



WiFi Blackbox - A Tamper-proof Forensic-ready Device for Wifi Networks



A. S. Wickramsekera
University of Colombo School of Computing
2022

Declaration

I hereby declare that the thesis is my original work and it has been written by me in its entirety. I have duly acknowledged all the sources of information which have been used in the thesis. This thesis has also not been submitted for any degree in any university previously.

Student Name: **Akila Shamendra Wickramasekara**

Registration Number: 2019/MIS/027

Index Number: 19770278

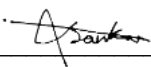
_____  _____

Signature of the Student & Date : 13-01-2022

This is to certify that this thesis is based on the work of **Mr. Akila Wickramasekara** under my supervision. The thesis has been prepared according to the format stipulated and is of acceptable standard.

Certified by,

Supervisor Name: Dr A.P Saykkara

_____  2023-01-15

Signature of the Supervisor & Date

Acknowledgments

First, I would like to convey my sincere gratitude to my project supervisor, Dr.Asanka Sayakkara for giving me immense support, constant guidance and suggestions for this project and being available for discussions in any time.

Secondly, I would like to extend my gratitude to all the academic and administrative staff of University of Colombo School of Computing for the great assistance they have given to me in numerous ways.

I would also like to thank to all the people in academia and industry who encouraged me and showed my potential.

Finally, I would like to express my gratitude to my spiritual mentor, Dr. Daisaku Ikeda, and Soka Gakkai International, for teaching me how to expand my potential and showing me the great path of wisdom.

Abstract

With the current trend of computer and communication technologies, wired computer networks are almost entirely being replaced by wireless networks. For this purpose, the IEEE 802.11 protocol, i.e., Wi-Fi, is increasingly being used. With this rapid increase in usage, Wi-Fi networks are turning into the most important target of network-based malicious activities.

It has been found that most of the successful Wi-Fi-related attacks in recent history have originated from internal threat actors, which amounts to a 60% of all attacks.

In order to make an impact on those kinds of internal Wi-Fi attacks, this research aims to create and deploy a forensic-ready device, called WiFi-Blackbox, for internal networks that acts as a passive control mechanism that monitors, stores, and communicates the activities in a Wi-Fi network and later detects anomalies and patterns from the gathered data and information.

The WiFi-Blackbox device is mainly integrated with five main hardware components and five software modules. The main board is a Raspberry Pi 4 single-board computer, which is combined with an RTL881cu USB Wi-Fi adapter that operates in WiFi monitor mode to observe nearby Wi-Fi networks constantly. To make the device tamper-proof, various physical and technical controls have been applied, such as an reliable internal power supply to face power failures and a backup 4G modem to face network failures. The device is also equipped with a GPS sensor to detect the location of the device, which can be used to prevent theft. The enclosure open notifier is another technical security control that detects physical tampering attempts on the device by attempting to open the sealed enclosure of the device. All the hardware components are controlled by internal software modules, which were written with a mixture of Python, PHP, and Shell scripts. The Sniffing module connects with the W-FI adapter and sniffs all the data frames, stores relevant frames, and decrypts them where necessary. The Anomaly Detection module checks for anomalies in each data frame and notify the user. It uses a third-party virus database to detect viruses, Trojans, and other malware. At the same time, it detects anomalies such as denial-of-service attacks. The devices Backup and Maintenance module is responsible for keeping backups of captured frames, while the System Monitor module helps to maintain a healthy system. The Communication module is the key to securely communicate between multiple WiFi-Blackbox devices and notify about incidents to each other..

The evaluation was done in a controlled environment with known infected files. Furthermore, stress testing was done by using 10 Wi-Fi networks in the deployed environment. The results indicate that a continuous data capturing and anomaly detection takes the maximum CPU power and reaches a CPU heat of 86C. Also, the device's data accuracy is about 99%, and the data corruption rate is less than 1%. This research indicates that internal Wi-Fi networks can be protected and WiFi-related activities can be tracked and saved for forensic purposes. At the same time, it shows the possibility of implementing a tamper-proof network monitoring platform and how to demotivate insider attackers by representing the device as a deterrent controller.

Contents

Chapter 1	4
INTRODUCTION	4
Overview	4
Background to the Research Study	4
Aim and Objectives	7
Scope and Limitation	7
Chapter 2	9
LITERATURE REVIEW	9
Overview	9
Raspberry Pi 4	9
Wireless Attacks	10
Insider Threats	12
Physically Unclonable Function (PUF)	12
Related Studies	13
Chapter 3	15
DESIGN AND DEVELOPMENT	15
Overview	15
High-Level Architecture	15
Design and Development of Hardware Parts	16
Main Board	16
WiFi Module (WM)	17
Power Management Unit (PMU)	18
Ceaseless/Continues Operation Manager Module	19
General Purpose Input Output (GPIO)	19
Outer Casing.	19
Design and Development of Software Modules	20
Sniffing Module (SM)	20
Anomaly Detection Module (ADM)	21
Communication Module (CM)	22
Backup Maintenance Module (BMM)	23
System Monitoring Module (SMM)	24
The database	25
Chapter 4	26
IMPLEMENTATION	26
Implementation of Hardware Device	26
Security Controls	28
Technical controls	28
Physical and Technical controls	29
Physical controls	29
Implementation of Software Modules	29

Graphical User Interface (GUI)	32
Final Product	35
Chapter 5	37
EVALUATION	37
Testing Phase 1	37
Testing Phase 2	38
Testing Phase 3	38
Testing Phase 4	39
Testing Phase 5	40
Testing hardware system	41
Chapter 7	43
CONCLUSION AND FUTURE WORK	43
Conclusion	43
Future Work	44
Bibliography	45

List of Figures

1. The distribution of reported attacks	9
2. Physical appearance of and components of Raspberry Pi 4.	13
3. Logic of time-delay based silicon PUF	15
4. High-Level Architecture	18
5. Components and Modules in WiFi Blackbox	19
6. Physical view of TP-Link TNWN821-N adapter and RTL881cu adapter	20
7. Usage of GeekPi Raspberry Pi UPS Hat	21
8. Battery Level Indicator of GeekPi Raspberry Pi UPS	22
9. Flow Chart of the Sniffing Module	24
10. Flow Anomaly Detection Module	25
11. Network Diagram of the Communication Module	26
12. Sample Data Object Communication within devices and cloud system	26
13. ERD of the table structure	29
14. 200 X 120 X 55mm Waterproof hard Plastic Enclosure	31
15. Transverse plane view of the hardware device	31
16. Flow of the PUF generation	33
17. How the config params should be defined	34
18. 4-way handshake	35
19. Decrypter code with Tshark commands	36
20. Landing Page of the Wifi- Blackbox GUI	38
21. GUI to manage protocol rules	39
22. GUI for the system health	39
23. GUI for the interface of the file	40
24. Final outlook of the product	41
25. Output of a phase 1 PCAP file open by Wireshark	42
26. Decrypted data frame opened in Wire Shark after the testing has been conducted	43
27. Captured data frames of DOS attack in wire shark software	44
28. Captured PCAP file with the infected file opened in Wireshark	44
29. Virus total API report	45
30. SMS trigger alert messages	46

List of Tables

1. Potential Attacks For WLAN	11
2. Technical specifications of Raspberry Pi 4 B	13
3. Main features and specifications GeekPi Raspberry Pi UPS	22
4. Key Features in the Cloud-Based Operational System	28
5. Functional features of the GUI	38

Chapter 1

INTRODUCTION

Overview

This chapter will outline the research background, motivation to the research, research objectives, scope, and also the limitations.

Background to the Research Study

With the digitalization of data, Information Security is becoming a key aspect in most industries and playing a vital role in the Information and Communication Technology sector. With the growth of data breaches and cyber attacks, most organizations are now putting more effort into protecting data and information against attacks. In recent history, statistics show that more than 60% of successful attacks have come as internal attacks to organizations (May 15 and 2020, 2020). Also, 63% of the internal security incidents occur because of the negligence and ignorance of the insiders who are responsible for the safety and security of the organization's data ("2020 Cost of Insider Threats Global report," 2020). In light of such a situation and the increasing growth of network-based attacks, most companies are now considering various physical and technical controls to overcome internal threats. In order to minimize information leakages from inside of an organization, companies also apply administrative controls. However, it is hard to eliminate human errors and mistakes through such administrative measures, which is still a considerable problem.

According to the Insider Threat Global report in 2022, there are 3 types of insider threats ("2020 Cost of Insider Threats Global report," 2020) as follows.

1. Negligent Insiders
 - These threats have been most common throughout recent years. as a percentage, it is about 56%, and the foremost reason for this threat is the negligence or the lack of awareness of the employees on information and responsibilities they handle.
2. Negligent Insiders whose accounts were stolen.
 - Attacks on credential theft, which amounts to 18% of attacks in 2022, pose a risk of losing accounts of insiders of an organization, which can occur due to intentional or unintentional leakage of login credentials.
3. Malicious Insiders
 - Insiders of an organization itself can act with malicious objectives posing a threat to the data of an organization. Over 26% of threats have been posed by such malicious insiders.

Figure 1 summarizes the collection of 6803 cases of insider attacks in 2022 as revealed by the 2022 Cost of Insider Threats: Global Report.

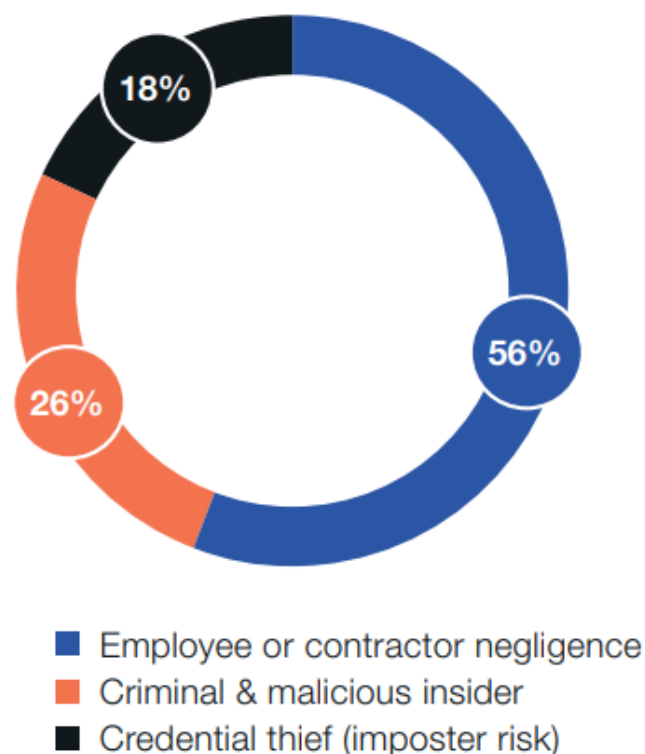


Figure 1: The distribution of reported attacks (“2022 Cost of Insider Threats Global Report,” 2021).

Due to the ease of deploying and maintaining networks, many companies use wireless technologies, such as Wi-Fi for their local area network (LAN), which uses multiple parts of the IEEE 802 protocol. Even though Wi-Fi creates an easy solution for data transfer, it opens up a new range of threats and vulnerabilities (“IEEE SA - The IEEE Standards Association - Home”) to an organization as there is a potential that anyone can easily eavesdrop on the data transferred via the Wi-Fi network. At the same time, this might create a backdoor for intruders, network worms, and trojan attacks. With worms and trojans, there is a high possibility to infect network-connected devices with malware or ransomware causing massive losses to a company. A most recent study shows that although Wi-Fi access is protected with encryption mechanisms, such as WPA2, unauthorized network access is still possible in most cases (“IEEE SA - The IEEE Standards Association - Home”, 2022).

Various attacks have been demonstrated on Wireless Local Area Networks (WLANs), each involving different risk factors. Table 1 depicts some of such attacks on WLANs (Salem et al., 2008).

Attack Name	Description	Attack Type
WEP shared key cracking	802.11 shared key authentications with a cracked shared key or default WEP keys.	Authentication Attacks
WPA-PSK Crack-ing	recovering a WPA PSK from captured key hand-shake frames using dictionary attack tools	Authentication Attacks
Application Login Theft	Capturing application layer credential information such as usernames and passwords by capturing string values	Authentication Attacks
AP Theft	Removing an Access Point from public access.	Denial of Service Attacks
RF Jamming	RF Jamming refers to transmitting some noise at the same frequency as the target WLAN.	Denial of Service Attacks
802.11 Beacon Flood	It will generate many counterfeit 802.11 beacons, making it hard for stations to find a legitimate AP.	Denial of Service Attacks
802.11 Data Deletion	Jamming an intended receiver to prevent delivery while simultaneously spoofing ACK frames for deleted data frames	Denial of Service Attacks
Intercept TCP sessions/SSL, SSH tunnels	Intercept TCP sessions or SSL/SSH tunnels in the evil twin AMan-in-the-Middle (MiM)P	Man-in-the-Middle Attacks
Evil Twin Access Point	It is masquerading as an authorized AP by beaconing the SSID to lure	Masquerading

Table 1. Potential Attacks For WLAN(Salem et al., 2008)

Even with technical controls such as Firewalls, Intruder Detection Systems (IDS), Intruder Prevention Systems (IPS), and connecting using a Virtual Private Network (VPS), it is challenging to keep an eye on the internal wireless network for malicious activities of insiders. It will be tricky to monitor and identify insider attackers in a Wi-Fi network as all the users were pre-authorized and identified as trusted users in the Wi-Fi network.

In order to identify such Wi-Fi network activities, there should be continuous monitoring and reporting. This research's motivation emerges from this essential requirement for Wi-Fi networks, that should be efficiently implemented within any Wi-Fi network, in any organization.

Aim and Objectives

This research aims to create and deploy a forensic-ready device for internal networks that acts as a passive control mechanism that monitors, stores, and communicates the activities in a Wi-Fi network and later detects anomalies and patterns from the gathered data and information.

Scope and Limitation

The solution consists of hardware and software components that will make an embedded tamper-proof device to capture Wi-Fi traffic. The device will design to be tamper-proof to protect captured Wi-Fi data. Furthermore, Physical Unclonable Function(PUF) will implement, which serves as a unique identifier for the device. This PUF will make a physical fingerprint for each device, making the device unclonable. Even the core feature of the system is to capture Wi-Fi data and store them, the system is competent in automatically detecting a limited number of attacks, and for this, it will use some third-party APIs.

Chapter 2

LITERATURE REVIEW

Overview

This chapter discusses the background and aspects related WiFi forensics from previous studies, Wi-Fi attacks and impacts, and physical devices and methods used to mitigate them.

Raspberry Pi 4

Raspberry Pi is a single-board compact computer designed to perform computing tasks in an efficient manner. It is capable of running Linux-based operating systems, and consists of a processor with ARM architecture (Ghael, 2020). It is widely used for research and IoT-type projects because of its capability of handling inputs and outputs easily by using its general-purpose input/output (GPIO). Table 1 shows the technical specifications of Raspberry Pi 4 B.

Processor	Broadcom BCM2711, Quad-core Cortex-A72 (ARM v8) 64-bit SoC @ 1.5GHz
RAM	8GB LPDDR4-3200 SDRAM
Bluetooth	Bluetooth 5.0, BLE
Wi-Fi	2.4 GHz and 5.0 GHz IEEE 802.11ac wireless
USB:	2 USB 3.0 ports; 2 USB 2.0 ports
Ethernet:	Gigabit Ethernet
HDMI:	2 × micro-HDMI ports (up to 4kp60 supported)
Storage:	MicroSD Card Slot
Power Supply	5.1V 3A USB Type C Power
Dimensions:	85.6mm × 56.5mm

Table 2. Technical specifications of Raspberry Pi 4 B (Ghael, 2020).

As Raspberry Pi contains a considerable amount of capabilities and its ease of customization, many IoT projects use this single board computer. At the same time, the capability of using many operating systems and many Linux-based tools has made it more straightforward to use as a network monitoring tool. As it uses very little power, it will be an ideal device for a long-run and cost-effective system.

This kind of flexibility makes creating a network monitoring device with Raspberry Pi ideal (Joshi et al., n.d.) Figure 2 shows the physical appearance of and its components.

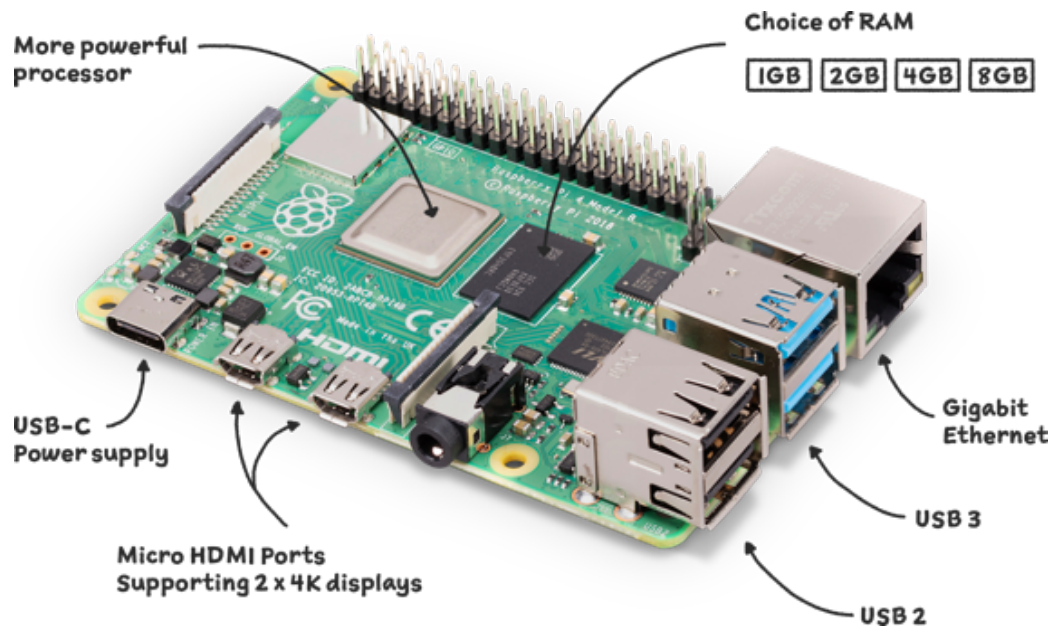


Figure 2: Physical appearance of and components of Raspberry Pi 4

Wireless Attacks

Wireless networks are more vulnerable to outside access as a physical means does not protect them. Attacks can be conducted on air, and the attacker only needs to have access to the transmission range of the specific Wi-Fi device or router. Various significant Wi-Fi attacks have been used to gain access to internal information (Hurley et al., 2004). The following is a list of common attacks on Wi-Fi networks as illustrated in the literature (Neu, 2016)

1. Packet Sniffing
 - Intruders may access an authorized Wi-Fi network and gasp the network traffic to gain any information being transferred through it.
2. Rogue Access Point
 - Plant an unauthorized access point in the network, to which users may automatically connect. When a user is connected, he/she may be a victim of ARP poisoning, packet captures, and Denial of Service attacks.
3. Man in the Middle Attack
 - An intruder may connect to an internal Wi-Fi network, make all packets pass through his/her device, and capture or manipulate the information being transferred. .
4. Jamming

- Jamming refers to making an access point unavailable to be used by its users. Jamming is done through several methods, such as flooding access points with deauthentication frames.
5. War Driving
- War Driving is the process of attackers seeking for vulnerable access points. This seeking process will be a physical event in which the attackers might move from place to place to seek the potential attackable access point.

With these kinds of wireless attacks, hardening and securing the wireless network is an absolute necessity. In order to do the security hardening, many methodologies are used. Making wireless networks 100% secure is an impossible task as there might be thousands of ways that an intruder or hacker might find to peep into the wifi traffic. However, there are significant methodologies that we can use to reduce the incidents in the wireless network and make it secure to an acceptable extent. One method is Preventing Rogue Access Points (APs). Connecting an unauthorized rogue access point to a network will create a safe passage for an intruder. In order to reduce this risk, there should be reasonable physical control for the equipment, and at the same time, a proper wireless security policy has to be defined. Also, limiting the number of MAC addresses per port will give good protection against Rogue Access Points (Mallery and Kelly, 2005).

Hardening Wireless Access Points will be another major aspect in wireless security. To achieve this there are five sub-steps defined in the literature (Mallery and Kelly, 2005):

1. Hardening remote administration
 - This step describes disabling the remote access for the device or changing the default username and password of the access point.
2. Configuring the Service Set Identifier (SSID)
 - In typical behaviour, any access point will broadcast its SSID, a unique identification name for the particular AP. Eliminate the risks related to SSID broadcast, it recommends disabling the SSID broadcast from the AP, and there are many tools which can grab the SSIDs and other information pertaining APs.
3. Configuring logging
 - Like in firewalls, enabling logs in the wifi AP will give better transparency against the network activities, which will be an excellent detective and deterrent control mechanism.
4. Configuring services
 - By configuring common services, such as Simple Network Management Protocol (SNMP), Network Time Protocol (NTP), and Dynamic Host Configuration Protocol (DHCP) will give additional protection to the wireless network.
5. Configuring wireless mode
 - Most access points support operating in 802.11a, 802.11b, and 802.11g wireless protocols; enabling only one protocol the users need will make the AP more secure.

Harding the wireless network will make the wireless network safer from intentional attackers and intruders. Nevertheless, it is hard to mitigate the risk of unintentional attacks and incidents from insiders or authorised users. Even if we add many more technical controllers and physical controls still, there might be a considerable gap between human factors and the controls (Bhagyavati et al., 2004).

Physically Unclonable Function (PUF)

A physically unclonable function (PUF) refers to a specific security measure used to protect the uniqueness of any hardware item. In other terms, a PUF will provide a unique ‘fingerprint’ for hardware items during manufacturing. Copying or guessing this unique fingerprint is computationally infeasible as it consists of micro and nanoscale hardware fabrications. There are several common types of PUFs that are used for the hardware defences.

1. Optical PUF or Non-silicon PUFs
 - a. Optical PUF builds using the doping method, which integrates with light scattering particles. A random and unique speckle pattern will be generated when a laser beam hits the material. Doping will be very rare to be repeated as the particle pattern is fully randomized with scattering particles.
2. Arbiter PUF (APUF) or time-delay based silicon PUFs
 - a. The method of this PUF uses a time gap or the frequency to create uniqueness. It will use a series of logic gates with an oscillator. Uniqueness is achieved by changing the set of gates' number of stages, as shown in Figure 3.

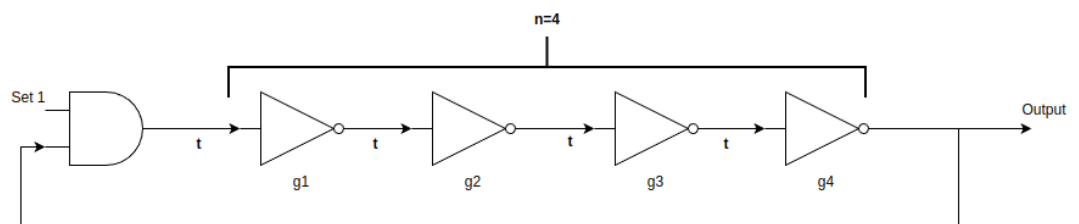


Figure 3 : Logic of time-delay based silicon PUF

$$\text{Frequency (f)} = \frac{1}{2}(\text{Number of NOT gate stages}(n) \times \text{Delay of each stage (t)})$$

3. Static random-access memory PUF (SRAM PUF) or intrinsic silicon PUFs
 - a. PUF is made at the manufacturing level of the SRAMs. Every transistor has its uniqueness, even if it is the same model and the same behaviour. This unique feature might be the threshold values or the gain factors. With this property and concatenation, thousands of transistors will make SRAMs their unique fingerprint (“SRAM PUF,” 2020).
4. RF PUF
 - a. RF PUF uses electromagnetic waves with unique frequencies to make a device unclonable or create its uniqueness.
5. Butterfly PUF
 - a. By using two latches or flip-flops the Butterfly PUF produces its uniqueness. It’s more similar to time delay PUF and SRAM PUF.
6. Magnetic PUF
 - a. It uses a magnetized perpendicular layer of MgO layers. Various kinds of skinny MgO layers will result in unclonable random patterns of magnetization within the coating, and this feature will be used as the unclonable behaviour (Chen et al., 2018).

Related Studies

In previous studies on insider and wireless attacks, researchers have attempted a couple of methodologies to identify them and address a better solution to reduce them. Some use profile monitoring, and some methods involve strong analytical strategies.

One study observed that the most common inside attacks are unauthorized extraction, duplication, or exfiltration of data, unauthorized data changes, destruction of critical assets and downloading from unauthorized sources. Also, another set of common attacks is the use of pirated software that might contain backdoors or malicious code, eavesdropping, spoofing and impersonating other users, misuse of resources for non-business related or unauthorized activities, and purposefully installing malicious software (Salem et al., 2008). To detect those inside attackers, a couple of methods were suggested in the same study. Some of the used methods were host-based user profiling (Salem et al., 2008). Using network-based sensors and a hybrid approach of network and user profiling gains some scientific results to detect those attacks. One study collected HTTP, SMB, SMTP, and FTP traffic over 13 months using the network profiling technique, and they used a Bayesian network to rank the insider threats. The authors conclude that identifying specific network events was difficult within a short period. The same study concludes that monitoring and profiling should be conducted for months and years or even decades (Salem et al., 2008). One comparative study discussed standard wireless scanning and monitoring tools such as KisMAC, NetSpot Mac OS X, Wi-Fi Scanner, and Wi-Fi Explorer (Singh and Kumar, 2018). Authors conclude this study by committing that users can troubleshoot issues like internet connectivity, speed, and congestion issues (Singh and Kumar, 2018).

In another research that uses Wi-Fi Network Signals as a Source of Digital Evidence, the authors explored the current status of digital forensics concerning new upcoming technologies. They also discussed the need for wireless protocols and technologies at various digital forensic investigations. Their study elaborates that two main areas need to be considered when doing an investigation. They were referred to as 'live' and 'post-mortem' spaces of an investigation (Turnbull and Slay, 2008). Live components refer to the state of the Wi-Fi device as in the current situation, and post-mortem information of an investigation means the extracted and analyzed data from the Wi-Fi devices. As per their study, these core components play a significant role in any analysis or investigation (Turnbull and Slay, 2008).

Wi-Fi network signals have been identified as a significant component of Digital Evidence. Because of the omnipresence of Wi-Fi, it is natural to have the potential for its misuse. With this kind of potential, the research area is still open to investigating a new kind of protocol and preventing such misuse. These will be necessary to extract and analyse the evidence from those incidents. At the same time, the same study ensures that the evidence is collected and preserved only by the specialties, and it will require technical and trained personal experiences for that task.

Even in highly secured environments and well-maintained security infrastructures, it will be an essential requirement to keep the audit logs of users' activities, critical systems, and events. Even though more automated systems are implemented, the system administrator is responsible for keeping audit logs and other forensic evidence. Relying upon one job role or person will create another threat level for post-incident activities such as recovery and investigations. In such cases, it will be hard to investigate the correct and untampered data by an investigator or an auditor ("Digital Evidence and Computer Crime - 3rd Edition", 2011).

Chapter 3

DESIGN AND DEVELOPMENT

Overview

This chapter discusses the research methodology and product design and development. It will cover detailed information on the analysis, development and deployment phases.

High-Level Architecture

A cluster of Wi-Fi BlackBoxes will enhance the security of a Wi-Fi network as it will communicate information such like previously identified anomalies and incidents. Figure 4 depicts how the Wi-Fi BlackBox cluster and its feature (labeled as "Forensic Ready Device" in the figure) will function in a network.

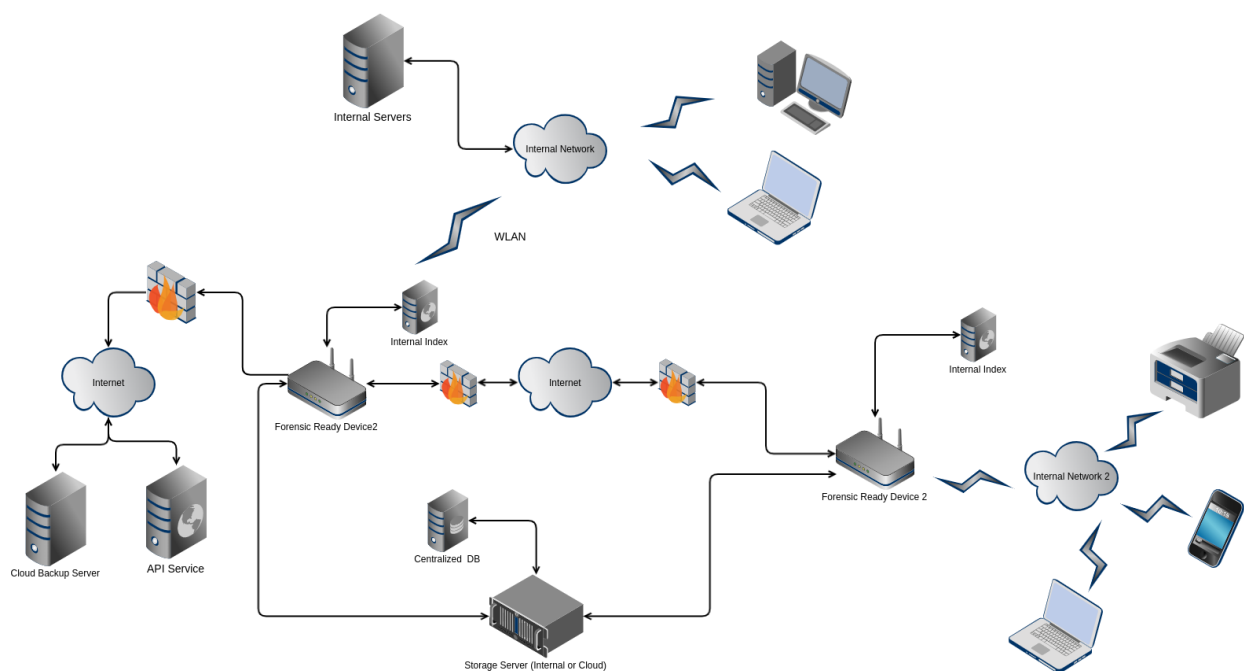


Figure 4 : High-Level Architecture

The development of the system involves two major parts as shown in Figure 5, which are the hardware part and the software part. The hardware part is further categorized into five components which are;

- Main Board,
- WiFi Module (WM)
- Power Management Unit (PMU)
- Ceaseless/Continues Operation Manager Module (COMM)
- General Purpose Input Output (GPIO) Unit
- Outer Casing

The software part was further categorized into five major components. They are;

- Sniffing Module (SM)
- Anomaly Detection Module (ADM)
- Communication Module (CM)
- Backup Maintenance Module (BMM)
- System Monitoring Module (SMM)

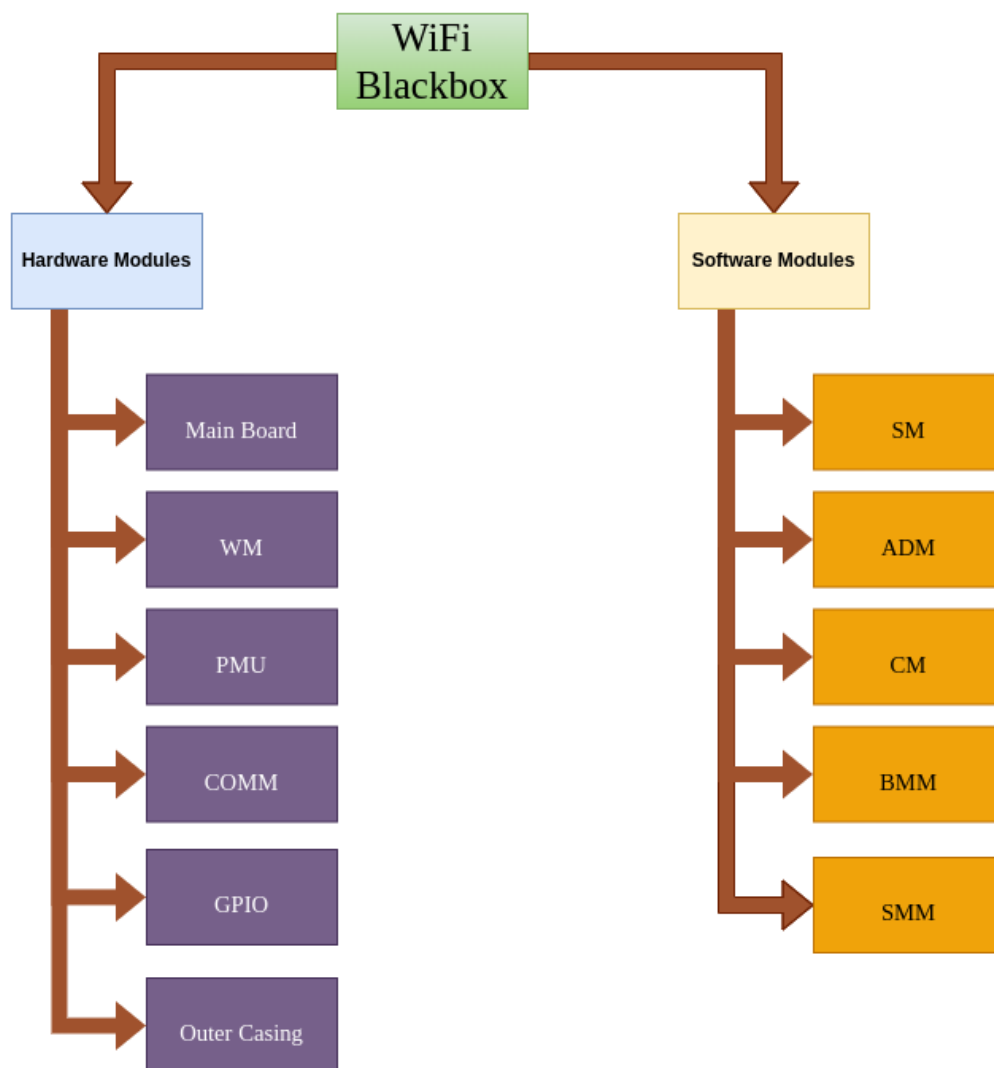


Figure 5 - Components and Modules in WiFi Blackbox

Design and Development of Hardware Parts

Main Board

Raspberry Pi 4 was used as the main board of the hardware system. It contains 2Gb of Random Access Memory (RAM) and a 1.5 GHz quad-core processor. It also uses Advanced Reduced Instruction Set Computing (ARISC), which handles processes faster with little power. The form factor is about 85.6 mm × 56.5 mm, making it portable and easy installation for the Wi-Fi BlackBox. Another critical point of raspberry pi is that it contains an inbuilt 40-pin GPIO which is used to manage external Input and Output of the Wi-Fi Blackbox.

WiFi Module (WM)

Wi-Fi Module is the core part of the Wi-Fi BlackBox as it is the main component to grab Wi-Fi data frames. Raspberry Pi single board computer contains an inbuilt 2.4 GHz and 5.0 GHz IEEE 802.11b/g/n/ac wireless Local Area Network (LAN). This inbuilt Wi-Fi module aims to connect to LAN and keep the network connectivity. However, for this project, the author used an External Wi-Fi adapter as the Wi-Fi Module as the author had to use Wi-Fi monitor mode to capture data frames in the air. In that case, it configured the built-in Wi-Fi module to keep the continuous internet and the External Wi-Fi adapter running in monitor mode to sniff and capture Wi-Fi data.

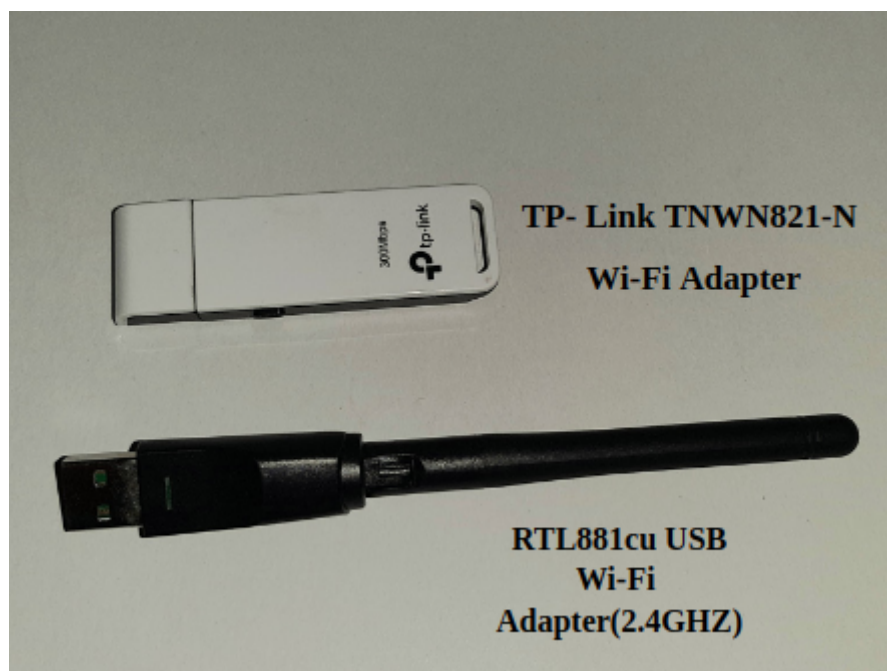


Figure 6: Physical view of TP-Link TNWN821-N adapter and RTL881cu adapter

An external USB Wi-Fi adapter was used to check the monitor mode's feasibility and behaviour. TP-Link TNWN821-N was the initial adapter which was connected and tested out. TP-Link adapter worked fine with Complex Instruction Set Computing (CISC) architecture but couldn't operate in Reduced Instruction Set Computer (RISC) architecture as its chipset was incompatible with RISC architecture. Since the system is planned to operate with Raspberry Pi, it requires an adapter that works with RISC architecture. Hence RTL881cu Wi-Fi adapter was used to test out with Raspberry Pi. With the Raspbian O/S, the RTL881cu Wi-Fi adapter worked successfully and was detected as an external Wi-Fi adapter. Hence it was proven that RealTek Limited (RTL) chipset was the best-suited chipset for this project. Figure 6 depicts the physical view of both Wi-Fi adapters.

Power Management Unit (PMU)

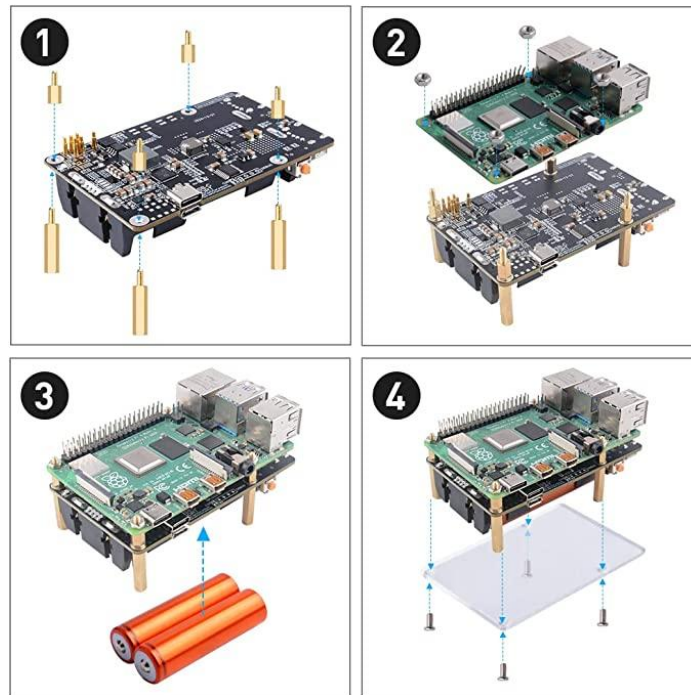


Figure 7: Usage of GeekPi Raspberry Pi UPS Hat

An uninterrupted power supply (UPS) is a must for the forensic-ready device. Hence there should be a suitable backup power supply for the Raspberry Pi. To gain this purpose, GeekPi Raspberry Pi UPS Hat was used as shown in Figure 7. It contains two INA219 chips onboard, which help to detect battery voltage. It can be directly connected to Raspberry Pi with a 5V output voltage and 4A max current load, as shown in Figure 9. With the help of two 18650 Batteries, it will supply continuous power to the Raspberry Pi. This module consists of four LED indicators showing the battery level, as shown in Figure 8. The main features and specifications of this module were explained in Table 3.

Features	Specifications
Enhanced power management	Current/voltage monitoring of Raspberry Pi power supply port.
Remote OTA firmware upgrade	Battery terminal current/voltage monitoring supports two-way monitoring of charge and discharge.
Programmable BACK-TO-AC auto power up	Independent RTC function.
Programmable sample period	OTA function (supports forced upgrade mode and active upgrade mode).
I2C communication	Power estimation

Table 3: Main features and specifications GeekPi Raspberry Pi UPS

The module contain 2 USB power outputs and 1 Type C power output which was an added advantage to get power for the backup internet connectivity used in this project.

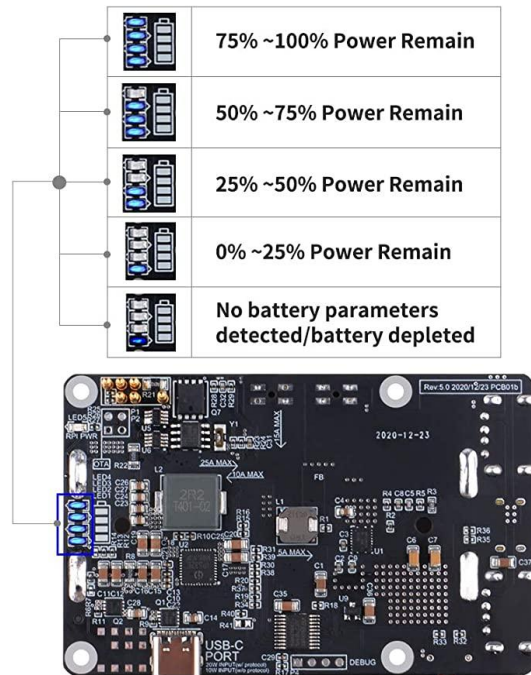


Figure 8: Battery Level Indicator of GeekPi Raspberry Pi UPS

Ceaseless/Continues Operation Manager Module

This hardware module's primary purpose is to maintain the adequate health of all hardware parts and maintain continuous operational support for the entire system. It consists of a Cooling fan and a High-Speed Downlink Packet Access (HSDPA) modem. Fan operates at 5V, which was connected to Pin 2(5v) and Pin 6(Ground) of GPIO of Raspberry Pi. HSDPA modem connected to USB of Geepi UPS.

General Purpose Input Output (GPIO)

Even the GPIO is called a separate module; it is an inbuilt feature in Raspberry Pi. GPIO is a crucial element for this project as it handles the outside input signals for the device. In this system author used 5v Pins, Ground Pins, TX/RX Pins for serial communication and General Input Pin to trigger a signal. Figure 9 shows the use of GPIO pins in Raspberry Pi. Global Positioning System (GPS) module, Reed Switch and DC Fan were connected with the GPIO and will elaborate more in future chapters.

Outer Casing.

A hard plastic outer casing was used to protect the inside hardware components. It is Waterproof and Dustproof up to the level of IP65. It contains four screw drives in the top lid and gives good protection for the internal components. The casing size is 200mm in width, 120mm wide and 75mm in height. In order to cool down the internal components and maintain good airflow, tiny holes were drilled nearby the DC Fan, and still, the protection was up to IP63 level.

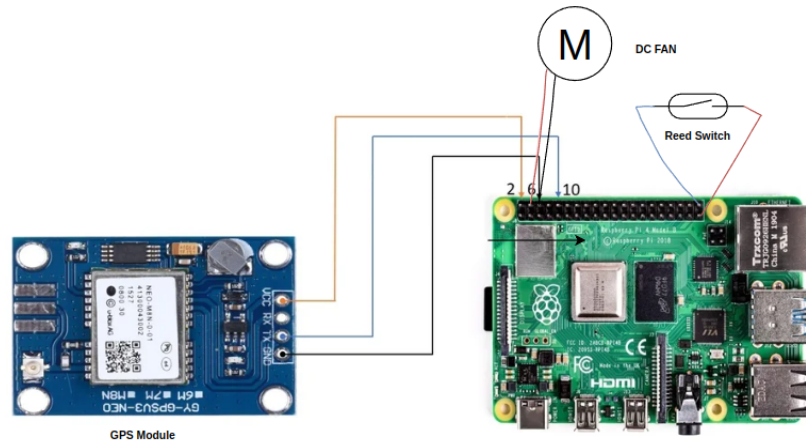


Figure 9: Use of GPIO pins in Raspberry Pi

Design and Development of Software Modules

Wi-Fi Blackbox consists of 5 main software modules. Sniffing Module (SM), Anomaly Detection Module (ADM), Communication Module (CM), Backup Maintenance Module (BMM) and the System Monitoring Module (SMM) are communicating with each other and maintaining the functions of the Wi-Fi BlackBox.

Sniffing Module (SM)

SM is the primary software part of the system. It is responsible for sniffing and storing the data frames in the wifi network. The sniffing module will capture the data frames, decrypt them, and then store them in the storage after the secure zipping the frame. It also maintains an index for the particular data frame in a MySQL Database. It will also generate a hash value for the particular captured data frame, which will preserve the integrity of the captured data fame.

The module was developed using Python and PHP and run as a continuous service on Raspberry Pi. In order to monitor the Wi-Fi networks, the Airon-ng script was used and to decrypt the data frames Airocrak script was used. To filter out captured data frames, Tshark was used. A major challenge was to capture the 4-way handshake of the particular Wi-Fi network. Without a successful 4-way handshake, it is impossible to decrypt the captured data frames, which will remain in IEEE 802.11 encryption. In order to capture the 4-way handshake author used a de-auth attack on the connected devices. This will be a time function that will make users disconnect from the particular Wi-Fi

network and connect back. This process takes less than a second, and the user will not notice it. Figure 09 elaborates on the flow of the SM.

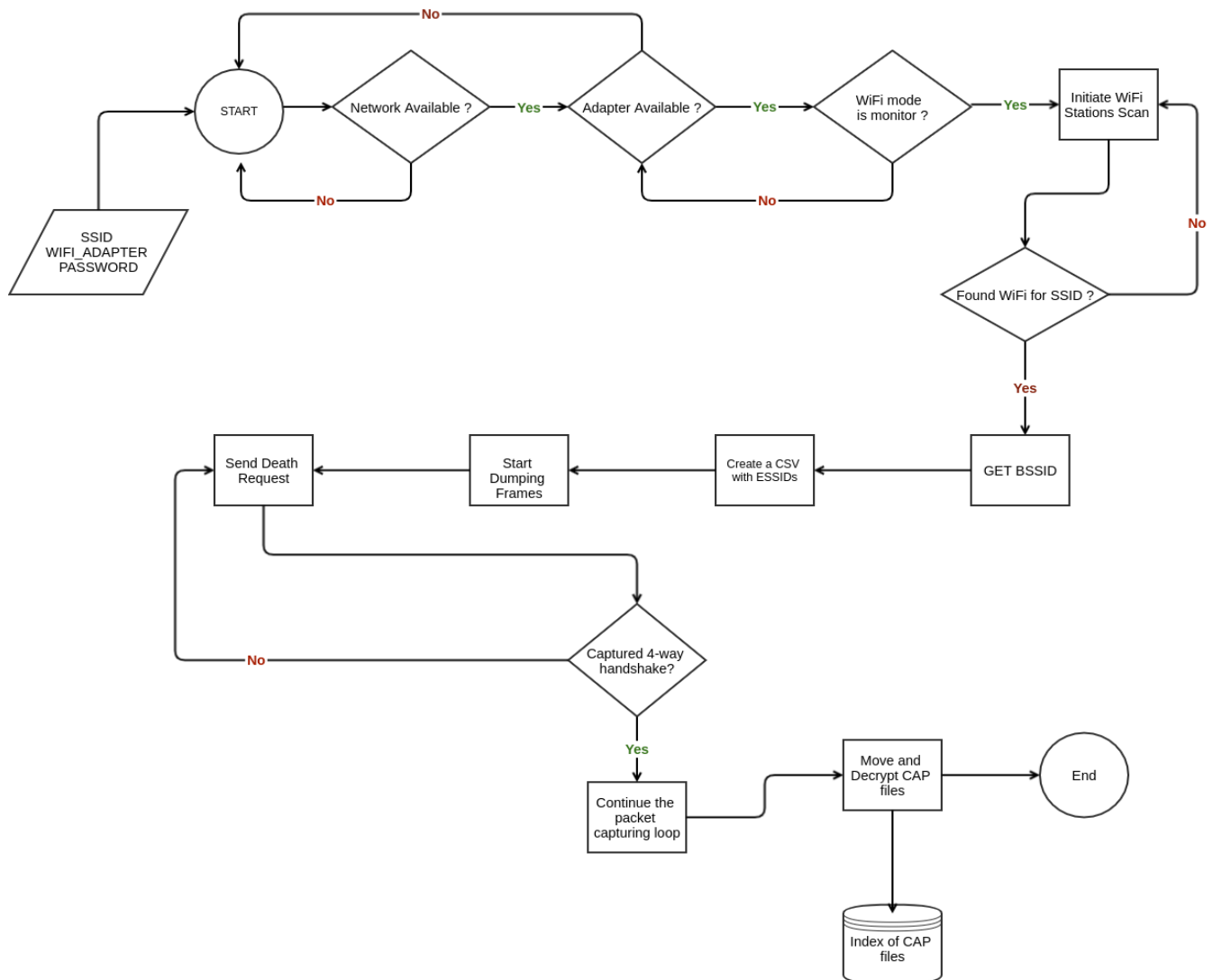


Figure 09: Flow Chart of the Sniffing Module

Before the process starts, it will require configuring five main variables in the code configuration file. They are as below

1. Service Set Identifier (SSID), which is used to identify the Wi-Fi network to be monitored.
2. Wi-Fi Password, which is used to decrypt the data frames captured from the Wi-Fi network.
3. Wi-Fi Adapter key, which is used to configure the monitor mode for the Wi-Fi adapter.
4. File Location for the airodrop CSV files, which will save after the capturing.
5. Password for the zip file encryption, which will use in the encryption process of the zip file. This will also be used in the file hash generation.

Anomaly Detection Module (ADM)

Another primary module in the Wi-Fi BlackBox is ADM. This module's behaviour determines the deviations or anomalies the user gave. This module acts as a service and will scan all the captured PCAP files by SM. ADM service will run every 5 minutes and process PCAP files for the time being. It will check for the rules which the admin user created. The rule will be a set of instructions to filter out data frames by time, destination IP address, Source IP address and protocol. Once the ADM finds matching data inside a PCAP file, it will create an Incident and alert as per the criticalness of the incident. ADM is a PHP service and it will run every 5 minutes by the corntab. Figure 10 explain the flow of the ADM.

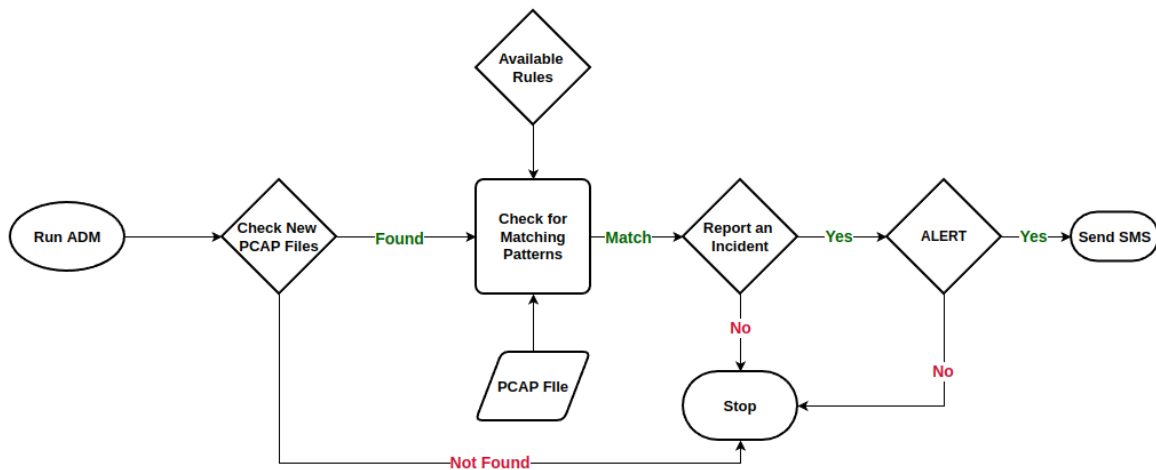


Figure 10: Flow Anomaly Detection Module

Communication Module (CM)

After detecting an anomaly or any incident communication module will do the communication part among Wi-Fi BlackBox devices. Via the CM, it will pass messages within the devices in the same network or within sub-networks in the same institution or company. CM will also keep the communication within the cloud-based system, which will keep the subscription information about every device.

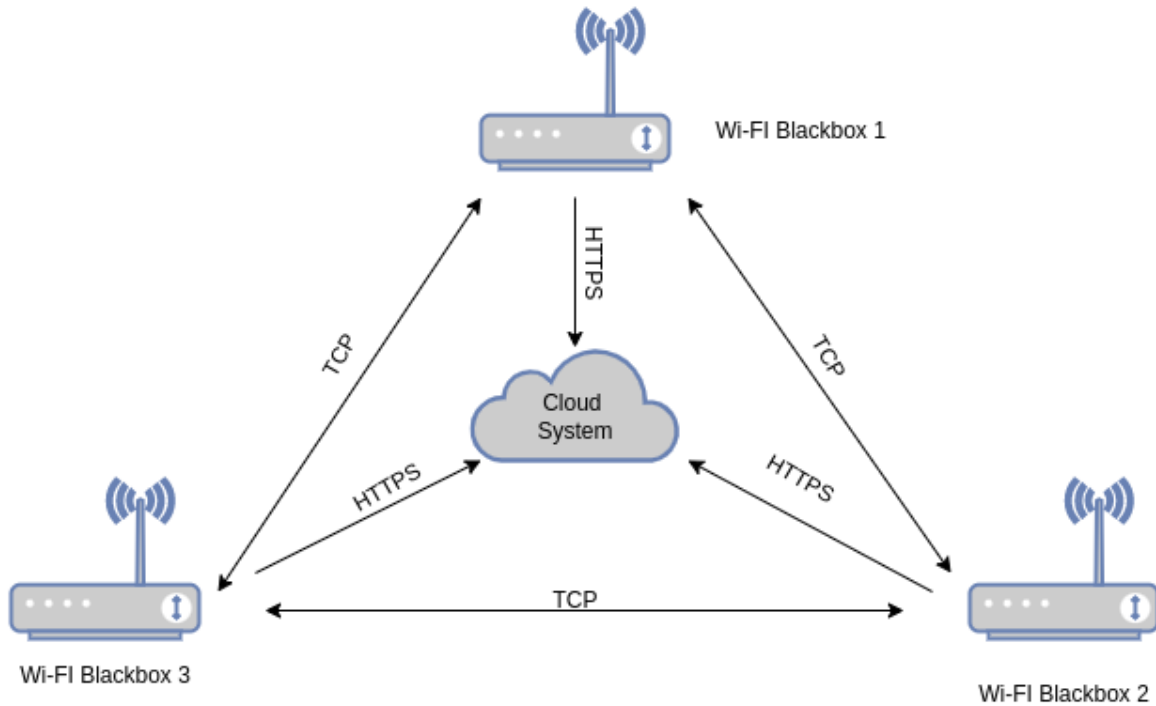


Figure 11. Network Diagram of the Communication Module

Figure 11 depicts the high-level network diagram of the communication module. CM will use Transmission Control Protocol (TCP) to keep the communication channel among every device. It uses Hyper Text Transfer Protocol Secure (HTTPS) to keep communication within the wifi black box and the cloud-based system. It will pass data via JavaScript Object Notation (JSON) format by using the REpresentational State Transfer (REST) Application Protocol Interface (API).

```

1 | {
2 |   "incident_id": "CRI_12344",
3 |   "incident_type": "2",
4 |   "route_data": {
5 |     "generated_ip": "192.168.3.21",
6 |     "destination_ip": "192.168.1.100",
7 |     "protocol": "ssh"
8 |   },
9 |   "incident_level": "1",
10 |  "file_data": {
11 |    "file_name": "",
12 |    "file_size": "",
13 |    "file_format": ""
14 |  },
15 |  "action_plan": "3",
16 |  "timestamp": "1666978753"
17 | }

```

Figure 12. Sample Data Object Communication within devices and cloud system

Figure 12 elaborates on an example data set which will pass within each Wi-Fi Blackbox device and Cloud System. Incident_id is a unique id for a particular incident that a single device might detect. Incident type is an integer value that represents an incident type like 'denial-of-service', 'unauthorized attempts to access a server or system', 'unauthorized file detected', 'unauthorized query detected', 'unknown device in the network' and 'unusual behaviour'. Route data will keep the information about track of Internet Protocol (IP) address and the protocols. The incident level will pass the severity of the incident, which an integer value will represent. These levels were defined as information, advisory, warning and critical. These levels were taken according to the rules created in the device software. File data was sending information about a file if any file was detected. If an incident is detected, the action plan will be an integer value for predefined actions. The black box user can define these action plans, and it will be executing a shell script, executing a command or a simple alert. If the configured action plan matches the communicated action plan, it will automatically execute the particular action plan. The timestamp is the date time for the particular message in milliseconds. The timestamps were used to keep logs within the system regarding the communication module and the related sub-modules.

Backup Maintenance Module (BMM)

Keeping backups of collected data frames is essential, and BMM will be the core module to take and maintain the backups. In the initiation stage of the system, it will be necessary to configure the BMM configurations, which include LOCAL_BACKUP_LOCATION, LOCAL_BACKUP_ZIP_PASSWORD, REMOTE_BACKUP_LOCATION, REMOTE_BACKUP_SFTP_IP, REMOTE_BACKUP_SFTP_PASSWORD, REMOTE_BACKUP_SFTP_USERNAME and REMOTE_BACKUP_ZIP_PASSWORD. If the device is registered with the cloud platform, backups will automatically push to the cloud storage. BMM will also provide a mechanism to track the free and remaining space of the device. When it reaches a threshold, BMM will notify the condition to a particular user via SMS, Email or via cloud system.

System Monitoring Module (SMM)

Maintaining excellent health and continuous monitoring is critical when maintaining any system's stability. SMM will keep track of every major incident related to system operations. It will keep a track log of each service that is running. These services include the Operating system, Input/Output services, Central Processing Unit services, Power Management Services and Temperature management services. At the same time, SMM will maintain a reboot schedule for the system, and this schedule will be defined in the configurations. With this functionality, it can be set to reboot the Wi-Fi Black-Box daily, weekly, monthly or at any given time. SMM will also sync up with the cloud software solution, which will keep track of running time and internal services.

Cloud Based Operational System

The Cloud-based Operational System is a software system that will help administrators or users to perform operational tasks for the particular Wi-Fi BlackBox. This system keeps the high-level details about every Wi-Fi Blackbox and their registration details. The Cloud-based Operational System will

be an optional platform for the users, and if they stop synchronizing data into the Cloud System, they can configure it from the Wi-Fi Blackbox itself. Cloud systems will maintain cloud storage for user accounts, and users can use the particular storage for backup. Cloud systems will provide crucial features mentioned in table 4.

Key Feature	Description
Manage User/Company Profile	This feature enables the central management of all the Wi-Fi Blackbox devices in a company. Authorised users can maintain roles by granting permissions for view, edit and delete operations.
Manage data backup zip files that are stored in cloud storage.	Data frames will be backed up as a password-protected zip file, and authorised users will be allowed to download and check the zip files and meta information of zip files. Users can also capture the hash code of the data frame.
Monitor System information	Authorized users may check the system information such as inside temperature, device voltage, space left in each device and up and its running time. This feature also provides an overview of the device's health.
Subscription Plan	The user/company can change or purchase a new subscription plan. The subscription plan replicates the allowed backup capacity per user/company monthly.

Table 4: Key Features in the Cloud-Based Operational System

The database

It needed a relational database to keep the related data manipulated by each module. MySql was used as per database, and proper indexes were used to handle high amounts of data. Ten tables were used in the initial database design, and 21 BTREE-type indexes were used for data indexing. Figure 13 depicts the table structure of the database design.

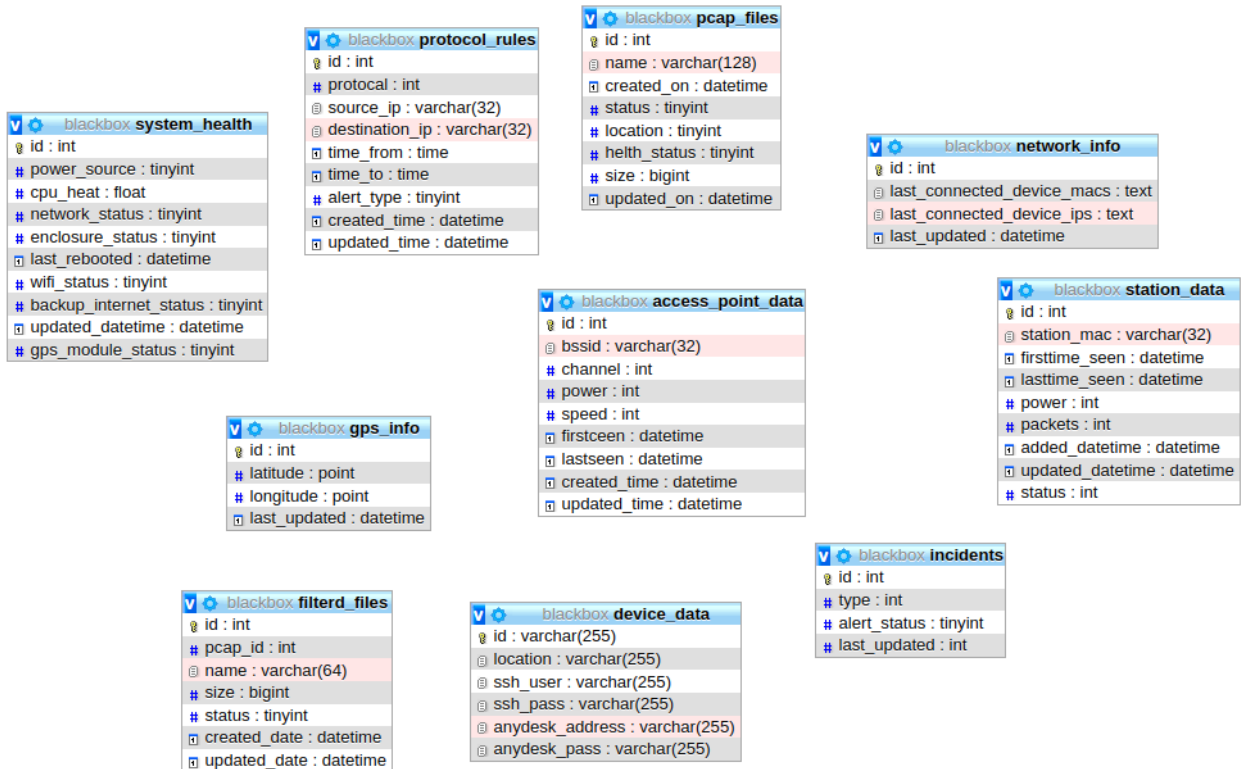


Figure 13 - ERD of the table structure

Chapter 4

IMPLEMENTATION

This chapter discusses the implementation of the hardware and software components and modules. It will also discuss the security control methods and mechanisms which were followed in creating the hardware device. Chapter 4 further discusses the technical blockers and challenges during the implantation process.

Implementation of Hardware Device

The base part of the system is the hardware device, developed with open-source hardware circuits and components. The central part of the device is the Raspberry Pi, which works as the heart of the system. Raspberry Pi mounted to a UPS Hat with the help of 4 bolt screws and nuts. This UPS hat was specially designed for powering up the Raspberry Pi with the support of 218650 Batteries. It supports a 4.2V to 4.5V 18650 lithium battery. It will provide 4A to 8A discharging current with 5V output (“EP-0136 - 52Pi Wiki,” n.d.). The main advantage is that raspberry Pi is easily pluggable with this UPS hat, and a micro USB cable powered the UPS hat. In order to gain steadiness and reliability, this cable was permanently mounted to the device. After the mounting, Raspberry Pi was turned on and tested to determine whether the O/S functionality was working as expected. The normal behaviour of the Raspberry Pi was confirmed even after adding the UPS.

As part of the next step of the hardware implementation, raspberry GPIO was connected to the GPS module, LDR module and the fan. GPIO connections were mounted, as shown in Figure 9 and retested by turning on the Raspberry Pi Module. With the indications of LEDs in each module, it was confirmed that the modules were getting the necessary power, and blinking LEDs demonstrate that some signalling process has been started.

After the basic hardware setup, a plastic enclosure was used as the casing and the protector for hardware components. For this, a 200 X 120 X 55mm Waterproof rugged Plastic Enclosure was used, as shown in figure 14. The enclosure contains a top lid sealed by a rubber strip and four screws which gives better protection from water damage.

All the components were fixed inside the box with the help of hot glue and screws and drilled small opening to get the power cable. At the same time mini fan was fixed into the enclosure, which will reduce the internal temperature when running the hardware devices. The GPS module and LDR module were also fixed, and the external 4G module was also fixed inside the enclosure. The 4G Wi-Fi module used one of the USB ports in the raspberry pi UPS module, and it kept the continuous power supply for the 4G network. Figure 15 shows the actual transverse plane view of the hardware device.

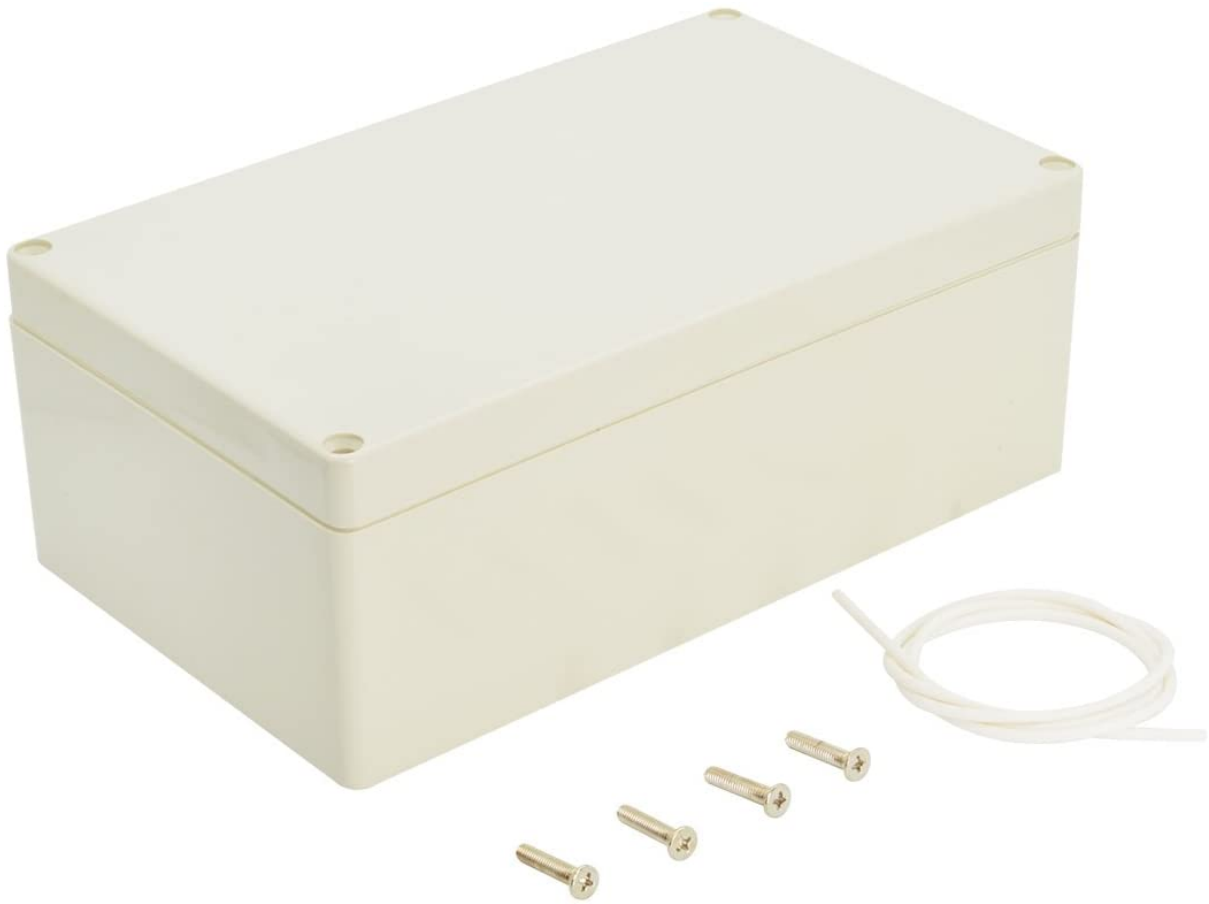


Figure 14 - 200 X 120 X 55mm Waterproof hard Plastic Enclosure

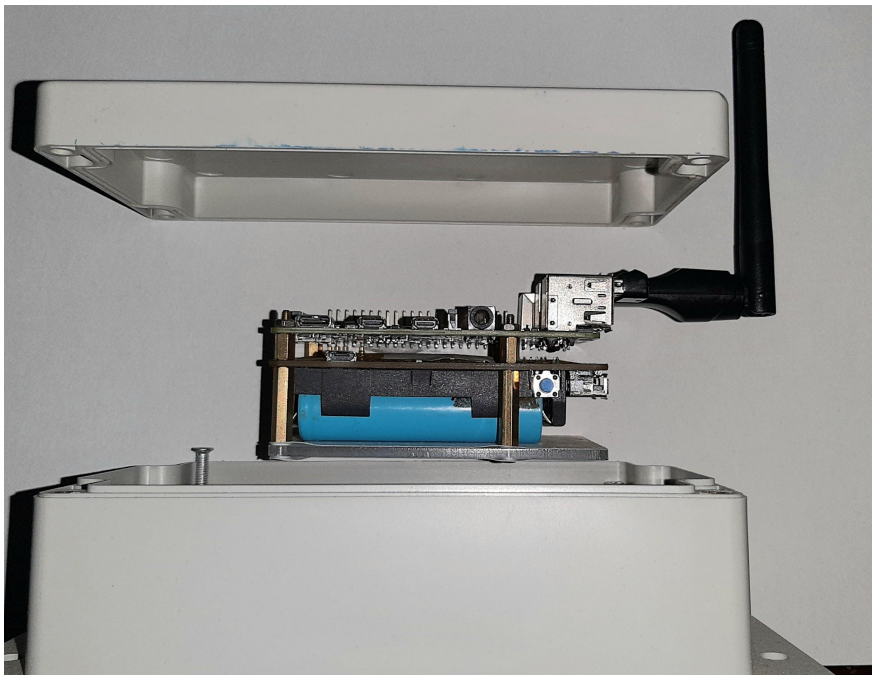


Figure 15 - Transverse plane view of the hardware device

Security Controls

In order to protect hardware components and the infrastructure several security controllers were used. Those security controllers were further categorised as Technical controls, Physical and Technical controls and Physical controls

Technical controls

Miniature UPS is a power backup option for the wifi-black box, and it holds more than 60 minutes in a power interruption. It was fixed at the bottom of the raspberry pi, and 218650 batteries were used to generate backup power.

Wifi-Blackbox is a handy device with a small form factor, and it may be easily grabbed and fetched away by unauthorised personnel. Even if the security was implemented in the promises, there might be a risk of stealing the device. Hence, an internal GPS tracker was implemented to track the device in case of stolen or unauthorised movement, for this NEO-6M GPS Module was used and connected to Raspberry Pi, RX, and TX pins for serial communication. GPS module operates at 2.7V to 3.6V, and the receiver type is 1575.42Mhz, 50 channel. It gives a horizontal position accuracy of 2.5m (“arduino-uart-gps-neo-6m-gps-modulu-k1c-9-datasheet.pdf,” n.d.).

Device uniqueness is a crucial factor when storing the wifi data to maintain the authenticity and integrity of the captured data. To maintain the uniqueness of the device, a physical unclonable function (PUF) was designed and used. This was implemented by considering the Universally unique identifier (UUID) of the storage device, Computer ID, CPU ID and User ID. All these IDs are concatenated and hashed by using the sha256 hashing algorithm. This Hashed value will be the digital fingerprint of the device, and it keeps the uniqueness of the physical microstructure inside the device. Hashed value is unpredictable and uncontrollable, which makes it virtually impossible to duplicate or clone the structure or the inside components. Figure 16 depicts the flow of the PUF generation.

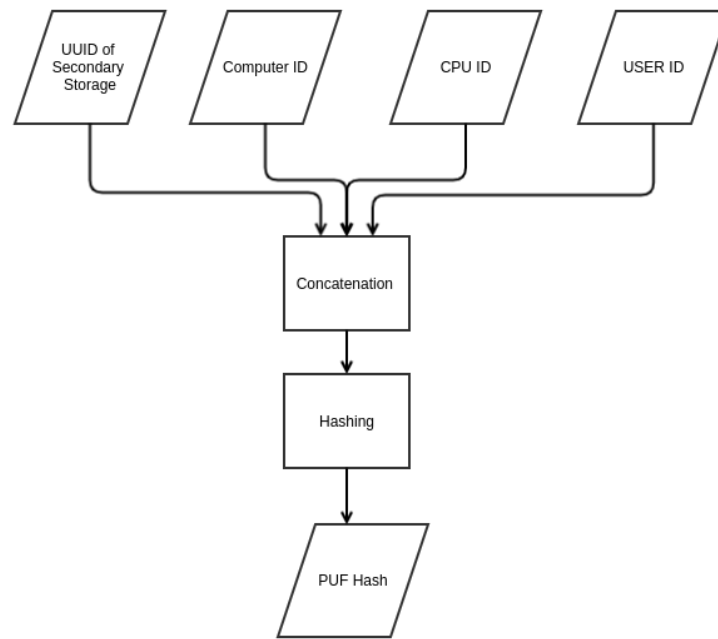


Figure 16 - Flow of the PUF generation.

Internet connectivity is a must for uninterrupted communication, and the device was equipped with a backup internet connection in a situation of external internet failure. This connection was separated from the internal network, using a 4G network to keep the connection. For this requirement, High-Speed Packet Access (HSPA) Universal Serial Bus (USB) modem was used, and it was connected to the miniature UPS module. This modem contains an inbuilt wifi router, which will be used as the secondary network for the device. In any case of network, interruption data was communicated via this secondary wifi network to which the 4G network is connected.

Physical and Technical controls

Open enclosure notifier is a physical and technical control used to detect unauthorized tampering or instructions to the hardware components. If some person tries to hack into the device or if someone opens the top lid of the enclosure, this module will activate and pass a signal to the Raspberry Pi. A read switch and a Light Photosensitive (LDR) sensor module are the key modules to trigger such activity. Those modules will pass a digital plus to the Raspberry Pi GPIO, and internal software will detect it. When this plus is triggered, the system knows that some tampering attempt is ongoing and will instantly alert the particular user.

Physical controls

Hardware components were protected with a rugged plastic enclosure, providing IPX3 protection. IPX3 will give water protection from water sprays up to 60°. This control will protect the inside electronic components from potential water damage.

Implementation of Software Modules

All the software modules were written using PHP and shell scripting. Program initiation starts with the ini.php file, which manages other modules and functions. In the ini.php file, DEAUTH_REQCOUNT and TIME_TO_DUMP should be defined as constant values. DEAUTH_REQCOUNT is the value of the de-authentication request, which will send at the de-authentication attack, and TIME_TO_DUMP is the interval for creating a new PCAP file when the sniffing is going on. Default DEAUTH_REQCOUNT is 20, and TIME_TO_DUMP is 300 seconds.

ini.php file requires config.php file and DbConn.php file, which contains general configurations and database configuration parameters. It also requires a Logger.php file which handles the logging functionality for all the modules. In the Logger.php file, there were some configuration values to define the log_dir as the log directory, log_file_name as the log file name, log_file_path as the log file path and log_file_extension as the log file extension. This incremental log will keep the timestamp, log level, name of the log entry and the actual log message. DbConn.php is a PHP database connection class. It will create the database connection and handles insert and select queries. For this class, the PHP MySQL library was used.

config.php is the core file to keep all the network-related parameters. The config.php file will require defining the Service Set Identifier (SSID) as a string value and the password of the wifi router. Also, it requires defining the WIFI adapter UUID and airdrop file name. In this config file user also have to define the key for the zip file encryption when moving the cap files as backups. In some config cloud Systems, API URLs and authentication tokens were defined. Figure 17 shows how the config parameter setting up in the config.php file

```
1 <?php
2 define ('SSID', 'Dialog 4G 085');
3 define ("PASS", '9Ec58De3');
4 define ("WIFI_ADAPTER", "wlx76011020bc2e");
5 define ("AIRDROP_FILE", 'airdrop.csv');
6
7 define ("FILE_STATUS" , [
8     ['statusId'=>0, 'description'=>'Encrypted Raw File'],
9     ['statusId'=>1, 'description'=>'Decrypted Raw File'],
10 ] );
```

Figure 17 - How the config params should be defined

The initiation PHP script will then execute the start method, check for the network connection details, and ensure the specific adapter is in monitor mode. After confirming the network status then, the script will execute six methods which were named remove CSV which is responsible for deleting existing CSV files, kill process method which was responsible for killing ongoing similar processes, restart network method which will be restarting the network, startMonitorMood method which starting the monitor mode and then airoDump and getBssidData methods will call to capture the data frames in the wifi network.

Packet capturing was done by using the Airomon module, and from the PHP script, Airodump and Airomon commands were executed. as below.

1. `exec('airmon-ng check kill', $output);`
2. `exec('airmon-ng start '.WIFI_ADAPTER, $output);`
3. `exec('service network-manager restart', $output);`
4. `exec('airmon-ng stop wlan0mon', $output);`
5. `exec('airmon-ng', $output);`
6. `exec('airodump-ng -w '.AIRDROP_FILE.' --output-format csv '.$interface.' > /dev/null 2>&1 & echo $!', $output);`
7. `exec('service network-manager status', $output);`
8. `exec("rm *.csv");`
9. `exec("rm *.netxml");`
10. `exec('aireplay-ng -0 '.DEAUTH_REQCOUNT.' -a '.$_stationdata['ap_bssid'].' -c '.$_stationdata['station_mac'].' '.$monitor_name.' > /dev/null 2>&1 & echo $!');`
11. `exec('airodump-ng -w dump'.$timee.' -c '.$bssidData[1].' --bssid '.$bssidData[0].' '.$monitor_name.' 2>&1 & echo $!');`

There were 11 principal executions that happened in order to complete 1 data collection cycle. This execution was executed as the root user. Hence permission issues will not arise within the program. To decrypt the data, it's crucial to capture the 4-way handshake of the wifi network for decrypting the data frame. In the 4-way handshake depicted in figure 18, the first Access Point will send a message, which will be grabbed by the client and sent back with Snounce and mic. This is the second message. Thirdly, the access point will send back a message with Group Temporal Key (GTK) and Master Session Key (MSK); this is the third message. Finally, the client will acknowledge by saying that the keys have been installed. This will be the fourth message. As it uses 4steps, it is defined as a 4way handshake.

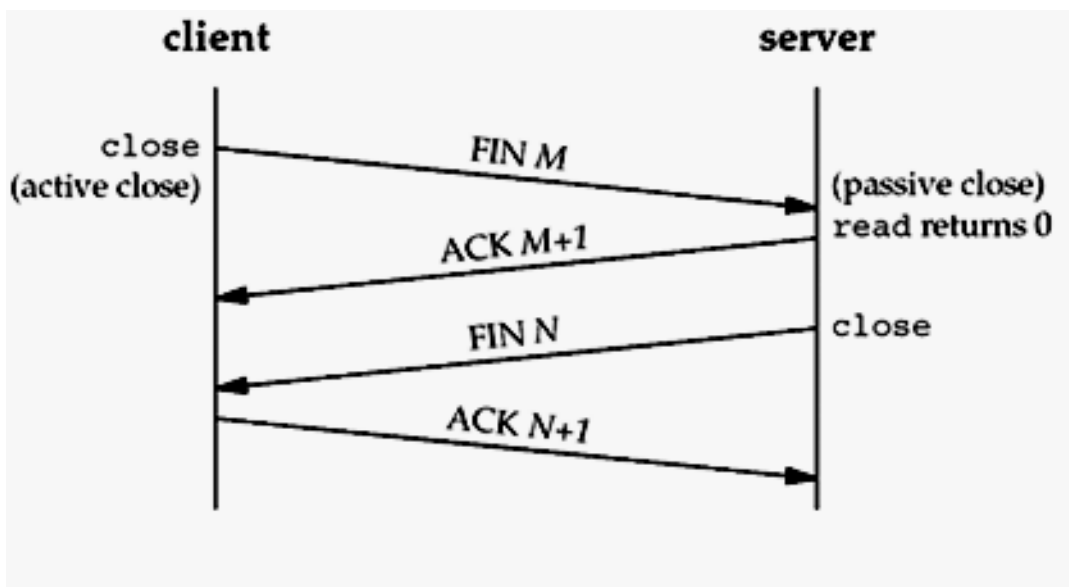


Figure 18 - 4-way handshake

This 4-way handshake is normally done when a client and an access point connect. Hence to capture the 4-way handshake, the death script will initiate an authentication request for every device in the network. This will be triggered timely; with this initiation, all the connected devices will disconnect and reconnect to the network. This will be executed in a defined time gap when the data collection is

done. This death attract is executed by using “sudo aireplay-ng --deauth 0 -a '\$bssid.' -c '\$con_device_id.' '\$interface.' > /dev/null 2>&1 & echo \$!” command and relevant parameter will be inserted when the program is running.

Data frames will be stored in another directory as a packet capture (CAP) file. This file location was known as an encrypted file store. In order to query the internal details of the CAP files, it is required to decrypt those files and examine them. For this task, another cron process will run every minute to scan for newly encrypted files. This cron process mainly runs with a PHP script, including the description part for the CAP files. The description script was written by using shell scripting with the help of Tshark commands. As shown in Figure 19, Tshark commands will execute to decrypt the data frames. At the same time, decrypted frames will store in a separate location with PCAP extension, as can be examined by the wire-shark software.

```
1 #!/bin/sh
2
3 infile=$1
4 outfile=$2
5 ssid='Dialog 4G 085'
6 psk='9Ec58De3'
7 echo 'infile: $infile'
8 echo 'outfile: $outfile'
9 echo 'psk: $psk'
10 echo 'ssid: $ssid'
11
12 tshark -r $infile -T fields -E separator=, -E quote=d, -E occurrence=a -E header=y -e frame.number -e frame.time_epoch -e _ws.col.Source -e _ws.col.Destination -e
   _ws.col.Protocol -e _ws.col.Info -w $outfile \
13   -o wlan.enable_decryption:TRUE \
14   -o 'wlan.80211_keys:\wpa-pwd',\"$(psk):$(ssid)\" > $outfile.csv
```

Figure 19 - Decrypter code with Tshark commands

The analyser is another PHP service which will analyse the decoded PCAP files in order to detect anomalies. This service will connect with Virustotal API to check and detect malware and viruses inside the PCAP files.

System monitoring service is another service which was developed by using python. It runs as a service in order to read GPIO signals and raspberry pi data such as temperature, voltage and storage capacity. This python service will listen to the digital signals of the GPIO and runs a trigger service according to the signal it's getting.

Graphical User Interface (GUI)

The system was equipped with a GUI which makes users operate and manage incidents and PCAP files. This UI is accessible via the web service which runs inside the device. GUI and all the operations are accessible via username and password and this username and password should be defined in the initial setup of the device. From this GUI many useful functions were given and Table 5 listed out the functional features of the GUI which represent by the Figure 20 and Figure 21. Figure 22 shows the GUI for the system health panel and Figure 23 represents the GUI for the interface of the file saving interface.

Feature Name	Description
Login	Maintain the login session of the user. Allows authorised personnel to do the required changes.
Incidents	Allow authorised users to view incidents.
Rules	<p>Rule is a trigger point to catch any kind of incident. Users can define a rule by defining as shown in Figure 18</p> <ul style="list-style-type: none"> ● Protocol ● Source IP ● Destination IP ● Time From ● Time To ● Alert Type <p>When the system has detected a match with a specifically created rule it will behave and act according to the alert type.</p>
System Health	<p>System health is an overview of the internal health status. As per the figure 19 It shows</p> <ul style="list-style-type: none"> ● CPU Temperature ● Power Source ● Network Status ● Enclosure Status ● Wifi-Status ● GPS Module Status ● Last Rebooted
Access Point data	<p>Access point data shows all the data related to the access point which includes BSSID</p> <ul style="list-style-type: none"> ● First time seen ● Last time seen ● channel ● Privacy ● Cipher ● Authentication ● Power ● beacons ● ESSID ● LAN IP
Station Data	<p>It shows the data related to connecting clients. It includes</p> <ul style="list-style-type: none"> ● Station MAC ● First time seen ● Last time seen ● Power

	<ul style="list-style-type: none"> • Packets • Added date-time • Updated date time • Status
GPS Info	<p>GPS info shows the GPS information about the device and it includes</p> <ul style="list-style-type: none"> • latitude • longitude • updated time
Files	<p>From the File menu, users can check and download the PCAP files. It also contains a filter to retrieve files for a specific time. Figure 20 shows the GUI inside the files menu including</p> <ul style="list-style-type: none"> • File name • File hash

Table 5 - Functional features of the GUI

GUI and the web back end was developed by using PHP and with Yii 2.0 framework and the database was MySQL. By using Create, Read, Update, Delete(CRUD) operation manager all the models, views and controllers were developed.

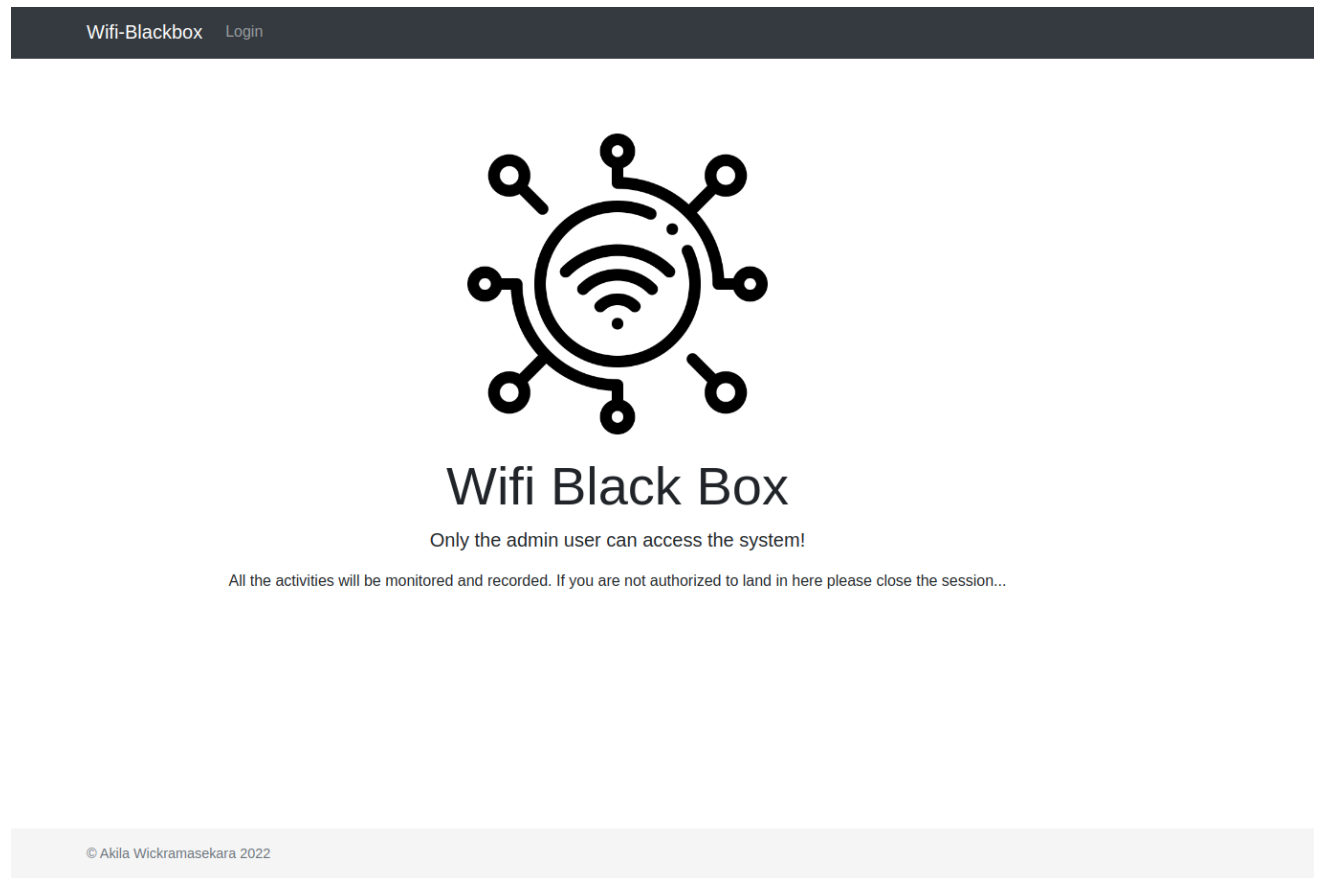


Figure 20 - Landing Page of the Wifi- Blackbox GUI

Home / Protocol Rules

Protocol Rules

Create Protocol Rules

Showing 1-3 of 3 items.

#	ID	Protocol	Source Ip	Destination Ip	Time From	Time To	Alert Type	
	<input type="text"/>	Select Pr <input type="text"/>	<input type="text"/>	<input type="text"/>				
1	1	TCP	192.168.8.123		17:00:00	23:30:00	Advisory	
2	2	TCP	192.168.8.123	192.168.8.123	02:18:50	05:00:00	Warning	
3	3	TCP			01:20:00	08:00:00	Information Statment	

Figure 21 - GUI to manage protocol rules

System Health

Showing 1-20 of 179 items.

#	Power Source	CPU Heat	Network Status	Enclosure Status	WiFi Status	GPS Module Status	Last Rebooted
	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
1	1	12.0	1	1	1	Up	2022-08-11 23:23:10
2	1	63.3	1	1	1	Down	2022-08-11 23:23:10

Figure 22 - GUI for the system health

Pcap Files

Showing 341-360 of 373 items.

#	File Name	File Hash	Created On	Status	Location	Health Status	Size
		<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
341	dump_20220814_072311.cap	(not set)	2022-08-14 07:23:11	2	1	1	1519472
342	dump_20220814_072833.cap	(not set)	2022-08-14 07:28:33	2	1	1	13200816
343	dump_20220814_073356.cap	(not set)	2022-08-14 07:33:56	2	1	1	11416076
344	dump_20220814_073918.cap	(not set)	2022-08-14 07:39:18	2	1	1	32017568
345	dump_20220814_074441.cap	(not set)	2022-08-14 07:44:41	2	1	1	1983616
346	dump_20220814_075003.cap	(not set)	2022-08-14 07:50:03	2	1	1	1294772
347	dump_20220814_075525.cap	(not set)	2022-08-14 07:55:25	2	1	1	941980
348	dump_20220814_080048.cap	(not set)	2022-08-14 08:00:48	2	1	1	858848

Figure 23 - GUI for the interface of the file

Final Product

After the physical parts and the software parts were implemented the final product was branded as the wifi-blackbox. The top lid was engraved by a Laser cutting machine and Figure 24 shows the final output of the device.

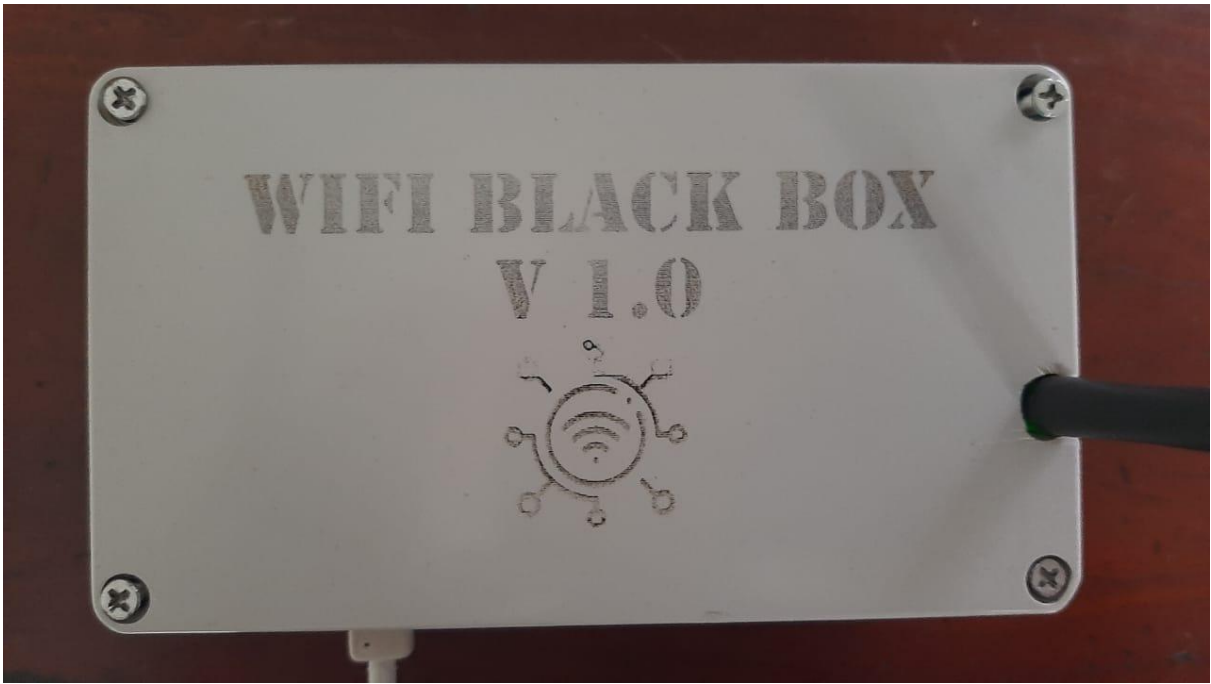


Figure 24 - Final outlook of the product

The description was tested more than 100 times as the de-authentication request was not initiated properly. Hence it was difficult to decrypt the collected data frames from the decrypter. After some test round de-authentication request was calibrated and continues to work as expected. Figure 26 shows the decrypted data frame after the testing has been conducted.

No.	Time	Source	Destination	Protocol	Length	Info
493	6.398358	Shenzhen 51:70:55 (d8:08:66:51:70:55)	(RA)	802.11	10	Acknowledgement, Flags=.....
494	6.398364	203.81.96.82	192.168.8.141	QUIC	126	Protected Payload (K9P)
495	6.398691	203.81.96.82	192.168.8.141	QUIC	126	Protected Payload (K9P)
496	6.398612	Shenzhen 51:70:55 (d8:08:66:51:70:55)	(RA)	802.11	10	Acknowledgement, Flags=.....
497	6.398616	Shenzhen 51:70:55 (d8:08:66:51:70:55)	Broadcast (ff:ff:ff:ff:ff:ff)	802.11	16	CF-End (Control-frame), Flags=.....
498	6.402835	157.240.7.20	192.168.8.141	QUIC	1310	Initial, SCID=5def4d3073ea1b98, PKN: 13112991, ACK, PADDING
499	6.402847	Shenzhen 51:70:55 (d8:08:66:51:70:55)	(RA)	802.11	10	Acknowledgement, Flags=.....
410	6.402851	Shenzhen 51:70:55 (d8:08:66:51:70:55)	Broadcast (ff:ff:ff:ff:ff:ff)	802.11	16	CF-End (Control-frame), Flags=.....
411	6.402857	192.168.8.141	203.81.96.82	QUIC	126	Protected Payload (K9P), DCID=490948314fb8c6c
412	6.402860	Shenzhen 51:70:55 (d8:08:66:51:70:55)	HuaweiTe_a2:06:72 (38:fb:14:a2:06:72)	802.11	28	802.11 Block Ack, Flags=.....
413	6.404020	157.240.7.20	192.168.8.141	QUIC	1310	Initial, SCID=5def4d3073ea1b98, PKN: 13112992, CRYPTO, PADDING
414	6.404039	157.240.7.20	192.168.8.141	QUIC	1310	Initial, SCID=5def4d3073ea1b98, PKN: 13112992, CRYPTO, PADDING
415	6.404644	Shenzhen 51:70:55 (d8:08:66:51:70:55)	(RA)	802.11	10	Acknowledgement, Flags=.....
416	6.404653	Shenzhen 51:70:55 (d8:08:66:51:70:55)	Broadcast (ff:ff:ff:ff:ff:ff)	802.11	16	CF-End (Control-frame), Flags=.....
417	6.408618	157.240.7.20	192.168.8.141	QUIC	1310	Handshake, SCID=5def4d3073ea1b98
418	6.407235	157.240.7.20	192.168.8.141	QUIC	1310	Handshake, SCID=5def4d3073ea1b98
419	6.407672	157.240.7.20	192.168.8.141	QUIC	1310	Handshake, SCID=5def4d3073ea1b98
420	6.407981	192.168.8.141	157.240.7.20	QUIC	1310	Initial, DCID=5def4d3073ea1b98, PKN: 12680452, ACK, PADDING
421	6.408320	192.168.8.141	157.240.7.20	QUIC	1310	Initial, DCID=5def4d3073ea1b98, PKN: 12680452, ACK, PADDING
422	6.408343	Shenzhen 51:70:55 (d8:08:66:51:70:55)	HuaweiTe_a2:06:72 (38:fb:14:a2:06:72)	802.11	28	802.11 Block Ack, Flags=.....
423	6.408652	157.240.7.20	192.168.8.141	QUIC	1310	Handshake, SCID=5def4d3073ea1b98
424	6.409122	157.240.7.20	192.168.8.141	QUIC	1310	Handshake, SCID=5def4d3073ea1b98
425	6.409995	157.240.7.20	192.168.8.141	QUIC	1310	Handshake, SCID=5def4d3073ea1b98
426	6.410899	157.240.7.20	192.168.8.141	QUIC	1310	Handshake, SCID=5def4d3073ea1b98
427	6.410815	Shenzhen 51:70:55 (d8:08:66:51:70:55)	(RA)	802.11	10	Acknowledgement, Flags=.....
428	6.411152	157.240.7.20	192.168.8.141	QUIC	1215	Handshake, SCID=5def4d3073ea1b98
429	6.411158	157.240.7.20	192.168.8.141	QUIC	158	Protected Payload (K9P)
430	6.411160	HuaweiTe_a2:06:72 (38:fb:14:a2:06:72)	Shenzhen 51:70:55 (d8:08:66:51:70:55)	802.11	28	802.11 Block Ack, Flags=.....
431	6.411169	Shenzhen 51:70:55 (d8:08:66:51:70:55)	Broadcast (ff:ff:ff:ff:ff:ff)	802.11	16	CF-End (Control-frame), Flags=.....
432	6.463983	2409:8c50:a00:2086::52	2402:4000:2380:c814:8c54:3759:61f3:d7b9	TCP	122	80 - 42784 [ACK] Seq=1 Ack=279 Win=2996 Len=0 TSval=1494165847 TSecr=1096024
433	6.463990	2409:8c50:a00:2086::52	2402:4000:2380:c814:8c54:3759:61f3:d7b9	TCP	122	80 - 42784 [ACK] Seq=279 Ack=197 Win=97552 Len=0 TSval=1696066 TSecr=1494165848
434	6.463995	2409:8c50:a00:2086::52	2402:4000:2380:c814:8c54:3759:61f3:d7b9	HTTP	318	HTTP/1.1 204 No Content
435	6.464086	Shenzhen 51:70:55 (d8:08:66:51:70:55)	(RA)	802.11	10	Acknowledgement, Flags=.....
436	6.464089	Shenzhen 51:70:55 (d8:08:66:51:70:55)	Broadcast (ff:ff:ff:ff:ff:ff)	802.11	16	CF-End (Control-frame), Flags=.....
437	6.464477	2402:4000:2380:c814:8c54:3759:61f3:d7b9	2409:8c50:a00:2086::52	TCP	122	42784 - 80 [ACK] Seq=279 Ack=197 Win=97552 Len=0 TSval=1696066 TSecr=1494165848
438	6.464524	Shenzhen 51:70:55 (d8:08:66:51:70:55)	HuaweiTe_a2:06:72 (38:fb:14:a2:06:72)	802.11	28	802.11 Block Ack, Flags=.....
439	6.550607	192.168.8.141	157.240.7.20	QUIC	165	Handshake, DCID=5def4d3073ea1b98
440	6.550617	Shenzhen 51:70:55 (d8:08:66:51:70:55)	HuaweiTe_a2:06:72 (38:fb:14:a2:06:72)	802.11	28	802.11 Block Ack, Flags=.....

Frame 1: 221 bytes on wire (1768 bits), 221 bytes captured (1768 bits) on interface 0
 IEEE 802.11 Beacon frame, Flags:
 IEEE 802.11 Wireless Management

```

0000 80 00 00 00 ff ff ff ff ff ff d8 08 66 51 70 55 .....fQpU
0010 d8 08 66 51 70 55 70 cf 90 01 46 13 00 00 00 00 .....fQpUp aF.....
0020 64 00 11 04 00 00 44 69 61 c6 0f 20 34 47 20 d.....D1alog 46
0030 30 30 30 00 02 04 00 06 06 c6 12 18 24 00 01 00 0005.....$
0040 05 04 00 01 00 00 07 06 55 53 20 01 0b 1e 2a 01 .....US * * *
0050 04 32 04 30 48 60 6c 2d 1a 2c 10 1f ff ff 00 00 2 0H L .....
0060 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0070 00 00 00 20 10 00 00 00 00 00 00 00 00 00 00 .....
0080 00 00 00 00 00 00 00 00 00 00 00 30 14 01 00 00 .....0
0090 0f ac 04 01 00 00 0f ac 04 01 00 00 0f ac 02 00 .....
00a0 00 0d 18 00 50 f2 02 01 01 00 00 83 a4 00 00 27 .....P.....L
00b0 a4 00 00 42 43 50 00 62 32 f7 00 0d 00 00 e9 4c .....P.....L
00c0 02 01 60 0d 18 00 50 f2 04 10 4a 00 01 10 10 44 .....P.....L
00d0 00 01 02 10 40 00 00 00 37 2a 00 01 20 .....I.....7*
  
```

Figure 26 - Decrypted data frame opened in Wire Shark after the testing has been conducted.

Testing Phase 2

Phase 2 of testing was testing the behaviour of protocols with the system. This was conducted within the same testing environment. TCP, HTTP, UDP and SSH protocols were tested and verified that the data capturing and the de-authentication worked without an issue.

Testing Phase 3

Phase 3 testing was initiated to identify the behaviours of the network attacks. For this testing same environment with 2 devices was used. For the testing ICMP attack or the denial of service, the attack was conducted from one device to the other device and checked the data capturing behaviour and the decryption. The decrypted data clearly shows the behaviours of a final of service attacks with ICMP data frames. Figure 27 elaborates on the captured data frame in wire shark software.

No.	Time	Source	Destination	Protocol	Length	Info
2285	47.286984	192.168.8.123	192.168.8.1	ICMP	134	Echo (ping) request id=0x0026, seq=2/512, ttl=64 (reply in 2287)
2287	47.287928	192.168.8.1	192.168.8.123	ICMP	134	Echo (ping) reply id=0x0026, seq=2/512, ttl=64 (request in 2285)
2681	48.251952	192.168.8.123	192.168.8.1	ICMP	134	Echo (ping) request id=0x0026, seq=3/768, ttl=64 (reply in 2683)
2683	48.252984	192.168.8.1	192.168.8.123	ICMP	134	Echo (ping) reply id=0x0026, seq=3/768, ttl=64 (request in 2681)
2856	49.289968	192.168.8.123	192.168.8.1	ICMP	134	Echo (ping) request id=0x0026, seq=4/1024, ttl=64 (reply in 2858)
2858	49.211908	192.168.8.1	192.168.8.123	ICMP	134	Echo (ping) reply id=0x0026, seq=4/1024, ttl=64 (request in 2856)
2945	50.211906	192.168.8.123	192.168.8.1	ICMP	134	Echo (ping) request id=0x0026, seq=5/1280, ttl=64 (reply in 2947)
2947	50.212024	192.168.8.1	192.168.8.123	ICMP	134	Echo (ping) reply id=0x0026, seq=5/1280, ttl=64 (request in 2945)
3891	51.212530	192.168.8.123	192.168.8.1	ICMP	134	Echo (ping) request id=0x0026, seq=6/1536, ttl=64 (reply in 3893)
3893	51.213048	192.168.8.1	192.168.8.123	ICMP	134	Echo (ping) reply id=0x0026, seq=6/1536, ttl=64 (request in 3891)
3239	52.216114	192.168.8.123	192.168.8.1	ICMP	134	Echo (ping) request id=0x0026, seq=7/1792, ttl=64 (reply in 3241)
3241	52.216632	192.168.8.1	192.168.8.123	ICMP	134	Echo (ping) reply id=0x0026, seq=7/1792, ttl=64 (request in 3239)
3481	53.215990	192.168.8.123	192.168.8.1	ICMP	134	Echo (ping) request id=0x0026, seq=8/2048, ttl=64 (reply in 3486)
3486	53.216632	192.168.8.1	192.168.8.123	ICMP	134	Echo (ping) reply id=0x0026, seq=8/2048, ttl=64 (request in 3481)
3783	54.216114	192.168.8.123	192.168.8.1	ICMP	134	Echo (ping) request id=0x0026, seq=9/2304, ttl=64 (reply in 3785)
3785	54.217144	192.168.8.1	192.168.8.123	ICMP	134	Echo (ping) reply id=0x0026, seq=9/2304, ttl=64 (request in 3783)
3821	55.217648	192.168.8.123	192.168.8.1	ICMP	134	Echo (ping) request id=0x0026, seq=10/2560, ttl=64 (reply in 3823)
3823	55.218882	192.168.8.1	192.168.8.123	ICMP	134	Echo (ping) reply id=0x0026, seq=10/2560, ttl=64 (request in 3821)
3896	56.218162	192.168.8.123	192.168.8.1	ICMP	134	Echo (ping) request id=0x0026, seq=11/2816, ttl=64 (reply in 3898)
3898	56.219194	192.168.8.1	192.168.8.123	ICMP	134	Echo (ping) reply id=0x0026, seq=11/2816, ttl=64 (request in 3896)
3948	57.222774	192.168.8.123	192.168.8.1	ICMP	134	Echo (ping) request id=0x0026, seq=12/3072, ttl=64 (reply in 3950)
3950	57.223802	192.168.8.1	192.168.8.123	ICMP	134	Echo (ping) reply id=0x0026, seq=12/3072, ttl=64 (request in 3948)
3958	58.226868	192.168.8.123	192.168.8.1	ICMP	134	Echo (ping) request id=0x0026, seq=13/3328, ttl=64 (reply in 3960)
3960	58.227898	192.168.8.1	192.168.8.123	ICMP	134	Echo (ping) reply id=0x0026, seq=13/3328, ttl=64 (request in 3958)

Figure 27 - Captured data frames of DOS attack in wire shark software

Testing Phase 4

In phase 4, it was evaluated the behaviour of transferring a known file with an infected trojan horse. Testing was done using the same environment and as Figure 28 shows captured data clearly showed that the file was infected. At the same time, the anomaly detection module was triggered and it shows the infection file report which was generated by the virus total API.

Figure 28 - Captured PCAP file with the infected file opened in Wireshark

With Virus total API it gives a clear report for the infected file which includes the basic property such as the MD5 hash of the file, the SHA-1 hash of the file, the SHA-256 hash of the file, File Type, TrID and File Size. Figure 29 demonstrates how the virus total API report was generated.

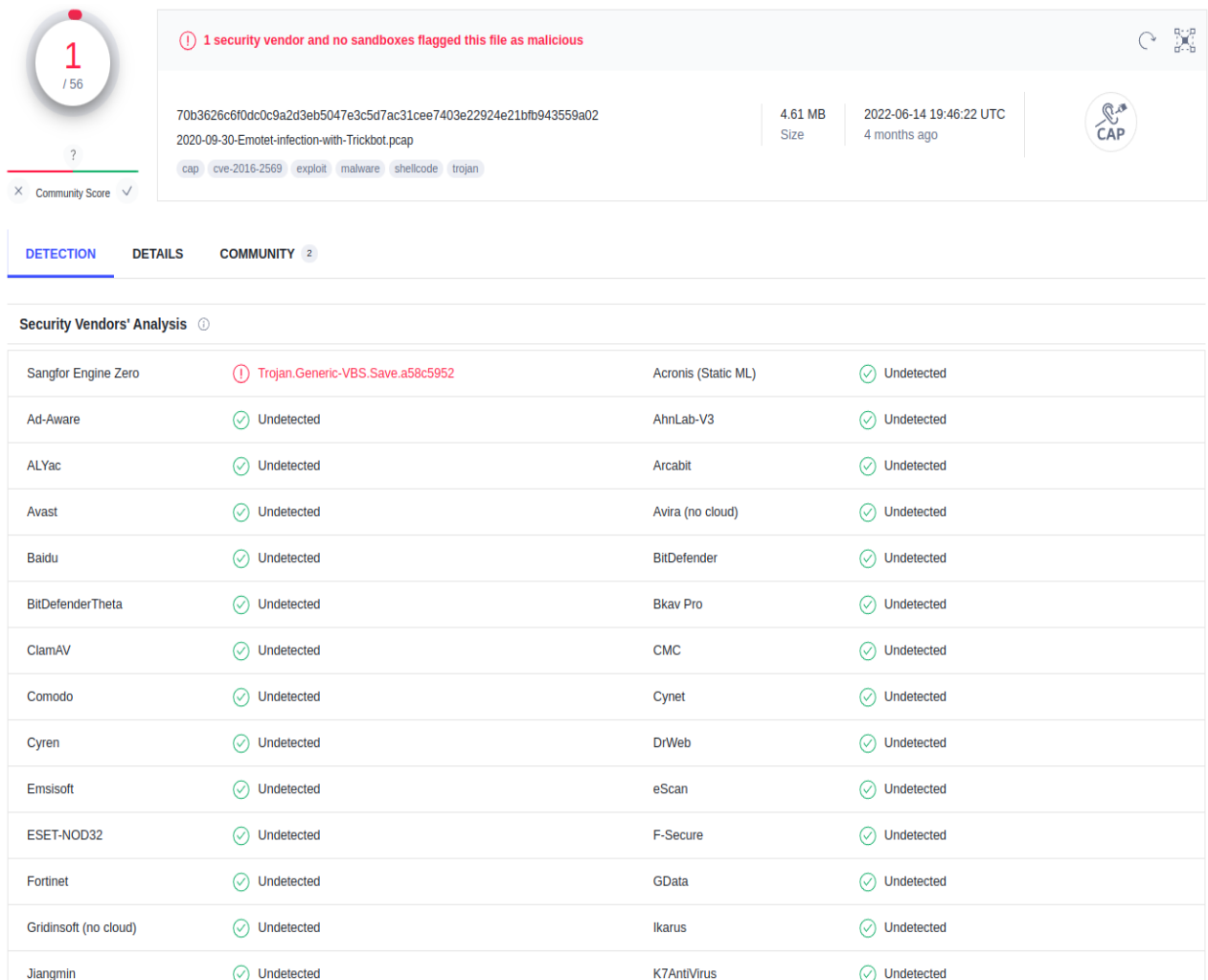


Figure 29 - Virus total API report

Testing Phase 5

With testing phase 5, it was tested the behaviour of the incident triggering part. For this same testing environment was used. For this testing, an incident was created from the GUI panel. It was chosen TCP as the protocol for the first scenario and then tried with the FTP protocol. As a result, the device successfully detected the created rule and created an incident according to the action of the rule. It was re-test to trigger an SMS as the incident reporting alert and within a 1minute of time incident was triggered, and reported and the SMS was sent to the mobile of the particular user. Figure 30 shows how the SMS trigger was done.

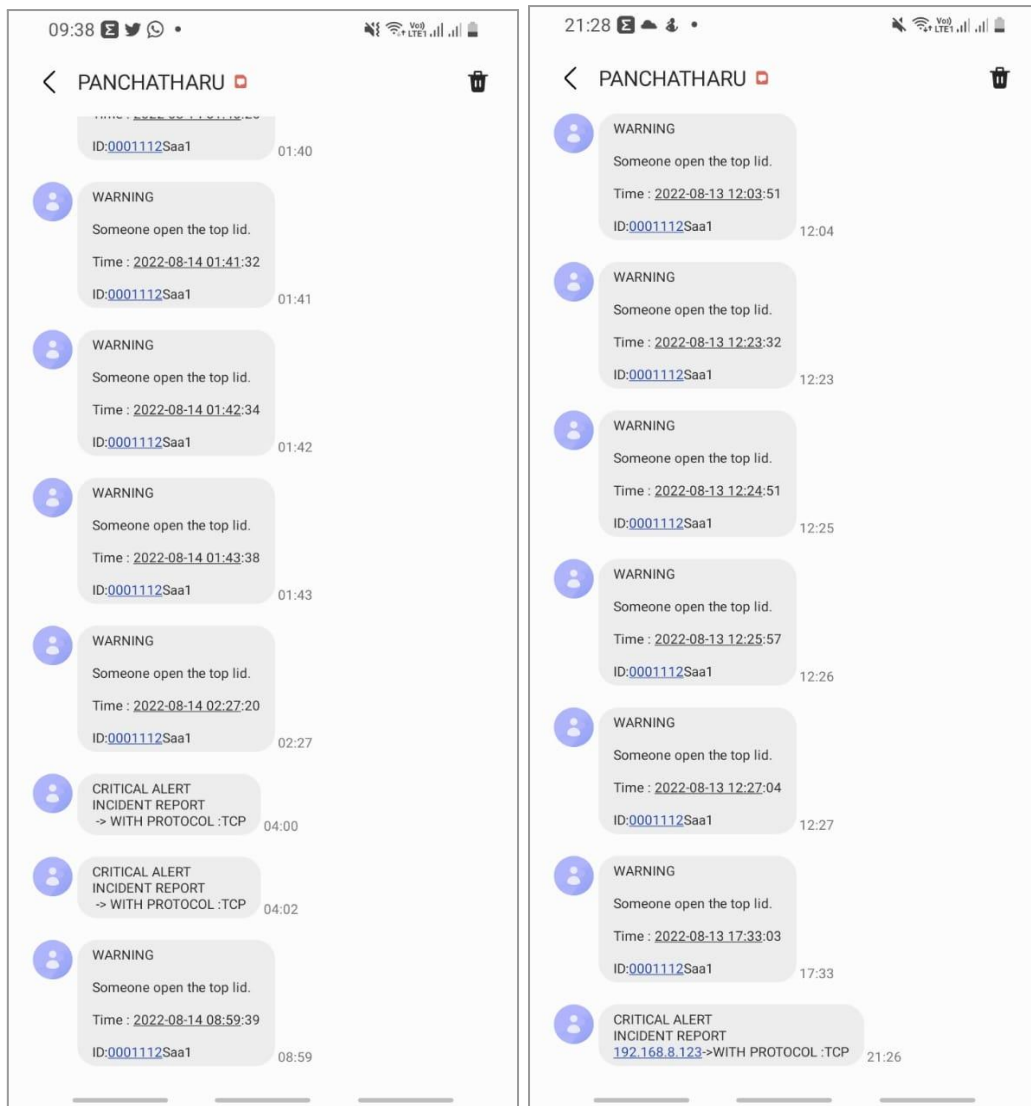


Figure 30 - SMS trigger alert messages

Testing hardware system

Hardware parts were tested again after the assembly of components and with the final testing phase it was tested the durability and the running time of the hardware parts was. For this, the testing environment was upgraded to 4 devices and kept continuous network traffic within the devices and from outside of the network. In this testing, the device temperature was monitored. Also, the power backup functionality was thoroughly tested with the heavy traffic load on the device. To test the physical and technical controls combined with the hardware parts, it was tried open the top lid of the device.

As the testing results when the lid is opened the intruder/ tampering detection module was activated and SMS was fired continuously to keep the user alert as it was an urgent scenario. Figure 27 Shows how the warning message was fired to the user's mobile phone when the top lid was opened. At the same time, CPU temperature was tested with a high amount of network traffic monitoring and with

the limitation of air intake to the device CPU gets overheated up to 85 degrees of celsius which was not a recommended value for the Raspberry Pi. hence the system automatically shuts down its operation to protect the CPU and the other hardware components from getting damaged. With these results, it determined that even if the fan is working with a high amount of network traffic it is required a better cooling solution for the device.

When testing the power backup functionality it showed that the high amount of network traffic will drain the battery so fast. As the processing, if high the power consumption of the device also gets high and in fact, it was monitored that the maximum power backup time is about 20 minutes with a considerable amount of network traffic.

Chapter 7

CONCLUSION AND FUTURE WORK

The outcome and the discussion of this project are summarized in the conclusion chapter. The first part of the chapter described the work and outcome gained via this research and the second part explains the improvement and future plan for improving this research.

Conclusion

The objective of this research was to create a forensic-ready device for internal wi-fi networks that acts as a passive control mechanism, that will monitor, store and communicate the activities and later detect anomalies and patterns from the gathered data and information.

The outcome of this research proved that internal wifi networks can be protected from insider attacks and related activities can be tracked and saved for forensic purposes. Also, the huge range of usability of this device proves that this can be easily configured and used by anyone who needs to protect their internal wifi network and for persons who need to audit their internal wifi network. Hence this device will be a practical solution for Business Owners/ CEOs, Security Auditors, Digital Forensic Analysis Persons, Researchers and Digital Protection Solution Vendors.

With this product and research, it Demonstrated how to design and implement a tamper-proof network monitoring platform. It also celebrates the technical and physical controls for simmer devices. As this device is developed by using open-source hardware and software, this research elaborates on the expandability and use of free and open-source technologies to create a better scalable and secure architecture.

Wi-Fi Blackbox contains PUF or physically unclonable function and this demonstrates how to protect the uniqueness of hardware or electronic components for any similar products.

This product will also work as a deterrent controller for the inside workers and will showcase how to minimise insider threats by changing the mindset of inside workers. This will demotivate insider attackers and gives a better defence network infrastructure for a company.

The device will continuously monitor and track network incidents and anomalies and this will be a great data-collecting platform. With the detection of highly valuable information regarding new attack behaviour or new vires, the Wi-Fi Blackbox system will Create a global peer-to-peer incident information network to keep a sustainable security mechanism. With this, there will be a huge impact on secure data sharing as all the networks are monitored actively. This will also help to Identify upcoming network threats inside an internal network or an external network.

Business intelligence data are the key facts for company management to make business-related decisions and the management every time relies on actual data for future predictions. As the Wi-Fi Blackbox collect network-related data and when this data is merged with business-related information

and conducting data mining exercise relevant to business strategies, there will be a huge impact on the outcome of the results. Managers and decision-makers can co-relate data with business data and gain important patterns to grow their businesses. So that this device will be also a data collector for a business information system.

Future Work

There will be many improvements for this device. To eliminate the main drawback of an overheating problem, it is designed to create a new outer case from aluminium and also it will include a better ventilation system. As the current power backup time is lesser its plans to increase the capacity of ampere-hours of the batteries. The storage of the device will be increased to store much more data frames and for the version 2 device, it is planned to improve the watchdog functionality that will track the system health in real-time.

The communication module is planned to improve further with a better and faster protocol for the internal communication of the devices. It is planned to create a machine learning algorithm to detect abnormal behaviours in data packets and incidents and this will be implemented as a value-added service for the system.

The cloud-based control system will be added as a subscription base service for the vendors and it will value-added services and backup storage will be costed for this subscription service. All the planned improvements will be released in the Wi-Fi Blackbox version 2 with a new form factor and a better-scale solution.

Bibliography

- 2020 Cost of Insider Threats Global report, 2020. . Res. Rep. 31.
- 2022 Cost of Insider Threats Global Report, 2021. 45.
- Bhagyavati, Summers, W., DeJoie, A., 2004. Wireless security techniques: an overview 82–87. <https://doi.org/10.1145/1059524.1059541>
- Chandel, R., 2021. Wireless Penetration Testing: PMKID Attack. Hacking Artic. URL <https://www.hackingarticles.in/wireless-penetration-testing-pmkid-attack/> (accessed 11.8.22).
- Chen, H., Song, M., Guo, Z., Li, R., Zou, Q., Luo, S., Zhang, S., Luo, Q., Hong, J., You, L., 2018. Highly Secure Physically Unclonable Cryptographic Primitives Based on Interfacial Magnetic Anisotropy. *Nano Lett.* 18, 7211–7216. <https://doi.org/10.1021/acs.nanolett.8b03338>
- Digital Evidence and Computer Crime - 3rd Edition [WWW Document], n.d. URL <https://www.elsevier.com/books/digital-evidence-and-computer-crime/casey/978-0-08-092148-8> (accessed 9.12.22).
- EP-0136 - 52Pi Wiki [WWW Document], n.d. URL <https://wiki.52pi.com/index.php/EP-0136> (accessed 11.2.22).
- Ghael, H., 2020. A Review Paper on Raspberry Pi and its Applications. <https://doi.org/10.35629/5252-0212225227>
- Hurley, C., Thornton, F., Puchol, M., Rogers, R. (Eds.), 2004. Chapter 9 - Attacking Wireless Networks, in: *WarDriving*. Syngress, Burlington, pp. 315–353. <https://doi.org/10.1016/B978-193183603-6/50013-7>
- IEEE SA - The IEEE Standards Association - Home [WWW Document], n.d. URL <https://standards.ieee.org/> (accessed 7.31.21).
- Joshi, A., Patil, D.S., Suryawanshi, M., Doke, A., Sharma, A., n.d. A Study on Portable Common Vulnerability Scanner on Raspberry Pi. *Int. J. Eng. Res.* 9, 5.
- Mallery, J., Kelly, P., 2005. *Hardening Network Security*. McGraw Hill Professional.
- May 15, C.C. on, 2020, 2020. The Definitive Cyber Security Statistics Guide for 2020. *Secur. Blvd.* URL <https://securityboulevard.com/2020/05/the-definitive-cyber-security-statistics-guide-for-2020/> (accessed 7.31.21).
- Neu, A., 2016. Types of Wireless Network Attacks | TechRoots. Phoenix TS. URL <https://phoenixts.com/blog/types-of-wireless-network-attacks/> (accessed 9.3.22).
- Salem, M.B., Hershkop, S., Stolfo, S.J., 2008. A Survey of Insider Attack Detection Research, in: *Stolfo, S.J., Bellovin, S.M., Keromytis, A.D., Hershkop, S., Smith, S.W., Sinclair, S. (Eds.), Insider Attack and Cyber Security, Advances in Information Security*. Springer US, Boston, MA, pp. 69–90. https://doi.org/10.1007/978-0-387-77322-3_5
- Singh, R., Kumar, S., 2018. A Comparative Study of Various Wireless Network Monitoring Tools, in: *2018 First International Conference on Secure Cyber Computing and Communication*

(ICSCCC). Presented at the 2018 First International Conference on Secure Cyber Computing and Communication (ICSCCC), IEEE, Jalandhar, India, pp. 379–384. <https://doi.org/10.1109/ICSCCC.2018.8703216>

SRAM PUF: The Secure Silicon Fingerprint [WWW Document], 1591689194843. . Wevolver. URL <https://www.wevolver.com/article/sram.puf.the.secure.silicon.fingerprint> (accessed 9.24.22).

Turnbull, B., Slay, J., 2008. Wi-Fi Network Signals as a Source of Digital Evidence: Wireless Network Forensics, in: 2008 Third International Conference on Availability, Reliability and Security. Presented at the 2008 Third International Conference on Availability, Reliability and Security, pp. 1355–1360. <https://doi.org/10.1109/ARES.2008.135>