Characterizing IPv6 HSPs with respect to Activity

Pranau Kumar
pranau@gatech.edu

Amanda Hsu
ahsu67@gatech.edu

Abstract

IPv6-wide network scanning is an open problem for academics, requiring the need of various strategies to efficiently scan the address space. Recent work has developed several techniques to narrow down the address space, as well as to generate addresses of interest. However, these techniques have scope for improvement as they may not accurately model routing practices of network operators running IPv6 networks.

In this paper, we explore the possibility of using Hyper-Specific Prefixes (HSPs) as a proxy for activity in subnets to determine if they can be used to reliably narrow the search space. Further, we build on previous work to characterize the nature of IPv6 HSPs and their prevalence in the global routing tables. We find that HSPs are most likely to be /64 prefixes, and that they are more likely to belong to /32 prefixes. We also find that they are more prevalent in Asia and North America with the top 10 ASNs leaking more than 57% of HSPs. Along with this, we noticed that an ASN from Indonesia to be advertising an abnormally large number of ASNs. Ultimately, we correlate the prevalence of HSPs with a dataset of known active IPv6 addresses and find that HSPs are unlikely to be a reliable indicator of activity in a subnet.

1 Introduction

Network scans of the IPv4 address space are ubiquitous. There are several tools such as ZMap [25], Nmap [12], Censys [5], Shodan [18], etc. that can be used to scan the IPv4 address space. These tools are used by researchers, security professionals, and criminals alike to perform exhaustive scans of the IPv4 address space. However, when it comes to IPv6, these tools fall short. There are no tools that can perform exhaustive scans of the IPv6 space because of the address space size. The IPv6 address space consists of 2^{128} addresses which is intractable to scan by brute-force [21]. This is a problem especially for researchers conducting large-scale measurement studies of the Internet.

1.1 IPv6-wide Scanning

In the past few years, several researchers have come up with techniques to try and address the problem of conducting IPv6-wide scans more efficiently. These techniques vary from guessing the Internet address structure [27], to using target generation algorithms to generate candidate addresses [32], or taking advantage of IPv6 address assignment policies to predict active addresses [30].

While these techniques have proven very useful, they are not without their limitations. The rapidly changing landscape of the IPv6 address space makes it difficult to predict its structure [38]. The ground truth is nebulous. There exists tremendous scope for researchers to come up with better ways to describe and scan the IPv6 address space.

1.2 Hyper-Specific Prefixes

We temporarily move away from IPv6 scanning to focus on Hyper-Specific Prefixes (HSPs). HSPs are prefixes (or routes) that are more specific than /24 (IPv4) or /48 (IPv6). While they are supposed to be filtered out by routers, as recommended by several BGP best practice guidelines [6, 10, 11, 24, 36], they are still announced by many ASs and sometimes propagate into the global routing tables. The reason for this may be accidental or intentional. Accidental HSP advertisements are usually caused by misconfiguration while intentional advertisements may be done to perform traffic engineering, address reassignments or blackholing of addresses [35].

1.3 Contributions

We hypothesize that the advertisement of HSPs can reveal information about the internal routing structure of an AS. ISPs commonly hand out /56 prefixes to Customer Premises Equipment (CPE) which then hands out /64 prefixes on a local home network [33]. If these prefixes were then to be advertised externally, it might serve as an indicator of the density of active IP addresses in that region. This information

can then be fed to target generation algorithms to generate candidate addresses for scanning the space.

In this paper, we set out to test this hypothesis by trying to characterize the prevalence of HSPs with other known indicators of activity – including the characteristics of the larger parent prefix, the WHOIS record for the HSP, and the number of active IPv6 addresses in the larger parent prefix. We did this by trying to answer the following questions:

1. Do HSPs indicate that there is a WHOIS assignment record? If so, how likely is this?
2. How many HSPs exist from records in each Regional Internet Registry (RIR)? Is this phenomena more common in different parts of the world?
3. If we observe an HSP of a specific size, what is the size of the corresponding larger routed prefix and with what probability?
4. If we observe an HSP, what is the probability that there is at least 1 active IPv6 address in the larger routed prefix?
5. Do specific providers tend to route more HSPs? Do these providers have more or less observable IPv6 addresses in the hitlist than providers who do not route HSPs?
6. Are HSPs from specific providers typically of one size? Are the corresponding larger routes from a specific providers typically of one size?

This paper is organized as follows. In Section 2, we discuss background and related work. In Section 3, we describe our approach, assumptions, and limitations. In Section 4, we present our results characterizing the nature of IPv6 HSPs and their reliability as an indicator of subnet activity. In Section 6, we discuss future work.

2 Related Work

In this section, we talk about the prior work on IPv6 scanning as well as work done on HSPs. Work in the area of IPv6 scanning can be broadly categorized into three sub-areas:

i. scraping IPv6 target addresses from various sources on the internet,

ii. inferring target addresses by analyzing properties of routing and address allocation, and

iii. generating target addresses using algorithms or machine learning models.

Since our work in this paper primarily involves (ii) and to a lesser extent (i), we focus on these two sub-areas in this section.

Fiebig et al. [26] discovered that DNS servers' denial of existence (NXDOMAIN) responses for IPv6 address prefixes can be exploited to mine IPv6 addresses from DNS servers by recursively querying for prefixes. They were able to gather 5.8M unique IPv6 addresses this way. However, since not all DNS servers respond in a similar fashion, their work was limited to those servers they could query.

Gasser et al. [29] built a more comprehensive list using a combination of active and passive data sources. Passive data sources included service provider network taps that were available to them. They also actively collected data from crawling various sources such as Alexa Top 1 million domain list [1], Rapid7 [14, 15] and CAIDA [9] DNS datasets, and Top-level Domain (TLD) zone files. The list consisted of almost 150M IPv6 addresses. To test the effectiveness of their list, they modified ZMap [25] to scan IPv6 addresses and found a significant number of addresses responsive to ICMPv6 pings. Czyz et al. [23] performed a similar analysis using the DNS datasets while also looking at firewall policies on dual-stack devices. They found that very few of the bits in the IPv6 addresses used by routers and servers had non-zero bits and if they did they were likely to be concentrated to either the LSBs or MSBs.

RFC 7707 [30] describes several methods of reconnaissance of IPv6 networks such as leveraging stateless address autoconfiguration (SLAAC) policies, using LSBs of addresses, or looking for human-readable text in them. It also lists other methods used above such as analyzing DNS records and performing traceroutes to active IPv6 addresses.

Shodan, a search engine for Internet of Things (IoT) devices, joined NTP pools and probed IPv6 addresses of devices querying the NTP pools [31]. Though it is uncertain how many addresses they learned this way.

Plonka and Berger [34] developed a novel method to classify addresses temporally and spatially. The spatial classification provides a visual way to determine the density of a prefix and whether it is a suitable target for scanning.

When it comes to HSPs, Sediqi et al. [35] were the first to study and characterize the prevalence of HSPs in the wild. They found that HSPs seen on collectors has increased year-over-year and that they're used primarily intentionally but with a large number of HSPs also being accidentally advertised. While their work focuses on both IPv4 and IPv6 HSPs and tries to speculate on their intended use in general, our paper largely builds on their findings and tries to contextualize the utility of HSPs for IPv6 scanning.

Apart from the above, there have been a few other studies exploring HSPs in the form of blog posts. In 2014 and 2015, two blog posts by Aben and Petrie [19, 20] discussed running experiments to determine how far IPv4 HSPs are propagate across BGP routing tables. They found that HSPs propagate to at most 25% of BGP peers [37].

3 Methodology

In this section, we describe the approach used to collect and analyze the HSP data used in this paper. We begin by outlining the data sources used, followed by a description of the analysis. Later, we state the assumptions in our approach and finally, we discuss the limitations.

3.1 Data Sources

Route Data. We used the RIPE RIS RIB archives [16] to collect the route data. The RIB dumps are available in the MRT format [22] and can be parsed using a tool like bgpdump [7]. For this paper, we primarily used the RIB dumps from August 1, 2022, 16:00 UTC. The RIPE RIS project maintain multiple collectors in different geographic locations and peered to several BGP peers. Since each peer on BGP has a different view of the Internet, each collector is expected to have a different view of the Internet. We primarily used data from the 'RRC01' collector.

The data from the RRC01 collector was parsed using the bgpdump tool to obtain the prefix and its origin ASN which was then recorded in a text file. This data was then fed to a Python script for further processing described in the next subsection. We ended up with around 1 million prefixes using this method.

We also made use of CAIDA's BGPStream [8] and RIPE's Statistics website [17] to verify the visibility of HSPs across collectors and time. BGPStream provides a CLI tool and Python bindings to query BGP data from the RIS and RouteViews projects. RIPE's Statistics website provides an easy-to-use web UI to get information about specific prefixes.

WHOIS Data. We parsed WHOIS allocation records from ARIN [3], APNIC, RIPE, and AFRINIC bulk records to get all IPv6 WHOIS allocations as of August 1, 2022. We made use of offline WHOIS dumps available to Georgia Institute of Technology researchers as many Regional Internet Registries (RIRs) do not provide open access to WHOIS data. The total number of WHOIS records across all RIRs was 1.1 million.

Hitlist Data. IPv6 hitlists are lists of IPv6 addresses that are known to be active. We made use of the IPv6 hitlist maintained by Gasser et al. [28, 38] to test our hypothesis. The hitlist contains of three different datasets - responsive IPv6 addresses, aliased prefixes, and non-aliased prefixes. For this paper, we used the responsive IPv6 addresses dataset which consists of 6.8 million IPv6 addresses as of Nov 28, 2022.

3.2 Analysis

Our objective in this paper was to characterize the nature of IPv6 HSPs with respect to their larger routed prefixes, WHOIS records, and active IPv6 addresses, and ultimately to test whether they are suitable to gauge subnet activity. This subsection describes the various analyses performed.

Gathering HSPs. We first filtered all the HSPs from the route data that we parsed using bgpdump. We used this to build a Python dictionary mapping the HSP to its origin ASN.

Finding the larger parent prefix of HSPs. We made use of the Python library, pytricia [13], to generate a prefix tree of all non-HSP routes. This gave us a tree like structure with the ability to query arbitrary prefixes and get the longest matching prefix from the tree. The pytricia tree forms the basis of many of the analyses we performed in this paper.

The pytricia tree uses a dictionary-like interface in Python. To initialize the tree's nodes, we used a dictionary assignment notation with the key being the prefix and the value being the origin ASN.

```
tree = pytricia.PyTricia()
tree['2001:db8::/32'] = 1234
# 1234 is the origin ASN
```

While building the tree, we also took into consideration that some prefixes might have multiple ASNs due to multi-origin ASNs. In such cases, we used a list to store the ASNs.

To query the longest prefix match of an HSP, we used the 'get_key()' method of the pytricia tree.

```
tree.get_key('2001:db8:1::1/128')
# returns '2001:db8::/32'
```

Computing statistics about HSPs and their larger parent prefixes. Using the pytricia tree of the non-HSP routes, we computed the following statistics for each HSP:

1. CDFs of HSP prefix lengths and the prefix lengths of their larger parent prefixes.
2. Number of distinct prefix lengths per HSP prefix length.
3. CDF plots of the larger parent prefixes' prefix lengths for the top 5 HSP prefix lengths
4. Probabilities of larger parent prefixes' prefix lengths for the top 5 HSP prefix lengths
5. Top 10 ASNs with the most number of HSPs
6. Top HSP prefix length and larger parent prefix length for each ASN

Computing statistics about HSPs and their WHOIS records. We combined all the WHOIS records from each of the RIRs into a single file and constructed a pytricia tree like before consisting of the WHOIS allocations. As before, we used this tree to compute the following statistics for each HSP:

1. Maximum, minimum and median prefix lengths of the WHOIS allocations that contain the HSP
2. *Number of HSPs that have no corresponding WHOIS record* – Computed by checking for tree queries that return '::/0'
3. *Number of HSPs that have an exact corresponding WHOIS record* – Computed by checking for tree queries that return the HSP itself
4. CDF plots of WHOIS prefix lengths for the top 5 HSP prefix lengths
5. *Geographic region of each HSP* – Computed by checking the origin RIR of the WHOIS allocations

Computing statistics about HSPs and active IPv6 addresses. Using the pytricia tree of HSPs and their larger parent prefixes, we generated a mapping of every HSP and its larger parent prefix. Using this mapping, we generated a new pytricia tree containing just these prefixes. We then used this tree to check whether a given IPv6 address from the hitlist belonged to an HSP or its larger parent prefix. We computed the following statistics using this data:

1. Maximum, minimum, and average number of active IP addresses per larger prefix
2. Top 10 and bottom 10 larger prefixes with the most and least number of active IP addresses
3. *Number of responsive IP addresses for the most leaky HSP ASNs* – Computed by searching for all HSP larger parent prefixes belonging to the top 10 ASNs with the most number of HSPs. We then generated a tree using these prefixes and then matched the IP addresses in the hitlist against this tree.

Computing statistics about HSPs and their visibility across collectors. We initially used BGPStream tools (pybgpstream and bgpreader) to query the RIS collectors for the HSPs that we gathered from the previous steps. However, due to inefficiencies in the queries crafted by us, the queries took either too long to finish or did not return correct results.

We then resorted to using the RIPE Statistics website to directly check the visibility of the HSPs both across time and across collectors. Since this is a manual process, we randomly sampled around 30 HSPs from the list and checked them against the website.

On the website, we searched for an HSP and then navigated to the Routing tab to look at its Routing history. Using the filter options on the widget, gave us the number of peers the HSP was visible to and during what periods of time it was visible. Figure 1 shows an example of the RIPE Statistics website's UI for getting the routing history a prefix.

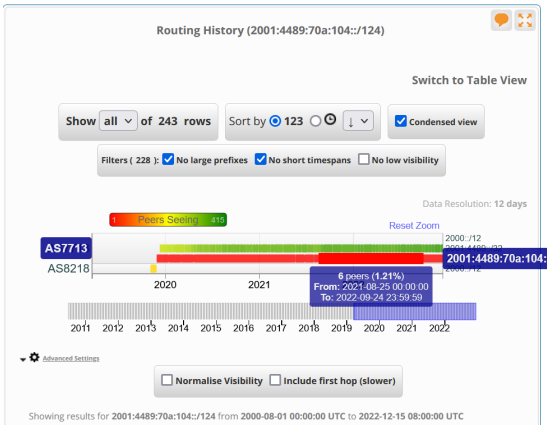

Figure 1: RIPE Statistics website's UI for getting the routing history of a prefix

IPv6 Network Scanner. We used an in-house network scanner to check the reachability of the IPv6 HSPs we obtained from our analysis. For this, we generated a list of IPv6 addresses from the HSPs by converting them to the first valid address in the prefix. We then used the scanner to launch ICMPv6 pings to these addresses and measured the hit rate of the addresses.

3.3 Assumptions

We made the following assumptions while performing our analysis:

1. We assumed that the HSPs we gathered from the route data were the only HSPs. While this is not necessarily true, we believe that the HSPs we gathered are a good representation to that enables us to draw some conclusions about their characteristics.
2. We assumed that HSPs are not used for malicious purposes.
3. We assumed that the HSPs leaked by most ASNs are useful to infer activity regardless of whether they propagate across the global routing tables or not. We believe that this is a reasonable assumption to make because our goal is to infer activity from the HSPs and not to actually exhaustively list all the HSPs.
4. We assumed that IPv6 hitlists are representative of the IPv6 address space. This might not be the case and is addressed further in Section 3.4.

3.4 Limitations

We were limited by a number of factors while performing our analysis:

1. We did not have access to the full WHOIS records for all the RIRs. We did not have access to any WHOIS bulk records from the Latin American registry (LACNIC). Further, the APNIC WHOIS records we had access to were not complete. This is because APNIC delegates allocation to local and national internet registries and thus the records it has are only for the allocations it makes directly. [2]

2. There is no ground truth for the density of allocations in the IPv6 address space. The IPv6 landscape is changing very rapidly. However, we still needed a metric to evaluate our hypothesis against which is why we used the IPv6 hitlist. As time goes on, the hitlist will become more representative of the density of allocations in the address space but as it stands it may not be the best metric to use.

3. The random sampling approach method use to check the visibility of HSPs across collectors and across time is not exhaustive and is prone to sampling bias. We believe that a better method is to exhaustively check every collector across multiple intervals and repeat our analysis for each of them. However, we the results we obtained are useful to qualitatively understand the characteristics of HSPs.

4 Results

In this section, we set out to answer the questions posed in Section 1 by presenting the results of our analyses in Section 3.2.

4.1 What do HSPs look like?

From our routing data, we filtered out 2173 Hyper-Specific Prefixes (HSPs). We find that HSPs are mostly /64s followed by /124s and /56s. Given what we know about HSPs so far, this is not surprising as /64s and /56s are the most commonly allocated prefixes. The presence of /124s suggest that these HSPs may be for blackholing [35].

The top 5 HSP prefixes are as follows (Table 1) and the counts for all prefix length is shown in Figure 2.

Prefix Length	Count
/64	1241
/124	167
/56	144
/127	131
/128	125

Table 1: Top 5 HSP prefix lengths

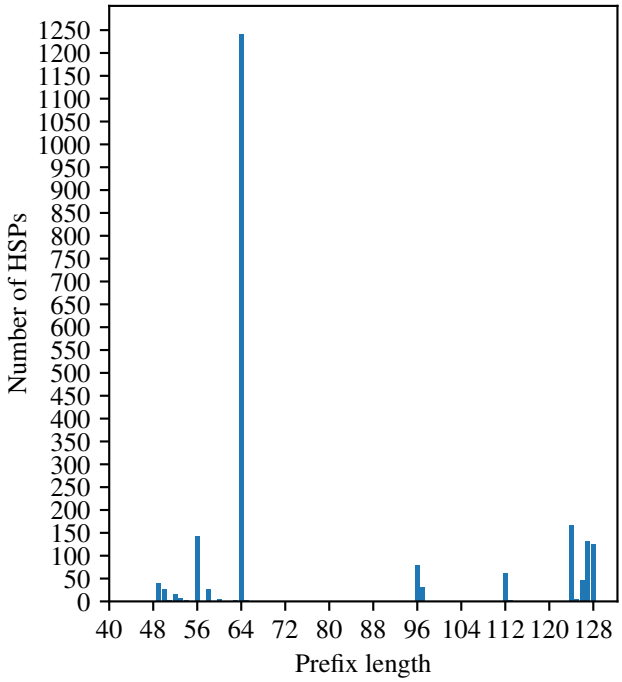

Figure 2: Distribution of HSP prefix lengths.

4.2 How do HSPs correlate to their larger parent prefixes?

We observe that most HSPs belong to a /32 prefix or a /48 prefix. /32s prefixes typically belong to large service providers or Content Delivery Networks (CDNs). /48 prefixes are typically the smallest route that should propagate onto global routing tables. This can be observed in Figure 3 where the CDF of HSP prefix length and larger parent prefixes' prefix length is shown. Similarly, in Figure 4, we see a similar trend for /64, /124, and /56 prefixes. However, we also note a significant number of /128 HSPs belong to /12 prefixes.

Overall, this means that an HSP of any size is likely to belong to a /32 prefix. We emphasize that this is different than the most common IPv6 route size of /48 found on global routing tables. This suggests,

1. HSPs are mostly leaked by large service providers or CDNs.

2. More importantly, HSPs are not indicative of internal routes used for general connectivity. We claim this because if HSPs were leakages of internal routes, we would expect them to aggregate to /48 or /56 prefixes. However, we see that HSPs most commonly aggregate to /32 prefixes. This suggests that HSPs are standalone prefixes that do not belong to a larger routing scheme dedicated to connectivity. What this implies is that the area in and around an HSP may not be a dense allocation area.

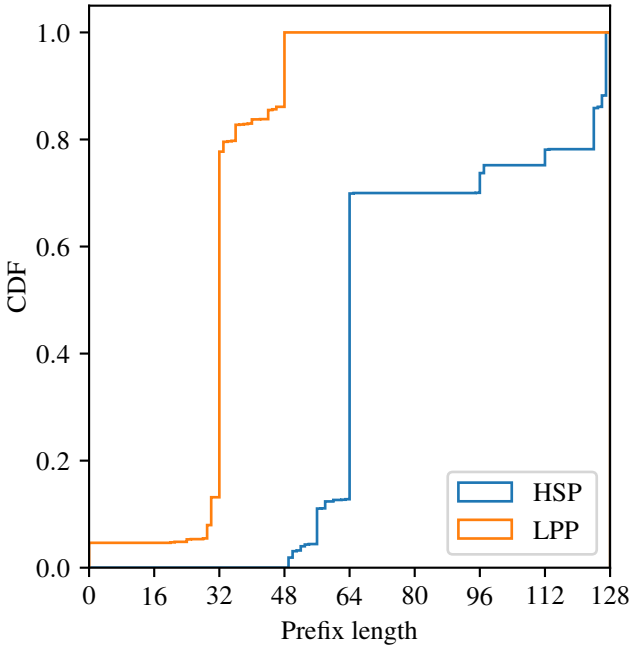

Figure 3: CDFs of HSP prefix length and larger parent prefixes' prefix length. Here, 'LPP' refers to the larger parent prefix.

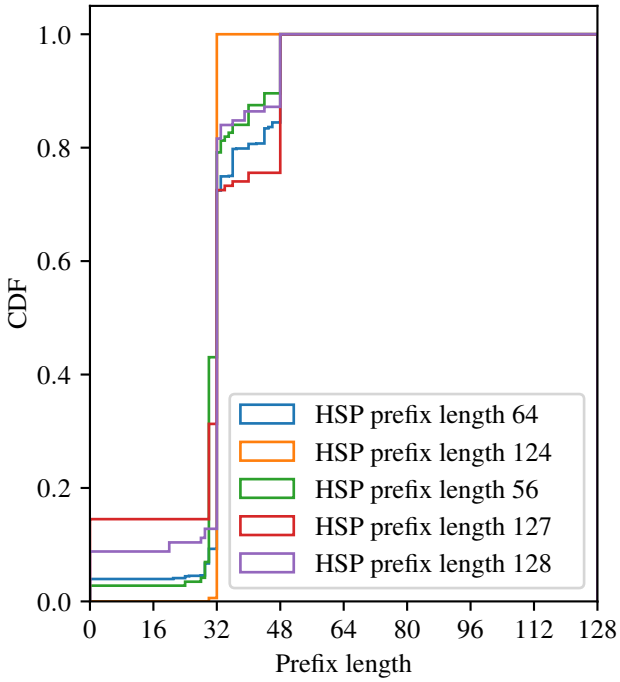

Figure 4: CDFs of HSP prefix length and larger parent prefixes' prefix length for the top 5 HSP prefix lengths.

4.3 How do HSPs compare to their WHOIS records?

We find that almost all HSPs can be traced back to a WHOIS allocation record. Once again, in Figure **??**, we notice that most HSPs belong to a /32 prefix WHOIS allocation. This further strengthens our claim from the previous section, 4.1.

Other noteworthy observations are that 4 HSPs do not have any WHOIS allocation record. This is likely indicative of a mistaken HSP advertisement or a mistake in the WHOIS records. We also find 27 instances where the HSP is equal to the WHOIS allocation. This is likely indicative of hosts that wish to be directly reachable using that prefix and is likely a legitimate record.

Location	Count
North America	783
Africa	41
Asia Pacific	932
Europe & Middle East	416

Table 2: Geographic regions of HSPs

4.4 Who leaks the most HSPs?

We observe that the top 5 ASNs leaking the most HSPs are as follows:

ASN	HSP Count
7713	552
28885	143
8151	112
262191	83
52468	82

Table 3: Top 5 ASNs leaking the most HSPs

ASN 7713. We immediately notice that ASN 7713 is a weird outlier among these ASNs. ASN 7713 belongs to PT Telekomunikasi Indonesia [4]. This is the largest telecom company in Indonesia. They own around 1000 IPv6 prefixes of which around 500 are HSPs. The most common HSPs advertised by them are /64s and /124s. The astute reader may notice here that this ASN singularly biases the HSP count for the Asia Pacific region.

When we first discovered this anomaly, we were curious as to why this was the case. We hypothesized that it was a

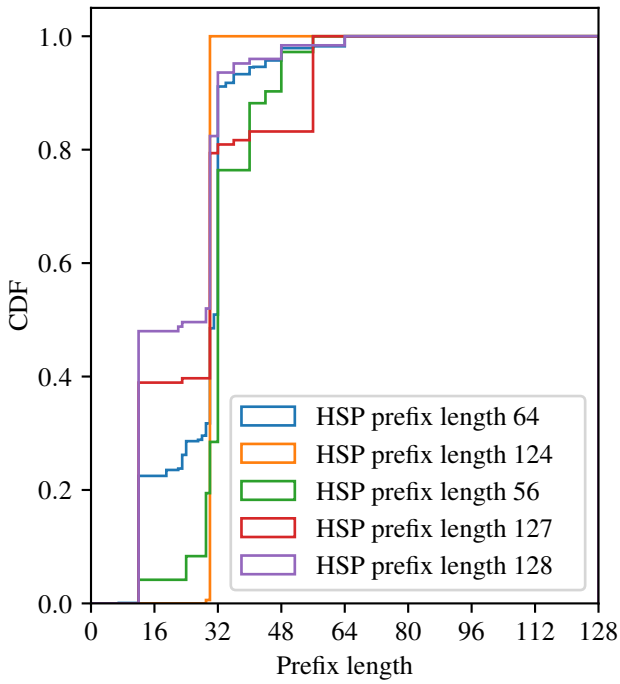

Figure 5: CDFs of WHOIS prefix lengths for top 5 HSP prefix lengths.

mistake made by the ISP and we just so happened to collect it in our dataset. However, further investigation leads us to believe that ASN has continued to advertise these HSPs for a considerable amount of time. We notice this today and going back as far as the beginning of 2022 or earlier. We can only speculate as to why so many HSPs are being advertised by this ASN.

4.5 How do HSPs correlate to IPv6 activity?

When looking for known active IPv6 addresses in HSPs and their larger parent prefixes, we find that the most leaky ASNs do not contain the most number of active IPv6 addresses. This is shown in Table 4 below. Most of these ASNs belong to CDNs which is not surprising.

In general, we find that HSPs larger prefixes' are likely to contain at least one active IPv6 address and an average of 1700 addresses. The IPv6 hitlist we used contained a total of 6.8 million addresses.

Are there more active IP addresses from providers who leak the most HSPs? The answer to this question is surprisingly no. We observe that the top 5 ASNs leaking the most HSPs contain an average or less than average number of active IPv6 addresses. If we can accept that the IPv6 hitlist is an acceptable representation of the density of address allocations, our claim that HSPs are not indicative of a dense allocation area is further strengthened. Table 5 shows the top 5 ASNs that leak the most HSPs and the number of active IPv6 addresses they contain in their HSP's larger parent prefixes.

Prefix	No. Responsive IPs	ASN
2001:558::/29	109,115	7922
2600:1400::/24	93,373	20940
2600:1400::/24	61,229	20940, 34164
2001:470::/32	37,804	6939
2001:1900::/32	25,957	3356

Table 4: Prefixes with the most number of active IPv6 addresses

ASN	No. Responsive IPs
7713	1,425
28885	569
8151	1,892
262191	30
52468	268

Table 5: Top 5 ASNs that leak the most HSPs and the number of active IPv6 addresses they contain

4.6 Activity of HSPs themselves

We tried to find out if the HSPs themselves are active by trying to ping the first IP address in the prefix. We made use of a network scanner that sent ICMPv6 pings to achieve this. We find that around 60% of these addresses are responsive to ICMPv6 pings. We hypothesize that this considerable number is due to the fact that we're directly pinging Internet infrastructure devices such as routers or firewalls that are generally configured to respond to ICMPv6 pings and also present on the first address of subnets. This is also an opportunity for future work to explore what kind of devices are responding to these pings.

4.7 Prevalence of HSPs

We find that most advertised HSPs tested on the RIPE Statistics website are visible across time. A lot of the advertisements have been visible since at least the beginning of 2022. However, we discovered that every HSP does not widely propagate across global routing tables. There were very few HSPs

seen by more than 5% of all RIS' collector peers. And even fewer HSPs seen by more than 10% of route collector peers.

5 Conclusion

In this paper, we investigated the nature of Hyper-Specific prefixes (HSPs) and whether they can be used to reliably narrow the search space for IPv6-wide network scanning. We found that HSPs are most likely to belong to /32 prefixes and are more prevalent in Asia and North America. We also found that there is very little indication that HSPs that are advertised by ASNs are used for general connectivity within the ASN's network. This is bolstered by the fact that leaky ASNs are less likely to have active IPv6 addresses in them. These findings suggest that HSPs are unlikely to be a reliable proxy for subnet activity. However, it is hard to conclusively prove that it is the case.

6 Future Work

There is a lot of scope for future work in this area. We would like to explore the following avenues of enquiry for future work:

1. *Quantifying the density of active IPv6 addresses in HSPs' larger prefixes* – We would like to explore the probability of finding an active IPv6 address in a larger parent prefix of a HSP when it is divided into subnets of the same size as the HSP. This would help us determine if there are any patterns in the distribution of active IPv6 addresses in HSPs' larger prefixes.

2. *Quantifying the prevalence of HSPs over time and across collectors* – In this paper, we showed qualitatively by randomly sampling HSPs that they don't propagate across many collectors and that they're likely to be advertised long periods of time. However, we would like to do this exhaustively for multiple collectors and over multiple time intervals.

3. *Exploring the kind of devices present in HSPs* – We found that 60% of HSPs respond to ICMPv6 pings on the first address of their prefix. We would like to explore the kind of devices that are present in these addresses and determine if any inferences can be derived from the findings.

Availability

In the interest of transparency and reproducibility, we make all our code available on Github - https://github.com/pranau97/ipv6-hsps

References

[1] Alexa top 1,000,000 sites | no longer available. https://www.alexa.com/topsites.

[2] Apnic – understanding address management hierarchy. https://www.apnic.net/manage-ip/manage-resources/address-management-objectives-2/address-management-objectives/.

[3] Arin whois records. https://ftp.arin.net/pub/stats/arin/.

[4] Asn information for asn 7713. https://www.whatismyip.com/asn/7713/.

[5] Attack surface management and data solutions | censys. https://censys.io/.

[6] Bgp prefix filtering. https://www.noction.com/knowledge-base/bgp-prefix-filtering/.

[7] bgpdump - utility and c library for parsing mrt files. https://github.com/RIPE-NCC/bgpdump.

[8] Bgpstream. https://bgpstream.caida.org/.

[9] Caida ipv6 dns names dataset. https://www.caida.org/catalog/datasets/ipv6_dnsnames_dataset/.

[10] Filtering small prefixes - bgp prefix filtering guide. https://bgpfilterguide.nlnog.net/guides/small_prefixes/.

[11] Ipv6 bgp filter recommendations. https://www.space.net/~gert/RIPE/ipv6-filters.html.

[12] Nmap: the network mapper - free security scanner. https://nmap.org/.

[13] Pytricia - a library for fast ip address lookup in python. https://github.com/jsommers/pytricia.

[14] Rapid7 fdns. https://opendata.rapid7.com/sonar.fdns_v2/.

[15] Rapid7 rdns. https://opendata.rapid7.com/sonar.rdns_v2/.

[16] Ripe bgp ribs raw data. https://ris.ripe.net/docs/20_raw_data_mrt.html.

[17] Ripestat. https://stat.ripe.net/.

[18] Shodan search engine. https://www.shodan.io/.

[19] Emile Aben. Has the routability of longer-than-/24 prefixes changed, 2015.

[20] Emile Aben and Colin Petrie. Propagation of longer-than-/24 ipv4 prefixes. ripe labs, 2014.

[21] Steven Michael Bellovin, Bill Cheswick, and Angelos D Keromytis. Worm propagation strategies in an ipv6 internet. 2006.

[22] Larry Blunk, M Karir, and C Labovitz. Multi-threaded routing toolkit (mrt) routing information export format. Technical report, 2011.

[23] Jakub Czyz, Matthew Luckie, Mark Allman, Michael Bailey, et al. Don't forget to lock the back door! a characterization of ipv6 network security policy. In *Network and Distributed Systems Security (NDSS)*, 2016.

[24] J Durand, I Pepeljnak, and G Doering. Bgp operations and security. Technical report, 2015.

[25] Zakir Durumeric, Eric Wustrow, and J Alex Halderman. {ZMap}: Fast internet-wide scanning and its security applications. In *22nd USENIX Security Symposium (USENIX Security 13)*, pages 605–620, 2013.

[26] Tobias Fiebig, Kevin Borgolte, Shuang Hao, Christopher Kruegel, and Giovanni Vigna. Something from nothing (there): collecting global ipv6 datasets from dns. In *International Conference on Passive and Active Network Measurement*, pages 30–43. Springer, 2017.

[27] Pawel Foremski, David Plonka, and Arthur Berger. Entropy/ip: Uncovering structure in ipv6 addresses. In *Proceedings of the 2016 Internet Measurement Conference*, pages 167–181, 2016.

[28] Oliver Gasser, Quirin Scheitle, Pawel Foremski, Qasim Lone, Maciej Korczynski, Stephen D. Strowes, Luuk Hendriks, and Georg Carle. Clusters in the expanse: Understanding and unbiasing ipv6 hitlists. In *Proceedings of the 2018 Internet Measurement Conference*, New York, NY, USA, 2018. ACM.

[29] Oliver Gasser, Quirin Scheitle, Sebastian Gebhard, and Georg Carle. Scanning the ipv6 internet: towards a comprehensive hitlist. *arXiv preprint arXiv:1607.05179*, 2016.

[30] Fernando Gont and Tim Chown. Network reconnaissance in ipv6 networks. Technical report, 2016.

[31] Dan Goodin. Using ipv6 with linux? you've likely been visited by shodan and other scanners, 2016.

[32] Austin Murdock, Frank Li, Paul Bramsen, Zakir Durumeric, and Vern Paxson. Target generation for internet-wide ipv6 scanning. In *Proceedings of the 2017 Internet Measurement Conference*, pages 242–253, 2017.

[33] Ramakrishna Padmanabhan, John P Rula, Philipp Richter, Stephen D Strowes, and Alberto Dainotti. Dynamips: analyzing address assignment practices in ipv4 and ipv6. In *Proceedings of the 16th International Conference on emerging Networking EXperiments and Technologies*, pages 55–70, 2020.

[34] David Plonka and Arthur Berger. Temporal and spatial classification of active ipv6 addresses. In *Proceedings of the 2015 Internet Measurement Conference*, pages 509–522, 2015.

[35] Khwaja Zubair Sediqi, Lars Prehn, and Oliver Gasser. Hyper-specific prefixes: gotta enjoy the little things in interdomain routing. *ACM SIGCOMM Computer Communication Review*, 52(2):20–34, 2022.

[36] Philip Smith, Rob Evans, and Mike Hughes. Ripe routing working group recommendations on route aggregation. *Document ripe-399, RIPE*, 2006.

[37] Stephen Strowes and Colin Petrie. Bgp even-more specifics in 2017, 2017.

[38] Johannes Zirngibl, Lion Steger, Patrick Sattler, Oliver Gasser, and Georg Carle. Rusty clusters? dusting an ipv6 research foundation. In *Proceedings of the 2022 Internet Measurement Conference*, New York, NY, USA, 2022. ACM.