Vulnerability Assessment Report

Project Title: Vulnerability Assessment of Metasploitable 2
Group - 6 - Ramjith M
Date: May 23, 2025
Target IP: 192.168.29.195

---

1. Project Overview

This report presents the results of a vulnerability assessment conducted on a deliberately vulnerable Linux machine, Metasploitable 2, hosted within a virtualized lab environment. The objective was to simulate a real-world security evaluation to detect critical vulnerabilities, assess their risk, and offer recommendations for remediation. This simulation reflects real-world penetration testing processes and vulnerability management strategies that help businesses identify and fix their security weaknesses before malicious actors can exploit them.

This report provides insight into how attackers might exploit system weaknesses and how security analysts can discover, document, and mitigate those risks effectively. The vulnerabilities found in this system are commonly encountered in outdated or misconfigured systems across various industries, emphasizing the importance of continuous security audits.

---

2. Tools Used

- **OpenVAS (Greenbone Security Assistant)**: Used for full vulnerability scanning. OpenVAS includes a vast database of known vulnerabilities (NVTs) and produces detailed results with CVSS scores and remediation suggestions.

- **Nmap**: Performed network reconnaissance, port scanning, service identification, and OS fingerprinting. It provided a first look at the open and potentially exploitable services.

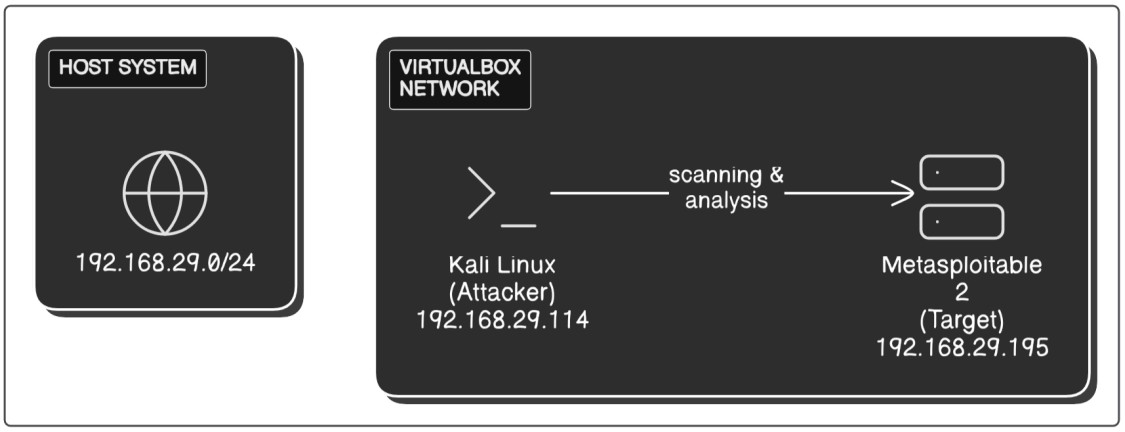
- **Angry IP Scanner**: A lightweight tool used to identify live hosts and basic port information across the local network.

- **VirtualBox**: The virtual lab environment was constructed using VirtualBox. Both the attacker machine (Kali Linux) and the target machine (Metasploitable 2) were configured as VMs within an internal NAT network.

Each of these tools was chosen for its specific utility and reliability within industry-standard cybersecurity workflows. This combination ensures a comprehensive approach to host discovery, vulnerability identification, and assessment reporting.

---

3. Network Diagram



The network environment was configured to simulate a typical network, with one attacker machine scanning and analyzing a single target. This setup reflects a common red team/blue team assessment scenario.

---

4. Key Vulnerability Findings (High Severity)

The vulnerabilities listed below were identified using OpenVAS and manually verified through service banners and configuration checks.

- **Apache Tomcat AJP RCE (Ghostcat) - CVE-2020-1938**
  - **Port**: 8009/tcp
  - **CVSS Score**: 9.8 (Critical)
  - **Summary**: The AJP protocol used by Apache Tomcat is exposed and exploitable, enabling attackers to read arbitrary files and execute commands on the host.
  - **Technical Insight**: This vulnerability can be exploited to gain full access to application source code or web configuration files, possibly escalating to remote code execution.
  - **Solution**: Update Tomcat to a secure version or disable AJP connector if not in use. Restrict connector to only local access if necessary.

- **vsftpd Backdoor Shell - CVE-2011-2523**
    - **Ports**: 21/tcp, 6200/tcp
    - **CVSS Score**: 9.8 (Critical)
    - **Summary**: A backdoored version of the vsftpd FTP daemon was found. This version opens a shell when a username with a smiley face is entered.
    - **Impact**: Allows unauthenticated attackers to gain shell access and potentially control the entire system.
    - **Solution**: Remove vsftpd 2.3.4 and verify the source of all downloaded binaries. Replace with a trusted, updated FTP server.
- **DistCC Remote Code Execution - CVE-2004-2687**
    - **Port**: 3632/tcp
    - **CVSS Score**: 9.3 (Critical)
    - **Summary**: The distcc service allows arbitrary command execution from any IP address without authentication.
    - **Impact**: Full command execution as the user running distcc, often root or daemon.
    - **Solution**: Uninstall distcc if not required. If used, configure it to allow access only from trusted IPs.
- **VNC Brute Force Authentication Bypass**
    - **Port**: 5900/tcp
    - **CVSS Score**: 9.0 (High)
    - **Summary**: The VNC server accepts weak or default credentials.
    - **Insight**: This can allow remote access to the system's graphical desktop and direct manipulation of services.
    - **Solution**: Change default password, enable encryption, and limit access using firewall rules.

- **Operating System End of Life - Ubuntu 8.04**
    - **CVSS Score**: 10.0 (Critical)
    - **Summary**: The system is running Ubuntu 8.04, which has not received security updates since 2013.
    - **Impact**: Unpatched vulnerabilities can be used to exploit the system without triggering modern defenses.
    - **Solution**: Replace with a supported operating system such as Ubuntu LTS 22.04 or newer.
- **rsh (Remote Shell) - Cleartext Authentication**
    - **Port**: 514/tcp
    - **CVSS Score**: 7.5 (High)
    - **Summary**: rsh allows unauthenticated remote shell access using insecure, plaintext communication.
    - **Insight**: Often accepts root logins without passwords if not configured properly.
    - **Solution**: Disable rsh entirely and enforce use of SSH with public key authentication.

---

5. Additional Observations

- Several services expose outdated software with multiple known vulnerabilities beyond the scope of the high-severity issues listed.
- Services such as telnet, RPC, and Samba were accessible without proper restrictions.
- MySQL and PostgreSQL were exposed to the network and may be accessible using default credentials.

---

6. Recommendations & Mitigation

1. Patch Management

Implement a consistent and automated patch management process to ensure software is updated regularly. Use tools like `unattended-upgrades` or enterprise patch managers.

2. Service Hardening

- Remove unnecessary services and daemons
- Use `chkconfig` or `systemctl` to disable startup for non-essential services

3. Credential Policy Enforcement

- Enforce minimum password complexity requirements
- Use PAM (Pluggable Authentication Modules) to enforce account lockouts

4. Firewalls and Network Controls

- Apply local firewall rules (UFW or iptables)
- Segment the network using VLANs or subnets for critical systems

5. Continuous Monitoring

- Enable logging for all inbound and outbound connections
- Use fail2ban to prevent brute force attacks
- Integrate log monitoring with tools like Graylog or Wazuh

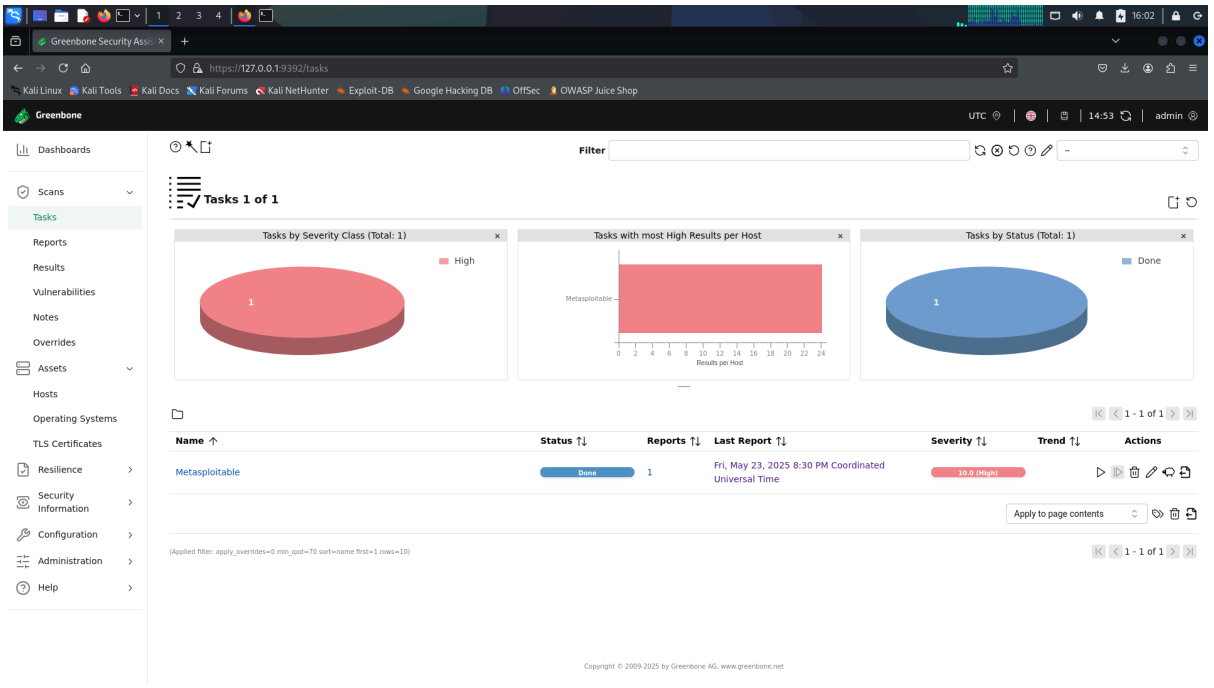
6. Vulnerability Management Lifecycle

- Schedule quarterly or monthly vulnerability scans
- Prioritize vulnerabilities using CVSS scores and exploitability
- Track remediation progress through a centralized dashboard or ticketing system

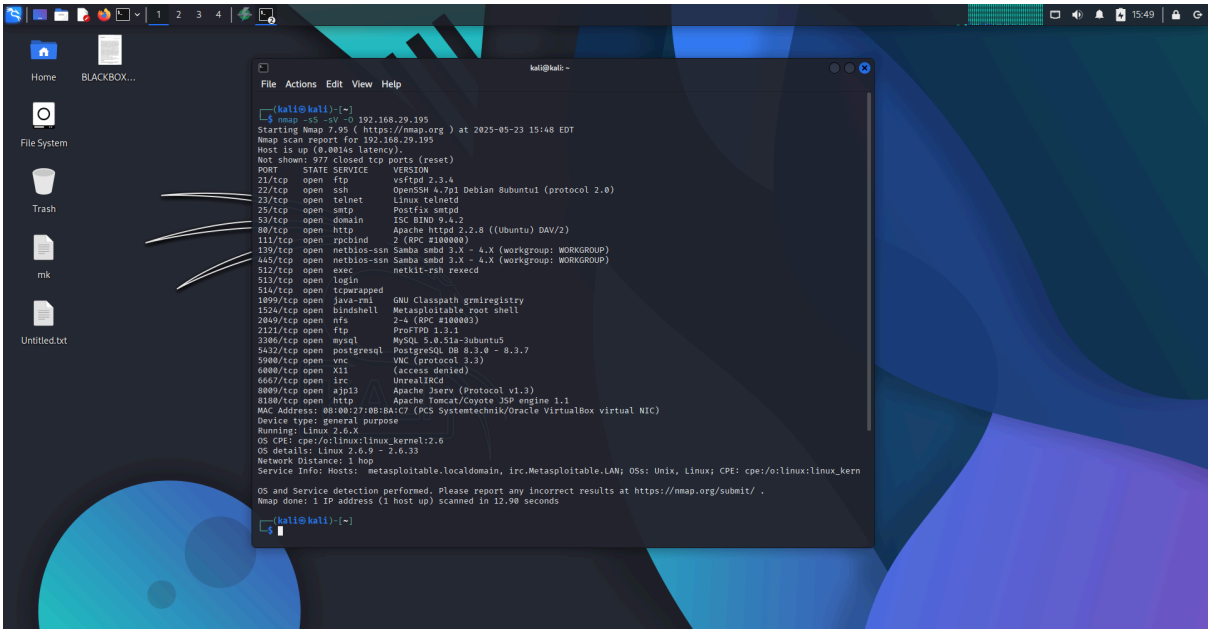
---

7. Supporting Evidence

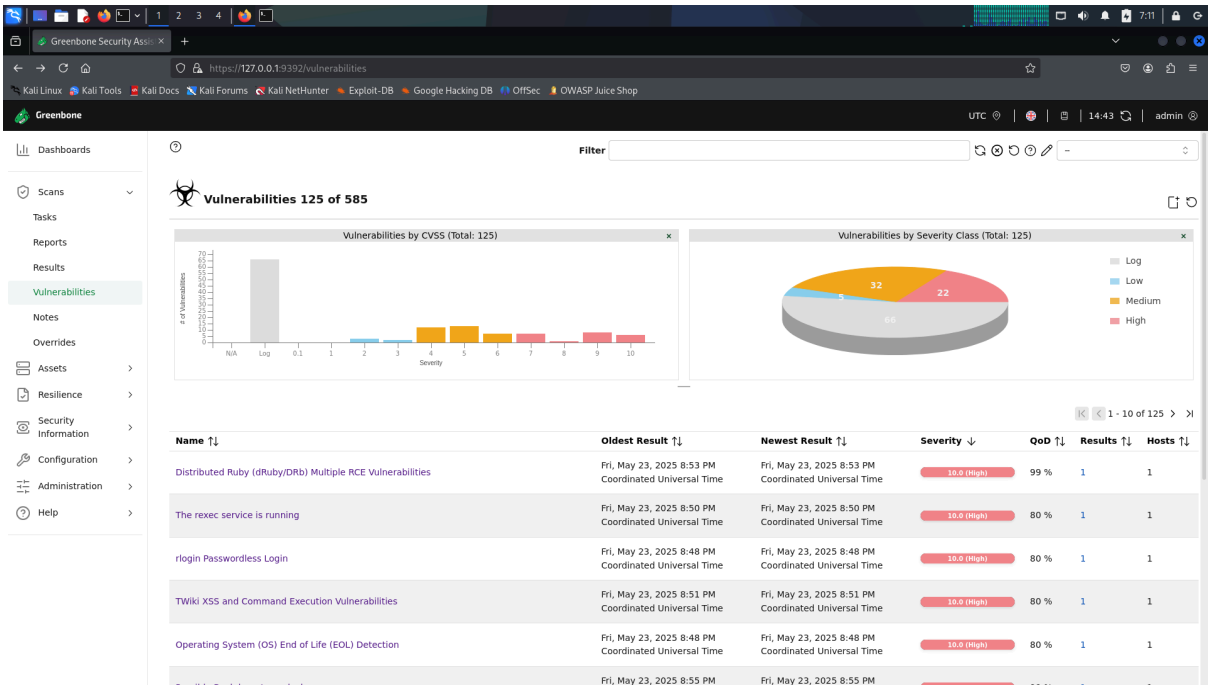
- Screenshot: OpenVAS Dashboard Summary



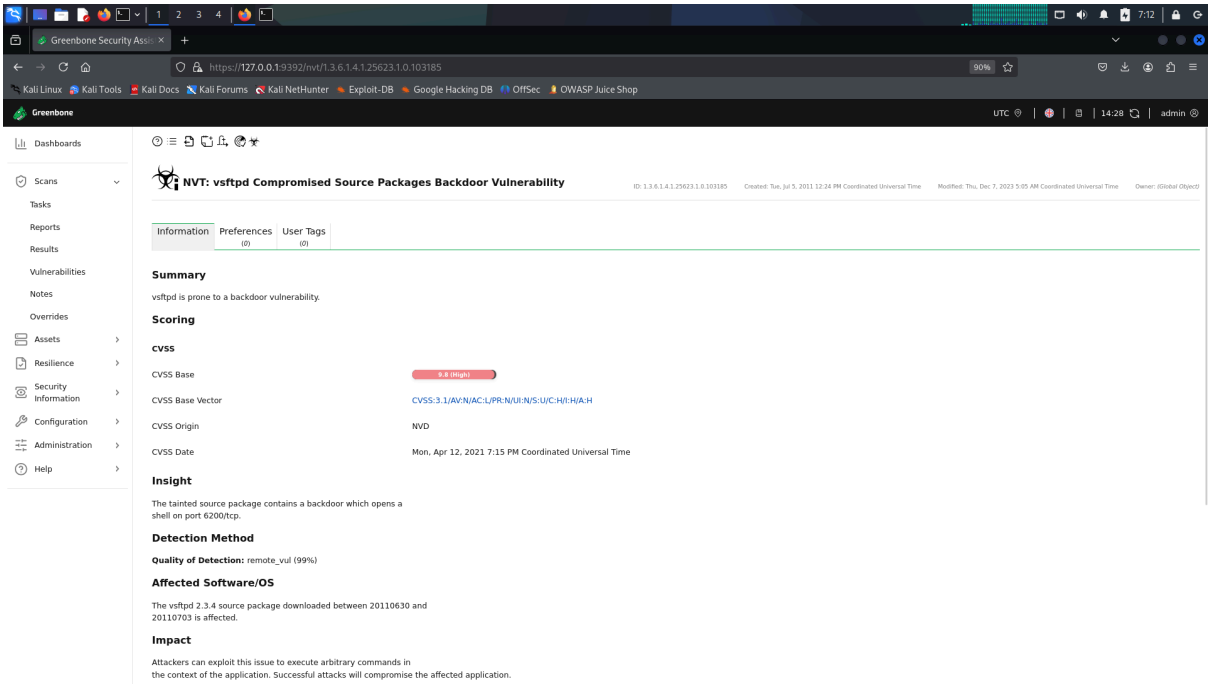
- Screenshot: Nmap Port Scan Results



- Screenshot: OpenVAS Top Vulnerabilities



- Screenshot: vsftpd Backdoor Detection



8. Conclusion

This vulnerability assessment has shown that Metasploitable 2 contains multiple exploitable services and critical misconfigurations, providing a rich environment for real-world attack simulation. While Metasploitable is designed to be insecure, the vulnerabilities encountered are reflective of those commonly found in poorly maintained legacy systems.

Effective vulnerability management, combined with a proactive patching strategy and access control enforcement, significantly reduces an organization’s attack surface. This report serves as a foundation for implementing best practices in secure system configuration and monitoring.

Group - 6 - Ramjith M