Analysis of the Impact of ICMP Flood Attack in IoT System

Zhiyong Liu, Kai Zheng, Xin Huang
Computer Science and Software Engineering
Xi'an Jiaotong-Liverpool University
Suzhou, China
Zhiyong.Liu15@student.xjtlu.edu.cn, Kai.Zheng@xjtlu.edu.cn, Xin.Huang@xjtlu.edu.cn

***Abstract*—Internet of Things (IoT) is widely used in various fields. It connects diverse devices and enables them to collect and exchange data. However, with the popularity of IoT, more and more attentions have been paid on the security of IoT. In this paper, experimenters use hping3 to launch a Denial of Service (DoS) attack to IoT. Hping3 is a penetration testing tool in kali Linux. The DoS attack is launched by the method of ICMP Flood. Furthermore, the analysis of impact on the victim is also given.**

Keywords—Internet of Things (IoT); Denial of Service (DoS) attack; ICMP Flood; Network security;

I. INTRODUCTION

The Internet of Things (IoT) paradigm conceives the ubiquitous interconnection and cooperation of devices in the Internet, which is considered as an applied extension of Internet. Because various countries make great efforts in developing IoT, it is regarded as the third wave of information industry following the Internet and mobile communication network [1].

In an IoT environment, every objects are connected based on the Internet. This system has the ability of sensing real-time data, such as temperature and humidity [2]. It collects various information about the environment and communicates these data with other things. The data collected can be efficient for improving the monitoring and control of various systems. Due to the advantage of interoperability and intelligence [3], IoT is used in a lot of domains, such as smart home, healthcare monitoring, transportation and so on.

However, IoT system is usually configured in wireless environment, so data transmitted is likely to cover all the terminals within the scope of WLAN, which provides the hackers more chance to attack [4]. Among all possible attacks, Denial of Service (DoS) attack is an effective and common form [5]. It is launched by a malignant intruder and aims to exhaust the resource of target. After the attack, victim is unable to provide normal service [6].

In this paper, A IoT environment is deployed. A DoS attack in IoT is designed and implemented. The DoS attack is launched by the method of ICMP Flood attack, while the results of experiment are beneficial to analyze the impact of DoS attack on victim.

The main contributions of this paper are listed as following:

- An ICMP Flood attack is launched in IoT system with the help of hping3 in Kali Linux.
- The CPU utilization, the time for attacking, the success rate of ICMP Flood attack on victim is shown.
- This paper provides a practical analysis of the ICMP Flood attack and the factors which influences the performance of ICMP Flood attack are shown.

This paper is organized as follow. Section II introduces the related work of this paper. Section III introduces the background knowledge of Internet Control Message Protocol (ICMP) and ICMP Flood attack. Section IV shows the experiment platform. Section V introduces the process of testing and Section VI is the conclusion part.

II. RELATED WORK

Saravanan Kumarasamy et al., [5] simulated the detection mechanism for SYN flooding attacks classified into three categories.

In [7]-[10], it introduces a prototype security framework for the internet of Things.

The authors in [11] identifies the harmless rete at which the ICMP traffic can be generated and resounded over the Internet.

III. BACKGROUND

In this section, we give the background knowledge of ICMP and the principle of ICMP Flood attack.

A. Definition of Internet Control Message Protocol

Internet Control Message Protocol (ICMP) is a sub-protocol of Internet Protocol Suit, it is used to transfer control message between IP hosts and routers. These control messages can be various.

ICMP is in the Internet layer of Internet Protocol Suit, it is not aimed to transfer user data, but it plays an important role in the process of data transfer [12]. As an error detection and reward mechanism, it is aimed to test the network connection and ensure the accuracy of attachment. The main functions are listed below.

- To detect whether remote host exists
- Establish and maintain the routing information
- ICMP Redirect
- Data Flow Control (ICMP use different types and codes to make host recognize different states of connection).

The most common example of ICMP is ping. It uses the ICMP protocol packet to detect whether the remote host can be reached. First, Ping sends a ECHO_REQUEST ICMP message to the target host and waits for the respond. When ECHO_REQUEST ICMP message reaches the target host, the target host will respond a ECHO_REPLY message to the sender. If sender receives this ECHO_REPLY message in valid time, it denotes that the connection is well between sender and target host.

B. ICMP Datagram Structure

8 bit	8 bit	16 bit
Type	Code	Checksum
Other message specific information		
Data section (variable size)		

Fig. 1. ICMP Message encapsulation

An ICMP packet has an 8-byte header and a data section of variable size. In this 8-byte header, the first 4 bytes have fixed format. The first byte is the type and the second byte is the code. These two bytes determine the kind of ICMP packet. The following two bytes denote the checksum, which is calculated by the header and data. The last 4 bytes are determined by the type and code of ICMP header [13].

ICMP messages can be divided into two groups [14]. One is error message, the other is query message. The followings are five common ICMP messages.

- **Echo:** When host uses the ping, it sends an ECHO_REQUEST ICMP message to the target. After receiving the request packet, target return an ECHO_REPLY ICMP message to the sender.
- **Destination unreachable:** When the router is not able to find the routing or the host for the datagram, it discards this datagram and send a ICMP destination unreachable message to the sender of this datagram.
- **Source quench:** If a host transmits the data to the destination fast while the destination is unable to process the data in the matching speed, the destination can send a ICMP source quench to remind the host decrease the speed of data transmitting.
- **Time Exceed:** When time to live (TTL) of IP packet decreases to 0, it is regarded as time exceeded. Then the router will send a ICMP time exceeded message to the sender to inform there occurs a time exceeded error.
- **Redirect:** Router sends the ICMP message to inform the host to modify the routing table.

C. ICMP Flood Attack

As we have already explained, an ICMP requests the target to process the ICMP request and respond to the server. In this procedure, it takes the resources of target, such as CPU. Therefore, for causing the target to operate abnormally, attacker can launch the ICMP Flood attack to exhaust the resources of victim [15].

Without waiting for replies, attacker sends the ICMP message as fast as possible to inundate the victim. Because the attacker needs to process the ICMP messages and responds to the sender, some resources of the victim are occupied to deal with these tasks. With the increasing speed of ICMP message transmitting, the target uses up all the resources and denies other system services.

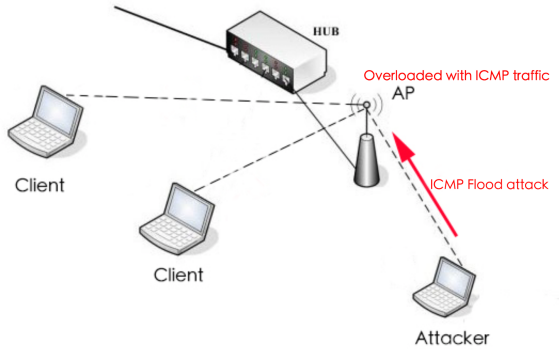

Fig. 2. ICMP Flood Attack

If the attacker has more superior bandwidth than the victim, or the target system is slow enough, the ICMP flood attack will be more successful. According to this, generally when attacker launches a ICMP Flood attack, a lot of attack hosts are used to increase the amount of the ICMP messages transmitting to the victim, which intensifies the effect of ICMP Flood attack on target machine [16].

In a wireless local area network environment, ICMP flood can be easily launched by the attacker. Especially in public WLAN, attacker can type the command “arp –a” in the terminal or using special tools to get the IP of other hosts online. After determining the target host, attacker use software to launch the ICMP Flood attack. The continuous attacks can be a reason for slow network and repressed traffic speed for legal machines.

IV. THE EXPERIMENT PLATFORM

A. The Framework of Experiment Platform

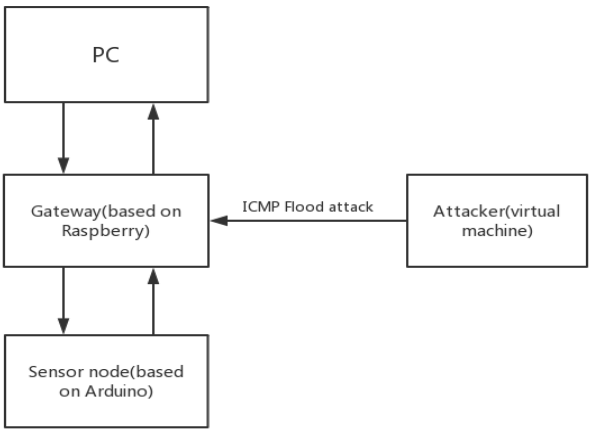

Fig. 3. The framework of IoT system

The platform is composed by following components:

- **Gateway:** It is based on the Raspberry, which connects other devices. The raspberry is pre-installed with the Dstat, which is a monitoring tool in Linux.
- **PC:** The PC is client which connects to wireless LAN. It can exchange data with sensor node through gateway.
- **Attacker:** The attacker is Kali Linux which is a virtual machine on the PC. It can use built-in tools to launch the ICMP Flood ICMP attack.
- **Sensor node:** It is based on Arduino, which can monitor the information of environment.

B. *The Experiment Devices in Lab*

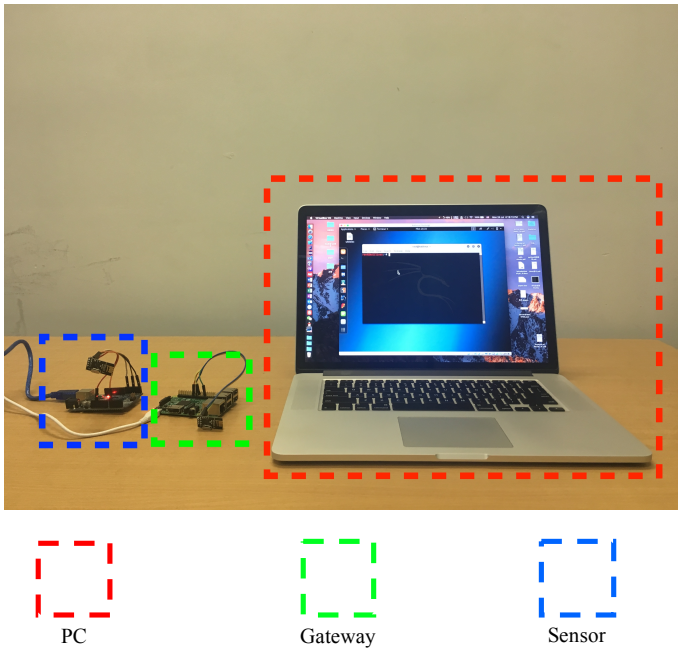

Fig. 4. The experiment devices in lab

The red box denotes the PC. The green box is the Kali Linux virtual machine on PC and the blue box stands for the AP based on Raspberry.

C. *Dstat*

The Dstat is a convenient monitoring tool in Linux. With this tool, the status information of Raspberry can be monitored through the terminal in real time. Dstat is a frequently used in performance test. Compared with direct commands in terminal, its preponderance is that it gathers different data in one window, which make user can monitor the status information more intuitively. The data are various, such as CPU utilization, memory footprint, disk information and so on.

D. *Kali Linux*

The attacker in this experiment is Kali Linux installed in virtual machine on PC. Kali Linux is a Debian-derived Linux distribution which is used to implements penetration testing and digital forensics. It consists of various penetration-testing programs including Nmap, Mdk3, Aircrack-ng and so on.

In this experiment, the programs used in kali Linux are listed following:

- **Hping3:** Hping3 is packet generator designed for TCP/IP protocol.
- **Wireshark:** Wireshark is a packet analyzer, it is designed for obtaining the data packets including HTTP, TCP, UDP and so on.

V. TESTING

In this experiment, the PC and the sensor node are connected through gateway which is based on Raspberry. PC and Sensor can exchange data through gateway. Attacker based on virtual machine launches the ICMP Flood attack to gateway to see the impact on gateway.

A. *Initial Status of Gateway*

Due to the Dstat tool pre-installed in raspberry, we can obtain the CPU utility of the gateway which is not attacked. Besides, we use the ping command to check the connection between PC and gateway. The result of the experiment is shown Table I.

TABLE I. RESULTS FOR INITIAL STATUS

Status	CPU utility	Respond time(Average)	Packet lose rate
Initial status	4%	6.774	0%

If the gateway is not attacked, the CPU utility is extremely low. We use PC to send 100 packets to gateway and the packet lose rate is 0%.

Among these 100 packets, we choose 10 packets and calculate the average respond time, which is very short. The respond time for these 10 packets are shown in Figure 5.

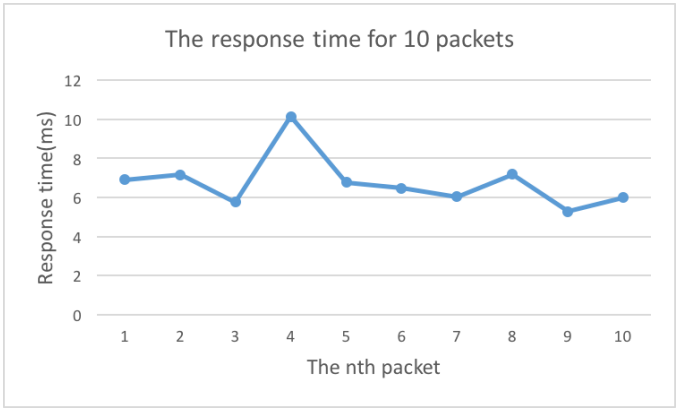

Fig. 5. The response time of the gateway

Finding. Figure 5 shows that the performance of gateway is normal although the respond time is fluctuating. The respond time of the gateway is mainly within 10ms, which is very short.

B. *ICMP Flood Attack on Gateway*

We use hping3 tool to launch the ICMP Flood attack, the parameters of hping3 we set are listed following:

- **Target IP** – Specifies the target IP which will be attacked. In our experiment, the IP is 192.168.12.1, which is the IP of gateway.
- **-1** – Specifies the protocol mode. "-1" denotes the ICMP mode. It means that only ICMP packets are sent.
- **-d** – Specifies the size of packet. The ICMP header is tagged on this packet before being encapsulated in IP datagram.
- **--flood** – It means sending the packets as fast as possible without showing incoming replies.

In this experiment, the size of ICMP packets launched by attacker is variable. We first chose three different sizes which are 250 bytes, 500 bytes and 750 bytes. While attacking, we use ping command to verify if the gateway can operate normally. The CPU utility, the time of success of attack (the time when first packet loses) and the packet lose rate are shown in TABLE II. The packet lose rate is calculated based on 100 packets.

TABLE II. RESULTS FOR ICMP FLOOD ATTACK IN DIFFENENT SIZES

Size of packets	CPU utility	Time of success of attack	Packet lose rate
250 bytes	28%	8.32s	56.9%
500 bytes	41%	4.34s	67.6%
750 bytes	59%	1.79s	80.4%

Finding. Table II shows that when the size of packet increases, the CPU utility is improved. The packet lose rate is proportional to the size of packet. Besides, the time of success of attack in all three scenarios are short. It falls from 8.23s to 1.79s with the increase of size, which means it is easier to attack successfully with the increase of size.

We choose 10 packets in each scenario, the respond time of these packets is shown in Figure 6.

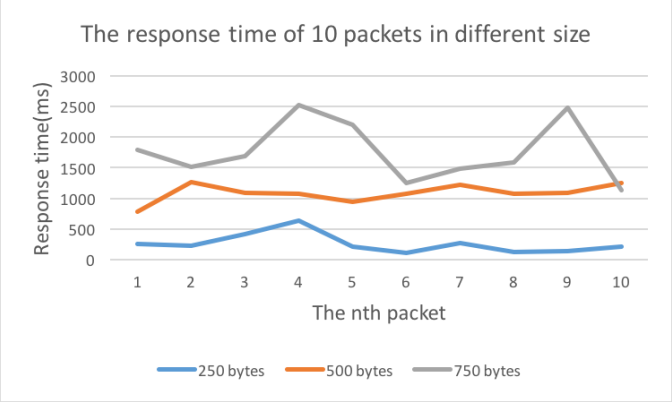

Fig. 6. The response time of 10 packets in different size

Finding. Figure 6 shows that larger size of packet causes longer response time. The increasing of response time denotes the better attack performance. It can be concluded that the performance of ICMP Flood attack will be more effective with the increase of size.

We use other extra size of ICMP packets to launch the attack in order to see the change of packet lose rate. The packet size is 1000 bytes, 1250 bytes, 1500 bytes. The packet lose rate of all different sizes is shown in Figure 7.

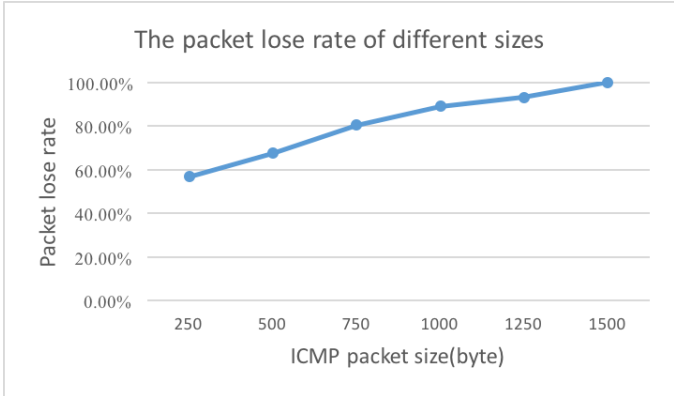

Fig. 7. The packet lose rate of different sizes

Finding. The packet varies from 250 bytes to 1500 bytes and the packet lose rate rises from 56.9% to 100% which means the performance of ICMP Flood attack becomes better when the size of packet increases.

VI. CONCLUSION

In this paper, experimenters deploy a IoT system as the victim of ICMP Flood attack. The gateway in the IoT system is attacked with the help of hping3 in Kali Linux on virtual machine. The results of experiment show that the increase of size of ICMP packet size can improve the performance of attack, which reflects in CPU utility, time of success of attack and packet lose rate in experiment

In the future, more types of ICMP attack will be studied and the performance will be evaluated.

ACKNOWLEDGMENT

This work has been supported by the XJTLU research development fund projects RDF140243 and RDF150246, as well as by the Suzhou Science and Technology Development Plan under grant SYG201516, and Jiangsu Province National Science Foundation under grant BK20150376.

This work was supported in part by the Natural Science Foundation of China under Grant No. 61401517, in part by the National High Technology Research and Development Program ("863" Program) of China under Grant No. 2015AA016001.

REFERENCES

[1] M. Elkhodr, S. Shahrestani, and H. Cheung, "The Internet of Things: New Interoperability, Management and Security Challenges," arXiv.org, vol. cs.NI. 17-Apr-2016.

[2] D. Makoshenko and I. Enkovich, "IoT development - Discovering, enabling and validation of real life IoT scenarios.," FMEC, pp. 159–164, 2017.

[3] T. Ochs and U. Riemann, “Internet of Things - The Power of the IoT Platform.,” IoTBDS, pp. 284–294, 2017.

[4] D. Minoli, K. Sohraby, and J. Kouns, “IoT security (IoTSec) considerations, requirements, and architectures.,” CCNC, pp. 1006–1007, 2017.

[5] S. Kumarasamy and A. Gowrishankar, “An Active Defense Mechanism for TCP SYN flooding attacks,” arXiv, vol. cs.CR, 2012.

[6] B. T. Wang and H. Schulzrinne, “Analysis of denial-of-service attacks on denial-of-service defensive measures,” presented at the GLOBECOM '03. IEEE Global Telecommunications Conference, pp. 1339–1343.

[7] X. Huang, P. Craig, H. Lin, and Z. Yan, “SecIoT - a security framework for the Internet of Things.,” Security and Communication Networks, vol. 9, no. 16, pp. 3083–3094, 2016.

[8] W. Bo, Y. Zhang, X. Hong, H. Sun, and X. Huang, “Usable Security Mechanisms in Smart Building,” presented at the 2014 IEEE 17th International Conference on Computational Science and Engineering (CSE), 2014, pp. 748–753.

[9] K. Zhao and L. Ge, “A Survey on the Internet of Things Security.,” CIS, pp. 663–667, 2013.

[10] N. Xue, X. Huang, and J. Zhang, “S2Net - A Security Framework for Software Defined Intelligent Building Networks.,” Trustcom/BigDataSE/ISPA, 2016.

[11] J. Udhayan and R. Anitha, “Demystifying and Rate Limiting ICMP hosted DoS/DDoS Flooding Attacks with Attack Productivity Analysis,” presented at the 2009 IEEE International Advance Computing Conference (IACC 2009), 2009, pp. 558–564.

[12] A. H. M. Taib, W. N. A. W. Ali, and N. S. Shaari, “ICMPV6 Vulnerability - The Importance of Threat Model and SF-ICMP6.,” IJMCMC, vol. 5, no. 2, pp. 78–100, 2013.

[13] W. Li, D. Zhang, J. Yang, and G. Xie, “On evaluating the differences of TCP and ICMP in network measurement.,” Computer Communications, vol. 30, no. 2, pp. 428–439, 2007.

[14] M. Yamana, K. Hirata, H. Shimizu, H. Nakatani, Toshifumi Kai, and K. Tsukamoto, “Simulation of IP Traceback for the Denial of Service Attack,” presented at the 2005 Symposium on Applications and the Internet Workshops (SAINT 2005 Workshops, 2005, pp. 110–113.

[15] M. Bogdanoski and A. Risteski, “Wireless Network Behavior under ICMP Ping Flood DoS Attack and Mitigation Techniques.,” IJCNIS, 2011.

[16] X. Yang, T. Ma, and Y. Shi, “Typical DoS/DDoS Threats under IPv6,” presented at the 2007 International Multi-Conference on Computing in the Global Information Technology (ICCGI'07), 2007, pp. 55–55.