More Specific Announcements in BGP

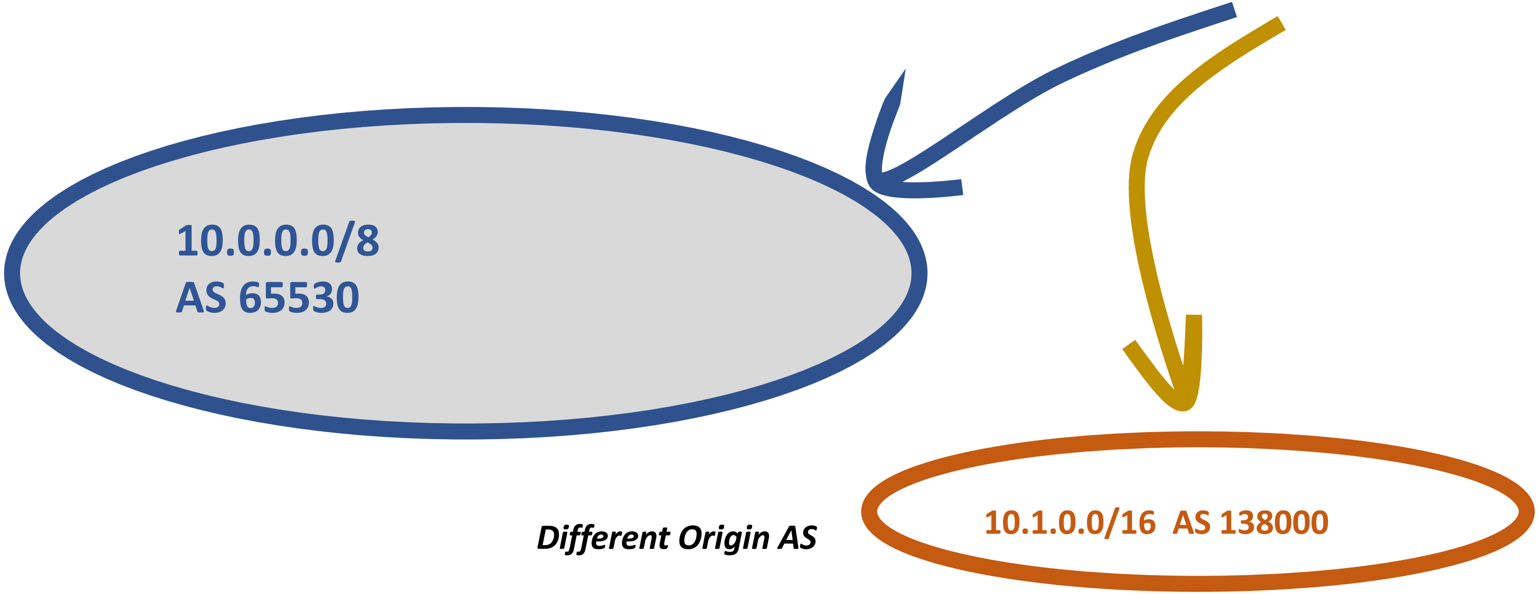
Geoff Huston
APNIC

What's a more specific?

A prefix advertisement that refines a “covering” advertisement

Why advertise a more specific?

I: To redirect packets to a different network: "**hole punching prefixes**"

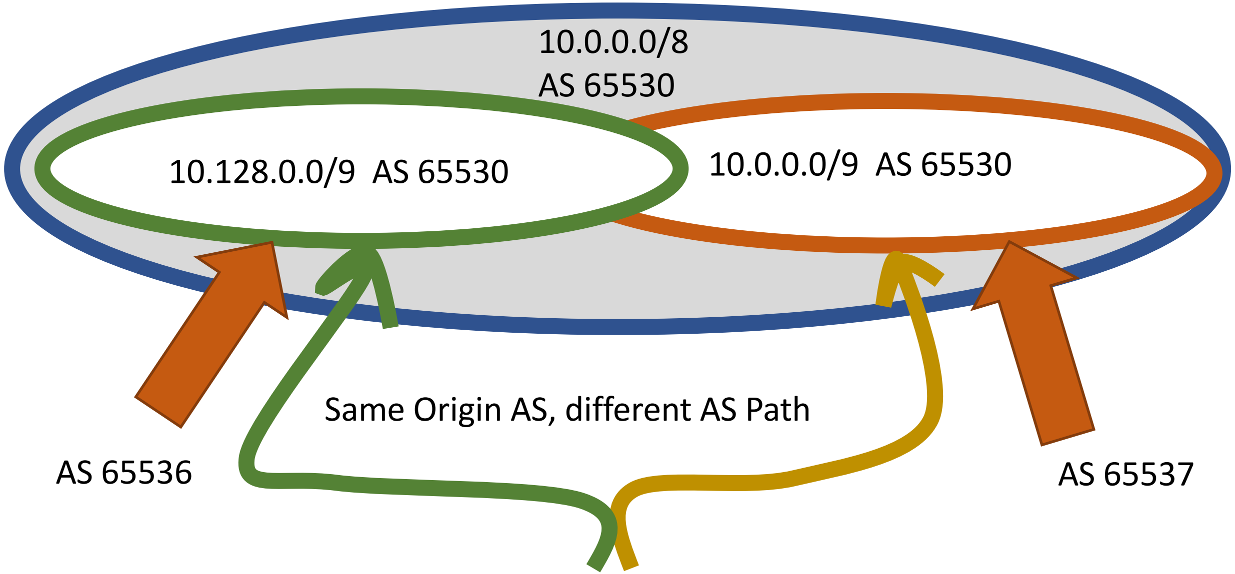


Example: Type I

```
   Network                Path
>* 72.249.184.0/21       4777 2497 3356 36024
>* 72.249.184.0/24       4777 2497 2914 40824 394094
```

Why advertise a more specific?

II: To redirect incoming traffic to different network paths: "**traffic engineering prefixes**"

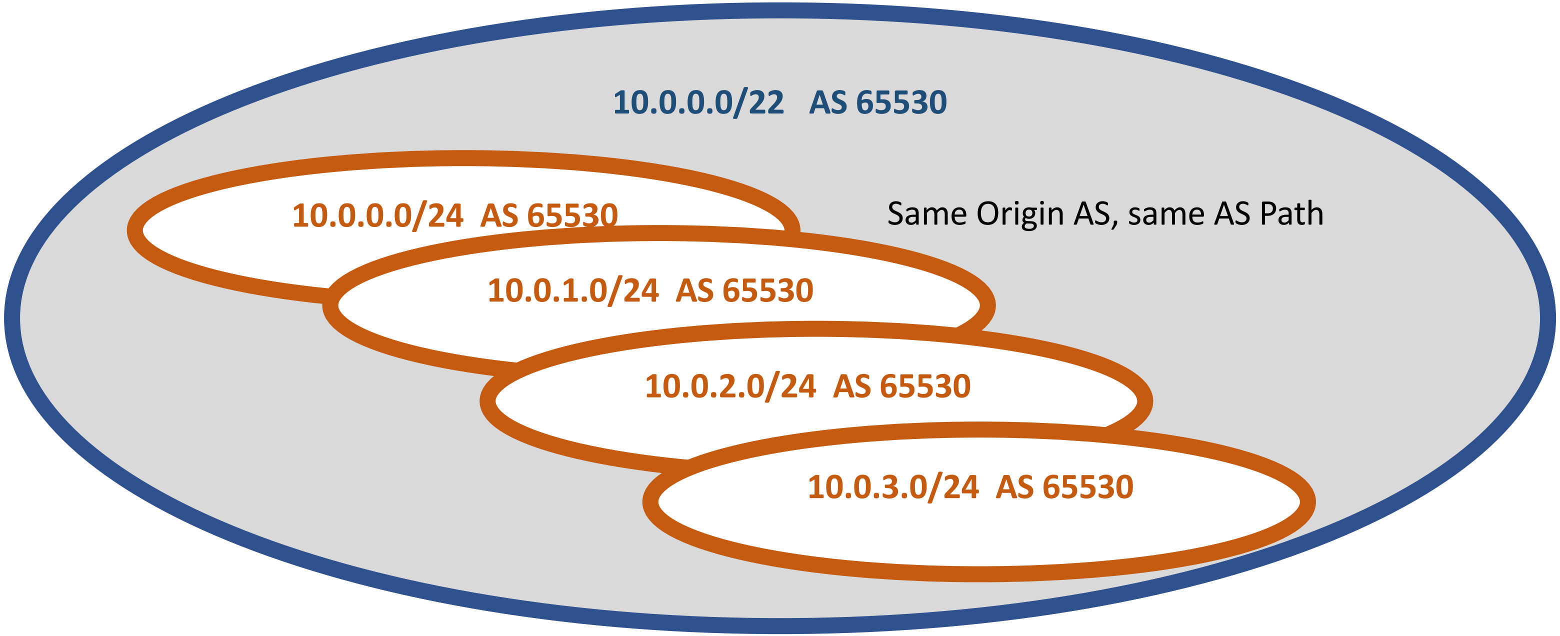


Example: Type II

```
 Network           Path
*> 1.37.0.0/16     4608 1221 4637 4775 i
*> 1.37.27.0/24    4608 1221 4637 4837 4775 i
*> 1.37.237.0/24   4608 1221 4637 4837 4775 i
```

Why advertise a more specific?

III: To mitigate more specific prefix hijacking: "**more specific overlays**"



Example: Type III

```
   Network             Path
*> 1.0.4.0/22        4608 4826 38803 56203 i
*> 1.0.4.0/24        4608 4826 38803 56203 i
*> 1.0.5.0/24        4608 4826 38803 56203 i
*> 1.0.6.0/24        4608 4826 38803 56203 i
*> 1.0.7.0/24        4608 4826 38803 56203 i
```

How many eBGP route advertisements are more specifics?

AS 131072 – 13 October 2017

	Routes		Advertised Address Span	
BGP Routes:	685,895		2.86B /32s	
More Specifics:	365,022	(53%)	1.04B /32s	(36%)

How many eBGP route advertisements are more specifics?

AS 131072 – 13 October 2017

Routes

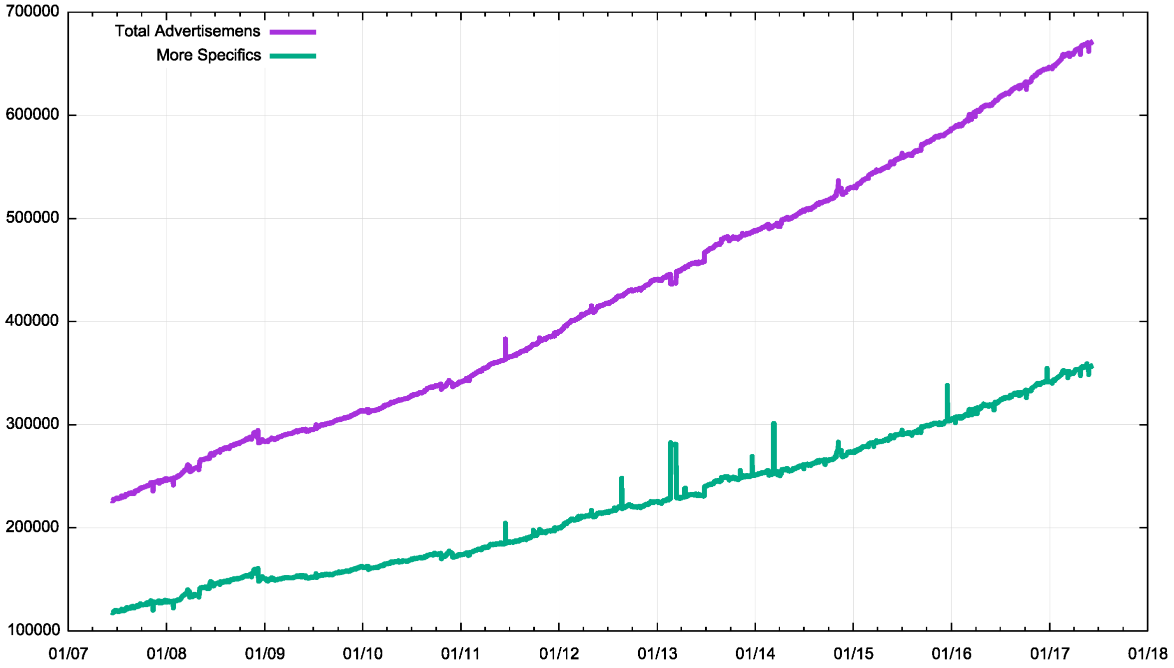
BGP Routes:

Mc

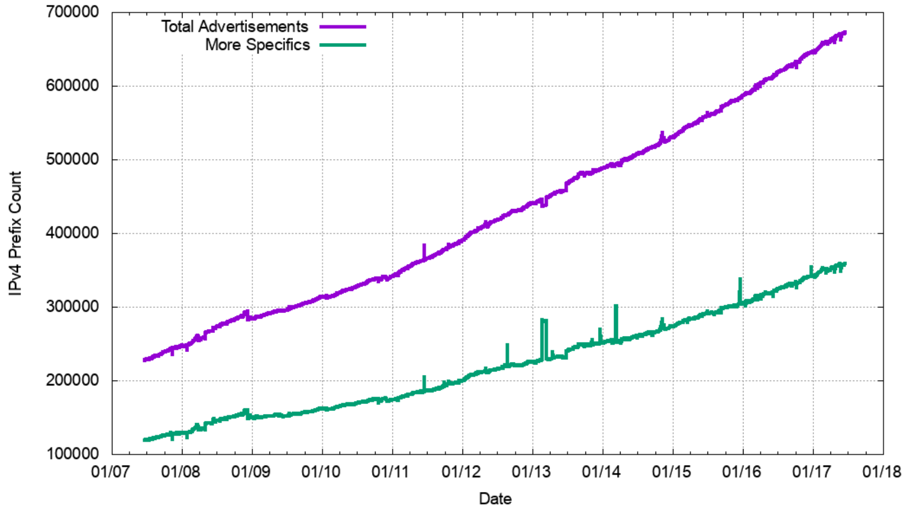
Over one half of the BGP routing table does not announce reachability to "new" destinations, but attempts to refine the way in which existing destinations can be reached

Has this changed over time?

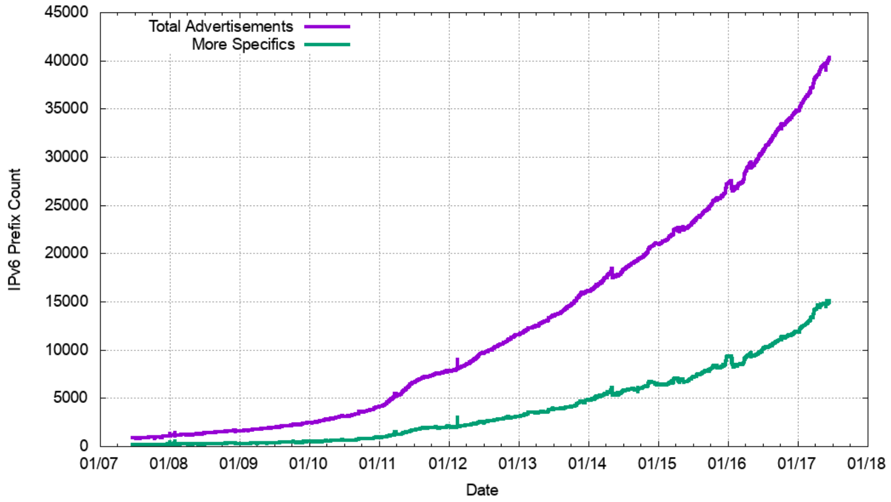
IPv4



Has this changed over time?



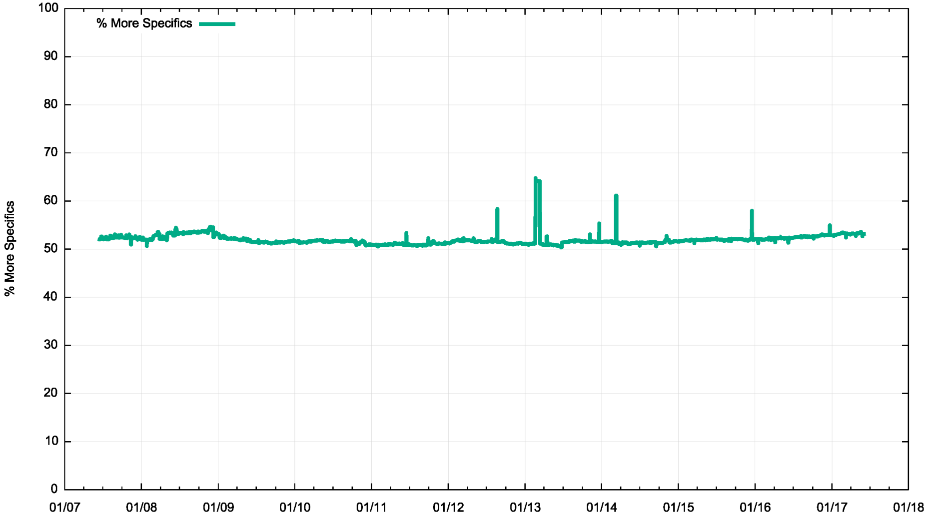
IPv4



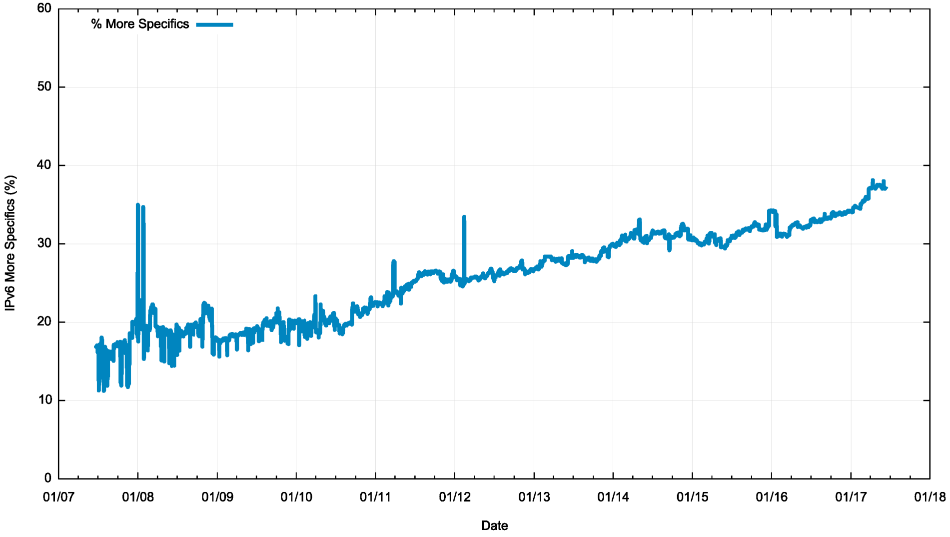
IPv6

Has this changed over time?

Lets use the ratio of More-Specifics to the total route set

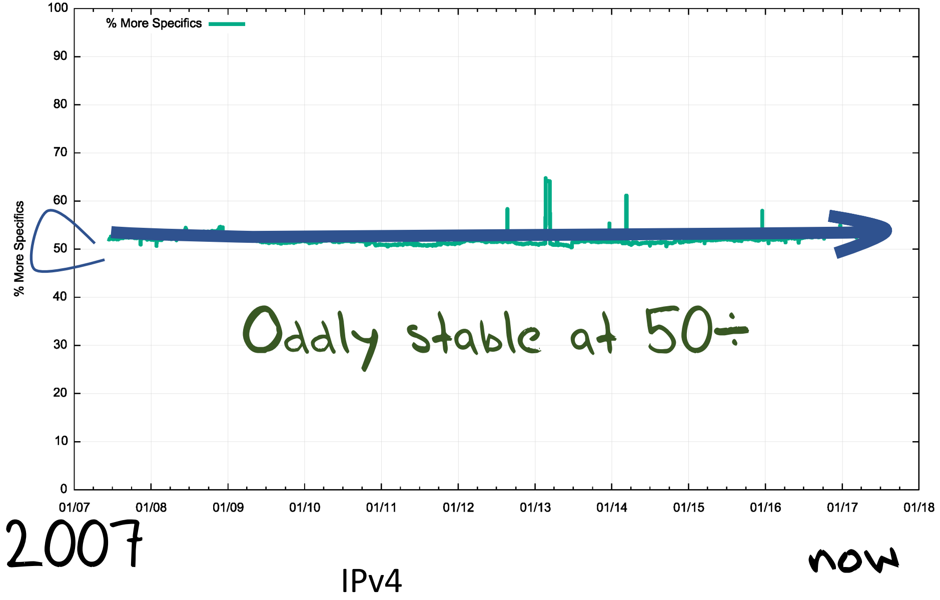


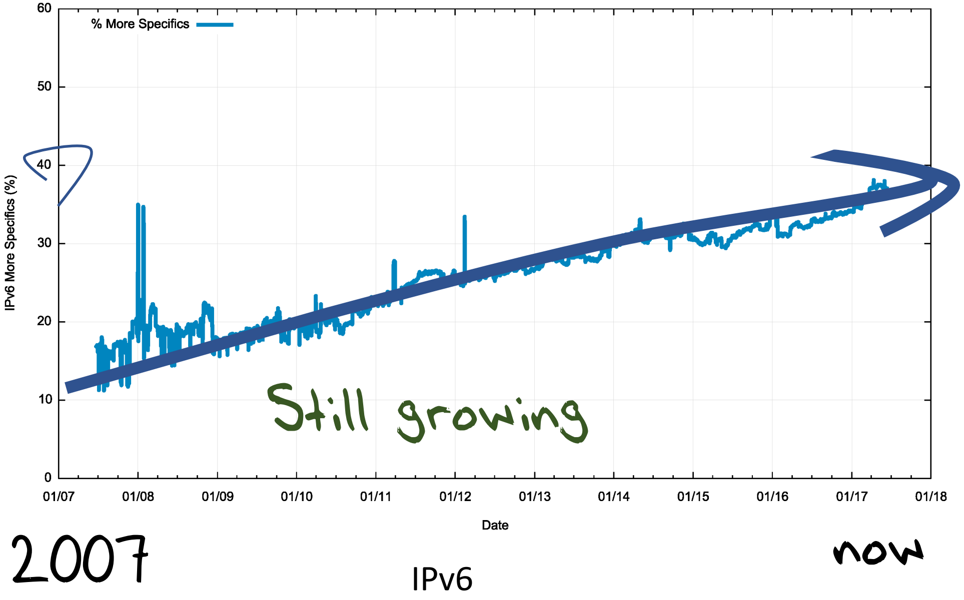
IPv4



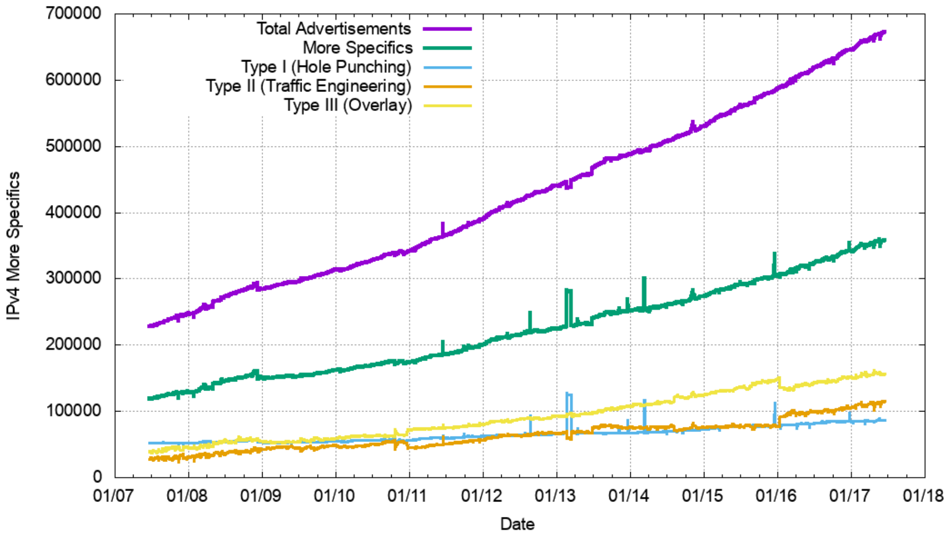
IPv6

Has this changed over time?

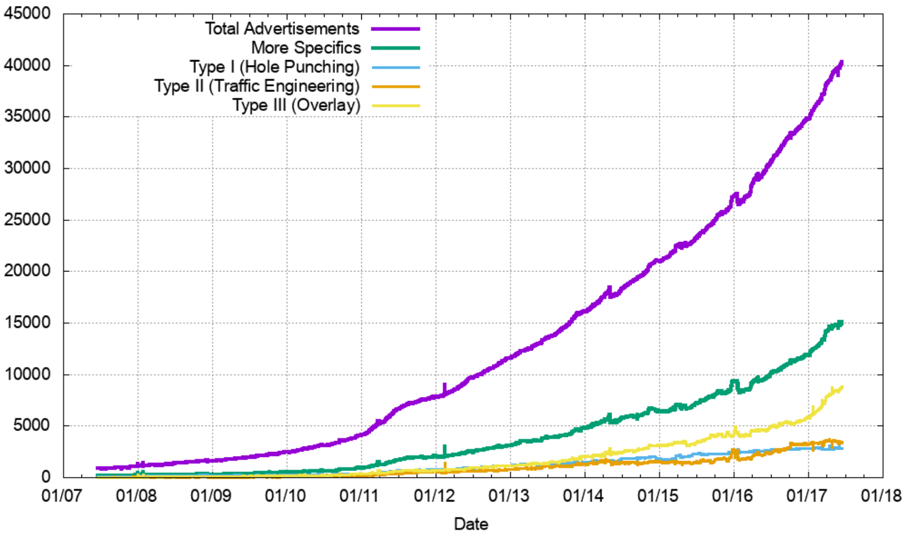




More Specific Types - Prefix Counts

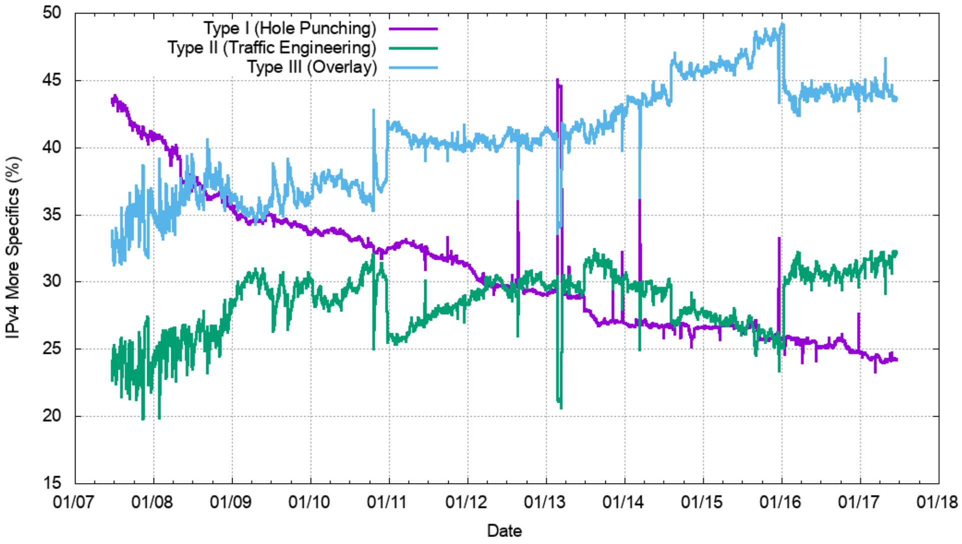


IPv4

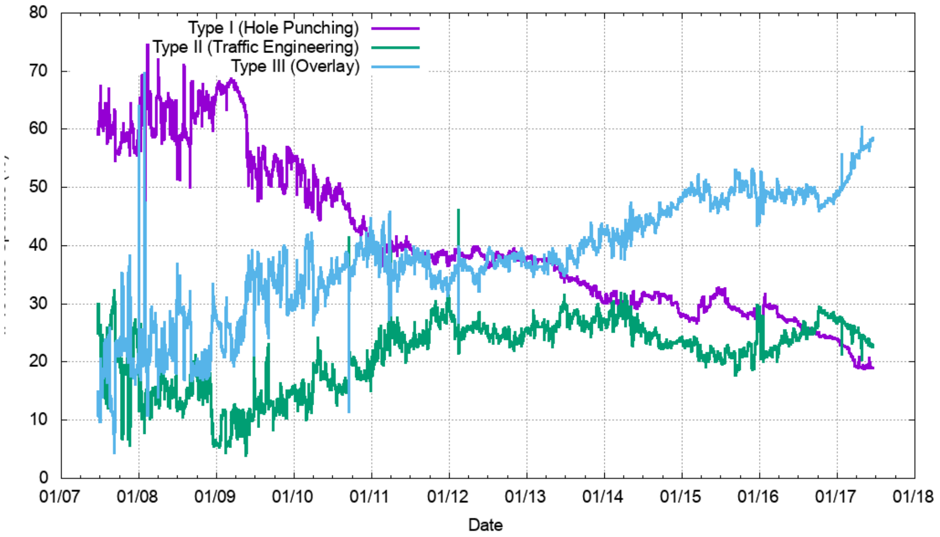


IPv6

More Specific Types - Relative Counts

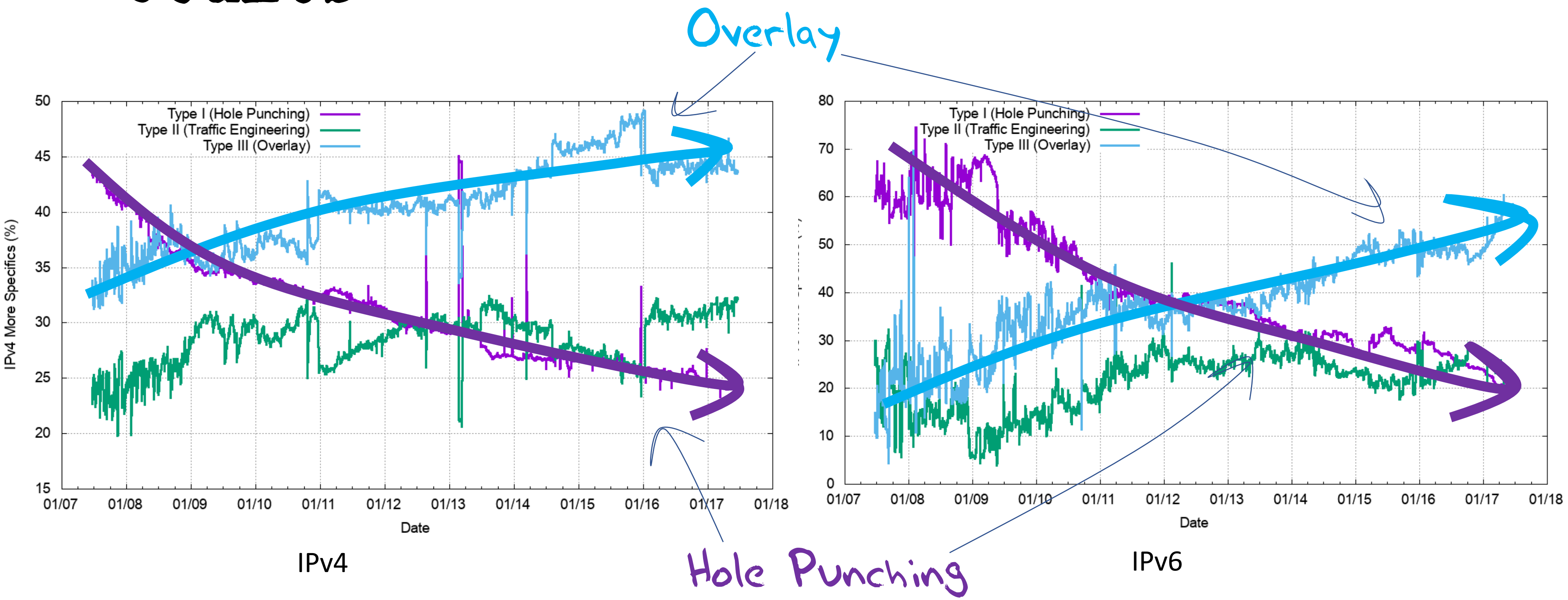


IPv4



IPv6

More Specific Types - Relative Counts



IPv4

IPv6

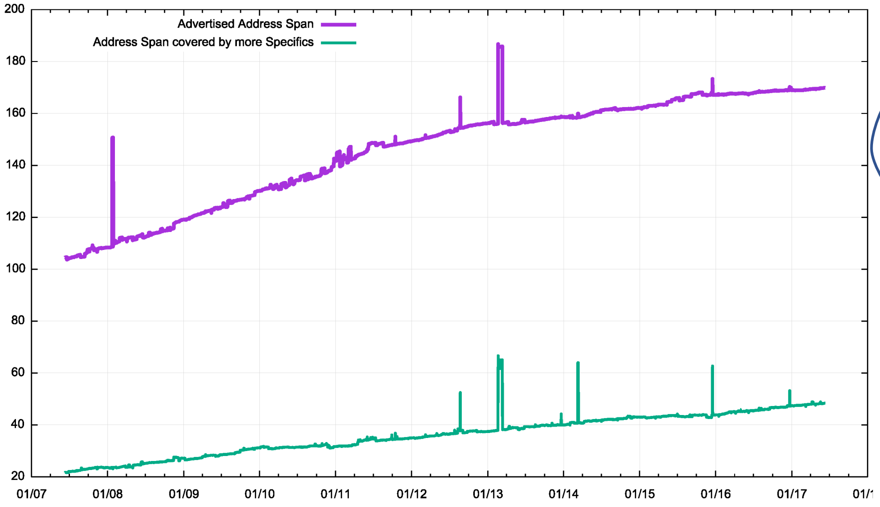
More Specific Types

In both IPv4 and IPv6:

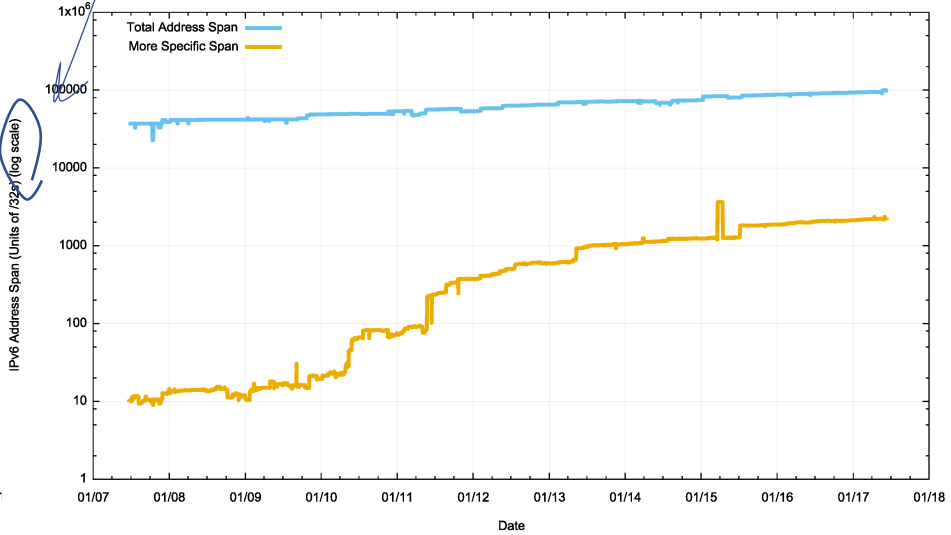
- Type I prefixes (”hole punching”) are declining over time (relatively)
- Type II prefixes (“traffic engineering”) have been relatively constant at some 30% of more specifics
- Type III prefixes (“overlays”) have risen (relatively) and are now the more prevalent form of advertised more specifics

What about Address Spans covered by more specifics?

Yes, this is a LOG scale



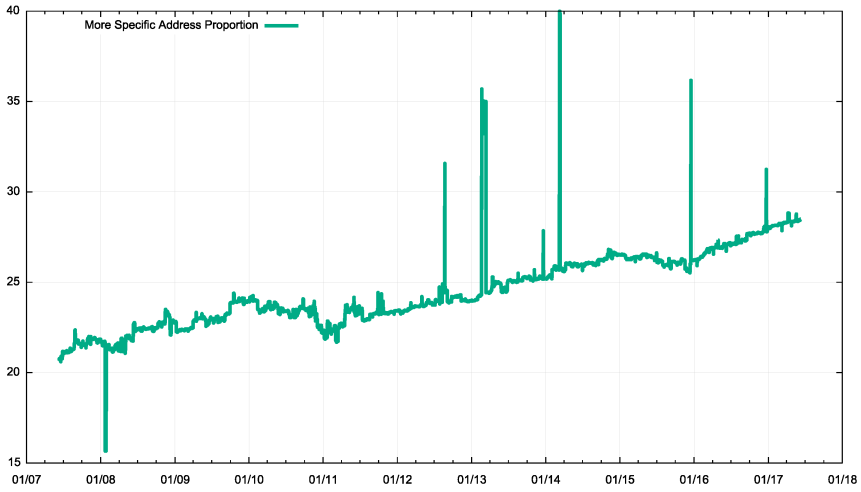
IPv4



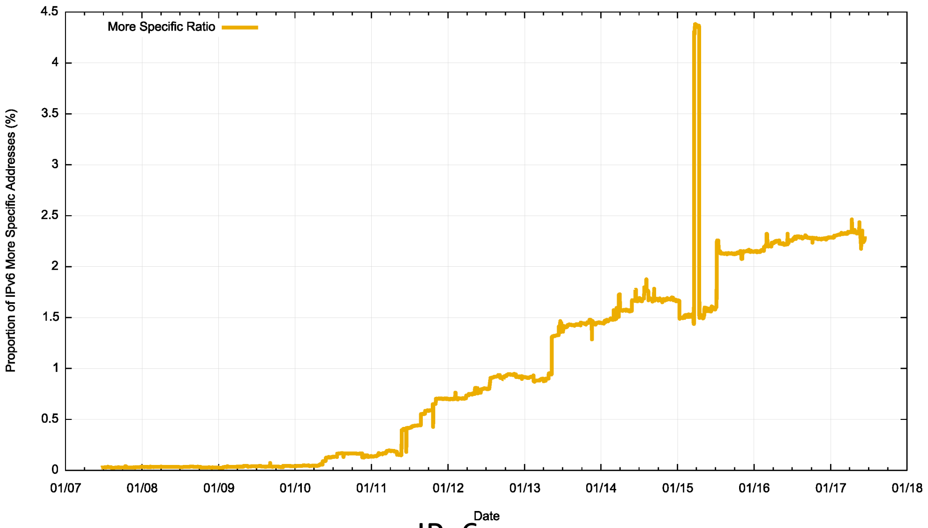
IPv6

What about Address Spans covered by more specifics?

Lets use the ratio of More-Specifics to the total address span



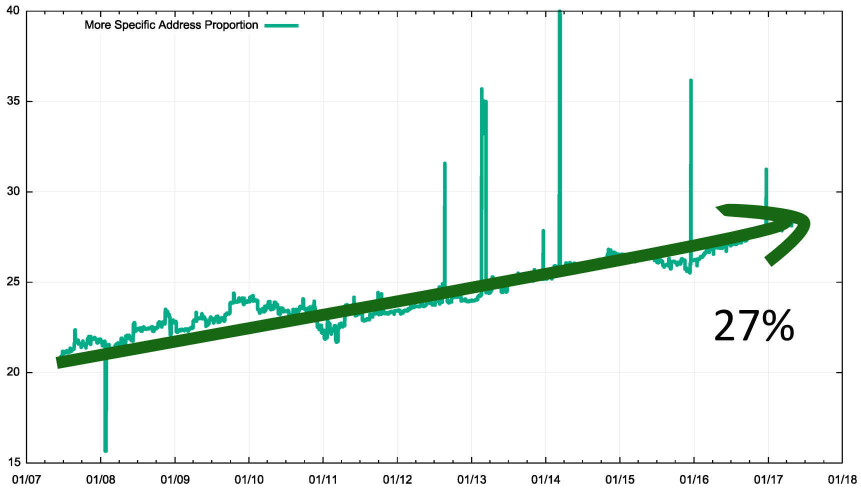
IPv4



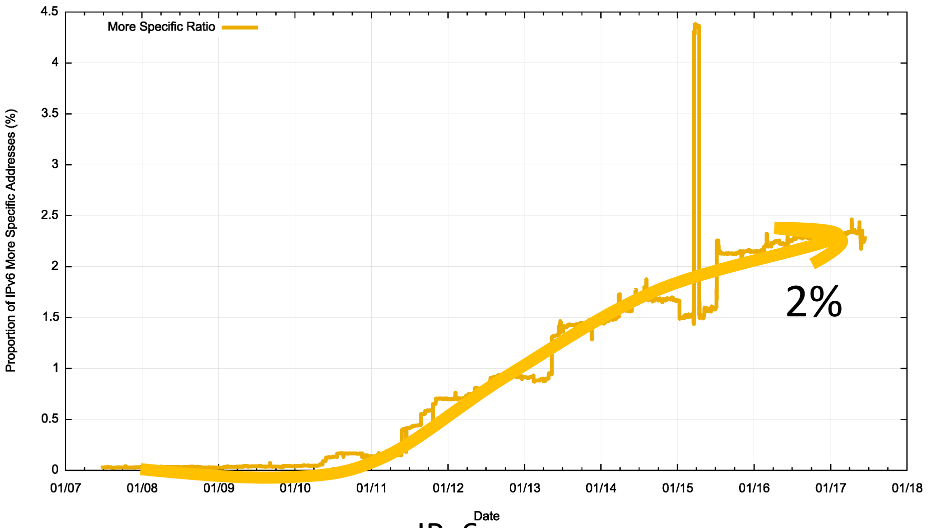
IPv6

What about Address Spans covered by more specifics?

Lets use the ratio of More-Specifics to the total address span

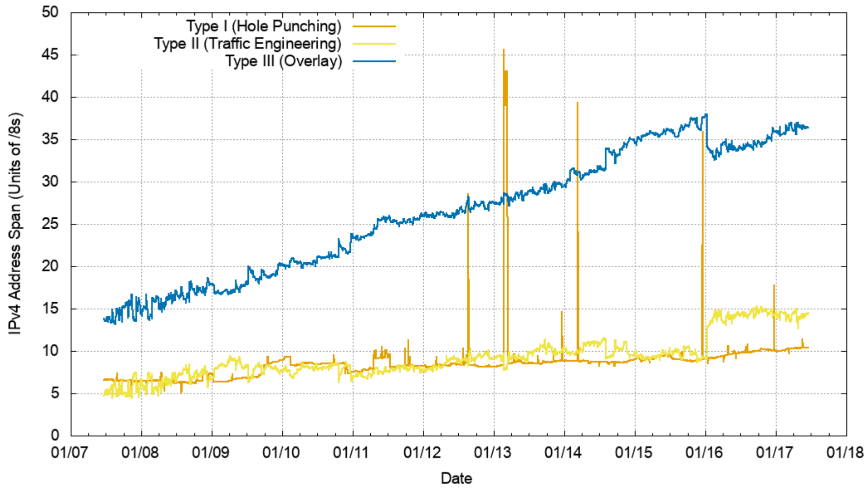


IPv4

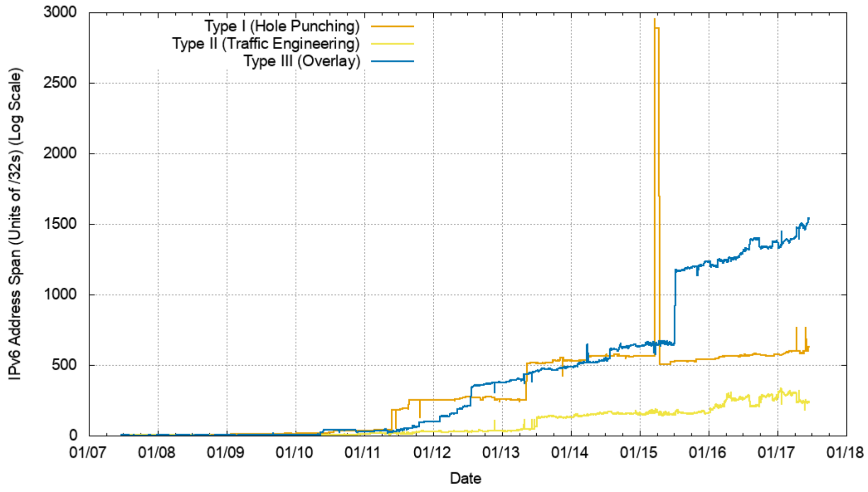


IPv6

Address Span: Breakdown into Types

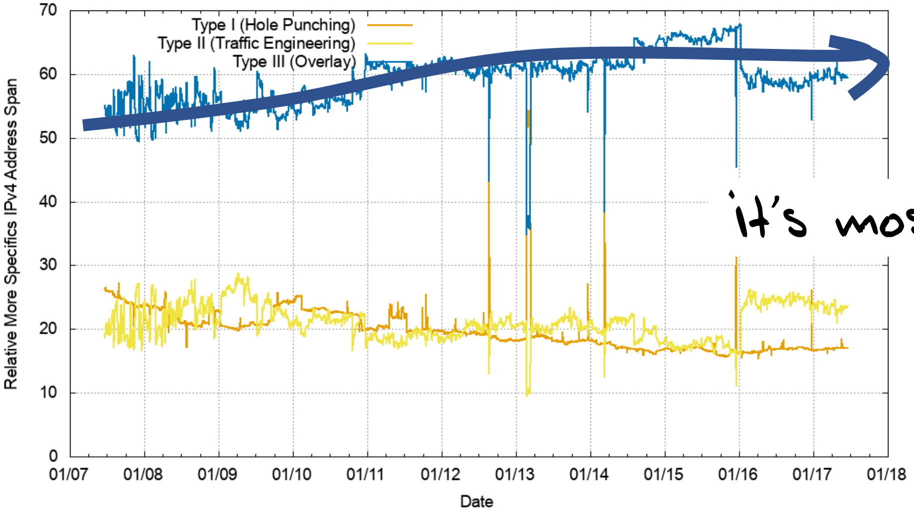


IPv4



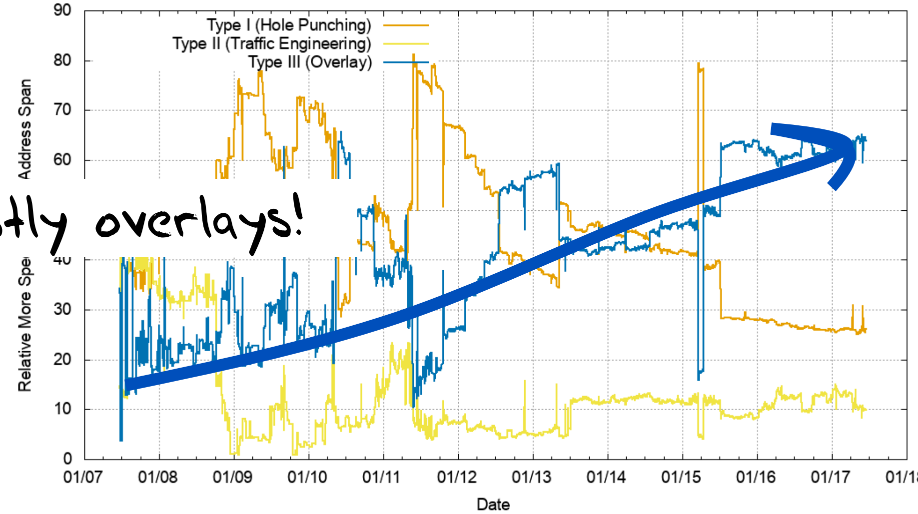
IPv6

% of Total Span: Breakdown into Types



IPv4

it's mostly overlays!



IPv6

Overlays are the majority of More Specifics

- The initial IPv6 network had little in the way of overlays and had a high proportion of Type I (Hole Punching) more specifics. This has changed over time and the recent profile is similar to IPv4

- In both protocols the largest block of more specific announcements in terms of address span are “overlays” where the AS Path of the enclosing aggregate and the more specific are identical

Overlays are the majority of More Specifics

Are Overlays "polluting" the BGP Space?

- Overlays do not change routing, but do these more specifics add to the routing load?

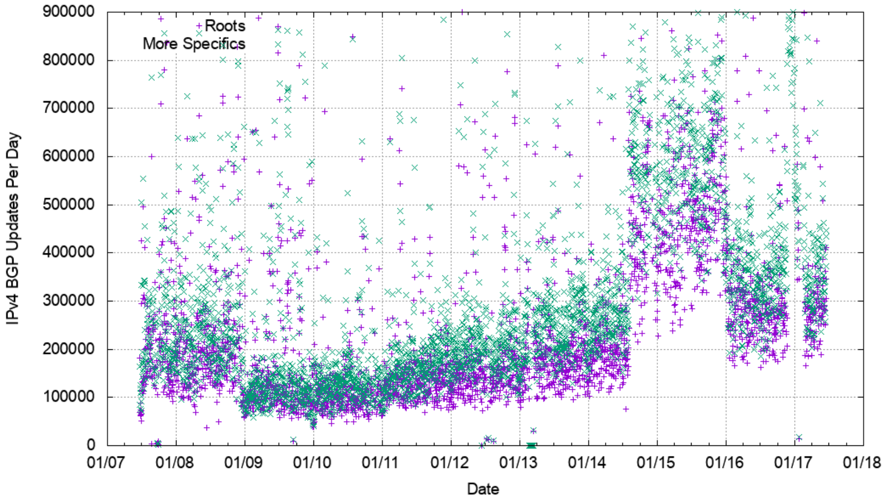
Updates

- Let’s look at “routing load” by looking at BGP updates

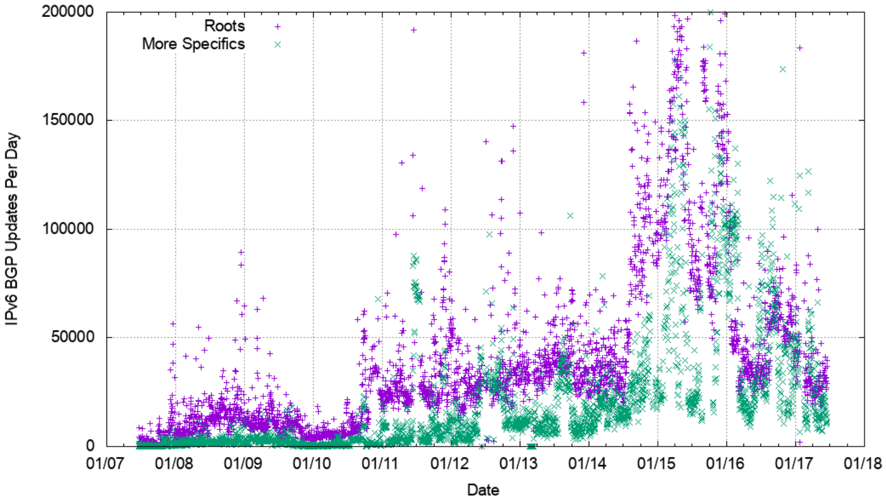
- Our questions are:
  - Are more specifics “noisier” than aggregates?

  and
  - Are overlays more active in terms of BGP Updates than other prefixes?

Update count by Prefix Type

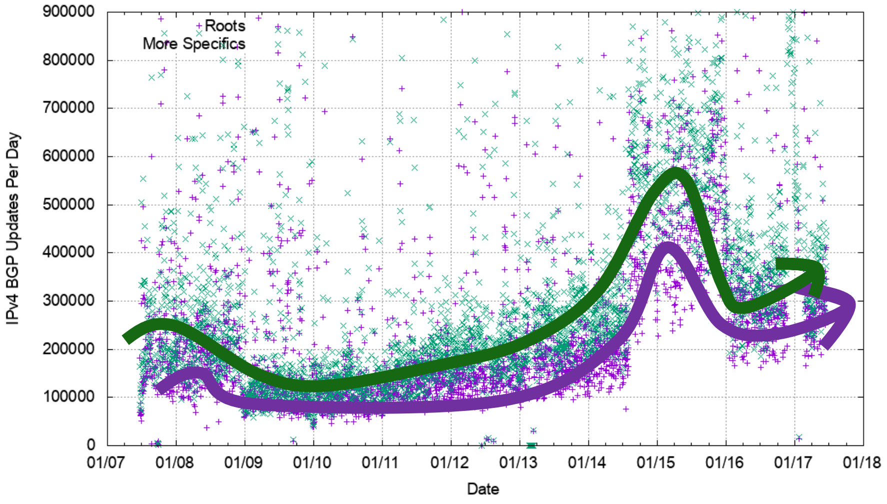


IPv4

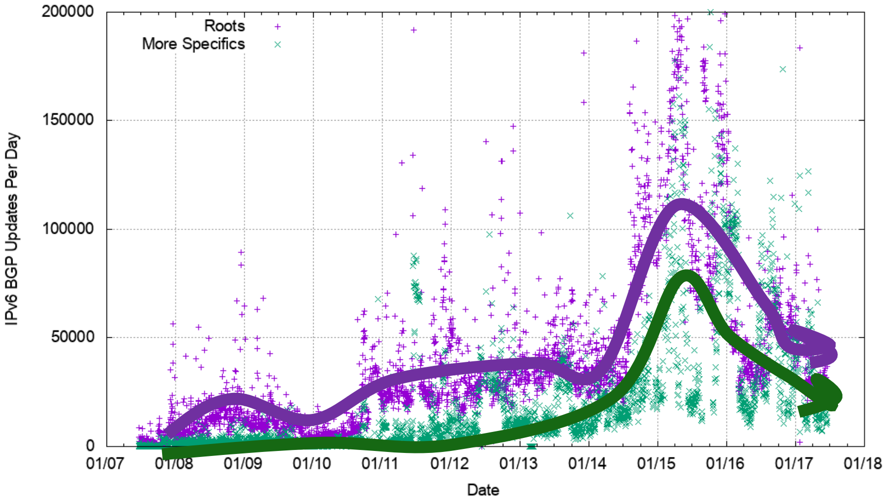


IPv6

Update count by Prefix Type



IPv4

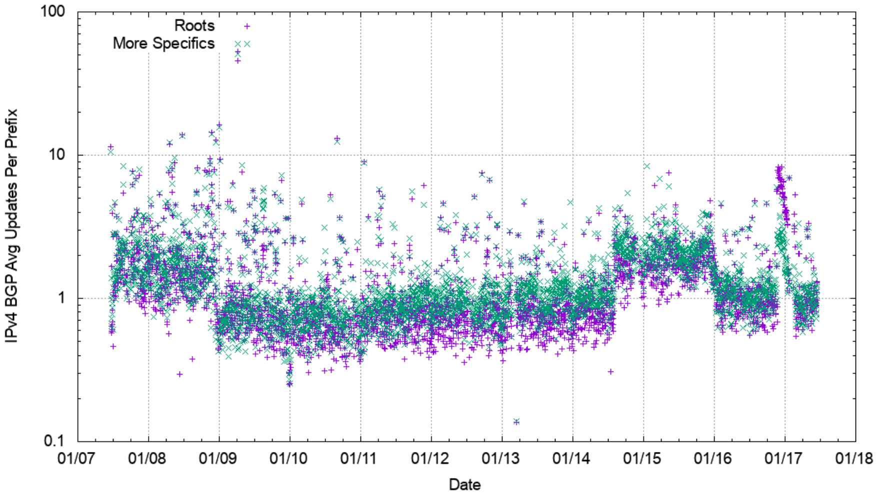


IPv6

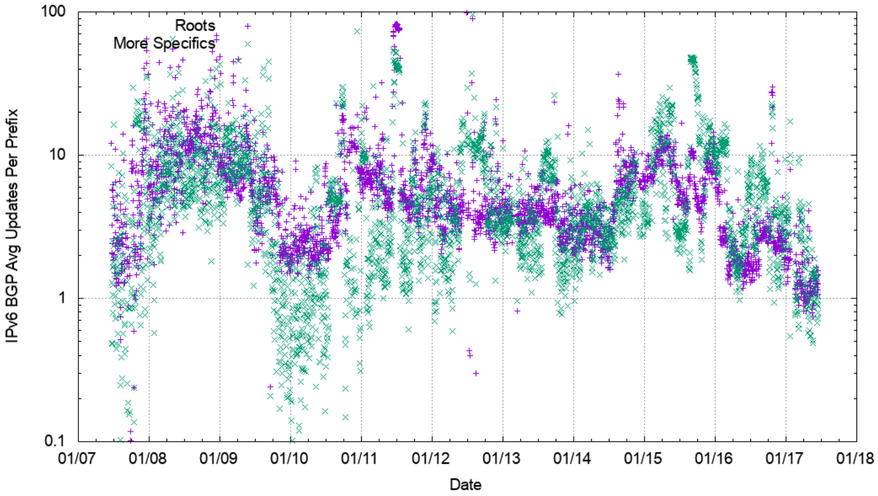
Update Count

- In IPv4 the update count for more specific prefixes is greater than the comparable count for root prefixes, while the opposite is the case in IPv6.
- But the relative count of more specifics is lower in IPv6
- Let’s “normalise” this by dividing the update count by the number of prefixes to get the average update count per prefix of each type

Relative Update count by Prefix Type



IPv4



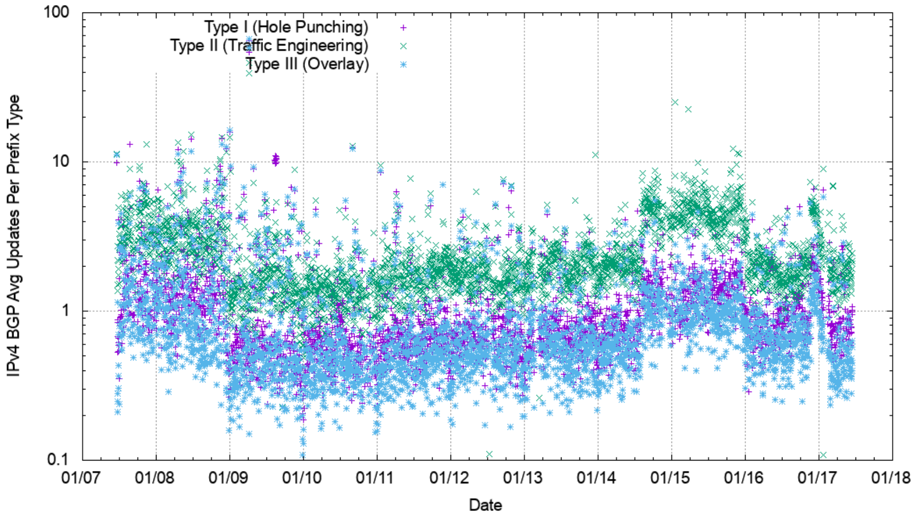
IPv6

Relative Updates

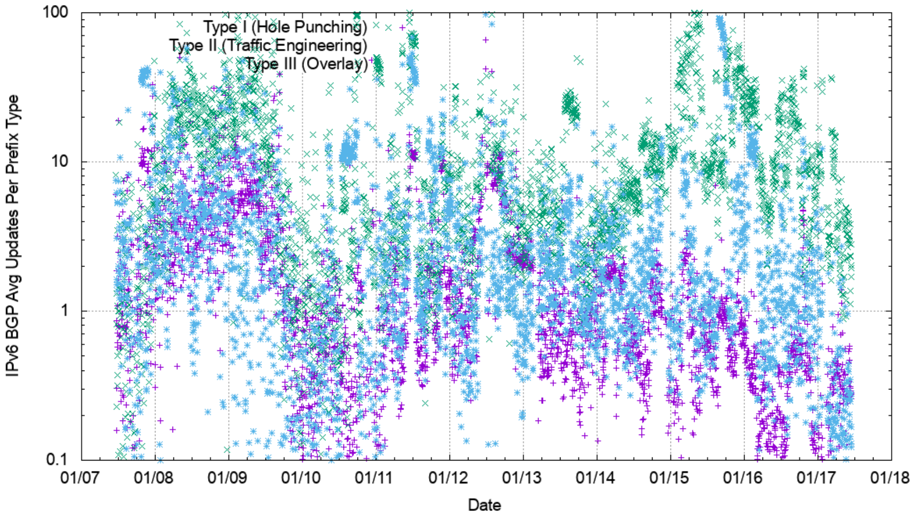
- On average, in IPv4 More Specifics are slightly noisier than IPv4 Roots, while in IPv6 roots and more specifics are equally likely to be the subject of BGP updates

- Are different types of more specifics more or less stable in BGP terms?

Average Number of Updates Per More Specific Prefix Type



IPv4

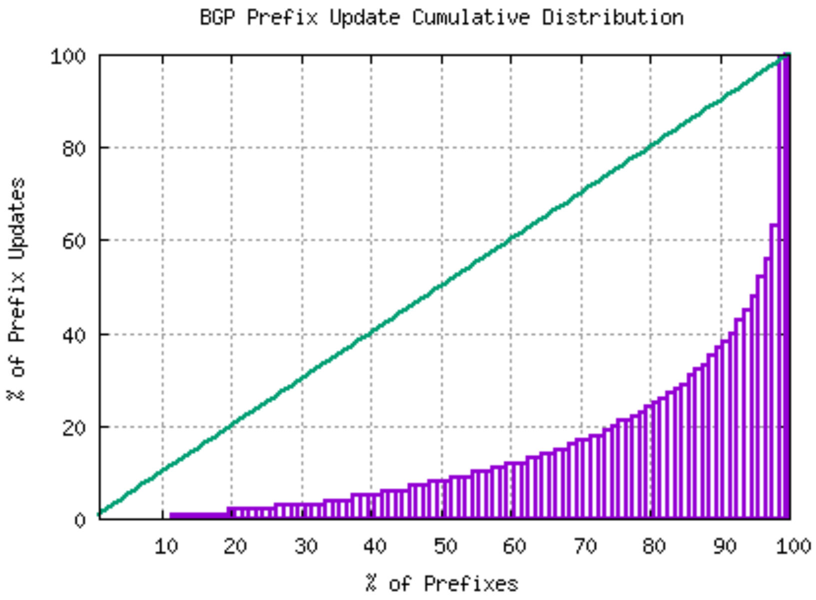


IPv6

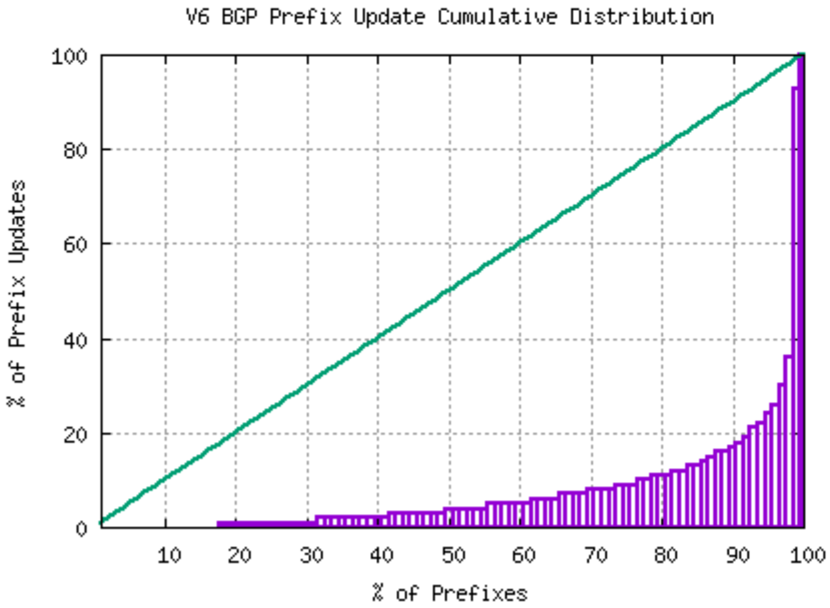
Average Number of Updates Per More Specific Prefix Type

- In IPv4 Type II Traffic Engineering Prefixes show a slightly higher level of BGP instability on average over Type I Hole Punching Prefixes, while Type III Overlay Prefixes show the highest levels of stability of more specifics
- In IPv6 this has only been apparent in the past three years which Type II Traffic Engineering Prefixes showing the greatest levels of BGP instability , and Type I Hole Punching more specifics showing the highest levels of relative stability

BGP Instability is heavily skewed

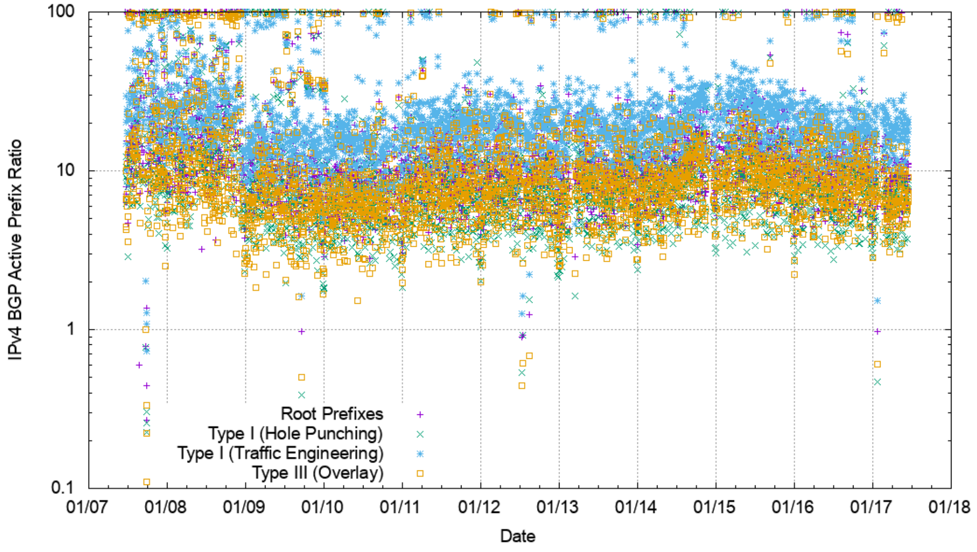


IPv4

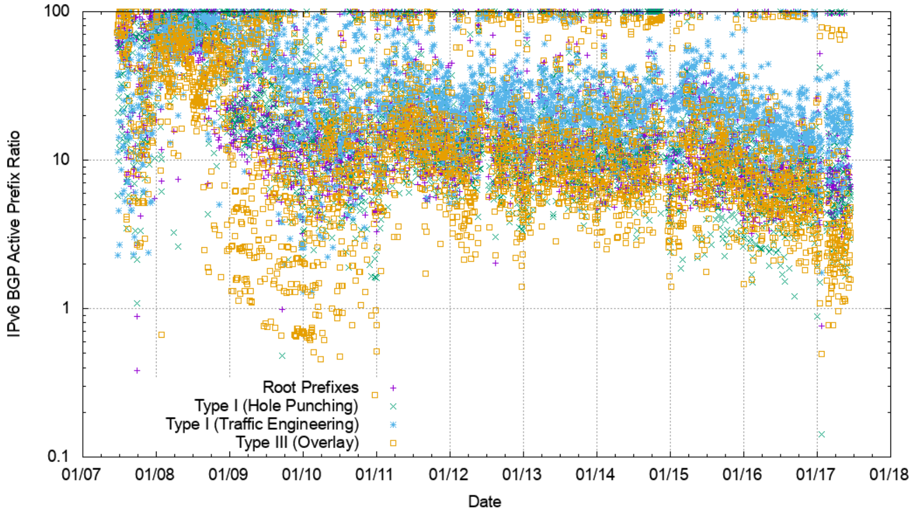


IPv6

What Type of Prefixes are more Unstable?



IPv4



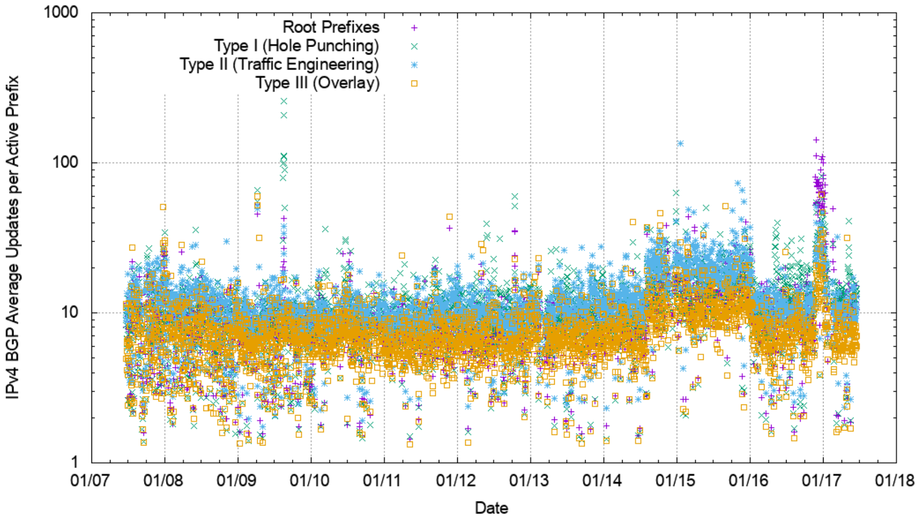
IPv6

What Type of Prefixes are more Unstable?

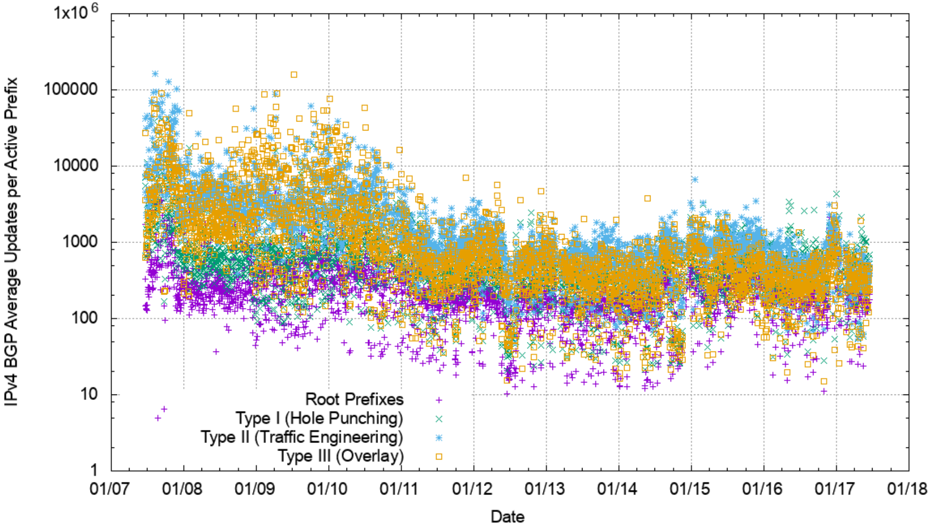
- Type II More Specific Prefixes (Traffic Engineering) are approximately twice as likely to be unstable than either root prefixes or other types of More Specifics in both IPv4 and IPv6

- This matches a rough intuition about the nature of more specifics, where overlays and hole punching would be expected to be as stable as root announcements

Average Number of Updates per Active Prefix Type



IPv4



IPv6

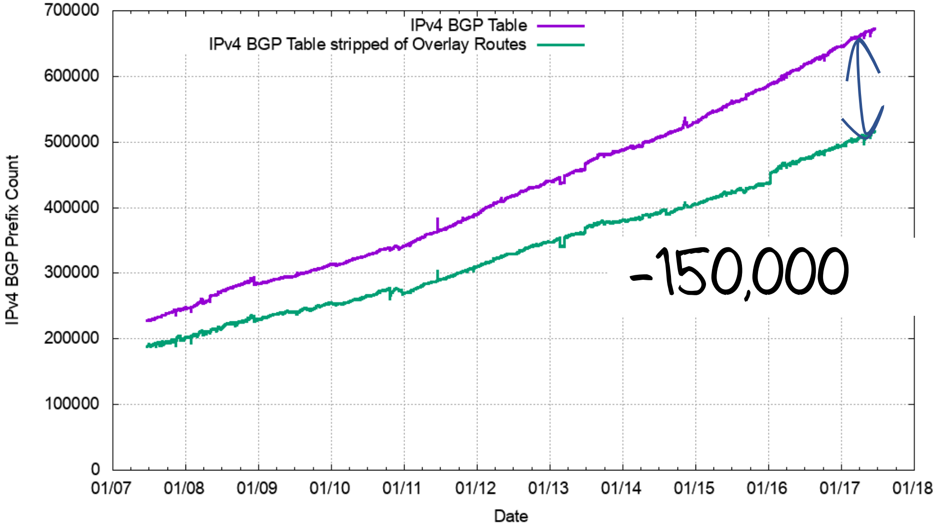
Average Number of Updates per Active Prefix Type

- In IPv4 Type II Traffic Engineering Prefixes have a greater average number of updates than other prefix types
- In IPv6 Root Prefixes tend to have a lower average number of updates than other prefix types
- Perhaps the significant message here is that IPv6 has a higher inherent level of instability – unstable prefixes in IPv6 have 100x more instability events per unstable prefix on average than unstable prefixes in IPv4

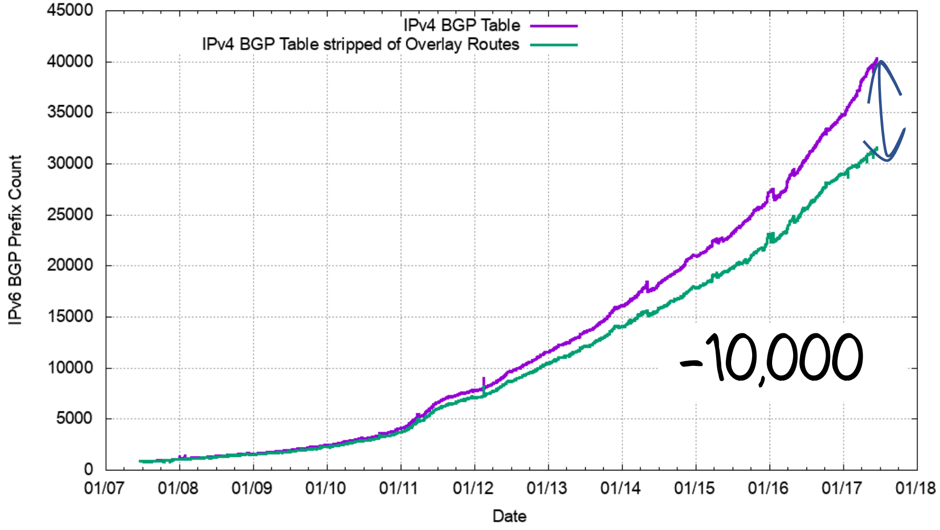
What if…

- All Overlay more specific prefixes were removed from the routing table?

Table Size Implications

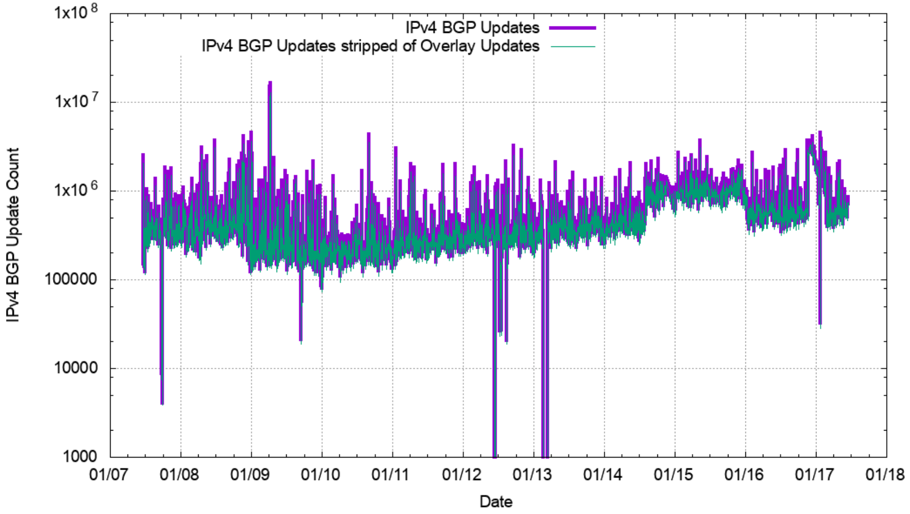


IPv4

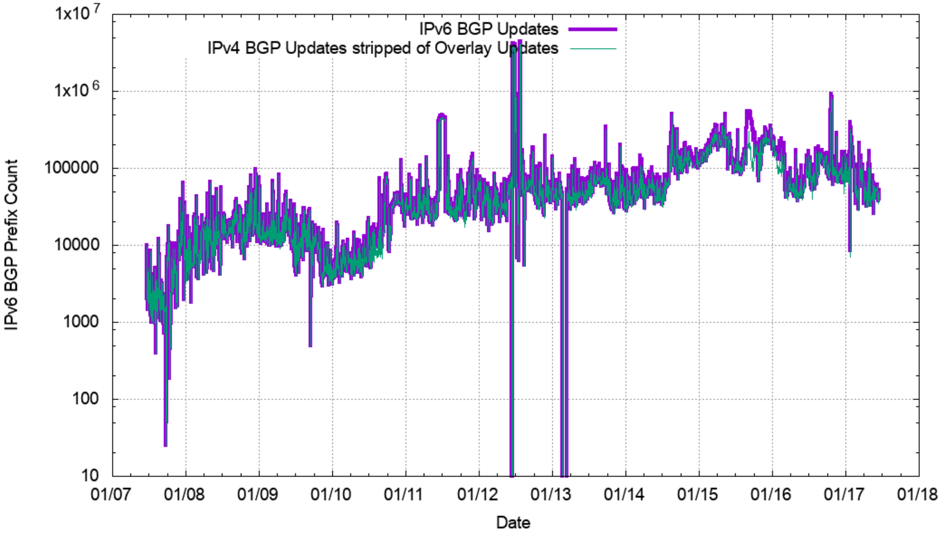


IPv6

Update Count Implications



IPv4



IPv6

What if…

- All Overlay more specific prefixes were removed from the routing table?
- Both IPv4 and IPv6 routing tables would drop in size by approximately 30%, as Overlay more specifics are now the predominate type of more specifics in the routing tables
- The rate of dynamic instability in BGP would not change by any significant amount, as overly more specifics are relatively stable prefixes

Summary of Findings

- More specifics add to both the size and the update load of BGP
- However BGP itself is both a reachability and a traffic engineering tool, and more specifics are often used to qualify reachability by traffic engineering. We have no other viable internet-wide traffic engineering tools, so this particular use of BGP really has no alternative
- Recent years have seen the decline of hole punching as more providers tend to treat their address blocks as integral units.
- Overlays are becoming more prevalent. While this has implications in terms of total table size it has no significant impact on BGP update rates.

Summary of Findings

- There is the question of total instability in IPv6 being far greater than IPv4, but this is not intrinsically an issue with more specifics, but a more general issue of BGP routing instability in IPv6
  - More investigation called for!

Thanks!