VULNERABILITY ASSESSMENT REPORT

Ubuntu Server 24.04 LTS — Lab Environment

Prepared by: Ruben

Date: February 25, 2026

Classification: Confidential

1. Executive Summary

This report presents the findings of a vulnerability assessment conducted against an Ubuntu Server 24.04 LTS system hosted in a virtualized lab environment. The assessment was performed using Nmap with vulnerability detection scripts to identify security weaknesses across all exposed services.

The assessment identified multiple critical and high-severity vulnerabilities, primarily stemming from an outdated Apache web server (version 2.4.25) and a known remote code execution vulnerability in OpenSSH. Immediate remediation is recommended for all critical findings.

Metric	Value
Target System	192.168.0.89 (Ubuntu Server 24.04 LTS)
Scan Date	February 25, 2026
Scanner	Nmap 7.95 with vuln scripts
Open Ports	3 (SSH, HTTP, HTTP-alt)
Critical Findings	**12+**
High Findings	**8+**
Medium Findings	**5+**

2. Scope and Methodology

2.1 Scope

The assessment targeted a single Ubuntu Server 24.04 LTS virtual machine (192.168.0.89) running on Oracle VirtualBox in a bridged network configuration. The server hosts two intentionally vulnerable web applications (DVWA and OWASP Juice Shop) for educational purposes.

2.2 Methodology

The assessment followed a structured approach:

1. Service discovery and version detection (Nmap -sV)
2. Default script scanning for common misconfigurations (Nmap -sC)
3. Operating system fingerprinting (Nmap -O)
4. Vulnerability script scanning against known CVE databases (Nmap --script vuln)

2.3 Tools Used

Tool	Version	Purpose

Nmap	7.95	Port scanning, service detection, vulnerability scanning
Nmap Scripting Engine	Built-in	CVE detection, enumeration, brute force testing
Vulners Database	Current	CVE correlation and exploit availability

3. Findings

3.1 OpenSSH Remote Code Execution (CVE-2024-6387)

Severity	CVSS Score	Port	Service
Critical	8.1	22/tcp	OpenSSH 9.6p1

Description: The installed version of OpenSSH (9.6p1) is vulnerable to CVE-2024-6387, known as "regreSSHion." This is a signal handler race condition that can lead to unauthenticated remote code execution as root. Multiple public exploits are available for this vulnerability.

Impact: An unauthenticated attacker could gain complete control of the server with root privileges. This is the highest-impact vulnerability identified in the assessment.

Remediation:

- Update OpenSSH to the latest patched version immediately
- As a temporary mitigation, set LoginGraceTime to 0 in sshd_config (note: this introduces a denial-of-service risk)
- Restrict SSH access to trusted IP ranges using firewall rules

3.2 Apache HTTP Server — Multiple Critical Vulnerabilities

Severity	CVSS Score	Port	Service
Critical	9.8 (highest)	80/tcp	Apache 2.4.25

Description: Apache HTTP Server version 2.4.25 (released 2017) contains numerous known vulnerabilities spanning multiple years of unpatched releases. Key CVEs include:

- CVE-2021-44790 (9.8) — Buffer overflow enabling remote code execution via mod_lua
- CVE-2023-25690 (9.8) — HTTP request smuggling allowing security control bypass
- CVE-2022-31813 (9.8) — IP-based authentication bypass via X-Forwarded-For manipulation
- CVE-2021-40438 (9.0) — Server-Side Request Forgery (SSRF) via mod_proxy
- CVE-2019-0211 (7.8) — Local privilege escalation to root

Impact: Attackers can achieve remote code execution, bypass authentication, or escalate privileges to root. The sheer number of critical vulnerabilities makes this the highest-risk service on the system.

Remediation:

- Upgrade Apache to the latest stable release (2.4.62+)
- Enable automatic security updates for the web server package
- Implement a Web Application Firewall (WAF) as an additional layer of defense

3.3 Missing HTTP Security Headers and Cookie Flags

Severity	CVSS Score	Port	Service
Medium	N/A	80/tcp	DVWA (PHP)

Description: The PHPSESSID cookie is set without the HttpOnly flag on both the root path and the login page. This allows client-side JavaScript to access the session cookie, enabling session hijacking through Cross-Site Scripting (XSS) attacks.

Impact: If an attacker finds an XSS vulnerability (which was confirmed in the Week 4 assessment), they can steal user session cookies and impersonate authenticated users, including administrators.

Remediation:

- Set the HttpOnly flag on all session cookies in php.ini: session.cookie_httponly = 1
- Set the Secure flag to prevent cookie transmission over HTTP: session.cookie_secure = 1
- Implement Content Security Policy (CSP) headers to mitigate XSS

3.4 Directory Listing and Information Disclosure

Severity	CVSS Score	Port	Service
Medium	N/A	80/tcp	Apache/DVWA

Description: Nmap's http-enum script identified several exposed directories and files that provide valuable information to attackers:

- /config/ — Configuration directory with directory listing enabled
- /docs/ — Documentation directory accessible to the public
- /external/ — External resources directory exposed
- /.gitignore — Reveals project structure and file organization
- /robots.txt — Discloses paths the site owner wants hidden from search engines
- /login.php — Identified as a possible admin interface

Impact: Exposed directories can leak sensitive configuration files, credentials, or application logic. The .gitignore file reveals the internal structure of the application, aiding attackers in targeting specific components.

Remediation:

- Disable directory listing in Apache configuration (Options -Indexes)
- Remove or restrict access to .gitignore, robots.txt, and non-essential directories
- Implement access controls on administrative paths

3.5 CORS Misconfiguration (Juice Shop)

Severity	CVSS Score	Port	Service
Medium	N/A	3000/tcp	Juice Shop

Description: The Juice Shop application returns Access-Control-Allow-Origin: * in its HTTP response headers. This permits any external website to make cross-origin requests to the application, bypassing the browser's same-origin policy.

Impact: A malicious website could make authenticated API requests to the Juice Shop on behalf of a logged-in user, potentially modifying data, extracting information, or performing actions without the user's knowledge.

Remediation:

- Restrict CORS to specific trusted domains instead of using a wildcard
- Validate the Origin header server-side before reflecting it

4. Risk Summary

Finding	Severity	CVSS	Status
OpenSSH RCE (CVE-2024-6387)	**Critical**	8.1	Requires immediate patching
Apache Multiple CVEs	**Critical**	9.8	Requires immediate upgrade
Missing HttpOnly Cookie Flag	**Medium**	—	Configuration change needed
Directory Listing / Info Disclosure	**Medium**	—	Configuration change needed
CORS Wildcard (Juice Shop)	**Medium**	—	Configuration change needed

5. Remediation Priorities

Based on the severity and exploitability of the findings, the following remediation order is recommended:

5.1 Immediate (Within 24 Hours)

5. Patch OpenSSH to address CVE-2024-6387 — public exploits exist for unauthenticated RCE
6. Upgrade Apache HTTP Server from 2.4.25 to the latest stable release
7. Restrict SSH access to known IP addresses via UFW firewall rules

5.2 Short-Term (Within 1 Week)

8. Set HttpOnly and Secure flags on all session cookies
9. Disable directory listing on all web server virtual hosts
10. Remove or restrict access to exposed files (.gitignore, robots.txt)
11. Configure proper CORS policies on all web applications

5.3 Ongoing

12. Enable automatic security updates for all installed packages
13. Implement regular vulnerability scanning on a weekly or monthly cadence
14. Deploy a host-based intrusion detection system (e.g., OSSEC or Wazuh)
15. Maintain an asset inventory with software versions for rapid CVE correlation

6. Conclusion

The target system presents a significant attack surface due to severely outdated software and insecure default configurations. The combination of a vulnerable OpenSSH service with public exploits and a critically outdated Apache web server means an attacker could achieve full system compromise with minimal effort.

The most urgent action is patching both OpenSSH and Apache, which would eliminate the majority of critical vulnerabilities identified in this assessment. The remaining medium-severity findings relate to configuration hardening that should be addressed as part of a broader security baseline.

This assessment demonstrates the importance of continuous vulnerability management. Software that is not regularly patched quickly accumulates exploitable vulnerabilities, as evidenced by the dozens of CVEs affecting the Apache 2.4.25 installation spanning from 2017 to 2025.