Jardel Thomas
Security Engineer
UBNetDef
Thursday, April 25th, 2024

1.0 Executive Summary

The recent penetration test conducted on the ServerNet IP range, particularly focusing on IP address 10.43.13.193, uncovered critical vulnerabilities, including flaws in essential services like MySQL Database, OpenSSH Server, and Netkit RSHD. These vulnerabilities pose significant security risks, such as data manipulation and unauthorized access. This report aims to concisely analyze the findings, their potential consequences, and recommended mitigation strategies. Additionally, it offers a prioritized list of findings categorized by priority and recommended actions to bolster the system's security resilience.

Priority	Category	Suggested Mitigation
HIGH	MySQL Database	Prioritize reviewing and tightening user permissions
HIGH	OpenSSH Sever	Update OpenSSH server to the latest patched version.

1.1.1 *Recommendations*

Two urgent cybersecurity measures should be taken without delay. Firstly, enhance MySQL database access controls by restricting administrative privileges to authorized individuals and regularly monitoring user activity to prevent unauthorized access or privilege escalation. Secondly, update the OpenSSH server promptly to the latest patched version to minimize the risk of unauthorized access and address known vulnerabilities. Establishing a regular update schedule ensures the server remains secure against potential exploits.

Contents

1.0 Executive Summary 2

1.1.1 Recommendations 2

Table of Contents 3

2.0 Scope and Methodology 4

2.1 Scope 4

2.2 Methodology 4

3.0 Findings 5

3.1 MySQL Database 5

3.2 OpenSSH 6

4.0 Works Cited 7

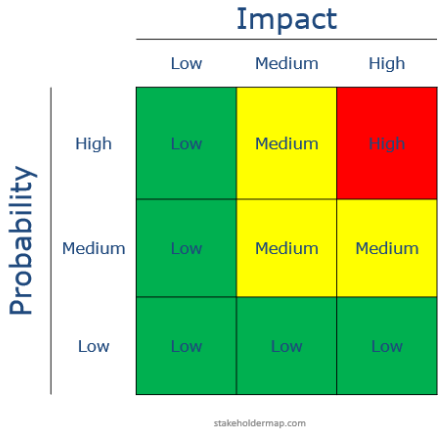
2.0 Scope and Methodology

2.1 Scope

The penetration test was conducted between 03/22 and 03/25 by our group to evaluate the security posture of the target system. The assessment covered devices within the ServerNet IP range, specifically targeting the IP address 10.43.13.193. Critical services discovered during the test included Netkit RSHD, OpenSSH Server, and MySQL Database Server. These services could potentially be exploited if not properly secured. The test was conducted as a White Box test, providing insights into the system's vulnerabilities and guiding prioritized remediation efforts.

2.2 Methodology

We employed a method that combined the assessment of impact and likelihood to score vulnerabilities found during the penetration test, enabling a structured evaluation of each vulnerability's severity. This strategy is illustrated in the accompanying risk matrix, which has low, medium, and high levels of impact and likelihood.



2.2.1 *Tools used.*

- Nmap
- **Burp Suite**

3.0 Findings

3.1 MySQl Database Server

Likelihood: High, Impact: High

The vulnerability in MySQL privilege escalation exposes a critical flaw in the database server's access control mechanisms, enabling unauthorized users to elevate their privileges beyond their designated level of access. This loophole may result from misconfigured user permissions or Database configuration, allowing attackers to bypass authentication and gain administrative privileges. With elevated access, malicious actors can execute administrative commands, manipulate data, and tamper with database configurations, posing a significant threat to the system's confidentiality, integrity, and availability. The potential consequences of this vulnerability include unauthorized data access, system compromise, financial losses, and reputational damage to the organization, underscoring the urgency of addressing and mitigating this risk.

To mitigate this vulnerability effectively, organizations must prioritize reviewing and tightening user permissions and access controls within the MySQL database server. Employing the principle of least privilege ensures that users are granted only the minimum level of access necessary for their tasks, reducing the likelihood of unauthorized privilege escalation. Regular auditing and monitoring of user activities and database transactions are essential for detecting and preventing unauthorized attempts to elevate privileges. Implementing additional security measures such as multi-factor authentication, encryption, and network segmentation can further bolster defenses against privilege escalation attacks. Furthermore, staying proactive with software updates and security patches is crucial to addressing any known vulnerabilities and fortifying the overall security posture of the system against evolving threats.

3.2 OpenSSH Server

Likelihood: High, Impact: High

The OpenSSH server was found to be vulnerable due to the use of an outdated version, which makes it vulnerable to known security flaws like privilege escalation and remote code execution exploits. By taking advantage of these flaws, attackers might be able to enter the system without authorization, execute malicious commands, and jeopardize the availability and integrity of the data. Using network scanning and enumeration techniques, the outdated version of the OpenSSH server was found during the penetration test. Research then confirmed the vulnerabilities that were linked to it.

Update the OpenSSH server to the most recent patched version from the official vendor repository right away to help mitigate this vulnerability. For all software and systems to receive security updates on time, a strong patch management procedure must be established. Furthermore, putting intrusion detection systems, access controls, and network segmentation into place can help identify and stop unauthorized access attempts, lowering the chance of exploitation. To strengthen the system's overall security posture and proactively find and fix any new or emerging vulnerabilities, regular security audits and penetration tests should be carried out.

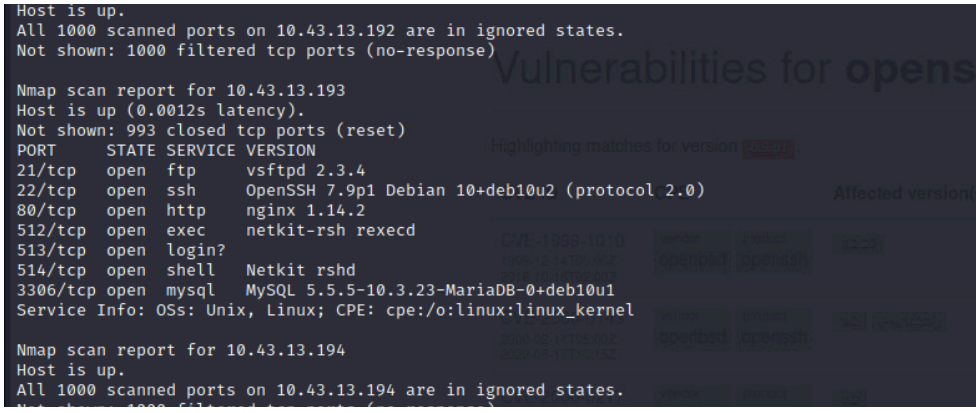


Figure 1: Nmap scan of Target device

4.0 Works Cited

K, P. (2023, November 8). *Privilege Escalation Vulnerability in MySQL / MariaDB / PerconaDB databases (CVE-2016-5616 / CVE-2016-6663)*. SecPod Blog. https://www.secpod.com/blog/privilege-escalation-vulnerability-in-mysql-mariadb-perconadb-databases/

https://www.cybersecurity-help.cz/vdb/openssh/openssh/7.9p1/