PAPER Special Section on Next-generation Security Applications and Practice

A Design of Automated Vulnerability Information Management System for Secure Use of Internet-Connected Devices Based on Internet-Wide Scanning Methods

Taeeun KIM[†a)], ***Nonmember and*** **Hwankuk KIM**[††b)], ***Member***

SUMMARY Any Internet-connected device is vulnerable to being hacked and misused. Hackers can find vulnerable IoT devices, infect malicious codes, build massive IoT botnets, and remotely control IoT devices through C&C servers. Many studies have been attempted to apply various security features on IoT devices to prevent IoT devices from being exploited by attackers. However, unlike high-performance PCs, IoT devices are lightweight, low-power, and low-cost devices and have limitations on performance of processing and memory, making it difficult to install heavy security functions. Instead of access to applying security functions on IoT devices, Internet-wide scanning (e.g., Shodan) studies have been attempted to quickly discover and take security measures massive IoT devices with weak security. Over the Internet, scanning studies remotely also exist realistic limitations such as low accuracy in analyzing security vulnerabilities due to a lack of device information or filtered by network security devices. In this paper, we propose a system for remotely collecting information from Internet-connected devices and using scanning techniques to identify and manage vulnerability information from IoT devices. The proposed system improves the open-source Zmap engine to solve a realistic problem when attempting to scan through real Internet. As a result, performance measurements show equal or superior results compared to previous Shodan, Zmap-based scanning.

key words: *OSINT, IoT security, Internet-wide scan, security management, vulnerability management*

1. Introduction

A modern IT environment is heavily dependent on and closely related to the Internet. Many people use diverse web services and mobile devices every day, creating and providing a wide variety of information. The performance (e.g., speed and reliability) of wireless communication (Wi-Fi, Bluetooth, Zigbee, etc.) has advanced, and the diffusion of small-size devices connected to the Internet (CCTV, Smart-Home, etc.) has increased sharply. The number of services that use various types of IoT devices is also increasing in line with the ongoing technological and environmental changes. IoT Analytics reported in 2018 that the number of Internet-Connected IoT devices in the market will reach 34.2 billion by 2025 as shown in Fig. 1 [1].

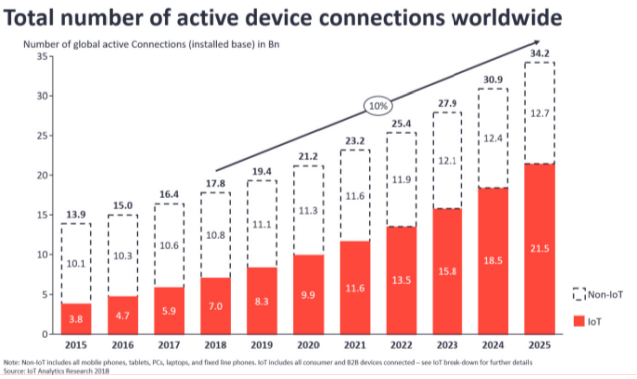


Fig. 1 Total number of active device connections worldwide (IoT analytics, 2018)

Contrary to the sharp increase in the number of IoT devices and services, however, research on those devices and services remains in the initial stage, while cyber-attacks that exploit the vulnerability of Internet-connected devices are on the rise. According to the results of a device vulnerability inspection conducted by CISCO in 2016, network devices such as routers and switches have 28 vulnerabilities on average. Also, 23% of those devices were found to have the vulnerability that had been announced five years before, and 10% of them were found to have a vulnerability dating back to more than a decade [2]. Generally speaking, such networks and IoT devices are not properly managed, i.e., they are not subjected to periodic firmware updates after the first installation by the user. Therefore, preventive measures - like quick scan - are needed for devices with an old vulnerability, because such devices can become major attack targets.

Vulnerable Internet-connected devices have several characteristics in common. First, vulnerable devices have no security function and use vulnerable version of OS, open-source, and communication protocols for the sake of convenient development. Second is a management difficulty caused by the characteristics of device usage. According to a report published by CISCO, devices such as CCTVs, IP routers, and printers are not directly connected to devices, so users tend not to apply security updates periodically, leaving them unattended. A device left in such vulnerable state may run normally but can be exploited maliciously by a hacker [27]–[39].

Manuscript received February 7, 2021.
Manuscript revised June 19, 2021.
Manuscript publicized August 2, 2021.
[†]The author is with Korea Information & Security Agency, Naju-si, Jeollanam-do, 58324 Korea.
[††]The author is with the Department of Security Engineering, Sangmyung University, Cheonan-si, 31066 Korea.
a) E-mail: tekim31@kisa.or.kr
b) E-mail: rinyfeel@smu.ac.kr (Corresponding author)
DOI: 10.1587/transinf.2021NGP0004

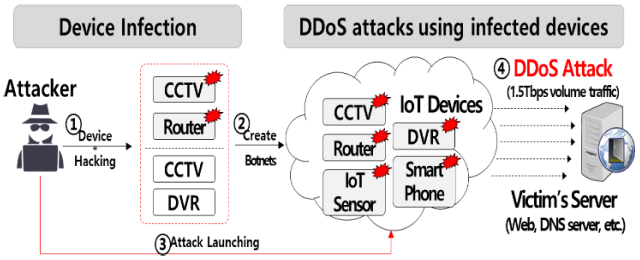


Fig. 2 Diagram of DDoS attack using vulnerable IoT devices

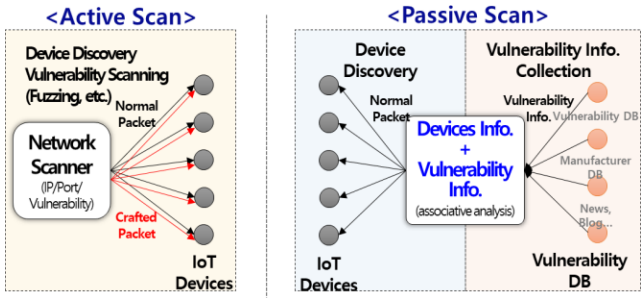


Fig. 3 Comparison of active scan and passive scan

The following is a true example of cyber-attacks carried out through vulnerable devices with Internet-connections as shown in Fig. 2. Two DDoS attacks targeted a French web hosting company (OHV) and a US DNS service provider (Dyn) in 2016. Unlike other attacks that create attack traffic by infecting other PCs, these two attacks shared a common characteristic the IoT device, in this case a CCTV, created a botnet. The attack was carried out by infecting the CCTV with a weak authentication mechanism using the malicious code "Mirai." Traffic generated by the infected device disabled the service by sending around 1 ∼ 30 Mbps of traffic per IP, amounting to 1.5 Tbps in all. The attack on Dyn caused the service shutdown of 1,200 large websites such as Twitter and Netflix. The incidents began from a vulnerable IoT device that either had no security function or used a default password without setting a password for authentication.

As described above, we are using many vulnerable devices that lack proper security to distribute convenient services quickly. Therefore, we need to develop a technology quite unlike the conventional method of managing a few devices such as PCs and servers.

Internet-Wide Scan" is a representative technology that locates many IP address-based devices and collects information from them. There is also a "security vulnerability analysis" technology that analyzes known vulnerabilities in the collected device information by using known vulnerability information. Using those technologies, this paper emphasizes the importance of advance prevention based on the identification rather than on the updating of vulnerable devices.

This paper proposes a system architecture that can detect vulnerabilities remotely in Internet-connected devices to prevent cyberattacks, and verifies their performance against existing techniques. This paper contributes the following:

First, the proposed technique addresses challenges (e.g., filtering by security equipment) that are difficult to apply on the real Internet to analyze security vulnerabilities in devices remotely connected to the Internet. Second, the proposed system performed better on a variety of performance metrics (scan speed, information collection speed, and vulnerability information analysis) than the representative OSINT tools, Zmap and Shodan. Third, various commercial services equipped with Shodan engines are being released to analyze remote device security vulnerabilities, and Korea's KISA (Korea Internet & Security Agency) is used as a security vulnerability check technology for small and medium-sized IoT devices using the proposed system.

2. Related Works

The proposed technique is an excessive study of systems that collect and analyze vulnerability information of devices connected to Internet remotely and manage vulnerabilities. In this chapter, we analyze the limitations of existing research by dividing it into Internet-wide scanning techniques and security vulnerability information analysis techniques as a prior study.

2.1 Internet-Wide Scan

The previous network scan technology collected device information by checking the operating system of a single device (single IP) and scanning open ports [3]–[6]. The vulnerability of a particular device was then searched to analyze its vulnerability using an aggressive technique. For example, well-known IDs and passwords are used to check the vulnerability of the default password, with the device attacked directly using the exploit code. This type of scan is called "Active Scan" including tools such as NMAP, Nessus, and Defensics as shown in Fig. 3. However, these tools are unsuitable for checking many remote devices. This is because the device cannot run normally as aggressive behavior or traffic is created by the device itself and may not run again due to a shutdown. Therefore, the "Passive Scan" technology has been developed to collect information on many remote devices quickly [7]–[16]. The comparison of characteristics of Active Scan and Passive Scan is shown in Table 1.

Passive Scan technology does not use an aggressive method to collect device information. This tool obtains the necessary information from the response message after sending a normal communication message to the device concerned to collect information. Also, majority of Internet-connected devices are targets of collection. The banner information that can be obtained when connecting to the Telnet service as well as the communication traffic header information can be collected quickly. Shodan, Censys, and Masscan are all tools that use this type of scan.

Table 1 Comparison of characteristics of active scan and passive scan

Category	Active Scan	Passive Scan
Type	- Vulnerability Scanning - Port Scanning, etc.	- IP Scanning, - Port Scanning, etc.
Characteristics	Authorized User → Credential Action	Unauthorized User → Non-Credential Action
Target	Internal devices on the local network (private IP)	Internet-connected remote device (public IP)
Vulnerability analysis	Known/Unknown Vulnerability Analysis	Known Vulnerability Information Analysis
Related tools	NMAP, Defensics, Nessus, etc.	ZMap, Shodan, Censys, etc.

Fig. 4 Basic search results for Shodan web services

John Matherly has developed and established "Shodan," a search engine that can retrieve information on Internet-connected devices as shown in Fig. 4. Shodan collects information from all the open ports of an Internet-connected device (where most of the information can be retrieved from the keywords included in the banner information). More than 160 protocols are supported, and the information on all the ports (65,535 ea.) available in the device can be collected. Besides, vulnerabilities like Heartbleed and Poodle are analyzed using the use and version information of SSL cryptographic communication.

However, the technical method such as scanning speed and method cannot be analyzed as the source code of Shodan is not opened as an open-source, except for the API [17]. Zakir Durumeric of the University of Michigan has developed Internet-Wide scan engines such as ZMap and ZGrab to collect Internet-connected device information at high speed [18]–[21], as shown in Fig. 5. These tools are opened as an open-source, and a search engine called "Censys" is provided using the opened tools. Censys collects information from 23 major protocols such as HTTP and TELNET. Like Shodan, Censys can retrieve keywords from banner information and provide encryption protocol information including the related vulnerability information [22].

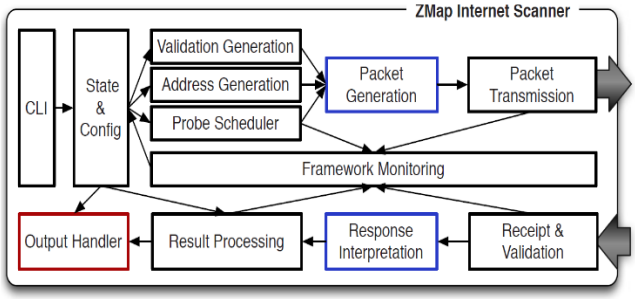


Fig. 5 Structure of ZMap internet scanner

Shodan and Censys differ in terms of the speed and amount of information collection. Shodan updates the information of the entire IPv4 address range every four weeks. However, the information includes all ports that can be used by the device. On the other hand, Censys updates major ports every two weeks. When Censys scans the entire IPv4 address range using one collection probe, the alive state of the device can be checked at 1:09:45.

However, Shodan and Censys are search engines that match the keyword retrieved by the user in the banner. An accurate result cannot be obtained easily using the keyword (e.g., product type) entered by the analyst to check the device vulnerability information. For example, when a keyword like "CCTV" is entered, Shodan and Censys may match the keyword in the CCTV sales website, but the analyst consequently needs to summarize, check, and analyze the search result.

2.2 Technology for Analyzing Security Vulnerability Information

Currently, security vulnerability technology can be classified according to whether or not it applies a direct action to the analysis target (device), i.e., in the same way as the classification of network scan technology.

A technology that does not apply a direct action to the analysis target is referred to as a "technology for analyzing security vulnerability information." The related area includes a technology that creates a structured database to manage vulnerability information and analyzes associations with the analysis target using the accumulated vulnerability information. Many organizations manage security vulnerability information according to their criteria. The CVE (Common Vulnerabilities and Exposures) managed by the NVD (National Vulnerability Database) are the representative criteria, while Bugtraq, VulDB, and device and software manufacturers also manage vulnerability information.

A "vulnerability information scan technology" finds and identifies a vulnerable device using the structured infor-

mation as described above. The vulnerability information scan technology analyzes the correlation between vulnerability information and device information after collecting the information from a device analysis. Diverse algorithms range from simple keyword matching to similarity analysis. To detect vulnerabilities like Heartbleed and Poodle, Shodan and Censys find the OpenSSL version information from the collected traffic and match it with the relevant vulnerability information for analysis [23]–[26].

The use of the White-box/Black-box test to detect a vulnerability in an analysis target is referred to as a "security vulnerability analysis technology." The White-box test analyzes the code (e.g., software source, binary) directly and includes methodologies such as Control flow testing, Data flow testing, Branch testing, and Path testing. Meanwhile, the Black-box test analyzes a problem that occurs when executing software, using methods such as Decision table testing, All-pairs testing, Error guessing, and Boundary value analysis. Recently, IAST (Interactive Application Security Testing), which combines these two test methods, was released as a product, i.e., Web Vulnerability Scanner (Acunetix) and Seeker (Contrast Security).

The security vulnerability analysis technology tries various methods (e.g., Active Network Scan) to find a defect directly from a device. However, these methods disrupt normal device operation, and they are not suitable for application to many Internet-connected devices. Therefore, this paper proposes a system that can manage many Internet-connected devices using the vulnerability information analysis technology.

3. Vulnerability Information Management System of Internet-Connected Device

This chapter proposes a system that can prevent cyber threats by searching for an Internet-connected device based on the IPv4 address, and then identifying the security vulnerability information. The proposed system uses the passive scan technology (Internet-Wide Scan) and the technology for analyzings security vulnerability information among the related technologies.

3.1 Composition of the Proposed Technology

A technology for collecting both device information and vulnerability information and for analyzing the vulnerability information contained in a device is needed to manage the vulnerability information of Internet-connected devices. Therefore, the proposed system consists of three functional modules the Internet-Connected Devices Scan Module, the Vulnerability Information Management Module, and the Devices Vulnerability Analysis Module as shown in Fig. 6.

The Internet-Connected Devices Scan Module sends communication packets to scan a device in the IPv4 address range and collects response packets to collect the device information. The Vulnerability Information Management Module crawls websites that provide vulnerability information to analyze the vulnerability information of the device. The Devices Vulnerability Analysis Module analyzes the correlation between the device information and the vulnerability information collected by the previous two modules and manages the result. Each module runs in a different server to maximize the system's performance.

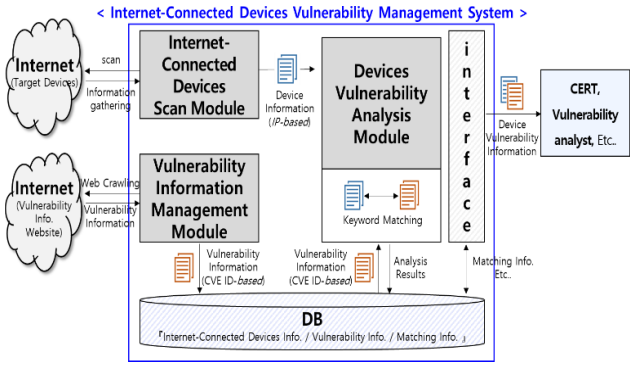


Fig. 6 Diagram of vulnerability management system for internet-connected devices

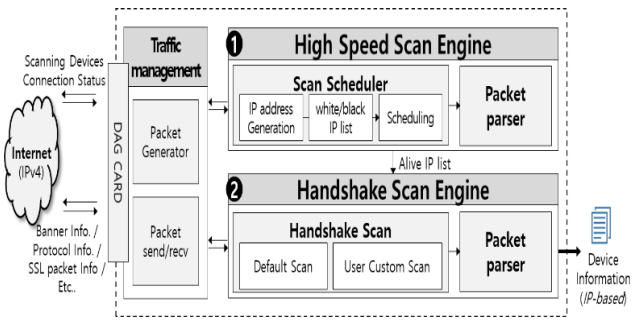


Fig. 7 Diagram of internet-connected devices scan module

3.2 Internet-Connected Devices Scan Module

Two sub-modules were developed using the ZMap and ZGrab open-sources to collect information on Internet-connected devices. Also, the traffic management module was separately implemented to control the scan traffic generated by the two collection modules, and its performance was enhanced by applying DAG-CARD and PF-RING. The traffic management function generates and sends scan traffic to collect device information and receives the response packet from the device. Figure 7 shows the composition of the Internet-Wide Scan Module.

1) High-speed scan engine The first sub-module is the "High-Speed Scan Engine" used to scan a device at high speed. This module performs the functions of scan scheduler and scan traffic parsing to scan the device connection state.

The scan scheduler creates an address list to scan about 4.3 billion IPv4 addresses. The process of creating a scan IP address list plays an important role in the Internet-Wide Scan technology. If the list of scanned IP addresses is created sequentially, it will be detected by security equipment

(e.g., firewall, IDS, IPS) as an attack, and more scans will be blocked. We used an algorithm that converts IP addresses into decimal numbers and circulates them to generate an address list randomly using the existing prototype. When the module was tested, however, it was detected by security equipment due to excessive traffic, and the packets were dropped. To solve these problems, a scan scheduler that classifies IP assignment authorities based on the Whois information and applies the delay time to the same address range was developed. In addition, the White/Black List function was applied to the scan scheduler to improve the Hit-rate.

If traffic is generated with the IP address created by the scan scheduler, the response message (ACK) can be received from the Internet-connected device. An "active IP list" is created by processing response packets to collect the port information of the device being responded to.

2) Handshake scan engine The second sub-module is the "Handshake Scan Engine" module, which is used to collect the information of each device port. This module collects the system connection banner, cryptographic communication information, and packet information from major device ports like the ZGrab open-source. However, a function that collects information from various ports according to the user definition is needed to prevent cyber-attacks. Therefore, functions were added to the 15 basic protocol scan functions provided by ZGrab, such as user-defined port and data, as well as to the random protocol scan by extracting traffic as PCAP. The development of these functions enabled us to expand the scope of the device information that can be collected from the entire IPv4 address range.

3.3 Vulnerability Information Management Module

To analyze vulnerability information for devices connected to the Internet, it is necessary to collect Open Vulnerability Information (CVE) information. The vulnerability information collected by the Vulnerability Information Management Module was based on Common Platform Enumeration (CPE), Common Fragment Enumeration (CWE), and Common Vulnerability Scoring System (CVSS), as defined by NVD's CVE standards.

This module also collects vulnerability information such as Bugtraq, VulDB, and Rapid7 to analyze vulnerability information for various device types connected to the Internet as well as CVE information from NVD. As a result, we collected approximately 100,000 CVE-related vulnerability information. The collected vulnerability information is divided into CVE types and non-CVE types. Because Bugtraq and VulDB have IDs assigned to each vulnerability and contain related CVE-IDs, mapping is possible in common on a CVE-ID basis. If a new vulnerability is found that is not registered with CVE, it is classified as "unusual vulnerability information" and treated as a type of vulnerability information.

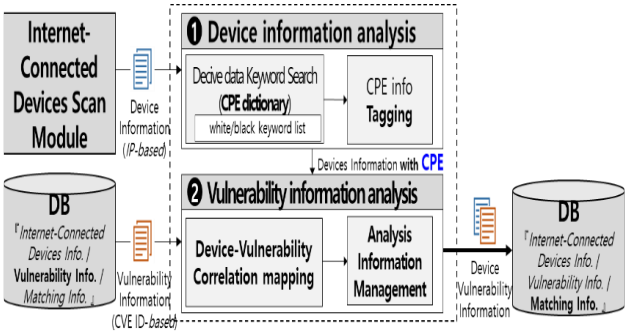


Fig. 8 Diagram of devices vulnerability analysis module

3.4 Devices Vulnerability Analysis Module

The device information and the common keyword in the vulnerability information should be matched to check the existence of vulnerability in the collected device information. The "Devices Vulnerability Analysis Module" is composed of two sub-modules (Fig. 8): the "device information analysis" module, which finds a keyword (e.g., manufacturer, product name) in the device information to identify the CPE information; and the "vulnerability information analysis" module, which then relates the CPE found by the device information analysis module to the vulnerability information.

1) Device information analysis

The device information analysis module identifies the CPE information from the banner, packet payload (data), and Handshake traffic information collected by scanning the device. The CPE information to be identified is the name and version information of the standardized product as defined in the "CPE dictionary." Any similarity with the CPE dictionary information is analyzed to identify the CPE information from the device information, and the information is identified as CPE if the result of the similarity analysis is above 80% to increase accuracy. A similarity analysis is conducted in order of manufacturer and product version if its accuracy in identifying the product name exceeds 80%.

However, the previously developed prototype caused many errors in identifying a device because the commonly used word (e.g., a, login) was included in the information registered in the CPE dictionary. Therefore, a function for creating and managing the White/Black List has also been added to manage frequently used English words or special keywords.

The pre-defined device/product type information is tagged to the CPE information of the device identified in this manner. Thus, device-type information is classified into 6 types, such as network equipment or IoT device, etc., while IoT devices are subdivided into 26 types, such as IP cameras, NVR/DVR, and devices vulnerable to malware infection, etc.

2) Vulnerability information analysis

The vulnerability information analysis module matches and analyzes the CPE information of the device identified

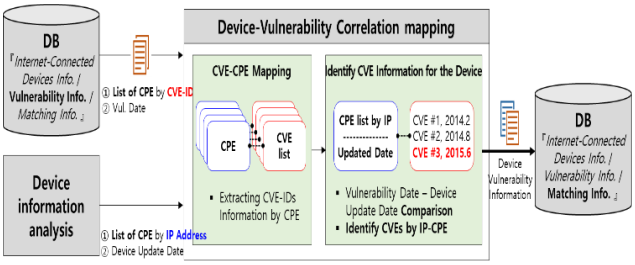


Fig. 9 Diagram of device-vulnerability correlation mapping function

by the previous device information analysis module and the CPE information included in the vulnerability information collected by the vulnerability information management module (Fig. 9). If the CPE identified from the device information is included in the vulnerability information, it means that the matching vulnerability is present in the device.

A simple keyword matching algorithm rather than a complex algorithm is used for CPE information matching because the device information and vulnerability information are processed in advance. The device-vulnerability information matched in this manner is created and managed in JSON format to share with the CERT and vulnerability analysts.

4. Analyzing the Performance of the Proposed Technology

This chapter verifies the performance of each proposed module and checks for areas requiring improvement by comparing them with existing technologies.

4.1 Performance of High-Speed Scan Engine (Internet Device Scan Rate)

The performance of high-speed scan engines is associated with many devices that use IP addresses. In order to quickly collect information on devices that increase geometrically, the operating cycle of the proposed system can be set by measuring the scheduling and traffic processing performance.

High-speed traffic processing performance is important to check the connection state of a device by scanning all 4.3 billion IPv4 address spaces. The scan traffic that checked the connection state of 3.680 billion IPv4 IP addresses, excluding the reserved IPs, was generated, and the speed of the processing module was tested. (Total number of generated IPs: 3,689,610,499 ea.)

The test was conducted using the BigTao measuring instrument in a test environment rather than in the actual network. Traffic was generated without delay at high speed to measure the maximum scanning speed. When tested in this manner, the normal scan speed could not be measured due to detection by security equipment.

$$\textit{Scanning speed (TPM)} = \textit{Generated packets per second} * 60 \quad (1)$$

Table 2 IP address high-speed scan engines test results

Round	Proposed Method (TPM)	ZMap(TPM)	Masscan (TPM)
1st	86,021,700	85,239,000	56,186,160
2nd	85,910,280	85,913,280	56,586,840
3rd	85,883,520	85,925,160	54,978,420
Average	85,938,500	85,692,480	55,917,140

The scanning speed is determined in TPM (Throughput Per Minute) by measuring the packet amount created every minute. The test verified the developed module, ZMap, and Masscan comparatively. TCP Syn packets were then generated in each compared module that scanned the connection state of each IP address, with the total number of frames counted for 10 minutes using the measuring instrument.

The results of the scanning speed test showed that it was measured at 85.940 million TPM per minute on average, and that the target IPv4 addresses (3.680 billion IP addresses) were scanned in 44 minutes and 8 seconds (Table 2).

4.2 Performance of the Handshake Scan Engine

The performance of a handshake scan engine means how much information can be captured by a device that uses an IP address. Therefore, the key performance of the handshake search engine is used to determine the amount of information that can be collected. However, when scanning all port information of a device that uses an IP address, the information collection speed will be greatly affected. We tested the ability to effectively collect information on the major ports where weaknesses are found. As a result, the device information collection ratio was calculated by comparing the number of IPs scanned using the proposed technology with the number of IPs of Shodan and Censys.

$$\textit{Information collection rate (\%)} = \textit{Number of scanned IPs by port} / \textit{Number of IPs scanned by Shodan (Censys)} * 100 \quad (2)$$

The test retrieves the IP list of the same protocol using the API provided by Shodan and Censys after extracting the IP list by protocol from the information collected by the Internet-Wide Scan engine. However, the number of scanned IPs was used to compare it with Shodan because the entire IP list of a specific port could not be obtained.

For the test, a total of 34 services (or ports), besides the information shown in Table 2 above, were compared with the Shodan and Censys data. It was found that the total number of IPs collected from every port (duplication was not removed) was 223,559,189 for the proposed technology, 212,428,355 for Shodan, and 173,496,395 for Censys (13 ports). Compared with Shodan and 34 ports, 105.24% more device information could be collected. Compared with Censys and 13 ports, however, 128.86% more device information could be collected.

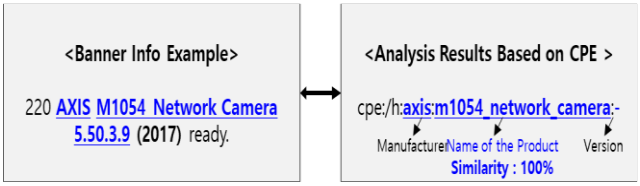


Fig. 10 Example of banner information analysis

4.3 Performance of Devices Vulnerability Analysis Module

The Devices Vulnerability Analysis Module analyzes the vulnerability present in a device by matching the device information and vulnerability information collected from the previous two modules. The results analyzed by this module are the stage of analyzing whether vulnerabilities actually exist in devices connected to the Internet. Therefore, unlike the method of analyzing by actual vulnerability test, it is important to collect as much information as possible and ensure the performance to analyze it in relation to it.

The data used in the performance measurement consisted of four types of port information (FTP, HTTP, SSH, and Telnet) that could identify the CPE information by analyzing the banner.

The test calculated the CPE matching rate that can map the software manufacturer and version information and the security vulnerability information among the collected device information. The vulnerability information can be found by identifying the CPE as the CVE vulnerability information includes the related device information in CPE format (an example of banner information analysis is shown in Fig. 10).

$$\textit{Vulnerability analysis rate (\%)} = \textit{Number of normal CPE identifications} / \textit{Total number of banners} * 100 \quad (3)$$

The vulnerability analysis rate calculates the rate of normal CPE identification among the total number of collected banners. A total of 100,000 banners were sampled from 4 types of protocol data that were suitable for the test.

The top 1,000 unique samples were selected randomly from the sampled banner. (Rate of the top 1,000 unique banners excluding duplication: FTP – 69%, HTTP – 96%, SSH – 99%, TELNET – 82%)

The analysis was conducted by setting the banner information and the CPE matching threshold value at 66%, 75%, and 80%, respectively. The vulnerability information analysis results are shown in Fig. 11 and Table 3.

The result of the vulnerability analysis rate test showed that the normal identification rate of the banner information was 87.38%. When the CPE matching threshold value was set above 80%, all four protocols showed the highest rate of normal CPE recognition from the device information.

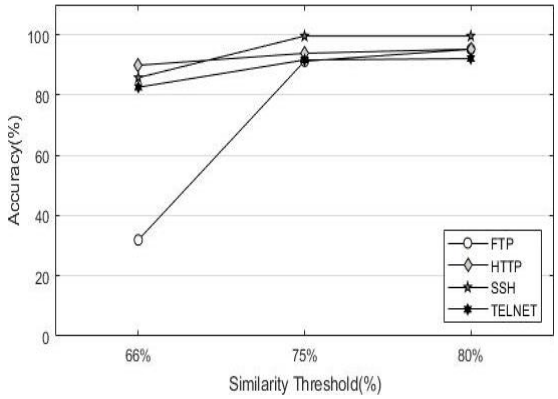


Fig. 11 Identification rate of the device information for CPE

Table 3 Result of the CVE analysis of device information

Service	Similarity Threshold 66%	Similarity Threshold 75%	Similarity Threshold 80%	Average
FTP	31.9%	91.2%	95.2%	72.8%
HTTP	89.9%	93.9%	95.2%	93%
SSH	85.8%	99.6%	99.6%	95%
Right	82.6%	91.6%	92.1%	88.8%

5. Conclusion

Recently, the number of Internet-connected devices has increased sharply due to the proliferation of IoT services. However, the security of IoT devices cannot be managed as easily as that of the existing PC and server use environment.

A security program (such as vaccine) cannot be installed in small devices, and network environment security (e.g., firewall) cannot be applied easily. Besides, IoT devices contain many vulnerabilities because they use vulnerable open-source software and Cut-down OS. Small, vulnerable devices can be exploited for cyber-attacks, including large-scale DDoS attacks. Therefore, vulnerabilities should be managed regularly to prevent such attacks.

This paper proposes a system structure for managing the vulnerabilities of Internet-connected devices. The developed system was also tested comparatively in various environments. The existing Internet-Wide Scan technology is being developed with focus on a method of collecting a large amount of device information quickly. Experts in vulnerability analysis detect a device's vulnerability using the information provided by Shodan and other service providers that use scan technology. The proposed system is designed to prevent cyber-attacks in advance by automating the process of manual vulnerability analysis using Internet scan. The proposed technology is already being used as a security service for the public. This service is offered to the public peoples and small businesses. With the consent of the service applicant, the result of scanning the device and port information connected to the network is generated. We will collect this result, analyze the applicant's equipment for weaknesses, and guide the result of the measures. This ser-

vice has a great effect in preventing cyberattacks that exploit vulnerable small devices in advance, and we will endeavor to provide the service in an automated manner in the future.

Acknowledgments

This work was supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No. 2016-0-00193, IoT Security Vulnerabilities Search, Sharing, and Testing Technology Development, 50%) and Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No.2021-0-00358, AI·Big data based Cyber Security Orchestration and Automated Response Technology Development, 50%).

References

[1] K.L. Lueth, "State of the IoT 2018: Number of IoT devices now at 7B," IoT Analytics., 2018. [Online]. Available: https://iot-analytics.com/state-of-the-iot-update-q1-q2-2018-number-of-iot-devices-now-7b/, accessed on Aug. 15, 2019.

[2] "2016 midyear cyber security report of cisco," Cisco., http://www.cisco.com/c/dam/m/en_ca/never-better/assets/files/midyearsecurity-report-2016.pdf, 2016. accessed on Aug. 15, 2019.

[3] Q. Xu, R. Zheng, W. Saad, and Z. Han, "Device Fingerprinting in Wireless Networks: Challenges and Opportunities," IEEE Communications Surveys & Tutorials, vol.18, no.1, pp.94–104, 2016.

[4] Y.-C. Chen, Y. Liao, M. Baldi, S.J. Lee, and L. Qiu, "OS Fingerprinting and Thethering Detection in Mobile Networks," Proc. 2014 Conference on Internet Measurement Conference, pp.173–180, 2014.

[5] Z. Shamsi, A. Nandwani, D. Leonard, and D. Loguinov, "Hershel: Single-Packet OS Fingerprinting," IEEE/ACM Trans. Netw., vol.24, no.4, pp.2196–2209, 2016.

[6] A. Costin, J. Zaddach, A. Francillon, and D. Balzarotti, "A Large-Scale Analysis of the Security of Embedded Firmwares," 23rd USENIX conference on Security Symposium 2014, pp.95–110, 2014.

[7] G. Bartlett, J. Heidemann, and C. Papadopoulos, "Understanding passive and active service discovery," Proc. 7th ACM SIGCOMM conference on Internet measurement - IMC '07, pp.57–70, 2007.

[8] M. Li, H. Chen, X. Huang, and L. Cui, "EasiCrawl: A Sleep-aware Schedule Method for Crawling IoT Sensors," 2015 IEEE 21st International Conference on Parallel and Distributed Systems (ICPADS), pp.148–155, 2015.

[9] Y. Kong and D. Shen, "Research on Collecting Real-Time Information on Dynamic Web Pages of Internet of Things," 2013 International Conference on Computational and Information Sciences, pp.563–566, 2013.

[10] D. Leonard and D. Loguinov, "Demystifying Service Discovery: Implementing an Internet-Wide Scanner," Proc. 10th annual conference on Internet measurement - IMC '10, pp.109–122, 2010.

[11] S. Khattak, D. Fifield, S. Afroz, M. Javed, S. Sundaresan, V. Paxson, S.J. Murdoch, and D. McCoy, "Do You See What I See? Differential Treatment of Anonymous Users," Proc. 2016 Network and Distributed System Security Symposium, 2016.

[12] G.C.M. Moura, C. Ganan, Q. Lone, P. Poursaied, H. Asghari, and M. van Eeten, "How Dynamic is the ISPs Address Space? Towards Internet-Wide DHCP Churn Estimation," 2015 IFIP Networking Conference (IFIP Networking), pp.1–9, 2015.

[13] R. Trapkickin, "Who is Scanning the Internet?," the Seminars Future Internet (FI) and Innovative Internet Technologies and Mobile Communications (IITM) 2015.

[14] D. Myers, E. Foo, and K. Radke, "Internet-wide Scanning Taxonomy and Framework," 13th Australasian Information Security Conference 2015.

[15] A.V. Arzhakov and I.F. Babalova, "Analysis of Current Internet Wide Scan Effectiveness," 2017 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIConRus), pp.96–99, 2017. (doi:10.1109/EIConRus.2017.7910503)

[16] Anonymous.: Internet census 2012, http://census2012.sourceforge.net/paper.html, accessed on Oct. 9, 2019.

[17] R. Bodenheim, J. Butts, S. Dunlap, and B. Mullins, "Evaluation of the ability of the Shodan search engine to identify Internet-facing industrial control devices," International Journal of Critical Infrastructure Protection, vol.7, no.2, pp.114–123, 2014.

[18] D. Adrian, Z. Durumeric, G. Singh, and J.A. Halderman, "Zippier ZMap: Internet-wide scanning at 10Gbps," 8th USENIX conference on Offensive Technologies 2014.

[19] Z. Durumeric, D. Adrian, A. Mirian, M. Bailey, and J.A. Halderman, "A Search Engine Backed by Internet-Wide Scanning," Proc. 22nd ACM SIGSAC Conference on Computer and Communications Security, pp.542–553, 2015.

[20] Z. Durumeric, E. Wustrow, and J.A. Halderman, "ZMap: Fast Internet-Wide Scanning and its Security Applications," 22nd USENIX conference on Security 2013.

[21] Z. Durumeric, M. Bailey, and J.A. Halderman, "An Internet-Wide View of Internet-Wide Scanning," 23rd USENIX conference on Security Symposium 2014, pp.65–78, 2014.

[22] B. Genge, P, Haller, and C. Enachescu, "Beyond Internet Scanning: Banner Processing for Passive Software Vulnerability Assesment," International Journal of Information Security Science 2015, vol.4, no.3, pp.81–91, 2015.

[23] B. Genge and C. Enăchescu, "ShoVAT: Shodan-based vulnerability assesment tool for Internet-facing services," Security and Communication Networks, vol.9, no.15, pp.2696–2714, 2016.

[24] Y.S. Ko, I.K. Ra, and C.S. Kim, "A Study on IP Exposure Notification System for IoT Devices Using IP Search Engine Shodan," International Journal of Multimedia and Ubiquitous Engineering, vol.10, no.12, pp.61–66, 2015.

[25] Z. Durumeric, F. Li, J. Kasten, J. Amann, J. Beekman, M. Payer, N. Weaver, D. Adrian, V. Paxson, M. Bailey, and J.A. Halderman, "The Matter of Heartbleed," Proc. 2014 Conference on Internet Measurement Conference, pp.475–488, 2014.

[26] A. Panchenko, F. Lanze, A. Zinnen, M. Henze, J. Pennekamp, K. Wehrle, and T. Engel, "Website Fingerprinting at Internet Scale," Proc. 2016 Network and Distributed System Security Symposium, 2016.

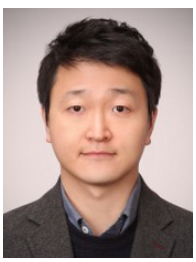
[27] J.H. Eom, "Security Threats Recognition and Countermeasures on Smart Battlefield Environment based on IoT," International Journal of Security and Its Applications, NADIA, ISSN: 1738-9976 (Print); 2207-9629 (Online), vol.9, no.7, pp.347–356, July 2015, http://dx.doi.org/10.14257/ijsia.2015.9.7.32.

[28] J.H. Yang and Y. Ryu, "Design and Development of a Command-line Tool for Portable Executable File Analysis and Malware Detection in IoT Devices," International Journal of Security and Its Applications, NADIA, ISSN: 1738-9976 (Print); 2207-9629 (Online), vol.9, no.8, pp.127–136, Aug. 2015, http://dx.doi.org/10.14257/ijsia.2015.9.8.10.

[29] J.T. Kim, "Requirement of Security for IoT Application based on Gateway System," International Journal of Security and Its Applications, NADIA, ISSN: 1738-9976 (Print); 2207-9629 (Online), vol.9, no.10, pp.201–208, Oct. 2015, http://dx.doi.org/10.14257/ijsia.2015.9.10.18.

[30] A. Patil, G. Bansod, and N. Pisharoty, "Hybrid Lightweight and Robust Encryption Design for Security in IoT," International Journal of Security and Its Applications, NADIA, ISSN: 1738-9976 (Print); 2207-9629 (Online), vol.9, no.12, pp.85–98, Dec. 2015, http://dx.doi.org/10.14257/ijsia.2015.9.12.10.

[31] S.H. Jung, J.C. An, J.Y. Park, Y.T. Shin, and J.B. Kim, "An Empirical Study of the Military IoT Security Priorities," International Journal of Security and Its Applications, NADIA, ISSN: 1738-9976 (Print); 2207-9629 (Online), vol.10, no.8, pp.13–22, Aug. 2016, http://dx.doi.org/10.14257/ijsia.2016.10.8.02.
[32] C. Sudhakar, N.T. Rao, and D. Bhattacharyya, "Smart Electronic Stick for Blind People: An IOT Application," International Journal of Security and Its Applications, NADIA, ISSN: 1738-9976 (Print); 2207-9629 (Online), vol.13, no.1, pp.1–10, March 2019, http://dx.doi.org/10.33832/ijsia.2019.13.1.01.
[33] H. Gao, Y. Xu, Y. Yin, W. Zhang, R. Li, and X. Wang, "Context-aware QoS Prediction with Neural Collaborative Filtering for Internet-of-Things Services," IEEE Internet Things J., 2019, https://doi.org/10.1109/JIOT.2019.2956827
[34] X. Ma, H. Gao, H. Xu, and M. Bian, "An IoT-based task scheduling optimization scheme considering the deadline and cost-aware scientific workflow for cloud computing," EURASIP Journal on Wireless Commun. and Networking, 2019, https://doi.org/10.1186/s13638-019-1557-3
[35] Y. Yun, J. Xia, Y. Li, Y. Xu, W. Xu, and L. Yu, "Group-Wise Itinerary Planning in Temporary Mobile Social Network," IEEE Access, vol.7, pp.83682–83693, 2019.
[36] Y. Yin, L. Chen, Y. Xu, J. Wan, H. Zhang, and Z. Mai, QoS Prediction for Service Recommendation with Deep Feature Learning in Edge Computing Environment, Mobile Networks and Applications, 2019. https://doi.org/10.1007/s11036-019-01241-7
[37] Y. Jin, X. Guo, Y. Li, J. Xing, and H. Tian, Towards Stabilizing Facial Landmark Detection and Tracking via Hierarchical Filtering: A new method, Journal of the Franklin Institute, vol.357, no.5, pp.3019–3037, 2020. DOI: 10.1016/j.jfranklin.2019.12.043.
[38] H. Gao, Y. Duan, L. Shao, and X. Sun, "Transformation-based processing of typed resources for multimedia sources in the IoT environment," Wirel. Netw., vol.27, no.5, pp.3377–3393, 2021. DOI: 10.1007/s11276-019-02200-6
[39] X. Ma, H. Gao, H. Xu, and M. Bian, "An IoT-based task scheduling optimization scheme considering the deadline and cost-aware scientific workflow for cloud computing," EURASIP Journal on Wireless Commun. and Networking, vol.2019, no.1, 2019, DOI: 10.1186/s13638-019-1557-3



Taeeun Kim received a master's degree from Soongsil University in Korea in 2007. His master's thesis was on safe node management protocols in ad hoc networks. He is a senior researcher at the Korea Internet & Security Agency and is working on the vulnerability of systems and networks. He is interested in information security, network security, and convergence security.

Hwankuk Kim received the B.S. and M.S. degrees in Computer Engineering from Korea Aerospace University in 1998 and 2000, and Ph.D. degrees in Korea University in 2017. He worked a researcher at ETRI from 2002 to 2006 and a cyber security research team manager at KISA from 2007 to 2020. Currently he is an assistant professor in the Sangmyung University. His research interests include 5G network security, software vulnerability analysis, IoT security and security data analysis.