Un "shunt" BGP observé en vrai

Stéphane Bortzmeyer
<stephane+blog@bortzmeyer.org>

Première rédaction de cet article le 21 novembre 2013. Dernière mise à jour le 26 novembre 2013

https://www.bortzmeyer.org/bgp-shunt.html

On sait depuis longtemps qu'il est trivial d'annoncer sur l'Internet des routes pour d'autres adresses IP que les siennes. On peut ainsi capter le trafic de sa victime pour la couper du réseau (attaque par déni de service) ou peut-être pour se faire passer pour sa victime et, par exemple, recevoir du courrier qui ne vous est normalement pas destiné. Mais cette attaque est vite détectée car la victime ne reçoit plus (ou plus beaucoup) de trafic. D'où l'idée, très ancienne, de réinjecter le trafic à sa victime, après espionnage ou modification, pour retarder cette détection. Cela se nomme un "*shunt*" BGP, en référence à un dispositif électrique. Une étude récente de Renesys <http://www.renesys.com/2013/11/mitm-internet-hijacking/> semble être la première à avoir mis en évidence cette attaque dans le monde réel.

L'attaque observée par Renesys <http://www.renesys.com/> comprend deux parties : une annonce BGP usurpée (comme dans la classique attaque de Pakistan Telecom contre YouTube <https://www.bortzmeyer.org/pakistan-pirate-youtube.html>) **et** un mécanisme (apparemment pas décrit dans l'article de Renesys) pour s'assurer de l'existence d'un chemin de retour, un "*clean path*" qui ne voit pas l'annonce BGP usurpée, de manière à ce qu'il transmette le trafic à la victime. La première partie, l'annonce BGP usurpée, est quelque chose de relativement fréquent dans l'Internet, la deuxième a été vérifiée par Renesys en envoyant des paquets vers un réseau détourné et en testant qu'ils arrivaient bien à destination, juste via une route très longue et très anormale (voir les traceroute dans l'article de Renesys <http://www.renesys.com/2013/11/mitm-internet-hijacking/>). L'un des détournements était fait vers la Biélorussie, l'autre vers l'Islande (attention, le vrai responsable n'est pas forcément dans ces deux pays, il peut être caché derrière un opérateur piraté).

Si la possibilité d'un "*shunt*" BGP était connue depuis longtemps, les détails pratiques (il n'est pas évident de détourner un préfixe IP depuis tout l'Internet, tout en maintenant le "*clean path*" pour le retour) n'ont été décrits qu'en 2008 dans un article fameux de Kapela et Pilosov <http://www.bortzmeyer.org/faille-bgp-2008.html>. Pour maintenir le chemin de retour, Kapela et Pilosov utilisaient l'"*AS prepending*", l'ajout des numéros d'AS des opérateurs du chemin de retour à l'annonce usurpée, afin que ces opérateurs n'acceptent pas cette annonce (cela n'a pas été fait ici, voir l'annonce plus loin). À noter que Kapela et Pilosov proposaient également des méthodes pour rendre la détection plus difficile, comme de modifier le TTL dans les paquets IP pour tromper traceroute (cette astuce ne semble pas

avoir été utilisée ici). Le travail de Kapela et Pilosov était théorique, il semble bien que les deux attaques étudiées par Renesys marquent le passage de leur méthode dans le monde réel.

Et les solutions ? À court terme, il est important de se rappeler qu'il **faut** chiffrer son trafic. Même si on pense être en sécurité car la communication est à courte distance (« mon trafic va uniquement de Denver à Denver, il n'y a pas de méchants à Denver »), l'étude de Renesys montre bien qu'un trafic local peut devenir distant grâce à l'attaque BGP et le faire passer par des endroits non sûrs. Quels que soient ses inconvénients <https://www.bortzmeyer.org/crypto-debug.html>, la cryptographie est une technologie à utiliser <https://www.bortzmeyer.org/crypto-protection.html>.

À un peu plus long terme, il faut mettre en place des systèmes de détection. Utiliser traceroute ne va pas forcément marcher (l'attaquant peut vous tromper en bricolant les TTL). Votre opérateur ou vous-même ont donc tout intérêt à utiliser des systèmes d'alarme BGP <https://www.bortzmeyer.org/alarmes-as.html>. L'attaquant peut faire bien des choses mais, par construction, il ne peut pas empêcher ses manipulations de se voir dans la table de routage globale. Bien sûr, ces systèmes n'empêcheront pas l'attaque mais, au moins, vous serez prévenus.

L'attaquant ne peut pas non plus violer les lois de la physique : en une milli-seconde, la lumière ne peut parcourir plus de 300 km. Un trajet de Denver à Denver qui prend moins d'une milli-seconde n'a certainement pas été détourné par l'Islande. Il faut donc mesurer le RTT et sonner l'alarme s'il augmente brusquement. Pour ceux qui utilisent les scripts de test compatibles Nagios, c'est le `rta`, le premier chiffre dans les seuils d'alerte de check_ping <https://www.monitoring-plugins.org/doc/man/check_ping.html> (le second étant le taux de pertes). Dans le futur, les sondes RIPE Atlas <https://atlas.ripe.net/> disposeront d'un mécanisme de test équivalent.

À plus long terme, la solution sera peut-être le déploiement massif de la RPKI <https://www.bortzmeyer.org/securite-routage-bgp-rpki-roa.html> mais on en est très loin aujourd'hui.

Plusieurs articles dans les médias ont été consacrés à cette attaque mais la plupart du temps, sans valeur ajoutée par rapport à l'article de Renesys, à part l'addition d'erreurs plus ou moins drôles. L'article d'Arik Hesseldahl <http://allthingsd.com/20131120/how-somebody-forced-the-worlds-internet-> n'est pas sans reproche (une erreur sur la notion d'attaque de l'homme du milieu) mais au moins il a fait un effort de pédagogie pour expliquer l'attaque à un public plus large que celui des lecteurs du blog de Renesys. J'en profite pour rappeler un très bon article sur la façon de faire une attaque en détournant le trafic, et comment le détourner, l'article d'Andree Toonk <http://www.bgpmon.net/accidentally-stealing-the-internet/>.

Et enfin, cherchons la vraie annonce BGP, dont Renesys ne donne qu'un résumé. (Attention, cela sera un peu plus technique.) On va se servir des archives de RouteViews <http://www.routeviews.org/>, librement accessibles en ligne et qui remontent à 1997. Renesys donne l'heure exacte d'une des attaques, 07 :36 :36 UTC le 31 juillet. Les URL des archives de RouteViews sont prévisibles donc on sait que l'annonce qui nous intéresse va être dans le fichier `bgpdata/2013.07/UPDATES/updates.20130731.0730.b` (`updates.ANNÉE MOIS JOUR . HEURE MINUTE`). On regarde les archives récoltées au LINX, plus proche de l'Islande (en vrai, j'avais d'abord regardé les archives récoltées par l'ISC en Californie, qui contenaient moins de choses). Donc :

```
% wget ftp://archive.routeviews.org/route-views.linx/bgpdata/2013.07/UPDATES/updates.20130731.0730.bz2

%  bunzip2 updates.20130731.0730.bz2
```

Le fichier ainsi obtenu est binaire, au format MRT (RFC 6396 [1]). On le transforme en texte avec bgpdump <https://bitbucket.org/ripencc/bgpdump/> :

```
% bgpdump updates.20130731.0730 > updates.20130731.0730.txt
```

Et on examine tranquillement le fichier texte. Connaissant l'heure de l'attaque et l'AS d'origine de l'annonce usurpée, on finit par trouver une des annonces mensongères :

```
TIME: 07/31/13 07:36:46
TYPE: BGP4MP/MESSAGE/Update
FROM: 195.66.236.35 AS6067
TO: 195.66.237.222 AS6447
ORIGIN: IGP
ASPATH: 6067 6677 48685
NEXT_HOP: 195.66.236.35
ANNOUNCE
  64.81.96.0/24
  64.81.97.0/24
...
```

Que voit-on dans cette annonce ? L'heure correspond à ce qu'indique Renesys (les collecteurs de RouteViews n'ont pas forcément des horloges exactes à la seconde près et, de toute façon, la propagation BGP n'est pas instantanée). Le message a un chemin d'AS qui commence en 48685 et continue en 6677, comme le notait Renesys, avant d'arriver à l'AS 6067, un opérateur anglais, fournisseur de RouteViews. C'est donc là qu'on voit que l'attaquant n'utilisait pas d'"*AS prepending*" pour se créer un chemin de retour. Enfin, on a les préfixes annoncés, appartenant à Megapath, un opérateur états-unien, qui n'a certainement pas de fournisseur en Islande. L'annonce est donc clairement anormale, même si, juridiquement parlant, on peut estimer qu'on n'a pas de preuve qu'elle soit malveillante. (Jolie analyse, mais j'ai triché, j'ai été guidé par un informateur qui veut rester anonyme.)

1. Pour voir le RFC de numéro NNN, https://www.ietf.org/rfc/rfcNNN.txt, par exemple https://www.ietf.org/rfc/rfc6396.txt