Fingerprinting Far Proximity from Radio Emissions*

Tao Wang, Yao Liu, and Jay Ligatti

University of South Florida, Tampa, FL 33620, USA
taow@mail.usf.edu, {yliu, ligatti}@cse.usf.edu

Abstract. As wireless mobile devices are more and more pervasive and adopted in critical applications, it is becoming increasingly important to measure the physical proximity of these devices in a secure way. Although various techniques have been developed to identify whether a device is close, the problem of identifying the *far proximity* (i.e., a target is at least a certain distance away) has been neglected by the research community. Meanwhile, verifying the far proximity is desirable and critical to enhance the security of emerging wireless applications. In this paper, we propose a secure far proximity identification approach that determines whether or not a remote device is far away. The key idea of the proposed approach is to estimate the far proximity from the unforgeable "fingerprint" of the proximity. We have validated and evaluated the effectiveness of the proposed far proximity identification method through experiments on real measured channel data. The experiment results show that the proposed approach can detect the far proximity with a successful rate of 0.85 for the non-Line-of-sight (NLoS) scenario, and the successful rate can be further increased to 0.99 for the Line-of-sight (LoS) scenario.

1 Introduction

As mobile platforms are more and more pervasive and adopted in critical applications, it is becoming increasingly important to measure the physical proximity of mobile devices in a secure way. For example, Implantable Medical Devices (IMDs) like pacemakers may grant access to an external control device only when that device is close enough [25]. As another example, contactless-payment systems (like Google Wallet), which enable users to make payments by placing a mobile device in the close proximity of a payment terminal, may require the mobile devices to be within several centimeters or even millimeters of the payment terminals.

Thus, verifying the *close proximity* has triggered significant attention and activity from the research community, and multiple techniques have been proposed to achieve the efficient identification of close proximity (e.g., [5, 7, 11, 12, 16, 24, 29]), including the well-known distance bounding protocols and their variants (e.g., [5, 24, 29]).

Although various techniques have been developed to identify whether a device is close, the problem of identifying the *far proximity* (i.e., a target is at least a certain distance away) has been neglected by the research community. Meanwhile, verifying the far proximity is desirable and critical to enhance the security of emerging wireless applications. By enforcing far proximity, in addition to traditional access control

* This work was supported by grant W911NF-14-1-0324 from Army Research Office (ARO)

and cryptographic approaches, we can enhance the security of various critical wireless applications, such as satellite communication, long-haul wireless TV, radio, and alarm broadcasting, and Marine VHF radio for rescue and communication services [2].

For example, GPS devices receive signals, presumably from satellites in space, to determine their locations. Ideally, the GPS devices could verify that received signals are from far-away sources, to avoid being deceived by a nearby adversary's signals. In cellular networks, mobile phones may at times expect to receive signals from particular cell towers. It has been demonstrated that adversaries can set up a fake short-range cell tower to fool nearby mobile phones [21,31]. To avoid being deceived by such a fake cell tower, it is desirable that mobile phones can authenticate that the signals they receive originate from a tower at an expected, further distance away.

Existing close proximity identification techniques (e.g., [7,11,16]) qualitatively decide whether or not a target is nearby, but they cannot be directly extended to address the far proximity identification problem. The qualitative decision that a target is not nearby doesn't quantitatively guarantee that the target is at least a certain distance away (i.e., in the far proximity).

Distance bounding protocols (e.g., [5, 24, 29] demonstrated their success in quantitatively estimating the distance between two wireless devices. However, they cannot be directly applied to enforce far proximity identification. In distance bounding protocols, a local device sends a challenge to a remote device, and the remote device replies with a response that is computed as a function of the received challenge. The local device then measures the round-trip time between sending its challenge and receiving the response, subtracts the processing delay from the round-trip time, and uses the result to calculate the distance between itself and the remote device. However, by delaying its response to a challenge, a dishonest remote device can appear to be arbitrarily further from the local device than it actually is.

In this paper, we develop a secure far proximity identification approach that can determine whether a remote device is far away. The key idea of the proposed approach is to estimate the proximity from the unforgeable "fingerprint" of the proximity. We develop a technique that can extract the fingerprint of a wireless device's proximity from the physical-layer features of signals sent by the device. The proximity fingerprints are closely related to the distance between the local and remote devices. They are easy to extract but difficult to forge. We also develop a novel technique that uses the proximity fingerprint to identify the lower bound of the distance between the local and the remote devices.

The contributions of this paper are: (1) we develop a novel fingerprinting technique that enables the local device to extract the fingerprint of a wireless device's proximity from the physical-layer features of signals sent by the device; (2) we discover the theoretical relationship between the proximity and its fingerprint, and we developed a technique that can use such a relationship to estimate the lower bound of the distance between the local and remote devices; and (3) we validate and evaluate the effectiveness of the proposed far proximity identification method through experiments on the real-world data. The experiment results show that the proposed approach can detect the far proximity with a success rate of 0.85 for the non-Line-of-sight (NLoS) scenario, and the success rate can be further increased to 0.99 for the Line-of-sight (LoS) scenario.

The rest of the paper is organized as follows. Section 2 describes our assumptions and system and threat models. Section 3 presents the proposed far proximity identification techniques. Sections 4 and 5 discuss the experimental evaluation and related work. Section 6 concludes this paper.

2 System and threat models

To facilitate the presentation, we refer to the local device, which verifies the proximity, as the *verifier* and the remote device, whose proximity is being verified, as the *prover*. The verification system consists of a verifier and a prover. Both are equipped with radio interfaces that can transmit and receive wireless signals.

The verifier aims to determine whether or not a prover is at least a certain distance away, and it analyzes the signals emitted by the prover to achieve this goal. The verifier can work in both *active* or *passive* modes. In the active mode, the verifier sends a message to the prover to initialize the proximity identification, and the prover cooperates with the verifier by sending wireless signals back to the verifier to enable the verification. In the passive mode, instead of actively sending out signals, the verifier monitors the wireless channel to capture the prover's signal. Once the prover's signals are captured, the verifier can identify the prover's proximity.

We assume that the prover is untrusted. The prover may provide the verifier with fake messages and wrong configuration information regarding its hardware and software settings, such as device type, signal processing delay, and protocols in use. The prover may intentionally delay its replies to the verifier's messages or send bogus replies at any time to mislead the verifier. However, we assume that the verifier can receive wireless signals sent by the prover. We assume that there are no metal shields on the straight line between the verifier and the prover to block wireless signals from the prover.

3 Far Proximity Verification

A simple and naive method to identify whether a prover is far away is to examine the received signal strength (RSS). A signal decays as it propagates in the air. Thus, it seems that strong RSS indicates a short signal propagation length and a close transmitter, whereas weak RSS strength implies a far-away transmitter. However, a dishonest prover can increase or decrease its transmit power to pretend to be close to, or far from, the verifier. The root reason for the failure of the naive method is that RSS can be easily forged. In this paper, we discover unforgeable and unclonable *fingerprints* of the proximity and propose techniques that can identify the far proximity based on these fingerprints.

3.1 Proximity Fingerprints

Because of the multipath effect [9], a signal sent by the prover generally propagates to the verifier in the air along multiple paths due to reflection, diffraction, and scattering. Each path has an effect (e.g., distortion and attenuation) on the signal traveling

on it [23]. A *channel impulse response* characterizes the overall effects imposed by the multipath propagation, and it reflects the physical feature of a wireless link [9]. Because it is difficult to change the physical feature, channel impulse responses have been used as "**link signatures**" to uniquely identify the wireless link between a wireless transmitter and a receiver [6, 23, 33].

Figure 1 (a) shows a simple example of multipath propagation. The signal sent by the prover is reflected by an obstacle (i.e., a building), and thus it travels along Path 1 (the direct path from the prover to the verifier), and Path 2 (the reflection path). The signal copy that travels along one path is usually referred to as a *multipath component* [9]. Let $r1$ and $r2$ denote the multipath components that travel along Path 1 and Path 2 respectively. Figure 1 (b) is an example of the corresponding channel impulse response, which shows that $r1$ arrives at the verifier first and the peak of the signal amplitude of $r1$ is A_{r1}, and $r2$ arrives after $r1$, and its peak is A_{r2}.

Intuitively, if the prover increases (decreases) the transmit power, both A_{r1} and A_{r2} will increase (decrease), but the prover cannot adjust its transmit power such that it arbitrarily manipulates only one of A_{r1} and A_{r2}, because it is difficult for the prover to identify and modify the physical paths over which multipath components propagate [23]. On the other hand, the length of the signal propagation path is closely related to the amplitude of the received signal. A far-away prover results in weaker A_{r1} and A_{r2} than a close prover. Based on this intuition, we give the definition of proximity fingerprint below.

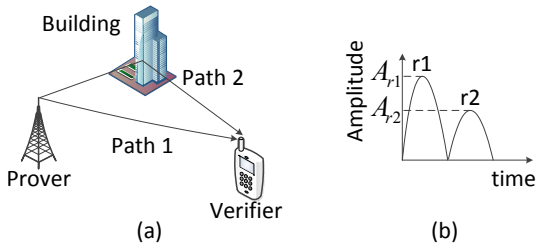

Fig. 1. An example of the multipath effect.

Definition 1 (Proximity Fingerprint) *Let A_{r1} and A_{r2} be the amplitudes of the first and the second received multipath components, respectively. The proximity fingerprint f is the ratio of A_{r1} to A_{r2}, i.e., $f = \frac{A_{r1}}{A_{r2}}$.*

Key Features of Proximity Fingerprints: It appears that an attacker (i.e., a dishonest prover or a third-party adversary against benign provers) could affect the proximity fingerprint by intentionally placing a reflector nearby the prover to generate a fake path, in addition to the direct signal path from the prover to the verifier.

However, at the verifier's view, the direct and fake paths are still one unresolvable path if the difference between the arrival times of the signals traveling on both paths is much smaller than the symbol duration, which is the transmission time of a wireless physical-layer unit [9]. To be successful, an attacker has to place the reflector far enough

away from the prover (i.e., δc meters, where δ is the symbol duration and c is the speed of light [9]), such that the difference between the two path arrival times is resolvable at the verifier. More crucially, at this distance the attacker must make sure that the prover's signal can exactly hit his reflector and be bounced back to the target verifier. However, it is quite uncertain for the prover's signal to be delivered to the reflector, then reflected by the reflector to the verifier due to the random scattering effect caused by long distance propagation [9].

For example, GPS satellites have a typical symbol duration of 0.01 second [1]. It is impractical for the satellite's signal to exactly hit a reflector that is 3,000,000 meters away, and moreover be reflected by the reflector to hit a target GPS navigation device on earth.

To summarize, proximity fingerprints are caused by wireless reflections somewhere, which the verifier does not need to know and identify. The verifier can easily extract A_{r1} and A_{r2} from the channel impulse response and compute the proximity fingerprint as A_{r1}/A_{r2}. Note that estimating the channel impulse responses is a must-have function for most modern wireless systems [9, 20]. But in order for the attacker to be successfully, the attacker has to know (1) how to pinpoint a far-away place to put a reflector or an active wireless device, and (2) exactly where to direct the reflector to shoot a needle in a haystack. Thus, significant practical hurdles exist for attacking proximity fingerprints. In this way, verifiers can easily extract proximity fingerprints, but it is difficult for attackers to forge or manipulate a specific fingerprint.

The attacker may also launch active attacks to undermine the verification of proximity fingerprints. In later section (3.4), we will discuss these active attacks and the corresponding countermeasures.

Impact of Directional Antennas: When directional antennas are used, the multipath effect may be reduced. However, directional antennas cannot provide perfect laser-like radio signals. For example, the beamwidth of a 3-element Yagi Antenna, the most common type of directional antenna, is 90 degrees in the vertical plane and 54 degrees in the horizontal plane [14]. Thus, it is not possible to completely eliminate the multipath effect, and accordingly the multipath propagation has been also considered in designing wireless communication systems equipped with directional antennas (e.g., [28,32]). The proximity fingerprint can be calculated based on a very limited number of paths (i.e., two paths), and thus it is compatible to wireless systems with directional antennas in use.

3.2 Far Proximity Identification Using Proximity Fingerprints

Based on the study of proximity fingerprint, we now reveal the relationship between the proximity fingerprint and the actual proximity, and we propose far proximity identification techniques that can provide fine granularity and lower bounds on proximity (i.e., the prover is at least a certain distance away from the verifier) using the proximity fingerprint

Far Proximity Identification To calculate the proximity of the prover, we first model the fingerprint of the proximity. We consider signal propagation in two typical wireless environments, i.e., the outdoor and the indoor environments.

Outdoor signal propagation: One of the most common models for outdoor signal propagation in urban, suburban, and rural areas is the Okumura Model [9]. According to the Okumura model, the signal path loss in decibels (dB) in urban areas can be modeled as

$$L(\text{dB}) = 69.55 + 26.16\log_{10}(f_c) - 13.82\log_{10}(h_{te}) - a(h_{re}, f_c) + (44.9 - 6.55\log_{10}(h_{te}))\log_{10}(d),$$

where d is the length of the path along which the signal propagates from the transmitter to the receiver, f_c is the central frequency, h_{te} and h_{re} are the transmitter's and the receiver's antenna heights respectively, and $a(h_{re}, f_c)$ is a correction factor computed using h_{re} and f_c [9]. Based on the Okumura Model, we give Lemma 1

Lemma 1. *The proximity fingerprint in the outdoor environment is* $\sqrt{(\frac{d_2}{d_1})^{\frac{\gamma}{10}}}$*, where* d_1 *and* d_2 *are the lengths of the paths along which the first and the second received multipath components travel respectively,* $\gamma = 44.9 - 6.55\log_{10}(h_{te})$*, and* h_{te} *is the transmitter's antenna height.*

Proof: The received signal power P_r can be represented as P_r(dB) = P_t(dB) - L (dB), where P_t is the transmit power. To facilitate the calculation, we change the unit of P_r from dB to watt (W). Thus, $P_r(\text{W}) = 10^{\frac{1}{10}(P_t(\text{dB}) - L(\text{dB}))} = \frac{P_t(\text{W})}{L(\text{W})}$, and $L(\text{W}) = 10^{\frac{1}{10}L(\text{dB})} = 10^{\frac{1}{10}(\beta + \gamma\log_{10}(d))}$, where $\beta = 69.55 + 26.16\log_{10}(f_c) - 13.82\log_{10}(h_{te}) - a(h_{re}, f_c)$ and $\gamma = 44.9 - 6.55\log_{10}(h_{te})$. The amplitude of a signal is the square root of the received signal power. Accordingly, $A_{r1} = \sqrt{P_{r1}(\text{W})} = \sqrt{\frac{P_t(\text{W})}{10^{\frac{1}{10}(\beta + \gamma\log_{10}(d_1))}}}$ and $A_{r2} = \sqrt{P_{r2}(\text{W})} = \sqrt{\frac{P_t(\text{W})}{10^{\frac{1}{10}(\beta + \gamma\log_{10}(d_2))}}}$, where d_1 and d_2 are the lengths of the paths along which the first and the second received multipath components travel respectively. Note that both multipath components have the same values for γ and β, because they are from the same signal source (i.e., the prover) and exhibit the same frequency f_c. Thus, the proximity fingerprint f can be written as $f = \frac{A_{r1}}{A_{r2}} = \sqrt{(\frac{d_2}{d_1})^{\frac{\gamma}{10}}}$. According to the Okumura Model, the signal path loss models in suburban and rural areas are $L_{suburban}(\text{dB}) = L(\text{dB}) - 2[\log_{10}(f_c/28)]^2 - 5.4$ and $L_{rural}(\text{dB}) = L(\text{dB}) - 4.78[\log_{10}(f_c)]^2 + 18.33\log_{10}(f_c) - K$, respectively, where K ranges from 35.94 (countryside) to 40.94 (desert). By using the same analysis, we can obtain similar result that f in the suburban and rural areas is $\sqrt{(\frac{d_2}{d_1})^{\frac{\gamma}{10}}}$. □

Indoor signal propagation: The path loss in the indoor environment can be usually represented by the ITU Indoor Propagation Model [20] as shown below

$$L(\text{dB}) = 20\log f_c + \lambda\log d + P_f(N_f),$$

where λ is the empirical path loss at the same floor, N_f denote the number of floors between the transmitter and receiver, and $P_f(N_f)$ denotes the floor penetration loss. Based on the ITU indoor model, we give Lemma 2

Lemma 2. *The proximity fingerprint in the indoor environment is* $\sqrt{(\frac{d_2}{d_1})^{\frac{\lambda}{10}}}$*, where* d_1 *and* d_2 *are the lengths of the paths along which the first and the second received multipath components travel respectively, and* λ *is the empirical floor penetration loss factor.*

Proof: As discussed earlier, the received signal power P_r can be represented as P_r(dB) = P_t(dB) - L (dB). By converting the unit of P_r from dB to W, we can obtain $P_r(\mathrm{W}) = \frac{P_t(\mathrm{W})}{L(\mathrm{W})} = \frac{P_t(\mathrm{W})}{10^{\frac{1}{10}(20\log f_c + \lambda \log d + P_f(N_f))}}$. The proximity fingerprint, the ratio of A_{r1} to A_{r2}, can be written as $f = \frac{\sqrt{P_{r1}(\mathrm{W})}}{\sqrt{P_{r2}(\mathrm{W})}} = \sqrt{(\frac{d_2}{d_1})^{\frac{\lambda}{10}}}$ □

Far proximity identification: Assume there are no large metallic obstacles that can significantly block the signal propagation between the verifier and the prover. The path that the first received multipath component usually travels along (i.e., Path 1) is roughly straight between the verifier and the prover due to penetration and diffraction-around-obstacles features of wireless signals [9]. Thus, d_1 approximately equals to the distance between the verifier and the prover. The lower bound of d_1 is given in Lemma 3.

Lemma 3. *Let d be the distance between the prover and the verifier. We have $d \geq \frac{c}{B(f^{\frac{2}{\alpha}}-1)}$, where c is the speed of light, B is the bandwidth of the communication system, α is the path loss exponent, and f is the proximity fingerprint.*

Proof: Let t denote the time at which the prover's signal starts to propagate to the verifier. Let t_1 and t_2 denote the arrival times of the first and the second received multipath components, respectively. Therefore, $d_1 = (t_1 - t)c$ and $d_2 = (t_2 - t)c = (t_1 - t)c + (t_2 - t_1)c = d_1 + \Delta c$, where $\Delta = t_2 - t_1$. From Lemmas 1 and 2, we know that for both the outdoor and indoor environments, the proximity fingerprint f can be generalized by the same expression $f = \sqrt{(\frac{d_2}{d_1})^{\alpha}}$, where α equals to $\frac{\gamma}{10}$ and $\frac{\lambda}{10}$ for the outdoor and indoor propagation respectively. The first received multipath component travels along the straight line between the verifier and the prover. Hence, the distance d between the verifier and the prover is equal to d_1. According to [9], for resolvable multiple path components, $\Delta \geq \frac{1}{B}$, where B is the bandwidth of the wireless communication system. Thus, $f = \sqrt{(\frac{d_2}{d})^{\alpha}} = \sqrt{(\frac{d+\Delta c}{d})^{\alpha}} \geq \sqrt{(\frac{d+\frac{c}{B}}{d})^{\alpha}}$ and we have $d \geq \frac{c}{B(f^{\frac{2}{\alpha}}-1)}$. □

Choosing α: For the outdoor signal propagation, according to the Okumura model, $\gamma = 44.9 - 6.55\log_{10}(h_{te})$, where h_{te} is the height of the transmitter's antenna. If the verifier has specific types of targets, for example, the verifier aims to verify the proximity of a satellite, a cellular base station, or a TV tower, then the verifier can directly compute γ by looking up the typical values of h_{te} from the corresponding wireless device handbooks. Alternatively, the verifier can also get an estimate of γ by using the typical transmitter antenna height in the outdoor environment (e.g., the typical transmitter antenna height ranges between 1 to 200 meters [20], and thus γ approximately lies between 44.9 and 29.83). After obtaining γ, the verifier can compute $\alpha = \frac{\gamma}{10}$. For the indoor signal propagation, $\alpha = \frac{\lambda}{10}$, where λ is the indoor path loss factor that doesn't rely on the antenna height and it can be obtained through empirical experiments.

Note that the path loss exponent α for both outdoors and indoors can be actually regarded as an attenuation factor that reflects the attenuation caused by the propagation path. Previous studies have performed extensive empirical experiments to measure typical values of such an attenuation factor in different wireless environments [9]. For example, the attenuation factor is 2.0 for vacuum free space, 2.7–3.5 for urban areas, 3.0–5.0 for suburban areas, and 1.6–1.8 for indoors [9]. In the following discussion,

without loss of generality, we use these typical empirical values of the attenuation factor as the example α. Nevertheless, the verifier can obtain α empirically using existing readily-available approaches (e.g., [3, 19]), and a real-measured attenuation factor can help to improve the accuracy of the proximity lower bound estimation.

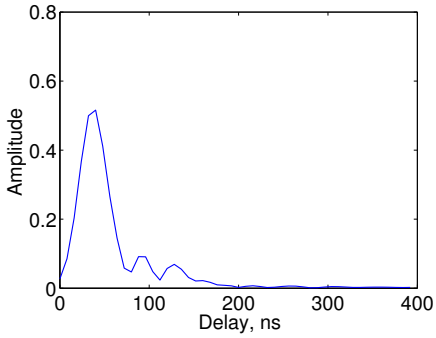

Fig. 2. An example of the real-measured channel impulse response obtained from the CRAWDAD data set

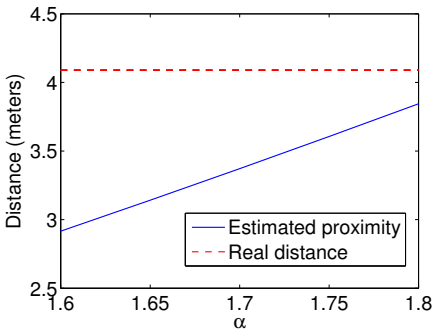

Fig. 3. Estimated lower bound v.s. the real distance

Experimental Examples Figure 2 shows an example of a real-measured channel impulse response obtained from the CRAWDAD data set [27], which contains channel impulse responses collected in an indoor environment with obstacles (e.g., cubicle offices and furniture) and scatters (e.g., windows and doors). The channel impulse response was measured when the distance between the transmitter and the receiver is 4.09 meters. From Figure 2, we can see that each received multipath component leads to a triangle in shape with a peak [23]. The second multipath component arrives at the receiver about 75 nanoseconds after the arrival of the first one. The proximity fingerprint is 5.6499. The channel impulse response was measured indoors, and thus α ranges between 1.6 and 1.8.

We use Lemma 3 to estimate the lower bound of the proximity of the transmitter, and Figure 3 shows the result. We can observe that the estimated lower bound increases as α increases. However, when α reaches the maximum value (i.e., 1.8) of the indoor environment, the real distance is still bounded by (i.e., greater than) the estimated lower bound. Specifically, when $\alpha = 1.8$, the lower bound of the proximity is 3.84 meters. This means the transmitter should be at least 3.84 meters away from the receiver. The actual distance between the transmitter and the receiver is 4.09 meters, which is slightly greater than the lower bound 3.84 meters.

Note that long-haul communications may desire a much relaxed tightness of the proximity lower bound. For example, GPS satellites running on the Low Earth Orbit have an altitude of approximately 2,000,000 meters (1,200 miles). With a proximity lower bound of 1,000,000 meters (i.e., the bound is less than the actual proximity by 50%), it would be possible to prevent most attackers from impersonating the satellites,

because it is usually very difficult for the attacker to achieve such a long transmission range.

3.3 System Design

In what follows, we show how the theoretical result of Lemma 3 can be used in a practical communication system to achieve the far proximity identification.

The verifier's objective is to find out the proximity lower bound of the prover, i.e., to verify that the prover is at least a certain distance away. According to Lemma 3, the proximity lower bound is computed by $\frac{c}{B(f^{\frac{2}{\alpha}}-1)}$. Thus, the verifier can simply compute this bound with the knowledge of the speed of light c, the system bandwidth B, the path loss exponent α, and the proximity fingerprint f. The speed of light c is a universal physical constant and the bandwidth B is a system configuration parameter, and both of them are known to the verifier. The path loss exponent α can be either obtained empirically, or can be determined using the typical values. The proximity fingerprint f is the only remaining factor that the verifier needs to decide to compute the lower bound.

As we discussed earlier, the fingerprint f is the ratio of A_{r1} to A_{r2}, where A_{r1} and A_{r2} are the amplitudes of the first and the second received multipath components. A_{r1} and A_{r2} can be extracted from the channel impulse response. A wireless packet is usually preceded by a preamble, a special data content that indicates the beginning of an incoming packet. When the prover sends a packet to the wireless channel, the verifier will first capture the preamble using the match filtering technique [10]; then the verifier knows that there is an incoming packet and continues to receive the payload. The preamble not only enables packet capture, but also enables the estimation of the channel impulse response at the verifier.

After receiving the preamble, the verifier can use existing channel estimation techniques (e.g., least-square (LS) and linear minimum mean squared error (LMMSE) estimators [4]) to estimate the channel impulse response from the preamble, and thereby obtain the values of A_{r1} and A_{r2} and the proximity fingerprint $f = A_{r1}/A_{r2}$. It is worth pointing out that using the preamble is not the only way to obtain A_{r1} and A_{r2}. The verifier can also use blinding estimation methods (e.g., [30]) to estimate the channel impulse response from the entire content of the preamble and the payload. In addition, the verifier can use hybrid methods (e.g., [13]) that combine preamble-based estimation and blind estimation together to improve the estimation accuracy. After obtaining the proximity fingerprint f and demodulating the payload and authentication information, the verifier then verifies the prover's proximity using Lemma 3.

3.4 Dealing with Jam-and-replay Attacks

To fool the verifier, the attacker may try to create a fake second path by using another active wireless device to send signals from a different direction. In this case, the attacker must make sure that there is no multipath effect for the signals traveling on the direct path (i.e., the path from the prover to the verifier) and the fake path (i.e., the path from the active wireless device to the verifier). Otherwise, the attacker cannot control and

guarantee that the fake path is exactly the second received path at the verifier side. Eliminating the multipath effect completely is normally regarded as infeasible.

However, the attacker may alternatively launch Jam-and-replay attacks to deceive the far proximity identification system. In the jam-and-replay attack, the attacker replays an intercepted signal from the prover at the attacker's own location, such that the verifier is fooled into taking the attacker's proximity as the prover's proximity. At the same time, the attacker jams the transmission to prevent the verifier from receiving the original signal from the prover; hence, traditional anti-replay mechanisms such as sequence numbers do not work.

A common method of addressing jam-and-replay attacks is to explore timestamps (e.g., [16]). In such a method, the sender includes a timestamp in the transmitted message, which indicates the time when a particular bit or byte called the anchor (e.g., the start of the message header) is transmitted over the air. Upon receiving a frame, the receiver can use this timestamp and its local message receiving time to estimate the message traverse time. An overly long time indicates that the message has been forwarded by an intermediate attacker.

Timestamps-based method requires clock synchronization between the sender and the receiver, but it generally has a low synchronization requirement in common wireless applications. For example, in an 11 Mbps 802.11g wireless network, the transmission of a typical 1500-byte TCP message requires 1.09 (i.e., $\frac{1500*8}{11\times10^3}$) milliseconds. Thus, the attacker at least doubles the transmission time of the message to 2.18 milliseconds. As long as the verifier and the prover have coarsely synchronized clocks that differ in the order of milliseconds, the verifier can detect jam-and-replay attacks. Note that the synchronization requirement can be further relaxed in GPS applications. GPS satellites have a transmission rate ranging between 20 bits/s and 100 bits/s [1]. The transmission of a standard 1500-bits GPS navigation message [1] takes 15 – 75 seconds, and accordingly the synchronization accuracy can be reduced to the order of seconds.

In addition, to launch jam-and-replay attacks, the attacker must send jamming signals to jam the wireless transmission. Jamming attacks have been extensively studied in the literature, and various techniques regarding jamming detection and countermeasures have been proposed (e.g., [9, 15, 26]). The prover and the verifier can also use existing jamming detection or anti-jamming techniques to discover the presence of jam-and-replay attacks, or to defend against such attacks.

4 Experimental Evaluation

4.1 Experiment Setup

Wireless propagation can be either line-of-sight (LoS) or non-LoS (NLoS). In LoS scenarios, there exist no major or very few obstacles residing between the transmitter and receiver, and thus LoS scenarios usually feature better signal quality. In NLoS scenarios, there exist a number of major obstacles between the transmitter and receiver, and NLoS scenarios are more complicated with higher signal distortion and sharper changes in signal strength.

Far proximity identification often applies to long-haul wireless communications (e.g., GPS) in outdoor environments, which are usually open and have a much stronger

feature in LoS than NLoS. Compared to outdoor environments, indoor environments like offices, residential homes, and shops, are more complicated due to the frequent occurrences of walls, people, furniture, cubicles, etc. Thus, indoor environments usually have a fairly large number of NLoS propagation paths. In our experiment, we choose the more challenging indoor environment for our evaluation to examine the worst-case performance of the proposed method.

We validate the proposed far proximity identification technique using the CRAWDAD data set [22], which contains more than 9,300 real channel impulse response measurements (i.e., link signatures) in a 44-node wireless network [27]. The measurement environment is an indoor environment with obstacles (e.g., cubicle offices and furniture) and scatters (e.g., windows and doors). More information regarding the CRAWDAD data set can be found in [22, 27].

We herein use *error rate* and *tightness of the bound* as metrics to evaluate the performance of the proposed technique in the real world. In addition, the proximity lower bound is computed based on a key factor, the proximity fingerprint. Thus, the proximity fingerprints plays a vital role in proximity identification. To further validate the the feasibility of using proximity fingerprints for proximity identification, we also perform experiments to reveal the relationship between the real distance and the proximity fingerprints. Our evaluation metrics are summarized below.

- **Error rate**: The error rate is the ratio of the number of failed trials (i.e., error happens in the trail) to the total number of trials.
- **Tightness of the bound**: Tightness is the normalized difference between the estimated lower bound and the real distance (i.e., $\frac{d-\beta}{d}$, where β is the estimated proximity lower bound, and d is the real distance between the verifier and the prover).
- **Proximity Fingerprints**: The proximity fingerprint is the ratio of the amplitude of the first received multipath component to that of the second one.

4.2 Experiment Results

Based on the CRAWDAD data set, we perform experiments under both LoS and NLoS scenarios to show the error rate, tightness of the bound, and the relationship between the proximity fingerprint and the distance.

We distinguish two types of channel impulse responses: if a LoS path exists and there are no obstacles between the transmitter and the receiver, we mark the corresponding channel impulse responses as LoS channel impulse responses. Otherwise, we mark them as NLoS channel impulse responses. Thus, we obtain two sets of data. The first set is formed by all LoS channel impulse responses, and the second one is formed by all NLoS channel impulse responses. We perform our experiments using both sets.

Error Rate Error rate vs. pathloss: To obtain the error rate, we experiment as follows. Let N_{LoS} denote the number of channel impulse responses in the LoS data set. For each channel impulse response in the data set, we compute the proximity fingerprint and the corresponding proximity lower bound using Lemma 3. We also compute the real distance between the transmitter and the receiver based on their coordinates. If the

lower bound is less than the real distance, we mark the trial as successful. Otherwise, we mark the trial as failed. Accordingly, the error rate is calculated as $\frac{N_f}{N_{LoS}}$, where N_f is the number of failed trails and N_{LoS} is the total number of trials. We perform the experiment again using the NLoS data set and obtain the corresponding error rate for the NLoS scenario.

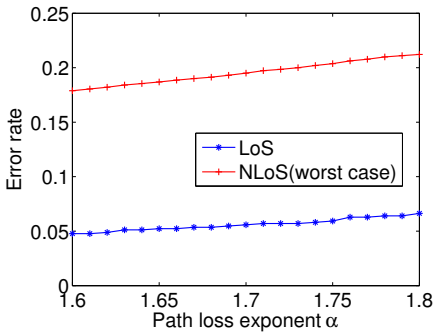

Fig. 4. Error rate as a function of pathloss exponent α.

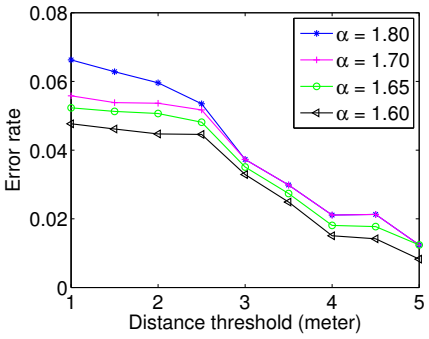

Fig. 5. Error rate as a function of the distance threshold $d_{threshold}$ in the LoS scenario

The channel impulse responses are collected from an indoor environment, and the corresponding pathloss exponent α empirically ranges between 1.6 and 1.8. Thus, we perform our experiment for different values of α in this range. Figure 4 plots the error rate as a function of α. The pathloss exponent α reflects how a signal is distorted and attenuated during its propagation, and a large α can result in higher signal distortion and attenuation. Accordingly, from Figure 4 we can observe that the error rate increases as α increases. However, when α reaches the maximum value for indoor environments, the achieved error rate in the LoS scenario is as low as 0.075. For the minimum α of 1.6, the proposed approach has a reduced error rate of 0.05.

For the NLoS scenario, we can still achieve an error rate between 0.17 and 0.22. Note that NLoS scenarios are the worst-case scenarios. Far proximity identification is typically used in outdoor environments, which have the stronger LoS feature. As shown in Figure 4, the error rate of LoS scenarios is much lower than that of the NLoS scenarios.

Error rate vs. distance: We then perform experiments to examine how the real distance affects the error rate. For each channel impulse response in the LoS data set, we compute the distance between the corresponding transmitter and the receiver. Let d_{max} and d_{min} denote the maximum and minimum distance among all computed distances. We calculate the error rate using the set formed by channel impulse responses whose corresponding distance are larger than a threshold value $d_{threshold}$. We start from $d_{threshold} = d_{min}$ and increase $d_{threshold}$ each time until $d_{threshold}$ reaches d_{max}. We perform the experiments again using the NLoS set.

Figure 5 shows the error rate as a a function of $d_{threshold}$ in the LoS scenario. The error rate decreases as $d_{threshold}$ increases. The obvious reason is that a larger

$d_{threshold}$ indicates a longer distance between the transmitter and the receiver, and thus a higher chance that the estimated proximity lower bound is less than the distance. When $d_{threshold}$ approaches the maximum distance between the sender and the receiver, the corresponding error rate is 0.01. When $d_{threshold}$ approaches the minimum distance, the error rate slightly increases but it is still a small rate that ranges between 0.05 and 0.07 for different α.

Figure 6 plots the error rate of the NLoS scenario for $\alpha = 1.80$, which results the worst error rate as compared to other values of α. Contrary to the LoS scenario, the error rate of the NLoS scenario increases as $d_{threshold}$ increases. That's because in the NLoS scenario a longer distance between the transmitter and the receiver indicates a higher chance that there are more obstacles, and thus a reduced proximity detection accuracy. The "worst worst case" happens when $d_{threshold}$ approaches the maximum distance d_{max} for the worst case NLoS scenario. However, as we can observe from Figure 6, the achieved error rate of the "worst worst case" is about 0.25. This means that we can successfully obtain the proximity lower bound for a majority number (75%) of verifiers. As $d_{threshold}$ decreases, the error rate decreases quickly. When $d_{threshold}$ approaches the minimum distance, the achieved error rate is about 0.15. Again, the experiment is performed in an indoor environment (e.g., WiFi and Bluetooth), which has a short signal propagation distance. Outdoor wireless applications (e.g., space communications and TV broadcasting) usually have the stronger LoS feature, and therefore can substantially benefit from the proposed method in terms of significantly reducing the error rate.

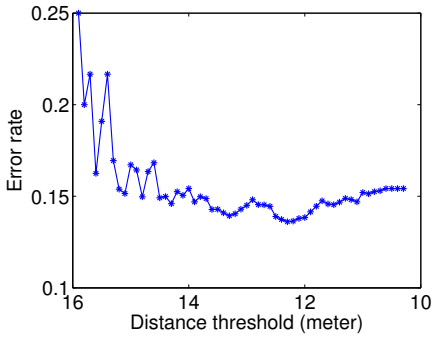

Fig. 6. Error rate as a function of the distance threshold $d_{threshold}$ in the NLoS scenario

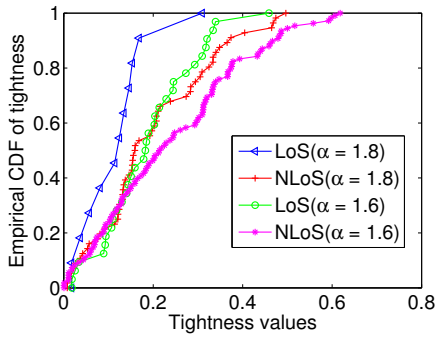

Fig. 7. The empirical CDFs of the tightness.

Tightness of the proximity bound Our second evaluation metric is the tightness of the bound. To evaluate the tightness, we perform the following experiments using LoS and NLoS data sets. In all experiments, the pathloss exponent α is set to the minimum and maximum values of 1.6 and 1.8. For each channel impulse response in the LoS data set, we compute the distance between the corresponding transmitter and the receiver and the proximity lower bound. Based on the bound and the actual distance, we can calculate the tightness of the bound. We then sort all the tightness values and compute the empirical

cumulative distribution function (CDF) for them. We perform the experiment again using NLoS data set and obtain the CDFs of the NLoS tightness values.

Figure 7 shows the CDF curves of the tightness computed using channel impulse responses collected in LoS and NLoS scenarios. For the LoS scenario with $\alpha = 1.8$, we can observe that 95% of the tightness values are less than 0.2. The indoor environment typically features a short propagation path, and thus a 0.2 tightness indicates a small absolute difference in distance. For example, if the distance between the transmitter and receiver is 5 meters, the achieved tightness can be around 1 meter. In particular, the maximum distance d_{max} between the transmitter and the receiver is about 11 meters, and the corresponding proximity bound is 9.56 meters, which is very close to the actual distance.

For the NLoS scenario with $\alpha = 1.8$, we can observe from Figure 7 that 90% of the tightness values are less than 0.3. Compared to the LoS Scenario, the NLoS scenario has a reduced performance due to the existence of obstacles. Again, the experiment is conducted based on short-range communications, and a 0.3 tightness still suggests a small absolute difference in distance. When α decreases to 1.6, the achieved tightness increases. That's because the corresponding estimated proximity lower bound decreases, and a decreased bound grows the difference between the bound and the real distance, and thus augments the tightness. However, for $\alpha = 1.6$, we can still observe that a great majority of the tightness values are fairly small, e.g., 95% and 80% of the tightness values are less than 0.25 and 0.3 in the LoS and the NLoS (worst-case) scenarios respectively. As we have discussed, such tightness of the bound is usually sufficient to prevent attackers from impersonating the transmitters in typical long-haul outdoor wireless applications.

Proximity fingerprint vs. distance The proximity fingerprint is an important parameter in computing the proximity lower bound. According to Lemma 3, the theoretical proximity lower bound is calculated as $\frac{c}{B(f^{\frac{2}{\alpha}}-1)}$. From this formula, we can easily derive that as the proximity fingerprint f increases (other parameters remain the same), the proximity lower bound decreases and vice versa. Note that the proximity lower bound reveals the least distance between the verifier and the prover. Thus, the increase of the proximity fingerprint f may also indicate the decrease of the real distance and vice versa. We plot the proximity fingerprint as a function of the distance in Figure 8. We can see that the proximity fingerprint in the NLoS scenario slightly differs from that of the LoS scenario in magnitude due to the reflection loss. However, for both scenarios, their proximity fingerprints exhibit the same tendency, i.e., they both decrease as the distance increases. This observation is consistent with our theoretical result.

5 Related Work

Related work falls into the following two areas.

(a) Distance Bounding Protocols: Distance bounding protocols are a class of protocols that determine an approximate distance between a local device and a remote device. (e.g., [5, 24, 29]). Distance bounding protocols and their variants are based on the common observation that the distance between the local and the remote devices is equal to

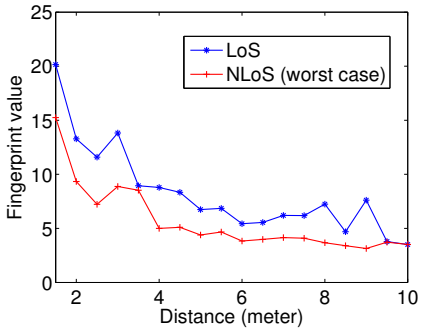

Fig. 8. Relationship between the distance and the proximity fingerprint.

the product of the speed of electromagnetic wave and the one-way signal propagation time. The approximate distance is obtained from a series of wireless packets exchanged between the local device and the remote device. Specifically, the local device sends a challenge to the remote device, which then replies with a response that is generated based on the challenge. The local device measures the round-trip time between sending the challenge and receiving the response, subtracts the processing delay from the round-trip time, and uses the result to compute the distance. Because the response is generated based on the challenge, the distance bounding protocol can prevent the remote device from pretending to be closer than it actually is by sending a fake response before it receives the challenge.

However, by delaying its response to a challenge, a remote device can appear to be arbitrarily further from the local device than it actually is. Hence, distance-bounding protocols cannot enforce lower bounds on proximity (i.e., requirements that the remote device be *at least* a certain distance from the local device). For this reason, the GPS-device and mobile-phone examples used for motivation in Section 1 cannot be enforced by distance-bounding protocols.

(b) Close Proximity Identification: There also exist traditional close proximity detection techniques (e.g., [8,17]) that can detect the presence of nearby objects without any physical contact. These techniques use electromagnetic field changes to identify a close object. A proximity sensor generates an electromagnetic field or a beam of electromagnetic radiation (e.g., infrared). If an object moves into the field range of the sensor, a field change can result, and thus the sensor senses the presence of the object. For example, a sound alert is triggered when a vehicle moves into the close proximity of a worker or an obstacle. However, traditional techniques cannot identify the proximity of a specific object, because the proximity sensor reports all nearby objects as long as those objects are in the field range.

Researchers later developed techniques that identify the close proximity of an individual target if the target can emit wireless signals (e.g., [7, 11, 18]). For example, based on the observation that a strong received signal usually indicates a close transmitter, Macii et al developed approaches that determine the proximity of the remote wireless device by measuring received signal strength [18]. However, the use of signal

strength to determine proximity was found to be insecure, as a dishonest remote device can easily pretend to be close to the local device by boosting its transmit power.

More recent efforts overcome this drawback with the assistance of special hardware [7, 11]. Cai et al. proposed a scheme that identifies the presence of a close wireless device by using multiple antennas [7]. Halevi et al. proposed to use ambient sensors to detect whether a Near-Field-Communication (NFC) device is nearby or not [11]. Although those approaches can prevent attackers manipulating transmit power to deceive the local device, they cannot be directly extended to address the far proximity identification problem. They output a decision regarding whether a target is nearby, but such a decision cannot guarantee that the target is at least a certain distance away. Also, the requirement of special hardware such as multiple antennas and ambient sensors introduces extra cost and may reduce their compatibility.

Liu et al. proposed a new close proximity identification approach that does not rely on special hardware [16]. By using the wireless physical features that uniquely identify a wireless link between a transmitter and a receiver, the proposed technique enables the local device to distinguish between a nearby and a far-away remote device. An attacker cannot manipulate such physical features to pretend to be close to the local device. However, similar to all previous approaches, this approach is a decision-based, i.e. outputs a simple "yes" or "no" to indicate whether the remote device is very close or not. Hence, it does not provide the quantitative lower bound of the proximity, which is the primary contribution of this paper.

6 Conclusion

In this paper, we proposed a far proximity identification approach that determines the lower bound of the distance between the verifier and the prover. The key idea of the proposed approach is to estimate the proximity lower bound from the unforgeable fingerprint of the proximity. We have examined the proposed approach through experimental evaluation using the CRAWDAD data set.

References

1. Gps signals. `http://en.wikipedia.org/wiki/GPS_signals`. [Online; accessed 27-July-2013].
2. Marine vhf radio. `http://en.wikipedia.org/wiki/Marine_VHF_radio`. [Online; accessed 13-July-2013].
3. N. Alam, A. T. Balaie, and A. G. Dempster. Dynamic path loss exponent and distance estimation in a vehicular network using doppler effect and received signal strength. In *Proceedings of 2010 Vehicular Technology Conference Fall (VTC 2010-Fall)*, pages 1–5, 2010.
4. M. Biguesh and A. B. Gershman. Training-based mimo channel estimation: A study of estimator tradeoffs and optimal training signals. *IEEE Transaction on Signal Processing*, 54(3):884–893, March 2006.
5. S. Brands and D. Chaum. Distance bounding protocols. In *Proceedings of EUROCRYPT*, pages 344–359, 1994.
6. V. Brik, S. Banerjee, M. Gruteser, and S. Oh. Wireless device identification with radiometric signatures. In *Proceedings of the 14th ACM International Conference on Mobile Computing and Networking (MobiCom '08)*, pages 116–127, 2008.

7. L. Cai, K. Zeng, H. Chen, and P. Mohapatra. Good neighbor: Ad hoc pairing of nearby wireless devices by multiple antennas. In *Proceedings of the Annual Network and Distributed System Security Symposium (NDSS 2011)*, 2011.
8. Z. Chen and R.C. Luo. Design and implementation of capacitive proximity sensor using microelectromechanical systems technology. *IEEE Transactions on Industrial Electronics*, 45(6):886–894, 1998.
9. A. Goldsmith. *Wireless Communications*. Cambridge University Press, New York, NY, USA, 2005.
10. K. Gunnam, G. Choi, M. Yeary, and Y. Zhai. A low-power preamble detection methodology for packet based rf modems on all-digital sensor front-ends. In *In Proceedings of the IEEE Instrumentation and Measurement Technology Conference*, 2007.
11. T. Halevi, D. Ma, N. Saxena, and T. Xiang. Secure proximity detection for nfc devices based on ambient sensor data. In *Proceedings of the European Symposium on Research in Computer Security (ESORICS 2012)*, 2012.
12. G. P. Hancke and M. G. Kuhn. An RFID distance bounding protocol. In *Proceedings of SecureComm'05*, pages 67–73, 2005.
13. C. Jinho. Equalization and semi-blind channel estimation for space-time block coded signals over a frequency-selective fading channel. *IEEE Transactions on Signal Processing*, 52(3):774 – 785, 2004.
14. L. B. Kuechle. Selecting receiving antennas for radio tracking. `http://www.atstrack.com/PDFFiles/receiverantrev6.pdf`.
15. A. Liu, P. Ning, H. Dai, Y. Liu, and C. Wang. Defending DSSS-based broadcast communication against insider jammers via delayed seed-disclosure. In *Proceedings of the 26th Annual Computer Security Applications Conference ACSAC '10*, December 2010.
16. Y. Liu, P. Ning, and H. Dai. Authenticating primary users' signals in cognitive radio networks via integrated cryptographic and wireless link signatures. In *Proceedings of 2010 IEEE Symposium on Security and Privacy (S&P '10)*, pages 286–301, May 2010.
17. P. H. Lo, C. Hong, S. C. Lo, and W. Fang. Implementation of inductive proximity sensor using nanoporous anodic aluminum oxide layer. In *Proceedings of 2011 International Solid-State Sensors, Actuators and Microsystems Conference (TRANSDUCERS)*, pages 1871–1874, 2011.
18. D. Macii, F. Trenti, and P. Pivato. A robust wireless proximity detection technique based on rss and tof measurements. In *Proceedings of 2011 IEEE International Workshop on Measurements and Networking (M&N'11)*, pages 31–36, 2011.
19. G. Mao, B. D. O. Anderson, and B. Fidan. Path loss exponent estimation for wireless sensor network localization. *The International Journal of Computer and Telecommunications Networking*, 51(10):2467–2483, 2007.
20. A. F. Molisch. *Wireless Communications, 2nd Edition*. Wiley India Pvt. Limited, 2007.
21. C. Paget. Practical cellphone spying. *DEF CON 18*, 2010.
22. N. Patwari and S. K. Kasera. CRAWDAD utah CIR measurements. `http://crawdad.cs.dartmouth.edu/meta.php?name=utah/CIR`.
23. N. Patwari and S. K. Kasera. Robust location distinction using temporal link signatures. In *MobiCom '07: Proceedings of the 13th annual ACM international conference on Mobile computing and networking*, pages 111–122, New York, NY, USA, 2007. ACM.
24. K. B. Rasmussen and S. Čapkun. Realization of rf distance bounding. In *Proceedings of the USENIX Security Symposium*, 2010.
25. K.B. Rasmussen, C. Castelluccia, T.S. Heydt-Benjamin, and S. Čapkun. Proximity-based access control for implantable medical devices. In *Proceedings of the 16th ACM Conference on Computer and Communications Security (CCS '09)*, 2009.
26. Robert A. Scholtz. *Spread Spectrum Communications Handbook*. McGraw-Hill, 2001.

27. SPAN. Measured channel impulse response data set. http://span.ece.utah.edu/pmwiki/pmwiki.php?n=Main.MeasuredCIRDataSet.
28. S. Sud. A low complexity spatial rake receiver using main beam multipath combining for a cdma smart antenna system. In *Proceedings of 2007 IEEE Military Communications Conference (MILCOM)*, pages 1–6, 2007.
29. N. O. Tippenhauer and S. Čapkun. Id-based secure distance bounding and localization. In *Proceedings of 2009 European Symposium on Research in Computer Security(ESORICS'09)*, 2009.
30. M. K. Tsatsanis and G. B. Giannakis. Blind estimation of direct sequence spread spectrum signals in multipath. *IEEE Transactions on Signal Processing*, 5(45):1241 – 1252, 1997.
31. R. Weinmann. The baseband apocalypse. *BlackHat DC*, 2011.
32. L. Yu, W. Liu, and R. J. Langley. Robust beamforming methods for multipath signal reception. *Digital Signal Processing*, 20(2):379–390, 2007.
33. J. Zhang, M. H. Firooz, N. Patwari, and S. K. Kasera. Advancing wireless link signatures for location distinction. In *MobiCom '08: Proceedings of the 14th ACM international conference on Mobile computing and networking*, New York, NY, USA, 2008. ACM.