Université Abou Bekr Belkaid جامعة أبي بكر بلقايد
تلمسان الجزائر

République Algérienne Démocratique et Populaire
Université Abou Bakr Belkaid– Tlemcen
Faculté des Sciences
Département d'Informatique

Mémoire de fin d'études

Pour l'obtention du diplôme de Master en Informatique

Option: *Réseaux et systèmes distribués* (R.S.D)

Inscrit Sous le n°:
Date le: 09 JUIL. 2012
Code: 1586

Thème

La Mobilité sur IPv6 : Configuration et Tests

جامعة أبي بكر بلقايد - تلمسان
المكتبة
كلية العلوم

Réalisé par :

- **BENZIANE Yassine**
- **BOUKLIKHA Amine**

Présenté le 04 Juillet 2012 devant le jury composé de MM.

- *Mme DIDI Fadoua* *(Présidente)*
- *Mme BELHABI Amel* *(Encadreur)*
- *Mr BENMAMMAR Badr* *(Examinateur)*
- *Mme LABRAOUI Nabila* *(Examinateur)*

Année universitaire: 2011-2012

Date le: 16 DEC. 2014
Code: 122

Remerciement

Nous tenons à remercier tout particulièrement Mme BELHABI Amel enseignant à l'université Abou Bekr Belkaid notre encadreur de mémoire pour son aide, son soutien, ses conseils, sa patience et sa générosité. Son ouverture d'esprit, sa disponibilité et ses analyses pertinentes ont contribué à rendre cette étude agréable et enrichissante.

Nous souhaitons remercier nos examinateurs d'avoir accepté de participer au jury de ce mémoire : Madame F.DIDI (présidente), Monsieur B.BENMAMARE, Madame N.LABRAOUI, Un grand merci à nôtres chef de département Monsieur A.BENAMMAR.

Nous nous n'oublierons pas de remercier tout le corps enseignement de notre université (faculté des sciences de l'ingénieur de Tlemcen) pour les conseils avises et les suggestions qui nous ont été proposées.

Enfin, nous tenons à remercier Mr BENDIMERAD Directeur du laboratoire LTT à notre université ainsi que son ingénieur de labo Mlle BENOUTMAN qui a mis à notre disposition tout le matériel nécessaire à notre PFE.

BIBLIOTHEQUE SCIENCES
BFST120

Dédicaces

Yassine

Je dédie ce travail :

A mes très chers parents qu'ils ont su m'apporter tout leur soutient, leurs pensées, leurs prières et leur amour.

A mes frères et sœurs, chacun est différent mais chacun est Merveilleux.

A mes oncle Sid Ahmed et Nourdinne, et ma meilleur tante dans le monde Karima.

A tout ma famille, et mes proches,

A tous mes amis, très particulièrement Mohamed, Omar, Chakib, Amine et Zineb et tous qui me Sont cher.

Amine :

Je dédie ce modeste travail :

A mes parents,

A toute la famille, frères et sœurs, pour leur soutien moral.

A tous mes amis Yassine, Souhila, Warda, Zineb, Zakaria, Omar, et à tous ceux qu'on aime et à toutes les personnes qui nous ont Prodigué des encouragements et se sont données la peine de nous soutenir durant cette période.

A mes chers enseignants sans aucune exception.

A Mlle BENOUTMEN.

Et à vous chers lecteurs.

Table de matières

Liste de figures

Liste des tableaux

Liste des abréviations

Mobile IPv6: Mobile Internet Protocol version 6.

IPv6: Internet Protocol version 6.

IPv4 : Internet Protocol version 4.

MIP6: Mobile Internet Protocol 6.

FreeBSD: Free Berkeley Software Distribution.

MN: Mobile Node.

HA: Home Agent.

CN: Correspondent Node.

BU: Binding Update.

BA: Binding Acknowledgment.

IPsec: Internet Protocol Security.

CoA: Care of address.

HoA: Home of address.

QoS: Quality of service.

DNS: Domain Name System.

IETF : L'Internet Engineering Task Force.

InterNIC : The Internet's Network Information Center.

CIDR : Classless Inter-Domain Routing.

NAT :Network address translation.

MTU : le maximum transmission unit.

DNS : Le Domain Name System (ou système de noms de domaine).

ICMP6 : Internet Control Message Protocol Version 6.

Résumé

La nouvelle génération de Protocole Internet, IPng (next generation), ou IPv6 va offrir de nouvelles capacités d'adressage, des options de sécurité, et bien d'autres fonctionnalités comme la mobilité qui est l'un des points majeurs sur lequel IPv6 a été conçu.

Dans ce projet nous montrerons la mobilité et ceci après avoir implémenté une pile protocolaire appelée KAME et configurer sous une plateforme FreeBSD, et tester les communications entre le nœud mobile et son correspondant et ceci même quand le nœud mobile est en déplacement d'un réseau vers un autre.

MOTS-CLÉS :
Mobile IPv6, IPv6, MIP6, KAME, FreeBSD.

Introduction Générale

Le protocole réseau le plus utilisé dans les réseaux actuels est le protocole IP, qui signifie « Internet Protocol ». La fonction principale de ce protocole est le routage. Pour cela il est nécessaire d'attribuer un numéro unique appelé adresse IP à la machine destinataire ainsi qu'à l'émetteur pour pouvoir lui retourner des informations.

La première version d'Internet Protocole c'est la version IPv4. Avec cette version, l'adressage se fait sur 32 bits, ce qui serait suffisant comme espace. Mais, IPv4 est un protocole qui est trop restreint de par son utilisation et qui est donc coûteux en termes d'adresses gaspillées. Bientôt, étant donné l'expansion de l'Internet, il n'existera plus d'adresse disponible sous IPv4, pour résoudre ce problème l'Internet Engineering Task Force (IETF)[1] a débuté des travaux pour mettre au point une nouvelle version du protocole IP, le protocole IPv6. Ce nouveau protocole et plus particulièrement sur sa mise en place dans les différents systèmes d'exploitation présents sur le marché Windows, linux, Mac, BSD …, ainsi que dans les équipements réseaux. Cette version va offrir de nouvelles capacités d'adressage (taille d'adresse 128 bits), des options de sécurité, et bien d'autres fonctionnalités qui vont faciliter les interconnexions globales, parmi ces fonctionnalités en peut trouver la mobilité des équipements qui est l'objectif de notre projet.

La principale problématique de la mobilité est certainement de pouvoir conserver les communications en cours de déplacement. L'intérêt d'IPv6 est de proposer des mécanismes d'auto-configuration mais aussi de permettre la mise en œuvre de la mobilité de façon transparente. Nous nous proposons de tester la mobilité sous une plateforme FreeBSD version 5.4 à l'aide de l'implantation fournie par le projet KAME[2]. Au premier temps, on a téléchargé la version de la pile kAME adapter avec la version FreeBSD 5.4, et configurer et après tester pour montrer la mobilité.

[1] L'Internet Engineering Task Force, abrégée IETF, littéralement traduit de l'anglais en « Détachement d'ingénierie d'Internet » est un groupe informel, international, ouvert à tout individu, qui participe à l'élaboration de standards Internet. L'IETF produit la plupart des nouveaux standards d'Internet..

[2] KAME Project met à disposition une implantation de la mobilité pour les distributions BSD sous IPv6.

Chapitre I : Partie 1 Protocole Internet Version 4

I. Introduction

Le protocole IP est un protocole de transport (niveau 3 du modèle OSI). Il est le protocole de base du réseau internet. Le protocole IP est un protocole dit non fiable. Il n'effectue pas de vérification que les paquets soient bien arrivés au destinataire mais fournit simplement un moyen de transporter le paquet vers son destinataire en utilisant un système d'adresses. L'Internet Protocol version 4 ou **IPv4** est la première version d'IP à avoir été largement déployée, et forme encore la base (en 2010) de l'Internet. Elle est décrite dans la RFC 791 de septembre 1981. IPv4 utilise une adresse IP sur 32 bits, ce qui est un facteur limitant à l'expansion d'Internet puisque seulement 4 294 967 296 (232) adresses sont possibles. Une adresse IPv4 est généralement écrite en notation décimale avec quatre nombres, compris entre 0 et 255, séparés par des points. Par exemple : 212.85.150.134. On parle de notation décimale pointée.[1]

Ce chapitre a pour objectif de donnée une structure de paquet IPv4, les différents classes d'adresses avec le masque de sous réseau.

II. Structure des paquets IPv4

II.1. Format du paquet IPv4

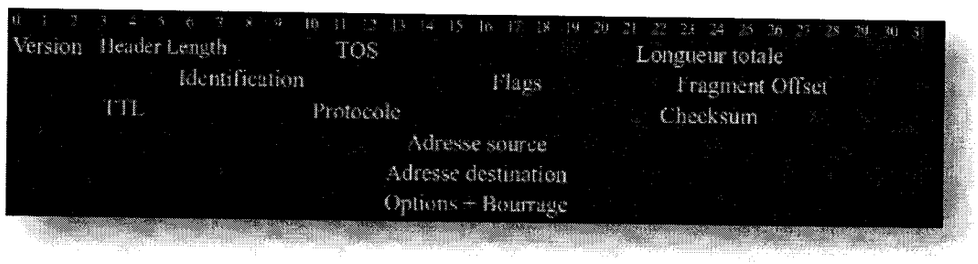


Figure I. 1: Structure de l'entête IPv4

II.2. Les champs de l'entête

Version : Ce champ indique la version du protocole IP utilisée (4 pour IPv4) (4 bits).
Header length : Ce champ indique la longueur de l'entête IP en nombre de groupe de 4 octets. La valeur de ce champ est obligatoirement comprise en 5 et 20 (4 bits).
TOS (Type Of Service) : Ce champ indique la priorité du paquet (1 octet).
Longueur totale : Ce champ indique la longueur totale du paquet en octets (2 octets).

Identification : Ce champ sert à indiquer le numéro de fragment lorsque le paquet est fragmenté (2 octets).

Flags : Ce champ sert à indiquer les flags (3 bits)

Fragment Offset : Ce champ sert à indiquer la position en nombre de groupe de 8 octets du fragment au sein du paquet (13 bits).

TTL : (Time To Live) Ce champ sert à indiquer la durée de vie du paquet (1 octet).

Protocole : Ce champ sert à indiquer le protocole utilisé au-dessus d'IP (1 octet).

Checksum : Ce champ est utilisé pour indiquer la somme de contrôle de l'entête (2 octets).

Adresse source : Ce champ est utilisé pour indiquer l'adresse IP de l'émetteur du paquet (4 octets).

Adresse destination : Ce champ est utilisé pour indiquer l'adresse IP du destinataire du paquet (4 octets).

Options : Ce champ est utilisé pour indiquer les options (0 à 40 octets).

Bourrage : Ce champ est utilisé pour finir de remplir le paquet si la taille n'est pas multiple de 8 bits (0 à 7 bits).

III. L'adressage IP

Le protocole **IP** assure la **livraison de l'information** sous forme de paquets. Cela signifie que toute l'information à envoyer sera fractionnée en petits bouts, appelés **paquets**. Ces paquets seront envoyés de manière indépendante sur le réseau et ça sera au destinataire de remettre les paquets dans le bon ordre et dès les assembler pour récupérer l'information.

Afin de pouvoir communiquer avec le bon destinataire, sans se tromper, il faudra nécessairement pouvoir identifier la machine que l'on recherche, et donc par extension toutes les machines du réseau. Pour identifier chaque machine, on utilise un système d'adressage spécifique : l'adressage IP [3][1].

[3] Une adresse IP (avec IP pour Internet Protocol) est un numéro d'identification qui est attribué à chaque appareil connecté à un réseau informatique utilisant l'Internet Protocol.

III.1 Les classes d'adresses

Une adresse IP identifiant à la fois le réseau et l'hôte, un système de classe d'adresse permet de déterminer le nombre de bits définissant le réseau et l'hôte en spécifiant implicitement un masque réseau.

Certaines classes disposent d'adresses IP publiques et privées. Les adresses publiques sont utilisées pour bâtir Internet, les privées pour constituer des réseaux privés. Il existe 5 classes d'adresses:

Classe A réseau identifié sur un octet.

Classe B réseau identifié sur 2 octets.

Classe C réseau identifié sur 3 octets.

Classe D utilisé pour le multicast.

Classe E réservé.

- **Classe A**

Les adresses de classe A identifient le réseau sur le premier octet. Les hôtes étant identifiés sur 3 octets, il est possible de mettre un grand nombre d'hôtes sur le réseau. Les adresses de classe A sont généralement utilisées pour constituer de très grands réseaux. Le premier octet de l'adresse commence toujours par un bit à 0 définissant ainsi la classe A sur les adresses 0.0.0.0 à 126.255.255.255
La classe dispose d'une adresse réseau privée: 10.0.0.0 - 10.255.255.255 (10.0.0.0/8).

Octet 1	Octet 2	Octet 3	Octet 4
Réseau	Hôte	Hôte	Hôte
255	0	0	0

Table I. 1: classe A

- **Classe B**

Les adresses de classe B identifient le réseau sur les 2 premiers octets. Le premier octet de l'adresse commence toujours par la séquence de bits 10 définissant ainsi le réseau sur 128.0.0.0 à 191.255.255.255. Cette classe dispose de 2 adresses réseaux privées: 172.16.0.0 - 172.31.255.255 (172.16/12) et 169.254.0.0 - 169.254.255.255.
Cette classe héberge aussi les adresses de loopback définissant l'hôte local: 127.0.0.1 à 127.255.255.255

Octet 1	Octet 2	Octet 3	Octet 4
Réseau	Réseau	Hôte	Hôte
255	255	0	0

Table I. 2: classe B

- **Classe C**

Les adresses de classe C identifient le réseau sur les 3 premiers octets. Le premier octet de l'adresse commence toujours par la séquence de bits 110 définissant ainsi le réseau sur 192.0.0.0 à 223.255.255.255

Octet 1	Octet 2	Octet 3	Octet 4
Réseau	Réseau	Réseau	Hôte
255	255	255	0

Table I. 3 : classe C

Cette classe dispose d'une adresse réseau privée: 192.168.0.0 - 192.168.255.255 (192.168.0.0/16).

- **Classe D**

Les adresses de classe D sont utilisées pour le multicast (un seul émetteur, un groupe de récepteurs). Le premier octet de l'adresse commence toujours par la séquence de bits 1110 définissant ainsi le réseau sur 224.0.0.0 à 239.255.255.255

- **Classe E**

Les adresses de classe E sont réservées et ne doivent pas être utilisées. Certaines organisations les utilisent pour effectuer des expérimentations. La classe E s'étend des adresses 240.0.0.0 à 255.255.255.255.

III.2 Attribution des adresses IP

Le but de la division des adresses IP en trois classes A, B et C est de faciliter la recherche d'un ordinateur sur le réseau. En effet avec cette notation il est possible de rechercher dans un premier temps le réseau que l'on désire atteindre puis de chercher un ordinateur sur celui-ci. Ainsi l'attribution des adresses IP se fait selon la taille du réseau [2].

Classe	Nombre de réseaux possibles	Nombre d'ordinateurs maxi sur chacun
A	126	16777214
B	16384	65534
C	2097153	254

Table I. 4: Attribution des adresses IP

Les adresses de classe A sont réservées aux très grands réseaux, tandis que l'on attribuera les adresses de classe C à des petits réseaux d'entreprise par exemple

III.3 Adresses IP réservées

Il arrive fréquemment dans une entreprise qu'un seul ordinateur soit relié à Internet, c'est par son intermédiaire que les autres ordinateurs du réseau accèdent à Internet (on parle généralement de proxy[4]). Dans ce cas, seul l'ordinateur relié à Internet a besoin de réserver une adresse IP auprès de l'INTERNIC[5]. Toutefois, les autres ordinateurs ont tout de même besoin d'une adresse IP pour pouvoir communiquer ensemble de façon interne. Ainsi, l'INTERNIC a réservé une poignée d'adresses dans chaque classe pour permettre d'affecter une adresse IP aux ordinateurs d'un réseau local relié à Internet sans risquer de créer de conflits d'adresses IP sur le réseau. Il s'agit des adresses suivantes:

- **Classe A** : toutes les adresses comprises entre 10.0.0.0 et 10.255.255.255, soit 16 777 216 machines.
- **Classe B** : toutes les adresses comprises entre 172.16.0.0 et 192.168.255.255 soit 1 048 576 machines.
- **Classe C** : toutes les adresses comprises entre 192.168.0.0 et 192.168.255.255 soit 5 536 machines.

[4] un proxy est alors un programme servant d'intermédiaire pour accéder à un autre réseau, généralement internet.
[5] InterNIC (The Internet's Network Information Center) était le service d'information enregistrant l'ensemble des noms de domaines d'Internet.

III.4 Les adresses particulières

Dans l'adressage IP, il existe un certain nombre d'adresses qui ne peuvent pas être appliquées à des machines. Ces adresses sont dites particulières et connues :

- Aucune adresse de machine ne peut commencer par 127. De plus l'adresse **127.0.0.1** est normalement atteignable par toutes les machines, même si la machine n'est pas connectée au réseau car c'est une adresse locale existant pour permettre des tests de connectivités.

- Aucune adresse de machine ne peut se terminer par 0. Les adresses se terminant par 0 sont des adresses dites de réseaux, c'est à dire qu'elles permettent d'identifier un réseau.

- Aucune adresse de machine ne peut se terminer par 255. Les adresses se terminant par 255 sont des adresses de diffusion. Ces adresses permettent de communiquer avec toutes les machines d'un même réseau en même temps. Ces adresses sont appelées adresses de ***broadcast***[6].

- Par extension, l'adresse 255.255.255.255 permet de contacter toutes les machines du domaine de diffusion. Le domaine de diffusion correspond à tous les réseaux connus par la machine qui diffuse.[3]

III.5 Masques de sous-réseau

a. Notion de masque

On fabrique un masque contenant des 1 aux emplacements des bits que l'on désire conserver, et des 0 pour ceux que l'on veut rendre égaux à zéro. Une fois ce masque créé, il suffit de faire un ET entre la valeur que l'on désire masquer et le masque afin de garder intacte la partie que l'on désire et annuler le reste. Ainsi, un masque réseau (en anglais *netmask*) se présente sous la forme de 4 octets séparés par des points (comme une adresse IP), il comprend (dans sa notation binaire) des zéros aux niveaux des bits de l'adresse IP que l'on veut annuler (et des 1 au niveau de ceux que l'on désire conserver) [2].

[6] Adresses de broadcast Elle est utilisée pour diffuser des paquets sur toutes les machines de votre sous-réseau.

Ce masquage divise donc un réseau de classe A (pouvant admettre 16777214 ordinateurs) en 4 sous-réseaux (d'où le nom de *masque de sous-réseau*) pouvant admettre 2^{22}ordinateurs, c'est-à-dire 4194304 ordinateurs. Au passage on remarque que le nombre d'ordinateurs possibles dans les deux cas est au total de 16777214 ordinateurs (4 x 4194304 – 2 = 16777214). Le nombre de sous-réseaux dépend du nombre de bits que l'on attribue en plus au réseau (ici 2). Le nombre de sous-réseaux est donc: [4]

Nombre de bits	**Nombre de sous-réseaux**
1	2
2	4
3	8
4	16
5	32
6	64
7	128
8 (impossible pour une classe C)	256

Table I. 5 : Le nombre des sous Réseau par rapport au nombre de bits

b. Intérêt d'un tel masque

Il y en a en fait plusieurs. Un d'entre eux est de pouvoir connaître le réseau associé à une adresse IP. En effet, comme nous l'avons vu précédemment, le réseau est déterminé par un certain nombre d'octets de l'adresse IP (1 octet pour les adresses de classe A, 2 pour les adresses de classe B, et 3 octets pour la classe C). De plus, nous avons vu que l'on note un réseau en prenant le nombre d'octets qui le caractérise, puis en complétant avec des 0. en généralisant, on obtient les masques suivants pour chaque classe:

- Pour une adresse de **Classe A**, seul le premier octet nous intéresse, on a donc un masque de la forme 11111111.00000000.00000000.00000000, c'est-à-dire en notation décimale: **255.0.0.0**
- Pour une adresse de **Classe B**, les deux premiers octets nous intéresse, on a donc un masque de la forme 11111111.11111111.00000000.00000000, c'est-à-dire en notation décimale: **255.255.0.0**
- Pour une adresse de **Classe C** on s'intéresse aux trois premiers octets, on a donc un masque de la forme 11111111.11111111.11111111.00000000, c'est-à-dire en notation décimale: **255.255.255.0** [4]

c. Création de sous-réseaux

Soit un réseau 34.0.0.0, et supposons que l'on désire que les deux premiers bits du deuxième octet permettent de désigner le réseau. Le masque à appliquer sera alors:11111111.11000000.00000000.00000000 C'est-à-dire 255.192.0.0, Si on applique ce masque, à l'adresse 34.208.123.12 on obtient:34.192.0.0

En réalité il y a 4 cas de figures possibles pour le résultat du masquage d'une adresse IP d'un ordinateur du réseau 34.0.0.0

- Soit les deux premiers bits du deuxième octet sont **00**, auquel cas le résultat du masquage est **255.0.0.0**
- Soit les deux premiers bits du deuxième octet sont **01**, auquel cas le résultat du masquage est **255.64.0.0**
- Soit les deux premiers bits du deuxième octet sont **10**, auquel cas le résultat du masquage est **255.128.0.0**
- Soit les deux premiers bits du deuxième octet sont **11**, auquel cas le résultat du masquage est **255.192.0.0**

Chapitre I : Partie 2 Protocole Internet Version 6

Rappel historique

Le réseau Internet est né au début des années 1970 aux États-Unis et a connu une croissance plutôt douce jusqu'à la fin des années 1980. L'avènement du web au début des années 1990, notamment comme outil de présence commerciale sur Internet, a entraîné un déploiement massif de millions de nouveaux nœuds du réseau et par conséquent un succès fulgurant. La croissance exponentielle de la demande d'adresses IP (numéros uniques assurant l'identification et la localisation des équipements réseau) a fait de l'Internet une victime de son propre succès. Et voilà une première prévision pour la « fin de l'Internet » en 1994 !

Aussitôt, des mesures d'urgence ont été décrétées et appliquées individuellement ou combinées afin d'« arrêter l'hémorragie ». Parmi ces mesures, on peut citer l'allocation exceptionnelle de blocs d'adresses de « classe B »1, la réutilisation des blocs de classes C2, puis l'abolition des classes dans les mécanismes d'allocation et de routage des préfixes IP (CIDR[7], Classless Internet Domain Routing3). S'y sont ajoutés par la suite l'« aménagement » d'un espace d'adressage privé ([RFC 1918[8]]4), l'utilisation de mandataires (proxy) ou de boîtiers de traduction d'adresse (NAT[9]) pour communiquer avec l'extérieur.

Mais parallèlement à l'application de ces mesures d'urgence, l'IETF a lancé dès 1993 les travaux de recherche pour préparer la succession d'IPv4 dont les limites venaient d'être démontrées [5].

I. Introduction

Le protocole IP a été conçu, il y a plus d'une vingtaine d'années, pour connecter des millions d'ordinateurs. Depuis quelques années, IP est victime de son succès et ne permet plus de répondre à la demande de connexion de milliards de machines informatisées dont disposeront les internautes de demain. La nouvelle génération d'IP, IPng (next generation), ou IPv6 va offrir de nouvelles capacités.

D'adressage, des options de sécurité, et bien d'autres fonctionnalités qui vont faciliter les interconnexionsglobales.IPv6 a été recommandé par les responsables de la nouvelle génération du protocole Internet de l'IETF (Internet Engineering Task Force) au cours

[7] Classless Inter-Domain Routing (CIDR) est une méthode d'allocation des adresses IP et le routage des paquets du protocole Internet.
[8] RFC 1918 est l'allocation des adresses pour les Internets privés.
[9] NAT (Network address translation), un mécanisme informatique permettant de faire communiquer un réseau local avec l'Internet.

du meeting de l'IETF de juillet 1994 à Toronto et définitivement spécifié par la recommandation RFC 1752.

Il est important de souligner que les changements des protocoles comme TCP ou IP affectent fatalement les applications existantes. En conséquence, de tels changements doivent être effectués avec précaution et seulement quand ils deviennent nécessaires [6].

II. Principales caractéristiques d'IPv6

- les adresses IPv6 sont codées sur 128 bits (1 milliard de réseaux).
- le principe des numéros de réseaux et des numéros d'hôtes est maintenu.
- IPv6 est conçu pour inter opérer avec les systèmes IPv4 (transition douce prévue sur 20 ans). L'adresse IPv6 peut contenir une adresse IPv4 : on place les 32 bit d'IPv4 dans les bits de poids faibles et on ajoute un préfixe de 96 bits (80 bits à 0 suivis de 16 bits à 0 ou 1)
- IPv6 utilise un adressage hiérarchique (identification des différents réseaux de chaque niveau) ce qui permet un routage plus efficace.
- IPv6 est prévu pour les systèmes mobiles : auto-configuration, notion de voisinage (Neighbors).
- IPv6 permet l'authentification et le chiffrement dans l'en-tête des paquets, ce qui permet de sécuriser les échanges. En effet IP v.6 intègre IPSec (protocole de création de tunnel IP avec chiffrement), qui garantit un contexte sécurisé par défaut.
- IPv6 intègre la qualité de service : introduction de flux étiquetés (avec des priorités).
- IPv6 prend mieux en charge le trafic en temps réel (garantie sur le délai maximal de transmission de datagrammes sur le réseau). [7]

III. Structure des paquets IPv6

III.1. Format du paquet IPv6

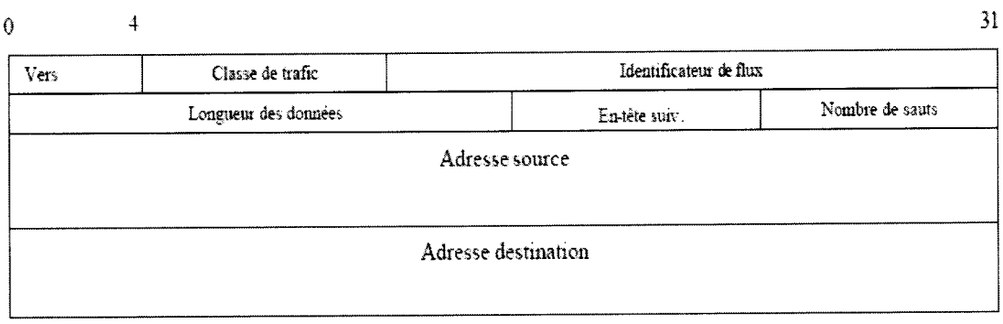


Figure I. 2: Format du paquet IPv6

a) *Version :* Seul champ inchangé par rapport à la version 4, contient la version IP du paquet. Seule modification, sur IPv6, sa valeur est 6.

b) *Classe de trafic :* Nature du trafic. Permet d'offrir un niveau de priorité aux paquets.

c) *Identificateur de flux :* Ce champ permet la mise en œuvre des fonctions de qualités de service. Il permet d'optimiser le routage par un acheminement plus rapide des données. Par ce champ, on peut donner un identifiant à la communication. Selon sa valeur, les routeurs du chemin reconnaissent la connexion et ne dépilent pas les informations, ils les transmettent directement.

d) *Longueur des données :* Longueur des paquets sans l'en-tête (en octets).

e) *En-tête suivant :* Ce champ indique le prochain entête dans le datagramme IPv6, c'est-à-dire l'emplacement des en-têtes optionnels s'ils existent.

f) *Nombre de sauts :* Ce champ remplace le champ « TTL » d'IPv4. Sa valeur, sur 8 bits, est décrémentée à chaque traversée d'un routeur. Si sa valeur atteint la valeur 0, le paquet est détruit et un message d'erreur est émis par ICMPv6.

g) *Adresse de destination :* Peut-être une adresse différente de l'adresse destination finale si l'option "Routing Header" est présente [8].

III.2. Entêtes optionnels

Le paquet IPv6 inclut un champ d'extension pour les fonctionnalités optionnelles (sécurité, source Routing, ...). Les options de IPv6 sont placées dans des en-têtes séparés, intercalés entre l'en-tête IPv6 et l'en-tête de la couche transport.

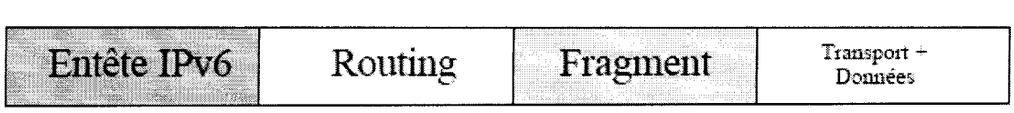


Figure I. 3: l'en-tête IPv6

IV. Adressage

IV.1. Adressage IPv6

L'adressage sous IPv6 se fait désormais sur 128 bits (32 bits sur IPv4).Sous IPv6 on n'a plus de notion de classes, on a seulement un adressage hiérarchique, par préfixe. L'adressage se fait désormais par l'attribution d'une partie d'adresse fixe qui définit le réseau, cette partie est appelée le préfixe. On a, en fait, une conservation de l'adressage CIDR sous IPv4.

Cet adressage hiérarchique consiste en l'attribution d'un préfixe au premier routeur d'un réseau, sur 10 bits par exemple. Le routeur du sous réseaux aura lui une partie fixe qui sera celle de son père plus un complément de préfixe qui lui sera imposé. Et ainsi de suie l'attribution des adresses se fait de manière hiérarchique.

IPv6 reconnaît trois types d'adressage :

- Adresse UNICAST : 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.
- Adresse MULTICAST : ce sont les successeurs des adresses broadcast (envoi à un ensemble de machines qui se doivent d'appartenir à une même classe). Une adresse de

  Type multicast désigne un groupe d'interfaces appartenant, en général, à des équipements 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.
- Adresse ANYCAST : Ce type d'adresse est nouveau en IPv6. 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 adresse de type anycast, il est routé à un seul des éléments du groupe et non pas à tous. Ce sera, par exemple, le plus proche au sens de la métrique des protocoles de routage. Ce type d'adresse est encore en cours d'expérimentation et est réservé pour le moment aux routeurs.

  Les adresses anycast ont deux points communs avec les adresses unicast : elles sont allouées dans le même espace d'adressage et ont les mêmes formats [8].

IV.2. Représentation des adresses

Une adresse IPv4 est un mot de 32 bits tandis qu'une adresse IPv6 est un mot de 128 bits. La taille des adresses a donc été quadruplée, ce qui permet d'obtenir un espace adressable en IPv6 nettement plus large que celui en IPv4. Une adresse sur IPv6 est un ensemble de 8 mots de 2 octets, qui sont en fait, 8 groupes de 4 lettres hexadécimales séparés par « : ».

Exemple : FEDC:BA98:7654:3210: FEDC:BA98:7654:3210.

Dans un champ, il n'est pas nécessaire d'écrire les zéros placés en tête. En outre plusieurs champs nuls consécutifs peuvent être abrégés par «::». Ainsi les deux notations suivantes sont équivalentes :

FEDC:0000:0000:0000:0400:A987:0043:210F

FEDC:400:A987:43:210F

Plus particulièrement, l'adresse formée uniquement par des zéros est représentée comme suit « :: »

Naturellement, pour éviter toute ambiguïté, l'abréviation «::» ne peut apparaître qu'une fois au plus dans une adresse.

La représentation des préfixes IPv6 est similaire à la notation CIDR utilisée pour les préfixes IPv4. Un préfixe IPv6 est donc représenté par la notation :

adresse-ipv6 / longueur-du-préfixe -en-bits

Les formes abrégées avec « ::» sont autorisées :

3EDC:BA98:7654:3210:0000:0000:0000:0000/64

3EDC:BA98:7654:3210:0:0:0:0/64

3EDC:BA98:7654:3210::/64

::/0 (Défaut)

Enfin on peut combiner l'adresse d'une interface et la longueur du préfix réseau associé en une seule notation : 3EDC:BA98:7654:3210:945:1321:ABA8:F4E2/64. [8]

- **D'autres types d'adresses**

✓ Adresse indéterminée : l'adresse indéterminée est utilisée comme adresse source par un équipement du réseau pendant son initialisation, avant d'acquérir une adresse. Sa valeur est 0:0:0:0:0:0:0:0 (ou en abrégé ::).

✓ Adresse de bouclage (loopback address) : L'adresse de bouclage vaut 0:0:0:0:0:0:0:1, soit en abrégé ::1. Il s'agit de l'équivalent de l'adresse 127.0.0.1

d'IPv4. Elle est utilisée par un équipement du réseau pour envoyer un paquet IPv6 à lui-même.[8]

- **Adresse compatible IPv4**

Si l'adresse sous Ipv4 est 134.157.4.16, elle devient alors : 0:0:0:0:0:0:0:134.157.4.16
Soit après compression :::134.157.4.16 [8].

IV.3. Durée de vie des adresses

IPv6 généralisant le plan d'adressage CIDR, les préfixes restent dans tous les cas la propriété des opérateurs. Ils ne peuvent plus être attribués "à vie" aux équipements. Pour faciliter la renumérotation d'une machine l'attribution d'une adresse à une interface est faites temporairement, les adresses IPv6 ne sont pas données mais prêtées. Une durée de vie est associée à l'adresse qui indique le temps pendant lequel l'adresse appartient à l'interface. Quand la durée de vie est épuisée, l'adresse devient invalide, elle est supprimée de l'interface et devient potentiellement assignable à une autre interface. Une adresse invalide ne doit jamais être utilisée comme adresse dans des communications. La valeur par défaut de la durée de vie d'une adresse est de 30 jours, mais cette durée peut être prolongée, ou portée à l'infini. L'adresse lien-local a une durée de vie illimitée [9].

V. Principales extensions apportées par IPv6

IPv6 a été conçu comme une évolution d'IPv4 et non comme un changement radical du protocole IP. Par conséquent, les applications fonctionnant sous IPv4 devraient fonctionner normalement sous IPv6.

Les changements entre IPv4 et IPv6 peuvent être classées de la manière suivante :

- ✓ capacités d'adressage et de routage étendues : la taille des adresses passe de 32 à 128 bits ;
- ✓ introduction d'un nouveau type d'adressage appelé anycast permettant d'identifier un groupe des machines et un paquet envoyé à une adresse anycast est délivré à une des machines appartenant au groupe désigné par l'adresse ;
- ✓ simplification du format de l'entête de paquet : pour réduire le coût de traitement des entêtes, certains champs ont été supprimés et d'autres sont rendus optionnels ;

- ✓ possibilités de définition de la qualité de service demandée par certains types d'applications (les applications temps réel notamment) ;
- ✓ capacités d'authentification et de confidentialité [6].

VI. Avantages et Inconvénients d'ipv6

VI.1. Avantages

- ✓ Espace d'adresses abondant.
- ✓ Configuration automatique.
- ✓ Mobilité.
- ✓ Sécurité IP intégrée.
- ✓ Communications de bout en bout.
- ✓ Extensibilité pour de nouvelles fonctions.
- ✓ Nouvelles opportunités commerciales.
- ✓ Rend possible l'« Internet des objets », à savoir un Internet global interconnectant les objets du quotidien et dominé par les communications de machine à machine.
- ✓ Améliorations sur le plan de l'automatisation, la productivité et l'efficacité
- ✓ Réduction stratégique des coûts.
- ✓ Engendre des modèles d'entreprise innovants dans de nombreux secteurs.

VI.2. Inconvénients

- ✓ La transition vers IPv6 peut entraîner des investissements
- ✓ Existence de techniques permettant de faire face aux limites d'IPv4
- ✓ Problèmes potentiels de sécurité et de confidentialité liés à la suppression des traductions d'adresses réseau (NAT)
- ✓ Peut impliquer des changements pour les modèles d'entreprise existants
- ✓ La durée de vie d'IPv4 peut être prolongée grâce à des améliorations supplémentaires permettant de remédier à l'épuisement des adresses
- ✓ Agrandissement potentiel de la table de routage globale si les adresses IPv6 ne sont pas attribuées avec diligence
- ✓ Les adresses IPv6 ne sont pas intuitives ; cependant, la configuration automatique facilite l'attribution des adresses [10].

VII. Mobilité dans IPv6

La mobilité dans IP s'adresse aussi bien à la mobilité des ordinateurs de bureau portables, qu'aux équipements enfouis (embarqués) dans les voitures, avions, etc.... Pour faciliter la mobilité des équipements, IPv6 offre la possibilité à un équipement de maintenir une connexion avec son adresse de base (adresse mère), tout en se déplaçant. Avant de partir en déplacement, les utilisateurs pourront demander à leur routeur de détourner leur trafic vers une adresse externe au sous-réseau. L'adresse externe est recalculée pour chaque sous-réseau externe visité. Cela permet de ne pas toucher aux entrées de DNS (Domain Name Service) pour retrouver les objets, en cas de mobilité.

Un mobile est toujours identifié par son adresse principale (appelée aussi adresse mère). Tant que le mobile se trouve dans son sous-réseau d'origine (sous-réseau mère), les paquets qui lui sont destinés sont délivrés en utilisant les mécanismes de routage conventionnels (c'est-à-dire en utilisant le préfixe de réseau). Lorsque le mobile est rattaché à un sous-réseau étranger, il devient joignable par une ou plusieurs adresses temporaires, en plus de son adresse mère. Les adresses temporaires sont obtenues par le mécanisme d'auto-configuration. La liaison entre une adresse mère et une adresse temporaire est appelée association.

Lorsqu'un mobile envoie un paquet alors qu'il se trouve hors de son sous-réseau mère, il positionne généralement comme adresse source une de ses adresses temporaires et ajoute dans une option destination son adresse principale.

Pour supporter la mobilité, il est nécessaire de disposer de structures de données qui servent à maintenir les associations des mobiles. L'environnement informatique d'un mobile est différent de celui des environnements informatiques habituels. En particulier, dans beaucoup de cas, les mobiles sont connectés au réseau sans fil, ce qu'il rend particulièrement vulnérables aux écoutes et aux différentes attaques. Il est parfois nécessaire de caché la position d'un mobile. [6]

VIII. Comparaison entre IPv4 et IPv6

Du fait que le protocole IPv6 apporte un nombre conséquent de nouveauté et qu'il est toujours en développement, une comparaison exhaustive entre les deux protocoles ne peut être envisagée. Cette section a pour but de mettre en avant les changements qui seront les plus évidents pour les administrateurs réseaux.

VIII.1. Différences au niveau des trames

La première chose que nous remarquons dans l'étude de la trame IPv6 concerne la taille des adresses IP. En effet, celles-ci sont désormais codées sur 128 bits contre 32 en IPv4. Cela permet donc d'allouer en théorie 340 282 366 920 938 463 374 607 431 768 211 456 adresses contre 4 294 967 296 pour IPv4. Cette très large plage d'adresse permet de résoudre le problème énoncé dans l'historique, à savoir une pénurie d'adresses IP même si on peut se demander pourquoi avoir pris un chiffre aussi grand (on peut ainsi attribuer 1024 adresses IP par mètre carré sur toute la surface du globe).
La deuxième chose que l'on constate est que le nombre de champs obligatoires dans l'entête de la trame a diminué : 6 dans IPv6 contre 13 dans IPv4. Certains de ces champs ont été remplacés par d'autres, ont été rendu optionnels voire supprimés. Cela permet un allègement de charge pour les routeurs. Enfin, la disparition du champ « checksum » est également un fait notoire. En effet, ce système de contrôle d'erreur a fini par devenir inutile avec l'évolution du matériel, qui devient de plus en plus fiable mais également parce qu'il est redondant, en effet, les protocoles des couches supérieures (comme TCP) et inférieures (comme Ethernet) incluent eux aussi un système permettant de contrôles l'ensemble du datagramme [11].

VIII.2. La fragmentation

La fragmentation consiste à découper un datagramme en datagrammes de plus petite taille. Ceci est dû aux différentes technologies employées par les réseaux, en effet, un réseau Ethernet et un réseau token ring par exemple ne transmettent pas des paquets de même taille. Cette taille est appelée MTU (Maximum Transfert Unit). Aussi, si un paquet doit passer sur un réseau dont le MTU est inférieur au réseau qu'il quitte, il doit être découpé de manière à pouvoir transiter sur ce réseau.
En IPv4, ce découpage est effectué par les éléments d'interconnexion telle que les routeurs ou les ponts. Cela impose une charge de travail considérable à cet équipement aussi, en IPv6, cette fragmentation est réalisée directement par l'émetteur qui doit découvrir par lui-même quel sera le plus petit MTU jusqu'au destinataire. IPv6 nécessite pour fonctionner correctement que la technologie employée pour la transmission garantisse une taille minimale de MTU de 1500 octets [11].

IX. Opportunités de IPv6

Les ressources IP sont à nouveau abondantes avec l'arrivée d'IPv6. L'équité dans l'accès à ces ressources au niveau mondial se trouve de fait rétablie. De ce point de vue, grâce à IPv6, l'innovation se trouve favorisée et l'économie numérique stimulée. C'est sans doute le plus l'avantage le plus important et le plus concret d'IPv6, l'exercice de recherche d'une « application qui tue » (killer application) s'étant révélé vain.

Alors que pour certaines technologies, il y a une bataille sur les ressources essentielles, pour IPv6, il en est autrement. Les adresses IPv6 étant abondantes, la bataille se situe plutôt au niveau de la maîtrise de la technologie IPv6 elle-même et de la disponibilité à temps des produits et services innovants qui s'y appuient [5].

X. IPv6 une longue phase de cohabitation avec IPv4 est inévitable

La prochaine version du protocole Internet va peu à peu s'immiscer dans les réseaux existants. La cohabitation avec IPv4 peut prendre plusieurs formes.

IPv6, la prochaine version du protocole Internet, ne va pas envahir les réseaux d'entreprise du jour au lendemain. Cependant, même si elle va mettre des années à se généraliser, et si les entreprises n'en ressentent pas encore le besoin, son avènement est inéluctable. Il faut donc, d'ores et déjà, se préparer à une longue phase de cohabitation avec IPv4.Au-delà du fait qu'IPv6 intègre des mécanismes de QoS et d'identification de l'émetteur de chaque trame à la façon d'IPSec, le nouveau protocole repousse surtout les limites d'adressage de son prédécesseur, avec une adresse codée cette fois sur 128 bits ($3,4 \times 10^{38}$ adresses), contre 32 bits (environ 4,29 milliards) auparavant. Cependant, les limites de l'espace d'adressage d'IPv4 ne constituent pas encore un moteur à l'adoption d'IPv6 en France. Tous les opérateurs ont des essais en cours, et certains proposent déjà des services IPv6, comme Nerim. Les sites ayant déployé la prochaine version du protocole IP sont, aujourd'hui, essentiellement des organismes de recherche et des universités.

Pour l'entreprise commerciale lambda, le besoin ne se fait pas encore ressentir. Il viendra avec de nouvelles applications qui requièrent de s'adresser directement au poste de l'utilisateur avec des communications de bout en bout ?" ce qui est remis en cause par les mécanismes de traduction d'adresses (NAT) largement déployés avec IPv4 ?", ou pour la prise en compte de la mobilité.

Plusieurs mécanismes pour marier les deux protocoles

IPv6 intègre en effet, en standard, de nombreuses fonctions qui étaient optionnelles avec IPv4, dont celles liées à la mobilité. Dans certains secteurs d'activité, IPv6 est déjà justifié, par exemple chez les fournisseurs de services d'hébergement : ces sociétés apprécieront de ne plus avoir à gérer les plans d'adressage identiques des réseaux privés de leurs clients. Pour faire cohabiter IPv6 avec les réseaux IPv4 existants, plusieurs méthodes sont disponibles.

Lorsque des communications IPv6 doivent traverser une infrastructure IPv4 (ou éventuellement l'inverse), des tunnels transportent IPv6 encapsulé dans des trames IPv4 (ou vice versa). Une solution plutôt destinée à des usages ponctuels, la traduction d'adresses permet, par exemple, d'éviter de faire passer à IPv6 un hôte que l'on ne souhaite pas faire évoluer sur le long terme.

Le mécanisme, de plus en plus employé ?" avec la maturation des premiers projets IPv6 ?", car mieux adapté aux déploiements à grande échelle, consiste à s'appuyer sur une infrastructure exploitant aussi bien IPv6 qu'IPv4. Cette technique est désormais utilisée par la plupart des routeurs de haut de gamme grâce à la présence d'une double pile IP.

- **Traduction d'adresses.** Une adresse IPv6 ne pouvant être placée dans un en-tête IPv4, une passerelle puise dans une base d'adresses IPv4 et les attribue dynamiquement.
- **Tunnels, avec encapsulation de paquets.** Configurés, de routeur à routeur, ils sont paramétrés à la main. Automatiques, d'hôte à hôte, ils exigent une adresse IPv4 au niveau de ces derniers.
- **Double pile IP**, en déployant des routeurs sachant gérer les deux protocoles, comme si l'on disposait de deux infrastructures réseau superposées.

Conclusion

Les évolutions technologiques ne peuvent utiliser celui-ci car il ne présente pas les fonctionnalités nécessaires. IPv6 a été conçu spécialement pour cela.

IPv6 est aussi le résultat d'un certain recul pris sur les réseaux et l'internet et les besoins futurs sont largement pris en compte ne serait-ce que concernant le nombre d'adresse IP possible, ce qui garantit ainsi qu'il sera très difficile de revenir la situation de pénurie actuelle.

L'en-tête IPv6 grâce à son champ option reste complètement ouvert à l'intégration d'extensions supplémentaires dans l'avenir. Ceci garantit à IPv6 une survie et une adaptabilité accrue pour les besoins futurs.

Le passage à IPv6 ne pourra se faire que de façon progressive, mais pratiquement toutes les technologies sont prêtes à l'utiliser qu'il s'agisse des ordinateurs avec les systèmes d'exploitation compatible IPv6 ou bien les équipements réseaux. Il ne reste plus qu'aux fournisseurs d'accès à franchir grandement le pas en suivant les quelques exemples déjà existants.

Nous l'avons évoqué durant ce chapitre, plusieurs avantages de protocole IPv6, parmi ses avantages on à la mobilité des ordinateurs, dans le chapitre suivant on vas le détailler.

Chapitre II : La mobilité IP

I. Introduction

L'évolution des technologies a rendu possible l'apparition de nouveaux types de machines. Ces appareils connus sous le nom de portables permettent à leurs utilisateurs de travailler sur différents sites. Les utilisateurs de ces ordinateurs ont de plus en plus besoin de communiquer alors qu'ils sont en déplacement. Jusqu'à présent ces liaisons s'effectuaient lorsque le portable se trouvait dans un lieu équipé, par exemple d'une connexion réseau ou d'une prise téléphonique. On parle alors d'ordinateurs nomades. Les communications sont stoppées dès que ces derniers sont débranchés de la prise réseau.

L'arrivée de nouvelles technologies de transmission de données sur des réseaux sans fil, permet de ne plus voir les ordinateurs portables comme de simples machines nomades. A l'heure actuelle, il est possible pour un ordinateur portable de communiquer même pendant ses mouvements d'où l'appellation d'ordinateurs mobiles.

La majorité des applications existantes est développée et conçue pour utiliser le protocole IP. En effet, celui-ci offre à une machine la possibilité d'être à la fois reliée à son réseau local, mais également de pouvoir dialoguer avec n'importe quelle autre machine sur l'Internet.

Le but recherché actuellement est d'offrir à ces ordinateurs mobiles la possibilité de se servir de l'Internet afin de communiquer avec d'autres machines, que celles-ci soient fixes ou mobiles. Comme le protocole IP a été conçu bien avant l'apparition de ces nouvelles technologies, ses fonctionnalités ne pouvaient tenir compte de ce nouveau type d'utilisation. Il a donc été nécessaire d'enrichir le protocole afin que celui-ci permette le support de la mobilité.

Ce chapitre a pour objectif de décrire comment la mobilité a tiré parti d'IPv6 pour améliorer les mécanismes définis dans IPv4. Il introduit la vision IETF de la mobilité et les différents éléments du réseau qui en découlent. Sur la base de scénario, il décrit les flux de signalisation et de données, échangés lors des épisodes de mobilité

II. Recherche de retours d'expérience

En un premier temps, nous avons voulu répertorier les rapports d'expérience sur l'implémentation de la pile KAME.

Les seuls documents suffisamment descriptifs étaient complètement obsolètes. Nous avons réussi à trouver quatre documents de référence :

1. Le premier protocole d'installation disponible est fourni avec le snap hebdomadaire de KAME. Ce document est très succinct et n'est pas été mis à jour avec les nouvelles fonctionnalités de FreeBSD et de KAME. Malgré tout, celui-ci nous a beaucoup aidés à découvrir l'arborescence du snap.
2. Par la suite, nous avons consulté le KAME Mobile IPv6 How To [14]. Ce document décrit pas à pas les instructions pour utiliser les fonctionnalités Mobile IPv6 de KAME. La dernière version publiée datait du 7 octobre 2003, ce qui rend le document complètement désuet. Mais grâce à ce manuel d'utilisation nous avons pu réaliser nos premiers pas avec KAME.
3. SHISA est une implémentation de Mobile IPv6, désormais intégrée à KAME. Le projet WIDE a décidé de regrouper les deux branches de Mobile IPv6 pour travailler ensemble vers une implémentation simple (appelée SHISA) au printemps 2004. SHISA supporte les Mobile Node, Home Agent et correspondent Node. Le projet SHISA est développé pour les plateformes KAME étant sur FreeBSD, NetBSD, OpenBSD. Mais son utilisation est optimale sur FreeBSD. De plus, cette implémentation nous a éclaircis sur les scripts de démarrage qui permettent d'actionner les options nécessaires au fonctionnement de Mobile IPv6.
4. Dans un dernier temps, nous avons étudié le rapport d'expérience d'un projet de fin d'études d'Elodie Leocmach & Jérôme Meyer. deux étudiants ont rédigé un document relatant l'installation et la configuration de leur réseau Mobile IPv6. La différence principale est qu'ils ont travaillé sur la version 5.3 de FreeBSD,

II. La Mobilité IPv4

L'IETF est composé de groupes de travail qui s'occupent chacun d'un domaine bien précis. L'un d'entre eux (Mobile IP) est chargé de proposer un protocole pour le support des mobiles sur l'Internet. Ce modèle est à l'heure actuelle reconnu comme le standard et de nombreux additifs de plusieurs groupes de recherche existent. C'est un protocole de niveau Réseau qui offre la caractéristique de garder une adresse IP unique quelle que soit la position géographique du mobile. Cette dernière Caractéristique (adresse IP unique quelle que soit la position du mobile dans le monde), n'est pas reconnue par l'ensemble des chercheurs travaillant sur les mobiles, car elle dénature le sens même de l'adresse IP (qui est de permettre de localiser une machine dans le monde grâce à son adresse Réseau). Un hôte mobile, que nous notons MH, se déplace de réseau en réseau et s'accroche à un point d'attachement appelée base. La base implémente des fonctionnalités de niveau 2 du modèle OSI. Elle assure une connectivité de niveau liaison de données. Elle permet l'échange d'informations avec un mobile par un canal radio ou bien infrarouge.[Thomas NOËL, Sébastien WOIRGARD, 2002].

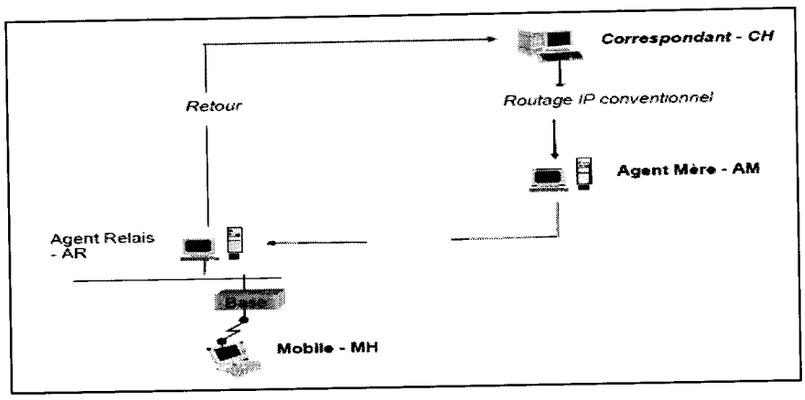


Figure II. 1: La mobilité sur IPv4.

Légende :
MH ou Mobile : Machine ou routeur changeant de point d'attachement sur l'Internet. Comme toute machine fixe, un mobile appartient initialement à un réseau sur l'Internet. C'est ce réseau, appelé réseau mère, qui a affecté son adresse IP au mobile.
AM ou Agent Mère : Routeur, situé dans le sous réseau mère d'un mobile, qui maintient un registre des fixations courantes des mobiles dont il a la charge. Il peut encapsuler les paquets de données pour les délivrer au MH tant que celui-ci est éloigné de son réseau mère.
CH ou Correspondant : Ordinateur (mobile ou non) désirant dialoguer avec un mobile.
AR ou Agent Relais : Routeur situé sur le réseau auquel est attaché un mobile, lors de l'un de ses déplacements. Il permet notamment de relayer les paquets vers le mobile.

Le partenaire d'une communication avec un mobile est appelé le correspondant (Figure II.1), il peut s'agir soit d'une machine fixe, soit d'un autre mobile. Quand un correspondant veut envoyer des données à un mobile, il se contente de formater un paquet IP ordinaire. L'adresse IP source sera celle du correspondant, l'adresse IP destination sera celle du mobile. Ces paquets arriveront, en utilisant le routage IP conventionnel sur le Réseau mère du mobile. A ce stade, un routeur appelé l'Agent Mère (AM) intercepte les dits paquets et les transmet vers la position courante du mobile. Du point de vue des protocoles Réseau, l'Agent Mère garde trace de la position courante des mobiles qu'il sert. La mobilité IPv4 définit deux moyens pour délivrer des paquets depuis l'Agent Mère vers un mobile. Le premier consiste à passer par un nœud intermédiaire appelé *Agent Relais* (AR). Ce nœud, se trouve sur le réseau visité par un ordinateur nomade, il sert de nœud relais entre l'Agent Mère et le mobile. C'est lui qui réceptionne les paquets envoyés par l'Agent Mère et les délivre au mobile. Cette solution a l'avantage d'éviter d'allouer une nouvelle adresse IP au mobile. En effet, celui-ci utilise toujours l'Agent Relais situé dans le réseau visité comme nœud intermédiaire. Ce mécanisme peut être comparé à un système de translation d'adresses, connu également sous le nom de NAT (Network Address Translation). La deuxième alternative proposée par l'IETF est d'utiliser un mécanisme d'auto configuration d'adresses. Cette solution permet à un mobile d'acquérir une adresse temporaire dans le nouveau réseau d'attachement. Dans ce cas, l'Agent Mère ne relaie plus les paquets vers un Agent Relais, mais envoie directement les paquets vers l'adresse temporaire du mobile. Cette alternative requière toute fois la réservation d'un pool d'adresses pour la gestion des mobiles dans un réseau.

Quelle que soit la solution utilisée, l'Agent Mère doit encapsuler les paquets interceptés pour les rediriger vers le nouveau réseau visité. En effet, il est inconcevable que l'Agent Mère modifie les paquets interceptés, cela poserait plusieurs problèmes. L'un d'entre eux, et non le moindre, obligerait l'Agent Mère à recalculer pour chaque paquet, le « *checksum* » qui se trouve dans les couches supérieures. L'encapsulation des paquets consiste à insérer un nouvel en-tête IP devant l'en-tête existant. Ce procédé permet de continuer à utiliser les mécanismes de routage IP conventionnels tout en permettant une redirection des paquets. Le nouvel en-tête (cf. Figure II.2), appelé en-tête IP d'encapsulation, contient comme adresse source l'adresse de l'Agent Mère et comme adresse destination, l'adresse du nœud relais (ou l'adresse temporaire du mobile) : [Thomas NOËL, Sébastien WOIRGARD, 2002].

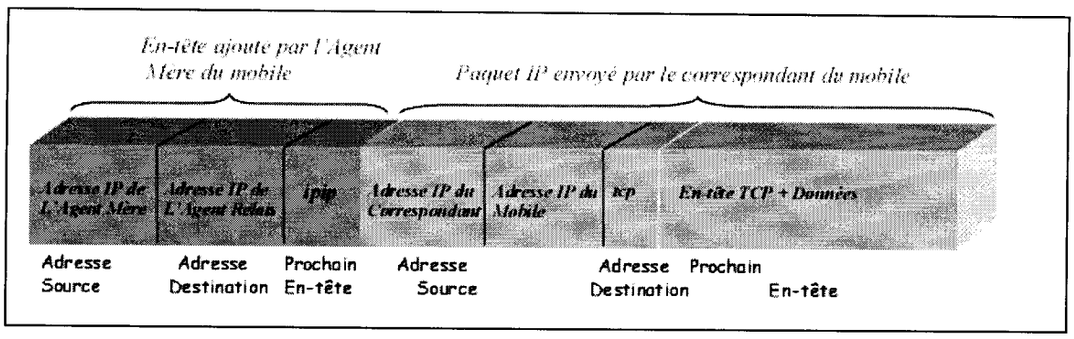


Figure II. 2: Encapsulation IP dans IP.

III. La Mobilité IPv6

III.1. Principes

Mobile IPv6 permet à un nœud de changer de réseau tout en gardant d'une façon transparente ses connections déjà établies. Chaque nœud est toujours identifié par son adresse IP d'origine. Un nœud spécial sur le réseau d'origine appelé Home Agent intercepte ainsi les paquets destinés au nœud mobile et les lui renvoie.

Ce mécanisme est complètement transparent aux couches supérieures (ainsi un nœud conserve son adresse IP d'origine dans les entrées DNS[11] par exemple).Le Home Agent peut rediriger les paquets vers le nœud car il connaît son emplacement actuel représenté par une adresse IP temporaire, intitulée Care-of Address. En effet le nœud informe son Home Agent qu'il a changé d'emplacement en lui fournissant sa nouvelle adresse. Nous allons dans la suite détailler tous les messages et les mécanismes mis en jeu. [12]

III.2 -Terminologie

Le Mobile Node possède une adresse permanente appelée Home Address dans son réseau mère. Il est associé à un routeur spécifique du réseau mère, son Home Agent. Pour le joindre dans le réseau mère, le routage IP classique sur l'adresse mère est utilisé. Par contre dans le réseau visité, il récupère une adresse temporaire (Care-of Address) qu'il signale à son Home Agent pour maintenir l'association adresse mère/adresse temporaire.

Le Home Agent maintient les associations adresse mère/adresse temporaire des Mobile Node connectés sur des réseaux visités. Il intercepte les paquets destinés aux Mobile Node absents du réseau mère tel le fonctionnement d'un proxy. Et dans un

[11] Le Domain Name System (ou DNS, système de noms de domaine) est un service permettant de traduire un nom de domaine en informations de plusieurs types qui y sont associées, notamment en adresses IP de la machine portant ce nom

dernier temps, il transfère ces paquets vers les adresses temporaires des Mobile Node. Le Correspondent Node peut être tout nœud Internet (station, serveur, routeur) fixe ou mobile. Il émet des paquets vers un Mobile Node en utilisant l'adresse mère de celui-ci. Le Correspondent Node peut ou non implémenter la mobilité [12].

III.3. Nouvelles options destination d'IPv6

Pour faire la correspondance entre l'adresse mère et l'adresse temporaire d'un mobile, une association (binding) stockée dans les caches des nœuds IPv6 est utilisée. Chaque nœud IPv6 maintient un cache des associations. De plus, chaque mobile maintient une liste des mises à jour des associations pour enregistrer les informations pour chaque Binding Update envoyé à leurs correspondants. Une entrée reste valide pendant la durée de validité de l'association. De plus, un Home Agent possède une adresse temporaire principale par mobile.

Le Home Agent stocke dans une table la correspondance entre les adresses mère et temporaire du mobile. Par la suite, le Home Agent intercepte les paquets destinés à l'adresse mère du mobile et les redirige vers la position courante du mobile en utilisant des mécanismes d'encapsulation.

Il y a quatre nouvelles options Destination :

- ✓ Binding Update (mise à jour de l'association) pour qu'un mobile puisse informer son Home

Agent ou ses correspondants de sa nouvelle adresse temporaire de manière sécurisée. L'adresse source d'un tel paquet doit être l'adresse mère du mobile ;

- ✓ Binding Acknowledgment (acquittement de l'association) pour acquitter, de manière sécurisée, un Binding Update. Un paquet contenant cette option est adressé à l'adresse mère du mobile en utilisant un en-tête de routage contenant l'adresse temporaire du mobile ;
- ✓ Binding Request (demande de mise à jour de l'association) pour demander à un mobile d'envoyer un Binding Update. Cette option est donc uniquement utilisée lorsqu'un correspondant pense que son association va bientôt expirer. Généralement cette option n'est utilisée que si le correspondant est en communication avec le mobile à ce moment-là.
- ✓ Home address (adresse mère) : cette option est incluse dans chaque paquet IPv6 envoyé du mobile au correspondant. Cette option est nécessaire pour que le correspondant considère que le paquet reçu ne provient pas du réseau visité

(adresse source contenue dans le champ source du datagramme IP) mais du réseau mère. L'avantage est que les correspondants sont capables de référencer les mobiles de façon non ambiguë, c'est à dire par leur adresse mère. Malheureusement, cette option nécessite 144 bits supplémentaires dans l'en-tête de tous les paquets, chiffre qui diminuera peut être grâce à la compression d'en-tête [12].

III.4. Fonctionnement de Mobile IPv6

a. Enregistrement auprès du Home Agent

Dès qu'un nœud détecte qu'il est dans un nouveau réseau, il procède à l'auto configuration d'une nouvelle adresse IPv6. Cette nouvelle adresse est la Care-of Address du nœud. Il envoie alors à son Home Agent un Binding Update (BU) contenant cette nouvelle adresse pour lui permettre de faire l'association avec l'adresse d'origine du nœud. Le Home Agent répond par un Binding Acknowledgment (BA).

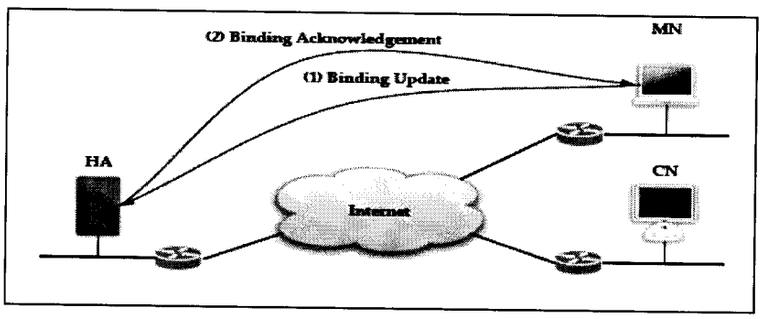


Figure II. 3 : Enregistrement du MN dans le HA

Quand un correspondant envoie un paquet, le Home Agent vérifie dans son cache des associations les entrées dont il dispose pour l'adresse destination du paquet. Si une telle entrée existe, le paquet est envoyé sans en-tête de routage à destination de l'adresse mère du mobile pour qu'il soit encapsulé1 par le Home Agent vers la position courante du mobile.

Finalement, il est important de ne pas rompre les communications lors d'un changement de point d'accès au réseau. C'est pourquoi les adresses mères (constantes au cours du temps) sont les seules adresses qui référencent les mobiles au niveau des couches supérieures puisque les adresses temporaires sont susceptibles de changer au cours d'une communication [12].

b. Mode Tunnel

Ce mode repose sur le principe que tout le trafic passe par le Home Agent. Celui-ci crée un tunnel formant une encapsulation IPv6-over-IPv6. La non-modification de la configuration des correspondants est un avantage de cette méthode. De plus, une meilleure gestion de la sécurité est assurée grâce à l'interdiction de trafic direct entre le Correspondent Node et le Mobile Node. Malheureusement, le trafic important nécessite d'être redirigé par des Home Agents robustes. Le système peut alors être sujet à un déni de service [12].

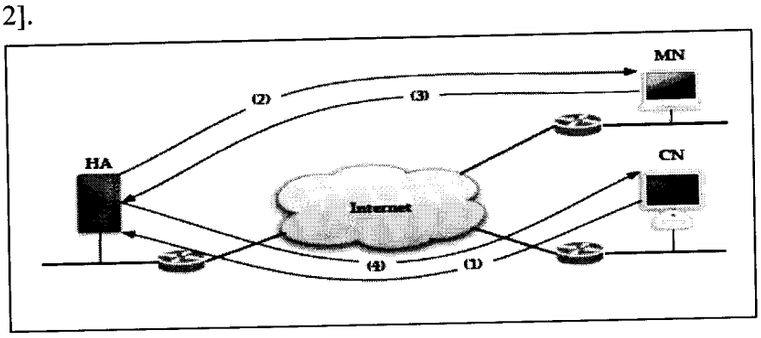


Figure II. 4: Mode tunnel

c. Routage triangulaire

Le travail du Home Agent consiste alors à intercepter les paquets destinés au nœud mobile. Cela est possible par la procédure suivante : envoi par le Home Agent d'un message Neighbors Avertissement sur son lien lui permettant d'associer son adresse MAC avec l'adresse IP d'origine du nœud mobile. Tout paquet destiné au nœud est ainsi reçu par le Home Agent qui se charge d'envoyer ce paquet au nœud mobile en utilisant le mécanisme d'encapsulation IPv6. Le nœud mobile envoie ses paquets directement au correspondant en utilisant comme adresse source sa Care-of Address (afin d'éviter le mécanisme d'egress filtering2), tout en incluant l'option Home Address afin de garantir la transparence de la mobilité.

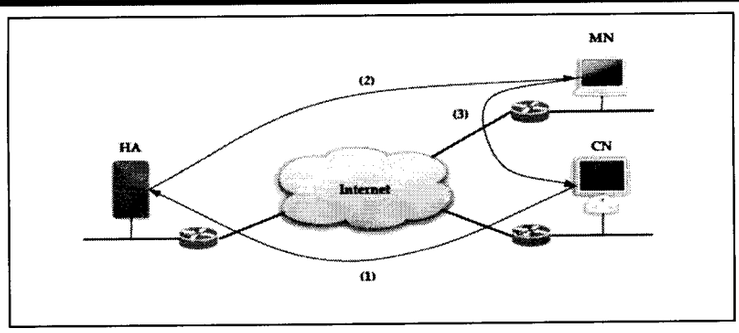


Figure II. 5: Le routage triangulaire

d. Optimisation de route

Pour éviter ce routage triangulaire, un nœud mobile peut envoyer un BU à tout nœuds correspondant, qu'il soit mobile ou stationnaire.

Ainsi le nœud correspondant effectue une association entre l'adresse d'origine du nœud mobile et sa Care-of Address. Dans ce cas, le nœud correspondant n'utilise pas l'encapsulation mais une autre méthode : avant d'envoyer un paquet le nœud consulte son cache pour trouver une association. Si elle existe, il envoie le paquet en y incluant l'option Routing Header. La route spécifiée dans cette option est constituée de deux sauts. Le premier est la Care-of Address du nœud, le second est l'adresse d'origine du nœud. Ainsi lorsque le nœud reçoit le paquet il l'envoie vers le saut suivant qui n'est autre que son adresse d'origine.

Le paquet fera donc un bouclage à l'intérieur de sa pile protocolaire garantissant ainsi la transparence aux couches supérieures. On évite ainsi une surcharge du Home Agent. Ce principe implique l'implémentation de la mobilité sur les correspondants ainsi que leur authentification. Ce principe de routage suppose donc également que le Correspondent Node implémente lui aussi Mobile IPv6.

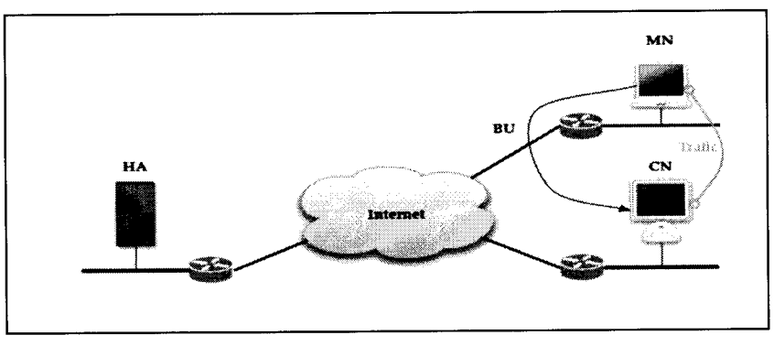


Figure II. 6: Optimisation de route

e. Détection du Home Agent

Mobile IPv6 fournit un mécanisme au nœud mobile de découvrir dynamiquement son Home Agent. Le nœud mobile se contente d'envoyer un message ICMPv6[12] intitulé Dynamic Home Agent Address Discovery Request avec comme adresse de destination l'adresse anycast dédiée aux Home Agents de son réseau d'origine. Ce message sera reçu par un des Home Agents présent sur le réseau. Le Home Agent envoie au nœud un message ICMPv6 de retour intitulé Dynamic Home Agent Address Discovery Reply en y incluant la liste de tous les Home Agents présents sur le réseau et leur ordre de priorité. Ainsi le nœud mobile est en mesure d'envoyer son BU à l'un de ces Home Agents.

[12] L'ICMP pour IPv6 (Internet Control Message Protocol Version 6) fait partie à part entière de l'architecture IPv6 et doit être complètement supportée par toutes les implémentations d'IPv6.
ICMPv6 est utilisé par les nœuds/hôtes IPv6 pour rapporter les erreurs trouvées dans le traitement de datagrammes, et pour effectuer d'autres traitements internes à cette couche, tels que des diagnostics (ICMPv6 « ping »).

IV. Conclusion

Le protocole proposé par l'IETF pour IPv4 permet de faire fonctionner les mobiles sur l'Internet tel qu'il existe actuellement. Il ne nécessite aucune modification en dehors des routeurs en charge des mobiles dans leur sous réseau mère. IPv6 permet au modèle de l'IETF de résoudre plusieurs problèmes [Kyntäjä, 1998]. En effet, la mobilité IPv4 définit un certain nombre d'extensions comme l'optimisation des routes pour éviter le routage en jambe de chien. Mais cette solution repose sur la modification des piles TCP/IP des machines dialoguant avec un mobile. Comme IPv6 est en cours d'élaboration, il est astucieux d'ajouter dès la création du protocole de nouvelles fonctionnalités lui permettant de supporter nativement, sur l'ensemble des nœuds sur l'Internet, le dialogue « optimisé » avec un mobile. De plus, ce modèle permet d'éviter l'utilisation d'un Agent Relais sur chacun des réseaux visités par un mobile. Les protocoles d'auto configuration présents dans les spécifications d'IPv6 permettent une gestion simplifiée de l'attribution d'adresses temporaires.

En conclusion, nous pouvons dire que l'utilisation du protocole Mobile IP devrait se répandre de plus en plus avec l'arrivée d'IPv6. Toutefois, l'ampleur de son utilisation dépendra également des équipements et des débits offerts par les fabricants d'équipements sans fil.

Le chapitre suivant contient les différentes informations sur le déploiement et le test de la mobilité.

Chapitre III : Partie 1 Déploiement

I. Choix de la pile Mobile IPv6

À l'heure actuelle, plusieurs implémentations de piles protocolaires existent pour Mobile IPv6 il nous a donc fallu faire un choix, en tenant certains paramètres en ligne de compte. Nous allons donc exposer les caractéristiques des piles existantes, puis les différents critères que nous avons considérés, pour enfin décrire la solution retenue.

I.1. Les piles existantes

I.1.1. KAME

Le projet KAME est un effort de plusieurs entreprises, ayant pour but la création d'un ensemble logiciel robuste, portant notamment sur IPv6 et IPsec. Des chercheurs provenant de plusieurs organisations japonaises se sont joints au projet. Cet effort évitera donc une inutile redondance de développement dans le même domaine, et générera une pile de grande qualité, avec des fonctionnalités avancées.

Le projet KAME a pour but de fournir des implémentations de référence libres de :

- IPv6 ;
- IPsec (pour IPv4 et IPv6) ;
- des implémentations de protocoles avancés tels que la mise en attente avancée de paquets, la mobilité, etc.

Sur des plateformes *BSD. À l'heure qu'il est, plusieurs plateformes sont supportées, parmi lesquelles FreeBSD, NetBSD et OpenBSD. Leurs codes respectifs sont développés séparément. [13]

I.1.2. MIPL

MIPL est une implémentation qui fût à l'origine un concours de développement logiciel à Helsinki University of Technology. Le but est de créer un prototype d'implémentation de Mobile IPv6 pour Linux. C'est une implémentation open source qui a été délivrée par GNU GPL. MIPL Mobile IPv6 pour Linux, jusqu'à sa version 1.1, était développé par le projet GO-Core à l'Université Technologique d'Helsinki. La version 1.1 est une implémentation de niveau noyau de la norme Mobile IPv6. Elle ne supporte pas IPsec.

MIPL Mobile IPv6 pour Linux version 2 (et ultérieure) est développée par le projet GO-Core en coopération avec le projet USAGI/WIDE. GO-Core et USAGI ont co-développé des extensions à la pile IPv6 de Linux afin de supporter Mobile IPv6.

L'objectif est d'intégrer, à terme, ces modifications dans l'implémentation de niveau noyau [13].

I.1.3. Cisco

L'implémentation Cisco a commencé avec la collaboration de Lancaster University pour Linux. Celle-ci n'est pas encore en production mais est en période de tests via la Cisco IPv6 support team. La recherche porte principalement sur le Home Agent et le Correspondent node. Nous n'avons pas retenu cette pile puisqu'elle est actuellement peu disponible et trop instable pour être testée en conditions réelles [13].

I.1.4. Microsoft

Cette implémentation a été produite en collaboration avec Lancaster University. L'implémentation pour Linux a été portée pour Windows 2000 ; elle est valide en exécutable et en format code source gratuitement, sous réserve de ratification de la clause de non-diffusion. Nous n'avons donc pas considéré cette pile, en raison des difficultés administratives qu'aurait posées cette clause [13].

I.2. Critères d'évaluation

I.2.1. Proxy Neighbors Discovery

Le processus de Proxy Neighbors Discovery a lieu lorsque le MN est hors du réseau mère ; le HA doit intercepter les paquets destinés au MN, et doit donc agir en tant que proxy, en effectuant la correspondance entre l'adresse physique du MN avec sa propre adresse IPv6. Ce mécanisme est indispensable dans un routage en mode tunnel, qui est la solution que nous avons retenue. Par conséquent, il est essentiel qu'il soit fonctionnel dans l'implémentation finale [13].

I.2.2. Encapsulation et décapsulation IPv6

L'encapsulation et la décapsulation IPv6 sont utilisées lorsque le HA communique avec le MN en mode tunnel (IPv6-over-IPv6). Pour la même raison que précédemment, ceci doit être pleinement fonctionnel [13].

I.2.3. DHAAD

Le mécanisme de Dynamic Home Agent Address Discovery est utilisé par le MN pour connaître l'adresse unicast du HA, en envoyant une requête à l'adresse anycast des HA de son réseau mère (suffixe ::fdff:ffff:ffff:fffe).

Le MN doit donc savoir acquérir cette adresse automatiquement, ce qui facilitera l'administration des MN en cas de renumérotation du HA du réseau mère [13].

I.2.4. Binding Management

On doit pouvoir facilement accéder à la liste des associations courantes, afin de faciliter l'administration [13].

I.2.5. Home Address Option

Lors de l'envoi de paquets depuis le MN, ce dernier doit indiquer son adresse au sein du réseau mère, dans un champ spécial du paquet IPv6 [13].

I.2.6. Détection de mouvement

Le MN doit détecter aussi rapidement que possible qu'il se trouve dans un réseau visité, ceci afin de paraître le plus transparent possible à l'utilisateur [13].

I.2.7. Smooth Handoff

Le MN doit effectuer son association aussi vite que possible, afin de limiter la perte de paquets engendrée par le déplacement [13].

I.2.8. IPsec

Le critère lié à la sécurité de la solution est crucial ; or il y a plusieurs implémentations d'IPsec, Nous étudierons donc les différents modes de protection offerts par les piles (secret partagé, Échange de clés, authentification, chiffrement, etc.) [13].

I.2.9. Échange de clés

Mécanisme de fonctionnement de l'échange de clés, s'il est disponible. Nous aborderons également la facilité de mise en œuvre [13].

I.2.10. Support des ordinateurs portables

Degré de support des piles Mobile IPv6 sur des ordinateurs portables ou Pocket PC [13].

I.2.11. Built-in MIP6

La pile est-elle intégrée au système d'exploitation ?

I.2.12. Patches

L'installation de patches est-elle nécessaire au bon fonctionnement de la pile ?

I.2.13. Mode d'installation

La pile est-elle facile à mettre en place ?

I.2.14. Stabilité

La pile offre-t-elle une stabilité suffisante pour un usage en production ?

I.3. Étude comparée : KAME vs MIPL

	MIPL	KAME
Proxy Neighbor Discovery	OK	OK
IPv6-over-IPv6	OK **(MIPL 0.9.1)**	OK
DHAAD	Supporté, mais problèmes de messages *anycast*	OK, **ainsi qu'assignement statique**
Home Address Option	OK, **mais doit être implémenté sur CN**	OK, **avec *fallback* sur HA pour être compatible avec autres CN**
Détection de mouvement	OK, **avec norme de base IPv6**	OK, **avec changement de CoA**
Smooth Handoff	OK	Non
IPsec	Authentification par secret partagé	OK, **implémentation optionnelle dans le noyau**
Echange de clés	OK, **MD5 ou SHA-1**	OK, **certificats X509 (Racoon)**
***Built-in* MIP6**	Non	Non, noyau à recompiler avec options

Figure III. 1: MIPL vs KAME

Nous avons donc choisi la pile KAME, étant donné les éléments de sécurité qu'elle apporte par rapport à la pile MIPL.

III. Le matériel et ses problèmes

La première difficulté du PFE fut de trouver les machines nécessaires à notre plateforme. Au début, nous étions équipés seulement d'un ordinateur pour le PFE. Ce PC était destiné, à la base, à une utilisation bureautique. Suite à la première réunion de projet, nous avons contacté Monsieur BENDIMERAD qui nous a confié son labo de recherche afin d'effectuer notre étude.

La deuxième difficulté était de trouver la bonne version de la pille KAME qui fonctionne correctement et sans bugs, on a essayé plus de vingt versions afin de trouver la bonne version adaptée avec FreeBSD version 5.4.

IV. Installation

Dans cette partie nous allons donner une procédure d'installation générale de la pile KAME sous FreeBSD 5.4 le snap utilisé est celui en date du 25 juillet 2005. Pour plus de détails voir (Annexe F).

Dans un premier temps nous avons récupérer le snap depuis l'adresse suivante :
http://ftp.naist.jp/pub/kame/snap/old/kame-20050725-freebsd54-snap.tgz

Après on a fait une préparation de d'un nouveau noyau et une installation des programmes utilisateur, afin de faire une installation de Shisa et modifier la configuration de démarrage, pour passer à l'étape de la mise en place des scripts de démarrage et enfin la mise en place de sécurité. Apres l'installation en passe vers le teste de cette installation.

Chapitre III : Partie 2
Tests

Après avoir réussi à configurer nos machines nous somme passé à la partie tests. Cette dernière consiste à tester la connectivité entre un CN et un MN dans différentes situations. Nous avons considéré le cas où le MN est dans son réseau mère ensuite le cas où ce même nœud se déplace vers un autre réseau (réseau visité).

I. Maquette de tests

Nous nous sommes basés sur une architecture simple du PFE IPv6, On a configuré un PC comme home agent, un autre pour le mobile routeur, et bien sur un autre pour le mobile correspondant. Pour le mobile node on a utilisé notre PC portable. Nous avons implémenté FreeBSD 5.4 avec un snap KAME (version 25 juillet) pour le mobile node et le home agent. Le routeur est connecté directement sur le VLAN. Il a été nécessaire de commander une nouvelle carte réseau supportant les tags VLAN. FreeBSD 5.4 est aussi installé sur la machine. Pour réaliser les fonctions de routage, nous avons configuré l'application Zébra (annexe E). Le routeur n'a pas besoin du dernier snap de KAME, la version stable intégrée à FreeBSD 5.4 suffit.

Nous avons utilisé un logiciel qui s'appelle VirtualBox qui permet de créer plusieurs machines virtuelles sur la même machine pour ne pas toucher les systèmes installés.

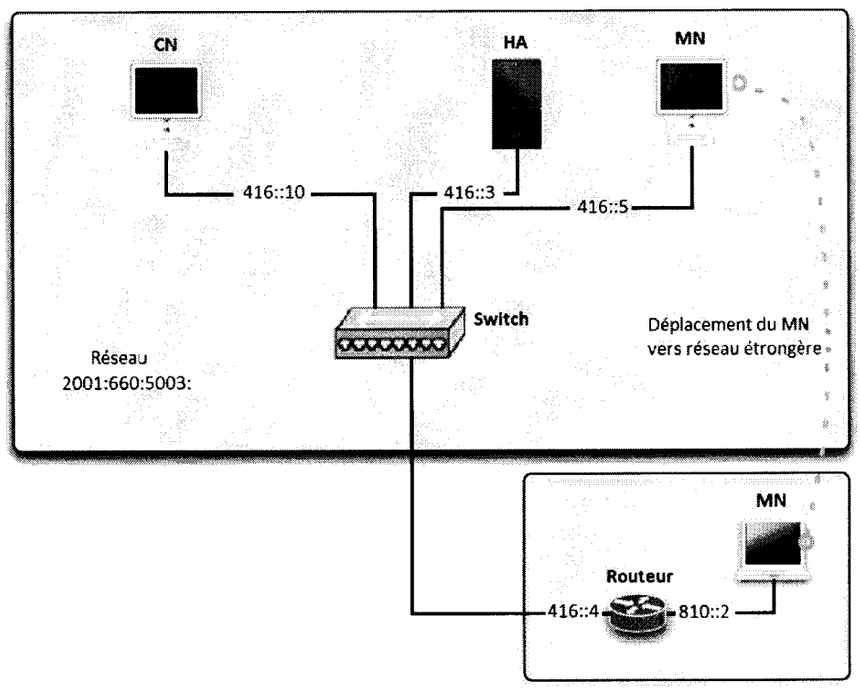


Figure III. 2 : Synoptique global de test

Le routeur basé sur une installation FreeBSD 5.4 et Zébra 0.95 (Annexe E), divise le réseau en deux sous-réseaux différents 416 et 810 après la modification du fichier de configuration /etc/rc.conf (Annexe D) avec l'utilisation de deux cartes réseaux. Au début le MN se trouve dans le même réseau avec le HA et CN (réseau 416) après on va brancher le MN dans le deuxième réseau c'est 2001 :660 :5003 :810 ::/64.

II. Scenario de tests

Le premier test consiste tout simplement à brancher le MN dans le réseau mère et d'en tester la connectivité, i.e. de faire un ping depuis le CN situé sur le même réseau 416, à destination de la HA du MN (adresse du MN dans le réseau mère) (Figure III.3)

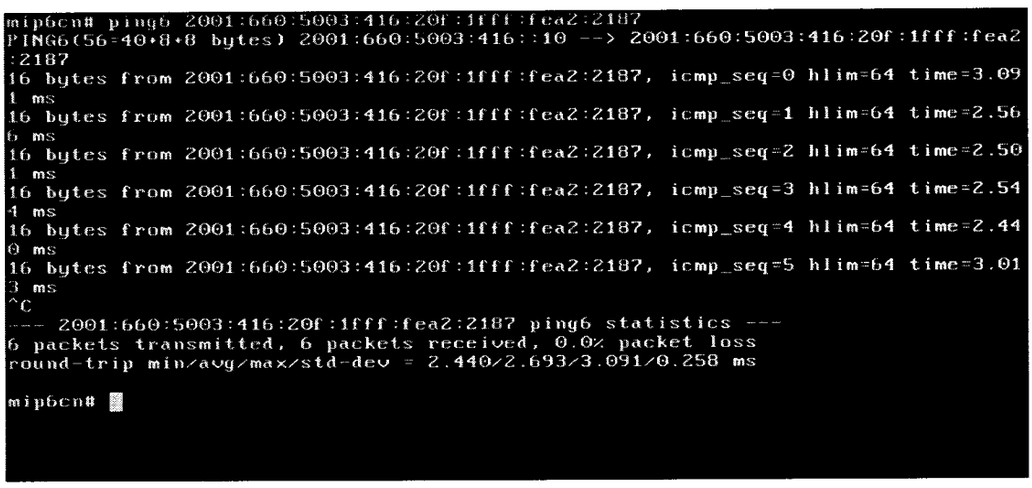


Figure III. 3: Test ping du CN vers MN

La connectivité est donc établie ; si on regarde dans la configuration réseau du MN (Figure III.4), on voit qu'il a bien acquis une adresse auto configurée sur le réseau mère

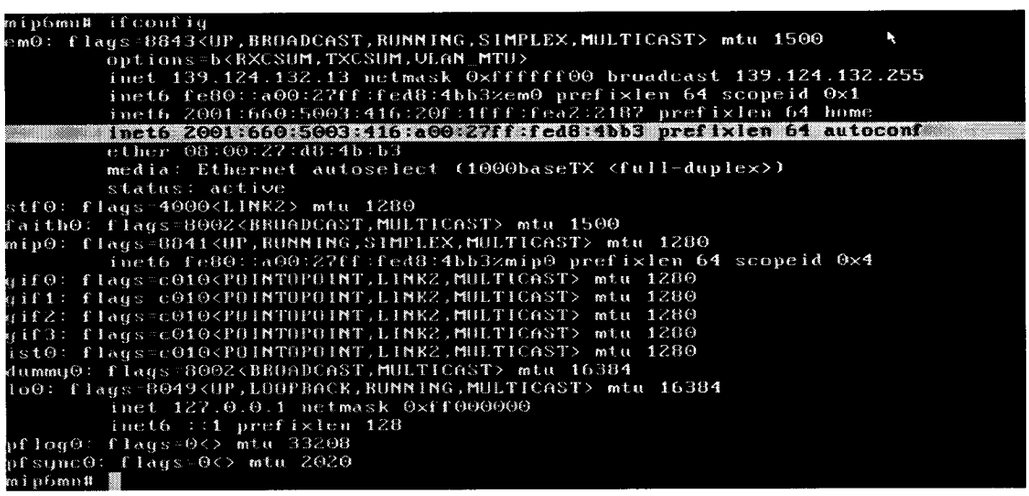


Figure III. 4 : Auto configuration au niveau de Mobile node

II.1. Déplacement du MN du réseau mère vers le réseau visité

Après le premier test, nous débranchons le MN du réseau 416, pour le brancher sur le réseau 810. Le nœud se rend compte qu'il a changé de réseau (grâce au Router Adversaire) ; il acquiert donc une Care-of Address par auto configuration, envoie une requête de DHAAD vers l'adresse anycast du HA, qui lui répond pour lui donner son adresse unicast. Le MN envoie au HA un Binding Update en lui indiquant sa CoA; ce BU est acquitté (Figure III.5).

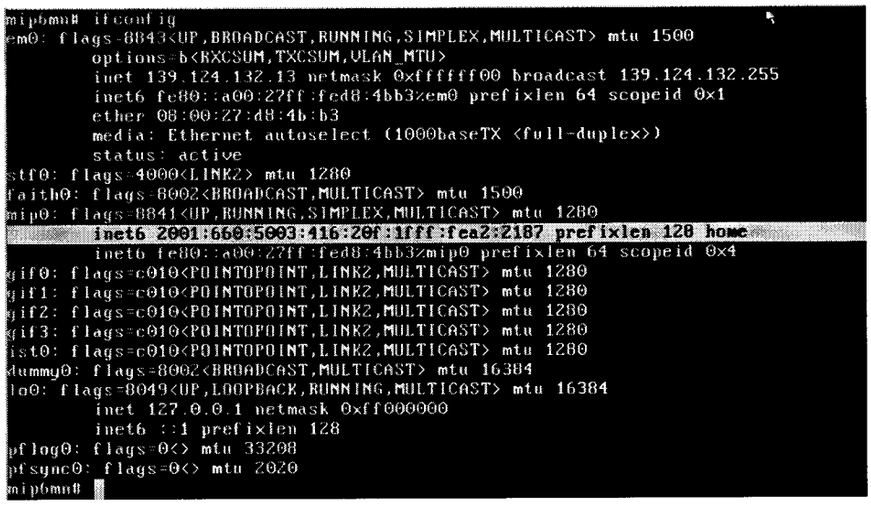


Figure III. 5 : Les adresses du mobile node avant branchement dans le réseau visité

On observe un changement au niveau de la configuration réseau (Figure III.6):

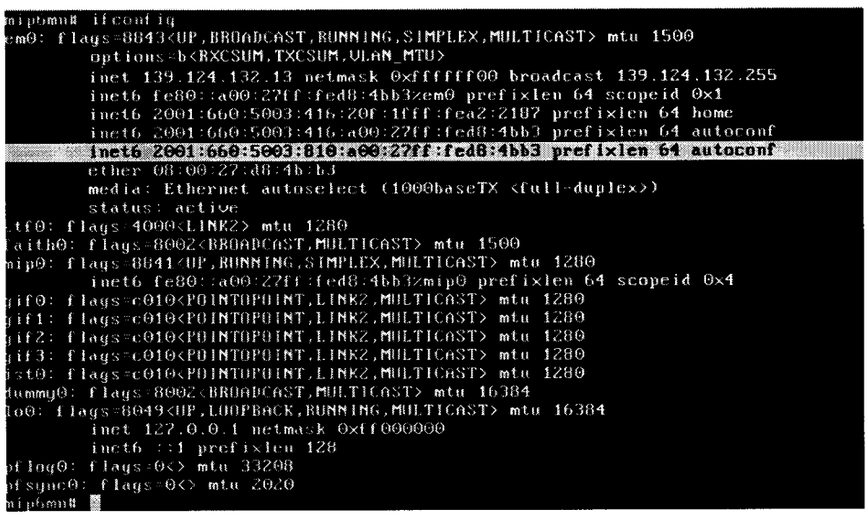


Figure III. 6 : Les adresses du mobile node après branchement dans le réseau visité

L'interface virtuelle mip0 a été activée, avec la HoA. L'interface em0, qui représente la carte réseau, est quant à elle configurée de manière automatique (autoconf), suivant le schéma classique IPv6.

Il faut ensuite s'assurer que le MN est joignable par sa Home Address. Nous faisons donc un ping de la même manière que précédemment, et nous parvenons bien à atteindre le MN. Il est à noter que pour le Correspondent Node, la manipulation est complètement transparente ; cependant, si on capture les paquets au niveau du réseau mère, on voit que le HA intercepte les paquets destinés à la HoA du MN, pour les rediriger vers la CoA du MN, avec l'en-tête de mobilité (Figure III. 7) :

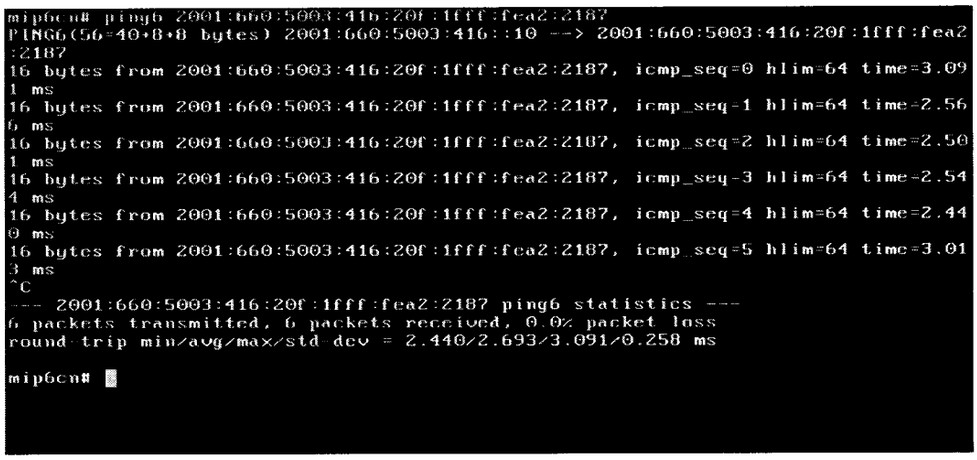


Figure III. 7 : Le deuxième Ping du CN vers MN

II.2. Retour du MN du réseau visité vers le réseau mère

Nous ramenons le MN dans le réseau mère, afin que celui-ci se désassocie au niveau du HA. Cette manipulation échoue si le temps écoulé entre la déconnexion du réseau visité et le branchement au réseau mère est inférieur à 40 secondes, soit la durée de vie de l'association au niveau du Home Agent. Dans le cas contraire, il n'y a pas de problème, le MN revient à sa configuration initiale. Ce problème ne devrait pas avoir d'incidence sur notre architecture, puisque le HA serait situé en cœur de réseau ; par conséquent, le MN va changer le réseau.

III. Impact sur le réseau universitaire

Après avoir étudié de manière approfondie le protocole de mobilité IPv6 nous proposons que ce protocole soit déployé au niveau de notre infrastructure universitaire. Cela aura deux objectifs essentiels : le premier étant de tester le protocole IPv6 de

manière permanente afin de pouvoir l'implémenter un jour au niveau de tout le réseau universitaire. Le deuxième étant d'assurer une mobilité des différents utilisateurs enseignants, chercheurs, étudiants ou employés.

Equipements nécessaires

L'idée est d'implémenter IPv6 au niveau d'un laboratoire, ce dernier sera équipé de quelques machines reliées entre elles via un Switch, une machine configurée en routeur une autre configurée comme étant un Home Agent.

IV. Les Perspectives

L'un des objectifs premiers de ce projet est d'étudier les services possibles que peut fournir le protocole Mobile IPv6. Nous allons donc aborder cet aspect au niveau utilisateur final.

L'avantage principal de Mobile IPv6 est qu'il permet à un utilisateur d'être joignable où qu'il soit, en conservant la même adresse. Cela permet également de maintenir les connexions en cours.

Le problème actuel est que le Mobile IPv6 est inhérent à la plupart des projets en développement, est que si la théorie semble être parfaite, les implémentations réelles le sont beaucoup moins. Comme nous avons pu le voir précédemment, nous avons rencontrés de nombreux problèmes de stabilité, ce qui rend l'implémentation actuelle totalement inutilisable. De plus, les environnements *.BSD ne sont pas particulièrement répandus sur les postes d'utilisateurs nomades; on les trouve bien plus souvent en cœur de réseau. Cela convient donc pour le Home Agent, mais pas pour le Mobile Node. Il serait donc prudent d'attendre une version plus mature au niveau du Home Agent, et de s'assurer de l'interopérabilité inter-piles.

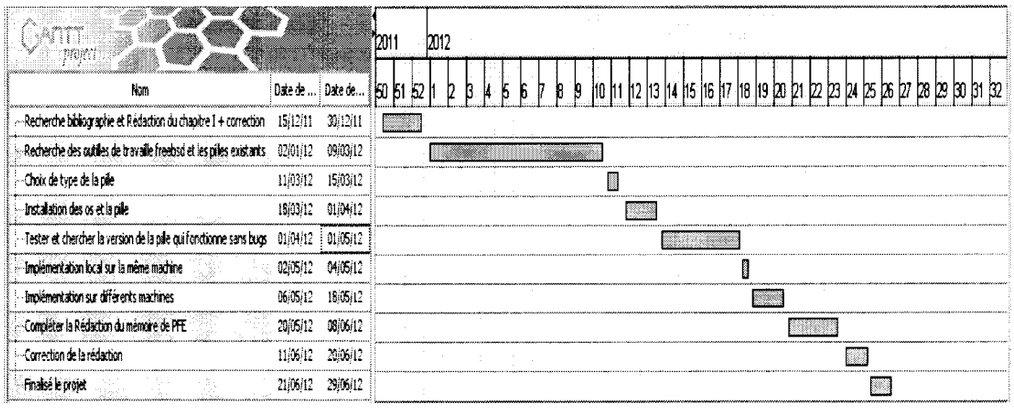
Nos perspectives, seraient d'installer ce protocole au niveau de notre réseau universitaire, pour faciliter la mobilité. Avec ce projet, on a ouvert les portes pour ceux qui sont intéressés pour continuer dans ce domaine, qui est très vaste. Nous avons donné L'essentiel pour une configuration IPv6 basique toutes les étapes nécessaires à l'installation et la configuration son incluse dans l'annexe et cela pour une éventuelle utilisation de notre mémoire.

Conclusion :

Une fois la préparation des outils matériels et logiciels sont terminés, il nous reste de tester le bon fonctionnement de la mobilité. Nous faisons un Ping depuis le CN vers le MN en utilisons la commande « ping6 » qui est dans le réseau mère. Apres, en déplace le MN vers un nouveau réseau et grâce à l'auto configuration le MN prendre une nouvelle adresse (adresse temporaire). On observe que le MN reste toujours connecter avec son CN au cours de déplacement depuis le réseau mère vers le réseau visité de manière transparente pour les utilisateurs.

Gestion de projet

I. Planning prévisionnel :



Nom	Date de ...	Date de...
Recherche bibliographie et Rédaction du chapitre I + correction	15/12/11	30/12/11
Recherche des outilles de travaille freebsd et les piles existants	02/01/12	09/03/12
Choix de type de la pile	11/03/12	15/03/12
Installation des os et la pile	16/03/12	01/04/12
Tester et chercher la version de la pile qui fonctionne sans bugs	01/04/12	01/05/12
Implémentation local sur la même machine	02/05/12	04/05/12
Implémentation sur différents machines	06/05/12	18/05/12
Compléter la Rédaction du mémoire de PFE	20/05/12	08/06/12
Correction de la rédaction	11/06/12	20/06/12
Finalisé le projet	21/06/12	29/06/12

Figure GP 1 : Diagramme de Gantt prévisionnel

N'ayant pas le matériel disponible les premières semaines, nous les avons passées à nous documenter sur la technologie et les retours d'expériences existants et une rédaction du chapitre I qui contient les détails sur le IPv4 et IPv6. La première date butoir fut le choix des outils logiciels à utiliser. Nous avons testé la difficulté d'utilisation de FreeBSD avant de proposer la solution KAME. De fait, nous l'avons installé en parallèle avec la sélection de la pile ce qui nous a beaucoup aidé dans les semaines suivantes.

Après une configuration complexe du HA, MN, CN, et le mobile routeur, pour réaliser une expérimentation sur un réseau local, nous avons préféré finalement réaliser une implémentation locale (sur la même machine en utilisant la machine virtuelle « VirtualBox » (Annexe E) qui permet d'ouvrir plusieurs systèmes sur une seule machine).

Conclusion Générale

Comme nous venons de le voir, les implémentations actuelles de Mobile IPv6 sont encore très Jeunes, et pas assez stables pour être pleinement exploitées.

L'installation d'une pile MIP6 aura été un vrai défi pour nous, contrairement à ce que nous avons pu penser au début de ce projet ; nous nous sommes rendus compte, au fur et à mesure de nos recherches, que le sujet était maîtrisé par une poignée de personnes à travers le monde, ce qui a posé des problèmes évidents de latence dans notre planning. Nous nous sommes donc efforcés de faire une synthèse de ce que nous avons compris, et de bien expliquer la manière dont nous avons fait l'installation, puisque c'est le point qui nous est apparu le plus ardu.

Ayant eu tous deux des sujets de stage ingénieur très différents (respectivement intégration réseau et conception logicielle), ce projet a été pour nous l'occasion d'avoir un aperçu du monde de la recherche dans le domaine des réseaux. Ce fût, nous pensons, une expérience très enrichissante, même si nous ne nous destinons pas (du moins pour l'instant) à une carrière de chercheur ; en effet, nos chemins se séparent à nouveau.

Néanmoins, nous serons désormais encore plus conscients des difficultés que peut poser l'implémentation d'un protocole apparemment sans faille.

Références Bibliographiques :

[1] http://reseau-linux.fr/conf/ip.php

[2] http://www.commentcamarche.net/contents/internet/ip.php3

[3] http://binaire-life.fr/net/cours-introduction-a-ladressage-ip

[4] http://www.01-depannage-informatique.com/tatw/upload/lofiversion/index.php?t385.html

[5] MohsenSouissi,IPv6 passeport pour l'Internet du Futur, les dossiers thématiques de l'AFNIC : www.afnic.fr/actu/presse/liens-utiles

[6] Z. MAMMERI, Introduction à IPv6, Cours de Réseaux, Université Paul Sabatier ((Toulouse III)

[7] http://www.linux-france.org/prj/edu/archinet/systeme/ch07s02.html

[8] Estelle COLIN, Fabrice BERNA, Le protocole Ipv6, Formation ,2001-2002

[9] SALMON Nicolas, COURS IPV6, 30/01/2010.

[10] Daniel O. Awduche,Les avantages du protocole IPv6 pour les entreprises,

[11] http://geekz.fr/Le-protocole-IPv6?artsuite=3

[12] Elodie Leocmach&&Jerome Meyer, Mobile ipv6:implémentation et test, rapport de projet de fin d'étude, INSA de Lyon, Juin 2005

[13] Elodie Leocmach&&Jerome Meyer, Mobile ipv6:implémentation et test, rapport de projet de fin d'étude, INSA de Lyon, Juin 2005

[14] http://www.kame.net/newsletter/20050707/

[15] Pierre-Emmanuel Goiffon, la mobilité IP , Rapport de stage de fin d'étude ,INSA de Lyon Département Télécommunication

[16] www.freebsd.org

[17] http://en.wikipedia.org/wiki/Shisa

[18] http://linux.softpedia.com/get/System/Networking/Zebra-14115.shtml

[19] http://fr.wikipedia.org/wiki/Oracle_VM_VirtualBox

Annexe A

Fichier rc.conf du Home Agent

```
#defaultrouter="139.124.132.250"

hostname="mip6ha.insa-lyon.fr"
keymap="fr.iso.acc"
linux_enable="YES"
sshd_enable="YES"
usbd_enable="YES"
inetd_enable="YES"

PATH=/usr/local/v6/sbin:/usr/local/v6/bin:${PATH}

moused_enable="YES"

#parametre ipv6
ipv6_enable="YES"
ipv6_network_interfaces="em0"
ipv6_defaultrouter="2001:660:5003:416::1"
ipv6_ifconfig_em0="2001:660:5003:416::3/64"
moused_enable="YES"

#parametre ipv6
ipv6_enable="YES"
ipv6_network_interfaces="em0"
ipv6_defaultrouter="2001:660:5003:416::1"
ipv6_ifconfig_em0="2001:660:5003:416::3/64"

#parametre mobile ipv6
ipv6_mobile_enable="YES"
ipv6_mobile_debug_enable="YES"
ipv6_mobile_security_enable="YES"
ipv6_mobile_config_dir="/usr/local/v6/etc/mobileip6"
ipv6_mobile_nodetype="home_agent"
ipv6_gateway_enable="YES"
ipv6_mobile_home_interface="em0"
ipv6_mobile_home_prefixes="2001:660:5003:416::/64"
```

Figure Annexe A : rc.conf du HA

Annexe B

Fichier rc.conf du Mobile Node

```
defaultrouter="139.124.132.250"

hostname="mip6mn.univ-mrs.fr"

keymap="fr.iso.acc"
linux_enable="YES"
inetd_enable="YES"
sshd_enable="YES"
usbd_enable="YES"

PATH=/usr/local/v6/sbin:/usr/local/v6/bin:${PATH}

ifconfig_em0="inet 139.124.132.13 netmask 255.255.255.0"
moused_enable="YES"

#parametre ipv6
ipv6_enable="YES"
ipv6_network_interfaces="em0 mip0"

#parametre mobile ipv6
ipv6_mobile_enable="YES"
ipv6_mobile_nodetype="mobile_node"
ipv6_mobile_security_enable="YES"
ipv6_mobile_config_dir="/usr/local/v6/etc/mobileip6"
ipv6_mobile_home_prefixes="2001:660:5003:416::/64"
ipv6_ifconfig_mip0="2001:660:5003:416:20f:1fff:fea2:2187 prefixlen 128 home"
```

Figure Annexe B : rc.conf du MN

Annexe C

Fichier config des Mobile Node et Home Agent

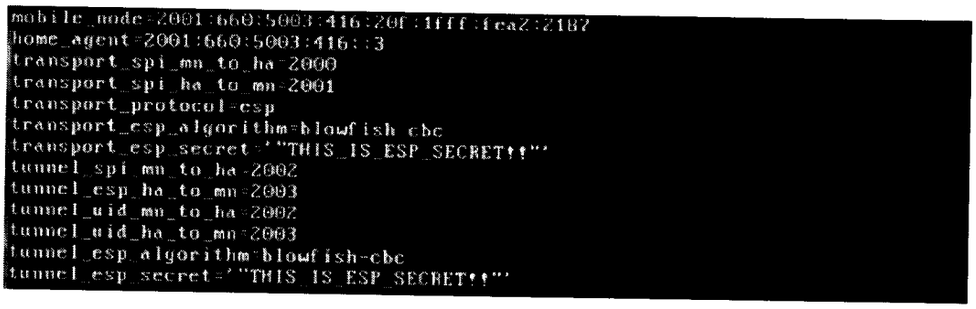


Figure Annexe C : fichier config

Annexe D

Fichier rc.conf du routeur

```
keymap="fr.iso.acc"
hostname="mip6router.insa-lyon.fr"
sshd_enable="YES"
usbd_enable="YES"

ipv6_enable="YES"
ipv6_gateway_enable="YES"
ipv6_router="/usr/sbin/route6d"
ipv6_router_flags=""

ipv6_network_interfaces="em0 em1"
ipv6_ifconfig_em0="2001:660:5003:416::2"
ipv6_ifconfig_em1="2001:660:5003:810::1"

ipv6_default_interface="em0"

rtadvd_enable="YES"
rtadvd_interfaces="em1"
```

Figure Annexe D: rc.conf du Router

Annexe E

Le projet KAME est la pile développée par le consortium japonais Wide. Ce consortium regroupe de nombreuses compagnie japonaise (Fujitsu,Hitachi, NEC , Toschiba,...) et l'objectif du projet KAME est de créer une implémentation de référence et gratuite d'IPv6 et d'IPsec.

Développée pour les systèmes d'exploitation BSD (FreeBSD, openBSD ,NetBSD), elle a été la première implémentation de la pile IPv6 à offrir des fonctionnalités avancées quant à la prise en charge du multicast IPv6. Le projet a commencé en 1998 et a été complété en Mars 2006 [15].

FreeBSD est un système d'exploitation avancé pour les plates-formes modernes de type serveur, station de travail et systèmes embarqués. Le code de base de FreeBSD a été développé, amélioré et optimisé continuellement pendant plus de trente ans. Il est développé et maintenu par une importante équipe de personnes. FreeBSD propose des fonctionnalités réseau avancées, une sécurité poussée et des performances de haut niveau. FreeBSD est utilisé par certains des sites web les plus visités ainsi que par la plupart des systèmes embarqués orientés réseau et des systèmes de stockage les plus répandus [16].

La pile Shisa MIPv6 est une implémentation du protocole MIPv6 et est inclus dans la pile KAME. Il a d'abord été créé en 2004. Au moment d'écrire ces lignes, le développement de la pile était encore en cours, notamment en ce qui concerne les fonctions de mobilité NEMO réseau.

La pile Shisa implémente MIPv6 comme spécifié dans [Joh04] et [Ark04], la mobilité du réseau ainsi que comme spécifié dans [Dev05].

Parce Mobile IP est par définition transparente à la couche de transport, aucun logiciel d'application n'a besoin d'interface avec la pile directement. Sur un nœud mobile, la pile Shisa offre un nouveau type d'interface réseau virtuelle. Le mip0 interface mobile peut être utilisé comme n'importe quelle interface réseau physique. Logiciel réseau qui nécessite un soutien de mobilité peut se lier datagramme ou sockets de flux à l'adresse du domicile affectée à l'interface mip0 [17].

[Dev05] www.feedface.com/howto/SHISA_Overview.pdf

Zébra est un protocole de routage multiserveur qui fournit le protocole TCP / IP basés sur les protocoles de routage. Il est destiné à être utilisé comme un serveur de routage et le réflecteur routeur. Non seulement une boîte à outils, il fournit toute la puissance de routage sous une nouvelle architecture. L'utilisateur peut changer dynamiquement la configuration et l'utilisation d'achèvement ligne de commande et de l'histoire de l'interface du terminal Il prend en charge le protocole BGP-4 tel que décrivent dans RFC1771 (A Border Gateway Protocol 4) ainsi que RIPv1, RIPv2 et OSPFv2. Contrairement aux traditionnels, les architectures monolithiques et même les soi-disant "nouvelles architectures modulaires" qui éliminent le fardeau de la transformation des fonctions de routage de la CPU et d'utiliser spéciaux puces ASIC lieu, Zébra logiciel offre une véritable modularité. Zébra est unique dans sa conception, car il a un processus pour chaque protocole [18].

Oracle VM VirtualBox (anciennement VirtualBox) est un logiciel de virtualisation créé par InnoTek. Il est disponible en tant qu'hôte sur les systèmes d'exploitation : Windows, Linux 32 et 64 bits, FreeBSD 32 et 64 bits et Mac OS X. Il supporte en tant qu'invité : Windows (dont Vista et 7), Linux 2.x, OS/2 Warp, OpenBSD et FreeBSD entre autres. Après plusieurs années de développement, VirtualBox a été publié sous la licence GNU GPL en janvier 2007.Le 12 février 2008, Sun Microsystems a annoncé un accord d'acquisition d'InnoTek. La version 2 de VirtualBox est sortie le 4 septembre 2008, elle intègre notamment des fonctionnalités supplémentaires dont le support des hôtes 64 bits, une interface Qt4 (Qt3 dans les versions précédentes) qui améliore l'intégration sous Gnome et l'utilisation de l'interface native sous Mac OS X^3 [19].

Annexe F

Installation

Nous allons détailler ici la procédure d'installation de la pile KAME sous FreeBSD 5.4. Le snap utilisé est celui en date du 25 juillet 2005.

1. Récupération du snap :

Le téléchargement du dernier snap peut se faire depuis l'adresse suivante :

http://ftp.naist.jp/pub/kame/snap/old/kame-20050725-freebsd54-snap.tgz

On peut alors décompresser et désarchiver le fichier.

```
% gzip -d kame-200500727-freebsd54-snap.tgz
% tar -xvf kame-20050727-freebsd54-snap.tar
```

Dans la suite de la procédure, on supposera que le répertoire ainsi créé se trouve au chemin

/usr/src/kame.

2. Préparation du nouveau noyau :

```
% cd /usr/src/kame
% make TARGET=freebsd5 prepare
% cd /usr/src/kame/freebsd5/sys/i386/conf/
% cp GENERIC.KAME MIP6K
% vi MIP6K
```

Pour le Home Agent, il faut alors décommenter options MIP6.

Pour le Mobile Node, il faudra décommenter options MIP6 et devicemip 1.

```
% /usr/sbin/config MIP6K
% cd ../compile/MIP6K/
% make depend
% make && make install
% fastboot
```

3. Installation des programmes userland :

```
% cd /usr/src/kame/freebsd5/
% make includes
% make install-includes
% make && make install
```

4. Installation de Shisa:

% cd /usr/src/kame/freebsd5/usr.sbin/shisad/

% make clean

% make&&makeinstall

On copie les exécutables dans les répertoires par défaut, afin d'être sûrs d'utiliser les bonnes versions.

% cp /usr/local/v6/bin/* /sbin

% cp /usr/local/v6/sbin/* /sbin

6. Modification de la configuration de démarrage :

Il faudra modifier les fichiers /etc/rc.conf du HA (cf. Annexe A, page 63) et du MN (cf.Annexe B, page 64).

7. Mise en place des scripts de démarrage :

Il faut copier deux scripts de démarrage dans /etc/rc.d, pour automatiser le lancement des démons relatifs à MIP6.

% cp /usr/src/kame/freebsd/etc/rc.d/network_ipv6_mobile /etc/rc.d/

Dans d'autre version de freebsd 5, il faut copier :

% cp /usr/src/kame/kame/kame/shisad/network_ipv6 /etc/rc.d/

% cp /usr/src/kame/kame/kame/shisad/network_ipv6_mobile /etc/rc.d/

Il ne reste plus ensuite qu'à redémarrer la machine pour avoir une configuration Mobile IPv6 en place.

% fastboot

8. Mise en place de la sécurité :

Il est impératif de mettre en place une sécurisation des échanges entre le MN et le HA, pour la signalisation. Kame/MIP6 utilise IPSec pour protéger les Binding Updates et Binding Acknowledgements.

Etant donné la relative complexité des paramètres, il y a quelques utilitaires qui nous aident dans notre tâche. En l'occurrence, on trouve mip6makeconfig.sh et mip6seccontrol.sh dans le répertoire /usr/usr/kame/kame/kame/shisad/.

Tout d'abord, nous avons créé un répertoire de configuration qui garde les fichiers de configuration MIP6 relatifs à IPSec.

% mkdir /usr/local/v6/etc/mobileip6

Ensuite, nous avons créé un répertoire pour chaque Mobile Node, le nom du répertoire étant arbitraire.

% mkdir /usr/local/v6/etc/mobileip6/mobile_node_0

Après, nous avons créé un fichier appelé config dans chaque répertoire (cf. Annexe C, page 65).

Finalement, il a fallu créer les fichiers d'initialisation des paramètres en utilisant l'utilitaire :

mip6makeconfig.sh.

% /usr/src/kame/kame/kame/shisad/mip6makeconfig.sh mobile_node_0

Où mobile_node_0 est le nom du répertoire que l'on a créé précédemment. Après l'exécution réussie de ce programme, six fichiers sont créés par dossier de nœud. Chaque fichier contient des paramètres IPSec. mip6seccontrol.sh est un programme qui actionne les paramètres IPsec.

Pour le Mobile Node:

% /usr/src/kame/kame/kame/shisad/mip6seccontrol.sh -m installall

Pour le Home Agent:

% /usr/src/kame/kame/kame/shisad/mip6seccontrol.sh -g installall

Enfin, il convient d'activer la sécurité dans le fichier rc.conf :

ipv6_mobile_security_enable="YES"

Un redémarrage est nécessaire pour prendre en compte ces changements.

% fastboot

Résumé

La nouvelle génération de Protocole Internet, IPng (next generation), ou IPv6 va offrir de nouvelles capacités d'adressage, des options de sécurité, et bien d'autres fonctionnalités comme la mobilité qui est l'un des points majeurs sur lequel IPv6 a été conçu.

Dans ce projet nous montrerons la mobilité et ceci après avoir implémenté une pile protocolaire appelée KAME et configurer sous une plateforme FreeBSD, et tester les communications entre le nœud mobile et son correspondant et ceci même quand le nœud mobile est en déplacement d'un réseau vers un autre.

Mots-Clès :
Mobile IPv6, IPv6, MIP6, KAME, FreeBSD.

Summary

The new generation of Internet Protocol IPng (next generation), will offer new IPv6 addressing capabilities, security options, and many other features such as mobility, which is one of the major points on which IPv6 been designed.

In this project we will show mobility from implementing a protocol stack called KAME stack and configure it over a FreeBSD platform, and test communications between correspondent node and mobile node when mobile node is reaching from a network to another.

Keyword:
Mobile IPv6, IPv6, MIP6, KAME, FreeBSD.

الملخص

يتميز الجيل الجديد لبروتوكول الإنترنت او ما يسمى IPv6 الجديد بقدرات التصدي، وخيارات الأمن، والعديد من الميزات الأخرى مثل التنقل، والتي هي واحدة من النقاط الرئيسية التي تم تصميمها لإصدار IPv6 .

في هدا المشروع سوف نطبق خاصية التنقل ودلك بالاعتماد على مشروع KAME الدى سوف ننصبه على نظام تشغيل FreeBSD و اختبار الاتصال في حالة تنقل المحمول.

الكلمة الرئيسية:

المحمول IPv6، IPv6 ، MIP6، KAME، FreeBSD .