

Rapport de Stage de fin d'études

Aristote / Renater

Renater



Remerciements

Je tiens à remercier mes tuteurs de stages : Bernard Tuy (GIP Renater, unité réseau du CNRS CNRS, président du G6) qui a su m'apporter son expérience pour me guider et me conseiller tout au long du projet, mais tout en me laissant assez indépendant sur la manière de conduire le projet. Jacques Prevost (GIP Renater, CEA, Trésorier de l'Association Aristote) qui a toujours su me donner les moyens nécessaires à la mise en œuvre de certaines manipulations, et qui m'a donc vraiment simplifié la tâche. Enfin, Farid Naït, maître de conférence à l'INSA de Lyon qui a su s'impliquer dans le stage en venant me rendre visite à deux reprises sur le site du GIP Renater.

Je remercie Jérôme Durand, précédent stagiaire de l'INSA, pour avoir documenté son travail. Cette documentation détaillée, m'a énormément aidé, surtout au début lorsqu'il a fallu que je reprenne en route le projet qu'il avait commencé 6 mois auparavant. J'ai pu ainsi démarrer mon stage dans des conditions optimales.

Je remercie Lionel David, autre stagiaire du DESS ART de l'Université Paris VII, avec qui j'ai effectué mes premières configurations IPv6 et multicast sur FreeBSD.

Je remercie également toutes les personnes qui ont collaboré au déploiement du M6Bone en particulier Konstantin Kabassanov du LIP6 et Luc Beurton de l'Université de Bretagne Sud, ainsi que tous les sites qui se sont connectés au M6Bone.

Je remercie Philippe Bourcier et Kostya Kortchinsky du GIP Renater pour leur aide précieuse au niveau de l'utilisation des systèmes BSD ainsi que sur les questions de sécurité dans le M6Bone.

Je remercie aussi les personnes du NOC IPv6 avec qui j'ai eu à travailler, en particulier Mathias Lefaucheur.

Enfin, je remercie l'ensemble de l'équipe du GIP Renater, qui m'a permis d'effectuer mon stage dans un cadre particulièrement agréable.

Sommaire

1.	INTRODUCTION	2
2.	LE CONTEXTE	2
2.1.	ARISTOTE	2
2.2.	RENATER	2
2.3.	LE GIP RENATER.....	4
2.4.	LE G6	4
2.5.	CADRE DU STAGE ET OBJECTIFS	5
3.	LES TECHNOLOGIES UTILISES.....	5
3.1.	IPV6.....	5
3.1.1.	<i>Adresses</i>	<i>6</i>
3.1.2.	<i>Autoconfiguration des interfaces connectées</i>	<i>7</i>
3.1.3.	<i>Format d'un datagramme IPv6.....</i>	<i>8</i>
3.2.	MULTICAST	8
3.2.1.	<i>Introduction.....</i>	<i>8</i>
3.2.2.	<i>Bibliographie (RFCs, drafts).....</i>	<i>9</i>
3.2.3.	<i>Quelques implantations actuelles.....</i>	<i>19</i>
3.3.	DSTM (DUAL STACK TRANSITION MECHANISM)	19
3.3.1.	<i>Présentation</i>	<i>19</i>
3.3.2.	<i>Fonctionnement.....</i>	<i>20</i>
3.3.3.	<i>Implantations actuelles</i>	<i>22</i>
4.	DEPLOIEMENT DU M6BONE	22
4.1.	OBJECTIFS.....	22
4.2.	CONTEXTE	22
4.2.1.	<i>Une technologie jeune.....</i>	<i>22</i>
4.2.2.	<i>Premiers déploiements au sein de Renater</i>	<i>23</i>
4.3.	LES ENJEUX	23
4.4.	REALISATION	23
4.4.1.	<i>Equipements utilisés.....</i>	<i>23</i>
4.4.2.	<i>Architecture globale du réseau</i>	<i>24</i>
4.4.3.	<i>La diffusion du séminaire X/Aristote du 6 juin 2002.....</i>	<i>30</i>
4.5.	DOCUMENTATION	34
5.	MISE EN PLACE D'UNE PLATE-FORME DSTM	35
5.1.	INTRODUCTION	35
5.2.	OBJECTIF	35
5.3.	REALISATION DE LA MAQUETTE.....	36
6.	CONCLUSION	37
	BIBLIOGRAPHIE.....	38

1. Introduction

La scolarité dans le département Télécommunications, Services et Usages comprend une période de stage en entreprise de 6 mois. Ce stage doit permettre d'appliquer les connaissances pratiques et théoriques acquises à l'INSA dans le cadre d'une entreprise. Il doit aussi permettre d'avoir une première expérience avec le travail d'ingénieur.

C'est donc dans ce cadre que j'ai effectué une période de 6 mois au sein du GIP Renater à Paris. Voici donc le rapport de stage expliquant ce que j'ai fait durant ce projet.

2. Le contexte

2.1. Aristote

Aristote est une association créée en 1988 régie par la loi de 1901, et qui regroupe de grands organismes ou entreprises français intéressés en tant qu'acteurs ou utilisateurs, à l'évolution des télécommunications.

L'objectif d'Aristote se situe dans le domaine des techniques, moyens, outils et services de communication informatique, notamment :

- Mettre en commun des efforts de prospective, d'étude et d'information faits par ses membres.
- Promouvoir l'élaboration et la mise en service de nouveaux produits, systèmes et services d'intérêt général au bénéfice de ses partenaires.
- Organiser ou encourager des actions avancées d'information ou de formation : séminaires d'intérêt général, séminaires de formation technique, journées d'étude thématiques.

Ces actions sont mises en œuvre par trois types d'activités :

- Les groupes de travail thématiques : ils rassemblent des personnes des organismes membres d'Aristote ainsi que des experts invités, et permettent de travailler sur des domaines comme le calcul scientifique distribué, ou encore les Interfaces Homme-Machine.
- Les actions pilotes : elles ont pour objectif d'effectuer des expériences d'évaluation de nouvelles technologies, d'évaluer de nouveaux produits en milieu d'utilisateurs évolués, voire de réaliser des validations préopérationnelles de nouveaux services.
- La transmission du savoir par le biais de l'organisation régulière de conférences.

L'association Aristote est un acteur important dans le développement et dans la diffusion de connaissances concernant les nouvelles technologies de l'information et de la communication.

2.2. Renater

Renater (Réseau National de Télécommunications pour l'Enseignement et la Recherche) a été créé dans les années 1990 dans le but de fédérer les infrastructures de télécommunications pour la Recherche et l'Enseignement.

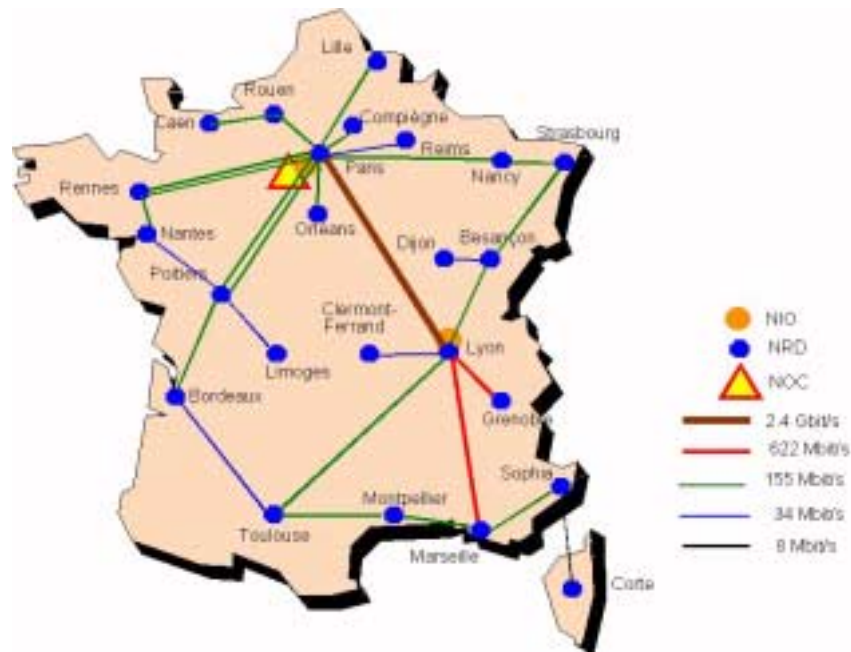
Il relie aujourd'hui plus de 600 sites qui ont leur activité dans le domaine de la recherche, la technologie, l'enseignement et de la culture, et leur permet de communiquer entre eux, et avec leurs homologues de l'étranger. Il permet aussi à tous les sites connectés d'accéder à l'Internet.

Le réseau est composé d'une épine dorsale constituée de 26 points de présence régionaux interconnectés à haut débit. Cette infrastructure fédère les réseaux régionaux, déployés avec le

soutien des collectivités territoriales dans le cadre de leur politique d'aménagement du territoire.

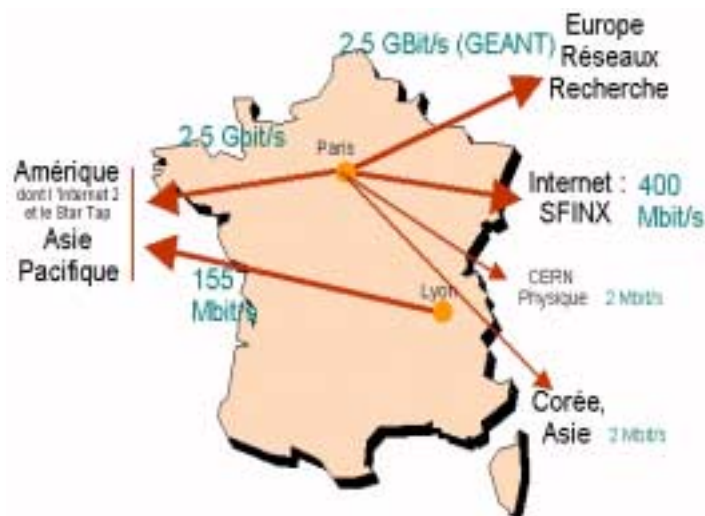
Le réseau Renater propose une couverture nationale et assure une continuité du service jusque dans les départements et territoires d'Outre-Mer. Le réseau RENATER est interconnecté aux grands réseaux de recherche européens via le réseau pan-européen GÉANT et au continent nord-américain via deux liaisons dont une est exclusivement dédiée aux projets de recherche avec les réseaux des grands organismes scientifiques nord-américains (vBNS, Internet 2, NASA, CANARIE) via le STAR TAP.

Épine dorsale du réseau :



Les débits proposés par l'épine dorsale de Renater 2 bis vont, suivant les liaisons, de 8 Mbit/s jusqu'à 2.4 Gbit/s sur la liaison Paris-Lyon.

L'épine dorsale est connectée au reste du monde par différentes liaisons :



Les services proposés par Renater :

- Un service IP classique avec une connectivité nationale et comme nous venons de le voir, internationale.
- Un service de réseau privé virtuel à bande passante réservée entre sites d'un même organisme ou pour réaliser un réseau thématique.
- Un service de diffusion IP multicast.
- Un service pilote IPv6. C'est un service qui est natif, grâce à des circuits ATM dédiés au transport des paquets v6.
- Le SFINX (Service for French Internet eXchange). C'est un point d'échange de trafic entre prestataires de service Internet ou opérateurs de télécommunications qui veulent échanger du trafic, sans transit et sans passer par des infrastructures internationales. Ce Point d'interconnexion est réparti sur trois sites à Paris reliés entre eux par du Gigabit Ethernet.

2.3. Le GIP Renater

Le réseau Renater est géré par le GIP Renater (Groupement d'Intérêt Public Renater). Celui-ci réunit de grands organismes de recherche et d'enseignement (CEA, CIRAD, CNES, CNRS, INRA, INRIA), ainsi que le Ministère en charge de l'Education Nationale, de la Recherche et de la Technologie.

Un GIP (Groupement d'Intérêt Public) est un organisme à but non lucratif, réunissant des administrations de l'Etat et des organismes publics pour une activité définie : dans le cas du GIP Renater il s'agit de la gestion du réseau Renater.

Le GIP Renater est le maître d'ouvrage de la partie commune de Renater, constituée de son épine dorsale Renater 2, des liaisons internationales, de ses actions pilotes, et du service SFINX. Il est le coordinateur technique et opérationnel global de l'ensemble du réseau Renater. Enfin, il est chargé de la supervision du réseau.

Le GIP Renater est financé par un budget alloué par le gouvernement, par l'Europe pour certains projets spécifiques, ainsi que par les contributions des sites connectés au réseau.

Le directeur du GIP Renater est M. Dany Vandromme, Professeur des Universités. L'équipe du GIP Renater comprend aujourd'hui un peu moins de 30 de personnes : ingénieurs, techniciens et personnel administratif.

2.4. Le G6

Le G6, Groupe d'expérimentation IPv6, est un groupe créé en novembre 1995 sous l'impulsion de Bernard Tuy (UREC) et d'Alain Durand (IMAG). Il cherche à regrouper les expérimentateurs de IPv6 en France pour les aider à partager leurs expériences et commencer à coordonner des actions communes.

En janvier 2000, le groupe devient une association loi 1901 qui a pour objectif de tester, de développer et de promouvoir IPv6. Elle est alors composée par des personnes issues du monde de la recherche, ou du monde de l'industrie (opérateurs de télécommunications, constructeurs, ou autres entreprises s'intéressant à la technologie).

Les actions du G6 sont diverses :

- Rédaction d'un Livre, *IPv6, Théorie et Pratique*, présentant de manière complète le protocole et ses implantations.
- Participation dans des projets permettant de mettre en œuvre, et de tester le protocole, et les services associés.
- Coordination des différentes actions concernant IPv6 en France.

- Retour d'expérience entre les différents membres du groupe.

La présidence du G6 est assurée par Bernard Tuy, entre autre responsable du déploiement IPv6 au sein de Renater.

2.5. Cadre du stage et objectifs

Mon stage s'est donc déroulé dans le cadre des deux organismes cités plus hauts. C'est en effet un partenariat entre l'association Aristote, et le GIP Renater. Administrativement, j'ai effectué un stage pour Aristote, mais j'ai effectué ce stage dans les locaux du GIP Renater, à Paris. J'avais pour ce stage deux tuteurs : Bernard Tuy, responsable des technologies IPv6 et multicast au sein du GIP Renater, et, au titre d'Aristote, Jacques Prévost (GIP Renater) chargé des applications avancées (visioconférences...)

Le sujet du stage proposé était : « Développement et mise en oeuvre expérimentale des outils et des procédures de vidéoconférence sur un réseau IPv6 multicast, ainsi que mises en oeuvre complémentaires d'autres outils IPv6 ».

La première partie sur le multicast IPv6 s'inscrit dans la continuité de ce qui a été fait par Jérôme Durand, étudiant du département Télécommunications de l'INSA de Lyon, qui a effectué son stage de fin d'étude dans même cadre que moi, et qui a mis en place un service de multicast IPv6 sur le Pilote IPv6 de Renater. Le point final de cette première partie est la diffusion en multicast IPv4 et IPv6 de la Conférence X-Aristote du 6 juin 2002. C'est une conférence d'une journée organisée par Aristote, et habituellement retransmise en multicast IPv4.

La deuxième partie du stage concerne l'étude et la mise en place d'une maquette de test d'un mécanisme de transition IPv4-IPv6, le DSTM (*Dual Stack Transition Mechanism*). L'objectif est de tester quelques implantations disponibles sur le réseau local du GIP Renater, afin de pouvoir proposer par la suite une architecture de déploiement sur l'ensemble du pilote IPv6 de Renater.

3. Les technologies utilisés

3.1. IPv6

Le réseau Internet actuel, basé sur le protocole IPv4 a été initialement prévu pour relier au maximum une centaine de machines. Hors on connaît tous le succès actuel d'Internet, et on a vu le nombre d'utilisateurs augmenter de manière exponentielle. Les plus pessimistes avaient prévu la saturation du réseau IPv4 dès 1994. Cette prévision ne s'est pas réalisée, car des mesures d'urgence ont été prises, comme par exemple le CIDR (Classless Interdomain Routing, RFC 1519), l'adressage privé (RFC 1918) ou encore l'utilisation du NAT (translation d'adresses). Cependant, la situation n'en est pas moins préoccupante, et la pénurie d'adresse IPv4 se fait sentir, particulièrement en Asie et en Afrique.

Une autre problème du protocole IPv4 concerne le routage : les nœuds du réseau IPv4 ont vu leur table de routage exploser ces dernières années. Cette augmentation des tables dues à une mauvaise hiérarchisation des adresses, rend la gestion du réseau IPv4 de plus en plus complexe pour les routeurs, et il était urgent de résoudre ce problème en agrégeant les adresses IP au maximum.

Pour ces deux raisons, l'IETF a travaillé sur une nouvelle version du protocole IP, permettant de pallier aux limites d'IPv4. Cette nouvelle version est la version 6 du protocole, la version 5 ayant été utilisée pour un protocole expérimental.

IPv6 est conçu pour s'affranchir des limitations d'IPv4, mais aussi pour prendre en compte les avancées issues des recherches sur les réseaux, comme l'autoconfiguration, la mobilité, le multicast ou encore la sécurité.

3.1.1. Adresses

Le nombre d'adresses disponibles était la principale limitation d'IPv4. Par conséquent, la taille d'une adresse IPv6 a été multipliée par 4, soit 128 bits, contre 32 bits pour IPv4.

Notation

La notation d'une adresse IPv6 se fait en découpant le mot de 128 bits en 8 mots de 16 bits séparés par le caractère « : », chacun d'eux étant représenté en hexadécimal.

Par exemple : fedc:0000:0000:400:a987:6543:210f

Cependant, il n'est pas nécessaire d'écrire les 0 en tête de chaque champ, et l'adresse devient alors fedc:0:0:0:400:a987:6543:210f

Enfin, plusieurs champs nuls consécutifs peuvent être abrégés par « :: », et l'adresse donnée en exemple devient alors fedc::400:a987:6543:210f. Bien entendu, pour éviter toute ambiguïté, il n'est possible d'utiliser cette abréviation qu'une seule fois dans une même adresse.

Différents types d'adresses

Le protocole IPv6 définit trois types d'adresses, les adresses unicast, multicast et anycast.

Une adresse unicast désigne une interface unique d'un équipement, et lorsqu'un paquet a comme adresse de destination une adresse de ce type, il sera acheminé vers cette interface ainsi identifiée.

Une adresse multicast désigne un groupe d'interfaces qui en général appartiennent à des nœuds différents pouvant être situés n'importe où dans l'Internet. Un paquet ayant comme adresse de destination une adresse multicast sera ainsi acheminé à l'ensemble des interfaces membres du groupe.

Le dernier type d'adresses est le type anycast. Les adresses anycast désignent, comme les adresses multicast, un groupe d'interface, la différence étant que lorsqu'un paquet a comme adresse de destination une telle adresse, il est acheminé à un des éléments du groupe et pas à tous.

Portée des adresses

Il est possible de définir trois types d'adresses unicast, suivant leur portée dans le réseau :

- Les adresses globales : ces adresses sont valides sur l'ensemble de l'Internet. Le préfixe utilisé actuellement pour ces adresses est 2000::/3.
- Les adresses site-local : la validité de ces adresses est restreinte à un site. Cela permet par exemple à un site qui n'est pas encore connecté à Internet de pouvoir faire des tests localement sans faire de demande de préfixe. Les adresses site-local utilisent le préfixe fec0::/48.
- Les adresses lien-local : Ce sont des adresses dont la validité est restreinte à un lien, c'est à dire à l'ensemble des interfaces directement connectées sans routeur intermédiaire. Ces adresses sont configurées automatiquement lors de l'initialisation de l'interface et permettent la communication entre nœuds voisins. Les adresses lien-local utilisent le préfixe fe80::/64.

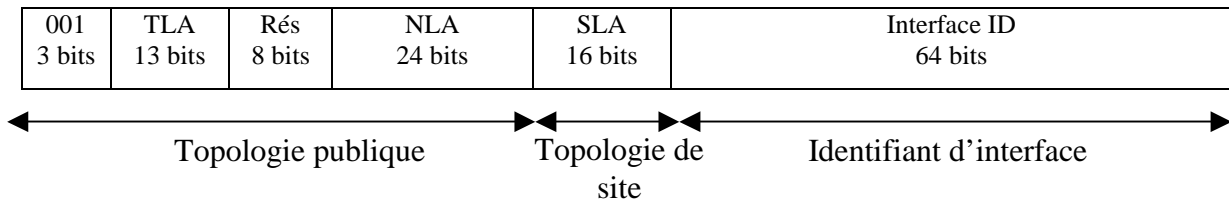
Plan d'adressage unicast

Un des problèmes majeurs d'IPv4 est la croissance incontrôlée des tables de routage. Ce problème est dû à une mauvaise agrégation des adresses dans les tables.

Lors de la mise au point d'IPv6, le plan d'adressage a été conçu de manière hiérarchisée de façon à pouvoir agréger les adresses, et à réduire au maximum la taille des tables de routage.

Le plan d'adressage retenu pour IPv6 et défini dans le RFC 2450 est donc un plan agrégé. Il comprend trois niveaux de hiérarchie :

- Une topologie publique (48 bits)
- Une topologie de site (16 bits)
- Un identifiant d'interface (64 bits)



Format de la topologie publique :

- Un préfixe 2000::- Le TLA (*Top Level Agregator* ou unité d'agrégation haute) Les TLA représentent les grands opérateurs internationaux
- Une partie réservée sur 8 bits dont l'adresse permet de faire évoluer le plan d'adressage.
- Une unité d'agrégation basse (NLA : *Next Level Agregator*) de 24 bits. Il constitue l'identificateur du site ou du domaine. Le découpage des NLA est réalisé par l'unité d'agrégation haute (TLA)

Format de la topologie de site :

La topologie de site correspondant au SLA (*Site Level Agregator*) est codé sur 16 bits, et est sous la responsabilité du gestionnaire du site.

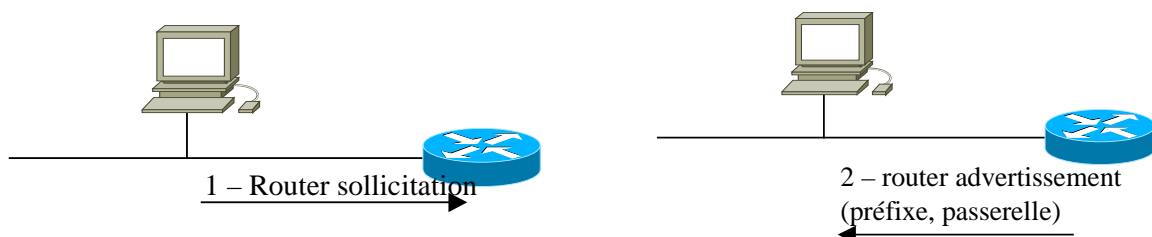
Enfin, l'adresse se termine par l'identifiant d'interface codé sur 64 bits.

3.1.2. Autoconfiguration des interfaces connectées

Une autre fonctionnalité d'IPv6 est de permettre l'autoconfiguration de n'importe quel interface connectée au réseau. L'objectif est que la connexion à un réseau IPv6 soit « *plug and play* ».

Par conséquent, lorsqu'une station se connecte à un réseau, elle doit pouvoir connaître, sans que l'utilisateur ou l'administrateur n'ait à renseigner aucune information, son adresse IP ainsi que la passerelle par défaut.

Lorsqu'une station se connecte au réseau, elle émet un paquet « *router solicitation* » sur le LAN, qui permet de faire savoir au routeur du LAN qu'elle a besoin de calculer son adresse. Le routeur du LAN répond par un paquet « *router advertisement* » qui contient les informations nécessaires à la configuration de la station, comme le préfixe du LAN ou encore la passerelle par défaut. Connaissant le préfixe du LAN, elle peut construire son adresse IPv6 à partir de son adresse MAC. Un mécanisme permet de vérifier ensuite l'unicité de l'adresse ainsi configurée.



3.1.3. *Format d'un datagramme IPv6*

Version 4 bits	Priorité 4 bits	Etiquette de flot 24 bits	
Longueur de la charge utile 16 bits		En-tête suivant 8 bits	Nombre maximal de sauts 8 bits
Adresse source 16 octets 128 bits			
Adresse destination 16 octets 128 bits			

Champ version : indique la version du protocole utilisé, IPv4 ou IPv6. Il sert à vérifier que le paquet est bien traité par la bonne couche réseau.

Champ priorité : ce champ permet d'indiquer aux routeurs la priorité relative des différents datagrammes transmis. Il peut prendre les valeurs allant de 0 à 15.

Champ étiquette de flot : permet de définir différents flots, que les nœuds du réseau peuvent alors traiter avec un comportement particulier (en terme de gestion des files d'attente par exemple).

Champ longueur de charge utile : ce champ indique le nombre d'octets d'information qui suivent les en-têtes de base et d'extension. Ce champ contient 16 bits, ce qui permet d'avoir jusqu'à 64 Ko de données utiles par datagramme.

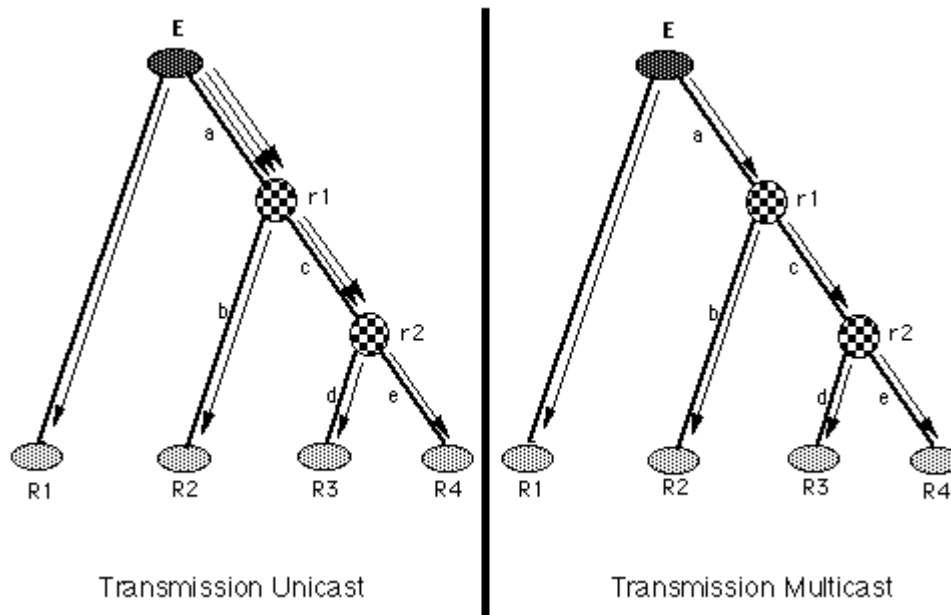
Champ en-tête suivant : ce champ indique le type du prochain en-tête d'extension.

Champ nombre maximum de sauts : ce champ a comme fonction d'éviter qu'un datagramme ne circule indéfiniment dans un réseau. La valeur contenue dans ce champ représente le nombre de routeurs que le datagramme peut traverser avant d'être détruit. Chaque fois que le datagramme traverse un routeur, la valeur de ce champ est décrétementée d'une unité. Lorsque la valeur atteint 0, le datagramme est détruit.

3.2. Multicast

3.2.1. *Introduction*

Le multicast, contrairement à l'unicast, permet de transmettre de l'information d'une source (ou éventuellement plusieurs sources) à plusieurs destinataires. Le multicast permet une optimisation de l'utilisation des ressources réseau, et notamment du backbone. En effet, l'information ne circule qu'une fois sur le backbone, alors qu'une transmission unicast aurait fait transiter la même information autant de fois qu'il y a de destinataires.



3.2.2. Bibliographie (RFCs, drafts)

Nous allons dans cette partie décrire quels sont les textes importants pour le multicast, dans le contexte particulier d'IPv6. Le multicast n'est en effet pas propre à l'IPv6, et de nombreux documents restent spécifiques au protocole IPv4. Les principaux documents auxquels je me suis intéressé sont les documents fournis par l'IETF (*Internet Engineering Task Force*), organisme chargé de la normalisation de tous les protocoles ayant rapport avec Internet. Les deux principaux types de documents publiés par l'IETF sont :

- Les *Internet drafts* (brouillons) : ce sont des textes proposés par des membres d'un groupe de travail, qui définissent une proposition, et qui doivent être discutés au sein du groupe de travail, afin d'aboutir à une possible standardisation. Ces drafts ont une durée de vie limitée à 6 mois.
- Les RFC (*Request for Comments*). Ces documents décrivent des standards qui ont été discutés entre les différents membres des groupes de travail de l'IETF. Ils sont la suite logique des Internet drafts, une fois que ceux-ci ont fait l'unanimité au sein des groupes de travail, et qu'ils ont donné lieu à une implantation au moins.

Les *Internet drafts* sont donc le reflet du travail actuel d'un groupe de travail. Ils peuvent exister en plusieurs versions, mises à jour au fur et à mesure de l'avancement des discussions au sein du groupe de travail. Les RFCs, une fois publiés sont statiques et ne sont pas destinés à être modifiés : ils peuvent simplement être rendus obsolètes par un autre RFC.

Définition

Le RFC qui a défini le concept du multicast est le RFC 1112 (*Host Extensions for IP Multicasting*, Août 1989). Ce RFC décrit ce qu'est le multicast, la transmission de datagrammes IP vers un groupe d'utilisateurs. La description du multicast dans ce RFC se fait par rapport au protocole IPv4, mais le fonctionnement général du multicast est indépendant de la version du protocole IP. Le RFC définit la notion de groupe de diffusion.

Une extension de cette définition du multicast est apparue plus récemment, et prend le nom de SSM, comme *Source Specific Multicast*. Le fonctionnement du SSM est défini dans le draft draft-ietf-ssm-arch-00.txt. Ce document explique que le SSM permet à chaque client de connaître et surtout de choisir de n'écouter qu'une source particulière. Des explications sur le déploiement du SSM sont fournies dans le draft draft-ietf-ssm-overview-03.txt

Le RFC 1458 (*Requirements for Multicast Protocols*, mai 1993), définit quant à lui les caractéristiques que doit avoir un protocole de transport multicast, surtout en ce qui concerne les adresses de groupe et les appartenances à ces groupes de diffusion.

L'anycast, qui est un cas particulier du multicast et qui fait partie intégrante de toute pile IPv6, est lui défini dans le RFC 1546 (*Host Anycasting Service*, novembre 1993).

Les adresses multicast IPv6

La définition du protocole IPv6 prévoit l'assignation d'une plage d'adresse réservée au multicast. La répartition des adresses IPv6 en fonction de leurs utilisations est décrite dans le RFC 2373 (*IP Version 6 Addressing Architecture*, juillet 1998). Les adresses multicast se voient assigner le préfixe ff00::/8. Le RFC décrit ensuite le format d'une adresse multicast IPv6 qui a la forme suivante :

8 bits	4 bits	4 bits	112 bits
1111	1111	flags	scope
F	F	group ID	

On retrouve alors :

- Le préfixe sur 8 bits : 11111111 ou ff en hexadécimal.
- Le champ flag de la forme 000T avec le bit T=0 si l'adresse multicast est assignée de manière permanente, et avec T=1 si l'adresse multicast est temporaire.
- Le champ scope, de 4 bits, permet de limiter la portée de la diffusion vers un groupe de diffusion sur un réseau. Voici les valeurs de ces champs qui sont définies dans le RFC :
 - 0 Réserve
 - 1 Portée nœud local
 - 2 Portée lien local
 - 5 Portée site local
 - 8 Portée organisation locale
 - E Portée globale
 - F Réserve
- Le champ Group ID est l'identificateur proprement dit du groupe de diffusion multicast.

Le RFC 2375 (*IPv6 Multicast Address Assignments*, juillet 1998) traite lui aussi de l'attribution des adresses IPv6, mais cette fois dans le cadre particulier des adresses multicast prédéfinies. Il liste l'ensemble des adresses multicast prédéfinies.

Le RFC 2776 (*Multicast-Scope Zone Announcement Protocol (MZAP)*, février 2000), définit le protocole MZAP, qui permet de découvrir les différentes zones multicast (définies par le champ scope) qui existent sur une zone géographique donnée. Ce protocole fournit aussi un mécanisme permettant de détecter les erreurs les plus courantes concernant la configuration du scope.

L'allocation des adresses de groupe multicast

Une adresse de groupe peut être fixe et avoir une durée de vie infinie (cas typique des adresses définies dans le RFC 2375), ou alors elle peut être allouée temporairement et dynamiquement (c'est ce qui se passe lors de la création d'une session audio-vidéo multicast). Il est alors nécessaire d'imaginer un mécanisme permettant d'allouer dynamiquement et pour une durée temporaire une adresse multicast à un groupe de diffusion.

Un tel mécanisme est décrit dans le RFC 2730 (*Multicast Address Dynamic Client Allocation Protocol (MADCAP)*, décembre 1999). Il définit un protocole, MADCAP, qui permet à un

client d'effectuer une demande d'adresse multicast auprès d'un serveur d'allocation d'adresse multicast. Ce protocole est une évolution de DHCP pour le multicast. La définition du protocole est complétée par le RFC 2907 (*MADCAP Multicast Scope Nesting State Option*, septembre 2000), qui définit de nouvelles options de fonctionnement.

Le RFC 2908 (*The Internet Multicast Address Allocation Architecture*, Septembre 2000) décrit aussi un mécanisme de la sorte. Ce RFC décrit tout d'abord quelles doivent être les caractéristiques d'un mécanisme d'allocation d'adresses multicast. Un tel mécanisme doit être à la fois robuste, disponible, instantané, avoir une probabilité de collision très faible tout en prenant en compte une possible restriction au niveau du nombre d'adresses disponibles.

Après cette présentation générale, les auteurs entrent un peu plus dans les détails de fonctionnement de ce système appelé MALLOC.

Un autre protocole permettant l'allocation d'adresses multicast est défini dans le RFC 2909 (*The Multicast Address-Set Claim (MASC) Protocol*, décembre 2000). Le RFC décrit le protocole MASC qui peut être utilisé pour l'allocation inter-domaine d'adresses multicast. Ce protocole est utilisé par un nœud du réseau (routeur par exemple), pour déclarer et allouer une ou plusieurs adresses à un nœud de domaine.

Le RFC 2771 (*An Abstract API for Multicast Address Allocation*, février 2000), décrit quant à lui une interface abstraite de service pour le service d'allocation dynamique d'adresse multicast. Cependant il ne décrit, pas à proprement parler une API concrète pour un langage spécifique, mais plutôt, en termes abstraits, la sémantique de ce service, en incluant les garanties qu'il doit fournir.

Mais il existe aussi des *Internet drafts* sur ce sujet.

Le draft draft-ietf-malloc-ipv6-guide-04.txt regroupe l'ensemble des fonctionnalités qui doivent être implantées par le service responsable de l'allocation des adresses multicast IPv6. Le but est de réduire au maximum la probabilité de collision d'adresses de groupe IPv6 multicast.

Enfin, le draft draft-ietf-malloc-malloc-mib-07.txt définit une portion de la MIB pour gérer l'allocation d'adresses multicast.

La gestion des groupes de diffusion multicast

La gestion des groupes multicast est assurée en IPv6 par le protocole MLD (*Multicast Listener Discovery*). Ce protocole fait partie intégrante de toute pile IPv6, puisque le support du multicast est nécessaire au bon fonctionnement d'IPv6, en particulier de l'autoconfiguration.

Ce protocole existe en deux versions. La première (MLDv1) est définie dans le RFC 2710 (*Multicast Listener Discovery (MLD) for IPv6*, Octobre 1999). C'est celle qui est implantée actuellement dans la majorité des piles IPv6. MLDv1 est directement dérivé de la version 2 du protocole IGMP pour IPv4.

Il existe aussi une version 2 de ce protocole, définie dans le draft draft-vida-mld-v2-02.txt. Ce draft est directement dérivé d'IGMPv3 pour IPv4. Comparé à MLDv1, MLDv2 permet le filtrage de sources multicast, qui permet de recevoir seulement le trafic multicast de certaines sources définies. Une implantation de ce protocole a été réalisée par les laboratoires LIP6 de l'Université Pierre et Marie Curie, Paris et par le LSIIT de l'Université Louis Pasteur de Strasbourg. Cette implantation fonctionne sur FreeBSD 4.3, avec la pile IPv6 de Kame.

MLDv1 :

Le protocole permet à chaque routeur IPv6 de découvrir la présence d'hôtes multicast sur les réseaux auxquels il est directement connecté, ainsi que les groupes de diffusion multicast auxquels sont abonnés les clients. Ces informations sont ensuite utilisées par le protocole de

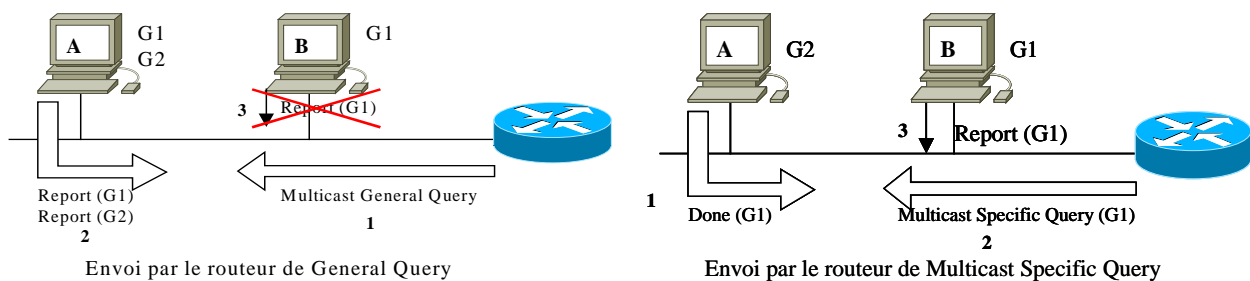
“*Maximum Response Delay*”, pour chacune des adresses multicast à laquelle elle est abonnée. Lorsque ce compte à rebours arrive à 0, et si aucune autre station abonnée au même groupe de diffusion ne l’a pas déjà envoyé, la station envoie un *Multicast Listener Report* à l’adresse du routeur ainsi qu’à l’adresse du groupe multicast. Ce mécanisme permet de n’envoyer qu’un seul report pour chaque groupe multicast, et ceci quelque soit le nombre de stations abonnées.

Lorsqu’un routeur reçoit un message de type Report, avec une adresse de groupe multicast inconnue du routeur, celui-ci l’ajoute à sa table MLD et initialise un compte à rebours à la valeur « *Multicast Listener Interval* ». Si l’adresse multicast est déjà présente dans la table MLD du routeur, le compte à rebours pour cette adresse est réinitialisé à la valeur « *Multicast Listener Interval* ». Lorsque un compte à rebours d’une adresse devient nul, le routeur assume alors qu’il n’y a plus de stations abonnées à cette adresse multicast sur le lien-local et l’adresse est alors supprimée de sa table MLD.

Lorsqu’une station commence à écouter une adresse multicast, elle doit envoyer un message « *Multicast Listener Report* » pour cette adresse, dans le cas où elle est la première station abonnée sur le lien.

Lorsqu’une station souhaite quitter un groupe, elle envoie alors un message « *Multicast Listener Done* » à l’ensemble des routeurs multicast du lien-local (ff02::2). Le champ *Multicast Address* contient alors l’adresse du groupe multicast dont se désabonne la station.

Lorsqu’un routeur reçoit un message « *Multicast Listener Done* », et si l’adresse multicast correspondant se trouve dans sa table MLD, il envoie alors un « *Multicast-Address-Specific Query* ». Si le routeur ne reçoit pas de « *Multicast Listener Report* » dans un délai prédéfini, il estime alors qu’il n’y a plus de clients abonnés au groupe multicast correspondant, et l’adresse est alors supprimée de sa table MLD.



La construction de l’arbre de diffusion

Nous allons voir dans cette partie les protocoles qui permettent de mettre en place l’arbre de diffusion multicast. L’arbre de diffusion multicast est l’ensemble des chemins d’une source multicast vers l’ensemble des destinataires d’un groupe. Pour chaque couple composé d’un groupe et d’une source, un nouvel arbre peut être mis en place. En général, chaque branche de l’arbre multicast est choisie de telle sorte qu’elle emprunte le plus court chemin.

La mise en œuvre d’un service d’acheminement multicast dont l’étendue est limitée au lien local est simple si la liaison offre un service de diffusion, service qui est offert par tous les réseaux locaux actuels.

Dans les autres cas, la diffusion s’appuie sur trois niveaux protocolaires :

- Le premier niveau est la fonction MLD du protocole ICMPv6, utilisée entre les stations situées sur un même sous-réseau IP et le routeur responsable du routage multicast, appelé routeur désigné (DR : *Designated Router*) au sein de ce sous-

réseau IP. Le routeur prend connaissance des groupes actifs présents sur le réseau grâce à MLD.

- Le deuxième niveau est fourni par les protocoles de construction d'arbres multicast internes. Ce type de protocole est chargé de la mise à jour des tables de routage des routeurs situés à l'intérieur du domaine multicast. Le domaine multicast est un ensemble de nœuds multicast administrés par la même entité. Un exemple de protocole de ce type est PIM.
- Le troisième niveau est fourni par les protocoles de construction d'arbre multicast externes. Ce type de protocole est chargé de la mise à jour des tables de routage des routeurs constituant l'infrastructure permettant l'interconnexion des domaines de routage. Dans ce cas, le routage est externe aux systèmes autonomes. Le protocole BGMP (*Border Gateway Multicast Protocol*) devrait être chargé de remplir ce rôle, mais aucune implantation n'est encore disponible.

Les protocoles de routage internes

Il existe deux types de protocoles de routage internes :

- Les protocoles travaillant en mode dense
- Les protocoles travaillant en épars mode (*sparse mode*)

Les protocoles travaillant en mode dense.

Cette famille de protocole part du principe que le nombre de clients abonnés aux sessions multicast est important et qu'il y a des clients abonnés sur la majorité des réseaux. Il y a alors inondation de l'ensemble des réseaux du domaine par le trafic multicast. Les routeurs n'ayant pas de clients abonnés à ce trafic demandent alors à leurs voisins de cesser l'émission vers eux : il y a alors un mécanisme d'élagage (*pruning*) des branches ne nécessitant pas le trafic multicast et c'est de cette manière que se crée l'arbre de diffusion. Cette méthode est cependant loin d'être optimale lorsque le nombre d'abonné reste limité.

Les principaux protocoles utilisant le dense mode sont DVMRP (*Distance Vector Routing Protocol*) et PIM DM (*Protocol Independant Multicast Dense Mode*).

DVMRP est défini dans trois Internet drafts : le draft-ietf-idmr-dvmrp-v3-10.txt est une mise à jour de la version 1 du protocole défini dans le RFC 1075. Le draft-ietf-idmr-dvmrp-v3-as-00.txt décrit quant à lui une architecture pour l'utilisation de DVMRP version 3 dans un domaine de routage multicast. Enfin, le draft draft-ietf-idmr-dvmrp-mib-11 définit une portion de la MIB pour l'utilisation de protocoles de supervision. Cependant, bien qu'ayant été très utilisé au début du multicast, ce protocole est actuellement en voie de disparition au profit de PIM.

Le protocole PIM DM est défini dans le draft draft-ietf-pim-dm-new-v2-01.txt. Ce protocole de construction de l'arbre multicast, comme son nom l'indique, est indépendant du protocole de routage unicast utilisé, c'est à dire qu'il peut être utilisé quelque soit le protocole de routage unicast utilisé. Cependant, ne gérant pas sa propre table de routage, il utilise la table de routage du protocole unicast. PIM DM est lui aussi de moins en moins utilisé au profit des protocoles travaillant en mode épars (PIM SM par exemple).

Les protocoles sparse mode :

Contrairement au dense mode, chaque routeur ayant des récepteurs enregistrés à un trafic multicast doit faire une demande explicite d'abonnement au groupe de diffusion multicast. Cette méthode évite l'envoi de trafic inutile sur les parties du réseau où il n'y a pas d'abonnés. Un exemple de protocole *Sparse Mode* est PIM SM.

PIM SM est défini dans le RFC 2362 (*Protocol Independent Multicast-Sparse Mode (PIM-SM): Protocol Specification*, Juin 1998). Cependant, bien que le RFC soit toujours officiellement valide, un draft (draft-ietf-pim-sm-v2-new-05.txt) visant à compléter ce RFC, et incluant notamment les extensions pour IPv6 est en cours de discussions à l'IETF.

PIM SM est un protocole de construction de l'arbre de diffusion indépendant du protocole de routage unicast sous-jacent, c'est à dire que, comme PIM DM, il ne tient pas à jour une table de routage spécifique, mais utilise la table de routage du protocole unicast utilisé, quel que soit ce protocole.

Cette définition (qui est celle décrite dans les normes) correspond au cas où la topologie multicast est native, c'est à dire que les réseaux unicast et multicast sont les mêmes. Cependant ceci n'est que très rarement le cas, et la topologie multicast est le plus souvent constituée de tunnels entre des équipements compatibles avec le multicast. Les topologies multicast et unicast sont alors différentes : pour cette raison il serait intéressant d'avoir à manipuler des tables de routage différentes, l'une pour le trafic unicast et l'autre pour le trafic multicast. PIM travaillant directement avec la table de routage unicast, on voit que la définition du protocole est totalement orientée pour les réseaux multicast natifs.

PIM SM construit pour chaque groupe un arbre de diffusion unidirectionnel, chaque arbre prenant racine sur un nœud spécifique appelé RP (*Rendez-vous Point*). Lorsqu'il y a plusieurs sources alimentant le même groupe, les paquets en provenance des différentes sources convergent vers le RP associé au groupe, puis à partir de celui-ci, les paquets empruntent (et donc partagent) l'arbre associé au groupe, ce qui leur permet d'atteindre tous les destinataires membres du groupe.

Le routeur RP est choisi par une élection (ou configuré statiquement) entre les différents routeurs multicast candidats RP du domaine PIM. Les messages utilisés pour l'élection sont des *RP-Candidate* :

```
11:58:57.966140 diogene.ipv6.lip6.fr > 2001:660:10a:6612::1: pim v2
Candidate-RP-Advertisement prefix-cnt=0 prio=1 holdtime=2m30s
RP=diogene.ipv6.lip6.fr
```

La découverte par les routeurs multicast du RP se fait grâce au mécanisme du *Bootstrap Router* (BSR). Un routeur d'un domaine PIM est élu (ou configuré) comme *Bootstrap Router*, et son rôle est d'émettre régulièrement à tous les routeurs qui ont été candidats au titre de *Bootstrap Router* des messages *Bootstrap* qui contiennent l'ensemble des RP du domaine.

```
11:53:43.036754 fe80::204:76ff:fe99:aa26 > ff02::d: pim v2 Bootstrap
tag=5649 hashmlen=126 BSRprio=15 BSR=2001:660:10a:6612::1 (group0: ff00::/8
RPcnt=4 FRPcnt=4 RP0=...) [hlim 1]
```

Le routeur désigné ou DR (routeur multicast qui gère le protocole PIM lorsque plusieurs routeurs d'un même LAN peuvent gérer le PIM) est lui aussi élu, et les messages utilisés pour son élection sont les messages PIM Hello.

```
12:03:47.235377 fe80::204:76ff:fe17:795a > ff02::d: pim v2 Hello (Hold-time
1m45s) [hlim 1]
```

Les messages Hello servent aussi pour connaître les routeurs PIM voisins sur chaque interface. Ces messages sont nécessaires car si un routeur n'a pas envoyé de PIM Hello, il est ignoré par les autres routeurs PIM.

Il est à noter que pour ces trois élections, il est possible de donner une priorité à un routeur pour le favoriser lors de cette election. Dans le cas où deux routeurs possèdent la même priorité, l'élection se fait en fonction de l'adresse IP des routeurs candidats.

Le fonctionnement du protocole suit trois phases principales :

Phase 1 : L'arbre partagé ou arbre RP

Un client souhaite recevoir du trafic multicast d'un groupe défini. Cette demande se fait en utilisant un protocole de gestion des groupes multicast, IGMP pour IPv4 ou MLD pour IPv6. Il y a alors election du DR pour le réseau auquel appartient la station.

Une fois le DR élu, celui-ci envoie un message PIM d'abonnement vers le RP du groupe multicast. On note PIM (*,G) join ce type de message car il permet la réception de paquets multicast pour le groupe G en provenance de n'importe quelle source. Le message va voyager de routeurs en routeurs jusqu'au RP du groupe, et dans chaque routeur traversé, un état associé à l'arbre multicast du groupe G est créé. Finalement, le PIM (*,G) join atteint soit le RP, soit un routeur possédant déjà un état (*,G) associé au groupe. Il y a alors création au niveau de ce routeur d'une branche de l'arbre RP, qui est partagé entre toutes les sources. Quand plusieurs récepteurs adhèrent à un groupe, leur message Join convergent vers le RP de ce groupe, ce qui forme l'arbre multicast pour ce groupe, arbre ayant pour racine ce RP. Cet arbre est appelé RPT (*Rendez-vous Point Tree*) et il est qualifié d'arbre partagé puisqu'il sera utilisé pour atteindre tous les destinataires du groupe quelque soit l'émetteur du paquet multicast.

Des messages PIM d'adhésion sont envoyés périodiquement tant qu'au moins un destinataire est membre du groupe. Quand tous les destinataires situés sur un sous-réseau IP quittent un groupe, le DR peut envoyer un message PIM d'élagage (PIM Prune). Une durée limite de validité étant associée à chaque adhésion, si aucun message PIM ne parvient, l'adhésion sera résiliée.

Dès qu'une station émet des données vers un groupe multicast, le DR envoie des paquets encapsulés dans un datagramme unicast ayant pour adresse de destination le RP associé au groupe. Le RP désencapsule alors les données et les propage suivant l'arbre RPT associé au groupe. Les paquets multicast suivent alors les informations (*,G) placées dans les routeurs formant l'arbre RPT. Ces paquets sont dupliqués aux nœuds qui forment de nouvelles branches, et donc parviennent à l'ensemble des destinataires membres du groupe. L'encapsulation des paquets vers le RP est appelé enregistrement et les paquets encapsulés sont des PIM register.

Voici un exemple de la trace générée par un tel paquet avec tcpdump

```
10:42:53.931265 diogene.ipv6.lip6.fr > 2001:660:10a:6613::1: pim v2
Register minotaure.ipv6.lip6.fr > ff0e::2:fd22: no next header [hlim 1]
```

Lorsque la phase 1 est terminée, le trafic est envoyé encapsulé par le DR vers le RP, et le RP le renvoie alors nativement sur l'arbre RP, et vers les récepteurs.

Phase 2 : L'acheminement spécifique

La phase 1 reste assez inefficace, puisqu'il est nécessaire d'encapsuler les données entre le DR et le RP, ce qui est une opération coûteuse pour un routeur.

Bien que la technique proposée lors de la phase 1 puisse perdurer indéfiniment puisque c'est le mode de fonctionnement par défaut, le RP va chercher à basculer vers une technique d'acheminement native, sans encapsulation. Dans ce cas, lorsque le RP reçoit un message PIM Register, contenant un paquet multicast provenant d'un émetteur S pour un groupe G, il peut entreprendre de construire un chemin d'acheminement spécifique pour l'émetteur noté (S,G). Pour ce faire, le RP envoie un message PIM (S,G) Join vers l'émetteur. Ce message provoque dans tous les routeurs traversés la création d'un état multicast spécifique (S,G). Ces états ne seront utilisés que pour transmettre les paquets multicast émis par S vers le groupe G. A la fin, ce message PIM (S,G) Join arrivera sur le sous-réseau IP hébergeant l'émetteur S.

Quand les paquets multicast arrivent nativement vers le RP, celui-ci reçoit les messages en double puisqu'il reçoit à la fois le trafic encapsulé et le trafic natif. Le RP détruit alors la copie des messages encapsulés et il envoie un message PIM Register-Stop vers le DR de l'émetteur afin de lui demander de stopper l'encapsulation devenue inutile.

Voici la trace d'un *packet register stop* avec tcpdump

```
17:54:08.812869 2001:660:10a:6612::1 > 2001:660:285:2:260:8ff:fe71:c72b:  
pim v2 Register-Stop group=ff0e::e002:f60d  
source=2001:660:285:2:a00:20ff:fe78:7323
```

A la fin de cette phase, le trafic émis par la source S pour le groupe G suit le chemin d'acheminement spécifique jusqu'au RP, puis utilise l'arbre RPT (associé au groupe G) pour atteindre tous les destinataires du groupe G. Là où les deux arbres se croisent, et pour certains destinataires proches de la source, le trafic peut passer du chemin d'acheminement spécifique à une branche de l'arbre RPT, en évitant ainsi un détour via le RP.

Phase 3 : L'arbre des plus courts chemins

La phase 2 supprime le surcoût introduit par l'encapsulation entre l'émetteur et le RP, cependant, cela n'optimise pas complètement le chemin suivi par les paquets multicast : pour de nombreux destinataires, le transit par le RP provoque un détour important si on compare ce chemin avec le chemin le plus court entre l'émetteur et chaque destinataire.

Pour obtenir des délais plus courts, le DR associé au destinataire peut lancer la construction d'un arbre des plus courts chemins spécifiques à l'émetteur S pour un groupe G. On désigne cet arbre par SPT (*Shortest Path Tree*). Dans ce cas, le DR émet un message PIM (S,G) Join vers l'émetteur. Cela crée des états spécifiques (S,G) dans les routeurs concernés sur le chemin de l'émetteur.

Un fois le message arrivé à l'émetteur ou à un routeur possédant déjà l'état (S,G), les paquets multicast émis par l'émetteur S n'ont plus qu'à suivre les états (S,G) présents dans les nœuds des routeurs formant l'arbre multicast (S,G).

De la même manière que précédemment, le DR du destinataire peut recevoir deux copies des paquets multicast, une provenant du RP et ayant suivi l'arbre RPT associé et une autre provenant de l'arbre des plus courts chemins. Le DR détruit alors les paquets provenant du RPT, et envoie un message PIM (S,G) Prune vers le RP.

Conclusion

A l'issue de ces trois phases, on peut avoir des destinataires qui reçoivent leur trafic multicast des trois manières différents, certains par le RPT (encapsulé ou non) et d'autres par le SPT. Le protocole PIM est donc un protocole qui peut s'adapter facilement à son environnement, en utilisant à bon escient les trois modes proposés non exclusifs.

Les protocoles inter-domaines

Les interactions entre les différents protocoles de routage multicast sont définies par le RFC 2715 (*Interoperability Rules for Multicast Routing Protocols*, octobre 1999). Ce RFC définit quelques règles que les protocoles de routage multicast doivent suivre afin de pouvoir collaborer convenablement.

Cependant, aucun protocole inter-domaine n'est pour l'instant implanté pour IPv6, ni sur des implémentations libres, ni sur des offres commerciales.

Performance

Le RFC 2432 (*Terminology for IP Multicast Benchmarking*. K. Dubray, octobre 1998) permet de définir de nombreux critères de performance mesurables dans le cas particulier du multicast. Pour chaque critère de performance, il est donné sa définition, son intérêt, et son unité de mesure.

Sécurité

L'usage du multicast devient de plus en plus courant : pour cette raison, il est nécessaire de pouvoir garantir un certain niveau de sécurité lors de l'utilisation du multicast. C'est pourquoi, l'IETF s'est penché sur le problème.

Tout d'abord, se pose le problème du chiffrement des flux multicast. Il est en effet important de pouvoir garantir à la fois l'intégrité des données véhiculées, l'authentification ainsi que la confidentialité des communications. Mais cela est problématique puisque cela nécessite de distribuer de manière sûre, une clé à chaque membre d'une session. Cette distribution des clés se fait traditionnellement par une entité du réseau, le KDC (*Key Distribution Centre*), mais cette méthode ne convient pas pour des communications à travers le monde entier, où les utilisateurs peuvent se trouver aux quatre coins du monde. De plus, le problème se complique encore lorsque l'on est en mode *Sender-Specific*, auquel cas le trafic doit être authentifié pour chaque émetteur. Le RFC 1949 (*Scalable Multicast Key Distribution*, mai 1996) permet de résoudre ce problème en proposant une solution évolutive au problème de la distribution des clés.

Le RFC 2588 (*IP Multicast and Firewalls*, mai 1999) définit le comportement des firewalls en ce qui concerne le trafic multicast. La configuration d'un firewall est en effet différente entre le multicast et l'unicast, et il est nécessaire de définir de nouvelles règles spécifiques au multicast.

Le draft draft-ietf-msec-gkmarch-02.txt présente un exemple pour une architecture de management de clés pour MSEC qui supporte de nombreuses applications, et de nombreux protocoles de transport.

Supervision

Le draft draft-ietf-malloc-malloc-mib-06.txt définit une portion de la MIB pour gérer l'allocation d'adresses multicast.

Transition v4-v6

Le draft draft-ietf-ngtrans-mtp-01.txt décrit un traducteur multicast IPv6-IPv4 qui joue le rôle de proxy IGMP/MLD.

Les applications

Quelques RFCs se focalisent sur le côté applicatif. Par exemple, le RFC 2090 présente les options nécessaires au fonctionnement multicast du TFTP (*Trivial File Transfert Protocol*). Il permet par exemple à plusieurs stations de télécharger un même fichier en optimisant les ressources réseau.

3.2.3. Quelques implantations actuelles

Kame

Kame est la pile IPv6 développée par le consortium japonais Wide. Ce consortium regroupe de nombreuses compagnies japonaises (Fujitsu, Hitachi, NEC, Toshiba, ...) et l'objectif du projet Kame est de créer une implantation 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. Elle est disponible en ftp anonyme sur <ftp.kame.net>.

6Wind

Société française issue de Thomson spécialisée dans les solutions d'accès pour les réseaux IPv6, elle a été l'un des premiers équipementiers à commercialiser un routeur d'accès IPv6. Et depuis le début de l'année, le routeur supporte le multicast IPv6, et sera donc intégré dans notre projet.

6Wind ayant prêté au GIP Renater des routeurs, nous avons avec eux des relations privilégiées. Nous pouvons utiliser leur matériel, les tester, remonter les éventuels bugs des implémentations avancées, voire même demander des modifications dans l'implémentation de certaines fonctions. En contre partie, nous essayons d'utiliser au maximum leur matériel, afin de pouvoir servir de référence auprès d'éventuels clients.

Cisco

Cisco offre la possibilité de migrer vers IPv6 avec certains de ses routeurs. Cependant, le multicast IPv6 n'est à ce jour supporté qu'avec une version de test de l'IOS.

Juniper

Juniper propose des routeurs IPv6, mais le multicast n'est pas encore proposé.

3.3. DSTM (Dual Stack Transition Mechanism)

3.3.1. Présentation

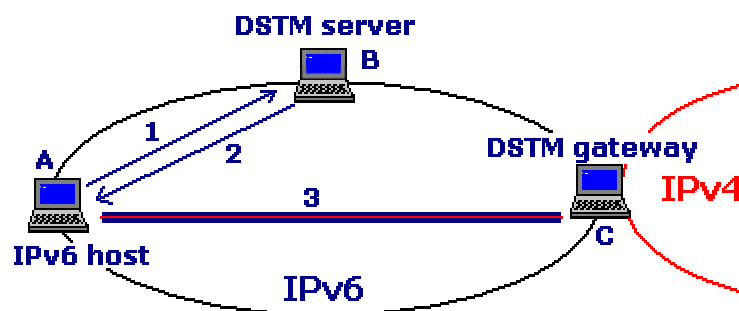
Le DSTM est un service qui permet, durant le déploiement d'IPv6, d'avoir des réseaux purs IPv6, et de pouvoir continuer à communiquer avec le monde v4. Cette connectivité est obtenue grâce à l'utilisation de tunnels automatiques IPv4 dans IPv6 ainsi que grâce à des adresses IPv4 temporaires allouées aux hôtes en ayant besoin.

Cette méthode de migration présente de nombreux avantages :

- Le système permet au site entièrement v6 de garder une connectivité avec le monde IPv4. Le site IPv6 n'est alors pas isolé du monde IPv4.
- Les applications qui ne sont pas encore portées sur IPv6 peuvent continuer à être utilisées, et c'est le serveur et la passerelle qui se charge de la communication avec le monde IPv4 à l'aide de tunnels IPv4 dans IPv6.
- Le réseau est configuré uniquement pour IPv6, et il n'y a pas besoin de prévoir un plan d'adressage IPv4.
- Le besoin en adresses IPv4 est réduit puisque les adresses IPv4 sont allouées temporairement, et réutilisées par la suite.
- Ce mécanisme est complètement transparent au type de protocole transporté ainsi que de l'application utilisée.

3.3.2. Fonctionnement

Le draft draft-ietf-ngtrans-dstm-08.txt définit le concept du DSTM. Il a été rédigé par Jim Bound, Laurent Toutain, Octavio Medina, Francis Dupont Hossam Afifi et Alain Durand. Le schéma suivant permet d'expliquer le fonctionnement du mécanisme DSTM.



Un réseau DSTM nécessite trois types d'équipements :

- Un serveur DSTM
- Une passerelle DSTM
- Un client DSTM

Ces trois types d'équipements se trouvent sur un réseau pur IPv6. L'objectif est alors de pouvoir permettre aux clients DSTM de pouvoir communiquer en IPv4 avec les clients du monde IPv4.

Lorsqu'un client DSTM souhaite communiquer avec un autre client en IPv4, il le fait savoir au serveur DSTM. Le rôle du serveur DSTM est d'administrer un pool d'adresse IPv4, et de fournir aux clients le désirant une adresse IPv4 temporaire lui permettant d'entrer en communication avec le monde IPv4. Le client doit alors configurer automatiquement sa pile IPv4 avec l'adresse fournie par le serveur. La réponse du serveur contient aussi le temps de vie de l'adresse fournie ainsi que les informations sur le DSTM Gateway (Tunnel End Point). Lorsque le client a configuré sa pile IPv4, les communications en IPv4 sont encapsulées en IPv6 jusqu'au TEP, dont le rôle est de décapsuler le trafic et de l'envoyer sur le réseau IPv4. La passerelle DSTM conserve une table de correspondance entre les adresses IPv4 et IPv6 des clients de l'Intranet.

Un exemple de déploiement de DSTM est fourni dans le draft draft-ietf-ngtrans-dstm-overview-00.txt

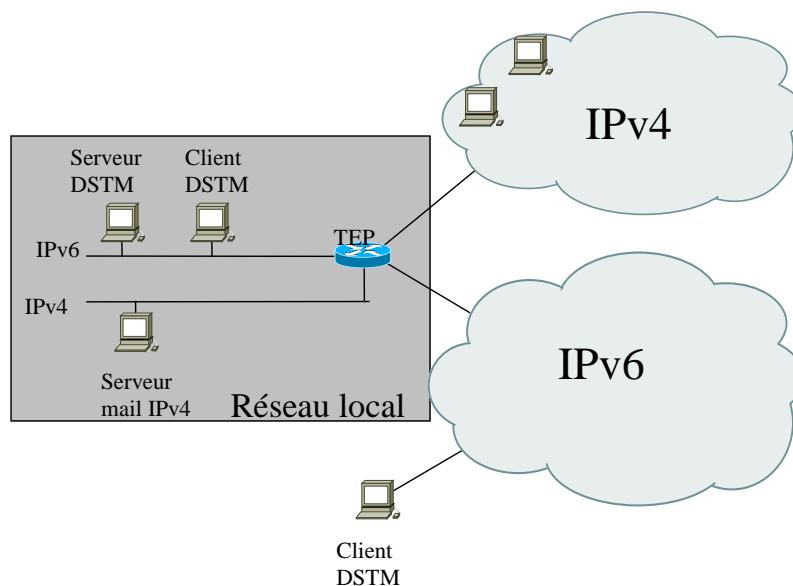
Dans les drafts que nous venons de voir, il est sous-entendu que les clients DSTM se trouvent sur le réseau local du site. Mais le draft draft-richier-dstm-vpn-00.txt définit un nouvel usage possible du mécanisme de DSTM, lorsque les clients sont situés en dehors du réseau local. Ce nouvel usage est appelé le scénario VPN.

Le DSTM peut être utilisé pour accéder à des ressources IPv4 lorsque seule une connectivité IPv6 est disponible, mais ceci n'est pas limité au seul réseau local. En effet, si un client DSTM se trouve en n'importe quel point d'un réseau IPv6, et que le TEP et serveur DSTM sont accessibles en v6, il est possible d'avoir une connectivité v4 en utilisant le mécanisme DSTM. Cette connectivité v4 peut être limitée au réseau local du site où se situe le serveur DSTM et le TEP (allocation d'adresses IPv4 privées) ou alors peut permettre une connectivité à l'ensemble du monde IPv4. Grâce à ce mécanisme, les opérateurs pourraient ne fournir qu'une connectivité IPv6 et obtenir la connectivité IPv4 sur l'ensemble de leur réseau en utilisant le mécanisme de DSTM. On peut aussi utiliser le DSTM lors de la phase de transition, si le site ne dispose pas de toutes ces ressources en IPv6, afin de pouvoir accéder à celles-ci quelque soit son lieu de connexion sur le réseau IPv6. Une application simple serait pour l'accès au serveur mail du site par exemple, si celui-ci est uniquement IPv4.

De plus, l'investissement pour la mise en place de ce mécanisme est minimal, puisqu'il suffit d'installer un serveur DSTM et un TEP.

La seule différence avec le scénario classique d'utilisation du DSTM concerne la sécurité : en effet, un client, avant d'obtenir une adresse IPv4 temporaire doit être authentifié au niveau du serveur DSTM.

Voici un schéma résumant l'utilisation de ce scénario VPN.



Afin de limiter l'usage du DSTM, une extension au système d'allocation d'adresses est apparue dans le draft draft-shin-ngtrans-dstm-ports-00.txt. Ce système permet à plusieurs clients DSTM d'utiliser en même temps la même adresse IPv4, grâce à l'utilisation de ports différents.

Lorsqu'un client DSTM demande à communiquer avec un hôte IPv4, le serveur DSTM, en plus de fournir l'adresse, peut fournir une plage de ports à utiliser. Cependant, cette solution ne convient que si l'application a la possibilité de choisir le port lors d'une communication.

Comme nous l'avons vu, DSTM requiert un mécanisme d'allocation d'adresse. Le mécanisme existant le plus utilisé est DHCPv6. C'est pour cette raison qu'il existe deux drafts permettant de définir de nouvelles options au protocoles DHCPv6, afin qu'il puisse fonctionner avec

DSTM. Le premier, draft-ietf-dhc-dhcpv6-opt-dstm-01.txt définit les options générales alors que draft-ietf-dhc-dhcpv6-opt-dstm-ports-01 définit les options permettant l'utilisation de la gestion des ports par le serveur DSTM.

Une variante du mécanisme, défini dans le draft draft-bereski-ngtrans-nd-dstm-01.txt, explique comment utiliser un routeur plutôt qu'un serveur pour l'allocation d'adresses. Le protocole utilisé alors pour l'allocation des adresses IPv4 serait alors Neighbor Discovery.

3.3.3. Implantations actuelles

Les implantations du DSTM sont encore assez rares : cependant, l'ENST Bretagne a implanté sur FreeBSD les trois composants nécessaires au mécanisme du DSTM. L'implantation est disponible pour FreeBSD 3.4 avec la pile IPv6 de l'INRIA et avec les versions 4.3, 4.4 et 4.5 associées à la pile Kame.

Tous les détails sur le travail de l'ENST Bretagne, et les sources des implantations sont disponibles sur <http://www.ipv6.rennes.enst-bretagne.fr/dstm/index.html>

Les routeurs 6Wind implantent aussi la fonction de DSTM. Ils jouent en effet le rôle de DSTM Gateway (*Tunnel End Point*).

4. Déploiement du M6Bone

4.1. Objectifs

Le premier objectif de ce stage est de remettre en place un service de test multicast sur le pilote IPv6 de Renater. Ce service doit permettre de connecter les sites connectés au pilote IPv6 de Renater, mais aussi des sites n'ayant pas encore de connexion IPv6. Ce réseau de test doit permettre de valider différentes configurations matérielles et logicielles en vue d'un déploiement en tant que service de production comparable au FMBone.

Démarré une première fois en octobre 2001, le service était stoppé lorsque je suis arrivé. Il a fallu donc remettre en route le service, tout en mettant à jour toutes les versions des systèmes d'exploitation des routeurs utilisés, afin de pouvoir passer de PIM DM à PIM SM.

Les équipements de Renater ne supportant pas le multicast IPv6, le M6Bone est un réseau de tunnels permettant d'encapsuler le trafic IPv6 multicast dans des paquets IPv6 ou IPv4 suivant la connectivité du site distant.

Le point final de cette partie concernant IPv6 multicast est la diffusion du Séminaire X-Aristote du 6 juin 2002, habituellement diffusés en multicast IPv4, en multicast IPv6.

L'objectif à plus long terme est la mise en place de ce service IPv6 multicast sur l'ensemble des POPs du pilote IPv6 de Renater 2.

4.2. Contexte

4.2.1. Une technologie jeune

IPv6 est un protocole jeune, et les réseaux IPv6 commencent à se déployer à travers le monde. Tous les constructeurs ne proposent pas de gamme IPv6, et ceux qui en proposent une n'offrent pas toujours toutes les fonctionnalités souhaitées. Le multicast fait souvent partie des fonctions qui ne sont pas implantées dans les routeurs commerciaux. Seuls 6Wind et Cisco permettent aujourd'hui de faire du multicast IPv6. Il est donc nécessaire d'utiliser des PC pour jouer le rôle de routeurs... Cependant, le fait d'utiliser des implantations récentes, et parfois de test implique qu'il est possible de rencontrer de nombreux bugs. Il est donc nécessaire de se tenir au courant régulièrement de l'évolution des différentes implantations, et ne pas hésiter à signaler un bug afin de le faire corriger.

4.2.2. Premiers déploiements au sein de Renater

Un service multicast IPv6 avait déjà été mis en place par Jérôme Durand, précédent stagiaire de l'INSA de Lyon entre juillet et décembre 2001. Le sujet de son stage était en effet le déploiement sur le pilote IPv6 de Renater d'un service de test multicast.

Il avait aussi permis la diffusion en multicast IPv6 du séminaire X-Aristote du 20 décembre 2001.

Cependant, lorsque je suis arrivé en mars, le service était arrêté, et il fallait redéployer le réseau. De plus, le service déployé au mois de décembre fonctionnait en PIM Dense Mode, qui est un protocole de routage assez inefficace. Ce choix avait été forcé par le fait que l'implantation de PIM Sparse Mode fournie avec la pile Kame ne fonctionnait pas correctement, et de nombreuses coupures dans les flux multicast avaient lieu. Il avait donc fallu se retourner vers le PIM DM. Cependant, le démon PIM SM présent sur la pile Kame, a été corrigé vers le début du mois de mars 2002. Il était donc possible de déployer un réseau Sparse Mode. De plus, les routeurs étaient alors des FreeBSD 4.3 or lorsque je suis arrivé, la dernière version disponible de FreeBSD était 4.5.

Je devais donc redéployer le réseau, en mettant à jour l'ensemble des routeurs en FreeBSD 4.5, ainsi qu'en utilisant la dernière pile Kame.

4.3. Les enjeux

Le problème d'IPv6 est qu'il est encore trop considéré comme un protocole de test et pas comme un protocole de production.

L'enjeu, à travers ce projet est de participer à la promotion du protocole IPv6 en France et à travers le monde, et de montrer qu'il est possible de mettre en place un service de haute technologie de diffusion multicast IPv6 permettant d'utiliser des applications de visioconférences multicast sur ce réseau.

De plus, ce projet permet à de nombreux acteurs de se joindre à nous en tant que sites clients connectés au réseau. La connexion au M6Bone est alors une bonne occasion de se former au protocole IPv6 et au multicast. Ils doivent eux-mêmes configurer les équipements nécessaires à leur connexion, ainsi que les stations de travail clientes. Cette phase de formation sur un réseau de test est importante en vue de l'utilisation future du protocole dans un contexte de production.

Le réseau M6Bone peut alors être le lieu d'expérimentations intéressantes dans le domaine d'IPv6, entre les différents membres du réseau. Il est possible de tester et de développer des applications multicast IPv6, ou encore des mécanismes de transition entre IPv4 et IPv6.

Enfin, ce projet sert de test grandeur nature pour les équipementiers et les développeurs, puisque nous utilisons dans ce réseau les fonctionnalités les plus avancées des implantations, celles justement qui peuvent encore contenir quelques bugs. Nous pouvons aussi indiquer quelles sont les fonctionnalités manquantes pour un fonctionnement optimal de leur équipement.

4.4. Réalisation

4.4.1. Equipements utilisés

Les équipement utilisés pour le déploiement du réseau multicast IPv6 (M6bone) sont de deux types : les routeurs, et les stations clientes, utilisées pour effectuer des tests de visioconférence multicast par exemple.

Routeurs

Lors du premier déploiement, les routeurs étaient exclusivement des PC FreeBSD. En effet, lorsque le projet a commencé en juillet 2001, seule la pile Kame du consortium WIDE permettait de faire du routage IPv6 multicast. Il n'était pas possible d'utiliser un routeur commercial car aucun ne permettait de faire du multicast en IPv6.

Lorsque j'ai commencé mon stage, Les PC sous FreeBSD n'étaient plus la solution unique pour faire du routage multicast, les routeurs 6wind intégrant désormais cette fonctionnalité. Cependant, pour redémarrer le service, j'ai choisi de continuer à utiliser des PC sous FreeBSD.

Mais une fois que le réseau a fonctionné correctement, et Renater disposant de 3 routeurs 6Wind 6200, j'ai essayé de les intégrer au réseau existant, et avec succès.

Stations clientes

Les stations clientes servent à faire principalement de la visioconférence en multicast.

Les applications utilisées sont appelées les outils du Mbone. Ces outils, développés par l'University College London (UCL), sont très utilisés dans le monde académique.

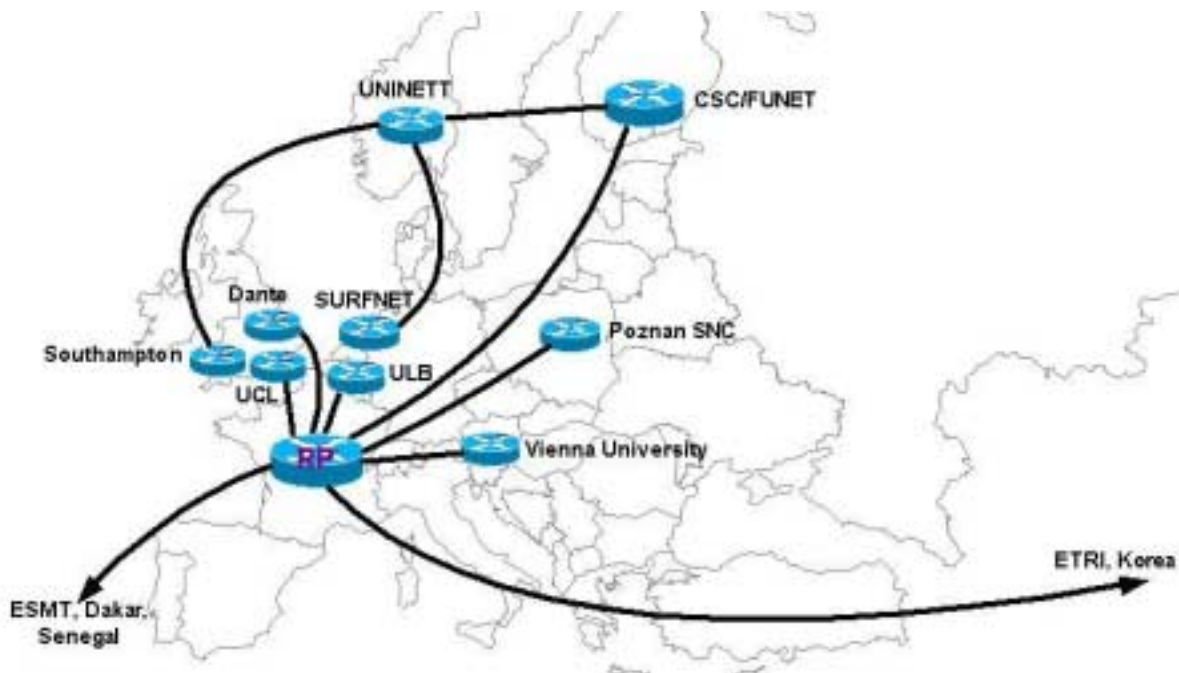
N'importe quelle station possédant une pile IPv6 et permettant de faire fonctionner les outils du Mbone peut convenir. Les systèmes d'exploitation correspondant aux deux critères sont nombreux, mais les principaux sont Windows 2000, Windows XP, Linux, FreeBSD.

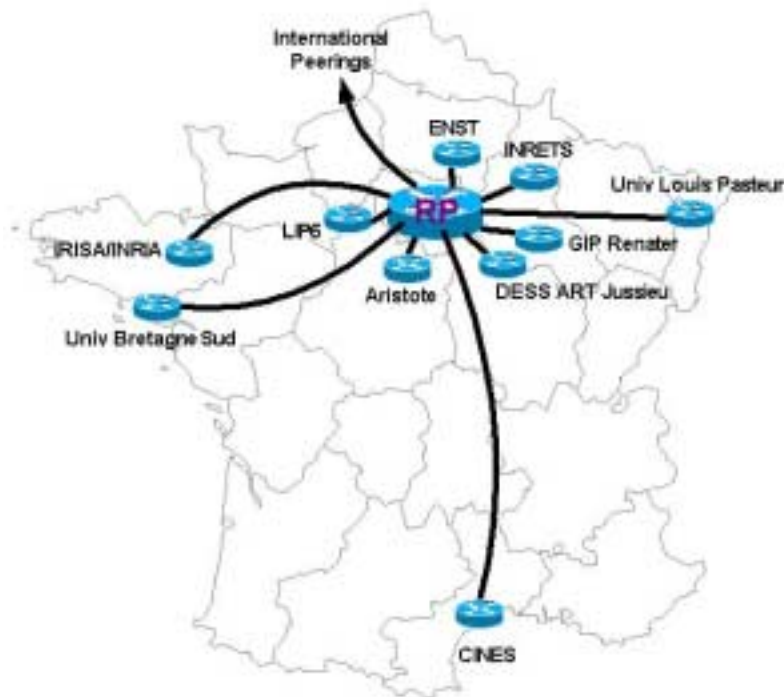
4.4.2. Architecture globale du réseau

Le M6bone est un réseau de tunnels entre un routeur central et les différents sites connectés aux réseaux. Il est nécessaire de mettre en place des tunnels entre le routeur central et les sites car le pilote IPv6 de Renater, basé sur du matériel Cisco, ne supporte pas le multicast. Il est donc nécessaire d'encapsuler le trafic multicast dans du trafic unicast pour permettre de traverser les équipements unicast du réseau.

L'architecture est une architecture en étoile, autour d'un routeur central, situé dans les locaux du GIP Renater.

Voici le schéma général du M6Bone, avec le routeur central, connecté par des tunnels aux sites connectés.





Routeur central

Equipement

Le routeur central du M6bone est un PC sous FreeBSD 4.5 avec la pile IPv6 Kame. Ce PC possède un tunnel pour chaque site connecté au M6bone. Ce tunnel peut être v6 dans v6 si les sites possèdent une connexion v6 ou alors il peut être v6 dans v4 si ils ne possèdent qu'une connexion v4.

Routing

Le routeur central possède une route par défaut statique vers le routeur d'accès IPv6 du GIP Renater. Le routage unicast est assuré par le protocole RIPng, implémenté sur Kame par le démon route6d. Route6d est utilisé sur l'ensemble des tunnels du routeur ainsi que sur l'interface physique sur laquelle se trouve le routeur du GIP Renater.

Au début, chaque site pouvait annoncer les réseaux qu'il souhaitait annoncer, en sachant qu'ils ne devaient n'annoncer que les préfixes participant au multicast IPv6. Cette solution avait pour avantage de laisser libre chaque site, mais elle supposait une entière confiance aux sites concernés. Je me suis cependant vite rendu compte que ce système devenait difficilement gérable au fur et à mesure que le réseau s'étendait. En effet, il nous est arrivé de recevoir sur le routeur central de nombreuses annonces ne correspondant pas du tout à des annonces du M6bone. Le problème est que cette « pollution » d'annonces RIPng se propage vers l'ensemble des sites connectés au réseau, ce qui n'est pas souhaitable. J'ai donc décidé de mettre en place un filtrage des annonces RIPng, c'est à dire que pour chaque interface tunnel, seuls les préfixes participant au M6bone sont acceptés sur le routeur central. Avec cette solution, le choix des préfixes utilisés se fait par chaque site lors de la première connexion au M6bone. Le filtrage se fait en utilisant l'option `-L` du démon de routage RIPng (route6d).

Le protocole multicast utilisé est PIM SM. Ce protocole est implémenté sur Kame par le démon pim6sd. De la même manière que route6d, pim6sd est lancé sur l'ensemble des interfaces tunnels ainsi que sur l'interface physique. Le routeur central est configuré pour être le point de rendez-vous principal du réseau, ainsi que le *bootstrap router*.

Enfin, il faut noter, que les tables de routage spécifiques au multicast ne sont pas encore implantées. Ce manque va nous poser quelques problèmes, pour la connexion d'un site ayant une connexion IPv6 :

En effet, lorsque le tunnel entre un site et le routeur central est établi, la route pour atteindre ce site passe par le tunnel, afin de pouvoir communiquer en multicast. Par conséquent, chaque site annonce avec RIPng les préfixes de son réseau dans le tunnel. Cependant, il y a un problème lors de la mise en place du tunnel. Comment atteindre le routeur distant pour mettre en place le tunnel, si le tunnel qui est la seule route connue n'existe pas !!

Il est alors nécessaire d'ajouter une route statique vers le routeur distant par le réseau classique unicast. Cette solution va faire en sorte que les communications entre le routeur central et le routeur d'accès IPv6 multicast du site se fassent toujours à travers le réseau unicast. Une conséquence est qu'il n'est pas possible d'utiliser des applications multicast sur cette station. Il est alors nécessaire d'avoir une station dédiée aux applications multicast.

Sécurité

La sécurité est un élément important dans la mise en place d'un tel réseau. Il est important de pouvoir limiter les connexions à leur strict minimum, afin d'éviter une éventuelle compromission du routeur central. La mise en place de cette politique de sécurité est d'autant plus importante que le routeur se trouve sur une partie du réseau du GIP qui n'est pas filtrée en entrée. La sécurité doit se faire pour les connexions en IPv4 bien sur, mais aussi en IPv6. Même si le protocole est encore peu répandu, il est possible, en piratant une machine en v6, de rebondir ensuite sur tout le monde v4 !!

Par conséquent, un firewall prenant en compte IPv4 et IPv6 doit être mis en place sur le réseau. Le firewall déployé est ipfw pour IPv4 et ip6fw pour IPv6. Ces deux firewalls peuvent être activés en recompilant le noyau de FreeBSD avec l'option correspondante. Reste ensuite à définir les règles les plus appropriées.

En IPv4, la tâche a été assez facile. Il a suffi de refuser tout trafic TCP et UDP venant de l'extérieur, et de ne laisser passer en entrée que le trafic venant des sites connectés au M6Bone. Par conséquent, chaque nouvelle connexion au M6Bone nécessite une nouvelle règle du firewall autorisant le trafic avec le routeur distant.

Pour les règles concernant IPv6, j'ai essayé de retranscrire les règles mises en place en IPv4, mais je n'y suis pas parvenu. La mise en place des mêmes règles avait pour effet de bloquer l'ensemble du trafic multicast. Après de nombreux tests, et avec l'aide précieuse de Konstantin Kabassanov, nous avons pu mettre en place un firewall IPv6 fonctionnant correctement. Les difficultés venaient du fait que le routeur possédait de nombreuses interfaces tunnels ayant chacune une adresse IP différente. En effet, il n'était pas possible de mettre en place une règle générique pour toutes les interfaces : il fallait entrer la même règle autant de fois qu'il y a d'interfaces, physiques et logiques.

La politique de filtrage est la suivante :

- 1- Autoriser tout le trafic venant de l'interface de loopback.
- 2- Autoriser tout le trafic ICMP sur le lien-local.
- 3- Autoriser tout le trafic multicast venant de n'importe quelle interface.
- 4- Autoriser les paquets PIM venant de n'importe quelle interface.
- 5- Autoriser tout le trafic venant du réseau local.
- 6- Autoriser tout le trafic ayant pour adresse source ou adresse de destination un site connecté au M6Bone.
- 7- Rejeter tout le reste.

Sites connectés

Peuvent être connectés trois types de sites.

- Les sites possédant une connexion IPv6.
- Les sites ne possédant qu'une connexion IPv4.
- Les utilisateurs isolés ne possédant qu'une connectivité IPv4.

Au 1^{er} Août 2002, voici la liste des sites qui sont connectés au M6Bone :

France

GIP Renater
Association Aristote (Paris)
LIP6 (Paris)
Université de Bretagne Sud (Vannes)
Université Louis Pasteur (Strasbourg)
CINES (Montpellier)
Showroom ENST Paris
DESS DART Jussieu (Paris)
INRETS (Joinville)
IRISA/INRIA (Rennes)

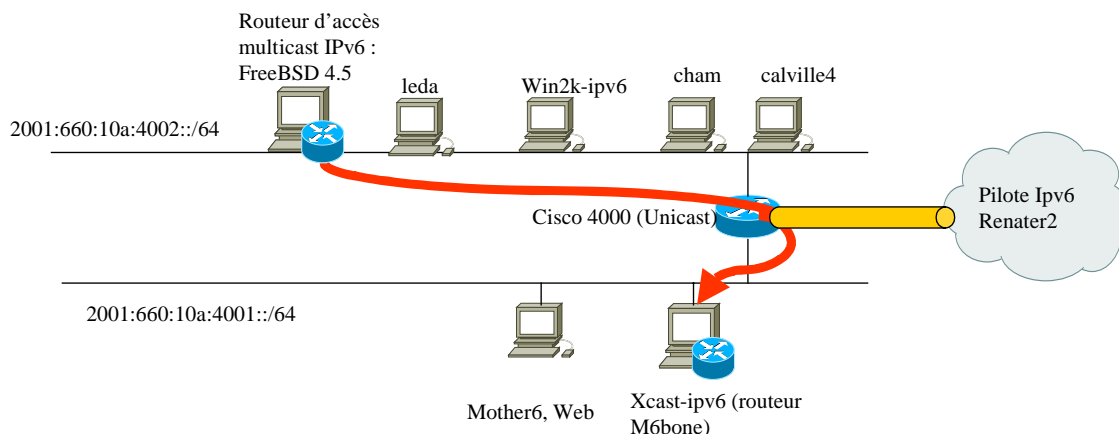
Etranger

ESMT Dakar
Vienna University Computer Center
(Autriche)
Poznan SNC (Pologne)
CSC/FUNET (Finlande)
Université Libre de Bruxelles (Belgique)
ETRI (Corée)
University College of London (UK)
Dante (Cambridge, UK)
UNINETT (Norvège)
University of Southampton (UK)
Surfnet (Holland)

Evolution de l'architecture locale GIP

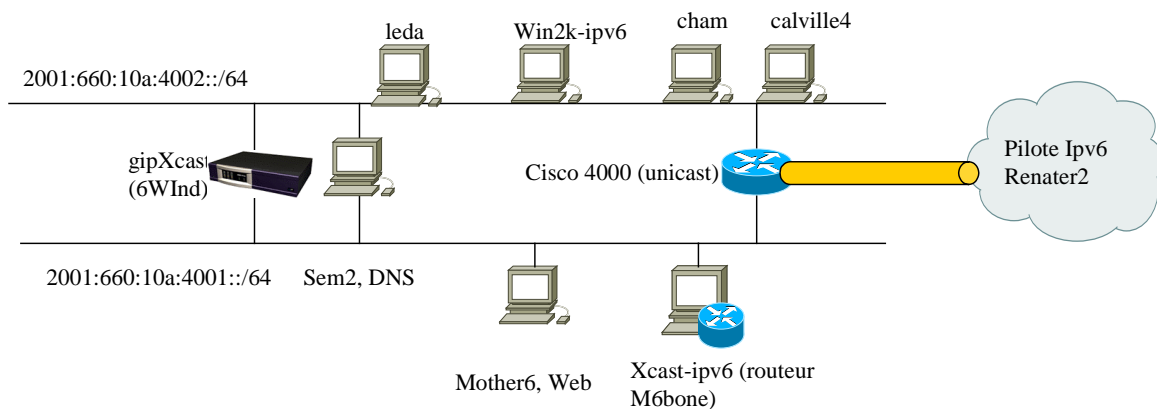
L'accès du GIP Renater au M6bone se fait par l'intermédiaire d'un routeur d'accès.

Lorsque je suis arrivé, le routeur prévu pour permettre l'accès du GIP au M6Bone était un PC FreeBSD. J'ai donc commencé par mettre en place le PC comme routeur d'accès. Le schéma du réseau était alors le suivant :



Le routeur central du M6Bone se trouvait sur le 2001:660:10a:4001::/64, alors que le routeur d'accès se trouvait sur le 2001:660:10a:4002::/64. Cette solution nous obligeait à encapsuler le trafic entre les deux routeurs, le Cisco 4000 ne supportant pas le multicast IPv6. Cette solution était donc loin d'être optimale.

C'est pour cette raison que nous avons rapidement décidé d'utiliser un des trois routeurs 6WindGate 6211 comme routeur d'accès, et de remplacer le PC FreeBSD en place. Celui-ci est directement connecté au M6Bone à travers le réseau local et évite ainsi l'utilisation d'un tunnel pour communiquer avec le routeur central.



Les machines clientes utilisent le préfixe 2001:660:10a:4002::/64, alors que le routeur central du réseau est sur le 2001:660:10a:4001::/64. Cette architecture permet de bien séparer le routeur central du réseau du GIP hébergeant les clients multicast IPv6.

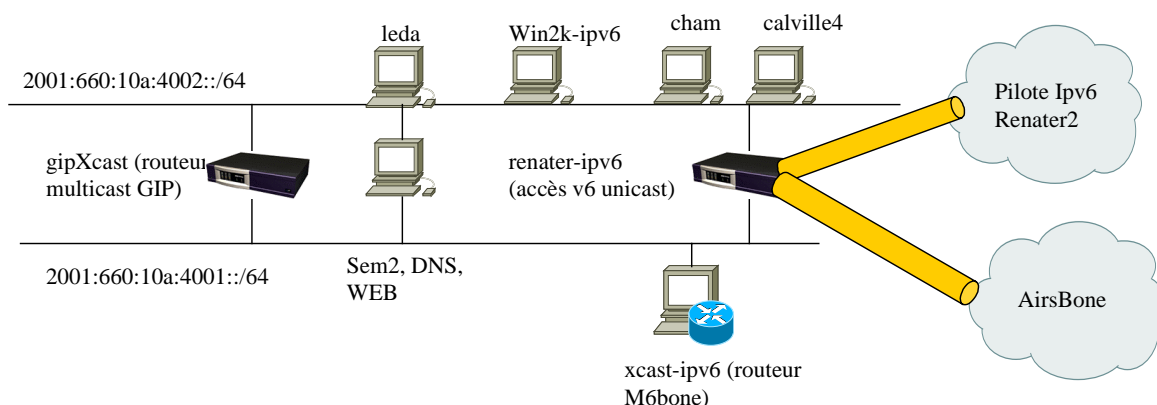
Le routeur d'accès du GIP possède lui une connexion sur les deux réseaux.

RIPng est utilisé par le 6Wind pour faire les annonces du préfixe 2001:660:10a:4002::/64 vers le routeur central. De son côté gipXcast reçoit l'ensemble des annonces RIPng de xcast-ipv6. De plus, Le 6WindGate a une route par défaut statique vers le routeur Cisco 4000. Cette route permet au routeur de rediriger le trafic unicast (dont les routes ne sont pas transmises par RIPng) vers le réseau unicast IPv6.

Le protocole de routage multicast utilisé est le même que sur l'ensemble du réseau, à savoir PIM SM.

Nous avons utilisé cette architecture jusqu'au mois de juin. Nous avons alors décidé de modifier notre accès IPv6. Il se faisait jusqu'alors avec le routeur Cisco 4000. Le problème est que l'implantation d'IPv6 sur le Cisco est assez limitée et ne permet pas par exemple de mettre une politique de filtrage. C'est la raison pour laquelle nous avons décidé de remplacer le routeur Cisco par un routeur 6Wind, qui permet le filtrage.

Pour réaliser cette migration, nous avons utilisé un deuxième routeur 6Wind. Nous avons configuré sur ce nouveau routeur le multicast IPv6 afin qu'il devienne le nouveau routeur d'accès au M6Bone. Une fois cette configuration effectuée et testée, Le multicast IPv6 a été supprimé du 6Wind existant.



Nous avons alors mis en place sur l'ancien routeur gérant le multicast un tunnel IPv6 dans IPv4 permettant la connexion avec le pilote IPv6 de Renater 2. Le protocole de routage utilisé dans ce tunnel est BGP4+. Une fois le tunnel en place, il est alors possible de faire annoncer au routeur d'accès 6Wind les deux préfixes 2001:660:10a:4001::/64 et 2001:660:10a:4002::/64 sur le lien local correspondant, à la place du Cisco. Les clients IPv6 du réseau local reçoivent alors automatiquement comme adresse de la passerelle l'adresse du routeur 6Wind.

Une fois les vérifications effectuées sur le bon fonctionnement de l'accès IPv6 au GIP, nous avons pu déconnecter le Cisco.

Comme on peut le voir sur le schéma, un tunnel a aussi été créé vers le AirBone qui est le backbone du projet @irs++.

Evolutions possibles du M6Bone

Gestion des scopes

Je n'ai effectué aucune manipulation permettant de tester le fonctionnement des scopes permettant de restreindre le trafic à une partie du domaine PIM. Il serait donc intéressant de pouvoir mettre en œuvre ce genre de test sur le M6Bone, afin de pouvoir voir si l'implantation existe, et si oui, si elle fonctionne correctement. Cette fonctionnalité permettrait de pouvoir limiter un trafic multicast à un site ou à un campus par exemple, et éviterait ainsi d'encombrer le réseau inutilement pour des trafics locaux.

Routeur central

Un prochain objectif serait de remplacer le routeur central actuel, un PC FreeBSD par un routeur 6WindGate 6200. Nous avons voulu le faire, mais l'implantation actuelle du 6OS 5.1.3-RELEASE ne le permet pas, et pour plusieurs raisons :

La première est due au fait que le nombre de tunnel est limité à 8 par type de tunnel. Par exemple, il n'est pas possible de faire plus de 8 tunnels 6in4, ou 8 tunnels 6in6. Le routeur central ayant pour le moment 12 tunnels 6in6, et ce nombre augmentant régulièrement, il n'a pas été possible de placer le 6WindGate en tant que routeur central.

De plus, nous nous sommes rendu compte que le protocole RIPng n'est pas supporté dans les tunnels 6in6. Or le RIPng est le protocole de routage unicast utilisé sur l'ensemble du M6bone. Il aurait donc fallu passer au routage statique pour utiliser ce routeur comme routeur central, ce qui n'est pas très intéressant.

De plus, il serait intéressant de pouvoir avoir deux tables de routages distinctes, une pour le multicast et une pour l'unicast, pour éviter les problèmes lors de l'établissement des tunnels 6in6. La gestion du routage serait alors plus fine qu'elle ne l'est actuellement.

Toutes ces remarques ont été transmises à la société 6Wind afin qu'ils puissent nous fournir au plus vite des solutions à ces problèmes. Ce retour d'expérience est assez important pour 6Wind, puisqu'il leur permet de mieux cerner les points forts et les faiblesses de leur équipement, et ainsi de faire évoluer au mieux leur gamme de produit. De plus, l'utilisation de leur matériel au sein de Renater, et sur un réseau d'avant-garde comme le M6Bone, permet à 6Wind d'avoir une référence intéressante auprès d'éventuels futurs clients.

La référence serait encore plus intéressante si le routeur était au cœur du réseau : c'est pourquoi ils devraient nous fournir une version du 6OS corrigeant les défauts de la version actuelle et nous permettant d'utiliser le 6WindGate en tant que routeur central.

Routeurs d'accès multicast

Jusqu'au début du mois de juin, la version de FreeBSD utilisée pour les routeurs d'accès devait obligatoirement être FreeBSD 4.5, avec la pile Kame.

En juin, FreeBSD 4.6 est sorti, et cela a posé quelques problèmes. En effet, des sites ont essayé de se connecter avec FreeBSD 4.6, sans que cela fonctionne. Après de nombreuses recherches, nous nous sommes rendu compte que le démon PIM SM de Kame pour FreeBSD 4.6 pre-release était buggé. Le bug a donc été corrigé, et FreeBSD 4.6 cohabite maintenant sans problèmes avec les autres équipements du réseau.

4.4.3. La diffusion du séminaire X/Aristote du 6 juin 2002.

Aristote est une association créée en 1988 et régie par la loi de 1901, et qui regroupe de grands organismes ou entreprises français intéressés en tant qu'acteurs ou utilisateurs, à l'évolution des réseaux. Une de ces activités est l'organisation régulière de séminaires sur des sujets techniques.

Ce séminaire est habituellement diffusé de deux manières différentes :

- Sur le Fmbone, en utilisant les outils sdr, vic et rat en multicast IPv4
- En streaming Real Audio.

LORS DE CETTE CONFERENCE, DEUX AUTRES EMISSIONS ONT ETE UTILISEES :

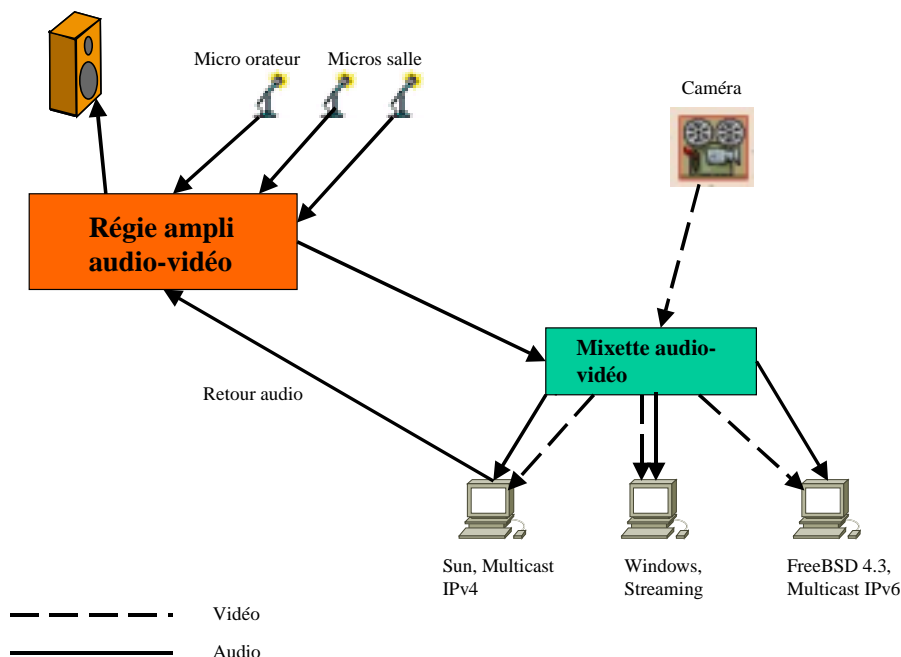
- Diffusion sur le M6bone en multicast IPv6.
- Diffusion de test en IPv4, à partir du flux en IPv6, en utilisant une passerelle IPv6<-> IPv4 développée par Luc Beurton, UBS.

A partir du site du séminaire, nous allons émettre donc trois flux différents : le multicast IPv4, le multicast IPv6 et le streaming Real Player.

Fonctionnement de la partie audio-vidéo

Les trois émetteurs se partagent une même source audio et vidéo, et c'est une table de mixage audio et vidéo qui redistribue l'audio et la vidéo aux trois stations chargées de l'émission sur le réseau.

Voici le schéma de l'architecture audio-vidéo du séminaire :



Les flèches permettent de montrer le sens des flux. On peut alors remarquer que la partie audio est unidirectionnelle, sauf pour la diffusion multicast IPv4 pour laquelle un retour était possible.

Schéma du réseau de diffusion

Le séminaire Aristote se déroule dans les locaux de l'Ecole Polytechnique, à Palaiseau. Le site ne dispose que d'une connectivité IPv4. La connectivité multicast IPv6 sera donc obtenue en faisant un tunnel v6 dans v4 entre un routeur d'accès IPv6 multicast à Polytechnique et le routeur central du M6bone au GIP Renater. (cf schéma ci-dessous)

L'idée retenue est de créer sur le site de Polytechnique un réseau uniquement v6 qui communique avec le réseau IPv4 grâce à un routeur v6/v4.

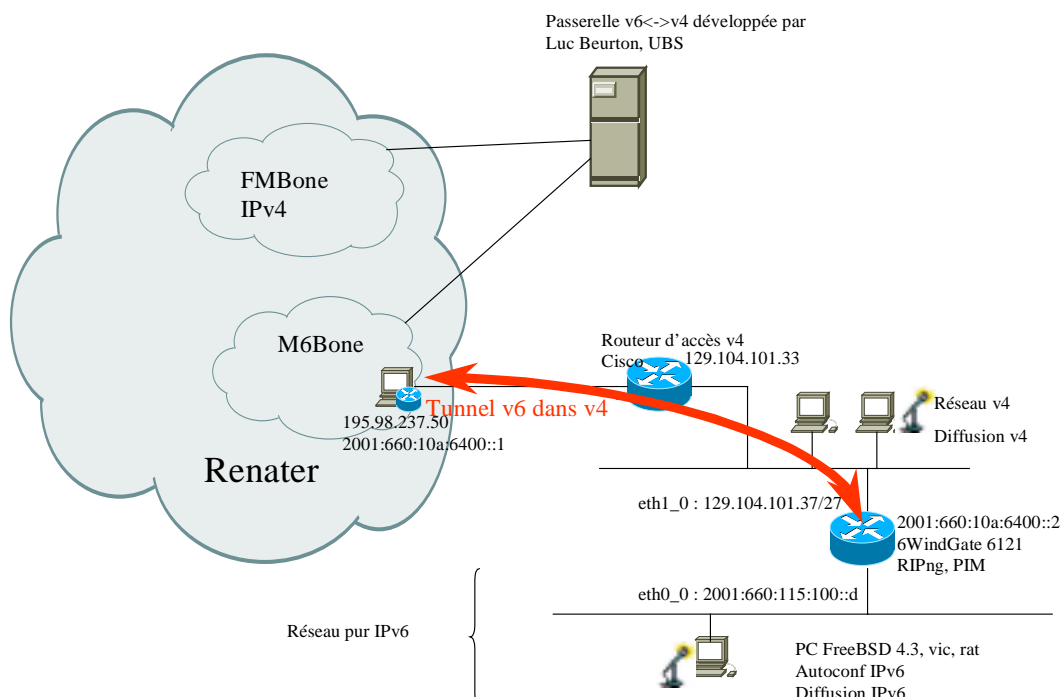
L'équipement informatique utilisé pour la diffusion du séminaire est constitué d'un routeur IPv6 multicast, d'une station émettrice (vic et rat IPv6), et d'un hub. Le routeur IPv6 multicast est un routeur 6Wind 6121, et la station est un FreeBSD 4.3.

Le routeur IPv6 est connecté à un réseau local qui est lui même connecté au routeur d'accès IPv4 de l'école Polytechnique. La communication entre les deux routeurs se fait donc en IPv4. Sur son autre interface, le 6Wind ne possède qu'une adresse IPv6, et ne communique avec la station FreeBSD qu'en IPv6. Le routeur 6Wind se charge alors de l'encapsulation du trafic IPv6 multicast en trafic IPv4 jusqu'au routeur central du M6Bone, dans les locaux du GIP Renater. Le routeur central redistribue ensuite les flux aux clients abonnés au groupe de diffusion. L'abonnement peut se faire à l'aide de l'utilitaire sdr v3.0 porté pour IPv6.

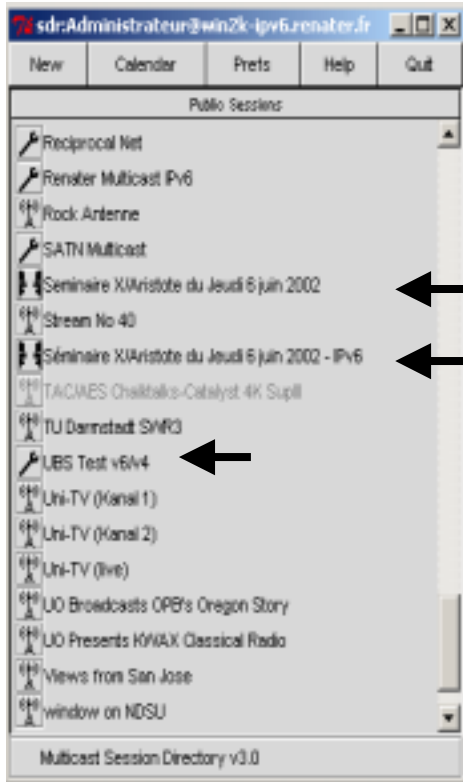
Le routeur 6Wind utilise pour le routage unicast, RIPng, qui est utilisé sur l'interface tunnel. Il reçoit aussi l'ensemble des annonces RIPng du routeur central du M6Bone. Pour le routage multicast, PIM SM est utilisé.

Le trafic peut ensuite être reçu par l'ensemble des sites connecté au M6bone, ainsi que par la passerelle v6-v4, qui retransmet les flux vers le FMBone IPv4.

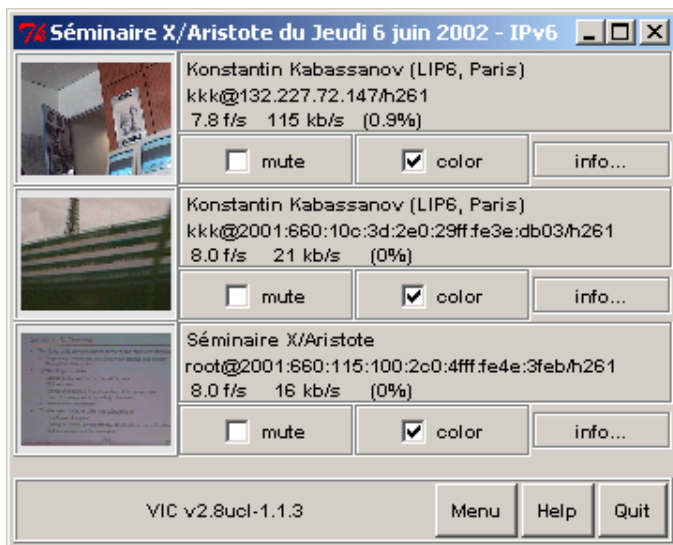
Voici le schéma du réseau utilisé.



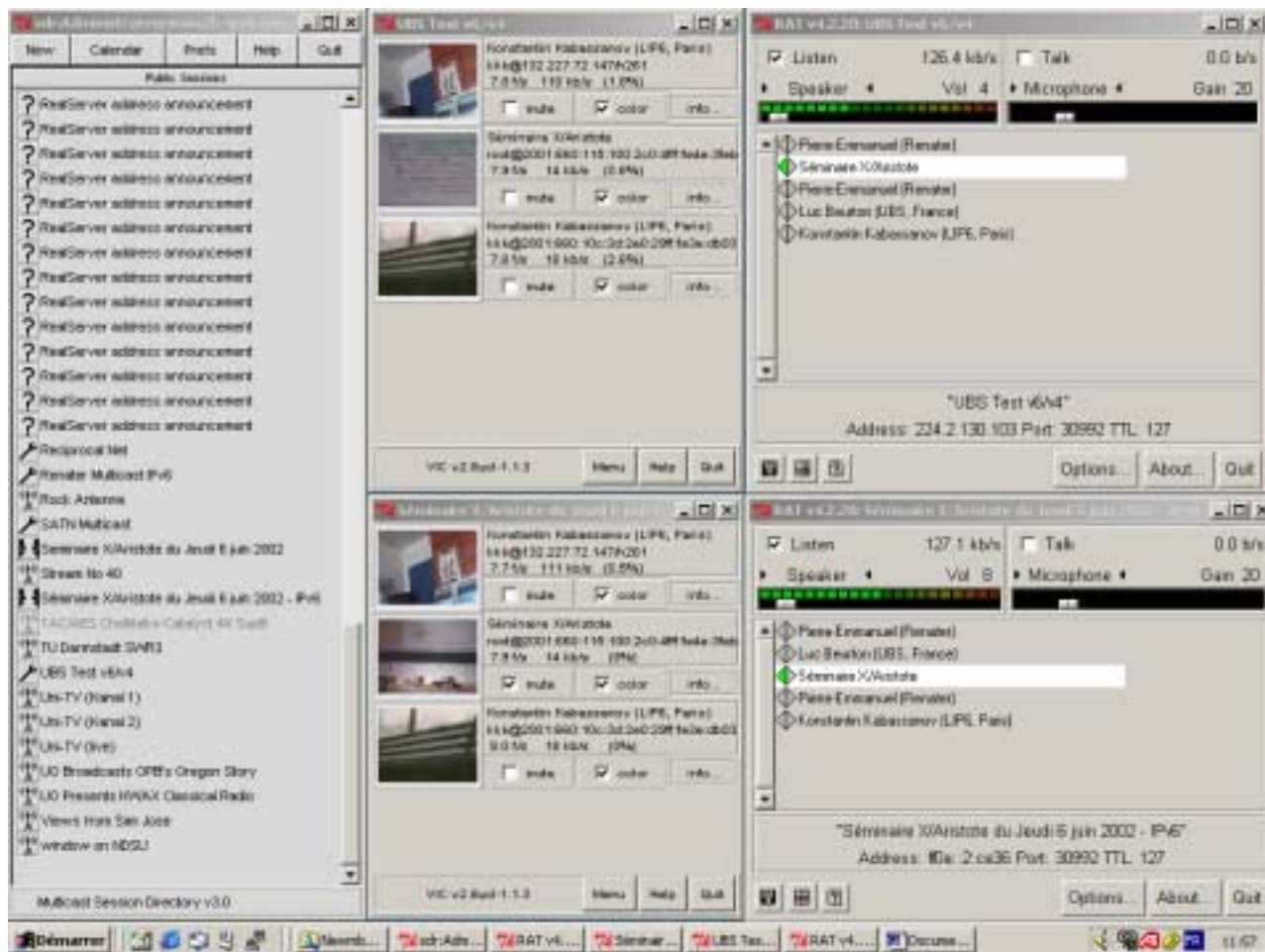
La retransmission de ce séminaire sur le M6Bone a été un succès : nous avons pu, durant toute la journée, diffuser ce séminaire directement en v6, et en v4 grâce à la passerelle. Voici quelques copies d'écrans :



Cette application, sdr, permet de voir « l'ensemble » des annonces de sessions multicast. Ici, la session 3.0 IPv6 du logiciel permet de voir à la fois les annonces de sessions IPv4 et IPv6. Nous pouvons voir trois sessions relatives au séminaire :
 « Séminaire X/Aristote du jeudi 6 juin 2002 » qui est l'annonce officielle de la retransmission en v4.
 « Séminaire X/Aristote du jeudi 6 juin 2002 – IPv6 » qui est l'annonce de retransmission en IPv6.
 « UBS Test v6-v4 » qui est l'annonce de la retransmission en v4 grâce à la passerelle v6-v4.



Copie d'écran de l'application vic utilisée pour l'émission de la vidéo. Nous pouvons voir trois vidéos :
 - La vidéo de la retransmission du séminaire avec une adresse IPv6
 - Une vidéo envoyée par Konstantin Kabassanov en v6
 - Une vidéo envoyée par Konstantin Kabassanov en IPv4, retransmise sur la session IPv6 par la passerelle v6<->v4
 Nous voyons donc que la passerelle permet une entière interopérabilité entre les deux mondes, v6 et v4.



Nous voyons sur cette copie d'écran les deux sessions de discussion en IPv6 et en IPv4 via la passerelle. Nous pouvons remarquer que l'on retrouve les mêmes participants sur les deux sessions, alors que certains sont connectés en IPv6 et d'autres en IPv4. La personne connectée en IPv4 peut voir le monde IPv6 et inversement, celui qui est connecté en IPv6 peut voir le monde IPv4, et cela avec la même session. La passerelle permet en quelque sorte d'"unifier" les deux mondes v6 et v4.

Bilan de la Diffusion

Le séminaire X/Aristote du 20 décembre 2001 avait déjà été diffusé en IPv6, mais quelques améliorations ont pu être mises en place lors de cette nouvelle transmission sur le M6Bone.

Tout d'abord, contrairement au séminaire de décembre où PIM DM était utilisé, nous avons pu utiliser PIM SM en IPv6, qui est beaucoup plus efficace. En effet, nous disposons maintenant de deux systèmes permettant de faire fonctionner PIM SM de manière satisfaisante, FreeBSD 4.5 ou 4.6 avec la pile Kame, ainsi que les routeurs 6Wind avec le 6OS à partir de la version 5.1.3 RELEASE.

De plus, comme nous venons de le voir, des routeurs commerciaux permettent de faire du multicast IPv6. Nous avons pu démontrer le bon fonctionnement des routeurs 6Wind lors de cette manifestation, ainsi que la compatibilité avec les PC-routeurs FreeDSD.

Nous avons diffusé la totalité du séminaire durant toute la journée sur le M6Bone, et sur le MBone grâce à la passerelle. Nous avons ainsi prouvé que le service proposé était stable, et permettait la connexion de plusieurs clients sans aucun problème particulier. L'objectif étant

de déployer un service IPv6 multicast sur le réseau Renater, ce genre d'expérience est intéressante.

Une autre évolution par rapport au 20 décembre concerne les sources audio et vidéo. En effet, la diffusion IPv6 et IPv4 le 20 décembre étaient complètement autonomes puisqu'il y avait deux caméras et deux micros pour chacune des diffusions. Nous avons voulu cette fois unifier les deux, en fournissant aux deux émissions les mêmes sources audio et vidéo. Une mixette permettait la distribution de la vidéo ainsi que du son sur les stations devant émettre.

Enfin, le test de passerelle IPv4<-->IPv6 offre des perspectives intéressantes, puisqu'elle permet, lors de l'utilisation des utilitaires du M6Bone, d'unifier le monde v6 et v4, en permettant au monde v6 de voir le monde v4, et inversement. Ce genre de passerelle est très intéressante lors de la migration entre les deux protocoles.

Cette évolution laisse entrevoir des perspectives intéressantes, car au delà de l'unification des sources audio et vidéo, il est possible d'unifier aussi l'émission. En effet, il est possible de ne diffuser qu'en IPv6 sur le site du séminaire, et d'envoyer ensuite le flux sur la passerelle qui se charge de le retransmettre sur le F6Bone en IPv4.

En conclusion, cette diffusion nous a permis de nous donner quelques idées sur la mise en place d'un service IPv6 multicast sur le réseau Renater. Nous avons ici utilisé une solution centralisée où tous les flux arrivaient sur le routeur central du M6Bone dans les locaux de Renater. Une mise en place plus générale du service pourrait se faire en intégrant un routeur IPv6 Multicast dans l'ensemble des Points d'Interconnexion Régionaux du Pilote IPv6. Ces routeurs serviraient alors de point d'accès au service Multicast. Chaque site aurait ensuite son propre routeur multicast relié en tunnel avec un des POPs du M6Bone. Cette solution de déploiement serait moins centralisée que la solution actuelle en étoile. Elle permettrait de plus de gérer des diffusions multicast avec des scopes différents. Si une diffusion ne concerne qu'un site, les flux n'auraient alors pas besoin de remonter jusqu'au routeur central, mais le routeur du PIR pourrait se charger de tout. Enfin, l'interconnexion avec le F6Bone pourrait se faire grâce à des passerelles IPv6 <-> IPv4. L'idéal serait d'avoir une passerelle sur chacun des PIRs, et ainsi de pouvoir translater toutes les sessions publiques afin de pouvoir les suivre indifféremment en IPv6 ou en IPv4, quelque soit la diffusion initiale.

4.5. Documentation

J'ai essayé tout au long de ce stage de documenter au maximum ce que j'ai effectué.

Le début du stage a surtout été de la recherche d'information en ce qui concerne IPv6 et le multicast. Cette recherche était nécessaire à ma compréhension du projet, et à son bon déroulement. J'ai donc été amené à lire de nombreux RFC¹ sur le sujet.

J'ai rédigé, à partir de cette recherche, une documentation détaillée sur le multicast permettant de parcourir l'ensemble des RFC de l'IETF existant sur le sujet. J'ai classé chaque RFC et ai fait un petit résumé sur chacun. Bien sûr, ce document ne remplace pas la lecture du RFC, mais il permet de se faire une idée sur son contenu avant de le lire si besoin est. Une version est disponible en ligne sur <http://sem2.renater.fr/biblio/xcastv6>

J'ai aussi mis à jour entièrement le site Web du M6Bone. Ce site a pour vocation de rendre la connexion au M6Bone le plus simple possible, en fournissant de nombreux exemples sur les différents types de matériel utilisable ou sur leur configuration. Il présente aussi les caractéristiques du réseau, les sites connectés, les sessions multicast à venir, etc. Ce site est disponible sur <http://sem2.renater.fr/m6bone>

¹ Tous les standards d'Internet sont décrits dans les RFC (Request For Comments). Ce sont des documents issus de l'IETF (Internet Engineering Task Force), organisme chargés de la normalisation des protocoles utilisés pour Internet.

De plus j'ai cherché à faciliter la tâche à la personne qui serait chargé de reprendre mon travail en réalisant des pages Web internes permettant de trouver rapidement un renseignement sur le M6Bone ou sur un des sites connectés. Par exemple, le site contient le nom, les contacts, les adresses IP du routeur distant, du tunnel de tous les sites connectés. De cette manière, un simple coup d'œil peut permettre de voir le renseignement recherché.

Il me semble que cette transmission de la connaissance acquise pendant le stage est primordiale si on veut que le projet soit suivi par la suite de manière efficace.

5. Mise en place d'une plate-forme DSTM

5.1. Introduction

Avec le développement progressif d'IPv6, va se poser le problème de la transition entre les deux réseaux IPv4 et IPv6. Le développement de techniques assurant la transition devient un point clé dans le développement à grande échelle du protocole IPv6. En effet, le succès d'IPv6 est fortement lié à l'interopérabilité avec IPv4, car la majorité des informations et des utilisateurs sont aujourd'hui disponibles avec IPv4, et il serait difficilement possible, sous prétexte de migrer vers IPv6 de se passer purement et simplement d'IPv4 !! Il faut donc prévoir des mécanismes qui permettent de migrer vers IPv6, sans pour autant se couper des ressources du monde IPv4.

De la même manière, il faut pouvoir permettre, dans la mesure du possible de ne pas restreindre l'accès aux ressources v6 uniquement aux personnes connectées en IPv6. La migration est donc un véritable challenge pour les années à venir.

Différents mécanismes de transition ont été proposés, parmi eux, l'utilisation d'une double pile IPv4/IPv6, le NAT PT, ISATAP ou encore le DSTM.

Nous avons choisi de travailler sur le DSTM car c'est un mécanisme puissant qui permet de s'affranchir totalement du plan d'adressage IPv4 sur le réseau local.

Nous avons donc voulu tester le fonctionnement et les implantations actuelles du DSTM, en vue d'une possible mise en œuvre sur le pilote IPv6 de Renater.

C'est à quoi j'ai consacré la deuxième partie de mon stage, du mois de juin au mois d'août.

5.2. Objectif

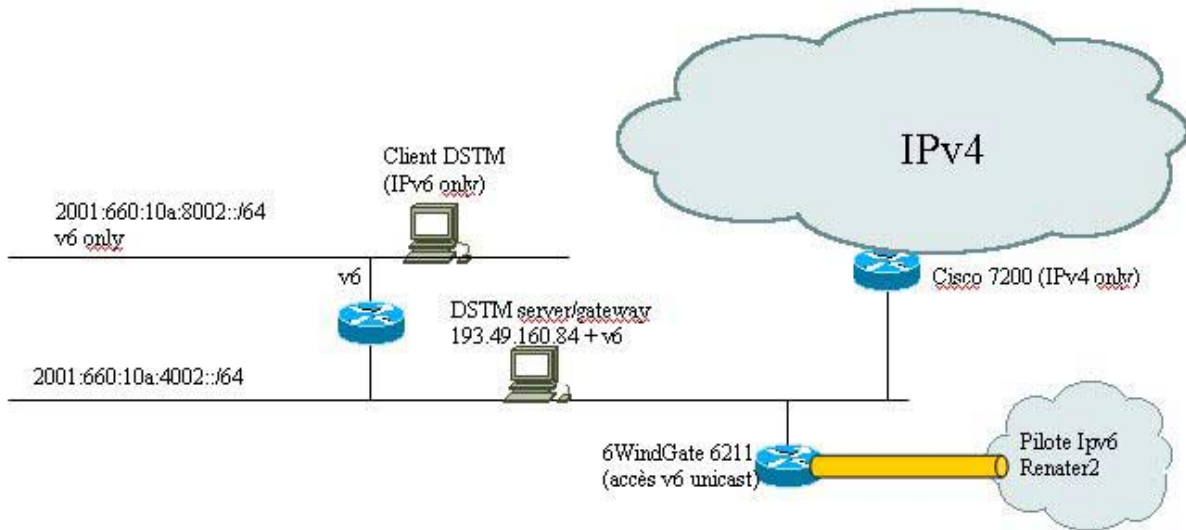
L'objectif final est la mise en place d'un service DSTM sur le pilote IPv6 de Renater.

Avant de déployer cette technologie sur l'ensemble du réseau Renater, il est cependant intéressant de pouvoir la tester localement.

C'est donc la méthode qui a été choisie. Nous avons souhaité tester les implantations du DSTM sur le réseau local de Renater. Cette expérimentation permet de valider le fonctionnement du protocole, ainsi que les différentes implantations existantes.

De plus, installer une maquette de DSTM est le meilleur moyen d'acquérir une meilleure connaissance du protocole qui sera utile lors d'un éventuel déploiement plus global sur l'ensemble du réseau Renater.

5.3. Réalisation de la maquette



Cette maquette permet de tester tout d'abord le mécanisme de DSTM sur le réseau local du GIP Renater. La maquette est composée de deux équipements : un serveur/passerelle DSTM et un client DSTM, tous deux des PC sous FreeBSD 4.5. Le client DSTM est localisé sur une partie du réseau uniquement IPv6 alors que le serveur est lui sur un brin Ethernet à la fois IPv6 et IPv4.

Les tests se sont révélés concluants, et le mécanisme a fonctionné correctement. Le client DSTM était capable de communiquer avec l'ensemble du monde IPv4, alors qu'il se trouve sur un sous-réseau uniquement IPv6. De plus, cette connectivité au monde IPv4 se fait de manière complètement transparente pour l'utilisateur : il suffit de lancer un programme (qui peut être lancé automatiquement au démarrage de l'ordinateur) pour que toutes les communications vers le monde IPv4 soient possibles.

Le succès de la mise en œuvre de la maquette nous a permis de mieux comprendre le fonctionnement de DSTM, en particulier en ce qui concerne le routage du pool d'adresse IPv4.

6. Conclusion

Le bilan de ce stage s'avère extrêmement positif.

J'ai eu la chance de pouvoir participer à plusieurs projets différents, avec bien entendu le déploiement du M6Bone, mais aussi le déploiement d'un service DSTM sur le pilote IPv6, ou encore l'évolution du réseau local IPv6 du GIP Renater. Ces travaux assez divers m'ont permis de me frotter à de nombreuses facettes du protocole IPv6, et m'ont permis d'acquérir de nombreuses connaissances sur le sujet.

J'ai pu donc acquérir de nombreuses connaissances pratiques dans de nombreux domaines comme IPv6, le routage, le multicast, ou encore l'administration de systèmes unix. Tous ces domaines viennent s'ajouter aux connaissances acquises durant la formation de télécommunication de l'INSA de Lyon et la complète de manière intéressante.

J'ai pu apprendre à travailler en collaboration avec des personnes venant du monde entier. En effet, de nombreuses personnes ont participé aux évolutions du M6Bone, et il a été très intéressant de pouvoir travailler à plusieurs sur un problème donné.

Je me suis aussi beaucoup amélioré au niveau de la recherche d'informations : j'ai du en effet rechercher des informations sur les protocoles, sur les normalisations en cours, et j'ai donc du me plonger dans des RFC, Internet drafts ou autres documentations techniques en anglais. Je n'avais pas l'habitude de lire ce genre de documents et je pense être maintenant plus efficace lors de recherches d'informations techniques.

En conclusion, ce stage s'est donc révélé très formateur, en de nombreux points. Je pense que ce stage va me permettre d'appréhender la vie professionnelle en ayant les connaissances théoriques et pratiques de base nécessaires au travail d'un ingénieur en télécoms.

Bibliographie

Livres

IPv6, Théorie et Pratique, 3ème édition
Gizelle Cizeault (Ouvrage du G6), O'Reilly

Deploying IP Multicast Networks (Volume 1)
Beau Williamson, Cisco Press

Internet

Site de Renater
<http://www.renater.fr/>

Site de l'association Aristote
<http://www.aristote.asso.fr>

La page du G6
<http://www.ipv6.pps.jussieu.fr/>

Site du M6Bone
<http://sem2.renater.fr/xcast>

Site de FreeBSD
<http://www.freebsd.org/>

Site de Kame
<http://www.kame.net/>

Site de 6Wind
<http://www.6wind.com/> en IPv4 ou <http://www.ipv6.6wind.com> en IPv6

Site de l'IETF
<http://www.ietf.org/>

Site de Konstantin Kabassanov (outils du mbone pour Windows 2000)
<http://www.kabassanov.com/>

Site de the University College of London
<http://www-mice.cs.ucl.ac.uk/multimedia/software/>

Site sur l'implantation du DSTM réalisée par l'ENST Bretagne.
<http://www.ipv6.rennes.enst-bretagne.fr/dstm/index.html>