Key Is In The Air: Hacking Remote Keyless Entry Systems

Omar Adel Ibrahim[1], Ahmed Mohamed Hussain[2], Gabriele Oligeri[1], and Roberto Di Pietro[1]

[1] College of Science and Engineering, Hamad Bin Khalifa University, Doha - Qatar
[2] Electrical Engineering Department, Qatar University, Doha - Qatar

Abstract. A Remote Keyless Systems (RKS) is an electronic lock that controls access to a building or a vehicle without using a traditional mechanical key. Although RKS have become more and more robust over time, in this paper we show that specifically designed attack strategies are still effective against them. In particular, we show how RKS can be exploited to efficiently hijack cars' locks.
Our new attack strategy—inspired to a previously introduced strategy named *jam-listen-replay*—only requires a jammer and a signal logger. We prove the effectiveness of our attack against six different car models. The attack is successful in all of the tested cases, and for a wide range of system parameters. We further compare our solution against state of the art attacks, showing that the discovered vulnerabilities enhance over past attacks, and conclude that RKS solutions cannot be considered secure—hence calling for further research on the topic.

1 Introduction

Remote Keyless Systems (RKS) are a critical component of modern car security. Such systems allow the user to lock/unlock the car without resorting to any mechanical key but only by clicking a button on the car's fob or even by getting close to the car itself. RKS mainly implements a request-response protocol between the fob and the car's radio transceiver with minimal security protection [3]. During the years, several security flaws have been identified and RKS evolved mitigating such attacks. An interesting example is the so-called *rolling codes* that prevent an eavesdropper to reuse a code sequence from the past. At each transmission, a new code is generated invalidating the old one by resorting to hash function computations. Unfortunately, rolling codes do not protect against either *proxy attacks* or *jam-listen-replay* attacks [11]. The first class of attacks involve to proxy the code sequence from a further distance to the car without the user consent. This is a classical attack that is played as follows: a user, leaving the fob unattended, allows an adversary to activate the fob (without stealing

it, just pressing the button) and to proxy the fob emitted code sequence to the car leveraging another radio technology such as WiFi, Bluetooth or GSM. Proxy attacks can be mitigated using distance bounding and proximity solutions [12]. Nevertheless, jam-listen-replay attacks are still an open issue due to the difficulty of mitigating jamming attacks. Indeed, the adversary prevents the reception of the code sequence by jamming the car radio transceiver, and at the same time, he logs it for the future hijacking of the car.

Contribution. This paper pushes further the analysis of the jam-listen-replay attack proposed in [11]. We propose an improved attack scenario by exploiting cheap hardware and commonly available Linux tools. We show the results of a real measurement campaign highlighting the effectiveness of the proposed attacks and comparing it against the ones introduced by [11]. We observe how, given the current state of the art, these types of attacks cannot be solved without resorting to novel authentication mechanisms, hence justifying further research efforts by both industry an academia on this topic.

Roadmap. Next section reviews RKS security state of the art. Section 3 details the attack scenario; Section 4 introduces the adopted equipment and its configuration, while Section 5 reports on our measurement campaign and discuss the differences of our attack with respect to the state-of-the-art. Finally, some concluding remarks are presented in Section 6.

2 Related work

A major family of attacks exploits jamming and two subsequent phases: preventing the delivery of the message to the car (by jamming) and recording the transmitted message for the subsequent re-transmission. An early contribution has been provided by [11]. Authors firstly propose an efficient brute-force technique for hacking garage doors remote controllers. Secondly, they introduce RollJam, a combined jamming and radio-recording technique enabling the adversary to hack the communications between the car and its associated fob. RollJam involves very cheap devices such as Teensy 3.1 and two CC1101 transceivers. RollJam works by preventing one or more messages to be delivered to the car from the fob while recording them. Eventually, RollJam allows the user to get in the car but a sequence of valid messages have been stolen and they can be reused later on for opening the car.

Authors in [5], and subsequently in [4], revised the jamming-based attack considering pulse electromagnetic interference despite of continuous interference. They analyzed the effects of pulsed interference on envelope detectors through both simulations and measurements. They also suggested an improved receiver design based on synchronous transmitter-receiver communications, which turn out to be more robust against pulsed interference.

Authors in [10] demonstrated the relay attack on Passive Key-less Entry Systems (PKES) used in modern cars. They set up two low-cost and powerful attack scenarios, using wireless and wired physical layer relays enabling the adversary

to open the car and start the engine by relaying the messages between the key and the car.

A general overview describing several techniques of potential attacks against passive entry systems is introduced in [3]. Authors proposed a solution to protect the vehicle from such attacks by exploiting the difference in power levels of the received bits.

3 Scenario

Our attack scenario involves three entities: the *car*, the car's owner (*user*) and the *adversary* who wants to steal the user's car. The adversary implements his strategy in 3 subsequent steps as depicted in Fig. 1: (i) *set-up*, (ii) *jamming and recording*, and (iii) *hijacking*.

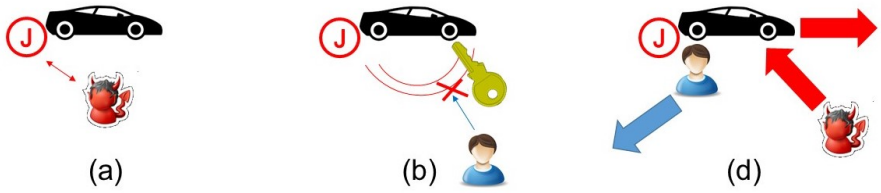

Fig. 1. The attack is performed in a sequence of 3 steps: (a) Jammer set-up and activation, (b) Jamming the communication between the user and the car and forcing the user to use the mechanical key, and finally (c) when the user leaves the car unattended, the adversary hijacks the car.

Set-up. This is a preliminary phase that is performed by the adversary when the car is left unattended by the user. Indeed, the adversary has to install a jammer on the car. As it will be clear in the following, the jammer is a very portable device mainly constituted by a Raspberry Pi v3 (RPiv3) connected to a HackRF One, a very cheap and ready to be deployed Software Defined Radio (SDR). The overall equipment can be hidden in several places outside of the car, e.g., by using a magnet under the car platform.

Jamming and recording. The equipment should be activated after its installation and it will prevent the communication between the fob and the car by jamming a specific frequency. Since the user will not be able to open the car by using the fob, after several attempts, he will resort to the mechanical key. Conversely, the adversary will record one or more code sequences transmitted by the fob (and never received by the car) by eavesdropping the fob-car communication channel.

Hijacking. The car's owner will eventually drive the car away and close it still using the mechanical key. We recall that a jammer is installed on the car preventing the fob to control the lock mechanism of the car. Subsequently, the

adversary will perform his attack by replaying one of the previously recorded code sequences, and allow him to hijack the car.

The only unknown parameter to the previous procedure is the communication frequency adopted by the car brand. The adversary can easily discriminate it by running a discovering session sensing fractions of the radio spectrum. Our experiments show that the majority of the cars we used adopts a frequency band close to 433MHz.

4 Equipment: Hardware, Software and Set-up configuration

Our system consists of 2 components: the **Jammer** and the code sequence **Logger**.

4.1 Jammer

We implemented a mobile jammer by connecting a Raspberry Pi v3 to a HackRF One and a power bank as depicted in Fig.2.

HackRF One: HackRF One is an open source, half-duplex Software Defined Radio device developed by Great Scott Gadgets and has the capability to receive or transmit radio signals starting from 1 MHz to 6 GHz.

ANT500 Antenna: ANT500 is a general purpose, telescopic antenna developed by Great Scott Gadgets and is designed to operate in the range from 75 MHz up to 1 GHz. Its length is configurable starting from 20 cm up to 88 cm.

Raspberry Pi v3: We installed GNU Radio on the RPiv3 and exploited the Python SDK to control the Hack RF One. The result is a script to transmit white Gaussian noise on a target frequency.

Power-bank: We adopted a generic power bank of 5000mA guaranteeing a long lasting life to our system (about half a day).

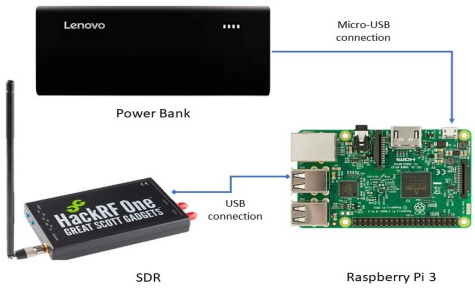

Fig. 2. The Jammer: An RPiv3 controls the HackRF One transmitting white Gaussian noise at the frequency of 434MHz. The power bank guarantees half a day of jamming activity.

Finally, we exploited the embedded WiFi in the RPiv3 to access it through SSH, changing the various jamming parameters and switching it on and off. We observe that the jamming frequency (433MHz) is far away from that one used by the WiFi (2.4GHz), and therefore, the jammer can be remotely controlled. We set all the gains for the HackRF One platform to 40dB, i.e., radio band (RF), intermediate band (IF) and base band (BB) gain. Finally, we set the sampling rate (sps) to 2M as an empirical trade-off to jam the fob-car communication without disturbing any other communications in the neighborhood.

4.2 Logger

The logger is mainly constituted by a mobile platform able to log the code sequence transmitted by the fob to the car. We adopted the following set-up:
Laptop: We configured a laptop with a Linux Ubuntu distribution and GNU Radio Companion.
HackRF One and ANT500 Antenna: A HackRF One has been connected to the above laptop to record all the code signals transmitted in the neighborhood. Figure 3 resumes our logger setup and the main connections.

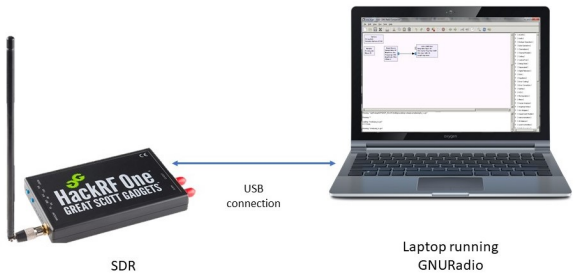

Fig. 3. The Logger: a laptop equipped with Ubuntu and GNURadio Companion is used to receive and log the code sequence transmitted by the fob.

We considered the following configuration for the SDR: frequency 434MHz, sampling rate 2M (sps), RF Gain 10dB, IF Gain 20 dB, and BB Gain 20 dB. We observe that the gains figures adopted by the logger are significantly different from that one used by the jammer. Indeed, the logger has to mitigate the noise power from the jammer in order to decode the code sequence transmitted by the fob. The above values are the result of several trials and they take also into account the relative distances between the jammer, the fob and the logger.

5 Measurement results

We performed several measurements in the parking of our university (College of Science and Engineering - Hamad Bin Khalifa University, Doha, Qatar) during

the week-end when the parking was empty, not to interfere with other users. The first step of our attack consists in identifying the communication frequency used by the fob-car communication link. Although different car brands might use different frequencies, there are mainly two different frequencies adopted worldwide [2]: 315MHz for North America for 433.92 MHz for Europe and Asia. Therefore, an adversary can easily detect which frequency band is used in a couple of consecutive trials. Other unknown parameters such as the modulation scheme (ASK, FSK, PSK) can be easily detected as well by using simple Linux tools such as gqrx [1].

We tested the attack on six different cars: Škoda Yeti (2016), Škoda Octavia (2009), Mazda 6 (2009), Toyota Rav4 (2014), Mitsubishi Pajero (2015) and Nissan Sunny (2014). Another minor challenge introduced by our attack is to find the most effective position for the jammer in the car. Of course, the best position is close to the car signal receiver, which unfortunately is unknown to the adversary. We tried several positions all around the target car taking into account that the jammer should remain hidden to the user and an optimal position turned out to be in the back of the car (for all the car models).

Our measurement scenario is constituted by the target car, the user with the fob and the logger displaced as in Fig. 4.

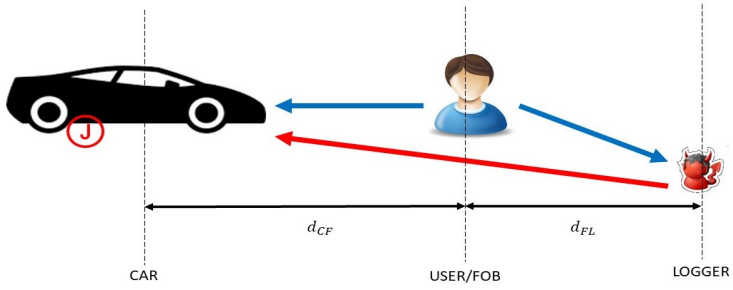

Fig. 4. Measurement scenario: the target car d_{CF} meters to the user (fob), which in turn is d_{FR} meters far away from the logger.

5.1 Results and discussion

We considered six different displacements, i.e., $d_{CF} \in \{5, 10\}$ and $d_{FL} \in \{1, 2, 3, 4\}$, while each configuration has been run 20 times as depicted in Table 1. Firstly, we observe that the chances of the attack being successful get reduced when the distance between the logger and the fob (d_{FL}) increases. Moreover, we highlight that the presence of the jammer itself partially prevents the logger to eavesdrop the code sequence. Indeed, this is proved by the fact that when the distance between the jammer/car and the fob (d_{CF}) gets larger, the logger can record a

good code sequence at 3 meters from the fob (that distance is reduced to 50cm when the fob is 5m away from the jammer).

Table 1. Measurement results.

d_{CF} (m)	d_{FL} (m)	Successful attack frequency
5	1	1
5	2	0.4
5	3	0.05
10	2	1
10	3	1
10	4	0.1

Comparison with [11] The attack proposed in [11] exploits the combination of both a logger and a jammer as well. Nevertheless, there are significant differences that make our attack scenario even more dreadful and effective. Firstly, the attack proposed in [11] involves a precisely tuned jammer in order to prevent the self-jamming phenomena, i.e., the jammer on the car prevents the logger to collect a clean code sequence. Our measurements show that the self-jamming phenomena does not happen if the logger is close to the fob (1 meter); moreover, if the distance between the user and the car is about 10m, the adversary has a wider range of action (up to 3m) having more chances to be hidden to the user. Secondly, our attack is much more flexible. Adopting a fully portable, and remotely controllable jammer, allows us to install the jammer on the car, preventing all the fob-car communications and intrinsically, having more chances to collect and log more code sequences. Finally, there is another significant difference with [11]: Kamkar et al. propose to collect two rolling codes: one for opening the car, while the second one for the future hijacking of the car. This approach might eventually turn out to be very difficult to implement. Indeed, every time the user clicks on the fob's button invalidates the previous codes (assuming the deployed RKS adopts the rolling code strategy). The user might keep trying to open and close the car, even after the car has been opened by the code sequence sent by the adversary. Unfortunately, this makes the solution proposed in [11] very dependent of the user's behaviour and consequently, the attack has to be strictly supervised by the adversary. Conversely, our attack scenario is more effective, since it prevents all the communications between the fob and the car, forcing the user to eventually use the mechanical key to enter the car.

Discussion. The proposed attack is very difficult to mitigate. An early strategy has been proposed in [11] involving a jammer detector. Although this strategy might detect an undergoing attack and raise an alarm to the user, it cannot be used to improve the robustness of the keyless communication system. This is mainly due to the intrinsic difficulty of dealing with jamming mitigation [6–9]. Moreover, depending on how "smart" is the car, the jammer might prevent other

on-board communications such as the authentication of the key itself, and therefore, preventing the engine to switch on.

6 Conclusion

We have proposed a novel scenario attack for remote keyless entry systems involving a new jamming strategy and a remotely controlled signal recorder. We tested the attack against six different car models considering different deployment strategies. The cheap HW employed, the easiness of attack deployment, and its effectiveness—always successful, even for a wide range of system parameters—show that RKS are still not secure and that further research by both industry and academia is needed.

References

1. Gqrx sdr, `http://gqrx.dk`, last accessed on 26-06-2018
2. Remote keyless systems, `https://en.wikipedia.org/wiki/Remote_keyless_system`, last accessed on 26-06-2018
3. Alrabady, A.I., Mahmud, S.M.: Some attacks against vehicles' passive entry security systems and their solutions. IEEE Transactions on Vehicular Technology 52(2), 431–439 (March 2003)
4. van de Beek, S., Leferink, F.: Vulnerability of remote keyless-entry systems against pulsed electromagnetic interference and possible improvements. IEEE Transactions on Electromagnetic Compatibility 58(4), 1259–1265 (Aug 2016)
5. van de Beek, S., Vogt-Ardatjew, R., Leferink, F.: Robustness of remote keyless entry systems to intentional electromagnetic interference. In: 2014 International Symposium on Electromagnetic Compatibility. pp. 1242–1245 (Sept 2014)
6. Di Pietro, R., Oligeri, G.: Jamming mitigation in cognitive radio networks. IEEE Network 27(3), 10–15 (May 2013)
7. Di Pietro, R., Oligeri, G.: Freedom of speech: Thwarting jammers via a probabilistic approach. In: Proceedings of the 8th ACM Conference on Security & Privacy in Wireless and Mobile Networks. pp. 4:1–4:6. WiSec '15, ACM, New York, NY, USA (2015)
8. Di Pietro, R., Oligeri, G.: Silence is Golden: Exploiting jamming and radio silence to communicate. ACM Trans. Inf. Syst. Secur. 17(3), 9:1–9:24 (Mar 2015)
9. Di Pietro, R., Oligeri, G.: Enabling broadcast communications in presence of jamming via probabilistic pairing. Computer Networks 116, 33 – 46 (2017)
10. Francillon, A., Danev, B., Capkun, S.: Relay attacks on passive keyless entry and start systems in modern cars. In: Proceedings of the Network and Distributed System Security Symposium (NDSS). Eidgenössische Technische Hochschule Zürich, Department of Computer Science (2011)
11. Kamkar, S.: Drive it like you hacked it: New attacks and tools to wirelessly steal cars. In: DEFCON 23 (2015)
12. Wang, X., Hou, X., Rios, R., Hallgren, P., Tippenhauer, N.O., Ochoa, M.: Location proximity attacks against mobile targets: Analytical bounds and attacker strategies. In: Proceedings of the European Symposium on Research in Computer Security (ESORICS) (September 2018)