RFC 7816 : DNS query name minimisation to improve privacy

Stéphane Bortzmeyer
<stephane+blog@bortzmeyer.org>

Première rédaction de cet article le 23 mars 2016

Date de publication du RFC : Mars 2016

https://www.bortzmeyer.org/7816.html

La meilleure façon de protéger les données contre la surveillance, c'est de ne pas avoir de données. Ce principe de base de la protection de la vie privée est souvent oublié. C'est pourtant un des deux piliers de la confidentialité, avec la protection technique des données. Le DNS a pendant longtemps transmis trop de données, et ce RFC décrit une technique qui va limiter les fuites, la "*QNAME minimisation*", ou « réduction de la question posée ». (Il a depuis été remplacé par le RFC 9156 [1].) Demandez à votre FAI ou à votre service informatique de l'activer !

Si vous regardez les vidéos sur le fonctionnement du DNS (comme celle-ci <https://www.youtube.com/watch?v=42pxnLZJXzo>) ou lisez les textes expliquant « le DNS pour les nuls », vous y trouverez souvent une fausse explication de la résolution DNS, où les serveurs faisant autorité <https://www.bortzmeyer.org/serveur-dns-faisant-autorite.html> ne reçoivent que la question strictement nécessaire. Ainsi, dans la vidéo citée ci-dessus, le serveur de la racine reçoit une question où ne figure que le TLD. Mais la réalité du DNS, aujourd'hui, est tout autre : les serveurs faisant autorité reçoivent la totalité de question originale. Ainsi, si vous visitez `www.alcooliques-anonymes.fr`, la racine voit que vous vous intéressez à l'alcoolisme, alors que ce n'était nullement nécessaire pour son travail (puisqu'elle ne connait que les TLD). Si votre logiciel BitTorrent demande `_bittorrent-tracker._tcp.domain.example`, les serveurs faisant autorité pour `.example` sauront que vous faites du BitTorrent, alors qu'ils ne connaissaient que les domaines situés immédiatement sous `.example`. Le RFC 7626 décrit plus en détail les problèmes de vie privée liés au DNS.

Dans le dernier cas, pour que la résolution se passe bien, il aurait suffit de demander à la racine « quels sont les serveurs de noms de `.example` » et à ces serveurs « quels sont les serveurs de noms de `domain.example` ». C'est le principe de base de la "*QNAME minimisation*".

1. Pour voir le RFC de numéro NNN, `https://www.ietf.org/rfc/rfcNNN.txt`, par exemple `https://www.ietf.org/rfc/rfc9156.txt`

Bien sûr, on pourrait chiffrer le trafic DNS (et le groupe de travail DPRIVE <https://tools.ietf.org/wg/dprive> de l'IETF travaille précisément sur ce sujet). Mais cela ne protège que contre un tiers écoutant le réseau : les serveurs faisant autorité <https://www.bortzmeyer.org/serveur-dns-faisant-autorite.html> verraient quand même les données et pourraient en faire ce qu'ils veulent. C'est pour cela qu'un principe de base en matière de protection de la vie privée est de marcher sur deux jambes (RFC 6973) : minimiser les données envoyées **et** les protéger. Lorsqu'on parle de vie privée, pas mal d'informaticiens réagissent en criant « cryptographie ! » alors que celle-ci ne protège pas contre tout et notamment pas contre le serveur à qui on parle.

Et pourquoi est-ce que les résolveurs DNS <https://www.bortzmeyer.org/resolveur-dns.html> envoyaient la question complète ("*full QNAME*" où "*QNAME*" veut dire "*Query NAME*") ? Uniquement parce que la protection de la vie privée n'était pas tellement prise en compte à l'époque ? Pas uniquement : il y avait quelques raisons techniques spécifiques à l'époque (le RFC les détaille mais, surtout, il ne faut pas oublier que toutes les décisions concernant le DNS n'ont pas forcément été mûrement réfléchies).

La section 2 décrit la "*QNAME minimisation*". Elle est mise en œuvre dans le **résolveur <https://www.bortzmeyer.org/resolveur-dns.html>** DNS (aucun changement n'est fait dans le serveur faisant autorité, la "*QNAME minimisation*" ne change pas le protocole DNS). Avant, lorsqu'il recevait une requête demandant l'adresse IPv6 pour `_foobar._tcp.sub.internautique.fr` et qu'il connaissait les serveurs faisant autorité pour `.fr`, mais pas ceux faisant autorité pour `internautique.fr`, le résolveur envoyait à l'AFNIC une requête avec comme QNAME ("*Query NAME*") le nom complet `_foobar._tcp.sub.internautique.fr` et comme QTYPE ("*Query TYPE*") `AAAA` (indiquant une demande d'adresse IPv6). Désormais, le résolveur moderne qui met en œuvre la "*QNAME minimisation*" enverra une requête avec le QNAME `internautique.fr` et le QTYPE `NS` (demande des serveurs de noms). Plus rigoureusement, la requête est faite avec un QNAME qui est l'original, où on a retiré les premiers composants, jusqu'à un seul composant de plus que celui pour lequel on connait les serveurs faisant autorité.

Les experts en DNS ont noté un problème : il n'y a pas forcément un jeu de serveurs faisant autorité pour chaque composant. Si je prends `www.st-cyr.terre.defense.gouv.fr`, il n'y a par exemple aujourd'hui pas de serveurs de noms pour `gouv.fr`, ce sont ceux de `.fr`. En termes techniques, il n'y a pas de limite de zone ("*zone cut*", cf. RFC 2181, section 6) à chaque composant. Dans le cas de ce dernier nom, il y a une limite de zone entre la racine et `.fr`, une autre entre `fr` et `defense.gouv.fr` mais pas entre `.fr` et `gouv.fr`. Un résolveur qui veut faire de la "*QNAME minimisation*" doit donc tenir compte des limites de zone. S'il valide avec DNSSEC, pas de problème, il les connait déjà, car leur connaissance est nécessaire au bon fonctionnement de DNSSEC. Sinon, il doit les trouver tout seul, par exemple avec l'algorithme de l'annexe A.

Est-ce un changement « légal » du fonctionnement du résolveur DNS ? La section 4 discute ce problème et conclut que oui. La "*QNAME minimisation*" est permise par les RFC existants (RFC 1034, section 5.3.3 et RFC 1035, section 7.2). C'est un changement unilatéral de la part du résolveur, ne nécessitant pas de changement dans les serveurs faisant autorité. Comme c'est un changement unilatéral, différents résolveurs pourront choisir de la mettre en œuvre de façon légèrement différente. L'annexe B décrit certaines de ces alternatives, comme d'utiliser des requêtes « traditionnelles » avec le nom de domaine complet, au démarrage du résolveur, attendant que le cache soit peuplé pour passer à la "*QNAME minimisation*", qui préserve la vie privée mais peut nécessiter davantage de paquets.

La "*QNAME minimisation*" ne change pas le protocole DNS. Elle ne pose donc pas de problème avec les vieux serveurs. En théorie car, en pratique, il existe pas mal de serveurs incorrects qui ne suivent pas les règles et poseront quelques problèmes (section 3 du RFC, voir aussi un intéressant exposé <https://indico.dns-oarc.net/event/21/contribution/9> et un « storify » d'une

discussion <https://storify.com/shuque/qname-minimization-dns-oarc>.) Le problème n'est en général pas dû aux serveurs en logiciel libre sérieux qui forment l'essentiel de l'infrastructure du DNS (BIND, NSD, Knot...) mais plutôt aux "*appliances*" boguées que certains s'obstinent à placer devant des serveurs qui marcheraient parfaitement autrement.

C'est par exemple le cas de certains répartiteurs de charge qui répondent aux requêtes pour certains QTYPE mais qui échouent lorsque le QTYPE vaut `NS` (répondant, par exemple `REFUSED`). Pire, certains ne répondent pas du tout, comme ceux de `www.ratp.fr`. Il s'agit bien d'une bogue, et qui cause plein de problèmes, pas seulement à la "*QNAME minimisation*".

Un autre problème est celui des serveurs bogués (comme djbns) qui ne réagissent pas correctement aux ENT. Qu'est-ce qu'un ENT ? Un "*Empty Non-Terminal*" (terme décrit dans le RFC 8499, section 7) est un domaine qui n'a pas d'enregistrements DNS mais qui a des sous-domaines qui en ont. `gouv.fr`, cité plus haut, est un ENT mais ceux-ci sont particulièrement fréquents sous `ip6.arpa`. Normalement, la bonne réponse à un ENT est le code `NOERROR`, sans enregistrements (ce qu'on appelle parfois `NODATA`, bien que ce dernier ne soit pas un code de retour explicite dans le DNS). Mais certains serveurs boguées répondent à la place `NXDOMAIN`, code qui indique normalement que le domaine n'existe pas (ce qui est faux). Voici ce que répond djbdns à une requête sur l'ENT `temporary.cr.yp.to` :

```
% dig A temporary.cr.yp.to
...
;; ->>HEADER<<- opcode: QUERY, status: NXDOMAIN, id: 23636
```

C'est bien un ENT puisqu'il y a des noms en dessous (par exemple `pairings.temporary.cr.yp.to`). Le résolveur qui se fierait à ce `NXDOMAIN` croirait alors que sa recherche est terminée et que le nom demandé n'existe pas. C'est pour cela que les mises en œuvre existantes de la "*QNAME minimisation*" ont des comportements spécifiques pour les réponses `NXDOMAIN`, pour contourner cette bogue.

La protection de la vie privée fait qu'on enverra moins de données. C'est bien le but. Résultat, les serveurs faisant autorité et les "*sniffeurs*" recevront moins d'information. Cela peut gêner certains gros silos de données, qui exploitaient cette information <http://blogs.verisigninc.com/blog/entry/minimum_disclosure_what_information_does>.

Et les performances ? Elles seront parfois meilleures et parfois pires, avec la "*QNAME minimisation*". Meilleures car le résolveur enverra moins de requêtes aux serveurs faisant autorité. Aujourd'hui, si un résolveur reçoit trois requêtes successives, pour `A.example`, `B.example` et `C.example`, les trois requêtes seront envoyées aux serveurs racine, et donneront toutes les trois un `NXDOMAIN` (puisque `.example` n'est pas délégué). Avec la "*QNAME minimisation*", seule la première requête déclenchera une demande à la racine, pour le nom `example`. Cela suffira au résolveur.

Par contre, les performances peuvent se dégrader dans certains cas. Si un nom comporte beaucoup de composants (c'est fréquent dans `ip6.arpa`), la recherche initiale des limites de zone nécessitera bien plus de paquets. Ceci dit, cela ne durera que le temps de remplir le cache, tout ira plus vite après, une fois que le résolveur connaitra les limites de zone.

Ce RFC est issu du projet « "*DNS privacy*" », lancé initialement au CENTR, puis passé à l'IETF (le premier RFC de ce projet avait été le RFC 7626 ; comme lui, cette idée de "*QNAME minimisation*" était née dans un avion de la KLM).

À noter que Verisign a un brevet dont ils prétendent qu'il couvre la QNAME minimisation <https://datatracker.ietf.org/ipr/search/?submit=draft&id=draft-ietf-dnsop-qname-minimisation
Ils promettent une licence (attention, le diable est dans les détails) gratuite et non-discriminatoire. Ces brevets ont bien perturbé la réflexion du groupe de travail. Personnellement, je pense que ce brevet n'a pas de sens : l'idée de "*QNAME minimisation*" est évidente et avait déjà été discuté plusieurs fois, mais sans laisser de trace écrite précise, ce qui permet à Verisign de prétendre qu'il n'y a pas de "*prior art*". Ce n'est sans doute pas un hasard si les deux premières mises en œuvre de la "*QNAME minimisation*" ont été faites en Europe, où il n'y a (normalement) pas de brevet logiciel. Ceci dit, lors des discussions sur la licence de ces brevets, en marge de la réunion IETF d'Honolulu, c'est Verisign qui a payé les boissons, reconnaissons-leur ce mérite.

La "*QNAME minimisation*" est mise en œuvre dans Unbound et dans le résolveur Knot <https://gitlab.labs.nic.cz/knot/resolver> (ce dernier n'étant pas encore officiellement publié). Pour Knot (qui le fait par défaut), voici le résultat vu par tcpdump d'une requête `dig -x` d'une adresse IPv6. Par exemple, le serveur racine n'a reçu qu'une demande pour `.arpa`. Notez aussi que Knot fait varier la casse (une protection contre certains empoisonnements) :

```
02:36:39.673268 IP6 2400:8900::f03c:91ff:fe69:60d3.54216 > 2001:e30:1c1e:1::333.53: 38773% [1au] NS? aRpA.
02:36:40.114074 IP6 2400:8900::f03c:91ff:fe69:60d3.59934 > 2001:dc3::35.53: 22056% [1au] NS? Ip6.aRPa. (37)
02:36:40.428545 IP6 2400:8900::f03c:91ff:fe69:60d3.47793 > 2001:500:86::86.53: 43002% [1au] NS? 2.ip6.arPA.
...
```

Pour se protéger contre les serveurs bogués dont je parlais plus haut (ceux qui répondent `NXDOMAIN` en cas d'ENT), Knot réessaie avec le QNAME complet lorsqu'il reçoit un `NXDOMAIN` (les deux dernières lignes). Mauvais pour la vie privée mais sans doute nécessaire aujourd'hui (ici, par la faute d'Akamai) :

```
02:34:45.050913 IP 106.186.29.14.51228 > 128.175.13.17.53: 24014% [1au] A? WwW.UpENn.edU. (42)
02:34:45.227102 IP 128.175.13.17.53 > 106.186.29.14.51228: 24014*- 2/0/1 CNAME www.upenn.edu-dscg.edgesuite.
02:34:45.228413 IP6 2400:8900::f03c:91ff:fe69:60d3.46525 > 2001:503:231d::2:30.53: 52576% [1au] NS? edGeSUIt
02:34:45.297319 IP6 2001:503:231d::2:30.53 > 2400:8900::f03c:91ff:fe69:60d3.46525: 52576- 0/17/15 (1034)
02:34:45.298284 IP 106.186.29.14.45604 > 23.61.199.64.53: 22228 [1au] NS? EdU-DScG.EdGesUITe.nET. (51)
02:34:45.373238 IP 23.61.199.64.53 > 106.186.29.14.45604: 22228 NXDomain*- 0/1/1 (114)
02:34:45.373795 IP 106.186.29.14.34320 > 72.246.46.66.53: 1355 [1au] A? WWW.UPenN.edu-dSCG.EdgESuItE.net. (6
```

Un autre exemple de ce repli sur les requêtes classiques est donné ici, lorsque je demande `www.long.verylong.d` Comme le TLD `.example` n'existe pas, Knot débraye hélas la "*QNAME minimisation*" :

```
20:08:49.615421 IP6 2400:8900::f03c:91ff:fe69:60d3.51723 > 2001:1398:1:21::8001.53: 19881% [1au] NS? ExaMpLE
20:08:49.900009 IP6 2400:8900::f03c:91ff:fe69:60d3.59917 > 2001:6d0:6d06::53.53: 40708% [1au] AAAA? www.LONg
```

Même chose avec un ENT où la réponse est pourtant correcte. Knot se méfie et réessaie sans "*QNAME minimisation*" :

```
20:14:15.479872 IP6 2400:8900::f03c:91ff:fe69:60d3.45418 > 2001:67c:1010:11::53.53: 18200% [1au] NS? gOUV.Fr
20:14:15.740424 IP 106.186.29.14.33850 > 194.0.36.1.53: 54260% [1au] A? www.ST-cYr.TerRE.DeFeNSe.GouV.fR. (6
```

Lorsque le cache commence à être rempli, Knot a besoin de moins de requêtes. Ici, je lui ai demandé l'adresse de `www.bortzmeyer.org`, et il connaissait déjà les serveurs de `.org`, il passe donc directement à la question « quels sont les serveurs de noms de `bortzmeyer.org` ? » :

```
20:08:20.420757 IP 106.186.29.14.41889 > 199.249.120.1.53: 39126% [1au] NS? bOrtzMEYeR.oRg. (43)
20:08:20.941797 IP 106.186.29.14.35536 > 217.70.190.232.53: 33709% [1au] AAAA? Www.bOrtZmEyER.Org. (47)
```

Unbound a la "*QNAME minimisation*" depuis la version 1.5.7 (sortie en décembre 2015, cf. l'historique de ce travail <https://www.nlnetlabs.nl/bugs-script/show_bug.cgi?id=648>). Ce n'est pas activé par défaut, il faut mettre dans la configuration :

```
server:
    qname-minimisation: yes
```

(Un pcap est disponible <https://www.shaftinc.fr/owncloud/index.php/s/c2S3Qg5RYBO3cHg>.) Pour vérifier si votre résolveur met bien en œuvre la "*QNAME minimisation*", vous pouvez tester avec le domaine `qnamemintest.internet.nl`. Ici, le résolveur est un Unbound récent :

```
% dig +nodnssec +short TXT qnamemintest.internet.nl
a.b.qnamemin-test.internet.nl.
"HOORAY - QNAME minimisation is enabled on your resolver :)!"
```

Avec un résolveur traditionnel (ici, Verisign Public DNS <https://www.verisign.com/en_US/innovation/public-dns/index.xhtml>, qui utilise Unbound mais une vieille version) :

```
% dig @64.6.64.6 +nodnssec +short TXT qnamemintest.internet.nl
a.b.qnamemin-test.internet.nl.
"NO - QNAME minimisation is NOT enabled on your resolver :("
```

À noter qu'à la réunion de l'OARC <https://indico.dns-oarc.net/event/22/> à Buenos Aires, Ralph Dolmans a présenté un très intéressant exposé technique sur la mise en œuvre de la "*QNAME minimisation*" dans Unbound.

Questions articles sur la "*QNAME minimisation*", notez celui de Robert Graham <http://blog.erratasec.com/2017/08/query-name-minimization.html>.