Trouver si un domaine a des jokers

Stéphane Bortzmeyer
<stephane+blog@bortzmeyer.org>

Première rédaction de cet article le 10 octobre 2008

https://www.bortzmeyer.org/recherche-jokers-dns.html

Le DNS permet à un domaine d'avoir des **jokers**, des enregistrements qui feront que le serveur DNS répondra systématiquement à tous les noms de domaine, qu'ils « existent » ou pas. Comment tester, sans accès aux données entrées, si un domaine a de tels jokers ?

Les jokers ("*wildcards*") sont un des points les plus contestés du DNS. Certains TLD les utilisent de façon à rabattre du trafic (essentiellement des fautes de frappe) vers un serveur de publicité (c'est ce que fait aujourd'hui .tk ou ce qu'à fait, avec beaucoup plus de publicité, .com, dans l'affaire connue - bien à tort - sous le nom de Sitefinder ; le registre de .fr, s'est par contre engagé à ne pas le faire <http://www.afnic.fr/actu/nouvelles/international/NN20030918>).

Pire, certains FAI peu scrupuleux utilisent une technique similaire sur les résolveurs DNS qu'ils mettent à la disposition de leurs clients, malgré l'avertissement du RFC 4924 [1] (voir aussi le communiqué de l'AFNIC <http://www.afnic.fr/actu/nouvelles/general/NN20070910>).

Comment détecter qu'il y a des jokers ? En lisant le RFC 1034, cela semble simple. Les jokers sont représentés par le caractère * et, si on veut tester `example.org`, on fait une requête DNS pour `*.example.org` et on voit si le domaine existe ou pas (attention à échapper le caractère * pour le shell Unix) :

```
% dig ANY \*.example.org
...
;; ->>HEADER<<- opcode: QUERY, status: NXDOMAIN, id: 48646
;; flags: qr rd ra; QUERY: 1, ANSWER: 0, AUTHORITY: 1, ADDITIONAL: 1
```

Le `NXDOMAIN` ("*No Such Domain*") indique que le domaine n'existe pas et qu'il n'y a donc pas de jokers. Avec `.tk`, par contre :

1. Pour voir le RFC de numéro NNN, `https://www.ietf.org/rfc/rfcNNN.txt`, par exemple `https://www.ietf.org/rfc/rfc4924.txt`

```
% dig ANY \*.tk
...
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 63123
;; flags: qr rd ra; QUERY: 1, ANSWER: 5, AUTHORITY: 7, ADDITIONAL: 1
...
;; ANSWER SECTION:
*.tk.                   86400   IN      MX      20 MX-HOST.DOT.tk.
```

Ici, le domaine existe : `.tk` a bien des jokers.

Mais le monde est plus compliqué que cela : on trouve des domaines qui ne répondent pas pour * mais qui ont quand même des jokers, et aussi le contraire. Il faut donc utiliser un algorithme plus perfectionné. Voici celui que j'ai développé (avec l'aide de Joe Abley et de plusieurs autres) :

— Envoyer une requête pour `*.DOMAINE.example`,
— Envoyer trois requêtes pour des noms choisis aléatoirement dans `DOMAINE.example`,
— Vérifier que les réponses coïncident.

Notez qu'aucun algorithme de recherche de jokers n'est parfait. Celui-ci a des faux négatifs (par exemple si la malchance fait que les noms choisis au hasard existent réellement) et des faux positifs (par exemple si les serveurs DNS ont des réponses variées - plusieurs serveurs avec des adresses IP différentes à chaque requête).

Le code de mise en œuvre en Python, utilisant dnspython <`https://www.bortzmeyer.org/dnspython.html`>, est disponible (en ligne sur `https://www.bortzmeyer.org/files/DNSwildcards.py`). Voici quelques tests :

```
% python DNSwildcards.py fr
fr does not have A wildcards
% python DNSwildcards.py -t TXT fr
fr does not have TXT wildcards

% python DNSwildcards.py tk
tk has A wildcards (['193.33.61.2', '195.20.32.103', '209.172.59.196', '217.119.57.22'])

% python DNSwildcards.py elastoplast.fr
elastoplast.fr has wildcards but no data for type A
% python DNSwildcards.py -t MX elastoplast.fr
elastoplast.fr has MX wildcards ([<DNS IN MX rdata: 10 mail1.beiersdorf.com.>, <DNS IN MX rdata: 50 mail2.be
```

Et, si on indique explicitement le résolveur (ici, celui d'OpenDNS, un service de résolution qui renvoie de fausses réponses avec de la publicité <`https://www.bortzmeyer.org/opendns-non-merci.html`>) :

```
# Avec le résolveur standard, cela marche
% python DNSwildcards.py bortzmeyer.org
bortzmeyer.org does not have A wildcards

# Avec OpenDNS, on récupère toujours l'adresse du serveur de publicités
% python DNSwildcards.py -r 208.67.222.222 bortzmeyer.org
bortzmeyer.org has A wildcards (['208.69.34.132'])
```