RFC 6333 : Dual-Stack Lite Broadband Deployments Following IPv4 Exhaustion

Stéphane Bortzmeyer
<stephane+blog@bortzmeyer.org>

Première rédaction de cet article le 14 août 2011

Date de publication du RFC : Août 2011

http://www.bortzmeyer.org/6333.html

Il existe une myriade de techniques de coexistence entre l'ancien protocole IPv4 et le nouvel IPv6, à tel point qu'on peut même avoir du mal à faire son choix (voir mon exposé à ce sujet <http://www.bortzmeyer.org/transition-ipv6-guilde.html>). Et le groupe de travail Soft Wires <http://tools.ietf.org/wg/softwire> (réseaux virtuels variés) de l'IETF en invente de temps en temps des nouvelles. Pour ne pas s'affoler devant cette multitude, il faut se rappeler que chacune de ces techniques a un but bien précis et sert dans un cas donné. DS-Lite ("*Dual-Stack Lite*"), objet de ce RFC, vise les FAI récents, qui n'ont jamais eu d'adresses IPv4 publiques en quantité et dont le réseau interne est en IPv6 depuis le début.

Donc, DS-Lite est à l'opposé de, par exemple, 6rd (RFC 5969 [1]), qui vise les FAI existants qui n'ont pas le courage de mettre à jour leur réseau IPv4. DS-Lite vise un autre problème : si, en 2011, je crée un nouveau FAI en Asie (APNIC a été le premier RIR dont le stock d'adresses IP est tombé à zéro), je n'obtiendrai dans le meilleur des cas qu'une poignée d'adresses IPv4 publiques. Mais le reste de l'Internet (et même le réseau local de mes clients, et leurs applications) est majoritairement en IPv4. Mon beau réseau tout neuf, qui pourra être IPv6 depuis le début puisqu'il n'aura pas à porter le poids de l'héritage, ne me servira donc à rien (sauf à voir ce blog, qui est accessible en IPv6). Ce cas n'était pas prévu à l'origine ; il y a eu largement assez de temps pour faire une transition plus simple de IPv4 vers IPv6 mais beaucoup d'acteurs ont traîné les pieds <http://www.bortzmeyer.org/ipv6-et-l-echec-du-marche.html>, nous amenant à la situation actuelle, où il faut déployer IPv6 sans pouvoir compter sur des adresses v4 publiques en quantité suffisante. DS-Lite arrive alors à la rescousse.

Le principe de DS-Lite est donc de connecter à l'Internet IPv4 (et bien sûr aussi IPv6) des machines IPv4 (les machines des clients) au dessus d'un FAI v6, et sans avoir beaucoup d'adresses IPv4 publiques.

1. Pour voir le RFC de numéro NNN, https://www.ietf.org/rfc/rfcNNN.txt, par exemple https://www.ietf.org/rfc/rfc5969.txt

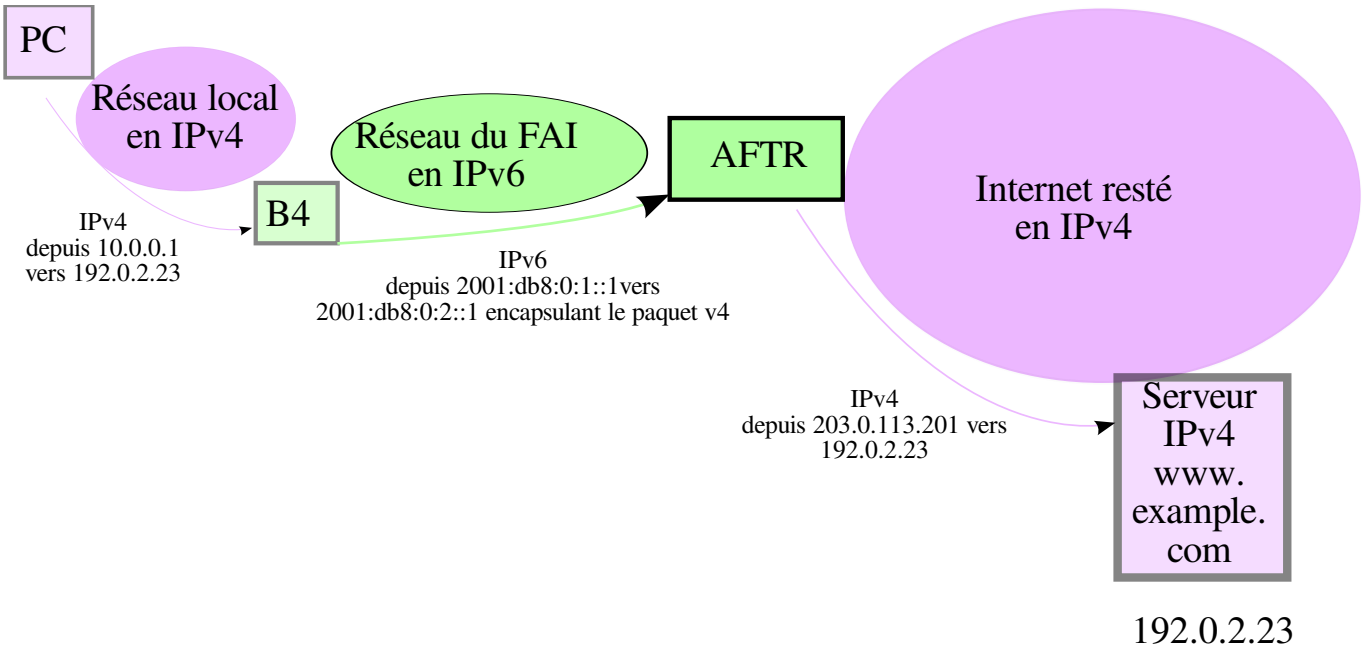
(Les machines purement IPv6 ne sont donc pas concernées, elles ont une connexion native normale, seules les machines/applications qui utilisent encore IPv4 doivent passer par ce bricolage.) Le principe : le réseau local du client a des adresses privées v4. La "*box*" encapsule les paquets v4 au dessus du réseau v6 (tunnel IPv4-dans-IPv6) jusqu'à un NAT géant (CGN) qui traduit ces adresses en adresses publiques (il faut donc avoir au moins quelques adresses IPv4 publiques). DS-Lite réutilise donc deux techniques éprouvées, le tunnel et le NAT.

Pour suivre la description détaillée, un peu de vocabulaire (section 3). DS-Lite nécessite deux composants :

— Le **B4** ("*Basic Bridging Broadband element*", le B4 étant un jeu de mots car il se prononce comme "*before*", indiquant que ce composant est **avant** le tunnel), qui est dans le réseau de M. Toutlemonde, et tunnele les paquets IPv4 de ce M. Toutlemonde vers :
— L'**AFTR** ("*Address Family Transition Router*", ce qui se prononce comme "*after*", indiquant que cet élément est **après** le tunnel), le CGN (routeur NAT géant) qui traduit des adresses IPv4 privées de M. Toutlemonde vers la poignée d'adresses IPv4 publiques qu'a pu obtenir le FAI.

Entre le B4 et l'AFTR, il n'y a qu'IPv6 : tous les paquets IPv4 doivent être tunnelés. Enfin, pour suivre ce RFC, il peut être utile de réviser le vocabulaire de la double-pile (RFC 4213) et du NAT (RFC 4787).

Voici un schéma d'une communication avec le serveur Web d'adresse `192.0.2.23`. L'adresse IPv4 publique utilisée par l'AFTR est `203.0.113.201`. Le réseau IPv6 du FAI utilise `2001:db8::32`. Le réseau local reçoit des adresses en `10.0.0.0/24` :



La section 4 décrit plus en détail les scénarios envisageables pour DS-Lite. Rappelez-vous d'un des points les plus importants des techniques de transition/coexistence v4/v6 : on n'est pas obligés de les déployer toutes, bien au contraire. Chacune s'applique à des cas spécifiques. Les avantages essentiels de DS-Lite, vus par le RFC :

— Une machine v4 ne communique qu'avec des machines v4 et une v6 qu'avec des v6. La traduction d'adresses se fait uniquement à l'intérieur d'une même famille (contrairement au NAT64 du RFC 7915).
— DS-Lite découple le déploiement d'IPv6 dans chacun des trois réseaux : celui du client final, celui du FAI et celui de l'Internet global. Plus besoin d'attendre que quelqu'un d'autre commence ou s'y mette.

— Le FAI n'a pas les limitations d'IPv4 dans son réseau interne (un vrai problème pour certains gros FAI, qui n'ont même pas assez d'adresses v4 pour leur propre réseau interne).
— Le tunnel entre le B4 et l'AFTR peut être situé où le FAI le désire. Le mode le plus courant sera sans doute avec le B4 dans le CPE et l'AFTR en sortie du réseau du FAI mais ce n'est nullement obligatoire. Cette souplesse permet à la solution de grossir tranquillement. Par exemple, si seuls les clients d'une certaine région utilisent DS-Lite, on met l'AFTR dans cette région puis, si on étend la solution à tout le monde, on reporte simplement le ou les AFTR plus loin et plus près du centre du réseau (cf. annexe A).
— Comme, du point de vue IPv4, les B4 sont directement connectés à l'AFTR, on peut envisager des techniques de contrôle du NAT par l'utilisateur (par exemple, réserver un port donné), peut-être avec le protocole PCP ("*Port Control Protocol*", RFC 6887).

J'ai indiqué qu'un déploiement typique mettrait le B4 dans le CPE, dans la "*box*". Ce n'est pas la seule possibilité. Mais ce sera sans doute la plus fréquente (l'annexe B décrit les différentes architectures, avec le détail des adresses). La section 4.2 décrit les caractéristiques d'un CPE ayant une fonction DS-Lite : il inclut un serveur DHCP pour distribuer des adresses IPv4 privées (RFC 1918) sur le réseau local, il ne fait **pas** de NAT lui-même (contrairement aux CPE d'aujourd'hui), il n'a pas d'adresse IPv4 publique, et, en IPv6, il est simplement un routeur ordinaire, sans traduction, ni particularités (connexion IPv6 native pour les clients qui en sont capables).

S'il n'y a pas de CPE (cas typique d'un "*smartphone*" connecté à un réseau 3G), la section 4.3 prend le relais : dans ce cas, la machine connectée doit être son propre B4.

La section 5 décrit en détail la fonction B4. Elle comprend la tunnelisation des paquets IPv4 vers l'AFTR. Au fait, comment le B4 connait-il son AFTR (section 5.4) ? Il doit être configuré avec l'adresse IPv6 de celui-ci, ou bien la récupérer via DHCP avec l'option du RFC 6334.

Pour le service DNS, le B4 doit connaître les adresses IPv6 des serveurs récursifs du FAI (rappelez-vous que la machine qui fait le B4 n'a typiquement pas d'adresse IPv4 publique), par exemple via DHCPv6. Les machines du réseau local, n'ayant pas forcément IPv6, il faut leur fournir un serveur récursif v4. Le RFC recommande que le B4 joue ce role et s'annonce lui-même comme récurseur (et suive alors les recommandations du RFC 5625).

Enfin, le B4 a besoin d'une adresse IPv4 à lui, pour les paquets dont il est l'origine. La plage `192.0.0.0/29` a été réservée pour cela (cf. section 10), le `192.0.0.2` étant pour le B4. Ce préfixe a d'ailleurs été élargi ultérieurement à d'autres systèmes que DS-Lite dans le RFC 7335.

La section 6 fait la même chose (décrire tous les détails) pour la fonction AFTR, qui est à la fois la terminaison du tunnel IPv4-and-IPv6 et le CGN ("*Carrier-Grade NAT*"). Par exemple, l'AFTR n'a pas de fonction DNS à assurer (le B4 fait la résolution sur IPv6). Il a lui aussi une adresse bien connue, `192.0.0.1`, et c'est celle qu'on verra sans doute souvent lors des traceroute.

Pour que tout cela marche bien, il y a quelques détails techniques à régler. La section 7 couvre ceux qui concernent le réseau : notamment, le tunnel doit respecter les règles des RFC 2473 et RFC 4213. Et la section 8 couvre les détails liés au NAT : possibilité pour un AFTR d'avoir plusieurs plages d'adresses IPv4 publiques, pour des ensembles de B4 différents, conformité impérative aux RFC sur le NAT, comme les RFC 4787, RFC 5508 et RFC 5382, précisions sur la possibilité pour l'AFTR d'être aussi un ALG (découragée, vue le nombre de clients que sert un AFTR, et le nombre de protocoles applicatifs présents et futurs), rappel que **tout** partage d'adresses, que ce soit par DS-Lite ou une autre méthode, engendre des ennuis (RFC 6269), etc.

L'annexe A du RFC est particulièrement intéressant pour les administrateurs réseaux car il détaille les scénarios de déploiement possibles. Il couvre des questions comme le placement des AFTR dans le réseau, ou la fiabilité requise des AFTR (ils ont un état donc on ne peut pas juste en multiplier le nombre).

Comme tout nouveau protocole, DS-Lite va soulever des questions de sécurité nouvelles, qu'étudie la section 11 du RFC. En soi, les problèmes de sécurité liés au NAT sont bien connus (RFC 2663 et RFC 2993). Mais déplacer la fonction NAT depuis une machine située chez l'utilisateur vers le réseau du FAI crée des nouveaux défis. Par exemple, les adresses IPv4 publiques, qui n'étaient partagées qu'entre les membres d'une même famille ou les employés d'une même entreprise, vont désormais être partagées entre des clients du même FAI, clients qui ne se connaissent pas. Si la HADOPI voit `203.0.113.201` commettre un crime grave (par exemple partager des œuvres d'art), et qu'elle veut couper l'utilisateur de cette adresse, la probabilité de bavure devient bien plus élevée. Enregistrer les adresses IP ne suffit donc plus, il faut noter l'adresse IP **et** le port (RFC 6302) et que l'AFTR enregistre ses tables de correspondance (identité du tunnel, protocole, adresses et ports), comme précisé en section A.4.

Vu le partage intensif d'adresses (bien plus important qu'avec les NAT sur le CPE), le nombre de ports devient une ressource critique. L'AFTR doit donc faire attention à empêcher les utilisateurs de monter une DoS (volontairement ou par accident) en s'attribuant tous les ports possibles. Par exemple, l'AFTR peut limiter le rythme d'allocation des ports, ou bien mettre une limite absolue au nombre de ports qu'un B4 peut s'allouer.

Enfin, l'AFTR doit veiller à ne pas se transformer lui-même en un outil facilitant les DoS. Par exemple, il ne doit pas permettre à l'autre extrémité du tunnel d'injecter des paquets IPv4 avec d'autres adresses sources que celles prévues (autrement, les réponses à ces paquets frapperaient un innocent).

Notre RFC ne mentionne pas les inconvénients et problèmes de DS-Lite : c'est un mécanisme complexe, avec plusieurs composants qui doivent travailler en bonne intelligence. DS-Lite dépend notamment d'un composant très sollicité, le CGN. Sera t-il suffisant lorsque des dizaines ou des centaines de réseaux locaux utiliseront le même AFTR ? En outre, comme indiqué plus haut, DS-Lite souffre des problèmes liés au partage d'adresses : les lois HADOPI ou LCEN ne seront pas contentes.

Si, à ce stade, vous êtes convaincu de l'intérêt de DS-Lite dans votre cas, où trouver des implémentations ? Il existe un AFTR en logiciel libre à l'ISC, disponible en `<http://www.isc.org/software/aftr>`. Comcast a aussi produit un code pour Linksys (apparemment pas très stable mais suffisant pour des tests) et un pour Mac OS (nommé ComcastDSLiteClient). L'ISC a rassemblé une documentation globale sur le B4 `<http://www.isc.org/software/aftr/build-b4>`. Enfin, les routeurs d'A10 ont la fonction d'AFTR. Verrons-nous bientôt la fonction de B4 dans tous les routeurs et "*boxes*" ? Impossible à dire pour l'instant.

Merci à Fabien Delmotte pour ses connaissances sur les mises en œuvre de DS-Lite.