CSE 4820 - Hacking the Mini Cooper - Final Project

Gabriel Andrade Silva, Cody Manning, Curtice Gough
Dept. of Electrical Engineering and Computer Science
Florida Institute of Technology
Melbourne, FL, USA
jsilva2021@my.fit.edu, cmanning2020@my.fit.edu, cgough2020@my.fit.edu

***Abstract*—This research explores the vulnerabilities of keyless entry systems in automobiles, specifically focusing on the 2016 Mini Cooper. Utilizing a HackRF Software-Defined Radio (SDR), we demonstrate two types of attacks: a relay attack and a Roll-Jam attack. Our findings highlight significant security flaws in keyless entry systems, emphasizing the need for improved security measures in automobile design. The study aims to raise awareness about these vulnerabilities and encourage further research into more secure alternatives for keyless entry systems. Please note that this research is intended for educational purposes and we strongly discourage the misuse of this information for illegal activities.**

I. Introduction

The project began almost as a joke, hacking a person's car seemed like something someone would see in a movie. One of the team members commented on hacking a car, but not an RC car; a real life one. Out of the three of us, only one owned a car that could be used for testing. The joke quickly became a real thought as we begun research on the viability of the project. It turns out that with the use of a Hack RF SDR (Software-defined radio) we could tackle the concept of unlocking the car without the use of the key fob[1]. Throughout the course of the project, we failed to make the replay attack work; because of the security protocols in place by BMW on the Mini Cooper's key fob. We were not dissuaded however; as we saw another avenue of attack in the blog we found before, the roll-jam attack[2]. Unfortunately, the roll-jam attacked succeeded, but not in the way we were hoping. We successfully jammed the fob to the Mini Cooper, but were unsuccessful in opening the car. Where we succeeded however; was desyncing the car fob from the car permanently. While we may not have succeeded in opening the car remotely, there are many instances of this approach working (without resulting in ruining the car fob).

II. Proposed Approach

The replay attack was our first take on hacking the mini cooper. The team all had a little bit of experience with the Hack RF tool, and they also had access to the IoT lab. The Hack RF was used to capture the signal from the Mini Cooper's fob, while we were out of range of actually unlocking the mini cooper. We then used a software tool called Universal Radio Hacker[3]. This tool allowed us to investigate the signals we were receiving, because the physical Hack RF was only useful for receiving (and sending) the key fob signals. This tool was built in Python and was our main way of communicating with the signals we received during testing.

The first step to doing any of this however, was finding information on the key fob itself. This information can be easily found by going to the website for the Federal Communications Commission (FCC) and inputting the FCC ID found on the fob itself. This FCC ID is legally required for all wireless devices in the United States. The FCC ID on our key fob was NBGIDGNG1[4]. Using the FCC ID and the lookup website, we were able to find the frequency that the key fob operated on, 433.2-434.64 MHz. This range is important for later. To begin, we captured the signal using the Hack RF, we picked it up on Universal Radio Hacker (See figure 1 below)

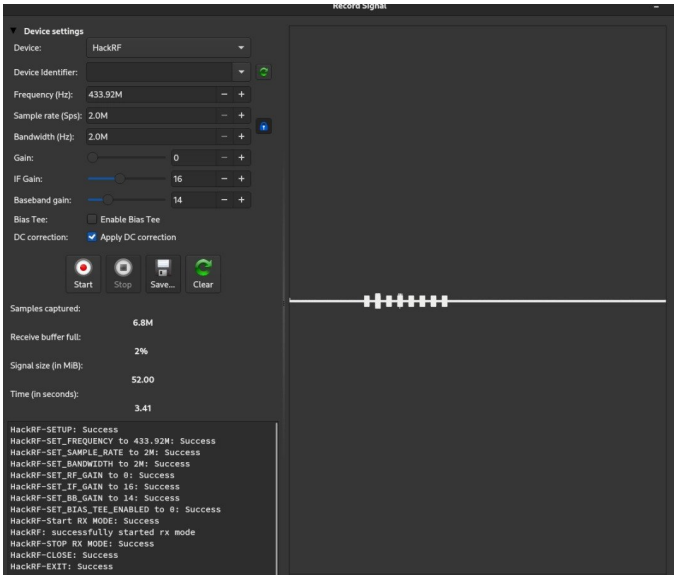

Fig. 1. Universal Radio Hacker picking up the signal from our key fob on the frequency range we found on the FCC lookup.

We took the Hack RF with the key fob signal loaded and went outside to the mini cooper. When we tried replaying the signal, we got no response. This was an unfortunate setback. We decided to troubleshoot to see if there was some kind of error we could have missed. Upon further inspection (which was us capturing the signal again multiple times), we noticed

that the frequency had a rolling code. This meant that it randomly switched frequencies (in the established range) and thus made the replay attack impossible. This was discouraging, but we decided to continue on with another plan of attack, the roll-jam attack.

A roll-jam attack is kind of similar to the replay attack, in that we capture a signal and try sending it back. The execution is pretty similar, but it is very different in practice. A roll-jam attack works by basically jamming the radio signal from the key fob. Because the signal is jammed, the person with the fob will try to press the unlock button again. After this, we have captured both signals from the key fob, and both have been jammed from the car. We then replay the first signal to unlock the car. Unbeknownst to the victim, the person who received the jammed signals has the next signal in the sequence (which has not expired yet). They can use this second signal to open the car whenever they like. An abbreviated version can be seen in the figure below.

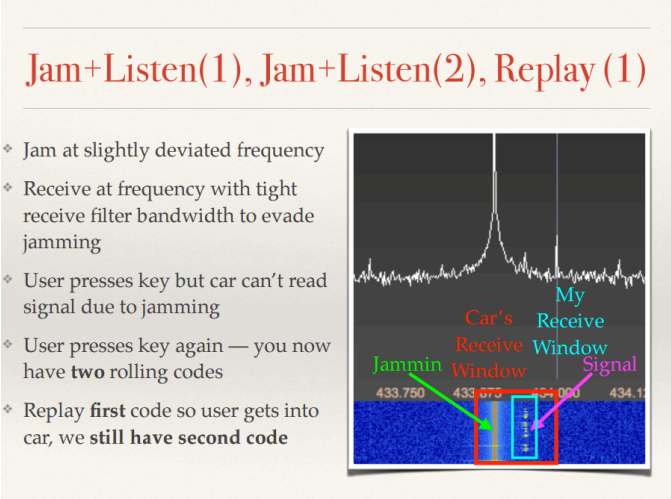

Fig. 2. An easy to understand graphic on how the roll-jam attack works.

In order to carry out the roll-jam attack, we needed more than the hack RF, we used what is known as a ”portapack”. A portapack is basically a Hack RF with a dedicated screen, and some software loaded on it. We used three portapacks, two to jam and one to send the signal we received. We set up everything, and properly jammed (both) frequencies. While we were doing this, we properly captured the signals. Upon cutting the jamming and sending the captured signal however, we got no response from the car. Even worse, the result of this ended up de-synchronizing the car fob, removing it's ability to remotely open the car (Opening the car with the key still works, but the fob itself cannot wirelessly open the car). This mean that a new fob will need to be purchased, or we will need to pay to have it reprogrammed. This is quite unfortunate, because we not only failed to remotely open the car, we also bricked the only key fob we had.

There are several possible factors which may have contributed to the ultimate failure of our roll-jam attack. All we have are educated guesses, but there is one hypothesis that seems more plausible than the others. Given the fact that the key fob operates on two separate frequencies, it is possible that one frequency could be used for transmitting the rolling authentication codes while the other frequency is used for transmitting the actual commands. Operating under this assumption, our hypothesis is that our replay attack successfully transmitted the rolling authentication code, but not the ”unlock” command. We only used one Portapack for the replay while jamming both frequencies. The frequency we chose to replay (433.20 Mhz) may have been the rolling code frequency. Since our replay caused the car to update its code, the car and the fob ended up de-synchronized.

One more interesting note is the existence of another form of attack, the RollBack attack. This is a derivative of the roll-jam attack that works by replaying a handful of previously captured signals to 'rollback' the cars counter. Even if these signals were already received by the car, they would still unlock the car. This exploit was first mentioned in a BlackHat 2022 presentation[5]. See the figure below for a quick and easy comparison of the rollback exploit.

Fig. 3. Three key take ways for the rollback attack, compared to the rolljam attack.

III. Conclusion

While we may have not have obtained the results we wanted, we did learn a lot about how the replay and roll-jam attacks worked on a fundamental level. We did a lot of research into various successful attempts at unlocking vehicles with this technology. Just because we were unable to open the car remotely doesn't necessarily mean that the project was a failure. Some examples of the same principal being applied to this concept are YouTube user TAKEAPART's video on the roll-jam attack [6] or James Chambers blogpost on the replay attack[1]. Both of these are examples of successful attacks that we failed to implement. An important note is that we may have been unable to hack the mini cooper successfully because of user error, or due to the fact that BMW (the manufacturer of the mini cooper) implemented security measures against these attacks. There is still a lot to do to ensure the security of key fobs, as anyone with a hack RF has the capability to jam and potentially unlock a car. Although we unfortunately broke our

key fob during the execution of this project, we did learn a lot. After all, Sacrifices must be made in the pursuit of science.

REFERENCES

[1] J. A. Chambers. (2023) Use hackrf sdr to lock / unlock car. https://jamesachambers.com/use-hackrf-sdr-to-lock-unlock-car/. [Online; accessed 3-March-2023].

[2] C. Coward. (2023) Hacking a car's key fob with a rolljam attack. [Online]. Available: www.hackster.io/news/hacking-a-car-s-key-fob-with-a-rolljam-attack-7f863c10c8da

[3] J. Pohl and A. Noack, "Universal radio hacker: A suite for analyzing and attacking stateful wireless protocols," in *12th USENIX Workshop on Offensive Technologies (WOOT 18)*. Baltimore, MD: USENIX Association, 2018. [Online]. Available: https://www.usenix.org/conference/woot18/presentation/pohl

[4] (2023) FCC ID NBGIDGNG1. [Online]. Available: https://fccid.io/NBGIDGNG1

[5] (2022) Rollback: A new time-agnostic replay attack. Black Hat USA. [Online]. Available: https://i.blackhat.com/USA-22/Thursday/US-22-Csikor-RollBack-A-New-Time-Agnostic-Replay-Attack.pdf

[6] TAKEAPART. (2023) Rolljam attack flipper zero & hackrf car unlock. YouTube. [Online]. Available: https://www.youtube.com/watch?v=YVYGeywlelU