



## 저작자표시-비영리-변경금지 2.0 대한민국

이용자는 아래의 조건을 따르는 경우에 한하여 자유롭게

- 이 저작물을 복제, 배포, 전송, 전시, 공연 및 방송할 수 있습니다.

다음과 같은 조건을 따라야 합니다:



저작자표시. 귀하는 원저작자를 표시하여야 합니다.



비영리. 귀하는 이 저작물을 영리 목적으로 이용할 수 없습니다.



변경금지. 귀하는 이 저작물을 개작, 변형 또는 가공할 수 없습니다.

- 귀하는, 이 저작물의 재이용이나 배포의 경우, 이 저작물에 적용된 이용허락조건을 명확하게 나타내어야 합니다.
- 저작권자로부터 별도의 허가를 받으면 이러한 조건들은 적용되지 않습니다.

저작권법에 따른 이용자의 권리는 위의 내용에 의하여 영향을 받지 않습니다.

이것은 [이용허락규약\(Legal Code\)](#)을 이해하기 쉽게 요약한 것입니다.

[Disclaimer](#)

국제학석사 학위논문

# A Study on the Malicious targeting of IoT Devices

IoT 장치의 악성 타겟팅에 관한 연구

박 민 진(Park, Min Jin)

국제학과(Department of International Studies)

정보법과학전공(Major in Legal Informatics & Forensic Science)

한림대학교 대학원

(Graduate School, Hallym University)

국제학석사 학위논문

# A Study on the Malicious targeting of IoT Devices

IoT 장치의 악성 타겟팅에 관한 연구

박 민 진(Park, Min Jin)

국제학과(Department of International Studies)

정보법과학전공(Major in Legal Informatics & Forensic Science)

한림대학교 대학원

(Graduate School, Hallym University)



Joshua I. James 교수지도

국제학 석사 학위논문

박민진의 석사 학위논문을 합격으로 판정함

2019 년    12 월    19 일

심사위원장    박 노 섭

---

심사위원    장 윤 식

---

심사위원    Joshua I. James

---

## 목차(TABLE OF CONTENTS)

List of Figures.....	vi
제 1 장 INTRODUCTION.....	1
1.1) Current Status.....	2
1.1.1 Number of IoT devices .....	2
1.1.2 Threats to the IoT .....	2
1.1.3 IoT Honeypot.....	4
1.2) Thesis Structure.....	5
제 2 장 RESEARCH QUESTIONS AND HYPOTHESES .....	8
2.1) Introduction .....	8
2.2) Thesis Statement.....	9
2.3) Research Questions.....	10
2.4) Hypotheses .....	11
2.5) Conclusions.....	11
제 3 장 BACKGROUND RESEARCH .....	13
3.1) Introduction .....	13
3.2) IoT .....	13

3.3) Honeypot .....	15
3.3.1 Types of honeypot.....	15
3.4) IoT Devices.....	17
3.5) IoT Security .....	18
3.6) IoT Honeypot .....	20
3.7) Conclusion .....	24
제 4 장 RELATED WORK OF IOT DEVICE SECURITY .....	25
4.1) Introduction .....	25
4.2) Hardware.....	25
4.2.1 Debug Pad .....	25
4.2.2 Hardware Backdoor .....	26
4.3) Software .....	27
4.3.1 OS (Operating System) .....	27
4.3.2. Firmware.....	28
4.3.3. Backdoor.....	29
4.4) Network.....	29
4.4.1 Sniffing .....	29
4.4.2 Port Scanning .....	30

4.5) Attack Data.....	30
4.5.1 DDOS (Denial-of-service attack).....	30
4.5.2 Kaspersky .....	31
4.6) Security Method .....	31
4.6.1 Guideline for security.....	31
4.6.2 Requirements.....	32
4.6.3 Platform .....	33
제 5 장 DEVICES USED FOR RESEARCH.....	35
5.1) Introduction .....	35
5.2) AI Smart Speaker .....	36
5.2.1 SKT Nugu .....	36
5.2.2 KT Giga Genie.....	36
5.2.3 Naver Clova .....	37
5.2.4 Google Google Home Mini.....	38
5.3) SSH (Secure Shell).....	39
5.4) FTP (File Transfer Protocol) .....	40
5.5) Expected Attack.....	41
5.5.1 TCP (Transmission Control Protocol).....	41

5.5.2 UDP (User Datagram Protocol).....	46
5.5.3 SSH (Secure Shell) .....	47
5.5.4 FTP (File Transfer Protocol).....	48
5.6) Conclusion .....	50
제 6 장 IOT HONEYPOT DATA COLLECTION METHODOLOGY .....	51
6.1) Introduction .....	51
6.2) Necessity of IoT Honeypot.....	51
6.3) The design of IoT Honeypot.....	52
6.4) Code of IoT Honeypot.....	53
6.5) IoT Honeypot Methodology.....	54
6.6) Devices.....	55
6.6.1 Giga Genie .....	55
6.6.2 Nugu .....	56
6.5.3 Clova.....	56
6.6.4 Google Home Mini .....	56
6.7) Procedures.....	58
6.8) Conclusion .....	59
제 7 장 DISCUSSION.....	60

7.1) Introduction .....	60
7.2) Quantitative Analysis.....	61
7.3) Conclusion .....	70
제 8 장 CONCLUSION .....	72
8.1) Introduction .....	72
8.2) Conclusion .....	73
8.3) Future Work.....	73
REFERENCE.....	75
APPENDIX.....	80
<Appendix 1> Code of IoT Honeypot.....	80
국문 초록.....	90
English Abstract .....	92

## List of Figures

Figure 1 Honeypot configuration [11] .....	15
Figure 2 SSH key exchange protocol .....	39
Figure 3 FTP configuration [45] .....	40
Figure 4 Diagram of IoT Honeypot setup .....	54
Figure 5 Profile files with open ports and protocol types.....	58
Figure 6 Most Attacked Devices.....	61
Figure 7 Attacked ports from profiles.....	62
Figure 8 Total number of attacks to device port.....	62
Figure 9 Attackers attempt to connect to the Google Home Mini. ....	64
Figure 10 Attackers keep sending pings to the IoT Honeypot.....	64
Figure 11 Common protocols of devices 1.....	65
Figure 12 Common protocols of devices 2.....	66
Figure 13 Attack Countries to device 1 .....	67
Figure 14 Attack Countries to device 2.....	68
Figure 15 Number of Attack countries per device .....	69

## 제1장 INTRODUCTION

The early Internet-of-Things (IoT) was connection-oriented, with monitoring and control at the center. It was a concept that built communication functions in various things and connected them to the Internet and made Internet-based communication between people-and-things and things-and-things. It can be described as a Machine to Machine communication (M2M). However, with the recent development of artificial intelligence (AI) technology [1], it is developing into intelligent IoT based on cognitive technology that learns, infers, and judges. Intelligent IoT uses AI and machine learning to interact with people and their surroundings with advanced capabilities through AI beyond programming execution. AI is leading the development of various intelligent objects such as autonomous vehicles, robots, and healthcare. It is also developing the capabilities of many things, including IoT, connected consumers, and industrial

systems. Representative examples of everyday life are smart homes and smart home appliances.

## **1.1) Current Status**

### **1.1.1 Number of IoT devices**

There are various devices such as air conditioners, refrigerators, lights. Among them, the AI smart speaker is easily obtained by anyone and is widely used in real life. The Ministry of Science and ICT announced [2] the fourth industrial revolution indicator for each sector, including AI speakers. The number of IoT device connections was 1400 million in December 2017, and it increased to 1865 million in December 2018. Also, the number of AI speaker sales increased from 200 million in March 2018 to 412 million in March 2019. At the end of 2019, it will be expected to 800 million.

### **1.1.2 Threats to the IoT**

IoT is convenient and easy to manage through one set. Sensors in IoT devices collect data, including sensitive data [3]. However, numerous IoT devices connected to the

Internet, such as IP cameras, AI speakers, and smart appliances, are exposed to security threats such as hacking and Distributed Denial of Service (DDoS) attacks. It is because manufacturers and users are less aware of IoT security. In a 2019 article [4], AI experts found a severe security flaw in AI speakers sold to consumers. If a user talks to the AI speaker controlled by a hacker, the hacker can listen to the conversation in real-time. In a more recent case [5], researchers in the US and Japan found a way to hack AI smart speakers using lasers. Encode voice commands and send them as laser light, the speaker responds. This is the principle that when light hits a diaphragm with a built-in speaker, the diaphragm vibrates and performs is recognized as user speech. This method is possible for up to 110 meters away. The device used for the experiment is 400 dollars, and anyone with malicious motives can attack AI speakers outside the home.

Also, like Mirai-botnet, which occurred in 2016, there is still a threat of massive DDoS attacks through hacking of IoT devices. The proliferation of industries based on IoT technologies, such as automated vehicles and smart cities, increases the risk of

these cyberattacks leading to life-threatening physical damage. It applies not only to homes but also to businesses, which is even more dangerous. In the case of a company, they use a centralized network connection. Thus, once an attack is made, critical business systems may go down. The IoT is the bridge between the real world and the digital world, so if the hacker attacks people in the digital world, it can affect the real world.

#### 1.1.3 IoT Honeypot

A honeypot is a system of application programs and data designed to lure hackers and attackers. It looks like a real system, but it can be designed to look like there are exploitable vulnerabilities to monitor the attacker's actions. It is already actively used overseas as a program that can attract attackers to fake the IoT environment to collect types and methods of attack and prepare countermeasures. Currently, studies using honeypots are being conducted in Korea [6], but there are hardly any studies that combine them with IoT. Also, in the case of IP cameras, there is data about how much of the attack was carried out. However, the AI speaker is only reporting vulnerabilities

to hack but has no data on accurately how much of the attack has been performed. Therefore, in the thesis, we will research about the attack on the IoT using AI smart speakers. A representative device of IoT echo systems, as well as a honeypot that examines the behavior of attackers.

## **1.2) Thesis Structure**

In Chapter 2, as the definition of the Internet of Things still varies, there are no exact definitions. We look at the definition of the IoT used in this paper. Also, when new technologies are released, vulnerabilities are also found, as is the IoT. With the focus on IoT, attacks appear to be more common than against other devices. Thus, research questions and hypotheses to see if they are attacked more than other devices.

In Chapter 3, talks about a brief explanation of the types of IoT and honeypot. Then we research the background of IoT devices, IoT security, and IoT honeypot. It talks about the stance of domestic IoT devices, overseas interest in IoT security, and the

actual situation of domestic IoT security. Also, it introduces cases of using a honeypot to analyze IoT attacks.

In Chapter 4, talks about the related work of IoT device security. As attention has focused on the Internet of Things, and as security advances have been made, security vulnerabilities have also been found. Through the chapter, we will see the types of security vulnerabilities that IoT device would have and the possible attacks can be done to IoT device.

In Chapter 5, as the Internet of Things has evolved, many IoT devices have been released. Among them, IoT devices studied in this paper will be described, as well as SSH and FTP protocols. Also, since IoT uses protocols, we will look at the types of attacks the protocol can receive.

In Chapter 6, explains the necessity of an IoT Honeypot, why we develop our own IoT honeypot and how the honeypot is designed and methodology to use the honeypot in the research as well as, expected attacks through the honeypot will be described.

In Chapter 7, collect the necessary data using the IoT Honeypot described in the previous chapter. Then, answer the proposed research questions and support or deny the hypotheses mentioned above by analyzing the data collected over a period of time.

Finally, in Chapter 8, we conclude with the results obtained from the research. Then, discuss what other additional research should follow in the future.

## 제 2 장 RESEARCH QUESTIONS AND HYPOTHESES

### 2.1) Introduction

In this chapter, we will research questions under investigation in the thesis study.

There is no agreed definition of the Internet of Things (IoT). IoT Agenda [7] defines the Internet of Things as a system that has a unique identifier and can send and receive data to computers, machines, connected through a network without a human being. IBM [8] defines it as connecting all devices with an Internet connection that can be turned on and off. SAP [9] defines sensors and APIs as a network of physical entities that exchange data over the Internet. In this paper, IoT can be understood through this definition which is, a technology that enables communication, sharing, and collection of information with each other through networks and the Internet such as people, objects, and spaces. Honeypots are primarily traps for malicious hackers. Like honey jars that lure bees, the goal is to attract hackers into honeypots and, usually, collecting useful

information. Honeypots are computers or computer systems that simulate the targets of probable cyber-attacks. By design, users use vulnerabilities in honeypots. For example, an administrator connects a honeypot on the network, adds data, and waits as if the computer has sensitive information. The admin can detect an attacker attempting to crack the machine. Also, hackers attacking honeypots to exploit vulnerabilities can reveal the hacking path and hacking techniques to trace back the hacker's information or the location of hacking.

## **2.2) Thesis Statement**

Before the Internet of Things developed, there were many attacks on other devices. However, with the rapid development of the IoT and a significant impact on people's lives, attackers are started to give focus on the IoT devices, and the number of attacks on vulnerabilities in the IoT seems to be far higher than the number of attacks on other devices. Thus, in this paper, we are focusing on attacks against IoT devices with this statement.

IoT devices are the subject of attacks more than other devices.

Through the thesis statement, we will derive the research questions and hypotheses. Therefore, we will see if the thesis statement indeed applies to reality.

## **2.3) Research Questions**

As a result of previous research, each time a new technology is released, the attack is also found as well. So, we started to look for various IoT devices. Among them, there was interest in speakers that are closely related to human life. Smart speakers equipped with Artificial Intelligence (AI) can communicate with people and analyze people's words to perform tasks. Machine learning also enables self-learning and inference, making it smart with data accumulated through continuous use. However, there is a problem because it collects such diverse and vast data. In a recent news article, the AI smart speaker was hacked, and the ability to listen to people was changed to an eavesdropping device. As IoT devices are becoming more common, the number of attacks on them seems to increase. As a result, attention has focused on IoT devices, raising questions about whether IoT devices receive more attacks than other devices. So, while researching AI smart speakers, various questions arose.

- Which IoT device gets attacked most?

- Which ports in profiles get more attacked?
- What types of data are the attackers sending?
- What protocol is standard per device/total?
- Where the IP addresses come from? To what device?
- Attack countries per device
- Are the attackers focusing on specific devices? Are they opportunistic or on purpose?

## 2.4) Hypotheses

Two hypotheses can be tested by answering research questions. one is 'Google Home Mini will get attacks more than other speakers because of the Market share,' and the other is 'Attackers are not focusing on specific devices.'

## 2.5) Conclusions

Based on the definitions of various IoTs, we defined the IoT to be used in the paper. In our doubts from previous researches, we created a thesis statement about whether an IoT device gets more attacked than other devices. Next, IoT honeypots and data analytics are used to validate research questions and hypotheses. Tested research

questions and hypotheses are used to determine whether the thesis statement is indeed being made in reality

## 제 3 장 BACKGROUND RESEARCH

### 3.1) Introduction

IoT devices offer a variety of conveniences to humans. In order to provide services to humans, IoT collects and processes sensitive data such as human status and information, and in the case of smart homes, it is connected to various devices to simplify control. However, its convenience comes with risks. IoT security is as essential as dealing with sensitive data, and prior research has been conducted to find out its vulnerability. In this chapter, we describe so far research related to IoT attacks. Among them, the honeypot is a system that can analyze and predict what attacks are coming into IoT devices and has already been prior research for analyzing attacks using honeypots.

### 3.2) IoT

There are various kinds of IoT [10], which can be classified according to data transmission, the behavior of things, and Artificial Intelligence (AI). In the case of the

first data transmission and reception, it may be divided into a data transmission object collecting and exporting data, a data reception object receiving data from outside, and a data transmission and reception object. Second, the behavior of things is classified as fluidity. A way that things move through programming, like a cleaning robot, a way that things fix in the place and work, like a smart fridge and AI speakers, and another way is that things like a smartwatch, where people can carry it. Lastly, in the case of Artificial Intelligence, it can be defined into two which are, the movement of the things is not only through the programming but itself analyze, process and determine the collected data and the things without AI and move as programmed by a human.

### 3.3) Honeypot

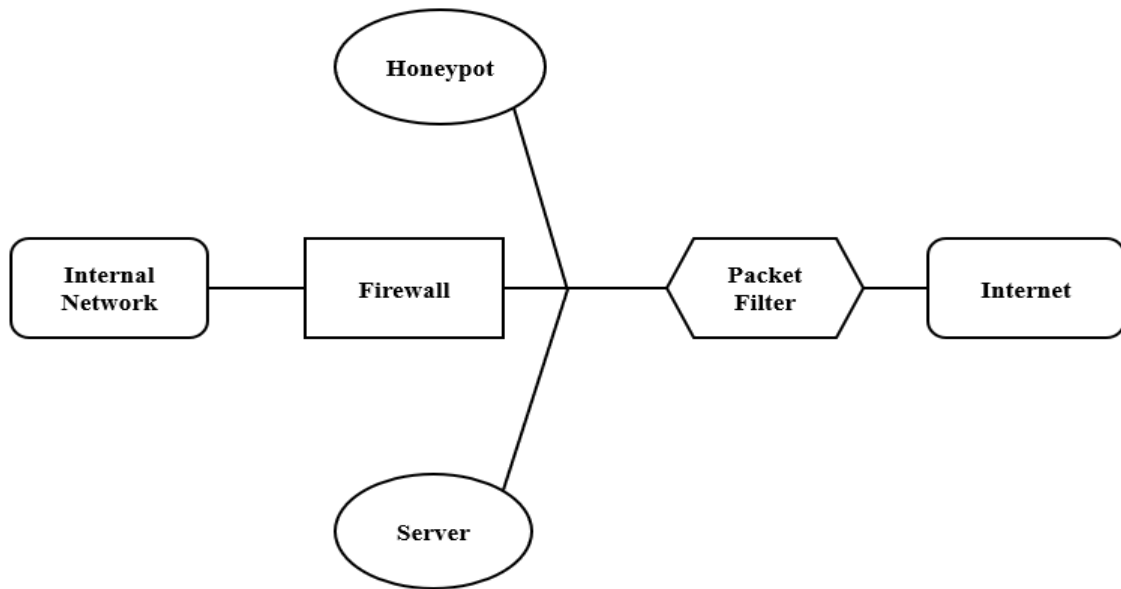


Figure 1 Honeypot configuration [11]

There is an advanced type of honeypot and honeynet. Usually, if a honeypot is a single system for getting hackers' information, then honeynet means a network that includes honeypots [12]. Figure 1 is a standard configuration of honeypot. It is placed above the firewall, emulates an internal network, and gathering the data.

#### 3.3.1 Types of honeypot

Honeypot types [13] can be divided into production honeypot and research honeypot. Production honeypots are typically used to protect organizations such as companies. It

is installed in the company network to be used for overall security and to mitigate risk. Research honeypots are used to study attacker tactics and techniques or to gather information. For example, when a system is compromised, such as a virus, it can collect information about how the attacker is attacking. Then, the honeypot can be classified [12] into two types, low-interaction and high-interaction. Low-interaction honeypots simulate the type of attacker frequently attacks. It is relatively easy to use because it uses fewer resources and does not require much code. High-interaction honeypots are more complicated to install than low-interaction honeypots, and it behaves like a real system. With virtual machines, it can run multiple on one physical computer. It is also easy to recover a system compromised by an attack.

#### 3.3.1.1 Dynamic Honeypot

Also, there is another type of honeypot, called Dynamic Honeypot. Kuwatly et al. [14] introduces the design for intrusion detection system using Dynamic Honeypot. They give a new way of the Dynamic Honeypot to implement in a real network system.

Budiarto et al. [15] discusses about making the honeypots in a simple way and talks about the configuration of Dynamic Honeypot. They introduce the Dynamic Honeypot as the honeypot that can work by plugging in to the network. It decides how they will deploy the environment.

### **3.4) IoT Devices**

A 2018 article [16] describes how vulnerable IoT devices are in Korea. As the number of electronic devices connected to the network increases, hacking aims to steal or harm personal information. In the case of smart speakers, hacker hacks and sends a private conversation to third parties. Internet of things and artificial intelligence devices are targeted for hacking because they have lower computing power than PCs, and this makes hackers easier to hack. AI and IoT are popularized, smart TVs, smart refrigerators, and automated cars on the rise. It means that the damage is expected to spread further.

A 2019 article [17] talks about the National Assembly's audit of the National Science and Technology Information and Telecommunications Broadcasting and Communications Commission pointed out the lack of measures to leak personal

information of Naver, Kakao, and three telecommunications, for AI devices including IoT speakers. Since 2017, the IoT security certification system has been implemented from the perspective of security internalization. However, out of the 17 cases, there are eight light grades and nine primary grades, and 0 international standard security grades. There are 8 million AI speakers, IoT-connected devices, but they criticized for no security certification. As the article describes, IoT devices are increasing, but there is a lack of countermeasures against devices.

### **3.5) IoT Security**

The Trend discussed [18] the five most active cybercrime underground communities in the world. The Russian community took the most live action on IoT-related attacks. Cybercriminals bought and sold the discovered vulnerabilities related to IoT devices on the website. The Portuguese were focused on the router infection called "KL DNS." They were looking for an opportunity to make attacks similar to router mass infection in Brazil in 2018. Next, the English hacking community talked about the specific vulnerability of Netgear routers and had actual codes for exploitation. Also, showing some interest in connected printers. For the Arabic community, they expressed their

interest by sharing the latest news about discoveries of IoT vulnerabilities. Lastly, the Spanish community showed interest in not protected and not authenticated devices to find the entrance for new attacks. For example, they are using Google dork to look for industrial refrigerators that are not protected. As the IoT industry is getting bigger, the attacker's interests are moving into IoT a lot. Thus, to prevent damage to it, some actions are needed.

According to the Korea Internet & Security Agency (KISA) [19], in 2019, reports of IoT-related security vulnerabilities are increasing every year. The number of reports is rapidly increasing from 6 in 2014 to 130 in 2015, 362 in 2016, 347 in 2017, and 387 in 2018. In the first half of 2019, 188 cases were received. The total number of reported IoT security vulnerabilities in 2019 is expected to increase more than the previous year. There are various types of IoT security vulnerabilities reported to KISA. First, exploit vulnerabilities that could allow hackers to gain administrative privileges. Second, vulnerabilities that can bypass security programs. Third, 'Authentication bypass' to access the authority of the IoT administrator page. Fourth, vulnerability to information leakage collected on IoT devices, and lastly, vulnerability to insert malicious commands

into IoT devices to execute malicious code were reported. KISA operates the 'Software (SW) New Vulnerability Reporting Reward', which is notified of IoT security vulnerabilities. At the same time, KISA has been implementing the 'IoT security certification system' to evaluate the security stability of IoT devices since December 2017, but the results are poor. According to the KT Institute for Economic Management, domestic IoT hacking damages are expected to reach 26.70 trillion won in 2030. It means is that it is good to operate the IoT reporting reward system, but it is urgent to prepare a practical plan through simulation and test to the attacks.

### **3.6) IoT Honeypot**

Gandhi et al. [20] aimed to protect the IoT environment by proposing HIoT POT (Honeypot for IoT environment). HIoT POT installed a honeypot and user database on Raspberry Pi and collected data. If the user DB in Raspberry Pi matched with the accessing user, it connected to the real IoT environment. If it did not match, it was considered as an intruder and connected to a fake IoT environment. HIoT POT collected logs and chat details of intruders connected to fake IoT environments. Also, HIoT POT sent a warning about the user to the real IoT environment. The collected logs showed

the intruder entered methodology, and it could be used for further research and as forensic evidence. The paper focused on detecting intruders at IoT devices. It would be helpful if there were more explanations of the result, such as what is the meaning of a graph. Also, detection alone is not enough to create a truly safe and secure IoT environment. A protection plan is also needed.

Anirudh et al. [21] conducted some tests to protect an IoT system from the Denial-of-Service (Dos) by using Honeypot. They proposed two models with scenarios to compare how the Honeypot is useful. The first scenario was using the Intrusion Detection System (IDS). If IDS found an oddity in the client's data, it sent it to a honeypot and collected log information about the attacker. Also, the collected data was managed in a database. The secondary scenario was that there was a log collected in advance, unlike the previous scenario. When a request came to IDS, it checked to see if it matches the client of the data in the log. If the data does not match, block it; if it does pass it. Through the scenarios, using honeypot was more effective than without it. It would be good if the paper talks about what happens when these models run on real cases.

Yin et al. [22] found a significant increase in telnet-based attacks against IoT devices. They implemented a new honeypot called IoT POT that copied IoT devices and captured telnet-based intrusions from several attacks. Additionally, proposed IoT BOX to process captured malware on different CPU architectures to more analysis of threats. Through these analyses, they discovered at least four DDOS malware targeted to IoT devices. This paper implemented the first honeypot for IoT devices based on telnet and tried to cover most of the CPU architectures.

Haris [23] analyzed the Mirai-based attack that happens to the Internet of Things (IoT) by proposing a multi-component solution. They implemented IoT honeypot, which had multi-component that worked with telnet traffic to handle the Mirai attack. The front-end component attracted the attacker's attention by interacting and answering the attacker's input. While the front-end component interacted with the attacker, the back-end component got the encrypted data that was captured and decrypted into readable form. Then, inform the user and saved it forever. After the test, they found out that Mirai did not target device vulnerability. It looked for the weak and default passwords that had never been changed seen it was operated. It would be useful if the

paper uses actual IoT devices and mentions what kind of passwords were vulnerable to this kind of attack.

Meng [24] implemented ThingPot by mimicking a Philips Hue smart lighting system - wireless LED lights bulb and wireless bridge - and with XMPP and REST API. This implementation was focused on the whole IoT platform. Besides, they have offered a Proof-of-Concept (PoC) and provided open-source code for it. The test had run for about one and a half months. During the analysis, the captured data presented that not many attackers were activated on XMPP. It indicated few things that XMPP made attackers hard to get to the device and its platform, or they were not interested in it yet, or the attacker's focus was not in the ThingPot but XMPP server.

On the other hand, attackers showed some interest in REST. They were trying to gain some data about the device and get control of it. The paper focused only on one IoT device but may try the same test to the different devices to have more concrete results.

### 3.7) Conclusion

We talked about the prior research related to the IoT and honeypot. Although there have been several prior researches that deal with sensitive information, there are still many risks. In the next chapter, we will look at the research that has been carried out for the attacks and security that IoT devices can receive.

## 제 4 장 RELATED WORK OF IOT DEVICE SECURITY

### 4.1) Introduction

The previous chapter talked about vulnerabilities in IoT devices, and this chapter talks about the security of IoT devices. With the development of IT technology and the development of IoT technology, the types of devices and services have increased exponentially, and so have the security threats. This chapter will cover what all kinds of possible attacks could be used to IoT, including against computers that can be turned into the IoT on different sides.

### 4.2) Hardware

#### 4.2.1 Debug Pad

Debug is the task or program that, at the end of the program's development, detects the error, and identifies its cause. The debug pad is a pad attached to the device for debugging. When accessed through the pad, it may be possible to access the system of

the device without any authentication. Thus, it makes attackers easy to take control of the device. Barnes [25] conducted a test using a debug pad on Amazon echo, which is an AI speaker. The prior work was that boot echo with a debug pad attached to the bottom of the device. They extended the work from previous work by taking the remote root shell access and was able to obtain the remote control on microphones. By conducting UART, they could see the device boot and configuration. Then, using some command lines, they examined the file system. After that, with the scripts that they wrote, they were able to figure out the interaction between the audio buffers and remote the service. However, since 2017, Amazon changed the structure of the pad on the mainboard to avoid external booting. Thus, this kind of physical vulnerability is available for the 2015 and 2016 version.

#### 4.2.2 Hardware Backdoor

The paper [26] talks about Smart Nest Thermostat, which can be attacked physically through the USB port. The Nest has security for software but hardware. If the attacker has a chance of physical access, it takes only 15 seconds to control the Nest. By

pressing the power button, the device switches to developer mode, inserting a USB drive during reset and loading user firmware that official Nest does not specify. Although a device is infected with a virus, there is no problem in using it, so the user cannot know even if the data on the device is leaked.

## **4.3) Software**

### **4.3.1 OS (Operating System)**

An article in 2016 [27] talks about the Linux kernel system, which can affect a lot to IoT devices. Linux kernel vulnerabilities discovered in 2016 include CVE-2016-0728, CVE-2015-1805, and CVE-2016-5195, called Dirty Cow. Since Android, iOS, and Mac OS are based on Linux, it can be said that most IoT devices have a Linux based OS system. Among other things, CVE-2015-1805, which is Dirty Cow, is dangerous enough to affect 97% of Android devices.

### 4.3.2. Firmware

#### 4.3.2.1 Different types of vulnerabilities and attacks

The article in 2019 [28] introduces eight different vulnerabilities and attacks that can be happened in firmware. first, unauthorized access that an attacker can easily exploit. Second, weak authentication with not strong encryption algorithms. Third, a hidden backdoor that makes the attacker access to the device easily. Forth password hash that is hard for users to change. Fifth, an encryption key that is proved inappropriate to use. Sixth, buffer overflow - give control to an attacker by using insecure coding. Seventh, open-source code - easy to be a target to attackers if there is no regular update, and eighth, debugging service - allows the attacker's internal access through the device.

#### 4.3.2.2 Control-Hijacking

The paper [29] briefly presents about control-hijacking vulnerabilities of IoT firmware that have been widely spread recently and some related work. Also, using metrics, they classify the recent discovered control-hijacking vulnerabilities.

### 4.3.3. Backdoor

In the 2019 article [30], attackers have infected backdoors with live update utilities installed on ASUS new versions of computers. Through this, attackers gained access and identified the targets by using about 600 MAC addresses that they previously had. The attackers then downloaded the malware to a running C&C server when a specific MAC address was found. Then they took ASUS's digital certificates and distributed them to the official update server. The attack affected users who activated ASUS live update utility. In the case of backdoors, manufacturers often plant them for maintenance purposes, such as a network. Therefore, it can be applied not only to computers but also applies to IoT devices with backdoors.

## 4.4) Network

### 4.4.1 Sniffing

In this article [31], it talks about ZigBee-sniffing drone. The Praetorian, which is an information security company, experimented with a drone that can detect the devices

connected to the Internet. Through the experiment, they found that the drone revealed the device's security settings, the manufacturer, and where the device is used, such as commercial or residential.

#### 4.4.2 Port Scanning

In the 2018 article [32], they picked port scanning as one of the network attack types. Port scanning is an attack that precedes the vulnerability of the application being used by the target. It checks the port address to determine the type of service. Major port scan methods include TCP scans, SYN scan, and ICMP message scan. Since all devices send receive data through the port, port scanning can be regarded as all IoT devices. For example, one of the AI speakers, Google Home Mini, has five fixed TCP open ports.

### 4.5) Attack Data

#### 4.5.1 DDOS (Denial-of-service attack)

Igloosecurity [33] talks about one of the DDoS attacks, which is Mirai-botnet. An attacker infected several vulnerable IoT devices with the malware Mirai, which

automatically searched for other vulnerable devices on the Internet and made them act as bots for DDoS attacks. It attacked the Dyn server, a major US Internet hosting company. Dyn caused a series of disruptions at major sites such as GitHub, Twitter, Netflix, and the New York Times that were receiving DNS services.

#### 4.5.2 Kaspersky

After building more than 50 honeypots around the world, Kaspersky [34] has detected 105 million attacks on IoT devices with 276,000 IP addresses. Kaspersky's IoT: Malware Story report contains data on the number of cyberattacks performed over time using honeypot data, the types of attacks used, and where the attacks occurred.

### 4.6) Security Method

#### 4.6.1 Guideline for security

Open Web Application Security Project (OWASP) [35] published principles of IoT security and guidance. Principles of IoT security were written in 2016 and covered the overall IoT system, components, and ecosystem. The IoT security guideline, written in

2017, describes three security guidelines that manufacturers, developers, and consumers must follow to improve the security of IoT products. The three different parts have in stock about IoT device security is restricting physical access. An example is the debug pad at the bottom of the Amazon echo. In the case of the Amazon echo, people could gain access to data and permissions after physical access through the debug pad. Thus, OWASP recommends disabling unnecessary or unused physical ports such as debug pads or USB ports.

#### 4.6.2 Requirements

This paper [36] argues that security features must be reflected from the requirements analysis stage, which is the early stage of development. Also, based on three essential features of the IoT environment: heterogeneity, resource constraints, and dynamic environment, the paper analyzed the security requirements for IoT in six aspects of the IoT environment - IoT networks, IoT clouds, IoT users, IoT attackers, IoT services, and IoT platforms.

#### 4.6.3 Platform

The article [37] talks about the IoT platform. The IoT platform is an integrated service that provides the elements needed to bring physical objects online. IoT platforms can be classified into four categories which are, end-to-end platforms, connectivity platforms, cloud platforms, and data platforms. To efficiently build and manage the IoT environment, companies are actively participating in building the IoT platform. However, in terms of security, there are still a few guidelines that can be found other than the security embedded in the IoT platform itself or the protection of the enterprise itself. Therefore, in this article, gives a guide for choosing an IoT platform.

GreenZone Security [38] introduced an end-to-end security platform that can encompass devices, networks, and service platforms in an IoT environment. It is equipped with several security technologies to optimize the IoT device environment, which is characterized by ultra-lightweight, low power, and low performance. They described that it plays an essential role in increasing the IoT security rate.

## 4.7) Conclusion

This chapter covered IoT device security. AS a computer connects to the Internet, IoT devices connect to the Internet. Thus, attacks that the computer can receive, also can be applied to IoT devices. In this paper, we will examine how much of these vulnerabilities and security problems are concentrated on IoT devices. In the next chapter, we will examine the devices used in the paper.

## 제 5 장 DEVICES USED FOR RESEARCH

### 5.1) Introduction

In this chapter, the types of devices used for the research will be described. Various IoT devices exist. Nowadays, smart home appliances such as smart refrigerators, smart TVs, and smart light are easily accessible to people. As a result, many attacks are aimed at people who do not have much security awareness. Therefore, in this paper, we study using AI smart speaker, one of the popular smart home products. As the IoT devices connect to the Internet, it uses TCP and UDP protocols to send and receive the data. Thus, we use two popular protocols, SSH and FTP, for comparison and to verify the data. In addition, we talk about the types of attacks that IoT devices and protocols can possibly get.

## 5.2) AI Smart Speaker

### 5.2.1 SKT Nugu

Nugu [39] is Korea's first AI speaker released by SK Telecom in September 2016. The exact model name is NU100, which can be operated by voice control, power, volume, mute, Bluetooth, voice recognition button, and smartphone application. To use Nugu, a user needs to download the Nugu application on the smartphone. After installing the application, the user can use the smartphone and Wi-Fi connection. In addition to the timer and alarm in Nugu itself, it can also be linked to Melon, Google, and Smart Home. After logging in, the user is logged in to the application continuously unless the user logs out.

### 5.2.2 KT Giga Genie

Giga Genie [40] is an AI speaker launched in 2017 by KT, a telecom company like SKT. The exact model name is KT Giga Genie CT1100. Unlike other speakers, set-top boxes and artificial intelligence are combined. It can be used instead of KT's existing set-top

box, which provides IPTV service, Olleh TV. It can also be used as a home cam by attaching a dedicated camera. It has a remote control and HDMI port so that it can be connected to a display and used as a TV. Other features like search, weather, alarm, and music are like those of existing AI speakers and can control the TV program. To use Giga Genie, the user must log in to the Giga Genie application. In the case of Giga Genie application, KT ID login, smartphone login, Kakao Talk login, Naver login, and Facebook login are available. If the user logs in once, the login is maintained unless the user logs out.

### 5.2.3 Naver Clova

Naver is one of the most used search engines in Korea. Clova [41] was launched in May 2017 as Naver's AI platform. Since then, Naver has introduced a smart speaker equipped with Clova, and various models have been released. The product used in this paper is Friends mini Minions NL-S22KR (Bob). Clova also needs to download the Naver Clova application for the initial configuration to use. Similar to the existing AI speakers, it has essential functions such as news, music, and alarm, and can be linked with other

smart services such as LG SmartThinG. The difference with other speakers is that it is portable, so the user does not have to connect the power supply if the battery is charged.

#### 5.2.4 Google Google Home Mini

The Google Home Mini (GHM) [42] is a smaller version of Google Home that is released by Google in October 2017 and released in 2018 in Korea. It uses Google Assistant as an AI platform. Basic functions such as music, news, and weather can be used, and the smart home service of Korean companies can be synced. Google Home Mini [43] supports different languages, including Korean, English, German, and French. The speaker supports multi-language mode, so when a user asks a question in Korean, it answers in Korean, and if the user asks in English, it answers in English. Also, if the user uses a Google account, the user can receive personal information in the user's account, such as schedules. To use GHM, the user needs the Google Home application, and the user will be logged in unless the user logs out.

### 5.3) SSH (Secure Shell)

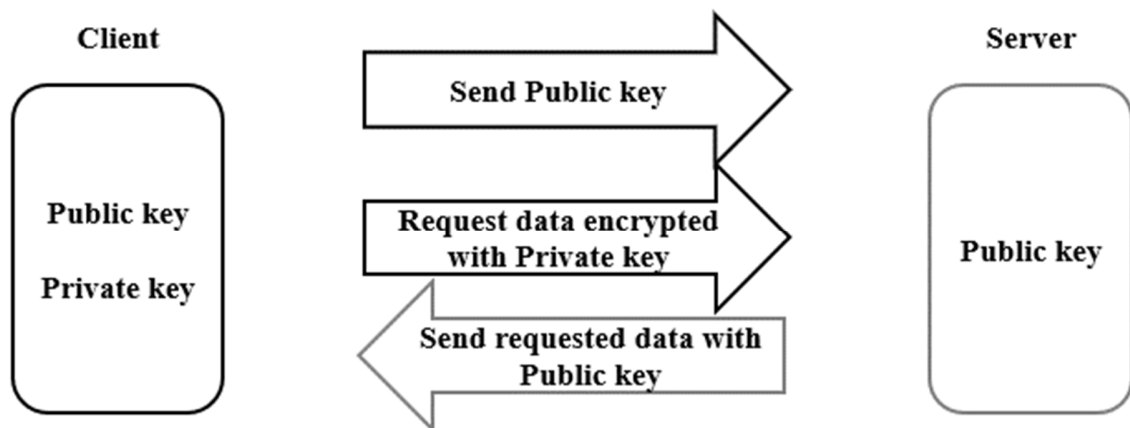


Figure 2 SSH key exchange protocol

SSH stands for Secure Shell. One of the network protocols, a protocol for secure communication when communicating with other computers connected to a network. Because all data over SSH is encrypted and traffic is compressed, the user gets fast transfers. It is more secure than the same network protocols, Telnet, and FTP. It is mainly used for data transmission and remote control. SSH [44] uses asymmetric cryptography (different keys used for encryption and decryption) that authenticate through public and private keys. Moreover, SSH's default port is 22. The way SSH works are shown in Figure 2. First, create a public and private key on the client-side and send the public key to the server. The data that the client wants to request is then encrypted

with the client's private key. Then, the server decrypts the data with the public key and encrypts the information the client wants with the public key. Even if a hacker intercepts the public key in the middle, there is no private key, so the hacker cannot decrypt the data that the server sends. At this point, the critical information is sent from the server to the client. Thus, SSH is safe by making hackers hard to decrypt.

#### 5.4) FTP (File Transfer Protocol)

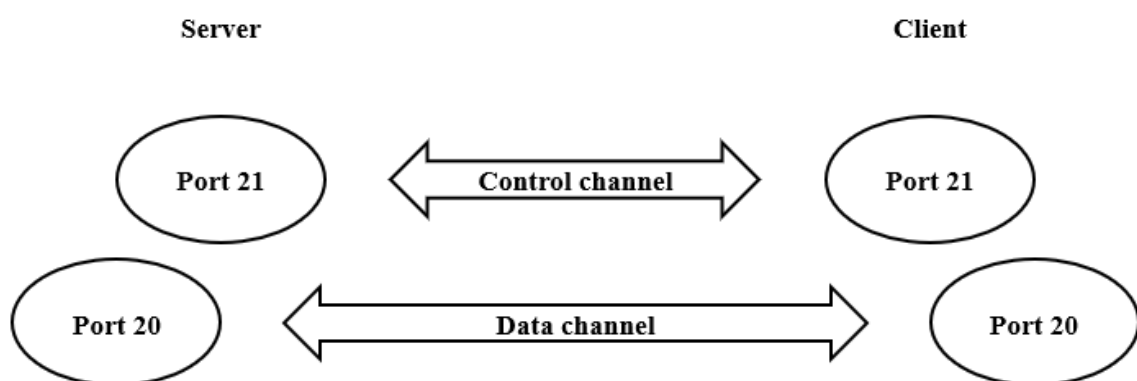


Figure 3 FTP configuration [45]

FTP stands for File Transfer Protocol. It is a TCP/IP protocol for sending and receiving files literally. Suitable for sending and receiving large files over a network. FTP [45] is a command-based protocol. Figure 3 shows the configuration of FTP. It has a control channel (port 21) for sending FTP commands and a data channel (port 20) for

transferring files between the client and server. Corresponding channels are created between the server providing FTP service and the connecting client. The FTP [46] server requires a user account and password to connect. This information is sent and received through the control channel. The actual file transmission and reception takes place over the data channel. FTP can also be used with a web browser or Windows Explorer, but it is more convenient to use an FTP-only client program. Because FTP client programs must send and receive many files in a row, they maintain a connection with the server by sending a persistent response message.

## **5.5) Expected Attack**

### **5.5.1 TCP (Transmission Control Protocol)**

#### **5.5.1.1 SYN Flooding**

SYN Flooding [47] is an attack that causes failure by sending an attacker a large number of SYN packets to the server to fill the server's queue and ignore new client connection requests. In more detail, in the SYN packet transfer phase, which is the first

phase of the 3-Way Handshaking - before an application program that communicates using TCP/IP protocol transmits data-, it establishes a session with the counterpart computer in advance. In order to guarantee the correct transmission, an attacker generates a large number of SYN packets and forwards them to the server. Then the server's backlog queue, which is used to accept TCP connection requests, becomes full, which results in a denial of service condition that causes subsequent connection requests to be ignored. If only the SYN packet is sent and the ACK packet, which is a response to SYN-ACK, is not sent, it is in Half Open mode for 75 seconds, and it continues to send SYN packet to fill the Backlog Queue and no longer receive new TCP connection.

#### 5.5.1.2 Tsunami SYN Flooding

While traditional SYN Flooding attacks, generate 40-60 bytes of traffic per packet, Tsunami SYN Flooding [48] attacks by adding and generating packet traffic with a size of 1000 bytes per packet. This type of DDoS attack uses the TCP protocol rather than UDP.

#### 5.5.1.3 TCP Connection Flooding

TCP Connection Flooding [47] is a type of attack that causes service overload by excessively triggering the TCP 3-Way Handshake process. The server receiving the attack traffic keeps trying to connect the ordinary TCP session to the session. It depletes the session processing resources of the server performing the service so that the regular session connection can no longer be performed. As a result, users who normally access can no longer access the service. It can be divided into three categories.

- DDoS Attacks that maintain TCP session connections
- DDoS Attacks repeating TCP session connection/disconnections
- DDoS Attacks that sends out traffic that looks like a normal transaction after a TCP session connection

#### 5.5.1.4 HTTP GET Flooding

In the case of the TCP Connection Flooding described above, normal transactions do not occur after the TCP 3-Way Handshake process. In contrast, HTTP GET Flooding [48] is a DDoS attack technique in which an additional standard transaction occurs after the TCP 3-Way Handshake process. Since the server receiving the attack traffic continuously requests the standard HTTP Get request along with the regular TCP session, the server performing the service must perform not only necessary TCP session processing but also HTTP request processing. It may cause an overloading of the HTTP processing module.

#### 5.5.1.5 SlowLoris

The attacker [49] maintains an open connection by requesting abnormal (incomplete) header value to a server after connecting to the target server and establishing a regular session. The standard header completes with 0d0a (CRLF), but SlowLoris sends an abnormal header value of 0d. The server determines that the transmission of the

header has not been completed and continues to maintain the connection. The steps are below:

- Send a GET request after a session is connected
- Send an incomplete request and keep an open connection
- The server will wait for the header
- The server enters DOS state depending on the number of connections
- Do not end because the header of the request is [0d0a0d0a]

#### 5.5.1.6 SlowRead

It [48] is a type of HTTP attack that delays TCP connection by slowly reading a response by manipulating buffer size and TCP window size. Take advantage of the fact that the webserver does not limit connection delays. The difference between SlowLoris and SlowRead is that it is holding the session longer, sending the HTTP request correctly, and reading the response slowly, rather than delaying the request. It is a way to continually delay a TCP connection in the data flow by manipulating the value of the TCP window size and receiving '0' or small data. The attacker and the target server

occupy connection support until the data transmission is completed. If this process occurs a lot, the connection resources of the target server are exhausted, and the service is denied.

## 5.5.2 UDP (User Datagram Protocol)

### 5.5.2.1 UDP Flooding

UDP Flooding [50] is a type of DoS attack in which a large number of UDP packets are sent to a user to make it impossible to use the standard service. Since UDP packets use spoofed IPs and ports, it is difficult to block them using IP filters. Because it consumes network bandwidth, all services of users, not specific services, are disabled, and there is no need for specific ports to be open. As with any flooding attack, there are no singularities or patterns in the packet itself, making it difficult to block. In the case of the UDP attack, unlike the SYN flood, the purpose is to consume network bandwidth. Therefore, since a single host is not valid, so the attack is configured by DDoS.

#### 5.5.2.2 Valve Source Engine Flooding

Valve source engine flooding [48] is UDP (amplification) attacks that are used to consume resources available to the server. The attack is designed to send TSource engine query requests to the game server, which means that the server cannot handle all the requests and handles many of the requests that make the game denial of service. This type of attack only applies to the gamers market.

#### 5.5.3 SSH (Secure Shell)

##### 5.5.3.1 Brute Force

It [51] is an attack that attempts to access SSH by indiscriminate ID and password substitution. It is a one-dimensional and simple assignment attack, but it is a compelling and intuitive way to penetrate all the possible keys until it finds the key. Similarly, there is a dictionary attack that matches words in a dictionary file, and a hybrid attack that adds numbers or special characters to words in a dictionary attack.

#### 5.5.3.2 GoScanSSH

GoScanSSH [51] is a malicious code that spreads after infecting an SSH server on a Linux system that is exposed online. The malware sets Raspberry Pi, Open Embedded Linux, OpenLEEC, and Huawei as major target systems to perform random attacks with about 7,000 account/password combinations. The malicious code randomly generates IPs, checks specific IP bands and domains, and scans additional infection targets except for government and military organizations. It performs random assignment attacks on port 22 of randomly generated IP and transmits system and login related information to the C2 server. When sending information, they use the Tor Web Proxy service to make it difficult to track. After successful login, upload, and run GoScanSSH malware to perform additional attacks.

#### 5.5.4 FTP (File Transfer Protocol)

FTP has the vulnerabilities that FTP does not encrypt user authentication information and vulnerability that exploits the characteristics of the FTP protocol.

#### 5.5.4.1 Bounce Attack

FTP Bounce attack [52] is a method of exploiting loopholes in the FTP protocol structure. In more detail, it is an attack that exploits structural weaknesses in FTP design that use control channels and data channels differently and do not identify the destination when creating a data channel. Using an anonymous FTP server, the attacker can manipulate the port command to scan the target network and have the FTP server send data to the attacker's destination.

#### 5.5.4.2 TFTP (Trivial FTP) Attack

TFTP [52] is a simple file transfer protocol application that can be installed on a workstation without a read-only memory or disk. Primarily used for delivering boot images to workstations that do not have their disk. There is a security vulnerability that uses the 69 UDP port and can access the specified directory without any authentication process. In TFTP attacks, if the access control is not properly controlled, an attacker can access arbitrary files by exploiting a weakness in TFTP.

#### 5.5.4.3 Anonymous FTP Attack

Anonymous FTP [53] service is a service that allows FTP access with an anonymous account. An anonymous account is an account with an ID of anonymous and no password (or any password). Allowing such anonymous accounts in public corporations or public institutions can cause serious security problems because any unauthorized user can access the server. If anonymous users even must write access, the attacker can upload malware and cause damage to multiple users.

### 5.6) Conclusion

This chapter talked about devices and protocols to be used for the research. AI smart speakers and two protocols are well known to the public. Thus, many attacks are expected. Based on this knowledge, we will be able to expect the data from the research.

## **제 6 장 IOT HONEYPOT DATA COLLECTION METHODOLOGY**

### **6.1) Introduction**

From the prior research, we have examined honeypots used in IoT. However, most of them used only honeypots to research, so the research using the IoT device directly was hard to find. It was also hard to find a way that fits the research we want. So, we made our own IoT honeypot for our research. In this chapter, we describe an IoT Honeypot to collect the data by pretending as AI smart speaker. Also, the expected attacks that we can get during the data collection.

### **6.2) Necessity of IoT Honeypot**

Many honeypots existed. All the honeypots only deal with specific protocols. However, the honeypot that we need is different from the current honeypot. We need the data of connection to the honeypot, but there is unlikely existed. Thus, we make the

first honeypot software, called IoT Honeypot, where we can load up the profiles and look like multiple devices. It does not have a fake virtual environment. It can emulate target device open TCP/UDP ports, and a profile defines what port and protocol to open. This allows us to quickly make a honeypot that looks like any real, internet-connected device. Also, it can create many different honeypots to compare attacks against different device types.

### **6.3) The design of IoT Honeypot**

The difference between the IoT Honeypot and the ordinary honeypot is that the IoT Honeypot does not respond to the protocols, only for open ports. The ordinary honeypot responds typically to the protocols and makes the virtual environment to let attackers coming in and hack the environment, but the IoT Honeypot is not. All we are interested in is how many times attackers are connecting to the port, not the full honeypot environment for each device because each device has a different system.

## 6.4) Code of IoT Honeypot

To make an IoT Honeypot, the GO language is in use. GO was first released in 2007 and officially announced in 2009 for Linux and Mac OS X platforms. In 2012, GO version 1.0 was released, and as of 2019, the latest version is GO 1.13.4. GO is a general-purpose programming language that follows a traditional compilation and linking model. GO was developed primarily for system programming and drew on the best of C++, Java, and Python. Like C++, GO is compiled through a compiler and is a statically-typed language. It is aimed at a simple and concise programming language and can be multi-processed. The code for IoT Honeypot is in Appendix A. Additional comments have been used between the codes with '//.'

## 6.5) IoT Honeypot Methodology

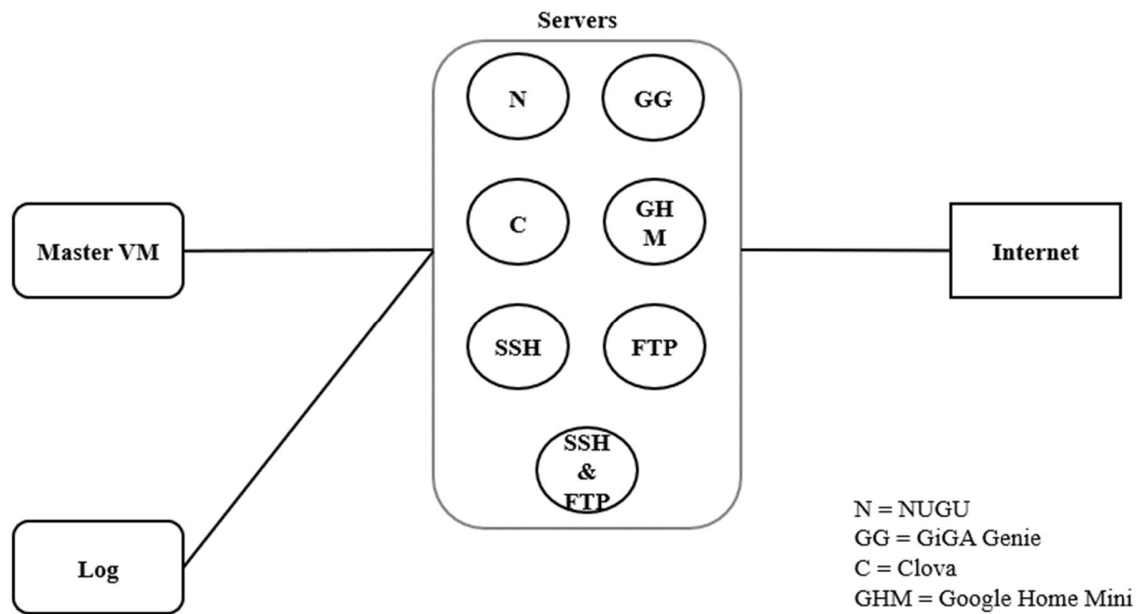


Figure 4 Diagram of IoT Honeypot setup

In this paper, honeypot, four AI smart speakers (Nugu, Giga Genie, Clova, and Google Home Mini) and two protocols (SSH and FTP) will be in use for the experiment. Figure 4 shows the setup of the IoT Honeypot. First, connect the speakers and laptop to the same access point to see what and which open port does speaker has. To check the open port, the GUI version of Nmap, Zenmap (version 7.70), is in use.

## 6.6) Devices

### 6.6.1 Giga Genie

Giga Genie has four TCP open ports, which are 7557, 7547, 8058, and 38520. Among these ports, 7547 is expected to be attacked most. The reason is that port 7547, which is called CWMP (CPE WAN Management Protocol) and known as TR-069(Technical Report 069). TR-069 is an Internet protocol based on XML and SOAP. This port allows the ISP to manage the router remotely. As a result, most routers operate with 7547 ports open by the TR-064 and TR-069 standards set by the Broadband Forum, for high-speed Internet. Attackers can use this to attack through the port. For example, in 2017 [54], foreign hackers hijacked vulnerable home routers and attacked WordPress sites. After analyzing vulnerabilities on routers and TR-069 provided by Korean telecommunications companies KT and LG, it was reported that the patch was not completed. So KT's Giga Genie, which uses port 7547, is expected to get most of the attacks than other ports.

### 6.6.2 Nugu

Nugu has UDP ports opened. Thus, it is expected to get UDP attacks that will be mentioned in 6.7) Procedures

### 6.5.3 Clova

Clova has UDP ports opened. Thus, it is expected to get UDP attacks that will be mentioned in 6.7) Procedures

### 6.6.4 Google Home Mini

Google Home Mini uses AJP (Apache Jserv Protocol) port (8009) and HTTPS (8443) port, where the vulnerability was found. Port 8009 is a port commonly used for Apache Tomcat. Apache Tomcat [55] is an open-source web server and servlet system that uses Java EE platforms such as Java Servlet, JavaServer Pages (JSP), Express Language, and Web Sockets to provide a pure Java HTTP web server environment. A recently discovered vulnerability is the Remote Code Execution Vulnerability (CVE-2019-0232), which occurs when running on Windows with enableCmdLineArguments enabled. It is a

bug in the Common Gateway Interface (CGI) servlet in which the Java Runtime Environment passes command arguments. The patch for this vulnerability has been updated. However, Apache Tomcat is a port that is likely to be attacked since it has been the target of attackers for years to date. Port 8443 is also used by Apache Tomcat and is usually used when configuring SSL. The port is vulnerable to an attack called Heartbleed, which is a web attack that exploits a vulnerability in OpenSSL. The maximum amount of memory that a client can request from the server is 64KB. If attackers request this information little by little and collect the letters, they can get useful information. As Google Home Mini uses these ports, the above attacks are expected to be found.

## 6.7) Procedures

			Clova	udp,773
				udp,1019
				udp,2967
				udp,17762
				udp,19650
				udp,20217
				udp,20678
		Nugu		udp,21318
			udp,1072	udp,21524
			udp,1782	udp,21784
			udp,2048	udp,26720
				udp,32777
		GHM		udp,34892
			tcp,8008	udp,39217
	Giga Genie		tcp,8009	udp,41638
			tcp,7547	udp,45818
			tcp,8012	udp,45818
			tcp,7557	udp,48189
			tcp,8443	udp,49226
			tcp,8058	udp,49226
			tcp,9000	udp,50612
			tcp,38520	udp,58631
				udp,10001

Figure 5 Profile files with open ports and protocol types

Once all the open ports are gathered, make it as profile file with each device's open ports and put it in the same folder like in Figure 5. Next, install a virtual box in the server computer for each speaker and protocols to use as a honeypot. At this point, the computer and the VirtualBox should be in public IP addresses so anyone can connect through the open ports. Then, install a program that can run the code. For the research, visual studio code has been used. When all the installations are done, run the code with profiles (ex. `sudo go run ballygul.go GigaGenie`). Currently, Nugu and Clova

use UDP port, Giga Genie, and GHM uses TCP port. SSH and FTP use the port they initially used. Next, a TCP dump will capture the packet and save it as a pcap file. After that, the collected data will be analyzed.

## **6.8) Conclusion**

In the chapter, we talked about an IoT Honeypot. We wanted to know if an IoT device would get more attacked than other devices, but the IoT honeypot used in the prior study did not have that technology. Thus, we made our honeypot. Then, we anticipated the attack that would be received while we launched the IoT Honeypot.

## 제 7 장 DISCUSSION

### 7.1) Introduction

In this chapter, we analyze the data that we collect by using the IoT Honeypot. The IoT Honeypot operated from November 21st to December 16th, and approximately 2 GB for each device packets have been gathered. Through the analysis, we can answer the research questions. All the packets from that port can be seen as an attack on an IoT device. As mentioned earlier, the device pretending by the IoT Honeypot does not have an environment that attackers can work like other normal honeypots. Because we only want connection data, so the IoT Honeypot device accepts the connection but nowhere to play on. Thus, anyone keeps talking to the device, and they are attacking because nobody should talk multiple times to the devices. Also, if the same IP address connects to multiple times, it is absolutely attacking.

## 7.2) Quantitative Analysis

The analysis will be conducted by answering each question one by one. The first question is, 'What IoT devices get the attack most?'

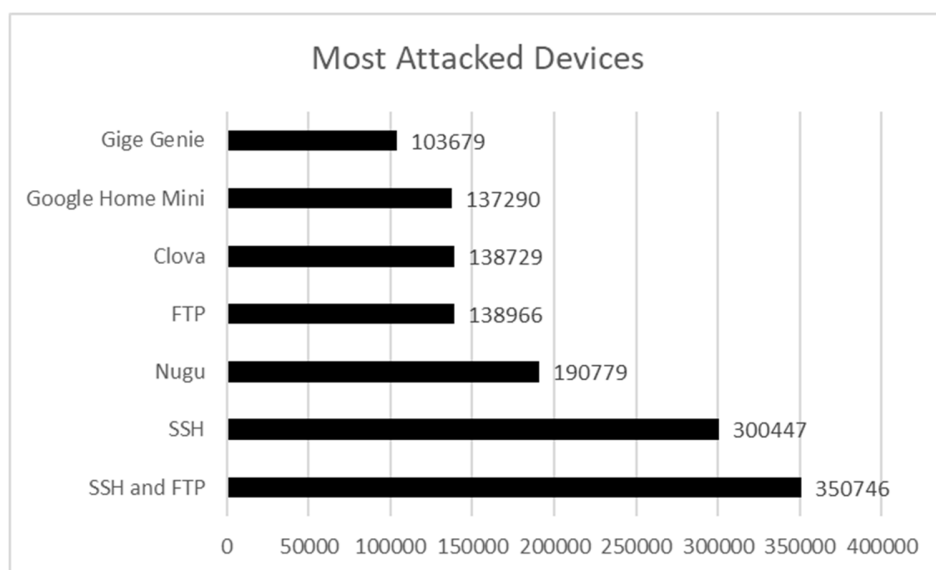


Figure 6 Most Attacked Devices

In Figure 6, the graph shows the most attacked devices during the research. It is counted with the packets they get from outside to their public IP addresses. In addition, it includes the packet from all open ports, not only port that we opened. It disproves that one of the hypotheses, which is 'Google Home Mini, will get attack more than other speakers because of the Market share.' Also, looking at the number of attacks does not

have a big difference. It can support the other hypothesis which is, 'Attackers are not focusing on specific devices'

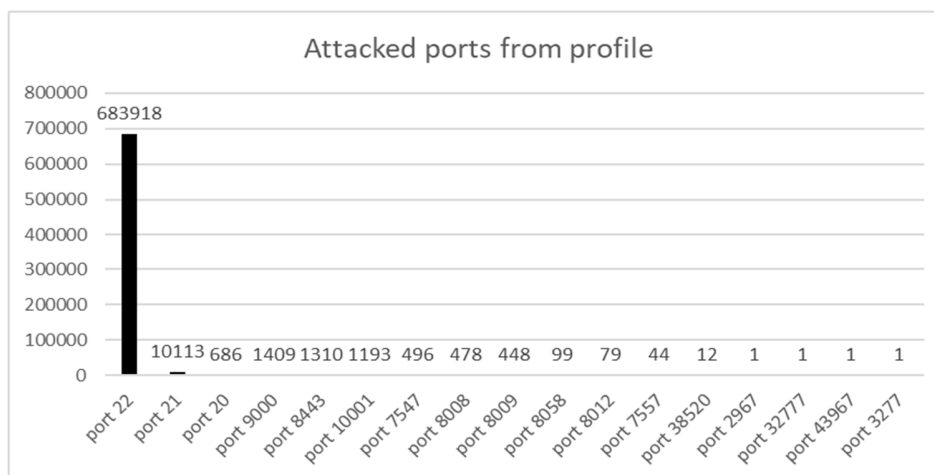


Figure 7 Attacked ports from profiles

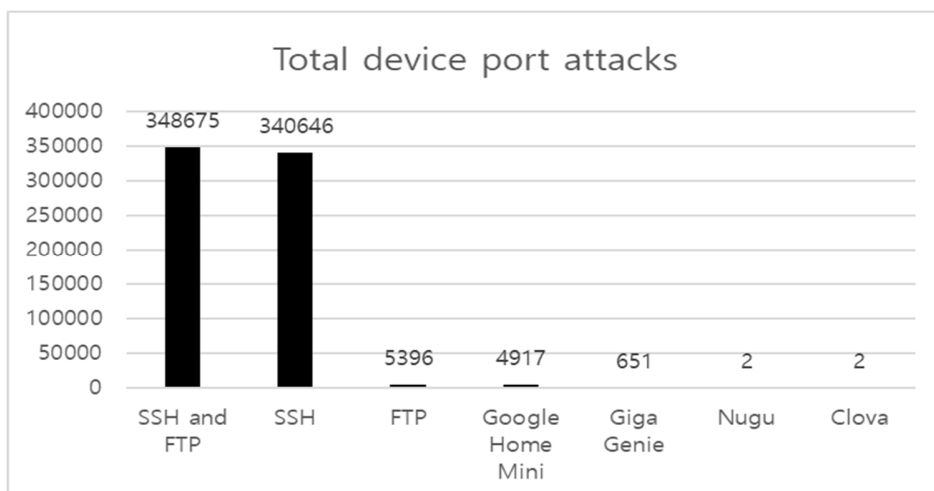


Figure 8 Total number of attacks to device port

The second question is, 'Which ports in profiles get more attacked?'. Figure 7 and 8 can be mean as how many people are trying to connect to the Honeypot. By looking it from the low to high, Clova and Nugu get two connections each into the UDP ports they opened as honeypots. The Giga Genie get 651 attacks, Google Home Mini get 4917 attacks, and FTP gets 5396 attacks. Through the port that we opened, Google Home Mini get the most attacks. As we mentioned previously, SSH, SSH, and FTP are well-known protocols, so they get incredibly high attacks through the port. Among the well-known protocols, SSH 22 port gets the most attacks. To check the data reliability, we compared the data from the study Abdou et al. [56] conducted using the SSH protocol. In the paper, they recorded the number of attempts to connect to SSH using a virtual machine. The research used six virtual machines. They conducted research for 373 days. By dividing total attempts per day, there were 8366 attempts. When we assume that we conduct research for 373 days, there are 12018 attempts per day. Because the previous research had six different attempts, and the average range among them were 26610 attempts to 1131 attempts. As our attempts are in the range so the data can be seen as reliable.

546...	44507.238321	198.108.67.48	210.115.255.247	TCP	66	30942 → 9000	[ACK] Seq=1 Ack=1 Win=29696 Len=0 TSval=667819953 TSecr=3208457...
546...	44507.252067	198.108.67.48	210.115.255.247	TCP	68	30942 → 9000	[PSH, ACK] Seq=1 Ack=1 Win=29696 Len=2 TSval=667819967 TSecr=32...
546...	44508.252135	198.108.67.48	210.115.255.247	TCP	66	30942 → 9000	[RST, ACK] Seq=3 Ack=1 Win=29696 Len=0 TSval=667820967 TSecr=32...
546...	44508.253699	198.108.67.48	210.115.255.247	TCP	74	55504 → 9000	[SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=66782096...
546...	44508.445255	198.108.67.48	210.115.255.247	TCP	66	55504 → 9000	[ACK] Seq=1 Ack=1 Win=29696 Len=0 TSval=667821158 TSecr=3208458...
546...	44509.522674	198.108.67.48	210.115.255.247	TCP	74	40166 → 9000	[SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=66782223...
546...	44509.523943	198.108.67.48	210.115.255.247	TCP	66	55504 → 9000	[RST, ACK] Seq=47 Ack=1 Win=29696 Len=0 TSval=667822237 TSecr=3...

Figure 9 Attackers attempt to connect to the Google Home Mini.

190...	1879.463204	35.154.90.185	210.115.255.246	ICMP	60	Echo (ping) request	id=0x001b, seq=15071/57146, ttl=224 (reply in 19002)
190...	1884.531770	13.235.49.136	210.115.255.246	ICMP	60	Echo (ping) request	id=0x0015, seq=2948/33803, ttl=224 (reply in 19045)
190...	1885.567949	13.235.49.136	210.115.255.246	ICMP	60	Echo (ping) request	id=0x0015, seq=7434/2589, ttl=224 (reply in 19056)
190...	1886.995117	3.10.221.34	210.115.255.246	ICMP	98	Echo (ping) request	id=0x000e, seq=5537/41237, ttl=22 (reply in 19072)
190...	1888.387343	13.233.194.24	210.115.255.246	ICMP	60	Echo (ping) request	id=0x0011, seq=13739/43829, ttl=224 (reply in 19085)
190...	1888.501874	3.10.214.203	210.115.255.246	ICMP	98	Echo (ping) request	id=0x001d, seq=10474/59944, ttl=23 (reply in 19088)
192...	1903.178642	3.10.221.34	210.115.255.246	ICMP	98	Echo (ping) request	id=0x0001, seq=23256/55386, ttl=26 (reply in 19229)

Figure 10 Attackers keep sending pings to the IoT Honeypot

The third question is, 'What types of data are the attackers sending?'. Figure 9 shows the attacker's attempt to connect to the Google Home Mini. It is not only the situation for Google Home Mini. We can find it from other honeypot data as well. As we mentioned previous, we do answer back when the attacker sends a packet. However, we do not have an environment. Thus, attackers think we do have an environment, so attempt to connect to the environment of honeypot. Figure 10 shows that attackers are checking whether it is a real device or not. We usually use ping to check whether the subject is working or not. Thus, it seems like attackers verify that the device is working and connecting to the device.

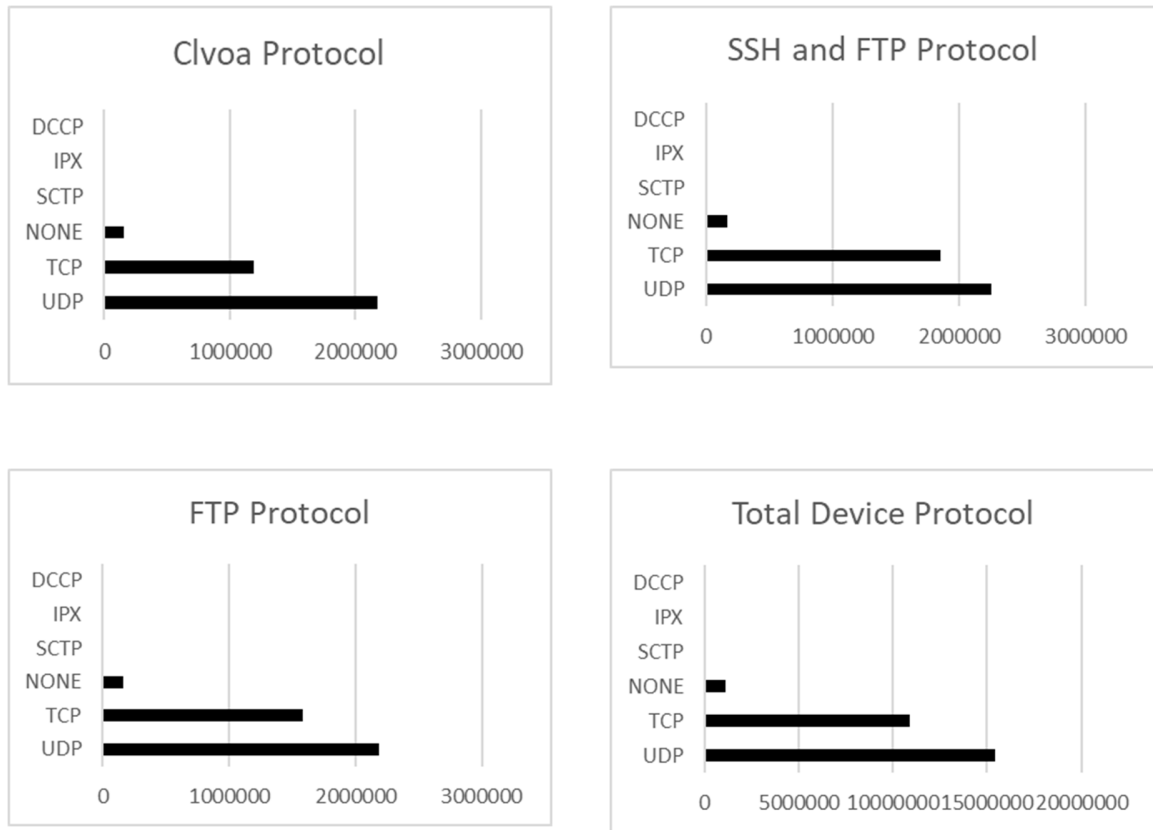


Figure 11 Common protocols of devices 1

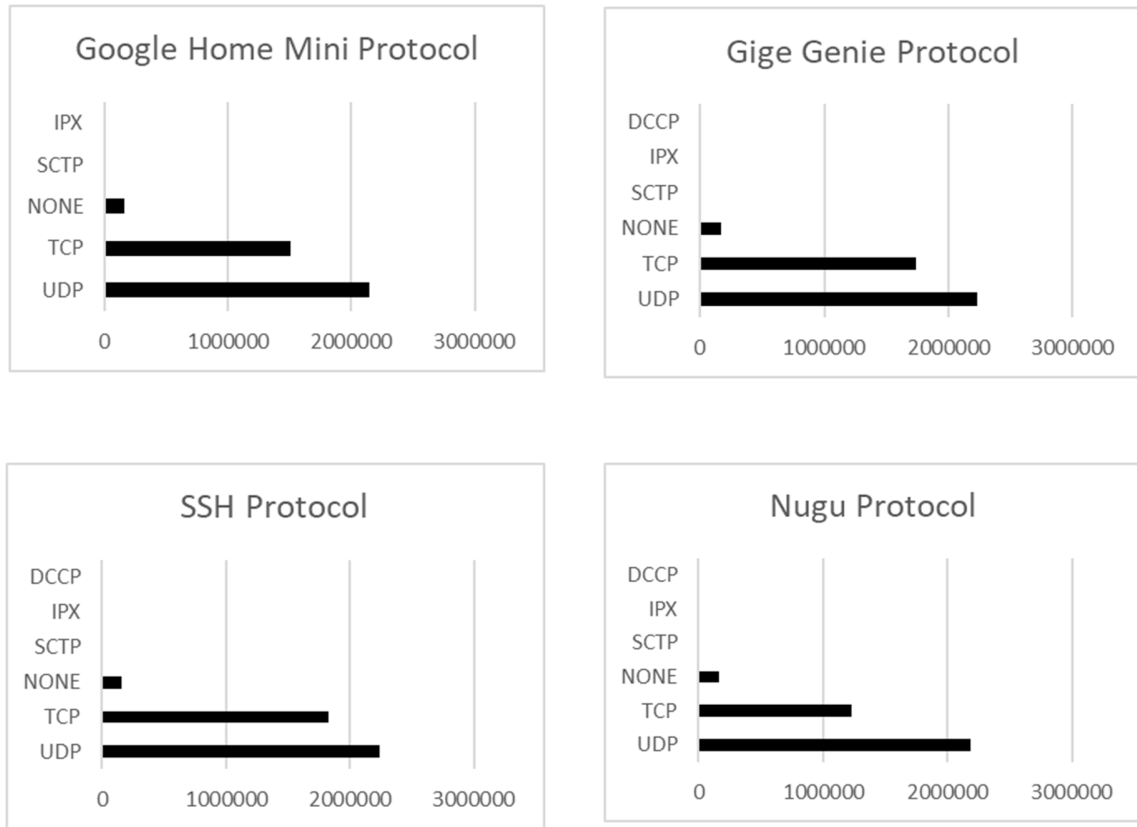


Figure 12 Common protocols of devices 2

The fourth question is, 'What protocol is common per device/total?'. In Figure 11 and 12, it shows protocol types for each device and in total. For each device, the UDP protocol is the highest, and for total, also UDP is the highest standard protocol that devices are using.

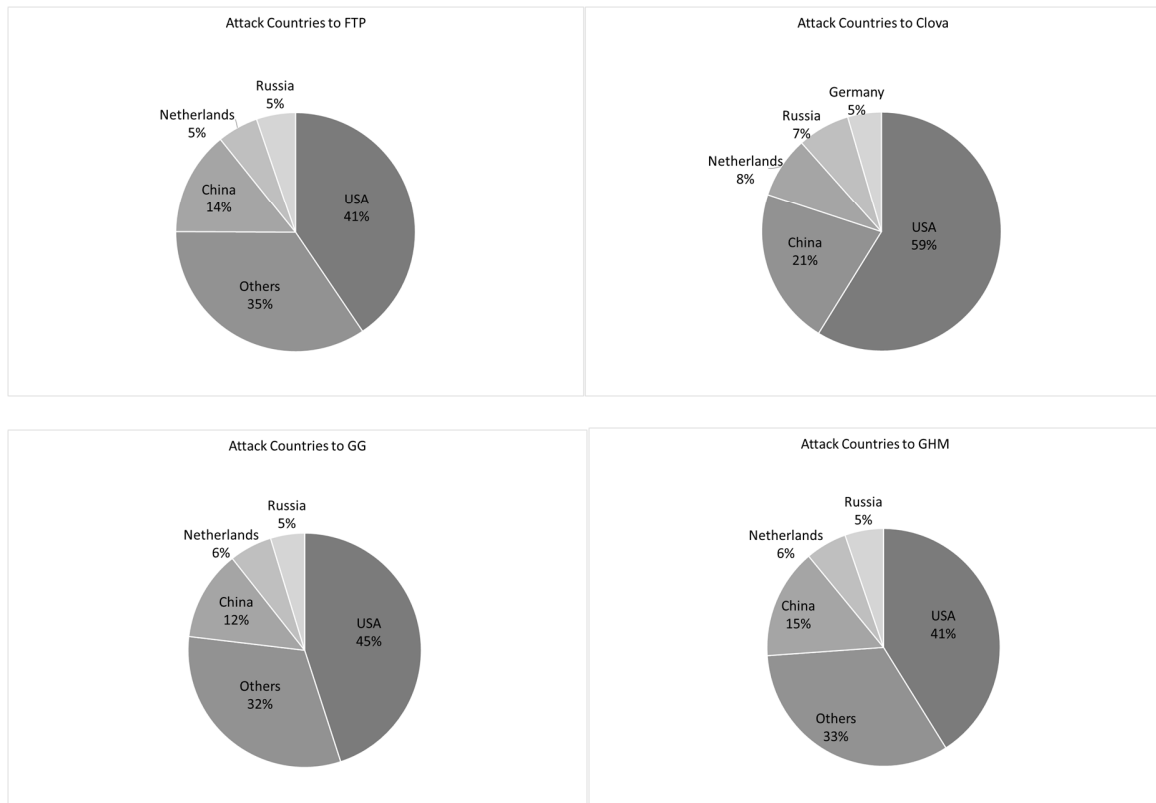


Figure 13 Attack Countries to device 1

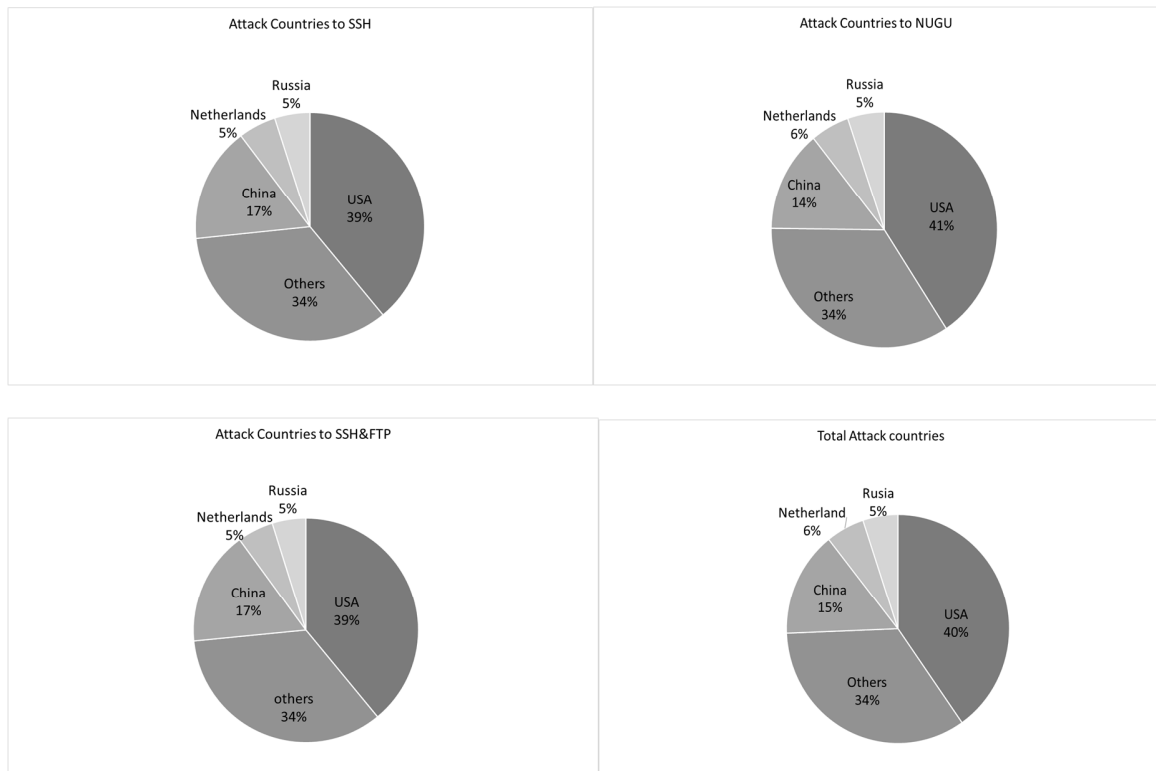
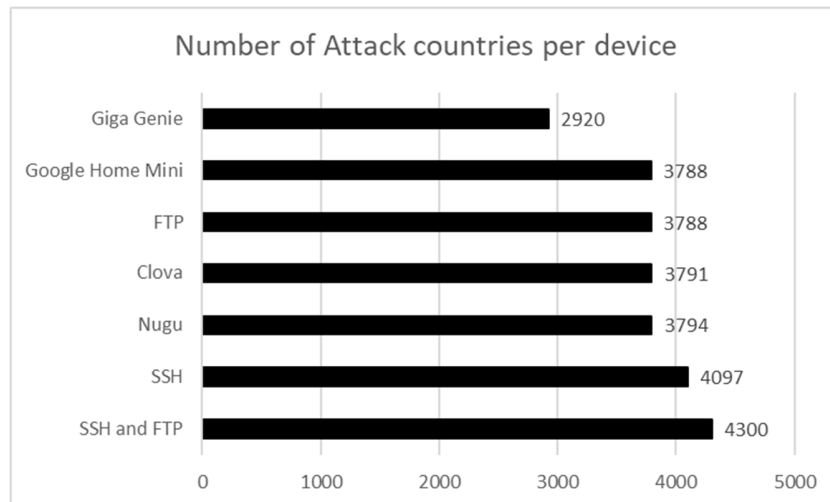


Figure 14 Attack Countries to device 2



**Figure 15 Number of Attack countries per device**

The fifth and sixth questions are 'Where the IP comes from? To what device?' and 'Attack countries per device' can be answered at the same time. Figure 13 and 14 shows filtering the IP addresses from the data and convert it to the country. In the percentage, less than 3% is classified in Others. The top 3 countries are the USA, China, and the Netherlands. Figure 15 shows the number of attack countries per device. It does not have a significant difference among devices. Thus, it supports one hypothesis, which is 'Attack is not focusing on a specific device.'

The seventh and eighth questions are 'Are attackers focusing on specific devices?' and 'Are they opportunistic? Or on purpose?'. Given the data we have analyzed so far, it is hard to see attackers focusing on a particular device because the number of packets coming into the device is not much different. Also, because the attackers tried to connect to the ports other than the one, we opened, it is more opportunistic than on purpose. However, multiple connections to devices can be seen as deliberate attacks.

### **7.3) Conclusion**

In the discussion, the data obtained from the IoT Honeypot was analyzed. For 25 days, about 2 GB of data was collected. With the data, we have answered the research questions and hypotheses. Through the analysis, the hypothesis is disproved that 'the Google Home Mini is more attacked than other speakers because of the Market share.' The reason is that the data coming into the open port was more of Google Home Mini, but when we checked in Figure 6 that the attacks of total data, Google Home Mini was similar or less than other speakers. Overall, it turns out that the Google Home Mini is not hypothesized that the market share is more attacked than other speakers. Also,

there was no attack targeting a specific port and device described previously, only a persistent connection attempt. There were many attacks that came into the port that was initially opened, rather than the port opened through the IoT Honeypot. In addition, the size of the data collected during the same period is not much different among devices. This proves that attackers do not focus on specific devices.

## 제 8 장 CONCLUSION

### 8.1) Introduction

In the early days, IoT, which was known only as of the concept of connecting people and things, and connecting things and things, is developing. As technology advances, more vulnerabilities are found. However, the analysis of vulnerabilities is not much compared to the technology developed rapidly. In this paper, we analyzed the vulnerabilities of IoT devices through AI smart speakers that are popular among IoT devices. To help with the research, IoT Honeypot was written in the Go language. As a result, there was no attack on a specific device, but we could confirm the attempt to connect to the device continuously, and we disprove one of the hypotheses that Google Home mini gets attacked more than other speakers. This means that market share has no impact. In addition, the size of collected data during the same period is similar among devices. This implies that attackers do not focus on specific devices.

## 8.2) Conclusion

Based on what we have seen, we have not done every single device obviously, but the devices that we have created an IoT Honeypot for, we either seeing that IoT devices are focused or not focused, and we have proven the two hypotheses. Among the AI smart speakers, the Google Home Mini was not the most attacked. Thus, our hypothesis is disproved. In addition, the same IP address was found to try to connect to the device multiple times. This can be seen as an intention by attackers to try to attack the device. However, the number is lower than the other two popular protocols, so it is hard to think as a device-intensive attack. Also, the IoT Honeypot can not only open specific ports but also collect packets from open ports, so when an incident occurs, people can use this honeypot to see which attacks come in and how many.

## 8.3) Future Work

All we are focused on is not only the speaker. It can be expanded to a smart home speaker. As the IoT devices are connected to each other to work, one smart home

device can affect all smart home devices. Therefore, in the future, none speaker IoT devices or smart industry IoT devices need to be done with this method. To find out the vulnerability of new IoT devices.

## REFERENCE

- [1] “Internet of Things Becomes Life! | Ministry of Science and Technology Information and Communication Webzine September 2018”. [Online]. Available at: <https://www.msit.go.kr/webzine/posts.do?postIdx=350>.
- [2] “190702 Release of the 4<sup>th</sup> Industrial Revolution Indicator\_1.pdf”.
- [3] Seong Hyun Na, "Personal Information Protection Issues in IoT Environment", KISDI, ISSN 2233-6583, 8 2015.
- [4] Jung Jin-wook, “Don't confess even you are lonely. Secret 'leaks' AI Speaker”, *MBC NEWS*, 08-10-2019. [Online]. Available at: [http://imnews.imbc.com/replay/2019/nwdesk/article/5536689\\_24634.html](http://imnews.imbc.com/replay/2019/nwdesk/article/5536689_24634.html).
- [5] “Lasers Can Hack Voice Assistants in Example Worthy of Mission Impossible But the Risk is Minimal for Consumers”, *Voicebot.ai*, 05-11-2019. [Online]. Available at: <https://voicebot.ai/2019/11/05/lasers-can-hack-voice-assistants-study/>.
- [6] Heo Jong-Oh, "A Study on the Construction of Global Honeypot System for Malicious Code Collection", *Journal of the Korean Information Science Society*, vol 37, 1D, pp 36-41, 6 2010.
- [7] “What is internet of things (IoT)? - Definition from WhatIs.com”, *IoT Agenda*. [Online]. Available at: <https://internetofthingsagenda.techtarget.com/definition/Internet-of-Things-IoT>.
- [8] “What is the Internet of Things, and how does it work?”, *Internet of Things blog*, 17-11-2016. [Online]. Available at: <https://www.ibm.com/blogs/internet-of-things/what-is-the-iot/>.
- [9] “What is Internet of Things (IoT)? | IoT Technology”, *SAP*. [Online]. Available at: <https://www.sap.com/korea/trends/internet-of-things.html>.
- [10] “IoT to open a new world (Internet of Things)”. [Online]. Available at: <https://brunch.co.kr/@brunchjwshim/54>.
- [11] “What is Honeypot and how does it improve security?”, *The Security Buddy*, 02-3-2017. [Online]. Available at: <https://www.thesecuritybuddy.com/data-breaches-prevention/what-is-honeypot/>.

- [12] “‘Understanding ‘Traps for hackers’ honeypots”, *ITWorld Korea*. [Online]. Available at: <http://www.itworld.co.kr/news/120233>.
- [13] “What is a honeypot and how can I protect my computer system?-Tip”. [Online]. Available at: <https://ko.play-and-more.com/8933-what-are-honeypots>.
- [14] I. Kuwatly, M. Sraj, Z. Al Masri and H. Artail, “A dynamic honeypot design for intrusion detection”, in *The IEEE/ACS International Conference on Pervasive Services, 2004. ICPS 2004. Proceedings.*, 2004, pp 95-104, doi: 10.1109/PERSER.2004.1356776.
- [15] R. Budiarto, A. Samsudin, C. W. Heong and S. Noori, “Honeypots: Why We Need A Dynamics Honeypots?”, School of Computer Sciences Universiti Sains Malaysia, ResearchGate, 2004.
- [16] “Strange document was printed...AI:IoT security still has a hole | Hankyung.com”. [Online]. Available at: <https://www.hankyung.com/it/article/201812289502g>.
- [17] “[International] ‘0 International Class AI Speakers’...IoT Urgently Needs Internalization of Security - Digital Today (DigitalToday)”. [Online]. Available at: <http://www.digitaltoday.co.kr/news/articleView.html?idxno=215582>.
- [18] “IoT Attack Opportunities Seen in the Cybercrime Underground - TrendLabs Security Intelligence Blog”. [Online]. Available at: <https://blog.trendmicro.com/trendlabs-security-intelligence/iot-attack-opportunities-seen-in-the-cybercrime-underground/>.
- [19] “delighIT.net 2.1”. [Online]. Available at: <http://delighit.net/post/get/34/14214/>.
- [20] U. D. Gandhi, P. M. Kumar, R. Varatharajan, G. Manogaran, R. Sundarasekar and S. Kadu, “HIoTPOT: Surveillance on IoT Devices against Recent Threats”, *Wirel. Pers. Commun.*, vol 103, No. 2, pp 1179-1194, 11 2018, doi: 10.1007/s11277-018-5307-3.
- [21] M. Anirudh, S. A. Thileeban and D. J. Nallathambi, “Use of honeypots for mitigating DoS attacks targeted on IoT networks”, in *2017 International Conference on Computer, Communication and Signal Processing (ICCCSP)*, 2017, pp 1-4, doi: 10.1109/ICCCSP.2017.7944057.
- [22] “IoTPOT: Analysing the Rise of IoT Compromises | USENIX”. [Online]. Available at: <https://www.usenix.org/conference/woot15/workshop-program/presentation/pa>.

- [23] H. Šemić and S. Mrdovic, “IoT honeypot: A multi-component solution for handling manual and Mirai-based attacks”, in *2017 25th Telecommunication Forum (TELFOR)*, 2017, pp 1-4, doi: 10.1109/TELFOR.2017.8249458.
- [24] “[1807.04114] ThingPot: an interactive Internet-of-Things honeypot”. [Online]. Available at: <https://arxiv.org/abs/1807.04114>.
- [25] “Alexa, are you listening?”, *F-Secure Labs*. [Online]. Available at: <https://labs.f-secure.com/archive/alexa-are-you-listening/>.
- [26] “Black Hat: Nest thermostat turned into a smart spy in 15 seconds | Computerworld”. [Online]. Available at: <https://www.computerworld.com/article/2476599/black-hat-nest-thermostat-turned-into-a-smart-spy-in-15-seconds.html>.
- [27] “Do Current OS Vulnerabilities Affect IoT? | IoT Security”. [Online]. Available at: <https://www.trendmicro.com/us/iot-security/news/2208>.
- [28] “Unsecured IoT: 8 Ways Hackers Exploit Firmware Vulnerabilities”, *Dark Reading*. [Online]. Available at: <https://www.darkreading.com/risk/unsecured-iot-8-ways-hackers-exploit-firmware-vulnerabilities/a/d-id/1335564>.
- [29] A. Mohanty, I. Obaidat, F. Yilmaz and M. Sridhar, “Control-hijacking Vulnerabilities in IoT Firmware: A Brief Survey”, p 4.
- [30] “More than 1 million ASUS computers hacked. domestic damage is ‘yet’”. [Online]. Available at: <http://www.ddaily.co.kr/news/article.html?no=179338>.
- [31] “ZigBee-sniffing drone maps hackable IoT devices”, *eeNews Europe*, 07-8-2015. [Online]. Available at: <https://www.eenewseurope.com/news/zigbee-sniffing-drone-maps-hackable-iot-devices>.
- [32] “Four types of network attacks businesses should know”, *boannews*, 08-11-2018. [Online]. Available at: <http://www.boannews.com/media/view.asp?idx=74416>.
- [33] “One Step Ahead igloosecurity”. [Online]. Available at: [http://www.igloosec.co.kr/BLOG\\_IoT%20%EB%B3%B4%EC%95%88%EC%9C%84%ED%98%91%EC%97%90%20%EB%94%B0%EB%A5%B8%20%EB%8C%80%EC%9D%91%EB%B0%A9%EC%95%88?searchItem=&searchWord=&bbsCatId=17&gotoPage=2](http://www.igloosec.co.kr/BLOG_IoT%20%EB%B3%B4%EC%95%88%EC%9C%84%ED%98%91%EC%97%90%20%EB%94%B0%EB%A5%B8%20%EB%8C%80%EC%9D%91%EB%B0%A9%EC%95%88?searchItem=&searchWord=&bbsCatId=17&gotoPage=2).
- [34] M. B. in S. on October 15, 2019 and 11:33 Am Pst, “Kaspersky honeypots find 105 million attacks on IoT devices in first half of 2019”, *TechRepublic*. [Online].

- Available at: <https://www.techrepublic.com/article/kaspersky-honeypots-find-105-million-attacks-on-iot-devices-in-first-half-of-2019/>.
- [35] “IoT Security Guidance - OWASP”. [Online]. Available at: [https://www.owasp.org/index.php/IoT\\_Security\\_Guidance](https://www.owasp.org/index.php/IoT_Security_Guidance).
  - [36] Young-Gap Kim, “Internet of Things Security Requirements”, Sejong University, Information and Communication Technology Promotion Center, 2017.
  - [37] “Essential component of IoT ‘IoT Platform’”, Techworld, 17-4-2019. [Online]. Available at: <http://www.epnc.co.kr/news/articleView.html?idxno=83017>.
  - [38] “Press Release | Green Zone Security”. [Online]. Available at: [http://greenzonesecu.com/bbs/board.php?bo\\_table=press\\_ko&wr\\_id=60](http://greenzonesecu.com/bbs/board.php?bo_table=press_ko&wr_id=60).
  - [39] “SKT NUGU”. [Online]. Available at: <https://www.nugu.co.kr>.
  - [40] “Giga Genie | Global No.1 KT”. [Online]. Available at: <https://gigagenie.kt.com/main.do>.
  - [41] “Naver Clova”. [Online]. Available at: <https://clova.ai/ko>.
  - [42] “Google Home, South Korea's official landing with six features | IT Dong-A”. [Online]. Available at: <https://it.donga.com/28149/>.
  - [43] “Google Home Mini”, *Google Store*. [Online]. Available at: [https://store.google.com/kr/product/google\\_home\\_mini](https://store.google.com/kr/product/google_home_mini).
  - [44] “[Ubuntu/Linux] everything of ssh public key”, I'm a developer, 19-7-2015. [Online]. Available at: <https://storycompiler.tistory.com/112>.
  - [45] “what is a FTP?.”, Naver | Foundation of the assembly of the computer. [Online]. Available at: <https://blog.naver.com/hdj20/40155944026>.
  - [46] FTP concept, SFTP, passive, active mode”, *D.O's IT*, 15-8-2017. [Online]. Available at: <https://dany-it.tistory.com/54>.
  - [47] “DDoS attack types#1”, Zippan, 15-6-2018. [Online]. Available at: <https://jihwan4862.tistory.com/122>.
  - [48] “DDoS attack types#2”, Zippan, 18-6-2018, 18-6-2018. [Online]. Available at: <https://jihwan4862.tistory.com/123>.
  - [49] “<<DoS/DDoS>> RUDY & Slowloris attack types”, Naver | High meaning!! Dreaming of Network and information Security Experts. [Online]. Available at: [https://blog.naver.com/p\\_rain/220687298632](https://blog.naver.com/p_rain/220687298632).

- [50] “UDP Flood DDoS Attack”, *Cloudflare*. [Online]. Available at:  
<https://www.cloudflare.com/learning/ddos/udp-flood-ddos-attack/>.
- [51] “KISA Internet protection & KrCERT”. [Online]. Available at:  
<https://www.boho.or.kr/>.
- [52] “[Information security engineer ]FTP Bounce attack”, *securityissue*, 30-10-2018.  
 [Online]. Available at: <https://securityissue.tistory.com/56>.
- [53] T. Greene, “FBI warns of attacks on anonymous FTP servers”, *Network World*, 28-3-2017. [Online]. Available at: <https://www.networkworld.com/article/3185873/fbi-warns-of-attacks-on-anonymous-ftp-servers.html>.
- [54] “WordPress site attack through vulnerable home router...South Korea is a risk”.  
 [Online]. Available at:  
[https://www.ahnlab.com/kr/site/securityinfo/secunews/secuNewsView.do?menu\\_dist=1&cmd=print&seq=26340](https://www.ahnlab.com/kr/site/securityinfo/secunews/secuNewsView.do?menu_dist=1&cmd=print&seq=26340).
- [55] “Apache Tomcat, Remote Code Execution Vulnerability Patch”, East Security Pill, 16-4-2019. [Online]. Available at: <https://blog.alzac.co.kr/2256>.
- [56] A. Abdou, D. Barrera and P. C. van Oorschot, “What Lies Beneath? Analyzing Automated SSH Brute-force Attacks”, in *Technology and Practice of Passwords*, Cham, 2016, pp 72-91.

# APPENDIX

## <Appendix 1> Code of IoT Honeypot

```
package main

import (
    "bufio"
    "fmt"
    "log"
    "math/rand"
    "net"
    "os"
    "os/exec"
    "path/filepath"
    "strconv"
    "strings"
    "sync"
    "time"
)
```

```

// Quick honeypot test code

//////////////////////////////// Program run functions //////////////////////////////////

// ReadLines reads a whole file into memory
func readLines(path string) ([]string, error) {
    file, err := os.Open(path)
    if err != nil {
        return nil, err
    }
    defer file.Close()

    var lines []string
    scanner := bufio.NewScanner(file)
    for scanner.Scan() {
        lines = append(lines, scanner.Text())
    }
    return lines, scanner.Err()
}

```

```

// Read the installed profiles from the profiles directory, and print them

func getProfiles() {

    fmt.Println("Installed profiles:")

    var files []string

    root := "profiles/"

    err := filepath.Walk(root, func(path string, info os.FileInfo, err error) error {

        if filepath.Ext(path) == ".pro" {

            files = append(files, path)

        }

        return nil

    })

    if err != nil {

        panic(err)

    }

    for _, file := range files {

        fmt.Println(file)

    }

}

// Print the user help information.

```

```

func getHelp() {

    fmt.Println("Usage: Ballygul [Profile]")

    fmt.Println("\tProfile: A device profile used to build the honeypot.")

    fmt.Println("")

    fmt.Println("Ballygul is a quick honeypot builder based on profile built from scans.")

    fmt.Println("Add new profiles in the 'Profiles' sub-directory. Make sure they end with 'GigaGenie.pro'")

    fmt.Println("Call be profile by name without the 'GigaGenie.pro' extension. ex: \"GigaGenie\"")

    fmt.Println("")

    getProfiles()

}

```

//////////////////////////////// Networking Functions //////////////////////////////////

```

func tcpFlow(port string) {

    protocol := "tcp"

    fmt.Println("Opening " + protocol + " port " + port)

    l, err := net.Listen(protocol, ":"+port)

    if err != nil {

        log.Println(err)

    }

    defer l.Close()

```

```

for {
    _, err := l.Accept()
    if err != nil {
        log.Println(err)
    }
    // This is where the byte response would go
    //c.Write([]byte("hello"))
}
}

```

```

func udpFlow(port string) {
    protocol := "udp"
    fmt.Println("Opening " + protocol + " port " + port)
    s, err := net.ResolveUDPAddr("udp4", ":"+port)
    if err != nil {
        fmt.Println(err)
        return
    }

```

```

    connection, err := net.ListenUDP("udp4", s)
    if err != nil {

```

```

        fmt.Println(err)

        return
    }

    defer connection.Close()

    buffer := make([]byte, 1024)

    r := rand.New(rand.NewSource(time.Now().Unix()))

    for {

        _, addr, err := connection.ReadFromUDP(buffer)

        data := []byte(strconv.Itoa(r.Int()))

        _, err = connection.WriteToUDP(data, addr)

        if err != nil {

            fmt.Println(err)

            return

        }

    }

}

```

```

// Function to run TCP dump automatically

```

```

func runTCPDump(wg *sync.WaitGroup, profile string) {

    defer wg.Done()

    binary, lookErr := exec.LookPath("tcpdump")

    if lookErr != nil {

        panic(lookErr)

    }

    // Use a variable for the name with the profile and date

    t := time.Now()

    args := []string{"tcpdump", "-C2048", "-w" + t.Format(time.RFC3339) + "-" + profile +
"capture.pcap"}

    fmt.Println(args)

    cmd := exec.Command(binary, "-C2048", "-w"+t.Format(time.RFC3339)+"-
"+profile+"capture.pcap")

    cmd.Start()

    //env := os.Environ()

    /*execErr := syscall.Exec(binary, args, env)

    if execErr != nil {

        panic(execErr)

    }*/

}

```

```

func main() {

    fmt.Println("Bally Gul v0.0.2")

    var profile string

    if len(os.Args) == 2 {

        profile = os.Args[1]

        //fmt.Println(profile)

        if _, err := os.Stat("profiles/" + profile + ".pro"); err == nil {

            fmt.Println("Profile " + profile + " found! Loading...")

            // Profile found, continue to the main functions.

        } else {

            //fmt.Println("Profile does not exist!")

            getHelp()

            os.Exit(1)

        }

    } else {

        getHelp()

        os.Exit(1)

    }

    // Get profiles

    lines, err := readLines("profiles/" + profile + ".pro")

    if err != nil {

```

```

        log.Fatalf("readLines: %s", err)
    }

    var wg sync.WaitGroup

    // Get the ports and listen on each port
    for _, line := range lines {
        //fmt.Println(i, line)

        var info = strings.Split(line, ",")

        //fmt.Println("Protocol is " + info[0])
        //fmt.Println("Port is " + info[1])

        if info[0] == "tcp" {

            // Call tcpFlow to set up TCP ports

            wg.Add(1)

            go tcpFlow(info[1])

        } else if info[0] == "udp" {

            // Call udpFlow to set up UDP ports

            wg.Add(1)

            go udpFlow(info[1])

        } else {

            fmt.Println("Found an unknown protocol. Expecting tcp or udp.")

            fmt.Println("Check the profile and try again.")

            os.Exit(1)
        }
    }

```

```
        }  
    }  
  
    wg.Add(1)  
  
    // Run TCPDump in the background  
    runTCPDump(&wg, profile)  
  
    wg.Wait()  
  
}
```

## IoT 장치의 악성 타겟팅에 관한 연구

2019.

석사학위논문

박민진

국제학과

지도교수: Joshua I. James

초기의 Internet of Things (IoT)는 Machine to Machine (M2M) 커뮤니케이션처럼 사물과 사물, 사물과 사람 사이의 일반적인 통신을 기반으로 연결하는 개념이었다. 그러나 최근 IoT는 통신과 센서 기능을 기기에 부착하여, 인공지능 (AI)과 머신러닝을 사용해 각종 사물들이 스스로 학습 및 판단하고, 생각할 수 있는 개념이라고 볼 수 있다. IoT의 급진적인 발전이 이루어지는 만큼, IoT의 취약점을 목표로 하는 공격도 다양해지고 있다. 하지만 취약점에 대한 분석은 크게 이루어지고 있지 않고 있다.

IoT가 현실 세계와 디지털 세계를 이어주는 만큼 사용자가 디지털 세계에서 해킹을 받으면 현실 세계에도 영향을 줄 수 있다. 본 논문에서는 가장 많이 사용되는 IoT 장치 중 하나인 AI 스마트 스피커를 대상으로 IoT 장치의 공격을 분석하였다. 연구를 진행하기 위해 공격 분석에 많이 사용되고 있는 허니팟을 목적에 맞게 IoT 허니팟으로 만들었다.

모든 IoT 스마트 장치는 인터넷을 통해 통신을 하기위해 포트를 가지고 있기 때문에 실제 장치가 사용하는 포트를 확인한 후, 포트를 하나의 프로파일로 만들었다. 그 후, 외부에서 접속 할 수 있게

IoT 허니팟에 퍼블릭 아이피를 부여하고, 해당 포트들을 열어준 다음 포트에 들어오는 데이터를 수집해서 분석 하였다.

포트를 통해 들어온 데이터를 분석한 결과, 장치를 목표로 한 공격은 발견되지 않았지만, 다양한 곳에서 포트에 접속하려고 시도한 것을 발견할 수 있었다. 본 논문에서 IoT 허니팟을 사용해 진행한 공격 분석은 장치의 포트만 확인할 수 있으면, 추후 스피커 이외의 다른 IoT 장치에도 사용될 수 있다.

주제어: 사물인터넷, 허니팟, 인공지능 스마트 스피커, IoT 장치 공격, IoT 포렌식

## A Study on the Malicious targeting of IoT Devices

2019.

Master's Degree

Park, Min Jin

Department of International Studies

Advisor Prof. Joshua I. James

In the early Internet of Things (IoT), like Machine to Machine (M2M) communication, was a concept of connecting things based on regular communication between things, things, and people. However, IoT is a concept that various objects can learn, judge, and think by using artificial intelligence (AI) and machine learning by attaching communication and sensor functions to devices. As the radical development of IoT is taking place, there are a variety of attacks targeting IoT device vulnerabilities. However, the analysis of vulnerabilities has not been done much.

IoT connects the real world with the digital world. Increasingly, if a user is hacked in the digital world, it can affect the real world. In this paper, we analyzed the attack of IoT devices targeting AI smart speaker, one of the most used IoT devices. To proceed with the research, a honeypot, which is widely used for attack analysis, was created.

We call it an IoT Honeypot. Every IoT smart device has a port for communicating over

the Internet, so after identifying the port that the real device uses, we have made the port as a profile. After that, the public IP address was assigned to the IoT Honeypot for external access, the ports were opened, and the data coming into the ports was collected and analyzed.

Analyzing the data coming in through the ports revealed no attacks targeting the device but found attempts to access the port from various places. Attack analysis using IoT Honeypot in this paper can be used for other IoT devices later if only the port of the device can be identified.

**Keywords:** Internet of Things, Honeypot, AI smart speaker, IoT device attack, IoT

**Forensic**

국제학석사 학위논문

# A Study on the Malicious targeting of IoT Devices

IoT 장치의 악성 타겟팅에 관한 연구

박 민 진(Park, Min Jin)

국제학과(Department of International Studies)

정보법과학전공(Major in Legal Informatics & Forensic Science)

한림대학교 대학원

(Graduate School, Hallym University)

국제학석사 학위논문

# A Study on the Malicious targeting of IoT Devices

IoT 장치의 악성 타겟팅에 관한 연구

박 민 진(Park, Min Jin)

국제학과(Department of International Studies)

정보법과학전공(Major in Legal Informatics & Forensic Science)

한림대학교 대학원

(Graduate School, Hallym University)



Joshua I. James 교수지도

국제학 석사 학위논문

박민진의 석사 학위논문을 합격으로 판정함

2019 년 12 월 19 일

심사위원장 박 노 섭

---

심사위원 장 윤 식

---

심사위원 Joshua I. James

---

## 목차(TABLE OF CONTENTS)

List of Figures .....	vi
제 1 장 INTRODUCTION.....	1
1.1) Current Status.....	2
1.1.1 Number of IoT devices .....	2
1.1.2 Threats to the IoT .....	2
1.1.3 IoT Honeypot.....	4
1.2) Thesis Structure.....	5
제 2 장 RESEARCH QUESTIONS AND HYPOTHESES.....	8
2.1) Introduction .....	8
2.2) Thesis Statement .....	9
2.3) Research Questions .....	10
2.4) Hypotheses.....	11
2.5) Conclusions.....	11
제 3 장 BACKGROUND RESEARCH .....	13
3.1) Introduction .....	13
3.2) IoT .....	13

3.3) Honeypot .....	15
3.3.1 Types of honeypot .....	15
3.4) IoT Devices .....	17
3.5) IoT Security .....	18
3.6) IoT Honeypot .....	20
3.7) Conclusion .....	24
제 4 장 RELATED WORK OF IOT DEVICE SECURITY .....	25
4.1) Introduction .....	25
4.2) Hardware.....	25
4.2.1 Debug Pad .....	25
4.2.2 Hardware Backdoor .....	26
4.3) Software .....	27
4.3.1 OS (Operating System) .....	27
4.3.2. Firmware.....	28
4.3.3. Backdoor .....	29
4.4) Network.....	29
4.4.1 Sniffing .....	29
4.4.2 Port Scanning.....	30

4.5) Attack Data.....	30
4.5.1 DDOS (Denial-of-service attack) .....	30
4.5.2 Kaspersky.....	31
4.6) Security Method .....	31
4.6.1 Guideline for security .....	31
4.6.2 Requirements .....	32
4.6.3 Platform .....	33
제 5 장 DEVICES USED FOR RESEARCH.....	35
5.1) Introduction .....	35
5.2) AI Smart Speaker.....	36
5.2.1 SKT Nugu .....	36
5.2.2 KT Giga Genie .....	36
5.2.3 Naver Clova.....	37
5.2.4 Google Google Home Mini.....	38
5.3) SSH (Secure Shell) .....	39
5.4) FTP (File Transfer Protocol) .....	40
5.5) Expected Attack.....	41
5.5.1 TCP (Transmission Control Protocol) .....	41

5.5.2 UDP (User Datagram Protocol) .....	46
5.5.3 SSH (Secure Shell) .....	47
5.5.4 FTP (File Transfer Protocol) .....	48
5.6) Conclusion .....	50
제 6 장 IOT HONEYPOT DATA COLLECTION METHODOLOGY.....	51
6.1) Introduction .....	51
6.2) Necessity of IoT Honeypot.....	51
6.3) The design of IoT Honeypot .....	52
6.4) Code of IoT Honeypot .....	53
6.5) IoT Honeypot Methodology.....	54
6.6) Devices.....	55
6.6.1 Giga Genie .....	55
6.6.2 Nugu .....	56
6.5.3 Clova.....	56
6.6.4 Google Home Mini.....	56
6.7) Procedures .....	58
6.8) Conclusion .....	59
제 7 장 DISCUSSION .....	60

7.1) Introduction .....	60
7.2) Quantitative Analysis .....	61
7.3) Conclusion .....	70
제 8 장 CONCLUSION .....	72
8.1) Introduction .....	72
8.2) Conclusion .....	73
8.3) Future Work .....	73
REFERENCE.....	75
APPENDIX .....	80
<Appendix 1> Code of IoT Honeypot.....	80
국문 초록.....	90
English Abstract .....	92

## List of Figures

Figure 1 Honeypot configuration [11] .....	15
Figure 2 SSH key exchange protocol .....	39
Figure 3 FTP configuration [45].....	40
Figure 4 Diagram of IoT Honeypot setup .....	54
Figure 5 Profile files with open ports and protocol types.....	58
Figure 6 Most Attacked Devices .....	61
Figure 7 Attacked ports from profiles .....	62
Figure 8 Total number of attacks to device port .....	62
Figure 9 Attackers attempt to connect to the Google Home Mini. ....	64
Figure 10 Attackers keep sending pings to the IoT Honeypot .....	64
Figure 11 Common protocols of devices 1 .....	65
Figure 12 Common protocols of devices 2 .....	66
Figure 13 Attack Countries to device 1 .....	67
Figure 14 Attack Countries to device 2 .....	68
Figure 15 Number of Attack countries per device.....	69

## 제1장 INTRODUCTION

The early Internet-of-Things (IoT) was connection-oriented, with monitoring and control at the center. It was a concept that built communication functions in various things and connected them to the Internet and made Internet-based communication between people-and-things and things-and-things. It can be described as a Machine to Machine communication (M2M). However, with the recent development of artificial intelligence (AI) technology [1], it is developing into intelligent IoT based on cognitive technology that learns, infers, and judges. Intelligent IoT uses AI and machine learning to interact with people and their surroundings with advanced capabilities through AI beyond programming execution. AI is leading the development of various intelligent objects such as autonomous vehicles, robots, and healthcare. It is also developing the capabilities of many things, including IoT, connected consumers, and industrial

systems. Representative examples of everyday life are smart homes and smart home appliances.

## **1.1) Current Status**

### **1.1.1 Number of IoT devices**

There are various devices such as air conditioners, refrigerators, lights. Among them, the AI smart speaker is easily obtained by anyone and is widely used in real life. The Ministry of Science and ICT announced [2] the fourth industrial revolution indicator for each sector, including AI speakers. The number of IoT device connections was 1400 million in December 2017, and it increased to 1865 million in December 2018. Also, the number of AI speaker sales increased from 200 million in March 2018 to 412 million in March 2019. At the end of 2019, it will be expected to 800 million.

### **1.1.2 Threats to the IoT**

IoT is convenient and easy to manage through one set. Sensors in IoT devices collect data, including sensitive data [3]. However, numerous IoT devices connected to the

Internet, such as IP cameras, AI speakers, and smart appliances, are exposed to security threats such as hacking and Distributed Denial of Service (DDoS) attacks. It is because manufacturers and users are less aware of IoT security. In a 2019 article [4], AI experts found a severe security flaw in AI speakers sold to consumers. If a user talks to the AI speaker controlled by a hacker, the hacker can listen to the conversation in real-time. In a more recent case [5], researchers in the US and Japan found a way to hack AI smart speakers using lasers. Encode voice commands and send them as laser light, the speaker responds. This is the principle that when light hits a diaphragm with a built-in speaker, the diaphragm vibrates and performs is recognized as user speech. This method is possible for up to 110 meters away. The device used for the experiment is 400 dollars, and anyone with malicious motives can attack AI speakers outside the home.

Also, like Mirai-botnet, which occurred in 2016, there is still a threat of massive DDoS attacks through hacking of IoT devices. The proliferation of industries based on IoT technologies, such as automated vehicles and smart cities, increases the risk of

these cyberattacks leading to life-threatening physical damage. It applies not only to homes but also to businesses, which is even more dangerous. In the case of a company, they use a centralized network connection. Thus, once an attack is made, critical business systems may go down. The IoT is the bridge between the real world and the digital world, so if the hacker attacks people in the digital world, it can affect the real world.

### 1.1.3 IoT Honeypot

A honeypot is a system of application programs and data designed to lure hackers and attackers. It looks like a real system, but it can be designed to look like there are exploitable vulnerabilities to monitor the attacker's actions. It is already actively used overseas as a program that can attract attackers to fake the IoT environment to collect types and methods of attack and prepare countermeasures. Currently, studies using honeypots are being conducted in Korea [6], but there are hardly any studies that combine them with IoT. Also, in the case of IP cameras, there is data about how much of the attack was carried out. However, the AI speaker is only reporting vulnerabilities

to hack but has no data on accurately how much of the attack has been performed. Therefore, in the thesis, we will research about the attack on the IoT using AI smart speakers. A representative device of IoT echo systems, as well as a honeypot that examines the behavior of attackers.

## **1.2) Thesis Structure**

In Chapter 2, as the definition of the Internet of Things still varies, there are no exact definitions. We look at the definition of the IoT used in this paper. Also, when new technologies are released, vulnerabilities are also found, as is the IoT. With the focus on IoT, attacks appear to be more common than against other devices. Thus, research questions and hypotheses to see if they are attacked more than other devices.

In Chapter 3, talks about a brief explanation of the types of IoT and honeypot. Then we research the background of IoT devices, IoT security, and IoT honeypot. It talks about the stance of domestic IoT devices, overseas interest in IoT security, and the

actual situation of domestic IoT security. Also, it introduces cases of using a honeypot to analyze IoT attacks.

In Chapter 4, talks about the related work of IoT device security. As attention has focused on the Internet of Things, and as security advances have been made, security vulnerabilities have also been found. Through the chapter, we will see the types of security vulnerabilities that IoT device would have and the possible attacks can be done to IoT device.

In Chapter 5, as the Internet of Things has evolved, many IoT devices have been released. Among them, IoT devices studied in this paper will be described, as well as SSH and FTP protocols. Also, since IoT uses protocols, we will look at the types of attacks the protocol can receive.

In Chapter 6, explains the necessity of an IoT Honeypot, why we develop our own IoT honeypot and how the honeypot is designed and methodology to use the honeypot in the research as well as, expected attacks through the honeypot will be described.

In Chapter 7, collect the necessary data using the IoT Honeypot described in the previous chapter. Then, answer the proposed research questions and support or deny the hypotheses mentioned above by analyzing the data collected over a period of time.

Finally, in Chapter 8, we conclude with the results obtained from the research. Then, discuss what other additional research should follow in the future.

## 제 2 장 RESEARCH QUESTIONS AND HYPOTHESES

### 2.1) Introduction

In this chapter, we will research questions under investigation in the thesis study.

There is no agreed definition of the Internet of Things (IoT). IoT Agenda [7] defines the Internet of Things as a system that has a unique identifier and can send and receive data to computers, machines, connected through a network without a human being. IBM [8] defines it as connecting all devices with an Internet connection that can be turned on and off. SAP [9] defines sensors and APIs as a network of physical entities that exchange data over the Internet. In this paper, IoT can be understood through this definition which is, a technology that enables communication, sharing, and collection of information with each other through networks and the Internet such as people, objects, and spaces. Honeypots are primarily traps for malicious hackers. Like honey jars that lure bees, the goal is to attract hackers into honeypots and, usually, collecting useful

information. Honeypots are computers or computer systems that simulate the targets of probable cyber-attacks. By design, users use vulnerabilities in honeypots. For example, an administrator connects a honeypot on the network, adds data, and waits as if the computer has sensitive information. The admin can detect an attacker attempting to crack the machine. Also, hackers attacking honeypots to exploit vulnerabilities can reveal the hacking path and hacking techniques to trace back the hacker's information or the location of hacking.

## **2.2) Thesis Statement**

Before the Internet of Things developed, there were many attacks on other devices. However, with the rapid development of the IoT and a significant impact on people's lives, attackers are started to give focus on the IoT devices, and the number of attacks on vulnerabilities in the IoT seems to be far higher than the number of attacks on other devices. Thus, in this paper, we are focusing on attacks against IoT devices with this statement.

IoT devices are the subject of attacks more than other devices.

Through the thesis statement, we will derive the research questions and hypotheses. Therefore, we will see if the thesis statement indeed applies to reality.

## **2.3) Research Questions**

As a result of previous research, each time a new technology is released, the attack is also found as well. So, we started to look for various IoT devices. Among them, there was interest in speakers that are closely related to human life. Smart speakers equipped with Artificial Intelligence (AI) can communicate with people and analyze people's words to perform tasks. Machine learning also enables self-learning and inference, making it smart with data accumulated through continuous use. However, there is a problem because it collects such diverse and vast data. In a recent news article, the AI smart speaker was hacked, and the ability to listen to people was changed to an eavesdropping device. As IoT devices are becoming more common, the number of attacks on them seems to increase. As a result, attention has focused on IoT devices, raising questions about whether IoT devices receive more attacks than other devices. So, while researching AI smart speakers, various questions arose.

- Which IoT device gets attacked most?

- Which ports in profiles get more attacked?
- What types of data are the attackers sending?
- What protocol is standard per device/total?
- Where the IP addresses come from? To what device?
- Attack countries per device
- Are the attackers focusing on specific devices? Are they opportunistic or on purpose?

## 2.4) Hypotheses

Two hypotheses can be tested by answering research questions. one is 'Google Home Mini will get attacks more than other speakers because of the Market share,' and the other is 'Attackers are not focusing on specific devices.'

## 2.5) Conclusions

Based on the definitions of various IoTs, we defined the IoT to be used in the paper. In our doubts from previous researches, we created a thesis statement about whether an IoT device gets more attacked than other devices. Next, IoT honeypots and data analytics are used to validate research questions and hypotheses. Tested research

questions and hypotheses are used to determine whether the thesis statement is indeed being made in reality

## 제 3 장 BACKGROUND RESEARCH

### 3.1) Introduction

IoT devices offer a variety of conveniences to humans. In order to provide services to humans, IoT collects and processes sensitive data such as human status and information, and in the case of smart homes, it is connected to various devices to simplify control. However, its convenience comes with risks. IoT security is as essential as dealing with sensitive data, and prior research has been conducted to find out its vulnerability. In this chapter, we describe so far research related to IoT attacks. Among them, the honeypot is a system that can analyze and predict what attacks are coming into IoT devices and has already been prior research for analyzing attacks using honeypots.

### 3.2) IoT

There are various kinds of IoT [10], which can be classified according to data transmission, the behavior of things, and Artificial Intelligence (AI). In the case of the

first data transmission and reception, it may be divided into a data transmission object collecting and exporting data, a data reception object receiving data from outside, and a data transmission and reception object. Second, the behavior of things is classified as fluidity. A way that things move through programming, like a cleaning robot, a way that things fix in the place and work, like a smart fridge and AI speakers, and another way is that things like a smartwatch, where people can carry it. Lastly, in the case of Artificial Intelligence, it can be defined into two which are, the movement of the things is not only through the programming but itself analyze, process and determine the collected data and the things without AI and move as programmed by a human.

### 3.3) Honeypot

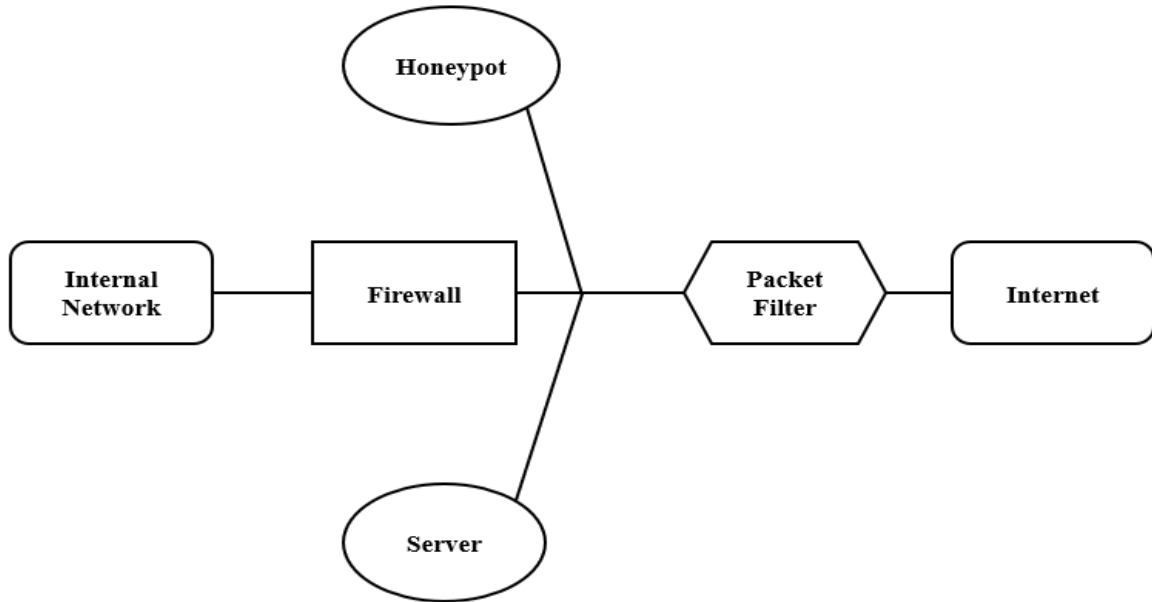


Figure 1 Honeypot configuration [11]

There is an advanced type of honeypot and honeynet. Usually, if a honeypot is a single system for getting hackers' information, then honeynet means a network that includes honeypots [12]. Figure 1 is a standard configuration of honeypot. It is placed above the firewall, emulates an internal network, and gathering the data.

#### 3.3.1 Types of honeypot

Honeypot types [13] can be divided into production honeypot and research honeypot. Production honeypots are typically used to protect organizations such as companies. It

is installed in the company network to be used for overall security and to mitigate risk. Research honeypots are used to study attacker tactics and techniques or to gather information. For example, when a system is compromised, such as a virus, it can collect information about how the attacker is attacking. Then, the honeypot can be classified [12] into two types, low-interaction and high-interaction. Low-interaction honeypots simulate the type of attacker frequently attacks. It is relatively easy to use because it uses fewer resources and does not require much code. High-interaction honeypots are more complicated to install than low-interaction honeypots, and it behaves like a real system. With virtual machines, it can run multiple on one physical computer. It is also easy to recover a system compromised by an attack.

#### 3.3.1.1 Dynamic Honeypot

Also, there is another type of honeypot, called Dynamic Honeypot. Kuwatly et al. [14] introduces the design for intrusion detection system using Dynamic Honeypot. They give a new way of the Dynamic Honeypot to implement in a real network system.

Budiarto et al. [15] discusses about making the honeypots in a simple way and talks about the configuration of Dynamic Honeypot. They introduce the Dynamic Honeypot as the honeypot that can work by plugging in to the network. It decides how they will deploy the environment.

### **3.4) IoT Devices**

A 2018 article [16] describes how vulnerable IoT devices are in Korea. As the number of electronic devices connected to the network increases, hacking aims to steal or harm personal information. In the case of smart speakers, hacker hacks and sends a private conversation to third parties. Internet of things and artificial intelligence devices are targeted for hacking because they have lower computing power than PCs, and this makes hackers easier to hack. AI and IoT are popularized, smart TVs, smart refrigerators, and automated cars on the rise. It means that the damage is expected to spread further.

A 2019 article [17] talks about the National Assembly's audit of the National Science and Technology Information and Telecommunications Broadcasting and Communications Commission pointed out the lack of measures to leak personal

information of Naver, Kakao, and three telecommunications, for AI devices including IoT speakers. Since 2017, the IoT security certification system has been implemented from the perspective of security internalization. However, out of the 17 cases, there are eight light grades and nine primary grades, and 0 international standard security grades. There are 8 million AI speakers, IoT-connected devices, but they criticized for no security certification. As the article describes, IoT devices are increasing, but there is a lack of countermeasures against devices.

### **3.5) IoT Security**

The Trend discussed [18] the five most active cybercrime underground communities in the world. The Russian community took the most live action on IoT-related attacks. Cybercriminals bought and sold the discovered vulnerabilities related to IoT devices on the website. The Portuguese were focused on the router infection called "KL DNS." They were looking for an opportunity to make attacks similar to router mass infection in Brazil in 2018. Next, the English hacking community talked about the specific vulnerability of Netgear routers and had actual codes for exploitation. Also, showing some interest in connected printers. For the Arabic community, they expressed their

interest by sharing the latest news about discoveries of IoT vulnerabilities. Lastly, the Spanish community showed interest in not protected and not authenticated devices to find the entrance for new attacks. For example, they are using Google dork to look for industrial refrigerators that are not protected. As the IoT industry is getting bigger, the attacker's interests are moving into IoT a lot. Thus, to prevent damage to it, some actions are needed.

According to the Korea Internet & Security Agency (KISA) [19], in 2019, reports of IoT-related security vulnerabilities are increasing every year. The number of reports is rapidly increasing from 6 in 2014 to 130 in 2015, 362 in 2016, 347 in 2017, and 387 in 2018. In the first half of 2019, 188 cases were received. The total number of reported IoT security vulnerabilities in 2019 is expected to increase more than the previous year. There are various types of IoT security vulnerabilities reported to KISA. First, exploit vulnerabilities that could allow hackers to gain administrative privileges. Second, vulnerabilities that can bypass security programs. Third, 'Authentication bypass' to access the authority of the IoT administrator page. Fourth, vulnerability to information leakage collected on IoT devices, and lastly, vulnerability to insert malicious commands

into IoT devices to execute malicious code were reported. KISA operates the 'Software (SW) New Vulnerability Reporting Reward', which is notified of IoT security vulnerabilities. At the same time, KISA has been implementing the 'IoT security certification system' to evaluate the security stability of IoT devices since December 2017, but the results are poor. According to the KT Institute for Economic Management, domestic IoT hacking damages are expected to reach 26.70 trillion won in 2030. It means is that it is good to operate the IoT reporting reward system, but it is urgent to prepare a practical plan through simulation and test to the attacks.

### **3.6) IoT Honeypot**

Gandhi et al. [20] aimed to protect the IoT environment by proposing HloTPOT (Honeypot for IoT environment). HloTPOT installed a honeypot and user database on Raspberry Pi and collected data. If the user DB in Raspberry Pi matched with the accessing user, it connected to the real IoT environment. If it did not match, it was considered as an intruder and connected to a fake IoT environment. HloTPOT collected logs and chat details of intruders connected to fake IoT environments. Also, HloTPOT sent a warning about the user to the real IoT environment. The collected logs showed

the intruder entered methodology, and it could be used for further research and as forensic evidence. The paper focused on detecting intruders at IoT devices. It would be helpful if there were more explanations of the result, such as what is the meaning of a graph. Also, detection alone is not enough to create a truly safe and secure IoT environment. A protection plan is also needed.

Anirudh et al. [21] conducted some tests to protect an IoT system from the Denial-of-Service (Dos) by using Honeypot. They proposed two models with scenarios to compare how the Honeypot is useful. The first scenario was using the Intrusion Detection System (IDS). If IDS found an oddity in the client's data, it sent it to a honeypot and collected log information about the attacker. Also, the collected data was managed in a database. The secondary scenario was that there was a log collected in advance, unlike the previous scenario. When a request came to IDS, it checked to see if it matches the client of the data in the log. If the data does not match, block it; if it does pass it. Through the scenarios, using honeypot was more effective than without it. It would be good if the paper talks about what happens when these models run on real cases.

Yin et al. [22] found a significant increase in telnet-based attacks against IoT devices. They implemented a new honeypot called IoT POT that copied IoT devices and captured telnet-based intrusions from several attacks. Additionally, proposed IoT BOX to process captured malware on different CPU architectures to more analysis of threats. Through these analyses, they discovered at least four DDOS malware targeted to IoT devices. This paper implemented the first honeypot for IoT devices based on telnet and tried to cover most of the CPU architectures.

Haris [23] analyzed the Mirai-based attack that happens to the Internet of Things (IoT) by proposing a multi-component solution. They implemented IoT honeypot, which had multi-component that worked with telnet traffic to handle the Mirai attack. The front-end component attracted the attacker's attention by interacting and answering the attacker's input. While the front-end component interacted with the attacker, the back-end component got the encrypted data that was captured and decrypted into readable form. Then, inform the user and saved it forever. After the test, they found out that Mirai did not target device vulnerability. It looked for the weak and default passwords that had never been changed seen it was operated. It would be useful if the

paper uses actual IoT devices and mentions what kind of passwords were vulnerable to this kind of attack.

Meng [24] implemented ThingPot by mimicking a Philips Hue smart lighting system - wireless LED lights bulb and wireless bridge - and with XMPP and REST API. This implementation was focused on the whole IoT platform. Besides, they have offered a Proof-of-Concept (PoC) and provided open-source code for it. The test had run for about one and a half months. During the analysis, the captured data presented that not many attackers were activated on XMPP. It indicated few things that XMPP made attackers hard to get to the device and its platform, or they were not interested in it yet, or the attacker's focus was not in the ThingPot but XMPP server.

On the other hand, attackers showed some interest in REST. They were trying to gain some data about the device and get control of it. The paper focused only on one IoT device but may try the same test to the different devices to have more concrete results.

### 3.7) Conclusion

We talked about the prior research related to the IoT and honeypot. Although there have been several prior researches that deal with sensitive information, there are still many risks. In the next chapter, we will look at the research that has been carried out for the attacks and security that IoT devices can receive.

## 제 4 장 RELATED WORK OF IOT DEVICE SECURITY

### 4.1) Introduction

The previous chapter talked about vulnerabilities in IoT devices, and this chapter talks about the security of IoT devices. With the development of IT technology and the development of IoT technology, the types of devices and services have increased exponentially, and so have the security threats. This chapter will cover what all kinds of possible attacks could be used to IoT, including against computers that can be turned into the IoT on different sides.

### 4.2) Hardware

#### 4.2.1 Debug Pad

Debug is the task or program that, at the end of the program's development, detects the error, and identifies its cause. The debug pad is a pad attached to the device for debugging. When accessed through the pad, it may be possible to access the system of

the device without any authentication. Thus, it makes attackers easy to take control of the device. Barnes [25] conducted a test using a debug pad on Amazon echo, which is an AI speaker. The prior work was that boot echo with a debug pad attached to the bottom of the device. They extended the work from previous work by taking the remote root shell access and was able to obtain the remote control on microphones. By conducting UART, they could see the device boot and configuration. Then, using some command lines, they examined the file system. After that, with the scripts that they wrote, they were able to figure out the interaction between the audio buffers and remote the service. However, since 2017, Amazon changed the structure of the pad on the mainboard to avoid external booting. Thus, this kind of physical vulnerability is available for the 2015 and 2016 version.

#### 4.2.2 Hardware Backdoor

The paper [26] talks about Smart Nest Thermostat, which can be attacked physically through the USB port. The Nest has security for software but hardware. If the attacker has a chance of physical access, it takes only 15 seconds to control the Nest. By

pressing the power button, the device switches to developer mode, inserting a USB drive during reset and loading user firmware that official Nest does not specify. Although a device is infected with a virus, there is no problem in using it, so the user cannot know even if the data on the device is leaked.

## **4.3) Software**

### **4.3.1 OS (Operating System)**

An article in 2016 [27] talks about the Linux kernel system, which can affect a lot to IoT devices. Linux kernel vulnerabilities discovered in 2016 include CVE-2016-0728, CVE-2015-1805, and CVE-2016-5195, called Dirty Cow. Since Android, iOS, and Mac OS are based on Linux, it can be said that most IoT devices have a Linux based OS system. Among other things, CVE-2015-1805, which is Dirty Cow, is dangerous enough to affect 97% of Android devices.

### 4.3.2. Firmware

#### 4.3.2.1 Different types of vulnerabilities and attacks

The article in 2019 [28] introduces eight different vulnerabilities and attacks that can be happened in firmware. first, unauthorized access that an attacker can easily exploit. Second, weak authentication with not strong encryption algorithms. Third, a hidden backdoor that makes the attacker access to the device easily. Forth password hash that is hard for users to change. Fifth, an encryption key that is proved inappropriate to use. Sixth, buffer overflow - give control to an attacker by using insecure coding. Seventh, open-source code - easy to be a target to attackers if there is no regular update, and eighth, debugging service - allows the attacker's internal access through the device.

#### 4.3.2.2 Control-Hijacking

The paper [29] briefly presents about control-hijacking vulnerabilities of IoT firmware that have been widely spread recently and some related work. Also, using metrics, they classify the recent discovered control-hijacking vulnerabilities.

### 4.3.3. Backdoor

In the 2019 article [30], attackers have infected backdoors with live update utilities installed on ASUS new versions of computers. Through this, attackers gained access and identified the targets by using about 600 MAC addresses that they previously had. The attackers then downloaded the malware to a running C&C server when a specific MAC address was found. Then they took ASUS's digital certificates and distributed them to the official update server. The attack affected users who activated ASUS live update utility. In the case of backdoors, manufacturers often plant them for maintenance purposes, such as a network. Therefore, it can be applied not only to computers but also applies to IoT devices with backdoors.

## 4.4) Network

### 4.4.1 Sniffing

In this article [31], it talks about ZigBee-sniffing drone. The Praetorian, which is an information security company, experimented with a drone that can detect the devices

connected to the Internet. Through the experiment, they found that the drone revealed the device's security settings, the manufacturer, and where the device is used, such as commercial or residential.

#### 4.4.2 Port Scanning

In the 2018 article [32], they picked port scanning as one of the network attack types. Port scanning is an attack that precedes the vulnerability of the application being used by the target. It checks the port address to determine the type of service. Major port scan methods include TCP scans, SYN scan, and ICMP message scan. Since all devices send receive data through the port, port scanning can be regarded as all IoT devices. For example, one of the AI speakers, Google Home Mini, has five fixed TCP open ports.

### 4.5) Attack Data

#### 4.5.1 DDOS (Denial-of-service attack)

Igloosecurity [33] talks about one of the DDoS attacks, which is Mirai-botnet. An attacker infected several vulnerable IoT devices with the malware Mirai, which

automatically searched for other vulnerable devices on the Internet and made them act as bots for DDoS attacks. It attacked the Dyn server, a major US Internet hosting company. Dyn caused a series of disruptions at major sites such as GitHub, Twitter, Netflix, and the New York Times that were receiving DNS services.

#### 4.5.2 Kaspersky

After building more than 50 honeypots around the world, Kaspersky [34] has detected 105 million attacks on IoT devices with 276,000 IP addresses. Kaspersky's IoT: Malware Story report contains data on the number of cyberattacks performed over time using honeypot data, the types of attacks used, and where the attacks occurred.

### 4.6) Security Method

#### 4.6.1 Guideline for security

Open Web Application Security Project (OWASP) [35] published principles of IoT security and guidance. Principles of IoT security were written in 2016 and covered the overall IoT system, components, and ecosystem. The IoT security guideline, written in

2017, describes three security guidelines that manufacturers, developers, and consumers must follow to improve the security of IoT products. The three different parts have in stock about IoT device security is restricting physical access. An example is the debug pad at the bottom of the Amazon echo. In the case of the Amazon echo, people could gain access to data and permissions after physical access through the debug pad. Thus, OWASP recommends disabling unnecessary or unused physical ports such as debug pads or USB ports.

#### 4.6.2 Requirements

This paper [36] argues that security features must be reflected from the requirements analysis stage, which is the early stage of development. Also, based on three essential features of the IoT environment: heterogeneity, resource constraints, and dynamic environment, the paper analyzed the security requirements for IoT in six aspects of the IoT environment - IoT networks, IoT clouds, IoT users, IoT attackers, IoT services, and IoT platforms.

### 4.6.3 Platform

The article [37] talks about the IoT platform. The IoT platform is an integrated service that provides the elements needed to bring physical objects online. IoT platforms can be classified into four categories which are, end-to-end platforms, connectivity platforms, cloud platforms, and data platforms. To efficiently build and manage the IoT environment, companies are actively participating in building the IoT platform. However, in terms of security, there are still a few guidelines that can be found other than the security embedded in the IoT platform itself or the protection of the enterprise itself. Therefore, in this article, gives a guide for choosing an IoT platform.

GreenZone Security [38] introduced an end-to-end security platform that can encompass devices, networks, and service platforms in an IoT environment. It is equipped with several security technologies to optimize the IoT device environment, which is characterized by ultra-lightweight, low power, and low performance. They described that it plays an essential role in increasing the IoT security rate.

## 4.7) Conclusion

This chapter covered IoT device security. AS a computer connects to the Internet, IoT devices connect to the Internet. Thus, attacks that the computer can receive, also can be applied to IoT devices. In this paper, we will examine how much of these vulnerabilities and security problems are concentrated on IoT devices. In the next chapter, we will examine the devices used in the paper.

## 제 5 장 DEVICES USED FOR RESEARCH

### 5.1) Introduction

In this chapter, the types of devices used for the research will be described. Various IoT devices exist. Nowadays, smart home appliances such as smart refrigerators, smart TVs, and smart light are easily accessible to people. As a result, many attacks are aimed at people who do not have much security awareness. Therefore, in this paper, we study using AI smart speaker, one of the popular smart home products. As the IoT devices connect to the Internet, it uses TCP and UDP protocols to send and receive the data. Thus, we use two popular protocols, SSH and FTP, for comparison and to verify the data. In addition, we talk about the types of attacks that IoT devices and protocols can possibly get.

## 5.2) AI Smart Speaker

### 5.2.1 SKT Nugu

Nugu [39] is Korea's first AI speaker released by SK Telecom in September 2016. The exact model name is NU100, which can be operated by voice control, power, volume, mute, Bluetooth, voice recognition button, and smartphone application. To use Nugu, a user needs to download the Nugu application on the smartphone. After installing the application, the user can use the smartphone and Wi-Fi connection. In addition to the timer and alarm in Nugu itself, it can also be linked to Melon, Google, and Smart Home. After logging in, the user is logged in to the application continuously unless the user logs out.

### 5.2.2 KT Giga Genie

Giga Genie [40] is an AI speaker launched in 2017 by KT, a telecom company like SKT. The exact model name is KT Giga Genie CT1100. Unlike other speakers, set-top boxes and artificial intelligence are combined. It can be used instead of KT's existing set-top

box, which provides IPTV service, Olleh TV. It can also be used as a home cam by attaching a dedicated camera. It has a remote control and HDMI port so that it can be connected to a display and used as a TV. Other features like search, weather, alarm, and music are like those of existing AI speakers and can control the TV program. To use Giga Genie, the user must log in to the Giga Genie application. In the case of Giga Genie application, KT ID login, smartphone login, Kakao Talk login, Naver login, and Facebook login are available. If the user logs in once, the login is maintained unless the user logs out.

### 5.2.3 Naver Clova

Naver is one of the most used search engines in Korea. Clova [41] was launched in May 2017 as Naver's AI platform. Since then, Naver has introduced a smart speaker equipped with Clova, and various models have been released. The product used in this paper is Friends mini Minions NL-S22KR (Bob). Clova also needs to download the Naver Clova application for the initial configuration to use. Similar to the existing AI speakers, it has essential functions such as news, music, and alarm, and can be linked with other

smart services such as LG SmartThinG. The difference with other speakers is that it is portable, so the user does not have to connect the power supply if the battery is charged.

#### 5.2.4 Google Google Home Mini

The Google Home Mini (GHM) [42] is a smaller version of Google Home that is released by Google in October 2017 and released in 2018 in Korea. It uses Google Assistant as an AI platform. Basic functions such as music, news, and weather can be used, and the smart home service of Korean companies can be synced. Google Home Mini [43] supports different languages, including Korean, English, German, and French. The speaker supports multi-language mode, so when a user asks a question in Korean, it answers in Korean, and if the user asks in English, it answers in English. Also, if the user uses a Google account, the user can receive personal information in the user's account, such as schedules. To use GHM, the user needs the Google Home application, and the user will be logged in unless the user logs out.

### 5.3) SSH (Secure Shell)

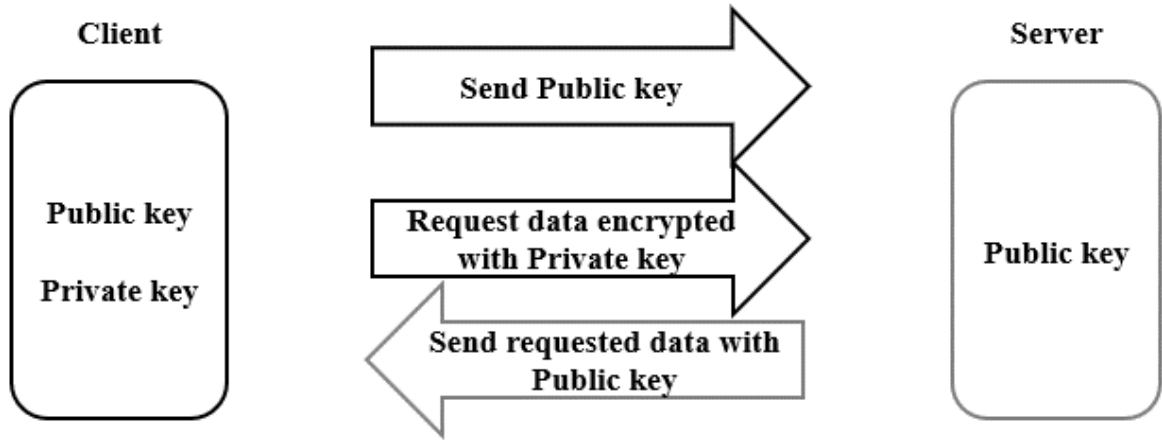


Figure 2 SSH key exchange protocol

SSH stands for Secure Shell. One of the network protocols, a protocol for secure communication when communicating with other computers connected to a network. Because all data over SSH is encrypted and traffic is compressed, the user gets fast transfers. It is more secure than the same network protocols, Telnet, and FTP. It is mainly used for data transmission and remote control. SSH [44] uses asymmetric cryptography (different keys used for encryption and decryption) that authenticate through public and private keys. Moreover, SSH's default port is 22. The way SSH works are shown in Figure 2. First, create a public and private key on the client-side and send the public key to the server. The data that the client wants to request is then encrypted

with the client's private key. Then, the server decrypts the data with the public key and encrypts the information the client wants with the public key. Even if a hacker intercepts the public key in the middle, there is no private key, so the hacker cannot decrypt the data that the server sends. At this point, the critical information is sent from the server to the client. Thus, SSH is safe by making hackers hard to decrypt.

#### 5.4) FTP (File Transfer Protocol)

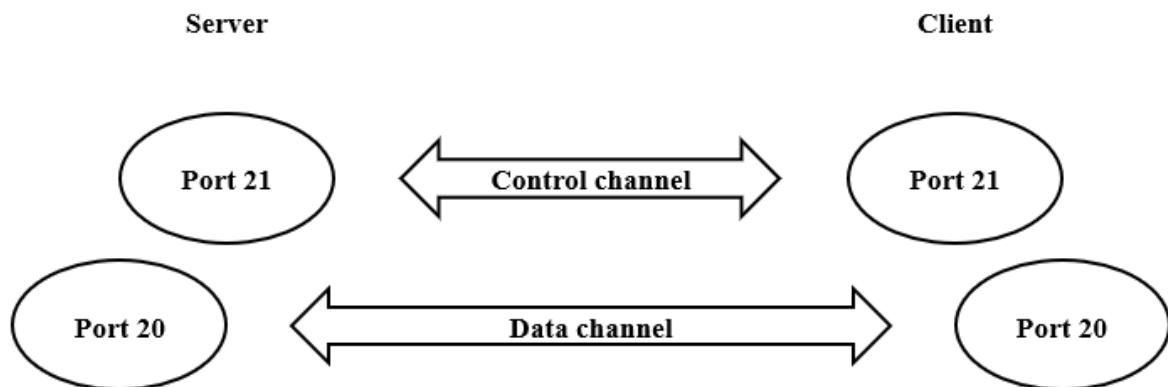


Figure 3 FTP configuration [45]

FTP stands for File Transfer Protocol. It is a TCP/IP protocol for sending and receiving files literally. Suitable for sending and receiving large files over a network. FTP [45] is a command-based protocol. Figure 3 shows the configuration of FTP. It has a control channel (port 21) for sending FTP commands and a data channel (port 20) for

transferring files between the client and server. Corresponding channels are created between the server providing FTP service and the connecting client. The FTP [46] server requires a user account and password to connect. This information is sent and received through the control channel. The actual file transmission and reception takes place over the data channel. FTP can also be used with a web browser or Windows Explorer, but it is more convenient to use an FTP-only client program. Because FTP client programs must send and receive many files in a row, they maintain a connection with the server by sending a persistent response message.

## **5.5) Expected Attack**

### **5.5.1 TCP (Transmission Control Protocol)**

#### **5.5.1.1 SYN Flooding**

SYN Flooding [47] is an attack that causes failure by sending an attacker a large number of SYN packets to the server to fill the server's queue and ignore new client connection requests. In more detail, in the SYN packet transfer phase, which is the first

phase of the 3-Way Handshaking - before an application program that communicates using TCP/IP protocol transmits data-, it establishes a session with the counterpart computer in advance. In order to guarantee the correct transmission, an attacker generates a large number of SYN packets and forwards them to the server. Then the server's backlog queue, which is used to accept TCP connection requests, becomes full, which results in a denial of service condition that causes subsequent connection requests to be ignored. If only the SYN packet is sent and the ACK packet, which is a response to SYN-ACK, is not sent, it is in Half Open mode for 75 seconds, and it continues to send SYN packet to fill the Backlog Queue and no longer receive new TCP connection.

#### 5.5.1.2 Tsunami SYN Flooding

While traditional SYN Flooding attacks, generate 40-60 bytes of traffic per packet, Tsunami SYN Flooding [48] attacks by adding and generating packet traffic with a size of 1000 bytes per packet. This type of DDoS attack uses the TCP protocol rather than UDP.

#### 5.5.1.3 TCP Connection Flooding

TCP Connection Flooding [47] is a type of attack that causes service overload by excessively triggering the TCP 3-Way Handshake process. The server receiving the attack traffic keeps trying to connect the ordinary TCP session to the session. It depletes the session processing resources of the server performing the service so that the regular session connection can no longer be performed. As a result, users who normally access can no longer access the service. It can be divided into three categories.

- DDoS Attacks that maintain TCP session connections
- DDoS Attacks repeating TCP session connection/disconnections
- DDoS Attacks that sends out traffic that looks like a normal transaction after a TCP session connection

#### 5.5.1.4 HTTP GET Flooding

In the case of the TCP Connection Flooding described above, normal transactions do not occur after the TCP 3-Way Handshake process. In contrast, HTTP GET Flooding [48] is a DDoS attack technique in which an additional standard transaction occurs after the TCP 3-Way Handshake process. Since the server receiving the attack traffic continuously requests the standard HTTP Get request along with the regular TCP session, the server performing the service must perform not only necessary TCP session processing but also HTTP request processing. It may cause an overloading of the HTTP processing module.

#### 5.5.1.5 SlowLoris

The attacker [49] maintains an open connection by requesting abnormal (incomplete) header value to a server after connecting to the target server and establishing a regular session. The standard header completes with 0d0a (CRLF), but SlowLoris sends an abnormal header value of 0d. The server determines that the transmission of the

header has not been completed and continues to maintain the connection. The steps are below:

- Send a GET request after a session is connected
- Send an incomplete request and keep an open connection
- The server will wait for the header
- The server enters DOS state depending on the number of connections
- Do not end because the header of the request is [0d0a0d0a]

#### 5.5.1.6 SlowRead

It [48] is a type of HTTP attack that delays TCP connection by slowly reading a response by manipulating buffer size and TCP window size. Take advantage of the fact that the webserver does not limit connection delays. The difference between SlowLoris and SlowRead is that it is holding the session longer, sending the HTTP request correctly, and reading the response slowly, rather than delaying the request. It is a way to continually delay a TCP connection in the data flow by manipulating the value of the TCP window size and receiving '0' or small data. The attacker and the target server

occupy connection support until the data transmission is completed. If this process occurs a lot, the connection resources of the target server are exhausted, and the service is denied.

## 5.5.2 UDP (User Datagram Protocol)

### 5.5.2.1 UDP Flooding

UDP Flooding [50] is a type of DoS attack in which a large number of UDP packets are sent to a user to make it impossible to use the standard service. Since UDP packets use spoofed IPs and ports, it is difficult to block them using IP filters. Because it consumes network bandwidth, all services of users, not specific services, are disabled, and there is no need for specific ports to be open. As with any flooding attack, there are no singularities or patterns in the packet itself, making it difficult to block. In the case of the UDP attack, unlike the SYN flood, the purpose is to consume network bandwidth. Therefore, since a single host is not valid, so the attack is configured by DDoS.

### 5.5.2.2 Valve Source Engine Flooding

Valve source engine flooding [48] is UDP (amplification) attacks that are used to consume resources available to the server. The attack is designed to send TSource engine query requests to the game server, which means that the server cannot handle all the requests and handles many of the requests that make the game denial of service. This type of attack only applies to the gamers market.

### 5.5.3 SSH (Secure Shell)

#### 5.5.3.1 Brute Force

It [51] is an attack that attempts to access SSH by indiscriminate ID and password substitution. It is a one-dimensional and simple assignment attack, but it is a compelling and intuitive way to penetrate all the possible keys until it finds the key. Similarly, there is a dictionary attack that matches words in a dictionary file, and a hybrid attack that adds numbers or special characters to words in a dictionary attack.

#### 5.5.3.2 GoScanSSH

GoScanSSH [51] is a malicious code that spreads after infecting an SSH server on a Linux system that is exposed online. The malware sets Raspberry Pi, Open Embedded Linux, OpenLEEC, and Huawei as major target systems to perform random attacks with about 7,000 account/password combinations. The malicious code randomly generates IPs, checks specific IP bands and domains, and scans additional infection targets except for government and military organizations. It performs random assignment attacks on port 22 of randomly generated IP and transmits system and login related information to the C2 server. When sending information, they use the Tor Web Proxy service to make it difficult to track. After successful login, upload, and run GoScanSSH malware to perform additional attacks.

#### 5.5.4 FTP (File Transfer Protocol)

FTP has the vulnerabilities that FTP does not encrypt user authentication information and vulnerability that exploits the characteristics of the FTP protocol.

#### 5.5.4.1 Bounce Attack

FTP Bounce attack [52] is a method of exploiting loopholes in the FTP protocol structure. In more detail, it is an attack that exploits structural weaknesses in FTP design that use control channels and data channels differently and do not identify the destination when creating a data channel. Using an anonymous FTP server, the attacker can manipulate the port command to scan the target network and have the FTP server send data to the attacker's destination.

#### 5.5.4.2 TFTP (Trivial FTP) Attack

TFTP [52] is a simple file transfer protocol application that can be installed on a workstation without a read-only memory or disk. Primarily used for delivering boot images to workstations that do not have their disk. There is a security vulnerability that uses the 69 UDP port and can access the specified directory without any authentication process. In TFTP attacks, if the access control is not properly controlled, an attacker can access arbitrary files by exploiting a weakness in TFTP.

#### 5.5.4.3 Anonymous FTP Attack

Anonymous FTP [53] service is a service that allows FTP access with an anonymous account. An anonymous account is an account with an ID of anonymous and no password (or any password). Allowing such anonymous accounts in public corporations or public institutions can cause serious security problems because any unauthorized user can access the server. If anonymous users even must write access, the attacker can upload malware and cause damage to multiple users.

### 5.6) Conclusion

This chapter talked about devices and protocols to be used for the research. AI smart speakers and two protocols are well known to the public. Thus, many attacks are expected. Based on this knowledge, we will be able to expect the data from the research.

## 제 6 장 IOT HONEYPOT DATA COLLECTION METHODOLOGY

### 6.1) Introduction

From the prior research, we have examined honeypots used in IoT. However, most of them used only honeypots to research, so the research using the IoT device directly was hard to find. It was also hard to find a way that fits the research we want. So, we made our own IoT honeypot for our research. In this chapter, we describe an IoT Honeypot to collect the data by pretending as AI smart speaker. Also, the expected attacks that we can get during the data collection.

### 6.2) Necessity of IoT Honeypot

Many honeypots existed. All the honeypots only deal with specific protocols. However, the honeypot that we need is different from the current honeypot. We need the data of connection to the honeypot, but there is unlikely existed. Thus, we make the

first honeypot software, called IoT Honeypot, where we can load up the profiles and look like multiple devices. It does not have a fake virtual environment. It can emulate target device open TCP/UDP ports, and a profile defines what port and protocol to open. This allows us to quickly make a honeypot that looks like any real, internet-connected device. Also, it can create many different honeypots to compare attacks against different device types.

### **6.3) The design of IoT Honeypot**

The difference between the IoT Honeypot and the ordinary honeypot is that the IoT Honeypot does not respond to the protocols, only for open ports. The ordinary honeypot responds typically to the protocols and makes the virtual environment to let attackers coming in and hack the environment, but the IoT Honeypot is not. All we are interested in is how many times attackers are connecting to the port, not the full honeypot environment for each device because each device has a different system.

## 6.4) Code of IoT Honeypot

To make an IoT Honeypot, the GO language is in use. GO was first released in 2007 and officially announced in 2009 for Linux and Mac OS X platforms. In 2012, GO version 1.0 was released, and as of 2019, the latest version is GO 1.13.4. GO is a general-purpose programming language that follows a traditional compilation and linking model. GO was developed primarily for system programming and drew on the best of C++, Java, and Python. Like C++, GO is compiled through a compiler and is a statically-typed language. It is aimed at a simple and concise programming language and can be multi-processed. The code for IoT Honeypot is in Appendix A. Additional comments have been used between the codes with '//.'

## 6.5) IoT Honeypot Methodology

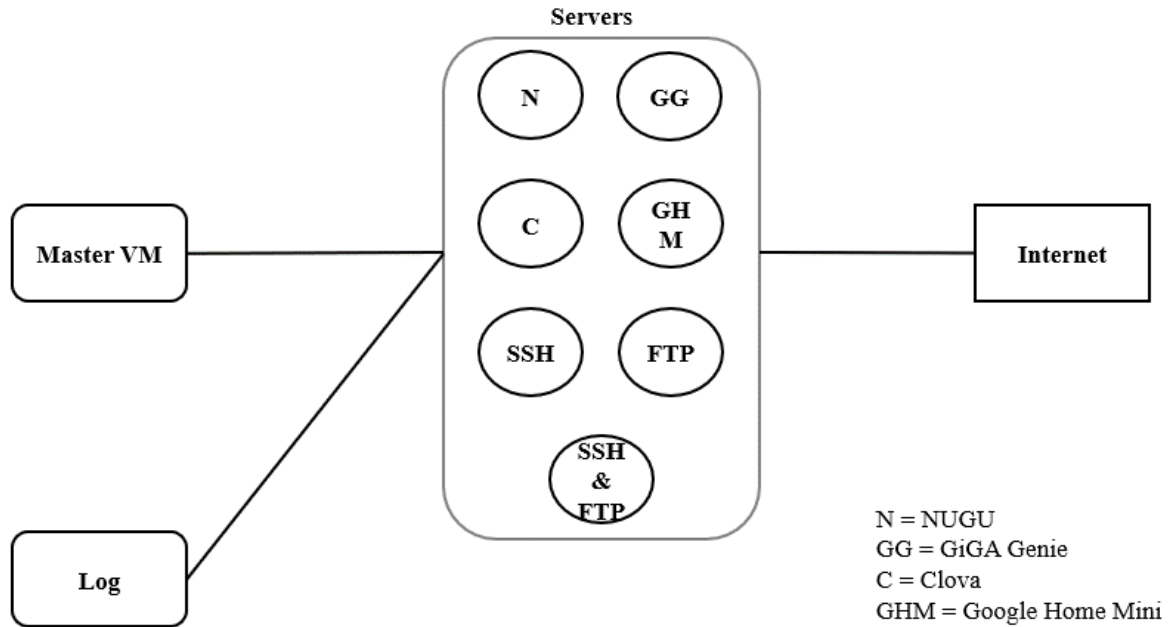


Figure 4 Diagram of IoT Honeypot setup

In this paper, honeypot, four AI smart speakers (Nugu, Giga Genie, Clova, and Google Home Mini) and two protocols (SSH and FTP) will be in use for the experiment. Figure 4 shows the setup of the IoT Honeypot. First, connect the speakers and laptop to the same access point to see what and which open port does speaker has. To check the open port, the GUI version of Nmap, Zenmap (version 7.70), is in use.

## 6.6) Devices

### 6.6.1 Giga Genie

Giga Genie has four TCP open ports, which are 7557, 7547, 8058, and 38520. Among these ports, 7547 is expected to be attacked most. The reason is that port 7547, which is called CWMP (CPE WAN Management Protocol) and known as TR-069(Technical Report 069). TR-069 is an Internet protocol based on XML and SOAP. This port allows the ISP to manage the router remotely. As a result, most routers operate with 7547 ports open by the TR-064 and TR-069 standards set by the Broadband Forum, for high-speed Internet. Attackers can use this to attack through the port. For example, in 2017 [54], foreign hackers hijacked vulnerable home routers and attacked WordPress sites. After analyzing vulnerabilities on routers and TR-069 provided by Korean telecommunications companies KT and LG, it was reported that the patch was not completed. So KT's Giga Genie, which uses port 7547, is expected to get most of the attacks than other ports.

### 6.6.2 Nugu

Nugu has UDP ports opened. Thus, it is expected to get UDP attacks that will be mentioned in 6.7) Procedures

### 6.5.3 Clova

Clova has UDP ports opened. Thus, it is expected to get UDP attacks that will be mentioned in 6.7) Procedures

### 6.6.4 Google Home Mini

Google Home Mini uses AJP (Apache Jserv Protocol) port (8009) and HTTPS (8443) port, where the vulnerability was found. Port 8009 is a port commonly used for Apache Tomcat. Apache Tomcat [55] is an open-source web server and servlet system that uses Java EE platforms such as Java Servlet, JavaServer Pages (JSP), Express Language, and Web Sockets to provide a pure Java HTTP web server environment. A recently discovered vulnerability is the Remote Code Execution Vulnerability (CVE-2019-0232), which occurs when running on Windows with enableCmdLineArguments enabled. It is a

bug in the Common Gateway Interface (CGI) servlet in which the Java Runtime Environment passes command arguments. The patch for this vulnerability has been updated. However, Apache Tomcat is a port that is likely to be attacked since it has been the target of attackers for years to date. Port 8443 is also used by Apache Tomcat and is usually used when configuring SSL. The port is vulnerable to an attack called Heartbleed, which is a web attack that exploits a vulnerability in OpenSSL. The maximum amount of memory that a client can request from the server is 64KB. If attackers request this information little by little and collect the letters, they can get useful information. As Google Home Mini uses these ports, the above attacks are expected to be found.

## 6.7) Procedures





				Clova
				udp,773
				udp,1019
				udp,2967
				udp,17762
				udp,19650
				udp,20217
				udp,20678
			Nugu	udp,21318
			udp,1072	udp,21524
			udp,1782	udp,21784
			udp,2048	udp,26720
			udp,17424	udp,32777
		GHM	udp,19294	udp,34892
		tcp,8008	udp,19605	udp,39217
	Giga Genie	tcp,8009	udp,32776	udp,41638
	tcp,7547	tcp,8012	udp,43967	udp,45818
	tcp,7557	tcp,8443	udp,44334	udp,45818
	tcp,8058	tcp,9000	udp,49194	udp,48189
	tcp,38520	tcp,10001	udp,58631	udp,49226
				udp,50612
 Clova.pro				
 GHM.pro				
 GigaGenie.pro				
 Nugu.pro				

Figure 5 Profile files with open ports and protocol types

Once all the open ports are gathered, make it as profile file with each device's open ports and put it in the same folder like in Figure 5. Next, install a virtual box in the server computer for each speaker and protocols to use as a honeypot. At this point, the computer and the VirtualBox should be in public IP addresses so anyone can connect through the open ports. Then, install a program that can run the code. For the research, visual studio code has been used. When all the installations are done, run the code with profiles (ex. `sudo go run ballygul.go GigaGenie`). Currently, Nugu and Clova

use UDP port, Giga Genie, and GHM uses TCP port. SSH and FTP use the port they initially used. Next, a TCP dump will capture the packet and save it as a pcap file. After that, the collected data will be analyzed.

## **6.8) Conclusion**

In the chapter, we talked about an IoT Honeypot. We wanted to know if an IoT device would get more attacked than other devices, but the IoT honeypot used in the prior study did not have that technology. Thus, we made our honeypot. Then, we anticipated the attack that would be received while we launched the IoT Honeypot.

## 제 7 장 DISCUSSION

### 7.1) Introduction

In this chapter, we analyze the data that we collect by using the IoT Honeypot. The IoT Honeypot operated from November 21st to December 16th, and approximately 2 GB for each device packets have been gathered. Through the analysis, we can answer the research questions. All the packets from that port can be seen as an attack on an IoT device. As mentioned earlier, the device pretending by the IoT Honeypot does not have an environment that attackers can work like other normal honeypots. Because we only want connection data, so the IoT Honeypot device accepts the connection but nowhere to play on. Thus, anyone keeps talking to the device, and they are attacking because nobody should talk multiple times to the devices. Also, if the same IP address connects to multiple times, it is absolutely attacking.

## 7.2) Quantitative Analysis

The analysis will be conducted by answering each question one by one. The first question is, 'What IoT devices get the attack most?'

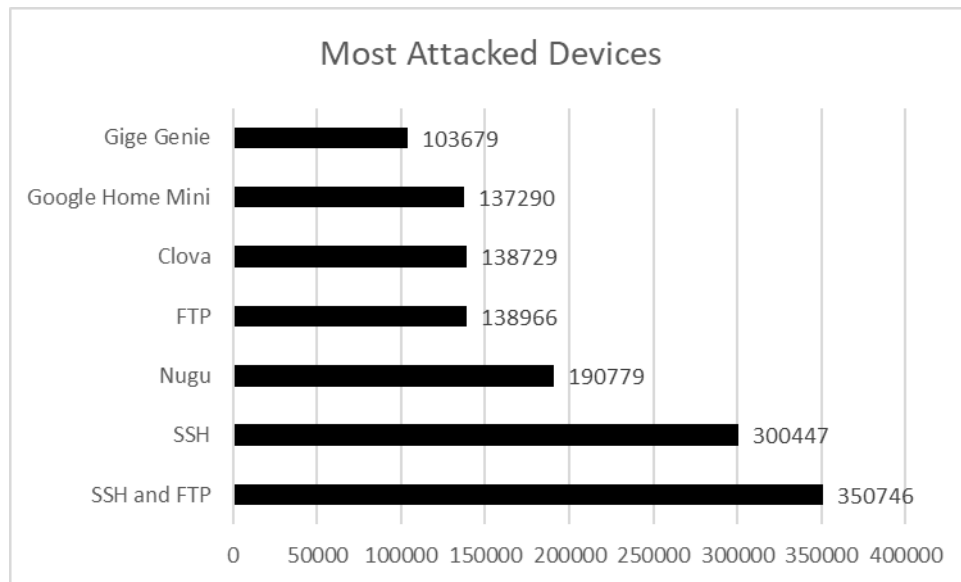


Figure 6 Most Attacked Devices

In Figure 6, the graph shows the most attacked devices during the research. It is counted with the packets they get from outside to their public IP addresses. In addition, it includes the packet from all open ports, not only port that we opened. It disproves that one of the hypotheses, which is 'Google Home Mini, will get attack more than other speakers because of the Market share.' Also, looking at the number of attacks does not

have a big difference. It can support the other hypothesis which is, 'Attackers are not focusing on specific devices'

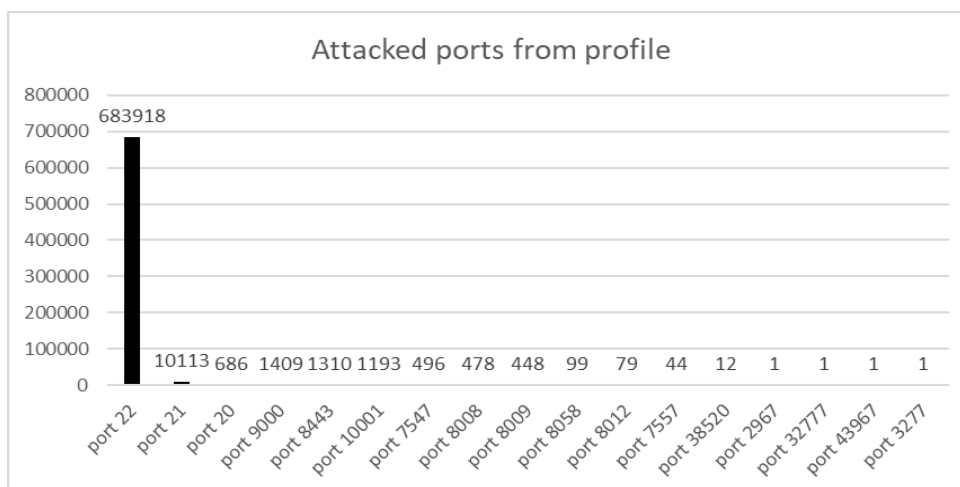


Figure 7 Attacked ports from profiles

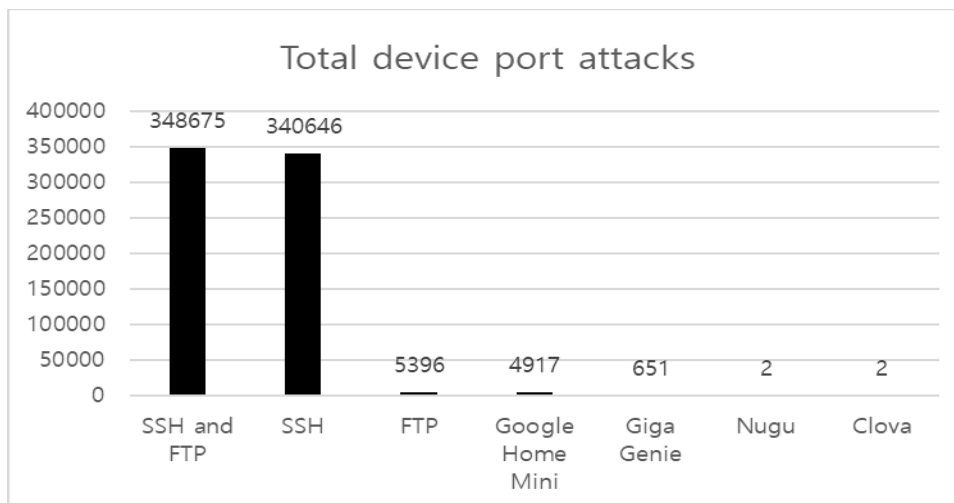


Figure 8 Total number of attacks to device port

The second question is, 'Which ports in profiles get more attacked?'. Figure 7 and 8 can be mean as how many people are trying to connect to the Honeypot. By looking it from the low to high, Clova and Nugu get two connections each into the UDP ports they opened as honeypots. The Giga Genie get 651 attacks, Google Home Mini get 4917 attacks, and FTP gets 5396 attacks. Through the port that we opened, Google Home Mini get the most attacks. As we mentioned previously, SSH, SSH, and FTP are well-known protocols, so they get incredibly high attacks through the port. Among the well-known protocols, SSH 22 port gets the most attacks. To check the data reliability, we compared the data from the study Abdou et al. [56] conducted using the SSH protocol. In the paper, they recorded the number of attempts to connect to SSH using a virtual machine. The research used six virtual machines. They conducted research for 373 days. By dividing total attempts per day, there were 8366 attempts. When we assume that we conduct research for 373 days, there are 12018 attempts per day. Because the previous research had six different attempts, and the average range among them were 26610 attempts to 1131 attempts. As our attempts are in the range so the data can be seen as reliable.

546_	44507.238321	198.108.67.48	210.115.255.247	TCP	66	30942 → 9000	[ACK] Seq=1 Ack=1 Win=29696 Len=0 TSval=667819953 TSecr=3208457...
546_	44507.252067	198.108.67.48	210.115.255.247	TCP	68	30942 → 9000	[PSH, ACK] Seq=1 Ack=1 Win=29696 Len=2 TSval=667819967 TSecr=32...
546_	44508.252135	198.108.67.48	210.115.255.247	TCP	66	30942 → 9000	[RST, ACK] Seq=3 Ack=1 Win=29696 Len=0 TSval=667820967 TSecr=32...
546_	44508.253699	198.108.67.48	210.115.255.247	TCP	74	55504 → 9000	[SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=66782096...
546_	44508.445255	198.108.67.48	210.115.255.247	TCP	66	55504 → 9000	[ACK] Seq=1 Ack=1 Win=29696 Len=0 TSval=667821158 TSecr=3208458...
546_	44509.522674	198.108.67.48	210.115.255.247	TCP	74	40166 → 9000	[SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=66782223...
546_	44509.523943	198.108.67.48	210.115.255.247	TCP	66	55504 → 9000	[RST, ACK] Seq=47 Ack=1 Win=29696 Len=0 TSval=667822237 TSecr=3...

Figure 9 Attackers attempt to connect to the Google Home Mini.

190_	1879.463204	35.154.90.185	210.115.255.246	ICMP	60	Echo (ping) request	id=0x001b, seq=15071/57146, ttl=224 (reply in 19002)
190_	1884.531770	13.235.49.136	210.115.255.246	ICMP	60	Echo (ping) request	id=0x0015, seq=2948/33803, ttl=224 (reply in 19045)
190_	1885.567949	13.235.49.136	210.115.255.246	ICMP	60	Echo (ping) request	id=0x0015, seq=7434/2589, ttl=224 (reply in 19056)
190_	1886.995117	3.10.221.34	210.115.255.246	ICMP	98	Echo (ping) request	id=0x000e, seq=5537/41237, ttl=22 (reply in 19072)
190_	1888.387343	13.233.194.24	210.115.255.246	ICMP	60	Echo (ping) request	id=0x0011, seq=13739/43829, ttl=224 (reply in 19085)
190_	1888.501874	3.10.214.203	210.115.255.246	ICMP	98	Echo (ping) request	id=0x001d, seq=10474/59944, ttl=23 (reply in 19088)
192_	1903.178642	3.10.221.34	210.115.255.246	ICMP	98	Echo (ping) request	id=0x0001, seq=23256/55386, ttl=26 (reply in 19229)

Figure 10 Attackers keep sending pings to the IoT Honeypot

The third question is, 'What types of data are the attackers sending?'. Figure 9 shows the attacker's attempt to connect to the Google Home Mini. It is not only the situation for Google Home Mini. We can find it from other honeypot data as well. As we mentioned previous, we do answer back when the attacker sends a packet. However, we do not have an environment. Thus, attackers think we do have an environment, so attempt to connect to the environment of honeypot. Figure 10 shows that attackers are checking whether it is a real device or not. We usually use ping to check whether the subject is working or not. Thus, it seems like attackers verify that the device is working and connecting to the device.

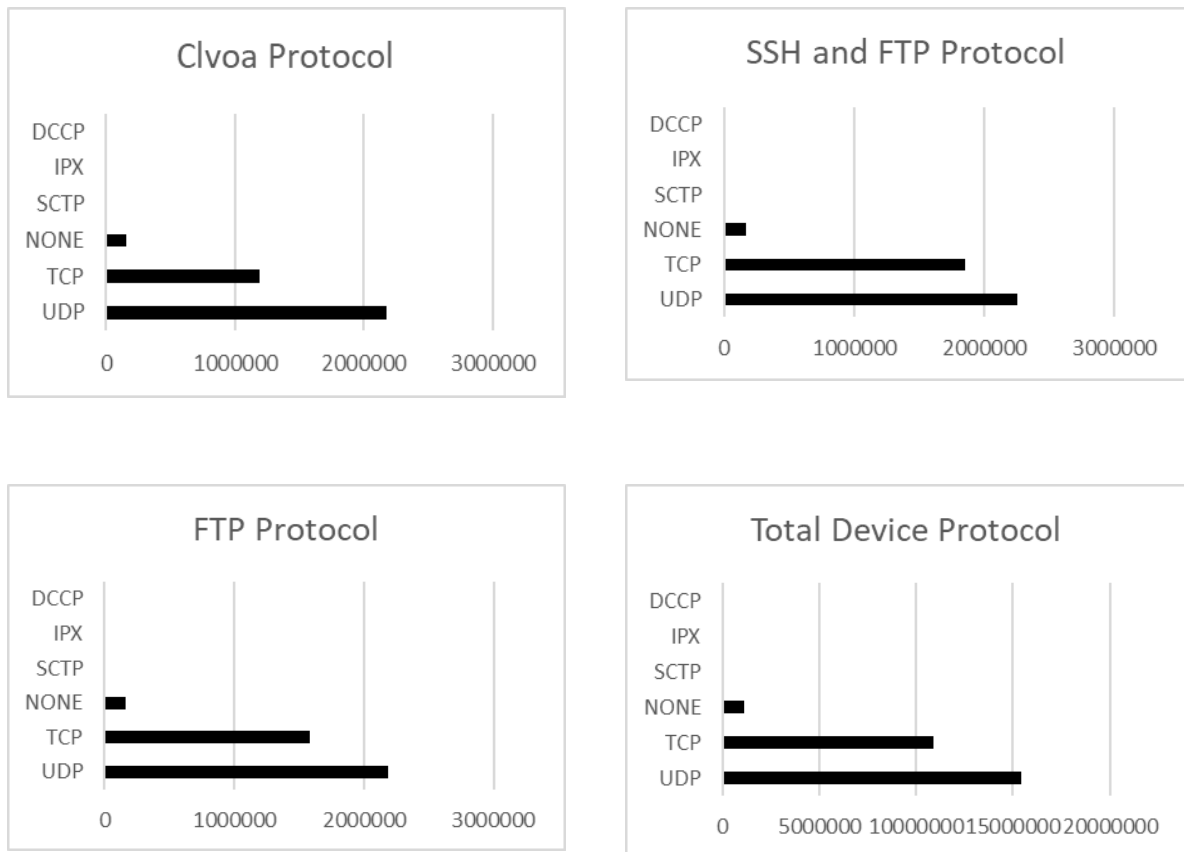


Figure 11 Common protocols of devices 1

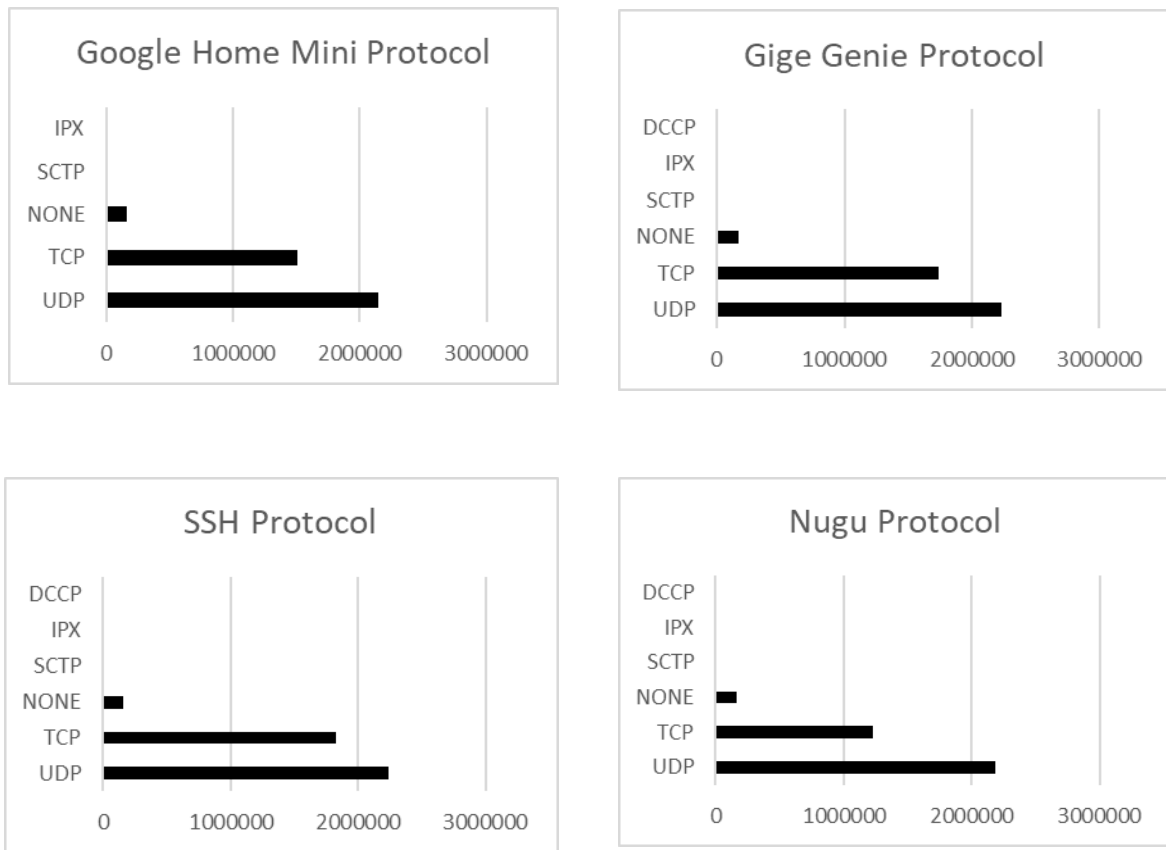


Figure 12 Common protocols of devices 2

The fourth question is, 'What protocol is common per device/total?'. In Figure 11 and 12, it shows protocol types for each device and in total. For each device, the UDP protocol is the highest, and for total, also UDP is the highest standard protocol that devices are using.

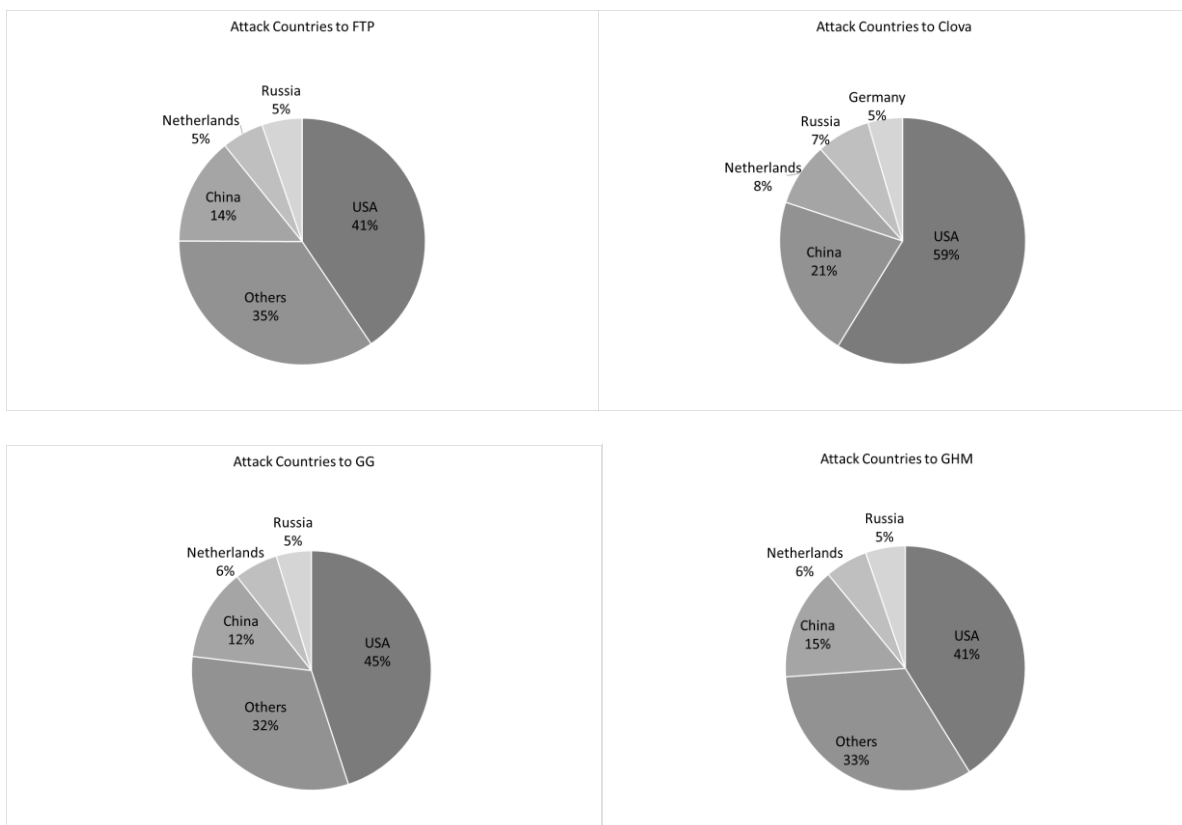


Figure 13 Attack Countries to device 1

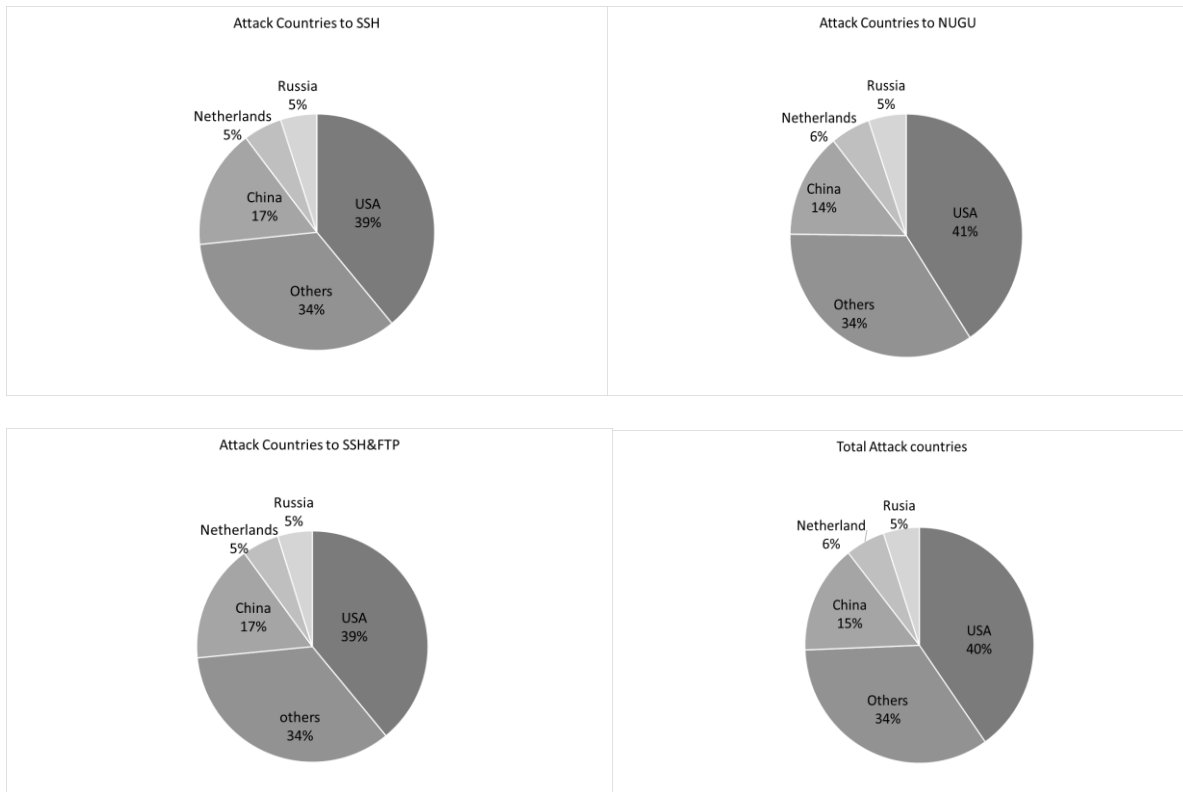
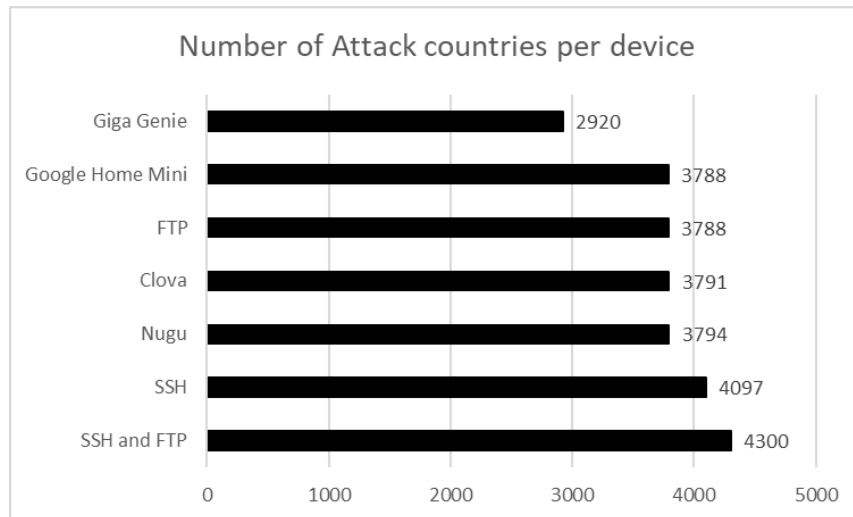


Figure 14 Attack Countries to device 2



**Figure 15 Number of Attack countries per device**

The fifth and sixth questions are 'Where the IP comes from? To what device?' and 'Attack countries per device' can be answered at the same time. Figure 13 and 14 shows filtering the IP addresses from the data and convert it to the country. In the percentage, less than 3% is classified in Others. The top 3 countries are the USA, China, and the Netherlands. Figure 15 shows the number of attack countries per device. It does not have a significant difference among devices. Thus, it supports one hypothesis, which is 'Attack is not focusing on a specific device.'

The seventh and eighth questions are 'Are attackers focusing on specific devices?' and 'Are they opportunistic? Or on purpose?'. Given the data we have analyzed so far, it is hard to see attackers focusing on a particular device because the number of packets coming into the device is not much different. Also, because the attackers tried to connect to the ports other than the one, we opened, it is more opportunistic than on purpose. However, multiple connections to devices can be seen as deliberate attacks.

### **7.3) Conclusion**

In the discussion, the data obtained from the IoT Honeypot was analyzed. For 25 days, about 2 GB of data was collected. With the data, we have answered the research questions and hypotheses. Through the analysis, the hypothesis is disproved that 'the Google Home Mini is more attacked than other speakers because of the Market share.' The reason is that the data coming into the open port was more of Google Home Mini, but when we checked in Figure 6 that the attacks of total data, Google Home Mini was similar or less than other speakers. Overall, it turns out that the Google Home Mini is not hypothesized that the market share is more attacked than other speakers. Also,

there was no attack targeting a specific port and device described previously, only a persistent connection attempt. There were many attacks that came into the port that was initially opened, rather than the port opened through the IoT Honeypot. In addition, the size of the data collected during the same period is not much different among devices. This proves that attackers do not focus on specific devices.

## 제 8 장 CONCLUSION

### 8.1) Introduction

In the early days, IoT, which was known only as of the concept of connecting people and things, and connecting things and things, is developing. As technology advances, more vulnerabilities are found. However, the analysis of vulnerabilities is not much compared to the technology developed rapidly. In this paper, we analyzed the vulnerabilities of IoT devices through AI smart speakers that are popular among IoT devices. To help with the research, IoT Honeypot was written in the Go language. As a result, there was no attack on a specific device, but we could confirm the attempt to connect to the device continuously, and we disprove one of the hypotheses that Google Home mini gets attacked more than other speakers. This means that market share has no impact. In addition, the size of collected data during the same period is similar among devices. This implies that attackers do not focus on specific devices.

## 8.2) Conclusion

Based on what we have seen, we have not done every single device obviously, but the devices that we have created an IoT Honeypot for, we either seeing that IoT devices are focused or not focused, and we have proven the two hypotheses. Among the AI smart speakers, the Google Home Mini was not the most attacked. Thus, our hypothesis is disproved. In addition, the same IP address was found to try to connect to the device multiple times. This can be seen as an intention by attackers to try to attack the device. However, the number is lower than the other two popular protocols, so it is hard to think as a device-intensive attack. Also, the IoT Honeypot can not only open specific ports but also collect packets from open ports, so when an incident occurs, people can use this honeypot to see which attacks come in and how many.

## 8.3) Future Work

All we are focused on is not only the speaker. It can be expanded to a smart home speaker. As the IoT devices are connected to each other to work, one smart home

device can affect all smart home devices. Therefore, in the future, none speaker IoT devices or smart industry IoT devices need to be done with this method. To find out the vulnerability of new IoT devices.

## REFERENCE

- [1] "Internet of Things Becomes Life! | Ministry of Science and Technology Information and Communication Webzine September 2018". [Online]. Available at: <https://www.msit.go.kr/webzine/posts.do?postIdx=350>.
- [2] "190702 Release of the 4<sup>th</sup> Industrial Revolution Indicator\_1.pdf".
- [3] Seong Hyun Na, "Personal Information Protection Issues in IoT Environment", KISDI, ISSN 2233-6583, 8 2015.
- [4] Jung Jin-wook, "Don't confess even you are lonely. Secret 'leaks' AI Speaker", *MBC NEWS*, 08-10-2019. [Online]. Available at: [http://imnews.imbc.com/replay/2019/nwdesk/article/5536689\\_24634.html](http://imnews.imbc.com/replay/2019/nwdesk/article/5536689_24634.html).
- [5] "Lasers Can Hack Voice Assistants in Example Worthy of Mission Impossible But the Risk is Minimal for Consumers", *Voicebot.ai*, 05-11-2019. [Online]. Available at: <https://voicebot.ai/2019/11/05/lasers-can-hack-voice-assistants-study/>.
- [6] Heo Jong-Oh, "A Study on the Construction of Global Honeypot System for Malicious Code Collection", *Journal of the Korean Information Science Society*, vol 37, 1D, pp 36-41, 6 2010.
- [7] "What is internet of things (IoT)? - Definition from WhatIs.com", *IoT Agenda*. [Online]. Available at: <https://internetofthingsagenda.techtarget.com/definition/Internet-of-Things-IoT>.
- [8] "What is the Internet of Things, and how does it work?", *Internet of Things blog*, 17-11-2016. [Online]. Available at: <https://www.ibm.com/blogs/internet-of-things/what-is-the-iot/>.
- [9] "What is Internet of Things (IoT)? | IoT Technology", *SAP*. [Online]. Available at: <https://www.sap.com/korea/trends/internet-of-things.html>.
- [10] "IoT to open a new world (Internet of Things)". [Online]. Available at: <https://brunch.co.kr/@brunchjwshim/54>.
- [11] "What is Honeypot and how does it improve security ?", *The Security Buddy*, 02-3-2017. [Online]. Available at: <https://www.thesecuritybuddy.com/data-breaches-prevention/what-is-honeypot/>.

- [12] “Understanding ‘Traps for hackers’ honeypots”, *ITWorld Korea*. [Online]. Available at: <http://www.itworld.co.kr/news/120233>.
- [13] “What is a honeypot and how can I protect my computer system?-Tip”. [Online]. Available at: <https://ko.play-and-more.com/8933-what-are-honeypots>.
- [14] I. Kuwatly, M. Sraj, Z. Al Masri and H. Artail, “A dynamic honeypot design for intrusion detection”, in *The IEEE/ACS International Conference on Pervasive Services, 2004. ICPS 2004. Proceedings.*, 2004, pp 95-104, doi: 10.1109/PERSER.2004.1356776.
- [15] R. Budiarto, A. Samsudin, C. W. Heong and S. Noori, “Honeypots: Why We Need A Dynamics Honeypots?”, School of Computer Sciences Universiti Sains Malaysia, ResearchGate, 2004.
- [16] “Strange document was printed...AI·IoT security still has a hole | Hankyung.com”. [Online]. Available at: <https://www.hankyung.com/it/article/201812289502g>.
- [17] “[International] ‘0 International Class AI Speakers’...IoT Urgently Needs Internalization of Security - Digital Today (DigitalToday)”. [Online]. Available at: <http://www.digitaltoday.co.kr/news/articleView.html?idxno=215582>.
- [18] “IoT Attack Opportunities Seen in the Cybercrime Underground - TrendLabs Security Intelligence Blog”. [Online]. Available at: <https://blog.trendmicro.com/trendlabs-security-intelligence/iot-attack-opportunities-seen-in-the-cybercrime-underground/>.
- [19] “delighIT.net 2.1”. [Online]. Available at: <http://delighit.net/post/get/34/14214/>.
- [20] U. D. Gandhi, P. M. Kumar, R. Varatharajan, G. Manogaran, R. Sundarasekar and S. Kadu, “HIoTPOT: Surveillance on IoT Devices against Recent Threats”, *Wirel. Pers. Commun.*, vol 103, No. 2, pp 1179-1194, 11 2018, doi: 10.1007/s11277-018-5307-3.
- [21] M. Anirudh, S. A. Thileeban and D. J. Nallathambi, “Use of honeypots for mitigating DoS attacks targeted on IoT networks”, in *2017 International Conference on Computer, Communication and Signal Processing (ICCCSP)*, 2017, pp 1-4, doi: 10.1109/ICCCSP.2017.7944057.
- [22] “IoT POT: Analysing the Rise of IoT Compromises | USENIX”. [Online]. Available at: <https://www.usenix.org/conference/woot15/workshop-program/presentation/pa>.

- [23] H. Šemić and S. Mrdovic, “IoT honeypot: A multi-component solution for handling manual and Mirai-based attacks”, in *2017 25th Telecommunication Forum (TELFOR)*, 2017, pp 1–4, doi: 10.1109/TELFOR.2017.8249458.
- [24] “[1807.04114] ThingPot: an interactive Internet-of-Things honeypot”. [Online]. Available at: <https://arxiv.org/abs/1807.04114>.
- [25] “Alexa, are you listening?”, *F-Secure Labs*. [Online]. Available at: <https://labs.f-secure.com/archive/alexa-are-you-listening/>.
- [26] “Black Hat: Nest thermostat turned into a smart spy in 15 seconds | Computerworld”. [Online]. Available at: <https://www.computerworld.com/article/2476599/black-hat-nest-thermostat-turned-into-a-smart-spy-in-15-seconds.html>.
- [27] “Do Current OS Vulnerabilities Affect IoT? | IoT Security”. [Online]. Available at: <https://www.trendmicro.com/us/iot-security/news/2208>.
- [28] “Unsecured IoT: 8 Ways Hackers Exploit Firmware Vulnerabilities”, *Dark Reading*. [Online]. Available at: <https://www.darkreading.com/risk/unsecured-iot-8-ways-hackers-exploit-firmware-vulnerabilities/a/d-id/1335564>.
- [29] A. Mohanty, I. Obaidat, F. Yilmaz and M. Sridhar, “Control-hijacking Vulnerabilities in IoT Firmware: A Brief Survey”, p 4.
- [30] “More than 1 million ASUS computers hacked. domestic damage is ‘yet’”. [Online]. Available at: <http://www.ddaily.co.kr/news/article.html?no=179338>.
- [31] “ZigBee-sniffing drone maps hackable IoT devices”, *eeNews Europe*, 07-8-2015. [Online]. Available at: <https://www.eenewseurope.com/news/zigbee-sniffing-drone-maps-hackable-iot-devices>.
- [32] “Four types of network attacks businesses should know”, *boannews*, 08-11-2018. [Online]. Available at: <http://www.boannews.com/media/view.asp?idx=74416>.
- [33] “One Step Ahead igloosecurity”. [Online]. Available at: [http://www.igloosec.co.kr/BLOG\\_IoT%20%EB%B3%B4%EC%95%88%EC%9C%84%ED%98%91%EC%97%90%20%EB%94%B0%EB%A5%B8%20%EB%8C%80%EC%9D%91%EB%B0%A9%EC%95%88?searchItem=&searchWord=&bbsCatId=17&gotoPage=2](http://www.igloosec.co.kr/BLOG_IoT%20%EB%B3%B4%EC%95%88%EC%9C%84%ED%98%91%EC%97%90%20%EB%94%B0%EB%A5%B8%20%EB%8C%80%EC%9D%91%EB%B0%A9%EC%95%88?searchItem=&searchWord=&bbsCatId=17&gotoPage=2).
- [34] M. B. in S. on October 15, 2019 and 11:33 Am Pst, “Kaspersky honeypots find 105 million attacks on IoT devices in first half of 2019”, *TechRepublic*. [Online].

- Available at: <https://www.techrepublic.com/article/kaspersky-honeypots-find-105-million-attacks-on-iot-devices-in-first-half-of-2019/>.
- [35] "IoT Security Guidance - OWASP". [Online]. Available at: [https://www.owasp.org/index.php/IoT\\_Security\\_Guidance](https://www.owasp.org/index.php/IoT_Security_Guidance).
- [36] Young-Gap Kim, "Internet of Things Security Requirements", Sejong University, Information and Communication Technology Promotion Center, 2017.
- [37] "Essential component of IoT 'IoT Platform'", Techworld, 17-4-2019. [Online]. Available at: <http://www.epnc.co.kr/news/articleView.html?idxno=83017>.
- [38] "Press Release | Green Zone Security". [Online]. Available at: [http://greenzonesecu.com/bbs/board.php?bo\\_table=press\\_ko&wr\\_id=60](http://greenzonesecu.com/bbs/board.php?bo_table=press_ko&wr_id=60).
- [39] "SKT NUGU". [Online]. Available at: <https://www.nugu.co.kr>.
- [40] "Giga Genie | Global No.1 KT". [Online]. Available at: <https://gigagenie.kt.com/main.do>.
- [41] "Naver Clova". [Online]. Available at: <https://clova.ai/ko>.
- [42] "Google Home, South Korea's official landing with six features | IT Dong-A". [Online]. Available at: <https://it.donga.com/28149/>.
- [43] "Google Home Mini", *Google Store*. [Online]. Available at: [https://store.google.com/kr/product/google\\_home\\_mini](https://store.google.com/kr/product/google_home_mini).
- [44] "[Ubuntu/Linux] everything of ssh public key", I'm a developer, 19-7-2015. [Online]. Available at: <https://storycompiler.tistory.com/112>.
- [45] "what is a FTP?.", Naver | Foundation of the assembly of the computer. [Online]. Available at: <https://blog.naver.com/hdj20/40155944026>.
- [46] FTP concept, SFTP, passive, active mode", *D.O's IT*, 15-8-2017. [Online]. Available at: <https://dany-it.tistory.com/54>.
- [47] "DDoS attack types#1", Zippan, 15-6-2018. [Online]. Available at: <https://jihwan4862.tistory.com/122>.
- [48] "DDoS attack types#2", Zippan, 18-6-2018, 18-6-2018. [Online]. Available at: <https://jihwan4862.tistory.com/123>.
- [49] "<<DoS/DDoS>> RUDY & Slowloris attack types", Naver | High meaning!! Dreaming of Network and information Security Experts. [Online]. Available at: [https://blog.naver.com/p\\_rain/220687298632](https://blog.naver.com/p_rain/220687298632).

- [50] “UDP Flood DDoS Attack”, *Cloudflare*. [Online]. Available at:  
<https://www.cloudflare.com/learning/ddos/udp-flood-ddos-attack/>.
- [51] “KISA Internet protection &KrcERT”. [Online]. Available at:  
<https://www.boho.or.kr/>.
- [52] “[Information security engineer ]FTP Bounce attack”, *securityissue*, 30-10-2018.  
[Online]. Available at: <https://securityissue.tistory.com/56>.
- [53] T. Greene, “FBI warns of attacks on anonymous FTP servers”, *Network World*, 28-3-2017. [Online]. Available at: <https://www.networkworld.com/article/3185873/fbi-warns-of-attacks-on-anonymous-ftp-servers.html>.
- [54] “WordPress site attack through vulnerable home router···South Korea is a risk”.  
[Online]. Available at:  
[https://www.ahnlab.com/kr/site/securityinfo/secunews/secuNewsView.do?menu\\_dist=1&cmd=print&seq=26340](https://www.ahnlab.com/kr/site/securityinfo/secunews/secuNewsView.do?menu_dist=1&cmd=print&seq=26340).
- [55] “Apache Tomcat, Remote Code Execution Vulnerability Patch”, East Security Pill, 16-4-2019. [Online]. Available at: <https://blog.alyac.co.kr/2256>.
- [56] A. Abdou, D. Barrera and P. C. van Oorschot, “What Lies Beneath? Analyzing Automated SSH Brute-force Attacks”, in *Technology and Practice of Passwords*, Cham, 2016, pp 72-91.

# APPENDIX

## <Appendix 1> Code of IoT Honeypot

```
package main
```

```
import (
```

```
    "bufio"
```

```
    "fmt"
```

```
    "log"
```

```
    "math/rand"
```

```
    "net"
```

```
    "os"
```

```
    "os/exec"
```

```
    "path/filepath"
```

```
    "strconv"
```

```
    "strings"
```

```
    "sync"
```

```
    "time"
```

```
)
```

```
// Quick honeypot test code

//////////////////////////////////// Program run functions //////////////////////////////////////

// ReadLines reads a whole file into memory
func readLines(path string) ([]string, error) {
    file, err := os.Open(path)
    if err != nil {
        return nil, err
    }
    defer file.Close()

    var lines []string
    scanner := bufio.NewScanner(file)
    for scanner.Scan() {
        lines = append(lines, scanner.Text())
    }
    return lines, scanner.Err()
}
```

```

// Read the installed profiles from the profiles directory, and print them

func getProfiles() {

    fmt.Println("Installed profiles:")

    var files []string

    root := "profiles/"

    err := filepath.Walk(root, func(path string, info os.FileInfo, err error) error {

        if filepath.Ext(path) == ".pro" {

            files = append(files, path)

        }

        return nil

    })

    if err != nil {

        panic(err)

    }

    for _, file := range files {

        fmt.Println(file)

    }

}

// Print the user help information.

```

```

func getHelp() {

    fmt.Println("Usage: Ballygul [Profile]")

    fmt.Println("\tProfile: A device profile used to build the honeypot.")

    fmt.Println("")

    fmt.Println("Ballygul is a quick honeypot builder based on profile built from scans.")

    fmt.Println("Add new profiles in the 'Profiles' sub-directory. Make sure they end with  
'GigaGenie.pro'")

    fmt.Println("Call be profile by name without the 'GigaGenie.pro' extension. ex:  
\"GigaGenie\"")

    fmt.Println("")

    getProfiles()

}

```

//////////////////////////////// Networking Functions //////////////////////////////////

```

func tcpFlow(port string) {

    protocol := "tcp"

    fmt.Println("Opening " + protocol + " port " + port)

    l, err := net.Listen(protocol, ":"+port)

    if err != nil {

        log.Println(err)

    }

    defer l.Close()

```

```

for {

    _, err := l.Accept()

    if err != nil {

        log.Println(err)

    }

    // This is where the byte response would go

    //c.Write([]byte("hello"))

}

}

```

```

func udpFlow(port string) {

    protocol := "udp"

    fmt.Println("Opening " + protocol + " port " + port)

    s, err := net.ResolveUDPAddr("udp4", ":"+port)

    if err != nil {

        fmt.Println(err)

        return

    }

```

```

    connection, err := net.ListenUDP("udp4", s)

    if err != nil {

```

```

        fmt.Println(err)

        return
    }

    defer connection.Close()

    buffer := make([]byte, 1024)

    r := rand.New(rand.NewSource(time.Now().Unix()))

    for {

        _, addr, err := connection.ReadFromUDP(buffer)

        data := []byte(strconv.Itoa(r.Int()))

        _, err = connection.WriteToUDP(data, addr)

        if err != nil {

            fmt.Println(err)

            return
        }

    }

}

// Function to run TCP dump automatically

```

```

func runTCPDump(wg *sync.WaitGroup, profile string) {

    defer wg.Done()

    binary, lookErr := exec.LookPath("tcpdump")

    if lookErr != nil {

        panic(lookErr)

    }

    // Use a variable for the name with the profile and date

    t := time.Now()

    args := []string{"tcpdump", "-C2048", "-w" + t.Format(time.RFC3339) + "-" + profile +
"capture.pcap"}

    fmt.Println(args)

    cmd := exec.Command(binary, "-C2048", "-w"+t.Format(time.RFC3339)+"-
"+profile+"capture.pcap")

    cmd.Start()

    //env := os.Environ()

    /*execErr := syscall.Exec(binary, args, env)

    if execErr != nil {

        panic(execErr)

    }*/

}

```

```

func main() {

    fmt.Println("Bally Gul v0.0.2")

    var profile string

    if len(os.Args) == 2 {

        profile = os.Args[1]

        //fmt.Println(profile)

        if _, err := os.Stat("profiles/" + profile + ".pro"); err == nil {

            fmt.Println("Profile " + profile + " found! Loading...")

            // Profile found, continue to the main functions.

        } else {

            //fmt.Println("Profile does not exist!")

            getHelp()

            os.Exit(1)

        }

    } else {

        getHelp()

        os.Exit(1)

    }

    // Get profiles

    lines, err := readLines("profiles/" + profile + ".pro")

    if err != nil {

```

```

        log.Fatalf("readLines: %s", err)
    }

    var wg sync.WaitGroup

    // Get the ports and listen on each port
    for _, line := range lines {

        //fmt.Println(i, line)

        var info = strings.Split(line, ",")

        //fmt.Println("Protocol is " + info[0])

        //fmt.Println("Port is " + info[1])

        if info[0] == "tcp" {

            // Call tcpFlow to set up TCP ports

            wg.Add(1)

            go tcpFlow(info[1])

        } else if info[0] == "udp" {

            // Call udpFlow to set up UDP ports

            wg.Add(1)

            go udpFlow(info[1])

        } else {

            fmt.Println("Found an unknown protocol. Expecting tcp or udp.")

            fmt.Println("Check the profile and try again.")

            os.Exit(1)
        }
    }

```

```
        }  
    }  
  
    wg.Add(1)  
  
    // Run TCPCDump in the background  
    runTCPCDump(&wg, profile)  
  
    wg.Wait()  
  
}
```

## IoT 장치의 악성 타겟팅에 관한 연구

2019.

석사학위논문

박민진

국제학과

지도교수: Joshua I. James

초기의 Internet of Things (IoT)는 Machine to Machine (M2M) 커뮤니케이션처럼 사물과 사물, 사물과 사람 사이의 일반적인 통신을 기반으로 연결하는 개념이었다. 그러나 최근 IoT는 통신과 센서 기능을 기기에 부착하여, 인공지능 (AI)과 머신러닝을 사용해 각종 사물들이 스스로 학습 및 판단하고, 생각할 수 있는 개념이라고 볼 수 있다. IoT의 급진적인 발전이 이루어지는 만큼, IoT의 취약점을 목표로 하는 공격도 다양해지고 있다. 하지만 취약점에 대한 분석은 크게 이루어지고 있지 않고 있다.

IoT가 현실 세계와 디지털 세계를 이어주는 만큼 사용자가 디지털 세계에서 해킹을 받으면 현실 세계에도 영향을 줄 수 있다. 본 논문에서는 가장 많이 사용되는 IoT 장치 중 하나인 AI 스마트 스피커를 대상으로 IoT 장치의 공격을 분석하였다. 연구를 진행하기 위해 공격 분석에 많이 사용되고 있는 허니팟을 목적에 맞게 IoT 허니팟으로 만들었다.

모든 IoT 스마트 장치는 인터넷을 통해 통신을 하기 위해 포트를 가지고 있기 때문에 실제 장치가 사용하는 포트를 확인한 후, 포트를 하나의 프로파일로 만들었다. 그 후, 외부에서 접속할 수 있게

IoT 허니팟에 퍼블릭 아이피를 부여하고, 해당 포트들을 열어준 다음 포트에 들어오는 데이터를 수집해서 분석 하였다.

포트를 통해 들어온 데이터를 분석한 결과, 장치를 목표로 한 공격은 발견되지 않았지만, 다양한 곳에서 포트에 접속하려고 시도한 것을 발견할 수 있었다. 본 논문에서 IoT 허니팟을 사용해 진행한 공격 분석은 장치의 포트만 확인할 수 있으면, 추후 스피커 이외의 다른 IoT 장치에도 사용될 수 있다.

주제어: 사물인터넷, 허니팟, 인공지능 스마트 스피커, IoT 장치 공격, IoT 포렌식

# **A Study on the Malicious targeting of IoT Devices**

2019.

Master's Degree

Park, Min Jin

Department of International Studies

Advisor Prof. Joshua I. James

In the early Internet of Things (IoT), like Machine to Machine (M2M) communication, was a concept of connecting things based on regular communication between things, things, and people. However, IoT is a concept that various objects can learn, judge, and think by using artificial intelligence (AI) and machine learning by attaching communication and sensor functions to devices. As the radical development of IoT is taking place, there are a variety of attacks targeting IoT device vulnerabilities. However, the analysis of vulnerabilities has not been done much.

IoT connects the real world with the digital world. Increasingly, if a user is hacked in the digital world, it can affect the real world. In this paper, we analyzed the attack of IoT devices targeting AI smart speaker, one of the most used IoT devices. To proceed with the research, a honeypot, which is widely used for attack analysis, was created.

We call it an IoT Honeypot. Every IoT smart device has a port for communicating over

the Internet, so after identifying the port that the real device uses, we have made the port as a profile. After that, the public IP address was assigned to the IoT Honeypot for external access, the ports were opened, and the data coming into the ports was collected and analyzed.

Analyzing the data coming in through the ports revealed no attacks targeting the device but found attempts to access the port from various places. Attack analysis using IoT Honeypot in this paper can be used for other IoT devices later if only the port of the device can be identified.

**Keywords:** Internet of Things, Honeypot, AI smart speaker, IoT device attack, IoT

**Forensic**