Scanning the Internet-wide Network Periphery: Towards IPv6 Vulnerability Discovery

Abstract

IPv6 vulnerabilities pose a substantial risk to Internet security at a global level. Fast scanning the Internet-wide network periphery can promptly discover those vulnerabilities, effectively mitigating the above threats. However, existing solutions in this field exhibit limitations, either in terms of inadequate scanning speed or low hit rates. The fundamental reason is that performing a brute-force scan across the extensive IPv6 address space (2^{128}) is unfeasible. Thus, a methodology is imperative to discover the maximum number of IPv6 periphery addresses with limited resources. Accordingly, we introduce 6Seeks, an innovative asynchronous IPv6 scanner capable of exploring numerous IPv6 prefixes simultaneously for efficient periphery discovery. Considering the uneven distribution of IPv6 periphery devices, 6Seeks incrementally adjusts the search directions, prioritizing the scanning of the address space with a higher concentration of IPv6 periphery, i.e., interaction-based scanning. Furthermore, we introduce a carefully crafted scanning architecture optimized for performance, enabling 6Seeks to efficiently scan at the highest probing rate theoretically without requiring excessive computation. Real-world experiments have shown that 6Seeks can discover over 20 million IPv6 periphery addresses on billion-scale scanning, representing an average 25× improvement over state-of-the-art methods with less time consumption. Furthermore, we utilized 6Seeks to uncover two high-risk vulnerabilities, namely IPv6 backdoor and reflection amplification, and conducted a thorough investigation into their global impacts.

1 Introduction

Since 2012, the percentage of Google users accessing services via IPv6 has skyrocketed from below 1% to reach 40% [1]. At the time of writing this paper, a considerable proportion of the top websites (21.1%) have already adopted IPv6 [2]. With the increasing accessibility of IPv6-capable devices, vulnerabilities in IPv6 pose an ever-growing threat to the Internet. In order to promptly discover these IPv6 vulnerabilities for effective mitigation of the above threats, fast scanning the network periphery on a large scale represents a crucial measure.

Firstly, the IPv6 network periphery, i.e., the last-hop routing device connecting the end-hosts or itself in the IPv6 Internet [3–6], is a critical component of the IPv6 network architecture. It not only enables direct Internet communication for end-users, thereby bearing the brunt of security threats from the open Internet [7, 8], but also hosts various network services and customer assets [4]. Secondly, understanding IPv6 periphery deployment can provide valuable insights into the security status (e.g., Common Vulnerabilities and Exposures) of IPv6 assets [11, 12], thereby enabling the implementation of more effective security measures. Moreover, fast collection of IPv6 periphery addresses can facilitate the generation of a comprehensive network topology mapping of the IPv6 infrastructure and promote Internet measurements for security [3, 9, 10]. Hence, we are highly motivated to scan the Internet-wide network periphery.

However, to solve this challenge is not trivial because the extensive address space of IPv6 (with a total of 2^{128} unique addresses) renders a thorough survey of all potential targets practically infeasible, as a brute-force scan of the entire address space would optimistically require 10 billion years. Undeniably, there has been remarkable progress made in this area: Xmap [4], as a state-of-the-art work, has successfully reduced the exploration scope by leveraging the observation that, an IPv6 address with a randomly generated Interface Identifier (lower 64 bits of an IPv6 address) is unlikely to be active except for IPv6 aliases. In such cases, a packet sent to such an address would be responded to by an ICMPv6 error message from the network periphery [13]. Intuitively, the entire scanning space has been narrowed down from 2^{128} to 2^{64} as only one probe is needed for each `/64` prefix. Notwithstanding, state-of-the-art (SOTA) techniques only enable probing of a relatively limited block of the IPv6 address space, typically a `/32` prefix. Thus, conducting Internet-wide scanning of IPv6 periphery in practice remains an unresolved issue.

To this end, the paper proposes 6Seeks, an asynchronous

IPv6 scanner designed for efficient Internet-wide periphery discovery. Its aim is to effectively scan IPv6 periphery addresses within a restricted probing budget. As a dynamic scanning methodology, 6Seeks considers the fundamental insight that expensive probes should prioritize areas of the IPv6 address space with higher rewards, given the practical limitations of scanning every address or `/64` network prefixes in IPv6. Specifically, 6Seeks involves partitioning the entire address space into a set of subprefixes for scanning tasks, which are then systematically scanned. The allocation of probing budgets to the respective subprefixes is based on their past performance in prior scans, a.k.a., interaction-based scanning. In a sense, the subprefixes that have demonstrated superior performance in previous scans are allocated higher budgets, thereby increasing the likelihood of successful IPv6 periphery discovery in subsequent rounds, drawing inspiration from the concept of reinforcement learning. To implement this dynamic methodology in real-world fast IPv6 scanning, a range of novel system-level designs has been proposed to ensure hit rate and scanning efficiency, including budget scheduling and control state mechanisms, which are explained in detail in § 3.

To demonstrate the capability of our scanner in discovering IPv6 vulnerabilities, we provide a detailed analysis of two high-risk vulnerabilities that were identified during the global network periphery scan conducted by 6Seeks, and evaluate the impact caused by these vulnerabilities:

IPv6 backdoor. While address autoconfiguration simplifies IPv6 deployment [14, 15], default practices can result in security issues. For instance, many ISPs now offer IPv6 prefix delegation by default [16], enabling endpoints to obtain global-routing IPv6 addresses for end-to-end Internet communication without manual assistance. However, it can expose vulnerable services on internal networks without users' knowledge. With an application layer scanning tool and a set of heuristic rules, we identified approximately 126K instances with IPv6 backdoors in 2745 autonomous systems during the global scan (see § 5). The results clearly demonstrate the severity of these vulnerabilities, as they enable hackers to easily compromise vulnerable hosts and implant malware, resulting in data breaches and system damage.

IPv6 reflection amplification. These reflection amplification vulnerabilities have received limited research attention due to the scarcity of IPv6 scanning technology. Recent global scanning conducted by 6Seeks within dozens of hours revealed that approximately 130K DNS services, 200K SNMP services, and 15K NTP services were identified as traffic amplifiers, while roughly 2 million IPv6 network periphery devices were found to respond to address spoofing. Moreover, in over 1,000 ASes, IPv6 middleboxes for Internet censorship purposes can also exhibit vulnerabilities to reflection amplification. These findings highlight the vulnerability of IPv6 networks to amplification attacks and the increasing threats associated with IPv6 address spoofing, demanding significant attention as the development of IPv6 scanning progresses.

The contribution of the paper could be summarized as follows:

- The paper develops a novel asynchronous scanning tool, 6Seeks, which pioneers the application of interaction-based scanning for efficient Internet-wide IPv6 periphery discovery. Real-world tests at a large scale demonstrate that the proposed dynamic algorithm and optimization designs enable 6Seeks to discover 25× more IPv6 periphery devices on average (removing aliases) while ensuring fast scanning speeds.

- A comprehensive analysis of the global IPv6 periphery is presented, based on the discovery of over 20 million IPv6 periphery devices through an Internet-wide scan using 6Seeks. These findings have significant potential for promoting IPv6 security and network measurement.

- This paper examines the backdoor vulnerability in IPv6, emphasizing the security issues associated with internal services in customer networks being exposed to the open Internet due to default practices in IPv6 address autoconfiguration. Using Internet-wide scans conducted by 6Seeks, we find that at least 126K hosts worldwide are at risk of these attacks, and some of them have Common Vulnerabilities and Exposures (CVEs), increasing their susceptibility to exploitation by attackers.

- This paper uncovers the increasing threats associated with reflection amplification in IPv6. Measurements from 6Seeks reveal a significantly higher number of IPv6 devices (including IPv6 middleboxes for Internet censorship) that can be exploited as reflectors than previously reported, thereby facilitating attackers in launching large-scale distributed denial-of-service (DDoS) attacks in IPv6 networks.

The remainder of the paper is organized as follows: § 2 provides a summary of the background and models related to the IPv6 network periphery. § 3 presents the system design of the 6Seeks scanner, while § 4 compares its performance with baselines. § 5 and § 6 systematically measure two newly-discovered IPv6 vulnerabilities. § 7 discusses the ethical considerations during internet scanning and network measurements. § 8 proposes countermeasures to address the growing threats posed by those IPv6 vulnerabilities. § 9 introduces related works, while § 10 concludes this paper.

2 Preliminaries

2.1 Background

IPv6 addressing system. The addressing system employed in IPv6 diverges from its predecessor, IPv4, and is characterized

by the adoption of a 128-bit address space, facilitating the allocation of a unique identifier, an IPv6 global unicast address, to Internet-connected devices [17]. The IPv6 addressing architecture encompasses a hierarchical structure consisting of a global routing prefix, a subnet identifier, and an interface identifier. The upper 64 bits of a global unicast address are composed of the global routing prefix and subnet identifier for routing networks, whereas the interface identifier (IID) is assigned to the lower 64 bits for identifying the particular device within a network.

IPv6 address pattern. Based on the interface identifiers (IIDs) [18], IPv6 address patterns that can be categorized as illustrated in Tab. 1.

Table 1: Categories of IPv6 Address Patterns

Address Patterns	IID Examples	Comments
EUI64[a]	0250:56ff:fe89:49be	embed MAC address 00:50:56:89:49:be and then flip 7_{th} bit
Embed-IPv4	0012:0122:0126:0072	embed IPv4 address 12.122.126.72
Low-byte	0000:0000:0000:f1b7	all zeros except the lower bytes
Pattern-bytes	0021:2222:0001:0001	more than two bytes of zeros
Randomized[b]	10de:51e8:eb66:7583	pseudorandom

[a]: The first three bytes represent Organizationally Unique Identifier [15].
[b]: Privacy Extensions for Stateless Address Autoconfiguration [19].

IPv6 address autoconfiguration. Both Dynamic Host Configuration Protocol for IPv6 (DHCPv6) [14] and Stateless Address Autoconfiguration (SLAAC) [15] enable devices on a network to automatically configure IPv6 addresses. A DHCPv6 server can also offer more extensive options, such as configuring DNS servers and prefix delegation. Therefore, DHCPv6 is preferred in large-scale networks, such as Service Provider Networks, due to its centralized management capabilities, while small-scale home networks take SLAAC due to its efficiency and simplicity. Unlike IPv4 addresses, IPv6 devices can obtain at least one `/64` IPv6 prefix in practice for Internet communications through both DHCPv6 and SLAAC, and can fully customize the interface identifiers (IIDs) of their IPv6 addresses, even if the DHCPv6 servers provide recommended addresses.

Prefix delegation. It is typically used in conjunction with DHCPv6 to enable a network router to assign a specific IPv6 prefix to other devices [16]. For instance, the customer router can request a delegated prefix from the ISPs' DHCPv6 servers and assign a distinct `/64` prefix from its delegated prefix to each of its LAN interfaces [20] and connected end-hosts for configuring the IPv6 global unicast addresses. However, improper use of prefix delegation also presents security risks by potentially exposing vulnerable private services on internal networks to the open Internet, as demonstrated in § 5.

Host model. In computer networking, a host model represents a design choice for the TCP/IP stack in a network operating system [21]. Briefly, devices that function as routers commonly utilize the *weak host model* to enhance network connectivity, allowing their systems to send or receive packets through a given interface regardless of whether the destination/source in the packet is assigned to that interface. In the *strong host model*, devices must strictly adhere to the correspondence between IP addresses and interfaces.

2.2 IPv6 Periphery Model

IPv6 network periphery, the opposite to the backbone or core network, serves as the last-hop routed device that connect either end-hosts or itself [3, 4], such as a customer premise router linked to the Service Provider networks and a smartphone attached to the mobile networks, respectively. These devices not only function as IPv6 routers in the entire Internet topology, participating in packet forwarding and route operations, but also serve as gateway devices to manage network prefixes. It enables end-hosts' Internet access and ensures the security of both the devices themselves and downstream devices. Consequently, IPv6 end-hosts benefit from end-to-end communication without complex address translation schemes. However, the IPv6 periphery is also exposed to inherent security threats and network attacks present in the Internet. Therefore, measuring the IPv6 periphery is crucial for understanding the status of these increasingly rampant cybercriminals targeting IPv6 networks and promoting risk mitigation.

Unlike servers and client devices that implement a strong host model, the IPv6 network periphery commonly implements a weak host model to enable route capacity [20, 22, 23]. According to application scenarios, IPv6 network periphery involves the two following topology models, as shown in Fig. 1:

Customer Premise Edge (CPE) model. CPE routers [24], as the last-hop routed device connecting end-hosts, typically feature two network interfaces: 1) WAN interfaces establish point-to-point connections with ISP routers and configure their WAN IPv6 addresses using network prefixes assigned by ISPs. 2) LAN interfaces function as internal network interfaces responsible for connecting the end-hosts. When both the CPE router and ISP support prefix delegation [16], each LAN interface and end-host would acquire at least one distinct `/64` prefix from the delegated prefixes for end-to-end Internet communications [20].

User Equipment (UE) model. The User Equipments (UE) refers to any device that functions as an IPv6 router, connecting the cellular network to a LAN [22]. A common scenario involves a smartphone connecting to a base station to access the mobile network. In such cases, the smartphone transitions from host-only mode to router-and-host mode [23]. The base station delegates a `/64` prefix to the UE, enabling it to manage the LAN that comprises only the smartphone itself.

3 The Scanner: 6Seeks

First, we introduce the periphery discovery strategy employed in 6Seeks. Next, we provide an overview of the 6Seeks scanner system. This is followed by an explanation of its algorithm

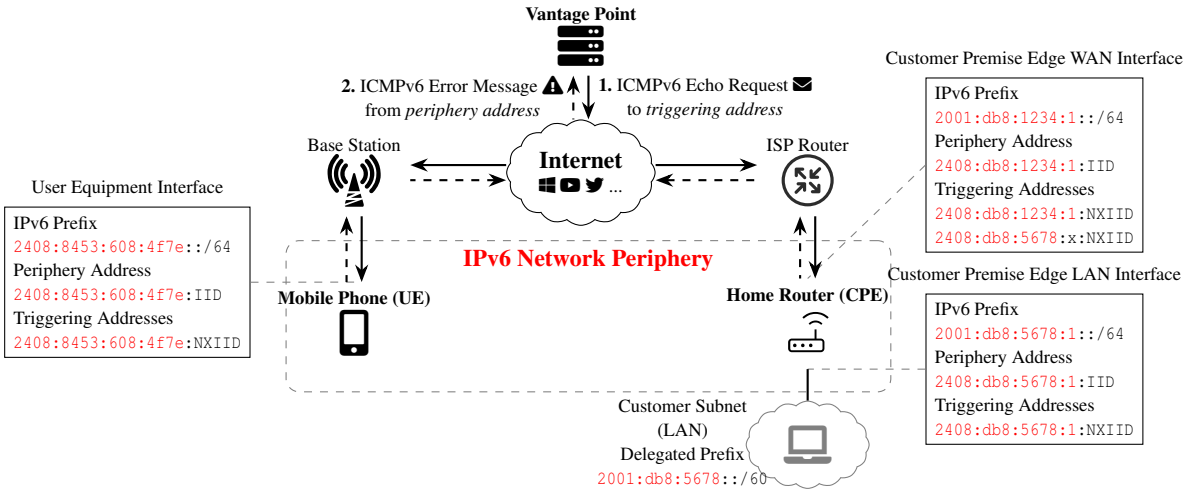


Figure 1: A mixed IPv6 periphery model involving Customer Premise Edge and User Equipment.

for budget allocation, which is designed to address the challenge of hit rate. Additionally, we discuss the mechanism for control state aimed at ensuring scanning efficiency.

3.1 Periphery Discovery Strategy

6Seeks employs a validated single-packet approach for IPv6 periphery discovery [4]. As a fundamental observation in computer networking, when a packet cannot reach the destination address (which might not exist), the router should generate an ICMP(v6) Destination Unreachable message [13]. Therefore, by using a probing packet with a carefully crafted destination address (namely triggering addresses as shown in Fig. 1) towards the prefixes assigned to an interface or delegated to a CPE, the IPv6 periphery devices (kindly recall that all the periphery devices function as an IPv6 router as mentioned in §2.2) will disclose their IPv6 periphery address in their responses.

For easy understanding, we will separately illustrate the exposure of the IPv6 address of the CPE LAN interface (see Fig. 1), as discovering other interface addresses can follow easily. Mostly, probes sent to the prefix delegated to a CPE would typically receive direct responses from the CPE WAN interfaces. However, since an IPv6 CPE must assign a separate `/64` from its delegated prefix(es) to each of its LAN interfaces [20], these probes might also receive responses from CPE LAN interfaces. In IPv4, responses from LAN interfaces would be discarded since the addresses of LAN interfaces are private IPv4 addresses provided by NAT. However, in IPv6, all LAN interfaces can obtain a public-routed prefix (from the CPE delegated prefix) to enable a packet directly transmitted from the internal network to the outside. Additionally, kindly recall that all the periphery devices are implemented with a weak host model [21], which means that responses from LAN interfaces could be sent back to our vantage point through the CPE WAN interface!

Overall, triggering address generation is crucial for IPv6 periphery discovery. Specifically, 6Seeks employed a randomized 64-bit Interface Identifier (IID) for the triggering address, making it highly unlikely to encounter an existing address within the large 2^{64} IID space. As a result, the scanning space for IPv6 periphery discovery is reduced from 2^{128} to 2^{64}, named "/64 prefix space" in the rest of this paper, since only one probe is necessary for each `/64` prefix. Nevertheless, the exploration of such vast "/64 prefix space" remains an open matter due to millions of years required for a brute-force scan, and 6Seeks is thus dedicated to addressing this challenge next.

3.2 System Overview

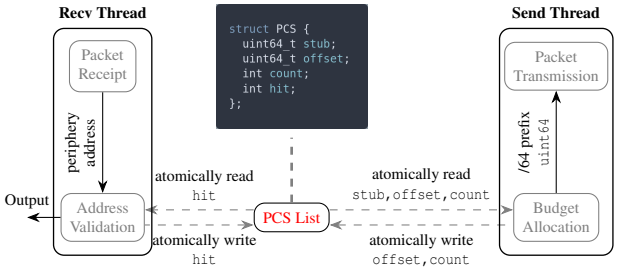


Figure 2: The architecture of 6Seeks.

In 6Seeks, the global-scale IPv6 blocks provided by users are divided into a series of orthogonal /32 prefixes (namely "subprefixes" in the paper), following the related works. These subprefixes are then utilized to initialize the corresponding Prefix Control States (PCSs). In the **Send** thread, the budget allocation module generates `/64` prefixes based on the PCS list, which are subsequently utilized for packet generation and transmission. In the **Recv** thread, the packet receipt module captures ICMPv6 unreachable error messages from periphery devices, while the address validation module filters out

duplicate IPv6 periphery addresses. Both the **Send** and **Recv** threads update the shared PCS list using atomic operations for read and write operations.

3.3 Dynamic Budget Allocation

Following the preprocessing of input data, the critical issue is to effectively allocate probes in order to attain a high hit rate while remaining within the allotted budget. Assuming, without loss of generality, that the input for 6Seeks consists of n subprefixes, each of which has an allocated budget denoted by $\{x_1,\ldots,x_n\}$ respectively. Formally, the problem can be restated as a combinatorial optimization problem [25], whose model $P=(X,f,g)$ comprises of:

- A finite set of candidate solutions X, that are defined over a series of vectors of decision variables $x \in X$. Specifically, the i_{th} subprefix of the solution x is allocated with a total of x_i scanning budgets for a particular solution, represented by $x=\{x_1,\ldots,x_n\}$.
- The object function f, that produces the number of discovered IPv6 periphery addresses $f(x)$, given a solution vector $x \in X$.
- The constraint function g, that presents the limitation of scanning budget b in this task, i.e., $g(x): x_i+\cdots+x_n \leq b$.

Accordingly, the goal of discovering IPv6 periphery addresses can be formulated as follows: maximize $f(x)$, subject to $g(x)$. And, it can be concluded that a solution $x^* \in X$ is the global optimum if and only if the following condition holds: $\forall x \in X, f(x^*) \geq f(x)$.

Obviously, there are various allocation strategies for the probes sent to different regions. As a reminder, the randomization scanning approach, such as Xmap [4], is known to produce a solution denoted as $x^s=\{x_1^s,\ldots,x_n^s\}$, where $x_1^s \approx \cdots \approx x_n^s$, which is unlikely to be the global optimum due to the uneven distribution of periphery addresses in the IPv6 address space. A more optimal approach would entail the budget allocation to subprefixes based on their respective global rewards. However, the lack of a priori knowledge regarding the distribution of periphery devices in the IPv6 network makes it impossible to know the global average rewards among the subprefixes before conducting the scanning.

To resolve it, 6Seeks employs a dynamic search strategy that utilizes feedback from received IPv6 periphery addresses so far to adjust the following budget allocation across all subprefixes, utilizing the concept of reinforcement learning. The fundamental consideration of the 6Seeks budget allocation algorithm (as shown) is to prioritize subprefixes with high rewards so far, while allocating budgets to subprefixes with low rewards in a more cautious manner.

Feedback-based algorithms commonly face the challenge of overfitting, e.g., exclusive exploitation of currently high-reward subprefixes without sufficient exploration of currently low-reward subprefixes may result in overlooking the potential IPv6 periphery addresses. To this end, we propose a solution whereby the budget allocation in a given round is not solely determined by the reward (hit rates). Rather, we implement it by sampling a Gaussian distribution $\mathcal{N}(\mu,\sigma^2)$ for the indicator γ of budget allocation. Specifically, for a subprefix that has been scanned with β probes and yielded α unique IPv6 periphery addresses, the parameters of its Gaussian distributions can be presented as: $\mu=\alpha/\beta$, $\sigma=\sqrt{\mu(1-\mu)}/\ln\beta$. Intuitively, the parameters μ and σ respectively reflect the cumulative reward rate of the corresponding subprefix and our confidence in it.

As shown in Fig. 3, in next-round scanning, A (even with a low cumulative reward rate) will be allocated with a high budget of ϕ probes (see Alg. 1) once its sampling exceed the threshold (0.5 in this example). Conversely, B (even with same cumulative reward rate as C) is more likely to be allocated with a low budget of ψ probes than C. Thus, the incorporation of sampling-based decision-making can facilitate the exploitation of high-reward subprefixes, while also enabling subprefixes trapped in local minima to conduct high-budget exploration opportunely.

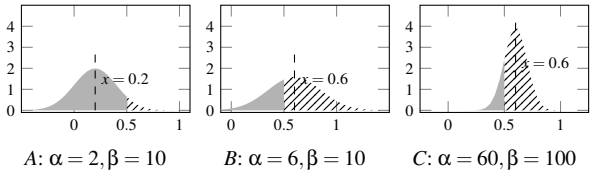


Figure 3: The posterior Gaussian distributions for sampling-based budget allocations in next scanning round (gray fills and line fill for low and high budget respectively) across three subprefixes, namely A, B and C.

Besides the exploration-exploitation trade-off, 6Seeks exhibits a novel feature in terms of its probe generation efficiency. Specifically, it can generate millions of probes within a few decision-making rounds. Conversely, directly utilizing existing solutions (e.g., bandit algorithms [26]) would lead to a significant decrease in probe generation efficiency since they are limited to producing only one outcome per step, eventually overwhelming the system.

In summary, the interaction-based budget allocation algorithm has demonstrated its effectiveness. However, implementing it on a multithreading model designed for high-speed scanning presents a challenge, namely the control state of each subprefix, which will be discussed in the next section.

3.4 Control State

To enable the efficient functioning of our scanner's probing logic, it is imperative to integrate a control state mechanism. This mechanism should store the response metadata associated with the relevant subprefixes, such as the discovery

Algorithm 1 Dynamic Budget Allocation

Require: PCS list $\{P_1, \ldots, P_n\}$, Sample threshold θ, High ϕ and Low ψ budget
Ensure: `uint_64` list `T` for /64 network prefixes

$\mathtt{T} \leftarrow \emptyset$
for $\mathtt{P} \in \{\mathtt{P}_1, \ldots, \mathtt{P}_n\}$ **do**
  $\mu \leftarrow \frac{\mathtt{P.hit}}{\mathtt{P.count}}, \sigma \leftarrow \frac{\sqrt{\mathtt{P.hit}(\mathtt{P.count}-\mathtt{P.hit})}}{\mathtt{P.count}\ln \mathtt{P.count}}$
  $\gamma \leftarrow \mathcal{N}(\mu, \sigma^2)$
  if $\gamma \geq \theta$ **then**
    $\mathtt{T} \leftarrow \mathtt{T} \cup \mathtt{PrefixGen}(\mathtt{P}, \phi)$
  else
    $\mathtt{T} \leftarrow \mathtt{T} \cup \mathtt{PrefixGen}(\mathtt{P}, \psi)$
  end if
end for

number and probing budget usage, in a manner that enables efficient retrieval for subsequent target generation. To fulfill this requirement, we propose implementing a dedicated data structure called the Prefix Control State, as depicted in Fig. 2. The Prefix Control State list in the 6Seeks implementation is

Algorithm 2 /64 Prefix Generation

Require: PCS `P`, Budget `b`
Ensure: `uint_64` list `T` for /64 network prefix

$\mathtt{T} \leftarrow \emptyset$
while $\mathtt{b} > 0$ **do**
  $\mathtt{t} \leftarrow \mathtt{P.stub} + \mathtt{P.offset}$
  $\mathtt{T} \leftarrow \mathtt{T} \cup \{\mathtt{t}\}$
  $\mathtt{P.offset} \leftarrow \mathtt{MULI} \times \mathtt{P.offset} + \mathtt{INC} \mod \mathtt{MOD}$
  $\mathtt{b} \leftarrow \mathtt{b} - 1, \mathtt{P.count} \leftarrow \mathtt{P.count} + 1$
end while

closely associated with two separate threads, namely, **Recv** and **Send**.

On the **Recv** thread side, upon receipt of a response packet, the Berkeley Packet Filter (BPF) is utilized to extract the expected ICMPv6 error messages. Following this, the packet validation process checks for duplicate periphery addresses using Bloom filters to prevent inaccuracies in reward estimation. This step is crucial as one IPv6 periphery address can respond to the probes towards different `/64` prefixes. For example, the `/60` prefix delegated to a CPE router comprises 16 `/64` prefixes (see Fig 1), each of which, except for the one assigned to the LAN interface, can trigger the CPE periphery to disclose its WAN IPv6 address. Finally, the corresponding sub-prefixes update their states in the `hit` fields of the Prefix Control State (PCS) list.

On the **Send** thread side, the focus lies on generating a specific number (allocated by Alg.1) of triggering addresses for the corresponding subprefixes. To tackle this challenge, we decouple the generation of `/64` prefixes and randomized Interface Identifiers, thereby reducing system complexity. However, if 6Seeks were to simply probe each `/64` prefix in numerical order, it would pose a risk of overloading destination networks with scan traffic and yield inconsistent results in the event of a remote transient network failure [27]. To circumvent this issue, 6Seeks produces the `/64` prefixes (uint64) using a linear congruential generator with suitable parameters (multiplier `MULI`, increment `INC`, and modulus `MOD`) in a designated `/64` prefix space, as shown in Alg.2. For example, to traverse the `/64` prefix space (2^{32}) of a subprefix `2001:1::/32` in a random permutation, 6Seeks initially defines the fields `stub` and `offset` of the PCS as the first `/64` prefix (`uint64:0x2001000100000000`) and a random value, respectively. One `/64` prefix can thus be obtained by summing the `stub` and `offset` values. Then, `offset` will be updated by the recurrence relation with parameters $\mathtt{MULI} = 1664525, \mathtt{INC} = 1013904223, \mathtt{MOD} = 2^{32}$ (see Alg. 2), where the period of `offset` recurrence is 2^{32}. In other words, 6Seeks can exhaustively traverse the entire 2^{32} space of `/64` prefixes without duplication while maintaining only negligible state (`uint64 stub` and `uint64 offset`). Finally, the triggering addresses are composed by concatenating the above `/64` prefixes with 64-bit Interface Identifiers provided by the fast *xorshift* random generator [28].

It is noteworthy that all update operations carried out on the Prefix Control State (PCS) lists are performed atomically to ensure data synchronization. However, a simple mutex assignment on the entire PCS block would encounter the problem of mutex contention, which frequently results in decreased system performance in multithreaded models [29]. To address this issue, 6Seeks implemented shard locks for mitigation, assigning a mutex to each PCS block (rather than the entire PCS list) before any data is read from or written to it. In this scenario, **Recv** and **Send** threads can concurrently update the PCS list without any competition or deadlocks, as it is highly unlikely that both of them access the same PCS block simultaneously. Our experimental results also demonstrate that allocating a budget of millions of probes only takes milliseconds on average, which is entirely feasible for Internet-wide scanning [30].

In summary, 6Seeks can achieve interaction-based IPv6 periphery scanning at a nearly system-level packet transmission rate, without additional time overhead.

4 Comparison Evaluation

We evaluate the performance of our scanner on real-world networks, in comparison with existing solutions, Xmap [4] and Edgy [3] (based on Yarrpv6 [31]).

4.1 Experiment Setup

We present the experimental setups as follows:

Datasets. IPv6 prefix data is used as a dataset for discovering IPv6 periphery. This dataset comprises publicly advertised IPv6 BGP prefixes obtained from RouteViews databases [32] (Covering 30789 public autonomous systems). These prefixes

serve as the data inputs for Edgy, Xmap and 6Seeks, allowing exclusion of unused IPv6 addresses/prefixes. Visualizing the vast scanning space can be challenging when there are diverse BGP prefixes. To mitigate this, we represent the BGP prefix dataset in standard units of /32 prefixes, as shown in Tab. 2.

Scanning Scale. To comprehensively evaluate its global status and provide an intuitive illustration of its uneven distribution, the IPv6 periphery scanning is conducted at two scales. 1) *Continent-level scanning* involves the utilization of Edgy, Xmap, and 6Seeks for IPv6 periphery discovery on six sets of continent-level prefixes (seeds) with billion-scale scanning budgets. Kindly note that evaluating Edgy's performance on a fixed budget is not suitable because Edgy utilizes traceroute-like methods, while Xmap and 6Seeks are ping-like scanners. Therefore, we only present its results for reference but not as a measure of its real-world performance on the Internet. 2) *Global-level scanning* involves the utilization of Xmap and 6Seeks for IPv6 periphery discovery on a total of 227,407 `/32` prefixes extracted from publicly available IPv6 BGP tables with ten-billion-scale scanning budgets.

Vantage Point. To mitigate potential bias from internet censorship on network measurements, two vantage points were deployed on well-connected virtual private servers located in Asia (AS3*963) and Americas (AS2*832). The Asia's vantage point uses a relatively premium host, while the one in Americas uses an entry-level host.

Table 2: Characteristics of Scanning Inputs

Range	2001::/16	240e::/16	2600::/16	2804::/16	2a02::/16	2c0f::/16
# /32	10994	4096	3351	6400	5561	699
Affiliation	APNIC	AfriNIC	ARIN	NIC.br	RIPE NCC	LACNIC

Probing Protocols. The protocol used by probes is unlikely to bias the results of IPv6 periphery scanning since only ICMPv6 unreachable error messages are considered valid responses from IPv6 periphery devices. In our comparison experiments, we adopted the ICMPv6 echo request as the probing payload due to it is designed for diagnostic purposes and is less intrusive compared to UDP/TCP probes, as referenced in prior works [3, 4, 11].

Aliasing Prefix. The hypothesis behind periphery discovery strategy is that an IPv6 address with a random Interface Identifier, cannot encounter any existing addresses among the vast address space (2^{64} or more) and thus triggers responses from the periphery devices. However, IPv6 scanning is inevitably affected by aliasing prefixes [33, 34]. Unlike IPv4, entire IPv6 prefixes can be linked to a single network interface. For instance, a host can support IPv6 aliasing prefixes by enabling the IP_FREEBIND Linux kernel option, which allows any probes towards the assigned prefixes to be received by this host and directly responded to with an ICMPv6 Echo Reply. Scanning on alias prefixes would provide more feedback, potentially causing subsequent scans to fall into the "alias trap," resulting in a significant waste of probing budgets. Therefore, we consider the triggering addresses that directly respond to the probes as IPv6 aliases in the ping-like methods (Xmap and 6Seeks), i.e., a variant of aliased prefix detection (APD) [34]. We also provide the number of aliased addresses in experiments for reference.

Evaluation Metric. As previously mentioned, it is not feasible to render a comprehensive list of IPv6 periphery addresses, referred to as A, by scanning every address in the entire IPv6 address space. Current scanning capabilities would require over 10,000 years to complete this task. Therefore, to quantitatively evaluate the ability of existing methods, we propose two metrics, namely hit rate and scanning efficiency. Formally, a methodology can be proposed to discover the set of periphery addresses, namely τ, within a limited budget b (i.e., the number of probing packets). The hit rate of IPv6 periphery discovery can be expressed as $\frac{|\tau|}{b}$, where $\tau \subset A$. Similarly, given the time consumption t required for the aforementioned IPv6 periphery discovery, the corresponding scanning efficiency can be expressed as $\frac{|\tau|}{t}$, where $\tau \subset A$.

Parameters. 6Seeks incorporates the parameters empirically (see Alg. 1) as follows: The high budget is $\phi = 100$, the low budget is $\psi = 1$, and the sampling threshold is $\theta = 0.01$. Additionally, in the ping-like methods (Xmap and 6Seeks), the *hop limits* of probes are all set to the maximum value of 255, ensuring that they can reach the network periphery.

4.2 Hit Rate Performance

The results demonstrate that 6Seeks outperforms the baselines in terms of hit rates in almost all scans. Specifically, 6Seeks achieves an average hit rate of 2.34‰, resulting in approximately 25× and 6× improvements compared to the state-of-the-art methods in continent-level scanning and global-level scanning, respectively. (see Tab. 9 for more details)

Additionally, the findings from continent-level scanning reveal significant variations in the number of IPv6 periphery addresses across different continent-level `/16` prefixes, despite the same probing budget allocation. For example, a considerably greater number of IPv6 periphery addresses can be discovered in the 2a02::/16 range compared to the 2c0f::/16 range. In essence, the distribution of IPv6 network periphery devices is uneven across the address space.

The location of vantage points can result in substantial geographic bias during large-scale IPv6 periphery scanning, as shown in Fig. 16. Due to network incidents (such as Internet censorship), periphery addresses situated in particular geographic regions may respond to vantage points in Americas but not to the one in Asia, or vice versa. For example, 6Seeks and Xmap found 60% and 40% more IPv6 periphery addresses in Americas, respectively, compared to Asia.

4.3 Scanning Efficiency

The unfettered deployment of scanners on real networks without appropriate probing restrictions may trigger traffic cur-

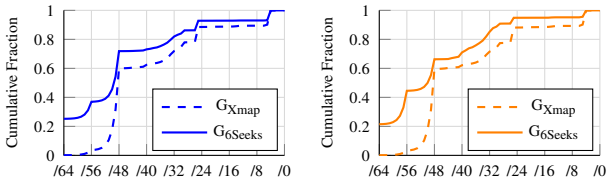
tailment measures on the local gateway as it is mistaken as a DDoS attack. Consequently, to prevent such an occurrence, we constrain the peak probing rate to 100 Kilo packets per second (Kpps). Additionally, we introduce the concept of *Probing Speed* as the average probing rate during the entire scanning procedure, namely the ratio of the total number of probing packets to the total time.

The results show that all scanners could reach the limit boundary of the probing speed at the vantage point. However, the probing speeds at the vantage point of Americas commonly decreased to varying degrees (see Tab. 9) as the probing host in Americas is entry-level. Nonetheless, the 6Seeks scanner remained the fastest scanner at the vantage points both in Americas and Asia, demonstrating our system reliability under limited conditions.

As a result, 6Seeks is considered the most efficient scanning method due to its ability to discover the highest number of periphery addresses and attain the fastest probing rates. When compared to other available methods, 6Seeks can achieve 25× and 8× improvements on average in terms of scanning efficiency in continent-level and global-level scanning respectively.

4.4 Periphery Validation

Validating the IPv6 periphery presents a significant challenge as obtaining the ground truth for millions of periphery addresses is unfeasible. Consequently, we focus on three following aspects to discriminate the discovered IPv6 addresses.



(a) Vantage Point at Asia (b) Vantage Point at Americas

Figure 4: Network prefix length distribution in global-level periphery scanning by 6Seeks and Xmap.

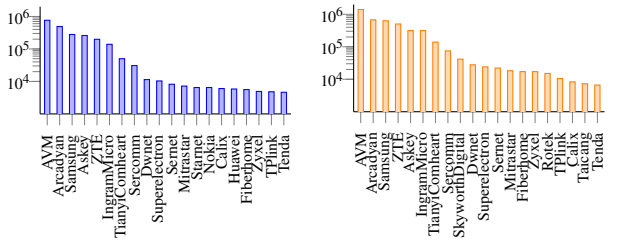
Network Prefix Length. As mentioned in the basic hypothesis (see § 3.1), periphery hosts would respond with ICMPv6 messages to triggering addresses within their assigned network prefix. Conversely, the network prefix length can be inferred from the length of common upper bits of the periphery address and the triggering address. As shown in Fig. 4, the addresses with `/48` or longer prefixes (defined by RIPE NCC [35] for a single End Site) account for the majority of discoveries for Xmap and 6Seeks, 60% and 70% of total respectively, which aligns with the requirements of IPv6 periphery model. A quarter of 6Seeks's discoveries belong to the addresses with `/64` prefixes (typically assigned to end sites), whereas Xmap's discoveries do not include any address

Table 3: Open Ports of IPv6 Periphery Addresses Discovered in Global-level Scanning

	VP	21_{TCP}	22_{TCP}	23_{TCP}	53_{UDP}	80_{TCP}	443_{TCP}	8080_{TCP}
G_{Xmap}	Asia	9519	19557	5729	3230	8781	11279	3530
	Americas	13496	30550	9100	4288	12216	15564	4681
G_{6Seeks}	Asia	26437	54766	17494	44186	68820	74330	70680
	Americas	56186	69183	22636	84442	94242	153085	143631

with the `/64` prefix. Furthermore, it can be inferred that IPv6 address allocation primarily follows the sizes of `/64`, `/56`, and `/48` based on corresponding inflection points in 6Seeks, which exactly consists with the IPv6 Address Allocation and Assignment Policy [35].

Device Vendor. As mentioned earlier, EUI64 addresses are generated by incorporating the device's MAC address to create a unique 64-bit Interface Identifier [15]. Conversely, we can extract the MAC address from EUI64 addresses to analyze the manufacturing vendors of IPv6 periphery devices. Fig. 5



(a) Vantage Point at Asia (b) Vantage Point at Americas

Figure 5: Top 20 periphery device vendors in global-level scanning by 6Seeks.

shows that the majority of devices with EUI64 addresses are produced by IoT vendors or consumer electronics companies. Their main products are precisely customer premises equipment, such as AVM and ZTE, as well as user equipment like Huawei and Samsung.

Open Port. IPv6 network periphery devices hosting various services will inevitably expose ports on the Internet. Therefore, by scanning these well-known ports, we can confirm that the addresses discovered by 6Seeks and Xmap belong to the IPv6 network periphery, as indicated in Tab. 3. Even with firewalls in place, hundreds of thousands of IPv6 addresses still respond to port probes. The port exposure of critical services like FTP, SSH, Telnet, DNS, HTTP/HTTPS, typically deployed on network periphery devices, has been identified in the addresses discovered by 6Seeks and Xmap, indicating their likely association with the periphery devices.

5 The Vulnerability: IPv6 Backdoor

In this section, we proposed a case study of a novel vulnerability where the IPv6 channels serve as a backdoor to expose the vulnerable services in the internal network to the open Internet, resulting in the risk of unauthorized access.

5.1 Threat Model

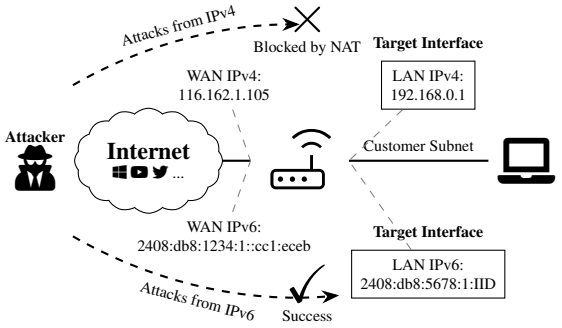


Figure 6: An illustration of IPv6 backdoor vulnerability.

To enhance the practicality of this vulnerability, we consider a common scenario in which a dual-stack home Wi-Fi router serves as the Internet gateway for user devices in a local area network (LAN), as depicted in Fig. 6. Various application services are deployed on the LAN interfaces, including a DNS cache server and the gateway's login page. From the perspective of IPv4 networks, these vulnerable services are inaccessible from external sources because the LAN IPv4 addresses are private addresses provided by Network Address Translation (NAT) gateways, and any request targeting an IPv4 private address would be discarded on the Internet. However, the LAN IPv6 addresses serve as a backdoor that enables an attacker to easily hack vulnerable application services on the LAN interface for cybercriminal activities.

Unlike previous backdoors, our vulnerability does not rely on specific application services or CVEs, but arises from the irrational design of the default IPv6 deployment. To enable end-to-end communications for numerous customer devices (e.g., smart home and IoT devices) on the Internet, carriers are accelerating support for IPv6 prefix delegation [16], where a CPE must be implemented with a *weak host model* and assign a separate globally routable `/64` prefix from its delegated prefix(es) to each of its LAN interfaces [20, 21]. As a result, the LAN interfaces are equipped with global Internet access, and requests from external sources would be forwarded through the WAN interface to the LAN interfaces. Due to a stereotypical experience on IPv4 networks, a large number of vulnerable services are deployed on the LAN interface by default, eventually resulting in the threats of IPv6 backdoor vulnerability.

5.2 Measurement Methodology

Previous works [4,27] focused on disclosing services exposed on IPv6 network periphery without considering some services that are inherently open on the Internet to users (e.g., remote connectivity and work collaboration), as IPv6 networks allow and are dedicated for this purpose. Thus, it poses a challenge to ascertain whether these services and addresses are intentionally or unintentionally exposed to the open Internet.

To this end, we propose a methodology for assessing the unintentional exposure of vulnerable services, which involves collecting the Subject Alternative Name fields reported in the TLS handshake log. Specifically, we utilize ZGrab2 [36] to scan the TLS protocols of network periphery addresses on ports 443 and 8080. When a network periphery device sends its digital certificate, it includes a subject field known as the Subject Alternative Name (SAN) [37, 38]. This field contains domain names or IP addresses associated with the server. By analyzing these fields, we can infer the user's intentions regarding the exposure of services, as shown in Fig.7.

(a) Unintentional Exposure (b) Intentional Exposure (c) Intentional Exposure

Figure 7: The examples of Subject Alternative Name in TLS handshake log.

We present the fundamental assumptions for inferring the user's intentions regarding the exposure of services as follows: Generally, it is unlikely that authorized users would access the exposed services on periphery devices using bare 128-bit IPv6 addresses as it doesn't make sense. Instead, it is common for periphery devices with available services to be associated with public domain names or, at the very least, a 32-bit public IPv4 address to enable convenient remote access. In other words, the purpose of how the users utilize the services (exposed to the Internet or not) can be inferred based on the following heuristic rules: *Users intend to expose the services deployed on network periphery devices to the open Internet if the SAN field includes at least one public domain name or one public IPv4 address. Otherwise, the network periphery may be at risk of IPv6 backdoors without the users' awareness.*

As shown in Fig. 7, the example of unintentional exposure could be inferred as the login page of a home router (a CPE) which should be visible on the open Internet. It would be regarded as an IPv6 backdoor as there are no public domain name or public IPv4 address but a bare IPv6 addresses. Conversely, another two cases are not within this threat as they are exposed by users with a public IPv4 address `194.210.x.x` or public domain name `xxxxxxrused.myfritz.net`.

5.3 Scanning Results

We collected the TLS handshake results of periphery addresses during the 6Seeks global scanning using ZGrab2 [36]. As a result, we identified a total of 126K unique addresses (51K in Asia and 75K in Americas) that expose a gateway login page to the open Internet, which ideally should have been restricted to the LAN interface within the internal network,

making them vulnerable to the IPv6 backdoors. Furthermore, we conducted banner grabbing to analyze the service versions of well-known services (e.g., DNS and HTTP) to provide additional insight into the extent of this threat.

The analysis of banner grabbing revealed that periphery devices at risk of IPv6 backdoor vulnerabilities primarily had 67K instances of SSH software, 126K instances of HTTP software, 74K instances of FTP software, and 39K instances of DNS software. Furthermore, several hundreds instances of other vulnerable services, such as RSTP and VNC, were disclosed.

To demonstrate the vulnerability of these devices exposed to the open Internet, we present the versions of the top software used in these vulnerable periphery devices, as shown in Tab. 4. Consistent with our predictions, the majority of the software is outdated and has long-standing vulnerabilities reported in the CVE database. For instance, the widely utilized DNS software dnsmasq has been documented to possess a minimum of 16 high-risk vulnerabilities [39] in our findings.

Table 4: Top Softwares Exposed by IPv6 Backdoor

FTP Services		HTTP Services	
Software (Number)	Version with CVE	Software (Number)	Version with CVE
bFTPd: 3803	2.2.2	Apache: 1200	2.2.x, 2.4.x
Pure-FTPd: 336	1.0.x	Boa/HTTP: 6990	0.93.15, 0.94.8
GNU inetutils: 1179	-	Lighttpd: 22164	1.4.x
FTPd: 1859	FreeBSD 6.00LS	Micro-httpd: 60102	2.0, 5.0, 6.0
Fritz!Box: 5185	-	Mini-httpd: 17005	1.19, 1.27
vsFTPd: 218	2.2.2, 2.3.4, 3.0.x	Nginx: 3207	1.4.x, 1.14.x
MikroTik/FTP: 3166	-	ZTE Web: 10484	1.0
SSH Services		**DNS Services**	
Software (Number)	Version with CVE	Software (Number)	Version with CVE
Dropbear: 46238	0.4x, 0.5x, 2017.x	Unbound: 2062	1.4.11, 1.4.13
OpenSSH: 41932	3.x,4.4,6.0,7.x	PowerDNS: 2082	4.1.11, 4.2.x
MikroTik/SSH: 9415	-	dnsmasq: 5291	2.2x, 2.4x, 2.7-2.8
Cisco SSH: 31411	-	BIND: 8774	9.5.x, 9.16.x

Fig. 8 demonstrates that the scope of IPv6 backdoors is not limited and the potential victims are widespread, particularly in Latin America and Europe (utilizing GeoLite2 [40] to locate IP addresses). Hence, The IPv6 network is currently at a high risk of IPv6 backdoors.

6 The Vulnerability: Reflection Amplification

The acquisition of a substantial number of IPv6 traffic reflectors presents challenges due to the continued absence of efficient IPv6 scanning tools [12]. Consequently, research on IPv6 reflection attacks has remained stagnant. The sole existing approach [41], which involves scanning the IPv4 address space to identify open dual-stack resolvers and subsequently cross-referencing their DNS records to obtain the addresses of corresponding IPv6 resolvers, ultimately finds only about 1,000 IPv6 traffic reflectors.

Using 6Seeks, we can quickly obtain a substantial number of IPv6 periphery addresses hosting a wide range of network services vulnerable to security exploitations (including traffic reflections), and report a novel vulnerability of reflection amplification on IPv6 Internet middleboxes.

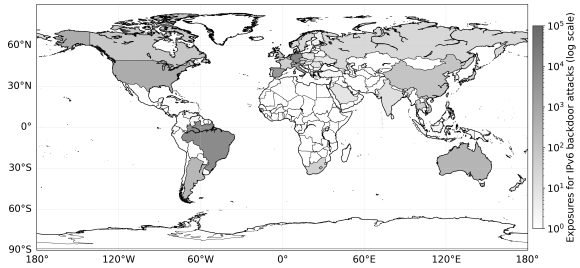


Figure 8: Geographical distribution of potential victims for IPv6 backdoors.

6.1 Threat Model

The vulnerability of IPv6 reflection amplification utilizes address spoofing to manipulate the reflector into redirecting a significant amount of traffic towards the victim hosts.

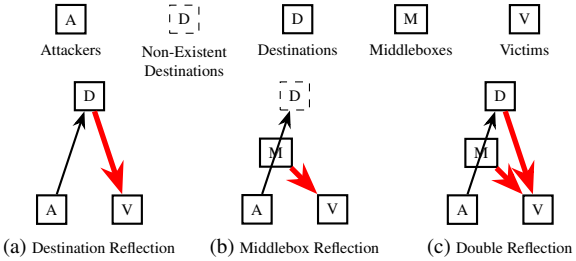


Figure 9: Types of vulnerabilities related to IPv6 reflection amplification. Thin arrows denote packets that trigger amplification while thick red ones is the amplification traffic.

Fig. 9 illustrates the types of IPv6 reflection amplification under consideration. The first case involves regular traffic reflection, where attackers falsify the source address of packets as the victim's, causing the destination hosts (reflectors) to respond to the victims. Like the first case, the second type of traffic reflection also employ address spoofing to redirect response traffic towards the victims. While the destination host is a non-existent IPv6 address (which is common in the vast IPv6 address space), there cannot theoretically be traffic amplification. However, in real-world IPv6 networks, certain packets with carefully crafted payloads can still facilitate IPv6 amplification attacks due to the presence of Internet censorship infrastructure (middleboxes). The final type combines the amplification capabilities of both the middleboxes and the destination hosts, resulting in increased traffic towards the victims.

6.2 Measuring IPv6 Address Spoofing

In this section, we assess the potential risks by analyzing IPv6 address spoofing in the global scanning results obtained from 6Seeks.

Methodology. Since conducting an amplification attack in the real-world Internet is illegal and immoral, we can only perform limited scanning of the IPv6 periphery devices sus-

ceptible to address spoofing with our two vantage points. For the periphery addresses scanned in Americas, we use the address of our VPS located in Asia as the source address for spoofed packets. These packets are sent from the vantage point in Americas to induce the periphery devices to amplify the traffic. The settings are reversed for addresses scanned in Asia. Additionally, we send only one packet per request type to each periphery address, and the scan rate is strictly limited to 10 Kpps.

Table 5: Address Spoofing on IPv6 Periphery Addresses Discovered in 6Seeks Global-level Scanning

Name/Port	Request	Vantage Point	
		Asia	Americas
DNS^{g}(UDP53)	"A" Record of `www.google.com`	413379	1106512
DNS^{b}(UDP53)	"A" Record of `www.baidu.com`	41169	77067
NTP(UDP123)	NTPv4 Client Unsynchronized	6475	8534
SNMP(UDP161)	GetRequest for OID `sysDescr`	60007	134683
ICMP	ICMPv6 Echo Request	973948	974584

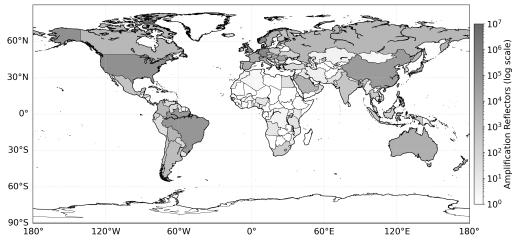


Figure 10: Geographical distribution of IPv6 network periphery addresses which exhibit the vulnerabilities to IPv6 reflection amplification.

Results. Fig.10 illustrates the widespread distribution of potential reflectors discovered by 6Seeks (utilizing GeoLite2 [40] to locate IP addresses). As shown in Tab.5, the address spoofing results from global periphery addresses, as discovered by 6Seeks, demonstrate the potential exploitation of approximately 130K DNS, 200K SNMP, and 15K NTP servers as reflection amplifiers. Furthermore, the experiments revealed that around 2 million IPv6 periphery addresses responded to address spoofing requests, and our vantage points, acting as "victims", were flooded with responses during the address spoofing experiments. Therefore, hosts connected to the IPv6 Internet could become potential victims of IPv6 amplification attacks, regardless of their geographic location.

In Fig. 11, a comparison of amplification factors for each type of request is presented. It is evident that certain requests can elicit spoofing responses from over 1 million IPv6 periphery addresses, with amplification factors exceeding one. Additionally, several thousand destination addresses in the IPv6 periphery provided a high amplification factor, with up to 10× amplification, and the maximum amplification factor reached 1091×.

Note that a significantly larger number of periphery addresses respond to DNS^{g} resolutions. However, these resolution records commonly contain errors, as they translate

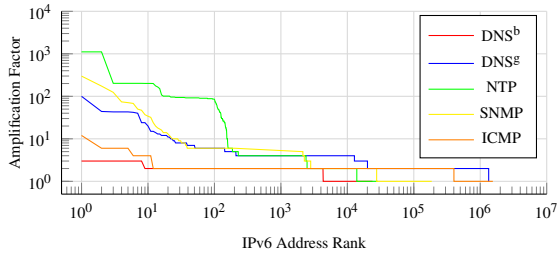


Figure 11: Rank order plot of the amplification factor received from each IP address across address spoofing requests.

domain names like `www.google.com` into IP addresses of sites with unintended content or non-existent IP addresses. This indicates that even with a fixed number of reflectors, the number of responses to address spoofing can still be influenced by the payload of probing packets. To uncover the underlying reasons for this discrepancy in probing payloads, we present the top Autonomous Systems (ASes) where the addresses respond to DNS^{g} requests, along with the number of periphery addresses, in Tab. 6. Additionally, we provide the corresponding results of DNS^{b} responses for comparison.

Table 6: Top 10 Autonomous Systems Responding DNS^{g} in Address Spoofing Measurements

(a) Vantage Point at Asia

#	-	DNS^{g}	DNS^{b}
1	**AS41*4**	199521	177
2	**AS98*8**	60586	515
3	**AS48*2**	41656	1265
4	**AS56*46**	12444	207
5	**AS24*45**	11318	41
6	**AS56*41**	11102	277
7	**AS24*44**	6350	1
8	**AS56*47**	4436	27
9	**AS24*47**	3886	45
10	**AS56*42**	3020	22

(b) Vantage Point at Americas

#	-	DNS^{g}	DNS^{b}
1	**AS41*4**	505423	22
2	**AS98*8**	167980	380
3	**AS48*2**	98920	1430
4	**AS56*46**	31200	326
5	**AS24*45**	26532	2
6	**AS56*41**	25584	330
7	**AS24*44**	17252	6
8	**AS56*47**	16586	1258
9	**AS24*47**	11639	54
10	**AS56*42**	11523	93

According to publicly available information [42], the domain name `*.google.com` is blocked by DNS hijacking in the above autonomous systems, whereas `www.baidu.com` is currently unaffected. Thus, any DNS^{g} directed at these autonomous domains will receive deceptive address resolutions from the Internet censorship infrastructure (middleboxes), causing inflated measurement results based on DNS^{g}, as shown in Fig. 9.

6.3 The Macro Amplifier

As previously discussed, meticulously designed DNS requests can result in network middleboxes used for Internet censorship efficiently amplifying network traffic to carry out amplification attacks against targeted entities. Compared to the open resolvers in the real world, these middleboxes have stronger traffic reflection capabilities, and therefore pose a greater threat, as shown in Fig. 11.

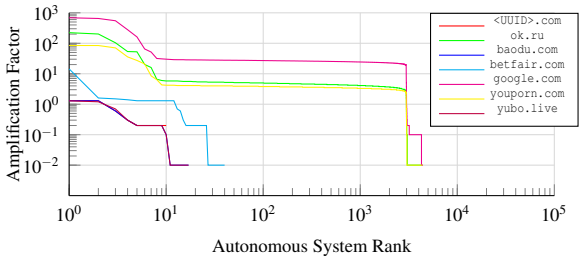


Figure 12: Rank order plot of the macro amplification factor received from each autonomous system across domain names.

Different from previous understandings of reflection attacks, this novel vulnerability to traffic amplification does not necessitate a predefined list of reflectors/amplifiers. Instead, the attacker utilizes all the IPv6 addresses within an autonomous system as a whole, referred to as the **macro amplifier** and sends DNS queries with meticulously crafted payloads to random IPv6 addresses within the **macro amplifier**. As a result, an influx of DNS resolution responses from the middleboxes could overwhelm the victims' networks.

To validate the practicality of this traffic amplification and investigate the number of **macro amplifiers**, we randomly sampled 2,000 addresses for each autonomous system from the Routeview BGP tables, resulting in the acquisition of 62 million IPv6 addresses for the destinations of address spoofing measurements. Considering the sparse IPv6 address space, it is unlikely that these addresses correspond to existing open resolvers. In order to mitigate measurement biases due to chance, we selected domain names that are blocked in certain autonomous systems according to reports from Censored Planet [42]. These domains include `ok.ru`, `www.google.com`, `www.betfair.com`, `www.youporn.com`, and `www.yubo.live`. Additionally, we chose one domain, `www.baidu.com`, that is unlikely to be blocked, and a non-existent domain (`<UUID>.com`) whose second level domain is a randomized universally unique identifier, to facilitate a comprehensive evaluation. Fig. 12 presents a comparison of the macro amplification factors (defined as the ratio of responses to probes for each respective Autonomous Systems) for the address spoofing payload. Notably, more than 4000 Autonomous Systems exhibit the macro amplification factors exceeding one for payload using these blocked domain names, with the **macro amplifier** achieving a maximum amplification factor of 668×.

It can be concluded that IPv6 traffic amplification might pose a greater threat compared to IPv4 due to the following reasons: 1) IPv6 enables end-to-end communications between end-hosts and internal devices, making IPv6 devices potential victims exposed directly to amplification attacks and allowing attackers to utilize those vulnerable IPv6 devices as reflectors. 2) Numerous vulnerable services have already been exposed to the IPv6 network due to prolonged neglect of IPv6 security. Thus, the rapid and easy discovery of IPv6 reflectors is not a challenging task, as demonstrated in our experiments. 3) In IPv6 networks, hackers can utilize weaponized middleboxes as traffic amplifiers for conducting DDoS attacks without requiring specific reflector addresses. And, we strongly advocate for attention to be given to the risks associated with IPv6 amplification attacks.

7 Ethical Considerations

Internet Scanning. We ensure good Internet citizenship as suggested by Partridge and Allman [43]. To minimize the impact on target networks, we 1) limit the uplink bandwidth to < 100 kpps. 2) probe each `/64` prefix only once through random permutation, thus spreading the traffic across a vast address space.

Traffic Reflection. During measuring the IPv6 Amplification Attacks, all the "victims" are acted upon by our vantage points. And, considering the potential traffic amplification, we limit the uplink bandwidth to 10 kpps for sending the address spoofing packets for ensuring reasonable bandwidth usage.

Responsible Disclosure. All addresses mentioned in the papers have been anonymized to protect personal privacy. In our experiments, it is necessary to disclose the relevant device vendors and autonomous systems for clear illustrations. Considering the sensitive nature of these studies in this paper, we have anonymized the associated autonomous system numbers using a wildcard "*" to avoid any potential conflicts of interest. We also report our findings to these vendors and autonomous systems, and we remove any results related to entities that explicitly refuse our disclosure requests. Additionally, we have disclosed all the vulnerabilities found by 6Seeks to the National Vulnerability Database of our country.

8 Countermeasures

IPv6 scanning technology is developing rapidly, and as a result, an increasing number of threats against IPv6 networks are being uncovered. In this paper, we identify two specific IPv6 attacks that occur during the global IPv6 periphery scan and are caused by either insecure default configurations or poor design choices. In light of these findings, we propose the following countermeasures:

Customer. Promptly verify if prefix delegation is enabled on the home router and disable it to avoid exposing user equipment and LAN's services to the open Internet if this is not the user's intention. For remote connectivity, ensure that all devices are equipped with protective software and strong access control. Unlike traditional IPv4 networks, devices connected to IPv6 lack the protection of private gateways and are vulnerable to hacking attacks, such as unauthorized access and denial-of-service attacks (DDoS).

Internte Service Providers. 1) Discrete support for prefix delegation. Regular consumers may unknowingly expose

their network devices to the open Internet by default, putting their privacy and devices at risk. Therefore, it is the responsibility of the providers to inform users and enable it discretely without user approval. 2) Promote source address validation. Implementing effective source address verification can greatly reduce the harm caused by address spoofing, such as DNS hijacking and amplification attacks. A simple solution is to follow the Network Ingress Filtering Strategy introduced by BCP38 [44]. This strategy not only prevents malicious behaviors that use IPv4 address spoofing but also mitigates newly discovered IPv6 amplification attacks. Even if ISPs do not immediately benefit from ingress filtering, a more secure Internet will significantly reduce network protection costs. 3) Limit censorship devices. The abuse of Internet censorship can be easily exploited by malicious third-party attackers, leading to data breaches, system downtime, and reputational harm. For example, middleboxes play a crucial role in amplification attacks related to Internet censorship. Therefore, limiting censorship measures can help mitigate this attack. Moreover, DNS hijacking will become ineffective with the adoption of DNS over HTTPS/TLS [45], but the damage caused by amplification attacks will persist until these devices are eliminated.

Researchers. The presence of bias stemming from geographic differences and Internet censorship is apparent and inevitable in network measurements. Therefore, it is essential to utilize diverse vantage points for a comprehensive assessment of the Internet.

9 Related Work

IPv6 periphery discovery. Numerous efforts have been devoted to exploring the IPv6 network globally, which involve active scanning for identifying patterns or structures [46–48], passive data collection [49], and releasing hitlists [34, 50, 51]. These approaches have facilitated the discovery of IPv6 network periphery addresses. For example, Beverly et al. [9] utilized known active IPv6 addresses and a randomization scanning tool called Yarrp to identify 1.3 million IPv6 router interface addresses. Similarly, Rye et al. [3] traced targets obtained from BGP advertisements or IPv6 hitlists to discover 64 million IPv6 router addresses. Li et al. proposed Xmap [4], an asynchronous scanning tool based on ZMap, which leverages the observations of IPv6 routing to reduce scanning scope. However, the search space for this task remains at 2^{64} without considering known active addresses and prefix allocation, and scanning hit rates remain low. It is evident that a more efficient (fast probes) and effective (high hit rates) approach is needed to fulfill the requirements of Internet-wide IPv6 periphery discovery. Our work aims to address these challenges.

Interaction-based scanning. It, also known as reinforcement learning-based network scanning, has been widely employed in network scanning methods such as 6hit [11] and 6scan [30]. 6hit was the first to use reinforcement learning for IPv6 target generation, resulting in significant improvements in active IPv6 address discovery. 6scan incorporated the reinforcement learning algorithm into an asynchronous scanning tool, greatly enhancing scanning efficiency. Similarly, FlashRoute [52], an asynchronous traceroute tool on a massive scale, employs interaction-based scanning concepts to dynamically reduce the number of tracing probes on a route if routers near the vantage points have already been explored. Overall, the fundamentals of reinforcement learning-based network scanning are to prioritize scanning targets based on their demonstrated efficacy in past scans, thereby facilitating incremental scanning and optimizing cost-effectiveness through prudent resource allocation and a reduction in redundant probing.

Amplification/Reflection Attack. Research on IPv6 reflection amplification attacks is rare due to the persistent misconception that the vast address space in IPv6 makes it difficult to collect sufficient traffic reflectors. As the only work related to IPv6 amplification attack, Hendriks, et al. [41] propose scanning the entire IPv4 address space and then turning to discover the dual-stack open resolvers for potential reflectors using the higher layer DNS protocol. As a result, 1038 unique IPv6 addresses are verified to be openly resolving and pioneeringly illustrate the feasibility of IPv6 amplification/reflection attacks. However, how to quickly reveal the numerous IPv6 traffic reflectors (not limited to the DNS protocol) to evaluate the threats of amplification attacks in IPv6 is still an open problem and we are thus dedicated to address it.

Additionally, Kevin, et al. [53] scanned the entire IPv4 Internet and found that a number of IPv4 network middleboxes (mostly attributing the Internet censorship) can be utilized as the reflectors for amplification attacks. In this paper, we also reveal the potential weaponizing middleboxes for reflected amplification in IPv6 and present a brief analysis of this "kill with a borrowed sword".

10 Conclusion

This paper presents 6Seeks, an efficient tool for scanning the periphery of the IPv6 network, aimed at promoting vulnerability discovery in the IPv6 Internet. 6Seeks utilizes an innovative interaction-based scanning approach and outperforms existing methods in terms of hit rate and scanning efficiency, thanks to its dynamic allocation strategy for probing budgets. Empirical experiments conducted in real-world networks demonstrate the capabilities of 6Seeks in investigating the IPv6 network periphery on the Internet-wide scale. Additionally, the discovery of two new vulnerabilities in IPv6 (i.e., IPv6 backdoor and reflection amplification) not only highlights the effectiveness of 6Seeks in mitigating threats to the IPv6 network but also underscores the urgent need to prioritize IPv6 network security.

References

[1] Google, "IPv6 Adoption Statistics," 2023. [Online]. Available: https://www.google.com/intl/en/ipv6/statistics.html

[2] W3Techs, "Usage Statistics of IPv6 for Websites," 2023. [Online]. Available: https://w3techs.com/technologies/details/ce-ipv6

[3] E. C. Rye and R. Beverly, "Discovering the IPv6 Network Periphery," in *Proc. PAM*. Springer, 2020, pp. 3–18.

[4] X. Li, B. Liu, X. Zheng, H. Duan, Q. Li, and Y. Huang, "Fast IPv6 Network Periphery Discovery and Security Implications," in *Proc. DSN*. IEEE, 2021, pp. 88–100.

[5] E. Rye, R. Beverly, and K. C. Claffy, "Follow the Scent: Defeating IPv6 Prefix Rotation Privacy," in *Proc. IMC*, 2021, pp. 739–752.

[6] S. J. Saidi, O. Gasser, and G. Smaragdakis, "One Bad Apple Can Spoil Your IPv6 Privacy," *ACM SIGCOMM Computer Communication Review*, vol. 52, no. 2, pp. 10–19, 2022.

[7] D. Plohmann, K. Yakdan, M. Klatt, J. Bader, and E. Gerhards-Padilla, "A Comprehensive Measurement Study of Domain Generating Malware," in *Proc. USENIX Security*, 2016, pp. 263–278.

[8] S. Hu, Q. A. Chen, J. Sun, Y. Feng, Z. M. Mao, and H. X. Liu, "Automated Discovery of Denial-of-Service Vulnerabilities in Connected Vehicle Protocols," in *Proc. USENIX Security*, 2021, pp. 3219–3236.

[9] R. Beverly, R. Durairajan, D. Plonka, and J. P. Rohrer, "In the IP of the Beholder: Strategies for Active IPv6 Topology Discovery," in *Proc. IMC*, 2018, pp. 308–321.

[10] L. Pan, J. Yang, L. He, Z. Wang, L. Nie, G. Song, and Y. Liu, "Your Router is My Prober: Measuring IPv6 Networks via ICMP Rate Limiting Side Channels," in *Proc. NDSS*, 2023.

[11] B. Hou, Z. Cai, K. Wu, J. Su, and Y. Xiong, "6Hit: A Reinforcement Learning-based Approach to Target Generation for Internet-wide IPv6 Scanning," in *Proc. INFOCOM*. IEEE, 2021, pp. 1–10.

[12] J. Ullrich, K. Krombholz, H. Hobel, A. Dabrowski, and E. Weippl, "IPv6 Security: Attacks and Countermeasures in a Nutshell," in *Proc. USENIX Security*, 2014.

[13] A. Conta, S. Deering, and M. Gupta, "Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification," Internet Requests for Comments, RFC Editor, RFC 4443, March 2006. [Online]. Available: http://www.rfc-editor.org/rfc/rfc4443.txt

[14] T. Mrugalski, M. Siodelski, B. Volz, A. Yourtchenko, M. Richardson, S. Jiang, T. Lemon, and T. Winters, "Dynamic Host Configuration Protocol for IPv6 (DHCPv6)," Internet Requests for Comments, RFC Editor, RFC 8415, November 2018.

[15] S. Thomson, T. Narten, and T. Jinmei, "IPv6 Stateless Address Autoconfiguration," Internet Requests for Comments, RFC Editor, RFC 4862, September 2007. [Online]. Available: http://www.rfc-editor.org/rfc/rfc4862.txt

[16] O. Troan and R. Droms, "IPv6 Prefix Options for Dynamic Host Configuration Protocol (DHCP) version 6," Internet Requests for Comments, RFC Editor, RFC 3633, December 2003.

[17] S. Deering and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification," Internet Requests for Comments, RFC Editor, STD 86, July 2017.

[18] R. Hinden and S. Deering, "IP Version 6 Addressing Architecture," Internet Requests for Comments, RFC Editor, RFC 4291, February 2006. [Online]. Available: http://www.rfc-editor.org/rfc/rfc4291.txt

[19] T. Narten, R. Draves, and S. Krishnan, "Privacy Extensions for Stateless Address Autoconfiguration in IPv6," Internet Requests for Comments, RFC Editor, RFC 4941, September 2007.

[20] H. Singh, W. Beebee, C. Donley, and B. Stark, "Basic Requirements for IPv6 Customer Edge Routers," Internet Requests for Comments, RFC Editor, RFC 7084, November 2013.

[21] R. Braden, "Requirements for Internet Hosts - Communication Layers," Internet Requests for Comments, RFC Editor, STD 3, October 1989. [Online]. Available: http://www.rfc-editor.org/rfc/rfc1122.txt

[22] J. Korhonen, J. Arkko, T. Savolainen, and S. Krishnan, "IPv6 for Third Generation Partnership Project (3GPP) Cellular Hosts," Internet Requests for Comments, RFC Editor, RFC 7066, November 2013.

[23] C. Byrne, D. Drown, and A. Vizdal, "Extending an IPv6 /64 Prefix from a Third Generation Partnership Project (3GPP) Mobile Interface to a LAN Link," Internet Requests for Comments, RFC Editor, RFC 7278, June 2014.

[24] J. Woodyatt, "Recommended Simple Security Capabilities in Customer Premises Equipment (CPE) for Providing Residential IPv6 Internet Service," Internet Requests for Comments, RFC Editor, RFC 6092, January 2011.

[25] B. H. Korte, J. Vygen, B. Korte, and J. Vygen, *Combinatorial Optimization*. Springer, 2011, vol. 1.

[26] M. N. Katehakis and A. F. Veinott Jr, "The Multi-Armed Bandit Problem: Decomposition and Computation," *Mathematics of Operations Research*, vol. 12, no. 2, pp. 262–268, 1987.

[27] Z. Durumeric, E. Wustrow, and J. A. Halderman, "ZMap: Fast Internet-wide Scanning and Its Security Applications," in *Proc. USENIX Security*, vol. 8, 2013, pp. 47–53.

[28] F. Panneton and P. L'ecuyer, "On the Xorshift Random Number Generators," *ACM Transactions on Modeling and Computer Simulation*, vol. 15, no. 4, pp. 346–361, 2005.

[29] N. R. Tallent, J. M. Mellor-Crummey, and A. Porterfield, "Analyzing Lock Contention in Multithreaded Applications," in *Proc. SIGPLAN*, 2010, pp. 269–280.

[30] B. Hou, Z. Cai, K. Wu, T. Yang, and T. Zhou, "6Scan: A High-Efficiency Dynamic Internet-Wide IPv6 Scanner With Regional Encoding," *IEEE/ACM Transactions on Networking*, 2023.

[31] R. Beverly, "Yarrp'ing the Internet: Randomized high-speed active topology discovery," in *Proc. IMC*, 2016, pp. 413–420.

[32] U. of Oregon, "Route views project," 2023. [Online]. Available: https://www.routeviews.org/routeviews/

[33] A. Murdock, F. Li, P. Bramsen, Z. Durumeric, and V. Paxson, "Target Generation for Internet-wide IPv6 Scanning," in *Proc. IMC*, 2017, pp. 242–253.

[34] O. Gasser, Q. Scheitle, P. Foremski, Q. Lone, M. Korczynski, S. D. Strowes, L. Hendriks, and G. Carle, "Clusters in the Expanse: Understanding and Unbiasing IPv6 Hitlists," in *Proc. IMC*. New York, NY, USA: ACM, 2018.

[35] A. APNIC and N. RIPE, "IPv6 Address Allocation and Assignment Policy," 2020. [Online]. Available: https://www.ripe.net/publications/docs/ripe-738

[36] T. Z. Project, "ZGrab2: Fast Go Application Scanner," 2023. [Online]. Available: https://github.com/zmap/zgrab2

[37] T. Dierks and E. Rescorla, "The Transport Layer Security (TLS) Protocol Version 1.2," Internet Requests for Comments, RFC Editor, RFC 5246, August 2008. [Online]. Available: http://www.rfc-editor.org/rfc/rfc5246.txt

[38] E. Rescorla, "The Transport Layer Security (TLS) Protocol Version 1.3," Internet Requests for Comments, RFC Editor, RFC 8446, August 2018.

[39] M. Corporation, "CVE Details," 2023. [Online]. Available: https://www.cvedetails.com/

[40] MaxMind, "GeoLite2 Free Geolocation," 2023. [Online]. Available: https://dev.maxmind.com/geoip/geolite2-free-geolocation-data

[41] L. Hendriks, R. de Oliveira Schmidt, R. van Rijswijk-Deij, and A. Pras, "On the Potential of IPv6 Open Resolvers for DDoS Attacks," in *Proc. PAM*. Springer, 2017, pp. 17–29.

[42] R. Sundara Raman, P. Shenoy, K. Kohls, and R. Ensafi, "Censored Planet: An Internet-wide, Longitudinal Censorship Observatory," in *Proc. CCS*, 2020, pp. 49–66.

[43] C. Partridge and M. Allman, "Ethical Considerations in Network Measurement Papers," *Communications of the ACM*, vol. 59, no. 10, pp. 58–64, 2016.

[44] P. Ferguson and D. Senie, "Network Ingress Filtering: Defeating Denial of Service Attacks which employ IP Source Address Spoofing," Internet Requests for Comments, RFC Editor, BCP 38, May 2000. [Online]. Available: http://www.rfc-editor.org/rfc/rfc2827.txt

[45] T. Bottger, F. Cuadrado, G. Antichi, E. L. Fernandes, G. Tyson, I. Castro, and S. Uhlig, "An Empirical Study of the Cost of DNS-over-HTTPS," in *Proc. IMC*, 2019, pp. 15–21.

[46] P. Foremski, D. Plonka, and A. Berger, "Entropy/IP: Uncovering Structure in IPv6 Addresses," in *Proc. IMC*, 2016, pp. 167–181.

[47] T. Yang, B. Hou, Z. Cai, K. Wu, T. Zhou, and C. Wang, "6Graph: A Graph-theoretic Approach to Address Pattern Mining for Internet-wide IPv6 Scanning," *Computer Networks*, vol. 203, p. 108666, 2022.

[48] T. Yang, Z. Cai, B. Hou, and T. Zhou, "6Forest: An Ensemble Learning-based Approach to Target Generation for Internet-wide IPv6 Scanning," in *Proc. INFOCOM*. IEEE, 2022, pp. 1679–1688.

[49] G. Song, J. Yang, L. He, Z. Wang, G. Li, C. Duan, Y. Liu, and Z. Sun, "AddrMiner: A Comprehensive Global Active IPv6 Address Discovery System," in *Proc. USENIX ATC*, 2022, pp. 309–326.

[50] J. Zirngibl, L. Steger, P. Sattler, O. Gasser, and G. Carle, "Rusty Clusters? Dusting an IPv6 Research Foundation," in *Proc. IMC*. New York, NY, USA: ACM, 2022.

[51] B. Hou, T. Yang, Z. Cai, K. Wu, and T. Zhou, "Search in the Expanse: Towards Active and Global IPv6 Hitlists," in *Proc. INFOCOM*. IEEE, 2023.

[52] Y. Huang, M. Rabinovich, and R. Al-Dalky, "Flashroute: Efficient Traceroute on a Massive Scale," in *Proc. IMC*, 2020, pp. 443–455.

[53] K. Bock, A. Alaraj, Y. Fax, K. Hurley, E. Wustrow, and D. Levin, "Weaponizing Middleboxes for TCP Reflected Amplification," in *Proc. USENIX Security*, 2021, pp. 3345–3361.

[54] O. Gasser and J. Zirngibl, "zesplot," 2023. [Online]. Available: https://ipv6hitlist.github.io/#zesplot

[55] L. Hendriks, "zesplot: IPv6 visualisation based on squarified treemaps," 2023. [Online]. Available: https://github.com/zesplot/zesplot

[56] M. Bruls, K. Huizing, and J. J. Van Wijk, "Squarified treemaps," in *Proc. Data Visualization*. Springer, 2000, pp. 33–42.

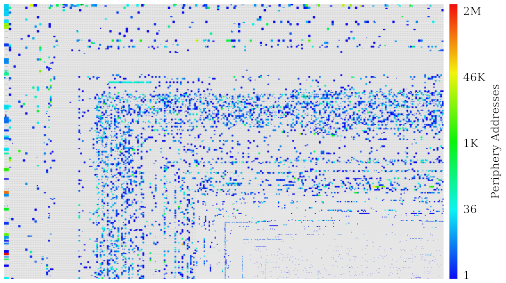
Appendix

In the appendix for address analysis, we integrated the scanning results obtained from each continent-level scan, namely P_{Edgy}, P_{Xmap} and P_{6Seeks}, respectively. Similarly, the results of global-level scans could be represented as G_{Xmap} and G_{6Seeks}.

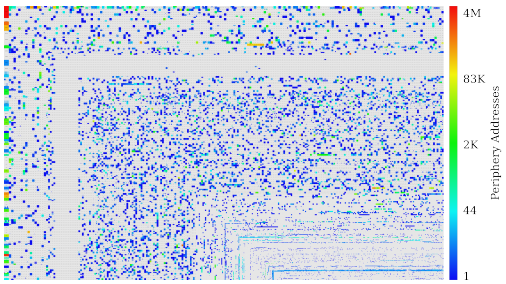
A Major Autonomous Systems

Tab. 7 presents the top 10 Autonomous Systems (ASes) along with the respective count of newly-discovered periphery addresses from each approach across different vantage points. It is evident that a small number of ASes dominated the majority of the addresses, with Xmap standing out as its leading AS contributing to approximately 70% of the discoveries. Both 6Seeks and Edgy show significant AS-level bias in the scanning results based on the vantage point's location, in contrast to Xmap. These biases can be attributed to Internet censorship, which causes probes towards some IPv6 network periphery devices to fail. In this scenario, 6Seeks' dynamic budget allocation focuses on ASes with more valid responses from its current vantage points, leading to a biased distribution of discoveries across ASes.

The concentration of discovered IPv6 network periphery addresses in a few autonomous systems is also observed in Xmap's global-level scanning, as shown in Tab. 7. Approximately half of the addresses discovered by Xmap are located in a single autonomous system, while the discoveries made by our 6Seeks scanner are spread across a wide range of autonomous systems.



(a) Periphery Addresses Discovered in $P_{Xmap} \cup G_{Xmap}$



(b) Periphery Addresses Discovered in $P_{6Seeks} \cup G_{6Seeks}$

Figure 13: All 187k BGP prefixes in RouteViews database, colored based on the number of IPv6 Periphery Addresses.

To further explore it, we integrated IPv6 network peripheral addresses from multiple vantage points and scanning scopes. We visualized the addresses in 22K BGP prefixes from RouteViews using the *zesplot* tool. The *zesplot* [54, 55] is a powerful visualization technique that intuitively represents the distribution of addresses in large IPv6 spaces using the squarified treemaps algorithm [56]. As depicted in Fig. 13, besides having a larger number of discoveries, the results obtained by 6Seeks cover a wider range of BGP prefixes compared to Xmap's results. This demonstrates that the global-scale scanning by 6Seeks provides a more comprehensive understanding of the distribution of IPv6 network peripherals on the Internet.

B Addressing Patterns

As mentioned in § 2.1, the Interface Identifiers of an IPv6 address, which constitute the lower 64 bits, are fully customized by users either manually or automatically. These identifiers are crucial for revealing important information about IPv6 network assets, such as device vendors. The results of addressing pattern classification are presented in Tab. 8.

The majority of the discovered IPv6 periphery addresses in each approach are derived from IPv6 address autoconfiguration methods, namely EUI64, low-byte, and random, accounting for approximately 98% of the total. Specifically, with a DHCPv6 server, the low-byte address is commonly

Table 7: Top 10 Autonomous Systems of Periphery Addresses Discovered By Each Methods

(a) P_{Edgy}

Asia *1063K*	Americas *685K*
AS4*34: 33.65%	**AS8*67:** 16.66%
AS4*12: 14.73%	**AS4*12:** 15.69%
AS5*32: 12.55%	**AS5*07:** 12.89%
AS8*67: 10.32%	**AS5*32:** 10.91%
AS3*03: 3.81%	**AS4*34:** 10.89%
AS3*09: 3.81%	**AS3*09:** 5.93%
AS1*4774: 2.44%	**AS6*61:** 3.42%
AS6*61: 2.19%	**AS3*49:** 1.91%
AS3*62: 1.78%	**AS2*019:**1.38%
AS3*49: 1.20%	**AS7*1:** 1.29%

(b) P_{XMap}

Asia *2351K*	Americas *2353K*
AS1*36: 65.94%	**AS1*36:** 65.86%
AS2*19: 11.29%	**AS2*19:** 11.21%
AS8*22: 4.19%	**AS8*22:** 4.13%
AS4*34: 3.01%	**AS4*34:** 3.07%
AS5*07: 2.74%	**AS5*07:** 2.69%
AS8*99: 0.85%	**AS8*99:** 0.87%
AS5*32: 0.76%	**AS5*32:** 0.75%
AS3*03: 0.66%	**AS3*03:** 0.67%
AS1*037: 0.61%	**AS1*037:** 0.61%
AS4*12: 0.54%	**AS4*12:** 0.55%

(c) P_{6Seeks}

Asia *12938K*	Americas *11517K*
AS5*07: 22.03%	**AS5*07:** 19.57%
AS4*34: 15.73%	**AS4*34:** 17.42%
AS1*36: 12.85%	**AS1*36:** 14.43%
AS5*32: 5.90%	**AS8*81:** 5.62%
AS8*81: 5.63%	**AS5*32:** 5.45%
AS3*03: 5.47%	**AS3*03:** 4.81%
AS4*12: 3.75%	**AS4*12:** 3.99%
AS1*21: 3.50%	**AS1*21:** 3.63%
AS7*38: 3.19%	**AS3*43:** 3.18%
AS3*43: 3.18%	**AS2*19:** 3.08%

(d) G_{Xmap}

Asia *1394K*	Americas *1982K*
AS1*36: 47.16%	**AS1*36:** 47.03%
AS8*22: 12.27%	**AS8*22:** 12.14%
AS2*19: 10.33%	**AS2*19:** 10.34%
AS2*16: 6.29%	**AS2*16:** 6.31%
AS4*64: 3.07%	**AS4*64:** 3.02%
AS3*15: 1.18%	**AS3*15:** 1.11%
AS8*51: 1.14%	**AS8*51:** 1.09%
AS5*07: 0.86%	**AS5*07:** 0.80%
AS4*34: 0.79%	**AS4*34:** 0.73%
AS3*20: 0.75%	**AS3*20:** 0.70%

(e) G_{6Seeks}

Asia *7978K*	Americas *13343K*
AS1*36: 21.22%	**AS6*39:** 19.95%
AS6*39: 12.12%	**AS1*36:** 12.93%
AS1*235: 11.47%	**AS3*15:** 7.87%
AS8*22: 6.19%	**AS5*07:** 5.95%
AS3*15: 5.75%	**AS3*20:** 4.44%
AS2*19: 4.58%	**AS4*34:** 4.32%
AS5*07: 4.49%	**AS8*22:** 4.03%
AS2*16: 2.67%	**AS2*19:** 2.90%
AS4*34: 2.66%	**AS8*81:** 2.13%
AS3*20: 2.44%	**AS2*16:** 1.60%

recommended for the IPv6 hosts through DHCPv6 Advertise messages [14]. Without a DHCPv6 server, IPv6 network periphery devices obtain unique addresses using Stateless Address Autoconfiguration (SLAAC) [15].

For example, periphery devices incorporate their MAC addresses into Interface Identifiers to generate EUI64 addresses, which are widely used in IoT devices and home gateways. Additionally, to enhance privacy, IPv6 periphery devices also utilize SLAAC privacy extensions to randomly select bits for Interface Identifiers, thus preventing tracking by providers. Therefore, these findings align with the IPv6 periphery model as illustrated in Fig. 1.

C Biases of Vantage Point Locations

The scanning results clearly indicate the presence of measurement biases associated with the vantage point locations, despite the limited availability of only two vantage points. Generally, 6Seeks and Xmap found 60% and 40% more IPv6

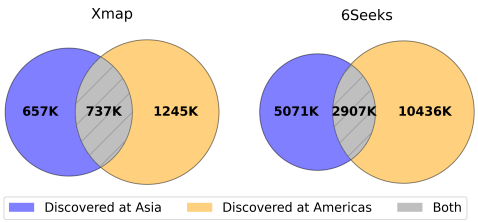
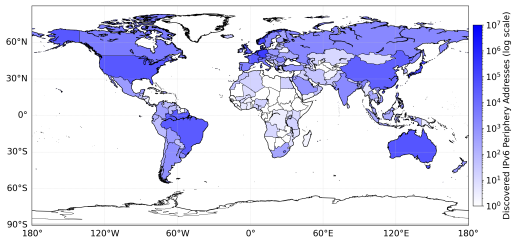
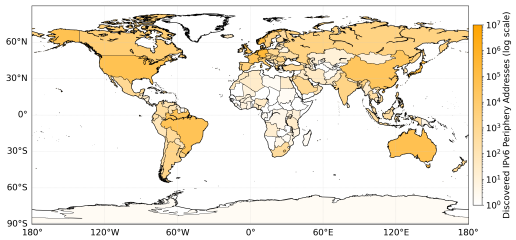


Figure 14: Venn diagram for IPv6 periphery addresses on global-level scanning at Asia and Americas.

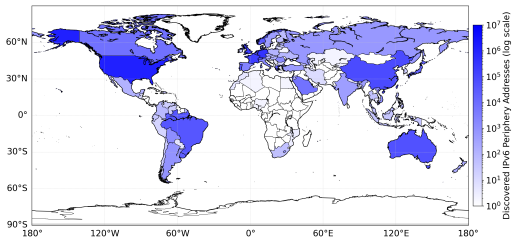


(a) Vantage Point at Asia

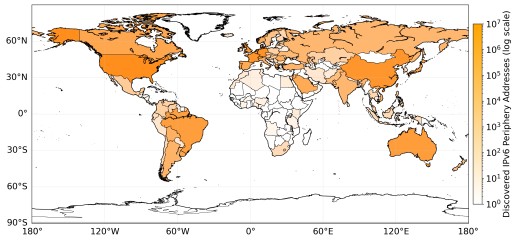


(b) Vantage Point at Americas

Figure 15: Geographical distribution of IPv6 periphery addresses in global-level scanning results of Xmap



(a) Vantage Point at Asia



(b) Vantage Point at Americas

Figure 16: Geographical distribution of IPv6 periphery addresses in global-level scanning results of 6Seeks

Table 8: Addressing Patterns of IPv6 Periphery Addresses Discovered by Each Methods

Addressing Pattern	P_{Edgy}		P_{Xmap}		P_{6Seeks}		G_{Xmap}		G_{6Seeks}	
	VP at Asia	*VP* at Americas	*VP* at Asia	*VP* at Americas	*VP* at Asia	*VP* at Americas	*VP* at Asia	*VP* at Americas	*VP* at Asia	*VP* at Americas
EUI64	408K (38.34%)	255K (37.23%)	542K (23.04%)	540K (22.94%)	4112K (31.78%)	3630K (31.52%)	456K (32.75%)	586K (32.24%)	2473K (31.00%)	4694K (35.18%)
Embed-IPv4	4670 (0.44%)	4624 (0.67%)	20921 (0.89%)	21125 (0.90%)	179K (1.38%)	87293 (0.76%)	29339 (2.11%)	41068 (2.26%)	142K (1.78%)	328K (2.45%)
Low-byte	114K (10.76%)	185K (27.06%)	595K (25.30%)	595K (25.31%)	4574K (35.35%)	3894K (33.82%)	335K (24.03%)	451K (24.80%)	1347K (16.88%)	2214K (16.59%)
Byte-pattern	689 (0.06%)	2259 (0.33%)	1668 (0.07%)	1938 (0.08%)	4566 (0.04%)	2763 (0.02%)	5104 (0.37%)	7242 (0.40%)	19934 (0.25%)	45392 (0.34%)
Random	535K (50.31%)	236K (34.46%)	1187K (50.49%)	1190K (50.57%)	4065K (31.42%)	3897K (33.84%)	566K (40.63%)	730K (40.15%)	3993K (50.04%)	6059K (45.41%)

periphery addresses in Americas, respectively, compared to Asia. This is primarily attributed to the higher responsiveness of most Autonomous Systems to probes from vantage points in Americas. In this scanning, Americas had 6097 and 5426 involved ASes, while Asia had 5351 and 5129 involved ASes (see Tab. 9). Furthermore, Tab. 7 illustrates that vantage points in Americas tend to uncover more addresses of IPv6 network periphery addresses, even within same autonomous systems.

Additionally, there is hardly any intersection among the triggering addresses from different vantage points as they are randomly generated. Nevertheless, Xmap and 6Seeks have disclosed a significant number of shared periphery addresses in both Asia and Americas, as shown in Fig. 14. This finding further confirms the effectiveness of IPv6 periphery discovery.

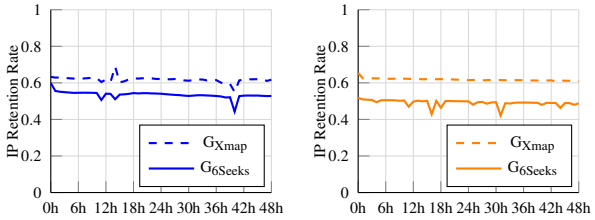
To visually demonstrate the measurement bias caused by vantage point locations, we present the geographical distribution of Xmap's and 6Seeks' global scanning results from multiple vantage points, as depicted in Fig. 15 and Fig. 16. According to 6Seeks' results, the majority of periphery addresses discovered at both vantage points are located in major world economies. Conversely, the vantage point in Americas outperforms in discovering network periphery addresses in other regions of the world.

D IP Churn

It is imperative to take into account the IP churn in large-scale scanning, including the Internet-wide discovering of IPv6 periphery. Specifically, the discovered IPv6 addresses experience churn over time after the scanning process. Maintaining a relatively low churn rate is crucial for ensuring the reliability of subsequent security applications.

To this end, we evaluate the retention rate (opposite of churn rate) within a 48-hour timeframe on the Internet-wide periphery discoveries of Xmap and 6Seeks, respectively.

As shown in Fig. 17, the ICMPv6 Echo Request/Reply is employed to validate IP retention once an hour. The retention rates of Xmap are comparatively higher than those of 6Seeks because 6Seeks has a much larger number of peripheral addresses and is, therefore, highly affected by IP churn [5]. Nevertheless, even after a period of 48 hours, 6Seeks still retains around 11 million peripheral addresses in its global-level scanning results, whereas Xmap retains just 2 million peripheral addresses in its global-level scanning results. Furthermore, considering the biases from ICMPv6 Echo Request/Reply

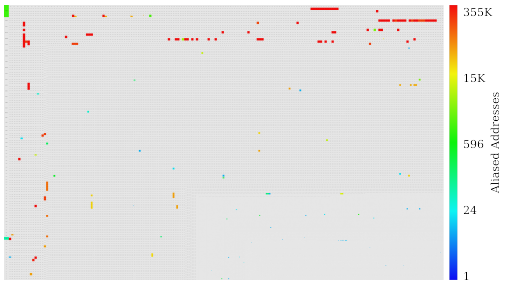


(a) Vantage Point at Asia (b) Vantage Point at Americas

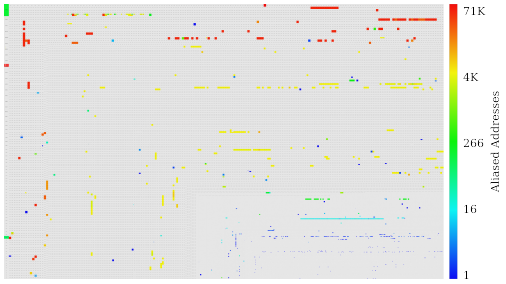
Figure 17: IP retention rates in global-level periphery scanning by 6Seeks and Xmap.

(where numerous network periphery devices fail to respond directly), the retention rates of 6Seeks and Xmap should have been higher in ground truth.

E Aliased Prefixes



(a) Aliased Addresses in G_{Xmap}



(b) Aliased Addresses in G_{6Seeks}

Figure 18: All 187k BGP prefixes in RouteViews database, colored based on the number of newly-discovered IPv6 aliased addresses.

For traceroute-like methods (Edgy), we utilize the list of aliased prefixes provided by Gasser [34, 54] for IPv6 alias

Table 9: Billion-scale Scanning Results On Each Range At Maximum Probing Rate of 100 Kpps

Scanning Range	Approach	VP Location	Alias Number	Discovery Number	Probing Packets[a]	Involved ASes	Hit Rate (‰)[b]	Time Cost	Probing Speed	Scann. Eff. (Num. per Sec.)
2001::/16	Edgy	Asia	158	224561	1.302e9	420	0.172	03:47:10	98.20Kpps	16.47
		Americas	73	215761	1.302e9	472	0.166	23:58:13	15.22Kpps	2.50
	Xmap	Asia	91712	418400	1.000e9	869	0.418	02:48:25	98.99Kpps	41.40
		Americas	92098	415534	1.000e9	897	0.416	04:56:07	56.30Kpps	23.63
	6Seeks	Asia	85780	2596251	1.000e9	927	**2.596**	02:48:10	99.10Kpps	**258.92**
		Americas	75834	2437789	1.000e9	932	**2.438**	03:42:26	74.92Kpps	**183.59**
240e::/16	Edgy	Asia	13	561477	1.027e9	55	0.547	03:03:24	95.90Kpps	51.02
		Americas	8	194755	1.027e9	36	0.190	19:38:22	14.67Kpps	2.75
	Xmap	Asia	4	87723	1.000e9	59	0.088	02:48:40	98.85Kpps	8.67
		Americas	6	90236	1.000e9	73	0.090	04:55:58	56.33Kpps	5.08
	6Seeks	Asia	35	2607948	1.000e9	79	**2.608**	02:50:10	97.94Kpps	**255.43**
		Americas	148	2555851	1.000e9	70	**2.556**	03:06:03	89.58Kpps	**228.96**
2600::/16	Edgy	Asia	481	16727	1.001e9	64	0.017	02:57:25	96.73Kpps	1.57
		Americas	280	14650	1.001e9	58	0.015	20:08:52	13.94Kpps	0.20
	Xmap	Asia	18794	11874	1.000e9	129	0.012	02:48:20	99.04Kpps	1.18
		Americas	19244	11858	1.000e9	132	0.012	04:52:01	57.09Kpps	0.68
	6Seeks	Asia	17426	23157	1.000e9	132	**0.023**	02:49:02	98.59Kpps	**2.28**
		Americas	19142	23297	1.000e9	134	**0.023**	03:08:20	88.49Kpps	**2.06**
2804::/16	Edgy	Asia	0	16476	1.089e9	492	0.015	03:13:30	96.45Kpps	1.42
		Americas	0	16024	1.089e9	581	0.015	17:49:01	17.18Kpps	0.26
	Xmap	Asia	92	89212	1.000e9	2816	0.089	02:48:07	99.18Kpps	8.84
		Americas	96	93042	1.000e9	2825	0.093	04:49:54	57.49Kpps	5.35
	6Seeks	Asia	144128	872146	1.000e9	2701	**0.872**	02:52:38	96.54Kpps	**84.20**
		Americas	113773	845393	1.000e9	2706	**0.845**	03:23:34	81.87Kpps	**69.22**
2a02::/16	Edgy	Asia	0	244166	1.481e9	107	0.165	04:18:18	98.32Kpps	15.75
		Americas	4	242437	1.481e9	136	0.164	28:18:20	14.70Kpps	2.38
	Xmap	Asia	2430231	1743285	1.000e9	541	1.743	02:48:09	99.16Kpps	172.79
		Americas	2427564	1741326	1.000e9	561	1.741	05:38:36	49.23Kpps	85.71
	6Seeks	Asia	1404428	6838737	1.000e9	525	**6.839**	02:50:44	97.61Kpps	**677.58**
		Americas	1104817	5653484	1.000e9	512	**5.653**	03:46:08	73.70Kpps	**416.68**
2c0f::/16	Edgy	Asia	0	148	1.001e9	39	1.47e-4	03:59:56	70.96Kpps	0.01
		Americas	6	1567	1.001e9	53	**1.56e-3**	22:07:53	12.56Kpps	0.02
	Xmap	Asia	5637	971	1.000e9	118	9.71e-4	02:48:12	99.12Kpps	**0.10**
		Americas	5620	980	1.000e9	123	9.80e-4	05:33:59	49.91Kpps	0.05
	6Seeks	Asia	2772609	981	1.000e9	112	**9.81e-4**	02:54:46	95.36Kpps	0.09
		Americas	2464847	1092	1.000e9	137	1.09e-3	03:28:07	80.08Kpps	**0.09**
Global	Xmap	Asia	92242700	1393674	1.000e10	5129	0.139	27:49:50	99.81Kpps	13.91
		Americas	97503632	1982242	1.000e10	5426	0.198	61:49:16	44.93Kpps	8.91
	6Seeks	Asia	33516882	7978299	1.000e10	5351	0.789	29:04:14	95.54Kpps	76.235
		Americas	3695e482	13343590	1.000e10	6097	**1.334**	34:44:42	79.92Kpps	**106.78**

[a]: In experiments, we do not impose the restrictions on the number of probing packets used in traceroute-like methods.
[b]: For reference only.

resolution. For the ping-like tools Xmap and 6Seeks, the basic hypothesis (see § 3.1) suggests that only aliased prefixes can respond directly to the probes with randomized Interface Identifiers. Accordingly, numerous aliases are disclosed during the large-scale IPv6 periphery discovery processes of Xmap and 6Seeks.

To confirm the reality of the identified aliased addresses, we cross-validate them with the known aliased prefixes. The results show that approximately 36.66% of the aliased addresses found by Xmap and 37.11% of those found by 6Seeks (i.e., the addresses that respond directly to ICMPv6 Echo Requests with ICMPv6 Echo Replies) during the global-level scanning are located at known aliased prefixes. Furthermore, with the APD detection algorithm [34], among the newly-discovered aliased addresses in the results of Xmap and 6Seeks, we identified 105,930 and 30,012 `/48` IPv6 alias prefixes, respectively, which have never been reported before.

Fig. 18 presents the visualization of the newly-discovered aliased addresses using the *zesplot* tool. The results from 6Seeks are widely spread across more BGP prefixes than Xmap's results, even though 6Seeks has less aliased addresses (kindly recall 6Seeks's interaction-based mechanism for avoiding the "alias trap"), which might promote the exploration of IPv6 aliased prefixes.

In a nutshell, the implementation of 6Seeks can greatly enhance researchers' comprehension of network dynamics, bolster network security measures, and actively contribute to the advancement of a sturdy and dependable Internet infrastructure.