RFC 7211 : Operations Model for Router Keying

Stéphane Bortzmeyer
<stephane+blog@bortzmeyer.org>

Première rédaction de cet article le 19 juin 2014

Date de publication du RFC : Juin 2014

https://www.bortzmeyer.org/7211.html

Il y a un gros travail en cours à l'IETF pour améliorer la sécurité du routage sur l'Internet. L'un des aspects de ce travail concerne la gestion des **clés** par les routeurs. Bien des solutions de sécurité nécessitent que les routeurs disposent de clés secrètes. Le groupe de travail KARP <http://tools.ietf.org/wg/karp> travaille sur les problèmes de gestion de ces clés (cf. RFC 6518[1]). Ce nouveau RFC du groupe décrit les problèmes opérationnels et les pratiques actuelles de gestion des clés dans les routeurs.

KARP a déjà produit plusieurs RFC </search?pattern=karp%26(routage|routeur)>. Celui-ci se veut particulièrement terre-à-terre. Il existe des tas de techniques pour sécuriser la communication entre routeurs. Mais comment sont-elles effectivement déployées ? Pour le savoir, il faut disposer d'un modèle décrivant les opérations de gestion de clés. C'est d'autant plus crucial que la sécurisation du routage présente quelques défis spécifiques. Par exemple, comme il faut que le routage fonctionne pour pouvoir contacter d'autres machines du réseau, les solutions fondées sur un serveur d'authentification central ne conviennent pas.

D'abord, le modèle de configuration d'un routeur, en section 3. La configuration consiste à indiquer au routeur les clés qu'il utilisera pour s'authentifier, ainsi que divers paramètres. Cela peut se faire manuellement ou automatiquement. La configuration automatique étant encore un sujet d'étude, notre RFC se concentre sur la configuration manuelle, via une structure abstraite, la **table des clés**. Ce n'est déjà pas évident car certains protocoles nécessitent une clé commune à tous les routeurs, et d'autres pas. Il faut donc pouvoir configurer les clés par interface réseau et par pair. Prenons l'exemple d'OSPF (RFC 2328) : sur tout lien réseau, tous les routeurs doivent utiliser la même clé. En revanche, d'un lien à l'autre, un même routeur peut se trouver à utiliser plusieurs clés différentes. Mais ce n'est pas obligatoire et, dans certaines organisations, on aura peut-être une clé pour chaque zone ("*area*") OSPF, avec seuls les

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

ABR ("*Area Border Router*") utilisant plusieurs clés. Bien sûr, avec une table de clés OSPF où les clés sont rangées par lien, on peut réaliser la politique « une clé par zone » en mettant la même clé pour tous les liens de la zone. Mais, si ce n'est pas vérifié par le système de gestion de la table de clés, il y a un risque de désynchronisation, par exemple parce qu'on change la clé d'un lien en oubliant les autres liens de la zone. Il serait donc souhaitable d'avoir un mécanisme d'héritage (« la clé du lien dépend de la zone ») permettant d'éviter cette désynchronisation accidentelle.

L'intégrité de la table des clés par rapport aux règles techniques des protocoles de routage est évidemment essentielle. Le routeur doit donc s'assurer que les algorithmes de cryptographie indiqués sont compatibles avec le protocole de routage, que la fonction de dérivation l'est également, etc. La table peut être modifiée via une quantité de méthodes (système de gestion de configuration, interface Web, CLI, etc) du moment que son intégrité est vérifiée.

Afin d'éviter les « clés éternelles », jamais changées (même lorsqu'un employé qui les connait quitte l'organisation), il faut aussi que le table prévoit un système d'expiration, pour pouvoir indiquer « cette clé est valable jusqu'au 15 avril 2014 » et qu'elle soit automatiquement marquée comme expirée après cette date.

Il existe des tas de façons d'utiliser des clés en cryptographie. La plus simple (et sans doute la plus courante aujourd'hui) est que les clés configurées dans le routeur soient partagées, et utilisées telles quelles dans le protocole de routage. Mais on peut aussi passer par une fonction de dérivation, qui va combiner la clé partagée avec des paramètres d'une session particulière, rendant ainsi la tâche plus difficile pour l'attaquant (même s'il a la clé, il ne pourra pas forcément s'insérer dans une session en cours). C'est par exemple ce que fait le protocole AO du RFC 5925 (dans sa section 5.2). Ou bien on peut se servir des clés préconfigurées pour s'authentifier le temps de générer des clés de session, par exemple avec Diffie-Hellman. Les clés sont parfois identifiées par un nombre court, parfois pas identifiées du tout et un routeur peut avoir à en essayer plusieurs avant de tomber sur la bonne.

Enfin, il y a le cas où les routeurs utilisent de la cryptographie asymétrique. La clé privée de chaque routeur ne quittera alors jamais le routeur, ce qui augmente la sécurité. (Ceci dit, cela empêchera de remplacer rapidement un routeur, ce qui est une opération souhaitable, et qui justifie donc une gestion centrale des clés.) L'authentification des autres routeurs pourra se faire via une PKI (ce qui peut être compliqué à mettre en place), ou bien en ajoutant la clé publique de chaque routeur à tous ses pairs (ce qui n'est pas pratique si on ajoute souvent de nouveaux routeurs). Pour identifier les clés, notre RFC recommande de se servir du RFC 4572 (section 5). Ensuite, les autorisations de tel ou tel routeur pair peuvent se faire en indiquant les clés (ce qui a l'inconvénient qu'il faut les refaire si un routeur change de clé) ou en indiquant une identité du routeur (ce qui nécessite, pour cette indirection supplémentaire, un moyen sécurisé de passer de l'identité du routeur à sa ou ses clés). Avec une PKI, le problème ne se posera évidemment pas.

On l'a vu, un problème spécifique à la sécurisation du routage est qu'il faut que le mécanisme marche sans dépendre du réseau puisque, avant que les routeurs soient authentifiés et échangent les routes, on n'aura peut-être pas de réseau. Cela limite sérieusement le rôle que peuvent jouer des serveurs centraux, genre RADIUS (RFC 2865). Pas question par exemple que le seul moyen d'obtenir les clés des pairs soit de demander à un serveur central. Cela ne veut pas dire que ces serveurs centraux soient inutiles. Ils peuvent aider à des opérations comme la diffusion de nouvelles clés en cas de préparation à un remplacement de clés ou comme la distribution de la clé d'un nouveau pair. Simplement, le réseau doit pouvoir démarrer et fonctionner sans eux.

La section 6 de notre RFC est consacrée au rôle de l'administrateur humain dans la gestion des clés. Deux exemples typiques où il est indispensable, l'arrivée d'un nouveau pair (il faut décider si on le laisse entrer ou pas) et la réparation en cas de problème. Si le réseau fonctionne déjà, l'arrivée d'un nouveau

pair peut être simplifiée car il peut suffire d'inscrire ce routeur dans une base de données et toutes les informations nécessaires peuvent être propagées aux autres routeurs en utilisant le réseau. Le cas d'un réseau tout nouveau démarrant à froid est évidemment plus compliqué.

Naturellement, le nouveau routeur doit être proprement configuré sur le plan de la sécurité ; il ne servirait à rien de protéger les clés utilisées par les protocoles de routage si on pouvait se connecter en SSH sur le routeur avec `login: admin password: admin` !

Et les pannes ? C'est que la sécurité apporte ses propres causes de panne. Vouloir sécuriser un réseau peut l'empêcher de fonctionner. Ainsi, l'expiration d'un certificat entraîne son refus par les autres routeurs et le retrait du routeur au certificat trop vieux... Même si le RFC n'en parle pas, une des raisons du faible déploiement des solutions de sécurisation du routage est la crainte d'avoir davantage de pannes. Le déploiement de toute solution de sécurité nécessitera des pratiques plus rigoureuses, par exemple une supervision des dates d'expiration des certificats.

Et une fois qu'on a décidé de déployer une jolie sécurité toute neuve avec une meilleure gestion des lettres de créance des routeurs, comment on fait ? La section 7 fait remarquer que le passage du système actuel au nouveau ne va pas forcément être sans douleur. Si on conçoit un réseau tout neuf, on peut le faire proprement dès le début. Mais si on a un réseau existant et qu'on veut y introduire de nouvelles pratiques ? Et sans tout casser ? Aujourd'hui, si des clés comme celles d'OSPF ou bien celles utilisées entre routeurs BGP sont très rarement changées, c'est souvent par peur que ce changement coupe le routage. Pour deux routeurs BGP appartenant à des organisations différentes, le changement du mot de passe MD5 (RFC 2385) doit être soigneusement coordonné et il faut que le pair respecte rigoureusement la procédure qui a été définie, notamment le moment du changement. Comme on s'en doute, c'est rarement fait.

Prenons l'exemple du nouveau mécanisme AO (RFC 5925) qui remplace l'authentification MD5. Il n'existe pas de mécanisme automatique pour savoir si le pair BGP utilise MD5 ou AO. Il faut donc un changement coordonné (« le 11 mars à 13h37 UTC, on passe à AO »). Si le logiciel utilisé permet de grouper les configurations en traitant tout un ensemble de pairs de la même façon, il faut en prime faire cela pour tous les pairs à la fois. Vu cette difficulté, il n'est pas étonnant que peu de sessions BGP soient aujourd'hui protégées par AO.

Enfin, le rappelle la section 8, beaucoup de mécanismes de sécurité dépendent d'une horloge à l'heure. Il est donc crucial de s'assurer que l'heure sur les routeurs est correctement synchronisée.