Vulnerability Assessment — Metasploitable (Using Nessus)

Executive summary

A vulnerability scan was performed against a Metaploitable Machine hosted on virtual machine using the Nessus scanner to identify security weaknesses, assess their potential impact, and recommend corrective actions. This assessment uncovered multiple issues spanning critical to informational severity levels and also providing remediation guidance.

Scope and environment

- **Target:** Metasploitable machine hosted in VirtualBox.
- **Scanning host:** Nessus installed on the Windows host machine.
- **Network configuration:** Host-only adapter configured in VirtualBox to allow direct communications between the host (scanner) and the guest (target).
- **Objective:** Discover open services, outdated components, misconfigurations, and potential backdoors; classify findings by severity and provide mitigation guidance.

Tools

- **Nessus** — vulnerability scanner used to enumerate services and detect known vulnerabilities.
- **VirtualBox** — platform used to run the Metasploitable VM.
- **Metasploitable** — the target intentionally configured with vulnerable services for testing.

Methodology

1. Verified network connectivity between the scanner and the Metasploitable VM.
2. Launched a full Nessus scan against the VM's IP address.
3. Collected results showing open ports, service banners, protocol versions, and matched CVEs.
4. Triage and prioritization were applied based on CVSS scores and exploitability to produce actionable remediation recommendations.

Summary of findings

Total vulnerabilities discovered: **105**, broken down by severity:

- **Critical:** 7
- **High:** 2
- **Medium:** 19
- **Low:** 8
- **Informational:** 69

Below are the most significant issues identified and suggested remediation steps.

Critical issues (examples)

1. **Apache Tomcat AJP Connector request injection (Ghostcat) — CVE-2020-1938**
    - **Impact:** Allows crafted requests to access sensitive files or potentially enable remote code execution via the AJP connector.
    - **Recommendation:** Disable or secure the AJP connector if not required, apply the vendor patch, and restrict access to trusted hosts via firewall rules.
2. **Detected shell backdoor (bind shell)**
    - **Impact:** Presence of a backdoor could enable an attacker to obtain full remote control over the host.
    - **Recommendation:** Isolate the system, perform a forensic analysis to confirm compromise, remove the backdoor, reinstall from a known-good image, and rotate any credentials that may have been exposed.
3. **SSLv2 and SSLv3 protocols enabled**
    - **Impact:** Older SSL/TLS protocol versions are susceptible to multiple attacks (e.g., POODLE), undermining confidentiality and integrity.
    - **Recommendation:** Disable SSLv2/SSLv3 on affected services and configure TLS 1.2+ with strong cipher suites. Test with SSL/TLS assessment tools before deployment.
4. **Outdated Apache Tomcat (SEoL / older than version 5.5.x)**
    - **Impact:** Very old Tomcat versions contain critical flaws that may permit remote code execution.
    - **Recommendation:** Upgrade Tomcat to a maintained release, apply security updates, and follow secure configuration guides (disable unused connectors, limit privileges, use least-privileged accounts).

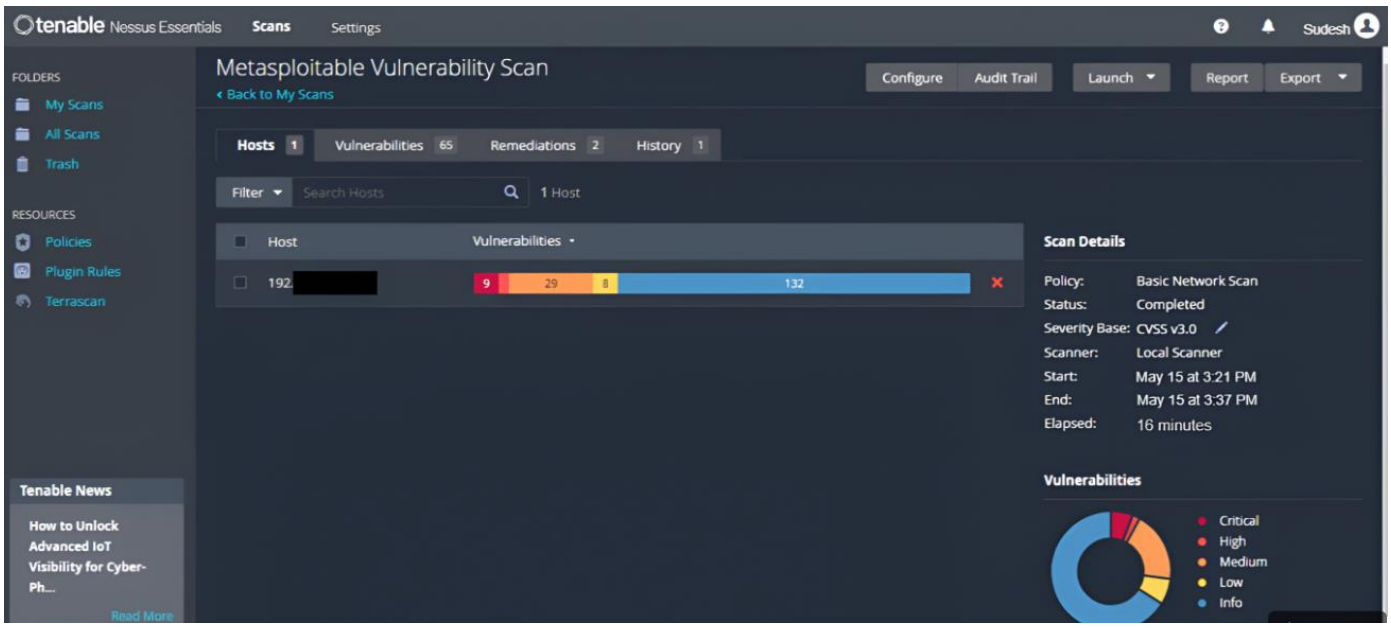
High/Medium issues

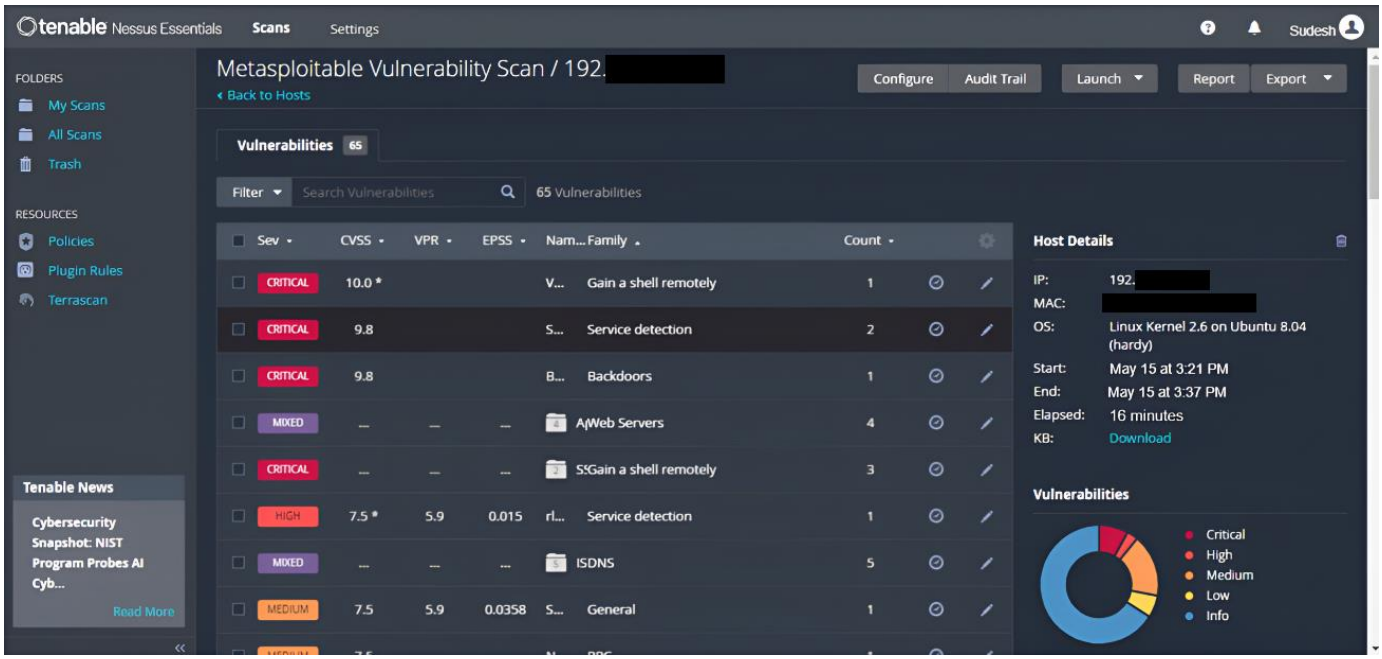
- **ISC BIND downgrade/reflected DoS (High)** — May lead to service disruption or downgrade attacks against DNS services. Patch BIND and apply rate-limiting controls.
- **rlogin service exposed (High)** — rlogin is an insecure remote login protocol lacking modern authentication and encryption. Disable rlogin and replace with SSH using key-based authentication.
- **NFS exports world-readable (Medium)** — NFS shares open to all users expose sensitive files. Restrict export permissions and use access control lists or export options limiting hosts.
- **Samba (Badlock-type) vulnerabilities (Medium)** — Samba flaws can enable privilege escalation or DoS. Ensure Samba is updated and follow hardening guidelines (disable SMBv1, enforce authentication policies).

Full report and artifacts

A comprehensive PDF containing all findings, scan output, and prioritized recommendations was produced.

Screenshots and exported Nessus output were captured to document scan configuration and key results (scan summary, critical finding list, and service details).





Lessons learned

- The scan reinforced the importance of network segmentation and minimizing exposed services in test and production environments.
- Maintaining up-to-date software and disabling legacy protocols substantially reduces the attack surface.
- Automated scans are valuable for discovery, but results require manual triage to assess contextual risk and false positives.

Recommendations and next steps

1. **Immediate remediation:** Address the critical items first — remove backdoors, patch or upgrade vulnerable services, and disable legacy protocols.

2. **Harden services:** Disable unnecessary daemons (e.g., rlogin), restrict access using firewalls and access control lists, and adopt secure configuration baselines.
3. **Retest:** After remediation, run a follow-up scan to confirm issues are resolved.
4. **Penetration testing:** Use the validated findings to perform controlled exploitation exercises (e.g., via Metasploit) to validate the effectiveness of mitigations. Do this in a safe, authorized environment.
5. **Broaden tooling:** Consider running complementary tools (Nmap for service discovery, OpenVAS for additional coverage) to cross-validate findings and improve detection coverage.
6. **Documentation and process:** Record procedures for patching, rebuild, and incident response; maintain a regular scan cadence to detect regressions or new issues.

Conclusion

The Nessus assessment identified multiple vulnerabilities across a range of severities on the Metasploitable VM. Prioritizing corrective actions based on business risk, starting with eliminating backdoors, patching critical services, and removing legacy protocols, will substantially reduce exposure. Regular scanning, combined with hardening and verification through follow-up tests, will improve the security posture over time.