12-4-2013

Verification Of Primitive Sub Ghz Rf Replay Attack Techniques Based On Visual Signal Analysis

Maxim Chernyshev
Edith Cowan University

DOI: 10.4225/75/57b3c136fb869
11th Australian Digital Forensics Conference. Held on the 2nd-4th December, 2013 at Edith Cowan University, Perth, Western Australia

VERIFICATION OF PRIMITIVE SUB-GHZ RF REPLAY ATTACK TECHNIQUES BASED ON VISUAL SIGNAL ANALYSIS

Maxim Chernyshev
Security Research Institute, Edith Cowan University
Perth, Australia
m.chernyshev@ecu.edu.au

Abstract
As the low-cost options for radio traffic capture, analysis and transmission are becoming available, some security researchers have developed open-source tools that potentially make it easier to assess the security of the devices that rely on radio communications without the need for extensive knowledge and understanding of the associated concepts. Recent research in this area suggests that primitive visual analysis techniques may be applied to decode selected radio signals successfully. This study builds upon the previous research in the area of sub-GHz radio communications and aims to outline the associated methodology as well as verify some of the reported techniques for carrying out radio frequency replay attacks using low-cost materials and freely available software.

Keywords
Radio frequency (RF) communications, replay attack methodology, visual signal analysis, protocol analysis, RF signal decoding, RfCat, sub-GHz RF spectrum, software-defined radio (SDR)

INTRODUCTION

Radio traffic is said to be rarely used in penetration testing during the reconnaissance phase, yet one is able to collect potentially valuable information through its capture and analysis (Neely, Hamerstone, & Sanyk, 2013). The modern equipment that is needed to facilitate the protocol analysis, as well as subsequent signal retransmission, does not have to be costly or unwieldy.

With the proliferation of software-defined radios (SDRs), which may be purchased for less than $20 USD, interested radio amateurs as well as researchers with limited funding are able to explore the field of radio communications (Smith, 2012). While SDRs are theoretically able to operate on frequencies between 55MHz and 2300MHz, with some specialised devices designed by Ossmann (2013) being able to cover the extended 30MHz-6000MHz range, this study focuses on the frequencies below 1000MHz, or the "sub-GHz" range. The sub-GHz range is of immense interest due to the high number and variety of devices that operate within that range. The aim of the study is to verify primitive visual signal analysis techniques that could potentially be useful in executing replay attacks against devices in the sub-GHz range without the need to acquire detailed knowledge of radio communications or expensive equipment.

DEVICES IN THE SUB-GHZ SPECTRUM

Devices that operate in the sub-GHz range commonly include alarm systems, various meters and sensors (such as proximity, pressure, and acceleration), home automation systems, remote camera flashes and shutter triggers, garage door openers (GDE), remote keyless entry systems (RKE), tyre pressure monitoring systems (TPMS), wireless door chimes, smart-grid technology components for meter to utility communications, cordless phones, mobile phones, medical devices as well as radio-frequency identification (RFID) devices (Atlas, 2012b; Harizanov, 2012; Liang & Liu, 2011; Sikken, 2009; SiliconLabs, 2010; Smith, 2012; Weber, 2013).

According to Liang and Liu (2011) and SiliconLabs (2010), sub-GHz frequencies tend to be the rational choice for such devices because utilising this portion of the spectrum facilitates the following features:

- Range
- Low interference

- Low power consumption

While it is also not unusual for the device designers to turn to the 2.4GHz band to achieve global interoperability through the utilisation of common standards such as Bluetooth, WLAN and ZigBee, devices that require prolonged battery-based operation are often designed to function in the sub-GHz bands.

Table 1. Some Common Regional Sub-Ghz Bands (Harney & O'Mahony, 2006)

Region	Relevant Standards	Frequency Bands (MHz)	Relevant Links
Europe	ERC REC 70-03 EN 300 220 (Sept. '00) EN 300 220 (Feb. '06)	 433.05 to 434.79 868.0 to 870 863.0 to 870	 http://www.ero.dk/ http://www.etsi.org
U.S.	FCC Title 47 Part 15.231 Part 15.247	 260 to 470 902 to 928	http://www.access.gpo.gov/nara/cfr/waisidx_04/47cfr15_04.html
Canada	RSS-210	260 to 470 902 to 928	http://strategis.ic.gc.ca/epic/internet/insmt-gst.nsf/en/sf01320e.html
Japan	ARIB STD-T67	426.0375 to 426.1125 429.175 to 429.7375	http://www.arib.or.jp/english/
China	RADIO REGULATIONS OF THE PEOPLE'S REPUBLIC OF CHINA	315.0 to 316.0 430.0 to 432.0	http://ce.cei.gov.cn/elaw/law/lb93i1e.txt
Australia	AS/NZS 4268:2003	433.05 to 434.79 915 to 928	http://www.acma.gov.au/ACMAINTER.131180

Most of the bands presented in Table 1 are also commonly known as the industrial-scientific-medical (ISM) bands (Frenzel, 2010). Specifically, the 433MHz band is often the popular choice of the global manufacturers, as it only requires a minor frequency adjustment to operate in Japan (Harney & O'Mahony, 2006). It should be noted that in Australia such devices are also referred to as the low interference potential devices (LIPDs), and their operation in the 433.05-434.79MHz band is supported through the relevant class licence as long as the maximum equivalent isotropically radiated power (EIRP) does not exceed 25 milliwatt (mW). Historically, the adoption of the 433MHz band in Australia was driven in the late 1990s by the growing demand for the importation and adoption of the devices from Europe, where, unlike in Australia, the 433MHz band is also the designated ISM band (ACMA, 1999).

ATTACK METHODOLOGY

At the time of writing, a number of articles and blog entries that describe the approaches taken to capture and reverse-engineer radio frequency (RF) signals are available on the Internet (Hohawk, 2012; Laurie, 2013; Weber, 2013). Unfortunately, the researcher has not been successful in locating published peer-reviewed articles specific to this topic in the context of devices commonly found in residential properties, apart from the ones that employ the KeeLoq block cipher (Aerts et al., 2012). Perhaps, this could be due to poor keyword selection, or otherwise goes to accentuate the perceived novelty of using low-cost materials to carry out RF replay attacks on consumer electronics.

In the context of this study, the activities associated with the capture, decoding and retransmission of the RF signals are referred to as "attacks", because successful signal decoding and retransmission are expected to result in taking partial or, in some cases, full control of the target device - contrary to the desire of its owner or the intention of the manufacturer. Based on the currently available reports, it is possible to derive a structured theoretical methodology for carrying out these attacks.

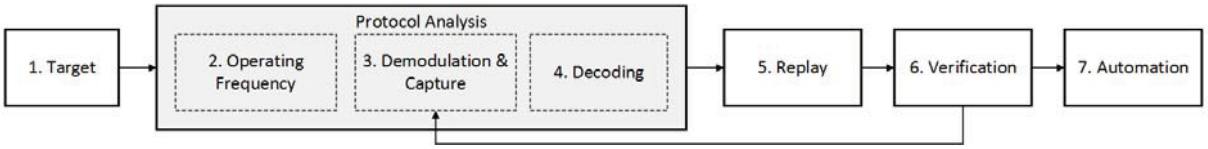

Figure 1. RF Replay Attack Methodology

Figure 1 represents the stages typically involved when attempting to carry out RF replay attacks. Essentially, the process commences with the selection of the target device. In a modern domestic setting, wireless door chimes, GDEs and RKEs are typically available and present a number of opportunities as potential targets.

Once the target is identified, the protocol analysis phase begins. This phase seems to be the most complex part of the process as it consists of multiple steps and often requires the repetition of some of the steps to achieve a successful outcome, which in this context means being able to reliably decode and retransmit the captured signal so that the target reacts to it.

The first step of this phase is concerned with the identification of the operating frequency of the selected device. In the United States, the Federal Communications Commission Identification (FCC ID) label attached to the device itself may be used to look up the operating frequency of the device online (Weber, 2013). In the United Kingdom, the Radio & Telecommunications Terminal Equipment (R&TTE) approval label is claimed to provide the ability to determine the frequency in a similar fashion (Laurie, 2013). Additionally, the information contained in various relevant patents may also be used to identify the frequency and even the specification of the communication protocol in some cases (Atlas, 2012b). Finally, the identification of the operating frequency may be achieved through a more tedious process of systematically scanning the spectrum for radio signals while ensuring an active transmission from the device in parallel. Various open-source command line tools as well as SDR graphical user interface (GUI) front-ends are readily available to assist and potentially automate this task.

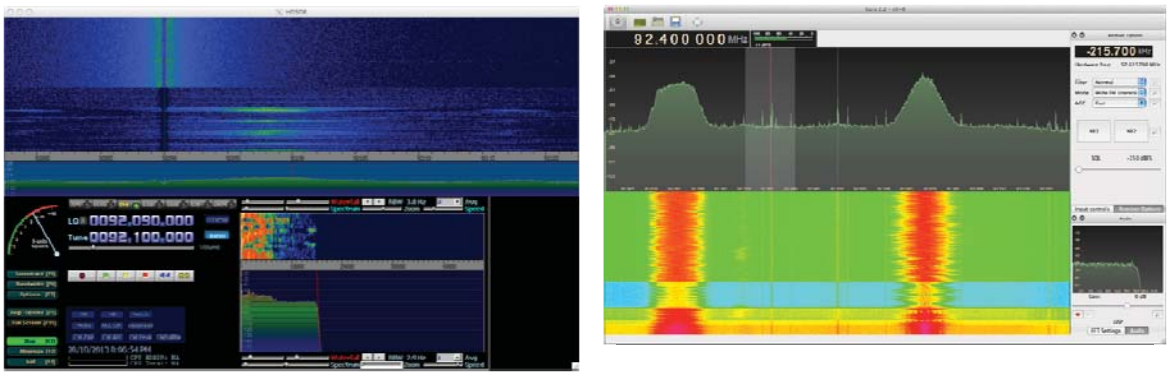

Figure 2. HDSDR (left) and Gqrx (right) SDR GUI Front-ends

The next step is to demodulate the raw signal and capture it into one of the common audio file formats, such as a WAV file. The recording of the signal may be performed using an SDR GUI such as HDSDR or Gqrx (Figure 2) and also via the means of a custom GNU Radio processing chain (Hohawk, 2012; Laurie, 2013). It should be noted that, while other methods potentially exist to achieve the same outcome, this approach is being described because it does not require deep understanding of RF-based communications. Also, the description of the various modulation schemes and the associated concepts is outside the scope of this research. The capture usually involves tuning an SDR to the designated operating frequency of the target device, using Amplitude Modulation (AM) as the demodulation scheme, since it is claimed to be the most likely used, and initiating the transmission from the device while recording at the same time.

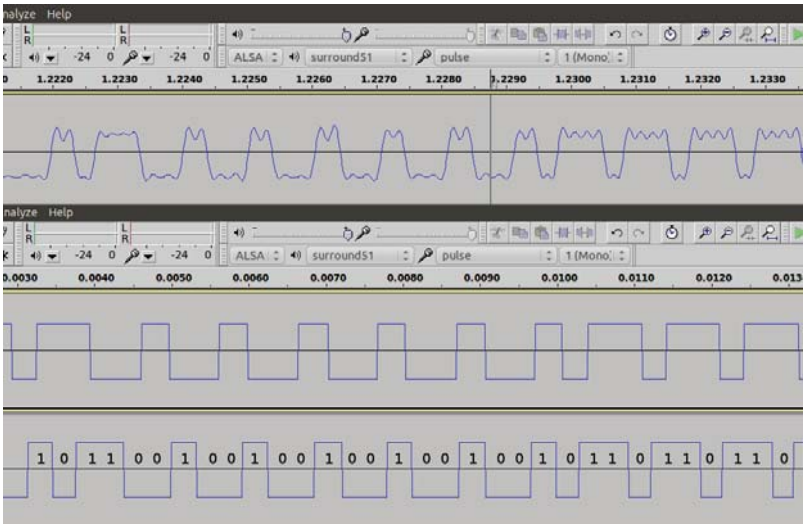

Figure 3. Captured Signal Wave (top), Square Signal Wave (middle), Decoded Binary Value Portion (bottom) (Laurie, 2013)

Once the signal has been captured, decoding is suggested to be carried out via the means of visual waveform analysis and decomposition. Essentially, the successful decoding relies on the ability to match the waveform peaks to respective data bits. Knowing the signal modulation scheme is required to convert the signal into binary form. Essentially, Amplitude Shift Keying (ASK) and On-Off Keying (OOK) modulation schemes, the latter being a simplification of ASK, are generally claimed to be the most likely schemes used due to low implementation cost and long battery life (Brown, Bagci, King, & Roedig, 2013; Holenarsipur, 2009). In the case of OOK, energy saving is achieved because the transmitter does not spend power when sending 0 or 1 bits depending on the specifics of the implementation pertinent to a particular device. Interestingly, previous research suggests that without having access to the specification of the communication protocol, it is not feasible to determine in advance whether a wave peak represents a 0 bit or a 1 bit and vice versa (Hohawk, 2012). Consequently, decoding is likely to turn into an iterative process with the outcome verification performed via signal retransmission.

To facilitate a better visual basis for the analysis, the captured signal may also be converted to square wave through the process of amplification. Figure 3 shows the visual differences between the originally captured wave and the resulting amplified square wave. Furthermore, the figure also represents the visual conversion process that is based on assuming the modulation protocol being OOK. In this case, the "highs" are assumed to represent the 1 bits and the "lows" the 0 bits respectively. The bit length also corresponds to the length of the shortest pulse. It should also be noted that it is common for the repeated bursts padded with whitespace to be transmitted as opposed to single signal instances. The final crucial component is the ability to determine the signal data rate or baud rate (Atlas, 2012b). In the context of visual analysis, measuring the length of the shortest pulse in a burst and dividing 1 by the measured length in seconds may be used to calculate the data rate (Hohawk, 2012). Given that this study is only concerned with primitive techniques, other modulation schemes, such as frequency shift keying (FSK), are considered to be out of scope. Analogically, sync words, modes and programmatic data (baud) rate detection are also not being described.

Having converted the signal into binary form provides the basis for subsequent replay. It is suggested that a tool called RfCat is well suited for this purpose, as it provides an easily accessible Python-based prompt for carrying out various RF-related research tasks interactively (Hohawk, 2012; Laurie, 2013; Weber, 2013). The tool is described in more detail in the next section. For the purpose of signal transmission, the binary form needs to be converted to hexadecimal form supported by Python. For example, the value of 01011100, which corresponds to 5C, may be passed to the interactive shell as “\x5C”. Provided that a compatible dongle is available and the RfCat firmware has been loaded onto

it, replaying the signal is a matter of using the interactive shell to:

1. Tune to the required operating frequency
2. Set the modulation scheme
3. Set the data rate
4. Transmit the string derived from the binary representation
5. Observe the reaction of the target device

If the replay is successful, the device is expected to respond to the signal. For instance, in the case of a wireless door chime, one should hear the bell ring. In the case of a garage door opener, the door should open or close depending on its previous state. As suggested, it is not granted that a successful replay will occur on the first attempt, at which point steps 3 to 6 will need to be repeated until a successful replay takes place. Finally, to ease the subsequent replays, one may choose to automate the steps required to configure the dongle and send the signal by writing a custom Python script. Interestingly, the eZ430-Chronos wireless development kit may also be used to carry out the transmission and there is an open-source tool available that aids with programming the watch (Laurie, 2013).

Using the process based on the methodology described above, a number of security researchers have been successful in carrying out replay attacks against a wireless door chime, remote gate and garage door opener, automotive remote car entry systems and a wireless power saver adaptor (Hohawk, 2012; Laurie, 2013). In the latter case, the consequences could be as dire as setting a residential property on fire, which would be likely to result in extensive property damage, injury or death (Tung, 2013). Subsequent comments left on the blog entry pages also indicate that ongoing research into communication protocols employed by wireless alarm systems have been taking place.

RFCAT

The open-source project that facilitates RF-based security research called RfCat is dubbed the "the Swiss army knife of sub-GHz radio" (Atlas, 2012a). The project has undergone a phase of active development by multiple contributors over the last two years with the last change committed in May 2013 at the time of writing. Officially, only the following three devices currently support the RfCat firmware:

- Radica IM Me wireless handheld device
- Texas Instruments CC1111 USB evaluation module kit
- Texas Instruments eZ430-Chronos wireless development kit (specifically, the CC1111-based USB access point)

In addition, a custom electronic badge has been specifically designed and built to support RfCat (Ossmann, 2012). While the project documentation contains a detailed set of instructions on loading the firmware onto the supported device, the interactive shell is sparsely documented and successful application of the tool in practice is said to require a sufficient level of understanding of radio communications (Weber, 2013).

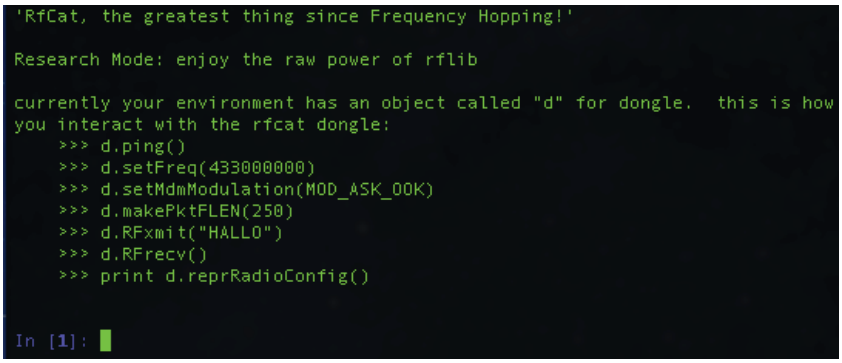

Figure 4. RfCat Interactive Shell - Welcome Message

Figure 4 presents the screenshot of the interactive RfCat shell that is shown immediately upon its launch. The shell provides a global object "d", which facilitates the necessary interface to the USB dongle. Based on the brief examination of the source code, the in-built help and the presentation by the original author, a number of key methods useful in the context of the study have been identified.

Table 2. RfCat Methods Found Useful in the Context of the Study

Method	Description
help(d)	Displays the in-built help available for all supported commands
help(d.[METHOD])	Displays the in-built help for the specified method, where [METHOD] should be replaced by name of the desired method
d.printRadioConfig	Prints the current detailed configuration of the dongle
d.setFreq	Sets the operating frequency
d.setMdmModulation	Sets the modulation scheme (such as OOK)
d.setMdmDRate	Sets the operating data (baud) rate
d.makePktFLEN	Sets the length of the transmitted or received packet to an arbitrary value
d.RFxmit	Transmits the signal corresponding to the supplied data value
d.RFlisten	Puts the dongle into monitoring mode that displays packets as they arrive
d.specan	Opens the spectrum analyser window

Table 2 only lists a small subset of the methods available, additional methods that support low-level signal discovery modes, as well as automated frequency scanning routines are also readily available. RfCat appears to be a versatile tool that has a number of potential applications to RF-based sub-GHz security research. The presented ability to listen to radio traffic in real time and have the decoded data presented on-screen in string form could potentially eliminate the need for the tedious visual protocol decomposition process and speed up the signal decoding phase by multiple orders of magnitude. Ultimately, the intention behind the tool was to implement a means of convenient access to the sub-GHz spectrum, so that security concerns associated with radio communications protocols employed by respective devices could be verified, justified and subsequently mitigated through informed device and protocol design improvements.

VERIFICATION

Aims and Assumptions

To verify the reported signal decoding and retransmission techniques in practice, the author has acquired the necessary equipment and selected a target device to achieve the following aims:

1. Capture the signal using an SDR
2. Decode the captured signal using visual waveform analysis method
3. Verify the accuracy of the resulting decoded signal via the means of a successful replay
4. Identify other potential ways of achieving the same outcome

The described verification process is based on the assumption that having a detailed understanding of radio communications is not required to be able to achieve a successful outcome. Furthermore, the author did not possess any of such knowledge prior to carrying out this research.

Materials

In order to follow the previously described methodology, the following materials were obtained:

- Terratec T-Stick DVB-T USB Stick (E4000) with metal antenna and antenna mount
- Texas Instruments eZ430-Chronos wireless development kit
- Texas Instruments CC-Debugger
- Generic soldering kit and solder wire
- A commodity laptop computer with x2 USB ports, Gqrx and Audacity installed

Because the eZ430-Chronos USB access point based on the CC1111 system-on-chip does not come with development header pins, debug interface connection to the CC-Debugger needs to be facilitated via the means of soldering.

Figure 5.eZ430-Chronos RF Access Point Connection to CC-Debugger Soldering Specification (top), the Resulting Successful Debugging Interface (bottom) (TI, 2013)

Figure 5 represents the soldering schematic appropriate for the USB dongle used in this study (top), as well as the resulting interface cable connected to the CC Debugger with the green LED indicating a successful connection (bottom). Once the debugging interface is functioning, the custom RfCat boot loader firmware needs to be loaded onto the access point using the Texas Instruments SmartRF Flash Programmer software. Having satisfied the necessary pre-requisites, the custom RfCat firmware may be loaded onto the dongle by following the instructions supplied with the tool. Subsequent capture and protocol analysis described in the next sections were carried out in an isolated laboratory setting.

Following the Methodology

A locally manufactured wireless door chime was acquired for the purpose of this study. This target was assumed to yield a higher potential of successful replay compared to devices that are known to employ sophisticated security mechanisms, such as RKEs (Aerts et al., 2012). The operating frequency of the device was initially discovered by visually inspecting the chime component case, as it was conveniently provided on the attached label. Subsequently, tuning the SDR to the discovered value and testing for signal presence while pushing the button on the remote verified the found operating frequency, which in this case was 433.92MHz.

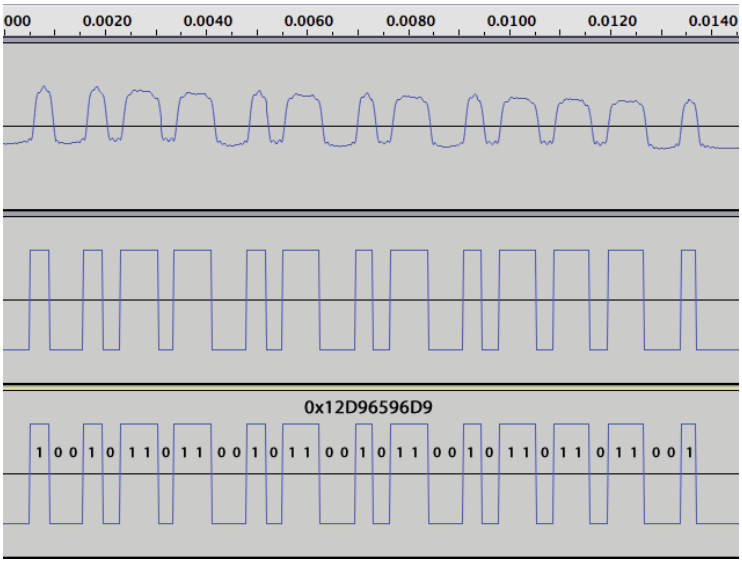

Figure 6. Captured Signal Wave (top), Square Signal Wave (middle), Decoded Binary and Hex Values (bottom)

Analogously to the previous reports, the modulation scheme employed by the device was assumed to be AM. At that stage, the frequency value and modulation scheme provided sufficient basis for signal capture. Once the signal was captured, it was established that the demodulated waveform comprised of about 60 short bursts, all of which were identical. At this point, due to the simplicity of the originally captured waveform the signal decoding could also be done without the need for conversion to square wave. Nevertheless, conversion was performed for subsequent demonstration purposes. Figure 6 represents the captured signal in its original form, as well as the resulting square wave and the decoded binary and hexadecimal values. The decoding assumes that the shortest peak represents the 1 bit. Measuring the length of the shortest peak in seconds and dividing 1 by the measured value provided the data rate of 2500 bits per second.

The initial replay was not successful because the whitespace between the bursts did not match that expected by the device. The length of the required whitespace was determined in an iterative manner to be 3 NULL bytes. Finally, sending the decoded binary value of “\x12\xD9\x65\x96\xD9” (payload) immediately followed by “\x00\x00\x00” (whitespace) 60 times resulted in a successful

replay as the target responded to the signal. Later, it was also determined that sending as little as 18 bursts also resulted in a successful replay. While subsequent automation was out of scope of the study, one could potentially create a script to brute force the associated signal key space to determine how quickly a successful replay may be achieved automatically after a code change or using another target of a similar kind.

CONCLUSION

Building upon the previous research, the study has presented a basic RF replay attack methodology based on visual signal analysis. In the context of simple devices that utilise the OOK modulation scheme, the presented methodology has been verified to be successful and proves that carrying out simple RF replay attacks does not require substantial knowledge of radio communications. The study has shown that a low-cost SDR coupled with a USB dongle compatible with the RfCat firmware provide a solid foundation for carrying out RF signal analysis and retransmission tasks. Ultimately, this emphasises the low level of skill and funds required to carry out RF attacks against other devices and the associated security challenges.

The study highlights a number of opportunities for future research. Firstly, other devices and more complex communication protocols could be analysed in greater detail to derive additional and potentially more efficient decoding techniques. For example, conversion of wave pulses into binary and hexadecimal representations could be automated using a custom script. Secondly, RF signals could potentially be used in locational fingerprinting. Thirdly, the features of RfCat that facilitate automated signal discovery and decoding without the need for visual waveform inspection could be studied in more detail. Finally and most importantly, research into ways of securing weak radio communication protocols could assist with protecting the sub-GHz spectrum from potential attacks in the future.

REFERENCES

ACMA. (1999). Spectrum at 434 MHz for low powered devices. from http://www.acma.gov.au/theACMA/spectrum-at-434-mhz-for-low-powered-devices

Aerts, W., Biham, E., Moitié, D., Mulder, E., Dunkelman, O., Indesteege, S., . . . Verbauwhede, I. (2012). A Practical Attack on KeeLoq. *Journal of Cryptology, 25*(1), 136-157. doi: 10.1007/s00145-010-9091-9

Atlas. (2012a). RfCat - Supported Dongles. Retrieved September 30, 2013, from https://code.google.com/p/rfcat/wiki/SupportedDongles

Atlas. (2012b). *Sub-Ghz or Bust*. Paper presented at the Black Hat USA 2012 Conference.

Brown, J., Bagci, I. E., King, A., & Roedig, U. (2013). *Defend your home!: jamming unsolicited messages in the smart home*. Paper presented at the Proceedings of the 2nd ACM workshop on Hot topics on wireless network security and privacy, Budapest, Hungary.

Frenzel, L. (2010). *Electronics Explained*: Newnes.

Harizanov, M. (2012). Controlling 433Mhz RF power sockets with a RFM12B module. Retrieved October 12, 2013, from http://harizanov.com/2012/04/controlling-433mhz-rf-power-sockets-with-a-rfm12b-module/

Harney, A., & O'Mahony, C. (2006). Wireless Short-Range Devices: Designing a Global License-Free System for Frequencies < GHz. *Analog Dialogue, 40-03*.

Hohawk, A. (2012). Hacking fixed key remotes. Retrieved September 12, 2013, from http://andrewmohawk.com/2012/09/06/hacking-fixed-key-remotes/

Holenarsipur, P. (2009). I'm OOK. You're OOK? Retrieved October 17, 2013, from

http://pdfserv.maximintegrated.com/en/an/AN4439.pdf

Laurie, A. (2013). You can ring my bell! Adventures in sub-GHz RF land... Retrieved October 3, 2013, from http://adamsblog.aperturelabs.com/2013/03/you-can-ring-my-bell-adventures-in-sub.html

Liang, F., & Liu, B. (2011). Research of Sub-GHz Wireless Sensor Network and Its Application in Grain Monitoring System. *Journal of Computational Information Systems, 7*(10).

Neely, M., Hamerstone, A., & Sanyk, C. (2013). *Wireless Reconnaissance in Penetration Testing*. Boston: Syngress.

Ossmann, M. (2012). The ToorCon 14 Badge. Retrieved October 22, 2013, from http://ossmann.blogspot.com.au/2012/10/the-toorcon-14-badge.html

Ossmann, M. (2013). HackRF, an open source SDR platform. Retrieved October 21, 2013, from http://www.kickstarter.com/projects/mossmann/hackrf-an-open-source-sdr-platform

Sikken, B. (2009). 433 MHz projects. Retrieved October 18, 2013, from http://bertrik.sikken.nl/433mhz/

SiliconLabs. (2010). Key Priorities for Sub-GHz Wireless Deployment.

Smith, C. (2012). Tracking planes for $20 or less. Retrieved October 22, 2013, from http://www.irrational.net/2012/08/06/tracking-planes-for-20-or-less/

TI. (2013). eZ430-Chronos™ Development Tool: Texas Instruments.

Tung, L. (2013). Burning down the house with an RF hacking watch. Retrieved October 17, 2013, from http://www.cso.com.au/article/456198/burning_down_house_an_rf_hacking_watch/

Weber, D. C. (2013). Radio Communication Analysis using RfCat. Retrieved October 20, 2013, from http://labs.inguardians.com/