RFC 6975 : Signaling Cryptographic Algorithm Understanding in DNSSEC

Stéphane Bortzmeyer
<stephane+blog@bortzmeyer.org>

Première rédaction de cet article le 4 juillet 2013

Date de publication du RFC : Juillet 2013

https://www.bortzmeyer.org/6975.html

Le système d'authentification du DNS DNSSEC repose sur la signature cryptographique des enregistrements DNS. Cette signature peut se faire par des algorithmes différents. Comment savoir si tel algorithme, récemment spécifié, est désormais d'un usage fréquent ou pas chez les résolveurs validants ? Ce nouveau RFC propose que le résolveur indique, dans sa requête DNS, la liste des algorithmes qu'il comprend. Cela permettra, dans le futur, de mesurer objectivement l'état du déploiement de tel ou tel algorithme.

L'algorithme utilisé dans une signature DNSSEC apparait dans l'enregistrement `RRSIG` sous forme d'un octet dont la signification est stockée dans un registre IANA <https://www.iana.org/assignments/dns-sec-alg-numbers/dns-sec-alg-numbers.xml#dns-sec-alg-numbers-1>. Ainsi, dans cette signature de `.fr` :

```
% dig +dnssec SOA fr.
...
;; ANSWER SECTION:
fr. 172800 IN SOA nsmaster.nic.fr. hostmaster.nic.fr. 2222283656 3600 1800 3600000 5400
fr. 172800 IN RRSIG SOA 8 1 172800 20130814151003 20130615141003 62646 fr. FSSGO0iZ6OBoSUE12Q/NYOU2f3AMNbOf/b4Fo
...
```

La signature est faite avec l'algorithme 8, RSA + SHA-256. Dans le cas de `ecdsa.isc.org` :

```
% dig +dnssec SOA ecdsa.isc.org
...
;; ANSWER SECTION:
ecdsa.isc.org. 3600 IN SOA ns-int.isc.org. hostmaster.isc.org. 2013052903 7200 3600 604800 3600
ecdsa.isc.org. 3600 IN RRSIG SOA 14 3 3600 20130712033157 20130612023157 30631 ecdsa.isc.org. o+Q1WDDeiCM3z2b793
...
```

Elle est fait avec 14, ECDSA avec SHA-384. On trouve également des numéros identifiant un algorithme, celui de condensation cryptographique, dans les enregistrements DS et un autre dans NSEC3 (les valeurs possibles étant dans un autre registre IANA <https://www.iana.org/assignments/ds-rr-types/ds-rr-types.xml>).

Actuellement, les serveurs faisant autorité pour une zone ne savent pas quels algorithmes sont compris par les résolveurs qui les interrogent. Le serveur qui fait autorité envoie toutes les signatures qu'il connait et le résolveur se débrouille. Reprenons le cas de .fr, signé avec RSA. Pourrait-on passer à ECDSA, normalisé dans le RFC 6605 [1], et qui a l'avantage de fournir des clés et des signatures plus courtes ? Le problème est qu'ECDSA dans DNSSEC est très récent et qu'on pense que peu de résolveurs sont capables de valider avec ECDSA. Mais quel pourcentage exactement ? Et, avec le temps, ce pourcentage augmentera. Comment saura-t-on qu'on est arrivé au point où la grande majorité des résolveurs parlent ECDSA ? Si ce nouveau RFC 6975 est déployé, il fournira un moyen de répondre à ces questions.

Le principe est d'utiliser trois nouvelles options EDNS pour signaler les algorithmes connus. Le résolveur mettra ces options dans la requête et le serveur faisant autorité pourra les examiner pour connaître les capacités de ses clients (notez bien que c'est uniquement à des fins d'information : le serveur ne doit **pas** modifier sa réponse en fonction de ces options). EDNS est normalisé dans le RFC 6891 et l'enregistrement OPT qu'il ajoute contient zéro, une ou plusieurs options (la liste des options possibles est dans un registre IANA <https://www.iana.org/assignments/dns-parameters/dns-parameters.xml#dns-parameters-11>). Les trois options ajoutées sont (section 2 de notre RFC) :

— DAU (code 5) : "*DNSSEC Algorithm Understood*" indique quels algorithmes de signature sont acceptés,
— DHU (code 6) : "*DS Hash Understood*" indique quels algorithmes de condensation sont acceptés par le résolveur pour les enregistrements DS,
— N3U (code 7) : "*NSEC3 Hash Understood*" donne la même information pour la condensation dans les enregistrements NSEC3.

L'encodage d'une option EDNS se fait en trois champs, le code sur deux octets (5, 6 ou 7 ici), la longueur des données sur deux octets et les données. Ici, les données consistent en une liste d'algorithmes, un octet pour chacun, dans un ordre quelconque (l'ordre n'exprime pas une préférence). Par exemple, un résolveur validant qui accepte RSA-SHA1, RSA-SHA256 et ECDSA-SHA384 encodera un DAU avec les octets {5, 0, 3, 5, 8, 14}. En pratique, le RFC estime que les trois options toutes ensemble devraient prendre de 22 à 32 octets (en comptant 6 à 10 algorithmes de signature, plus quelques uns pour la condensation).

Bon, et une fois le format défini, on s'en sert comment ? Pour les clients (les résolveurs), c'est en sections 3 et 4 du RFC. Si le client valide, il ajoute une, deux ou trois des nouvelles options dans sa requête. Sinon, il ne doit pas les utiliser.

Si le résolveur valide et qu'il reçoit une requête ayant déjà une de ces options, il doit ajouter ses propres algorithmes. La liste finale sera donc l'union des deux. Un simple relais ("*forwarder*") qui ne valide pas mais qui reçoit une requête ayant une de ces options, doit passer l'option telle quelle car ce qui compte, ce sont les capacités du résolveur qui fera la validation DNSSEC.

Et le serveur faisant autorité ? C'est en section 5. Le point important est qu'il ne **doit rien** faire. Il suit le même algorithme que d'habitude et, par exemple, il envoie les mêmes signatures quels que soient les algorithmes compris par l'émetteur. Les trois nouvelles options sont là pour information uniquement.

1. Pour voir le RFC de numéro NNN, https://www.ietf.org/rfc/rfcNNN.txt, par exemple https://www.ietf.org/rfc/rfc6605.txt

D'autres part, les trois options DAU, DHU et N3U sont uniquement dans les requêtes. Le serveur faisant autorité n'en met pas dans les réponses.

Revenons aux administrateurs d'une zone qui voudraient mesurer le déploiement d'un nouvel algorithme cryptographique. Que faire si les clients n'envoient pas cette option ? La section 6 conseille de les considérer comme du vieux logiciel, n'ayant pas les nouvelles options, ni les nouveaux algorithmes. C'est un peu court, à mon avis, car on peut aussi imaginer qu'il y aura des résolveurs récents qui n'enverront pas cette option, peut-être par souci de discrétion (ce point est également couvert dans la section 7, consacrée aux risques de sécurité).

Il ne semble pas exister beaucoup de mises en œuvre de ce RFC. Par exemple, Wireshark, au moment de la publication du RFC, ne savait pas décoder ces options et affichait juste :

```
Additional records
    <Root>: type OPT
        Name: <Root>
        Type: OPT (EDNS0 option)
        UDP payload size: 1200
        Higher bits in extended RCODE: 0x0
        EDNS0 version: 0
        Z: 0x0
        Data length: 8
        Data
```

Depuis r50840 <http://anonsvn.wireshark.org/viewvc?revision=50840&view=revision>, Wireshark a le code nécessaire et on n'a plus qu'à patienter que cela arrive dans une version officielle.

L'examen des requêtes envoyées à un gros serveur de noms ne montre actuellement pratiquement pas d'utilisation de cette option.

Trois ans après la publication du RFC, une discussion lors d'une réunion OARC <https://dns-oarc.net/> semblait indiquer qu'il n'avait connu aucun déploiement et devait être considéré comme un échec.