RFC 6883 : IPv6 Guidance for Internet Content Providers and Application Service Providers

Stéphane Bortzmeyer
<stephane+blog@bortzmeyer.org>

Première rédaction de cet article le 8 mars 2013

Date de publication du RFC : Mars 2013

https://www.bortzmeyer.org/6883.html

Ce nouveau RFC sur IPv6 ne spécifie pas un protocole réseau mais rassemble simplement un ensemble de conseils pour les gens qui gèrent des serveurs Internet et qui veulent les rendre accessibles en IPv6. Bien sûr, en 2013, tous les serveurs devraient depuis longtemps être ainsi accessibles. Mais le déploiement d'IPv6 a pris du retard et ce RFC est une des nombreuses tentatives pour tenir la main des retardataires qui hésitent encore à traverser le gué.

Pourquoi faut-il une stratégie de déploiement IPv6 ? Pourquoi ne pas tout simplement se dire « Au diable les mines ! En avant toute ! » ? Parce que l'administrateur typique d'un serveur Internet pense à ses utilisateurs et il veut que l'introduction d'IPv6 ne rende pas les choses moins bonnes qu'en IPv4. Certaines techniques de transition et/ou de coexistence entre IPv4 et IPv6 ayant eu des effets négatifs, il est prudent de réfléchir et de ne pas se lancer aveuglément.

Comme indiqué plus haut, ce RFC ne concerne pas le pur utilisateur qui veut se connecter en IPv6, mais le gérant de serveurs, qui est un professionnel de l'hébergement, ou simplement une organisation qui gère ses propres serveurs (serveurs Web ou autres). Il n'est pas écrit pour l'expert IPv6 (qui n'apprendra rien de nouveau) mais pour l'administrateur système/réseaux de l'hébergeur de serveurs.

La stratégie recommandée par le RFC est celle dite de la « **double pile** » (RFC 6180 [1]), qui consiste à avoir à la fois IPv4 et IPv6 sur les serveurs. Comme on ne va pas déployer IPv6 instantanément partout, elle se subdivise en deux approches : de l'extérieur vers l'intérieur (on rend les serveurs externes accessibles en IPv6, peut-être via un relais comme décrit en section 7, puis on migre progressivement l'infrastructure interne) ou bien de l'intérieur vers l'extérieur (on passe l'infrastructure, les serveurs de

1. Pour voir le RFC de numéro NNN, https://www.ietf.org/rfc/rfcNNN.txt, par exemple https://www.ietf.org/rfc/rfc6180.txt

base de données, les serveurs LDAP, etc, en IPv6, avant de petit à petit ouvrir IPv6 à l'extérieur). La première approche, de l'extérieur vers l'intérieur, a l'avantage qu'un bénéfice visible pour les utilisateurs est immédiat. Elle marche même si certains services internes sont coincés en IPv4 (par exemple une vieille application tournant sur un vieux système, indispensable à votre activité mais qu'on ne peut pas mettre à jour). Certaines organisations se vantent donc d'être accessibles en IPv6 alors qu'à l'intérieur, le déploiement du nouveau protocole est loin d'être achevé.

La deuxième approche, de l'intérieur vers l'extérieur, est surtout intéressante pour les organisations de petite taille (qui contrôlent tous les systèmes internes) ou récentes (qui n'ont plus d'AS/400 en production, et encore puisque même celui-ci a IPv6 <http://publib.boulder.ibm.com/infocenter/iseries/v5r3/topic/rzai2/rzai2.pdf>). Avec cette approche, l'activation d'IPv6 sur les serveurs externes sera simple, tout sera en place avant.

Le fournisseur de services doit aussi se rappeler qu'il ne contrôle pas les clients. Certains resteront en IPv4 pur pendant longtemps, d'autres seront double-pile et on verra sans doute apparaître bientôt des clients purement IPv6. Certains clients (les mobiles, par exemple) passeront même d'IPv4 à IPv6 au cours d'une seule visite sur le site Web.

Bref, le déploiement d'IPv6 nécessitera quelques études préalables, en vérifiant les capacités du logiciel déployé (cf. RFC 8504).

Maintenant, en route vers le déploiement. Première étape suggérée par le RFC, la formation (section 3). Aujourd'hui, rares sont les formations initiales qui mentionnent IPv6 (en France, la situation dans les IUT semble particulièrement mauvaise). C'est donc souvent dans l'entreprise qu'il faudra former les employés aux différences entre IPv4 et IPv6. Il faut surtout éviter le syndrome de l'expert IPv6 isolé. En 2013, tout professionnel IP doit connaître IPv6 aussi bien qu'IPv4. C'est donc la totalité du personnel informaticien qui va devoir progresser. Le RFC dit « "*everybody who knows about IPv4 needs to know about IPv6, from network architect to help desk responder*" ».

Le but est d'éviter des résultats ridicules comme ce réseau où, lors du déploiement d'IPv6, les adresses comportant les chiffres hexadécimaux entre A et F étaient soigneusement évitées, de peur de créer de la confusion chez les techniciens ! Évidemment, si on a des techniciens assez arriérés pour ne pas savoir gérer de l'hexadécimal, on a un problème... Autre anecdote embarrassante, cet employé du "*help desk*" qui parlait à un client de « "*one Pv6*" » car il avait cru voir un 1 dans IPv6... L'un des buts de la formation est d'éviter ce genre de bavure.

L'approche du RFC est d'éviter de traiter IPv6 à part, par exemple d'avoir une procédure de remontée des problèmes différente pour IPv6, menant vers l'unique gourou IPv6. Plutôt qu'un « service IPv6 » spécialisé, le RFC recommande qu'IPv6 soit simplement traité comme IPv4.

Dernier point délicat sur la formation, le meilleur moment où la faire. Trop tôt, et elle sera oubliée lorsque le déploiement effectif commencera. Il faut donc essayer de la synchroniser avec le déploiement.

Ensuite, il faut obtenir une connectivité IPv6 (section 4). Elle peut être **native** (directement sur la couche 2, comme en IPv4) ou via un tunnel géré. Historiquement, bien des expérimentations IPv6 ont été faites en se connectant par un tunnel non géré et les résultats (en performances et en fiabilité) ont découragé beaucoup de gens de continuer avec IPv6. Le RFC recommande **très** fortement de ne pas perdre de temps avec les tunnels et de faire du natif. Toutefois, si on se connecte via un tunnel, cela doit être un tunnel géré (le RFC ne donne pas de noms mais c'est, par exemple, un service qu'Hurricane Electric <https://www.bortzmeyer.org/ipv6-he.html> offre très bien). Les solutions avec des tunnels automatiques, non gérés, comme 6to4 (que le RFC ne cite pas) sont à fuir.

Un tunnel géré a quand même des défauts : les performances seront inférieures, et la MTU réduite va mener à des problèmes pénibles à déboguer <https://www.bortzmeyer.org/mtu-et-mss-sont-dans-un-resea html>. Bref, le fournisseur de services ne devrait pas utiliser autre chose qu'une connexion native avant d'offrir un accès de production à ses services.

Notez que cela ne concerne que le fournisseur : on ne contrôle pas les clients et il faut donc se préparer à ce que certains utilisent des bricolages non fiables comme Teredo (cf. section 9).

Étape suivante (dans le RFC ; dans le déploiement, on peut paralléliser ces tâches), l'infrastructure IPv6 (section 5). D'abord, l'attribution des adresses. Comme en IPv4, on peut utiliser des adresses PI ou PA, selon ses besoins et selon la politique du RIR local. Si on utilise du PA, il faut se préparer à de futures renumérotations et, pour cela, bien relire le RFC 4192. Naturellement, si le fournisseur de service a hébergé ses services dans le "*cloud*", il n'a pas le choix, il devra suivre la politique de l'hébergeur.

Pour gérer les adresses IPv6 locales, on peut utiliser un logiciel spécialisé, un IPAM. Pour distribuer les adresses aux machines, on peut le faire manuellement ou via un programme qui automatise partiellement la gestion du parc, comme Puppet ou Chef. Mais on peut aussi les distribuer par DHCP (RFC 8415). Le RFC suggère DHCP, car il facilitera une éventuelle renumérotation.

Plus exotique est la possibilité de ne pas avoir d'adresses statiques pour les serveurs mais de compter uniquement sur l'auto-configuration sans état d'IPv6 (SLAAC, RFC 4862), combinée avec des mises à jour dynamiques du DNS (RFC 3007). Dans ce cas, ce sont les noms des serveurs qui sont statiques, pas leurs adresses IP. Le RFC note bien que c'est une possibilité théorique mais que personne n'a encore essayé cela pour un grand réseau de production.

Pour la topologie du réseau (physique, via les câbles, ou logique, via les VLAN), le RFC recommande d'avoir la même topologie en IPv4 et IPv6. Ce n'est pas obligatoire mais cela facilitera grandement le débogage lorsqu'il y aura un problème.

Une fois les adresses attribuées, le routage : tous les protocoles de routage ont la capacité de router IPv6, même RIP (RFC 2080). La méthode la plus courante aujourd'hui est de déployer deux routages parallèles et indépendants, un pour IPv4 et un pour IPv6. Toutefois, deux protocoles de routage, IS-IS (RFC 5308) et OSPF (RFC 5838), ont la capacité de gérer les deux versions de manière intégrée, ce qui peut simplifier la gestion du réseau. (Pour OSPF, c'est encore très peu déployé, la plupart des sites OSPF font tourner deux routages parallèles.) Comme les concepts de routage sont les mêmes en IPv6 et en IPv4, il ne devrait pas y avoir trop de problèmes pour l'équipe qui s'en occupe.

Parmi les points qui risquent d'être différents en IPv6 figurera le cas où un site a plusieurs préfixes PA (un pour chaque fournisseur de connectivité). Il faudra alors soigner le routage pour éviter de faire sortir via le fournisseur A un paquet dont l'adresse source est prise dans le préfixe du fournisseur B : si A met en œuvre les filtrages recommandés par les RFC 2827 et RFC 3704, le paquet sera jeté.

C'est seulement lorsque tout est déployé et testé, au niveau adressage et routage, qu'on pourra envisager d'annoncer les adresses IPv6 des serveurs dans le DNS. Ne faites pas l'erreur d'un gros fournisseur de raccourcisseur d'URL qui avait fait cette annonce alors que son serveur ne marchait pas en IPv6 ! Ou comme l'avait fait Yahoo qui avait mis le AAAA <https://lists.afrinic.net/pipermail/afripv6-discuss/2013/001302.html> sans configurer Apache.Testez donc d'abord (par exemple avec un nom spécifique comme ipv6.example.com, comme expliqué en section 9). Ensuite, mettez les enregistrements AAAA dans le DNS. Dès qu'ils seront publiés, le trafic arrivera.

Un équipement souvent placé devant les serveurs est le répartiteur de charge (section 6). Comme tous les intermédiaires, ils sont une source importante de problèmes. Après un long retard, la plupart gèrent aujourd'hui IPv6, mais il est important d'insister auprès des fournisseurs de tels équipements pour qu'IPv6 ait le même niveau de performance qu'IPv4.

Le logiciel sur les serveurs, maintenant (section 8). Dans les systèmes d'exploitation, il n'y a guère de problème, tout fonctionne en IPv6 depuis longtemps. C'est le cas des noyaux mais aussi des « grands » logiciels serveurs comme Apache, NSD ou Postfix. Mais les applications spécifiques, c'est une autre histoire, certaines ne sont pas forcément prêtes. Pour les applications client, le fait d'utiliser des noms de domaines au lieu des adresses IP aide beaucoup. Pour les serveurs, il peut y avoir des surprises, par exemple au moment de journaliser la requête, lorqu'on voudra enregistrer l'adresse IP du client. Des tests soigneux seront nécessaires.

Beaucoup de fournisseurs de service géo-localisent leurs utilisateurs. Dans ce cas, les bases de géolocalisation typiques comme GeoIP n'étant pas forcément prêtes pour IPv6, il y aura des problèmes, à moins de passer à une méthode de géolocalisation plus fiable que celle fondée sur l'adresse IP, par exemple avec la technique HELD du RFC 5985.

Lorsque le client change d'adresse IP (ce qui arrive avec les mobiles), il peut même changer de version d'IP. Si les "*cookies*" distribués sont liés à une adresse IP, il faudra en tenir compte.

On l'a vu plus haut, le fournisseur de service ne contrôle pas ses clients. Il peut tout à fait arriver que ceux-ci ont choisi des techniques de transition/coexistence entre IPv4 et IPv6 et que celles-ci soient peu fiables (par exemple Teredo). Avec l'approche recommandée dans ce RFC, la double pile, il n'y aura normalement pas de requêtes venant de NAT64 (RFC 6146). Mais on peut imaginer d'autres problèmes, y compris des doubles traductions, de v6 en v4 puis encore en v6, et que le serveur ne peut pas détecter.

Reste le cas des tunnels automatiques comme le 6to4 du RFC 3068. Le RFC 6343 donne plein de bons conseils pour limiter les dégâts. Les deux points importants, du côté du serveur, sont :

— S'assurer que la détection de la MTU du chemin fonctionne (ce qui veut dire, en pratique, ne pas bloquer bêtement tout ICMP),
— S'assurer que le serveur gère bien la négociation de la MSS de TCP (RFC 2923).

Mais l'hébergeur a un pouvoir limité : il peut vérifier que les paquets ICMP passent bien sur son réseau mais il ne peut rien garantir quant aux autres réseaux traversés. C'est pour cela que 6to4 est désormais abandonné (RFC 7526).

Certains fournisseurs de service ont donc choisi une approche plus violente en configurant leurs serveurs DNS pour ne **pas** envoyer les enregistrements AAAA au résolveur si celui-ci est dans un réseau dont on sait qu'il a des problèmes, notamment de PMTUd. Encore plus pessimiste, l'approche inverse où on n'envoie les AAAA qu'aux réseaux dont on sait qu'ils ont une connectivité IPv6 native correcte (Google avait utilisé ce mécanisme au début). Cette solution, décrite dans le RFC 6589, est coûteuse en temps (il faut gérer la liste noire ou la liste blanche) et pas tenable sur le long terme.

Malheureusement, les pouvoirs du fournisseur de service s'arrêtent là. Il ne contrôle pas la chaîne complète jusqu'au client et ne peut donc pas garantir à celui-ci un vécu parfait. La voie la plus prometteuse semble être le déploiement progressif du mécanisme "*happy eyeballs*" (globes oculaires heureux) du RFC 6555.

Et si le fournisseur de service utilise des CDN (section 10) ? Cela peut marcher, avec IPv6 comme avec IPv4 si le fournisseur du CDN suit, comme celui du serveur principal, les recommandations de ce RFC. À noter que, le CDN étant géré par une organisation différente de celle qui gère le serveur original, on

aura parfois, si le CDN est plus avancé, un contenu partiellement accessible en IPv6 alors que le serveur original n'a pas IPv6. Sinon, parmi les pièges spécifiques aux CDN, la synchronisation. Le contenu à servir est distribué à tous les nœuds du CDN via l'Internet et ce processus n'est pas instantané. Comme le contenu n'est pas forcément servi en IPv4 et en IPv6 par les mêmes machines, un visiteur peut avoir, pendant les périodes où la synchronisation est en cours, un contenu différent selon qu'il y accède en v4 ou en v6.

La section 12 rassemble ensuite une liste hétérogène de problèmes potentiels supplémentaires. Par exemple, comme vu plus haut dans le cas du CDN, un service moderne comprend typiquement plusieurs éléments, parfois hébergés à des endroits différents (pensez à une page Web contenant du JavaScript et du Flash stockés sur les serveurs de Google). Dans ce cas, le service peut être partiellement accessible en IPv6 sans même que son gérant ne s'en rende compte. Si le visiteur a des problèmes de connectivité IPv6, les équipes du fournisseur de service risquent de ne pas comprendre tout de suite ce qui se passe en cas d'« IPv6 involontaire ».

C'est donc l'occasion de rappeler (section 13) que le déploiement d'IPv6 n'est pas juste un coup sans lendemain, qu'on pourrait sous-traiter, en « mode projet » : au contraire, il faut assurer la maintenance et le fonctionnement opérationnel dans le futur. Cela n'a rien d'extraordinaire et ne nécessite pas de compétences surhumaines. Il existe des réseaux et des sites en double pile depuis de nombreuses années. Mais il faut le faire.

Un exemple parmi d'autres : il est important d'assurer la supervision aussi bien en IPv6 qu'en IPv4. Cela concerne bien sûr les réseaux (pinguer les adresses IPv6 comme les adresses IPv4) mais aussi les services (si un service a été accidentellement lancé sans écoute en IPv6, il faut que cela soit détecté). À noter que, avec la plupart (peut-être tous) les logiciels de supervision existants, tester des machines ou des services qui écoutent sur plusieurs adresses IP (que celles-ci soient une adresse v4 et une adresse v6 ou bien plusieurs adresses v4) est souvent un peu pénible et pas vraiment prévu. Par exemple, avec Icinga <https://www.bortzmeyer.org/icinga.html>, il faut mettre explicitement deux tests, un en IPv4 et un en IPv6, le programme de test ne testant qu'une seule des adresses du service. Si on ne fait pas attention, on risque donc de se rassurer en voyant l'écran vert du logiciel de supervision, sans avoir perçu que le service ne marchait plus en IPv6 (ou en v4, selon le cas).

Et la sécurité (section 14) ? Elle est évidemment aussi importante en IPv6 qu'en IPv4 mais on manque encore d'expérience. En pratique, début 2013, très peu d'attaques sont lancées en IPv6 (elles bénéficient par contre d'une exposition médiatique démesurée, justement parce qu'elles sont rares). Cela ne durera pas éternellement et les responsables réseaux et systèmes doivent donc se préoccuper de ces risques. En gros, comme IPv6 est uniquement une nouvelle version d'IPv4, fondée sur les mêmes concepts, la quasi-totalité des risques présents en IPv4 (de l'attaque par déni de service aux tentatives de se connecter au serveur SSH) existent en IPv6.

Aujourd'hui, il est difficile de répondre simplement à une question comme « est-ce plus dangereux en IPv6 ? » Comme on l'a vu, il y a beaucoup moins d'attaques. Mais les défenses sont également moins bonnes : il est fréquent qu'une organisation soit protégé par un pare-feu en IPv4 mais que tout le trafic IPv6 passe librement. En fait, aujourd'hui, aussi bien les attaquants que les défenseurs ne prennent guère en compte IPv6. On a donc un équilibre... instable. Sur ce sujet de la sécurité d'IPv6, on peut consulter mon exposé à l'ESGI <https://www.bortzmeyer.org/ipv6-securite.html>.