Where To Find Devices: Exploring EUI-64 in IPv6

Ting-Han Chen
University of Twente
Enschede, Netherlands
t.h.chen@utwente.nl

Jeroen van der Ham-de Vos
University of Twente
Enschede, Netherlands
j.vanderham@utwente.nl

Bart Batenburg
University of Twente
Enschede, Netherlands
b.batenburg@student.utwente.nl

Abstract

EUI-64 is one of the initial methods for generating IPv6 addresses that uses a device's MAC address. Later analyses showed problems with this method, leading to traceability and privacy concerns, and other IPv6 address-generation methods became preferred.

In this work, we investigate the current prevalence of EUI-64 IPv6 addresses within our university and national research and education networks. We have discovered 4 million unique addresses and analysed their generation patterns across different vendors. In our analysis, we have identified highly predictable EUI-64-generated identifiers used by known vendors and manufacturers, including concentrated address allocations and sequential patterns, which pose security and privacy risks. To mitigate these issues, we have compiled our findings into actionable insights for vendors and network operators to speed up EUI-64-enabled device discovery in IPv6 subnet scanning. We aim to contribute to IPv6 scanning for IoT device discovery and EUI-64 risk mitigation for academic research and network administration.

CCS Concepts

• **Networks** → **Network layer protocols**; • **Security and privacy** → **Network security**.

Keywords

IoT, Fingerprinting, IPv6, EUI-64

ACM Reference Format:
Ting-Han Chen, Jeroen van der Ham-de Vos, and Bart Batenburg. 2026. Where To Find Devices: Exploring EUI-64 in IPv6. In *11th Workshop on Traffic Measurements for Cybersecurity (WTMC '26), November 15–19, 2026, The Hague, Netherlands.* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3846375.3849092

1 Introduction

The gradual transition from IPv4 to IPv6 represents a significant architectural shift in terms of Internet scalability, flexibility, and support for modern networked environments [11]. A key feature introduced with IPv6 is Stateless Address Autoconfiguration (SLAAC) [17, 27], a mechanism that allows devices to automatically generate their own IPv6 addresses without requiring manual configuration or a centralised DHCP server. Historically, a widely adopted approach for SLAAC address generation has been the Extended Unique Identifier (EUI-64) format [17]. Under this method, a device derives its 64-bit Interface Identifier (IID) from its MAC address by splitting the MAC address into its Organizationally Unique Identifier (OUI) and device-specific Network Interface Controller (NIC) portion, interposing the fixed value `FF:FE`, and flipping the Universal/Local (U/L) bit [3]. This process ensures that each device can generate a unique and valid IPv6 address with minimal network overhead.

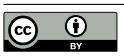
While the EUI-64 format simplifies address assignment and promotes address uniqueness, it introduces privacy and security concerns [21]. Since the MAC address is embedded directly into the IPv6 address, network activity can be linked back to a specific device. This makes devices vulnerable to tracking across networks, fingerprinting, and long-term profiling, particularly if the same MAC-derived IID is reused across multiple locations [19, 24]. In response to these risks, alternative mechanisms, such as privacy extensions and temporarily randomised addresses, were developed to obfuscate the IID and mitigate traceability [16]. Despite these standard mitigations, EUI-64 addresses have still been observed across the Internet due to legacy hardware, unpatched software [4, 19, 29], and manufacturers that continue to rely on EUI-64-based addressing schemes [19, 20].

Nevertheless, the continued presence of EUI-64 provides a valuable research opportunity. Analysing these addresses allows us to identify which device classes and vendors have not yet adopted privacy-preserving alternatives. This can shed light on the persistence of legacy or insecure configurations within the broader Internet ecosystem. Furthermore, because EUI-64 exposes MAC addresses, systematic manufacturer allocation patterns become visible during bulk institutional deployments with sequential addressing. Evaluating these configurations can uncover product families, specific hardware models, and internal manufacturing sequences, particularly when MAC addresses are assigned in a highly predictable or sequential order.

1.1 Contributions

In this work, we examine the landscape of EUI-64 usage to highlight the privacy and security implications that persist in production networks. Our findings provide actionable insights for academic IPv6 deployment studies and enterprise network administration. Specifically, our primary contributions are as follows:

- We demonstrate the prevalence of EUI-64, despite the long-existing privacy risks and its deprecation in 2017 [8].
- We analyse large-scale network datasets to identify EUI-64-generated IPv6 addresses.
- We uncover predictable address allocation and sequential patterns across different device categories and vendors.
- We compile actionable insights to accelerate IPv6 subnet scanning for IoT device discovery and categorisation.



This work is licensed under a Creative Commons Attribution 4.0 International License.
WTMC '26, The Hague, Netherlands
© 2026 Copyright held by the owner/author(s).
ACM ISBN 979-8-4007-3017-7/2026/11
https://doi.org/10.1145/3846375.3849092

2 Background and Related Work

EUI-64 uses the MAC address to generate an identifier. However, the generation process can be easily reversed, thus exposing the MAC address to the Internet. This process can be considered a leak of information, as MAC addresses are typically intended to be known only within the local network and used to send packets at the Ethernet layer [5]. Moreover, EUI-64 facilitates easier topology discovery within ISP networks. Li et al. [14] and Saidi et al. [24] observed EUI-64-generated identifiers and profiled the manufacturers using the format, identifying and categorising Customer Premises Equipment (CPE), such as routers, switches, and access points. Furthermore, Rye et al. [20, 23] and Zirngibl et al. [29] have observed the prevalence of EUI-64 in their recent large-scale IPv6 measurements and hitlists.

In addition to information leakage, EUI-64-based IIDs introduce significant privacy concerns for the devices that utilise them. Although the prefix will rotate for each network a device joins, the IID part of the IPv6 address, which is the last 64 bits of the address, remains unchanged for devices across networks. This is because the identifier is generated from the MAC address, which is set by the manufacturer. Saidi et al. [24] show that a single device at home with this identifier enables tracking and even compromises the end-user's privacy. Moreover, Rye et al. [19] reveal that CPE, which can be considered an IoT device, often has an EUI-64-based address on its internal port. This is still reachable from the WAN side and can thus be used to track across the entire network, since a device's IID remains the same. In addition, Rye et al. [18, 19] point out that unless an operator removes all devices in the network that use EUI-64, it is still possible to track these devices.

To help address the issues, Zohaib et al. [30] developed tools to automatically detect these privacy-leaking devices and provide even non-technical users with clear information about their behaviour. On the other hand, both Li et al. [14] and Rye et al. [19, 23] reported their findings, including the prevalence of EUI-64 in stakeholders' networks, to vendors and manufacturers, and suggested adopting privacy extensions and other measures. However, Rye et al. discussed a trade-off between adopting privacy extensions and the device deployment process [19], especially for deployed products and end-user motivation [23], thereby slowing the mitigation of EUI-64-related privacy issues. Moreover, end-users are often unaware of their home devices using EUI-64 addresses [24], unless informed by ISPs or other stakeholders [14, 19].

Although researchers have observed and analysed EUI-64 in their IPv6 measurements [4, 22, 23, 29], the format is not yet directly used for dedicated IPv6 scanning. Both Groat et al. [9] and Saidi et al. [24] utilised EUI-64 addresses for passive device and network tracking, rather than actively seeking out hosts. Closer to our analysis, Yang et al. [28] implemented a router fingerprinting model to label vendors by extracting OUIs from EUI-64 addresses. While most related work looked up the manufacturer to generalise about the device type, we have not found work that focuses on in-depth EUI-64 address patterns and vendor-specific assignments.

3 Methodology

We conducted our initial experiment in the IoT Lab, utilising the university network dataset to develop our pre-processing methods. Building on our experiences with the lab and the university network, we conducted our analysis on the dataset from the national research and education network.

3.1 Local Discovery in the IoT Lab

We conducted experiments on 27 distinct devices within our university's IoT lab to observe and analyse the real-world generation and utilisation of EUI-64 addresses. The experimental environment consisted of an isolated IPv6 test network managed by an OPNsense 24.7 router virtual machine hosted on a Proxmox server. To ensure strict network isolation, the IoT devices were segmented into dedicated VLANs via managed switches and access points. This topology allowed the router to enforce a firewall against unsolicited inbound connections while simultaneously monitoring subnet traffic. Device IP addresses were tracked via the router's Address Resolution Protocol (ARP) and Neighbor Discovery Protocol (NDP) caches [15]. For external connectivity, the router obtained an SLAAC address and a delegated prefix from the university network. We advertised this prefix on the test VLAN alongside Router Advertisements (RAs) and automated DHCPv6 [1], providing the latter to ensure DNS configuration for clients unable to obtain it directly from RAs. Following the normative requirements of RFC 4291 [3] and RFC 4862 [17], we utilised a standard /64 subnet prefix to allow connected devices to automatically derive their 64-bit interface identifiers. The architectural reasoning behind fixing this 64-bit boundary is further analysed in RFC 7421 [2].

After mapping the VLAN configurations to the devices' MAC addresses, the hardware was integrated into our dedicated measurement network. We then monitored the router's NDP cache, which logs the IPv6 addresses used by local devices to communicate with external networks. The cache initially showed two devices actively transmitting data to external services, each with a self-generated EUI-64 SLAAC address. Because several devices in the cache exhibited multiple IPv6 addresses, we supplemented our passive monitoring with active network discovery. Following the methodology proposed in RFC 7707 [7], we pinged the all-node link-local multicast address (`FF02::1`) with source from a local interface. To identify additional EUI-64-enabled infrastructure, we extracted the hardware MAC addresses from these local multicast responses, computed their corresponding global EUI-64 addresses using the delegated network prefix, and issued directed pings to those specific destinations. This active discovery phase identified seven responding devices.

Finally, combining the data from the NDP cache and active probing yielded eight unique EUI-64 devices. Seven of these devices constructed their IPv6 addresses directly from their manufacturer-assigned OUIs and hardware MAC addresses. Conversely, one device utilised a randomised MAC address to generate its EUI-64 identifier, thereby incorporating privacy-preserving address generation. Our empirical lab analysis confirms that hardware MAC addresses can still be reliably extracted from EUI-64 formatted IPv6 addresses. These findings demonstrate that despite well-documented privacy issues and modern architectural deprecations, various commercial IoT devices continue to rely on traditional EUI-64 address generation schemes.

3.2 Pre-Processing Methods with the University Network

Following the network topology of the lab setup, we turn to our university network, which serves as the foundation for our analysis method. This environment includes not only university staff and students but also residents and campus infrastructure, providing both wired and wireless access through services such as Eduroam [10] and IoTroam. The network connects a wide variety of devices, ranging from personal laptops and mobile phones to servers, routers, and virtual machines. This diversity allowed us to study EUI-64 deployment at a larger scale in a semi-open environment.

The dataset comprises IPv6 addresses, timestamps indicating when each address was last observed, and the MAC addresses recorded by campus routers for each IPv6 address. To extract the MAC address from the EUI-64 address, we developed a script that reverses the EUI-64 format for every entry in the dataset, compares each derived MAC against the provided address, and records whether the values match. We then combined the dataset with a list of OUIs retrieved from the IEEE OUI registry [13] to obtain a vendor name for each MAC address.

To categorise the devices, we grouped them by vendor names. In addition, we merged vendor groups that have multiple OUIs but do not use the same spelling in all their entries. For instance, TP-Link was observed as Tp-link or TP-LINK, while AMPAK varied between AMPAK Technologies,Inc and AMPAK Technologies, Inc, which is a minor difference in spacing. Finally, we performed manual scanning to check for consistent open ports, protocol responses, or ICMP behaviour between devices sharing similar MAC address prefixes, to confirm whether devices in a group likely belonged to the same device type.

3.3 Analysis Methods with the National Research and Education Network

The national research and education network (NREN) offers a significantly larger scale than the IoT Lab and the university network. It interconnects a wide range of universities, research institutes, and associated infrastructure. This leads to a more diverse set of devices and network configurations. Its dataset was sufficiently large to enable more in-depth analysis and broader coverage.

This dataset from the NREN is derived from flow data that was pre-filtered for 64-bit IIDs containing the `FF:FE` pattern. As we do not have the original MAC addresses to confirm the EUI-64 format, we instead matched the extracted OUIs against the IEEE registry, dropping entries with unrecognised OUIs [1].

3.3.1 Statistical Patterns in MAC Usage. To quantify the distribution and concentration of the MAC addresses extracted from EUI-64 IPv6 addresses, we applied Shannon Entropy (Ent) [25], Gini Index (Gini) [12], and Top 5 coverage (T_5) within each manufacturer's subset. Moreover, we computed the metrics on the NIC-specific part at two granularities: Prefix, the first two characters of the NIC portion (byte 4 of the EUI-64), and Full NIC, the full three-byte NIC portion (bytes 4 through 6 of the EUI-64). Grouping the dataset by vendor OUI, we use the Gini Index to measure inequality in the distribution of NIC values and Top 5 coverage to measure MAC-address concentration. This coverage represents the proportion of observations from the five most frequent Prefixes and Full NICs. Alongside entropy, these metrics characterise each manufacturer's MAC-address diversity, concentration, and reuse. While they do not conclusively reveal the underlying manufacturing process or sequential assignment, these patterns can suggest behaviours like MAC-address randomisation or repeated address generation.

3.3.2 Group Detection with DBSCAN. To detect allocation patterns in MAC addresses assigned by specific manufacturers, we used Density-Based Spatial Clustering of Applications with Noise (DBSCAN) [26]. DBSCAN can effectively process the dataset because it discovers natural, density-based groupings without a predefined number of clusters. Each MAC address was represented as a 6-byte vector, and Euclidean distance was used to measure similarity between addresses. We chose this metric because it treats each byte equally, effectively capturing positional similarities like shared prefixes. We set the minimum sample parameter to 8 by trial and error. In DBSCAN, this threshold controls the density required for a core point, not the final cluster size, so clusters with fewer than 8 members can still occur when border points are attached. The parameter provided the best balance between forming meaningful clusters and ensuring coverage across manufacturers. To determine the optimal maximum distance (EPS), we developed an automated script. Starting at 1.0, the script incrementally increases EPS per OUI until reaching 6 to 39 clusters per vendor subset. The lower limit guarantees enough data for comparison, while the upper limit keeps manual analysis manageable. Ultimately, these bounds act as pragmatic workload constraints rather than statistical thresholds.

3.3.3 Group Cohesion with Byte-Wise Entropy. To determine how tightly clustered the DBSCAN-identified clusters are and to infer manufacturer allocation strategies, we selected Byte-Wise Entropy for the investigation. Within each cluster, we computed the entropy of each byte position in the MAC addresses. Lower entropy indicates structured, predictable address assignment patterns, while higher entropy suggests randomness or dispersion. This byte-level analysis allows us to quantify the underlying structure and group behaviour of MAC address allocations from different vendors.

4 Ethical Considerations

Our study is based on three different datasets: experiment results from our IoT lab, a dataset from our university network, and a dataset from our NREN. We discuss the ethical considerations of each separately:

The IoT lab is a walled-off network that only contains devices explicitly used for experimentation. We were the only users of the lab during our network measurements, so the dataset with addresses did not involve any personal data.

For our campus network, we first sought permission from our ethics committee and discussed GDPR compliance with our privacy officer before being able to access the data. The university ICT team provided the network database, which contains IPv6 addresses and MAC addresses, along with associated VLANs, for a single year. This database was built from snapshots of the Neighbor Discovery

[1] Randomly generated IPv6 addresses may coincidentally include the `FF:FE` pattern, but given that we filter for valid OUIs, we consider this insignificant to our results.

Tables of the IPv6 gateways on campus. Any concerning findings were immediately discussed with the ICT team.

The final dataset is from our NREN. We reached an agreement with the NREN to acquire a dataset of EUI-64-enabled devices, comprising flow data from the entire national network over two months. These flow captures represent a snapshot of traffic to or from devices using IPv6 addresses that followed the EUI-64 generation pattern, such as `FF:FE` insertion, across the network. To avoid privacy concerns, the data was filtered in advance by the provider to include only the relevant part of the IPv6 address, the IID, which contains the embedded MAC address information.

5 Results

We present the results of our investigation into EUI-64-based IPv6 addresses and the MAC addresses embedded within, starting with a brief overview of our university network dataset and then focusing on our NREN dataset.

5.1 Results of the University Network

From our university network dataset, we identified 17,639 (11%) of 160,155 devices with unique MAC addresses across 44 VLANs that used EUI-64-generated IPv6 addresses. Of the EUI-64-based addresses, 470 had been created in multiple VLANs. This shows not only that some manufacturers still use EUI-64, but also that tracking is possible, as suggested by Groat et al. [9].

After processing the dataset, three clear trends emerged: a large number of MAC addresses were associated with chip manufacturers, several were associated with hypervisor or virtualisation vendors, and server manufacturers were heavily represented. In contrast, relatively few handheld device vendors were found, which aligns with expectations. As discussed in Section 2, most mobile operating systems now default to using privacy extensions or randomised IIDs, making them less likely to appear in this dataset. Nonetheless, EUI-64 is used not just by IoT devices, but also by servers and virtual machines, often within local infrastructure. The brief results here served as our foundation to analyse the NREN dataset.

5.2 Results of the National Research and Education Network

The dataset comprises 4,112,525 MAC addresses derived from IIDs across 22,292 OUIs over two months. Within the dataset, 4,074,593 addresses (99%) match the IEEE OUI registry and correspond to 10,539 unique OUIs. The remaining 11,753 unassigned OUIs, accounting for 37,932 addresses (1%), represent false positives, which are excluded from our analysis.

To be noted, the dataset shares 649 valid, unique MAC addresses with the university dataset. This confirms the persistence of MAC-assigned IPv6 addresses across different network layers, regardless of the different administrations. We summarise our findings and highlight the key takeaways from our analysis.

5.2.1 Vendor Distribution and Outlier. We present the frequently observed manufacturers and vendors, sorted by the number of MAC addresses, in Table 1. This table also includes the most observed vendors in our university dataset, as well as those that stood out. While the device population comprises various IoT vendors, chip manufacturers, and CPE producers, a single vendor dominates the NREN dataset. Specifically, **AVM**, the manufacturer of the Fritzbox router, accounts for 2,790,365 of the 4,112,525 total addresses (67.9%). This high volume of consumer CPE in an academic network likely stems from router deployments in university-affiliated residential settings, such as student housing or dormitories. This discovery also resembles the frequent CPE observations over the last 5 years [14, 20, 28, 29], as discussed in Section 2. Additionally, we identified server and desktop hardware vendors, and a small number of mobile device brands. On the other hand, 361,577 addresses under the **IEEE Registration Authority** likely represent unregistered placeholder OUIs [20]. We excluded the outlier from our analysis to avoid distorting the statistics.

Table 1: Vendor Distribution and MAC Usage.

Vendor	MAC	Prefix			Full NIC		
		Ent	Gini	T_5	Ent	Gini	T_5
AVM	2,790,365	7.9744	0.1063	2.86	21.2399	0.0772	0.00
DWnet	124,507	4.6439	0.8381	29.19	16.9023	0.0116	0.01
Sagemcom	87,758	6.2395	0.7399	16.20	16.4175	0.0019	0.01
Amazon	87,083	7.9979	0.0302	2.24	16.4046	0.0027	0.01
Sonos	76,182	7.8131	0.2725	4.74	16.2103	0.0034	0.01
Ogemray	65,387	4.9666	0.7275	36.66	15.9967	0.0000	0.01
Synology	54,628	7.8078	0.2831	4.47	15.7369	0.0002	0.02
Huawei	32,986	7.9496	0.1489	3.57	15.0073	0.0012	0.03
Century	30,550	5.6651	0.3926	19.51	14.8989	0.0000	0.02
TP-Link	19,747	7.8780	0.2305	4.49	14.2684	0.0005	0.05
ZTE	17,528	7.8577	0.2480	5.11	13.8376	0.0777	0.39
QNAP	14,301	6.4788	0.4984	18.32	13.8038	0.0000	0.03
Samsung	13,757	7.9791	0.0959	2.78	13.7469	0.0005	0.07
Sercomm	13,545	2.6251	0.9462	93.54	13.7255	0.0000	0.04
VMware	12,552	6.0697	0.5937	42.59	13.6152	0.0002	0.06
Xiaomi	9,933	7.5103	0.4562	7.47	13.2776	0.0002	0.07
Motorola	9,768	7.9479	0.1467	3.42	13.2528	0.0005	0.10
Intel	8,427	7.9365	0.1496	5.30	13.0391	0.0008	0.12
Asus	8,235	7.7373	0.2748	9.93	13.0073	0.0001	0.07
Proxmox	6,865	7.9774	0.0996	2.84	12.7450	0.0000	0.07
Xensource	6,175	5.8098	0.5448	42.30	12.5922	0.0000	0.08
Dell	5,932	7.7968	0.2750	7.52	12.5343	0.0000	0.08
Raspb. Pi	5,531	7.9235	0.1787	4.32	12.4330	0.0002	0.11
Azurewave	5,298	7.0062	0.6023	16.35	12.3705	0.0004	0.13
HP	4,679	7.7230	0.3020	9.83	12.1920	0.0000	0.11
Ampak	3,586	7.3591	0.4924	11.52	11.8076	0.0003	0.17
SWSoft	1,854	0.6102	0.9379	96.01	10.8564	0.0000	0.27
Philips	870	5.1181	0.6444	30.69	9.7649	0.0000	0.57
Google	544	7.5437	0.3022	6.25	9.0875	0.0000	0.92

5.2.2 Statistical Patterns in MAC Usage. We identified behaviours consistent with MAC address randomisation or repetition. Based on Table 1, two distinct types of manufacturer behaviour emerged from this analysis:

Concentrated Allocation. Low entropy and high concentration in the NIC portion suggest the reuse of very few address blocks, thereby increasing the risk of device identification and tracking.

Manufacturers such as **Sercomm**, **DWnet**, **Philips**, and **Shenzhen Ogemray (Ogemray)** exhibit high prefix centralisation ($T_{5\%}$) and low entropy, suggesting serialised NIC assignment, likely due to factory-flashed firmware or hardware provisioning process. In contrast, **SWSoft** displays highly concentrated NIC prefixes despite diverse full three-byte NIC portions, suggesting its addresses are assigned from a constrained set of prefix ranges.

Concentrated MAC address assignments correlate with specific operational constraints, such as hardware manufacturing and virtualisation. This results in highly predictable and non-randomised EUI-64 IPv6 addresses.

Diversified Allocation. Even distribution and less concentration in the NIC portion suggest either diverse production batches or privacy-driven randomisation. Manufacturers and vendors, including **Amazon**, **AVM**, and **Samsung**, exhibited high entropy and low Gini value, reflecting highly varied prefixes and full NIC values. This suggests complex manufacturing and decentralised pipelines using multiple NIC suppliers. Meanwhile, **Synology**, **Sonos**, and **Proxmox** also displayed wide diversity in NIC components, with no strong clustering, suggesting address-assignment strategies that resist easy discovery.

Diversified MAC address assignment, driven by supply chain complexity or proactive privacy measures, can complicate device tracking and fingerprinting via EUI-64. Such an address assignment reduces the privacy risks.

5.2.3 Group Detection with DBSCAN. Our DBSCAN clustering identified 355 groups, each distinguished by OUIs. We present the average cluster sizes and densities in Table 2, which varied significantly across vendors. These clustering results highlight which vendors utilise structured or iterative address assignment processes:

- **DWNet** formed tightly grouped clusters with an EPS of 5.0. Its largest cluster contained 75,231 MAC addresses, making it the most prominent vendor in the dataset.
- **Sagemcom** got clusters with an EPS of 7.0, with the largest group reaching 68,305 addresses. This suggests large-scale uniform deployment, likely for modems and routers provided by ISPs.
- **Synology** produced loosely grouped clusters at an EPS of 14.0, with its largest group having 45,238 addresses. This indicates that MAC address assignment patterns are internally consistent across its NAS systems and network appliances.
- **Xensource** formed small clusters at an EPS of 4.0, with its largest cluster consisting of 2,278 addresses. This corresponds to its virtual machines, which are assigned MAC addresses systematically.

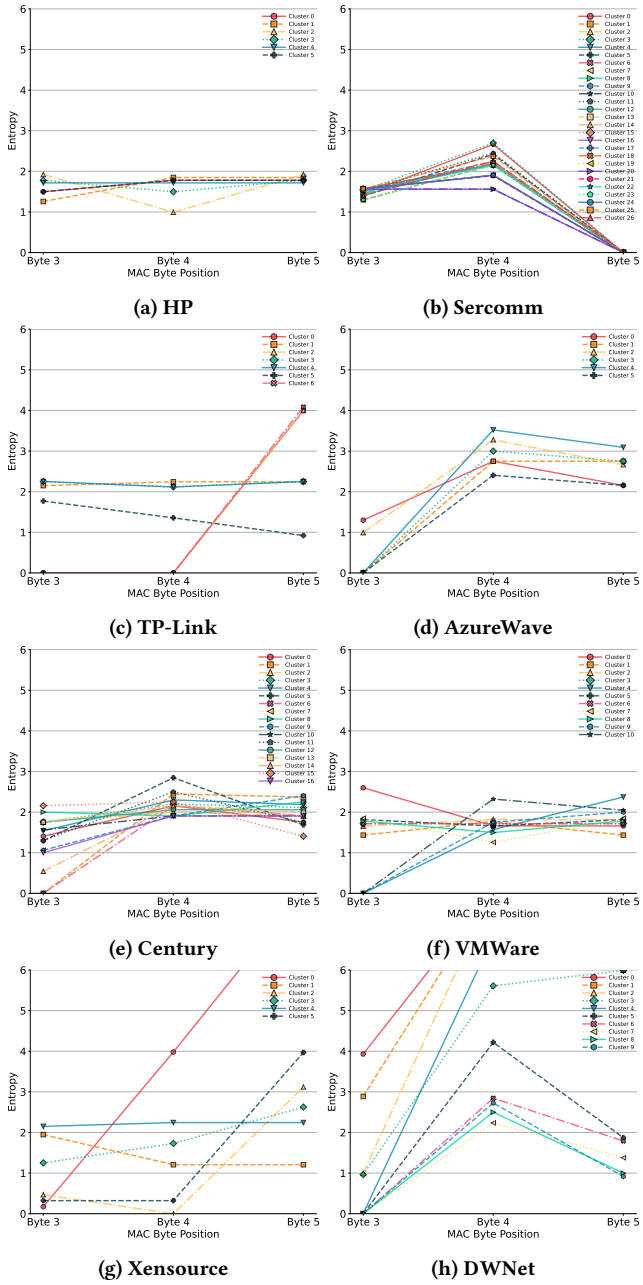
The DBSCAN-based group detection shows that MAC address assignment patterns are highly vendor-specific and reveal how different types of hardware are deployed. Manufacturers of large-scale physical infrastructure, such as **Sagemcom** and **Synology**, use a highly structured, predictable address pattern that forms massive clusters, while **Xensource**, with its virtual environments, forms smaller, tighter clusters.

5.2.4 Group Cohesion with Byte-Wise Entropy. We generated entropy plots for manufacturers and vendors, which are presented in Figure 1 with our observations as follows:

Table 2: Vendor Group Detection with Clusters.

Vendor	EPS	Cluster	Max	Min	Avg
DWnet	5	10	75,231	7	12,397.7
Sagemcom	7	6	68,305	8	11,391.5
Ogemray	10	22	47,859	8	2,958.82
Synology	14	18	45,238	2	3,003.72
Xensource	4	6	2,278	10	396.00
TP-Link	4	7	48	10	28.86
Intel	3	6	34	7	21.83
Ampak	15	8	29	8	11.50
VMware	2	11	26	8	16.45
SWSoft	1.5	9	25	8	11.67
ZTE	15	12	22	5	9.58
Dell	2	7	20	8	15.57
ASUSTek	2	6	19	14	17.33
HP	2	6	19	16	17.67
QNAP	5	9	18	8	10.89
Philips	14	11	17	8	10.09
Rasp. Pi	17	10	16	8	9.90
Google	59	7	15	8	10.43
Azurewave	12	6	14	8	9.50
Proxmox	10	12	14	6	8.83
Sercomm	2	27	14	8	9.15
Xiaomi	11	7	13	8	9.00
Huawei	14	11	12	8	9.09
Motorola	19	9	12	8	9.11
Amazon	14	16	11	8	8.25
Century	3	17	11	8	8.53
Samsung	23	6	11	5	8.00
Sonos	5	9	11	8	8.67
AVM	2.3	22	9	8	8.14

- **Hewlett-Packard (HP)** showed consistently low entropy across its clusters, indicating tight groupings and a sequential address assignment.
- **Sercomm** displayed zero entropy in the final byte of some clusters, meaning these MAC addresses only differ in their more significant earlier bytes.
- **TP-Link** had identical values in bytes 3 and 4, with variation only in byte 5, the final part. This is a clear indication of highly systematic, sequential grouping.
- **AzureWave** and **Century Trading (Century)** both show a pattern where byte 3 is identical, and only the last two bytes vary, suggesting that they group MAC addresses to manage a limited address space.
- **VMWare** produced scattered results with high entropy, rather than the expected sequential MAC generation for virtual machines. The dispersion may be caused by virtual machines being removed without their addresses being reused.
- **Xensource** showed strong, low-entropy grouping patterns, consistent with our expectations for virtual machine deployments. Its cluster 2 and, especially, cluster 5, exhibit the expected low-entropy pattern.



(a) HP (b) Sercomm

(c) TP-Link (d) AzureWave

(e) Century (f) VMWare

(g) Xensource (h) DWNet

Figure 1: Vendor Group Cohesion with Entropy Plot

- **DWNet** had extremely high entropy values with no obvious pattern, making it too heterogeneous or inconsistently labelled for meaningful entropy analysis.

Analysing the byte-wise entropy of the MAC addresses successfully uncovers the internal allocation strategies of different manufacturers. While hardware vendors, such as **HP**, **Sercomm**, and **TP-Link**, use highly deterministic, sequential assignment strategies that often differ by only a single byte, other environments exhibit unexpected dispersion, such as **VMware**, or complete heterogeneity, as in **DWNet**. To conclude, the entropy serves as a reliable metric for identifying whether a manufacturer relies on structured grouping or randomised address assignments.

6 Security and Scanning Suggestions

Our analysis of 4 million EUI-64-enabled IPv6 addresses reveals concentrated vendor allocations and predictable MAC address generation, including fixed NIC bytes and cloned MAC addresses. Adversaries could exploit these to map network topologies, even extending to non-EUI-64 devices [23, 24]. To mitigate these risks, we urge vendors and manufacturers to adopt privacy extensions or semantically opaque interface identifiers [6, 8]. Furthermore, network operators shall consider promoting these privacy measures [19, 23] and informing the unaware end-users [24].

To accelerate EUI-64-enabled device discovery and mitigate the associated risks, we recommend that network operators leverage the predictable patterns we identified. EUI-64 addresses already reduce the /64 subnet search space from 2^{64} to 2^{48} by setting the middle 16 IID bits to `FF:FE`. By targeting specific vendor OUIs and concentrated NIC prefixes, such as **TP-Link** and **Sercomm** (as shown in Table 1), operators can reduce the search space from millions to mere thousands of addresses per subnet. Moreover, using Table 2 and Figure 1 to identify devices with constant or sequential leading NIC bytes can further reduce the final search space to 2^8. This enables efficient subnet scanning by vendor or device type.

7 Conclusion

Despite known privacy risks for over a decade, EUI-64 remains prevalent, particularly within IoT and virtualised environments. Since embedded MAC addresses expose manufacturer information and production categories, predictable NIC patterns with selective OUI enable highly targeted IPv6 scanning by both malicious actors and network defenders.

Our clustering and pattern-detection analysis provides a targeted candidate list for IPv6 subnet scanning using EUI-64. In addition, flow data and router neighbour tables help identify active EUI-64 IPv6 addresses. Constrained by dataset scale and the complex manufacturer behaviours, our suggestions cannot support network scanning at scale, but only within subnets. Nevertheless, this work provides insights into EUI-64 usage and supports network operators and vendors in measuring and mitigating the risks EUI-64 poses.

Acknowledgments

This work is supported by the INTERSECT project, Grant No. NWA 1160.18.301, funded by the Netherlands Organisation for Scientific Research (NWO). We gratefully acknowledge the active support of the management teams of the IoT Cyberlab and the ICT infrastructure service (LISA) at the University of Twente. Above all, we extend our deepest gratitude to the administrators at SURF, the Dutch National Research and Education Network, whose support was vital to this research. In particular, we appreciate the insightful feedback from the Workshop on Traffic Measurements for Cybersecurity (WTMC) program committee, which helped us refine our work. Finally, we note the use of Gemini Pro 3.1 for grammar editing.

References

[1] Michael Carney, Charles E. Perkins, Bernie Volz, Ted Lemon, and Jim Bound. 2003. *Dynamic Host Configuration Protocol for IPv6 (DHCPv6).* Request for Comments RFC 3315. Internet Engineering Task Force. doi:10.17487/RFC3315

[2] Brian E. Carpenter, Tim Chown, Fernando Gont, Sheng Jiang, Alexane Petrescu, and Anew Yourtchenko. 2015. *Analysis of the 64-Bit Boundary in IPv6 Addressing.* Request for Comments RFC 7421. Internet Engineering Task Force. doi:10.17487/RFC7421

[3] Steve E. Deering and Bob Hinden. 2006. *IP Version 6 Addressing Architecture.* Request for Comments RFC 4291. Internet Engineering Task Force. doi:10.17487/RFC4291

[4] Oliver Gasser, Quirin Scheitle, Sebastian Gebhard, and Georg Carle. 2016. Scanning the IPv6 Internet: Towards a Comprehensive Hitlist. doi:10.48550/arXiv.1607.05179

[5] George Thomas. 2011. Understanding Ethernet Switches and Routers. *ESSENTIALS: A Technical Supplement to Control Network* (2011). https://ccontrols.com/pdf/Essentials0411.pdf

[6] Fernando Gont. 2014. *A Method for Generating Semantically Opaque Interface Identifiers with IPv6 Stateless Address Autoconfiguration (SLAAC).* Request for Comments RFC 7217. Internet Engineering Task Force. doi:10.17487/RFC7217

[7] Fernando Gont and Tim Chown. 2016. *Network Reconnaissance in IPv6 Networks.* Request for Comments RFC 7707. Internet Engineering Task Force. doi:10.17487/RFC7707

[8] Fernando Gont, Alissa Cooper, Dave Thaler, and Will (Shucheng) LIU. 2017. *Recommendation on Stable IPv6 Interface Identifiers.* Request for Comments RFC 8064. Internet Engineering Task Force. doi:10.17487/RFC8064

[9] Stephen Groat, Matthew Dunlop, Randy Marchany, and Joseph Tront. 2011. IPv6: Nowhere to Run, Nowhere to Hide. In *2011 44th Hawaii International Conference on System Sciences.* 1–10. doi:10.1109/HICSS.2011.258

[10] GÉANT. 2026. Eduroam. https://eduroam.org/

[11] Ala Hamarsheh. 2025. Assessing the Progress of Transition to IPv6: A Global Perspective. *IETE Journal of Research* (April 2025), 1–15. doi:10.1080/03772063.2025.2490718

[12] Joe Hasell. 2023. Measuring Inequality: What Is the Gini Coefficient? *Our World in Data* (June 2023). https://ourworldindata.org/what-is-the-gini-coefficient

[13] IEEE. 2026. Standards OUI. https://standards-oui.ieee.org/

[14] Xiang Li, Baojun Liu, Xiaofeng Zheng, Haixin Duan, Qi Li, and Youjun Huang. 2021. Fast IPv6 Network Periphery Discovery and Security Implications. In *2021 51st Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN).* 88–100. doi:10.1109/DSN48987.2021.00025

[15] Thomas Narten. 1999. Neighbor Discovery and Stateless Autoconfiguration in IPv6. *IEEE Internet Computing* 3, 4 (July 1999), 54–62. doi:10.1109/4236.780961

[16] Thomas Narten, Richard P. Draves, and Suresh Krishnan. 2007. *Privacy Extensions for Stateless Address Autoconfiguration in IPv6.* Request for Comments RFC 4941. Internet Engineering Task Force. doi:10.17487/RFC4941

[17] Thomas Narten, Tatsuya Jinmei, and Susan Thomson. 2007. *IPv6 Stateless Address Autoconfiguration.* Request for Comments RFC 4862. Internet Engineering Task Force. doi:10.17487/RFC4862

[18] Erik Rye. 2022. Defeating IPv6 Prefix Rotation Privacy. https://blog.apnic.net/2022/01/31/defeating-ipv6-prefix-rotation-privacy/

[19] Erik Rye, Robert Beverly, and K C Claffy. 2021. Follow the Scent: Defeating IPv6 Prefix Rotation Privacy. In *Proceedings of the 21st ACM Internet Measurement Conference (IMC '21).* Association for Computing Machinery, New York, NY, USA, 739–752. doi:10.1145/3487552.3487829

[20] Erik Rye and Dave Levin. 2023. IPv6 Hitlists at Scale: Be Careful What You Wish For. In *Proceedings of the ACM SIGCOMM 2023 Conference (ACM SIGCOMM '23).* Association for Computing Machinery, New York, NY, USA, 904–916. doi:10.1145/3603269.3604829

[21] Erik Rye, Jeremy Martin, and Robert Beverly. 2019. EUI-64 Considered Harmful. arXiv:1902.08968 [cs] doi:10.48550/arXiv.1902.08968

[22] Erik C. Rye and Robert Beverly. 2020. Discovering the IPv6 Network Periphery. In *Passive and Active Measurement*, Anna Sperotto, Alberto Dainotti, and Burkhard Stiller (Eds.). Springer International Publishing, Cham, 3–18. doi:10.1007/978-3-030-44081-7_1

[23] Erik C. Rye and Robert Beverly. 2023. IPvSeeYou: Exploiting Leaked Identifiers in IPv6 for Street-Level Geolocation. In *2023 IEEE Symposium on Security and Privacy (SP).* 3129–3145. doi:10.1109/SP46215.2023.10179376

[24] Said Jawad Saidi, Oliver Gasser, and Georgios Smaragdakis. 2022. One Bad Apple Can Spoil Your IPv6 Privacy. *SIGCOMM Comput. Commun. Rev.* 52, 2 (June 2022), 10–19. doi:10.1145/3544912.3544915

[25] ScienceDirect. 2026. Shannon Entropy. https://sciencedirect.com/topics/engineering/shannon-entropy

[26] scikit-learn. 2026. DBSCAN. https://scikit-learn.org/stable/modules/generated/sklearn.cluster.DBSCAN.html

[27] Junaid Latief Shah and Heena Farooq Bhat. 2020. Towards a Secure IPv6 Autoconfiguration. *Information Security Journal: A Global Perspective* 29, 1 (Jan. 2020), 14–29. doi:10.1080/19393555.2020.1716117

[28] Yifan Yang, Ling Hu, Tao Yang, Xionglve Li, Bingnan Hou, and Zhiping Cai. 2026. Efficient Router Fingerprinting in IPv6 Networks. *Comput. J.* (Feb. 2026). doi:10.1093/comjnl/bxag004

[29] Johannes Zirngibl, Lion Steger, Patrick Sattler, Oliver Gasser, and Georg Carle. 2022. Rusty Clusters? Dusting an IPv6 Research Foundation. In *Proceedings of the 22nd ACM Internet Measurement Conference (IMC '22).* Association for Computing Machinery, New York, NY, USA, 395–409. doi:10.1145/3517745.3561440

[30] Ali Zohaib and Amir Houmansadr. 2023. Automated Detection of IPv6 Privacy Leakage in Home Networks. *Free and Open Communications on the Internet* (2023). https://petsymposium.org/foci/2023/foci-2023-0005.php