

RFC 2460 : Internet Protocol, Version 6 (IPv6) Specification

Stéphane Bortzmeyer

<stephane+blog@bortzmeyer.org>

Première rédaction de cet article le 1 juillet 2008

Date de publication du RFC : Décembre 1998

<https://www.bortzmeyer.org/2460.html>

Si IPv4 est normalisé dans un seul document, le RFC 791¹, le candidat à sa succession IPv6 est décrit de manière plus modulaire, dans plusieurs RFC dont notre RFC 2460, essentiellement consacré au format des paquets. Il a depuis été remplacé par le RFC 8200.

Remplaçant le premier RFC sur le format des paquets IPv6, le RFC 1883, notre RFC 2460 est resté stable pendant dix-huit ans, alors que la plupart des RFC sur IPv6 ont connu des révisions; les autres RFC importants pour IPv6 sont le RFC 4291 sur l'adressage - certainement le moins stable -, le RFC 4861 sur le protocole de découverte des voisins, le RFC 4862 sur l'autoconfiguration des adresses et le RFC 4443 sur ICMP.

À l'heure actuelle, le déploiement de ce RFC est très limité : peu de paquets IPv6 circulent sur Internet et il est loin d'avoir succédé à IPv4. Le fait d'avoir un numéro de version plus élevé ne suffit pas! Notre RFC 2460 présente sans hésiter IPv6 comme le successeur d'IPv4, ce qui reste pour l'instant un vœu pieux.

IPv6 reprend largement le modèle qui a fait le succès d'IPv4 : un protocole de datagrammes, un routage jusqu'au prochain saut (peu ou pas de routage de bout en bout), des datagrammes acheminés « au mieux possible », un en-tête de paquet simple et où les champs sont de taille fixe. La section 1 du RFC résume les différences entre les formats de paquets IPv4 et IPv6.

La plus spectaculaire des différences est bien sûr l'augmentation de la taille des adresses. Toujours fixe (contrairement à des concurrents où les adresses étaient de tailles variables), cette taille passe de 32 à 128 bits. Le maximum théorique d'adresses (en supposant une allocation parfaite) passe donc de

1. Pour voir le RFC de numéro NNN, <https://www.ietf.org/rfc/rfcNNN.txt>, par exemple <https://www.ietf.org/rfc/rfc791.txt>

quatre milliards (ce qui ne permet même pas d'allouer une adresse à chaque personne sur Terre) à plus de $10^{\wedge}40$, un nombre qui défie l'imagination.

Compte-tenu de l'épuisement rapide des adresses IPv4 <<http://www.potaroo.net/tools/ipv4/>>, cette augmentation est à elle seule une bonne raison de migrer vers IPv6. Elle reflète les changements de paradigme successifs en informatique, de « un ordinateur par entreprise » au début des années 70, lorsque IPv4 a été conçu, à « un ordinateur par département » au cours des années 80 puis à « un ordinateur par personne » avec l'explosion de la micro-informatique et à « plusieurs ordinateurs par personne » de nos jours lorsque le cadre dynamique et branché se promène avec davantage d'appareils électroniques sur lui que n'en portait James Bond dans ses premiers films. Les quatre milliards d'adresses d'IPv4 sont donc aujourd'hui bien insuffisantes.

Mais ce changement de format des adresses est également la faiblesse d'IPv6 : il ne permet en effet pas à des machines v4 et v6 de communiquer directement et les premiers adopteurs du nouveau protocole ne peuvent donc pas parler aux anciens, ce qui a puissamment freiné le déploiement d'IPv6.

Les autres changements apportés par IPv6 sont moins connus mais semblaient à l'époque justifier le nouveau protocole :

- Le format de l'en-tête est simplifié, facilitant la tâche des routeurs (en pratique, IPv4 garde son avantage, la complexité de son format étant compensée par l'expérience des développeurs et la quantité de main d'œuvre qui a travaillé à optimiser les routeurs IPv4).
- Les options sont à la fois plus libres, notamment en taille, et moins contraignantes à gérer pour les routeurs,
- Les services d'authentification et de confidentialité ont été inclus (cet argument est souvent cité en faveur d'IPv6 mais il est largement bidon, IPv4 ayant désormais les mêmes capacités, et elles n'ont pas souvent été intégrées dans les implémentations IPv6).

La section 3 détaille le format du nouvel en-tête de paquet.

```

+++++
|Version| Traffic Class |           Flow Label           |
+++++
|           Payload Length           | Next Header | Hop Limit |
+++++
|
+
|
+           Source Address
+
|
+++++
|
+
|           Destination Address
+
|
+
|
+++++

```

Contrairement aux paquets IPv4, il n'y a pas de champ « Options » de taille variable. Le champ « Version » contient évidemment le numéro 6 (il y a eu des protocoles ayant reçu le numéro 5 ou 7 <<https://www.iana.org/assignments/version-numbers>> mais ils n'ont jamais été déployés). Le champ DSCP, ex-TOS, d'IPv4 est remplacé par les champs "Traffic class" et "Flow label" qui semblent très peu utilisés en pratique (sections 6 et 7 du RFC, ainsi que l'annexe A; le "flow label" a été ensuite mieux normalisé dans le RFC 6437).

Le champ *"Next header"* indique le type de l'en-tête suivant. La plupart du temps, c'est l'en-tête d'un protocole de transport comme TCP (numéro 6) ou DCCP (numéro 33), dont les numéros sont stockés dans un registre IANA <<https://www.iana.org/assignments/protocol-numbers>>. Mais cela peut être aussi un en-tête d'extension de la couche réseau par exemple 44 pour les fragments (ces numéros sont stockés dans le même registre IANA). Notez que, depuis, le RFC 6564 a modifié les règles pour les futurs en-têtes, imposant un format commun. (Pour analyser les paquets IPv6, voir mon article « Analyser les en-têtes IPv6 avec pcap <<https://www.bortzmeyer.org/analyse-pcap-ipv6.html>> ».)

Vu avec `tshark`, la version texte de l'analyseur Wireshark, et son option `-v`, voici un paquet IPv6 contenant lui-même du TCP :

```
Internet Protocol Version 6
0110 .... = Version: 6
  [0110 .... = This field makes the filter "ip.version == 6" possible: 6]
.... 0000 0000 .... .... .... .... = Traffic class: 0x00000000
.... .... .... 0000 0000 0000 0000 0000 = Flowlabel: 0x00000000
Payload length: 40
Next header: TCP (0x06)
Hop limit: 64
Source: 2a01:e35:8bd9:8bb0:21e:8cff:fe76:29b6 (2a01:e35:8bd9:8bb0:21e:8cff:fe76:29b6)
Destination: 2a01:e35:8bd9:8bb0:21c:23ff:fe00:6b7f (2a01:e35:8bd9:8bb0:21c:23ff:fe00:6b7f)
```

La section 4 présente les en-têtes d'extension. Grande nouveauté d'IPv6, ils sont situés entre l'en-tête IP normal et l'en-tête de la couche transport. La plupart d'entre eux sont ignorés par les routeurs situés sur le trajet et traités seulement à l'arrivée (alors qu'un routeur IPv4 devait, en théorie, toujours examiner les options). Notre RFC normalise certaines de ces extensions, mais d'autres peuvent être ajoutées par des RFC ultérieurs comme par exemple le RFC 3775 qui normalise le *"Mobility Header"* (numéro 135, section 6.1 du RFC).

C'est ainsi que la section 4.3 décrit l'en-tête d'extension « *"Hop-by-hop"* », une de celles qui doivent être examinées par tous les routeurs sur le trajet. Elle contient des options (décrites en section 4.2), qui sont stockées sous forme de tuples TLV. Le type de chaque option indique, dans ses deux premiers bits, si l'option peut être ignorée ou non.

Le symétrique de cet en-tête est le « *"Destination"* », section 4.6, qui stocke, de la même façon, les options qui ne doivent être traitées que par le destinataire. (Pour mettre un tel en-tête dans vos paquets, voir mon article <<https://www.bortzmeyer.org/destination-options-ipv6.html>>.)

La section 4.4 décrit l'en-tête « *"Routing"* » qui permet d'indiquer la route qu'on souhaite voir le paquet prendre, ou bien d'enregistrer la route effectivement suivie. Notons qu'un champ, le « *"Routing Type"* », indique le type de routage souhaité et que sa valeur 0 ne doit plus être utilisée : en raison de problèmes de sécurité sérieux, le *"Type 0 Routing Header"* a été abandonné dans le RFC 5095.

La section 4.5 décrit l'en-tête « *"Fragment"* » qui permet de représenter des paquets IP qui, trop longs, ont été fragmentés par l'émetteur (contrairement à IPv4, les routeurs IPv6 n'ont pas le droit de fragmenter).

La section 5 décrit les questions liées à la taille des paquets. IPv6 impose une MTU minimale de 1280 octets et, en théorie, une machine qui n'enverrait que des paquets de cette taille (ou plus petits) n'aurait jamais de problème. Au delà, on doit utiliser la découverte de la MTU du chemin, spécifiée dans le RFC 1981 mais dont le moins qu'on puisse dire est qu'elle n'est pas très fiable (cf. RFC 4821).

Les différences avec le RFC 1883 sont décrites après la bibliographie. Elles sont en général de peu d'importance. Par exemple, la MTU minimale passe à 1280 octets (après des débats enfiévrés <<http://playground.sun.com/ipv6/minutes/IPng-Meeting.Feb95.txt>>, à une époque où LocalTalk était une technologie assez courante, le RFC 1883 utilisait 576 octets). Outre changement, la possibilité d'envoyer des très grands paquets, les "*jumbograms*", a disparu du RFC de base et figure désormais dans un RFC dédié, le RFC 2675. Le RFC 8200, qui a remplacé notre RFC, n'a pas apporté de changements cruciaux.