In-Network SYN Flooding DDoS Attack Detection Utilizing P4 Switches

Pegah Golchin, Leonard Anderweit, Julian Zobel, Ralf Kundel, Ralf Steinmetz
Multimedia Communications Lab (KOM)
Technical University of Darmstadt, Germany
Contact: pegah.golchin@kom.tu-darmstadt.de

***Abstract*—With the rapid development of Internet applications, the demand for reliable online services similarly increases. However, *Distributed Denial-of-Service* (DDoS) attacks disrupt the accessibility and the availability of online services. Therefore, DDoS detection and mitigation are crucial tasks to achieve high service availability. In this paper, we propose a novel in-network detection scheme for SYN flooding, the most prevalent type of DDoS attacks. By relocating the attack detection from a centralized controller to programmable P4 switches, the detection time is reduced, and the workload is distributed in the network. Extending passive classification methods, we propose an active detection mechanism, identifying SYN flooding DDoS attacks by selective packet dropping. By this, we expect more accurate detections compared to the state-of-the-art under congested network conditions.**

***Index Terms*—SYN flooding attack, DDoS attack, SDN, P4**

I. Introduction

Within recent years, *Distributed Denial-of-Service* (DDoS) attacks have increased in number, duration, and extent [1]. DDoS attacks are cyber attacks that attempt to overwhelm networking and computing resources of a victim by targeting it from a multitude of different locations. In general, DDoS attacks are categorized in volumetric-based, protocol-based, and application-based attacks [2]. We focus on the protocol-based *SYN Flooding* attacks in this work, as they constitute around 80% of all DDoS attacks [3]. SYN flooding exploits the three-way handshake process for TCP connection establishment between client and server, pretending demand for new TCP connections by sending large numbers of spoofed synchronization packets (SYN) to the server. This consumes a disproportional large amount of network and computing resources of the target server, which is sending synchronization acknowledgments (SYN-ACK) in turn and waiting for an acknowledgment packet (ACK) to open the TCP connection. The server's backlog queue eventually overflows with half-open TCP connections, denying any further processing of TCP connection requests at the server and, thus, access to the provided service.

Softwarized network architectures, i.e., *Software-Defined Networks* (SDN), provide several strengths such as simplification, flexibility, and improved network management capabilities by decoupling control plane and data plane. Thus, the centralized controller has a global view of the network leading the centralized network management. With available computational resources in the controller, it is possible to develop complex algorithms for different management tasks such as network security, congestion control, and load balancing [4]. Especially for network security, the fast detection and mitigation of anomalous behavior like DDoS attacks is a crucial task. Nevertheless, this requires flows to be routed over a controller with a DDoS attack detector. Besides requiring significant computational resources for DDoS attack detection algorithms in general [5], classifying all flows in a controller will exceed the available processing power, especially for flooding-based attacks. However, processing power directly in the data plane has improved greatly with high-performance programmable hardware switches based on P4 [6] and FPGAs [7]. This enables DDoS attack detection without re-directing flows over a central controller, improving speed and workload balance for in-network flooding-based DDoS attack detection.

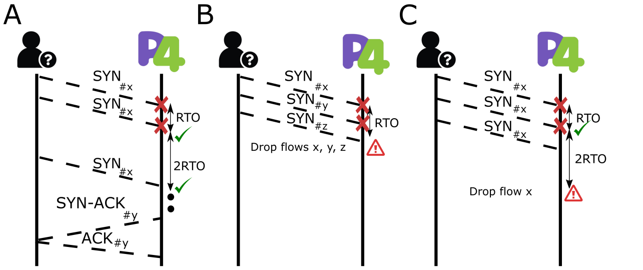

Fig. 1. If the P4 switch drops a subset of SYN packets, a normal user sends corresponding retransmission packets with the same sequence number in the retransmission time, which increase exponentially (A), while an attacker sends SYN packets with random sequence numbers and spoofed IP addresses within the constant time (B). Even if the attacker sends the duplicate packets, the second retransmitted packet will not arrive in the time range of (RTO, 2RTO] as the retransmission time is constant (C).

In this work, we propose a novel approach for SYN flooding DDoS attack detection by leveraging the retransmission mechanism of SYN packets in the three-way handshake of TCP connection establishment. As demonstrated in Figure 1, dropping a subset of SYN packets allows to differentiate flows from normal and malicious users. We apply this approach on P4-programmable switches to avoid flow duplication to the controller. In addition, we expect that applying this idea improves the *detection accuracy* and *detection speed*, which are the main challenges of SYN flooding DDoS attack detection.

II. Proposed Idea

We propose to implement rapid in-network SYN flooding attack detection in P4 switches to avoid flow forwarding time to a central SDN controller. It is separated into two modules: *Attack Threshold Assumption* and *SYN Packet Retransmission Checking*.

A. Attack Threshold Assumption

In order to make a balance between the SYN flooding attack detection accuracy and the connection establishment latency for the legitimate users, an entropy-based threshold is considered similar to the work of Siris *et al.* [8]. The entropy threshold is calculated based on the number of usual and unusual handshake sequences. The unusual handshake sequences are handshakes in which the server has not received any response from the client or has not received a corresponding ACK packet. While entropy is greater than the threshold under the normal traffic, the increase of unusual handshake sequences during SYN flooding attacks decrease the entropy below the threshold [8]. However, this method is not able to correctly distinguish attacks under network congestion, as the number of falsely labeled unusual handshake sequences increase with dropped packets by congestion avoidance. Therefore, it will incorrectly detect SYN flooding attacks in a congested network, which reduces the detection accuracy. To overcome this issue, we consider this method only as a potential indicator for a possible DDoS attack, which requires further investigation.

B. SYN Packet Retransmission Checking

After unusual handshake sequences indicate a possible DDoS attack, we apply a checking mechanism for SYN packet retransmission behavior. For that, the P4 switch drops a subset of SYN packets and its first retransmission in each suspicious flow. During this, the flow information is logged, i.e., the flow ID, sequence number, timestamp, IP addresses, and TCP ports. The respective flow is monitored for two expected *SYN packet retransmissions* in a time period to distinguish between a legitimate and a malicious user. The retransmission time should be considered to range between the packet Round Trip Time (RTT) and the recommended first *Retransmission Timeout* (RTO), typically 3 seconds. The second retransmission time is expected in the range $(RTO, 2 \times RTO]$ [9]. Typically, the retransmission time of the two continuous retransmitted SYN packets should exponentially increase while it is constant for the SYN flooding attack. Therefore, as depicted in Figure 1 (A), the first legitimate retransmitted SYN packet arrives with the same sequence number and source IP address as the first dropped SYN packet within the corresponding retransmission time. After this second packet is dropped again by the switch, a third packet with the same information is expected within the time range of $(RTO, 2 \times RTO]$. In contrast, for a malicious user, either no other SYN packet is received at all or its information probably won't match that of the dropped packet, as SYN flooding attack tools generate SYN packets with random sequence numbers and random source IP addresses. As illustrated in Figure 1 (C), if the attacker sends duplicate SYN packets, the retransmission time between the first and the second packets remains constant and the attack can be detected.

Especially for congested network situations, we expect to detect SYN flooding attacks more accurately than, e.g., a standalone entropy-based detection. Consequently, we expect to reduce the rate of successful SYN flooding attacks as the presented extension allows the direct in-network assessment of an actual attack on a P4 switch. Thus, SYN flooding attacks are directly mitigated by the P4 switches filtering the respective flows.

III. Conclusion

In order to avoid disturbances in the availability and accessibility of online services by DDoS attacks, there is a need to detect and mitigate malicious flows. SDN is a softwarized network architecture improving network flexibility by disaggregating the data and control plane. In this work, we propose a SYN flooding DDoS attack detection scheme that can be realized within P4-programmable switches in the data plane. We expect a faster detection by implementing this method in the data plane than implementing a centralized controller. In addition, we expect a reduction in false positive SYN attack detection for various network behavior, resulting in a more balanced detection accuracy and detection speed. Several issues need to be addressed in future work. For instance, the attack assumption threshold should be investigated further to achieve a higher SYN attack detection accuracy.

Acknowledgment

This work is funded by the Federal Ministry of Education and Research of Germany (BMBF) within the CELTIC-NEXT Flagship Project AI-NET-PROTECT and in parts by the German Research Foundation (DFG) within the Collaborative Research Center (CRC) 1053 MAKI and the LOEWE initiative within the project *Nature 4.0 – Sensing Biodiversity* and the *emergenCITY* center.

References

[1] N. Z. Bawany, J. A. Shamsi, and K. Salah, "DDoS Attack Detection and Mitigation using SDN: Methods, Practices, and Solutions," *Arabian Journal for Science and Engineering*, vol. 42, no. 2, 2017.

[2] Z.-Y. Shen, M.-W. Su, Y.-Z. Cai, and M.-H. Tasi, "Mitigating SYN Flooding and UDP Flooding in P4-based SDN," in *Proc. of the 22nd Asia-Pacific Network Operations and Management Symposium (APNOMS)*. IEEE, 2021.

[3] Kaspersky, "DDoS attacks in Q3 2019," (Accessed on 04.03.2022). [Online]. Available: https://securelist.com/ddos-report-q3-2019/94958/

[4] W. Xia, Y. Wen, C. H. Foh, D. Niyato, and H. Xie, "A Survey on Software-Defined Networking," *IEEE Communications Surveys & Tutorials*, vol. 17, no. 1, 2014.

[5] Q. Niyaz, W. Sun, and A. Y. Javaid, "A Deep Learning-based DDoS Detection System in Software-Defined Networking (SDN)," *arXiv preprint arXiv:1611.07400*, 2016.

[6] P. Bosshart, G. Gibb, H.-S. Kim, G. Varghese, N. McKeown, M. Izzard, F. Mujica, and M. Horowitz, "Forwarding Metamorphosis: Fast Programmable Match-action Processing in Hardware for SDN," *ACM SIGCOMM Computer Communication Review*, vol. 43, no. 4, 2013.

[7] R. Kundel, K. Eryigit, J. Markussen, C. Griwodz, O. Abboud, R. Hark, and R. Steinmetz, "Host Bypassing: Direct Data Piping from the Network to the Hardware Accelerator," in *14th IEEE International Symposium on Embedded Multicore/Many-core Systems-on-Chip (MCSoC)*, 2021.

[8] V. A. Siris and F. Papagalou, "Application of Anomaly Detection Algorithms for Detecting SYN Flooding Attacks," in *Proc. of the IEEE Global Telecommunications Conference (GLOBECOM)*. IEEE, 2004.

[9] A. Ford, C. Raiciu, M. Handley, O. Bonaventure *et al.*, "TCP extensions for multipath operation with multiple addresses," 2013.