Déploiement d'IPv6 dans

les réseaux Enseignement-Recherche de Grenoble

Roger Clot
DSIGU (Direction des Sytèmes d'Information de Grenoble Universités)
Roger.Clot@grenet.fr

Raoul Dorge
DSIGU (Direction des Sytèmes d'Information de Grenoble Universités)
Raoul.Dorge@grenet.fr

Eric Jullien
DSIGU (Direction des Sytèmes d'Information de Grenoble Universités)
Eric.Jullien@grenet.fr

David Rideau
DSIGU (Direction des Sytèmes d'Information de Grenoble Universités)
David.Rideau@grenet.fr

Résumé

La DSI Grenoble Universités (ex CICG - Centre Interuniversitaire de Calcul de Grenoble) assure la mutualisation des réseaux des différents établissements universitaires grenoblois ainsi que ceux des grands organismes de recherche, par l'exploitation du réseau métropolitain Tigre. L'ensemble de ces réseaux est ainsi interconnecté avec le réseau régional Amplivia et le réseau national RENATER.

Parallèlement et depuis de nombreuses années, l'IMAG (Institut de Mathématiques Appliquées de Grenoble) et l'INRIA (Institut National de Recherche en Informatique et Automatisme) Rhône-Alpes travaillent dans le domaine d'IPV6. Ces derniers mois, un transfert d'expertise a été progressivement réalisé entre l'IMAG et la DSI Grenoble Universités, car l'utilisation de cette technologie devient une réalité, transitant ainsi du monde expérimental vers un cadre d'exploitation quotidienne.

La DSI Grenoble Universités dispose d'une expérience reconnue dans le déploiement et l'exploitation des réseaux. Par ailleurs, elle participe sur le plan national au projet IPv6-Adire, initié par la Direction de la Recherche du Ministère de l'Éducation Nationale, de l'Enseignement Supérieur et de la Recherche (la coordination de ce projet a été confiée à l'Université Louis Pasteur avec le soutien de Jean-Paul Le Guigner). A ce niveau, un ensemble d'actions coordonnées a été défini. C'est dans ce cadre plus particulier du projet de déploiement d'IPv6 que la DSI Grenoble Universités souhaite faire partager son expérience en détaillant les différents aspects à prendre en compte pour un déploiement réussi.

Mots clefs

IPv6 Adire, plan d'adressage, routage, filtrage, Mobilité, dual stack, Dhcp v6, Autoconfiguration, Source Address Selection, Privacy

1 Contexte et objectifs

1.1 Le contexte

Les experts universitaires de Grenoble travaillent depuis plusieurs années sur l'IPv6 notamment à travers la mise en place du réseau expérimental IPv6 mondial (6bone) en partenariat avec des instituts tels que l'IMAG et l'INRIA. Dans ce contexte riche en pionniers de la technologie, la DSI Grenoble Universités qui opère les réseaux fédérant la majeure partie des sites Enseignement Recherche de Grenoble (CNRS, INRIA, IUFM, Synchrotron, etc.) a voulu faire un pas de plus vers le déploiement de ce protocole à grande échelle.

Parallèlement à cette prise de conscience des avantages offerts par IPv6 (évolution et simplicité de connexion de bout en bout), le Ministère de l'Éducation Nationale a décidé de promouvoir le déploiement systématique du protocole au sein des Universités. Ainsi, un groupe de travail a été créé au niveau national (intitulé IPv6 Adire[1]) dans lequel Grenoble s'est tout naturellement impliqué.

1.2 Les objectifs

Après divers maquettages (tests d'architectures, protocoles, sécurité, …) et pré déploiements, l'objectif de la DSI

[1] http://ipv6.u-strasbg.fr/doku.php

Grenoble Universités est désormais de déployer (ou d'aider à déployer) IPv6 chez tous les acteurs de l'enseignement supérieur et de la recherche de Grenoble.

Les premiers travaux du groupe de travail national ont mis en évidence toute une série d'étapes à franchir afin de mener à bien une telle évolution des infrastructures. L'article s'attachera donc à les balayer dans l'ordre et précisera pour chacune comment elles ont été remplies.

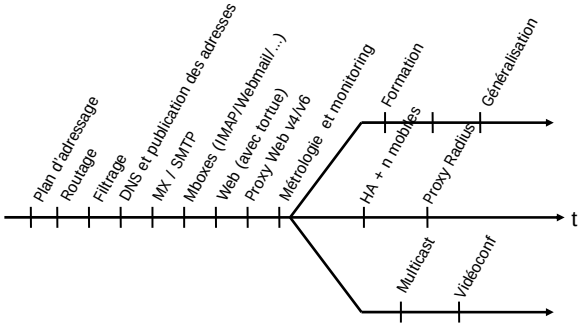

Figure 1 – *Carte routière du déploiement*

La figure ci-dessus reprend une carte routière qui a été établie au cours des réunions du groupe de travail IPv6Adire. On note toute une série d'étapes successives à mener à bien si l'on veut réussir un tel déploiement. A partir de la fourche, plusieurs options sont possibles et nous verrons dans la suite de cet article quels sont les choix effectués pour notre déploiement.

Le schéma de notre article est donc simple, il reprend pas à pas les étapes répertoriées dans la figure ci-dessus, et précise pour chacune les choix réalisés, les contraintes observées et les résultats obtenus.

Enfin nous avons jugé nécessaire de traiter deux points qui ne figurent pas dans la carte routière d'origine et que nous avons déclinés sous forme de paragraphes dans cet article :

- La gestion de services spécifiques : En plus des services de bases tels que le DNS, la messagerie, etc., nous avons déployé d'autres services que nous décrirons également dans cet article.
- La gestion des postes de travail : Il nous a paru également important de s'intéresser aux différentes options qui nous sont offertes pour le déploiement des postes de travail. Là encore un chapitre spécifique est prévu pour traiter ce point important.

2 Le plan d'adressage

Etape obligatoire, le plan d'adressage est tout naturellement le premier point technique à régler. Contrairement à IPv4 et ses types de classes, IPv6 impose une certaine structuration des plans d'adressage. En effet, la hiérarchisation des adresses IPv6 est obligatoire. Du coté Enseignement supérieur et recherche, RENATER qui joue le rôle d'opérateur, nous attribue ce que l'on appelle un préfixe que l'on peut ensuite « découper » à notre guise.

2.1 Le plan d'adressage sur Renater

Toute adresse IPv6 est constituée de 128 bits. RENATER en temps qu'opérateur s'est vu attribuer un préfixe de 32 bits par le RIPE NCC[2] (http://www.ripe.net). Une fois son plan d'adressage réalisé, chaque « client » de chaque Nœud RENATER (ou NR) se voit allouer à son tour un « sous-préfixe », sous-ensemble du préfixe de RENATER qu'il peut à son tour découper en différentes plages. Cela donne :

- La première partie de l'adresse est 2001:0660:: (Préfixe en /32 alloué à RENATER).
- Les 16 bits suivants définissent l'identifiant du NR (8 bits) et l'identifiant du site (8bits).

  Pour Grenoble, l'identifiant de NR est 53 (préfixe en /40), chaque site se voyant attribuer un NLA-ID[3] sur les 8 bits suivants (en /48). La figure 2 ci-dessous détaille les différents NLA-ID attribués aux établissements grenoblois.
- Il reste alors 16 bits (SLA[4]) dans la partie de l'identifiant du réseau. Chaque site est alors libre d'utiliser et gérer cette partie comme il l'entend.

En ce qui concerne les acteurs Enseignement Sup. et recherche, RENATER nous a attribué plusieurs préfixes en /48 que l'on peut découper en deux familles :

- Les préfixes utilisateurs (/48),

  Ces préfixes « utilisateurs » sont des préfixes alloués à chaque entité fédérée sur le NR grenoblois : on peut regrouper dans cette famille les Universités et Instituts de Grenoble, et comme exemple citons le préfixe attribué à l'IMAG : 2001 :660 :5301 /48.
- Les préfixes réseaux de collecte (/48).

  Tigre jouant le rôle d'opérateur métropolitain, il s'est vu attribuer en propre un préfixe permettant de créer son propre niveau d'adressage pour les sites raccordés.

---

[2] RIPE NCC : Réseaux IP Européens Network Coordination Centre : organisme chargé de l'attribution des adresses Ip pour l'Europe.

[3] NLA-ID : Network Level Aggregator- IDentification

[4] SLA-ID : Site Level Aggregator - IDentification

Le préfixe attribué dans ce cas est le 2001 :660 :2408 /48

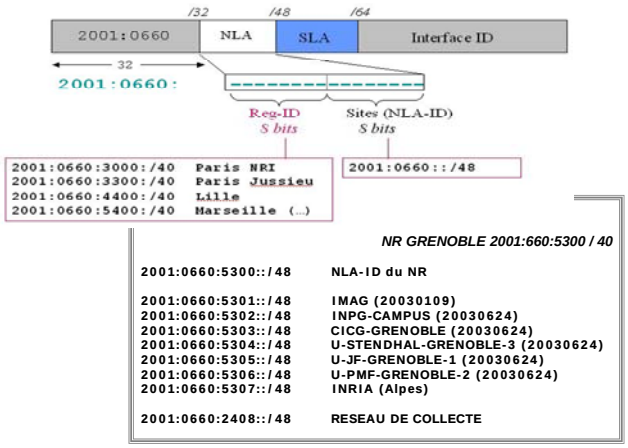

Figure 2 – *Addressage IPv6 Renater et NR Grenoblois*

Remarque : On constate que RENATER a fait le distinguo entre préfixes attribués aux entités (qui respectent une certaine hiérarchie simplifiant par extension le routage) et ceux distribués aux réseaux dits de collecte. Ainsi les préfixes « réseaux de collecte » sont attribués dans une autre plage que celle dédiée aux NR.

2.2 Le plan d'adressage Métropolitain Tigre

En temps qu'opérateur métropolitain nous avons eu toute latitude pour établir un plan d'adressage. Grenoble étant composé de plusieurs sites répartis sur l'agglomération, nous avons choisi une solution simple.

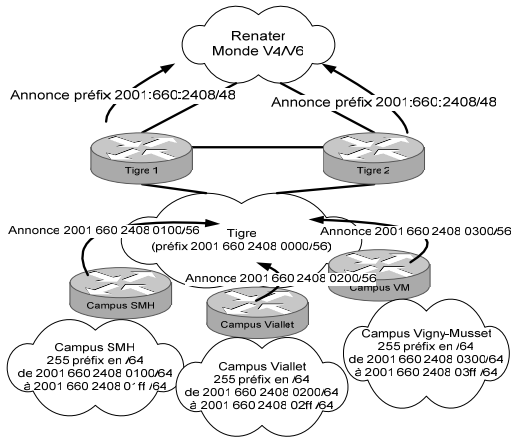

Figure 3 – *Addressage IPv6 pour le réseau métropolitain*

La figure 3 ci-dessus n'est pas exhaustive du nombre de sites gérés, mais elle est représentative de la manière dont sont attribués les préfixes.

L'idée est de réaliser un découpage hiérarchique sur deux niveaux :

- Au niveau Tigre, préfixes en /56 (255 campus possibles),
- Sur chaque campus, préfixes en /64 (255 sous réseaux possibles par campus).

2.3 Le plan d'adressage ipv6 sur les sites

Les Universités et établissements associés sont souvent répartis sur les différents campus. Comme ils possèdent chacun un préfixe /48, le découpage est laissé à leur discrétion, chacun pouvant décider un découpage conforme au RFC 2450[5] ou non. Aucune règle n'a été fixée à ce niveau.

3 Le routage

Le service IPv6 est déjà disponible dans chaque point de présence régional de l'épine dorsale RENATER (les routeurs étant « double pile[6] »). Cela facilite donc tout déploiement d'IPv6 dans les réseaux de collecte et au-delà.

3.1 Le routage au niveau métropolitain

Une fois le plan d'adressage entériné, il a fallu s'intéresser aux protocoles de routage à mettre en œuvre.

Pour ce qui est du réseau métropolitain et de l'accès à RENATER, nous avons tout naturellement tenté de pérenniser la solution existante. En effet, notre réseau métropolitain étant déjà composé de sessions mBGP[7] entre les différents acteurs de la plaque, nous avons tout simplement reconduit le principe en IPv6 pour le raccordement de nos sites.

Avantages :

- L'investissement humain sur mBGP pour IPv6 quand on maîtrise mBGP pour IPv4 est négligeable,
- La majorité des routeurs de l'architecture métropolitaine pouvait supporter une mise à jour logicielle afin d'adjoindre le routage d'IPv6 en fonctionnement double pile (IPv4/IPv6).

Remarques :

L'étape de déploiement n'est toutefois pas achevée car certains routeurs ne sont pas aptes à basculer en double

[5] le RFC 2450 préconise une certaine hiérarchisation au niveau des préfixes en fonction de critères tels que communauté d'utilisateurs, sous réseaux, composantes, etc.

[6] Un routeur dit « double pile » est un routeur capable de traiter à la fois les paquets IPv4 mais aussi IPv6.

[7] mBGP voulant dire multiprotocol BGP et non pas multicast BGP. Pour l'intégration d'IPv6 dans BGP, consultez le RFC 2545

pile. Nous attendrons autant que faire se peut leur renouvellement pour irriguer les derniers sites de manière native. Dans le cas contraire, (le cas ne s'est pas encore posé), nous verrons en détail plus loin dans ce document que deux solutions peuvent leur être proposées en attendant la mise à jour de l'équipement métropolitain :

- Utilisation d'un routeur IPv6 dédié (en parallèle du routeur IPv4 actuel) à base de logiciels libres ;
- Utilisation de tunnels IPv6/IPv4 permettant d'encapsuler les paquets IPv6 dans de l'IPv4.

3.2 Le routage au niveau des sites

A ce niveau de réseau, le protocole majoritairement utilisé à l'heure actuelle est OSPF v2. Là encore, pour un investissement minimum (tant sur le plan humain que matériel) il nous a paru tout naturel d'utiliser le pendant du protocole IPv4 en version IPv6 : OSPF v3. Il n'est cependant pas exclu de passer à IS-IS plus tard pour des raisons de simplification de tables de routage (IS-IS n'utilisant qu'une seule table pour les deux protocoles).

Les différences entre les deux versions de protocoles se résument à peu de choses qui ne remettent pas fondamentalement en cause l'architecture actuelle (différences minimes au niveau des LSA[8], …). Le seul problème rencontré est, encore une fois, la non disponibilité du protocole de routage au niveau des équipements déjà en place. Ainsi, deux types de traitement sont utilisés comme le montre la figure 4 ci-dessous.

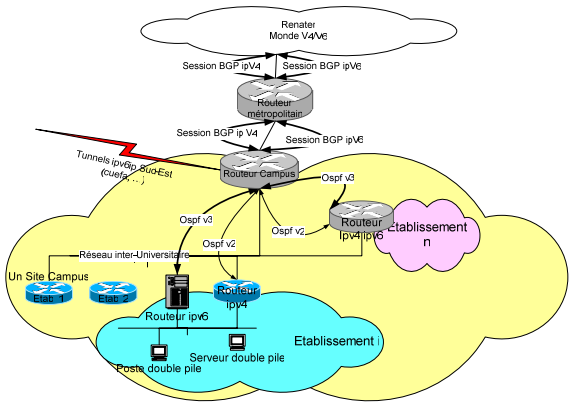

Figure 4 – *Synthèse du routage*

Pour les établissements disposant d'équipements de routage capables d'intégrer la double pile V4/V6 ainsi que le protocole ospf-v3 : la configuration OSPF v3 se superpose tout naturellement à celle d'OSPF v2 (établissement n sur la figure 4).

[8]LSA : Link State Advertisement

Pour les autres, un routeur capable de traiter IPv6 leur est proposé. Ainsi le réseau de l'établissement est irrigué par deux routeurs spécifiques : IPv4 et IPv6 (établissement i sur la figure 4).

Le paragraphe suivant traite de la solution retenue pour effectuer ce type de service de manière efficace et peu onéreuse.

3.3 Les routeurs à base de logiciels libres

Si l'on prend l'exemple de la DSI Grenoble Universités, le routeur IPv4 utilisé actuellement n'est pas capable de traiter nativement le protocole IPv6. Nous avons donc mis en place un second routeur spécifique.

Coté architecture, cela donne donc deux routeurs ; un routeur IPv4 et un routeur IPv6. Entre ces deux routeurs, un lien de type 802.1Q permet de faire transiter tous les VLANS qui doivent recevoir IPv4 et IPv6.

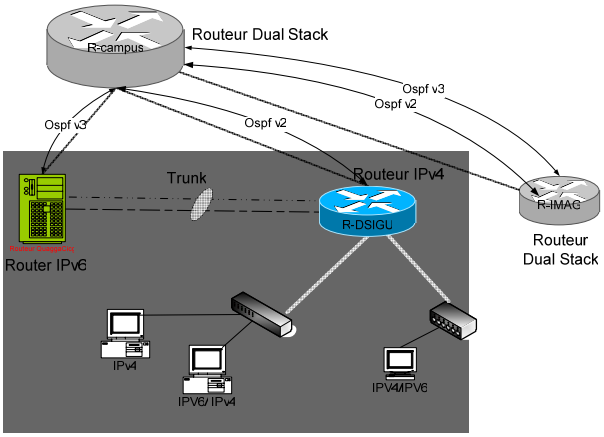

Figure 5 – *Exemple de routage dissocié entre IPv4 et IPv6*

3.3.1 La Plateforme matérielle

Etant donné la philosophie retenue pour le routage (à base de logiciel libre), il nous a fallu chercher une plateforme matérielle bon marché capable de performances importantes. Notre choix s'est porté sur un matériel de type DELL PowerEdge 1850.

Ce Serveur au format rack 1U, équipé de la technologie PCI Express nous offre une puissance et performance suffisante pour une utilisation de type routeur. Coté réseau, il est possible d'équiper la machine de plusieurs ports Giga, la bande passante annoncée pouvant aller jusqu'à 64 Gb/s.

3.3.2 La Plateforme logicielle

Pour nos tests préliminaires, les serveurs routeurs ont utilisé deux systèmes d'exploitation : linux et FreeBSD.

- Du coté de Linux, les premiers tests ont été effectués sur une plateforme Redhat. Rapidement,

de par sa facilité de mise en œuvre, nous avons opté pour une solution packagée Devil Linux. En effet, il suffit d'un CD bootable pour gérer le système, une clé USB, voire même une disquette pour stocker la configuration. Nous avons cependant rencontré des difficultés pour la mise en œuvre du filtrage IPv6 : ip6tables (inclus dans le produit) est, de notre point de vue, assez lourd et ne possède pas de mode stateful (http://www.iptables.org/ 9/9/2005).

- Nous nous sommes donc tournés vers FreeBSD. L'installation du système est plus compliquée mais ce système d'exploitation bénéficie de couches réseaux performantes, il est très répandu et bénéficie de nombreuses contributions dans le domaine des réseaux (nous verrons plus loin dans cet article que FreeBSD dispose de nombreux produits en « standard » pour la sécurité et qu'il fait partie des produits fréquemment utilisés pour l'implémentation des fonctions de mobilité en IPv6).

A l'heure de l'écriture de ces pages, nous n'avons pas terminé nos tests mais FreeBSD est le système que nous avons retenu pour la base système d'exploitation de nos routeurs libres. Il nous reste désormais à trouver le pendant de la version bootable CD Devil linux sur une base FreeBSD (xorp semble un bon prétendant), ce qui simplifierait l'exploitation dans un déploiement à plus grande échelle.

3.3.3 Le logiciel de routage

Quelle que soit la plateforme déployée, le routage est assuré par le logiciel quagga (http://www.quagga.net). Quagga est une branche de ZEBRA qui implémente OSPFv2 OSPFv3, RIP v2, RIPng, mBGP, etc.

L'architecture est composée de plusieurs démons : Zebra, ospfd, ospf6d, bgpd. En ce qui concerne la syntaxe, on retrouve fortement celle de l'IOS des routeurs Cisco ce qui veut dire que la prise en main est assez rapide pour tout connaisseur…

Pour notre déploiement, nous avons essentiellement utilisé la partie OSPF v3. A noter quelques soucis de fonctionnement lors de la configuration d'Area filles. Sinon, globalement le service fonctionne de manière fiable et durable comme sur un routeur utilisant de l'OSPF v2.

4 La sécurité

Ce paragraphe ne traite que de l'aspect filtrage/firewalling déployé sur les différents équipements de réseaux qui permettent de transmettre les paquets IPv6.

4.1 Sur les routeurs existants

Au niveau métropolitain et inter-universitaire, nous disposons majoritairement de routeurs de marque Cisco (6000, 7000). La sécurité déployée en IPv6 est, comme en IPv4, basée sur des ACL[9].

Avec les versions d'IOS déployées, nous disposons d'ACLs standards et étendues, comme en IPv4. On dispose également d'ACLs créées dynamiquement, dites réflexives. Si la mise en œuvre d'ACLs IPv6 diffère peu de celles en IPv4, il existe quand même quelques subtilités liées au protocole (une ACL vide correspond a un permit ip any any, et une règle implicite autorise les messages ICMPv6 qui permettent la découverte de voisins[10] par exemple).

Avec un peu de recul, il est préférable de bien intégrer la cinématique des échanges IPv6 avant tout déploiement de filtrage, des effets de bord désagréables pouvant survenir si l'on ne la maîtrise pas correctement (lorsqu'un poste IPv6 cherche à s'insérer dans un réseau par exemple)…

Enfin nous avons rencontré quelques problèmes de cohabitation d'ACL en V4 et V6 mais cela reste marginal[11].

4.2 Sur les routeurs à base de logiciels libres

Plusieurs types de filtrage ont été testés d'abord sous Linux puis sous FreeBSD puisqu'il reste le seul système d'exploitation privilégié pour le routage :

- IPFILTER (IPF) Firewall ;
- IPFIREWALL (IPFW) ;
- Packet Filter (PF) de OpenBSD.

Dans les dernières versions de FreeBSD, PF est le firewall d'OpenBSD intégré en standard (contrairement aux versions précédentes).

Pour résumer, PF est celui que nous avons retenu :

- Il est performant (utilisé dans certaines applications conséquentes et à haut débit) ;
- Au-delà de la syntaxe très simplifiée, un autre gros avantage de **pf** sur les systèmes de filtrage sous Linux est qu'il est possible de charger l'ensemble des règles en une seule commande ;
- L'enregistrement des événements est écrit sous format binaire compatible tcpdump.

---

[9] ACL : Access Control Lists

[10] Paquets de type Neighbor Discovery

[11] Sur un Cisco 6500 : la présence d'une ACL IPv4 restreignant l'accès à l'équipement empêchait tout accès contrôlé en IPv6…

5 Les services de base

Une fois l'infrastructure opérationnelle et fiable, le déploiement des services de base peut alors commencer.

5.1 Le DNS

Le service de noms est tout naturellement la première brique utilisée.

Avant de « convertir » directement nos DNS d'exploitation IPv4, nous avons dans un premier déploiement déclaré un sous domaine ipv6.grenet.fr délégué à un serveur de tests. Une fois habitués et rassurés sur la fiabilité des produits, la deuxième étape a consisté à transformer le serveur « officiel », un linux redhat, en DNS double pile. La seule vraie difficulté est qu'il faut se familiariser avec la syntaxe des nouvelles adresses IPv6, on peut dire que toute personne maîtrisant les serveurs DNS IPv4 ne devrait pas avoir trop de mal à les convertir en IPv6.

Remarque : une fois cette bascule réalisée, il est impératif d'assurer la continuité de service du DNS en IPv6 autant qu'en IPv4. En effet, une fois l'activation du protocole effectuée, les clients vont prioritairement l'utiliser pour toute résolution de nom. Cela veut dire qu'il faut également assurer la disponibilité de l'infrastructure qui l'irrigue en IPv6 car sinon, gare aux ennuis (attention au filtrage également)!

5.2 La messagerie

Les boîtes aux lettres DSI Grenoble Universités sont hébergées sur un serveur linux Redhat accessible via POP et IMAP. La migration vers IPv6 a été réalisée sans grande difficulté.

Remarque : Un client IPv4/IPv6 essaie tout d'abord une connexion IPv6. Si le service n'est pas actif, on constate des time-out et des ralentissements conséquents. Il est préférable comme pour le DNS que le service soit constamment accessible en IPv4 mais aussi en IPv6…

Du coté de notre relais de messagerie, nous sommes équipés d'une solution redhat double pile également. Le produit de messagerie utilisé à l'heure actuelle étant postfix.

5.3 Le web

C'est le premier service que nous avons mis en œuvre sans aucune difficulté également.

Remarques : ne pas oublier d'adapter les fichiers de restriction d'accès s'ils sont à base d'adresses IPv4.

Ne pas oublier également d'installer le logo Kame constitué d'une tortue qui ne bouge que si l'accès utilisateur est réalisé en IPv6 (http://www.kame.net/) !

6 Les services spécifiques

En plus des services que l'on peut qualifier de standards, nous avons démarré certains services plus spécifiques qui se sont révélés utiles dans notre déploiement. Comme annoncé dans l'introduction, nous allons dans ce paragraphe les aborder rapidement.

6.1 Looking glass

Lorsque l'on déploie à grande échelle un routage IPv6 il est intéressant de disposer d'outils de contrôle permettant à la fois de se familiariser avec le protocole, mais aussi de vérifier que tout est optimal en régime stable. Un service de type looking-glass permet de collecter toute une série d'informations sur les tables de routage des routeurs, sur l'état des sessions BGP, etc. (Il en existe un nombre important dans l'Internet comme on peut le constater à l'adresse suivante http://www.traceroute.org).

Nous avons donc installé un looking-glass sur un serveur FreeBSD (http://lookingglass.ipv6.grenet.fr/). Les options disponibles sont ensuite les mêmes que celles d'un looking-glass IPv4.

6.2 Un service IPv6 dans IPv4 pour les postes de travail : tunnels ISATAP[12]

Ce service est typiquement dédié à une population de postes de travail double pile clairsemés dans un monde purement IPv4.

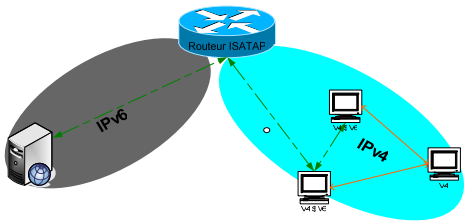

Figure 6 – *Exemple d'architecture ISATAP*

ISATAP établit un tunnel automatique entre un routeur IPv6 et un poste IPv6/v4 dans un réseau IPv4. Nous avons testé cette solution sur des postes Windows XP qui établissent automatiquement des tunnels vers un routeur Cisco de type 7200. La solution fonctionne et permet de traiter simplement des postes de travail isolés dans un monde IPv4.

[12] Intra-Site Automatic Tunnel Addressing Protocol

6.3 Un service IPv6 dans IPv4 pour les sites : tunnels manuels IPv6/IPv4

Ce service est utilisé par des sites qui ne disposent pas d'une connexion native en IPv6 vers l'extérieur. Ainsi ils peuvent quand même utiliser et déployer une infrastructure IPv6 interne en attendant d'être raccordés nativement en IPv6 vers le reste du monde.

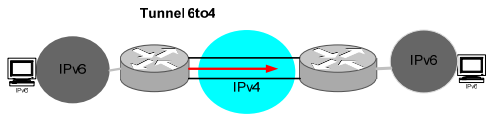

Figure 7 – *exemple de tunnel IPv6/IPv4*

Les paquets IPv6 sont encapsulés dans les paquets IPv4, les tunnels sont cette fois-ci « montés » entre deux routeurs et sont configurés manuellement (à la différence des tunnels ISATAP du paragraphe précédent qui concernent les postes de travail et qui sont montés automatiquement).

Avec la généralisation progressive du protocole et des services IPv6, ce service tend à disparaître. Cependant il est encore utilisé dans certains cas et n'a pas posé de problèmes particuliers dans son déploiement.

6.4 Le service Nat-PT[13]

Dernier type de service spécifique mis en place, une passerelle NAT-PT est un service de translation permettant à un utilisateur IPv6 de communiquer de manière « transparente » avec un utilisateur IPv4 et vice versa. Couplé avec un service de type DNS-ALG[14] (une application permettant la traduction des requêtes DNS entre les deux mondes v6 et v4), la passerelle a été testée avec succès sur maquette.

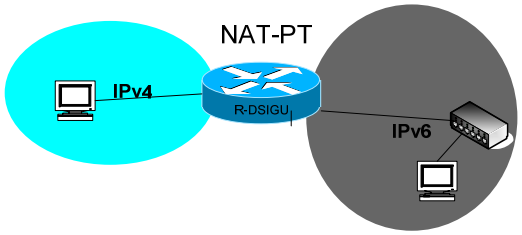

Figure 8 - *Passerelle NAT-PT*

Cependant, aucun déploiement opérationnel n'est pour le moment prévu. Avec du recul, il semblerait que le recours à ce type de solution soit à éviter le plus possible : cette méthode de transition est la plus complexe (elle ne résout en outre pas encore tous les problèmes de traduction) et nécessite de bien maîtriser les différents composants impliqués dans la passerelle.

7 La supervision et la métrologie

7.1 La supervision

Etant donné que nous souhaitions faire évoluer nos outils de supervision, nous avons tenté d'intégrer le critère « IPv6 Compliant » dans notre choix de plateforme. Convaincus par ailleurs de l'intérêt des outils du monde libre et à code ouvert, nous avons donc mené une étude comparative et testé plus ou moins rapidement différentes plateformes de supervision OpenSource capables de superviser un vaste réseau, composés de serveurs hétérogènes, fournissant des services variés. En résumé HPOpenView en mieux et gratuit.

Actuellement, parmi les outils investigués tels Nagios, NMIS et OpenNMS, aucun ne peut prétendre être compatible IPv6, car les évolutions à apporter dans le code touchent tous les aspects de ces outils :

- La pile réseau, socle de l'interaction avec le système pour effectuer des connexions natives IPv6 ;
- Le protocole SNMP et la collection de MIBs trop spécifiquement orientées IPv4, pour lesquelles l'IETF a préféré redéfinir des spécifications plus généralistes. Pendant ce temps, les constructeurs ont défini des formats propriétaires, rendant l'inter-opérabilité plus délicate ;
- Les traitements particuliers pour différents services applicatifs et l'adaptation du pseudo-code client ;
- L'interface utilisateur qui effectue souvent des contrôles de formatage sur la notation décimale pointée, conversions @IP – DNS, etc.

Nous avons choisi de placer notre confiance dans la plateforme OpenNMS (http://www.opennms.org), qui n'offre à ce jour aucune compatibilité IPv6, tout comme les autres, mais qui nous donne entière satisfaction pour la supervision de nos réseaux, serveurs et services IPv4. L'adaptation pour IPv6 est donc un processus auquel nous essaierons de contribuer dans la communauté OpenSource du projet. La conférence du mois de décembre prochain correspondant à cet article, sera l'occasion de faire un point d'avancement concret sur ces tentatives.

7.2 La métrologie

Côté métrologie, les outils que nous utilisons s'appuient sur la technologie Netflow, format propriétaire bien connu, développé par Cisco et pris comme référence par l'IETF pour recommander une implémentation qui s'approcherait au mieux de la définition du standard IPFIX.

---

[13] NAT-PT : Network Address Translation-Protocol Translation (RFC 2766)

[14] DNS-ALG : Domain Name Service – Application Layer Gateway

Nous avons activé l'export de netflows au format V9 sur notre maquette, mais la mise en production sur les routeurs des réseaux métropolitains et de campus grenoblois a été ajournée, en attendant d'autres retours d'expérience plus concluants quant à la stabilité de la fonctionnalité.

Nous continuons donc nos efforts de développement OpenSource autour de la sonde Ephora[15], afin de la rendre capable de décoder des trames IPv6 et de générer ensuite des Netflows V9. Reste à faire évoluer également les plateformes qui collectent et exploitent les-dits Netflows. Nous continuons également notre réflexion sur le projet Netsec[16], dont l'objectif est de proposer une organisation des flux dans un SGBD relationnel afin de pouvoir effectuer des recherches détaillées sur différents critères applicables à la brique élémentaire qu'est le flow. Aucun résultat précis à présenter à l'heure où nous écrivons ces lignes, uniquement des évolutions planifiées pour les toutes prochaines semaines...

8 La mobilité IPv6

Gros apport du nouveau protocole IPv6, la mobilité devrait permettre à tout mobile de garder la même adresse quelque soit le lieu sur lequel il se déplace : les connexions et sessions déjà établies sont alors conservées, une optimisation du routage est même possible entre les différents éléments impliqués dans une communication. Autre avantage du modèle, l'adjonction d'IPsec permet d'assurer une sécurisation devenue nécessaire dans ce contexte de « libre » circulation des mobiles.

C'est dans ce contexte que nous avons réalisé un pilote permettant de jauger de l'opportunité d'un déploiement à grande échelle de ce modèle dans un proche avenir.

8.1 Les éléments utilisés

Trois éléments sont utilisés pour démontrer le fonctionnement d'une solution de mobilité IPv6 :

- **Les Mobile Nodes ou MN** : ce sont les éléments mobiles (Ordinateur portable en général) qui nécessitent de conserver leur adresse IPv6 d'origine quel que soit l'endroit où ils se déplacent.
- **Les Correspondant Nodes ou CN** : ce sont des éléments qui cherchent à établir une connexion avec un Mobile Node (un serveur, un ordinateur personnel, un autre MN, etc.). Ces CN peuvent ou non être conscient de l'utilisation de la fonctionnalité Mobile : deux comportements seront alors possibles (communication optimisée ou via tunnel).
- **Les Home Agent ou HA** : pièce essentielle du dispositif, un HA joue le rôle d'intermédiaire entre un CN et un MN lorsque ce dernier se déplace sur un réseau qui n'est pas celui d'origine. Le HA permet de garder le contact avec tout MN : soit il permettra d'établir un tunnel vers le MN en balade, soit il permettra d'établir un lien direct entre CN et MN si la fonction est autorisée.

Pour notre pilote, nous avons utilisé les éléments spécifiques suivants :

- Pour le Home Agent **(HA)** : routeur Cisco 7200 équipé d'une version d'IOS 12.4.2T
- Pour le Mobile Node **(MN)** : postes clients PC équipés du système d'exploitation linux Fedora Core 3, enrichi des éléments nécessaires au fonctionnement d'un nœud IPv6 mobile : Mipl en version 2 (http://www.mobile-ipv6.org)
- Pour le Correspondant Node **(CN)** : deux types de CN ont été testés, ceux de type Windows non équipés des éléments de mobilité et d'autres mobile IPv6 « aware » à base de Fedora Core 3/ Mipl v2 également (comme les MN). Ces deux types de clients nous ont permis de tester plusieurs cas prévus par le modèle (utilisation du mode tunnel ou en routage direct).

Le premier constat de notre pilote concerne les postes de travail équipés de systèmes d'exploitation Windows. Si à l'heure actuelle une grande partie des ordinateurs portables en sont équipés, il est apparemment impossible à un utilisateur standard de récupérer des couches réseaux mobile IPv6. En conséquence, tous nos tests ont porté sur des clients équipés de systèmes d'exploitation libres.

Le deuxième constat du pilote concerne également la mise en place de la fonctionnalité mobile IPv6 sur les postes clients. De notre point de vue, nous sommes encore dans une phase de pré déploiement de la technologie : il faut à l'heure actuelle installer des patchs ou des packages qui évoluent régulièrement. Pas question encore de trouver la fonction pré compilée ou packagée dans tout système d'exploitation. L'utilisateur doit passer par un processus d'intégration réservé à des initiés.

Pour ce qui est des tests à proprement parler, les résultats sont conformes à ce que l'on trouve dans la littérature. Notre pilote intégrait des implémentations de sources différentes (Cisco et Mipl v2) qui ont inter opéré correctement.

Nous avons par exemple pu faire fonctionner des communications en mode tunnel (cf figure 9) : Le CN initie une connexion vers un MN qui, en cours de communication, change de réseau physique et logique : la session n'est pas rompue et le trafic est encapsulé à partir du HA vers l'adresse où se trouve le Mobile Node (c'est pour le CN totalement transparent).

[15] http://ephora.grenet.fr

[16] http://netsec.grenet.fr

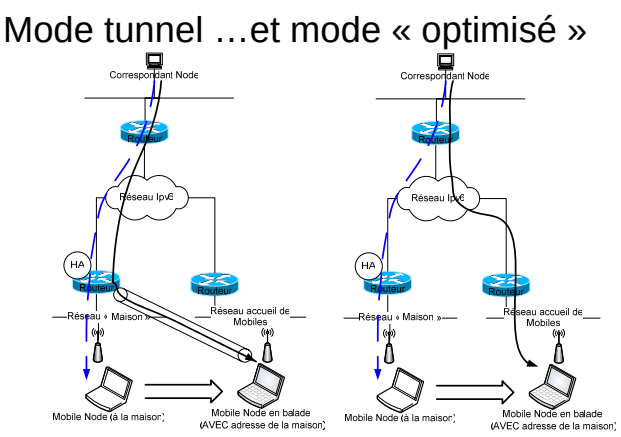

Figure 9 – *Gestion « comparée » d'un Mobile Node*

Nous avons également fait fonctionner des sessions TCP en « mode direct » (cf figure 9 également), avec succès. Dans ce mode de fonctionnement, les deux parties doivent être équipées de la fonctionnalité mobile IPv6.

A l'heure où nous écrivons ces pages nous n'avons pas configuré de Home Agent Cisco capable d'intégrer complètement la fonctionnalité IPsec. Cette fonctionnalité est pourtant essentielle car elle doit permettre de protéger les flux entre MN, CN et HA. Sans elle, pas question de déployer la mobilité car la sécurité des échanges ne peut pas être garantie. Le constructeur nous a informé qu'une version devrait être disponible sous peu. En attendant, un HA à base de logiciel libre (Mipl) est en cours de déploiement dans un contexte sans-fil ARREDU (www.cru.fr/nomadisme-sans-fil/arredu).

9 Du coté des postes de travail

Ce chapitre s'attardera davantage sur les spécificités du poste client lié à IPv6.

9.1 Configuration

Différents modes de configuration IPv6 sont possibles :

- Statique ;
- Stateless address autoconfiguration ;
- Stateful address autoconfiguration ;
- Combinaison des 2 modes précédents.

9.1.1 Configuration statique

C'est le mode le plus fastidieux puisqu'il faut renseigner manuellement les informations de base du client (adresse IPv6, préfixe et passerelle par défaut).

9.1.2 Stateless Address Autoconfiguration

Ce mode de configuration est décrit par le RFC 2462 [1]. C'est un mode automatique dans lequel le client acquiert une partie des informations par les messages périodiques du routeur local. Le reste des informations est complété par le client à partir de données locales. Le routeur local diffuse périodiquement des messages de type « Router Advertisement » qui contiennent le préfixe et sa longueur du sous réseau affecté. Les 64 bits de l'identifiant d'interface sont construits à partir de l'adresse Ethernet du client. Cet identifiant d'interface est alors le même pour les adresses lien local et globale.

9.1.3 Stateful Address Autoconfiguration DHCPv6

Dans ce mode, le client obtient ses paramètres de configuration d'un serveur DHCPv6. Le RFC 3315 [2], qui décrit ce protocole, est relativement récent, ce qui fait que peu d'implémentations de serveur DHCPv6 sont réellement disponibles. Citons :

- Serveur DHCPv6 du projet KAME
- Serveur DHCPv6 « Dibbler » (klub.com.pl/dhcpv6/)
- Serveur DHCPv6 Cisco (IOS ≥ 12.3(4)T)

9.1.4 Combinaison des modes « stateless » et « statefull »

L'inconvénient du mode « stateless » est qu'il ne permet pas de récupérer l'adresse IPv6 du serveur de noms. Le mode « stateless » n'est donc pas réellement automatique s'il est nécessaire de renseigner l'adresse du serveur DNS dans la configuration du résolveur. Il est possible de combiner les modes « stateless » et « statefull » pour bénéficier des avantages propres à chacun. Ainsi la configuration d'adresse IPv6 peut se faire en automatique pour profiter de la facilité de mise en œuvre d'un poste client, et les compléments d'information peuvent provenir d'un serveur (central) DHCPv6. Le RFC 3736 [3] parle d'ailleurs de « Stateless DHCP Service for IPv6 ».

9.2 Sélection de l'adresse source

Un plan d'adressage IPv6 peut comprendre une multitude d'adresses. Parmi celles-ci, des adresses unicast de différentes étendues (« link-local », « site-local » ou « global »), peuvent être « preferred » ou « deprecated ». La question du choix des adresses source et destination opérée par un système d'exploitation est donc importante pour que les administrateurs réseaux puissent prédire le comportement de leurs systèmes.

Le RFC 3484 [4] décrit les algorithmes de sélection des adresses sources et destination. Ces algorithmes spécifient le comportement par défaut pour toutes les implémentations IPv6. Une application peut très bien adopter un autre choix, ces algorithmes ne proposant qu'une sélection par défaut.

L'algorithme de sélection de l'adresse de destination prend en entrée une liste d'adresses de destination, et produit en sortie une nouvelle liste triée d'adresses destination. Il classe ensemble les adresses IPv4 et IPv6. L'algorithme de sélection de l'adresse source retourne en sortie une seule adresse (source) pour une adresse destination donnée. Cet algorithme s'applique seulement aux adresses destination IPv6. Ces deux algorithmes s'appuient sur une table de « policy » qui opère selon le principe du préfixe le plus long, un peu à la manière d'une table de routage. La table de « policy » fournit, pour une adresse donnée, deux informations : une valeur de Précédence et une valeur de Label.

Grossièrement, la valeur de Précédence est utilisée pour trier les adresses destination (une valeur élevée sera préférée). La valeur de Label permet quant à elle de choisir une adresse source à utiliser pour une adresse de destination donnée. Chaque algorithme s'appuie sur une liste de règles assez complexes qui ne sont pas détaillées dans cet article. Pour une recherche dans la table de « policy », une adresse IPv4 sera représentée par une adresse IPv4 mappée. Le RFC 3484 propose la table de « policy » par défaut suivante :

```
Prefix          Precedence Label

::1/128         50         0  <- @ de loopback
::/0            40         1  <- @ par défaut
2002::/16       30         2  <- tunnel 6to4
::/96           20         3  <- @IPv4 compatible
::ffff:0:0/96   10         4  <- @IPv4 mappée
```

L'effet principal de cette table est de préférer d'abord les adresses (source et destination) natives IPv6.

Windows XP implémente la table de « policy » telle que proposée par le RFC 3484. FreeBSD implémente également la table de « policy », mais elle n'est pas active par défaut. Sur GNU/Linux, la table de « policy » ne semble pas implémentée.

9.3 Adresses Anonymes

Dans le mode de configuration « stateless », l'identifiant de l'interface est intimement lié à l'adresse Ethernet du poste client (et donc à l'utilisateur associé), quelque soit le préfixe IPv6 du plan d'adressage. Le RFC 3041[5] propose ainsi la création d'une adresse temporaire supplémentaire IPv6 susceptible de changer fréquemment et générée de manière pseudo aléatoire dans le but de préserver l'anonymat des utilisateurs.

Sur Windows XP, cette extension est active par défaut. De plus, cette adresse temporaire est sélectionnée de préférence à l'adresse IPv6 globale obtenue par auto configuration « stateless ». Sur FreeBSD et GNU/Linux, cette extension n'est pas active par défaut.

9.4 Recommandations DSI GU

Les postes clients Windows XP sont fortement présents dans l'environnement de la DSI GU. Les restrictions significatives liées à Windows XP (résolveur s'appuyant sur un « transport » IPv4 seulement, pas de client DHCPv6) nous ont conduit à retenir le mode de configuration « stateless » en double pile IPv4/IPv6. La nécessité de mettre à jour la base DNS limite l'utilisation de l'adresse temporaire que nous préconisons de désactiver. Pour l'instant manuelle, cette mise à jour est un peu fastidieuse. Pour pensons pouvoir automatiser partiellement cette tâche par l'intermédiaire de scripts, mais ce n'est encore qu'un projet. Les postes FreeBSD et GNU/Linux peuvent supporter l'auto configuration « stateless DHCP ». Cette solution a été validée entre un serveur « KAME » DHCPv6 et un client « Dibbler » Debian pour attribuer l'adresse IPv6 de notre serveur de nom. Cette configuration permet d'envisager des clients IPv6 seulement …

10 Perspectives

IPv6 prend peu à peu pied dans les réseaux Universitaires. Pour notre déploiement, nous avons créé un groupe de travail technique regroupant les différents acteurs de la plaque Grenobloise. Si des points sont encore à traiter (tels que les proxies ou le multicast), globalement tout est prêt pour se lancer dans un déploiement généralisé. La majorité des problèmes d'infrastructure est désormais correctement traitée, tout est opérationnel au même titre qu'IPv4.

Il reste cependant des points pour lesquels on manque de maturité (mobile IP), mais pour nous la question n'est plus de savoir s'il faut ou non déployer. Certains doutent de l'intérêt du déploiement dès à présent, il est de notre point de vue important de se préparer assez tôt afin d'éviter d'être mis devant le fait accompli : le vrai challenge maintenant est de réfléchir à la problématique de déploiement des postes de travail avec tout ce que cela comporte (problème de sécurité, de disponibilité d'auto configuration, …). Rappelons seulement que Microsoft prévoit d'intégrer une nouvelle pile IPv6 incluant la mobilité dans son prochain système d'exploitation client pour 2006…

Bibliographie

[1] S. Thomson and T. Narten, *IPv6 Stateless Address Autoconfiguration*, RFC 2462, December 1998.

[2] R. Droms, Ed. Bound, J.Volz, B., Lemon, T., Perkins, C. and M. Carney. *Dynamic Host Configuration Protocol for IPv6 (DHCPv6)*, RFC 3315, July 2003.

[3] R. Droms. *Stateless Dynamic Host Configuration Protocol (DHCP) Service for IPv6*. RFC 3736, 2004.

[4] R. Draves. *Default Address Selection for Internet Protocol version 6 (IPv6)*. RFC 3484, February 2003.

[5] T. Narten, R. Draves. *Privacy Extensions for Stateless Address Autoconfiguration in IPv6*. RFC 3041, 2001.