Qualité des clés cryptographiques

Stéphane Bortzmeyer
<stephane+blog@bortzmeyer.org>

Première rédaction de cet article le 6 juin 2015

https://www.bortzmeyer.org/qualite-cles.html

A priori, une clé cryptographique en vaut une autre, non ? Ce sont toutes des chaînes de bits incompréhensibles ? Mais non : une clé peut, sous une apparence normale, avoir des caractéristiques qui la rendent plus ou moins vulnérable à certaines attaques cryptanalytiques. Deux analyses récentes ont testé des grands ensembles de clés et ont parfois trouvé des failles.

La première porte sur les clés PGP stockées dans les serveurs de clé. Ces serveurs sont utilisés pour distribuer les clés publiques. Étant librement accessibles, ils permettent d'analyser par la suite la qualité des clés. C'est ce qu'a fait l'étude « *"A look at the PGP ecosystem through the key server data"* <https://eprint.iacr.org/2015/262.pdf> ». L'auteur télécharge les clés depuis les serveurs qui fournissent un moyen d'accès en masse, puis les analyse. Il a développé pour cela un analyseur en Python. (J'ai déjà parlé <https://www.bortzmeyer.org/nouvelle-cle-pgp.html> du manque regrettable d'outil de test automatique de la qualité d'une clé PGP.)

Je vous préviens tout de suite, cette analyse n'a montré aucun grave problème (tant mieux ! À moins que les problèmes n'aient pas été détectés...) Par exemple, une faiblesse connue de DSA (et ECDSA) est la nécessité d'utiliser une valeur imprévisible (et pas forcément aléatoire, contrairement à ce qu'on lit souvent, cf. RFC 6979 [1]) à **chaque signature**. Le non-respect de cette règle a permis, par exemple, le libre accès à la PlayStation <http://www.exophase.com/20540/hackers-describe-ps3-security-as-epic-fail->. L'auteur a donc cherché des exemples de plusieurs signatures (les serveurs de clés stockent des clés signées par d'autres clés, donc on a aussi un lot de signatures) faites avec le même paramètre. Les deux seuls cas trouvés concernaient des signatures incorrectes (les serveurs de clés ne testent pas les données soumises et on peut donc trouver des données corrompues) ou bien les clés de PrimeFactors <http://www.primefactors.com>, une société qui commercialise des solutions cryptographiques (!) et qui a répondu à l'alerte de sécurité de l'auteur par une explication montrant qu'ils n'avaient pas compris le problème. Bon rappel que les seules solutions de sécurité sérieuses sont en logiciel libre.

1. Pour voir le RFC de numéro NNN, https://www.ietf.org/rfc/rfcNNN.txt, par exemple https://www.ietf.org/rfc/rfc6979.txt

D'autres tests faits sur les clés PGP ont également été des échecs (du point de vue du chercheur : pour la sécurité, c'est une bonne chose !) Par exemple, deux clés RSA partageaient un facteur premier mais l'adresse associée à la clé, `alice@example.com` donne à penser qu'il ne s'agissait pas d'une vraie clé, avec une vraie utilisatrice, mais plutôt d'une expérimentation avec des clés délibérement vulnérables.

Bref, chou blanc, pas de problème sensationnel noté pendant l'analyse des clés PGP.

Une autre étude a donné des résultats plus inquiétants. « *"Auditing GitHub users[Caractère Unicode non montré [2]] SSH key quality"* <https://blog.benjojo.co.uk/post/auditing-github-users-keys> » es le récit d'un audit des clés SSH utilisés sur GitHub (dont j'ai découvert à cette occasion qu'elles sont publiques, voir par exemple les miennes en). L'auteur a pu faire des statistiques, par exemple sur le type de clés (nette domination de RSA, DSS loin derrière et les courbes elliptiques encore plus).

Mais il a aussi trouvé des choses plus inquiétantes comme des clés RSA de seulement 256 bits (!) Une telle clé se factorise en 25 minutes sur un vieux processeur. Plus étonnant, GitHub stocke encore plusieurs clés qui ont été générées par la bogue Debian. Et ces clés « protègent » (cela a été corrigé depuis) des dépôts logiciels importants comme les bibliothèques cryptographiques de Python (!).

C'est donc l'occasion de rappeler que la cryptographie ne fait pas de miracles : il ne suffit pas d'utiliser PGP ou SSH pour être en sécurité, il faut aussi s'assurer de la solidité des clés. Il faudrait vraiment que des programmeurs courageux développent un site Web où on pourrait soumettre ses clés privées (non, je rigole, ses clés publiques) SSH ou PGP et avoir des diagnostics, du genre « clé privée prévisible suite à la bogue Debian » ou bien « clé trop courte » ou encore « facteur premier déjà vu ».

Autres articles sur ce sujet :

— « *"There[Caractère Unicode non montré]s no need to panic over factorable keys[Caractère Unicode non montré]just mind your Ps and Qs"* <https://freedom-to-tinker.com/blog/nadiah/new-research-theres-no-need-panic-over-factorable-keys-just-mind-your-ps-and-qs/> » (recherche publiée sur <https://factorable.net/>).
— « *"Batch-GCDing Github SSH Keys"* <http://cryptosense.com/batch-gcding-github-ssh-keys/>.

2. Car trop difficile à faire afficher par LaTeX