# Module 05: Vulnerability Analysis

Vulnerability analysis is the process of identifying, quantifying, and prioritizing the vulnerabilities in a system. This phase follows Scanning and Enumeration, using the data gathered to determine which known security flaws exist on the target.

## 1. Vulnerability Analysis Concepts

### What is a Vulnerability?

A vulnerability is a weakness in the design, implementation, or operation of a system that could be exploited to violate the system's security policy.

### Objectives of Vulnerability Analysis

* **Identify Security Gaps:** Discover weaknesses before an attacker does.
* **Prioritize Risks:** Determine which flaws pose the greatest threat to the organization.
* **Compliance:** Ensure the organization meets regulatory requirements (PCI-DSS, HIPAA, etc.).
* **Resource Allocation:** Help management decide where to spend the security budget.

## 2. Vulnerability Research & Classification

### Research Resources (EXAM CRITICAL)

You must be familiar with these databases:

* **CVE (Common Vulnerabilities and Exposures):** A list of publicly disclosed cybersecurity vulnerabilities. Format: CVE-YYYY-NNNN.
* **NVD (National Vulnerability Database):** The U.S. government repository of standards-based vulnerability management data, synchronized with the CVE list.
* **Full-Disclosure / Bugtraq:** Mailing lists for detailed vulnerability reports.
* **Exploit-DB:** A database of exploits for known vulnerabilities.

### Vulnerability Classifications

* **Misconfigurations:** Default settings, open shares, or unnecessary services.
* **Default Passwords:** Using "admin/admin" or "1234".
* **Buffer Overflows:** Sending more data than a buffer can hold to overwrite memory.
* **Unpatched Software:** Operating systems or apps missing security updates.
* **Design Flaws:** Logical errors in how a protocol or application works.

## 3. Vulnerability Scoring (CVSS)

The **Common Vulnerability Scoring System (CVSS)** provides a numerical score reflecting the severity of a vulnerability.

### CVSS Metric Groups

1. **Base Score (Static):** Represents the intrinsic qualities of a vulnerability.
   * *Exploitability:* Attack Vector (AV), Attack Complexity (AC), Privileges Required (PR), User Interaction (UI).
   * *Impact:* Confidentiality (C), Integrity (I), Availability (A).
2. **Temporal Score (Changes over time):** Represents the current state of exploit techniques or code availability. Includes Exploit Code Maturity, Remediation Level, and Report Confidence.
3. **Environmental Score (User-specific):** Represents the impact within a specific organization's environment, allowing for the adjustment of the base score based on local security controls.

### Severity Levels

* **Low:** 0.1 - 3.9
* **Medium:** 4.0 - 6.9
* **High:** 7.0 - 8.9
* **Critical:** 9.0 - 10.0

## 4. Vulnerability Assessment Types

* **Active Assessment:** Directly interacting with the target using a scanner (e.g., Nessus) to find vulnerabilities.
* **Passive Assessment:** Sniffing network traffic to identify versions of software and potential flaws without "touching" the target.
* **Host-Based Assessment:** Installing an agent or using credentials to look inside the OS for misconfigurations and local flaws.
* **Network-Based Assessment:** Scanning the network for exposed services and weak protocols.
* **Database Assessment:** Specifically looking for SQL injection points or weak database permissions.

## 5. Vulnerability Management Life Cycle

1. **Baseline:** Identify assets and set the scope.
2. **Vulnerability Assessment:** Run the scans.
3. **Risk Assessment:** Analyze findings and assign priority.
4. **Remediation:** Patch the systems or apply workarounds.
5. **Verification:** Re-scan to ensure the patch worked.
6. **Monitor:** Continuous monitoring for new threats.

## 6. Vulnerability Assessment Tools (Expanded)

|  |  |  |
| --- | --- | --- |
| **Tool** | **Focus Area** | **Description** |
| **Nessus** | General Network | The industry standard for identifying vulnerabilities, misconfigurations, and malware. |
| **OpenVAS (GVM)** | General Network | A comprehensive open-source vulnerability scanner and manager. |
| **Qualys Guard** | Cloud/Enterprise | A SaaS-based vulnerability management platform used by large organizations. |
| **Nikto** | Web Server | A specialized tool for finding dangerous files, outdated server software, and configuration issues on web servers. |
| **Acunetix** | Web Applications | An automated web application security testing tool that audits apps for SQLi, XSS, etc. |
| **Saint** | Vulnerability/Exploit | A tool that combines vulnerability scanning with exploitation capabilities. |
| **Nmap (NSE)** | Network Discovery | Uses the Nmap Scripting Engine (e.g., --script vuln) to find specific vulnerabilities during the scanning phase. |
| **GFI LanGuard** | Patch Management | Acts as a virtual security consultant by managing patches and auditing networks. |

## 7. Hands-On Lab Sessions (CEH v13 Practice)

### Lab 1: Professional Scanning with Nessus

Install Nessus

Step 1 - sudo apt update && sudo apt upgrade -y

Step 2: Download Nessus for Linux (Debian/Ubuntu)

<https://www.tenable.com/downloads/nessus>

sudo dpkg -i /home/kaushalya/Downloads/Nessus-10.11.1-debian10\_amd64.deb

Start service

sudo systemctl start nessusd

sudo systemctl enable nessusd

sudo systemctl status nessusd

open browser

<https://localhost:8834>

create your activation code

Choose **Nessus Essentials** (FREE – best for labs)

for me..

nessus - activation code

TSFU-UR5F-USFR-ETSF-UUNU

Verify and create admin credentials

For example

admin

admin

* **Goal:** Identify and prioritize vulnerabilities on a Windows/Linux target.
* **Steps:**
  1. Log into the Nessus interface (https://localhost:8834).
  2. Create a **New Scan** and select **Basic Network Scan**.
  3. Enter the target IP address and name the scan "CEH\_Module5\_Practice".
  4. Under **Settings**, navigate to **Report** and ensure "Allow users to edit scan results" is enabled.
  5. Run the scan and examine the "Vulnerabilities" tab. Sort by **Severity** (Critical/High).
  6. Click on a Critical vulnerability to view its **CVE ID**, **CVSS Score**, and **Remediation** advice.

### Lab 2: Web Server Assessment with Nikto

* **Goal:** Discover misconfigured web server files and outdated versions.
* **Action:** Open the Parrot OS terminal.
* **Command:** nikto -h 10.10.10.15
* **Analysis:** Look for headers like X-Frame-Options (Clickjacking protection) and directories like /admin/ or /config/ that should be hidden.

### Lab 3: Manual Vulnerability Verification (Anonymous FTP)

* **Goal:** Verify an "Anonymous FTP" vulnerability reported by a scanner.
* Action: 1. Attempt to log in: ftp 10.10.10.15  
  2. When prompted for a username, type anonymous.  
  3. For the password, type anything (e.g., guest).  
  4. If successful, run ls to see if you can view files. This confirms the vulnerability found by tools like Nessus.

### Lab 4: Vulnerability Discovery with Nmap NSE

* **Goal:** Quickly check for specific vulnerabilities during the scan phase.
* **Action:** nmap --script vuln 10.10.10.15
* **Practical Note:** This is often the fastest way to find high-impact flaws like **EternalBlue (MS17-010)** or **Heartbleed**.

## 8. Countermeasures

* **Patch Management:** Automate the deployment of security updates using tools like WSUS or GFI LanGuard.
* **Hardening:** Disable unused services (e.g., Telnet, FTP) and change default credentials immediately.
* **Least Privilege:** Ensure users and services only have the access they absolutely need.
* **Vulnerability Shielding:** Use an IPS/IDS to block traffic targeting a known unpatched flaw while waiting for a patch window.
* **Configuration Auditing:** Regularly compare system settings against security benchmarks (e.g., CIS Benchmarks).