RFC 9718 : DNSSEC Trust Anchor Publication for the Root Zone

Stéphane Bortzmeyer
<stephane+blog@bortzmeyer.org>

Première rédaction de cet article le 25 janvier 2025

Date de publication du RFC : Janvier 2025

https://www.bortzmeyer.org/9718.html

Le mécanisme d'authentification des informations DNS nommé DNSSEC repose sur la même structure arborescente que le DNS : une zone publie un lien sécurisé vers les clés de ses sous-zones. Un résolveur DNS validant n'a donc besoin, dans la plupart des cas, que d'une seule clé publique, celle de la racine. Elle lui servira à vérifier les clés des TLD, qui serviront à valider les clés des domaines de deuxième niveau et ainsi de suite. Reste donc à configurer la clé de la racine dans le résolveur : c'est évidemment crucial, puisque toute la sécurité du système en dépend. Si un résolveur est configuré avec une clé fausse pour la racine, toute la validation DNSSEC est menacée. Comment est-ce que l'ICANN, qui gère la clé principale de la racine, publie cette clé cruciale ? C'est documenté dans ce RFC qui remplace le RFC 7958 [1], avec plusieurs changements, comme l'abandon des signatures PGP.

Notez que ce nouveau RFC documente l'existant, déjà mis en œuvre, et ne prétend pas décrire la meilleure méthode. Notez aussi que ce format et cette méthode de distribution pourraient encore changer à l'avenir, comme ils l'ont fait depuis la sortie du RFC 7958.

Si vous voulez réviser DNSSEC d'abord, outre les RFC de base sur ce système (RFC 4033, RFC 4034, RFC 4035...), notez surtout le RFC 6781, qui décrit les questions opérationnelles liées au bon fonctionnement de DNSSEC.

Les clés publiques configurées dans les résolveurs qui valident avec DNSSEC, sont appelées « points de départ de la confiance » ("*trust anchors*"). Un point de départ de la confiance est une clé dont l'authenticité est admise, et non pas dérivée d'une autre clé, via une chaîne de signatures. Il en faut au moins un, celui de la racine, bien que certains résolveurs en ajoutent parfois deux ou trois pour des zones qu'ils veulent vérifier indépendamment. Lorsque le résolveur recevra une réponse de la racine,

1. Pour voir le RFC de numéro NNN, https://www.ietf.org/rfc/rfcNNN.txt, par exemple https://www.ietf.org/rfc/rfc7958.txt

signée, il l'authentifiera avec la clé publique de la racine (le point de départ de la confiance). S'il veut vérifier une réponse d'un TLD, il l'authentifiera avec la clé publique du TLD, elle-même signée (et donc authentifiée) par la clé de la racine. Et ainsi de suite même pour les zones les plus profondes.

(Notez qu'il existe deux clés pour la plupart des zones, la KSK - "*Key Signing Key*", et la ZSK - "*Zone Signing Key*", mais on ne s'intéresse ici qu'aux KSK, c'est elles qui sont signées par la zone parente, et configurées comme points de départ de la confiance.)

La gestion de la clé de la racine par l'ICANN (via sa filiale PTI <https://pti.icann.org/>) est décrite dans leur DPS (DNSSEC Practice Statement) <https://www.iana.org/dnssec/procedures>.

Le RFC rappelle aussi qu'il y a d'autres possibilités d'installation d'un point de départ de la confiance. Par exemple, si un tel point a été configuré une fois, ses remplacements éventuels peuvent être faits via le RFC 5011.

La section 2 du RFC décrit le format des clés publiées par l'IANA. Il s'agit d'un fichier XML contenant les condensats des clés, utilisant le format de présentation du RFC 4034. Leur syntaxe formelle est exprimée en Relax NG, le schéma est en section 2.1 du RFC (copie locale (en ligne sur https://www.bortzmeyer.org/files/dnssec-root-trust-anchors.rnc)). Une vérification du fichier avec rnv <http://www.davidashen.net/rnv.html> fonctionne :

```
% rnv  dnssec-root-trust-anchors.rnc root-anchors.xml
root-anchors.xml
```

Voici un exemple du fichier XML (à ne pas prendre comme s'il faisait autorité, évidemment) :

```
<TrustAnchor id="0C05FDD6-422C-4910-8ED6-430ED15E11C2"
             source="http://data.iana.org/root-anchors/root-anchors.xml">
    <Zone>.</Zone>
    <KeyDigest id="Klajeyz" validFrom="2017-02-02T00:00:00+00:00">
        <KeyTag>20326</KeyTag>
        <Algorithm>8</Algorithm>
        <DigestType>2</DigestType>
        <Digest>E06D44B80B8F1D39A95C0B0D7C65D08458E880409BBC683457104237C7F8EC8D</Digest>
        <PublicKey>AwEAAaz/...</PublicKey>
        <Flags>257</Flags>
    </KeyDigest>
</TrustAnchor>
```

L'élément <KeyTag> indique l'identifiant de la clé, actuellement 20326, comme on peut le voir avec dig :

```
% dig +multi +nodnssec . DNSKEY
...
;; flags: qr rd ra ad; QUERY: 1, ANSWER: 2, AUTHORITY: 0, ADDITIONAL: 1
...
;; ANSWER SECTION:
. 86254 IN DNSKEY 257 3 8 (
AwEAAaz/tAm8yTn4Mfeh5eyI96WSVexTBAvkMgJzkKTO
                                        ...
) ; KSK; alg = RSASHA256 ; key id = 20326
...
```

L'attribut `id` de l'élément `<TrustAnchor>` est un UUID. L'attribut `id` de l'élément `<KeyDigest>` sert à identifier un condensat particulier. Le fichier peut également comporter des commentaires XML (mais ce n'est pas le cas actuellement). Le fichier indique aujourd'hui le condensat de la clé et la clé mais cette dernière est optionnelle (on ne s'en sert pas par la suite).

Pour produire un enregistrement DS à partir de ce fichier XML, il suffit de mettre `<KeyTag>`, `<Algorithm>`, `<DigestType>` et `<Digest>` bout à bout. Par exemple, avec le fichier XML ci-dessus, cela donnerait :

```
. IN DS 20326 8 2 E06D44B80B8F1D39A95C0B0D7C65D08458E880409BBC683457104237C7F8EC8D
```

(Des résolveurs comme Unbound acceptent ce format, pour le point de confiance de départ.)

Ici, je n'ai montré qu'une clé mais il peut y en avoir plusieurs, donc plusieurs `<KeyDigest>`. En novembre 2024, le fichier comportait trois clés, l'ancienne 19036, abandonnée il y a des années, l'actuelle 20326 et la future 38696 (qui n'était pas encore publiée dans le DNS mais qui l'a été en janvier 2025).

Comment récupérer le fichier XML de manière à être sûr de son authenticité ? C'est ce que spécifie la section 3 du RFC : on utilise HTTPS (ou HTTP tout court mais ce n'est pas forcément une bonne idée et, de toute façon, aujourd'hui, une politique HSTS le rend difficile). L'URL est `https://data.iana.org/root-anchors/`. Si on fait confiance à l'AC utilisée par l'ICANN, on peut être sûr de l'authenticité du fichier. (Mais notez, comme le fait remarquer gregR `<https://mamot.fr/@gregr>`, que `data.iana.org` et `iana.org` n'ont pas d'enregistrement CAA, RFC 8659.)

Une autre solution est de le récupérer en HTTP et de le vérifier avec la signature fournie, en CMS (RFC 5652) - son URL est `https://data.iana.org/root-anchors/root-anchors.p7s`. (Non, je ne sais pas comment la vérifier.)

Bien sûr, un résolveur validant n'est pas obligé d'utiliser ces points de départ de la confiance. Cela reste une décision locale. La très grande majorité des résolveurs utiliseront sans doute plutôt un paquetage fourni par leur système d'exploitation, comme `dns-root-data` sur Debian.

Pour les amateurs d'histoire, l'annexe B rappelle que la clé 19036 (également appelée KSK-2010) avait été générée au cours d'une cérémonie à Culpeper, le 16 juin 2010. Elle avait été publiée dans le DNS pour la première fois le 15 juillet 2010. La clé actuelle, la 20326 (également nommée KSK-2017), a été également générée à Culpeper le 27 octobre 2016 et publiée le 11 novembre 2018. La liste des cérémonies de création de clés est en ligne `<https://www.iana.org/dnssec/ceremonies>`.

L'annexe A de notre RFC contient les changements depuis le RFC 7958. Notamment :
— Correction d'une sérieuse erreur technique `<https://www.rfc-editor.org/errata/eid5932>`,
— Suppression des certificats qui étaient distribués avec les versions précédentes `<https://data.iana.org/root-anchors/old/2017-02-03/>`,
— Suppression de la signature OpenPGP,
— Et divers points de détail.