1. Executive Summary

This report documents the Vulnerability Assessment conducted on a deliberately vulnerable virtual machine (Metasploitable 3) using industry-standard security tools. The objective of this assessment was to identify security weaknesses, evaluate associated risks, and recommend remediation measures. Multiple high and medium severity vulnerabilities were identified, primarily due to outdated services, insecure configurations, and missing security controls. If exploited, these issues could allow unauthorized access, privilege escalation, and potential compromise of sensitive data.

2. Scope and Objectives

2.1 Scope

- Target System: Metasploitable 3 (Vulnerable VM)
- Environment: Local virtualized lab setup
- Assessment Type: Vulnerability Assessment (No exploitation beyond validation)

2.2 Objectives

- Identify open ports and running services
- Detect known vulnerabilities (CVEs)
- Assess risk using CVSS and risk matrix
- Provide remediation recommendations

3. Testing Environment Setup

3.1 Tools and Platforms

- Host OS: Windows / Linux
- Virtualization Tool: Oracle VirtualBox
- Attacker Machine: Kali Linux
- Target Machine: Metasploitable 3

3.2 Setup Steps

- Installed Kali Linux as the attacker machine.
- Downloaded Metasploitable 3 from GitHub.
- Configured both VMs in VirtualBox using Host-Only / Internal Network mode.
- Verified network connectivity between Kali Linux and Metasploitable 3.

3.3 Installation of Metasploitable 3 in windows

- Open PowerShell.
- mkdir metasploitable3-workspace.
- cd metasploitable3-workspace
- Invoke-WebRequest-Uri "https://raw.githubusercontent.com/rapid7/metasploitable3/master/Vagrantfile" -OutFile "Vagrantfile"
- vagrant up

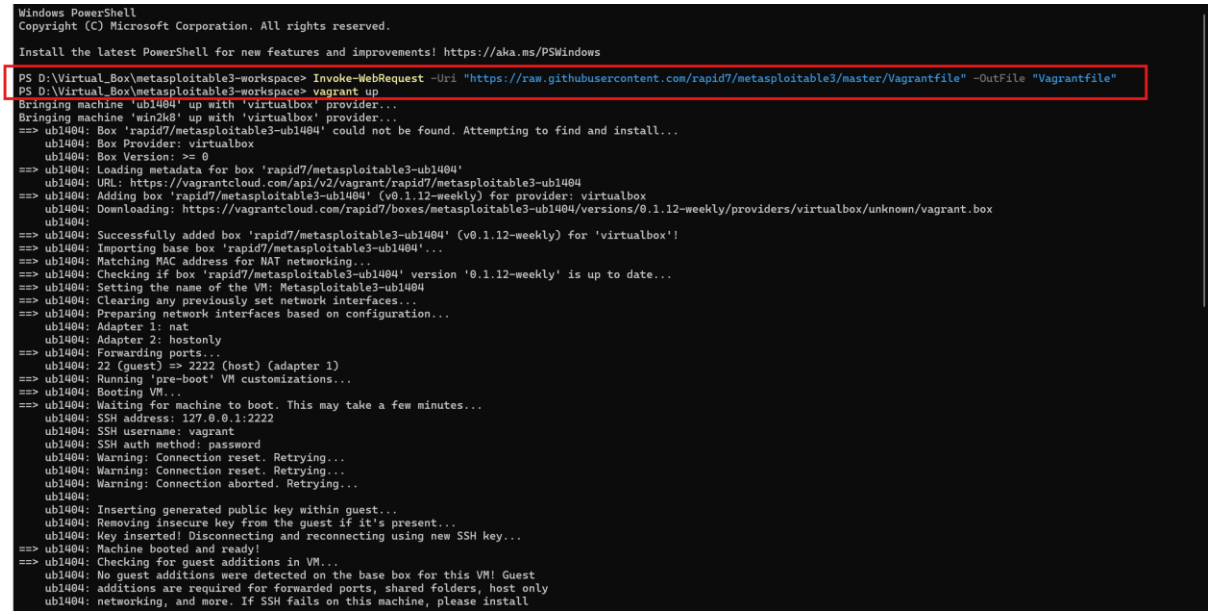

Figure 1: Installation of metasploitable 3

4. Methodology

The assessment followed a standard vulnerability assessment methodology:

- Information Gathering
- Vulnerability Scanning
- Risk Assessment
- Documentation and Reporting

5. Vulnerability Scanning

5.1 Tools Used

- OpenVAS (Greenbone Vulnerability Manager)
- Nmap
- Nikto Web Server Scanner

5.2 Scanning Procedure

OpenVAS:

- Service started using the command:
  sudo gvm-start
- Target IP address of Metasploitable 3 was scanned.
- Vulnerabilities were analyzed based on CVSS score, severity, and CVE references.

Nikto:

- Nikto was used to identify web server misconfigurations and outdated components.
- Command: **nikto -h 192.168.29.13 -o nikto_report.html**

Figure 2: Nikto Scan

Nikto Scan: https://drive.google.com/file/d/1RDTQtMtfJWCqJg41e0w5kevO9_9h0Cov/view

Nmap:

- Performing network scans on a Metasploitable VM.
- **Command:** nmap -sV 192.168.29.13 -T4
  -sV = Identifies service version number
  -T4 = Faster scan speed (aggressive timing)

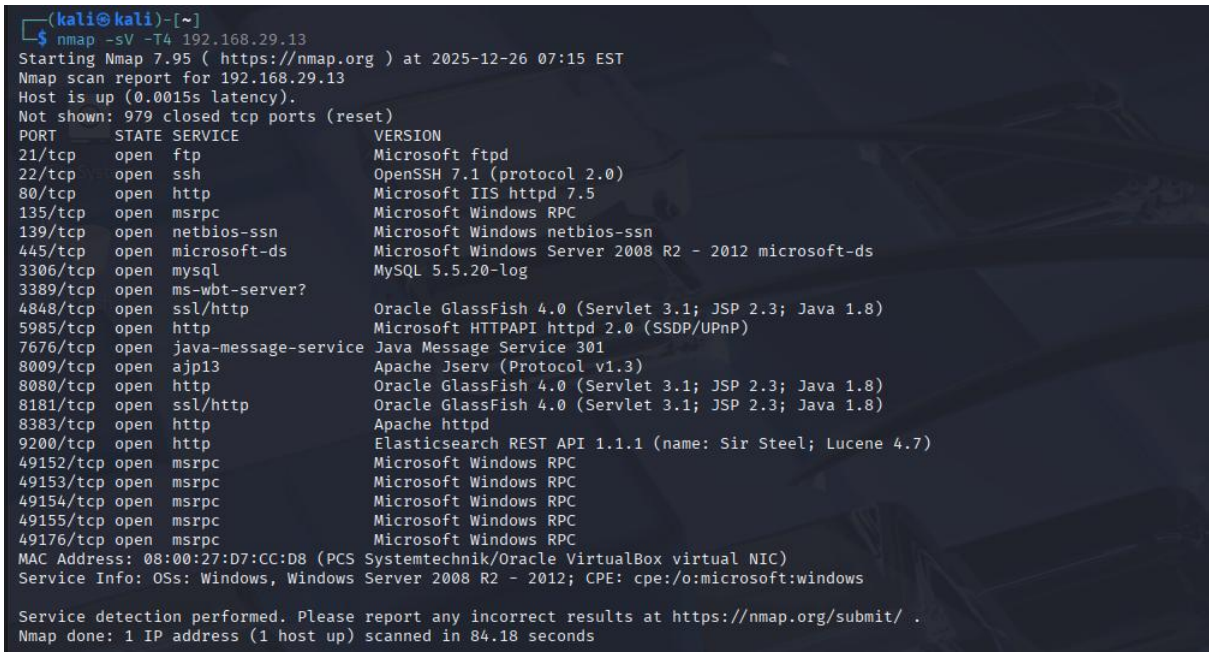

Figure 3: Nmap scan

6. Findings and Analysis

6.1 Open Ports and Services

Port	Service	Description
21	FTP	Microsoft FTP service (ftpd)
22	SSH	Secure Shell service (OpenSSH 7.1)
80	HTTP	Microsoft IIS HTTP Server 7.5
135	MSRPC	Microsoft Windows Remote Procedure Call
139	NetBIOS-SSN	NetBIOS Session Service
445	Microsoft-DS	SMB service (Windows Server 2008 R2–2012)
3306	MySQL	MySQL Database Server 5.5.20
3389	MS-WBT-Server	Remote Desktop Protocol (RDP)
3920	SSL	Encrypted service (ssl/exasoftport1)
4848	HTTPS	Oracle GlassFish Admin Console
5985	HTTP	Microsoft HTTPAPI (WinRM / UPnP)
7676	Java-Message-Service	Java Message Service (JMS)
8009	AJP13	Apache JServ Protocol
8080	HTTP	Sun GlassFish Open Source Edition 4.0
8181	SSL	Secure GlassFish service
8383	HTTP	Apache HTTP Server
9200	HTTP	Elasticsearch REST API (REST API 1.1.1)
49152	MSRPC	Microsoft Windows RPC (Dynamic Port)
49153	MSRPC	Microsoft Windows RPC (Dynamic Port)
49154	MSRPC	Microsoft Windows RPC (Dynamic Port)
49155	MSRPC	Microsoft Windows RPC (Dynamic Port)
49176	Java-RMI	Java Remote Method Invocation

6.2 Vulnerability Tracker

https://docs.google.com/spreadsheets/d/1LLQB1rSTlB7rBLmyPCeD7DOjFv136JRSopGFKSzJueQ/edit?usp=sharing

7. Risk Assessment

7.1 CVSS Scoring

CVSS v3.1 was used to evaluate the severity of vulnerabilities based on exploitability and impact.

8. Remediation Recommendations

8.1 General Recommendations

- Regularly update and patch all services
- Disable unnecessary services and ports
- Enforce strong authentication mechanisms
- Conduct periodic vulnerability assessments

8.2 Specific Fixes

- **CVE-2016-1908 – OpenSSH (Port 22):**
  -Apply the latest OpenSSH security patch by upgrading to version 7.2 or later from the official OpenSSH repository: https://www.openssh.com/portable.html.
  -Disable X11 forwarding in the SSH configuration file (sshd_config) by setting X11Forwarding no and restart the SSH service.
- **CVE-1999-0501 – SSH Service (Default Credentials):**
  -Remove all default credentials and enforce strong authentication mechanisms.
  -Disable password-based authentication and enable SSH key-based authentication in sshd_config.
  -Refer to OpenSSH hardening documentation: https://www.openssh.com/manual.html.
- **CVE-2016-2183 – HTTPS Service (Port 8383):**
  -Disable weak SSL/TLS cipher suites such as 3DES and RC4.
  -Configure the web server to allow only strong ciphers and enforce TLS 1.2 or higher.
  -Refer to OWASP TLS hardening guidelines:
  https://cheatsheetseries.owasp.org/cheatsheets/Transport_Layer_Protection_Cheat_Sheet.html.
- **CVE-2016-10009 – OpenSSH (Port 22):**
  -Upgrade OpenSSH to version 7.4 or later using vendor-provided patches or the official OpenSSH release page: https://www.openssh.com/portable.html.
  -Remove legacy cryptographic algorithms and restart the SSH service.
- **CVE-2016-6210 – OpenSSH (Port 22):**
  -Mitigate user enumeration by upgrading OpenSSH to version 7.3 or later.
  -Limit authentication attempts, enable consistent authentication failure responses, and deploy brute-force protection such as Fail2Ban.
  -Refer to OpenSSH security advisories: https://www.openssh.com/security.html.
- **CVE-1999-0502 – FTP Service (Port 21):**
  -Change all default FTP credentials and disable anonymous access.
  -If FTP is not required, disable the service entirely and migrate to SFTP.
  -Refer to FTP server hardening documentation from the vendor.
- **Java JMX (Port 1617):**
  -Enable authentication and SSL for JMX remote management.
  -Restrict JMX access to trusted IP addresses only and block external access using firewall rules.
  -Refer to Oracle JMX security documentation:
  https://docs.oracle.com/javase/8/docs/technotes/guides/management/agent.html.
- **CVE-2013-2566 – RDP Service (Port 3389):**
  -Disable weak SSL/TLS cipher suites and enforce strong encryption for RDP connections.
  -Enable Network Level Authentication and apply the latest Windows security updates.
  -Refer to Microsoft RDP security guidance: https://learn.microsoft.com/en-us/windows-server/remote/remote-desktop-services/security.
- **CVE-2011-3389 – Multiple TLS Services:**
  -Disable deprecated TLS versions 1.0 and 1.1 across all affected services.
  -Enable TLS 1.2 or higher and apply vendor-recommended SSL/TLS patches.

9. Conclusion

The vulnerability assessment successfully identified multiple security weaknesses in the target system. Most issues were related to outdated software and insecure configurations. Applying the recommended remediation steps will significantly reduce the attack surface and improve the overall security posture. This lab exercise demonstrates a practical understanding of vulnerability assessment tools, risk analysis, and professional security reporting.

10. References

- OpenVAS Documentation:
  https://greenbone.github.io/docs/
- Nikto Documentation:
  https://cirt.net/Nikto2
- CVSS v3.1 Specification:
  https://www.first.org/cvss/v3.1/specification-document
- OWASP Testing Guide:
  https://owasp.org/www-project-web-security-testing-guide/