Netflow Based HTTP Get Flooding Attack Analysis

Jungtae Kim, Jong-Hyun Kim and Ikkyun Kim
Information Security Research Division
Electronics & Telecommunications Research Institute
Daejeon, Republic of Korea
e-mail: {jungtae_kim, jhk, ikkim21}@etri.re.kr

Koohong Kang
[2]Dept. of Information and Communications Engineering
Seowon University
Cheongju, Republic of Korea
e-mail: khkang@seowon.ac.kr

Abstract— The paper proposes a security analysis method using the netflow information to analyze the HyperText Transfer Protocol (HTTP) get flooding attacks. As it is hard to distinguish from the normal Web accesses and further severely disturb the normal Web user accesses, the attack is considered as one of the most effective Distributed Denial-of-Service (DDoS) attacks. In this paper, we propose an analysis method of the HTTP Get flooding attacks based on the netflow information. In particular, the byte over packet per flow ratio helps to achieve the attack detection without the individual packet processing overheads.

Keywords-HTTP Get Flooding; Netflow; DDoS Attack; **.**

I. INTRODUCTION

Netflow is a feature that was introduced on Cisco routers that provides the ability to collect IP network traffic as it enters or exits an interface [1]. The major advantage of utilizing the flow data is that it helps to analyze the network traffic usage and further enables network security enhancement. The flow is generally defined by the 7 unique key fields including the following information: source and destination IP address, source and destination port, layer 3 protocol type, type of service byte**,** and the input logical interface [2]. In order to utilize the netflow information to analyze the DDOS traffic, a system requires to have the following components: flow exporter, which processes packets to produce flow data, the preconfigured flow collectors and storages. Consequently, the flow collectors store and index the collected flows for search purposes. Later, an analysis application then analyzes the stored flow data for the network traffic or security analysis purposes. Based on the above systems components with the netflow information, we propose a network anomaly detection method based on the detailed analysis on the HTTP Get Flooding Attacks.

II. LITERATURE REVIEW

The HTTP Get flooding attacks are being exploited in the most efficient way among Denial-of-Service (DoS) type attacks aimed at the Web server application layer [3]. The attack is specially designed to send a large volume of the HTTP-Get requests to the targeted Web applications and servers. The attacks are initiated by virus infected zombie PCs under the control of Command and Control (C&C) server. Consequently, the victim's Web server is unable to reply to the normal user requests due to the processing overheads. Since these attack packets contain the normal HTTP requests, Web servers cannot easily distinguish between normal user's HTTP-Get request messages and the malicious requests [4]. The advantage of the approach presented in this paper is that the netflow helps the network administrator to identify network anomalies by monitoring the detailed traffic flows information rather than the conventional network security devices including the firewall, Intrusion Detection System (IDS) or Intrusion Prevention System (IPS), which obviously involve an extra cost of deploying the devices, as well as a change of network settings to capture the traffic information for signature based analysis. Although the default setting for the netflow information exports is set depending on the switch or router manufacturers, such as the inactive timer set at 15 sec and the active timer set at 1800 sec, the flow analysis is helpful in case of the HTTP Get Flooding attack, which has a unique characteristic with the repeated short TCP 3-Way Handshake periods. The paper introduces an experimental setting with system and network configurations in Section III. Details of the attack analysis technique using netflow information with the analysis results are described in Section IV. Finally, Section V concludes the paper with future works.

III. EXPERIMENTAL SETTINGS

Figure 1 shows 2 minute attacks to conduct the HTTP Get flooding attacks. The Command and Control server, with the Netbot Attacker [5], was installed in a separate external network from the attack target network. The target network was configured with 3 zombie PCs with 2 Web servers hyperlinked unidirectional.

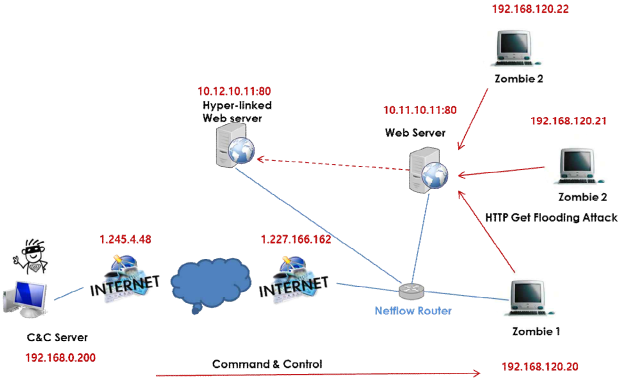

Figure 1. Network Configuration of the HTTP Get flooding attack using Netbot Attacker.

ISBN: 978-1-61208-582-1

IV. Attack Analysis Technique using Netflow

Although conventional HTTP Get flooding attack detection adopts a method that specifically analyzes the contents of the packet, especially installed and operated in the input of particular Website or Web server [4], our approach proposes an analysis method of the HTTP Get flooding attacks based on the netflow information rather than the detailed network traffic statistics. Figures 2-4 show the flow information for the attack caused by using the Netbot Attacker by flow duration, number of packet, and byte size, respectively. The attack flow is generated for 2 minutes (Figure 2), and the number of packets (Figure 3) within the flow is fixed in its size. The flow analysis experiment was conducted considering a 2 minutes attack from 3 zombie PCs with a break. The total number of flows measured were approximately 22,985 at each zombie PC which includes the recursive HTTP Get request and reply messages with the TCP 3-way Handshake packets (48 Byte SYN & SYN ACK and 40 byte ACK, FIN & RST ACK).

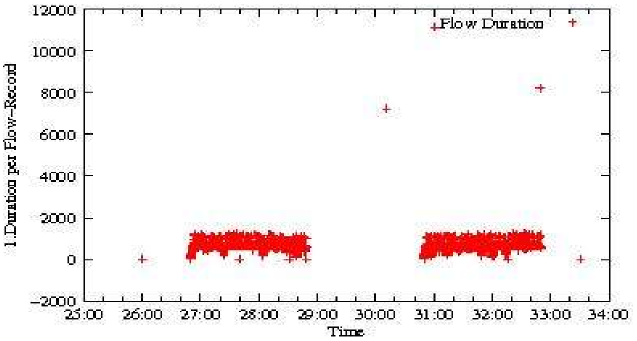

Figure 2. Flow Duration for HTTP Get flooding attack patterns.

The results, as depicted in Figure 2, show that most of the attack related flow duration fell into within the 2000 ms (2 secs) boundary with short TCP sessions. Figures 3 and 4 show the total number of packets and bytes per flow record of the attack. The machine generated attack by zombie PCs was fixed in its packet size of 6 with 285 byte size and additional 5 & 7 packets with 245 and 333 byte size, respectively, due to the reset (RST) packet. Figure 5 provides a Byte over Packet Ratio (BPR) per flow record for the HTTP Get flooding attack and the results are listed in Table I. The proposed analysis result shows that, the BPR is 47~49 for the HTTP Get flooding attack.

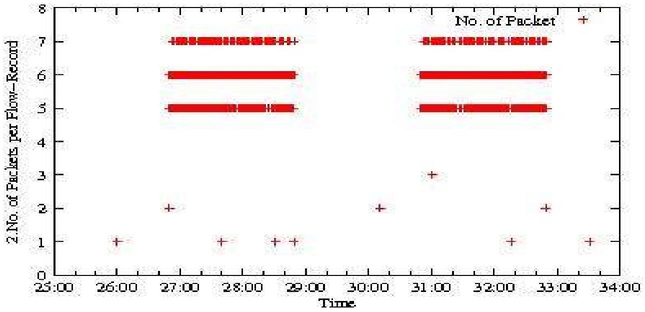

Figure 3. Number of packets per flow for HTTP Get flooding attack.

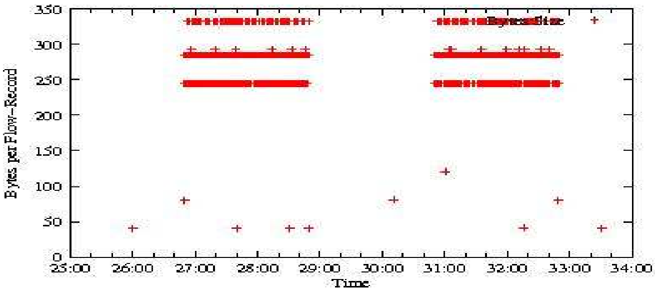

Figure 4. Byte size per flow for HTTP Get flooding attack patterns.

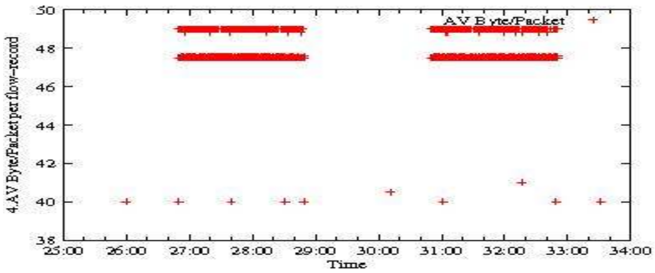

Figure 5. Byte / Packet per flow for HTTP Get flooding attack.

TABLE I. HTTP GET FLOODING ATTACK PATTERNS.

No. PACKET	*BYTE SIZE*	*BYTE/PACKET*
5	245	49.00000
6	285	47.50000
7	333	47.57143

V. Conclusion and Future Work

We propose an analysis method of the HTTP Get flooding attacks based on the netflow information rather than the detailed network traffic statistics, such as the packer per second (pps) and total byte size. In particular, machine generated attack patterns show that a specific BPR can be applied to detect the DDoS attack.

Acknowledgment

This work was supported by Institute for Information & communications Technology Promotion(IITP) grant funded by the Korea government(MSIP) (No.R-20160222-002755, Cloud based Security Intelligence Technology Development for the Customized Security Service Provisioning).

References

[1] Cisco IOS NetFlow, Cisco Systems, Inc. https://www.cisco.com/c/en/us/products/ios-nx-os-software/ios-netflow/index.html [accessed July 2017]

[2] NetFlow Export Datagram Format, Cisco Systems, Inc. http://www.cisco.com/c/en/us/td/docs/net_mgmt/netflow_collection_engine/3-6/user/guide/format.html [accessed July 2017]

[3] Y. Choi and et. al, "AIGG Threshold Based HTTP GET Flooding Attack Detection," in Proc. of WISA 2012, pp 270-284, 2012. [accessed July 2017]

[4] Y. Kim and et. al, "HTTP Get Flooding Detection Technique based on Netflow Information," in Proc. Of Internet 2016, pp 26, 2016. [accessed July 2017]

[5] Netbot Attacker VIP 4.7 Version. http://cfs13.tistory.com/image/5/tistory/2009/02/19/18/02/499d203101657 [accessed July 2017]