Task 3rd: Vulnerability Scanning with Nessus Essentials

3rd Task SUBMISSION REPORT OF ELEVATE LABS CYBERSECURITY INTERNSHIP

NAME	**SANDEEP MEWARA**
Submitted to:	**Elevate Labs**
Name of the Academic Institute	**Ganpat University**

REPORT SUBMITTED TO

As part of the Cyber Security Internship, I have completed "Task 3rd: Vulnerability Scanning with Nessus Essentials" by following all steps as instructed. Below is a detailed summary of each step followed during the task.

Task 3rd: Vulnerability Scanning with Nessus Essentials

Task Objective

To gain hands-on experience in identifying security vulnerabilities on a local Linux system using widely adopted vulnerability assessment tool: **Nessus Essentials**. This exercise helps in understanding the vulnerability scanning process, CVSS scoring, mitigation strategies, and practical security assessment reporting.

Tools Used

Tool	Description
Nessus Essentials	A free-tier vulnerability scanner developed by Tenable. Used globally for professional vulnerability assessments.

Test Environment Configuration

Component	Details
Operating System	Kali Linux (Rolling) – Debian-based distribution with Linux kernel 6.12.25-amd64
Host Type	Virtual Machine (VMware detected via MAC: 00:0c:29:30:8a:91)
Target IP (IPv4)	127.0.0.1 (localhost — scanned by Nessus)
Loopback IP	127.0.0.1
Docker Interface	docker0 – 172.17.0.1 detected as a bridge interface
MAC Address	00:0c:29:30:8a:91 (VMware Virtual Adapter)
Hostname	kali (detected by Nessus)
Network Interfaces	lo, eth0, docker0
Running Services	Apache HTTP Server, SSH, Docker containers, Nessus daemon (port 8834)
Installed Software	Apache 2.4.63, Node.js 20.11.0, Log4j (1.x and 2.x), Python Tornado, Trivy, ClamAV
Firewall Status	UFW not actively detected (presumed disabled or not interfering with scans)
NAT/Bridged Mode	Likely Bridged or NAT mode (VM uses local loopback and host-only networking)

Task 3rd: Vulnerability Scanning with Nessus Essentials

Vulnerability Scanning with Nessus Essentials

About Nessus Essentials

Nessus is a vulnerability scanner developed by Tenable. Its Essentials version is free for up to 16 IP addresses, making it ideal for labs and students.

Step-by-Step Process

1. Download and Install

wget https://www.tenable.com/downloads/api/v1/public/pages/nessus/downloads/<nessus-deb-package>

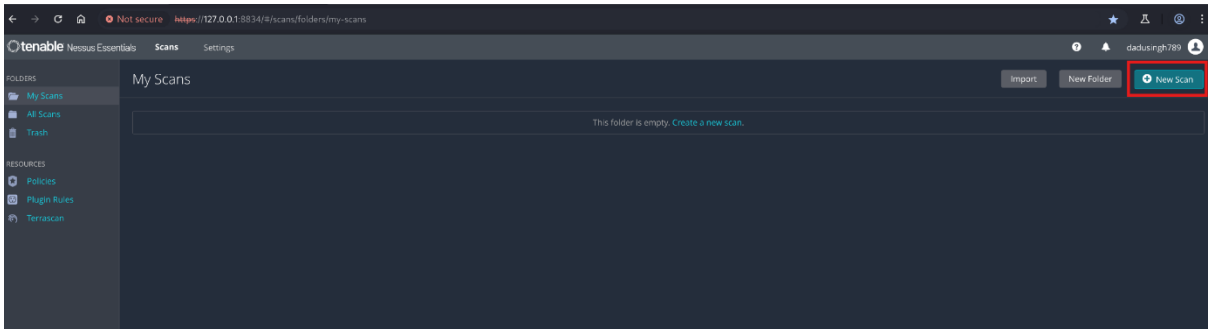
sudo dpkg -i <Nessus-deb-package>

sudo systemctl start nessusd

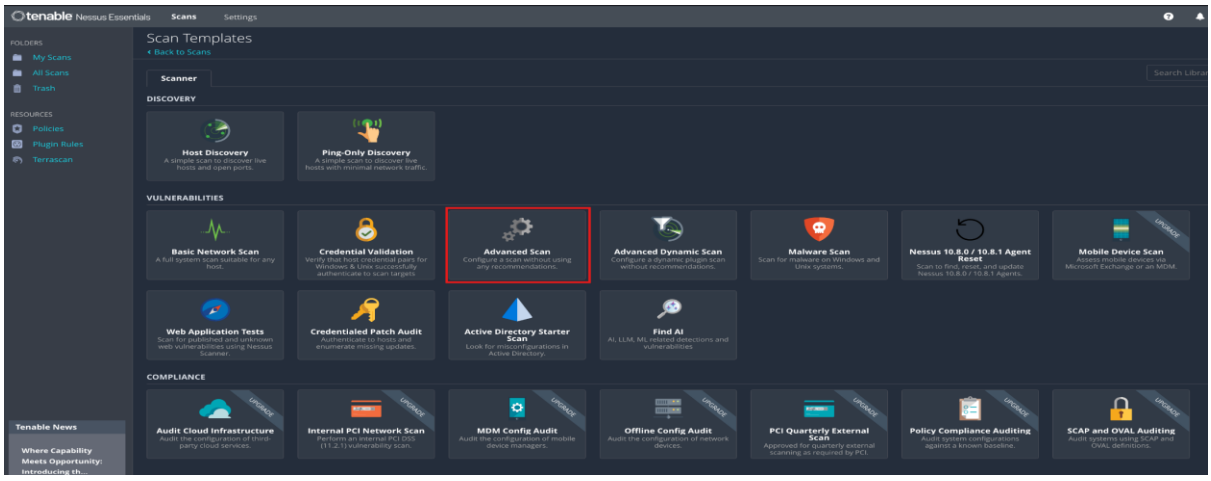
2. Setup via Web Interface

- URL: https://localhost:8834
- Choose **Nessus Essentials**
- Register for free activation code at Tenable
- Create local admin user
- Plugin download begins (~25 mins)

3. New Scan Creation

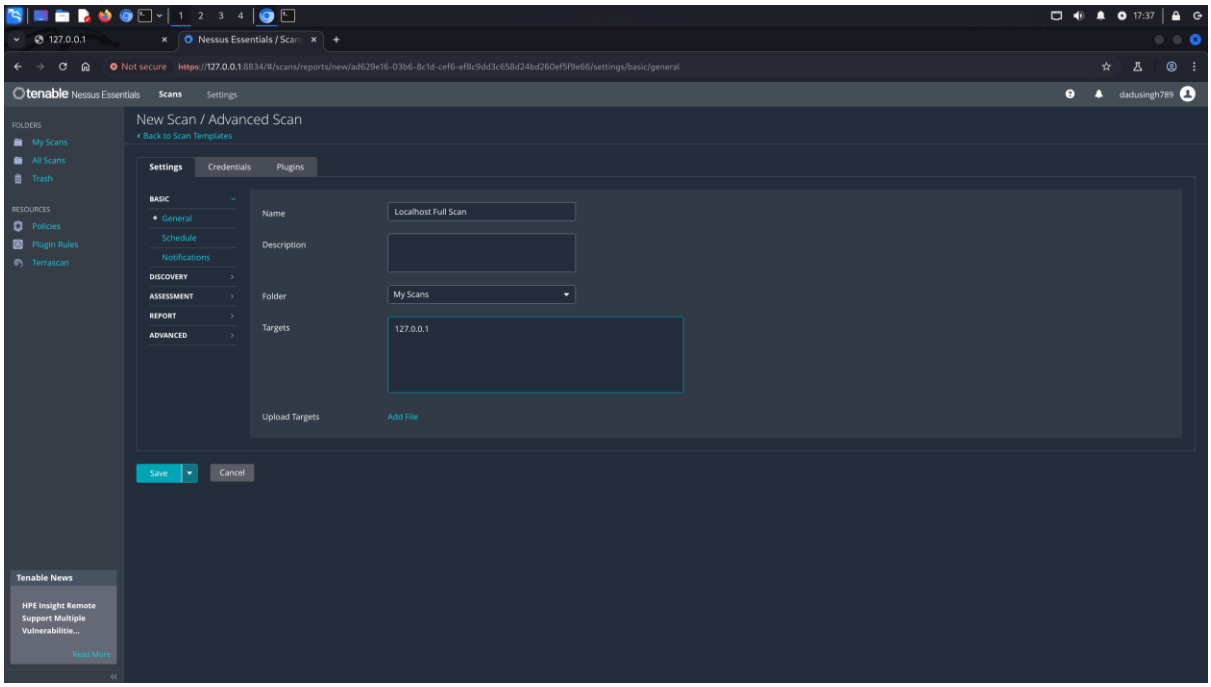


- Go to **Scans → New Scan → Advance Scan**

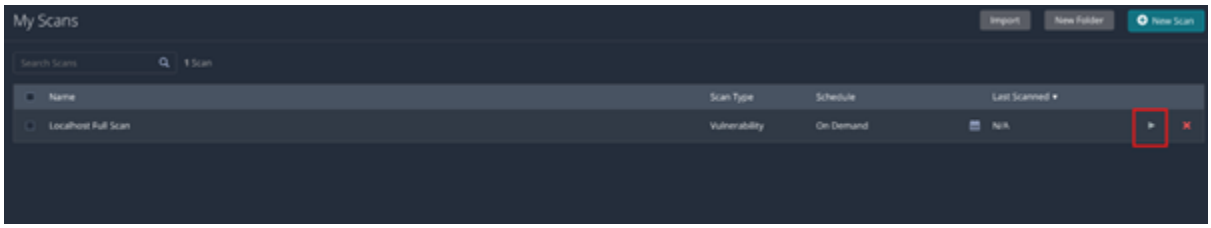


Task 3rd: Vulnerability Scanning with Nessus Essentials

- Name: Localhost Full Scan
- Target: 127.0.0.1
- Save

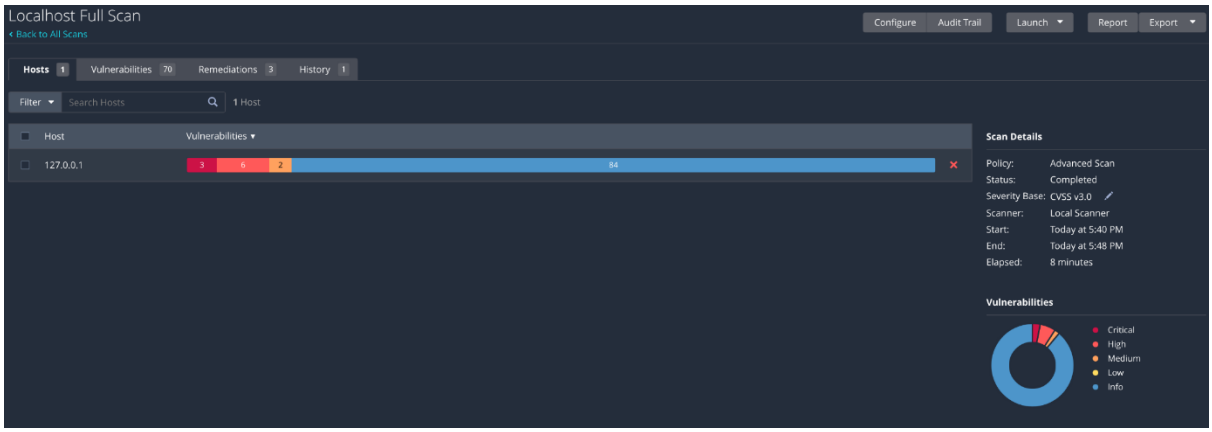


- Launch Scan
    - From the scan list, click the **play (▶)** icon next to your scan
    - Nessus will begin scanning



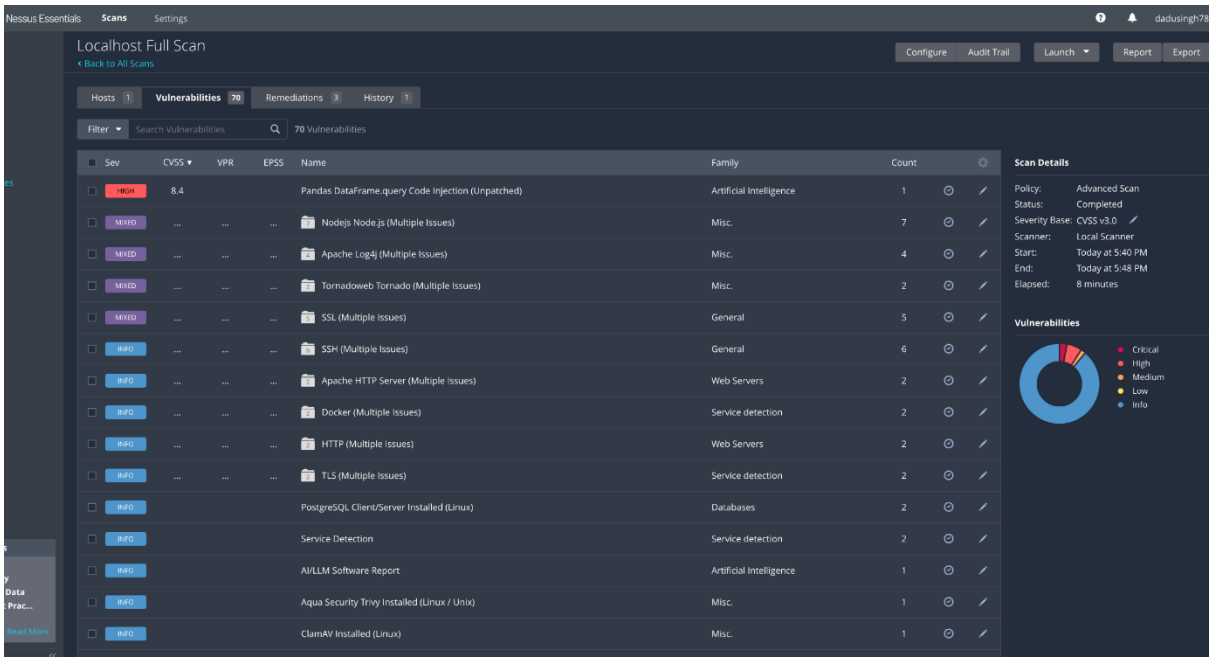
Step 4: Review Scan Results

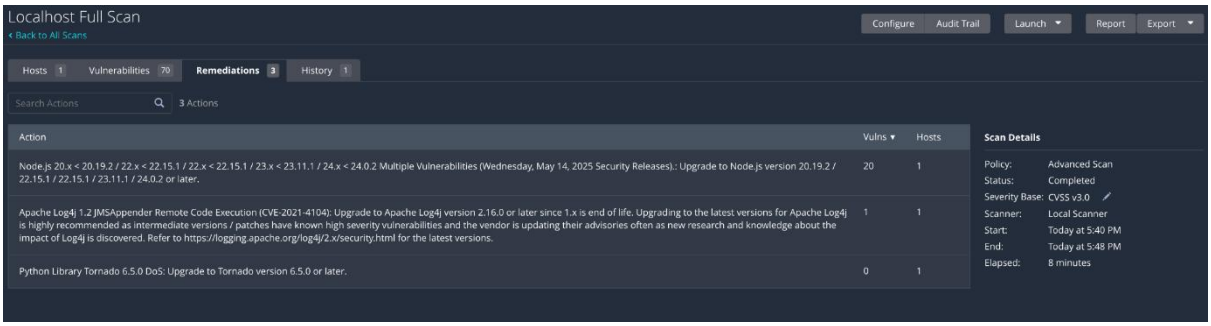
1. **Click the scan after it's finished**



Task 3rd: Vulnerability Scanning with Nessus Essentials

2. **View:**
    - **Summary (number of vulnerabilities)**
    - **Details (CVE, affected software, risk levels)**
    - **Nessus recommendations**

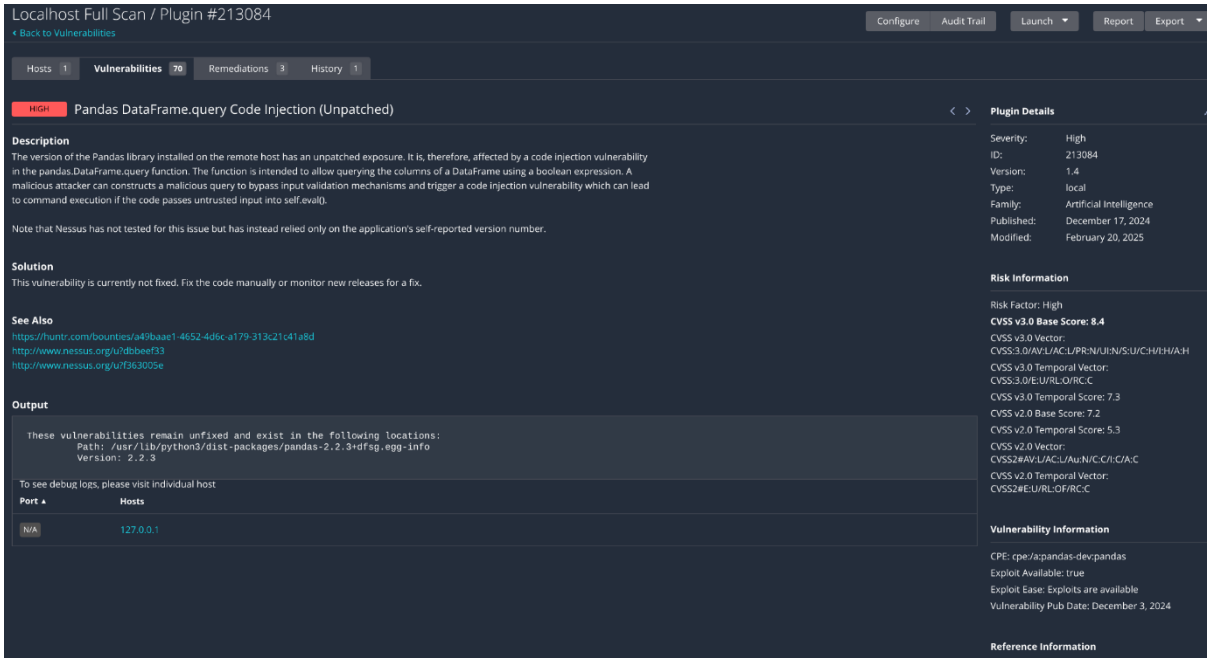




Step 5: Document Critical Vulnerabilities

- **Focus on Critical and High severity issues**
- **Note:**
    - **CVE**
    - **Affected software**
    - **Fix/patch URL**
    - **Risk description**

Task 3rd: Vulnerability Scanning with Nessus Essentials



Task 3rd: Vulnerability Scanning with Nessus Essentials

Nessus Scan Summary: -
Scan Target: 127.0.0.1 (Kali Linux)

Scanner: Tenable Nessus Essentials

OS Detected: Linux Kernel 6.12.25-amd64

Severity	Count
Critical	3
High	5
Medium	2
Low	0
Info	79

Key Critical & High Vulnerabilities

1. Apache Log4j 1.x Multiple Vulnerabilities

- **CVE IDs: CVE-2019-17571, CVE-2020-9488, CVE-2022-23302, CVE-2022-23305, CVE-2022-23307**
- **Severity: High / Critical**
- **CVSS v3 Base Score: 9.8–10.0**
- **Impact: Remote Code Execution (RCE), SMTP MITM, unpatched use**
- **Recommendation: Upgrade to Log4j ≥ 2.17.1, remove JMSAppender**
- **Path: /usr/share/javasnoop/lib/log4j-1.2.16.jar**

2. Node.js Multiple Vulnerabilities

- **Version Detected: 20.11.0**
- **CVEs: CVE-2024-22019, CVE-2024-21892, CVE-2025-23167, CVE-2024-27983**
- **Severity: Critical / High**
- **Impact: DoS, Memory Leak, Path Traversal, Request Smuggling**
- **Recommendation: Upgrade to Node.js ≥ 20.19.2**
- **Path: /usr/lib/python3/dist-packages/playwright/driver/node**

3. Apache Log4j JMSAppender RCE

- **CVE: CVE-2021-4104**
- **Severity: Medium**
- **Impact: RCE when JMSAppender enabled**
- **Recommendation: Upgrade Log4j and remove JMSAppender**

- **Path: /usr/share/javasnoop/lib/log4j-1.2.16.jar**

4. SSL Certificate Cannot Be Trusted

- **Port Affected: 8834**
- **Impact: Self-signed cert not trusted**
- **Severity: Medium**
- **Fix: Replace with CA-issued SSL certificate**

5. Python Tornado DoS Vulnerability

- **CVE: CVE-2025-47287**
- **Version: 6.4.2**
- **Severity: High**
- **Impact: Log flooding via malformed input**
- **Fix: Upgrade to Tornado ≥ 6.5.0**

Detected Installed Components

Software	Version
Apache HTTP Server	2.4.63
Log4j (multiple)	1.2.16, 2.17.1, 2.19.0, 2.24.3
Node.js	20.11.0
Tornado (Python)	6.4.2
Docker	Installed
Trivy	0.62.1
ClamAV	1.4.2
Containerd	1.7.24

Network Information (Enumerated by Nessus)

Interface	Address	Description
eth0	192.168.10.128	Main Kali adapter
docker0	172.17.0.1	Docker bridge
lo	127.0.0.1	Loopback interface
Hostname	kali	Local VM
MAC	00:0c:29:30:8a:91	VMware virtual adapter

Task 3rd: Vulnerability Scanning with Nessus Essentials

Recommendations Summary

Action	Recommendation
Patch Log4j	Upgrade and remove all instances of Log4j 1.x
Patch Node.js	Upgrade to Node.js ≥ 20.19.2
Fix SSL	Use certificates signed by a trusted CA
Audit Docker	Review insecure Dockerfiles and exposed ports
Upgrade Tornado	Use Tornado ≥ 6.5.0 to fix DoS issue
Harden Access	Disable unused services and limit remote exposure

Next Steps

1. **Apply patches and updates**
2. **Regenerate SSL certificates**
3. **Harden firewall and limit exposed services**
4. **Re-run vulnerability scans to confirm remediation**
5. **Maintain updated documentation and compliance logs**