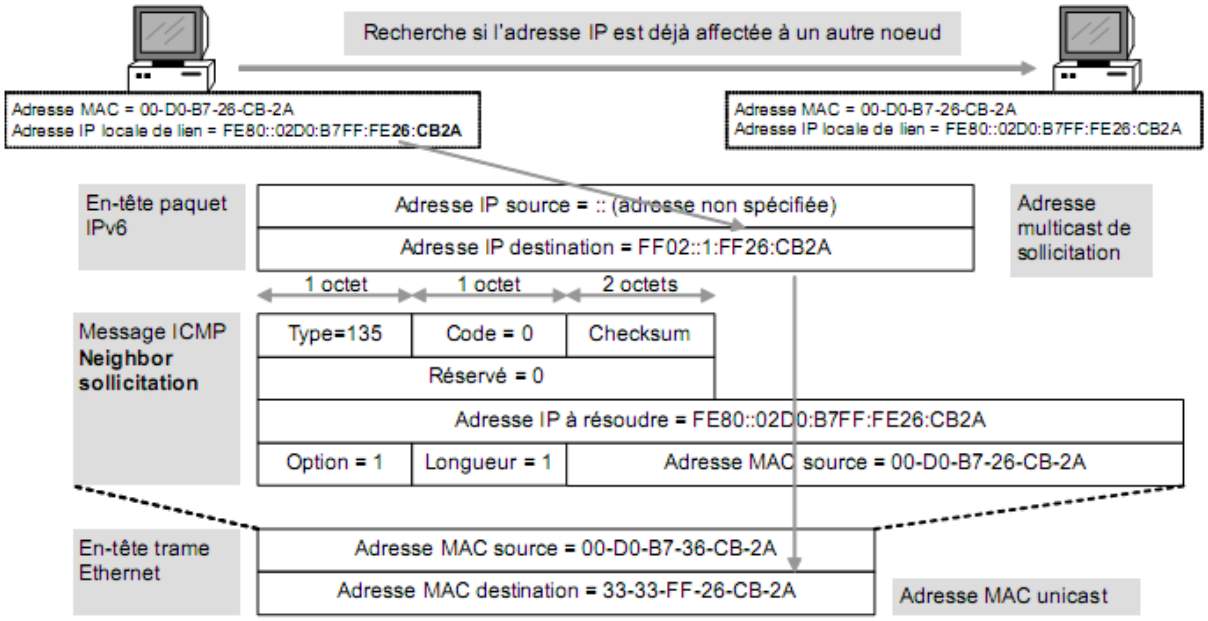

Figure 2.23 : *Détection d'adresse dupliquée*

Si aucune réponse n'est reçue, l'adresse est supposée être libre et est affectée à l'interface.

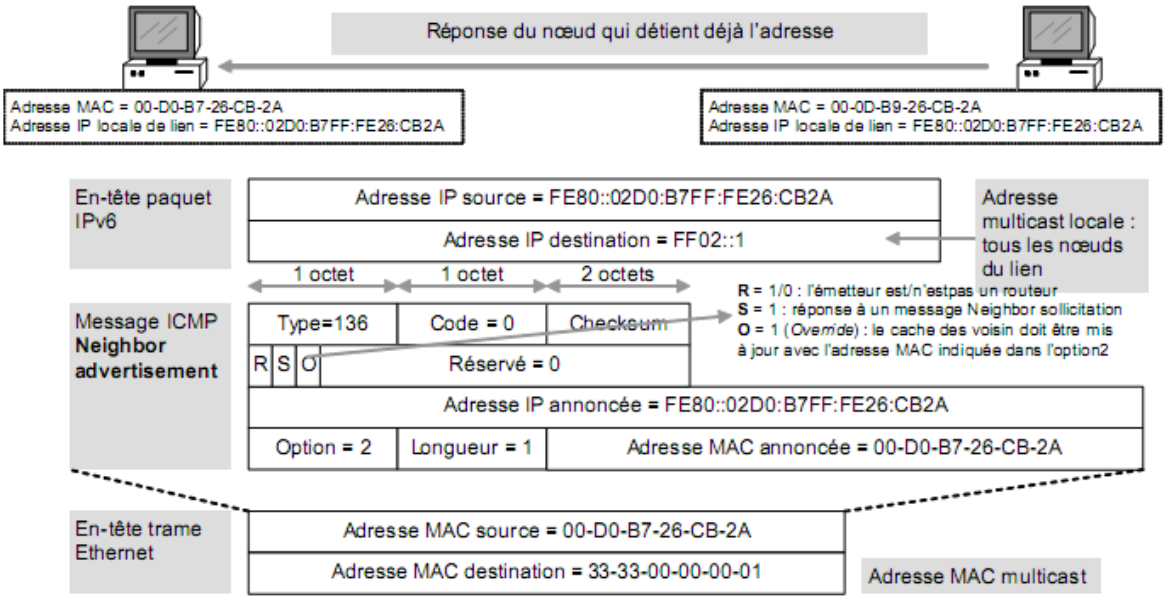

Figure 2.24 : *Réponse d'un nœud qui détient l'adresse*

2.9 DHCPv6

Avec l'autoconfiguration avec état, un hôte IPv6 obtient les adresses et/ou d'autres informations de configuration par l'intermédiaire d'un serveur. Ce mécanisme est bâti sur le modèle client/serveur et repose sur l'utilisation du protocole DHCPv6. Au niveau du serveur, une table de correspondance est maintenue afin de distinguer les adresses utilisées de celles qui ne le sont pas. [16]
Voici ci-dessous un exemple d'échange d'informations entre un client et un serveur :

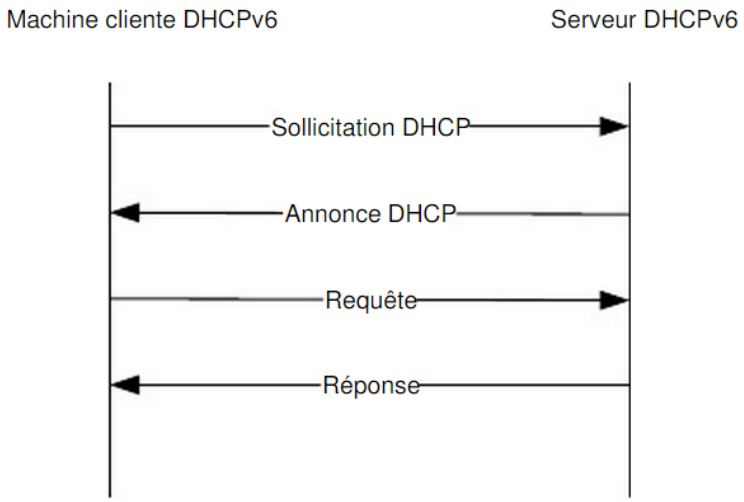

Figure 2.25 : *Echange entre client et serveur DHCPv6*

Le client envoie les requêtes vers le port 547 et recevra les réponses par le port 546, les échanges reposent sur UDP et la fiabilité est assurée par le fait que le client répète ses messages jusqu'à obtenir une réponse en attendant un timeout.

Le client envoie en premier lieu une sollicitation vers l'adresse FF02::1:2 qui est l'adresse multicast de tous les agents DHCP, en effet dans le cas où le serveur n'est pas sur le même segment que le client, on peut utiliser des relais, l'adresse multicast FF02::1:2 concerne les serveurs et les relais. Le serveur répond et le client peut faire une requête en spécifiant les options souhaitées et enfin, le serveur répond avec les options. [16]

Nous pouvons réfléchir à l'utilité de DHCPv6 en nous basant sur le principe que IPv6 dispose de beaucoup d'adresses, de tellement d'adresses qu'il est nécessaire de changer complètement les habitudes prises avec IPv4 d'économiser des adresses, d'optimiser au mieux le plan d'adressage. DHCP pour IPv4 a été une des techniques utilisée pour pallier à la pénurie d'IPv4 ; avec IPv6 l'objectif n'est pas le même, on peut envisager que les adresses distribuées peuvent être gardées de manière pérenne par les clients.

2.10 IPv6 et résolution des noms

2.10.1 Nommage direct : du nom vers les adresses

La correspondance entre un nom de domaine et son (ou ses) adresse(s) IPv4 est réalisée en associant au nom en question un ou plusieurs enregistrements DNS de type A. Chaque enregistrement contient une valeur qui est une adresse IPv4. [1]

De manière analogue à l'enregistrement A, le nouveau type d'enregistrement AAAA (appelé « Quad A ») défini pour IPv6, permet d'établir la correspondance entre un nom de domaine et son (ou une de ses) adresse(s) IPv6. Une machine ayant plusieurs adresses IPv6 globales a en principe autant d'enregistrements AAAA publiés dans le DNS. Une requête DNS de type AAAA concernant une machine particulière retourne dans ce cas tous les enregistrements AAAA publiés dans le DNS et correspondant à cette machine. [7] [11]

Le format textuel d'un enregistrement AAAA tel qu'il apparaît dans le fichier de zone DNS est le suivant : *Nom IN AAA Adresse*

Exemple : host1.microsoft.com IN AAAA 2001:DB8::1:DD48:AB34:D07C:3914

2.10.2 Nommage inverse : de l'adresse vers les noms

L'enregistrement de type PTR (Pointer Records), stocké sous l'arbre DNS inverse in-addr.arpa, permet d'établir la correspondance entre une adresse IPv4 et un (ou plusieurs) nom(s). C'est ce même type d'enregistrement PTR, qui, stocké sous l'arbre DNS inverse ip6.arpa, permet de mettre en correspondance une adresse IPv6 avec un ou plusieurs noms de domaines. [11]

Une adresse IPv6 est transformée en un nom de domaine publié sous l'arborescence inverse ip6.arpa de la manière suivante : les 32 demi-octets formant l'adresse IPv6 sont séparés par le caractère « **.** » et concaténés dans l'ordre inverse au suffixe ip6.arpa. [11]

Par exemple l'adresse 2001:660:3006:1::1:1 est transformée en le nom de domaine inverse suivant: 1.0.0.0.1.0.0.0.0.0.0.0.0.0.0.0.1.0.0.0.6.0.0.3.0.6.6.0.1.0.0.2.ip6.arpa

2.11 Découverte du MTU

Le MTU (Maximum Transmission Unit) est la taille maximale d'un paquet IP non fragmenté pouvant être transporté par la couche liaison. Étant donné qu'un paquet IP peut traverser des réseaux de nature très différente, le MTU résultant doit correspondre au plus petit MTU trouvé. Par exemple,

le MTU d'une trame Ethernet v2 est de 1 500 octets, celui d'une trame Ethernet 802.3, de 1 492 octets et celui d'une trame Frame Relay, de 4 096 octets.

La procédure utilisée est la suivante :

- Initialement, le MTU correspond à celui de la couche liaison de l'interface par laquelle le premier paquet est envoyé.

- Si un routeur constate que la taille du paquet est supérieure au MTU de la couche liaison de l'interface vers laquelle il doit transmettre le paquet, il renvoie à l'émetteur un message ICMP « Packet too big » qui contient le MTU correspondant à la couche liaison en cause.

- L'émetteur ajuste son MTU et renvoie le paquet en le fragmentant éventuellement. Il procède ainsi de suite, tant qu'un routeur situé sur la route d'acheminement du paquet lui envoie le même type d'erreur.

- Toutes les cinq ou dix minutes, l'émetteur essaye d'augmenter son MTU, afin de détecter un éventuel changement dans la topologie du réseau. En cas d'échec, il reçoit les mêmes messages d'erreur.

2.12 Gestion des groupes de diffusion

La procédure MLD (Multicast Listener Discovery) permet d'enregistrer les stations appartenant à un groupe multicast [7]. Elle est identique à IGMP employé avec IPv4, mais utilise des messages ICMP équivalents.

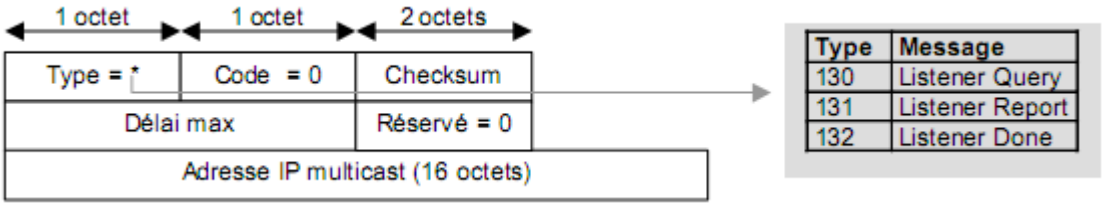

Figure 2.26 : *Message MLD*

Les messages MLD sont constitués d'un en-tête IP, d'une extension « Hop-by-hop » comportant l'option « Router Alert », et d'un des trois messages ICMP ci-dessous.

Les messages « Listener Query » sont envoyés par les routeurs pour rechercher les membres d'un groupe (recherche spécifique à une adresse multicast) ou les membres de tous les groupes (recherche générale). [13]

Les membres des groupes répondent à cette sollicitation ou se manifestent d'eux-mêmes en envoyant un message « Listener Report ». Dès que l'un d'eux quitte le groupe, il envoie un message « Listener Done ». [13]

L'option « Router Alert », décrite par la RFC 2711, indique au routeur de traiter le message MLD, même s'il n'est pas à l'écoute de l'adresse multicast parce que n'ayant aucun membre pour ce groupe. [13]

2.13 Routage en IPv6

Les algorithmes de routage n'ont pas beaucoup changé avec IPv6. Les modifications consistent essentiellement à les adapter au nouveau format des adresses IPv6. Et les protocoles de routage IPv6 profitent des propriétés de multicast et d'authentification.

Une conséquence de l'apparition du routage IPv6 est que les équipements doivent alors prendre en compte les deux piles de protocoles, IPv4 et IPv6. Cela doit être pris en considération lors de l'activation des protocoles de routage.

Comme dans IPv4, il faut faire la distinction entre deux grandes familles de protocoles de routage : les protocoles de routage internes (IGP : Interior Gateway Protocols) et externes (EGP : Exterior Gateway Protocols). C'est la notion de système autonome ou Autonomous System (AS) qui permet de faire la différence en définissant la portée des échanges d'informations de routage effectuée par ces protocoles de routage. Ainsi, la propagation des préfixes internes à un AS se fait par un IGP, alors que les annonces de préfixes entre AS se fait par un EGP.

Il ne sera pas détaillé ici le fonctionnement de tel ou tel protocole, mais plutôt les changements qui ont été nécessaires afin de prendre en compte la technologie IPv6 dans les protocoles de routage existants pour IPv4. Ce paragraphe traite les trois protocoles IGP suivants : RIPng (équivalent de RIPv2 pour IPv4), ISIS et OSPFv3 (équivalent de OSPFv2 pour IPv4), ainsi que du protocole de routage externe BGP.

2.13.1 RIPng

C'est le premier protocole de routage dynamique proposé par IPv6 par le RFC 2080.

RIPng est une simple extension à IPv6 du protocole RIPv2 d'IPv4. Il en hérite les mêmes caractéristiques à savoir le nombre maximum de saut qui est de 15 et les mises à jour qui sont régulières. Les paquets RIPv2 et RIPng sont identiques, seule la fonction d'authentification a disparu. Elle est en effet inutile car RIPng peut s'appuyer sur les mécanismes de sécurité IPsec disponibles en IPv6. [11]

2.13.2 IS-IS

Ce protocole de routage n'a pas subi beaucoup de modification dans le nouveau protocole. En effet, IS-IS est un protocole de routage interne à état de lien. C'est un protocole de niveau 3 qui s'appuie sur une couche 2 de type Ethernet 802.2. Cet élément mérite d'être signalé car cela rend ce protocole indépendant d'IP que ce soit IPv4 ou IPv6. [11]

2.13.3 OSPFv3

Le troisième protocole de routage interne, basé sur l'algorithme du plus court chemin, s'appelle OSPF (Open Shortest Path First). Relativement plus difficile à mettre en œuvre que RIPng. Il est plus efficace dans les détections et la suppression des boucles dans les phases transitoires.

OSPF a été adapté à IPv6 par le RFC 2740 et est passé de la version 2 à la version 3. La plupart des algorithmes implémentés dans OSPF2 ont été réutilisés ; bien évidemment, certains changements ont été nécessaires en vue de l'adaptation aux fonctionnalités d'IPv6.

OSPF utilise les adresses lien-local pour l'échange sur les liens. Le mécanisme de sécurité d'OSPFv2 a été remplacé par les mécanismes d'authentification et de confidentialité d'IPsec. [11]

2.13.4 BGP

BGP4 est le protocole de routage externe actuellement utilisé pour le routage global de l'Internet IPv4. Ce protocole est l'objet d'évolutions constantes. L'une d'entre elles est le RFC 2858 qui rend BGP4 multi-protocole en introduisant la notion de famille d'adresse (exemple : IPv4, IPv6, IPX...) et de sous-famille d'adresse (exemple : unicast, multicast). Le RFC 2545 précise l'usage des extensions multi-protocoles pour le cas d'IPv6. [11] [13]

Pour faire référence à leur capacité de traitement des routes multicast, ce protocole de routage est souvent appelé MBGP (Multicast BGP) ou BGP4+ pour faire référence à leur capacité de traitement de routes IPv6. [11]

2.14 Conclusion

Ainsi, le protocole IPv6 apporte de grande innovation par rapport à son prédécesseur IPv4. Non seulement au niveau de l'adressage et de l'en-tête, tout a été repensé pour principalement pallier aux limitations rencontrées en IPv4 mais tout en gardant les points forts de celui-ci. L'adresse Broadcast a été délaissée au profit de l'adresse Multicast, le protocole ARP au profit du Neighbor Discovery ou Découverte des voisins et les protocoles de routage ont changé de manière à les adapter au nouveau format de l'adresse IP incluant de nouvelles propriétés telles que l'authentification et le multicast pour ne citer que les plus marquants.

Certes, l'utilisation du protocole IPv6 ne peut être que bénéfique pour les utilisateurs finaux, les fournisseurs d'accès Internet, les ingénieurs, … bref toute la communauté d'Internet mais qu'en est-il de son implémentation ? Peut-on passer du jour au lendemain à l'IPv6 ? Une coexistence entre IPv4 et IPv6 est-elle possible ? Le troisième chapitre essayera de répondre à ces questions en présentant les technologies de transition de l'IPv4 vers l'IPv6.

CHAPITRE 3
TECHNOLOGIES DE TRANSITION IPv4 VERS IPv6

3.1 Introduction

Les transitions d'un protocole à un autre ne sont pas faciles. Et la transition du protocole IPv4 vers le protocole IPv6 ne fait pas exception. L'étape de standardisation des protocoles de base d'IPv6 étant achevée, le développement de techniques assurant la transition devient un point clé dans le déploiement à grande échelle du protocole. Pour certaines catégories d'applications comme le mail, le web, le succès d'IPv6 est fortement lié à l'interopérabilité avec IPv4 puisque jusqu'à présent la majorité des informations et des utilisateurs ne sont accessibles qu'avec cette version du protocole. Il faut donc pouvoir amorcer dès maintenant une transition qui permettra de passer à IPv6.

La phase de transition doit être simple, ou au minimum moins compliquée qu'une utilisation prolongée d'IPv4. Il n'y a pas de jour J pour le passage d'IPv4 à IPv6, il n'y a également pas d'échéance pour la disparition du protocole IPv4. Les raisons qui vont nécessité ce passage vont être fortement liées à la pénurie d'adresses pour les nouveaux réseaux et aux fonctionnalités de configuration automatique que requièrent un grand espace d'adressage.

Cette transition sous-entend bien évidement une coexistence entre les parties prenantes à savoir l'IPv4 et l'IPv6 et même si l'objectif final est de basculer entièrement en IPv6, les technologies de transition qui seront présentées dans ce troisième chapitre ne sont que des solutions à court terme pour permettre ce basculement.

Pour coexister avec une infrastructure IPv4 et pour passer à une éventuelle migration en une infrastructure tout IPv6 (full IPv6 ou IPv6-only), les standards de transition en IPv6 définissent les mécanismes de transition suivants :

- Le tunneling IPv6 sur IPv4 ou IPv6-over-IPv4 tunneling
- La double pile ou Dual Stack
- La translation d'adresses

D'autres solutions de transition existent également comme le cas du MPLS (Multi Protocol Label Switching).

3.2 Tunneling

Le tunneling IPv6 sur IPv4 ou IPv6-over-IPv4 tunneling est l'encapsulation des paquets IPv6 avec un en-tête IPv4 de sorte que les paquets IPv6 puissent être envoyés sur une infrastructure IPv4. [7]

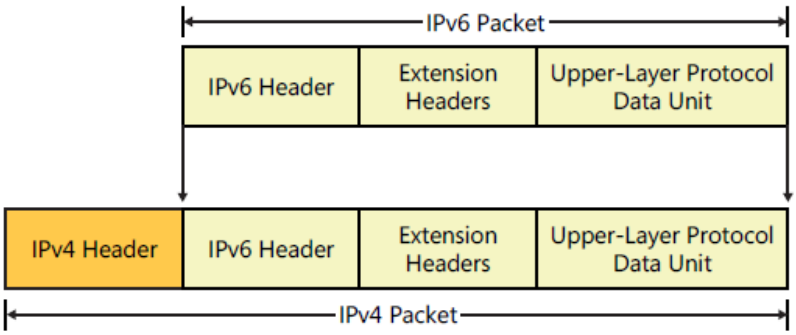

Figure 3.01 : *IPv6-over-IPv4 tunneling*

3.2.2 Types de configurations

Le RFC 4213 définit les configurations de tunneling suivantes pour encapsuler le trafic IPv6 entre les nœuds IPv6/IPv4 sur une infrastructure IPv4 :

- Routeur à routeur ou Router-to-router
- Hôte à routeur et routeur à hôte ou Host-to-router and router-to-host
- Hôte à hôte ou host-to-host

3.2.2.1 Routeur à routeur

Dans la configuration d'un tunnel routeur à routeur, deux routeurs IPv6/IPv4 connectent 2 réseaux IPv6 sur une infrastructure IPv4. [7] [14]

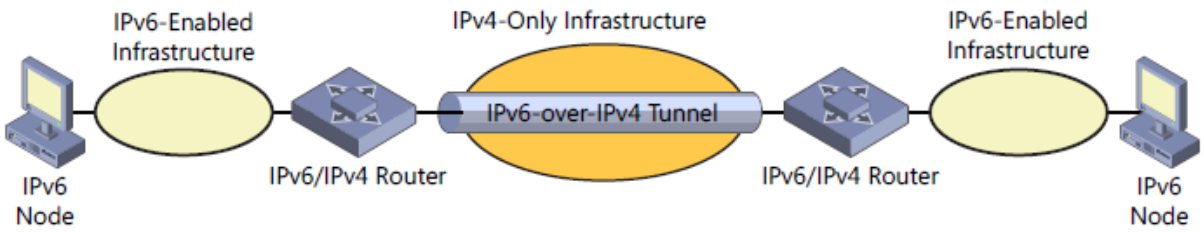

Figure 3.02 : *Router-to-router tunneling*

3.2.2.2 Hôte à routeur et routeur à hôte

Dans la configuration d'un tunnel hôte à routeur et routeur à hôte, un hôte IPv6/IPv4 qui réside dans une infrastructure IPv4 utilise un tunnel IPv6-over-IPv4 pour joindre un routeur IPv6/IPv4. [7] [14]

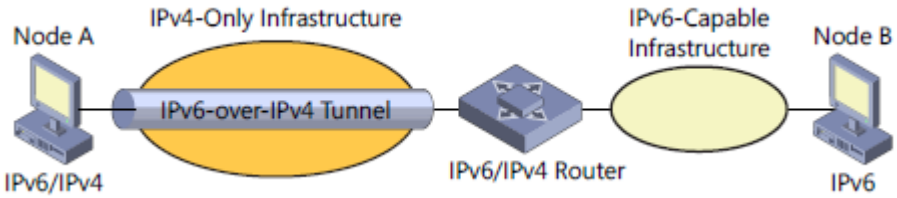

Figure 3.03 : *Host-to-router et router-to-host tunneling*

3.2.2.3 Hôte à hôte ou host-to-host

Dans la configuration d'un tunnel hôte à hôte, un nœud IPv6/IPv4 qui réside dans une infrastructure IPv4 utilise un tunnel IPv6-over-IPv4 pour joindre d'autres nœuds IPv6/IPv4 qui résident dans la même infrastructure IPv4. [7] [14]

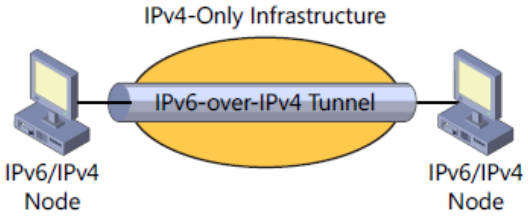

Figure 3.04 : *Host-to-host tunneling*

3.2.3 Types de tunnel

On distingue 2 types de tunnel d'après le RFC 4213 :

- Les tunnels configurés ou manuels
- Les tunnels automatiques

Les tunnels configurés requièrent une configuration manuelle du tunnel local et distant. Pour créer manuellement un tunnel IPv6 sur Windows, on utilise la commande suivante :
netsh interface ipv6 add v6v4tunnel [interface=]Name [localaddress=]LocalIPv4Address [remoteaddress=] RemoteIPv4address [7]

- *Name* est le nom de la nouvelle interface du tunnel.
- *LocalIPv4Address* est une adresse IPv4 de l'ordinateur et correspond au tunnel local.
- *RemoteIPv4address* est l'adresse IPv4 du tunnel distant.

Il faut créer des interfaces de tunneling sur les routeurs dans les deux côtés du tunnel et ajouter des routes qui utilisent ces interfaces.

La figure 3.05 illustre un exemple de configuration manuelle d'un tunnel et juste en-dessous les commandes effectuées sur le routeur R1 et le routeur R2.

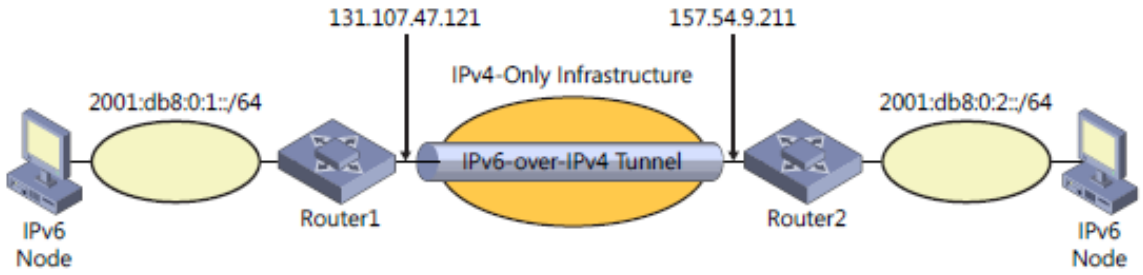

Figure 3.05 : *Exemple de configuration manuelle d'un tunnel*

Commandes sur R1 :

netsh interface ipv6 add v6v4tunnel TunnelTo2 131.107.47.121 157.54.9.211

netsh interface ipv6 add route 2001:db8:0:2::/64 TunnelTo2

Commandes sur R2 :

netsh interface ipv6 add v6v4tunnel TunnelTo1 157.54.9.211 131.107.47.121

netsh interface ipv6 add route 2001:db8:0:1::/64 TunnelTo1

Pour le cas du tunnel automatique, c'est un tunnel qui ne requière pas de configuration manuelle. On distingue 3 types de tunnels automatiques :

- 6to4
- ISATAP (Intra-Site Automatic Tunnel Addressing Protocol)
- Teredo

3.2.3.2 6to4

- ***Principe***

C'est une technologie de tunneling définie dans le RFC 3056 qui est utilisée pour fournir une connexion unicast IPv6 entre des sites IPv6 à travers l'Internet IPv4. La figure 3.06 nous montre la structure d'une adresse 6to4. [7]

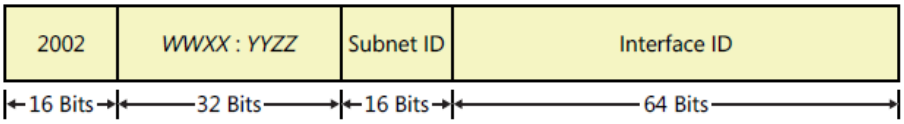

Figure 3.06 : *Structure d'une adresse 6to4*

L'adresse 6to4 est constituée des éléments suivants :

- 2002::/16 est l'espace d'adressage réservé à 6to4 ;
- WWXX:YYZZ est la représentation hexadécimale d'une adresse IPv4 publique (w.x.y.z) attribuée à un site ou hôte sur Internet ;
- le préfixe de sous-réseau (/16) et l'identifiant d'interface (/64).

Le protocole IPv6 pour Windows attribue automatiquement à l'interface de tunneling 6to4 l'adresse : 2002:WWXX:YYZZ::WWXX:YYZZ, dans lequel WWXX:YYZZ est la notation hexadécimale d'une adresse IPv4 publique attribuée à une interface de l'ordinateur. S'il y a de multiples adresses publiques IPv4, l'interface de tunneling 6to4 affectera plusieurs adresses de la même forme. [7]

Par exemple, pour un ordinateur exécutant Windows 8 qui a une adresse IPv4 publique 131.107.0.1, IPv6 affecte l'adresse 2002:836B:1::836B:1 à l'interface de tunneling 6to4. Ce type particulier d'adresse 6to4, qui utilise un Subnet ID de 0 et une Interface ID ::WWXX:YYZZ, est utilisé pour des routeurs 6to4 Windows.

- ***Exemple 6to4 Tunneling***

Un hôte A exécutant Windows 8 est configuré avec l'adresse IPv4 publique 131.107.0.1. Un autre hôte B exécutant Windows 8 est configuré avec l'adresse IPv4 157.54.0.1. L'IPv6 sur l'hôte A se configure automatiquement l'adresse 6to4 :

2002:836B:1::836B:1 sur son interface de tunneling 6to4, et l'hôte B se configure automatiquement une adresse 6to4 : 2002:9D36:1::9D36:1 également sur son interface de tunneling 6to4. Les deux hôtes A et B sont directement connectés à l'Internet IPv4 comme le montre la figure 3.07. [7]

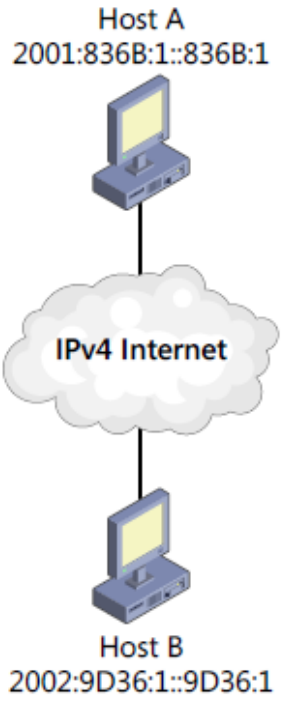

Figure 3.07 : *Exemple de configuration 6to4*

Lorsque l'hôte A envoie un trafic IPv6 à l'hôte B destiné à accueillir l'adresse 6to4 de B, la source et la destination des adresses pour les en-têtes IPv6 et IPv4 sont énumérées dans le tableau 3.01. Pour tester la connectivité entre les hôtes 6to4, il nous suffit d'utiliser l'outil Ping.

Champ	**Valeur**
Adresse Source IPv6	2002:836B:1::836B:1
Adresse Destination IPv6	2002:9D36:1::9D36:1
Adresse Source IPv4	131.107.0.1
Adresse Destination IPv4	157.54.0.1

Tableau 3.01: *Exemple d'adresse 6to4*

– ***Les composants 6to4***

Un réseau 6to4 est composé d'hôtes 6to4, de routeurs 6to4, d'hôtes/routeurs 6to4 et de relais 6to4. Les composants représentés sur la figure 3.08 sont les suivants :

- Hôtes 6to4 (*6to4 Host*) : hôte natif IPv6 qui est configuré avec au moins une adresse 6to4 (une adresse globale avec le préfixe 2002::/ 16). Les Hôtes 6to4 ne nécessitent pas de support ou manuelle de configuration et peuvent créer des adresses 6to4 en utilisant des mécanismes standards d'autoconfiguration d'adresse.

- Routeur 6to4 (*6to4 Router*) : Un routeur IPv6/IPv4 qui utilise une interface de tunneling 6to4 pour transmettre des données entre les hôtes 6to4 d'un site et d'autres routeurs 6to4 ou relais 6to4 sur Internet IPv4. Les routeurs 6to4 peuvent nécessiter une configuration manuelle.
- Relais 6to4 (*Relay 6to4*) : Un routeur IPv6/IPv4 qui transmet des données entre routeurs 6to4 et hôtes/routeurs 6to4 sur l'Internet IPv4 et les hôtes sur l'Internet IPv6.

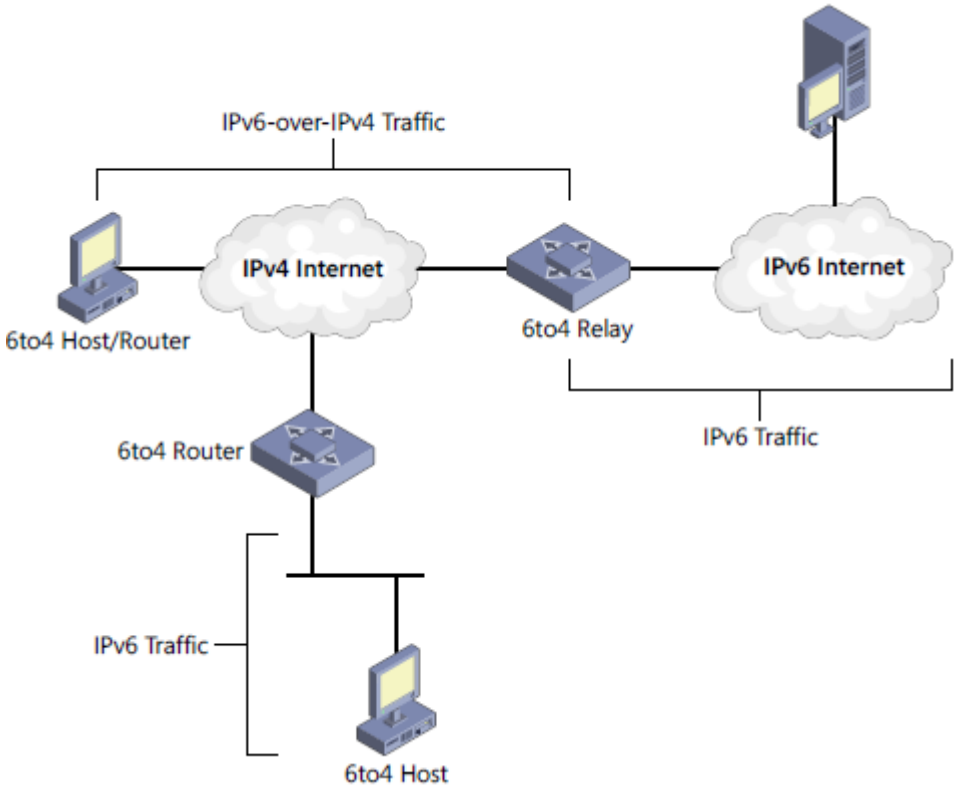

Figure 3.08 : *Composants 6to4 sur l'Internet IPv4 et IPv6*

3.2.3.3 ISATAP

ISATAP (Intra-Site Automatic Addressing Protocol) est conçu afin de permettre d'une part à un client Dual Stack isolé dans un LAN IPv4 d'atteindre le réseau IPv6 à l'extérieur du LAN, et d'autre part pour fournir une connexion unicast IPv6 entre des hôtes IPv6/IPv4 à travers un intranet IPv4.

Les adresses ISATAP ont l'un des formats suivants :

- *64-bitUnicastPrefix:0:5EFE:w.x.y.z*
- *64-bitUnicastPrefix:200:5EFE:w.x.y.z*

L'adresse ISATAP a donc :

- un préfixe 64 bits (*64-bitUnicastPrefix*) qui peut être un préfixe de n'importe quelle adresse unicast, incluant le lien local, le site local et le global unicast. [7]
- ::0:5EFE:w.x.y.z et ::200:5EFE:w.x.y.z sont les identifiants des interfaces locales. Pour ::0:5EFE:w.x.y.z, w.x.y.z est l'adresse IPv4 privée. Pour ::200:5EFE:w.x.y.z, w.x.y.z représente également l'adresse IPv4 privée. [7]

Par défaut, Le protocole IPv6 des machines Windows installe automatiquement des adresses link-local ISATAP (FE80::5EFE:w.x.y.z ou FE80::200:5EFE:w.x.y.z) sur l'interface du tunnel ISATAP pour les adresses IPv4 correspondant à l'interface du LAN.

- ***Exemple de tunneling ISATAP***

L'hôte A est configuré avec une adresse IPv4 de 10.40.1.29. L'hôte B est configuré avec une adresse IPv4 de 192.168.41.30. IPv6 dans l'hôte A l'adresse ISATAP FE80::5EFE:10.40.1.29 assignée à son interface de tunneling ISATAP, et l'hôte B a l'adresse ISATAP FE80::5EFE:192.168.41.30 sur son interface de tunneling ISATAP. La figure 3.09 nous montre cet exemple de configuration. [7]

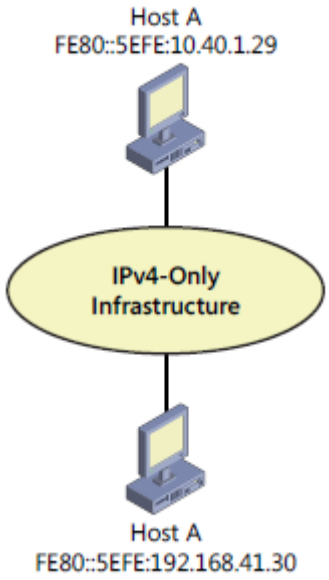

Figure 3.09 : *Exemple de configuration ISATAP*

3.2.3.4 Teredo

Microsoft a également lancé son propre protocole de tunnel automatique appelé *Teredo*. Teredo a été conçu pour pallier aux déficiences de 6to4. En effet, Teredo permet à un hôte relié à un réseau IPv4 de communiquer avec le reste du monde, sans routeur particulier sur son LAN, notamment derrière un réseau IPv4 NATé. [7]

Il utilise un tunnel UDP en IPv4, qui a donc la possibilité de traverser les passerelles NAT. En raison de l'utilisation d'UDP en plus d'IPv4 et d'IPv6, le paquet fait 8 octets de plus qu'en 6to4.

Les adresses Teredo ont le format suivant :

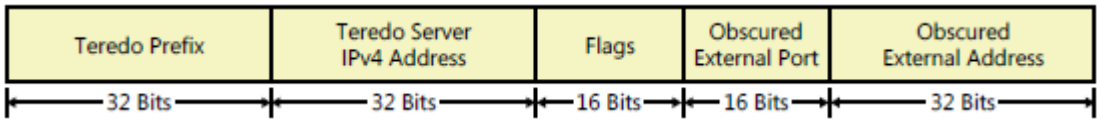

Figure 3.10 : *Format de l'adresse Teredo*

- Teredo prefix : les premiers 32 bits sont pour le préfixe Teredo, qui sont les mêmes pour toutes les adresses Teredo. Le préfixe Teredo définit dans le RFC 4380 est 2001::/32.
- Teredo server IPv4 Address : le champ 32 bits qui suit contient l'adresse IPv4 publique du serveur Teredo.
- Le champ flags dépend du type de NAT (0 pour un NAT d'IP source dynamique classique et 8000 si le NAT gère une association statique port externe/IP privée interne).
- Le champ port contient le numéro du port UDP. Le port UDP sur lequel le serveur et le relai Teredo écoutent est le port 3544.

– ***Composants Teredo***

Une infrastructure Teredo contient les composants suivants :

- Client Teredo ou *Teredo Client* : c'est un nœud IPv4/IPv6 qui supporte une interface de tunneling Teredo. Le client communique avec le serveur Teredo pour obtenir un préfixe d'adresse. [7]
- Serveur Teredo ou *Teredo Server* : c'est un nœud IPv4/IPv6 qui est connecté à l'Internet IPv4 et l'Internet IPv6. Le principal rôle du serveur est d'assister les clients dans la configuration des adresses et de faciliter la communication initiale entre les clients Teredo et d'autres clients Teredo ou entre des clients Teredo et des hôtes IPv6. Le serveur écoute par défaut sur le port UDP 3544. [7]
- Relai Teredo ou *Relay Teredo* : c'est un routeur IP6/IPv4 qui peut router les paquets entre les clients Teredo sur l'Internet IPv4 et les hôtes IPv6 sur l'Internet IPv6. Dans certain cas,

le relai Teredo interagit avec le serveur Teredo pour l'aider à faciliter la communication initiale entre des clients Teredo et des hôtes IPv6. [7]

- Relai hôte Teredo ou *Teredo Host Relay* : c'est un nœud IPv6/IPv4 qui a une interface et une connexion entre l'Internet IPv4 et l'Internet IPv6 et peut communiquer directement avec les clients Teredo sur l'Internet IPv4 sans l'intermédiaire d'un relai Teredo. [7]

La figure suivante nous montre ces composants dans une infrastructure Teredo.

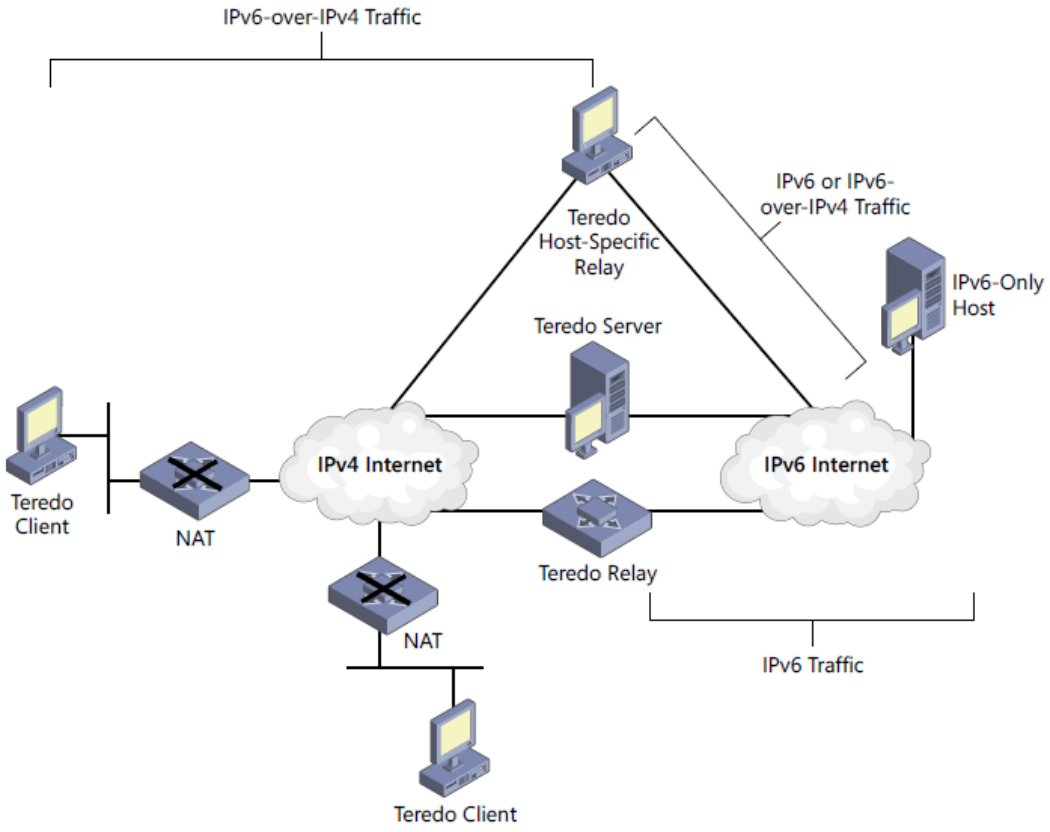

Figure 3.11 : *Infrastructure Teredo*

3.3 Dual Stack

Un nœud Dual Stack a le support complet des deux versions du protocole Internet. Ce type de nœuds est souvent connu sous le nom de nœud IPv4/IPv6 c'est-à-dire qu'IPv4 et IPv6 coexistent sur la même machine en tant que deux protocoles natifs standards et ils communiquent directement et séparément avec l'extérieur. Ce mécanisme décrit dans le RFC 2893 est l'un des moyens de transition le plus utilisé actuellement :

- Dans la transmission avec un nœud IPv6, il se comporte comme un nœud IPv6-only ;
- Dans la transmission avec un nœud IPv4, il se comporte comme un nœud IPv4-only.

Un nœud IPv6/IPv4 a au moins une adresse pour chaque version du protocole Internet. Il utilise la configuration statique ou le DHCP pour configurer une adresse IPv4 et l'autoconfiguration pour configurer une adresse IPv6. [7]

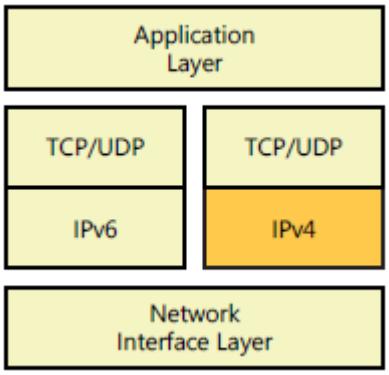

Figure 3.12 : *Architecture Dual stack*

3.4 NAT-PT

NAT-PT (Network Address Translation – Protocol Translation) est un outil de migration conçu pour aider les utilisateurs dans la transition de leurs réseaux IPv4 vers IPv6.

C'est un mécanisme qui permet à un réseau uniquement IPv6 d'interagir avec des équipements IPv4 et vice versa. On utilise un boitier qui peut être un routeur ou un ordinateur avec un logiciel spécifique, celui-ci fait correspondre des adresses IPv6 à des adresses IPv4 lorsqu'un équipement v6 veut communiquer avec un v4. Les équipements raccordés en v4 ou en v6 n'ont besoin d'aucun paramétrage particulier. [7] [11]

Il devient donc possible de rendre accessible via IPv6 toute une infrastructure hébergeant des services bien que les services soient à l'origine en IPv4 et le restent. Les communications depuis l'extérieur vers l'infrastructure peuvent se faire en IPv4 comme auparavant ou en IPv6 via la translation d'adresses.

Les utilisateurs peuvent utiliser un système statique ou des systèmes dynamiques pour l'application du protocole NAT-PT dans leur réseau.

3.4.1 NAT-PT statique

La figure 3.13 illustre l'utilisation classique du NAT-PT statique, dans ce cas de figure on traduit une adresse IPv4 par une adresse ipv6 fixe et vice versa.

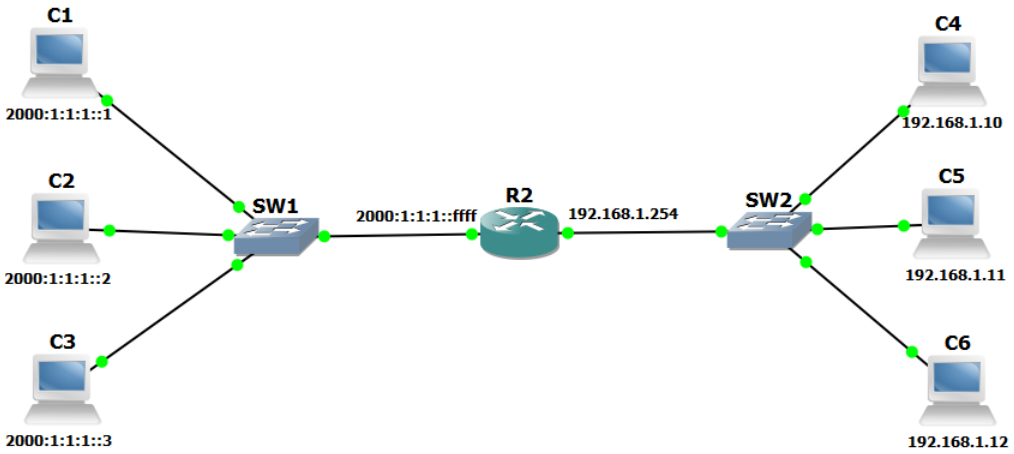

Figure 3.13 : *NAT-PT statique*

	IPv6	Correspond à	IPv4
V6V4	2000:1:1:1::1/64	→	192.168.1.10
	2000:1:1:1::2/64	→	192.168.1.11
	2000:1:1:1::3/64	→	192.168.1.12

Tableau 3.02: *Translation V6V4 statique*

Chaque adresse IPv6 est convertie en adresse IPv4 correspondante, on nomme ce sens translation V6V4.

Dans l'autre cas, chaque adresse IPv4 est convertie en adresse IPv6 correspondante, on nomme ce sens de translation V4V6. Dans ce sens, il est pratique de convertir l'adresse IPv4 en hexadécimal et de l'ajouter après le préfixe IPV6. Le préfixe IPv6 NAT-PT (2000::/96 dans notre cas) est réservé pour les adresses IPv4, il peut faire partie du réseau IPv6 existant, ou utiliser le préfixe d'un réseau différent.

	IPv4	Correspond à	IPv6
V4V6	192.168.1.10	→	2000:C0A8:00A
	192.168.1.11	→	2000:C0A8:00B
	192.168.1.12	→	2000:C0A8:00C

Tableau 3.03: *Translation V4V6 statique*

3.4.2 NAT-PT dynamique

Dans ce cas de figure, des pools d'adresses sont alloués lors de la transition d'un réseau IPv6 à un réseau IPv4 et vice versa.

	IPv6	Correspond à	IPv4
V6V4	2000:1:1:1::/96	→	192.168.1.1
			192.168.1.2
			192.168.1.3

Tableau 3.04: *Translation V6V4 dynamique*

Le nombre d'adresse dans le pool détermine le nombre de connexions simultanées possibles.

	IPv4	Correspond à	IPv6
V4V6	192.168.1.0/24	→	2000::/96

Tableau 3.05: *Translation V4V6 dynamique*

Il est à noter que le protocole NAT-PT a été classé comme déprécié par le RFC 4966, dont voici les raisons : NAT-PT est considéré comme un mécanisme de transition vers IPv6 et en aucun cas il ne doit limiter les possibilités de communications qu'offre IPv6 ; dans le cas du NAT-PT dynamique, on a moins d'adresses IPv4 que d'adresses IPv6 et sans une gestion fine des timeout on pourrait avoir un manque d'adresses IPv4 ; il peut y avoir des pertes d'informations étant donné que des nouveaux champs sont apparus avec IPv6 comme flow label ou encore des extensions que l'on ne peut pas translater comme les entêtes de routage ou de mobilité ; concernant ICMPv6, il y a des nouvelles fonctionnalités, celles-ci ne sont pas rétro compatibles ; au niveau de la sécurité il est possible de surcharger les mémoires qui font la correspondance avec les translations avec des attaques de déni de service ; enfin, des incompatibilités sont présentes avec le mécanisme de DNS. [7] [11]

C'est pour ces raisons que l'IETF a choisi de classer comme déprécié NAT-PT, pour éviter de le déployer à grande échelle en causant beaucoup d'incompatibilité ; seulement il est dit que si NAT-PT est utilisé dans des cas particuliers où les problèmes cités ont été considérés et ne sont pas présents, alors NAT-PT est une solution envisageable.

3.5 NAT64 / DNS64

Plus récemment, en Avril 2011, a été publiée la norme décrivant ledit successeur de NAT-PT : NAT 64, RFC 6146. Pour mémoire, l'IETF conseillait de ne pas utiliser la translation comme mécanisme de transition à IPv6, la solution conseillée était d'activer IPv6 en laissant IPv4 et, petit à petit, de désactiver IPv4 ; seulement le déploiement d'IPv6 a été repoussé et il est maintenant trop tard pour faire une transition « douce », la translation redevient une nécessité et NAT64 apporte son lot de nouveautés pour corriger les défauts de NAT-PT évoqués précédemment avec notamment DNS64 qui apporte des nouvelles fonctionnalités et corrige des défauts concernant le mécanisme DNS, cependant la translation reste toujours contraire à l'un des principes fondamentaux d'IP : la communication directe de pair à pair. [7] [11]

DNS64 permet à un hôte uniquement en IPv6 de communiquer avec un serveur uniquement IPv4 ; le client IPv6 interroge un serveur DNS qui lui renvoie soit :

- Une adresse IPv6 si la requête DNS retourne un enregistrement IPv6.
- Une adresse IPv6 mappant IPv4 si la requête DNS retourne seulement un enregistrement IPv4 (c'est ce qui est illustré par le point « a » dans la figure 3.14).

Dans le cas où l'adresse IPv6 retournée est classique, le mécanisme de translation n'est pas nécessaire et les paquets peuvent être routés classiquement.

Dans le cas où une adresse IPv6 mappant IPv4 est retournée, le routeur effectuant du NAT64 qui a été paramétré en accord avec le serveur DNS64 réalise une translation de l'adresse IPv6 vers l'adresse IPv4 contenue dans l'adresse IPv6 mappant IPv4 ; cette adresse est à un format reconnu avec le préfixe 64:ff9b::/96 définit dans le RFC 6052.

Au retour, l'équipement de translation fait l'opération inverse. Un exemple simple de cette communication entre un client IPv6 et un serveur IPv4 est illustré par la figure 3.14.

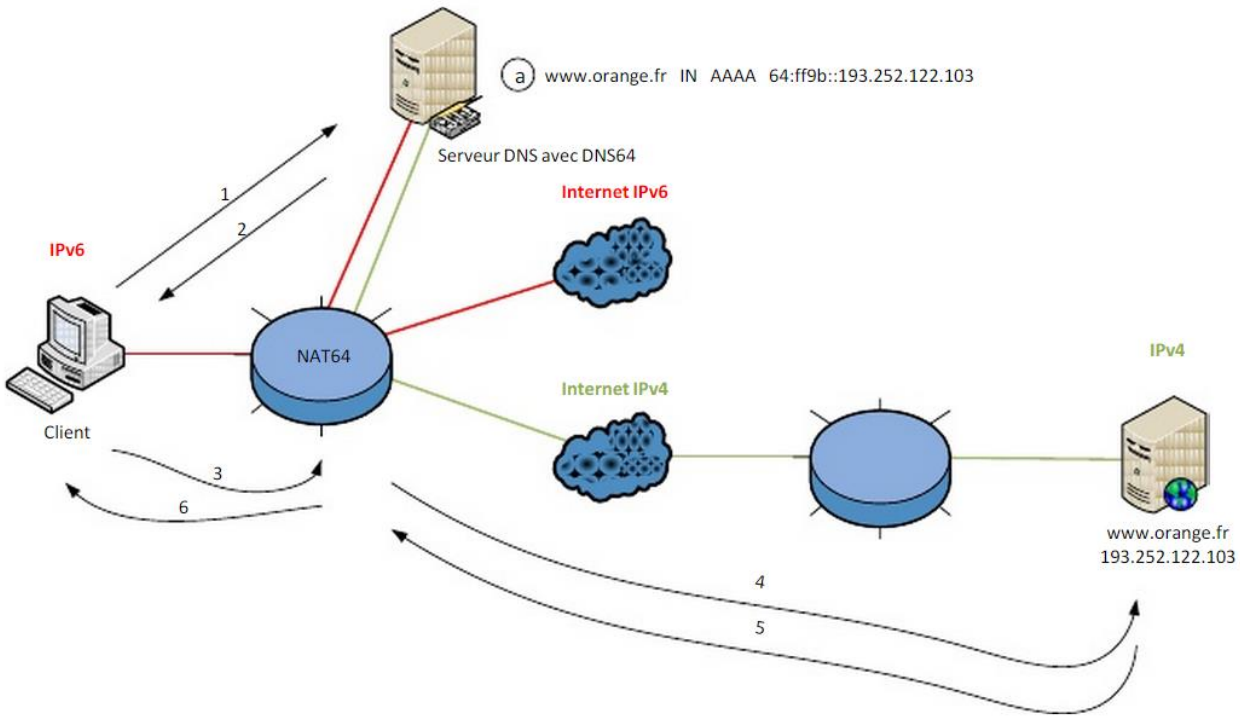

Figure 3.14 : *Illustration du mécanisme DNS64 et NAT-64*

Pour synthétiser nous avons une solution de translation qui permet à des hôtes uniquement IPv6 de joindre des hôtes uniquement IPv4 et vice versa : NAT-PT. Nous avons une solution plus orientée, NAT 64 couplée à DNS 64 qui permet à des hôtes uniquement IPv6 sur un site donné de joindre des hôtes IPv4 à l'extérieur de ce site.

3.6 MPLS comme outil de transition IPv4 vers IPv6

Utiliser un cœur de réseau MPLS pour transporter des flux IPv6 permet d'interconnecter des îlots IPv6 au travers d'un cœur de réseau IPv4 MPLS. Cette solution est intéressante dans le cadre d'un déploiement d'IPv6 car MPLS commute des labels et non pas des en-têtes IP. Elle offre donc l'avantage de ne pas avoir à mettre à jour les routeurs de cœur.

Trois usages qui lient IPv6 à MPLS peuvent être différenciés :

– *Transition IPv4 vers IPv6* : dans ce contexte, MPLS a été identifié comme une technologie permettant le transport de flux IPv6 à moindre coût. En effet, une fois que le paquet IPv6 est encapsulé dans une trame MPLS sur le routeur d'entrée (le PE-routeur en terminologie MPLS), celle-ci est commutée comme toute autre trame sur les routeurs MPLS de cœur (les

P-routeurs). Cette méthode, appelée 6PE (IPv6 Provider Edge) permet de connecter des sites distants IPv6 au travers d'un réseau de cœur MPLS IPv4. [8] [9]

- *Mise en place des tunnels MPLS* : des protocoles spécifiques (LDP : Label Distribution Protocol, TDP : Tag Distribution Protocol) ou adaptés (BGP, RSVP) construisent les chemins MPLS (les LSP : Label Switched Path) sur la base des informations contenues dans les tables de routage interne. [8] [9]
- *Réseaux privés virtuels* : les L3 VPN MPLS représentent le service le plus utilisé de la technologie MPLS. Ils permettent le déploiement de réseaux privés (virtuels car une seule infrastructure physique est utilisée) en assurant une étanchéité entre eux, tout comme si chaque réseau était physiquement différent. Ils se basent sur le RFC 2547 (BGP/MPLS VPN), à laquelle des extensions ont été ajoutées pour le support d'IPv6.

3.6.1 Technique 6PE

La technique 6PE permet de connecter des îlots IPv6 entre eux au travers d'un cœur de réseau IPv4 MPLS [8]. L'architecture utilisée est la suivante :

- Tunnels MPLS dans le cœur ;
- Utilisation de MP-iBGP pour annoncer les préfixes IPv6.

Ce mode, décrit dans la RFC 4798, est nommé ainsi en référence à IPv6 et aux PE des VPN MPLS (RFC 2547), mais contrairement à ces derniers, la technique 6PE ne permet pas de faire des VPN. Ainsi, les préfixes IPv6 annoncés par les routeurs CE sont placés dans la table de routage globale du routeur 6PE. La technique 6PE décrit un mode de transition vers IPv6 pour un réseau IPv4 /MPLS dans lequel :

- Comme dans un mode tunnel, les routeurs de cœur ne sont pas doubles pile. Ils restent en IPv4 sans aucune modification ; [8]
- Les routeurs de périphérie raccordant des clients ou des sous réseaux IPv6 sont double pile. Ils utilisent MP-iBGP pour s'échanger les préfixes de leurs clients avec eux même comme next hop ; [8]
- Les paquets IPv6 des clients sont reçus nativement par les 6PE, encapsulés par le routeur 6PE d'entrée (ingress), décapsulés par le routeur 6PE de sortie (egress) puis envoyés aux clients sur une interface IPv6 native (ou double pile). [8]

3.6.2 Réseaux privés virtuels IPv6 sur MPLS

Cette technique crée des VPN IPv6 en utilisant les LSP MPLS d'un cœur de réseau IPv4 MPLS (le cœur de réseau n'est pas IPv6 MPLS mais bien IPv4 MPLS), offrant ainsi l'avantage d'utiliser le cœur de réseau MPLS déjà existant.

3.7 Conclusion

On a passé en revue dans ce troisième chapitre les principales technologies de transition IPv4 vers IPv6. IPv6 constitue la pierre angulaire de l'évolution nécessaire d'IP vers les nouveaux champs d'application que laissent entrevoir les réseaux sans-fil, la mobilité et l'accessibilité quasi permanente des terminaux. Il n'est, en effet, pas question de mettre à rebout une technologie existante qui fonctionne, mais de permettre le développement de nouvelles idées et applications. La partie simulation se proposera de mettre en œuvre ces technologies de transition sous le logiciel de simulation GNS3.

CHAPITRE 4
SIMULATION SOUS GNS3

4.1 Introduction

Cette dernière partie se propose enfin de mettre en œuvre les technologies de transition IPv4 vers IPv6 présentées dans le chapitre précédent. Il est primordial dans un premier temps de définir les configurations de base en IPv6 c'est-à-dire quelques points techniques sur la configuration et l'autoconfiguration des interfaces puis on passera dans la simulation des technologies de transition. Quatre de ces technologies seront simulées à savoir : le Dual Stack, le Tunneling configuré, le Tunneling automatique ISATAP et enfin le NAT-PT statique.

4.2 Outils de simulation

4.2.1 Présentation de GNS3

GNS3 ou Graphical Network Simulator 3rd version est un émulateur réseau libre avec une interface graphique à l'image de Packet Tracer, Network Visualizer, … Sa différence avec ces derniers c'est sa capacité de reproduction du comportement des IOS et des machines. Il suit le même principe de fonctionnement que les machines virtuelles (VMWare, VirtualBox, …) qui émule un OS (Windows7, linux, …) dans un environnement virtuel.

Ainsi, GNS3 utilise des véritables IOS (Interface Operating System) de Cisco dans un environnement virtuel à travers un ordinateur.

GNS3 est particulièrement intéressant pour :

- L'entrainement, la pédagogie et la familiarisation avec les produits et les technologies de Cisco System,
- Tester les fonctionnalités d'un IOS,
- La vérification rapide de configuration à déployer plus tard dans un environnement de production.

On peut émuler avec GNS3 les gammes de routeurs suivantes à condition que nous disposions de l'image de son IOS :

- Cisco 1700
- Cisco 2600
- Cisco 3600

- Cisco 3700
- Cisco 7200

Ce logiciel fonctionne avec Dynamips et Dynagen.

Les étapes d'installation et de configuration du logiciel sont présentées en annexe.

4.2.2 Présentation de Dynamips et Dynagen

Dynamips est un émulateur de routeurs Cisco capable de faire fonctionner des images IOS de Cisco non modifiées. Ces images d'IOS fonctionnent comme si elles s'exécutaient sur de véritables routeurs. Le rôle de Dynamips n'est pas de remplacer de véritables routeurs mais de permettre la réalisation des architectures avec de vraies versions d'IOS.

Dynagen est un module s'interfaçant avec Dynamips grâce au mode hyperviseur. Dynagen facilite la création et la gestion des architectures grâce à un fichier de configuration simple décrivant la topologie du réseau à simuler.

4.3 Routeurs utilisés

Les gammes de routeur utilisées pour cette simulation sont les gammes 3600 et 3700. Les noms des IOS utilisés sont respectivement « c3640-jk9s-mz.124-16 » et « c3725-adventerprisek9-mz.124-15.T7 ».

Ces deux types de routeur supportent l'Internet protocole version 6 mais avec l'IOS « c3725-adventerprisek9-mz.124-15.T7 », on peut gérer le NAT-PT. L'IOS « c3640-jk9s-mz.124-16 » sera utilisé pour les technologies Dual Stack, Tunneling configuré et ISATAP tandis que dans la partie NAT-PT statique, on utilisera l'IOS « c3725-adventerprisek9-mz.124-15.T7 ».

4.4 Configurations basiques en IPv6

On va voir dans ce qui suit quelques points techniques sur la configuration et l'autoconfiguration des interfaces en IPv6.

4.4.1 Configuration manuelle des interfaces

– Activation d'IPv6 sur une interface spécifique :

```
R1>enable
R1#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#interface fastEthernet 0/0
R1(config-if)#ipv6 enable
R1(config-if)#no shutdown
```

Figure 4.01 : *Activation d'IPv6 sur une interface*

La commande *ipv6 enable* autorise la configuration d'adresses IPv6 sur l'interface concernée, et provoque la génération d'une adresse link-local comme le montre la figure 4.02.

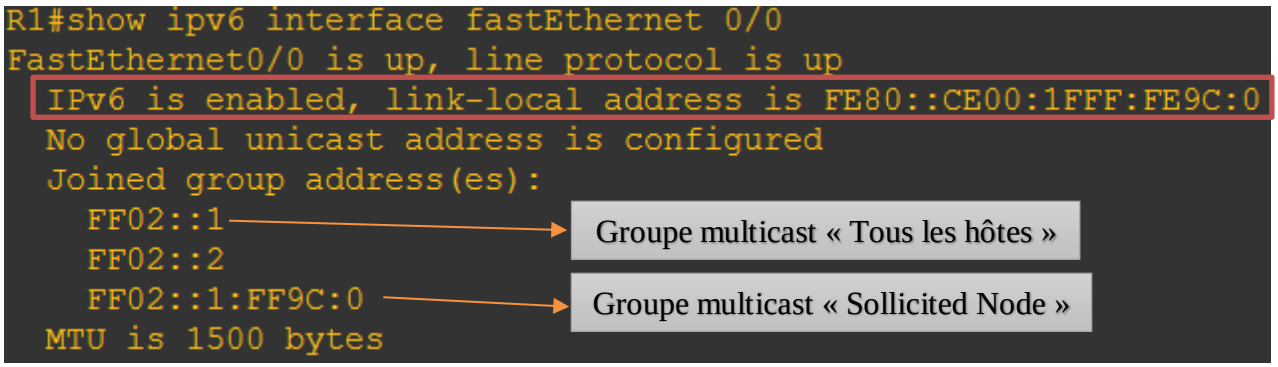

Figure 4.02 : *Affichage sommaire des interfaces IPv6*

– Activation du routage en IPv6 :

```
R1(config)#ipv6 unicast-routing
```

Figure 4.03 : *Activation du routage en IPv6*

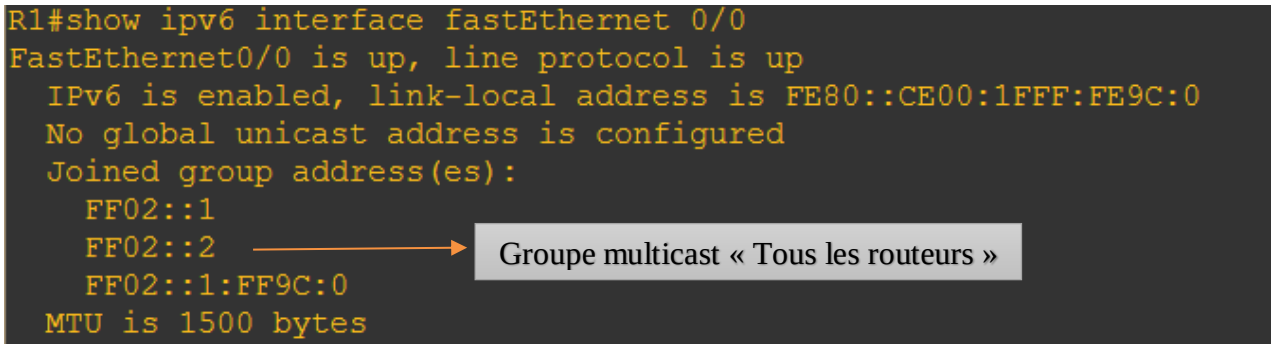

Figure 4.04 : *Affichage sommaire des interfaces IPv6*

La commande *ipv6 unicast-routing* active les fonctionnalités de routage unicast IPv6, les interfaces actives en IPv6 rejoindront également le groupe multicast FF02::2 (tous les routeurs du réseau). Sans cette commande le routeur se comporte comme un simple hôte IPv6.

– Configuration d'une adresse unicast manuelle :

```
R1(config)#interface fastEthernet 0/0
R1(config-if)#ipv6 address 2001:ABCD::1/64
```

Figure 4.05 : *Configuration d'une adresse unicast manuelle*

La commande *ipv6 address* suivie d'une adresse ipv6 et de son préfixe nous permet de configurer statiquement une adresse unicast. Ici les 128 bits de l'adresse sont définis.

```
R1#show ipv6 interface fastEthernet 0/0
*Mar  1 00:16:26.871: %SYS-5-CONFIG_I: Configured from console by console
R1#show ipv6 interface fastEthernet 0/0
FastEthernet0/0 is up, line protocol is up
  IPv6 is enabled, link-local address is FE80::CE00:1FFF:FE9C:0
  Global unicast address(es):
    2001:ABCD::1, subnet is 2001:ABCD::/64
  Joined group address(es):
    FF02::1
    FF02::2
    FF02::1:FF00:1
    FF02::1:FF9C:0
  MTU is 1500 bytes
```

Figure 4.06 : *Adresse unicast globale*

– Configuration d'une adresse unicast IPv6 EUI−64 :

```
R1(config)#interface fastEthernet 0/0
R1(config-if)#ipv6 address 2001:ABCD::/64 eui-64
```

Figure 4.07 : *Configuration d'une adresse unicast IPv6 EUI-64*

La commande *ipv6 address 2001:ABCD::/64 eui-64* configure l'adresse unicast globale selon la méthode EUI-64, 2001:ABCD:: suivi de l'identifiant EUI-64 dérivé de l'adresse MAC de l'interface.

```
R1#show interface fastEthernet 0/0 | include bia
  Hardware is AmdFE, address is cc00.1f9c.0000 (bia cc00.1f9c.0000)
R1#show ipv6 interface fastEthernet 0/0
FastEthernet0/0 is up, line protocol is up
  IPv6 is enabled, link-local address is FE80::CE00:1FFF:FE9C:0
  Global unicast address(es):
    2001:ABCD::1, subnet is 2001:ABCD::/64
    2001:ABCD::CE00:1FFF:FE9C:0, subnet is 2001:ABCD::/64 [EUI]
  Joined group address(es):
    FF02::1
    FF02::2
    FF02::1:FF00:1
    FF02::1:FF9C:0
  MTU is 1500 bytes
```

Figure 4.08 : *Adresse unicast EUI-64*

– Configuration d'une adresse link-local :

```
R1(config)#interface fastEthernet 0/0
R1(config-if)#ipv6 address FE80::2 link-local
```

Figure 4.09 : *Configuration d'une adresse link-local*

Lorsque l'adresse link-local est configurée manuellement, elle modifie la valeur de l'EUI-64 également, modifiant ainsi l'adresse globale de l'interface si celle-ci utilise le format EUI-64 comme le montre la figure 4.10.

```
R1#show ipv6 interface fastEthernet 0/0
FastEthernet0/0 is up, line protocol is up
  IPv6 is enabled, link-local address is FE80::2
  Global unicast address(es):
    2001:ABCD::1, subnet is 2001:ABCD::/64
    2001:ABCD::2, subnet is 2001:ABCD::/64 [EUI]
  Joined group address(es):
    FF02::1
    FF02::2
    FF02::1:FF00:1
    FF02::1:FF00:2
  MTU is 1500 bytes
```

Figure 4.10 : *Adresse link-local*

4.4.2 Autoconfiguration des interfaces

Comme on l'a vu dans le chapitre 2 paragraphe 2.8, en plus du fait qu'on peut configurer manuellement une adresse IPv6 tout comme en IPv4, IPv6 dispose de moyen plus efficace et moins

fastidieux de configurer les interfaces : c'est ce qu'on appelle *autoconfiguration*. On va voir dans ce qui suit un exemple concret d'autoconfiguration en IPv6.

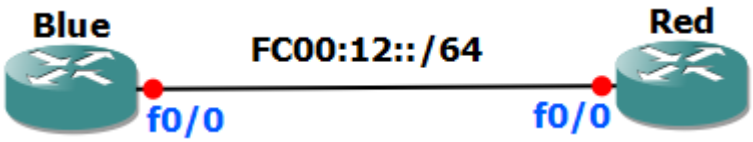

Figure 4.11 : *Autoconfiguration de l'interface f0/0 du routeur Blue*

Etapes de configurations :

- Configuration d'une adresse de site local FC00:12::1/64 sur l'interface f0/0 du routeur Red.
- Configuration du routeur Red pour émettre le préfixe FC00:12::/64 en utilisant le Neighbor Discovery Router Advertisements (ND RA)
- Configuration du préfixe : 2 heures pour les paramètres « valid » et « prefered lifetime »
- Configuration du routeur Red pour qu'il s'annonce en étant le « routeur par défaut » toutes les 20 secondes avec un intervalle lifetime de 50 secondes.
- Configuration du routeur Blue pour qu'il configure automatiquement son adresse IPv6 sur l'interface f0/0.

```
Red(config)#interface fastEthernet 0/0
Red(config-if)#ipv6 address FC00:12::1/64
Red(config-if)#do show ipv6 interface brief
FastEthernet0/0            [administratively down/down]
    FE80::CE20:1CFF:FEDC:0
    FC00:12::1
```

Figure 4.12 : *Configuration d'une adresse sur l'interface fastEthernet0/0*

Une fois que l'adresse de l'interface a été configurée, on définit le préfixe à émettre avec la commande *ipv6 nd prefix*. On ajoute ensuite le « valid lifetime » et le « preferred lifetime ».

```
Red(config-if)#ipv6 nd prefix FC00:12::/64 ?
  <0-4294967295>  Valid Lifetime (secs)
  at              Expire prefix at a specific time/date
  infinite        Infinite Valid Lifetime
  no-advertise    Do not advertise prefix
  <cr>

Red(config-if)#ipv6 nd prefix FC00:12::/64 7200 ?
  <0-4294967295>  Preferred Lifetime (secs)
  infinite        Infinite Preferred Lifetime

Red(config-if)#ipv6 nd prefix FC00:12::/64 7200 7200
```

Figure 4.13 : *Configuration du préfixe neighbor discovery*

```
Red(config-if)#ipv6 nd ra-interval 20
Red(config-if)#ipv6 nd ra-lifetime 50
Red(config-if)#ipv6 unicast-routing
Red(config)#^Z
Red#i
*Mar  1 00:04:14.527: %SYS-5-CONFIG_I: Configured from console by console
Red#debug ipv6 nd
ICMP Neighbor Discovery events debugging is on
```

Figure 4.14 : *Configuration du ra-interval et ra-lifetime du routeur Red*

Une fois les configurations des paramètres du préfixe terminées, il ne reste plus qu'à configurer le routeur Blue de façon à ce qu'il s'autoconfigure son interface fastEthernet0/0 par la commande *ipv6 address autoconfig.*

```
Blue(config-if)#ipv6 address autoconfig
Blue(config-if)#
*Mar  1 00:05:58.771: ICMPv6-ND: Sending NS for FE80::CE00:10FF:FEF4:0 on FastEthernet0/0
Blue(config-if)#
*Mar  1 00:05:59.775: ICMPv6-ND: DAD: FE80::CE00:10FF:FEF4:0 is unique.
*Mar  1 00:05:59.779: ICMPv6-ND: Sending NA for FE80::CE00:10FF:FEF4:0 on FastEthernet0/0
*Mar  1 00:05:59.779: ICMPv6-ND: Address FE80::CE00:10FF:FEF4:0/10 is up on FastEthernet0/0
Blue(config-if)#
*Mar  1 00:06:01.775: ICMPv6-ND: Sending RS on FastEthernet0/0
*Mar  1 00:06:01.795: ICMPv6-ND: Received RA from FE80::CE01:10FF:FEF4:0 on FastEthernet0/0
*Mar  1 00:06:01.795: ICMPv6-ND: DELETE -> INCMP: FE80::CE01:10FF:FEF4:0
*Mar  1 00:06:01.795: ICMPv6-ND: INCMP -> STALE: FE80::CE01:10FF:FEF4:0
*Mar  1 00:06:01.795: ICMPv6-ND: Sending NS for FC00:12::CE00:10FF:FEF4:0 on FastEthernet0/0
*Mar  1 00:06:01.795: ICMPv6-ND: Autoconfiguring FC00:12::CE00:10FF:FEF4:0 on FastEthernet0/0
Blue(config-if)#
*Mar  1 00:06:02.795: ICMPv6-ND: DAD: FC00:12::CE00:10FF:FEF4:0 is unique.
*Mar  1 00:06:02.795: ICMPv6-ND: Sending NA for FC00:12::CE00:10FF:FEF4:0 on FastEthernet0/0
*Mar  1 00:06:02.795: ICMPv6-ND: Address FC00:12::CE00:10FF:FEF4:0/64 is up on FastEthernet0/0
Blue(config-if)#
*Mar  1 00:06:17.731: ICMPv6-ND: Received RA from FE80::CE01:10FF:FEF4:0 on FastEthernet0/0
```

Figure 4.15 : *Autoconfiguration de l'interface fastEthernet0/0 du routeur Blue*

4.5 Mise en œuvre du Tunneling configuré

4.5.1 Architecture

Pour comprendre l'architecture à simuler, il faut se référer au paragraphe 3.2 du chapitre 3.
Le réseau à simuler est composé de 5 routeurs de la gamme 3600. Notre objectif ici c'est de mettre en œuvre un « IPv6-over-IPv4 tunneling configuré » puis de vérifier le bon fonctionnement du réseau en effectuant quelques tests de connectivité.

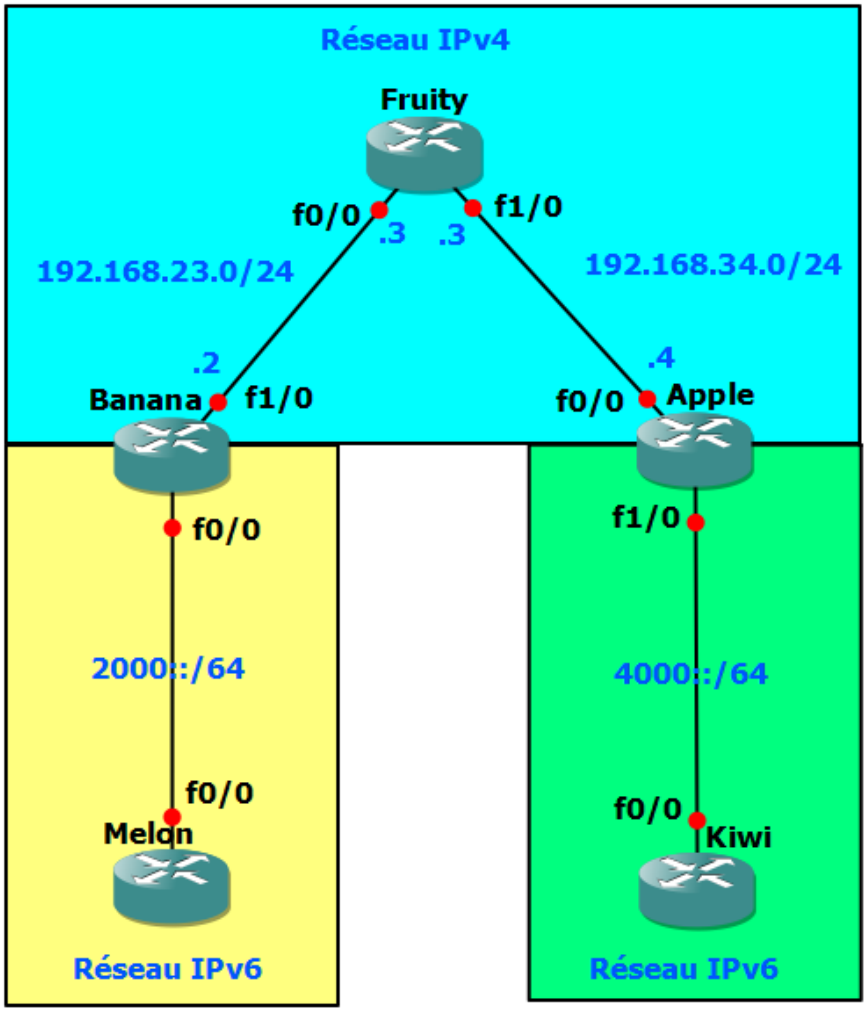

Figure 4.16 : *Tunneling*

4.5.2 Etapes de configurations

Après configuration des interfaces, on procède aux étapes de configurations suivantes :

- OSPFv2 a été configuré dans le réseau IPv4 pour connecter les routeurs entre eux.

- Configuration d'un tunnel IPv6-over-IPv4 entre les routeurs Banana et Apple. Utilisation du préfixe 3000::/64 pour l'interface du tunnel.
- Activation du protocole de routage RIPng sur les routeurs Melon, Banana, Apple et Kiwi.
- Enfin, vérification de la connectivité entre les réseaux 2000::/64 et 4000::/64.

Remarquons que les configurations des routeurs sont présentées dans l'Annexe 1.

4.5.3 Tests et résultats

Pour faire un test de connectivité dans un réseau, nous disposons de l'outil *ping*. Et si nous voulons capturer l'échange de paquet entre les différents routeurs, nous pouvons utiliser le logiciel Wireshark.

Comme on le voit sur la figure 4.17, les routeurs Melon et Kiwi (adresse IPv6 : 4000::CE09:17FF:FEF0:0) sont bien connectés entre eux.

```
Melon>en
Melon#ping 4000::CE09:17FF:FEF0:0

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 4000::CE09:17FF:FEF0:0, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 72/93/152 ms
```

Figure 4.17 : *Connectivité entre Melon et Kiwi*

La figure 4.18 nous montre une capture de paquet effectuée sur l'interface fastEthernet1/0 du routeur Apple, on voit que pour le cas d'IPv6, c'est bel et bien le protocole ICMPv6 qui est sollicité.

No.	Time	Source	Destination	Protocol	Length	Info
7	18.085404000	4000::ce09:17ff:fef0	2000::ce07:18ff:fe94:0	ICMPv6	114	Echo (ping) request id=0x0703,
8	18.180387000	2000::ce07:18ff:fe94	4000::ce09:17ff:fef0:0	ICMPv6	114	Echo (ping) reply id=0x0703, s
9	18.190418000	4000::ce09:17ff:fef0	2000::ce07:18ff:fe94:0	ICMPv6	114	Echo (ping) request id=0x0703,
10	18.275432000	2000::ce07:18ff:fe94	4000::ce09:17ff:fef0:0	ICMPv6	114	Echo (ping) reply id=0x0703, s
11	18.285434000	4000::ce09:17ff:fef0	2000::ce07:18ff:fe94:0	ICMPv6	114	Echo (ping) request id=0x0703,
12	18.365446000	2000::ce07:18ff:fe94	4000::ce09:17ff:fef0:0	ICMPv6	114	Echo (ping) reply id=0x0703, s
13	18.375448000	4000::ce09:17ff:fef0	2000::ce07:18ff:fe94:0	ICMPv6	114	Echo (ping) request id=0x0703,
14	18.455459000	2000::ce07:18ff:fe94	4000::ce09:17ff:fef0:0	ICMPv6	114	Echo (ping) reply id=0x0703, s
15	18.465461000	4000::ce09:17ff:fef0	2000::ce07:18ff:fe94:0	ICMPv6	114	Echo (ping) request id=0x0703,
16	18.550442000	2000::ce07:18ff:fe94	4000::ce09:17ff:fef0:0	ICMPv6	114	Echo (ping) reply id=0x0703, s

Figure 4.18 : *Capture de paquet avec wireshark*

4.6 Mise en œuvre de ISATAP

4.6.1 Architecture

Le réseau à simuler est composé de 3 routeurs toujours de la gamme 3600 : ISATAPRouter, MiddleMan et ISATAPClient. L'objectif ici c'est de mettre en œuvre un tunnel automatique ISATAP ou Intra-Site Automatic Addressing Protocol entre le routeur ISATAPRouter et le routeur ISATAPClient puis de vérifier également le bon fonctionnement du réseau.

Figure 4.19 : *ISATAP*

4.6.2 Etapes de configurations

- OSPFv2 a été configuré dans le réseau pour assurer une connectivité IPv4.
- Configuration d'une interface tunnel0 sur le routeur ISATAPRouter et utilisation du mode ISATAP.
- Configuration de l'adresse IPv6 2001::1/64 sur l'interface tunnel0 et utilisation de la méthode EUI-64 pour les 64 derniers bits.
- Configuration d'une interface tunnel0 sur le routeur ISATAPClient et utilisation du mode ISATAP.
- Configuration automatique de l'adresse IPv6 du routeur ISATAPClient.

4.6.3 Tests et résultats

La commande *show ipv6 interface brief* nous permet vérifier la création du tunnel ISATAP au niveau routeur ISATAPRouter et du routeur ISATAPClient,

```
ISATAPRouter#sh ipv6 int br
FastEthernet0/0            [up/up]
Loopback0                  [up/up]
Tunnel0                    [up/up]
    FE80::5EFE:101:101
    2001::5EFE:101:101
```

```
ISATAPClient#sh ipv6 int br
FastEthernet0/0            [up/up]
Tunnel0                    [up/up]
    FE80::C0A8:1703
    2001::C0A8:1703
```

Figure 4.20 : *Tunnel ISATAP*

Et les routeurs sont bien connectés entre eux.

```
ISATAPClient#ping 2001::5EFE:101:101

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001::5EFE:101:101, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 20/72/128 ms
```

Figure 4.21 : *Connectivité entre ISATAPRouter et ISATAPClient*

4.7 Mise en œuvre du Dual Stack

4.7.1 Architecture

Comme on l'a déjà précisé lors de la présentation des technologies de transition, un nœud Dual Stack a le support complet des deux versions de protocole.

Les protocoles IPv4 et IPv6 coexistent sur la même machine en tant que deux protocoles natifs standards.

Pour le cas de notre simulation, on a utilisé :

- Deux « Switch-routers » SWR1 et SWR2 ;
- Quatre « VPCS » ou Virtual PC Simulator

Le VPCS est un outil qui peut simuler jusqu'à neuf ordinateurs. Il est très utile pour effectuer des "ping", "traceroute" entre les appareils définis dans la topologie de GNS3. Contrairement à de la virtualisation traditionnelle (VirtualBox), il consomme très peu de mémoire.

L'objectif ici c'est de mettre en œuvre la technologie Dual Stack. Ici les deux « switch-routers » SWR1 et SWR2 se conduisent comme des nœuds IPv4/IPv6, on a configuré les protocoles « RIPv2 » et « RIPng » de façon à permettre des communications entre des hôtes IPv4 ou entre des hôtes IPv6 ou entre des hôtes IPv4/IPv6.

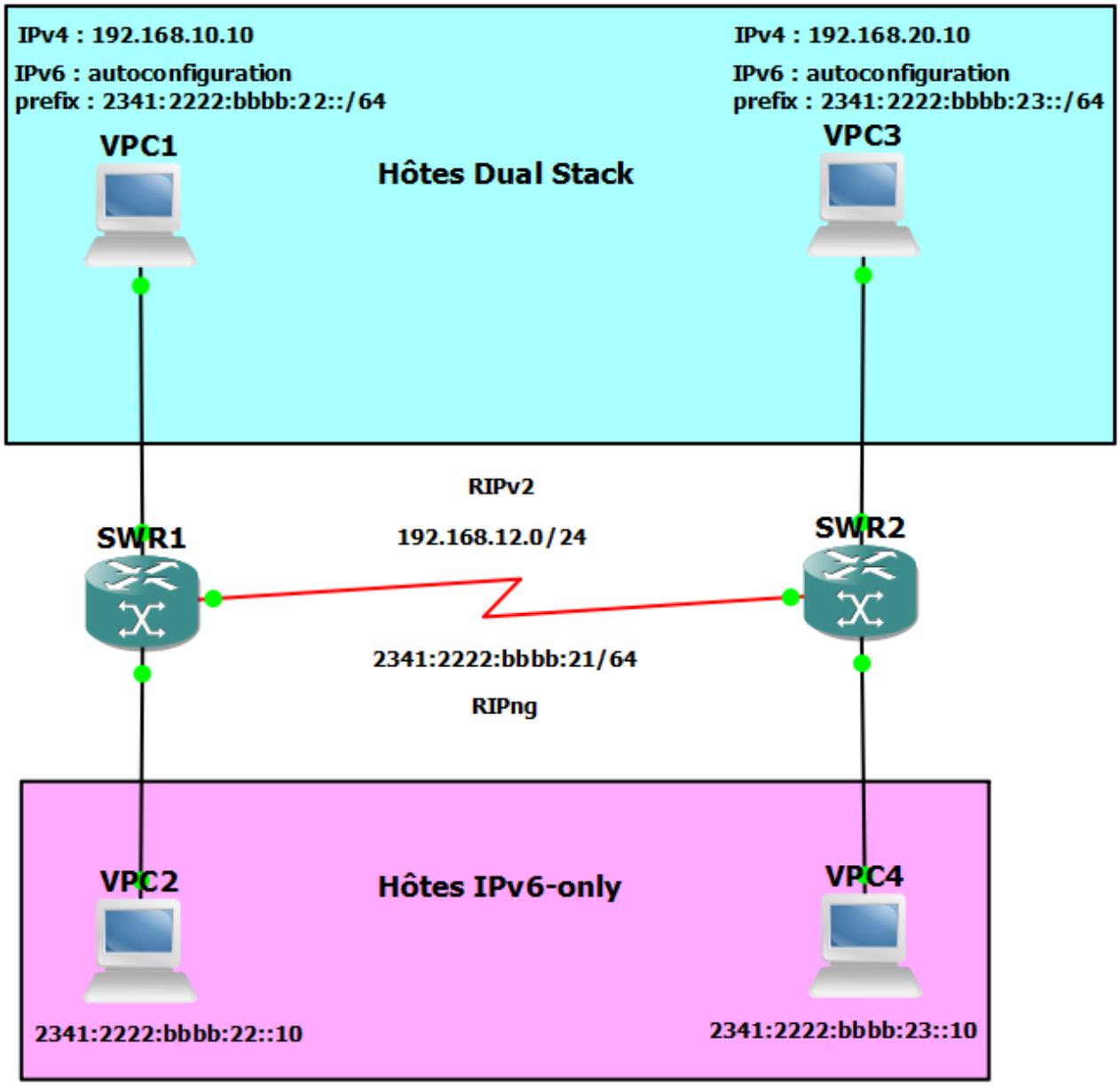

Figure 4.22 : *Dual stack*

4.7.2 Etapes de configurations

- Configuration des adresses IP des « VPCS ».
- Configuration des interfaces séries pour simuler une liaison WAN (Wide Area Network) entre deux réseaux distants.
- Configuration du protocole « RIPv2 » et « RIPng » sur les routeurs.

4.7.3 Tests et résultats

Une fois l'architecture Dual Stack mise en place, nous passons aux tests de connectivité entre les machines VPCS.

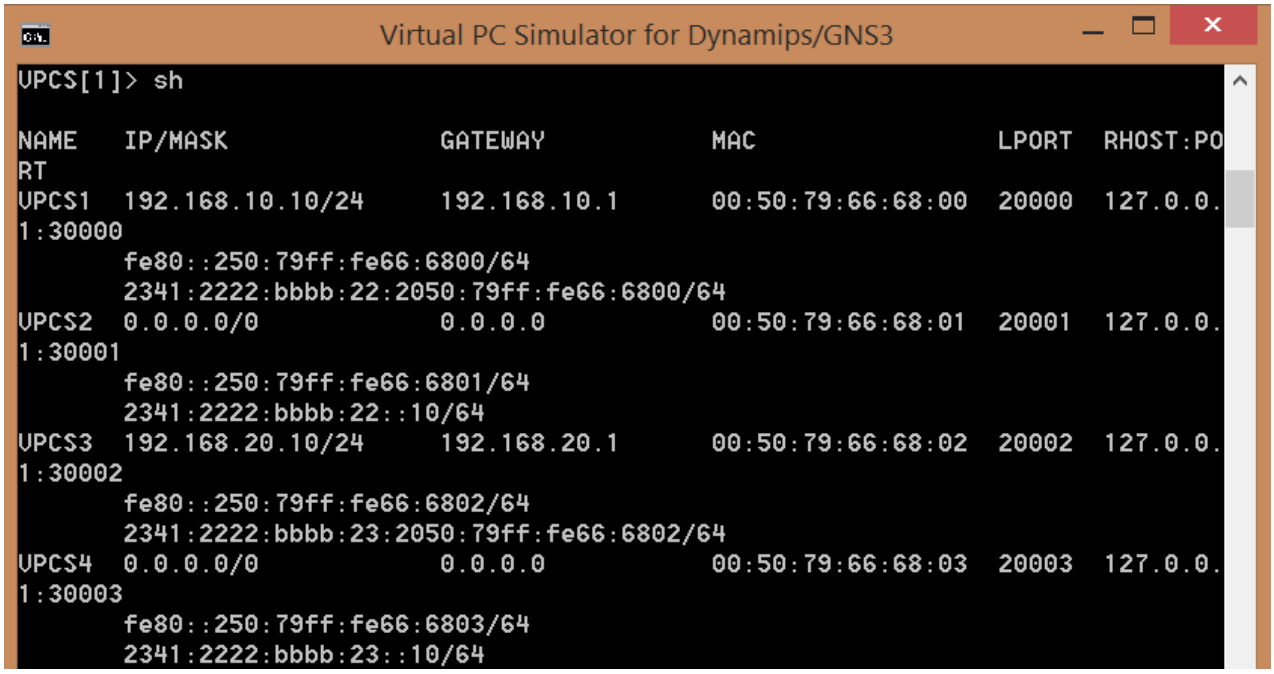

Figure 4.23 : *Les configurations des VPCS*

– Premier test : Communication entre les hôtes IPv6-only

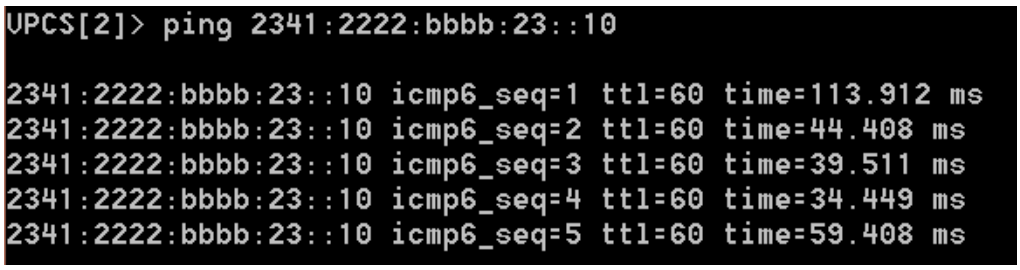

Figure 4.24 : *VPC2 et VPC3 connectées*

– Deuxième test : Communication entre les hôtes Dual Stack

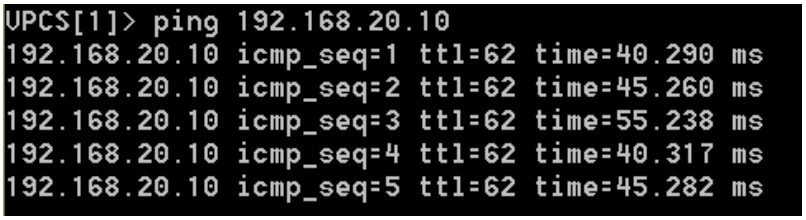

Figure 4.25 : *VPC1 et VPC3 connectées*

– Troisième test : Communication entre hôte Dual Stack et hôte IPv6-only

```
VPCS[1]> ping 2341:2222:bbbb:22::10

2341:2222:bbbb:22::10 icmp6_seq=1 ttl=64 time=1.077 ms
2341:2222:bbbb:22::10 icmp6_seq=2 ttl=64 time=0.974 ms
2341:2222:bbbb:22::10 icmp6_seq=3 ttl=64 time=0.946 ms
2341:2222:bbbb:22::10 icmp6_seq=4 ttl=64 time=1.009 ms
2341:2222:bbbb:22::10 icmp6_seq=5 ttl=64 time=0.912 ms
```

Figure 4.26 : *VPC1 et VPC2 connectées*

4.8 Mise en œuvre du NAT-PT statique

4.8.1 Architecture

Le réseau à simuler est composé de 5 routeurs de la gamme 3700 et de 2 switchs :

- Routeurs : Platinum, Bronze, Gold, Silver
- Switchs : SW1 et SW2

L'objectif est de mettre en œuvre un NAT-PT (Network Address Translation – Protocol Translation) statique.

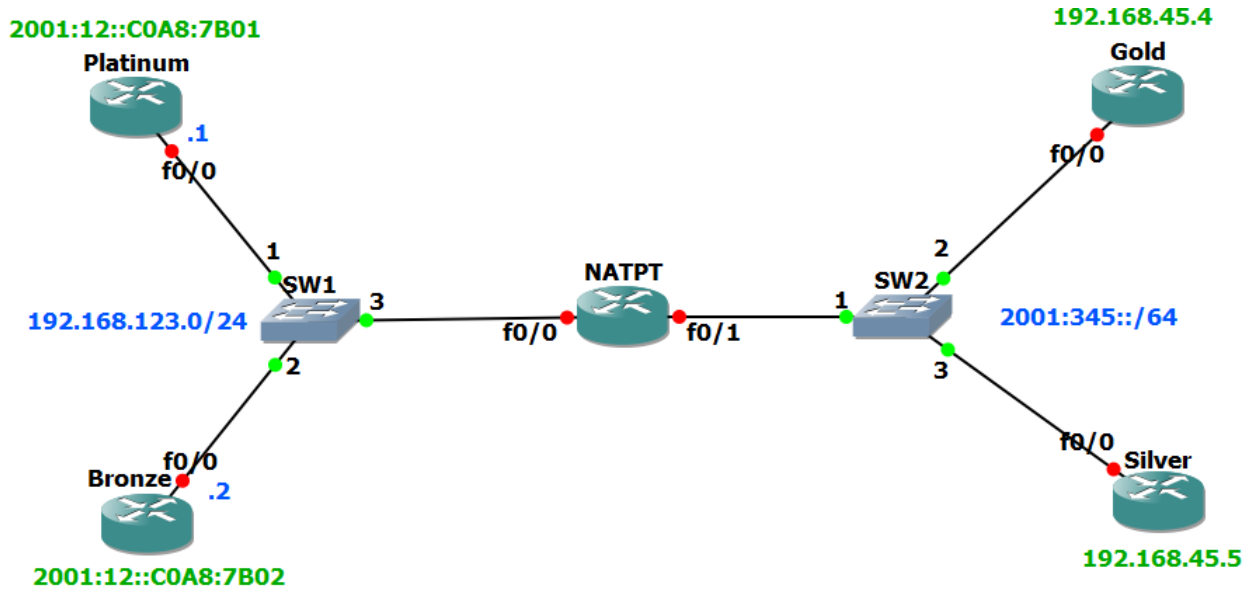

Figure 4.27 : *NAT-PT statique*

4.8.2 Etapes de configurations

- Pas de configurations particulières sur les routeurs raccordés au routeur NAT-PT mis à part les configurations des interfaces de chaque routeur ;
- Configuration du routeur NATPT pour que l'on puisse joindre le routeur Platinum à l'adresse 2001:12::C0A8:7B01 ;

- Configuration du routeur NATPT pour que l'on puisse joindre le routeur Bronze à l'adresse 2001:12::C0A8:7B02 ;
- Configuration du routeur NATPT pour que l'on puisse joindre le routeur Gold à l'adresse 192.168.45.4 ;
- Configuration du routeur NATPT pour que le routeur Platinum soit atteignable à l'adresse 192.168.45.5.

4.8.3 Tests et résultats

Comme il a été expliqué dans le paragraphe concernant le NAT-PT statique, la translation V6V4 consiste à convertir chaque adresse IPv6 en adresse IPv4. Dans l'autre cas, chaque adresse IPv4 est convertie en adresse IPv6, on nomme ce sens de translation V4V6. La figure 4.28 illustre cette translation d'adresse.

```
NATPT#show run | include nat
 ipv6 nat
 ipv6 nat
ipv6 nat v4v6 source 192.168.123.1 2001:12::C0A8:7B01
ipv6 nat v4v6 source 192.168.123.2 2001:12::C0A8:7B02
ipv6 nat v6v4 source 2001:345::4 192.168.45.4
ipv6 nat v6v4 source 2001:345::5 192.168.45.5
ipv6 nat prefix 2001:12::/96
```

Figure 4.28 : *Translation d'adresse v4v6 et v6v4*

Une fois la translation d'adresse terminée, il ne reste plus qu'à vérifier le bon fonctionnement du réseau en effectuant quelques tests de connectivité.

```
Silver#ping 2001:12::C0A8:7B01 repeat 1

Type escape sequence to abort.
Sending 1, 100-byte ICMP Echos to 2001:12::C0A8:7B01, timeout is 2 seconds:
!
Success rate is 100 percent (1/1), round-trip min/avg/max = 72/72/72 ms
```

Figure 4.29 : *Ping de l'adresse IPv6 de Platinum*

```
Gold#ping 2001:12::C0A8:7B02 repeat 1

Type escape sequence to abort.
Sending 1, 100-byte ICMP Echos to 2001:12::C0A8:7B02, timeout is 2 seconds:
!
Success rate is 100 percent (1/1), round-trip min/avg/max = 72/72/72 ms
```

Figure 4.30 : *Ping de l'adresse IPv6 de Bronze*

```
Platinum#ping 192.168.45.5 repeat 1

Type escape sequence to abort.
Sending 1, 100-byte ICMP Echos to 192.168.45.5, timeout is 2 seconds:
!
Success rate is 100 percent (1/1), round-trip min/avg/max = 48/48/48 ms
```

Figure 4.31 : *Ping de l'adresse IPv4 de Silver*

```
Bronze#ping 192.168.45.4 repeat 1

Type escape sequence to abort.
Sending 1, 100-byte ICMP Echos to 192.168.45.4, timeout is 2 seconds:
!
Success rate is 100 percent (1/1), round-trip min/avg/max = 84/84/84 ms
```

Figure 4.32 : *Ping de l'adresse IPv4 de Gold*

Au niveau du routeur NATPT, on peut visualiser les translations d'adresses effectuées par celui-ci en utilisant la commande *debug ipv6 nat* tout en effectuant des tests de connectivité entre les autres routeurs.

```
NATPT#debug ipv6 nat
IPv6 NAT-PT debugging is on
NATPT#
*Mar  1 00:12:39.615: IPv6 NAT: icmp src (192.168.123.2) -> (2001:12::C0A8:7B02), dst (192.168.45.4) -> (2001
345::4)
*Mar  1 00:12:39.671: IPv6 NAT: icmp src (2001:345::4) -> (192.168.45.4), dst (2001:12::C0A8:7B02) -> (192.16
.123.2)
NATPT#
*Mar  1 00:14:04.543: IPv6 NAT: icmp src (2001:345::5) -> (192.168.45.5), dst (2001:12::C0A8:7B01) -> (192.168
.123.1)
*Mar  1 00:14:04.599: IPv6 NAT: icmp src (192.168.123.1) -> (2001:12::C0A8:7B01), dst (192.168.45.5) -> (2001:
345::5)
```

Figure 4.33 : *Mode debugging du routeur NATPT*

4.9 Conclusion

A travers ces simulations, nous nous rendons compte que GNS3 est un outil de réseau puissant et indispensable pour l'évaluation et la simulation des réseaux avant le lancement dans un environnement réel. Ce dernier chapitre nous a permis de mettre en pratique toutes les parties théoriques abordées dans les chapitres précédents que ce soit la configuration et l'autoconfiguration des interfaces, le Dual Stack, le tunneling et la translation d'adresse. Une fois les technologies de transition bien imprégnées et le fonctionnement global d'IPv6 bien compris, la partie mise en œuvre devient aisée.

CONCLUSION GENERALE

Dans le cadre de ce mémoire, nous nous sommes intéressés au protocole IP, plus particulièrement au protocole IPv6 successeur d'IPv4. L'objectif est de se familiariser avec le protocole IPv6 et de mettre en œuvre les mécanismes de transition vers celui-ci dans un environnement de simulation proche de la réalité en utilisant le logiciel GNS3.

Cet ouvrage nous a permis de conclure que le protocole IPv6 apporte bien des évolutions par rapport à son prédécesseur tant au niveau du nombre d'adresse disponible qu'au niveau du format des paquets. En tenant compte de l'évolution de la technologie de l'informatique et de la télécommunication ainsi que la croissance considérable des utilisateurs, la recherche d'une solution pour remédier à la pénurie d'adresse IPv4 et de la coexistence des protocoles IPv4 et IPv6 s'est avérée utile. Une fois largement déployé, IPv6 permettra une évolution du réseau vers de nouveaux concepts (IMS : IP Multimedia Subsystem), ce qui est difficile voire impossible avec IPv4 vu le peu d'espace dont on dispose. En effet, le potentiel d'adressage autorisé par IPv6 intéresse tout particulièrement les opérateurs télécoms mobiles dont les futurs services hauts débits ne se conçoivent qu'à travers une adresse individuelle.

L'étude du protocole IPv6 couvre un large éventail d'applications, on s'est limité dans ce mémoire aux concepts de base mais d'autres domaines comme la mobilité et la sécurité en IPv6 peuvent être étudiées.

Le protocole IPv6 répondra surement aux objectifs édictés : supporter des milliards d'ordinateurs, en se libérant de l'inefficacité de l'espace des adresses IP actuelles ; permettre aux routeurs de router les datagrammes plus rapidement ; faciliter la diffusion multidestinataire (multicast) ; accorder à l'ancien et au nouveau protocole une coexistence pacifique ; fournir une meilleure sécurité (authentification et confidentialité) que l'actuel protocole IP ; donner la possibilité à un ordinateur de se déplacer sans changer son adresse.

Ces dernières années, de nombreuses initiatives ont donc été lancées afin d'accélérer le déploiement de celui-ci sur le réseau. C'est par exemple le cas du « World ipv6 Launch », qui a regroupé les principaux acteurs du web : Microsoft, Facebook, Yahoo, Google, etc. Cependant, la transition vers IPv6 n'est pas l'affaire d'un seul pays, étant donné que l'internet est un système d'interconnexion mondial des réseaux d'équipements utilisant le standard IP, la transition d'un pays isolé ne suffit pas.

ANNEXES

ANNEXE 1 : CONFIGURATIONS DES ROUTEURS

A1.1 Fiche de référence de la configuration des routeurs du tunneling configuré

```
Banana#configure terminal
Banana(config)#interface tunnel 0
Banana(config-if)#tunnel source fastEthernet 1/0
Banana(config-if)#tunnel destination 192.168.34.4
Banana(config-if)#tunnel mode ipv6ip
Banana(config-if)#ipv6 address 3000::1/64
Banana(config-if)#exit
Banana(config)#ipv6 unicast-routing
Banana(config)#ipv6 router rip TAG
Banana(config-rtr)#int f0/0
Banana(config-rtr)#ipv6 rip TAG enable
Banana(config-if)#int tunnel0
Banana(config-if)#ipv6 rip TAG enable

Apple(config)#configure terminal
Apple(config-if)#ipv6 address 3000::2/64
Apple(config-if)#tunnel source f0/0
Apple(config-if)#tunnel mode ipv6ip
Apple(config-if)#tunnel destination 192.168.23.2
Apple(config-if)#exit
Apple(config)#ipv6 unicast-routing
Apple(config)#ipv6 router rip TAG
Apple(config-rtr)#int tunnel0
Apple(config-if)#ipv6 rip TAG enable

Melon#configure terminal
Melon(config)#ipv6 unicast-routing
Melon(config)#ipv6 router rip TAG
Melon(config-rtr)#exit
Melon(config)#int f0/0
Melon(config-if)#ipv6 rip TAG enable

Kiwi#configure terminal
Kiwi(config)#ipv6 unicast-routing
Kiwi(config)#ipv6 router rip TAG
Kiwi(config-rtr)#int f0/0
Kiwi(config-if)#ipv6 rip TAG enable
```

A1.2 Fiche de référence de la configuration des routeurs du tunneling ISATAP

```
ISATAPRouter#configure terminal
ISATAPRouter(config)#ipv6 unicast-routing
ISATAPRouter(config)#int tunnel 0
ISATAPRouter(config-if)#ip address 2001::1/64 eui-64
ISATAPRouter(config-if)#no ipv6 nd suppress-ra
ISATAPRouter(config-if)#tunnel source loopback 0
ISATAPRouter(config-if)#tunnel mode ipv6ip isatap

ISATAPClient#configure terminal
ISATAPClient(config)#int tunnel0
ISATAPClient(config-if)#ipv6 address autoconfig
ISATAPClient(config-if)#ipv6 enable
ISATAPClient(config-if)#tunnel mode ipv6ip
ISATAPClient(config-if)#tunnel source fastEthernet 0/0
ISATAPClient(config-if)#tunnel destination 1.1.1.1
```

A1.3 Fiche de référence de la configuration des routeurs du Dual Stack

```
R1#configure terminal
R1(config)#router rip
R1(config-router)#version 2
R1(config-router)#no auto-summary
R1(config-router)#net 192.168.12.0
R1(config-router)#net 192.168.10.0
R1(config-router)#^Z
R1#conf t
R1(config)#int f0/0
R1(config-if)#ipv6 address 2341:222:bbbb:22::1/64
R1(config-if)#no shutdown
R1(config-if)#exit
R1(config)#int s0/0
R1(config-if)#ipv6 address 2341:2222:bbbb:21::/64 eui-64
R1(config-if)#exit
R1(config)#ipv6 unicast-routing
R1(config)#ipv6 router rip CISCO
R1(config-rtr)#exit
R1(config)#int s1/0
R1(config-if)#ipv6 rip CISCO enable

R2#configure terminal
R2(config)#router rip
R2(config-router)#version 2
```

```
R2(config-router)#no auto-summary
R2(config-router)#net 192.168.12.0
R2(config-router)#net 192.168.20.0
R2(config-router)#^Z
R2#conf t
R2(config)#int f0/0
R2(config-if)#ipv6 address 2341:222:bbbb:23::1/64
R2(config-if)#no shutdown
R2(config-if)#exit
R2(config)#int s0/0
R2(config-if)#ipv6 address 2341:2222:bbbb:21::/64 eui-64
R2(config-if)#exit
R2(config)#ipv6 unicast-routing
R2(config)#ipv6 router rip CISCO
R2(config-rtr)#exit
R2(config)#int s1/0
R2(config-if)#ipv6 rip CISCO enable
```

A1.3 Fiche de référence de la configuration du routeur NAT-PT

```
NATPT#configure terminal
NATPT(config)#ipv6 unicast routing
NATPT(config)#ipv6 nat prefix 2001:12::/96
NATPT(config)#ipv6 nat v4v6 source 192.168.123.1 2001:12::C0A8:7B01
NATPT(config)#ipv6 nat v4v6 source 192.168.123.2 2001:12::C0A8:7B02
NATPT(config)#ipv6 nat v6v4 source 2001:345::4 192.168.45.4
NATPT(config)#ipv6 nat v6v4 source 2001:345::5 192.168.45.5
NATPT(config)#int f0/0
NATPT(config-if)#ipv6 nat
NATPT(config-if)#int f0/1
NATPT(config)#ipv6 nat
```

ANNEXES 2 : ORGANISMES EN CHARGE DE LA GESTION DES RESSOURCES D'ADRESSAGE D'IP

Comme toute ressource limitée, l'espace d'adressage est soumis à un ensemble de règles pour l'attribution et la gestion des adresses. Même si avec IPv6, les problèmes de pénurie d'adresses sont levés, il demeure qu'une mauvaise gestion de l'allocation d'adresses peut rapidement augmenter la taille des tables de routage des routeurs et par conséquent sérieusement entraver la croissance d'Internet. Ainsi existe-t-il des organismes pour la gestion de ces ressources d'adressage.

A2.1 IANA

L'IANA (Internet Assigned Numbers Authority) est une organisation dont le rôle est la gestion de l'espace d'adressage IP d'Internet, et des autres ressources partagées de numérotation requises soit par les protocoles de communication sur Internet, soit pour la connexion de réseaux à Internet.

A2.2 ICANN

L'ICANN (Internet Corporation for Assigned Names and Numbers) est chargé de la gestion de l'espace d'adressage IP au niveau mondial. L'ICANN définit les procédures d'attribution et de résolution de conflits dans l'attribution des adresses, mais délègue la gestion pure de ces ressources à des instances régionales et locales, dans chaque pays, appelées " Internet Registries ".

A2.3 RIR

Le RIR alloue les adresses IP au LIR. Il y a 5 " Regional Internet Registries " (RIR) : l'APNIC pour la région Asie-Pacifique, l'ARIN pour l'Amérique et le RIPE pour l'Europe l'AfriNIC pour l'Afrique ainsi que le LACNIC pour l'Amérique Latine.

A2.4 LIR

Les adresses IP sont finalement allouées à l'utilisateur final qui en fait la demande par un Local Internet Registry (LIR). Un LIR est généralement un fournisseur d'accès à Internet ou une grande organisation comme les entreprises multinationales. Il est sous l'autorité de l'instance régionale de gestion de l'adressage.

ANNEXES 3 : TABLEAU DE CONVERSION ENTRE NOMBRE BINAIRE, HEXADECIMAL ET DECIMAL

Binaire	Hexadécimal	Décimal
0000	0	0
0001	1	1
0010	2	2
0011	3	3
0100	4	4
0101	5	5
0110	6	6
0111	7	7
1000	8	8
1001	9	9
1010	A	10
1011	B	11
1100	C	12
1101	D	13
1110	E	14
1111	F	15

Tableau A3.01 : *Binaire-Hexadécimal-Décimal*

BIBLIOGRAPHIE

[1] Guy Pujolle, « *Réseaux et télécoms* », 3ème Edition, 2012.

[2] Cisco Networking Academy, « *CCNA, Module {1,2,3,4}, version 4.0* », Cisco Systems, 2013.

[3] J.M Edouard, *« Introduction à Internet et aux réseaux »*, mars 2008.

[4] A. Dulaunoy, *« Introduction à TCP/IP et aux routeurs de type IOS (Cisco) »*, Version 0.1b, 2012.

[5] E. Randriarijaona, *« Réseaux TCP/IP »*, cours L3-TCO, Dép. Tél. ESPA, A.U : 2012-2013.

[6] S. Znaty, J.L. Dauphin, *« IP Multimedia Subsystem : Principes et Architecture »*, EFORT, 2005.

[7] J. Davies, *« Understanding IPv6 »*, Microsoft 3rd Edition, 2012.

[8] J. Davies, *« IPv6 Transition Mechanisms »*, Microsoft 2nd Edition , 2012.

[9] S. Mohsen, *« Migration vers IPv6 : Scénarios et mécanismes »*, AFNIC, Juin 2008.

[10] A. Patrick, *« Co-existence et transition IPv4-IPv6 »*, Université Pierre & Marie Curie, 2012.

[11] T. Kouladérou, *« Stratégie nationale de migration de l'IPv4 vers l'IPv6 »*, Institut National Polytechnique Félix Houphouët Boigny, 2010

[12] *« IPv6, théorie et pratique », http://livre.point6.net/*, Janvier 2014

[13] Network Working Group, *« RFC 791 : Internet Protocol Specification », http://www.ietf.org/rfc/rfc0791.txt*, 2000

[14] Network Working Group, *« RFC 2460 : Internet Protocol, Version 6 (IPv6) Specification », http://www.ietf.org/rfc/rfc2460.txt*, 2003

[15] Network Working Group, *« RFC 2893 : Transition Mechanisms for IPv6 Hosts and Routers », http://www.ietf.org/rfc/rfc2893.txt*, 2005

[16] Network Working Group, *«RFC 3315 : Dynamic Host Configuration Protocol for IPv6 (DHCPv6) », http://www.ietf.org/rfc/rfc3315.txt*, 2005

FICHE DE RENSEIGNEMENTS

Nom : RABEMIAFARA MAMIHARIMALALA

Prénom : Nantenaina

Adresse : VQ1D Ankadivory Mandroseza Centre, Antananarivo 101

Téléphone : +261 32 67 552 92

Email : ***rabemiafara_nantenaina@yahoo.fr***

Titre du mémoire :

" MISE EN ŒUVRE DES TECHNOLOGIES DE TRANSITION IPv4 VERS IPv6 "

Nombre de pages : 102

Nombre de tableaux : 18

Nombre de figures : 79

Directeur de mémoire :

Nom : RANDRIAMANAMPY

Prénoms : Samuel

Téléphone : +261 33 14 410 08

Email : ***aviamotor_83@yahoo.fr***

RESUME

Avec l'épuisement rapide et proche de l'espace d'adressage IPv4, il est devenu une priorité pour les fournisseurs de services, des entreprises, fabricants d'appareils IP, développeurs d'applications, et les gouvernements de commencer leurs propres déploiements d'IPv6. Une migration transparente d'IPv4 à IPv6 est difficile à réaliser. Par conséquent, plusieurs mécanismes sont nécessaires qui assurent un changement pas à pas et indépendant en IPv6. Non seulement la transition, mais également l'intégration de l'IPv6 est nécessaire dans les réseaux existants. Les solutions (ou mécanismes) peuvent être divisées en trois catégories : le Dual Stack, le Tunneling et la translation d'adresse. Dual Stack est le moyen le plus souple de déployer IPv6 dans les environnements IPv4 existants : IPv6 peut être activé partout où IPv4 est activé. Dans ce projet, on a mis en œuvre dans GNS3 les mécanismes de transitions en IPv6 en utilisant des routeurs CISCO. Le fonctionnement du réseau peut être observé en capturant les paquets dans les interfaces des routeurs en utilisant Wireshark (Analyseur de paquet).

Mots clés : Multicast, transition, IP, tunneling, autoconfiguration, paquet, encapsulation

ABSTRACT

With the exhaustion of the IPv4 addressing space quickly approaching, it has become a high priority for service providers, enterprises, IP appliances manufacturers, application developers, and governments to begin their own deployments of IPv6. A seamless migration from IPv4 to IPv6 is hard to achieve. Therefore, several mechanisms are required which ensures stepwise and independent change to IPV6. Not only the transition, integration of IPv6 also required into the existing networks. The solutions (or mechanisms) can be divided into three categories: Dual Stack, Tunneling and address translation. Dual-stack is the most versatile way to deploy IPv6 in existing IPv4 environments : IPv6 can be enabled wherever IPv4 is enabled. In this project the transitions mechanisms are implemented in GNS3 using CISCO routers. The operation of this network can be observed by capturing the packets in the router interfaces using Wireshark (Packet analyser).

Keywords : Multicast, transition, IP, tunneling, autoconfiguration, packet, encapsulation