RFC 8886 : Secure Device Install

Stéphane Bortzmeyer
<stephane+blog@bortzmeyer.org>

Première rédaction de cet article le 6 octobre 2020

Date de publication du RFC : Septembre 2020

https://www.bortzmeyer.org/8886.html

Quand vous êtes opérateur réseau et que vous avez à installer une nouvelle machine, par exemple un routeur, dans un centre de données lointain, vous avez deux solutions : envoyer un employé faire l'installation sur place, ce qui peut être coûteux, ou bien demander à la société qui gère le centre de données de le faire pour vous, ce qui n'est pas forcément génial question sécurité, puisque, selon la façon dont c'est fait, ils auront peut-être la possibilité de regarder ou de bricoler la configuration de la machine. Ce nouveau RFC propose un moyen simple d'améliorer la sécurité dans le deuxième cas, en chiffrant un fichier de configuration avec la clé publique de la machine. Celle-ci pourra le déchiffrer après téléchargement, en utilisant sa clé privée. (Cette solution nécessitera donc un changement des équipements pour que chacun ait une paire clé publique / clé privée.)

Précisons la question à résoudre. Ce problème est une variante d'un problème bien classique en informatique, celui du "*bootstrap*". Par exemple, si la nouvelle machine est protégée par un mot de passe, il faut bien rentrer ce mot de passe la première fois. Si c'est un employé du centre de données qui le fait, il connaitra le mot de passe. Et si on le change juste après ? Rien ne dit qu'il n'aura pas laissé une porte dérobée. Il n'y a pas de solution générale et simple à ce problème. Notre RFC se contente d'une solution partielle, qui améliore la sécurité.

On va donc s'imaginer à la place de la société Machin, un gros opérateur réseau dont l'essentiel des équipes est à Charleville-Mézières (pourquoi pas ?). La société a des POP dans beaucoup d'endroits, et la plupart du temps dans ces centres de données qu'elle ne contrôle pas, où elle loue juste de l'espace. Comme les équipements réseau tombent parfois en panne, ou, tout simplement, ne parviennent plus à gérer le trafic toujours croissant, la société doit de temps en temps installer de nouveaux matériels, routeurs, commutateurs, etc. Si le centre de données où il faut installer est proche de Charleville-Mézières, tout va bien : un employé de l'opérateur Machin se rend sur place, le fabricant du routeur a déjà livré la nouvelle machine, il n'y a plus qu'à la "*racker*" et à la configurer (« "*Enter the password for the administrative account*" ») ou, en tout cas, faire une configuration minimale qui permettra de faire ensuite le reste via SSH, RESTCONF (RFC 8040 [1]) ou une autre technique d'administration à distance.

1. Pour voir le RFC de numéro NNN, https://www.ietf.org/rfc/rfcNNN.txt, par exemple https://www.ietf.org/rfc/rfc8040.txt

Mais supposons que la machine à installer, mettons que ce soit un gros routeur, doive être placé chez un centre de données éloigné, mettons Singapour (exemple choisi parce que c'est vraiment loin). Même avant la pandémie de Covid-19, un voyage à Singapour n'était guère raisonnable, du point de vue du temps passé, et du point de vue financier. Mais les centres de données ont tous un service d'action à distance ("*remote hands*") où un employé de la société qui gère le centre peut appuyer sur des boutons et taper des commandes, non ? C'est en effet très pratique, mais ce service est limité à des cas simples (redémarrer…) et pas à une configuration complète. Celle-ci poserait de toute façon un problème de sécurité car l'employé en question aurait le contrôle complet de la machine pendant un moment.

(Une autre solution, apparemment non mentionnée dans le RFC, est que le fabricant du routeur livre à Charleville-Mézières l'appareil, qui soit alors configuré puis envoyé à Singapour. Mais cela impose deux voyages.)

Bon, et une solution automatique ? La machine démarre, acquiert une adresse IP, par exemple par DHCP (RFC 2131 et RFC 8415), et charge sa configuration sous forme d'un fichier récupéré, par exemple avec TFTP (RFC 1350). Mais cette configuration peut contenir des informations sensibles, par exemple des secrets RADIUS (RFC 2865). TFTP n'offre aucune sécurité et ce n'est pas très rassurant que de laisser ces secrets se promener en clair dans le centre de données singapourien. Il existe des contournements variés et divers pour traiter ces cas, mais aucun n'est complètement satisfaisant.

Bref, le cahier des charges de notre RFC est un désir de confidentialité des configurations du nouveau routeur. (Il y a une autre question de sécurité importante, l'authenticité de cette configuration, mais elle n'est pas traitée dans ce RFC. Des solutions, un peu complexes, existent, comme SZTP - RFC 8572 ou BRSKI - RFC 8995.) La solution présentées dans notre RFC n'est pas complète. Elle améliore les procédures d'installation (voir par exemple la documentation de celle de Cisco `<https://www.cisco.com/c/en/us/td/docs/ios-xml/ios/fundamentals/configuration/15mt/fundamentals-15-cf-autoinstall.html>`) mais ne prétend pas résoudre tout le problème.

Ce RFC ne spécifie pas un protocole, mais décrit une méthode, que chaque constructeur d'équipements réseau devra adapter à ses machines et à leurs particularités. Cette méthode est conçue pour ces équipements (routeurs et commutateurs, notamment), mais pourrait être adaptée à des engins comme les serveurs. Mais en attendant, ce RFC se focalise sur le matériel réseau.

La section 2 du RFC résume la solution. Elle nécessite que la machine à configurer ait un mécanisme de démarrage automatique de l'installation ("*autoinstall*" ou "*autoboot*" dans les documentations), permettant d'obtenir une adresse IP (typiquement avec DHCP) puis de récupérer un fichier de configuration (typiquement avec TFTP mais cela peut être HTTP ou un autre protocole de transfert de fichiers, les options DHCP 66 - "*Server-Name*" - ou 150 - "*TFTP server address*" pouvant être utilisées pour indiquer le serveur). Le nom du fichier est par exemple issue de l'option DHCP 67 - "*Bootfile-Name*".

La méthode nécessite un identificateur, comme le numéro de série ou bien l'adresse MAC de l'équipement réseau qui est en cours d'installation. (Rien n'interdit d'utiliser un autre identificateur, par exemple un UUID - RFC 4122, mais l'avantage du numéro de série est qu'il est en général marqué sur la machine et facilement accessible, donc la personne dans le centre de données à Singapour peut le lire et le transmettre à la société Machin.)

Le RFC donne un exemple concret. La société Machin commande un nouveau routeur à un fabricant de routeur, mettons Truc, en indiquant l'adresse de livraison à Singapour. Truc doit indiquer à Machin le numéro de série de l'engin. Et il faut qu'une paire clé publique / clé privée spécifique à cet engin soit générée, la clé publique étant communiquée à l'acheteur (Machin). [Cette étape est l'une des deux seules que ne font pas les routeurs actuels.] Pendant que le routeur voyage vers Singapour, un employé

de Machin prépare la configuration du routeur, et la chiffre avec la clé publique de ce routeur particulier. Puis il la place sur le serveur de fichiers, indexée par le numéro de série. Les employés du centre de données mettent le routeur dans l'armoire et le câblent. Le routeur démarre et voit qu'il n'a pas encore été configuré. Il récupère une adresse IP puis contacte le serveur de fichiers. Il récupère la configuration, la déchiffre [deuxième étape qui n'est pas encore mise en œuvre dans un routeur typique d'aujourd'hui] et l'installe. Désormais, le routeur est autonome. Les équipes de l'opérateur Machin peuvent le contacter par les moyens habituels (SSH, par exemple). Un éventuel attaquant pourra regarder le trafic ou, plus simplement, demander le fichier de configuration au serveur de fichiers, mais il ne pourra pas le déchiffrer.

Cette solution ne nécessite que peu de changements aux mécanismes d'aujourd'hui. La section 3 du RFC décrit ce que les fabricants de routeurs devront faire, pour qu'elle soit disponible. Le routeur devra disposer d'une paire clé publique / clé privée, générée lors de sa fabrication. Et le fabricant doit pouvoir accéder à cette clé facilement (les clés cryptographiques ne peuvent typiquement pas être affichées sur un petit écran LCD...), afin de la communiquer à l'acheteur. (On peut aussi envisager le RFC 7030 ou bien le RFC 8894.) Le fabricant de routeurs doit avoir un mécanisme de communication de ces clés au client. (Comme il s'agit de clés publiques, il n'est pas indispensable que ce soit un mécanisme confidentiel, mais il faut évidemment qu'un tiers ne puisse pas y ajouter des clés de son choix.) Le RFC recommande X.509 (RFC 5280) comme format pour ces clés, même s'il n'y a pas forcément besoin des méta-données de X.509 comme la date d'expiration.

Et le client, l'acheteur du routeur ? La section 4 décrit ce qu'il doit faire. Il faut réceptionner et transmettre le numéro de série du routeur (typiquement, cela veut dire une communication entre le service Achats et les Opérations, ce qui n'est pas toujours facile), puis récupérer la clé publique. Et enfin, il faut chiffrer la configuration (le RFC suggère S/MIME - RFC 5751).

Quelques détails supplémentaires figurent en section 5. Par exemple, si le routeur le permet (il faut du haut de gamme...), la clé privée peut être stockée dans un TPM. (Le RFC suggère également de jeter un œil à la "*Secure Device Identity*" de la norme IEEE Std 802-1AR <https://standards.ieee.org/standard/802_1AR-2018.html>.)

Et la section 7 du RFC revient sur la sécurité de ce mécanisme. Il sera de toute façon toujours préférable à la solution actuelle, qui est de transmettre la configuration en clair. Mais cela ne veut pas dire que tout soit parfait. Ainsi, un attaquant qui a un accès physique à la machine pendant son trajet, ou bien une fois qu'elle est arrivée dans le centre de données, peut extraire la clé privée (sauf si elle est stockée dans un TPM) et déchiffrer à loisir.

On a dit au début que le but de ce RFC n'était pas de trouver un moyen d'authentifier la configuration. (Une signature ne serait pas réaliste, car il y a peu de chances que le fabricant mette les clés publiques de chaque opérateur dans le routeur.) Un attaquant qui contrôle, par exemple, le réseau sur lequel se trouve l'équipement réseau qui se configure, peut donc diriger les requêtes TFTP vers un serveur de son choix et envoyer une configuration « pirate » à cet équipement. (Il peut même la chiffrer puisque la clé publique de l'équipement sera sans doute facilement accessible publiquement, pour faciliter la tâche du client.)

Et, bien sûr, rien ne protège contre un vendeur malhonnête. Si le fabricant du routeur ou du commutateur est malveillant (ou piraté par un de ses employés, ou par un pirate extérieur), il peut tout faire. C'est le problème de la "*supply chain*" en sécurité, et il est très difficile à résoudre. (Cf. les craintes concernant les pratiques de Huawei mais il serait naïf de penser que seuls les fabricants chinois soient malhonnêtes. Voyez aussi mon article sur les portes dérobées dans les routeurs <https://www.bortzmeyer.org/porte-derobee-routeur.html>.)

Si vous aimez le concret, l'annexe A du RFC donne les commandes OpenSSL pour effectuer les différentes manipulations décrites par ce RFC. (J'ai modifié les commandes du RFC, qui ne marchaient pas pour moi.) Le fabricant du routeur va d'abord générer la paire de clés sur le routeur, la clé publique étant ensuite emballée dans un certificat (`SN19842256` étant le numéro de série du routeur), et auto-signer :

```
%  openssl req  -new -nodes -newkey rsa:2048 -keyout key.pem -out SN19842256.csr
...
Common Name (e.g. server FQDN or YOUR name) []:SN19842256
...

%  openssl req -x509 -days 36500 -key key.pem -in SN19842256.csr  -out SN19842256.crt
```

L'opérateur réseau va fabriquer un fichier de configuration (ici, trivial) :

```
% echo 'Test: true' > SN19842256.cfg
```

Ensuite, récupérer la clé (`SN19842256.crt`) et chiffrer la configuration :

```
%  openssl smime -encrypt -aes-256-cbc -in SN19842256.cfg  -out SN19842256.enc -outform PEM SN19842256.crt

% cat SN19842256.enc
-----BEGIN PKCS7-----
MIIB5gYJKoZIhvcNAQcDoIIB1zCCAdMCAQAxggGOMIIBigIBADByMFoxCzAJBgNV
BAYTAkFVMRMwEQYDVQQIDApTb21lLVN0YXRlMSEwHwYDVQQKDBhJbnRlcm5ldCBX
...
```

On copie ensuite la configuration chiffrée sur le serveur de fichiers.

Lorsque le routeur démarre, il charge ce fichier et exécute une commande de déchiffrement :

```
%  openssl smime -decrypt -in SN19842256.enc  -inform PEM -out config.cfg -inkey key.pem

% cat config.cfg
Test: true
```

Pour l'instant, aucun fabricant de routeur ne semble avoir annoncé une mise en œuvre de ce RFC. Comme le RFC décrit un mécanisme générique, et pas un protocole précis, les détails dépendront sans doute du vendeur.