Clusters in the Expanse: Understanding and Unbiasing IPv6 Hitlists

Oliver Gasser, Quirin Scheitle, Paweł Foremski, Qasim Lone, Maciej Korczyński, Stephen D. Strowes, Luuk Hendriks, Georg Carle

IMC 2018, Boston
THE EXPANSE of the IPv6 Address Space
THE EXPANSE OF THE IPv6 ADDRESS SPACE
THE

EXPLANSE

of the IPv6 Address Space
Previous work

Previous work on IPv6 address space analysis

- Dhamdhere et al. (2012)
- Czyz et al. (2014)
- Plonka and Berger (2015, 2017)
- Ullrich et al. (2015)
- Gasser et al. (2016)
- Rohrer et al. (2016)
- Foremski et al. (2016)
- Murdock et al. (2017)
- Fiebig et al. (2017, 2018)
- Borgolte et al. (2018)
Open questions

1. How balanced are different hitlist sources?
Open questions

1. How balanced are different hitlist sources?

2. Can we identify addressing schemes to find new addresses?
Open questions

1. How balanced are different hitlist sources?
2. Can we identify addressing schemes to find new addresses?
3. What is the influence of aliased prefixes on IPv6 hitlists?
Open questions

1. How balanced are different hitlist sources?

2. Can we identify addressing schemes to find new addresses?

3. What is the influence of aliased prefixes on IPv6 hitlists?

4. How does cross-protocol responsiveness in IPv6 differ from IPv4?
Open questions

1. How balanced are different hitlist sources?
2. Can we identify addressing schemes to find new addresses?
3. What is the influence of aliased prefixes on IPv6 hitlists?
4. How does cross-protocol responsiveness in IPv6 differ from IPv4?
5. Is there a benefit of using more than one address learning tool?
1. How balanced are different hitlist sources?
Hitlist sources

Where can we learn potential IPv6 addresses?
Hitlist sources

Where can we learn potential IPv6 addresses?

- Domainlists
- FDNS
- CT
- AXFR
- Bitnodes
- RIPE Atlas
- Scamper
Hitlist sources

Where can we learn potential IPv6 addresses?

Figure 1: Cumulative runup of IPv6 addresses.
Hitlist sources
Where can we learn potential IPv6 addresses?

![Cumulative runup of IPv6 addresses](image)

**Figure 1:** Cumulative runup of IPv6 addresses.

Address distribution
- Many addresses from domainlists, CT, and scamper
- Rapid increase of scamper addresses due to CPE routers
Hitlist sources

How balanced are the addresses from different sources?
Hitlist sources

How balanced are the addresses from different sources?

Figure 2: AS distribution for hitlist sources.
Hitlist sources

How balanced are the addresses from different sources?

Autonomous System distribution

- Unbalanced (CT, domainlists) vs. balanced (RIPE Atlas)

Figure 2: AS distribution for hitlist sources.
Hitlist sources

How much of the announced address space do we cover?
Excursion: Visualizing prefixes

Visualizing prefixes using Hilbert space-filling curves

Figure 3: IPv4
Excursion: Visualizing prefixes

Visualizing prefixes using Hilbert space-filling curves

Figure 3: IPv4

Figure 4: IPv6

Figures by Ben Cartwright-Cox https://blog.benjojo.co.uk/post(scan-ping-the-internet-hilbert-curve)
Hitlist sources

How much of the announced address space do we cover?

Figure 5: Number of addresses per prefix.

zesplot

• IPv6 prefix visualization tool
• Input: set of IPv6 prefixes
• Each plotted as rectangle
• Prefixes of same AS and size are plotted adjacently
• Color based on metric (e.g. number of addrs. in prefix)

BGP prefix distribution

• Good coverage of BGP prefixes: 25.5 k of 51.2 k
• Some prefixes with many addresses
Hitlist sources

How much of the announced address space do we cover?

Figure 5: Number of addresses per prefix.
Hitlist sources

How much of the announced address space do we cover?

zesplot

- IPv6 prefix visualization tool
- Input: set of IPv6 prefixes
- Each plotted as rectangle
- Prefixes of same AS and size are plotted adjacently
- Color based on metric (e.g. number of addrs. in prefix)

Figure 5: Number of addresses per prefix.
Hitlist sources

How much of the announced address space do we cover?

Figure 5: Number of addresses per prefix.

**zesplot**

- IPv6 prefix visualization tool
- Input: set of IPv6 prefixes
- Each plotted as rectangle
- Prefixes of same AS and size are plotted adjacently
- Color based on metric (e.g. number of addrs. in prefix)

BGP prefix distribution

- Good coverage of BGP prefixes: 25.5 k of 51.2 k
- Some prefixes with many addresses
2. Can we identify common addressing schemes in our hitlist?
Entropy clustering
Understand addressing patterns in IPv6 hitlists
Entropy clustering
Understand addressing patterns in IPv6 hitlists

Networks have different entropy fingerprints

1. Fingerprint each network
2. Feed to k-means clustering
3. Plot median fingerprints and cluster popularity

Gasser et al. — Clusters in the Expanse: Understanding and Unbiasing IPv6 Hitlists
Entropy clustering
Understand addressing patterns in IPv6 hitlists

Networks have different entropy fingerprints

1. Fingerprint each network
Entropy clustering
Understand addressing patterns in IPv6 hitlists

Networks have different entropy fingerprints

1. Fingerprint each network
2. Feed to k-means clustering
Entropy clustering
Understand addressing patterns in IPv6 hitlists

Networks have different entropy fingerprints

1. Fingerprint each network
2. Feed to k-means clustering
3. Plot median fingerprints and cluster popularity

Gasser et al. — Clusters in the Expanse: Understanding and Unbiasing IPv6 Hitlists
Entropy clustering

IPv6 interface identifiers (IIDs)

Figure 6: Hitlist addressing schemes for IIDs.

The IPv6 networks we cover employ predictable IIDs. Also visible: privacy extensions, modified EUI-64 ($\text{ff:fe}$).
Entropy clustering

IPv6 interface identifiers (IIDs)

Figure 6: Hitlist addressing schemes for IIDs.

- The IPv6 networks we cover employ predictable IIDs
- Also visible: privacy extensions, modified EUI-64 (ff:fe)
Figure 7: Hitlist addressing schemes for full addresses.
Entropy clustering

Full IPv6 fingerprints

Figure 7: Hitlist addressing schemes for full addresses.

- Just a handful of schemes on the Internet
- Addresses largely predictable
3. What is the influence of aliased prefixes on IPv6 hitlists?
Detecting aliased prefixes

Taxonomy:

- **Alias**: another address of the same host
- **Aliased prefix**: whole prefix bound to the same host
- **Bias**: some hosts overrepresented due to aliased prefixes
Detecting aliased prefixes

Taxonomy:

- **Alias**: another address of the same host
- **Aliased prefix**: whole prefix bound to the same host
- **Bias**: some hosts overrepresented due to aliased prefixes

![Diagram showing multi-level aliased prefix detection using pseudo-random probing.](image)

**Figure 8**: Multi-level aliased prefix detection using pseudo-random probing.
Detecting aliased prefixes

Results

• Only 3.2% of prefixes are aliased
• But 46.6% of addresses are in aliased prefixes
• Validated using fingerprinting (iTTL, TCP opts, timestamps)
Detecting aliased prefixes

Results

Figure 9: All prefixes covered by hitlist.

Figure 10: Aliased prefixes.

• Only 3.2% of prefixes are aliased
• But 46.6% of addresses are in aliased prefixes

Validated using fingerprinting (iTTL, TCP opts, timestamps)
Detecting aliased prefixes

Results

- Only 3.2% of prefixes are aliased
- But 46.6% of addresses are in aliased prefixes
- Validated using fingerprinting (iTTL, TCP opts, timestamps)
4. How does cross-protocol responsiveness in IPv6 differ from IPv4?
Cross-protocol responsiveness

• If address responds on protocol X, how likely is it to respond on protocol Y?
• Goal: Identify relevant addresses for specific measurements
Cross-protocol responsiveness

Figure 11: Cross-protocol responsiveness between services.
Cross-protocol responsiveness

Figure 11: Cross-protocol responsiveness between services.

- If responsive to any of the probes → at least 89% probability it will answer to ICMPv6
Cross-protocol responsiveness

<table>
<thead>
<tr>
<th></th>
<th>ICMP</th>
<th>TCP/80</th>
<th>TCP/443</th>
<th>UDP/53</th>
<th>UDP/443</th>
</tr>
</thead>
<tbody>
<tr>
<td>ICMP</td>
<td>1</td>
<td>0.95</td>
<td>0.93</td>
<td>0.89</td>
<td>0.99</td>
</tr>
<tr>
<td>TCP/80</td>
<td>0.29</td>
<td>1</td>
<td>0.91</td>
<td>0.61</td>
<td>0.99</td>
</tr>
<tr>
<td>TCP/443</td>
<td>0.069</td>
<td>0.1</td>
<td>0.14</td>
<td>1</td>
<td>0.029</td>
</tr>
<tr>
<td>UDP/53</td>
<td>0.017</td>
<td>0.035</td>
<td>0.054</td>
<td>0.0065</td>
<td>1</td>
</tr>
<tr>
<td>UDP/443</td>
<td>0.007</td>
<td>0.011</td>
<td>0.017</td>
<td>0.005</td>
<td>1</td>
</tr>
</tbody>
</table>

**Figure 11:** Cross-protocol responsiveness between services.

- If responsive to any of the probes → at least 89% probability it will answer to ICMPv6 vs. 73% in IPv4
Cross-protocol responsiveness

Figure 11: Cross-protocol responsiveness between services.

- If responsive to any of the probes → at least 89% probability it will answer to ICMPv6 vs. 73% in IPv4
- Web protocols: QUIC → HTTPS and HTTP, HTTPS → HTTP; but not the other way around

Gasser et al. — Clusters in the Expanse: Understanding and Unbiasing IPv6 Hitlists
5. Is there a benefit of using more than one address learning tool?
Learning new addresses

Techniques to learn new addresses

- Entropy/IP: Generate new addresses by leveraging entropy of seed addresses
  - Similar approach to grouping addresses based on their structure as shown earlier

Examples of generated addresses:
- 2001:0db8:0407:8000::4
- 2001:0db8:0407:8000::5
- 2001:0db8:0407:8000::8
- Likely other valid addresses:
  - 2001:0db8:0407:8000::6
  - 2001:0db8:0407:8000::7
Learning new addresses

Techniques to learn new addresses

- **Entropy/IP**: Generate new addresses by leveraging entropy of seed addresses
  - Similar approach to grouping addresses based on their structure as shown earlier
- **6Gen**: Generate new addresses in dense address regions
  - If we see addresses
    - 2001:0db8:0407:8000::4
    - 2001:0db8:0407:8000::5
    - 2001:0db8:0407:8000::8
  - Likely other valid addresses
    - 2001:0db8:0407:8000::6
    - 2001:0db8:0407:8000::7
Learning new addresses
How well do Entropy/IP and 6Gen perform?

- Input: All previously found IPv6 addresses
- Responsiveness: 278 k (of 118 M) and 489 k (of 129 M)
Learning new addresses
How well do Entropy/IP and 6Gen perform?

- Input: All previously found IPv6 addresses
- Responsiveness: 278 k (of 118 M) and 489 k (of 129 M)
- Overlap of only 675 k generated addresses
- 10x higher response rate for overlapping addresses
Learning new addresses
How well do Entropy/IP and 6Gen perform?

- Input: All previously found IPv6 addresses
- Responsiveness: 278 k (of 118 M) and 489 k (of 129 M)
- Overlap of only 675 k generated addresses
- 10x higher response rate for overlapping addresses

Table 1: Top 5 responsive protocol combinations for Entropy/IP and 6Gen.

<table>
<thead>
<tr>
<th>ICMPv6</th>
<th>TCP/80</th>
<th>TCP/443</th>
<th>UDP/53</th>
<th>UDP/443</th>
<th>Entropy/IP</th>
<th>6Gen</th>
</tr>
</thead>
<tbody>
<tr>
<td>✓</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>41.1 %</td>
<td>66.8 %</td>
</tr>
<tr>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>X</td>
<td>X</td>
<td>12.3 %</td>
<td>9.2 %</td>
</tr>
<tr>
<td>X</td>
<td>X</td>
<td>X</td>
<td>✓</td>
<td>X</td>
<td>23.1 %</td>
<td>7.3 %</td>
</tr>
<tr>
<td>✓</td>
<td>✓</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>3.4 %</td>
<td>4.9 %</td>
</tr>
<tr>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>X</td>
<td>✓</td>
<td>6.1 %</td>
<td>3.2 %</td>
</tr>
</tbody>
</table>

- Different host populations
Community contributions

Reproducibility

- We publish data, code, and analysis scripts
- DOI: 10.14459/2018mp1452739

Software and tools published on GitHub

- ZMapv6
- zesplot
- Entropy clustering
- New Entropy/IP generator
- Entropy/IP open-sourced (thanks to Akamai)
IPv6 Hitlist Service

A one-off analysis is all well and good, but what if I need an up-to-date IPv6 hitlist for my research starting next month?
IPv6 Hitlist Service

A one-off analysis is all well and good, but what if I need an up-to-date IPv6 hitlist for my research starting next month?

ipv6hitlist.github.io
IPv6 Hitlist Service

A one-off analysis is all well and good, but what if I need an up-to-date IPv6 hitlist for my research starting next month?

ipv6hitlist.github.io

- Daily IPv6 hitlists and aliased prefixes available for download
- Interactive zesplots
- Continuously updated graphs
Sources differently balanced
Repeating addressing schemes
Aliased prefixes
Conditional responsiveness
Learning unknowns
ipv6hitlist.github.io
Oliver Gasser <gasser@net.in.tum.de>
Sources differently balanced

Aliased prefixes

Conditional responsiveness

Repeating addressing schemes

ipv6hitlist.github.io

Oliver Gasser <gasser@net.in.tum.de>
Sources differently balanced

Repeating addressing schemes

Oliver Gasser &lt;gasser@net.in.tum.de&gt;
Sources differently balanced

Repeating addressing schemes

Aliased prefixes

ipv6hitlist.github.io

Oliver Gasser <gasser@net.in.tum.de>
Sources differently balanced

Repeating addressing schemes

Aliased prefixes

Conditional responsiveness

Oliver Gasser <gasser@net.in.tum.de>
Sources differently balanced

Repeating addressing schemes

Aliased prefixes

Conditional responsiveness

Learning unknowns
Sources differently balanced
Repeating addressing schemes
Aliased prefixes
Conditional responsiveness
Learning unknowns

ipv6hitlist.github.io
Oliver Gasser <gasser@net.in.tum.de>