TouchKey: Touch to Generate Symmetric Keys by Skin Electric Potentials Induced by Powerline Radiation

YUCHEN MIAO, Zhejiang University, China
CHAOJIE GU, Zhejiang University, China
ZHENYU YAN, The Chinese University of Hong Kong, China
SZE YIU CHAU, The Chinese University of Hong Kong, China
RUI TAN, Nanyang Technological University, Singapore
QI LIN, Uppsala University, Sweden
WEN HU, University of New South Wales, Australia
SHIBO HE, Zhejiang University, China
JIMING CHEN, Zhejiang University, China

Secure device pairing is important to wearables. Existing solutions either degrade usability due to the need of specific actions like shaking, or they lack universality due to the need of dedicated hardware like electrocardiogram sensors. This paper proposes TouchKey, a symmetric key generation scheme that exploits the skin electric potential (SEP) induced by powerline electromagnetic radiation. The SEP is ubiquitously accessible indoors with analog-to-digital converters widely available on Internet of Things devices. Our measurements show that the SEP has high randomness and the SEPs measured at two close locations on the same human body are similar. Extensive experiments show that TouchKey achieves a high key generation rate of 345 bit/s and an average success rate of 99.29%. Under a range of adversary models including active and passive attacks, TouchKey shows a low false acceptance rate of 0.86%, which outperforms existing solutions. Besides, the overall execution time and energy usage are 0.44 s and 2.716 mJ, which make it suitable for resource-constrained devices.

CCS Concepts: • **Human-centered computing → Ubiquitous and mobile computing**; • **Computer systems organization → Embedded and cyber-physical systems**; • **Networks →** *Mobile and wireless security*.

Additional Key Words and Phrases: Key generation, wearables, induced body electric potential

1 INTRODUCTION

With the advances of wireless communication and embedded computing technologies, wearables have undergone rapid growth in quantity and remarkable development in functionality. The global market size of wearable technologies was valued at US$ 54.8 billion in 2020 and is projected to reach US$ 184.4 billion by 2031 [37]. The wearables have enabled a broad spectrum of applications [15], including fitness tracking, health monitoring, and medical diagnosis. In particular, device-to-device communication is a fundamental system function of modern wearables, enabling data transmission, fusion, and cooperation. Before performing device-to-device communication, a necessary step is to securely pair the devices to protect the data communicated afterwards.

To establish a secure wireless communication channel, the devices can adopt public key cryptography approaches such as Diffie–Hellman (DH) key exchange [8]. Since an unauthenticated key exchange mechanism is vulnerable to the monster-in-the-middle (MITM) attack, it typically needs a certificate authority (CA) to provide a basis of trust. However, it is difficult to implement the compute-intensive public key cryptography schemes on resource-constrained wearables. To counteract the MITM attack, a common practice is to grant users/devices access to an external channel through which they can information-theoretically authenticate short values (i.e.,

Authors' addresses: Yuchen Miao, Zhejiang University, China, miaoyc@zju.edu.cn; Chaojie Gu, Zhejiang University, China, gucj@zju.edu.cn; Zhenyu Yan, The Chinese University of Hong Kong, China, zyyan@cuhk.edu.hk; Sze Yiu Chau, The Chinese University of Hong Kong, China, sychau@ie.cuhk.edu.hk; Rui Tan, Nanyang Technological University, Singapore, tanrui@ntu.edu.sg; Qi Lin, Uppsala University, Sweden, qi.lin@it.uu.se; Wen Hu, University of New South Wales, Australia, wen.hu@unsw.edu.au; Shibo He, Zhejiang University, China, s18he@zju.edu.cn; Jiming Chen, Zhejiang University, China, cjm@zju.edu.cn.

out-of-band authenticated key exchange [34]). For example, the user is asked to enter a pin code or scan a quick-response (QR) code [28]. However, these methods cannot be applied to wearables that lack user interfaces like earbuds.

Recently, the touch-to-access scheme has emerged as a promising solution [30, 48, 52]. It generates symmetric keys among the devices that aim to pair or authenticate with each other by sensing a common body signal. This scheme exploits the entropy that lies in the random signal source. If the keys generated independently by two devices are the same, the two devices can be considered to have physical contact with the same person. The same-body contact can be viewed as a practical way of establishing mutual trust among the wearables, because the external adversarial devices that attempt to obtain physical access to the user's human body can be easily discerned by the user. Existing schemes exploit biometric characteristics like heartbeat [30] and electromyography [52]. However, they require dedicated sensors. Other motion signal-based schemes ask users to do specific actions like walking [48], shaking [42], or pressing a button [26]. These schemes require explicit user involvement and are vulnerable to video analytics [5].

The aforementioned approaches have different usability limitations and security vulnerabilities. To achieve ubiquitous and secure pairing on wearables, this paper proposes TouchKey, a touch-to-access scheme by exploiting the skin electric potential (SEP) induced by the powerline electromagnetic radiation (EMR). The powerline electromagnetic emanations from the power networks (e.g., cables and sockets) and electric appliances are ubiquitous indoors. Since the human body spontaneously transforms the emanations to SEP due to the body antenna effect [6], the SEP can be observed without the mandatory requirement of specific actions. From our measurements, we find that: (1) the SEPs share a high similarity when measured by two sensors that are on the same human body and close to each other; (2) the SEP has sufficient randomness for extracting a secure cryptographic key. Thus, TouchKey takes the SEP as the random signal source for key generation.

Generating keys based on the SEP requires addressing several technical challenges that arise from the characteristics of the SEP. First, the strength of the SEP varies dramatically in different environments due to the complicated distribution of the powerlines. We design a signal preprocessing scheme for SEP extraction in diverse environments. Second, from extensive measurements, we observe that varied device placement on the human body, hardware imperfections cause discrepancies in SEP data. We design a quantization method with guard band to discard potential mismatched samples and preserve potential matched samples. Third, the bit mismatch cannot be eliminated by signal preprocessing and quantization. We design a novel reconciliation approach by combining an effective index-based method and a classic challenge-and-reply protocol to achieve mutual confirmation on the common key.

We have implemented prototypes of our design on commercial off-the-shelf (COTS) wireless devices. We conduct extensive experiments in various practical scenarios to evaluate the performance and robustness of TouchKey. Besides, we design both passive attacks and active attacks to evaluate the security of TouchKey. And a new attack model is first proposed to infer the correct key by utilizing the information leaked during key generation. The results show that TouchKey achieves a high key generation rate of 345 bit/s. Comprehensive adversary models, including imitation, replay, injection, and inference, are studied. Under these adversary models, TouchKey shows a low false acceptance rate of 0.86%. TouchKey outperforms existing schemes in terms of key generation rate and false acceptance rate. Besides, TouchKey takes 0.44 s and 2.716 mJ to generate a key, which makes it suitable for resource-constrained devices. TouchKey can be used in most indoor environments and outdoor environments with powerline cables around that can induce electric potential on human body [51].

The contributions of this work can be summarized in the following aspects:

- To the best of our knowledge, this paper is the first work that uses on-body SEPs induced by powerline EMR to generate keys. We show that the SEP has sufficient randomness for extracting robust cryptographic keys.

- We propose an efficient key generation scheme for wearables based on SEPs, where we apply a power spectral density-based signal preprocessing method, an error-tolerant quantization method, and a lightweight reconciliation method. The proposed scheme can be implemented on low-cost and resource-constrained devices.
- We propose four types of attack models, including an active attack model targeting the spectral characteristics of the signal source and a first proposed inference attack model targeting information leakage during key generation.
- We implement the proposed scheme on real devices and conduct extensive experiments under different settings to evaluate its usability, efficiency, and security. The results show that the proposed approach outperforms the state-of-art solutions in key generation speed, robustness against attacks, and total execution time.

2 RELATED WORK

Body Signal-based Key Generation System. Body signal key generation systems can be broadly categorized into biometric-based systems and human motion-based systems. The unique and random nature of biometric characteristics makes them hard to be forged. Previous studies have found that inter-pulse interval (IPI) extracted from heartbeat is highly random [35] and electromyography (EMG) is a quasi-random process [32]. Thus, existing works [3, 30, 46, 52] use these signals to generate keys. However, biometric-based systems suffer from a common deficiency of the low key generation rate – many systems take more than 30 s for a 128-bit key. Moreover, they require dedicated sensors to collect data.

Differently, human motion-based systems achieve a high bit generation rate. Xu *et al.* [48] and Lin *et al.* [29] exploit the gait signal to generate keys. Shen *et al.* [42] exploit the shaking signal to generate the same key between two persons during handshaking. Li *et al.* [26] utilize the timestamps of the user doing specific actions (like pressing a button) with random pauses. However, motion-based systems require explicit user involvement and are vulnerable to video attacks that film and analyze the motion to forge a key [5]. Some systems use other on-body signals to generate keys, such as body channel characteristics (e.g., frequency-dependent signal attenuation [40]).

Our system is the first to exploit the SEP induced by powerline EMR for key generation. The most related work to ours is Harness [20], which leverages on-body signals induced by a 2.4GHz radio communication signal. However, Harness only works when the two pairing devices have the same RF transceiver module. In contrast, TouchKey only needs pervasive ADC. Additionally, Harness noticeably degrades the original communication performance of the RF transceiver [20].

Wireless Signal-based Key Generation System. Lots of efforts have been devoted to using wireless signals for key generation. Systems utilizing the radio communication signal rely on its spatiotemporal variation nature, e.g., WiFi [53], LoRa [47], and ZigBee [2]. However, they require pairing devices to support the same transmission protocol. Many of them [2, 38, 53] require special antennas.

Various ambient environmental signals are also exploited to generate keys, like audio [31], noise and luminosity [33], and temperature & humidity [43], with dedicated sensors. Han *et al.* [14] and Farrukh *et al.* [10] leverage inter-event timings collected by different sensing modality to generate keys. Lee *et al.* [24] use the EMR noise extracted from sockets as the key material. However, pairing devices need to be plugged into sockets on the same circuit breaker and a special data collection device is required. Key generation systems using ambient environmental signals rely on spatial proximity. Thus, they are vulnerable to co-located adversaries.

SEP-based Application. The SEP has been used for various applications, including synchronization [27, 50], localization [11], gesture recognition [6, 7], and authentication [51]. Yan *et al.* [51] exploit SEPs to achieve device authentication by a similarity-based detection algorithm. However, its performance degrades significantly even

when legitimate devices are at a reasonable distance, e.g., from the palm to the elbow. However, it takes 5 s to collect data. Moreover, key generation is not in the scope of their work. As the first attempt to use powerline SEPs to generate keys, our system can be used for authentication and pairing.

3 PRELIMINARY

3.1 Measurement Setup

Fig. 1. The measurement device consists of a TI CC2650 Launchpad, a DC charger module, and a lithium-ion battery.

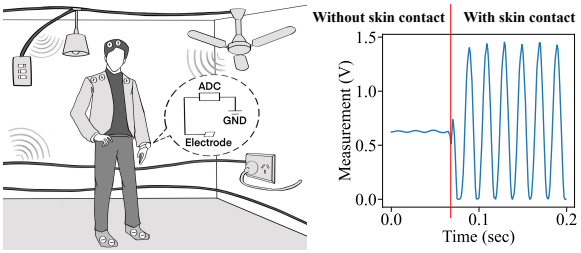


Fig. 2. ADC captures SEP via body antenna effect.

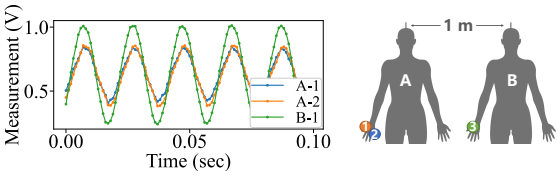
As shown in Fig. 1, we prototype the measurement device using TI CC2650 Launchpad [16]. The roles of the measurement devices in our experiments are *collector* and *relay*. The collector samples SEP at a rate of 1 kHz. A conductive wire is connected to its analog input port as an electrode for physical contact with the wearer's skin. The collector is powered by a lithium-ion polymer battery, while the relay is connected to the computer using a USB cable. The collector transmits sampled data to the relay over wireless. The relay forwards the received data to the computer for further analysis. Before data collection, collectors synchronize their clocks for data alignment.

Ethics Consideration. All tests in this paper only collect SEP data from human subject skin passively. They do not actively inject or induce any extra signal. In addition, all devices are commercial-off-the-shelf devices working on a 3.7-volt lithium-ion battery with no risk to the human subject. This research has been approved by the research ethic committee of the authors' institutes where the experiments are carried out.

3.2 Powerline EMR & Body Antenna Effect

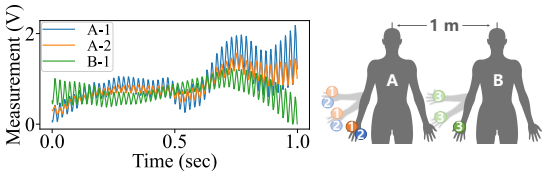
First, we illustrate the body antenna effect. From electrostatics, the human body can be viewed as a low-impedance conductor [36, 51]. When exposed to an electric field, it builds up a specific surface charge distribution to reach an electrostatic equilibrium. The electric fields from the powerlines and electric appliances form a composite time-varying electric field that oscillates at the mains frequency. In accordance, the body surface charge distribution varies over time at the mains frequency. As shown in Fig. 2, by physical contact between the skin and the analog-to-digital converter (ADC), the latter can capture the charge distribution variation in the form of potential difference between the input pin and the floating ground (GND). The measurement trace of the ADC before and after the physical contact with a human body is established. Without physical contact, the ADC captures the EMR in the air with weak amplitude and an unclear frequency of around 50 Hz (i.e., the mains frequency in our region). With physical contact, the ADC captures SEP with significant amplitude and salient 50 Hz frequency.

Second, we illustrate the characteristics of SEP. ADC measures the difference between the input pin and the GND. When the input pin is connected to user's skin via a conductive wire, it senses the potential on the human body's skin. Note that the human body can be considered as an uncharged object with an equipotential surface. This means that the potentials captured by the input pin at any position on the same body are the same at any



(a) Raw SEPs (b) Sensor placements

Fig. 3. SEPs measured by three sensors on two persons who keep still.



(a) Raw SEPs (b) Sensor placements

Fig. 4. SEPs measured by three sensors on two persons who perform random hand movements.

time. Since wearables are battery-powered, their GND pins are not connected to the earth (i.e., zero potential) and thus capture location-dependent potentials. Based on the above analysis, the readings of two ADCs will only be the same when they are placed on the same body at close locations. Fig. 3(a) shows the SEP traces collected on human subjects A and B respectively when both of them keep still 1 m apart. Person A holds two sensors in the right palm and person B holds another sensor in the right palm as Fig. 3(b) shows. From the data, the SEPs on the same person are more similar than signals from different persons. This is due to two affecting factors: a) people have unique body parameters like shape, gender, and posture, which affect the body surface charge distribution; and b) due to the complicated EMR sources distribution in the space, ADCs apart capture different potentials on their GND pins. Thus, the SEP traces measured on different human subjects are distinct. Fig. 4(a) shows the SEPs measured on person A and person B 1 m apart respectively when A performs random movements and B mimics A. As Fig. 4(b) shows, they hold the sensors in the same way as in the previous experiment. From the data, the amplitude of the alternating current (AC) component and the waveform of the direct current (DC) component change over time. However, the changes on the same person follow similar trends, while the changes on different people differ a lot.

4 TouchKey's MODEL AND THREAT MODEL

4.1 System Model

TouchKey aims to establish a secure communication channel between COTS wireless wearables by generating symmetric keys in an untrusted public network. We assume one legitimate wearable Alice, who possesses a token that shows the user's identity, wants to pair with a wearable Bob, who does not have a pre-shared secret with Alice. When performing TouchKey, Alice and Bob need to have physical contact with the legitimate user to capture the similar SEPs. After the keys generated by Alice and Bob respectively are agreed to be perfectly the same, Bob is considered to be legitimate and they successfully pair with each other. The token and other sensitive information are allowed to be passed from Alice to Bob through the established secure channel. Therefore, TouchKey provides an unobtrusive and spontaneous authentication and pairing approach for users. For example, a user has a smartwatch (Alice) and a smartphone (Bob) and both of them have installed the TouchKey application. After working out in the gym, the user wants to update some privacy-sensitive data like heart rate to Bob. Then, Alice and Bob launch TouchKey, and they generate symmetric keys by exploiting measured SEPs respectively. Once the keys are confirmed to be identical, Bob authenticates his identity to Alice and they use this key to establish a secure communication channel.

We assume a generic system model that is applicable to COTS wearables equipped with low-cost ADC, driven by resource-constrained processors and lacking common user interfaces. TouchKey can be implemented on COTS wireless devices without the requirement of adding extra hardware components, since the SOCs on most COTS devices including CC2650 LaunchPad have multiple ADCs available. In addition, it is possible to reuse ADCs in different applications using an analog multiplexer. Note that as our system is applicable to wearables attached on the body surface, the application for implantable devices is not covered.

4.2 Threat Model

To validate the security of our scheme, we assume an adversary aims to steal the user's private information during the key generation process. The attacker is capable of eavesdropping and modifying any message exchanged by legitimate devices on the public network before keys are generated for symmetric encryption. Furthermore, the attacker has full knowledge of the key generation mechanism including the predefined parameters and the hardware of the legitimate device. Thus, the attacker can implement the same clock synchronization and information reconciliation process as the legitimate devices. We assume the attacker cannot directly place sensors on the user, as this would soon be noticed by the user. We do not consider the denial of service (DoS) attack because it has been extensively investigated [44, 49] and is not specific to TouchKey. We also do not consider the attacks on clock synchronization, because they cause effects similar to DoS. In particular, we consider the following four attacks.

Imitation Attack: An imitation attacker is capable of observing the user's motion and mimicking it to generate a key based on the attacker's own SEP. This is the most common type of attack which can happen in any condition. To address this attack, we need to ensure that the generated keys of different users are distinct.

Injection Attack: An injection attacker can introduce strong electromagnetic noises to force legitimate devices to agree on the key as the attacker desires. This attack can be simulated by running various electric appliances, which will not be easily sensed by legitimate users on public occasions.

Replay Attack: A replay attacker may have access to the historical data of legitimate devices and the historical keys. Then, the attacker tries to pair with legitimate devices by two means of directly using previous keys or generating a new key by reusing the historical legitimate data. Freshness is needed to defeat this attack.

Inference Attack: An inference attacker can fully utilize the leaked partial information during the reconciliation process and try to infer some details of the raw data. Then, the attacker tries to guess the converted key with the help of inferred details.

5 TouchKey DESIGN

5.1 Design Overview

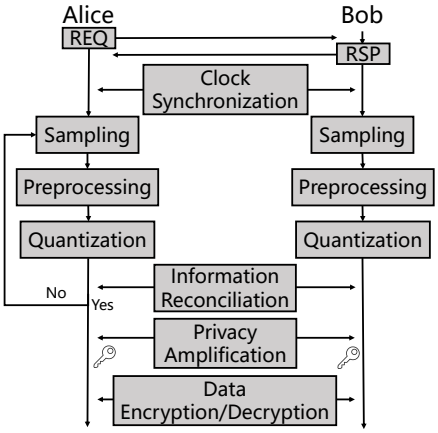


Fig. 5. Overview of TouchKey.

TouchKey is designed as a pairing system to establish a secure wireless communication channel between wearables. It exploits the SEP to generate keys. To extract SEPs in diverse environments, we design a power spectral density (PSD)-based filtering scheme to strike a balance between the usability and the randomness of the SEP data (see

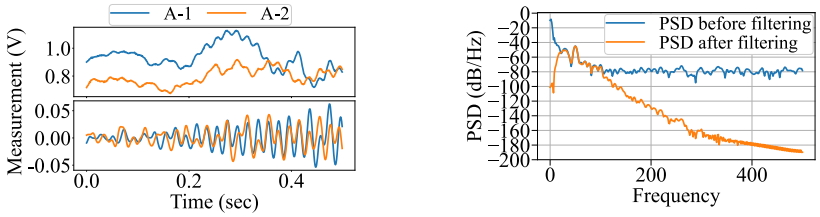
§5.2). To tackle discrepancies in SEP data, we design a guard band-based quantization method so that the potential mismatched samples which fall in the guard band are discarded, while the potential matched samples are preserved (see §5.3). To simultaneously resolve the bit mismatch and save communication bandwidth, we design a novel reconciliation approach by combining an effective index-based method [48] and a classic challenge-and-reply protocol to achieve two way confirmation on the common key (see §5.4).

Fig. 5 shows the workflow of TouchKey. Suppose Alice wants to send private data to Bob, Alice initiates the handshaking by broadcasting a REQ request. After receiving REQ, Bob responds with RSP. Then, they synchronize their clocks and start to record SEPs. They preprocess the raw data and quantize the data into binary series respectively. Afterward, they apply reconciliation to eliminate bit mismatches and to confirm whether identical source keys have been generated. If so, privacy amplification is applied for randomness extraction and finally, the key is used for symmetric key encryption. Otherwise, the key generation process is restarted.

5.2 Signal Preprocessing

The SEP is induced by the powerline EMR, whose intensity and distribution cannot be manipulated by the wearer. Before utilizing SEPs to generate keys, TouchKey examines whether the device is in a signal-rich environment.

To verify the device is in a signal-rich environment, TouchKey utilizes the instantaneous PSD to estimate the strength of the collected SEP. First, we calculate PSD by Welch's average periodogram method. Then, we change the scale to logarithmic, i.e., decibel, to amplify the difference between powers of weak EMR field with a larger slope by decibel(dB) = $10 \log_{10}(P)$. Finally, we find the maximum power value around 50 Hz as SEP strength and compare it with a predefined threshold. A threshold of -35 dB is a good setting for the CC2650-based system from offline test results in §6.7.1. If the maximum power value is below this threshold, it means the device is in a relatively weak powerline EMR field environment. One example of this occurs in the corridor. The measurement trace from two sensors on the same person is shown in the upper figure of Fig. 6(a) and the corresponding PSD is the blue trace in Fig. 6(b). We can see that the 50 Hz powerline component is overwhelmed by the DC component. In order to increase the similarity between raw data, we apply a 6th-order bandpass Butterworth filter with a lower cutoff frequency at 20 Hz and an upper cutoff frequency at 100 Hz. The bottom figure of Fig. 6(a) and the orange trace in Fig. 6(b) show the measurement trace and PSD after applying the filter. When the maximum power value is beyond this threshold, we use the raw data to generate the keys directly. Using raw data, we can keep all the frequency information and introduce more randomness to the key. Thus, the PSD-based filter scheme provides a trade-off between usability and randomness of raw data.



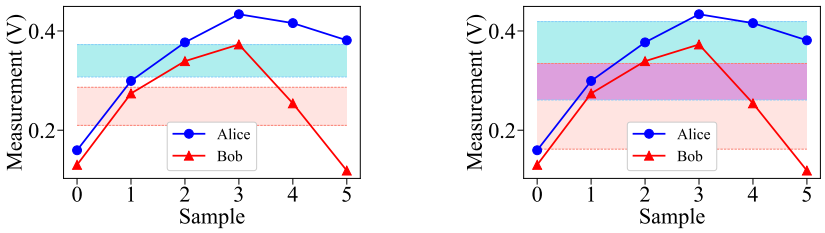
(a) Weak SEP before/after bandpass filtering.

(b) The PSD of weak SEP before/after bandpass filtering.

Fig. 6. Signal preprocessing

5.3 Quantization

In this step, Alice and Bob aim to convert a series of data samples to a binary string. In order to ultimately serve as symmetric keys, the generated binary strings should have fewer mismatch bits and higher randomness. TouchKey



(a) $\alpha = 0.4$ (b) $\alpha = 0.9$

Fig. 7. Quantization example with different scaling factor α. Different colors represent different SEP traces and their corresponding guard bands. The pink color denotes the overlapping between guard bands.

employs the guard band-based quantization method [29, 53] to convert SEP samples into bit series X_{Alice} and X_{Bob}.

We segment the SEP data using a moving window with no overlap. W is used to denote the window length. For each window, the most intuitive quantization method is to use the mean value as the boundary for encoding 1 or 0. However, the samples close to the boundary have higher probabilities to be encoded into mismatched bits. Therefore, a guard band is inserted in the middle so that all samples that fall into this guard band are discarded. This guard band is determined by two dynamic thresholds q_+ and q_- using a scaling factor α as follows:

$$q_+ = \mu + \alpha \cdot \sigma, \quad q_- = \mu - \alpha \cdot \sigma,$$

where μ and σ are the mean and standard deviation of SEP samples in a particular window. The SEP samples whose values are larger than q_+ are encoded as bit "1", and samples whose values are less than q_- are encoded as bit "0". The final key is formed by concatenating bit sequences from all the windows.

Fig. 7 shows the guard band-based quantization process in a single window ($W = 6$), where each series of samples has its own calculated guard band distinguished by blue and red. Mismatch bits arise from two aspects. First, some samples are encoded by one party while discarded by the other, such as samples #1 and #4 in Fig. 7(a). Second, some samples are encoded into different bits by different parties, such as sample #5 in Fig. 7(a). Tuning the scaling factor α has a two-sided effect on system performance. A larger α leads to more discarded samples, so the bit generation slows down, but the matching rate of the bit series increases. For $\alpha = 0.4$ in Fig. 7(a), 4 bits are generated from 6 samples with 1 mismatch bit, while for $\alpha = 0.9$ in Fig. 7(b), only 2 bits are generated but without mismatch bits.

5.4 Reconciliation

The bit series that Alice and Bob obtain after quantization may have some mismatches as Fig. 7(a) shows. For a successful symmetric key encryption, the keys should reach a 100% bit agreement rate. TouchKey resolves this bit mismatch problem by proposing a reconciliation method that effectively combines an index-based method [38, 48] and a classic challenge-and-reply protocol. We propose this improved method to achieve two-way confirmation of the key, which mainly solves the drawback of the index-based method where only one party has the ability to confirm the agreement of the keys and enhances the security.

The main idea of the index-based method is each party only retains the bits converted from common sample indexes, which is the intersection of sample indexes kept by both parties. We use Fig. 7(b) for illustration. After quantization, Alice yields $X_{Alice} = 01$ at index 0 and 3, while Bob obtains $X_{Bob} = 0110$ at index 0, 2, 3, and 5. During reconciliation, Alice sends index list $I_{Alice} = \{0, 3\}$ to Bob. Bob then calculates the common sample indexes

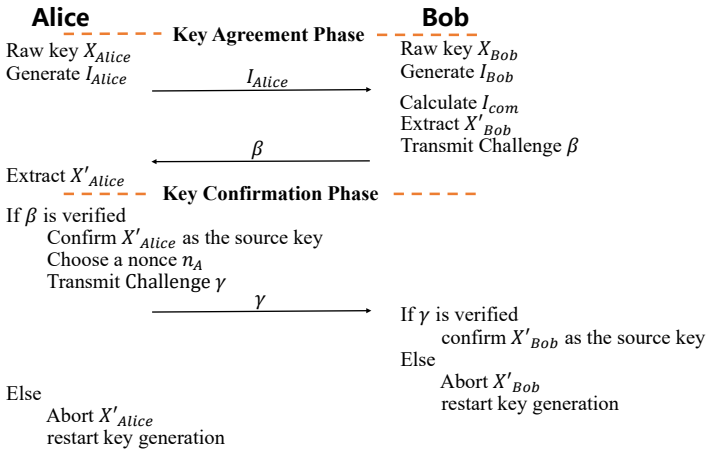


Fig. 8. Reconciliation Protocol

I_{com} by intersecting I_{Alice} with $I_{Bob} = \{0, 2, 3, 5\}$ and sends back $I_{com} = \{0, 3\}$. Finally, they extract the bits at index in I_{com} respectively to derive the same key as $X'_{Alice} = X'_{Bob} = 01$.

Fig. 8 presents the detailed workflow of the reconciliation protocol. Alice first sends the index list I_{Alice} to Bob. Bob then calculates the common sample indexes kept by both parties $I_{com} = I_{Alice} \cap I_{Bob}$, extracts a key X'_{Bob} from the raw key X_{Bob} and replies with $\beta = \{I_{com}, M_1\}$, where $M_1 = MAC(X'_{Bob}, I_{com})$. Here the "$MAC(key, message)$" is the message authentication code (MAC) [4]. Since I_{com} is random, it can be viewed as a nonce and β is the challenge. Upon receiving β, Alice extracts the key X'_{Alice} in the same way, which is the last step of the key agreement phase. Then, Alice calculates and verifies whether $MAC(X'_{Alice}, I_{com})$ is equal to M_1. If so, she confirms X'_{Alice} as the source key, chooses a nonce n_A and sends a challenge $\gamma = \{n_A, M_2\}$, where $M_2 = MAC(X'_{Alice}, I_{com}||n_A)$ to Bob. Here "||" denotes concatenation. Alice records the nonce for rejecting the replay attacks. Upon receiving γ, Bob verifies whether M_2 is equal to $MAC(X'_{Bob}, I_{com}||n_A)$. If so, she confirms X'_{Bob} as the source key. So far, both parties have confirmed the source key. If either party fails the key confirmation, she aborts the source key and prepares to restart the key generation process.

Before a secure communication channel is established, there may exist a powerful on-path attacker, who can eavesdrop or modify the exchanged message without exposing his presence. Our protocol is resilient to eavesdropping because the unique SEP which serves as the shared secret will not be exchanged. If the attacker modifies any message, the subsequent verification will fail since MAC provides message integrity. Specifically, if the attacker modifies I_{com}, it cannot modify M_1 accordingly since it does not know the key X'_{Bob}, which would fail the verification at Alice. Therefore, the reconciliation between legitimate devices ends in failure without revealing any private information. Further analysis on the MITM attack is in §7.4.

The index-based method originally only allows for one-way confirmation, where Alice confirms the generated key at the end of the reconciliation process. Bob must wait for a "Success" or "Request for restart" message to determine the outcome of the key generation. However, there is a risk that the key generation request to Bob is from an attacker. Without two-way confirmation, Bob cannot authenticate the peer is drawing from the same SEP source and may be deceived, compromising the system's security. In the enhanced method, the attacker is required to send the challenge γ. Bob can then verify the attacker's identity by checking the MAC value in γ.

After reconciliation, a source key is extracted from an entropy source. However, the source key may not be uniformly random. Also, several keys may be required for a single application. Thus, a typical approach is to perform privacy amplification, where key derivation functions like HKDF [23], randomness extractors [9, 13] and

universal hash families [19, 45] can be applied. TouchKey applies the SHA2-256 hashing algorithm for simplicity. After the privacy amplification, the final keys can be used to build a secure communication channel. TouchKey inherits standard cryptographic techniques like MAC and SHA2-256 for reconciliation and privacy amplification.

TouchKey can work in a pipeline mode in real time to further reduce the key generation time and prevent it from doubling due to the failure in key confirmation, since preprocessing and quantization are window-based methods. After several rounds of this pipeline, a substring of the key can be derived by performing reconciliation. This is repeated until the key reaches the desired length and is agreed upon between wearables.

6 EVALUATION

To evaluate the efficiency and security of TouchKey, a range of influence factors are investigated, including different wearers, various environments, devices' proximity, wearer motion statuses, protocol parameters, workflow components, wearable devices, and different attack scenarios. Afterward, the randomness of generated keys, as well as time and energy usages, are measured. Finally, the performance comparison is made with existing relevant works.

6.1 Experiment Setup

Metrics. The following five metrics are used to characterize the performance of TouchKey:

- *Bit generation rate (BGR)*: This is the number of bits generated per second, which characterizes the speed of key generation.
- *Bit agreement rate (BAR)*: This is the number of matched bits over the length of the key, which characterizes the similarity between two keys.
- *False rejection rate (FRR)*: This is one minus the ratio between the number of times that two legitimate devices generate perfectly matched keys and the total number of experiments, which characterizes the possibility of failing to establish a secure channel.
- *False acceptance rate (FAR)*: This is the rate of the attacker successfully generating the same key as legitimate users, which characterizes the possibility of the attacker successfully compromising a secure channel.
- *Entropy*: It is calculated by Shannon Entropy and characterizes the amount of information each bit can provide. It ranges in [0,1] for binary series, where larger entropy indicates more randomness of the generated keys.

Testing Subjects. We recruit 31 volunteers (13 females and 18 males) to participate in our experiments. The testing subjects are highly diverse in age, weight, and height. Specifically, the age of our volunteers ranges from 21 to 52. The weight ranges from 40 kg to 96 kg. The height ranges from 150 cm to 183 cm.

6.2 Different Wearers

We collect a set of data involving a user U and 30 other users $P_1, P_2, \ldots, P_{30}$. In the ith experiment ($1 \leq i \leq 30$), P_i holds two legitimate devices in P_i's palm and U holds an illegal device in U's palm. In each experiment, U and P_i keep still about 1 m apart. The data collection of each experiment consisted of five one-minute periods. Fig. 9 shows the FRR and FAR of this experiment. Although each experiment shows different FRR and FAR, the performance of TouchKey is consistent across different wearers. The maximum FRR is 2.00%, while the maximum FAR is 0.40%.

6.3 Various Environments

Two wearers conduct experiments in seven different indoor environments, including the study room, living room, kitchen, bedroom, dormitory, laboratory, and corridor. One wearer holds two legitimate devices and the

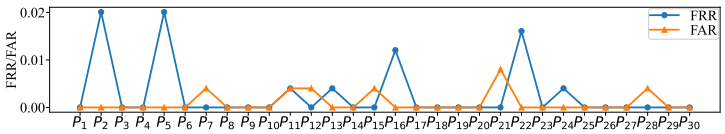


Fig. 9. FRR/FAR versus different wearers.

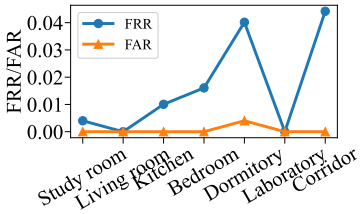


Fig. 10. FRR/FAR versus different environments.

other holds an illegal device as an attacker. Fig. 10 shows the measured performance. The FAR is less than 0.50%, which suggests that TouchKey is practically secure in various indoor environments. Although the efficiency of TouchKey is affected by the environment, the FRR is always below 5%, which suggests that TouchKey has strong universality.

6.4 Devices' Proximity

Since various commercial wearables are attached on different parts of the human body surface, it is essential to investigate the impact of devices' proximity. In this set of experiments, one legitimate device is held in the user's left palm, the other legitimate device is placed on five different locations: the left palm, wrist, elbow, right palm, and face. An attacker is involved and holds an illegal device in the palm. Fig. 11 shows that the FRR increases to about 3% when the other legitimate device is placed in right palm and face. In principle, the similarity of raw data decreases as distance increases, because the potentials of devices' GNDs have greater differences. As a result, the extracted keys are less likely to be the same. The maximum FRR and FAR are 2.87% and 0.86%, respectively, which are acceptable for using TouchKey on various wearables.

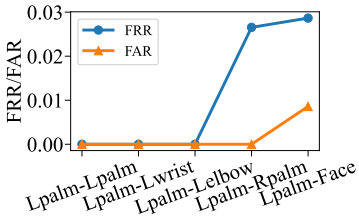


Fig. 11. Impact of devices' proximity.

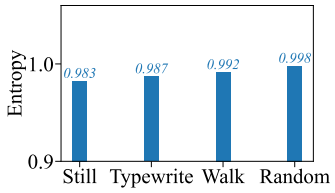
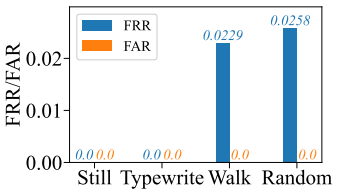
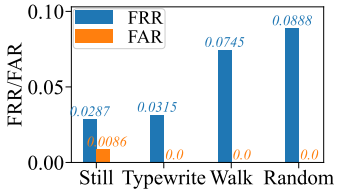


Fig. 12. Impact of motion on entropy.



(a) Two devices on the palm.



(b) One on palm and the other on face.

Fig. 13. Impact of motion on FRR and FAR.

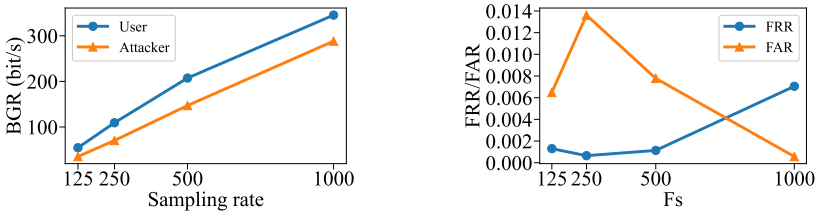
6.5 Motion Impact

Since humans perform various movements during daily activities, we need to evaluate the impact of movements on TouchKey. In this experiment, we first ask the legitimate user to hold two devices on the left palm and to keep still, typewrite, walk, and perform random movements respectively during data collection. At the same time, the attacker is asked to mimic the legitimate user. Fig. 12 shows the entropy of the key. From keeping still to performing random movements, the entropy of generated keys increases as the movements becomes more complex. This is because the movements introduce extra randomness to the SEP. Fig. 13(a) shows the FRR and FAR of the key. TouchKey has satisfactory performance when the legitimate user keeps still and typewrites, i.e., both FRR and FAR are 0%. When the legitimate user walks or performs random movements, the FRR increases to around 2%. Then, we evaluate the impact of proximity and motion simultaneously on TouchKey. We ask the legitimate user to hold one device on the left palm and the other device on the face, which is the worst-performing devices' proximity in section 6.4. Fig. 13(b) shows the FRR and FAR. The FRR reaches a maximum of 8.88% when the legitimate user is asked to do random movements. However, since TouchKey can generate a key within 0.5 seconds shown in §6.6.2, it is not troublesome for users to pair two devices in their palms before wearing devices onto other body parts.

6.6 Parameter Configuration

In this set of experiments, we use a data set that contains all of the above data to evaluate the influence of parameters on TouchKey in all conditions.

6.6.1 Impact of Sampling Rate. We evaluate the influence of sampling rate by downsampling from 1 kHz to 500 Hz, 250 Hz, and 125 Hz, respectively. Fig. 14(a) shows that the BGR increases with the sampling rate. The BGR increases to 345 bits/s for legitimate devices when the sampling rate is 1 kHz. From Fig. 14(b), when the sampling rate is below 500 Hz, the FRR is lower than 0.2%, while FAR is higher than 0.6%. This is because these sampling rates cannot capture enough information to differentiate the legitimate user and the attacker. When the sampling rate is 1 kHz, the FRR is 0.71% and the FAR is 0.06%. With a higher sampling rate, more detailed information which distinguishes legitimate devices from attackers is included. Thus, we choose a sampling rate of 1 kHz in our setting.



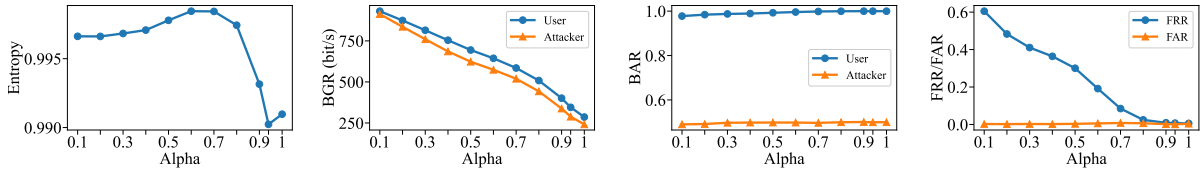
(a) BGR (b) FRR/FAR

Fig. 14. Impact of sampling rates on BGR, FRR, and FAR.

6.6.2 Impact of α and W. The scaling factor α and sliding window width W affect the performance of quantization. Thus, we need to carefully configure them. First, we demonstrate the impact of α. From Fig. 15(a), the entropy of the generated keys is greater than 0.98, which shows that generated keys have high randomness. Fig. 15(b) shows the BGR decreases with α. This is because, when α increases, the width of the guard band becomes larger accordingly and more samples fell into the guard band are discarded. Nevertheless, the BGR is larger than 250 bit/s for all α configurations. This shows that TouchKey can generate a pair of 128-bit keys in around half a second. Fig. 15(c) shows the average BAR. The BAR between legitimate devices increases from 97.76% to 99.98% as α increases, for more potential mismatched samples are discarded. Whereas, the BAR between the legitimate device

and the attacker is close to 0.5, which is the probability of random guesses. From Fig. 15(d), the FRR decreases from 60.50% to 0.56%. Yet FAR fluctuates, reaches the top when $\alpha = 0.7$ and decreases to the minimum when $\alpha = 0.94$. Thus, we choose $\alpha = 0.94$ in our setting based on the resulted good performance in BGR, FRR, and FAR.

Then, we investigate the impact of window length W. Fig. 16 shows the FRR and FAR under various W settings. The FRR is less than 0.1% when W is no smaller than 6 milliseconds. The FAR fluctuates with W but reaches the minimum when $W = 6$. Thus, a window length of 6 milliseconds is adequate for TouchKey.



(a) Entropy. (b) BGR. (c) BAR. (d) FRR and FAR.

Fig. 15. TouchKey performance with different α.

6.7 Workflow Components

6.7.1 Impact of Preprocessing. We examine the impact of signal preprocessing with various filtering thresholds. We apply various thresholds to the dataset collected at different locations described in §6.3. Fig. 17 shows the impact on FRR and FAR. As the threshold increases from -50 dB to -35 dB, the FRR decreases rapidly, falling by 5.24%, while the FAR keeps the same. As the threshold increases from -35 dB to -20 dB, the FRR decreases at a much lower speed, falling by 0.12%, while the FAR increases from 0.06% to 0.24%. In order to balance the efficiency and security, TouchKey takes the filtering threshold of -35 dB.

The reason for -35 dB being an effective threshold can be explained by observing the SEP strength at different locations shown in Fig. 18. In Fig. 18, each point represents the strength of SEP that generates a key. The SEPs collected in the corridor have low signal strength (≤ -35 dB) due to lower density of electrical of sockets and cables. Additionally, as shown in Fig. 6(b), the DC noise may overwhelm the 50 Hz component of the weak powerline EMR in some cases, which may decrease the similarity of SEP and increase the FRR. When we continue to increase the filtering threshold, the improvement of similarity and FRR by filtering is limited because the raw data collected at other locations have relatively high signal strength. Whereas due to filtering, part of the frequency information of raw data is lost, signal randomness is reduced, and the similarity between legitimate and illegal signals is increased, which in turn improves FAR.

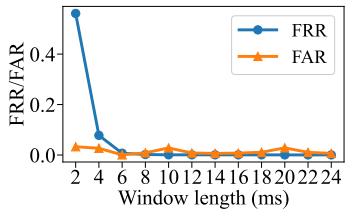


Fig. 16. Impact of window length on FRR and FAR.

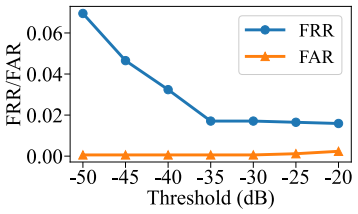


Fig. 17. Impact of filtering threshold on FRR and FAR.

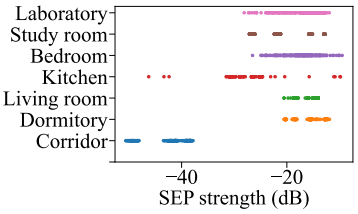
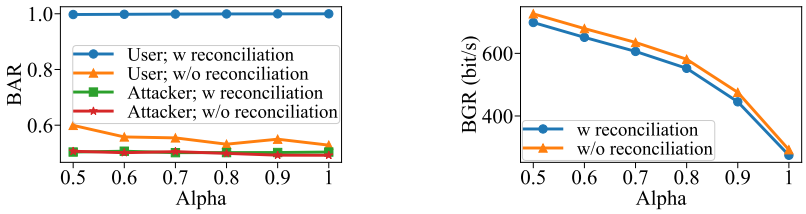


Fig. 18. SEP strength in different locations.

6.7.2 Impact of Reconciliation. Reconciliation is necessary for eliminating the mismatched bits in Alice's and Bob's keys. We evaluate the performance of reconciliation. Fig. 19(a) shows the BAR. With reconciliation, the BAR between legitimate devices increases to nearly 100%, while for the attacker, the BAR increases a little but is

still lower than 60%. From Fig. 19(b), we can see the BGR decreases about 20 bit/s after reconciliation, which is a trade-off against the rise of BAR.



(a) BAR. (b) BGR.

Fig. 19. BARs and BGRs with/without reconciliation.

6.8 Impact of Device

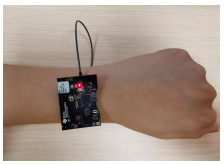


Fig. 20. CC2650 SensorTag

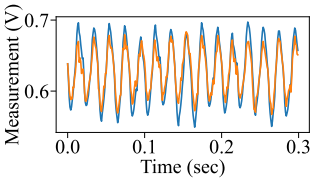


Fig. 21. Raw data collected by two SensorTags from a user.

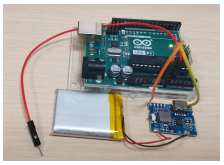


Fig. 22. Arduino UNO

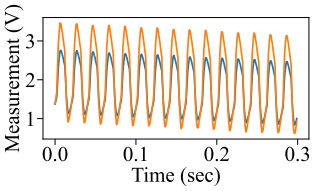


Fig. 23. Raw data collected by two Arduino from a user.

To evaluate the universality of TouchKey with the configuration determined as above, we migrate TouchKey to two other devices, i.e. CC2650 SensorTag [18] and Arduino UNO [1]. Measurement setups are shown in Fig. 20 and Fig. 22. We deploy TouchKey to devices of different vendors, i.e., two Sensortags, two Arduino boards, and one Sensortag and one Arduino, respectively. Besides, an attacker is involved. The data collected from SensorTags and Arduino boards are shown in Fig. 21 and Fig. 23, respectively. The FRR and FAR of pairing SensorTags are 0.40% and 0%; the FRR and FAR of pairing Arduino boards are 0% and 0%; the FRR and FAR of pairing Sensortag and Arduino are 2.52% and 0%, respectively. Hence, TouchKey performs well on both SensorTag and Arduino.

6.9 Adversarial Scenarios

6.9.1 Imitation Attack. Previous evaluations have introduced the imitation attack and shown the robustness of our system against imitation attacks in various situations. We further explore the impact of different imitation attackers and the distance between the legitimate user and the attacker.

Different attackers. We first conduct an experiment on the influence of different attackers. This experiment involves one legitimate user U and 30 attackers $P_1, P_2, \ldots, P_{30}$. The results in Fig. 24 show the FAR is consistent and always less than 0.8% for different attackers. This shows that the imitation attack has a very low success rate for compromising TouchKey.

Distance to the attacker. In this experiment, the attacker is asked to stand at different distances from the legitimate user. We evaluate the distance of 0.10 m, 0.25 m, 0.50 m, 0.75 m, and 1 m. Fig. 25 shows the distribution of the BAR between the legitimate device and the attacker. All average BARs fluctuate around 50% with the maximum of 50.93%. Besides, there are lots of outliers; the maximum one has a BAR of 79.69%. Since the maximum BAR is not 100%, the FAR is always 0%. This means that it is infeasible for the attacker to cheat TouchKey even

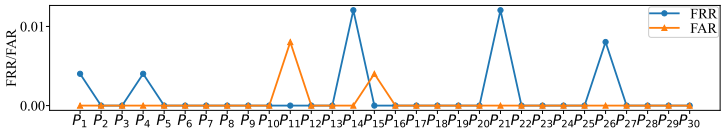


Fig. 24. Impact of different attackers on FRR and FAR.

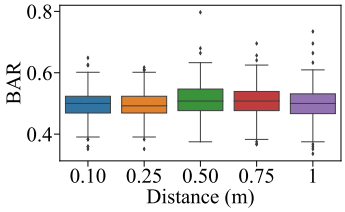


Fig. 25. Impact of distance to the attacker on BAR.

when the distance is very close. The FRRs for all distances are less than 1.61%. Thus, the efficiency of TouchKey is also not influenced by the distance.

6.9.2 Injection Attack. In injection attacks, we assume the attacker can introduce strong EMR noise to the environment where legitimate users are located by running some high-wattage electric appliances. Appliances with motors create an EMR noise synchronous to the AC power, while appliances with switched-mode power supplies emit EMR noise with frequency determined by internal oscillators. As a result, the EMR noise emitted by appliances may weaken or strengthen the powerline EMR to force Alice and Bob to agree on a key designed by the attacker. Thus, we conduct a set of experiments with an air conditioner, TV, microwave oven, and fan running separately. The user and the attacker stand close to one appliance at a time to collect SEPs. Fig. 26 shows the FRR and FAR of the injection attack. The maximum FRR is 2.41% and it occurs when turning on the TV. The FAR is always less than 1%. Therefore, the injection attacks from normal electric appliances do not degrade the security and efficiency of TouchKey.

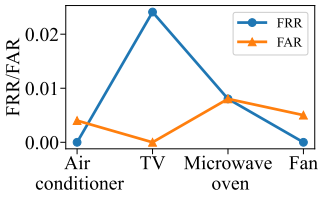


Fig. 26. FRR and FAR vs. different injection devices.

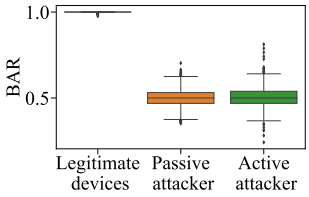


Fig. 27. BAR achieved by the user and replay attacker.

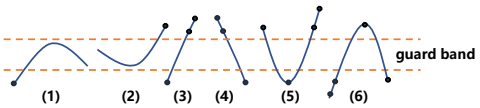


Fig. 28. Six quantization conditions.

6.9.3 Replay Attack. In replay attacks, we assume the attacker has access to the historical data of the legitimate devices as well as the historical keys. Then, the attacker tries to pair with legitimate devices by two means, which are *passive replay attack* and *active replay attack*. The passive replay attacker assumes that TouchKey is a bad random number generator, so that a previously used key has a high probability of being generated again. Thus, this attacker utilizes the previous legitimate key directly to pair with legitimate devices. The effectiveness of this attack can be understood by comparing the BAR of two consecutive legitimate keys. The active replay attack is to use historical legitimate data to re-generate a key with eavesdropped messages in the current time. To evaluate the active attack scenario, we split the legitimate data into two segments, and use the former segment as illegal data to pair with the latter segment. The distribution of BAR is shown in Fig. 27. We can see the BAR between legitimate devices is near 100%, and only a few outliers are below 100%, which suggests a low FRR. However, the average BARs achieved by passive replay attack and active replay attack are both approximately 50%. The maximum BAR achieved by the attacker is lower than 80%, which suggests that the attacker fails to generate an identical key for once. Hence, TouchKey is robust to replay attacks.

6.9.4 Inference Attack. So far, the attack models we evaluated make full use of I_{com}, i.e., the common index list between both legitimate devices. Specifically, we utilize illegitimate or expired data to execute the same protocol with eavesdropped I_{com}. It is confirmed that due to the uniqueness and temporal uncorrelation of SEP, attackers have a low probability of successful attack, which is less than 0.86% in all conditions. However, we noticed I_{Alice} may reveal more information about Alice's raw data than that I_{com} may reveal, since I_{Alice} is obtained directly after quantization. Therefore, it is necessary to devise an attack model that fully utilizes I_{Alice}. None of the existing systems using index-based method has verified the system security against the information leakage of I_{Alice}. We first propose an attacker who can infer some details of the raw data based on I_{Alice}, and then guess the legitimate key with the help of these details. We test this attack on TouchKey and prove TouchKey can resist this attack.

The inference attacker's algorithm is described as follows. Fig. 28 shows six typical quantization types for a sine wave, where the samples outside the dashed lines are retained. For types like (5) or (6), the sample list consists of three discontinuous fragments such as {0,3,5}. Thus, the attacker can infer that the sample indexed "3" is either the highest or lowest vertex. First, the attacker assumes it to be the highest vertex and infers the key, then takes the inverse of each bit to derive the key of another case. The attack is viewed as successful if either key matches the legitimate key. As discontinuous fragments are converted to different values of 0/1, this window is converted to {0,1,0}. Then, the attacker deduces other vertices as {13,23...}, since the SEP has a basic frequency component around 50 Hz. The attacker further matches each window of I_{Alice} into type (1) to (6), calculates the distance to the nearest highest vertex, and derives the converted bits. Finally, the attacker applies reconciliation. The inference attack has an average BAR of 65.39% and a FAR of 0%. Since the generated keys have sufficient randomness (see §6.11) and the attacker's guess of vertices cannot be verified, the attacker does not know which bits of his guess match the legitimate key. As a result, with a 128-bit key, TouchKey provides around 80 bits of security (i.e., $log_2(\frac{1}{0.65}^{128})$). Therefore, the attacker has a low probability to infer the correct key based on the shortfall of the guard band-based quantization method and the characteristic of SEP. This is because that movements, interferences, and randomness in EMR noise can introduce unpredictable patterns to TouchKey.

6.10 Time and Energy Usage

To validate the feasibility of running TouchKey on battery-powered mobile devices, we implement TouchKey on CC2650 LaunchPad using TI-RTOS. The MAC algorithm described in §5.4 is implemented by AES-EMAC utilizing TI-RTOS Driver for AES [17]. We measure the execution time with the system clock and energy usage with Rigol DP832A [39] of three major stages (i.e., measurement, quantization, and reconciliation). For the execution time, we run TouchKey 100 times. To measure the power usage as accurately as possible, we first measure a baseline power by only starting up the TI-RTOS system and then subtracting this baseline from the power reading. The power usage for each stage is an average value for over 80 s of measurement. The energy usage for each stage is the multiplication of the execution time and the power. Table 1 summarizes the results. The overall execution time and energy usage of TouchKey are 0.44 s and 2.716 mJ, respectively. Such an energy cost is negligible to the capacity of a wearable's battery, e.g., $4.28 \times 10^3 J$ for Apple Watch Series 7 [12].

6.11 Randomness

We test the randomness by applying the NIST suite of statistical tests [41] to all 128-bit keys after reconciliation. If P-value returned by NIST is larger than 1%, it indicates that the tested key has sufficient randomness. The test results shown in Table 2 show that keys generated from our approach are highly random. Since the keys after reconciliation have high randomness, they are not easily cracked by offline brute force attacks.

Table 1. Time and Energy usage of TouchKey

	Time (ms)	Power (mW)	Energy (mJ)
Measurement	371.01	1.60	0.594
Quantization	12.59	3.36	0.042
Reconciliation	59.34	34.88	2.070

Table 2. NIST Test Results

NIST Statistical Test	P-value
Frequency	0.435665
Block Frequency	0.858423
Runs	0.031735
Longest Run	0.139241
FFT Test	0.194043
Non overlapping Template	1.000000
Linear Complexity	0.454698
Serial	0.026849
Approximate Entropy	1.000000
Cumulative Sums	0.598698
Random Excursions	0.131382
Random Excursions Variant	0.166546

6.12 Comparison

We compare TouchKey with prior related systems in terms of BGR, execution time, FRR, and FAR that characterize usability, efficiency, and security. For TouchKey, we present the average BGR, the average FRR, and the worst FAR under all evaluation conditions. For other systems, we directly use the results presented in their publications.

Table 3 compares TouchKey with other six state-of-the-art schemes. Specifically, AeroKey, harnessing, and ours utilize EMR. Shake-n-shake, KEHKey, H2B, EMG, and ours leverage shaking, gait, heartbeat, electromyography, and SEP, respectively, all of which belong to on-body signals. The time usage in the table indicates the overall execution time for the main stages in the key generation process for a 128-bit key.

TouchKey has the highest BGR of 345 bit/s. Other schemes like AeroKey, EMG, and H2B have a relatively low BGR because they take a long segment of data to generate a few bits in the quantization stage. TouchKey also has an advantage in overall execution time. Schemes like KEHKey and H2B apply a compressive sensing-based reconciliation method that requires solving a l_1 minimization problem, which is computationally intensive and may degrade the user experience. Differently, except for a fast quantization method, TouchKey also applies a lightweight and time-efficient reconciliation method. In terms of FRR, TouchKey has an average FRR of 0.71%, which is the second best among the existing schemes. Besides, TouchKey has the lowest FAR of 0.86% among schemes that provide FAR results.

Table 3. Performance Comparison

	BGR (bit/s)	Time (s)	FRR	FAR
TouchKey (Ours)	**345**	**0.44**	0.71%	**0.86%**
AeroKey [25]	5.8	24	3.4%	3.4%
Harnessing [20]	138	0.97	3.1%	5.8%
Shake-n-shake [42]	98.4	1.47	1.6%	1.6%
KEHKey [29]	12.57	64.18	0.0%	N/A
H2B [30]	3	177.67	4.4%	N/A
EMG [32]	5.51	23.23	11.2%	N/A

7 DISCUSSIONS

7.1 Applicability to Environments

The SEP we utilize is induced by ambient EMR fields in the environment. Inside the building, the ubiquitous cables, sockets, and electric appliances generate strong EMR fields, thus providing a signal-rich environment. However,

the EMR field is too weak in outdoor environments. Therefore, TouchKey is not suitable for outdoor environments. Since people spend most of their time indoors (e.g., 87% for Americans [22]), TouchKey has satisfactory availability.

7.2 Reconciliation Method

The fuzzy commitment scheme [21] is a widely-used reconciliation method. However, the common practice is to use exclusive-OR [26, 52] when two parties encrypt and decrypt the codeword with the quantized keys. To perform XOR, these two quantized keys need to have the same length, which requires the quantization method to have a fix key generation rate. While in the quantization method of TouchKey, the key generation rate of quantization is not fixed, such that the widely-used fuzzy commitment scheme may not apply to our approach directly. We leave it as future work to investigate the applicability and benefits of fuzzy commitment scheme in generating keys from SEP.

7.3 Injection Attack

TouchKey has shown its security performance by rejecting injection attacks of running high-wattage electric appliances in §6.9.2. However, if the attacker generates a strong EMR field that overrides the powerline EMR, the attacker may directly infer the correct key. However, generating such fields requires bulky equipment. Because the length of the antenna is usually half of the wavelength, for low-frequency EMR at 50 Hz, the antenna length needs to be about 6×10^6 m. Thus, such an injection attack is not practical.

7.4 Monster-in-the-middle Attack

An MITM attacker can make independent connections with legitimate devices and relay or alter the contents of the conversation between them, while the legitimate devices are not aware of the attacker. During the reconciliation period, the attacker who has full control of the public network may perform an MITM attack. However, an MITM attack can rarely succeed in TouchKey. In the public network, Alice and Bob only exchange the index of samples, not the shared secret of raw SEP data or converted keys. When the attacker tries to impersonate Alice and sends the $I_{attacker}$ to Bob, some retained samples are prone to convert to different bits due to the discrepancy of SEP data between Bob and the attacker. Thus, it is hard for the attacker to generate a key identical to Bob's.

8 CONCLUSION

In this paper, we present TouchKey, a novel key generation approach for wearables that exploits SEP induced by powerline EMR to generate a common symmetric key between legitimate devices. The core idea is that the SEP signal has high randomness and is distinctive to different people. Compared to prior key generation schemes, TouchKey is a lightweight scheme that neither requires any user involvement nor relies on any dedicated sensor. On a prototype, TouchKey demonstrates its efficiency under various settings with a low FRR of 0.71% and a high BGR of 345 bit/s; its security against various types of attacks with a low FAR of 0.86%; its usability with total execution time and energy usage of 0.44 s and 2.716 mJ, respectively. To summarize, TouchKey is a secure and practical key generation approach for low-cost and resource-constrained wearables.

ACKNOWLEDGMENTS

Thank all the colleagues and volunteers for helping conduct experiments. Thank the anonymous editors and reviewers for the valuable comments. This work is supported by the National Science Foundation of China (NSFC) under Grant No. U1909207, No. 62202407, the Research Grants Council (RGC) of Hong Kong, China, under GRF Grants No. 14214022, the State Key Laboratory of Industrial Control Technology, Zhejiang University, China (No.ICT2022B67), and in part by the Ministry of Education, Singapore, under its Academic Research Fund Tier 1 (RG88/22). Chaojie Gu is the corresponding author.

REFERENCES

[1] [n. d.]. Arduino UNO R3. https://docs.arduino.cc/hardware/uno-rev3.

[2] Tomoyuki Aono, Keisuke Higuchi, Takashi Ohira, Bokuji Komiyama, and Hideichi Sasaoka. 2006. Ieee802. 15.4 esparskey (encryption scheme parasite array radiator secret key). *Electronics and Communications in Japan (Part I: Communications)* 89, 12 (2006), 31–44.

[3] Shu-Di Bao, Carmen CY Poon, Yuan-Ting Zhang, and Lian-Feng Shen. 2008. Using the timing information of heartbeats as an entity identifier to secure body sensor network. *IEEE transactions on information technology in biomedicine* 12, 6 (2008), 772–779.

[4] Mihir Bellare, Ran Canetti, and Hugo Krawczyk. 1996. Keying hash functions for message authentication. In *Annual international cryptology conference*. Springer, 1–15.

[5] Arne Bruesch, Ngu Nguyen, Dominik Schürmann, Stephan Sigg, and Lars Wolf. 2019. Security properties of gait for mobile device pairing. *IEEE Transactions on Mobile Computing* 19, 3 (2019), 697–710.

[6] Gabe Cohn, Daniel Morris, Shwetak Patel, and Desney Tan. 2012. Humantenna: using the body as an antenna for real-time whole-body interaction. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 1901–1910.

[7] Gabe Cohn, Daniel Morris, Shwetak N Patel, and Desney S Tan. 2011. Your noise is my command: sensing gestures using the body as an antenna. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 791–800.

[8] Whitfield Diffie and Martin E Hellman. 2019. New directions in cryptography. In *Secure communications and asymmetric cryptosystems*. Routledge, 143–180.

[9] Zeev Dvir, Swastik Kopparty, Shubhangi Saraf, and Madhu Sudan. 2013. Extensions to the method of multiplicities, with applications to Kakeya sets and mergers. *SIAM J. Comput.* 42, 6 (2013), 2305–2328.

[10] Habiba Farrukh, Muslum Ozgur Ozmen, Faik Kerem Ors, and Z Berkay Celik. 2022. One Key to Rule Them All: Secure Group Pairing for Heterogeneous IoT Devices. In *2023 IEEE Symposium on Security and Privacy (SP)*. IEEE Computer Society, 1693–1709.

[11] Tobias Grosse-Puppendahl, Xavier Dellangnol, Christian Hatzfeld, Biying Fu, Mario Kupnik, Arjan Kuijper, Matthias R Hastall, James Scott, and Marco Gruteser. 2016. Platypus: Indoor localization and identification through sensing of electric potential changes in human bodies. In *Proceedings of the 14th Annual International Conference on Mobile Systems, Applications, and Services*. 17–30.

[12] GSMARENA. [n. d.]. Apple Watch Series 7 Aluminum. https://www.gsmarena.com/apple_watch_series_7_aluminum-11107.php.

[13] Venkatesan Guruswami, Christopher Umans, and Salil Vadhan. 2009. Unbalanced expanders and randomness extractors from Parvaresh–Vardy codes. *Journal of the ACM (JACM)* 56, 4 (2009), 1–34.

[14] Jun Han, Albert Jin Chung, Manal Kumar Sinha, Madhumitha Harishankar, Shijia Pan, Hae Young Noh, Pei Zhang, and Patrick Tague. 2018. Do you feel what I hear? Enabling autonomous IoT device pairing using different sensor types. In *2018 IEEE Symposium on Security and Privacy (SP)*. IEEE, 836–852.

[15] Shibo He, Kun Shi, Chen Liu, Bicheng Guo, Jiming Chen, and Zhiguo Shi. 2022. Collaborative Sensing in Internet of Things: A Comprehensive Survey. *IEEE Communications Surveys & Tutorials* (2022).

[16] TEXAS INSTRUMENTS. [n. d.]. LAUNCHXL-CC2650: SimpleLink™ CC2650 wireless MCU LaunchPad™ Development Kit. https://www.ti.com/tool/LAUNCHXL-CC2650?keyMatch=CC2650%20LAUNCHPAD.

[17] TEXAS INSTRUMENTS. [n. d.]. TI-RTOS Drivers: CryptoCC26XX.h File Reference. https://dev.ti.com/tirex/explore/content/tirtos_cc13xx_cc26xx_2_21_00_06/products/tidrivers_cc13xx_cc26xx_2_21_00_04/docs/doxygen/html/_crypto_c_c26_x_x_8h.html.

[18] TEXAS INSTRUMENTS. [n. d.]. TIDC-CC2650STK-SENSORTAG: SimpleLink™ multi-standard CC2650 SensorTag™ kit reference design. https://www.ti.com/tool/TIDC-CC2650STK-SENSORTAG?keyMatch=CC2650%20SENSORTAG.

[19] 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 15th annual international conference on Mobile computing and networking*. 321–332.

[20] Wenqiang Jin, Ming Li, Srinivasan Murali, and Linke Guo. 2020. Harnessing the ambient radio frequency noise for wearable device pairing. In *Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security*. 1135–1148.

[21] Ari Juels and Martin Wattenberg. 1999. A fuzzy commitment scheme. In *Proceedings of the 6th ACM conference on Computer and communications security*. 28–36.

[22] Neil E Klepeis, William C Nelson, Wayne R Ott, John P Robinson, Andy M Tsang, Paul Switzer, Joseph V Behar, Stephen C Hern, and William H Engelmann. 2001. The National Human Activity Pattern Survey (NHAPS): a resource for assessing exposure to environmental pollutants. *Journal of Exposure Science & Environmental Epidemiology* 11, 3 (2001), 231–252.

[23] Hugo Krawczyk. 2010. Cryptographic extraction and key derivation: The HKDF scheme. In *Annual Cryptology Conference*. Springer, 631–648.

[24] Kyuin Lee, Neil Klingensmith, Suman Banerjee, and Younghyun Kim. 2019. Voltkey: Continuous secret key generation based on power line noise for zero-involvement pairing and authentication. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 3, 3 (2019), 1–26.

[25] Kyuin Lee, Yucheng Yang, Omkar Prabhune, Aishwarya Lekshmi Chithra, Jack West, Kassem Fawaz, Neil Klingensmith, Suman Banerjee, and Younghyun Kim. 2022. AEROKEY: Using Ambient Electromagnetic Radiation for Secure and Usable Wireless Device Authentication.

Proc. ACM Interact. Mob. Wearable Ubiquitous Technol. 6, 1, Article 20 (mar 2022), 29 pages. https://doi.org/10.1145/3517254

[26] Xiaopeng Li, Qiang Zeng, Lannan Luo, and Tongbo Luo. 2020. T2pair: Secure and usable pairing for heterogeneous iot devices. In *Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security*. 309–323.

[27] Yang Li, Rui Tan, and David KY Yau. 2017. Natural timestamping using powerline electromagnetic radiation. In *2017 16th ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN)*. IEEE, 55–66.

[28] Kuan-Chieh Liao and Wei-Hsun Lee. 2010. A novel user authentication scheme based on QR-code. *Journal of networks* 5, 8 (2010), 937.

[29] Qi Lin, Weitao Xu, Guohao Lan, Yesheng Cui, Hong Jia, Wen Hu, Mahbub Hassan, and Aruna Seneviratne. 2020. KEHKey: Kinetic energy harvester-based authentication and key generation for body area network. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 4, 1 (2020), 1–26.

[30] Qi Lin, Weitao Xu, Jun Liu, Abdelwahed Khamis, Wen Hu, Mahbub Hassan, and Aruna Seneviratne. 2019. H2B: Heartbeat-based secret key generation using piezo vibration sensors. In *Proceedings of the 18th International Conference on Information Processing in Sensor Networks*. 265–276.

[31] 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 Twentieth ACM International Symposium on Mobile Ad Hoc Networking and Computing*. 311–320.

[32] Roberto Merletti and Philip J Parker. 2004. *Electromyography: physiology, engineering, and non-invasive applications*. Vol. 11. John Wiley & Sons.

[33] Markus Miettinen, Nadarajah Asokan, Thien Duc Nguyen, Ahmad-Reza Sadeghi, and Majid Sobhani. 2014. Context-based zero-interaction pairing and key evolution for advanced personal devices. In *Proceedings of the 2014 ACM SIGSAC conference on computer and communications security*. 880–891.

[34] Moni Naor, Lior Rotem, and Gil Segev. 2020. Out-Of-Band Authenticated Group Key Exchange: From Strong Authentication to Immediate Key Delivery. In *1st Conference on Information-Theoretic Cryptography (ITC 2020) (Leibniz International Proceedings in Informatics (LIPIcs), Vol. 163)*, Yael Tauman Kalai, Adam D. Smith, and Daniel Wichs (Eds.). Schloss Dagstuhl–Leibniz-Zentrum für Informatik, Dagstuhl, Germany, 9:1–9:25. https://doi.org/10.4230/LIPIcs.ITC.2020.9

[35] Carmen CY Poon, Yuan-Ting Zhang, and Shu-Di Bao. 2006. A novel biometrics method to secure wireless body area sensor networks for telemedicine and m-health. *IEEE Communications Magazine* 44, 4 (2006), 73–81.

[36] J Patrick Reilly. 2012. *Applied bioelectricity: from electrical stimulation to electropathology*. Springer Science & Business Media.

[37] Allied Market Research. March, 2022. Wearable Technology Market by Device, by Product Type, by Application: Global Opportunity Analysis and Industry Forecast, 2020-2031. https://www.alliedmarketresearch.com/wearable-technology-market.

[38] Girish Revadigar, Chitra Javali, Wen Hu, and Sanjay Jha. 2015. DLINK: Dual link based radio frequency fingerprinting for wearable devices. In *2015 IEEE 40th Conference on Local Computer Networks (LCN)*. IEEE, 329–337.

[39] RIGOL. [n. d.]. Programmable linear DC power supply DP800 Series. https://www.rigol.eu/products/dc-power/dp800.html.

[40] Marc Roeschlin, Ivan Martinovic, and Kasper Bonne Rasmussen. 2018. Device Pairing at the Touch of an Electrode. In *NDSS*, Vol. 18. 18–21.

[41] Andrew Rukhin, Juan Soto, James Nechvatal, Miles Smid, and Elaine Barker. 2001. *A statistical test suite for random and pseudorandom number generators for cryptographic applications*. Technical Report. Booz-allen and hamilton inc mclean va.

[42] Yiran Shen, Fengyuan Yang, Bowen Du, Weitao Xu, Chengwen Luo, and Hongkai Wen. 2018. Shake-n-shack: Enabling secure data exchange between smart wearables via handshakes. In *2018 IEEE international conference on pervasive computing and communications (PerCom)*. IEEE, 1–10.

[43] Babins Shrestha, Nitesh Saxena, Hien Thi Thu Truong, and N Asokan. 2014. Drone to the rescue: Relay-resilient authentication using ambient multi-sensing. In *International Conference on Financial Cryptography and Data Security*. Springer, 349–364.

[44] Yong Wang, Garhan Attebury, and Byrav Ramamurthy. 2006. A survey of security issues in wireless sensor networks. (2006).

[45] Yunchuan Wei, Kai Zeng, and Prasant Mohapatra. 2012. Adaptive wireless channel probing for shared key generation based on PID controller. *IEEE Transactions on Mobile Computing* 12, 9 (2012), 1842–1852.

[46] Fengyuan Xu, Zhengrui Qin, Chiu C Tan, Baosheng Wang, and Qun Li. 2011. IMDGuard: Securing implantable medical devices with the external wearable guardian. In *2011 Proceedings IEEE INFOCOM*. IEEE, 1862–1870.

[47] Weitao Xu, Sanjay Jha, and Wen Hu. 2018. Lora-key: Secure key generation system for lora-based network. *IEEE Internet of Things Journal* 6, 4 (2018), 6404–6416.

[48] Weitao Xu, Girish Revadigar, Chengwen Luo, Neil Bergmann, and Wen Hu. 2016. Walkie-talkie: Motion-assisted automatic key generation for secure on-body device communication. In *2016 15th ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN)*. IEEE, 1–12.

[49] Wenyuan Xu, Wade Trappe, Yanyong Zhang, and Timothy Wood. 2005. The feasibility of launching and detecting jamming attacks in wireless networks. In *Proceedings of the 6th ACM international symposium on Mobile ad hoc networking and computing*. 46–57.

[50] Zhenyu Yan, Yang Li, Rui Tan, and Jun Huang. 2017. Application-layer clock synchronization for wearables using skin electric potentials induced by powerline radiation. In *Proceedings of the 15th ACM Conference on Embedded Network Sensor Systems*. 1–14.

[51] Zhenyu Yan, Qun Song, Rui Tan, Yang Li, and Adams Wai Kin Kong. 2019. Towards touch-to-access device authentication using induced body electric potentials. In *The 25th Annual International Conference on Mobile Computing and Networking*. 1–16.

[52] Lin Yang, Wei Wang, and Qian Zhang. 2016. Secret from muscle: Enabling secure pairing with electromyography. In *Proceedings of the 14th ACM Conference on Embedded Network Sensor Systems CD-ROM*. 28–41.

[53] Kai Zeng, Daniel Wu, An Chan, and Prasant Mohapatra. 2010. Exploiting multiple-antenna diversity for shared secret key generation in wireless networks. In *2010 Proceedings IEEE INFOCOM*. IEEE, 1–9.