Nouvelle version de Zonecheck, la 3.0, avec tests DNSSEC

Stéphane Bortzmeyer
<stephane+blog@bortzmeyer.org>

Première rédaction de cet article le 22 juin 2010

http://www.bortzmeyer.org/zonecheck-3-0.html

Le 22 juin, une nouvelle version de Zonecheck <http://www.zonecheck.fr/> a été publiée <http://lists.gnu.org/archive/html/zonecheck-announce/2010-06/msg00000.html>, apportant notamment les tests DNSSEC.

Écrit en Ruby, Zonecheck <http://www.zonecheck.fr/> est un logiciel de test de la configuration d'une zone DNS, pour voir si elle est correctement installée sur les serveurs de noms. Cette vérification nécessiterait à la main des dizaines d'appels à la commande dig, pour s'assurer que tout est correct. Zonecheck automatise le tout. D'une manière générale, tester doit être fait avec un logiciel car c'est une opération longue et fastidieuse, qu'un humain a peu de chances de mener jusqu'au bout correctement.

Ainsi, je peux voir si mes serveurs de noms sont tous bons :

```
% zonecheck bortzmeyer.org
ZONE  : bortzmeyer.org.
NS <= : a.dns.gandi.net. [217.70.179.40]
NS    : c.dns.gandi.net. [217.70.182.20]
NS    : b.dns.gandi.net. [217.70.184.40]

       ______________________
      ,----------------------.|
~~~~ |    avertissement     || ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
      `----------------------'
a> Les serveurs de nom font tous partie du même AS
 | Conseil: ZoneCheck
 |   Afin d'éviter de perdre la connectivité avec les serveurs DNS
 | autoritaires en cas de problèmes de routage dans le Système Autonome,
 | il est conseillé d'héberger les serveurs sur différents AS.
 `----- -- -- - -  -
 :   Tous les serveurs de noms font partie du même Système Autonome (AS
 : numéro 29169), essayez d'en héberger certains sur un autre.
 `..... .. .. . .  .
=> générique

...
==> SUCCÈS (mais 5 avertissement(s))
```

Et, à quelques avertissements près, c'est bon.

L'installation de Zonecheck 3 nécessite la bibliothèque Ruby DNSruby <http://rubyforge.org/projects/dnsruby> qui ne semble pas exister en paquetage pour ma Debian, je l'ai donc installé à la main, avec RubyGems :

```
% sudo aptitude install rubygems # Si nécessaire
% sudo gem install dnsruby
```

(DNSruby remplace l'ancienne bibliothèque Nresolv, utilisée par Zonecheck 2, est permet notamment les tests DNSSEC.)

Principale nouveauté de la version 3 : DNSSEC. Par défaut, Zonecheck 3 détecte tout seul si une zone est signée ou pas, mais les tests DNSSEC, dans ce cas, ne mènent qu'à des avertissements. Si on veut être sûr que DNSSEC est activé sur la zone, il faut utiliser l'option `-s` (ou `--securedelegation`) :

```
% zonecheck iis.se
...
==> SUCCESS (but 7 warning(s))
```

Aucun message mais les tests DNSSEC ont été faits (simplement, Zonecheck n'avait rien à signaler). Si on veut imposer qu'ils réussissent :

```
% zonecheck -s "" iis.se
...
==> SUCCESS (but 7 warning(s))
```

Et sur une zone DNSSEC qui a des problèmes, cela donne quoi ? Comme beaucoup de domaines en `.GOV`, `uspto.gov` est configuré avec les pieds :

```
% zonecheck -s "" uspto.gov
ZONE  : uspto.gov
NS <= : dns1.uspto.gov [151.207.240.50]
NS    : dns2.uspto.gov [151.207.246.51]
...
      _____________
     ,-------------.|
~~~~ |    fatal    || ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
     `-------------'
f> [TEST supports EDNS]: DNS Timeout
=> dns2.uspto.gov/151.207.246.51

==> FAILURE (and 3 warning(s))
```

DNSSEC impose en effet EDNS (RFC 2671 [1]) et ce serveur ne répond pas. Que se passe t-il ? *"DNS timeout"*, nous dit Zonecheck. Vérifions avec dig :

1. Pour voir le RFC de numéro NNN, `https://www.ietf.org/rfc/rfcNNN.txt`, par exemple `https://www.ietf.org/rfc/rfc2671.txt`

```
% dig +dnssec @dns2.uspto.gov ANY uspto.gov

; <<>> DiG 9.6-ESV-R1 <<>> +dnssec @dns1.uspto.gov ANY uspto.gov
; (1 server found)
;; global options: +cmd
;; connection timed out; no servers could be reached
```

Le même serveur répond si on lui demande `dig +dnssec @dns1.uspto.gov DNSKEY uspto.gov` car la réponse, dans ce cas, est plus petite (moins de 1200 octets). Il y a donc clairement un problème de MTU à l'office des brevets états-uniens, qui plante tout client qui essaie de faire du DNSSEC avec eux (le problème leur a été signalé il y a des semaines mais la bureaucratie locale ne semble pas pressée de le résoudre).

Zonecheck permet aussi d'indiquer la clé qu'on veut voir dans la zone (telle qu'elle est annoncée par le parent ou bien via une page Web sécurisée). Par exemple, si je sais que l'enregistrement DS (cf. RFC 4034, section 5) de `frobbit.se` est censé être `3E91111144999944301CD5E4E33692352C0BD024` :

```
%  zonecheck -s "DS:3E91111144999944301CD5E4E33692352C0BD024:SHA-1" frobbit.se
ZONE  : frobbit.se
NS <= : ns1.frobbit.se [208.79.80.118]
NS    : ns.cafax.se [192.71.228.17]
NS    : ns2.frobbit.se [85.30.129.39, 2A02:80:3FFE::39]
...
w> The length of the key should be higher for this algorithm
 :   the key length is 1024 bits for the algorithm : RSA SHA 1
 `..... .. .. . .  .
=> ns2.frobbit.se/85.30.129.39
=> ns.cafax.se/192.71.228.17
=> ns1.frobbit.se/208.79.80.118
=> ns2.frobbit.se/2A02:80:3FFE::39

==> SUCCESS (but 13 warning(s))
```

Et, à part que la clé est un peu courte, le domaine est bien signé comme indiqué chez son parent.

On peut automatiser cette tâche de trouver le DS, chez le registre DLV (RFC 5074) de l'ISC par exemple. Ce petit script shell le fait :

```
dnssechashalgo=1
dnssechashalgoname=SHA-1

zonecheck-dlv() {
    if [ -z "$1" ]; then
echo "Usage: $0 domain-name"
return 1
    fi
    domain=$1
    sha1ds=$(dig +short DLV $domain.dlv.isc.org | awk "/ (3|5|7) +$dnssechashalgo / {print \$4}" | tail -n 1)
    if [ -z "$sha1ds" ]; then
echo "$domain does not seem to be in dlv.isc.org"
return 1
    fi
    zonecheck -s "DS:$sha1ds:$dnssechashalgoname" $domain
}
```

Ce script peut être utilisé ainsi :

```
%  zonecheck-dlv   secure64.com
ZONE  : secure64.com
NS <= : ns1.secure64.com [64.92.220.221]
NS    : ns2.secure64.com [216.17.193.194]
...
==> SUCCESS (but 4 warning(s))
```

D'autres zones ne sont pas aussi correctes, même sans DNSSEC. Voici un exemple des messages d'erreur :

```
% zonecheck zataz.com
ZONE  : zataz.com
NS <= : ns1.eurodns.com [80.92.65.2]
NS    : ns2.eurodns.com [80.92.67.140]

       ______________________
     ,----------------------.|
~~~~ |    avertissement    || ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
     `----------------------'
a> Les serveurs de nom font tous partie du même AS
 | Conseil: ZoneCheck
 |   Afin d'éviter de perdre la connectivité avec les serveurs DNS
 | autoritaires en cas de problèmes de routage dans le Système Autonome,
 | il est conseillé d'héberger les serveurs sur différents AS.
 `----- -- -- - -  -
 :   Tous les serveurs de noms font partie du même Système Autonome (AS
 : numéro 24611), essayez d'en héberger certains sur un autre.
 `..... .. .. . .  .
=> générique

       ______________
     ,--------------.|
~~~~ |    fatal    || ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
     `--------------'
f> L'adresse IP de l'enregistrement MX ne peut pas être résolue
 :   Le MX mailing.zataz.com représentant le nom du serveur ne possède
 : pas d'adresse IP.
 `..... .. .. . .  .
=> ns1.eurodns.com/80.92.65.2

==> ÉCHEC (et 1 avertissement(s))
```

Et, en effet, le serveur de courrier désigné n'existe pas :

```
% dig @ns1.eurodns.com MX zataz.com
...
;; ANSWER SECTION:
zataz.com.              3600    IN      MX      10 mail.zataz.com.
zataz.com.              3600    IN      MX      20 mailing.zataz.com.
...

% dig @ns1.eurodns.com A mailing.zataz.com
...
;; ->>HEADER<<- opcode: QUERY, status: NXDOMAIN, id: 49758
```

À noter que Zonecheck sépare la **politique** (quels tests on fait et un résultat positif est-il obligatoire) du moteur de tests qui les applique. L'administrateur système peut éditer le profil des tests et retirer (ou rendre facultatif) certains tests, à volonté. Par exemple, par défaut, Zonecheck teste que les adresses IP des serveurs de noms ne sont pas des "*bogons*" mais, s'ils le sont, cela ne produit qu'un simple avertissement :

```
<check name="ip_bogon"   severity="w" category="ip"/>
```

Si on trouve ce test trop laxiste, il suffit de changer `severity` de 'w' ("*warning*") à 'f' ("*fatal*"). Si, à l'inverse, on trouve ce test inutile ou tout simplement trop coûteux en ressources, on peut le désactiver complètement en retirant l'élément `<check>` du profil. C'est une caractéristique unique parmi les logiciels de tests DNS.

Zonecheck est un logiciel libre sous licence GPL. Merci à Fabien Chabanne pour avoir pris en charge cette nouvelle version, et à Stéphane d'Alu pour le programme original.