# Shared Secret Key Generation by Exploiting Inaudible Acoustic Channels

YOUJING LU and FAN WU, Shanghai Jiao Tong University
QIANYI HUANG, Southern University of Science and Technology & Peng Cheng Laboratory
SHAOJIE TANG, University of Texas at Dallas
LINGHE KONG and GUIHAI CHEN, Shanghai Jiao Tong University

To build a secure wireless networking system, it is essential that the cryptographic key is known only to the two (or more) communicating parties. Existing key extraction schemes put the devices into physical proximity and utilize the common inherent randomness between the devices to agree on a secret key, but they often rely on specialized hardware (e.g., the specific wireless NIC model) and have low bit rates. In this article, we seek a key extraction approach that only leverages off-the-shelf mobile devices, while achieving significantly higher key generation efficiency. The core idea of our approach is to exploit the fast varying inaudible acoustic channel as the common random source for key generation and wireless parallel communication for exchanging reconciliation information to improve the key generation rate. We have carefully studied and validated the feasibility of our approach through both theoretical analysis and a variety of measurements. We implement our approach on different mobile devices and conduct extensive experiments in different real scenarios. The experiment results show that our approach achieves high efficiency and satisfactory robustness. Compared with state-of-the-art methods, our approach improves the key generation rate by 38.46% and reduces the bit mismatch ratio by 42.34%.

CCS Concepts: • **Security and privacy → Mobile and wireless security**; • **Networks → Mobile and wireless security**; *Security protocols;*

Additional Key Words and Phrases: Key extraction, inaudible acoustic signal, channel estimation



This is the extended version of the article "FREE: A Fast and Robust Key Extraction Mechanism via Inaudible Acoustic Signal" in the Proc. of MobiHoc 2019.
This work was supported in part by National Key R&D Program of China No. 2019YFB2102200, in part by China NSF Grants No. 62002150, No. 62025204, No. 62072303, No. 61972252, and No. 61972254; in part by the Key-Area Research and Development Program of Guangdong Province No. 2020B0101390001 and No. 2020B010164001; in part by Alibaba Group through Alibaba Innovation Research Program; and in part by Tencent Rhino Bird Key Research Project. The opinions, findings, conclusions, and recommendations expressed in this paper are those of the authors and do not necessarily reflect the views of the funding agencies or the government.
Authors' addresses: Y. Lu, F. Wu (corresponding author), L. Kong, and G. Chen, Shanghai Jiao Tong University, 800 Dongchuan Road, Shanghai, P.R. China, 200240; emails: luyoujing@sjtu.edu.cn, fwu@cs.sjtu.edu.cn, linghe.kong@sjtu.edu.cn, gchen@cs.sjtu.edu.cn; Q. Huang, Southern University of Science and Technology & Peng Cheng Laboratory; email: huangqy@sustech.edu.cn; S. Tang, University of Texas at Dallas, Richardson, the United States; email: shaojie.tang@utdallas.edu.

1550-4859/2021/09-ART13 $15.00
https://doi.org/10.1145/3480461

**ACM Reference format:**
Youjing Lu, Fan Wu, Qianyi Huang, Shaojie Tang, Linghe Kong, and Guihai Chen. 2021. Shared Secret Key Generation by Exploiting Inaudible Acoustic Channels. *ACM Trans. Sen. Netw.* 18, 1, Article 13 (September 2021), 26 pages.
https://doi.org/10.1145/3480461

## 1 INTRODUCTION

Nowadays, as the mobile devices have our privacy data, e.g., banking information and sensitive healthcare data, communication security becomes increasingly important. Different from wired network, wireless communication is usually ad hoc, where there are no preexisting infrastructures that support centralized cryptographic key generation and distribution. Classic key generation algorithms, e.g., Diffie-Hellman key agreement protocol, rely upon the computational hardness of assumption, whose vulnerabilities have been identified in Reference [49]. Therefore, this poses a big challenge when it comes to generating and sharing cryptographic keys among mobile devices in a secure manner.

To address this issue, the most common approach for generating the cryptographic key is to use the common inherent randomness between the entities. These efforts can be classified into two categories. The first category of approaches exploits the reciprocity of wireless channel and extracts the channel information to generate secret keys. For example, radio-telepathy extracts secret keys from unauthenticated wireless channels [34]. Wang et al. exploited channel phase randomness to generate secret keys [59]. The second category of approaches utilizes the randomness of environmental sensing to extract secret keys. Their main idea is to put the devices into physical proximity to get similar sensing data to generate a common secret key. For example, Bichler et al. exploited acceleration data of shaking process to generate secret keys [5]. MAGIK utilizes the dynamic geomagnetic field sensing data to extract secret keys [47]. However, existing efforts often rely on specialized hardware or have low key generation rates. For example, the average key generation rate of radio-telepathy is 1.17 secret bits per second in mobile cases [34]. An enhanced approach has been proposed in References [45] with a key generation rate of ∼10–20 bits per second. Although some approaches can achieve higher key generation efficiency, they often require specialized hardware, e.g., Intel 5300 **Network Interface Card (NIC)** [16] or Atheros AR 9380 NIC, and a laptop that is compatible with above NICs [63, 64]. Therefore, there is a lack of practical key extraction methods that have high key generation efficiency and can be implemented on off-the-shelf mobile devices.

In this article, we seek a key generation approach that is compatible with off-the-shelf mobile devices and can achieve significantly higher key generation efficiency. The core idea of our approach is to leverage the randomness of acoustic channel to extract secret keys. Our study is motivated by the following observations from field tests. First, off-the-shelf mobile devices are usually equipped with microphones and speakers, which can be used to transmit and receive acoustic signals. Second, users can shake their mobile devices to vary the acoustic channels and bring in more randomness, thus improving the key generation rate. Third, the key generation rate can be further improved by enabling parallel communication, i.e., users can extract random sequences from acoustic channels while communicate in wireless channel for reconciliation. Besides, we choose inaudible frequency bands for not disturbing others.

However, it is highly non-trivial to realize this idea. In particular, we are facing three challenges: The first challenge is to identify an effective way for mobile users to generate similar random streams from acoustic channels, the second one is how to quantize the random sequences into bit

streams, and the last one is how to reconciliate a common secret key from two similar bit streams in a secure manner.

To address these challenges, we propose **Fast and Robust key Extraction mEchanism (FREE)**. We first study the feasibility of utilizing the randomness in acoustic channels for key extraction from both theoretical analysis and experiments. Fortunately, the acoustic channel is proved to have the significant properties of temporal variation, channel reciprocity, and spatial decorrelation, and thus it is a great medium to establish secret keys. Then, we transmit inaudible acoustic signal to estimate the acoustic channel among devices and get the channel response. To quantize the acoustic channel response, we use an adaptive secret bit generation method to quantize a channel tap into a signal bit or multiple bits. To generate an identical bit stream, we design a protocol for the two entities to reconciliate the mismatched bits.

To evaluate the performance of FREE, we build the FREE prototype on different pairs of mobile devices, and conduct extensive experiments in different scenarios. The experiment results validate the effectiveness and efficiency of FREE.

We now summarize the main contributions of this article.

- We consider using the randomness of inaudible acoustic channel for key extraction and demonstrate that it is an appropriate medium to establish secret key. Our approach utilizes the temporal variation, channel reciprocity, and spatial decorrelation properties of acoustic channel to defend against eavesdropping, approaching, and repeating imitation attacks.
- We implement FREE on the commodity mobile devices. Experiment results show that the key generation rate is significantly higher than existing solutions [22, 30, 34, 47, 63, 64]. Compared with state-of-the-art methods [47, 64], our approach improves the key generation rate by 38.46% and reduces the bit mismatch ratio by 42.34%.

The rest of this article is organized as follows. Section 2 presents the system model and attack model. Section 3 studies the feasibility of using the randomness of acoustic channel for key extraction. Section 4 presents the design details of FREE. Section 5 analyzes the security of FREE. Section 6 evaluates the performance of FREE in real-word experiments. Section 7 discusses other possible attacks. Section 8 reviews the related work, and Section 9 concludes this article.

## 2 SYSTEM OVERVIEW

In this section, we present the overview of our system model and attack model.

### 2.1 System Model

We illustrate our system model in Figure 1. There are two legitimate mobile users, Alice and Bob, who are located in physical proximity. To prevent the passive adversary, Eve, from eavesdropping their communication, they need a common secret key to establish a secure and authentic channel between them. Alice and Bob are both equipped with an off-the-shelf mobile device, such as a consumer-grade mobile phone. They extract the randomness of acoustic channel from the **public inaudible acoustic channel (PIAC)** between them and use **public wireless channel (PWC)** to exchange some messages to reconciliate a common secret key from the random channel response to establish a secure wireless channel, as shown in Figure 1.

Our goal is to utilize the randomness of inaudible acoustic channel to extract secret keys, while achieving a higher key generation rate and lower bit mismatch ratio.

### 2.2 Attack Model

Next, we introduce potential attacks from a passive adversary, Eve. Eve can overhear all signals transmitted through the public inaudible acoustic channel and the public wireless channel. Eve

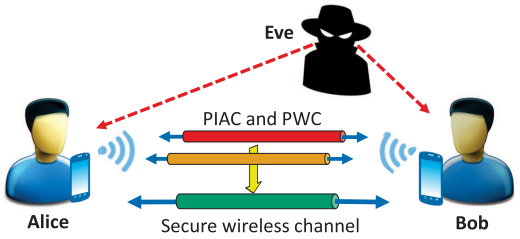



Fig. 1. System model. Alice and Bob estimate the public inaudible acoustic channel (PIAC) to get the randomness of the acoustic channel. They exchange reconciliation information through public wireless channel (PWC), and extract an identical cryptographic key from the randomness to establish a secure wireless channel for guaranteeing the security of the following transmitted messages. A passive adversary, Eve, can eavesdrop all acoustic and wireless signals transmitted through PIAC and PWC but cannot extract the same cryptographic key to decrypt the transmitted messages.

also can estimate her own acoustic channel and get the channel response. She knows the key extraction algorithm and the parameter settings. We assume that Eve is not too close to either Alice or Bob, i.e., Eve is at least 5 cm from Alice and Bob. We also assume that Eve's goal is to intercept the cryptographic key instead of jamming their communications. If Eve jams the communication between Alice and Bob (e.g., transmitting high-power acoustic signals), then it is blocking the key extraction between Alice and Bob. When the acoustic channel communication is blocked, neither Alice nor Bob can extract the secret key. Thus, the attacker cannot get the key neither.

In particular, we mainly consider the following three kinds of attacks.

- Eavesdropping attack: Eve can eavesdrop all signals transmitted through the public inaudible acoustic channel and the public wireless channel between Alice and Bob, and she knows everything about the key extraction algorithm. Thus, she can analyze the captured acoustic and wireless signals to guess the secret key.
- Approaching attack: To get similar channel estimation, Eve can approach to one legitimate user to receive similar acoustic signals from the other legitimate user, but Eve is at least 5cm away from both legitimate users. Eve intends to exploit the similarities to generate the same cryptographic key as Alice or Bob.
- Repeating imitation attack: After Alice and Bob finished key extraction and left the site, Eve finds a partner Dave to imitate the key extraction process conducted by Alice and Bob, i.e., they estimate the response of the inaudible acoustic channel, and then try to extract the same secret key as Alice or Bob.

## 3 FEASIBILITY STUDIES

In this section, we study the feasibility of using acoustic channel information to generate secret key. We first validate three important properties of acoustic channel, including temporal variation, channel reciprocity, and spatial decorrelation. These three properties together serve as the basis of our approach. The frequency band of the tested acoustic signals ranges from 0 Hz to 22 kHz; 22 kHz is the maximum audio frequency that can be played by ordinary mobile devices. We have tested the channel estimation using different devices in four kinds of scenarios. The details of channel estimation and the experiment settings can be found in Section 4.2 and Section 6.1, respectively.

**Temporal variation**: We first observe that due to multipath propagation, acoustic signals could reach the receiving microphone by two or more paths. Causes of multipath propagation include various obstacles, static or mobile, acting as reflectors to the channel paths, changing the reflection,

refraction, and scattering of the signals. Besides, the movements of the transmitter and receiver, e.g., the shaking of the mobile devices, also change the original channel paths.

We can formulate the signal propagation as below:

$$y(t) = \sum_{i=1}^{M} a_i x(t-\tau_i) = \sum_{i=1}^{M} a_i e^{j2\pi f_c(t-\tau_i)} s(t-\tau_i) = h(t) * x(t). \quad (1)$$

In the above formula, we assume that the received acoustic signal $y(t)$ is the sum of signals propagating via $M$ paths, respectively. The signal transmitted along path $i$ has amplitude $a_i$ and delay $\tau_i$, which are determined by reflectors and the signal travel distance. $x(t)$ and $s(t)$ are the transmitted passband and baseband signals at time $t$, respectively; $f_c$ is the center frequency of passband; and $h(t)$ is the **channel impulse response (CIR)**. $h(t) = \sum_{i=1}^{M} a_i e^{j2\pi f_c(t-\tau_i)} \delta(t-\tau_i)$, where $\delta(t)$ is Dirac's delta function [44].

We aim to use the time-varying acoustic channel to generate random sequences. The channel estimation from the received baseband symbol is a discrete output of $h(t)$ sampled every $T_s$ interval [58], which is

$$h[n] = \sum_{i=1}^{M} a_i e^{j2\pi f_c(t-\tau_i)} \delta(t-\tau_i) \mathrm{sinc}(n-\tau_i W), \quad (2)$$

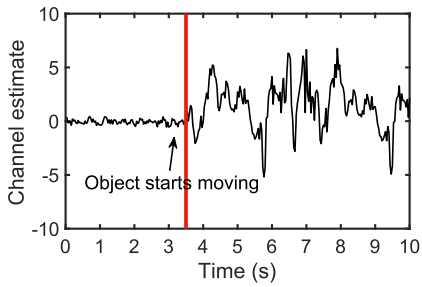
where $\mathrm{sinc}(t) = \frac{\sin(\pi t)}{\pi t}$, $W = \frac{1}{T_s}$. Generally, CIR is regarded as a discrete-time filter in Linear Time-Invariant system, and $h[n]$ is called the $n$th channel tap. Since the acoustic channel is time varying, the channel estimation $h(t)$ is also time varying.

We verify the temporal variation of acoustic channel in experiments. We conduct extensive experiments in corridors and our labs. We estimate the acoustic channels between several different pairs of mobile phones and averaged the first 10 channel taps of all pairs, as shown in Figure 2(a). The channel estimates are from the experiment measurement, and channel estimation procedure is introduced in Section 4. Figure 2(a) illustrates the variation of acoustic channel in a short time. When a reflector, such as a person, starts moving (at around 3.5 s), the acoustic channel changes instantly and distinctly. Thus, the temporal variation of acoustic channel offers enough randomness for key extraction. Besides, as the channel is changing with time, FREE can defend against repeating imitation attack, as the channel is quite different at two moments.

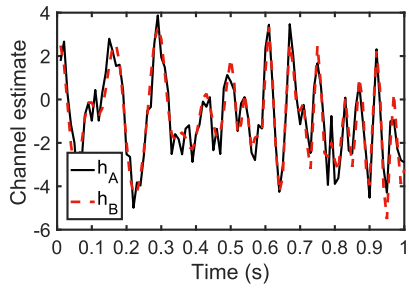
**Channel reciprocity**: We next show that at the same carrier frequency, the multipath and fading of Alice → Bob direction are similar to the Bob → Alice direction in a short period. Channel reciprocity is a fundamental property of signal wave propagation [48], and it is the basis of using acoustic channel to generate the common secret key. More specifically, we use symbol $\mathbf{h}$ to denote the channel parameter, i.e., channel impulse response $h(t)$. To get the channel parameter $\mathbf{h}$, Alice and Bob calculate the channel estimates $\mathbf{h_A}$ and $\mathbf{h_B}$, respectively. Theoretically, $\mathbf{h_A}$ and $\mathbf{h_B}$ should be highly correlated.

We validate the channel reciprocity property in both indoor and outdoor environments. Figure 2(b) shows that the channel estimates $\mathbf{h_A}$ and $\mathbf{h_B}$ in outdoor environments have high correlations. The indoor experiment also shows the same result.

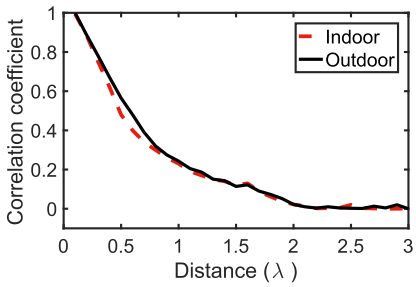
**Spatial decorrelation**: We further observe that when the adversary Eve is more than a half wavelength away from legitimate users Alice and Bob, their mutlipath and fading are uncorrelated. This property guarantees the security of legitimate users' key extraction process. It has been shown that both large-scale fading and small-scale fading contribute to channel variation [14]. The small-scale fading is dominant in our experiment, due to the short travelling distance of the signal and short time duration. In small-scale fading, the signals decorrelate over one half-wavelength of the acoustic signal (e.g., 0.85 cm when the frequency is 20 kHz) [14].




(a) Temporal variation.




(b) Channel reciprocity.




(c) Spatial decorrelation.

Fig. 2. The properties of acoustic channel: temporal variation, channel reciprocity, and spatial decorrelation.

We also validate the spatial decorrelation property of the acoustic channel in both indoor and outdoor environments. Figure 2(c) shows the correlation between Eve's and legitimate user's channel estimation when the distance varies from 0 to 3 wavelengths. The carrier frequency is 20 kHz, and the wavelength $\lambda$ is 1.7 cm. We use Pearson correlation coefficient to measure the correlation between their channel estimation. Pearson correlation coefficient is defined as $\rho_{\mathbf{h_A},\mathbf{h_B}} = \frac{E[(\mathbf{h_A}-\mu_{\mathbf{h_A}})(\mathbf{h_B}-\mu_{\mathbf{h_B}})]}{\sigma_{\mathbf{h_A}}\sigma_{\mathbf{h_B}}}$, where $\sigma_{\mathbf{h_A}}$ and $\sigma_{\mathbf{h_B}}$ are the standard deviation of $\mathbf{h_A}$ and $\mathbf{h_B}$, $\mu_{\mathbf{h_A}}$ and $\mu_{\mathbf{h_B}}$ are the mean of $\mathbf{h_A}$ and $\mathbf{h_B}$, and $E$ is the expectation. We observe that both indoor and outdoor correlation coefficients are lower than 0.2 when the distance is greater than one wavelength. Therefore, Eve cannot receive similar acoustic signal when she is located 5 cm away from the legitimate users.

## 4 DESIGN OF FREE

In this section, we present the architecture and design details of FREE.

### 4.1 Design Rationale

We illustrate the architecture of FREE in Figure 3. The architecture is mainly divided into four components: acoustic channel estimation, quantization, reconciliation, and privacy amplification.

The above three properties make it possible to use acoustic channel randomness to extract secret key. Temporal variation of acoustic channel offers enough randomness for key extraction. Channel reciprocity is the basis for Alice and Bob having similar channel estimation. Spatial decorrelation makes users resistant against eavesdropping and approaching attacks. We next

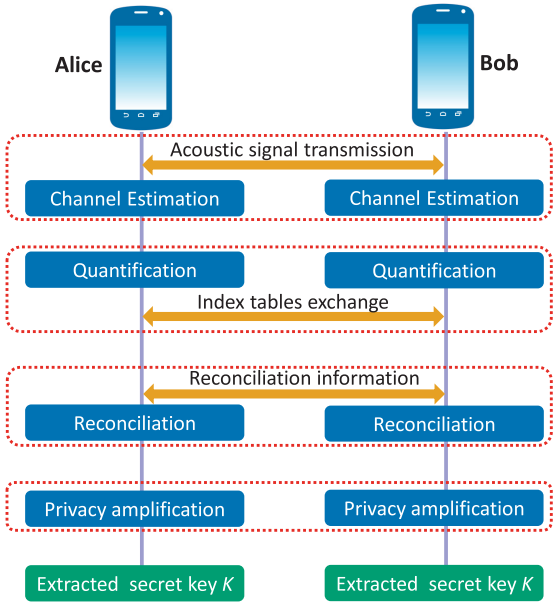



Fig. 3. The architecture of FREE.

address the following two challenges: (1) how to extract similar sequences from random channel response and (2) how to generate a common secret key from the similar sequences.

To extract similar random sequences, Alice and Bob first need to estimate the acoustic channels. There are many methods for channel estimation. Given the implementation on mobile devices, we first consider **Least-Square (LS)** method [46], which has low computation complexity. But the LS method is greatly affected by the noise, so we also consider **Minimum Mean-Square Error (MMSE)** method [51] and **Linear Minimum Mean-Square Error (LMMSE)** method [43]. Alice and Bob transmit acoustic signals to each other and then use the received signals from each other to estimate the acoustic channel and get the channel responses.

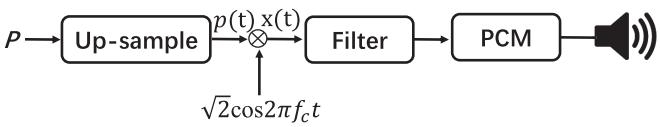
To generate a common secret key, Alice and Bob first need to quantize the channel responses into bit streams. To improve the key generation rate, we choose the adaptive secret bit generation method to quantize a channel tap into a signal bit or multiple bits [22]. After quantization, Alice and Bob get similar bit streams with some mismatched bits. To eliminate the mismatched bits, we design a reconciliation protocol for Alice and Bob to obtain an identical bit stream.

In the above process, the interactions between Alice and Bob may leak some information about the secret bit streams. To eliminate such leakage, Alice and Bob perform the privacy amplification on their own bit stream, respectively. Finally, both Alice and Bob acquire a common secret key to establish a secure wireless channel between them.

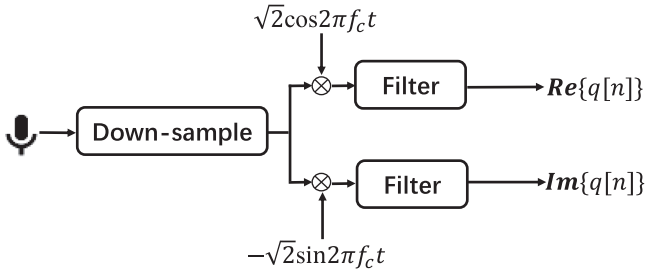
## 4.2 Design Details

*4.2.1 Acoustic Channel Estimation.* We propose an inaudible acoustic signal transmission scheme to estimate the acoustic channel [70]. Since most adults can only hear the sound on the frequency band lower than 18 kHz, we choose to use frequency band from 18 kHz to 22 kHz, with a bandwidth of 4 kHz, which can be captured by microphones embedded in general smartphones and tablets.

Then we present transmitter design and receiver design.




(a) Transmitter design




(b) Receiver design

Fig. 4. Design of transmitter and receiver.

**Transmitter design**: To estimate the acoustic channel, transmitter transmits a known training sequence, which is indicated as $P = \{p_1, p_2, ..., p_L\}$, where $L$ is the sequence length. The training sequence can be any random bit stream. We adopt the channel estimation method in Reference [46] and use a 26-bit Global System for Mobile Communication training sequence, which is well known for good performance on synchronization and widely used in channel estimation. Then the training sequence $P$ is modulated using **Gaussian Filtered Minimum Shift Keying (GMSK)**, which maps bit 0 and 1 to baseband symbol $-1$ and 1, respectively.

Figure 4(a) depicts the system diagram of the transmitter. We first upsample the baseband symbol at a rate of $\frac{f_s}{B}$, where $f_s$ and $B$ represent sampling rate and bandwidth, respectively. The purpose of upsampling is to smooth discontinuity, by zero padding and low-pass filtering [44]. Let $f_c$ represent the center frequency of passband. We transform the signal frequency into the passband signal: $x(t) = \sqrt{2}\cos(2\pi f_c t)p(t)$, where $p(t)$ and $x(t)$ represent upsampled baseband and passband signals, respectively.

To remove the noise outside the transmission band, we filter the signal $x(t)$ with passband filter from $f_c - \frac{B}{2}$ to $f_c + \frac{B}{2}$ Hz. Then, the processed signal is transmitted by the speaker. Since the training sequence is fixed, the generated signal is also fixed. To reduce the computation overhead, we save the generated signal as a Waveform Audio file with 16-bit Pulse Coded Modulation, which can be played by most of mobile devices.

To avoid inter-frame interference, the transmitter cannot continuously transmit training sequence with no gap. We insert zeros at the end of training sequence, as shown in Figure 5. The gap between training sequences must be long enough to prevent a frame from interfering the previous frame. It has been shown that 24 zeros are long enough to avoid inter-frame interference in our experiments. Finally, we refer to the training sequence padded with zeros as a frame, consisting of 50 bits. Since the baseband symbol interval is $T_s = \frac{1}{B} = 0.25$ ms, each frame lasts 12.5 ms.

**Receiver design**: Figure 4(b) demonstrates the processing of received signal in receiver's side. The received signal $y[n]$ from microphone is converted into baseband symbol $q[n]$ as follows:

$$\begin{aligned} q[n] &= \sqrt{2}\cos(2\pi f_c t)y(t) - j\sqrt{2}\sin(2\pi f_c t)y(t) \\ &= \sqrt{2}e^{-j2\pi f_c t}y(t), \end{aligned} \tag{3}$$

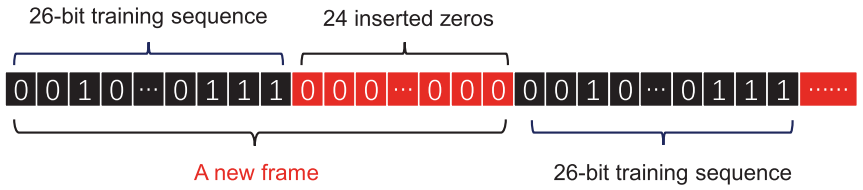



Fig. 5. Training sequence with inserted zeros.

where $t$ is the time when the $k$th baseband symbol is sampled, that is, $t = k \times T_s$, where $T_s$ is the corresponding baseband symbol interval. We multiply $y(t)$ with $\sqrt{2}cos(2\pi f_c t)$ and $-sin(2\pi f_c t)$, and get the real and imaginary parts of received signal, respectively. Then both of real and imaginary parts are processed by low-pass filtering and down-sampling.

Then the receiver uses energy detection and cross-correlation to detect the arrival of a frame. We use energy detection to roughly determine the starting point of a frame: We set a threshold $\delta$ to measure whether the magnitude of three consecutive symbols is larger than the threshold, which indicates the starting point of a frame. The threshold $\delta$ is set empirically in our study, and the parameter relies on the sensitivity of the microphone and the volume of the speaker. After that, we use cross-correlation method to find precise starting point of the frame.

To estimate the acoustic channel, we first consider LS method [46], which is of low computation overhead. In LS channel estimation, we first need to determine a reference length $X$ and a guard length $Y$, and $L = X + Y$ is the length of the training sequence. The guard length $Y$ determines the number of channel taps that we can estimate. To balance the number of channel taps and the estimation quality, we choose $X = 16$ and $Y = 10$ in our study. Readers can find more details in Reference [46].

As mentioned above, the training sequence is denoted as $P = \{p_1, p_2, ..., p_i, ..., p_L\}, p_i \in \{-1, +1\}$. Then the corresponding circulant matrix $\mathbf{M} \in \mathbb{R}^{X \times Y}$ is formed as:

$$\mathbf{M} = \begin{bmatrix} m_Y & m_{Y-1} & \cdots & m_1 & m_0 \\ m_{Y+1} & m_Y & \cdots & m_2 & m_1 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ m_{Y+X-1} & m_{Y+X-2} & \cdots & m_X & m_{X-1} \end{bmatrix}. \tag{4}$$

Let $y = \{y_1, y_2, ..., y_i, ..., y_{X+Y}\}$ represent the received training sequence. Then we get the acoustic channel estimation as follows:

$$\hat{h}_{LS} = (\mathbf{M}^H \mathbf{M})^{-1} \mathbf{M}^H y_Y, \tag{5}$$

where $\mathbf{M}^H$ and $\mathbf{M}^{-1}$ indicate the Hermitian and inverse matrices of $\mathbf{M}$, respectively. $y_Y = \{y_{Y+1}, y_{Y+2}, ..., y_{Y+X}\}$, which offers the random source for key generation.

From Equation (5), we can see that the most computationally expensive part is the computation of $(\mathbf{M}^H \mathbf{M})^{-1}\mathbf{M}$, which is a matrix multiplication, and its computation complexity is $O(X \times Y)$. Given that both $X$ and $Y$ are small constants, the computation overhead is very low for mobile devices.

LS method has been widely used for channel estimation. However, its performance is greatly affected by the channel noise. When there is obvious channel noise, we need to mitigate the influence of channel noise. MMSE method exploits the statistical characteristics of the channel to get more accurate channel estimation. LMMSE method is a simplified version of the MMSE method. It greatly reduces the computation complexity of the channel estimation.

**MMSE channel estimation**: MMSE channel estimation considers the influence of channel noise in statistical characteristics. The basic idea of MMSE is to minimize the Mean-Square Error, as shown in

$$J(\hat{h}) = E\{(h - \hat{h}) * (h - \hat{h})\}, \tag{6}$$

where $h$ is the actual channel frequency response and $\hat{h}$ is the estimated channel frequency response. $\hat{h}$ can be calculated by Wiener filtering,

$$\hat{h} = w_1 y_1 + w_2 y_2 + \cdots + w_Y y_Y = W^T y, \tag{7}$$

where $y$ is the received training sequence and $W^T$ is the channel taps of Wiener filtering. Let $e$ represent the error between the actual channel frequency response and the estimated channel frequency response, i.e., $e = h - \hat{h}$. To minimize Equation (6), let $\frac{\partial J(\hat{h})}{\partial w_k} = 0$, then we have

$$\frac{\partial J(\hat{h})}{\partial w_k} = \frac{\partial E[(h - \hat{h}) * (h - \hat{h})]}{\partial w_k} = \frac{\partial E[(e * e)]}{\partial w_k} = -2E(e * y_k) = 0. \tag{8}$$

Put Equation (7) into Equation (8), then we have

$$\begin{aligned} E(e * y_k) &= E\{(h - W^T y) * y_k\} \\ &= E(h * y_k) - W^T E(y * y_k) = 0 \\ &\Rightarrow W^T = R_{hy} R_{yy}^{-1}, \end{aligned} \tag{9}$$

where $R_{hy}$ is the cross-correlation matrix of the received training sequence and channel frequency response and $R_{yy}$ is auto-correlation matrix of the received training sequence, i.e.,

$$R_{hy} = R_{hh} M^H, R_{yy} = M R_{hh} M^H + \sigma^2 I, \tag{10}$$

where $R_{hh}$ is the auto-correlation matrix of channel frequency response, $\sigma^2$ is the variance of Gauss white noise, and $I$ is the identity matrix. Then, we can derive the channel estimation as

$$\hat{h}_{MMSE} = R_{hy} R_{yy}^{-1} y = R_{hh} M^H (M R_{hh} M^H + \sigma^2 I)^{-1} y = R_{hh} \{R_{hh} + \sigma^2 (M^H M)^{-1}\}^{-1} \hat{h}_{LS}. \tag{11}$$

From the above analysis, we can see that MMSE estimation method uses statistical characteristics and considers the effects of channel noises. Compare with LS method, MMSE method reduces the interference of noise and enhance the accuracy of channel estimation. However, MMSE method requires more complex matrix inversion operation, which limits its application scenarios. Therefore, it is necessary to simplify the MMSE method.

**LMMSE channel estimation**: LMMSE replaces the $(MM^H)^{-1}$ in Equation (11) with $E\{(MM^H)^{-1}\}$. Thus, LMMSE replaces the instantaneous power with the average power. With the same modulation scheme and the same probability of different points in the modulation method, we have

$$E\{(MM^H)^{-1}\} = E\left\{\left|\frac{1}{M_k}\right|^2\right\} I, \tag{12}$$

where $I$ is an identity matrix.

Let $\beta = E(|M_k|^2)E(|\frac{1}{M_k}|^2)$ and $SNR = \frac{E|M_k|^2}{\sigma^2}$, then the channel estimation of LMMSE method is

$$\hat{h}_{LMMSE} = R_{hh} \left(R_{hh} + \frac{\beta}{SNR} I\right)^{-1} \hat{h}_{LS}, \tag{13}$$

where $\beta = 1$ when using the GMSK modulation scheme. Compared with MMSE method, LMMSE can reduce the computation complexity.

To get enough similar random sequences for key extraction, Alice and Bob need to continuously transmit modulated acoustic signal to each other. After acoustic communication accomplished, both Alice and Bob get channel responses, which are denoted by $\hat{H}_A$ and $\hat{H}_B$, respectively.

*4.2.2 Quantization.* Once Alice and Bob get the acoustic channel response, they need to quantize these channel responses to bit streams. To improve secret key generation rate, we use the **Adaptive Secret Bit Generation (ASBG)** method to quantize a channel tap into a signal bit or multiple bits [22]. Multiple bit quantization generates more secret bits but with higher bit mismatch ratio, and thus it has higher communication overheads for reconciliation. Users can choose to use single or multiple bit quantization depending on the application scenarios. When the user requires faster key generation, he/she can choose to use multiple bit quantization. On the contrary, he/she can choose to use single bit quantization. For example, in file transfer, we can use single bit quantization, because we are less sensitive to latency in file transfer; while in pay-and-go scenarios, we may prefer multiple bit quantization.

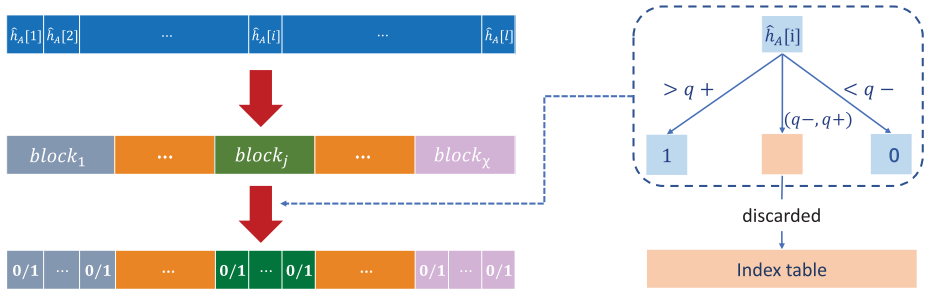
To illustrate the single bit quantization method, we suppose that Alice and Bob each get a channel tap sequence, denoted by $\hat{H}_A = \{\hat{h}_A[1], \hat{h}_A[2], \ldots, \hat{h}_A[l]\}$ and $\hat{H}_B = \{\hat{h}_B[1], \hat{h}_B[2], \ldots, \hat{h}_B[l]\}$, respectively, where $l$ is the length of channel estimation. The process of single bit quantization method is illustrated as follows:

- Alice divides $\hat{H}_A = \{\hat{h}_A[1], \hat{h}_A[2], \ldots, \hat{h}_A[l]\}$ into small blocks of size $\chi$, which is an adjustable parameter. Bob also performs the same operation.
- For each block, they calculate two adaptive thresholds $q^+$ and $q^-$, $q^+ = mean + \alpha * \sigma$ and $q^- = mean - \alpha * \sigma$, where $\alpha > 0$, $mean$ is the average of the magnitude of the $\hat{h}$ in a block, and $\sigma$ is the standard deviation.
- Alice compares her channel estimation to the two thresholds, $q^+$ and $q^-$. If channel estimate $\hat{h}_A[i] > q^+$, then $\hat{h}_A[i]$ is recorded as 1; if $\hat{h}_A[i] < q^-$, then $\hat{h}_A[i]$ is recorded as 0; when $\hat{h}_A[i]$ lies in between $q^+$ and $q^-$, then $\hat{h}_A[i]$ is discarded, and the index $i$ is recorded in an index table $T_A$. Above steps are illustrated in Figure 6(a). Bob performs the same operations on $\hat{H}_B$, and generates index table $T_B$.
- After the above operations accomplished, Alice and Bob exchange their index table, $T_A$ and $T_B$. They only keep the channel estimation that are not discarded by either of them. Finally, they obtain the bit stream $S_A$ and $S_B$, respectively.

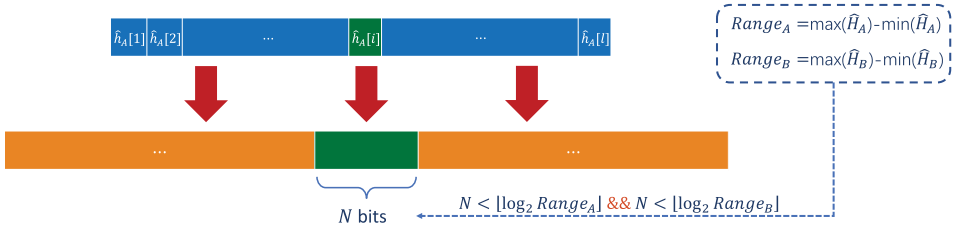
In the single bit quantization, the adaptive thresholds are calculated for each block separately. We set the block size $\chi$ as 30 empirically. We can also quantize each channel estimates to multiple bits. The process of multiple bit quantization is as follow:

- Alice finds the minimum and maximum of $\hat{H}_A$ to calculate the $Range_A$, $Range_A = \max(\hat{H}_A) - \min(\hat{H}_A)$. Bob finds $Range_B$.
- Determine $N$, which is the number of bits for quantizing a channel estimation. $N$, must satisfy $N < \lfloor log_2 Range_A \rfloor$ and $N < \lfloor log_2 range_B \rfloor$.
- Divide $Range$ into $M = 2^N$ intervals, and choose $N$-bit assignment for each of $M$ intervals. Above steps are illustrated in Figure 6(b). To reduce the mismatch ratio, we use Gray code to encode them.
- For each channel estimation, Alice and Bob extract $N$ bits, according to their location in $M$ intervals. Finally, they obtain the bit stream $S_A$ and $S_B$, respectively.

*4.2.3 Reconciliation.* Alice and Bob aim to generate an identical bit stream as the secret key $K$. Due to the mismatched bits in $S_A$ and $S_B$, they need to perform reconciliation to remove these mismatched bits. We consider using extended Binary Gray Code $G_{24}$ for reconciliation [53]. $G_{24}$




(a) Single bit quantization of Alice.




(b) Multiple bit quantization of Alice.

Fig. 6. The processes of single bit quantization and multiple bit quantization on Alice's side.

can encode 12 bits to 24 bits, correcting any 3 error bits and monitoring 7 error bits [7]. Due to the high correlations between $S_A$ and $S_B$, $S_A$ and $S_B$ can be treated as two codewords, which are both distorted from a common bit sequence, according to the encoding theory.

Next, we illustrate the process of reconciliation. At first, both Alice and Bob encode their bit streams to Gray Code sequence. As shown in Figure 7, Alice gets Gray Code sequence $W_A = E(S_A) = [S_A, F_A]$, where $E(\cdot)$ is an encoding function, and $F_A$ is the parity check sequence. Then, Alice sends the difference $Z$ between $S_A$ and $F_A$ to Bob. Bob calculates the codeword $\bar{W}_B = [S_B, S_B - Z]$ and decodes it to $\bar{S}_A = D(\bar{W}_B)$. Next, Bob calculates the number of mismatched bits between $\bar{S}_A$ and $S_B$. If it is greater than 3, then Bob will discard the sequence $S_B$ and notify Alice to discard sequence $S_A$. Otherwise, Bob will replace $S_B$ with $\bar{S}_A$, and generate a common sequence $K = S_A = \bar{S}_A$ with Alice.

*4.2.4 Privacy Amplification.* In the above reconciliation stage, Alice sends some information such as $Z$ to Bob through public wireless channel. Thus, Eve can deduce some privacy information about the secret sequence. Privacy amplification can mitigate this problem by reducing the length of the secret sequence $K$. We can adopt the methods based on *leftover hash lemma* [22, 45], which is constructed by a universal hash function family. The details are as follows (take Alice's operations as an example):

- Step 1: Selects a big prime $m$ with 256-bit length, and sends it to Bob;
- Step 2: Decomposes the key $K$ into $r+1$ digits $K = < k_0, k_1, \ldots, k_i, \ldots k_r >$, where $k_i \in 0, 1, \ldots, m-1$;
- Step 3: Picks $a = < a_0, a_1, \ldots, a_i, \ldots a_r >$ at random, where $a_i \in 0, 1, \ldots, m-1$, and sends it to Bob;
- Step 4: Calculates $h_a(K) = (\sum_{i=0}^{r} a_i k_i)\ mod\ m$.

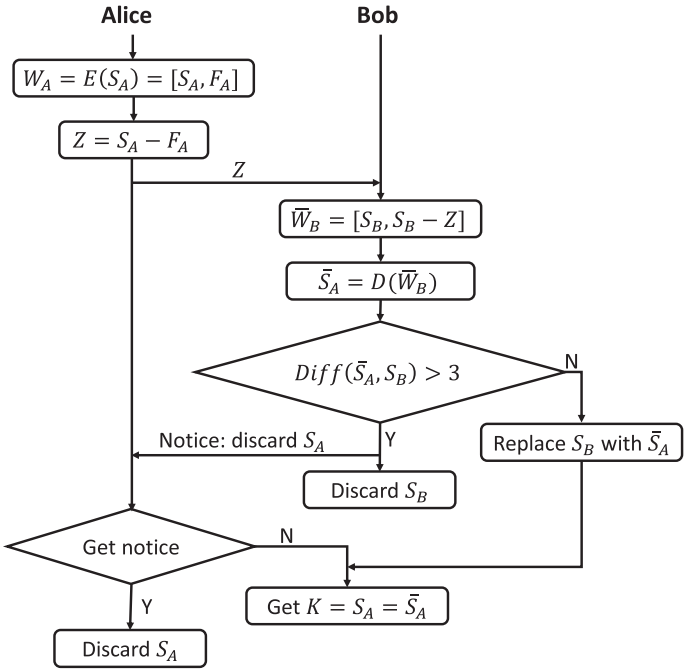



Fig. 7. The flowchart of reconciliation.

Privacy amplification generates shorter bits with higher entropy. After going through the privacy amplification, Alice and Bob acquire the final secret key.

## 5 SECURITY ANALYSIS

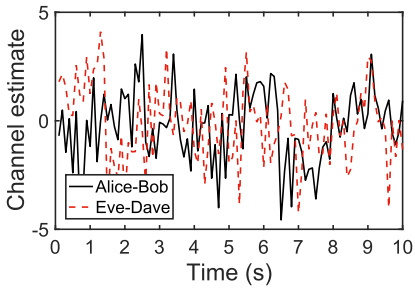
In this section, we analyze the security performance of FREE.

### 5.1 Against Eavesdropping Attack

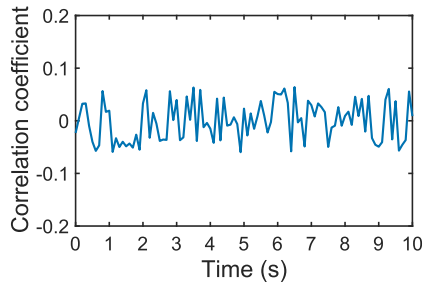
In the eavesdropping attack, Eve can overhear all communication transmitted through public acoustic channel and wireless channel. Then we analyze what information can be obtained by Eve. In the channel estimation stage, Eve can listen and receive all of the acoustic signal transmitted by Alice and Bob. Since Eve receives the acoustic signal through different acoustic channel, Eve cannot get the same channel parameters as Alice or Bob. In quantization stage, she can know the exchanged index tables, which records the positions of removed bits. Since Eve does not know the bit stream $S_A$ and $S_B$, she cannot know the content of removed parts either. In the information reconciliation stage, Eve can get the difference $Z$ between $S_A$ and $F_A$, and the discarding notice. But this leakage risk is eliminated by privacy amplification. Thus, Eve cannot deduce the secret key.

### 5.2 Against Approaching Attack

In the approaching attack, Eve approaches to Alice or Bob to receive the acoustic signals transmitted from the other party. Eve wants to use her proximity to Alice or Bob to estimate similar acoustic channel parameters. But the spatial decorrelation of acoustic channel, presented

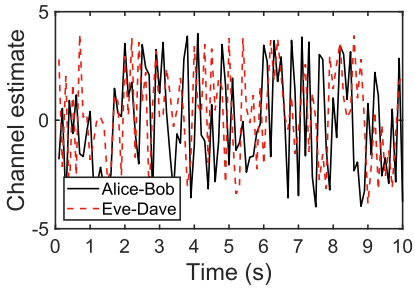



(a) Channel estimation indoor

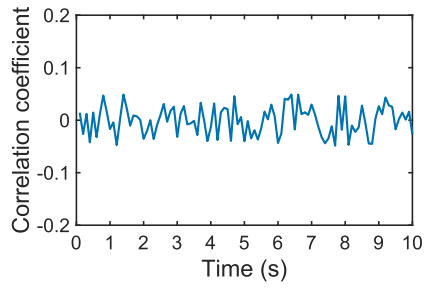



(b) The correlation coefficients of channel estimation indoor

Fig. 8. The channel estimation and their correlation coefficients of Alice–Bob and Eve–Dave at the same indoor location.




(a) Channel estimation outdoor




(b) The correlation coefficients of channel estimation outdoor

Fig. 9. The channel estimation of Alice–Bob and Eve–Dave at the same outdoor location and their correlation coefficients.

in Section 3, illustrates that the correlation is lower than 0.2 when the distance is greater than one wavelength $\lambda$. For example, when the center frequency is 20 kHz and the sound speed is 340 m/s, the wavelength $\lambda = \frac{340m}{20k} = 1.7$ cm. In real-life scenarios, we cannot allow others to put their mobile devices so close to our own devices. As a result, Eve cannot guess the similar secret key by approaching the legitimate users.

### 5.3 Against Repeating Imitation Attack

Under the repeating imitation attack, after legitimate users left the site, Eve finds a partner Dave to imitate the communication between Alice and Bob. Their goal is to estimate similar acoustic channel to generate the same secret key as Alice and Bob. But the acoustic channel varies all the time due to the moving of mobile devices and the dynamic environment, as presented in Section 3. Therefore, Eve and Dave cannot capture the same acoustic channel response even in the same place.

This has been validated by our experiments. We studied the channel response of Eve and her partner and compared them to the channel estimation between Alice and Bob. We compute the correlation coefficients of the two channel estimation in both indoor and outdoor scenarios. Results are shown in Figure 8 and Figure 9. We recorded the channel estimates of Alice-Bob for 10 s in indoor environment. After both Alice and Bob left from the original positions, we recorded

Table 1. Experiment Scenarios

| Index | State | Environment |
|---|---|---|
| A | Static | Indoor |
| B | Static | Outdoor |
| C | Mobile | Indoor |
| D | Mobile | Outdoor |

the channel estimates of Eve-Dave in the next 10 s. We aligned them in Figure 8(a) for a better comparison. We can observe that channel estimation are different in different periods. The outdoor experiments show the same results. We also computed the correlation coefficients of Alice–Bob and Eve–Dave, as shown in Figure 8(b). We can see that the correlation coefficients range from −0.1 to 0.1, showing the irrelevance between the two estimations. Figure 9 shows the same results of the experiments conducted outdoor. Thus, Eve cannot get similar channel estimation as the legitimate users.

## 6 EVALUATION

In this section, we evaluate the performance of FREE.

### 6.1 Methodology

We have conducted extensive experiments with four participants, named Alice, Bob, Eve, and Dave (Eve's partner). Each of them holds a mobile device (e.g., Nexus 7, MEIZU MX 6, Xiaomi 3), all equipped with a microphone and a speaker. The illegitimate users, Eve and Dave, both are located more than 5 cm away from Alice and Bob.

*6.1.1 Implementation and settings.* We implement Android-based prototype of FREE on the mobile devices. We use Bluetooth to offer the public wireless channel between mobile devices. We call the Android API, AudioRecord(*) and AudioTrack(*), to transmit and receive acoustic signals, and the sampling rate is 44.1 kHz. We set up Alice's device facing Bob's device. At the beginning, Alice, as an initiator, sends Bob a synchronization signal. After receiving Bob's acknowledgement, Alice starts to transmit acoustic signal with band from 18 to 22 kHz, in every 16 ms. After receiving Alice's acoustic signal, Bob starts to transmit her own acoustic signal with the same band, in every 16 ms. At the same time, they shake the mobile devices together to generate more randomness. Alice's device is also facing to Bob's device when the two devices are shaken, and it is better to make the microphone and speaker of one device face the speaker and microphone on the other device, and shake the two devices with similar speed. Then, Alice and Bob start to compute the channel estimation by using the LS method and quantize them into secret bit streams. They exchange their index table to finish the quantization by Bluetooth. Next, they perform reconciliation to generate the same secret bit stream and perform the privacy amplification to extract a common secret key.

We conduct experiments in different scenarios, i.e., indoor, outdoor, mobile, and static, as shown in Table 1. Mobile means that users move around in a small area, but keep the distance between them within 2 m; static means that the users keep still when generating the secret key. The speaker volume is at the highest level when we evaluate the influences of distance, channel estimation methods, quantization and encoding methods.

*6.1.2 Metrics.* To evaluate the performance, we consider the following three metrics.

- **Bit Generation Rate (BGR)**: BGR is defined as the number of generated secret bits in a second. The more secret bits generated, the higher the efficiency of key extraction.

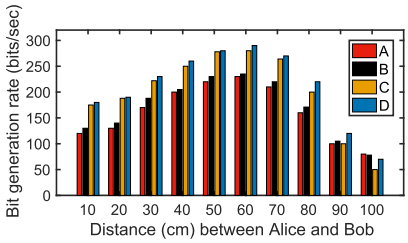



Fig. 10. Bit generation rate in different distance.

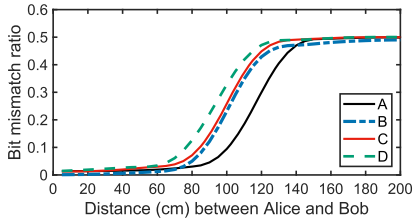


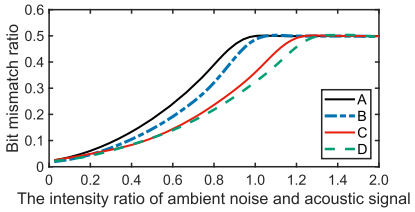

Fig. 11. Bit mismatch ratio in different distance.

- **Bit Mismatch Ratio (BMR)**: BMR is defined as the number of mismatch bits over the number of all generated bits in a second. The lower bit the mismatch ratio, the higher the robustness of key extraction.
- **Randomness and Entropy (RE)**: RE is used to evaluate the quality of the generated keys. We use an extensively used randomness tool, NIST test, to measure the randomness of the generated keys. Besides, we compute the entropy of the generated secret keys. The higher the entropy, the better quality of the generated secret key.

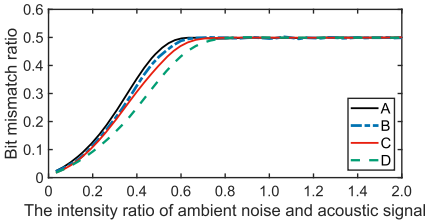
### 6.2 Influence of Distance

To study how the distance affects the performance of FREE, we conduct the experiments at different distances. Figure 10 illustrates the secret bit generation rates of legitimate users under different distances for four kinds of scenarios. We find that the key generation rate is lower than 100 bits/s when the distance is greater than 90 cm. The reason is that the correlation of Alice's and Bob's channel estimation decrease with increasing distance. In real scenarios, the signals transmitted by Alice and Bob are not transmitted exactly along the same path, due to the distance between the microphone and speaker embedded on the mobiles. To reduce the system error, one user makes the microphone and speaker of her device face the speaker and microphone on the other user's device, respectively. But the system error still increases with increasing distance. We also find that the key generation rate is lower than 200 bits/s when the distance is lower than 40 cm. The reason is that there is not enough randomness for key extraction when Alice and Bob are located closely. We find that the bit generation rate is greater than 260 bits/s, when the distance is between 50 and 70 cm. The reason is that Alice and Bob can capture more channel randomness and the system error is small in this distance range. To generate a 512-bit cryptographic key, FREE only needs a couple of seconds. We also find that the key generation rate in the outdoor environment is higher than that in the indoor environment. The bit generation rate in mobile state is higher than that in the static state. The reason is that the outdoor environment and mobile state offer more channel diversity and sufficient randomness.

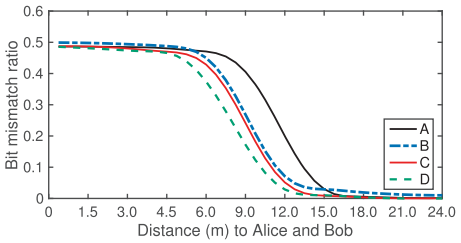
Figure 11 illustrates the bit mismatch ratios of legitimate users in different distance for the four scenarios. We find that the bit mismatch ratio increases obviously when the distance is greater than 60 cm. The main reason is that the correlation between Alice's and Bob's channel estimation decreases with increasing distance, due to the increasing system error. The propagation paths of the signal Alice receives are not exactly the same as that of the signal Bob receives, since the speaker and the microphone are not co-located on a device. Therefore, the longer the distance between the devices, the greater the error of channel estimation. The mismatch ratio is around 0.5 when the distance is greater than 120 cm. Thus, FREE cannot work well beyond this distance range; 120 cm can be set as the authenticate distance, and a longer distance (e.g., 2 m) can be set as the safe distance [64]. Thus, a device has the maximum bit mismatch ratio 0.5 when it is out of




(a) The bit mismatch ratio in different noise intensities.




(b) The bit mismatch ratio when Alice and Bob receive different levels of ambient noises.




(c) The bit mismatch ratio when the ambient noise source in different distances.

Fig. 12. The bit mismatch ratio in different ambient noise conditions.

the safe distance. The bit mismatch ratio under the outdoor and mobile scenarios increases earlier than that of other scenarios due to more complicated and fast-varying acoustic channel.

### 6.3 Influence of Ambient Noises

We also evaluate the performance of FREE in the environment with ambient noises. To study the effects of ambient noises, another mobile device is located 5 m away from both Alice and Bob, sending noises with frequency from 18 to 22 kHz to interfere the key extraction process of Alice and Bob. Figure 12(a) illustrates the bit mismatch ratio of Alice and Bob in different noise intensities and scenarios. The $X$ axis denotes the intensity ratio of the ambient noises and the acoustic signal transmitted between Alice and Bob. We can see that the bit mismatch ratio increases with the increasing intensity ratio. The bit mismatch ratio is under 0.2 when the intensity ratio is smaller than 0.5. But the mismatch ratio approaches 0.5 (the maximum value) when the intensity ratio is larger than 1. Therefore, we can turn up the intensity of transmitted acoustic signal as high as possible to reduce the bit mismatch ratio of FREE in the environment with obvious ambient noises.

Figure 12(b) illustrates the bit mismatch ratio when Alice and Bob receive different ambient noises due to their location difference. To implement this scenario, another mobile device (the ambient noise source) is located 100 cm away from Alice and 50 cm away from Bob. As shown in Figure 12(b), the $X$ axis denotes the intensity ratio of the ambient noises transmitted by the ambient noise source and the acoustic signal transmitted between Alice and Bob. We can see the bit mismatch ratio approximates to 0.5 when the intensity ratio is larger than 0.6. The bit mismatch ratio grows rapidly when the two devices receive different levels of noise. But we can also turn up the intensity of transmitted acoustic signal to reduce the bit mismatch ratio.

Figure 12(c) illustrates the bit mismatch ratio when the ambient noise source is in different distances to Alice and Bob. The ambient noise source transmits the acoustic signal with the same intensities of Alice and Bob. As shown in Figure 12(c), the $X$ axis denotes the ambient noise source's

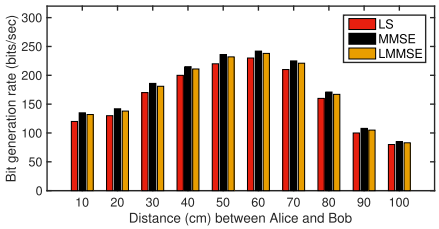



Fig. 13. Bit generation rate under different channel estimation methods.

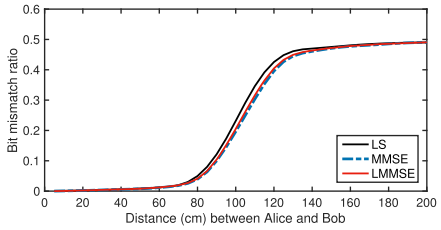



Fig. 14. Bit mismatch ratio under different channel estimation methods.

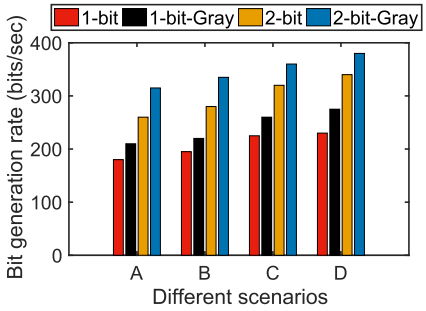



Fig. 15. Bit generation rate in different quantization and encoding methods.

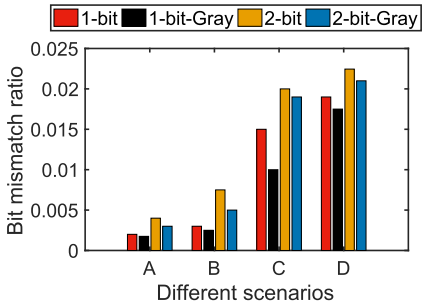



Fig. 16. Bit mismatch ratio in different quantization and encoding methods.

distance to Alice and Bob. We can see the bit mismatch ratio decreases when the distance grows. When the distance is larger than 15 m, the bit mismatch ratio is smaller than 0.05.

### 6.4 Influence of Channel Estimation Methods

We evaluate the performance of FREE when using different channel estimation methods, including LS, MMSE, and LMMSE. As shown in Figure 13 and Figure 14, the $X$ axis denotes the distance between Alice and Bob; the $Y$ axis denotes the bit generation rate and bit mismatch ratio, respectively. We can see that LS method has the lowest bit generation rate and highest bit mismatch ratio due to the channel noises. On the contrary, MMSE has the highest bit generation rate and the lowest bit mismatch ratio due to its consideration of channel noise. Compare with MMSE method, LMMSE method has a slightly lower bit generation rate and higher bit mismatch ratio. It illustrates that the simplified MMSE cannot reach the same performance as MMSE. However, the performance of LMMSE is still better than that of LS.

### 6.5 Influence of Quantization and Encoding Methods

We also evaluate the performance of FREE with different quantization and encoding methods. In this experiment, the quantization mainly includes 1-bit quantization method and 2-bit quantization method; the encoding methods mainly include binary encoding method and gray encoding method. The distance range is 50–70 cm.

Figure 15 compares the bit generation rate of FREE with different quantization, encoding methods under different scenarios. We can see that the key generation rate of 2-bit quantization is higher than that of 1-bit quantization. This is because 2-bit quantization can extract more bits to

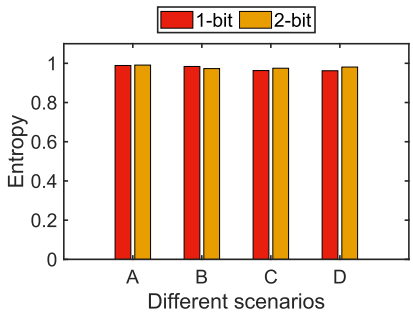



Fig. 17. Entropy of different quantization methods.

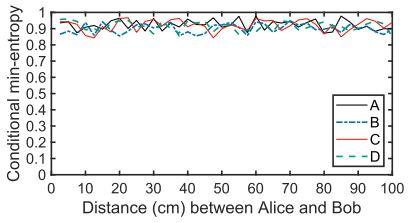



Fig. 18. Conditional min-entropy of generated keys.

generate more secret bits. The key generation rate of gray encoding is higher than that of binary encoding. This is because gray encoding has a stronger error correction capability.

Figure 16 compares the bit mismatch ratio of FREE with different quantization, encoding methods under different scenarios. We can see that bit mismatch ratio of 2-bit quantization is higher than that of 1-bit quantization. It is because that 2-bit quantization causes more quantization errors. The bit mismatch ratio of gray encoding is lower than that of binary encoding. It is also because that gray encoding has a stronger error correction capability.

Figure 17 shows the entropy of secret bits with different quantization methods. We can see there is no obvious difference between 1-bit quantization and 2-bit quantization. Both of them have the entropy ranging from 0.95 to 0.99.

### 6.6 Randomness of Extracted Key

We also validate the randomness of the key generated by FREE. For experimental setting, the distance between Alice and Bob is ~50–100cm, and we use the single bit quantization method and gray coding. Previous efforts used NIST test to measure the randomness of the generated keys [15, 20, 38, 41, 55, 64, 69, 71, 72, 74, 75, 79]. We also utilize NIST test to measure the generated 300 sequences, and compute their $p$ values for eight types of tests, as listed in Table 2. The last column in Table 2 is the testing results on 300 **pseudo-random sequences (PRS)** generated by a random sequence generator. The sequence is marked as random if all $p$ values are greater than 0.05. We can see that the generated secret keys pass all types of tests, and the extracted keys have similar performance as the pseudo-random sequences. Thus, the extracted keys have good quality in randomness.

### 6.7 Conditional Min-Entropy of Generated Keys

To evaluate the security of generated keys, we compute their (average-case) conditional min-entropy $\widetilde{H}_\infty(X/Y)$ [13]. $\widetilde{H}_\infty(X/Y)$ is defined as

$$\widetilde{H}_\infty(X/Y) \stackrel{def}{=} -\log(E_{y\leftarrow Y}[\max_x Pr[X = x|Y = y]]) \tag{14}$$

in Reference [12] and Reference [13]. The adversary Eve was located 5 cm away from Alice. Previous efforts also used conditional min-entropy to measure the knowledge of the adversary about the secret keys [9, 25, 71–73]. To simplify the computation of conditional min-entropy, the lengths of variable $X$ and variable $Y$ are set to be 1. We get the statistical results of the average conditional probability $Pr[X = x|Y = y]$ and the expectation $E_{y\leftarrow Y}[\max_x Pr[X = x|Y = y]]$ from the bits at the corresponding positions of these two bit strings. Figure 18 shows the (average-case)

Table 2. NIST Statistical Test Results

| Test | A | B | C | D | PRS |
|---|---|---|---|---|---|
| Monobit Frequency | 0.662 | 0.745 | 0.911 | 0.773 | 0.811 |
| Longest Run of 1s | 0.714 | 0.654 | 0.843 | 0.892 | 0.726 |
| FFT | 0.509 | 0.782 | 0.838 | 0.737 | 0.811 |
| Approximate Entropy | 0.801 | 0.783 | 0.903 | 0.887 | 0.884 |
| Cumulative Sums (Fwd) | 0.570 | 0.642 | 0.915 | 0.793 | 0.623 |
| Cumulative Sums (Rev) | 0.773 | 0.752 | 0.902 | 0.917 | 0.715 |
| Block Frequency | 0.717 | 0.736 | 0.825 | 0.914 | 0.724 |
| Runs | 0.753 | 0.796 | 0.821 | 0.833 | 0.819 |
| Serial | 0.505 | 0.674 | 0.818 | 0.839 | 0.693 |
| | 0.602 | 0.718 | 0.772 | 0.790 | 0.766 |

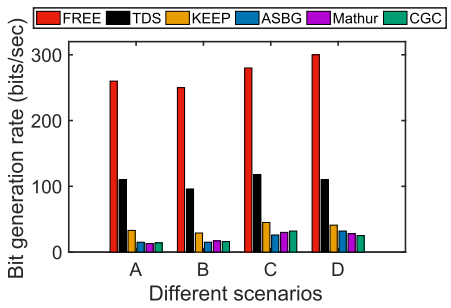



Fig. 19. Comparison of bit generation rate.

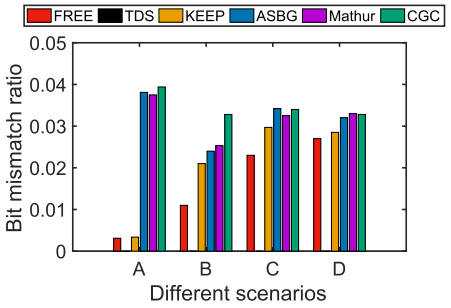



Fig. 20. Comparison of bit mismatch ratio.

conditional min-entropy in different scenarios. We can see the conditional min-entropy in these scenarios are all greater than 0.82, and some of them approximate to 0.99. Thus, the generated keys show satisfactory security against the eavesdropping of the adversary, Eve.

### 6.8 Comparisons of Key Extraction Approaches

We compare FREE with existing key generation approaches, i.e., Mathur et al. [34], ASBG [22], CGC [30], KEEP [63], TDS [64]. Mathur et al. [34] extracted a secret key from the unauthenticated wireless channel. ASBG [22] extracts secret key from wireless signal strength in real environments. CGC [30] exploits channel response to extract secret keys. KEEP [63] extracts secret keys by using the channel state information from orthogonal frequency-division multiplexing. TDS [64] exploits the fine-grained channel state information that is recorded by **channel state information (CSI)** tool to extract secret key.

First, we need to align the baseline of comparisons. In the approach proposed by Mathur et al., there are two parameters, $\alpha$ and $m$. We set $\alpha$ and $m$ as 0.35 and 2, respectively, as this is the best combination for bit generation rate. For ASBG, CGC, and KEEP, We set $\alpha$ and fragment size as 0.35 and 50, respectively, as these parameters are best for low bit mismatch ratio. For TDS, we set block size $\chi$ as 6 in static sate and 4 in mobile state. The distance between Alice and Bob is within 4 cm. For FREE, we set block size $\chi$ as 30. The distance between Alice and Bob is within 80 cm.

We report the bit generation rates of different approaches in Figure 19. We can see FREE has a much higher bit generation rate than other approaches.

We report the bit mismatch ratio of different approaches in Figure 20. FREE has around 0.5% to 3.0% bit mismatch ratio, which is lower than other approaches except for TDS.

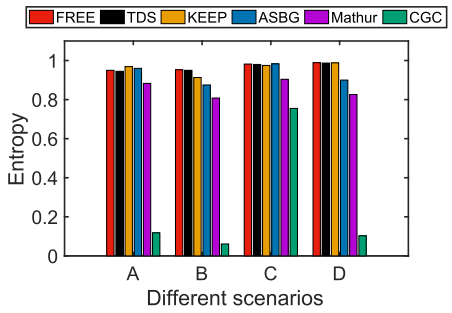



Fig. 21. Comparison of entropy.

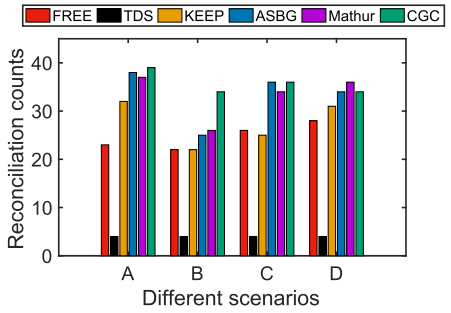



Fig. 22. Comparison of number of messages sent during reconciliation.

We report the entropy in Figure 21. The entropy measures the uncertainty of the keys generated by different methods. We can see that the average entropy of FREE is approximately 0.98, which is satisfactory, compared with other methods.

We report the number of messages sent during reconciliation in Figure 22. FREE needs to exchange reconciliation information for ~22–28 times in a second. This is because FREE needs one reconciliation message for every 12 bits in a bit sequence. In our experiments, we count the mean values of FREE's key generation rate and bit mismatch ratio in different scenarios. The statistical results show that, compared with the state-of-the-art methods, FREE improves the key generation rate by 38.46% and reduces the bit mismatch ratio by 42.34%.

In summary, FREE has a significant higher bit generation rate and great performance on the entropy, bit mismatch ratio and reconciliation overheads.

## 7 DISCUSSION

In this section, we discuss possible jamming attacks against FREE and possible scenario of FREE. Here we mainly analyze a potential situation where a proactive attacker tries to transmit high-power acoustic signals to overwhelm the transmissions between the two legitimate devices, so that the keys generated are highly correlated with the high-power acoustic signals. On the one hand, high-power ambient noises lead to high bit mismatch ratio. It has been validated by the result of the experiments conducted with different intensities of background noise in Section 6.5. As a result, when the attacker transmits high-power acoustic signals, it is jamming the key extraction process between the devices. When the acoustic channel communication is blocked, the two devices cannot extract the secret key. Thus, the attacker cannot get the key either. On the other hand, the attacker may confuse Alice and Bob about which one is the right signal for channel estimation. Thus, when Alice/Bob believes that she/he is estimating the channel state using the signal sent from Bob/Alice, she/he is actually working on the signal sent from the attacker instead. Then, the attacker can secretly relay and possibly alter the communication between Alice and Bob, who believe that they are directly communicating with the legitimate one. It is called **man-in-the-middle attack (MITM)**. However, MITM attack is out of the scope of the research on key extraction. To solve this problem, we may consider using authentication, tamper detection, or forensic analysis techniques during or after the process of secret key extraction in the future work.

We consider a possible scenario for the application of FREE. Smart home plays a more and more important role in our life. We consider using key extraction to realize access control. For example, a smart device (e.g., Amazon Echo) is equipped with a speaker and microphone, we can generate secret key between a mobile phone and the smart device to achieve device pairing, and only paired mobile phone can open the smart device.

## 8 RELATED WORK

Secret key extraction has been studied for many years [2, 11, 17, 21, 26, 37, 39, 42, 62, 77, 81]. In wireless network, the security of data transmission is guaranteed by the security protocols of upper layers. Physical layer security also needs to be guaranteed by encryption schemes. To achieve information theoretic-security at the physical layer, many existing works exploit the unpredictable and random characteristics of wireless channel to establish cryptographic key [3, 6, 8, 10, 18, 27, 33, 56, 61, 66, 68, 76, 78, 80]. Ahlswede et al. [1] and Maurer, et al. [35] discussed the key generation theoretically. Hershey et al. [19] first proposed the idea of using channel measurements to extract secret keys. Then, plenty of works exploit wireless channel measurements to extract secret keys. The channel measurements include arrival of angle [4], phase [50], and received signal strength [22, 28, 34, 60]. Then, CSI has been extensively exploited for key extraction that can achieve higher key generation rates. But CSI-based key extraction schemes are only compatible with specific wireless NIC models [24, 30, 31, 63, 64]. Some works use the sensor data to generate secret keys. Bichler et al. and Mayrhofer et al. exploited the acceleration data of shaking process for key generation and secure device pairing [5, 36]. MAGIK uses the dynamic geomagnetic field sensor readings to extract secret key [47]. But they cannot resist against the imitation attack. In addition, some works use audio signals to authenticate legitimate users [23, 29, 32, 40, 52, 54, 57, 65]. Schürmann, et al. used the similar ambient audio pattern to authenticate legitimate users and secure communication [52], but their method underperforms the key generation efficiency and cannot resist against imitation attack. Spartacus [57] uses audio to establish spontaneous interactions between mobile devices, but cannot guarantee their transmission confidentiality. Sound-Proof [23] uses ambient audio to validate the proximity to authenticate users, but cannot generate secret keys. Lu, et al. [32] used the acoustic channel estimation to extract the secret key. Comparing with it, we have brand new channel estimation methods MMSE and LMMSE to improve the channel estimation accuracy. Xu, et al. [67] proposed a similar scheme that uses inaudible acoustic signal for key agreement of IoT devices, but they did not present the design rationale and details of the key agreement. GeneWave [65] uses acoustic signal for authentication and key agreement. However, it relies on public key system to exchange the secret key. Some works [54] use ambient audio for secure pairing, but they must use Diffie-Hellman protocol to generate secret key.

## 9 CONCLUSION

In this article, we have studied how to achieve high key generation efficiency with commodity mobile devices. We have proposed FREE, which is a fast and robust key extraction mechanism that uses the randomness of inaudible acoustic channel to establish a secure wireless channel between two mobile devices. We have carefully studied and validated the feasibility of utilizing acoustic channel randomness for key extraction through theoretical analysis and extensive experiments. We also have implemented FREE on mobile devices, e.g., Nexus 7, MEIZU MX 6, and Xiaomi 3. The results of experiments show the high efficiency and satisfactory robustness of FREE. Compared with existing solutions for key establishment, FREE has two advantages: First, FREE has significantly higher key generation rate, which can generate a 512-bit cryptographic key in two seconds; Second, FREE can work on off-the-shelf mobile devices (e.g., smartphones, tablets).

## REFERENCES

[1] Rudolf Ahlswede and Imre Csiszár. 1993. Common randomness in information theory and cryptography. I. Secret sharing. *IEEE Trans. Inf. Theory* 39, 4 (1993), 1121–1132.

[2] Mohanad Alhasanat, Saud Althunibat, Khalid A. Darabkh, Abdullah Alhasanat, and Moath Alsafasfeh. 2020. A physical-layer key distribution mechanism for IoT networks. *Mobile Netw. Appl.* 25, 1 (2020), 173–178.

[3] Abhijit Ambekar, Mohamed Hassan, and Hans D. Schotten. 2012. Improving channel reciprocity for effective key management systems. In *Proceedings of the International Symposium on Signals, Systems, and Electronics (ISSSE'12)*. 1–4.

[4] Tomoyuki Aono, Keisuke Higuchi, Makoto Taromaru, Takashi Ohira, and Hideichi Sasaoka. 2005. Wireless secret key generation exploiting the reactance-domain scalar response of multipath fading channels: RSSI interleaving scheme. In *Proceedings of the IEEE European Microwave Association (EuMA'05)*. 173–176.

[5] Daniel Bichler, Guido Stromberg, Mario Huemer, and Manuel Löw. 2007. Key generation based on acceleration data of shaking processes. In *Proceedings of the ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp'07)*. 304–317.

[6] Matthieu Bloch and Joao Barros. 2011. *Physical-layer Security: From Information Theory to Security Engineering*. Cambridge University Press.

[7] A. Robert Calderbank, G. David Forney, and Alexander Vardy. 1999. Minimal tail-biting trellises: The Golay code and more. *IEEE Trans. Inf. Theory* 45, 5 (1999), 1435–1455.

[8] Mingsheng Cao, Dajiang Chen, Zhongye Yuan, Zhiguang Qin, and Chunwei Lou. 2018. A lightweight key distribution scheme for secure D2D communication. In *Proceedings of the International Conference on Selected Topics in Mobile and Wireless Networking (MoWNeT'18)*. 1–8.

[9] Chunyi Chen and Huamin Yang. 2018. Shared secret key generation from signal fading in a turbulent optical wireless channel using common-transverse-spatial-mode coupling. *Opt. Express* 26, 13 (2018), 16422–16441.

[10] Dajiang Chen, Zhen Qin, Xufei Mao, Panlong Yang, Zhiguang Qin, and Ruijin Wang. 2013. Smokegrenade: An efficient key generation protocol with artificial interference. *IEEE Trans. Inf. Forens. Secur.* 8, 11 (2013), 1731–1745.

[11] Yuan Ding, Junqing Zhang, and Vincent F. Fusco. 2016. Retrodirective-assisted secure wireless key establishment. *IEEE Trans. Commun.* 65, 1 (2016), 320–334.

[12] Yevgeniy Dodis, Leonid Reyzin, and Adam Smith. 2004. Fuzzy extractors: How to generate strong keys from biometrics and other noisy data. In *Proceedings of the International Conference on the Theory and Applications of Cryptographic Techniques*. Springer, 523–540.

[13] Matthew Edman, Aggelos Kiayias, Qiang Tang, and Bülent Yener. 2016. On the security of key extraction from measuring physical quantities. *IEEE Trans. Inf. Forens. Secur.* 11, 8 (2016), 1796–1806.

[14] Andrea Goldsmith. 2005. *Wireless Communications*. Cambridge University Press.

[15] René Guillaume, Fredrik Winzer, Andreas Czylwik, Christian T. Zenger, and Christof Paar. 2015. Bringing PHY-based key generation into the field: An evaluation for practical scenarios. In *Proceedings of the IEEE 82nd Vehicular Technology Conference (VTC'15)*. 1–5.

[16] Daniel Halperin, Wenjun Hu, Anmol Sheth, and David Wetherall. 2011. Tool release: Gathering 802.11 n traces with channel state information. *ACM SIGCOMM Comput. Commun. Rev.* 41, 1 (2011), 53–53.

[17] Jehad M. Hamamreh, Haji M. Furqan, and Huseyin Arslan. 2018. Classifications and applications of physical layer security techniques for confidentiality: A comprehensive survey. *IEEE Commun. Surv. Tutor.* 21, 2 (2018), 1773–1828.

[18] Xiaofan He, Huaiyu Dai, Yufan Huang, Dong Wang, Wenbo Shen, and Peng Ning. 2014. The security of link signature: A view from channel models. In *Proceedings of the IEEE Conference on Communications and Network Security (CNS'14)*. 103–108.

[19] John E. Hershey, Amer A. Hassan, and Rao Yarlagadda. 1995. Unconventional cryptographic keying variable management. *IEEE Trans. Commun.* 43, 1 (1995), 3–6.

[20] Christopher Huth, René Guillaume, Paul Duplys, Kumaragurubaran Velmurugan, and Tim Güneysu. 2016. On the energy cost of channel based key agreement. In *Proceedings of the 6th International Workshop on Trustworthy Embedded Devices (TrustED'16)*. 31–41.

[21] Christopher Huth, René Guillaume, Thomas Strohm, Paul Duplys, Irin Ann Samuel, and Tim Güneysu. 2016. Information reconciliation schemes in physical-layer security: A survey. *Comput. Netw.* 109 (2016), 84–104.

[22] Suman Jana, Sriram Nandha Premnath, Mike Clark, Sneha K. Kasera, Neal Patwari, and Srikanth V. Krishnamurthy. 2009. On the effectiveness of secret key extraction from wireless signal strength in real environments. In *Proceedings of the ACM Annual International Conference on Mobile Computing and Networking (MobiCom'09)*. 321–332.

[23] Nikolaos Karapanos, Claudio Marforio, Claudio Soriente, and Srdjan Capkun. 2015. Sound-Proof: Usable two-factor authentication based on ambient sound. In *Proceedings of the USENIX Security Conference*. 483–498.

[24] Havish Koorapaty, Amer A. Hassan, and Sandeep Chennakeshu. 2000. Secure information transmission for mobile radio. *IEEE Commun. Lett.* 4, 2 (2000), 52–55.

[25] Konrad-Felix Krentz and Gerhard Wunder. 2015. 6doku: Towards secure over-the-air preloading of 6lowpan nodes using phy key generation. In *Proceedings of European Conference on Smart Objects, Systems and Technologies (Smart SysTech'15)*. 1–11.

[26] Guyue Li, Chen Sun, Junqing Zhang, Eduard Jorswieck, Bin Xiao, and Aiqun Hu. 2019. Physical layer key generation in 5G and beyond wireless communications: Challenges and opportunities. *Entropy* 21, 5 (2019), 1–16.

[27] Xin Li, Minmei Wang, Huazhe Wang, Ye Yu, and Chen Qian. 2019. Toward secure and efficient communication for the internet of things. *IEEE/ACM Trans. Netw.* 27, 2 (2019), 621–634.
[28] Zi Li, Qingqi Pei, Ian Markwood, Yao Liu, and Haojin Zhu. 2018. Secret key establishment via RSS trajectory matching between wearable devices. *IEEE Trans. Inf. Forens. Secur.* 13, 3 (2018), 802–817.
[29] Xiaohui Liang, Tianlong Yun, Ronald Peterson, and David Kotz. 2017. LightTouch: Securely connecting wearables to ambient displays with user intent. In *Proceedings of the IEEE International Conference on Computer Communications (INFOCOM'17)*. 1–9.
[30] Hongbo Liu, Yang Wang, Jie Yang, and Yingying Chen. 2013. Fast and practical secret key extraction by exploiting channel response. In *Proceedings of the IEEE International Conference on Computer Communications (INFOCOM'13)*. 3048–3056.
[31] Yanpei Liu, Stark C. Draper, and Akbar M. Sayeed. 2012. Exploiting channel diversity in secret key generation from multipath fading randomness. *IEEE Trans. Inf. Forens. Secur.* 7, 5 (2012), 1484–1497.
[32] Youjing Lu, Fan Wu, Shaojie Tang, Linghe Kong, and Guihai Chen. 2019. FREE: A fast and robust key extraction mechanism via inaudible acoustic signal. In *Proceedings of the ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc'19)*. 311–320.
[33] Suhas Mathur, Alex Reznik, Chunxuan Ye, Rajat Mukherjee, Akbar Rahman, Yogendra Shah, Wade Trappe, and Narayan Mandayam. 2010. Exploiting the physical layer for enhanced security [security and privacy in emerging wireless networks]. *IEEE Wireless Commun.* 17, 5 (2010).
[34] Suhas Mathur, Wade Trappe, Narayan Mandayam, Chunxuan Ye, and Alex Reznik. 2008. Radio-telepathy: Extracting a secret key from an unauthenticated wireless channel. In *Proceedings of the ACM Annual International Conference on Mobile Computing and Networking (MobiCom'08)*. 128–139.
[35] Ueli M. Maurer. 1993. Secret key agreement by public discussion from common information. *IEEE Trans. Inf. Theory* 39, 3 (1993), 733–742.
[36] Rene Mayrhofer and Hans Gellersen. 2009. Shake well before use: Intuitive and secure pairing of mobile devices. *IEEE Trans. Mobile Comput.* 8, 6 (2009), 792–806.
[37] Reem Melki, Hassan N. Noura, Mohammad M. Mansour, and Ali Chehab. 2019. A survey on OFDM physical layer security. *Phys. Commun.* 32 (2019), 1–30.
[38] Wilson S. Melo, Raphael Machado, and Luiz F. R. C. Carmo. 2018. Using physical context-based authentication against external attacks: Models and protocols. *Secur. Commun. Netw.* (2018).
[39] Markus Miettinen and N. Asokan. 2018. Ad-hoc key agreement: A brief history and the challenges ahead. *Comput. Commun.* 131 (2018), 32–34.
[40] Markus Miettinen, Thien Duc Nguyen, N. Asokan, Ahmad-Reza Sadeghi, Razvan Deaconescu, Costin Carabas, Iulia Manda, William Enck, Mihai Chiroiu, Ninghui Li, et al. 2018. Revisiting context-based pairing in IoT. In *Proceedings of the 55th Design Automation Conference (DAC'18)*.
[41] Kemedi Moara-Nkwe, Qi Shi, Gyu Myoung Lee, and Mahmoud Hashem Eiza. 2018. A novel physical layer secure key generation and refreshment scheme for wireless sensor networks. *IEEE Access* 6 (2018), 11374–11387.
[42] Amitav Mukherjee, S. Ali A. Fakoorian, Jing Huang, and A. Lee Swindlehurst. 2014. Principles of physical layer security in multiuser wireless networks: A survey. *IEEE Commun. Surv. Tutor.* 16, 3 (2014), 1550–1573.
[43] Minseok Noh, Yusung Lee, and Hyuncheol Park. 2006. Low complexity LMMSE channel estimation for OFDM. *IEE Proc. Commun.* 153, 5 (2006), 645–650.
[44] Alan V. Oppenheim. 1999. *Discrete-time Signal Processing*. Pearson Education India.
[45] Sriram Nandha Premnath, Suman Jana, Jessica Croft, Prarthana Lakshmane Gowda, Mike Clark, Sneha Kumar Kasera, Neal Patwari, and Srikanth V. Krishnamurthy. 2013. Secret key extraction from wireless signal strength in real environments. *IEEE Trans. Mobile Comput.* 12, 5 (2013), 917–930.
[46] Markku Pukkila. 2000. Channel estimation modeling. *Nokia Research Center* 17 (2000), 66.
[47] Fudong Qiu, Zhengxian He, Linghe Kong, and Fan Wu. 2017. MAGIK: An efficient key extraction mechanism based on dynamic geomagnetic field. In *Proceedings of the IEEE International Conference on Computer Communications (INFOCOM'17)*. 1–9.
[48] Theodore S. Rappaport et al. 1996. *Wireless Communications: Principles and Practice*. Vol. 2. Prentice Hall, Hoboken, NJ.
[49] Jean-Fransico Raymond and Anton Stiglic. 2000. Security issues in the Diffie-Hellman key agreement protocol. *IEEE Trans. Inf. Theory* 22 (2000), 1–17.
[50] Akbar Sayeed and Adrian Perrig. 2008. Secure wireless communications: Secret keys through multipath. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP'08)*. 3013–3016.
[51] Louis L. Scharf and Cédric Demeure. 1991. *Statistical Signal Processing: Detection, Estimation, and Time series Analysis*. Vol. 63. Addison-Wesley Reading, MA.

[52] Dominik Schürmann and Stephan Sigg. 2013. Secure communication based on ambient audio. *IEEE Trans. Mobile Comput.* 12, 2 (2013), 358–370.
[53] Chen Shen, Hao Li, Gokhan Sahin, Hyeong-Ah Choi, and Yogendra Shah. 2018. Golay code based bit mismatch mitigation for wireless channel impulse response based secrecy generation. *IEEE Access* 7 (2018), 2999–3007.
[54] Stephan Sigg, Yusheng Ji, Ngu Nguyen, and An Huynh. 2012. AdhocPairing: Spontaneous audio based secure device pairing for Android mobile devices. In *Proceedings of the International Workshop on Security and Privacy in Spontaneous Interaction and Mobile Phone Use (IWSSI/SPMU'12)*. 1–6.
[55] Ankit Soni, Raksha Upadhyay, and Abhay Kumar. 2019. Wireless physical layer key generation with improved bit disagreement for the internet of things using moving window averaging. *Phys. Commun.* 33 (2019), 249–258.
[56] Sumei Sun, Yongdong Wu, Boon Shyang Lim, and Hieu Duy Nguyen. 2017. A high bit-rate shared key generator with time-frequency features of wireless channels. In *Proceedings of the IEEE Global Telecommunications (GLOBECOM'17)*. 1–6.
[57] Zheng Sun, Aveek Purohit, Raja Bose, and Pei Zhang. 2013. Spartacus: spatially-aware interaction for mobile devices through energy-efficient audio sensing. In *Proceedings of the ACM Annual International Conference on Mobile Systems, Applications, and Services (MobiSys'13)*. 263–276.
[58] David Tse and Pramod Viswanath. 2004. *Fundamentals of Wireless Communication*. Cambridge University Press.
[59] Qian Wang, Hai Su, Kui Ren, and Kwangjo Kim. 2011. Fast and scalable secret key generation exploiting channel phase randomness in wireless networks. In *Proceedings of the IEEE International Conference on Computer Communications (INFOCOM'11)*. 1422–1430.
[60] Qian Wang, Kaihe Xu, and Kui Ren. 2012. Cooperative secret key generation from phase estimation in narrowband fading channels. *IEEE J. Select. Areas Commun.* 30, 9 (2012), 1666–1674.
[61] Xiaoxue Wang, Yanzhao Hou, Xueqing Huang, Dongru Li, Xiaofeng Tao, and Jin Xu. 2018. Security analysis of key extraction from physical measurements with multiple adversaries. In *Proceedings of the IEEE International Conference on Communications Workshops (ICC Workshops'18)*. 1–6.
[62] Yongpeng Wu, Ashish Khisti, Chengshan Xiao, Giuseppe Caire, Kai-Kit Wong, and Xiqi Gao. 2018. A survey of physical layer security techniques for 5G wireless networks and challenges ahead. *IEEE J. Select. Areas Commun.* 36, 4 (2018), 679–695.
[63] Wei Xi, Xiang-Yang Li, Chen Qian, Jinsong Han, Shaojie Tang, Jizhong Zhao, and Kun Zhao. 2014. KEEP: Fast secret key extraction protocol for D2D communication. In *Proceedings of the IEEE/ACM International Symposium on Quality of Service (IWQoS'14)*. 350–359.
[64] Wei Xi, Chen Qian, Jinsong Han, Kun Zhao, Sheng Zhong, Xiang-Yang Li, and Jizhong Zhao. 2016. Instant and robust authentication and key agreement among mobile devices. In *Proceedings of the ACM SIGSAC Conference on Computer and Communications Security (CCS'16)*. 616–627.
[65] Pengjin Xie, Jingchao Feng, Zhichao Cao, and Jiliang Wang. 2017. GeneWave: Fast authentication and key agreement on commodity mobile devices. In *Proceedings of the IEEE International Conference on Network Protocols (ICNP'17)*. 1688–1700.
[66] Weitao Xu, Sanjay Jha, and Wen Hu. 2018. Exploring the feasibility of physical layer key generation for LoRaWAN. In *Proceedings of the IEEE International Conference on Trust, Security and Privacy in Computing and Communications/IEEE International Conference On Big Data Science And Engineering (TrustCom/BigDataSE'18)*. 231–236.
[67] Weitao Xu, Zhenjiang Li, Wanli Xue, Xiaotong Yu, Jia Wang, Chengwen Luo, Wei Li, and Albert Y. Zomaya. 2020. Inaudible acoustic signal based key agreement system for IoT devices. In *Proceedings of the 18th Conference on Embedded Networked Sensor Systems (SenSys'20)*. 689–690.
[68] Nan Yang, Lifeng Wang, Giovanni Geraci, Maged Elkashlan, Jinhong Yuan, and Marco Di Renzo. 2015. Safeguarding 5G wireless communication networks using physical layer security. *IEEE Commun. Mag.* 53, 4 (2015), 20–27.
[69] Mike Yuliana et al. 2019. A simple secret key generation by using a combination of pre-processing method with a multilevel quantization. *Entropy* 21, 2 (2019), 192.
[70] Sangki Yun, Yi-Chao Chen, Huihuang Zheng, Lili Qiu, and Wenguang Mao. 2017. Strata: Fine-grained acoustic-based device-free tracking. In *Proceedings of the ACM Annual International Conference on Mobile Systems, Applications, and Services (MobiSys'17)*. 15–28.
[71] Christian Zenger, Jan Zimmer, and Christof Paar. 2015. Security analysis of quantization schemes for channel-based key extraction. In *Proceedings of the 12th EAI International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services (MobiQuitous'15)*. 267–272.
[72] Christian Zenger, Jan Zimmer, Jan-Felix Posielek, and Christof Paar. 2015. On-line entropy estimation for secure information reconciliation. In *Proceedings of the 12th EAI International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services (MobiQuitous'15)*. 254–259.
[73] Christian T. Zenger, Mario Pietersz, Jan Zimmer, Jan-Felix Posielek, Thorben Lenze, and Christof Paar. 2016. Authenticated key establishment for low-resource devices exploiting correlated random channels. *Comput. Netw.* 109 (2016), 105–123.

[74] Christian T. Zenger, Jan Zimmer, Mario Pietersz, Jan-Felix Posielek, and Christof Paar. 2015. Exploiting the physical environment for securing the internet of things. In *Proceedings of the 2015 New Security Paradigms Workshop*. 44–58.
[75] Junqing Zhang, Ming Ding, David López-Pérez, Alan Marshall, and Lajos Hanzo. 2019. Design of an efficient OFDMA-based multi-user key generation protocol. *IEEE Trans. Vehic. Technol.* 68, 9 (2019), 8842–8852.
[76] Junqing Zhang, Trung Q. Duong, Alan Marshall, and Roger Woods. 2016. Key generation from wireless channels: A review. *IEEE Access* 4 (2016), 614–626.
[77] Junqing Zhang, Trung Q. Duong, Roger Woods, and Alan Marshall. 2017. Securing wireless communications of the internet of things from the physical layer, an overview. *Entropy* 19, 8 (2017), 1–16.
[78] Junxing Zhang, Sneha K. Kasera, and Neal Patwari. 2010. Mobility assisted secret key generation using wireless link signatures. In *Proceedings of the IEEE International Conference on Computer Communications (INFOCOM'10)*. 1–5.
[79] Junqing Zhang, Roger Woods, Trung Q. Duong, Alan Marshall, Yuan Ding, Yi Huang, and Qian Xu. 2016. Experimental study on key generation for physical layer security in wireless communications. *IEEE Access* 4 (2016), 4464–4477.
[80] Xiaojun Zhu, Fengyuan Xu, Edmund Novak, Chiu C. Tan, Qun Li, and Guihai Chen. 2013. Extracting secret key from wireless link dynamics in vehicular environments. In *Proceedings of the IEEE International Conference on Computer Communications (INFOCOM'13)*. 2283–2291.
[81] Yulong Zou, Jia Zhu, Xianbin Wang, and Lajos Hanzo. 2016. A survey on wireless security: Technical challenges, recent advances, and future trends. *IEEE* 104, 9 (2016), 1727–1765.

Received May 2020; revised July 2021; accepted August 2021