Warsaw University of Technology

FACULTY OF
ELECTRONICS AND INFORMATION TECHNOLOGY

Institute of Computer Science

Bachelor's diploma thesis

in the field of study Computer Science
and specialisation Computer Systems and Networks

Detection and mitigation of Slow HTTP POST attack

Anatolii Do
student record book number 309762

thesis supervisor
mgr inz. Paweł Radziszewski

WARSAW 2024

Detection and mitigation of Slow HTTP POST attack

Abstract. TODO

Contents

1. Introduction

Over the past few decades, the Internet has become a vital part of modern society, revolutionizing the ways we communicate, conduct business, and access information. Its invasive influence spans various sectors: from personal communication and entertainment to political and economic processes. Businesses, in particular, have embraced the power of the Internet to reach global markets, introduce on-line operations, and enhance productivity. Nowadays, it is common for successful enterprises to run a web asset, that constitutes the major source of their profit. The profound reliance on this technology emphasizes once again its undisputable importance for today's world.

However, such heavy dependence on the Internet also brings significant risks. As more critical operations move online, the ground for cyberattacks expands along with it and, as the result, the development of new protection mechanisms has become an increasingly urgent task. One of the most spread cyber threats are Denial of Service (DoS) and its variant Distributed Denial of Service (DDoS). In fact, the first major cyberattack the world has encountered was the DoS attack [1].

Denial of Service is a malicious action to disrupt the normal functioning of a server, service, or network by overwhelming its resources with excessive internet traffic. This overloads the system's resources, making it unable to respond to legitimate user requests. The effects of a successful DoS attack can be serious, as they are aimed on including temporary service outages, prolonged downtime, and significant financial and reputational losses [2]. When DoS attack is initiated from many distributed sources, for example a botnet, such attack is called Distributed Denial of Service (DDoS).

One type of DoS attacks is Slow HTTP POST (a.k.a. RUDY or Slow Body) attack, which is based on the use of the HTTP protocol and are aimed at depleting server resources by sending slow HTTP POST requests. Unlike other types of DoS attacks, Slow HTTP POST attacks do not require a large number of requests or a botnet to achieve their goal. They can be carried out using just a few slow connections, making them particularly dangerous and difficult to detect.

The purpose of this thesis is to study and develop methods to detect and mitigate a particular type of DoS attacks — Slow HTTP POST. This work will discuss the basics of Slow HTTP POST attacks, common defense strategies, including the use of modern techniques. An analysis will be made of the effectiveness of various protection methods and the limitations one may face when using them.

1.1. Description of the work

1.1.1. Motivation

Even though DoS attacks exist since computer networks boom, they are well studied, and many anti-DoS solutions have been made up today (cloud, software, hardware, and even powered by AI), there are still no ideal protection, capable of elimination 100% of the threats. One should remember that any security is a process, not a result. Criminals constantly evolve, finding new vulnerabilities and exploiting them, and the defenders always have to adapt to the

new threats, making the situation to resemble a “Tom and Jerry” animation series.

In 2023, the cybersecurity landscape has witnessed a marked increase in DoS attacks, with a notable rise in the use of application-layer attacks such as Slow HTTP attacks. According to the "ENISA Threat Landscape 2023" report, DoS/DDoS attacks accounted for approximately 20% of all incidents, being the second threat in the chart.

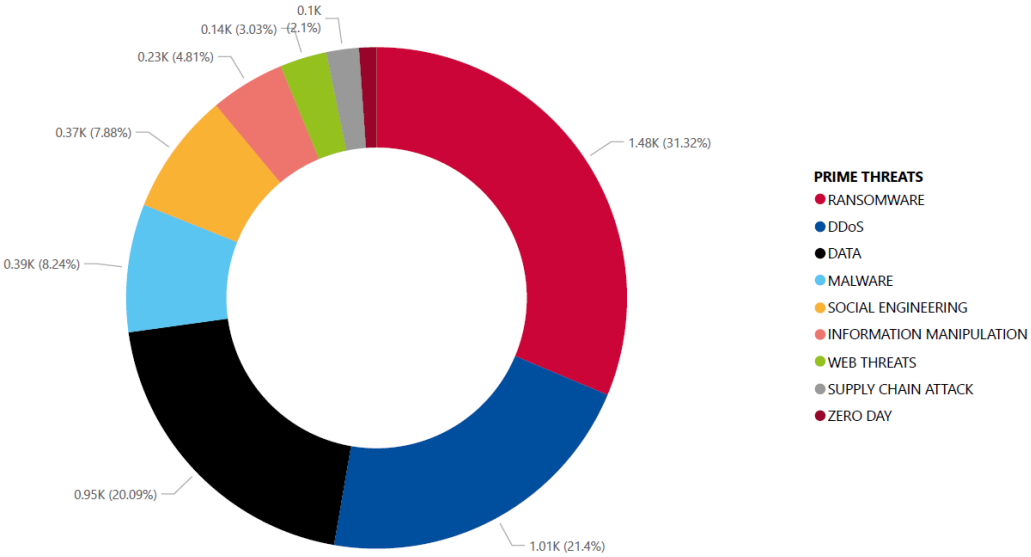

Figure 1.1. Breakdown of analyzed incidents by threat type. ENISA Threat Landscape 2023 [3]

Speaking of the DoS/DDoS attacks aside, they are increasing both in terms of size and complexity, reaching 4.5 million attacks mitigated in Q1 2024 by Cloudflare, 37% of which were HTTP type attacks [3]. The report highlighted that Slow HTTP attacks have become more frequent and sophisticated, exploiting vulnerabilities in web server configurations and the HTTP protocol itself. The report indicated a 25% year-over-year increase in such attacks, emphasizing the urgent need for robust detection and mitigation mechanisms.

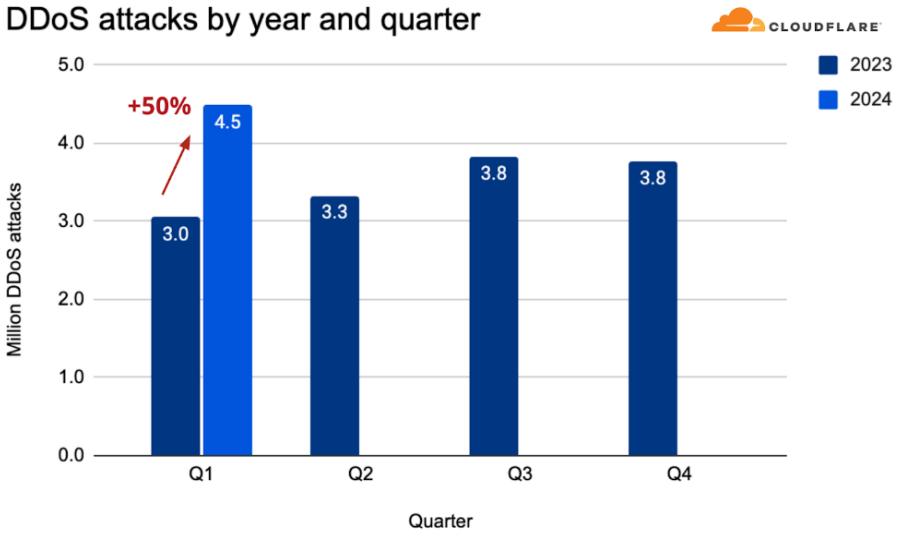

Figure 1.2. DDoS attacks by year and quarter. Source [4].

A particularly dangerous aspect of slow HTTP DoS/DDoS attacks is its ability to merge with legitimate traffic, that is typical to users with slow Internet connection. This feature makes the detection by traditional IDS/IPS (Intrusion Detection/Prevention Systems) a challenging task. This type of attack can be launched from a single computer and does not require many resources from attacker due to the principal of exploit's operation, which will be described in detail in chapter 2 of this work. The vulnerability is easy exploitable, thanks to readily available tools, such as "R.U.D.Y." and "Slowloris", that will be reviewed later. This accessibility increases the potential for these attacks to be carried out by a wide range of adversaries, from 'script-kiddies' to advanced hackers.

General mitigation strategies of slow HTTP POST attacks typically involve adjusting web-server configurations, handling whole incoming traffic, rather than malicious packets only and may worsen the overall performance of the webserver together with user experience, for example, making the service not available for users with slow Internet.

1.1.2. Delimitation

In further, this thesis will focus on using the Apache web server as a primary testing ground for Slow HTTP attacks due to several reasons.

First is the nature of the attack. Apache is a thread-based server, which makes it more vulnerable to Slow HTTP attacks compared to event-based servers like Nginx. In a thread-based architecture, each incoming connection is handled by a separate thread. Slow HTTP attacks take advantage of this by keeping many connections open for a long time, which ties up server resources like memory and CPU. On the other hand, event-based servers use a single-threaded, non-blocking approach to deal with multiple concurrent connections, making them less vulnerable to these types of attacks.

Second, Apache is one of the most widely used web servers globally. As of June 2024, it powers about 30% of all active websites, according to W3Techs [5]. By focusing on Apache, the findings of this research can benefit a large number of websites and organizations, making the research more impactful.

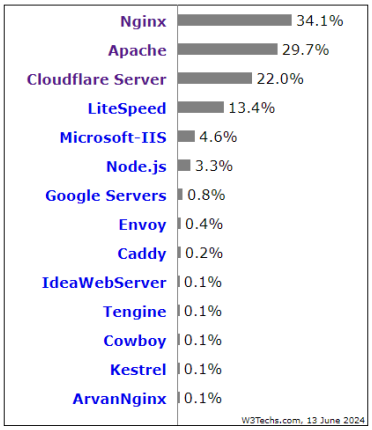

Figure 1.3. Percentages of websites using various web servers. [5]

Third, Apache's open-source nature is a big advantage for research purposes. It allows us to freely modify and configure the server to create various attack scenarios and test different mitigation strategies. In addition, Apache supports adjustable modules security modules. This flexibility is essential for thoroughly understanding Slow HTTP attacks and developing effective solutions.

1.1.3. Goal of the work

Aim of this work is to design and implement a solution that effectively detects and mitigates slow HTTP POST attacks. This will be achieved by focusing on behavior patterns, used by common attacking tools. This involves understanding the nature of how these attacks operate and the signatures they leave behind in web traffic. Other main objectives include:

1) Study common mitigation strategies, that can be done on Apache web server.

2) Conduct comprehensive tests to evaluate the performance of the solution, simulating various attack scenarios without adverse effect on legitimate traffic

3) Analysis of the test results to measure the accuracy, efficiency and reliability of the solution.

1.1.4. Research questions

TODO

1.1.5. Structure of the work

TODO

2. Theoretical background and related works

2.1. Denial of Service

Denial of service attacks are a security threat where an attacker sends a large number of fake requests to a host or server, causing the target host to deny access to authorized users. Consequently, the service from the host becomes unavailable, compromising the system's availability – one of the three fundamental concepts of the information security, dictated by CIA triad[1]. When a denial-of-service attack is launched from multiple computers, it is called to as a Distributed Denial of Service (DDoS) attack, which has become the most used type of DoS attack due to prevalence of botnets, consisted of thousands of compromised hosts.

Due to the wide variety of attacks, classifying them is helpful in clarifying the process of defending against DoS attacks. The classification of DoS attacks is shown in Figure 2.1:

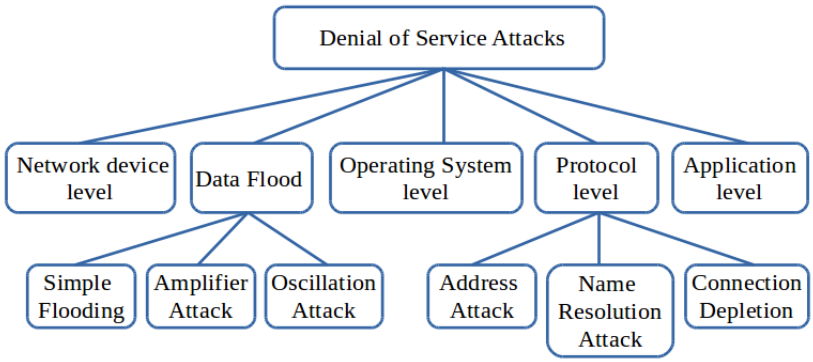

Figure 2.1. The classification of Denial of Service attacks. Source [6]

2.2. Application layer Denial of Service

Application layer DoS attacks target the 7th abstraction layer of the OSI reference model[2], aiming to exhaust specific application resources that work on application-level protocols (such as HTTP, DNS, FTP, SMTP, SSH, etc.). These resources can be CPU, memory or concurrent database connections limit, which are usually significantly smaller than the number of concurrent TCP connections. Because of that, such attacks do not require many resources to initiate, making them accessible to wider range of attackers.

Unlike volumetric attacks, which try to overwhelm the networks with high traffic (flood), application layers attacks use legitimate-looking, low-bandwidth requests to deplete server resources. This stealthy approach makes detection of the attack by traditional anti-DoS/DDoS tools challenging, as they often search for anomalies, such as huge subsequent attempts to establish SYN-ACK handshakes[3]. In case of application-level attacks, it is required to successfully establish the connection before launching them.

Since malicious packets merge withing legitimate-looking traffic, they often bypass

[1] https://www.fortinet.com/resources/cyberglossary/cia-triad
[2] https://en.wikipedia.org/wiki/OSI_model
[3] https://www.geeksforgeeks.org/tcp-3-way-handshake-process/

traditional defense mechanisms like firewalls and intrusion detection systems (IDS). Advanced techniques such as rate limiting and anomaly detection must be employed, which can be resource-intensive, difficult to configure and often have negative impact user performance.

The subject of this research, slow HTTP POST attack, is a type of slow application layer DoS attack.

2.3. Slow HTTP Denial of Service attacks

Slow HTTP Denial of Service (DoS) attacks are a subset of application layer attacks, performed in low-bandwidth, that exploit the HTTP protocol to exhaust server resources. The core principle of these attacks is to hold server connections open for as long as possible, exploiting keep-alive mechanism of HTTP protocol. This is achieved by sending keep-alive HTTP messages in subsequent packet's body, before it reaches keep-alive time-out and opening new connections, to exhaust available resources such as memory or CPU and preventing legitimate users from accessing the service.

Three known types of Slow HTTP DoS were first described and categorized in [7]. According to the research, this type of attack can be classified into four categories:

A. **Slow HTTP Headers (a.k.a. slow HTTP GET or Slowloris).** Named after its execution tool, Slowloris attack firstly establishes connection with the attacked server, then sends HTTP GET requests along with incomplete HTTP Header. The protocol was designed in a way, that before processing the request, it makes server to wait (approximately 26 seconds) [8], for the complete request to deal the with slow networks. The attacker sends next HTTP Header just before this time out, so that the server cannot release the allocated resources and open a new one. Eventually, the attacked server has to maintain many parallel connections and results in resource starvation, causing Denial of Service. The illustration of the attack presented below:

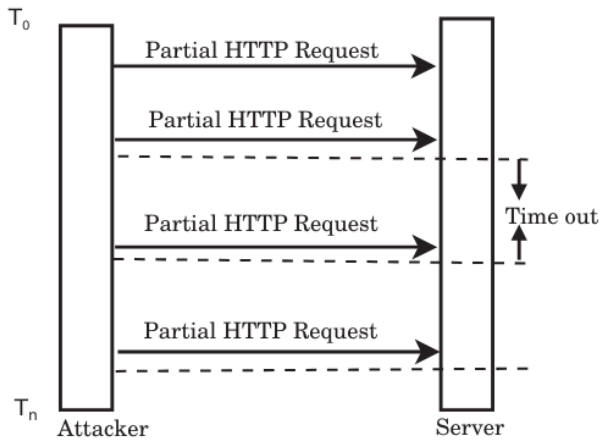

Figure 2.2. Slowloris attack. Source [9]

B. **Slow HTTP Body (a.k.a. slow HTTP POST or RUDY).** Also called after hacking tool, RUDY attack, after establishing a connection with the server, sends complete HTTP Header, but HTTP body has large Content-Length value, so that server must wait (around 300 seconds) [8] until it receives the entire body, before closing the connection. The content is being sent in small 1-byte chunks within POST requests at very slow rate, as if it was sent from a host with slow Internet connection. This results in several long simultaneous connections, tying up server resources and making them unavailable for legitimate users.

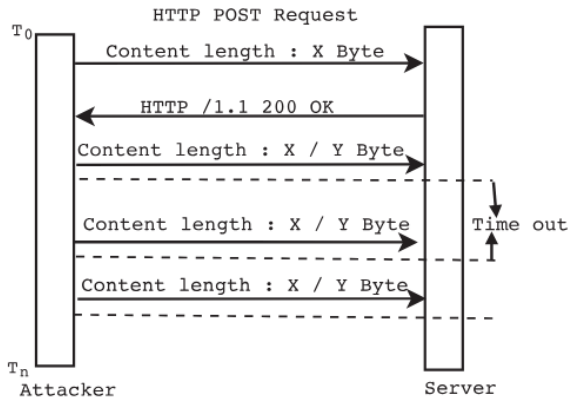

Figure 2.3. Slow HTTP POST attack. Source [9]

C. **Slow Read attack.**
TODO

D. **Apache Killer (Range) attack.**
TODO

2.4. Slow HTTP POST attack execution

TODO

Bibliography

[1] https://www.fbi.gov/news/stories/morris-worm-30-years-since-first-major-attack-on-internet-110218
[2] https://www.kaspersky.com/about/press-releases/2015_collateral-damage-26-of-ddos-attacks-lead-to-data-loss
[3] https://www.enisa.europa.eu/publications/enisa-threat-landscape-2023
[4] https://blog.cloudflare.com/ddos-threat-report-for-2024-q1
[5] https://w3techs.com/technologies/overview/web_server
[6] https://www.researchgate.net/publication/323194627_A_Review_of_Defense_Against_Slow_HTTP_Attack
[7] https://www.researchgate.net/publication/242654051_Slow_DoS_Attacks_definition_and_categorisation
[8] https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=7784605
[9] https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8300082

List of Figures

TODO

List of Tables

TODO