Penetration Testing Report

8/17/2024

Report For:

CCS4350 – Ethical Hacking

Prepared by: **Illuminati**

21UG0886-A.I.D. Fernando

21UG0858-S.A.A.D. Karunathilaka

21UG0887-N.V.P.Y. Nahalla

21UG0132-Deshsan Jayawardhana

21UG0776-J.A.N.A.Jayakody

Table of Contents

1. Introduction

1.1 Overview

This penetration testing report presents the results of a security assessment conducted on a vulnerable Ubuntu 12.04 LTS virtual machine (VM). The primary objective of this assessment was to identify and exploit vulnerabilities, providing a comprehensive analysis and recommendations for improving the security posture.

1.2 Scope and Objectives

The scope of the penetration test was limited to the provided VM image. The objectives were to:

- Identify live hosts, open ports, and services running on the VM.
- Analyze the services for known vulnerabilities.
- Exploit identified vulnerabilities to gain unauthorized access.
- Document evidence of exploitation and provide recommendations for mitigation.

1.3 Tools and Methodologies

Tools used during the penetration test included:

- **Reconnaissance:** whois, dig, theHarvester
- **Scanning and Enumeration:** Nmap, enum4linux, Nikto
- **Vulnerability Analysis:** Nmap scripts, searchsploit
- **Exploitation:** Metasploit, msfvenom
- **Proof of Hacking:** Screenshots, command outputs, session logs

2. Reconnaissance

2.1 Findings

- **Live Hosts:** Detected the target VM at IP address 192.168.244.1
- **Open Ports:** No open ports were identified on the VM.

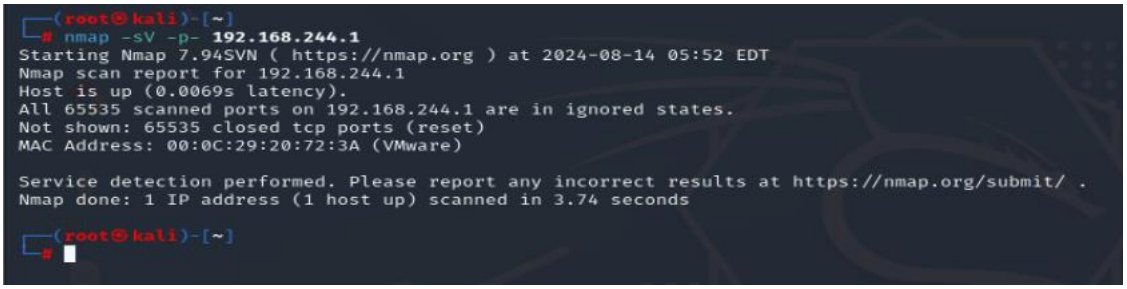
2.2 Evidence

- **Screenshot of Reconnaissance :**

3. Scanning and Enumeration

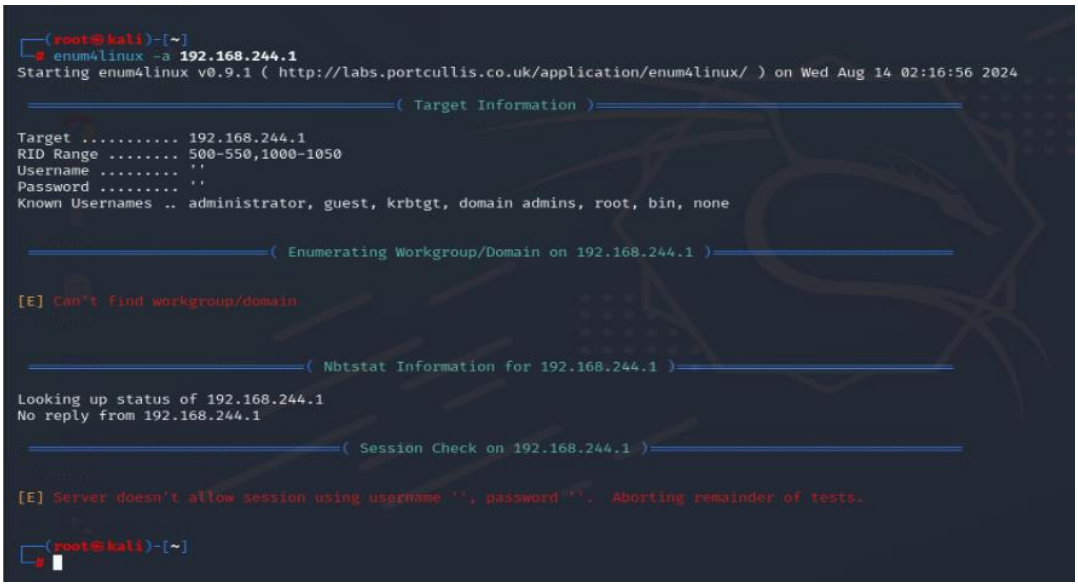
3.1 Network Scanning

- **Objective:** Perform network scanning to identify potential vulnerabilities in the target VM.
- **Results:** No vulnerabilities or open ports were identified during the network scan.
- **Evidence:**



3.2 User Enumeration

- **Objective:** Enumerate users, shares, and other network resources to identify potential points of access.
- **Results:** No user or resource information was successfully enumerated.
- **Evidence:**



3.3 Web Server Vulnerability Scanning

- **Objective:** Scan the web server for potential vulnerabilities.
- **Results:** No vulnerabilities were found during the Nikto scan.
- **Evidence:**

4. Exploitation

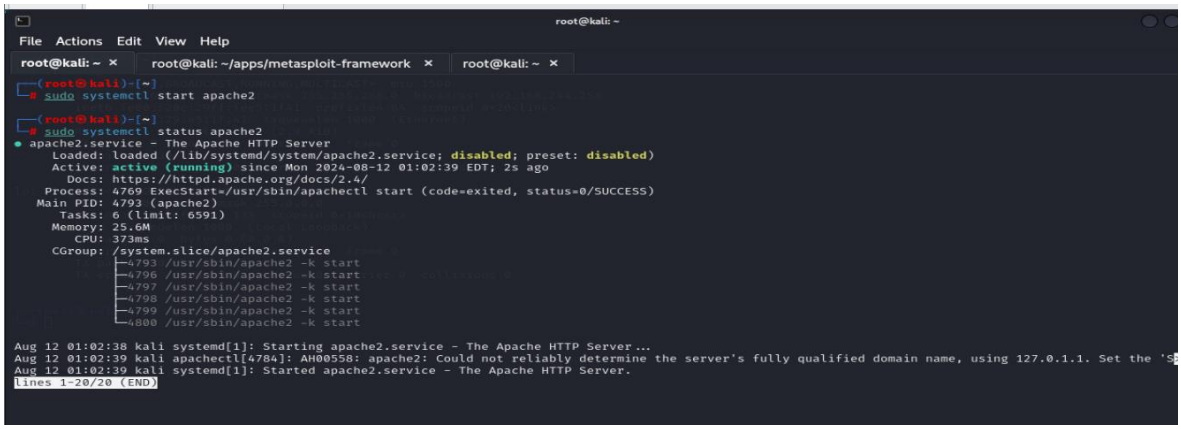
4.1 Exploitation Process

- **Vulnerability:** Apache HTTP Server directory traversal vulnerability.
- **Exploit Used:** Metasploit module multi/handler and msfvenom.
- **Steps Taken:**
    1. **Created Payload:**
        - **Payload:** linux/64/meterpreter/reverse_tcp
        - **Command Used:**

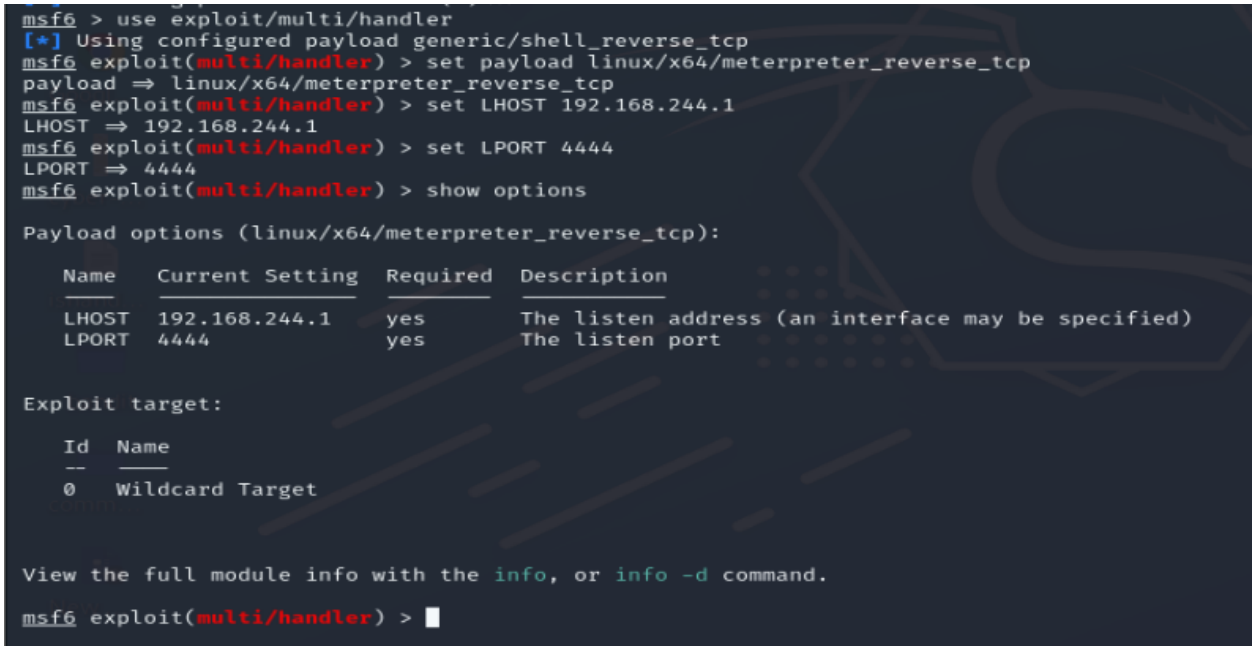
        - **Description:** Generated a trojan payload (trojan.elf) that creates a reverse TCP connection back to the attacker's machine.
    2. **Deployed the Payload:**
        - **Moved Payload to Web Directory:**
            - **Command Used:**

        - **Description:** The trojan payload was moved to the Apache web server's root directory to make it accessible via HTTP.

3. **Started the Apache Server:**
    - **Command Used:**

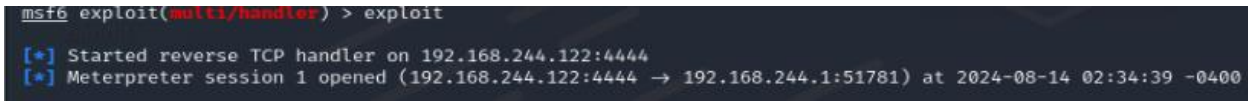


    - **Description:** Started the Apache server to host the trojan payload.
4. **Configured Metasploit Handler:**
    - **Module Used:** multi/handler
    - **Configuration:** Configured the Metasploit multi/handler to handle incoming connections from the trojan payload.
    - **Commands Used:**

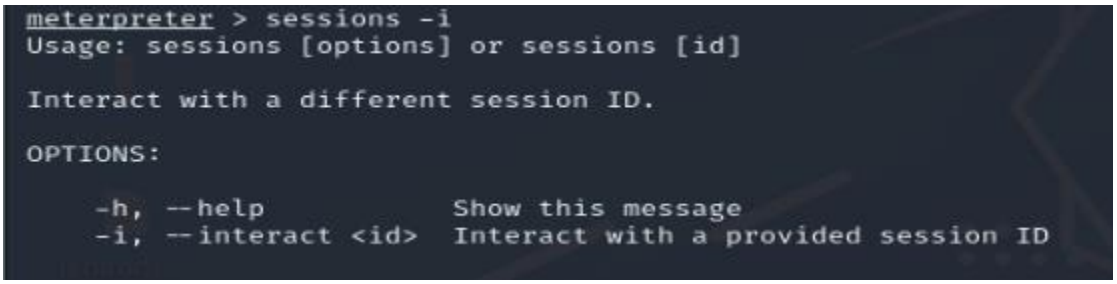


4.2 Evidence

- **Metasploit Output:**



- **Session Logs:**



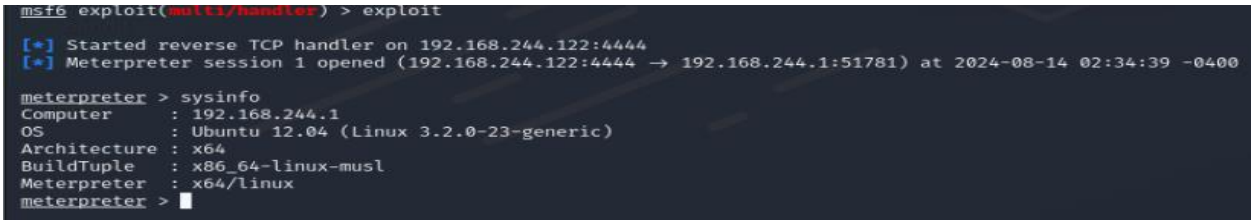
4.3 Tools Used

- **Metasploit:** Utilized to handle the reverse connection and manage the payload.
- **Msfvenom:** Used to create the trojan payload.
- **Apache:** Used to host the payload file to facilitate the attack.

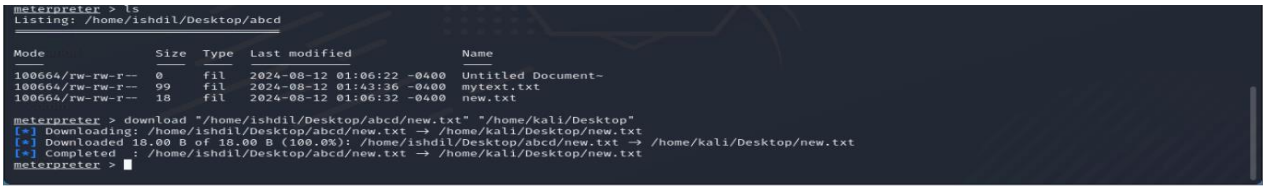
5. Proof of Hacking

5.1 Evidence of Successful Exploitation.

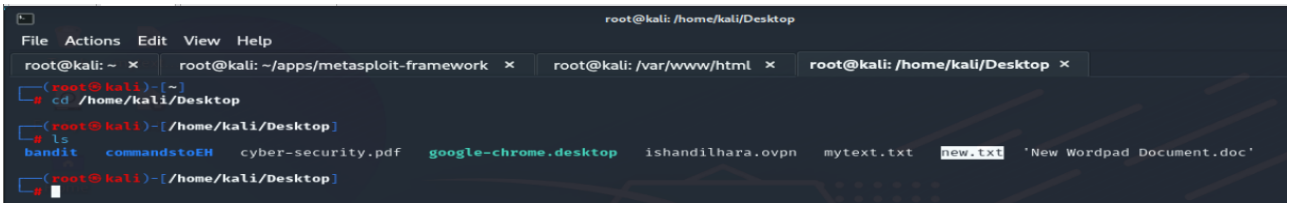
- **System Information Retrieval:**



- **Downloading a File:**



- **Uploading a File:**



5.2 Summary of Findings

- **Proof of Access:** The successful establishment of a Meterpreter session allowed access to the target system. Key actions performed include retrieving system information and manipulating files on the remote system.
- **Sensitive Information Access:** Demonstrated access by showing the sysinfo command output and files downloaded/uploaded.
- **Additional Evidence:** The screenshots illustrate successful file transfer operations and system information retrieval, confirming the effectiveness of the exploitation.

6. Incident Response and Mitigation

This section describes the steps to handle and address the security incident, focusing on detecting, analyzing, containing, and removing the threat. It also provides recommendations for improving the security posture to prevent future incidents.

6.1 Incident Response

a. Detection:

- **Identify the Breach:**
    - **Indicators:** Monitor for unusual network traffic, unauthorized access logs, and alerts from intrusion detection systems (IDS).
    - **Tools:** Utilize monitoring tools and conduct log analysis to identify anomalies.
    - **Detection Methods:**
        - **Log Analysis:** Examine system and application logs for signs of unauthorized access.
        - **Network Monitoring:** Look for unusual patterns or unexpected connections.

b. Analysis:

- **Determine the Scope of the Attack:**
    - **Affected Systems:** Identify which systems were compromised.
    - **Attack Vector:** Understand how the attacker gained access, whether through a vulnerability or social engineering.
    - **Analysis Techniques:**
        - **Forensic Analysis:** Investigate file changes, system modifications, and command histories.
        - **Vulnerability Assessment:** Re-assess the system to identify additional vulnerabilities.

c. Containment:

- **Isolate the Affected System:**
    - **Immediate Actions:** Disconnect the compromised system from the network to prevent further damage.
    - **Tools:** Employ network segmentation and firewall rules to limit access.
    - **Containment Measures:**
        - **Change Credentials:** Reset passwords for compromised accounts.
        - **Block Malicious IPs:** Update firewall rules to block IP addresses associated with the attack.

d. Eradication:

- **Remove the Threat:**
    - **Clean the System:** Eliminate malicious files, backdoors, and trojans from the affected system.
    - **Patch Vulnerabilities:** Apply patches and updates to fix the exploited vulnerabilities.
    - **Eradication Steps:**
        - **Run Antivirus Scans:** Use antivirus software to detect and remove malware.
        - **Update Software:** Ensure all software and services are updated with the latest security patches.

6.2 Mitigation Recommendations

a. Immediate Recommendations:

- **Review and Strengthen Security Policies:**
    - **Password Policies:** Implement robust password policies and multi-factor authentication (MFA).
    - **Access Controls:** Review and limit user permissions based on the principle of least privilege.
    - **Recommendations:**
        - **Audit User Accounts:** Regularly review user accounts and access levels.
        - **Enhance Network Security:** Deploy intrusion prevention systems (IPS) and update firewall rules regularly.

b. Long-Term Recommendations:

- **Implement Regular Security Audits:**
    - **Schedule Audits:** Conduct regular security assessments to uncover and address vulnerabilities.
    - **Penetration Testing:** Perform periodic penetration tests to evaluate defenses and simulate potential attacks.
    - **Long-Term Measures:**
        - **Security Awareness Training:** Educate users on security best practices and phishing prevention.
        - **Incident Response Plan:** Develop and update an incident response plan and conduct regular drills to ensure preparedness.

7. Conclusion

In this section, summarize the findings from the penetration test, the overall security posture of the environment, and provide actionable recommendations for mitigating the identified vulnerabilities.

7.1 Summary of Findings

- **Overview:** The penetration test conducted on the target VM revealed critical security weaknesses and vulnerabilities. Despite the lack of open ports, a successful exploitation was achieved using a Trojan, demonstrating the presence of exploitable issues.
- **Key Findings:**
    - **Vulnerability Identified:** The test exploited an Apache HTTP Server directory traversal vulnerability, allowing unauthorized access to the system.
    - **Exploitation Success:** The creation and deployment of a Trojan file led to a successful Meterpreter session, indicating that the system was compromised.

7.2 Overall Security Posture

- **Current Security Status:** The compromised VM's security posture was found to be inadequate due to the presence of exploitable vulnerabilities and insufficient defensive measures.
- **Impact:** The ability to exploit the system highlights significant gaps in the current security measures, necessitating immediate remediation and long-term improvements.

7.3 Recommendations

- **Immediate Actions:**
    - **Patch Vulnerabilities:** Address the identified vulnerabilities by applying relevant patches and updates.
    - **Enhance Monitoring:** Improve system and network monitoring to detect and respond to potential threats more effectively.
- **Long-Term Improvements:**
    - **Regular Security Audits:** Schedule and perform regular security assessments and penetration tests to identify and address new vulnerabilities.
    - **Training and Awareness:** Implement security training programs to increase awareness among users and reduce the risk of social engineering attacks.
    - **Incident Response Plan:** Develop a comprehensive incident response plan and conduct regular drills to ensure readiness for future incidents.

8. Appendices

8.1 Additional Information, Scripts, and Tools Used During the Test

- **Scripts Used:**

1. Trojan Creation Script

- **Description:** This script generates a Trojan executable using msfvenom and places it in the Apache HTTP Server's directory for serving.
- **Purpose:** Used to create and deploy a Trojan payload that connects back to the attacker's machine.
- **Used tools:** msfvenom ,Apache HTTP Server

2. Reconnaissance Script

- **Description:** Gathers information about the target VM through various reconnaissance tools.
- **Purpose:** To collect information such as domain details and email addresses.
- **Used tools:** whois,dig,theHarvester

3. Network Scanning Script

- **Description:** Uses Nmap to perform a comprehensive scan of the target IP address.
- **Purpose:** To discover open ports and services on the target system.
- **Used tools:** Nmap

4. Exploitation Script

- **Description:** Uses Metasploit to exploit known vulnerabilities in the target system.
- **Purpose:** To execute the exploit and gain access to the target system.
- **Used tools:** Metasploit Framework

5. Post-Exploitation Script

- **Description:** Manages the Meterpreter session and performs post-exploitation tasks.
- **Purpose:** To list and manage active sessions after exploitation.

8.2 Team Information

- **Team Name:** Illuminati
- **Team Members:**
    - **Member 1:** 21UG0886-A.I.D. Fernando
    - **Member 2:** 21UG0858-S.A.A.D. Karunathilaka
    - **Member 3:** 21UG0887-N.V.P.Y. Nahalla
    - **Member 4:** 21UG0132-Deshsan Jayawardhana
    - **Member 5:** 21UG0776-J.A.N.A.Jayakody
- **Contribution of Each Member:**
    - All team members contributed equally, with each responsible for 20% of the overall work.