Conservatoire National des Arts et Métiers

PARIS

CENTRE REGIONAL ASSOCIE DE HAUTE-NORMANDIE

———

MEMOIRE

présenté en vue d'obtenir

le diplôme d'INGENIEUR C.N.A.M.

en

INFORMATIQUE

par

Romain LANGLOIS

——

IPv6 et Multicast

Soutenu le 24 Juin 2004

——

JURY :

Président : M.Anceau

Membres : M. Dieudonné, M. Prigent, M. Voisin, M. Wender

Résumé et mots clés :

IPv6 et Multicast

Installation et configuration du protocole IPv6 sur différentes machines ainsi que sur un réseau de test, avant d'implémenter IPv6 sur le réseau d'exploitation de l'INSA de Rouen.
Le réseau IPv6 mis en place servira de support à différents tests de diffusion en mode Multicast.

L'IETF, organisme technique en charge des protocoles de l'Internet, s'interroge sur les futures évolutions des réseaux. L'évolution la plus importante à ce jour est le choix du successeur du protocole IPv4 (Internet Protocol version 4) assurant actuellement le transport des données sur le réseau Internet est défini : Il s'agit d'IPv6.
Cette nouvelle version du protocole IP se base sur le retour d'expérience issu de l'utilisation d'IPv4 afin de modifier, simplifier, supprimer ou intégrer certaines fonctions. Ce protocole intéresse actuellement le monde de l'Enseignement et de la Recherche et un certain nombre d'entreprises. Certains pays, comme le Japon, encouragent son utilisation.
Parmi les fonctions développées avec IPv6 se trouve la diffusion en mode Multicast (diffusion d'un émetteur vers plusieurs récepteurs, en simultané). Cette fonction existait en IPv4 mais a été affinée. Les techniques concernant le Multicast sont très importantes car un certain nombre de nouvelles applications vont devoir les utiliser.
Ce rapport détaille la mise en place d'un réseau IPv6, comprenant des ordinateurs fonctionnant sous divers systèmes d'exploitation, ainsi qu'un routeur Zebra. Ce réseau a ensuite été utilisé afin de tester la diffusion en mode multicast. L'information à diffuser, choisie arbitrairement, a été une vidéo. Cela posait les contraintes de diffusion utilisant une bande passante importante, en temps réel. Notons que la diffusion d'une vidéo vers plusieurs clients gagne en efficacité en Multicast. L'outil choisi pour la diffusion est VLC, partie de VideoLAN.
IPv6 et VLC semblent, après plusieurs tests, supporter facilement la diffusion Multicast. Cela permet d'envisager des essais de plus grande envergure et d'autres applications de ce mode de diffusion.

Mots Clés : Réseau – Internet Protocol v6 – IGMP – PIM – Multicast – VideoLAN

Keywords : Network - Internet Protocol v6 – IGMP – PIM – Multicast – VideoLAN

TABLE DES MATIERES :

Remerciements :

Je tiens à remercier le personnel et les enseignants du Cnam, Centre Régional Associé de Haute-Normandie, où j'ai effectué l'ensemble du Cycle Probatoire en Informatique.

Les responsables du Service Informatique et Réseau de l'INSA de Rouen , qui m'ont accueilli pendant la réalisation du mémoire, ont su faire preuve d'écoute et de disponibilité, malgré des emplois du temps chargés.

Au sein de ce service, j'ai pu bénéficier de matériel et d'indépendance, en plus du soutien nécessaire à la mise en oeuvre des expérimentations. L'équipe technique du CRIHAN m'a également apporté une aide précieuse.

Je souhaite enfin remercier les professeurs, professionnels et amis qui ont pris le temps de relire ce mémoire, ainsi que les membres du Jury auxquels j'aurai le plaisir de présenter mon travail.

Introduction

Dans l'Internet actuel avec le protocole **IP** (Internet Protocol) version 4, les adresses IPv4 sont sur quatre octets non utilisables entièrement. Ceci limite leur nombre disponible au niveau mondial. Certains pays disposent donc de très peu d'adresses, en Asie (Japon, Chine), dans les pays en voie de développement (Afrique) ou dans les pays vivant actuellement l'explosion des réseaux (Russie, Europe de l'Est...). Les pays occidentaux (Europe de l'Ouest et Etats-Unis) risquent également d'être confronté à ce problème de pénurie dans l'avenir.

Une des manières de contourner ce problème, mise en place actuellement, est d'utiliser des **adresses privées**. (RFC 1918 - address allocation for private internets) . On attribue au client une adresse non routable dans l'Internet lui permettant d'effectuer des requêtes auprès de serveurs, après translation en adresse publique. On peut ainsi créer des réseaux complets, isolés du reste du monde, en attribuant provisoirement une adresse officielle aux quelques hôtes désirant communiquer avec le monde extérieur. Les hôtes communiquant entre eux à l'intérieur du réseau n'ont pas besoin d'adresse officielle.

Ceci suppose que tous les clients disposant d'adresse privée ne souhaitent pas sortir en même temps. On peut alors utiliser des applications « client-serveur » mais cela ne facilite pas l'utilisation de nouvelles technologies basées sur d'autres modèles .

En attribuant des adresses publiques, visibles de l'extérieur, en quantité à chaque utilisateur, l'**IPv6** prône l'égalité avec une vision « serveur-serveur » ou, pour utiliser un terme qui s'est propagé dans le monde applicatif, « **pair à pair** » (peer to peer).

De nouvelles fonctions peuvent ainsi devenir accessibles au public, comme des serveurs personnels, des objets intelligents, la diffusion et les échanges audio et vidéo à grande échelle, la diffusion de logiciels et de mise à jour, la mobilité,... Certains domaines innovants sont déjà en train d'imaginer de nouveaux modes d'exploitation, par exemple dans le cadre d'une maison domotique, adaptée à des exigences de confort ou de santé. Les objets ainsi adressés ne seront plus forcément un ordinateur mais un téléphone, une caméra, un réfrigérateur ou un équipement médical.

Cette profusion d'adresses ne doit pas faire oublier d'autres innovations comme l'autoconfiguration, tendant à rendre le branchement des terminaux « plug and play ». La Qualité de Service, l'identification du type de service ainsi que l'authentification et la sécurisation, utiles pour le commerce en ligne, sont également améliorées.

Se basant sur le retour d'expérience d'IPv4, le **Broadcast** a été abandonné pour être remplacé par un **Multicast** mieux implémenté. Le réseau pourra ainsi servir à envoyer en masse des émissions télévisées ou audio, données intéressant les clients encore bien souvent diffusées par des systèmes **multipoints**, gourmands en ressource.

Il est très important de noter que la **DoD** (Departement of Defense) a décidé le 9 Juin 2003 que tous ses réseaux fonctionneront en IPv6 en 2008 et qu'à partir du 1er Octobre 2003, tous les matériels et logiciels achetés par la DoD doivent supporter IPv6. Il faut rappeler que la DoD est à l'origine de la création des protocoles **TCP/IP** et a imposé l'utilisation de ceux-ci en 1983 sur son réseau ARPAnet, l'ancètre de l'Internet. Le monde de la recherche expérimente actuellement l'IPv6. L'organisation **IPv6 TASK FORCE** se donne pour objectif de réunir les acteurs du déploiement d'IPv6 afin de partager les expériences, coordonner les actions et devenir force de proposition envers les différents gouvernements. L'IPv6 Task Force possède des antennes en Amérique du Nord, Brésil, Europe (antennes locales dans 11 pays dont la France), en Iran, en Inde, en Chine et au Japon. Aux Etats-Unis, MoonV6 fait collaborer plusieurs universités avec taskforce Nord-Américaine. En Europe, **GEANT** (intégrant l'ex-6bone) est une dorsale implémentée pour permettre aux techniciens et chercheurs de tester IPv6.

Le premier objectif de GEANT est de trouver des réponses techniques pour la création d'un réseau de production en résolvant les problèmes de DNS, de routage inter-réseaux et de mettre en place des solutions Multicast. Ceci implique également de s'intéresser aux outils de transition entre IPv4 et IPv6.

Le second objectif, moins matériel, est de comprendre les implications du développement d'IPv6 pour l'utilisateur et les applications . Le projet GEANT complète des projets locaux, comme le projet ISABEL, de l'université de Madrid, qui souhaite utiliser IPv6 pour développer la téléconférence et le travail collaboratif.

En France, le réseau **RENATER** , après avoir effectué des tests avec un préfixe réservé aux expérimentations, a obtenu une classe d'adresses IPv6 officielle. Certaines applications, comme les vidéo-conférences, utilisant les outils Vic et Rat, sont depuis longtemps en phase de test.

Le réseau **SYRHANO**, Système Réseau de Haute Normandie, milite également pour le développement d'IPv6, dans un but de formation et d'expérimentation. **L'INSA de Rouen**, membre du réseau SYRHANO, a obtenu récemment un préfixe IPv6 : 2001:660:7403::

Au cours de mon stage au sein du **SIR**, Service Informatique et Réseau de l'Insa de Rouen, j'ai été chargé d'expérimenter des émissions Multicast sur IPv6. L'établissement ne possédant pas à mon arrivée de réseau IPv6, j'ai dû mettre en place un réseau de test, qui a été la première expérience de l'établissement en la matière. Une fois le fonctionnement de ce réseau avéré, j'ai pu participer à la mise en place du routage IPv6 sur le routeur principal de l'établissement.

Je me suis par ailleurs intéressé au mode d'émission Multicast, tant du point de vue théorique que pratique. Ayant expérimenté auparavant l'émission en mode multipoints sur le plan professionnel avec les serveurs Realvidéo, j'ai choisi comme support concret la diffusion de vidéo. Plusieurs produits sont disponibles. Mon choix s'est porté sur Videolan, projet « open source » réalisé par les étudiants de l'école Centrale de Paris. Ce projet est le plus abouti en matière de Multicast supportant IPv6, il dispose par ailleurs d'une interface utilisateur conviviale (pour le client) et est supporté par plusieurs systèmes d'exploitation (Linux, Windows, BeOS, MacOS X...).

Après une présentation théorique de l'IPv6 puis du principe du Multicast, ce mémoire décrira le coté pratique de la mise en place du réseau IPv6 et du logiciel Videolan sur différents types de machine.

Remarque :

Lors des recherches pour ce mémoire, j'ai constaté que certaines normes, notamment en ce qui concerne IP v6, sont toujours en évolution. Des RFC sont régulièrement diffusées, précisant ou rendant obsolètes les RFC précédentes. Certaines voies, comme par exemple celles concernant l'attribution des adresses IP v6 sont testées, abandonnées et remplacées. Ces changements se répercutent sur la pertinences des monographies, articles et documents basés sur ces RFC.

Dans un soucis d'actualisation du mémoire, les RFC ont été prises en compte jusqu'à la date limite pour la rédaction. La dernière RFC prise en compte est la RFC 3701 6bone (IPv6 Testing Address Allocation) Phaseout *diffusée en Mars 2004.*

Cet avertissement ne concerne que la partie théorique du rapport, la bibliographie en fin de mémoire précise les RFC étudiées et devenues obsolètes.

I. Théorie de l' Internet Protocol version 6

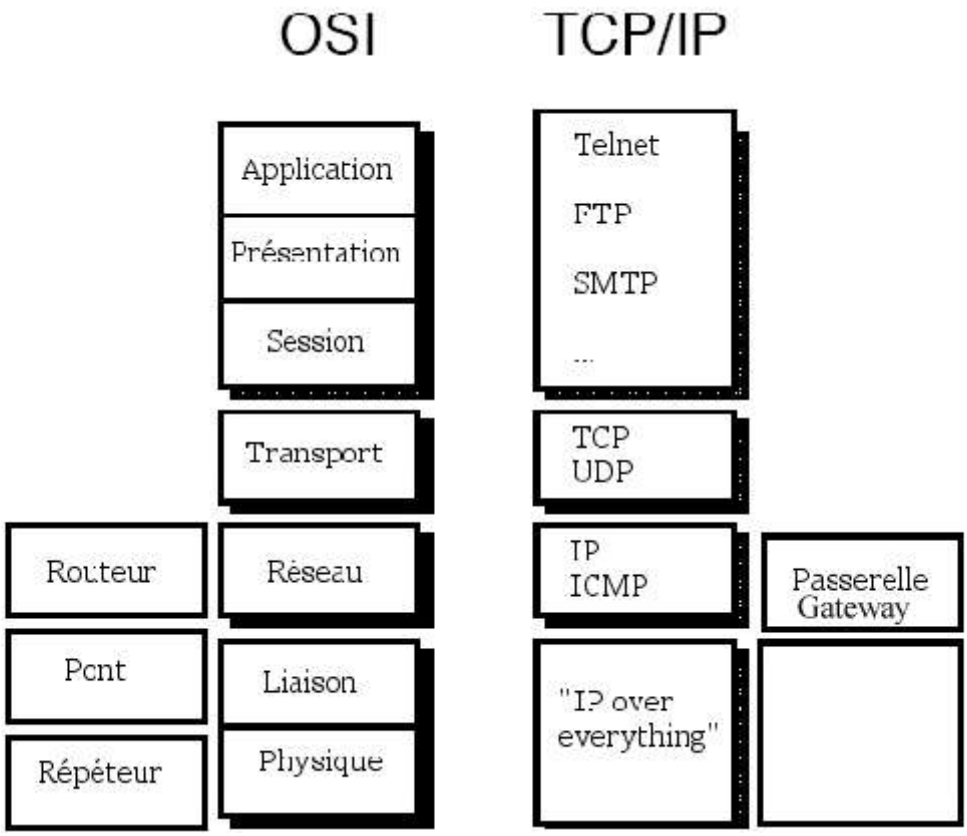
1. Rappel : Protocole IP (Internet Protocol)

IP est un protocole réseau à base de paquets, de niveau 3 (niveau réseau) dans la hiérarchie des sept couches OSI (Open System Interconnexion).

Le niveau 3 correspond à la première couche où les adresses utilisées sont logiques et non physiques. IP, utilisé dans le réseau local et sur Internet s'est imposé comme le protocole unificateur.

Le protocole Internet est un élément essentiel de la famille de protocoles TCP/IP. IP n'a ni connexion ni sécurité. Son rôle est d' encapsuler les données et de les transmettre. Il est aussi responsable de l'adressage, qu'il effectue sur la base de l'adresse source et de l'adresse cible.

(figure : les couches du modèle osi et TCP / IP)



Le réseau Internet, auparavant utilisé par les universités, les industries de pointe, et les gouvernements dès le milieu des années 1990, est aujourd'hui indispensable aux entreprises et aux sociétés commerciales. Il est utilisé par un grand nombre d'individus et de systèmes exprimant les uns et les autres des besoins différents. Par exemple : la convergence imminente de l'ordinateur, des réseaux, de l'audiovisuel et de l'industrie des loisirs, de la surveillance, de la vidéo à la demande, du télé-achat ou du commerce électronique, sans compter l'utilisation des particuliers (« surf » sur internet, mails, chat...).

Dans ces circonstances, le protocole IP doit offrir plus de **flexibilité** et d'**efficacité**, **plus d'adresses** et **corriger les problèmes** découverts lors de l'utilisation intensive d'IPv4.

2. IPv6: Objectifs

Les objectifs principaux de ce nouveau protocole sont :

- Supporter des milliards d'adresses.
- Réduire la taille des tables de routage.
- Simplifier le protocole, pour permettre aux routeurs de router les datagrammes plus rapidement.
- Fournir une meilleure sécurité (authentification et confidentialité) que l'actuel protocole IP.
- Accorder plus d'attention au type de service, et notamment aux services associés au trafic temps réel.
- Faciliter la diffusion Multicast en permettant de spécifier le périmètre de diffusion.
- Donner la possibilité à un ordinateur de se déplacer sans changer son adresse.
- Permettre au protocole une évolution future.
- Accorder à l'ancien et au nouveau protocole une coexistence pacifique.

Pour cela, IPv6 maintient certaines fonctions d'IPv4, en écarte, minimise ou affine certaines jugées obsolètes. De nouvelles fonctions sont ajoutées.

Bien qu' IPv6 ne soit pas compatible avec IPv4, il est compatible avec tous les autres protocoles Internet, dont TCP, UDP, ICMP, IGMP, OSPF, BGP et DNS ; parfois, de légères modifications sont requises (notamment pour fonctionner avec de longues adresses). Les « tunnels » permettent de faire coexister les 2 protocoles : on encapsule des paquets IPv6 dans du IPv4, lorsque deux réseaux IPv6 n'ont pour communiquer qu'un « passage » par le monde IPv4.

3. Historique

- Avant 1993, plusieurs propositions circulaient déjà pour améliorer ou compléter le standard Ipv4. Citons :
    - **IPAE** : IP Address Encapsulation, proposition visant à ajouter une couche entre la couche IP et la couche TCP qui contiendrait uniquement une adresse « publique ». Ce projet se positionne comme une amélioration et non un remplacement d'IPv4.
    - **Nimrod** : New Internet Routing and Addressing Architecture : Cette architecture se basait essentiellement sur la source des paquets, qui défini la route à partir d'un plan du réseau (grands axes ...) . Ce projet souhaite favoriser la mobilité et le Multicast.
    - Simple CLNP qui deviendra **TUBA** (TCP UDP over CNLP Addressed Network) : Cette proposition se base sur une **vision très hierarchique des adresses**..
    - **SIPP** (Simple Internet Protocol Plus, basé sur l'avant-projet **IPAE** décrit plus haut et reprenant des propositions des standards **SIP** et **PIP**) propose des **adresses sur 64 bits** pouvant supporter plusieurs niveaux de hiérarchie. L'idée d'adresses identifiant des régions plutôt qu'un noeud précis est introduite, de même qu'un champ **scope** (étendue) pour les adresses multicast. **L'en-tête IPv4 est simplifié** et contient un identificateur de flux afin d'assurer la qualité de service minimale pour certaines applications (real-time...). L'authentification fait aussi partie des préoccupations de SIPP.
    - **CATNIP** (Common Architecture for the Internet, basé sur TP/IX) propose des adresses à longueur variable et intègre dans un en-tête relativement court les informations nécessaires à TP/IX, Novell IPX et CLNP (Connectionless networking protocol), trois protocoles sur lesquels le groupe a basé son travail.

C'est en 1993 que la RFC 1550 lance un appel afin de définir le cahier des charges pour un nouveau protocole internet : **IP Next Generation (Ipng) White Paper Solicitation**.

Cet appel insiste sur plusieurs points : Il faut essayer de définir l'étendue de la sollicitation du réseau (quel usage, quelle quantité de donnée), et l'échelle de temps (quel temps de réponse acceptable ?). Se posent alors les questions du niveau de sécurité souhaitable, de la politique de routage... La RFC s'interroge également sur l'importance future des hôtes mobiles, sur la nature des médias qui seront utilisés (onde, fibre, câble ... maintient-on le « IP over everything »?), sur les moyens d'administrer le réseau, de

réserver des ressources, sur la tolérance aux erreurs... Un extrait de cette RFC est présent en annexe 2.1 .

Trois candidats proposent alors leur standard : CATNIP, SIPP et TUBA.

- La direction du groupe **IPng** (next generation) décide qu'aucun des trois standard proposé ne satisfait totalement les demandes du livre blanc. Cependant, le standard **SIPP** s'en approche le plus, et peut servir de base de travail au développement d'IPv6. Quelques modifications sont expressément demandées, comme la mise en place d'adresses sur 128 bits.

- Solution retenue: SIPP (1995).

- Actuellement, le protocole IPv6 est en phase d'expérimentation. Certaines normes peuvent encore évoluer.

1. Brève présentation d'IP v6

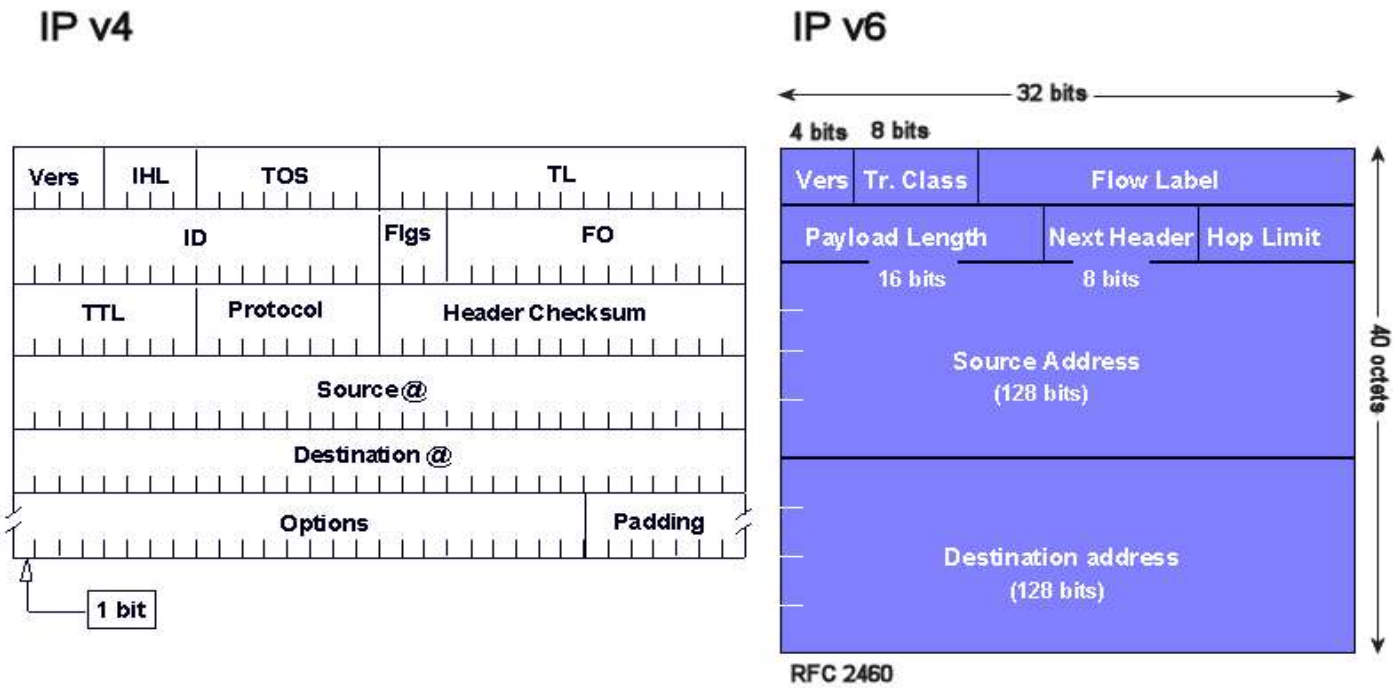
IPv6 utilise des adresses plus longues qu'IPv4. Elles sont codées sur 16 octets et permettent 2^128 adresses (3,4.10^38). Cela équivaut à 6 x 10^23 adresses par mêtre carré.

La caractéristique majeure d'IPv6 est la **simplification de l'en-tête des datagrammes**. L'en-tête du datagramme de base IPv6 ne comprend que 7 champs (contre 13 pour IPv4). Certains champs obligatoires de l'ancienne version sont maintenant devenus optionnels. De plus, la façon dont les options sont représentées est différente : elle permet aux routeurs d'ignorer plus simplement les options qui ne leur sont pas destinées. Ces deux dernières fonctions accélèrent le temps de traitement des datagrammes par les routeurs.

D'autre part IPv6 apporte une plus grande **sécurité** : l'authentification et la confidentialité constituent les fonctions de sécurité majeures du protocole IPv6.

L'auto-configuration a également été prévue, et peut se faire grâce aux routeurs (notion de préfixe) : cette fonction permettra de remplacer les actuels serveurs DHCP.

4. Comparaison des en-têtes IP v4 et IP v6



Premières observations :

- Le champ **Header Length** (IHL) est supprimé.
- **ToS** (Type of Service) devient **Flow Label (identificateur de flux).**
- **Total Length** (TL) devient **Payload Length (longueur des données utiles).**
- **ID, Flags** et **Fragment Offset** sont supprimés.
- **TTL** devient **hop limit (Nombre de sauts).**
- **Protocol** devient **Next Header** (**En-tête suivant** : Les valeurs de ce champ peuvent être identiques à celles d'IPv4 mais peuvent contenir également des informations plus précises : remplacement du champ options d'IPv4)
- **Le Header CS** est supprimé, son temps de calcul faisait baisser les performances, alors que les couches liaisons de données peuvent remplir ce rôle.
- Les adresses passent de 32 à 128 bits
- L'alignement passe de 32 à 64 bits

Tous les ordinateurs et routeurs conformes à IPv6 doivent supporter les datagrammes de 576 octets. Cette règle donne à la fragmentation un rôle secondaire. De plus, quand un

ordinateur envoie un trop grand datagramme IPv6, contrairement à ce qu'il se passe avec la fragmentation d'IPv4, le routeur qui ne peut le transmettre retourne un message d'erreur à la source. Ce message précise à l'ordinateur source d'interrompre l'envoi de nouveaux datagrammes de cette taille vers la destination. Avoir un ordinateur qui transmet immédiatement des datagrammes à la bonne dimension est bien plus efficace que de d'avoir des routeurs qui les fragmentent à la volée.

5. Détail des champs de l'en-tête IPv6 :

- Le champ **Version** est toujours égal à 4 bits pour IPv6. Pendant la période de transition de IPv4 vers IPv6, les routeurs devront examiner ce champ pour savoir quel type de datagramme ils routent.

- Le champ **Classe de trafic** (codé sur 8 bits) est utilisé pour distinguer les sources qui doivent bénéficier du contrôle de flux des autres. Des priorités de 0 à 7 sont affectées aux sources capables de ralentir leur débit en cas de congestion. Les valeurs 8 à 15 sont assignées au trafic temps réel (les données audio et vidéo en font partie) dont le débit doit être garanti. Cette distinction des flux permet aux routeurs de mieux réagir en cas de congestion. Dans chaque groupe prioritaire, le niveau de priorité le plus faible correspond aux datagrammes les moins importants.

- Le champ **Identificateur de flux** contient un numéro unique choisi par la source qui a pour but de faciliter le travail des routeurs et de permettre la mise en oeuvre les fonctions de **qualité de service** comme RSVP (Resource reSerVation setup Protocol). Cet indicateur peut être considéré comme une marque de contexte pour le routeur. Le routeur peut alors faire un traitement particulier : choix d'une route, traitement en "temps-réel" de l'information, ... Le champ identificateur de flux peut être rempli avec une valeur aléatoire qui servira à référencer le contexte. La source gardera cette valeur pour tous les paquets qu'elle émettra pour cette application et cette destination. Le traitement est optimisé puisque le routeur n'a plus à consulter que cinq champs pour déterminer l'appartenance d'un paquet. De plus, si une extension de confidentialité est utilisée, les informations concernant les numéros de port sont masquées aux routeurs intermédiaires.

- Le champ **Longueur des données utiles** (en anglais payload) sur deux octets, ne contient que la taille des données utiles, sans prendre en compte la longueur de l'en-tête. Pour des paquets dont la taille des données serait supérieure à 65535 ce champ vaut 0 et l'option jumbogramme de l'extension de "proche en proche" est utilisée.

- Le champ **En-tête suivant** a une fonction similaire au champ protocole du paquet IPv4: Il identifie tout simplement le prochain en-tête (dans le même datagramme IPv6). Il peut s'agir d'un protocole (de niveau supérieur ICMP, UDP, TCP, ...) ou d'une extension.

- Le champ **Nombre de sauts** remplace le champ **TTL** (Time-to-Live) en IPv4. Sa valeur (sur 8 bits) est décrémentée à chaque noeud traversé. Si cette valeur atteint 0 alors que le paquet IPv6 traverse un routeur, il sera rejeté avec l'émission d'un message ICMPv6 d'erreur. Il est utilisé pour empêcher les datagrammes de circuler indéfiniment. Il joue le même rôle que le champ **Durée de vie** d'IPv4, à savoir qu'il contient une valeur représentant le nombre de sauts ou de pas (hops) qui est décrémenté à chaque passage dans un routeur.

- Viennent ensuite les champs **Adresse source** et **Adresse de destination**. Après de nombreuses discussions, il fut décidé que les adresses de longueur fixe égales à 16 octets constituaient le meilleur compromis.

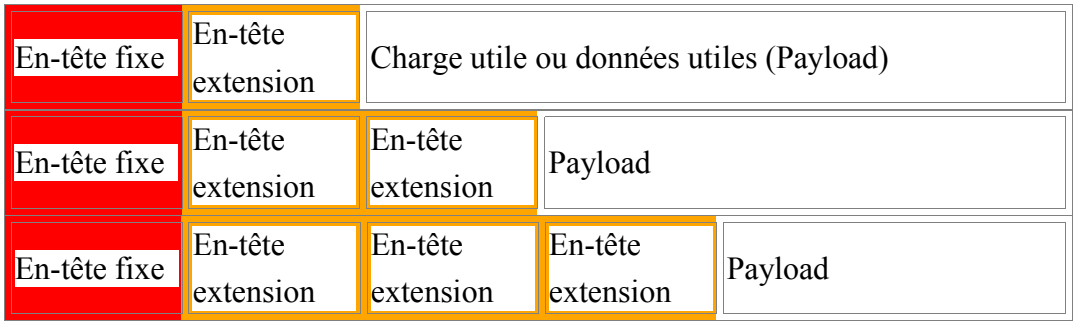
6. En-tête optionnels (ou en-tête d'extension)

Cet en-tête fournit une information complémentaire optionnelle. Si plus d'un en-tête est présent, ils doivent apparaître immédiatement après l'en-tête fixe, de préférence dans l'ordre de la liste. Ces en-têtes optionnels sont identifiés par le champ **Next Header**.

Certains en-têtes ont un format fixe, d'autres contiennent un nombre variable de champs variables. Pour cela, chaque item est codé sous forme d'un triplet (Type, Longueur, Valeur). Le **Type** est un champ d'un octet qui précise la nature de l'option. Les différents **Types** ont été choisis de façon à ce que les 2 premiers bits disent aux routeurs qui ne savent pas exécuter l'option ce qu'ils doivent faire.
Les choix pour ces routeurs sont alors :

- sauter l'option,
- détruire le datagramme,
- détruire le datagramme et retourner un message ICMP à la source,
- détruire le datagramme sans retourner de message ICMP s'il s'agit d'un datagramme multidestinataire : Cela permet en effet que la source d'un datagramme multidestinataire ne soit pas inondée de messages de retour.

Les en-têtes ont donc une longueur multiple de 8 octets, on conserve l'alignement à 64 bits.



Ces en-têtes optionnels peuvent être :

- des en-têtes des protocoles de niveau supérieur
  - TCP (6).
  - UDP (17).
  - ICMP (58).

- des en-têtes propres à IPv6
  - Pas après pas / Hop-by-Hop (0).
  - Routage / Routing (43) .
  - Fragmentation / Fragment (44).
  - Options de destination /Destination options (60).
  - Authentification (51) .
  - Données sécurisées / Encapsulating Security Payload (50).
  - Pas d'en-tête / No header (59) .

Décrivons les en-têtes propres à IPv6 :

- L'en-tête **Pas-après-pas** (hop by hop) contient des informations destinées à tous les routeurs sur le chemin.

- L'en-tête **Routage** (routing) donne la liste d'un ou de plusieurs routeurs qui doivent être visités sur le trajet vers la destination. Deux formes de routage sont mises en oeuvre de façon combinée : le routage strict (la route intégrale est définie) et le routage lâche (seuls les routeurs obligatoires sont définis). Les 4 premiers champs de l'en-tête d'extension **Routage** contiennent 4 entiers d'un octet :
  - le type d'en-tête suivant,
  - le type de routage (couramment 0),
  - le nombre d'adresses présentes dans l'en-tête (1 à 24),
  - une adresse donnant la prochaine adresse à visiter.

  Le dernier champ commence à la valeur 0, il est incrémenté à chaque adresse visitée. Quand le routage est ainsi défini on parle de Source routing.

- L'en-tête **Fragmentation** (fragment) traite de la fragmentation de manière similaire à IPv4. L'en-tête contient l'identifiant de datagramme, le numéro de fragment et un bit précisant si d'autres fragments suivent. Dans IPv6, contrairement à IPv4, seul l'ordinateur source peut fragmenter le datagramme. Les routeurs sur le trajet ne le peuvent pas. Cela permet à l'ordinateur source de fragmenter le datagramme en morceaux et d'utiliser l'en-tête **Fragmentation** pour transmettre les morceaux.

- L'en-tête **Authentification** fournit un mécanisme permettant au destinataire d'un datagramme de s'assurer de l'identité de la source.

- L'en-tête **Option de destination** (destination options) est utilisé pour des champs qui n'ont besoin d'être interprétés et compris que par l'ordinateur destinataire. Dans la version originale d'IPv6, la seule option de destination qui a été définie est l'option nulle. Elle permet de compléter cet en-tête par des 0 pour obtenir un multiple de 8 octets. Cet en-tête ne sera pas utilisé dans un premier temps. Il a été défini pour s'assurer que les nouveaux logiciels de routage pourront le prendre en compte, au cas où quelqu'un envisagerait un jour une option de destination.

- L'en-tête **Données sécurisées** (Encapsulating Security Payload ou ESP) : La transformation ESP permet d'assurer la confidentialité, l'authentification et l'intégrité des datagrammes IP en les chiffrant conformément à ce qui est défini par les Security Association. ESP assure aussi une protection contre le rejeu des paquets, grâce au champ " Sequence Number " de l'entête ESP. Les transformations ESP chiffrent des portions des datagrammes, peuvent encapsuler ces portions dans d'autres datagrammes IP et peuvent effectuer des contrôles d'intégrité via un Integrity Check Value (ICV). Tout comme pour l'authentification, l'entête IP du datagramme original est conservé et les autres champs sont sécurisés et précédés d'une entête ESP .

7.Adressage IPv6

7.1 Représentation de l'adresse

L'adresse peut être représentée des trois façons suivantes :

- **forme préférée** : "x:x:x:x:x:x:x:x" où x représente les valeurs hexadécimales des 8 portions de 16 bits de l'adresse. Exemple:
  5f06:b500:89c2:a100:0000:0800:200a:3ff7
- **forme abrégée** : un groupe de plusieurs champs de 16 bits mis à 0 peut-être remplacé par la notation "::". La séquence "::" ne peut apparaître qu'une seule fois dans une adresse. Exemple d'adresses :
  5f06:b500:89c2:a100::800:200a:3ff7 = 5f06:b500:89c2:a100:0:800:200a:3ff7
  ff80::800:200a:3ff7 = ff80:0:0:0:0:800:200a:3ff7
  ::1 = 0:0:0:0:0:0:0:1
- **forme mixée** : lorsqu'on est dans un environnement IPv4 et IPv6, il est possible d'utiliser une représentation textuelle de la forme: "x:x:x:x:x:x:d.d.d.d", où les 'x' sont les valeurs hexadécimales des 6 premiers champs de 16 bits et les 'd' sont les valeurs décimales des 4 derniers champs de 8 bits de l'adresse Exemple d'adresse :
  ::137.194.168.93

  Remarque : puisque IPv6 utilise des adresses hexadécimales, cette dernière adresse IPv4 équivaut à 0:0:0:0:0:0:89C2:A851

7.2 Allocation de l'espace d'adressage

Les premiers bits de l'adresse (le **préfixe**) définissent le type de l'adresse. IPv6 supporte trois types d'adressage : **Unicast** (adresse représentant une seule machine), **Multicast** (adresse utilisée pour la diffusion d'une information à plusieurs machines) et **Anycast** (adresse pour s'adresser à la première machine apte à répondre).
Les adresses commençant par 4 zéros sont réservées aux adresses IPv4 (cela permet de faire cohabiter les deux protocoles, voir l'exemple de la forme mixée ci-dessus).

Au niveau de l'attribution des adresses, on a d'abord pensé effectuer **un découpage géographique grâce aux préfixes**. Cette technique ressemble à celle utilisée pour l'attribution des adresses MAC des machines. L'utilisation de **préfixes** séparés pour les adresses affectées à un fournisseur et les adresses affectées à une zone géographique (Asie, Europe, Amérique du Nord) constituaient un compromis entre deux visions différentes de l'évolution du réseau Internet. Cette idée, développée dans la RFC 2073 a été abandonnée.

La RFC 2928 (Initial IPv6 Sub-TLA ID Assignments) défini les préfixes attribués au plus haut niveau actuellement. Les préfixes sont définis selon les autorités qui les distribuent.

Binary Value	IPv6 Prefix Range	Assignment
0000 000X XXXX X	2001:0000::/29 – 2001:01F8::/29	IANA
0000 001X XXXX X	2001:0200::/29 - 2001:03F8::/29	APNIC
0000 010X XXXX X	2001:0400::/29 - 2001:05F8::/29	ARIN
0000 011X XXXX X	2001:0600::/29 - 2001:07F8::/29	RIPE NCC
0000 100X XXXX X	2001:0800::/29 - 2001:09F8::/29	(future assignment)
0000 101X XXXX X	2001:0A00::/29 - 2001:0BF8::/29	(future assignment)
0000 110X XXXX X	2001:0C00::/29 - 2001:0DF8::/29	(future assignment)
0000 111X XXXX X	2001:0E00::/29 - 2001:0FF8::/29	(future assignment)
0001 000X XXXX X	2001:1000::/29 - 2001:11F8::/29	(future assignment)
. . .		
. . .		
1111 111X XXXX X	2001:FE00::/29 - 2001:FFF8::/29	(future assignment)

L'IANA alloue les adresses au RIR (Regional Internet Registry) chargés chacun d'une partie du monde. Nous voyons sur ce tableau l'APNIC (Asia Pacific Network Information Centre), l'ARIN (American Registry for Internet Number, chargé de l'Amérique du Nord et de l'Afrique - partie au sud du sahara), la RIPE NCC (Réseau IP Européens chargés également de l'Afrique du Nord).

Par exemple, le préfixe de RENATER (Réseau de l'Enseignement et de la Recherche en France) est 2001:0660 /32 .
Le préfixe de l'Insa de Rouen, raccordé à RENATER est 2001:660:7403::/48 .
Le préfixe de l'Université de Caen est 2001:660:7101:: /48.

On remarque que ces préfixes appartiennent aux classes attribuées par le RIPE NCC, responsable de l'attribution des adresses pour l'Europe.

IPv6 peut gérer 2 sortes d'adresses : publiques et locales.

Les adresses de **liens** et de **sites** locaux n'ont qu'une spécification locale. Elles peuvent être réutilisées par d'autres organisations sans qu'il y ait de conflit. Elles ne peuvent pas être propagées hors des limites des organisations, ce qui les rend bien adaptées à celles qui utilisent des gardes-barrières pour protéger leur réseau privé du réseau Internet car elles sont invisibles de l'extérieur. Ces adresses sont équivalentes aux adresses privées déjà utilisées en IPv4. Notons que l'adresse **Lien Local** est présente dès l'installation d'IPv6 sur une machine.

Le type d'adresse IPv6 est indiqué par les premiers bits de l'adresse qui sont appelés le **Préfixe de Format** (Format Prefix).

- L'allocation initiale de ces préfixes est la suivante:

Allocation	Préfixe	Fraction de l'espace
Adresses "Link Local Use"	1111 1110 10	1/1024
Adresses "Site Local Use"	1111 1110 11	1/1024
Adresses Multicast	1111 1111	1/256

15 % de l'espace d'adressage est actuellement alloué. Les 85% restants sont réservés pour des usages futurs .

Ces normes, ainsi que les adresses **Loopback** et **Unspecified** sont définies par la RFC 3513.

Address type	Binary prefix	IPv6 notation
Unspecified	00...0 (128 bits)	::/128
Loopback	00...1 (128 bits)	::1/128
Multicast	11111111	FF00::/8
Link-local Unicast	1111111010	FE80::/10
Site-local Unicast	1111111011	FEC0::/10
Global Unicast	(everything else)	

L'annexe 1 schématise le principe d' attribution des adresses.

8. Adresse Unicast

Suivant le rôle qu'elle va jouer, une adresse **Unicast IPv6** peut être décomposée en plusieurs parties, chaque partie définissant un niveau de hiérarchie différent.

- Au minimum, l'adresse peut avoir une forme non structurée:

128 bits
Adresse de l'hôte (Interface ID)

- On peut utiliser un préfixe de sous-réseau afin d'affiner l'authentification de l'adresse.

n bits	128-n bits
Subnet prefix	Interface ID

- L'utilisation de plusieurs préfixes permettent une authentification plus précise.

n bits	80-n bits	48 bits
Subscriber prefix	Subnet prefix	Interface ID

- On peut encore affiner le modèle structurel d'une organisation en rajoutant un niveau de hiérarchie supplémentaire

s bits	n bits	m bits	128-s-n-m bits
Subscriber prefix	Area prefix	Subnet prefix	Interface ID

Ce modèle hiérarchique a un impact très important sur le routage, car il permet de faire de l'agrégation d'adresse suivant la même hiérarchie. L'utilisation de la hiérarchie au sein des adresses Unicast simplifie donc grandement la structure du réseau et réduit les tables de routage en permettant une agrégation aisée des différentes adresses d'une même organisation.

Notons que lors d'une utilisation de la configuration automatique, l' Interface ID est générée à partir de l'adresse MAC de la machine.

9. Adresse Multicast

Les adresses de diffusion multidestinataire disposent d'un champ **Drapeau** (Flag en anglais - 4 bits) et d'un champ **Envergure** (Scope - 4 bits) à la suite du préfixe, puis d'un champ **Identificateur de groupe** (Group ID - 112 bits). L'un des bits du drapeau distingue les groupes permanents des groupes transitoires. Le champ **Envergure** permet une diffusion limitée sur une zone.

Plus précisément, une adresse **IPv6 Multicast** identifie une adresse de groupe :

- Un hôte peut appartenir à autant de groupes qu'il le souhaite.
- Les adresses Multicast ont le format suivant :

8 bits	4 bits	4 bits	112 bits
1111 1111	flag	scope	Group ID

- 1111 1111 (FF) : identifie l'adresse comme étant Multicast.
- flag 0000 : assigné de façon permanente (Well Known).
- flag 0001 : assigné de façon provisoire (Transcient).
- scope : valeur qui définit l'étendue d'un groupe Multicast.

Tableau des valeurs « scope »

0: reserved	4: (unassigned)	8: organization-local	C: (unassigned)
1: node-local	5: site-local	9: (unassigned)	D: (unassigned)
2: link-local	6: (unassigned)	A: (unassigned)	E: global
3: (unassigned)	7: (unassigned)	B: (unassigned)	F: reserved

Par exemple, lorsqu'une diffusion Multicast est effectuée avec une valeur Scope égale à 2, la diffusion ne s'effectue que sur le lien local. Lorsque cette valeur est égale à 8, le flux ne franchit pas les limites de l'entreprise ou de l'organisation. La diffusion ne se fait donc pas par le réseau Internet mondial.

- group ID: identifie un groupe Multicast au sein de l'étendue spécifiée.

Actuellement, quelques adresses Multicast sont définies par défaut :

Adresse	Appellation	Descriptif
FF01::1	All nodes	groupe de tous les hôtes au niveau node-local
FF02::1	All nodes	groupe de tous les hôtes au niveau link-local
FF01::2	All routers	groupe de tous les routeurs au niveau node-local
FF02::2	All routers	groupe de tous les routeurs au niveau link-local
FF02::1:0	DHCP server	groupe de tous les serveurs DHCP au niveau link-local
FF02::1:XXXX:XXXX	Solicited-Node	Adresse formée à partir de l'adresse Unicast ou Anycast d'un hôte

La RFC 2373 décrit comment d'autres adresses Multicast peuvent être mises en place :

> The current approach [ETHER] to map IPv6 Multicast addresses into IEEE 802 MAC addresses takes the low order 32 bits of the IPv6 Multicast address and uses it to create a MAC address. Note that Token Ring networks are handled differently. This is defined in [TOKEN]. Group ID's less than or equal to 32 bits will generate unique MAC addresses. Due to this new IPv6 Multicast addresses should be assigned so that the group identifier is always in the low order 32 bits as shown in the following:
>
> | 8 | 4 | 4 | 80 bits | 32 bits |
> |---|---|---|---|---|
> | 11111111 | flgs | scop | reserved must be zero | group ID |
>
> While this limits the number of permanent IPv6 Multicast groups to 2^32 this is unlikely to be a limitation in the future. If it becomes necessary to exceed this limit in the future Multicast will still work but the processing will be sightly slower.
>
> Additional IPv6 Multicast addresses are defined and registered by the IANA [MASGN].

On remarque dans l'adresse définie par cette RFC : Les huit premiers bits sont égaux à 1 : il s'agit d'une adresse multicast (par convention). Le champ drapeau signale si cette adresse est affectée de façon permanente ou si il s'agit d'une adresse provisoire. Le champ SCOPE montre le périmètre de diffusion.

10. Adresse Anycast

En plus de supporter l'adressage point à point classique (Unicast) et l'adressage de diffusion multidestinataire (Multicast) IPv6 supporte un nouveau type d'adressage de **diffusion au premier vu** (Anycast).

Cette technique est similaire à la diffusion multidestinataire dans le sens où l'adresse de destination est un groupe d'adresses. Plutôt que d'essayer de livrer le datagramme à tous les membres du groupe, on essaie de le livrer à un seul membre du groupe, celui le plus proche ou le plus à même de le recevoir.

Une adresse **anycast** est une adresse qui est assignée à plus d'une interface. Elle appartient à l'espace d'adressage Unicast (espace public ou site local)

- Un paquet envoyé à une adresse anycast est dirigé vers l'interface la plus proche.
- Généralement, le but de cette adresse est de joindre le routeur ou le serveur DNS le plus proche.
- Syntaxiquement une adresse anycast est identique à une adresse Unicast .
- La seule adresse anycast définie actuellement est l'adresse "Subnet-Router", son format est le suivant :

n bits	128-n bits
Subnet prefix	000000000000

L'utilité d'une telle adresse peut être, par exemple, lorsqu'un élément mobile (portable) souhaite communiquer avec un routeur de son réseau d'origine.

11. Adresse Broadcast :

Présente en IPv4, l'adresse Broadcast permettait d'envoyer un message à toutes les machines. Cependant, l'usage a révélé que généralement peu de machines étaient réellement « intéressées » par ce type de message. Un Multicast sélectif semblant être une meilleure solution, l'adresse Broadcast est supprimée.

II.Théorie de la diffusion Multicast

1. Quelques précisions :

Le **Multicast** est la diffusion d'une information entre une source et plusieurs destinataires. Si on souhaite s'adresser à tous les autres membres du réseau local, on parle de **Broadcast**. Cette possibilité a été interdite par IPv6. On peut définir le Multicast comme une diffusion de 1 vers n machines.

Distinguons le **Multicast** du **Multipoint**. Dans le cas du **Multicast**, le nombre de connexions est inférieur au nombre de destinataires. Dans le cas du **multipoint**, le nombre de connexions est égal au nombre de destinataires.

Les participants à une diffusion Multicast constituent un **groupe Multicast** ou **groupe de diffusion**. Une adresse Multicast ne peut être que destinataire. C'est l'adresse de l'ensemble des machines abonnées au **groupe de diffusion**. La source est connue par son adresse **Unicast**. Remarquons que la **source** (l'émetteur du flux) n'est pas nécessairement membre du groupe auquel elle diffuse.

L'intérêt du Multicast est d'économiser les ressources réseau, les ressources des routeurs, de la source et la bande passante (les données ne circulent qu'une seule fois sur le même lien). Cela permet de nouvelles applications telle la diffusion de vidéo à la demande, télésurveillance, télévision mais également diffusion et mise à jour de logiciels, jeux, diffusion des informations liées à la bourse etc...

On peut résumer une session Multicast ainsi :

- Les **émetteurs** (source) et **récepteurs** (membres) sont distincts.
- Les **hôtes** disent aux routeurs de quels **groupes** ils sont membres.
- Ils ne reçoivent que les informations destinées à ce groupe.
- Ils ne disent rien sur les groupes vers lesquels ils émettent.
- Les routeurs doivent écouter toutes les adresses Multicast pour transmettre les datagrammes qui leur sont demandés.
- Les routeurs utilisent des protocoles de routage appropriés.

2. Equipements nécessaires au Multicast.

L'ensemble des composants réseau (routeurs, cartes réseau des machines) doit supporter les fonctionnalités Multicast (**IGMP**), de même que les systèmes d'exploitation des équipements et des machines. On sait que par défaut, la carte Ethernet écoute l'adresse Broadcast et son adresse propre, fixée en PROM. Le driver doit donc explicitement lui indiquer d'écouter les adresses Multicast.

Pour information, le Multicast est supporté en natif par Linux, FreeBSD, Windows à partir des versions 9x et NT. Il semblerait que des problèmes apparaissent sous Mac OS pour les versions antérieures à Mac OS X.2 .

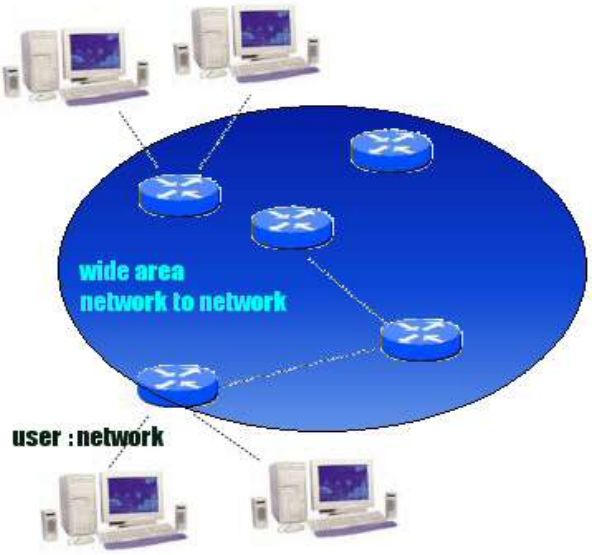
Certains équipements de niveau 2 ne savent pas traiter les trames Multicast. D'autres les traitent comme du Broadcast. Citons les protocoles de routage spécifiques **IGMP** pour les équipements les plus évolués, ainsi que les protocoles propriétaires **CGMP** et **RGMP** (Cisco).

Les équipements de niveau 3 récents implémentent souvent **DVRMP** (version 3.9 minimum), **PIM**, **SM**, parfois plus. Sous Cisco, il est recommandé d'utiliser un IOS de version supérieure à 12.1. La topologie Multicast peut être native (de loin la plus simple à administrer) ou non-native. On encapsule alors les datagrammes dans des tunnels Unicast. Cela peut poser des problèmes au niveau du routage.

3. Protocoles de routage Multicast

On peut diviser les protocoles Multicast en deux types :

- Les protocoles utilisés en « **local area** » (**IGMP**) ou **USER/NETWORK** (Ces protocoles se chargent de la distribution des informations à l'utilisateur final).
- Les protocoles « **Wide area** » (**DVMRP**, **MOSPF**, **PIM...**) ou **NETWORK/NETWORK**, de réseau à réseau. Ces protocoles assurent la transmission de l'information entre plusieurs routeurs, par exemple.



3.1 Local Area : De l'utilisateur au réseau avec Internet Group Management Protocol (IGMP)

Intégré à IP, il existe trois versions d'**IGMP** (plus une version 0 utilisée lors des tests de mise en place de la première version. L'utilisation d'IGMP v0 est bien sur maintenant fortement déconseillée) :

- version 1 publiée en 1989 (RFC 1112) : version n'utilisant que le **membership report** et le **membership query** – voir plus loin.
- version 2 en 1997 (RFC 2236) : cette version implémente le **leave** et **membership report** v2.
- version 3 en 2002 (RFC 3376), qui inclut des notions de filtrage (inscription à un groupe multicast avec une ou plusieurs adresses source spécifique).

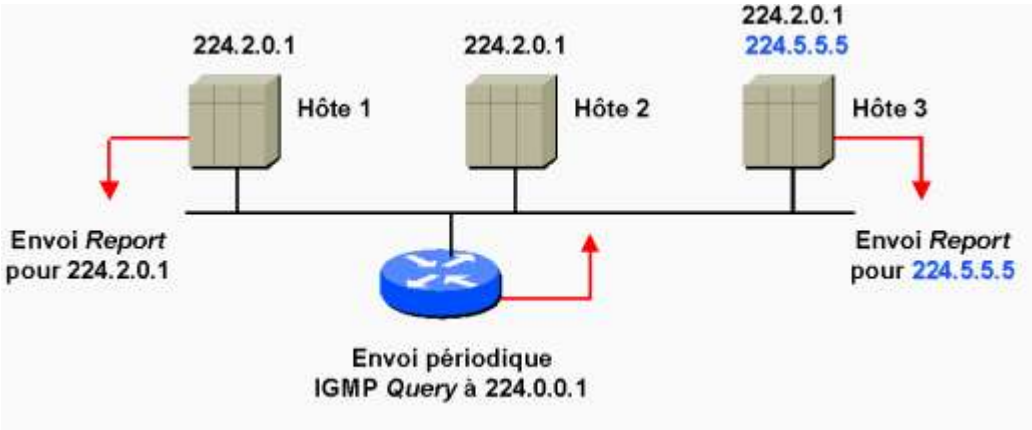
Chaque version est compatible avec la version ultérieure. Il semble qu'une version 4 soit à l'étude (pas de RFC trouvée sur le site de l'IETF pour l'instant).

IGMP est chargé de la distribution des paquets Multicast aux hôtes membres du groupe sur le **LAN**. On parle de « last hop » ou dernier saut du paquet. Il permet aux hôtes de s'abonner ou se désabonner à un groupe en demandant au routeur une copie des paquets du groupe.

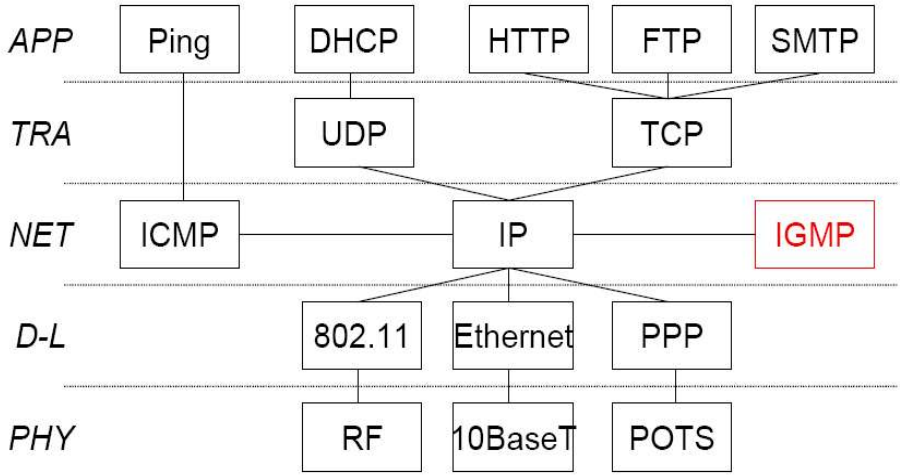
Quelques principes fondateurs d'IGMP (basés sur la version 2) **:**

- Le routeur envoie régulièrement des Membership Queries : il interroge les hôtes Multicast afin de savoir de quels groupes ils sont membres. Cette demande se fait toutes les 60 ou 120 secondes.
- L'hôte répond au routeur avec un Membership Report.
- Tout hôte membre d'un groupe Multicast écoute les Membership Reports envoyés par les autres hôtes :
  - Si un autre hôte envoie un Membership Report pour un groupe auquel j'appartiens, je n'envoie pas de Membership Report pour ce groupe.
  - Cela économise les ressources réseau.
- Les Membership Reports sont envoyés à l'adresse Multicast du groupe concerné.
- Les Leave Reports sont envoyés par défaut à l'adresse Multicast « tous routeurs ».
- Quand un hôte envoie un Leave Report, le routeur envoie automatiquement un membership Query. Si celui-ci n'obtient aucune réponse, il cesse de transmettre les paquets concernés. Le groupe est réputé « sans abonné local ».

Dans l'exemple suivant, le routeur envoie son IGMP Query. L' Hôte 1, membre du groupe 224.2.0.1 répond. L'hôte 2, membre du même groupe, ne répond pas car il a entendu la réponse du 1. L'hôte 3 est membre de plusieurs groupes. Il ne répond pas pour le 224.2.0.1 mais répond pour le 224.5.5.5

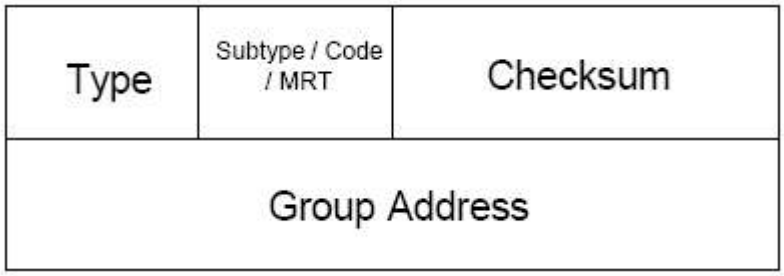


Ce protocole bien qu'encapsulé par IP est du niveau 3. C'est un protocole non fiable et non connecté.



L'en-tête IGMP suit directement l'en-tête IP :

Il est constitué ainsi :



-**Type** indique le type d'information contenue dans le paquet. Il peut être :

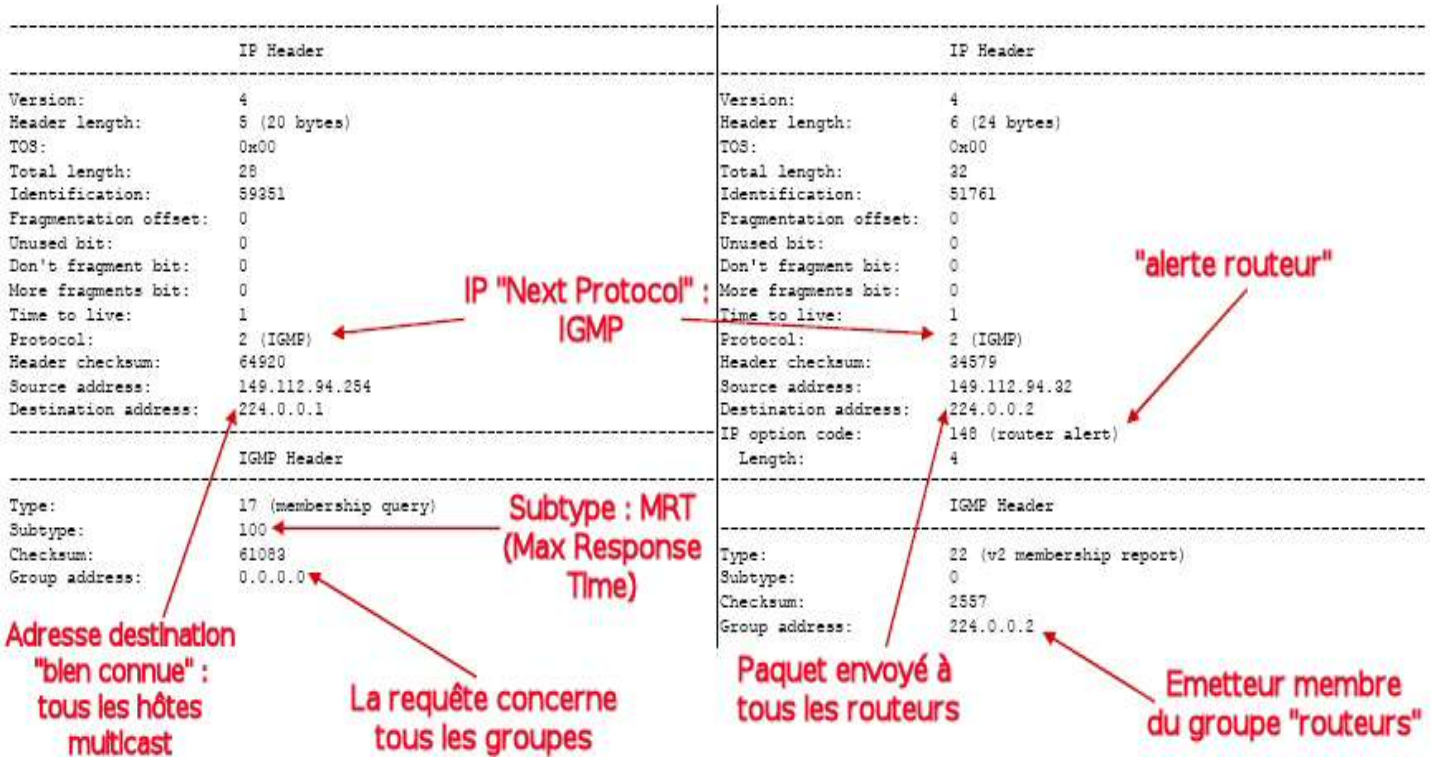
- 0x11 : **Membership Query** : utilisé par les routeurs pour interroger les hôtes Multicast sur leur identité et sur les groupes desquels ils sont membres.
- 0x12 : **Membership Report v1** : utilisés par les hôtes et les routeurs pour rejoindre un groupe (idem pour Membership report v2 et v3).
- 0x13 : **DVMRP** : utilisé dans les communications routeur-à-routeurs DVMRP.
- 0x14 : **PIM v1** : communication routeur-à-routeur PIM.
- 0x16 : **Membership Report v2.**

- 0x17 : **Leave** (quitte un groupe) : utilisé par les routeurs et les hôtes Multicast.
- 0x22 : **Membership Report v3.**

- **Subtype / Code / MRT** dépend de la version et du message. MRT (Maximum Response Time) est utilisé lors de requêtes pour indiquer le temps de réponse maximum.
- **Checksum** : sur 16 bits
- **Group address** : l'adresse du groupe, classe D en IPv4

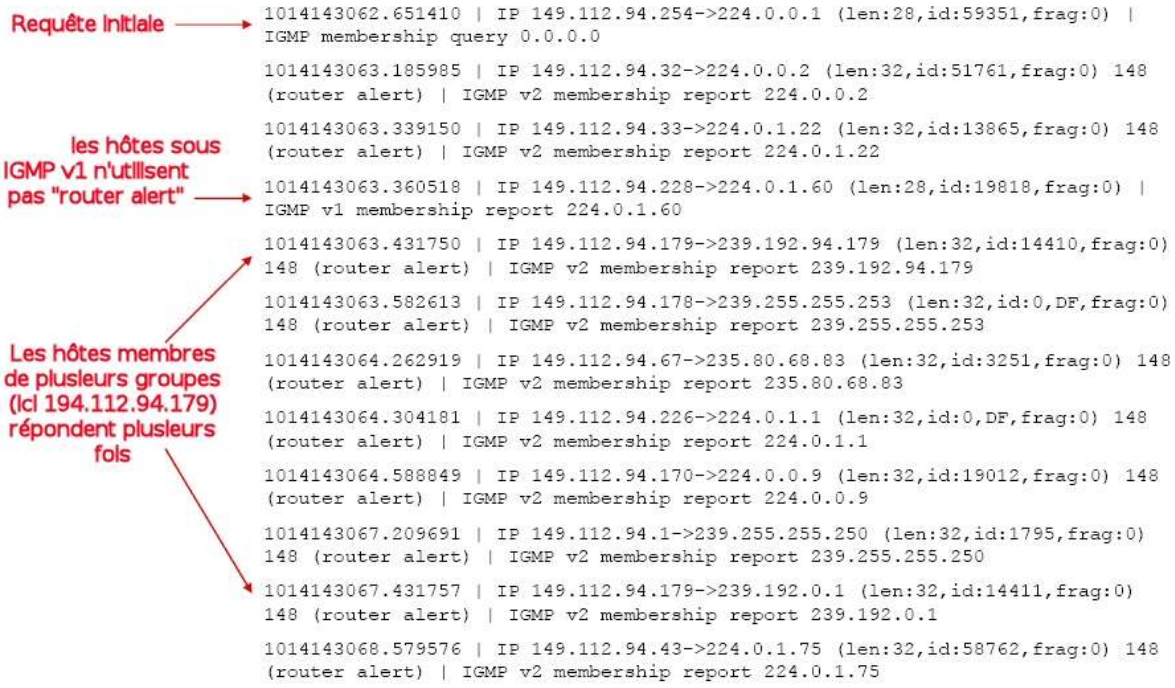
Observons les **en-têtes** de deux paquets émis par un routeur.

A gauche, un **Membership Query** .
A droite, le **membership report**.



Remarque : ces exemples sont basés sur IPv4. Le fonctionnement d'IGMP est similaire en IPv6.

Voyons maintenant un extrait du dialogue entre le routeur et les hôtes :



Si le réseau local comprend plusieurs routeurs, un **Designated Router** est élu (en IGMP v1, au hasard. Dans les versions ultérieures, c'est le routeur qui possède la plus petite adresse IP). Le **Designated Router** est le seul à envoyer des **Membership Queries**. Le **Designated Router** n'est pas forcément le « routeur d'entrée » des paquets Multicast. Lorsqu'un **Designated Router** reçoit un **Leave**, il envoie un **Query** directionnel vers les hôtes qui ont été abonnés au groupe. Cela réduit les temps de latence dans la cessation de diffusion.

IGMP snooping donne au commutateur des fonctions de niveau 3. Il peut découvrir dynamiquement les sources et autres routeurs Multicast. Il examine le paquet pour savoir s'il s'agit d'un paquet IGMP (rejoindre ou quitter) et interopère avec les commutateurs utilisant CGMP. Ce protocole a pour objectif d'être traité en hardware.

3.2 Wide Area ou network to network, dialogue entre routeurs.

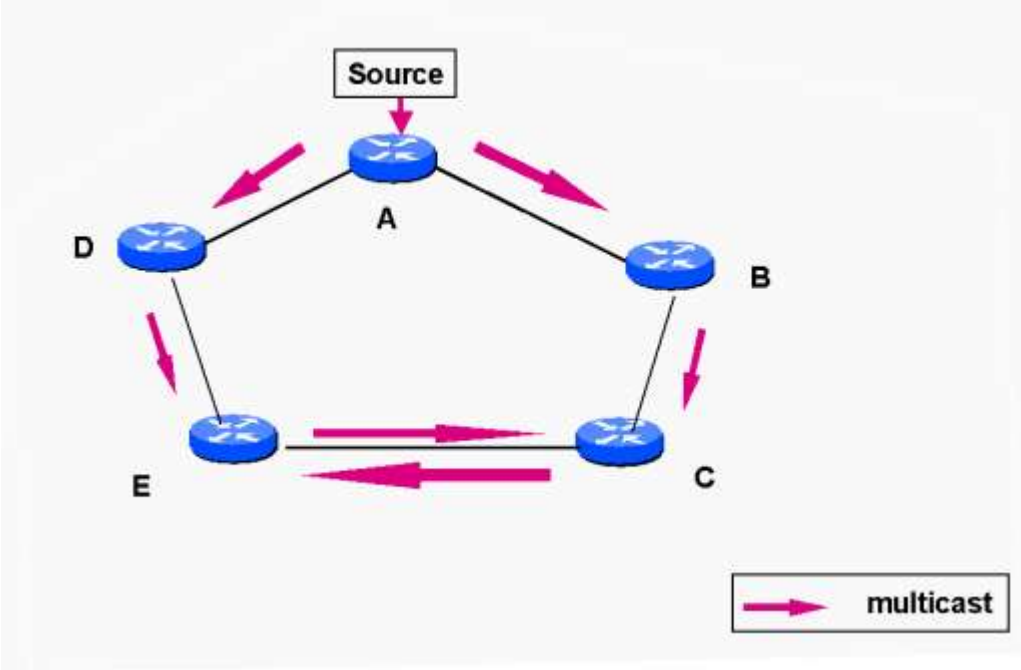
Plusieurs protocoles traitent du routage Multicast sur « grandes distances ». Nous détaillerons ici DVMRP et PIM, en citant ensuite les protocoles remplissant les mêmes rôles.

DVMRP: Distance Vector Multicast Routing Protocol

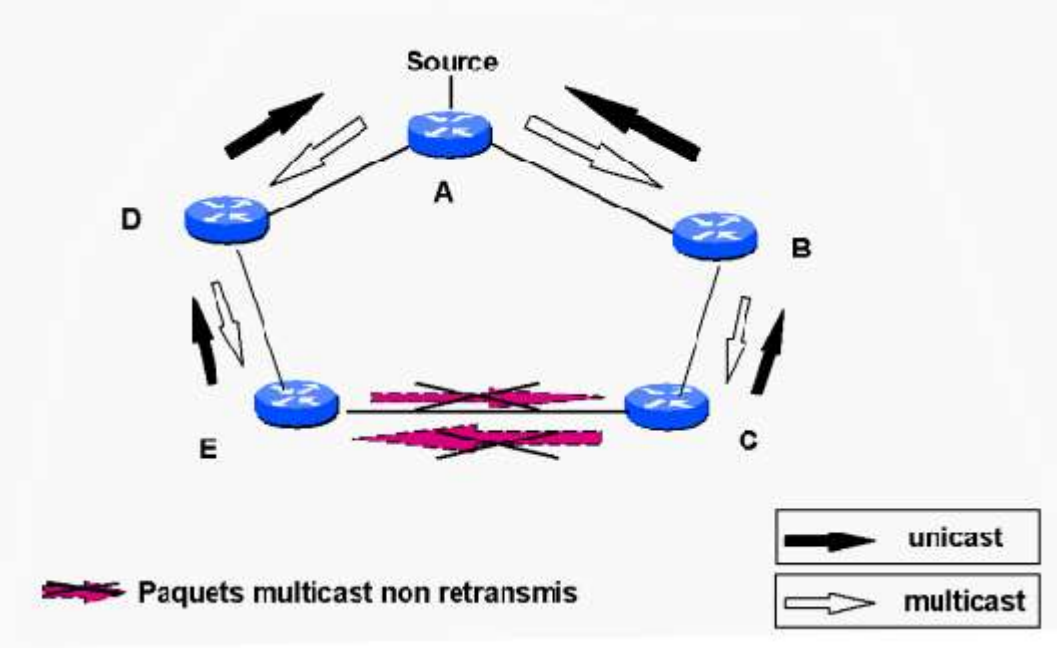
Implémenté dans certains routeurs (pas cisco) ou dans le mrouted des unix (version 3.8 minimum), le DVMRP utilise les techniques de **flooding (inondation)** et **pruning (élagage).**

- Flooding : On « inonde » le réseau Multicast. Les « branches » non intéressées le signalent.
- Elles sont alors élaguées (pruning).
- L'algorithme RPF (Reverse Path Forwarding) est utilisé pour éviter les boucles :
  - le routeur transmet un paquet Multicast si le datagramme est reçu sur l'interface utilisée pour envoyer des paquets Unicast vers la source (Reverse path).
  - Test RPF : Oui – Le paquet est retransmis, on inonde. Non: Le paquet est jeté.
  - Le paquet est retransmis vers toutes les interfaces non élaguées sauf l'interface RPF.

Exemple: inondation sans RPF

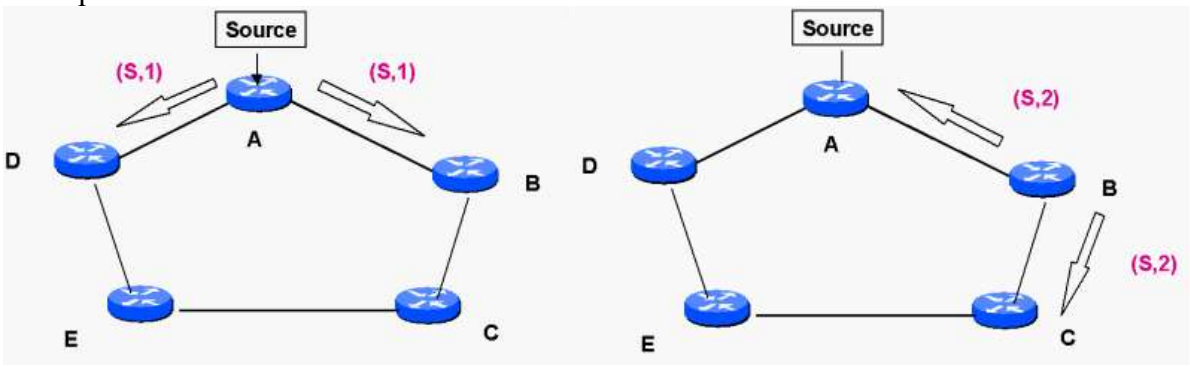


Avec RPF



DVMRP utilise son propre routage Unicast (variante de RIP) pour déterminer son critère RPF, pour localiser les sources Multicast. Les routeurs DVMRP s'échangent leurs tables de routage. Ces tables sont constituées de Destination, Masque et Métrique. Les destinations sont les adresses Unicast des sources, la métrique est le nombre de routeurs Multicast à franchir pour atteindre la source. L'objectif est de construire un arbre minimal de diffusion à partir de la source. Le même protocole de routage est alors utilisé dans tout l'arbre de diffusion.

Exemple :



Le Routeur B décide que le routeur A est en amont de la source vidéo. En théorie, B devrait envoyer des (S,2) vers A et C. En fait, il envoie à A une information de métrique de routage dite Empoisonnée (poison reverse), égale à l'infini (RFC 1112). Conséquence : le routeur B attend de A l'information de la source, A ne compte pas sur B pour cette information. C compte sur B.

Ces échanges entre routeurs se font grâce à IGMP version 3.

PIM : Protocol Independent Multicast (RFC 2362)

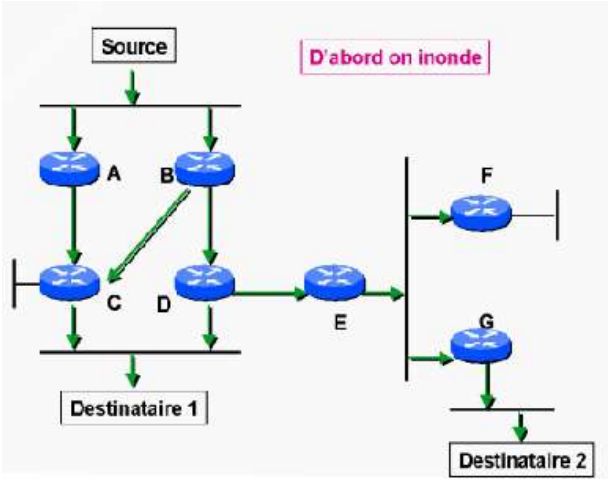
Fonctionne en deux modes, **dense** et **sparse**. Pim se repose sur le protocole Unicast sous-jacent pour le poison reverse et le RPF. Il utilise l'arbre basé sur la source s'il existe, sinon il créée un arbre partagé.

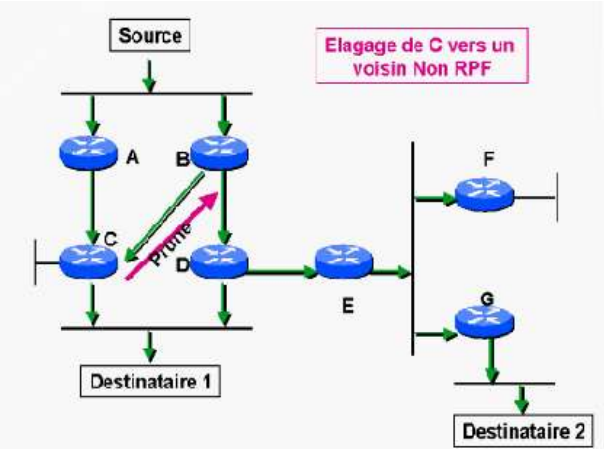
PIM DM – Dense Mode

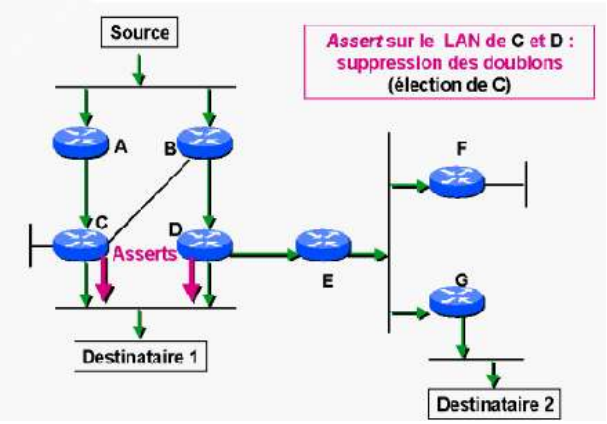
On peut rapprocher ce protocole de DVMRP. Il utilise le flooding (inondation), le pruning (élagage) et les greffes. Les arbres sont construits à partir de la source émettrice en utilisant RPF. Lorsqu'il y a plusieurs routeurs, l'un est élu. Pim DM n'a jamais été standardisé par l'IETF.

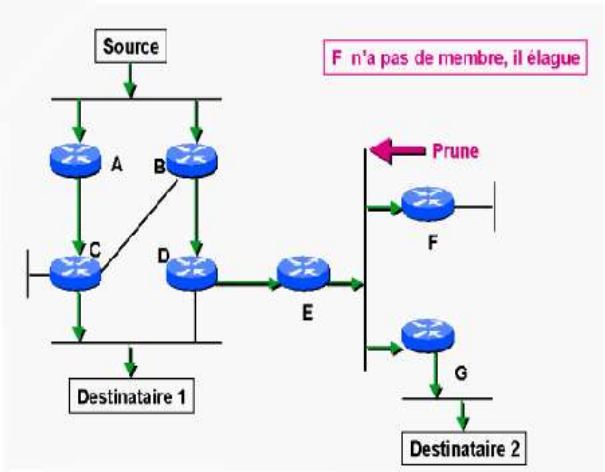
- Elagage : lorsqu'un routeur n'a pas d'hôte ou routeur Multicast en aval, il envoie un **prune packet** à la source émettrice pour ce groupe Multicast via les interfaces RPF. Il arme un Time Out.
- Greffe : A la réception d'une nouvelle demande d'un récepteur local ou d'un routeur en aval pour un nouveau groupe (ou pour un groupe précédemment élagué), le routeur envoie un graft packet vers le routeur RPF pour éviter l'attente d'expiration du timeout d'élagage.

Voyons schématiquement un exemple de diffusion PIM-DM:




Source
Elagage de C vers un voisin Non RPF
A
B
C
Prune
D
E
F
G
Destinataire 1
Destinataire 2


Source
Assert sur le LAN de C et D : suppression des doublons (élection de C)
A
B
C
D
Asserts
E
F
G
Destinataire 1
Destinataire 2

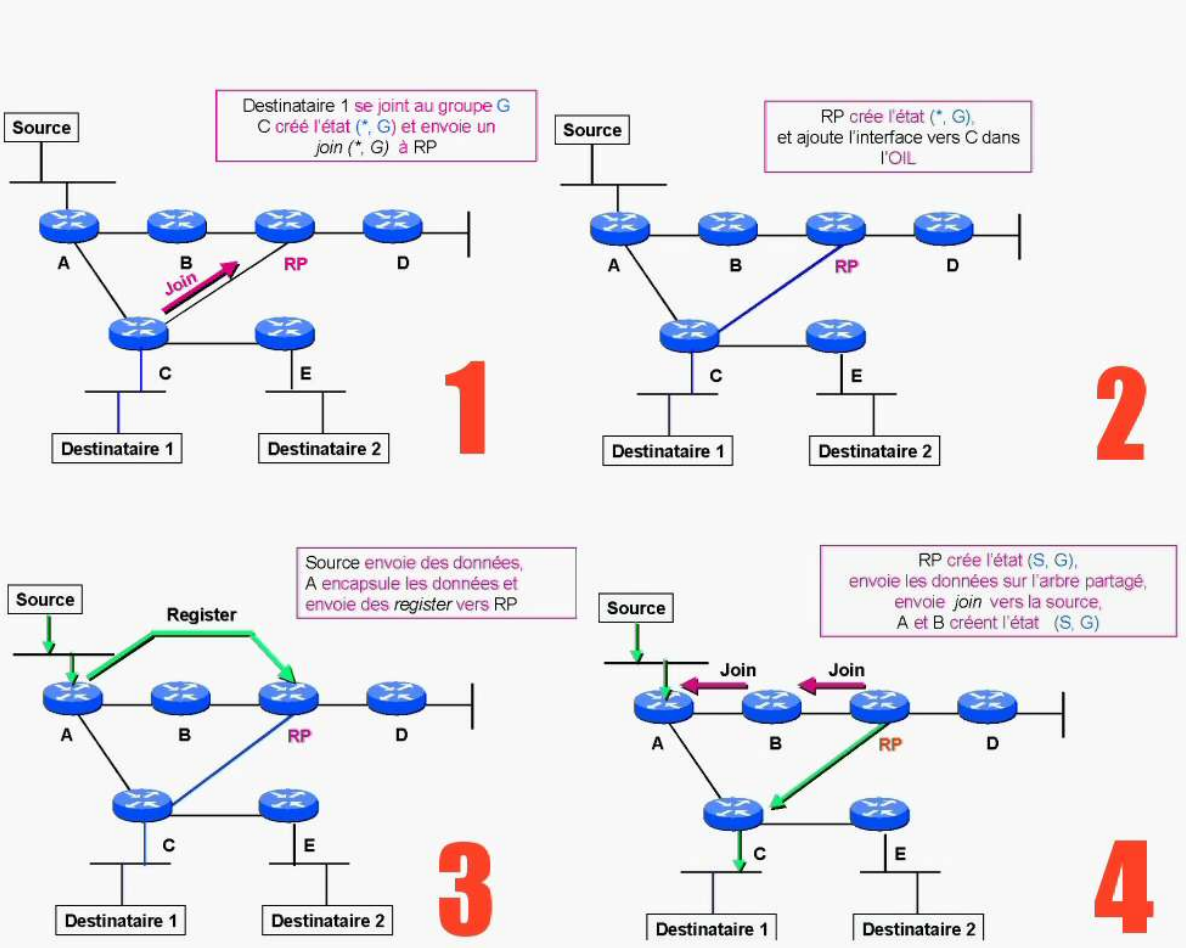

Source
F n'a pas de membre, il élague
A
B
C
D
Prune
E
F
G
Destinataire 1
Destinataire 2

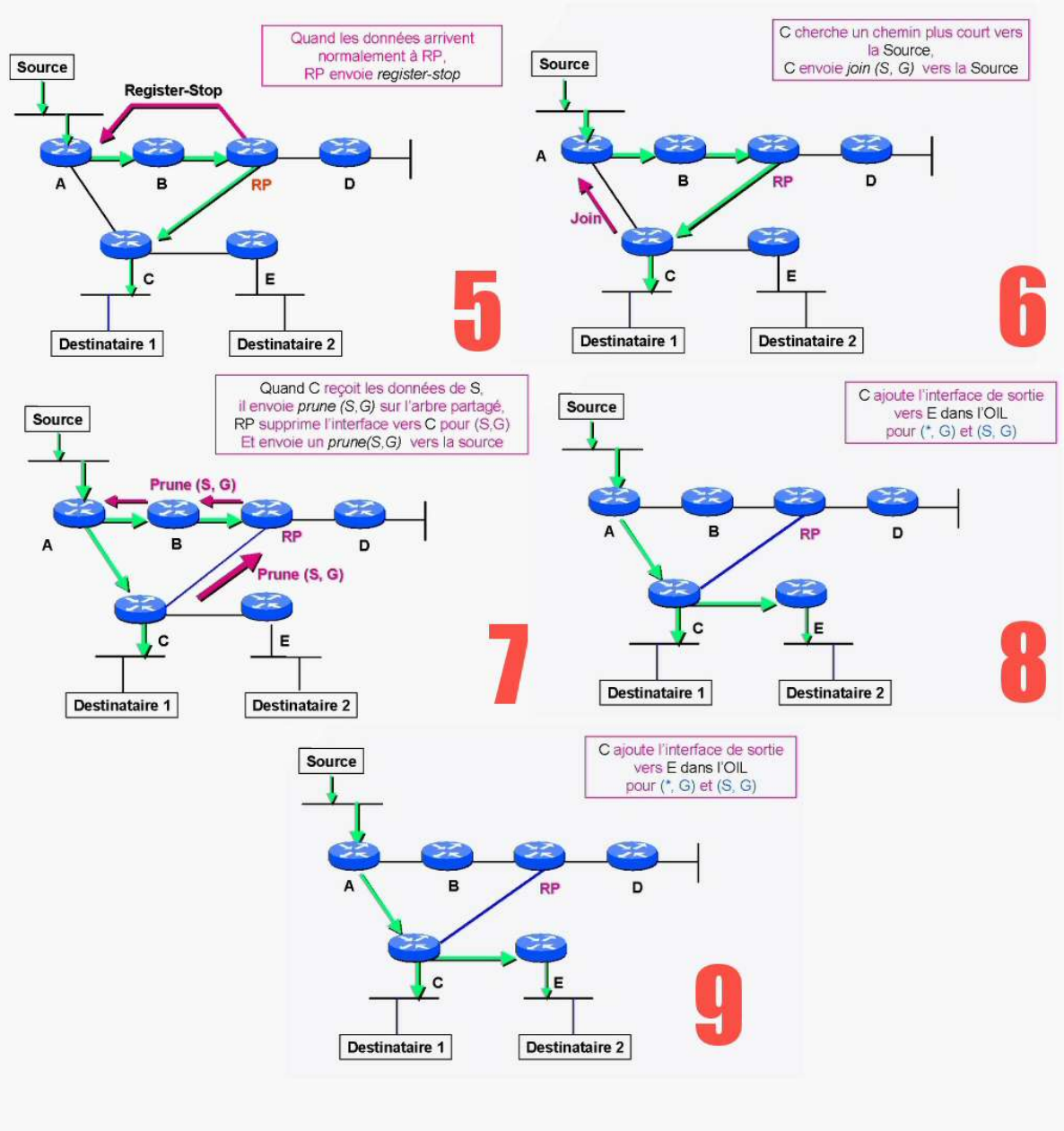
PIM-SM Sparse Mode.

La principale différence avec la méthode Dense Mode est la mise en place d'un abonnement explicite (join) :

- La source s'enregistre auprès d'un point de rendez-vous (RP).
- Le RP est la racine de l'arbre de diffusion Multicast partagé (RPT).
- Le RP est configuré statiquement ou connu dynamiquement par Auto-RP ou candidate RP (PIMv2).
- Pour s'abonner, le destinataire envoie un Join au RP.
- Il peut y avoir plusieurs RP, un groupe n'est enregistré qu'auprès d'un seul RP.

Il n'y a pas d'inondation, le flux Multicast parcourt le RPT ou/puis l'arbre dont la source est l'origine (SPT), les routeurs feuilles peuvent se joindre à l'arbre et les paquets Multicast ne vont que là où c'est utile. On utilise le RP pour tester les interfaces RPF de l'arbre partagé (*,G), on utilise la source pour tester les interfaces RPF de l'arbre basé sur la source (S,G). Les états (S,G) sont préférés aux états (*,G). Voyons un exemple graphiquement.





Quelques autres protocoles, en bref :

Il existe une version **PIM SDM** (Sparse Dense Mode) implémentée uniquement sur Cisco. Elle permet aux routeurs de se comporter en Sparce mode si un RP est connu, sinon en Dense Mode.

PIM SSM (Source spécific Multicast) est une version simplifiée de PIM permettant l'émission de 1 vers n (pas n vers m). Elle repose sur IGMPv3 et abandonne l'idée d'un RP.

Un domaine PIM est composé d'un ou plusieurs RP.

Lors de routages inter-domaines, des protocoles tels MBGP (Multicast Border Gateway Protocol ou Multiprotocol extension for BGP) ou MSDP (Multicast Source Discovery Protocol) sont mis en place.

MBGP est une extension de BGP4, ce protocole défini par la RFC2283 permet de router au sein d'un domaine PIM (iMPGP) ou entre les domaines PIM distincts (eMBGP) en établissant des « peerings » MBGP.

MSDP est un protocole permettant de connaître les sources Multicast. Il s'agit d'échanges entre les RP.

MLD SNOOPING :

Les paquets MLD sont interceptés par le processeur du commutateur ou par un ASIC spécifique. Le commutateur doit examiner le contenu des messages MLD pour déterminer quels ports veulent quels trafics . Les messages peuvent être :

- MLD membership reports .
- MLD leave messages .

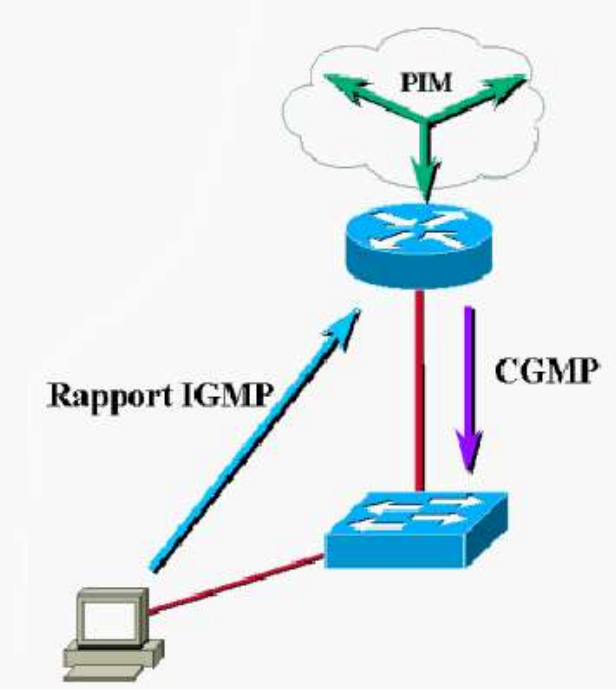
La charge de travail pour le commutateur est importante (il doit traiter toutes les trames multicast), et augmente en même temps que le trafic multicast. Cela nécessite un matériel spécialisé et coûteux.

CGMP — Cisco Group Multicast Protocol

Ce protocole propriétaire traite également des trames entre les commutateurs et les routeurs . le routeur envoie des paquets Multicast aux commutateurs à une adresse MAC conventionnelle (0100.0cdd.dddd). Le paquet CGMP contient trois éléments :

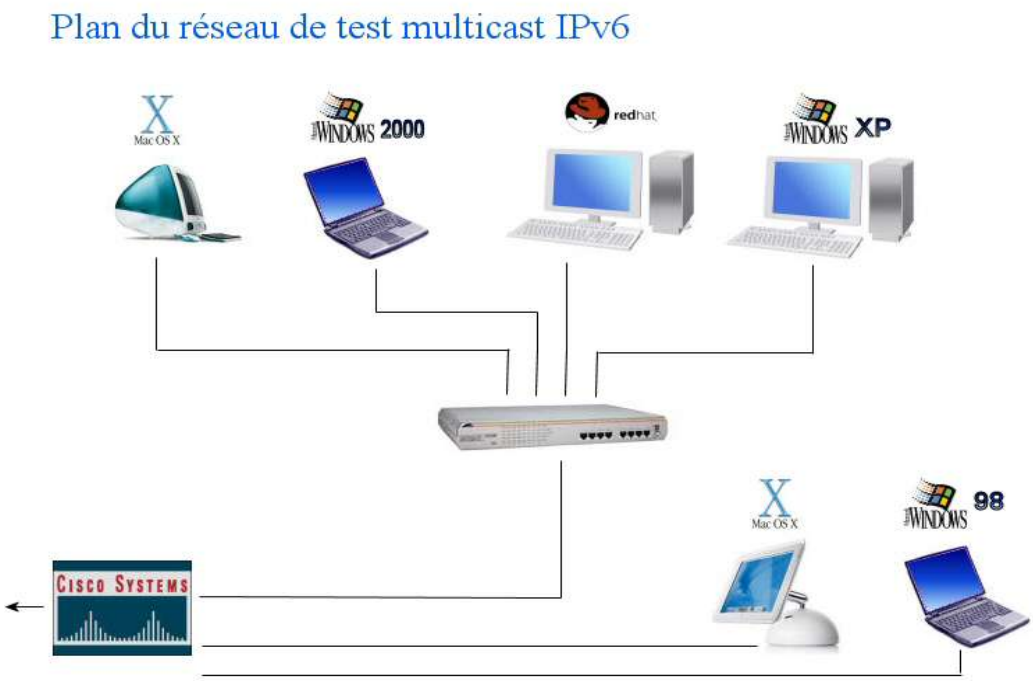
- le Type : Join or Leave
- la MAC address du client MLD
- l'adresse Multicast du groupe

Le commutateur utilise l'information du paquet CGMP pour ajouter ou supprimer une entrée pour une MAC address Multicast particulière. Ce protocole programme dynamiquement les commutateurs en fonction des messages IGMP. Les performances de commutation Multicast sont dites de niveau 2. Ce protocole nécessite des routeurs et commutateurs Cisco , mais fonctionne sur les Cisco d'entrée de gamme.



MOSPF est l'extension Multicast de OSPF. Il inclut une information Multicast dans l'annonce de l'état du lien OSP pour construire les arbres de distribution. Chaque routeur maintient une image à jour de la topologie du réseau. L'algorithme Dijkstra est utilisé pour calculer l'arbre du plus court chemin. L'inconvénient de ce protocole est qu'il ne fonctionne que sur les réseaux basés sur OSPF. L'algorithme Dijkstra tourne pour tous les couples Multicast (S,G) ce qui pose des problèmes d'échelle et ne fonctionne pas pour les arbres partagés. Cela rend ce protocole inapproprié pour les grandes interconnexions de réseaux avec plusieurs sources et récepteurs

III . Mise en Pratique



Lors de la réalisation de mon stage au service informatique et réseau de l'Insa de Rouen , j'ai pu bénéficier de plusieurs machines disposant de différents systèmes d'exploitation. Le réseau de test que je gérai directement se composait de quatre machines.

1 – Un I-mac sur lequel j'ai installé **mac OS X Panther,**

2 – Un PC tournant sous **Windows 2000 Pro,**

3 – Un PC **linux,**

4 – Un PC tournant sous **Windows XP.**

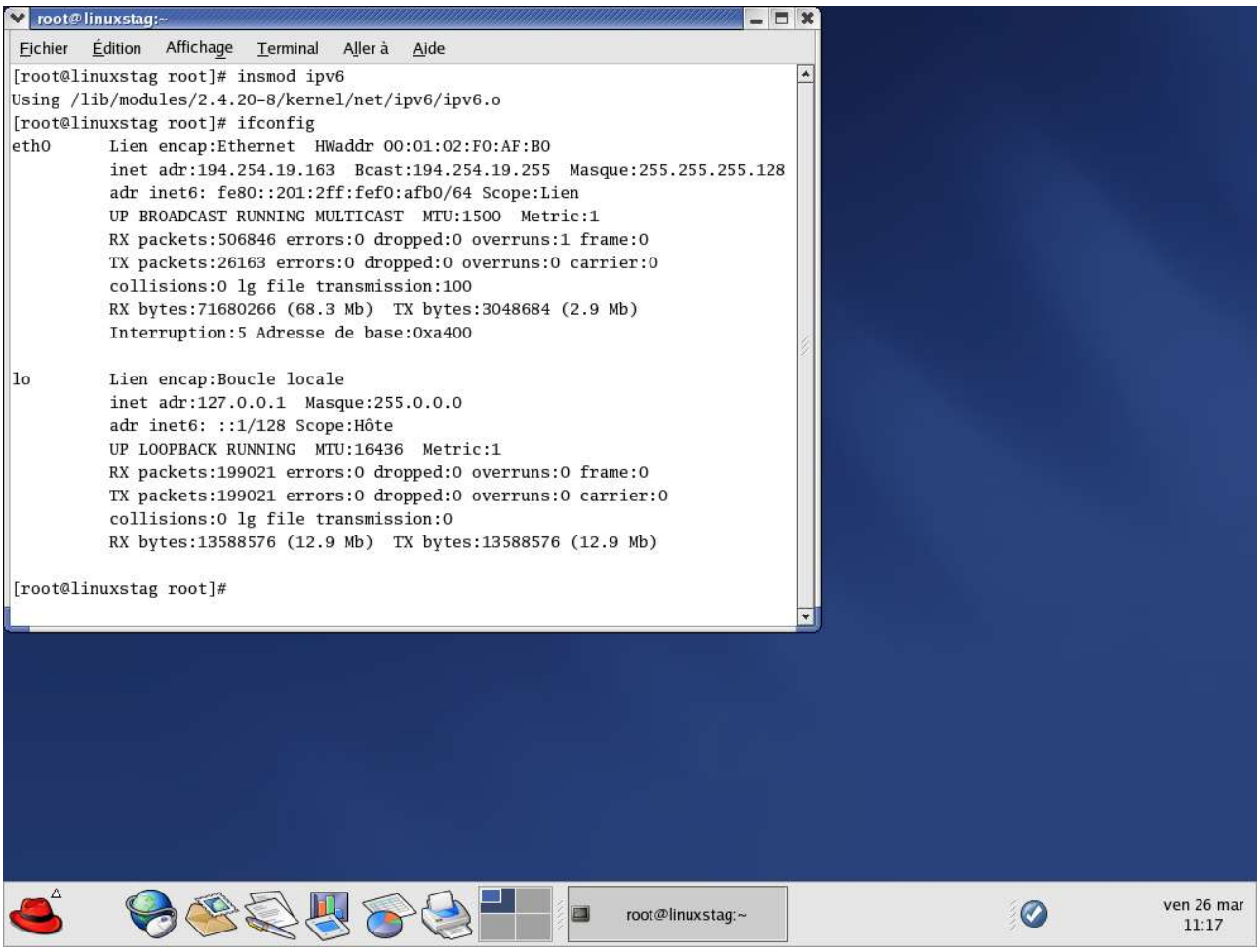
Le tout étant relié par un switch **CentreCom FS 708**. Ce mini réseau est relié à un routeur **CISCO**, le même que les postes des deux ingénieurs du service. Ces deux ordinateurs supplémentaires ont pu servir de clients lors des expérimentations.

La première étape a consisté à installer IP version 6 sur toutes ces machines.

I. INSTALLATION D'IPv6

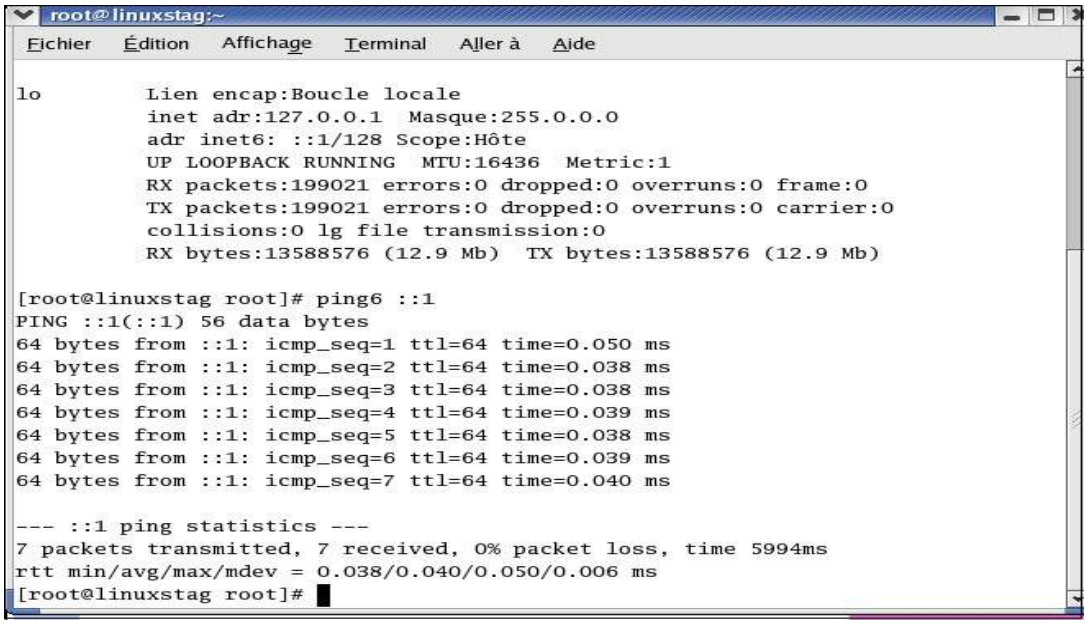
1. Installation sous Linux

Afin de partir sur une machine « propre », j'ai choisi de réinstaller Linux , dans une version qui comprenait IPv6 (noyau à partir du 2.4.x, bien que les premières fonctions aient été implémentées dès le 2.1.8) . J'ai décidé d'installer une **Redhat 9**, dernière version disponible en Octobre 2003. J'ai choisi d'installer une version minimale puis d'ajouter un par un les packages nécessaires.
L'activation d'IP V6 se fait par la commande `insmod ipv6`.



La commande ifconfig nous donne les différentes adresses de la carte ethernet. On remarque une adresse IPv6 commençant par `fe80` (dans eth0, ligne adr inet6:). Il s'agit de l'adresse « lien local » (voir partie théorique).

Afin de vérifier qu'IPv6 est bien en place, on peut faire un ping sur l'adresse loopback : ping6 ::1



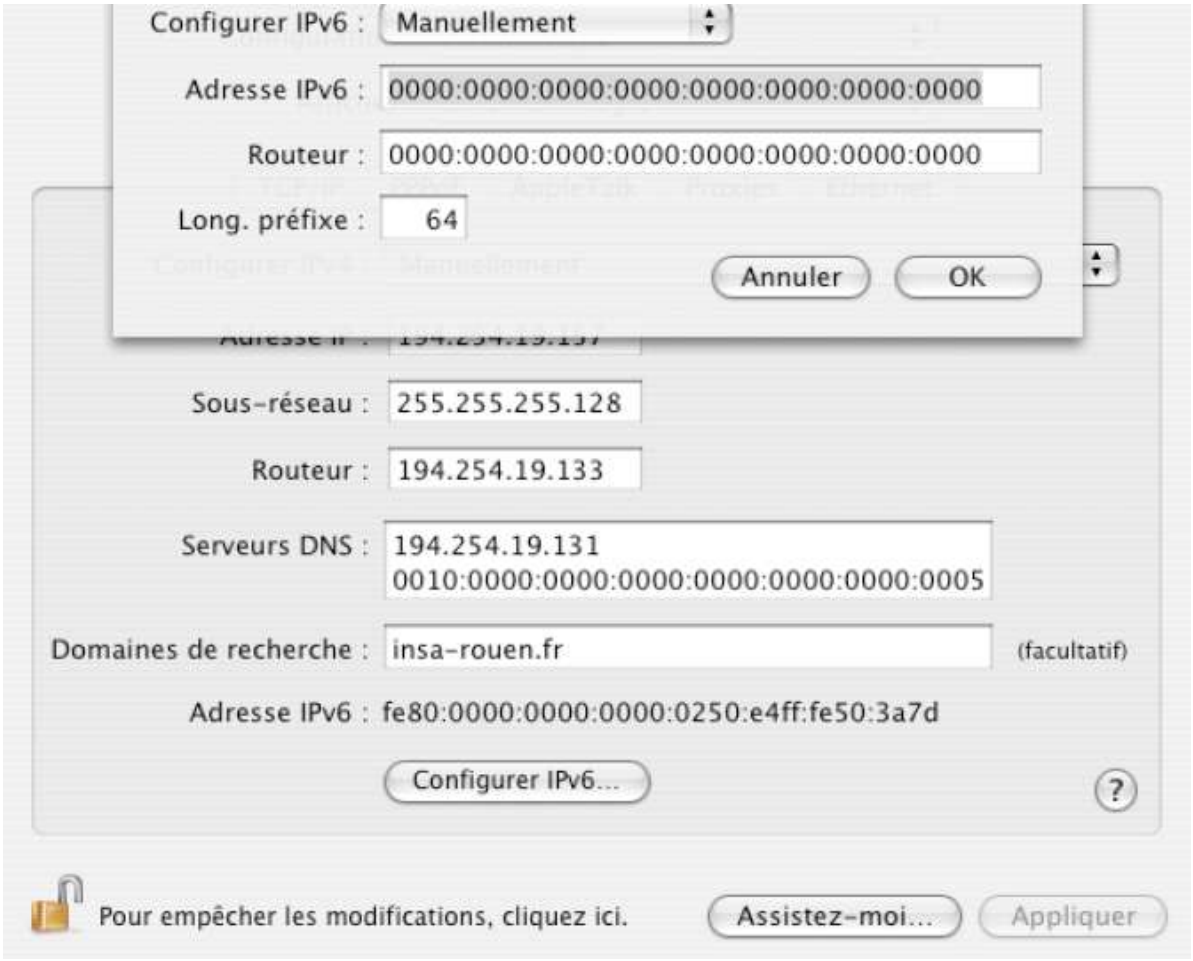
Le ping fonctionne bien. On décide d'entrer un numéro IP fixe.

ifconfig eth0 add 10::5

Le ping fonctionne, on peut ajouter un nom avec l'adresse 10::5 dans le fichier HOSTS

2. Installation sous MAC OSX

Mac OS X gère bien IPv6. La configuration peut se faire via le terminal (DARWIN) avec un ifconfig en0 ou via l'interface graphique.



On peut également entrer le nom MacV6 dans le DNS, ainsi qu'une adresse IPv6 fixe (10::6). Remarquons l'adresse IPv6 (en bas de la capture de la page), il s'agit de l'adresse lien local. Les adresses IPv6 et Routeur ne sont pas encore entrées.

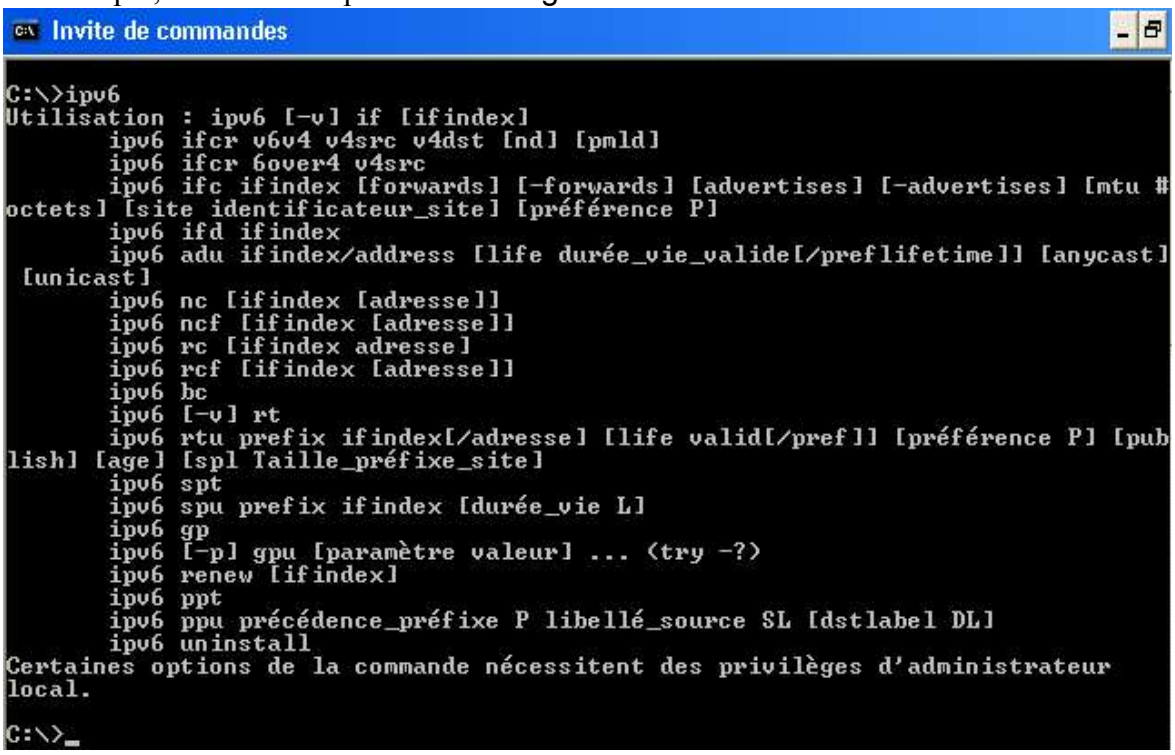
3. Installation sous WINDOWS 20000

La couche IPv6 n'est pas implémentée en Windows 2000. Après de longues recherches, on peut trouver une **pile IPv6** (IPv6 stack) sur le site recherche et développement de Microsoft.(tcpIPv6-001205.exe). Cette pile est fournie telle-quelle, avec un minimum d'explications. Microsoft ne compte pas développer cette pile plus en avant, et elle est laissée à disposition des utilisateurs avec toutes les réserves possibles.

L'installation se fait en plusieurs étapes :

1- Bien que ce ne soit pas documenté, on s'aperçoit que le protocole refuse de s'installer si on n'a pas auparavant mis en place le Service Pack 1. En Novembre 2003, le dernier service pack disponible est le SP4, qui contient tous les SP précédents. Je l'ai donc installé.
2- Le fichier EXE téléchargé doit être décompressé sous DOS avec la commande setup-x
3- Il faut installer le protocole réseau dans « Propriétés réseau – Protocoles – Installer – Disque Fourni ».

Les commandes Microsoft diffèrent des commandes standard.
Par exemple, IPv6 if correspond à ifconfig.



4. Installation sous WINDOWS – Autres versions

Sous **Windows 98** et les versions antérieures, il n'existe apparemment pas de pile IPv6.

Sous **Windows XP**, une fois téléchargée la mise à jour, l'installation se fait sans problème. La capture d'écran ci-dessous vient d'une version Windows XP. Il s'agit de l'exécution de la commande ipv6 if. Nous remarquons plusieurs interfaces.

L'interface 4 correspond à la carte Ethernet. L'adresse de la couche liaison est l'adresse MAC de la carte. Vient ensuite l'adresse Lien Local, configurée automatiquement. Puis le logiciel nous signale que le multicast est déjà implémenté, avec les adresses FF01::1 (tous les groupes au niveau noeud local) et FF02::2 (tous les groupes au niveau lien local, adresses faisant partie des « well known addresses »).

Les interfaces 3 et 2 servent à la création de tunnels pouvant faire cohabiter Ipv4 et IPv6. Notons que l'interface 2 a pour adresse Lien local la forme mixée fe80::5cfc:194.254.19.165 (voir I.8, adressage IPv6). 194.254.19.165 est bien sûr l'adresse Ipv4 de la machine.

L'interface 1 correspond à l'adresse Loopback interface de bouclage). Son adresse est ::1.

```
C:\Documents and Settings>ipv6 if
Interface 4 : Ethernet: Connexion au réseau local
(E8E1E130-87F7-4A46-BD27-9215E1FA1C76)
  utilise la découverte de voisins
  utilise la découverte de routeur
  adresse de la couche de liaison : 00-04-76-99-27-ae
    preferred link-local fe80::204:76ff:fe99:27ae, vie infinite
    multidiffusion interface-local ff01::1, 1 refs, non signalable
    multidiffusion link-local ff02::1, 1 refs, non signalable
    multidiffusion link-local ff02::1:ff99:27ae, 1 refs, dernier informateur
  MTU de liaison 1500 (MTU de liaison réelle 1500)
  limite de sauts actuelle 128
  durée d'attente pour la communication 34500ms (base 30000ms)
  intervalle de retransmission 1000ms
  DAD transmet 1
Interface 3 : Pseudo-interface de tunneling 6to4
(A995346E-9F3E-2EDB-47D1-9CC7BA01CD73)
  n'utilise pas la découverte de voisin
  n'utilise pas la découverte de routeur
  préférence de routage 1
    preferred global 2002:c2fe:13a5::c2fe:13a5, vie infinite
  MTU de liaison 1280 (MTU de liaison réelle 65515)
  limite de sauts actuelle 128
  durée d'attente pour la communication 22500ms (base 30000ms)
  intervalle de retransmission 1000ms
  DAD transmet 0
Interface 2 : Pseudo-interface de tunnels automatiques
(48FCE3FC-EC30-E50E-F1A7-71172AEEE3AE)
  n'utilise pas la découverte de voisin
  n'utilise pas la découverte de routeur
  préférence de routage 1
  Adresse IPv4 imbriquée EUI-64 : 0.0.0.0
  adresse de couche de liaison : 0.0.0.0
    preferred link-local fe80::5efe:194.254.19.165, vie infinite
  MTU de liaison 1280 (MTU de liaison réelle 65515)
  limite de sauts actuelle 128
  durée d'attente pour la communication 33500ms (base 30000ms)
  intervalle de retransmission 1000ms
  DAD transmet 0
Interface 1 : Pseudo-interface de bouclage
(6BD113CC-5EC2-7638-B953-0B889DA72014)
  n'utilise pas la découverte de voisin
  n'utilise pas la découverte de routeur
  adresse de la couche de liaison :
    preferred link-local ::1, vie infinite
    preferred link-local fe80::1, vie infinite
  MTU de liaison 1500 (MTU de liaison réelle 4294967295)
  limite de sauts actuelle 128
  durée d'attente pour la communication 39500ms (base 30000ms)
  intervalle de retransmission 1000ms
  DAD transmet 0
```

5. autoconfiguration et routage : ZEBRA

Afin de faire communiquer les machines entre elles, mais également de bénéficier de l'autoconfiguration avec préfixe, il a été décidé d'adjoindre un routeur IPv6 à ce mini réseau. Par mesure de sécurité, et pour pouvoir effectuer les tests en toute tranquillité, il n'a pas semblé souhaitable d'utiliser le routeur Cisco de l'Insa.

Après quelques recherches, j'ai choisi d'installer **ZEBRA** sur le poste Linux, pour plusieurs raisons :

- Zebra est un « émulateur cisco » : il en reprend les commandes de base.
- Logiciel gratuit sous licence Open Source, il est reconnu comme performant.
- Quelques recherches m'ont montré qu'il avait déjà été mis en place dans plusieurs organismes (Universités ...).
- Il est reconnu comme l'un des meilleurs routeurs logiciels en IPv6

Son principal inconvénient par rapport à un routeur Cisco est qu'il ne route pas le Multicast.

Zebra est le démon de routage fourni avec la distribution Linux RedHat. Cependant, la version fournie est la 0.93, elle ne route pas IPv6. J'ai téléchargé et installé la version 0.94

Une fois la version téléchargée et décompressée, on exécute
./configure

Cette commande vérifie entre-autre si IPv6 est installé sur la machine. Si ce n'est pas le cas, il faut relancer le configure après un insmod IPv6

make
make install

Il faut ensuite éditer le fichier Zebra.conf pour y ajouter les lignes
hostname linuxstag.insa-rouen.fr
password stag2003

Si la ligne password est absente du fichier conf, Zebra refusera la connexion en TELNET pour la configuration. (/usr/local/etc/zebra.conf)

A partir de cette étape, Zebra est installé. Il faut maintenant le configurer.

Les commandes comprises par Zebra sont proches des commandes Cisco :

hostname : choisi le nom du routeur.
password : ajoute un mot de passe pour les communications Telnet
enable password : pour le mot de passe en mode « enable »
configure terminal : configure le routeur
wr t : écrit la configuration sur le terminal
wr m : sauve la configuration en mémoire.

Dans le menu configuration, on retrouve les commandes

ip address
ipv6 address
multicast
ip route

On lance Zebra dans une fenêtre Terminal :

zebra -d

telnet localhost 2601

en

conf t

ip address 194.254.19.16X / XX (si on entre un mauvais masque, Zebra provoque un message « unknown command ». Notons que l'on peut employer le / pour les masques)

ipv6 nd send-ra (le routeur envoie un send advertisement)

ipv6 nd prefix-advertisement 2001:660:7403:1001/48 (le préfixe de l'insa est 2001:660:7403. Pour le réseau de test, on a décidé d'y ajouter le préfixe 1001)

IPv6 address 2001:660:7403:1001::1/48 (on force l'attribution d'une adresse IP fixe au routeur)

write m

Le routeur se reconnaît avec le PING. On note que c'est alors Zebra qui répond, et non la machine Linux (On peut faire le test en attribuant à Zebra une adresse différente de celle de la machine qui l'héberge).

Sur MacOS ou Windows, la fonction configuration automatique donne maintenant un numéro IPv6 commençant par le bon préfixe .

Grâce au routeur, notre mini-réseau fonctionne. On constate qu'un PING6 effectué sur une des machines membre du réseau vers une autre obtient une réponse satisfaisante.

Le mini réseau IPv6 fonctionne.

Remarque : Comme nous l'avons vu, IPv6 propose un très grand nombre d'adresses. Cela permet de hierarchiser et de développer l'adressage de manière logique. Si nous avons adjoint 1001 au préfixe de l'Insa pour définir notre réseau de test, il est clair qu'une réfléxion doit être menée sur le plan d'adressage, au niveau de l'établissement. L'annexe 6 reprend la réfléxion menée sur le sujet par le service informatique de l'Université de Caen.

II. Emission Multicast

Afin d'avoir un support simple pour l'émission Multicast, j'ai choisi de diffuser de la vidéo. J'avais déjà expérimenté dans le cadre professionnel un serveur **RealVideo** qui , à l'époque, diffusait en Multipoints.

Mon choix sur le logiciel de diffusion s'est porté sur **VIDEOLAN,** un projet développé par les étudiants de l'école Centrale de Paris.

Ce projet, open source, repose sur deux programmes principaux : **VLC** (VideoLan Client) et **VLS** (VideoLan Serveur). **VLC** était à l'origine le produit servant de Client. Cependant, il peut dans ses dernières versions diffuser un flux en direct. Il ne s'agit pas de vidéo à la demande (plutôt du domaine de **VLS**) mais de la vidéo envoyée en direct, à partir d'une source en streaming ou pré-existante. Pour réaliser les tests Multicast, j'ai choisi de me limiter à ce mode de diffusion, c'est pourquoi je n'ai installé que **VLC**. Mon fichier source pour les tests était un fichier MPEG 2. L'objectif que je voulais atteindre était l'envoi de vidéo et la captation du flux par plusieurs machines, sans faire du Multipoint.

Notons qu'il existe plusieurs applications compatibles avec le Multicast. Quelques exemples sont donnés en Annexe 7.

1. Installation de VLC sur les différentes machines

Sous **Window**s et **Mac OSX** l'installation est automatique.

Sous **Linux**, un certain nombre de librairies sont requises:

* libdvbpsi (obligatoire) ,
* mpeg2dec (obligatoire) ,
* libdvdcss si on désire pouvoir lire des DVD encryptés ,
* libdvdplay si on souhaite profiter des menus DVD ,
* a52dec pou pouvoir décoder le son AC3 (A52), souvent utilisé dans les DVDs ,
* ffmpeg, libmad, faad2 pour lire des fichiers MPEG 4 / DivX ,
* libogg & libvorbis pour lire des fichiers Ogg/Vorbis .

Seuls les tarballs fournis sur le site videolan.org sont officiellement supportés.

Pour chacune des librairies :

* il faut décompresser

```
% tar xvzf library.tar.gz
```

ou

```
% tar xvjf library.tar.bz2
```

* configurer :

```
% cd library
% ./configure

% make
# make install
```

Il faut ensuite télécharger les sources de la dernière version : récupérer le fichier vlc-version.tar.gz depuis la page de téléchargement des sources de VLC.

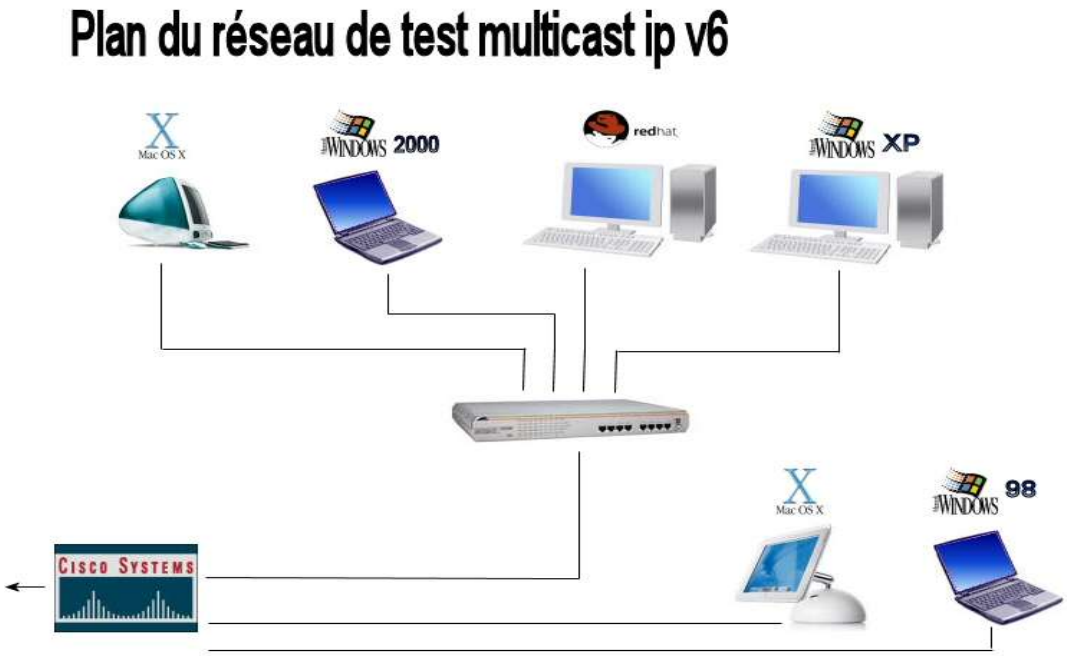
```
% tar xvzf vlc-version.tar.gz
% cd vlc-version
```

* pour un VLC très basique, utiliser :

```
% ./configure
% make
% su
Password:  [Root Password]
# make install
```

2. Mise en place de l'émission

Une fois les clients installés sur les 4 machines, voici l'utilisation proposée :



Le poste Linux servira de routeur (**ZEBRA**) en IPv6.
Le poste Windows **2000** sera le VLC diffusant la vidéo (serveur).
Les autres postes (**Windows** et **Mac**) seront les clients VLC.

Il est à noter que le poste Windows 2000 est un Pentium 90. Ses spécifications lui suffisent à envoyer de la vidéo en ligne mais pas à l'afficher en même temps.

Le protocole choisi pour l'émission est l'**UDP** car :
- Il est économique en ressources au niveau de l'Unicast.
- Il est bien supporté dans le cas du Multicast.

Les essais se feront en 2 étapes :
1 – Diffusion Unicast IPv4 et v6.
2 – Diffusion Multicast IPv4.

Après ces deux étapes, IPv6 sera mis en place sur le routeur CISCO de l'INSA et des essais pourront être réalisés en multicast IPv6.

3. Options avancées lors de la diffusion d'un flux

Le flux de sortie (stream output) possède plusieurs modules. Chaque module apporte des fonctionnalités, et il est possible de chaîner les modules pour combiner ces possibilités .

Voici la liste des modules disponibles :

- **standard** "envoie" le flux grâce à un module de sortie: par exemple, UDP, fichier, HTTP, ...
- **transcode** permet de transcoder à la volée l'audio et la vidéo de votre flux d'entrée (si l'ordinateur dispose de suffisamment de puissance) .
- **duplicate** permet de créer une seconde chaîne dans laquelle le flux sera traité séparément .
- **display** permet d'afficher le flux d'entrée, comme VLC le ferait normalement. Utilisé avec le module duplicate, il permet de voir le flux en même temps qu'il est envoyé.
- **es** permet de séparer les flux élémentaires (ES) d'un flux d'entrée .

Chaque module prend des options. Voici la syntaxe à utiliser :

```
% vlc input_stream --sout '#module1{option1=...,option2=...}:#module2
{option1=...,option2=...}:...'
```

Par exemple, pour transcoder et envoyer un flux :

```
% vlc input_stream --sout '#transcode{options}:#standard{options}'
```

Les option peuvent être ::

- **access:** cette option décrit la manière d' envoyer le flux : file, udp, rtp, http.
- **mux:** signale le codec à utiliser. Il peut être complété par: avi (format AVI) ogg (format Ogg) ps (format MPEG2-PS) ts (format MPEG2-TS) .
- **url**:Il s'agit de l'emplacement du fichier, sinon, l'adresse IP Multicast ou Unicast .
- **sap**: pour annoncer le flux par SAP/SDP. L'option contient le nom sous lequel le programme sera annoncé .
- **slp**: comme sap, en utilisant le protocole SLP.
- **sap_ipv**: spécifie si vous les annonces SAP sont envoyées en IPv4 -défaut- ou IPv6. La valeur à utiliser est 4 ou 6 .

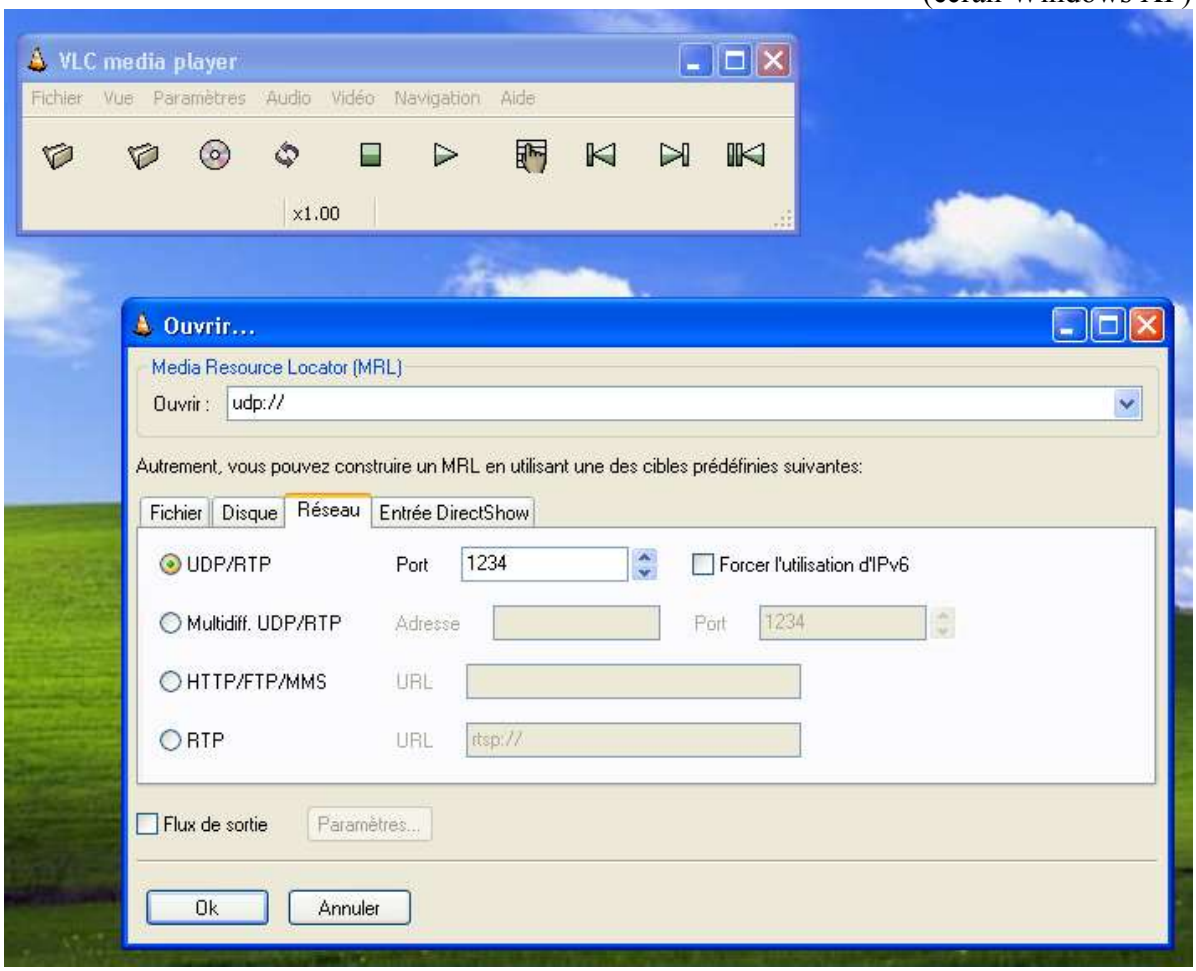
Note: pour le Multicast, on peut utiliser l'option globale --ttl 12 pour régler le TTL.

Ces options sembleront plus explicites lors des expérimentations.

4. Diffusion Unicast IPv4

Coté client :

(écran Windows XP)



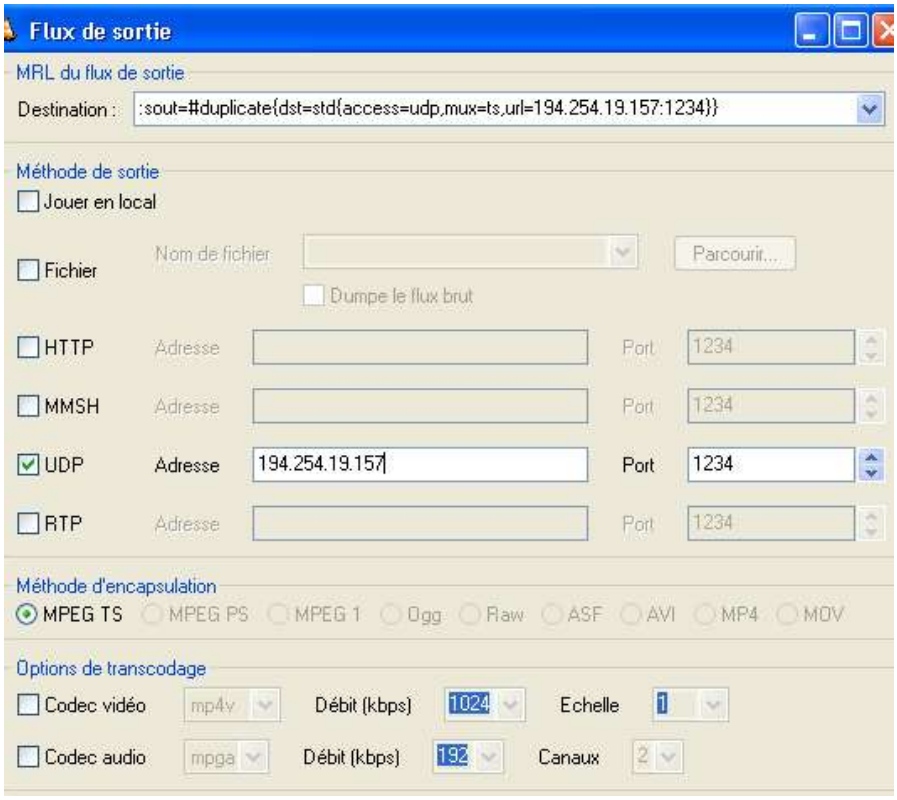
Vu les choix effectués précédemment dans le mode de diffusion, l'écoute coté client est relativement simple.

Il suffit de choisir dans le menu « fichier » : « Ouvrir un flux réseau » puis écouter en **UDP** le port 1234 (port par défaut de VLC). Si une vidéo compréhensible par VideoLan atteint cette machine, elle est automatiquement affichée.

La capture (sous Mac OS ci dessous) montre que le flux est ouvert en UDP sans aucune analyse de la part du client.

Remarquons également que l'écoute se fait « à la volée ». Le client UDP écoute sur sa propre carte mais ne connaît pas l'adresse de l'émetteur. Par ailleurs, il ne sait pas non plus depuis combien de temps l'émission est commencée (le temps reste figé à 0). Il peut joindre et quitter le l 'écoute sans que cela n'influence l'émetteur.

coté émetteur :



Une fois la source vidéo choisie (DVD, fichier présent sur le disque ou flux vidéo venant d'une carte d'acquisition), il faut sélectionner le mode de diffusion. La ligne en haut de la fenêtre correspond aux options de VLC lorsqu'on l'utilise en mode « commande ». (voir partie II.3 : options avancées lors de l'émission de flux).

On note l'adresse destinataire en UDP ainsi que l'encapsulation. Les différents modes d'encapsulation peuvent avoir des intérêts divers selon les clients. Dans le cas d'un parc multi-machine, l'encapsulation MPEG est à recommander.

Le serveur peut éventuellement transcoder la vidéo « à la volée » afin de l'émettre. Cela demande une machine d'émission un peu plus puissante mais peut être intéressant lorsque la vidéo source est dans un standard non universel, par exemple en DIVX.

Notons que la version MacOs X de VLC ne propose l'émission de vidéo qu'en mode de commande.

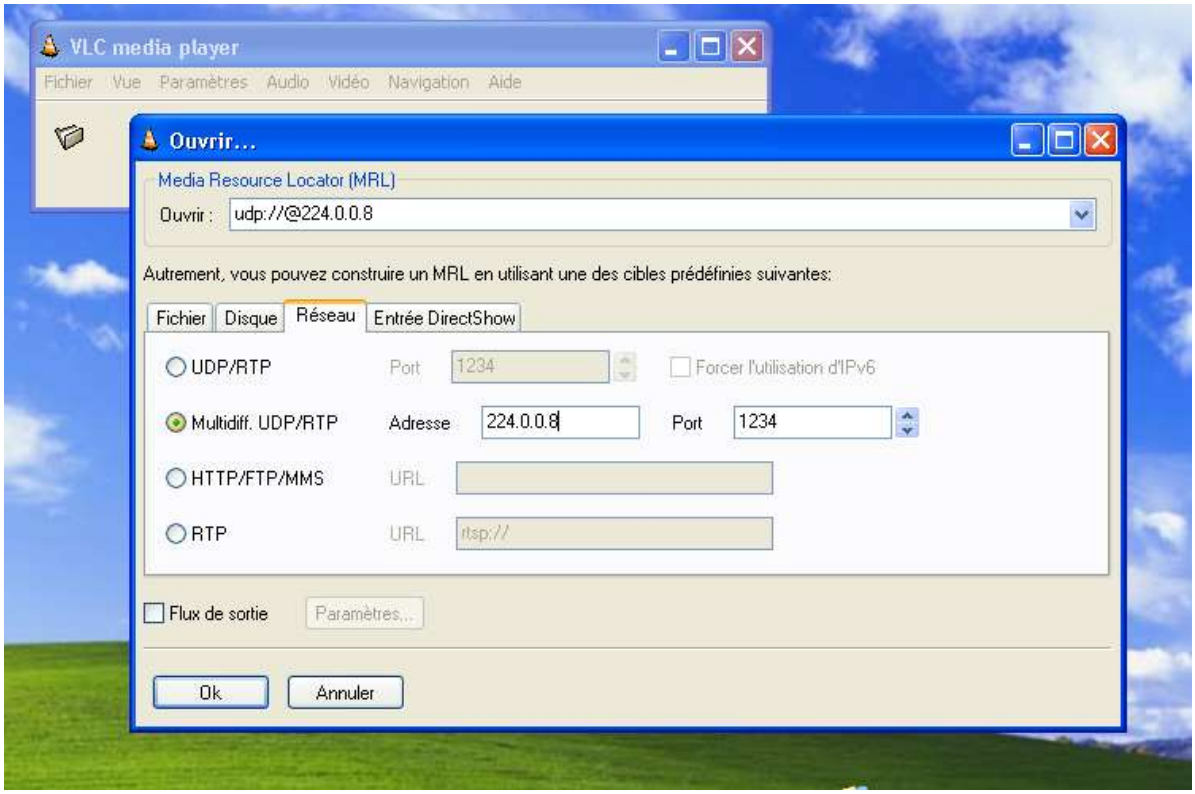
5. Emission Unicast IPv6

Les dernières versions de VLC supportent IPv6. Côté client, il n'y a rien à modifier en théorie. Cependant, on remarque sous Mac Os X que le flux fonctionne nettement mieux lorsqu'on force l'utilisation d'IPv6.

6. Emission Multicast Ipv4

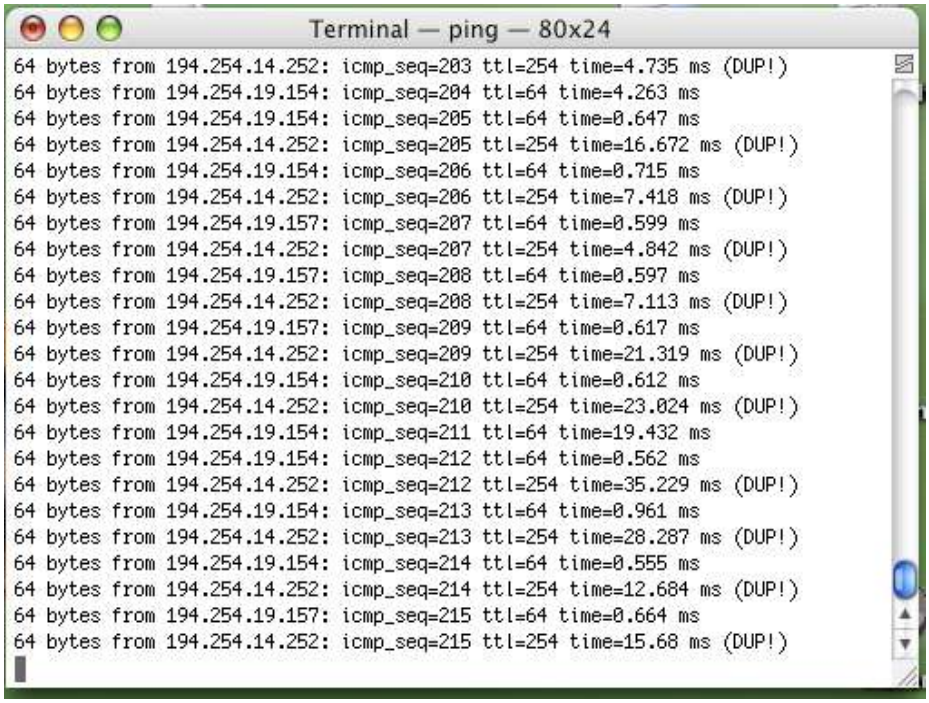
Les clients sont mis en place sur plusieurs machines. A chaque fois, ils écoutent sur leur propre carte. L'adresse choisie pour l'émission Multicast est 224.0.0.8 (adresse connue comme adresse Multicast mais non réservée). On choisi toujours d'émettre en UDP.

Au niveau du client, il est nécessaire de signaler quel flux Multicast est écouté.



Notons qu'il est possible d'organiser une diffusion Multicast sans abonnement préalable en émettant sur l'adresse **224.0.0.1** (tous les hôtes Multicast du réseau). Le client se contente alors d'écouter sans préciser l'adresse de la diffusion, mais on retombe alors dans le travers du Broadcast.

Lorsque seule la machine diffusant le flux émet, aucune réponse n'est obtenue sur un ping 224.0.0.8. Si on connecte un client, c'est ce dernier qui répond au ping. Si plusieurs clients sont connectés, ils répondent tous.



Dans cet exemple, les clients .154 et .157 sont connectés et écoutent l'émetteur sur le 224.0.0.8. La machine 194.254.14.252 est un switch qui semble écouter le Multicast (swhumarc.insa-rouen.fr).

7. Dernière étape, configuration du CISCO de l'Insa en IPv6

Le mini-réseau fonctionne, ainsi que le multicast en IPv4. Il est maintenant temps de tester le procédé en IPv6. L'inconvénient de Zebra est qu'il ne route pas le multicast. Il faut donc configurer un routeur CISCO en Ipv6.

La configuration doit se faire au niveau global et au niveau des interfaces.

Une adresse IPv6 doit être configurée pour l'interface qui fait suivre les paquets IP. Configurer une adresse IPv6 pour une interface lui donne directement une adresse lien-local et active l'IPv6. Par ailleurs, l'interface écoutera les groupes multicast suivants :

- Noeuds sollicités pour le multicast FF02::1:FF00/104.
- Tous les noeuds multicast FF02::1.
- Tous les routeurs multicast FF02::2.

L'adresse noeuds sollicités (solicited nodes) est utilisée pour la découverte des voisins.

Pour configurer une adresse sur une interface, on peut utiliser les commandes suivantes :

interface + type d'interface, numéro d'interface

ipv6 address ipv6prefix/longueur préfixe eui-64 . La commande eui-64 permet d'automatiser la configuration (autoconfiguration IPv6). Si on n'utilise pas la commande eui-64, il faut assigner une adresse complète.

```
interface eth0
ipv6 address [une_ip_de_votre_bloc]/64
ipv6 nd ra-interval 60
ipv6 nd ra-lifetime 180
ipv6 nd prefix-advertisement [votre_segment_ipv6]::/64 360 60 autoconfig
no cdp enable
```

Si nous détaillons ces commandes :
On souhaite affecter une adresse à l'interface ETH0, première interface ethernet.
L'adresse IP est une adresse fixe.
RA correspond au Router Advertisement.
On annonce ensuite le préfixe du réseau, afin de bénéficier de l'auto-configuration.

J'ai préparé les commandes afin de les transmettre à l'ingénieur chargé du réseau à l'Insa. Cela m'a amené à contacter le CRIHAN, organisme raccordant l'INSA au reste du monde.

```
ipv6 unicast-routing

router bgp <as-insa>
 neighbor 2001:660:2001:A000:101:103::1 remote as 2187

address family ipv4
 no neighbor 2001:660:2001:A000:101:103::1 activate

address family ipv6
 neighbor 2001:660:2001:A000:101:103::1 activate
 redistribute static
```

Les commandes ci-dessous sont attribuées à une interface.

```
conf t
ipv6 address 2001:660:7403:1001::1/64
ipv6 nd ra-interval 60
ipv6 nd ra-lifetime 180
ipv6 nd prefix-advertisement 2001:660:7403:1001::/64 360 60 autoconfig
no cdp enable
```

Le routeur supporte maintenant l'IPv6:

```
ipv6 unicast-routing
...
!
interface GigabitEthernet0/1
  description Interco MSA et SER
                ip address 10.255.19.1 255.255.255.0
                no ip mroute-cache
```

```
                duplex auto
                speed auto
                media-type rj45
                no negotiation auto
 ipv6 address 2001:660:7403:1001::1/64
 ipv6 nd ra-interval 60
 ipv6 nd ra-lifetime 180
 ipv6 nd prefix 2001:660:7403:1001::/64
 no cdp enable
!
...
!
router bgp 64560
                no synchronization
                 bgp log-neighbor-changes
 neighbor 2001:660:2001:A000:101:103:0:1 remote-as 2187
       neighbor 192.168.69.1 remote-as 2074
 neighbor 193.48.154.137 remote-as 2187
                no auto-summary

 address-family ipv6
 neighbor 2001:660:2001:A000:101:103:0:1 activate
 no synchronization
 redistribute connected
 redistribute static
 exit-address-family
 !
```

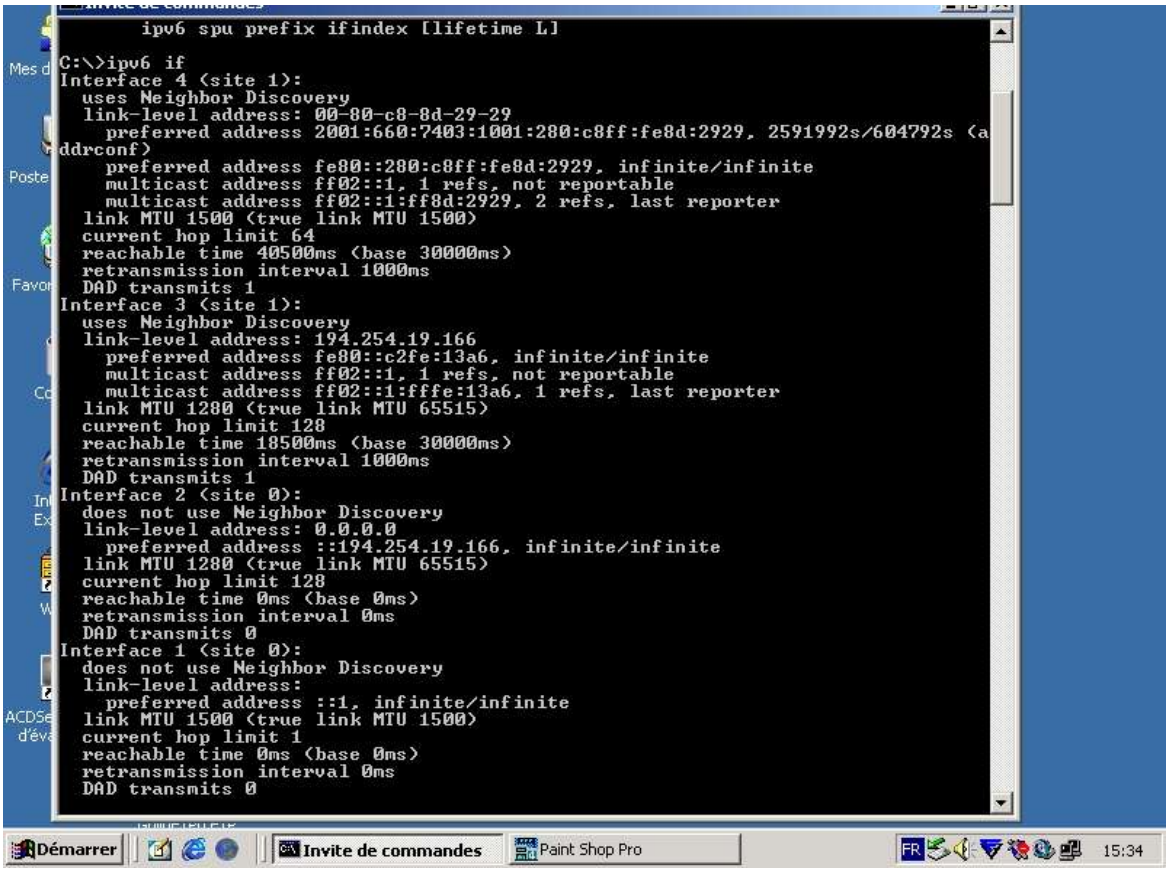
Nous avons exécuté quelques commandes afin de nous assurer du bon fonctionnement :

```
agir#sh ipv6 ?
  access-list  Summary of access lists
  cef          Cisco Express Forwarding for IPv6
  interface    IPv6 interface status and configuration
  local        IPv6 local options
  mtu          MTU per destination cache
  nat          IPv6 NAT-PT information
  neighbors    Show IPv6 neighbor cache entries
  ospf         OSPF information
  prefix-list  List IPv6 prefix lists
  protocols    IPv6 Routing Protocols
  rip          RIP routing protocol status
  route        Show IPv6 route table entries
  routers      Show local IPv6 routers
  traffic      IPv6 traffic statistics
  tunnel       Summary of IPv6 tunnels


agir#sh ipv6 interface gigabitEthernet 0/1
GigabitEthernet0/1 is up, line protocol is up
  IPv6 is enabled, link-local address is FE80::2B0:8EFF:FE6B:981B
  Description: Interco MSA et SER
  Global unicast address(es):
    2001:660:7403:1001::1, subnet is 2001:660:7403:1001::/64
  Joined group address(es):
    FF02::1
    FF02::2
    FF02::1:FF00:1
    FF02::1:FF6B:981B
  MTU is 1500 bytes
  ICMP error messages limited to one every 100 milliseconds
  ICMP redirects are enabled
  ND DAD is enabled, number of DAD attempts: 1
  ND reachable time is 30000 milliseconds
```

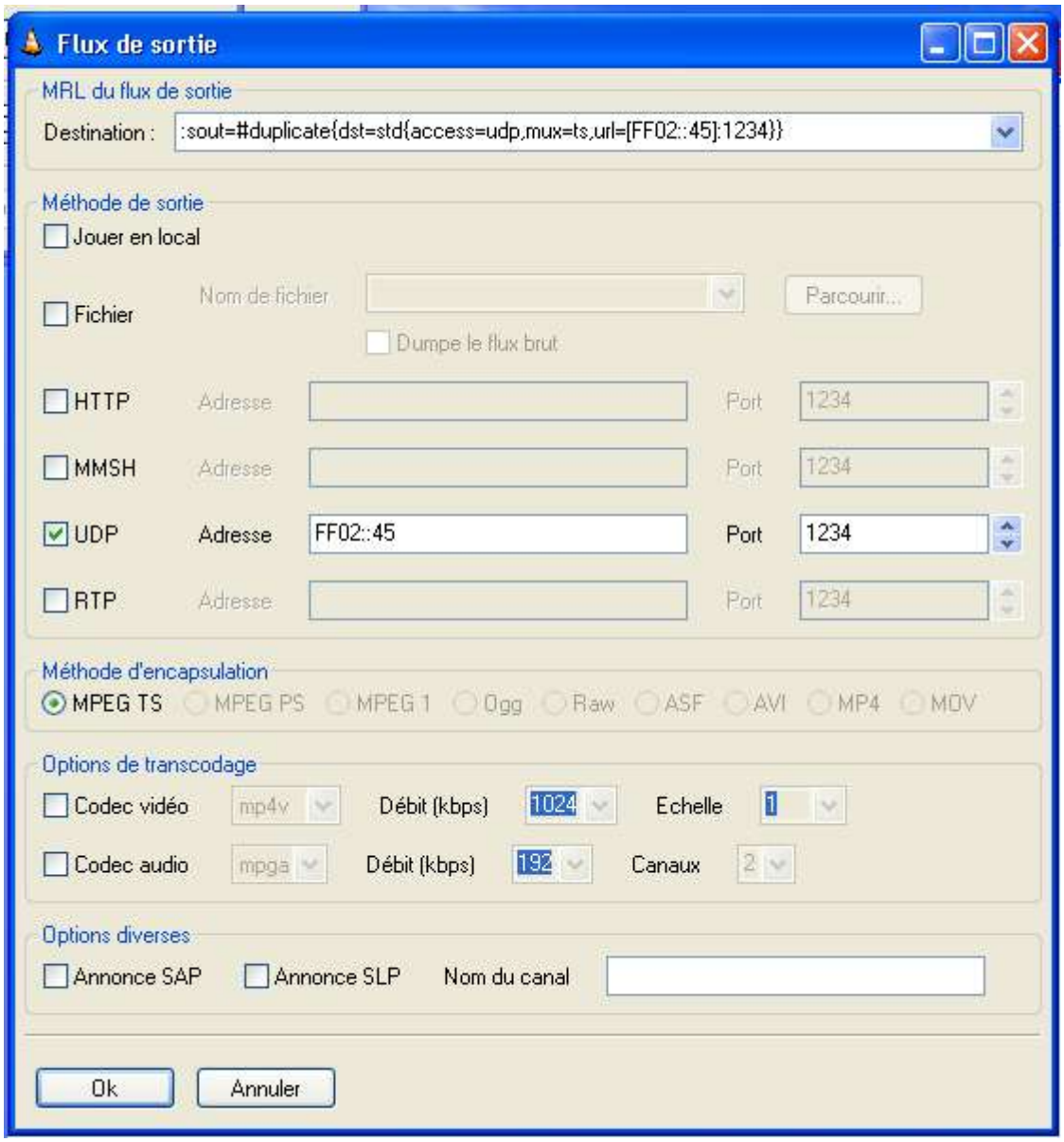
```
  ND advertised reachable time is 0 milliseconds
  ND advertised retransmit interval is 0 milliseconds
  ND router advertisements are sent every 60 seconds
  ND router advertisements live for 180 seconds
  Hosts use stateless autoconfig for addresses.


agir#sh ipv6 route
IPv6 Routing Table - 4 entries
Codes: C - Connected, L - Local, S - Static, R - RIP, B - BGP
       U - Per-user Static route
       I1 - ISIS L1, I2 - ISIS L2, IA - ISIS interarea
       O - OSPF intra, OI - OSPF inter, OE1 - OSPF ext 1, OE2 - OSPF
ext 2
C   2001:660:7403:1001::/64 [0/0]
     via ::, GigabitEthernet0/1
L   2001:660:7403:1001::1/128 [0/0]
     via ::, GigabitEthernet0/1
L   FE80::/10 [0/0]
     via ::, Null0
L   FF00::/8 [0/0]
     via ::, Null0
```

Remarque : Aujourd'hui, l'annonce du préfixe INSA ne se fait que sur le VLAN d'interco avec le routeur d'entrée de l'Insa . Nous attendons le code Cisco de routage IPv6 sur les commutateurs 3750 (responsables du routage IPv4 dans l'INSA) afin de pouvoir utiliser IPv6 sur tous les VLANs.

Branchons un ordinateur sur le CISCO. On remarque qu'il obtient bien le préfixe INSA (Interface 4, preferred address) auquel nous avons ajouté 1001 pour le réseau de test .

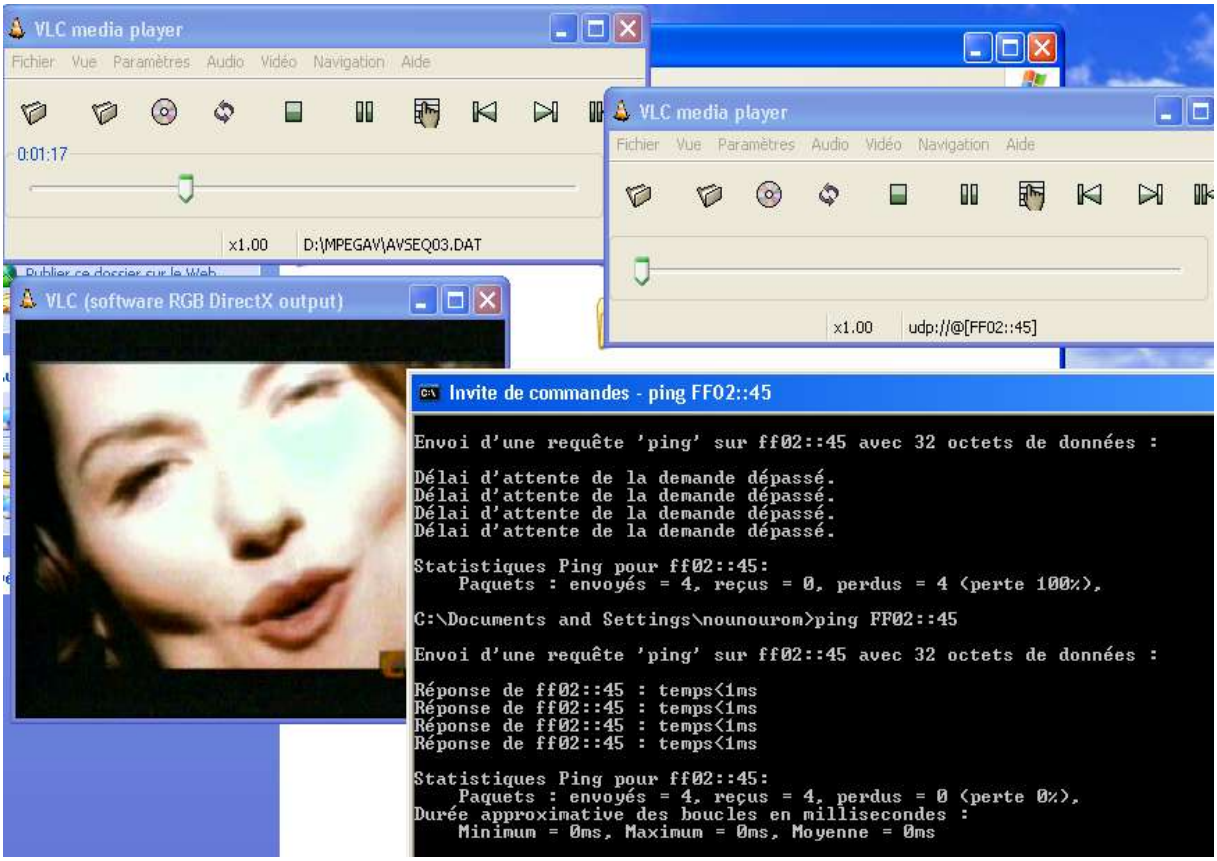


Relions deux machines au CISCO. On décide d'émettre sur une adresse Multicast FF02::45. La capture d'écran ci-dessous reprend la configuration du VLC serveur.



On remarque que notre client n'arrive pas à ouvrir le flux. Le problème peut venir de deux éléments : soit le routeur Cisco n'est pas configuré pour router le multicast, soit le logiciel VLC, encore en développement, ne sait pas traiter le multicast en IPv6. Nous décidons donc de faire tourner un VLC client et un VLC serveur sur la même machine, et d'envoyer le flux sur une adresse multicast que le client écoutera. La capture d'écran montre, à gauche, le VLC serveur (on remarque en bas de celui-ci le nom fichier diffusé. Il possède une barre de temps). A droite, le VLC Media Player Client. Il ne possède pas la barre de temps (il ne connaît pas la durée du flux) et écoute sur l'adresse FF02::45. C'est le client qui affiche la vidéo visible.

Alors que le serveur émettait, un PING a été effectué sur l'adresse FF02::45. Celui-ci est resté sans réponse.
Le même PING a été effectué une fois que le client écoutait l'adresse multicast. On remarque une réponse rapide. L'adresse Multicast en IPv6 répond, comme nous l'avons vu en Ipv4, avec l'interface du client.



VLC semble donc un outil adapté pour la diffusion vidéo Multicast en IPv6. La configuration du Cisco doit prendre en compte cette diffusion.

Conclusion

Comme nous l'avons vu, le monde de la recherche, mais également les industries de pointe, s'intéressent de plus en plus à IPv6. Si ce protocole est toujours en développement, les applications le prenant en compte sont de plus en plus nombreuses. L'INSA de Rouen est, depuis mon stage, raccordé en IPv6 au réseau RENATER. Suite à la configuration du routeur d'entrée de l'INSA, il est possible, depuis un poste situé au Service Informatique, d'utiliser l'autoconfiguration IPv6 afin d'obtenir une adresse publique (routable dans l'Internet). On peut alors consulter les sites disponibles uniquement en IPv6 (comme http://www6.crihan.fr). Dès que les routeurs Cisco 3750 pourront router l'IPv6, cette possibilités sera étendue à tout l'INSA (actuellement, seule une version bêta du système d'exploitation pour 3750 est disponible).

La mise en place de ce protocole sous les différents systèmes d'exploitation tend à se simplifier, et on peut raisonnablement penser que les équipements réseau géreront pour la plupart l'auto-configuration IPv6 d'ici quelques années. A ce niveau, l'administrateur réseau devra réfléchir sur le plan d'adressage de son établissement. Il lui sera donné la possibilité de mettre en place un adressage logique et hierarchisé, sans que le nombre d'adresses ne soit une limite.

Par contre, il faudra vraissemblablement prendre en compte le fait que la majorité des adresses seront publiques afin de modifier la politique de sécurité de l'établissement. En donnant à chacun la possibilité d'établir un serveur sur son propre équipement, IPv6 pose la question de la responsabilité de l'utilisation du réseau vis à vis de la loi. IPv6 dégage donc le responsable réseau du raccordement physique d'une machine (l'IPv6 est « plug and play ») mais l'amène à mener une reflexion sérieuse sur l'organisation du réseau et son utilisation.

Le Multicast, économique en ressource réseau , est renforcé en IPv6. J'ai pu, au cours de mon stage au sein du Service Informatique et Réseau de l'Insa de Rouen tester VLC, outils de diffusion et de réception vidéo en temps réel. Cet envoi de « flux » vidéo en direct semble être une bonne application du Multicast. Il est important de noter que l'option « envergure » de l'adresse multicast, introduite par IPv6, peut être utilisée pour affiner la diffusion. On pourra alors imaginer un flux envoyé aux utilisateurs d'un même bureau (envergure Lien Local), d'une même organisation.

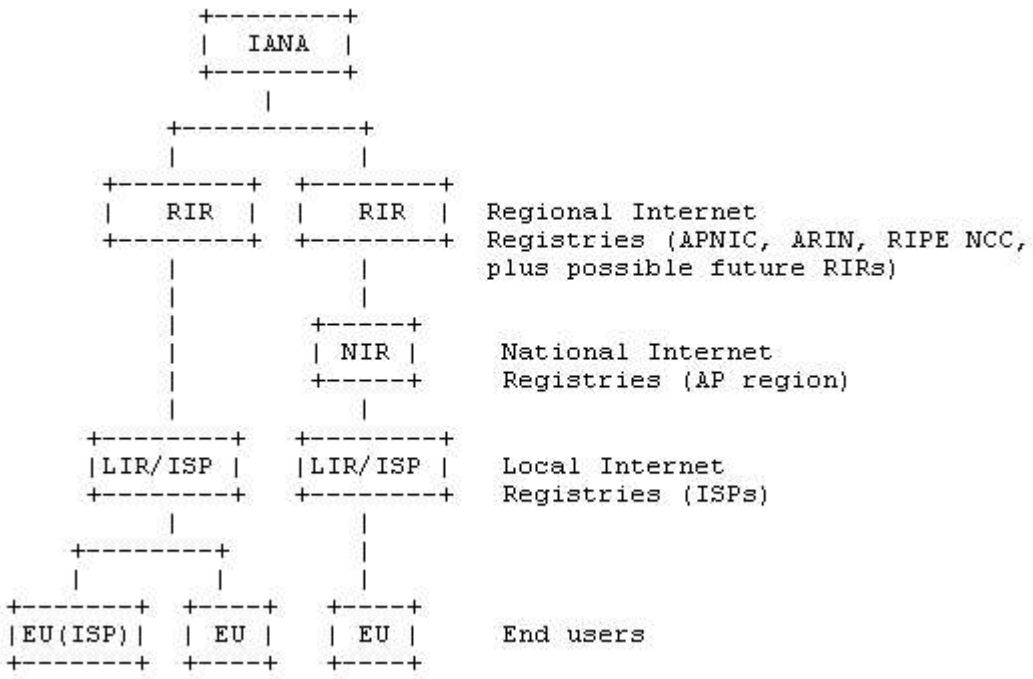
Alors qu'actuellement, de plus en plus de Fournisseurs d'Accès à Internet proposent au grand public la diffusion de chaînes de télévision via ADSL (offres Freebox, France Télécom...), la combinaison d'IPv6 et de la diffusion Multicast semble être techniquement intéressante afin de faire évoluer ces offres. Free et Wanadoo proposent d'ailleurs maintenant aux particuliers une adresse IPv6 Publique. La Qualité de Service et la Sécuration des données offertes par IPv6 seront vraissemblablement bientôt exploitée.

D'autres utilisations de ces techniques restent encore à découvrir, dans les domaines de l'informatique grand public et professionnelle et également de la communication et de la domotique.

Annexes

Annexe 1 : Attribution des adresses IP

Selon l' IPv6 Address Allocation and Assignment Policy, disponible sur le site de l'IANA (iana.org), la distribution des espaces d'adresses IPv6 s'effectue selon le shemas reproduit ci-dessous. Chaque organisme assigne un préfixe à hierarchiquement inférieur.



Notons qu'on parle de région « du monde ». Les registres régionaux concernent donc plusieurs pays. Suivant les RFC 2928 et 2373bis, l'IANA a distribué les adresses aux RIRs (Registres régionaux) existants à partir de 2001::/16 . Les utilisateurs finaux recevront généralement des plages d'adresses /48 (RFC3177). Cependant l'IANA se réserve le droit d'effectuer plus tard une redistribution d'adresses, après un retour d'expérience sur IP v6

Pour plus de détails, http://www.iana.org/ipaddress/IPv6-allocation-policy-26jun02 (en anglais).

Annexe 2 : Les RFC ayant trait à IPv6

Annexe 2.1 RFC 1550 (Décembre 1993)

Ce qui suit est un extrait de la RFC 1550, cité ici pour référence. Le résumé de cet extrait se trouve dans la partie « historique du protocole IPv6 », Première partie, chapitre 3.

5. Engineering considerations

In past discussions the following issues have been raised as relevant to the IPng selection process. This list is in no particular order. Any or all of these issues may be addressed as well as any other topic that the author feels is germane, but do not exceed the 10 page limit, please.

5.1 Scaling - What is a reasonable estimate for the scale of the future data networking environment? The current common wisdom is that IPng should be able to deal with 10 to the 12th nodes.

5.2 Timescale - What are reasonable time estimates for the IPng selection, development and deployment process or what should the timeframe requirements be? This topic is being evaluated by the ALE working group and a copy of all white papers that express opinions about these topics will be forwarded to that group.

5.3 Transition and deployment - Transition from the current version to IPng will be a complex and difficult process. What are the issues that should be considered The TACIT working group will be discussing these issues and a copy of all white papers that express opinions about these topics will be forwarded to that group.

5.4 Security - What level and type of security will be required in the future network environment? What features should be in an IPng to facilitate security?

5.5 Configuration, administration and operation - As networks get larger and more complex, the day to day operational aspects become ever more important. What should an IPng include or avoid in order to minimize the effect on the network operators?

5.6 Mobile hosts - How important is the proliferation of mobile hosts to the IPng selection process? To what extent should features be included in an IPng to assist in dealing with mobile hosts?

5.7 Flows and resource reservation - As the data networks begin to get used for an increasing number of time-critical processes, what are the requirements or concerns that affect how IPng should facilitate the use of resource reservations or flows?

5.8 Policy based routing - How important is policy based routing? If it is important, what types of policies will be used? What requirements do routing policies and potential future global architectures of the Internet bring to IPng? How do policy requirements interact with scaling?

5.9 Topological flexibility - What topology is anticipated for the Internet? Will the current general topology model continue? Is it acceptable (or even necessary) to place significant topological restrictions on interconnectivity of networks?

5.10 Applicability - What environment / marketplace do you see for the application of IPng? How much wider is it than the existing IP market?

5.11 Datagram service - Existing IP service is "best effort" and based on hop-by-hop routed datagrams. What requirements for this paradigm influence the IPng selection?

5.12 Accounting - How important a consideration should the ability to do accounting be in the selection of an IPng? What, if any, features should be included in an IPng to support accounting functions?

5.13 Support of communication media - IPv4 can be supported over most known types of communications media. How important is this same flexibility to an IPng?

5.14 Robustness and fault tolerance - To the extent that the Internet built from IPv4 has been highly fault tolerant, what are ways that IPng may avoid inadvertent decrease in the robustness (since some things may work despite flaws that we do not understand well). Comment on any other ways in which this requirement may affect the IPng.

5.15 Technology pull - Are there technologies that will pull the Internet in a way that should influence IPng? Can specific strategies be developed to encompass these?

5.16 Action items - suggested charges to the directorate, working groups or others to support the concerns or gather more information needed for a decision.

6. Security Considerations

This RFC raises no security issues, but does invite comment on the security requirements of IPng

Annexe 2.2 : Liste des RFC

Il faut ajouter à cette liste les RFC traitant de l'adressage, beaucoup plus récentes (par exemple, les adresses Loopback et unspecified sont définies dans la RFC 3513, citée dans le rapport).

Les RFC suivantes traitent de l'en-tête IPv6, ses objectifs et la manière de cohabiter avec IPv4

RFC3056: Connection of IPv6 Domains via IPv4 Clouds February 2001 B. Carpenter, K. Moore
RFC3053: IPv6 Tunnel Broker January 2001 A. Durand, P. Fasano, I. Guardini, D. Lento
RFC3041: Privacy Extensions for Stateless Address Autoconfiguration in IPv6 January 2001 T. Narten, R. Draves
RFC3019: IP Version 6 Management Information Base for The Multicast Listener Discovery Protocol January 2001 B. Haberman, R. Worzella
RFC2976: The SIP INFO Method October 2000 S. Donovan
RFC2974: Session Announcement Protocol October 2000 M. Handley, C. Perkins, E. Whelan
RFC2928: Initial IPv6 Sub-TLA ID Assignments September 2000 R. Hinden, S. Deering, R. Fink, T. Hain
RFC2921: 6BONE pTLA and pNLA Formats (pTLA) September 2000 B. Fink
RFC2894: Router Renumbering for IPv6 August 2000 M. Crawford
RFC2893: Transition Mechanisms for IPv6 Hosts and Routers August 2000 R. Gilligan, E. Nordmark
RFC2874: DNS Extensions to Support IPv6 Address Aggregation and Renumbering July 2000 M. Crawford, C. Huitema
RFC2848: The PINT Service Protocol: Extensions to SIP and SDP for IP June 2000 S. Petrack, L. Conroy
RFC2772: 6Bone Backbone Routing Guidelines February 2000 R. Rockell, R. Fink
RFC2767: Dual Stack Hosts using the Bump-In-the-Stack Technique (BIS) February 2000 K. Tsuchiya, H. Higuchi, Y. Atarashi
RFC2766: Network Address Translation - Protocol Translation (NAT-PT) February 2000 G. Tsirtsis, P. Srisuresh
RFC2765: Stateless IP/ICMP Translation Algorithm (SIIT) February 2000 E. Nordmark
RFC2740: OSPF for IPv6 December 1999 R. Coltun, D. Ferguson, J. Moy
RFC2732: Format for Literal IPv6 Addresses in URL's December 1999 R. Hinden, B. Carpenter, L. Masinter
RFC2711: IPv6 Router Alert Option. October 1999 C. Partridge, A. Jackson
RFC2710: Multicast Listener Discovery (MLD) for IPv6. October 1999 S. Deering, W. Fenner, B. Haberman
RFC2675: IPv6 Jumbograms. August 1999 D. Borman, S. Deering, R. Hinden
RFC2590: Transmission of IPv6 Packets over Frame Relay Networks. May 1999 A. Conta, A. Malis, M. Mueller
RFC2553: Basic Socket InterfaceExtensions for IPv6. March 1999 R. Gilligan, S. Thomson, J. Bound, W. Stevens
RFC2545: Use of BGP-4 Multiprotocol Extensions for IPv6 Inter-Domain Routing March 1999 P. Marques, F. Dupont
RFC2543: RFC 2543 - Session Initiation Protocol March 1999 M. Handley, H. Schulzrinne, E. Schooler, J. Rosenberg

RFC2529: Transmission of IPv6 over IPv4 Domains without Explicit Tunnels. March 1999 B. Carpenter, C. Jung
RFC2526: Reserved IPv6 Subnet Anycast Addresses. March 1999 D. Johnson, S. Deering
RFC2509: IP Header Compression over PPP. February 1999 M. Engan, S. Casner, C. Bormann
RFC2508: Compressing IP/UDP/RTP Headers for Low-Speed Serial Links. February 1999 S. Casner, V. Jacobson
RFC2507: IP Header Compression. February 1999 M. Degermark, B. Nordgren, S. Pink
RFC2497: Transmission of IPv6 Packets over ARCnet Networks. January 1999 I. Souvatzis
RFC2492: IPv6 over ATM Networks. January 1999 G. Armitage, P. Schulter, M. Jork
RFC2491: IPv6 over Non-Broadcast Multiple Access (NBMA) networks. January 1999 G. Armitage, P. Schulter, M. Jork, G. Harter
RFC2474: Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers. December 1998 K. Nichols, S. Blake, F. Baker, D. Black
RFC2473: Generic Packet Tunneling in IPv6 Specification. December 1998 A. Conta, S. Deering
RFC2472: IP Version 6 over PPP. December 1998 D. Haskin, E. Allen
RFC2471: IPv6 Testing Address Allocation. December 1998 R. Hinden, R. Fink, J. Postel (deceased)
RFC2470: Transmission of IPv6 Packets over Token Ring Networks. December 1998 M. Crawford, T. Narten, S. Thomas
RFC2467: Transmission of IPv6 Packets over FDDI Networks. December 1998 M. Crawford
RFC2466: Management Information Base for IP Version 6: ICMPv6 Group. December 1998 D. Haskin, S. Onishi
RFC2465: Management Information Base for IP Version 6: Textual Conventions and General Group. December 1998 D. Haskin, S. Onishi
RFC2464: Transmission of IPv6 Packets over Ethernet Networks. December 1998 M. Crawford
RFC2463: Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification. December 1998 A. Conta, S. Deering
RFC2462: IPv6 Stateless Address Autoconfiguration. December 1998 S. Thomson, T. Narten
RFC2461: Neighbor Discovery for IP Version 6 (IPv6). December 1998 T. Narten, E. Nordmark, W. Simpson
RFC2460: Internet Protocol, Version 6 (IPv6) Specification. December 1998 S. Deering, R. Hinden
RFC2454: IP Version 6 Management Information Base for the User Datagram Protocol December 1998 M. Daniele
RFC2452: IP Version 6 Management Information Base. December 1998 M. Daniele
RFC2450: Proposed TLA and NLA Assignment Rules. December 1998 R. Hinden
RFC2428: FTP Extensions for IPv6 and NATs. September 1998 M. Allman, S. Ostermann, C. Metz
RFC2406: IP Encapsulating Security Payload (ESP). November 1998 S. Kent, R. Atkinson
RFC2402: IP Authentication Header. November 1998 S. Kent, R. Atkinson
RFC2401: Security Architecture for the Internet Protocol. November 1998 S. Kent, R. Atkinson
RFC2375: IPv6 Multicast Address Assignments. July 1998 R. Hinden, S. Deering
RFC2374: An IPv6 Aggregatable Global Unicast Address Format. July 1998 R. Hinden, M. O'Dell, S. Deering
RFC2373: IP Version 6 Addressing Architecture. July 1998 R. Hinden, S. Deering
RFC2365: Administratively Scoped IP Multicast July 1998 D. Meyer
RFC2327: SDP: Session Description Protocol April 1998 M. Handley, V. Jacobson
RFC2292: Advanced Sockets API for IPv6. February 1998 W. Stevens, M. Thomas
RFC2185: Routing Aspects of IPv6 Transition. September 1997 R. Callon, D. Haskin
RFC2081: RIPng Protocol Applicability Statement. January 1997 G. Malkin
RFC2080: RIPng for IPv6. January 1997 G. Malkin, R. Minnear
RFC2030: Simple Network Time Protocol (SNTP) Version 4 for IPv4, IPv6 and OSI. October 1996 D. Mills
RFC1981: Path MTU Discovery for IP version 6. August 1996 J. McCann, S. Deering, J. Mogul

RFC1924: A Compact Representation of IPv6 Addresses. April 1996 R. Elz
RFC1888: OSI NSAPs and IPv6. August 1996 J. Bound, B. Carpenter, D. Harrington, J. Houldsworth, A. Lloyd
RFC1887: An Architecture for IPv6 Unicast Address Allocation. December 1995 Y. Rekhter, T. Li, Editors
RFC1886: DNS Extensions to support IP version 6. December 1995 S. Thomson, C. Huitema
RFC1884: IP Version 6 Addressing Architecture December 1995 R. Hinden, S. Deering, Editors
RFC1881: IPv6 Address Allocation Management. December 1995 IAB, IESG
RFC1829: The ESP DES-CBC Transform. August 1995 P. Karn, P. Metzger, W. Simpson
RFC1828: IP Authentication using Keyed MD5. August 1995 P. Metzger, W. Simpson
RFC1810: Report on MD5 Performance June 1995 J. Touch
RFC1809: Using the Flow Label Field in IPv6. June 1995 C. Partridge
RFC1752: The Recommendation for the IP Next Generation Protocol. January 1995 S. Bradner, A. Mankin
RFC 1550 : White paper solicitation

Annexe 3 - Résolution DNS IPv6 avec BIND 9

Bind 9 supporte IPv6. Il est possible de le configurer pour résoudre les adresses IPv6

Résolution d'adresses en utilisant les enregistrement de type AAAA
L'enregistrement AAAA est similaire au A en IPv4.

Il spécifie une adresse entière en un seul enregistrement. Par exemple,

```
$ORIGIN example.com.
host          3600    IN      AAAA    3ffe:8050:201:1860:42::1
```

Les enregistrements PTR
Exemple:

```
; IPv6 (REVERSE) ADDRESS LOOKUPS

; VT RFC1897 prefix is 5F05:2000/32
secondary       0.0.0.2.5.0.f.5.IP6.INT         dns/IPv6/r
primary         8.5.0.0.0.0.8.5.d.a.0.8.0.0.0.2.5.0.f.5.IP6.INT dns/IPv6/ee/r
primary         8.2.0.0.0.0.8.2.d.a.0.8.0.0.0.2.5.0.f.5.IP6.INT dns/IPv6/cs/r

; Provide address-to-hostname resolution for the loopback address
primary               0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.IP6.INT
dns/IPv6/loopback

forwarders      128.173.4.247 128.173.4.113
```

Exemple de configuration BIND :

fichier named.conf :

```
zone "F.E.4.8.8.2.0.3.0.0.2.1.e.f.f.3.ip6.int" {
        type master;
        file "/etc/namedb/3ffe_1200_3028_84EF___64.rev";
        also-notify {
                     194.117.194.82;
                    };
        };
```

fichier F.E.4.8.8.2.0.3.0.0.2.1.e.f.f.3.ip6.int :

```
; 3ffe:1200:3028:825d::/64

$TTL 1H

$ORIGIN b.5.2.8.8.2.0.3.0.0.2.1.e.f.f.3.ip6.int.
0 IN SOA ns0.teraii.dhs.org. admin.teraii.dhs.org. (
                        2001021024
                        43200
                        900
                        2419200
                        86400 )
   IN    NS    ns0.teraii.dhs.org.
   IN    NS    ns1.teraii.dhs.org.
$ORIGIN 0.b.5.2.8.8.2.0.3.0.0.2.1.e.f.f.3.ip6.int.

1.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0 IN PTR routeur.IPv6.teraii.dhs.org.
7.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0 IN PTR stupid-host.IPv6.teraii.dhs.org.
5.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0 IN PTR station1.IPv6.teraii.dhs.org.
21.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0 IN PTR ftp.IPv6.teraii.dhs.org.
80.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0 IN PTR www.IPv6.teraii.dhs.org.
6667.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0 IN PTR irc.IPv6.teraii.dhs.org.
```

Annexe 4 : signets Internet

Ces signets complètent la bibliographie.

Organismes officiels :

http://www.ietf.org Internet Engineering Task Force
http://www.iana.org Internet Assigned Number Authority
http://www.renater.fr Réseau National de Télécommunications pour la Technologie, l'Enseignement et la Recherche . Ce site diffuse de nombreux cours.
Http://www.cru.fr – Comité Réseau des Universités
http://www.rfc-editor.org – Site archivant les RFC

Réseaux, généralités :

Cours de M. Philippe Wender, enseignant au Cnam de Rouen
http://webcnam.insa-rouen.fr/intranet/rezo.htm
Cours de M. François Laissus
http://www.laissus.fr/cours/cours.html
Site généraliste
http://www.routage.org

IPv6 :

Lors de mes recherches sur Internet, j'ai découverts que de nombreux sites présentaient des informations érronées ou basées sur des RFC obsolètes. Aussi il est recommandé la plus grande attention à la personne effectuant des recherches.

Site réalisé par Solaris, filiale de Sun Microsystems :
http://playground.sun.com/pub/ipng/html/ipng-main.html
Cours de l'UREC (Unité Réseau du CNRS) :
http://www.urec.fr/cours/Reseau/IPv6/
Rapport sur les enjeux d'IPv6, mission Telecom du Ministère des Finances
http://www.telecom.gouv.fr/documents/IPv6/body.htm
Cours IPv6 de M. Philippe Dax, professeur à l'Ecole Nationale Supérieure des Telecom
http://www.ntdios.org/winsock/IPv6.htm
Site amateur traitant d' IPv6 sour Linux/Unix
http://www.docmaster.org/articles/linux120.htm
Site Olympus Zone sur IPv6:
http://www.olympus-zone.net/page_1033_fr_Blue.html
Page de l'IETF sur la proposition IPAE – IP Adress Encapsulation (cf Historique, I-3)
http://www.ietf.org/html.charters/OLD/ipae-charter.html)
SIPP Overview, site internet du groupe de travail sur le standart SIPP (cf Historique, I-3)
http://town.hall.org/trendy/sipp/sipp-overview.html
RFC 1475 décrivant TP/IX
http://www.ietf.org/rfc/rfc1475.txt
Page IBM sur TCP/IP :
http://www.auggy.mlnet.com/ibm/3376c216.html

Multicast :

Cours Multicast diffusé par RENATER
http://www.renater.fr/Video/IPv6/Multicast-Oct2003/
Cours IGMP de M. Michael Borella, Professeur à Northwestern University, Chicago
http://www.borella.net/mike/MITP432/06%20igmp.pdf

Videolan :

Site officiel : http://www.videolan.org et plus particulièrement http://www.videolan.org/doc/vlc-user-guide/fr/vlc-user-guide-fr.txt

Installation d' IPv6 :

Microsoft Windows 2000 : http://msdn.microsoft.com/downloads/sdks/platform/tpIPv6.asp
Linux IPv6 HowTo : http://www.redhat.com/mirrors/LDP/HOWTO/Linux%2BIPv6-HOWTO/index.html
Cisco : http://www.cisco.com/warp/public/732/Tech/IPv6/

Annexe 5 : Acronymes

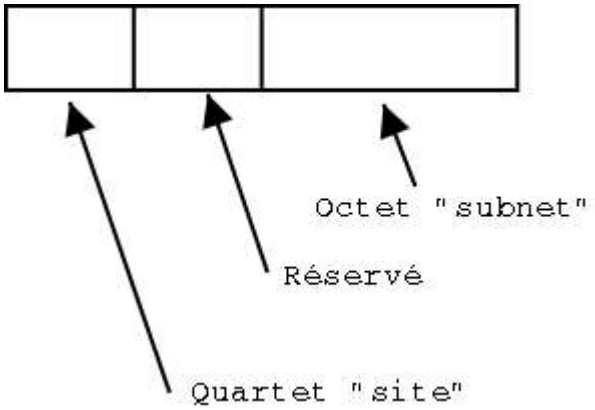
BGP – Border Gateway Protocol
CGMP – Cisco Group Multicast Protocol
DHCP – Dynamic Host Configuration Protocol
DNS – Domain Name System
DoD – Departement of Defense (USA)
DVMRP – Distance Vector Multicast Routing Protocol
ICMP – Internet Control Message Protocol
IETF – Internet Engineering Task force
IGMP – Internet Group Management Protocol
IP – Internet Protocol
MBGP – Multicast Border Gateway Protocol
MLD Snooping – Multicast Listener Discovery Snooping
MOSPF – Multicast Extension of OSPF (Open Shortest Path First)
MSDP – Multicast Source Discovery Protocol
OSPF - Open Shortest Path First
OSI – Open System Internconnexion
PIM – Protocol Independant Multicast
RFC – Request For Comments
SAP – Session Announcement Protocol
SIPP – Simple Internet Protocol Plus
TCP – Transmission Control Protocol
ToS – Type of Service
TTL – Time To Live
UDP – User Datagram Protocol
VLC – VideoLan Client

Annexe 6 : Adressage IPv6 à l'Université de Caen : découpage en sous-réseaux

Une réunion s'est tenue au département d'informatique le 17 janvier 2003. Y ont participé Davy Gigan, Vincent Lenouvel, François Moret, Véronique Robert, Hélène Rousset et Jean Saquet. Le but était de réfléchir à un plan d'adressage pouvant satisfaire aussi bien les besoins actuels d'expérimentations que le futur plan d'adressage global de l'université. La solution proposée ci-dessous ne doit bien entendu pas être considérée comme définitive, toute critique constructive est la bienvenue.

Plan d'adressage proposé:

Nous disposons des 16 bits du champ SLA, les valeurs possibles vont de 0000 à FFFF.
Les bits de poids forts doivent être le plus proche de la racine d'un point de vue routage (car l'adressage est hiérarchique et géographique). L'idée est la suivante :



Plan d'adressage possible :

- 00 00 à 00 1F Caen info soit 32 sous-réseaux
- 00 20 à 00 2F Caen criuc soit 16 sous-réseaux
- 00 30 à 00 3F Caen IUT soit 16 sous-réseaux . .
- 10 00 à 10 0F Caen pedagogie soit 16 sous-réseaux
- ...
- 50 00 à 50 0F Cherbourg école ingénieur
- 60 00 à 60 0F St Lo
- 70 00 à 70 0F Alençon
- ...
- F0 00 à FF FF Réservé pour un usage futur

Le deuxième quartet n'est pour l'instant pas utilisé mais permettra d'étendre soit la portée du 1er quartet, si par exemple 16 sites ne suffisent pas, soit la portée des 3 et 4ème quartets si l'on désire plus de subnets à l'intérieur d'un site.

Le premier quartet est donc utilisé pour le site : de 0 à 4 site de Caen, 5 site de Cherbourg, 6 St Lo, 7 Alençon, ...
La mise en oeuvre d'un tel plan d'adressage nécessitera une coopération avec Vikman, car les différents sites de l'université de Caen sont reliés au réseau régional en plusieurs points

Annexe 7: Quelques applications Multicast.

- **Catalogue des sessions**
  - **sdr** (Session Directory next generation) : annonce les conférences en cours ou à venir avec un calendrier .
  - lancement automatique des applications.
- **Audioconférence**
  - **vat** (Visual Audio Tool) : audio-conférence .
  - **rat** (Robust Audio Tool) : audio-conférence .
  - **fphone** (Free Phone) : téléphone Internet .
- **Vidéoconférence**
  - **vic** (Video Internet Conferencing) : video-conférence.
  - **ivs** (Inria Videoconferencing System) : video/audio-conférence .

- **Tableau blanc partagé**
  - **wb** (White-Board) : tableau blanc partagé .
- **Texte partagé**
  - **nt** (Network Text editor) : éditeur de texte partagé .
- **Images disséminées**
  - **imm** (Images Multicast) : diffusion d'images JPEG .

On peut y ajouter plusieurs applications non-dédiées, mais supportant le multicast : Videolan ...

Bibliographie

Monographies / Etudes :

IPv6 :

L.Peterson – Computer Network : a system approach – ed Morgan Kaufmann – 2003 *
M.Blanchett – Config IPv6 Cisco IOS – ed Morgan Kaufmann - 2003 *
J.Davies – Understanding IPv6 – ed Microsoft Press – 2002 *
G.Cizauet –IPv6 Théorie et Pratique – ed O'Reilly – 2002

Multicast :
G.Cizauet – IPv6 Théorie et Pratique – ed O'Reilly – 2002
B.Williamson –Developping IPv6 Multicast Networks – ed Cisco Press – 2003 *

Configuration :

P.Albitz et Cricket Liu –DNS et BIND – ed O'Reilly – 2003
M.Welsch – La Bible Linux – ed Micro Applications – 2003

Articles

N.Noury, V.Rialle, Habitats Intelligents pour la Santé, systèmes et équipements dans Techniques de l'Ingénieur, 04/2003
L.Toutain G.Rubino, Routage dans les réseaux Internet dans Techniques de l'Ingénieur, 02/2000 (chapitre consacré à IPv6 d'intérêt « historique »)

* ouvrage en anglais.