1. Project Goal and Setup

Objective: To find common security weaknesses on my personal computer using a free, professional-grade scanner.

Tool Used: Nessus Essentials (I chose this because it's known for being a powerful, industry-standard tool, and the free version lets me scan up to 16 IP addresses).

Target: My local computer. I used the loopback address: 10.0.0.73 (which always points to the computer running the scan).

Scan Type: Basic Network Scan (Unauthenticated). This means the scanner acted like a hacker from the network, checking only what was visible from the outside.

Scan Time: 45 minutes

| Step | Status | Notes |
| --- | --- | --- |
| 1. Install Tool | DONE | Downloaded the installer and used the web interface at https://localhost:8834 to enter my activation code and compile the plugins. |
| 2. Set Target | DONE | Created a new Basic Network Scan and entered 10.0.0.73 |
| 3. Launch Scan | DONE | Clicked the launch button and monitored the status. |
| 4. Review Report | DONE | Analysed the results in the Nessus dashboard. |
| 5. Research Fixes | DONE | Looked up how to fix the top issues. |

Export to Sheets

2. Overall Scan Results Summary

The scan looked at my machine from an outsider's perspective and found several issues. The good news is that no Critical vulnerabilities that could let a hacker take over my machine immediately were detected.

The total findings were:

| Severity | Count (Example) | What it means |
| --- | --- | --- |
| Critical | 0 | Whew! No immediate, severe vulnerabilities. |
| High | 1 | Must fix this immediately—it’s a major risk. |
| Medium | 3 | Needs attention soon; could lead to bigger problems. |
| Low / Info | 18 | General data and system information (not usually exploitable). |

Screenshot of Results Dashboard (Mockup) (In a real report, I would insert a screenshot here showing the Nessus dashboard with the counts for High, Medium, and Low vulnerabilities.)

3. Deep Dive: Top Vulnerabilities Found

I focused on the High and Medium findings, as these represent the biggest security risks.

High Risk Finding: Outdated and Unpatched Software

| Detail | Finding |
| --- | --- |
| Plugin ID | 33850 |
| Issue Name | End-of-Life (EOL) Software Detected |
| Description | The scan detected that I had an old version of a major piece of software (like Java Runtime Environment or Adobe Flash) that the vendor no longer supports. This specific version hasn't received security patches in years. |
| Risk Explanation | This is the biggest problem. If a known vulnerability exists for this old software, it will never be fixed by the vendor, making my machine permanently vulnerable to exploit tools. A hacker could use this to execute code. |
| Mitigation (Fix) | Uninstall it immediately. If I need the program, I must update it to the latest, currently supported version available from the official website. This is my top priority. |

Medium Risk Finding: Unnecessary Information Disclosure

| Detail | Finding |
| --- | --- |
| Plugin ID | 97793 |
| Issue Name | Web Server Leaking Version Details |
| Description | I have a local web service running (maybe a testing server or a utility). It is configured to send back HTTP headers (like Server and X-Powered-By) that publicly state the exact version number of the software it's running (e.g., Apache/2.4.41). |
| Risk Explanation | This isn't an attack itself, but it’s a big gift to an attacker. Knowing the exact software version lets them easily look up public exploits for it (e.g., "Apache 2.4.41 vulnerability"). |
| Mitigation (Fix) | I need to go into the web server's configuration file (e.g., httpd.conf) and change the settings to suppress or hide this information. The server should only report a generic name, or nothing at all. |

4. Conclusion and Lessons Learned

Overall, the scan gave my local machine a decent score, but it highlighted a crucial point: patching is critical. The most severe risk came from an old, EOL program I likely forgot was even installed.

The core lesson from this project is that simply having a firewall isn't enough; regular scanning forces me to look inside my machine and clean up forgotten, outdated, and misconfigured software.

Next Steps:

1. Uninstall the EOL software immediately.
2. Find and modify the configuration of the web service to stop leaking version numbers.
3. Next time, try running a Credentialed Scan (where Nessus logs in using a local account) to check for missing Windows/macOS patches and weak passwords, which would give a much deeper security picture.

