From Exposed to Exploited: Drawing the Picture of Industrial Control Systems Security Status in the Internet Age

Yixiong Wu[1], Jianwei Zhuge[1,2], Tingting Yin[1], Tianyi Li[3], Junmin Zhu[4], Guannan Guo[5], Yue Liu[6] and Jianju Hu[7]

[1] *Institute of Network Science and Cyberspace, Tsinghua University, Beijing, China*

[2] *Beijing National Research Center for Information Science and Technology, Beijing, China*

[3] *Peking University, Beijing, China*

[4] *Shanghai Jiao Tong University, Shanghai, China*

[5] *School of Computer Science and Technology, University of Science and Technology of China, Hefei, China*

[6] *Qi An Xin Technology Research Institute, Beijing, China*

[7] *Siemens Ltd., China*

Keywords: Internet-facing ICS Devices, Passive Vulnerability Assessment, Device Search Engine.

Abstract: The number of Internet-facing industrial control system(ICS) devices has risen rapidly due to remote control demand. Going beyond benefits in maintenance, this also exposes the fragile ICS devices to cyber-attackers. To characterize the security status of Internet-facing ICS devices, we analyze the exposed ICS devices and their vulnerabilities. Considering the ethic, we design and implement ICScope, a passive vulnerability assessment system based on device search engines. Firstly, ICScope extracts the ICS device information from the banners returned by multiple search engines. Then, ICScope filters out the possible ICS honeypots to guarantee accuracy. Finally, ICScope associates ICS vulnerabilities with each ICS device. Over the past year, our measurements cover more than 466,000 IPs. We first perform a comprehensive measurement of Internet-facing ICS devices from Dec 2019 to Jan 2020. We find that there are about 49.58% of Internet-facing ICS devices that can be identified are affected by one or more vulnerabilities. We also conduct three times experiments from Jun 2020 to Dec 2020 to monitor the security status of Internet-facing ICS devices. We observe a slowly decreasing trend in the number of vulnerable ICS devices during our experiment period.

1 INTRODUCTION

With the rapid growth and deep integration of Information Technology and Operational Technology, more and more industrial control system (ICS) devices with computing and communication capabilities are introduced to smart factories, buildings, and cities. The proprietary protocols used in these devices are originally designed to communicate between devices in isolated operation LANs. As a result, security protections for ICS communication are usually missing. Besides, for economic considerations, ICS devices are rarely updated or upgraded. Administrators usually avoid upgrades to reduce the possibility of system crashes. Thus, the life cycle of ICS vulnerabilities is much longer than other kinds of vulnerabilities. However, more and more ICS devices are exposed to the Internet nowadays. This is usually for the convenience of data collection and remote maintenance. However, it also makes the fragile ICS devices exposed to the cyber-attacks. In 2017, Triton attacked a petrochemical plant in the Middle East, resulting in an emergency shutdown of a critical industrial control system (Di Pinto et al., 2018). In 2019, the Triton was relaunched. Cyber-attacks like these towards ICS have existed for a long time and have already caused substantial damage.

To mitigate this type of threat, both the research community and the industry take proactive actions. ICS-CERT[1] provides annual assessment reports for public ICS vulnerabilities. Kaspersky [2] further analyzes these reports and summarizes the vulnerability characteristics. However, both of them did not an-

[1] https://us-cert.cisa.gov/ics

[2] Industrial Control Systems Vulnerabilities Statistics, Reported by Kaspersky in 2019

Wu, Y., Zhuge, J., Yin, T., Li, T., Zhu, J., Guo, G., Liu, Y. and Hu, J.
From Exposed to Exploited: Drawing the Picture of Industrial Control Systems Security Status in the Internet Age.
DOI: 10.5220/0010327902370248
In *Proceedings of the 7th International Conference on Information Systems Security and Privacy (ICISSP 2021)*, pages 237-248
ISBN: 978-989-758-491-6

alyze the distribution of vulnerabilities in the actual network. Existing works also focus on the discovery of online ICS devices. Both Mirian et al. (Mirian et al., 2016) and Feng et al. (Feng et al., 2016) have proposed their device discovery approaches to give an Internet-Wide view of ICS devices. However, they focus more on the number of exposed devices rather than vulnerable devices. Further works behind online ICS devices' detection are to perform a vulnerability assessment for the Internet-facing ICS devices. Characterizing the vulnerable Internet-facing ICS devices is highly desirable for the interests of securing Internet-facing ICS devices.

To identify a device's vulnerabilities, one instinctive reaction is to perform a vulnerability assessment using vulnerability scanners like Nessus (*Active Mode*). However, such techniques cannot be applied to identify an ICS device's vulnerability. Because sending many packets to ICS systems regards as an intrusive behavior. Therefore, we prefer the passive vulnerability assessment (*Passive Mode*), which is based on a search engine to identify a device's vulnerabilities without any interaction. Most previous works on passive vulnerability assessment focus on common services, i.e., HTTP, IMAP. For example, ShoVAT (Genge and Enăchescu, 2016) focuses on several common services, and Scout (O'Hare et al., 2019) only supports HTTP service. Because of the inconsistent banner format between common services and ICS devices, their methods are not considered feasible. There are also existing works focus on ICS services. Both Samtani et al. (Samtani et al., 2016) and Leverett et al. (Leverett and Wightman, 2013) identify the vulnerabilities for online ICS devices. However, the scope of their works is limited to special ICS services. Moreover, both of them scan target ICS devices. In this paper, we aim to answer the following question:

> ***Q:*** What is the security status of Internet-facing ICS devices in the whole public IP address space?

To answer this question, we propose a passive vulnerability assessment system based on device search engines. For ethical consideration, our system ICScope uses passive mode to avoid intrusive behaviors. We face three main technical challenges to build ICScope. *1)* The information returned by device search engines is often incomplete. *2)* There are many ICS honeypots in the results of device search engines. *3)* It is not easy to accurately find the vulnerabilities that affect the ICS devices.

To address these challenges, we build ICScope with data enrichment module, honeypot detection module, and vulnerability association module. For the first challenge, the data enrichment module integrates the device information extracted from multiple search engines to enrich the incomplete device information. It also uses the fingerprint database. For the second challenge, the honeypot detection module utilizes the non-interaction features to detect ICS honeypots, such as ISP, ICS honeypot fingerprinting. It also detects ICS honeypots by comparing the results between multiple search engines. For the last challenge, the vulnerability association module builds vulnerability trees to identify each ICS device's vulnerabilities.

We have implemented ICScope and performed a large-scale measurement of ICS devices' security status in the whole public address space. We collect 270,283 Internet-facing ICS devices on four device search engines from Dec 2019 to Jan 2020. Among these ICS devices, ICScope has detected 21,578 ICS honeypots (about 7.98%). Excluding these ICS honeypots, we observe that 106,382 ICS devices can extract complete ICS device entries. And ICScope regards 52,739 of them (about 49.58%) as vulnerable ICS devices. We further analyze the vulnerable ICS devices and related vulnerabilities. We find that the vulnerable ICS devices are distributed in the vast majority of countries. Among the vulnerabilities, most of them are in high or critical severity level and can be remotely exploitable with little or no limitation. Over six months, our measurement shows a slowly decreasing trend in the number of vulnerable ICS devices. We also use ICScope to measure the impact of a 0day vulnerability. The result shows that the vulnerability affects 3,999 ICS devices.

Contribution. Our contributions can be summarized as follows:

1. We design and implement a passive vulnerability assessment tool ICScope. ICScope can automatically obtain ICS devices' information from multiple device search engines and correlate the vulnerabilities of each device.
2. By using ICScope, we perform the large-scale measurement on the security status of all Internet-facing ICS devices in the whole public IP address space and find that 49.58% of the Internet-facing ICS devices with complete device entries are vulnerable. We also further characterizing the vulnerable ICS devices.
3. We proposed a non-interaction ICS honeypot detection approach to detect the ICS honeypots without intrusive behaviors. This method has detected 21,578 ICS honeypots (about 7.98%) on our datasets.

Our results reveal the severe and considerable security risks faced by ICS devices in the current cyberspace. At the end of the paper, we discuss the mitigation measures against this security threat.

2 MOTIVATION AND CHALLENGES

2.1 Motivation

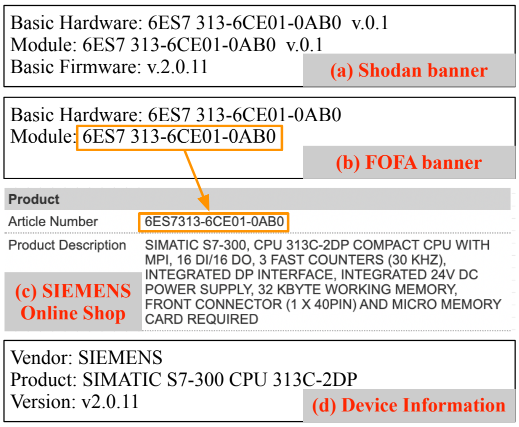
The risk of Internet-facing ICS devices is a persistent problem. Most existing works focus on what devices have been exposed to the Internet or the attacker's malicious behaviors (Fachkha et al., 2017). Researchers perform vulnerability assessments for Internet-facing ICS devices (Samtani et al., 2016; Leverett and Wightman, 2013). However, these works are usually limited in scale and scope. More importantly, their approaches cannot avoid intrusive behaviors.

In this paper, we seek to provide a more comprehensive and holistic view of Internet-facing ICS devices' security status via a large-scale empirical measurement. We obtain more comprehensive Internet-facing ICS devices by integrating multiple search engines' data and perform passive vulnerability assessment on them. We further analyze the vulnerable ICS devices and related vulnerabilities across multiple dimensions, e.g., geolocation distribution, vulnerability ratio. We also conduct a non-interaction ICS honeypot detection to decrease false positives while previous vulnerability assessment works not.

2.2 Challenges

There are three main challenges in designing a passive vulnerability assessment system.

Incomplete Banners. Not all banners returned by device search engines contain complete device information. For example, the different banners of the same ICS device are respectively returned by Shodan and FOFA (see Figure 1a and Figure 1b). Compared to Shodan, the FOFA's banner lacks the *Basic Firmware* information used to extract version information. Moreover, both of them do not contain vendor and product information. Existing works that reconstruct CPE names from banners are limited to the lack of device information. Fortunately, we observe that device search engines' capabilities are different, and part of device information can be identified through a particular field in ICS protocol. Based on this observation, we try to extract complete device information from banners returned by multiple device search engines based on ICS protocol features. For the ICS device in Figure 1, we extract its version from the Shodan banner. Through the article number found in the banner, we obtain its vendor and product from the SIEMENS online shop.



```
Basic Hardware: 6ES7 313-6CE01-0AB0  v.0.1
Module: 6ES7 313-6CE01-0AB0  v.0.1
Basic Firmware: v.2.0.11
```

```
Basic Hardware: 6ES7 313-6CE01-0AB0
Module: 6ES7 313-6CE01-0AB0
```

Product	
Article Number	6ES7313-6CE01-0AB0
Product Description	SIMATIC S7-300, CPU 313C-2DP COMPACT CPU WITH MPI, 16 DI/16 DO, 3 FAST COUNTERS (30 KHZ), INTEGRATED DP INTERFACE, INTEGRATED 24V DC POWER SUPPLY, 32 KBYTE WORKING MEMORY, FRONT CONNECTOR (1 X 40PIN) AND MICRO MEMORY CARD REQUIRED

```
Vendor: SIEMENS
Product: SIMATIC S7-300 CPU 313C-2DP
Version: v2.0.11
```

Figure 1: Device information for SIMATIC S7-300 with banners returned by Shodan and FOFA.

Honeypot Detection. Industrial honeypots are wildly used in the detection of industrial control security threats (Serbanescu et al., 2015). The device search engines also add honeypots to the query results to monitor the abuse of them by attackers. Previous works (Feng et al., 2016) that utilize network fingerprinting technology (Comer and Lin, 1994) are not working here due to the lack of interaction. We leverage the non-interaction features to detect ICS honeypots, such as the fingerprints of open-source ICS honeypots and the device inconsistencies in different search engines.

Associate Vulnerabilities. Previous works (Genge and Enăchescu, 2016; O'Hare et al., 2019) utilize the most similar CPE name reconstructed from banner to find possible vulnerabilities. However, this approach is limited to ICS. First, some ICS devices might have more than one CPE name, while some have none. For example, the device's CPE name in Figure 1 can be `cpe:/h:siemens:simatic_s7-300_cpu:2.0.11` or `cpe:/h:siemens:simatic_s7-300_cpu_313:2.0.11`. Second, CPE data used to identify the impact range of a vulnerability in the National Vulnerability Database (NVD) is exaggerated or underestimated (Dong et al., 2019). Rather than reconstructing the CPE name, we construct vulnerability trees from multiple public vulnerability libraries and use them to associate vulnerabilities.

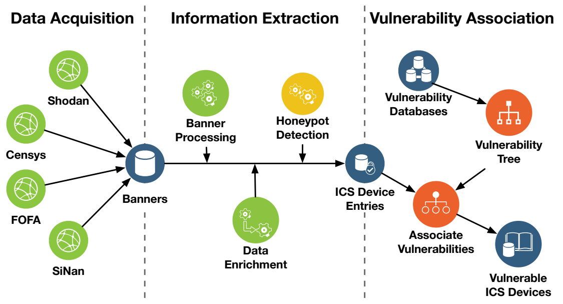


Figure 2: Architecture of the ICScope.

3 ICScope

In this work, we design and implement a search engine based automatic passive vulnerability assessment tool for ICS, ICScope, and present its design in this section.

3.1 Overview

The overall architecture of ICScope is shaped in Figure 2, which consists of three modules. First, the *data acquisition module* (§3.2) collects all Internet-facing ICS device data via the Application Programming Interface (API) of search engines. Then, the *information extraction module* (§3.3) extracts device information, i.e., vendor names, product names, and versions, from the ICS device banners returned by search engines. The *information extraction module* (§3.3.3) also detects and excludes the ICS honeypots in the results. Finally, the *vulnerability association module* (§3.4) constructs vulnerability trees from the public vulnerability libraries and utilizes it and the extracted device information to associate possible vulnerabilities.

3.2 Data Acquisition

The search engines provide APIs with search filters, which are special keywords for special properties. For example, `protocol:s7` is used to find all Internet-facing ICS devices in FOFA[3], which use the S7 protocol. As reported by Guo et al., the data integration of multiple device search engines can obtain more comprehensive Internet-facing ICS devices (Guo et al., 2018). Table 1 shows the industrial control protocols and device search engines supported by ICScope. ICScope leverages the APIs and special search filters to acquires all Internet-facing ICS devices from different device search engines' databases.

Table 1: Protocols supported by ICScope. ✓: supported by device search engine; △: enriched by ICScope.

Protocol Name	Device Search Engines				Enrich
	Shodan	Censys	FOFA	SiNan	
Modbus	✓	✓	✓	✓	-
Siemens S7	✓	✓	✓	✓	△
DNP3	✓	✓	✓	✓	-
Niagara Fox	✓	✓	✓	✓	△
BACnet	✓	✓	✓	✓	△
EtherNet/IP	✓	-	✓	✓	△
GE-SRTP	✓	-	✓	✓	-
HART-IP	✓	-	✓	✓	-
PCWorx	✓	-	✓	✓	△
MELSEC-Q	✓	-	✓	✓	-
OMRON FINS	✓	-	✓	✓	△
Crimson v3	✓	-	✓	✓	-
CoDeSys	✓	-	✓	✓	△
IEC 60870-5-104	✓	-	✓	✓	-
ProConOS	✓	-	✓	✓	-
VertX Edge	-	-	✓	✓	△
Moxa NPort	-	-	✓	✓	-
Lantronix UDP	-	-	✓	✓	-
Koyo DirectNet	-	-	-	✓	-
HollySys MACS	-	-	-	✓	-
Proficy-webspace	-	-	-	✓	-
Siemens License	-	-	-	✓	-
Total	15	5	18	22	8

3.3 Information Extraction

As shown in Figure 2, the information extraction module consists of three submodules:

3.3.1 Banner Processing

This submodule takes the banners as input and outputs the device information. An ICS device entry can be characterized by the tuple $\langle vendor, product, version \rangle$. Except for the ICS device entry, device information also includes feature information for data enrichment or honeypot detection, i.e., the article number in the S7 protocol. However, the fields of device information are different in different ICS protocols. We analyzed 22 ICS protocols and wrote parsers for them separately.

3.3.2 Data Enrichment

The challenge here is that the banners returned by search engines are incomplete. To address this, we take the following three ways to enrich the incomplete ICS device entry:

Data Enrichment based on Industrial Control Protocol. Some attributes of ICS device entry might show as identifiers rather than the exact value in some ICS protocols. For example, BACnet defines a numerical ”vendor identifier” to arbitrate between non-standard

[3] https://fofa.so/

messages of different manufacturers. Each vendor identifier has been assigned to a vendor. For the proprietary protocols, we can identify their vendors directly, e.g., SIEMENS for S7 protocols. Based on these observations, we manually build several mappings between identifiers and exact values. ICScope utilizes these mappings to supplement the missing attributions.

Data Enrichment based on Fingerprint Database. Some banner fields can be used to identify the device information, such as the product-ID system's field. An example is the article number `6ES7 313-6CE01-0AB0` in Figure 1, which is the SIEMENS product-ID system's identifier. We utilize these fields to build the fingerprint database for ICS devices.

Data Enrichment based on Multiple Search Engines. Device search engines perform differently on each ICS protocols. Excepting the number of discovered devices, the banners can also be different. As shown in Figure 1, Shodan can acquire *Basic Firmware*, but FOFA not. The strategy is to integrate the device information extracted from multiple search engines for complete ones. ICScope supports four search engines: Shodan, Censys (Durumeric et al., 2015), FOFA, and SiNan[4].

3.3.3 Honeypot Detection

To achieving accuracy in discovering vulnerable Internet-facing ICS devices, ICScope detects ICS honeypots within the results in passive mode. ICScope adopts the following approaches:

Multi-source Comparison. Although the search engines' scanning time is different, the industrial control system devices are usually neither upgraded nor replaced for the duration. So the data information for an ICS device should be the same at different search engines. If the data information is inconsistent, we regard this ICS device as an ICS honeypot. Notice that if multiple ICS protocols are found for one ICS device, we can not verify whether it is an ICS honeypot based on current features. We regard them as ICS honeypots to decrease false positives. After inspecting these IPs, we find they might be verified by the non-ICS ports' information, such as the HTTP service ports. We leave it as our future work.

Fingerprinting Detection. We utilize fingerprint-based methods to detect ICS honeypots. We extract characteristics from the open-source industrial honeypot as fingerprints, such as default configuration, prompt information.We also find that some fields in the ICS protocol are assigned to a unique device, i.e., the MAC in Moxa NPort protocol. ICScope leverages such fingerprinting to detect ICS honeypots.

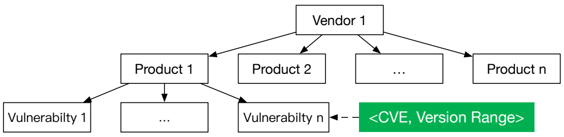


Figure 3: Data structure used to store vulnerabilities (Vulnerability Tree). Green color denotes the information stored on the leaf.

ISP-based Detection. ICS honeypots are often deployed on cloud service. However, no cloud service provider has provided online industrial control device services. Therefore, we can reasonably infer that the devices with an IP belonging to cloud service providers are ICS honeypots.

3.4 Vulnerability Association

Vulnerability Tree. We adopt a multi-forked tree forest to store all vulnerability information, and each tree represents all vulnerabilities that affected the same vendor. As shown in Figure 3, each tree takes the vendor as the root node. The intermediate nodes are the vulnerable products that belong to the vendor, and each leaf node represents a related vulnerability. Each leaf node stores the CVE name and corresponding version range of the vulnerability. Notice that the different intermediate nodes might denote the same device. For example, the ICS device in Figure 1 has two different intermediate nodes (`simatic_s7-300_cpu` and `simatic_s7-300_cpu_313`) on the vulnerability tree. For more comprehensive vulnerabilities, we use multiple vulnerability libraries (NVD, China National Vulnerability Database of Information Security (CNNVD), and SecurityFocus) to construct the vulnerability trees. We also use the data from Exploit-DB, SecurityFocus, PacketStorm to determine whether there is a public exploit for a vulnerability.

Associate Vulnerabilities. We utilize the information extraction results and vulnerability tree to associate ICS devices with vulnerabilities. We adopt a version-based associating scheme. We assume that an ICS device is affected by a vulnerability when its version is in the vulnerability affected scope. The overall association strategy is shaped in Algorithm 1. For each ICS device, we first use the `sim_lookup()` function to find similar vendors and products in the vulnerability tree. Through the vulnerability tree, we decrease unnecessary comparisons. Second, we extract all vulnerabilities associated with similar vendors and products in the vulnerability tree. For each vulnerability, we check whether the ICS device's version is in the vulnerability affected scope. If the version is affected,

[4] An internal device search engine newly developed by QiAnXin, china in 2018.

Algorithm 1: Associate ICS device with vulnerabilities.

```
Input: ICS device entry $\langle vendor, product, version \rangle$, Vulnerability Tree($VT$)
Output: a set of possible CVE names ($R_{cve}$)
  $R_{cve} \leftarrow \{\}$
  $vt\_vendors \leftarrow$ all root nodes of $VT$
  for each $vt\_vendor \in vt\_vendors$ do
    if $sim\_lookup(vendor, vt\_vendor)$ then
      $vt\_products \leftarrow$ all sub-nodes of $vt\_vendor$ in $VT$
      for each $vt\_product \in vt\_products$ do
        if $sim\_lookup(product, vt\_product)$ then
          $vt\_vuls \leftarrow$ all leaves of $vt\_product$ in $VT$
          for each $\langle cve, versionRange \rangle \in vt\_vuls$ do
            if $match(version, versionRange)$ then
              $R_{cve} \leftarrow R_{cve} + cve$
            end if
          end for
        end if
      end for
    end if
  end for
```

we add the CVE name of the vulnerability into the set of possible CVE names.

4 DEMYSTIFYING INTERNET-FACING ICS DEVICES SECURITY STATUS

In this section, we use ICScope to evaluate the security status of Internet-facing ICS devices in the whole public IP address space. We first introduce the datasets used by ICScope and then we analyze the security status of Internet-facing ICS devices. Finally, we compare our work with the vulnerability detection function provided by Shodan.

4.1 The Datasets of ICScope

The data collected by ICScope can be divided into two parts: data of Internet-facing ICS devices from the device search engine and vulnerability information from public vulnerability database. We retrieve Internet-facing ICS devices from the four device search engines between Dec 2019 and Jan 2020. We obtain 76,489/219,238/78,363/85,531 device information from Shodan, Censys, FOFA and SiNan, respectively. The device number of FOFA is much larger than the others because the time span of the FOFA is one year, but the others not. The independent IPs for the above results is 270,283. To monitor Internet-facing ICS devices, we also conducted the experiments per three months from Jun 2020 to Dec 2020. We obtain 229,048/225,473/242,034 Internet-facing ICS devices in Jun 2020, Sep 2020, and Dec 2020, respectively. We crawl the published vulnerability information from NVD, CNNVD, Exploit-DB, SecurityFocus and PacketStorm, and merged all the information into 286,269 distinct enriched vulnerability records, of which 160,795 are from CVE and 129,653 are from NVD.

4.2 Discovery Approaches Validation

First, we evaluate the performance of our honeypot detection module. Shodan provides an interface[5] to query the probability that an IP is a honeypot. Among the above ICScope datasets, Shodan only ensures 55 IPs as honeypots. We use these IPs as a cross-validation dataset to validate the honeypot detection module results. For these ICS honeypots, ICScope can detect 52 of them (about 94.55%). We manually inspect the remaining 3 IPs. We notice that the information extraction module returns none for two of them. However, our approach is based on the device information returned by the information extraction module. The last one seems like a normal ICS device. The reason might be the limitation of the features in our approach.

Secondly, we evaluate the precision of our device discovery approach. The ICS devices reside in the remote Internet space. We cannot perform active-mode scanning due to ethical considerations. We also try to contact several vendors with their devices exposed, but none of them accept our request to fill a survey obtaining the ground truth of the exposed devices. Therefore, we report the 319 vulnerable ICS devices located in our own country to the relevant Computer Emergency Response Team (CERT). Until the time of paper submission, the CERT team has dealt with 59 of them and they confirm that ICScope identifies the vendor, product and version information and the associated vulnerabilities of all 59 devices correctly. The results show that our device discovery approach has high accuracy. And we plan to share the ICScope tool and discovered vulnerable ICS devices to the interested CERT teams and help to deal with the exposed and vulnerable ICS devices.

4.3 The Honeypots in Internet-facing ICS Devices

Among the 270,283 Internet-facing ICS devices, ICScope regards 21,578 of them (about 7.98%) as ICS honeypots. Among these ICS honeypots, the multi-source comparison-based approach detects 18,428 of them (about 85.40%), while the ISP-based approach

[5] https://honeyscore.shodan.io/

detects 3,653 (about 16.93%). Furthermore, the fingerprinting-based approach detects 1,216 (about 5.64%). Notice that an ICS honeypot might occur multiple times on the above results.

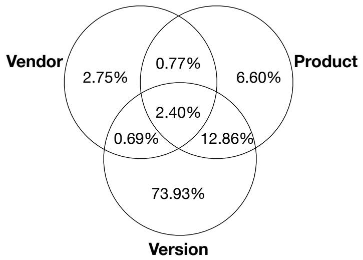
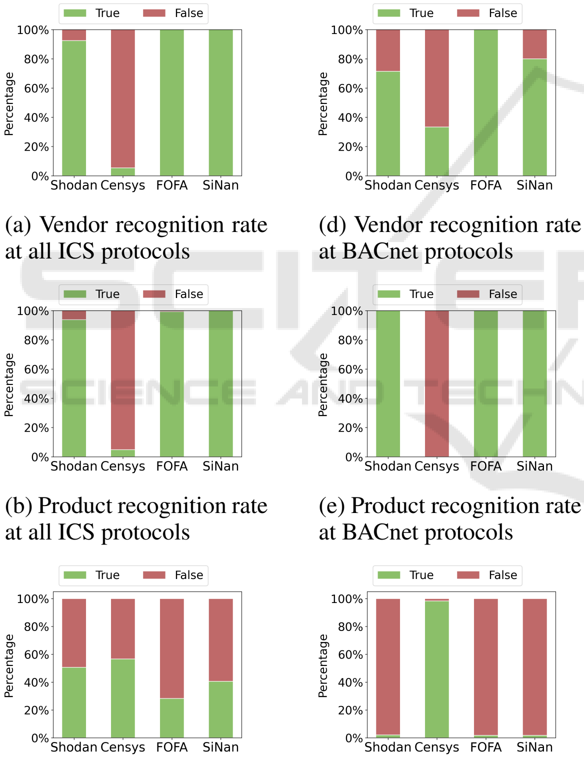


Figure 4: Attribution-distribution of inconsistent ICS devices. The three circles represent the percentage of devices with *vendor*, *product*, and *version* inconsistency among all inconsistent devices.



(a) Vendor recognition rate at all ICS protocols

(d) Vendor recognition rate at BACnet protocols

(b) Product recognition rate at all ICS protocols

(e) Product recognition rate at BACnet protocols

(c) Version recognition rate at all ICS protocols

(f) Version recognition rate at BACnet protocols

Figure 5: Recognition rate of different attributions.

4.4 The Advantage of Multiple Device Search Engines

Inconsistencies in Device Search Engines. We first analyze the inconsistencies between the different search engines. Since the probability of upgrading or replacing the device during the scanning cycle is extremely low, the differences indicate that it may be a honeypot. We are mainly concerned about the inconsistency of the *vendor*, *product*, and *version*. As shown in Figure 4, most of the inconsistencies are related to the *version*. Devices with *version* inconsistency account for 89.89% of the total inconsistent devices. It is worth noting that search engines identified entirely different results on 2.4% of inconsistent devices. We tag these inconsistent devices as honeypots.

Data Enrichment with Multiple Device Search Engines. As mentioned in Section 2.2, the results of search engines, including incomplete data. This problem can be alleviated by data enrichment. Different search engines have different ability to recognize different attributes of devices. We extract the attributes that can be identified by different search engines and combine them to get complete device information.

Figure 5 shows the different abilities of the four search engines on recognizing different device attributes. From Figure 5a, Figure 5b and Figure 5c, we can know that the search engines, except for Censys, performs well in recognition of *vendor* and *product*. But all the four search engines are weak in recognizing *version*, among which, Censys get the best recognition rate of 56.6%. Taking the BACnet protocol as an example, we can obtain the *version* information from Censys and the other information from the other three search engines (See Figure 5d, Figure 5e, Figure 5f).

4.5 The ICS Devices Affected by Public Vulnerabilities

For the 22 industrial control protocols mentioned in this paper, we can completely extract vendor, product and version information in only 9 of them (See Table 2). We obtain 215,649 ICS devices using one of the 9 ICS protocols from device search engines. After excluding 15,294 devices that may be industrial controlled honeypots, 106,382 (about 53.10%) devices can extract complete information. For those devices with complete information, there are 52,739 (about 49.58%) devices affected by one or more vulnerabilities. Figure 6 shows the percentage of vulnerable ICS devices in different protocols. Next, we focus on the analysis of the impact of public vulnerabilities on these 52,739 devices.

Protocol-distribution of Vulnerable ICS Devices. Table 2 shows the impact of public vulnerabilities in ICS devices. We find that about half of Internet-facing ICS devices are still vulnerable, thus having the risks of being compromised. More than half of the ICS protocols have at least 40% vulnerable de-

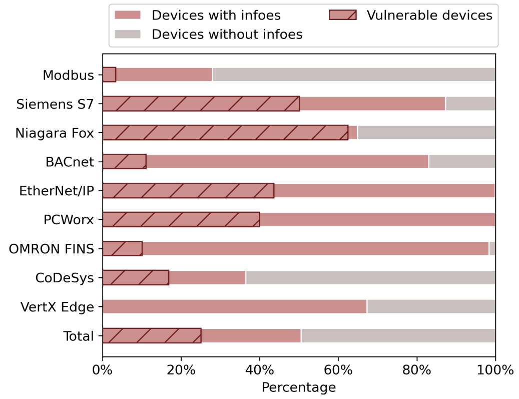


Figure 6: Vulnerable device percentage at protocol-level.

Table 2: Statistics result of vulnerable devices.

Protocol Name	Vulnerable	Devices	Vulnerable Percentage
Modbus	3461	28600	12.10%
Siemens S7	4400	7663	57.42%
Niagara Fox	35669	37060	96.25%
BACnet	1918	14405	13.31%
EtherNet/IP	4006	9171	43.68%
PCWorx	1444	3610	40.00%
OMRON FINS	208	2032	10.24%
CoDeSys	1633	3526	46.31%
VertX Edge	0	315	0.00%
Total	52739	106382	49.58%

**Devices:* refers to total number of ICS devices with complete vendor, product and version information.

vices. In particular, the percentage of vulnerable ICS devices using Niagara Fox protocol, the most serious protocol, is close to 100%. For Vertx Edge, we only find one remote code execution vulnerability of an HID VertX/Edge device in the public vulnerability databases, more specific, from SecurityFocus and PacketStorm. However, there is no detailed impact scope of this vulnerability. Thus the assessment result for VertX Edge is 0.00%.

Geo-distribution of Vulnerable ICS Devices. We adopt the Hilbert curve (Moon et al., 2001) to visualize the vulnerable ICS device distribution in the Internet space. As shown in Figure 7, the distribution of vulnerable ICS devices is irregular. It is hard to find any correlation between vulnerable ICS devices and IP. Although we cannot find any correlation at IP-layer, we can also investigate the correlation between spatial location and vulnerable ICS device distribution. Figure 8 shows the geographical distribution of Internet-facing ICS devices affected by public vulnerabilities. These vulnerable ICS devices are distributed in the vast majority of countries around the world, among which the United States, Italy, Canada, France and Spain are on the top. In particular, the United States is the only country that has over 10,000 vulnerable devices (21,156), which accounts for 40.11% of all vulnerable Internet-facing ICS devices. We further analyze the proportion of vulnerable ICS devices in each country. Among the six continents, North America has the highest proportion of vulner- able ICS devices, with an overall rate of 39.48%. This is mainly because the proportion of vulnerable ICS devices in the United States and Canada with more than 1000 ICS devices is as high as 40.72% and 35.48%, respectively. The rest are Australia (20.99%), South America (20.23%), Europe (19.93%), Asia (11.66%) and Africa (9.15%). As shown in Figure 9, more than 90% of the Niagara Fox vulnerable devices can be found in the top 10 countries, and there is a long-tail effect in their vulnerable ICS devices distribution. There are similar distribution characteristics in other vulnerable ICS devices. At the city-level, Figure 10 shows a long-tail effect in vulnerable ICS devices. The Niagara Fox protocol has the most extensive coverage. More than 43% of the Niagara Fox vulnerable devices are located in the top 50 cities, 60% of the Niagara Fox are in the top 134 cities. OMRON FINS has the

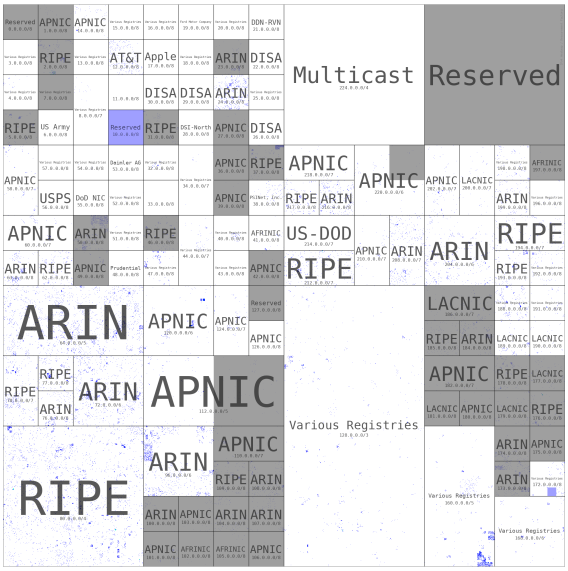


Figure 7: Hilbert curve heatmap of vulnerable ICS devices in the IPv4 addresses space.

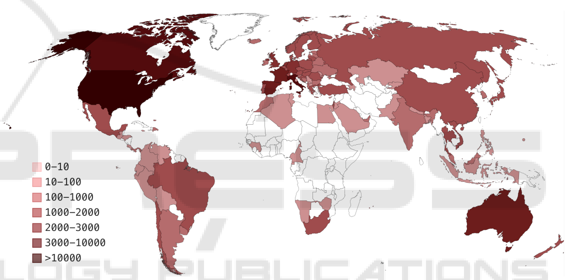


Figure 8: Geo-distribution of vulnerable ICS devices.

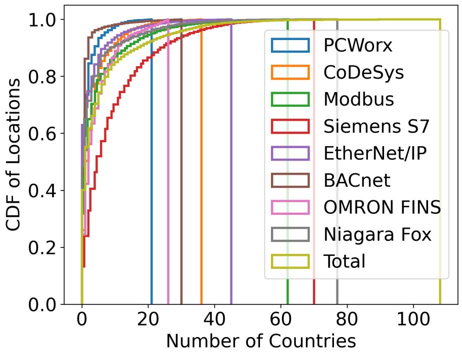


Figure 9: Location distribution of vulnerable ICS devices at country-level.

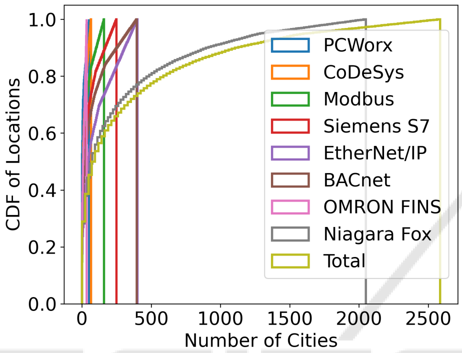


Figure 10: Location distribution of vulnerable ICS devices at city-level.

least coverage. The cities which have ICS devices using OMRON FINS protocol are only 34.

Vulnerable ICS Devices over Time. Among the three experiments over six months, we find 46,489/38,466/37,027 vulnerable ICS devices, respectively, and the details are shown in Figure 11. There is a slowly decreasing trend in the number of vulnerable ICS devices for most ICS protocols during our measurement period. This indicates that people had taken some measures to mitigate the risk of Internet-facing ICS devices, but still not enough counter-measures. However, the Siemens S7 protocol shows an anomalously increasing trend in the number and percentage of vulnerable ICS devices. This mainly due to the growing number of exposed SIMATIC S7-1200 devices.

4.6 The Statistics of ICS Vulnerabilities

We then carry on the statistical analysis of the ICS vulnerabilities mentioned above. We find that only 207 different vulnerabilities affect a total of 52,739 ICS devices. We classify these vulnerabilities according to protocols and then further analyze the impact scope, CVSSv3 score, and life cycle of vulnerabilities under different protocols.

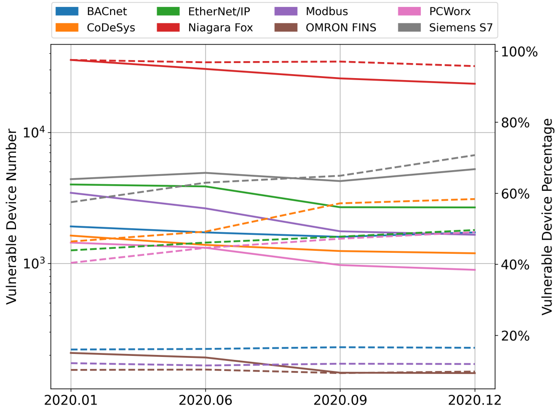


Figure 11: Vulnerable ICS devices over time. The solid lines represent the number of vulnerable ICS devices. The dash lines represent the percentage of vulnerable ICS devices.

Top 5 Vulnerabilities in Different ICS Protocols. Table 3 shows the top 5 ICS vulnerabilities in different industrial control protocols. For Niagara Fox protocol, the three vulnerabilities we found affect about 96.25% of all Niagara Fox ICS devices with complete information. There are similar characteristics in PCWorx and CoDeSys protocols. For such vulnerabilities, ICS engineers should take measures to fix or mitigate them as soon as possible.

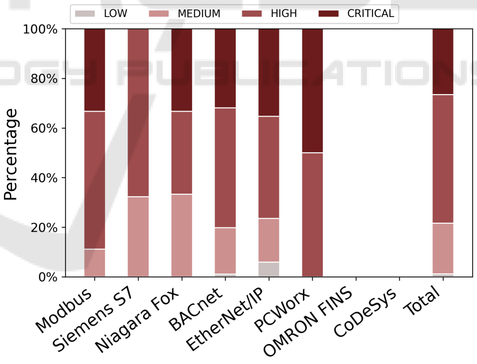


Figure 12: Severity distribution of ICS vulnerabilities in different protocols.

The Severity of ICS Vulnerabilities in Different ICS Protocols. We use the Common Vulnerability Scoring System(CVSS)(version 3) to assess the severity of ICS vulnerabilities. CVSS version 3.0 was released in June 2015, and vulnerabilities released before that date have no CVSSv3 score. 46 out of 207 ICS vulnerabilities are released before June 2015, and the ICS vulnerabilities of OMRON FINS and CoDeSys are all among them. As shown in Figure 12, the ICS vulnerabilities with high or critical severity in all ICS protocols are at least 60%. The severity of vulnerabilities in the PCWorx protocol, the most severe proto-

Table 3: Top 5 vulnerabilities in different ICS protocols.

Modbus		Siemens S7		BACnet		EtherNet/IP	
CVE	Numbers	CVE	Numbers	CVE	Numbers	CVE	Numbers
CVE-2015-7937	3095	CVE-2017-2680	4384	CVE-2019-18249	1770	CVE-2017-7898	3147
CVE-2015-6461	3077	CVE-2017-2681	4384	CVE-2016-4495	37	CVE-2017-7899	3147
CVE-2015-6462	3077	CVE-2017-12741	4337	CVE-2016-3155	36	CVE-2017-7901	3147
CVE-2018-7241	2462	CVE-2019-10936	4246	CVE-2018-7779	27	CVE-2017-7902	3147
CVE-2018-7242	2462	CVE-2018-13815	3021	CVE-2018-7795	25	CVE-2017-7903	3147
Total vul num: 22		Total vul num: 56		Total vul num: 93		Total vul num: 29	

Niagara Fox		PCWorx		OMRON FINS		CoDeSys	
CVE	Numbers	CVE	Numbers	CVE	Numbers	CVE	Numbers
CVE-2017-16744	35669	CVE-2019-9201	1444	CVE-2015-0987	208	CVE-2014-0760	1633
CVE-2017-16748	35669	CVE-2019-10953	605	CVE-2015-1015	195	CVE-2014-0769	1633
CVE-2018-18985	35669	-	-	-	-	-	-
Total vul num: 3		Total vul num: 2		Total vul num: 2		Total vul num: 2	

**Numbers:* refers to the number of devices affected by the vulnerabilities.

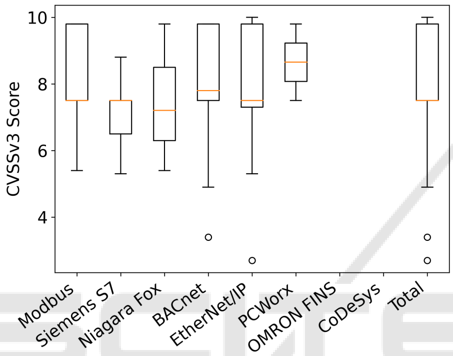


Figure 13: Average score and outliers of ICS vulnerabilities in different protocols.

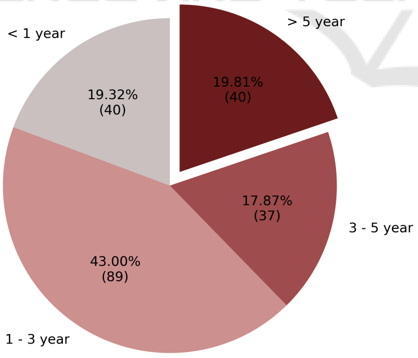


Figure 14: Vulnerabilities Existing Time.

col, is high or critical. Figure 13 shows the CVSSv3 scores of vulnerabilities in different protocols. The average scores revolve around 7.5, and the scores of most vulnerabilities are spread across the 7.5-10 spectrum. Among the ICS vulnerabilities, 69.23% are remotely exploitable, 74.52% do not require any special conditions, and 66.35% could be executed without authorization.

The Timeline of ICS Vulnerabilities in Different ICS Protocols. Figure 14 shows the disclosure time of the vulnerabilities mentioned above. We can find that most of the vulnerabilities are disclosed within 3 years, yet 19.80% have been disclosed for more than 5 years. Among these vulnerabilities, we find that there are 40 vulnerabilities (about 19.32%) which exploits are published on the Internet. This indicates that these corresponding affected devices are in a highly dangerous state.

4.7 The ICS Devices Affected by 0day Vulnerabilities

ICScope also supports the evaluation of the impact of 0day vulnerabilities. We use ICScope to evaluate the impact of a 0day vulnerability we found in Schneider Electric device. This vulnerability affects four different products, including Modicon M580 (version ≤ 3.10). We find that 3,999 Schneider Electric devices exposed to the Internet are affected by this vulnerability. We have reported this vulnerability to the vendor and waiting for the vendor to fix it.

4.8 Comparison with Shodan Vulnerability Detection

Shodan also tries to verify whether the target device is affected by known vulnerabilities. We use this function of Shodan to detect the vulnerability of the ICS devices that the complete information can be extracted. For 106,382 different IP addresses, Shodan finds only 3,472 (about 3.26%) associated with one or more vulnerabilities. Moreover, the vulnerabilities detected by Shodan are all Web-based vulnerabilities instead of ICS vulnerabilities. Comparing with Shodan, ICScope finds that 52,739 Internet-facing ICS devices (about 49.58%) are affected by at least one ICS vulnerabilities. As for ICS vulnerability detection, ICScope performs better than Shodan.

5 RELATED WORK

Passive Vulnerability Assessment in Cyberspace. Passive vulnerability assessment is an eco-friendly and resource-conserving research method for large scale security measurement towards Cyber-Space. B Genge et al. are the first to propose the Shodan-based passive vulnerability assessment tool, ShoVAT (Genge and Enăchescu, 2016). The CPE based vulnerability matching technique used by ShoVAT is heavily dependent on the quality of its CPE dictionary. Jamie O'Hare et al (O'Hare et al., 2019) improved the previous work and proposed Scout to perform a passive vulnerability assessment of HTTP services on the Internet. Compared with ShoVAT, Scout focuses on the accuracy of vulnerability matching. By comparing with the scanning result of OpenVAS, Scout's results are more practical. However, neither of the works mentioned above is suitable for the industrial control scenario. Their matching techniques depends on the quality of the banners returned by search engines. However, it is common that the banners are incomplete in ICS. We mitigate this problem with our data enrichment module. Moreover, previous works on passive vulnerability assessment rarely consider the honeypot problem. ICScope try to address it to increase the accuracy of vulnerability detection.

Online ICS Device Discovery. Previous works on the discovery of online ICS devices are always a hot topic on ICS security. Mirian et al. (Mirian et al., 2016) extend ZMap (Durumeric et al., 2013) to support four ICS protocols and scan the public IPv4 address space to discover online ICS devices. Differently, Feng et al. (Feng et al., 2016) investigates 17 industrial control protocols and develop their ICS device discovery system to conduct ICS device discovery. Although both of them have tried to minimize the number of probing packets, their systems still need to interact with target devices. By using search engines, ICScope can discover online ICS devices without any communication interactions. By using multiple search engines, ICScope can better tolerate the limitation of scanning nodes.

ICS Honeypot Detection. The honeypots in ICS security are often used to perform ICS threat analysis (Mirian et al., 2016; Vasilomanolakis et al., 2016; Serbanescu et al., 2015). There are few works considering ICS honeypot detection. Feng et al. (Feng et al., 2016) propose a learning model to determine the probability of an ICS honeypot and a heuristic algorithm to verify it with the least number of packets. However, both the learning model and the algorithm are based on the probing packets. The ICS honeypot detection approach used in ICScope is based on the non-interaction features. So it has no probing packet with ICS devices.

6 DISCUSSION

Ethical Considerations. From an ethical perspective, we perform our measurements in a completely passive mode to avoid any potential damage. The data used for ICScope contains no sensitive data, and the data provided by public search engines are publicly available. However, because the vulnerable ICS devices are still unfixed, we cannot publish our datasets and over 52K vulnerable ICS devices directly. Instead, we report vulnerable ICS devices in our country to relevant CERT, and we also try to contact several vendors.

Limitations. First, the above experiment results only show a subset of the security status of Internet-facing ICS devices. The actual situation should be more serious. This is a limitation because it is difficult or impossible to obtain all Internet-facing ICS devices via search engines, and not all ICS devices can be extracted full information. Second, while we obtain vulnerable Internet-facing ICS devices, we still do not confirm whether there is any defensive measure on them. Because ICScope is based on version to identify vulnerabilities rather than proof of concept(PoC). Third, the supporting ICS protocols of ICScope is limited to the ability of search engines. ICScope supports one ICS protocols only if its banners contain key fields. Fourth, a lack of raw probing packets makes it harder to extract device information or detect ICS honeypots because the banners generated by search engines might lose some information.

Honeypot Detection. It is hard to detect ICS honeypots accurately in passive mode. For example, ICS devices might share one IP by port forwarding. However, this is also a fingerprint to detect honeypots. We cannot distinguish them without the help of probing packets. So we regard all of them as ICS honeypots to reduce false positives. Fortunately, we find that this problem might be corrected by the data from non-ICS port, and we leave it as our future work.

Mitigation. We need to take some defensive measures to mitigate the risk of Internet-facing ICS devices. The intuitive measure is to keep the latest version of each ICS device. However, this is difficult or impossible in the actual ICS environment. Instead, we can take the following defensive measures:

Add Firewall Policy. By adding firewall to restrict source IP and destination port, we can prevent the attack from accessing the vulnerable ICS devices.

Network Isolation. Isolate the ICS devices that need

Internet connection from the ones that do not need to prevent the attacker from accessing the vulnerable devices.

7 CONCLUSION

As a crucial component of modern city infrastructure, the industrial control system exists in every corner of our life. However, the Internet-facing ICS devices are under the risk of known vulnerabilities. In this work, we develop ICScope to discover the vulnerable Internet-facing ICS devices. Base on the results, we perform a comprehensive analysis of the security status for online ICS devices. We find that 49.58% of Internet-facing ICS devices that we can extract complete device information are affected by known vulnerabilities. The most serious ICS protocol is Niagara Fox, which proportion of vulnerable devices is even as high as 96.25%. We observe that most of the vulnerable devices are affected by the same vulnerability in Niagara Fox, PCWorx, and CoDeSys protocols. In all ICS protocols, at least 60% of the ICS vulnerabilities are with high or critical level severity. We also observe a slowly decreasing trend in the number of vulnerable ICS devices during our six-month measurement period. Moreover, our measurement results only present the lower limit of the actual situation. In response to these severe industrial control security issues, we also discuss the mitigation measures, such as add firewall policy.

ACKNOWLEDGEMENTS

We would like to thank all anonymous reviewers for their valuable feedback that greatly helped us improve this paper. Besides, we would like to thank Yuxiang Lu, Zhenbang Ma, Yu Wang, for their helping in our work.

REFERENCES

Comer, D. E. and Lin, J. C. (1994). Probing tcp implementations. In *Usenix Summer*, pages 245–255.

Di Pinto, A. A., Dragoni, Y., and Carcano, A. (2018). Triton: The first ics cyber attack on safety instrument systems. In *Proc. Black Hat USA*, pages 1–26.

Dong, Y., Guo, W., Chen, Y., Xing, X., Zhang, Y., and Wang, G. (2019). Towards the detection of inconsistencies in public security vulnerability reports. In *28th {USENIX} Security Symposium ({USENIX} Security 19)*, pages 869–885.

Durumeric, Z., Adrian, D., Mirian, A., Bailey, M., and Halderman, J. A. (2015). A search engine backed by internet-wide scanning. In *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*, pages 542–553. ACM.

Durumeric, Z., Wustrow, E., and Halderman, J. A. (2013). Zmap: Fast internet-wide scanning and its security applications. In *22nd {USENIX} Security Symposium ({USENIX} Security 13)*, pages 605–620.

Fachkha, C., Bou-Harb, E., Keliris, A., Memon, N. D., and Ahamad, M. (2017). Internet-scale probing of cps: Inference, characterization and orchestration analysis. In *NDSS*.

Feng, X., Li, Q., Wang, H., and Sun, L. (2016). Characterizing industrial control system devices on the internet. In *2016 IEEE 24th International Conference on Network Protocols (ICNP)*, pages 1–10. IEEE.

Genge, B. and Enăchescu, C. (2016). Shovat: Shodan-based vulnerability assessment tool for internet-facing services. *Security and communication networks*, 9(15):2696–2714.

Guo, G., Zhuge, J., Yang, M., Zhou, G., and Wu, Y. (2018). A survey of industrial control system devices on the internet. In *2018 International Conference on Internet of Things, Embedded Systems and Communications (IINTEC)*, pages 197–202. IEEE.

Leverett, É. and Wightman, R. (2013). Vulnerability inheritance programmable logic controllers. In *Proceedings of the Second International Symposium on Research in Grey-Hat Hacking*.

Mirian, A., Ma, Z., Adrian, D., Tischer, M., Chuenchujit, T., Yardley, T., Berthier, R., Mason, J., Durumeric, Z., Halderman, J. A., et al. (2016). An internet-wide view of ics devices. In *2016 14th Annual Conference on Privacy, Security and Trust (PST)*, pages 96–103. IEEE.

Moon, B., Jagadish, H. V., Faloutsos, C., and Saltz, J. H. (2001). Analysis of the clustering properties of the hilbert space-filling curve. *IEEE Transactions on knowledge and data engineering*, 13(1):124–141.

O'Hare, J., Macfarlane, R., and Lo, O. (2019). Identifying vulnerabilities using internet-wide scanning data. In *2019 IEEE 12th International Conference on Global Security, Safety and Sustainability (ICGS3)*, pages 1–10. IEEE.

Samtani, S., Yu, S., Zhu, H., Patton, M., and Chen, H. (2016). Identifying scada vulnerabilities using passive and active vulnerability assessment techniques. In *2016 IEEE Conference on Intelligence and Security Informatics (ISI)*, pages 25–30. IEEE.

Serbanescu, A. V., Obermeier, S., and Yu, D.-Y. (2015). Ics threat analysis using a large-scale honeynet. In *3rd International Symposium for ICS & SCADA Cyber Security Research 2015 (ICS-CSR 2015) 3*, pages 20–30.

Vasilomanolakis, E., Srinivasa, S., Cordero, C. G., and Mühlhäuser, M. (2016). Multi-stage attack detection and signature generation with ics honeypots. In *NOMS 2016-2016 IEEE/IFIP Network Operations and Management Symposium*, pages 1227–1232. IEEE.