IoT Vulnerability Scanning: A State of the Art

Ahmed Amro

Norwegian University of Science and Technology, Gjøvik, Norway
ahmed.amro@ntnu.no

Abstract. Our modern life becomes more and more dependent on technology and services provided through an increasing number of deployed devices "Things" which are connected over networks that can sometimes be accessed remotely via the Internet. Although this Internet of Things (IoT) has led to innovations and improvements to our way of life, it has created many issues, especially related to cybersecurity. Ensuring the security of the IoT ecosystem can be achieved using pro-active security processes, including vulnerability scanning. In this paper, we capture the state of the art of the process that is IoT vulnerability scanning to determine its popularity and maturity. We have captured the different motivations for vulnerability scanning, the scanning space, process, and faced challenges. A Systematic Literature Review (SLR) has been conducted to achieve this goal, and the results are presented hereof. Moreover, we conducted a group of experiments to assess the status of IoT services and their associated vulnerabilities in the Nordic countries and found that additional work is needed to improve the security of the IoT ecosystem.

Keywords: IoT · IIoT · Vulnerability Scanning · Shodan

1 Introduction

Innovations are being witnessed every day that affect every aspect of our daily lives. You can start your day by waking up to a smart alarm that knows your schedule, use a smart toaster to perfect your breakfast, go to your work where you are surrounded with connected devices, printers that know what documents you print, video cameras capturing your movement, sensors, and controllers that control critical industrial operations and so on. These innovations led to the creation of the paradigm known as the Internet of Things (IoT) and its emerging sub-domain the Industrial IoT (IIoT).

The number of connected devices to the Internet is increasing at an incredible rate. According to statistics published in May 2019, 22 Billion devices were connected in 2018, expected to reach 50 Billion by 2030 [4]. The advantages of IoT devices are clear to many. They improve the quality of life at home by providing features such as entertainment, smart monitoring, and security. They also facilitate daily business operations, such as printing, perimeter monitoring, etc. IoT has found its way also to industrial facilities to improve the overall production process using IIoT. These advantages are related to the nature of IoT

devices which can be characterized by connectivity to networks and the Internet, cheap, simple to install, and many others. These characteristics aided in the widespread and increased adoption of this technology. However, the advantages to the users and market investors are what make IoT security worrisome due to their susceptibility to cyber threats.

An enormous amount of resources exists discussing attacks against the IoT ecosystem. Types of attacks differ based on the characteristics of the targeted IoT devices. Some attacks target short-range communication technology such as Bluetooth which requires proximity to targeted devices [28]. Others target networked devices that are connected to home, industrial, corporate, or university networks while others target devices that are connected to the Internet. To mention a few, in 2019, an attack was discovered that makes two million IoT devices discoverable and susceptible to hijacking with no solution at the time of writing the article [20]. Another attack targeted internet connectivity in a specific geographical area through carrying a Distributed Denial of Service (DDoS) attack against 900,000 routers, blocking their owners from online access [23].

Due to the connectivity nature of IoT, it makes compromised devices a threat not only to the functionality of the devices themselves, to their owners or their operating environment but the threat can extend to any Internet-connected device, its owner, and its environment. This was the case for the Mirai botnet, where a group of compromised IoT devices all over the globe was leveraged to launch attacks against hosts in other geographical areas including disrupting DNS providers affecting groups of web servers which consequently affected millions of users [2]. What makes IoT devices a preferable target to attackers; other than their connectivity nature, is the broad availability of vulnerabilities [18]. Moreover, other characteristics regarding IoT devices such as being cheap and simple to install complicates the integration of proper security functions into these devices, which in turn deepens the hole that is called IoT security.

A major factor in improving the posture of the IoT security domain is the proposal of recent IoT standardized communication protocols that integrates security features or suggest guidelines for the manufacturers and users. Nevertheless, the adoption of such standards is not optimal due to implementation flaws [17]. Moreover, the fast deployment (quick-to-market) of IoT due to their cost, simplicity, and functionality makes security assessment a limitation to the market [17]. Thus comes the need to capture the security status of already connected devices, even with their adoption of new security-enhanced standards and after-deployment security assessment. A growing direction to bridge the gap between the rapid deployment of IoT devices and improving the security posture of the Internet and private networks against them is by performing proactive security assessment, including IoT vulnerability scanning. Security researchers are challenged and generously rewarded for hacking an IoT operating system called Azure Sphere to improve its security [26]. Moreover, Bureau Veritas, a leading certification company, has recently targeted the certification of IoT devices by performing security assessments using IoT vulnerability scanning techniques to improve the market value of their customers' products [17]. In addition to

that, proactive vulnerability scanning has been utilized to improve the security posture of the TLS certificate ecosystem [25].

This paper aims to capture the state-of-the-art of IoT vulnerability scanning in the literature to comprehend the popularity and maturity of such an approach in improving the security posture in the IoT domain. For this sake, a Systematic Literature Review (SLR) has been conducted, and its results are presented in this paper. The SLR methodology applies the guidelines proposed by Okoli and Schabram (2010) [22], which is the most relevant to the domain of this study. Three digital libraries were searched for articles containing the phrases "IoT" AND scan AND "vulnerability" in their metadata section, the libraries are Scopus, IEEE, and Science Direct. In total, 25 unique works were found relevant to the scope of this paper in the initial selection phase. After a deeper study of the results, two major work categories were discovered, five works focused on IoT device scanning without a focus on vulnerability scanning, and the remaining 20 focused mainly on vulnerability scanning which is relevant to the scope of this paper. Moreover, due to time limitations, only half of the remaining works were thoroughly studied based on prioritization criteria favoring recent works with higher research impact measured by their citation count. Overall, the freshness of the research filed is indicated by the low amount of literature, but, the growing interest in it is clearly witnessed and can be seen in Figure 1. The contributions of this paper are summarized below:

1. The promising research direction of IoT vulnerability scanning is highlighted by the results of the conducted SLR.
2. A vulnerability scanning process is proposed based on the observed literature.
3. A vulnerability scanning space is proposed which is useful for visualizing the different scanning processes and can be used as basis for measuring expected time and complexity requirements.
4. The status of the most relevant vulnerabilities in the Nordic countries are assessed which is useful to shape cyber security solutions that are more relevant to the market as well as direct research directions.

2 Vulnerability Scanning: State-of-the-Art

2.1 Scanning goals

Among the studied literature, the main observed goal for performing vulnerability scanning is to investigate security and privacy issues with some works aiming to enforce security rules [21]. Secondary goals are related to developing security solutions for IoT and IIoT [3,9,10,17,21], certification of IoT and IIoT devices to improve their market value [17], while others aim to provide platform for threat information sharing in IoT [10].

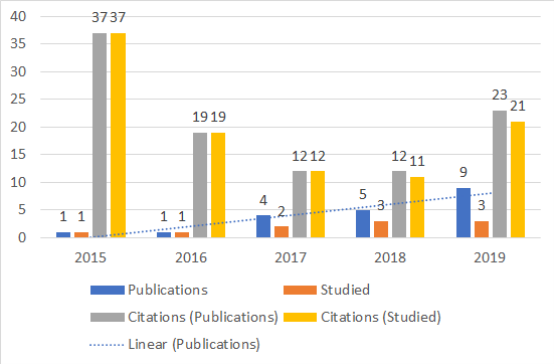


Fig. 1: Number of publications in the field of IoT vulnerability scanning, the number of studied works in this paper and the citation count of both categories

2.2 Scanning Space

From the studied scanning processes, we were able to identify an IoT vulnerability scanning space in which all the observed scanning processes reside. The scanning space as shown in Figure 2 consists of three dimensions, the x-axis represents the IPv4 address space reflecting the scanned hosts, the y-axis represents the port numbers reflecting scanned services, and the z-axis represents the scanned vulnerabilities. We also captured the effect on the scanning process from time and complexity perspectives when attempting to cover more areas in each axis. It was observed that scanning more hosts and more ports are relatively simple but time-consuming. For instance, scanning 3.702 Billion IP addresses consumes on average 1 hour and 8 minutes for each protocol in a specific port with a limited time difference between the different protocols [10]. On the other hand, detecting vulnerabilities requires additional processing and more complicated logic. The Figure also reflects the most observed scanned ports and IoT vulnerabilities.

IPv4 addresses (x-axis) : Regarding Internet Protocols (IP), only IPv4 scanning has been observed. To the best of our knowledge, no work has yet accomplished a full scan for IPv6 addresses due to its large space. Nevertheless, *Shodan* [16], an IoT search engine (more in Section 2.4) collects IPv6 addresses during the IPv4-based scanning [12]. The coverage of IPv4 scanning differs from work to work. Table 1 depicts the different types of networks targeted for coverage in the studied literature. Note that some works performed or discussed several scanning processes with different network types; therefore, they appear in multiple categories. Some works have scanned the entire IPv4 in search of specif vulnerabilities in specific ports, others scanned small and home networks, others scanned large networks while others focused on a country level.

Another aspect that has been discussed regarding IPv4 scanning is related to address randomization. Some works proposed algorithms to generate random

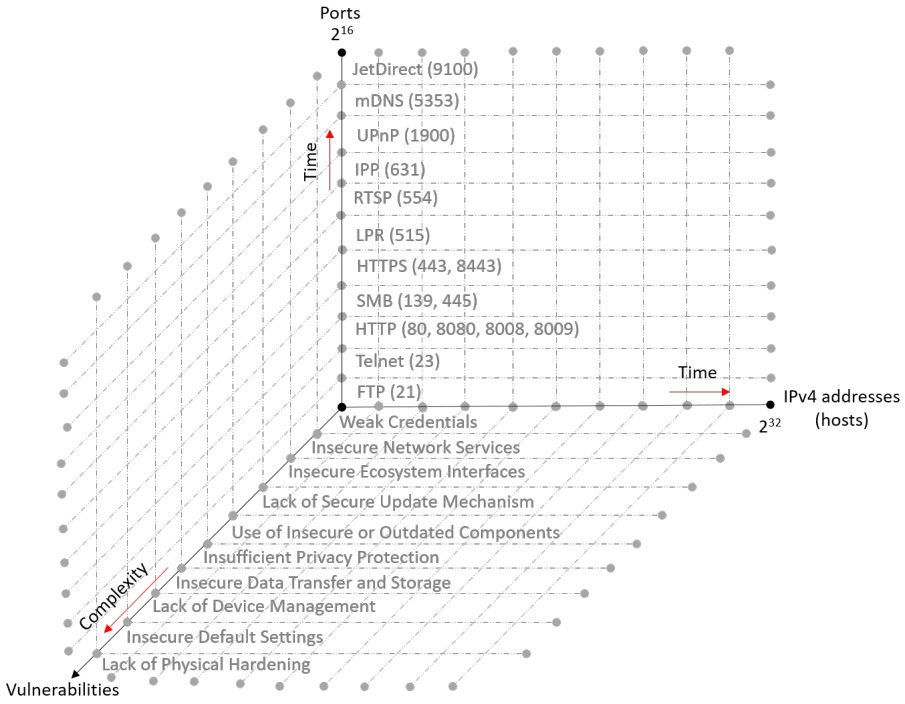


Fig. 2: IoT Vulnerability Scanning space

IPv4 addresses for the scanning process in an attempt to avoid the detection and scanning prevention by security solutions such as firewalls which can easily detect sequential IPv4 scanning [9].

Table 1: Observed Network types in the literature

Internet-wide			
[9, 10, 25]			
Country			
[1]			
Testing Environment	Local Active Network	Industrial Network	SDN Network
[3, 21, 27]	[1, 3, 11, 17, 25]	[1, 3, 17, 25]	[21]

Port numbers (y-axis) : Scanning processes differ in their coverage of ports. Most works scan the entire port numbers looking for open ports to perform banner grabbing to fingerprint device type and infer additional details to be utilized in further vulnerability scanning and analysis. However, some works only

target specific vulnerabilities associated with a specific protocol, thus covering only a subset of the port number space. Kumar et al [11] analyzed scanning data collected from 83 million IoT devices in 16 million homes aiming to reflect the current status of the IoT domain. In their work, they were able to identify the most popular open IoT services and their ports (y-axis in Figure 2).

Vulnerabilities (z-axis) : As mentioned before, the number of discovered IoT vulnerabilities is increasing. In 2018, the Open Web Application Security Project (OWASP) published the top 10 IoT vulnerability categories (z-axis in Figure 2) as part of a dedicated project targeting IoT security [18]. Ogunnaike and Lagesse [21] proposed that systematic vulnerability scanning should be according to the OWASP IoT vulnerability category, and we agree with this notion.

2.3 Scanning Challenges

There are several challenges associated with IoT vulnerability scanning related to the device type identification, visibility, and management of legacy devices as well as some ethical aspects that should be considered but would limit the scanning results.

Device type and Operating system identification : As mentioned before, the amount of IoT devices is immense; their types and operating systems as well are increasing with innovations every day. Only in the home environment, 14 categories of devices have been observed based on their functionality (network node, mobile device, work appliance, game consoles, etc.) produced by 14,3 thousand manufacturers [11]. Machine learning techniques have been applied to improve the identification of device types [11] and operating systems [9]. Communicating with these devices to scan them and identifying their associated vulnerabilities require varying levels of scrutiny, especially since most of them do not adhere to certain standards.

Legacy devices Industrial environments rely on a wide range of devices, and some of them are relatively old. Such devices mostly apply proprietary or legacy software with out-of-business providers, and some of these devices cannot be discovered using traditional scanners [3]. Even scanners that are tweaked to discover such devices do not usually account for many devices due to the large variety of them [3]. Therefore, scanning such devices constitutes a great challenge that has been addressed in the literature by several works [1, 3, 17, 25].

Ethical Considerations Vulnerability scanning could reveal information that might be utilized during malicious activities such as revealing personal information or harming the reputation of some companies. Therefore, some works have addressed this issue and argued that this might have affected the value and validity of the results [1, 11]. The validity can be affected when only passive

scanning of previously available scanning data has been performed without active scanning to validate, leading to uncertainty regarding the current status of the scanned devices. On the other hand, the value of the scan can be reduced when some users request that their network should not be scanned or choose not to share the scanning results which could lead to reduced data collection.

Some works claimed that they had acquired permission before scanning the network [14]. Others provided home users with clarified request to approve the collection of user-triggered vulnerability scanning data [11]. Other works claimed that they only queried (Passive scanning) *Shodan* and *Censys* without performing any active scanning [1, 25]. The rest of the studied works either used their equipment in their networks or did not mention ethical considerations.

2.4 Scanning Process

After studying the different scanning methods in the different works, an overview of the observed steps in the scanning processes has been identified and presented in Figure 3.

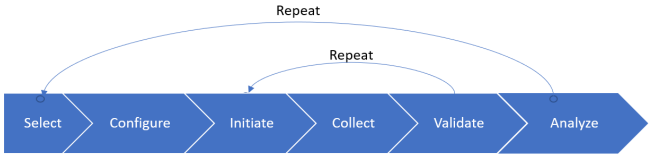


Fig. 3: Overview of IoT device and vulnerability scanning process

A brief description of each step is mentioned below:

– **Select**: The first step is determining what tool or platform to utilize for scanning. The most reference tools are *Shodan* [16], *Censys* [5], *Masscan* [6], and *Nmap* [13]. *Shodan* and *Censys* are both online search engines that perform periodic scanning of the IPv4 address space, store results, index them, and make them available for searching. Both platforms can be used for free or with a subscription for advanced functionalities such as on-demand scanning. Other tools that can be selected that are not necessarily vulnerability scanners are network traffic capturing tools such as *Wireshark* [24], which captures network traffics and stores them for later analysis. We categorize the scanning process that utilizes such platforms as a "Passive Scanning" process. *Masscan* and *Nmap*, on the other hand, are open-source tools used mainly for active port scanning and relatively limited vulnerability scanning capabilities. We categorize the scanning process that utilizes such tools as an "Active Scanning" process. A simplified scanning model that combines active and passive scanning is shown in Figure 4, this model is followed by *Censys* and *Shodan*. Furthermore, some works have proposed their own platforms

for Active and/or Passive vulnerability scanning with improvements over available tools in aspects such as IP address randomization, OS fingerprinting, device type identification, and advanced vulnerability identification and management.

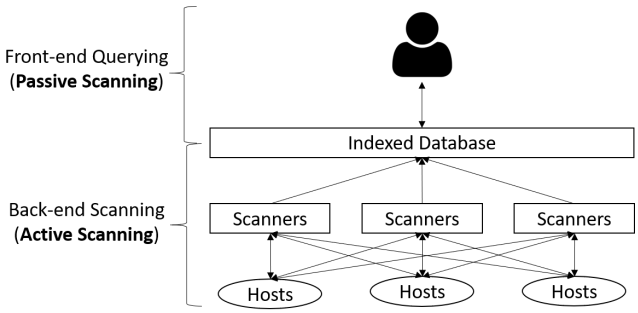


Fig. 4: Simplified Scanning Model combining Active and Passive Scanning

- **Configure**: The determination of the scanning scope by tuning the parameters for active scanning directly influences the scanning time and is dependent on the goal of the scanning process. Time for passive scanning, on the other hand, is not influenced by the configuration but the configuration influence the amount of returned results. The configurable parameters are mainly related to the three dimensions captured in the scanning space(Section 2.2). Figure 5 shows a visualization of different types of scanning determined by different configurations and how they would look in the defined scanning space. The increased area suggests increased time or/and complexity. Some scanning processes target the discovery of specific vulnerability (e.g. Heartbleed) in the entire IPv4 space, others aim to identify all vulnerabilities in a home, corporate or industrial network, and many other scanning processes have been observed.
- **Initiate**: There is a difference between the initiation of the scanning process (active scanning) and the initiation of the search process (passive scanning). This difference should be considered in evaluating the freshness of the identified vulnerabilities. For instance, an active scan could uncover a vulnerability and record it at a certain time. After a while, when this vulnerability is searched and found, it will not necessarily mean that the vulnerability still exists, maybe it was fixed during the time difference between the active scan and the passive scan. Another aspect has been identified regarding the initiation step; some tools initiate the scanning process automatically and periodically such as *Shodan* and *Censys* in the back-end while others require a human to initiate the scanning process by invoking the selected tools, such as Wi-Fi Inspector [11].
- **Collect**: Different tools collect different types of data. Some works grab service banners (e.g. FTP) when establishing a connection with devices. Oth-

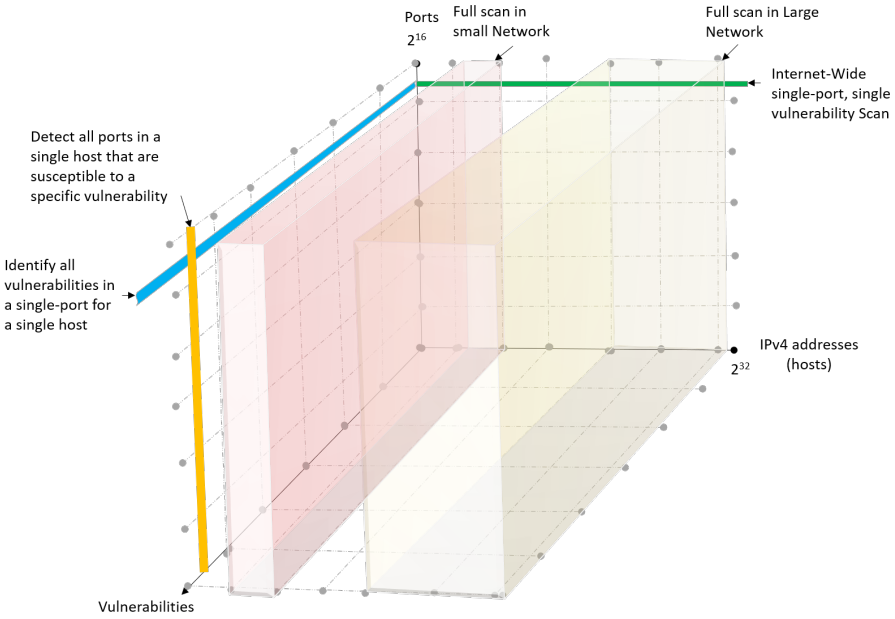


Fig. 5: Visualizing examples of scanning processes in the scanning space

ers collect protocol headers and responses (e.g. HTTP) while others utilize crafted requests to trigger informative responses. On the other hand, some relay on capturing the communicated packets and massages. *Shodan* and *Censys* both perform a group of collection methods including banner grabbing and capturing of protocol headers. Then they index and host the results of the back-end scanning in an online database available for querying. On the other hand, other tools, especially the ones that perform active scanning, return the results within the tool itself (Command Line Interface (CLI) or graphical) or save it into a file or database for later analysis. Some works proposed the application of the Structured Threat Information Expression (STIX) [8] as a format for saving the discovered vulnerabilities which can be useful for threat information sharing [10].

– **Validate**: Many aspects could influence the validity of the search results. Other than the difference between scanning times mentioned before (in the initiation step), the algorithm for vulnerability identification could be based upon high-level conditions and parameters, such as open port, or protocol header value without further verification whether the vulnerability is actually exploitable. For instance, Al-Alami et al [1] scanned hosts on *Shodan* with default credentials based on the presence of FTP response code 230 which means successful login; *Shodan* suggests this after attempting to log in using a list of most common credentials. Knowing that *Shodan* scans the entire IPv4 at least once each month [9], the assumed vulnerability could have been

resolved but still appears in the results, therefore should be validated using active scanning, bearing in mind the associated ethical considerations.
Some processes utilize additional tools to validate the results of the scanning tools. For instance, the results of *Shodan* discovered open ports have been validated using *Nmap* [14]. Other processes perform in-tool validation as part of the scanning process by invoking certain modules able to communicate with the target devices and actively validate the discovered vulnerabilities [3, 11, 17, 21].

- **Analyze**: The amount and format of the results can be overwhelming. Therefore, post-processing and analysis are usually where most of the work is required. Additional tools are usually utilized for additional analysis such as using *binwalk* [7] for analysis of the identified firmware looking for vulnerabilities. Moreover, some works targeted the assessment of the identified vulnerabilities through the analysis of related information and metrics such as Common Vulnerabilities and Exposures (CVE), the Common Vulnerability Scoring System (CVSS) and others. Overall, this step should determine if the scanning process has accomplished the goal it was intended for. The observed targets for analysis include TLS certificates, weak cryptographic algorithms (hashing, encryption, and digital signatures), open ports, CVE's associated with discovered device type or operating systems, and their CVSS scores, devices' firmware, weak credentials, clear images, and video and many others.
- **Repeat**: IoT vulnerability scanning is goal-oriented, utilizing the available tools, techniques, and information to reach a conclusion. Usually, the process is iterative either entirely or partially. For instance, some works use the same selected tools, same configurations but initiate the process at different times to capture the difference in the state of certain vulnerabilities over a period of time, such as capturing the security state of the TLS ecosystem by performing the same scan process twice over three years [25]. Other works have conducted multiple scanning processes using different tools, configurations, analysis, etc, in order to detect different vulnerabilities in different devices.

3 Nordic IoT and IIoT Telescope: Empirical Study

In this section, we present our conducted experiments to capture the connectivity status of IoT and IIoT devices and some of their associated vulnerabilities in the Nordic countries. Due to the location of NTNU in Norway and considering the cultural, economic, and industrial ties between Norway and its neighboring Nordic countries, we decided to focus our study on them. The Nordic countries are considered globally influential and residing in a stable geographical region consisting of Norway, Sweden, Denmark, Finland, and Iceland [19]. Two experiments were conducted, the first experiment aimed to capture the discoverability of devices listening to the most common observed ports. The second experiment aimed to uncover the status of certain vulnerabilities in the discovered devices. Such experiments shed some light over IoT connectivity in the Nordic region to

direct the market toward more relevant solutions as well as focus research directions toward the most relevant protocols. Moreover, similar experiments can be conducted as a source of threat intelligence.

Both experiments followed the scanning process presented in Section 2.4. We **selected** Shodan as our scan tool to avoid any legal and ethical issues associated with active scanning. Then, we **configured** the search parameters for each performed experiment specifying the elements of the proposed scanning space. The IP addresses were specified by choosing the country, ports chosen by specifying the port numbers, and the vulnerabilities specified by their CVE or signature strings. We **initiated** the scans using Shodan's CLI tool in Linux and **Collected** the results by saving them into files. We **validated** the scan results by repeating the scanning several times and documenting the latest results. Finally, we **analyzed** the collected results and presented our analysis in this paper.

3.1 Nordic Connectivity

In this experiment, we captured the status of IoT and IIoT connectivity in the Nordic countries. We utilized Shodan to query the number of discovered connections in the most common IoT services and their observed ports, namely, JetDirect (9100), mDNS (5353), UPnP (1900), IPP (631), RTSP (554), LPR (515), HTTPS (443,8443), SMB (139 and 445), HTTP (80, 8080, 8008 and 8009), Telnet (23) and FTP (21). In addition to the most common IIoT services, namely, Modbus (502), DNP3 (20000), and RPC (135 and 102). Firstly, we aimed to assess the Nordic connectivity with these services on a global level. It can be observed from Figure 6 that Sweden has relatively higher connections as the number of connections in these ports constitutes 0,53% of the global connections and ranking as the 25th globally.

Secondly, we aimed to assess the connectivity for each of these services. The highest numbers of connections were associated with HTTP and HTTPS services constituting 88,96% (51,29% and 37,57% for HTTP and HTTPS respectively) of the total connections among the services under analysis. It can be observed from Figure 7 that the less secure service that is HTTP is more dominantly implemented across the Nordic countries with a relative balance in Denmark between HTTP and HTTPS.

Then, the connectivity status of the remaining IoT services is captured in Figure 8. It can be observed that the FTP service is the most deployed across all countries except Norway in which telnet is the most common.

Finally, we aimed to assess the connectivity status of most common IIoT services. As shown in Figure 9, Distributed Network Protocol 3 (DNP3) protocol is the most common across all countries except Sweden, where Modbus protocol stands higher. In fact, we discovered that Sweden ranks as the 7th globally in the number of Modbus connections discovered by Shodan. It is worth mentioning that the DNP3 protocol has much higher connections globally than Modbus (260888 and 15543, respectively). We argue that these numbers are alarming and reflect a high degree of connectivity for IIoT devices using protocols that are known for their security issues.

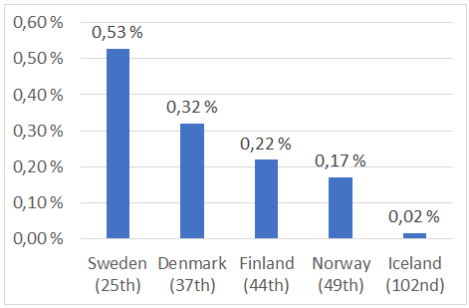


Fig. 6: The Nordic shares and rankings in the global connectivity with IoT and IIoT Services

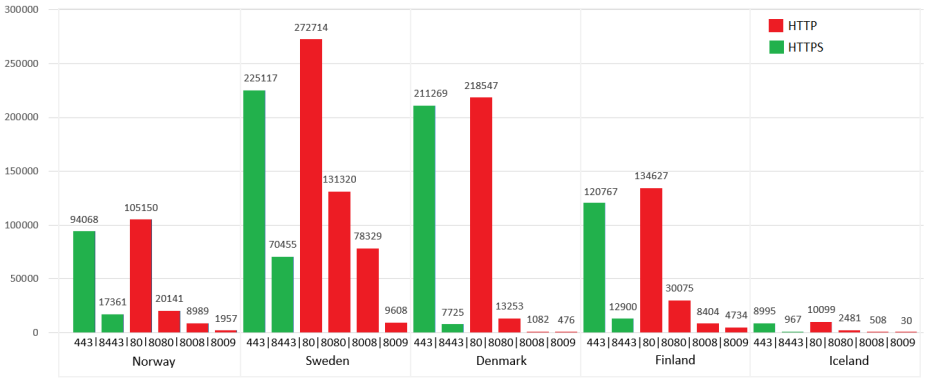


Fig. 7: The number of HTTP and HTTPS connections in the Nordic countries at different service ports

Having in mind the connectivity status presented previously, we aimed to assess the security status of the analyzed services. By utilizing the Shodan Exploits database [15] we searched for the exploits associated with these services. HTTP and HTTPS have the largest amount (2151 and 115 respectively) then, FTP (157), SMB (33), Telnet (17), and RPC (15) with many available exploits. Considering the relatively large number of FTP exploits, and the relatively large number of discovered devices running FTP (Figure 8) we argue that FTP service is the most exposed service in the Nordic region which makes it a candidate enabler for launching a wide range of cyberattacks. Similar concerns are drawn for the Telnet and SMB services.

3.2 Vulnerability Scanning

We searched Shodan for a group of vulnerabilities observed in the literature, namely, in the work of Al-Alami [1] in which the authors searched for these vulnerabilities in Jordan. The searched vulnerabilities are Heartbleed (CVE-2014-

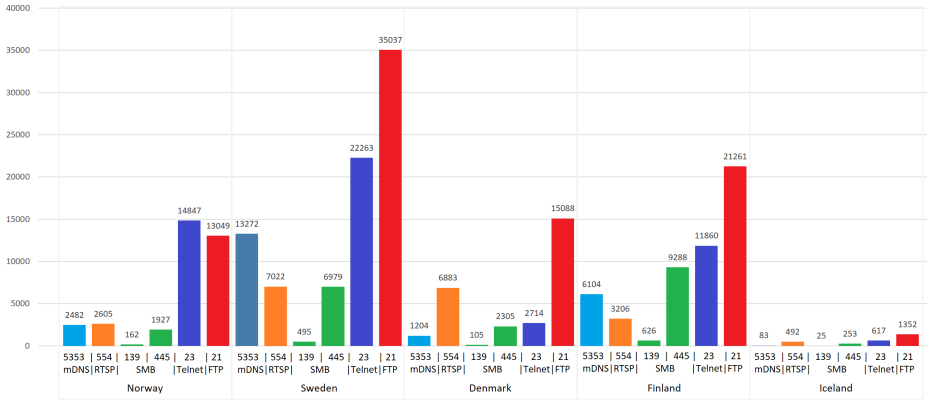


Fig. 8: Number of connections with mDNS, RTSP, SMB, Telnet and FTP services in the Nordic countries

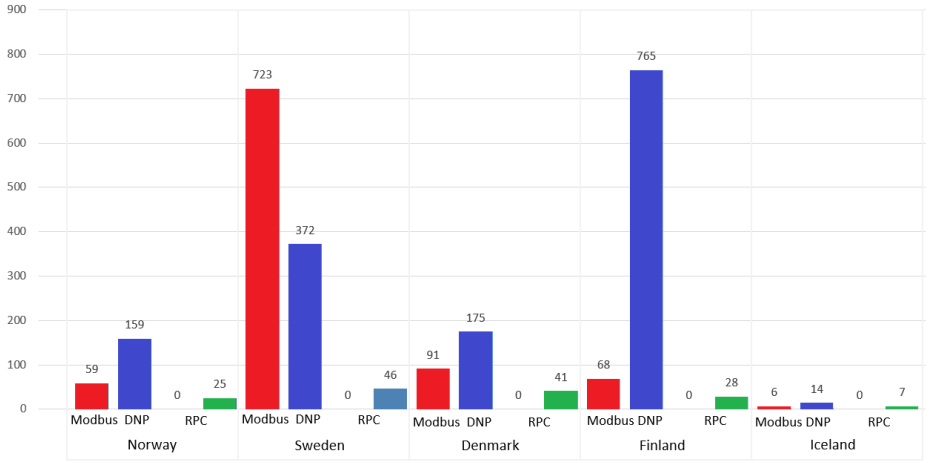
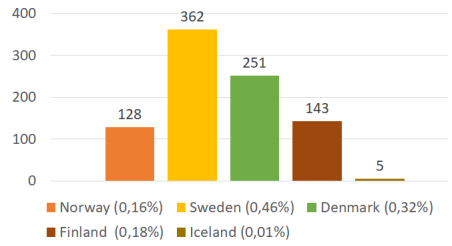


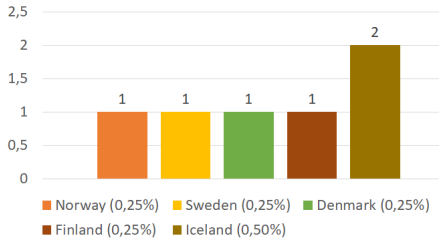
Fig. 9: Number of connections with IIoT services in the Nordic countries

0160), TicketBleed (CVE-2016-9244), SMB anonymous login, and FTP weak authentication. Heartbleed vulnerability allows for the stealing of information by exploiting the OpenSSL cryptographic library. As shown in Figure 10a Sweden is the Nordic country with the highest number of devices vulnerable to Heartbleed and hosts 0,46% of the vulnerable devices globally. TicketBleed vulnerability is similar to Hearbleed but allows for steeling less amount of information and affects proprietary TLS stack [1]. It can be observed from Figure 10b that the vulnerability affects very few devices in the Nordic countries. As a matter of fact, our search revealed that TicketBleed is less exiting globally than Hearbleed, with only 404 vulnerable devices compared to 78414 for Heartbleed. Moreover, the SMB anonymous login vulnerability allows the exposure of folders,

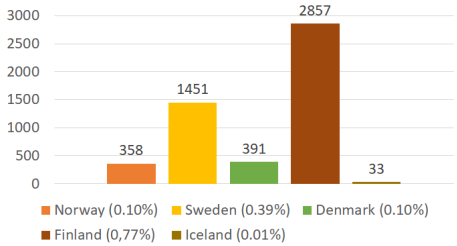
files, and printers. Figure 10c depicts the number of vulnerable devices in the Nordic countries and Finland hosting the majority of these devices. Finally, the FTP weak credentials vulnerability points to the devices with easy and guessable credentials brute-forced by Shodan. As shown in Figure 10d, Sweden is hosting the most vulnerable devices.



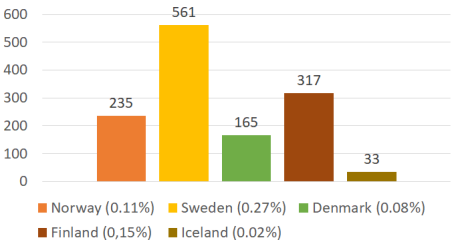
(a) Heartbleed (CVE-2014-0160)



(b) TicketBleed (CVE-2016-9244)



(c) SMB Anonymous Login



(d) FTP Weak Credentials

Fig. 10: Status of certain vulnerabilities in the Nordic Countries

4 Conclusion

In this paper, we have conducted a Systematic Literature Review (SLR) to capture the state of the art of vulnerability scanning in the Internet of things (IoT). We have observed a growing interest in the field indicated by the growing amount of literature and research impact.

The main goal for performing vulnerability scanning as observed in the literature is to investigate security and privacy issues in the connected "things". Additional goals could be related to developing security solutions, usage as a source of information for threat sharing as well as certification of IoT products. The main challenges faced during scanning for vulnerabilities are related to the large number of devices provided by different manufacturers with functionalities not adhering to certain standards.

An overview of the observed vulnerability scanning process is presented in this paper. The process consists of 6 main iterative steps, select, configure, ini-

tiate, collect, validate, and analyze. All the studied works apply in one way or another each of the identified steps in the presented process. A scanning space has also been identified in which all the observed scanning processes reside. The indicated space can be used to visualize the scanning process and assess its coverage, complexity, and time requirements.

Moreover, the availability of accessible tools to perform scanning at a varying degree of detail is observed. But, improvements are yet to be made in the aspects of supporting new and legacy device types and their operating systems as well as the discovery of new vulnerabilities which is a continuous operation in the field of IoT. Furthermore, the availability of such tools allows for different unexplored use cases to capture the state of IoT and their security in many domains which could pave the way for future research.

Finally, an empirical study was conducted in this paper, following the proposed scanning process and using the available scanning tools. The study aimed to capture the status of the connectivity and exposure of vulnerabilities in the Nordic countries. Among the observations is the relatively high exposure of IIoT protocols especially Modbus and DNP in Sweden and Finland considering their associated insecurities. In addition to that, although Heartbleed vulnerability has been around for a while, it still exists in the Nordic Countries accumulating a total of 889 vulnerable devices. Finally, the number of vulnerable devices due to insecure implementation of FTP and SMB protocols is also high (1311 and 5090 respectively). We argue that the total number of vulnerable devices considering only the scanned 4 vulnerabilities (7295) could establish a sufficient base for launching a large-scale attack such as the one originating from the Mirai botnet.

References

1. Al-Alami, H., Hadi, A., Al-Bahadili, H.: Vulnerability scanning of iot devices in jordan using shodan. In: 2017 2nd International Conference on the Applications of Information Technology in Developing Renewable Energy Processes & Systems (IT-DREPS). pp. 1–6. IEEE (2017)
2. Antonakakis, M., April, T., Bailey, M., Bernhard, M., Bursztein, E., Cochran, J., Durumeric, Z., Halderman, J.A., Invernizzi, L., Kallitsis, M., et al.: Understanding the mirai botnet. In: 26th {USENIX} Security Symposium ({USENIX} Security 17). pp. 1093–1110 (2017)
3. Antrobus, R., Green, B., Frey, S., Rashid, A.: The forgotten i in iiot: a vulnerability scanner for industrial internet of things (2019)
4. Department, S.R.: Number of connected devices worldwide 2030 (Feb 2020), `https://bit.ly/Statista-IoTConnectivity`
5. Durumeric, Z., Adrian, D., Mirian, A., Bailey, M., Halderman, J.A.: A search engine backed by internet-wide scanning. In: Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security. pp. 542–553 (2015)
6. Graham, R.D.: Masscan: Mass ip port scanner. URL: https://github.com/robertdavidgraham/masscan (2014)
7. Heffner, C.: Binwalk: Firmware analysis tool. URL: https://code. google.com/p/binwalk/(visited on 03/03/2013) (2013)

8. Intelligence, O.C.T., Committee, T., et al.: Structured threat information expression (stix) (2017)
9. Kim, H., Kim, T., Jang, D.: An intelligent improvement of internet-wide scan engine for fast discovery of vulnerable iot devices. Symmetry **10**(5), 151 (2018)
10. Ko, E., Kim, T., Kim, H.: Management platform of threats information in iot environment. Journal of Ambient Intelligence and Humanized Computing **9**(4), 1167–1176 (2018)
11. Kumar, D., Shen, K., Case, B., Garg, D., Alperovich, G., Kuznetsov, D., Gupta, R., Durumeric, Z.: All things considered: an analysis of iot devices on home networks. In: 28th {USENIX} Security Symposium ({USENIX} Security 19). pp. 1169–1185 (2019)
12. LLC, M.: How to hack a self-driving car with low tech paint and other serious artificial intelligence... (May 2019), `https://bit.ly/IPv6Scanning`
13. Lyon, G.F.: Nmap network scanning: The official Nmap project guide to network discovery and security scanning. Insecure (2009)
14. Markowsky, L., Markowsky, G.: Scanning for vulnerable devices in the internet of things. In: 2015 IEEE 8th International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS). vol. 1, pp. 463–467. IEEE (2015)
15. Matherly, J.: Shodan exploits, 2015. Accessed: April (2017)
16. Matherly, J.: Complete guide to shodan. Shodan, LLC (2016-02-25) **1** (2015)
17. Maurin, T., Ducreux, L.F., Caraiman, G., Sissoko, P.: Iot security assessment through the interfaces p-scan test bench platform. In: 2018 Design, Automation & Test in Europe Conference & Exhibition (DATE). pp. 1007–1008. IEEE (2018)
18. Miessler, D., Smith, C.: Owasp internet of things project. OWASP Internet of Things Project-OWASP (2018)
19. Nupi: The nordic countries - landing page, `https://bit.ly/Nordic-COntries`
20. O'Donnell, L.: 2 million iot devices vulnerable to complete takeover ... (Apr 2019), `https://threatpost.com/iot-devices-vulnerable-takeover/144167/`
21. Ogunnaike, R.M., Lagesse, B.: Toward consumer-friendly security in smart environments. In: 2017 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops). pp. 612–617. IEEE (2017)
22. Okoli, C., Schabram, K.: A guide to conducting a systematic literature review of information systems research (2010)
23. Oltermann, P.: Briton admits to cyber-attack on deutsche telekom (Jul 2017), `https://bit.ly/Guardian-DeutscheTelekom`
24. Orebaugh, A., Ramirez, G., Beale, J.: Wireshark & Ethereal network protocol analyzer toolkit. Elsevier (2006)
25. Samarasinghe, N., Mannan, M.: Another look at tls ecosystems in networked devices vs. web servers. Computers & Security **80**, 1–13 (2019)
26. Sheridan, K.: Microsoft challenges security researchers to hack azure sphere (May 2020), `https://bit.ly/Microsoft-Challenge`
27. Tekeoglu, A., Tosun, A.Ş.: A testbed for security and privacy analysis of iot devices. In: 2016 IEEE 13th International Conference on Mobile Ad Hoc and Sensor Systems (MASS). pp. 343–348. IEEE (2016)
28. Vijayan, J.: Most bluetooth devices vulnerable to impersonation attacks (May 2020), `https://bit.ly/VulnerableBluetooth`