kriativ.tech

Edição Nº 9 – 11-10-2021
ISSN Print: 1646-9976 | ISSN Online: 2184-223X |
http://www.kriativ-tech.com
DOI: 10.31112/kriativ-tech-2021-10-57

Detection and Prevention of TCP SYN Flood DoS Attacks: Concepts

Pedro Ramos Brandao
Coordinator Professor at Instituto Superior de Tecnologias Avançadas – pedro.brandao@istec.pt
Jeremias Tavares
Master Degree Student at Instituto Superior de Tecnologias Avançadas - jeremias.tavares@my.istec.pt

Abstract: Internet security is a topic of high importance, and in recent years it has gained greater popularity with the growing wave of DoS attacks perpetuated through TCP SYN Flood. This kind of attack has several types of motivation: political, unfair competition, human evil. It is intended to deepen the concepts related to this type of attack architecture and which vulnerabilities are exploited that possibly facilitate the success of the SYN Flood. The central attack prevention systems, IDS and IPS, are presented conceptually. A simulation of the attack in a virtually recreated environment is depicted as proof of concept and execution. In contrast, there is evidence of the greater demand and growing sophistication of the means of detection and prevention using modern technologies.

Keywords: Cybersecurity, IDS, IPS, SYN Flood, DoS.

I. Introduction

In recent years the Internet has become more popular globally, and security has a fundamental role in ensuring the quality of service and digital privacy. The responsiveness of available services is critical for the vast majority of people and represents an essential factor in the experience of using and disseminating information. However, the dangers from attacks have become more popular. For example, one of the adverse effects of the pandemic caused by the covid-19 virus was the increase in the number of perpetrated attacks causing further disruption to all legitimate Internet service users. The well-known SYN Flood attack belongs to the Deny of Services (DoS) typology (Figure 1). It is directed at the transport layer, specifically at the TCP protocol through a flooding attack caused by sending synchronization datagrams.

Figure 1. Deny of Service Attacks

Given its simplicity of execution, inexperienced attackers can perform it, causing significant damage to web services. This type of attack aims to deplete server bandwidth and resources and has been ranked the most popular since 2017[1]. This paper intends to address fundamental concepts to realize the importance of developing mechanisms to detect and prevent and

DoS SYN Flood attacks and demonstrate the practical attack's complexity through proof of concept.

The remaining essay will be structured as follows: Chapter II - the relevant and related work is presented; Chapter III - the approach made to the problem presented and its methodology is presented; Chapter IV - the results of the simulated attack are presented; Chapter V - the results are discussed; Chapter VI - the conclusion is drawn.

II. Related Work

The detection tools for SYN Flood attacks have evolved significantly. Although endless models and algorithms using deep learning (DL) have been developed in the last decades, the effects are still genuinely adverse. They represent a constant alert in the security of network typologies [2]. The detection and prevention models analyzed in [2] use neural networks and consist of mathematical models capable of performing information processing and the network structure of the human brain. Therefore, the detection of DoS attacks represents one of the biggest challenges for cybersecurity.

Shen has developed an attack prediction model known as Teresias xspace, whose approach is centered on Recurrent Neural Networks (RNN) and allows for predicting the possibility of the occurrence of imminent attacks on a given host machine based on observations [3]. The study [4] showed 97% effectiveness in detecting and isolating attacks on mobile clouds using a DL approach and implementing a layered learning algorithm using a Restricted access Boltzmann Machine (RBM). The model is fitted using documented data to simulate considerable volumes for detection. According to Zhang et al. [5] the application of the Kullback-Leibler divergence allows detecting stealth attacks, although, in its genesis, there is the assumption that the traffic targeted for inspection is Gauss distributed. In the prediction model in [6], the DL-based approach merges autoencoder with SVM to register intrusions. In [7], the continuously ranked probability score (CRPS) is used to quantify the heterogeneity between a new observation and the traffic distribution considered to be normal. The CRPS approach, in the face of a DoS attack, consists of the comparison measurement of each traffic novelty in the network regarding the attack-free traffic distribution. Also, in [7], the CRPS method presents results based on rigorous, sharp, and accurate comparative analysis, assuming that large CRPS values represent a potential attack in progress. In [12][7] based on [13], the bounds are defined as follows:

$$CRPS(F,x) = \int_{-\infty}^{\infty} (F(y) - 1\{y \geqslant x\})^2 dy$$

Formula 1.

Shin et al. [8] developed AVANT-GUARD, a solution capable of reducing the impact caused by saturation attacks using change flow management on devices with OpenFlow, based on the SYN Proxy implementation, i.e., only total TCP 3-Way Handshake flows are exposed (Figure 2). In [8], the switch and/ or router take on the role of proxy when synchronizations are requested by sending packet-in to the controllers only when a connection to the client is established, thus making it difficult for the SYN Flood to reach the target server (Figure 3).

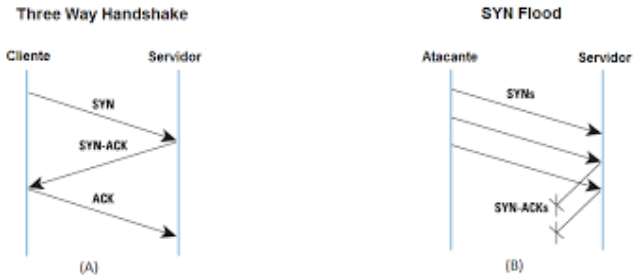

Figure 2: TCP protocol correct operation (left) vs. TCP SYN Flood modus operandi (right). Adapted[1].

[1] https://www.google.com/url?sa=i&url=http%3A%2F%2Fwww.defesacibernetica.ime.eb.br%2Fpub%2Frepositorio%2F2012-Rasinhas_Pedro.pdf&psig=AOvVaw2NRwPHierlnNIyDDQt_gaP&ust=1627077701410000&source=images&cd=vfe&ved=0CAsQjRxqFwoTCPD44ovl9_ECFQAAAAAdAAAAABAD

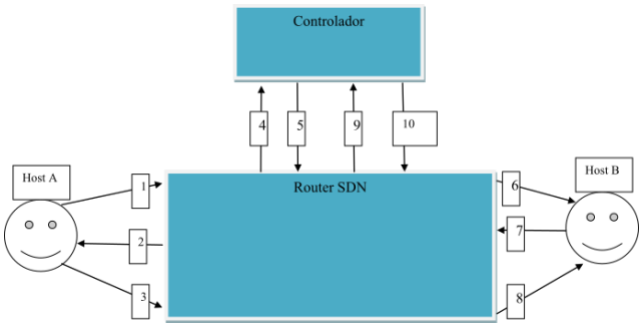

Figure 3. AVANT-GUARD architecture

More recently, in [9], the solution known as INDIGSOL (Indigenous Solution) was presented, a hybrid solution based on four modules: Node registration and validation; Packet capture; Dynamic control summation, and Hook activator. Compared to other existing approaches, the study results in [9] show gains in detecting a more significant number of malicious packets, a faster detection, and decreased response time or attack handling.

OPERETTA [14] allows SYN Flood detection in SDN networks, is installed on the controllers and acting as a proxy and facilitates the establishment of connections between client and server. After successful three-way handshake validation, an RST flag is sent to the client (an additional step), requesting re-establishing the connection. As per [14], a limit of unsuccessful connections is set for each host; once this limit is exceeded, the controller blocks the host, assuming it to be an attacker.

SLICOTS [15] was developed as a tool for preventing and detecting SYN Flood attacks, using the monitoring of the TCP three-way handshake process by implementing temporary rules. At the same time, the connection is not ended, in the meantime in a half-open state. Instead, after a significant number of half-open, a rule is put in place to block the suspect host permanently.

III. Approach

Several authors consider Software-defined networks (SDN) a next-generation networking technology for computer science and engineering [10]. This is because SDN technology provides a fusion of existing network infrastructures, converting it into a structurally centralized network capable of supporting the open communication protocol, OpenFlow [11] (Figure 4).

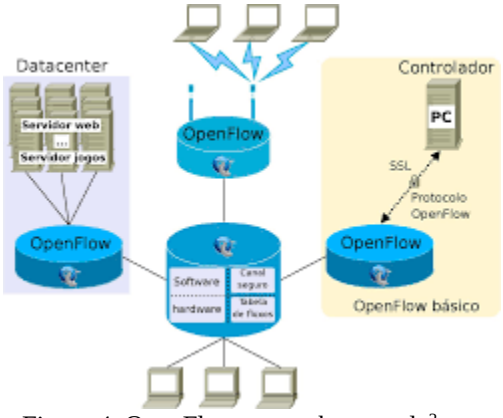

Figure 4. OpenFlow network example[2]

SDNs provide a global view of the controller, making it easier for the network administrator to make decisions regarding intrusion security through intrusion detection prevention systems (IDPS). As discussed in the previous chapter, and given the complexity and dynamics of network traffic, for [10], it is necessary to adopt measures that allow the correct exploitation of the real-time traffic monitoring capabilities of SDN controllers. Through the exploitation of existing vulnerabilities in network protocols, and as such, it is considered essential to improve the mechanisms to prevent attacks that aim to steal information and, in some cases, precede extortion practices. DoS attacks objectively exhaust the resources of the target infrastructure to cause its unavailability. The SYN Flood is a

[2] Source: https://www.researchgate.net/profile/Christian-Esteve-Rothenberg/publication/266292305_OpenFlow_e_redes_definidas_por_software_um_novo_paradigma_de_controle_e_inovacao_em_redes_de_pacotes/links/542fec440cf27e39fa99b9a7/OpenFlow-e-redes-definidas-por-software-um-novo-paradigma-de-controle-e-inovacao-em-redes-de-pacotes.pdf

direct, volumetric type of attack, characterized by sending large volumes of traffic directly to the potential victim via a large number of Transmission Control Protocol (TCP) connection requests, in such a way as to cause the exhaustion of available resources. SYN flooding attacks occur at the fourth layer of the OSI model, the transport layer, specifically in the TCP protocol, and benefit from the Three-Way Handshake. The original Internet protocol (IP) address of the attacking machine is changed in synchronization requests (SYN) to the server. Subsequently, the server responds to each request with an SYN-ACK packet (Synchronize-acknowledge) and waits for the ACK datagram packet to establish the connection. Still, this response will never be sent since the source address of the SYN request is false. Therefore, the link will remain in the semi-open SYN-RECV state; IP is out of reach for the victim. Furthermore, practical intrusion detection and prevention methods are fundamental for recognizing the characteristics of attacks on a server. The Intrusion detection system (IDS) (Figure 5) aims to detect security breaches through misuse or anomalies in a given system.

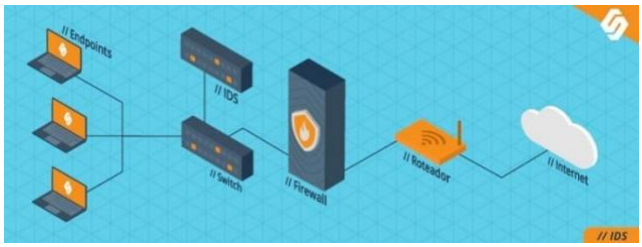

Figure 5. Implementation of an IDS system on the network[3]

The function of IDS is to detect, identify, and respond to unlawful activities. When detected by IDSes, internal or external threats result from the sharing of data from various sources across the network and other systems[4]. An IDS is a network security technology created to detect vulnerability exploits against an application or computer and is considered a listening device. An IDS's primary function is to detect possible threats and communicate them to an administrator by monitoring traffic. More recently, the intrusion prevention system (IPS) has been developed to represent a network security/prevention technology that scans network traffic flows to detect and prevent vulnerability exploits[5]. Generally speaking, the IPS sits behind the firewall, providing a complementary analysis layer. It is common for companies to adopt systems with hybrid architecture with integrated IDS/IPS called IDPS. For instance, in [16], we can check the SYN Flood attack environment and the normal flow of packets sent and received via TCP (Figure 6).

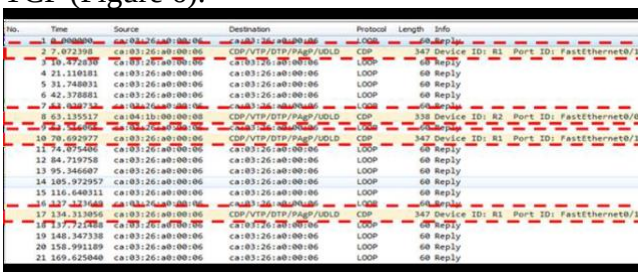

Figure 6. Network flow before SYN Flooding

From the real-time observation of SYN Flooding, we notice an increasing number of packets received with the same source (Figure 7) to cause a high number of half-open connections. Thus, flood attacks objectively cause the server's resources to be exhausted.

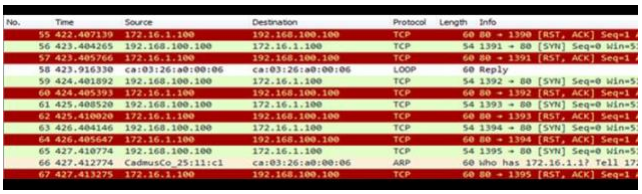

Figure 7. Network Flow during SYN Flooding

[3] https://blog.starti.com.br/ids-ips/

[4] https://www.paloaltonetworks.com/cyberpedia/what-is-an-intrusion-detection-system-ids

[5] https://www.paloaltonetworks.com/cyberpedia/what-is-an-intrusion-prevention-system-ips

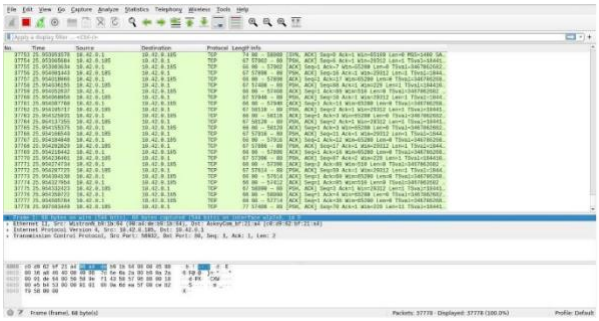

Figure 8. Wireshark Operation Analysis

IV. Results

A test environment was emulated, and the number of requirements needed to elaborate the exemplification of the TCP SYN Flood attack was demonstrated, namely: a Virtual machine with the Kali Linux system (figure 9).

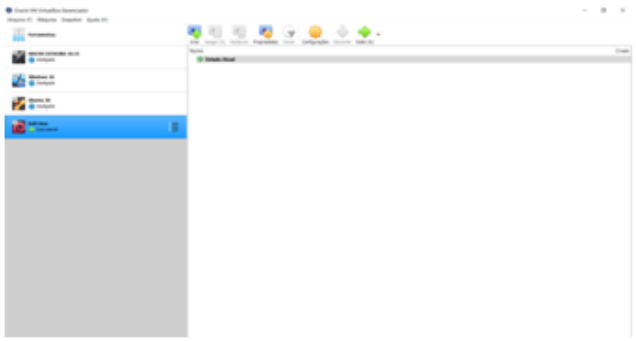

Figure 9. Virtual environment, Kali Linux attacker machine

As mentioned in the previous chapter, it is a DoS attack and acts on the TCP protocol that SYN Flood. To plan the attack, the following minimum is required (Figure 10):

A. To enter the target address - RHOSTS.
B. Directing the attack to an available port - RPORT.
C. Masking the source IP - SHOST.
D. Taking TIMEOUT into account.

On the other hand, implementing an IDS solution (Figure 11) on a server requires additional requirements, a more significant number of resources, and human assets.

Figure 10. Metasploit® Environment

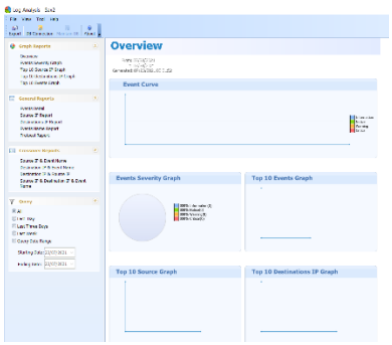

Figure 11. SAX2 IDS Interface

V. Discussion

The main concepts related to the detection and prevention of DoS SYN Flood attacks are crucial to understanding the current level of security and that which is intended to be achieved in the future. Indeed, one could discuss integrating technologies based on DL or artificial intelligence algorithms into all IDS/IPS systems. However, it would be necessary to adapt the services to the reality of most companies and analyze the cost-benefit ratio of security implementation.

VI. Conclusion

The paradigm of SYN Flood attack detection and prevention mechanisms, proven to offer numerous benefits and take advantage of the information generated and shared, on the other hand, faces challenges for the future. The speed of

execution of flooding attacks and the need for few resources were concluded. In contrast, prevention mechanisms necessarily have the challenge of exponentially increasing the complexity of their mitigation models.

VII. References

[1] A. Verma, R. Saha, G. Kumar, and T. Kim, "The Security Perspectives of Vehicular Networks: A Taxonomical Analysis of Attacks and Solutions", *Appl. Sci.*, vol. 11, no. 10, 2021, doi: 10.3390/app11104682.

[2] A. E. Ibor, F. A. Oladeji, O. B. Okunoye, and O. O. Ekabua, "Conceptualisation of Cyberattack prediction with deep learning", *Cybersecurity*, vol. 3, no. 1, 2020, doi: 10.1186/s42400-020-00053-7.

[3] Y. Shen, E. Mariconti, P. A. Vervier, and G. Stringhini, "Tiresias: Predicting Security Events Through Deep Learning", in *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*, 2018, pp. 592–605, doi: 10.1145/3243734.3243811.

[4] K. K. Nguyen, D. T. Hoang, D. Niyato, P. Wang, D. Nguyen, E. Dutkiewicz, "Cyberattack detection in mobile cloud computing: A deep learning approach", 2018 IEEE Wireless Communications and Networking Conference (WCNC), Barcelona, pp. 1-6 2018, http://hdl.handle.net/10453/131928.

[5] Q. Zhang, K. Liu, Y. Xia, & A. Ma, "Optimal Stealthy Deception Attack Against Cyber-Physical Systems", *IEEE transactions on cybernetics*, *50*(9), 2019, 3963–3972. https://doi.org/10.1109/TCYB.2019.2912622

[6] M. Al-Qatf, Y. Lasheng, M. Al-Habib, K. Al-Sabahi, "Deep learning approach combining sparse autoencoder with SVM for network intrusion detection", IEEE Access 6, pp. 843–856, 2018.

[7] B. Bouyeddou, B. Kadri, F. Harrou, Y. Sun, "DDOS-attacks detection using an efficient measurement-based statistical mechanism", *Eng. Sci. Technol. an Int. J.*, vol. 23, no. 4, pp. 870–878, 2020, doi: 10.1016/j.jestch.2020.05.002.

[8] S. Shin, V. Yegneswaran, P. Porras, G. Gu, "AVANT-GUARD: Scalable and Vigilant Switch Flow Management in Software-Defined Networks", in *Proceedings of the 2013 ACM SIGSAC Conference on Computer & Communications Security*, 2013, pp. 413–424, doi: 10.1145/2508859.2516684.

[9] M. Junaid *et al.*, "An Indigenous Solution for SYN Flooding", *Rev. GEINTEC-GESTAO Inov. E Tecnol.*, vol. 11, no. 4, pp. 2998–3022, 2021

[10] M. Rahouti, K. Xiong, N. Ghani, F. Shaikh, "SYNGuard: Dynamic threshold-based SYN flood attack detection and mitigation in software-defined networks", *IET Networks*, vol. 10, no. 2, pp. 76–87, 2021, doi: https://doi.org/10.1049/ntw2.12009.

[11] N. McKeown *et al.*, "OpenFlow: Enabling innovation in campus networks", *Comput. Commun. Rev.*, vol. 38, pp. 69-74, 2008, doi: 10.1145/1355734.1355746.

[12] E. Grimit, T. Gneiting, V. Berrocal, N.A. Johnson, "The continuously ranked probability score for circular variables and its application to mesoscale forecast ensemble verification", Quart. J. R. Meteorol. Soc. 132 (621C) 2006, https://doi.org/10.1256/qj.05.235.

[13] J. Matheson, R. Winkler, "Scoring rules for continuous probability distributions", Manage. Sci. 22 (10) 1087–1096, 1976, https://doi.org/10.1287/mnsc.22.10.1087.

[14] S. Fichera, L. Galluccio, S. Grancagnolo, G. Morabito, S. Palazzo, "Operetta: An OpenFlow-based remedy to mitigate TCP synflood attacks against web servers", Computer Networks, 92:89–100, 2015.

[15] R. Mohammadi, R. Javidan, M. Conti. "Slicots: An sdn-based lightweight countermeasure for TCP syn flooding attacks", IEEE Transactions on Network and Service Management, 2017.

[16] B. A. Khalaf *et al.*, "A simulation study of syn flood attack in a cloud computing environment," *AUS J.*, vol. 26, no. 1, pp. 188–197, 2019.