Quel est le vrai facteur d'amplification lors d'une attaque utilisant le DNS ?

Stéphane Bortzmeyer
<stephane+blog@bortzmeyer.org>

Première rédaction de cet article le 29 mars 2013

http://www.bortzmeyer.org/amplification-dns-combien.html

À l'occasion de l'attaque par déni de service contre Spamhaus/Cloudflare <http://pro.clubic.com/it-business/securite-et-donnees/actualite-550362-spamhaus-ddos-cyberbunker.html>, plusieurs articles ont tenté de chiffrer l'amplification DNS qui servait de base à l'attaque et, comme les chiffres ronds passent mieux, ont souvent parlé d'un « facteur 100 ». Est-ce exact ? En fait, c'est plus compliqué.

Rappelons le principe de l'attaque : le méchant envoie une requête DNS en usurpant l'adresse IP source de la victime (ce qui est souvent possible, trop peu de FAI ayant déployé les RFC 2827 [1] et RFC 3704). À qui l'envoie t-il ? Il y a deux variantes de l'attaque, une utilisant les serveurs faisant autorité et l'autre utilisant des résolveurs ouverts (laisser de tels résolveurs est très mal <http://www.bortzmeyer.org/fermer-les-recursifs-ouverts.html>). L'attaque Spamhaus/Cloudflare utilisait apparemment les seconds (plus nombreux et moins bien gérés). Ces résolveurs reçoivent la requête et répondent à ce qu'ils croient être le demandeur mais qui est en fait la victime. C'est donc une partie à trois, l'Attaquant, le Réflecteur et la Victime. Mais je ne vais pas vous faire un court complet sur les attaques DNS par réflexion + amplification, je voulais juste calculer le facteur d'amplification. Et, ce qui est pratique, est qu'il est le même dans les deux variantes : la quasi-totalité des serveurs de noms, récursifs ou faisant autorité, a une limite de taille de la réponse à 4 096 octets. C'est la valeur mentionnée par le RFC 6891 et c'est la valeur par défaut de BIND (modifiable avec le paramètre `max-udp-size`), de NSD (modifiable avec les paramètres `ipv4-edns-size` et `ipv6-edns-size`:). et d'Unbound (qui, par contre, ne permet pas de changer cette valeur). Certains changent cette configuration (à ma connaissance, les seuls serveurs de TLD qui n'envoient jamais de réponse aussi grande sont ceux de `.com`, apparemment limités à 1 460 octets).

Donc, quelle que soient les techniques utilisées (et, notamment, qu'on se serve de DNSSEC ou pas), on aura au maximum une réponse de 4 096 octets. Créons, comme si on était un attaquant, un enregistrement DNS (type TXT car sa partie Valeur n'est pas de taille fixe) de 3 927 octets (ça nous fera une réponse de 4 072 octets, quasiment le maximum) et interrogeons un résolveur (si vous voulez créer de tels TXT, j'ai utilisé ce petit script (en ligne sur `http://www.bortzmeyer.org/files/create-dns-txt.py`)). tcpdump nous montre :

1. Pour voir le RFC de numéro NNN, `https://www.ietf.org/rfc/rfcNNN.txt`, par exemple `https://www.ietf.org/rfc/rfc2827.txt`

```
12:19:41.556276 00:1a:a0:ac:4a:0c > 00:10:db:ff:40:70, ethertype IPv4 (0x0800), length 94: (tos 0x0, ttl 64,
    192.134.6.113.49032 > 149.20.64.21.53: [bad udp cksum 0x9c6e -> 0x2fd9!] 13395+ [1au] TXT? depardieu.exa

12:19:42.352888 00:10:db:ff:40:70 > 00:1a:a0:ac:4a:0c, ethertype IPv4 (0x0800), length 1154: (tos 0x0, ttl 5
    149.20.64.21.53 > 192.134.6.113.49032: 13395 q: TXT? depardieu.example. 1/2/1 depardieu.example. [23h59m
12:19:42.352960 00:10:db:ff:40:70 > 00:1a:a0:ac:4a:0c, ethertype IPv4 (0x0800), length 1514: (tos 0x0, ttl 5
    149.20.64.21 > 192.134.6.113: ip-proto-17
12:19:42.353000 00:10:db:ff:40:70 > 00:1a:a0:ac:4a:0c, ethertype IPv4 (0x0800), length 1514: (tos 0x0, ttl 5
    149.20.64.21 > 192.134.6.113: ip-proto-17
```

Trop grosse pour la MTU d'Ethernet, la réponse a été fragmentée en trois datagrammes. Maintenant, calculons. Prenons les tailles en couche 2 (le premier "*length*", le second, après l'étiquette `UDP`, étant en couche 3, mais cela ne change pas grand'chose au facteur d'amplification). Question : 94 octets. Réponse : 1154+1514+1514 = 4 182 octets. Le facteur d'amplification, dans ce cas idéal, est de 44...

On voit qu'on est loin du facteur 100 vu dans certains articles. En grattant un peu, on peut récupérer quelques octets (essayer de trouver des noms plus courts, par exemple) mais cela ne fera pas de miracle. Donc, pas de facteur 100 en vue, même si le facteur effectif, aux alentours de 45, est déjà largement suffisant pour permettre des attaques violentes.

Une autre solution, bien sûr, est de tricher. Par exemple en comparant les tailles, non pas en octets sur le réseau mais en charge utile uniquement (couche 7). Dans cas, la requête est bien plus petite et on a des amplifications purement théoriques mais bien plus jolies (c'est que ce que fait Bernstein dans le script awk qui accompagne un de ses exposés anti-DNSSEC <`http://cr.yp.to/talks/2012.06.04/slides.pdf`> et il trouve en effet presque un facteur 100).

À noter que le facteur d'amplification discuté ici est le **BAF** ("*Bandwidth Amplification Factor*"). Dans certaines attaques, il faut plutôt considérer le **PAF** ("*Packet Amplification Factor*"). Ce sont des attaques où l'ennemi cherche à obtenir le plus de paquets possibles (et pas le plus d'octets possibles), car certaines ressources Internet sont plutôt limitées en nombre de paquets par seconde. (Merci à Christian Rossow pour la jolie terminologie BAF et PAF.)