PeeringDB Accuracy

Is blind faith reasonable?

Job Snijders

job.snijders@atrato.com

NANOG58
Who am I?

Job Snijders

Network Architect @ AS 5580 - Atrato IP Networks

Founder of NLNOG RING

Proud member of PeeringDB promo team

Twitter: @JobSnijders

Hobbies: IP Routing, LISP, MPLS, IPv6,

Shoe size: 45/EU
What is PeeringDB?
(first iteration of PDB in early sixties)
What is PeeringDB?

- Almost a third of the critical DFZ ASNs register (partly) their interconnection options in PeeringDB
- True game changer in Peering/IXP market
- Run by volunteers
- Data is 100% user submitted (think wikipedia)
Example peering record

PeeringDB Accuracy - Job Snijders
Current interfaces to PDB

Today
• MySQL (peeringdb.com:3306)
• Nightly SQL Dump (use this for automation!)
• Web interface

Future
• RESTful APIs
• Modern stuff
Problem statement

Is PeeringDB worthy of my trust?

Can I use the data in automated tooling for peering?
(one button peering)
Methodology

• 2 data sources are used as input for validating the dataset:
  – PeeringDB SQL dump
  – Actual ‘show bgp sum’ from many networks

• Throw all data into database, compare
Submit your **show bgp sum**

For my [NANOG58 talk](#) I will assess the accuracy of the current data in PeeringDB. The approach is to compare three data sources: PeeringDB itself, public data from IXP operators and the ultimate source of truth: the ultimate confirmation that an IP address belongs to an ASN at a certain IXP, is when multiple people submit data that shows they actually have a BGP session established with a certain IP and ASN. This tool will accept IOS, Brocade, JUNOS and CSV formatted data.

This tool will extract **ONLY** the following two pieces of information: (remote IP, remote ASN). And this information is only admitted to the data set when the session looks alive. **All other information such as your ASN, the amount of received prefixes and sent prefixes is of no relevance to the research and will be purged from the data set.**

In the below text field you can submit a copy/paste from 'show ip bgp sum', 'show ipv6 bgp sum', 'show bgp sum' or a plain CSV styled data submission. If you used the CSV format, column 1 must be the remote ASN, column 2 must be the remote IPv4 or IPv6 address.
source of truth. your show bgp sum submissions.

ASN, column 2 must be the remote IPv4 or IPv6 address.

telnet@edge1.ams5.nl#show ip bgp sum | i 195.69

<table>
<thead>
<tr>
<th>IP</th>
<th>AS</th>
<th>Status</th>
<th>Time</th>
<th>P</th>
<th>R</th>
<th>Value</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>195.69.144.1</td>
<td>1200</td>
<td>ESTAB</td>
<td>39d16h34m</td>
<td>2</td>
<td>1</td>
<td>3284</td>
<td>0</td>
</tr>
<tr>
<td>195.69.144.26</td>
<td>26496</td>
<td>ADMDN</td>
<td>7d22h59m</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>195.69.144.29</td>
<td>8304</td>
<td>ESTAB</td>
<td>8d10h50m</td>
<td>19</td>
<td>0</td>
<td>3284</td>
<td>0</td>
</tr>
<tr>
<td>195.69.144.32</td>
<td>12871</td>
<td>ESTAB</td>
<td>8d10h53m</td>
<td>13</td>
<td>0</td>
<td>3284</td>
<td>0</td>
</tr>
<tr>
<td>195.69.144.33</td>
<td>559</td>
<td>ESTAB</td>
<td>13d2h48m</td>
<td>96</td>
<td>0</td>
<td>3284</td>
<td>0</td>
</tr>
<tr>
<td>195.69.144.35</td>
<td>12859</td>
<td>ESTAB</td>
<td>41d16h4m</td>
<td>49</td>
<td>0</td>
<td>3284</td>
<td>0</td>
</tr>
<tr>
<td>195.69.144.38</td>
<td>4589</td>
<td>ESTAB</td>
<td>80d18h45m</td>
<td>230</td>
<td>0</td>
<td>3284</td>
<td>0</td>
</tr>
<tr>
<td>195.69.144.39</td>
<td>112</td>
<td>ESTAB</td>
<td>80d18h45m</td>
<td>1</td>
<td>0</td>
<td>3284</td>
<td>0</td>
</tr>
<tr>
<td>195.69.144.42</td>
<td>9145</td>
<td>ESTAB</td>
<td>80d18h45m</td>
<td>114</td>
<td>0</td>
<td>3284</td>
<td>0</td>
</tr>
<tr>
<td>195.69.144.43</td>
<td>2611</td>
<td>ESTAB</td>
<td>27d8h15m</td>
<td>48</td>
<td>0</td>
<td>3284</td>
<td>0</td>
</tr>
<tr>
<td>195.69.144.47</td>
<td>13127</td>
<td>ESTAB</td>
<td>30d5h29m</td>
<td>54</td>
<td>0</td>
<td>3284</td>
<td>0</td>
</tr>
<tr>
<td>195.69.144.48</td>
<td>3265</td>
<td>ESTAB</td>
<td>80d18h46m</td>
<td>37</td>
<td>0</td>
<td>3284</td>
<td>0</td>
</tr>
<tr>
<td>195.69.144.49</td>
<td>1140</td>
<td>ESTAB</td>
<td>29d8h56m</td>
<td>7</td>
<td>0</td>
<td>3284</td>
<td>0</td>
</tr>
<tr>
<td>195.69.144.51</td>
<td>8708</td>
<td>ESTAB</td>
<td>5d3h42m</td>
<td>600</td>
<td>0</td>
<td>3284</td>
<td>0</td>
</tr>
</tbody>
</table>
Confirm your peerings

We've parsed your data, and found the peerings listed below. Is this what you want to submit?

<table>
<thead>
<tr>
<th>ASN</th>
<th>IP</th>
<th>Probable IXP</th>
<th>State</th>
</tr>
</thead>
<tbody>
<tr>
<td>112</td>
<td>195.69.144.39</td>
<td>AMS-IX</td>
<td>UP</td>
</tr>
<tr>
<td>559</td>
<td>195.69.144.33</td>
<td>AMS-IX</td>
<td>UP</td>
</tr>
<tr>
<td>1140</td>
<td>195.69.144.49</td>
<td>AMS-IX</td>
<td>UP</td>
</tr>
<tr>
<td>1200</td>
<td>195.69.144.1</td>
<td>AMS-IX</td>
<td>UP</td>
</tr>
<tr>
<td>2611</td>
<td>195.69.144.43</td>
<td>AMS-IX</td>
<td>UP</td>
</tr>
<tr>
<td>3265</td>
<td>195.69.144.48</td>
<td>AMS-IX</td>
<td>UP</td>
</tr>
<tr>
<td>4589</td>
<td>195.69.144.38</td>
<td>AMS-IX</td>
<td>UP</td>
</tr>
<tr>
<td>5390</td>
<td>195.69.144.55</td>
<td>AMS-IX</td>
<td>UP</td>
</tr>
</tbody>
</table>
Networks which contributed data

Atrato, PCCW, Leaseweb (and many others!)

Statistics

• 33500 unique BGP sightings
• 256 individual submissions
• Over period of 7 days
Source #1: PDB

- 4738 networks in PeeringDB (10% of DFZ)
- 15142 connections are described in PDB
  - IP Address field currently is VARCHAR(128)

Example:

<table>
<thead>
<tr>
<th>Public Peering Exchange Points</th>
<th>ASN</th>
<th>IP Address</th>
<th>Mbit/sec</th>
</tr>
</thead>
<tbody>
<tr>
<td>DE-CIX Frankfurt</td>
<td>5580</td>
<td>80.81.194.66</td>
<td>50000</td>
</tr>
<tr>
<td>DE-CIX Frankfurt</td>
<td>5580</td>
<td>2001:7f8::15cc:0:2</td>
<td>50000</td>
</tr>
</tbody>
</table>
Worst IP address entries

91.232.229.x
2a01.6e00.0010::1
2001:7f8:20:101(2)::244(6):28
“Multiple”
“Soon”
“Coming Aug 1st, 2012”
Parsing / Scoring

1559 entries don’t parse out of the box, let’s try to save a few:

```python
def scrub(ip_address):
    ip = ip_address.strip()
    ip, s, t = ip.partition(' / ')
    try:
        ip = IPAddress(ip)
        return ip
    except:
        return None
```

Voila! 877 entries are recovered 😊
Usability of PDB

96.5%

Of user submitted data is parsable

(only 682 entries kinda suck)
Source #2 – Ground truth

When networks report seeing an established sessions with a certain (IP, ASN) tuple, we’ll take it as ground truth.

Statistics!

• 33500 sightings (28% IPv6 – 72% IPv4)
• 11300 unique (IP, ASN) tuples
• 8937 unique tuples recognizable as IXP IP
Source #2 – Ground truth - Results

• 6721 out of 8937 sightings match with PDB
  – 75% of sightings have corresponding PDB entries
  – 44% of entire PDB data-set has been verified now

• 64 IP addresses conflict between truth and parsable peering records
What are those 64 conflicts?

- 57 IPv4 versus 7 IPv6 entries.
  - (are we making less mistakes with IPv6? 😊)

- 60% - old stale data - IP recycling @ IXP
- 26% - migrations (mergers / acquisitions)
- 14% - False information in PDB (usually typos)
- 8% - No idea, couldn’t figure it out
The end score!

- 75% overlap between peeringdb and sightings
- 44% of peeringdb has now been verified
- Only 64 conflicts...

99% of parsable entries are correct.
Conclusion

• Yes, you can trust peeringDB if you accept fault margin ~ 1%
• Scrubbing the entries is worth the effort
  – 2 lines of python saved 60% of unparsable records
• Keep it up!
How can you improve your part?

- These common guidelines are used for entering data in PeeringDB (although not enforced)
  - 1 IPv4 or IPv6 address per entry
    - (yes, this messes up the total capacity, but who cares)
  - IP addresses without subnet length (not add /22)
  - Etc ....
Automation efforts

- When you start automating your peering, take these things into consideration:
  - Scrubbing the data is worth the effort!
  - PeeringDB 2.0 should include a ‘freshness’ factor for records to work around stale data
  - Should we create a feedback loop with data-sets like these and sanitize the peeringdb dataset?
Questions?

Ps. I will delete the peeringdb census data