FH University of Applied Sciences TECHNIKUM WIEN

Proposal

Degree Program BIF

Course Scientific Writing

Vulnerabiltiy Analysis of the SMB protocol implementation on Windows Server

By: Paul Bogza
Student number: if22b078

Advisor: FH-Prof. DI Alexander Mense

Vienna, 02.12.2024

Introduction/Background

The Server Message Block (SMB) protocol is a network file-sharing protocol enabling applications on a computer to read, write, and manage files, as well as request services from server programs within a network. SMB operates over TCP/IP protocol or other network protocols. Through SMB, an application (or its user) can access files and resources located on a remote server, facilitating the reading, creation, and updating of files on that server. Additionally, SMB can interface with any server program configured to handle SMB client requests. It serves as a fabric protocol integral to Software-defined Data Center (SDDC) technologies, including Storage Spaces Direct and Storage Replica. [7].

The first version of Server Message Block, SMB version 1, was released by Microsoft in 1996, and since then has been present on every windows system up until 2017. As of 2017, since the “Windows 10 Fall Creators Update”, SMB version 1, is no longer installed by default on windows systems with the version number 1709 or higher and can be uninstalled manually from existing systems [8], it has been superseded by SMB version 2.

SMB version 1 has been the target of many cyberattacks over the past decade [10]. The most recent and notable of these attacks was the WannaCry ransomware, which leveraged the MS17-010 vulnerability [2]. MS17-010 allows an unauthenticated attacker to execute arbitrary code remotely on a target sever which is running SMB version 1, the attacker can achieve this by sending a specially crafted packet to the server [3]. The exploit for the MS17-010 vulnerability was originally developed by the National Security Agency (NSA) and given the name “Eternal Blue” [9].

This type of ransomware attack can damage millions of computer systems and cause millions in financial damage to affected companies [11] if there are no countermeasures in place. Malware creation tools, which are an emerging phenomenon in dark web markets [6], facilitate the potential for more such attacks to be carried out in shorter periods of time. There are many papers that examine how this type of malware functions, such as [5], but not enough show how the attackers find these exploits in the first place. This paper will delve into the technical analysis of the MS17-010 vulnerability and how an attacker may approach finding and exploiting it. The analysis will make use of static, dynamic, and reverse engineering methodologies since these are the most commonly used methods for finding such vulnerabilities and analyzing malware [12].

Research question

How do attackers approach finding vulnerabilities in protocols such as Server Messaging Block (SMB) and what counter measures can be taken to prevent it?

This paper aims to help defenders gain a deeper understanding of how attackers approach a target system, what methodologies and techniques they use to find and exploit software vulnerabilities, as well as create a Proof-of-Concept of what such an attack or exploit might look like. Ultimately this can be beneficial for mitigating or preventing such attacks from being carried out on critical infrastructure.

Methods

The methods that will be used in this paper encompass static and dynamic analysis of software, as well as reverse engineering. Static analysis means that code is not being executed in order to be analysed. Instead, it examines the software for signs of malicious behavior. There are technical inidicators for such behavior, such as the file name, strings contained in the file such as IP addresses or domains, or the file header data. In addition to this, disassemblers or network analyzers can be used to inspect the behavior and collect information. For dynamic analysis one executes the software in a safe environment in order to observe its behavior and gain a deeper understanding and comprehend the intention of the software without the risk of damaging or infecting an important system or network. Reverse engineering is the practice of using debuggers or disassemblers in order to decrypt encrypted data or analyze the logic of a given software and find hidden capabilities which were not previously displayed in the static or dynamic analysis [12].

These methods were chosen because they are the most promising and common ways of analyzing software, they are well-documented and there are a plethora of tools to aid in ones pursuit of understanding [12].

The expected results of this paper are a Proof-of-Concept which demonstrates a potential exploit for a software vulnerability in SMB version 1, as well as a penetration testing report of how an attacker might approach compromising a target system.

References

[1] D. -Y. KAO, S. -C. HSIAO and R. TSO, "Analyzing WannaCry Ransomware Considering the Weapons and Exploits," *2019 21st International Conference on Advanced Communication Technology (ICACT)*, PyeongChang, Korea (South), 2019, pp. 1098-1107, doi: 10.23919/ICACT.2019.8702049.

[2] M. Fujimoto, W. Matsuda and T. Mitsunaga, "Detecting attacks leveraging vulnerabilities fixed in MS17-010 from Event Log," *2019 IEEE Conference on Application, Information and Network Security (AINS)*, Pulau Pinang, Malaysia, 2019, pp. 42-47, doi: 10.1109/AINS47559.2019.8968703.

[3] *Microsoft Security Bulletin MS17-010 - Critical*, https://learn.microsoft.com/en-us/security-updates/securitybulletins/2017/ ms17-010

[4] M. Aljaidi *et al*., "NHS WannaCry Ransomware Attack: Technical Explanation of The Vulnerability, Exploitation, and Countermeasures," *2022 International Engineering Conference on Electrical, Energy, and Artificial Intelligence (EICEEAI)*, Zarqa, Jordan, 2022, pp. 1-6, doi: 10.1109/EICEEAI56378.2022.10050485.

[5] B. Fiore, K. Ha, L. Huynh, J. Falcon, R. Vendiola and Y. Li, "Security Analysis of Ransomware: A Deep Dive into WannaCry and Locky," *2023 IEEE 13th Annual Computing and Communication Workshop and Conference (CCWC)*, Las Vegas, NV, USA, 2023, pp. 285-294, doi: 10.1109/CCWC57344.2023.10099114.

[6] R. A. Awad and K. D. Sayre, "Automatic Clustering of Malware Variants", *2016 IEEE Conference on Intelligence and Security Informatics (ISI 2016)*, pp. 298-303, 2016.

[7] Overview of file sharing using the SMB 3 protocol in Windows Server, https://learn.microsoft.com/en-us/windows-server/storage/file-server/file-server-smb-overview

[8] SMBv1 is not installed by default in Windows 10 version 1709, Windows Server version 1709 and later versions, https://learn.microsoft.com/en-us/windows-server/storage/file-server/troubleshoot/smbv1-not-installed-by-default-in-windows

[9] E. Nakashima and C. Timberg, „NSA officials worried about the day its potent hacking tool would get loose. Then it did.“, May 16 2017, https://www.washingtonpost.com/business/technology/nsa-officials-worried-about-the-day-its-potent-hacking-tool-would-get-loose-then-it-did/2017/05/16/50670b16-3978-11e7-a058-ddbb23c75d82_story.html

[10] CVE Mitre, https://cve.mitre.org/cgi-bin/cvekey.cgi?keyword=smbv1

[11] R. Singh, S. Singh and A. Singh, "REvil Ransomware," *2024 International Conference on Innovations and Challenges in Emerging Technologies (ICICET)*, Nagpur, India, 2024, pp. 1-6, doi: 10.1109/ICICET59348.2024.10616279.

[12] Malware Analysis, K. Baker, April 17 2023, https://www.crowdstrike.com/en-us/cybersecurity-101/malware/malware-analysis/