Follow the Scent: Defeating IPv6 Prefix Rotation Privacy

Erik Rye
CMAND
rye@cmand.org

Robert Beverly
Naval Postgraduate School
rbeverly@nps.edu

kc claffy
CAIDA/UC San Diego
kc@caida.org

ABSTRACT

IPv6's large address space allows ample freedom for choosing and assigning addresses. To improve client privacy and resist IP-based tracking, standardized techniques leverage this large address space, including privacy extensions and provider prefix rotation. Ephemeral and dynamic IPv6 addresses confound not only tracking and traffic correlation attempts, but also traditional network measurements, logging, and defense mechanisms. We show that the intended anti-tracking capability of these widely deployed mechanisms is unwittingly subverted by edge routers using legacy IPv6 addressing schemes that embed unique identifiers.

We develop measurement techniques that exploit these legacy devices to make tracking such moving IPv6 clients feasible by combining intelligent search space reduction with modern high-speed active probing. Via an Internet-wide measurement campaign, we discover more than 9M affected edge routers and approximately 13k /48 prefixes employing prefix rotation in hundreds of ASes worldwide. We mount a six-week campaign to characterize the size and dynamics of these deployed IPv6 rotation pools, and demonstrate via a case study the ability to remotely track client address movements over time. We responsibly disclosed our findings to equipment manufacturers, at least one of which subsequently changed their default addressing logic.

CCS CONCEPTS

• **Networks → Network measurement**; **Network privacy and anonymity**.

KEYWORDS

IPv6, prefix rotation, EUI-64 privacy

ACM Reference Format:
Erik Rye, Robert Beverly, and kc claffy. 2021. Follow the Scent: Defeating IPv6 Prefix Rotation Privacy. In *ACM Internet Measurement Conference (IMC '21), November 2–4, 2021, Virtual Event, USA.* ACM, New York, NY, USA, 14 pages. https://doi.org/10.1145/3487552.3487829

1 INTRODUCTION

IPv6 deployment has seen rapid growth in recent years, with residential broadband being one significant driver [23]. The large IPv6 address space affords networks and clients considerable flexibility in allocating and using addresses, as well as posing unique "needle-in-the-haystack" challenges to network measurement. For instance, best practices for IPv6 dictate a /64 allocation or larger [21] – implying that a single residential customer has more IPv6 addresses than the entire IPv4 address space. Within their allocated prefix, client addresses can change regularly as clients choose random and ephemeral addresses via IPv6 *privacy extensions* [20], a standard to resist adversarial tracking and correlation. As we show in this work, some providers make use of the large IPv6 address space to further improve privacy by also, periodically and regularly, changing the entire prefix allocated to customers [19] – a process we term *prefix rotation.*

In this work, we develop techniques that allow us to mount an Internet-wide active campaign to measure the prevalence of IPv6 prefix rotation. At the heart of our approach is the surprisingly common presence of deployed Customer Premises Equipment (CPE), i.e., routers inside customer's homes, that use a legacy addressing standard employing EUI-64 [28]. In this form of IPv6 addressing, the lower 64 bits of the CPE's address, the Interface Identifier (IID), are unique, fixed, and persistent. By sending active probes into candidate network pools, we elicit IPv6 responses containing the embedded IID from these CPE. Our insight is to exploit these legacy static IIDs to measure and characterize the real-world operational deployment of prefix rotation in terms of provider IPv6 prefix pool sizes, delegation sizes, and rotation strategies.

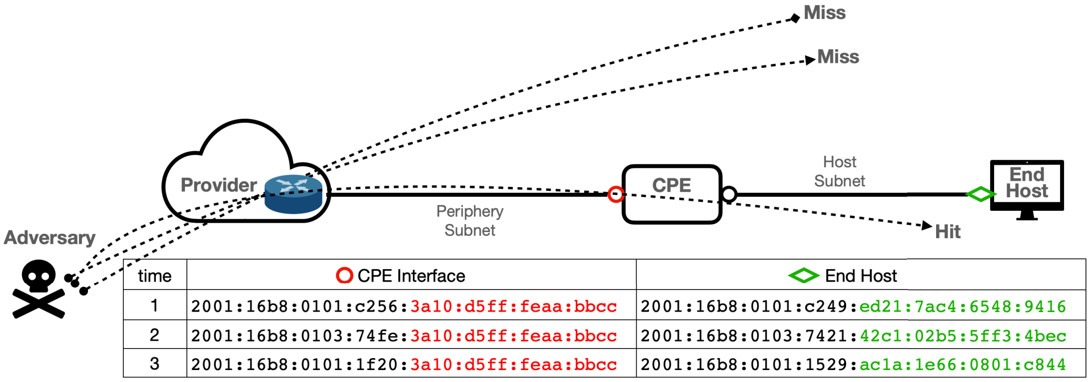
A second consequence of CPE using static EUI-64 IIDs is the potential to track clients as their prefix rotates. The primary obstacle to finding a client's new prefix after a rotation is the very large IPv6 search space; it is impractical to probe the entirety of a provider's prefix, which is often as large as a /32, i.e., 2^{96}. To make tracking attacks feasible, we combine high-speed active probing [12, 30] and extend recent innovations in IPv6 topology discovery [27]. With our novel technique, we demonstrate how a remote third-party can efficiently track a residential IPv6 client with affected CPE, even as their entire address, including its prefix, changes over time.

Next, we mount a large-scale Internet-wide active measurement campaign to characterize the scope and extent of IPv6 prefix rotation in the wild by probing ~18k /48 prefixes daily. Our measurements discover 110M unique EUI-64 addresses assigned to CPE using 9M distinct IIDs, i.e., the same IID is seen in multiple prefixes – demonstrating that IIDs are moving between prefixes in practice. Such prefix rotation is evident in approximately 100 ASes across 25 different countries. While these confirmed rotators are a small fraction of all IPv6 ASes, they represent a lower-bound of at least 9M Internet users.

Not only does our work shed light on this previously understudied aspect of production IPv6 deployment, it highlights the fact that a non-trivial fraction of deployed CPE unintentionally subvert IPv6 privacy mechanisms by permitting users and networks to be tracked by proxy. In sum, our contributions include:

IMC '21, November 2–4, 2021, Virtual Event, USA

ACM ISBN 978-1-4503-9129-0/21/11…$15.00
https://doi.org/10.1145/3487552.3487829



time	○ CPE Interface	◇ End Host
1	2001:16b8:0101:c256:3a10:d5ff:feaa:bbcc	2001:16b8:0101:c249:ed21:7ac4:6548:9416
2	2001:16b8:0103:74fe:3a10:d5ff:feaa:bbcc	2001:16b8:0103:7421:42c1:02b5:5ff3:4bec
3	2001:16b8:0101:1f20:3a10:d5ff:feaa:bbcc	2001:16b8:0101:1529:ac1a:1e66:0801:c844

Figure 1: In IPv6, the CPE (e.g., home router) is a routed hop and both the CPE and end host have public addresses. The endhost's address is ephemeral: the lower 64 bit IID changes (privacy extensions) while the upper 64 bit prefix changes (provider rotation). Despite this, an attacker can track unsuspecting users when their home router uses a static lower 64 bits (in red). We show how an adversary can efficiently send active probes to find a CPE, thereby revealing these users.

- A measurement technique to identify providers that implement IPv6 prefix rotation, and large-scale measurements to discover and characterize these allocations (§3).
- Demonstration that currently deployed IPv6 anti-tracking and anti-correlation mechanisms (ephemeral addresses and prefix rotation) can be rendered moot by CPE that implement legacy standards (§3).
- Internet-wide measurements showing that this problem affects >9M customers across dozens of networks worldwide (§4).
- Per-network CPE homogeneity analysis, revealing that half of 87 studied networks contain a single manufacturer that dominates ≥80% of their deployment (§5).
- A case study demonstrating the ability to correlate seemingly unrelated IPv6 traffic flows over time using our methods with 60-90% accuracy (§6).
- Vulnerability remediation via communication with a major CPE vendor (§8).

The remainder of this paper is organized as follows. Section 2 provides background on privacy-relevant features and standards in IPv6, and explores three different security and privacy threats and harms enabled by CPE with EUI-64 addressing. Next, Section 3 describes our active measurement method to bound, per-provider, the space of possible locations to which a CPE and its hosts may have moved. We show that this vulnerability affects millions of devices and customers in Section 4 and detail the vulnerable ecosystem in Section 5. Section 6 demonstrates the real-world feasibility of our approach in the wild via a case study to track CPE and hosts over time. Section 7 describes the ethical guidelines adhered to during this study. Finally, we detail our experience with an equipment manufacturer in Section 8 and conclude with recommendations for long-term remediation.

2 BACKGROUND AND RELATED WORK

2.1 Background

Operational security and privacy issues of IPv6 have been well cataloged [11]. Of particular relevance to our work are three IPv6 mechanisms: 1) SLAAC addresses; 2) privacy extensions; and 3) prefix rotation.

- **SLAAC addresses**: When using EUI-64 Stateless Address Autoconfiguration (SLAAC) addresses [28], an endhost forms its full 128 bit IPv6 address by combining the provider-assigned prefix with 64 least-significant bits (the IID or "interface identifier") that are derived from the interface's IEEE hardware MAC address. This feature ensures address uniqueness but exposes to layer-3 the host's layer-2 information, which allows the host to be trivially tracked across network changes.
- **Privacy extensions**: IPv6 clients commonly use *privacy extension* addressing, wherein the client chooses a random, ephemeral lower 64-bit IID [20] in order to resist IP-based tracking and correlation. As per the RFC, "changing the interface identifier over time makes it more difficult for eavesdroppers and other information collectors to identify when different addresses used in different transactions actually correspond to the same node." Thus, whereas multiple hosts behind a NAT may map to one public IPv4 address, a single residential host may use many different public IPv6 addresses. Indeed, a large CDN found that more than 90% of IPv6 addresses appear only once in a long-running data collection [25].
- **Prefix rotation**: While privacy extensions protect clients when changing networks, IP-based tracking is still possible via the customer's assigned prefix [15]. Some providers additionally deploy "temporary-mode" DHCPv6 [19] in order to regularly and periodically change the prefixes allocated to customers. Such *prefix rotation* is intended to prevent tracking – now both the IID (lower 64 bits) and the prefix (high 64 bits) change, often daily.

In IPv6, the CPE, e.g., the cable modem in a customer's home, is a routed hop. As shown in Figure 1, between the provider and the CPE is the *periphery subnet*, while a second *host subnet* for the LAN may be part of the periphery subnet, or distinct. When probing toward any target address within a customer host subnet, active

IPv6 topology discovery elicits a response from the CPE, the red colored interface in Figure 1.

Whereas IPv6 privacy extensions are enabled by default and used today in all modern desktop and mobile operating systems, EUI-64 addresses are still commonly used in deployed CPE, likely because many CPE run embedded, old, unmaintained, or proprietary operating systems. For example, Figure 1 shows the CPE obtaining new prefixes over three points in time. Note that while the addresses remain in the provider's allocated prefix (`2001:16b8::/32`), the periphery prefix within the rotation pool changes for each sample and the EUI-64 IID remains constant (in this example, corresponding to the CPE's MAC `38:10:d5:aa:bb:cc`).

2.2 Related Work

Recent large-scale IPv6 traceroute measurements discovered 30M EUI-64 last hop addresses in the Internet [27] corresponding to CPE interfaces facing the upstream provider. While Rye's study found 30M distinct IPv6 addresses with EUI-64 IIDs, only 16M of those IIDs were unique, implying that the periphery subnet was changing and using prefix rotation [27]. While [27] developed an algorithm for general IPv6 periphery discovery, this work explores how the widespread use of legacy EUI-64 addresses by CPE effectively removes endhost privacy enhancements offered by prefix rotation and ephemeral addresses. Other active measurement studies have explored IPv6 topology discovery more generally, rather than exclusively on periphery discovery, such as Beverly et al. [8] and Gasser et al. [13].

Related to our study are efforts to perform address-based blocking of infected or abusive IPv6 hosts. The same privacy-enhancing mechanisms designed to protect users serve to complicate efforts by e.g., content providers, to block, filter, or rate-limit attacks by IPv6 address or prefix [17]. Our work sheds light on the practical difficulties of protecting IPv6 networks from attack, provides new insights on address lifetimes induced by prefix rotation, and provides a potential means to track and correlate attack traffic as part of a defensive mechanism. Although recent work [24] examines IPv6 address lifetimes using several thousand privileged vantage points located within customer networks [22], we demonstrate that an unprivileged attacker can infer the same information for orders of magnitude more networks using active probing.

3 TRACKING ENDHOST SUBNETS

Discovering the instantaneous binding between a particular, known IPv6 endhost and its CPE is straightforward: simply run a traceroute toward that destination IPv6 address to induce the CPE to return an ICMPv6 hop limit (née TTL) exceeded message. Since the source IPv6 address of this ICMPv6 message is the interface on the route toward the original probe [10], the response reveals the CPE's WAN address, which may contain an EUI-64 IID. Tracking a user by her CPE's static EUI-64 IID, after both her periphery prefix and IID change, requires discovering an IPv6 traceroute destination that responds with this CPE's EUI-64 IID. *Finding* the correct destination that will reveal the binding, amid the vast IPv6 address space, is hard and the crux of the problem we tackle.

At first blush, this search space appears impractically large: the lower 64 bits are all changed and the new subnet may be anywhere in the provider's global prefix. Since many providers have an allocated (from their Regional Internet Registry) address prefix of /32 or larger, a brute force search would require $2^{128-32} = 2^{96}$ probes – a size impossible to probe. We make this probing feasible via high-speed active probing (§3.1) and intelligent search space reduction (§3.2).

Our methodology has two phases: 1) determining the size of customer prefix allocation for each provider (§3.2.1); and 2) finding the rotation pool to which the customer belongs (§3.2.2). We describe our active probing technique to inferentially perform each of these tasks next.

3.1 Active Probing

Prior work, e.g., [8, 27], utilized yarrp [7] to perform high-speed active IPv6 topology mapping. While yarrp is ideal for full topology probing (i.e., finding the sequence of router interfaces along a path to a destination), we are not interested in the entire forwarding path, only the last hop toward an IPv6 destination. The IIDs of hosts within the customer prefix are ephemeral and unknown. However, the recommended IPv6 behavior is for the last hop to return an ICMPv6 unreachable message when an address probed at random within the prefix does not exist [10]. This ICMPv6 message exposes the CPE's address facing the provider, and requires only a single probe into the entire customer prefix allocation. That is, we need not select a responsive endhost in order to induce a response from the CPE responsible for the host subnet.

Throughout this work, we utilize the zmap6 IPv6 extensions [30] to the high-speed zmap prober [12] developed by Gasser et al. We identify networks that exist internal to the CPE (the "host subnet" in Figure 1), and send ICMPv6 Echo Request probes to random IIDs in these host subnets. We probe at 10k packets per second from a well-connected vantage point in a European IXP.

Sending probes to the host subnet typically generates one of several ICMPv6 messages. Generally, these responses are *Destination Unreachable* errors (*Administratively Prohibited*, *No Route to Destination*, and *Address Unreachable* are common), but we also observe *Hop Limit Exceeded* responses as well. These type and code combinations indicate different OS behaviors when receiving a packet destined for a nonexistent host in the internal subnet. The particular response type itself does not matter in our application; all of these messages reveal the source address of the CPE in reply. Mistaking responses from intermediate hops as CPE is unlikely: managed network infrastructure is typically statically addressed, while we focus specifically on EUI-64 IPv6 addresses. Furthermore, the Media Access Control (MAC) addresses embedded in the EUI-64 addresses we discover align with major CPE vendors' Organizationally Unique Identifiers (OUIs).

yarrp may elicit a response in some instances that zmap does not, e.g., because of particular ICMPv6 filtering policies. However, zmap more unambiguously reveals the IPv6 periphery because intermediate hops do not reply with *Hop Limit Exceeded* ICMPv6 messages as they do in yarrp. Reduced ICMPv6 error messages also helps avoid potential ICMPv6 response rate limiting [8]. Because we are interested only in periphery (CPE) responses, as opposed to responses from core and access infrastructure, using zmap allows

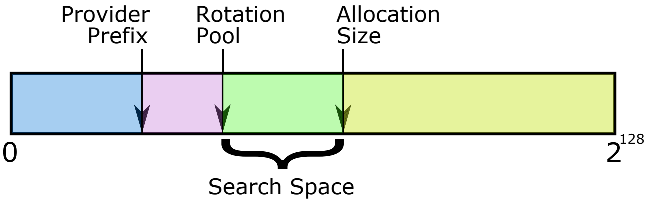


Figure 2: Limiting the search space to track IPv6 hosts: for a provider, we infer the: i) size of the allocations to customers; and ii) range of prefixes used for rotation.

us to probe at a higher rate than yarrp without receiving responses from portions of the Internet topology tangential to our study.

3.2 Bounding the Search Space

Bounding the search space requires understanding the size of prefix allocated to customers (for instance a /64 versus a /56), the size of the rotation pool (for instance bits 33-64 in Figure 1), and potentially the rotation pool identifier (for providers that have more than one address pool). A key challenge is that there is no standard or operational consistency in IPv6 deployment, hence each of these variables *differs by provider.* A significant contribution of our work is a large-scale Internet measurement campaign to empirically characterize provider IPv6 deployments and allocation behaviors in the wild.

As depicted in Figure 2, the size of prefix that a provider allocates to customers bounds the search space from above, while the rotation pool size bounds the search space from below. Thus, rather than searching an impossibly large space, we need only probe within the green shaded area to scalably locate a particular CPE.

In our canonical example (Figure 1), both the CPE interface and endhost interface change over three time steps. The adversary knows that a binding exists between the endhost and the MAC address of their CPE. Here, the search space is reduced to a rotation pool of a /46, while the customer prefix allocation size is a /64. The adversary thus probes random targets for each /64 within this /46 range until it elicits a response from a CPE containing the correct EUI-64 IID ($E[] = 2^{18-1}$ probes, or about 13 seconds at 10kpps). This example depicts two probes that pass through the provider's network but miss the CPE, while a third probe reaches the CPE and therefore returns a response that reveals the CPE's new address to the attacker.

3.2.1 Customer Prefix Allocation Size Discovery. We seek to better understand *how* providers allocate address space to their customers. RFC 6177 [21] recommends that providers allocate to customers prefixes ranging in size from /48 to /64, but leaves the decision up to individual service providers. Because we target networks *internal* to CPE, when our probes are unable to be routed by the CPE because of administrative policy or because the host to which they are addressed does not exist, the CPE replies with an ICMPv6 error that identifies it as responsible for that network (§3.1).

Because EUI-64 SLAAC addressing requires a subnet no smaller than /64, /64 networks are the smallest recommended allocation size [16, 21] for end sites. Thus, when we probe a target within each constituent /64 subnet of a rotation pool, we infer the size of the host subnet internal to the CPE by observing contiguous probed /64 networks that elicit a response from the same address.

For example, consider a /48 network from which a provider allocates /56 subnets to customers. A CPE device, if responsive to probes addressed to destinations in its host subnet, will originate ICMPv6 error messages if those destinations cannot be reached. Many devices will prohibit traffic to internal addresses that was not initiated from the host subnetwork, and it is exceptionally unlikely to guess a device's address behind the CPE; therefore, in practically all cases the probe traffic will be undeliverable. Thus, for each of the $2^{64-56} = 256$ probes with random IIDs sent to each /64 within the /56 assigned to the CPE device, the CPE will originate an error message with its external IPv6 address (red interface in Figure 1) as the source. By observing consistent source addresses in replies for contiguous ranges of target addresses, we are able to infer how the provider partitions its address space for customer prefix assignments.

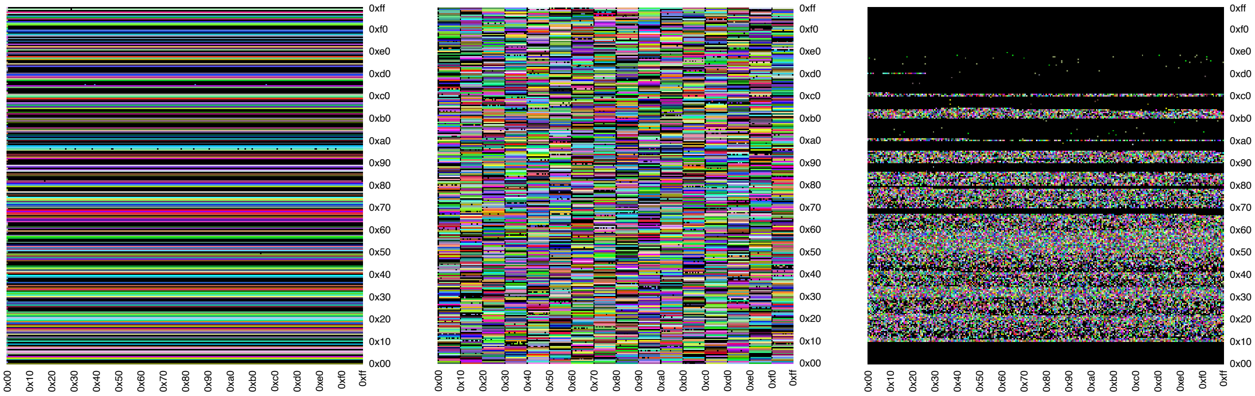
To illustrate the methodology, we characterize three different providers via active probing on September 7, 2020. Figure 3 exposes three distinct allocation strategies deployed in practice. Figure 3a represents a /48 prefix belonging to Entel [4], a Bolivian telecommunications company. Each horizontal colored band represents a different address responding to our probes. Colors are selected only to allow for visual differentiation and to show contiguous regions; the particular color is not significant in these figures. The y-axis represents the 7^{th} byte of the probe destination address, and x-axis represents the 8^{th} byte of the destination address, i.e., the last two octets of the high 64 bits.

The consistent response address for targets with a fixed 7^{th} byte, while varying the 8^{th} byte (shown by the color banding), indicates that the provider in Figure 3a allocates /56 prefixes. The black horizontal bands, interspersed throughout the /48, are /56 networks for which we received no response. These /56s may be unallocated by the provider, or the devices assigned to those /56s may silently drop our probes.

Figure 3b shows the same analysis from our probing of a Bosnian service provider's /48 (BH Telecom [3]). BH Telecom allocates /60 networks to its customers, which are depicted as 16 short, horizontal lines within each /56. Some /60s are unallocated or the associated CPE is unresponsive to our probes, as evidenced by the contiguous black lines interspersed throughout. Some /60s exhibit black pixels within them, indicating potential packet loss or the CPE device not responding to probes.

Finally, Figure 3c is a /48 prefix delegated to a third provider, Starcat [6], a Japanese ISP. The customer-allocated subnets within this /48 are /64s, depicted by the heavily-pixelated portions of the Figure. Of the three sub-figures in Figure 3, Figure 3c exhibits the largest unresponsive portions, with significant space in the upper quarter of the /48 prefix that did not respond to our probes, likely because it was unallocated to any CPE.

Algorithm 1 provides pseudocode of our technique to infer an AS's prefix allocation sizes. For each Autonomous System (AS), we calculate the median inferred allocation size from all the EUI-64 IIDs' inferred allocation sizes. Service providers may have one or more prefix allocation sizes; providers that offer different classes of



(a) Entel (Bolivia): /56 allocations

(b) BH Telecom (Bosnia): /60 allocations

(c) Starcat (Japan): /64 allocations

Figure 3: An adversary can infer providers' customer address allocation policies, enabling efficient scanning for targeted tracking (§6). The y-axis of plots represents the 7^{th} byte of a probed address, while the x-axis denotes the 8^{th} byte; each pixel represents a probed /64 network. Each color represents a different responsive source address, while black indicates no response was received when probing to an address in that /64 network.

service or differentiate between customer types exhibit distributions with clusters around multiple prefix sizes.

Inferring the customer prefix allocation size allows an adversary to reduce the number of probes necessary to enumerate the CPE devices present within a prefix. For instance, rather than sending a probe to an address in each /64 within the Entel /48 in Figure 3a, a tracker that knows a priori that Entel allocates /56 networks to customers need only send 256 probes (one to an address in each /56) in order to elicit a response from each CPE, decreasing probing cost by 99.6%. In contrast, the Starcat /48 in Figure 3c requires an adversary to probe each /64, as /64s are allocated to CPE devices.

3.2.2 Identifying Rotation Pools. Next, we seek to determine the size of the rotation pools within which allocated customer prefixes rotate per provider. Across all EUI-64 addresses discovered in a provider, we first find the maximum numeric distance between any two /64 periphery prefixes containing that EUI-64. We then compute the median across all per-EUI-64 ranges to determine the likely pool size for the provider as a whole. Pseudocode for this inference is provided in Algorithm 2 in the Appendix. Algorithm 2 is essentially identical to Algorithm 1; the two algorithms differ in that the allocation size inference requires a $< response\ address, target\ address >$ mapping and uses target addresses to infer the LAN network size, while the rotation pool size calculation requires only the set of responses addresses and uses these to calculate the distance that periphery addresses can travel in an AS. Rotation pool size inference can be complicated by devices rotating out of our probe window, leading to missing observations and erroneously inferring a smaller rotation pool size than exists in reality.

An EUI-64 IID's rotation pool is bounded by the entirety of the provider's IPv6 allocation (assuming the CPE owner does not change providers), and a /64, meaning that the CPE IID was in the same prefix throughout the duration of our study. This does not preclude a rotation interval longer than our measurement campaign, however.

4 DISCOVERING PREFIX-ROTATING PROVIDERS

Our methodology for tracking and traffic correlation applies to any IPv6 provider, regardless of whether they employ prefix rotation (if they do not, tracking is trivial.) If EUI-64 IIDs in an AS to be tracked are known a priori, the active probing techniques discussed in §3 can be employed to track these devices. We next consider an off-path adversary that wishes to find *all* networks that employ prefix rotation and track *all* users within those networks. In §6 we demonstrate the real-world ability of an off-path attacker (us) to perform this tracking against a set of clients.

Given the massive IPv6 address space, we bootstrap using existing large-scale traceroute campaigns to identify networks with EUI-64 addresses at the network periphery. We use a traceroute campaign [9] conducted from March to April 2019 by CAIDA that issued a traceroute to a single target in each /48 subnetwork of all networks /32 or smaller that were advertised in the Border Gateway Protocol (BGP). Even though the CAIDA measurements were taken more than a year before ours, this "seed" data valuably informs our target selection for address discovery and evaluation of prefix rotation (§4.3) by identifying /48 networks that produce last responsive hops that have EUI-64 IIDs.

The CAIDA seed data contains 32,325 /48 networks with a unique responsive EUI-64 last hop address – that is, no other target address in a different /48 resulted in the same last hop EUI-64 address. These 32k /48 networks are part of 938 distinct /32 networks from 627 ASes; we use these /32 networks as the starting point of an active measurement campaign to discover CPE currently using EUI-64 addresses, detect prefix rotation, and ultimately track targeted devices. The subsequent steps included three tasks:

1. Expansion and validation of seed EUI-64 prefixes
2. Candidate /48 EUI-64 density inference
3. Prefix rotation detection

4.1 Seed /48 Expansion and Validation

For the 938 /32 target networks obtained from the seed data, we select a random /64 in each of the /32 network's constituent /48s, and append a random IID to produce 61,472,768 target addresses (938 × 65536 × 1). As detailed in §3.1, we use zmap to probe to each target and collect responses. This probing step validates that the CAIDA seed data produced /48 prefixes that generate EUI-64 addresses at the periphery, as well as attempts to discover other /48s within the same /32 that produce EUI-64 addresses. We find 48,970 /48 networks that produce a unique EUI-64 address response (expanding the seed data), and use these in the next step of probing.

4.2 Candidate /48 EUI-64 Density Inference

We conduct a second round of active measurements to determine the *density* of EUI-64 periphery addresses within candidate /48 networks. Here, density refers to the number of unique EUI-64 response addresses received divided by the number of probes sent to target addresses in the /48. Different EUI-64 densities can arise from variation in the size of prefix allocated to the customer. For example, a provider allocating /48 networks to end sites might have an EUI density of $\frac{1}{2^{64-48}}$, while on the opposite end of the spectrum, a provider allocating /64s to end sites may have a theoretical maximum density of 1 if each /64 is allocated to a different CPE with an EUI-64 address. A prefix's density is influenced both by the presence of CPE using EUI-64 addresses and the size of the customer allocations.

We send one probe to each /56 in the /48 network generated from §4.1 (approximately 12.5M new probes). We aggregate the results by target /48, counting each unique EUI-64 response generated by probing to a destination in each target /48. If the number of unique EUI-64 responses per target sent (the unique EUI-64 response density) is less than 0.01 (meaning that the number of unique EUI-64 responses was 2 or fewer), we classify this target /48 network as low density, and omit it from future probing. We chose this threshold to eliminate prefixes allocated to a single device or load balanced between two interfaces; while networks that assign /48s to endhosts can also rotate prefixes, the exhaustive probing we perform in §4.3 and §5 prohibit inclusion of these networks due to the additional time required to probe them, while yielding few distinct responses. All other networks (> 2 EUI-64 responses) we classify as high density, and use them for host discovery probing. By this conservative definition, we discover 17,513 high density and 27,429 low density target /48 networks. For the 4,028 remaining networks, we obtain no responses after probing them in §4.1, and so omit them from subsequent probing.

4.3 Prefix Rotation Detection

After identifying /48 networks that we are confident currently generate EUI-64 responses when probed, we turn to the last step: identifying which networks exhibit *prefix rotation.* We concatenate a random IID to each /64 of the 17.5k high density prefixes to generate ~1.1B target addresses. We use zmap to probe each of these addresses in a random order, attempting to elicit a response from CPE routers. We repeat this scan again in the same order 24 hours later.

With two complete scans of the 1.1B addresses, we filter for $< target, response >$ pairs where the response is an EUI-64 IPv6 address in either scan. Then, we remove the $< target, response >$ pairs that are common between the two scans. We do this to isolate targets with an EUI-64 responsive address that changes over time. These scans reveal EUI-64 address changes, potentially indicating a short interval between when a prefix is released by one CPE device and then reallocated to another. Our approach also discovers shifts between a responsive EUI-64 address and no response, which suggests that the prefix was allocated to a CPE and then returned to the unallocated pool, or from an EUI-64 address to a different type of IPv6 address, such as SLAAC with privacy extensions. We observed EUI-64 responsive addresses change in 12,885 /48 networks.

Our approach, where two snapshots are taken 24 hours apart and compared, will likely fail to capture networks whose prefix rotation interval is on a longer timescale (e.g., 6 months). While a provider's rationale for rotating prefixes may vary – for instance to prevent static addresses for a particular service tier, to manage IP reputation, or for privacy – we focus on networks that routinely and frequently rotate delegated customer prefixes as these are the most challenging to track from a privacy perspective. We also chose not to set a threshold for prefix rotation (e.g., 75% of responsive EUI-64 addresses leave a prefix between the two snapshots) in order to allow for a gradual or non-uniform rotation. Such a threshold could be set to further restrict the definition of prefix rotation, however. In future work, we plan to more exhaustively explore the range of provider behaviors, including rotations on a weekly or monthly basis.

The result of these three steps is a set of ~13k /48 networks that are likely employing prefix rotation. Table 1 lists the ASN and countries with the most rotating /48 prefixes. Versatel (AS8881, a large German residential provider) dominates, accounting for 40% of all /48s; as a consequence, Germany is the most prevalent rotating country with 46% of all /48s. AS8881 is a large residential DSL provider that rotates CPE periphery prefixes on a daily basis, making its prefixes more likely to be flagged as rotators than service providers with a longer rotation period. However, more than 100 ASes, across 25 different countries, have at least one rotating /48. While this is a small fraction of the total number of IPv6 ASes, this represents 16% of the total ASes with an EUI-64 address discovered by the original CAIDA routed /48 scan.

Furthermore, during the three stages outlined in §4, we discovered 19.4M total IPv6 addresses. Of these 19.4M total addresses, 14.8M were EUI-64 addresses, while 4.6M were not. Only 6.2M IIDs of the 14.8M EUI-64 addresses were unique, indicating that a significant fraction of the EUI-64 addresses rotated during this detection phase.

We next describe a measurement campaign conducted over 1.5 months against these identified rotating /48 networks in order to characterize provider allocation policies and characterize prefix rotation behavior in the wild.

5 MEASURING INTERNET-WIDE PREFIX ROTATION BEHAVIOR

Having identified 13k /48 networks exhibiting likely prefix rotation (§4.3), we probed these networks on a daily basis. Our measurement

Table 1: Top 5 ASNs and Countries in Probe Campaign by Number of /48 Prefixes Probed

ASN	# /48	Country	# /48
8881	5,149	DE	5,985
6799	3,386	GR	4,063
1241	635	CN	1,126
9808	608	BR	561
3320	530	BO	264
96 Other ASNs	2,577	20 Other Countries	886
Total	12,885	**Total**	12,885

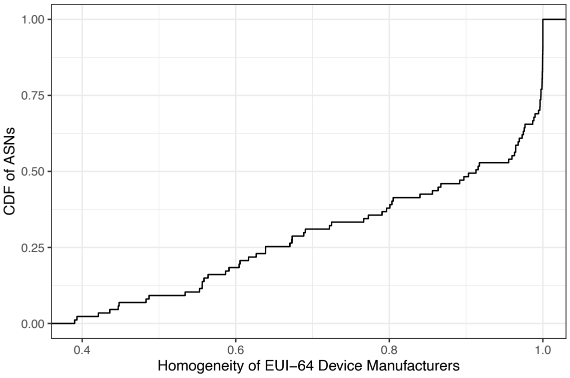


Figure 4: Fraction of devices belonging to the most common manufacturer per ASN. Manufacturer homogeneity is indicative of ISP-provided CPE; highly homogeneous ASes aid attackers focused on specific device brands or models. Eighty-seven ASes are represented; ASes with <100 EUI-64 IID are excluded.

campaign spanned 44 days from late July to early September 2020 and probed 844M addresses each day. To ensure temporal consistency across daily zmap runs, we probed the same addresses every 24 hours in the same order (same zmap random seed), beginning at the same time each day.

Over the course of our campaign, we sent over 37B ICMPv6 Echo Requests to these 844M destination addresses. We received more than 24B responses from 134M unique IPv6 addresses, including 110M unique EUI-64 addresses. Of these EUI-64 addresses, we observed only 9M distinct IIDs, indicating that we observed the same EUI-64 IID appearing in many different addresses throughout our study.[1]

5.1 Provider EUI-64 Homogeneity

EUI-64 IIDs are formed by setting the Universal/Local (U/L) bit of the hardware MAC address, and inserting `ff:fe` between the third and fourth bytes. Hence, it is trivial to recover the CPE's Internet-facing MAC address by reversing this process. Because the three high-order bytes (commonly known as the OUI) of a MAC address encode information about the manufacturer of the device or interface, we use publicly-available information [5] to study the per-AS distribution of CPE manufacturers we discover.

Service providers exhibit distinct manufacturer fingerprints. For example, we discovered 205,559 distinct MAC addresses embedded in EUI-64 addresses belonging to NetCologne (AS8422), a German ISP. Of these, 99.98% (205,527) were in OUIs belonging to AVM [2], a German electronics company that produces Fritz!Box CPE routers. Of the remaining 32 MAC addresses, 24 belonged to OUI registered to Lancom Systems, another European CPE manufacturer, one was registered to Zyxel Communications, and the final seven did not resolve to any OUI listed by the IEEE.

Viettel Group (AS7552), a Vietnamese ISP, similarly exhibited a high degree of CPE homogeneity, but with a different set of manufacturers. These differences reflect geographic market presence of device manufacturers, as well as relationships between providers and the products they lease or sell to subscribers. We discovered 420,248 distinct MAC addresses within EUI-64 addresses belonging to Viettel prefixes. Here, 99.6% (418,611) of these MAC addresses are registered to ZTE Corporation, a Chinese manufacturer.

Figure 4 examines the distribution of manufacturers between and within ASes. We define an AS's homogeneity as the fraction of unique EUI-64 IIDs belonging to the most common device vendor per AS:

$$homogeneity(ASN) = max\left(\frac{\text{unique EUI-64 IID(manufacturer)}}{\text{total unique EUI-64 IID}}\right)$$

We exclude ASes with fewer than 100 EUI-64 IIDs to prevent ASes with few EUI-64 IIDs from skewing the distribution. We observe a that high device manufacturer homogeneity is common; in the 87 remaining ASes these EUI-64 addresses belong to, more than half have a homogeneity index > 0.9, and three-quarters of ASes have a homogeneity index above 0.67. Even in the least homogeneous ASes, more than one out of three devices is manufactured by the same entity. Despite this level of homogeneity within ASes, more than 200 distinct manufacturers are observed throughout the ASes we characterize.

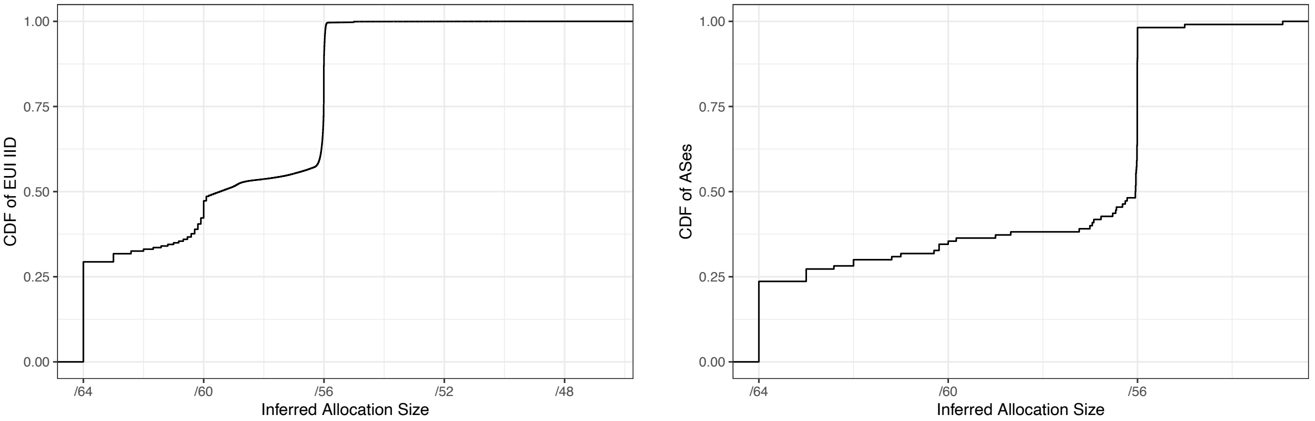
Exposing device information via embedded MAC addresses opens up a network to targeted attacks, for instance for an attacker that has identified a vulnerability specific to a certain vendor or model. Such an attacker might scan networks with many devices manufactured by that vendor to find potential targets, and eliminate the need to search networks where few or no devices made by the vulnerable manufacturer appear. Further, a high-degree of device homogeneity can make individual network less robust and resilient.

5.2 Prefix Allocation Sizes

The subplots of Figure 5 demonstrate our ability to infer the size of allocations delegated to subscribers by Internet Service Providers (ISPs). First, Figure 5a displays the CDF of inferred allocation sizes for all EUI-64 IID discovered on a single day of probing. The most common allocation size delegated to a CPE is /56 at about 40% of all EUI-64. The /64 allocation size is also common, at approximately 30% of all EUI-64, and an inflection point at /60 indicates that this size is also allocated, albeit with less frequency in our data. Figure 5b depicts the cumulative density of median allocation sizes by AS; /56 is the most commonly allocated size among the providers we

---

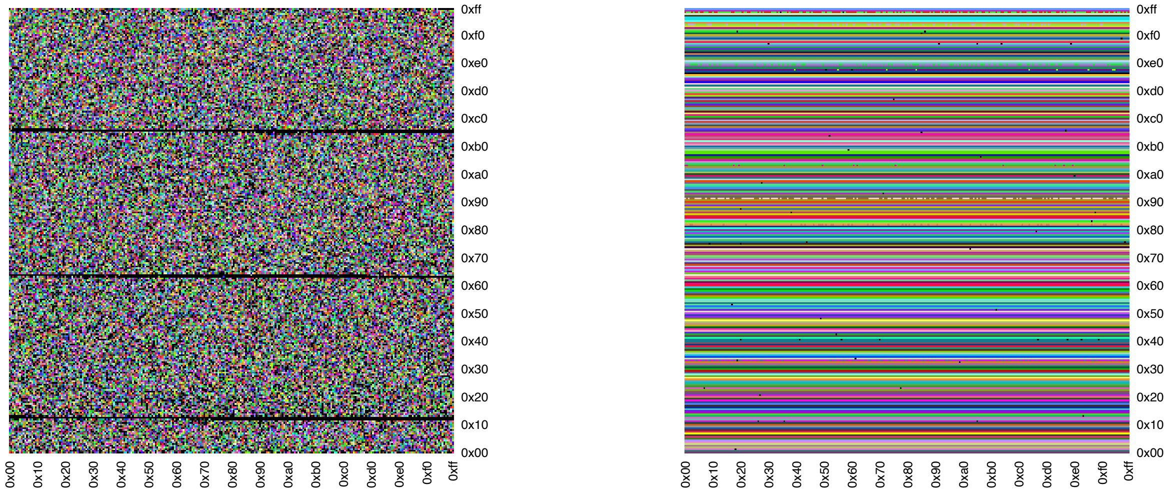
[1] §5.5 examines pathologies including MAC reuse



(a) Inferred allocation size of EUI-64 IIDs as a CDF of all EUI-64 IIDs. A plurality of IIDs (~40%) are allocated /56 subnets by their service provider, while ~30% are allocated /64s. An inflection point at /60 indicates it is a less-common allocation size.

(b) Median inferred prefix allocation sizes of ASes. A /56 is the most commonly allocated prefix size of the providers we probed representing about half of the probed ASes. Approximately one-quarter of the ASes allocate /64 networks to their customers.

Figure 5: Inferred Allocation Sizes as CDFs of EUI-64 IIDs and ASes.



(a) A Versatel prefix (2001:16b8:501::/48) with inferred /64 customer allocations.

(b) A Versatel prefix (2001:16b8:11f9::/48) with inferred /56 customer allocations.

Figure 6: A provider exhibiting multiple prefix allocation sizes.

probed, accounting for 50% of all ASes. We infer that about one-quarter of ASes allocate a /64 to customers, with another one-quarter using between /64 and /56 prefixes. Some of these ASes allocate /60s (e.g., BH Telecom, Figure 3b), others allocate /56 or /64 with noise introduced in the inference by prefix rotation, and still others provide multiple allocation sizes for different customers. Figure 6 depicts two /48s from the same ISP, one of which is divided into /56 allocations while the other is split into /64 subnets.

Understanding a provider's prefix allocation size(s) can potentially enable an adversary to increase their scanning efficiency. For instance, assume that an attacker attempting to track a CPE device that uses EUI-64 addresses knows that the AS the device resides in typically allocates /56 networks to end sites. The adversary then needs only to probe each /56 subnetwork when attempting to discover the targeted device. We observed some ASes allocate multiple prefix sizes to end sites, likely due to a variety of customer types or service plans. In these cases, an adversary may choose to scan initially assuming the larger allocation size to potentially benefit from scan efficiency. In the event the targeted device goes undiscovered, a second scan using the smaller allocation size may be necessary to receive a reply.

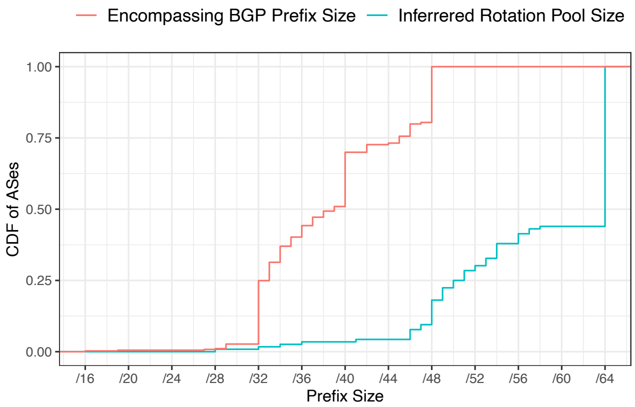


Figure 7: Inferred rotation pool sizes of ASes vs BGP-advertised prefix sizes. The difference between the BGP-advertised prefix and rotation pool size represents an attacker's cost savings attempting to track a device with an EUI-64 address, as addresses tend to stay within their rotation pools for long periods of time.

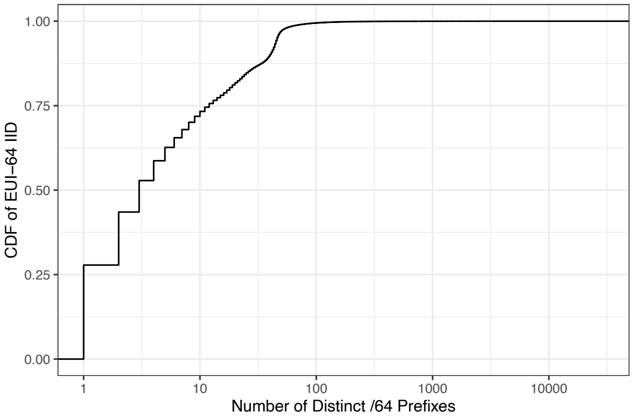


Figure 8: Number of unique /64 prefixes per EUI-64 IIDs (*x*-axis log-scale). Most (~70%) EUI-64 are found in more than one /64, indicating they rotate prefixes during the observation window.

5.3 Validating Prefix Rotation Detection

Figure 7 depicts the inferred rotation pool sizes for the 101 ASes to which the EUI-64 addresses belong, as well as the encompassing BGP prefix sizes for the EUI-64 addresses. We use Routeviews [26] global BGP data to map response addresses to BGP prefixes.

We infer a rotation pool size of /64 for more than half of the ASes we probed. This is indicative of non-rotation within these ASes, /64 is the minimum allocation size possible to allow IPv6 hosts to perform SLAAC. While our initial probing in §4 was designed to find prefixes with rotation pools, it is sensitive to the appearance or disappearance of EUI-64 addresses that may be caused by customers joining the prefix or turning off their CPE, rather than larger-scale rotation events. Conversely, approximately half of the ASes we probed during our measurement campaign *did* exhibit measurable prefix rotation, as evidenced by rotation pool sizes > /64. Finally, we compare the inferred rotation pool size, representing the range within which we *observed* an EUI-64 IID rotate addresses, to the BGP-advertised encompassing prefixes, which represent the *possible* range within which the EUI-64 IID might appear. The difference between the two lines is approximately a /16 throughout, indicating that an EUI-64 IID typically rotates within only $\frac{1}{2^{16}}$ of the possible range. This allows an adversary attempting to track an EUI-64 device to bound their search space to a modest subnetwork of the encompassing BGP prefix, increasing their likelihood of success.

Figure 8 is a CDF of the number of distinct /64 network prefixes in which we observed each EUI-64 IID (alternatively, a distinct CPE MAC address) in responses to our probes. We observed about one-quarter of EUI-64 IIDs in only one /64 during the course of our study. There are two potential reasons this might occur. First, the containing address never rotates. Although we attempted to identify rotating prefixes (§4), our method may erroneously categorize targets as potential rotators due to new devices appearing between the first and second snapshots in the rotation identification stage (§4.3). A second explanation is that the address *does* rotate prefixes, but the prefix that it rotates to does not belong to the set of prefixes that we probe daily. In this case, we would not observe the IID for some period of the measurement campaign. More than 70% of distinct IIDs appeared in more than one /64 prefix during our study, indicating that they rotated prefixes at least once. Finally, a small number of IIDs rotated across an extreme number of /64 prefixes, including one IID that appeared in nearly 30,000 distinct /64 networks. We discuss these anomalies in greater detail in §5.5.

5.4 Rotation Pool Behavior

To better understand the lower-level dynamics exhibited by prefix-rotating providers, we investigated the behavior of EUI-64 addresses *within* rotation pools. Figure 9 displays the observed /64 prefixes for three EUI-64 IIDs in AS8881. Each IID rotates throughout the observation window within the same /46, AS8881's inferred rotation pool size (§3.2.2). The figure shows that each EUI-64 IID's /64 prefix increments each day; when the value is greater than the /46 rotation pool, the /64 prefix simply wraps modulo 2^{18} to remain within the /46. This causes EUI-64 IIDs #1 and 2 (red and green lines) to appear in three /48 prefixes before wrapping modulo the /46 rotation pool for the first several days, while EUI-64 IID #3 alternates between 2001:16b8:1d01::/48 and 2001:16b8:1d02::/48. This helps scope an attacker's prediction of what prefix an IID will have in the future.

Next, we look at the number of EUI-64 addresses in each /48 within a rotation pool, again focusing on AS8881. Figure 10 plots the density of EUI-64 addresses within each /48 of a /46 rotation pool. For each hour over the course of a week, we probed portions of the AS8881 address space to elicit responses from EUI-64 addresses within the 2001:16b8:100::/46 prefix. Figure 10 shows that prefix reassignment predominantly occurred between midnight and 0600 CEST each day. Generally, on a given day, one /48 prefix contained the most addresses, one /48 contained close to none, and the other two /48s contained the smaller remainder, changing densities in opposite directions from each other.

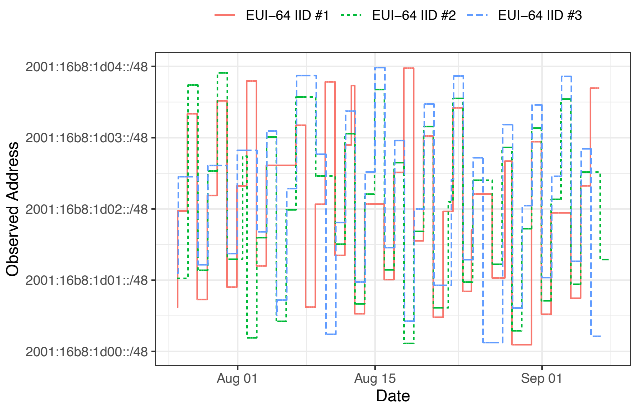


Figure 9: Three AS8881 EUI-64 IIDs' assigned /64 prefixes over time. Each IID's network prefix increments each day modulo the size of the rotation pool (/46).

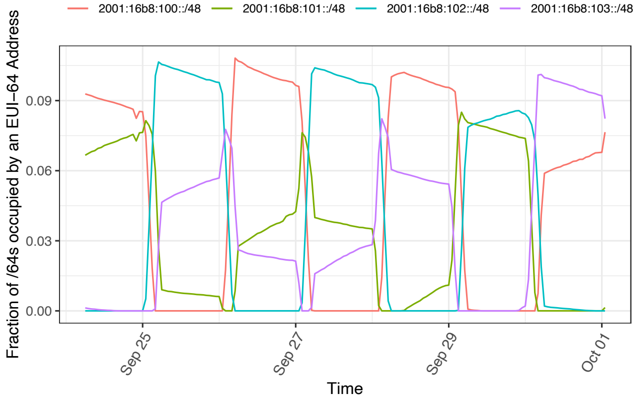


Figure 10: Address density of an AS8881 (DE) /46 rotation pool over time. Prefix reassignment typically occurs during the early morning hours; on any given day, one prefix contains the majority of EUI-64 addresses between them, another contains almost none, and the remaining two are changing densities in opposing directions.

5.5 Pathologies

We observed several phenomena during our campaign that bear further study, as they fell outside of the behaviors we expected EUI-64 IIDs to exhibit. First, of the 9M distinct EUI-64 IIDs we observed, 10k of these were observed in multiple ASes. One, corresponding to the MAC address `00:00:00:00:00:00`, was observed in 12 distinct ASes, likely because this MAC is used as a default address, particularly when the interface may not have a pre-programmed MAC address (such as a cellular interface on a hotspot device.)

Another category of multiple-AS EUI-64 IIDs is more difficult to explain; in this subset, the same IIDs appeared in networks on different continents within the same day for the duration of our study. Most of these EUI-64 IIDs correspond to a single manufacturer.

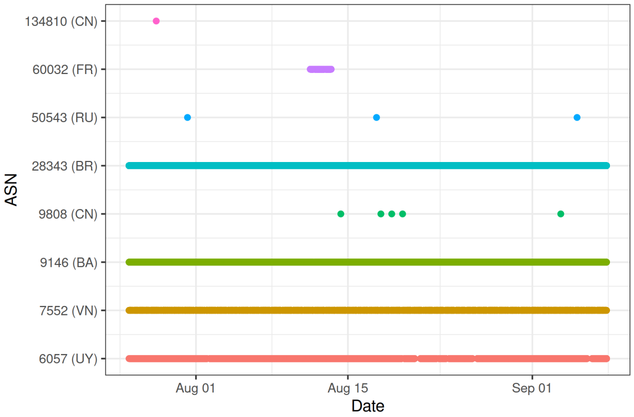


Figure 11: Responses from a single EUI-64 IID by ASN over time. Sustained observations in multiple countries throughout the observation window suggests MAC address reuse by a vendor, decreasing its utility as a trackable identifier.

Figure 11 depicts the observations of a single EUI-64 address in multiple ASes distributed globally during our measurement campaign. In this case, we observed an EUI-64 IID appear daily in ASes located in Uruguay, Vietnam, Bosnia, and Brazil, as well as occasionally in China, Russia, and France. Due to the geographic distribution of the ASes, this is likely a MAC address reused multiple times by the manufacturer in violation of the standard [1]. Finally, many of these 10k addresses that appear in multiple ASes shift from one AS to another, i.e., stop appearing in the first AS after the switch. This behavior is evidence that the customer has changed service providers, especially if the providers serve the same region. Figure 12 illustrates this behavior as seen from two different devices, one of the two moving from AS8881 (2001:16b8::/32 addresses) to AS3320 (2003:e2::/32 addresses) in early August, and the other in the opposite direction in early September. Neither are seen in the previous AS again after being observed in the new AS. Because both ASes are German residential service providers, this behavior appears to represent either a customer switching Internet service plans, or an instance in which the backup provider became the primary for a dual-homed device.

6 DEVICE TRACKING CASE STUDY

In order to demonstrate our technique's efficacy for tracking IPv6 CPE, we conduct an experiment that emulates an adversary interested in tracking specific EUI-64 IIDs as they change prefixes. By utilizing the previously described techniques, this section demonstrates the real-world feasibility of an attacker re-identifying traffic from clients despite the deployment of technologies intended to prohibit such re-identification.

Over the course of a day, we probed the targets described in §4.3 to discover current EUI-64 addresses. Then, we selected ten EUI-64 IPv6 addresses at random, with the caveats that no two addresses came from the same country or AS, and excluding EUI-64 IIDs that we had observed previously in multiple ASes (§5.5). We then probed

Table 2: Characteristics of ten prefix-changing EUI-64 IIDs tracked over one week

EUI-64 IID	Mean Probes / StdDev	BGP Prefix	ASN	CC	# Days	# /64 Prefixes	EUI-64 IID	Mean Probes / StdDev	BGP Prefix	ASN	CC	# Days	# /64 Prefixes
#1	4238 / 0	/48	7552	VN	7	2	**#6**	28,331.5 / 10,867.3	/32	56044	CN	2	2
#2	125,679.7 / 95,694.0	/32	8422	DE	7	7	**#7**	9,293 / 5,575.6	/32	9146	BA	7	6
#3	379 / 315.7	/32	262557	BR	7	2	**#8**	14,523 / 4,558.5	/40	8881	DE	2	2
#4	41,413.3 / 17,522.0	/37	27699	BR	7	3	**#9**	1,363.3 / 530.3	/32	10834	AR	5	2
#5	152,676 / 82,977.9	/33	14868	BR	7	2	**#10**	860.3 / 284.8	/48	200924	DE	3	3

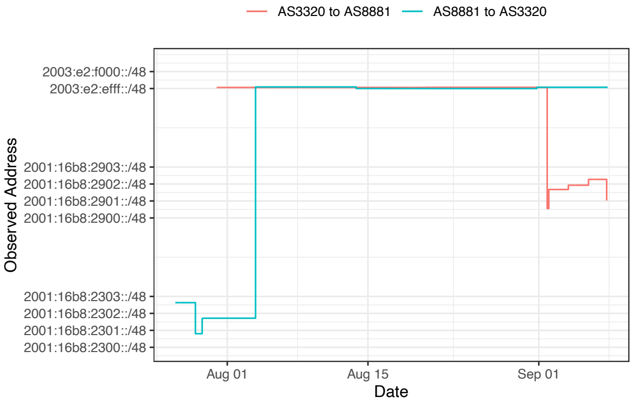


Figure 12: Two EUI-64 IID changing between German ISPs (y-axis not to scale). Neither are seen in their former provider's network after shifting to their new network, indicating that their owners changed service providers rather than are utilizing a backup link during an outage.

over an additional six days to attempt to track these IIDs for one week.

In addition to demonstrating that CPE can be tracked over time, another central aim of this case study is to validate the space-reduction approach depicted in Figure 2. For each of the ten selected addresses, we leveraged the allocated prefix size and rotation pool size inferences we made for each AS in §3.2.1. We used the allocation size inference in order to reduce the number of probes a naïve search algorithm might send: one probe per /64 in the suspected target prefixes. As an example, for providers who commonly allocate /56 prefixes to customers, we need send only one probe to an address within that subnet in order to elicit a CPE reply – thereby sending only $\frac{1}{256}$ the number of probes of a naïve scanner. We incorporated the inferred rotation pool sizes to ensure we probe the space an address is *likely* to rotate to. For example, some ASes use /46 rotation pools and we chose a target in each allocation size block throughout the entire pool. While this somewhat offsets the cost savings we obtain by probing only once per allocation size, it widens the target aperture to encompass the range of addresses within which an EUI-64 address is likely to fall. This approach increases the number of observations and decreases the likelihood that we "lose track" of an EUI-64 IID.

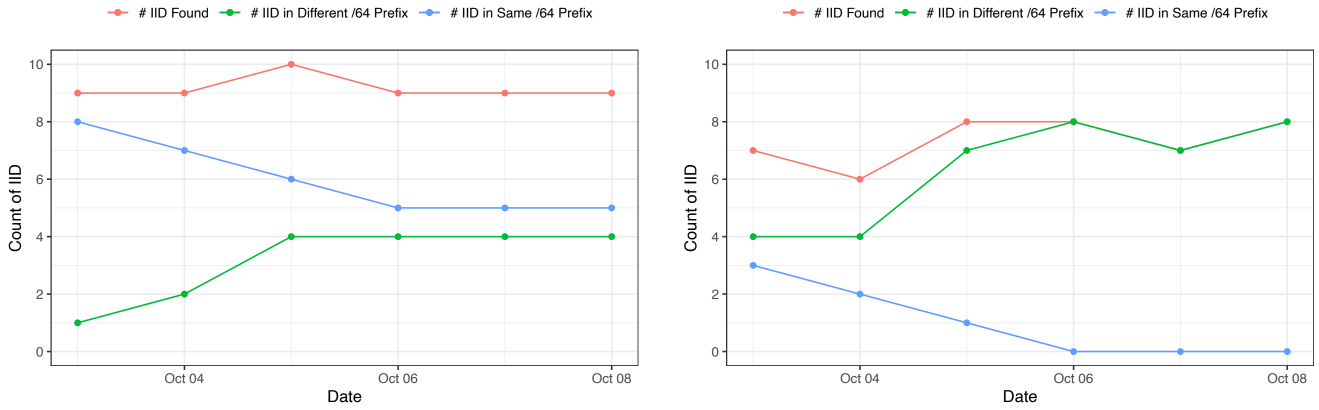
Figure 13a displays the number of IIDs successfully tracked each day. Over all six days, we discovered 9 to 10 IID (out of 10) targets. This suggests that, across device manufacturers, countries, and ASes, tracking CPE over time via EUI-64 addressing is quite feasible.

We also observed that the number of IIDs whose prefixes changed during the course of the study increased from 1 to 4 IIDs. This matches our intuition, in that devices may only rotate prefixes after several days. Half of the IIDs we selected remained in the same prefix over the duration of the experiment. This likely indicates that these addresses were not in networks that rotate prefixes sub-weekly and whose addresses are therefore trivially tracked.

Rather than selecting EUI-64 addresses at random, we next chose 10 EUI-64 IIDs that *did* exhibit prefix rotation during our tracking time frame. Figure 13b plots the number of IIDs we discovered during the study; in this case, we discovered slightly fewer IIDs per day than without the rotation restriction, with a maximum of 8, and a minimum daily IID discovery of 6 of the 10 IIDs. Unlike Figure 13a, Figure 13b shows that all of the IIDs we tracked eventually changed prefixes by the fourth day of probing. Despite this, we still discovered 8 of the 10 EUI-64 IIDs, emphasizing that EUI-64 addressing undermines the privacy goals of prefix rotation.

Table 2 summarizes the characteristics of the 10 prefix-rotating EUI-64 IIDs tracked over a week, including the number of probes sent until they were located in their new prefix. If the EUI-64 IID was not found on a given day, we include the total number of probes that we sent. As an example, we discovered "EUI-64 IID #3" after sending only 379 probes on average each day. Because "EUI-64 IID #3" is part of a BGP-advertised /32 allocation, exhaustively searching the space would entail up to 2^{32} probes in order to enumerate each /64 that could potentially be allocated to a CPE device. At a rate of 10k packets per second, this probing would take nearly five days, during which period the EUI-64 IID might again change prefixes. By contrast, using our target selection method and inferred prefix allocation and rotation pool sizes, we discover this EUI-64 IID over the course of a week using only 2,274 total probes in two different EUI-64 IPv6 addresses. "EUI-64 IID #2", which was discovered in 7 different /64 networks over the week of probing, took only 754k packets total to locate, equivalent to 75 seconds of active probing at 10k packets per second.

While the majority of the EUI-64 IIDs were discovered each day of the tracking case study, some were not, and two were observed on only two days during the week. We consider two explanations for this behavior. First, our inferences of the provider's allocated prefix and rotation pool sizes may be incorrect. If one (or both) of these inferences is incorrect, our target generation technique, which sends one probe to each allocation-sized block within the prefix rotation pool, may not probe to an address within the new allocation assigned to the CPE. For instance, if we infer a /56 allocated prefix size for a provider, but the CPE is actually within a block that receives a /64 for its internal subnetwork, it is unlikely that our probes will target the CPE /64 as the probed /56 is chosen at random. Alternatively, if we underestimate the prefix rotation pool size, the



(a) Discovered EUI-64 IIDs during targeted tracking. While some IID rotate prefixes, others do not; regardless of rotation, 9 of 10 are found consistently over the course of a week.

(b) Discovered prefix-rotating EUI-64 IIDs during targeted tracking. Even when targeting only devices that rotate prefixes, we can track most IIDs during the week-long observation window.

Figure 13: Results from tracking two sets of ten EUI-64 IID over a week.

CPE may rotate out of the address space we are probing. If the address pool is larger than our estimate, then in the future, we may observe this EUI-64 again when it rotates back into our probing range. If, however, a device *changes* rotation pools, the EUI-64 will go unobserved for the duration of the probing unless some other event forces it back into the probed rotation pool. In this case, a motivated adversary may attempt to simply identify many or all of a provider's rotation pools, and enumerate those to find the device again.

A second explanation for the disappearance of an EUI-64 IID is the removal of a device from service, whether by a change in provider (§5.5), or an extended outage. In these instances, an adversary would not expect to see the device return, and should cease probing this EUI-64 IID after some time without positive results.

7 ETHICAL CONSIDERATIONS

We followed established ethical principles and recommended practices for high-speed Internet probing during this work [12]. Our probing consisted of sending ICMPv6 Echo Requests using zmap6. ICMPv6 messages are used for diagnostic and error-reporting purposes, and are considered less obtrusive than UDP or TCP probes. In order to minimize the risk of ICMPv6 rate limiting, which is mandatory, and to reduce the load on transit and destination networks, zmap6 randomizes the order of probing. We probed at a conservative rate, coordinated with the administrators of the vantage point, and ran an informative website from the vantage point providing contact information to opt-out of our experiment. We received no opt-out requests.

We contacted and initiated a security report with the CPE manufacturer whose devices comprise 22% of the EUI-64 IID we discovered to disclose the privacy vulnerability. We sought to work with device manufacturers to bring awareness to the tracking implications of EUI-64 CPE addresses and to remediate the privacy flaw by ensuring future iterations of their CPE OSes implement SLAAC privacy extensions (§8).

8 REMEDIATION

EUI-64 SLAAC addressing in CPE devices enabled the targeted tracking case study of §6. The most straightforward way to protect consumers from this type of tracking is for CPE manufacturers to ensure their products are capable of SLAAC privacy extensions and to enable privacy extensions by default. Because service providers often partner with CPE vendors to offer their products directly to customers (as evidenced in §5.1), they also have an active role to play in ensuring these devices provide an adequate level of privacy.

We contacted a CPE manufacturer that appeared prominently (~2 million MAC addresses) in our results regarding the privacy and tracking implications of their continued use of EUI-64 SLAAC addresses. A company representative informed us that the motivation for their use of EUI-64 SLAAC addresses is to optimize connection setup time. Specifically, implementing SLAAC with privacy extensions requires the device to perform Duplicate Address Detection (DAD) [20, 28]. We believe that the benefit of using the privacy extensions of SLAAC addresses outweighs the delay incurred while DAD is performed, particularly in light of RFC 4429 [18], which allows an interface to be used while DAD completes. When presented with our results, the manufacturer agreed that the privacy vulnerabilities of continued EUI-64 use outweighed the marginally-increased interface setup time, and indicated that they will implement SLAAC with privacy extensions rather than EUI-64 SLAAC in the next release of their Operating System (OS) in early 2022 [14].

Further, RFC 4941 [20] specifies only that a device implementing SLAAC with privacy extensions `SHOULD` generate a new, random IID each time its network changes. Our work shows that `SHOULD` is too weak, and the privacy goals of this standard dictate that the CPE `MUST` do so in order to prevent the same type of tracking using the randomized IID rather than an EUI-64 IID.

In the event that a device cannot support SLAAC privacy extensions due to some technical limitation (unlikely, given that IPv6 privacy extensions were introduced in 2001), ISPs should provide a mechanism to inform users about their increased vulnerability to tracking.

9 SUMMARY AND CONCLUSIONS

We used large-scale active measurements to demonstrate a widespread vulnerability with deployed IPv6 privacy enhancing technologies that randomize IIDs and rotate address assignments assigned to a single customer. Although these enhancements are ubiquitously deployed at the edge, the CPE market lags behind. The legacy IPv6 standard does not require the use of SLAAC privacy extensions, so these CPE devices are still compliant with the IPv6 standard; however, EUI-64 addressing undercuts modern IPv6 privacy features supported by all major modern OSes. As a result, any adversary that can send active probes toward random IP addresses in target prefixes, is potentially able to track devices in that prefix. An adversary could use similar scalable active probing methods to identify which IPv6 prefixes are likely to be fruitful targets for such tracking.

We demonstrated methods for both approaches and applied them to a case study of device tracking, finding the problem spans CPE devices observed in over 100 ASes and 25 countries. Simply put, our measurements show that the privacy/anti-tracking mechanisms used in IPv6 today do not work without additional coordination with the CPE. Based on our findings, a major CPE vendor, whose products account for over 2 million distinct MAC addresses in our corpus, has deprecated EUI-64 addressing in its forthcoming OS release. This change will materially improve user privacy.

The security and privacy community has devoted extraordinary effort over the last decade to developing privacy-enhancing technologies intended for pervasive and general-purpose use. To be effective, such efforts must often span technical, policy, and standards development work. But without empirical study of how these technologies are deployed, we have limited understanding of which privacy-enhancing technologies are truly achieving their intended goals. This study represents an example of the interplay between privacy-enhancing technologies, and the context in which these technologies are deployed, which in many cases can inadvertently subvert the privacy enhancements. Worse, in the example we studied, users receive no notification of this interaction, which could cause harm relative to not having the privacy enhancements in the first place, because the enhancements offer a false sense of privacy and security. We believe the research and standards development community should consider deprecating legacy IPv6 behavior that uses static EUI-64 addresses, or at least requiring notification to the user of privacy implications.

Content and service providers must also account for prefix rotation when blocking attack traffic originating from IPv6 users. The IPv4 paradigm of denying or rate-limiting a single address or range of addresses is ineffective when client prefixes may rotate daily. Our results suggest that future work is needed to better understand how providers might rethink employing such security mechanisms in the face of these deployed IPv6 technologies.

Alleviating the privacy and tracking concerns we demonstrate arise from EUI-64 addressing CPE has a straightforward solution – CPE vendors *must* ensure their OSes support SLAAC privacy extensions and enable their use by default. If supporting privacy extensions is technically infeasible or undesirable, users must be warned about their increased vulnerability to tracking by their service provider or equipment vendor. Service providers must also take an active role in certifying that the products they supply to customers meet these privacy and anti-tracking requirements. Customers have the right to expect that their privacy will be protected using state-of-the-art, privacy-enhancing addressing schemes, rather than jeopardized by CPE employing legacy standards.

ACKNOWLEDGEMENTS

We thank Will van Gulik and Young Hyun for providing measurement infrastructure, Jan Schöllhammer from AVM GmbH for his responsiveness to our vulnerability disclosure, and the anonymous reviewers for feedback. Kirstin Thordarson's thesis [29] provided valuable early insight for this study. This work was supported in part by NSF grants CNS-1855614 and CNS-1901517. Views and conclusions are those of the authors and should not be interpreted as representing the official policies or position of the U.S. government or the NSF.

REFERENCES

[1] IEEE Standard for Local and Metropolitan Area Networks: Overview and Architecture. *IEEE Std 802-2014 (Revision to IEEE Std 802-2001)*, pages 1–74, 2014.
[2] AVM, 2020. https://en.avm.de/.
[3] BH Telecom, 2020. https://www.bhtelecom.ba/.
[4] Entel Bolivia, 2020. https://www.entel.bo/.
[5] IEEE OUI database, 2020. http://standards-oui.ieee.org/oui.txt.
[6] Starcat Cable Network, 2020. http://www.starcat.co.jp.e.lh.hp.transer.com/.
[7] Robert Beverly. Yarrp'ing the Internet: Randomized High-Speed Active Topology Discovery. In *Proceedings of ACM Internet Measurement Conference (IMC)*, November 2016.
[8] Robert Beverly, Ramakrishnan Durairajan, David Plonka, and Justin P. Rohrer. In the IP of the Beholder: Strategies for Active IPv6 Topology Discovery. In *Proceedings of ACM Internet Measurement Conference (IMC)*, November 2018.
[9] CAIDA. The CAIDA UCSD IPv6 Routed /48 Topology Dataset, 2019. https://www.caida.org/data/active/ipv6_routed_48_topology_dataset.xml.
[10] A. Conta, S. Deering, and M. Gupta (Ed.). Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification. RFC 4443 (Internet Standard), March 2006. Updated by RFC 4884.
[11] A. Cooper, F. Gont, and D. Thaler. Security and Privacy Considerations for IPv6 Address Generation Mechanisms. RFC 7721 (Informational), March 2016.
[12] Zakir Durumeric, Eric Wustrow, and J Alex Halderman. Zmap: Fast internet-wide scanning and its security applications. In *22nd USENIX Security Symposium (USENIX Security 13)*, pages 605–620, 2013.
[13] Oliver Gasser, Quirin Scheitle, Pawel Foremski, Qasim Lone, Maciej Korczyński, Stephen D. Strowes, Luuk Hendriks, and Georg Carle. Clusters in the Expanse: Understanding and Unbiasing IPv6 Hitlists. In *Proceedings of ACM Internet Measurement Conference (IMC)*, 2018.
[14] AVM GmbH. EUI-64 Vulnerability Disclosure.
[15] F. Gont and T. Chown. Network Reconnaissance in IPv6 Networks. RFC 7707 (Informational), March 2016.
[16] IAB and IESG. Recommendations on IPv6 Address Allocations to Sites. RFC 3177 (Informational), September 2001.
[17] Frank Li and David Freeman. Towards A User-Level Understanding of IPv6 Behavior. In *Proceedings of ACM Internet Measurement Conference (IMC)*, October 2020.
[18] N. Moore. Optimistic Duplicate Address Detection (DAD) for IPv6. RFC 4429, April 2006.
[19] T. Mrugalski, M. Siodelski, B. Volz, A. Yourtchenko, M. Richardson, S. Jiang, T. Lemon, and T. Winters. Dynamic Host Configuration Protocol for IPv6 (DHCPv6). RFC 8415 (Proposed Standard), November 2018.
[20] T. Narten, R. Draves, and S. Krishnan. Privacy Extensions for Stateless Address Autoconfiguration in IPv6. RFC 4941 (Draft Standard), September 2007.
[21] T. Narten, G. Huston, and L. Roberts. IPv6 Address Assignment to End Sites. RFC 6177 (Best Current Practice), March 2011.
[22] RIPE NCC. RIPE Atlas, 2021. https://atlas.ripe.net/.

[23] Erik Nygren. At 21Tbps, Reaching New Levels of IPv6 Traffic, 2020. https://blogs.akamai.com/2020/02/at-21-tbps-reaching-new-levels-of-ipv6-traffic.html.

[24] Ramakrishna Padmanabhan, John P Rula, Philipp Richter, Stephen D Strowes, and Alberto Dainotti. DynamIPs: Analyzing Address Assignment Practices in IPv4 and IPv6. In *Proceedings of the 16th International Conference on emerging Networking EXperiments and Technologies*, pages 55–70, 2020.

[25] David Plonka and Arthur Berger. Temporal and Spatial Classification of Active IPv6 Addresses. In *Proceedings of ACM Internet Measurement Conference (IMC)*, 2015.

[26] Routeviews. University of Oregon Route Views Project, 2020. http://www.routeviews.org/routeviews/.

[27] Erik C Rye and Robert Beverly. Discovering the IPv6 Network Periphery. In *International Conference on Passive and Active Network Measurement*, pages 3–18. Springer, 2020.

[28] S. Thomson, T. Narten, and T. Jinmei. IPv6 Stateless Address Autoconfiguration. RFC 4862, September 2007.

[29] Kirstin E Thordarson. Analysis of EUI-64-Based Addressing and Associated Vulnerabilities. Master's thesis, Monterey, CA; Naval Postgraduate School, 2020.

[30] tumi8. ZMapv6: Internet Scanner with IPv6 Capabilities, 2021. https://github.com/tumi8/zmap.

APPENDIX: ALGORITHMS

Algorithm 1 Allocation_Size(AS)

```
Input: <periphery response, target address> map $M$ for AS $A$
Output: Inferred allocation size
  $eui \leftarrow []$
  for $r \in responses$ do
      if isEUI($r$) then
          $e \leftarrow$ extractEUI($r$)
          $eui[e] = M[r]$
  $sizes \leftarrow []$
  for $e \in eui$ do
      $min\_r \leftarrow (min(eui[e]) \gg 64)$
      $max\_r \leftarrow (max(eui[e]) \gg 64)$
      $size \leftarrow log_2(max\_r - min\_r)$
      $sizes \leftarrow size \bigcup sizes$
  return $median(sizes)$
```

Algorithm 2 Rotation_Pool_Size(AS)

```
Input: Periphery responses for AS $A$
Output: Rotation pool size
  $eui \leftarrow []$
  for $r \in responses$ do
      if isEUI($r$) then
          $e \leftarrow$ extractEUI($r$)
          $eui[e] = r$
  $sizes \leftarrow []$
  for $e \in eui$ do
      $min\_r \leftarrow (min(eui[e]) \gg 64)$
      $max\_r \leftarrow (max(eui[e]) \gg 64)$
      $size \leftarrow log_2(max\_r - min\_r)$
      $sizes \leftarrow size \bigcup sizes$
  return $median(sizes)$
```