Eavesdropping and Emulating MIFARE Ultralight and Classic Cards Using Software Defined Radio

Ilias Giechaskiel[1]

Abstract

In this report, we describe a Software-Defined Radio (SDR) approach for eavesdropping on Near Field Communications (NFC) and Radio Frequency Identification (RFID) cards operating at 13.56MHz. We show that GNU Radio and Python make a great platform for prototyping, while maintaining sufficient performance for passive attacks without extensive optimizations and using only modest processing power. We successfully eavesdrop on real MIFARE Ultralight and Classic 1K cards by capturing the raw radio waves with a home-made antenna. We recover the plaintext of both reader and tag fully by demodulating the incoming radio waves, parsing individual bits and error detection codes into packets, and then decrypting them if necessary. On the transmission side, we achieve full software emulation of the reader and of MIFARE Ultralight and Classic 1K cards (including encryption), and partial hardware emulation, where we correctly modulate the signal, but not within the strict timing limits of the protocol. Our transmissions can also be used to prevent legitimate communication by interfering with the intended reader or tag signals.

[1] *CDT in Cyber Security, University of Oxford, Oxford, United Kingdom*

1. Introduction

Contactless cards and tags have become very popular in recent years, with everyday applications including e-passports [21], ticketing [22, 5, 7], access control [23], and payment [12, 6] systems. However, as these devices operate wirelessly, adversaries can pick up the radio signals and eavesdrop on the communication between a tag and a reader. Traditionally, such attacks on radio communications required dedicated hardware for particular frequencies and modulation types, but with the advent of Software-Defined Radio (SDR), it is possible to use generic equipment and perform the demodulation in software. Even so, despite a range of embedded devices and Field-Programmable Gate Arrays (FPGAs) that are capable of various attacks on Near Field Communication (NFC), Radio Frequency Identification (RFID), and related technologies, to the best of our knowledge no open-source SDR implementation exists for High-Frequency (HF) NFC.[1]

To this end, we developed such an implementation on an Ettus Research Universal Software Radio Peripheral (USRP) using Python and GNU Radio with an antenna made out of simple wires that allows passive eavesdropping on reader-tag communication. Though our implementation is easily extensible, we focused on MIFARE cards by NXP Semiconductors, since MIFARE has "a market share of more than 77% in the transport ticketing industry", with "150 million reader and 10 billion contactless and dual interface IC's sold" [15]. Specifically, we use Ultralight [17] and Classic 1K [16] cards, as the former does not employ any encryption, while the latter uses a broken cryptographic algorithm (Section **??**), making them ideal candidates for such exploration. Moreover, we achieve full software and partial hardware reader and tag emulation, that can also be used to jam signals between a legitimate tag and reader. In summary, our contributions are as follows:

1. We implement in pure Software-Defined Radio a demodulator for NFC/RFID readers and tags operating in the 13.56MHz frequency, which decodes radio waves into plaintext packets
2. We test our implementation by eavesdropping on real MIFARE Classic 1K and Ultralight communications with an RFID reader using a home-made antenna and a USRP, successfully decoding any encrypted packets
3. We additionally implement in software the emulation of both readers and tags, including encryption if necessary
4. Though our transmission capabilities are hindered by the strict timing requirements of the protocol, we show how our implementation can jam real reader-tag communications and prevent successful transmission of data
5. Overall, our work shows that prototyping using Software-Defined Radio is sufficient in practice for passive attacks, without the need for extensive optimizations or heavy computing power

2. Related Work

Early work on RFID Hacking was conducted in a non-academic context, and focused on finding vulnerabilities in access control systems [19, 22]. Later, Buettner and Wetherall [3, 4, 2] experimented more systematically with RFID and SDR, but focused primarily on Gen 2 cards operating at 900MHz. Their work was extended by others, typically in the context of proposing better protocols [1, 25], but still for Ultra High

[1] Though they exist for UHF Gen2 cards. See `https://github.com/brunoprog64/rfid-gen2` and `https://github.com/yqzheng/usrp2reader` for instance.

Frequencies (UHF), with the exception of a recent work by Hassanieh et al. [9], which also included an extension to HF.

There have also been a number of designs use microcontrollers and Field-Programmable Gate Arrays (FPGAs) for signal processing, such as the Proxmark 3 [24], RFIDler [11]. Though such projects allow the use of custom firmware for additional functionality, they also require dedicated hardware in their design.

The MIFARE Classic cryptographic protocol was reverse-engineered by Nohl et al. by dissolving the plastic surrounding the chips, recovering the individual logic gates and converting them to a high-level algorithm [14]. Garcia et al. then discovered additional vulnerabilities of the protocol based on its nested authentication and parity bits [7]. Due to the wide range of applications of the MIFARE Classic, the topic became very popular for Master's thesis projects [20, 5, 18, 23], which found additional vulnerabilities, or examined the problem within the context of a specific application.

Finally, given the widespread availability of NFC-enabled mobile devices, researchers have also focused on NFC relay attacks using mobile phones [12, 6], as well as exploring [13] and protecting [8] the NFC mobile phone stack.

Acknowledgments

I would like to thank Kasper Rasmussen for being my supervisor, for his invaluable help throughout the project, and for entrusting me with his expensive equipment. I would also like to thank Simon Crowe for helping shape the focus of the project. Finally, I would like to thank Kellogg College and their Research Support Grant which enabled the purchase of some of the components that were used in this project.

References

[1] BRIAND, A., ALBERT, B., AND GURJAO, E. Complete software defined RFID system using GNU radio. In *IEEE International Conference on RFID-Technologies and Applications (RFID-TA 2012)*.

[2] BUETTNER, M., AND WETHERALL, D. A software radio-based UHF RFID reader for PHY/MAC experimentation. In *IEEE International Conference on RFID (RFID 2011)*.

[3] BUETTNER, M., AND WETHERALL, D. A flexible software radio transceiver for UHF RFID experimentation. Tech. Rep. UW-CSE-09-10-02, University of Washington - Computer Science Department, 2009.

[4] BUETTNER, M., AND WETHERALL, D. A "Gen 2" RFID monitor based on the USRP. *SIGCOMM Comput. Commun. Rev.* (2010).

[5] DE KONING GANS, G. Analysis of the MIFARE classic used in the OV-Chipkaart project. Master's thesis, Radboud University Nijmegen, 2008.

[6] FRANCIS, L., HANCKE, G., MAYES, K., AND MARKANTONAKIS, K. *Practical Relay Attack on Contactless Transactions by Using NFC Mobile Phones*.

[7] GARCIA, F. D., ROSSUM, P. V., VERDULT, R., AND SCHREUR, R. W. Wirelessly pickpocketing a Mifare Classic card. In *Proceedings of the 30th IEEE Symposium on Security and Privacy (SSP 2009)*.

[8] GUMMESON, J. J., PRIYANTHA, B., GANESAN, D., THRASHER, D., AND ZHANG, P. Engarde: Protecting the mobile phone from malicious nfc interactions. In *Proceeding of the 11th Annual International Conference on Mobile Systems, Applications, and Services (MobiSys 2013)*.

[9] HASSANIEH, H., WANG, J., KATABI, D., AND KOHNO, T. Securing RFIDs by randomizing the modulation and channel. In *12th USENIX Symposium on Networked Systems Design and Implementation (NSDI 2015)*.

[10] KASPER, T., VON MAURICH, I., OSWALD, D., AND PAAR, C. Chameleon: A versatile emulator for contactless smartcards. In *Information Security and Cryptology (ICISC 2010)*.

[11] LAURIE, A. RFIDler. `https://github.com/ApertureLabsLtd/RFIDler`. Accessed: 2015-06-19.

[12] MICHAEL, W. Performing relay attacks on ISO 14443 contactless smart cards using NFC mobile equipment. Master's thesis, Der Technischen Universität München, 2010.

[13] MILLER, C. Exploring the NFC attack surface. Presented at Black Hat USA 2012.

[14] NOHL, K., EVANS, D., STARBUG, S., AND PLÖTZ, H. Reverse-engineering a cryptographic RFID tag. In *Proceedings of the 17th Conference on Security Symposium (USENIX Security 2008)*.

[15] NXP SEMICONDUCTORS. About MIFARE. `https://www.mifare.net/en/about-mifare/`. Accessed: 2015-06-19.

[16] NXP SEMICONDUCTORS. *MIFARE Classic 1K - Mainstream contactless smart card IC for fast and easy solution development*, 2 2011. Rev. 3.1.

[17] NXP SEMICONDUCTORS. *MIFARE Ultralight contactless single-ticket IC*, 7 2014. Rev. 3.9.

[18] PENRI-WILLIAMS, K. E. Implementing an RFID 'Mifare Classic' attack. Master's thesis, City University London, 2009.

[19] PLÖTZ, H. RFID hacking. Presented at the 23rd Chaos Communication Congress (CCC 2006).

[20] PLÖTZ, H. Mifare classic – eine analyse der implementierung. Master's thesis, Humboldt-Universität zu Berlin, 2008.

[21] RICHTER, H., MOSTOWSKI, W., AND POLL, E. Fingerprinting passports. NLUUG spring conference on security (2008).

[22] RYAN, R., ANDERSON, Z., AND CHIESA, A. Anatomy of a subway hack. Presented at the 16th DEFCON Hacking Conference (DEFCON 2008).

[23] TAN, W. H. Practical attacks on the MIFARE Classic. Master's thesis, Imperial College London, 2009.

[24] WESTHUES, J. Proxmark 3. `https://github.com/Proxmark/proxmark3`. Accessed: 2015-06-19.

[25] ZHENG, Y., AND LI, M. ZOE: Fast cardinality estimation for large-scale RFID systems. In *INFOCOM* (2013).