Exam Cram Notes: Vulnerability Management

1. Overview

Vulnerability management is a **proactive approach** to identifying, assessing, prioritizing, and remediating security vulnerabilities in an organization's IT infrastructure. It helps minimize attack surfaces and reduce cybersecurity risks.

---

2. The Vulnerability Management Lifecycle

1️⃣ **Identification –** Discover vulnerabilities through scanning and threat intelligence.
2️⃣ **Evaluation –** Assess the risk level of each vulnerability (CVSS scoring).
3️⃣ **Prioritization –** Determine which vulnerabilities to fix first based on severity and exploitability.
4️⃣ **Remediation –** Apply patches, configuration changes, or compensating controls.
5️⃣ **Verification –** Test fixes and confirm vulnerabilities are eliminated.
6️⃣ **Continuous Monitoring –** Regularly scan systems and update risk assessments.

---

3. Vulnerability Identification Methods

A. Vulnerability Scanning

✅ **Automated Scanning Tools –** Detect missing patches, misconfigurations, and known exploits.
✅ **External vs. Internal Scans –** Check public-facing and internal assets.
✅ **Credentialed vs. Non-Credentialed Scans –** Deep vs. surface-level scans.

🔹 **Tools:** Nessus, OpenVAS, Qualys, Rapid7 Nexpose

B. Penetration Testing

✅ Simulates real-world attacks to **exploit vulnerabilities**.
✅ Provides **a deeper analysis** beyond automated scans.
✅ Identifies **zero-day vulnerabilities** that scanners might miss.

🔹 **Tools:** Metasploit, Burp Suite, Kali Linux

C. Threat Intelligence

✅ Uses **threat feeds, databases, and advisories** to stay updated on new threats.
✅ Common sources: **MITRE ATT&CK, CVE Database, NIST NVD, OWASP Top 10**

---

4. Evaluating & Prioritizing Vulnerabilities

A. Common Vulnerability Scoring System (CVSS)

✅ Rates vulnerabilities on a scale of **0-10** (higher = more severe).
✅ **Three Categories:**

- **Base Score** – Impact & exploitability.
- **Temporal Score –** Changes over time due to patches/exploits.
- **Environmental Score –** Organization-specific risk factors.

🔷 **Example:** A CVSS 9.8 vulnerability (Remote Code Execution) is more critical than a CVSS 5.5 (Denial-of-Service).

B. Risk-Based Prioritization

✅ Consider:

- **Likelihood of Exploitation** (Is there an active exploit in the wild?)
- **Asset Criticality** (Is the affected system mission-critical?)
- **Potential Impact** (What is the damage if exploited?)

✅ Prioritize:

- 🔥 **Critical vulnerabilities** with active exploits (patch immediately).
- ⚠️ **High-severity issues** on key systems (schedule quick fixes).
- 🟡 **Low-risk vulnerabilities** (fix during routine maintenance).

---

5. Vulnerability Remediation Strategies

A. Patch Management

✅ Apply vendor **security patches** as soon as possible.
✅ Test patches in **sandbox environments** before deploying.
✅ Maintain an **update schedule** for OS, apps, and firmware.

🔷 **Patch Management Tools:** WSUS, SCCM, Ivanti

B. Configuration Hardening

✅ Disable **unnecessary services, ports, and default accounts**.
✅ Apply **secure baseline configurations** for OS, networks, and applications.
✅ Follow **CIS Benchmarks and NIST Security Guides**.

C. Compensating Controls

✅ If patches aren't available:

- Implement **firewall rules, IPS/IDS, access control changes**.
- Use **network segmentation** to isolate vulnerable assets.
- Monitor for **suspicious activity** on vulnerable systems.

---

6. Verification & Ongoing Monitoring

A. Re-Scanning & Testing

✅ Conduct post-remediation **scans** to confirm fixes.
✅ Use **penetration testing** to validate security controls.

B. Continuous Monitoring & Reporting

✅ Deploy **SIEM tools (Splunk, ELK, Microsoft Sentinel)** to detect exploitation attempts.
✅ Maintain detailed **audit logs** of vulnerabilities and resolutions.
✅ Regularly update **risk assessments** and security policies.

---

7. Key Exam Takeaways

✅ **Vulnerability Management is a continuous cycle** (Identify → Evaluate → Prioritize → Remediate → Verify).
✅ Use **automated scanners (Nessus, Qualys) & penetration testing** to detect vulnerabilities.
✅ Apply **patches first** for critical issues, use **compensating controls** if patches aren't available.
✅ **Monitor logs, SIEM alerts, and threat intelligence feeds** for new threats.
✅ Follow industry best practices (**CIS Benchmarks, CVSS Scoring, NIST Guidelines**).