RFC 8624 : Algorithm Implementation Requirements and Usage Guidance for DNSSEC

Stéphane Bortzmeyer
<stephane+blog@bortzmeyer.org>

Première rédaction de cet article le 2 septembre 2019

Date de publication du RFC : Juin 2019

https://www.bortzmeyer.org/8624.html

Quel algorithme de cryptographie choisir pour mes signatures DNSSEC, se demande l'ingénieur système. Lesquels doivent être reconnus par le logiciel que j'écris, s'interroge la programmeuse. Il y a bien un registre IANA <https://www.iana.org/assignments/dns-sec-alg-numbers/dns-sec-alg-numbers.xml#dns-sec-alg-numbers-1> des algorithmes normalisés mais c'est juste une liste non qualifiée, qui mêle des algorithmes de caractéristiques très différentes. Ce nouveau RFC vise à répondre à cette question en disant quels sont les algorithmes recommandés. Il remplace l'ancien RFC 6944 [1], qui est modifié considérablement. Notamment, il marque l'avantage désormais donné aux courbes elliptiques par rapport à RSA.

La précédente liste d'algorithmes possibles datait donc du RFC 6944. D'autres algorithmes ont été ajoutés par la suite. Certains sont devenus populaires. Par exemple, ECDSA est maintenant suffisamment répandu pour qu'un résolveur validant ne puisse plus raisonnablement l'ignorer. D'autres algorithmes ont été peu à peu abandonnés, par exemple parce que les progrès de la cryptanalyse les menaçaient trop.

Aujourd'hui, le développeur qui écrit ou modifie un signeur (comme ldns <https://www.nlnetlabs.nl/projects/ldns/>, utilisé par OpenDNSSEC) ou un logiciel résolveur validant (comme Unbound ou Knot <https://www.knot-resolver.cz/>) doit donc se taper pas mal de RFC mais aussi pas mal de sagesse collective distillée dans plusieurs listes de diffusion pour se faire une bonne idée des algorithmes que son logiciel devrait gérer et de ceux qu'il peut laisser tomber sans trop gêner ses utilisateurs. Ce RFC vise à lui simplifier la tâche, en classant ces algorithmes selon plusieurs niveaux.

1. Pour voir le RFC de numéro NNN, https://www.ietf.org/rfc/rfcNNN.txt, par exemple https://www.ietf.org/rfc/rfc6944.txt

Notre RFC 8624 détermine pour chaque algorithme s'il est **indispensable** ("*MUST*", nécessaire pour assurer l'interopérabilité), **recommandé** ("*RECOMMENDED*", ce serait vraiment bien de l'avoir, sauf raison contraire impérieuse), **facultatif** ("*MAY*", si vous n'avez rien d'autre à faire de vos soirées que de programmer) ou tout simplement déconseillé ("*NOT RECOMMENDED*"), voire **à éviter** ("*MUST NOT*", pour le cas de faiblesses cryptographiques graves et avérées). Il y a deux catégorisations, une pour les signeurs (le cas de l'administratrice système cité au début), et une pour les résolveurs qui valideront. Par exemple, un signeur ne devrait plus utiliser RSA avec SHA-1, vu les faiblesses de SHA-1, mais un résolveur validant doit toujours le traiter, car des nombreux domaines sont ainsi signés. S'il ignorait cet algorithme, bien des zones seraient considérées comme non signées.

La liste qualifiée des algorithmes se trouve dans la section 3 : ECDSA avec la courbe P-256, et RSA avec SHA-256, sont les seuls indispensables pour les signeurs. ED25519 (RFC 8080) est recommandé (et sera probablement indispensable dans le prochain RFC). Plusieurs algorithmes sont à éviter, comme DSA, GOST R 34.10-2001 (RFC 5933) ou RSA avec MD5 (RFC 6151). Tous les autres sont facultatifs.

Pour les résolveurs validants, la liste des indispensables et des recommandés est un peu plus longue. Par exemple, ED448 (RFC 8080) est facultatif pour les signeurs mais recommandé pour les résolveurs.

La même section 3 justifie ces choix : RSA+SHA-1 est l'algorithme de référence, celui qui assure l'interopérabilité (tout logiciel compatible DNSSEC doit le mettre en œuvre) et c'est pour cela qu'il reste indispensable pour les résolveurs, malgré les faiblesses de SHA-1. RSA+SHA-256 est également indispensable car la racine <`http://www.root-dnssec.org/wp-content/uploads/2010/06/icann-dps-00.txt`> et la plupart des TLD l'utilisent aujourd'hui. Un résolveur qui ne comprendrait pas ces algorithmes ne servirait pas à grand'chose. RSA+SHA-512 ne pose pas de problème de sécurité, mais a été peu utilisé, d'où son statut « non recommandé » pour les signeurs.

D'autre part, le RFC insiste sur le fait qu'on ne peut pas changer le statut d'un algorithme trop vite : il faut laisser aux ingénieurs système le temps de changer leurs zones DNS. Et les résolveurs sont forcément en retard sur les signeurs : même si les signeurs n'utilisent plus un algorithme dans leurs nouvelles versions, les résolveurs devront continuer à l'utiliser pour valider les zones pas encore migrées.

Depuis le RFC 6944, ECDSA a vu son utilisation augmenter nettement. Les courbes elliptiques sont clairement l'avenir, d'où leur statut mieux placé. Ainsi, une zone DNS qui n'était pas signée et qui va désormais l'être devrait choisir un algorithme à courbes elliptiques, comme ECDSA ou EdDSA (RFC 8032 et RFC 8080). Avec ECDSA, il est recommandé d'utiliser l'algorithme déterministe du RFC 6979 pour générer les signatures. Les zones actuellement signées avec RSA devraient migrer vers les courbes elliptiques. Une chose est sûre, la cryptographie évolue et ce RFC ne sera donc pas éternel.

Le RFC note d'ailleurs (section 5) que le remplacement d'un algorithme cryptographique par un autre (pas juste le remplacement d'une clé) est une opération complexe, à faire avec prudence et après avoir lu les RFC 6781 et RFC 7583.

Ah, et parmi les algorithmes à courbes elliptiques, GOST (RFC 5933) régresse car l'ancien algorithme R 34.10-2001 a été remplacé par un nouveau qui n'est pas, lui, normalisé pour DNSSEC. L'algorithme venant du GOST avait été normalisé pour DNSSEC car les gérants du `.ru` disaient qu'ils ne pouvaient pas signer avec un algorithme étranger mais, finalement, ils ont utilisé RSA, ce qui diminue sérieusement l'intérêt des algorithmes GOST.

Outre les signeurs et les résolveurs, le RFC prévoit le cas des registres, qui délèguent des zones signées, en mettant un enregistrement DS dans leur zone. Ces enregistrements DS sont des condensats de la clé publique de la zone fille, et, ici, SHA-1 est à éviter et SHA-256 est indispensable.

Aujourd'hui, les mises en œuvre courantes de DNSSEC sont en général compatibles avec ce que demande le RFC. Elles sont parfois trop « généreuses » (RSA+MD5 encore présent chez certains), parfois un peu trop en retard (ED448 pas encore présent partout).