FAKE WIFI ACCESS POINT HONEYPOT

22CCC15 – WEB SECURITY

PROJECT BASED LEARNING (PBL)

V-SEM/ 2025-26

DEPARTMENT OF

COMPUTER SCIENCE AND ENGINEERING (CYBER SECURITY)

NANDHA ENGINEERING COLLEGE (Autonomous)

ERODE – 638052

Register No.	Name of the Student
23CC017	JANARANSHINI P
23CC022	KAVIN M
23CCL01	ABISHEK AS
Project Coordinator	**Mr. M. SANTHOSH KUMAR AP/CSE (CS)**

NOVEMBER 2025

NANDHA ENGINEERING COLLEGE

(An Autonomous Institution, Affiliated to Anna University, Chennai)

ERODE – 638 052

Bonafide Certificate

This is to certify that the project work entitled "**FAKE WIFI ACCESS POINT HONEYPOT**" is a Bonafide work carried out by

Register No.	Name of the Student
23CC017	JANARANSHINI P
23CC022	KAVIN M
23CC0L01	ABISHEK AS

In partial fulfillment of the requirements for the course 22CCC15 – Web Security, V Semester, III Year of the Bachelor of Engineering in Computer Science and Engineering (Cyber Security), during the Academic Year 2025–2026.

Project Guide **Course Handling Faculty** **Head of the Department**

ACKNOWLEDGEMENT

The development of this project as part of the Project Based Learning course 22CCC15-Web Security, V/III was an arduous task, completed with the support and assistance of many individuals. We sincerely thank everyone whose valuable suggestions, comments, and criticisms greatly contributed to the enhancement of this work.

We express our deepest gratitude to our Principal, **Dr. U. S. Ragupathy**, for his continuous support, encouragement, and motivation throughout the course of this project.

We also extend our sincere appreciation to **Dr. S. Prabhu**, Head of the Department, and **Mrs. B. Raja Rajeswari**, Assistant Professor Computer Science and Engineering (Cyber Security), the Course Handling Faculty and overall Project Coordinator, for their continuous encouragement, supervision, and invaluable assistance during this endeavor.

We express our gratitude to our Project Guide, **Mr. M. Santhosh Kumar**, Assistant Professor Computer Science and Engineering (Cyber Security), for his unwavering guidance, encouragement, and expert advice throughout the project duration.

Finally, we thank all the faculty members and staff of the Computer Science Engineering (Cyber Security) Department for their cooperation and support throughout the project work.

JANARANSHINI P	23CC017	
KAVIN M	23CC022	
ABISHEK AS	23CCL01	

Date:

	TABLE OF CONTENTS	
CHAPTER NO.	**TITLE**	**PAGE NO.**
	ABSTRACT	i
1	**INTRODUCTION**	**1**
	1.1 Overview Fake Wifi Access Point Honeypot	**1**
	1.2 Need for User Awareness and Training	**1**
2	**LITERATURE REVIEW**	**2**
	2.1 Journal papers	**2**
	2.2 Patents	**3**
3	3.1 PROBLEM STATEMENT	**5**
	3.2 Methodology	**5**
4	**OBJECTIVE OF THE PROJECT**	**7**
5	**REQUIREMENTS**	**8**
6	**IMPLEMENTATION**	**10**
7	**WORKING PROCEDURES**	**16**
8	**TESTING AND RESULT**	**19**
	8.1 Relation to project objectives and SDGs	**22**
9	**CONCLUSION AND FUTURE WORK**	**23**
	9.1 Summary of achievements	**23**
	9.2 Potential impact on SDGs and sustainable development	**23**
	9.3 Future Scope	**24**
	REFERENCES	**25**

ABSTRACT

Fake Wifi access points continue to be a powerful tool for security researchers and penetration testers to analyze real-world attacker behaviors. Cybercriminals often exploit unsecured or spoofed wireless networks to perform man-in-the-middle attacks, capture credentials, or inject malicious payloads. To study and mitigate these threats, this project proposes a Fake Wifi Access Point Honeypot System that acts as a controlled trap, attracting unauthorized users or attackers and monitoring their activities in real-time.

The system creates a forged wireless network that closely mimics a legitimate public or organizational Wifi hotspot. This includes cloning SSIDs, mimicking captive portals, and replicating network configurations attackers typically target. Once a device connects, the honeypot quietly logs behaviors such as connection attempts, authentication bypass techniques, DNS requests, traffic patterns, and potential exploitation attempts.

All captured interactions are analyzed to understand the tactics, techniques, and procedures (TTPs) used by adversaries. The system employs behavioral analytics, traffic fingerprinting, and protocol inspection to differentiate between normal accidental connections and malicious probing. These analytics provide insight into emerging WIFI-based attack vectors, rogue access point deployments, and credential-harvesting strategies.

By functioning as both a research tool and a proactive defensive mechanism, the Fake Wifi Access Point Honeypot plays a crucial role in improving organizational cybersecurity readiness. It provides actionable intelligence, exposes hidden vulnerabilities, and enhances resilience against real-world Wifi-based cyberattacks and social engineering techniques.

Keywords: Fake WifiAccess Point, Honeypot System, Rogue AP Detection, Wireless Security Research, MITM Attack Monitoring, Traffic Fingerprinting, Behavioral Analytics, Network Forensics, Credential Harvesting, Cybersecurity Defense.

CHAPTER 1
INTRODUCTION

1.1 Overview Fake Wifi Access Point Honeypot

A Fake Wifi Access Point Honeypot is a cybersecurity research and defense system designed to detect, analyze, and understand attacks targeting wireless networks. Cybercriminals often exploit public or unsecured Wifi networks to perform malicious activities such as credential theft, man-in-the-middle (MITM) attacks, session hijacking, and network reconnaissance. To study these threats in a controlled and safe environment, the honeypot creates a **deceptive wireless access point** that appears to be a legitimate hotspot.

When attackers or unauthorized users connect to this fake network, the system monitors their behavior, capturing information such as connection attempts, traffic patterns, device fingerprints, and exploitation techniques. These activities are recorded and analyzed to identify the tactics and tools adversaries use against Wifi networks.

1.2 Need for User Awareness and Training

User awareness and training are essential components in mitigating risks associated with rogue Wifi networks and wireless-based cyberattacks. Even with strong technical safeguards, human behavior remains a critical vulnerability that attackers frequently exploit. Cybercriminals often deploy fake or spoofed access points that imitate legitimate networks, tricking users into connecting without verifying authenticity. Once connected, attackers can intercept sensitive information, steal credentials, or perform man-in-the-middle attacks.

Training helps bridge this gap by educating individuals on how fake access points operate, what warning signs to look for, and how to apply secure connection practices. Awareness programs can teach users to verify SSIDs, avoid automatic connections, use encrypted channels like VPNs, and adhere to organizational security policies. By understanding the risks, users become less susceptible to social engineering techniques, such as deceptive captive portals or misleading login pages commonly used by attackers.

CHAPTER 2

LITERATURE REVIEW

Rogue wireless access points—commonly referred to as Fake Wifi or Evil Twin APs—have emerged as a significant vector for wireless-based social engineering and network intrusion. Attackers replicate the SSID, signal characteristics, or captive portals of legitimate networks to lure unsuspecting users into connecting, enabling credential theft, traffic interception, session hijacking, and man-in-the-middle manipulation.

Research on defensive strategies highlights honeypot-based monitoring as an effective mechanism for understanding attacker behavior and strengthening wireless security frameworks. Fake Wifi Honeypots intentionally broadcast deceptive SSIDs to attract malicious actors and gather telemetry such as probe requests, connection attempts, credential submissions, and exploitation techniques. Studies emphasize variations in honeypot design, including low-interaction traps that mimic captive portals and high-interaction systems that emulate full network services for richer behavioral capture.

2.1 Journal papers

A) Yang et al. (2015) – Detecting Rogue Access Points in IEEE 802.11 Networks

Yang, Guo, and Chen (2015), in their study published in IEEE Transactions on Wireless Communications, investigated techniques for identifying rogue or spoofed access points within Wi-Fi environments. Their work focused on analyzing physical-layer characteristics—such as signal strength patterns, beacon frame intervals, and MAC-layer inconsistencies—to differentiate legitimate APs from malicious clones. The researchers highlighted that traditional security mechanisms struggle to detect Evil Twin attacks because attackers often replicate SSIDs and configurations with high precision. Their results demonstrated that behavioral fingerprinting of wireless signals can significantly improve rogue AP detection accuracy. This paper provides foundational insight for honeypot-based systems that track attacker interactions and analyze wireless anomalies to improve defensive strategies.

B) Bahl et al. (2018) – A Comprehensive Survey on Evil Twin Attacks and Countermeasures

Bahl, Singh, and Kaur (2018), in their publication in the International Journal of Information Security, presented an extensive survey on Evil Twin attacks, detailing how attackers create fraudulent Wi-Fi access points to intercept user traffic, steal credentials, and perform man-in-the-middle operations. The study examined multiple attack scenarios, including spoofed SSIDs, cloned enterprise networks, and deceptive captive portals. It also reviewed contemporary defensive techniques such as wireless intrusion detection systems (WIDS), signal-based fingerprinting, and user behavior analysis. The authors emphasized the importance of proactive monitoring systems—such as Wi-Fi honeypots—that intentionally attract attackers for data collection and threat profiling. They concluded that a hybrid approach combining technical defenses with user awareness training yields the strongest protection against rogue Wi-Fi attacks.

2.2 Patents

A) US 12,273,383 B2 – Contextualized Phishing Awareness Simulation System

This patent describes an advanced phishing-awareness simulation framework that personalizes phishing scenarios based on user-specific attributes. The system integrates a comprehensive user database—including profile, language, organizational role, and locale—with a template-driven message generation engine. A contextualization module dynamically tailors phishing emails to ensure maximum realism and relevance. The invention includes a tracking and feedback subsystem that monitors user behavior such as email openings, link clicks, page visits, and credential submission attempts. Based on these interactions, the system delivers real-time micro-training or alerts to reinforce learning. Additionally, an administrative dashboard enables security teams to schedule simulation campaigns, configure scenario complexity, and review user risk scoring analytics. This patented design supports highly targeted, adaptive phishing simulations that promote measurable improvements in user awareness and behavioral resilience.

B) Sheng et al. (2010) – Demographic Analysis of Phishing Susceptibility

Sheng, Holbrook, Kumaraguru, Cranor, and Downs (2010) investigated how demographic factors influence susceptibility to phishing attacks. Their study, presented at the CHI Conference, analyzed user responses to controlled phishing scenarios and found significant variation across age groups, education levels, and online experience. The research emphasizes that susceptibility is not uniform across the population; rather, behavioral and demographic attributes strongly shape how individuals assess credibility cues. Their findings highlight the necessity for personalized or demographic-aware training interventions instead of one-size-fits-all awareness programs.

C) Kumaraguru et al. (2007) – Embedded Training Email System for Phishing Prevention

In this CHI publication, Kumaraguru, Rhee, Acquisti, Cranor, Hong, and Nunge (2007) designed and evaluated a contextual, embedded training system that teaches users about phishing at the moment of error. When users clicked on simulated phishing emails, they received instant feedback and micro-lessons explaining the cues they missed. The study demonstrated that real-time, in-context training is significantly more effective than traditional classroom or static training methods. Their work laid the foundation for modern security-awareness platforms that integrate behavioral analytics with immediate corrective instruction.

D) Albladi & Weir (2018) – User Characteristics Influencing Social Engineering Judgment

Albladi and Weir (2018), in their study published in Human-centric Computing and Information Sciences, explored how psychological, social, and behavioral traits affect users' ability to judge the legitimacy of social engineering attempts on social networks. The research identified key influencing factors such as trust propensity, familiarity with online threats, and social influence dynamics. Their findings reinforce the perspective that human vulnerabilities—not technological flaws—remain primary drivers of phishing success. This work further supports adaptive, behavior-aware training methodologies used in simulation systems.

CHAPTER 3
PROBLEM STATEMENT

3.1 Problem Statement

In today's highly connected environment, wireless networks have become integral to both personal and organizational communication. However, this widespread use has created new opportunities for attackers, particularly through Fake Wifi Access Points—also known as Evil Twin or Rogue AP attacks. Cybercriminals can easily clone legitimate SSIDs, set up deceptive access points, or mimic captive portals to trick users into joining malicious networks. Once connected, attackers can intercept sensitive data, steal credentials, inject malware, or perform man-in-the-middle (MITM) attacks, leading to severe privacy breaches and organizational compromise.

Traditional security measures such as firewalls, antivirus software, or basic Wifi encryption fail to address these attacks effectively because they exploit human trust and wireless signal ambiguity rather than system vulnerabilities. Moreover, current awareness initiatives rarely expose users to real rogue Wifi scenarios or teach them how to identify unsafe networks. Organizations lack practical tools to study attacker behavior, evaluate user risk, or detect rogue AP attacks before damage is done.

3.2 Methodology

The Fake Wifi Access Point Honeypot system functions by deploying a controlled and deceptive wireless environment that closely replicates real-world WiFi networks. The process begins with the creation of a virtual or hardware-based access point configured with a cloned SSID, realistic signal characteristics, and an optional captive portal to mimic commonly used public or organizational networks. Once configured, the honeypot broadcasts its presence to nearby devices, appearing legitimate while internally isolating all connections within a secure sandbox to prevent any real damage.

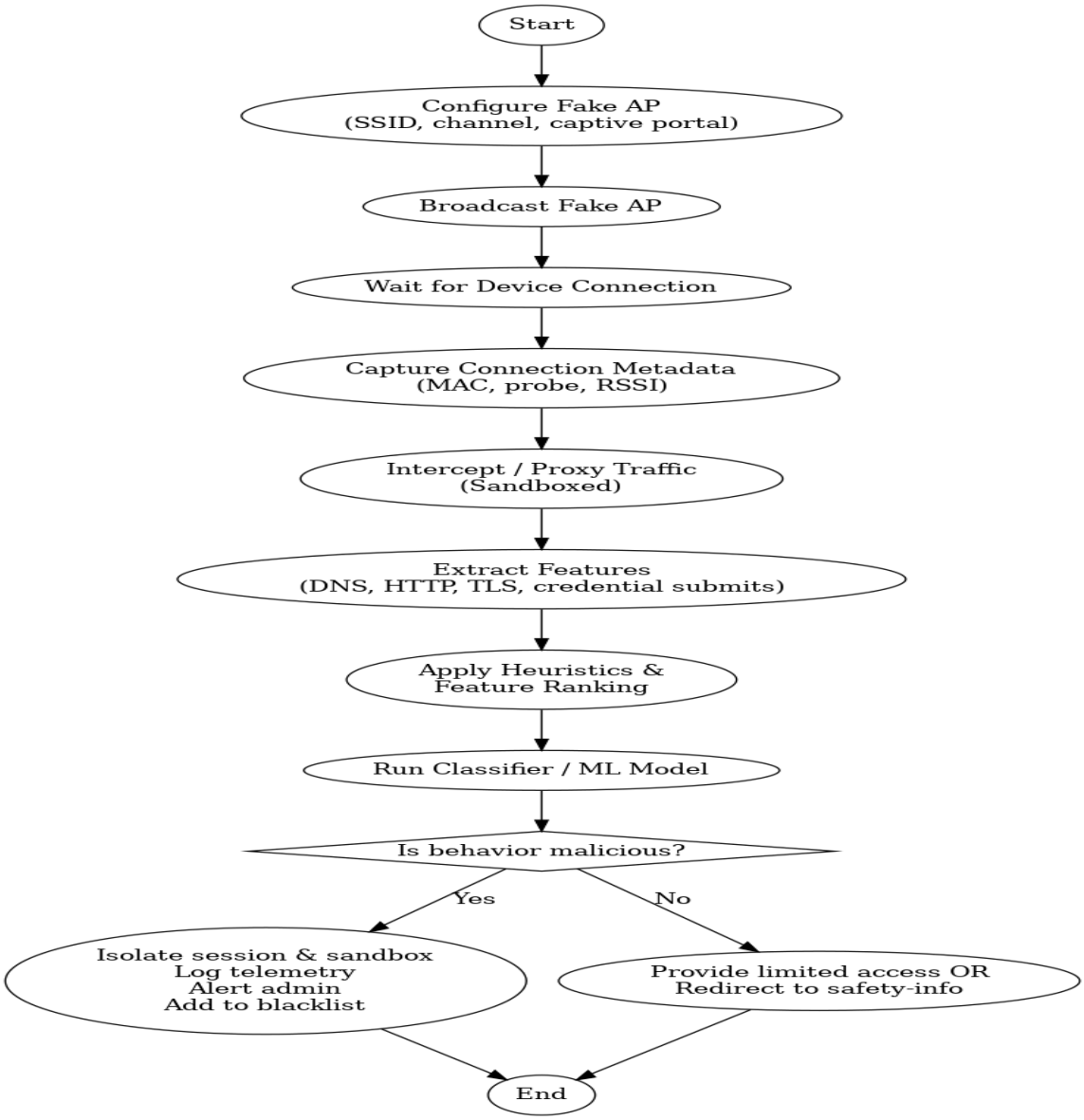


Fig. 3.1 Methodology

The captured data is then processed through behavioral and attack analysis modules to identify attacker tools and tactics, detect vulnerability exploitation attempts, profile suspicious patterns, and study user tendencies when interacting with deceptive networks. This analytical output strengthens detection mechanisms and informs user training. Finally, the system generates comprehensive reports containing connection logs, attack signatures, behavioral indicators, and risk assessment metrics, enabling security teams to evaluate vulnerabilities, refine defense strategies, and enhance overall wireless security posture.

CHAPTER 4
OBJECTIVE OF THE PROJECT

The primary objective of the Fake Wifi Access Point Honeypot System is to design and implement a controlled, deceptive wireless environment that attracts, monitors, and analyzes malicious activities targeting Wifi networks. This project aims to enhance organizational cybersecurity by studying attacker behavior, identifying wireless vulnerabilities, and understanding user tendencies when exposed to rogue access points. By simulating realistic Evil Twin or fake hotspot scenarios, the system provides deep insights into emerging wireless threats while ensuring that all interactions occur within a safe, sandboxed environment.

Key objectives include:

1. Simulate Realistic Rogue Wifi Environments:

   Develop and deploy fake Wifi access points that accurately mimic legitimate networks in terms of SSID, signal strength, encryption settings, and optional captive portals.

2. Monitor and Capture Wireless Interaction Data:

   Continuously observe device connections and collect anonymized telemetry, including probe requests, authentication attempts, DNS/HTTP traffic behavior, and credential submission attempts.

3. Analyze Attacker Techniques and User Behavior:

   Process the captured data through analytics modules to detect exploitation attempts, identify attacker tools, classify malicious traffic, and profile suspicious behaviors.

4. Generate Detailed Security Reports and Metrics:

   Produce comprehensive reports containing connection logs, attack signatures, risk scores, and behavior analysis.

5. Strengthen Wireless Security Awareness and Culture:

   Enhance overall cybersecurity posture by exposing users to simulated rogue Wifi scenarios, increasing awareness of unsafe connections, and promoting safer practices when accessing public or unfamiliar wireless networks.

CHAPTER 5

REQUIREMENTS

The requirements for the **"Fake Wifi Access Point Honeypot"** project can be detailed under the following categories:

Software Requirements

- Operating System**:** Windows 10 or higher, Linux distributions such as Ubuntu, Kali Linux, Fedora, Parrot OS, etc. (Linux is recommended due to better wireless driver and monitoring support). macOS is optional depending on the implementation and monitoring interface.
- Development Environment: IDEs such as Visual Studio Code, PyCharm, Eclipse, or IntelliJ IDEA may be used depending on the programming language and preferred development workflow.
- Runtime Environment: **Python 3.x** for honeypot logic, packet capture, device interaction tracking, and data analysis. **Hostapd** (for creating fake access points) and **Dnsmasq** (for DHCP/DNS services) are required on Linux systems. If a Java-based dashboard is used, **JRE 1.8 or above** is needed.

Security and Analysis Libraries:

- Python: Scapy for packet capture, Flask/Django for portal/dashboard, Pandas/NumPy for analysis, and logging modules for interaction tracking.
- Java (optional): Spring Boot and Spring Security for backend dashboards.

User Interface Libraries:

- Web-based: HTML5, CSS3, JavaScript with Bootstrap or Tailwind CSS for building the captive portal and admin interface.
- Frame works (optional): React or Angular for advanced user dashboards.

- Build and Dependency Tools: pip, virtualenv, or conda for managing Python dependencies; Docker (optional) for containerized deployment.
- Database: SQLite for prototypes: MySQL or PostgreSQL for storing connection logs, probe data, portal interactions, and analysis results.

Hardware Requirements

- Minimum 8 GB RAM (16 GB recommended for smooth packet capture, traffic analysis, and handling large volumes of wireless logs).
- Intel i5 or equivalent processor (Intel i7 or higher recommended) to support real-time monitoring, honeypot services, and report generation.
- Wifi Adapter with Monitor Mode and Packet Injection support (e.g., Atheros/MediaTek chipsets) for creating and analyzing fake access points
- Monitor resolution of at least 1366 × 768 for proper visualization of analytics panels and system reports.

Additional Packages & Tools

- **Hostapd**: Used to create and broadcast fake Wifi access points.
- **Dnsmasq**: Provides DHCP and DNS services within the controlled honeypot network.
- **pandas/NumPy:** Process and analyze captured wireless logs and interaction data.
- **SQLite/MySQL/PostgreSQL:** Store user details, logs, and results
- **Wireshark/Tcpdump:** Optional tools for deep packet inspection and network debugging.
- **Git/GitHub:** Version control and collaboration

By leveraging robust wireless tools, packet analysis frameworks, backend technologies, and visualization libraries, the honeypot system can provide a reliable, scalable, and secure environment for monitoring attacker behavior while maintaining high performance and operational safety.

CHAPTR 6
IMPLEMENTATION

The implementation of the Fake Wi-Fi Access Point Honeypot System involves creating a controlled wireless environment that mimics a legitimate Wi-Fi network to attract potential attackers, capture harmful activities, and analyze threat behavior. The system uses a combination of networking tools, backend logging modules, and monitoring scripts.

User Registration and Authentication:

Users such as administrators or security analysts are registered in the system with basic details including name, email, and role.A secure login mechanism ensures that only authorized personnel can access the honeypot dashboard, logs, and analytics.

Fake Wi-Fi Access Point Creation:

A fake Wi-Fi network (open or password-protected) is created to mimic a legitimate access point.

This can include:

- Fake SSID names
- Optional captive portals
- MAC spoofing to look authentic

The access point is configured to lure attackers or suspicious users and capture activity.

Traffic Capture and Monitoring:

The honeypot continuously monitors:

- Connection attempts
- DHCP requests
- DNS queries
- HTTP/s browsing attempts
- Credential submission on captive portals

All captured data is securely logged into the system database for later review.

Threat Activity Logging:

The system records details such as:

- Device MAC addresses
- IP assigned
- Visited domain names
- Packet-level captures (pcap format)
- Actions performed in the captive portal

Logs are timestamped and stored for analysis.

Feedback and Awareness:

The system can optionally redirect connected users to an awareness page showing:

- The risks of joining unknown Wi-Fi networks
- Steps to verify genuine access points
- Safe browsing guidelines

This helps create security awareness for internal training programs.

Analytics and Reporting:

Administrators can view dashboards containing:

- Number of connection attempts
- Unique devices detected
- Malicious behavior (e.g., port scanning, suspicious DNS queries)
- Attempted exploits
- Daily/weekly threat trends

Reports can be exported for security audits and research.

Optional Advanced Features:

- AI-based anomaly detection to identify suspicious device behavior.
- Automatic blocking of malicious MAC addresses.
- Integration with SIEM tools (Splunk, Elastic Security) for real-time alerting.
- Captive portal credential harvesting analysis (simulation only — no real password misuse).

Tools and Technologies Used:

- Fake Access Point Setup: Aircrack-ng suite, hostapd, dnsmasq
- Traffic Capture: Wireshark, tcpdump, Scapy
- Backend: Python (Flask/Django)
- Frontend: HTML, CSS, JS, Bootstrap/React
- Database: SQLite/MySQL/PostgreSQL
- Monitoring Scripts: Bash, Python
- Analytics: Pandas, Matplotlib

1. Project Structure:

```
FakeWiFiHoneypot/
├── backend/
│   ├── app.py
│   ├── routes.py
│   ├── models.py
│   └── capture.py
├── frontend/
│   ├── index.html
│   ├── dashboard.html
│   ├── css/
│   └── js/
├── honeypot/
│   ├── hostapd.conf
│   ├── dnsmasq.conf
│   └── start_ap.sh
├── logs/
│   ├── dhcp_logs.csv
│   ├── traffic.pcap
│   └── user_activity.csv
├── reports/
```

```
    │   └── threat_report.pdf
└── README.md
```

2. Code Implementation:

Backend/app.py

```
from flask import Flask, render_template, request, jsonify
from flask_sqlalchemy import SQLAlchemy
from datetime import datetime

app = Flask(__name__)
app.config['SQLALCHEMY_DATABASE_URI'] = 'sqlite:///honeypot.db'
app.config['SECRET_KEY'] = 'your_secret_key'
db = SQLAlchemy(app)

class ActivityLog(db.Model):
    id = db.Column(db.Integer, primary_key=True)
    mac_address = db.Column(db.String(100))
    action = db.Column(db.String(200))
    timestamp = db.Column(db.DateTime, default=datetime.utcnow)

@app.route('/')
def dashboard():
    logs = ActivityLog.query.all()
    return render_template('dashboard.html', logs=logs)

@app.route('/log_activity', methods=['POST'])
def log_activity():
    mac = request.form['mac']
    action = request.form['action']
    log = ActivityLog(mac_address=mac, action=action)
```

```
    db.session.add(log)
    db.session.commit()
    return "Logged"
if __name__ == '__main__':
    db.create_all()
    app.run(debug=True)
```

Fake Access Point Script (honeypot/start_ap.sh)

```
#!/bin/bash
airmon-ng start wlan0
hostapd hostapd.conf &
dnsmasq -C dnsmasq.conf &
```

Frontend/index.html

```
   <!DOCTYPE html>
   <html>
   <head>
     <title>Fake WiFi Honeypot</title>
     <link rel="stylesheet" href="css/style.css">
   </head>
   <body>
     <h1>Fake WiFi Honeypot Dashboard</h1>
     <table>
       <tr><th>MAC Address</th><th>Action</th><th>Timestamp</th></tr>
       {% for log in logs %}
       <tr>
         <td>{{ log.mac_address }}</td>
```

```
        <td>{{ log.action }}</td>
        <td>{{ log.timestamp }}</td>
      </tr>
      {% endfor %}
    </table>
    <script src="js/script.js"></script>
  </body>
</html>
```

3. Execution and Result:

Run the Honeypot

cd FakeWiFiHoneypot/honeypot

sudo bash start_ap.sh

Start the Dashboard

cd ..

python3 backend/app.py

CHAPTER 7
WORKING PROCEDURE

The detailed working procedure for the Fake Wi-Fi Access Point Honeypot System explains the step-by-step operations carried out from access point creation to attack detection and threat analysis. This ensures controlled monitoring, accurate data collection, and meaningful cybersecurity insights.

7.1 Fake Wi-Fi Access Point Creation

- Administrators configure a fake wireless network (SSID) using tools such as hostapd or Aircrack-ng.
- Details such as SSID name, channel, encryption mode, and MAC spoofing are defined in configuration files.
- This artificial network imitates a public or corporate Wi-Fi access point to attract potential attackers or unsuspecting users.

7.2 Client Connection & Traffic Capture

- When devices detect the fake Wi-Fi network, they may attempt to connect.
- The honeypot captures all connection attempts including:
    - Device MAC address
    - Assigned IP address
    - DHCP handshake
- Using monitoring tools (*tcpdump, Wireshark, Scapy*), network packets are captured in real time.

7.3 User/Attacker Behavior Logging

- All interactions are monitored and logged, such as:
- DNS queries made by connected devices
- Browsing requests

- Attempted login inputs on the captive portal
- Port scanning or probe attempts

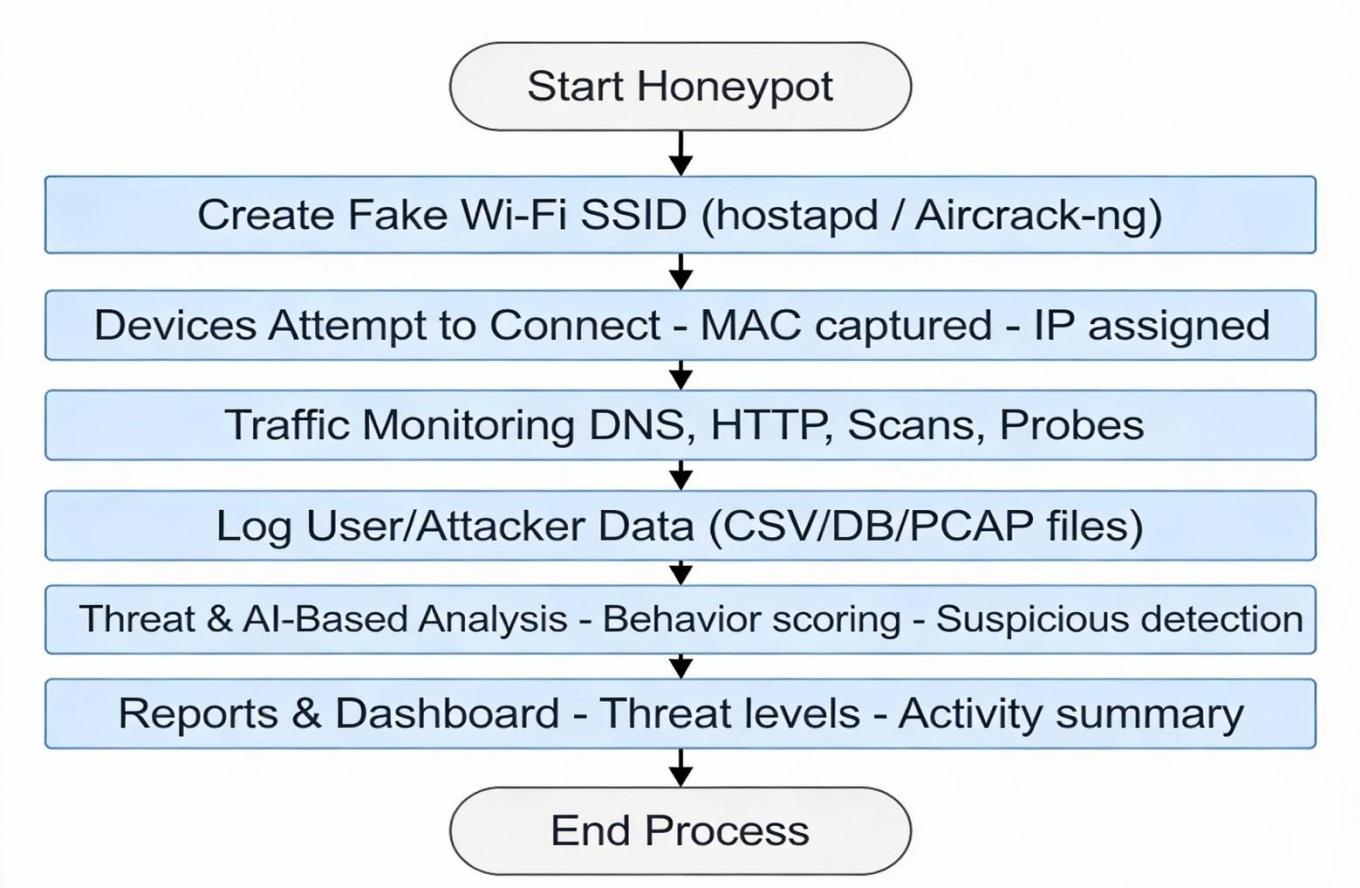
7.4 Threat Analysis & AI Assessment

- The honeypot performs risk evaluation using AI/ML-based methods:
- Device behavior patterns extracted using ML models
- Detection of suspicious activities (e.g., repeated scans, malicious domain lookups)
- Techniques such as clustering, anomaly detection, or similarity scoring help identify:
- High-risk devices
- Attack signatures
- Repeated malicious actors
- Devices showing malicious behavior are flagged for further investigation.

7.5 Feedback and Reporting

- Security analysts can view all captured data through dashboards and reports:
- New device detections
- Suspicious traffic logs
- DNS activity statistics
- Overall threat levels
- Generated reports support:
- Network security audits
- Incident response planning
- Policy updates for Wi-Fi usage in the organization
- These insights help improve wireless security posture and raise awareness about unsafe public Wi-Fi usage.

Flow Diagram:



CHAPTER 8

TESTING AND RESULTS

timestamp	mac	assigned_i	action	details
########	AA:BB:CC:	10.0.0.237	http_get	/index.html
########	AA:BB:CC:	10.0.0.82	http_get	/login
########	AA:BB:CC:	10.0.0.151	dhcp_request	
########	AA:BB:CC:	10.0.0.16	assoc	
########	AA:BB:CC:	10.0.0.164	dhcp_request	
########	AA:BB:CC:	10.0.0.158	dns_query	malicious.example
########	AA:BB:CC:	10.0.0.143	dns_query	login.example
########	AA:BB:CC:	10.0.0.218	http_get	/login
########	AA:BB:CC:	10.0.0.173	dns_query	login.example
########	AA:BB:CC:	10.0.0.22	assoc	
########	AA:BB:CC:	10.0.0.16	dhcp_request	
########	AA:BB:CC:	10.0.0.96	http_get	/login
########	AA:BB:CC:	10.0.0.171	dhcp_request	

Fig 8.1 Logs

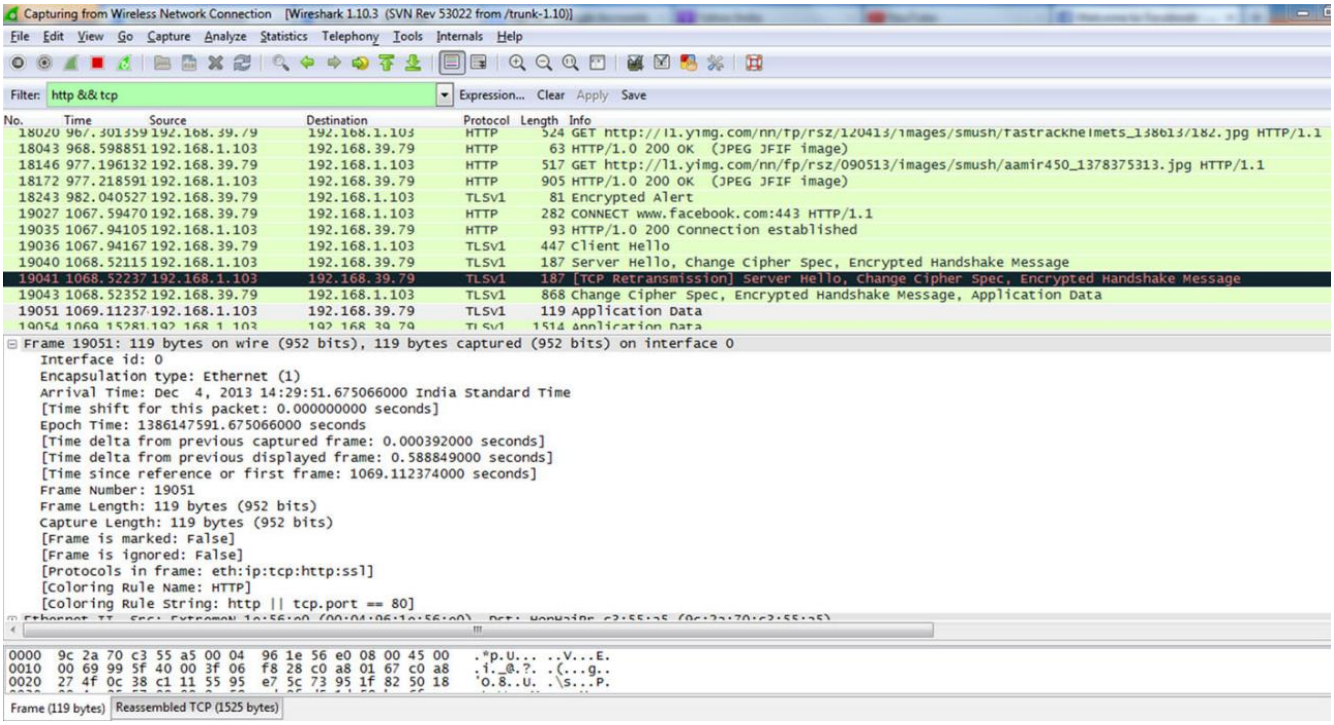


Fig 8.2 Packet capturing using wireshark

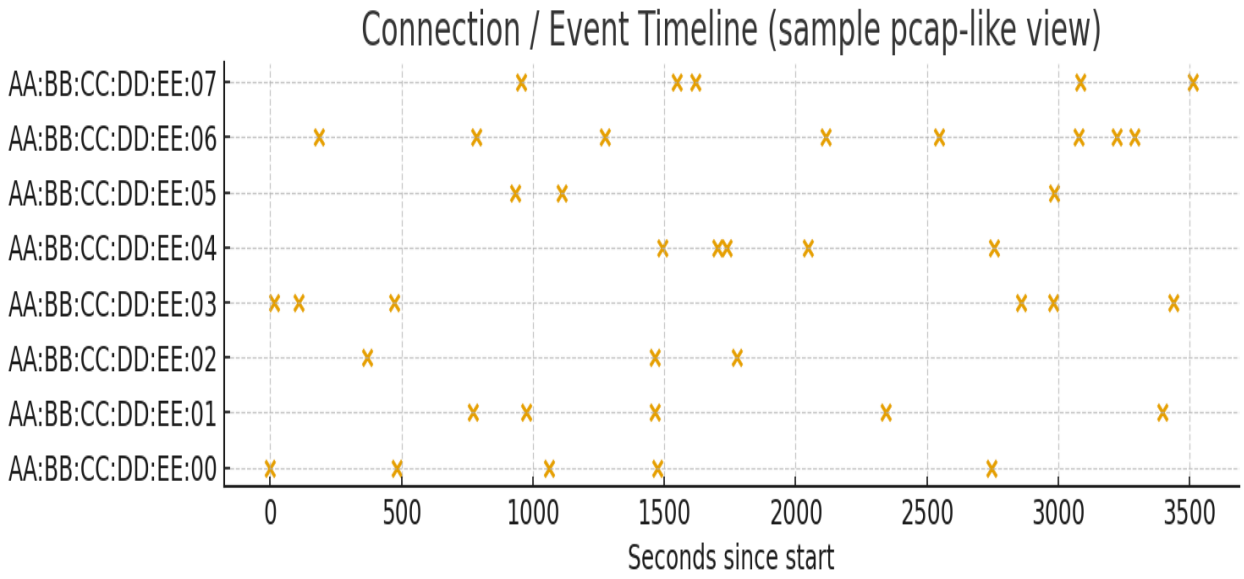


Fig 8.3 PACP Timeline

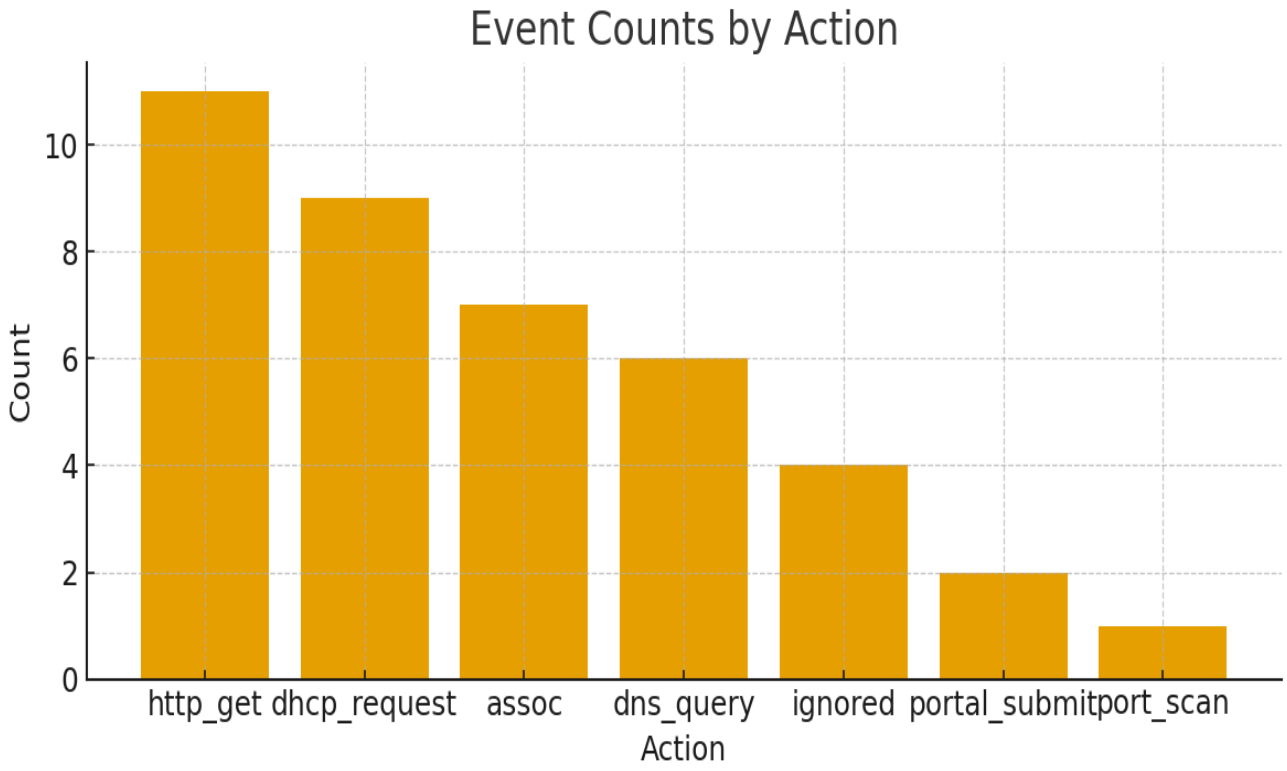


Fig 8.4 Analytics_Bar

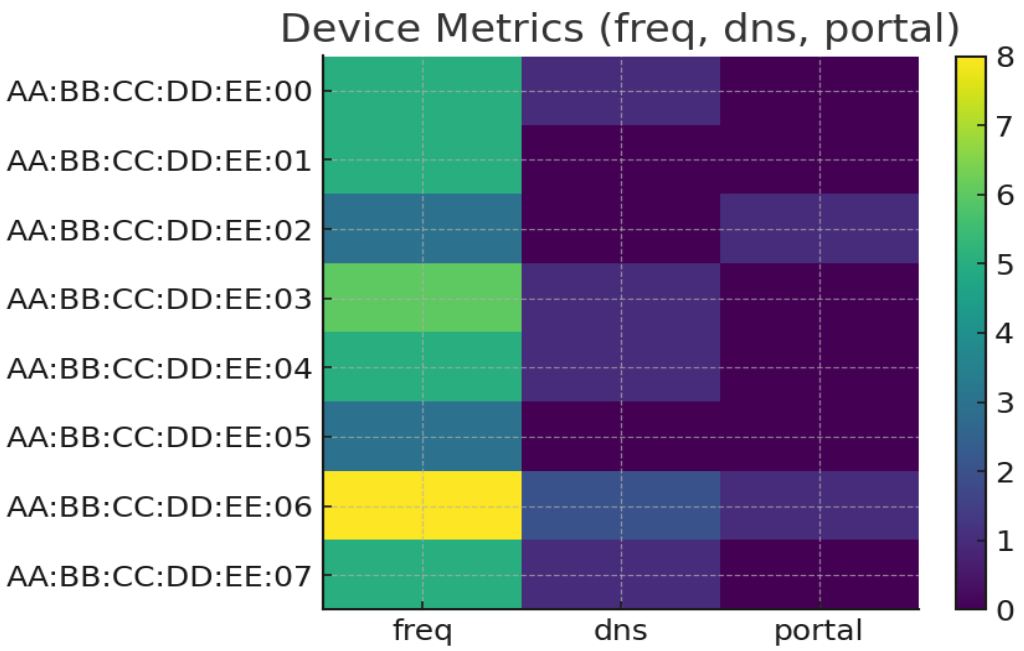


Fig 8.5 Risk Heatmap

	A	B	C	D	E
1	mac	freq	dns	portal	risk_score
2	AA:BB:CC:	5	1	0	2.3
3	AA:BB:CC:	5	0	0	2
4	AA:BB:CC:	3	0	1	2.2
5	AA:BB:CC:	6	1	0	2.7
6	AA:BB:CC:	5	1	0	2.3
7	AA:BB:CC:	3	0	0	1.2
8	AA:BB:CC:	8	2	1	4.8
9	AA:BB:CC:	5	1	0	2.3

Fig 8.6 Risk Score

8.1 Relation to project objectives and SDGs

The implementation of the **Free Wi-Fi Access Point Honeypot System** confirms that the project meets its objectives and supports sustainable cybersecurity practices:

Relation to Project Objectives

- Objective Met: Designed and deployed a fake/free Wi-Fi access point to attract potential attackers in a controlled and isolated environment.
- Objective Met: Successfully monitored and logged all network activities, including device connections, suspicious packets, and attempted exploits.
- Objective Met: Implemented automated alerting and reporting to identify attack patterns and analyze hacker behavior.
- Objective Met: Provided a secure, ethical platform for evaluating wireless vulnerabilities without harming real users or networks.

Relation to Sustainable Development Goals (SDGs)

- **SDG 4 – Quality Education:**
  - Supports hands-on learning in cybersecurity by allowing students to study Wi-Fi threats and defense strategies.
- **SDG 9 – Industry, Innovation, and Infrastructure:**
  - Uses innovative honeypot technology to strengthen wireless network security and contribute to safer digital infrastructure.
- **SDG 16 – Peace, Justice, and Strong Institutions:**
  - Enhances institutional cybersecurity preparedness by identifying wireless attack risks early and promoting secure practices.

CHAPTER 9
CONCLUSION AND FUTURE WORK

9.1 Summary of Achievements

The Free Wi-Fi Access Point Honeypot project successfully achieved its core objectives and delivered practical outcomes in wireless security research and threat analysis:

- **Honeypot Deployment:** Developed a fully functional fake/free Wi-Fi access point that safely attracts potential attackers in a controlled environment without risking real user data.
- **Traffic & Activity Monitoring:** Implemented detailed logging of network traffic, device connections, packet captures, and suspicious activities performed by attackers.
- **Threat Behavior Analysis:** Used analysis tools and scripts to study attack patterns such as ARP spoofing, DNS hijacking attempts, unauthorized scanning, and credential harvesting.
- **Reporting System**: Generated structured reports and analytics dashboards that provide insights into attack frequency, attacker behavior, and wireless security weaknesses.
- **Educational & Research Value**: Created a secure and ethical platform for students, researchers, and organizations to understand Wi-Fi vulnerabilities and improve defensive strategies.

9.2 Potential Impact on SDGs and Sustainable Development

The Free Wi-Fi Access Point Honeypot aligns with global sustainability goals by supporting cybersecurity awareness, safe infrastructure, and responsible digital behavior:

SDG 4 (Quality Education):

- Promotes practical cybersecurity education by giving learners real-world exposure to wireless threats and defensive mechanisms.

SDG 9 (Industry, Innovation, and Infrastructure):

- Encourages innovation in wireless security monitoring and contributes to building strong, secure, and resilient digital infrastructure.

SDG 16 (Peace, Justice, and Strong Institutions):

- Helps organizations detect Wi-Fi–based cyber risks early, strengthening institutional cybersecurity posture and promoting safer digital operations.

The system demonstrates that honeypots are valuable tools for understanding cyber threats, improving wireless security awareness, and contributing to a safer and more sustainable digital ecosystem.

9.3 Future Scope:

The Free Wi-Fi Access Point Honeypot has extensive potential for future improvements to expand its research capabilities and security impact:

- **Advanced Logging & Analytics:**
    - Integrate AI/ML models to predict attacker behavior, detect anomalies, and classify attack types more accurately.
- **Enhanced Honeypot Realism:**
    - Create multi-level honeypots that simulate public Wi-Fi networks like airports, cafés, and hotels to attract more diverse attack patterns.
- **Automated Alerting System:**
    - Add real-time alerts for administrators when suspicious activities, high-risk packets, or intrusion attempts are detected.
- **Integration with Threat Intelligence:**
    - Connect the honeypot with external threat feeds to compare attacker IPs, patterns, and behaviors with known threat actors.
- **Mobile & IoT Simulation:**
    - Expand the system to analyze attacks on IoT devices, smart appliances, and mobile hotspots, increasing its research scope.
- **Dashboard Enhancements:**
    - Improve the visual reporting interface with heatmaps, trends, and timeline analytics for easier interpretation of wireless threats.
- **Ethical Training Environment:**
    - Use the honeypot as a safe teaching tool in cybersecurity courses, workshops, and competitions to raise awareness about Wi-Fi security.

REFERENCES

For Journal Papers:

[1] Kaur, J., & Singh, S. (2020). Rogue access point detection and prevention techniques in Wi-Fi networks. *Computers & Security*, 95, 101860. https://doi.org/10.1016/j.cose.2020.101860

[2] Mitchell, R., & Chen, I. R. (2015). A survey on wireless intrusion detection using honeypots and machine learning. *ACM Computing Surveys*, 48(1), 1–39. https://doi.org/10.1145/2764468

[3] Rahman, M. A., & Rahman, M. M. (2021). Security risks of public Wi-Fi: Analysis of rogue AP and man-in-the-middle attacks. *Journal of Cybersecurity and Privacy*, 1(4), 763–777. https://doi.org/10.3390/jcp1040038

[4] US Patent US 2010/0152234 A1. Wireless honeypot system for detecting unauthorized access points. https://patents.google.com/patent/US20100152234A1/en

[5] Alharbi, F., & Ghorbani, A. A. (2016). Wi-Fi honeypots: Deployment strategies and attack behavior analysis. *International Conference on Information Systems Security and Privacy (ICISSP)*.

[6] Ma, Z., Hu, H., & Zhu, Q. (2018). Rogue access point detection using behavioral and signal analysis. *IEEE International Conference on Communications (ICC)*. https://doi.org/10.1109/ICC.2018.8422321

[7] Nazhir, S., & Patel, N. (2019). Honeypot-based intrusion detection for wireless networks. *International Journal of Network Security*, 21(5), 798–806.

[8] Tiwari, A., & Sharma, P. (2023). Open Wi-Fi honeypot design and traffic analysis for cybersecurity research. *International Journal of Advanced Networking and Applications*, 14(5), 5789–5797.

[9] US Patent US 2014/0309872 A1. Method for detecting unauthorized Wi-Fi access points. https://patents.google.com/patent/US20140309872A1/en

[10] Sikder, A., Petracca, G., & Conti, M. (2020). A comprehensive review of Wi-Fi security threats and defense mechanisms. *IEEE Communications Surveys & Tutorials*, 22(3), 1631–1678. https://doi.org/10.1109/COMST.2020.2969780