Outils pour obtenir des informations BGP publiques

Stéphane Bortzmeyer
<stephane+blog@bortzmeyer.org>

Première rédaction de cet article le 4 septembre 2020. Dernière mise à jour le 19 mai 2022

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

Cette page (que j'espère maintenir à jour) rassemble les outils existants pour obtenir de l'information sur les annonces BGP, même si on n'a pas d'accès à des routeurs BGP.

Le protocole de routage BGP est sans doute le système le plus crucial pour le bon fonctionnement de l'Internet. Contrairement à des protocoles applicatifs comme HTTP, il n'est pas prévu que tout le monde puisse parler BGP : seule une partie des routeurs le fait et, sauf si vous travaillez chez un acteur important de l'Internet, vous n'avez probablement pas accès à un routeur BGP, encore moins un routeur de la DFZ. D'où l'intérêt de divers outils et services qui permettent d'obtenir des informations BGP sans avoir cet accès privilégié.

D'ailleurs, même si vous avez accès à un ou plusieurs routeurs BGP, cela n'est pas forcément suffisant. Vu la façon dont fonctionne BGP, tous les routeurs ne voient pas la même chose (même les routeurs de la DFZ) et de tels outils sont donc utiles même pour les professionnels du réseau. Attention, certains de ces outils sont simples à utiliser, d'autres plus complexes mais dans tous les cas, comprendre ce qu'ils affichent nécessitent des compétences dans le fonctionnement de l'Internet, et dans le protocole BGP.

Cet article regroupe les outils que **j**'utilise. Vous pouvez m'en suggérer d'autres (ou bien corriger des erreurs) mais cette liste est forcément incomplète et subjective. Alors, commençons tout de suite par le principal outil dont je me sers, RIPEstat <https://stat.ripe.net/>. RIPEstat est une interface Web notamment vers les données récoltées par le RIS <https://ris.ripe.net/> ("*Routing Information Service*"), un ensemble de centaines de machines parlant BGP et qui s'appairent avec tout le monde pour récolter le plus d'informations BGP possibles. En échange d'une adresse IP, d'un préfixe ou d'un AS, vous pouvez obtenir plein d'informations. On va se concentrer sur celles liées au routage. Prenons par exemple le préfixe 2a01:e30::/28, utilisé pour les clients de Free. (Si vous ne connaissez pas le préfixe, entrez l'adresse IP, RIPEstat trouvera le préfixe englobant le plus spécifique.) Voici ce qu'affiche l'onglet « Routage » de RIPEstat, en :

(Une minorité de routeurs du RIS voit ce préfixe ; il n'est sans doute pas annoncé à tout le monde, et il existe un /26 plus générique. Rappelez-vous ce que j'ai dit que tout les routeurs BGP ne voient

pas la même chose.) Ce "*Routing status*" n'est qu'un des "*widgets*" disponible, plus bas dans la page vous trouverez de nombreuses autres informations. J'aime bien le "*widget*" historique qui permet de voir comment a été annoncé ce préfixe dans le passé :

Et aussi le rythme des mises à jour, souvent indicatifs d'un problème. Ici, par exemple, le "*widget*" "*BGP update activity*" montre la panne de Level 3/CenturyLink <https://seenthis.net/messages/873934> du 30 août 2020. On a donné comme ressource à voir l'AS 3356 (celui de Level 3) et zoomé pour n'avoir que la partie intéressante. On voit alors le gros surcroit d'activité BGP engendré par le problème chez Level 3. C'est toujours visitable aujourd'hui, grâce aux URL intégrant la date : .

Du fait qu'il existe un URL stable pour les informations de RIPEstat, on peut facilement embarquer du RIPEstat <https://www.bortzmeyer.org/stat-ripe.html> dans ses pages Web, ses outils de supervision, etc.

RIPEstat est très gourmand en ressources, vu son utilisation massive de plein de JavaScript. Vous avez intérêt à avoir une machine riche en RAM et, même ainsi, vous verrez souvent l'avertissement (ici de Firefox) comme quoi un script ralentit la machine :

Le RIS, le réseau derrière RIPE stat peut aussi être interrogé en ligne de commande. C'est ce RIS qui alimente mon service bgp.bortzmeyer.org :

```
% curl -s https://bgp.bortzmeyer.org/2a03:2880:f0fc:c:face:b00c:0:35
2a03:2880:f0fc::/48 32934
```

Cette petite fonction shell peut vous faciliter la vie :

```
bgprouteris () {
 if [ -z "$1" ]
 then
  echo "Usage: bgprouteris IP-address"
  return 1
 fi
 curl -s https://bgp.bortzmeyer.org/$1
 echo ""
}
```

Il est de toute façon bon de ne pas dépendre d'un seul service, même géré par une organisation sans but lucratif et fondée sur un projet commun comme l'est le RIPE-NCC. Tout service peut disparaitre ou tomber en panne précisement au moment où on en a besoin (si on veut investiguer un problème en cours, par exemple). Une alternative intéressante est bgp.tools <https://bgp.tools/>. C'est plus léger que RIPEstat (mais moins riche) et cela se concentre sur des informations essentielles, donc cela peut être pratique pour des utilisateurs moins familiers de BGP. (Je ne trouve pas sur quelles données ils s'appuient pour afficher leurs informations : rappelez-vous que les informations BGP ne sont pas les mêmes partout, d'où l'importance d'avoir un grand nombre de routeurs situés un peu partout, comme le RIS. Je ne connais pas la représentativité des collecteurs d'informations de bgp.tools.)

Voici par exemple ce que voit bgp.tools sur le préfixe 2a01:e30::/28 cité plus haut (URL) :

Et sur l'AS associé :

Vous avez noté que dans les informations sur le préfixe, la rubrique "*Upstreams*" (transitaires) était vide. bgp.tools ne l'affiche pas lorsqu'il y a un préfixe plus général et visible plus globalement, ce qui est le cas ici (rappelez-vous que le `2a01:e30::/28` n'est pas annoncé partout). Avec le préfixe général, on a bien l'information :

En prime, bgp.tools nous prévient que Free n'a qu'un seul transitaire en IPv6, Cogent et que celui-ci refuse de s'appairer avec Hurricane Electric, ce qui prive les abonnés Free d'une partie de l'Internet.

Dans la série « sites Web pour récupérer des informations BGP », beaucoup de gens utilisent qui donne, par exemple :

Pour les amateurs de ligne de commande, il y a aussi `bgpstuff.net` :

```
% curl -s https://bgpstuff.net/route/185.89.219.12
Route is 185.89.219.0/24 for 185.89.219.12
```

Et si on veut le numéro d'AS, pas juste les préfixes :

```
% curl https://bgpstuff.net/origin/185.89.219.12
The origin AS for 185.89.219.12 is AS32934
```

Jusqu'ici, je n'ai listé que des outils Web (ou en tout cas HTTP). Et si on n'aime pas le Web ? Les mêmes informations sont souvent disponibles par d'autres protocoles, par exemple whois. (RIPEstat a également une API, que je n'utilise personnellement pas.) Le RIS est ainsi interrogeable par whois <`http://www.ripe.net/ris/riswhois.html`> :

```
% whois -h riswhois.ripe.net 2a01:e30::/28
...
route6:       2a01:e30::/28
origin:       AS12322
descr:        PROXAD Free SAS, FR
lastupd-frst: 2020-04-13 01:00Z  2001:7f8:20:101::208:223@rrc13
lastupd-last: 2020-09-02 07:37Z  2001:7f8:20:101::209:93@rrc13
seen-at:      rrc00,rrc01,rrc03,rrc04,rrc05,rrc07,rrc10,rrc11,rrc12,rrc13,rrc15,rrc20,rrc21,rrc23
num-rispeers: 156
source:       RISWHOIS
```

Il y a évidemment moins d'information que par le Web mais cela peut suffire. Si on veut juste une correspondance entre une adresse IP et l'AS qui l'annonce, Team Cymru <`https://team-cymru.com/`> propose plusieurs outils <`https://team-cymru.com/community-services/ip-asn-mapping/`> comme whois :

```
% whois -h whois.cymru.com 80.67.169.12
AS      | IP               | AS Name
20766   | 80.67.169.12     | GITOYEN-MAIN-AS The main Autonomous System of Gitoyen (Paris, France)., FR
```

Autre serveur whois, chez `bgp.tools` :

```
% whois -h bgp.tools 2a00:e00:0:5::2
AS      | IP                                       | BGP Prefix          | CC | Registry | Allocated  | AS N
8304    | 2a00:e00:0:5::2                          | 2a00:e00::/32       | FR | RIPE     | 2008-11-07 | Ecri
```

Team Cymru a aussi une passerelle DNS. Celle-ci nécessite d'inverser les différents composants de l'adresse IP. Par exemple, pour `204.62.14.153`, il faudra interroger `153.14.62.204.origin.asn.cymru.com`. Ça peut s'automatiser avec awk :

```
% dig +short  TXT $(echo 204.62.14.153 | awk -F. '{print $4 "." $3 "." $2 "." $1 ".origin.asn.cymru.com" }')
"46636 | 204.62.12.0/22 | US | arin | 2008-12-24"
```

Pour IPv6, cette inversion peut se faire avec le programme ipv6calc <`http://www.deepspace6.net/projects/ipv6calc.html`>. On peut créer une fonction shell pour se faciliter la vie :

```
% which bgproutednscymru
bgproutednscymru () {
 address=$1
 if echo $address | fgrep -q : -; then
     domain=$(echo $1 | ipv6calc --addr2ip6_arpa | sed 's/ip6\.arpa\.$/origin6.asn.cymru.com./')
 else
     domain=$(echo $1 | awk -F. '{print $4 "." $3 "." $2 "." $1 ".origin.asn.cymru.com" }')
 fi
 dig +short TXT $domain
}

% bgproutednscymru 80.67.169.12
"20766 | 80.67.160.0/19 | FR | ripencc | 2001-05-21"

% bgproutednscymru 2001:910:800::12
"20766 | 2001:910::/32 | FR | ripencc | 2002-09-24"
```

Le service RouteViews <`http://www.routeviews.org`> a également une passerelle DNS, mais uniquement pour IPv4, avec le domaine `aspath.routeviews.org`. Elle indique le chemin d'AS (vers le collecteur de RouteViews), pas uniquement l'origine. Avec une fonction analogue à celle ci-dessus, on obtient :

```
% which bgproutednsrouteviews
...
 dig +short TXT `echo $1 | awk -F. '{print $4 "." $3 "." $2 "." $1 ".aspath.routeviews.org" }'` | \
              awk -F\" '{print "AS path: " $2 "\nRoute: " $4 "/" $6}'
}

% bgproutednsrouteviews 80.67.169.12
AS path: 53767 3257 1299 20766
Route: 80.67.160.0/19
```

Un exemple de son utilisation figure dans mon article sur un opérateur nord-coréen <https://www.bortzmeyer.org/star-jv-transtelecom.html>.

Plus original, il existe un bot sur le fédivers (documenté ici <https://www.bortzmeyer.org/fediverse-bot-bgp.html>) pour récupérer l'AS d'origine d'une adresse IP : .

J'ai parlé d'API à propos de RIPEstat. Personnellement, j'utilise l'API de QRator <https://radar.qrator.net/>. Il faut s'enregistrer sur le site (la plupart des services présentés ici ne nécessitent pas d'enregistrement) pour obtenir une clé d'API puis lire la documentation <https://api.radar.qrator.net/> (l'API produit évidemment du JSON). J'ai fait une fonction shell pour me faciliter la vie :

```
bgpqrator () {
    if [ -z "$1" ]; then
 echo "Usage: bgpqrator IP-address"
 return 1
    fi
    curl -s -X GET "https://api.radar.qrator.net/v1/lookup/ip?query=$1" \
  -H  "accept: application/json" -H "QRADAR-API-KEY: $(cat ~/.qrator)" | \
 jq .
}
```

Et cela me permet de faire :

```
% bgpqrator 2a01:e30::/28
{
  "meta": {
    "status": "success",
    "code": 200
  },
  "data": [
    {
      "id": "12322",
      "name": "PROXAD",
      "short_descr": "Free SAS",
      "prefix": "2a01:e00::/26",
      "as_num": "12322",
      "found_ips": "{2a01:e30::/28}"
    },
    {
      "id": "12322",
      "name": "PROXAD",
      "short_descr": "Free SAS",
      "prefix": "2a01:e30::/28",
      "as_num": "12322",
      "found_ips": "{2a01:e30::/28}"
    }
  ]
}
```

Un point important de BGP aujourd'hui est la possibilité de signer les informations pour améliorer la sécurité, avec l'infrastructure nommée RPKI. Pour vérifier ces signatures, on peut installer son propre validateur <https://www.ripe.net/manage-ips-and-asns/resource-management/certification/tools-and-resources> comme Routinator <https://nlnetlabs.nl/projects/rpki/routinator/

> (après tout, toutes les données de la RPKI sont publiques) mais c'est un peu compliqué à faire et surtout à maintenir, donc il peut être plus intéressant d'utiliser des services en ligne. Par exemple, fait cette vérification et vous affiche le résultat (cf. la copie d'écran plus haut). De même, RIPEstat affiche la validité d'une annonce comparée aux IRR et aux ROA :

Autre excellent outil de vérification de la cohérence entre ce qui est annoncé et les bases de données (IRR) et la RPKI, IRRexplorer <https://irrexplorer.nlnog.net/> (c'est un logiciel libre, vous pouvez aussi l'installer chez vous).

Notez que je ne connais pas encore de moyen simple de récupérer les ROA sur un site Web. Les services ci-dessus indiquent juste le résultat de la validation, pas le ROA d'origine. La seule méthode pour l'instant semble être de récupérer tout le contenu de la RPKI connu d'un point de publication (pour le RIPE-NCC, c'est avec rsync en `rsync://rpki.ripe.net/repository`) puis de le lire avec des outils comme OpenSSL (pour un certificat, `openssl x509 -inform DER -text -in NOMDUFICHIER.cer`).

Jusqu'à présent, on a vu des techniques qui indiquaient une vue « globale », supposant qu'on avait à peu près le même résultat sur tous les routeurs BGP. En pratique, on sait que ce n'est pas vrai, les différents routeurs ne voient pas exactement la même chose, et il est souvent utile de regarder ce que voit un routeur particulier. C'est le rôle des "*Looking Glasses*". Il en existe beaucoup, mais pas toujours là où on voudrait. (Pour un problème récent <https://www.bortzmeyer.org/routage-divers.html>, je cherchais un "*looking glass*" chez Algérie Télécom, sans en trouver.) Bref, il faut utiliser les annuaires comme et ils ne sont évidemment jamais à jour, on a des mauvaises surprises. C'est un cas où il faut parfois compter sur les moteurs de recherche.

Aux joies du Web moderne avec tous ses gadgets et son interactivité graphique, et même aux outils plus techniques qu'on vient de voir, on peut souhaiter préférer l'analyse qu'on fait soi-même à partir de données brutes. On télécharge des fichiers rassemblant les données BGP (soit l'état de la RIB du routeur, soit les annonces BGP) et on les analyse avec le programme de son choix. Un format standard existe même pour ces fichiers, MRT, normalisé dans le RFC 6396[1]. Un exemple d'utilisation de ces fichiers figure dans mon article sur une panne à Saint-Pierre-et-Miquelon <https://www.bortzmeyer.org/panne-saint-pierre-miquelon.html>.

Où peut-on trouver de tels fichiers ? RouteViews <http://www.routeviews.org/> en fournit, une archive <http://archive.routeviews.org/> qui remonte à 2001…Chose amusante, la seule taille de ces fichiers peut indiquer un problème car les perturbations de l'Internet se traduisent en général par une augmentation importante des mises à jour BGP. Ainsi, la panne de Level 3/CenturyLink <https://seenthis.net/messages/873934> du 30 août 2020 se voit très bien (à partir de 10 :00 h UTC) :

On peut aussi avoir de telles données via le RIS, cf. la documentation <https://www.ripe.net/analyse/internet-measurements/routing-information-service-ris/ris-raw-data>. C'est sur ces fichiers issus du RIS que s'appuie le bot fédivers cité plus haut <https://www.bortzmeyer.org/fediverse-bot-bgp.html>.

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