Spectrum of Engineering Sciences
ISSN (e) 3007-3138 (p) 3007-312X
Volume 4, Issue 1, 2026

AEROSTRIKE: A REAL-TIME AI-DRIVEN FRAMEWORK FOR WIRELESS NETWORK THREAT DETECTION AND EXPLOITATION

Zain Ali Shahid[1], Sundas Amin[2], Ali Sufyan[*3], Adnan Majeed[4], Attaullah[5]

[1,2,3]Department of Information Security, The Islamia University of Bahawalpur, Bahawalpur, Pakistan
*[*3,5]Department of Information and Communication Engineering, The Islamia University of Bahawalpur, Bahawalpur, Pakistan*

[*3]ali.sufyan@iub.edu.pk

DOI: https://doi.org/10.5281/zenodo.18384245

Keywords
Wireless Security, Penetration Testing, AI Vulnerability Assessment, Automated Exploitation, Anomaly Detection

Article History
Received: 02 December 2025
Accepted: 12 January 2026
Published: 27 January 2026

Copyright @Author
Corresponding Author: *
Ali Sufyan

Abstract

The rapid expansion of wireless are expending rapidly connecting millions of devices, homes, and industries globally. However, this immense growth has brought with it a significant security crisis as a traditional methods for finding network weaknesses are too slow and difficult to use. Most security experts still rely on manual tools which require typing many commands, and that can often lead to mistakes and let dangerous things through. We introduce Aerostrike, a user-friendly system that automate the entire security scanning process using Artificial Intelligence (AI). Instead of using different tools for different tasks, Aerostrike combines network scanning, threat detection, and password testing into single smart platform. It uses a machine learning method called Random Forest to accurately identify different types of devices and traffic, while other method called Isolation Forest automatically detect unusual behavior that may indicates an attack. By automatically handling these complex tasks, Aerostrike helps security teams find threats and vulnerabilities much faster and with lower error. Our tests show that this system is more reliable than manual methods and helps users quickly solve problems with clear, AI driven advice.

I. INTRODUCTION

The foundation of modern digital environments is built upon wireless technology, serving as the critical infrastructure for everything from autonomous vehicles to the Internet of Things (IoT) [1] [2]. The further progress of the use of 5G wireless and Wi-Fi 7, such a hyper-connected space is extremely high- speed, yet it is also very vulnerable to security threats [3]. Mobile networks create an enormous scope of attack since data can pass through wireless networks whether they are intercepted or attacked at secure distances [4]. In this dynamic nature, the use of manual penetration testing method of traditional security approaches is no longer effective in this environment as professionals are finding it difficult to cope up with the sheer number of devices and the rate of contemporary exploitation [5]. The emerging security approaches are mainly based on a disjointed ecosystem of old fashioned tools that fits the last century of computing. The utilities that are covered by the industry standards are often designed as isolated silos with practitioners required to use their own human middleware by chaining command-line tools together to scan, log and crack them [6]. It is a fragmented workflow, which is man-sensitive, and in most cases lacks the skill to prioritize threats. Also, these tools produce gigabytes of raw data and no context that is not a significant weakness when an automated script is used by attackers to take advantage of vulnerabilities in milliseconds [7]. In a bid to correct these systemic failures, the introduction of Artificial Intelligence (AI) and Machine Learning (ML) systems into the offensive security systems has become a functional requirement [8]. AI promises such features as the processing of data sets faster than people are capable of, and algorithms such as the Random Forest and Isolation Forest can automatically differentiate between the legitimate traffic flow and bad anomalies [9] [10]. Nevertheless, there is still a large void in the market in the unified structures that can facilitate the gap between defensive monitoring and offensive exploitation [11]. The proposed paper Aerostrike is a total, AI- powered model that aims at automating the whole lifecycle of wireless penetration testing. Breaking the constraints of manual action, Aerostrike is hacked into machine learning onto the scanning and exploitation cycle. Our system uses the Random Forest (RF) algorithms to achieve high-fidelity classification of protocols and the use of the Isolation Forest to identify the anomalies of the Isolation Forest that are caused by the zero-days protocols and which are not recognized by older signature based tools. This research will be based initially on merging reconnaissance and exploitations into a single cohesive GUI, or real-time AI-driven risk scoring, or even a smart engine in reporting which changes technical findings into convenient remediation strategies.

The primary objectives of this study are:

- Automated scanning and risk assessment engine for identification and privatization of vulnerable wireless networks.
- A comprehensive attack framework that support multiple Wi-Fi attack techniques, including WPS exploitation, WPA handshake capturing, and WEP cracking.
- Real-time network traffic analysis module that employs machine learning algorithms to detect anomalies.

The rest of the article is categorized in many sections. In section 2, the related work is summarized. In section 3, methodology which examine the user interface, various data storage methods and performance optimization is presented while in section 4, we discuss the experimental configuration, performance outcomes, and resource utilization. The limitations and future work have been discussed in Section 5. Finally, the article is concluded in Section 6.

II. RELATED WORK

The field of wireless security is also experiencing the paradigm shift due to the convergence of the next level of connectivity (5G, Wi-Fi 7) and rapid development of artificial intelligence. This review will summarize the most recent findings from 2021 to 2025 in order to contextualize Aerostrike by classifying the state of the art into the developing threat scenario, the limitations of conventional testing tools, machine learning-based intrusion detection, and autonomous penetration testing agents. The advent of wireless networks has served as a cause of a complex array of security challenges due to the rapid growth of the networks. The current literature highlights the fact that networks have been found to be more vulnerable to attacks with the increased presence in the critical infrastructure. According to [12], the IoT devices spread in smart cities have exposed the networks to the threat of unauthorized entry, network breaches, and Denial of Service (DoS) attacks exponentially due to the growth of the devices in the environment. This is because such devices do not have strong security measures, and therefore provide easy access to the attackers [13]. Equally, a study of the Industrial Internet of things (IIoT) notes that the sheer numbers of connected gadgets which is estimated to go into billions create traffic blocks where malicious actions can go unnoticed without much efforts [14]. The cellular-Wi-Fi network separation is also disappearing. The study presented in [15] is concerned with the security issues that arise due to the integration of 5G and Wi-Fi access networks, whereby traditional raw packet inspection cannot keep up with such high transmission rates of these next-generation networks. Moreover, even with developed encryption, basic flaws are still on-hand. Although WPA3 was meant to solve the weaknesses of WPA2. In [16], the vulnerability of WPA3 networks to downgrade attacks, particularly during transition states, is demonstrated. Hence, this highlights the need for tools capable of actively testing for and identifying configuration-level vulnerabilities before they can be exploited. This can also be leniently supported by [17], who claim that automated testing structures are necessary to verify the implementations with regard to the side-channel attacks. Literature published between 2021 and 2025 includes commentary on earlier manual penetration testing toolkits, highlighting their usage, limitations, and relevance in contemporary security assessments. Although such tools as Aircrack- ng, Kismet and Wireshark are essential, they are often doubted when it comes to the active environment. One such theme is the lack of efficiency due to the fragmenting nature of tools, practitioners combine many Command Line Interface (CLI) tools, which introduces latency and human error [18] [19]. According to the state pen-testing report [20], manual testing in the present system has been greatly changed to automated verification as it is too slow to track and control the vulnerability. In addition, the conventional vulnerability scanning can usually overwhelm security teams with messages instead of Information to act upon. According to the 2025 annual insights report by Horizon3.ai [21], in wireless environments comprising hundreds of access points, a human tester cannot physically or cognitively assess each target simultaneously; therefore, automated prioritization of targets is necessary. Besides, according to the latest surveys of the penetration testing tool, it has been observed that legacy systems do not keep up with the agentic character of current attacks where attackers can adjust their campaigns on a real-time basis with the assistance of AI [22]. Zhou et al. [23] also showed that automated scripting can significantly outperform manual execution in industrial IoT environments; however, it still lacks the adaptive decision- making capabilities of AI-driven approaches. Moreover, the contemporary software stacks located on the modern wireless gateways consist of highly complex code, and the conventional source code representations combined with the current state of the software analysis tools which cannot be used to identify the vulnerability of the systems [24]. Machine Learning (ML) has become an essential element of the current security architectures in order to overcome such scalability problems. Random Forest (RF) has become a powerful and important module of network traffic classification because of its ability to be interpreted and its effectiveness. As demonstrated in the article [25], an optimized RF classifier may reach a testing accuracy of over 93 percent to detect Distributed Denial of Service (DDoS) attack in the IoT networks due to its resistance to distorted and missing information. RF is continuously ranked higher in comparative studies and computational efficiency over other shallow learning models [26], [27]. As an example, Wang et al. [28] observed that RF had 99% F1-score when classifying encrypted traffic between VPN, which is better than the Support Vector Machines (SVM). Isolation Forest (iForest) is extensively stated as the state-of-the-art when it comes to the detection of unknown or zero-day-anomalies. The benefits of an isolation-based detector of anomalies are highlighted in [29], has a linear time complexity, and can proceed to isolate anomalies without profiling the normal behavior. The recent implementations have shown the effectiveness of iForest in detecting signal spoofing in wireless sensor networks [30] and jamming attacks which do not follow typical metric distributions. Certain studies [31] have used iForest to identify certain artifacts like anomalies of authentication packets, the subtle deviations in signal strength (RSSI) [32] pattern and abnormal inter-arrival time of frames. Although Deep Learning (DL) classifiers such as Convolutional Neural Network (CNNs) and Long Short-Term Memory (LSTMs) are highly accurate [33], [34], they usually consume a significant amount of computational power [35], so lightweight ML models, such as RF and iForest, should be used in a real-time, edge-deployed auditing device. Moreover, Zhao et al. [36] recently suggested Mallows like criterion as a means of optimizing RF due to anomaly detection, with which it continues to cement its application in the context of an imbalanced data. The most advanced studies are on the transition between the uses of automated scripts to the solution of autonomous agents, which have the ability to involve in reasoning. Reinforcement Learning (RL) is transforming the field of automated penetration testing by enabling the agents to learn the best strategy of attack by trial and error [37]. The most recent literature by Lopez-Montero et al. [38] and Li [39] explain how the RL agents may find attack paths within web applications and smart grids that may escape the attention of human specialists. In addition to that, reporting and remediation are being reinvented by the combination of Generative AI (GenAI) and Large Language Models (LLMs). Ferrag et al. [40] overview the applications of the LLM to cybersecurity and observe that they analyze threat data to formulate human-readable reports. Tools such as PentestGPT [41], AutoPentester [42] and Pentest-R1 [43] are based on a model that interprets scan findings, makes reasoning of attack chains, and proposes the specific command of the exploit. Nevertheless, it is also possible to come across the warnings on the literature about the existence of the so-called hallucinations and the necessity of human check [44], indicating that GenAI was created to support human judgment and not to overpower it. According to recent reports by Marketsand Markets [45] and Acuvity [46], these AI-driven security tools are expected to increase. However, they also state that there is a danger of so-called shadow AI and that requires proper governance. All these studies [47], [48] confirm that the future of wireless security is integrated frameworks which incorporate offensive capability and intelligent, ML-driven analysis which supports the architectural design of Aerostrike.

III. METHODOLOGY

Aerostrike is an architectural philosophy based on a modular layered design paradigm to guarantee high extensibility and scalability with minimum latency usually caused by mono- lithic security tools. This framework takes the whole lifecycle of penetration testing, including first-time reconnaissance, through to post-exploitation analysis, and puts it all in the same environment, a single asynchronous execution environment, unlike the functionality of a traditional command-line utility that works independently. The system is organically split into three separate abstraction layers namely the user interface layer, the processing engine layer and the output generation layer as shown in Figure 1.

The user interface layer is the command and control interface, which is achieved through the use of GUI Manager, which is used to isolate complex CLI commands into a visual flow. This layer can be configured, whereby, operators can set configuration parameters of attack and scan time whereas the real-time status monitor gives a real time feedback of monitor traffic metrics and the status of the attack.

Below this is the processing engine layer which is the computing center of the structure. The most significant logic modules are located in this layer; they are network discovery module to scan the entry points of wireless, Attack execution module to exploit the entry points automatically and traffic monitoring module to inspect deep packet. Lastly, there is the output generation layer which takes care of data persistence and reporting. It makes sure that the data of the assessment is noted as presently approved through the data logger as well as transferred into actionable intelligence, by using the report generator.

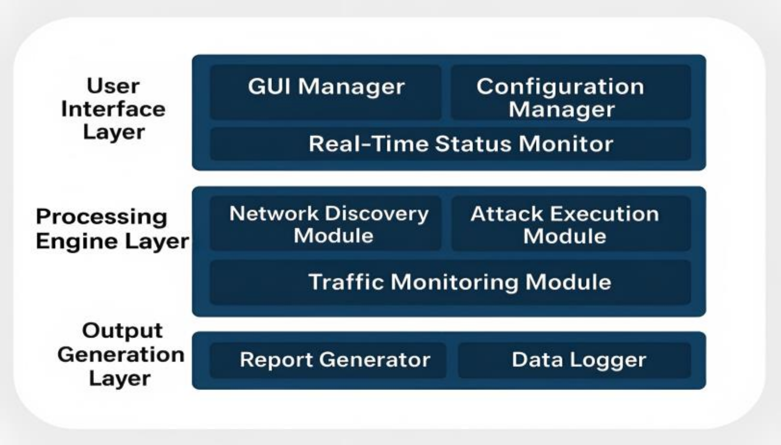
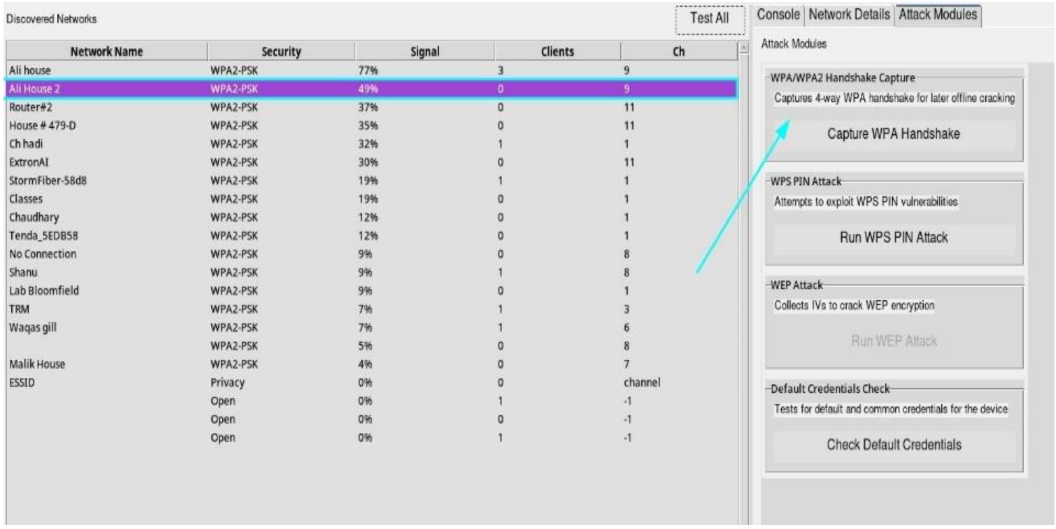


Fig. 1. The Aerostrike framework's system architecture.

A. Core Functional Modules

The functions of the framework are performed by three major modules that are geared towards automation of the sequential stages of a penetration test. Reconnaissance and network discovery the evaluation phase starts with the network discovery module which will be involved in passive scanning of the wireless spectrum. Instead of actively interrogating devices and thereby activating intrusion detection systems (IDS), Aerostrike passively gathers broadcast frames and in the process fetches Service Set Identifiers (SSID), BSSID, encryption scheme, and Relative Signal Strength Indicators (RSSI). To ensure operation continuity, the discovery interface uses an auto-refresh technology which is used to dynamically populate the detected networks without manual user intervention and this reduced the chances of the absence of temporary access points. The outcome of this step is represented in Figure 2.



Network Name	Security	Signal	Clients	Ch
Ali house	WPA2-PSK	77%	3	9
Ali House 2	WPA2-PSK	49%	0	9
Router#2	WPA2-PSK	37%	0	11
House # 479-D	WPA2-PSK	35%	0	11
Ch hadi	WPA2-PSK	32%	1	1
ExtronAI	WPA2-PSK	30%	0	11
StormFiber-58d8	WPA2-PSK	19%	1	1
Classes	WPA2-PSK	19%	0	1
Chaudhary	WPA2-PSK	12%	0	1
Tenda_5EDB58	WPA2-PSK	12%	0	1
No Connection	WPA2-PSK	9%	0	8
Shanu	WPA2-PSK	9%	1	8
Lab Bloomfield	WPA2-PSK	9%	0	1
TRM	WPA2-PSK	7%	1	3
Waqas gill	WPA2-PSK	7%	1	6
	WPA2-PSK	5%	0	8
Malik House	WPA2-PSK	4%	0	7
ESSID	Privacy	0%	0	channel
	Open	0%	1	-1
	Open	0%	0	-1
	Open	0%	1	-1

Captures 4-way WPA handshake for later offline cracking

Attempts to exploit WPS PIN vulnerabilities

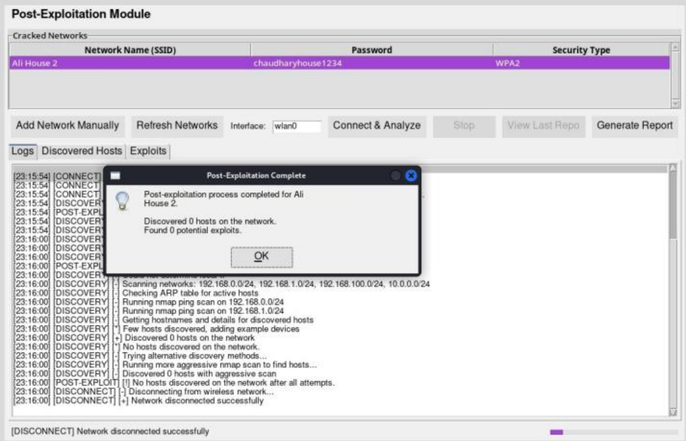
Collects IVs to crack WEP encryption

Tests for default and common credentials for the device

Fig. 2. Network Discovery Results Ordered by Signal Strength.

Automated attack orchestration to eliminate the inefficiency and human error found with tool chaining. Aerostrike, uses attack execution module which can be orchestrated automatically. When the operation sequence named Test All is activated, the system attempts to perform a test on each of the networks in the target list, whereby the priorities given by the system are determined by the signal strength and security posture of the networks. The engine completes one after another with a series of attacks that are specific to the vectors, such as WPA/WPA2 handshake capture to attacks based on an offline dictionary, WPS PIN, and WEP cracking using the Initialization Vector (IV), respectively. This algorithm can guarantee a standard and repeatable evaluation process. The Big Data automated sequence is presented in Figure 3.

The post-exploitation enumeration once a breach has been successful the system moves to the post-exploitation module to determine the extent of the compromise. The framework will automatically connect to the network that has been compromised and map it internally. This is done by recognizing the active hosts, excising the metadata that is vendor specific and defining the range of IP to build a topology of the internal network. This step offers essential transparency within the mist of linked apparatus and legitimizes the degree of the susceptibility, as displayed in Figure 4.



Post-Exploitation Module

Cracked Networks

Network Name (SSID)	Password	Security Type
Ali House 2	chaudharyhouse1234	WPA2

Interface: wlan0

Post-Exploitation Complete

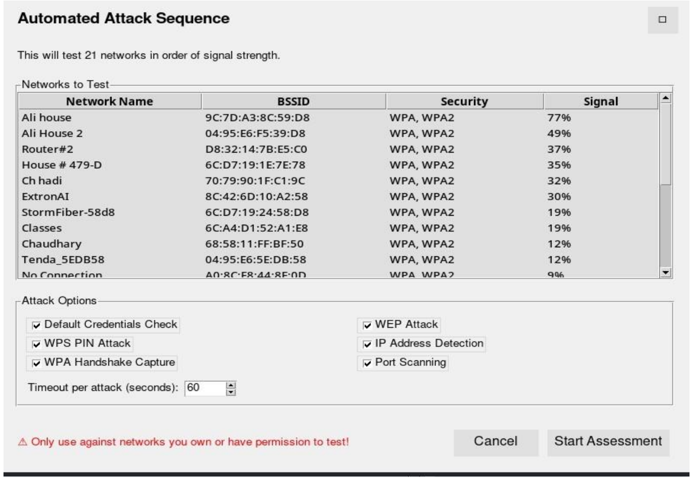
Post-exploitation process completed for Ali House 2.

Discovered 0 hosts on the network.
Found 0 potential exploits.

```
[23:15:54] [CONNECT]
[23:15:54] [CONNECT]
[23:15:54] [CONNECT]
[23:15:54] [DISCOVER
[23:15:54] [POST-EXPL
[23:15:54] [DISCOVER
[23:15:54] [DISCOVER
[23:16:00] [DISCOVER
[23:16:00] [DISCOVER
[23:16:00] [DISCOVER
[23:16:00] [POST-EXPL
[23:16:00] [DISCOVER
[23:16:00] [DISCOVERY] [-] Scanning networks: 192.168.0.0/24, 192.168.1.0/24, 192.168.100.0/24, 10.0.0.0/24
[23:16:00] [DISCOVERY] [-] Checking ARP table for active hosts
[23:16:00] [DISCOVERY] [-] Running nmap ping scan on 192.168.0.0/24
[23:16:00] [DISCOVERY] [-] Running nmap ping scan on 192.168.1.0/24
[23:16:00] [DISCOVERY] [-] Getting hostnames and details for discovered hosts
[23:16:00] [DISCOVERY] [*] Few hosts discovered, adding example devices
[23:16:00] [DISCOVERY] [+] Discovered 0 hosts on the network
[23:16:00] [DISCOVERY] [*] No hosts discovered on the network.
[23:16:00] [DISCOVERY] [-] Trying alternative discovery methods...
[23:16:00] [DISCOVERY] [-] Running more aggressive nmap scan to find hosts...
[23:16:00] [DISCOVERY] [-] Discovered 0 hosts with aggressive scan
[23:16:00] [POST-EXPLOIT] [!] No hosts discovered on the network after all attempts.
[23:16:00] [DISCONNECT] [-] Disconnecting from wireless network...
[23:16:00] [DISCONNECT] [+] Network disconnected successfully
```

[DISCONNECT] Network disconnected successfully

Fig. 3. Automated Attack Sequencer Performing Sequential Wireless Exploitation Without User Interaction.



Automated Attack Sequence

This will test 21 networks in order of signal strength.

Networks to Test

Network Name	BSSID	Security	Signal
Ali house	9C:7D:A3:8C:59:D8	WPA, WPA2	77%
Ali House 2	04:95:E6:F5:39:D8	WPA, WPA2	49%
Router#2	D8:32:14:7B:E5:C0	WPA, WPA2	37%
House # 479-D	6C:D7:19:1E:7E:78	WPA, WPA2	35%
Ch hadi	70:79:90:1F:C1:9C	WPA, WPA2	32%
ExtronAI	8C:42:6D:10:A2:58	WPA, WPA2	30%
StormFiber-58d8	6C:D7:19:24:58:D8	WPA, WPA2	19%
Classes	6C:A4:D1:52:A1:E8	WPA, WPA2	19%
Chaudhary	68:58:11:FF:BF:50	WPA, WPA2	12%
Tenda_5EDB58	04:95:E6:5E:DB:58	WPA, WPA2	12%
No Connection	A0:8C:F8:44:8F:0D	WPA, WPA2	9%

Attack Options

☑ Default Credentials Check
☑ WPS PIN Attack
☑ WPA Handshake Capture
☑ WEP Attack
☑ IP Address Detection
☑ Port Scanning

Timeout per attack (seconds): 60

⚠ Only use against networks you own or have permission to test!

Fig. 4. Post-exploitation internal network enumeration after successful wireless compromise.

B. Algorithmic Approach to Threat Detection

The traditional signature-based detection comparing to Aerostrike. It uses machine learning fusion algorithms which are used to classify traffic and find anomalies. Traffic classification using the RF to perform the classification of the network protocols and determine the benign and malicious patterns of network traffic, the framework uses the RF classifier and it is an ensemble algorithm of learning that builds numerous decision trees during the training time. For a given input vector x, the model aggregates the votes from individual trees to determine the final class prediction ŷ. This approach is selected for its robustness against overfitting and its ability to handle the high-dimensionality inherent in network packet data. The Gini Impurity, utilized to measure the quality of a split in the decision trees, is calculated as,

$$G = 1 - \sum_{\{i=1\}}^{\{C\}} (p_i)^2$$

Where p_i represents the probability of a traffic sample belonging to class **i** among **C** total classes.

Anomaly detection with an isolation forest to detect anomalies in the form of ”zero-day” threats or employees performing otherwise unusual activity that does not correspond to the established standards, the system applies the use of an isolation forest algorithm. isolation forest unlike the distance-based techniques identifies anomalous cases by isolating data points explicitly. It is based on the concept that anomalies are limited and can be isolated closely on top of the root of the tree.

The anomaly score s(x, n) for an observation x is defined as:

$$s(x,n) = 2^{-\frac{E(h(x))}{c(n)}}$$

Where h(x) is the path length required to isolate the sample, E(h(x)) is the average path length across the forest, and c(n) is the average path length of unsuccessful searches in a Binary Search Tree (BST) for n instances. When a score is close to 1 it means that there is high chance of the malicious anomaly, and an alert will be triggered at the monitoring interface.

C. Implementation Strategy

It is executed on Aerostrike framework, which is written in Python, 3.11, which is chosen as having a large ecosystem of networking and data science libraries. The development plan focuses attention to non-blocking I/O and efficient memory management to provide stability in the infrastructures with a lack of resources.

The basic network packet processing is constructed using Scapy 2.5.0, which does low-level packet manipulation, sniffing and injecting packets. In order to avoid the frozen state of the Graphical User Interface (GUI) application developed in Tkinter in case of heavy network execution, the system uses the power of Asyncio. This enables network scanning, capture of handshake and analysis of traffic to be performed in parallel as asynchronous tasks, breaking the connation between the interface rendering logic and the processing logic.

Log management and data processing are done using NumPy 1.24.0 and pandas 2.0.0 that can offer the required computational power to use real-time traffic view and re- porting. In addition, the system connects OpenAI GPT API to digest technical scan data and create natural-language remediation strategies to link raw data with practical security insights. The storage used to store data is handled through a hybrid approach: the history and logs of the session are stored through the help of SQLite databases whereas configuration templates are stored in the form of JSON files and external compliance auditing is achieved with the help of CSV exports.

IV. Experimental Evaluation and Results Analysis

A rigorous validation of the efficacy of the Aerostrike framework was performed with a complete experimental ex- amination under the controlled laboratory setting aimed at simulation of the real-world situation of the heterogeneity of wireless environments. The key goal was to evaluate the performance of the system in four significantly important dimensions i.e., the accuracy of vulnerability detection, the accuracy of risk scoring, the efficacy of automation and the efficient use of computational resources.

A. Experimental Testbed Configuration

The assessment was based on a variety of testbed with more than 20 wireless access points (APs) to simulated typical Small Office/Home Office (SOHO) anesthetic conditions. In order to make the testing of the framework versatility solid, a variety of security settings consisting of the legacy WEP and Open settings to the current WPA/WPA2

(PSK) setting and WPS setting was used as the target infrastructure. The machine learning modules were tested with simulated network traffic comprised of a diverse array of clients such as laptops, smartphones, and IoT devices to produce realistic streams of data in order to be analyzed by the machine learning modules.

B. Vulnerability Detection and Risk Assessment Performance

The framework had a high level of efficiency on the categorization of networks in accordance with their encryption standards and detection of anomalies in configuration. Through coordinating the mismatched protocol errors to the risk scoring engine of the system driven by AI, Aerostrike was able to prioritize high-risk targets automatically. In particular, the system had 100 percent success rate in detection of WEP encrypted networks and key recovery took few minutes owing to the weaknesses of the protocol. In the case of WPA/WPA2 infrastructures, the framework was accurate in 4-way handshakes, allowing the latter to be attacked offline with a dictionary, and at the same time revealing WPS-enabling devices that could be attacked by brute-force of PIN. These findings confirm existent capability in the system at automating the reconnaissance stage, filtering noise to provide the user with exploit paths to be exploited.

C. Efficiency and Automation Analysis

One of the contributions of Aerostrike includes the automation of operations by reducing operational latency. In order to measure this, the framework was compared with industry-standard applications in particular the Aircrack-ng and Wireshark application of seconds needed to attain a complete cycle of discovering, analyzing, and reporting.

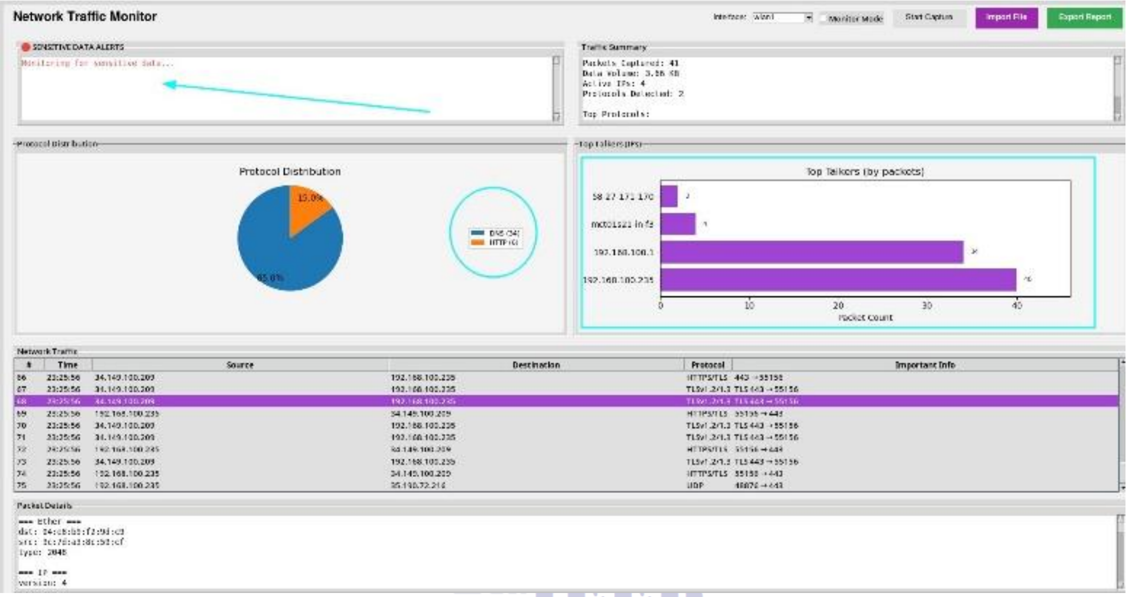
TABLE I
Technical Performance Comparison of Wireless Penetration Tools

Tool	Automation	Time (min)	Usability
Aircrack-ng	Manual	45	CLI / Expert
Wireshark	Semi-Automated	30	Intermediate
Aerostrike	Full Automation	10–15	GUI / User-Friendly

According to Table 1, Aerostrike is much more effective than legacy tools and shortens the period of its assessment to around 10 to 15 minutes. Although Aircrack-ng has an option of fine-grained control, it uses command-line execution by users, which has a latency factor that takes as long as 45 minutes to accomplish a comparable workflow. The ability of aerostrike to combine scanning, attack and reporting in one pipeline removes the overhead caused by context-switching that usually characterizes manual penetration testing.

D. Real-Time Traffic Analysis and Resource Utilization

After the successful network association, a framework changes to the real time traffic monitoring. Asynchronous execution is used in this module to monitor the packet streams without negative impact on the system responsiveness and operation is made smooth even in resource-strained systems.



```
Monitoring for sensitive data...
```

```
Packets Captured: 41
Data Volume: 3.56 KB
Active IPs: 4
Protocols Detected: 2

Top Protocols:
```

#	Time	Source	Destination	Protocol	Important Info
66	23:25:56	34.149.100.209	192.168.100.235	HTTPS/TLS	443 → 55156
67	23:25:56	34.149.100.209	192.168.100.235	TLSv1.2/1.3	TLS 443 → 55156
68	23:25:56	34.149.100.209	192.168.100.235	TLSv1.2/1.3	TLS 443 → 55156
69	23:25:56	192.168.100.235	34.149.100.209	HTTPS/TLS	55156 → 443
70	23:25:56	34.149.100.209	192.168.100.235	TLSv1.2/1.3	TLS 443 → 55156
71	23:25:56	34.149.100.209	192.168.100.235	TLSv1.2/1.3	TLS 443 → 55156
72	23:25:56	192.168.100.235	34.149.100.209	HTTPS/TLS	55156 → 443
73	23:25:56	34.149.100.209	192.168.100.235	TLSv1.2/1.3	TLS 443 → 55156
74	23:25:56	192.168.100.235	34.149.100.209	HTTPS/TLS	55156 → 443
75	23:25:56	192.168.100.235	35.190.72.214	UDP	48876 → 443

```
=== Ether ===
dst: 84:c8:b1:f2:9d:c3
src: 3c:7d:a3:8c:5d:cf
type: 2048

=== IP ===
version: 4
```

Fig. 5. Real-time Network Traffic Monitor displaying protocol distribution and sensitive data alerts.

The interface will give instant visual response to network behavior as shown in Figure 5. This system classifies traffic by protocol (i.e. between flows of HTTP and DNS) and detects Top Talkers to find active IP addresses. Most importantly, the module has its sensitive data alert system that alerts any unencrypted transmission of some important details in real-time. This visualization feature enables network spikes to be immediately compared to particular host process with analysts, which would otherwise need close examination in a packet analyzer.

E. Comparative Operational Assessment

In an attempt to further contextualize the benefits of the suggested framework, a bigger comparative analysis was carried out in terms of complexity of setup and the requirements of operation. The findings, which have been condensed in Table 2, indicate the trend of moving to the expert-independent process of manual operations to automated, semi-autonomous processes.

TABLE II
COMPARATIVE ANALYSIS WITH TRADITIONAL TOOLS

Criteria	Legacy Tools	Aerostrike	Gain
Setup Time	30–45 min	5–10 min	~80% Faster
Automation	Manual / Scripted	Full Automation	100%
Reporting	Hours (Manual)	Instant	~90% Saved
Skill Level	Expert	Beginner / Interm.	Accessible
Workflow	Tool Switching	All-in-One	Simplified

The information shows that with Aerostrike setup, time is saved by about 80%, as compared to the traditional tool chains. The democratization of advanced security auditing is achieved by the utilization of an AI-enabled Integration of an all-in-one workflow and aids the reduction of the need to have specialized knowledge when it comes to a particular command-line syntax, as well as making advanced security auditing more accessible. Moreover, the automated report generation functionalist over- comes a major engagements point of professional activities, which makes hours of tedious paperwork a real-time action.

V. LIMITATIONS AND FUTURE ENHANCEMENTS

Although Aerostrike shows impressive progress in automating the wireless penetration testing, it must be noted that the existing limitations must be considered to put it into perspective and define the way of its future evolution.

A. Current System Limitations

Although the framework has been shown to be effective under a controlled setting, there are a number of inherent constraints that influence the scope of its operation under highly complex or hardened settings:

Dependency on Dictionary Attacks: The existing version of the system is based on heavy possibilities to use external wordlists to conduct WPA/WPA2-PSK cracking attempts. The system will therefore be weak with powerful, randomly constructed passwords that are not within regular dictionaries, hence reducing the authenticity of the security test to strong credentials of high qualities.

Dataset Constraints in Machine Learning: The machine learning models that are a part of the system have been trained on a limited controlled dataset. This limits the training scope of the models indicating that they are prone to decreased accuracy when employed in such adverse real-world scenarios as high traffic or rough conditions or when operating with unseen traffic patterns that cannot be well aligned to the training data.

Hardware Compatibility:
The inherent hardware capability of the performance consists in the necessity of wireless adapters that can allow monitor mode and injecting packets. The incompatibility between the hardware and driver may cause mixed outcomes or even not be able to run a particular injection based attack.

B. Future Work and Strategic Enhancements

In order to overcome these limitations and increase the functional playability of the framework, it will have its future development based on the architectural scalability aspect and high degree of technological integration.

1) Technological and Algorithmic Improvements: The sub- sequent versions will focus on increasing the level of intelligence of the central processing unit. In order to enhance the accuracy of the traffic classification and detection of anomalies in the complex networks, deep learning architecture and the ensemble-based model were suggested to be integrated. In addition, the extent of testing will also be intensified to the extent of testing a wider range of attack vectors and security settings to cover a broad range of threats as they arise. One of the most essential planned improvements is integrating Aerostrike with a live threat feed and CVE databases so that the system can automatically detect and prioritize any new vulnerabilities and maladjustments reported in real-time.

2) Platform Extension and Deployment: Platform flexibility will also be considered in development in case it can be used by more people. It is also envisioned that a mobile version that is small can be used, and thus increases the portability of field- based tests in which it would be unfeasible to carry a laptop. Also, a browser-based web interface will be built centrally in case of enterprise implementation. This will enable distant functions, multi-user cooperations, and report centralization to make the device better adaptable to big security staff.

VI. Conclusion

This paper introduces Aerostrike, an artificial intelligence- assisted wireless penetration testing framework that aims to rationalize and automatize the complicated process of the Wi- Fi security testing. The given solution is efficient in the context of the main concerns of conventional wireless testing tools namely the manual aspect of the testing process, high time- consumption, and the fragmentation of the reporting functions. Aerostrike, conducted an experimental validation of their high capability in the field of automatizing network discovery, free-to-use machine learning, and performance on the risk quantification of a standard Wi-Fi exploitation to generate professional-level security reports. The added benefit of the framework is that it incorporates real-time traffic monitoring and AI-based remediation advice, which helps to eliminate the understanding between technical exploitation and applied security enhancements. In the future, additional studies will be made on enhancements in attack capabilities as well as refinement of AI-driven analysis engine so that the system can be made flexible and efficient in the ever-changing cybersecurity environment.

References

GSMA, *The Mobile Economy 2025*, GSMA Intelligence, London, U.K., 2025. [Online]. Available: https://www.gsma.com/mobileeconomy/

Cisco Systems, *Cisco Annual Internet Report (2024–2025)*, Cisco White Paper, 2025. [Online]. Available: https://www.cisco.com/c/en/us/solutions/executive-perspectives/annual-internet-report.html

ENISA, *Threat Landscape for 5G Networks*, European Union Agency for Cybersecurity, 2024. [Online]. Available: https://www.enisa.europa.eu/publications/threat-landscape-for-5g

S. Park, J. Kim, and H. Kim, "Security vulnerabilities and attack trends in modern Wi-Fi networks," Computer Networks, vol. 232, p. 109889, 2024.

S. Axelsson and U. Franke, "Automation in penetration testing: State of practice and research challenges," Computers & Security, vol. 124, p. 102978, 2023.

A. Behl and K. Behl, *Cyberwar: The Next Threat to National Security and What to Do About It*, Oxford University Press, Oxford, U.K., 2017.

MITRE Corporation, *MITRE ATT&CK® for Enterprise*, MITRE, 2024. [Online]. Available: https://attack.mitre.org/

M. Ring, S. Wunderlich, D. Scheuring, D. Landes, and A. Hotho, "A survey of machine learning techniques for cyber security," *IEEE Communications Surveys & Tutorials*, vol. 21, no. 1, pp. 1–40, 2019.

L. Breiman, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.

F. T. Liu, K. M. Ting, and Z.-H. Zhou, "Isolation forest," in *Proc. IEEE International Conference on Data Mining (ICDM)*, 2008, pp. 413–422.

S. Axelsson, "The limits of automated intrusion detection," *ACM Transactions on Information and System Security*, vol. 3, no. 3, pp. 186–205, 2000.

T. Zhukabayeva, Z. Ahmad, A. Adamova, N. Karabayev, Y. Mardenov, and D. Satybaldina, "Penetration testing and machine learning-driven cybersecurity framework for iot and smart city wireless networks," *IEEE Access*, 2025.

M. Naeem, Y. Salam, A. Azeem, A. Sattar, A. Sufyan, and F. Salam, "Customizable digital control system for domestic application with iot capability," in *2021 International Conference on Computing, Electronic and Electrical Engineering (ICE*

Cube), 2021, pp. 1–6.

S. A. Elsaid and A. Binbusayyis, "An optimized isolation forest based intrusion detection system for heterogeneous and streaming data in the industrial internet of things (iiot) networks," *Discover Applied Sciences*, vol. 6, no. 9, p. 483, 2024.

C. Hamroun, A. Fladenmuller, M. Pariente, and G. Pujolle, "Intrusion detection in 5g and wi-fi networks: A survey of current methods, challenges & perspectives," *IEEE Access*, 2025.

A. Tareef, Y. M. Allawi, A. A. Alkasasbeh, A. Abadleh, W. Alamro,

M. Alghamdi, A. I. Zreikat, and H. Kang, "A machine learning approach for detecting wpa3 downgrade attacks in next-generation wi-fi systems," *PLoS One*, vol. 20, no. 9, p. e0331443, 2025.

M. J. BAWANEH, O. M. AL-HAZAIMEH, M. M. AL-NAWASHI,

M. H. AL-BSOOL, and E. HANANDAH, "Enhanced iot cybersecurity through machine learning-based penetration testing," *Applied Computer Science*, vol. 21, no. 2, pp. 96–110, 2025.

Bishop Fox, "Pen testing tools 2024: The good, the bad, and the fragmented," Bishop Fox Blog, 2024. [Online]. Available: https://bishopfox.com/blog/pen-testing-tools-2024

Attaullah, A. Sufyan, M. Mujeeb-Ur-Rehman, B. Noreen, and S. Amin, "Trends, capabilities, and challenges in modern cyber defense: A systematic review of detection and response technologies," *Spectrum of Engineering Sciences*, vol. 4, no. 1, p. 464–503, Jan. 2026. [Online].
Available: https://thesesjournal.com/index.php/1/article/view/1905

Pentera, "The state of pentesting 2025," Pentera Research, Tech. Rep., 2025. [Online]. Available: https://pentera.io/resources/reports/global- state-of-pentesting-2025-survey-report

Horizon3.ai, "Annual insights report: The state of cybersecurity in 2025," Horizon3.ai, Tech. Rep., 2025. [Online]. Available: url = https://horizon3.ai/wp-ontent/uploads/2025/06/HS2503$_I$*nsightsReport.pdf,*

Cobalt, "State of pentesting report 2025," Cobalt Research, Tech. Rep., 2025. [Online]. Available: https://www.cobalt.io/state-of-pentesting

L. Wang, X. Shi, Z. Li, Y. Jiang, S. Tan, Y. Jiang, J. Cheng, W. Chen,

X. Shen, Z. LI *et al.*, "Automated penetration testing with llm agents and classical planning," *arXiv preprint arXiv:2512.11143*, 2025.

B. Casey, J. C. Santos, and G. Perry, "A survey of source code representations for machine learning-based cybersecurity tasks," *ACM Computing Surveys*, vol. 57, no. 8, pp. 1–41, 2025.

M. Sasi, O. R. Adegboye, and A. Alzubi, "Explainable and optimized random forest for anomaly detection in iot networks using the rime metaheuristic," *Electronics*, vol. 14, no. 22, p. 4465, 2025.

S. Ragul and S. Tamilselvi, "Classification of iot network traffic using random forest classifier," *Int. J. Adv. Trends Eng. Manag. III*, 2024.

E. Tuyishime, M. Martalo`, P. A. Cotfas, V. Popescu, D. T. Cotfas, and A. Rekeraho, "Resource-efficient traffic classification using feature selection for message queuing telemetry transport-internet of things network-based security attacks," *Applied Sciences*, vol. 15, no. 8, p. 4252, 2025.

Y. S. Razooqi and A. Pekar, "Binary vpn traffic detection using wavelet features and machine learning," *arXiv preprint arXiv:2502.13804*, 2025.

H. Xiang, X. Zhang, H. Hu, L. Qi, W. Dou, M. Dras,

A. Beheshti, and X. Xu, “Optiforest: Optimal isolation forest for anomaly detection,” in *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence, IJCAI-23*,

E. Elkind, Ed. International Joint Conferences on Artificial Intelligence Organization, 8 2023, pp. 2379–2387, main Track. [Online]. Available: https://doi.org/10.24963/ijcai.2023/264

A. A. S. AlQahtani, T. Alshayeb, M. Nabil, and A. Patooghy, “Lever- aging machine learning for wi-fi-based environmental continuous two- factor authentication,” *IEEE Access*, vol. 12, pp. 13 277–13 289, 2024.

A. Rachwał, P. Karczmarek, A. Rachwał, and R. Stegierski, “Isolation forest with exclusion of attributes based on shapley index,” *IEEE Access*, 2024.

S. Yonbawi, A. Afzal, M. Yasir, M. Rizwan, and N. Kryvinska, “Transferability evaluation in wi-fi intrusion detection systems through machine learning and deep learning approaches,” *IEEE Access*, vol. 13,

pp. 11 248–11 264, 2025.

M. H. Moharam, K. Ashraf, H. Alaa, M. Ahmed, and H. A. El-Hakim, “Real-time detection of wi-fi attacks using hybrid deep learning models on nodemcu,” *Scientific Reports*, vol. 15, no. 1, p. 32544, 2025.

M. B. Banko´, S. Dyszewski, M. Kra´lova´, M. B. Limpek, M. Pa- paioannou, G. Choudhary, and N. Dragoni, “Advancements in machine learning-based intrusion detection in iot: Research trends and chal- lenges,” *Algorithms*, vol. 18, no. 4, p. 209, 2025.

M. Asif, S. Shahid, and M. Rafiq-uz-Zaman, “Immersive technologies, awe, and the evolution of retail in the metaverse,” International Premier Journal of Languages & Literature, vol. 3, no. 4, pp. 713–748, 2025. [Online]. Available: https://ipjll.com/ipjll/index.php/journal/article/view/295

M. Khenwar and M. Nawal, “Challenges and limitations of ids: A comprehensive assessment and future perspectives.”

G. Zhao, L. Wang, and X. Wang, “A mallows-like criterion for anomaly detection with random forest implementation,” *PLoS One*, vol. 20, no. 6, p. e0323333, 2025.

M. R. Naeem, R. Amin, M. Farhan, F. S. Alsubaei, E. Alsolami, and M. D. Zakaria, “Cyber security enhancements with reinforcement learning: A zero-day vulnerabilityu identification perspective,” *PLoS One*, vol. 20, no. 5, p. e0324595, 2025.

D. Lo´pez-Montero, J. L. A´ lvarez-Aldana, A. Morales-Mart´ınez, M. Gil- Lo´pez, and J. M. A. Garc´ıa, “Reinforcement learning for automated cybersecurity penetration testing,” *arXiv preprint arXiv:2507.02969*, 2025.

Y. Li, “Deep reinforcement learning-based automated penetration testing for active distribution networks,” Ph.D. dissertation, Concordia Univer- sity, 2024.

M. A. Ferrag, F. Alwahedi, A. Battah, B. Cherif, A. Mechri, N. Tihanyi,

T. Bisztray, and M. Debbah, “Generative ai in cybersecurity: A com- prehensive review of llm applications and vulnerabilities,” *Internet of Things and Cyber-Physical Systems*, 2025.

G. Deng, Y. Liu, V. Mayoral-Vilches, P. Liu, Y. Li, Y. Xu, T. Zhang,

Y. Liu, M. Pinzger, and S. Rass, “Pentestgpt: An llm-empowered automatic penetration testing tool,” *arXiv preprint arXiv:2308.06782*, 2023.

Y. Ginige, A. Niroshan, S. Jain, and S. Seneviratne, “Autopentester: An llm agent-based framework for automated pentesting,” *arXiv preprint arXiv:2510.05605*, 2025.

H. Kong, D. Hu, J. Ge, L. Li, H. Li, and T. Li, “Pentest-r1: Towards autonomous penetration testing reasoning optimized via two-stage rein- forcement learning,” *arXiv preprint arXiv:2508.07382*, 2025.

M. Asif, A. Ali, and F. A. Shaheen, "Assessing the Effects of Artificial Intelligence in Revolutionizing Human Resource Management: A Systematic Review," Social Science Review Archives, vol. 3, no. 4, pp. 2887–2908, 2025. doi: 10.70670/sra.v3i3.1055.

M. Aslam and M. Asif, "Organizational Power Structures and the Reproduction of Gender Inequality," Apex Journal of Social Sciences, vol. 4, no. 1, pp. 57-67, 2025. [Online]. Available: https://apexjss.com/index.php/AJSS/article/view/19

S. Markovic, "Llms can assist with vulnerability scoring, but context still matters," Help Net Security, Dec 2025. [Online]. Available: https://www.helpnetsecurity.com/2025/12/26/llms-automated- vulnerability-assessment

Marketsand Markets, "Generative AI cybersecurity startup assessment report 2025," 2025. [Online]. Available:https://www.marketsandmarkets.com/blog/ICT/generative-ai-cybersecurity-startup

M. Asif and R. J. Asghar, "Managerial accounting as a driver of financial performance and sustainability in small and medium enterprises in Pakistan," Center for Management Science Research, vol. 3, no. 7, pp. 150–163, 2025. doi: 10.5281/zenodo.17596478.

Acuvity, "2025 state of AI security report," 2025. [Online]. Available: https://acuvity.ai/2025-state-of-ai-security/

A. Sufyan, K. B. Khan, O. A. Khashan, T. Mir, and U. Mir, "From 5g to beyond 5g: A comprehensive survey of wireless network evolution, challenges, and promising technologies," *Electronics*, vol. 12, no. 10, p. 2200, 2023.

Y. Usman, H. Oladipupo, A. D. During, A. Robert, and R. Chataut, "Ai, ml, and llm integration in 5g/6g networks: A comprehensive survey of architectures, challenges, and future directions," *IEEE Access*, 2025.