DETECTING THE PRESENCE OF HIDDEN WIRELESS EAVESDROPPERS

BY

ANADI CHAMAN

THESIS

Submitted in partial fulfillment of the requirements
for the degree of Master of Science in Electrical and Computer Engineering
in the Graduate College of the
University of Illinois at Urbana-Champaign, 2018

Urbana, Illinois

Adviser:

Assistant Professor Haitham Hassanieh

ABSTRACT

This thesis explores the possibility of detecting the hidden presence of wireless eavesdroppers. Such eavesdroppers employ passive receivers that only listen and never transmit any signals, making them very hard to detect. We show that even passive receivers leak RF signals on the wireless medium. This RF leakage, however, is extremely weak and buried under noise and other transmitted signals that can be 3-5 orders of magnitude larger. Hence, it is missed by today's radios. We design and build Ghostbuster, the first device that can reliably extract this leakage, even when it is buried under ongoing transmissions, in order to detect the hidden presence of eavesdroppers. Ghostbuster does not require any modifications to current transmitters and receivers, and can accurately detect the eavesdropper in the presence of ongoing transmissions. Empirical results show that Ghostbuster can detect eavesdroppers with more than 95% accuracy up to 5 meters away.

To my parents, for their love and support.

ACKNOWLEDGMENTS

I would like to sincerely thank my adviser, Professor Haitham Hassanieh, for the invaluable support he provided to me during my work. The numerous interactions that I had with him over the past two years have helped me grow as a researcher. I would also like to thank Professor Romit Roy Choudhury whose holistic guidance on research and life in general, has been a great source of motivation for me. Many thanks to my collaborators, Jiaming Wang and Jiachen Sun, who contributed greatly to the work presented in this thesis. Additional thanks to all my labmates and colleagues at Systems and Networking Research Group (SyNRG) for their help and guidance.

I would like to thank my friends, both at UIUC and back in India, without whom life would not have been so enjoyable. Special thanks to my close friend and labmate Suraj Jog, who helped me believe in myself during tough times. The large variety of discussions that I had with him have been a great source of learning and fun. I must also thank our former office assistant Carol Wisniewski, whose warm and motherly presence in my life helped me in times when I missed my family, thousands of miles away in India.

Finally, I would like to extend my gratitude towards my family members who have always had my back. It is thanks to the endless love, support and advice of my parents, Chaman Lal Sharma and Savita Chaman, that I could achieve what I did. The credit for what I am today entirely goes to them. Additionally, I am extremely lucky to have an amazing brother, Anmol Chaman, who has been my confidant and dearest friend to this day. I am also thankful for all the other members of my family, who I love and appreciate tremendously.

CONTENTS

Chapter 1

INTRODUCTION

Eavesdropping on wirelessly transmitted data is a longstanding security threat in wireless networks. Wireless radios often rely on cryptographic solutions to defend against eavesdropping. However, encryption standards are under constant attack and can certainly suffer from security loopholes. Vanhoef et al. [1] is a classic example that shows how the universally adopted WPA2 WiFi security standard is vulnerable to a "key reinstallation attack," allowing the attacker to decrypt packets. Various side channel attacks have also been shown to exploit electromagnetic or acoustic signals to extract the encryption key [2, 3, 4, 5]. Furthermore, due to low cost, low power requirements, some wireless IoT systems adopt weak encryption protocols or lack encryption altogether, leaving them widely exposed to eavesdropping [6, 7, 8, 9].

Unfortunately, detecting the presence of an eavesdropper remains an open problem. Wireless receivers are passive devices that only listen in on the medium without transmitting any signal. Hence, there are no practical solutions today to discover eavesdroppers. Yet, such a capability can serve as a strong primitive in defending against eavesdroppers. In light of this situation, *we ask whether it is possible to detect the hidden presence of a passive wireless eavesdropper planted in the environment?*

This thesis takes the first steps towards positively answering this question. In doing so, we rely on the following key observation: even though a wireless eavesdropper does not produce active signals, its underlying hardware does leak RF signals on to the frequency spectrum. Specifically, every wireless receiver must use a local oscillator to generate a sinusoidal signal at the center frequency of operation, e.g., 2.4 GHz or 5 GHz for WiFi. This sinusoidal signal is mixed with the received wireless signal to down-convert it to baseband for digital sampling and processing as shown in Figure 1.1. Even if the wireless radio is only receiving and not transmitting, this sinusoidal signal can leak back through the antenna onto the wireless medium.

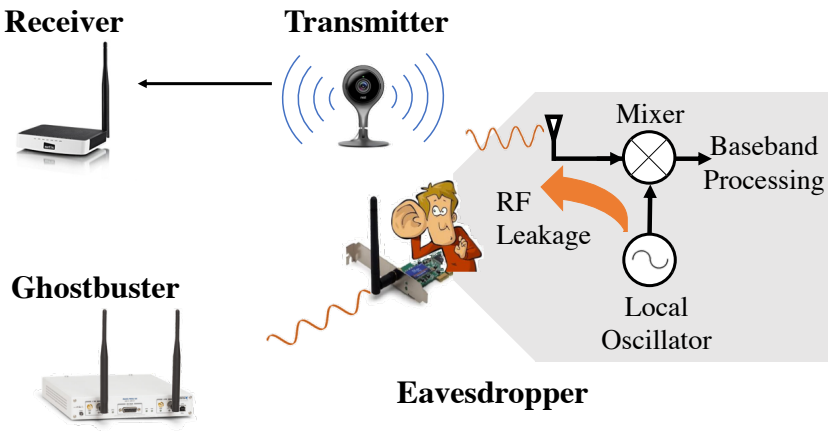

Figure 1.1: Ghostbuster's threat model.

The leakage, however, is extremely weak. In fact, it is significantly below the noise floor and hence cannot be detected by today's wireless receivers. One solution is to capture the signal over a long time window, e.g. 1 sec, and compute a multi-million point FFT over all collected samples in order to average the leakage and bring it above the noise floor.[1] However, taking such a large time window is bound to include packets transmitted on the wireless medium as well as leakages from other legitimate receivers. As a result, the eavesdropper's leakage will be buried under other transmitted signals which can be three to five orders of magnitude larger.

In addressing the above challenge, we introduce Ghostbuster, a device that can extract the leakage of a wireless eavesdropper buried under noise and transmitted signals without requiring any modifications to current transmitters and receivers. Ghostbuster leverages MIMO, multiple antenna systems, to separate the leakage from the transmitted signals in the antenna/spatial domain. Specifically, Ghostbuster can estimate the wireless channel from the transmitter and use it to zero-force the transmitted signals. Once the signal source is canceled, the leakage from the eavesdropper is revealed.

Of course, a robust separation of the weak leakage would require very efficient MIMO cancellation of the transmitted signals. While current MIMO algorithms can separate signals transmitted from two or more sources of comparable power for the purpose of decoding data bits, they cannot suffi-

[1]Note that the receiver used to capture the signal must configure its own local oscillator to a slightly shifted center frequency in order to ensure that its own local sinusoid does not overshadow the eavesdropper's leakage.

ciently cancel the transmitted signal for the purpose of extracting the leakage. To understand why, recall that Ghostbuster must take an FFT over a very large time window which would include several transmitted packets. Hence, over this time window the transmitted signal will exhibit discontinuities. These discontinuities manifest as artifacts and spurious frequencies in the frequency domain that are hard to cancel with standard MIMO techniques and ultimately leave a residual that continues to mask the leakage from the eavesdropper.

To perfectly cancel the transmitted signal, Ghostbuster leverages a two-stage recovery and cancellation algorithm that performs cancellation in two domains: spatial domain and frequency domain. In the first stage, Ghostbuster uses a new recovery algorithm that can extract the values of the frequencies in the continuous frequency spectrum in order to properly estimate the artifacts caused by discontinuities in the signal and cancel them in the frequency domain. In the second stage, Ghostbuster estimates and computes a higher resolution wireless channel across the different MIMO antennas that allows us to efficiently cancel the transmitted signals and the artifacts after taking a very large FFT in the spatial domain. This enables Ghostbuster to extract the leakage signals from the eavesdropper.

In addition to canceling transmitted signals, Ghostbuster must also separate the leakage of the eavesdropper from the leakage of other legitimate receivers. To do so, Ghostbuster leverages the frequency dimension where hardware imperfections cause small frequency offsets at different receivers. Ghostbuster exploits these hardware imperfections to separate the leakages from different receivers.

We built a prototype of Ghostbuster with multiple antennas using USRP N210 software-defined radios. We evaluated Ghostbuster in an office building using both USRPs as eavesdroppers at 900 MHz, 1.8GHz and 5GHz as well as WiFi cards placed in monitor mode at 5 GHz. We also ran experiments in a large empty parking lot where there were no WiFi signals. Our results show that Ghostbuster can detect the presence of USRP eavesdropper with more than 95% accuracy up to 5 meters despite ongoing transmissions and leakages from other receivers. For WiFi card based eavesdroppers, Ghostbuster can detect them with 89% accuracy up to 1 meter in the presence of ongoing transmissions and leakages from other WiFi cards.

Ghostbuster, to the best of our knowledge, is the first system that can

practically discover the mere presence of hidden eavesdroppers passively listening on the wireless medium even in the presence of ongoing transmissions and other receivers. In doing so, Ghostbuster does not require any changes to the current wireless transmitters and receivers being used today. Hence, it provides a readily deployable active defense layer against eavesdropping.

While Ghostbuster can extract and separate the leakage of the eavesdropper from other transmitters and receivers, a current limitation of our system is its inability to tell which leakage corresponds to the eavesdropper and which corresponds to the legitimate receiver. Our current threat model assumes that Ghostbuster knows the number of legitimate receivers since they are typically in plain sight and hence, can discover an eavesdropper by detecting additional leakage signals. While this might be difficult in some office and home WiFi networks, it can be adopted in secure facilities where the number of radios can be known and controlled. Chapter 6 discusses the future work to overcome such limitation.

Chapter 2

PROFILING RF LEAKAGE

We will start by characterizing the RF leakage from wireless receivers. We will focus on two types of receivers: WiFi cards and software-defined radios.

2.1 WiFi Cards

An adversary can configure a WiFi card to operate in monitor mode. In this case, the card will not transmit any packets and will only receive, which would hide the presence of the eavesdropper. While WiFi operates in the 2.4 GHz and 5 GHz bands, the leakage need not be at the same frequency. The exact frequency of the leakage depends on the hardware architecture of the receiver and the frequency of the local oscillator. We will describe several architectures commonly used in off-the-shelf WiFi receivers. The choice of architecture and trade-offs depends on circuit optimizations which are beyond the scope of the work in this thesis. However, it is important to examine these designs in order to determine the frequency of leakage that will allow us to detect the presence of an eavesdropper.

We examined over 20 WiFi cards in desktops, laptops, cellphones and access points that run different protocols including 802.11a, b, g, n, ac. The cards use WiFi chipsets from three main manufacturers: Intel, Qualcomm, and Broadcom. Figure 2.1 shows four different simplified architectural designs used in these chipsets. The first design, shown in Figure 2.1(a), is direct conversion where the local oscillator generates a 2.4 GHz or 5 GHz signal that is directly used to down-convert the received signal to baseband. The second design, shown in Figure 2.1(b), is commonly used in WiFi cards that operate at 2.4 GHz. The local oscillator generates a 4.8 GHz signal that is then divided to generate a 2.4 GHz which is mixed with the received signal. In this case, the strongest leakage observed is at 4.8 GHz.

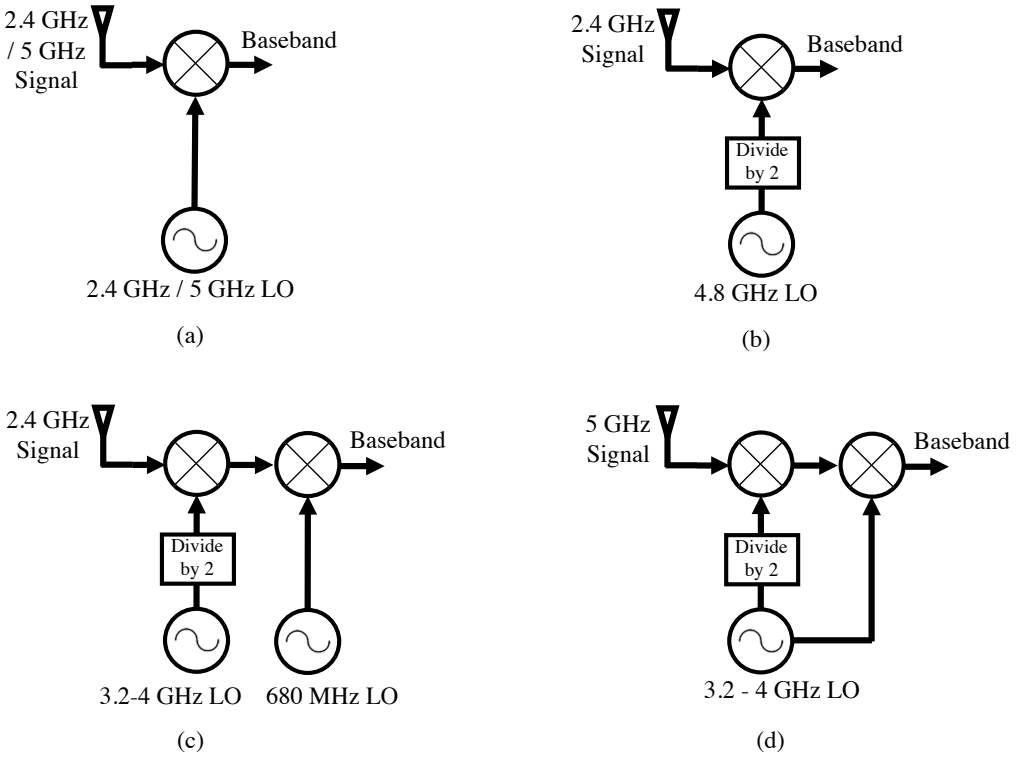

Figure 2.1: Simplified receiver architectures used in commercial off-the-shelf WiFi cards.

The third and fourth leverage a heterodyne (multi-stage) architecture where the signal is first down-converted to an intermediate frequency and then again to baseband. For 2.4 GHz, shown in Figure 2.1(c), the main local oscillator generates a signal between 3.2 GHz and 4 GHz that is divided to generate a signal between 1.6 GHz and 2 GHz. This signal is used to down-convert the received signal to a center frequency of 680 MHz. Then, another local oscillator at 680 MHz is used to down-convert it to baseband. In this case, the leakage is observed at both frequencies of the two oscillators. However, our observation reveals that the 680 MHz is typically weaker. Finally, for 5 GHz, the signal is also down-converted twice. However, a single local oscillator is used as shown in Figure 2.1(d).

To formalize this, we can express the frequency of the leaked signal f_l as a function of the center frequency f_c for each one of the architectures:

$$\begin{array}{lrcl} (a) & f_l & = & f_c \\ (b) & f_l & = & 2 \times f_c \\ (c) & f_l & = & 2 \times (f_c - 680 \text{ MHz}) \\ (d) & f_l & = & 2/3 \times f_c \end{array} \tag{2.1}$$

It is important to understand the different designs in order to determine at which frequency the leakage will occur and hence to set Ghostbuster to receive at the desired frequency. As described earlier, Ghostbuster will set its own center frequency f_c' slightly shifted from the leakage frequency to ensure the leakage from its own local oscillator does not overwhelm the leakage from the eavesdropper, i.e. Ghostbuster sets $f_c' = f_l + \Delta f$. It can then capture samples over a long time window and take an FFT of the captured samples. If an eavesdropper is present, it should see a sharp spike in the FFT bin corresponding to a frequency of Δf.

To verify the feasibility of detecting leakage from WiFi cards, we conducted experiments on over 20 cards. To avoid interference from WiFi devices and access points, we ran the experiments in a large parking lot with no WiFi interference. We placed the WiFi cards in monitor mode to ensure they are only receiving and used a USRP software radio placed 1 meter away to measure the leakage. For each card, we collected signals over a window of 1 second and measured the SNR of the leakage. We verified that the leakage disappears once the card is turned off.

Table 2.1: Leakage measured 1 meter away for different WiFi eavesdroppers listening at 2.437 GHz

WiFi Chipset/USRP Daughterboard	Design	Leakage freq. in GHz	Leakage SNR @ 1m in dB
Broadcom: BCM43xx, BCM4329, BCM4360, BCM4352, BCM43526	a	2.437	12.8 –23.0
Intel: 4965	c	3.514	19.8
Intel: 3165, 5100, 5300	b	4.874	12.6–19.7
Intel: 7260, 7265, 8260	b	4.874	10.1–13.1
Qualcomm: AR93XX	b	4.874	11.3
Qualcomm: AR9271, AR9485, AR9170	b	4.874	7.2–14.3
USRP N210: SBX board	a	2.437	50.8
USRP N210: CBX board	a	2.437	50.2
USRP N210: UBX board	a	2.437	53.4

Table 2.2: Leakage measured 1 meter away for different WiFi eavesdroppers listening at 5.745 GHz

WiFi Chipset/USRP Daughterboard	Design	Leakage freq. in GHz	Leakage SNR @ 1m in dB
Broadcom: BCM43xx, BCM4329, BCM4360, BCM4352, BCM43526	a	5.745	10.7-25.01
Intel: 4965	a	5.745	10.7
Intel: 3165, 5100, 5300	d	3.65	20.4-22.2
Intel: 7260, 7265, 8260	a	5.745	12.6–16.0
Qualcomm: AR93XX	d	3.65	21.1
USRP N210: CBX board	a	5.745	56.7
USRP N210: UBX board	a	5.745	57.5

Tables 2.1 and 2.2 show the resulting leakage frequency and SNR of the leaked signal when the WiFi card is set to frequency bands with center frequency $f_c = 2.437$ GHz and $f_c = 5.745$ GHz respectively. Since many WiFi cards, laptops and cellphones use the same underlying WiFi chipset, we report the results for different chipsets. The tables show that the frequency of the leakage depends on the WiFi chipset architecture and the leaked frequency matches the expected value derived from Equation (2.1). The tables also show that for all architectures, the leakage at 1 m is above 7 dB and can reach 25 dB. Hence, it can be detected at even farther distances.

Two points are worth noting:

- Higher SNR can be achieved by averaging over a longer time window. Specifically, we have used a time window of 1 sec. By collecting more

samples from a window that is $K\times$ larger, the SNR of the leakage signal can increase by $10\log_{10}(K)$ dB.

- The best eavesdropper strategy is to use a card with a direct conversion architecture as shown in Figure 2.1(a). This ensures that the leakage will be in the same frequency band and will be masked by other transmitters and receivers in the environment.

2.2 USRP Software-Defined Radios

An adversary can also use a USRP software-defined radio to eavesdrop on ongoing transmissions. The advantage of using USRPs is that they are frequency and protocol independent. An adversary can configure the software radio to receive at any frequency between 10 MHz and 6 GHz and can decode the received signal in software, independent of protocol. To operate at different frequencies, the USRP software radio requires an RF daughterboard that supports the frequency range of operation. All daughterboards use the direct conversion architecture and hence are expected to leak at the center frequency of operation.

We experiment with USRP N210 using three daughterboards: SBX (400 MHz – 4.4 GHz), CBX (1.2 GHz – 6 GHz) and UBX (10 MHz – 6 GHz). Tables 2.1 and 2.2 also show the leakage for an eavesdropper 1 meter away using a USRP software radio with these daughterboards to eavesdrop on WiFi packets. In this case, the SNR of the leakage signal is around 50 dB which is significantly higher than WiFi cards. This, however, is expected since the USRP's RF circuits use a simple hardware architecture, whereas WiFi chips are heavily optimized and use state-of-the-art components that minimize leakage.

Chapter 3

GHOSTBUSTER

While the above shows the feasibility of detecting RF leakage from WiFi cards and software-defined radios, it assumes there are no transmissions on the medium. However, this is not the case in practice since taking an FFT over a large time window (e.g. 1 sec) is bound to include transmitted packets. To address this, we introduce Ghostbuster, a device that can extract the RF leakage of a wireless eavesdropper even if it is buried under large transmitted signals.

We will describe Ghostbuster in the context of WiFi networks. For simplicity, we will first focus on the case where there is a single WiFi transmitter and an eavesdropper. We will specifically describe Ghostbuster's algorithms for OFDM based packet transmissions since OFDM is the most prevalent modulation scheme used today.

In order to extract the eavesdropper's leakage, Ghostbuster must first nullify the transmitted packets along two dimensions:

- **Spatial Dimension:** Ghostbuster leverages MIMO to cancel the transmitted signal and separate it from the eavesdropper's RF leakage.

- **Frequency Dimension:** Ghostbuster estimates and cancels artifacts and spurious frequencies resulting from discontinuities in the time domain signal.

3.1 Spatial Cancellation with MIMO

Consider a Ghostbuster system with a two antenna MIMO. Let $y_1(t)$ and $y_2(t)$ be two time-domain signals received concurrently on each antenna. Let $x(t)$ be the transmitted signal and $e(t)$ be the eavesdropper's leakage. Ghost-

buster receives:

$$\begin{aligned} y_1(t) &= h_{e1}e(t) + h_{t1}x(t) \\ y_2(t) &= h_{e2}e(t) + h_{t2}x(t) \end{aligned} \tag{3.1}$$

where h_{t1} and h_{t2} are channels from the transmitter, and h_{e1} and h_{e2} are channels from the eavesdropper to the two MIMO receivers of Ghostbuster. We can rewrite the above equation in vector format:

$$\vec{y} = \vec{h_e}e(t) + \vec{h_t}x(t) \tag{3.2}$$

Figure 3.1 shows a representation of these vectors in the antenna space. By projecting on a direction $\vec{h}_t^{\perp}$ orthogonal to $\vec{h_t}$, we can cancel the signal from the transmitter. The remaining projection $\vec{e_p} = \vec{h}_t^{\perp} \cdot \vec{y}$ will only correspond to the eavesdropper's leakage.

However, most wireless systems today like WiFi and LTE use OFDM. Wideband OFDM signals experience frequency-selective fading. The wireless channel h must be computed per OFDM bin and cancellation must be performed per bin. Hence, we can rewrite the above equations per OFDM frequency bin:

$$\begin{aligned} \widehat{Y}_1(f) &= H_{e1}(f)\widehat{E}(f) + H_{t1}(f)\widehat{X}(f) \\ \widehat{Y}_2(f) &= H_{e2}(f)\widehat{E}(f) + H_{t2}(f)\widehat{X}(f) \end{aligned} \tag{3.3}$$

In the equations above, the eavesdropper's leakage signal is a single sinusoid, i.e. $e(t) = cos(2\pi f_l t)$ and $E(f) = \delta(f - f_l)$. Hence, the leakage appears in a single OFDM bin. Typically, this is the DC (zero) OFDM bin since, as discussed earlier, the optimal strategy of the eavesdropper is to set its center frequency, the same as the transmitted signal. The DC OFDM bin contains the signal from the transmitter's local oscillator and thus modulated data bits are not sent in the DC bin.

It is, however, hard to estimate the channel of the transmitter's signal in the DC bin in order to cancel this signal. First, no known preamble bits are ever sent in the DC bin that would allow us to estimate $H_{t1}(f_{DC})$ and $H_{t2}(f_{DC})$. Moreover, at any point, the bin contains the sum of the transmitter's signal and the eavesdropper's leakage, making it hard to separate the two and estimate the channels.

To address this issue, we rely on two key observations. First, we do not need to know the exact values of the channels. The ratio of the channels is

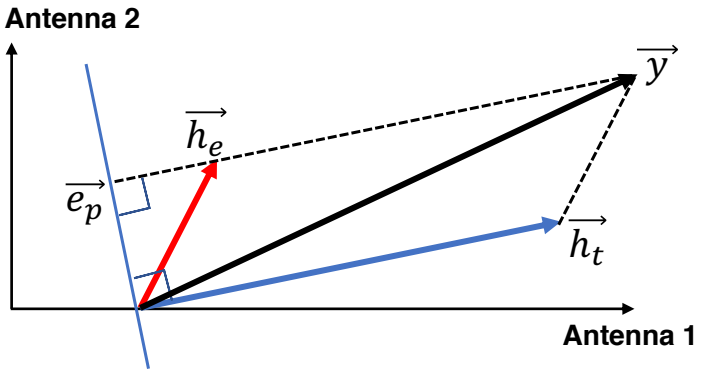

Figure 3.1: Ghostbuster leverages MIMO to cancel the signal from the transmitter in the Antenna Space.

sufficient for cancelling the transmitter's signal. Specifically, if we know the ratios, we can compute:

$$\begin{aligned} &\widehat{Y}_1(f) - \frac{H_{t1}(f)}{H_{t2}(f)}\widehat{Y}_2(f) \\ &\quad = \left(H_{e1}(f) - \frac{H_{t1}(f)}{H_{t2}(f)}H_{e2}(f)\right)\widehat{E}(f) = C\widehat{E}(f) \end{aligned} \tag{3.4}$$

where C is some constant.

Second, the signal from the transmitter's oscillator and the eavesdropper's leakage are not completely aligned in frequency. In particular, the oscillators of the transmitter and eavesdropper are not physically synchronized. Hence, they exhibit a small frequency offset which is typically referred to as carrier frequency offset ($CFO = \Delta f_c$). This offset can be used to separate the two signals along the frequency dimension by taking an FFT over a large time window. By separating the two signals into two different frequency bins, we can find a frequency bin that contains only the transmitter's signal on both of Ghostbuster's receivers, i.e.,

$$\begin{aligned} \widehat{Y}_1(f) &= H_{t1}(f)\widehat{X}(f) \\ \widehat{Y}_2(f) &= H_{t2}(f)\widehat{X}(f) \end{aligned} \tag{3.5}$$

By taking the ratio of the signals in this bin, we can compute the ratio of the channel even if $\widehat{X}(f)$ does not contain a known preamble bit.[1]

[1]Note that within a very narrow frequency band, e.g. 1 OFDM bin, the wireless channel is flat.

Unfortunately, simply taking an FFT over a large time window is not sufficient to separate the signals in frequency domain. Over a large time window, the signals exhibit discontinuities that result in artifacts and spurious frequencies which mask the eavesdropper's leakage. Hence, in order to detect the eavesdropper's leakage and cancel the transmitter's signals, Ghostbuster must first estimate and cancel the impact of discontinuities along the frequency dimension.

3.2 Frequency Cancellation of Artifacts

Before we describe how Ghostbuster deals with discontinuities, it is important to first understand why discontinuities result in artifacts and spurious frequencies.

A. Discontinuities & Artifacts:

To better understand this, let us first focus on a single OFDM subcarrier in a single OFDM symbol shown in Figure 3.2(a). The transmitter takes an inverse FFT of the OFDM symbol to transform it to the time domain before it transmits it on the wireless channel as shown in Figure 3.2(b). When Ghostbuster receives this symbol, it takes an FFT over a much larger time window. This process can be viewed as taking a much longer periodic time signal and windowing it to the length to the OFDM symbol as shown in Figure 3.2(d). Multiplying a signal with a window in time is equivalent to convolving it with a *sinc* function[2] in frequency. Thus, the subcarrier is convolved with a sinc and once Ghostbuster takes a very long FFT, the side-lobes of the sinc will appear and mask the eavesdropper's leakage.

[2]The *sinc* function is defined as $\text{sinc}(x) = \sin(x)/x$.

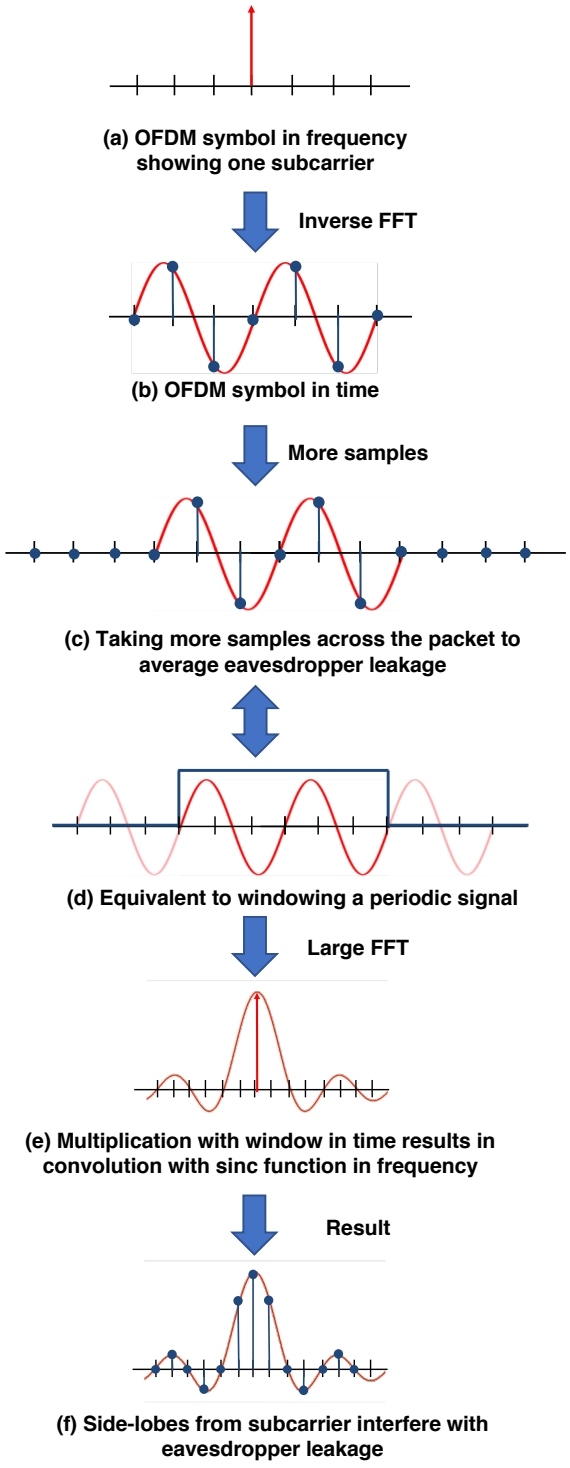

Figure 3.2: Discontinuities: Lack of periodicity of the OFDM symbols across the entire packet results in spurious frequencies and artifacts that continue to mask the eavesdropper's leakage.

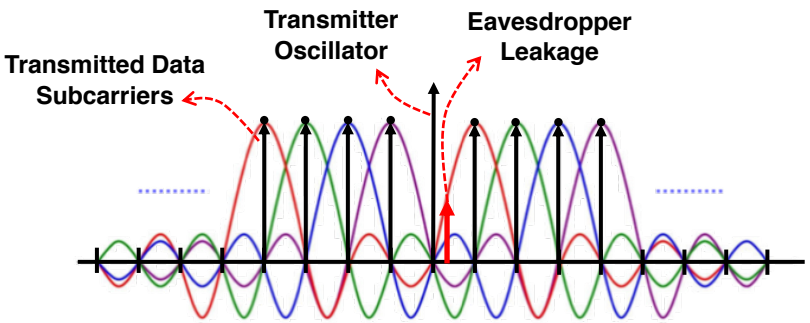

Figure 3.3: OFDM subcarriers' side-lobes after taking a very long FFT.

The above description only considered one subcarrier from a single OFDM data symbol. However, side-lobes from all the subcarriers from all OFDM data symbols in the packet are going to sum up together, which manifests as artifacts and spurious frequencies. Figure 3.3 shows that the side-lobes from many OFDM subcarriers combine together.

B. Estimating & Cancelling Artifacts:

Consider an OFDM symbol with N subcarriers. Let $x(t)$ be the time domain version of this OFDM symbol received at Ghostbuster. In the discrete domain, we have:

$$x(t) = \sum_{k=0}^{N-1} a_k e^{j2\pi f_k t/N} + w(t), \ t = 0, 1, ..., N-1 \tag{3.6}$$

where $w(t)$ is additive white Gaussian noise, f_k is the frequency of the k^{th} OFDM subcarrier[3] and a_k is a complex amplitude corresponding to modulated data bit weighted by the wireless channel. For example, for BPSK modulation $a_k = \pm H(f_k)$ where $H(f_k)$ is the wireless channel.

In order to eliminate the side-lobes generated by these subcarriers, we need to know the continuous values of the frequency estimates f_k as well as the accurate values of the amplitudes a_k. The best estimates of $\tilde{f}_k$ and $\tilde{a}_k$ would

[3]Note that due to CFO, the frequency of each subcarrier is shifted and no longer aligned with integers of the FFT grid.

minimize the following error function:

$$E(\tilde{\mathbf{a}}, \tilde{\mathbf{f}}) = \sum_{t=0}^{N-1} \left| x(t) - \sum_{k=0}^{N-1} \tilde{a}_k e^{j2\pi \tilde{f}_k t/N} \right|^2 \tag{3.7}$$

where $\tilde{\mathbf{a}}$ is a vector of $\tilde{a}_k$ and $\tilde{\mathbf{f}}$ is a vector of $\tilde{f}_k$.

Ghostbuster uses an iterative algorithm in order to minimize the above error function in Equation (3.7). It first finds $\tilde{\mathbf{a}}$ that minimizes E for a fixed $\tilde{\mathbf{f}}$. It then fixes $\tilde{\mathbf{a}}$ and finds $\tilde{\mathbf{f}}$ that minimizes E. Ghostbuster iterates back and forth until the algorithm converges and the error is minimized.

Hence, in each iteration, Ghostbuster

- **Solves for $\tilde{\mathbf{a}}$ given a fixed $\tilde{\mathbf{f}}$:** In this case, the error function E is convex in $\tilde{\mathbf{a}}$. In fact, the above optimization is a weighted least squares problem and has the following closed-form solution:

$$\tilde{a}_k = \frac{1}{N} \sum_{t=0}^{N-1} x(t) e^{-j2\pi \tilde{f}_k t/N} \tag{3.8}$$

- **Solves for $\tilde{\mathbf{f}}$ given a fixed $\tilde{\mathbf{a}}$:** In this case, the error function E is non-convex in $\tilde{\mathbf{f}}$ due to the complex exponentials. However, if we have good initial estimates of $\tilde{f}_k$ that are within a small interval around f_k, then the function becomes convex within this interval and we can use gradient descent to minimize it.

We start by showing that the error function is convex given good initial estimates of $\tilde{f}_k$. Let us consider a single subcarrier f_k. Our goal is to find $\tilde{f}_k$ and $\tilde{a}_k$ that minimize the error function:

$$E(\tilde{f}_k, \tilde{a}_k) = \sum_{t=0}^{N-1} \left| a_k e^{j2\pi f_k t/N} - \tilde{a}_k e^{j2\pi \tilde{f}_k t/N} \right|^2 \tag{3.9}$$

We prove the following theorem about the error function:

Theorem 1. *The error function $E(\tilde{f}_k, \tilde{a}_k)$ is convex for $\tilde{f}_k \in [f_k - \alpha, f_k + \alpha]$ for any $\alpha < 2/5$.*

We provide the proof of the above theorem in Appendix A. Here, we present the intuition behind it. Specifically, we show that the error function above

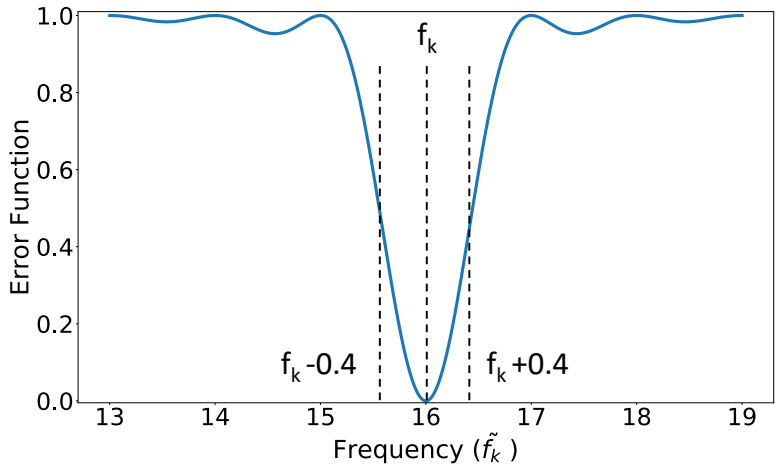

Figure 3.4: Normalized error function for a single subcarrier at $f_k = 16$, $a_k = 1$ and $N = 64$.

is the negative of the aliased sinc function and is given by:

$$E(\tilde{f}_k, \tilde{a}_k) = |a_k|^2 N - |a_k|^2/N \left(\frac{\sin(\pi(f_k - \tilde{f}_k))}{\sin(\pi(f_k - \tilde{f}_k)/N)} \right)^2 \tag{3.10}$$

Figure 3.4 shows an illustration of the normalized error function, $E(\tilde{f}_k, \tilde{a}_k)/N$ for $f_k = 16$, $a_k = 1$ and $N = 64$. As can be seen, if the initial estimate $\tilde{f}_k$ used for gradient descent is within a $f_k \pm 2/5$, the error function is convex. Hence, we can use gradient descent to minimize the error and achieve the global minimum as the solution for $\tilde{\mathbf{f}}_k$.

But how do we obtain good initial estimates of f_k? To do so, we leverage the standard OFDM decoder. Specifically, we have:

$$f_k = k + N\Delta f_c/B \tag{3.11}$$

where k is the integer index of the subcarrier on the FFT grid, Δf_c is the carrier frequency offset (CFO) and B is the bandwidth of the OFDM symbol. OFDM decoding naturally estimates the coarse CFO at the beginning of the packet as well as the residual CFO for every data symbol. We can use these CFO estimates to obtain very good initial estimate of f_k on which we can run gradient descent to further minimize the error.[4]

[4]Note that simply using these initial estimates as the true estimates results in a small residual error that accumulates across symbols and continues to prevent us from accurately

Given an initial $\tilde{\mathbf{f}}_{init}$ and $\tilde{\mathbf{a}}_{init}$, Ghostbuster iterates between solving for $\tilde{\mathbf{f}}$ and solving for $\tilde{\mathbf{a}}$ until the error function converges. Ghostbuster does this for every OFDM symbol in order to be able reconstruct and subtract the side-lobes and, hence, eliminate the artifacts.

3.3 Overall Algorithm

In this section, we put together Ghostbuster's overall cancellation algorithm. As described earlier, Ghostbuster performs cancellation both in the spatial and frequency dimensions using the following steps:

1. For each packet in the signal, Ghostbuster decodes the packet using standard OFDM decoding to obtain initial estimates of $\tilde{\mathbf{f}}$ from carrier frequency offset estimation.

2. For each symbol, Ghostbuster iterates between solving weighted least squares and gradient descent to solve the following optimization problem.

$$\tilde{\mathbf{f}}^*, \tilde{\mathbf{a}}^* = \arg\min_{\tilde{f}_k, \tilde{a}_k} \sum_{t=0}^{N-1} \left| x(t) - \sum_{k=0}^{N-1} \tilde{a}_k e^{j2\pi \tilde{f}_k t/N} \right|^2 \tag{3.12}$$

3. Ghostbuster then uses the estimates $\tilde{\mathbf{f}}^*$ and $\tilde{\mathbf{a}}^*$ to recover an accurate reconstruction $\tilde{x}(t)$ of $x(t)$. Ghostbuster does this for each MIMO receiver and then subtracts all the subcarriers of $\tilde{x}(t)$ other than the DC subcarrier from $x(t)$ to eliminate side-lobes that create artifacts.

4. Ghostbuster then uses the MIMO receivers to null the remaining transmitter's signal in the DC bin and its side-lobes from the packet as described in Section 3.1.

5. Finally, Ghostbuster combines the samples from all the nulled packets and takes a very large FFT across all samples.

6. At this point, the transmitter's signal including the artifacts are completely nulled and the eavesdropper's leakage is revealed.

detecting the eavesdropper's leakage.

A pseudocode of this algorithm is shown in Algorithm 1.

Algorithm 1 Ghostbuster's Cancellation Algorithm

for k^{th} packet **do**
  for m^{th} MIMO Receiver **do**
    Decode packet using standard OFDM decoder.
    for Each OFDM Symbol **do**
      $\tilde{\mathbf{f}}^{(0)} \leftarrow$ CFO coarse & fine estimates
      $i \leftarrow 1$
      while $E(\tilde{\mathbf{f}}^{(i-1)}, \tilde{\mathbf{a}}^{(i-1)}) \geq$ Threshold **do**
        $\tilde{\mathbf{a}}^{(i)} \leftarrow \textsc{WeightedLeastSq}(\tilde{\mathbf{f}}^{(i-1)}, x_m(t))$
        $\tilde{\mathbf{f}}^{(i)} \leftarrow \textsc{GradientDescent}(\tilde{\mathbf{a}}^{(i)}, x_m(t))$
        $i \leftarrow i + 1$
      end while
      $\tilde{x}_m(t) \leftarrow \tilde{\mathbf{a}}^*, \tilde{\mathbf{f}}^*$ (other than the DC bin)
      $r_m(t) \leftarrow x_m(t) - \tilde{x}_m(t)$
    end for
    $p_m(t) \leftarrow$ combination of $r_m(t)$ from all symbols
    $P_m(f) \leftarrow FFT(p_m(t))$
    $H_m(f_{DC}) \leftarrow P_m(f_{DC})$
  end for
  $s_k(t) \leftarrow$ spatial cancellation using each $H_m(f_{DC})$
end for
$s(t) \leftarrow$ combination of $s_k(t)$ from all packets
$S(f) \leftarrow FFT(s(t))$
Find spike of eavesdropper's RF leakage.

For the above algorithm to work in practice, Ghostbuster must address two issues in order to efficiently estimate and cancel the artifacts.

(1) Sampling & Packet Detection Offset: In order to perform the above optimization and reconstruct the artifacts that result from discontinuities, Ghostbuster must be able to detect the exact start of the packet and estimate and compensate for any sampling offsets. To do this, Ghostbuster leverages the fact that a packet detection and sampling offset Δt manifests as a linear

phase versus frequency in the frequency domain. Formally,

$$x(t - \Delta t) = \sum_{k=0}^{N-1} a_k e^{j2\pi f_k (t-\Delta t)/N} \tag{3.13}$$

Thus, we can write the phase in frequency as:

$$\phi = 2\pi f_k \Delta t / N \tag{3.14}$$

By performing linear regression on the phase versus the frequency, Ghostbuster can accurately estimate and compensate for sampling offsets.

(2) Cyclic Prefix: The above optimization was described in the context of N samples of the OFDM symbol. However, for each OFDM symbol, the transmitter appends a cyclic prefix which is a simple repetition of CP samples of the symbol in time domain. Ghostbuster must model the artifacts while taking the cyclic prefix into account.

One option is to run the optimization problem over all the samples including the cyclic prefix. Unfortunately, the cyclic prefix is typically large. For example, in WiFi, the cyclic prefix could be as large as 1/4 of the symbol length. Running the optimization problem on $N + CP$ samples significantly breaks the orthogonality of the subcarriers, rendering our initial estimates of f_k outside the convex region and thus yielding poor results. Another option is to run the optimization problem only over the N samples and use the result to reconstruct the cyclic prefix. However, this too yields poor results since the error increases with more samples outside the N symbol samples on which we ran the optimization.

To address this, Ghostbuster splits the symbol into two overlapping regions of length N. The first takes samples from $[0, 1, \cdots, N-1]$. The second takes samples from $[CP, CP+1, \cdots, CP+N-1]$. Ghostbuster runs the optimization algorithm on both regions and then combines the results by using the second region to estimate and compensate for the CP samples.

3.4 Detecting Eavesdroppers in the Presence of Other Receivers

So far, we have focused on the case where there is a single transmission and RF leakage only from the eavesdropper. However, in practice multiple legitimate receivers might be present and listening on the wireless medium. These receivers will also leak RF signals from their local oscillators. Ghostbuster can separate these leakages from the leakage of the eavesdropper along the frequency dimension. Leveraging the fact that different receivers have different CFOs due to hardware imperfections, Ghostbuster can separate the leakage from different receivers by taking a very large FFT over a long time window.

Ghostbuster can use time windows of 1 second to tens of seconds, to separate leakage from different receivers as long as the CFO between them is larger than tens of Hz. Typical values of CFO in practice, however, are 100s of Hz to few kHz. Hence, Ghostbuster can easily separate the leakage from multiple receivers as we will show in our results. Ghostbuster can then count the number of receivers in the environment and check against the expected number of legitimate receivers to detect the presence of an eavesdropper. Note, however, that this requires knowing the number of legitimate receivers a priori which is a current limitation of Ghostbuster as discussed in Chapter 6.

Chapter 4

IMPLEMENTATION

We implemented Ghostbuster using USRP N210 software-defined radios and evaluated in an indoor office environment with standard multipath. We experimented with two types of eavesdroppers:

- **USRP Software-Defined Radio:** We used USRP N210 as eavesdropper and we ran experiments in the 900 MHz ISM band to minimize interference and more easily collect benchmark results of Ghostbuster's performance. We also ran experiments at 1.8 GHz and 5.745 GHz.

- **WiFi Cards:** We used WiFi cards on MacBook Pro laptops. The cards were placed in monitor mode and set to the 5.745 GHz WiFi band. We chose this band since it was unused in our office building.

In each experiment, we placed one USRP as transmitter. The USRP transmits standard WiFi packets with OFDM modulation. We varied the location of the eavesdropper from a few cm to 14 meters. We ran the experiments in a total of 500 locations for USRP eavesdroppers and 430 locations for WiFi card eavesdroppers. In each location, we collected 1 second long measurements. We also varied the number of receivers listening in on these transmissions and leaking RF signals from their oscillators.

Chapter 5

EVALUATION RESULTS

In order to provide some insights into Ghostbuster's performance, we first provide some microbenchmark results and then evaluate the overall performance. We start by examining the case when there are no other transmitters or receivers and evaluate how well Ghostbuster can detect the leakage of the eavesdropper. We then move to the case when there is a single transmitter continuously sending OFDM packets. Finally, we evaluate Ghostbuster in the presence of transmissions as well as multiple receivers.

To evaluate the performance of Ghostbuster, we use the following metrics:

- *False Negative Rate:* Ratio of the number of runs where Ghostbuster failed to detect the presence of an eavesdropper to the total number of runs where the eavesdropper was present.
- *False Positive Rate:* Ratio of the number of runs where Ghostbuster falsely detected the presence of an eavesdropper to the total number of runs where eavesdropper was not present.
- *Hit Rate:* Ratio of the number of runs where Ghostbuster correctly detects the presence of an eavesdropper to the total number of runs where eavesdropper was present.
- *Detection Accuracy:* Ratio of number of runs where Ghostbuster correctly classified the presence of an eavesdropper to the total number of runs.
- *Count of Receivers:* Counting the correct number of receivers in range.
- *Leakage SNR:* Signal-to-noise ratio of the RF leakage per FFT frequency bin.

5.1 Eavesdropper's RF Leakage

In this part, we evaluate what happens in the absence of other transmitters and receivers. We start by examining the variation of the SNR of RF leakage versus distance.

We use one MacBook Pro laptop as an eavesdropper and a USRP as a Ghostbuster receiver. We place the eavesdropper at a total of 105 different locations at distances varying from 1 m to 7 m. We collect samples over a window of 1 sec and take an FFT over all the samples. We repeat the same experiment with a USRP eavesdropper set to listen at 5.745 GHz. We place it at a total of 210 locations at distances ranging from 1 m to 14 m and use an FFT window on 10 ms. Figure 5.1 and 5.2 show the variation of the leakage SNR versus distance for the two cases. As expected the SNR decreases with distance, but even at 14 meters the leakage SNR from the USRP is around 10 dB and can be accurately detected. Similarly, the leakage SNR from the WiFi card is around 16 dB at a distance of 7 m. However, WiFi cards require a much longer FFT window of 1 sec to achieve such SNR. Note that due to multipath, the SNR in different locations at the same distance can vary by as much as 18 dB.

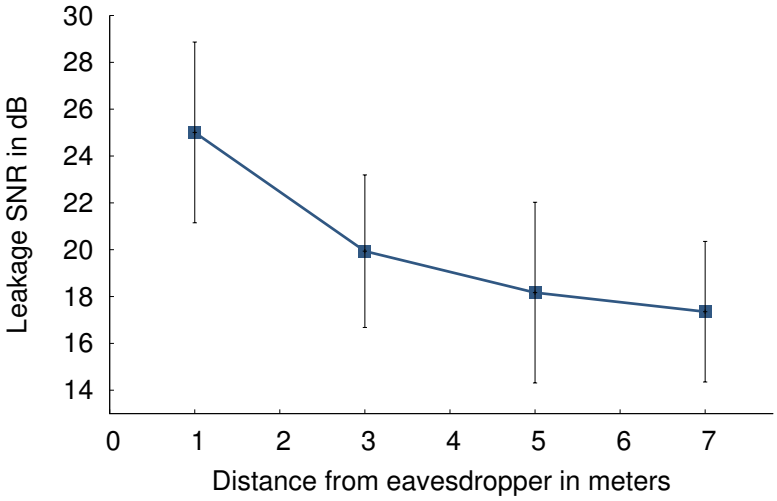

Figure 5.1: SNR in dB versus Ghostbuster's distance from a Wifi card eavesdropper with FFT window size of 1 sec.

Next we wish to examine how small an FFT window we can use while accurately detecting the presence of the eavesdropper. For, this we did ex-

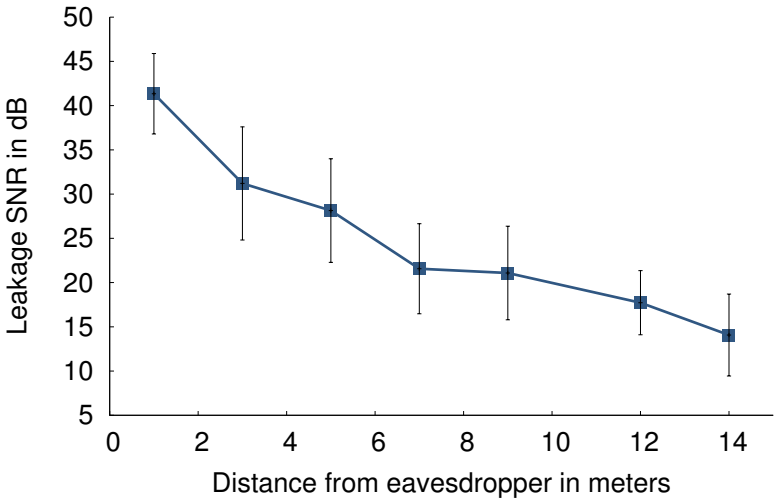

Figure 5.2: SNR in dB versus Ghostbuster's distance from a USRP eavesdropper with FFT window size of 10 ms.

periments with a MacBook Pro laptop and a USRP at 1 m and 5 m away from Ghostbuster. We collected measurements at 30 locations for both. We start from a small FFT window of length 10 μs and increase it till 1 s. We classify an eavesdropper to be present if the SNR per FFT bin is more than 6 dB. Using this threshold, we computed Ghostbuster's hit rate in detecting the eavesdropper versus the FFT window size at 1 m as shown in Figure 5.3 and 5 m as shown in Figure 5.4. The results show that performance of eavesdropper detection improves for both USRP and MacBook as the FFT window is increased, and can reach 100% as the window length reaches 100 ms at a distance of 1 m and 1 sec at a distance of 5 m.

5.2 Detection in the Presence of Ongoing Transmissions

In this part, we evaluate what happens in the presence of ongoing transmissions. We vary the number of MIMO receivers on Ghostbuster between 2, 3 and 4. We conducted experiments by placing the eavesdropper at 350 different locations at distances varying from 1 m to 5 m from Ghostbuster. We run Ghostbuster's algorithm over an FFT window of 5 ms. We compute the false positives and false negative rates in 4 cases:

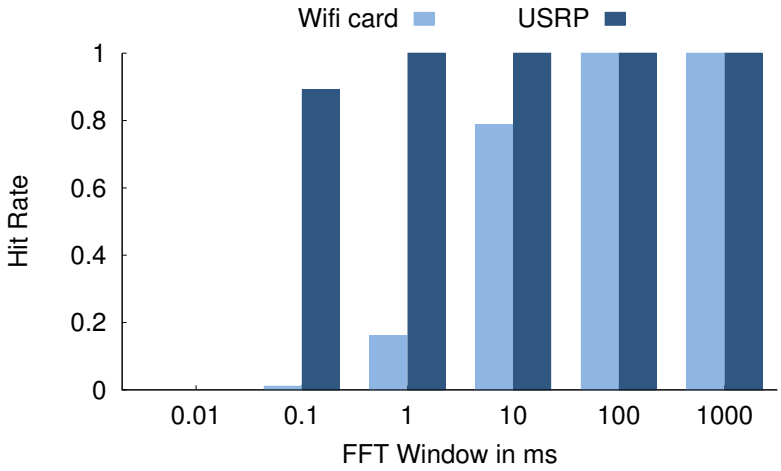

Figure 5.3: Hit rate for WiFi card and USRP eavesdroppers versus FFT window size when the eavesdropper is placed 1 m away from Ghostbuster.

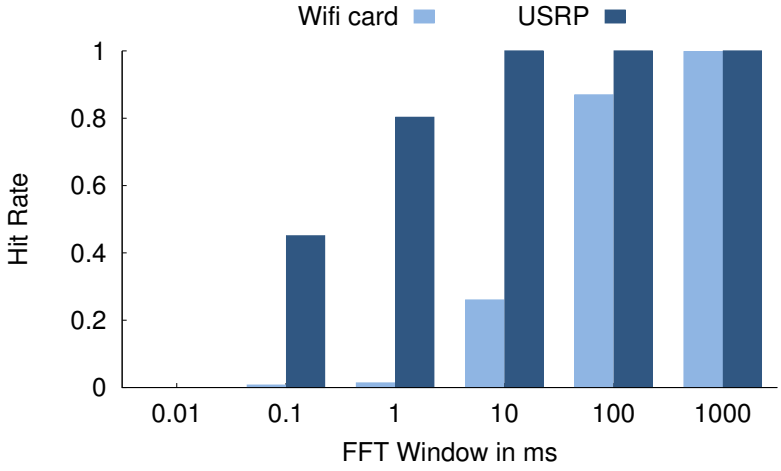

Figure 5.4: Hit rate for WiFi card and USRP eavesdroppers versus FFT window size when the eavesdropper is placed 5 m away from Ghostbuster.

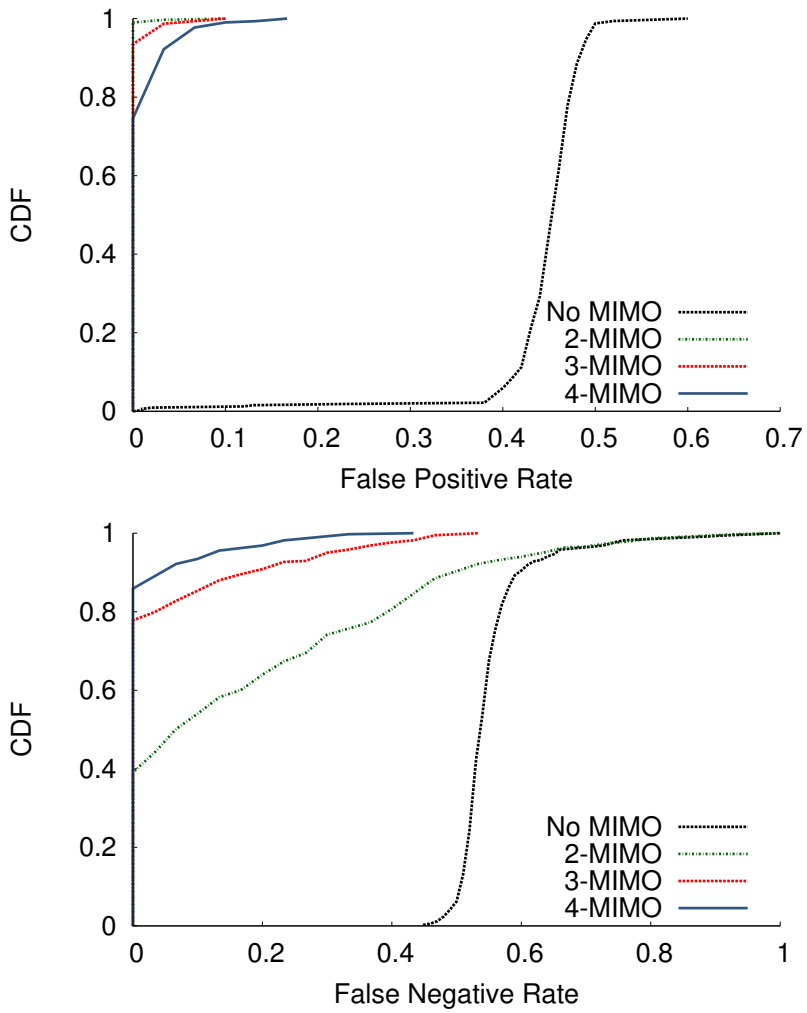

Figure 5.5: CDF of false positive and false negative rates showing impact of MIMO chains on Ghostbuster's performance.

- No MIMO: Ghostbuster has a single receiver
- 2 MIMO receivers at Ghostbuster
- 3 MIMO receivers at Ghostbuster
- 4 MIMO receivers at Ghostbuster

Figure 5.5 shows the cumulative distributions of false positive and false negative rates obtained for the 4 cases. As expected in the absence of MIMO, it is hard to separate the eavesdropper's leakage from the transmitter's signal especially in the DC bin, and hence the false negative and false positive rates

are roughly 50% which is no better than a random guess as to whether the eavesdropper is there or not. Just by using 2 MIMO already, the results improve a lot with median false positives of zero and median false negatives of 10%. As we add more MIMO chains, the median false negative rate goes down to zero for 3 MIMO and 4 MIMO.

An interesting observation, however, arises by looking at the 90th percentile. The 95th percentile false negative rate is 70%, 20% and 10% for 2 MIMO, 3 MIMO and 4 MIMO respectively. The 95th percentile false positive rate is 0%, 0.1% and 3% for 2 MIMO, 3 MIMO and 4 MIMO respectively. This can be a bit counterintuitive.

Adding more MIMO chains does reduce the false negative rate since it provides better separation in the higher dimensional antenna space. By projecting onto a space orthogonal to the transmitter's signal, the transmitter's signal is nulled and the eavesdropper's leakage is revealed. With larger MIMO, the extra dimensions in the orthogonal space amplify the leakage and the probability of missing the presence of an eavesdropper (i.e. false negatives) decreases. On the other hand, in the absence of an eavesdropper, errors in channel estimation would cause an imperfect nulling of the transmitter's signal and a residual error in the orthogonal space. With larger MIMO, the extra dimensions in the orthogonal space would also amplify this residual error and hence the false positives would increase.

Luckily, our results in Figure 5.5 show that the increase in false positives is tolerable and the gains in decreasing false negatives that come from using larger MIMO are much more significant. This can be better understood by examining the receiver operating characteristic (ROC) curve which shows the variation in true positive rate (1- false negative rate) versus the false positive rate as we sweep the detection threshold. Figure 5.6 shows the ROC curve for the same experiment. We have zoomed the ROC curve to better visualize the result. As can be seen, larger MIMO provides a better ROC curve with higher true positive rate and lower false positive rate. This result shows that the rate at which MIMO helps improve the eavesdropper's leakage is much higher than the rate at which it increases the residual error from nulling.

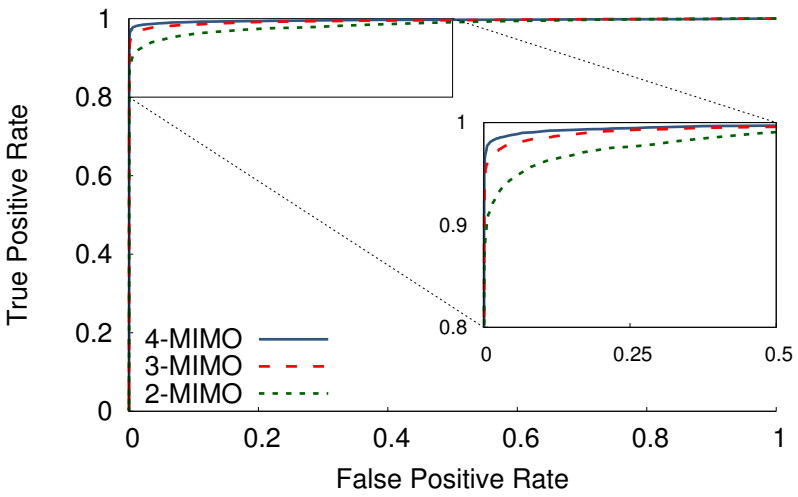

Figure 5.6: ROC curves for varying MIMO chain length.

5.3 Detection in the Presence of Other Receivers and Ongoing Transmissions

So far we have discussed recovering the leakage of an eavesdropper in the presence of ongoing transmissions. In this section, we focus on the scenario when other receivers are present. Our goal is to estimate the correct number of receivers in order to be able to identify whether an eavesdropper is there or not.

In our experiments, we took 4 USRP N210 radios, one of them being an eavesdropper and the other three being legitimate receivers. The radios were placed at 200 different locations with distances varying from 1 m to 5 m from Ghostbuster. In each location, we ran Ghostbuster's algorithm over an FFT window of 750 ms using a 2 MIMO receiver.

Figure 5.7 shows the confusion matrix of overall classification probability of detecting i given the presence of j receivers for distances varying from 1 to 5m. The figure shows that the classification accuracy is above 95% for 0, 1, and 2 receivers and remains above 89% for 4 receivers.

We repeat the above experiment for WiFi cards. The laptops are placed 1 m away from Ghostbuster and an FFT window of 1.25 s is used with a 2 MIMO receiver. We vary the number of cards between 0, 1, and 2. Figure 5.8 represents the confusion matrix for WiFi cards. For 1 card, the accuracy of detection is about 92% and drops to 89% for 2 cards. This is

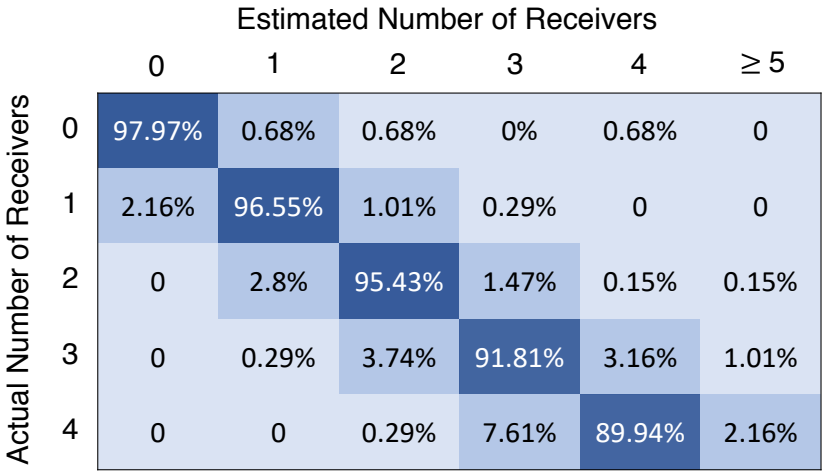

Figure 5.7: Confusion matrix of classification probabilities obtained on experiments on USRP receivers in the range 1 m to 5 m.

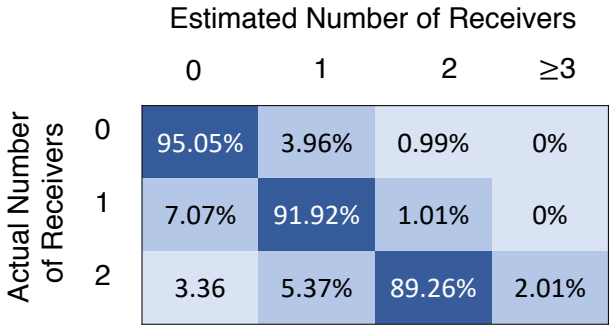

Figure 5.8: Confusion matrix of classification probabilities obtained on experiments on WiFi cards.

expected since as we have seen, the leakage from WiFi card is much smaller. This result can potentially be improved by adding more MIMO chains and increasing the FFT window. However, this would require handling an even larger computational load.

Finally, we examine the overall accuracy for detecting an eavesdropper using a COTS WiFi card. We ran an experiment with one transmitter, one legitimate receiver using a MacBook laptop and one eavesdropper using another MacBook laptop. We vary the size of the FFT window and compute the detection accuracy which incorporates both false positives and false negatives. Figure 5.9 shows the detection accuracy versus the window size. As

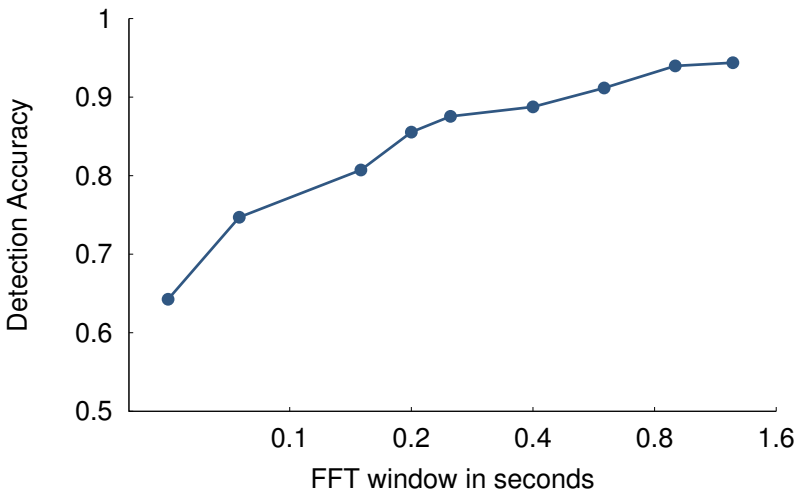

Figure 5.9: Detection accuracy achieved for WiFi cards versus changing FFT window length.

can be seen, as we increase the window size the accuracy increases, and for a window size of 1.25 sec, Ghostbuster can achieve a detection accuracy of 94%.

Chapter 6

LIMITATIONS AND DISCUSSION

Ghostbuster takes the first steps towards detecting the presence of passive eavesdroppers. However, there are several limitations that require further work before such system can be used in practice.

• **Identifying the Eavesdropper's Leakage:** A current limitation of Ghostbuster is its inability to distinguish the leakage of a legitimate receiver from the leakage of an eavesdropper. Hence, to be able to detect the presence of an eavesdropper, our current threat model assumes that Ghostbuster knows the number of legitimate receivers in the environment. By counting the number of receivers, Ghostbuster can detect the presence of an eavesdropper. While this is a strong assumption, we do believe the current system can still be useful for certain highly secured facilities where the number of receivers is known a priori and the mere presence of an additional receiver would trigger an alarm.

To address this limitation in future work, one can potentially leverage the fact that legitimate receivers constantly transition between transmitting and receiving signals. An eavesdropper, on the other hand, is passive and does not transmit packets to avoid being detected. By correlating the receivers' leakage with the transmitters' leakage in the DC OFDM bin, one can potentially discover which receivers never transmit and identify them as the eavesdroppers. This idea, however, requires significant research and is left for future work.

• **Detection Range:** Ghostbuster's current detection range in the presence of ongoing transmissions is limited to 5 meters for software-defined radios and 1 meter for COTS WiFi cards. These ranges are significantly lower than the current range of WiFi transmissions. One solution is to deploy several Ghostbuster receivers in the environment to ensure coverage. Alternatively,

for certain applications, one can potentially decrease the transmit power to ensure that the detection range and transmission range are similar and detection is possible within the transmission range.

The detection range, however, can be increased by taking larger time windows as we have shown in Chapter 5. In our experiments, the maximum time window used is 1.25 sec due to our hardware constraints. While increasing the time window would have improved the range, it comes at the cost of high computational complexity as we discuss next.

• **Computational Overhead:** Ghostbuster's algorithm must process samples from a very large time window. The larger the time window, the higher the detection accuracy, but this comes at the cost of higher computational overhead. In our current implementation, we can process time windows up to 1.25 sec with 25 million samples on an Intel i7 machine with 16 GB memory in 30 secs. The main two sources of computational complexity are the gradient descent optimization and the computation of a several million-point FFT. Luckily, both are highly parallelizable. A more efficient implementation can potentially reduce the processing time. However, the trade-off between detection accuracy and computational overhead will remain.

• **Packet Collisions and MIMO Transmitters:** Our current evaluation assumes that the received transmissions come from a single antenna transmitter and have not experienced collisions. Some WiFi devices, however, do use MIMO and their packets can experience collisions. To address such cases, Ghostbuster must use more MIMO antennas to separate the signals in the spatial domain. Specifically, an n-antenna MIMO receiver can decode n signals in parallel. Hence, for k antenna transmitter or a collision of k packets, Ghostbuster needs $k+1$ antennas to be able to project on a space orthogonal to all transmissions in order to null them and reveal the eavesdropper.[1]

[1]In certain cases of packet collisions, the packet preambles might overlap and prevent us from properly estimating the channels of the transmitters. In such cases, we would need to rely on statistical techniques like PCA or ICA to separate the transmissions.

Chapter 7

RELATED WORK

7.1 Eavesdropper Detection

Detecting eavesdroppers has been studied in the literature. These studies, however, have been largely analytical. The closest to our work is a theoretical work on detecting eavesdroppers from RF leakage [10]. This work, however, requires all RF devices in the vicinity to periodically pause communication so they can sense a "clear" channel, and thereby listen for the leakages from passive eavesdroppers. Unfortunately, this is not practical in real settings, since other wireless transmitters and receivers in the vicinity may not turn off. Ghostbuster, in contrast, can detect eavesdroppers in the presence of other transmissions without requiring any modifications to the transmitters. Ghostbuster is also implemented and empirically tested.

In [11], the authors propose a method for detecting eavesdroppers in the context of near-field inductively-coupled communication, e.g. RFID based smart cards. Specifically, the inductive coupling channel can be computed using the relative-geometry and the properties of the transmitter and receiver. An eavesdropper in the vicinity would also couple with the transmitter and receiver and hence change the channel. The change in channel can be used as an indicator of the eavesdropper's presence. Unfortunately, in our context the communication is far-field and thus the presence of an eavesdropper does not change the wireless channel between the transmitter and the receiver.

7.2 Radio Detection

RF leakage has also been studied in the context of cognitive radio networks. Cognitive radios need to detect the presence of "primary" devices; such de-

tection can be valuable for the "secondary" device to back out and avoid interfering with "primary" devices. A body of work in this domain has proposed theoretical analysis [12, 13] on the achievable SNR of leakage signals and detection range. In [14], the authors propose similar results in the context of WiMAX and UWB co-existence. Reference [13] demonstrates the feasibility of detecting the leakage by connecting the output of a TV tuner to a light diode configured to detect the desired frequency. Reference [15] also shows the possibility of detecting leakage from USRP B210 up to 50 cm. All the above work, however, assumes a single "primary" receiver and no transmissions making the problem relatively easy. The presence of transmissions would negate the need for detecting RF leakage since in the context of cognitive radios, if the channel is not idle, a "secondary" device must switch to a different channel. The core problem formulation in Ghostbuster, on the other hand, is adversarial. Hence, Ghostbuster must continue to detect RF leakage in the presence of ongoing transmissions.

TV detector vans have been used by the BBC channel in the UK to identify users who are not paying the license fees but still tune their TV to the BBC channel. It has been speculated that this is done by detecting the RF leakage from the TV tuner. However, BBC refuses to disclose any information on how the vans work. Furthermore, recent discoveries suggest that this is simply a PR stunt and there is no evidence that these detectors actually work [16, 17, 18]. In fact, the detectors have never been used to prosecute any of the people who did not pay the license fees [18].

A body of work aims to detect the presence of radio receivers in the context of remotely triggered explosives [19, 20, 21, 22, 23]. The work proposes actively transmitting a known stimulation signal that triggers the receiver circuit to reflect unintended electromagnetic transmissions. By using FMCW radar signals as the stimulus, [20, 21] can further range the receiver's location. However, all this work assumes a super-heterodyne or a super-regenerative receiver architecture which are far less common in WiFi cards as can be seen in Tables 2.1 and 2.2. The work has been experimentally tested only for frequencies < 500 MHz where signals can propagate farther. Furthermore, transmitting the stimulation signals requires halting ongoing communication to avoid interference. On the other hand, Ghostbuster leverages an orthogonal passive approach that focuses on WiFi communication and does not require transmitting a stimulation signal. Ghostbuster has also been demon-

strated to work for direct conversion receiver architectures, in the presence of ongoing transmissions and for frequencies up to 5.7 GHz.

RF and EM leakage has also been used to launch side-channel attacks. In [24, 5], the authors detected low-frequency EM leakage from smart-cards and ultimately demonstrate that cryptographic keys can be completely deciphered from this leakage. The work [25, 2] also shows that EM leakage from powerful computers can reveal the programs running on them.

7.3 Leakage Suppression

Finally, we are aware of arguments that suggest that leakages can be suppressed, either through modifications in the circuit design, or via physical packaging and shielding [26, 27, 28]. However, the commercial off-the-shelf (COTS) wireless devices do not employ leakage cancellation circuits, neither is there any special shielding to the best of our knowledge. As discussed earlier, we have successfully detected leakage from various RF devices across different vendors. The work mentioned in this thesis aims at thwarting attacks with such COTS devices.

Chapter 8

CONCLUSIONS

This thesis takes the first practical steps toward developing systems that can detect the hidden presence of eavesdroppers. Our results show that one can reliably detect the presence of an eavesdropper up to five meters, even in the presence of other receivers and ongoing transmissions. One can potentially push these results further with more computational power that would allow one to compute even larger FFTs. We believe that such capability can serve as a strong primitive that provides a defense-in-depth against eavesdropping.

REFERENCES

[1] M. Vanhoef and F. Piessens, "Key reinstallation attacks: Forcing nonce reuse in wpa2," in *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*. ACM, 2017, pp. 1313–1328.

[2] D. Genkin, L. Pachmanov, I. Pipman, and E. Tromer, "Stealing keys from PCs using a radio: Cheap electromagnetic attacks on windowed exponentiation," in *International Workshop on Cryptographic Hardware and Embedded Systems*. Springer, 2015, pp. 207–228.

[3] D. Genkin, L. Pachmanov, I. Pipman, A. Shamir, and E. Tromer, "Physical key extraction attacks on pcs," *Communications of the ACM*, vol. 59, no. 6, pp. 70–79, 2016.

[4] D. Genkin, A. Shamir, and E. Tromer, "RSA key extraction via low-bandwidth acoustic cryptanalysis," in *International cryptology conference*. Springer, 2014, pp. 444–461.

[5] D. Genkin, I. Pipman, and E. Tromer, "Get your hands off my laptop: Physical side-channel key-extraction attacks on PCs," *Journal of Cryptographic Engineering*, vol. 5, no. 2, pp. 95–112, 2015.

[6] S. Gollakota, H. Hassanieh, B. Ransford, D. Katabi, and K. Fu, "They can hear your heartbeats: non-invasive security for implantable medical devices," in *ACM SIGCOMM Computer Communication Review*, vol. 41, no. 4. ACM, 2011, pp. 2–13.

[7] K. Koscher, A. Juels, V. Brajkovic, and T. Kohno, "EPC RFID tag security weaknesses and defenses: passport cards, enhanced drivers licenses, and beyond," in *Proceedings of the 16th ACM conference on Computer and communications security*. ACM, 2009, pp. 33–42.

[8] R. Verdult, F. Garcia, and J. Balasch, "Gone in 360 secs: Hijacking with hitag2," in *Usenix Security*, 2012.

[9] H. Hassanieh, J. Wang, D. Katabi, and T. Kohno, "Securing rfids by randomizing the modulation and channel," in *NSDI*, 2015, pp. 235–249.

[10] A. Mukherjee and A. L. Swindlehurst, "Detecting passive eavesdroppers in the MIMO wiretap channel," in *Acoustics, Speech and Signal Processing (ICASSP), 2012 IEEE International Conference on*. IEEE, 2012, pp. 2809–2812.

[11] L. R. Varshney, P. Grover, and A. Sahai, "Securing inductively-coupled communication," in *Information Theory and Applications Workshop (ITA), 2012*. IEEE, 2012, pp. 47–53.

[12] S. Park, L. E. Larson, and L. B. Milstein, "An RF receiver detection technique for cognitive radio coexistence," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 57, no. 8, pp. 652–656, 2010.

[13] B. Wild and K. Ramchandran, "Detecting primary receivers for cognitive radio applications," in *New Frontiers in Dynamic Spectrum Access Networks, 2005. DySPAN 2005. 2005 First IEEE International Symposium on*. IEEE, 2005, pp. 124–130.

[14] S. Park, L. E. Larson, and L. B. Milstein, "Hidden mobile terminal device discovery in a UWB environment," in *Ultra-Wideband, The 2006 IEEE 2006 International Conference on*. IEEE, 2006, pp. 417–421.

[15] J. C. Manco Vásquez, J. M. Ibáñez Díaz, J. Vía Rodríguez, L. I. Santamaría Caballero et al., "Detection of radio receivers: An experimental evaluation approach," University of Cantabria, 2015.

[16] B. Dowell, "Do TV detector vans exist? Leaked document suggests they could be a ruse," *Radio Times*, September 2013.

[17] C. Carter, "Myth of the TV detector van?" *The Telegraph*, September 2013.

[18] G. Adams, "Are TV detector vans just a cunning con trick?" *Daily Mail*, October 2013.

[19] V. Thotla, M. T. A. Ghasr, M. J. Zawodniok, S. Jagannathan, and S. Agarwal, "detection of super-regenerative receivers using Hurst parameter," *IEEE Transactions on Instrumentation and Measurement*, vol. 62, no. 11, pp. 3006–3014, Nov 2013.

[20] C. Stagner, A. Conrad, C. Osterwise, D. G. Beetner, and S. Grant, "A practical superheterodyne-receiver detector using stimulated emissions," *IEEE Transactions on Instrumentation and Measurement*, vol. 60, no. 4, pp. 1461–1468, 2011.

[21] C. Stagner, M. Halligan, C. Osterwise, D. G. Beetner, and S. L. Grant, "Locating noncooperative radio receivers using wideband stimulated emissions," *IEEE Transactions on Instrumentation and Measurement*, vol. 62, no. 3, pp. 667–674, 2013.

[22] S. A. Seguin, "Detection of low cost radio frequency receivers based on their unintended electromagnetic emissions and an active stimulation," Ph.D. dissertation, Missouri University of Science and Technology, 2009.

[23] C. Stagner, "Detecting and locating electronic devices using their unintended electromagnetic emissions," Ph.D. dissertation, Missouri University of Science and Technology, 2013.

[24] K. Gandolfi, C. Mourtel, and F. Olivier, "Electromagnetic analysis: Concrete results," in *Cryptographic Hardware and Embedded Systems, CHES 2001*. Springer, 2001, pp. 251–261.

[25] A. Zajic and M. Prvulovic, "Experimental demonstration of electromagnetic information leakage from modern processor-memory systems," *IEEE Transactions on Electromagnetic Compatibility*, vol. 56, no. 4, pp. 885–893, 2014.

[26] S. Jayasuriya, D. Yang, and A. Molnar, "A baseband technique for automated lo leakage suppression achieving¡- 80dbm in wideband passive mixer-first receivers," in *Custom Integrated Circuits Conference (CICC), 2014 IEEE Proceedings of the*. IEEE, 2014, pp. 1–4.

[27] B. Lindqvist and M. Isberg, "Homodyne receiver minimizing oscillator leakage," June 25 1996, US Patent 5,530,929.

[28] G. K. Kannell, R. E. Myer, and K. Sreenath, "Local oscillator leak cancellation circuit," Dec. 26 2000, US Patent 6,167,247.

APPENDIX A: PROOF OF THEOREM 1

Theorem 1. *The error function $E(\tilde{f}_k, \tilde{a}_k)$ is convex for $\tilde{f}_k \in [f_k - \alpha, f_k + \alpha]$ for any $\alpha < 2/5$.*

Proof. Recall that the error function $E(\tilde{f}_k, \tilde{a}_k)$ is defined as:

$$E(\tilde{f}_k, \tilde{a}_k) = \sum_{t=0}^{N-1} \left| a_k e^{j2\pi f_k t/N} - \tilde{a}_k e^{j2\pi \tilde{f}_k t/N} \right|^2 \tag{A.1}$$

We wish to find the $\tilde{f}_k$ and $\tilde{a}_k$ that minimize the error. For a fixed $\tilde{f}_k$, the $\tilde{a}_k$ that minimizes the error can be obtained by solving a weighted least squares problem. This is done by projecting the given samples on the vector of $e^{j2\pi \tilde{f}_k t/N}$. Thus,

$$\begin{aligned} \tilde{a}_k = & \quad \arg\min_{\tilde{a}_k} E(\tilde{f}_k, \tilde{a}_k) \\ = & \quad \frac{1}{N} \sum_{t=0}^{N-1} a_k e^{j2\pi (f_k - \tilde{f}_k) t/N} \\ = & \quad \frac{a_k}{N} \frac{e^{j2\pi (f_k - \tilde{f}_k)} - 1}{e^{j2\pi (f_k - \tilde{f}_k)/N} - 1} \end{aligned} \tag{A.2}$$

Given the above solution for $\tilde{a}_k$, we now show the error function is convex in $\tilde{f}_k$. We can expand the error function as follows:

$$\begin{aligned} E(\tilde{f}_k, \tilde{a}_k) = & \sum_{t=0}^{N-1} \left| a_k e^{j2\pi f_k t/N} \right|^2 + \sum_{t=0}^{N-1} \left| \tilde{a}_k e^{j2\pi \tilde{f}_k t/N} \right|^2 \\ & - \sum_{t=0}^{N-1} a_k \tilde{a}_k^* e^{j2\pi (f_k - \tilde{f}_k) t/N} - \sum_{t=0}^{N-1} a_k^* \tilde{a}_k e^{-j2\pi (f_k - \tilde{f}_k) t/N} \\ \\ & = N|a_k|^2 + N|\tilde{a}_k|^2 \\ & - a_k \tilde{a}_k^* \frac{e^{j2\pi (f_k - \tilde{f}_k)} - 1}{e^{j2\pi (f_k - \tilde{f}_k)/N} - 1} - a_k^* \tilde{a}_k \frac{e^{-j2\pi (f_k - \tilde{f}_k)} - 1}{e^{-j2\pi (f_k - \tilde{f}_k)/N} - 1} \end{aligned}$$

where $(*)$ is the complex conjugate operator. We can then replace $\tilde{a}_k$ from Equation (A.2) to get:

$$\begin{aligned} E(\tilde{f}_k, \tilde{a}_k) = & \quad N|a_k|^2 + |a_k|^2/N \left| \frac{e^{j2\pi(f_k - \tilde{f}_k)} - 1}{e^{j2\pi(f_k - \tilde{f}_k)/N} - 1} \right|^2 \\ - & \quad 2|a_k|^2/N \left| \frac{e^{j2\pi(f_k - \tilde{f}_k)} - 1}{e^{j2\pi(f_k - \tilde{f}_k)/N} - 1} \right|^2 \\ = & \quad |a_k|^2 N - |a_k|^2/N \left(\frac{\sin(\pi(f_k - \tilde{f}_k))}{\sin(\pi(f_k - \tilde{f}_k)/N)} \right)^2 \end{aligned}$$

Thus, the error function is the negative of the square of discrete sinc function. Figure 3.4 shows an example of the error function. By taking the second derivative of $E(\tilde{f}_k, \tilde{a}_k)$ with respect to $\tilde{f}_k$, we can show that $\forall N$,

$$\frac{\partial^2 E}{\partial \tilde{f}_k^2} > 0 \quad \forall \tilde{f}_k \in [f_k - 0.4, f_k + 0.4] \tag{A.3}$$

Hence, the error function $E(\tilde{f}_k, \tilde{a}_k)$ is convex in $\tilde{f}_k$ within the interval $[f_k - 0.4, f_k + 0.4]$ around f_k. It is sufficient to ensure the initial values of $\tilde{f}_k$ are within this interval to guarantee that the gradient descent converges to the optimal minimum of the error function. □