Groupe de travail Réseau
Request for Comments : 3484
Catégorie : En cours de normalisation

R. Draves, Microsoft Research
février 2003
Traduction Claude Brière de L'Isle

Choix d'adresse par défaut pour le protocole Internet version 6 (IPv6)

Statut du présent mémoire
Le présent document spécifie un protocole de l'Internet en cours de normalisation pour la communauté de l'Internet, et appelle à des discussions et suggestions pour son amélioration. Prière de se référer à l'édition en cours des "Protocoles officiels de l'Internet" (STD 1) pour voir l'état de normalisation et le statut de ce protocole. La distribution du présent mémoire n'est soumise à aucune restriction.

Résumé
Le présent document décrit deux algorithmes, pour le choix d'adresse de source et pour le choix d'adresse de destination. Les algorithmes spécifient un comportement par défaut pour toutes les mises en œuvre du protocole Internet version 6 (IPv6). Ils n'outrepassent pas les choix faits par les applications ou les protocoles de couche supérieure, ni n'empêchent le développement de mécanismes plus avancés de choix d'adresse. Les deux algorithmes partagent un contexte commun, incluant un mécanisme facultatif qui permet aux administrateurs de fournir la politique qui peut outrepasser le comportement par défaut. Dans les mises en œuvre de double pile de protocoles, l'algorithme de choix d'adresse de destination peut prendre en considération aussi bien les adresses IPv4 que IPv6 – selon les adresses de source disponibles, l'algorithme peut préférer les adresses IPv6 aux adresses IPv4, ou vice-versa.

Tous les nœuds IPv6, y compris les hôtes et les routeurs, doivent mettre en œuvre le choix d'adresse par défaut comme défini dans la présente spécification.

Table des matières

1. Introduction

L'architecture d'adressage IPv6 [RFC2373] permet que plusieurs adresses d'envoi individuel soient allouées aux interfaces. Ces adresses peuvent avoir des portées d'accessibilité différentes (liaison locale, site local, ou mondiales). Ces adresses peuvent aussi être "préférées" ou "déconseillées" [RFC2462]. Les considérations de confidentialité ont introduit les concepts de "adresses publiques" et "adresses temporaires" [RFC3041]. L'architecture de mobilité introduit les "adresses de rattachement" et "adresses d'entretien" [RFC3775]. De plus, les situations de multi rattachement vont résulter en plus d'adresses par nœud. Par exemple, un nœud peut avoir plusieurs interfaces, dont certaines sont des tunnels ou des interfaces virtuelles, ou un site peut avoir plusieurs rattachements de FAI *(fournisseur d'adresse Internet)* avec un préfixe mondial par FAI.

Le résultat final est que les mises en œuvre de IPv6 vont très souvent se trouver en face de plusieurs adresses de source et de destination possibles lors de l'initiation d'une communication. Il est souhaitable d'avoir des algorithmes par défaut, communs à toutes les mises en œuvre, pour choisir les adresses de source et de destination afin que les développeurs et les administrateurs puissent réfléchir sur, et prédire, le comportement de leurs systèmes.

De plus, des mises en œuvre de piles duelles ou hybrides, qui prennent en charge à la fois IPv6 et IPv4, vont très souvent avoir besoin de choisir entre IPv6 et IPv4 lors de l'initiation de la communication. Par exemple, quand la résolution de noms du DNS donne à la fois des adresses IPv6 et IPv4 et que la pile de protocoles réseau a disponibles à la fois des adresses de source IPv6 et IPv4. Dans de tels cas, une simple politique de toujours préférer IPv6 ou de toujours préférer IPv4 peut produire un mauvais comportement. Par exemple, supposons qu'un nom du DNS se résolve en une adresse IPv6 mondiale et une adresse IPv4 mondiale. Si le nœud a alloué une adresse IPv6 mondiale et une adresse IPv4 auto-configurée 169.254/16 [RFC3927], alors IPv6 est le meilleur choix pour la communication. Mais si le nœud a alloué seulement une adresse IPv6 de liaison locale et une adresse IPv4 mondiale, alors IPv4 est le meilleur choix pour la communication. L'algorithme de choix d'adresse de destination résout cela avec une procédure unifiée pour choisir entre les deux adresses IPv6 et IPv4.

Les algorithmes du présent document sont spécifiés comme un ensemble de règles qui définissent un ordre partiel sur l'ensemble des adresses qui sont disponibles à l'utilisation. Dans le cas du choix d'adresse de source, un nœud a normalement plusieurs adresses allouées à ses interfaces, et les règles d'ordre des adresses de source de la section 5 définissent quelle adresse est la "meilleure" à utiliser. Dans le cas de choix d'adresse de destination, le DNS peut retourner un ensemble d'adresses pour un certain nom, et une application a besoin de décider laquelle utiliser d'abord, et dans quel ordre essayer les autres si la première n'est pas accessible. Les règles d'ordre d'adresse de destination de la section 6, lorsque elles sont appliquées à l'ensemble d'adresses retourné par le DNS, fournissent un tel ordre recommandé.

Le présent document spécifie séparément le choix d'adresse de source et le choix d'adresse de destination, mais en utilisant un contexte commun de sorte qu'ensemble les deux algorithmes donnent des résultats utiles. Les algorithmes essayent de choisir des adresses de source et de destination de portée et d'état de configuration approprié (préféré ou déconseillé au sens de la RFC2462). De plus, le présent document suggère une méthode préférée, du plus long préfixe correspondant, pour choisir parmi les adresses par ailleurs équivalentes en l'absence de meilleures informations. Le présent document spécifie aussi des astuces de politique pour permettre un outrepassement administratif du comportement par défaut. Par exemple, en utilisant ces astuces, un administrateur peut spécifier un préfixe de source préféré à utiliser avec un préfixe de destination, ou préférer des adresses de destination avec un préfixe à des adresses avec un autre préfixe. Ces astuces donnent à l'administrateur de la souplesse pour traiter certains scénarios de multi rattachement et de transition, mais ils ne sont certainement pas une panacée.

Les règles de choix spécifiées dans le présent document NE DOIVENT PAS être construites pour outrepasser le choix explicite d'une application ou d'une couche supérieure d'une adresse légale de destination ou de source.

1.1 Conventions utilisées dans ce document

Les mots clés "DOIT", "NE DOIT PAS", "EXIGE", "DEVRA", "NE DEVRA PAS", "DEVRAIT", "NE DEVRAIT PAS", "RECOMMANDÉ", "PEUT", et "FACULTATIF" dans ce document sont à interpréter comme décrit dans le BCP 14, [RFC2119].

2. Contexte du fonctionnement des algorithmes

Notre contexte pour les choix d'adresse découle de l'architecture de mise en œuvre la plus courante, qui sépare le choix de l'adresse de destination du choix de l'adresse de source. Par conséquent, nous avons deux algorithmes distincts pour ces tâches. Les algorithmes sont conçus pour bien fonctionner ensemble et ils partagent un mécanisme pour l'outrepassement de politique administrative.

Dans cette architecture de mise en œuvre, les applications utilisent des API [RFC2553] comme getaddrinfo() qui retournent une liste d'adresses à l'application. Cette liste peut contenir des adresses aussi bien IPv6 que IPv4 (parfois représentées par des adresses transposées en IPv4). L'application passe alors une adresse de destination à la pile réseau avec connect() ou sendto(). L'application va alors normalement essayer la première adresse de la liste, décrivant la liste d'adresses jusqu'à trouver une adresse qui convienne. Dans tous les cas, la couche réseau n'est jamais en situation d'avoir besoin de choisir une adresse de destination entre plusieurs solutions de remplacement. L'application peut aussi spécifier une adresse de source avec bind(), mais souvent l'adresse de source est laissée non spécifiée. Donc, la couche réseau choisit souvent une adresse de source entre plusieurs choix.

Par conséquent, il est prévu que les mises en œuvre de getaddrinfo() utilisent l'algorithme de choix d'adresse de destination spécifié ici pour trier la liste des adresses IPv6 et IPv4 qu'elles retournent. De son côté, la couche réseau IPv6 va utiliser l'algorithme de choix d'adresse de source lorsque une application ou couche supérieure n'a pas spécifié d'adresse de source. L'application de la présente spécification au choix d'adresse de source dans une couche réseau IPv4 est peut-être possible mais cela n'a pas été exploré plus avant ici.

Les applications qui se comportent bien DEVRAIENT itérer à travers la liste d'adresses retournée de getaddrinfo() jusqu'à ce qu'elles trouvent une adresse qui fonctionne.

Les algorithmes utilisent plusieurs critères pour prendre leurs décisions. L'effet combiné est de préférer les paires d'adresse destination/source pour lesquelles les deux adresses sont de portée ou type égal, de préférer les plus petites portées aux plus grandes pour l'adresse de destination, de préférer les adresses de source non déconseillées, d'éviter d'utiliser des adresses transitoires lorsque les adresses natives sont disponibles, et toutes choses égales par ailleurs, de préférer les paires d'adresse qui ont le plus long préfixe commun possible. Pour le choix d'adresse de source, les adresses publiques [RFC3041] sont préférées aux adresses temporaires. Dans les situations mobiles [RFC3775], les adresses de rattachement sont préférées aux adresses d'entretien. Si une adresse est simultanément une adresse de rattachement et une adresse d'entretien (ce qui indique que le nœud mobile est "a la maison" pour cette adresse) alors l'adresse de rattachement/d'entretien est préférée aux adresses qui sont seulement une adresse de rattachement ou seulement une adresse d'entretien.

La présente spécification permet en option la possibilité d'une configuration administrative de politique qui puisse outrepasser le comportement par défaut des algorithmes. L'outrepassement de politique prend la forme d'un tableau configurable qui spécifie les valeurs de préséance et les préfixes de source préférés pour les préfixes de destination. Si une mise en œuvre n'est pas configurable, ou si une mise en œuvre n'a pas été configurée, alors le tableau de politique par défaut spécifié dans le présent document DEVRAIT être utilisé.

2.1 Tableau de politique

Le tableau de politique est un tableau de recherche de la plus longue correspondance de préfixe, un peu comme un tableau d'acheminement. Étant donnée une adresse A, une recherche dans le tableau de politique produit deux valeurs : une valeur de préséance Precedence(A) et un classement ou étiquette Label(A). La valeur de préséance Precedence(A) est utilisée pour trier les adresses de destination. Si Precedence(A) > Precedence(B), on dit que l'adresse A a une plus forte préséance que l'adresse B, ce qui signifie que notre algorithme va préférer trier l'adresse de destination A avant l'adresse de destination B.

La valeur d'étiquette Label(A) permet des politiques qui préfèrent utiliser un préfixe particulier d'adresse de source avec un préfixe d'adresse de destination. Les algorithmes préfèrent utiliser l'adresse de source S avec une adresse de destination D si Label(S) = Label(D). Les mises en œuvre de IPv6 DEVRAIENT prendre en charge le choix d'adresse configurable via un mécanisme au moins aussi puissant que les tableaux de politique définis ici. Noter qu'au moment de la rédaction du présent document, il n'y a que peu d'expérience d'utilisation de politiques qui choisissent à partir d'un ensemble possible d'adresses IPv6. Lorsque on aura plus d'expérience, les politiques par défaut recommandées pourront changer. Par conséquent, il est important que les mises en œuvre fournissent un moyen de changer les politiques par défaut lorsque plus d'expérience aura été obtenue. Les paragraphes 10.3 et 10.4 donnent des exemples des sortes de changements qui pourraient être nécessaires.

Si une mise en œuvre n'est pas configurable ou n'a pas été configurée, elle DEVRAIT alors fonctionner conformément aux algorithmes spécifiés ici en conjonction avec le tableau de politique par défaut suivant :

Préfixe	Préséance	Étiquette
::1/128	50	0
::/0	40	1
2002::/16	30	2
::/96	20	3
::ffff:0:0/96	10	4

Un effet du tableau de politique par défaut est de préférer utiliser les adresses de source natives avec des adresses de destination natives, des adresses de source 6à4 [RFC3056] avec des adresses de destination 6à4, et des adresses de source compatibles v4 [RFC2373] avec des adresses de destination compatibles v4. Un autre effet du tableau de politique par défaut est de préférer la communication qui utilise des adresses IPv6 à la communication qui utilise des adresses IPv4, si les adresses de source correspondantes sont disponibles.

Les entrées de tableau de politique pour les préfixes d'adresse à portée limitée PEUVENT être qualifiées avec un indice de zone facultatif. Si il en est ainsi, une entrée de tableau de préfixes est seulement confrontée à une adresse durant une recherche si l'indice de zone correspond aussi à l'indice de zone de l'adresse.

2.2 Longueur de préfixe commun

On définit la longueur du préfixe commun CommonPrefixLen(A, B) de deux adresses A et B comme la longueur du plus long préfixe (en regardant les bits de plus fort poids, ou les plus à gauche) que les deux adresses ont en commun. Elle va de 0 à 128.

3. Propriétés d'adresse

Dans les règles données dans les paragraphes qui suivent, les adresses de différents types (par exemple, IPv4, IPv6, diffusion groupée et envoi individuel) sont comparées les unes par rapport aux autres. Certains de ces types d'adresse ont des propriétés qui ne sont pas directement comparables aux autres. Par exemple, les adresses IPv6 d'envoi individuel peuvent être "préférées" ou "déconseillées" [RFC2462], alors que les adresses IPv4 n'ont pas une telle notion. Les transpositions suivantes sont définies pour comparer de telles adresses en utilisant les règles de classement (par exemple, pour utiliser des adresses "préférées" de préférence à des adresses "déconseillées").

3.1 Comparaisons de portée

Les adresses de destination de diffusion groupée ont un champ de portée de quatre bits qui contrôle la propagation du paquet en diffusion groupée. L'architecture d'adressage IPv6 définit des valeurs du champ Portée pour les portées d'interface locale (0x1), de liaison locale (0x2), de sous-réseau local (0x3), d'administration locale (0x4), de site local (0x5), d'organisation locale (0x8), et mondiale (0xE) [RFC3513].

L'utilisation de l'algorithme de choix d'adresse de source en présence d'adresses de destination en diffusion groupée exige la comparaison de la portée d'une adresse d'envoi individuel avec la portée d'une adresse de diffusion groupée. On transpose une portée de liaison locale en envoi individuel en portée de liaison locale en diffusion groupée, de site local en envoi individuel en site local en diffusion groupée, et de portée mondiale en envoi individuel en portée mondiale en diffusion groupée. Par exemple, le site local en envoi individuel est égal au site local en diffusion groupée, qui est plus petit que l'organisation locale en diffusion groupée, qui est plus petit que le mondial en envoi individuel, qui est égal au mondial en diffusion groupée.

On écrit Scope(A) pour signifier la portée de l'adresse A. Par exemple, si A est une adresse d'envoi individuel de portée liaison locale et si B est une adresse de diffusion groupée de portée de site local, alors Scope(A) < Scope(B).

Cette transposition fusionne implicitement les frontières de site d'envoi individuel et les frontières de site de diffusion groupée [RFC3513]].

3.2 Adresses IPv4 et adresses transposées en IPv4

Le choix d'un algorithme d'adresse de destination fonctionne aussi bien sur les adresses IPv6 que IPv4. À cette fin, les adresses IPv4 devraient être représentée comme des adresses transposées en IPv4 [RFC2373]. Par exemple, pour rechercher la préséance sur d'autres attributs d'une adresse IPv4 dans le tableau de politique, pour rechercher l'adresse IPv6 correspondant à celle transposée en IPv4.

Les adresses IPv4 ont des portées qui sont allouées comme suit. Les adresses IPv4 auto-configurées [RFC3927], qui ont le préfixe 169.254/16, ont une portée allouée de liaison locale. Les adresses privées IPv4 [RFC1918], qui ont les préfixes 10/8, 172.16/12, et 192.168/16, ont une portée allouée de site local. Les adresses de bouclage IPv4 du paragraphe 4.2.2.11 de la [RFC1918], qui ont le préfixe 127/8, ont une portée allouée de liaison locale (de façon analogue au traitement de l'adresse de bouclage IPv6 de la section 4 de la [RFC3513]). Les autres adresses IPv4 ont une portée allouée mondiale.

Les adresses IPv4 devraient être traitées comme ayant l'état de configuration "préféré" (au sens de la RFC2462).

3.3 Autres adresses IPv6 avec adresses IPv4 incorporées

Les adresses compatibles IPv4 [RFC2373], transposées IPv4 [RFC2373], traduisibles en IPv4 [RFC2765] et les adresses 6à4 [RFC3056] contiennent une adresse IPv4 incorporée. Pour les besoins du présent document, ces adresses devraient être traitées comme ayant une portée mondiale.

Les adresses compatibles IPv4, transposées en IPv4, et traduisibles en IPv4 devraient être traitées comme ayant le statut de configuration de "préféré" (au sens de la RFC2462).

3.4 Adresses de bouclage IPv6 et autres préfixes de format

L'adresse de bouclage devrait être traitée comme ayant une portée de liaison locale [RFC3513, section 4] et l'état de configuration "préféré" (au sens de la RFC2462).

Les adresses NSAP et autres adresses avec des préfixes de format encore indéfini devraient être traitées comme ayant une portée mondiale et un état de configuration de "préféré" (au sens de la RFC2462). Des normes ultérieures pourront outrepasser ce traitement.

3.5 Adresses de mobilité

Certains nœuds peuvent prendre en charge la mobilité en utilisant les concepts d'adresse de rattachement et d'adresse d'entretien (par exemple, voir la [RFC3775]). Du point de vue conceptuel, une adresse de rattachement est une adresse IP allouée à un nœud mobile et utilisée comme adresse permanente du nœud mobile. Une adresse d'entretien est une adresse IP associée à un nœud mobile lorsque il visite une liaison étrangère. Lorsque un nœud mobile est sur sa liaison de rattachement, il peut avoir une adresse qui soit simultanément une adresse de rattachement et une adresse d'entretien.

Pour les besoins du présent document, il est suffisant de savoir si ses propres adresses sont ou non conçues comme des adresses de rattachement ou des adresses d'entretien. La question de savoir si une adresse devrait être vue comme une adresse de rattachement ou une adresse d'entretien sort du domaine d'application du présent document.

4. Adresses de source candidates

L'algorithme de choix d'adresse de source utilise le concept d'un "ensemble candidat" d'adresses de source potentielles pour une certaine adresse de destination. L'ensemble candidat est l'ensemble de toutes les adresses qui pourraient être utilisées comme adresse de source ; l'algorithme de choix d'adresse de source va prendre une adresse dans cet ensemble. On écrit CandidateSource(A) pour noter l'ensemble candidat pour l'adresse A.

Il est RECOMMANDÉ que les adresses de source candidates soient l'ensemble des adresses d'envoi individuel allouées à l'interface qui sera utilisé pour envoyer à la destination (l'interface "sortante"). Sur les routeurs, l'ensemble candidat PEUT inclure les adresses d'envoi individuel allouées à tout interface qui transmet les paquets, sous réserve des restrictions décrites ci-dessous.

Discussion : Le mécanisme Redirect de la découverte de voisin de la [RFC2461] exige que les routeurs vérifient que l'adresse de source d'un paquet identifie un voisin avant de générer un Redirect, de sorte qu'il est avantageux pour les hôtes de choisir des adresses de source allouées à l'interface sortante. Les mises en œuvre qui souhaitent prendre en charge l'utilisation d'adresses de source mondiales allouées à une interface de bouclage devraient se comporter comme si l'interface de bouclage générait et transmettait le paquet.

Dans certains cas, l'adresse de destination peut être qualifiée avec un indice de zone ou d'autres informations qui vont contraindre l'ensemble candidat.

Pour les adresses de destination de diffusion groupée et de liaison locale, l'ensemble des adresses de source candidates DOIT seulement inclure des adresses allouées aux interfaces qui appartiennent à la même liaison que l'interface sortante.

Discussion : La restriction aux adresses de destination en diffusion groupée est nécessaire parce que les algorithmes de transmission de diffusion groupée actuellement déployés utilisent des vérifications de transmission sur le chemin inverse (RPF, *Reverse Path Forwarding*).

Pour les adresses de destination de site local, l'ensemble des adresses de source candidates DOIT inclure seulement les

adresses allouées aux interfaces qui appartiennent au même site que l'interface sortante.

En aucun cas, les adresses d'envoi à la cantonade, les adresses de diffusion groupée, et l'adresse non spécifiée NE DOIVENT être incluses dans un ensemble candidat.

Si une application ou couche supérieure spécifie une adresse de source qui n'est pas dans l'ensemble candidat pour la destination, la couche réseau DOIT alors traiter cela comme une erreur. L'adresse de source spécifiée peut influencer l'ensemble candidat en affectant le choix de l'interface sortante. Si l'application ou couche supérieure spécifie une adresse de source qui est dans l'ensemble candidat pour la destination, la couche réseau DOIT alors respecter ce choix. Si l'application ou couche supérieure ne spécifie pas d'adresse de source, la couche réseau utilise alors l'algorithme de choix d'adresse de source spécifié au paragraphe suivant.

Sur les nœuds IPv6 seul qui prennent en charge SIIT [RFC2765, en particulier la section 5], si l'adresse de destination est une adresse transposée en IPv4, l'ensemble candidat DOIT alors ne contenir que des adresses traduisibles en IPv4. Si l'adresse de destination n'est pas une adresse transposée en IPv4, l'ensemble candidat NE DOIT alors PAS contenir d'adresse traduisible en IPv4.

5. Choix d'adresse de source

L'algorithme de choix d'adresse de source produit comme résultat une seule adresse de source à utiliser avec une certaine adresse de destination. Cet algorithme ne s'applique qu'aux adresses de destination IPv6, pas aux adresses IPv4.

L'algorithme est spécifié ici en termes de liste de règles de comparaison de paires qui (pour une certaine adresse de destination D) imposent un ordre de "supérieur à" aux adresses de l'ensemble candidat CandidateSource(D). L'adresse en tête de la liste à l'achèvement de l'algorithme est celle qu'il a retenue.

Noter que conceptuellement, un tri de l'ensemble de candidats est effectué dans lequel un ensemble de règles définit l'ordre des adresses. Mais comme le résultat de l'algorithme est une seule adresse de source, une mise en œuvre n'a pas besoin en réalité de trier l'ensemble ; elle a seulement besoin d'identifier la valeur "maximum" qui termine en tête de la liste triée.

L'ordre des adresses dans l'ensemble candidat est défini par une liste de huit règles de comparaison par paires, chaque règle plaçant un ordre "supérieur à", "inférieur à", ou "égal à", sur deux adresses de source l'une par rapport à l'autre (et par rapport à cette règle). Dans le cas où une certaine règle produit une égalité, c'est-à-dire donne un résultat "égal à" pour les deux adresses, les règles restantes sont appliquées (dans l'ordre) pour les seules adresses qui sont à égalité pour les départager. Noter que si une règle produit un seul clair "vainqueur" (ou ensemble de "vainqueurs" dans le cas d'égalité) les adresses qui ne sont pas dans l'ensemble vainqueur peuvent être éliminées des calculs suivants, les règles suivantes n'étant appliquées qu'aux adresses restantes. Si les huit règles échouent à choisir une seule adresse, un moyen de départage non spécifié devrait être utilisé.

Quand on compare deux adresses SA et SB de l'ensemble candidat, on dit "préférer SA" pour signifier que SA est "supérieur à" SB, et de façon similaire, on dit "préférer SB" pour signifier que SA est "inférieur à" SB.

Règle 1 : Préférer la même adresse.
Si SA = D, alors préférer SA. De même, si SB = D, alors préférer SB.

Règle 2 : Préférer la portée appropriée.
Si Scope(SA) < Scope(SB) : Si Scope(SA) < Scope(D), alors préférer SB et autrement préférer SA. De même, si Scope(SB) < Scope(SA) : Si Scope(SB) < Scope(D), alors préférer SA et autrement préférer SB.

Règle 3 : Éviter les adresses déconseillées.
Les adresses SA et SB ont la même portée. Si une des deux adresses de source est "préférée" et si l'une d'elles est "déconseillée" (au sens de la RFC2462) préférer alors celle qui est "préférée".

Règle 4 : Préférer les adresses de rattachement.
Si SA est simultanément une adresse de rattachement et une adresse d'entretien et si SB ne l'est pas, préférer SA. De même, si SB est simultanément une adresse de rattachement et une adresse d'entretien et si SA ne l'est pas, alors, préférer SB. Si SA est juste une adresse de rattachement et si SB est juste une adresse d'entretien, alors préférer SA. De même, si SB est juste une adresse de rattachement et si SA est juste une adresse d'entretien, alors préférer SB.

Les mises en œuvre devraient fournir un mécanisme qui permette à une application d'inverser le sens de cette préférence et de préférer les adresses d'entretien plutôt que les adresses de rattachement (par exemple, via des extensions d'API

appropriées). L'utilisation de ce mécanisme ne devrait affecter que les règles de choix pour l'application invocatrice.

Règle 5 : Préférer l'interface sortante.
Si SA est alloué à l'interface qui sera utilisée pour envoyer à D et si SB est alloué à une interface différente, alors préférer SA. De même, si SB est alloué à l'interface qui sera utilisée pour envoyer à D et si SA est alloué à une interface différente, alors préférer SB.

Règle 6 : Préférer l'étiquette qui correspond.
Si Label(SA) = Label(D) et si Label(SB) <> Label(D), alors préférer SA. De même, si Label(SB) = Label(D) et si Label(SA) <> Label(D), alors préférer SB.

Règle 7 : Préférer les adresses publiques.
Si SA est une adresse publique et si SB est une adresse temporaire, alors préférer SA. De même, si SB est une adresse publique et si SA est une adresse temporaire, alors préférer SB.

Les mises en œuvre DOIVENT fournir un mécanisme permettant à une application d'inverser le sens de cette préférence et de préférer les adresses temporaires aux adresses publiques (par exemple, via les extensions d'API appropriées). L'utilisation du mécanisme devrait n'affecter que les règles de choix pour l'application invocatrice. Cette règle évite de possibles échecs des applications du fait de la durée de vie relativement courte des adresses temporaires ou à cause de la possibilité que la recherche inverse d'une adresse temporaire n'échoue ou retourne un nom aléatoire. Les mises en œuvre pour lesquelles les considérations de confidentialité l'emportent sur les soucis de compatibilité de ces applications PEUVENT inverser le sens de cette règle et préférer par défaut les adresses temporaires aux adresses publiques.

Règle 8 : Utiliser le préfixe qui a la plus longue correspondance.
Si CommonPrefixLen(SA, D) > CommonPrefixLen(SB, D), alors préférer SA. De même, si CommonPrefixLen(SB, D) > CommonPrefixLen(SA, D), alors préférer SB.

La règle 8 peut être outrepassée si la mise en œuvre a d'autres moyens de choisir entre les adresses de source. Par exemple, si la mise en œuvre sait plus ou moins quelle adresse de source va résulter en les "meilleures" performances de communication.

La règle 2 (préférer la portée appropriée) DOIT être mise en œuvre et recevoir une priorité élevée parce qu'elle peut affecter l'interopérabilité.

6. Choix d'adresse de destination

L'algorithme de choix d'adresse de destination prend une liste d'adresses de destination et trie les adresses pour produire une nouvelle liste. Il est spécifié ici sous la forme de la comparaison d'une paire d'adresses DA et DB, où DA apparaît avant DB dans la liste originale.

L'algorithme trie ensemble les adresses IPv6 et IPv4. Pour trouver les attributs d'une adresse IPv4 dans le tableau de politique, l'adresse IPv4 devrait être représentée comme une adresse transposée en IPv4.

On écrit Source(D) pour indiquer l'adresse de source choisie pour une destination D. Pour les adresses IPv6, le paragraphe précédent spécifie l'algorithme de choix d'adresse de source. Le choix de l'adresse de source pour les adresses IPv4 n'est pas spécifié dans le présent document.

On dit que Source(D) est indéfini si aucune adresse de source n'est disponible pour la destination D. Pour les adresses IPv6, ce n'est le cas que si CandidateSource(D) est l'ensemble vide.

La comparaison par paires des adresses de destination consiste en dix règles, qui devraient être appliquées dans l'ordre. Si une règle détermine un résultat, alors les règles restantes ne sont pas pertinentes et devraient être ignorées. Les règles suivantes agissent comme départage pour les règles précédentes. Voir à la section précédante une plus longue description de la façon dont les règles de départage de comparaison par paires peuvent être utilisées pour trier une liste.

Règle 1 : Éviter les destinations inutilisables.
Si DB est connu pour être injoignable ou si Source(DB) est indéfini, alors préférer DA. De même, si DA est connu pour être injoignable ou si Source(DA) est indéfini, alors préférer DB.

Discussion : Une mise en œuvre peut savoir qu'une certaine destination est injoignable de plusieurs façons. Par exemple, la destination peut être atteinte à travers une interface réseau qui se trouve actuellement débranchée. Par

exemple, la mise en œuvre peut conserver pendant un certain temps des informations provenant de la détection d'inaccessibilité de voisin [RFC2461]. Dans tous les cas, la détermination d'inaccessibilité pour les besoins de cette règle dépend de la mise en œuvre.

Règle 2 : Préférer la portée qui correspond.
Si Scope(DA) = Scope(Source(DA)) et si Scope(DB) <> Scope(Source(DB)), alors préférer DA. De même, si Scope(DA) <> Scope(Source(DA)) et si Scope(DB) = Scope(Source(DB)), alors préférer DB.

Règle 3 : Éviter les adresses déconseillées.Si Source(DA) est déconseillée et si Source(DB) ne l'est pas, alors préférer DB. De même, si Source(DA) n'est pas déconseillée et si Source(DB) est déconseillé, alors préférer DA.

Règle 4 : Préférer les adresses de rattachement.
Si Source(DA) est simultanément une adresse de rattachement et une adresse d'entretien et si Source(DB) ne l'est pas, alors préférer DA. De même, si Source(DB) est simultanément une adresse de rattachement et une adresse d'entretien et si Source(DA) ne l'est pas, alors préférer DB.

Si Source(DA) est juste une adresse de rattachement et si Source(DB) est juste une adresse d'entretien, alors préférer DA. De même, si Source(DA) est juste une adresse d'entretien et si Source(DB) est juste une adresse de rattachement, alors préférer DB.

Règle 5 : Préférer l'étiquette qui correspond.
Si Label(Source(DA)) = Label(DA) et si Label(Source(DB)) <> Label(DB), alors préférer DA. De même, si Label(Source(DA)) <> Label(DA) et si Label(Source(DB)) = Label(DB), alors préférer DB.

Règle 6 : Préférer la plus fore préséance.
Si Precedence(DA) > Precedence(DB), alors préférer DA. De même, si Precedence(DA) < Precedence(DB), alors préférer DB.

Règle 7 : Préférer le transport natif.
Si DA est atteint via un mécanisme de transition encapsulant (par exemple, IPv6 dans IPv4) et si DB ne l'est pas, alors préférer DB. De même, si DB est atteint via encapsulation et si DA ne l'est pas, alors préférer DA.

Discussion : 6-sur-4 [RFC2529], ISATAP [RFC4214], et tunnels configurés [RFC1933] sont des exemples de mécanismes de transition encapsulants pour lesquels l'adresse de destination n'a pas de préfixe spécifique et donc ne peut recevoir une préséance inférieure dans le tableau de politique. Une mise en œuvre PEUT généraliser cette règle en utilisant un concept de préférence d'interface, et en donnant à des interfaces virtuelles (comme les interfaces encapsulant IPv6 dans IPv4) une préférence inférieure à celle des interfaces natives (comme les interfaces ethernet).

Règle 8 : Préférer la plus petite portée.
Si Scope(DA) < Scope(DB), alors préférer DA. De même, si Scope(DA) > Scope(DB), alors préférer DB.

Règle 9 : Utiliser le préfixe qui a la plus longue correspondance.
Lorsque DA et DB appartiennent à la même famille d'adresses (tous deux sont IPv6 ou tous deux sont IPv4) : si CommonPrefixLen(DA, Source(DA)) > CommonPrefixLen(DB, Source(DB)), alors préférer DA. De même, si CommonPrefixLen(DA, Source(DA)) < CommonPrefixLen(DB, Source(DB)), alors préférer DB.

Règle 10 : Autrement, laisser l'ordre inchangé.
Si DA précédait DB dans la liste d'origine, préférer DA. Autrement, préférer DB.

Les règles 9 et 10 peuvent être outrepassées si la mise en œuvre a d'autres moyens de trier les adresses de destination. Par exemple, si la mise en œuvre sait d'une façon ou d'une autre quelles adresses de destination vont donner les "meilleures" performances de communications.

7. Interactions avec l'acheminement

La présente spécification de choix d'adresse de source suppose que l'acheminement (plus précisément, le choix d'une interface sortante sur un nœud qui a plusieurs interfaces) est fait avant le choix de l'adresse de source. Cependant, les mises en œuvre peuvent utiliser des considérations d'adresse de source pour le départage lors du choix entre des chemins par ailleurs équivalents.

Par exemple, supposons qu'un nœud a des interfaces sur deux liaisons différentes, les deux liaisons ayant un routeur de fonctionnement par défaut. Les deux interfaces ont des adresses mondiales préférées (au sens de la RFC2462). Lorsque on envoie à une adresse de destination mondiale, si il n'y a pas de raison liée à l'acheminement de préférer une interface à l'autre, une mise en œuvre peut alors choisir de préférence l'interface sortante qui va lui permettre d'utiliser l'adresse de source qui partage un plus long préfixe commun avec la destination.

Les mises en œuvre peuvent aussi utiliser le choix du routeur pour influencer le choix de l'adresse de source. Par exemple, supposons un hôte sur une liaison avec deux routeurs. Un routeur annonce un préfixe mondial A et l'autre routeur annonce un préfixe mondial B. Lorsque il envoie via le premier routeur, l'hôte peut préférer les adresses de source avec le préfixe A et lorsque il envoie via le second routeur, préférer les adresses de source avec le préfixe B.

8. Considérations de mise en œuvre

L'algorithme de choix d'adresse de destination a besoin d'informations sur les adresses de source potentielles. Une stratégie possible de mise en œuvre est que getaddrinfo() appelle la couche réseau avec une liste d'adresses de destination, trie la liste dans la couche réseau avec la pleine connaissance actuelle des adresses de source disponibles, et retourne la liste triée à getaddrinfo(). C'est simple et donne le meilleur résultat mais introduit la surcharge de l'appel à un autre système. Une façon de réduire cette surcharge est de mettre en antémémoire la liste triée des adresses dans le résolveur, de sorte que les appels suivants pour le même nom n'aient pas besoin de retrier la liste.

Une autre stratégie de mise en œuvre est de faire appel à la couche réseau pour restituer les informations d'adresse de source et de trier alors la liste des adresses directement dans le contexte de getaddrinfo(). Pour réduire la surcharge dans cette approche, les informations d'adresse de source peuvent être mises en antémémoire, ce qui amortit la surcharge de la restitution sur plusieurs appels à getaddrinfo(). Dans cette approche, la mise en œuvre peut n'avoir pas connaissance de l'interface sortante pour chaque destination, de sorte qu'elle PEUT utiliser une définition plus lâche de l'ensemble candidat durant le rangement des adresses de destination.

Dans tous les cas, si la mise en œuvre utilise des informations mises en antémémoire et éventuellement périmées dans son application de choix d'adresse de destination, ou si le rangement d'une liste d'adresses de destination en antémémoire pourrait être périmé, alors elle devrait s'assurer que l'ordre des adresses de destination retourné à l'application n'est pas dépassé depuis plus d'une seconde. Par exemple, une mise en œuvre peut faire un appel système pour vérifier si des entrées de tableau d'acheminement ou des allocations d'adresses de source qui pourraient affecter ces algorithmes ont changé. Une autre stratégie est d'utiliser un compteur d'invalidation qui est incrémenté chaque fois qu'un état sous-jacent a changé. En mettant en antémémoire la valeur actuelle du compteur d'invalidation avec l'état déduite et en comparant ultérieurement à la valeur du moment, la mise en œuvre pourra détecter si l'état déduit est potentiellement périmé.

9. Considérations pour la sécurité

Le présent document n'a pas d'impact direct sur la sécurité de l'infrastructure de l'Internet.

Noter que la plupart des algorithmes de choix d'adresse de source, y compris celui spécifié dans le présent document, exposent à un souci potentiel de confidentialité. Un nœud hostile peut déduire des corrélations entre les adresses d'un nœud cible en sondant le nœud cible avec des paquets de demande qui forcent l'hôte cible à choisir son adresse de source pour les paquets de réponse. (Peut-être parce que les paquets de demande sont envoyés à une adresse de diffusion groupée ou d'envoi à la cantonade, ou peut-être parce que le protocole de couche supérieure choisi pour l'attaque ne spécifie pas une adresse de source particulière pour ses paquets de réponse.) En utilisant des adresses différentes pour lui-même, le nœud hostile peut causer l'exposition des propres adresses du nœud cible.

10. Exemples

La présente section contient un certain nombre d'exemples, d'abord de comportement par défaut et ensuite, démontrant l'utilité de la configuration d'un tableau de politiques. Ces exemples ne sont fournis qu'à titre d'illustration ; ils ne devraient pas être considérés comme normatifs.

10.1 Choix d'adresse de source par défaut

Les règles de choix d'adresse de source, en conjonction avec le tableau de politique par défaut, produisent le comportement suivant :

Destination: 2001::1
Adresses sources candidates : 3ffe::1 ou fe80::1
Résultat : 3ffe::1 (préférer la portée appropriée)

Destination: 2001::1
Adresses sources candidates : fe80::1 ou fec0::1
Résultat : fec0::1 (préférer la portée appropriée)

Destination: fec0::1
Adresses sources candidates : fe80::1 ou 2001::1
Résultat : 2001::1 (préférer la portée appropriée)

Destination : ff05::1
Adresses sources candidates : fe80::1 ou fec0::1 ou 2001::1Résultat : fec0::1 (préférer la portée appropriée)

Destination : 2001::1
Adresses sources candidates : 2001::1 (déconseillée) ou 2002::1
Résultat : 2001::1 (préférer la même adresse)

Destination : fec0::1
Adresses sources candidates : fec0::2 (déconseillée) ou 2001::1
Résultat : fec0::2 (préférer la portée appropriée)

Destination : 2001::1
Adresses sources candidates : 2001::2 ou 3ffe::2
Résultat : 2001::2 (plus long préfixe correspondant)

Destination : 2001::1
Adresses sources candidates : 2001::2 (adresse d'entretien) ou 3ffe::2 (adresse de rattachement)
Résultat : 3ffe::2 (préférer l'adresse de rattachement)
Destination : 2002:836b:2179::1
Adresses sources candidates : 2002:836b:2179::d5e3:7953:13eb:22e8 (temporaire) ou 2001::2
Résultat : 2002:836b:2179::d5e3:7953:13eb:22e8 (préférer l'étiquette correspondante)

Destination : 2001::d5e3:0:0:1
Adresses sources candidates : 2001::2 ou 2001::d5e3:7953:13eb:22e8 (temporaire)
Résultat : 2001::2 (préférer l'adresse publique)

10.2 Choix d'adresse de destination par défaut

Les règles de choix d'adresse de destination, en conjonction avec le tableau de politique par défaut et les règles de choix d'adresse de source, produisent le comportement suivant :

Adresses sources candidates : 2001::2 ou fe80::1 ou 169.254.13.78
Liste d'adresses de destination : 2001::1 ou 131.107.65.121
Résultat : 2001::1 (src 2001::2) puis 131.107.65.121 (src 169.254.13.78) (préférer la portée qui correspond)

Adresses sources candidates : fe80::1 ou 131.107.65.117
Liste d'adresses de destination : 2001::1 ou 131.107.65.121
Résultat : 131.107.65.121 (src 131.107.65.117) puis 2001::1 (src fe80::1) (préférer la portée qui correspond)

Adresses sources candidates : 2001::2 ou fe80::1 ou 10.1.2.4
Liste d'adresses de destination : 2001::1 ou 10.1.2.3
Résultat : 2001::1 (src 2001::2) puis 10.1.2.3 (src 10.1.2.4) (préférer la préséance la plus élevée)

Adresses sources candidates : 2001::2 ou fec0::2 ou fe80::2
Liste d'adresses de destination : 2001::1 ou fec0::1 ou fe80::1
Résultat : fe80::1 (src fe80::2) puis fec0::1 (src fec0::2) puis 2001::1 (src 2001::2) (préférer la plus petite portée)

Adresses sources candidates : 2001::2 (adresse d'entretien) ou 3ffe::1 (adresse de rattachement) ou fec0::2 (adresse d'entretien) ou fe80::2 (adresse d'entretien)

Liste d'adresses de destination : 2001::1 ou fec0::1
Résultat : 2001:1 (src 3ffe::1) puis fec0::1 (src fec0::2) (préférer l'adresse de rattachement)

Adresses sources candidates : 2001::2 ou fec0::2 (déconseillée) ou fe80::2
Liste d'adresses de destination : 2001::1 ou fec0::1
Résultat : 2001::1 (src 2001::2) puis fec0::1 (src fec0::2) (éviter les adresses déconseillées)

Adresses sources candidates : 2001::2 ou 3f44::2 ou fe80::2
Liste d'adresses de destination : 2001::1 ou 3ffe::1
Résultat : 2001::1 (src 2001::2) alors 3ffe::1 (src 3f44::2) (plus long préfixe correspondant)

Adresses sources candidates : 2002:836b:4179::2 ou fe80::2
Liste d'adresses de destination : 2002:836b:4179::1 ou 2001::1
Résultat : 2002:836b:4179::1 (src 2002:836b:4179::2) puis 2001::1 (src 2002:836b:4179::2) (préférer l'étiquette correspondante)

Adresses sources candidates : 2002:836b:4179::2 ou 2001::2 ou fe80::2
Liste d'adresses de destination : 2002:836b:4179::1 ou 2001::1
Résultat : 2001::1 (src 2001::2) puis 2002:836b:4179::1 (src 2002:836b:4179::2) (préférer la préséance la plus élevée)

10.3 Configuration de la préférence pour IPv6 ou IPv4

Le tableau de politique par défaut donne aux adresses IPv6 une préséance plus élevée qu'aux adresses IPv4. Cela signifie que les applications vont utiliser IPv6 de préférence à IPv4 quand les deux sont également utilisables. Un administrateur peut changer le tableau de politique pour préférer les adresses IPv4 en donnant au préfixe ::ffff:0.0.0.0/96 une préséance plus élevée :

Préfixe	**Préséance**	**Étiquette**
::1/128	50	0
::/0	40	1
2002::/16	30	2
::/96	20	3
::ffff:0:0/96	100	4

Ce changement du tableau de politique par défaut produit le comportement suivant :

Adresses sources candidates : 2001::2 ou fe80::1 ou 169.254.13.78
Liste d'adresses de destination : 2001::1 ou 131.107.65.121
Résultat inchangé : 2001::1 (src 2001::2) puis 131.107.65.121 (src 169.254.13.78) (préférer la portée qui correspond)

Adresses sources candidates : fe80::1 ou 131.107.65.117
Liste d'adresses de destination : 2001::1 ou 131.107.65.121
Résultat inchangé : 131.107.65.121 (src 131.107.65.117) puis 2001::1 (src fe80::1) (préférer la portée qui correspond)

Adresses sources candidates : 2001::2 ou fe80::1 ou 10.1.2.4
Liste d'adresses de destination : 2001::1 ou 10.1.2.3
Nouveau résultat : 10.1.2.3 (src 10.1.2.4) puis 2001::1 (src 2001::2) (préférer la préséance la plus élevée)

10.4 Configuration de la préférence pour des adresses à portée limitée

Les règles de choix d'adresse de destination donnent la préférence aux destinations de plus petite portée. Par exemple, une destination de site local sera triée avant une destination de portée mondiale lorsque les deux sont par ailleurs également convenables. Un administrateur peut changer le tableau de politique pour inverser cette préférence et trier les destinations mondiales avant les destinations de site local, et les destinations de site local avant les destinations de liaison locale :

Préfixe	Préséance	Étiquette
::1/128	50	0
::/0	40	1
fec0::/10	37	1
fe80::/10	33	1
2002::/16	30	2
::/96	20	3
::ffff:0:0/96	10	4

Ce changement au tableau de politique par défaut produit le comportement suivant :

Adresses sources candidates : 2001::2 ou fec0::2 ou fe80::2
Liste d'adresses de destination : 2001::1 ou fec0::1 ou fe80::1
Nouveau résultat : 2001::1 (src 2001::2) puis fec0::1 (src fec0::2) puis fe80::1 (src fe80::2) (préférer la plus forte préséance)

Adresses sources candidates : 2001::2 (déconseillée) ou fec0::2 ou fe80::2
Liste d'adresses de destination : 2001::1 ou fec0::1
Résultat inchangé : fec0::1 (src fec0::2) puis 2001::1 (src 2001::2) (éviter les adresses déconseillées)

10.5 Configuration d'un site multi rattachements

Considérons un site A qui a une relation d'affaires importante avec un autre site B. Pour la prise en charge de leurs besoins d'affaires, les deux sites ont contracté un service avec des performances très élevées auprès d'un FAI. C'est en plus de la connexion Internet normale que les deux sites ont avec des FAI différents. Le FAI à hautes performances est coûteux et les deux sites souhaite ne l'utiliser que pour leur trafic d'affaire le plus important entre eux.

Chaque site a deux préfixes mondiaux, un du FAI à hautes performances et un avec le FAI normal. Le site A a le préfixe 2001:aaaa:aaaa::/48 avec le FAI à hautes performances et le préfixe 2007:0:aaaa::/48 avec son FAI normal. Le site B a le préfixe 2001:bbbb:bbbb::/48 avec le FAI à hautes performances et le préfixe 2007:0:bbbb::/48 avec son FAI normal. Tous les hôtes dans les deux sites ont deux adresses enregistrées dans le DNS.

L'acheminement au sein des deux sites dirige la plus grande partie du trafic sur la sortie du FAI normal, mais l'acheminement dirige le trafic envoyé au préfixe 2001 de l'autre site sur la sortie pour le FAI hautes performances. Pour empêcher l'utilisation involontaire de leur connexion avec le FAI hautes performances, les deux sites mettent en œuvre un filtrage d'entrée pour éliminer le trafic entrant en provenance du FAI hautes performances qui ne vient pas de l'autre site.

Le tableau de politique par défaut et les règles de sélection d'adresses produisent le comportement suivant :

Adresses sources candidates : 2001:aaaa:aaaa::a ou 2007:0:aaaa::a ou fe80::a
Liste d'adresses de destination : 2001:bbbb:bbbb::b ou 2007:0:bbbb::b
Résultat : 2007:0:bbbb::b (src 2007:0:aaaa::a) puis 2001:bbbb:bbbb::b (src 2001:aaaa:aaaa::a) (préfixe à plus longue correspondance)

En d'autre termes, lorsque un hôte du site A initie une connexion avec un hôte du site B, le trafic ne tire pas parti de leurs connexions au FAI hautes performances. Ce n'est pas le comportement qu'ils désirent.

Adresses sources candidates : 2001:aaaa:aaaa::a ou 2007:0:aaaa::a ou fe80::a
Liste d'adresses de destination : 2001:cccc:cccc::c ou 2006:cccc:cccc::c
Résultat : 2001:cccc:cccc::c (src 2001:aaaa:aaaa::a) puis 2006:cccc:cccc::c (src 2007:0:aaaa::a) (plus longue correspondance de préfixe)

En d'autres termes, lorsque un hôte du site A initie une connexion avec un hôte d'un autre site C, le trafic inverse peut revenir par le FAI à hautes performances. Là encore, ce comportement n'est pas désiré.

Cette situation difficile démontre les limitations de l'heuristique de la plus longue correspondance de préfixe dans les situations de rattachement multiple.

Cependant, les administrateurs des sites A et B peuvent réaliser leur comportement désiré par la configuration du tableau de politique. Par exemple, ils peuvent utiliser le tableau de politique suivant :

Préfixe	Préséance	Étiquette
::1	50	0
2001:aaaa:aaaa::/48	45	5
2001:bbbb:bbbb::/48	45	5
::/0	40	1
2002::/16	30	2
::/96	20	3
::ffff:0:0/96	10	4

Ce tableau de politique produit le comportement suivant :

Adresses sources candidates : 2001:aaaa:aaaa::a ou 2007:0:aaaa::a ou fe80::a
Liste d'adresses de destination : 2001:bbbb:bbbb::b ou 2007:0:bbbb::b
Nouveau résultat : 2001:bbbb:bbbb::b (src 2001:aaaa:aaaa::a) puis 2007:0:bbbb::b (src 2007:0:aaaa::a) (préférer la préséance la plus élevée)

En d'autres termes, lorsque un hôte du site A initie une connexion avec un hôte du site B, le trafic utilise le FAI à hautes performances comme désiré.

Adresses sources candidates : 2001:aaaa:aaaa::a ou 2007:0:aaaa::a ou fe80::a
Liste d'adresses de destination : 2001:cccc:cccc::c ou 2006:cccc:cccc::c
Nouveau résultat : 2006:cccc:cccc::c (src 2007:0:aaaa::a) puis 2001:cccc:cccc::c (src 2007:0:aaaa::a) (plus long préfixe correspondant)

En d'autres termes, lorsque un hôte du site A initie une connexion avec un hôte d'un autre site C, le trafic utilise le FAI normal comme désiré.

Références

[RFC1812] F. Baker, "Exigences pour les routeurs IP version 4", juin 1995. *(Mise à jour par la RFC2644)*

[RFC1918] Y. Rekhter et autres, "Allocation d'adresse pour les internets privés", BCP 5, février 1996.

[RFC1933] R. Gilligan, E. Nordmark, "Mécanismes de transition pour hôtes et routeurs IPv6", avril 1996. (*Obsolète, voir* RFC4213) *(P.S.)*

[RFC2026] S. Bradner, "Le processus de normalisation de l'Internet -- Révision 3", (BCP0009) octobre 1996. *(Remplace* RFC1602, RFC1871) (*MàJ par* RFC3667, RFC3668, RFC3932, RFC3979, RFC3978, RFC5378)

[RFC2119] S. Bradner, "Mots clés à utiliser dans les RFC pour indiquer les niveaux d'exigence", BCP 14, mars 1997.

[RFC2373] R. Hinden, S. Deering, "Architecture d'adressage IP version 6", juillet 1998. (*Obsolète, voir* RFC4291) *(P.S.)*

[RFC2461] T. Narten, E. Nordmark, W. Simpson, "Découverte de voisins pour IP version 6 (IPv6)", décembre 1998. (*Obsolète, voir* RFC4861) *(D.S.)*

[RFC2462] S. Thomson, T. Narten, "Autoconfiguration d'adresse IPv6 sans état", décembre 1998. (*Obsolète, voir* RFC4862) *(D.S.)*

[RFC2529] B. Carpenter, C. Jung, "Transmission d'IPv6 sur des domaines IPv4 sans tunnels explicites", mars 1999. *(P.S.)*

[RFC2553] R. Gilligan, S. Thomson, J. Bound, W. Stevens, "Extensions de base d'interface de prise pour IPv6", mars 1999. (*Obsolète, voir* RFC3493) (*MàJ par* RFC3152) *(Information)*

[RFC2765] E. Nordmark, "Algorithme de traduction IP/ICMP sans état (SIIT)", février 2000. *(P.S.)*

[RFC3041] T. Narten, R. Draves, "Extensions de confidentialité pour l'auto-configuration d'adresse sans état dans IPv6", janvier 2001. (*Obsolète, voir* RFC4941) *(P.S.)*

[RFC3056] B. Carpenter, K. Moore, "Connexion des domaines IPv6 via des nuages IPv4", février 2001. *(P.S.)*

[RFC3513] R. Hinden et S. Deering, "Architecture d'adressage du protocole Internet version 6 (IPv6)", avril 2003. *(Obsolète, remplacée par la RFC4291)*

[RFC3775] D. Johnson, C. Perkins, J. Arkko, "Prise en charge de la mobilité dans IPv6", juin 2004. *(P.S., Obs. voir* RFC6275*)*

[RFC3927] S. Cheshire, B. Aboba, E. Guttman, "Configuration dynamique des adresses IPv4 de liaison locale", mai 2005. *(P.S.)*

[RFC4214] F. Templin et aures, "Protocole d'adressage en tunnel automatique intra-site (ISATAP)", octobre 2005. (*Obsolète, voir* RFC5214, *Information)*

Remerciements

L’auteur tient à remercier de leurs contributions le groupe de travail IPng et en particulier Marc Blanchet, Brian Carpenter, Matt Crawford, Alain Durand, Steve Deering, Robert Elz, Jun-ichiro itojun Hagino, Tony Hain, M.T. Hollinger, Jinmei Tatuya, Thomas Narten, Erik Nordmark, Ken Powell, Markku Savela, Pekka Savola, Hesham Soliman, Dave Thaler, Mauro Tortonesi, Ole Troan, et Stig Venaas. De plus, les réviseurs anonymes de l’IESG ont effectué de nombreux et importants commentaires et suggestions de précisions.

Adresse de l’auteur

Richard Draves
Microsoft Research
One Microsoft Way
Redmond, WA 98052
USA
téléphone : +1 425 706 2268
mél : richdr@microsoft.com

Déclaration complète de droits de reproduction

Remerciement
Le financement de la fonction d’édition des RFC est actuellement fourni par la Internet Society.