Engineer

IPv6

Version du protocole IP aux adresses de 128 bits

Ce que c’est

IPv6, dont le nom complet est protocole Internet version 6, est la version du protocole Internet conçue pour succéder à IPv4, décrite par la RFC 791. Sa spécification actuelle est la RFC 8200 de juillet 2017, une norme Internet (STD 86) qui a remplacé la RFC 2460. Le changement le plus visible est la taille d’une adresse : 128 bits au lieu des 32 d’IPv4. Tout IPv4 compte 4 294 967 296 adresses ; un seul sous-réseau IPv6 de taille habituelle, un /64, en contient le carré.

Le reste du paquet a changé aussi. La RFC 8200 supprime certains champs d’en-tête d’IPv4 ou les rend facultatifs pour réduire le coût de traitement, place les informations facultatives dans des en-têtes d’extension séparés après l’en-tête principal et renomme la durée de vie, Time to Live, en limite de sauts, Hop Limit, que chaque nœud de transfert diminue d’une unité. Il n’y a pas de somme de contrôle de l’en-tête. Les routeurs ne divisent jamais un paquet : en IPv6, seule la source fragmente, chaque lien doit transporter des paquets d’au moins 1 280 octets, et les émetteurs sont vivement encouragés à utiliser la découverte de la MTU du chemin, Path MTU Discovery, pour déterminer la taille des paquets qu’un chemin peut transporter.

Une adresse s’écrit en huit groupes de 16 bits en hexadécimal, séparés par des deux-points. La RFC 4291 permet d’omettre les zéros initiaux d’un groupe et de remplacer une suite de groupes entièrement nuls par un double deux-points, qui ne peut apparaître qu’une fois : 2001:db8:0:0:8:800:200c:417a peut ainsi s’écrire 2001:db8::8:800:200c:417a. Puisque cela laisse plusieurs façons d’écrire la même adresse, la RFC 5952 d’août 2010 en choisit une : des minuscules, aucun zéro initial et le double deux-points sur la plus longue suite de groupes nuls — la première en cas d’égalité —, jamais sur un seul groupe. Hormis les adresses particulières comme celles de bouclage et les adresses locales de lien, toutes les

adresses et tous les préfixes d'exemple de cet article proviennent d'une plage de documentation : 2001:db8::/32, réservée par la RFC 3849, ou, pour IPv4, les blocs réservés par la RFC 5737.

La RFC 4291 change aussi ce à quoi l'adresse est rattachée. Les adresses appartiennent aux interfaces, pas aux machines, et une interface peut en posséder plusieurs à la fois. Toute interface doit avoir une adresse locale de lien sous fe80::/10, valable uniquement sur son propre lien. L'adresse de bouclage est ::1. Il n'y a pas de diffusion générale : la multidiffusion remplit ce rôle, et ff02::1 atteint tous les nœuds du lien. Les adresses locales uniques sous fc00::/7, définies dans la RFC 4193, servent à l'intérieur d'un site. Les adresses de monodiffusion globale sont utilisées pour communiquer au-delà du site local. Pour la plupart des adresses, les 64 derniers bits constituent l'identifiant d'interface, d'où la taille /64 d'un segment réseau ordinaire.

Les hôtes s'attribuent généralement leurs propres numéros. Avec l'autoconfiguration d'adresses sans état de la RFC 4862, les routeurs annoncent les préfixes d'un lien dans des annonces de routeur périodiques, Router Advertisements. Chaque hôte combine un préfixe avec son identifiant d'interface et, avant d'utiliser le résultat, exécute la détection des adresses dupliquées, Duplicate Address Detection, en envoyant des sollicitations de voisin, Neighbor Solicitation, pour vérifier qu'aucun autre nœud ne possède l'adresse. DHCPv6 est l'autre possibilité lorsqu'un site veut contrôler plus étroitement les attributions exactes, et les deux peuvent fonctionner côte à côte.

Utilisez le concept pour prendre une décision

Utilisez « Qu'est-ce qu'IPv6 ? » pour éclairer une décision réelle. Nous pouvons vous aider à relier ce concept au système que vous construisez ou achetez. Du Wi-Fi, des pare-feu et des accès à distance configurés pour travailler de partout, en laissant entrer uniquement les personnes autorisées.

Découvrir nos prestations Contactez-nous →

Vérifiez les hypothèses derrière le terme

Appuyez-vous sur l'explication et les réalisations liées pour formuler les questions de votre projet. Nous pouvons examiner les hypothèses avec vous avant le choix d'un outil. 30 instances VPS Ubuntu Linux administrées — 30 serveurs Linux gérés, avec récupération prévue

30 instances VPS Ubuntu Linux administrées Contactez-nous →

Pourquoi c'est important

Pour de nombreuses équipes, IPv6 n'est pas un projet qu'elles décident de lancer. Le principe de la RFC 4862 est qu'un hôte n'a besoin de personne pour y saisir une adresse : placez un serveur sur un lien où un routeur annonce un préfixe, et il peut obtenir seul une adresse IPv6 globale. Toute liste d'autorisation ou règle de

pare-feu, tout analyseur de journaux et toute sonde de santé conçus uniquement pour IPv4 présentent alors un angle mort de la taille exacte du nouvel espace d'adressage.

Cet angle mort passe facilement inaperçu, car les deux versions sont des protocoles distincts qui ne partagent rien sur le réseau. Un filtre de paquets examine leurs en-têtes séparément, une limite de débit fondée sur des adresses IPv4 ne voit jamais un client IPv6, et un test qui ne se connecte qu'à 127.0.0.1 ne dit rien de ::1. Un hôte verrouillé sur une version et ouvert sur l'autre n'a, en pratique, pas de pare-feu.

IPv6 dépend aussi davantage de ses messages de contrôle qu'IPv4. Découverte des voisins, annonces de routeur et découverte de la MTU du chemin passent toutes par ICMPv6. Une habitude héritée d'IPv4, qui consiste à rejeter ICMP à la périphérie, casse donc IPv6 de façons qui ne se révèlent parfois qu'au début des transferts de données. Et comme une adresse possède plusieurs écritures valides, le code qui compare les adresses comme des chaînes se contredit lui-même.

À quoi cela ressemble en pratique

Le module standard `ipaddress` de Python analyse toutes les écritures valides et affiche une forme courte, sans zéros initiaux et avec la plus longue suite de groupes de zéros comprimée. C'est donc un moyen rapide de planifier un préfixe et de vérifier que deux écritures désignent la même adresse. La session ci-dessous est une illustration propre à cet article, exécutée sur Python 3.15 avec des adresses du préfixe de documentation :

```
>>> import ipaddress
>>> site = ipaddress.ip_network("2001:db8:42::/48")
>>> site.num_addresses == 2**80
True
>>> lans = site.subnets(new_prefix=64)
>>> next(lans), next(lans)
(IPv6Network('2001:db8:42::/64'), IPv6Network('2001:db8:42:1::/64'))
>>> len(list(site.subnets(new_prefix=64)))
65536
>>> a = ipaddress.ip_address("2001:0DB8:0000:0000:0001:0000:0000:0001")
>>> a
IPv6Address('2001:db8::1:0:0:1')
>>> a.exploded
'2001:0db8:0000:0000:0001:0000:0000:0001'
>>> a == ipaddress.ip_address("2001:db8:0:0:1::1")
```

```
True
>>> ipaddress.ip_address("fe80::1%eth0").scope_id
'eth0'
```

Un /48 laisse 80 bits libres et, découpé en segments /64, en fournit 65 536, numérotés par le quatrième groupe. L'adresse saisie en majuscules avec tous ses zéros revient sous la forme courte 2001:db8::1:0:0:1 : le double deux-points couvre la première de deux suites de zéros de même longueur. Elle est jugée égale à 2001:db8:0:0:1::1, une autre écriture valide des mêmes 128 bits. La dernière ligne montre une adresse locale de lien portant sa zone, le `%eth0` qui indique à quel lien elle appartient.

Sur un hôte Linux utilisant nftables, la deuxième partie de cette illustration est un jeu de règles qui autorise SSH depuis un bureau avec les deux versions. Le manuel nft(8) décrit `inet` comme une famille hybride dont les tables voient les paquets IPv4 comme IPv6, tandis que `ip saddr` lit l'en-tête IPv4 et `ip6 saddr` l'en-tête IPv6 :

```
add table inet filter
add chain inet filter input { type filter hook input priority filter; }
add rule inet filter input ip saddr 198.51.100.0/24 tcp dport 22 accept
add rule inet filter input ip6 saddr 2001:db8:42::/48 tcp dport 22 accept
add rule inet filter input tcp dport 22 drop
```

Sans la quatrième ligne, la connexion d'un administrateur depuis le préfixe IPv6 du bureau ne correspond à aucune des règles d'autorisation et est rejetée, un échec au moins visible. L'échec plus discret est l'inverse : un jeu de règles construit uniquement dans la famille `ip` , sur un hôte possédant aussi une adresse IPv6, n'examine jamais les paquets IPv6.

La réalisation Provisionnement d'un deuxième serveur à partir de code, de septembre 2026, aborde la même exposition du côté du provisionnement. La première machine avait été créée à la main et son pare-feu ajouté ensuite, laissant une période où un hôte neuf, avec sa politique initiale de mots de passe, était joignable depuis Internet. Le deuxième n'est créé qu'après son pare-feu : le play du pare-feu s'exécute avant celui du serveur, et les pare-feu des deux fournisseurs sont gérés directement par leurs API REST. Cette page ne mentionne pas IPv6 : les règles ci-dessus sont donc une illustration de cet article, pas celles de cet hôte. Ce que l'article ajoute, c'est que le pare-feu doit couvrir toutes les adresses que l'hôte possédera, IPv6 comprise, avant que l'hôte existe.

Où les problèmes surviennent

Un pare-feu qui ne connaît qu'IPv4. Les règles de la famille `ip`, ou les listes d'autorisation ne contenant que des préfixes IPv4, abandonnent toutes les adresses IPv6 de l'hôte à ce qui suit. Écrivez les règles dans la famille `inet` ou maintenez les tables `ip` et `ip6` synchronisées, associez à chaque règle `ip saddr` une règle `ip6 saddr` et testez chaque service en vous connectant à son adresse IPv6 comme à son adresse IPv4.

Rejeter ICMPv6 en bloc. La RFC 4890, consacrée au filtrage de ces messages par les pare-feu, l'énonce clairement : « ICMPv6 est essentiel au fonctionnement d'IPv6. » Un routeur qui ne peut pas transférer un paquet trop grand pour le lien suivant le rejette et renvoie un message de paquet trop grand, Packet Too Big. La RFC 8201 précise que si le filtrage arrête ces messages, la source ne découvre jamais la MTU du chemin. Il en résulte ce qu'elle appelle une connexion en trou noir : la négociation TCP réussit, puis le transfert se bloque. La RFC 4890 cite aussi les sollicitations et annonces de voisin, Neighbor Solicitation et Neighbor Advertisement, ainsi que les sollicitations et annonces de routeur, Router Solicitation et Router Advertisement, parmi les messages qu'un nœud ne doit pas rejeter pour participer à un réseau IPv6. Acceptez les types nommés `packet-too-big`, `nd-neighbor-solicit`, `nd-neighbor-advert`, `nd-router-solicit` et `nd-router-advert` par nft avant toute règle de rejet plus large.

Comparer les adresses comme du texte. La RFC 5952 donne l'exemple d'une recherche de 2001:db8:0:0:1::1 dans un tableur qui ne trouve pas une entrée écrite 2001:db8::1:0:0:1. C'est important lorsqu'on consulte cette liste avant d'attribuer une nouvelle adresse pour vérifier qu'elle n'est pas déjà utilisée. Analysez les adresses avant de les comparer, puis stockez et affichez une forme canonique, comme la forme courte que le module `ipaddress` de Python renvoie avec `str()`.

Une adresse accolée à un port. 2001:db8::1:80 peut désigner une adresse ou une adresse suivie du port 80 ; la RFC 5952 qualifie cette écriture d'ambiguë et recommande des crochets, [2001:db8::1]:80, comme dans les URL. Écrivez toujours les crochets et, dans le code, séparez l'hôte du port au crochet fermant plutôt qu'au dernier deux-points.

Un préfixe de lien qui n'est pas un /64. L'autoconfiguration exige que le préfixe et l'identifiant d'interface totalisent 128 bits. Avec des identifiants de 64 bits, la RFC 4862 impose à l'hôte d'ignorer les préfixes annoncés d'une autre longueur :

les hôtes d'un lien dont les annonces de routeur portent un /72 ou un /56 ne forment donc aucune adresse à partir de ce préfixe. Prévoyez un /64 par lien et découpez les allocations plus grandes en /64, comme dans la session ci-dessus.

Des adresses locales de lien sans leur zone. Chaque interface possède une adresse dans fe80::/10, et la RFC 4291 limite ces adresses à un seul lien : sur un hôte à plusieurs interfaces, l'adresse seule ne dit donc pas quel lien est visé. La documentation de Python précise que deux adresses qui ne diffèrent que par leur zone ne sont pas égales. Conservez la zone avec l'adresse (fe80::1%eth0), ou attribuez aux services une adresse globale ou locale unique et gardez les adresses locales de lien pour le lien lui-même.

Pour aller plus loin

- RFC 8200 : spécification du protocole Internet version 6 (IPv6) — la norme IPv6 actuelle, avec le format de l'en-tête, les en-têtes d'extension et les règles de fragmentation et de taille des paquets.
- RFC 4291 : architecture d'adressage d'IP version 6 — types d'adresses, formes textuelles, adresses locales de lien et de multidiffusion, et identifiants d'interface de 64 bits.
- RFC 5952 : recommandation pour la représentation textuelle des adresses IPv6 — la manière canonique d'écrire une adresse et les difficultés causées par la multiplicité des écritures.
- RFC 4862 : autoconfiguration d'adresses IPv6 sans état — formation des adresses par les hôtes à partir des préfixes annoncés et détection des doublons.
- RFC 4890 : recommandations pour le filtrage des messages ICMPv6 par les pare-feu — messages ICMPv6 qu'un pare-feu doit laisser passer et raisons de cette obligation.
- ipaddress : bibliothèque de manipulation d'IPv4 et d'IPv6 — la référence de Python pour analyser, comparer et découper les adresses et réseaux IPv6.

Glossaire

Choisissez selon vos contraintes réelles

Présentez le parcours de déploiement, les risques opérationnels et les personnes d'astreinte. Nous pouvons définir un changement avec des vérifications, une procédure de reprise et une responsabilité claire.

Consulter les tarifs Contactez-nous →

Rendez la terminologie concrète

Demandez comment « Qu'est-ce qu'IPv6 ? » s'applique à votre organisation. Un exemple concret tiré de votre système actuel suffit pour engager la discussion.

Découvrir nos prestations

welcome@engineer.company +45 50 10 61 84 Copenhague, Danemark

Emportez cette page avec vous

Fiche de contact

Créé avec soin et amour par D.

CVR 45969703 · Tagensvej 105C, 3. th, 2200 København N

Droit d'auteur © 2025 – aujourd'hui · Engineer ApS · Tous droits réservés.

Imprimé depuis https://engineer.company/fr/glossary/ipv6/