Secret Key Establishment via RSS Trajectory Matching Between Wearable Devices

Zi Li, Qingqi Pei, *Senior Member, IEEE*, Ian Markwood, Yao Liu, *Member, IEEE*, and Haojin Zhu, *Senior Member, IEEE*

***Abstract*— Recently, people have witnessed a remarkable growth in the number of smart wearable devices. Accompanied with the development of a contactless data transmission technique, the lack of effective secret key establishment between lightweight wearable devices which support contactless data transmission technique becomes a security bottleneck. In this paper, we propose a novel wireless key establishment method by moving or shaking the wearable wireless devices. Instead of received signal strength (RSS) itself, we denote the RSS trajectories of two moving wireless devices as the materials of secret key. Moreover, inspired by channel reciprocity in a channel feature-based key establishment technique, we propose the concept of reciprocity of RSS trajectory that guarantees that even when the RSSs of two devices are the same, the identical RSS trajectories of two devices can successfully generate the secret key. In addition, to effectively utilize the RSS trajectories, we design a novel quantization scheme by considering the entropy and efficiency of key generation. Furthermore, we analyze the security of this key establishment procedure in an eavesdropped and monitored environment. We also perform an evaluation of 64-, 128-, 192-, and 256-b key generation in indoor/outdoor environment, and the results indicate that the times are 0.22/0.33, 0.61/0.74, 0.95/1.02, and 1.28/1.46 s, respectively. In addition, the ranges of efficiency and entropy are 0.654–0.795 and 0.968–0.993.**

***Index Terms*— Key establishment, RSS trajectory, wireless devices, quantization.**

I. Introduction

WEARABLE devices equipped with sensors have been one of the remarkable outcomes in people's daily life over the past 10 years. These smart, wearable devices are gaining popularity and becoming an important part of e-healthcare, sports and fitness applications [31]. The devices like FitBit Flex, Nike+ Fuel band measure the person's physiological data, monitor activity and sleep quality, and sync wirelessly to the personal devices/base station (BS). The BS can then upload this data to a cloud based database to facilitate access by the hospital authority or caretakers for timely treatment. The lightweight, resource constrained wearable devices communicate with each other by using Wi-Fi, Blue tooth, UWB or other short range communication technologies. Wireless channels, upon which information is transmitted from one device to another, are public and can be accessed by wireless devices without authorization. This nature of wireless communications results in private information collected by wearable devices being transferred publicly, providing a potentially lucrative attack space. For instance, eavesdropping of confidential data and injection of malicious commands which can cause adverse effects on a person's health. Since these wearable devices handle sensitive health information, securing the information is crucial to ensure trustworthy and usable wireless communication. Intuitively and broadly speaking, wireless wearable devices must share a secret key to encrypt and decrypt messages during a wireless conversation.

Key establishment, the process wherein two individuals construct a secret key over a public medium, is fundamental in enabling wireless networking security through encryption. Traditional schemes used in current wearable devices are based on cryptographic technology, but these schemes require communication entities to be equipped with expensive specialized computing devices or chips because of their computational complexity. For example, Diffie-Hellman [10] cryptosystem, the oldest public key system still in use, allows two individuals to agree on a shared secret key, even though they can only exchange messages over public channels. Although active research efforts work to apply traditional cryptographic-based methods such as public key infrastructure (PKI) to wireless networks, these methods are not suitable for wearable devices. That is because these low-end wearable devices are required to be compact, non-intrusive and energy efficient. These requirements impose strict constraints on the resources available for sensor-node operation (transmission power, computation power, memory size, bandwidth, etc.).

Recently, researchers have begun constructing novel key establishment techniques uniquely applicable to the environment of wireless communication systems. These employ wireless channel characteristics which are unique based on the positioning of the involved devices, such as physical layer characteristics and received signal strength (RSS). Huang and Jiang [16] first present the concept of using physical layer characteristics of wireless channels for key establishment. Channel phase and channel impulse response (CIR) [45]

Manuscript received March 20, 2017; revised August 13, 2017 and October 18, 2017; accepted October 18, 2017. Date of publication October 30, 2017; date of current version December 19, 2017. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Walid Saad. *(Corresponding author: Qingqi Pei.)*

Z. Li and Q. Pei are with the State Key Laboratory of Integrated Services Networks, Xidian University, Xi'an 710071, China (e-mail: qqpei@mail.xidian.edu.cn).

I. Markwood and Y. Liu are with the Department of Computer Science and Engineering, University of South Florida, Tampa, FL 33620 USA.

H. Zhu is with the Department of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai 200240, China.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIFS.2017.2768020

are typical physical layer characteristics regarded as successful metrics to share keys between communication entities. [41] presents a practical opportunistic secret communication system, letting the legitimate sender communicate secret messages right away over wireless channels under the wiretap channel model.

These methods establish a shared key between Alice and Bob by exploiting wireless channel reciprocity property, which states that a transmitter and a receiver observe the same channel characteristics (e.g., RSS, CIR, etc.) from the wireless link between them at the same time. Note that existing approaches are required to operate at simplex communication mode. Specifically, Alice first sends a signal to Bob, who then measures the channel characteristics from the received signal and replies a signal to Alice, so that she can measure channel characteristic from Bob's signal.

In this paper, we posit that a secret key can also be extracted under the condition that two entities can transmit and receive signals at the same time. We find that under certain conditions, such as short time and distance, the path loss in the propagation model is a function of distance, if other variables remain the same. Intuitively, when the distance increases, the RSS of both entities decreases, and vice versa. This phenomenon inspires us to propose a new key establishment technique utilizing the variation trend of RSS between two devices, which we name RSS trajectory reciprocity. We propose a new key establishment method that is independent of channel selections and supports multiplex communication to enable Alice and Bob to capture common trajectory features simultaneously. Alice and Bob can send radio signals over two different frequency channels at the same time. Both frequency channels will exhibit the same trajectory feature, i.e., decreasing RSS when they move apart and increasing RSS when then they move close. Our proposed method, combining these two kinds of techniques together in some way, is a new method to get the advantages: richer source information from CIR-based methods and low computational overhead from RSS-based methods.

The proposed key establishment scheme is suitable to be applied in wearable devices which have frequent movements and short range communications. That is because the movements of the devices lead to a variation of distance, and well-researched wireless propagation models specify the relationship between distance and path loss, which forms the RSS trajectories measured. As shown in Figure 1, two devices are moved or shaken when their owners mean to establish a secret key. Once the devices sample the received signals and calculate the RSS trajectories, our designed mean-value quantization scheme is used to parse the RSS trajectories into bit sequences. Because of the complex wireless environment and other interferences (i.e. random noise), we detail an error correction scheme to correct the mismatch bits and enhance the security. Furthermore, we analyze the security of this key establishment procedure in an eavesdropped or monitored environment.

The first and crucial step is quantization. Although several methods have been proposed for key establishment via RSS using one or two fixed thresholds, these are not suitable for

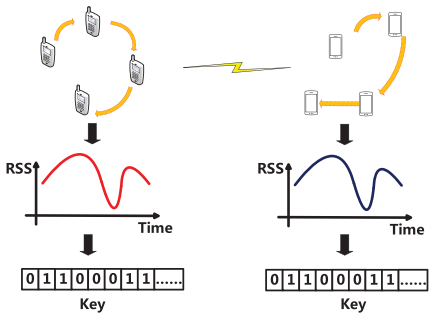


Fig. 1. Basic principle of the proposed method.

parsing RSS trajectories. Because devices moving while measuring RSS will observe an increase or decrease of RSS over time, but neighboring samples will have similar values due to the high sample rate relative to physical movement. Also, the environment and device movement may cause interference for RSS, so the quantization scheme needs to be reasonably flexible and resistant to interference without being lax in security. Next, the resultant bit sequence calculated by each device can not be used directly as the secret key because of mismatched bits. While the RSS trajectory reciprocity property hypothesizes the changes in measured RSS should be equivalent, a small number of mismatched bits may exist which must be corrected through information reconciliation and privacy amplification to generate the final identical secret key.

Our experimental results indicate that the proposed key establishment method can dynamically generate a secret key of various lengths for a pair of moving devices. We evaluate generation of 64, 128, 192, and 256-bit keys, which can be generated in indoor/outdoor environments in 0.22s/0.33s, 0.61s/0.74s, 0.95s/1.02s, and 1.28s/1.46s, respectively. We also measure the efficiency of our technique, denoted as the ratio between the actual bits and the required key length. Range of efficiency is from 0.795 to 0.654 in the indoor environment and 0.754 to 0.664 in the outdoor environment.

The main contribution of this paper is three-fold. Firstly, we propose the concept of RSS trajectory reciprocity, where the RSS measurements of two devices have the same fluctuations as they are moved relative to each other. Secondly, we propose a novel key establishment technique for pairs of devices, which functions of the movement of both and a "virtual" full-duplex mode. This method could obtain high efficiency with a relative low cost. We offer a lemma analyzing and proving the rationality and demonstrate its efficacy in generating secret keys. Thirdly, we design a novel mean-value quantization scheme to facilitate key generation, which divides the collection of samples into several sub-sequences instead of directly quantizing the sample values one by one. Two sub-sequence division methods are proposed. We analyze the security of our key establishment technique and prove that it can defend against eavesdropping because of the unpredictable RSS trajectories involved.

In what follows, we detail how we address the aforementioned technical challenges to establish a key for two devices. Section II establishes our research space, while Section III, IV and V present our research efforts.

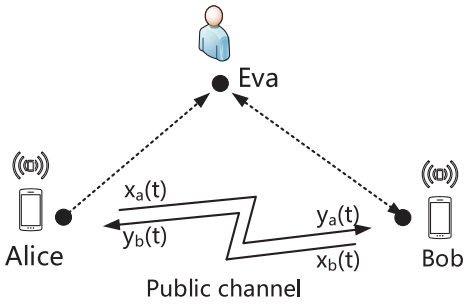


Fig. 2. System model.

Sections VI and VII discuss evaluation results and related work, respectively. Section VIII concludes this paper.

II. System Model and Assumptions

This section discusses the environment pertinent to the proposed key establishment technique, including the assumptions enabling the proposed technique.

A. System Model

Figure 2 describes the system model for the proposed key establishment technique under the possibility of eavesdropping. *Alice* and *Bob* represent the two communicating wearable devices who transmit and receive signals through a public channel, and *Eve* represents the adversary. Unlike typical traditional key establishment techniques, these two devices transmit and receive signals at the same time. When *Alice* starts transmitting signals to *Bob*, *Bob* is transmitting signals to *Alice* as well. The receiving modules of two entities receive signals as soon as the transmitting signals arrive.

When two parties wish to generate secret keys, the users shake or move the devices in a short interval (e.g. 1 or 2 seconds), and the movements of the devices lead to a variation of distance. Well-researched wireless propagation models specify the relationship between distance and path loss, which forms the RSS trajectories measured.

Application Space: The proposed key establishment technique, using motion trajectories, should be applied in a short range communication scenario, rather than between two long-distance devices due to the limited power and transmission range of wearable devices. This method can be applied to both LoS and NLoS scenarios. The barriers and obstacles between two wearable devices can cause the attenuation of wireless signals, but the RSSs trajectories of two devices will still change synchronously and the key establishment process is not affected by the attenuation of wireless signals. The proposed method can be combined with short range communication technologies, including Infra-red [21], Near Field Communication (NFC) [36], and recent 60GHz communications [19]. These short range technologies mainly focus on improving the transmission rate and communication throughput, but have not fully examined the security properties, especially the robust authentication between a transmitter and a receiver.

The application scenario is suitable for communication between wearable devices. Existing short range communication technologies include Infra-red [21], Near Field Communication (NFC) [36], and recent 60GHz communications [19]. These short range technologies mainly focus on the speed of communications, but ignore necessary security properties, especially authentication between transmitter and receiver. For example, near field communication (NFC) does not provide hardware support for encryption of transmitted data. Also, the accuracy of hardware embedded in wireless devices decreases when working in a long distance.

Despite these shortcomings, short range communications have become more and more popular. Several recent lines of mobile devices are equipped with various short range communication chips for small data transfers over short distances. These communications usually require a face-to-face interaction within 10 meters, in constrained spaces such as conference rooms in an office building. Because the communication range is short, the people communicated with are likely to be familiar friends, family members, merchants, or costumers. Given the potential privacy value of information that may be exchanged in such situations, their security is highly important.

B. Assumption

We assume that each of the two devices has the ability to transmit and receive signals at the same time. Several sensor chips, for example RS-485, RS-442,SP 3328 in Texas Instrument [4] and network card 82599 and chip MAX 289 in Intel [3], have support full duplex mode. These sensors including RF chips, sensors and network card are able to be utilized in wearable devices. What's more, several researches have been done in different fields on full-duplex Body Area Networks [12], [30] and sensor networks [20], [38]. So this assumption could be regarded as a common, reasonable assumption. It is impractical to assume all devices support full-duplex mode, such that all pairs of devices could communicate with each other simultaneously using only one frequency. But our concept of virtual full-duplex mode, utilizing two nearby frequencies to communicate, makes the assumption realistic. Specifically, if a device A uses frequency f_1 to transmit and f_2 to receive, device B consequently transmits at f_2 and receives at f_1.

During the key establishment process, we assume variables in the propagation model remain the same, with the exception of distance (and consequently path loss). This is reasonable because the proposed technique occurs in a short-range distance and over a short time (the time of shaking or moving devices is within 1 or 2 seconds).

We lastly assume that signals in frequency f_1 and those in f_2 do not interfere with each other. Modern technologies such as Code Division Multiple Access (CDMA) have the ability to address the problem and validate this assumption.

III. Preliminary: RSS Trajectory Reciprocity

As discussed briefly in the Introduction, RSS trajectory reciprocity is a new concept which is akin to channel reciprocity. This property is that two devices with full-duplex mode or "virtual" full-duplex mode will extract the same variation trends in RSS for signals received from each other.

Our key establishment technique is based on this property: if the two sets of RSS measurements have the same trajectories, the result of quantization can theoretically achieve the same bit sequence for use in key creation, as presented in Section IV.

Before basing our key establishment on this property, we must illustrate its existence. In this section, we first describe how the movement of two devices relative to each other forms measurable RSS dependent on distance and signal frequency only. Then we show in Lemma 1 the mathematical basis for RSS trajectory reciprocity based on mutual relative motion and a static frequency.

A. Motion of Two Devices

Two different types of relative device motions are considered here. One option is that only one device moves while the other is stationary, in scenarios such as using iPhone or Android devices [1], [2] to make payments at the checkout of supermarkets or stores. The second option is that both devices move, such as when two people wish to transfer photos and send messages using mobile apps.

In either case, the movement of the device leads to a change of distance between the two, and this change of distance affects the RSS of each device. Wireless channels have fundamental performance limitations due to signal attenuation, as is common knowledge, and moreover, they are extremely random and not easily analyzed. Thus the wireless signal propagation model is actually a collection of models addressing different circumstances. We focus on the short distance communication scenario, where the short range propagation model is suitable.

In the short range signal propagation model, the received signal strength falls off with distance. Without loss of generality, we consider the signal propagation in two typical wireless environments, outdoor and indoor. The proposed technique can be extended to other circumstances whose models find path loss dependent on distance.

Outdoor Signal Propagation: One of the most common models for outdoor signal propagation is the Okumura Model [14]. According to this model, the path loss in decibels (dB) is defined as

$$L(dB) = 69.55 + 26.16\log_{10}(f_c) - 13.82\log_{10}(h_{te}) - a(h_{re}, f_c) - (44.9 - 6.55\log_{10}(h_{te}))\log_{10}(d), \quad (1)$$

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 receiver's antenna heights, respectively, and $a(h_{re}, f_c)$ is a correction factor computed using h_{re} and f_c. In our key establishment scenario with devices such as smart phones and tablet PCs, the antenna heights of devices can be ignored because of their tiny size. After the extraction of the RSSs in the quantization step, the height of the antennas becomes a constant, which does not affect our results. We use Lemma 1 to validate this characteristic in Section III-B. In Lemma 1, the RSS trajectory at each sample is the first order derivative of the RSS , and the RSS trajectories of the two devices are both positive or both negative. For outside case, the first order derivative of RSS turns to $-(d(\ln 10))^{-1}$, where d is the distance. Note that the height of antenna does not affect the positive or negative signs.

TABLE I
SYMBOLS

f_a	Transmitted Frequency of A
	Received Frequency B
f_b	Transmitted frequency B
	Received Frequency A
$R_a(i)$	RSS of $i-th$ sample in devices A
$R_b(i)$	RSS of $i-th$ sample in devices B
d	Distance between A and B
L	Path Loss
P_a	Transition Power of A
P_b	Transition Power of B
N	$NumberofSamples$
D	$DistanceRange$

Indoor Signal Propagation: Indoor path loss is often represented by the ITU Indoor Propagation Model [26] as shown below:

$$L(dB) = 20\log f_c + \lambda \log d + P_f(N_f), \quad (2)$$

where λ is the empirical path loss at the same floor, N_f denotes the number of floors between the transmitter and receiver, and $P_f(N_f)$ denotes the floor penetration loss. Since, again, we focus on Line-of-Sight propagation, $P_f(N_f)$ is regarded as a constant here.

B. RSS Trajectory Reciprocity

We here illustrate the calculation of an RSS trajectory and then prove a lemma demonstrating RSS trajectory reciprocity. The symbols appeared in the lemma are described in Table I.

To refresh, RSS trajectory reciprocity is the property that the RSSs between two devices have the same variation trend during the same time of measurement, i.e. that the two RSSs measured by the two devices increase or decrease simultaneously and proportionally when the distance between them changes. However, simultaneous variation does not mean the two devices necessarily observe the same RSS. We point out that the RSS trajectory can be regarded as a function of RSS at each time quantum.

To find the trajectories of R_a and R_b, an efficient method observes the first derivative of the RSS "function" $f(t)$ (series of measurements) at each point t (measurement timestamp). Because the first derivative of f, written as $f'(t)$ or as $\frac{\partial f(t)}{\partial t}$, is the slope of the tangent to f at time t. It describes the change in $f(t)$ over the change of t.

Informally, the reasoning behind Lemma 1 is as follows. Since we have assumed f_a, f_b, P_a and P_b are fixed values during key establishment process, the distance d changes continuously in both outdoor and indoor environments shown in Equations 1 and 2. Actually, the distance d for each device is relative to the other and has the same value at each time for both devices no matter how the two devices moves. So the first derivative of R_a and R_b, denoted as R'_a and R'_b, have $R'_a(t) = R'_b(t)$ at any given time t. This means R_a and R_b have the same slope, or increasing/decreasing trend, at each time. Based on the above logic, the trajectory of R_a is the

same as that of R_b. The following lemma proof demonstrates this mathematically.

Lemma 1: If vector $\overrightarrow{R_a} = (R_a(1), R_a(2), R_a(i), \cdots)$, $i \in N$, and vector $\overrightarrow{R_b} = (R_b(1), R_b(2), R_b(j), \cdots)$, $j \in N$, then $R'_a(i) \cdot R'_b(j) \geq 0$ when $i = j$.

Proof: We assume f_a, f_b, P_a and P_b are fixed values during key establishment process. According to the concept of path loss [26], we denote the mapping g between RSS and distance as $g : N \to D$, $R(i) = P - L(d)$, for all $i \in N$, and all $d \in D$.

We denote R_a and R_b as

$$\begin{cases} R_a(i) = P_b(i) - L(i), & i \in N \\ R_b(j) = P_a(j) - L(j), & j \in N \end{cases} \tag{3}$$

From Equations 1 and 2, we know that path loss L is a function of center frequency and distance as, broadly, $L = F(d, f)$.

Thus, an equivalent formulation of Equation 3 is

$$\begin{cases} R_a(d) = P_b - L(f_b, d) \\ R_b(d) = P_a - L(f_a, d) \end{cases} \tag{4}$$

With $R'_a(d)$ and $R'_b(d)$ as the first derivatives of $R_a(d)$ and $R_b(d)$, respectively, and the assumption of fixed values f_a, f_b, P_a and P_b, we have

$$\begin{cases} R'_a(d) = -F'(d) \\ R'_b(d) = -F'(d) \end{cases} \tag{5}$$

Now we consider our two environmental cases.

Case 1: (Indoor Environment): We combine Equations 2 with 5 to simplify $R'_a(d)$ and $R'_b(d)$,

$$\begin{cases} R'_a(d) = \lambda(d(\ln 10))^{-1} \\ R'_b(d) = \lambda(d(\ln 10))^{-1} \end{cases} \tag{6}$$

where $d \neq 0$. Then we have

$$R'_a(d) \cdot R'_b(d) = \lambda^2(d(\ln 10))^{-2} > 0 \tag{7}$$

Case 2: (Outdoor Environment): We combine Equations 1 with 5 to simplify $R'_a(d)$ and $R'_b(d)$,

$$\begin{cases} R'_a(d) = -[d(\ln 10)]^{-1} \\ R'_b(d) = -[d(\ln 10)]^{-1} \end{cases} \tag{8}$$

where $d \neq 0$. Then we have

$$R'_a(d) \cdot R'_b(d) = (d(\ln 10))^{-2} > 0 \tag{9}$$

With Equations 7 and 9, we have proved the lemma for Cases 1 and 2, respectively. ■

Based on this lemma, we know the first derivative of R_a and R_b are the same at each point d within the overall domain of d. This means that R_a aligns with R_b as d changes, regardless of what the distance value is. We conclude, then, that the trajectory of R_a is the same as that of R_b.

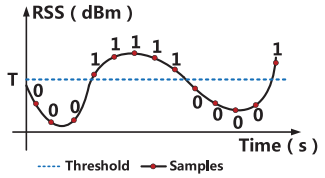


Fig. 3. One threshold.

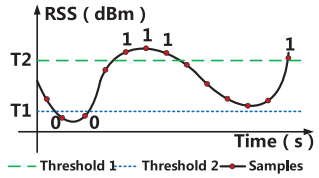


Fig. 4. Two thresholds.

IV. Key Sequence Generation

In the previous section we demonstrated the existence of the RSS trajectory reciprocity property. In this section, we present how to generate secret key sequences from two RSS trajectories. The proposed mean-value quantization scheme is described first. Then we utilize Bloom filter and Karhunen-Loeve Transform (KLT) to correct the mismatch bits and guarantee the randomness of key bit sequences. We also analyze the entropy of the proposed key generation scheme at last.

A. Mean-Value Quantization Scheme

1) Traditional Schemes: Quantization is the first step of wireless key establishment, often employing thresholds to parse the sample values into binary bits based on certain channel metrics. Traditional quantization schemes [22], [29] use one or two thresholds to quantize samples to binary 0 or 1. Two archetypal examples are shown in Figure 3 and 4. In both figures, each point represents a sample value. Figure 3 is the basic threshold quantization scheme with only one threshold T. When the sample value is larger than T, it is encoded as 1; otherwise the sample value is encoded as 0.

The quantization scheme shown in Figure 4 is a more advanced scheme which has two thresholds $T1$ and $T2$. Sample values larger than the higher threshold $T2$ are encoded as 1, and those smaller than the lower threshold $T1$ are encoded as 0. Other samples values located between $T1$ and $T2$ will be dropped, and security may be enhanced by randomly dropping additional sample values.

2) The Proposed Mean-Value Quantization: Traditional quantization schemes using thresholds are not suitable for our device-moving situation, for the following reasons. When we move or shake the devices in a short range communication scenario, the RSSs for two devices vary a lot, because the relative distance between the two devices increases or decreases significantly. Based on our physical experiments, the sample values vary notably (the range could be -70dBm to -30dBm). Thus, using a fixed threshold may generate a series of all 1 sequences or all

0 sequences. Consider an example where two devices undergo a motion away from each other. Suppose sample values are $V = (v(1), v(2), \cdots, v(i), \cdots), i \in \mathbb{Z}^+$, threshold is t. If $v(1), \cdots, v(i) > t$ and $v(i+1), \cdots, v(k) < t$. The result of quantization in this scenario is a series of all ones from the 1_{st} to i_{th} samples followed by a series of all zeros from the $(i+1)_{th}$ to k_{th} samples.

Instead of quantizing the sample values one by one, we design a mean-value quantization scheme, which divides the whole collection of samples into several sub-sequences, called intervals. We then calculate the mean-value and the middle point value of each interval. The middle point value, acting as a threshold, is the average of the values in two ends of a interval. And then each interval is encoded by comparing the mean value to the middle point value. A mean larger than the middle point results in a one while the reverse results in a zero. Algorithm 1 describes this quantization algorithm.

Algorithm 1 Mean-Value Quantization Scheme

Input: Sample values $\overrightarrow{S} = s_1, s_2, s_3, \cdots$, quantization interval l;
Output: A binary bit sequence $\overrightarrow{K}$;
1: $w = \overrightarrow{S}/l$;
2: **for** $i = 1 : w$ **do**;
3: $m_i = \frac{1}{l}\sum_{j=1}^{l}(w_{i,j})$;
4: $h_i = \frac{1}{2}(w_{i,1} + w_{i,l})$;
5: **if** $m_i < h_i$ **then**
6: $k_i = 0$
7: **else if** $m_i \geq h_i$ **then**
8: $k_i = 1$;
9: **end if**
10: **end for**
11: $\overrightarrow{K} \leftarrow k_1, k_2, k_3, \cdots, k_w$;

As shown in Figure 6, we consider device A wishing to establish a session key with device B. Then, device A begins transmission of a continuous signal which contains an impulse and a timestamp $t0$. Device B receives the impulse at time t1, and begins sending a continuous signal as well, containing an impulse at the timestamp $t2$. Then device A receives the impulse at time $t3$. Due to the "virtual" full-duplex communication mode, the travel time $t1 - t0$ of Signal A is equal to the travel time $t3 - t2$ of Signal B. Then device could calculate the signal travel time through $t1 - t0$. So the two devices will have an appoint time $t3$, which equals to $t2 + t1 - t0$.

3) Fixed and Dynamic Intervals: Two possibilities exist for partitioning the now-uncorrelated RSS data into intervals, by time length or by sample count. The former we refer to as fixed; the latter as dynamic, as shown in Figure 5. Specifically, the quantization scheme collects S samples over a duration of time T. Each device (A and B) has, specified by its owner, a preference on the length N of the resultant key (N_a and N_b). At this time, the interval type is chosen.

The fixed interval length l is defined as $l = \frac{T}{N}\sigma$, where the error compensation weight σ is used to allow for a certain

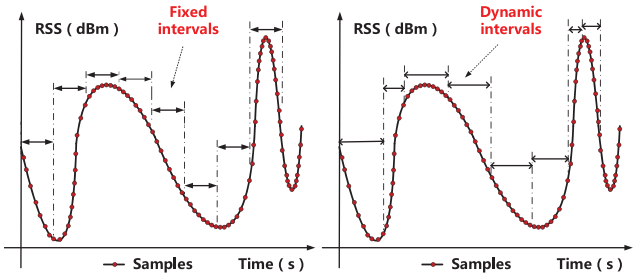


Fig. 5. Two intervals in quantization scheme.

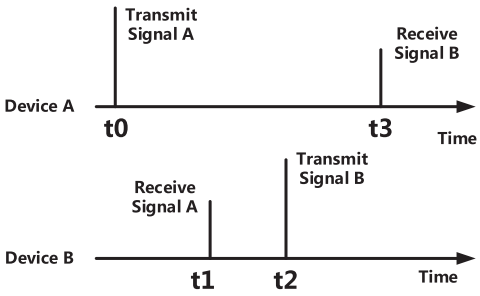


Fig. 6. Synchronization.

expected percentage of mismatched bits while guaranteeing the required secret key length. Key lengths and the statistical likelihood of mismatched bits are discussed in Section VI, and so consequently is the value of σ. Meanwhile, the size of each dynamic interval is set as $l = \frac{S}{N}\sigma$.

Selecting a suitable interval type depends on the type of motion the users will impart on their devices. With less erratic movement, the changes in RSS will be more constant with respect to time. Thus, fixed intervals are more suitable for slower movements, while dynamic intervals can handle faster motions.

B. Error Correction

According to the RSS trajectory reciprocity property, the transmitter and receiver should observe the same quantization output. However, due to imperfect reciprocity [6] and random noise, there may exist a small number of mismatched bits between the two outputs. Thus, error correction of the key sequence, including reconciliation and privacy amplification, is applied to achieve an identical final secret key.

1) Information Reconciliation: Information reconciliation is the process of finding and correcting mismatched bits of the quantization outputs of the two devices. Here we bring in the Bloom filter to help Alice and Bob find out and correct the bit mismatches.

2) The Bloom Filter: A Bloom filter [7] is a space-efficient probabilistic data structure, which can be used to test whether an element is a member if a set. More specifically, it is able to indicate an element is either *definitely not* in the set or *possible* in the set. Said differently, there are false positives but no false negatives. A Bloom filter is essentially a bit-array S of q bits, where $S = (s_1, \ldots, s_q,)$, and all the q bits are initially set to be 0s for an empty Bloom filter. There are also k hash

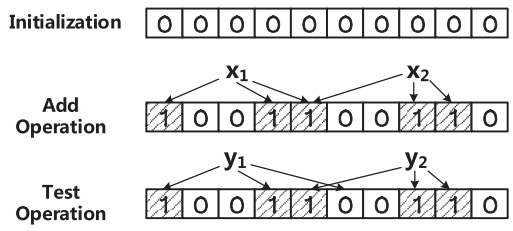


Fig. 7. An example of Bloom filter, where x_1 and x_2 form the set while y_1 and y_2 are the test elements. y_1 is in the set and y_2 is not.

functions h_i, for $1 \le i \le k$. Note that non-cryptographic hash functions are sufficient for implementing Bloom filters.

To add a new element x to a Bloom filter S, compute $h_i(x)$, and set $s_{h_i(x)}$ as 1, for $1 \le i \le k$, and indicate it is an element of set if these k positions in S are all in 1, which is,

$$\wedge_{i=1}^{k} = s_{h_i(y)} = s_{h_1(y)} \wedge \cdots \wedge s_{h_k(y)} = 1. \tag{10}$$

The above operations can be denoted as the add operation and the test operation, where the running time of each is $O(k)$. Figure 7 describes an example of Add and Test operations of a Bloom filter.

Figure 8 describes the main idea of the proposed information reconciliation. To use Bloom filter in our proposed method, firstly, we call the bit sequence extracted from quantization as the original sequence. Specifically, K_a=(k_{a1}, k_{a2}, …, k_{al}) and K_b=(k_{b1}, k_{b2}, …, k_{bl}) are the two original sequences of devices A and B. Each of the sequences has l binary bits.

Then, the bit sequences K_a and K_b are divided by every 10 continuous bits, respectively. Each of the 10 continuous bits is regarded as a block, or an element, which is the input of a Bloom filter. So the results after this operation of sequences K_a and K_b are $K_a^{'} = (k_{a1}, k_{a2}, \ldots, k_{al'})$ and $K_b^{'} = (k_{b1}, k_{b2}, \ldots, k_{bl'})$, where $l^{'} = l/10$.

Thirdly, the Bloom filter embedded in each device calculates the designed hash functions with $K_a^{'}$ and $K_b^{'}$ and fills the results in the bit-arrays S_{aq} and S_{bq} of q bits. Here S_{aq} is generated from the Bloom filter in device A and S_{bq} is generated from the Bloom filter in device B.

Finally, the two devices exchange the bit-arrays S_{aq} and S_{bq}. Utilizing the Test operation mentioned above, Device A tests whether each block of $K_a^{'}$ is in the set S_{bq}, and Device B tests whether each block of $K_b^{'}$ is in the set S_{aq}. According to the relative of ($K_a^{'}$, S_{bq}) and ($K_b^{'}$, S_{aq}), device A and device B will get the same results. For example, if device A tests that the $i-th$ block of $K_a^{'}$ are not in the set S_{bq}, it means those bits in K_a generating the $i-th$ block of $K_a^{'}$ are different from the bits with the same positions in K_b.

After finding out the mismatch bits, the device who wants to establish a secret key drops off the mismatch bits. Thus the two devices will obtain the same bit sequence K.

3) Discussion: The probability of false positive of a Bloom filter is [40]

$$P_{fp} = (1 - (1 - \frac{1}{m})^{kt})^t \approx (1 - e^{-kt/m})^k), \tag{11}$$

where t is the number of elements which has been added in the Bloom filter, m is the length of the Bloom filter and k is the number of hash functions. From the analysis in [25], we know that when $k = \ln 2 \cdot (m/n)$, the minimized probability of false positive $P_{fp} = (\frac{1}{2})^k \approx (0.6185)$. So let $P_{fp} \le \epsilon$, we can get

$$m \ge t \frac{\log_2(\frac{1}{\epsilon})}{\ln 2} = t \cdot \log_2 e \cdot \log_2(\frac{1}{\epsilon}) \approx 1.44t \log_2(\frac{1}{\epsilon}). \tag{12}$$

That is to say, to ensure the the minimized probability of false positive, the length m of the Bloom filter is at least $1.44 \log_2(\frac{1}{\epsilon})$ times larger than the number t of elements which have been added in the Bloom filter.

4) Privacy Amplification: After reconciliation, Alice and Bob agree on a common secret key sequence. Simply concatenating the bits generated from RSS does not necessarily produce a random secret key, as RSS measured from the wireless fading channel often has high correlation among successive measurements, which can lead to low randomness over time and therefore within the key establishment. Moreover, reconciliation leaks some information to an attacker. One countermeasure would be an increase in sampling time, to allow for more fluctuations to grow the randomness, but this would also detract from usability. We utilize Karhunen-Loeve Transform (KLT) [11] to augment the randomness of the key material by decorrelating the key bit sequence after information reconciliation. KLT, widely used in data analysis and compression, is a mathematical procedure whereby any complicated data set can be optimally decomposed into a finite, and often small, number of modes, which are obtained from the eigenvectors of the data autocorrelation matrix. A primary purpose of KLT is to reduce correlated information to its independent basis, which is by definition uncorrelated.

For our key establishment technique, suppose the key bit sequence $\mathbf{K}$=($K(1)$, $K(2)$, $K(i)$, $\cdots$)$^{\mathbf{T}}$, $i \in L$, where $K(i)$ denotes the i-th bit, and L is the number of bit sequence. Let $\mathbf{R}$ be the correlation matrix of the key bit sequence $\mathbf{K}$ given by

$$\begin{aligned}\mathbf{R} = E(\mathbf{K}\mathbf{K}^{\mathbf{H}}) &= E[\begin{pmatrix} K(1) \\ \vdots \\ K(L) \end{pmatrix} \begin{pmatrix} K(1)^* & \cdots & K(L)^* \end{pmatrix}] \\ &= \begin{bmatrix} E[K(1)K(1)^*] & \cdots & E[K(1)K(L)^*] \\ \vdots & \ddots & \vdots \\ E[K(L)K(1)^*] & \cdots & E[K(L)K(L)^*] \end{bmatrix}\end{aligned} \tag{13}$$

where E is the expectation operator and $E[K(j)K(j)^*$ is the autocorrelation of $K(j)$, and $E[K(j)K(k)^*$ is the crosscorrelation between $K(j)$ and $K(k)$, $j \ne k$. Note that $\mathbf{R}$ is Hermitian. Let the unitary matrix which diagonalizes $\mathbf{R}$ be defined as Φ such that

$$\begin{aligned} \Phi^{-1} &= \Phi, \Phi\Phi^H = \mathbf{I}, \\ \Lambda &= \Phi^H \mathbf{R} \Phi = \Phi^{-1} \mathbf{R} \Phi, \\ \Lambda &= \mathbf{Diag.}[\lambda_1, \lambda_2, \cdots, \lambda_L,]. \end{aligned} \tag{14}$$

Here, λ_i, $i = 1, \cdots, L$ are the eigenvalues of $\mathbf{R}$. We sort corresponding eigenvalues in a descending order, such that $\lambda_1 > \lambda_2 > \cdots \lambda_L$. Φ is called the KLT matrix and it decorrelates the key bit sequences $\mathbf{K}$. This can be seen when

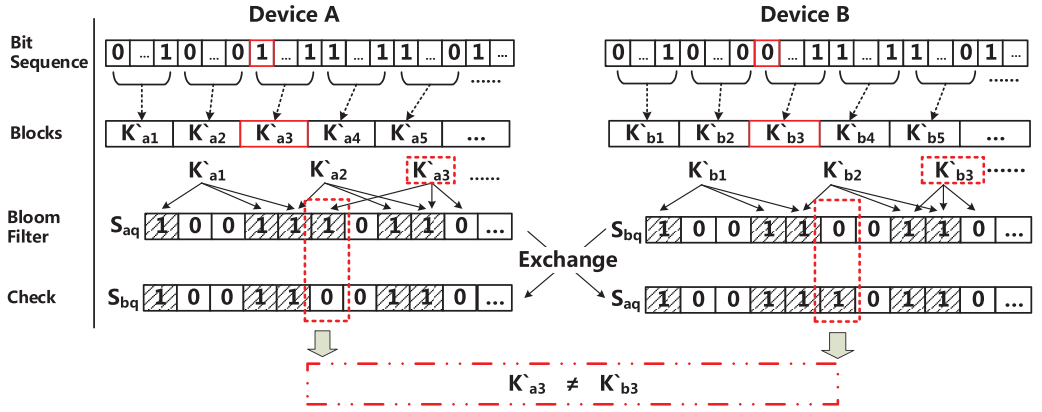


Fig. 8. The main idea of information reconciliation using Bloom fliter.

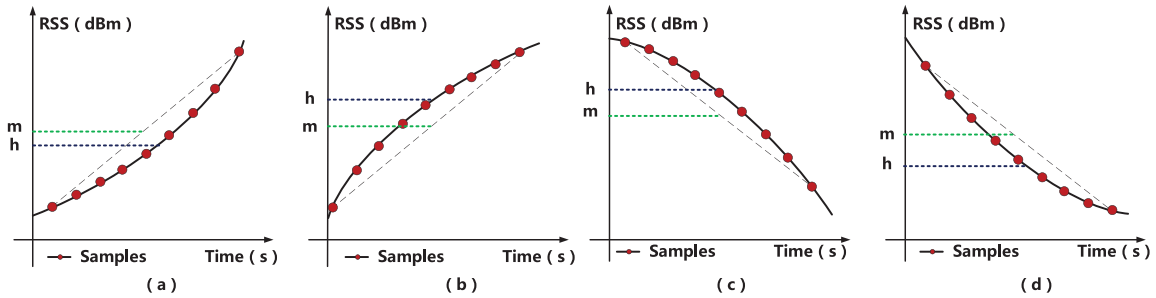


Fig. 9. The four types of distance change respective to increase and decrease of acceleration. Here, h denotes the midpoint and m denotes the mean. Subfigures (a) and (d) are concave function situations while subfigures (b) and (c) are convex function situations.

the forward and inverse KLT are considered. Let $\mathbf{K_{de}}$ be the forward transform of $\mathbf{K}$

$$\mathbf{K_{de}} = \begin{bmatrix} e_{11} & e_{12} & \cdots & e_{1S} \\ e_{21} & e_{22} & \cdots & e_{2S} \\ \vdots & \vdots & \ddots & \vdots \\ e_{S1} & e_{11} & \cdots & e_{SS} \end{bmatrix} (\mathbf{K} - M_K) \tag{15}$$

where $e_{ij}, i, j \in L$ means the the j-th component of the i-th eigenvector. $\mathbf{K_{de}}$ is an uncorrelated sequence and the final key sequence established by our proposed method.

C. Entropy

Entropy characterizes the uncertainty associated with a random variable and is used here to evaluate the security strength of the shared secret key. A higher entropy indicates a larger uncertainty for a random variable and thus a more difficult secret key to deduce. Entropy is defined as follows:

$$H = -\sum_{i=1}^{n} p_{x_i} \log p_{x_i} \tag{16}$$

where $x_i \in (x_1, , x_n)$, $(x_1, , x_n)$ are possible values of a discrete random variable X.

To analyze the relationship between entropy and the moving speed, we identify four types of distance change relative to type of acceleration. A distance can increase while the acceleration of the two devices away from each other increases or decreases, for example. The four cases are shown in Figure 9; there, h denotes the midpoint and m denotes the mean, the comparison of which determines the encoding as previously stated. Based on our quantization scheme, samples in the cases of Figure 9 (a) and (c) are encoded to binary 1, while samples shown in (b) and (d) are encoded to binary 0. We label the sampling rate r and sampling times for the four cases in Figure 9 (a), (b), (c), and (d) as t_a, t_b, t_c, and t_d, respectively. Then the total sampling time $T = t_a + t_b + t_c + t_d$, and for ease of display for like encodings we also write $t_1 = t_a + t_c$ and $t_0 = t_b + t_d$. The entropy calculation is dependent on the interval type chosen.

1) Dynamic Interval: As introduced in Section IV-A.2, The interval length $l = \frac{S}{N}$ where S is the total number of samples, and N represents the bit length. The numbers of bits output for each case are $\frac{rt_a}{l}$, $\frac{rt_b}{l}$, $\frac{rt_c}{l}$, and $\frac{rt_d}{l}$, so the number of bits set to 1 are $\frac{rt_a}{l} + \frac{rt_c}{l} = \frac{rt_1}{l}$, and the number of bits set to 0 are $\frac{rt_b}{l} + \frac{rt_d}{l} = \frac{rt_0}{l}$. So the probabilities of the two bit values are

$$\begin{cases} p_0 = \dfrac{r(t_b + t_d)}{lN} = \dfrac{rt_0}{lN} \\ p_1 = \dfrac{r(t_a + t_c)}{lN} = \dfrac{rt_1}{lN} \end{cases} \tag{17}$$

Combining Equations 16 and 17, we obtain the entropy as

$$H = -\frac{t_1}{T} \log \frac{t_1}{T} - \frac{(T - t_1)}{T} \log \frac{(T - t_1)}{T}. \tag{18}$$

2) Fixed Interval: The fixed interval length $l = \frac{T}{N}$, where T is the total sampling time. So the output bits number $\frac{t_a}{l}$, $\frac{t_b}{l}$, $\frac{t_c}{l}$, and $\frac{t_d}{l}$ in 4 cases, so the number of bits set to 1 are $\frac{t_a}{l} + \frac{t_c}{l} = \frac{t_1}{l}$, and the number of bits set to 0 are $\frac{t_b}{l} + \frac{t_d}{l} = \frac{t_0}{l}$.

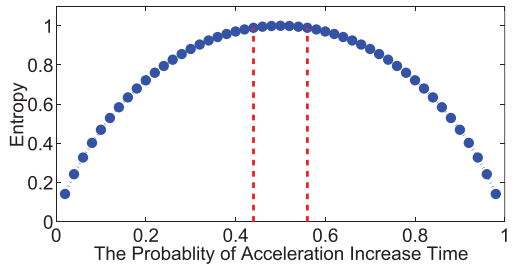


Fig. 10. The relationship between entropy and probability of t_{ac}.

The probabilities of the two bit values are

$$\begin{cases} p_0 = \dfrac{t_b + t_d}{lN} = \dfrac{t_0}{lN} \\ p_1 = \dfrac{t_a + t_c}{lN} = \dfrac{t_1}{lN} \end{cases} \tag{19}$$

Thus, using Equations 16 and 19, the entropy is

$$H = -\frac{t_1}{T}\log\frac{t_1}{T} - \frac{(T-t_1)}{T}\log\frac{(T-t_1)}{T}. \tag{20}$$

From Equations 18 and 20, we conclude that the entropy is only related to the time of acceleration increase and decrease. In Section VI, we will empirically measure the entropy to validate the strength of keys generated from RSS trajectories.

Figure 10 indicates the relationship between entropy and probability of the time of acceleration increase t_{ac}. The dots indicate the entropy values with different probabilities. The entropy first increases then decreases in the probability range (0, 1). Those dots between to dash lines represent the entropy $H > 0.9$. We find that with a total sampling time of T, the probability of time of acceleration increase must be in the range of 0.44 to 0.56 to ensure the entropy $H > 0.9$, as indicated in two vertical dash lines. If acceleration increases account for exactly half of T, the entropy will be maximized at 1. We consequently suggest movements cause an oscillating distance between devices when using the proposed key establishment mechanism. This allows for many distinct positive and negative acceleration events and a roughly equivalent ratio between them, maximizing the resultant key entropy.

V. Security Analysis

Our technique finds its security in the RSS trajectory reciprocity property and the randomness of relative devices locations, and namely the fact that on one side, RSS trajectories measured by the two communicators cannot be predicted with great ease by an adversary not co-located with either, even while monitoring the wireless channels and signals, on the other side, the randomness of noise and mismatch bits caused by the relative positions do the second layer security protection when the RSS trajectories are predicted. In this section, to analyze the security of proposed technique, the attack model is shown first. Then theoretical analysis is presented and an brief example is given. Them we further analyze the security under powerful attackers who are able to predict the relative distance of the two devices.

A. Attack Model

We consider an adversary M, who eavesdrops all the wireless communication between $Alice$ and Bob during key establishment. We assume the adversary has the ability to 1) measure wireless radio channels between itself and the two users when $Alice$ and Bob are communicating with each other; 2) obtain the secret key quantization algorithm and corresponding parameters for key sequence generation; and 3) cannot be very close to either $Alice$ or Bob (at least half of wavelength away).

Our key establishment method is applied in short range communication system, so the two devices are in a close, Line-of-Sight proximity. The adversary can receive signals in his transmission range, but must be close to the two devices to pinpoint their exact locations. Approaching the two devices increases the likelihood of discovery by normal users. Furthermore, the movement interaction between devices usually lasts less than 1 or 2 seconds, which greatly complicates the logistics of an attacker locating both devices.

Finally, the property of spatial decorrelation makes it impossible for an illicit device located further than $\lambda/2$ from a legitimate device to measure the same wireless channel. Therefore, even if an adversary can measure the RSS trajectory of a legitimate device, this trajectory will not exhibit the same pattern as that measured by the legitimate device, and the same secret key will not be extracted.

We demonstrate the security in two situations. 1) defense of RSS trajectories prediction when one of the devices' location is given. 2) defense of key sequence prediction from the randomness of noise and mismatch bits caused by relative devices locations, when the approximate locations of both devices are obtained by an adversary

B. 1st Situation: Defense of RSS Trajectories Prediction

Considering (x_a, y_a), (x_b, y_b) and (x_m, y_m) as the coordinates of devices A, B, and adversary M, respectively, we wish to prove that if M knows one device's location (without loss of generality) (x_b, y_b), he cannot obtain the other device's location (x_a, y_a).

Again, we reasonably assume the transmit and receive power, gains, and frequencies are fixed values during the one time key establishment. Then, from Equations 1 and 2, we treat all the elements except the distance d as a constant C, finding the relationship between RSS R and distance d as $R = P_t - C\log(d)$, where P_t is the transmit power. As previously discussed, distance d is continuously changing when $Alice$ and Bob shake their devices. Thus, in order to obtain the RSS trajectory between them, the adversary has to discover the distance d at all times t, in order to calculate how d changes with t.

Figure 11 illustrates possible locations of the three entities at some time t. Building from our attack model, M can know the coordinates of one device, say (x_b, y_b) of B without loss of generality, but does not know coordinates (x_a, y_a) of A.

As shown in Figure 11, the triangle formed by A_1, B and M. Here, by the law of cosines [8], A_1B is

$$A_1B = \sqrt{(A_1M)^2 + (BM)^2 - 2(A_1M)(BM)\cos\alpha} \tag{21}$$

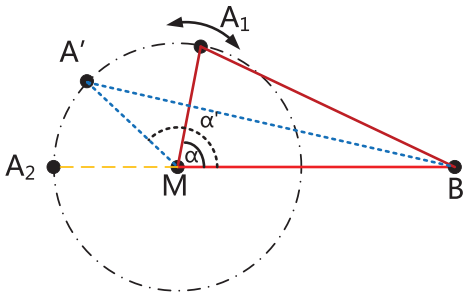


Fig. 11. The relationship between three entities at time t.

Because M knows (x_b, y_b), (x_m, y_m), thus Equation 21 becomes a (x_{a1}, y_{a1}) function F of angle α as

$$(x_{a1}, y_{a1}) = F(\alpha) \tag{22}$$

This function is a ternary linear equation. To guarantee a unique solution M still requires other two equations.

Note that M can monitor the wireless channel to measure the RSS transmitting from A_1 and B. M can then easily calculate A_1M and BM based on Equations 1 and 2. We also denote A_1M by coordinates as

$$A_1M = \sqrt{(x_{a1} - x_m)^2 + (y_{a1} - y_m)^2} \tag{23}$$

Because M knows A_1 and (x_m, y_m), this equation becomes a function G of angles x_{a1} and y_{a1} as

$$y_{a1} = G(x_{a1}) \tag{24}$$

Note that Figure 11 is the configuration at time t, so (x_{a1}, y_{a1}), (x_b, y_b), and (x_m, y_m) are fixed at this time.

Combining Equations 22 and 24, the equation becomes $\forall, \alpha \in (0, 2\pi)$

$$\text{Circle } C: \ (x_{a1} - a)^2 + (y_{a1} - b)^2 = (A_1M)^2 \tag{25}$$

where Equation 25 in an x-y coordinate system represents a circle C. In other words, A_1 can be any position on the circle C, and M cannot obtain the valid single location (x_{a1}, y_{a1}).

In summary, the adversary cannot determine the actual position of one device at any time, on the condition of knowing its own position and that of the other device.

It is next important to evaluate the possibility of the adversary constructing a rough understanding of the RSS trajectory. To identify the changes in AB may be possible through a coarse knowledge of how MA and MB change. We perform several simulations with A and B in various configurations relative to M. We calculate the distances AM, BM and AB as A and B change. The results are presented in Table II, where ↗ represents an increase of distance and ↘ indicates a decrease. This table illustrates that an increase in distance AB can be caused by four different reconfigurations of A and B relative to M. Likewise, a decrease in AB can be caused by four different such reconfigurations. In testing, the distribution of these eight categories is uniform; the highest percentage is 13.9%, and the lowest is 11.4%. This indicates that the adversary cannot use any statistical rules to find a relationship between distance changes for AB from AM and BM.

TABLE II
SIMULATION RESULTS

AB	↗			
MA	↗	↗	↘	↘
MB	↗	↘	↘	↗
%	12.9%	13.2%	12.4%	13.5%
AB	↘			
MA	↘	↘	↗	↗
MB	↘	↗	↗	↘
%	11.9%	11.4%	12.5%	13.3%

From the above analysis, we conclude that an adversary cannot know the actual positions of both devices to derive the distance between them. Because the RSS trajectory is a function of this distance, the adversary with Low-level technologies cannot obtain the RSS trajectories between these devices or the corresponding secret key. Smart adversaries that can utilize an AoA (Angle of Arrival) estimation and some other technical means have the ability to get RSS trajectories but still cannot obtain the final key. Because the final key sequence requires to take out the mismatch bits in different locations from the bit sequence generated by RSS trajectories. The mismatch bits caused by environment factors cannot be obtained by adversaries due to the random and time-varying nature of the environment factors.

C. An Example

Here we give an example to illustrate that an adversary cannot create statistical rules for distance changes between the two devices A and B. Figure 12(a), (b), (c) and (d) describe the motion trails of the two devices at successive timestamps. Figure 12(a) denotes the initial state, where L_{a1}, L_{b1} and L_{ad1} are the relative distances between the Attacker and Device A, Attacker and Device B and Devices A and B, respectively. The two devices move randomly as in realistic human shaking. Figure 12(b), (c) and (d) are the movement sketches of the two devices and the green long dash lines in each sub-figure represent the initial state which is used as a comparison. The black dash lines and arrows in each sub-figure indicate the moving trails of both devices.

We define R_a and R_b as the RSS of device A and B, while R_{ta} and R_{tb} are the RSS from A and B measured by the adversary. We use a matrix $\mathbf{R}$ to indicate the four RSS values in Figure 12(a), (b), (c) and (d).

$$\mathbf{R} = \begin{bmatrix} R_{a1} & R_{b1} & R_{ta1} & R_{tb1} \\ R_{a2} & R_{b2} & R_{ta2} & R_{tb2} \\ R_{a3} & R_{b3} & R_{ta3} & R_{tb3} \\ R_{a4} & R_{b4} & R_{ta4} & R_{tb4} \end{bmatrix} \tag{26}$$

where the four columns of the matrix represent the RSS in each sub-figure.

We then use another matrix $\mathbf{D}$ to indicate the change of distance from Figure 12(a) to Figure 12(d),

$$\mathbf{D} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ ↗ & ↗ & ↗ & ↘ \\ ↘ & ↘ & ↗ & ↗ \\ ↘ & ↘ & ↘ & ↗ \end{bmatrix} \tag{27}$$

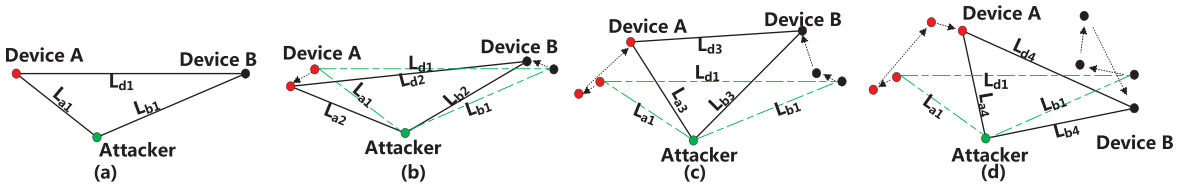


Fig. 12. Example of defending threats.

where as before $\nearrow$ represents an increase, $\searrow$ indicates a decrease, and 0 represents the initial distance, Obviously, $\mathbf{D}_{1,i}$ and $\mathbf{D}_{2,i}, i \in [1,4]$, which indicate the actual distance L_d, have the same changing trends $\nearrow$, $\nearrow$, $\searrow$. However, the distances L_a and L_b do not have same trend as L_d.

Note that this example has only four steps, which means only four samples. In our proposed situation, we know from the Nyquist sampling theorem that in 1 second, the devices will obtain at least $\frac{1}{f_b}$ samples, where f_b is the bandwidth. Then the adversary has a probability of only $2^{-\frac{T}{f_b}}$ to statistically infer the distance change correctly during the whole sampling time T. Because with the increase of T, we have $\lim_{T\to\infty} 2^{-\frac{T}{f_b}} \to 0$,

We conclude that the change of distance between two devices is extremely hard to obtain by random selection, so the adversary cannot realistically obtain the RSS trajectory.

D. 2nd Situation: Defense of Key Sequence Prediction

In the first situation mentioned above, we demonstrate the difficulty for the adversary to obtain all the RSS trajectories between the devices only knowing the location of one device. However, some adversaries may have the ability to detect all of this RSS information through AoA (Angle of Arrival) estimation or some other ways. Note that in our proposed key establishment method, obtaining all the RSS trajectories is not the same as knowing the key sequence, even if the algorithms in error correction are public.

The error correction of our key establishment method utilizes Bloom Filter to find out and correct the mismatched bits. We point out that these mismatched bits are generated by the interference, thermal noise, differences in hardware, etc [29]. These elements are unique to two devices in different locations and the uniqueness provides the security of key establishment process. We consider the situation in which two devices generate the key sequence using our proposed scheme, and an attacker is able to obtain the approximate locations of both devices or the change of relative distance between the two normal devices. This means the attacker could estimate a perfect RSS trajectory of the two devices because the locations or the relative distance are not affected by those environmental factors. We denote the bit sequence after quantization as S_p for the attacker, for a perfect RSS trajectory, while the bit sequences after quantization for the two devices are S_a and S_b, respectively. Suppose S_p, S_a and S_b are all l-bit length sequence, and l_d bits are dropped in both S_a and S_b for the error compensation. The attacker is required to drop the same l_d bits at the same locations of S_p to finally obtain the key with $(l - l_d)$ bits. From Table III in the experimental results we find $130\%(l - l_d) = l$. $l_d = (3/13)l$. The probability P_a for an attacker to find out the l_d bits in the same locations of S_p is:

$$P_a = \frac{1}{l_d!\binom{l}{l_d}} = \frac{1}{l \times (l-1) \times (l-2) \cdots (l - l_d + 1)} \tag{28}$$

where $\binom{}{}$ is binomial coefficient. This probability in which the attacker could obtain the final secret key is an extreme small value. For a 128-bit key, $(l - l_d) = 128$, $l = \lceil 128 \times 1.3 \rceil = 167$, $l_d = 39$, the probability of the attacker to obtain the key sequence after error correction is $P_{a128} = \frac{1}{39!\binom{167}{39}}$.

Due to the analysis of the two situations, the security of our scheme is demonstrated.

VI. Experimental Results

In the evaluation section, we quantify the speed, reliability, and efficiency achieved by our technique. We first introduce the experimental setup and evaluation metrics, and then evaluate the performance within this setup. To make our method more sense, we further do experiments in sensors which are embedded in wearable devices.

A. Experimental Setup

Our prototype system is built on two Universal Software Radio Peripheral (USRP) [13] N210 models, which send signals and capture and store raw channel samples for post processing. To evaluate our key establishment process, we physically move the two devices in random paths while they transmit and receive signals. As previously introduced, we employ a virtual full-duplex mode by broadcasting signals of two very close frequencies, here 2.4 GHz on one device and 2.395 GHz on the other. We have verified that the proximity between 2.4 GHz and 2.395 GHz cannot cause obvious interference, which meets the assumption in Section II.

Because our key establishment method can dynamically generate secret keys with different length to satisfy different security levels, we experiment with a few representative key lengths, including 64, 128, 192 and 256-bit keys. We generate a key 60 times for each key length, in both outdoor and indoor environments, for a total of 480 key generations, and measure these according to the metrics listed next.

To demonstrate the practicability of our technique, we further do experiments using the CC2530 [4] sensor which can be embedded in wearable devices. Two CC2530 sensors are

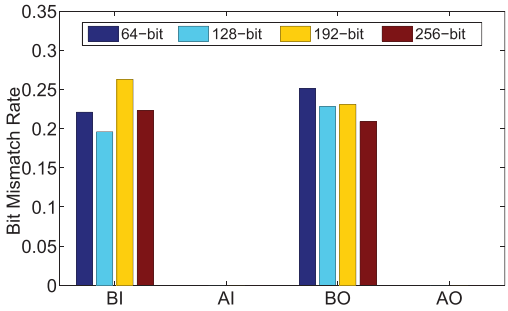


Fig. 13. Bit mismatch rate after quantization and error compensation in both indoor and outdoor environments. Here in the x-axis, BI, BO, AI and AO represent before error compensation indoors, before error compensation outdoors, after error compensation indoors and after error compensation indoors, respectively.

tied to the wrists of each of two people who stand in sight-distance range of each other. As mentioned earlier, we use a virtual full-duplex mode in our technique, so the two chips are tied up as one device (one for transmitting and another for receiving). We consider Los and NLoS situations, respectively.

B. Metrics

The below three metrics are used in the evaluation of any wireless key establishment scheme, including ours.

1) Bit Mismatch Rate: The percentage of bits in disagreement between the initial bit sequences output from the quantization process at the two devices. This informs as to the performance of the quantization and what reconciliation will be necessary.

2) Key Generation Rate: The number of secret keys generated per unit time, presented here as the reciprocal, or the average time required to establish one secret key. This term is crucial to describe the technique's usability, for if a scheme is highly secure and robust but takes minutes to construct a key, it will have very low application as typical users will not stand to wait that long.

3) Efficiency: The ratio between the total number of bits required to quantize and the desired key length. After quantization, we apply error correction to drop any mismatched bits, so the total number of bits required is the desired key length plus any extra error compensation bits needed to make up for those dropped.

C. Bit Mismatch Rate

We must analyze the bit mismatch rate to measure how well the two devices' measured RSS trajectories match up and how well our quantization and error compensation steps perform, as well as to determine what error correction will be necessary to arrive at a final key of sufficient length. Figure 13 shows the bit mismatch rate after the quantization and error correction steps, respectively, for the 60 experiments per key length indoors and outdoors. For the indoor scenario, the bit mismatch rates after quantization for 64, 128, 192, and 256-bit keys are 0.221, 0.196, 0.263 and 0.223, respectively (shown at BI in the figure). For the outdoor scenario, those rates are 0.251, 0.228, 0.231 and 0.209 (shown at BO in the figure).

TABLE III
THE ACTUAL BITS GENERATED WITH DIFFERENT σ VALUES

		Successful experiments/total			
Key length		σ	$110\%\sigma$	$120\%\sigma$	$130\%\sigma$
Indoor	64	33/60	43/60	55/60	60/60
	128	31/60	42/60	53/60	60/60
	192	27/60	46/60	60/60	60/60
	256	32/60	39/60	56/60	60/60
Outdoor	64	30/60	41/60	57/60	60/60
	128	27/60	41/60	60/60	60/60
	192	29/60	38/60	59/60	60/60
	256	33/60	43/60	54/60	60/60

The difference is small between outdoor and indoor cases, indicating that roughly similar performance may be expected wherever two humans and their devices interact.

While the values of AI and AO in Figure 13 reflect the bit mismatch rates after error compensation, which directly drop down to 0, in indoors and outdoors, respectively. The reason is that we utilize Bloom Filter and KL-T to guarantee the consistency of the two bit sequence. From Table III, it is observed that almost 30% of the bits after quantization are dropped during error correction.

1) Error Compensation: The necessity of error compensation is mentioned in Section IV-A.2, to allow for the dropped bits in information reconciliation.

Supposing N bits are required as a secret key, if we specify interval size by the equations introduced in Section IV-A.2, $\frac{T}{N}$ or $\frac{S}{N}$ without a weight σ, N bits are extracted by quantization. However, after error correction, we will drop some N_r mismatched bits due to the bit mismatch rate r. Thus, the final key sequence length is $N - N_r$ bits, which does not meet the requirement of N bits. The simplest method to make up this bit loss is to generate more bits in the quantization step. We define the error compensation weight σ dependent on bit mismatch rate r as $\sigma = 1 - r$.

The length of final key sequence becomes

$$\begin{cases} \dfrac{S}{l}(1-r) = \dfrac{S}{\frac{S\sigma}{N}}(1-r) = N & \text{(Dynamic interval) or} \\ \dfrac{T}{l}(1-r) = \dfrac{T}{\frac{T\sigma}{N}}(1-r) = N & \text{(Fixed interval),} \end{cases} \tag{29}$$

Here it is evident that the required N bit secret key is generated with the help of the weight σ.

However, this assumes a static bit mismatch rate of exactly r, which may vary case to case in actual key generation instances. If we use the average error compensation weight σ, some experiments with more mismatched bits will not have extra bits enough to generate a sufficiently long key (while others will have more than enough). Consequently, we perform the 480 experiments again with different values of σ to find what weight is necessary to ensure a very high probability of sufficient bit length. So in Table III, we use for different σ to see the number of experiments which does not meet the required bits number.

From Table III, we find that when the error compensation is the average σ, almost half of the experiments cannot generate the required key length, which is not surprising. With an

TABLE IV
AVERAGE KEY GENERATION TIME

Length(bits)	64	128	192	256
Times in indoor(s)	0.22	0.61	0.95	1.28
Times in outdoor(s)	0.33	0.74	1.02	1.46

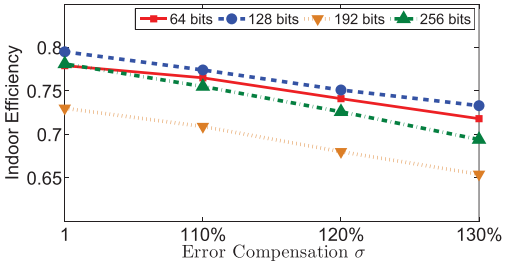


Fig. 14. Efficiency in indoor environment.

increase to 120% of the average weight, all 60 experiments generating 192-bit keys inside and 128-bit keys outside generate enough bits, but the other options do not. When the error compensation weight is set to 130% of the average σ, however, all the experiments meet their demands.

D. Key Generation Rate

Next, we analyze our 480 key generations to find the typical time requirements. Table IV contains the average time results of generating a key 60 times for each key length, inside and outside. We observe that the time spent in indoor environment is a little less than that in outdoor environment, as there is more inference outside caused by random noise and possibly other wireless devices using the same frequency. Regardless, the device interaction time is well within 1 or 2 seconds, for all bit lengths tested, allaying any concerns about usability.

E. Efficiency

As previously mentioned, the bit mismatch rate fluctuates due to environmental factors, so we cannot predict the exact bit mismatch rate before we do key establishment. We found before that a σ that is 130% of the average can provide enough excess bits to safely achieve the required key length in all of our experiments, but in practice we cannot obtain the actual exact value of σ necessary for a key establishment that has not taken place. We denote the efficiency ρ of key establishment as the ratio between the actual bits obtained with the approximated σ and the required key length, as $\rho = \frac{N'}{N}$ where N' is the actual number of bits generated in a real key establishment, N is the required key length. Figure 14 shows the efficiency indoors and Figure 15 outdoors. Inside, the 128 bit keys have the highest average efficiency, while that for the 192 bit keys is the lowest, ranging from 0.654 to 0.730. Outside, all key lengths have a similar range of efficiency from 0.754 to 0.664. Comparing Figure 14 to 15, the efficiencies in the former are higher than the those in the latter, except for the 192 bits key. Since the frequency used is 2.4 Ghz, which is an open access frequency band, the experiment results may have a slight negative impact from other wireless signals occupying this frequency.

In Figures 14 and 15, also note that all efficiencies decrease with larger error compensation weight σ, which ranges as before from the average σ to 130% of the average. This is

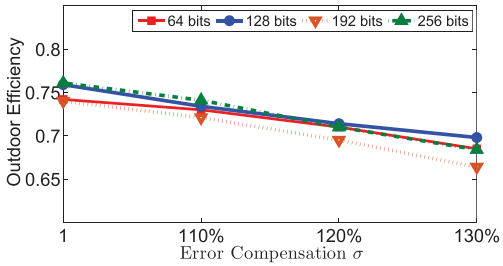


Fig. 15. Efficiency in outdoor environment.

TABLE V
ENTROPY

key length		P_0	P_1	Entropy
Outdoor	64	0.597	0.403	0.972
	128	0.409	0.591	0.976
	192	0.414	0.586	0.978
	256	0.604	0.396	0.968
Indoor	64	0.467	0.533	0.996
	128	0.561	0.439	0.989
	192	0.572	0.428	0.985
	256	0.443	0.557	0.993

due to some excess compensation bits which are processed but unused. For example, one of the experiments required 70 bits to compensate for mismatched bits. 120%σ results in 69 added bits, which is insufficient, but 130%σ provides 75. Then $75 - 70 = 5$ bits are wasted.

F. Entropy

As mentioned earlier, the entropy is only related to the time of acceleration increase and decrease. Here we measure the increase and decrease time for each of our bit lengths (64, 128, 192 and 256), and calculate the average probability of binary 1 and 0. The results are presented in Table V, where P_1 is the probability of a binary 1 generated from an acceleration increase (shown in Figure 9(a) and (c)), and P_0 is the probability of a binary 0 generated from an acceleration decrease (shown in Figure 9(b) and (d)). Visible in this table, the P_0 and P_1 vary greatly for and do not correlate with different key lengths, while the entropy is fairly constant near 1 for all experiments. The lowest entropy is 0.968 for outdoor 256-bit key generation, while the highest is 0.993 for indoor, 256-bit generation. Although entropies in indoor environment are slightly better than those in outdoor, the performance of entropy is acceptable everywhere.

G. Randomness of the Final Key

It is important to test whether the generated keys are random or not, because that randomness is helpful in determining the suitability of a generator for the cryptographic application. We utilize the NIST randomness test suite for random number generators [34] to evaluate the ï'nal established keys. It is difficult to include the details of all NIST tests, but the NIST statistical test gives the p-values of different random test processes, and the p-values indicate the probability that the key sequence is generated by a random process. If the p-value is less than 1%, the randomness hypothesis is rejected, which means the key is not random. Table I shows the test results of p-Values which are all greater than 1%. The results indicate the generated key passes the randomness test. Table VI shows

TABLE VI

P-VALUE OF NIST RANDOMNESS TESTING

NIST Test	p-Value
Frequency	0.627458
FFT Test	0.018753
Longest Run	0.095233
Linear Complexity	0.808840
Block Frequency	0.914620
Cumulative Sums	0.409815
Approximate Entropy	0.942183
Non-Overlapping Template	0.997572

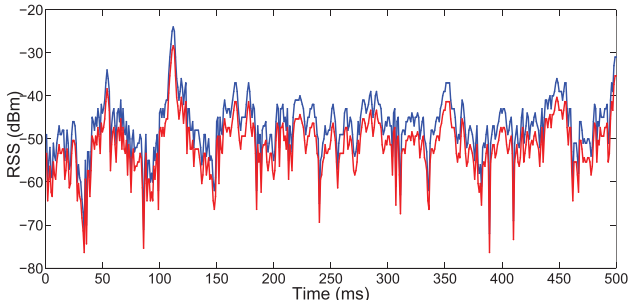


Fig. 16. The transmitting and receiving RSSs in LoS.

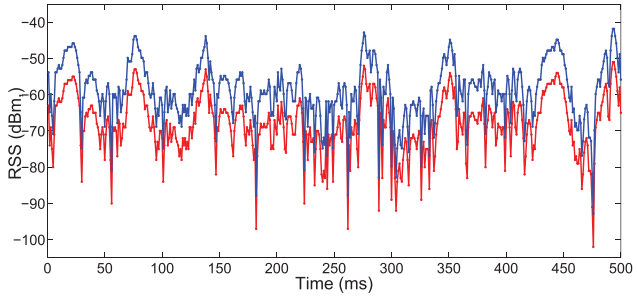


Fig. 17. The transmitting and receiving RSSs in NLoS.

the test results of p-Values which are all greater than 1%. The results indicate the generated key pass the randomness test.

H. Experiments With Sensors

We also experiment with those sensors able to be used in wearable devices. Figure 16 shows the RSSs received by both devices in the LoS scenario. Figure 17 shows those received in an NLoS scenario, where a desk is between the two devices. For both scenarios, the RSSs of both devices have approximately the same slope even though the RSSs in the NLoS scenario are weaker. As mentioned in Section IV-A, our scheme does not consider the absolute value of RSSs but the trajectories of RSS, and thus the RSSs in both figures can provide feasible data for our key establishment scheme.

Figure 18 shows the bit mismatch rate for the 60 experiments per key length indoors and outdoors. For the indoor scenario, the bit mismatch rate before quantization using USRPs devices and sensors for generating 128-bit keys are 0.196 and 0.224, respectively (shown at BI in the figure). For the outdoor scenario, those rates are 0.228 and 0.254 (shown at BO in the figure), respectively. We find that the results of USRPs is a little better than those of sensors because of the more powerful and precise capabilities. And the difference between outdoor and indoor cases is small, which indicates that our method could work in both indoor and outdoor environments.

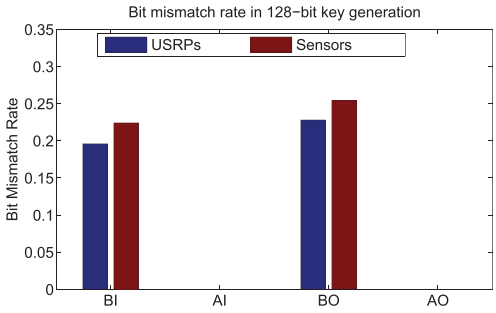


Fig. 18. Bit mismatch rate in 128-bit key generation between USRPs and sensors. Here in the x-axis, BI, BO, AI and AO represent before error compensation indoors, before error compensation outdoors, after error compensation indoors and after error compensation indoors, respectively.

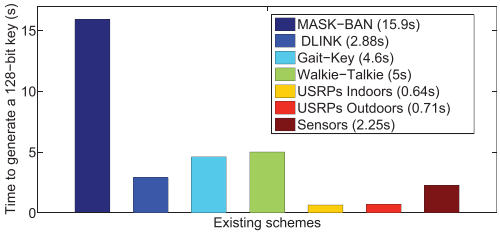


Fig. 19. Comparison with existing schemes.

While the values of AI and AO in Figure 18 reflect the bit mismatch rates using different equipments after error compensation, which directly drop down to 0, in indoors and outdoors, respectively. The reason has been mentioned in Section VI-D.

Figure 19 presents the time to generate a 128-bit key between some newly existing schemes focusing on the key generation in wearable devices, including [32], [35], [46] and [33], and our proposed scheme. These existing schemes require 15.9s, 2.88s, 2.56s, 4.6s, and 5s to generate a 128-bit key, respectively. Our scheme functions indoors and outdoors with USRPs in 0.64s and 0.71s, respectively. With commonly available sensors, generating a 128-bit key needs 2.25s in an indoor environment because the computing capability of sensor chips equipped with wearable devices are much weaker than USRPs.

VII. RELATED WORK

Received signal strength (RSS) or received signal power is the most commonly used feature for secret key establishment. Because it is easy to measure and with currently widely accessible equipment. However, RSS may vary at different receivers, so the key generation rate of RSS based methods is low. Previous works mainly focused on exploiting temporal and spatial variations of wireless channel [9], [18], [24], [28], multiple antenna diversity [49], and multiple frequencies [43] for secret bit generation between a pair of wireless devices. The recent work [23] proposes to defend against threats of eavesdropping and fake data injection in underwater acoustic networks (UANs), providing an overview of the advantages of RSS-based key generation and explore the major challenges from the unique features of acoustic communications in underwater systems. through experiment results of sea trails. The authors also discuss viable solutions to improve the performance of RSS-based key generation in oceans.

RSS based secret key generation mechanisms for wearable devices or so-called Body Area Networks (BAN) [15], [27] are derived from the scheme mentioned above. Ali *et al.* [5] uses filtering to reduce the discrepancy between the channel samples at two ends and have proposed a scheme to extract approximately matching keys without using reconciliation methods. However, efficiency of the scheme is platform specific and bit rate is too low (0.14 bps). Latest work [32] proposes DLINK which utilizes lightweight dual antennas to dynamically identifies the channel links with sufficient fluctuation for secret key generation. DLINK generates a 128 bit key in 2.88 s, which is nearly 5 times faster compared to the a most recent scheme [48].

Key establishment using physical layer characteristics, such as channel impulse response (CIR) and channel phase, are first proposed in [16]. Compared to RSS, physical layer characteristics are much richer source of secret information but higher computational overhead. The paper [44] proposes the reciprocity theorem, which has become the most important theorem in this kind of technique. Zeng *et al.* [42] review different types of existing techniques based on quantization, handling communication errors and the feasibility and security issues related to these techniques. This paper also summarizes emerging topics on channel reciprocity based key establishment. A recent paper [17] provides an efficient secret key generation mechanism using multipath relative delay from Ultra-wideband (UWB) channels. The authors study a statistical characterization of UWB channels in a residential scenario, and evaluate key-mismatch probability in NLOS and LOS UWB channels. Another latest work [45] uses channel state information (CSI) as the common secret among legitimate to design and implement a key agreement protocol for mobile devices.

Practical methods using physical layer characteristics to generate a secret key for wearable devices and BAN can be grouped into two categories. The first category is using the channel as the encryption/decryption function - see [37], [39], [47] for example. The second category is using the wireless channel for extracting a symmetric encryption key to be used in a symmetric encryption algorithm - see [35], [48] for examples. It is recognized that the second category offers low implementation complexity when based on the reciprocal quantization of an estimated parameter from the channel.

VIII. Conclusion

In this paper, we describe a novel key establishment method that uses the distance variation trends caused by the motion paths of the two devices relative to each other. In order to prove the similarity of distance variation trends measured by RSS at both two devices, we analyze the gathered RSS data at each and calculate the time derivatives of the data for comparison. Based on their verified sameness, we construct a key extraction scheme to generate a bit sequence to use as a secret key. We also demonstrate that attackers monitoring the wireless channel cannot compromise the key establishment process. We validate the performance of the key establishment method through 60 experiments for each of four bit lengths, both indoors and outside, finding the method both efficient and robust.

References

[1] (2016). *Androidpay: Pay With Your Phone in Stores*. [Online]. Available: https://www.android.com/pay/

[2] (2016). *Applepay: Cashless Made Effortless*. [Online]. Available: http://www.apple.com/apple-pay/

[3] Intel Corporation. (2016). *Literature: Max Devices*. [Online]. Available: http://www.intel.com/

[4] Texas Instrument Corporation. (2016). *cc2530, The Second Generation System-Onchip Solution for 2.4 GHz IEEE 802.15.4/RF4CE/ZigBee*. [Online]. Available: http://www.TI.com/

[5] S. T. Ali, V. Sivaraman, and D. Ostry, "Eliminating reconciliation cost in secret key generation for body-worn health monitoring devices," *IEEE Trans. Mobile Comput.*, vol. 13, no. 12, pp. 2763–2776, Dec. 2014.

[6] B. Azimi-Sadjadi, A. Kiayias, A. Mercado, and B. Yener, "Robust key generation from signal envelopes in wireless networks," in *Proc. 14th ACM Conf. Comput. Commun. Secur.*, 2007, pp. 401–410.

[7] B. H. Bloom, "Space/time trade-offs in hash coding with allowable errors," *Commun. ACM*, vol. 13, no. 7, pp. 422–426, Jul. 1970.

[8] J. Buddenhagen, C. Ford, and M. May, "Nice cubic polynomials, pythagorean triples, and the law of cosines," *Math. Mag.*, vol. 65, no. 4, pp. 244–249, 1992.

[9] J. Croft, N. Patwari, and S. K. Kasera, "Robust uncorrelated bit extraction methodologies for wireless sensors," in *Proc. 9th ACM/IEEE Int. Conf. Inf. Process. Sensor Netw.*, Apr. 2010, pp. 70–81.

[10] W. Diffie and M. E. Hellman, "New directions in cryptography," *IEEE Trans. Inf. Theory*, vol. IT-22, no. 6, pp. 644–654, Nov. 1976.

[11] R. Dony, "Karhunen–Loeve transform," in *The Transform and Data Compression Handbook*. Boca Raton, FL, USA: CRC Press, 2001.

[12] J. Dricot, G. Ferrari, S. Roy, F. Horlin, and P. Doncker, "Outage, local throughput, and achievable transmission rate of narrowband body area networks," in *Proc. COST 2100*, Vienna, Austria, 2009.

[13] Ettus. (2017). *Universal Software Radio Peripheral*. [Online]. Available: https://www.ettus.com/

[14] A. Goldsmith, *Wireless Communications*. Cambridge, U.K.: Cambridge Univ. Press, 2005.

[15] S. Heaney, W. Scanlon, and E. Garcia-Palacios, "Effect of environmental multipath on line of sight body to body communication at 2.45 GHz," in *Proc. Loughborough Antennas Propag. Conf.*, Nov. 2012, pp. 1–4.

[16] J. E. Hershey, A. A. Hassan, and R. Yarlagadda, "Unconventional cryptographic keying variable management," *IEEE Trans. Commun.*, vol. 43, no. 1, pp. 3–6, Jan. 1995.

[17] J. Huang and T. Jiang, "Secret key generation exploiting Ultra-wideband indoor wireless channel characteristics," *Secur. Commun. Netw.*, vol. 8, no. 13, pp. 2329–2337, 2015.

[18] S. Jana, S. N. Premnath, M. Clark, S. K. Kasera, N. Patwari, and S. Krishnamurthy, "On the effectiveness of secret key extraction from wireless signal strength in real environments," in *Proc. Int. Conf. Mobile Comput. Netw. (MOBICOM)*, Beijing, China, Sep. 2009, pp. 321–332.

[19] O. Jo, S. Chang, C. Kweon, J. Oh, and K. Cheun, "60 GHz wireless communication for future Wi-Fi," *ICT Exp.*, vol. 1, no. 1, pp. 30–33, 2015.

[20] S. Kodera, Y. Naruse, Y. Kawahara, T. Watanabe, and S. Saruwatari, "Poster: A medium access control protocol for full-duplex wireless information and power transfer," in *Proc. ACM Conf. Embedded Netw. Sensor Syst.*, 2015, pp. 399–400.

[21] C. M. Lee *et al.*, "An introduction to the NASA Hyperspectral InfraRed Imager (HyspIRI) mission and preparatory activities," *Remote Sens. Environ.*, vol. 167, pp. 6–19, Sep. 2015.

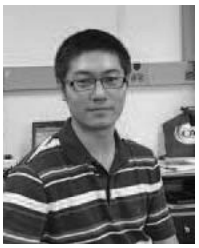
[22] H. Liu, J. Yang, Y. Wang, and Y. Chen, "Collaborative secret key extraction leveraging received signal strength in mobile wireless networks," in *Proc. INFOCOM*, Mar. 2012, pp. 927–935.

[23] Y. Luo, L. Pu, Z. Peng, and Z. Shi, "RSS-based secret key generation in underwater acoustic networks: Advantages, challenges, and performance improvements," *IEEE Commun. Mag.*, vol. 54, no. 2, pp. 32–38, Feb. 2016.

[24] S. Mathur, W. Trappe, N. Mandayam, C. Ye, and A. Reznik, "Radio-telepathy: Extracting a secret key from an unauthenticated wireless channel," in *Proc. 14th ACM Int. Conf. Mobile Comput. Netw.*, 2008, pp. 128–139.

[25] A. Broder and M. Mitzenmacher, "Network applications of bloom filters: A survey," *Internet Math.*, vol. 1, no. 4, pp. 485–509, 2004.

[26] A. Molisch, *Wireless Communications*. Hoboken, NJ, USA: Wiley, 2007.
[27] D. Neirynck, C. Williams, A. Nix, and M. Beach, "Personal area networks with line-of-sight MIMO O-ration," in *Proc. IEEE 63rd Veh. Technol. Conf. (VTC-Spring)*, May 2006, pp. 2859–2862.
[28] N. Patwari, J. Croft, S. Jana, and S. Kasera, "High-rate uncorrelated bit extraction for shared secret key generation from channel measurements," *IEEE Trans. Mobile Comput.*, vol. 9, no. 1, pp. 17–30, Jan. 2013.
[29] S. N. Premnath *et al.*, "Secret key extraction from wireless signal strength in real environments," *IEEE Trans. Mobile Comput.*, vol. 12, no. 5, pp. 917–930, May 2013.
[30] M. Rapin, J. Wacker, and O. Chetelat, "Two-wire bus combining full duplex body-sensor network and multilead biopotential measurements," *IEEE Trans. Biomed. Eng.*, to be published.
[31] G. Revadigar, C. Javali, H. J. Asghar, K. B. Rasmussen, and S. Jha, "Mobility independent secret key generation for wearable healthcare devices," in *Proc. EAI Int. Conf. Body Area Netw.*, 2015, pp. 294–300.
[32] G. Revadigar, C. Javali, W. Hu, and S. Jha, "DLINK: Dual link based radio frequency fingerprinting for wearable devices," in *Proc. IEEE 40th Conf. Local Comput. Netw.*, Oct. 2015, pp. 329–337.
[33] G. Revadigar, C. Javali, W. Xu, and W. Hu, "Secure key generation and distribution protocol for wearable devices," in *Proc. IEEE Int. Conf. Pervasive Comput. Commun. Workshops*, Mar. 2016, pp. 1–4.
[34] A. L. Rukhin *et al.*, *SP 800-22 Rev. 1A. A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications*. Gaithersburg, MD, USA: NIST Special Publication, 2010.
[35] L. Shi, J. Yuan, S. Yu, and M. Li, "MASK-BAN: Movement-aided authenticated secret key extraction utilizing channel characteristics in body area networks," *IEEE Internet Things J.*, vol. 2, no. 1, pp. 52–62, Feb. 2015.
[36] Y. Sun, W. Wang, B. Da, L. Yang, and Y. Haihua, "Wireless network area limiting method and system based on near field communication," U.S. Patent 0219262 A1, Jan. 30, 2014.
[37] G. Tsouri and D. Wulich, "Securing OFDM over wireless time-varying channels using subcarrier overloading with joint signal constellations," *EURASIP J. Wireless Commun. Netw.*, vol. 2009, no. 1, p. 437824, Dec. 2009.
[38] T. Vermeulen and S. Pollin, "Energy-delay analysis of full duplex wireless communication for sensor networks," in *Proc. Global Commun. Conf.*, 2014, pp. 455–460.
[39] J. W. Wallace and R. K. Sharma, "Automatic secret keys from reciprocal MIMO wireless channels: Measurement and analysis," *IEEE Trans. Inf. Forensics Security*, vol. 5, no. 3, pp. 381–392, Sep. 2010.
[40] B. Wang, M. Li, and H. Wang, "Geometric range search on encrypted spatial data," *IEEE Trans. Inf. Forensics Security*, vol. 11, no. 4, pp. 704–719, Apr. 2016.
[41] Q. Wang *et al.*, "Walls have ears! Opportunistically communicating secret messages over the wiretap channel: From theory to practice," in *Proc. 22nd ACM SIGSAC Conf. Comput. Commun. Secur.*, 2015, pp. 376–387.
[42] T. Wang, Y. Liu, and A. V. Vasilakos, "Survey on channel reciprocity based key establishment techniques for wireless systems," *Wireless Netw.*, vol. 21, no. 6, pp. 1835–1846, Mar. 2015.
[43] M. Wilhelm, I. Martinovic, and J. Schmitt, "Secret keys from entangled sensor motes: Implementation and analysis," in *Proc. 3rd ACM Conf. Wireless Netw. Secur.*, 2010, pp. 139–144.
[44] R. Wilson, D. Tse, and R. A. Scholtz, "Channel identification: Secret sharing using reciprocity in ultrawideband channels," *IEEE Trans. Inf. Forensics Security*, vol. 2, no. 3, pp. 364–375, Sep. 2007.
[45] W. Xi *et al.*, "Instant and robust authentication and key agreement among mobile devices," in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur.*, 2016, pp. 616–627.
[46] W. Xu, G. Revadigar, C. Luo, and N. Bergmann, "Walkie-talkie: Motion-assisted automatic key generation for secure on-body device communication," in *Proc. ACM/IEEE Int. Conf. Inf. Process. Sensor Netw.*, Apr. 2016, pp. 1–12.
[47] C. Ye, S. Mathur, A. Reznik, Y. Shah, W. Trappe, and N. B. Mandayam, "Information-theoretically secret key generation for fading wireless channels," *IEEE Trans. Inf. Forensics Security*, vol. 5, no. 2, pp. 240–254, Jun. 2010.
[48] J. Yuan, L. Shi, S. Yu, and M. Li, "Authenticated secret key extraction using channel characteristics for body area networks," in *Proc. ACM Conf. Comput. Commun. Secur.*, 2012, pp. 1028–1030.
[49] K. Zeng, D. Wu, A. Chan, and P. Mohapatra, "Exploiting multiple-antenna diversity for shared secret key generation in wireless networks," in *Proc. IEEE INFOCOM*, Mar. 2010, pp. 1–9.



Zi Li received the B.S. degree in communication engineering from the School of Telecommunication, Xidian University, in 2011, where he is currently pursuing the Ph.D. degree in information security. His research interests focus on wireless networks and physical layer security.

Qingqi Pei (SM'15) received the B.S., M.S., and Ph.D. degrees in computer science and cryptography from Xidian University, in 1998, 2005, and 2008, respectively. He is currently a Professor and a member of the State Key Laboratory of Integrated Services Networks. He is also a Professional Member of the ACM and a Senior Member of the Chinese Institute of Electronics and China Computer Federation. His research interests focus on cognitive network, data security, and physical layer security.

Ian Markwood received the B.S. degree in mathematics from Hillsdale College, Hillsdale, MI, USA, in 2011, and the M.S. degree in computer science from the University of South Florida, Tampa, FL, USA, where he is currently pursuing the Ph.D. degree in computer science. He is focusing on cyber-physical systems security.

Yao Liu received the Ph.D. degree in computer science from North Carolina State University in 2012. She is currently an Assistant Professor with the Department of Computer Science and Engineering, University of South Florida, Tampa, FL, USA. Her research is related to computer and network security, with an emphasis on designing and implementing defense approaches that protect emerging wireless technologies from being undermined by adversaries. Her research interests also include the security of cyber-physical systems, especially in smart grid security. She was a recipient of the Best Paper Award at the Seventh IEEE International Conference on Mobile Ad-Hoc and Sensor Systems.

Haojin Zhu (M'09–SM'16) received the B.Sc. degree in computer science from Wuhan University, China, in 2002, the M.Sc. degree in computer science from Shanghai Jiao Tong University, China, in 2005, and the Ph.D. degree in electrical and computer engineering from the University of Waterloo, Waterloo, ON, Canada, in 2009. Since 2017, he has been a Full Professor with the Computer Science Department, Shanghai Jiao Tong University. He published 35 international journal papers, including JSAC, TDSC, TPDS, TMC, TWC, and TVT, and 60 international conference papers, including ACM CCS, ACM MOBICOM, ACM MOBIHOC, IEEE INFOCOM, and IEEE ICDCS. His current research interests include network security and privacy enhancing technologies. He received a number of awards including: the SMC Young Research Award of Shanghai Jiao Tong University in 2011, the IEEE ComSoc Asia-Pacific Outstanding Young Researcher Award in 2014, the Top 100 Most Cited Chinese Papers Published in International Journals in 2014, the Supervisor of Shanghai Excellent Master Thesis Award in 2014, the Outstanding Youth Post Expert Award for Shanghai Jiao Tong University in 2014, the Distinguished Member of the IEEE INFOCOM Technical Program Committee in 2015, and the Young Scholar Award of Changjiang Scholar Program by Ministry of Education, China, in 2016. He was a co-recipient of the Best Paper Awards of IEEE ICC in 2007, Chinacom in 2008, and IEEE GLOBECOM Best Paper Nomination in 2014.