Wi-Fi Hacking with a raspberry pi

Faculty of Natural Science

Department of Computer Science

Zukisa Dyantyi

3567302@myuwc.ac.za

Project supervisor

Dr M Norman

mnorman@myuwc.ac.za

Project Co-Supervisor

Mr. M Mutemwa

mmutemwa@csir.co.za

ABSTRACT

The objective of this project is to create Cyber Security awareness and show people how easily their devices can be attacked using small a device like a Raspberry Pi. The prototype that will be built for demonstration on the Raspberry Pi will automatically scan the 2.4 and 5.0 Gigahertz (GHz) Radio Frequencies (RF) used by Wi-Fi devices for communications in order to determine the hardware and software information of these devices. In some cases, where possible the prototype will attempt to connect to Wi-Fi networks that have weak encryption algorithms or authentication mechanisms in order to further elicit more hardware and software information. Once an attacker has obtained the hardware and software information of a device, the attack can then craft specialized attacks towards that device in order to achieve the attacker's desired goal. This report will provide more information about the hardware and software necessary to carry out the project.

1 INTRODUCTION

1.1 Cyber Security

Cyber security is the practice of protecting systems, networks, and programs from digital attacks. These digital attacks are aimed at accessing, changing, or destroying sensitive information; extorting money from user [1]. The objective of this project is to build a hacking tool on a raspberry pi that automatically scans the 2.4 and 5.0 Gigahertz (GHz) Radio Frequencies (RF) used by Wi-Fi devices for communications in order to determine the hardware and software information of these devices. In some cases, where possible the prototype will attempt to connect to Wi-Fi networks that have weak encryption algorithms or authentication mechanisms. The prototype built can attempt to connect to the network either using a brute force or other techniques of choice such as rainbow table.

1.2 Problem identification and justification

The number of IoT devices is increasing, that means the connection between these devices has increased. By design IoT devices are not built with security in mind. The inherent capability to protect the information the IoT device processes or stores. Having many devices connected, there are high chances of digital attacks or cyber-attacks that can be launched against them. These devices connect, interact and exchange data. When there is a breach between these devices there will be leaked of information and other personal data leaking. The project objective is to educate peers around campus about the importance of strong password encryption.

2 LITERATURE REVIEW

After reading the paper on security of the Internet of Things (IoT) we came to the conclusion that outside attackers can gain access to the network through many different ways especially if the network is vulnerable for some reasons for example the network vulnerability is

weak encryption [2]. The paper focused more on Wi-Fi networks secured with WPA2 which was considered to be a secured network until October 2017 when Key Reinstallation Attack (KRACK) was announced. The paper state that WPA2 passwords are still vulnerable to attacks if weak passwords are used and it also shows that any device with outdated software are vulnerable to attacks [2]. IoT device users should avoid connecting to suspicious Wi-Fi networks and leave their devices unattended. To summarize the paper, it showed how multiple devices on the same network can be attacked because of one device's venerability and a variety of activities can be accomplished using a Raspberry Pi for example controlling lights, turning any TV to smart TV etc., especially if the Raspberry Pi connected to the Wi-Fi network.

The book [3], the focus of this book is to turn a raspberry into hacking arsenal and it also focused to those who have low budget, small form hacking tool that is remotely accessible. Implementation of this was done by running kali Linux on a Raspberry Pi. They placed the middle attack; middle attack is when attacker is adversary placing herself in the middle of the communication. Methods to exploit targets using attack tools are provided. Testing was done by running kali Linux OS on a Raspberry Pi and they used low power process that can run about one or two days on external battery. The testing was done from remote location and since they created a portable device security testing was done in different location.

3 USER REQUIREMENTS
3.1 Introduction

As mentioned in the introduction of the project proposal the number of devices connected to the network has increased and so the chances of cyber-attacks are also high. Users of these devices use Wi-Fi networks to accomplish their tasks or to share information with their colleagues or peers. Some of the Wi-Fi networks these users use have weak encryption which result the risk of being attack. This project will educate IoT device users the importance of strong password encryption either on their application such as E-mails or social media account, the project will focus more on Wi-Fi networks.

3.2 User's view of the project

This project is proposed by the Council for Scientific and Industrial Research (CSIR) and technical guidance will be provided by the CSIR team. Their view of the project: "The aim of the project is to build a tool on a raspberry pi that can automatically scan and attempt to connect to Wi-Fi networks with weak encryption.
Step 1: the candidate should build a tool on Raspberry Pi capable of connecting to Wi-Fi networks with basic Wi-Fi encryption key enabled on them.

Step 2: Using a phone, the candidate should setup dummy Wi-Fi hotpots using different encryption options and test that the hacking tool works as designed in step 1.

Step 3: Walk around campus with raspberry pi. The raspberry pi should attempt to connect to the network

either using brute force or rainbow table or other techniques of choice.

Step 4: Document the findings and notify the owners of the Wi-Fi networks that were found to have weak encryption key."

3.3 Description of the project

Stakeholders (CSIR) require Wi-Fi hacking tool to be built on a Raspberry Pi. The objective of this project is to detect how many networks are available either around campus or within the building. Which of these networks have weak and strong encryptions algorithms, which type of network protocol (WPA2, WEP or EAP) and what RF signal Wi-Fi network is broadcasting on either 2.0 or 5.0 Ghz. To extend the project, the hacking tool on the Raspberry pi should be able to extract information such as Wi-Fi network name, the Wi-Fi connected device MAC address, make of the device, software or operating system on the device. This Raspberry Pi should send all information to a computer that will store all the results.

3.4 Expectations from the project

The tool will be built on a Raspberry Pi and should automatically scan and where possible attempt to connect to any Wi-Fi network detected either using a brute force or rainbow table or any technique of choice. The tool should work on any environment either around campus or within the building. This tool must retrieve as much as information about the Wi-Fi network as possible and the devices on that network, which Wi-Fi networks are available, type of encryption, Wi-Fi network authentication type, broadcasting RF and all the information should be displayed on command prompt. Figure 1 shows some expectations from the project and they are also the functional requirements.
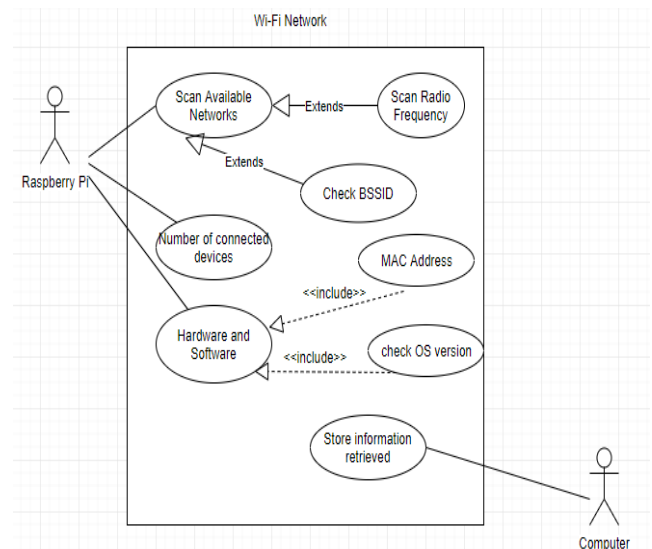

Figure 1: Use Case showing functional requirements of the project.

4 REQUIREMENTS ANALYSIS

4.1 Purpose of project

Purpose of this project is to educate campus community about Cyber Security. Educate means users must understand and comply with basic data security principles like choosing strong passwords and backing up data. The number of IoT devices has increased to point where security has to be an emerging priority. The project will determine the Wi-Fi network information and the type of devices connected to the network (hardware & software information of the devices). The project will be useful to teach Wi-Fi network owners and those who connect in it the importance of strong password encryption.

4.2 Scope of the system

The hacking tool will automatically scan and where possible attempt to connect to Wi-Fi networks with weak encryption. This tool will connect to the network using brute force. The tool built should retrieve information about network (such as what type of encryption used and type of network protocol), which networks are available, channel Wi-Fi broadcasting on and devices connected to the network. The hacking tool will not access the devices it will only give information such as the MAC address and Operating System of the device. The information about the network found weak by the tool will not be shared without the permission of network owner.

4.3 Objectives and criteria of the project

The project will be successful when these following set of objectives are met.

　　1. The Raspberry Pi is programmed to connect to Wi-Fi networks with weak encryption.
2. Raspberry Pi should manage to retrieve some information about the network and devices connect in it.
3. When the hacking tool can send all the information retrieved from the network to the external computer that will act as data center.
4. When the findings are documented and Wi-Fi network owners with weak encryption are notified.

4.4 Current system

The installation of Kali Linux and the update to the full version has been completed. Current state of the system is shown in figure 2.

Figure 2: Current system of hacking tool.

4.5 Proposed system

The hacking tool should be able to detect which networks are available and check either the encryption used is strong or weak. The hacking tool should determine which security protocol used (e.g. WPA2, WEP etc.) and it should be able show channel Wi-Fi is broadcasting on. Figure 3 shows the sequence tool will be able to accomplish all expectations.
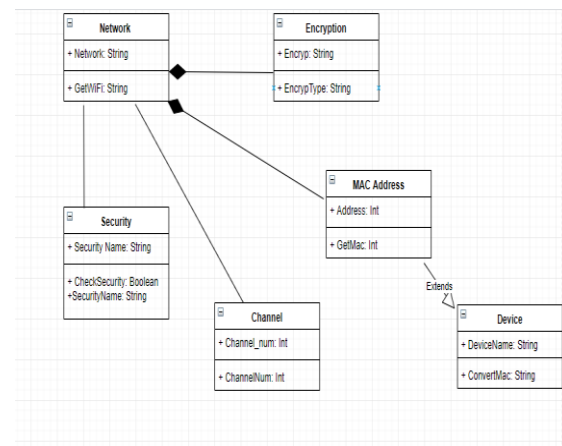

Figure 3: UML Class Diagram.

5 NON-FUNCTIONAL REQUIREMENTS

5.1 Response time

The hacking tool built should not take more 45 seconds to display available networks, type of security network is using and it should take less than 1 minutes to determine whether the network used weak or strong encryption and channel network broadcasting on.

5.2 Reliability

Equipment and tools that will be used to carry out this project will be tested separable before integrated together. The project will be tested by using dummy Wi-Fi hotpots that will be set using mobile phone.

6 USER INTERFACE SPECIFICATIONS

According to stakeholders and research done the user interface should be on command prompt, it should be a simple terminal-based application. The interface of this project will be the terminal display. The terminal should display all the information required by the project (e.g. available networks, type of security protocol network use etc.). Figure 4 shows how the user interface will be like.

Figure 4: Expected User Interface.

7   DESIGN

7.1   Data Design

The project does not have or will not require much data to work on but there will be a data store, which will be used to document the results found during the project. This data will also be used to make an analysis. When hacking is carried out, Radio Frequencies, Access Points (network name available) and security protocol the network is using to protect network system must be discovered. All of these discoveries found during hacking will be used to determine more information about the hardware and the software of the devices connected to the network.

Discovered information (networks available, radio frequencies etc.) are the first piece of data and can be describe as entry points in this hacking project.

The objective of this project is to create cyber security awareness. All the data encountered by the hacking tool must be used to document findings and alert owners of the Wi-Fis networks that were found to have weak encryption keys. Figure 5 shows data flow during the course of the project.
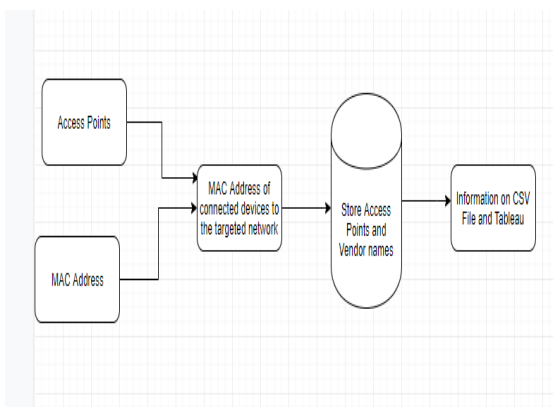

Figure 5: Data Progression During the Project.

7.2 Architectural Design

Completion of the project should result in a document which has information about Wi-Fi network that is targeted and connected devices to that certain network. All functional requirements include the document can be achieved using HCXtools. HCXtools contain a set of tools that can be used for penetration testing and monitoring.

The tool that will be used from HCXtools is AirCrack -ng [4]. Aircrack -ng is a tool to assess Wi-Fi networks security. Aircrack -ng focuses on different areas of Wi-Fi security such as monitoring, cracking, attacking and testing. This project will fall under the monitoring and cracking category of functions that Aircrack -ng provides. Aircrack -ng provide a tool named Airodump -ng. Airodump -ng is a tool that will be used to capture packets and export information to a csv file (this is achieved by using -write option) in order to use Airodump -ng we must run Airmon -ng first.

7.3   Interface Design

The stakeholders decided that the interface should be a terminal based application. During the meetings as we were discussing about the prototype, we have concluded to create a user interface changing the terminal based application. The idea is create an interface with one or more buttons where we can select any button on the interface then start penetration or retrieving some information about the network being targeted. There are many applications on Kali Linux that have a designed UI (User Interface) and we can use one of the python libraries.

7.4 High-Level and Low-Level Design

In order the project to be complicated there are some actions and steps needs to be taken. There are quite steps need to be taken, figure 6 shows high level design and figure 7 shows low level design. Low level design gives more information about the steps. The following are the steps must be taken explained in detailed.
1. Airmon-ng: First step is to start the wireless interface in monitor mode and it create new interface for to use.

2. Airodump-ng: Airodump-ng allows us to name our own interface and view details about the network such as BSSID, CHANNEL, ESSID etc. It monitors all the data for the network we are trying to capture the handshake. Allows you to enter BSSID and name of targeted network.

3. Aireplay-ng: Tool is the that make handshake successful, it uses deauth command to capture a handshake.

4. Handshake: After Aireplay-ng handshake is done and the MAC address of all connected devices are shown under the column STATION, which make it easy to know and convert those MAC addresses to vendor name, through python script.

5. Aircrack-ng: This is the tool that does penetration in this project. Aircrack-ng use the word-list to check the possible password, so it necessary to have a word-list before penetration. This may take some time especially

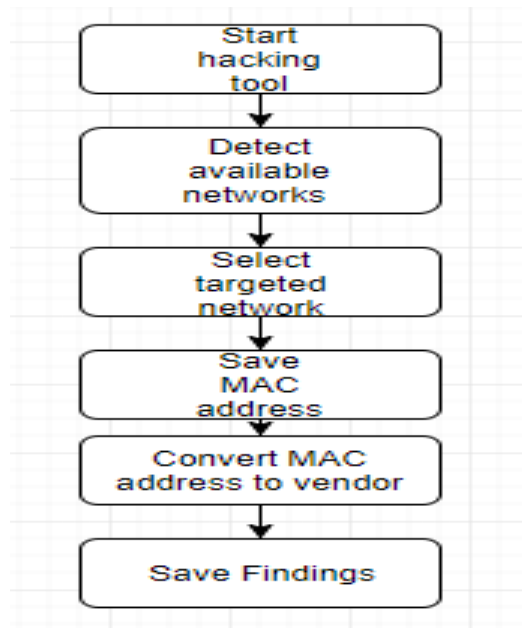on raspberry pi so the suggestion is to not use a large wordlist.
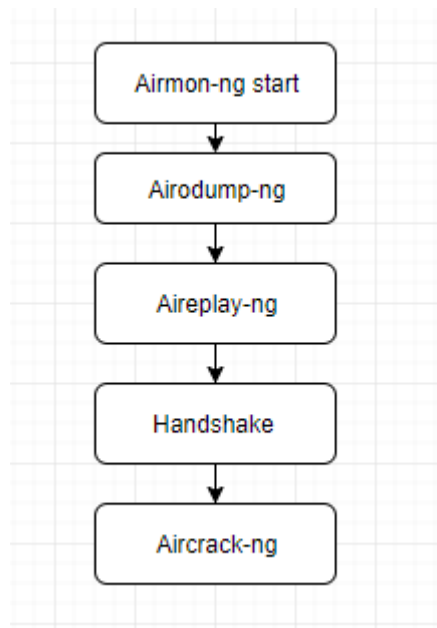

Figure 6: High Level Design.

Figure 7: Low Level Design.

8 PROTOTYPE

8.1 Evolutionary prototype

The prototype for this project is an evolutionary prototype it will evolve to the final product. The prototype will undergo a series of refinements and eventually will become a fully developed solution or product.

8.2 Hardware

The hardware components used to this prototype and they will remain until the final product or hacking tool. The hardware components consist of the following:
1. Raspberry Pi 3
2. Micro SD card
3. Desktop Monitor

8.3 Software

Aircrack-ng tool is the tool found in Kali Linux that is used to penetrate security protocols such as WPA/WPA2 and WEP. This tool has four utilities which are used in four attack phases that take place to recover the key, these utilities are Airmon-ng, Airodump-ng, Aireplay-ng and Aircrack-ng-. The application Wifite will be used for detecting access points and MAC address of connected devices to the targeted network. Most of the Wifite will be used to discover MAC addresses of connected to the targeted network.

The prototype will have the following software components:

1. Kali Linux Operating System
2. Wifite
3. Aircrack-ng

Not only these components will make the prototype successful. But Python script will also be used to convert MAC address to vendor name. The script uses API provided by Nivel Technologies Ltd. This API uses HTTP requests to their servers. Nivel technologies provide a PHP GET example that is converted to python which will be used on this project, I have provided an example in figure 8 and a python script that request on Nivel Technologies Ltd a vendor name on their servers in figure 9.
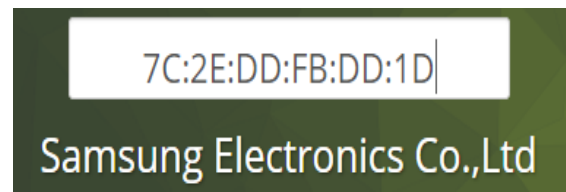

Figure 8: Vendor name from Nivel Technologies Ltd servers.

```
import requests

error = "{\"errors\":{\"detail\":\"page not foound\"}}"

URL = "http://api.macvendors.com"

macAd = "7C:2E:DD:FB:DD:1D"

PL = URL + macAd

req = requests.get(url = PL)

if req.text == error:
    print("no name Mac Adderess found")
else:
    print("MAC ADDERESS :" + macAd)
    print("VENDOR NAME :"+req.text)
```

Figure 9: Python script to request vendor name on Nivel Technoogies Ltd.

9   IMPLEMENTATION

9.1 Introduction

The project prototype is an evolutionary prototype which will go through series of stages, it will be develop to the final product. The prototype built in design phase, was designed with built in tools and other external tools available in Kali Linux, tools such as Airodump-ng, Nmap, Airomon-ng and other tools that are required to support these tools.

   The implementation phase will discard some of the tools replacing them with self-written code. The preferred language is Python. Python is one of the languages used for hacking. Python is considered as the best language for hacking because it lets you do a fast reconnaissance of the target network and makes prototyping much faster, makes it easy to write automated scripts and it's an easy-to-read language that is helpful for beginning ethical hacking[Techlog360]. Basically in this phase is the conversion of tools available on Kali Linux to Python scripts.

9.2 Software

The prototype mentioned in section 8 had set up most hardware and software needed, most was software replacement and making the functionality of some tools being accomplished with code. As mentioned the preferred language is Python based on certain reasons include being the most used language on system vulnerability testing.

   Python has libraries that can help to accomplish functionalities by using tools, these libraries can be installed through the command line. Before installing some libraries you are required to set some environment where libraries will be installed. These libraries include Python module Wi-Fi which provide read and write access to wireless network card capabilities and cell module return cell list of objects, each cell object has attributes that can be used to retrieve information about the hardware and software of the Wi-Fi network.

   There are libraries used to convert MAC Address of devices on the network. These libraries include requests which allows you to send HTTP/1.1 extremely easily. Pprint module help to pretty-print Python data which can used as input to the interpreter.

   Scapy is a Python library used to sniff and capture nearby packets between the connected devices. In the same code it also help us to identify clients in the network, but most of these library helps us to replace the tool named airodump-ng.

9.3 Hardware

The hardware used will remain the same throughout the project, these hardware include the following.

- Raspberry Pi Model 3+
- SD Card
- Monitor
- Keyboard, Mouse and Charger 5V

9.4 Functions

There are three Python scripts initially which perform different functions. These Python scripts are combined in a sense as they will perform together. The first script detects all available Wi-Fi networks available and displays all the information about each Wi-Fi network. Information detected will be saved in a csv file where the information will be considered as first or initial data for the project. The second script is the sniffing script which find all clients on networks and it is the script that replaces the tool used on previous prototype which is airodump-ng. This script replaces airodump-ng found in Kali Linux. Last script is the one that converts all MAC Addresses to vendor names of connected devices and include MAC Address of Wi-Fi devices. Previously there was no convention of MAC Addresses and there was no code that detected Wi-Fi networks and clients connected, these are the changes made in implementation phase. Most important change will occur is the wordlist that contain possible passwords, we will the increase in number of possibilities which will results increase in size and time taken to crack the passwords. Testing in this phase may took days or weeks depending on the difficulty of the password.

9.5 Testing

Testing in this phase will be different from the previous phase. In this we will try to penetrate Wi-Fi networks with differences in length, and increase the difficulty level of the password. Password we managed to crack previously contained only digits (e.g. 2568488) but in this phase we will increase the difficulty of the password by increase the length and add some alphabets (e.g. 5896zd). The final testing will be on strong password. Strong passwords contain special characters, numbers and alphabets. In the design phase we crack Wi-Fi network we had knowledge about, so now we will change password and add some alphabets and numbers on the same network. The targeted network for final testing will be unknown and will be one of networks available around campus.

9.6 Conclusion

The implementation phase is more about converting tools used on the prototype to Python code which will perform the same functionality as tools found in Kali Linux. The go about which language used and what libraries and modules being used in order to accomplish our goal which to convert tools to any programming language.

10 TESTING

10.1 Introduction

The hacking prototype should detect all available Wi-Fi networks and those networks that are hidden, give

information about each network detected and devices connected to it and where possible the hacking should give the password of the targeted network when the attack has been launched. The hacking tool built should be tested on these requirements in order to be considered done. These are the requirements the hacking tool should meet but there are other requirements the system has to meet such as saving all the information to CSV file, read the CSV file to convert MAC addresses to vendor name and able to communicate with macvendors API.

10.2 Testing Strategies

Testing strategy for this project does not involve users it is built for technical people who have basic knowledge Kali Linux and understanding of how to use a Raspberry Pi. The first strategy for this project is unit testing, Unit testing involves testing individual source codes. In this project I have used some tools available from Kali Linux and I have replaced most of them with Python. The unit testing will be done on python script created, there are four to five scripts created and these scripts include the scripts that detect Wi-Fi network hidden or not, convert Mac addresses to vendor names, create wordlist which will be used during penetration and sniffing script to capture packets between network and clients for handshake. The other testing strategy that will be used to test this hacking tool is system testing. There are different types of system testing but for this project we focus on hardware and software since there is integration between Kali Linux, Python and Raspberry Pi. The test is conducted to check the environment for Python allow all libraries and modules to be install, What Python packages came with Kali Linux and the interaction between Raspberry Pi and the Wi-Fi network.

10.3 Testing Design

10.3.1 Unit Testing
The unit testing will be testing each script but the scripts will run simultaneously starting from the detecting script because the outputs will be the information that will be needed by the converting script, the output of detecting is a CSV file that has information about each network information such as BSSID, SIGNAL, RADIO FREQUENCY, ENCRYPTION TYPE and MAC ADDRESSES. The second script will read CSV file, in CSV file it will read MAC ADDRESSES column and it will output the vendor names of each network. The other Python script which is generating wordlist will not be included in the testing due to time it take to generate a wordlist. The last script is the sniffing script that will be tested during the penetration of the targeted network. The penetration scripts contains command lines that are used for hacking a network. To include to the unit testing the hacking will be conducted using different passwords with change in different level of difficulty from weak password (which contain number only), medium password (contain number and alphabet) and strong password (contain numbers, special character and lower case letters and upper case letters).

10.3.2 System Testing
There are two software's used on the project is Python and Kali Linux. I have encountered some difficulties of installing some Python libraries some came with Kali Linux some I had to install them manually, even after I have installed them manually there were some issues such as libraries where not stored in the same directly. In this test I will test if the environment created has all the libraries and modules installed. In terms of hardware and software there issues regarding the time taken by the penetration it heat up the Raspberry Pi to a point where the Pi will be slow and crush, so the test is to check time complexity on the Python scripts and minimize the time the penetration take.

10.4 Test Cases

Testing is performed on checking the hacking tool built detect all Wi-Fi networks available and retrieve information about each network, before testing the Pi need to be on monitor mode then and the output of this is the CSV file contain information of each network. Second testing is on converting mac addresses on the CSV file to MAC vendors, the input in this is the CSV file and the output are vendor names. Third testing is check if python script capture packets between client and network. Final testing is on different level difficulty of passwords from weak password to strong password.

10.5 Test Report

The hacking tool detect Wi-Fi networks available and those that are hidden by the owners. Information about each network detected is saved on CSV file, Fig.10 show how the information in the CSV is saved. The CSV file is useful to determine which network to target, for example looking at security protocol the network is using. The Python script read MAC address column in the CSV file created by the script that detect networks and convert MAC addresses to vendor names. The wordlist contain numbers, alphabets and special characters.

SSID	ENCRYPTION	RADIO FREQUENCY	MAC ADDRESS	CHANNEL	ENCRYPTION TYPE	SIGNAL
TP-LINK_D21F	TRUE	2.457 GHz	18:D6:C7:85:D2:1F	10	wpa2	-14
UWC-Guest	FALSE	2.412 GHz	40:01:7A:AF:B9:A0	1	None	-58
UWC-CAMPUS	TRUE	2.412 GHz	40:01:7A:AF:B9:A1	1	wpa	-59
eduroam	TRUE	2.412 GHz	40:01:7A:AF:B9:A2	1	wpa	-59
WIFI-Support(Limited	FALSE	2.412 GHz	40:01:7A:AF:B9:A3	1	None	-60
UWC-WifiPortal	FALSE	2.412 GHz	40:01:7A:AF:B9:A4	1	None	-58
UWC-Guest	FALSE	2.437 GHz	40:01:7A:BE:80:40	6	None	-52
UWC-CAMPUS	TRUE	2.437 GHz	40:01:7A:BE:80:41	6	wpa	-51
eduroam	TRUE	2.437 GHz	40:01:7A:BE:80:42	6	wpa	-49
WIFI-Support(Limited	FALSE	2.437 GHz	40:01:7A:BE:80:43	6	None	-49
UWC-WifiPortal	FALSE	2.437 GHz	40:01:7A:BE:80:44	6	None	-47
UWC-Guest	FALSE	2.437 GHz	40:01:7A:BE:E2:80	6	None	-76

Figure 10: CSV file with information about each network detected.

The minimum number of a password is 4, so the wordlist contain possible passwords from 4. Testing has been conducted on the networks that have weak, median and strong password. The time taken by these three different testing are different and the difficulty level of password does not determine the time. The hacking network is time consuming in this case the minimum time taken by each attack is approximately 1 day to 3 days.

11 CONCLUSION

The project is to build a hacking tool on a small device that require small amount of power and allow you to move around. The device used in the project is Raspberry Pi since it has Wi-Fi enable, Bluetooth enable, Ethernet enable etc. The hacking tool built should automatically scan the networks available around and retrieve information such as Radio Frequency (2.5 GHz or 5 GHz), signal, BSSID, MAC addresses etc. where possible the hacking tool should attempt to connect to Wi-Fi network with weak encryption. There are quite many ways these can be accomplished, but in this project it is accomplished using Raspberry Pi and Kali Linux as operating system. There is portion of Python used to meet the requirements. Kali Linux have quite many tools to penetrate Wi-Fi networks, tools used in the project Aircrack-ng, Aircrack-ng have quite many tools such as Airodump-ng, Aireplay-ng etc. The penetration is conducted on different passwords from weak passwords to strong passwords. There are quite researches conducted on penetration of Wi-Fi networks with focus on security protocols, whenever there is a new security protocol there is research conducted on checking the vulnerabilities. The aim of this research is to educate public and enterprises on how the penetration can easily be conducted and how they can protect them self's from these cyber-attacks. Having hacker in your network may cause you serious damages such as loss of sensitive information, banking details and other information, when the hacker have this information can withdraw money from all your accounts, make loans and sell the information on third party if you have your company information. Having a strong password and changing daily may protect you from many attacks.

References

[1]   CISCO, "Security," *CISCO/Security*, 2018. [Online]. Available: https://www.cisco.com/c/en/us/products/security/what-is-cybersecurity.html. [Accessed: 14-Feb-2019].

[2]   hash3liZer, "No Title," 2018. [Online]. Available: https://www.shellvoide.com/python/how-to-code-a-simple-wireless-sniffer-in-python/. [Accessed: 05-Aug-2019].

[3]   A. L. and J. Muniz, *Penetration Testing with Raspberry Pi*. Birmingham,UK: Packt Publishing Ltd., 2015.

[4]   V. Kumkar, A. Tiwari, P. Tiwari, A. Gupta, and S. Shrawne, "Vulnerabilities of Wireless Security protocols (WEP and WPA2)," *Int. J. Adv. Res. Comput. Eng. Technol.*, vol. 1, no. 2, pp. 2278–1323, 2012.

[5]   J. F. and S. A. Tyler Williams, "security of the internet of things(iot)," *Digitalcommons.murraystate.edu*, 2017. [Online]. Available: https://www.google.com/search?rlz=1C1AVFC_enZA833ZA833&ei=myOCXMrnCeGU1fAPkqOooAI&q=security+of+the+internet+of+things%28iot%29+murray+state+university&oq=%22security+of+the+internet+of+things%28IoT%29%22+murray+state+&gs_l=psy-ab.1.0.33i160.6166.12427..1. [Accessed: 03-Mar-2019].