PAPER • **OPEN ACCESS**

Slowloris DoS Attack Based Simulation

To cite this article: Shima Sabri *et al* 2021 *IOP Conf. Ser.: Mater. Sci. Eng.* **1062** 012029

Slowloris DoS Attack Based Simulation

Shima Sabri[1*], Noraini Ismail[2] and Amir Hazzim[1]
[1]Faculty of Computing and Multimedia, Kole Universiti Poly-Tech Mara, Jalan 6/91, Taman Shamelin Perkasa, Cheras, 56100 Kuala Lumpur
[2]Faculty of Ocean Engineering Technology and Informatics, Universiti Malaysia Terengganu, 21030 Kuala Nerus, Terengganu Darul Iman, Malaysia

*Corresponding e-mail: shimasabri@kuptm.edu.my

Abstract. Denial of Service (DoS) attack is a type of cybercrime when an internet site is unavailable to be accessed by the user. The DoS attack is one of the popular attacks which can be launched by using a single machine and could take down many web servers by sending a lot of request to the server until it is disconnected. The objective of this paper is to simulate the launch and prevention against the DoS attack towards the website. The simulation of DoS attack is implemented by using ActivePerl Language and tested by using Slowloris DoS Attack. Both software shows a good combination of simulation program because it supports each other to exhaust the available connections on the servers. The result shows that the website connection is successfully unavailable to be accessed by legitimate user. In addition, the website still could not be served even clearing the cache of the browser. Finally, this paper discusses about future solution of DoS attack and recommendation for secure environment architecture.

Keywords: DoS Attack, Apache server, Slowloris DoS attack, simulation

1. Introduction

People are using website as one of communication technologies to manage life. Websites allows user to carry out online activities such as e-payment, shopping, reservation, booking as well as storing and retrieving information. To do this, a website must use apache web server to have the capability to manage and organize an online process. The problem arises when the websites that are using the Apache server becomes a main target by attackers to launch Denial of Service (DoS) attack [1]. DoS attack is a kind of cyber-attack in which the perpetrator seeks to make a machine or network resource unavailable to its intended users by temporarily or indefinitely disrupting services of a host connected to the Internet [1]. Services affected may include email, websites, online accounts (e.g., banking), or other services that rely on the affected computer or network. Authorized user will find that the website fails to provide the page in a short time and may lead to system coming down for a duration of time and then ultimately causing huge amount of revenue loss. Denial of Service has the most devastating effect, putting tremendous pressure on security experts to bring out effective defence solutions on a computer or a network [11]. These attacks could be implemented with a variety of tools and codes. However, there is a lack of solution for DoS attack has took place over the internet for nearly a decade. Hence, it becomes essential to carry out these attacks in small test bed environments in order to extrapolate the patterns of known threats, detecting and blocking them efficiently. These real time attacks are measured and analysed using the network traffic monitors. Also, this simulation presents defence strategies that could be executed in order to mitigate these attacks [1]. As shown in Figure 1, the attacker could inject multiple packets into the channels. As a result, the communication between the networks components are

blocked, which is seen as the DoS attack [2]. On top of that, the hacker aims to attack the system in order to degrade the estimation performance by injecting interference packets into the channel to block the packets from the victim [2].

In this paper, we point out traditional approach of Slowloris DoS attack based on statistical method, the signature methods and the cluster analysis. According the authors in [3], they stated that this method can recognize the Dos attacks of transport and network layer by filling the channel capacity.

This paper focuses on the web framework that uses the Apache server to become the key target for attackers to initiate Denial of Service (DoS) attack. To solve this problem, we simulate and launch a DoS attack using ActivePerl Language and test it using Slowloris DoS Attack. Results of the simulation reveals that the website is unreachable even after clearing the cache of a browser used by an attacker.

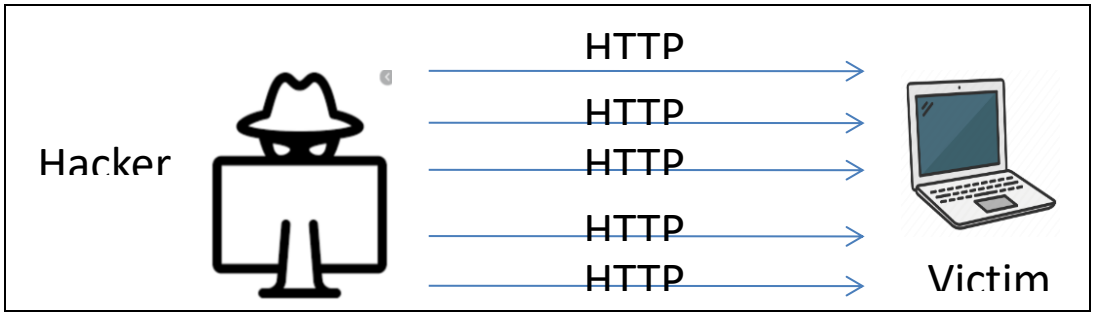

Figure 1. DoS attack.

2. An Online Instruction Detection Approach to Identify Low-Rate DoS Attack

The main purpose of a DoS attack is to consume the resources of a targeted victim. A victim may be a server, a router or any other kind of peripheral device that is connected to a network. This simulation will concentrate on HTTP server. In the past years, the popular DoS attack that has been introduced is Low-Rate Denial of Service attack that uses tiny amount of bandwidth [4].

There are several types of attack that is categorized as Low-Rate DoS Attacks that perform slow DoS attack. Although most of the slow rate DoS attack target the HTTP, they also can be adapted to attack other protocol. Table 1 summarises the comparison for DoS attack technique.

Slowloris attack is the most popular. It exploit the HTTP protocol by establishing a large amount of pending request with the targeted web server [5]. Additionally, the SlowLoris is underpinning the evolution of variety attacks such like SlowComm [5], Slow Next attack [4] , Slow Read Attack [6], and SlowReq Attack [7].

All of the attacks above are known as low and slow attack. Slowloris attack [8] is one of the most known slow rate Denial of Service attacks. Slowloris works by sends legitimate HTTP request to the web server with a catch, as it is incomplete HTTP. Due to this, the attack forces the targeted web server to wait for the end of the request, which will never be send by the attacker because the HTTP request is incomplete. In addition, Slowloris sends to the server a first part of incomplete HTTP request, therefore the wait timeout is triggered. After the tie wait timeout expired, an additional HTTP parameter is sent to the server, and the timeout is triggered again.

In relation to the above explanation, the objective of this paper is to attack the server by using Slowloris that allow perpetrator to overwhelm a targeted server by opening and maintaining many simultaneous HTTP connections between the attacker and the victim server. Further, by implementing DoS attack, the countermeasure can be recommended, thus enhancing the security level of the server to perform the task.

Table 1. Comparison for DoS attack technique.

Name of the Malicious	Malicious Behaviour	Parameter	Evaluation Parameter
DoS attack on Wireless Sensor Network	Disrupt privacy security data	Sensor nodes	Limited sensor node energy
SlowComm	Long request DoS attack	Internet service	Saturating the available connection
Resilient load frequency control	Time delay system model	Uncertainty of the system	Sleep internal and attack internal
R-U-Dead-Yet (RUDY / Slow HTTP POST RUDY)	Exploits the body of POST request.	Opens up desired number of concurrent connections	POST request
Slow Next attack	Exploiting the protocol persistency	Permits client to send multiple request in the same connection	Catching a specific amount of connection
Slow Read Attack	Sending legitimate HTTP request to the server at regular rate	Delay and will slow down the response of the server	Identifying a small client side reception buffer in the SYN packets sent to the victim during the connection establishments
SlowReq Attack	Makes the server to wait to an never-ending wait	Incomplete HTTP request is sent to the server	Each packet is composed by a single character
Slowloris	Opening connections to a targeted Web server and then keeping those connections open as long as it can	Utilizing partial HTTP requests	Incomplete HTTP request

3. Methodology

The detection and mitigation mechanism designed here are effective for small network topologies and can be extended to analogous large domains. Slowloris DoS Attack tool can be executed using multiple programming language such as Phyton, Perl, Go, Java, and Shell. The most common programming language used to run the attack is Perl and Phyton. This simulation will use Perl as a programming language purposely to extract information from a text file and printing out a report or for converting a text file into another form.

3.1. Active Perl

Perl is a software known as “Practical Extraction and Reporting Language”. As the Slowloris script is using Perl programming language, Perl is needed to run the script and to complete the DoS attack to the targeted website. As the attack will be executed just using the “Command Prompt” Perl is needed to have a functioning DoS attack.

3.2. Oracle VM VirtualBox

Oracle VM VirtualBox is used to install Linux Server. This is purposely to create two different environments in a VirtualBox and at the same time to implement a DoS attack protection onto the server.

The purpose of creating two servers is to simulate the attack and to demonstrate how the attack would be executed by a hacker when the hacker is trying to perform DoS attack on Apache web server. As VirtualBox can run multiple Operating systems by using single host in this way, this simulation attack will be run in emulated environment.

The simulation will be using two machines. Ubuntu will be used to run the Apache2 and to host the website and another machine observe on how DoS attack affects the normal users and normal traffic.

4. DoS Attack Architecture

The Slowloris attack was design to flood the webserver by sending multiple requests under the same IP just make sure that the webserver crashed or make it operate ineffectively. Based on Figure 2, the architecture of the attack starts from the attacker and down to the user. The attacker first launches the attack by sending multiple HTTP GET request to the website. The website then send to the webserver for processing in order to complete the request. Due to Slowloris sending too much request for the server to handle, it will be slowing the server down by overwhelming all meaningless request. Slowloris will begin to open connections. As a new one free up from someone else using the website, they will open until they got all the connection. As a result, there is no room for legitimate users to request that particular website.

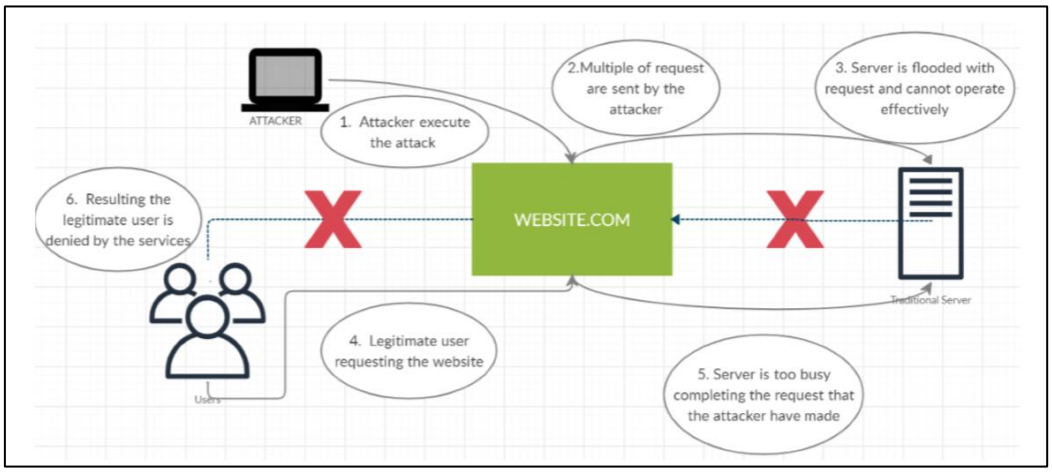

Figure 2. DoS attack architecture.

The command line to execute the DoS attack is shown in Figure 3 below: C:\Perl64>perl slowloris.pl –dns 192.168.56.1

Figure 2. Command Prompt interface to execute the attack.

5. Results and Discussion

The execution of DoS attack by using Slowloris DoS Attack tool is presented and the result is shown in Figure 4 and Figure 5. To check if the website is still working, the webpage needs to be refreshed. As a result, the output would be the webpage is unable to load.

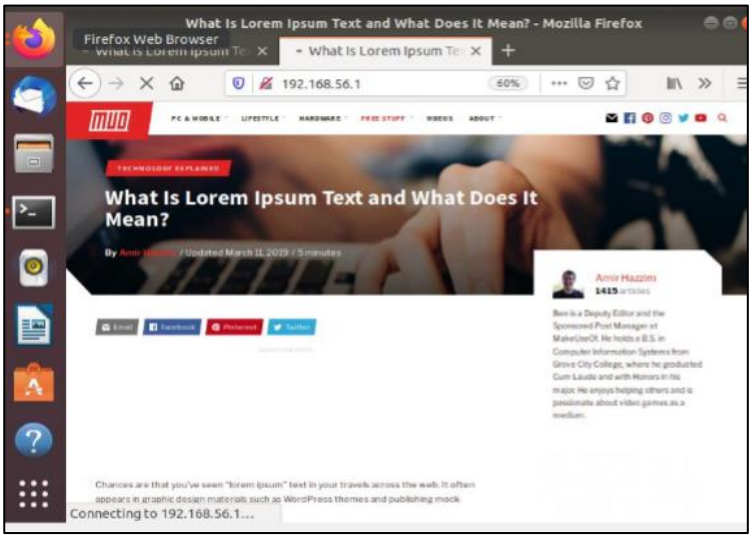

Figure 4. Interface of testing webpage.

Based on Figure 6, the attack is successfully running and flooding the server with multiple request. When the user or the victim browse through the internet and visit the sites, the browser saves several contents and data in temporary storage. This temporary storage is called "cache". To get rid of the fluff,

the cache needs to be cleared from time to time. But in this DoS attack case, the website is not served even after clearing the cache of the browser.

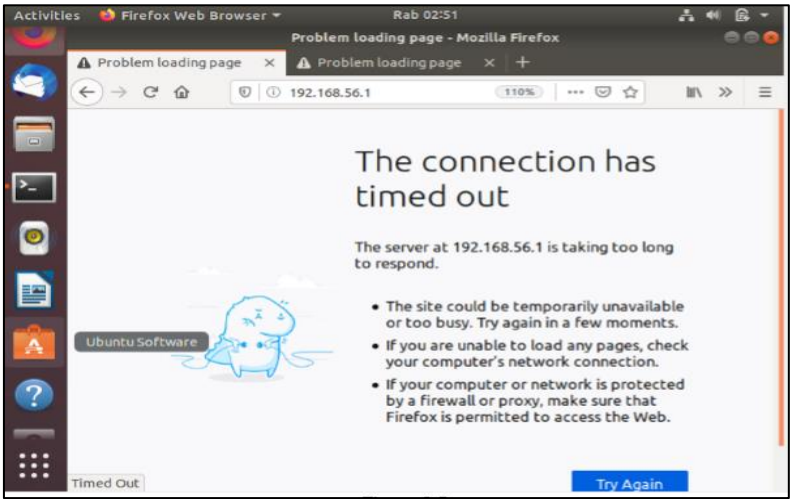

Figure 5. The connection timed out interface.

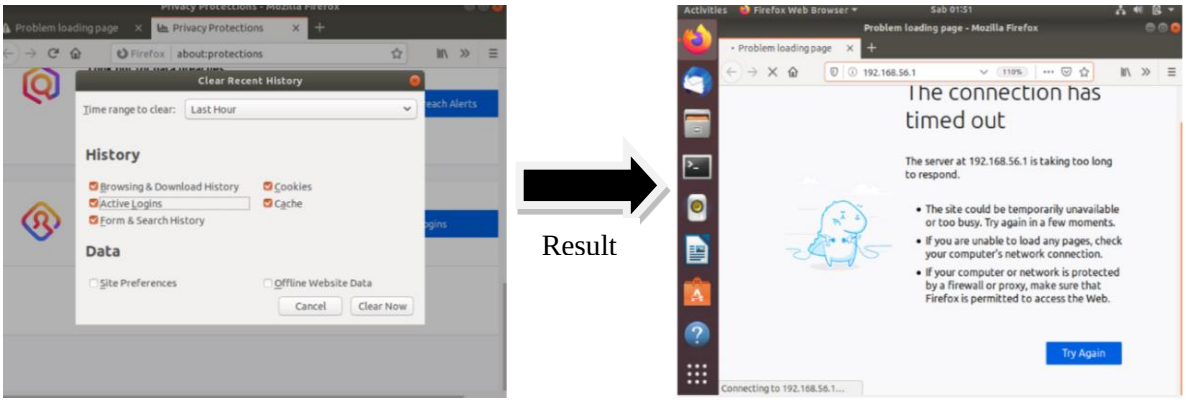

Figure 6. Clearing the cached of the browser and the connection timed out.

6. Conclusions

In order to learn the potential behaviour and help protect the system, designing a type of DoS attack may help to reveal the technique as well as the algorithm. Users may take a measurement of a process and strategies to plan for suitable prevention actions. The invader intends to block the communication channels by DoS attack in order to degrade the estimation performance. Therefore, in this paper, we studied the reliable channel case. In real applications, the communication is likely to be wireless and may be unreliable. Therefore, the simulation is the first step while the unreliable channel case can be left for future work. We do not directly consider the defence and protection of the system in this paper, but aim at pointing out the way DoS attack is implemented, which pave the right direction in providing the protection.

Acknowledgments
We gratefully thank the Faculty of Computing and Multimedia, Kolej Universiti Poly-Tech MARA for providing the opportunity to join ICCEM 2020, Research Management Center (RMC) KUPTM for Micro Grant Support and Dr. Adam Shukri Ali for his proofreading service.

References

[1] H H Jazi, H Gonzalez, N Stakhanova and A A Ghorbani 2017 *Computer Networks* **03** 018
[2] M A Shruthi C V 2018. *International Journal of Scientific Research in Computer Science, Engineering and Information Technology* 175-180
[3] C Yang, W Yang and H Shi 2018 *IET Control Theory & Applications* 1244-1253
[4] A L a A C I Duravkin 2014 *Problems of Infocommunications Science and Technology* 102-106
[5] E Cambiaso, G Papaleo, G Chiola and M Aiello 2015 *Computational Intelligence in Security for Information Systems Conference* (Burgos)
[6] E Cambiaso, G Papaleo and M Aiello 2017 *Journal of Information Security and Applications* 23-31
[7] S Tayama and H Tanaka 2017 *International Conference on Mobile and Wireless Technology* (Kuala Lumpur)
[8] M Aiello, G Papaleo and E Cambiaso 2014 *International Joint Conference SOCO'13-CISIS'13-ICEUTE'13 Advances in Intelligent Systems and Computing* **239**
[9] C C I M d D E C Luis Campo Giralte 2013 *Computers and Electrical Engineering* **39** (7) 2252-2262
[10] Z Cheng, D Yuea, S Hua, C Huanga, Chunxia Doub and Lei Chena 2019 *Electrical Power and Energy System* 105496
[11] M Abushwereb, M Mustafa, M Al-kasassbeh and M Qasaimeh 2020 7