RFC 6862 : Keying and Authentication for Routing Protocols (KARP) Overview, Threats, and Requirements

Stéphane Bortzmeyer
<stephane+blog@bortzmeyer.org>

Première rédaction de cet article le 2 mars 2013

Date de publication du RFC : Mars 2013

https://www.bortzmeyer.org/6862.html

Le groupe de travail KARP <http://tools.ietf.org/wg/karp> de l'IETF travaille sur une partie bien spécifique du vaste programme de la sécurisation du routage sur l'Internet : il s'occupe de l'authentification réciproque des routeurs et de comment l'améliorer. Ce RFC fait l'analyse des menaces pesant sur cette authentification et expose le cahier des charges pour les nouveaux RFC qui vont s'attaquer à ces menaces (le premier RFC du groupe KARP, le RFC 6518 [1], le recouvre partiellement mais se focalisait sur les tâches à long terme à accomplir).

Ce RFC 6862 ne spécifie pas un nouveau protocole, il sert de base au travail concret que le groupe KARP a désormais entamé. D'abord, l'analyse des menaces : comme le documente la section 8.1 du RFC 4948 (le compte-rendu d'un atelier sur le trafic Internet non sollicité), la meilleure solution pour un méchant qui veut perturber l'activité des autres est souvent de s'attaquer aux routeurs. Le RFC 4593 détaillait les attaques contre les protocoles de routage, et le RFC 6039 dénonçait le fait que les mécanismes de sécurité existant dans les routeurs étaient bien primitifs, par exemple parce qu'ils ne permettaient pas de changer les clés sans perturber sérieusement le fonctionnement du réseau (ce qui casserait les SLA), avec le résultat que bien des opérateurs ne changent jamais ces clés. (On peut aussi consulter le RFC 4949, un glossaire de la sécurité sur l'Internet.)

Quels sont les protocoles de routage concernés ? BGP, OSPF, IS-IS, LDP et RIP ont tous déjà des mécanismes cryptographiques de protection de la communication entre routeurs. Le but du projet KARP est d'améliorer ces mécanismes, pour lesquels il existe déjà du code qui tourne et de l'expérience opérationnelle, pas de les jeter et de les remplacer (principe dit « "*crawl, walk, run*" <http://www.atworkmgt.nl/Upload/cwr5%205.jpg> », autrement dit, ramper lorsqu'on n'a pas le choix, marcher si on peut,

1. Pour voir le RFC de numéro NNN, https://www.ietf.org/rfc/rfcNNN.txt, par exemple https://www.ietf.org/rfc/rfc6518.txt

courir dans le meilleur des cas ; c'est tiré d'une citation célèbre de Martin Luther King). KARP est donc un projet **incrémental** (section 2.2). Parmi les raisons qui ne permettent pas d'envisager un passage à court terme vers des solutions radicalement différentes, avec de la cryptographie forte, partout, il y a le fait que les routeurs actuellement déployés ont des capacités matérielles très variables, par exemple, ils n'ont pas forcément d'accélérateur cryptographique. D'une manière générale, KARP essaie d'être réaliste et de coller aux témoignages des opérateurs, tels qu'exprimés dans le rapport Arbor <http://pages.arbornetworks.com/rs/arbor/images/ISR2008_EN.pdf>.

Ce rapport note, par exemple, qu'aucun des opérateurs interrogés n'utilise IPsec entre ses routeurs (alors qu'il résoudrait un bon nombre des problèmes de sécurité de KARP). Par contre, 70 % utilisent les mots de passe MD5 sur les sessions BGP externes et 55 % sur les internes. Les clés sont très peu changées (des clés de plus de cinq ans sont courantes), en raison des conséquences (en BGP, changer la clé coupe la session, faisant osciller les routes) et des difficultés opérationnelles que cause leur changement (il faut se concerter avec le pair dont, bien souvent, on n'a plus les coordonnées, puis organiser le changement pile au même moment). Tout ceci pour un ROI nul. La décision du PHB est donc vite prise : on ne change pas les clés.

Le rapport Arbor cite même un opérateur qui utilisait la même clé pour tous ses clients et, à ceux qui s'inquiétaient de la sécurité et demandaient une clé spécifique, répondait que ce n'était pas possible. C'est gonflé, mais cela simplifiait beaucoup la vie de cet opérateur.

Ce travail de KARP, comme décrit dans le RFC 6518, aura deux phases :
— La phase 1 portera sur cette amélioration de l'existant,
— La phase 2 portera sur un protocole de gestion de clés.

Un bon exemple d'amélioration par déploiement incrémental a été pour BGP, où l'antédiluvien mécanisme de sécurité du RFC 2385 a d'abord été remplacé par l'AO ("*Authentication Option*") du RFC 5925, alors même qu'il n'existait pas encore de protocole de gestion de clés, que plusieurs cryptographes estimaient indispensable. AO résout une partie des problèmes des mots de passe MD5 (notamment l'agilité cryptographique, la possibilité de changer d'algorithme) mais n'essaie pas de tout résoudre d'un coup.

Un peu de vocabulaire au passage : une KDF ("*Key Derivation Function*") est une fonction qui génère une clé à partir d'une autre clé et de données diverses. Par exemple, une KDF possible est de concaténer la clé de départ (clé de dérivation) et une adresse IP, puis de condenser cryptographiquement le tout.

Un KMP ("*Key Management Protocol*", le protocole qui fera l'objet de la phase 2) est un protocole de gestion de clés cryptographiques, permettant leur distribution aux routeurs. Le but d'un KMP est de permettre le changement relativement fréquent des clés (à l'heure actuelle, le changement des clés dans les protocoles de routage est manuel : comme c'est une opération pénible, elle n'est en général pas faite et les routeurs gardent en général les mêmes clés toute leur vie).

Une PSK ("*Pre-Shared Key*") est une clé secrète partagée entre N routeurs, par exemple les mots de passe d'OSPF v2 ou bien les « mots de passe MD5 » de BGP. En l'absence de KMP, elles sont distribuées manuellement (copier-coller dans un terminal virtuel, par exemple).

Une bonne analyse de sécurité nécessite d'être conscient des capacités de l'adversaire. Première catégorie, les attaquants extérieurs à l'organisation ("*outsiders*", section 3.1.1). Ils n'ont pas les clés. Notre RFC fait une distinction importante entre les attaquants extérieurs mais situés sur le chemin entre deux routeurs ("*on-path attackers*") et ceux situés en dehors ("*off-path attackers*"). Les premiers peuvent écouter le trafic et injecter des paquets qui seront indistinguables des vrais. Les seconds ne peuvent pas écouter le trafic, peuvent parfois injecter des paquets mais, comme ils travaillent en aveugle, ils auront beaucoup plus de mal à les faire accepter comme authentiques. Le méchant situé sur le chemin (le plus dangereux) peut donc :

— rejouer des vieux paquets (qui pouvaient être signés pour l'authentification),
— insérer des paquets,
— changer des paquets.

Les méchants extérieurs, qu'ils soient sur le chemin ou pas, peuvent aussi tenter de découvrir un mot de passe par force brute, en essayant de se connecter avec plein de mots de passe courants. Ou essayer de découvrir les clés par des essais systématiques (qui peuvent réussir si la clé est trop courte.)

Deuxième catégorie, les attaquants qui ont réussi à avoir une clé (section 3.1.2). Un exemple typique est un ex-employé mécontent (le taux de rotation du personnel est élevé dans ce métier). C'est à cause d'eux qu'il faut changer régulièrement les clés : on ne peut jamais être 100 % sûr que la clé n'a pas fuité et il est donc nécessaire de ne pas l'utiliser éternellement.

Enfin, il y a les byzantins, les pires (section 3.1.3). Ce sont les participants légitimes qui, pour une raison ou une autre, trahissent. Cela peut être un routeur bogué, qui émet des paquets incorrects. Ou un routeur piraté par le méchant. Comme le projet KARP vise à une meilleure authentification des pairs, le cas d'un pair légitime, mais traître, n'est pas traité dans le cadre de ce projet. (Idem pour un routeur qui jouerait correctement les protocoles de routage, mais jetterait ensuite les paquets, créant un trou noir.)

Avec de telles capacités, même les autres attaquants peuvent causer beaucoup de dégâts. À noter que les attaques par déni de service utilisant les protocoles de routage font partie des menaces considérées par KARP, par exemple des paquets signés ou chiffrés qui entraînent des calculs cryptographiques lourds. Les attaques par déni de service volumétriques (pure force brute, on envoie plein de paquets ordinaires) ne sont par contre pas prises en compte par KARP. D'autres attaques par déni de service sont des objectifs de KARP. Par exemple, si on faisait tourner BGP sur TLS, pour le sécuriser, un attaquant sur le chemin pourrait toujours envoyer des paquets TCP `RST` (`ReSeT`) qui couperaient la connexion. KARP vise à empêcher aussi ce genre d'attaques. (Si l'attaquant en question n'est pas sur le chemin, et travaille donc en aveugle, cette attaque sera plus compliquée pour lui, voir le RFC 5961.)

Il y aussi trois « non-objectifs », que KARP n'essaie même pas d'atteindre :
— Protection de la vie privée. Les sniffeurs purement passifs peuvent apprendre des choses, mais pas perturber le routage. Les sections 2.1 et 3.3 rappellent donc que la confidentialité est un non-but.
— Vérification de la validité des messages. En effet, faire confiance à un routeur n'est pas la même chose que faire confiance au contenu de ses mises à jour. Les attaques comme celle de Pakistan Telecom <https://www.bortzmeyer.org/pakistan-pirate-youtube.html> dépendent donc d'un autre groupe de travail <https://www.bortzmeyer.org/securite-routage-bgp-rpki-roa.html>.
— La non-répudiation des paquets par les routeurs.

Et les solutions ? KARP se propose de mieux sécuriser les communications entre routeurs (il existe déjà des tas de techniques, mais imparfaites). Cette amélioration doit être incrémentale (pas question de faire de la table rase).

Le cahier des charges du travail de KARP (la « phase 1 », l'amélioration de l'existant, est déjà largement entamée) est donc ici, sous forme d'une longue liste d'exigences (section 4 du RFC) :
— Une meilleure description de ce qu'on protège,
— Des algorithmes cryptographiques sûrs (MD5 doit être abandonné),
— Une agilité de ces algorithmes, permettant leur remplacement (la question n'est pas de savoir **si** un algorithme cryptographique sera cassé mais **quand**),
— Changement des clés ("*rekeying*") sans casser les sessions en cours,
— Mécanismes pour une utilisation plus sûre des PSK (secrets partagés),
— Une sécurité par pair (et pas globale),
— Possibilité d'utiliser aujourd'hui la gestion manuelle des clés (la seule réaliste actuellement) tout en permettant l'usage futur d'un KMP (protocole de gestion automatique des clés),
— La compatibilité avec l'existant, si on veut avoir la moindre chance que ce soit déployé un jour.