IPv6 en 2011, état, techniques et prévisions ou « Comment assurer une transition heureuse »

Stéphane Bortzmeyer
AFNIC
bortzmeyer@nic.fr

GUILDE, Grenoble, 12 avril 2011

Rappel : pourquoi IPv6

1. Quantité massive d'adresses permettant de :
   - 1.1 Attribuer une adresse à chacun (IPv4 est trop petit, même si l'allocation était parfaite)
   - 1.2 Permettre un changement de modèle, s'éloignant de l'ordinateur classique (Internet des objets : ma montre, mon téléphone et mes chaussures ont une adresse IP).
   - 1.3 Permettre des plans d'adressage rigolos (condensats cryptographiques de clés dans les adresses).
2. Et c'est à peu près tout (pas de changement au routage, c'est exprès ; pas d'innovations, IPsec par exemple n'est pas spécifique de v6).

État actuel : IPv4 est fini

Depuis des années, il y avait pénurie : essayez d'obtenir une adresse IPv4 pour chaque ordinateur chez vous.

Le 3 février 2011, l'IANA a épuisé sa réserve

À partir de juillet 2011, les RIR vont épuiser la leur et passer en mode « le dernier /8 » (adresses distribuées encore plus au compte-gouttes). Le premier sera APNIC `http://www.apnic.net/community/ipv4-exhaustion/graphical-information` : au 7 avril, il ne reste qu'un /8 et demi

Résultat, il va falloir déployer IPv6 alors qu'IPv4 est déjà terminé.

Horreurs en tout genre

Résultat de la pénurie, déploiement massif du NAT :

1. Sans que cela soit officiel, l'architecture de l'Internet a changé, il n'y a plus de bout en bout,
2. Déployer de nouveaux services très différents devient quasi-impossible,
3. Complication du code (STUN, ICE), on tunnele tout sur HTTP.

Les conséquences sont politiques, pas seulement techniques : pas de serveur chez soi, difficulté pour le pair-à-pair, difficulté pour la téléphonie. Cela ne fait pas que des malheureux ! (HADOPI, opérateurs voix traditionnels)

Le zoo des NAT

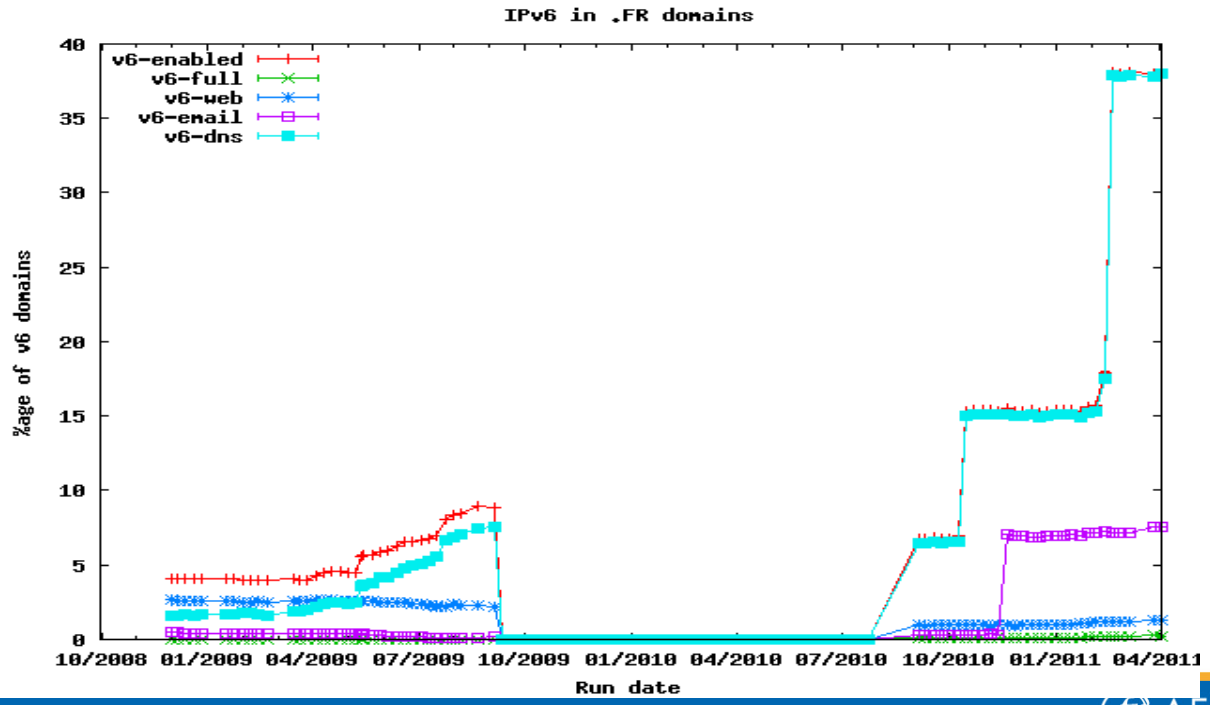
- NAT44 : l'actuel,
- NAT444 : ce qu'ont déjà asiatiques et africains : plus une seule adresse IP publique chez le client,
- NAT64 : une des techniques de co-existence, pour les premiers qui seront purement en IPv6,
- NAT66 : pour les organisations citées plus haut, qui trouvent que le NAT a des avantages,
- NAT464 et NAT646 pour bientôt ?

État actuel : IPv6 n'est pas allé loin

- Mises en œuvre : c'est quasiment parfait pour les gros logiciels libres très visibles. Pour les petits utilitaires, ou pour le logiciel privateur. . .
- FAI, opérateurs et hébergeurs : certains en France (pas tous), très peu aux USA, aucun en Afrique.
- Enseignement : très en retard. Dans le meilleur des cas, un cours de 2 heures ≪ Réseau avancé : IPv6 ≫ tout à la fin d'une année où tous les exemples étaient en IPv4.

Statistiques

Obtenues par DNSwitness sur .FR : près de 40 % des domaines ont au moins un serveur de noms en IPv6. Mais c'est beaucoup moins brillant pour le Web.



Autres statistiques de déploiement

Chez Google `http://www.nanog.org/meetings/nanog49/presentations/IPv6atGoogle.pdf` ≪ *Most "native" IPv6 users are in France* ≫

Prédiction

(Photo de Craig Wiggenhorn, récupérée sur Wikimedia Commons)

Migration : le plan originel

Il est faux de prétendre qu'il n'y avait pas de plan de migration

RFC 3750, 4057, 4213, 5211... Tous ces plans étaient fondés sur la **double pile**.
seulement-v4 ⟶ tout-le-monde-v4-certains-v6 ⟶ tout-le-monde-v4-et-v6 ⟶ certains-v4-tout-le-monde-v6 ⟶ seulement-v6

Ce schéma n'est plus réaliste aujourd'hui puisqu'il n'y a plus d'adresses v4.

On va se crasher mais on peut encore le faire à 50 km/h plutôt qu'à 200 km/h.

Les plans actuels de coexistence et de transition

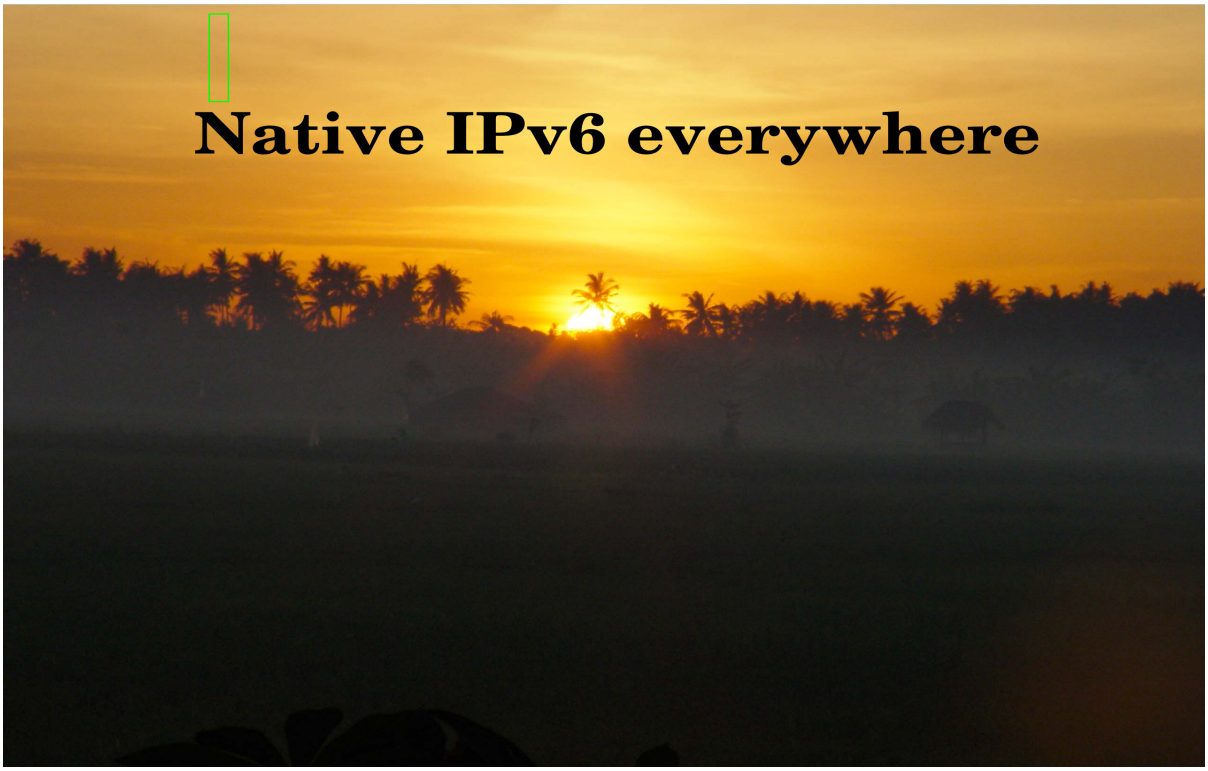
Le choix est vaste, tunnels 6in4, 6rd, NAT64, DS-lite...

Le plus important, avec tous ces plans...

Est de comprendre lequel s'applique à votre problème. Il ne faut pas les déployer aveuglément, chacun sert à quelque chose de précis ! On n'est surtout pas obligé de maîtriser **toutes** ces techniques.

Bonne page Wikipédia, très complète `http://en.wikipedia.org/wiki/IPv6_transition_mechanisms`

Le vrai objectif (à garder en tête)



(Photo anonyme, récupérée sur Wikimedia Commons)

Faire un choix

Pour choisir la technique adaptée à votre cas : faites l'inventaire de votre réseau et de ses partenaires. Qu'est-ce qui manque ? Qu'est-ce qui n'est pas IPv6 ?

1. Des adresses publiques IPv4 et/ou des applications qui gèrent IPv6 : DS-Lite
2. Mon réseau de collecte (DSLAM...) : 6rd
3. Mon FAI n'est pas v6 : tunnel
4. Moi, ça va, c'est le reste de l'Internet qui est encore partiellement v4 : NAT64

Ceux dont on ne parle pas

Certaines techniques sont vraiment trop horribles

Et ne sont donc pas mentionnées ici (6to4, Teredo...). D'autres ont été officiellement abandonnées (NAT-PT...).

Les tunnels

Le but : connecter une île IPv6 à l'Internet IPv6, au dessus d'un Internet IPv4. Très utile aujourd'hui, bien des FAI ne fournissant pas IPv6.

Le principe : on encapsule des paquets IPv6 dans de l'IPv4. Il y a plusieurs méthodes, avec négociation automatique via le protocole TSP (RFC 5572) ou par configuration manuelle.

Les fournisseurs : plusieurs fournisseurs gratuits comme SixXS, Hexago/Freenet. Je recommande Hurricane Electric.

Les inconvénients : tous les tunnels diminuent la MTU du chemin. En théorie, c'est sans conséquence mais en pratique...

Exemple de tunnel manuel sur une Debian

```
auto 6in4
iface 6in4 inet6 v4tunnel
   # Le POP d'Hurricane Electric au Panap
   endpoint 216.66.84.42
   address 2001:470:1f12:420::2
   netmask 64
   # "gateway" n'a pas l'air de marcher ?
   up route -A inet6 add ::/0 dev 6in4
```

Les paquets IPv6 seront encapsulés en v4, délivrés à 216.66.84.42, qui les décapsulera.

6rd

Le but : pour un FAI, fournir une connectivité IPv6 à ses clients alors que le réseau du FAI a des composants purement v4 qu'on ne peut pas bouger (par exemple les DSLAM)

Le principe : Encapsulation des paquets v6 par le CPE (la *box*) et décapsulation juste avant la sortie du réseau du FAI

Les logiciels : c'est désormais dans le noyau Linux. Cisco a aussi une mise en œuvre.

Les normes : RFC 5969 (remplace le 5569)

Les inconvénients : tunnel, donc problèmes de MTU

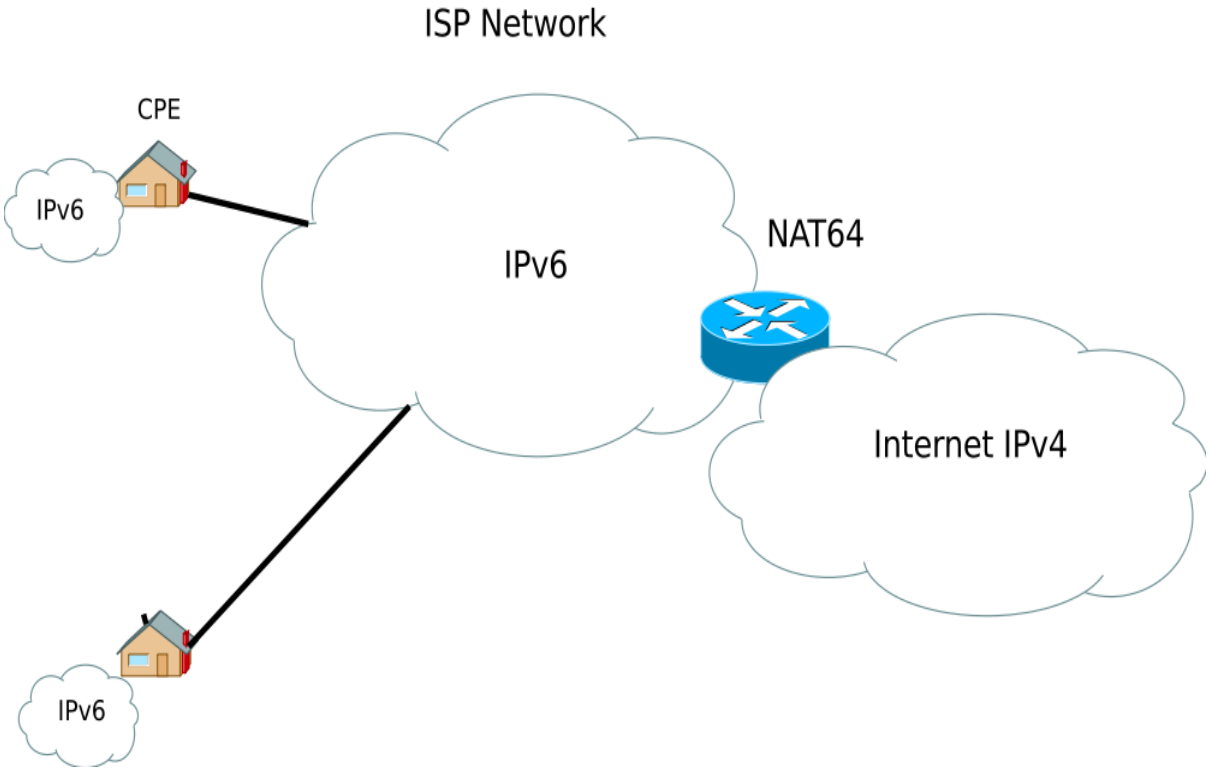
NAT64

Le but : connecter à l'Internet IPv4 des machines entièrement IPv6. Compte-tenu de l'épuisement d'IPv4, de telles machines vont bientôt exister.

Le principe : traduire dynamiquement les paquets IPv6 en IPv4 et réciproquement. Un peu comme le NAT44 mais on ne change pas que l'adresse.

```
% telnet twitter.com 80
Trying 64:ff9b::80f2:f0f4...
Connected to twitter.com.
...
```

NAT64 en image



(Dessin de Mro, récupéré sur Wikimedia Commons)

DNS64

Les détails : comme la machine IPv6 ne demande que des adresses v6 (enregistrement AAAA), le DNS doit mentir : DNS64 fabrique des adresses v6 à partir des v4.

BIND $\geq$ 9.8 a cette possibilité

```
options {
        ...
        dns64 64:ff9b::/96 { // The well-known prefix
               mapped { !rfc1918; any; };
                      // Never synthetize AAAA records
                      // for private addresses
         };
```

NAT64, suite

Les logiciels : deux NAT64 en logiciel libre, Ecdysis et Tayga. Bientôt dans les routeurs et *boxes* ?

Les normes : RFC pas encore sortis mais presque.

Les inconvénients : peut poser des problèmes avec DNSSEC. Change la MTU. Complication accrue.

Exemple avec Ecdysis

C'est un module noyau Linux pour mettre sur le routeur (qui doit avoir une adresse v4).

```
% sudo ./nat64-config.sh
...
% netstat -r -n  -Ainet6
Kernel IPv6 routing table
Destination       Next Hop   Flag Met Ref Use If
64:ff9b::/96      ::         U    1024 0     0 nat64
...
```

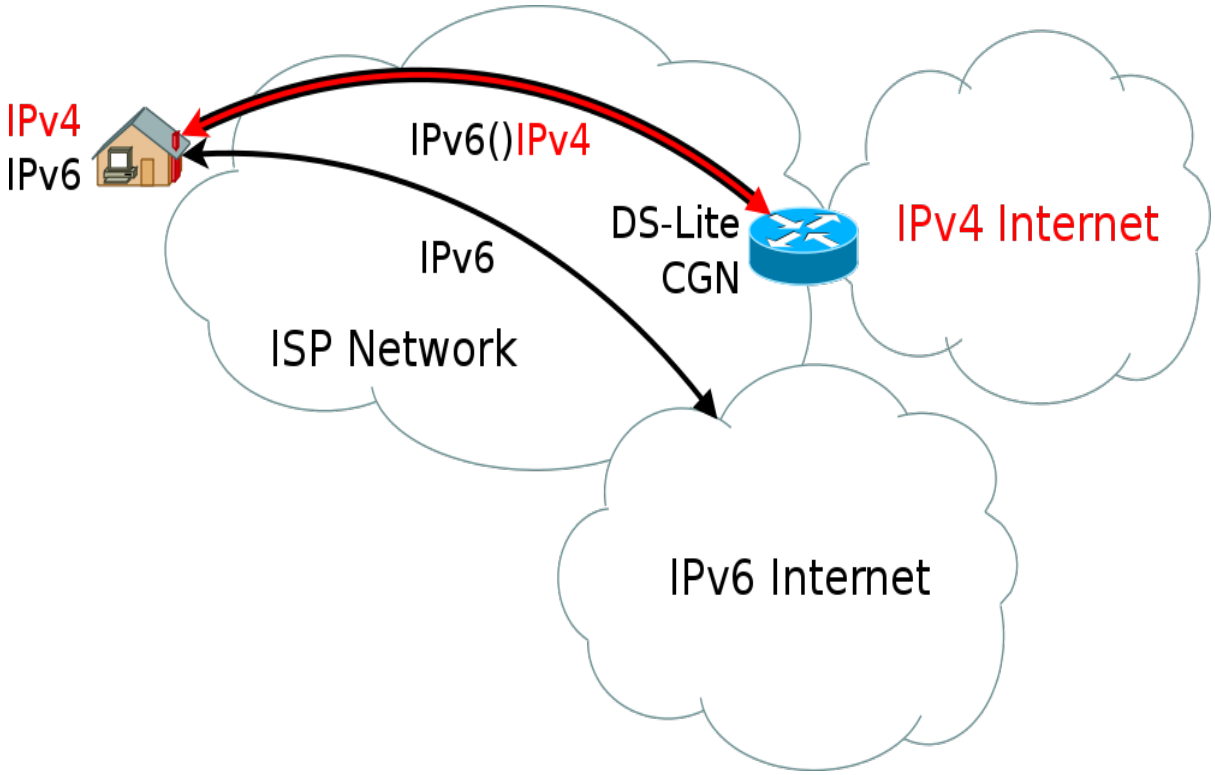
DS-Lite

Le but : connecter à l'Internet IPv4 et IPv6 des machines IPv4 au dessus d'un FAI v6, et sans avoir d'adresses IPv4.

C'est donc le contraire de 6rd : DS-lite cible les nouveaux entrants, dont le réseau sera v6 dès le début et qui n'auront jamais d'adresses v4 publiques.

Le principe : le réseau local du client a des adresses privées v4. La *box* encapsule les paquets v4 au dessus du réseau v6 jusqu'à un NAT géant (CGN pour *Carrier-Grade NAT*)

DS-Lite en image



(Dessin de Mro, récupéré sur Wikimedia Commons)

DS-Lite, suite

Les logiciels : un AFTR (un des composants, le CGN) en logiciel libre à l'ISC. Bientôt dans les routeurs et *boxes* ?

Les normes : RFC pas encore sortis.

Les inconvénients : très complexe. Le CGN sera t-il suffisant ? Problèmes liés au partage d'adresses : HADOPI ne sera pas contente.

Rendre heureux les globes oculaires

Et pour ceux qui ont suivi les recommandations et qui sont en double-pile ? Que se passe t-il s'ils se connectent à un serveur double-pile ?

1. M. Michu a un Windows de moins de dix ans, et un navigateur Web correct, Firefox, il a IPv6,
2. Sur le réseau local de M. Michu, son petit neveu a activé IPv6, la machine Windows se configure toute seule,
3. Google active IPv6 pour un service critique vital : YouTube,
4. Mais le FAI de M. Michu ne route pas IPv6 ou bien le fait mal. Firefox prend des plombes avant de timeouter.
   M. Michu, furieux, dit du mal d'IPv6 sur Twitter, où on lui explique comment couper IPv6.
5. Il sera difficile de convaincre M. Michu de réessayer...

Les solutions pour les globes oculaires

Le *whitelisting* DNS, à la Google : n'envoyer de AAAA qu'aux réseaux placés sur une liste blanche (seuls les bons sont mis sur la liste).

Autres solutions ? Changer les applications pour faire des tentatives de connexion en parallèle
`http://www.isc.org/community/blog/201101/how-to-connect-to-a-multi-homed-server-over-tcp`

Conclusion

À vous de l'écrire

L'avenir d'IPv6 et de l'Internet dépend de vous