Équipe d'ingénierie de l'Internet (IETF)
Request for Comments : 8981
RFC rendue obsolète : 4941
Catégorie : Sur la voie de la normalisation
ISSN : 2070-1721
Traduction Claude Brière de L'Isle

F. Gont, SI6 Networks
S. Krishnan, Kaloom
T. Narten, Microsoft Research
R. Draves, Microsoft Research
février 2021

Extension d'adresse temporaire pour l'autoconfiguration d'adresse sans état dans IPv6

Résumé
Le présent document décrit une extension à l'auto configuration d'adresse IPv6 sans état qui cause la génération par les hôtes d'adresses temporaires avec des identifiants d'interface rendus aléatoires pour chaque préfixe annoncé avec l'auto configuration activée. Changer d'adresses au fil du temps limite la fenêtre de temps durant laquelle des espions et autres collecteurs d'information peuvent facilement effectuer des correlations d'activité du réseau fondées sur l'adresse quand la même adresse est employée pour plusieurs transactions par le même hôte. De plus, cela réduit la fenêtre d'exposition d'un hôte accessible via une adresse qui est révélée par suite d'une communication active. Le présent document rend obsolète la RFC 4941.

Statut de ce mémoire
Ceci est un document de l'Internet sur la voie de la normalisation.

Le présent document a été produit par l'équipe d'ingénierie de l'Internet (IETF). Il représente le consensus de la communauté de l'IETF. Il a subi une révision publique et sa publication a été approuvée par le groupe de pilotage de l'ingénierie de l'Internet (IESG). Tous les documents approuvés par l'IESG ne sont pas candidats à devenir une norme de l'Internet ; voir la Section 2 de la RFC7841.

Les informations sur le statut actuel du présent document, tout errata, et comment fournir des réactions sur lui peuvent être obtenues à http://www.rfc-editor.org/info/rfc8981

Table des Matières

1. Introduction

La [RFC4862] spécifie l'auto configuration d'adresse sans état (SLAAC, *Stateless Address Autoconfiguration*) pour IPv6, qui résulte normalement en ce que les hôtes configurent une ou plusieurs adresses IPv6 "stables" composées d'un préfixe réseau annoncé par un routeur local et d'un identifiant d'interface (IID, *Interface Identifier*) généré en local. La sécurité et les implications pour la confidentialité de telles adresses ont été discutées en détails dans les [RFC7721], [RFC7217], et [RFC7707]. Le présent document spécifie une extension à SLAAC pour générer des adresses temporaires qui peut aider à atténuer certains des problèmes susmentionnés. Le présent document est une révision de la RFC 4941 et la rend formellement obsolète. La Section 5 décrit les changements par rapport à la [RFC4941].

Le choix d'adresse par défaut pour IPv6 a été spécifié dans la [RFC6724]. Dans certains cas, la détermination d'utiliser des adresses stables plutôt que des adresses temporaires peut seulement être faite par une application. Par exemple, certaines applications peuvent toujours vouloir utiliser des adresses temporaires, tandis que d'autres peuvent vouloir les utiliser seulement dans certaines circonstances ou pas du tout. Une interface de programmatio d'application (API, *Application Programming Interface*) comme celle spécifiée dans la [RFC5014] peut permettre à des applications individuelles d'indiquer une préférence pour l'utilisation d'adresses temporaires.

La Section 2 donne des informations sur les fondements. La Section 3 décrit une procédure pour générer des adresses temporaires. La Section 4 discute les implications du changement des IID. La Section 5 décrit les changements par rapport à la [RFC4941].

1.1 Terminologie

Les mots clés "DOIT", "NE DOIT PAS", "EXIGE", "DEVRA", "NE DEVRA PAS", "DEVRAIT", "NE DEVRAIT PAS", "RECOMMANDE", "PEUT", et "FACULTATIF" en majuscules dans ce document sont à interpréter comme décrit dans le BCP 14, [RFC2119], [RFC8174] quand, et seulement quand ils apparaissent tout en majuscules, comme montré ci-dessus.

Les termes "adresse publique", "adresse stable", "adresse temporaire", "IID constant", "IID stable", et "IID temporaire" sont à interpréter comme spécifié dans la [RFC7721].

Le terme de "adresses de portée globale" est utilisé dans ce document pour se référer collectivement à des "adresses mondiales en envoi individuel" comme défini dans la [RFC4291] et à des "adresses locales uniques" comme défini dans la [RFC4193], et non à des "adresses accessibles mondialement" comme défini dans la [RFC8190].

1.2 Position du problème

Les adresses générées en utilisant SLAAC [RFC4862] contiennent un identifiant d'interface incorporé, qui peut rester stable dans le temps. Chaque fois qu'un identifiant fixe est utilisé dans plusieurs contextes, il devient possible de corréler des activités qui semblent sans relation en utilisant cet identifiant.

La corrélation peut être effectuée par :

* Un attaquant qui est sur le chemin entre l'hôte en question et le ou les homologues avec lesquels il communique, qui peut voir les adresses IPv6 présentes dans les datagrammes.
* Un attaquant qui peut accéder aux enregistrements de communication des homologues avec lesquels l'hôte a communiqué.

Comme l'identifiant est incorporé dans l'adresse IPv6, il ne peut pas être caché. Le présent document propose une solution à ce problème en générant des identifiants d'interface qui varient dans le temps.

Noter qu'un attaquant, qui est sur le chemin, peut être capable d'effectuer des corrélation significatives sur la base :

* du contenu de la charge utile des paquets non chiffrés sur le réseau ;
* des caractéristiques des paquets, comme la taille et le rythme des paquets.

L'utilisation d'adresses temporaires ne va pas empêcher de telles corrélations, ni empêcher un observateur sur la liaison (par exemple, le routeur par défaut de l'hôte) de tracer toutes les adresses de l'hôte.

2. Fondements

Cette Section discute le problème plus en détails, donne le contexte pour évaluer la signification des problèmes dans des environnements spécifiques, et fait des comparaisons avec les pratiques existantes.

2.1 Utilisation étendue du même identifiant

L'utilisation d'IID non changeants pour former des adresses est une instance spécifique du cas plus général où un identifiant constant est réutilisé sur une période étendue et dans plusieurs activités indépendantes. Chaque fois que le même identifiant est utilisé dans plusieurs contextes, il devient possible que cet identifiant soit utilisé pour corréler des activités qui semblent sans rapport. Par exemple, un renifleur de réseau placé stratégiquemet sur une liaison traversée par tout le trafic de/vers un hôte particulier pourrait garder la trace des destinations avec lesquelles un hôte communique et à quels moments. Dans certains cas, de telles informations peuvent être utilisées pour déduire des choses, comme à quelle heure un employé est actif, quand quelqu'un est à la maison, etc. Bien qu'il pourrait apparaître que changer une adresse régulièrement dans de tels environnements serait souhaitable pour diminuer les problèmes de confidentialité, on devrait noter que la portion préfixe réseau d'une adresse sert aussi comme un identifiant constant. Tous les hôtes à, disons un domicile, vont avoir le même préfixe de réseau, qui identifie la localisation topologique de ces hôtes. Cela a des implications sur la confidentialité, bien que pas de la même granularité que le problème que traite le présent document. Spécifiquement, tous les hôtes au sein d'un domicile pourraient être groupés pour les besoins de la collecte d'informations. Si le réseau contient un très petit nombre d'hôtes -- disons, juste un -- changer juste l'IID ne va pas améiorer la confidentialité car le préfixe sert comme un identifiant constant.

Une des exigences pour corréler des activités activités apparamment sans rapport est l'utilisation (et la réutilisation) d'un identifiant qui est reconnaissable avec le temps dans différents contextes. Les adresses IP fournissent une exemple évident, mais il y en a d'autres. Par exemple :

* De nombreux hôtes ont aussi des noms DNS associés à leurs adresses, et dans ce cas, le nom DNS sert d'identifiant similaire. Bien que le nom DNS associé à une adresse donne plus de travail à obtenir (il peut exiger une interrogation du DNS) l'information est souvent directement disponible. Dans ce cas, changer l'adresse d'un hôte dans le temps aurait peu d'effet sur les problèmes soulevés dans le présent document, sauf si le nom DNS est aussi changé au même moment (voir la Section 4).

* Les navigateurs de la Toile et les serveurs échangent normalement des "mouchards" [RFC6265]. Les mouchards permettent aux serveurs de la Toile de corréler une activité actuelle avec une activité précédente. Un usage courant est de renvoyer des annonces ciblées à un utilisateur en utilisant le mouchard fourni par le navigateur pour identifier les interrogations antérieures qui ont été faites (par exemple, pour quel type d'informations). Sur la base des interrogations antérieures, des annonces peuvent être ciblées pour correspondre aux intérêts (supposés) de l'utilisateur final.

L'utilisation d'un identifiant constant au sein d'une adresse est un problème spécial, parce que les adresses sont une exigence fondamentale de communication et ne peuvent pas être facilement cachées aux espions et autres parties. Même quand les couches supérieures chiffrent leurs charges utiles, les adresses dans les en-têtes de paquet apparaissent en clair. Par conséquent, si un hôte mobile (par exemple, une tablette) accède au réseau à partir de différentes localisations, un espion pourrait être capable de retracer les mouvements de cet hôte mobile de place en place, même si les charges utiles de couche supérieure étaient chiffrées.

Changer les adresses dans le temps limite la fenêtre temporelle pendant laquelle les espions et autres collecteurs d'informations peuvent corréler de façon triviale les activités du réseau quand la même adresse est employée pour de multiples transactions par le même hôte. De plus, cela réduit la fenêtre d'exposition durant laquelle un hôte est accessible via une adresse qui est révélée par suite de communications actives.

Les implications de sécurité et de confidentialité des adresses IPv6 sont discutées en détails dans les [RFC7721], [RFC7707], et [RFC7217].

2.2 Approches possibles

Une approche, compatible avec l'architecture SLAAC, serait de changer la portion IID d'une adresse au fil du temps. Changer l'IID peut rendre plus difficile de chercher les adresses IP dans des transactions indépendantes et d'identifier lesquelles correspondent réellement au même hôte, à la fois dans le cas où la portion préfixe d'acheminement d'une adresse change et quand elle ne le fait pas.

De nombreux hôtes fonctionnent à la fois comme clients et comme serveurs. Dans ce cas, l'hôte aura besoin d'un nom (par exemple, un nom de domaine DNS) pour son utilisation comme serveur. Que l'adresse reste fixe ou change a peu d'impact sur la confidentialité, car le nom reste constant et sert d'identifiant constant. Cependant, quand il agit comme client (par exemple, pour initier une communication) un tel hôte peut vouloir varier les adresses qu'il utilise. Dans ces environnements, on peut avoir besoin de plusieurs adresses : une adresse stable associée au nom, qui est utilisée pour accepter les demandes de connexion entrantes des autres hôtes, et une adresse temporaire utilisée pour dissimuler l'identité du client quand il initie des communications.

Par ailleurs, un hôte qui fonctionne seulement comme client peut vouloir employer seulement des adresses temporaires pour la communication publique.

Pour rendre difficile de faire des suppositions élaborées quant à savoir si deux IID différents appartiennent au même hôte, l'algorithme pour générer des identifiants de remplacement doit inclure des entrées qui aient une composante imprévisible du point ce vue des entités extérieures qui collectent des informations.

3. Description du protocole

Les paragraphes qui suivent définissent les procédures de génération des adresses temporaires IPv6.

3.1 Lignes directrices pour la conception

Les adresses temporaires ont les propriétés suivantes :

1. Les adresses temporaires sont normalement employées pour initiatier des sessions sortantes.

2. Les adresses temporaires sont utilisées pour une courte période (normalement de quelques heures à quelques jours) et sont ensuite déconseillées. Les adresses déconseillées peuvent continuer d'être utilisées pour établir des connexions mais ne sont pas utilisées pour initier de nouvelles connexions.

3. Les nouvelles adresses temporaires sont générées au fil du temps pour remplacer les adresses temporaires expirées (c'est-à-dire, qui deviennent déconseillées et finalement invalidées).

4. Les adresses temporaires doivent avoir une durée de vie limitée ("durée de vie valide" et "durée de vie préférée" limitées d'après la [RFC4862]). La durée de vie d'une adresse devrait être encore plus réduite quand des événements significatifs pour la confidentialité (comme quand a lieu le rattachement d'un hôte à un réseau différent, ou la génerétion d'une nouvelle adresse aléatoire de contrôle d'accès au support (MAC, *Media Access Control*)). La durée de vie des adresses temporaires doit être statistiquement différente pour des adresses différentes, afin qu'il soit difficile de prédire ou déduire quand une nouvelle adresse temporaire est générée ou de corréler une nouvelle adresse générée avec une existante.

5. Par défaut, une adresse est générée pour chaque préfixe annoncé par SLAAC. Les identifiants d'interface résultants doivent être statistiquement différents quand les adresses sont configurées pour des préfixes différents ou des interfaces réseau différentes. Cela signifie que, avec deux adresses, il doit être difficile à une entité extérieure de déduire si les adresses correspondent au même hôte ou interface réseau.

6. Il doit être difficile à une entité extérieure de prédire les identifiants d'interface qui vont être employés pour les adresses temporaires, même en connaissant l'algorithme ou la méthode employé pour les générer et/ou en connaissant les IID employés précédemment pour d'autres adresses temporaires. Ces IID doivent être sémantiquement opaques [RFC7136] et ne doivent pas suivre de schéma spécifique.

3.2 Hypothèses

L'algorithme suivant suppose que, pour une certaine adresse temporaire, une mise en œuvre peut déterminer le préfixe d'où elle a été généré. Quand une adresse temporaire est déconseillée, une nouvelle adresse temporaire est générée. Les durées de vie valide et préférée pour la nouvelle adresse dépendent des valeurs de durée de vie correspondantes établies pour le préfixe d'où elle a été générée.

Finalement, le présent document suppose que, quand un hôte initie des communications sortantes, les adresses temporaires peuvent avoir la préférence sur des adresses stables (si il en est de disponibles) quand l'appareil est configuré à le faire. La [RFC6724] rend obligatoire que les mises en œuvre fournissent un mécanisme permettant à une application de configurer sa préférence pour les adresses temporaires sur les adresses stables. Elle permet aussi à une mise en œuvre de préférer les adresses temporaires par défaut, afin que les connexions initiées par l'hôte puissent utiliser des adresses temporaires sans exiger une activation spécifique de l'application. Le présent document suppose aussi qu'une API va exister pour permettre à des applications individuelles d'indiquer si elles préfèrent utiliser des adresses temporaires ou stables et outrepasser les réglages par défaut du système (voir, par exemple, la [RFC5014]).

3.3 Génération d'IID aléatoires

Les paragraphes qui suivent spécifient des exemples d'algorithmes pour générer des IID temporaires qui suivent les lignes directrices du paragraphe 3.1 du présent document. L'algorithme spécifié au pararaphe 3.3.1 suppose qu'un générateur de nombres pseudo aléatoires (PRNG, *pseudorandom number generator*) est disponible sur le système. L'algorithme spécifié au paragraphe 3.3.2 permet la réutilisation du code par les hôtes qui mettent en œuvre la [RFC7217].

3.3.1 IID aléatoires simples

Une approche est de choisir un nombre pseudo aléatoire de la longueur appropriée. Un hôte employant cet algorithmz devrait générer les IID comme suit :

1. Obtenir un nombre aléatoire d'un PRNG qui peut produire des nombres aléatoires d'au moins autant de bits qu'exigé pour le IID (voir l'étape suivante). La [RFC4086] spécifie les exigences d'aléa pour la sécurité.

2. Le IID est obtenu en prenant dans le nombre aléatoire obtenu à l'étape précédente autant de bits que nécessaire. Voir dans la [RFC7136] le nombre nécessaire de bits (c'est-à-dire, la longueur de l'IID). Voir aussi dans la [RFC7421] la discussion des implications de la longueur de l'IID pour la confidentialité. Note : il n'y a pas de bits spéciaux dans un IID [RFC7136].

3. L'IID résultant IID DOIT être comparé aux IID IPv6 réservés [RFC5453] [IANA-IID] et aux IID déjà employés dans une adresse de la même interface réseau et du même préfixe de réseau. Si un identifiant inacceptable a été généré, un nouvel IID devrait être généré en répétant l'algorithme depuis la première étape.

3.3.2 Génération de IID avec des fonctions pseudo aléatoires

L'algorithme de la [RFC7217] peut être augmenté pour la génération d'adresses temporaires. L'avantage en est qu'un hôte pourrait employer un seul algorithme pour générer des adresses stables et temporaires en employant les paramètres appropriés.

Les hôtes vont employer l'algorithme suivant pour générer l'IID temporaire :

1. Calculer un identifiant aléatoire avec l'expression :

   RID = F(Préfixe, Net_Iface, Identifiant_de_réseau, Heure, Compteur_DAD, clé_secrète)

Où :

RID : Identifiant aléatoire

F() : fonction pseudo aléatoire (PRF) qui NE DOIT PAS être calculable de l'extérieur (sans connaissance de la clé secrète). F() DOIT aussi être difficile à inverser, afin qu'il résiste aux tentatives d'obtenir la clé_secrète, même quand des échantillons donnés du résultat de F() et la connaisssance ou le contrôle des autres paramètres d'entrée. F() DEVRAIT

produire un résultat d'au moins autant de bits qu'exigé pour le IID. BLAKE3 (clé de 256 bits, de résultat de longueur arbitraire) [BLAKE3] est une option possible pour F(). Autrement, F() pourrait être mis en œuvre avec un code d'authentification de message (HMAC) [RFC2104] de hachage de clé. HMAC-SHA-256 [FIPS-SHS] est une option possible pour une telle solution de remplacement de mise en œuvre. Note: l'utilisation de HMAC-MD5 [RFC1321] est considérée comme inacceptable pour F() [RFC6151].

Préfixe : préfixe à utiliser pour SLAAC, comme appris d'un message d'annonce de routeur ICMPv6.

Net_Iface : adresse MAC correspondant à la carte d'interface réseau sous-jacent, dans le cas où la liaison utilise des identifiants de couche de liaison IEEE 802. Employer l'adresse MAC pour ce paramètre (plutôt que les autres options suggérées dans la [RFC7217]) signifie que la regénération d'une adresse MAC aléatoire va résulter en une adresse temporaire différente.

Identifiant_de_réseau : données spécifiques du réseau qui identifient le sous réseau auquel cette interface est rattachée -- par exemple, l'identifiant d'ensemble de service (SSID, *Service Set Identifier*) IEEE 802.1 correspondant au réseau auquel cette interface est associée. De plus, les "Procédures simples pour détecter le rattachement au réseau dans IPv6" ("Simple DNA") [RFC6059] décrivent les idées qui pourraient être développées pour générer un paramètre Identifiant_de_réseau. Ce paramètre DEVRAIT être employé si une forme de "Identifiant_de_réseau" est disponible.

Heure : représentation de l'heure dépendante de la mise en œuvre. Un exemple possible est la représentation des systèmes de style UNIX [OPEN-GROUP], qui mesure le temps en termes de nombre de secondes écoulées depuis le 1er janvier 1970 (00:00:00 en temps universel coordonné (UTC, *Coordinated Universal Time*)). L'ajout de l'argument "Heure" résulte en IID différents (statistiquement) dans le temps.

Compteur_DAD : compteur employé pour résoudre le conflit où un identifiant inacceptable a été généré. Ce peut être le résultat de la détection d'adresses dupliquées (DAD, *Duplicate Address Detection*) ou de l'étape 3 ci-dessous.

clé_secrète : clé secrète qui n'est pas connue de l'attaquant. La clé secrète DEVRAIT être d'au moins 128 bits. Elle DOIT être initialisée à un nombre pseudo aléatoire (voir dans la [RFC4086] les exigences d'aléa pour la sécurité) quand le système d'exploitation est "réamorcé". La clé secrète NE DOIT PAS être employée pour d'autre objet que celui discuté dans ce paragraphe. Par exemple, les mises en œuvre NE DOIVENT PAS employer la même clé_secrète pour la génération d'adresses stables [RFC7217] et la génération d'adresses temporaires via cet algorithme

.

2. L'IID est finalement obtenu en prenant autant de bits que nécessaire de la valeur de RID (calculée à l'étape précédante) en commençant par le bit de moindre poids. Voir dans la [RFC7136] le nombre de bits nécessaire (c'est-à-dire, la longueur de l'IID). Voir aussi dans la [RFC7421] la discussion des implications de confidentialité de la longueur de l'IID. Note : il n'y a pas de bits spéciaux dans un IID [RFC7136].

3. L'IID résultant DOIT être comparé aux IID IPv6 réservés [RFC5453] [IANA-IID] et aux IID déjà employés dans une adresse de la même interface réseau et de même préfixe de réseau. Dans le cas où un identifiant inacceptable aurait été généré, le compteur_DAD devrait être incrémenté de 1, et l'algorithme devrait être redémarré à la première étape.

3.4 Génération des adresses temporaires

La [RFC4862] décrit les étapes pour générer une adresse de liaison locale quand une interface devient activée, ainsi que les étapes pour générer des adresses pour d'autres portées. Le présent document étend la [RFC4862] comme suit. Quand on traite une annonce de routeur avec une option Informations de préfixe portant un préfixe pour les besoins de l'auto configuration d'adresse (c'est-à-dire que le bit A est établi) l'hôte DOIT effectuer les étapes suivantes :

1. Traiter l'option Informations de préfixe comme spécifié dans la [RFC4862], en ajustant les durées de vie des adresses temporaires existantes, avec la contrainte globale qu'aucune adresse temporaire ne devrait rester "valide" ou "préférée" plus longtemps que respectivement "TEMP_VALID_LIFETIME" ou "TEMP_PREFERRED_LIFETIME - DESYNC_FACTOR". Les variables de configuration "TEMP_VALID_LIFETIME" et "TEMP_PREFERRED_LIFETIME" correspondent respectivement à la durée de vie maximum valide et à la durée de vie maximum préférée des adresses temporaires.
   Note : DESYNC_FACTOR est la valeur calculée quand l'adresse a été créée (voir l'étape 4 ci-dessous).

2. Une façon dont une mise en œuvre peut satisfaire les contraintes ci-dessus est d'associer à chaque adresse temporaire une heure de création (appelée CREATION_TIME) qui indique l'heure à laquelle l'adresse a été créée. Quand on met à jour la durée de vie préférée d'une adresse temporaire existante, elle va être réglée à expirer à toute heure antérieure :

l'heure indiquée par la durée de vie reçue ou (CREATION_TIME + TEMP_PREFERRED_LIFETIME - DESYNC_FACTOR). Une approche similaire peut être utilisée avec la durée de vie valide.
Note : DESYNC_FACTOR est la valeur calculée quand l'adresse a été créée (voir l'étape 4 ci-dessous).

3. Si l'hôte n'a pas configuré d'adresse temporaire pour le préfixe correspondant, il DEVRAIT créer une nouvelle adresse temporaire pour ce préfixe.
   Note : par exemple, un hôte pourrait mettre en œuvre des politiques spécifiques du préfixe comme de ne pas configurer d'adresses temporaires pour le préfixe d'adresse d'envoi individuel unique locale IPv6 (ULA) [RFC4193].

4. Quand il crée une adresse temporaire, DESYNC_FACTOR DOIT être calculé et associé à la nouvelle adresse créée, et les valeurs de durée de vie d'adresse DOIVENT être déduites du préfixe correspondant comme suit :
   * Sa durée de vie valide est le plus petit de la durée de vie valide du préfixe et de TEMP_VALID_LIFETIME.
   * Sa durée de vie préférée est le plus petit de la durée de vie préférée du préfixe et de TEMP_PREFERRED_LIFETIME - DESYNC_FACTOR.

5. Une adresse temporaire n'est créée que si cette durée de vie préférée calculée est supérieure à REGEN_ADVANCE unités de temps. En particulier, une mise en œuvre NE DOIT PAS créer une adresse temporaire avec une durée de vie préférée de zéro.

6. Les nouvelles adresses temporaires DOIVENT être créées en ajoutant un IID aléatoire au préfixe reçu. Le paragraphe 3.3 du présent document spécifie des exemples d'algorithmes pour générer l'IID aléatoire.

7. L'hôte DOIT effectuer une DAD sur l'adresse temporaire générée. Si la DAD indique que l'adresse est déjà utilisée, l'hôte DOIT générer une nouvelle IID aléatoire et répéter les étapes précédentes comme approprié (en commençant à l'étape 4) jusqu'à TEMP_IDGEN_RETRIES fois. Si, après TEMP_IDGEN_RETRIES tentatives consécutives, l'hôte est incapable de générer une adresse temporaire unique, l'hôte DOIT enregistrer une erreur système et NE DEVRAIT PAS tenter de générer une adresse temporaire pour le préfixe considéré pour la durée du rattachement de l'hôte au réseau via cette interface. Cela permet aux hôtes de récupérer de défaillances occasionnelles de DAD ou autrement d'enregistrer des collisions d'adresses récurrentes.

3.5 Expiration des adresses temporaires

Quand une adresse temporaire devient déconseillée, une nouvelle DOIT être générée. Ceci est fait en répétant les actions décrites au paragraphe 3.4, en commençant à l'étape 4). Noter que, en fonctionnement normal, sauf pour la période transitoire où une adresse temporaire est regénérée, au plus une adresse temporaire par préfixe devrait être dans un état non déconseillé à tout moment sur l'interface considérée. Noter que si une adresse temporaire devient déconseillée par suite du traitement d'une option Informations de préfixe avec une durée de vie préférée de zéro, une nouvelle adresse temporaire NE DOIT alors PAS être générée (en réponse à la même option Informations de préfixe). Pour s'assurer qu'une adresse temporaire préférée est toujours disponible, une nouvelle adresse temporaire DEVRAIT être regénérée légèrement avant que la précédente soit déconseillée. Cela permet qu'un temps suffisant soit donné pour éviter des conditions de concurrence dans le cas où la génération d'une nouvelle adresse temporaire n'est pas instantanée, comme quand la DAD doit être effectuée. L'hôte DEVRAIT commencer le processus de regéneration d'adresse REGEN_ADVANCE unités de temps avant qu'une adresse temporaire soit déconseillée.

Comme optimisation facultat*ve*, une mise en œuvre PEUT supprimer une adresse temporaire déconseillée qui n'est pas utilisée par des applications ou des couches supérieures, comme précisé à la Section 6.

3.6 Régénération des adresses temporaires

La fréquence à laquelle les adresses temporaires changent dépend de l'utilisation de l'appareil (par exemple, à quelle fréquence il initie de nouvelles communications) et des objectifs de l'utilisateur final. Les principaux soucis de confidentialité des adresses paraissent impliquer les adresses utilisées pendant de longues périodes (d'une semaine à un an). Plus une adresse change fréquemment, moins il est faisable de collecter ou coordonner des informations sur les IID. De plus, le coût de collecte des informatione et des tentatives de les corréler sur la base des IID va seulement être justifié si assez d'adresses contiennent des identifiants qui ne changent pas pour que cela en vaille la peine. Donc, avoir un grand nombre de clients qui changent leur adresse quotidiennement ou toutes les emaines est probablement suffisant pour résoudre la plupart des soucis de confidentialité.

Il y a aussi des coûts de client découlant d'un grand nombre d'adresses associées à un hôte (par exemple, pour faire des

recherches d'adresse, le besoin de joindre de nombreux groupes de diffusion groupée, etc.). Donc, changer fréquemment les adresses (par exemple, toutes les quelques minutes) peut avoir des implications de performances.

Les hôtes qui suivent la présente spécification DEVRAIENT générer de nouvelles adresses temporaires au fil du temps. Cela peut se faire en générant une nouvelle adresse temporaire REGEN_ADVANCE unités de temps avant qu'une adresse temporaire devienne déconseillée. Comme décrit ci-dessus, cela produit des adresses avec une durée de vie préférée de pas plus de TEMP_PREFERRED_LIFETIME. La valeur de DESYNC_FACTOR est une valeur aléatoire calculée quand une adresse temporaire est générée ; elle assure que les clients ne génèrent pas de nouvelles adresses à une fréquence fixe et que les clients ne se synchronisent pas les uns avec les autres et ne génèrent pas de nouvelles adresses exactement au même moment. Quand la durée de vie préférée expire, une nouvelle adresse temporaire DOIT être générée en utilisant l'algorithme spécifié au paragraphe 3.4 (en commençant à l'étape 4).

Parce que la fréquence à laquelle il est approprié de générer de nouvelles adresses varie d'un environnement à l'autre, les mises en œuvre DEVRAIENT fournir aux utilisateurs d'extrémité la capacité de changer la fréquence à laquelle les adresses sont regénérées. La valeur par défaut est donnée dans TEMP_PREFERRED_LIFETIME et est d'un jour. De plus, l'heure exacte à laquelle invalider une adresse temporaire dépend de la façon dont les applications sont utilisées par les utilisateurs d'extrémité. Donc, la valeur par défaut suggérée de deux jours (TEMP_VALID_LIFETIME) peut n'être pas appropriée dans tous les environnements. Les mises en œuvre DEVRAIENT fournir aux utilisateurs d'extrémité la capacité d'outrepasser ces valeurs par défaut.

Finalement, quand une interface se connecte à une nouvelle liaison (différente) les adresses temporaires existantes pour l'interface correspondante DOIVENT être supprimées, et de nouvelles adresses temporaires DOIVENT être générées pour être utilisées sur la nouvelle liaison, en utilisant l'algorithme du paragraphe 3.4. Si un appareil passe d'une liaison à une autre, générer de nouvelles adresses temporaires assure que l'appareil utilise des IID aléatoires différents pour les adresses temporaires associées aux deux liaisons, rendant plus difficile de corréler les adresses provenant des deux liaisons différentes comme venant du même hôte. L'hôte PEUT suivre tout processus qui lui est disponible pour déterminer que le changement de liaison s'est produit. Un de ces processus est décrit dans "Simple DNA" [RFC6059]. Détecter les changements de liaison empêcherait les événements de fermeture/établissement de liaison de causer la regénération (inutile) d'adresses temporaires.

3.7 Considérations de mise en œuvre

Les appareils qui mettent en œuvre la présente spécification DOIVENT fournir un moyen pour que l'utilisateur final active ou désactive explicitement l'utilisation des adresses temporaires. De plus, un site pourrait souhaiter désactiver l'utilisation des adresses temporaires afin de simplifier le débogage et les opérations du réseau. Par conséquent, les mises en œuvre DEVRAIENT fournir un moyen pour que les administrateurs de système de confiance activent ou désactivent l'utilisation des adresses temporaires.

De plus, les sites pourraient souhaiter activer ou désactiver sélectivement l'utilisation des adresses temporaires pour certains préfixes. Par exemple, un site pourrait souhaiter désactiver la génération d'adresses temporaires pour les préfixes ULA [RFC4193] tout en générant quand même des adresses temporaires pour tous les autres préfixes annoncés via des PIO pour la configuration d'adresse. Un autre site pourrait souhaiter activer la génération d'adresses temporaires seulement pour les préfixes 2001:db8:1::/48 et 2001:db8:2::/48 tout en la désactivant pour tous les autres préfixes. Pour prendre en charge ce comportement, les mises en œuvre DEVRAIENT fournir un moyen d'activer et désactiver la génération des adresses temporaires pour des sous gammes de préfixes spécifiques. Ce réglage par préfixe DEVRAIT outrepasser les réglages globaux sur l'hôte par rapport aux sous gammes de préfixes spécifiées. Noter que le réglage par préfixe peut être appliqué à toutes les granularités, et pas nécessairement celle du sous réseau.

3.8 Paramètres de protocole et variables de configuration définis

Les paramètres de protocole et les variables de configuration définis dans le présent document incluent :

TEMP_VALID_LIFETIME : par défaut : 2 jours. Les utilisateurs devraient être capables d'outrepasser la valeur par défaut.

TEMP_PREFERRED_LIFETIME : par défaut : 1 jour. Les utilisateurs devraient être capables d'outrepasser la valeur par défaut. Note : la valeur de TEMP_PREFERRED_LIFETIME DOIT être inférieure à la valeur de TEMP_VALID_LIFETIME, pour éviter le cas pathologique où une adresse est employée pour de nouvelles communications mais devient invalide en moins d'une seconde, interrompant ces communications.

REGEN_ADVANCE 2 + (TEMP_IDGEN_RETRIES * DupAddrDetectTransmits * RetransTimer / 1000)
Raison : ce paramètre est spécifié comme une fonction d'autres paramètres de protocole, pour tenir compte du temps éventuellement passé en DAD dans le pire scénario de TEMP_IDGEN_RETRIES. Cela empêche le cas pathologique où la génération d'une nouvelle adresse temporaire n'est pas lancée avec une anticipation suffisante, comme lorsque une nouvelle adresse préférée est générée avant que l'adresse temporaire actuellement préférée devienne déconseillée. RetransTimer est spécifié dans la [RFC4861], tandis que DupAddrDetectTransmits est spécifié dans la [RFC4862]. Comme RetransTimer est spécifié en unités de millisecondes, cette expression emploie la constante "1000", afin que REGEN_ADVANCE soit exprimé en secondes.

MAX_DESYNC_FACTOR 0,4 * TEMP_PREFERRED_LIFETIME. Limite supérieure de DESYNC_FACTOR.
Raison : régler MAX_DESYNC_FACTOR à 0,4 TEMP_PREFERRED_LIFETIME résulte en ce que les adresses ont des durées de vie statistiquement différentes, et un maximum de trois adresses temporaires concurrentes quand les valeurs par défaut spécifiées dans cette section sont employées.

DESYNC_FACTOR : valeur aléatoire dans la gamme de 0 à MAX_DESYNC_FACTOR. Elle est alculée chaque fois qu'une adresse temporaire est générée, et est associée à l'adresse correspondante. Elle DOIT être inférieure à (TEMP_PREFERRED_LIFETIME - REGEN_ADVANCE).

TEMP_IDGEN_RETRIES : valeur par défaut : 3.

4. Implications du changement des IID

Le désir de protéger la confidentialité individuelle peut entrer en conflit avec le désir de maintenir et déboguer efficacement un réseau. Avoir des clients qui utilisent des adresses qui changent dans le temps va rendre plus difficile de retracer et isoler les problèmes de fonctionnement. Par exemple, quand on cherche les traces d'un paquet, il pourrait devenir plus difficile de déterminer si on voit le comportement causé par un seul hôte errant ou par un certain nombre d'entre eux.

Il est actuellement recommandé que les déploiements de réseau fournissent plusieurs adresses IPv6 provenant de chaque préfixe aux hôtes généralistes [RFC7934]. Cependant, dans certains scénarios, l'utilisation d'un grand nombre d'adresses IPv6 peut avoir des implications négatives sur les appareils du réseau qui ont besoin de maintenir des entrées pour chaque adresse IPv6 dans certaines structures de données (par exemple, SAVI [RFC7039]). Par exemple, l'utilisation active concurrente de multiples adresses IPv6 va augmenter le trafic de découverte de voisin si les antémémoires de voisins dans les appareils du réseau ne sont pas assez grandes pour mémoriser toutes les adresses sur la liaison. Cela peut impacter les performances et l'efficacité énergétique sur les réseaux où la diffusion groupée est coûteuse (voir par exemple [MCAST-PROB]). De plus, certains appareils de sécurité de réseau pourraient incorrectement déduire une falsification d'adresse IPv6 si les adresses temporaires sont regénérées très fréquemment.

L'utilisation d'adresses temporaires peut causer des difficultés inattendues avec certaines applications. Par exemple, certains serveurs refusent d'accepter des communications provenant de clients pour lesquels ils ne peuvent pas transposer l'adresse IP en un nom DNS. C'est-à-dire, ils effectuent une interrogation DNS PTR pour déterminer le nom DNS correspondant à une adresse IPv6, et peuvent alors aussi effectuer une interrogation AAAA sur le nom retourné pour vérifiier qu'il se retranspose bien en la même adresse. Par conséquent, les clients qui ne sont pas correctement enregistrés dans le DNS peuvent être incapables d'accéder à certains services. Cependant, le nom DNS d'un hôte (si il ne change pas) servirait d'identifiant constant. Le large déploiement de l'extension décrite dans le présent document pourrait mettre au défi la pratique de la "validation" fondée sur la recherche DNS inverse, qui a peu de validité, bien que largement mise en œuvre. Afin de répondre au défi du serveur, les hôtes pourraient enregistrer les adresses temporaires dans le DNS en utilisant des noms aléatoires (par exemple, une version de chaîne de l'adresse aléatoire elle-même) bien qu'au prix d'une complexité accrue.

De plus, certaines applications peuvent ne pas se comporter de façon robuste si une adresse devient invalide alors qu'elle est encore utilisée par l'application ou si l'application ouvre plusieurs sessions et attend d'elles qu'elles utilisent toutes la même adresse.

La [RFC4941] a employé un IID temporaire aléatoire pour générer un ensemble d'adresses temporaires, de telle façon que les adresses temporaires configurées à un moment donné pour plusieurs préfixes SLAAC emploient le même IID. Partager le même IID sur plusieurs addresses permettait à un hôte de joindre seulement un groupe de diffusion groupée de nœud sollicité par ensemble d'adresses temporaires.

Le présent document exige que les IID de toutes les adresses temporaires sur un hôte soient statistiquement différentes les

unes des autres. Cela signifie que quand un réseau emploie plusieurs préfixes, chaque adresse temporaire d'un ensemble va résulter en une adresse différente de diffusion groupe de nœud sollicité, et, donc, le nombre de groupes de diffusion groupée qu'un hôte doit joindre devient une fonction du nombre de préfixes SLAAC employés pour générer les adresses temporaires.

Donc, un réseau qui emploie plusieurs préfixes peut exiger des hôtes qu'ils se joignent à plus de groupes de diffusion groupée que dans le cas des mises en œuvre de la RFC 4941. Si le nombre de groupes de diffusion groupée était assez grand, un hôte pourrait avoir besoin de changer le réglage de la carte d'interface réseau en mode de promiscuité. Cela pourrait causer le traitement par l'hôte de plus de paquets que strictement nécessaire et pourrait avoir un impact négatif sur la durée de vie des batteries et des performances du système en général.

On note que comme le présent document réduit la valeur par défaut de TEMP_VALID_LIFETIME de 7 jours (dans la [RFC4941]) à 2 jours, le nombre d'adresses temporaires concurrentes par préfixe de SLAAC va être plus petit que pour les mises en œuvre de la RFC 4941 ; donc, le nombre de groupes de diffusion groupée pour un réseau qui emploie, disons, entre 1 et 3 préfixes, va être similaire au nombre de ces groupes pour les mises en œuvre de la RFC 4941.

Les mises en œuvre concernées par le nombre maximum de groupes de diffusion groupée qui serait exigé pour s'y joindre par suite des adresses configurées, ou par le nombre global d'adresses configurées, devraient envisager d'appliquer des limites spécifiques, par exemple, le nombre maximum d'adresses configurées, le nombre maximum de préfixes SLAAC qui sont employés pour l'autoconfiguration, et/ou le ratio maximum pour TEMP_VALID_LIFETIME / TEMP_PREFERRED_LIFETIME (qui en fin de compte contrôle le nombre approximatif d'adresses temporaires concurrentes par préfixe SLAAC). Beaucoup de ces limites de configuration sont déjà disponibles dans les mises en œuvre de SLAAC et de la RFC 4941. On note que ces limites configurables sont destinées à empêcher des comportements pathologiques (par opposition à simplement limiter l'usage des adresses IPv6) car les mises en œuvre de IPv6 sont supposées développer l'usage d'adresses multiples [RFC7934].

5. Changements significatifs par rapport à la RFC 4941

Cette Section résume les changements substantiels du présent document par rapport à la RFC 4941.

En gros, le présent document introduit les changements suivants :

* Corrige un certain nombre de fautes dans l'algorithme pour générer des adresses temporaires. Les fautes concernées incluent l'utilisation de MD5 pour calculer les IID temporaires, et réutiliser le même IID pour plusieurs préfixes (voir [RAID2015] et la [RFC7721] pour les détails).

* Permet aux hôtes d'employer seulement des adresses temporaires. La [RFC4941] supposait que les adresses temporaires étaient configurées en plus des adresses stables. Le présent document n'implique pas ni n'exige la configuration d'adresses stables ; donc, les mises en œuvre peuvent maintenant configurer à la fois des adresses stables et des adresses temporaires ou des adresses temporaires seulement.

* Supprime la recommandation que les adresses temporaires soient désactivées par défaut. Ceci est en ligne avec le BCP 188 ([RFC7258]) et aussi avec le BCP 204 ([RFC7934]).

* Réduit la durée de vie valide maximum par défaut pour les adresses temporaires (TEMP_VALID_LIFETIME). TEMP_VALID_LIFETIME a été réduit de 1 semaine à 2 jours, diminuant le nombre normal d'adresses temporaires concurrentes de 7 à 3. Cela réduit la tension possible sur les éléments de réseau (voir les détails à la Section 4).

* DESYNC_FACTOR est calculé chaque fois qu'une adresse temporaire est générée et est associée à l'adresse temporaire correspondante, afin que chaque adresse temporaire ait une durée de vie préférée statistiquement différente, et donc les adresses temporaires ne sont pas générées à une fréquence spécifique.

* Change l'exigence de ne pas essayer de regénérer des adresses temporaires TEMP_IDGEN_RETRIES fois défaillances de DAD consécutives de "NE DOIT PAS" en "NE DEVRAIT PAS".

* La discussion sur les implications de sécurité et de confidentialité des différentes techniques de génération d'adresses a été remplacée par des références aux travaux récents dans ce domaine ([RFC7707], [RFC7721], et [RFC7217]).

* Le présent document incorpore les errata soumis (au moment de la rédaction) poir la [RFC4941] par Jiri Bohac et

Alfred Hoenes.

6. Travaux futurs

Une mise en œuvre pourrait vouloir garder trace des adresses utilisées par les couches supérieures afin d'être capable de supprimer une adresse temporaire déconseillée des structures de données internes quand aucun protocole de couche supérieure n'en utilise (mais pas avant). Ceci est différent des approches actuelles, où des adresses sont supprimées d'une interface quand elles deviennent invalides [RFC4862], indépendamment de si les protocoles de couche supérieure sont ou non en train de les utiliser. Pour les connexions TCP, de telles informations sont disponibles dans les blocs de contrôle. Pour les applications fondées sur UDP, il se peut que seules les applications aient connaissance des adresses qui sont actuellement utilisées. Par conséquent, une mise en œuvre va généralement devoir utiliser des heuristiques pour décider quand une adresse n'est plus utilisée.

7. Considérations relatives à l'IANA

Le présent document n'appelle à aucune action de la part de l'IANA.

8. Considérations sur la sécurité

Si un très petit nombre d'hôtes (disons, seulement un) utilisent un certain préfixe pendant de longues périodes, changer juste la partie identifiant d'interface de l'adresse peut n'être pas suffisant pour atténuer la corrélation d'activité réseau fondée sur l'adresse, car le préfixe agit comme identifiant constant. Les procédures décrites dans le présent document sont plus efficaces quand le préfixe est raisonnablement non statique ou utilisé par un très grand nombre d'hôtes. De plus, si une adresse temporaire est utilisée dans une session où l'utilisateur s'authentifie, toute notion de "confidentialité" pour cette adresse est compromise pour la ou les parties qui reçoivent les informations d'authentification.

Bien que le présent document discute des moyens de limiter la durée de vie des identifiants d'interface pour réduire la capacité des attaquants d'effectuer une corrélation d'activité du réseau fondée sur l'adresse, la méthode décrite est estimée être inefficace contre des formes sophistiquée d'analyse du trafic. Pour augmenter l'efficacité, on peut devoir considérer l'utilisation de techniques plus évoluées, comme l'acheminement en pelure d'oignon [ONION].

Le filtrage d'entrée a été et est déployé comme un moyen d'empêcher l'utilisation d'adresses de source usurpées dans des attaques de déni de service réparties (DDoS, *Distributed Denial of Service*). Dans un réseau avec un grand nombre d'hôtes, de nouvelles adresses temporaires sont créée à un très fort taux. Cela pourrait rendre difficile aux mécanismes de filtrage d'entrée/sortie de distinguer entre un changement légitime d'adresses temporaires et des adresses de source usurpées, qui sont "dans le préfixe" (en utilisant un préfixe topologiquement correct et un identifiant d'interface non existant). Ceci peut être traité en utilisant des mécanismes de contrôle d'accès par adresse sur le point d'entrée du réseau -- bien que, comme noté à la Section 4, il y ait des coûts correspondants pour le faire.

9. Références

9.1 Références normatives

[RFC2119] S. Bradner, "Mots clés à utiliser dans les RFC pour indiquer les niveaux d'exigence", BCP 14, DOI 10.17487/RFC2119, mars 1997. *(MàJ par* RFC8174)

[RFC4086] D. Eastlake 3rd, J. Schiller, S. Crocker, "Exigences d'aléa pour la sécurité", juin 2005. (*Remplace* RFC1750) (BCP0106)

[RFC4193] R. Hinden, B. Haberman, "Adresses IPv6 en envoi individuel uniques localement", octobre 2005. *(P.S.)*

[RFC4291] R. Hinden, S. Deering, "Architecture d'adressage IP version 6", février 2006. (*MàJ par* 5952 et 6052, 8064) *(D.S.)*

[RFC4861] T. Narten et autres, "Découverte du voisin pour IP version 6 (IPv6)", septembre 2007. (*Remplace* RFC2461) *(D.S. ; MàJ par* RFC8028, RFC8319, RFC8425, RFC9131)

[RFC4862] S. Thomson et autres, "Auto configuration d'adresse IPv6 sans état", septembre 2007. (*Remplace* RFC2462) *(D.S.)*

[RFC5453] S. Krishnan, "Identifiants d'interface IPv6 réservés", février 2009. *(P.S.)*

[RFC6724] D. Thaler, R. Draves, A. Matsumoto, T. Chown, "Choix de l'adresse par défaut pour IPv6", septembre 2012. (*Remplace la* RFC3484) *(P.S.)*

[RFC7136] B. Carpenter, S. Jiang, "Signification des identifiants d'interface IPv6", février 2014. (*MàJ* RFC4291) *(P.S.)*

[RFC8174] B. Leiba, "Ambiguïté des mots clés en majuscules ou minucules dans la RFC2119", mai 2017. BCP14. *(MàJ* 2119)

9.2 Références pour information

[BLAKE3] O'Connor, J., Aumasson, J. P., Neves, S., and Z. Wilcox- O'Hearn, "BLAKE3: one function, fast everywhere", 2020, <https://blake3.io/>.

[FIPS-SHS] NIST, "Secure Hash Standard (SHS)", FIPS PUB 180-4, DOI 10.6028/NIST.FIPS.180-4, août 2015, <https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.180-4.pdf>.

[IANA-IID] IANA, "Reserved IPv6 Interface Identifiers", <https://www.iana.org/assignments/ipv6-interface-ids>.

[MCAST-PROB] Perkins, C. E., McBride, M., Stanley, D., Kumari, W., and J. C. Zuniga, "Multicast Considerations over IEEE 802 Wireless Media", travail en cours, Internet-Draft, draft-ietf-mboned-ieee802-mcast-problems-13, 4 février 2021, <https://tools.ietf.org/html/draft-ietf-mboned-ieee802-mcast-problems-13>.

[ONION] Reed, M.G., Syverson, P.F., and D.M. Goldschlag, "Proxies for Anonymous Routing", Proceedings of the 12th Annual Computer Security Applications Conference, DOI 10.1109/CSAC.1996.569678, décembre 1996, <https://doi.org/10.1109/CSAC.1996.569678>.

[OPEN-GROUP] The Open Group, "The Open Group Base Specifications Issue 7", Section 4.16 Seconds Since the Epoch, IEEE Std 1003.1, 2016, <http://pubs.opengroup.org/onlinepubs/9699919799/basedefs/contents.html>.

[RAID2015] Ullrich, J. and E.R. Weippl, "Privacy is Not an Option: Attacking the IPv6 Privacy Extension", International Symposium on Recent Advances in Intrusion Detection (RAID), 2015, <https://publications.sba-research.org/publications/Ullrich2015Privacy.pdf>.

[RFC1321] R. Rivest, "Algorithme de résumé de message MD5", avril 1992. *(Information)*

[RFC2104] H. Krawczyk, M. Bellare et R. Canetti, "HMAC : Hachage de clés pour l'authentification de message", février 1997.

[RFC4941] T. Narten et autres, "Extensions de confidentialité pour l'auto configuration d'adresse sans état dans IPv6", septembre 2007. (*D.S. ; remplace* RFC3041 ; *remplacée par* RFC8981*)*

[RFC5014] E. Nordmark et autres, "API de prises IPv6 pour la sélection d'adresse de source", septembre 2007. *(Information)*

[RFC6059] S. Krishnan, G. Daley, "Procédures simples pour détecter le rattachement au réseau en IPv6", novembre 2010. *(P.S.)*

[RFC6151] S. Turner, L. Chen, "Mise à jour des considérations de sécurité pour les algorithmes de résumé de message MD5 et le HMAC-MD5", mars 2011. (*MàJ* RFC1321, RFC2104) *(Information)*

[RFC6265] A. Barth, "Mécanisme de gestion d'état HTTP", avril 2011. *(Remplace la* RFC2965) *(P.S.)*

[RFC7039] J. Wu et autres, "Cadre pour l'amélioration de validation d'adresse de source (SAVI)", octobre 2013. *(Information)*

[RFC7217] F. Gont, "Méthode pour générer des identifiants d'interface sémantiquement opaques avec l'autoconfiguration d'adresse IPv6 sans état (SLAAC)", avril 2014. *(P.S.)*

[RFC7258] S. Farrell, H. Tschofenig, "La surveillance envahissante est une attaque", BCP0188, mai 2014. *(MàJ par* RFC8044)

[RFC7421] B. Carpenter, et autres, "Analyse de la frontière de 64 bits dans l'adressage IPv6", janvier 2015.. *(Information)*

[RFC7707] F. Gont, T. Chown, "Reconnaissance de réseau dans les réseaux IPv6", mars 2016. *(Information)*

[RFC7721] A. Cooper, F. Gent, D. Thaler, "Considérations de sécurité et de confidentialité pour les mécanismes de génération d'adresse IPv6", mars 2016. *(Information)*

[RFC7934] L. Coliti, et autres, "Recommandations pour la disponibilité des adresses d'hôte", juillet 2016. BCP 204.

[RFC8190] R. Bonica, et autres, "Mise à jour des registres d'adresse IP d'utilisation particulière", juin 2017. BCP 153 *(MàJ* 6890)

Remerciements

Fernando Gont était le seul auteur de ce document (une révision de la RFC 4941). Il souhaite remercier (par ordre alphabétique) Fred Baker, Brian Carpenter, Tim Chown, Lorenzo Colitti, Roman Danyliw, David Farmer, Tom Herbert, Bob Hinden, Christian Huitema, Benjamin Kaduk, Erik Kline, Gyan Mishra, Dave Plonka, Alvaro Retana, Michael Richardson, Mark Smith, Dave Thaler, Pascal Thubert, Ole Troan, Johanna Ullrich, Eric Vyncke, Timothy Winters, et Christopher Wood qui ont fourni de précieux commentaires sur les premières versions du projet de ce document.

Le présent document incorpore les errata soumis pour la RFC 4941 par Jiri Bohac et Alfred Hoenes (au moment de la rédaction).

Suresh Krishnan était le seul auteur de la RFC 4941 (une révision de la RFC 3041). Il tient à reconnaître les contributions du groupe de travail IPv6 et, en particulier, de Jari Arkko, Pekka Nikander, Pekka Savola, Francis Dupont, Brian Haberman, Tatuya Jinmei, et Margaret Wasserman pour leurs commentaires détaillés.
l
Rich Draves et Thomas Narten étaient les auteurs de la RFC 3041. Ils souhaitent remercier de leurs contributions le groupe de travail IPv6 et, en particulier, Ran Atkinson, Matt Crawford, Steve Deering, Allison Mankin, et Peter Bieringer.

Adresse des auteurs

Fernando Gont
SI6 Networks
Segurola y Habana 4310, 7mo Piso
Villa Devoto
Ciudad Autonoma de Buenos Aires
Argentina
mél : fgont@si6networks.com
URI : https://www.si6networks.com

Richard Draves
Microsoft Research
One Microsoft Way
Redmond, WA
United States of America
mél : richdr@microsoft.com

Suresh Krishnan
Kaloom
mél : suresh@kaloom.com

Thomas Narten
mél : narten@cs.duke.edu