

RFC 3971 : SEcure Neighbor Discovery (SEND)

Stéphane Bortzmeyer

<stephane+blog@bortzmeyer.org>

Première rédaction de cet article le 15 septembre 2009

Date de publication du RFC : Mars 2005

<https://www.bortzmeyer.org/3971.html>

Pour trouver l'adresse MAC d'un voisin, une machine IPv6 utilise le protocole NDP ("*Neighbor Discovery Protocol*"), normalisé dans le RFC 4861¹. Ce protocole ressemble beaucoup au traditionnel ARP (RFC 826) d'IPv4 et il partage notamment sa principale vulnérabilité : une machine peut tout à fait se faire passer pour une autre et recevoir ainsi les paquets qui ne lui sont pas destinés. En outre, NDP dispose d'autres fonctions qu'ARP, comme la découverte des routeurs, fonctions qui sont tout aussi critiques et tout aussi vulnérables. Au début d'IPv6, l'enthousiasme pour IPsec était sans limite et il était donc censé résoudre ce problème, comme beaucoup d'autres. En réalité, IPsec a été peu implémenté et encore moins déployé et il a donc fallu trouver une autre solution, SEND ("*SEcure Neighbor Discovery*") qui fait l'objet de ce RFC.

Les détails sur les vulnérabilités de NDP figurent dans le RFC 3756. Ils sont aussi résumés dans la section 11.1 du RFC 4861 et rappelés dans la section 1 de notre RFC 3971. Exemple ironique, à chaque réunion physique de l'IETF, certains portables sont configurés pour créer des réseaux ad hoc et envoient des annonces RA ("*Router Advertisement*") concurrents de ceux du routeur légitime. La seule solution est de donner au micro les adresses MAC des routeurs « pirates » pour les filtrer.

Cette section 1 explique aussi, en plus des problèmes généraux d'IPsec, pourquoi il ne peut pas être utilisé pour sécuriser un protocole de découverte des voisins et du réseau, en raison des limites du "*bootstrap*" (on a besoin du réseau pour IKE mais on a besoin d'IPsec pour initialiser le réseau..).

Sur un réseau local qui n'offre pas de sécurité physique (Ethernet sans 802.1x ou bien WiFi), NDP est très vulnérable et c'est la raison d'être de SEND.

Résumé brièvement, SEND consiste à utiliser des adresses CGA (adresses auto-signées) sur les machines non-routeuses et des signatures cryptographiques avec certificat pour authentifier les annonces des routeurs.

La section 3 résume les fonctions de NDP :

1. Pour voir le RFC de numéro NNN, <https://www.ietf.org/rfc/rfcNNN.txt>, par exemple <https://www.ietf.org/rfc/rfc4861.txt>

- Découverte automatique des routeurs (une autre solution étant d'utiliser DHCP),
- Auto-configuration, sans état, des adresses IP des machines (RFC 4862),
- Détection des conflits d'adresses (lorsque deux machines ont reçues la même adresse IP),
- Résolution d'adresses IP en adresses MAC (section 7.2 du RFC 4861).

Toutes ces fonctions sont mises en œuvre par des messages portés sur ICMP (RFC 4443) qui contiennent des données et des options encodées en TLV. (Voir la liste dans le registre IANA <<https://www.iana.org/assignments/icmpv6-parameters>>.)

Comment fonctionne SEND? C'est l'objet du reste du RFC, à commencer par la section 4 qui décrit le fonctionnement général. SEND comprend :

- Un système de certificats, permettant de valider les annonces d'un routeur, en suivant un chemin de certificats signés (l'annonce du routeur n'a pas besoin d'inclure tous les certificats du chemin, des nouveaux messages NDP permettent au récepteur de demander les certificats qui lui manquent).
- Des adresses cryptographiques, les CGA du RFC 3972, permettent de s'assurer que la machine qui répond aux requêtes "*Neighbor Discovery*" est bien la « propriétaire » de l'adresse. (Donc, contrairement aux annonces des routeurs, SEND n'utilise pas de certificat pour authentifier les simples machines. Cette possibilité est uniquement mentionnée pour un usage possible dans le futur.)
- Une nouvelle option NDP, *RSA signature*, permet de vérifier l'intégrité du message. La clé publique utilisée pour la signature est récupérée, soit par les certificats, si l'émetteur est un routeur, soit, pour le cas de CGA par une autre option qui permet d'inclure cette clé dans les messages. À noter que RSA est le seul algorithme normalisé, pour garantir l'interopérabilité (puisque SEND tourne dans des environnements où il ne serait pas réaliste de se lancer dans des négociations entre émetteur et récepteur). S'il fallait changer d'algorithme, par exemple suite aux progrès de la cryptanalyse, il faudrait alors développer une nouvelle version de SEND (ce qui est précisément en cours).
- Et enfin un mécanisme de "*nonce*" contre les attaques par rejeu.

Le détail de ces nouvelles options et de leur format figure en section 5. Par exemple, l'option CGA (section 5.1) permet au répondeur qui utilise des adresses IP CGA d'inclure la clé publique utilisée dans la réponse. La section 5.1.1 détaille comment l'émetteur doit remplir cette option et la 5.1.2 comment le récepteur peut l'utiliser pour vérifier que le détenteur de la clé est bien celui de l'adresse

La section 5.2, elle, normalise l'option CGA *signature*, où le message SEND contient une signature au format PKCS#1. Là encore, la section décrit en détail ce que doit faire l'émetteur pour remplir cette option et le récepteur pour la vérifier. Toute aussi indispensable, la section 5.2.3 traite de la configuration de l'option par l'administrateur système et note qu'une mise en œuvre de SEND doit pouvoir être configurée pour vérifier cette signature par différents moyens, comme une "*trust anchor*", une clé publique de départ, à laquelle on fait confiance et qui peut à son tour signer d'autres clés.

Un enjeu de sécurité important de SEND est la protection contre le rejeu. Pour cela, les options *Timestamp* et *Nonce*, décrites en section 5.3 sont indispensables. *Timestamp* indique le temps de l'émetteur et permet, si les horloges de l'émetteur et du récepteur sont raisonnablement synchronisées (la section 5.3.4.2 précise les tolérances admises), d'empêcher qu'un méchant n'enregistre une réponse tout à fait légitime et ne la réutilise plus tard, alors qu'elle n'est plus valable. *Nonce*, elle, stocke un nombre aléatoire (de taille variable, au moins 48 bits) envoyé par le solliciteur et retourné par le répondant. Ainsi, on peut être sûr que le paquet SEND est bien une réponse à une question qu'on avait posé. Les règles de génération imposent l'usage de ces deux options (et les récepteurs doivent considérer les messages SEND sans une de ces options comme étant non-SEND donc non sûrs).

La section 6 entreprend de résoudre le difficile problème de la validation des certificats des routeurs. La méthode normale, si on n'a pas déjà l'information dans son magasin de certificats, est d'aller la chercher sur l'Internet. Mais, ici, comme il s'agit de découvrir le routeur, cette solution n'est pas possible

puisque, justement, on n'a pas encore de routeur pour aller sur l'Internet. Comment SEND traite-t-il ce problème ?

Le modèle général est celui de X.509, comme l'utilisent, par exemple, les navigateurs Web. Chaque machine doit avoir un magasin de certificats (qui ne sont pas forcément les mêmes que ceux utilisés, après la configuration d'IP, pour des activités comme le Web) et se sert de ce ou ces certificats comme "*trust anchors*", comme points de départ de la validation (sections 6.1 et 6.5). La machine terminale n'a pas de certificats à elle (à part avec CGA, elle ne s'authentifie pas, seul le routeur le fait). Le routeur, lui, reçoit son certificat et sa clé privée. Le certificat l'autorise non seulement à agir comme routeur mais également à annoncer certains préfixes IP et pas les autres.

Qui attribue ces certificats ? La section 6.2 mentionne une autorité de certification mondiale unique pour les routeurs... qui n'existe pas (et n'est sans doute pas près d'exister). Elle pourrait émaner de l'IANA ou bien d'un consortium formé par les RIR.

Une autre solution mentionnée par le RFC (nommée dans le RFC le « modèle décentralisé », un terme incorrect), et la seule possible aujourd'hui, est celle d'un nuage de diverses autorités de certification, chacune reconnue par certains sites mais pas par d'autres. Cela rend très difficile la mobilité (un portable en visite sur un autre site n'a aucun moyen de valider les annonces SEND du routeur local).

Le format des certificats, lui, est expliqué en section 6.3. C'est du X.509 v3, avec le profil des RFC 3779 et RFC 5280, RFC qui permet de décrire dans le certificat les ressources Internet qu'on peut annoncer (les adresses IP, par exemple, section 4.2.1.6 du RFC 5280).

Enfin, pour permettre la réception de certificats non contenus dans le magasin, mais nécessaires pour boucler la chaîne de validation, la section 6.4 prévoit des types de messages ICMP *Certificat Path Solicitation* et *Certification Path Advertisement*. Avec elles, on peut construire des messages pour demander ou obtenir des certificats.

L'usage des adresses IP par SEND fait l'objet de la section 7. Une machine SEND est censée n'utiliser que CGA (section 7.1) et donc pas les adresses IP temporaires du RFC 4941. Sécurité ou vie privée, il faut choisir ! Pour travailler avec d'autres adresses, rien n'est encore trop précisé, à part l'API du RFC 5014. Autre exemple (section 7.4), SEND ne fournit aucun moyen d'authentifier les messages NDP pour les adresses IPv6 statiques, ou bien les adresses autoconfigurées à partir de l'adresse MAC.

SEND est aujourd'hui très peu déployé. Comme tous les protocoles récents (il date quand même de plus de quatre ans), il fait face à des problèmes de transition avec l'ancienne méthode (qui est de ne pas sécuriser du tout). La section 8 expose ces problèmes. Comme, en pratique, une machine SEND va certainement rencontrer des réseaux non-SEND, le RFC demande que les messages non sécurisés soient également acceptés. La machine SEND qui reçoit deux sortes de messages doit évidemment donner la priorité à ceux qui sont sécurisés. Le RFC recommande également une option de configuration « paranoïaque » où seuls les messages NDP sécurisés soient acceptés (à condition qu'elle ne soit pas active par défaut). Inversement, le RFC admet une option de ne pas envoyer de messages SEND (au cas où ils perturbent le réseau mais, si tout le monde suit la norme IPv6 correctement, cela ne devrait pas arriver, les options SEND des messages NDP devraient simplement être ignorées) mais cette option doit être activée explicitement.

La cohabitation de réseaux SEND (sécurisés) et non-SEND (non sécurisés) peut avoir pour conséquence la réception d'un message non sécurisé qui essaie de remplacer l'information issue d'un message sécurisé. Notre RFC 3971 impose que les caches (comme celui des voisins qui, sur Linux, peut être affiché avec `ip`

-f inet6 neighbour show) doivent garder trace du caractère sûr ou non sûr de l'information et ne doivent pas remplacer une information sûre par une non sûre.

SEND est entièrement voué à la sécurité donc l'obligatoire section Sécurité pourrait être inutile mais, ici, la section 9.1 couvre un problème utile : les menaces que SEND ne résout pas. Ainsi, SEND ne protège pas la confidentialité des échanges NDP. SEND ne rend pas complètement sûr un réseau physique non sûr (par exemple, une machine malveillante peut toujours envoyer des paquets avec une adresse MAC source mensongère).

À l'heure actuelle, un groupe de travail de l'IETF, csi <<http://tools.ietf.org/wg/csi>>, travaille à améliorer SEND. Par exemple, SEND est limité à SHA1 et RSA alors que le groupe de travail explore la meilleure façon d'utiliser de nouveaux algorithmes cryptographiques.

Quelles mises en œuvres de SEND sont disponibles ? DoCoMo en avait une mais qui n'est plus maintenue. On trouve quelques bouts de code par ci par là, quelques projets pas finis <<http://code.google.com/p/ipv6-send-cga/>>, mais apparemment pas d'implémentation libre complète qui marche sur des systèmes comme Linux ou FreeBSD. Donc, je ne crois pas que je pourrai activer SEND sur ma Freebox. En revanche, Cisco a une mise en œuvre de SEND <http://www.cisco.com/en/US/docs/ios/ipv6/configuration/guide/ip6-first_hop_security.html> sur ses routeurs (mais pas, semble t-il, Juniper).

Pourquoi SEND n'est-il pratiquement pas déployé ? Il y a sans doute plusieurs raisons : le déploiement est difficile puisqu'il faut mettre la clé publique des routeurs dans **chaque** machine et la sécurité du protocole NDP n'est pas vraiment un problème aujourd'hui (à part sur le réseau de Defcon, les attaques réelles sont rares).