12

Automated Detection and Security Assessment for Linux-Based Devices

19 - Advay Gujar
25 - Aditya Kareer
29 - Srujana Makarande
34 - Dhruuv Naik

Current Scenario

The security landscape is increasingly complex and time-consuming, with evolving vulnerabilities requiring constant manual intervention. This overwhelming task leads to inefficiency and burnout among security teams.

Need for Proposed System

Collaboration between network tools is crucial for comprehensive security management, as it enables seamless data sharing and enhances threat detection. Integrating tools ensures a more effective and coordinated response to vulnerabilities and attacks.

Problem Statement

Develop an automated vulnerability assessment tool that integrates with multiple security services (e.g., nmap, snort, etc.) and automates their scheduling, analysis and visualization as cron jobs for Linux-based devices.

This system aims to enhance security posture by providing a unified, efficient approach to identifying and addressing vulnerabilities, ensuring comprehensive coverage and streamlined operations.

Scope of the problem

The project aims to automate vulnerability assessments by integrating and scheduling multiple security tools for Ubuntu-based devices, optimizing efficiency and coverage.

Future enhancements may include real-time monitoring capabilities and advanced analytics to further improve vulnerability detection and response.

Literature Summary

Core	Finding/Result	Gap
The paper provides a comprehensive review of automated tools and techniques for vulnerability assessment and management, discussing their capabilities, methodologies, and applications. [1]	● Automated tools significantly improve the efficiency of vulnerability assessments and can provide comprehensive coverage across different systems.	● The paper lacks detailed comparisons between tools and does not address the integration challenges or practical deployment issues in real-world environments.

Literature Summary

Core	Finding/Result	Gap
The paper demonstrates that Vulnerability Assessment and Penetration Testing (VAPT) are effective proactive measures for preventing cyber attacks by identifying and addressing vulnerabilities before they can be exploited. [2]	• VAPT provides a structured approach to detecting and mitigating vulnerabilities, enhancing proactive cyber attack prevention. • The paper outlines various VAPT techniques and tools, both premium and open source, and details the entire lifecycle of the VAPT process.	• The paper may not address the integration of VAPT with other security measures or its effectiveness in different organizational contexts. • Limited discussion on the potential limitations or challenges of implementing VAPT in real-world scenarios.

Literature Summary

Core	Finding/Result	Gap
Net-Nirikshak 1.0 is an automated tool for vulnerability assessment and penetration testing, focusing on SQL Injection vulnerabilities. [3]	● It efficiently detects and exploits SQL Injection vulnerabilities, providing detailed reports and maintaining confidentiality by cleaning up traces.	● The tool's effectiveness against advanced or customized security measures and its ability to handle complex, zero-day vulnerabilities is not fully addressed.

References

[1] D. Luo and J. Wu, "Ranking Online Reviews Based on Consumer Preferences," in 2019 IEEE 19th International Conference on Software Quality, Reliability and Security Companion (QRS-C), Sofia, Bulgaria, 2019, pp. 342-347, doi: 10.1109/QRS-C.2019.00070.

[2] A. Lamba, "Cyber Attack Prevention Using VAPT Tools (Vulnerability Assessment & Penetration Testing)," Cikitusi Journal for Multidisciplinary Research, vol. 1, no. 2, pp. 1-7, July-Dec. 2014. Available: https://ssrn.com/abstract=3516069.

References

[3] S. Shah and B. M. Mehtre, "An automated approach to Vulnerability Assessment and Penetration Testing using Net-Nirikshak 1.0," in 2014 IEEE International Conference on Advanced Communications, Control and Computing Technologies, Ramanathapuram, India, 2014, pp. 707-712, doi: 10.1109/ICACCCT.2014.7019182.

Thank you!