PCASA: Proximity based Continuous and Secure Authentication of Personal Devices

Pengfei Hu*, Parth H. Pathak†, Yilin Shen‡, Hongxia Jin‡, Prasant Mohapatra*
*Computer Science Department, University of California, Davis, CA, USA
Email: {pfhu, pmohapatra}@ucdavis.edu
†Computer Science Department, George Mason University, Fairfax, VA, USA
Email: phpathak@gmu.edu
‡Samsung Research America, Mountain View, CA, USA
Email: {yilin.shen, hongxia.jin}@samsung.com

***Abstract*—User's personal portable devices such as smartphone, tablet and laptop require continuous authentication of the user to prevent against illegitimate access to the device and personal data. Current authentication techniques require users to enter password or scan fingerprint, making frequent access to the devices inconvenient. In this work, we propose to exploit user's on-body wearable devices to detect their proximity from her portable devices, and use the proximity for continuous authentication of the portable devices. We present PCASA which utilizes acoustic communication for secure proximity estimation with sub-meter level accuracy. PCASA uses Differential Pulse Position Modulation scheme that modulates data through varying the silence period between acoustic pulses to ensure energy efficiency even when authentication operation is being performed once every second. It yields an secure and accurate distance estimation even when user is mobile by utilizing Doppler effect for mobility speed estimation. We evaluate PCASA using smartphone and smartwatches, and show that it supports up to 34 hours of continuous authentication with a fully charged battery.**

I. Introduction

There has been a tremendous growth in the number of personal devices a typical user owns, carries and wears. Devices such as smartphones, tablets and laptops are at constant risks of being left unattended and personal data being stolen. User's proximity to these devices is a strong indication of whether these devices are within user's vicinity and physical control or not. With increasing popularity of wearable devices like smartwatches, fitness trackers and smartglasses, it is possible to exploit their proximity with the portable devices (e.g. smartphones, tablets) for user authentication. For example, today's smartphones (Android Smart Lock [1]) can detect user's smartwatch within its Bluetooth range [2], use this information to infer user's presence and remain unlocked for user's convenience. However, such techniques only provide a coarse-grained control because they rely on RSS (Received Signal Strength) which is known to be unreliable [3] for authentication purposes. On the other hand, accurate estimation of proximity of user's wearable device(s) from her portable device(s) can enable a secure and flexible authentication of the portable device(s).

Accurate estimation of proximity between user's personal devices is the challenge. First and foremost challenge is that it is difficult to measure the proximity at sub-meter level accuracy. Previous approaches [4], [5] have suggested to use ambient RF signal to detect if a given set of devices are in the same RF context. Due to the high variations introduced by interference and multi-path effects, these approaches are limited to very low accuracy and longer estimation times. The second challenge is that such authentication should rely on identity verification of the personal devices, which in turn requires an active communication between the devices. Numerous acoustic based approaches [6]–[11] have been proposed to measure proximity with higher accuracy using Time Of Arrival (TOA) or Time Difference Of Arrival (TDOA) methods. However, these techniques are not designed for authentication which makes them vulnerable to many types of security attacks such as the spoofing attack. The last challenge is that because proximity based authentication needs to be performed continuously, it is crucial to ensure that the proximity detection technique consumes very low energy even with authentication rate of one authentication per second and support user's mobility during proximity estimation. Previous approaches of acoustic communication cannot be directly applied because they either are not suitable beyond very short range ($< 1m$) applications [12], [13] or they cannot support user mobility [14]. More importantly, none of the previous research on acoustic communication or proximity measurement address the energy efficiency problem.

In this paper, we design and evaluate PCASA, a proximity-based continuous and secure authentication scheme for user's personal devices. PCASA uses user's wearable device (e.g. smartwatch) as a vouching device for authenticating her portable device (e.g. smartphone, tablet, laptop) by accurately measuring the distance between the two. PCASA has three important features:

(1) Secure - PCASA is designed to defend against the attackers who aim to get illegitimate access to user's portable device when the user is away, by masquerading user's wearable device that is physically close enough to gain the access.

(2) Accurate - PCASA relies on acoustic communication using the part of the ultrasonic spectrum that is inaudible to human ears. It leverages the existing speaker and microphone in the mobile devices to send and receive data and to estimate the proximity with sub-meter accuracy in real-time even when the user is mobile.

(3) Energy Efficient - To our knowledge, PCASA is the first of its kind system that can perform continuous authentication using acoustic signals. Even with authentication being performed every second, PCASA consumes very low energy

through the use of an energy efficient modulation scheme. PCASA is suitable for wearable devices which have very limited battery capacity.

Contributions of this work can be summarized as follows:

(1) PCASA is designed to defend against zero-effort attacks and spoofing attacks with special consideration to user mobility using carefully designed protocol and encrypted messages. It ensures that an attacker cannot gain the illegitimate access to a user's portable device by impersonating her wearable device within the safety range.

(2) For communication between devices, we adopt Differential Pulse Position Modulation (DPPM) - which utilizes idle duration between the acoustic pulses to modulate the data. DPPM's properties of high energy efficiency and low demodulation error are very well suited for continuous communication between user's personal devices.

(3) We implement PCASA and evaluate it using multiple smartphones (Samsung Galaxy S4, S5, S6 and iPhone 6S) and smartwatches (Apple Watch and Samsung Gear S2). With a fully charged battery, it could support up to 34 hours continuous authentication with average proximity estimation error being less than 0.25 m when the user is mobile.

The rest of this paper is organized as follows. Section II provides an overview of our system. We describe the proximity-based authentication scheme in Section III, including both the basic PCASA and with user mobility. Section IV presents the energy efficient modulation scheme. The evaluation of PCASA is provided in Section V. Section VI discusses the related work, followed by the conclusion in Section VII.

II. System Overview

In this section, we first discuss proximity-based authentication and its design challenges, and provide an overview of our PCASA system. We then discuss the attack model that PCASA aims to defend against.

A. Authentication using Proximity

PCASA is built on the fact that more and more users are adopting wearable devices such as smartwatches, wrist-worn fitness trackers, etc. that are already authenticated by the user as they are always on users' body. We refer to these wearable devices as *vouching devices*. Users also carry other types of devices such as smartphone, tablet, laptop etc. which are not always within users' vicinity and/or physical control. We refer to these portable devices as *authenticating devices*. The central motivation behind the design of PCASA is that if there is a secure means of detecting the proximity of user's wearable device(s) from her portable device(s), it is possible to control and authenticate user's access to the portable device(s).

Since the vouching device and the authenticating device are usually physically close when the legitimate user uses the authenticating device, it can be automatically unlocked when distance is small enough to meet users' personal needs. Such alternative primary authentication can help users by avoiding the hassle of either typing in the password or using the fingerprint sensor. When the user is away from the authenticating device, the device can remain locked to secure user's personal information. With accurate proximity detection, it is possible

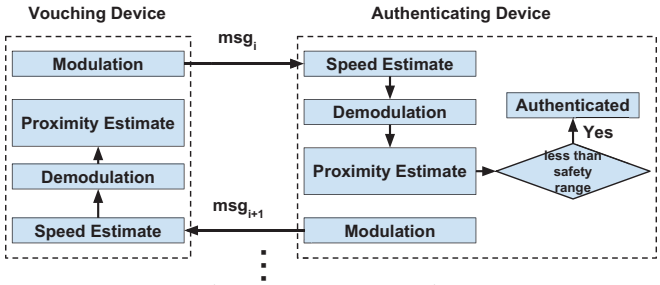


Fig. 1: System Overview

for users to customize their preferences about the distance beyond which they would like their devices locked.

Apart from authentication, the user proximity can also be used by the authenticating device to customize and configure applications. For example, a smartphone can either (i) show the notifications on the screen when user is very close (within hand's reach or screen readable), (ii) show the notifications but hide the content when user is within a room distance or (iii) turn off the notifications when user is far away or outside the range. Proximity detection has many similar applications, however, our primary focus in this work is on authentication.

B. Challenges

The use of proximity enables an intuitive way of device authentication where user is not required to proactively perform any action (e.g. enter password, user fingerprint sensor). However, there are many challenges in realizing it in practice.

(1) High accuracy: Proximity-based authentication requires that the distance between the vouching and the authenticating device is determined with sub-meter level accuracy. Although acoustic communication can provide this level of accuracy, user mobility and the resultant Doppler effect introduce significant challenges in accurate distance estimation. This is especially important given that the vouching devices are wearables which constantly move with user's body motion.

(2) Energy efficiency: In order to ensure secure authentication, it is necessary that the proximity detection is carried out continuously. This requirement can incur very high energy consumption overhead on the mobile devices. Hence, it is necessary that the acoustic communication is energy efficient and computationally inexpensive to be implemented on the wearable devices.

In real-world application scenarios, it is possible that devices of many users are continuously performing the proximity detection operations in parallel. Hence, it is desirable that these multiple pairs of devices can operate securely and efficiently without any interference to each other.

C. PCASA System

Fig.1 provides an overview of PCASA, a continuous proximity-based authentication system that is secure, accurate and energy efficient. In PCASA, the vouching device continuously sends a connection request message on an acoustic channel. This message contains its identity and is signed by its key shared with the authenticating device. In our system, the authenticating device and the vouching device are assumed to have conducted a one-time device pairing for exchanging

their hardware binding information (e.g., their MAC addresses - $\mathsf{MAC_A}$ and $\mathsf{MAC_V}$). If the authenticating device could successfully receive the message and retrieve the identity of the vouching device, it indicates that these two devices are within communication range. The authenticating device will send back its identity to the vouching device establishing the connection. Once the vouching device verifies the identity of authenticating device, the connection is established and they will then engage in the continuous authentication phase.

After the connection is established, both devices will serve as transmitter as well as receiver. The transmitter will send out a message containing useful timing information to the peer device at a fixed interval. The receiver will estimate its relative speed with the transmitter based on the frequency shift of the incoming acoustic signal according to the Doppler effect. Then it demodulates the incoming signal to retrieve information. The receiver will estimate the distance to the transmitter based on the retrieved information and the speed estimate. If the receiver is the authenticating device and the distance is less than a predefined threshold, then it is authenticated. The authentication distance threshold can be set by users as per their security preferences.

D. Attack Model

We assume that a legitimate user can select a safety range $\boldsymbol{R}$. When the distance between the authenticating device and the vouching device is no larger than $\boldsymbol{R}$, the authenticating device can be accessed by the legitimate user but cannot be accessed by attackers. In this paper, we are interested in defending against attackers whose goal is to get unauthorized access to the authenticating device (i.e., when user is away, a.k.a. the distance between two devices is larger than $\boldsymbol{R}$). PCASA is built on following assumptions. First, we assume that the attacker is restricted to practical computational bounds that cannot infer the shared key between the vouching device and authenticating device before they safely update that key. Second, we assume that the authentication between vouching and authenticating devices takes place only when the vouching device starts to move, i.e., the user starts to move. This avoids unnecessary authentication and power consumption in practice. Lastly, we assume that the authenticating device and the vouching device are loosely synchronized. Given these assumptions, we consider the following two types of attacks:

1) Zero-Effort Attacks: The attacker directly tries to access the authenticating device while the authenticating device is out of legitimate user's vicinity or control but is authenticated. This type of attack exists in RF based approaches and is usually caused by inaccurate proximity estimation. One of the best examples is the popular Bluetooth authentication. The authenticated device will remain authenticated within the communication range ($\approx 10m$) of Bluetooth. The sub-meter accuracy of Bluetooth signal strength based approaches can result in cases where it is possible that the authenticating device is not in user's sight, and an attacker could easily access the authenticated device.

2) Spoofing Attacks: In the second type of attacks, the attacker impersonates the vouching device to pretend to be physically much closer to the authenticating device than the real vouching device. The current state-of-the-art proximity detection approach [6] are vulnerable to this type of attacks. Specifically, we consider two types of spoofing attacks:

(a) Replay Attacks: The attacker uses his/her own device to record signals from the vouching device. Then it replays the recorded signal from a short distance ($< \boldsymbol{R}$) to spoof the authenticating device and making it believe that vouching device is present.

(b) Relay Attacks: The attacker uses his/her own devices to create a faster channel to relay all messages between the vouching and authenticating devices, aiming to fake a smaller distance between the two devices. As shown in Fig. 2, the attacker uses two malicious devices close to the vouching device and the authenticating device. The fast channel can be established through RF.

Fig. 2: Relay Attack in Proximity based Authentication

III. Secure Proximity Protocol

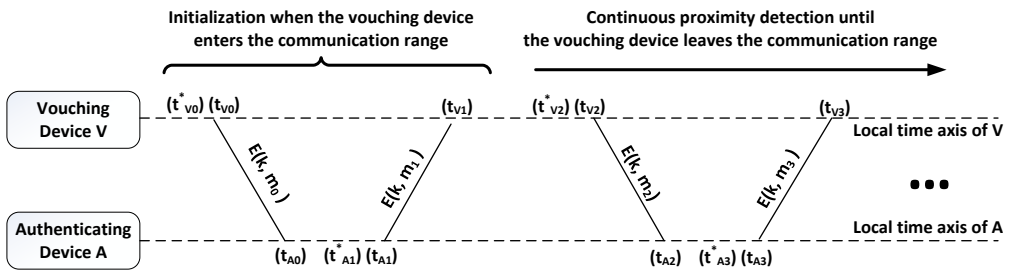
We note that PCASA can not only accurately estimate the distance between the authenticating device and the vouching device, it does so in a secure manner such that it can be useful in numerous proximity-based security related services including authentication and secure notifications. In the rest of section, we first present the secure proximity protocol of PCASA. Next, we extend PCASA to a more practical case where a user is mobile during the estimation of proximity. Lastly, we provide a comprehensive security analysis of PCASA based on the attack model discussed in Section II-D.

A. PCASA Protocol Description

Figure 3 shows the overview of PCASA protocol. PCASA requires both the authenticating and vouching devices to be equipped with a speaker and a microphone. The continuous authentication relies on the connection between the vouching device and the authenticating device through the acoustic channel. The vouching device takes the responsibility to initialize the connection by continuously sending a connection request before establishing the connection. In order to support multiple pairs of devices for authentication, we use Frequency Division Multiplexing (FDM) to divide the total bandwidth into several channels. Each pair of devices can exchange messages over an available channel. More details about the channel division will be discussed in Section III-B2.

Once the vouching device enters the communication range of the authenticating device, the connection request could be received by the authenticating device. As shown in Fig. 3(a), the proximity detection will be conducted continuously after the authenticating and vouching devices establish the connection.

The first successfully received connection request by the authenticating device is denoted as m_0 which contains the encrypted identity (MAC address $\mathsf{MAC_V}$) of the vouching device. This message will be modulated onto the acoustic signal through our novel modulation scheme (discussed in Section IV) and transmitted through the speaker. As there exists a delay



(a) Protocol Overview (b) Transmitted Messages

Fig. 3: PCASA Protocol: (a) shows how PCASA protocol works during the period that the vouching device is within the communication range of authenticating device, including an initialization phase and continuous proximity detection; All message are encrypted; (b) illustrates each message which contains the sender's timestamp and the time difference.

between issuing a command to send the signal and actually emitting the signal, we denote the time of issuing the command as t^*_{V0} and the emitting time as t_{V0} for m_0.

The authenticating device monitors all the channels and once it receives the transmitted signal on a channel, it will try to demodulate the signal through the demodulation scheme described in Section IV-B. If the authenticating device can successfully decrypt the demodulated message with its shared key (with the vouching device) and retrieve the identity, it considers this as a connection request coming from its paired vouching device and marks the arrival time as t_{A0}. The authenticating device will then reply to the vouching device with a message m_1 which includes its encrypted MAC address $\mathsf{MAC_A}$ and mark the emitting time of m_1 as t_{A1}. The vouching device will denote the arrival time of m_1 as t_{V1} and perform the same process as the authenticating device. If the vouching device could successfully retrieve the identity of the authenticating device, the connection is established.

After successfully establishing the connection, the content of the messages will be different, which is shown in in Fig. 3(b). We denote the message as m_i where $i = 2, 3, \ldots$. The vouching device sends an acoustic signal to the authenticating device by modulating the message m_2. The content of m_2 contains a timestamp t^*_{A2} which denotes when the message was modulated and the time difference $t_{A1} - t_{A0}$. The purpose of adding t^*_{A2} is to ensure the order of messages that corresponds to each authentication round and the freshness of the message to prevent the replay attack, while the time difference is used for distance estimation. At last, the authenticating device can calculate its distance from the vouching device based on time-of-flight using the following equation:

$$\frac{c}{2}[(t_{V1} - t_{V0}) - (t_{A1} - t_{A0})] \tag{1}$$

where c is the speed of sound in air ($340m/s$ [15]). Next, the authenticating device sends message m_3 to the vouching device with content including its own timestamp and time different $t_{A2} - t_{A1}$. Once the vouching device receives the message, it can also obtain the proximity estimate. Through one proximity estimation per message, the entire process is conducted continuously as long as the vouching device is within the communication range.

For the time difference, $t_{A(i+1)} - t_{Ai}$ and $t_{V(i+1)} - t_{Vi}$, the system usually provides millisecond level timestamp, the corresponding resolution of proximity estimation will be $1\ ms \times 340\ m/s = 34\ cm$ which is too large to provide accurate proximity estimation. As the device keeps recording at a sampling rate of $44.1\ kHz$ when it sends a message, the message will also be recorded by itself. Based on the recorded signal, the transmitter could easily count the audio samples between the emitting point of its own signal and the arrival point of the received signal. Hence, we use count of the samples instead of the time difference. It could provide $\frac{1\ s}{44100} \times 340\ m/s = 7.7\ mm$ distance resolution which is sufficient for our proximity application.

B. PCASA *with User Mobility*

We now consider a common scenario where a user is moving while the proximity authentication is carried out.

1) Measure the Proximity when Moving: To address the mobility of user, we need to estimate the relative speed of device movement. As shown in Figure 4, without loss of generality, we assume that the vouching device moves during the transmission and its relative speed is v. Then, we have the distance estimation $d^0_{AV} = c(t_{A0} - t_{V0})$, which is calculated when the signal s_0 arrives at the vouching device. We denote d^{0*}_{AV} as the distance between authenticating and vouching devices when the signal s_0 leaves the authenticating device. Next, the vouching device sends signal s_1. Similarly, we can get the following distance d_{AV} between them when the signal s_1 arrives the authenticating device as $d^1_{AV} = c(t_{V1} - t_{A1})$, and the distance d^{1*}_{AV} when the signal s_1 leaves vouching device.

Since the relative speed of devices is much lower than the speed of acoustic signal, we consider $d^0_{AV} \approx d^{0*}_{AV}$ and $d^1_{AV} \approx d^{1*}_{AV}$. As the vouching device incurs some delay in issuing the signal s_1 while it is moving, we could get

$$d^1_{AV} - d^0_{AV} = v(t_{V1} - t_{V0}) \tag{2}$$

With Doppler effect, we can estimate the relative speed of movement v. Since no synchronization is required with $t_{V1} - t_{V0}$, then the right hand side can be obtained. By summing up d^0_{AV} and d^1_{AV}, we get

$$d^1_{AV} + d^0_{AV} = c[(t_{V1} - t_{V0}) - (t_{A1} - t_{A0})] \tag{3}$$

Likewise, we can calculate the value of right hand side since no synchronization is required. Therefore, with Equations (2) and (3), we can easily get d^0_{AV} and d^1_{AV}. To this end, we can calculate the current distance d^2_{AV} as follows

$$d^2_{AV} = d^1_{AV} - v(t_{V2} - t_{V1}) \tag{4}$$

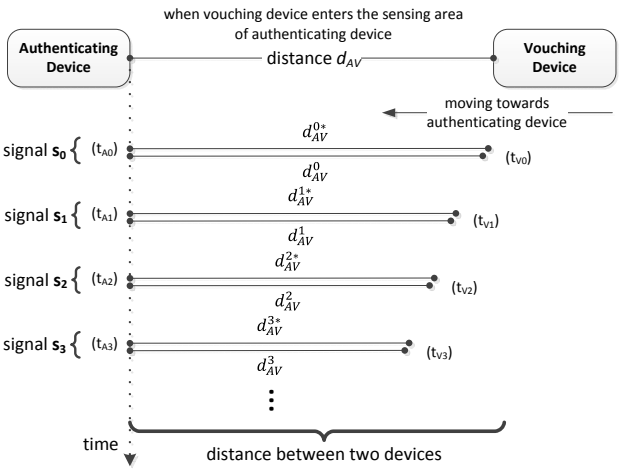


Fig. 4: Proximity estimation when the user is moving

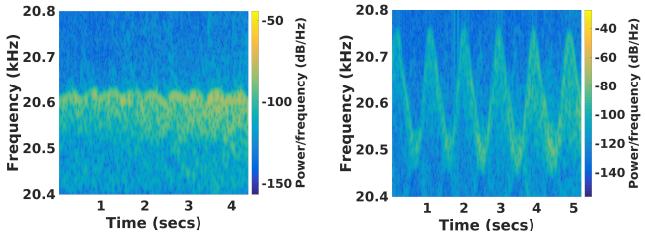
2) Measure the Relative Speed using Doppler Effect: We know from Equation (4) that to measure the proximity while there is a relative movement between the authenticating device and the vouching device, it is necessary to determine the relative speed (v) of the movement. In this section, we show how we can use the Doppler effect for estimating v.

Doppler effect states that if there is a relative movement between the sender and the receiver, the frequency of the received signal will shift by $f = \frac{v}{v_a} f_0$ where f_0 is original frequency, v is the relative speed between sender and receiver and v_a is the speed of the acoustic signal. For example, the sound speed is $340m/s$ at $25°C$, and if the original frequency of the acoustic signal is $20kHz$ and the frequency shift is 1 Hz, the speed of the relative movement can be calculated as $\frac{1*340}{20k} = 0.017\ m/s = 1.7\ cm/s$.

In real-world user mobility scenarios, the estimation of v is not straight forward even using Doppler effect. This is because user's motion is not uniform especially when the user is walking. If the vouching device is user's smartwatch, the watch swings back and forth on user's arm while walking. Similarly, if the user's smartphone is the vouching device, it also swings back and forth while being in user's pocket.

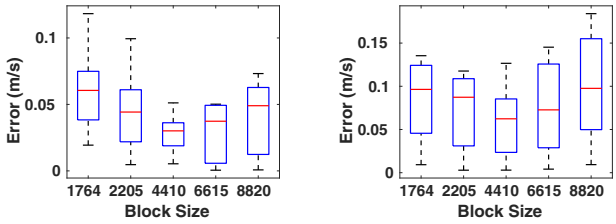
We investigate the two common scenarios of human walking: 1) with Samsung Galaxy S6 phone in the pocket; 2) with an Apple watch on user's wrist. The signal used for detection of Doppler shift is generated at a frequency of 20.5 kHz by another smartphone (Samsung Galaxy S5). Figs. 5(a) and 5(b) show the spectrogram for both scenarios and resultant variations in frequency. It is clear that the effect of swinging motion and non-uniform speed of walking is significant and needs to be addressed.

To get a better estimate of average speed for a period of time, we could split the period into shorter periods and estimate the speed for each short period to make the speed estimate in real-time. However, the short period will result in a smaller block with fewer samples of acoustic signal, which in turn reduces the frequency resolution. The frequency resolution is calculated as F_s/N where N is the number of samples of the acoustic signal within the block and F_s is the sampling rate. For example, performing speed estimation at every 200 ms, the available samples are 8820, the frequency resolution is $44100/8820 =$ 5 Hz and corresponding speed estimation resolution is $5 * 1.7\ cm = 8.5\ cm$. This way, there exists a trade-off between the speed estimation resolution and the rate of speed estimation.



(a) Spectrogram - Walk with phone in pocket (b) Spectrogram - Walk with watch on hand

Fig. 5: Frequency shift due to user's walking activity



(a) Error - Walk with phone in pocket (b) Error - Walk with watch on hand

Fig. 6: Trade-off between speed estimation error and frequency resolution with different number of samples (block size)

We empirically determine the size of the block of the available samples that can achieve a balance between the real-time speed estimation and frequency resolution. In the experiments, the user carries a Galaxy S6 in her pocket and Apple Watch on her wrist at the same time to walk towards and away from a Galaxy S5 for 4 times. We process the recorded data with block sizes of $1764, 2205, 4410, 6615$ and 8820 samples which correspond the $40ms, 50ms, 100ms, 150ms$ and $200ms$ interval speed respectively. For each kind of interval, we average all the interval speed to get the estimate speed during that period. For the ground truth of speed, we mark the fixed distance (36 $inches$) on the ground for each step and measure the step times using the accelerometer data from another smartphone wrapped to user's chest. We use the accelerometer data to derive precise step duration and calculate the ground truth speed for the fixed distance. Fig. 6 shows the error in speed estimation with sample block sizes of 1764, 2205, 4410, 6615 and 8820. It is observed that block size of 4410 provides relatively lower mean error in speed estimation and variation compared to other block sizes. Thus, we use 4410 sample points (100 ms) in this work.

As mentioned before, to support multiple pairs of devices, we use frequency division multiplexing to split the total bandwidth into several channels. As the movement of the vouching device will cause a frequency shift, it is required to ensure sufficient channel spacing. Based on our experiments, we find that frequency shift is no larger than 400 Hz, which leads to each channel's bandwidth to be 800 Hz.

C. Security Analysis

1) Zero-Effort Attacks: It can be defended against as long as the distance between the authenticating and vouching devices can be accurately detected. As shown in the evaluation results

in Section V, the proximity estimate error is less than 25 cm across all the devices used in our experiments. Therefore, PCASA can defend against the zero-effort attacks.

2) Spoofing Attacks: We conduct the security analysis for replay attack and relay attack respectively:

Replay attacks: In our protocol, the content of transmitted messages vary constantly except m_0 and m_1 in the initialization phase. Due to the loose synchronization between vouching and authenticating devices, the attacker cannot record the acoustic signals in one session of communication and spoof the authenticating device at a later time. Therefore, the only thing that an attacker can do is to record the acoustic signal from the vouching device, then jam the vouching device, and replay the signal immediately at a closer distance to the authenticating than the vouching device. Note that the attacker has to finish these actions in real time in order to conduct a successful replay attack.

In a simpler case that a user does not move, the message will be delayed by the attacker before it reaches the authenticating device, resulting in a larger arrival time. According to Equation (1), the estimated distance will increase. It indicates that the authenticating device will always obtain a larger proximity estimate, disallowing access to the attacker.

In the case when user (vouching device) is moving, we consider d^1_{AV} according to Equations (2) and (3)

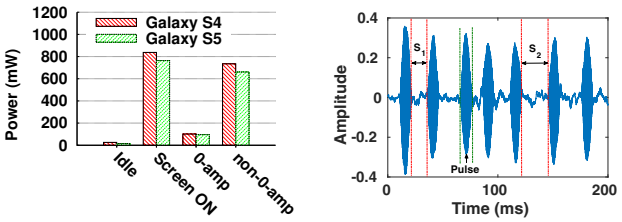
$$d^1_{AV} = (c - v)(t_{V1} - t_{V0}) - \frac{c}{2}(t_{A1} - t_{A0}) \tag{5}$$

When the attacker conducts replay attacks, the arrival time t_{V1} will be increased (assuming that $t_{A1} - t_{A0}$ is a constant since this is controlled by the authenticating device to decide the interval before sending the next signal $A1$ after receiving $A0$). Since, in practice, the speed of sound much larger than user's moving speed, i.e., $c \gg v$, d^1_{AV} will increase accordingly. This leads to the increase of d^2_{AV} according to the Equation (4). Thus, PCASA can defend against the replay attacks.

Relay attacks: In this case, the malicious devices have to be close enough to both vouching device and authenticating device to make the relay attack work. This is because the malicious device need to record the signal before sending it through the fast channel. Now that the communication only happens when the vouching device moves in PCASA, the attacker has to closely follow the user who is equipped with vouching device. However, this is impossible to remain undetected at all times without attracting user's attention. Therefore, the mechanism of PCASA can naturally defend against relay attacks.

IV. Energy Efficient Distance Estimation

Since the proximity-based authentication needs to be performed continuously, it is required that the acoustic communication is energy efficient. In this section, we show how Differential Pulse Position Modulation (DPPM, proposed in [16]) can be used to meet following two requirements: (1) decrease the energy consumption significantly compared to the previous modulation schemes for ultrasonic signals; (2) it should be possible to implement modulation and demodulation on devices with limited computational capability (such as wearables).



(a) Power Consumption (b) DPPM Symbols

Fig. 7: Energy Efficient Modulation: (a) shows the power consumption of different components on smartphone. 0-amp: speaker plays zero-amplitude sound, non-0-amp: speaker plays non-zero-amplitude sound. (b) A DPPM symbol is the zero-amplitude duration between two non-zero-amplitude pulses.

A. Modulation

The challenge with utilizing the acoustic communication is that speaker consumes considerable energy in devices such as a smartphone. We measure the power consumption of speaker using a Monsoon power monitor on Samsung Galaxy S4 and S5 smartphones. The results of power consumption are shown in Fig. 7(a). The devices are placed in the airplane mode during the power measurements. We compare the power consumption of Screen ON, Idle (with Screen OFF), Speaker playing *0-amplitude* sound (with Screen OFF) and Speaker playing *non-0-amplitude* sound (with screen OFF). We observe that speaker's average power consumption when playing no-0-amplitude sound is significantly higher and closer to that of *Screen ON*. However, the power consumption of speaker when playing 0-amplitude sound is much lower and close to that of Idle.

There are two properties of current smartphone speakers that motivate the use of DPPM - (1) Speaker can play very short duration of sound ($< 10\ ms$), (2) It consumes very small amount of power when the speaker is producing a sound with zero amplitude. DPPM modulates the data by varying the 0-*amplitude time* between the non-zero amplitude acoustic signals. As shown in Fig. 7(b), s_1 and s_2 are two DPPM symbols which are 0-amplitude time periods of different duration distinguished by a short non-zero amplitude acoustic signal (referred as pulse or delimiter).

1) Inter-Symbol Pulse: While using acoustic signal for continuous proximity estimation, it is required that the users of the devices do not perceive/hear any sound. It is known that human ear can hear sound in the range of [20 Hz, 20 KHz], however, the sound above 17 KHz is typically inaudible [17]. Hence, we use the frequency band [17 KHz, 22 KHz] for producing inter-symbol pulse. Since speakers are electromechanical devices, abrupt change in amplitude and frequency causes speakers to produce an audible "click" noise in practice [18]. To address this, we use a double sideband amplitude-modulation signal as the pulse whose outline serves as a signal envelope. The envelope signal concentrates the power more at the center frequency and reduces the audible artifacts at the lower frequencies, eliminating the click noise.

2) DPPM Symbol: We assume a simple linear constellation for DPPM symbols in this work. Let's denote the length of the first symbol s_0 as T_0, then the duration of the time symbols can be represented as $\{T_i | T_i = T_0 + i\delta, i = 0, 1, \cdots, N-1\}$, where

δ is the minimum difference of two adjacent DPPM symbols and N is a power of two. If we assume that each symbol will appear with equal probability, the average duration of a time symbol is $T_0^{(s)} + \frac{(N-1)\delta}{2}$. Denoting the length of pulse as T_d, the total transmission time of the message with L bits will be $T_{msg} = \left(T_d + T_0 + \frac{(N-1)\delta}{2}\right) \frac{L}{log_2 N} + T_d$.

3) Message Length: Based on the DPPM symbol duration, we can now determine the length of the message as it was shown in Fig. 3(b). The first part of the message is a timestamp which is used to ensure the order of the messages and prevent the replay attack. The length of timestamp is chosen to be 26 bits, which guarantees unique timestamps for 2 years at a resolution of one second. This is sufficient to prevent the replay attack. The second part of the message is the time difference which is represented by the count of sample points. Its length is chosen to be 18 bits to represent number of samples upto 5 seconds (44100×5). This way, the total length of the message is 44 bits. Since the second derivative of T_{msg} is positive when $N \geq 2$, T_{msg} is convex. Based on this, we find that order of 16 could achieve the shortest message duration ($505\ ms$) among all the orders of DPPM. This means that according to the protocol description in Section III-A, the device could perform two consecutive authentications in just over 1 second ($505 + 505 = 1010\ ms$).

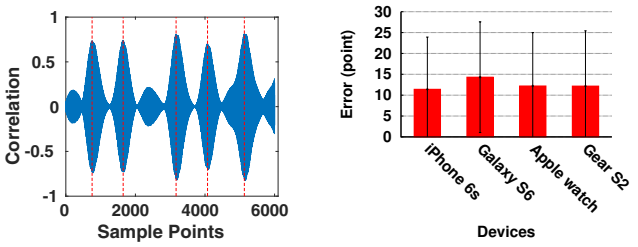
B. Demodulation

At the receiver side, the transmitted signal is demodulated by first applying a frequency filter and then detecting pulses.

1) Pulse Detection: To overcome the effects of background noise, we apply a bandpass filter to the incoming signal to only let the frequency of the current channel pass. After applying the bandpass filter, we detect the pulses by calculating the correlation between the received signal and the reference pulse. The correlation is calculated by sliding a window of size w (set equal to the length of pulse) over the received signal with step one.

2) Multipath Removal: The transmitted sound signal is reflected from surrounding objects, arriving at the receiver from multiple paths. This multipath effect introduces additional challenge in demodulating the received signal. The reflected signal can arrive immediately after the line-of-sight signal (also the shortest path signal) or can be delayed for a long time depending on the position of the reflecting object. The severely attenuated signal could be easily removed as the correlation is much small than the first incoming signal. We use a threshold of $0.5 \times max(r)$ to filter all small correlations. The receiver uses the first peak of correlation as the start point of the pulse.

Figure 8(a) shows the result of pulse detection. To get the ground truth of the start of pulse, we use the transmitter's original entire message signal to do cross correlation with the recorded signal. As shown in Fig. 8(b), the detection error on four devices (Samsung Galaxy S6, iPhone 6S, Samsung Gear S2 and Apple watch) is no larger than 30 sample points. Please note that start of the first pulse of each massage is the message arrival time which will be used for measuring proximity between the two devices.



(a) Result of Pulse Detection (b) Detection Error of Pulse Start

Fig. 8: Pulse Detection. (a) We use threshold to filter all the small correlation which corresponds to the second and later arrival signals as they are severely attenuated. (b) We compare the detected start of pulse with the ground truth on four devices, the error is less than 30 sample points.

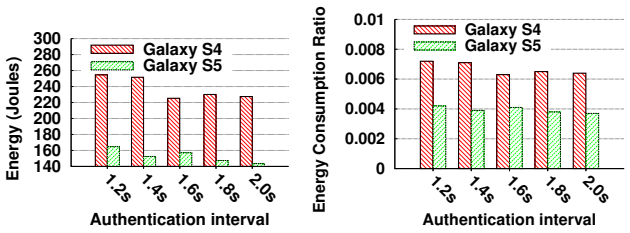
V. Evaluation

We evaluate the performance of our proximity based authentication scheme PCASA from both energy consumption and accuracy of distance estimation aspects.

Experiment Setup and Implementation In our experiments, we use the following devices - Samsung Galaxy S4 (1), Samsung Galaxy S5 (2), Samsung Galaxy S6 (1), iPhone 6S (1), Apple Watch (1), Samsung Gear S2-LTE (1). All the devices are equipped with a speaker and a microphone. In all our experiments, the acoustic signal is generated at 20 kHz and the speaker is set at the highest volume. The sampling rate of the microphone for recording is set to $44.1\ kHz$. We implement an application on the smartphones that performs the DPPM modulation and demodulation as well as the proximity calculation. The message used in the experiment is 44 bits long which consist 26 bits timestamp and 16 bits time difference (represented by the count of samples) as shown in Fig. 3(b). We will discuss the roles of each type of devices with individual experiments.

Energy Consumption Because only Samsung Galaxy S5 and S4 smartphones can be interfaced with Monsoon power meter for accurate real-time power consumption measurement, we use these smartphones for energy consumption experiments. We measure the energy consumption of PCASA with different authentication speeds, i.e., one authentication every $1.2, 1.4, 1.6, 1.8$ and 2 seconds. For each authentication rate, we measure the energy consumption for 15 minutes on the devices. The result is shown in Fig. 9(a). We also find ratio of authentication energy consumption to the total battery energy capacity for both the devices. The energy consumption ratio is presented in Fig. 9(b). As expected, lower authentication interval results in higher ratio. Based on the ratio, we can find that Galaxy S4 (which consumes more energy than Galaxy S5) can perform continuous authentication for up to $1/0.007 * 15$ minutes = 34 hours at a rate of one authentication every 1.2s.

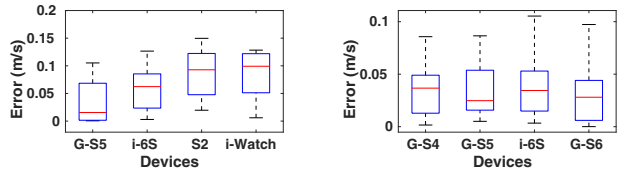
Speed Estimate To estimate the speed based on Doppler effect, we consider two scenarios, 1) devices on the wrist, 2) devices in the pocket. For the first scenario, we use Galaxy S6 as transmitter while using Galaxy S5, iPhone 6S, Gear S2 and Apple watch as the receivers. For the second scenario, the receivers are changed to Galaxy S4, Galaxy S5, Galaxy S6 and iPhone 6S. Galaxy Nexus is used as a reference device



(a) Energy consumption at different authentication rates (b) Energy Consumption Ratio

Fig. 9: PCASA energy consumption on smartphones

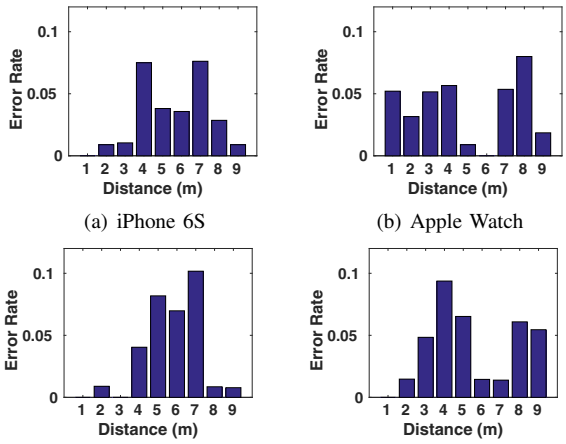
to record the accelerometer data for calculating ground-truth speed as discussed in Section III-B2. The transmitter sends 44 bits messages every 1.2 seconds, meanwhile, the user carries the receiver and moves towards and away from the transmitter for 10 rounds. We then calculate the average speed during each period. Fig 10 shows the error of speed estimate in comparison with the ground-truth speed. It show that the wrist-worn devices have relatively higher estimation error than the devices in the pocket. The reason is because the swinging motion causes higher speed fluctuations compared to walking activity. As we cannot monitor the real-time fluctuation due to the limitation of frequency resolution, the estimate error of swinging motion could be higher than reported. However, we find that the speed estimation error on all devices in our experiments does not exceed 0.15 m/s.



(a) Device worn on wrist (b) Device carried in pocket

Fig. 10: Speed Estimate Error

DPPM Symbol Error To estimate the performance of modulation and demodulation of DPPM, we conduct experiments to test the DPPM symbol error rate on iPhone 6S, Galaxy S6, Apple Watch and Samsung Gear S2. We group iPhone 6S and Apple Watch as one pair and the other two devices as the other pair. Each device serves as both the transmitter and receiver. In the experiment, we fix the position of iPhone 6S and Galaxy S6 at the same position, and move the Apple watch and Gear S2 from 1 m to 9 m. Each transmitter sends the 44 bits messages every 1.2 seconds for one minute. After the receiver demodulates the signal, the error rate could be calculated based on the original message from the transmitter. It is observed in Fig. 11 that the error rate on all devices is less than 0.1. It is interesting to observe that the error does not increase along with the distance as multipath plays a more important role in accurate demodulation. As shown in Fig. 11(a) and 11(c), the iPhone 6S and Galaxy S6 have similar error rate pattern (higher error rate in the center) while both Apple Watch and Gear S2 have the opposite pattern. This is most likely due to the multipath effect. As we fix the two smartphones at the same position, they experience similar multipath effect which results in similar error rate.



(a) iPhone 6S (b) Apple Watch

(c) Galaxy S6 (d) Gear S2

Fig. 11: Symbol Error Rate

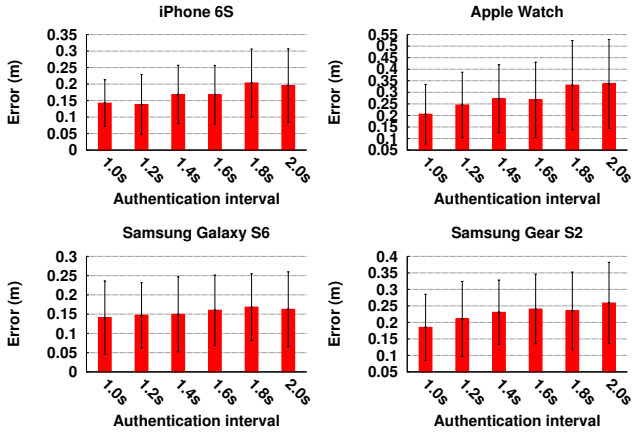


Fig. 12: Proximity estimation error in mobile scenarios

Proximity Estimation For proximity detection, we conduct experiment under the scenarios where the vouching device moves while the authenticating device stays stationary. The experiment is based on two pairs of devices (1) an iPhone 6S and an Apple watch and (2) Samsung Galaxy S6 and Gear S2 watch. The phones serve as authenticating devices while the watches act as the vouching devices. The authenticating device is kept at one fixed position and the user carries the vouching device and moves towards and away from the authenticating device. Five different authentication intervals are considered, *i.e.* 1.2, 1.4, 1.6, 1.8 and 2 seconds. For each kind of interval, the vouching and authenticating devices alternately send one message every authentication interval and stop after sending 20 messages. Fig. 12 shows the proximity estimate error for each device. For each interval, the figure shows the average error with variation over 20 messages. On each device, the proximity estimation error increases along with the authentication interval. The results are in agreement with Eqs. (2) and (4) which show the proximity estimate is related with the speed estimate and the message interval/authentication frequency. It can be observed that average error of proximity estimation is no more than $0.25m$ even when the user is mobile.

VI. Related Work

Proximity Detection: There are numerous works applying acoustic signal to measure distance based on Time of Arrival (TOA), Time Difference of Arrival (TDOA), and so on. Many existing research have developed on acoustic localization techniques [6]–[9], [19]. Techniques proposed in [7] requires custom-built hardware which makes them much less practical. Other approaches [8], [9] only achieves resolution in meters. Peng et al. [6] proposed an acoustic signal based protocol to estimate the distance between two devices by avoiding time synchronization and it could achieve centimeter-level accuracy. Although this work showed the accurate ranging with centimeter errors, it is demonstrated under ideal circumstances including no user mobility, unlimited battery power, and sufficiently long computation time. Moreover, these techniques are vulnerable to many attacks such as man-in-the-middle attack, therefore not suitable for applications with high security requirements.

Authentication Methods: As one of the most widely used authentication method, password suffers from various security and usability issues [20]–[22]. Since users have high cognitive load to remember password for different devices [21], some users intends to reuse one password [20] that makes it even easier for attackers to guess [22]. As an alternative of password based authentication, more and more research has focused on biometric authentication recently [23], which unfortunately also suffers from different security and usability problems. [24] showed that it is possible to trick fingerprint readers by forming a mold that can imitate a finger. Meng et al. [25] found that an attacker can be trained to imitate a user's keystroke dynamics behaviors. Serwadda et al. [26] showed that touch-based authentication approaches are also vulnerable to forgery attacks where an attacker programs a robot to replay collected touch strokes. Proximity based authentication or access control has attracted more attention recently. Most of the proximity based authentication apply ambient RF signal as proof of physical proximity for co-located devices [4], [5], [27]. As the RF signal fluctuates significantly spatially and temporally, Rasmussen et al. [14] proposed a proximity-based access control scheme for implantable medical devices using acoustic signal. Recently, another work [12] attempted to mimic the NFC with acoustic signals. Since it is designed for very short-range (several centimeters) communication, this work cannot be directly used in our focused applications (e.g., authentication, secure notification, etc.) that usually needs the estimation of larger distances.

VII. Conclusions

In this paper, we presented PCASA, a proximity based continuous and secure authentication scheme for personal devices. We showed that the proximity of wearables on user's body from her personal device such as a smartphone can be used for authenticating the personal device. PCASA utilizes Differential Pulse Position Modulation for energy efficient acoustic communication and performs an accurate distance estimation even when the user is mobile. Evaluation shows that PCASA can enable continuous authentication with 25 cm proximity estimation error in presence of user mobility. As part of the future work, we plan to further develop DPPM, which can be used for energy efficient high data rate communication between the wearables and the portable devices. We will also extend the application of proximity estimation service beyond the authentication purposes for customizing applications and configuring device settings and preferences.

References

[1] "Android SmartLock." https://get.google.com/smartlock/.

[2] "Bluetooth Proximity Profile." https://developer.bluetooth.org/TechnologyOverview/Pages/PXP.aspx.

[3] A. Srivastava, J. Gummeson, M. Baker, and K.-H. Kim, "Step-by-step detection of personally collocated mobile devices," HotMobile '15.

[4] A. Kalamandeen, A. Scannell, E. de Lara, A. Sheth, and A. LaMarca, "Ensemble: cooperative proximity-based authentication," MobiSys '10.

[5] S. Mathur, R. Miller, A. Varshavsky, W. Trappe, and N. Mandayam, "Proximate: proximity-based secure pairing using ambient wireless signals," MobiSys '11, pp. 211–224, ACM, 2011.

[6] C. Peng, G. Shen, Y. Zhang, Y. Li, and K. Tan, "Beepbeep: a high accuracy acoustic ranging system using cots mobile devices," SenSys'07.

[7] L. Girod, M. Lukac, V. Trifa, and D. Estrin, "A self-calibrating distributed acoustic sensing platform," SenSys '06, ACM, 2006.

[8] C. V. Lopes, A. Haghighat, A. Mandal, T. Givargis, and P. Baldi, "Localization of off-the-shelf mobile devices using audible sound: architectures, protocols and performance assessment," *ACM SIGMOBILE Mobile Computing and Communications Review*, 2006.

[9] J. Scott and B. Dragovic, "Audio location: Accurate low-cost location sensing," in *PerCom '05*, pp. 1–18, Springer, 2005.

[10] N. B. Priyantha, A. Chakraborty, and H. Balakrishnan, "The cricket location-support system," in *MobiCom '00*, pp. 32–43, ACM, 2000.

[11] A. Harter, A. Hopper, P. Steggles, A. Ward, and P. Webster, "The anatomy of a context-aware application," *Wireless Networks*, vol. 8, no. 2/3, pp. 187–197, 2002.

[12] R. Nandakumar, K. K. Chintalapudi, V. Padmanabhan, and R. Venkatesan, "Dhwani: secure peer-to-peer acoustic nfc," in *ACM SIGCOMM Computer Communication Review*, vol. 43, pp. 63–74, ACM, 2013.

[13] G. E. Santagati and T. Melodia, "U-wear: Software-defined ultrasonic networking for wearable devices," MobiSys '15, 2015.

[14] K. B. Rasmussen, C. Castelluccia, T. S. Heydt-Benjamin, and S. Capkun, "Proximity-based access control for implantable medical devices," in *CCS '09*, pp. 410–419, ACM, 2009.

[15] D. A. Bohn, "Environmental effects on the speed of sound," *Journal of the Audio Engineering Society*, vol. 36, no. 4, pp. 223–231, 1988.

[16] P. B. Kaplan, "Pulse-position modulation for signal identification," tech. rep., DTIC Document, 1972.

[17] S. Yun, Y.-C. Chen, and L. Qiu, "Turning a mobile device into a mouse in the air," in *MobiSys '15*, pp. 15–29, ACM, 2015.

[18] P. Lazik and A. Rowe, "Indoor pseudo-ranging of mobile devices using ultrasonic chirps," SenSys'12, pp. 99–112, ACM, 2012.

[19] W. Huang, Y. Xiong, X.-Y. Li, H. Lin, X. Mao, P. Yang, Y. Liu, and X. Wang, "Swadloon: Direction finding and indoor localization using acoustic signal by shaking smartphones," *IEEE Transactions on Mobile Computing*, vol. 14, no. 10, pp. 2145–2157, 2015.

[20] J. Bonneau, "Fawkescoin: A cryptocurrency without public-key cryptography (transcript of discussion)," in *Cambridge International Workshop on Security Protocols*, pp. 359–370, Springer, 2014.

[21] J. Bonneau, C. Herley, P. C. van Oorschot, and F. Stajano, "Passwords and the evolution of imperfect authentication," *Commun. ACM*, vol. 58, pp. 78–87, June 2015.

[22] J. Ma, W. Yang, M. Luo, and N. Li, "A study of probabilistic password models," in *Security and Privacy (SP), 2014 IEEE Symposium on*, pp. 689–704, May 2014.

[23] A. K. Jain, A. Ross, and S. Pankanti, "Biometrics: a tool for information security," *IEEE Transactions on Information Forensics and Security*, vol. 1, pp. 125–143, June 2006.

[24] "Fogery attacks to fingerprint." http://www.ccc.de/en/updates/2013/ccc-breaks-apple-touchid.

[25] C. M. Tey, P. Gupta, and D. Gao, "I can be you: Questioning the use of keystroke dynamics as biometrics," NDSS '13, 2013.

[26] A. Serwadda and V. V. Phoha, "When kids' toys breach mobile phone security," CCS '13, (New York, NY, USA), pp. 599–610, ACM, 2013.

[27] A. Varshavsky, A. Scannell, A. LaMarca, and E. De Lara, "Amigo: Proximity-based authentication of mobile devices," in *UbiComp '07*, pp. 253–270, Springer, 2007.