RFC 2993 : Architectural Implications of NAT

Stéphane Bortzmeyer
<stephane+blog@bortzmeyer.org>

Première rédaction de cet article le 19 septembre 2013

Date de publication du RFC : Novembre 2000

https://www.bortzmeyer.org/2993.html

Ce RFC est la synthèse (vieille de treize ans, mais toujours juste) de l'analyse architecturale du NAT par l'IETF. Elle décrit les conséquences du NAT pour l'Internet et appelle à considérer son usage avec prudence (comme chacun peut le voir, cet appel n'a pas été entendu). C'est **le** document à lire pour avoir une vision de haut niveau du NAT.

Techniquement, le NAT a été décrit en 1994 dans le RFC 1631 [1] (et dans le RFC 2663). Cette méthode était un des moyens envisagés pour gérer le problème de l'épuisement des adresses IPv4 <https://www.bortzmeyer.org/epuisement-adresses-ipv4.html>. Mais le RFC 1631 appelait à la prudence : « *"NAT has several negative characteristics that make it inappropriate as a long term solution, and may make it inappropriate even as a short term solution."* » Entre le RFC 1631 et ce RFC 2993 (et encore plus depuis), la traduction d'adresses a été très largement déployée, sans tenir compte de ses inconvénients. Cassandre n'est jamais écoutée... Il avait même été question d'arrêter le développement et le déploiement d'IPv6, avec l'argument que le NAT suffisait à résoudre le problème de l'épuisement d'adresses IPv4 (ce point est développé en détail dans la section 8 du RFC). Il est d'autant plus difficile de faire une analyse des avantages et inconvénients du NAT que le débat est souvent très passionnel. Les Pour font remarquer que le NAT fonctionne sans intervention particulière, pour des applications très populaires (comme le Web ou le courrier). Mieux, le NAT résout un vieux problème d'IP, la confusion de l'identificateur et du localisateur <https://www.bortzmeyer.org/separation-identificateur-localisateur.html> : l'adresse IP interne est un identificateur, l'externe un localisateur. Le NAT divise l'Internet, autrefois unifié autour de l'unicité de l'adresse IP, en plusieurs domaines indépendants, chacun ayant son propre espace d'adressage (souvent tiré du RFC 1918). Les Contre reprochent au NAT son caractère de bricolage et sa non-transparence pour beaucoup d'applications, notamment pair-à-pair. En effet, si l'application vérifie que l'en-tête IP reste cohérent (c'est le cas de l'AH d'IPsec, cf. section 9) ou si elle transporte des adresses IP dans ses données (comme le fait SIP, et pour de bonnes raisons), elle ne fonctionnera pas à travers le NAT sans gros efforts de programmation (le principe du NAT est de déplacer les

1. Pour voir le RFC de numéro NNN, https://www.ietf.org/rfc/rfcNNN.txt, par exemple https://www.ietf.org/rfc/rfc1631.txt

coûts, des FAI vers les programmeurs). Le NAT encourage donc l'ossification de l'Internet vers les seules applications client/serveur. Les Contre notent aussi (RFC 2101) que la division en domaines d'adressage différents font que l'adresse IP cesse d'être unique, et que la résolution d'un nom en adresse donne un résultat qui va dépendre de l'endroit où on pose la question.

Plus fondamentalement, l'Internet repose sur un petit nombre d'excellents principes architecturaux, auxquels il doit son succès, notamment le principe de bout en bout. Ce principe a par exemple pour conséquences qu'il n'y a pas d'état maintenu dans le réseau : un équipement intermédiaire peut redémarrer (et donc tout oublier), un changement de routes peut faire passer par de nouveaux équipements intermédiaires et les extrémités, les machines terminales <https://www.bortzmeyer.org/terminal-host.html> ne s'apercevront de rien, elles ne connaissaient absolument pas les équipements intermédiaires utilisés. Cela contribue fortement à la résilience du réseau. Le NAT met fin à cela.

J'ai écrit NAT jusqu'à présent mais, en fait, le mécanisme de traduction d'adresse le plus répandu dans l'Internet est plutôt le NAPT ("*Network Address and Port Translation*"), c'est à dire qu'il ne traduit pas juste une adresse dans une autre mais un couple {adresse interne, port interne} vers un couple {adresse externe, port externe}. Une grande partie des problèmes qu'on rencontre avec le NAT sont en fait spécifiques au NAPT et n'existeraient pas avec du « vrai » NAT.

Le RFC note que bien des choses qu'on peut dire contre le NAT (cassage du principe de bout en bout, problème pour les applications pair-à-pair) s'appliquent aussi aux pare-feux. La principale différence est que le but explicite d'un pare-feu est de s'insérer sur le trajet et de perturber les communications. En cas de problème, son rôle est donc clair. Au contraire, le NAT est souvent perçu comme transparent, et on oublie plus facilement son rôle, notamment lorsqu'il faut déboguer.

S'appuyant sur le RFC 2663, la section 2 du RFC couvre la terminologie, qui est souvent malmenée lorsqu'il s'agit du NAT :

— NAT ("*Network Address Translation*") est normalement la traduction d'une adresse IP vers une autre. On parle de "*static NAT*" lorsque la correspondance entre les deux adresses est fixe et de "*dynamic NAT*" lorsqu'elle change automatiquement. Le "*static NAT*" nécessite autant d'adresses internes que d'externes et ne règle donc pas le problème de l'épuisement des adresses IPv4.
— NAPT ("*Network Address Port Translation*") est ce que font réellement les routeurs abusivement appelés NAT, comme la "*box*" typique de M. Michu. Elle utilise un numéro de port de la couche 4 pour démultiplexer les paquets à destination de l'adresse IP publique.

Pour étudier l'impact du NAT sur l'Internet, il faut d'abord définir ce qu'est l'Internet. La définition du RFC est que l'Internet est une concaténation <https://www.bortzmeyer.org/catenet.html> de réseaux utilisant les technologies TCP/IP. Selon cette définition, le réseau privé d'une entreprise fait partie de l'Internet. Une définition plus restrictive serait de dire que l'Internet est l'ensemble des machines ayant une adresse IP publique, et pouvant donc envoyer un paquet à toute autre machine de l'Internet. Selon cette seconde définition, un réseau utilisant des adresses IP privées (RFC 1918) et connecté via un NAT ne fait pas partie de l'Internet. En effet, comme le pare-feu, le NAT isole le réseau privé et ne permet pas d'accéder à tous les services de l'Internet. Dans cette définition stricte, seul le réseau public est l'Internet (avec un grand I puisque c'est un objet unique comme l'Atlantique ou le Kilimandjaro).

La section 4 rappelle le principe de bout en bout décrit plus haut (ou dans les RFC 1958 ou RFC 2775). Le document de référence sur ce principe est l'article de J.H. Saltzer, D.P. Reed et David Clark, « "*End-To-End Arguments in System Design*" <http://web.mit.edu/saltzer/www/publications/endtoend/endtoend.pdf> ». Que dit ce principe ? Que tout état mémorisé par une machine du réseau doit être dans les extrémités. Si une machine à l'extrémité redémarre, la communication est fichue de toute façon. En revanche, il faut éviter que le redémarrage d'une machine intermédiaire ne coupe la communication. En effet, il y a beaucoup de machines intermédiaires (faites un traceroute pour voir) et, si la panne de

l'une d'elles perturbait le trafic, l'Internet serait très fragile. Cela illustre la grande différence entre un routeur normal et un routeur NAT : lorsque le routeur normal redémarre (ou change, parce qu'on passe par un autre chemin), la communication continue. Au contraire, le redémarrage d'un routeur NAT fait perdre la table de correspondance entre les tuples {adresse, port} externes et les internes, arrêtant toutes les sessions en cours.

Quelques conséquences importantes du principe de bout en bout sont donc :

— Pas besoin de synchroniser l'état des différents routeurs dans le réseau : un routeur n'a pas de mémoire. En cas de problème, router par un autre chemin est trivial. (Notez que c'est pour cela que faire du NAT lorsqu'on a plusieurs sorties vers l'Internet est très difficile, car il faut que tous les routeurs NAT se partagent leur table de correspondances externe¡-¿interne.)
— Le nombre de machines terminales croit plus vite que celui des routeurs. Si les routeurs devaient garder un état proportionnel au nombre de machines terminales, le réseau passerait bien moins à l'échelle.
— L'absence d'état et, plus généralement d'« intelligence » dans les routeurs, fait que les décisions de sécurité n'ont pas besoin d'être gérées par le réseau. Deux machines qui communiquent s'occupent de leur sécurité et n'ont pas à faire confiance aux équipements intermédiaires. C'est important, par exemple, lorsqu'on voyage, et qu'on ne peut pas être sûr de la confiance à accorder au réseau de l'hôtel ou de l'aéroport.
— Le point ci-dessus s'applique à tous les services : l'absence d'intelligence dans le réseau (qui est liée d'ailleurs à l'idée plus récente de neutralité du réseau <https://www.bortzmeyer.org/neutralite.html>) permet à deux machines consentantes de mettre en œuvre des nouveaux services, sans devoir attendre que quoi que ce soit ait été déployé dans le réseau. Un réseau où une grande partie du travail serait faite dans le réseau lui-même et pas aux extrémités, serait très hostile à l'innovation. Au contraire, le succès de l'Internet est dû en grande partie à la possibilité de déployer de nouvelles applications sans permission. (Et je rajoute que l'un des principaux problèmes de l'Internet d'aujourd'hui est son ossification, due largement au NAT, qui empêche de déployer, par exemple un nouveau protocole de transport comme SCTP.)

Les deux sections suivantes sont consacrées à une mise en comparaison des avantages et inconvénients du NAT (la section 11, en conclusion, les résumera sous forme d'un tableau simple). D'abord, en section 5, ses avantages, dans la résolution de plusieurs problèmes réels :

— En masquant les adresses IP internes aux observateurs extérieurs, il permet de se dispenser de renumérotation lorsqu'on change d'opérateur (le RFC 5887, par la suite, a analysé la difficulté de la renumérotation).
— En cas d'usage intermittent, le NAT permet d'économiser les adresses IP globales en ne les attribuant qu'en cas d'usage effectif. (Cet argument n'a plus guère de sens aujourd'hui, avec les accès « "*always on*" » comme l'ADSL.)
— L'usage d'adresses privées, auto-allouées localement par l'utilisateur, dispense de justifier, auprès d'une grosse bureaucratie <http://www.ripe.net/ripe/docs/current-ripe-documents/request-forms-supporting-notes>, l'obtention d'adresses IP uniques.
— Pour beaucoup d'applications, notamment celles de type client/serveur, aucun changement n'est nécessaire dans les machines terminales <https://www.bortzmeyer.org/terminal-host.html> lorsqu'on déploie le NAT.

Toutes ces raisons expliquent le succès du NAT.

Et les inconvénients ? La section 6 en dresse une liste :

— Le NAT casse le principe de bout en bout.
— Le NAT remplace un modèle simple par un modèle plus complexe, plus fragile et plus difficile à déboguer.
— Il crée un SPOF (le redémarrage du routeur NAT coupe toutes les sessions de toutes les machines du réseau local). Ce point est détaillé dans la section 7.1.
— Il perturbe des techniques de sécurité comme le AH d'IPsec (RFC 4302). Pour cette raison, mais aussi pour d'autres, AH n'a guère été déployé et cet argument n'est donc peut-être pas le plus fort (alors que le RFC, à l'époque, le considérait comme « "*One of the greatest concerns*" »).

— Les adresses privées semblent très sympas au début (on se les alloue soi-même sans s'embêter et sans rien demander à personne) mais sont une grosse source d'ennuis lorsque deux réseaux privés communiquent, par exemple suite à une fusion d'entreprises, ou si on se connecte à deux réseaux distants utilisant la même plage d'adresses (la section 7.6 donne un exemple).
— Le NAT casse les protocoles qui ont une connexion de contrôle séparée de la connexion de données, ce qui est fréquent dans le multimédia (SIP, H.323, et tous les protocoles qui fonctionnent en indiquant une référence à une machine tierce). La connexion de données se fait typiquement en utilisant des adresses IP indiquées par le pair... qui ne marcheront pas car ce sont des adresses privées, le pair ne savait pas qu'il était NATé.
— Le NAT augmente la complexité de la connexion : c'est un facteur de plus à prendre en compte lorsqu'on débogue.
— Les connexions entrantes sont impossibles par défaut (ce qui pose problème si on veut héberger un serveur sur le réseau local). On peut sur les routeurs NAT typiques définir une correspondance d'un port bien connu (par exemple 80 pour HTTP) vers une machine interne mais cela ne permet qu'un seul serveur pour chaque port bien connu.
— Depuis l'écriture de ce RFC 2993, UPnP, puis le PCP ("*Port Control Protocol*") du RFC 6887, ont facilité un peu la vie des applications coincées derrière un routeur NAT, en leur permettant un contrôle limité des correspondances statiques entre {adresse, port} externes et internes.

Il y a aussi des inconvénients plus techniques, comme le fait que le délai d'attente de TCP avant de réutiliser un identificateur de connexion (l'état `TCP_TIME_WAIT`) peut vite être trop long pour un routeur NAT actif, le soumettant à la tentation d'ignorer ce délai minimum (deux fois la durée de vie d'un segment dans le réseau, normalement).

Par contre, ce RFC oublie un inconvénient important du NAPT (mais pas du NAT) : le partage d'adresses IP publiques, dont les conséquences négatives ont été analysées onze ans plus tard dans le RFC 6269.

La question de la sécurité du NAT occupe la section 9. C'est un point d'autant plus important que certaines personnes prétendent parfois que le NAT améliore la sécurité <`https://www.bortzmeyer.org/nat-et-securite.html`>. En fait, notre RFC note que cela peut être tout le contraire, en créant l'illusion d'une barrière, d'un pare-feu, le NAT peut abaisser le niveau de sécurité. Le RFC critique aussi cette idéologie du pare-feu, qui considère que toutes les menaces sont externes, dans le style du "*limes*" romain. Si on pense qu'on est protégé derrière son routeur NAT, c'est qu'on ignore une règle de base de la sécurité, qui est que l'attaque vient souvent de l'intérieur.

Si le NAT, par définition, casse les techniques de sécurité qui protègent l'intégrité du paquet, comme l'AH d'IPsec, d'autres techniques de sécurité peuvent passer malgré le NAT. C'est le cas de TLS ou de SSH. Notez que ces techniques fonctionnent dans la couche 7 et restent donc vulnérables à des attaques qu'IPsec aurait empêchées, comme les "*resets*" TCP (cf. RFC 2385).

La section 10 contient quelques conseils de déploiement, pour le cas où on décide de faire du NAT. Le RFC recommande de chercher un mécanisme de résolution de noms qui gère le fait que les adresses soient différentes en interne et en externe, et suggère qu'un serveur DNS intégré au routeur NAT est sans doute dans la meilleure position pour faire cela. Il insiste sur le déploiement d'ALG pour les applications qui ne passent pas à travers le NAT (ou bien l'établissement d'un tunnel au dessus du NAT). Il fait remarquer (c'était une idée novatrice à l'époque) qu'il existe plein de variantes de NAPT, avec des comportements différents. Par exemple, si une application émet un paquet depuis le tuple {adresse interne, port interne} vers une destination extérieure, son tuple interne va être réécrit en {adresse externe, port externe}. Si cette application écrit ensuite à une autre destination extérieure, depuis le même tuple {adresse interne, port interne}, celui-ci sera t-il réécrit dans le même tuple {adresse externe, port externe} ou pas ? Tous les routeurs NAT n'ont pas forcément le même comportement sur ce point, et cela influe beaucoup sur les chances d'une application de passer à travers le NAT (RFC 4787).

Le RFC a aussi des suggestions plus surprenantes comme l'idée d'un registre des adresses privées, pour diminuer les risques de collision dans le cas où un client a des VPN vers deux réseaux privés utilisant la même plage d'adresses. Si cela a un sens au sein d'une même organisation (dont on peut espérer qu'elle a déjà un plan d'adressage cohérent, évitant les collisions), l'idée d'un registre plus large va à l'encontre du principe des adresses privées.

Depuis la sortie de ce RFC, l'IETF a accepté le principe d'un travail pour limiter les dégâts du NAT, le groupe de travail Behave <https://www.bortzmeyer.org/behave-wg.html>, qui a produit plein d'excellents RFC. L'idée même d'un tel travail gênait certains à l'IETF, car ils pensaient qu'améliorer le NAT allait retarder son démantèlement souhaitable. Mais la prévalence du NAT a conduit à une position de compromis, avec un certain nombre de techniques permettant de vivre avec le NAT.

En résumé, ce RFC est hostile à la généralisation du NAT, non pas en disant qu'il ne marche pas (« "*it is possible to build a house of cards*" ») mais en notant qu'il restera toujours un bricolage fragile. Le problème de l'épuisement des adresses IPv4 <https://www.bortzmeyer.org/epuisement-adresses-ipv4.html> doit se résoudre par le déploiement d'IPv6, pas par le NAT. On notera que cet avis n'a pas été suivi dans les treize années écoulées depuis la parution de ce RFC : l'IETF peut dire ce qu'elle veut, cela ne change rien en pratique. Les décisions ne sont pas prises par elle.