RFC 8080 : Edwards-Curve Digital Security Algorithm (EdDSA) for DNSSEC

Stéphane Bortzmeyer
<stephane+blog@bortzmeyer.org>

Première rédaction de cet article le 15 février 2017

Date de publication du RFC : Février 2017

https://www.bortzmeyer.org/8080.html

Ce RFC (premier RFC du groupe CURDLE <https://datatracker.ietf.org/wg/curdle/>) spécifie comment utiliser les algorithmes de cryptographie à courbe elliptique Ed25519 et Ed448 dans DNSSEC.

Contrairement à ce qu'on a pu parfois lire sous la plume de trolls ignorants, DNSSEC, mécanisme d'authentification des enregistrements DNS, n'est en rien lié à RSA. Au contraire, comme tous les protocoles cryptographiques de l'IETF, il dispose d'une propriété nommée **agilité cryptographique**. Ce nom désigne un système utilisant la cryptographie qui n'est pas lié à un algorithme cryptographique particulier. Il peut donc en changer, notamment pour suivre les progrès de la cryptanalyse, qui rend l'abandon de certains algorithmes nécessaire. Aujourd'hui, par exemple, RSA semble bien démodé, et les courbes elliptiques ont le vent en poupe. Aucun problème pour DNSSEC : aussi bien les clés que les signatures disposent d'un champ numérique qui indique l'algorithme cryptographique utilisé. Les valeurs possibles de ce champ figurent dans un registre IANA <https://www.iana.org/assignments/dns-sec-alg-numbers/dns-sec-alg-numbers.xml#dns-sec-alg-numbers-1>, registre auquel viennent d'être ajoutés (cf. sections 5 et 7) 15 pour Ed25519 et 16 pour Ed448.

Notez que ces algorithmes ne sont pas les premiers algorithmes à courbes elliptiques normalisés pour DNSSEC : le premier avait été GOST R 34.10-2001 (RFC 5933 [1]), il y a six ans, et le second ECDSA (RFC 6605).

Les algorithmes cryptographiques Ed25519 et Ed448, instances de EdDSA, sont spécifiés dans le RFC 8032. Ils peuvent d'ailleurs servir à d'autres systèmes que DNSSEC (par exemple SSH, cf. RFC 7479).

1. Pour voir le RFC de numéro NNN, https://www.ietf.org/rfc/rfcNNN.txt, par exemple https://www.ietf.org/rfc/rfc5933.txt

Les détails pratiques pour DNSSEC, maintenant (section 3 du RFC). Notre nouveau RFC est court car l'essentiel était déjà fait dans le RFC 8032, il ne restait plus qu'à décrire les spécificités DNSSEC. Une clé publique Ed25519 fait 32 octets (section 5.1.5 du RFC 8032) et est encodée sous forme d'une simple chaîne de bits. Une clé publique Ed448 fait, elle, 57 octets (section 5.2.5 du RFC 8032).

Les signatures (cf. section 4 de notre RFC) font 64 octets pour Ed25519 et 114 octets pour Ed448. La façon de les générer et de les vérifier est également dans le RFC 8032, section 5.

Voici un exemple de clé publique Ed25519, et des signatures faites avec cette clé, extrait de la section 6 du RFC (**attention**, il y a deux erreurs <https://www.rfc-editor.org/errata_search.php?eid=4935>, les RFC ne sont pas parfaits) :

```
example.com. 3600 IN DNSKEY 257 3 15 (
             l02Woi0iS8Aa25FQkUd9RMzZHJpBoRQwAQEX1SxZJA4= )

example.com. 3600 IN DS 3613 15 2 (
             3aa5ab37efce57f737fc1627013fee07bdf241bd10f3b1964ab55c78e79
             a304b )

example.com. 3600 IN MX 10 mail.example.com.

example.com. 3600 IN RRSIG MX 3 3600 (
             1440021600 1438207200 3613 example.com. (
             Edk+IB9KNNWg0HAjm7FazXyrd5m3Rk8zNZbvNpAcM+eysqcUOMIjWoevFkj
             H5GaMWeG96GUVZu6ECKOQmemHDg== )
```

Et, pour une vraie clé dans un vrai domaine, cette fois sans erreur :

```
% dig DNSKEY ed25519.monshouwer.eu

; <<>> DiG 9.9.5-9+deb8u9-Debian <<>> DNSKEY ed25519.monshouwer.eu
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 46166
;; flags: qr rd ra; QUERY: 1, ANSWER: 2, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags: do; udp: 4096
;; QUESTION SECTION:
;ed25519.monshouwer.eu. IN DNSKEY

;; ANSWER SECTION:
ed25519.monshouwer.eu. 3537 IN DNSKEY 257 3 15 (
    2wUHg68jW7/o4CkbYue3fYvGxdrd83Ikhaw38bI9dRI=
    ) ; KSK; alg = 15; key id = 42116
ed25519.monshouwer.eu. 3537 IN RRSIG DNSKEY 15 3 3600 (
    20170223000000 20170202000000 42116 ed25519.monshouwer.eu.
    Gq9WUlr01WvoXEihtwQ6r7t9AfkQfyETKTfm84WtcZkd
    M04KEe+4xu9jqhnG9THDAmV3FKASyWQ1LtCaOFr5Dw== )

;; Query time: 0 msec
;; SERVER: 127.0.0.1#53(127.0.0.1)
;; WHEN: Wed Feb 15 13:13:27 CET 2017
;; MSG SIZE  rcvd: 215
```

Quelques questions de sécurité pour conclure ce RFC (section 8). Les clés Ed25519 font l'équivalent de 128 bits de sécurité (et 224 pour Ed448). Du fait de l'existence d'algorithmes efficaces pour les casser sur un ordinateur quantique, cela ne sera pas suffisant le jour où on disposera d'un tel ordinateur. Ce jour est probablement très lointain (bien que la NSA, organisme de confiance, dise le contraire <https://www.iad.gov/iad/library/ia-guidance/ia-solutions-for-classified/algorithm-guidance/cnsa-suite-and-quantum-computing-faq.cfm>).

Enfin, la même section 8 rappelle l'existence des attaques trans-protocoles : il ne faut pas utiliser une clé dans DNSSEC et dans un autre protocole.

À noter que ce RFC est un pas de plus vers une cryptographie 100 % Bernstein, avec cette adaptation des algorithmes utilisant Curve25519 à DNSSEC. Bientôt, l'ancien monopole de RSA aura été remplacé par un monopole de Curve25519.

Apparemment, le RFC est un peu en avance sur le logiciel, les systèmes DNSSEC existants ne gèrent pas encore Ed25519 ou Ed448 (à part, semble-t-il, PowerDNS et, bientôt, DNS Go <https://github.com/miekg/dns/pull/458>. BIND a eu EdDSA à partir de la version 9.12.)