MobileKey: A Fast and Robust Key Generation System for Mobile Devices

Keqi Song
City University of Hong Kong
Shenzhen Research Institute, City
University of Hong Kong
PRC
kqsong2-c@my.cityu.edu.hk

Zimeng Zhu
Tongji University
PRC
2130397@tongji.edu.cn

Huanqi Yang
City University of Hong Kong
Shenzhen Research Institute, City
University of Hong Kong
PRC
huanqi.yang@my.cityu.edu.hk

Tao Ni
City University of Hong Kong
Shenzhen Research Institute, City
University of Hong Kong
PRC
taoni2-c@my.cityu.edu.hk

Weitao Xu*
City University of Hong Kong
Shenzhen Research Institute, City
University of Hong Kong
PRC
weitaoxu@cityu.edu.hk

ABSTRACT

Wireless key generation is promising in establishing a pair of secret keys for ubiquitous Wi-Fi networks. However, existing Wi-Fi-based key generation systems are not always applicable in dynamic mobile wireless environments because they are completed on static personal computers. To fill this gap, this paper proposes a novel key generation system for dynamic mobile devices, named *MobileKey*. We conduct extensive experiments and analysis to explore the feasibility of wireless key generation for mobile devices. Furthermore, we propose a fast and robust key generation scheme suitable for mobile devices. Evaluation in real-world environments shows that our system can achieve up to 5000 bit/s key generation rate and 99.1% key matching rate. Compared with state-of-the-art systems, *MobileKey* improves the key generation rate by 25×.

CCS CONCEPTS

• **Security and privacy** → *Mobile and wireless security*.

KEYWORDS

Device pairing; Key Generation; Channel State Information

ACM Reference Format:
Keqi Song, Zimeng Zhu, Huanqi Yang, Tao Ni, and Weitao Xu*. 2022. *MobileKey*: A Fast and Robust Key Generation System for Mobile Devices. In *Proceedings of the 2022 ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp/ISWC '22 Adjunct), September 11–15, 2022, Cambridge, United Kingdom.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3544793.3560413

1 INTRODUCTION

Wi-Fi-enabled mobile devices (e.g., smartphones, tablets, and laptops) have been widely adopted in our daily life, which actively exchange information and share various data. However, these emerging applications are vulnerable to malicious attacks due to the broadcast nature of wireless communication. Therefore, ensuring secure communications and data transmission confidentiality between mobile devices is of great significance. For instance, employees in offices need to share with each other daily working documents and information on important company decisions using smartphones or tablets.

Due to the high complexity of asymmetric encryption and the limited computational resources of mobile devices, traditional public key infrastructure [20] is not suitable for securing the communication among Internet of things (IoT) devices. In addition, because of the difficulty of large-scale deployment and the lack of key revocation capability, methods such as Pre-Shared Keys [5] falls short in key generation in IoT device communication [7, 10]. Recently, dynamic key generation based on physical layer information of wireless signals (i.e., cryptographic keys) has attracted more attentions [13]. In a dynamic key generation, both sender and receiver generate random sequences to be used as keys based on three principles of the wireless fading channel, including channel proximity, spatial variation, and temporal variation [23]. Specifically, channel proximity describes the channel characteristics between two devices are nearly identical, spatial variation means that different locations can affect the strength of the wireless signals due to multi-path effects, and temporal variation illustrates that the movement of objects in the environment can cause changes in the wireless signal. For instance, prior dynamic key generation systems such as *TDS* [21] and *ProxiMate* [14] were developed based on the principle of channel proximity. However, existing studies [1, 8, 9, 12, 16, 17, 21, 24] mainly focus on generating keys on static personal computers, which is not applicable in a real-world scenario where users exploit mobile devices for communication and data transmission.

In this paper, we propose the first key generation system for Wi-Fi-enabled mobile devices, named *MobileKey*. First, we conduct

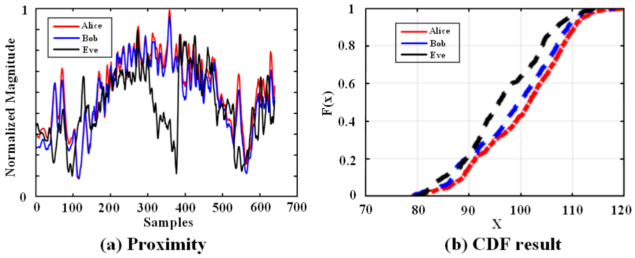
UbiComp/ISWC '22 Adjunct, September 11–15, 2022, Cambridge, United Kingdom
© 2022 Association for Computing Machinery.
ACM ISBN 978-1-4503-9423-9/22/09…$15.00
https://doi.org/10.1145/3544793.3560413

a preliminary study to demonstrate the feasibility of dynamic key generation for Wi-Fi-enabled mobile devices. Afterward, we propose a fast and robust key generation scheme consisting of data pre-processing, quantization, reconciliation, and privacy amplification [2]. Based on this scheme, we have implemented *MobileKey* on commodity smartphones in real-world environments. The empirical results indicate our system achieves fast and robust key agreement compared with the existing key generation systems, achieving up to 5000 bit/s key generation rate and 99.1% key matching rate. Furthermore, we analyze the factors that impact *MobileKey*'s performance and discuss relevant application scenarios. Besides, *MobileKey* shows the potential of applying a dynamic key generation solution on commodity smartphones, which ensures the security of mobile devices in a more portable and practical manner.

2 PRELIMINARY STUDY

In this section, we conduct a preliminary study to verify whether the extracted Channel State Information (CSI) values in mobile devices satisfy the three bases of key generation, i.e., channel proximity, spatial correlation, and time variation.

Channel proximity. Channel proximity guarantees that only devices in close physical proximity (less than 0.5λ where λ is the wavelength) can agree on the same key, and other devices outside a certain distance cannot generate the key [14, 21]. To verify the channel proximity property in mobile devices, two devices that are physically close together receive Wi-Fi signals simultaneously, and the other device that receives the signal is physically farther apart. From Figure 1 we can see the CSI values of the legitimate devices are close to each other while the eavesdropper has different CSI signals. Also, from the Cumulative Distribution Function (CDF) result, we observe that the CDF of Eve is also different from Alice and Bob. Although Alice and Bob's Channel proximity holds, it is obvious to observe that their CSI sequences are not identical due to multi-path effects [14] as well as channel noise.

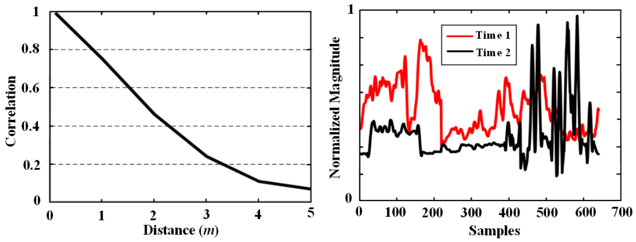


(a) Proximity (b) CDF result

Figure 1: Channel proximity.

Spatial decorrelation and time variation. According to channel proximity [14, 21], if two devices are separated by half a wavelength, the channels are statistically irrelevant. To verify the spatial correlation property, we change the distance between Alice and Bob from 0.1 m to 5 m. The channel correlation between these two devices drops off sharply as the physical distance increases, as shown in Figure 2. When the distance between Alice and Bob is 3 cm, the correlation achieves 0.98. However, the correlation coefficient can be lower than 0.1 when the distance is 5 m. In addition, to verify the time correlation property, we conduct an experiment by shaking the first device and keeping the second device still. From Figure 2 we can see that the CSI values of the two devices are significantly different.

From the above preliminary, we know that the CSI values of mobile devices could satisfy channel proximity, spatial correlation, and time variation. Therefore, we design a key generation system to perform key generation for Wi-Fi-enabled mobile devices.



(a) Spatial decorrelation (b) Time variation

Figure 2: Spatial decorrelation and time variation.

3 SYSTEM ARCHITECTURE

In this section, we first introduce an overview of *MobileKey*, then present the design details.

3.1 System Model

Figure 3 shows the system model of *MobileKey*. In *MobileKey*, we assume that two legitimate Wi-Fi-enabled mobile devices, Alice and Bob, are going to securely exchange their private information. They have no prior shared secret and are physically close to each other (less than 0.5λ where λ is the wavelength). Since Alice and Bob are physically close to each other, according to the channel proximity, they can agree on the same key. At the same time, attacker Eve, who is away from legitimate devices, also tries to generate the key. If Eve moves close to Alice and Bob, it will be easily seen by the users of Alice and Bob. Alice, Bob, and Eve can hear a public Wi-Fi source, as shown in Figure 3. Eve has complete knowledge of the proposed method and algorithms. As discussed in Section 2, the channels between legitimate users and eavesdroppers are statistically uncorrelated. Therefore, it is impossible for Eve to generate the key same as Alice and Bob. Therefore, the communication between two legitimate users based on the physical layer of the wireless signal can be securely encrypted.

3.2 System Design

Figure 4 shows the workflow of *MobileKey*. This workflow consists of four main stages: pre-processing, quantization, reconciliation, and privacy amplification.

Pre-processing. In the first step, Alice and Bob perform channel probing by receiving multiple Wi-Fi packets to obtain channel measurements. After channel probing, we use Fourier transform to remove environmental noise and use the moving average method commonly used in time series analysis, as shown in Figure 5.

Quantization. In the quantization stage, Alice and Bob use a multi-level quantization method to convert the CSI values into binary bits. We use the mean and standard deviation (std) of the CSI sequence z_i to generate the threshold as the reference levels.

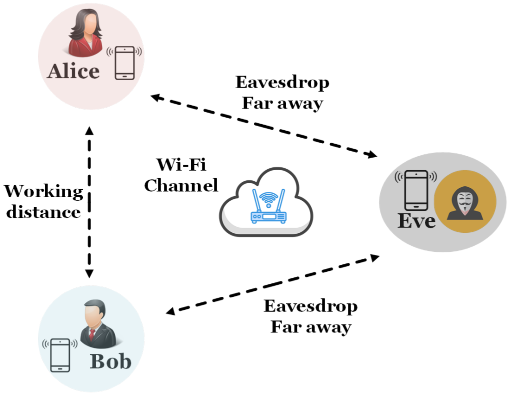


Figure 3: System model of *MobileKey*.

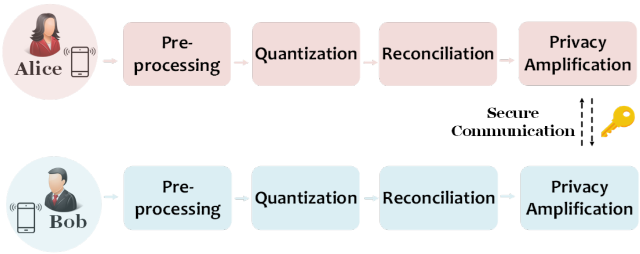


Figure 4: System workflow.

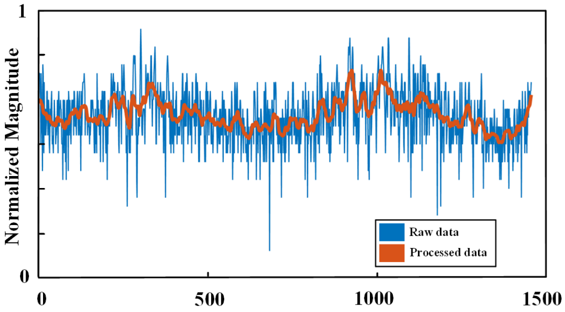


Figure 5: Data pre-processing.

As shown in Figure 6, the Alice's and Bob's initial keys K_i can be calculated from the moving window of the magnitude of processed Wi-Fi CSI sequences according to Equation 1. We empirically let α_1 equal 0.1 and α_2 equal 0.6.

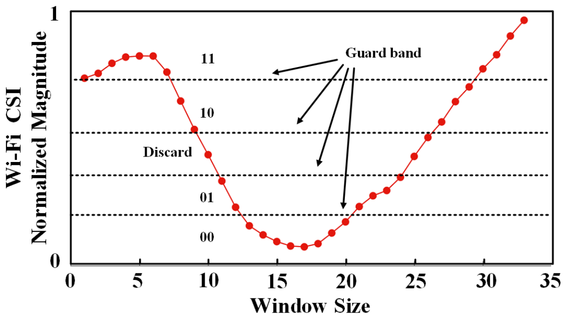


Figure 6: Illustration of quantization process.

$$K_i = \begin{cases} 00, & z_i < \text{mean} - \alpha_2 * \text{std}, \\ 01, & \text{mean} - \alpha_2 * \text{std} \le z_i < \text{mean} - \alpha_1 * \text{std}, \\ 10, & \text{mean} + \alpha_1 * \text{std} \le z_i < \text{mean} + \alpha_2 * \text{std}, \\ 11, & z_i \ge \text{mean} + \alpha_2 * \text{std} \end{cases} \tag{1}$$

Reconciliation and privacy amplification. In the reconciliation and privacy amplification stage, we adopt the compressed sensing (CS) based reconciliation method [22] to remove the partial key mismatch problem caused by environmental noise. Compressed sensing [3], as a sparse sampling theory, can obtain high-dimension sparse matrices from a small number of low dimension measurement signals. The use of CS-based reconciliation method can reduce transmission overhead because this method transmits compressed vectors instead of the original keys. Previous research found that the CS-based method outperforms error-correction code based reconciliation approaches [22]. Based on CS-based reconciliation, even if an illegal eavesdropper Eve captures the transmission on a public broadcast, the original key cannot be recovered [4, 11, 22]. Although information reconciliation increases the matching level between Alice and Bob, this also reveals part of the information to the eavesdropper because the compressed vectors are transmitted over a public channel. Therefore, in privacy amplification stage we adopt a general hash function SHA (e.g., AES-128) to improve the randomness of the final dynamic key and achieve privacy amplification.

4 EVALUATION

4.1 Experimental Setup

Figure 7 shows the experimental setup of *MobileKey*. We used a HUAWEI 4G Pro 2 router as the public Wi-Fi source and used three commodity mobile phones (Nexus 5 *2 and Nexus 6P) as Alice, Bob, and Eve. Each mobile phone was installed with firmware *Nexmon* according to [6, 19] so that each device could get raw Wi-Fi CSI data. In the experiment environment, Alice and Bob (Nexus 5 *2) were in close physical proximity ($0.5\lambda \approx 3$ cm). Eve (Nexus 6P) performs eavesdropping attacks remotely (50 cm).

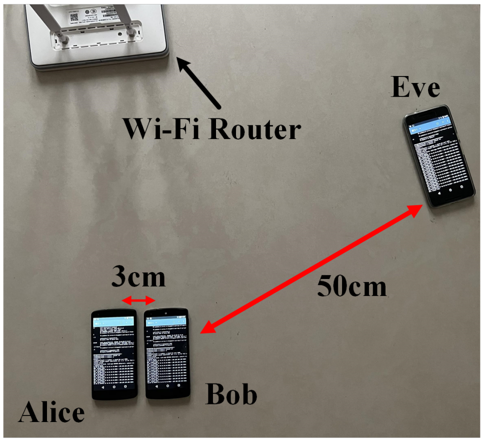


Figure 7: Experimental setup.

4.2 Overall Performance

We evaluate the performance of *MobileKey* by using four performance metrics: randomness, Key Matching Rate (KMR), and Key

Generation Rate(KGR). We compare *MobileKey* with two representative proximity-based key generation systems, *TDS* [21] and *ProxiMate* [14].

Randomness. The randomness of the secret keys is evaluated using the National Institute of Standards and Technology (NIST) set of statistical tests [18]. *p*-values produced in the suite show how random the generated key is. The randomness hypothesis is rejected if the *p*-value is less than 1%, meaning that the key is not random. Table 1 shows us all *p*-values are more than 0.01 for different kinds of tests, indicating that the generated keys by *MobileKey* have strong randomness.

Table 1: NIST Testing.

Test Name	*p*-values
Frequency	0.7317
Block Frequency	0.6852
LongestRun	0.8532
FFT	0.5341
Linear Complexity	0.9605

KMR. KMR is the percentage of matching bits over all generated bits between two legitimate users. In this evaluation, we set the distance between Alice and Bob to 2 cm, 5 cm while keeping still. As shown in the Figure 8, *MobileKey* has a slight improvement over *TDS* [21] in previous studies [15, 21]. When the distance changed from 2 cm to 5 cm, the KMR of *TDS* decreased slightly, while the key matching rate of *MobileKey* did not. Specifically, *MobileKey* increases the KMR by 2% and 21% compared with *TDS* and *ProxiMate*, respectively.

KGR. KGR refers to the speed at which dynamic keys are generated. When calculating KGR, we only consider the case where the generated keys match exactly, and if the generated dynamic keys do not match, they will be discarded. Since dynamic keys require a fixed length of 128 bits, the KGR should be high. Otherwise, the authentication between users Alice and Bob will spend more time. In this test, we set the distance between Alice and Bob to 2 cm and kept still. As described in Section 4.1, our system can achieve high KGR, up to 5000 bit/s. Compared with previous studies [14, 21], *MobileKey* can increase the KGR by 25× and 33× compared with *TDS* and *ProxiMate*, respectively.

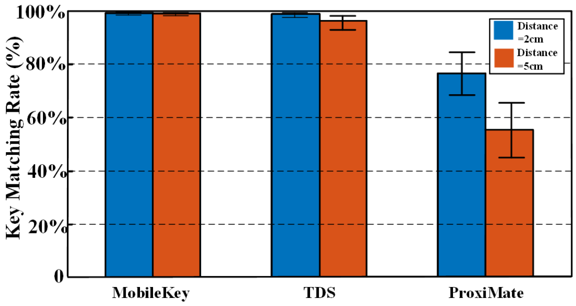


Figure 8: KMR Testing.

4.3 Impact of Window Size

As described in Section 3.2, the size of the hamming window affects the secret key generated by *MobileKey*. To examine the impact of window size, we set the distance between Alice and Bob as 2 cm and calculate the correlation between Alice's CSIs and Bob's CSIs by changing the window sizes from 0 to 64. As shown in Figure 9, the correlation is highest when the window size is 32. This is because if the size is too small, there is not much entropy; however, if the size is too big, the correlation between legitimate users may be decreased because the changes of CSI values in Alice and Bob's channels will be smoothed. Therefore, we choose 32 as size of the hamming window in *MobileKey* to achieve the best performance.

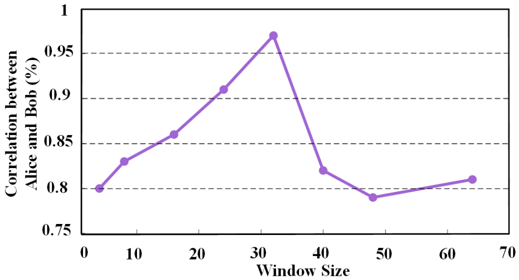


Figure 9: Impact of window size.

4.4 Impact of Mobile Scenarios

Device movement is easily overlooked when evaluating such physical layer-based dynamic key systems. It is an ideal scenario where the devices are all stationary. In a real-world scenario, there is an occlusion caused by an object moving between the router and the device. This obviously results in a signal mismatch due to multipath effects and Doppler phase shift. Some previous studies explain the reasons for this mismatch [21]. Our experiment in this part is that Alice, Bob, and Eve will keep stationary at the beginning, and then all of them move around the router. But the basis is still that Alice and Bob's devices are close, while Bob's physical distance is farther. As shown in Figure 10, we conducted experiments for different moving speeds of the devices, including still, walking, and running. When the stage changed from stationary to walking, the correlation of Alice-Bob dropped from 0.98 to 0.82. We can also observe that the correlation even decreased to 0.57 while running. Also, the value of Alice-Eve decreased from 0.51 to 0.19 when the scenario of walking changed to running, indicating that fast movement has a significant impact on the *MobileKey* compared with the slight impact of the slow movement. To sum up, *MobileKey* is more suitable for the scenarios where the devices are still or move slowly.

4.5 Security Analysis

We use correlation to evaluate whether eavesdropper Eve can obtain CSI sequences similar to legitimate nodes. Correlation shows the correlation coefficient of the CSI amplitudes received by different devices. This parameter can be used to estimate KMR in early feasibility experiments. In this test, the distance between Alice and Bob was 2 cm, and the distance between eavesdroppers and legitimate users was 50 cm. All these three devices are kept still. As shown in Figure 11, the correlation between Alice and Bob is 0.98, much higher than that between legitimate users and attacker.

4.6 Limitations

The experimental results show the promising performance of the proposed *MobileKey* in different scenarios. However, the current system still has certain limitations. Firstly, due to the short wavelength of Wi-Fi signal, the distance between the devices is required

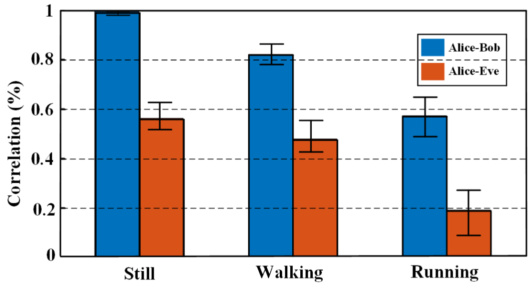


Figure 10: Impact of mobile scenarios.

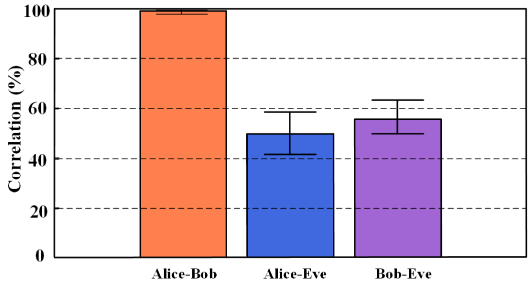


Figure 11: Correlation between every two devices.

to be relatively close. Moreover, the system has only been tested on only one type of phone. It is possible that the network cards of different mobile phones will cause deviations in the received Wi-Fi signal. And due to the external firmware used, this system is difficult to optimize the process of collecting data to reduce power consumption.

5 CONCLUSION

In this paper, we propose the first key generation system for Wi-Fi-enabled mobile devices, *MobileKey*. Our preliminary study shows the feasibility of Wi-Fi-based key generation on mobile devices, and we propose a key generation scheme to generate secret keys in a fast and robust way. The experimental results indicate that *MobileKey* can achieve over 99.1% key matching rate while achieving up to 5000 bit/s key generation rate. Compared with state-of-the-art systems, *MobileKey* improves the key generation rate by 25×.

ACKNOWLEDGMENTS

The work described in this paper was substantially sponsored by the project 62101471 supported by NSFC and was partially supported by the Shenzhen Research Institute, City University of Hong Kong. This paper was partially supported by a grant from the Research Grants Council of the Hong Kong Special Administrative Region, China (Project No. CityU 21201420). The work was partially supported by Shenzhen Science and Technology Funding Fundamental Research Program (Project No. 2021Szvup126), NSF of Shandong Province (Project No. ZR2021LZH010), Changsha International and Regional Science and Technology Cooperation Program (Project No. kh2201023), and a grant from Chow Sang Sang Group Research Fund sponsored by Chow Sang Sang Holdings International Limited (Project No. 9229062).

REFERENCES

[1] Syed Taha Ali, Vijay Sivaraman, and Diethelm Ostry. 2013. Eliminating reconciliation cost in secret key generation for body-worn health monitoring devices. *IEEE Transactions on Mobile Computing* 13, 12 (2013), 2763–2776.

[2] Christian Cachin and Ueli M Maurer. 1997. Linking information reconciliation and privacy amplification. *Journal of Cryptology* 10, 2 (1997), 97–110.

[3] David L Donoho. 2006. Compressed sensing. *IEEE Transactions on Information Theory* (2006).

[4] Jiayao Gao, Weitao Xu, Salil Kanhere, Sanjay Jha, Jun Young Kim, Walter Huang, and Wen Hu. 2021. A novel model-based security scheme for LoRa key generation. In *Proceedings of the 20th International Conference on Information Processing in Sensor Networks (co-located with CPS-IoT Week 2021)*. 47–61.

[5] Xinrui Ge, Jia Yu, Hanlin Zhang, Chengyu Hu, Zengpeng Li, Zhan Qin, and Rong Hao. 2019. Towards achieving keyword search over dynamic encrypted cloud data with symmetric-key based verification. *IEEE Transactions on Dependable and Secure computing* 18, 1 (2019), 490–504.

[6] Francesco Gringoli, Matthias Schulz, Jakob Link, and Matthias Hollick. 2019. Free your CSI: A channel state information extraction platform for modern Wi-Fi chipsets. In *Proceedings of the 13th International Workshop on Wireless Network Testbeds, Experimental Evaluation & Characterization*. 21–28.

[7] 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.

[8] Amer A Hassan, Wayne E Stark, John E Hershey, and Sandeep Chennakeshu. 1996. Cryptographic key agreement for mobile radio. *Digital Signal Processing* 6, 4 (1996), 207–212.

[9] 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.

[10] Jun Young Kim, Ralph Holz, Wen Hu, and Sanjay Jha. 2017. Automated analysis of secure internet of things protocols. In *Proceedings of the 33rd Annual Computer Security Applications Conference*. 238–249.

[11] 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.

[12] Hongbo Liu, Yang Wang, Jie Yang, and Yingying Chen. 2013. Fast and practical secret key extraction by exploiting channel response. In *2013 Proceedings IEEE INFOCOM*. IEEE, 3048–3056.

[13] Hongbo Liu, Jie Yang, Yan Wang, and Yingying Chen. 2012. Collaborative secret key extraction leveraging received signal strength in mobile wireless networks. In *2012 Proceedings IEEE Infocom*. IEEE, 927–935.

[14] Suhas Mathur, Robert Miller, Alexander Varshavsky, Wade Trappe, and Narayan Mandayam. 2011. Proximate: proximity-based secure pairing using ambient wireless signals. In *Proceedings of the 9th international conference on Mobile systems, applications, and services*. 211–224.

[15] 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 14th ACM international conference on Mobile computing and networking*. 128–139.

[16] Ueli M Maurer. 1993. Secret key agreement by public discussion from common information. *IEEE transactions on information theory* 39, 3 (1993), 733–742.

[17] Girish Revadigar, Chitra Javali, Wen Hu, and Sanjay Jha. 2015. DLINK: Dual link based radio frequency fingerprinting for wearable devices. (2015), 329–337.

[18] 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.

[19] Matthias Schulz, Jakob Link, Francesco Gringoli, and Matthias Hollick. 2018. Shadow Wi-Fi: Teaching smartphones to transmit raw signals and to extract channel state information to implement practical covert channels over Wi-Fi. In *Proceedings of the 16th Annual International Conference on Mobile Systems, Applications, and Services*. 256–268.

[20] Kyung-Ah Shim. 2015. A survey of public-key cryptographic primitives in wireless sensor networks. *IEEE Communications Surveys and Tutorials* (2015).

[21] 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 2016 ACM SIGSAC Conference on Computer and Communications Security*. 616–627.

[22] 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.

[23] Weitao Xu, Junqing Zhang, Shunqi Huang, Chengwen Luo, and Wei Li. 2021. Key Generation for Internet of Things: A Contemporary Survey. *ACM Computing Surveys (CSUR)* 54, 1 (2021), 1–37.

[24] Kai Zeng, Daniel Wu, An Chan, and Prasant Mohapatra. 2010. Exploiting multiple-antenna diversity for shared secret key generation in wireless networks. (2010), 1–9.