RFC 9803 : Extensible Provisioning Protocol (EPP) mapping for DNS Time-To-Live (TTL) values

Stéphane Bortzmeyer
<stephane+blog@bortzmeyer.org>

Première rédaction de cet article le 14 juin 2025

Date de publication du RFC : Juin 2025

<https://www.bortzmeyer.org/9803.html>

Traditionnellement, les registres de noms de domaine publiaient toutes les informations des sous-domaines avec la même durée de vie maximale (TTL : "*Time To Live*"). Certain-es client-es demandent à pouvoir choisir un TTL plus court (ou, parfois, plus long) et ce RFC décrit comment faire cela avec le protocole EPP.

Cet EPP, normalisé dans les RFC 5730¹ et suivants, est aujourd'hui le plus utilisé par les gros registres pour recevoir les commandes de leurs BE. (Cf. cet article de l'Afnic <<https://www.afnic.fr/observatoire-ressources/papier-expert/que-se-passe-t-il-quand-on-enregistre-un-nom-de-domaine>>.) Tel que normalisé dans le RFC 5731, EPP ne permet pas, lors de la création d'un nom de domaine, de spécifier le TTL des enregistrements NS ("*NameServer*") qui seront publiés. Prenons l'exemple de l'enregistrement d'un `.fr`. Le ou la titulaire va sur l'interface Web de son BE, réserve le nom `cyberstructure.fr`, le BE transmet la demande de création en utilisant EPP et le registre publie ensuite dans le DNS les informations de délégation, ici récupérées avec `dig` :

```
% dig @d.nic.fr cyberstructure.fr NS
...
;; flags: qr rd; QUERY: 1, ANSWER: 0, AUTHORITY: 9, ADDITIONAL: 1
...
;; AUTHORITY SECTION:
cyberstructure.fr. 3600 IN NS puck.nether.net.
cyberstructure.fr. 3600 IN NS fns2.42.pl.
cyberstructure.fr. 3600 IN NS ns2.bortzmeyer.org.
cyberstructure.fr. 3600 IN NS ns4.bortzmeyer.org.
cyberstructure.fr. 3600 IN NS ns2.afraid.org.
cyberstructure.fr. 3600 IN NS fns1.42.pl.
cyberstructure.fr. 3600 IN NS nsl.bortzmeyer.org.
...
```

1. Pour voir le RFC de numéro NNN, <https://www.ietf.org/rfc/rfcNNN.txt>, par exemple <https://www.ietf.org/rfc/rfc5730.txt>

À aucun moment, la-e titulaire ou le BE n'a spécifié de TTL (les 3 600 secondes), il a été choisi par le registre seul. Certain-es titulaires peuvent pourtant avoir des préférences différentes. (Je rappelle que la notion de TTL est définie rigoureusement dans le RFC de terminologie DNS, le RFC 9499, voir sa section 5.)

Si vous voulez creuser ce concept de délégation, l'article de l'Afnic sur DELEG <<https://www.afnic.fr/observatoire-ressources/papier-expert/la-delegation-dns-et-sa-possible-evolution>> l'explique dans sa première partie.

L'exemple ci-dessus montrait les enregistrements NS ("*NameServer*") mais ce problème concerne aussi les DS (RFC 9364), la colle et les éventuels DNAME (par exemple pour mettre en œuvre le RFC 6927).

Ce RFC normalise une extension à EPP pour permettre au client (le BE) d'indiquer le TTL souhaité pour la délégation. (Ce qui ne signifie pas que le registre satisfera ce souhait, lisez plus loin.) Plus précisément, il étend les classes ("*mappings*") EPP pour les noms de domaine (RFC 5731 et pour les serveurs de noms (RFC 5732), les seules classes concernées par le DNS (rappelez-vous qu'EPP n'est pas limité aux noms de domaine, c'est un protocole d'avitaillement générique).

La section 1.2 du RFC contient les informations essentielles. D'abord, l'espace de noms `urn:ietf:params:xml:ns:epp:ttl` (enregistré à l'IANA <<https://www.iana.org/assignments/xml-registry/xml-registry.xml#ns>>), qui identifie cette extension (le RFC utilise l'alias `ttl` mais rappelez-vous qu'en XML, c'est l'espace de noms qui compte, l'alias est choisi par l'auteur du document XML). Ensuite, l'élément `<ttl:ttl>` (donc, en complet, `{urn:ietf:params:xml:ns:epp:ttl-1.0}ttl`), qui peut être ajouté aux demandes EPP ou aux réponses. Parmi ses attributs :

- `for` (le seul qui soit obligatoire) indique le type de données DNS auxquelles l'élément s'applique (uniquement des types concernés par la délégation, NS, DS, DNAME, A et AAAA, ainsi que la valeur spéciale `custom`, qu'il faut compléter un attribut `custom` qui suit l'expression rationnelle de la section 3.1 du RFC 6895, et évidemment utilise un type enregistré <<https://www.iana.org/assignments/dns-parameters/dns-parameters.xml#dns-parameters-4>>),
 - `min` (uniquement dans les réponses) qui indique la valeur minimale que le serveur (donc le registre) accepte (le registre est toujours libre de sa politique et peut refuser une valeur trop basse ou trop haute, répondant avec le code d'erreur EPP 2004, "*Parameter value range error*"),
 - `max`, la contrepartie de `min`, avec les mêmes propriétés,
 - `default`, qu'on ne trouve également que dans les réponses, et qui permet de connaître la valeur qu'utiliserait le serveur si le client ne demande rien de spécifique.
- L'élément `<ttl:ttl>` a pour contenu une valeur en secondes. Si cette valeur est vide, le serveur prendra la valeur par défaut (mais autant ne pas mettre d'élément `<ttl:ttl>` dans ce cas).

L'extension de ce RFC décrit des possibilités mais le registre, qui gère le serveur EPP, reste libre de ne pas tout accepter. Par exemple, il peut refuser l'utilisation de l'attribut `custom`, ou bien la restreindre à certains types (et, si le client demande quand même, répondre avec le code d'erreur EPP 2306 "*Parameter value policy error*"). Les types qui ont une syntaxe privilégiée (pas besoin de l'attribut XML `custom`) sont ceux qu'on peut actuellement trouver au dessus de la coupure de zone (section 7 du RFC 9499).

Vous avez noté que, dans les types qui ont une syntaxe privilégiée dans ce RFC, il y a les types pour les adresses IP. Ils sont utilisés pour la colle DNS (section 7 du RFC 9499), si le serveur EPP met en œuvre la classe EPP pour les serveurs de noms (RFC 5732).

Un deuxième élément XML est spécifié par ce RFC, `<ttl:info>`, qui sert à demander ou à indiquer des informations sur la politique du serveur (voir des exemples plus loin).

La section 2 décrit dans quelles commandes EPP on peut ajouter les nouveaux éléments. Je ne vais pas toutes les détailler, juste montrer quelques exemples du XML envoyé et des réponses. Ainsi, ici, le client va utiliser la commande `<epp:info>` pour se renseigner sur la politique du serveur :

```

...
C: <info>
C:   <domain:info
C:     xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
C:       <domain:name>example.com</domain:name>
C:     </domain:info>
C:   </info>
C: <extension>
C:   <ttl:info
C:     xmlns:ttl="urn:ietf:params:xml:ns:epp:t1-1.0"
C:     policy="false"/>
C: </extension>
...

```

Et la réponse du serveur :

```

...
S: <response>
S:   <result code="1000">
S:     <msg>Command completed successfully</msg>
S:   </result>
S:   <resData>
...
S:     <domain:name>example.com</domain:name>
S:     <domain:status s="ok"/>
...
S:   </resData>
S:   <extension>
S:     <t1:infData
S:       xmlns:t1="urn:ietf:params:xml:ns:epp:t1-1.0">
S:         <t1:t1 for="NS">172800</t1:t1>
S:         <t1:t1 for="DS">300</t1:t1>
S:       </t1:infData>
...
S:   </response>
...

```

Le serveur indique que le TTL des enregistrements NS sera de deux jours. Sinon, vous avez repéré le `policy="false"` ? S'il avait été à `true`, le serveur aurait renvoyé l'information sur sa politique pour tous les types d'enregistrements DNS (section 2.1.1.2).

Créons maintenant un nom de domaine en spécifiant un TTL :

```

C: <command>
C:   <create>
C:     <domain:create
C:       xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
C:         <domain:name>example.com</domain:name>
C:         <domain:ns>
C:           <domain:hostObj>ns1.example.com</domain:hostObj>
C:           <domain:hostObj>ns1.example.net</domain:hostObj>
C:         </domain:ns>
C:       </domain:create>

```

```
C: </create>
C: <extension>
C: <ttml:create
C:   xmlns:ttml="urn:ietf:params:xml:ns:epp:ttml-1.0">
C:   <ttml:ttml for="NS">172800</ttml:ttml>
C:   <ttml:ttml for="DS">300</ttml:ttml>
C: </ttml:create>
C: </extension>
C: </command>
C: </epp>
```

Cela marcherait aussi dans un `<domain:update>`, pour changer les TTL.

Je l'ai déjà dit plusieurs fois, la demande du client ne va pas forcément être honorée par le serveur. La section 3 de notre RFC détaille l'interaction entre cette demande et la politique du serveur (donc du registre). Ainsi, le serveur peut limiter les types d'enregistrement DNS à qui ces commandes s'appliquent, par exemple en n'acceptant de changer le TTL que pour les DS (d'où l'intérêt des demandes avec `policy=...`, pour savoir à l'avance).

D'autre part, le TTL des enregistrements publiés peut changer en dehors d'EPP, par une action du registre. Auquel cas, le registre (section 4) peut gentiment prévenir ses clients par le système de messagerie du RFC 8590.

La possibilité pour le client d'indiquer un TTL souhaité a aussi des conséquences opérationnelle (section 5). Ainsi, des TTL courts vont accroître la charge sur les serveurs faisant autorité <https://www.bortzmeyer.org/serveur-dns-faisant-autorite.html> puisque les résolveurs <https://www.bortzmeyer.org/resolveur-dns.html> auront besoin de demander plus souvent, les expirations se rapprochant. Souvent, les utilisateurs veulent des TTL courts, pour la réactivité, et n'ont pas conscience des conséquences, d'autant plus que l'utilisation des serveurs DNS faisant autorité est gratuite pour eux. C'est pour cela que le registre peut fixer des valeurs minimales (et maximales, pour traiter d'autres problèmes) aux TTL souhaités par le client.

Une erreur parfois commise est de penser qu'un changement du TTL (via une commande EPP `<epp:update>`) va se voir instantanément. Mais, en fait, les informations, avec l'ancien TTL, sont encore dans la mémoire de plusieurs résolveurs. Il est donc recommandé de planifier <https://www.bortzmeyer.org/changement-adresse-et-dns.html> les changements à l'avance, en tenant compte de cette mémorisation (section 5.2 du RFC). D'ailleurs, la politique du registre aussi peut changer, et les TTL par défaut, ainsi que les valeurs maximales et minimales, sont susceptibles d'être modifiées (un registre sérieux va informer ses utilisateurices de ces changements mais parfois, c'est oublié).

Un gros morceau de notre RFC est consacré à la sécurité (section 6) car les TTL ont une influence sur ce sujet. Premièrement, le "*fast flux*" (changement rapide des données de délégation DNS, cf. RFC 9499, section 7). Des malveillants utilisent cette technique pour rendre l'investigation numérique plus difficile et pour échapper à certaines règles de filtrage (voir le rapport SAC-025 du SAAC <https://www.icann.org/en/system/files/files/sac-025-en.pdf>). Un TTL court peut faciliter ce changement rapide. Le rapport du SAAC suggère un TTL minimum de 30 minutes et/ou de limiter le nombre de changements qu'on peut faire par jour.

Si un malveillant réussit à obtenir les secrets qui lui permettent de soumettre des demandes de modification (au passage, pensez à activer l'authentification à plusieurs facteurs), il mettra sans doute des TTL

très élevés pour que son détournement <<https://www.bortzmeyer.org/attaques-noms-domaine-explication.html>> dure plus longtemps. D'où l'intérêt d'une valeur maximale pour les TTL.

Notre extension à EPP pour les TTL figure désormais dans le registre IANA des extensions <<https://www.iana.org/assignments/epp-extensions/epp-extensions.xml#epp-extensions-1>> et sa syntaxe complète, si vous êtes curieux-se, figure dans la section 8 (au format des schémas XML). Question mises en œuvre, vous en trouverez au moins deux, dans le SDK de Verisign <https://www.verisign.com/en_US/channel-resources/domain-registry-products/epp-sdks/index.xhtml> et dans le logiciel Pepper <<https://github.com/gbxyz/pepper/>> (plus exactement dans la bibliothèque qu'il utilise, Net : :EPP <<https://metacpan.org/pod/Net%3A%3AEPP>>).