Attaques contre le DNS et limitation de trafic

Stéphane Bortzmeyer
<stephane+blog@bortzmeyer.org>

Première rédaction de cet article le 11 juin 2012

https://www.bortzmeyer.org/dns-rate-limiting-and-attacks.html

---

L'actualité fait en ce moment la part belle aux attaques par déni de service utilisant le DNS. On observe des rapports <https://isc.sans.edu/diary/DNS+ANY+Request+Cannon+-+Need+More+Packets/13261>, de longues discussions sur des listes comme NANOG ou dns-operations <https://lists.dns-oarc.net/mailman/listinfo/dns-operations>, mais aussi via des canaux plus discrets. Une technique est souvent discutée : la limitation de trafic, technique qui était pourtant généralement ignorée autrefois.

D'abord, un point **très** important : il n'y a pas **un** type d'attaque avec des caractéristiques bien définies. Il y a des tas de sortes d'attaques en ce moment, visant des cibles très différentes, et avec des outils bien distincts. Il ne faut donc pas essayer de trouver **le** coupable, ni de mettre au point une technique qui fera face efficacement à **toutes** les attaques.

Quelles sont les attaques par déni de service possibles via le DNS ? (Pour des raisons évidentes, je ne parle que de celles qui sont publiques et connues.) Première différenciation entre attaques, la victime :

— Le méchant peut en vouloir à un serveur DNS donné, serveur récursif ou bien faisant autorité. Il lui envoie alors des requêtes DNS en masse. Si la réponse est plus grosse que la question (ce qui est presque toujours le cas avec le DNS), le méchant obtient en même temps une **amplification**. Le serveur souffrira des requêtes... mais surtout des réponses qu'il génère lui-même. (Notez que des attaques par amplification ont été vues avec d'autres protocoles UDP comme SNMP, NTP - cf. mon article <https://www.bortzmeyer.org/ntp-reflexion.html> - et même Call of Duty <http://cert.lexsi.com/weblog/index.php/2011/10/18/422-new-dos-attack-amplified-through-gaming-servers>.)
— Le méchant attaquant peut en vouloir à un tiers (qui a ou n'a pas de serveur DNS), il va alors tricher sur son adresse IP source (ce qui est trivial pour le protocole UDP, le plus commun pour le DNS) et envoyer les requêtes à un serveur (récursif ou faisant autorité). Le serveur va alors servir de **relais** et répondre à la victime (en prime, si le gérant du serveur s'en aperçoit, il va accuser la victime de l'attaquer).

Ou bien on peut classer selon l'attaquant :

— S'il a à sa disposition un botnet, il peut dire la vérité sur l'adresse source (ou pas, selon ses goûts et ses buts), et il a à sa disposition de nombreuses machines donc n'a pas forcément besoin d'une grosse amplification pour faire des dégâts.

— S'il n'a pas de botnet à utiliser, il va devoir tricher sur l'adresse IP source (utiliser la sienne le trahirait) et il ne peut pas bombarder intensivement avec ses seules machines, il va avoir besoin d'amplification.

Pour une attaque par réflexion, l'attaquant peut viser un serveur faisant autorité (ceux-ci sont souvent de grosses machines bien connectées et ils peuvent être très efficaces comme relais) ; il aura intérêt à choisir des serveurs faisant autorité pour une zone qui contient de nombreux enregistrements de grande taille. (Si vous cherchez, c'est le champ `MSG SIZE` à la fin de la sortie de dig.) Ou bien il peut viser un serveur récursif. Ceux-ci n'acceptent normalement que les requêtes provenant de leur réseau (RFC 5358 [1]) mais ils sont souvent mal configurés et ouverts à tout l'Internet, ce qui les rend très dangereux <https://www.bortzmeyer.org/fermer-les-recursifs-ouverts.html> d'autant plus qu'ils sont très nombreux. (Par exemple, bien des CPE fournis par les FAI sont des résolveurs ouverts.) L'attaquant peut alors utiliser des tas de noms de domaine différents (rendant plus difficile les contre-mesures), ou bien utiliser un domaine à lui, avec d'énormes enregistrements qu'il choisit.

Normalement, une machine ne devrait pas pouvoir tricher sur son adresse IP (RFC 2827 et RFC 3704). Mais, en pratique, cette astuce reste largement possible.

Parmi les contre-mesures figure la limitation de trafic. J'ai déjà écrit un article sur son utilisation pour empêcher les abus sur un résolveur ouvert <https://www.bortzmeyer.org/rate-limiting-dns-open-reso
html> et un autre sur les limites de Linux/Netfilter pour limiter intelligement <https://www.bortzmeyer.
org/dns-netfilter-u32.html>. Cette technique a aussi été décrite par Google <https://developers.
google.com/speed/public-dns/docs/security>. Ceux qui utilisent FreeBSD seront contents de découvrir que le serveur racine F utilise FreeBSD et est protégé par :

```
add     pipe 1          udp     from any to any 53 in
pipe 1  config  mask src-ip 0xffffffff buckets 1024 bw 400Kbit/s queue 3
add     pipe 2          tcp     from any to any 53 in
pipe 2  config  mask src-ip 0xffffffff buckets 1024 bw 100Kbit/s queue 3
```

Enfin, pour limiter l'amplification, on peut aussi empêcher le serveur de noms d'envoyer des paquets trop gros :

```
// BIND
max-udp-size 1460

# nsd
ipv4-edns-size: 1460
ipv6-edns-size: 1460
```

Si la réponse dépasse cette taille, le serveur mettra le bit TC ("*TrunCation*") à 1, poussant le client à réessayer en TCP (chose qu'il ne pourra pas faire s'il ment sur son adresse IP).

Résultat des attaques actuelles et des limites des outils existants, ce ne sont pas moins de deux efforts indépendants qui sont en cours pour ajouter des fonctions de limitation de trafic à BIND, ce qui permettra peut-être un meilleur contrôle de ce service. Pour comprendre la différence entre les deux efforts, il faut se rappeler :

1. Pour voir le RFC de numéro NNN, `https://www.ietf.org/rfc/rfcNNN.txt`, par exemple `https://www.ietf.org/rfc/rfc5358.txt`

— Que le but de l'administrateur réseaux n'est pas juste d'arrêter l'attaque (il suffirait d'éteindre la machine), mais aussi de le faire sans perturber le service qu'il doit rendre. Pas question donc de jouer au cow-boy en bloquant aveuglément des innocents.
— Que les contre-mesures à une DoS ont souvent, lorsqu'elles sont mal conçues, la propriété d'aggraver l'attaque, par exemple en imposant au serveur une consommation de mémoire qui va l'handicaper.

La première approche vise avant tout à ne pas avoir de « faux positifs ». Elle est décrite dans la Tech Note TN-2012-1 <http://ss.vix.com/~vixie/isc-tn-2012-1.txt> de l'ISC sous le nom de "*Response Rate Limiting*" (DNS RRL). Et elle est mise en œuvre dans un patch de BIND récemment publié <http://www.redbarn.org/dns/ratelimits>. La configuration ressemble à :

```
config {
    // ...
        rate-limit {
                responses-per-second 5;
                window 5;
        };
}; // Not yet per-type or per-domain such as "rate-limit only ANY queries"
```

La deuxième approche vise à limiter la consommation mémoire sur le serveur. La plupart des mécanismes de limitation de trafic (comme ceux indiqués dans mon article <https://www.bortzmeyer.org/rate-limiting-dns-open-resolver.html> ou comme ceux utilisés dans le patch ci-dessus) consomment de la mémoire proportionellement au nombre de préfixes IP attaquants (et, en IPv6, ça peut aller vite). D'où l'idée <http://fanf.livejournal.com/122111.html> d'utiliser des filtres de Bloom pour avoir un coût qui ne dépend pas du nombre de préfixes, au prix d'un certain nombre de faux positifs (adresses IP qui seront limitées alors qu'elles n'auraient pas dû l'être ; les filtres de Bloom sont probabilistes). Cette approche est en cours de programmation <https://github.com/fanf2/bind-9/blob/master/doc/misc/ratelimiting>.

Pour l'instant, aucune de ces deux solutions ne permet une limitation dépendant du type de requêtes, ou du nom demandé. De même, il n'y a pas encore de limitation liée au comportement du client (par exemple, limiter ceux qui re-posent la même question avant l'expiration du TTL). Un tel test du comportement ne marcherait de toute façon que pour les serveurs faisant autorité, qui ont normalement en face d'eux des résolveurs/cache. Pour les résolveurs, leurs clients n'ayant souvent pas de cache, un tel test ne serait pas possible.

Voilà, il est encore trop tôt pour dire quelle solution s'imposera. En attendant, prudence : méfiez-vous des recettes toutes faites et rappelez-vous qu'il existe plusieurs sortes d'attaque. Que diriez-vous d'un médecin qui ne connaitrait qu'un seul médicament et le prescrirait dans tous les cas ? La limitation de trafic est un médicament puissant et utile, mais qui ne doit être utilisé qu'après une analyse de l'attaque.

---

https://www.bortzmeyer.org/dns-rate-limiting-and-attacks.html