Vulnerability analysis in RF locking systems of vehicles in Bogotá, Colombia

Juan Carlos Martínez-Quintero [1], Edith Paola Estupiñán-Cuesta[2], Angie Tatiana Choachi-Sarmiento[3]
[1,2,3] Telecommunications Engineering Program, Universidad Militar Nueva Granada, Colombia

Article Info

Article history:

Received May 5, 2020
Revised Feb 1, 2021
Accepted Feb 18, 2021

Keywords:

Brute-Force Attack
Replay Attack
Decoding
Software Defined Radio
Locking Systems

ABSTRACT

A car electronic security system aims to prevent its theft, the theft of its parts or of elements in its interior. Studying these systems allows identifying and mitigating vulnerabilities. Nowadays, there are different types of attacks on these systems to exploit their vulnerabilities, such as replay, relay, brute-force or jamming attacks, among others. In the last five years in Bogotá, Colombia, the average number of stolen vehicles was 3,073 per year. This research project proposes the detection of vulnerabilities in the security system of vehicles in this city. A sample of 43 vehicles of different brands sold and registered in the city is taken. The replay attack was executed, as well as a modification of the brute-force attack. Results show that most of the implemented security systems in Bogotá are susceptible of being successfully attacked through the proposed methods. The analysis done on the brute-force attack highlights a considerable reduction in time for unlocking the vehicle compared to the conventional attack in more vulnerable RKE systems. Replay attacks turn successful in great part of the sample and, furthermore, it is concluded that the unlocking key code can be generated from the locking one.

Corresponding Author:

Juan Carlos Martínez-Quintero
Telecommunications Engineering Program
Universidad Militar Nueva Granada, Calle 100 Campus
Bogotá, Colombia
E-mail Address: juan.martinezq@unimilitar.edu.co

1. INTRODUCTION

Implementation of security systems in motor vehicles has always been present, whether it is through mechanical systems or complex electronic systems. Manufacturers develop security mechanisms to avoid car theft, the theft of its parts or of elements in its interior. In Bogotá, Colombia, motor car theft daily affects owners according to data delivered by the National Police. Most thefts occur when users park their vehicles on the street, on parking bays, in front of their homes, or in public car parks. In the last five years in Bogotá, the average number of stolen vehicles was 3,073 per year [1]. The most affected city localities were Kennedy and Puente Aranda, specifically in the neighbourhoods of Kennedy Central, Galán, Patio Bonito, Castilla, and Tintal. The highest theft frequency is between July and September, and the preferred colour by delinquents is white. The most appealing motor vehicles for theft are luxury vehicles, followed by SUVs, 4WD vehicles, and trucks [2]. There are different motor vehicle theft methods and delinquents are always looking for new techniques, so as not to be detected. In this sense, vulnerabilities in the electronic security systems of vehicles of different brands are currently made evident. These vulnerabilities are exposed in different scientific articles and lately, on blogs and videos. The most popular types of attack to exploit these vulnerabilities are *replay*, *relay* and *brute-force* attacks. Replay attacks consist in storing the radiofrequency (RF) signal sent by the system key to do the ulterior retransmission, and thus, have access to the vehicle [3]. Relay attacks generally involve two people working together. One is next to the vehicle, while the other is near the key. The interchange of messages between the real key and the vehicle is given by the two individuals carrying out the attack, since they set a channel of communication that is strange to the system. The control unit does not identify that the real key is in a remote place [4]. There is another type of attack called *brute-force* that generates all possible keys for the security systems until there is a match and unlocks the vehicle [5]. From a technical point of view, the security mechanisms for vehicles imply the direct use of alarms, which are considered security systems capable of alerting some abnormal state that may be present in the vehicle, whether it is forcing of handles or bumps on the surface. Traditionally, accessing and starting a motor vehicle is done through a physical key to unlock the door and the starting systems; however, the door and alarm unlock systems by RF have become popular and have evolved with different brands and models of vehicles. Among these systems, to highlight some, there are PKES (Passive Keys Entry and Start) and RKE (Remote Keyless Entry) systems. The RKE systems are designed for the owner to press a button to lock and unlock the vehicle and the PKES ones normally do not require any action by the user for unlocking [6]. The purpose for each system is to constantly be improved and to evaluate its proper functioning to identify flaws that allow refinement; vehicle security systems are no exception. From the point of view of informatics security, specifically the ethical hacking branch, it is defined that attacks can be used as a tool to identify these vulnerabilities and to work in favour of mitigating or eliminating them. The exploitation of vulnerabilities present in the security systems requires intrusion tests that help to verify and evaluate the physical and logical security of the information systems [7]. In the present work, the research is focused on developing attacks and identifying flaws to mitigate them, which is commonly called white hat hacking [8]. This project shows the reality of electronic security systems for the most used vehicles in Bogotá, and experimentally exposes different vulnerabilities in a study that involves types of systems, brands, and models.

The use of SDR (software-defined radio) devices has become popular lately, and it is an accessible technology nowadays. Its main feature is the possibility to reconfigure the implementation of RF systems. This article exposes vehicle security system vulnerabilities at a specific location by using SDR technology in two types of attacks. These vulnerabilities can be more highly exploited due to the easy acquisition of SDR platforms and the lack of sophisticated security systems in common vehicles. This situation can be common in different countries with similar socioeconomic features from this region, given the fact that brands and providers are the same. This article also presents a reverse engineering analysis that allows reducing times in a brute-force attack for RKE systems with fixed code technology.

There are many proposed research projects that evaluate the possible attacks to which security systems of vehicles are exposed; hereunder, the most relevant projects that contribute to the structure of the current one will be highlighted. In [9], a study was done on the attacks to the PKES system in high-end vehicles. The attack was intended to access, start, and drive the vehicle away, by means of the retransmission of RF messages between the car and the key. Tests were made in 10 vehicles of 8 different manufactures concluding that the attack was 90% successful and the remaining 10% could not be executed due to robustness of the security systems. In [10], an analysis of vulnerabilities in the car keys is proposed based on the Hitag2 mechanism. The objective is to program a key fob and to carry out the extraction of the code, so that later, by means of a transponder, the code of the key can be imitated, and thus, be used to open the vehicle. This analysis was done in 8 vehicles, and it determined the possibility of recovering the secret key code sent by the alarm control unit of the vehicle once it is locked or unlocked.

In [11], a strategy attack is carried out, which is called *jam-listen-and-replay* or *replay* attack. This attack requires the transmission of interference and a recorder of signals; the attack is evaluated with six models of vehicles, thus exposing the vulnerabilities presented by these systems, and it is concluded that the solution for keyless remote systems cannot be dimmed safe. In [12], PKES systems are analysed in high-end vehicles and it is revealed that systems are based on a 40-bit encryption for starting and unlocking. It is concluded that these systems are vulnerable to relay attacks, in addition to not performing mutual authentication to unlock the vehicle. In [13], a Hitag2 attack is presented which allows recovering the encrypted key of a vehicle, and thus, the cloning of its remote control. This attack was tested in 17 cars from 8 manufactures and it was successful in 100% of the cases. In [14] and evaluation of 2 attacks is done: Jam-listen-and-replay on the RKE and PKES systems; an Android app was developed to detect the presence of a strong constant signal to a determined frequency. It is concluded that it is easy to detect signals from an alarm control unit through low-cost SDR devices. In [15], the recording of a signal and then its transmission to unlock a vehicle are proposed, without having access to the real code; the tests were done with 6 brands of cars that work at different frequencies. It is concluded that the signals sent by the control unit of the vehicle are within reach of anyone when carrying out a replay attack and that it is possible to capture and transmit them again. In [16], attacks with Hitag2 are studied to identify weaknesses of an alarm encryption, by executing three practical attacks that recover the secret key code using wireless communication. This attack was made with more than 20 vehicles from multiple brands and models, finding thus, several weaknesses within the systems.

Different research projects revealed some of the technologies and hardware used for making the attacks. In [17], a transceiver system based on RTL is studied, a hardware that includes an RF mixer and a USRP. A cost evaluation is done between these devices to carry out a reception and transmission of radio signals. It is concluded that the proposed method that uses the RF and RTL-SDR mixer is the best for the transmission and reception of signals due to its low cost in comparison to the USRP. In [18], the evaluation of the low-cost RTL-SDR dongle is done through GNU-Radio and SDR. Coding of digital signal processing (DSP) algorithms is studied to process radio signals coming from any type of device over the frequency range of 57 kHz to 92.9 Mhz. The facility to listen to, capture and transmit radio signals with any SDR device is determined.

On the other hand, from the studies oriented towards detecting and mitigating attacks in motor vehicles the following can be underlined: In [19], a vehicle security system is presented through GSM/GPRS networks for the control and monitoring of the vehicle. The system, by means of a call to the vehicle phone number, can control it and activate or deactivate the alarm, open and close the locks, start and stop the vehicle, turn on and off the air-conditioning, open the hood and the trunk, and also, request information of the vehicle physical variables such as temperature, gas and oil levels. The car status information is sent via text messages to the user's mobile phone. In [20] an analysis of vulnerabilities is done via wireless technology applied in vehicles in Guayaquil, Ecuador. Through simulation, the vehicle central computer is emulated to develop hacking tests using CAN-BUS networks. It is concluded that it is possible to hack the central computer by means of these networks, and a proposal is made to solve the detected vulnerabilities through a Link Management Protocol (LMP) algorithm. In [21], a detection method of a relay attack in vehicles is proposed by means of the intensity of an RF signal. The method allows, through the intensity of the signal, to deduce if the receiving signal belongs to the car control unit or if it is the retransmission of one of the attackers' device.

Once the theoretical exploration is performed, vulnerabilities were detected in most of the vehicle security systems, which can be exploited through attacks such as replay, relay, signal jamming, and brute-force ones. This research project proposes the detection of vulnerabilities in the security systems of 43 vehicles of different brands, sold and registered in the city of Bogotá between 2000 and 2019. The replay attack is proposed, as well as a modification to the brute-force one. This modification is presented from the analysis of known security system encryption in vehicles from different manufacturers. The implemented brute-force attack allows a reduction of substantial time in the unlocking of a vehicle compared to the brute-force attack using all possible combinations for the number of bits of a code. For the execution of the attacks, the low-cost software-defined radio (SDR) HackRF One platform and the GNURadio software are used. Two scenarios are proposed in which the SDR system is used to capture, replicate, or generate the RF signals with which the attacks are made in the operation frequency of the key fob. The SDR system consists of the SDR platform and a PC. The type of blocking system, the key code number of bits, and the carrier modulation, encryption and frequency were analysed, and the attacks algorithms were implemented. The geographical limitation of the study reveals average situations of the region that make these kinds of attacks prevail even today.

This article is initially organised with a state of the art that confirms the importance of evaluating the vulnerabilities within these security systems. The second section presents a theoretical contextualization of security systems in vehicles, exiting attacks, SDR, among others. The third and fourth section present the proposed scenarios and the analysis of obtained results, respectively. Finally, conclusions on the executed attacks are presented.

2. RESEARCH METHOD – THEORETICAL CONTEXTUALIZATION

2.1 Theoretical contextualization on alarms

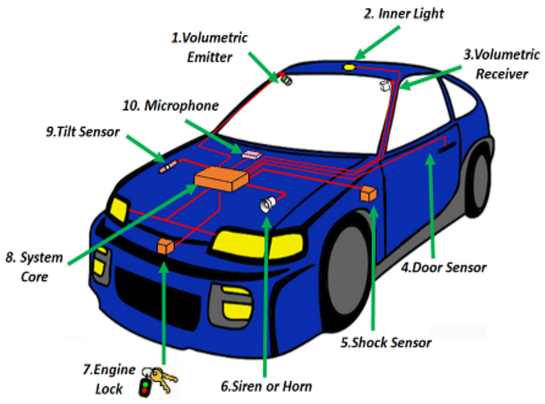
An alarm is considered a system capable of alerting and announcing the change of an abnormal state presented in any setting. In a vehicle, it is an electronic security device considered as an audible system capable of alerting cases such as: forcing of doors, tilting or sudden movement, breakage of glass, among others. Accessing and starting a vehicle is done by means of a physical key and a control command system (alarm) which enables unlocking the door lock system and the starting of the vehicle. However, alarm and door locking unlocking systems through RF have become popular. Alarms in vehicles enable the detection of the signal by means of RF sensors and send the signal to the alarm central kernel, which analyses the signal, identifies its authenticity, and suspends the car security elements. Alarms have average ranges of 10 to 20 m when unlocking a vehicle, and they have operation frequencies of 315, 370, and 434 MHz. As time progresses, alarms have evolved in line with different brands and models of vehicles in the market. Two security systems can be found in vehicles: PKES and RKE. RKE are electronic access systems that can be controlled in the distance. PKES are systems that work automatically when the user is near the vehicle. They unlock the door when the user is coming closer or when the door handle is pulled and lock it when the user walks away or touches the vehicle when exiting. Table 1 describes some of the main characteristics of these kinds of systems [13][15][22].

Table 1. Characteristics of vehicle security systems

Systems	Characteristics
PKES	• This system includes an RF control unit or a smart card. • The module within the vehicle continually sends encrypted messages. • If the control unit is within range, it responds. • If encrypted messages are correct, the vehicle and key fob identify each other, and the door opens. • It always uses different codes avoiding replay attacks.
RKE	• The user who holds the device presses a button to lock or unlock. • This system operates at 315 MHz for vehicles manufactured in North America. • It operates at 433,92 MHz for European, Japanese, and Asian vehicles. • It enables controlling the car starting systems, the security alarm, horn, lights, and trunk.

2.1.1 Alarm systems

Alarm systems are passive security elements capable of warning about possible issues or abnormal situations, thus fulfilling a deterrent function. These systems are composed of one or more sensors connected to an alarm horn. Most systems have door, movement, tilting and vibration sensors, and a horn and radiofrequency receiver to enable wireless control. The main components for a vehicle alarm system can be seen in Figure 1. [13]



1. *It is responsible for emitting the alert sound in case any sensor is activated.*
2. *This light comes on when the doors are poorly closed.*
3. *It is activated when breakage of a glass is detected, since breaking the glass has its own frequency.*
4. *It is responsible for alerting the owner if the door, trunk, or hood is not properly closed or secured.*
5. *It is responsible for alerting if someone hits or violently moves the vehicle; the sensor sends a signal to the core that indicates the intensity of the movement.*
6. *It emits sounds that help to identify the owner of the vehicle if someone is trying to violate the safety of the vehicle.*
7. *It is a physical device that allows full internal or external control of the car alarm.*
8. *Its function is to close or open the switches that activate the alarm system devices.*
9. *It is activated when the car experiences or detects any inclination above ground level.*
10. *This works in conjunction with the volumetric receiver since hearing a very loud noise will activate the alarm.*

Figure 1. Components of an alarm system in vehicles

2.2 Types of Attacks

An attack consists in taking advantage of a system or software vulnerability, to obtain some sort of benefit; for a vehicle security system, an attack is considered when a person takes advantage of a weakness or flaw in the software or hardware to be able to steal it or to steal belongings within it. Among attacks to the electronic system of a vehicle the following can be underlined [23]:

- **Replay Attack:** In this type of attack, the attacker takes the content of the message, it means, the signal sent by the alarm control unit, and stores it to subsequently perform the retransmission of the information and thus, have access to the vehicle. Afterwards, the attacker can make use of the obtained information at any moment [3].
- **Relay Attack:** It is done to vehicles equipped with smart keys. To execute the attack, a receiver and a transmitter that mutually amplify signals are needed; the transmitter must be located as close as possible to the vehicle, and the receiver must be located at 8 meters or less from the alarm control unit. The attack can be successful because the vehicle detects the signal that was replicated from the owner's original key. [4]
- **Signal Jamming Attack (Master Key):** This attack requires a control unit identical to that of the owner of the car to deactivate the alarm and have access to it; in some cases, this control unit also works as a starting key. With this device the security system circuit is altered and there is no need for physical forcing of the vehicle [24].

- **Brute-Force Attack:** This type of attack involves trying to decipher a password by trial and error. Attackers try several random combinations until the correct pattern matches. Another way to perform this attack is to capture the signal sent by the alarm to, subsequently, decode it and obtain its key code, thus, being able to transmit it and have access to the vehicle**.**[5]

2.3 SDR (Software-Defined Radio)

SDR is defined as: "Radio in which some or all functions of the physical layer are defined by software". It is defined as a collection of hardware and software technologies where the operative functions of the radio are implemented by modifiable software or firmware that works with programmable processing technologies. These devices include FPGA, DSP, General Purpose Processor (GPP), System on a Chip (SoC), and specific programmable applications. The use of these technologies enables adding new capabilities and wireless functions to the existing radio systems without requiring new hardware. [25] The most common SDR platforms are USRP, RTL, and HackRF One. The latter provides the same functions as the USRP but at a low cost. Both USRP and HackRF can transmit or receive radio signals in the range from 1 MHz to 6 GHz. HackRF One is an open-source hardware platform that can be used as a USB peripheral or can be programmed for standalone operation. [26]

3. EXPERIMENTAL DESIGN AND SIGNAL ANALYSIS

3.1 Proposed Scenarios

Two scenarios are proposed to detect and exploit vulnerabilities in RF key fob security systems. In both scenarios, an SDR system is used to capture or generate the RF signals with which attacks are performed on the key fob frequency operation. The SDR system consists of the SDR platform and a computer. Table 2 shows the technical characteristics of the equipment used for both scenarios.

Table 2. Characteristics of equipment used for proposed scenarios

Equipment	*Characteristics*
SDR Hack RF one Platform	• Half-duplex Transceptor. • Up to 20 million samples per second. • 8-bit quadrature samples (8-bits I and 8-bits Q). • Compatible with GNU Radio. • RX and TX gain configurable by software and baseband filter. • Software controlled antenna port power (50 mA to 3.3 V). • SMA female connector. • Clock input and output for synchronization. • High speed 2.0 USB. • Open source.
Vehicle	• To execute the attacks, 43 vehicles were chosen from models between 1998 to 2019. Their characteristics are random since they vary depending on the model of each one of them.
Scenario Characteristics	• Scenario 1: Key fob (Its characteristics are random since they vary depending on model and manufacturer) • Sample Frequency: 2Msps • Carrier Frequency: 315MHz, 370MHz, or 443MHz

The first scenario is shown in Figure 2. In this scenario, the signal is generated by the key fob that normally controls access and locking of the vehicle. The SDR system works as a passive signal receiver that crosses the environment freely and unrestrictedly. The attack consists in retransmitting the signal captured by means of the SDR platform, cheating the security system into believing it is the real key fob transmission to accomplish its deactivation. The signal is retransmitted using the same SDR platform used for the capture (it is also possible to use one of similar characteristics). This attack is known as "Replay Attack". The software used for capturing, storing, and retransmitting the signal is called GNURadio. The distance between devices (PC and HackRF One) and the vehicle depends on the space where the car is located to carry out the execution of the proposed scenario.

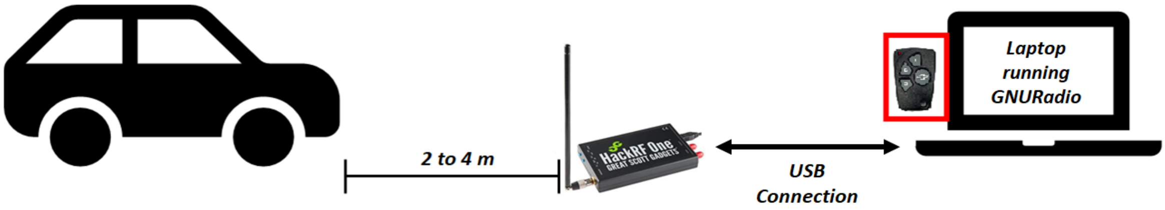


Figure 2. Assembly implemented for Scenario 1

Figure 3 shows the second proposed scenario. In this scenario, there is no available key fob to generate the RF signal that opens the vehicle. In this case, it starts from knowing the signal that must be transmitted. It is necessary to know the type of locking system, the number of bits in the key code, modulation, encryption, and carrier frequency. The main principle is the implementation of an algorithm to generate passwords through a binary counter. These passwords are encoded, modulated, and transmitted with characteristics of a real signal using the SDR platform. This attack is known as "***Brute-Force Attack***" and seeks to generate all possible passwords for the security system until there is a match and the car is unlocked. The generation and transmission of signals were carried out through the GNURadio software. To know the unlocking signals of an RKE-type security system, the received signal from a real system key fob was analysed using Matlab.

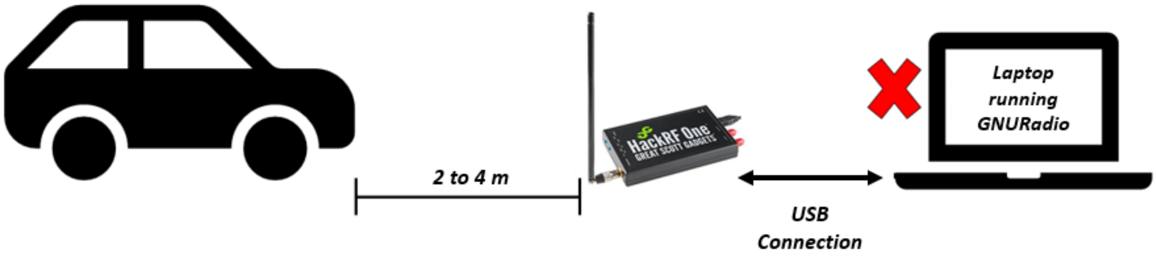
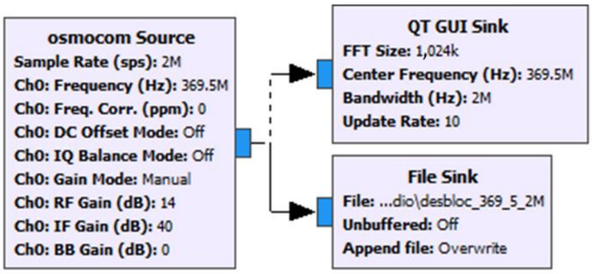


Figure 3. Assembly implemented for Scenario 2

3.2 Attack Execution, Processing and Analysis

3.2.1 Replay Attack

A replay attack is executed in two parts. The first part is storage, for which the block diagram of Figure 4 is defined and implemented through GNURadio. This allows storing the signal sent by the RF key fob that travels freely through the air. The "Osmocom Source" block is used to configure the HackRF parameters. The "Ch0: Frequency (Hz)" parameter configures the main oscillator frequency; this frequency can vary depending on the security system key fob that is being analysed. The "Sample Rate (SPS)" parameter is used to configure the sample frequency, and, for practical effects, it is equivalent to the bandwidth of the received signal due to the SDR device IQ configuration. The RF and IF gain is configured to maximum values to obtain a good signal level at a greater distance. The "File Sink" block enables storing the received signal in a complex format (IQ). The signal is stored without any modification. The "QT GUI Sink" block processes the signal through FFT (Fast Fourier Transform) and presents the result in a graphical user interface. Figure 4 shows how the signal from a key that works in a frequency close to 370 MHz is stored. The spectrum delivered by this block diagram is shown in Figure 5. In the central spectrum frequency (369.5 MHz), an undesired peak generated by the SDR platform is observed. This peak is the result of the mixing process in the RF stage of the HackRF platform. The highest observed peak in the spectrum is the signal in frequency transmitted by the key fob. If the local oscillator is tuned at the same key frequency, interference is generated between the two peaks and it is possible that the attack fails due to decrease of the signal quality. For this reason, the local oscillator is configured under the transmission frequency of the key fob.



osmocom Source
Sample Rate (sps): 2M
Ch0: Frequency (Hz): 369.5M
Ch0: Freq. Corr. (ppm): 0
Ch0: DC Offset Mode: Off
Ch0: IQ Balance Mode: Off
Ch0: Gain Mode: Manual
Ch0: RF Gain (dB): 14
Ch0: IF Gain (dB): 40
Ch0: BB Gain (dB): 0

QT GUI Sink
FFT Size: 1,024k
Center Frequency (Hz): 369.5M
Bandwidth (Hz): 2M
Update Rate: 10

File Sink
File: ...dio\desbloc_369_5_2M
Unbuffered: Off
Append file: Overwrite

Figure 4. Signal Storing Program in GNURadio

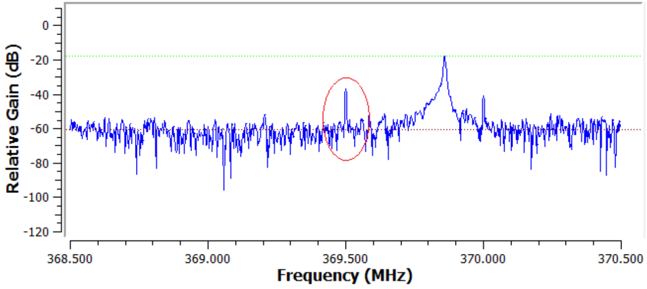
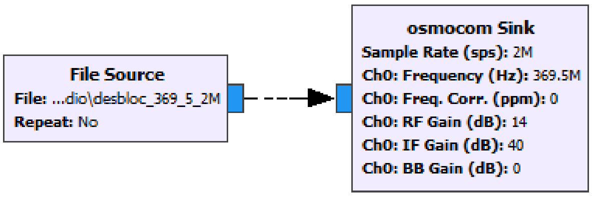


Figure 5. Spectrum in frequency of the key signal

The second part of the attack consists in transmitting the stored signal. The block diagram in Figure 6 shows how the stored signal transmission is done. The program consists of 2 blocks; the first block, "*File Source*", reads the signal stored in a complex format and delivers it to the "*Osmocom Sink*" block. The latter is configured with the same parameters from Figure 4. The execution of this block diagram enables the transmission of the stored signal coming from the security system key fob. In this sense, the vehicle receiving system obtains the same signal that would be transmitted by the original key.



File Source
File: ...dio\desbloc_369_5_2M
Repeat: No

osmocom Sink
Sample Rate (sps): 2M
Ch0: Frequency (Hz): 369.5M
Ch0: Freq. Corr. (ppm): 0
Ch0: RF Gain (dB): 14
Ch0: IF Gain (dB): 40
Ch0: BB Gain (dB): 0

Figure 6. GNURadio signal transmission program

3.2.2 Brute-Force Attack

For the execution of the ***brute-force attack***, two tests were carried out. This attack is based on the analysis of the signal transmitted by the security system key fob. The signal was analysed using Matlab. Figure 7 shows a baseband signal into the time domain. The acquisition of the signal was performed under similar conditions as in Figure 5.

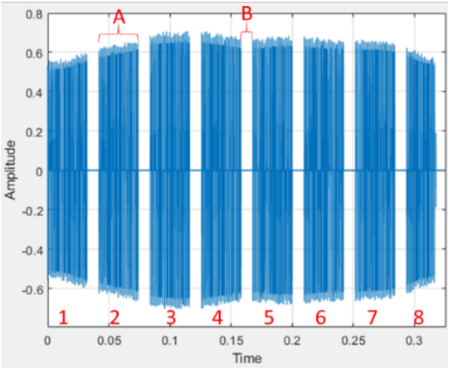


Figure 7. Signal transmitted by the baseband key

In Figure 7, the letter "A" represents a 25-bit key coded and modulated in ASK (Amplitude Shift Keying). The 25-bit key is repeated eight times in each transmission. Letter "B" is the space (without transmission) between each repetition. In total, there are 7 blank spaces (B), which are equivalent to the transmission time of 6 bits each. Figure 8 portrays a diagram of the complete frame composition.

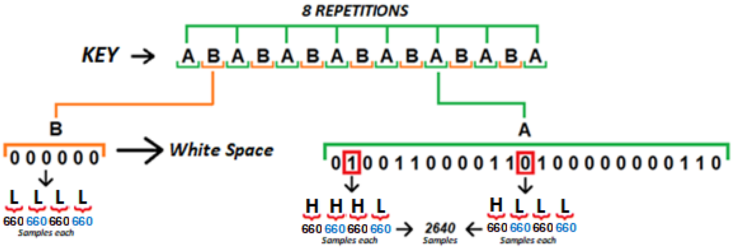


Figure 8. Key formation and composition

In the lower part of Figure 8, the coding of each bit and the blank spaces are also detailed. Each of the 25 bits is coded and modulated in ASK as shown in Figure 9. The coding of a "1" is a sequence of three high levels and one low level. The coding of a "0" is a sequence of one high level and three low levels. Each level (high or low) has a duration of 330µs, which is equivalent to 660 samples if a sampling rate of 2 Msps is used as in this case. A blank space "B" is equivalent to 6-bit time. Each of these spaces equivalent to one bit is coded with a sequence of four levels in low (L), that is 24 L for each "B".

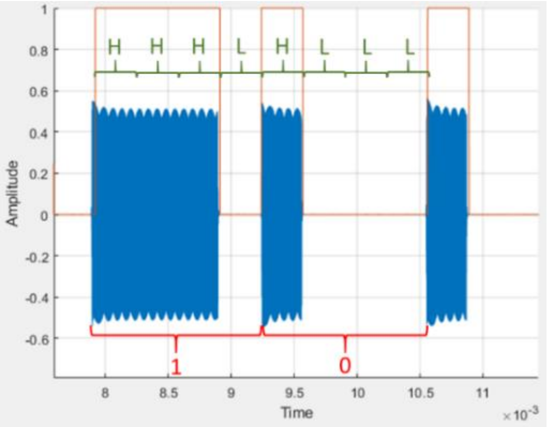


Figure 9. "1" – "0" sequence coded and modulated in ASK

To generate the signal from SDR, it is appropriate to handle the signal in terms of the number of samples per level or per bit. The number of samples per bit is determined from Equation 1.

$$n_b = n_L \times L_b \qquad (1)$$

Where n_b is the number of physical samples per bit, n_L is the number of physical samples per level, and L_b is the number of levels per bit. In this case, $n_L = 660$ and $L_b = 4$, therefore, $n_b = 2{,}640$ samples. The number of samples for the entire key is given by Equation 2.

$$n_k = n_b \times Nb_{A+B} \times N_R \qquad (2)$$

Where n_k is the number of samples of the key, Nb_{A+B} is the number of bits of the sequence "A" plus the equivalent bits of the blank sequence "B", and NR is the number of repetitions of sequences "A" and "B" together. For this case, $n_b = 660\ samples$, $Nb_{A+B} =$ 31 bits y $N_R = 8$, therefore, $n_k = 654{,}729\ samples$, considering a "B" at the end of the frame in Figure 8. The duration of the key T_k can be found using Equation 3.

$$T_k = \frac{n_k}{f_s} \qquad (3)$$

Where f_s is the sampling frequency: In this case, $T_k = 0.32736s$. This last time is the same it would take for an SDR system to generate a key in a brute-force attack. If a 25-bit counter is used to generate consecutive keys, the maximum worst-case attack time can be calculated from Equation 4.

$$T_{MA} = T_k \times 2^n \qquad (4)$$

For Equation 4, T_{MA} is the maximum execution time of the attack and n is the number of bits in the key. Solving the equation for the 25-bit key gives a maximum attack execution time of 10,984,378s or 127.13 days. The total number of keys generated is $2^n = 33{,}554{,}432$. Figure 10 illustrates the flowchart of the program developed to execute the brute-force attack.

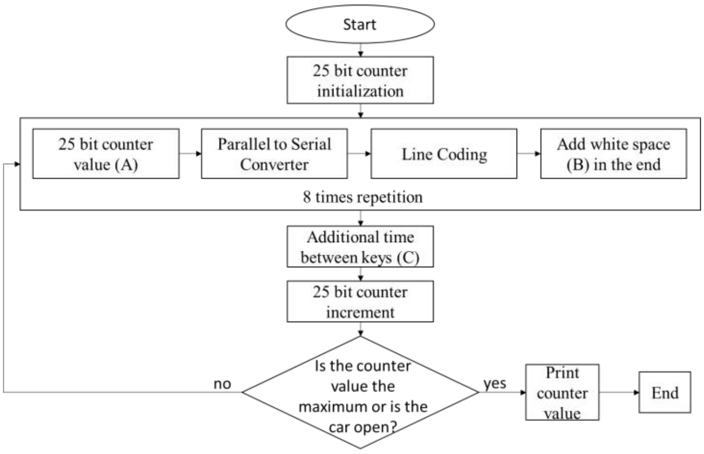

Start
25 bit counter initialization
25 bit counter value (A)
Parallel to Serial Converter
Line Coding
Add white space (B) in the end
8 times repetition
Additional time between keys (C)
25 bit counter increment
Is the counter value the maximum or is the car open?
no
yes
Print counter value
End

Figure 10. Flowchart for key generation and transmission

The program uses a 25-bit counter as a key generator; each count represents a testing "A" key. The counter passes through a parallel-to-serial converter to achieve the bit-by-bit transmission. Each bit is coded as shown in Figure 9; after the transmission of the last bit, a blank space "B" is generated. The red signal shown in Figure 11 was generated with a counter value started with the code of a testing vehicle and without subsequent counter increment. Figure 11 shows the generated signal vs. the original stored signal from the testing vehicle. The times meet the analysis described in this section.

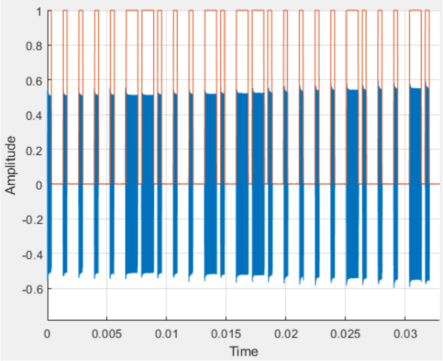


Figure 11. Generated signal (red) vs. actual stored key (blue)

To test the brute-force attack, a counter was started 50 values behind the key value of the testing vehicle, and it was increased to 50 values afterwards. The counter is incremented by leaving only a "B" space between the end of a key and the beginning of another. When the attack is executed, the vehicle does not open; however, when the counter passes the correct key, the security system makes a particular sound. Subsequently, by performing the same experiment, the time space between each key was increased before the counter increment. Each test increments an additional "B" space at the end of the key, and it is maintained for the 100 iterations. At the end, the vehicle is unlocked with 25 "B" spaces as shown in Figure 12. The 25 repetitions of "B" form a "C" sequence.

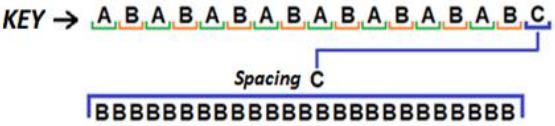


Figure 12. Initial key plus "C" spacing

Equation 5 is a modification of equation 2 to determine the total number of samples in the key and the additional "C" space.

$$n_k = n_b \times (Nb_{A+B} \times N_R + Nb_C) \quad (5)$$

In this equation Nb_C is the equivalent number of bits in "C" space. Considering that "B" has a 6-bit equivalent -as defined at the beginning of this section- "C" would have an equivalent number of bits 25 times greater, which means 150 bits. This gives a total number of samples of $n_k = 1{,}050{,}720$ per key. According to Equation 3, the duration of a key is $T_k = 0.52536$s, and, using Equation 4, a maximum attack execution time can be obtained, which is $T_{MA} = 17{,}628{,}156s$ or 204.02 days using all possible combinations. Performing this attack ensures that the vehicle will be unlocked when the keys match. Nonetheless, a long-time attack as such is not practical.

4 RESULTS AND DISCUSSION

4.1 Replay Attack

The replay attack was executed in different vehicles like those shown in Figures 13 and 14. In these Figures, the real scenario of the assembly displayed in Figure 2 appears.

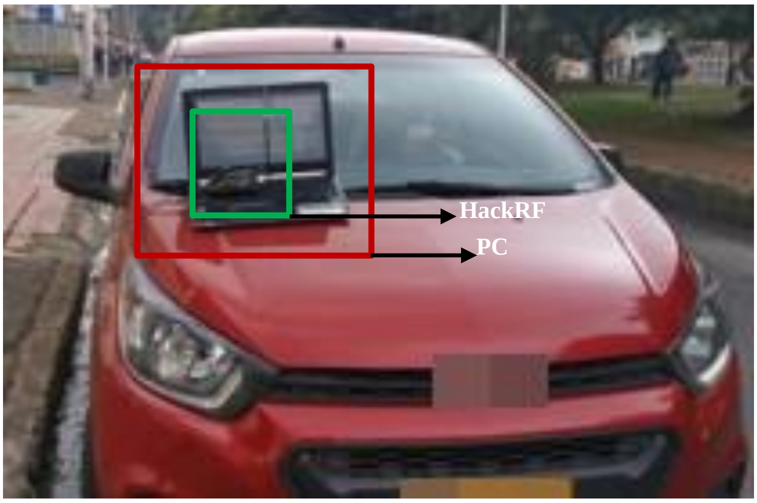


Figure 13. Evidence 1 of scenario 1

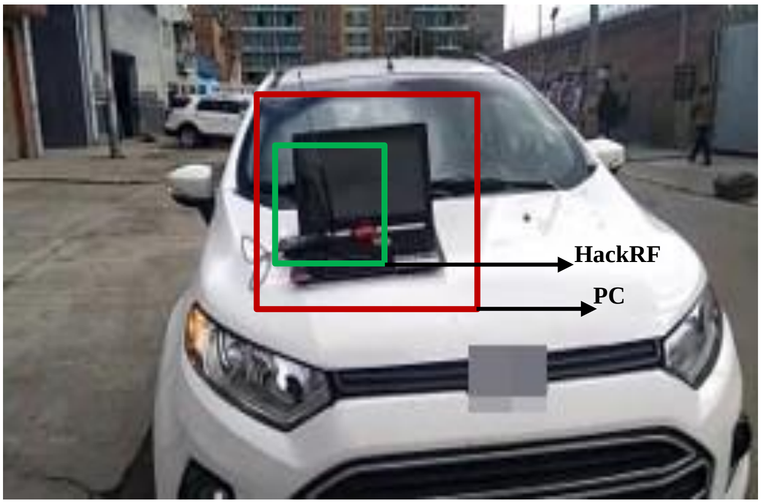


Figure 14. Evidence 2 of scenario 1

Table 3 shows the results of the attack and some basic characteristics of the security system for 43 different models of -randomly selected- vehicles registered in the city of Bogotá. The vehicles shaded in blue have a security system that uses a technique called "rolling code", with which the transmitter and receiver generate identical pseudo-random keys based on the same seed. When the key is afar, it may generate codes that the receiver cannot collect; to solve this problem, the receiver (in the vehicle) generates the next 256 keys and accepts any of them in that range, as long as it does not repeat. To execute the attack in this type of vehicle, one or more key signals are stored away from the receiver, where it has no coverage, more than 10 m in practice. After a successful execution of a stored key, it can no longer be used to open the vehicle again. If several keys are stored, each one can only once be used to open the vehicle; under these conditions, the attack works.

Table 3. Characteristics of the implemented scenario

ID Vehicle	***Model***	***Autospam Signal***	***Frequency (MHz)***	***Attack Result***	***Security System***
1	2015	No	433.427	Successful	***RKES***
2	2012	No	433.427	Successful	***RKES***
3	2008	No	433.427	Successful	***RKES***
4	2015	No	433.427	Successful	***RKES***
5	2013	No	433.427	Successful	***RKES***
6	2008	No	315.166	Successful	***RKES***
7	2016	No	433.427	Successful	***RKES***
8	2018	No	433.427	Successful	***RKES***
9	2010	No	433.427	Successful	***RKES***
10	2017	No	433.427	Successful	***RKES***
11	2010	No	433.427	Successful	***RKES***
12	2009	No	433.427	Successful	***RKES***
13	2012	No	433.427	Successful	***RKES***
14	2012	No	433.427	Successful	***RKES***
15	2015	No	433.427	Successful	***RKES***
16	2015	No	433.427	Successful	***RKES***
17	2008	No	316.264	Successful	***RKES***
18	2011	No	433.427	Successful	***RKES***
19	2010	No	433.427	Successful	***RKES***
20	2008	No	433.427	Successful	***RKES***
21	2016	No	433.427	Successful	***RKES***
22	2015	No	433.427	Successful	***RKES***
23	2016	No	433.427	Successful	***RKES***
24	2017	No	433.427	Successful	***RKES***
25	2016	No	433.427	Successful	***RKES***
26	2018	No	433.427	Successful	***RKES***
27	2014	No	433.427	Successful	***RKES***
28	2014	No	433.427	Successful	***RKES***
29	2012	No	315.497	Successful	***RKES***
30	2019	No	433.427	Successful	***RKES***
31	2000	No	433.427	Successful	***RKES***
32	2011	No	433.427	Successful	***RKES***
33	1998	No	433.427	Successful	***RKES***
34	2008	No	433.427	Successful	***RKES***
35	2012	No	433.427	Successful	***RKES***
36	2015	No	433.427	Successful	***RKES***
37	2011	No	433.427	Successful	***RKES***
38	2000	No	433.427	Successful	***RKES***
39	2008	No	433.427	Successful	***RKES***
40	2018	Yes	433.427	Not Successful	***PKES***
41	2019	Yes	314.166	Not Successful	***PKES***
42	2015	Yes	433.427	Not Successful	***PKES***
43	2008	Yes	433.842	Not Successful	***PKES***

Once the attack was carried out, it was determined that 39 from the 43 vehicles managed to be unlocked, which is equivalent to 90.69% of the total samples. The 51.16% of vehicles (22) are opened simply by retransmitting the key signal without any additional conditions. The 39.53% of all vehicles (17) have a “rolling code” technique coding system and are opened with the condition described above. The remaining vehicles have the particularity that the type of security system is PKES. The latter type of security system is more robust and cannot be unlocked through this attack. In this type of system, when the stored signal is replicated, the vehicle is locked during some time and the owner must carry out a procedure to unlock it; at this point, it is vulnerable to the theft of its belongings in the case of being approached by criminals.

4.2 Brute-Force Attack

The execution of this type of attack as described in section 3.2.2 is only possible for the 22 vehicles in Table 3, which support the replay attack without any conditions because they do not have an additional coding; however, as mentioned before, the execution of the attack is not viable from a temporary point of view. Based on this impediment, an analysis of the codes provided by each key fob was performed, considering the brand and model of each vehicle, to determine any possible relationship between them. The 22 vehicles were grouped into 4 brands which were named W, X, Y, and Z, corresponding to 4 of the most representative manufacturers in the market at a national level; the letters are used instead of the manufacturers’ own names. The analysis of the codes grouped by brand and model resulted in the matching of bits in certain positions within the 25 bits. For example, the three lightest bits of all manufacturers' keys match and have the value "010". These fixed values decrease the time T_{MA} because the value of n in Equation 4 decreases to 21. The conventions that were used in the Tables for each brand to show the bit matches found are defined below. Table

4 sets out the convention for showing coincidence in the least significant bits of two or more codes within the same brand. Yellow, for example, indicates a match in bit 3, in addition to the three fixed bits mentioned above. So, if it is yellow with dashes, it indicates that the coincidence in bit 3 is matched at "1", and if it is yellow without dashes, it indicates that bit 3 is matched at "0".

Table 4. Least significant bit conventions

BRAND W, X, Y y Z

24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Description Matching Bit Position
																					🟨				Bits (0-3), bit 3 is 0
																					▨				Bits (0-3), bit 3 is 1
																				🟦	🟨				Bits (0-4), bit 4 is 0
																				▨	🟨				Bits (0-4) bit 4 is 1
																			🟧	🟦	🟨				Bits (0-5), bit 5 is 0
																			▨	🟦	🟨				Bits (0-5), bit 5 is 1
																		◼	🟧	🟦	🟨				Bits (0- 6), bit 6 is 0
																		▨	🟧	🟦	🟨				Bits (0-6), bit 6 is 1
																	🟥	◼	🟧	🟦	🟨				Bits 0-7
																🟪	🟥	◼	🟧	🟦	🟨				Bits 0-8

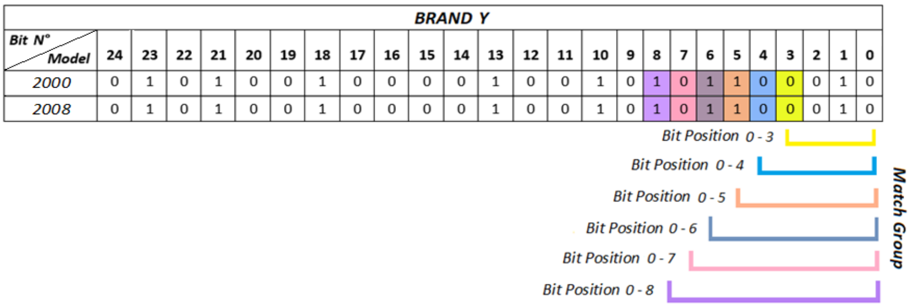
Table 5 establishes the convention for showing coincidence in the most significant bits of two or more codes within the same brand. For the convention handled in the most significant bits, green represents coincidences of 2 or 3 bits, but it can vary with dashes, dots, or other for different matching combinations in the 3 bits.

Table 5. Most significant bit conventions

BRAND W, X, Y y Z

24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Description Matching Bit Position
🟩	🟩	🟩																							Bits (22-24), bit 23 and 22 are 0
▨	▨	▨																							Bits (22-24), bit 23 and 22 are 1
🟩	🟩	🟩																							Bits (22-24), bit 24 and 22 are 0
🟩	🟩	🟩																							Bits (22-24), bit 24 and 22 are 1
▨	▨	▨																							Bits (22-24), bit 22 is 1
🟩	🟩	🟩	▨																						Bits (21-24), bit 21 is 0
🟩	🟩	🟩	🟨																						Bits (21-24), bit 21 is1
🟩	🟩	🟩	🟨	🟦																					Bits 24-20
🟩	🟩	🟩	🟨	🟦	🟧																				Bits 24-19
🟩	🟩	🟩	🟨	🟦	🟧	◼																			Bits 24-18

Figures 15 and 16 show an example of the colour convention for the 2000 and 2008 models of ***brand Y***, in the least significant bits, and the 2010 and 2017 models of ***brand X***, in the most significant bits.

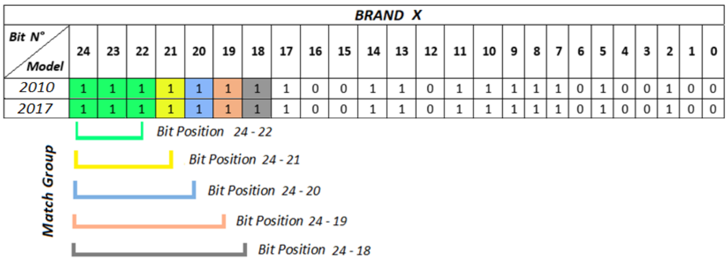


BRAND Y

Bit N° / Model	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
2000	0	1	0	1	0	0	1	0	0	0	0	1	0	0	1	0	1	0	1	1	0	0	0	1	0
2008	0	1	0	1	0	0	1	0	0	0	0	1	0	0	1	0	1	0	1	1	0	0	0	1	0

Bit Position 0 - 3
Bit Position 0 - 4
Bit Position 0 - 5
Bit Position 0 - 6
Bit Position 0 - 7
Bit Position 0 - 8
Match Group

Figure 15. Colour assignment identification according to the coincides in the LSB (Least Significant Bit)



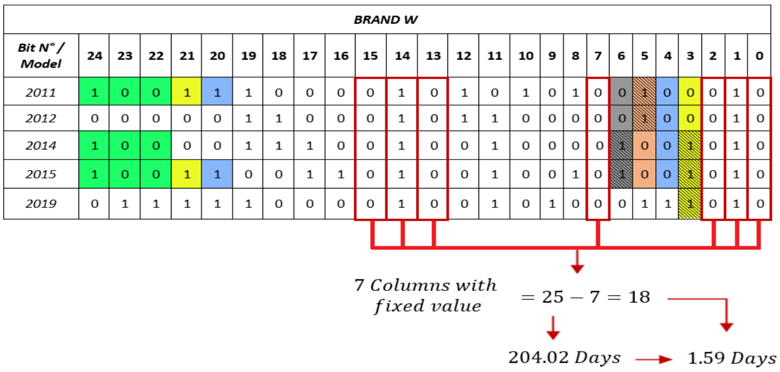
BRAND X

Bit N° / Model	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
2010	1	1	1	1	1	1	1	1	0	0	1	1	0	1	1	1	1	1	0	1	0	0	1	0	0
2017	1	1	1	1	1	1	1	1	0	0	1	1	0	1	1	1	1	1	0	1	0	0	1	0	0

Bit Position 24 - 22
Bit Position 24 - 21
Bit Position 24 - 20
Bit Position 24 - 19
Bit Position 24 - 18
Match Group

Figure 16. Colour assignment identification according to MSB matches (Most Significant Bit)

Tables 6 to 9 show different matches that have been grouped for each of the 4 brands. The tables show the model of the vehicle and the security system unlocking key. In Tables 6, 8, and 9, 7 columns coincide in different positions per brand. The columns that coincide are enclosed with a red border, regardless of their filling colour, as shown in Figure 17. If the 7 fixed bits for these brands are considered, n decreases to 18 in Equation 4 and the maximum execution time of the T_{MA} attack results in 1.59 days.



BRAND W																									
Bit N° / Model	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
2011	1	0	0	1	1	1	0	0	0	0	1	0	1	0	1	0	1	0	0	1	0	0	0	1	0
2012	0	0	0	0	0	1	1	0	0	0	1	0	1	1	0	0	0	0	0	1	0	0	0	1	0
2014	1	0	0	0	0	1	1	1	0	0	1	0	0	1	0	0	0	0	1	0	0	1	0	1	0
2015	1	0	0	1	1	0	0	1	1	0	1	0	0	1	0	0	1	0	1	0	0	1	0	1	0
2019	0	1	1	1	1	1	0	0	0	0	1	0	0	1	0	1	0	0	0	1	1	1	0	1	0

Figure 17. Example of time reduction T_{MA} by fixed value in 7 columns

In Table 6, additionally to the coincidences per column, there is a coincidence in the least significant bits depending on the model of the vehicle. In the vehicles of models 2011 and 2012, there is a match in the 7 least significant bits; therefore, if we add the match in the bits from 13 to 15, we have a total of 10 matching bits. It decreases T_{MA} to 0.199 days (4.78 hours). The same happens with the vehicles of models 2014 and 2015, with a variation in the value of bits 3, 5, and 6. In addition to this, there is also a match in the 3 most significant bits, having a total of 13 matching bits in those years. For the latter case, T_{MA} would go down to 0.024 days (0.5977 hours). For the vehicle of model 2019, the maximum coincidence is 8 bits.

Table 6. Brand W Unlocking Keys

BRAND W																											
Coincidences (Bit 20 - 24)	Bit N° / Model	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Coincidences (Bit 0 - 7)
● ● ●	2011	1	0	0	1	1	1	0	0	0	0	1	0	1	0	1	0	1	0	0	1	0	0	0	1	0	● ● ● ●
	2012	0	0	0	0	0	1	1	0	0	0	1	0	1	1	0	0	0	0	0	1	0	0	0	1	0	● ● ● ●
●	2014	1	0	0	0	0	1	1	1	0	0	1	0	0	1	0	0	0	0	1	0	0	1	0	1	0	● ● ● ●
● ● ●	2015	1	0	0	1	1	0	0	1	1	0	1	0	0	1	0	0	1	0	1	0	0	1	0	1	0	● ● ● ●
	2019	0	1	1	1	1	1	0	0	0	0	1	0	0	1	0	1	0	0	0	1	1	1	0	1	0	●

Table 7 is the one with most coincidences per column, with a total of 9. With 9 fixed bits, the time T_{MA} is reduced to 0.3984 days (9.563 hours). Besides, additional coincidences can be made in bit 6 and bits 20, 22, and 23, depending on the model of the vehicle.

Table 7. Brand X Unlocking Keys

BRAND X																											
Coincidences (Bit 19 - 24)	Bit N° / Model	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Coincidences (Bit 0 - 7)
● ● ●	2010	1	1	1	1	1	1	1	1	0	0	1	1	0	1	1	1	1	1	0	1	0	0	1	0	0	●
●	2012	1	0	0	1	1	0	1	0	1	0	0	1	1	0	0	0	1	0	0	1	0	0	1	0	0	●
●	2015	1	0	0	1	0	1	1	0	1	1	1	0	0	1	1	1	1	0	1	1	0	0	1	0	0	● ●
● ● ●	2017	1	1	1	1	1	1	1	0	0	0	1	1	1	1	0	0	0	0	1	1	0	0	1	0	0	● ●
● ●	2018	1	1	1	1	1	0	1	1	0	1	1	1	0	0	1	1	0	1	1	1	0	0	1	0	0	●

In Table 8, apart from the coincidences in the columns, other coincidences can be found in bits 4, 5, 6, 7, 8, 22, 23, and 24, depending on the model. However, in this Table, there is a 2008 vehicle that is out of the matching patterns.

Table 8. Brand Y Unlocking Keys

BRAND Y

Coincidences (Bit 20 - 24)	*Bit N° / Model*	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	*Coincidences (Bit 0 - 8)*
●●	*2000*	0	1	0	1	0	0	1	0	0	0	0	1	0	0	1	0	1	0	1	1	0	0	0	1	0	●●●●●
	2008	1	1	1	1	0	0	1	0	0	0	0	1	1	1	1	1	1	0	1	1	0	0	0	1	0	●●●●●
●●	*2008*	1	0	1	0	0	0	1	1	0	1	0	1	1	0	0	0	0	0	1	1	1	0	0	1	0	
	2008	0	0	1	0	0	0	0	0	0	1	1	1	0	1	0	1	0	1	1	0	0	0	0	1	0	●●
●●	*2009*	1	0	1	0	0	1	1	0	0	0	0	1	0	1	0	1	0	0	0	0	0	0	0	1	0	●●●●
●●	*2010*	0	1	0	1	0	1	0	0	0	0	1	1	0	1	0	0	1	0	0	0	0	0	0	1	0	●●●●
●	*2011*	0	1	0	0	0	0	0	0	0	0	1	1	1	1	0	0	1	1	0	0	0	0	0	1	0	●●●

Table 9 is the one with the fewest number of coincidences. Nonetheless, it maintains 7 columns with fixed values.

Table 9. Brand Z Unlocking Keys

													BRAND Z														
Coincidences (Bit 21- 22)	*Bit N° / Model*	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	*Coincidences (Bit 0 - 6)*
	2011	0	0	0	0	1	1	0	0	1	0	0	0	1	1	1	0	0	1	0	1	0	0	0	1	0	●●●
	2015	1	1	0	0	1	0	0	0	1	0	0	0	1	1	1	0	1	0	1	1	1	0	0	1	0	
	2016	1	0	0	0	0	0	0	0	1	0	0	0	1	1	0	0	0	1	1	1	0	0	0	1	0	●●
●	*2016*	0	0	1	0	0	0	0	1	0	0	1	0	1	0	1	1	1	0	0	0	0	0	0	1	0	●
●	*2018*	0	0	1	0	1	1	0	1	0	0	0	1	0	0	1	1	0	0	0	1	0	0	0	1	0	●●●

Table 10 shows the decrease of the maximum execution time from the brute-force attack T_{MA} due to fixed or matching values within the keys. Brands W, Y, and Z have at least 7 coincidences, and the brand X has at least 9. These coincidences may increase depending on the model of the vehicle; therefore, it can be up to 13 known or fixed bits within a key. In Table 10, only fixed bits by columns for the different brands are shown.

Table 10. Brute-force attack time reduction analysis for brands W, X, Y, and Z

# FIXED BITS	*# POSSIBLE KEYS*	T_{MA} *(DAYS-HOURS)*	*BRAND W*	*BRAND X*	*BRAND Y*	*BRAND Z*
2	2^{23}	51 – 1224.17				
3	2^{22}	25.5 – 612.08				
4	2^{21}	12.75 – 306.04				
5	2^{20}	6.37 – 153.02				
6	2^{19}	3.18 – 76.51				
7	2^{18}	1.59 – 38.25	•		•	•
8	2^{17}	0.79 – 19.12				
9	2^{16}	0.39 – 9.56		•		
10	2^{15}	0.19 – 4.78				
11	2^{14}	0.09 – 2.39				
12	2^{13}	0.04 -1.19				
13	2^{12}	0.02 – 0.59				

5 CONCLUSIONS

From this research project, it is concluded that the security systems implemented in a large part of the vehicles in Bogotá are susceptible to successful attacks using the proposed methods. It is evidenced that vehicles sold in recent years have vulnerable security systems and, in many cases, with weak technologies. This may come as a result of the final cost of the vehicle, which is affected by the implementation of a more modern security system, apart from other accessories; this fact may influence purchase decisions. Table 3 shows that even vehicles sold in recent years -such as 2018 or 2019- have the most vulnerable security systems (although there are exceptions). This situation could be more alarming if we consider that the useful life of a vehicle in Colombia can be longer than 20 years, unlike other countries, where a vehicle fleet is more frequently renewed. The latter implies that the vehicle may be vulnerable to theft of belongings left inside it when it is no longer attractive to be completely stolen. On the other hand, it is concluded that, from the analysis, the brand

and model of the most vulnerable RKE systems allow to carry out brute-force attacks in just few hours, only in case there is no possibility to carry out a replay attack first. Furthermore, such attack can be used in a place with a high density of vehicles and when executed, one or more vehicles can be opened. It is worth mentioning that in Bogotá, the model of a vehicle can be deduced from its license plate.

Regarding replay attacks, it can be concluded that they are not 100% effective, because, in the analysed sample, 4 vehicles have PKES systems that do not allow this type of attack. However, in the reviewed literature, vulnerabilities in relation to other types of attacks are evident. To execute a replay attack in its basic form, it is necessary for the user to press the unblocking button, so the signal can be stored and later retransmitted. Concerning the latter, an important conclusion is that the unlocking code can be generated from the locking one. In the tests performed on the 22 vehicles to which replay attacks could be applied unconditionally, it became clear that the locking code only differs from the unlocking one in the last three bits. Therefore, by using the signal generation method discussed in section 3.2.2, it is possible to generate the unlocking signal from the locking one. On the other hand, for the replay attack, there are 17 vehicles that use the "rolling code" technique and have the condition that the signal must be stored away from the vehicle coverage and will work only once. In theory, this would not represent a vulnerability if the owner of the vehicle always carries the keys; however, there can be several situations in which the owner can be separated from them. For example, in many car parks in Bogotá, it is required to leave the car keys there to park more vehicles in available spaces. Also, many others have valet parking service, just to mention some cases.

Although replay and brute-force attacks are widely known, SDR platforms are used to implement them in this article. Additionally, a reverse engineering method on RKE systems with fixed code technology is presented. On the other hand, a discussion regarding more vulnerabilities in vehicles depending on the research location is promoted.

ACKNOWLEDGMENT

This work was developed by the GISSIC research group, as a product derived from the Maxwell hotbed of research funded by the at Universidad Militar Nueva Granada.

REFERENCES

[1] J.-T. A.-M. J. Lozano, “Hurto de automotores [Motor vehicle theft] | Policía Nacional de Colombia.” [Online]. Available: https://www.policia.gov.co/delitos-de-impacto/hurto-de-automotores. [Accessed: 17-Feb-2020].

[2] J. González Penagos, “Robo de carros y motos en Bogotá [Car and motorbike theft in Bogotá] | ELESPECTADOR.COM,” 2019. [Online]. Available: https://www.elespectador.com/noticias/bogota/robo-de-carros-y-motos-un-drama-que-sigue-rodando-articulo-833861. [Accessed: 18-Sep-2019].

[3] S. Drimer and S. J. Murdoch. Keep your enemies close: distance bounding against smartcard relay attacks. In Proceedings of 16th USENIX Security Symposium, Berkeley, CA,USA, 2007. USENIX Association.

[4] G. P. Hancke, K. Mayes, and K. Markantonakis. Confidence in smart token proximity: Relay attacks revisited. Computers & Security, 28(7):615–627, 2009.

[5] C. Tori, Hacking Etico [Ethical Hacking] Por Carlos Tori, 1ra ed. Cesar Cerr. Rosario, Argentina: Tori, Carlos, 2008.

[6] T. Waraksa, K. Fraley, R. Kiefer, D. Douglas, and L. Gilbert. Passive keyless entry system. US patent 4942393, 1990.

[7] Baloch, R., 2017. Ethical hacking and penetration testing guide. 1st ed. Boca Raton, Florida: CRC Press, pp.1,7.

[8] A. Oncins Rodríguez, Seguridad informática [IT security], 3rd ed. Cornellá de Llobregat (Barcelona): ENI, 2013, pp. 34,35, 133, 134.

[9] A. Francillon, B. Danev, and S. Capkun, “Relay Attacks on Passive Keyless Entry and Start Systems in Modern Cars.”

[10] P. Schwabe, “Press to unlock : Analysis , reverse-engineering and implementation of HITAG 2-based Remote Keyless Entry systems.” 2018.

[11] O. A. Ibrahim, A. M. Hussain, G. Oligeri, and R. Di Pietro, “Key is in the Air: Hacking Remote Keyless Entry Systems,” 2019, pp. 125–132.

[12] L. Wouters, E. Marin, T. Ashur, B. Gierlichs, and B. Preneel, “Fast, Furious and Insecure: Passive Keyless Entry and Start Systems in Modern Supercars,” IACR Trans. Cryptogr. Hardw. Embed. Syst., vol. 2019, Issu, no. 3, pp. 66–85, 2019.

[13] F. D. Garcia and D. Oswald, Lock It and Still Lose It-On the (In)Security of Automotive Remote Keyless Entry Systems. 2016.

[14] D. MARCHETTI, “Hacking cars by means of software defined radio,” Jul. 2018.

[15] A. Marklind and S. Marklind, “Kandidatuppsats Relay-attacker,” 2019.

[16] R. Verdult, F. D. Garcia, and J. Balasch, “Gone in 360 seconds: Hijacking with hitag2,” Proc. 21st USENIX Secur. Symp., pp. 237–252, 2012.

[17] M. B. Sruthi, M. Abirami, A. Manikkoth, R. Gandhiraj, and K. P. Soman, “Low cost digital transceiver design for Software Defined Radio using RTL-SDR,” in 2013 International Mutli-Conference on Automation, Computing, Communication, Control and Compressed Sensing (iMac4s), 2013, pp. 852–855.

[18] M. A. Wickert and M. R. Lovejoy, “Hands-on software defined radio experiments with the low-cost RTL-SDR dongle,” 2015 IEEE Signal Process. Signal Process. Educ. Work. SP/SPE 2015, pp. 65–70, 2015.

[19] Y. Bedoya Giraldo, C. F. Salazar Giraldo, J. F. Muñoz Lozano, and I. Mecatrónico, "Implementación, control y monitoreo de un sistema de Seguridad vehicular por redes GSM/GPRS," [Implementation, control, and monitoring of a vehicle security system through GSM/GPRS networks] 2013.

[20] A. y TOMALÁ, "Análisis De Vulnerabilidades Vía Comunicación Inalámbrica En Los Cerebros De Vehículos Marca Bmw Modernos De La Ciudad De Guayaquil Utilizando Dispositivos Electrónicos," [Vulnerability analysis via wireless communication of cores from BMW vehicles in Guayaquil by using electronic devices] J. Vis. Lang. Comput., vol. 11, no. 3, pp. 287–301, 2018.

[21] G.-H. Kim, K.-H. Lee, S.-S. Kim, and J.-M. Kim, "Vehicle Relay Attack Avoidance Methods Using RF Signal Strength," Commun. Netw., vol. 05, no. 03, pp. 573–577, 2013.

[22] T. Yang, L. Kong, W. Xin, J. Hu and Z. Chen, "Resisting relay attacks on vehicular Passive Keyless Entry and start systems," 2012 9th International Conference on Fuzzy Systems and Knowledge Discovery, Sichuan, 2012, pp. 2232-2236.

[23] D. Méndez Ávila, INVESTIGACIÓN Y ELABORACIÓN DE UN INSTRUCTIVO SOBRE LAS HERRAMIENTAS HACKER MÁS UTILIZADAS EN EL ÁMBITO INFORMÁTICO. [Research and manufacture of a user's manual on the most common hacking tools in information technology] Ecuador, 2011, p. 27.

[24] M. Sliti, W. Abdallah and N. Boudriga, "Jamming Attack Detection in Optical UAV Networks," 2018 20th International Conference on Transparent Optical Networks (ICTON), Bucharest, 2018, pp. 1-5

[25] K. Vachhani and R. A. Mallari, "Experimental study on wide band FM receiver using GNURadio and RTL-SDR," 2015 International Conference on Advances in Computing, Communications and Informatics (ICACCI), Kochi, 2015, pp. 1810-1814

[26] I. Martoyo, P. Setiasabda, H. Y. Kanalebe, H. P. Uranus and M. Pardede, "Software Defined Radio for Education: Spectrum Analyzer, FM Receiver/Transmitter and GSM Sniffer with HackRF One," 2018 2nd Borneo International Conference on Applied Mathematics and Engineering (BICAME), Balikpapan, Indonesia, 2018, pp. 188-192.

BIOGRAPHY OF AUTHORS

Juan Carlos Martínez-Quintero received the M.Sc. degree in Autonomous Systems of Production from Universidad Tecnológica de Pereira in 2013. He is currently a professor at Universidad Militar Nueva Granada. His research interests include Mobile Networks, SDR, Communication Systems, and digital signal processing.

Edith Paola Estupiñán-Cuesta received the M.Sc. Degree in Electronic Engineering from Universidad Pontificia Javeriana in 2013. She is currently working at Universidad Militar Nueva Granada. Her research interests include Mobile Networks, Traffic analysis, and Data and Management Networks.

Angie Tatiana Choachi-Sarmiento is a Telecommunications Engineering Student from Universidad Militar Nueva Granada. During her major, she joined the GISSIC research group (Maxwell hotbed of research), which has led her to discover how much she can learn from research activities and discussions.