# RFC 6144 : Framework for IPv4/IPv6 Translation

Stéphane Bortzmeyer
<stephane+blog@bortzmeyer.org>

Première rédaction de cet article le 28 avril 2011

Date de publication du RFC : Avril 2011

https://www.bortzmeyer.org/6144.html

---

Le groupe de travail Behave <http://tools.ietf.org/wg/behave> de l'IETF a travaillé pendant longtemps <https://www.bortzmeyer.org/behave-wg.html> sur les mécanismes permettant de communiquer malgré la présence d'un routeur NAT sur le trajet. Ayant terminé ce travail avec succès, il a été muni d'une nouvelle charte pour travailler sur la transition IPv4 -¿ IPv6, notamment sur un mécanisme de traduction pour permettre à des machines purement v4 de parler à des machines purement v6 et réciproquement. Le groupe « Behave 2 », en ayant fini avec NAT44 s'est donc attaqué à cette tâche et ce RFC est le deuxième de cette nouvelle série (le premier avait été le RFC 6052 [1]), qui décrit NAT64.

Il ne décrit pas une technique ou un protocole mais fournit seulement le cadre général dans lequel vont s'inscrire les autres RFC du nouveau groupe Behave comme le RFC 6145 sur le protocole de traduction ou le RFC 6147 sur l'aide que lui apporte le DNS. La liste de ces documents est en section 3.3.

En effet, l'IETF avait déjà normalisé une technique de traduction d'adresses entre IPv4 et IPv6, le NAT-PT (RFC 2766), technique qui avait été abandonnée officiellement par le RFC 4966. La traduction d'adresses entre v4 et v6 n'était pas forcément considérée mauvaise en soi mais NAT-PT était vu comme une mauvaise solution, pour les raisons expliquées dans le RFC 4966.

Mais il est vrai que la traduction d'adresses a toujours été mal vue à l'IETF et que NAT-PT a du faire face à une forte opposition. Le modèle de transition d'IPv4 vers IPv6 promu par l'IETF (RFC 4213) a toujours été d'activer IPv6 au fur et à mesure, de laisser coexister IPv4 et IPv6 sur le réseau et les machines pendant un moment (stratégie de la « double pile ») puis, une fois qu'IPv6 était généralisé, de supprimer IPv4 petit à petit. Cette méthode était encore raisonnable en 2005, lorsque le RFC 4213 a été publié, mais, aujourd'hui, après des années de ratiocination <https://www.bortzmeyer.org/

1. Pour voir le RFC de numéro NNN, https://www.ietf.org/rfc/rfcNNN.txt, par exemple https://www.ietf.org/rfc/rfc6052.txt

ipv6-et-l-echec-du-marche.html>, alors que l'épuisement des adresses IPv4 est effectif <https://www.bortzmeyer.org/epuisement-adresses-ipv4.html>, elle n'est plus réaliste : nous n'avons tout simplement plus le temps de procéder à une telle migration ordonnée (cf. section 1.4). Des machines purement v6 vont apparaître, tout simplement parce qu'il n'y aura plus d'adresses v4, alors que les machines purement v4 existant aujourd'hui n'auront pas encore commencé à migrer. Les mécanismes de traduction d'adresses d'une famille à l'autre prennent donc tout leur intérêt (section 1 du RFC). Le RFC 5211 propose donc un autre plan de transition, faisant sa place à ces mécanismes.

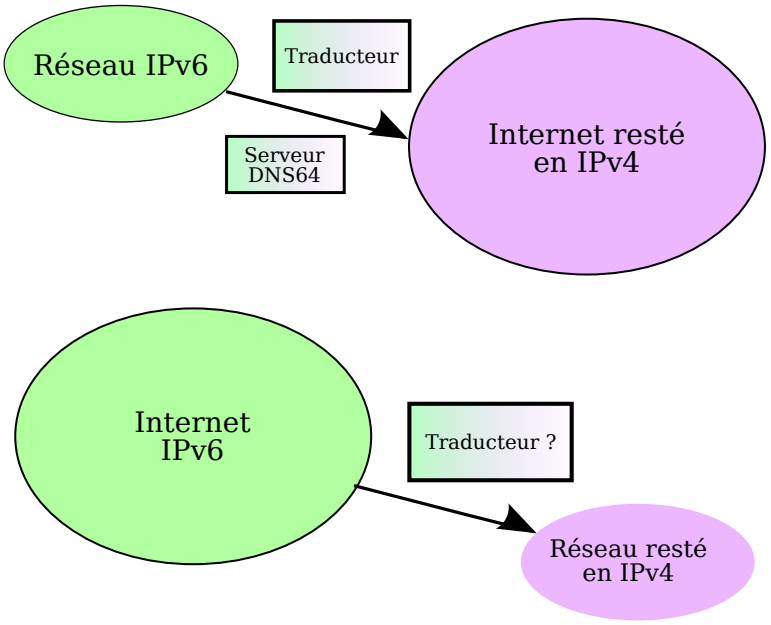
La section 1.1 explique pourquoi considérer les mécanismes de traduction d'adresses, à côté des deux techniques du RFC 4213 (la double-pile mais aussi les tunnels). Si la traduction d'adresses est souvent utilisé pour de mauvaises raisons (par exemple parce qu'elle assurerait la sécurité des machines), elle présente aussi des avantages, notamment le fait qu'elle soit, avec les ALG (cf. section 3.1.5), la seule à permettre la communication entre des machines purement v4 et des machines purement v6. Elle a donc sa place dans les stratégies de transition à moyen terme.

La section 1.3 fixe les objectifs de la traduction d'adresses. Il ne s'agit pas de résoudre tous les problèmes (NAT-PT avait souffert d'objectifs irréalistes) mais d'attraper les fruits les plus bas. Il y a (en simplifiant beaucoup), deux types d'application : les client/serveur comme SSH, où le serveur attend les connexions d'un client, et les pair-à-pair comme SIP, où tout le monde initie et accepte des nouvelles connexions. Celles-ci de subdivisent à leur tour en pair-à-pair « d'infrastructure » (SIP, SMTP, ...) où les connexions sont typiquement d'assez courte durée et pair-à-pair « d'échange » comme avec BitTorrent, où on choisit ses pairs (le plus fiable, le plus rapide) et où on maintient de longues sessions avec eux. Quelles sont les conséquences pratiques de cette typologie ? Les applications client/serveur nécessitent des connexions d'un client v4 vers un serveur v6 (ou réciproquement), idéalement sans que les équipements de traduction d'adresses aient à garder un état. Les applications pair-à-pair nécessitent aussi de telles connexions, et dans les deux sens cette fois, mais elles seront sans doute moins nombreuses que celles d'un client vers un serveur, et les solutions avec état seront donc moins problématiques. Enfin, dans le cas de machines purement clientes (des Minitel), il n'est pas nécessaire d'ouvrir des connexions vers elles. Dans tous les cas, s'il faut garder un état pour la traduction, il est très nettement préférable qu'il soit entièrement dans les équipements réseaux (typiquement le routeur) et que la traduction ne nécessite aucune aide de la part de la machine aux extrémités. NAT-PT exigeait de l'état dans deux endroits (le traducteur lui-même et le serveur DNS) et que ces états soient coordonnés. Un routeur NAT IPv4 actuel a un état mais qui est entièrement dans le routeur et qui ne nécessite aucune participation d'une autre machine.

Enfin, un problème de vocabulaire quasiment philosophique. Faut-il parler de « transition » vers IPv6 sachant que ce mot connote une idée d'achèvement (un jour, il n'y aura plus d'IPv4) alors que rien n'est moins sûr et que, de toute façon, cela prendra de nombreuses années ? Certains préfèrent dire que des mécanismes comme la traduction d'adresses sont des mécanismes de « coexistence » plutôt que de « transition » (section 1.4).

La section 2 du RFC s'attache ensuite à décrire les scénarios où les mécanismes de transition pourront être utiles. Par exemple, en 2.1, le scénario est celui d'un FAI récemment apparu dans un endroit où il est très difficile d'obtenir des adresses IPv4 (par exemple en Asie ou en Afrique) et qui est donc entièrement en IPv6. Ses clients veulent quand même se connecter à des services IPv4. Des techniques comme celles du RFC 7915 peuvent alors être utilisées (rappelez-vous que notre RFC 6144 ne décrit qu'un cadre, pas des techniques spécifiques). Le scénario de la section 2.4 représente le problème inverse : un réseau ancien et jamais mis à jour, entièrement en IPv4, et qui veut accéder aux services uniquement v6. Bien plus difficile à résoudre (le petit nombre d'adresse IPv4 ne permet pas de faire une traduction d'adresses efficace, on doit consommer également les ports et ils ne sont que 65536 par adresse IP), ce problème est considéré comme « hors-sujet ». On ne peut pas résoudre tous les malheurs du monde et ce réseau devra utiliser d'autres techniques (probablement des ALG, même si le RFC ne les cite pas ici).

Le cadre complet de ces solutions à base de traduction d'adresses est en section 3. La section 3.1 liste les composants du cadre :




— La traduction d'adresses (section 3.1.1), comment une adresse v6 est traduite en v4 et réciproquement, ce qui implique un préfixe v6 et une méthode de dérivation des adresses v4 à partir de v6 et réciproquement. Le RFC 6052 donne les détails. Pour les traductions sans état, le même algorithme sert dans les deux sens (v4 vers v6 et l'inverse). Pour celles avec état, l'algorithme ne sert que dans un sens et la table des états est utilisée en sens inverse.
— La modification des paquets IP rendue nécessaire par la traduction d'adresses (par exemple dans ICMP puisqu'un paquet ICMP d'erreur contient une partie du paquet original, avec des adresses IP). La section 3.1.2 renvoie au RFC 6145 qui couvre le cas sans état.
— Le maintien d'un état pour les traductions avec état (section 3.1.3), traitée dans le RFC 6146.
— L'aide que doit apporter le DNS à ces opérations (section 3.1.4), normalisée dans le RFC 6147, qui prévoit la génération automatique d'enregistrements AAAA lorsque seul un enregistrement A existe.
— Les ALG ("*Application Layer Gateways*", section 3.1.5) qui sont nécessaires pour certaines applications comme FTP (le RFC sur FTP64 n'est pas encore paru) qui inclus les adresses IP dans ses données.

Une fois les composants de la solution listés (on voit qu'il faut beaucoup de choses pour faire fonctionner l'Internet malgré le retard pris à migrer vers IPv6), la section 3.2 expose leurs modes de fonctionnement. La traduction sans état (section 3.2.1) passe mieux à l'échelle et maintient la transparence du réseau de bout en bout. Mais elle ne peut pas être utilisée pour tous les scénarios. La traduction avec état (section 3.2.2) peut alors prendre le relais.

Il n'existe pas de solution magique à tous les problèmes. Le déploiement bien trop lent d'IPv6 a laissé une situation peu satisfaisante, et qui ne pourra pas être complètement résolu par de nouveaux protocoles. La section 5 rappelle donc que certaines questions n'ont pas de solution présente. Mais le groupe Behave a dû travailler dans l'urgence, pour fournir des solutions avant l'épuisement d'IPv4 et s'est donc focalisé sur les problèmes les plus « faciles ». Peut-être dans de futurs RFC ?

À noter qu'un FAI britannique a déjà un tel service déployé sur son réseau `<http://aaisp.net.uk/kb-broadband-ipv6-nat64.html>`.