# RFC 7108 : A Summary of Various Mechanisms Deployed at L-Root for the Identification of Anycast Node

Stéphane Bortzmeyer
<stephane+blog@bortzmeyer.org>

Première rédaction de cet article le 15 janvier 2014

Date de publication du RFC : Janvier 2014

https://www.bortzmeyer.org/7108.html

---

L'"*anycast*" a des tas d'avantages pour les serveurs DNS mais il a au moins un inconvénient : le débogage peut être plus difficile, car deux points de mesure peuvent tomber sur une instance différente du service. Si toutes les instances sont réellement identiques, ce n'est pas un problème. Mais si l'une d'elles a un comportement, ou des données, légèrement différent, de laquelle s'agit-il lorsqu'un utilisateur se plaint que « j'interroge `ns2.nic.example` et sa réponse n'est pas correcte ? » Ce RFC documente l'ensemble des techniques utilisées par l'un des serveurs racine du DNS, le serveur L <http://l.root-servers.org/> géré par l'ICANN.

Le problème se résume donc à « j'interroge `l.root-servers.net` et je voudrais savoir sur quelle instance je suis tombé, et ceci sans faire appel à des connaissances internes à l'ICANN ». On peut s'intéresser à cette information par curiosité, ou bien pour déboguer un problème (comme dans l'exemple du paragraphe précédent) ou encore pour évaluer la bonne distribution des sites "*anycast*" (comme dans mon article aux RIPE labs <https://labs.ripe.net/Members/stephane_bortzmeyer/using-atlas-udm-to-find-the-popular-instances-of-a-dns-anycast-name-server>).

Est-ce une bonne chose que cette information soit disponible à l'extérieur ? Oui, répond clairement l'ICANN, on gère un service qui est public et destiné au public et cette transparence est nécessaire, notamment du point de vue opérationnel. Et puis la localisation de ces serveurs n'est pas vraiment un secret (contrairement à ce qu'on a parfois lu dans des articles de presse sensationnalistes). Félicitons donc l'ICANN pour cet effort de publication (tous les serveurs racine du DNS ne le font pas, loin de là, cf. section 6 pour des éléments à ce sujet) et passons à la technique.

La gestion de services "*anycast*" est décrite dans le RFC 4786 [1]. Cette technique permet de répartir sur toute la planète les serveurs qui répondent à une adresse (pour L, ce sont actuellement `199.7.83.42`

1. Pour voir le RFC de numéro NNN, `https://www.ietf.org/rfc/rfcNNN.txt`, par exemple `https://www.ietf.org/rfc/rfc4786.txt`

en IPv4 et 2001:500:3::42 en IPv6). Beaucoup d'information est disponible en ligne sur la gestion de la racine, notamment en <http://www.root-servers.org/>. Une liste des instances de « L-root » (actuellement 143 sites et 273 machines) est également en ligne <http://l.root-servers.org/>.

Comment sont nommées ces instances de L (section 3 de notre RFC) ? Chaque copie de L-root a un nom, <IATA Code><NN>.l.root-servers.org où <IATA Code> est un code IATA identifiant l'aéroport le plus proche. Cette technique de nommage des équipements réseau est très courante chez les opérateurs Internet. Par exemple, cdg est l'aéroport de Roissy près de Paris. Le nombre ¡NN¿ sert à distinguer deux machines proches du même aéroport (par exemple dans le même centre de traitement de données). Si un site est situé à mi-chemin entre deux aéroports, on en choisit arbitrairement un.

Bien, c'est très joli, ces noms mais comment trouve-t-on celui de l'instance de L-root qui nous a répondu ? La section 4 du RFC décrit les différentes méthodes possibles. Commençons par NSID ("*Name Server Identifier*", normalisé dans le RFC 5001). C'est une option du DNS qui permet de demander au serveur, s'il le veut bien, de renvoyer son nom. Testons là avec dig :

```
% dig +nsid @l.root-servers.net SOA .
...
;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags: do; udp: 4096
; NSID: 6b 62 70 30 31 2e 6c 2e 72 6f 6f 74 2d 73 65 72 76 65 72 73 2e 6f 72 67  \
        (k) (b) (p) (0) (1) (.) (l) (.) (r) (o) (o) (t) (-) (s) (e) (r) (v) (e) (r) (s) (.) (o) (r) (g)
```

La question posée (le SOA de la racine) ne nous intéresse pas, on voulait juste le nom du serveur, ici kbp01.l.root-servers.org (donc à Kiev). C'est la méthode recommandée et la plus fiable. Si vous voulez signaler à l'ICANN un problème avec L-root, c'est cette technique qu'il faut utiliser pour envoyer un rapport de bogue utile. C'est d'ailleurs vrai pour la majorité des serveurs anycast DNS (par exemple, le serveur d.nic.fr qui fait autorité pour .fr accepte également cette option). Par contre, dans certains cas, cela ne marchera pas car NSID dépend de EDNS (RFC 6891) et certains équipements réseaux bloquent (bêtement) EDNS.

Une autre technique existe, bien plus ancienne que NSID et qui est sans doute plus connue des administrateurs réseaux : interroger le serveur sur les noms hostname.bind ou id.server, avec le type TXT et la classe CH. Par exemple :

```
% dig @l.root-servers.net CH TXT hostname.bind
...
;; ANSWER SECTION:
hostname.bind. 0 CH TXT "lys01.l.root-servers.org"
...
```

(C'est près de Lyon.) Notez bien que .bind et .server ne sont pas des vrais TLD. Ils n'ont de signification que locale, sur le serveur en question. Chaque machine répondra de manière différente. Comme son nom l'indique, hostname.bind a été à l'origine introduit par le logiciel BIND mais a depuis été adopté par d'autres (L-root utilise nsd <https://www.bortzmeyer.org/nsd.html>). id.server, qui donne le même résultat, a été créé pour faire moins spécifique à BIND mais n'a jamais été très populaire.

hostname.bind et id.server ne dépendent pas d'EDNS et ne nécessitent pas un client DNS qui comprend l'option NSID. Mais ils ont tous les deux un gros défaut : si on reçoit une réponse surprenante

et que, dans la foulée, on fait une requête `hostname.bind` pour enquêter, rien ne dit que le routage n'aura pas changé entre temps, et qu'on n'aura pas désormais affaire à une autre instance. NSID, où l'option voyage en même temps que la requête, n'a pas cet inconvénient.

Aussi bien NSID que `hostname.bind` nécessitent un client DNS (comme dig ou drill <http://www.nlnetlabs.nl/projects/drill/>), ce que tout le monde n'a pas sur sa machine (notamment sur Windows). Et, même si on a un client DNS, taper la ligne de commande correctement peut être un exploit inaccessible à certains utilisateurs. Lorsqu'un problème se pose avec, semble t-il, une instance spécifique de L-root, il vaudrait mieux pouvoir donner des instructions simples à l'utilisateur qui signale le problème. C'est le but de la dernière technique, `identity.l.root-servers.org`, présentée en section 4.4. Chaque instance de L-root fait autorité pour ce nom mais donne une réponse différente, révélant l'identité de l'instance. Ce nom a un enregistrement TXT mais surtout un enregistrement A (une adresse IPv4) donc, si on peut interroger ce nom avec un client DNS, comme ci-dessus :

```
% dig identity.l.root-servers.org
...
;; ANSWER SECTION:
identity.l.root-servers.org. 3600 IN A 77.88.206.134
```

(en profitant du fait que le type A est la valeur par défaut pour dig donc n'a pas besoin d'être spécifiée), on peut aussi demander à l'utilisateur de juste faire :

```
% ping identity.l.root-servers.org
PING identity.l.root-servers.org (77.88.206.134) 56(84) bytes of data.
64 bytes from hostmaster.kiev.customer.top.net.ua (77.88.206.134): icmp_req=2 ttl=47 time=86.5 ms
64 bytes from hostmaster.kiev.customer.top.net.ua (77.88.206.134): icmp_req=4 ttl=47 time=90.7 ms
...
```

et d'indiquer la valeur qu'il voit pour l'adresse. Au passage, la traduction d'adresse en nom nous indique qu'on touche bien l'instance de Kiev. ping, lui, est installé partout. **C'est le gros intérêt de cette technique, son accessibilité pour des non-techniciens.** Elle est spécifique à L-root (alors que les deux précédentes sont « standard » et largement déployées). Notez bien que ping est juste un moyen (simple et toujours disponible) de faire une résolution de nom en adresse IPv4. Rien ne garantit que cette adresse sera routable, c'est juste un identificateur unique permettant de distinguer les différentes instances de L-root.

Si on a un client DNS (ici, drill <http://www.nlnetlabs.nl/projects/drill/>), on peut aussi demander le TXT, plus riche :

```
% drill identity.l.root-servers.org TXT
...
;; ANSWER SECTION:
identity.l.root-servers.org. 3600 IN TXT "kbp01.l.root-servers.org" "Kiev" "" "Ukraine" "Europe"
```

On trouve le nom du serveur, suivant le schéma de nommage indiqué plus haut, la ville en clair, le pays et la région (au sens ICANN du terme <http://meetings.icann.org/regions>).

Pour cette dernière technique, celle avec `identity.l.root-servers.org`, on n'a pas interrogé directement `l.root-servers.net` mais on est passé par le serveur récursif (le résolveur) utilisé par la machine de l'utilisateur. Le résolveur peut être sur un réseau différent et donc parfois utiliser une

instance de L-root différente de celle de son client, lorsque celui-ci fait des requêtes directes. En outre, le cache du résolveur peut compliquer l'interprétation des résultats. Par exemple, les réponses à une requête A et à une TXT peuvent être incohérentes, si elles sont entrées dans le cache à des moments différents et que les routes avaient changé entre temps.

Si vous voulez voir la liste complète des instances de L-root, elle est stockée dans le DNS, sous forme d'un (très gros) enregistrement TXT dans `nodes.l.root-servers.org`. On l'interroge en TCP car la réponse sera en général trop grande pour le serveur DNS s'il utilise UDP :

```
% dig +tcp nodes.l.root-servers.org TXT
...
;; ANSWER SECTION:
nodes.l.root-servers.org. 28681 IN TXT "pom01.l.root-servers.org" "Port Moresby" "" "Papua New Guinea" "Asia
nodes.l.root-servers.org. 28681 IN TXT "ppt01.l.root-servers.org" "Papeete" "Tahiti" "French Polynesia" "Asi
nodes.l.root-servers.org. 28681 IN TXT "ppt02.l.root-servers.org" "Papeete" "Tahiti" "French Polynesia" "Asi
nodes.l.root-servers.org. 28681 IN TXT "prg01.l.root-servers.org" "Prague" "" "Czech Republic" "Europe"
...
```

Si vous aimez les détails techniques sur le fonctionnement de `identity.l.root-servers.org`, voyez la section 5. `identity.l.root-servers.org` est une sous-zone de `l.root-servers.org`, déléguée à un seul serveur de noms, dont l'adresse IP est dans le même préfixe que `l.root-servers.net` :

```
% dig NS identity.l.root-servers.org
...
;; ANSWER SECTION:
identity.l.root-servers.org. 3600 IN NS beacon.l.root-servers.org.
...
```

Chacun des serveurs sert une zone `identity.l.root-servers.org` différente, reflétant la localisation du serveur. La zone `identity.l.root-servers.org` n'est donc pas cohérente et c'est fait exprès. Cela n'empêcherait pas la signature avec DNSSEC mais cela la compliquerait (installer les logiciels et générer des clés sur chaque nœud...) et donc `identity.l.root-servers.org` n'est pas signé (de toute façon, les zones au-dessus ne le sont pas non plus).