Analyzing IoT Hosts in the IPv6 Internet

Peter Jose
Max Planck Institute for Informatics
pjose@mpi-inf.mpg.de

Said Jawad Saidi
Max Planck Institute for Informatics
jsaidi@mpi-inf.mpg.de

Oliver Gasser
Max Planck Institute for Informatics
oliver.gasser@mpi-inf.mpg.de

***Abstract*—Users and businesses are increasingly deploying Internet of Things (IoT) devices at home, at work, and in factories. At the same time, we see an increase in the use of IPv6 for Internet connectivity. Even though the IoT ecosystem has been the focus of recent studies, there is no comprehensive analysis of IoT end-hosts in the IPv6 Internet to date.**

In this paper we perform an in-depth analysis of IPv6-reachable IoT hosts using active measurements. We run measurements targeting 530M IPv6 addresses on six popular IoT-related protocols. With 36.4K hosts in 156 countries we find 380× fewer IoT-speaking end-hosts compared to IPv4. Moreover, we conduct a security analysis for TLS-enabled IoT-speaking hosts identifying up to 57% untrusted certificates, with up to 32% being self-signed and 25% being expired. Finally, we plan to publish our measurement results, tools, and a website dashboard to foster further research in the area.

I. Introduction

The Internet of Things (IoT) has become increasingly prevalent over the last few years. Consumers use IoT devices for entertainment (e.g., smart TVs), home automation (e.g., Google Home, Amazon Alexa), or home surveillance (e.g., Ring Home Security System). In addition, the industry is also making use of IoT (i.e., industrial IoT) to automate production processes. Studies estimate that by 2025 there will be 28 billion IoT devices [2].

Consequently, the IoT ecosystem has been the focus of research in recent years. Several studies have analyzed various aspects of the IoT ecosystem, such as identifying IoT devices in the Internet [29], [33], [41], [42], [48], leakage of privacy-sensitive information by IoT devices [45], [47], and servers contacted by IoT devices [45], [49]

Despite the growing importance of IPv6 [28], [44], the IoT ecosystem in the IPv6 Internet has remained understudied. This may be due to various factors such as difficulty in identifying target addresses in the vast IPv6 address space [25], [26], a lack of IPv6 support of IoT clients until recently [32], and relatively low deployment of IPv6 in ISP networks [13]. Moreover, commercial network intelligence platforms such as Censys only started to release IPv6 data recently [11]. Similar to IPv4 [54], however, we find these platforms to have relatively low coverage of IoT-protocol-speaking hosts in the IPv6 Internet.

Due to these shortcomings in the state-of-the-art, we currently lack a good understanding of IoT-speaking end-hosts in IPv6. Therefore, this paper aims to address this gap by detecting, characterizing, and analyzing the deployment of IoT-speaking end-hosts in the IPv6 Internet using active measurements. Specifically, this work makes the following main contributions:

- **IoT IPv6 deployment:** We perform a large-scale active measurement study on the IPv6 Internet. We target $530M$ IPv6 addresses for six IoT-related protocols (AMQP, CoAP, MQTT, OPC UA, XMPP, and Telnet) running on eleven ports. In total, we find $36.4K$ hosts in 156 countries and 3177 ASes, respectively. Moreover, Telnet is the most frequent one in our measurements, and we find support for multiple protocols on the same host to be relatively rare. Furthermore, we find IoT-speaking end-hosts to be relatively stable in terms of responsiveness 62.5%.
- **Security analysis:** Next, we perform an in-depth study of TLS security properties of IoT-speaking end-hosts. For XMPP and MQTT, we find that most of these end-hosts support the latest version of TLS, whereas this is only the case for less than 40% of AMQP end-hosts. We analyze TLS certificates sent by these end-hosts and find that 19.2% of them are self-signed and 45.0% are untrusted. Additionally, we find that 25.0%, 23.7%, and 13.0% of certificates are expired for AMQPs, MQTTs, and XMPPs, respectively.
- **Measurement tools and dashboard:** To run our active measurement study, we develop custom extensions to ZMap and ZGrab2. To foster further research in the IoT ecosystem by fellow scientists, we publish our custom-developed measurement tools, including ZMap probe packets for DTLS and ZGrab2 modules for AMQP, CoAP, DTLS, MQTT, and XMPP [4]. Moreover, we plan to publish raw measurement results to foster reproducibility in the Internet measurement community. Finally, we provide a publicly accessible dashboard as an easy way to interact with our results [3].

II. Background

This section provides background information on the popular IoT protocols considered in this study. Internet and sensing networks combine to form the IoT paradigm allowing machine-to-machine communication [27]. An IoT network consists of IoT devices and servers interacting and performing their respective roles. IoT devices have different computation power ranging from simple embedded sensors with limited resources (computational power, energy, and memory) to advanced and powerful ones. Usually, the protocols developed for IoT devices tend to be lightweight to accommodate constrained resources [8], [27].

The most commonly used protocols in IoT applications that we scan are the Message Queuing Telemetry Trans-

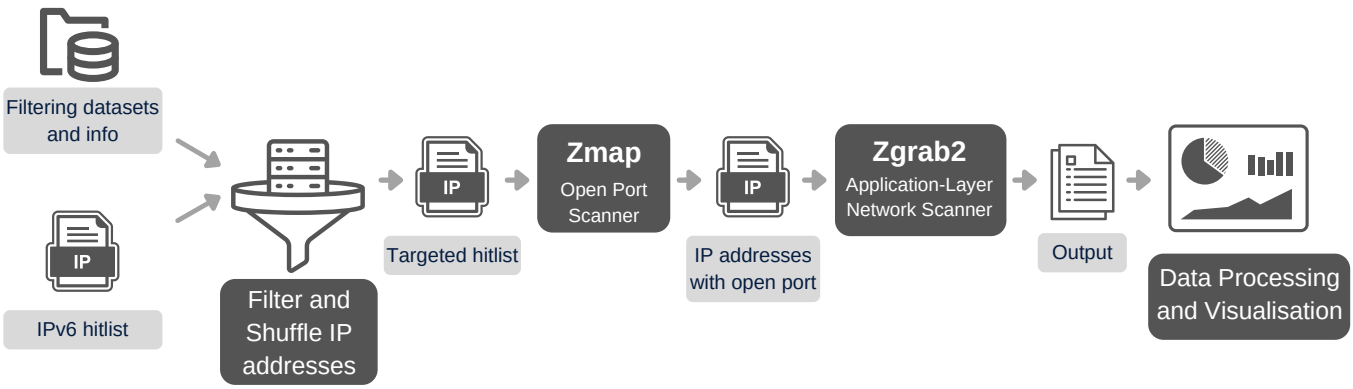

Fig. 1: Scanning pipeline for discovering IoT-speaking hosts in the IPv6 Internet.

port **(MQTT)** protocol, the Constrained Application Protocol **(CoAP)**, the Extensible Messaging and Presence Protocol **(XMPP)**, the Advanced Message Queuing Protocol **(AMQP)**, the Open Platform Communication Unified Architecture **(OPC UA)** protocol, and the Telnet protocol [5], [23], [52]. These protocols use both TCP and UDP as their transport protocols. However, most of them primarily use TCP. For CoAP, UDP is recommended for constrained node networks owing to the larger resource requirement for CoAP over TCP [9]. Protocols over TCP use TLS, and those over UDP use DTLS in their secured versions. An exception is Telnet, which does not provide a secure version.

MQTT: MQTT follows a client-server publish/subscribe model designed in a lightweight event-driven approach making it ideal for usage in constrained environments.

CoAP: CoAP follows the client-server interaction model inspired by the Hypertext Transfer Protocol (HTTP). CoAP provides an option to increase the communication's reliability by using a confirmable message.

XMPP: XMPP is an open-source messaging protocol designed originally for text messaging and application-to-application messaging. It is an Extensible Markup Language (XML) and text-based protocol that uses both request/response and publish/subscribe architectures over TCP. Data in the form of XML stanzas [50] are used in the communication between the XMPP client and server.

AMQP: AMQP is a message-oriented, lightweight, and open-source protocol. This protocol is designed for the publish/subscribe and request/response architectures [23].

OPC UA: OPC UA is a cross-platform, open-source communication protocol that aims at the Service Oriented Architecture (SOA). The protocol supports both publish/subscribe and client/server methods. The entire specification of the OPC classic is enhanced and unified by the creation of OPC UA. The protocol is mainly deployed for industrial applications [23].

Telnet: Telnet provides an application-layer protocol providing bi-directional text-oriented communication via a terminal service. Due to the simplicity of Telnet, it is still very popular on embedded systems, even though it is not heavily used on servers and workstations anymore [37].

III. Methodology

In this section, we detail our methodology on identifying IoT-protocol speaking hosts in the IPv6 Internet. For this, we need to address several challenges namely, the vast IPv6 space address space, IoT protocol selection, and the lack of support for some IoT protocols in existing tools. We start by generating IPv6 targets. Next, we explain the rationale behind selecting the IoT protocols and finally the implementation details of our scanning tools. Figure 1 illustrates an overview of our methodology.

A. Target Generation

The vast address space of IPv6 makes it infeasible to scan all of it. Hence, we use the publicly available hitlist provided by Gasser et al. [26], [58], which is comparatively less biased and provides supplementary data aiding the filtration of aliased prefixes from the hitlist.

In the filtering step, we first identify and remove addresses in aliased prefixes. This helps us avoid overcounting hosts, i.e., avoiding a single machine responding to all addresses in a prefix. For this, we employ multi-level aliased prefix detection technique [25].

Next, complying with the best practices in active measurement, we remove prefixes which are listed on our internal blocklist. The blocklist contains prefixes from previous studies, where network administrators have asked us not to scan their addresses. See Section III-F for more details on ethical considerations. From here on, we refer to this filtered hitlist as the targeted hitlist.

B. Protocol Selection

Communication protocols are an integral part of IoT systems. The selection of one protocol suitable for different IoT applications is faced with several dilemmas that need to consider energy efficiency, security, and quality of service.

Protocol	Unsecured Port	Secure Port	Transport
CoAP	5683	5684	UDP
MQTT	1883	8883	TCP
XMPP	5222	5223	TCP
AMQP	5672	5671	TCP
OPC UA	4840	4843	TCP
Telnet	23	n/a	TCP

TABLE I: TCP/UDP based categorization of protocols and ports considered for our measurements.

Recent studies dealing with IoT-speaking hosts using active measurement considered protocols such as MQTT, AMQP, UPnP, CoAP, XMPP, and Telnet [5], [52], [54]. In our work, except for UPnP, we also consider these protocols. Moreover, we consider OPC UA, a unified version of the OPC classic with its service-oriented architecture [52]. For each protocol—with the exception of Telnet—, we consider both the secured and non-secured version. Considering these factors, we select the five protocols listed in Table I. Although some protocols can use TCP and UDP as transport layer protocols, we focus on only the most recommended ones. For example, using CoAP over UDP is recommended for constrained node networks owing to the larger resource requirement for CoAP over TCP [9].

Although IANA has specified standard secured and unsecured ports for most of our protocols, a protocol can still be served on non-standard ports [31]. For XMPP, we consider the port corresponding to the client communication. Moreover, IANA has yet to specify the secured port of XMPP; hence we use its conventional port 5223 [50]. In our study, except for XMPP, we only consider the standard ports for all the protocols. For a complete list of protocols and their ports, we refer to Table I.

In the subsequent sections, we refer to the secure version of each protocol with a suffix "s", e.g., the secure version of XMPP is referred to as "XMPPs".

C. Open Port Scans

Next, we scan for open ports on the targeted hitlist using a modified version of ZMap [20] that supports IPv6 [12], targeting one IoT port at a time. Among the various options provided by ZMap, we use two scan options: (1) TCP SYN scans to identify ports supporting TCP-based protocols and (2) application-specific UDP scans for ports supporting UDP-based protocols. Further, to support UDP-based protocols, we create custom probes for CoAP and DTLS. We send one probe per protocol per target IP address and record all IP addresses with at least one successful response.

D. Application-Layer Network Scans

Using the IP addresses with successful responses from the open port scan, we conduct application-layer handshakes for each protocol using ZGrab2 [56]. Except for Telnet, the default implementation of ZGrab2 does not support any of the protocols

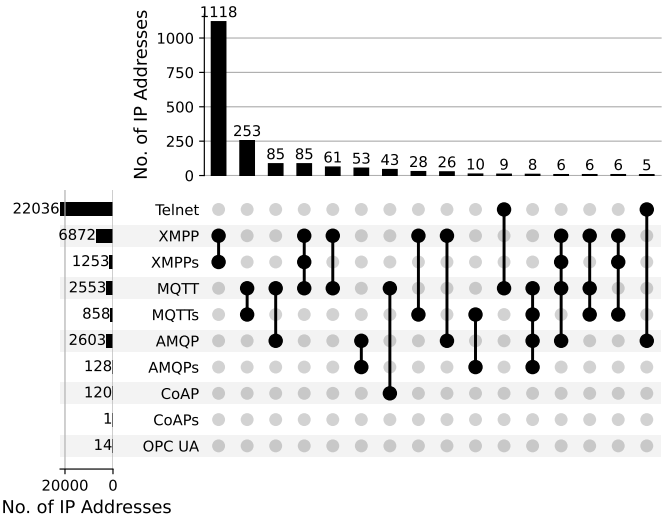

Fig. 2: Upset plot with the horizontal section showing the overall number of responsive IP addresses per protocol and the vertical section showing the number of IP addresses hosting different protocol combinations.

listed in Table I. We extend ZGrab2 by adding corresponding modules.

Furthermore, we modify ZGrab2 by adding TLS 1.3 support to analyze all standardized TLS versions. We also extend ZGrab2 to support DTLS 1.2, by incorporating the library by Pion [43] to add support for a secure version of our UDP-based protocols. We plan to release our ZGrab2 tools and modules to facilitate further research on IoT protocols and other UDP-based protocols.

In total we run three measurement campaigns over a period of six months, spanning from January to June 2022.

E. Limitations

Even though we use a multi-step scanning pipeline, our approach has limitations. First, protocols such as AMQP, XMPP, and Telnet can also be used for other non-IoT applications. Second, we do not differentiate between IoT devices and servers. Therefore, further investigation is needed to distinguish between them. Third, our coverage of IPv6 hosts is limited to our underlying hitlist. Finally, unlike the recent work in IPv4 [54], we do not apply honeypot detection techniques to infer whether a given target is a honeypot or not.

F. Ethical considerations

We take several measures to ensure that our scanning does not cause harm to routers or networks. To minimize the impact, we use a low measurement load, sending only one packet per destination and port. In addition, as by the policy of our institution, we filter out IP addresses belonging to Russia due to the Ukraine-Russian war to avoid raising alarms. We also randomly spread the load at each target IPv6 in the hitlist and coordinate with local network administrators to ensure that our scanning does not harm the local or upstream network.

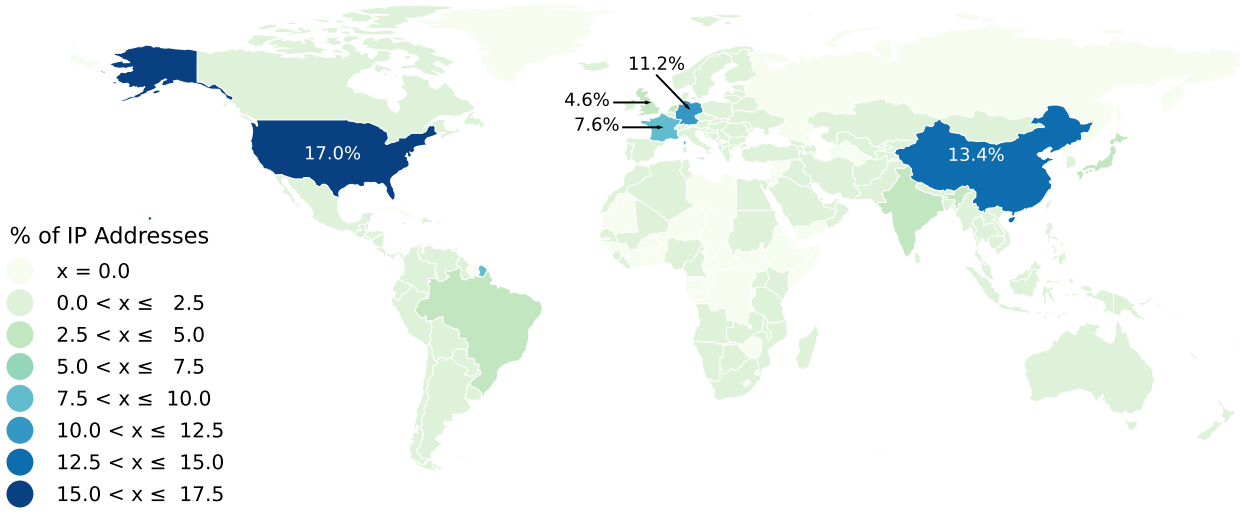

Fig. 3: Distribution of IP addresses hosting popular IoT protocols based on geographical location.

To ensure that our active scanning is ethical and does not result in complaints or opt-out requests, we follow the best current practices [18], [20], [40]. One of these practices is ensuring that our prober IP address has a meaningful DNS PTR record. We also set up a web server with experiment and opt-out information on the measurement machine. During our active experiments we did not receive any complaints or opt-out requests.

IV. IoT Hosts in the IPv6 Internet

In this section, we go through the results from our Internet-scale active measurements using different IoT protocols. We discuss the characteristics of the responsive hosts from a protocol, geographical, temporal, and network type perspective.

A. Responsive Protocols

We use results from our latest measurement (2022-06-29) to analyze the responsiveness of different protocols. Figure 2 shows the the popularity of each protocol in terms of the number of hosting IP addresses (horizontal bars) and the number of IP addresses hosting different protocol combinations (vertical bars). Although we observe 37 combinations of protocols in our data, we only illustrate those with at least two ports and five occurrences for readability.

Consistent with recent work in IPv4 [54], we observe that the Telnet protocol is the most popular with ∼22K IPs. However, we also find a relatively high number of XMPP-speaking hosts, making XMPP the second most popular protocol. We notice that there are particularly fewer CoAP hosts in IPv6 in comparison to previous IPv4 work. When doing a protocol-by-protocol comparison, we find 380× fewer IoT-speaking end-hosts in IPv6 compared to IPv4 (34.2K vs. 13M).

B. Geographical Characterization

Next, we present our analysis of responsive hosts from a geographical perspective based on our latest measurement (2022-06-29). We geo-locate responsive hosts using the MaxMind GeoLite2 database [55] on a country level. We first analyze the contribution of responsive IPv6 addresses per country, and subsequently, we drill-down and analyze the country-protocol distribution of IPv6 addresses.

In Figure 3, we examine the geographical distribution of the IP addresses that host at least one of our studied protocols. Our observation is that the distribution is skewed in terms of the number of countries represented. Specifically, we find that the United States, China, Germany, France, and the United Kingdom to account for more than 53% of all observed IP addresses. For the remaining countries, we can see a long tail in the distribution, with IP addresses distributed among 151 countries and no single country accounting for more than 4% of the total IPs.

Next, we analyze per-country differences for responsive protocols. In Figure 4, we break down the percentage of IP addresses that each country contributes for each protocol. We show the top 25 countries and group the remaining countries into the "Other" category. Our first takeaway is that all top 25 countries have some occurrences of our top protocols, namely, XMPP and MQTT. For XMPP, we also observe that the majority of IPs are in the United States. However, we find that for Telnet, 21.5% of hosts are attributed to China, which is in line with previous studies focusing on IPv4 [36].

C. Temporal Characterization

Next, we analyze temporal stability of IPv6 IoT-speaking end-hosts. To investigate this, we repeat our experiments three times over a six-month period, with each experiment conducted roughly two months apart. We first present an overview of the

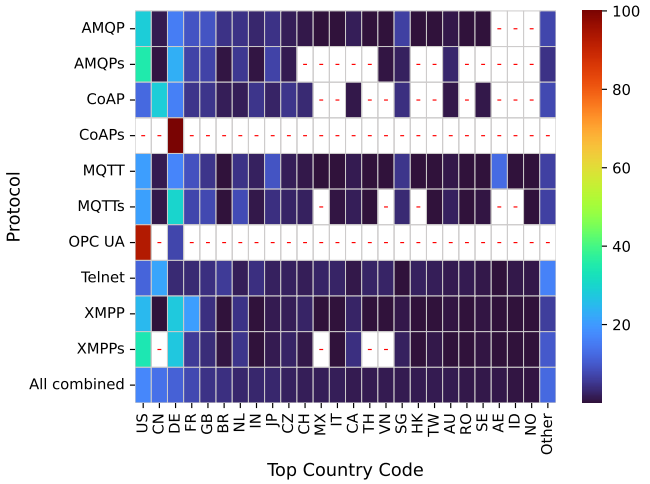

Fig. 4: Heatmap distribution of IP addresses hosting popular IoT protocols in different countries.

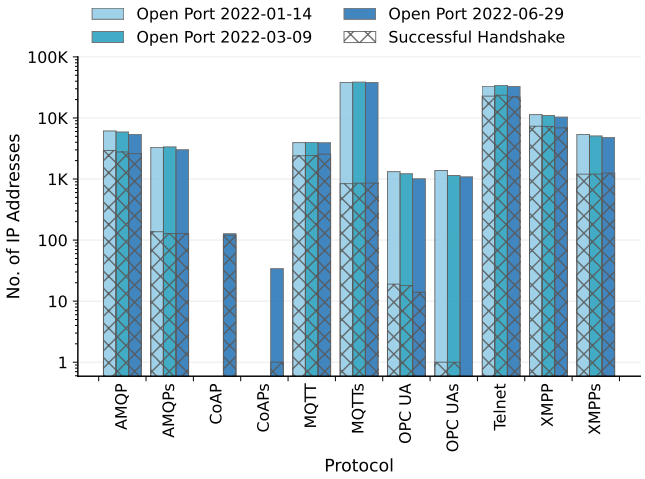

Fig. 5: Comparison of open ports and successful handshakes across our three measurements.

total number of responsive hosts in each of the three scans and then focus on the changes in the number of responsive hosts between subsequent scans.

In Figure 5, we illustrate the number of hosts discovered at each scan for different protocols. Each bar corresponds to the number of hosts with an open port for the respective protocol on the scan date. We annotate the number of hosts with a successful handshake by applying a lattice-like pattern. In the first two scans, our toolchain did not support the CoAP protocol, hence we only show the result for the latest scan.

From January to July 2022, we observe a decline in the total number of responsive IPv6 addresses in our underlying hitlist [25], before increasing again in July 2022. In line with this, we observe a slight decline in the number of responsive hosts for the majority of the protocols. Nevertheless, the proportion of successful handshakes remains consistent. For the MQTT and MQTTs protocols, the number of hosts with an open port remains stable and we also observe a slight increase in the

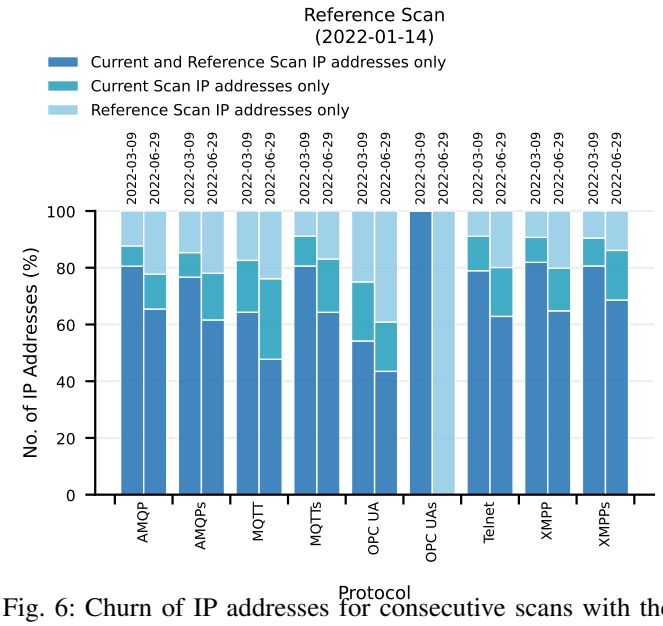

Fig. 6: Churn of IP addresses for consecutive scans with the first scan used as a reference.

proportion of successful handshakes.

Another notable observation is the substantial difference in the proportion of successful handshakes in secure and non-secure versions of each protocol. We notice a higher rate of successful handshake in non-secure versions compared to the secure versions. For example, in AMQP, we perform a successful protocol handshake for 48.3% of hosts with open ports, while in its secure version, the success rate is only 4.3%. Upon investigation, we track down these differences to TLS handshake fails, as in the majority of cases we do not reach the actual protocol handshake. The explanations for such behaviors can be that end-hosts may use Server Name Indication (SNI) [21] or expect their clients to present certificates to complete the TLS handshake [49].

Next, we analyze the address churn, i.e., the change in responsive IPv6 addresses between our measurements. In Figure 6, we present the percentage change in the number of responsive hosts relative to the first scan (reference scan). The bars represent subsequent scans and indicate the percentage of IP addresses that are observed in both scans, the reference scan only, and the subsequent scan only. Our observations show that for all protocols, the latest scan on June 29, 2022, has the smallest overlap with the reference scan. This is expected since the two scans are conducted almost six months apart. Nevertheless, there is still an overlap of at least 40% for most protocols. One exception is the secure OPC UA protocol, where we see no overlap in the latest scan, since the single responsive host in the first two scans did not respond anymore. Overall, when we consider all protocols, we observe an overlap of 62.5% between the first and last scan.

D. *Characterizing the Networks*

In this section, we investigate the characteristics of the networks where our responding hosts are located. In particular,

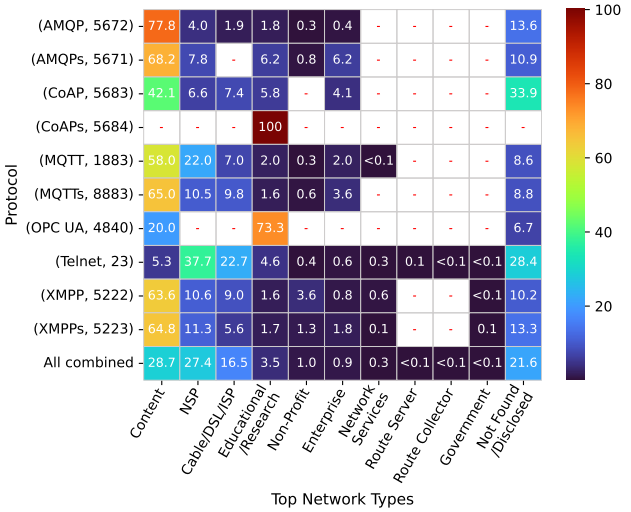

Fig. 7: Heatmap % IPs hosting our protocols in network types.

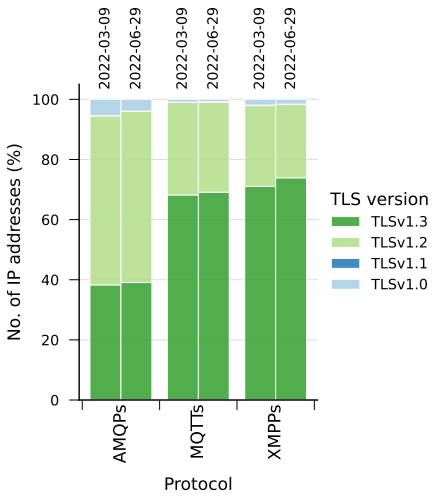

Fig. 8: Comparison of the distribution of highest TLS version supported by secured TCP-based protocols AMQPs, MQTTs and XMPPs for two scans conducted three months apart.

we are interested to identify the network types, such as whether they belong to content providers or eyeball networks. For this, we utilize the PeeringDB dataset [1], a dataset that helps network operators to make decisions regarding peering requests at Internet Exchange Points (IXPs) or interconnection facilities. This dataset provides information about Autonomous Systems (AS) networks, including their network types and traffic volumes. We begin by mapping our IP addresses to AS Numbers (ASN) using the routeviews [57] dataset. Then, we perform a lookup of the ASNs in the PeeringDB to identify the network types.

In Figure 7, we present the percentage of IoT hosts for each protocol-port across all network types, again based on our latest measurement (2022-06-29). Firstly, for most protocols, the majority of IoT hosts are found within Content networks. This suggests that the observed IoT hosts likely function as IoT backend servers that support the IoT devices themselves [49]. However, when examining Telnet hosts, we observe that they are predominantly located in Network Service Provider (NSP) and Internet Service Provider (ISP) categories, accounting for over ~60% in both cases combined. Although we only observed a handful of CoAPs and OPC UA responding hosts, they were unexpectedly in educational/research networks. Additionally, for roughly 22% of IoT hosts, no entry was found in PeeringDB (~17%) or the network type was intentionally undisclosed(4.8%). Overall, across all protocols (bottom row of the figure), the largest network category is Content providers ~29%, closely followed by NSPs ~27%.

V. Security Analysis

In this section we analyze different aspects of TLS security for the found IoT hosts: We investigate the supported TLS versions, certificate issuers, expiration of certificates, self-signed certificates, and non-trusted certificates. We provide comparison of security analysis only for the protocols AMQPs, MQTTs, and XMPPs and not for OPC UAs and CoAPs for which we do not discover more than one host. The only discovered OPC UAs speaking host supports TLS 1.2 and uses an expired certificate. The CoAPs host we discover supports DTLS 1.2.

A. TLS Versions

Since the first TLS standard was published in 1999 as TLS 1.0 [15], several updated versions of the protocol followed over the years [16], [17], with the latest TLS 1.3 version [46] being released in 2018. Newer TLS versions come with improved security characteristics; therefore, the maximum supported TLS version can be used to assess the security posture of a host [39]. Hence, we analyze the maximum advertised TLS version from IoT hosts for the TLS-enabled versions of the AMQPs, MQTTs, and XMPPs protocols.

In Figure 8, we show the maximum advertised TLS version for AMQPs, MQTTs, and XMPPs for two measurements (March and June 2022). We observe that for MQTTs and XMPPs, around 70% of IPs offer the most recent TLS version 1.3. In contrast, for AMQPs, we see a much lower prevalence of TLS 1.3 with just below 40%. The majority of the remaining share for all three protocols is allocated to TLS 1.2, with TLS 1.0 making up small single-digit percentages. Moreover, we see a slight increase in TLS 1.3 support across all protocols in the three months period.

These relatively high deployment numbers of TLS 1.3 seem to be in line with Web server support [30], [34], [53], but somewhat contrast the support of TLS 1.3 by IoT client devices, where we see a much lower prevalence of TLS 1.3 support [39].

B. Certificate Issuers

In total, we see 2239 hosts sending certificates for AMQPs, MQTTs, and XMPPs. As has been studied before [10], [24], the same certificate can be used on multiple hosts. We find that a single certificate is sent by 247 hosts. Overall, the majority of certificates are unique, resulting in 1647 unique certificates.

Issuer	Total	AMQPs	MQTTs	XMPPs
Let's Encrypt	61.1%	32.0%	77.4%	53.0%
Unknown	18.8%	17.2%	2.9%	30.0%
Sectigo Limited	5.3%	21.8%	2.1%	5.7%
DigiCert Inc	1.7%	2.3%	0.7%	2.3%
ZeroSSL	0.9%	0.9%	0.8%	0.8%

TABLE II: Top 5 certificate issuers with share for AMQPs, MQTTs, and XMPPs.

Protocol	Self-signed	Expired	Untrusted	Total
AMQPs	3.9%	25.0%	57.8%	128
MQTTs	1.8%	23.7%	39.7%	858
XMPPs	32.8%	13.0%	47.3%	1253

TABLE III: Percentage of self-signed, expired, untrusted, and total number of certificates per IoT protocol.

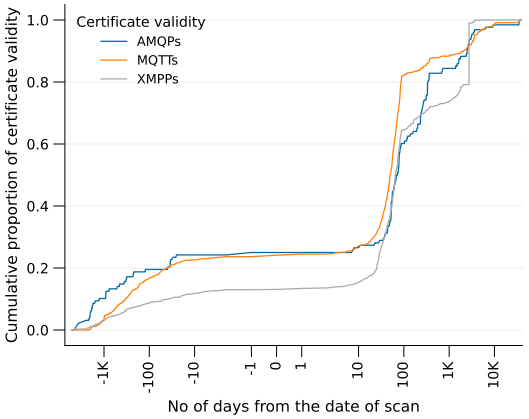

Fig. 9: ECDF of validity in terms of certificates expiry used by secured TCP-based Protocols AMQPs, MQTTs and XMPPs.

Next, we analyze the issuers of TLS certificates sent by IoT IPs. In Table II, we show the overall top 5 certificate issuers by the number of certificates. We can see that overall, Let's Encrypt is the dominant issuer for IoT TLS certificates, spanning from 32% to 77%, depending on the protocol. This is consistent with Let's Encrypt's dominance in the Web PKI [22], where it is even more pronounced.

Interestingly, we find a relatively large number of certificates with "Unknown" issuers for AMQPs and XMPPs, i.e., where the issuer organization field is empty. We investigate this artifact in-depth and find that for AMQPs, the majority seem to be certificates created by the message broker software RabbitMQ[1].

In XMPPs, on the other hand, almost all of these are self-signed snake oil certificates [35][2] sent by addresses owned by Google and hinting at the lack of SNI used of our scans in the certificate's issuer fields[3].

C. Certificate Characteristics

Finally, we investigate how many certificates belonging to IoT IPs are self-signed, expired, or untrusted, as shown in Table III. We find that the share of self-signed certificates for AMQP and MQTT is below 4%. Interestingly, almost one-third of all XMPP certificates are self-signed. We investigate this high number manually and find that they are again the self-signed snake oil certificates sent by addresses owned by Google.

Moreover, we analyze the expiry of certificates. Again, we find similar percentages for AMQP and MQTT (around 25%), whereas, for XMPP, the percentage is much lower at 13%. Manual investigation shows that this is due to a larger share of expired non-self-signed certificates for AMQP as well as MQTT (around 23%) compared to XMPP (8.8%). We find that the top issuers for these expired certificates are Sectigo and Let's Encrypt for AMQP and MQTT, respectively. We also analyze the number of days since a certificate has expired, as shown in Figure 9. We see that the majority of expired certificates have expired by more than 100 days, with a non-negligible share reaching more than three years (1k days). We also see that the majority of non-expired certificates are valid for less than 90 days, hinting at the duration of certificates issued by popular free CAs such as Let's Encrypt, which we also see for the Web PKI [22].

Next, using ZGrab2's runtime checks we analyze whether IoT TLS certificates are trusted by browsers. We find that AMQPs has the largest share of untrusted certificates with almost 60%, followed by XMPPs (47%) and MQTTs (40%). When looking at these untrusted certificates in detail, we find that for AMQPs there are three reasons for untrusted certificates: unknown issuers as discussed in the previous section (43.2%), signed by an untrusted CA (32.4%), and expired certificate (24.3%). For MQTTs the most common reasons for untrusted certificates are expiry (53.4%), unknown error (29.9%), and signed by an untrusted CA (16.1%). Finally, for XMPPs the by far largest share is due to the snakeoil certificate sent by Google-owned IPs (70.0%), followed by expired certificates (15.3%) and signed by an untrusted CA (14.5%).

VI. Related Work

In this section we present a brief overview of related work and compare their results to ours. Although there are numerous studies analyzing various aspects in the IoT ecosystem [29], [33], [41], [42], [45], [45], [47], [48], [49], most of them focus exclusively on the IPv4 Internet.

IoT deployment: In 2021, Srinivasa et al. [54] identified misconfigured IoT devices in the IPv4 Internet using active scan measurements. Their work primarily focused on scanning MQTT, AMQP, XMPP, CoAP, and Telnet on the standard unsecured ports. Our work extends the list of protocols to include OPC UA, a protocol that is extensively used in Industrial IoT [38]. Further, we scan the IPv6 Internet instead of

[1]Issuer: `CN=TLSGenSelfSignedtRootCA, L=$$$$`, see this GitHub issue [6].

[2]Snake Oil certificates are self-signed certificates that are typically installed by default along with some packages or the operating system itself.

[3]Issuer: `OU=No SNI provided\; please fix your client., CN=invalid2.invalid`

IPv4 and plan to release all data and analysis scripts. Comparing the results from our scan with the results published by Srinivasa et al., we find Telnet to be the most popular IoT-related protocol for both the IPv4 and IPv6 Internet. The second most popular protocol is XMPP for IPv6 as opposed to MQTT for IPv4. Considering the common protocols scanned on unsecured ports, the total number of IP addresses hosting IoT protocols for IPv4 is $13M$ compared to $34.2K$ for IPv6. However, it is worth to note that Srinivasa et al. considered port 2323 in addition to the standard port 23 for Telnet and the server port 5269 in addition to port 5222 for XMPP, which increases the chances of discovery for the Telnet and XMPP protocols.

In 2020, Dahlmanns et al. [14] conducted an IPv4-wide scan to discover devices responding to OPC UA probes. They discovered a maximum of 2069 OPC UA speaking IoT devices in a period of seven months from February to August 2020. In our measurements spanning six months (January to June 2022), we discover a maximum of 19 OPC UA speaking IoT devices in any particular scan. We find the chances of a successful application-layer handshake on an open OPC UA port to be higher for IPv6 (1.4%) as opposed to IPv4 (0.5%).

Vulnerability search engines: Shodan [51] and Censys [19] are search engines that provide the data collected from Internet-wide scans by periodically performing active scans on TCP and UDP based protocols. Historically, both search engines scanned only the IPv4 Internet. More recently, they have slowly started releasing scan results for the IPv6 Internet for a few ports as well. While Shodan does not publicly reveal their scanning pipeline, Censys initially used tools built on open source software such as the ZMap and ZGrab2 [7], [59] to scan the IPv4 Internet. Neither Censys nor Shodan released their latest scanning tools for scanning the IPv6 Internet to be used by the research community.

Comparing the results from our scan with the results published by Shodan, we observe that Shodan has not discovered any OPC UA speaking hosts in the IPv6 Internet. Furthermore, for all the six protocols we have considered, Shodan has not managed to discover any devices on secured ports. Moreover, the number of discovered IoT-speaking devices from our scans is 150 times higher ($34K$ vs. 225) than the data reported by Shodan, which indicates that Shodan has either an inefficient scanning methodology, it has been blocklisted by networks, or they do not use comprehensive target lists.

Comparing the results from our latest scan conducted on June 29, 2022, with the IPv4 data obtained from Censys for all the port-protocol combinations except for OPC UA [14], we find $100.6K$ devices in IPv6 as opposed to Censys's $4.4M$ in the IPv4 address space. Furthermore, the total number of IP addresses hosting popular IoT protocols on their standard ports for IPv4 is $2.6M$ compared to $36.4K$ for IPv6. Summing up these observations, around 59.7% of IP addresses with open ports in IPv4 result in a successful application-layer handshake compared to 36.2% in IPv6. Hence, the chances of discovering protocols on their standard ports are lower in IPv6 compared to IPv4 for this hitlist.

VII. Conclusion

In this paper we performed an in-depth analysis of the IPv6 IoT ecosystem using active measurements. We scanned 530M IPv6 addresses on six popular IoT-related protocols and found 36.4K IoT-speaking end-hosts in 156 countries. In comparison with IPv4, we identified 380× fewer IoT-speaking end-hosts. Our security analysis for TLS-enabled IoT protocols showed up to 57% untrusted certificates, with up to 32% being self-signed and 25% being expired. Finally, we plan to publish our results, tools, and web dashboard for further research.

References

[1] "PeeringDB," https://www.peeringdb.com.

[2] I. Analytics, "IoT 2019 in Review: The 10 Most Relevant IoT Developments of the Year," https://iot-analytics.com/iot-2019-in-review/, 2020.

[3] Anonymized author(s), "Dashboard: Discovering IoT-Speaking Hosts in IPv6 Internet," Anonymized URL, 2022.

[4] ——, "ZMap and ZGrab2 probe modules for DTLS, AMQP, CoAP, MQTT, and XMPP," Anonymized URL, 2022.

[5] L. Babun, K. Denney, Z. B. Celik, P. McDaniel, and A. S. Uluagac, "A survey on IoT platforms: Communication, security, and privacy perspectives," *Computer Networks*, vol. 192, p. 108040, 2021.

[6] L. Bakken, "Rabbitmq issue on github: Tls options for rabbit application supersede tls options for rabbitmq_management," https://github.com/rabbitmq/rabbitmq-management/issues/800, Apr. 2020.

[7] C. Bennett, A. Abdou, and P. C. van Oorschot, "Empirical Scanning Analysis of Censys and Shodan," 2021.

[8] P. N. Bideh, J. Sönnerup, and M. Hell, "Energy Consumption for Securing Lightweight IoT Protocols," in *Proceedings of the 10th International Conference on the Internet of Things*, ser. IoT '20. New York, NY, USA: Association for Computing Machinery, 2020.

[9] C. Bormann, S. Lemay, H. Tschofenig, K. Hartke, B. Silverajan, and B. Raymor, "CoAP (Constrained Application Protocol) over TCP, TLS, and WebSockets," RFC 8323, Feb. 2018. [Online]. Available: https://www.rfc-editor.org/info/rfc8323

[10] F. Cangialosi, T. Chung, D. Choffnes, D. Levin, B. M. Maggs, A. Mislove, and C. Wilson, "Measurement and Analysis of Private Key Sharing in the HTTPS Ecosystem," in *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, 2016, pp. 628–640.

[11] Censys, "Research Access to Censys Data," https://support.censys.io/hc/en-us/articles/360038761891-Research-Access-to-Censys-Data, Feb 2022.

[12] Chair of Network Architectures and Services at TUM, "Zmapv6: Internet scanner with ipv6 capabilities," https://github.com/tumi8/zmap, Jul. 2022, gitHub repository, Accessed: 26-October-2022.

[13] J. Czyz, M. Allman, J. Zhang, S. Iekel-Johnson, E. Osterweil, and M. Bailey, "Measuring ipv6 adoption," *SIGCOMM Comput. Commun. Rev.*, vol. 44, no. 4, p. 87–98, aug 2014.

[14] M. Dahlmanns, J. Lohmöller, I. B. Fink, J. Pennekamp, K. Wehrle, and M. Henze, "Easing the Conscience with OPC UA: An Internet-Wide Study on Insecure Deployments," in *Proceedings of the ACM Internet Measurement Conference*, ser. IMC '20. New York, NY, USA: Association for Computing Machinery, 2020, p. 101–110.

[15] T. Dierks and C. Allen, "The TLS Protocol Version 1.0," RFC 2246 (Historic), RFC Editor, Fremont, CA, USA, Jan. 1999, obsoleted by RFC 4346, updated by RFCs 3546, 5746, 6176, 7465, 7507, 7919. [Online]. Available: https://www.rfc-editor.org/rfc/rfc2246.txt

[16] T. Dierks and E. Rescorla, "The Transport Layer Security (TLS) Protocol Version 1.1," RFC 4346 (Historic), RFC Editor, Fremont, CA, USA, Apr. 2006, obsoleted by RFC 5246, updated by RFCs 4366, 4680, 4681, 5746, 6176, 7465, 7507, 7919. [Online]. Available: https://www.rfc-editor.org/rfc/rfc4346.txt

[17] ——, "The Transport Layer Security (TLS) Protocol Version 1.2," RFC 5246 (Proposed Standard), RFC Editor, Fremont, CA, USA, Aug. 2008, obsoleted by RFC 8446, updated by RFCs 5746, 5878, 6176, 7465, 7507, 7568, 7627, 7685, 7905, 7919, 8447, 9155. [Online]. Available: https://www.rfc-editor.org/rfc/rfc5246.txt

[18] D. Dittrich and E. Kenneally, "The Menlo Report: Ethical Principles Guiding Information and Communication Technology Research," *SSRN Electronic Journal*, 08 2012.

[19] Z. Durumeric, D. Adrian, A. Mirian, M. Bailey, and J. A. Halderman, "A Search Engine Backed by Internet-Wide Scanning," in *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*, ser. CCS '15. New York, NY, USA: Association for Computing Machinery, 2015, p. 542–553.

[20] Z. Durumeric, E. Wustrow, and J. A. Halderman, "ZMap: Fast Internet-wide Scanning and Its Security Applications," in *22nd USENIX Security Symposium (USENIX Security 13)*. Washington, D.C.: USENIX Association, Aug. 2013, pp. 605–620.

[21] D. Eastlake 3rd, "Transport Layer Security (TLS) Extensions: Extension Definitions," RFC 6066 (Proposed Standard), RFC Editor, Fremont, CA, USA, Jan. 2011, updated by RFCs 8446, 8449. [Online]. Available: https://www.rfc-editor.org/rfc/rfc6066.txt

[22] S. M. Farhan and T. Chung, "Exploring the evolution of the TLS certificate ecosystem," in *Proceedings of the 2023 Passive and Active Measurement Conference*, 2023.

[23] S. Figueroa-Lorenzo, J. Añorga, and S. Arrizabalaga, "A Survey of IIoT Protocols: A Measure of Vulnerability Risk Analysis Based on CVSS," *ACM Comput. Surv.*, vol. 53, no. 2, apr 2020.

[24] O. Gasser, B. Hof, M. Helm, M. Korczynski, R. Holz, and G. Carle, "In Log We Trust: Revealing Poor Security Practices with Certificate Transparency Logs and Internet Measurements," in *Passive and Active Measurement Conference 2018*, Mar. 2018.

[25] O. Gasser, Q. Scheitle, P. Foremski, Q. Lone, M. Korczynski, S. D. Strowes, L. Hendriks, and G. Carle, "Clusters in the expanse: Understanding and unbiasing ipv6 hitlists," in *Proceedings of the 2018 Internet Measurement Conference*. New York, NY, USA: ACM, 2018.

[26] O. Gasser, Q. Scheitle, S. Gebhard, and G. Carle, "Scanning the IPv6 Internet: Towards a Comprehensive Hitlist," in *Proc. of 8th Int. Workshop on Traffic Monitoring and Analysis*, Louvain-la-Neuve, Belgium, Apr. 2016.

[27] A. Giri, S. Dutta, S. Neogy, K. Dahal, and Z. Pervez, "Internet of Things (IoT): A Survey on Architecture, Enabling Technologies, Applications and Challenges," in *Proceedings of the 1st International Conference on Internet of Things and Machine Learning*, ser. IML '17. New York, NY, USA: Association for Computing Machinery, 2017.

[28] Google, "Google IPv6 Statistics," https://www.google.com/intl/en/ipv6/statistics.html, 2022.

[29] H. Guo and J. Heidemann, "Detecting iot devices in the internet," *IEEE/ACM Transactions on Networking*, vol. 28, no. 5, pp. 2323–2336, 2020.

[30] R. Holz, J. Hiller, J. Amann, A. Razaghpanah, T. Jost, N. Vallina-Rodriguez, and O. Hohlfeld, "Tracking the deployment of TLS 1.3 on the Web: A story of experimentation and centralization," *ACM SIGCOMM Computer Communication Review*, vol. 50, no. 3, pp. 3–15, 2020.

[31] L. Izhikevich, R. Teixeira, and Z. Durumeric, "LZR: Identifying Unexpected Internet Services," in *USENIX Security Symposium*, 2021.

[32] K. Landefeld, "Ipv6 - making the internet end-to-end addressable again," https://www.dotmagazine.online/issues/digital-identities/ipv6, [Online; accessed 25-March-2023].

[33] A. Lavrenovs and G. Visky, "Exploring features of http responses for the classification of devices on the internet," in *2019 27th Telecommunications Forum (TELFOR)*, 2019, pp. 1–4.

[34] H. Lee, D. Kim, and Y. Kwon, "TLS 1.3 in practice: How TLS 1.3 contributes to the internet," in *Proceedings of the Web Conference 2021*, 2021, pp. 70–79.

[35] A. Maghsoudlou, L. Vermeulen, I. Poese, and O. Gasser, "Characterizing the vpn ecosystem in the wild," in *Passive and Active Measurement: 24th International Conference, PAM 2023, Virtual Event, March 21–23, 2023, Proceedings*. Springer, 2023, pp. 18–45.

[36] A. Mangino, M. S. Pour, and E. Bou-Harb, "Internet-scale insecurity of consumer internet of things: An empirical measurements perspective," *ACM Trans. Manage. Inf. Syst.*, vol. 11, no. 4, oct 2020.

[37] L. Metongnon and R. Sadre, "Beyond Telnet: Prevalence of IoT Protocols in Telescope and Honeypot Measurements," in *Proceedings of the 2018 Workshop on Traffic Measurements for Cybersecurity*, ser. WTMC '18. New York, NY, USA: Association for Computing Machinery, 2018, pp. 21–26.

[38] Microsoft, "OPC UA," https://azure.github.io/Industrial-IoT/opcua.html, 2023.

[39] M. T. Paracha, D. J. Dubois, N. Vallina-Rodriguez, and D. Choffnes, "IoTLS: understanding TLS usage in consumer IoT devices," in *Proceedings of the 21st ACM Internet Measurement Conference*, 2021, pp. 165–178.

[40] C. Partridge and M. Allman, "Ethical Considerations in Network Measurement Papers," *Commun. ACM*, vol. 59, no. 10, p. 58–64, sep 2016.

[41] A. Pashamokhtari, N. Okui, Y. Miyake, M. Nakahara, and H. H. Gharakheili, "Inferring connected iot devices from ipfix records in residential isp networks," in *2021 IEEE 46th Conference on Local Computer Networks (LCN)*. IEEE, 2021, pp. 57–64.

[42] R. Perdisci, T. Papastergiou, O. Alrawi, and M. Antonakakis, "IoTFinder: Efficient Large-Scale Identification of IoT Devices via Passive DNS Traffic Analysis," in *IEEE European Symposium of Security and Privacy*, 2020.

[43] Pion, "DTLS Go Package," https://pkg.go.dev/github.com/pion/dtls/v2, accessed: 2022-7-27.

[44] M. Piraux, T. Barbette, N. Rybowski, L. Navarre, T. Alfroy, C. Pelsser, F. Michel, and O. Bonaventure, "The multiple roles that ipv6 addresses can play in today's internet," *SIGCOMM Comput. Commun. Rev.*, vol. 52, no. 3, p. 10–18, sep 2022.

[45] J. Ren, D. J. Dubois, D. Choffnes, A. M. Mandalari, R. Kolcun, and H. Haddadi, "Information Exposure From Consumer IoT Devices: A Multidimensional, Network-Informed Measurement Approach," in *ACM IMC*, 2019.

[46] E. Rescorla, "The Transport Layer Security (TLS) Protocol Version 1.3," RFC 8446 (Proposed Standard), RFC Editor, Fremont, CA, USA, Aug. 2018. [Online]. Available: https://www.rfc-editor.org/rfc/rfc8446.txt

[47] S. J. Saidi, O. Gasser, and G. Smaragdakis, "One bad apple can spoil your ipv6 privacy," *ACM SIGCOMM Computer Communication Review*, vol. 52, no. 2, p. 10–19, jun 2022.

[48] S. J. Saidi, A. M. Mandalari, R. Kolcun, H. Haddadi, D. J. Dubois, D. Choffnes, G. Smaragdakis, and A. Feldmann, "A haystack full of needles: Scalable detection of iot devices in the wild," in *ACM IMC*. New York, NY, USA: Association for Computing Machinery, 2020, pp. 87–100.

[49] S. J. Saidi, S. Matic, O. Gasser, G. Smaragdakis, and A. Feldmann, "Deep dive into the iot backend ecosystem," in *ACM IMC*, ser. IMC '22. New York, NY, USA: Association for Computing Machinery, 2022, p. 488–503.

[50] P. Saint-Andre, "Extensible Messaging and Presence Protocol (XMPP): Core," RFC 3920, Oct. 2004. [Online]. Available: https://www.rfc-editor.org/info/rfc3920

[51] Shodan, "Shodan dashboard," https://www.shodan.io/dashboard, accessed: 2023-03-20.

[52] J. Sidna, B. Amine, N. Abdallah, and H. El Alami, "Analysis and Evaluation of Communication Protocols for IoT Applications," in *Proceedings of the 13th International Conference on Intelligent Systems: Theories and Applications*, ser. SITA'20. New York, NY, USA: ACM, 2020.

[53] M. Sosnowski, J. Zirngibl, P. Sattler, and G. Carle, "DissecTLS: A Scalable Active Scanner for TLS Server Configurations, Capabilities, and TLS Fingerprinting," in *Proceedings of the 2023 Passive and Active Measurement Conference*, 2023.

[54] S. Srinivasa, J. M. Pedersen, and E. Vasilomanolakis, "Open for hire: Attack trends and misconfiguration pitfalls of iot devices," in *Proceedings of the 21st ACM Internet Measurement Conference*, ser. IMC '21. New York, NY, USA: Association for Computing Machinery, 2021, p. 195–215.

[55] The MaxMind Company, "Geolite2 free geolocation data," https://dev.maxmind.com/geoip/geolite2-free-geolocation-data, Oct. 2022, accessed: 06-October-2022.

[56] The ZMap Team, "Zgrab 2.0," https://github.com/zmap/zgrab2, Jul. 2022, gitHub repository, Accessed: 28-September-2022.

[57] University of Oregon, "Routeviews Project," http://www.routeviews.org/, 2022.

[58] J. Zirngibl, L. Steger, P. Sattler, O. Gasser, and G. Carle, "Rusty Clusters? Dusting an IPv6 Research Foundation," in *Proceedings of the 2022 Internet Measurement Conference*. New York, NY, USA: ACM, 2022.

[59] M. Zolotykh, "Study of Crawlers of Search Engine 'Shodan.io'," *2021 Ural Symposium on Biomedical Engineering, Radioelectronics and Information Technology (USBEREIT)*, pp. 0419–0422, 2021.