RFC 2136 : Dynamic Updates in the Domain Name System (DNS UPDATE)

Stéphane Bortzmeyer
<stephane+blog@bortzmeyer.org>

Première rédaction de cet article le 16 janvier 2007. Dernière mise à jour le 16 octobre 2010

Date de publication du RFC : Avril 1997

https://www.bortzmeyer.org/2136.html

Ce RFC étend officiellement le traditionnel protocole DNS aux mises à jour dynamiques. Traditionnellement, les serveurs DNS faisant autorité utilisaient l'information contenue dans un fichier statique. À partir du moment où existe le "*Dynamic DNS Update*", ils peuvent aussi être mis à jour en temps quasi-réel, par exemple pour suivre les changements de topologie.

Ainsi, un ordinateur portable qui se déplace mais souhaiterait être toujours connu sous son nom (mettons `monportable.dyn.example.org`) peut utiliser les mises à jour dynamiques pour prévenir le serveur DNS maître de sa nouvelle adresse IP. La mise à jour peut être lancée par le portable lui-même (le serveur DHCP peut aussi être client "*Dynamic DNS Update*" pour que les adresses qu'il attribue se retrouvent dans le DNS (option ddns-update-style du serveur DHCP de l'ISC).

L'idée originale avait été documentée dans l'article « "*The Design and Implementation of the BIND Servers*" <http://www.eecs.berkeley.edu/Pubs/TechRpts/1984/CSD-84-177.pdf> » en 1984 (et mise en œuvre dans le serveur de noms développé par les auteurs) mais n'a été normalisé que longtemps après.

Notre RFC spécifie donc la façon un peu complexe dont la requête DNS doit être composée (certains champs du protocole ont dû être réaffectés). Une des particularités des mises à jour dynamiques est l'existence de **pré-conditions** à l'opération. Si ces pré-conditions, indiquées par le client, ne sont pas remplies, la mise à jour n'est pas faire par le serveur (les deux implémentations citées ci-dessous permettent de déclarer ces pré-conditions).

Le programme nsupdate, distribué avec BIND, permet de faire facilement ces mises à jour. Il existe aussi des bibliothèques pour des programmes qu'on écrit soi-même comme l'excellent Net : :DNS <http://www.net-dns.org/> pour Perl (un exemple complet <http://www.net-dns.org/docs/Net/DNS/Update.html> est en ligne). Voici un exemple de script lançant nsupdate (sur Debian, c'est à mettre dans le fichier `/etc/dhclient-exit-hooks` (ou dans le répertoire `/etc/dhcp3/dhclient-exit-hooks.d`), il sera ainsi lancé par le client DHCP de l'ISC lorsqu'il aura obtenu une adresse IP).

```
#!/bin/sh

cd /etc/bind

if [ x$reason = xBOUND ]; then
# Attention, le fichier ".key" doit apparemment également être là
nsupdate -kKexample-dyn-update.+157+18685.private -d <<EOF
  server nsupdate.example.org
  zone dyn.example.org
  update delete monportable.dyn.example.org
  update add monportable.dyn.example.org 300 A $new_ip_address
  send
EOF
fi
```

[Je suppose que cela peut fonctionner avec d'autres clients DHCP comme pump mais je n'ai pas testé.] nsupdate utilise ici TSIG (l'option -k) pour s'authentifier. Il supprime l'ancienne liaison nom-¿adresse et en ajoute une nouvelle. La variable new_ip_address a été créée par le client DHCP avant qu'il n'appelle notre script.

Comme l'usage de TSIG (cf. RFC 2845 [1]) nécessite que les horloges des deux machines soient synchronisées, il faut veiller à ce que ces horloges soient correctes (autrement, BIND notera quelque chose du genre `request has invalid signature: TSIG example-dyn-update: tsig verify failure (BADTIME)`).

On peut aussi faire les mises à jour depuis son langage de programmation favori et on trouve de nombreux exemples en ligne comme <http://www.net-dns.org/docs/Net/DNS/Update.html> pour Perl ou bien <http://www.dnspython.org/examples.html> pour Python. (Voici un exemple plus complet en Python (en ligne sur `https://www.bortzmeyer.org/files/dns-dynamic-update.py`), avec dnspython <https://www.bortzmeyer.org/dnspython.html>.)

Sur le serveur DNS BIND, il aura fallu générer la clé :

```
% dnssec-keygen -a HMAC-MD5 -b 512 -n HOST example-dyn-update
```

La commande ci-dessus créera les fichiers `.key` et `.private` qu'on pourra envoyer aux clients. À noter que cette commande nécessite beaucoup d'entropie pour son générateur aléatoire et qu'elle peut donc prendre de nombreuses minutes à s'exécuter, quelle que soit la vitesse de la machine.

La configuration correspondante, est :

```
// For dynamic updates
key "example-dyn-update." {
        algorithm hmac-md5;
        secret "CLE-SECRET=";
};
zone "dyn.example.org" {
```

1. Pour voir le RFC de numéro NNN, `https://www.ietf.org/rfc/rfcNNN.txt`, par exemple `https://www.ietf.org/rfc/rfc2845.txt`

```
        type master;
        // Old and insecure way: authenticate by IP address
        //allow-update {
        //      my-networks;
        //};
        // http://www.linux-mag.com/2001-11/bind9_01.html
        // See #2700
        allow-update {
                key "example-dyn-update.";
        };
        // Future work:
        // http://www.oreilly.com/catalog/dns4/chapter/ch11.html#38934
        //update-policy {
        //      grant *.dyn.example.org self IGNORED;
        //      deny * wildcard *;
        //};
        // If you want also to restrict it to some IP addresses (untested):
        //acl address_allow { 10/8; };
        //acl address_reject { !address_allow; any; };
        //allow-update { !reject; key "...."; };
        file "/etc/bind/db.dyn.example.org";
};
```

.

Lors d'une mise à jour dynamique, on trouvera dans le journal de BIND quelque chose comme :

```
23-May-2012 22:19:56.255 client 127.0.0.1#47960: signer "foobar-example-dyn-update" approved
23-May-2012 22:19:56.255 client 127.0.0.1#47960: updating zone 'foobar.example/IN': deleting rrset at 'www.fooba
23-May-2012 22:19:56.255 client 127.0.0.1#47960: updating zone 'foobar.example/IN': adding an RR at 'www.foobar.
```

Certains registres de noms de domaines, comme Nominet pour .uk utilisent ces mises à jour dynamiques pour ajouter ou modifier des domains (Nominet l'a lancé en février 2005 <http://lists.nominet.org.uk/pipermail/nom-announce/2005-February/000147.html> et a fait une bonne présentation au RIPE-NCC <http://www.ripe.net/ripe/meetings/ripe-52/presentations/ripe52-dns-dynamic-updates.pdf>).

On notera que MS-Windows, par défaut, « téléphone à la maison », c'est-à-dire qu'il tente de mettre à jour les serveurs DNS avec son adresse. S'il n'a pas été correctement configuré, sa requête parvient aux serveurs de la racine du DNS, leur imposant une forte charge. C'est pour cela qu'il est souvent recommandé de le débrayer <http://www.caida.org/research/dns/disable_dns_updates.xml>.