High Volume DDOS Attack Mitigation on an SDN Software Defined Network

Presenting by:

Luis Eduardo Escobar Garcés

Advised by Dr. Jesús Arturo Pérez Díaz and MC Noe Marcelo Yungaicela Naula

February 2021

Abstract - In a world where there are more and more interconnected devices, the increase in DDOS-type attacks is a concern as it is one of the most common types of attacks. This work seeks to implement a software-defined network (SDN) with connected switches and devices, to simulate a flood-type DDOS attack with TCP or UDP protocols, through the traffic flow of normal users and malicious packets that seek to destroy the SDN memory to end the server service, and mitigate the attack using techniques such as machine learning, deep learning, some variant of the SYN cookie technique or the use of entropy. There is already a way to identify attacks (IDS) proposed by the advisors of this work, so it is not the objective of this work.

I INTRODUCTION

DDOS attacks are one of the most frequent cyber-attacks today [1]. The architecture of the SDN allows to have a better control over the network, but being centralized, it becomes vulnerable to this type of attack, this causes that methods are sought to solve this problem. To begin, the following concepts with which you will be working are defined: Flood, DDOS and SDN type attack.

DDOS

CISCO defines DDOS attacks (Distributed Denial Of Service) as the bombardment of simultaneous data requests to a central server in which the attacker generates these requests from multiple compromised systems with the aim of depleting the bandwidth of target's internet and RAM to crash the target's system or disrupt their business [2].

SDN

Software Defined Network (SDN) is a network architecture in which the control plane and the data plane are separated, this allows to have network visibility, centralized control, scheduling capability, software-based traffic analysis, easily implement security functions within the network, such as firewall, intrusion detection system (IDS), intrusion prevention system (IPS) among others [3].

FLOOD ATTACK

A flood attack is a type of DDOS attack that seeks to fill with malicious traffic through botnets (networks of zombie computers that are remotely controlled by cybercriminals) to block the server service. Flood attacks can be carried out at layer 3 (network), 4 (transport) and 7 (application), of the OSI model [4].

II PROBLEMATIC AND JUSTIFICATION

DDOS attacks have been presented with higher frequency in recent years, At the end of 2020, the FBI recommended that schools implement security mechanisms against DDOS attacks as there are attacks large enough to take them out of service [5]. Kaspersky mentions that the average loss of companies being attacked by DDOS, can be 20 thousand dollars per hour [6].

DDOS-type attacks may not greatly affect the profits of small or medium-sized businesses, but it can affect their status and the level of trust of their customers, in a survey of IT security professionals at Infosecurity, it stated that 42% of those surveyed, that the loss of customers trust was the worst effect of DDOS [7]. SDN-type architectures allow better control of the network, but by having network control centralized, SDNs can be the victim of attacks such as traffic flooding (flood) by botnets, which makes this architecture vulnerable and this is why, that it is necessary to find a way to prevent, identify and mitigate attacks these attacks to avoid the economic loss and trust of customers due to lack of service. This work will be more focused on the configuration of a SDN with Mininet and ONOS and implementing the mitigation strategy since there is already an intrusion detection system (IDS) provided by the advisers of this work.

III STATE OF THE ART

Due to the centralized control architecture of SDN, contributions have been made to identify and mitigate flood-type DDOS attacks.

In [3] Safaa Mahrach, in January 2020 presented a proposal to mitigate DDOS flood attacks using a SDN architecture, they made a program with the P4 programming language, the program connects the controller with the switch to install it on the latter and checks the flow table where the TCP protocol monitors, then checks if the packet received by the client is SYN, if it is, it generates a SYN Cookie based on data from the received packet, then it is hashed to ensure the integrity of the packet and it sends it to the client, if the client responds with a TCP ACK and matches the local SYN, it is accepted and if not that traffic is rejected. They ensure that their strategy does not use much memory space since it is done directly on the switch and that 100% of the benign packets are accepted, while the bad ones are rejected.

In [9] by Nguyen Ngoc Tuan, in February 2020 contributed in a way to mitigate flood-type DDOS attacks in SDN architectures using machine learning with the Nearest-Neighbor (KNN) algorithm and XGBoost, which is an algorithm based on machine learning that belongs to the increasing gradient. The algorithm mitigates flood attacks for TCP-SYN and ICMP protocols, with an efficiency to mitigate the attack of 98% in TCP-SYN and 99% in ICMP, benign traffic is not affected. The KNN algorithm was used in the entropy values after having calculated the ports or the number of ICMP packets. For their dataset they used the CAIDA 2007 dataset with a size of 21 GB [9].

In [9] Jesús Galeano Brajones, contributed to the detection and mitigation of flood-type DDOS attacks in an SDN. The algorithm works as follows, they retrieve the switch states, the entropy values are calculated for the source IP, destination IP, source port and destination port, when the entropy values fall and the algorithm detects it, depending on the size of the window, flow rules are added to the switches using OpenFlow to mitigate attackers, and benign traffic remains intact. In 10 tests that were carried out, in 3 of them the attack was not detected and mitigated, the problem that occurred was with the size of the window, which is 60 seconds, if the attack is launched just when the size of the window ends it causes the entropy values do not vary enough with the expected thresholds, and at the beginning of the next time window, no attack is detected because the entropy values did not exceed the predicted limits. The ratio of false positives detected depends on the value given to ϴ of the window size, giving a percentage of 20% in false positives.

In [10] Marcos et al. contributed with an identification and mitigation technique for DDOS attacks in an SDN. For detection it is used a Gated Recurrent Units (GRU) a supervised deep learning method that have 2 sub-modules, 1st sub-module can go back to decide whether to use important historical data of the traffic flow for analysis, and the 2nd sub-module that classifies them as normal or abnormal traffic. The mitigation module divided in 2 sub-modules, the 1st sub-module determines the optimal countermeasure against the detected anomaly, the 2nd sub-module sends the drop policy to the SDN controller for implementation. For identification the proposal was tested against CNN, LSTM, DNN, SVM, LR, KNN and GD algorithms, considering accuracy, precision, recall and F-measure, showing that GRU had an attack identification success average of 97.09% of the previous variables with CICIDS2018 dataset and a 99.94% with CICDDOS2019 dataset. For mitigation since the mitigation approach directly depends on the detection's method efficiency it was tested the number of legitimate flows dropped and the malicious flows not dropped, with legitimate flows dropped, GRU achieved the most balanced outcome, guaranteeing both the detection of malicious flows and preserving legitimate ones, for malicious flows not dropped GRU once more achieved the most balanced outcomes in this test scenario.

In [11] Metheus et al. contributed with a detection and mitigation system in SDN architectures for DDOS and Portscan attacks. The proposal has three modules, the first module employs a Deep Learning algorithm called Long Short-term memory (LSTM) that allows to predict the normal behavior on the network learning long term dependencies, formed by 3 gates, forget, input and output gate in order to choose what previous data is necessary to remember and use, then it is calculated the Shannon's entropy of non-qualitative data like source ports and ip, and destiny port to work with them, the next step is to define the threshold between the predicted traffic and the real traffic using Bienaymé-Chebyshev's inequality. The second module has 2 sub-modules the first one detects anomalous events, using fuzzy logic theory to help with the decision taking for anomaly detection, the second sub-module uses membership function called Gaussian membership function and the anomaly score to determine rather the traffic is normal, portscan or DDOS. The third module establish an Event Condition – Action model (ECA) to choose a proper action, in this module it is implemented a Safe List that contains IP flow attributes of legit users for a determined time of 5 minutes if there is a case of a flash crowd anomaly of benign users, the IP's and ports are compared with the safe list to not drop their traffic flow. The technique proposed were tested in 2 scenarios, one with a emulated scenario, and the 2nd one with CICDDOS 2019 dataset, the precision, recall and AUC were tested, in the 1st scenario LSTM-FUZZY had an average of more of 99% of success with these variables, in the 2nd scenario more of 99% of success and compared to the others methods compared in the literature, the proposal is success in detecting and mitigation.

Reviewing the previous contributions, the 5 works find a solution to identify and mitigate flood-type DDOS attacks, however there are some factors that can be analyzed, Safaa MAHRACH in [3] proposes to use the SYN cookie technique in which it works as the "authentication" of other systems, its high success rate is good and also ensures that this technique can be implemented in SDN or normal architectures, however the testing was done in a range of 0 to 1000 TCP-SYN packets per second, it would be interesting to do a test with millions of packets and verify that the bandwidth and success rate are not altered. With respect to the second work by Nguyen Ngoc Tuan in [8], the percentage of success in the identification and mitigation of 98% is high, in addition to using machine learning so that the size of the time window is modified depending on the traffic received to achieve a better mitigation efficiency. The third work [9] by Jesús Galeano-Brajones, also uses the use of entropy for the identification and mitigation of attacks, however they present a problem with the size of the time window since it does not adapt with respect to the flow of traffic it receives, causing that when the cycle of the time window ends, if the attack is received in that period of time, the attack will not be identified, in addition to having a percentage of 20% of false positives, which in the first and second job they don't have. The fourth work in [10] Marcos et al proposed a GRU supervised deep learning method Gated Recurrent Units that can go back to use important historical data for its analysis having a identification success average of 97.09% of the previous variables with CICIDS2018 dataset and a 99.94% with CICDDOS2019 dataset, nevertheless its mitigation method is 100% related with the identification method efficiency it would be interesting to evaluate the drop time window usage of the mitigation module to minimize the computational cost and improve the mitigation outcomes. In [11] Metheus et al. proposed LSTM-FUZZY that is a deep learning algorithm that can use historical data in order to learn previous patterns of the normal traffic flow and abnormalities in the network, for both scenarios the 1st with a simulated attack and 2nd with CICDDOS2019 dataset had an average of success in the recall, precision and AUC of more than 99% identifying and mitigating Portscan or DDOS attacks, nevertheless it would be interesting to test with more scenarios with other topologies and to incorporate mitigation policies to meet new demands that might emerge in SDN environments.

The purpose of the proposed project is to implement an SDN in mininet and configure the ONOS controller for the mitigation of DDOS flood attacks in either TCP or UDP protocols, using machine learning algorithms or entropy or variants of the SYN cookie technique, or deep learning, which mitigates attacks and reduce window time size problems if the algorithm requires it.

IV DEFINITION OF OBJECTIVES

GENERAL OBJECTIVE

The objective of this work is to implement a mitigation technique for high volume DDOS attacks like flood TCP-SYN or UDP in a simulated environment with mininet and the ONOS controller.

SECUNDARIES OBJECTIVES

Review and document of different DDOS high-volume mitigation strategies used, make an analysis and tests of the performance, response time, scalability of the results of the proposed mitigation algorithm in the SDN, implemented with mininet and ONOS controller.

V CONTRIBUTIONS AND EXPECTED PRODUCTS

At the end of this work, it is expected to deliver an algorithm in code for mitigation of flood-type DDOS attacks with TCP or UDP protocols in a virtual machine with an SDN in a simulated environment with Mininet and the ONOS controller, with the trained IDS given by the advisers of this work, implementing mitigation techniques such as entropy, SYN cookie or KNN (K nearest neighbor) supervised machine learning algorithm or other machine learning algorithm.

Also it is expected to deliver a tesina with the review of several techniques for mitigating DDOS high volume attacks like flood attack, and with the analysis, tests, evaluation of performance, response time and scalability of the proposed algorithm in code for mitigation of flood-type DDOS attacks.

VI METHODOLOGY AND ACTIVITY PLAN

In the fig 1, we can see that the project is divided into 4 phases. The first is the identification of different mitigation techniques in complex DDOS attacks, where mitigation strategies for large volumes of attacks such as SYN flood, UDP flood, and Domain name Server attack amplification will be investigated, analyze the mitigation strategies found, mention its advantages, requirements and limitations, identify mechanisms for the validation of the mitigation of these attacks. The second phase is to implement the mitigation strategy, identify the IDS configuration that is based on Deep learning, recognize the available parameters that the IDS provides to properly define the mitigation mechanisms, improve an existing strategy to mitigate high-volume attacks. In the third phase, it is to evaluate the implemented solution, perform online tests of the proposed strategy using the existing testbed, and evaluate the evaluated parameters based on the previously reviewed literature. In the fourth phase, report all the technical and scientific discoveries of the project.

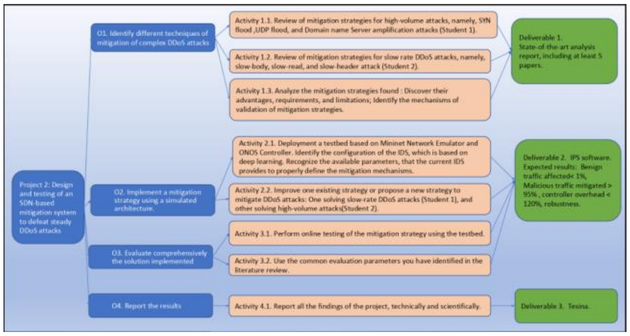
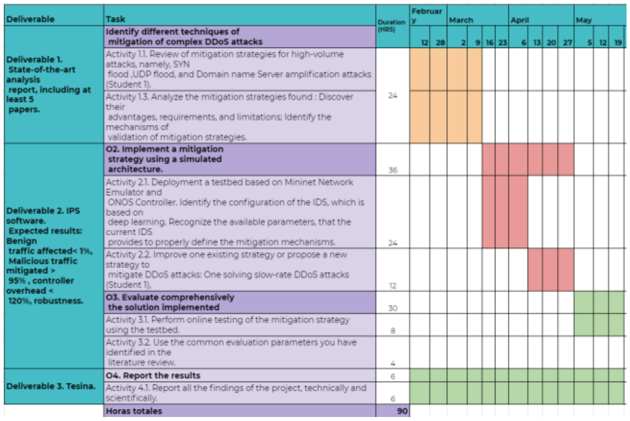

Project 2: Design and testing of an SDN-based mitigation system to defeat steady DDoS attacks
O1. Identify different techniques of mitigation of complex DDoS attacks
O2. Implement a mitigation strategy using a simulated architecture.
O3. Evaluate comprehensively the solution implemented
O4. Report the results
Activity 1.1. Review of mitigation strategies for high-volume attacks, namely, SYN flood ,UDP flood, and Domain name Server amplification attacks (Student 1).
Activity 1.2. Review of mitigation strategies for slow rate DDoS attacks, namely, slow-body, slow-read, and slow-header attack (Student 2).
Activity 1.3. Analyze the mitigation strategies found : Discover their advantages, requirements, and limitations; Identify the mechanisms of validation of mitigation strategies.
Activity 2.1. Deployment a testbed based on Mininet Network Emulator and ONOS Controller. Identify the configuration of the IDS, which is based on deep learning. Recognize the available parameters, that the current IDS provides to properly define the mitigation mechanisms.
Activity 2.2. Improve one existing strategy or propose a new strategy to mitigate DDoS attacks: One solving slow-rate DDoS attacks (Student 1), and other solving high-volume attacks(Student 2).
Activity 3.1. Perform online testing of the mitigation strategy using the testbed.
Activity 3.2. Use the common evaluation parameters you have identified in the literature review.
Activity 4.1. Report all the findings of the project, technically and scientifically.
Deliverable 1. State-of-the-art analysis report, including at least 5 papers.
Deliverable 2. IPS software. Expected results: Benign traffic affected< 1%, Malicious traffic mitigated > 95% , controller overhead < 120%, robustness.
Deliverable 3. Tesina.

Fig. 1. General objectives

To complete the objectives of the project, a plan was developed that will be defined as in the fig 2.



Deliverable	Task	Duration (HRS)	February		March				April				May		
			12	28	2	9	16	23	6	13	20	27	5	12	19
Deliverable 1. State-of-the-art analysis report, including at least 5 papers.	**Identify different techniques of mitigation of complex DDoS attacks**														
	Activity 1.1. Review of mitigation strategies for high-volume attacks, namely, SYN flood ,UDP flood, and Domain name Server amplification attacks (Student 1).	24	🟧	🟧	🟧	🟧									
	Activity 1.3. Analyze the mitigation strategies found : Discover their advantages, requirements, and limitations; Identify the mechanisms of validation of mitigation strategies.		🟧	🟧	🟧	🟧									
Deliverable 2. IPS software. Expected results: Benign traffic affected< 1%, Malicious traffic mitigated > 95% , controller overhead < 120%, robustness.	**O2. Implement a mitigation strategy using a simulated architecture.**	36					🟥	🟥	🟥	🟥	🟥	🟥			
	Activity 2.1. Deployment a testbed based on Mininet Network Emulator and ONOS Controller. Identify the configuration of the IDS, which is based on deep learning. Recognize the available parameters, that the current IDS provides to properly define the mitigation mechanisms.	24					🟥	🟥	🟥						
	Activity 2.2. Improve one existing strategy or propose a new strategy to mitigate DDoS attacks: One solving slow-rate DDoS attacks (Student 1),	12								🟥	🟥	🟥			
	O3. Evaluate comprehensively the solution implemented	30											🟩	🟩	🟩
	Activity 3.1. Perform online testing of the mitigation strategy using the testbed.	8											🟩	🟩	🟩
	Activity 3.2. Use the common evaluation parameters you have identified in the literature review.	4													
Deliverable 3. Tesina.	**O4. Report the results**	6	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩
	Activity 4.1. Report all the findings of the project, technically and scientifically.	6	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩
	Horas totales	**90**													

Fig. 2. Gantt diagram

REFERENCES

[1] Cisco. Cyber attack—What are common cyberthreats? (s. f.). Cisco. Retrieved 20 of February of 2021, from https://www.cisco.com/c/en/us/products/security/common-cyberattacks.html

[2] Cisco. What is a ddos attack? Distributed denial of service. (s. f.). Cisco. Retrieved 21 of February of 2021, from https://www.cisco.com/c/en/us/products/security/what-is-a-ddos-attack.html

[3] Mahrach, Safaa & Haqiq, Abdelkrim. (2020). DDoS Flooding Attack Mitigation in Software Defined Networks. International Journal of Advanced Computer Science and Applications. 11. 10.14569/IJACSA.2020.0110185.

[4] Weisman Steve. What is a DDoS attack? (2020). Retrieved 21 of February of 2021, from https://us.norton.com/internetsecurity-emerging-threats-what-is-a-ddos-attack-30sectech-by-norton.html

[5] Oleg Kupreev, Ekaterina Badovskaya, Gutnikov Alexander. Ataques DDoS en el cuarto trimestre de 2020. (2021). Retrieved 21 of February of 2021, from https://securelist.lat/ddos-attacks-in-q4-2020/92850/

[6] Karspesky. Distributed denial of service: Anatomy and impact of ddos attacks. (2021, January 13). Usa.Kaspersky.Com. https://usa.kaspersky.com/resource-center/preemptive-safety/how-does-ddos-attack-work

[7] Group, I. D. M. (2018, September 13). La p&eacute;rdida de la confianza, principal consecuencia de los ataques DDoS | Security and Risk Management. Discover The New. https://discoverthenew.ituser.es/security-and-risk-management/2018/09/la-perdida-de-la-confianza-principal-consecuencia-de-los-ataques-ddos

[8] Tuan, N. N., Hung, P. H., Nghia, N. D., Tho, N. V., Phan, T. V., & Thanh, N. H. (2020). A DDoS Attack Mitigation Scheme in ISP Networks Using Machine Learning Based on SDN. Electronics, 9(3), 413. MDPI AG. Retrieved from http://dx.doi.org/10.3390/electronics9030413

[9] Galeano-Brajones, J., Carmona-Murillo, J., Valenzuela-Valdés, J. F., & Luna-Valero, F. (2020). Detection and Mitigation of DoS and DDoS Attacks in IoT-Based Stateful SDN: An Experimental Approach. Sensors, 20(3), 816. MDPI AG. Retrieved from http://dx.doi.org/10.3390/s20030816

[10] Marcos V.O. Assis, Luiz F. Carvalho, Jaime Lloret, Mario L. Proença. (2020). A GRU deep learning system against attacks in software defined networks, Journal of Network and Computer Applications, Volume 177, 2021, 102942, ISSN 1084-8045. Retrieved from: https://doi.org/10.1016/j.jnca.2020.102942

[11] M. P. Novaes, L. F. Carvalho, J. Lloret and M. L. Proença. (2020) "Long Short-Term Memory and Fuzzy Logic for Anomaly Detection and Mitigation in Software-Defined Network Environment," in IEEE Access, vol. 8, pp. 83765-83781, 2020, Retrieved from: 10.1109/ACCESS.2020.2992044.