ISSN: 3107-6513

WPT-Analyzer: An Integrated WiFi Penetration Testing Tool for Wireless Network Vulnerability Assessment

Aikeeya Shri I, Mr. R. Karan
Department of Computer Science, Rathinam College of Arts and Science (Autonomous), Coimbatore, Tamil Nadu, India.

Abstract – The widespread adoption of wireless networks across personal, academic, and professional environments has created a critical need for effective WiFi security assessment tools. Existing penetration testing solutions are often complex, fragmented, or inaccessible to non-expert users, limiting their value as educational and practical security resources. This paper presents WPT-Analyzer, an integrated WiFi Penetration Testing Tool designed to provide comprehensive wireless network vulnerability assessment through a unified, modular framework. The proposed system is capable of scanning nearby WiFi networks to collect key parameters including SSID, signal strength, and encryption type; performing deep security analysis to identify weak encryption standards and misconfigurations; executing controlled dictionary-based password strength evaluation; and generating structured reports in HTML, TXT, and JSON formats. Implemented using Python on a Kali Linux environment, WPT-Analyzer emphasizes ethical hacking principles by requiring proper authorization before testing. The system supports both real-time and demo modes, making it suitable for both learners and cybersecurity professionals. Results demonstrate the tool's ability to accurately classify network security levels, flag unsecured connections, and generate clear, actionable vulnerability reports with minimal user effort.

Keywords – **WiFi Penetration Testing, Wireless Network Security, WPA2, WPA3, Packet Analysis, Network Scanning, Ethical Hacking, Vulnerability Assessment, Aircrack-ng, Cybersecurity Education.**

1. Introduction

The proliferation of wireless networks has fundamentally transformed modern communication, providing seamless internet connectivity across homes, educational institutions, and enterprises. However, the convenience of WiFi technology is accompanied by significant security vulnerabilities. Many networks continue to operate with weak passwords, outdated encryption protocols such as WEP, and improperly configured access points, creating opportunities for unauthorized access, data interception, and denial-of-service attacks. Despite the severity of these risks, a considerable portion of users remain unaware of the security posture of their own networks.

WiFi penetration testing is a disciplined methodology for evaluating the security of wireless networks by simulating real-world attack scenarios in a controlled, authorized environment. Through systematic scanning, packet analysis, and vulnerability identification, penetration testing enables organizations and individuals to understand their security gaps before malicious actors can exploit them. The practice is rooted in the principles of ethical hacking, requiring explicit authorization and strict operational boundaries.

Existing penetration testing solutions present notable barriers to adoption. Many professional-grade tools such as Aircrack-ng, Wireshark, and Nmap require advanced technical expertise to configure and operate effectively. Others rely on fragmented toolchains, compelling users to switch between multiple applications to complete a comprehensive security assessment. Furthermore, most existing tools lack user-friendly reporting mechanisms, making it difficult for learners and non-specialist users to interpret and act on their findings.

This paper introduces WPT-Analyzer, an integrated WiFi Penetration Testing Tool that consolidates network scanning, security analysis, controlled password testing, and structured report generation into a single, accessible framework. Developed using Python on Kali Linux, the system adopts a modular architecture that allows each component to be independently developed, tested, and extended. WPT-Analyzer is designed for dual-purpose use: as an educational platform for cybersecurity students and as a practical tool for security analysts conducting authorized wireless assessments. All operations are performed within ethical and legal boundaries, requiring explicit authorization before any network testing is initiated.

2. Related Works

Wireless network security has been an active area of research since the widespread deployment of 802.11 protocols. Early studies identified fundamental weaknesses in the Wired Equivalent Privacy (WEP) encryption standard, demonstrating that it could be compromised within minutes using passive traffic capture and statistical analysis. These findings prompted the development of the Wi-Fi Protected Access (WPA and WPA2) standards, which introduced stronger

encryption through the Temporal Key Integrity Protocol (TKIP) and Advanced Encryption Standard (AES) respectively.

Password strength evaluation in wireless networks has been extensively explored through dictionary attacks and brute-force methodologies. Tools such as Aircrack-ng have become standard references in academic and professional settings for demonstrating the vulnerability of weak passphrases. Research has shown that commonly used dictionary words and simple alphanumeric combinations can be recovered in a matter of hours given sufficient computational resources and captured handshake data.

Network scanning frameworks including Nmap and Kismet have been widely studied for their role in passive and active WiFi reconnaissance. These tools provide detailed enumeration of nearby access points, connected clients, and open ports but are typically intended for experienced users and lack integrated analysis pipelines. Similarly, Wireshark enables granular packet inspection but presents a steep learning curve for novice users.

More recent research has explored the application of machine learning for anomaly detection in wireless traffic, achieving improved accuracy in identifying unauthorized access and rogue access points. However, these approaches generally require substantial datasets and computational resources not available in typical educational environments. Several proposed unified frameworks for WiFi security assessment have demonstrated the value of modular designs but have not sufficiently addressed accessibility for learners or automated reporting for non-expert audiences.

WPT-Analyzer addresses these gaps by providing a complete, automated, and user-friendly penetration testing pipeline that integrates scanning, analysis, and reporting within a single Python-based framework, making comprehensive WiFi security assessment accessible to both students and practitioners.

3. System Design

The design of WPT-Analyzer follows a modular architecture that separates distinct functional responsibilities into independent components, each communicating through well-defined interfaces. This approach reduces coupling between components, improves maintainability, and allows new testing techniques or analysis methods to be integrated without modifying the core system. The primary contributions of the proposed system are:

- Automated detection and enumeration of nearby WiFi networks including hidden SSIDs, with collection of SSID, signal strength, encryption type, and channel information.
- In-depth security analysis of detected networks, including evaluation of encryption standards, identification of open and weakly configured access points, and assessment of connected device activity.
- Controlled dictionary-based password strength testing to identify weak or easily guessable passphrases, operated strictly within authorized and ethical boundaries.
- Automated structured report generation in HTML, TXT, and JSON formats, providing actionable vulnerability summaries for both technical and non-technical stakeholders.
- Support for both real-time and demo modes, enabling use without specialized hardware for educational and simulation purposes.

3.1 Network Scanning Module

The Network Scanning Module constitutes the data acquisition layer of the WPT-Analyzer framework. It uses the system's wireless network adapter — configured in monitor mode — to passively detect and enumerate nearby access points. For each discovered network, the module collects SSID, signal strength in dBm, encryption type (Open, WPA, WPA2, WPA3), operating channel, and BSSID. The module continuously refreshes the list of detected networks to ensure data currency and is capable of identifying hidden SSIDs. Output is presented in a structured tabular format via the command-line interface, providing an at-a-glance overview of the wireless environment.

3.2 Analysis Module

The Analysis Module processes the data collected by the scanning module to evaluate the security posture of each detected network. It classifies networks by encryption strength, flagging open networks as high-risk, WPA/WPA2 networks with potentially weak passwords as moderate-risk, and WPA3 networks as lower-risk pending further configuration review. The module also performs packet capture using Scapy-based routines to analyze traffic patterns, identify connected devices, and detect potential rogue access points or spoofed SSIDs. Automated vulnerability detection reduces manual effort and provides consistent, reproducible analysis results. The module generates intermediate structured data passed to both the password testing and reporting modules.

3.3 Password Testing Module

The Password Testing Module evaluates the strength of WiFi passwords using controlled dictionary-based simulation techniques. The module operates strictly

within authorized environments and uses predefined wordlists to test whether a network's passphrase can be recovered through common dictionary attack methods. This approach mirrors real-world attack techniques while remaining fully contained within the testing environment, with no packets transmitted to external systems. Results indicate password strength — strong, moderate, or weak — and provide users with practical guidance on improving passphrase security. The module is automated and requires minimal user configuration, making it accessible for learners without deep technical expertise.

3.4 Reporting Module

The Reporting Module aggregates results from all preceding modules and produces structured, readable outputs in three formats: HTML for visually organized presentation, TXT for lightweight text documentation, and JSON for programmatic processing and integration with other security tools. Each report includes a network summary section, a detailed vulnerability analysis section, password testing outcomes, and a security recommendations section. Reports are automatically generated upon completion of analysis, stored in a designated output directory, and can be shared, archived, or submitted for academic and professional purposes. The structured format ensures that both technical and non-technical readers can identify key security issues and required actions.

3.5 User Interface Module

The User Interface Module manages interaction between the user and the system through a clean, intuitive command-line interface. Users launch the tool by executing the main program file, after which the interface guides them through network selection, analysis configuration, and output options. Outputs are formatted using structured tables and clearly labeled sections to improve readability. The interface also supports a demo mode in which pre-loaded simulated network data is used, enabling users to explore the tool's capabilities without requiring a specialized wireless adapter supporting monitor mode. This dual-mode design significantly broadens the accessibility of the system for educational use.

4. System Architecture

The WPT-Analyzer system follows a pipeline-based modular architecture with a clearly defined data flow from input acquisition to output generation. The user interacts with the system through the Command-Line Interface, which serves as the entry and control point for all operations. Upon execution, the Network Scanning Module acquires raw wireless network data from the environment using the wireless adapter configured in monitor mode. External data sources comprise live network traffic and, in demo mode, pre-loaded simulated datasets.

Collected data flows into the Analysis Module, where security evaluation and packet inspection are performed. Analysis results are passed concurrently to the Password Testing Module and the Reporting Module. The Password Testing Module applies dictionary-based assessment techniques to evaluate passphrase strength and returns classified results to the reporting pipeline. All processed data is organized by the Reporting Module into the final output formats and persisted in the output storage directory. Throughout execution, the User Interface Module renders progress updates, intermediate results, and final outputs to the terminal in a structured, readable format.

This architecture ensures that individual modules can be independently tested, updated, or replaced without disrupting the overall system workflow. The modular boundaries also facilitate future extension, such as the integration of machine learning-based anomaly detection or a graphical user interface layer.

5. Implementation

WPT-Analyzer is implemented primarily in Python 3.x, running on Kali Linux — a Linux distribution specifically optimized for penetration testing and network security operations. Python was selected for its extensive library ecosystem, strong support for network programming, and widespread use in cybersecurity education and practice. Bash scripting is employed for system-level operations including enabling monitor mode on the wireless adapter and invoking external tools where necessary.

Core packet capture and network analysis functionality is provided by the Scapy library, which enables low-level manipulation and inspection of wireless frames. Network enumeration leverages Python's socket module alongside Scapy-based scanning routines. Data processing and feature organization are handled using standard Python data structures, while report generation uses built-in file I/O and string formatting for TXT and JSON outputs, with HTML reports generated through template-based rendering. Integration with Aircrack-ng is supported for advanced password strength evaluation workflows.

The following sample code illustrates the core network scanning and analysis workflow of the WPT-Analyzer main module:

6. Results and Output Analysis

The WPT-Analyzer was evaluated against a set of simulated wireless network environments representing a

range of security configurations, from open unsecured networks to WPA3-protected enterprise access points. The tool successfully completed all phases of the assessment pipeline — scanning, analysis, password testing, and report generation — across all test scenarios.

6.1 Network Scanning Output

During scanning, the system detected and enumerated three representative networks within the test environment. Results were presented in a structured tabular format via the CLI, showing SSID, signal strength, and encryption type for each discovered network. The tool correctly identified the open Guest_Network as carrying elevated security risk, the WPA2 Home_Network as requiring password strength evaluation, and the WPA3 Office_WiFi as carrying lower baseline risk. Hidden SSIDs were also successfully detected during extended scanning.

6.2 Security Analysis and Password Testing

The Analysis Module processed packet captures from each test network, confirming encryption classifications and identifying the open network as vulnerable to passive eavesdropping and unauthorized access. For WPA2-protected networks, the Password Testing Module was applied using a standard dictionary wordlist. Weak passphrases commonly derived from dictionary words were successfully identified, validating the module's effectiveness in simulating real-world credential recovery scenarios. WPA3 networks demonstrated resilience against the dictionary-based testing methodology due to their Simultaneous Authentication of Equals (SAE) handshake mechanism.

Table 1. WPT-Analyzer Detection and Analysis Summary

Network SSID	Encryption	Signal (dBm)	Risk Level	Password Strength
Home_Network	WPA2	-45	Moderate	Weak (Recoverable)
Office_WiFi	WPA3	-60	Low	Strong (SAE Protected)
Guest_Network	Open	-70	High	None / No Auth

6.3 Report Generation

The Reporting Module successfully generated assessment reports in all three supported formats (HTML, TXT, and JSON) following completion of analysis. HTML reports rendered a clearly structured multi-section layout with network summaries, vulnerability highlights, and security recommendations. JSON outputs provided machine-readable structured data suitable for integration with external security management systems. All reports were automatically saved to the designated output directory, with generation requiring no additional user input beyond the initial scan configuration.

7. Objective and Scope

The primary objective of this research is to design and implement an accessible, integrated, and ethically grounded WiFi penetration testing tool that consolidates the key stages of wireless security assessment within a single Python-based framework. WPT-Analyzer aims to reduce the complexity and fragmentation inherent in existing multi-tool penetration testing workflows, making comprehensive WiFi security assessment achievable by users with varying levels of technical expertise.

The scope of this work encompasses the design, implementation, and evaluation of the WPT-Analyzer system across four functional modules: network scanning, security analysis, password strength testing, and structured report generation. The system targets WiFi networks utilizing Open, WPA, WPA2, and WPA3 security configurations in both indoor and campus network environments. The tool is designed primarily for educational use by cybersecurity students, supplemented by practical application in authorized security assessments. Future scope includes integration of machine learning-based anomaly detection, support for a graphical user interface, and extension to multi-network simultaneous testing.

8. Security and Ethical Considerations

WPT-Analyzer is developed strictly for educational, research, and authorized security assessment purposes. The tool must only be used on WiFi networks for which the user holds explicit written authorization from the network owner or administrator. Unauthorized use of any component of this system against networks without proper consent constitutes a violation of applicable computer crime and data protection laws, including the Information Technology Act in India and equivalent legislation in other jurisdictions.

All packet capture and password testing operations are performed locally within the authorized testing environment. The system does not transmit captured data to external servers, does not store personally identifiable user credentials, and does not implement any offensive capabilities beyond controlled dictionary-based password evaluation. The tool is designed to reinforce ethical hacking principles: all activities must be conducted in a safe, controlled environment, with full transparency and documented authorization. Users are strongly encouraged to obtain formal ethical hacking certifications such as CEH or OSCP before conducting authorized assessments in professional contexts.

9. Conclusion

This paper presented WPT-Analyzer, an integrated and accessible WiFi Penetration Testing Tool designed to address the critical limitations of fragmented, complex, and inaccessible existing wireless security assessment solutions. By consolidating network scanning, security analysis, controlled password testing, and structured report generation within a modular Python-based framework, WPT-Analyzer provides a comprehensive and user-friendly platform for wireless vulnerability assessment.

Evaluation results demonstrated the system's ability to accurately classify network security levels across Open, WPA2, and WPA3 configurations, identify vulnerable passphrases through dictionary-based testing, and produce detailed, actionable reports in multiple formats. The tool's support for both real-time and demo modes makes it equally suitable for practical cybersecurity work and educational use, significantly broadening its accessibility compared to existing professional-grade alternatives.

Future development of WPT-Analyzer will focus on the integration of machine learning-based anomaly detection for more intelligent threat identification, the addition of a graphical user interface for improved accessibility, support for multi-network simultaneous testing, and the implementation of encrypted logging and role-based access control to further strengthen the tool's security profile. These enhancements will advance WPT-Analyzer toward a production-ready, comprehensive wireless security assessment platform suitable for both academic and professional deployment.

10. References

[1] Fluhrer, S., Mantin, I., and Shamir, A., "Weaknesses in the key scheduling algorithm of RC4," Proceedings of the 8th Annual International Workshop on Selected Areas in Cryptography (SAC), Lecture Notes in Computer Science, Springer, 2001, pp. 1–24.

[2] Stubblefield, A., Ioannidis, J., and Rubin, A.D., "A key recovery attack on the 802.11b wired equivalent privacy protocol (WEP)," ACM Transactions on Information and System Security, vol. 7, no. 2, 2004, pp. 319–332.

[3] Tews, E. and Beck, M., "Practical attacks against WEP and WPA," Proceedings of the 2nd ACM Conference on Wireless Network Security (WiSec), 2009, pp. 79–88.

[4] Vanhoef, M. and Piessens, F., "Key reinstallation attacks: Forcing nonce reuse in WPA2," Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security (CCS), 2017, pp. 1313–1328.

[5] IEEE Standard 802.11i-2004: Amendment 6: Medium Access Control (MAC) Security Enhancements, IEEE, 2004.

[6] Khasawneh, M. and Kajman, I., "Performance comparison of WEP, WPA and WPA2 in wireless networks," International Journal of Computer Science and Information Security, vol. 10, no. 8, 2012.

[7] Nmap Security Scanner, G. Lyon, "Nmap Network Scanning: The Official Nmap Project Guide," Insecure.com LLC, 2009.

[8] Kershaw, M., "Kismet wireless network detector, sniffer and IDS," édition en ligne: http://www.kismetwireless.net, 2002.

[9] Martineau, A. and Ramsey, T., "Aircrack-ng: An introduction to the aircrack-ng suite," Hakin9 Magazine, vol. 3, no. 2, 2008.

[10] Orebaugh, A., Ramirez, G., and Beale, J., Wireshark and Ethereal Network Protocol Analyzer Toolkit, Syngress Publishing, 2007.

[11] Komanduri, S. and Hutchings, D.R., "Of passwords and people: Measuring the effect of password-composition policies," Proceedings of the ACM CHI Conference on Human Factors in Computing Systems, 2011, pp. 2595–2604.

[12] Sheng, S., Broderick, L., Koranda, C.A., and Hyland, J.J., "Why Johnny still can't encrypt: Evaluating the usability of email encryption software," Proceedings of the Symposium on Usable Privacy and Security (SOUPS), 2006.

[13] Bittau, A., Handley, M., and Lackey, J., "The final nail in WEP's coffin," Proceedings of the IEEE Symposium on Security and Privacy, 2006, pp. 386–400.

[14] Lashkari, A.H., Danesh, M.M.S., and Samadi, B., "A survey on wireless security protocols (WEP, WPA and WPA2/802.11i)," Proceedings of the 2nd IEEE International Conference on Computer Science and Information Technology, 2009.

[15] Harkins, D., "Simultaneous authentication of equals: A secure, password-based key exchange for mesh networks," Proceedings of the 2nd International Conference on Sensor Technologies and Applications, 2008, pp. 839–844.

11. Acknowledgment

ISSN: **3107-6513**

This article is the outcome of the research work carried out in the Department of Computer Science, Rathinam College of Arts and Science (Autonomous), Coimbatore. The author expresses sincere gratitude to the Department for providing the necessary resources, laboratory facilities, and academic support required for the successful completion of this work. Special thanks are extended to Mr. R. Karan, MCA, Assistant Professor, Department of Information Technology, for his invaluable guidance, mentorship, and constructive feedback throughout the project. The author also acknowledges the encouragement and support of Dr. T. Velumani, Head of the Department of Information Technology, and the class advisor Dr. M. Ramaraj, whose inestimable support made this work possible. Heartfelt gratitude is expressed to the management of Rathinam College of Arts and Science — Dr. Madan A Sendhil, Chairman, and Mrs. Shima Sendhil, Secretary — for providing an excellent academic environment. This research was carried out as part of the B.Sc. (Digital and Cyber Forensic Science) program, and the author dedicates this work to her parents and friends for their unwavering moral support and encouragement.