EV Charging Infrastructure Discovery to Contextualize Its Deployment Security

Khaled Sarieddine, Mohammad Ali Sayed, *Graduate Student Member, IEEE*, Chadi Assi, *Fellow, IEEE*, Ribal Atallah, Sadegh Torabi, *Member, IEEE*, Joseph Khoury, Morteza Safaei Pour, and Elias Bou-Harb, *Senior Member, IEEE*

***Abstract*—Electric Vehicle Charging Stations (EVCSs) have been shown to be susceptible to remote exploitation due to manufacturer-induced vulnerabilities, demonstrated by recent attacks on this ecosystem. What is more alarming is that compromising these high-wattage IoT systems can be leveraged to perform coordinated oscillatory load attacks against the power grid which could lead to the instability of this critical infrastructure. In this paper, we investigate a previously sidelined aspect of EVCS security. We analyze the deployment security of EVCSs and highlight operator-induced vulnerabilities rendering the ecosystem exposed to remote intrusions. We create an advanced discovery technique that leverages Web interface artifacts to dynamically discover new charging station vendors. As a result, we uncover 33,320 charging station management systems in the wild. Consequently, we study the deployment security of the charging stations and identify that 28,046 EVCSs were found to be vulnerable to eavesdropping, and around 24% of the studied EVCSs are deployed with default configurations exposing the ecosystem to a Mirai-like attack vector. Aligned with this finding, we discover that the EVCS ecosystem has been targeted by nefarious IoT malware such as Mirai and its variants. This demonstrates that further security measures should be implemented by vendors and operators to ensure the security of this vital ecosystem. Consequently, we provide a comprehensive recommendation for securing the deployment of EVCSs.**

***Index Terms*—Electric vehicle charging ecosystem, malware, fingerprint, forensics, power grid, darknet.**

Manuscript received 5 January 2023; revised 10 July 2023 and 12 September 2023; accepted 14 September 2023. Date of publication 4 October 2023; date of current version 7 February 2024. This research was conducted and funded as part of the Concordia University/Hydro-Quebec/NSERC research collaboration project titled: "Large-Scale Integration of EVCSs into the Smart Grid: A comprehensive cyber-physical study and security assessment." Grant reference: ALLRP 567144-21. The associate editor coordinating the review of this article and approving it for publication was J. J. Yang. *(Corresponding author: Chadi Assi.)*

Khaled Sarieddine, Mohammad Ali Sayed, and Chadi Assi are with the Security Research Centre, Concordia Institute for Information Systems Engineering, Concordia University, Montreal, QC H3G 1M8, Canada (e-mail: chadi.assi@concordia.ca).

Ribal Atallah is with the System Resilience, R&D Department, Hydro-Quebec Research Institute, Montreal, QC J3X 1S1, Canada.

Sadegh Torabi is with the Center for Secure Information Systems, George Mason University, Fairfax, VA 22035 USA.

Joseph Khoury and Elias Bou-Harb are with the Cyber Center for Security and Analytics, The University of Texas at San Antonio, San Antonio, TX 78249 USA.

Morteza Safaei Pour is with the Department of Management Information Systems, San Diego State University, San Diego, CA 92182 USA.

Digital Object Identifier 10.1109/TNSM.2023.3318406

I. Introduction

CLIMATE change, increased greenhouse gas emissions and the irreparable impact they could have on our way of life have forced governments to embrace a new green mindset focused on the environment. To reduce the emissions of the transportation sector, countries are shifting towards the adoption of electric vehicles (EVs). This adoption rate has been growing exponentially in the last few years [1]. In the first 5 months of 2022, 3.2 million new EVs were registered worldwide, which is more than the total EVs registered in 2020 [2]. Meanwhile, charging infrastructure is being deployed rapidly to match the increasing charging demand [3]. For instance, Canada invested more than $400 million to address the lack of charging and refueling stations [4]. This massive push towards EVs is causing a compelling change in the transportation sector and simultaneously the power grid that serves as the critical infrastructure supporting the energy needs of the EV ecosystem. Electric vehicle charging stations (EVCSs) are high-wattage Internet-Enabled devices that are connected and controlled by remote entities (e.g., customers, operators, or manufacturers). The remote capabilities instilled in the ecosystem are meant to improve user experience and provide operators and consumers with the ability to start/stop, pay for charging, view the status of charging, etc. There are two types of EVCSs, public and private. The public EVCSs are utilized for commercial purposes and require remote management as they exist in large numbers. The EVCS ecosystem provides a vital service for customers and business owners, especially with the emergence of EV fleets, which depend on the ecosystem to operate. Additionally, these EVs and EVCSs can provide ancillary services and support the power grid frequency control in times of need [5]. Thus, securing this system is of utmost importance due to its connection to critical infrastructure such as the power grid [6].

Recent events have demonstrated that EVCS ecosystem attacks are on the rise. A backdoor was exploited by malicious adversaries to impact the availability of charging stations in Russia [7]. Whereas, in England, EVCSs were rendered unavailable while displaying inappropriate images [8]. Moreover, not only charging stations are a victim of cyber-attacks, but in November 2021 vulnerabilities were found in the mobile application of a United Kingdom domestic car charging provider that revealed the full names, addresses, and charging history of consumers, impacting the confidentiality

of the system and its integrity [9]. Moreover, a malicious swarm of EVCSs could be used to induce a disturbance on the power grid and create instability and possibly blackouts [3], [10], [11]. The adversary could command the swarm to periodically switch on and off synchronously to alter the grid behavior by impacting the generators' speed. Different attacks could be launched accordingly, such as switching attacks [3], and dynamic attacks [12]. While the current numbers of deployed EVCSs are not high enough to create a detrimental impact on the power grid, these numbers are expected to keep increasing for the coming years; thus increasing the risk of attacks initiated from this ecosystem against the grid.

The original manufacturer produces EVCSs, which are bought by charging station operators. The operator is responsible for managing, controlling, and updating the charging station firmware and beyond the point of sale, the manufacturer has no control. This highlights the operator's liability in securing the EVCS ecosystem. Consequently, it is imperative to study the security posture of the ecosystem and take into consideration deployment security. In this work, we aim at assessing the current security measures implemented by operators in securing the EVCS ecosystem by studying the deployment strategies of EVCSs worldwide and examining prominent EVCS vendors and the various tools they provide to manage their charging stations.

To secure the EVCS ecosystem and the power grid especially, discovering, cataloging, and annotating the EVCS hosts is of utmost importance. Consequently, we also develop an approach to extend our knowledge of the EVCS ecosystem by identifying charging station management systems and creating an advanced discovery mechanism. We are among the first to assess the EVCS ecosystem's susceptibility to remote attacks due to the lack of proper security measures adopted by charging station operators rather than manufacturers. We further our study by performing an in-depth analysis of the malware threat landscape impacting the EVCS ecosystem. Recent reports indicate that malware might be used to stop or slow down the charging stations [13]. The adversary behind malware attacks against the EVCSs can have several objectives one of which is distributed denial of service (DDoS) to prevent users from charging and holding it at ransom [13] which would impact the availability of the charging infrastructure. The presence of malware on EVCSs compromises the whole ecosystem and provides new attack vectors that could impact the power grid. The current landscape shows that the ecosystem is vulnerable; however, little to no research has been done to understand the current malware threat landscape (e.g., if the ecosystem is infected with malware). Of the 33,320 EVCSs we discovered in the wild, 84.17% are vulnerable to Man-in-the-Middle (MitM) attacks showing a prevalence in the lack of proper deployment security accounting for 95% of the discovered vendors. This is attributed to the lack of secure communication protocols like HTTPS.

Moreover, we discovered that 15.7% of the studied EVCSs are deployed with default configuration exposing the system to Mirai-like malware. Consequently, we scan the darknet network telescope and discover that the EVCS ecosystem is being infected by malware which requires further efforts in secure deployment security. To this end, we summarize our contributions as follows:

- This work addresses the challenge of creating a scalable EVCS discovery mechanism that leverages artifacts extracted from WebUIs. We utilize unique features to correlate the extracted WebUIs to the EVCS ecosystem to discover 33,320 charging stations in the wild belonging to 22 vendors. It is indeed imperative to discover EVCSs due to the sensitive service they provide. We devise advanced fingerprinting techniques by utilizing Google dorks and leverage translation of Web banners to increase the number of identified hosts. We bootstrap device search engines and create an advanced discovery mechanism. This work addresses the limitations of previous EVCS discovery mechanisms [14], [15] and discovered 17 new vendors that were not discovered before and expanding our knowledge of the EVCS ecosystem.
- We subsequently assess their deployment security and have discovered remote exploits. We identified that about 84% of the discovered charging stations are vulnerable to MitM attacks due to the lack of a secure communication protocol like HTTPS. This is exposing the ecosystem to a multitude of intrusions that might impact the ecosystem based on the CIA triad. Moreover, we show that 15.7% of the discovered charging stations are operating with insecure configurations exposing the ecosystem to a Mirai-like malware. We then scan the darknet, i.e., network telescope and discover that malware is indeed infecting the EVCS ecosystem. To the best of our knowledge, we are the first to verify the existence of EVCSs infected with malware showing the imminent threat facing this ecosystem. Using our advanced fingerprinting mechanism we were able to increase the number of discovered EVCSs on the darknet by 339% as compared to scans based on EVCSs discovered by previous methodologies [14], [15] by discovering hard-to-discover vendors.
- We provide a comprehensive recommendation to secure the EVCS ecosystem from discovery and hinder adversaries from targeting the EVCS ecosystem which would require the collaboration of manufacturers and operators.

The remainder of this paper is organized as follows. In Section II, we present background information and basic concepts related to the EV ecosystem. In Section III we present related work that focuses on IoT security and discovery. In Section IV, we discuss the methodology and details of our discovery mechanism and security analysis. Finally, we provide concluding remarks in Section VI.

II. Background

The EV charging ecosystem is a heterogeneous system composed of cyber and physical components that interact to provide vital charging services to customers and businesses. In what follows, we provide details about the different components of the EVCS ecosystem.

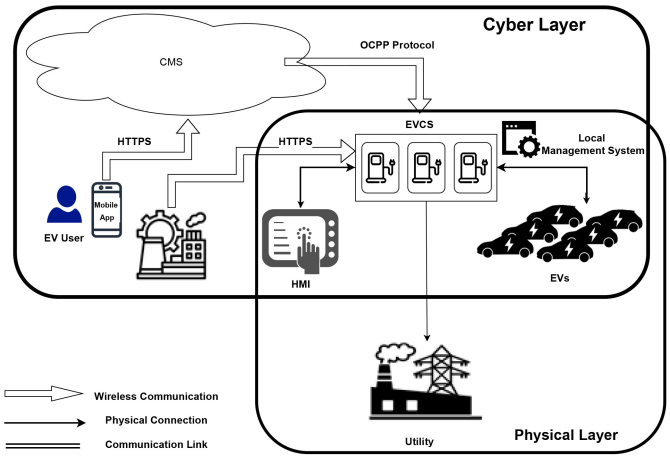

Fig. 1. Overview of the EV ecosystem.

A. Overview of the EV Ecosystem Components

The EV ecosystem is evolving to match the demand for EVs and their charging requirements. It is composed of multiple interconnected components that are utilized to ensure the availability of charging services to EV owners. The plug-in hybrid electric vehicles and the battery electric vehicles are the two main types of vehicles that are dependent on the ecosystem [3]. Once the EV is connected to the EVCS, it communicates to the EVCS its information such as the MAC address, status of charge, etc. This communication between the EV and the EVCS is governed by multiple standards such as ISO 15118 and IEC 65180.

The public EV ecosystem is composed of multiple cyber-layer components (mobile application, EVCS firmware, communication protocols, etc.) as shown in Figure 1, that leverage the cloud management system as the liaison to monitor and manage the interaction of these components. The management system provides the mobile application and its users with remote capabilities. The mobile application sends http/s requests that are interpreted into Open Charge Point Protocol (OCPP) [16] requests and forwarded to the EVCS by the cloud management system. It is worth noting that the OCPP provides a wide range of functionalities that simplifies the management of EVCSs such as start, stop, firmware update, etc. The management system provides supervisory control and data acquisition system (SCADA) to gather data in real-time from remote EVCS locations to control equipment and conditions for commercialization and remote management. It is worth noting that EVCSs are equipped with firmware that is used to interface the cyber and physical components of the charging station. The firmware hosts a Web server that provides a Web interface to manage the individual charging stations which is called a local management system. Moreover, there are EVCS cloud management systems [14], [15] that are used to remotely manage charging stations.

The EVCSs are high-wattage IoT devices that are connected to critical infrastructure (i.e., the power grid). EVCSs can either be AC or DC and are classified based on their charging rate. Level 2 chargers are the most common public EVCSs while Level 3 DC chargers are also being deployed to provide charging rates of 40 kW to 360 kW, decreasing charging times and enhancing the user experience.

B. Security of the EVCS Ecosystem (Attacks and Implications)

Compromising the EVCS ecosystem has detrimental impacts on multiple stakeholders including the connected critical infrastructure and the customers. It has been proven to be vulnerable to intrusion. The work in [17] shows that the OCPP protocol [16] is susceptible to man-in-the-middle attacks. The adversary can leverage OCPP to eavesdrop, impersonate, and alter charging requests. Consequently, providing the adversary with a new attack vector to steal user information (e.g., financial), impersonate, and cause denial of service. Moreover, in [14], [15] the authors discover multiple firmware vulnerabilities that allow remote adversaries to control EVCSs such as SQL injection and XSS. Such vulnerabilities are vendor-induced due to the lack of proper secure development of the firmware. Moreover, in [18], assessed the system design of the complex charging station infrastructure and identified that weak end-to-end authentication between the user and his vehicle could be exploited by adversaries to create a Denial of Service, hijacking charging/discharging sessions. Aside from academia, Kaspersky [19] discovered multiple vulnerabilities in ChargePoint Home chargers such as OS command injection, arbitrary file read, stack buffer overflow, etc.

While the aforementioned vulnerabilities impact the user, they can be leveraged to impact the connected critical infrastructure. In [3], the authors highlight a new class of attacks that could be launched by the EVCS ecosystem which could impact the grid stability. EVCS switching attacks utilize a swarm of EVCS botnets that are commanded to be turned on and off synchronously for a certain period. Such attacks impact the generator speeds and the grid frequency. The continuous exposure to switching attacks leads to load shedding and possible blackouts. Moreover, other types of mass charging attacks [20] also exist and are harder to detect as the adversary utilizes a single charging request over a swarm of EVCSs to increase the load on the ecosystem during peak hours, causing transmission losses and possibly line overloading and tripping. Different variations of these attacks could be mounted by exploiting the EVCS ecosystem. Moreover, the authors in [18], demonstrate the possible impacts of oscillatory load attacks initiated by the EVCS ecosystem leveraging the weak interactions between the components on the critical infrastructure leading to monetary losses for the utility and grid instability leading to power line cuts.

III. Related Works

In this section, we survey and discuss previous work that tackled IoT and cyber-physical system device discovery mechanisms and provide a detailed security assessment of cyber-physical systems.

Different commercial search engines exist that are used to discover, catalog, and annotate Internet-connected devices by scanning the entire IP address space. For example, Shodan [21] and Censys [22] are two commercial services

that are used to discover devices. These device search engines gather information about all devices directly connected to the Internet. Search engines query devices for various publicly available information. The bulk of the data is taken from banners, which are metadata about software that's running on a device. While these search engines provide access to structured data, they still lack the ability to label the devices due to the wide variety of IoT devices that are connected [14], [15].

Nasr et al. [14], [15], created an EVCS management system discovery mechanism that leverages passive scanning device search engines. Their approach mainly identifies charging stations that possess EVCS-related keywords and login forms in their Web interface/banners. They were able to discover 44 EVCS charging vendors accumulating to 27,439 EVCS hosts, where the majority of the discovered hosts are cloud management systems. The authors utilized Shodan, Censys, and Zoomeye, however, the authors note that they were able to discover more than 90% of the EVCS hosts using Zoomeye whereas the others were only able to discover around 5000 hosts only. Moreover, it is worth highlighting that the authors disregarded the presence of EVCS hosts that do not embed EVCS keywords or do not provide a login form thus, limiting their discovery technique. Some charging station vendors do not provide a login form as the Web interface is only used to display the status of the charging station and might provide different services to manage the charging station remotely such as SSH. Moreover, the authors did not take into consideration the need for translation to identify EVCSs in the wild and expand the knowledge of the ecosystem. EVCS fingerprinting is essential as it can provide utilities and attackers with a comprehensive view of the ecosystem. Finally, the authors utilized penetration testing techniques to identify vulnerabilities induced by the manufacturer/vendor such as SQL injection, XSS, etc.

In [23], the authors created an Acquisitional Rule-based Engine (ARE) for discovering IoT devices in the wild. ARE is an engine that creates association rules used to identify the discovered generic IoT devices (routers, IP cameras, etc.), that leverages the Apriori algorithm to dynamically identify IoT devices. They extract product names that follow the observation that a general IoT device product name is a combination of letters and numbers (perhaps containing "-"). Moreover, they utilize device entity recognition that requires access to a predefined list of vendors and product names. ARE engine generates rules that are used to identify IoT devices in a fine-grained manner as compared to other existing tools. However, due to the lack of standardization in the EVCS ecosystem, such a mechanism fails to identify EVCSs as they do not follow a standardized naming convention and hence a comprehensive list of vendors and their respective products does not exist. Moreover, in [24] the authors fingerprint industrial control system management devices by actively scanning mobile communication networks in Japan and the United States of America and manually inspecting Web pages. They were able to discover 21 device models accumulating to 890 hosts. They further their study by performing penetration testing techniques on 3 device models and identified 13 0-day vulnerabilities. Moreover, they developed and deployed honeypots that imitate remote ICS devices and monitored attackers' behavior to study the imminent threat that these devices are facing. However, their work only focused on attacker behavior disregarding the malware threat landscape. Similarly, in [25], the authors work on discovering Internet-connected vehicles while developing an approach that is similar to the approach proposed in [14], [15], and discovered 733 hosts belonging to 12 vendors and then further studied the usage of vulnerable service and identified that 91.6% of the vendors are running vulnerable services rendering the Internet-connected vehicles exposed to cyber-attacks. Moreover, Costin et al. [26] utilized supervised machine learning to classify firmware images and correlate them to the WebUI interface. Whereas, Wang et al. [27] proposed an engine for identifying IoT devices by utilizing the similarity between the response data of different IoT devices of the same vendor or product based on the structure and style of the response data. Additionally, Yu et al. [28] proposed a firmware identification method by analyzing Web page content. In contrast to other device types, EVCS has limited and non-trivial banners where most EVCMS products are closed-sourced, in addition to the lack of banner rules for identifying them [15]. Furthermore, EVCMS's lack of standardization among developers and vendors resulting makes it unfeasible to use existing approaches to fingerprint EVCSs [15]. In our work, we focus on discovering local-charging station management systems that are hosted on high-wattage IoT devices (EVCSs). Consequently, we propose the usage of Google Dorks, translation, and selecting mobile communication networks rather than using predefined keyword searches to help expand our knowledge of the EVCS ecosystem by discovering new devices dynamically and assisting in the manual inspection of Web pages. Moreover, we evaluate different security policies and issues that are put to ensure deployment security and discovered a lack of proper security in 84% of the hosts. Furthermore, we also analyze the vulnerable service running on the discovered hosts along with studying the malware threat landscape in the EVCS ecosystem that is proving itself to be an imminent threat.

IV. Methodology

To understand the current threat landscape facing the EVCS ecosystem due to deployment (in)security, we describe our overall methodology for device discovery in Figure 2. We also illustrate our deployment security analysis, which is among the first attempts in the EVCS ecosystem. First, we analyze the different deployment strategies and create a discovery mechanism that aids in identifying new EVCSs with an accessible Web interface with the aim to create a robust mechanism and increase the number of discovered hosts that do not necessarily embed EVCS-related keywords. Charging station vendors might create EVCS Web interfaces that do not include any of the keywords that were utilized in [14], [15] as a means to create their initial discovery seed (e.g., charging station, EVCS, OCPP, etc.), but rather only include vendor or product names that require domain knowledge. Consequently, we assess the security of these EVCSs in the wild by studying their deployment security namely focusing

TABLE I
LITERATURE SYSTEMATIZATION OF KNOWLEDGE AND COMPARISON TO PREVIOUS WORK

Reference	Type of Device	Limited Search Space	Banner-Based Search	Rule-Based Search	Feature-Based Search	Keyword Specific Search	Translation module	Security Flaws & Exposed Services	Vulnerability Assessment	Malware Investigation
Our Work	**EVCS**		✓	✓	✓		✓	✓		✓
Nasr et. al. [14]	EVCS	✓	✓		✓	✓			✓	
Nasr et al. [15]	EVCS	✓	✓		✓	✓			✓	
Ueda et al. [25]	EVs		✓		✓			✓		
Sasaki et. al. [24]	ICS		✓		✓			✓	✓	
Feng et al. [23]	IoT		✓	✓						
Costin et al. [26]	IoT		✓		✓					
Wang et al. [27]	IoT		✓		✓					
Yu et al. [28]	IoT		✓		✓					

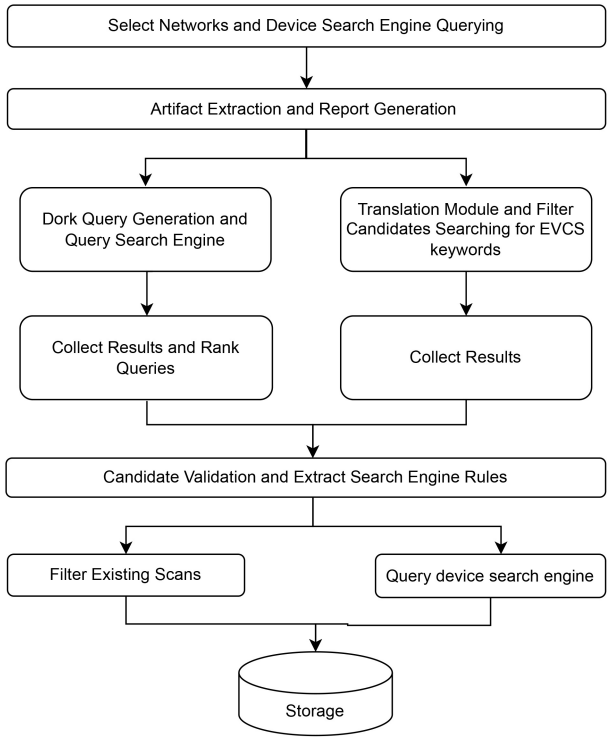

Fig. 2. Overall Advanced Discovery Methodology.

on OWASP-Top 10 deployment security-related risks such as security misconfiguration, vulnerable and outdated services, etc. Finally, we provide comprehensive recommendations on how to secure the deployment of the EVCSs which also requires considerable effort from the EVCS manufacturers as well.

A. Device Discovery

EVCS management systems do not expose unique services that allow their identification unless configured incorrectly. Search engines which utilize Internet-wide scans and other protocols such as Modbus do not allow us to discover or uniquely identify EVCSs because these services are not restricted to EVCSs and are not used by all of them. On the other hand, some EVCSs do have a Web user interface that could be used to identify them as part of the EV ecosystem or belonging to an EVCS vendor. The challenge arises in distinguishing these devices among the massive number of hosts with Web interfaces, noting that in some cases these EVCSs do not have any EVCS-related keywords, especially since the lack of standardization in the ecosystem provides a considerable challenge in identifying these devices.

Our fingerprinting technique is visualized in Figure 2. We leverage the observation that device manufacturers embed keywords in their websites that might indicate the manufacturer/vendor and give an indication about the device. However, another challenge exists since there is no consolidated list of manufacturers and their Web interfaces that allow us to easily search for EVCS hosts. In this work, we aim at addressing the limitations of [14], [15], by not limiting the search to a subset of hosts that possess EVCS keywords. Consequently, we select networks similar to [24].

1) Network Selection and Device Search Engine Querying: While Internet-wide scans would identify an overwhelming number of WebUIs, we start our process by selecting specific networks where the presence of EVCS is more probable. Similar to [24], which aimed at identifying remote management systems of industrial control system devices, we expect a higher concentration of such hosts in mobile data communication networks which were part of the seed used to identify hosts. Consequently, we select ISPs as a seed for our approach thus, not limiting ourselves to a predefined seed related to EVCS keywords. We collect the WebUIs present in selected networks in Finland, France, Italy, Germany, the United States, and Canada (e.g., Vodaphone Italia). We selected networks in these countries as it has been shown in [14], [15] to have a high concentration of EVCS hosts. We were able to discover new hosts in the same area showing the advantage of our approach. The IP address range of the ISPs is obtained from publicly available AS numbers and IP address assignment information. Consequently, we leverage device search engines that regularly scan the Internet and gather information about these networks. Namely, we utilize Zoomeye [29], as it showed the best performance compared to the other device search engines.

2) Artifact Extraction and Report Generation: Scans will provide us with EVCS banners that exist in a certain network. Consequently, we leverage the fact that EVCSs embed keywords in their WebUIs that could be used to uniquely identify them. It is worth noting that the EVCSs will share highly similar WebUIs, whereas regular websites will have a higher

entropy due to the heterogeneity of the information they contain [24]. Moreover, other IoT devices, digital video recorders, and routers will also share similar WebUIs within each family of devices. We define $\mathcal{W}$ as the candidate WebUI and $\mathcal{K}$ as a set of fields that need to be extracted from $\mathcal{W}$. Namely, we create a report for each $\mathcal{W}$ that contains $\forall k_i \in \mathcal{K}_{\mathcal{W}}$. $\mathcal{K}$ includes the title of the tab, title of the page, headers (h1-h4), file names, paragraph fields, footer, images source link, links href, and URL links in the embedded Javascript. Consequently, each report will include a list of keywords. We further filter our candidates by rigorously filtering based on generic IoT device keywords. Some types of IoT devices, such as IP cameras, might embed keywords in their WebUI that identifies them uniquely and gives us an indication that these are not EVCSs which allows us to filter out candidates. Moreover, we further filter the reports by removing time, date, and header information along with generic stop words using the NLTK [30] python package. NLTK is a natural language toolkit that is used to work in computational linguistic to tokenize and tag text, identify named entities, and remove stop words. These generated reports provide us with a defined list of keywords that are used in our google dork tool. Candidates that do not contain unique words are then discarded as general IoT devices.

3) Dork Query Generation, and Search Engine: We leverage the generated reports to identify unique keywords found in WebUI. To distinguish EVCS local management systems, we leverage the fact the vendors will embed data that would identify the product/vendor in the HTML code. Product names in the EVCS ecosystem do not conform to the naming convention of IoT devices thus, increasing the complexity of identifying EVCSs and rendering the methodology proposed in [23] limited to generic IoT devices. After generating a set of reports $\mathcal{R}$ for the Web interfaces, we identify the relevance of that document to the EVCS ecosystem by using Google Dorks. When Google crawls the Web to index pages for its search engines, it retrieves terabytes of data. However, whenever a user searches for something on Google, millions of records are retrieved, and following their proprietary ranking algorithm it will show thousands of search results. Consequently, the user will need to go through each and every document to identify how relevant it is to their search goal. Thus, we utilize Google Dorks which is a technique used to help limit the number of retrieved results by directing the search engine to search for these keywords in certain websites or by curating a query that has certain criteria. Instead of searching for the keywords on Google and checking their relevance manually, we use an advanced searching technique that allows us to dynamically find EVCSs. This advanced search technique allows us to find information not readily available on websites. Google Dorking can return information difficult to locate. We utilize two main websites, Chargemap [31] and Plugshare [32], that are continuously updated as new vendors join. They provide a platform for locating EVCSs by the users and also might include news about the EVCS ecosystem. Such platforms continuously reflect the newest charging networks that are joining and provide a comprehensive corpus for the EVCS ecosystem. We curate queries such that we direct our search to specific

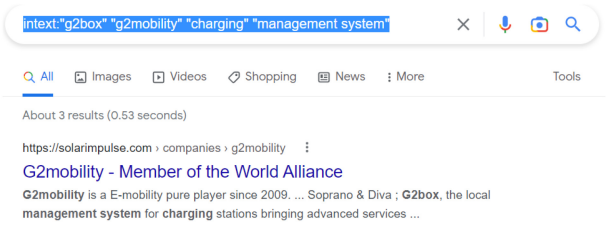

Fig. 3. Google dork snippet.

websites that are related to the EVCSs. Additionally, we also curate queries where we search for keywords extracted from the HTML banners along with two keywords "charging" and "management system" which retrieve results that contain the keywords along with "charging management system". These queries give us very high confidence that the retrieved pages are related to the EVCS ecosystem. Instead of manual search for information on Google and trying to create relevancy between the keywords and the retrieved results, we utilize the Dorking technique to identify information in unstructured data such as Plugshare and Chargemap.

Formally, we can define our query generation using Equation (1) defined below:

$$\begin{aligned} &\text{Let } \mathtt{K} \text{ be the set of all combinations of keywords in Report } R_i \\ &\text{Let } \mathbb{K} \text{ be the number of keywords in Report } R_i \\ &\text{Let } k \text{ be the number of keywords chosen as input to the Query} \\ &\text{Select c} = \binom{\mathbb{K}}{k} \text{ where } k \in [1, \mathbb{K}] \end{aligned} \tag{1}$$

We utilize the keyword combinations with our queries and retrieve the results. Three different query templates were used as shown below

$$\begin{aligned} \text{query}_1 &= \text{site: “chargemap.com” intext: “keyword}_\text{c}\text{”} \\ \text{query}_2 &= \text{site: “plugshare.com” intext: “keyword}_\text{c}\text{”} \\ \text{query}_3 &= \text{intext: “keyword}_\text{c}\text{” “charging management system”} \end{aligned}$$

We show in Figure 3 a sample query_1. As for the others, we follow a similar mechanism. For example, an example of the query would be intext: "SENEC" "charging" + "management system", where we ensure that the used keywords are related to the EVCS ecosystem by leveraging the search algorithm that is provided by Google. The results of the queries help us create a correlation between the keywords discovered and the EVCS ecosystem. Programatically, our search queries can be formatted as "search engine/search?q=site:"chargemap.com"+intext: "g2mobility"+&btnG=Search", where the mark (?) indicates the end of the URL, and the (&) separates arguments, q is the start of the query, the plus mark (+) represents space, and btnG=Search denotes that the search button is pressed on the Web interface [23].

Consequently, after the results are collected, each query can then be ranked based on its relevance using Equation (2):

$$ChargeScore(q) = \sum \mathtt{tf}_{EVCS_k} \sum \mathtt{tf}_{t,d}\mathtt{idf}_t \quad (2)$$

where tf is term frequency, idf is inverse document frequency, q is the query, t is the term in the query, and d is the results of each query that will get a score and sorted by decreasing ChargeScore. Namely, ChargeScore is the tf-idf weighted by the EVCS keywords term frequency. The charge score takes into account if EVCS ecosystem keywords are found in the search results denoted by EVCS_k, showing that it has greater relevance to the EVCS ecosystem. We can then calculate the repetition of query words in the document (tf), thus showing that query keywords are present in our search results. Finally, the relative rarity of a term in the collection of results per query is calculated. This is denoted by the IDF showing the unevenness of term distribution in the corpus. This measures the informativeness of the terms, which will be very low for queries with general terms. The usage of Google-Dorking techniques alongside the ChargeScore allows us to identify accurately which queries are the most relevant to the EVCS ecosystem. Thus, showing that the studied banner of a specific host is actually an EVCS which we later validate. The higher the ChargeScore is, the higher our confidence that these query results might actually be for an EVCS vendor.

4) Translation Module and Filtering Results: In this work, we shed light on the importance of using translation to discover new EVCSs. EVCS vendors might customize WebUIs and keywords based on the country of deployment. Thus, utilizing keywords of one language to search for EVCSs will hinder the discovery of EVCS candidates. Consequently, we translate EVCS-related keywords to different languages, mainly, Italian, French, German, and Spanish (e.g., Système de gestion des bornes de recharge, Management system für Ladestationen). We filter the WebUIs collected using this list of keywords we generated which allowed us to identify EVCSs that possess EVCS-related keywords in English as well as different languages.

5) Validation and Search Engine Queries Generation: Consequently, we validate the candidates by calculating the body hash of the banners to cluster them. This led to the discovery of 28 main banner groups that we manually explore and leverage to create search engine rules. The search engine rules are utilized to scale up our discovery mechanism by leveraging a combination of artifacts that we extract from each report $\mathcal{R}$ that would uniquely identify the candidate such as the title, file names, footer information, HTML attribute, etc. and using them as a search query on Zoomeye [29] device search engine. It is worth mentioning that the queries generated out of the previously mentioned artifacts extracted provide a unique signature that allows us to uniquely identify similar devices with similar banners. Then we utilized the hash of the banners to further validate the similarity. Finally, we filter our previous scans and query device search engines and store the results for future analysis. To scale up the detection of devices using the hosts we identified, we leverage devices search engines to increase our results by utilizing the keywords we determined as EVCS management system. We continuously followed the same approach and identified 28 device signatures accumulating 33,320 EVCS hosts belonging to 22 different vendors.

B. Deployment Security

We study the susceptibility of the EVCS ecosystem to remote attacks. We aim to understand the current threat facing the EVCS ecosystem. We evaluate the security, and privacy based on the General Data Protection Regulations (GDPR) password policies compliance among other security concerns that would expose the EVCS ecosystem to a multitude of attacks. Moreover, we study the malware threat landscape by providing a deeper understanding of the current threat facing the EVCS ecosystem. To this end, we propose the framework depicted in Figure 4 for analyzing the deployment of EVCSs to assess their deployment strategies and security practices [33], [34], [35] as summarized below.

1) Authentication Secrets Leakage: We evaluate the communication protocol used by the operators to interact with the charging station management system. Namely, we try to identify the redirection to an encrypted communication channel to secure the interaction with the EVCS. Consequently, we leverage Zoomeye [29] to identify the communication protocol utilized by the charging station operators. We also confirm that by interacting with the EVCSs and transmitting a username and a password (i.e., admin, 123) using their portal we are able to identify authentication secrets transmitted in plaintext by inspecting the traffic collected using Wireshark [36]. We search for the transmitted username or password that can be leaked via the request URL and requests' payload.

2) SSLStrip Attack: To check for SSLStrip attacks, we check for the lack of HTTP Strict Transport Security (HSTS) enforcement. HSTS is a widely supported standard to protect visitors and ensure that their browsers always connect to a website over HTTPS. HSTS exists to remove the need for the common, insecure practice of redirecting users from http:// to https:// URLs. We connect to the online portal while mimicking common use case scenarios. We then utilize Burpsuite [37] to check for the lack of HSTS. Such misconfiguration means that HTTPS redirects may be putting the operators at risk. This is classified as a medium-risk vulnerability and represents low-hanging fruit for adversaries.

3) Online password Brute-Force and Rate-Limiting: Due to the connectivity of EVCSs to critical infrastructure and the features that this Web portal provides (firmware update, change configuration, etc.) protecting the EVCSs from password brute force attacks is imperative. Especially that lack of rate limiting could also lead to Denial of Service. Consequently, we use Burp Suite [37] to test the existence of rate-limiting mechanisms. To keep the load on the server minimal, we test the presence of defensive mechanisms by 50 attempts on the EVCS from a single computer. We continue to monitor the performance of the EVCS to ensure that we did not impact its performance.

4) Insecure Configuration: We investigate the usage of default configurations that are found in the manufacturer's

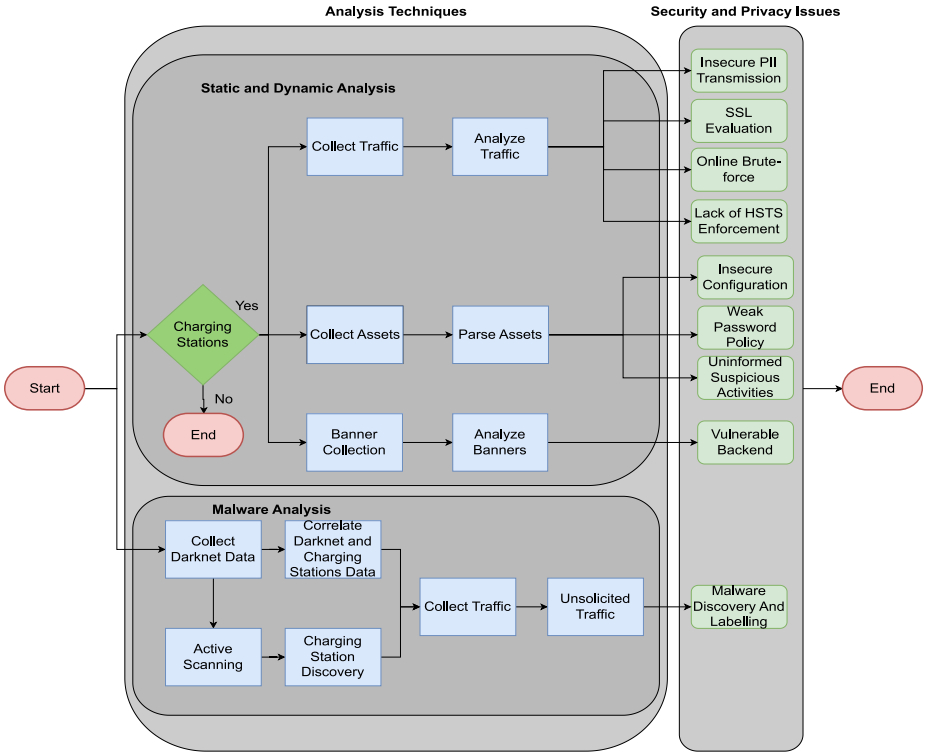

Fig. 4. Overall Deployment Security Analysis Framework.

manuals. This investigation is done to analyze the deployment security followed by the operators. Operators have exposed their devices to the Internet without taking security precautions to protect the ecosystem. Consequently, we investigate further deployment security measures of the operator by analyzing the configuration for 10 different vendors. We created an automated tool, that identifies the vendor of the target and tries one pair of login and password from the vendor's manuals, without trying to brute-force other combinations thus, minimizing our impact on the studied systems. Due to ethical concerns, the tool is specifically designed to return the count of successful logins and the IP hosts, without retrieving any information or any further access to the Web interface. We would like to highlight that this exposes the ecosystem to a Mirai-like attack vector (Mirai originally targeted services with the default configuration and brute-forced the login). However, we do not need to utilize brute force since we identified the specific login pair for each vendor accurately following our discovery methodology. The importance of such testing for insecure configuration lies in lowering barriers for the adversary to create an impact on the ecosystem and the connected power grid. The adversary can perform denial of service on the EVCS [14], [15], on the backend [38], can perform oscillatory load attacks which impact power grid stability [3], [12], [14], [15], [18]. Consequently, we reported our results by communicating with the manufacturer or the operator to help raise awareness.

5) Weak Password Policy and Uninformed Suspicious Activities: EVCS vendors provide the operators with the ability to change the password of their accounts that allow them to access the EVCS Web portal. The password policy instilled determines the flexibility of the operator to utilize weak passwords. Consequently, to review the password policies we utilize open source intelligence (e.g., manuals) or through communicating with owners of the charging stations to understand the security controls implemented for each vendor whenever possible. Moreover, we also study the features instilled to report uninformed suspicious activities such as changing passwords.

6) Backend Assessment: Due to ethical/legal concerns, we refrain from using any invasive vulnerability scanning tools to assess the backend servers. Instead, we look into the backends' software components as disclosed by Web servers frameworks in their HTTP response headers. The vulnerable backend utilized by the EVCSs exposes them to a wide range of attacks and vulnerabilities if exploited by an adversary. Consequently, we study the EVCS backend components when possible such as "Server" and "X-Powered-By" to determine the risks associated with them. We then match these components against the CVE database to detect known vulnerabilities associated with these versions since a considerable number of the CVEs exist with an exploitable proof of concept.

C. Malware Analysis

Next, we investigate the malware threat landscape in the EVCS ecosystem through the methodology in Figure 4. We start by examining the EVCSs' presence on a network telescope and extracting artifacts from their network traffic. The network telescope is a portion of IP address spaces dedicated to observing inbound Internet traffic. The main outcome of the network telescope is to detect and log malicious traffic that

originates from malware and viruses [39] that perform scanning actions by sending probes. We utilize the UC San Diego network telescope under CAIDA stewardship. The network is globally routed and accounts for approximately $\frac{1}{256^{th}}$ of all IPv4 Internet addresses that carry almost no legitimate traffic because there are few provider-allocated IP addresses in this prefix. The data is pre-processed and legitimate traffic is discarded from the incoming packets. The remaining data represent a continuous view of anomalous unsolicited traffic (e.g., the scanning of address space by attackers or malware looking for vulnerable targets) [40]. Consequently, we correlate the EVCSs discovered from our fingerprinting methodology with the CAIDA dataset by cross-referencing the two datsets. The detection is based on 3 million IP addresses that are detected on the darknet as scanners after monitoring traffic from February 2022 till October 2022. Namely we collect darknet scans around every two months on the following dates:

- 26, 27, 28 February 2022.
- 07, 08, 09, 10, 11 April 2022.
- 10, 11, 12, 13, 14 July 2022.
- 13, 14 October 2022.
- 15 March 2023 to 13 April 2023 every two days.

1) Active Scanning: Moreover, we scale our fingerprinting of EVCSs on the darknet by actively scanning the hosts with inbound traffic ($\sim$ 2 million) on 179 ports that we collected from the unique set of ports that are used by different EVCS vendors and operators as a result of our fingerprinting mechanism. We do not limit our scanning to known traditional HTTP and HTTPS ports due to the fact that EVCS manufacturers provide flexibility to operators to assign unusual ports to access their Web portals. For example, Schneider EVLink EVCSs provide flexibility to the operator to assign a port between 1 and 9999 for hosting the EVCS Web portal. Consequently, we utilize a two-stage approach to scan EVCSs to avoid being detected as malicious and scanning the whole port range. We first send TCP probing requests to determine the open ports based on the received replies. To this end, we utilize Zmap [41] which is a fast single-packet network scanner optimized for Internet-wide network surveys. We then utilize the resulting hosts with their respective ports for an application-layer handshake to retrieve and collect Web banners using Zgrab [22] which is a stateful application-layer scanner, written in Go language and supports HTTP/HTTPS protocols. Consequently, after collecting the Web banners, we extract artifacts following the proposed methodology discussed in Section IV-A. We scale our findings of EVCSs on the darknet by collecting Web artifacts and filtering the results similar to the approach discussed above. We utilize active scanning on the darknet hosts to minimize the impact of our scanning on uninfected devices in the wild.

2) Malware Family Identification: While the existence of the EVCS traffic on the darknet is proof of malicious EVCS behavior, we further our analysis of the traffic to identify the signature of the scanners/malware (e.g., Zmap, nmap, Mirai botnet, etc.). Mainly, we focus in this work on the Mirai malware and its variants. We collect the inbound traffic ($\sim$4 million packets). We note that to identify the Mirai

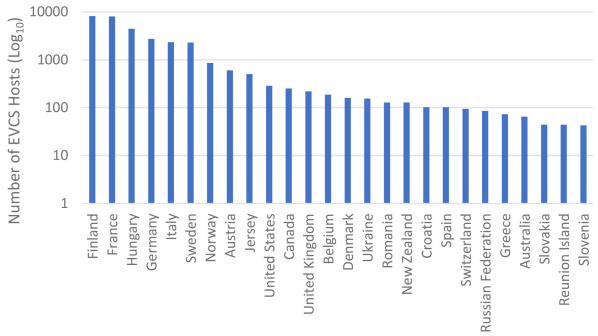

Fig. 5. EVCSs Distribution Per Country.

malware and its variants, we extract artifacts from the sent packets. Mirai and variants have a unique TCP SYN signature where the probes sent by an infected device have a TCP sequence number (normally a random 32-bit integer) equal to the destination IP address [42]. This is used to attribute the scanning to a Mirai or a variant. It is worth highlighting that Mirai traffic originating from an IP address that is associated with an EVCS is an indication that the EVCS is indeed infected [42], [43], [44], [45]. We highlight that it is statistically impossible for legitimate traffic originating from 100s of EVCSs to have a scanning signature identical to Mirai without being infected. Additionally, the data that is retained by CIADA consists entirely of malicious behavior since all legitimate traffic is filtered and discarded. While focusing on network traffic limits our result, we plan in our future work to utilize active artifact extracting tools [43] to get a deeper understanding of the other unlabelled scanners that we discovered in the EVCS ecosystem.

V. Experimental Results

We provide a detailed discussion of our results that show the exposure of EVCSs to the Internet, providing a new attack vector for adversaries to exploit. Our discovery shows the lack of proper network layer defenses to protect the charging infrastructure from remote intruders and the lack of proper security practices by the vendors and operators as they are both equally liable for securing this ecosystem.

A. EVCS Discovery

We illustrate in Figure 5 the geographical distribution of the discovered EVCSs. We show that EVCS management systems are mainly concentrated in Europe where Finland, Hungary, and France account for around 61% of all the discovered EVCS management systems. While this is expected because of the chosen scanned networks, we chose other networks in North America and discovered a low number of EVCSs with exposed management systems. This is attributed to the fact that the EVCS operators and vendors in North America utilize different types of EVCSs that do not deploy management systems per device but rather connect them to the operator's cloud management systems. Indeed, we examine the EVCS deployment of 6 different vendors in North America and discover that their deployment strategy and choice of

EVCSs are keeping them from being discovered using online tools as they do not possess any Web interface that might leak information indicating their correlation to the EVCS ecosystem. Partly, we attribute this to the strict government policies and interest in the security of the EVCS ecosystem [46]. However, we managed to identify Flo EVCSs by identifying their communication gateway that is used by the EVCS to communicate with the back-end systems. Flo is a charging station manufacturer that operates in North America. Through our analysis using Open Source Intelligence (OSINT) techniques, where we leverage, collect and analyze publicly shared information by the manufacturer/operator (e.g., commissioning guides, installation manuals, etc.) to get a deeper understanding of the deployment strategy, we identified that Flo charging stations are deployed with a Digi router, namely, DIGI INDUSTRIAL GATEWAY–COMMUNICATION NETWORK LTE (4G) AND HSPA+. Consequently, we leverage these keywords such as Digi to explore the report dataset that we collected using our aforementioned approach. After careful inspection of the retrieved candidates, we were able to identify the communication gateway. While Flo communication gateways do not provide a Web interface for configuration, however, they do possess open SSH services that are running outdated versions. Thus, identifying them is important for assessing the security of the infrastructure, especially since they are utilized to route OCPP traffic that is used to manage and configure the charging station remotely.

After identifying EVCS management systems, we group the hosts based on the extracted titles. Consequently, we identify 28 clusters of devices. We notice that Ensto, Chago, Garo, Mennekes, and Bender possess 2 clusters each which shows that there are variations of the same product. After further inspection, we identify that these products are of two different firmware versions. Moreover, we identify two EVLink signatures where the difference between these also accounts for newer firmware being deployed on the EVCS that changes the banner and the interface. For example, older EVLink EVCSs possess "Charging Station" as a title whereas newer ones possess "EVSE Web portal". We further elaborate on the security concern that arises from finding multiple signatures that could be attributed to running old firmware versions. Consequently, this shows the constant need to update and discover new signatures to identify EVCSs of known or unknown vendors. For our subsequent study, we utilize the wide range of open HTTP and HTTPS ports that are known to be used to operate an EVCS management system. We note that 53% of the discovered EVCSs operate on known HTTP and HTTPS ports (e.g., 80, 8080, 443, 8443, etc.), whereas the rest operate on unusual ports such as port 30, 10000, etc. We discover 179 unique ports where hosts operate EVCS management systems. This increases the complexity of identifying EVCS management systems. Some EVCS vendors' discoveries might be more straightforward than others. For example, Etrel EVCSs based on their installation guide recommend operating the EVCS on port 10000 and incrementing by one every time you need to add a new EVCS in the same location. Indeed, 90% of the Etrel EVCSs operate on port 10000 which aids in identifying this vendor in the future. Furthermore, EVLink

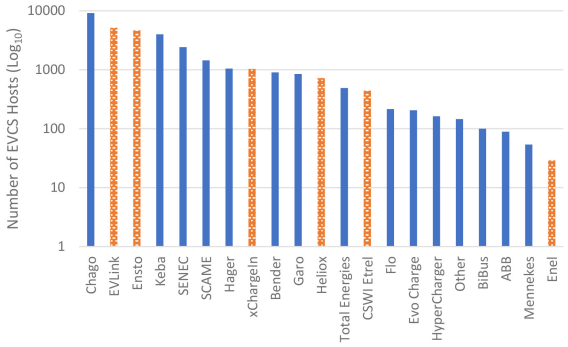

Fig. 6. EVCS Discovery Results (Uniquely discovered hosts are highlighted with solid blue).

TABLE II
PRECISION OF THE QUERIES USING DEVICE SIGNATURES

	TP	FP	Precision (%)
Query #1	4	0	100
Query #2	5	1	83.33
Query #3	19	5	79.17

which is manufactured by Schneider, operates on more than 100 ports with 97% of them operating on unusual ports such as 9100, 2082, etc. Thus, the policies instilled by the manufacturer hinder the discovery of EVCSs and add a layer of complexity in discovering them based on services and ports.

Performance Evaluation: In order to numerically assess the efficiency of the retrieval process using queries 1, 2, and 3, we utilize the precision metric. Precision is equal to $\frac{|TP|}{|FP+TP|}$, where *TP* is the number of true positive device signatures and *FP* is the number of false positive device signatures. Using precision allows us to quantify the actual proportion of EVCSs out of the total retrieved results by each query.

In Table II we show the precision of the different queries used. Queries 1 and 2 use a corpus that is directly related to the EVCS ecosystem (Chargemap and Plugshare) and achieve a precision of 100% and 83.33%. Whereas, Query 3 which uses a more general corpus (World Wide Web) achieves a precision of 79.17%. Overall, the precision of all the queries combined is 82.35%. However, the results are then vatted and validated and all non-EVCS results are discarded. As a result, the reported 33,320 hosts are verified to be purely EVCSs, and their distribution over different vendors is depicted in Figure 6.

B. Remote Compromise

Following the methodology in Section IV. We analyzed EVCS management systems which include 33,320 EVCS distributed over 22 vendors. We devise a non-invasive security approach that could be used on other cyber-physical systems to assess the risk of remote exploits. Although, these vulnerabilities that we highlight might exist in other IoT devices, however, the EVCS ecosystem is widely distributed over very large geographical areas and connected to a very critical infrastructure. Thus, the existence of such vulnerabilities is concerning. Moreover, businesses are dependent on the service it is providing, thus, providing an attack vector that would have an economic impact in case of disruption of services.

TABLE III
OVERALL RESULTS FOR SECURITY FLAWS IN EVCS MANAGEMENT SYSTEMS LABELED FOLLOWING THE THREAT MODEL: ◒ ON-PATH ATTACKER; ● REMOTE ATTACKER, BLANK: NO FLAW FOUND

Security Flaw	Attack Vector	# of Vendors	# EVCS Hosts
Insecure Configuration	●	10	5,240
Vulnerable Backend	●	3	9,150
Insecure Authentication	◒	18	28,046
Weak password policy	●	10	21,246
Uninformed Suspicious Activity	●	11	24,519
Online Password Bruteforce	●	12	22,506

Finally, this lowers the barrier for adversaries to attack the ecosystem at scale highlighting the ecosystem's widespread deployment insecurity. The EVCS management system is commissioned to manage individual EVCSs remotely. Namely, the portal provides the operator with the ability to change EVCS configuration, CMS communication, CMS control over the individual EVCS, reboot, firmware update, logs, and sensitive user information. The EVCS also provides power-related functionalities such as setting the charging rate and load shedding.

In this work, we focus on studying the ability of an on-path and remote attacker to impact and intrude into EVCSs by assessing the access control measures instilled. Through our investigation, we discover that the communication between the operator and the management systems occurs over unencrypted channels rendering them vulnerable to Man-in-the-Middle attacks impacting 28,046 EVCSs belonging to 22 vendors except for Hager and Flo as they do not provide password protection but rather a status update that the EVCS is running. EVCS provide access to their Web server over HTTP without enforcing HTTPS and HSTS to redirect the connection to a secure and encrypted one. HTTPS uses TLS (SSL) to encrypt normal HTTP requests and responses and to digitally sign those requests and responses. Thus, hindering any on-path adversary to eavesdrop on the communication and conserve the integrity of the data transferred.

Moreover, EVLink EVCSs are running a "mini-httpd 1.19 19dec2003" server, which is an early version of mini-httpd with 3 known CVEs impacting 3971 hosts. We group the hosts based on the server information and we notice the EVLink EVCSs possess two different signatures. Namely, that is because of a software update the vendor introduced. We notice that multiple devices do not provide any information about the backend system showing that some of the operators have updated their firmware. However, a considerable number did not update their firmware and accounts for 76.73% of all the discovered EVLink EVCSs. Moreover, we discover that a considerable number of EVCS are running vulnerable backends. Namely, SCAME that is running light httpd 1.4.28 that has 9 CVEs with 6 out of 9 that are of critical or high severity. This impacted 216 EVCS hosts. However, we notice that some of the EVCS operators provide partial/no information about their backend showing that the majority of the operators updated their EVCS management systems. Finally, Hager is running TwistedWeb 12.2.0 with 2 known CVEs rated as high severity impacting 963 EVCSs distributed worldwide. We note that the proper security practice is to hide the backend system operating on the EVCS and we note that the majority (94%) of the vendors provided new updates that would hide such sensitive information from adversaries.

Moreover, we study remote attacks on the EVCSs and we discover that 62.5% of the EVCSs with password protection are vulnerable to password brute-force to the management portal that is used to configure the EVCS. Moreover, we continue to study the password policies implemented by the vendor and the presence of intrusion monitoring in case of a password change on the system. The password policy implemented by EVLink, Ensto, Mennekes, Chago, Garo, Bender, EvoCharge, HyperCharge, Etrel, and SCAME is very weak and does not have a minimum requirement of digits allowing the operator to use any password weakening the security of the ecosystem. Whereas, the Keba charging station forces a minimum of 10-character passwords with no two identical characters repeated. Moreover, we note that none of these EVCSs provide a reporting service in case of a password change, which impacts 73.58% of the discovered EVCSs.

Finally, we test these hosts for insecure configuration by testing the default logins. We scrape the manuals of the EVCSs we discovered by searching for default credentials that are utilized during setup. Mainly we test that for 10 vendors EVLink, Ensto, Mennekes, Chago, Garo, Bender, EvoCharge, HyperCharger, Etrel, and SCAME. While other EVCSs are provided with different ways of configuration and setup. For example, Eaton EVCSs provide a default password to each EVCS that is found on a configuration label in the EVCS. Consequently, we utilize our tool that connects to the EVCS management system and attempts to log in using the default credentials that we identified through scraping the configuration guides with no impact on the host, although they are vulnerable to brute-force attacks. Our non-invasive tool showed that 15.7% of the EVCSs discovered are being deployed without proper security measures by the operator. We note that alongside we discover more than 200 EVCS cloud management systems belonging to Garo that are operating without authentication providing the adversary with access to scheduling, schedules, firmware updates, and EVCS status. Our tool could be used to provide adversaries with a Mirai-like attack vector, noting that the original Mirai malware targeted the Telnet services with default credentials similar to the current situation. This could be used to launch attacks against the EVCS ecosystem with the aim of impacting the connecting critical infrastructure, confidentiality, integrity, and availability of the ecosystem. The adversary, after connecting to the management portal, will have access to multiple sensitive functionalities such as the firmware update, and configuration, which could be used to hold the operator at ransom and impact the ecosystem. Thus, highlighting the important role of both the operator and the manufacturer's lack of best security practices to secure the ecosystem. We provide a comprehensive recommendation in Section V-D.

C. EVCS Malware Investigation

As part of our investigative study to identify the current imminent threat that is facing the EVCS ecosystem, we focus

on identifying whether the EVCS ecosystem is a victim of malware attacks. We then aim to identify the type of malware that is infecting the ecosystem.

We mainly focus on Mirai which utilizes scanning activities (TCP-SYN) to find victims on the Internet. Whenever the scanner receives a reply from a victim device, the malware tries to either brute-force or exploit vulnerabilities in the device. The earliest versions of Mirai started using brute force to login into unprotected telnet services. However, after posting the Mirai-source code online, Mirai variants started to appear targeting different services and customized towards certain vulnerabilities. As part of the cyber kill chain, malware propagation is crucial to increase the number of infected victims. Consequently, scanning activities are initiated by malware to probe IP addresses that are not allocated to any device but rather belong to CAIDA, thus showing the malicious intent of their activity [43]. We discover 79 EVCSs that were participating in scanning activities on the Internet. We first identify the IP addresses of EVCSs that were collected in the discovery phase, then we investigate their presence in the Darknet. This presence of an IP on the Darknet gives us a clear indication that the associated EVCS is participating in scanning activities. The results are then vetted by checking that the IP address is still connected to the same device with the same banner. Thus, we were able to confirm that the discovered devices are indeed EVCSs. The presence of malware is able to infect the ecosystem shedding light on the importance of securing this ecosystem proactively due to its connection to critical infrastructure.

Consequently, we investigate the type of malware that is infecting the EVCS instances by inspecting the packets it generates. Mirai malware creates packets with a unique signature where each probe has a unique TCP sequence number (normally a 32-bit integer), which is equal to the destination IP address [42]. We note that the probability that the TCP sequence number is equal to the destination IP address is $\frac{1}{2^{32}}$ showing that this is an accurate identification of Mirai variants [42]. Roughly, around 4 million data points were collected between January 2021 and October 2022. Consequently, following the approach suggested in [42], [47] we identify the scans that targeted the IPv4 space at an estimated rate of at least five packets per second. Through this work, we show that the EVCS ecosystem is a victim of traditional malware such as Mirai and its variants and requires extra attention due to its connection to critical infrastructure. While malware numbers might seem small in the EVCS ecosystem, we must keep in mind that the total number of public EVCSs is around 1.7, million which is still a very small number. In comparison, in 2016, when the Mirai Malware first surfaced, there were over 14.8 billion devices and Mirai infected around 600,000, representing a ratio of 40 Mirai infections per million IoT devices. Along the same lines, the ratio of infected EVCSs represents 33 Mirai infections per million EVCSs. This is to demonstrate that even though this is a relatively new environment, it is not safe from infection with Mirai malware families, however, it entails a greater risk due to the connection of the EVCS to critical infrastructure.

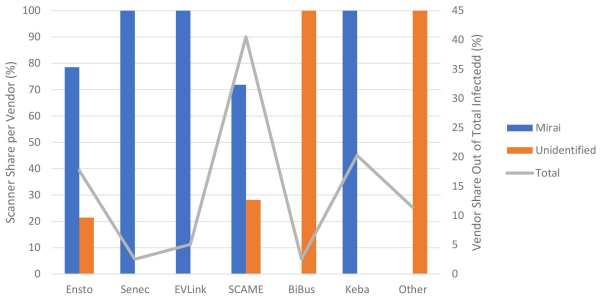

Fig. 7. Distribution among the discovered EVCS hosts.

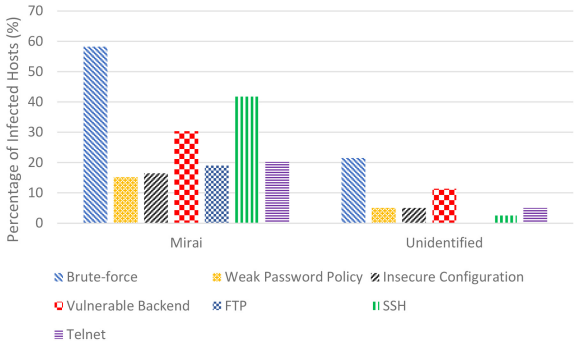

Fig. 8. Distribution of discovered security issues and open services among infected hosts.

After further investigation of the infected samples, we categorized their distribution based on the vendor in Figure 7. The columns labeled Mirai and unidentified show the percentage of each type of malware among the infected hosts from each vendor. On the other hand, the line labeled Total shows the percentage of infected hosts from each vendor with respect to the total discovered EVCS hosts on the darknet. SCAME EVCSs account for 40% out of the total number of discovered hosts on the darknet followed by Keba and Ensto accounting for 20% and 17% respectively. Moreover, the Mirai-infected EVCSs account for 70% of the infected samples. This high share of Mirai is relatively understandable as new variants have been created and launched after the leakage of the source code. We note that through our analysis of the Mirai EVCSs, they generate probing requests with an average rate of 141 packets per second showing a clear indication of maliciousness in the behavior. Whereas, for the unrecognizable scanner we identify 3 different average probing rates (30.3, 168, and 446). The different probing rates give us a clear indication of maliciousness and the possible presence of 3 different malware types other than Mirai, which require further investigation. We plan in our future work to use a real-time artifact extractor proposed in [43] to identify the type of these scanners. Furthermore, the presence of a low probing rate of 17 packets per second shows that there might be stealthy malware operating on the EVCS ecosystem.

We further investigate the presence of security issues on infected EVCSs. We illustrate the distribution of such issues in Figure 8, based on the malware type. We note that these issues could be the probable entry point of the malware to

the ecosystem. We note that 30% of the discovered Mirai-infected hosts are running vulnerable backends belonging to EVLink and SCAME. Whereas 16% are running with insecure configuration which provides the adversary with admin privileges over the EVCS management system. Consequently, an adversary can leverage the weak deployment security to inject malware into the EVCS by exploiting the weak access controls implemented and gaining access to different injection points such as the firmware update field [14], [15]. Moreover, the adversary could also modify the configuration of the EVCS and change the backend communication links which would allow them to remotely control the EVCSs using the OCPP protocol. Moreover, we note that 57% of the discovered hosts are vulnerable to brute force attacks whereas 15% possess a weak password policy that could be utilized by malware to get access to EVCS hosts. Moreover, we highlight that Mirai-infected EVCS hosts operate sensitive services that are well known to be used by malware to propagate, especially Mirai. The three main services are FTP, SSH, and Telnet where the majority of the Mirai-infected device operates at least one of these services. Moreover, we note that the malware could be infecting the embedded router of these devices exposing the ecosystem to a wide range of attacks. The existing vulnerabilities of the OCPP protocol allow adversaries to launch replay attacks [17], [48]. Thus, an infected router could be used to launch replay attacks allowing adversaries to launch oscillatory load attacks, steal electricity, and steal user information (e.g., financial information). We highlight that the responsibility behind such security concerns falls upon the vendor and the operator. Where the vendor is responsible for the policies implemented and the operator is responsible for the security beyond the deployment of EVCSs.

It is worth noting that out of completeness for our malware threat landscape analysis we investigated the presence of EVCS-specific malware by analyzing the IoTPot dataset [49] and VirusTotal. The IoTPot dataset contains 92,056 IoT malware samples collected from 2016 to 2020 and the VirusTotal dataset contains malware samples collected from 2016 to 2022. We then extract strings using the Linux string utility to create a report for each malware binary. We then search for EVCS-related keywords in their binaries. While we did not find any EVCS-specific malware, we expect to see new variants as this system is proving itself to be vulnerable to remote attacks and is already being infected by traditional malware. To ensure the fairness of our methodology due to the originally selected networks, we actively scan the darknet dataset to identify new unseen EVCS hosts from September 2022 till November 2022. We note that the EVCS hosts discovered are mainly found in Europe, where 60% are found in Italy and Sweden. Whereas the rest are distributed all over Europe (Finland, Hungary, France, Germany, Romania, Croatia) and Australia.

In 2023 alone we discovered 455 EVCS participating in unsolicited scanning. Where 57% are identified as Mirai as shown in Figure 9. In terms of geographical distribution Italy was by far the country with the largest share of infected EVCS with 40.5% of infected hosts in 2022 and 48.1% in 2023. Whereas Sweden had the second largest share at 20.25%

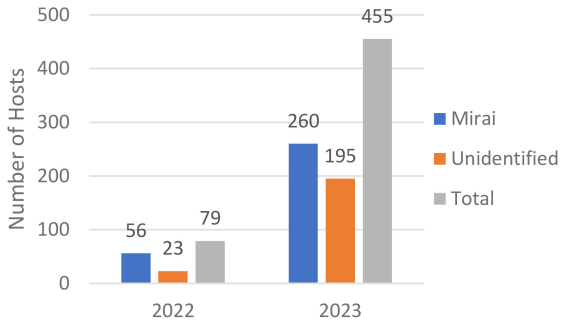

Fig. 9. Number of hosts discovered in 2022 and 2023.

in 2022 and France had the second largest share in 2023 at 25.93%. Additionally, we investigated the EVCS-specific malware by leveraging the updated IoTPot data that provides IoT malware samples from 2020 till the end of 2022. Although no specific EVCS malware was discovered. The presence of Mirai on these EVCSs proves that general IoT malware poses an imminent threat to the EVCS ecosystem.

D. Recommendations

Discovering devices is a double edge sword. Security analysts and utility operators could use it to identify EVCSs at scale, also adversaries could use it to target the EVCS instances through their vulnerable services. Various techniques could be used to protect the EVCS ecosystem, some of which are described below.

1) Manufacturer Recommendations: To ensure that the newest and urgent security patches are implemented the manufacturer should contribute to securing the ecosystem. We recommend that manufacturers implement backward-compatible over-the-air updates that allow them to push the newest updates with minimum interaction from the operator. Current methods utilized allow the operator to install the firmware manually through the configuration portal, or through OCPP, however, we notice that there is a considerable number of operators that are not updating their firmware in a timely manner. Moreover, we recommend that the manufacturer implement a strong password policy that forces the operator to change the password upon setup, making the EVCSs access more secure, or following the same method utilized by some of the Etrel EVCSs. Moreover, utilizing a notification service to inform the respective operators of security events is recommended. Consequently, we recommend utilizing Two Factor Authentication to increase the complexity for the adversary in accessing these Web portals.

2) Operator Recommendations: First, EVCS operators need to deploy a middleware that would block untrusted traffic. This is achieved based on a traffic management filter in which the filter rules rely on IP reputation and the abnormal behavior shown by the scanning parties. Operators must prevent unauthorized access to their HTTP webserver which would hinder adversaries from accessing their interface. Such techniques are basic countermeasures to prevent the characterization of EVCSs based on the services and their respective HTTP Web server. However, more advanced techniques could

be employed such as moving target defense which increases the uncertainty and complexity for attackers by reducing their window of opportunity and increasing the costs of their probing and attack efforts. Thus, changing the mapping of an internal IP address and ports to a random external port would increase the cost of detecting the exposed services by adversaries. Thus, an advanced management technique could be employed, where the EVCS would broadcast regularly to the management system the path needed to access its Web portal. The aim here is to make it harder for the adversary to access services and guess information. Moreover, we recommend following the deployment strategy adopted by ABB. Connecting to EVCSs would be through a centralized management system that the manufacturer configures for the operator such as the TerraConfig portal. Continuous patching by the operator is needed with the lack of automated firmware updates by manufacturers. Finally, ensuring communication occurs over encrypted and secure channels is of utmost importance to prevent MitM attacks.

VI. Conclusion and Future Works

In this paper, we devise a new approach for discovering EVCS that helps to expand our knowledge of the EVCS ecosystem. Our discovery technique identified more than 33,320 EVCS hosts belonging to 22 different vendors. We investigated the deployment threat landscape and the security of the EVCS ecosystem. We discovered that the lack of authentication is prevalent in the EVCS ecosystem and impacts 84% of the discovered hosts. Whereas, the password policies do not comply with the recommended GDPR policies which expose the EVCS ecosystem due to the lack of deployment security by the vendor. Consequently, we also discover that 15% of the hosts are deployed with default configuration which renders the ecosystem to Mirai-like malware that exploits the default logins to gain access to the ecosystem. We examined the services that are running on these EVCSs and were able to conclude that although manufacturers update their firmware regularly some operators fail to patch their systems. Moreover, we discover the presence of malware in the EVCS ecosystem which is still limited, however, our results show it is increasing over time. The impact of having malware on the EVCS ecosystem is drastic given its ability to impact the power grid. The presence of man-in-the-middle [17] renders the charging ecosystem vulnerable to impersonation, repudiation, and denial of service. We plan in our future work to collaborate with different charging station operators to identify charging stations that do not provide a management system to get a deeper understanding of the threat landscape. Additionally, we plan to create a method that dynamically interacts with the infected EVCSs to gather more granular artifacts to further our knowledge of the malware threat landscape. We were able through packet inspection to identify Mirai as the main contributor to the malware threat to the EVCS ecosystem. Finally, we plan to expand our study by examining the presence of malware on connected EVs and creating EVCS honeypots similar to those created for remote industrial control systems.

References

[1] R. Irle. "Global EV sales for 2021." 2021. [Online]. Available: https://www.ev-volumes.com/news/ev-sales-for-2021/

[2] IEA. "Global EV outlook 2022—Analysis." 2022. [Online]. Available: https://www.iea.org/reports/global-ev-outlook-2022

[3] M. A. Sayed, R. Atallah, C. Assi, and M. Debbabi, "Electric vehicle attack impact on power grid operation," *Int. J. Elect. Power Energy Syst.*, vol. 137, May 2022, Art. no. 107784.

[4] N. R. Canada. "Government of Canada." Oct. 2021. [Online]. Available: https://www.nrcan.gc.ca/energy-efficiency/transportation-alternative-fuels/zero-emission-vehicle-infrastructure-program/21876

[5] M. A. Sayed, M. Ghafouri, R. Atallah, M. Debbabi, and C. Assi, "Protecting the future grid: An electric vehicle robust mitigation scheme against load altering attacks on power grids," *Appl. Energy*, vol. 350, Aug. 2023, Art. no. 121769.

[6] S. Hamdare et al., "Cybersecurity risk analysis of electric vehicles charging stations," *Sensors*, vol. 23, no. 15, p. 6716, 2023.

[7] "Russian EV charging stations hacked with 'Putin is a D***head' message." Accessed: Mar. 2022. [Online]. Available: https://www.independent.co.uk/news/world/europe/putin-charging-station-hacked-ukraine-russia-b2026260.html

[8] "Isle of wight: Council's electric vehicle chargers hacked to show porn site." Accessed: Apr. 2022. [Online]. Available: https://www.bbc.com/news/uk-england-hampshire-61006816

[9] A. Laughlin. "Pod point electric car chargers: Security flaw may have put 140 000 app users' data at risk—Which? News." Accessed: Feb. 2022. [Online]. Available: https://www.which.co.uk/news/article/pod-point-electric-car-chargers-security-flaw-may-have-put-140000-app-users-data-at-risk-auIw98m8nI0u

[10] I. Zografopoulos, N. D. Hatziargyriou, and C. Konstantinou, "Distributed energy resources cybersecurity outlook: Vulnerabilities, attacks, impacts, and mitigations," *IEEE Syst. J.*, early access.

[11] D. Jafarigiv, R. Zgheib, M. Au, R. Atallah, and M. Kassouf, "An integrated transmission and distribution grid model for the cybersecurity analysis of an EV ecosystem," in *Proc. IEEE 11th Int. Conf. Smart Grid (icSmartGrid)*, 2023, pp. 1–7.

[12] M. A. Sayed, M. Ghafouri, M. Debbabi, and C. Assi, "Dynamic load altering EV attacks against power grid frequency control," in *Proc. IEEE Power Energy Soc. Gen. Meeting (PESGM)*, 2022, pp. 1–5.

[13] "Consumer watch: EV charging stations targeted by hackers." Accessed: May 2022. [Online]. Available: https://www.wvnews.com/news/wvnews/consumer-watch-ev-charging-stations-targeted-by-hackers/article_ac15597c-d554-11ec-a461-97f2cdab8157.html

[14] T. Nasr, S. Torabi, E. Bou-Harb, C. Fachkha, and C. Assi, "Power jacking your station: In-depth security analysis of electric vehicle charging station management systems," *Comput. Security*, vol. 112, Jan. 2022, Art. no. 102511.

[15] T. Nasr, S. Torabi, E. B. Harb, C. Fachkha, and C. Assi, "ChargePrint: A framework for Internet scale discovery and security analysis of EV charging management systems," in *Proc. NDSS*, 2023, pp. 1–3.

[16] Open Charge Alliance. "OCPP 2.0.1 protocols." 2021. [Online]. Available: https://www.openchargealliance.org/protocols/ocpp-201/

[17] C. Alcaraz, J. Lopez, and S. Wolthusen, "OCPP protocol: Security threats and challenges," *IEEE Trans. Smart Grid*, vol. 8, no. 5, pp. 2452–2459, Sep. 2017.

[18] K. Sarieddine, M. A. Sayed, S. Torabi, R. Atallah, and C. Assi, "Investigating the security of EV charging mobile applications as an attack surface," *ACM Trans. Cyber Phys. Syst.*, to be published.

[19] Kaspersky. "ChargePoint home security research." Accessed: Jul. 9, 2023. [Online]. Available: https://media.kasperskycontenthub.com/wp-content/uploads/sites/43/2018/12/13084354/ChargePoint-Home-security-research_final.pdf

[20] E. U. Soykan, M. Bagriyanik, and G. Soykan, "Disrupting the power grid via EV charging: The impact of the SMS phishing attacks," *Sustain. Energy Grids Netw.*, vol. 26, Jun. 2021, Art. no. 100477.

[21] "Shodan." Accessed: Jul. 9, 2023. [Online]. Available: https://www.shodan.io/

[22] Z. Durumeric, D. Adrian, A. Mirian, M. Bailey, and J. A. Halderman, "A search engine backed by Internet-wide scanning," in *Proc. 22nd ACM Conf. Comput. Commun. Security*, Oct. 2015, pp. 542–553.

[23] X. Feng, Q. Li, H. Wang, and L. Sun, "Acquisitional rule-based engine for discovering Internet-of-Things devices," in *Proc. USENIX*, 2018, pp. 327–341.

[24] T. Sasaki, A. Fujita, C. H. Gañán, M. van Eeten, K. Yoshioka, and T. Matsumoto, "Exposed infrastructures: Discovery, attacks and remediation of insecure ICS remote management devices," in *Proc. IEEE Symp. Security Privacy (SP)*, 2022, pp. 2379–2396.
[25] T. Ueda, T. Sasaki, K. Yoshioka, and T. Matsumoto, "An Internet-wide view of connected cars: Discovery of exposed automotive devices," in *Proc. 17th Int. Conf. Availability Rel. Security*, 2022, pp. 1–8.
[26] A. Costin, A. Zarras, and A. Francillon, "Towards automated classification of firmware images and identification of embedded devices," in *Proc. 32nd IFIP TC 11 Int. Conf. ICT Syst. Security Privacy Protect. SEC*, May 2017, pp. 233–247.
[27] X. Wang, Y. Wang, X. Feng, H. Zhu, L. Sun, and Y. Zou, "IoTTracker: An enhanced engine for discovering Internet-of-Things devices," in *Proc. IEEE 20th Int. Symp. World Wireless Mobile Multimedia Netw. (WoWMoM)*, 2019, pp. 1–9.
[28] D. Yu, L. Zhang, Y. Chen, Y. Ma, and J. Chen, "Large-scale IoT devices firmware identification based on weak password," *IEEE Access*, vol. 8, pp. 7981–7992, 2020.
[29] "Cyberspace search engine." Accessed: Jul. 9, 2023. [Online]. Available: https://www.zoomeye.org/
[30] "NLTK." Accessed: Jun. 2022. Accessed: Jul. 9, 2023. [Online]. Available: https://www.nltk.org/
[31] Chargemap. "Charging stations for electric cars." Accessed: Jul. 9, 2023. [Online]. Available: https://chargemap.com/
[32] Plugshare. "EV charging station map—Find a place to charge." Accessed: Jul. 9, 2023. [Online]. Available: https://www.plugshare.com/
[33] B. Reaves et al., "Mobile money, mobile problems: Analysis of branchless banking applications," *ACM Trans. Privacy Security*, vol. 20, no. 3, pp. 1–31, 2017.
[34] X. de C. de Carnavalet and M. Mannan, "Killed by proxy: Analyzing client-end TLS interception software," in *Proc. NDSS*, 2016, pp. 1–17.
[35] S. Ali, M. Elgharabawy, Q. Duchaussoy, M. Mannan, and A. Youssef, "Betrayed by the guardian: Security and privacy risks of parental control solutions," in *Proc. ACSAC*, 2020, pp. 69–83.
[36] "Wireshark." Accessed: Jul. 9, 2023. [Online]. Available: https://www.wireshark.org/
[37] "BURP suite—Application security testing software." Accessed: Jul. 9, 2023. [Online]. Available: https://portswigger.net/burp
[38] A. G. Morosan and F. Pop, "OCPP security-neural network for detecting malicious traffic," in *Proc. Int. Conf. Res. Adapt. Converg. Syst.*, 2017, pp. 190–195.
[39] U. Harder, M. W. Johnson, J. T. Bradley, and W. J. Knottenbelt, "Observing Internet worm and virus attacks with a small network telescope," *Electron. Notes Theor. Comput. Sci.*, vol. 151, no. 3, pp. 47–59, 2006.
[40] "The UCSD network telescope." Accessed: Jan. 2018. [Online]. Available: https://www.caida.org/projects/network_telescope/
[41] Z. Durumeric, E. Wustrow, and J. A. Halderman, "ZMap: Fast Internet-wide scanning and its security applications," in *Proc. USENIX Security Symp.*, 2013, pp. 605–620.
[42] M. Antonakakis et al., "Understanding the MIRAI botnet," in *Proc. 26th USENIX Security Symp. (USENIX Security)*, 2017, pp. 1093–1110.
[43] J. Khoury, M. S. Pour, and E. Bou-Harb, "A near real-time scheme for collecting and analyzing IoT Malware artifacts at scale," in *Proc. 17th Int. Conf. Availability Rel. Security*, 2022, pp. 1–11.
[44] V. Rammouz et al., "Helium-based IoT devices: Threat analysis and Internet-scale exploitations," in *Proc. 19th Int. Conf. Wireless Mobile Comput. Netw. Commun. (WiMob)*, 2023, pp. 206–211.
[45] M. S. Pour, C. Nader, K. Friday, and E. Bou-Harb, "A comprehensive survey of recent Internet measurement techniques for cyber security," *Comput. Security*, vol. 128, May 2023, Art. no. 103123.
[46] K. Harnett et al., *DOE/DHS/DOT Volpe Technical Meeting on Electric Vehicle and Charging Station Cybersecurity Report*, John A. Volpe Nat. Transp. Syst. Center, Cambridge, MA, USA, 2018.
[47] Z. Durumeric, M. Bailey, and J. A. Halderman, "An Internet-wide view of {Internet-wide} scanning," in *Proc. 23rd USENIX Security Symp. (USENIX Security)*, 2014, pp. 65–78.
[48] Z. Garofalaki, D. Kosmanos, S. Moschoyiannis, D. Kallergis, and C. Douligeris, "Electric vehicle charging: A survey on the security issues and challenges of the open charge point protocol (OCPP)," *IEEE Commun. Surveys Tuts.*, vol. 24, no. 3, pp. 1504–1533, 3rd Quart., 2022.
[49] Y. M. P. Pa, S. Suzuki, K. Yoshioka, T. Matsumoto, T. Kasama, and C. Rossow, "IoTPOT: Analysing the rise of IoT compromises," in *Proc. 9th USENIX Workshop Offensive Technol. (WOOT)*, 2015, pp. 1–8.