

# **Implémentation IPv6 dans un contexte PME/Ecole**

**Travail de Bachelor réalisé en vue de l'obtention du Bachelor HES**

par :

**Philippe Cattin**

Conseiller au travail de Bachelor :

**Gérard Ineichen**

**Carouge, 15.11.2012**

**Haute École de Gestion de Genève (HEG-GE)**

**Filière Informatique de Gestion**

# Déclaration

Ce travail de Bachelor est réalisé dans le cadre de l'examen final de la Haute école de gestion de Genève, en vue de l'obtention du titre (...). L'étudiant accepte, le cas échéant, la clause de confidentialité. L'utilisation des conclusions et recommandations formulées dans le travail de Bachelor, sans préjuger de leur valeur, n'engage ni la responsabilité de l'auteur, ni celle du conseiller au travail de Bachelor, du juré et de la HEG.

« J'atteste avoir réalisé seul(e ) le présent travail, sans avoir utilisé des sources autres que celles citées dans la bibliographie. »

Fait à Carouge, le 15.11.2012

Philippe Cattin

## Remerciements

Tout d'abord, je tiens à remercier profondément M. Gérard Ineichen pour m'avoir proposé ce sujet très intéressant, d'avoir été d'une grande aide dans les différentes difficultés rencontrées, pour sa disponibilité tout au long du projet et pour la relecture du dossier.

Merci également à M. André Seydoux pour son aide et ses conseils pour la réalisation du prototype à base de NAT-PT.

Je tiens encore à remercier les relecteurs et relectrices qui ont eu la patience de lire et de corriger mon travail. J'ajoute également à ma liste M. Michael Pilloud et M. Miguel Tavares, avec qui j'ai travaillé dans la même salle, et qui m'ont motivé à avancer et soutenu tout au long du projet.

Pour terminer, je remercie sincèrement Mlle. Noémie Lepdor qui a su me supporter durant toute la période du travail et poser un regard critique et pertinent sur les problèmes rencontrés.

# Résumé

Aujourd'hui, l'adressage IPv6 est encore sous-utilisé. La diversité et la complexité des différents protocoles utilisables, les problèmes de comptabilité ainsi que les différentes technologies proposées pour la mise en place d'un environnement IPv6 rebutent plus d'une entreprise ou d'une école. Mais, face à une proche pénurie d'adresses IPv4, ces acteurs qui n'ont pas encore fait un pas en avant sont forcés d'évoluer au plus vite.

Dans un premier temps il est expliqué dans les grandes lignes les différentes évolutions de l'adressage IPv6 ainsi que ses avantages par rapport à son grand-frère IPv4. Puis il est question des différentes solutions qui existent aujourd'hui pour réaliser une transition en IPv6, ainsi que leurs tenants et aboutissants.

Ensuite, une partie pratique se concentre sur la mise en place de deux prototypes implémentant chacun une technologie différente. Le premier utilise un fournisseur de service qui propose une solution à base de tunnel pour obtenir une connectivité au monde IPv6. Le second opte pour une solution à base de translation d'adresse avec le protocole NAT-PT, qui permet à un îlot IPv6 pur de communiquer avec le monde IPv4.

Pour ces prototypes, il faut que l'installation dispose d'une sortie sur internet pleinement fonctionnelle. Il faut également que le réseau dispose de tous les composants propres à son fonctionnement comme un serveur DNS. Différentes solutions pour l'adressage doivent être mises en place également, comme la configuration automatique ou encore l'adressage statique. Le réseau doit être opérationnel pour accueillir autant des machines sous Windows 7 que celles sous Linux (Ubuntu 12.04).

Ce travail fournit donc une base théorique ainsi que deux applications pratiques pour la mise en place d'une solution simple et adaptée au cas d'une petite entreprise ou encore d'une école souhaitant implémenter une solution en IPv6.

**Mots clés :** IPv6, IPv4, 6to4, transition, migration, tunnel broker, tunnel, NAT-PT, double-pile, NAT64, DNS64, translation, proxy, implémentation.

# Table des matières

Déclaration.....	i
Remerciements .....	ii
Résumé .....	iii
Table des matières.....	iv
Liste des Figures.....	vii
Introduction .....	1
1. L'IPv4 à bout de souffle.....	2
1.1 Raréfaction et marché de l'IPv4.....	3
2. Déploiement et implantation d'IPv6 .....	4
2.1 Historique .....	4
2.1.1 World IPv6 Day .....	5
2.1.2 Implantation en Suisse.....	6
3. IPv6, quels avantages ? .....	7
3.1 Nombre d'adresses disponibles .....	7
3.2 NAT supprimé.....	7
3.3 Mobilité.....	7
3.4 Optimisation des en-têtes de paquet .....	8
3.5 Protocole IPsec natif.....	8
4. Protocoles et technologies de transition.....	9
4.1 La double-pile.....	9
4.2 Techniques de tunnel .....	10
4.2.1 Tunnels statiques.....	10
4.2.2 Tunnels 6to4 .....	11
4.2.3 Tunnels 6rd (IPv6 rapid déploiement) .....	12
4.2.4 Tunnels brokers .....	12
4.2.5 Teredo.....	13
4.3 Techniques de translation.....	15
4.3.1 NAT-PT .....	15
4.3.1.1 DNS-ALG .....	17
4.3.2 NAT64/DNS64 .....	17
4.3.2.1 DNS64 .....	17
4.3.2.2 NAT64 .....	18
4.3.2.3 Implémentations possibles .....	19
4.4 Serveurs mandataires (proxies) .....	19
5. Mise en place de la partie pratique .....	20
5.1 Choix des prototypes .....	20
5.2 Matériel utilisé .....	20
5.2.1 Routeurs .....	21
5.2.2 Commutateurs .....	21
5.2.3 Ordinateurs .....	21

5.2.4	Serveur .....	21
<b>5.3</b>	<b>Installation .....</b>	<b>21</b>
5.3.1	Sortie vers l'extérieur .....	21
5.3.2	Serveur DNS .....	21
5.3.3	Réseau local .....	22
5.3.4	Réseau global .....	22
<b>5.4</b>	<b>Procédures de configuration et tests.....</b>	<b>23</b>
5.4.1	Configuration des machines en IPv6 .....	23
5.4.2	Tests .....	23
5.4.2.1	Fonctionnement du réseau local .....	23
5.4.2.2	Sortie internet .....	23
5.4.2.3	Messagerie .....	24
<b>6.</b>	<b>Prototypes avec « Tunnel broker » .....</b>	<b>25</b>
<b>6.1</b>	<b>Implantation avec « Hurricane Electric » .....</b>	<b>25</b>
6.1.1	Enregistrement.....	25
6.1.2	Création d'un tunnel.....	25
6.1.3	Informations du tunnel .....	26
6.1.4	Configuration du tunnel.....	27
6.1.5	Adresse IP dynamique .....	28
6.1.6	Adressage du réseau local.....	28
6.1.7	Configuration des machines .....	29
6.1.8	Tests de fonctionnement.....	30
6.1.9	Configuration finale du tunnel .....	31
<b>6.2</b>	<b>Implantation avec SixXs .....</b>	<b>32</b>
6.2.1	Enregistrement.....	32
6.2.2	Création du tunnel.....	32
6.2.3	Configuration du tunnel.....	33
6.2.4	Configuration du réseau local .....	33
6.2.5	Configuration des machines .....	34
6.2.6	Tests de fonctionnement.....	34
6.2.7	Configuration finale du tunnel .....	35
<b>6.3</b>	<b>Implantation avec Gogo6 .....</b>	<b>36</b>
6.3.1	Enregistrement.....	36
6.3.2	Création du tunnel.....	36
6.3.2.1	Installation et configuration du programme GogoCLIENT .....	37
	37	
6.3.2.2	Connexion et informations du tunnel.....	38
6.3.3	Configuration des machines .....	39
6.3.4	Configuration de la globalité du réseau .....	40
6.3.4.1	Configuration du deuxième sous-réseau .....	40
6.3.4.2	Configuration du routage statique .....	41
6.3.5	Tests de fonctionnement.....	41
6.3.6	Configuration du tunnel.....	42
<b>6.4</b>	<b>Conclusion des prototypes avec tunnel .....</b>	<b>43</b>
<b>7.</b>	<b>Prototype avec translation d'adresse NAT-PT .....</b>	<b>44</b>
<b>7.1</b>	<b>Adressage des sous-réseaux en IPv6.....</b>	<b>44</b>
7.1.1	Méthode de configuration des VLAN .....	44

<b>7.2</b>	<b>Configuration du NAT-PT sur le routeur .....</b>	<b>45</b>
7.2.1	<i>Translation des adresses IPv6 en IPv4.....</i>	45
7.2.2	<i>Communication IPv6/IPv4.....</i>	46
7.2.2.1	Configuration de l'adresse du DNS.....	46
7.2.2.2	Configuration de l'IPv4-mapped.....	46
<b>7.3</b>	<b>Tests et fonctionnement.....</b>	<b>47</b>
7.3.1	<i>Communication d'un hôte IPv6 vers un hôte IPv4 .....</i>	47
7.3.2	<i>Sortie d'un hôte IPv6 sur internet.....</i>	48
<b>7.4</b>	<b>Configuration du prototype.....</b>	<b>48</b>
<b>7.5</b>	<b>Conclusion du prototype.....</b>	<b>49</b>
	<b>Conclusion.....</b>	<b>51</b>
	<b>Bibliographie .....</b>	<b>53</b>
	<b>Annexe 1 Configuration du tunnel Hurricane Electric .....</b>	<b>56</b>
	<b>Annexe 2 Script de démarrage d'AICCU .....</b>	<b>59</b>
	<b>Annexe 3 Fonctionnement IPv6 avec tunnel SixXs.....</b>	<b>60</b>
	<b>Annexe 4 Fonctionnement IPv6 avec tunnel Freenet6 .....</b>	<b>61</b>
	<b>Annexe 5 Configuration du Routeur NATPT .....</b>	<b>62</b>
	<b>Annexe 6 Expérimentations avec DNS64 et NAT64 .....</b>	<b>65</b>
	<b>Annexe 7 Configuration HedgeRouteur pour tunnel Freenet6.....</b>	<b>67</b>

# Liste des Figures

Figure 1	Disponibilités IPv4 des RIR .....	2
Figure 2	Logo World IPv6 Launch.....	5
Figure 3	Statistiques IPv6 Cisco pour la Suisse.....	6
Figure 4	En-tête de paquet IPv6.....	8
Figure 5	Fonctionnement réseau double pile.....	9
Figure 6	Encapsulation d'un paquet IPv6 dans un tunnel.....	10
Figure 7	Encodage 6to4.....	11
Figure 8	Fonctionnement réseau 6rd.....	13
Figure 9	Fonctionnement Tunnel Broker.....	14
Figure 10	Fonctionnement Teredo.....	15
Figure 11	Fonctionnement du NAT-PT.....	17
Figure 12	Fonctionnement de DNS-ALG.....	18
Figure 13	Résolution enregistrement A.....	19
Figure 14	Serveur proxy.....	20
Figure 15	Installation du réseau.....	23
Figure 16	Informations du tunnel Hurricane Electric.....	27
Figure 17	Schéma du tunnel Hurricane Electric.....	32
Figure 18	Informations du tunnel SixXs.....	33
Figure 19	Schéma du tunnel SixXS.....	36
Figure 20	Fenêtres de configuration du programme gogoCLIENT.....	38
Figure 21	Statut du tunnel Freenet6.....	40
Figure 22	Schéma du tunnel Freenet6.....	43
Figure 23	Schéma NAT-PT.....	49



# Introduction

La progression d'IPv6 en Europe commence à faire réfléchir plus d'une entreprise ou d'une école. Plus d'une d'entre elles souhaitent faire la transition, mais comment et avec quels outils ?

De plus, plusieurs grandes questions se posent :

- Pourquoi passer à un nouveau protocole, alors que l'actuel fonctionne très bien ?
- Peuvent-elles garder leurs configurations actuelles en IPv4, tout en permettant d'assurer une communication avec le monde IPv6 ?
- Finalement, peuvent-elles se passer totalement de l'IPv4 ?

Ce travail a pour but de présenter l'état de l'art de l'adressage, de décrire les solutions de transition possibles, ainsi que de tester par le biais de prototypes deux solutions qui apporteront des réponses aux questions ci-dessus.

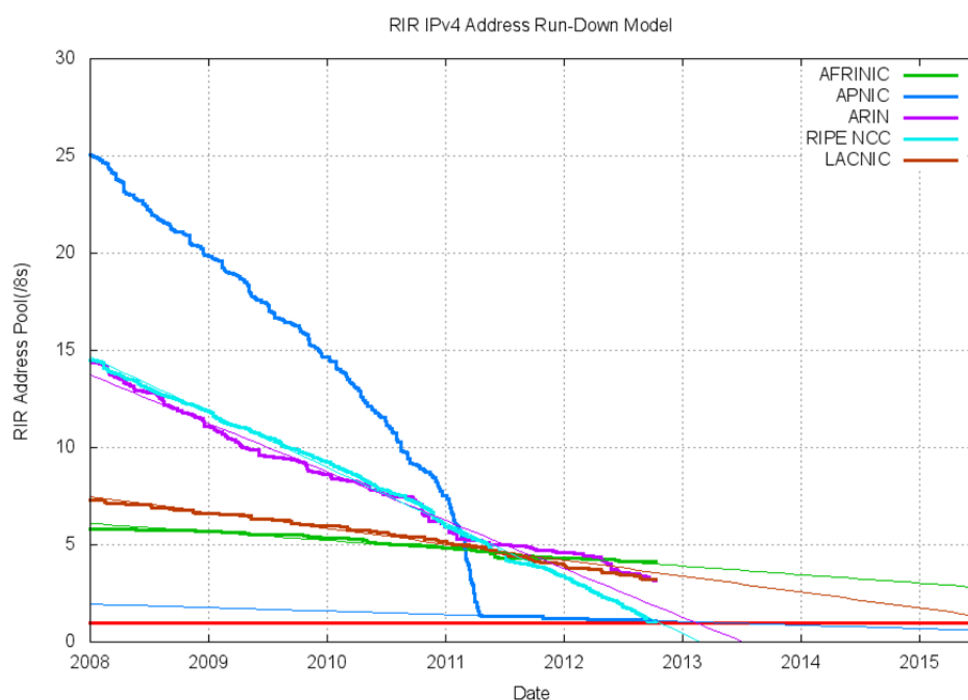
Depuis le début de ma formation, et c'est toujours le cas aujourd'hui, le domaine des réseaux m'a particulièrement intéressé et passionné. C'est donc tout naturellement que je me suis tourné vers un sujet dans ce domaine. Ce choix a été fait pour plusieurs raisons. Tout d'abord, il permettait de tenir à jour les connaissances acquises durant le cursus scolaire tout en y incluant une dose de nouveauté et d'apprentissage. L'IPv6 marquant un tournant dans l'histoire d'internet et des réseaux, il m'a paru important de m'y intéresser et de travailler dessus. Ensuite, la maîtrise de ce sujet m'a semblé également très utile pour un futur travail dans le domaine des réseaux, dans lequel des connaissances en IPv6 pourraient être requises ou appréciées. Face à une transition de plus en plus recommandées vers des réseaux en IPv6, le besoin en personnes qualifiées dans le sujet sera sûrement là. Le côté pratique du travail est également un gros atout, cela me permettra de me rendre compte de la réalité technique du sujet et des problèmes qui en découlent, et ne pas rester avec uniquement un bagage théorique.

# 1. L'IPv4 à bout de souffle

Annoncée depuis quelques années déjà, la pénurie d'adresses IPv4 en Europe semble se rapprocher à grands pas. En effet, le 14 septembre 2012, une lettre d'information<sup>1</sup> du RIR<sup>2</sup> RIPE NCC (chargé de la distribution des adresses IPv4 pour l'Europe), explique qu'il a distribué son dernier bloc d'adresses IPv4 disponible. Une réserve d'environ 17 millions d'adresses<sup>3</sup> est néanmoins encore à sa disposition mais une politique de distribution plus restrictive a été mise en place.

Au niveau mondial, et d'après les statistiques suivantes, on peut voir que le RIR nord-américain (ARIN) a encore du stock pour plus d'une année, alors que le RIR d'Asie et du Pacifique (APNIC) est déjà à court depuis plus d'une année.

**Figure 1**  
**Disponibilités IPv4 des RIR**



Source : <http://www.potaroo.net/tools/ipv4/index.html>

---

<sup>1</sup> Source : site web du RIPE NCC

<sup>2</sup> Registre Internet Regional, organisme qui alloue les blocs d'adresses IP dans sa zone géographique. Source : Wikipédia.

<sup>3</sup> Source : site web du RIPE NCC

## **1.1 Raréfaction et marché de l'IPv4**

En conséquence de cette proche pénurie, une politique de marché de l'IPv4 commence doucement à se mettre en place. De nouvelles politiques sont apparues, permettant notamment le transfert ou la vente d'adresses. Les RIR des régions d'Amérique du Sud et d'Afrique sont les seuls à ne pas autoriser ce type de pratiques pour l'instant.

En mars 2011, Microsoft a lancé les hostilités en rachetant 666'624 adresses à la société en faillite Nortel pour 7.5 millions de dollars<sup>4</sup>, soit 11.25\$ par adresse.

Une société de service américaine, « Hilco Streambank », a fait de la revente d'adresses IPv4 une de ses spécialités avec plusieurs grosses ventes réalisées sur des blocs de 65535 adresses (/16)<sup>5</sup>. Elle possède également en partenariat sa propre plateforme de revente sur internet.

Certaines bourses aux adresses sont également apparues, comme les sites <http://www.tradeipv4.com/> ou encore <http://ipv4marketgroup.com>.

Une étude récente de l' « Internet Governance Project »<sup>6</sup>, basée sur les ventes dans la région nord-américaine, montre que dans la première moitié 2012 26% des allocations d'adresses IPv4 se sont faites par le biais du marché, contre seulement 5% dans l'année 2011.

On le voit clairement, l'adresse IPv4 est en train de devenir un bien rare, qui commence déjà à se monnayer.

Reste à voir quelle sera l'évolution de ce nouveau phénomène. Le cours des adresses va-t-il flamber, va-t-il y avoir spéculations ? Est-ce que certaines entreprises en cours de création seront laissées sur le bas-côté et ne pourront pas, ou à prix cher, obtenir leur sésame ?

Seul l'avenir nous le dira.

---

<sup>4</sup> Voir l'article à cette adresse : <http://www.zdnet.fr/actualites/revente-d-adresses-ipv4-microsoft-debourse-75-millions-de-dollars-39759394.htm>

<sup>5</sup> Source : site internet d'Hilco Streambank

<sup>6</sup> Source : étude « Dimensionning the Elephant : An empirical analysis of the IPv4 number market »

## 2. Déploiement et implantation d'IPv6

### 2.1 Historique

Le protocole IPv4 a commencé son développement dans les années 1970, et suite aux expérimentations et améliorations successives a été finalement standardisé en 1981 dans la RFC 791<sup>7</sup>. D'abord utilisé par l'armée, puis progressivement par les grandes instances scientifiques et les universités, ce n'est qu'au début des années nonante que le protocole connaît une croissance exponentielle avec l'ouverture de l'internet à tout à chacun, entreprise ou privé.

En 1992, le protocole est victime de son succès. L'attribution trop généreuse des plages d'adresses, notamment celles de classe B ( $2^{16}$  adresses), fait que le nombre d'adresses disponibles diminue drastiquement. De plus, les routeurs commencent à atteindre les limites de leur capacité à cause de la taille sans cesse grandissante de la table de routage d'internet, qu'ils doivent supporter.

Dès l'année suivante et dans les années qui suivirent, des solutions sont développées pour enrayer la consommation d'adresses. Dans ces mesures citons les plages d'adresses de classe B rationnées, le CIDR<sup>8</sup> qui fait son apparition, l'aménagement de plages d'adresses privées (RFC 1918<sup>9</sup>) ainsi que l'installation de systèmes comme le NAT<sup>10</sup> ou encore le proxy.

Dès 1993 également, au vu des problèmes actuels, des travaux de recherche pour l'élaboration d'un nouveau protocole sont lancés. Différents groupes de travail y participent, et chacun essaie d'imposer sa solution. C'est finalement le protocole « IPng » qui a été retenu, renommé et standardisé en IPv6 dans la RFC 1883 (Internet Protocol, version 6)<sup>11</sup> en décembre 1995.

---

<sup>7</sup> Source : <http://www.ietf.org/rfc/rfc791.txt>

<sup>8</sup> Classless Inter-Domain Routing (CIDR), mécanisme qui a permis de segmenter les plages d'adresses de l'époque en de plus petites, permettant une meilleure gestion de l'attribution de celles-ci. Pour plus d'informations vous pouvez consulter le lien suivant : <http://www.commentcamarche.net/contents/internet/le-cidr>.

<sup>9</sup> <http://tools.ietf.org/html/rfc1918>

<sup>10</sup> Network Address Translation, ou traduction d'adresse réseau. Fait correspondre une ou plusieurs adresses IP internes non routables à une adresse IP externe routable. Source : Wikipédia

<sup>11</sup> Source : <http://tools.ietf.org/html/rfc1883>

### 2.1.1 World IPv6 Day

Le 8 juin 2011 a eu lieu le premier « World IPv6 Day »<sup>12</sup>. Cet événement, créé par l'internet society (ISOC), a permis d'effectuer un premier test grandeur nature l'IPv6 sur les principales plateformes internet. Des sites comme Facebook, Google, Youtube, ainsi que plus de milles autres sites participant à l'événement ont ainsi activé l'IPv6 sur leur plateforme durant 24 heures pour mesurer la faisabilité du déploiement du protocole.

**Figure 2**

**Logo World IPv6 Launch**



Source : Site du World IPv6 Launch<sup>13</sup>

Le 6 juin 2012, toujours sous le parrainage de l'internet society, a eu lieu le « World IPv6 Launch Day »<sup>14</sup>. Cette fois-ci, les fournisseurs d'accès, les fabricants d'équipements réseaux et les sociétés d'internet participant à l'événement ont activé définitivement le protocole IPv6 sur leurs plateformes et équipements, afin de le rendre disponible au plus grand nombre. Cette journée a été marquée d'une pierre rouge et correspond au lancement officiel d'IPv6 dans le monde.

En Suisse, l'association « Swiss IPv6 council »<sup>15</sup>, qui se veut en faveur de l'intégration de l'IPv6 au sein du territoire, a participé à l'événement. Elle coopère étroitement avec les fournisseurs d'accès, les fabricants d'équipements et les entités concernées par le déploiement de la technologie. Les grands fournisseurs d'accès (Swisscom, Sunrise ou encore Switch) en font partie.

---

<sup>12</sup> Source : site de l'internet society

<sup>13</sup> Logo à disposition sur le site du World IPv6 Launch

<sup>14</sup> Source : site du World IPv6 Launch

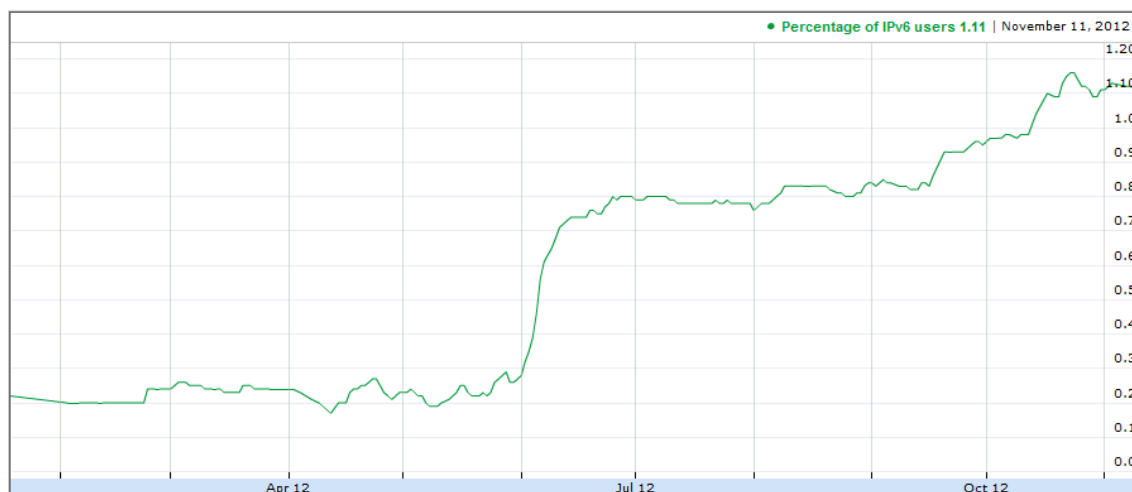
<sup>15</sup> Voir leur site internet pour plus d'informations : <http://www.swissipv6council.ch/fr>

### 2.1.2 Implantation en Suisse

Malgré les récents efforts, en Suisse et d'après les estimations fournies par Cisco, il y aurait seulement entre 1.11% d'utilisateurs possédant une connexion IPv6.

Figure 3

#### Statistiques IPv6 Cisco pour la Suisse



Source : 6lab Cisco<sup>16</sup>

A l'intérieur du territoire, le fournisseur d'accès Switch (qui fournit les universités et hautes écoles) propose et encourage à une connectivité IPv6 depuis 2004<sup>17</sup>. L'EPFL ou encore l'UNIL ont déjà déployé la technologie sur leurs réseaux. Du côté des offres pour les entreprises et écoles, Swisscom propose déjà la compatibilité IPv6 avec certains de leurs services. D'après une interview d'un employé<sup>18</sup>, d'ici à 2013 l'ensemble de leurs services seront à jour. L'opérateur Sunrise propose également depuis 2011 une solution IPv6 destinée à ses abonnés<sup>19</sup>. D'autres, comme VTX<sup>20</sup>, étudient ou testent sérieusement le protocole.

Il semble qu'un panel suffisant d'offres existe aujourd'hui, autant pour les entreprises que les écoles. Il ne leur reste qu'à évoluer vers l'IPv6.

<sup>16</sup> Programme Cisco de statistiques d'utilisation d'IPv6 dans le monde ou un pays en particulier. Chiffres changeants dans le temps, visiter le lien suivant pour obtenir les chiffres actuels : <http://6lab.cisco.com/stats/index.php>.

<sup>17</sup> Voir les service IPv6 de Switch : <http://www.switch.ch/network/services/ipv6/index.html>

<sup>18</sup> Source : Article web Swisscom

<sup>19</sup> Source : Article web Sunrise

<sup>20</sup> Voir <http://ipv6.vtx.ch/ipv6/index.php?p=3>

### 3. IPv6, quels avantages ?

Outre le fait qu'il n'y a pratiquement plus d'adresses IPv4 disponibles, il y a d'autres raisons qui poussent naturellement à passer à IPv6. En voici les principales :

#### 3.1 Nombre d'adresses disponibles

Avec l'arrivée des tablettes, des téléphones mobiles, de la domotique dans les maisons, la demande en adresse est exponentielle. On est passé de  $2^{32}$  (4 milliards) adresses IPv4 à  $2^{128}$  (~340 milliards de milliards de milliards de milliards) adresses IPv6. Ce nombre astronomique répond donc amplement à cette demande.

#### 3.2 NAT supprimé

L'utilisation quasi systématique du NAT et des adresses privées pour recouvrir au manque d'adresses disponibles a créé différents problèmes. Les communications pair à pair (comme la visio-conférence, la voix sur IP) sont bridées, les routeurs ont une surcharge énorme de travail pour réaliser cette tâche. L'IPv6 permet, dans la grande majorité des cas, de supprimer ce mécanisme grâce au nombre d'adresses disponibles et de laisser place à une connectivité pair à pair plus fluide, rapide et pratique.

La disparition du NAT peut faire peur à certains au niveau de la sécurité, car les machines du réseau sont toutes directement accessibles depuis l'extérieur. Même si ce dernier a disparu, la sécurité peut être très bien assurée par un pare-feu à état qui fera le même travail sur le routeur, avec bien évidemment une sécurité supplémentaire sur chaque machine.

#### 3.3 Mobilité

Le protocole Mobile IP<sup>21</sup>, déjà présent mais peu performant en IPv4, a été mis au goût du jour et permet à un appareil de rester connecté même en changeant de réseau. Cette fonctionnalité est tout à fait adaptée au monde d'aujourd'hui, avec de plus en plus d'appareils nomades, comme les téléphones, tablettes et ordinateurs portables.

---

<sup>21</sup> Source : Wikipédia

### 3.4 Optimisation des en-têtes de paquet

Les en-têtes de paquet IP ont été allégés pour IPv6.

Le champ *checksum* a été retiré, réduisant ainsi la charge de travail de ces derniers. Les champs d'options sont relégués dans des en-têtes d'extension (*Next header*) qui seront seulement contrôlés par le destinataire final, réduisant une nouvelle fois le travail des routeurs.

Un nouveau champ, *Traffic class* (classe de trafic) permet de définir un type de traitement spécial pour les paquets. S'il s'agit de données audio ou vidéo par exemple, le traitement ne sera pas le même que pour un trafic normal, et leur priorité sera plus grande pour éviter toute congestion sur le réseau. Le champ *Flow label* (Identificateur de flux) permet de définir une appartenance à un paquet et de mettre un œuvre des services comme la qualité de service. Enfin, la taille des paquets (MTU<sup>22</sup>) est fixe et d'un minimum de 1280 octets.

**Figure 4**

**En-tête de paquet IPv6**

Version	Traffic class	Flow label	
Payload length		Next header	Hop limit
Source address			
Destination address			

Source : Documentation Oracle<sup>23</sup>

### 3.5 Protocole IPsec natif

Le protocole de sécurité internet (IPsec)<sup>24</sup> est implanté de base avec IPv6. Il permet une sécurité à l'aide entre autres du chiffrement (cryptographie) et de contrôle d'intégrité des données. Il n'est néanmoins pas obligatoire de configurer, mais cet outil est fourni pour une sécurité accrue du protocole.

---

<sup>22</sup> Maximum Transmission Unit, taille maximale d'un paquet pouvant être transmis en une fois sur une interface. Source : Wikipédia

<sup>23</sup> Source : <http://docs.oracle.com/cd/E19957-01/820-2982/images/HeaderFormat.gif>

<sup>24</sup> Source : Wikipédia



## 4. Protocoles et technologies de transition

Dans ce chapitre, il est question d'expliquer les possibilités actuelles de transition. Elles peuvent être catégorisées en quatre parties :

- La double-pile
- Techniques de tunnel
- Techniques de translation
- Serveurs mandataires (proxies)

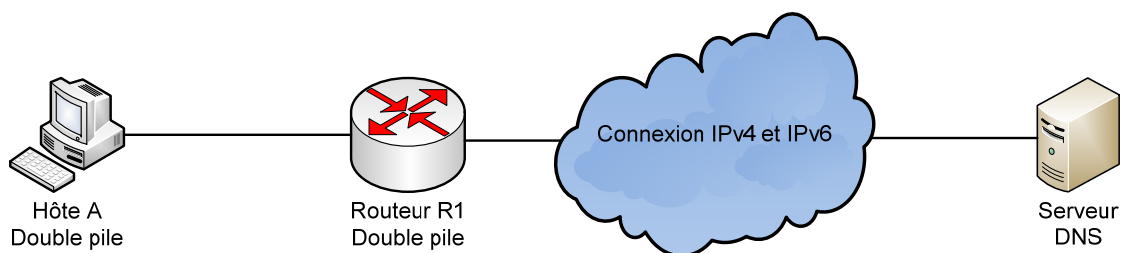
Ces technologies peuvent ou doivent être utilisées conjointement suivant les cas.

### 4.1 La double-pile

Largement déployée aujourd'hui, cette solution requiert que l'entièreté des équipements du réseau doive disposer en même temps d'un adressage IPv4 et d'un autre IPv6. La mise en place d'une double pile dans un réseau d'entreprise ou d'école peut être laborieuse. Il faudra gérer deux plans d'adressage, créer des règles séparées pour les pare-feu, maintenir des tables DNS séparées, etc.

**Figure 5**

#### Fonctionnement réseau double pile



Pour se connecter à un site web par exemple, le client va demander les enregistrements A et AAAA<sup>25</sup> au serveur DNS. Une tentative de connexion avec l'adresse IPv6 est d'abord réalisée, et si elle échoue une connexion IPv4 prend le relais.

---

<sup>25</sup> Un enregistrement A correspond à un enregistrement DNS IPv4.  
Un enregistrement AAAA correspond à un enregistrement DNS IPv6.

## 4.2 Techniques de tunnel

Un problème peut se poser lorsque le fournisseur d'accès ne possède pas d'infrastructure IPv6. Comment le réseau d'école ou d'entreprise IPv6 peut-il communiquer avec un serveur distant IPv6 alors que le lien qui les unit est en IPv4 ?

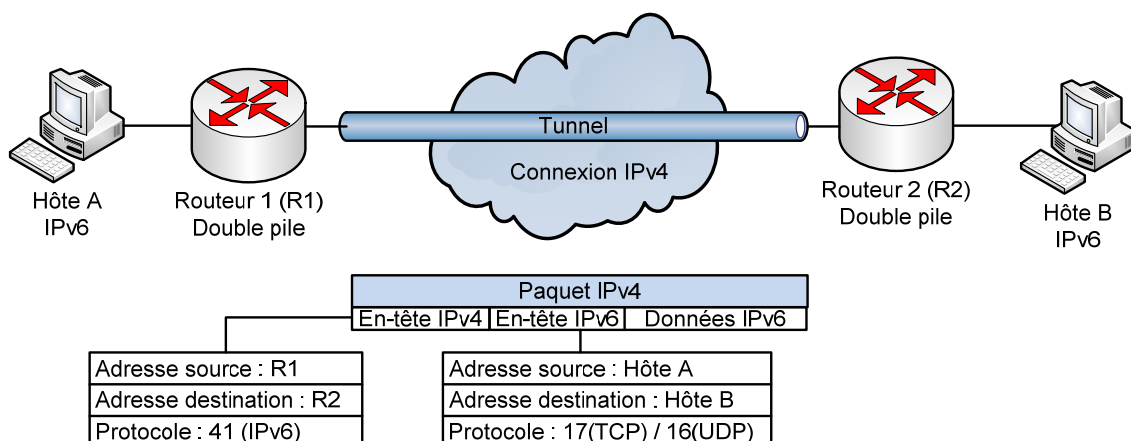
La solution est la création d'un tunnel entre ces deux sites.

Les paquets IPv6 vont être encapsulés dans un paquet IPv4 à l'entrée du tunnel. Pour cela, un en-tête IPv4 sera ajouté au paquet IPv6. Le paquet IPv6 se retrouvera donc dans la zone de données du paquet IPv4. Le paquet sera considéré comme tout à fait fonctionnel et routable.

A la sortie du tunnel, l'en-tête IPv4 sera retiré et le paquet IPv6 sera restitué et transmis à sa destination. Ce travail est effectué à l'aide du protocole IP n° 41 (RFC 2473)<sup>26</sup>, qui a été mis en place pour ce type de communication.

Figure 6

### Encapsulation d'un paquet IPv6 dans un tunnel



### 4.2.1 Tunnels statiques

Ce type de tunnel, correspondant à la figure 6 plus haut, est comparable à un tunnel classique très utilisé aujourd'hui pour relier des sites distants.

Il est néanmoins très limité, car il concerne seulement la connexion entre deux machines. Il faudrait alors en configurer un nouveau manuellement pour chaque connexion souhaitée. La navigation internet est donc ici impossible.

<sup>26</sup>

Source : <http://www.ietf.org/rfc/rfc2473.txt>

### 4.2.2 Tunnels 6to4

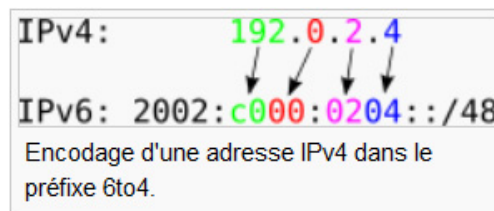
La technologie 6to4 repose sur l'encapsulation vue précédemment, mais y ajoute quelques fonctionnalités afin d'obtenir une création de tunnel automatique. Un préfixe IPv6 `2000::/16` est utilisé pour attribuer une adresse IPv6 à chaque poste du réseau. Une adresse 6to4 a le format suivant :

[2002] : [Adresse IPv4 publique en Hexadécimal] : [ID du sous-réseau] : [ID interface]

Tout d'abord, un préfixe global sera généré à partir de l'adresse IPv4 du routeur :

**Figure 7**

#### Encodage 6to4



Source : Wikipedia, article « 6to4 »<sup>27</sup>

Avec ce préfixe on pourra configurer les adresses 6to4 des postes du sous-réseau, de façon automatique (DHCPv6, auto-configuration) ou de façon statique. Chaque poste aura donc dans son adresse 6to4 l'adresse du routeur lui étant associée.

Lorsqu'une connexion doit s'établir entre deux routeurs 6to4, le premier va extraire l'adresse IPv4 du routeur de destination (à partir de l'adresse IPv6 de l'hôte de destination) et, ainsi, il peut assurer la communication avec celui-ci. Si une connexion veut s'effectuer entre un routeur 6to4 et un hôte n'intégrant pas ce protocole, un relais 6to4 peut être introduit pour assurer la communication. Ces relais possèdent tous l'adresse unicast 192.88.99.1 (pour être atteint facilement) ou peuvent être configurés manuellement.

Les problèmes inhérents à cette technologie sont les relais. En effet, la qualité de la connexion dépend alors de leur proximité et de leur disponibilité. Il n'existe pas non plus de politique de sécurité et de vérification du contenu ; de plus aucune vérification n'est faite sur la légitimité du relais 6to4, celui-ci pouvant être utilisé à mauvais escient. Pour ces raisons et d'autres, une requête de l'IETF pour déprécier ce protocole a été proposée en avril 2011.

<sup>27</sup>

Source: [http://upload.wikimedia.org/wikipedia/commons/9/96/6to4\\_convert\\_address.svg](http://upload.wikimedia.org/wikipedia/commons/9/96/6to4_convert_address.svg)

### 4.2.3 Tunnels 6rd (IPv6 rapid déploiement)

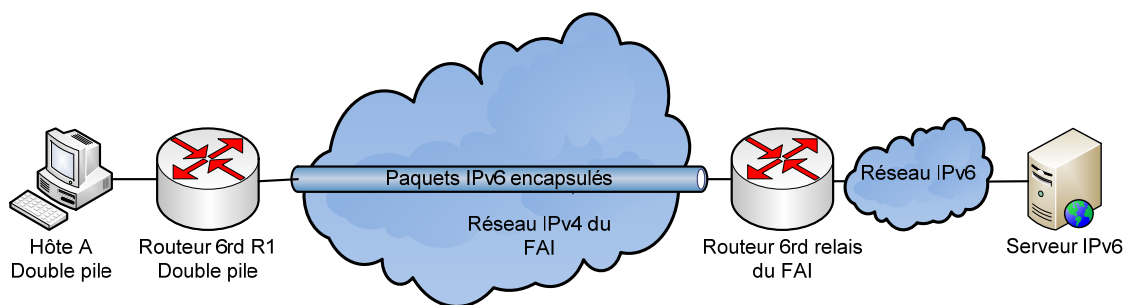
Pour pallier aux problèmes posés par les relais 6to4, une solution a été trouvée par M. Rémi Despres, ingénieur français en informatique et télécommunications. Ces relais sont rattachés aux fournisseurs d'accès, qui assurent leur fonctionnement et leur intégrité.

Le préfixe 6to4 est remplacé par un préfixe unique pour chaque fournisseur d'accès, ce qui limite le trafic au réseau de celui-ci et ainsi assure une meilleure sécurité.

Dans le cas d'une communication entre un hôte IPv6 et un serveur IPv6, le routeur R1 va encapsuler les paquets dans de l'IPv4 et le transmettre directement au routeur relais du FAI, qui lui va se charger de le décapsuler et de le router à sa destination.

**Figure 8**

#### Fonctionnement réseau 6rd



La technologie 6rd, standardisée par l'IETF dans la RFC 5969<sup>28</sup>, a notamment été installée à grande échelle par l'opérateur Free en France.

### 4.2.4 Tunnels brokers

Ce type de tunnel est proposé par des sociétés ou des communautés tierces. Ce service peut être payant ou gratuit selon le fournisseur. Certains fournisseurs limitent également leur service à un pays, une région, ou à un type d'utilisateur.

Comme avec le protocole 6rd, c'est la société qui gère le tunnel, les relais et l'attribution des adresses IPv6 par le biais de leurs propres préfixes. L'architecture est ici basée sur un modèle client-serveur.

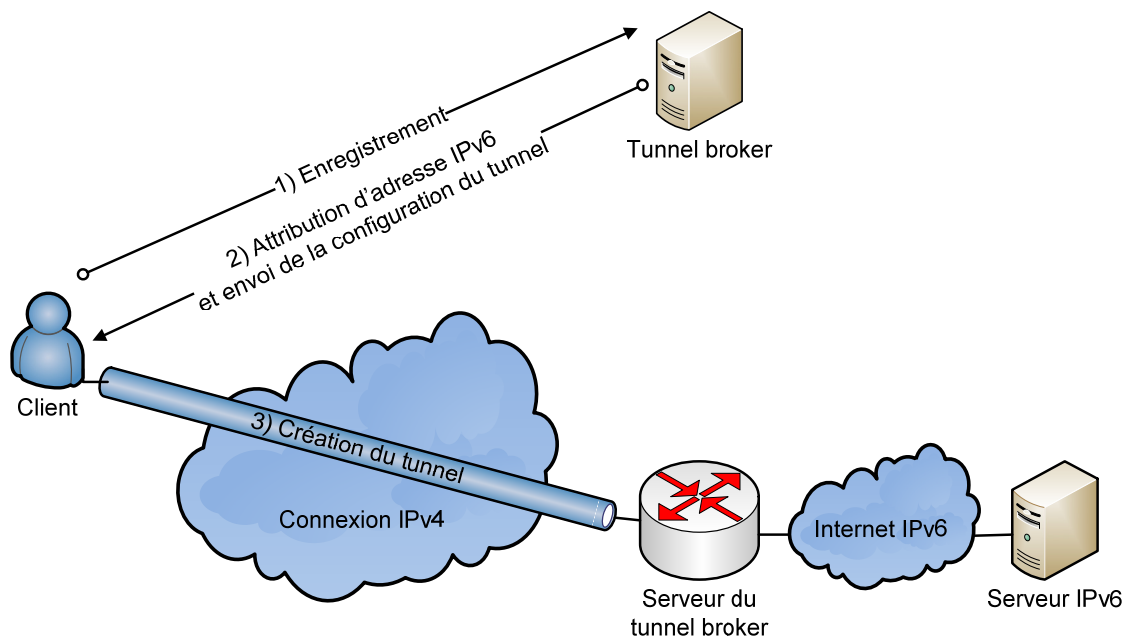
---

<sup>28</sup> Source : <http://tools.ietf.org/html/rfc5969>

Ici le client va négocier avec le fournisseur de tunnel pour obtenir les informations dont il a besoin pour créer le tunnel. Ces informations arrivent soit sous la forme d'un fichier de configuration, soit sous la forme d'un programme à installer sur la machine cliente (routeur ou ordinateur). De plus, il recevra une adresse IPv6 comme point d'entrée ainsi qu'un pool d'adresses s'il désire configurer le reste de son réseau.

**Figure 9**

### Fonctionnement Tunnel broker



Le tunnel, une fois configuré, est statique. C'est le serveur qui va agir comme relais, décapsuler le trafic et l'acheminer vers la destination IPv6.

Cette solution peut permettre une certaine sécurité, car les tunnels peuvent être chiffrés. Mais, en contrepartie, on dépend totalement du fournisseur et de l'état de leur matériel, des pannes de serveurs ou des maintenances.

#### 4.2.5 Teredo

Le protocole Teredo, développé par Microsoft<sup>29</sup>, a la particularité de faire l'encapsulation des adresses IPv6 avec le protocole UDP et donc de permettre aux machines derrière un NAT de pouvoir communiquer avec l'extérieur. En effet, dans les autres cas de tunnel, les routeurs doivent gérer l'encapsulation avec le protocole 41 (6to4, 6rd).

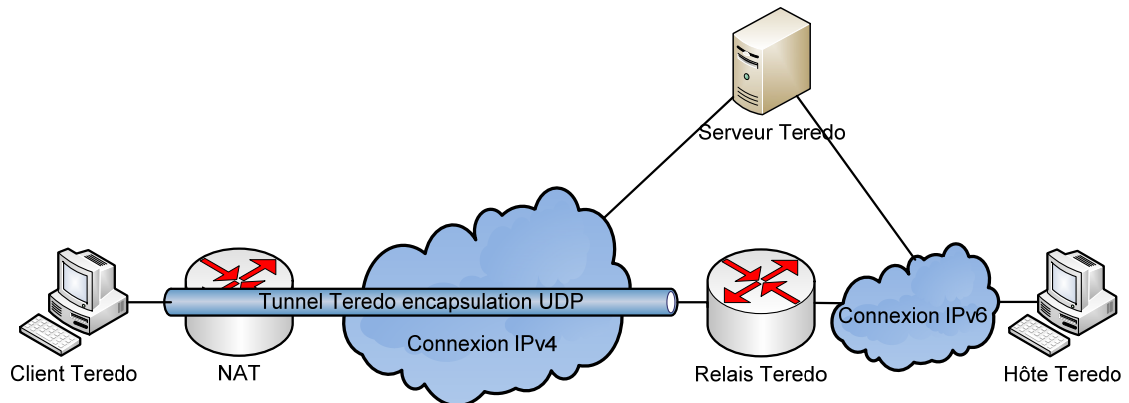
<sup>29</sup> Source : Documentation Microsoft

Ces machines peuvent être incompatibles avec cette encapsulation (celle avec le protocole 41) dans le cas du NAT, ou certains pare-feu peuvent bloquer le processus également.

Le préfixe utilisé est le `2001::/32`. Pour obtenir son adresse IPv6, le client devra la configurer avec l'aide du serveur Teredo. Celui-ci va définir automatiquement le type de NAT derrière lequel se trouve le client et s'y adapter pour permettre son franchissement.

**Figure 10**

**Fonctionnement Teredo**



Comme le montre le schéma suivant, Teredo fonctionne sous un modèle client/serveur, avec un relais pour accéder aux ressources IPv6. Le serveur est là pour négocier les informations de connexion entre les hôtes et garder celle-ci en état. Le relais quant à lui s'occupe uniquement de l'acheminement du trafic.

### **4.3 Techniques de translation**

Ce type de technique repose sur le même principe de NAT actuel, sauf qu'il permet à un réseau interne purement IPv6 de communiquer avec le monde IPv4 en réalisant une traduction entre ces deux mondes. Les réseaux étant encore majoritairement en IPv4, il est nécessaire de devoir garder contact avec eux.

Dans le cas d'un réseau interne avec une double pile, si la machine à contacter est en IPv6, la connexion IPv6 sera utilisée ; sinon, les mécanismes de translation seront utilisés. L'implantation d'une telle technologie a néanmoins son avantage, celui d'être débarrassé, plus ou moins selon les techniques décrites plus bas, des restrictions et obligations liées à l'adressage IPv4 et donc d'évoluer dans un environnement purement IPv6.

#### **4.3.1 NAT-PT<sup>30</sup>**

Le protocole de translation NAT-PT fonctionne comme le NAT actuel et souffre des mêmes limitations que ce dernier. Le fonctionnement est le suivant : des paquets envoyés par un hôte IPv6 vers un hôte IPv4 devront avoir leur adresse source et de destination changées en IPv4 (par le routeur officiant le NAT-PT), et inversement pour assurer la communication.

Comme le NAT, il existe plusieurs types de translation possible :

- Statique : A une adresse IPv4 correspondra une adresse IPv6 et vice-versa. Une règle de translation doit être fixée pour chaque hôte.
- Dynamique : Un pool d'adresses IPv4 est disponible pour les translations. A chaque adresse IPv4 correspond un hôte IPv6 sortant.
- PAT (Port Address Translation) : Permet à de multiples adresses IPv6 de correspondre à une adresse IPv4 en utilisant le numéro de port.

Pour des problèmes de sécurité, de stabilité et d'autres, cette technologie a été dépréciée par l'IETF en 2007 dans la RFC 4966<sup>31</sup>. C'est la première technologie mise au point pour ce type de cas.

---

<sup>30</sup> Network Address Translation and Protocol Translation

<sup>31</sup> Source : <http://tools.ietf.org/html/rfc4966>

Un préfixe, fixé à `2001::/96` (préfixe de base, pouvant être remplacé par n'importe quel autre préfixe) permet aux hôtes IPv6 de contacter un hôte IPv4. Chaque hôte IPv4 sera vu par le réseau IPv6 comme ceci :

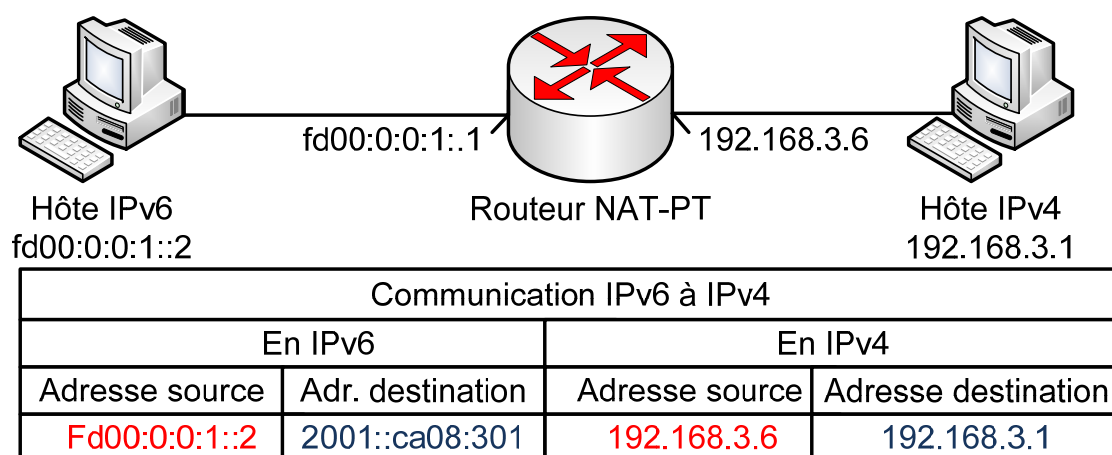
`2001::<IPv4 en hexadécimale>`

Par exemple, l'hôte IPv4 avec l'adresse `192.168.3.1` sera vu par un hôte IPv6 comme ayant l'adresse `2001::c0a8:301`. Une règle de NAT doit être instaurée entre les deux adresses.

Le mécanisme appelé IPv4-Mapped<sup>32</sup> permet de se passer de la configuration d'une règle de NAT et de s'adresser à n'importe quel hôte IPv4 avec l'adresse préfixée. Pour ce faire, quand le routeur reçoit un paquet d'un hôte IPv6, il va reconnaître que le préfixe utilisé est celui du NAT-PT et retirer les 32 derniers bits qui correspondent à l'adresse IPv4 de destination de l'adresse IPv6 de destination.

**Figure 11**

### Fonctionnement du NAT-PT



Dans ce cas, le routeur va garder dans sa table les correspondances entre les adresses afin d'assurer la communication bidirectionnelle entre les deux hôtes.

<sup>32</sup> Source : Documentation IBM - IPv4-mapped IPv6 addresses  
Source : Wikipédia

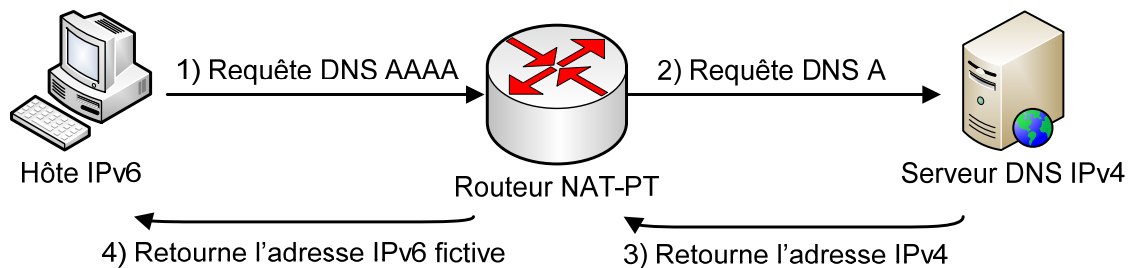


#### 4.3.1.1 DNS-ALG<sup>33</sup>

Ce mécanisme est utilisé conjointement au NAT-PT pour assurer le service DNS au réseau IPv6. Il permet la traduction des réponses DNS IPv4 embarquées dans les paquets IPv4 en réponses DNS IPv6.

Figure 12

#### Fonctionnement de DNS-ALG



L'hôte IPv6 commence par émettre une requête AAAA au DNS IPv4. Le routeur, interceptant le paquet, va traduire celle-ci en requête d'adresse IPv4. Le serveur va alors retourner un enregistrement IPv4 du site internet au routeur, qui lui va ajouter le préfixe à celle-ci avec l'aide du DNS-ALG afin de créer une requête DNS AAAA fictive et la transmettre à l'hôte. Ce dernier pourra alors directement se connecter à la page internet grâce au mécanisme IPv4-mapped.

#### 4.3.2 NAT64/DNS64

Le couplage des protocoles NAT64 et DNS64 est la solution la plus actuelle au problème de communication entre les machines IPv6 et IPv4. Ces technologies succèdent au protocole NAT-PT qui a été dépréciée par l'IETF. Les deux sont interdépendantes et nécessaires pour réaliser la communication.

##### 4.3.2.1 DNS64

Le protocole DNS64 (RFC 6147)<sup>34</sup> agit comme un DNS normal. Il gère les deux cas de demande de résolution de nom de domaine, c'est-à-dire celles avec un nom de domaine IPv4 et celles avec un autre en IPv6 :

<sup>33</sup> DNS Application Layer Gateway

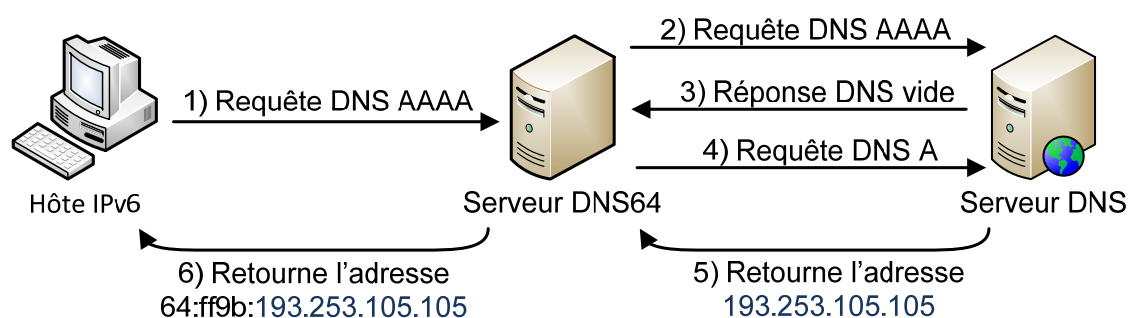
<sup>34</sup> Source : <http://tools.ietf.org/html/rfc6147>

Pour la résolution d'un nom de domaine IPv6, tout se passe avec ce protocole. Le client envoie sa requête au serveur DNS64, qui lui va interroger le serveur DNS pour obtenir l'adresse IPv6, et la renvoyer à l'hôte.

Dans le cas où le serveur DNS du site web possède seulement un enregistrement IPv4, c'est un peu différent. Le serveur DNS64 va essayer dans un premier temps d'obtenir un enregistrement AAAA. Le serveur web va alors retourner un message vide signalant qu'il ne possède pas d'adresse IPv6. Le serveur DNS64 va alors renvoyer une requête DNS A. Il va récupérer l'adresse et l'encapsuler à l'aide d'un algorithme dans une adresse IPv6. Une fois cela effectué, il va envoyer cette adresse IPv6 à l'hôte.

**Figure 13**

### Résolution enregistrement A



L'encapsulation ici utilise le même procédé que le NAT-PT, c'est-à-dire l'utilisation d'un préfixe, ici `64:ff9b/96` et de l'adresse IPv4 pour former une adresse IPv6. On peut clairement voir que le DNS64 a remplacé le mécanisme DNS-ALG présent dans le NAT-PT.

#### 4.3.2.2 NAT64<sup>35</sup>

Le protocole NAT64 (RFC 6146) est l'intermédiaire qui va réaliser la communication entre le réseau IPv6 et celui en IPv4. Vu que le serveur DNS64 a renvoyé une adresse IPv4 encapsulée dans une IPv6, le NAT64 devra se charger de décapsuler le paquet IPv4 pour le transmettre plus loin, mais également gérer le retour d'information et la ré-encapsulation pour les communications entrantes.

<sup>35</sup>

Source : <http://tools.ietf.org/html/rfc6146>

Comme son homologue NAT-PT, il va regarder le type de trafic qu'il reçoit. Si celui-ci embarque le préfixe *64:ff9b/96*, il va le décapsule et l'envoyer sur sa patte IPv4. Sinon, il envoie directement le trafic sur sa patte IPv6.

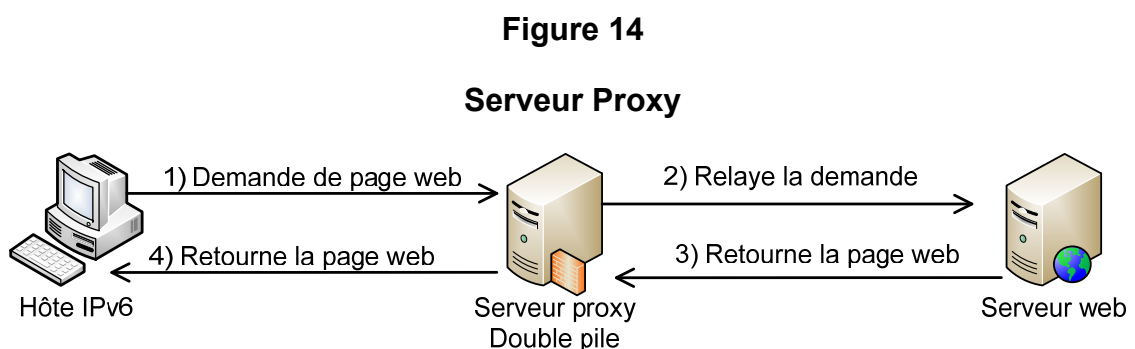
#### 4.3.2.3 Implémentations possibles

Il existe plusieurs solutions pour réaliser le protocole, en voici quelques-unes :

- Microsoft : Windows serveur 2012 avec Direct Access.
- Cisco : Le pare-feu ASA 5500 implémente le NAT64 et le DNS64<sup>36</sup>.
- Linux : Ecdysis et Tayga pour le NAT64  
Bind9 pour le DNS64

#### 4.4 Serveurs mandataires (proxies)

Pour que le réseau IPv6 puisse se connecter au monde IPv4, il est également possible d'installer un serveur proxy qui va servir d'intermédiaire entre ces deux mondes. Le fonctionnement est le suivant :



L'hôte IPv6 va envoyer sa demande de page web au proxy en s'adressant à lui en IPv6. Le proxy va aller chercher, avec sa connexion IPv4, la page demandée. Une fois obtenue, il va la retourner à l'hôte IPv6. Ce fonctionnement est également possible avec des hôtes IPv4 cherchant à obtenir du contenu web IPv6. Les serveurs mandataires peuvent également être utilisés pour mettre à disposition un site internet IPv4 aux réseaux IPv6.

Du côté des implémentations, il faut regarder du côté de *Windows Forefront Threat Management Gateway 2010* ou encore de *Squid* pour une solution libre.

---

<sup>36</sup> Source : Blog Cisco IPv6

## **5. Mise en place de la partie pratique**

### **5.1 Choix des prototypes**

Dans un premier temps, plusieurs solutions ont été retenues. Obtenir un adressage IPv6 de la part de Switch pour une sortie en IPv6, implémenter un tunnel grâce à un tunnel broker, le mécanisme de translation du réseau IPv6 pour une sortie en IPv4 avec les protocoles NAT64/DNS64, ainsi qu'une solution à base de proxy.

N'ayant pu obtenir d'adressage IPv6 dans les délais du travail de la part du fournisseur d'accès Switch, cette solution a dû être abandonnée. Après discussions avec M. Ineichen, deux manières de procéder ont été retenues parmi les choix restants pour expérimenter le protocole IPv6 :

- La première vise à obtenir une connectivité au monde IPv6 avec un réseau local en IPv6 à l'aide d'un fournisseur de tunnel.
- La seconde vise à obtenir une connectivité au monde IPv4 à partir d'un réseau local en IPv6. Dans un premier temps, les protocoles NAT64 et DNS64 ont été envisagés car étant les plus récents et les plus fonctionnels. Le matériel Cisco supportant ces protocoles est limité à la gamme des ASA 5500.

L'école ne possédant pas ce type de matériel, le choix s'est porté sur le protocole NAT-PT qui lui est supporté sur les routeurs à disposition.

Ces deux prototypes doivent répondre à des impératifs clairs. Les hôtes des réseaux locaux doivent disposer uniquement d'un adressage IPv6, pouvoir accéder à internet, communiquer entre eux, disposer des services d'un serveur DNS ainsi qu'implémenter les services de base que tout réseau d'entreprise ou d'école devrait avoir.

Le matériel à utiliser est celui de l'école, comprenant les équipements Cisco et les ordinateurs.

### **5.2 Matériel utilisé**

La première étape a été de faire avec les limitations du matériel à disposition. Il a fallu trouver les bonnes images systèmes compatibles IPv6 pour les routeurs. De plus, certaines images n'implémentent que partiellement le protocole et/ou ne permettent pas telle ou telle fonctionnalité.

### **5.2.1 Routeurs**

Le routeur de tête de réseau utilisé est un Cisco 2651XM avec l'IOS 12.3 (26).

Le routeur utilisé pour le NAT-PT est un Cisco 2651XM AdvanceEntrepriseK9 avec l'IOS 12.4(25).

### **5.2.2 Commutateurs**

Les commutateurs utilisés sont de la gamme Cisco 3550.

### **5.2.3 Ordinateurs**

Les ordinateurs sont ceux fournis par la HEG.

Les systèmes d'exploitation utilisés sont les suivants :

- Windows 7 professionnel avec Service Pack 1 installé.
- Ubuntu 12.04

### **5.2.4 Serveur**

Le serveur utilisé ici est un Windows serveur 2008 version entreprise.

## **5.3 Installation**

### **5.3.1 Sortie vers l'extérieur**

Pour obtenir une connectivité avec le monde extérieur et internet, le routeur de tête du réseau est relié à un routeur VDSL par le biais du réseau cantonal. La connexion VDSL a deux adresses de sortie. Une préférence de connexion a été fixée sur une de ces adresses pour éviter tout problème de routage asymétrique et également pour avoir toujours la même adresse pour la configuration du tunnel.

### **5.3.2 Serveur DNS**

Un serveur DNS a été ajouté aux rôles du serveur Windows 2008. L'installation est classique et suit les consignes données pendant les cours réseaux. Un domaine active directory, appelé *proto.heg*, a également été mis en place. Les adresses DNS du fournisseur VTX, qui fournit la connexion, ont été ajoutées dans le serveur DNS pour pouvoir surfer sur internet.

### 5.3.3 Réseau local

Pour respecter une situation d'entreprise ou d'école, des VLAN séparés (2,3 et 4) ont été configurés sur le switch1 :

- Le VLAN 2 est utilisé par le serveur DNS et le prototype NAT-PT.
- Les VLAN 3 et 4 sont utilisés pour le prototype avec tunnel.

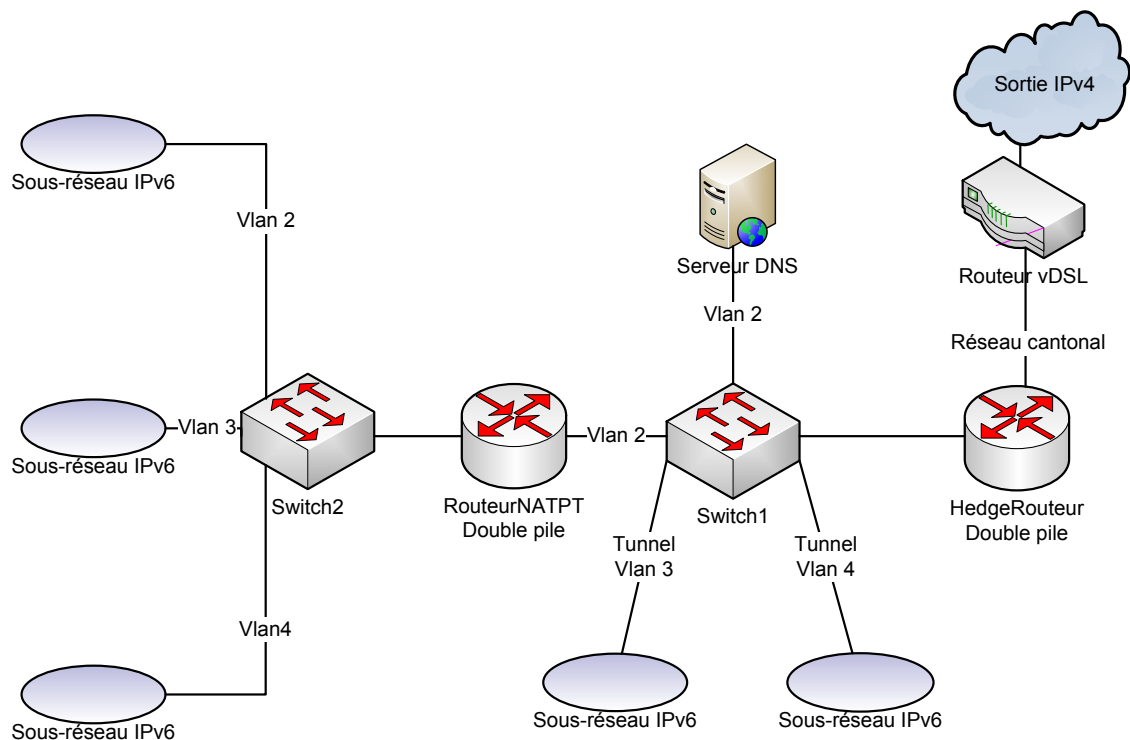
Le routeur *HedgeRouteur* se charge d'assurer le routage et les communications entre les VLAN.

De la même manière, trois VLAN ont été configurés sur le *switch2* pour représenter trois sous-réseaux différents, qui eux utilisent le NAT-PT.

### 5.3.4 Réseau global

Le schéma ci-dessous correspond à la finalité de l'installation réalisée durant le projet ; néanmoins, les équipements sont uniquement configurés pour obtenir une connectivité IPv4 à ce stade du dossier. Il a été mis ici afin de comprendre comment les éléments s'imbriquent et appréhender les problèmes et solutions proposées.

**Figure 15**  
**Installation du réseau**



## **5.4 Procédures de configuration et tests**

Pour s'assurer du bon fonctionnement des différents prototypes et des machines, il a été jugé nécessaire de présenter ici les étapes des différents tests qui seront effectuées, ceci afin d'éviter toute répétition. Sont également présentes les méthodes de configuration des machines. Merci de vous référer à cette partie en cas de besoin.

### **5.4.1 Configuration des machines en IPv6**

Pour que les stations se configurent avec le préfixe annoncé par le routeur, il faut activer l'auto configuration IPv6 sur celles-ci.

Pour les machines sous Windows 7, aller dans le menu de configuration de la carte réseau :

Démarrer → Panneau de configuration → Centre Réseau et partage → Connexion au réseau local → Propriétés
---

Deuxièmement, dans les propriétés du protocole IPv6, l'option « Obtenir une adresse IPv6 automatiquement » est choisie.

Un *ipconfig* dans une fenêtre de commande permet de vérifier que l'adresse IPv6 a bien été configurée.

Pour les machines sous linux, il faut également aller dans le menu de configuration réseau :

Paramètres système → Réseau → Filaire → Options
---

Pour les paramètres IPv6, choisir l'option « Automatique ».

### **5.4.2 Tests**

#### **5.4.2.1 Fonctionnement du réseau local**

Les machines doivent pouvoir pinger les autres machines, ainsi que le serveur DNS et le *HedgeRouteur* afin de vérifier les branchements et le routage. Les machines doivent pouvoir également effectuer un *nslookup* sur les équipements du domaine, et si possible entrer dans celui-ci. Ces tests permettent de valider que le réseau local fonctionne correctement.

#### **5.4.2.2 Sortie internet**

Pour les machines officiant avec le protocole NAT-PT, s'assurer qu'elles aient bien accès à l'internet IPv4.

Pour les machines implémentant le tunnel, deux tests permettent de s'assurer du bon fonctionnement sur internet :

- Le site [www.test-ipv6.com](http://www.test-ipv6.com) permet de valider la connexion IPv6. Cela permettra de confirmer que l'adresse de sortie est celle du tunnel.
- Quelques sites internet ayant activé IPv6 seront testés aléatoirement, afin d'éviter que les pages ne se retrouvent en cache et faussent les résultats. Une liste très complète de sites en IPv6 peut être trouvée à l'adresse suivante : [http://www.ipv6forum.com/ipv6\\_enabled/approval\\_list.php](http://www.ipv6forum.com/ipv6_enabled/approval_list.php). C'est en piochant dans cette liste que les tests seront effectués.

Plusieurs navigateurs seront testés, comme *Mozilla Firefox*, *Chrome* ou *Internet Explorer*. Ces trois navigateurs sont configurés de sorte à privilégier la connexion IPv6 à IPv4.

Finalement, il faut s'assurer que les deux prototypes marchent indépendamment l'un de l'autre.

#### **5.4.2.3 Messagerie**

Incontournable en entreprise ou dans une école, la messagerie sera également testée. Le serveur de messagerie web *Gmail* sera testé, car possédant une connectivité IPv4 et IPv6. Les tests pourront ainsi être faits sur les deux prototypes.



## 6. Prototypes avec « Tunnel broker »

Pour réaliser ce travail, il a d'abord fallu trouver quels étaient les fournisseurs de tunnel actuels et quelles étaient leurs implémentations dans le monde. Trois d'entre eux ont été retenus pour leur sérieux, leur implémentation en Europe, leur communauté active et les solutions qu'ils proposent. Il s'agit de *Hurricane Electric*<sup>37</sup>, *SixXS*<sup>38</sup>, et dans une moindre mesure *Gogo6*<sup>39</sup>, ce dernier manquant de communauté, de différents points d'accès dans le monde et de stabilité.

Le premier de la liste a été choisi pour commencer les expérimentations car étant le plus actif et le mieux documenté.

### 6.1 Implantation avec « Hurricane Electric »

#### 6.1.1 Enregistrement

Tout d'abord, il faut s'enregistrer sur le site internet <http://www.tunnelbroker.net/>. La création de compte requiert les informations privées de la personne ou de l'entreprise, et, après vérification, le compte est créé. Un email contenant les informations de connexion est finalement envoyé dans les heures qui suivent.

#### 6.1.2 Création d'un tunnel

Tout d'abord, il existe deux sortes de tunnels configurables :

- Création d'un tunnel BGP<sup>40</sup> : Ce type de tunnel est destiné aux entreprises, personnes ou écoles possédant déjà une ou plusieurs plage d'adresses IPv6 disponibles et souhaitant les utiliser au travers du tunnel.

Le protocole BGP est utilisé ici pour que ces plages d'adresses soient déclarées et puissent être reconnues et routées par le tunnel.

- Création d'un tunnel régulier : Ce type de tunnel va fournir au client une plage d'adresses IPv6 qu'il pourra utiliser à son gré s'il souhaite donner à son réseau une connectivité IPv6. De base, la création d'un tunnel permet seulement à l'hôte l'implémentant de profiter de la connexion IPv6.

---

<sup>37</sup> <http://www.tunnelbroker.net/>

<sup>38</sup> <http://www.sixxs.net/>

<sup>39</sup> <http://www.gogo6.com/freenet6/tunnelbroker>

<sup>40</sup> Border Gateway Protocol, protocole utilisé entre les fournisseurs d'accès et ses clients

C'est la deuxième solution qui va être utilisée pour ce prototype, ne possédant pas de plage d'adresses IPv6 disponible. A noter que l'on peut créer jusqu'à cinq tunnels différents avec ce fournisseur.

Pour créer le tunnel, deux choses sont requises :

- L'adresse IPv4 publique de sortie du réseau. Ici l'adresse du routeur vDSL.
- Faire son choix dans une liste de serveurs éparpillés dans le monde. Pour la Suisse il existe un tunnel à Zurich, sur lequel la préférence s'est portée par quiétude de proximité. Ce serveur va être le point de destination du tunnel.

### 6.1.3 Informations du tunnel

Une fiche récapitulative, ci-dessous, permet de voir que le tunnel a bien été créé et quels sont ses détails. L'adresse du client IPv4 peut être changée à tout moment. Des serveurs DNS sont également fournis pour assurer la navigation internet en IPv6.

Figure 16

#### Informations du tunnel *Hurricane Electric*

Tunnel Details	
<b>IPv6 Tunnel</b>   Example Configurations   Advanced	
Tunnel ID: 179779	<a href="#">Delete Tunnel</a>
Creation Date:	Oct 16, 2012
Description:	<input type="text"/>
<b>IPv6 Tunnel Endpoints</b>	
Server IPv4 Address:	216.66.84.42
Server IPv6 Address:	2001:470:1f12:101d::1/64
Client IPv4 Address:	212. [redacted]
Client IPv6 Address:	2001:470:1f12:101d::2/64
<b>Available DNS Resolvers</b>	
Anycasted IPv6 Caching Nameserver:	2001:470:20::2
Anycasted IPv4 Caching Nameserver:	74.82.42.42
<b>Routed IPv6 Prefixes</b>	
Routed /64:	2001:470:1f13:101d::/64
Routed /48:	2001:470:ccd8::/48 [X]

Source : Détails du tunnel de mon compte Hurricane Electric

Créer le tunnel ne suffit pas, il faut également le paramétrer sur l'équipement désiré.

### 6.1.4 Configuration du tunnel

Il existe deux choix possibles :

- Configurer le tunnel sur le routeur de sortie.
- Configurer le tunnel sur un équipement (ordinateur ou serveur) situé derrière le NAT du routeur de sortie. Il faut que le protocole 41 soit autorisé par le NAT.

N'ayant pas accès à la configuration du routeur vDSL, la deuxième solution a été adoptée en plaçant le tunnel sur le *HedgeRouteur*.

Pour cela, il faut commencer par activer le routage IPv6, désactivé par défaut, sur le routeur :

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

La configuration du tunnel en soit peut être trouvée sur la page du tunnel de l'utilisateur, sur son compte, sous l'onglet *Example Configurations*. Il existe un paramétrage pour une panoplie de systèmes d'exploitation différents (Windows XP/7, Linux, OpenBSD, Apple Airport, Cisco IOS,...).

Ici la configuration pour l'IOS Cisco a été prise et appliquée au routeur :

```
HedgeRouteur(config)#interface Tunnel0
HedgeRouteur(config-if)#no IP address
HedgeRouteur(config-if)#ipv6 enable
HedgeRouteur(config-if)#ipv6 address 2001:470:1f12:101d::2/64
HedgeRouteur(config-if)#tunnel source 172.18.67.190
HedgeRouteur(config-if)#tunnel destination 216.66.80.98
HedgeRouteur(config-if)#tunnel mode ipv6ip
HedgeRouteur(config-if)#no shutdown
HedgeRouteur(config-if)#exit
```

Pour faire fonctionner le tunnel avec un équipement situé derrière un NAT, il suffit (d'après la documentation du site) de remplacer l'adresse source du tunnel (qui est normalement l'adresse publique IPv4 de sortie) par l'adresse privée IPv4 de l'équipement. Ici, c'est l'adresse de la patte de sortie du *HedgeRouteur* vers le réseau cantonal qui est utilisée.

Il faut encore rajouter une route par défaut qui enverra tous le trafic IPv6 en direction du tunnel :

```
HedgeRouteur(config)#ipv6 route ::/0 Tunnel0
```

Si aucun serveur DNS interne n'est disponible, il est aussi possible d'ajouter les serveurs DNS fournis avec le tunnel. Pour ce prototype, le serveur DNS du réseau est utilisé.

### 6.1.5 Adresse IP dynamique

Pour configurer un tunnel, une adresse statique est requise. Si l'équipement sur lequel le tunnel doit être installé à une adresse dynamique, une solution existe. Dans la gestion du compte, sous l'onglet *Advanced*, une option permet de mettre à jour l'adresse IP de l'équipement en utilisant le système de DNS dynamique du fournisseur.

Cette option n'a pas eu lieu d'être dans le prototype.

### 6.1.6 Adressage du réseau local

Pour que le reste du réseau soit configuré en IPv6 et utilise le tunnel, deux choix sont possibles :

- Il n'y a qu'un seul sous-réseau à adresser :

Le préfixe **2001:470:1f13:101d::/64** (soit 18 quintillions d'adresses, plus que nécessaire) doit être obligatoirement utilisé pour la configuration.

Ce sous-réseau est celui attribué par le fournisseur de tunnel et est le seul à être routé dans le tunnel, ce qui veut dire qu'utiliser un autre sous-réseau n'est pas possible ici.

- Il y a plusieurs sous-réseaux attenants à adresser :

Le préfixe **2001:470:ccd8::/48** va être utilisé. Pour ce faire, il faut tirer de ce préfixe des sous-réseaux en /64. Pour rappel la taille des sous-réseaux, contrairement à IPv4, est fixe et d'une taille de /64. Avec un préfixe en /48, on peut tirer 65'536 sous-réseaux utilisables. Dans notre cas ils sont les suivants :

```
2001:470:ccd8:0000::/64 à 2001:470:ccd8:ffff::/64
```

Les 16 bits en rouge représentent en IPv6 l'adresse du sous-réseau.

Il ne reste plus qu'à choisir les sous-réseaux à utiliser et adresser les différents sous-réseaux. Quand une machine voudra utiliser le tunnel, celui-ci va reconnaître qu'elle implémente bien le préfixe attribué en /48 et autoriser la connexion.

Dans ce prototype cas, il y a plusieurs sous-réseaux à adresser, donc la deuxième solution est adoptée.

Les deux préfixes retenus sont les suivants :

```
2001:470:ccd8:0001::/64
2001:470:ccd8:0002::/64
```

Il faut maintenant configurer le premier sous-réseau, en commençant par adresser l'interface correspondant à celui-ci et activer l'IPv6 dessus :

```
HedgeRouteur(config)#interface fa 0/1.2
HedgeRouteur(config-if)#ipv6 address 2001:470:ccb8:0001::1/64
HedgeRouteur(config-if)#ipv6 enable
```

Puis, le mécanisme d'auto configuration sans-état est utilisé pour adresser les machines du réseau.

Tout d'abord il faut paramétrer le temps entre chaque envoi d'annonce du préfixe et le temps de vie de ceux-ci, puis entrer le préfixe souhaité :

```
HedgeRouteur(config-if)#ipv6 nd ra-interval 60
HedgeRouteur(config-if)#ipv6 nd ra-lifetime 180
HedgeRouteur(config-if)#ipv6 nd prefix 2001:470:ccd8:0001::/64
HedgeRouteur(config-if)#exit
```

L'interface fast-ethernet 0/1.3 est paramétrée de la même manière, mais avec le second préfixe.

### 6.1.7 Configuration des machines

Les machines sont configurées de façon à recevoir automatiquement leur adresse IPv6, comme décrit au point 5.4.1.

### 6.1.8 Tests de fonctionnement

Les tests de fonctionnement du réseau local ont réussi. Les machines ont pu entrer dans le domaine Active Directory et utiliser le DNS pour résoudre les noms des machines du réseau.

Malheureusement, aucune connectivité internet IPv6 n'a pu être obtenue, et impossible de contacter le serveur de destination du fournisseur en IPv6.

Le protocole 41, utilisé pour encapsuler les paquets envoyés dans le tunnel, peut être bloqué par du NAT à des niveaux différents, ce qui est sûrement la cause du non fonctionnement du prototype :

- Par le routeur vDSL, qui n'est pas configuré pour accepter ce type de communication.
- Par le fournisseur d'accès, qui pourrait bloquer ce protocole pour des raisons de sécurité ou autres. Si le protocole est bien bloqué à ce niveau, il n'existerait aucun moyen de contourner le problème.
- Un pare-feu pourrait également bloquer le protocole.

Il semble néanmoins que le trafic passe bien depuis le routeur de sortie jusqu'au routeur vDSL, mais aucun trafic de retour ne semble parvenir. Des tentatives de permissions (access-lists) ont été testées afin de permettre le trafic du protocole 41 sur le NAT du routeur, mais sans succès.

Après vérifications, aucun pare-feu ne barre la route.

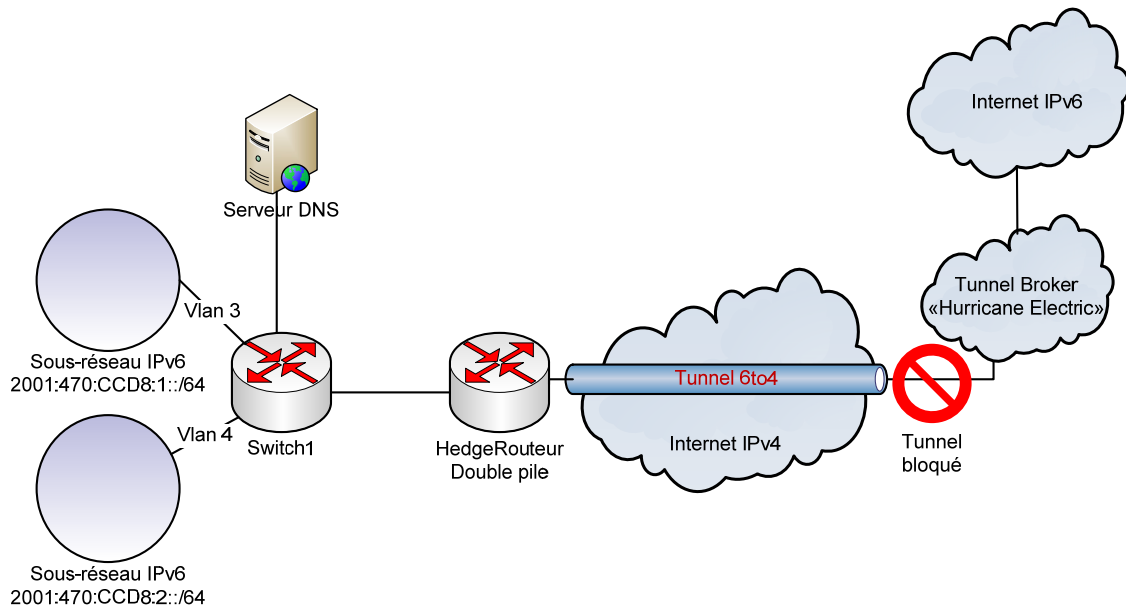
Aucune solution n'a malheureusement pu être trouvée pour résoudre ce problème.

### 6.1.9 Configuration finale du tunnel

Finalement, voilà à quoi correspond l'implémentation du tunnel après configuration :

**Figure 17**

**Schéma du tunnel *Hurricane Electric***



Comme on peut le voir, les sous-réseaux ont été correctement configurés, cela répond aux exigences du prototype. Mais il a été impossible d'obtenir une connectivité à travers le tunnel.

Pour pallier à ce problème, les deux autres fournisseurs de tunnel, SixXs et Gogo6, ont été choisis. Ils fournissent des solutions qui permettent d'encapsuler les paquets non pas avec le protocole 41 mais dans des paquets UDP, permettant ainsi de passer outre les problèmes de NAT et de protocole bloqué.

## 6.2 Implantation avec SixXs

### 6.2.1 Enregistrement

Tout d'abord, il faut s'enregistrer sur la page <https://www.sixxs.net/signup/create/>. La création de compte est un peu plus restrictive, car les informations sont bien contrôlées et un seul compte par personne est autorisé.

### 6.2.2 Création du tunnel




Le tunnel repose sur le même principe qu'avec Hurricane Electric. Il faut sélectionner un point d'accès (POP) dans une liste. Ici, celui de Genève est sélectionné. Puis il faut choisir quel type de tunnel on désire. Deux options sont possibles :

- Faire un tunnel qui utilise le protocole 41 pour l'encapsulation. Il est précisé qu'il peut y avoir des problèmes avec les équipements derrière un NAT.
- Faire un tunnel appelé AYIYA (Anything In Anything) qui propose de faire de l'encapsulation des paquets IPv6 dans des paquets IPv4 à l'aide des protocoles UDP, TCP ou SCTP<sup>41</sup>. Cela permet de passer outre les problèmes de NAT.

N'ayant pas réussi à faire fonctionner un tunnel avec le protocole 41 avec Hurricane Electric, la deuxième solution a été choisie. Une fiche récapitulative donne les informations du tunnel créé :

**Figure 18**

#### Informations tunnel SixXs

<b>Tunnel Name</b>	My First Tunnel
<b>PoP Name</b>	chgva01
<b>PoP Location</b>	Geneva, Switzerland 
<b>PoP IPv4</b>	46.20.243.4
<b>TIC Server</b>	tic.sixxs.net (default in AICCU)
<b>Your Location</b>	Geneve, Switzerland 
<b>Your IPv4</b>	AYIYA, currently 212. 
<b>IPv6 Prefix</b>	2a02:2528:ff00:175::1/64
<b>PoP IPv6</b>	2a02:2528:ff00:175::1
<b>Your IPv6</b>	2a02:2528:ff00:175::2
<b>Created</b>	2012-10-06 10:36:18 UTC
<b>Last Alive</b>	2012-11-02 14:45:05 UTC
<b>Last Dead</b>	2012-11-02 00:15:01 UTC
<b>Uptime</b>	0 days (based on latency check)
<b>Config State</b>	AYIYA (automatically enabled on the fly)
<b>PoP Status</b>	Live Tunnel Status on the PoP

Source : Détails du tunnel de mon compte SixXs

---

<sup>41</sup> Voir <http://www.sixxs.net/tools/ayiya/>



### 6.2.3 Configuration du tunnel

Pour créer le tunnel, un logiciel nommé AICCU (Automatic IPv6 Client Utility)<sup>42</sup> est à installer sur la machine qui fera office de serveur d'accès au monde IPv6. Le logiciel ne pouvant être installé sur du matériel Cisco, une machine Windows 7 a d'abord été envisagée. Mais les drivers pour Windows 7 ne sont pas compatibles avec l'envoi de messages d'auto-configuration pour le reste du réseau local ; alors, une machine Ubuntu 12.04, qui offrait elle cette possibilité, a été choisie à la place.

Pour l'installer, il suffit de lancer une fenêtre de commande et d'écrire :

```
sudo apt-get install aiccu43
```

A la fin de l'installation, le nom et le mot de passe du compte SixXs sont demandés. Normalement, aucune autre configuration ne devrait être faite, mais en cas de pépins ou si des changements sont à faire, le fichier de configuration du tunnel `/etc/aiccu.conf` (en annexe dans les cd-rom) peut être changé. Une route par défaut est directement créée, qui sert à envoyer tout trafic IPv6 en direction du tunnel. Le logiciel ayant des difficultés à s'initialiser au redémarrage de l'ordinateur, un script a été créé<sup>44</sup> afin de rectifier ceci.

### 6.2.4 Configuration du réseau local

Un préfixe en /64 est donné d'office pour configurer le reste des machines du sous-réseau. Ici il est le suivant :

```
2a02:2528:ff00:8175::/64
```

Pour obtenir un préfixe en /48 qui, rappelons-le, permet de configurer plusieurs sous-réseaux, il faut faire une demande spéciale. Malheureusement, la demande n'a pas été acceptée et il faudra se contenter d'adresser seulement un seul réseau local.

Pour que la machine Ubuntu officie comme routeur, le démon *radvd*, qui permet l'envoi de *router advertisement* sur une machine linux, est installé avec la commande suivante :

```
sudo apt-get install radvd
```

---

<sup>42</sup> Voir <https://www.sixxs.net/tools/aiccu/>

<sup>43</sup> Source : Documentation SixXs

<sup>44</sup> Script disponible en Annexe 2

Il faut encore créer le fichier `radvd.conf` pour qu'il fonctionne :

```
Sudo nano /etc/radvd.conf
```

Puis entrer les commandes ci-dessous<sup>45</sup> qui permettront d'envoyer des avertissements routeurs avec un préfixe, qui eux permettront aux machines les recevant de se configurer toutes seules :

```
Interface eth0
{
    AdvSendAdvert on;
    Prefix 2a02:2528:ff00:8175::/64
    {
        AdvOnLink on;
        AdvAutonomous on;
    };
};
```

Finalement, le démon *radvd* doit être relancé :

```
sudo invoke-rc.d radvd restart
```

Voilà, la machine Ubuntu officie maintenant comme routeur. Toutes les communications désirant sortir en IPv6 passeront par elle automatiquement.

### 6.2.5 Configuration des machines

En plus de leur configuration classique en IPv4, les machines sont configurées de façon à recevoir automatiquement leur adresse IPv6, comme décrit au point 5.4.1.

### 6.2.6 Tests de fonctionnement

Les machines du réseau ont passé sans problèmes les tests de fonctionnement du réseau local. La sortie sur l'internet IPv6 a également été réalisée, l'adresse de sortie est bien celle du tunnel, soit `2a02:2528:ff00:175::2` ; pour confirmation, se référer à l'annexe 3. Différents sites web en IPv6 ont également été consultés. Il n'y a pas eu de problèmes de temps d'attentes anormalement longs ou de déconnexions intempestives. L'envoi de mail n'a pas posé de problèmes non plus.

---

<sup>45</sup> Source : Documentation Radvd

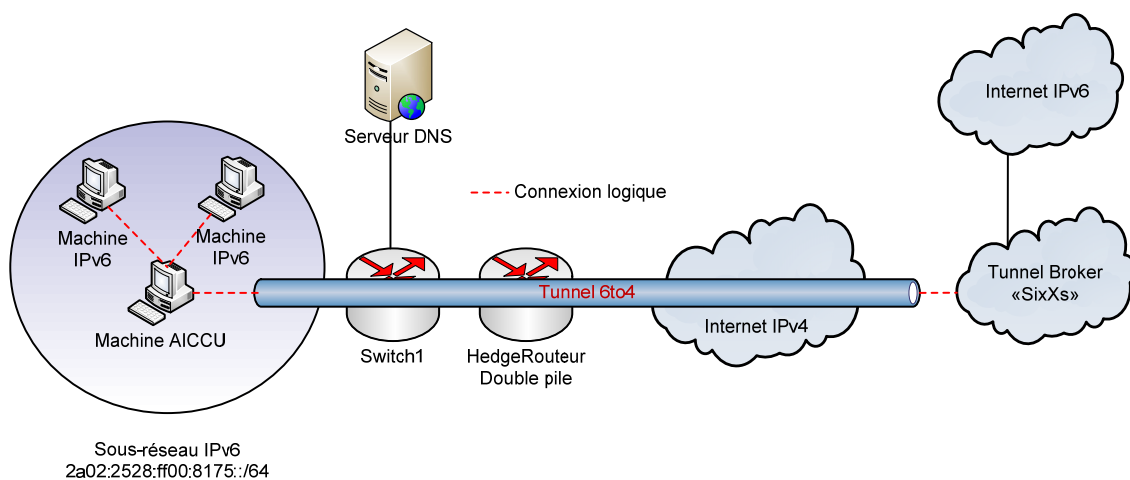
Le prototype n'est malheureusement pas pleinement fonctionnel car seul un sous-réseau a pu être configuré, un préfixe en /48 n'ayant pas pu être obtenu dans les temps du travail.

### 6.2.7 Configuration finale du tunnel

Finalement, voilà à quoi correspond l'implémentation logique du tunnel après configuration :

Figure 19

Schéma du tunnel SixXs



Pour utiliser le tunnel, les machines IPv6 envoient leur trafic en direction du *HedgeRouteur*, auquel elles sont connectées. Mais grâce à la configuration de la route par défaut sur la machine AICCU, tout ce trafic est intercepté par celle-ci et envoyé directement dans le tunnel. Le trafic est encapsulé en entrant dans le tunnel, décapsulé à la sortie et enfin transmis à destination.

Comme dit plus haut, un seul sous-réseau a pu être configuré et obtenir la connectivité du tunnel, ce qui n'est pas conforme aux exigences du prototype. Une troisième solution est encore disponible afin de résoudre ce problème, celle de prendre le fournisseur de tunnel *Gogo6*.

## **6.3 Implantation avec Gogo6**

Cette partie a été réalisée en dernier, car les serveurs ont eu de gros problèmes de disponibilité, ceci à partir de début octobre, et une maintenance prolongée a été effectuée afin que le service puisse être à nouveau disponible.

Pendant cette période, il était toujours possible d'obtenir un tunnel, mais seulement de manière anonyme, ce qui empêchait d'obtenir un préfixe. Ce n'est qu'une semaine avant la fin du travail que la situation s'est débloquée et que les tests ont pu reprendre.

### **6.3.1 Enregistrement**

Pour obtenir un tunnel, il faut d'abord s'enregistrer dans la communauté Gogo6 à l'adresse suivante : <http://www.gogo6.com/profiles/members/>. Une fois l'enregistrement terminé, il faut également se créer un compte pour utiliser leur service *Freenet6*, qui lui propose la création d'un tunnel.

Pour ce faire, depuis la page d'accueil, aller dans sous l'onglet *Service* et choisir l'option *Setup your Freenet6 account*.

Un compte est requis pour l'obtention d'un préfixe (en /56) qui permet, rappelons-le, d'adresser le ou les sous-réseaux requérant le service du tunnel. Si une seule machine souhaite accéder à l'internet IPv6, un compte n'est pas requis.

Gogo6 possède trois serveurs d'accès, un à Montréal, un à Amsterdam et le dernier à Taipei. Pour l'instant, et pour des causes de maintenance des accès et des enregistrements des comptes, seules les inscriptions au serveur de Montréal sont disponibles. Le compte a donc été créé avec ce serveur d'accès.

### **6.3.2 Création du tunnel**

Un programme, appelé *GogoCLIENT* est à installer sur la machine souhaitant implémenter le tunnel. La disponibilité de celui-ci est limitée aux systèmes d'exploitation comme Windows, Linux ou Mac OS.

Il n'est donc pas possible d'installer directement le tunnel sur une plateforme comme un routeur Cisco, Jupiter ou d'autres constructeurs. Néanmoins, une option permet de propager le tunnel au routeur connecté à la machine qui l'implémente.

### 6.3.2.1 Installation et configuration du programme GogoCLIENT

Le programme *GogoCLIENT* (version basique 1.2 pour Windows 32 bits) a été téléchargé à l'adresse suivante : <http://www.gogo6.com/profile/gogoCLIENT> (l'accès à celle-ci requiert d'être enregistré avec son compte).

L'installation du programme est classique : il faut accepter les conditions d'utilisations, choisir les composants à installer (pour cette installation les paramètres sont laissés par défaut) et choisir le dossier de destination (ici par défaut aussi). Pendant l'installation, il est demandé si la carte réseau virtuelle de Gogo6 doit être installée ; il faut accepter.

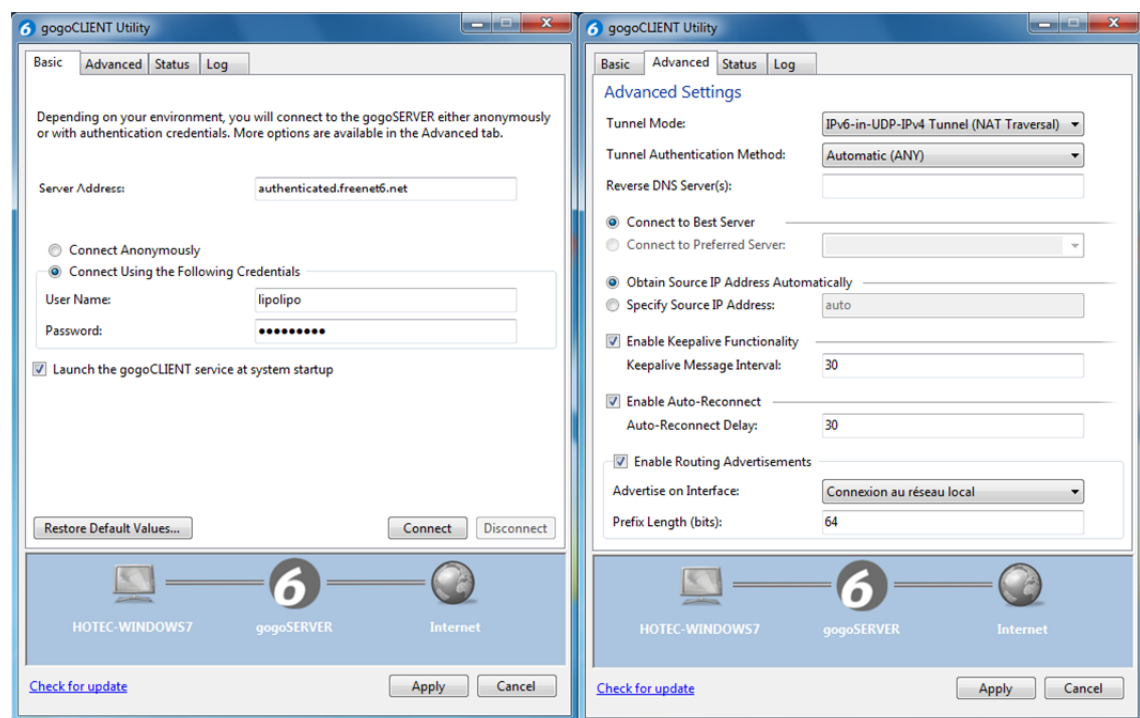
L'installation terminée, il faut lancer le programme par le chemin d'accès suivant :

Démarrer → Tous les programmes → gogo6 → gogoCLIENT → gogoCLIENT Utility

Voilà à quoi correspondent les onglets qui servent à configurer le tunnel :

Figure 20

### Fenêtres de configuration du programme GogoCLIENT



Dans l'onglet *Basic*, il faut entrer *authenticated.freenet6.net* comme adresse de serveur, c'est lui qui va se charger d'authentifier le compte auprès du serveur sur lequel on s'est enregistré. Il faut renseigner ce dernier en cochant l'option *Connect Using the Following Credentials* et entrer le nom d'utilisateur et le mot de passe du compte Freenet6 créé auparavant.

Dans l'onglet *Advanced*, il faut choisir *IPv6-in-UDP-IPv4 Tunnel (NAT Traversal)*, car l'ordinateur est derrière le NAT du routeur. On peut voir que la méthode d'encapsulation est la même que celle utilisée pour le prototype *SixXs*, soit l'encapsulation dans des paquets UDP, ce qui permet de traverser le NAT.

Plusieurs choix sont possibles quant à la méthode d'authentification : en texte clair, avec un algorithme de cryptage MD5 ou du chiffrement 3DES1. Suivant le niveau de sécurité à apporter à la connexion, une méthode peut-être préférable à une autre, la plus sécurisée étant 3DES1 et la moins sécurisée en texte clair. Pour ce prototype, la méthode est laissée en automatique.

Le choix de connexion à un serveur en particulier a été délégué à la connexion automatique au meilleur serveur, car l'enregistrement au serveur de Montréal n'a pas fonctionné tout de suite. Il a fallu contacter l'assistance client pour régler ce problème. Un accès au serveur d'Amsterdam a été configuré également par leurs soins en cas de soucis supplémentaires. Avec ces deux accès, le choix se portera, lors de la connexion, sur le plus rapidement atteignable des deux.

L'option *Enable Keepalive functionality* est activée. Cette option est utilisée ici, car elle permet de maintenir le mappage du NAT pour UDP en état (en envoyant par intervalle régulier de 30 secondes des messages au serveur) pour que le tunnel fonctionne correctement. L'option *Auto-reconnect* est également cochée au cas où le tunnel tomberait, ce qui a été le cas par exemple lors de mise en veille prolongée de l'ordinateur.

Ensuite, l'option *Enable routing advertisements* permet à la machine d'officier comme routeur et de configurer les machines du même sous-réseau grâce à l'envoi des messages d'auto-configuration contenant le préfixe à utiliser.

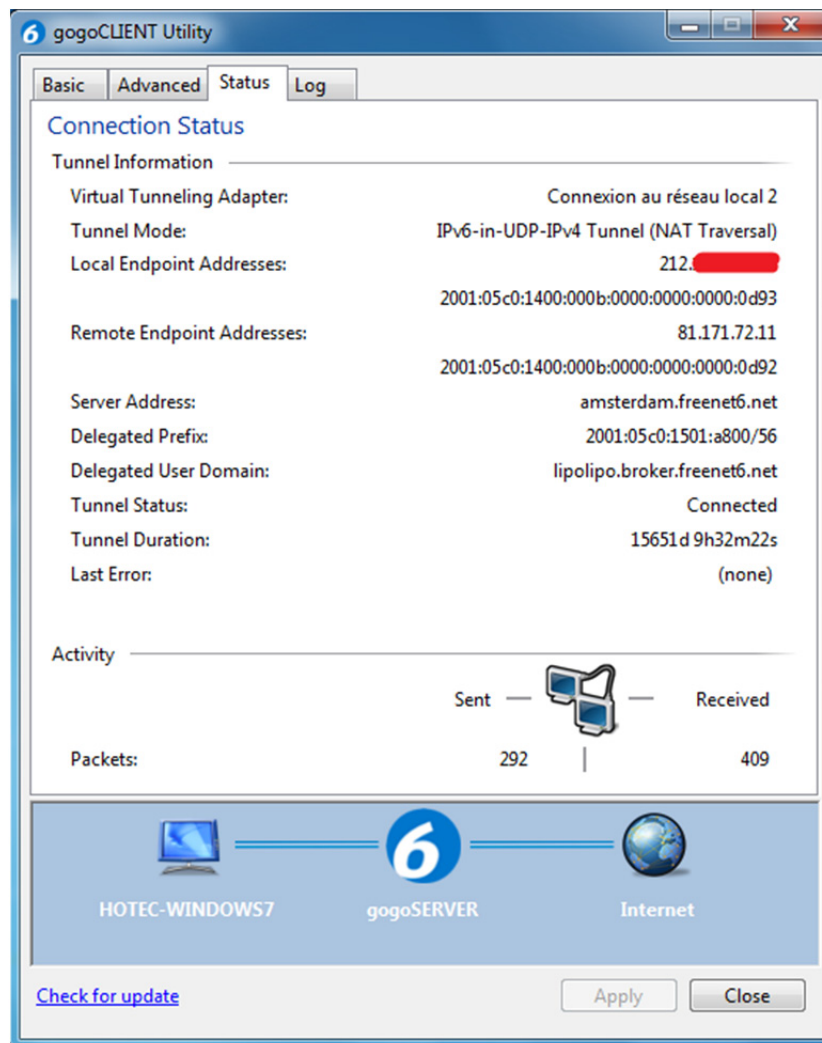
Pour terminer, il faut confirmer les options en cliquant sur le bouton *Apply*.

### **6.3.2.2 Connexion et informations du tunnel**

Les paramètres correctement configurés, il ne reste plus qu'à se connecter. La figure 21, à la page suivante, permet de voir quelles sont les informations de création du tunnel.

Figure 21

### Statut du tunnel Freenet6



Comme on peut le voir, une adresse IPv6 (ici *Local Endpoint Addresses*) a été attribuée à la machine. Le serveur choisi a été celui d'Amsterdam. Enfin, un préfixe permettant d'adresser d'autres sous-réseaux (en /56) a pu être obtenu. L'adresse publique du routeur vDSL a été masquée en rouge pour des raisons de sécurité.

### 6.3.3 Configuration des machines

En plus de leur configuration classique en IPv4, les machines sont configurées de façon à recevoir automatiquement leur adresse IPv6, comme décrit au point 5.4.1. Le programme se chargeant d'envoyer les messages d'auto-configuration, aucune autre configuration n'est à ajouter.

A des fins de tests, certaines machines ont été uniquement adressées en IPv6.

### 6.3.4 Configuration de la globalité du réseau

Pour ce faire, deux choix ont été envisagés afin que le deuxième sous-réseau (le VLAN 3) puisse obtenir une connectivité IPv6 :

- La machine implémentant le tunnel peut déléguer l'utilisation de celui-ci à une autre machine. Pour ce faire, on peut passer la machine en mode *proxy*. Ce mode permet à la machine de demander un tunnel pour un autre équipement. De ce fait, on peut configurer celui-ci sur le *HedgeRouteur*. On peut alors configurer l'adressage du deuxième sous-réseau sur le routeur et les machines passeront directement par son tunnel pour sortir en IPv6.
- La machine implémentant le tunnel agit comme routeur de sortie IPv6 pour le deuxième sous-réseau. Pour ce faire, une configuration de routage statique doit être mise en place pour que les machines du deuxième sous-réseau s'adressent directement au tunnel natif.

Malheureusement, le tunnel à configurer sur le routeur doit obligatoirement posséder une adresse publique, ce qui nous ramène aux problèmes de NAT et de protocole 41 vus précédemment.

La deuxième solution a donc été naturellement adoptée.

#### 6.3.4.1 Configuration du deuxième sous-réseau

Le tunnel étant installé dans le VLAN 4, seules les machines s'y trouvant peuvent recevoir les messages d'auto-configuration et obtenir une connectivité IPv6. Pour que les machines du VLAN3 obtiennent aussi leur accès, il faut commencer par configurer l'adressage sur le routeur. Pour cela, le préfixe `2001:05c0:1501:a801::/64`, tiré du préfixe en /56 reçu par le fournisseur de tunnel est utilisé :

```
HedgeRouteur(config)#interface fa 0/1.2
HedgeRouteur(config-if)#ipv6 address 2001:05c0:1501:a801::1/64
HedgeRouteur(config-if)#ipv6 enable
HedgeRouteur(config-if)#ipv6 nd ra-interval 60
HedgeRouteur(config-if)#ipv6 nd ra-lifetime 180
HedgeRouteur(config-if)#ipv6 nd prefix 2001:05c0:1501:a801::/64
HedgeRouteur(config-if)#exit
```

Les machines du VLAN3 doivent être configurées également de sorte à recevoir leur adresse IPv6 de façon automatique.



#### 6.3.4.2 Configuration du routage statique

Pour que le trafic IPv6 des hôtes du deuxième sous-réseau soit envoyé au tunnel, trois choses sont nécessaires.

Premièrement, activer le routage IPv6 et configurer la patte du routeur correspondant au VLAN4 avec une adresse IPv6 statique.

```
HedgeRouteur(config)#ipv6 unicast-routing
HedgeRouteur(config)#interface fa 0/1.3
HedgeRouteur(config-if)#ipv6 address 2001:05c0:1501:a800::2/64
HedgeRouteur(config-if)#ipv6 enable
HedgeRouteur(config-if)#exit
```

L'adresse choisie ici est une des adresses du sous-réseau du VLAN4.

Deuxièmement, il faut ajouter sur le routeur une route par défaut qui va envoyer tout trafic IPv6 vers l'ordinateur qui possède le tunnel via son interface de sortie vers celui-ci :

```
HedgeRouteur(config)#ipv6 route ::/0 fastEthernet 0/1.3 2001:05c0:1501:a800::1
```

Enfin, sur la machine implémentant le tunnel, il faut ajouter une route statique qui définit le chemin d'accès du deuxième sous-réseau à l'ordinateur. Pour cela, dans une fenêtre de configuration lancée en mode administrateur, écrire :

```
netsh interface ipv6 add route 2001:05c0:1501:a801::/64 « Connexion au réseau local » 2001:05c0:1501:a800::2
```

Cette route définit que pour contacter le deuxième sous-réseau (ici 2001:05c0:1501:a801::/64), il faut passer par l'adresse du routeur précédemment configurée (2001:05c0:1501:a800::2) via la carte de réseau local. La configuration du *HedgeRouteur* peut être retrouvée en annexe 7.

#### 6.3.5 Tests de fonctionnement

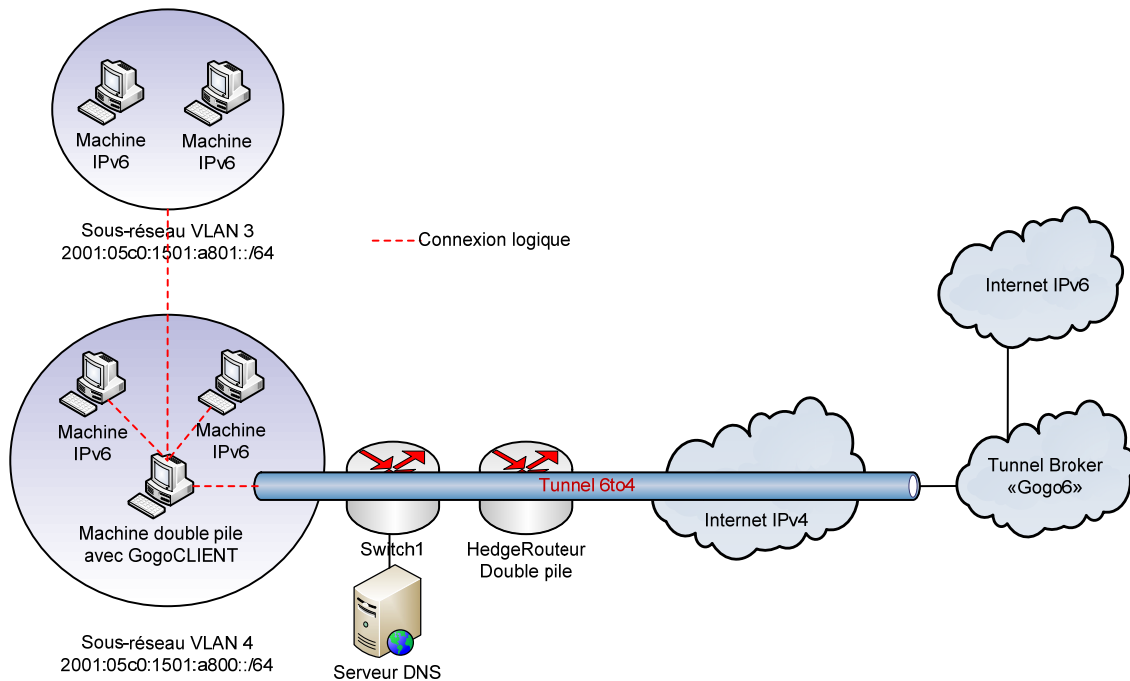
Les machines des VLAN 3 et 4 ont passé sans problèmes les tests de fonctionnement du réseau local. La sortie sur l'internet IPv6 a également été réalisée, l'adresse de sortie est bien celle du tunnel ; pour confirmation se référer à l'annexe 4. Différents sites web en IPv6 ont également été consultés. Il n'y a pas eu de problèmes de temps d'attentes anormalement longs ou de déconnexions intempestives. L'envoi de mail n'a pas posé de problèmes sur des machines IPv6, autant sur la messagerie web Gmail qu'avec le client Thunderbird.

### 6.3.6 Configuration du tunnel

Finalement, voilà à quoi correspond l'implémentation de ce prototype :

Figure 22

Schéma du tunnel Freenet6



Ici, comme pour le prototype avec *SixXs*, les machines du VLAN4 utilisent le tunnel par le biais de la machine avec le gogoCLIENT. Les machines du VLAN3, elles, grâce au routage statique configuré auparavant, sont aussi reliées à celle-ci et peuvent utiliser le tunnel pour obtenir leur connectivité IPv6.

## **6.4 Conclusion des prototypes avec tunnel**

Après deux tentatives infructueuses, une troisième a permis de remplir les conditions imposées pour ce prototype et donc de permettre tout le réseau de fonctionner en IPv6. Dans le cas d'une entreprise ou d'une école, deux solutions sont adoptables :

- Placer le tunnel sur le routeur de sortie du réseau : celui-ci doit, dans la majorité des cas, posséder une adresse publique. Les solutions de tunnels statiques des fournisseurs *Hurricane Electrics* et *SixXs* sont là pour ce type d'installation. La préférence se porterait néanmoins sur le premier, car étant plus professionnel (il possède son propre réseau international), il possède une communauté plus active et réactive sur les forums et offre ses services en toute gratuité sans conditions.
- Placer le tunnel sur un ordinateur du réseau qui fera office de routeur : cette solution permet à un réseau derrière du NAT d'obtenir une connectivité IPv6. Les deux solutions possibles sont celles de *SixXs* et *Gogo6*. Le choix est difficile à faire entre les deux fournisseurs, car chacun possède ses avantages et inconvénients. *Gogo6* semble avoir couramment des problèmes d'indisponibilités avec ses serveurs, mais la configuration requiert moins de connaissances informatiques et est plus claire. De l'autre côté, *SixXs* est moins facile à prendre en main, certaines options sont impossibles avec Windows, mais son service est bien soutenu par des serveurs opérationnels. Le choix de l'un ou de l'autre dépend de ces paramètres.

Les problèmes rencontrés ont été soit externes (serveurs du fournisseur en panne, impossible d'obtenir un préfixe pour plusieurs sous-réseaux), soit le fait de limitations internes (routeur avec adresse privée, problèmes techniques durant les installations, adaptabilité du matériel).

Pour une petite entreprise ou une école, mettre en place une telle solution peut être problématique, car il faut des équipements supportant IPv6, ce qui n'est pas le cas de tous, et le remplacement de l'existant peut être onéreux. Il est judicieux de faire d'abord un inventaire de l'équipement existant et de sa compatibilité avec le protocole pour s'assurer que l'implantation est possible.

Pour terminer, cette solution à base de tunnel semble, à part une légère perte de performance due au passage par les serveurs du fournisseur et les possibles problèmes d'indisponibilité des serveurs, très correcte et offre une solution gratuite tout à fait fonctionnelle pour activer IPv6 sur son réseau.

## 7. Prototype avec translation d'adresse NAT-PT

Ce prototype a, rappelons-le, pour but de donner à trois sous-réseaux uniquement IPv6 une connectivité au monde IPv4. Rappelons également que la solution à base de NAT64 et de DNS64 est plus actuelle, plus stable et plus robuste que le NAT-PT et le remplace avantageusement, ce dernier ayant été déprécié et ne devant donc normalement plus être utilisé.

Mais son implémentation requiert du matériel Cisco onéreux et non disponible à l'école, c'est pourquoi NAT-PT a quand même été choisi pour réaliser le prototype.

### 7.1 Adressage des sous-réseaux en IPv6

Dans ce prototype, trois sous-réseaux sont à adresser, ceux des VLAN 2, VLAN 3 et VLAN 4 du *Switch2*. Pour cela, il a été jugé intéressant de mettre en œuvre les trois solutions actuelles d'adressage en IPv6, soit le DHCPv6, l'auto-configuration sans-état et l'adressage statique, ceci afin de mieux comprendre et d'expérimenter ces techniques.

#### 7.1.1 Méthode de configuration des VLAN

Chaque sous-réseau aura donc sa propre configuration :

- Pour le VLAN 2 → Auto-configuration sans état
- Pour le VLAN 3 → DHCPv6
- Pour le VLAN 4 → Adressage statique

Nous ne nous attarderons pas sur les détails des paramétrages effectués, ceux-ci peuvent être retrouvés à l'annexe 5 (fichier de configuration du *RouteurNATPT*).

Pour vérifier le bon fonctionnement de chaque méthode, des tests ont été effectués sur une machine Windows 7 et une autre avec Ubuntu 12.04. Tests passés avec succès, sauf pour le DHCPv6 où il a fallu forcer l'attribution du serveur DNS à la main. Sinon toutes les machines ont pu se configurer automatiquement en entrant sur chaque sous-réseau.

Toutes les machines n'ont, bien entendu, pas d'adressage IPv4. Celui-ci a été désactivé sur toutes les machines.

Des plages d'adresses privées IPv6 ont été sélectionnées, car les machines officient derrière du NAT.

## 7.2 Configuration du NAT-PT sur le routeur

Pour le prototype, trois choses sont nécessaires pour le bon fonctionnement du NAT-PT sur le routeur :

- Configurer la translation d'adresses.
- Configurer l'adresse du DNS pour que les hôtes IPv6 puissent l'utiliser.
- Paramétrer le mappage des adresses IPv4 avec un préfixe IPv6 pour la communication entre les deux mondes.

### 7.2.1 Translation des adresses IPv6 en IPv4

Pour commencer, il faut spécifier sur chaque interface active du *RouteurNATPT* la commande suivante, afin d'annoncer que celles-ci vont travailler avec le NAT-PT<sup>46</sup> :

```
RouteurNATPT(config-if)#ipv6 nat
```

Pour réaliser la translation, il faut commencer par choisir une méthode de NAT-PT parmi les trois disponibles (statique, dynamique, et PAT). La méthode dynamique a été choisie, en prenant comme pool d'adresses celles inutilisées du sous-réseau dans lequel se trouve le routeurNATPT.

Ce sous-réseau est le suivant : 192.168.3.0/28. Les adresses 192.168.3.1 à 192.168.3.4 étant prises par les autres équipements, les adresses suivantes ont été sélectionnées :

```
192.168.3.5 – 192.168.3.14
```

L'adresse 192.168.3.15 n'a évidemment pas été incluse dedans, car c'est l'adresse réservée pour le *broadcast*<sup>47</sup>.

Pour configurer le pool d'adresses, il faut préciser plusieurs paramètres. Le terme v6v4 permet de dire que l'opération de translation se fera dans le sens IPv6 → IPv4. Il faut donner un nom au pool et définir quelles adresses lui attribuer. Enfin, le masque de sous-réseau de celles-ci doit être connu. Ce qui donne finalement la commande suivante :

```
RouteurNATPT(config)#ipv6 nat v6v4 pool ipv4Pool 192.168.3.5 192.168.3.14 prefix-length 28
```

<sup>46</sup> Source : Documentation Cisco

<sup>47</sup> Voir [http://fr.wikipedia.org/wiki/Broadcast\\_%28informatique%29](http://fr.wikipedia.org/wiki/Broadcast_%28informatique%29)

Il faut également définir une *access-list* qui définit quelles sont les adresses IPv6 autorisées à être traduites. Ici, ce seront toutes celles des trois sous-réseaux configurés :

```
RouteurNATPT(config)#ipv6 access-list listIPv6
RouteurNATPT(config-ipv6-acl)#permit ipv6 fd00:1234:5678:abcd::/64 any
RouteurNATPT(config-ipv6-acl)#permit ipv6 fd00:1234:5678:cafe::/64 any
RouteurNATPT(config-ipv6-acl)#permit ipv6 fd00:1234:5678:beef::/64 any
RouteurNATPT(config-ipv6-acl)#exit
```

Pour terminer cette partie, il faut définir la règle de NAT avec les deux options dernièrement configurées :

```
RouteurNATPT(config)#ipv6 nat v6v4 source list listIPv6 pool ipv4Pool
```

## 7.2.2 Communication IPv6/IPv4

Pour qu'un hôte IPv6 puisse communiquer avec un hôte IPv4, celui-ci est obligé de posséder une représentation IPv6 de ce dernier. Rappelons-le, l'adresse IPv4 est concaténée avec le préfixe *2001::/96* pour former une adresse IPv6 fictive qui peut être utilisée comme adresse de destination pour un hôte IPv6. L'adresse IPv4 sera décapsulée de l'adresse IPv6 à son passage sur le *RouteurNATPT*.

### 7.2.2.1 Configuration de l'adresse du DNS

Les hôtes IPv6 ayant besoin de s'adresser au serveur DNS constamment, une règle statique de NAT doit être établie afin qu'ils puissent établir une communication :

```
RouteurNATPT(config)#ipv6 nat v4v6 source 192.168.3.3 2001::C0A8:303
```

Lorsqu'une machine IPv6 enverra un paquet avec comme adresse de destination *2001:C0A8:302*, le routeur la traduira en son adresse IPv4, et inversement lors du retour ce qui assure une communication bidirectionnelle.

### 7.2.2.2 Configuration de l'IPv4-mapped

Il n'est pas suffisant que de configurer uniquement l'accès au serveur DNS, il faut également configurer la communication entre un hôte IPv6 et n'importe quelle adresse IPv4 (comme dans le cas du surf sur internet). Pour cela, la technique IPv4-mapped est utilisée.

Pour commencer, il faut créer une *access-list* définissant quels sont les hôtes pouvant communiquer avec le monde IPv4 (représenté par le préfixe 2001::/96).

```
RouteurNATPT(config)#ipv6 access-list aclIPv6
RouteurNATPT(config-ipv6-acl)#permit ipv6 fd00:1234:5678:abcd::/64 2001::/96
RouteurNATPT(config-ipv6-acl)#permit ipv6 fd00:1234:5678:cafe::/64 2001::/96
RouteurNATPT(config-ipv6-acl)#permit ipv6 fd00:1234:5678:beef::/64 2001::/96
RouteurNATPT(config-ipv6-acl)#exit
```

Puis activer le préfixe avec le protocole IPv4-mapped et y appliquer l'*access-list* :

```
RouteurNATPT(config)#ipv6 nat prefix 2001::/96 v4-mapped aclIPv6
```

Quand le *RouteurNATPT* va recevoir un paquet IPv6, il va regarder son préfixe. Si celui-ci correspond à celui configuré ici, il va regarder dans l'*access-list* si l'adresse source est autorisée à passer ou non. Si oui, il va effectuer la translation en décapsulant l'adresse IPv4 de l'adresse IPv6 préfixée.

## 7.3 Tests et fonctionnement

### 7.3.1 Communication d'un hôte IPv6 vers un hôte IPv4

Un ping, réussi, a été effectué depuis une machine IPv6 sur le *HedgeRouteur*. Voici ce qu'il en ressort dans la table de translation :

**Figure 23**

#### **Ping du HedgeRouteur**

```
RouteurNATPT#show ipv6 nat translations
Prot  IPv4 source      IPv6 source
----  -
      IPv4 destination  IPv6 destination
----  -
icmp  192.168.3.11,1    FD00:1234:5678:CAFE:5554:A95F:D6CE:7B75,1
      192.168.3.1,1    2001::C0A8:301,1
|
```

Comme on peut le voir, un enregistrement *icmp* est apparu après avoir pingé le routeur. L'IPv6 de la machine source a été traduite en une IPv4 du pool d'adresse configuré auparavant. L'IPv4 de destination a également été extraite de l'IPv6 de destination. Tout est correctement configuré. Les tests du fonctionnement du réseau local ont été concluants.

### 7.3.2 Sortie d'un hôte IPv6 sur internet

La capture suivante, réalisée grâce au logiciel Wireshark<sup>48</sup>, permet de voir comment se passe la communication entre l'hôte IPv6 et le serveur DNS :

Figure 24

#### Demande AAAA au serveur DNS

Fd00:1234:5678:cafe: 2001::c0a8:303	DNS	Standard query 0xc2f6 AAAA age.hes-so.ch
2001::c0a8:303	Fd00:1234:5678:caf DNS	Standard query response 0xc2f6 CNAME agx.hefr.ch AAAA 2001::a062:f0dc

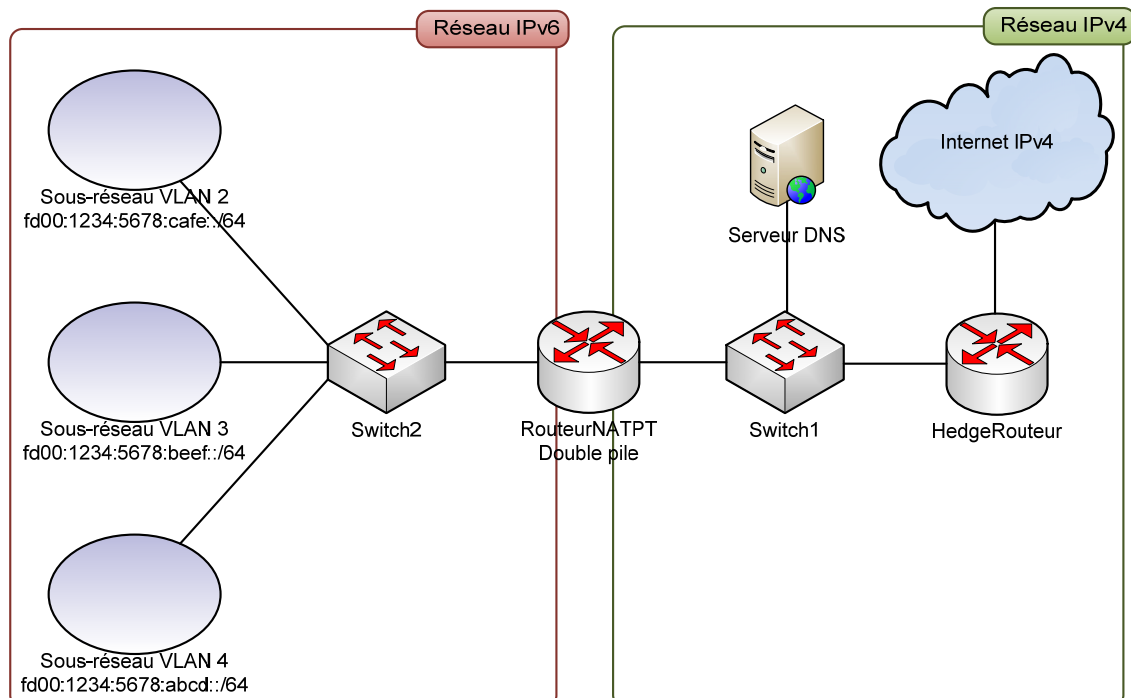
On peut voir que l'adresse IPv6 retournée par le DNS est bien configurée avec le préfixe `2001::/96` que l'on a configuré auparavant. Le mécanisme DNS-ALG a correctement fait son travail conjointement au *routeur NATPT*.

## 7.4 Configuration du prototype

Finalement, voilà à quoi correspond l'implantation de ce prototype :

Figure 25

#### Schéma NAT-PT



48

Logiciel de capture de paquets IP. Source : <http://www.wireshark.org/>



Comme on peut le voir, il y a deux zones clairement distinctes. La zone IPv6 pure contenant les sous-réseaux, et la zone IPv4 contenant le DNS, le *HedgeRouteur* et la connexion vers l'extérieur.

Pour le DNS, il y a deux types d'installation possibles :

- Soit installer un serveur DNS dans la zone IPv6, qui lui sera relié à un serveur DNS dans la zone IPv4. Ce serveur DNS IPv6 s'occupe de son domaine IPv6 et délègue les tâches IPv4 au DNS IPv4.
- Soit directement relier les hôtes à un serveur DNS IPv4. Les requêtes DNS passeront par le routeur et seront traduites, comme vu plus haut.

Dans ce prototype, le serveur DNS a été installé dans la zone IPv4, car celui-ci doit également être disponible pour les autres équipements en IPv4. De plus, la redondance qu'il y aurait eu avec un serveur DNS IPv6 n'aurait pas apporté de changements majeurs dans la configuration.

Mais il aurait tout à fait été possible d'opter pour la première solution, pour cela il suffirait d'installer un nouveau serveur DNS et d'y ajouter les enregistrements AAAA des hôtes IPv6, et de le faire pointer sur l'adresse du DNS IPv4. Par contre, il aurait été impossible de se passer de serveur DNS IPv4, car les réponses des requêtes doivent obligatoirement passer le routeur officiant le NAT-PT.

## **7.5 Conclusion du prototype**

Les spécifications pour ce prototype ont été correctement remplies. Cependant, il reste les problèmes inhérents à la technologie NAT-PT.

Au niveau applicatif, les gestionnaires de mises à jour ont fonctionné correctement. Les services de messageries ont pu également être utilisés

Le surf sur l'internet IPv4 (*http* et *https*) semble plutôt bien fonctionner, même si certains sites restent inaccessibles (comme *Youtube* par exemple) ou certaines lenteurs anormales se font ressentir. Certains sites ne sont pas affichés correctement également. La connexion peut être perdue occasionnellement pour des raisons inconnues.

Pour ces raisons et d'autres qui ont fait que le NAT-PT a été déprécié, une implémentation en entreprise ou dans une école semble peu optimale voir inopérante.

Il aurait été très intéressant de pouvoir obtenir le matériel pouvant faire fonctionner les protocoles NAT64 et DNS64 pour se faire une idée plus actuelle de la situation de ces technologies de translation et de voir si ce type de technologie permet de se passer d'IPv4 (sauf pour la connexion externe évidemment) tout en restant fonctionnel, fiable et sécurisé.

Les problèmes rencontrés ont été les suivants : certaines documentations étaient obsolètes et il a fallu du temps avant de trouver la bonne. Les informations quant à la mise en place de cette technologie ne sont pas légion, du fait de l'abandon du protocole. Un problème est également survenu sur une machine Windows 7, qui refusait de forcer sa connexion IPv6, rendant impossible les requêtes corrects au serveur DNS. Une tentative avec une version plus récente a été effectuée et a réglé le problème.

Pour terminer, la mise en place de ce prototype a été très intéressante, et a permis de démontrer que l'on peut se passer d'IPv4 (à part pour le serveur DNS qui doit être avec cet adressage) dans un réseau aujourd'hui. Même si certains problèmes sont encore présents, la fonctionnalité est là.

## Conclusion

A l'aboutissement de ce travail, plusieurs conclusions peuvent être tirées.

Tout d'abord, le protocole IPv6 semble gentiment prendre sa place dans le monde de par les efforts effectués par les associations de promotion ainsi que par les entreprises, les équipementiers et les fournisseurs d'accès qui se penchent sérieusement sur la question. La proche pénurie d'IPv4 aidant, fort est à parier qu'IPv6 a de beaux jours devant lui. Tant qu'IPv4 sera présent, ce qui sera le cas encore surement quelques années voir beaucoup plus, il faudra assurer un contact avec ce monde grâce aux différentes techniques vues dans ce dossier. A terme, espérons qu'il n'y aura plus besoins de les utiliser (ou sporadiquement) et qu'on verra enfin apparaitre une connectivité IPv6 générale qui montrera les pleines capacités du protocole.

Dans les cas des entreprises ou des écoles, plusieurs solutions peuvent donc être mises en place :

- La double pile native est de loin la plus facile à installer et la plus robuste. Elle requière néanmoins d'obtenir un accès IPv6 auprès de son fournisseur. Pour une école, il faut regarder du côté de *Switch*. Pour une entreprise, du côté de *Swisscom* ou *Sunrise* pour la Suisse, les autres fournisseurs se penchant sur la question.
- Une solution à base de tunnel peut être instaurée de différentes manières, cela permet d'obtenir une connectivité IPv6 fonctionnelle, réactive et stable comme vu dans les prototypes réalisés.
- Elles peuvent également essayer de se passer totalement de l'IPv4 dans le réseau interne en mettant en place des techniques de translation comme le NAT-PT (en plus d'un accès IPv6). Après expérimentations, cette solution ne semble pas encore complètement au point et souffre encore de trop nombreux défauts pour être jugée comme acceptable. Le couplage NAT64/DNS64 semble corriger la majorité des défauts du NAT-PT, et a déjà été testé à petite ou grande échelle<sup>49</sup> avec plus ou moins de succès et pourrait donc être utilisé également, mais avec précaution.
- Utiliser un serveur proxy peut aussi permettre d'accès au monde IPv6 avec des hôtes IPv4. Une expérimentation pourrait être intéressante à réaliser.

---

<sup>49</sup> Voir annexe 5, *Expérimentations avec DNS64 et NAT64* pour plus d'informations.

Bien évidemment, chacune des solutions est à utiliser selon les disponibilités du fournisseur d'accès, les moyens matériels mis à disposition et les connaissances requises pour leur implémentation

Au niveau de l'expérience personnelle, ce travail a été très enrichissant. Ne possédant que de brèves connaissances théoriques en IPv6, j'étais loin de me douter du potentiel de ce protocole et des possibilités qui en découlent.

L'apprentissage des différentes implémentations possibles m'ont également fait comprendre que le monde informatique est lentement en train de basculer vers le tout IPv6, et que des connaissances dans ce domaine sont nécessaires si on ne veut pas rester dans le passé. J'ai également appris que la communication et la cohabitation entre les deux protocoles sont souvent difficiles et que de nombreux problèmes peuvent en découler.

La partie pratique n'a pas été de tout repos, et m'a permis de mettre en pratique mes connaissances acquises du protocole IPv6. Cela a été très enrichissant autant d'un point de vue personnel que professionnel.

Finalement, il a fallu gérer mon temps, m'organiser, faire avec les imprévus et les problèmes techniques, ce qui a été également une grande source d'apprentissage !

# Bibliographie

- **WEBOGRAPHIE**

RIP NCC. NETWORK COORDINATION CENTRE. *Site internet du registre internet national RIP NCC* [en ligne] <http://www.ripe.net/> (Consulté du 23 septembre au 15 novembre 2012)

POTAROO. IPV4 ADDRESS REPORT. *Statistiques sur l'allocation des adresses IPv4* [en ligne] <http://www.potaroo.net/tools/ipv4/index.html> (Consulté le 23 septembre 2012)

HILCO STREAMBANK. IPv4 : *Site internet de la société Hilco Streambank, page sur le marché IPv4* [en ligne] <http://www.hilcostreambank.com/IPv4.asp> (Consulté le 25 septembre 2012)

INTERNET SOCIETY. *Site internet de l'internet society* [en ligne] <https://www.internetsociety.org/> (consulté du 17 septembre au 30 septembre 2012)

WORLD IPV6 LAUNCH : *Site internet de l'événement world ipv6 launch* [en ligne] <http://www.worldipv6launch.org/> (consulté du 17 septembre au 30 septembre 2012)

SWISS IPV6 COUNCIL : *Site de l'organisme Swiss IPv6 Council* [en ligne] <http://www.swissipv6council.ch/fr> (consulté du 17 septembre au 30 septembre 2012)

SWISSCOM. *Swisscom s'engage à l'occasion du World IPv6 Day* [en ligne] <http://www.swisscom.ch/solutions/News-Dialogue-fr/Swisscom-s-engage-a-l-occasion-du-World-IPv6-Day> (consulté le 22 septembre 2012)

SUNRISE COMMUNICATIONS AG. *Business-sunrise lance le premier produit avec soutien IPv6* [en ligne] <http://www.presseportal.ch/fr/pm/100000688/100709065/business-sunrise-lance-le-premier-produit-avec-soutien-ipv6> (consulté le 22 septembre 2012)

VTX. *IPv6 with VTX* [en ligne] <http://ipv6.vtx.ch/ipv6/index.php?p=3> (consulté le 22 septembre 2012)

FHADJ. NAT64, NAT46 et NAT66 disponible dans les produits de sécurité Cisco : ASA 5500. *Le blog IPv6blog.cisco.fr* [en ligne]. Mis en ligne le 4 octobre 2012. <http://ipv6blog.cisco.fr/2012/10/04/nat64-nat46-et-nat66-disponible-dans-les-produits-asa/> (consulté le 4 octobre 2012)

- **DOCUMENTS ELECTRONIQUES**

MILTON, Mueller, BRENDEN, Kuerbis, HADI, Asghari. *Dimensioning the Elephant : An empirical analysis of the IPv4 number market* [en ligne]. 2012, 14p.

[internetgovernance.org/wordpress/wp-content/uploads/IPv4marketTPRC20121.pdf](http://internetgovernance.org/wordpress/wp-content/uploads/IPv4marketTPRC20121.pdf)

(consulté du 24 septembre au 30 septembre 2012)

PERREAULT, Simon, DIONNE, Jean-Philippe, BLANCHET, Marc. *Ecdysis : Open-Source DNS64 and NAT64* [en ligne] 2010, 7 p.

<http://www.viagenie.ca/publications/2010-03-13-asiabsdcon-nat64.pdf> (consulté du 30

septembre au 10 octobre)

HAZEYAMA, H., UENO, Y. *Experiences from an IPv6 only-network in the WILD Camp Autumn 2011* [en ligne] Internet-draft, 2012, 20 p. [http://tools.ietf.org/pdf/draft-](http://tools.ietf.org/pdf/draft-hazeyama-widcamp-ipv6-only-experience-00.pdf)

<http://tools.ietf.org/pdf/draft-hazeyama-widcamp-ipv6-only-experience-00.pdf> (consulté du 30 septembre au 10

octobre)

ARKKO, J., KERANEN, A. *Experiences from an IPv6 only-network* [en ligne] Internet-draft, 2011, 20 p. <http://tools.ietf.org/html/draft-arkko-ipv6-only-experience-04> (consulté

du 30 septembre au 10 octobre)

HERRB, Matthieu. *Comment vivre avec un réseau IPv6 pur ?* [en ligne] CNRS LAAS, 2011, 8p. [https://2011.jres.org/archives/57/paper57\\_article.pdf](https://2011.jres.org/archives/57/paper57_article.pdf) (consulté du 30

septembre au 10 octobre)

- **DOCUMENTATION TECHNIQUE & GENERALE**

WIKIPEDIA. *Site de l'encyclopédie libre* [en ligne]

[http://fr.wikipedia.org/wiki/Wikip%C3%A9dia:Accueil\\_principal](http://fr.wikipedia.org/wiki/Wikip%C3%A9dia:Accueil_principal) (consulté du 17

septembre au 15 novembre 2012)

MICROSOFT TECHNET. *Ressources Windows pour professionnels* [en ligne]

<http://technet.microsoft.com/fr-fr/default> (consulté du 2 octobre au 15 novembre 2012)

CISCO. *Implementing NAT-PT for IPv6* [en ligne], 2010

[http://www.cisco.com/en/US/docs/ios/ipv6/configuration/guide/ip6-nat\\_trnsln.pdf](http://www.cisco.com/en/US/docs/ios/ipv6/configuration/guide/ip6-nat_trnsln.pdf)

(consulté du 2 octobre au 15 novembre 2012)

CISCO. *Cisco IOS IPv6 command reference* [en ligne] 2011

[http://www.cisco.com/en/US/docs/ios/ipv6/command/reference/ipv6\\_book.html](http://www.cisco.com/en/US/docs/ios/ipv6/command/reference/ipv6_book.html)

(consulté du 2 octobre au 15 novembre 2012)

LINUX IPv6 HOW TO. *Router advertisement daemon (radvd)* [en ligne] 2009  
<http://www.tldp.org/HOWTO/Linux+IPv6-HOWTO/hints-daemons-radvd.html> (consulté du 2 octobre au 15 novembre)

SIXXS. *Aiccu / Installation on Ubuntu* [en ligne] 2010  
<http://www.sixxs.net/wiki/Aiccu/InstallationOnUbuntu> (consulté du 2 octobre au 15 novembre)

GOGO6. *How to configure the gogoCLIENT for Freenet6 Tunnel on Windows* [en ligne] 2009  
<http://www.gogo6.com/forum/topics/how-to-configure-the> (consulté du 2 octobre au 15 novembre)

## Annexe 1

### Configuration du tunnel Hurricane Electric

```
HedgeRouteur#show run
Building configuration...

Current configuration : 2065 bytes
!
version 12.3
service timestamps debug datetime msec
service timestamps log datetime msec
service password-encryption
!
hostname HedgeRouteur
!
boot-start-marker
boot-end-marker
!
enable secret 5 $1$zMw$uCDCRF1D0keFgup4Xkhtv0
!
no network-clock-participate slot 1
no network-clock-participate wic 0
aaa new-model
!
aaa session-id common
ip subnet-zero
ip cef
!
ip domain name protoHEG
ip name-server 192.168.3.3
!
ipv6 unicast-routing
!
username admin secret 5 $1$K7be$xe11YgAUAJvHi8NIh5t.7.
!
interface Tunnel0
```



```
no ip address
ipv6 address 2001:470:1F12:101D::2/64
ipv6 enable
tunnel source 172.18.67.190
tunnel destination 216.66.84.42
tunnel mode ipv6ip
!
interface FastEthernet0/0
ip address 172.18.67.190 255.255.255.192
ip nat outside
duplex auto
speed auto
!
interface FastEthernet0/1
no ip address
duplex auto
speed auto
!
interface FastEthernet0/1.1
encapsulation dot1Q 2
ip address 192.168.3.1 255.255.255.240
ip nat inside
ipv6 address FD00:0:0:1::1/64
ipv6 enable
!
interface FastEthernet0/1.2
encapsulation dot1Q 3
ip nat inside
ipv6 address 2001:470:CCD8:1::1/64
ipv6 enable
ipv6 nd ra-interval 60
ipv6 nd ra-lifetime 180
ipv6 nd prefix 2001:470:CCD8:1::/64
!
interface FastEthernet0/1.3
encapsulation dot1Q 4
ipv6 address 2001:470:CCD8:2::1/64
```

```
ipv6 enable
ipv6 nd ra-interval 60
ipv6 nd ra-lifetime 180
ipv6 nd prefix 2001:470:CCD8:2::/64
!
ip nat inside source list 100 interface FastEthernet0/0 overload
ip http server
ip classless
ip route 0.0.0.0 0.0.0.0 172.18.67.129
!
access-list 100 permit ip 192.168.3.0 0.0.0.15 any
access-list 100 permit ip 192.168.4.0 0.0.0.15 any
ipv6 route ::/0 Tunnel0
!
line con 0
password 7 05080F1C2243
line aux 0
line vty 0 4
password 7 02050D480809
transport input all
!
!
end
```

## Annexe 2

### Script de démarrage d'AICCU

```
Update-rc.d -f aiccu remove
Vi /etc/netork/if-up.d/aiccu
# !/bin/bash
/etc/init.d/aiccu start

Chmod a+x /etc/network/if-up.d/aiccu

Vi /etc/netork/if-down.d/aiccu
# !/bin/bash
/etc/init.d/aiccu stop

Chmod a+x /etc/network/if-down.d/aiccu
```

Source : <http://weblog.frlinux.net/?p=524>

## Annexe 3

# Fonctionnement IPv6 avec tunnel SixXs

The screenshot shows a web browser window with the address bar displaying 'test-ipv6.com'. The page has a navigation bar with links: 'Test IPv6', 'FAQ', 'World IPv6 Launch', 'Local Times', 'Mirrors', and 'Stats'. The main heading is 'Test your IPv6 connectivity.' Below this is a tabbed interface with 'Summary' selected. The summary section contains several status messages:

- Your IPv4 address on the public Internet appears to be 212.1.1.1
- Your IPv6 address on the public Internet appears to be 2a02:2528:ff00:175::2
- Your IPv6 service appears to be: chgva01.sixxs.net ipmax
- The [World IPv6 Launch](#) day is June 6th, 2012. **Good news!** Your current browser, on this computer and at this location, are expected to keep working after the Launch. [\[more info\]](#)
- ✓ Congratulations! You appear to have both IPv4 and IPv6 Internet working. If a publisher publishes to IPv6, your browser will connect using IPv6. Your browser prefers IPv6 over IPv4 when given the choice (this is the expected outcome).
- ✓ Your DNS server (possibly run by your ISP) appears to have IPv6 Internet access.

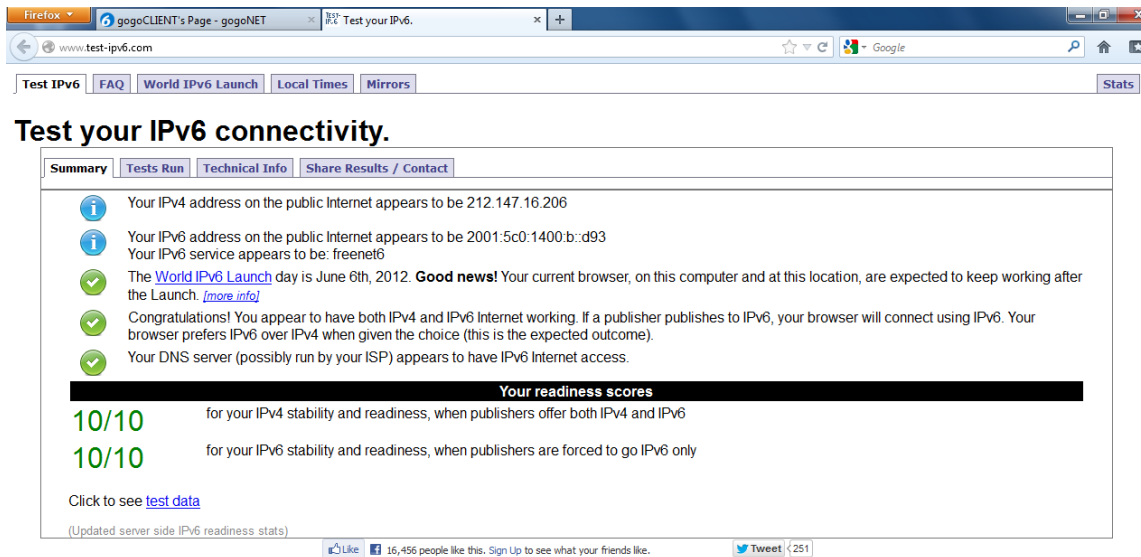
Below these messages is a section titled 'Your readiness scores' with a black background header:

- 10/10** for your IPv4 stability and readiness, when publishers offer both IPv4 and IPv6
- 10/10** for your IPv6 stability and readiness, when publishers are forced to go IPv6 only

At the bottom, there is a link 'Click to see [test data](#)' and a small note '(Updated server side IPv6 readiness stats)'.

## Annexe 4

# Fonctionnement IPv6 avec tunnel Freenet6



The screenshot shows a Firefox browser window with the URL [www.test-ipv6.com](http://www.test-ipv6.com). The page has a navigation bar with links: Test IPv6, FAQ, World IPv6 Launch, Local Times, Mirrors, and Stats. The main heading is "Test your IPv6 connectivity." Below this is a tabbed interface with "Summary" selected. The summary section contains several status items:

- Your IPv4 address on the public Internet appears to be 212.147.16.206
- Your IPv6 address on the public Internet appears to be 2001:5c0:1400:b:d93
- Your IPv6 service appears to be: freenet6
- The [World IPv6 Launch](#) day is June 6th, 2012. **Good news!** Your current browser, on this computer and at this location, are expected to keep working after the Launch. [\[more info\]](#)
- ✓ Congratulations! You appear to have both IPv4 and IPv6 Internet working. If a publisher publishes to IPv6, your browser will connect using IPv6. Your browser prefers IPv6 over IPv4 when given the choice (this is the expected outcome).
- ✓ Your DNS server (possibly run by your ISP) appears to have IPv6 Internet access.

Below these items is a section titled "Your readiness scores" with two scores:

- 10/10** for your IPv4 stability and readiness, when publishers offer both IPv4 and IPv6
- 10/10** for your IPv6 stability and readiness, when publishers are forced to go IPv6 only

At the bottom, there is a link to [test data](#) and a note "(Updated server side IPv6 readiness stats)". Social media sharing buttons for Facebook and Twitter are also present.

## Annexe 5

### Configuration du Routeur NATPT

```
Building configuration...

Current configuration : 2273 bytes
!
version 12.4
service timestamps debug datetime msec
service timestamps log datetime msec
service password-encryption
!
hostname RouteurNATPT
!
boot-start-marker
boot-end-marker
!
enable secret 5 $1$SZqY$TFevIX68ldDLM.COErHFU/
!
no aaa new-model
memory-size iomem 10
no network-clock-participate slot 1
no network-clock-participate wic 0
ip cef
!
!
!
!
ip name-server 192.168.3.3
ip auth-proxy max-nodata-conns 3
ip admission max-nodata-conns 3
!
ipv6 unicast-routing
ipv6 cef
ipv6 dhcp pool poolVlan3
prefix-delegation fd00:1234:5678:beef::/64 00030001000DED191F80
prefix-delegation pool poolVlan3 lifetime 1800 60
dns-server 2001::C0A8:303
domain-name proto.heg
!
!
!
!
username admin secret 5 $1$oVY6$u8uBIE5SD8RHJqUzJVcfb0
!
!
ip ssh version 1
!
!
interface FastEthernet0/0
ip address 192.168.3.2 255.255.255.240
```

```

duplex auto
speed auto
ipv6 enable
ipv6 nd suppress-ra
ipv6 nat
!
interface Serial0/0
no ip address
shutdown
no fair-queue
!
interface FastEthernet0/1
no ip address
duplex auto
speed auto
!
interface FastEthernet0/1.1
encapsulation dot1Q 2
ipv6 address fd00:1234:5678:cafe::1/64
ipv6 enable
ipv6 nd ra-interval 60
ipv6 nd ra-lifetime 180
ipv6 nd prefix fd00:1234:5678:cafe::/64 360 60
ipv6 nat
!
interface FastEthernet0/1.2
encapsulation dot1Q 3
ipv6 address fd00:1234:5678:beef::1/64
ipv6 enable
ipv6 nat
ipv6 dhcp server poolVlan3 rapid-commit
!
interface FastEthernet0/1.3
encapsulation dot1Q 4
ipv6 address fd00:1234:5678:abcd::1/64
ipv6 enable
ipv6 nat
!
interface Serial0/1
no ip address
shutdown
!
interface Serial0/2
no ip address
shutdown
!
ip forward-protocol nd
ip route 0.0.0.0 0.0.0.0 192.168.3.1
!
!
ip http server
no ip http secure-server
!
ipv6 nat translation dns-timeout 120
ipv6 nat v4v6 source 192.168.3.3 2001::C0A8:303

```

```
ipv6 nat v6v4 source list listIPv6 pool ipv4Pool
ipv6 nat v6v4 pool ipv4Pool 192.168.3.5 192.168.3.14 prefix-length 28
ipv6 nat prefix 2001::/96 v4-mapped aclIPv6
!
!
ipv6 access-list listIPv6
permit ipv6 fd00:1234:5678:abcd::/64 any
permit ipv6 fd00:1234:5678:cafe::/64 any
permit ipv6 fd00:1234:5678:beef::/64 any
!
ipv6 access-list aclIPv6
permit ipv6 fd00:1234:5678:abcd::/64 2001::/96
permit ipv6 fd00:1234:5678:cafe::/64 2001::/96
permit ipv6 fd00:1234:5678:beef::/64 2001::/96
!
control-plane
!
!
!
line con 0
password 7 060506324F41
login
line aux 0
line vty 0 4
password 7 0822455D0A16
login
transport input all
!
!
end
```



## Annexe 6

### Expérimentations avec DNS64 et NAT64

Dans le monde, plusieurs personnes ou organismes se sont penchés sur la question de savoir s'ils pouvaient se passer de l'IPv4 dans leur réseau à l'aide des protocoles DNS64 et NAT64 :

- Au Canada, en 2010, la société *Viagénie*, active dans la recherche et le développement des technologies réseaux, a créé le projet *Ecdysis*<sup>50</sup>. Celui-ci a pour but d'offrir une solution open-source à base des deux protocoles.
- Au Japon, en 2011, un réseau purement IPv6 a été instauré et testé sur plus d'une centaine de participants dans le cadre du *WIDE Camp Autumn 2011*<sup>51</sup>.
- En 2011 également, des employés d'Ericsson, une société suédoise de télécommunication, ont testé sur des petits réseaux d'une dizaine de travailleurs leur prototype d'IPv6 pur.<sup>52</sup>
- Toujours en 2011, en France, un ingénieur de l'Université de Toulouse a également tenté de se passer d'IPv4 et a étudié les résultats<sup>53</sup>

D'autres exemples de tests de ces technologies ont été effectués à différentes dates, différents endroits et avec différents matériels, il serait trop long et non pertinent d'en faire la liste complète.

Les conclusions de ces expérimentations sont de part et d'autres pratiquement similaires.

- Le surf sur internet fonctionne correctement, même s'il reste quelques petits problèmes, comme le cas des adresses IPv4 littérales refusant de fonctionner. Les services de messageries fonctionnent correctement.
- Des problèmes ont été repérés sur certains systèmes d'exploitation, dont certains composants réagissent mal avec IPv6 seulement.

---

<sup>50</sup> Source : document *Ecdysis : Open-Source DNS64 and NAT64*

<sup>51</sup> Source : document *Experiences from an IPv6 only-network in the WILD Camp Autumn 2011*

<sup>52</sup> Source : document *Experiences from an IPv6 only-network*

<sup>53</sup> Source : document *Comment vivre avec un réseau IPv6 pur ?*

- Le serveur DNS64 ne renvoie pas une adresse AAAA correcte ou encore certaines pages restant inaccessibles. De plus, il se peut que ce serveur se retrouve surchargé de par le grand nombre d'entrées dans sa table.
- Certains protocoles qui transportent des adresses IP dans la charge utile des paquets (le FTP par exemple) ne fonctionnent pas avec le NAT64, ou très mal.

Ce qu'on peut sortir de toutes ces expérimentations, c'est que la solution est aujourd'hui viable dans un environnement contrôlé, mais souffre encore de quelques bugs et incompatibilités qui seront, espérons-le, corrigés avec le temps et permettront de se passer sans problème d'IPv4 à l'intérieur d'un réseau.

## Annexe 7

### Configuration HedgeRouteur pour tunnel Freenet6

```
HedgeRouteur#show run
Building configuration...

Current configuration : 1986 bytes
!
version 12.3
service timestamps debug datetime msec
service timestamps log datetime msec
service password-encryption
!
hostname HedgeRouteur
!
boot-start-marker
boot-end-marker
!
enable secret 5 $1$zMw$uCDCRF1D0keFgup4Xkhtv0
!
no network-clock-participate slot 1
no network-clock-participate wic 0
aaa new-model
!
!
aaa session-id common
ip subnet-zero
ip cef
!
!
ip domain name protoHEG
ip name-server 192.168.3.3
!
ipv6 unicast-routin
!
!
username admin secret 5 $1$K7be$xe11YgAUAJvHi8NIh5t.7.
```

```

!
!
interface Tunnel0
 no ip address
!
interface FastEthernet0/0
 ip address 172.18.67.190 255.255.255.192
 ip nat outside
 duplex auto
 speed auto
!
interface Serial0/0
 no ip address
 shutdown
!
interface FastEthernet0/1
 no ip address
 duplex auto
 speed auto
!
interface FastEthernet0/1.1
 encapsulation dot1Q 2
 ip address 192.168.3.1 255.255.255.240
 ip nat inside
!
interface FastEthernet0/1.2
 description VLAN 3
 encapsulation dot1Q 3
 ip address 192.168.4.1 255.255.255.240
 ip nat inside
 ipv6 address 2001:5C0:1501:A801::1/64
 ipv6 enable
 ipv6 nd ra-interval 60
 ipv6 nd ra-lifetime 180
 ipv6 nd prefix 2001:5C0:1501:A801::/64
!
interface FastEthernet0/1.3

```

```
description VLAN 4
encapsulation dot1Q 4
ip address 192.168.5.1 255.255.255.240
ip nat inside
ipv6 address 2001:5C0:1501:A800::2/64
ipv6 enable
ipv6 nd suppress-ra
!
interface Serial0/1
no ip address
shutdown
!
interface Serial0/2
no ip address
shutdown
!
interface Serial0/3
no ip address
shutdown
!
ip nat inside source list 100 interface FastEthernet0/0 overload
ip http server
ip classless
ip route 0.0.0.0 0.0.0.0 172.18.67.129
!
!
access-list 100 permit ip 192.168.3.0 0.0.0.15 any
access-list 100 permit ip 192.168.4.0 0.0.0.15 any
access-list 100 permit ip 192.168.5.0 0.0.0.15 any
ipv6 route ::/0 FastEthernet0/1.3 2001:5C0:1501:A800::1
!
!
line con 0
password 7 05080F1C2243
line aux 0
line vty 0 4
password 7 02050D480809
```

```
transport input all
```

```
!
```

```
end
```