Channel-Envelope Differencing Eliminates Secret Key Correlation: LoRa-Based Key Generation in Low Power Wide Area Networks

Junqing Zhang, Alan Marshall, *Senior Member, IEEE*, and Lajos Hanzo, *Fellow, IEEE*

***Abstract*—This paper presents automatic key generation for long-range wireless communications in low power wide area networks (LPWANs), employing LoRa as a case study. Differential quantization is adopted to extract a high level of randomness. Experiments conducted both in an outdoor urban environment and in an indoor environment demonstrate that this key generation technique is applicable for LPWANs, and shows that it is able to reliably generate secure keys.**

***Index Terms*—Internet of Things, low power wide area networks, physical layer security, key generation, LoRa/LoRaWAN**

I. Introduction

The Internet of Things (IoT) is capable of connecting people, things, and the environment. This revolution heavily relies on secure data communications, which are currently maintained by classic cryptographic algorithms and protocols. In particular, public key cryptography (PKC) has been the de facto scheme for distributing keys to the users in modern communication and computer networks. However, its application in the IoT remains a challenge owing to the limited computational and battery capacity, as well as the requirement of a public key infrastructure for distributing the public keys.

Key generation from the wireless channel between any pair of users has become a promising design alternative to complement PKC. The keys generated can be used for the symmetric encryption schemes in different layers of the protocol stack, e.g., the Wi-Fi Protected Access (WPA) for the Wi-Fi MAC layer encryption or for Transport Layer Security (TLS) in the transport layer. It is particularly for protecting IoT systems that contain large numbers of resource-limited devices [1]. A comparison of resource and energy consumption between the key generation and elliptic curve-based Diffie-Hellman (ECDH) procedure, which is a popular PKC scheme, has been carried out in [2]. Key generation has been demonstrated to be more cost-efficient. Explicitly, ECDH consumes 98 times more energy and imposes 1289 times higher complexity than key generation, when both are implemented by an 8-bit Intel MCS-51 micro-controller [2]. In addition, key generation does not require any assistance from a third party, which is suitable for many decentralized or device-to-device IoT applications.

The received signal strength indicator (RSSI) has been the most popular parameter because of its wide availability in the transceivers and network interface cards. This has been evidenced by its wide applications in Wi-Fi [3]–[6], ZigBee [7], [8] and Bluetooth [9], etc. However, all these wireless techniques only support operations in short-range environments, typically within 100 meters. The channel may be deemed reciprocal in such environments. For example, we carried out key generation for Wi-Fi in an indoor office scenario [5]. The RSSI varied from -50 dBm to -25 dBm and random keys were generated from the reciprocal measurements.

In reality, many IoT applications operate in longer-range environments, e.g., vehicular communications. There have been several long range standards designed for low-power wide area networks (LPWANs), including LoRa, Narrowband IoT, and Sigfox, etc. A very recent conference contribution applies key generation with LoRa [10], but the experiments are carried out in short-range environments, since the received power only has a 20 dBm variation. In contrast to short-range wireless communications, the channel conditions in long-range networks may vary significantly due to the shadowing of buildings in urban environments.

This paper investigates the key generation in LPWANs with long range communications, by employing LoRa as a case study. Our work observed large RSSI variations of the devices, and used differential quantization to extract the channel's randomness. Experiments have been carried out both in an outdoor urban environment and in an indoor environment. The system is shown to exhibit beneficial channel reciprocity as confirmed in terms of cross-correlation and key disagreement ratio (KDR), and a sufficiently high degree of randomness.

II. Overview of LoRa/LoRaWAN

LoRa is a physical layer modulation technique developed by Semtech while LoRaWAN is the MAC protocol maintained by the LoRa Alliance [11]. This section briefly introduces the relevant background and a detailed introduction can be found in [12].

A. Physical Layer

LoRa uses chirp spread spectrum (CSS) modulation, which is immune to multipath and Doppler shift. It is quite robust

Manuscript received xxx; revised xxx; accepted xxx. Date of publication xxx; date of current version xxx. L. Hanzo would like to gratefully acknowledge the ERC's financial support of his Advanced Fellow Grant. The review of this paper was coordinated by xxx.

J. Zhang and A. Marshall are with the Department of Electrical Engineering and Electronics, University of Liverpool, Liverpool, L69 3GJ, U.K. (emails: Junqing.Zhang@liverpool.ac.uk; Alan.Marshall@liverpool.ac.uk)

L. Hanzo is with the School of ECS, University of Southampton, Southampton SO17 1BJ, U.K. (email: lh@ecs.soton.ac.uk)

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

Digital Object Identifier xxx

and achieves a receive sensitivity as low as -148 dBm, which is eminently suitable for long range communications. The main parameters include bandwidth, spreading factor and code rate, which can be adjusted according to the specific requirements of sensitivity, communication range, and data rate.

B. MAC Layer

LoRaWAN relies on a star network topology involving gateways and end devices. According to the different configurations of the receive windows at the end device, there are three device types, namely Class A, B, and C. The Class A functionality is mandatory, which is explained in this paper.

The Class A end device can initiate the uplink transmission. It will then open two receive windows after a certain delay. In other words, the gateway can only send a downlink frame to the end device, provided that it receives an uplink frame. The power consumption of the end device is thus kept very low. LoRaWAN also defines the so-called confirmed data message type, which must be acknowledged by the receiver. The confirmed data message and its ACK message constitute a pair of bidirectional transmissions, which can be leveraged for key generation.

C. LoRaWAN Security Mechanism

LoRaWAN has a rigorous security mechanism for protecting both the application payload and the communication sessions. The encryption algorithm is based on the one used in IEEE 802.15.4, which employs advanced encryption standard (AES) with a key length of 128 bits. LoRaWAN defines two activation methods, namely activation by personalization (ABP) and over-the-air activation (OTAA). In the ABP, the session keys are programmed into the end devices during manufacturing, which cannot be updated. In the OTAA, the session keys are generated from the device's root keys, including AppKey and NwkKey. However, similar to other symmetric encryption schemes, the distribution technique of the device's root keys is not defined in the standard. Inspired by this, we will propose an innovative key generation scheme by exploiting the unpredictable features of the wireless channel between any pair of devices.

III. Key Generation Protocol

A full key generation protocol usually includes channel probing, quantization, information reconciliation, and privacy amplification [1]. A pair of legitimate users, Alice and Bob, will carry out channel probing by performing bidirectional channel measurements. Once sufficient results are collected, they will separately convert the analog measurements into binary sequences using a quantizer. Since there may be mismatch between the keys at Alice and Bob due to noise and asynchronous sampling, information reconciliation is adopted for allowing them to agree on the same keys. Finally, privacy amplification is employed to remove the information leakage. These four steps will be discussed in detail as follows.

Channel probing harvests the randomness from the wireless channel. During the i^{th} probe, Alice sends a packet to Bob who will measure the RSSI $X_B(i)$. Upon receiving it, Bob will reply a message to Alice, who will measure the RSSI $X_A(i)$. Alice and Bob will keep these bidirectional transmissions until they collect sufficient data. An example of the RSSI of Alice and Bob collected from an outdoor experiment is shown in Fig. 1, and the detailed setup will be discussed in Section IV. It is worth noting that RSSI measurements can be carried out during regular data transmissions and no dedicated packet exchange will be required.

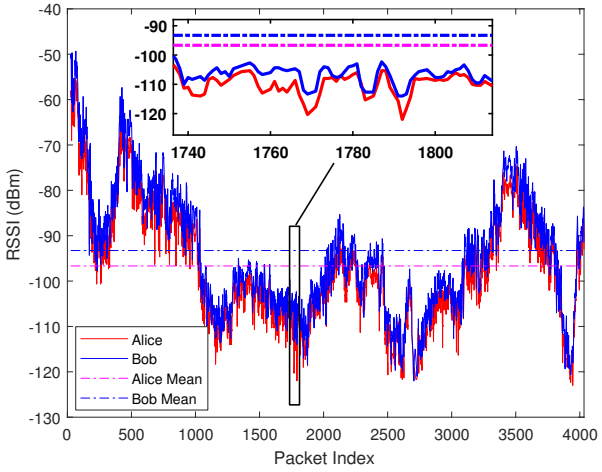

Fig. 1. RSSI of Alice and Bob of experiment carried out within the campus of University of Liverpool.

Quantization in key generation discretizes the analog measurements into a binary sequence, which works in a similar manner to the classic analog-to-digital converter (ADC). Absolute value-based quantization is commonly used for comparing measurements to thresholds and then assigning binary values to the outcome. For example, the mean value-based quantization will assign a 1 to any data above the mean value and a 0 to any data below the mean value. However, the RSSI output of Fig. 1 varies from -123 dBm to -49 dBm, which is quite a large variation. There are many consecutive samples above/below the mean value. Hence, the mean value-based quantizer will result in long runs of continuous 1s and 0s, which will not be random at all. Owing to this impediment, it is not adopted in this paper. This scheme may be improved by first partitioning the measurements into smaller blocks and then quantizing each block separately, as in the adaptive secret bit generation of [3]. However, due to the large variation of RSSI output in LoRa measurements, it is challenging to determine the block size.

We propose to carry out the quantization based on the differential value, namely the difference between adjacent values, as shown in Algorithm 1. The differential quantization concept was originally proposed in [4]. For each user u, $u = \{A, B\}$, it will carry out the quantization separately. Whenever a new RSSI, $X_u(i+1)$, is measured, the user u will compare it to the previous one, $X_u(i)$, and assign $K_u(i)$ as 1/0, when it is larger/smaller than the previous RSSI. The RSSI measurement may not be very accurate because of using low cost hardware, hence the RSSI resolution ϵ is introduced. The RSSI values having variation smaller than ϵ are thus dropped in order to improve the robustness against the measurement

Algorithm 1 Differential-based quantization algorithm

INPUT: X_u % RSSI of user u
INPUT: ϵ % RSSI resolution
OUTPUT: K_u % Generated key sequence of user u

```
for i ← 1 to N − 1 do
    if X_u(i + 1) > X_u(i) + ε then
        K_u(i) = 1
    else if X_u(i + 1) < X_u(i) − ε then
        K_u(i) = 0
    else
        X_u(i) dropped
    end if
end for
```

Algorithm 2 Information reconciliation - secure sketch

INPUT: K_A, K_B % Quantized keys of Alice and Bob
INPUT: C % ECC set shared by Alice and Bob
OUTPUT: K_A, $K_{B'}$ % Reconciled key

1: Alice randomly selects a code c from an ECC set C
2: Alice calculates $s = \text{XOR}(K_A, c)$
3: Alice transmits s to Bob through a public channel
4: Bob receives s
5: Bob calculates $c_B = \text{XOR}(K_B, s)$
6: Bob decodes c_B to get c % When $dis(c - c_B) < t$
7: Bob calculates $K_{B'} = \text{XOR}(c, s) = K_A$ % Alice and Bob agree on the same key

imperfection. As each packet has a unique packet sequence index, the index of the dropped RSSI values is shared between Alice and Bob, so that they can maintain a common index.

Compared to the absolute value-based quantization, differential quantization captures the relative changes of the RSSI values and the channel conditions. This is beneficial because it produces the key bits based on the comparison between the adjacent measurements, which does not require any adaptive adjustment based on the channel conditions. It is therefore much more lightweight for implementation.

Alice and Bob will respectively produce K_A and K_B after quantization. However, as shown in Fig. 1, the channel measurements X_A and X_B are not identical, which results in disagreement between K_A and K_B. Information reconciliation is thus employed to correct the disagreement. Secure sketch is one of the popular protocols [13], as shown in Algorithm 2. It exploits the correction capability of error correction codes (ECCs) [14], e.g., BCH, LDPC, etc. The ECC has a maximum error correction capability of t errors. When the key disagreement, quantified by the Hamming distance, is lower than t, it can be corrected. Finally, because there is information exchanged publicly during the information reconciliation, privacy amplification, e.g., by employing hash function, is used to remove the information leakage.

IV. Experimental Evaluation

A. Setup

A testbed was built using Arduino Uno and LoRa/GPS Shield that uses Semtech SX1276 as the LoRa transceiver. The

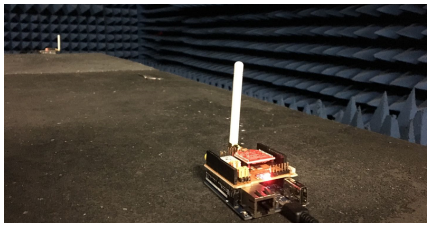

Fig. 2. The placement of Alice and Bob in the anechoic chamber, University of Liverpool.

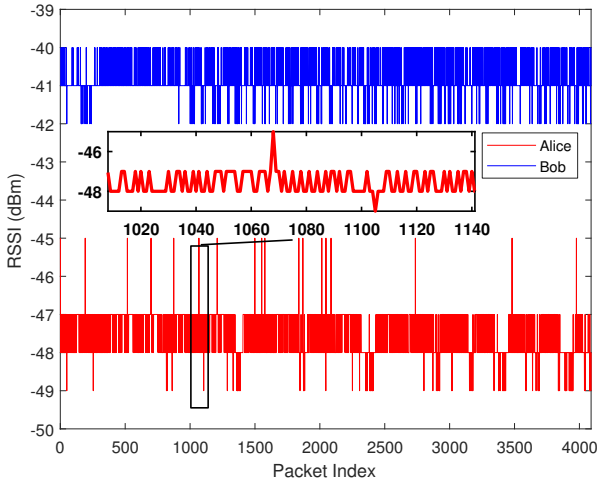

Fig. 3. RSSI of Alice and Bob of experiment carried out in the anechoic chamber, University of Liverpool.

RadioHead library [15] is used, which provides the function to obtain the packet's RSSI. RSSI has been used extensively in key generation to represent the link quality, and is also used in this paper. Two LoRa modules, termed as Alice and Bob, are configured with the same parameters, including carrier frequency of 868.1 MHz, bandwidth of 125 kHz, transmission power of 13 dBm and spreading factor of 7. These two modules will carry out bidirectional channel measurements, as introduced in Section III. The RSSI values are transferred to the PC via a serial port and further processed by Matlab.

Even when there is no channel variation or interference, the received power may fluctuate because of the imperfect hardware characteristics. In order to quantify the resolution of RSSI, we carried out a calibration experiment in an anechoic chamber at the University of Liverpool. As shown in Fig. 3, two LoRa devices were placed about two meters apart, which is a totally static and line-of-sight (LoS) scenario with no interference from other networks. The experiment ran for about 15 minutes and collected 4000 packets at each side. The RSSI of Alice and Bob is shown in Fig. 3. As can be observed, while there are some spikes in Alice's RSSI, most of the RSSI values of Alice and Bob only have a 2 dBm variation. Therefore, we set $\epsilon = 2$ for the differential quantization.

We then carried out two tests. Alice was placed in an indoor office of the second floor of the Department of Electrical Engineering and Electronics building (EEE), University of Liverpool, the green point in Fig. 4. In the outdoor experiment, Bob was moving at a walking speed, i.e., about 2 meters

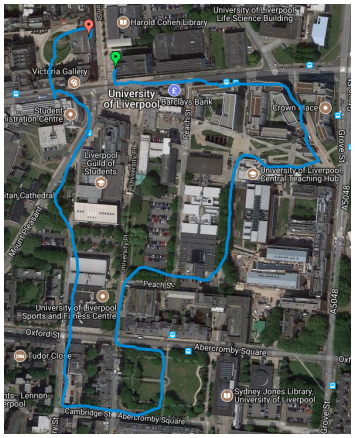

Fig. 4. The trajectory of Bob in the campus of University of Liverpool.

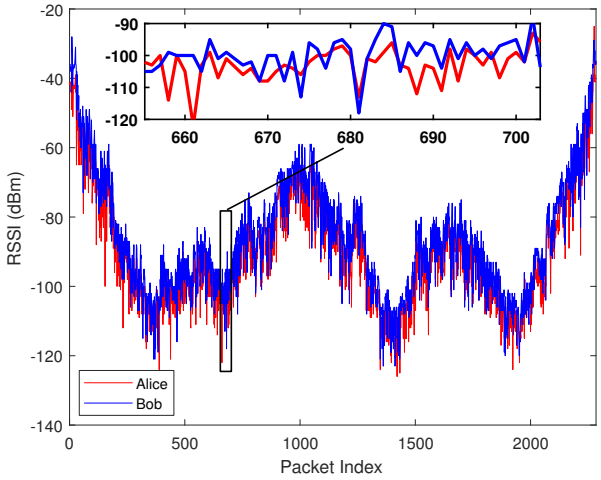

Fig. 5. RSSI of Alice and Bob of experiment carried out inside the building of EEE Department, University of Liverpool.

per second, in the campus of University of Liverpool. Bob moved from the green point to the red point with the detailed trajectory shown in Fig. 4. This is a typical urban environment with many buildings causing severe path loss and shadowing. The farthest distance between Alice and Bob in the experiment was about 500 meters. The experiment lasted 21 minutes and collected about 4000 packets in total. In the second (indoor) experiment, Bob was moving inside the six-storey EEE Department building up and down. This is a typical indoor environment with rich multipath. The indoor experiment lasted 10 minutes and collected about 2300 packets.

B. Results

The RSSI of the outdoor urban and indoor experiments is shown in Fig. 1 and Fig. 5, respectively. Both have very large variations. We use Pearson's cross-correlation coefficient and KDR to characterize the channel reciprocity, and randomness test to evaluate the quality of the key sequence [1]. Cross-correlation describes the similarity between any two signals while KDR measures the ratio of different bits between two sequences. Randomness of the key determines the security level because a non-random key will be subject to brute force attacks.

Pearson's cross-correlation coefficient is defined as

$$\rho = \frac{\sum_{i=1}^{N}(X_A(i)-\mu_{X_A})(X_B(i)-\mu_{X_B})}{\sqrt{\sum_{i=1}^{N}(X_A(i)-\mu_{X_A})^2}\sqrt{\sum_{i=1}^{N}(X_B(i)-\mu_{X_B})^2}}, \tag{1}$$

where μ_{X_u} is the mean value of X_u. The KDR is defined as

$$KDR = \frac{\sum_{i=1}^{l_k}|K_A(i)-K_B(i)|}{l_k}, \tag{2}$$

where l_k is the length of the key sequence. The correlation coefficients in the outdoor and indoor experiments are 0.9582 and 0.9689, respectively. The correlation coefficients are high, which indicates a high grade of reciprocity. The KDRs in the outdoor and indoor experiments are 0.0529 and 0.0399, respectively. The KDR is quite low when differential quantization is used. As we analyzed in our previous work [16], a BCH (n,k,t) code can correct t/n mismatch, e.g., a BCH (15,3,3) can correct 20% mismatch. The KDR in this paper is thus well within a classic code's correction capability.

TABLE I
RANDOMNESS TEST RESULTS

	Outdoor	Indoor
Sequence Length	397	376
Frequency	0.515	0.537
Block frequency	0.905	0.677
Runs	0.023	0.343
Longest run of 1s	0.887	0.331
DFT	0.792	0.85
Serial	0.3, 0.76	0.048, 0.048
Appro. entropy	0.065	0.1
Cum. sums (fwd)	0.538	0.432
Cum. sums (rev)	0.896	0.919

The National Institute of Standards and Technology (NIST) randomness test suite is the most popular tool for evaluating the randomness of the true/pseudo random number generator [17]. It has been widely applied in key generation research and it is also adopted in this paper for evaluating the randomness of the key sequence generated. Each test will return a p-value and when this is bigger than a threshold, e.g., 0.01 in this paper, the sequence passes the test. The randomness test results of the quantized key sequence is shown in Table I and the key sequence passes all the tests.

V. KEY GENERATION WITH LORAWAN

We have applied key generation relying on LoRa in the previous sections. While we can exploit the transmissions between the LoRaWAN gateway and end devices for key generation, there are special features and configurations in LoRaWAN, which require further careful considerations.

The LoRaWAN standard supports up to 16 channels in total. For example, The Things Network, a global IoT network hosting thousands of LoRaWAN gateways, defines eight frequencies [18]. In order to decrease the interference, the end device changes the carrier frequency in a pseudo-random fashion for every transmission. This pseudo-random frequency hopping is detrimental to key generation, because the channel conditions of different frequencies are not reciprocal.

Fortunately, LoRaWAN also specifies that the downlink ACK frame should be at the same frequency as that of the corresponding uplink data frame [11]. The end device first transmits an uplink confirmed data packet to the gateway. Upon receiving the data packet, the gateway will respond with a downlink ACK frame. The carrier frequencies of these two packets should be the same. Therefore, the key generation probing process can be carried out using the confirmed uplink data frame and downlink ACK frame pairs.

VI. Discussion

The keys generated can be used for any cryptographic schemes, which require a symmetric key. These schemes do not require a fast key update rate. For example, Wi-Fi recommends changing the keys, a 128-bit binary sequence, every hour. In our scheme, each differential comparison will produce one bit. Hence the key generation rate is less than or equal to 1 bit per measurement, which should be sufficiently fast to generate keys at the required rate.

Key generation requires a dynamically fluctuating channel in order to produce random keys. When users are stationary, the channel variation is introduced by objects moving in the environment. Even in a totally static environment, multiple antennas and frequency diversity can be exploited [19]. However, in many of the smart city and intelligent transportation systems, user movements and channel variations are indeed present as a much needed source of channel randomness.

Key generation is subject to passive eavesdropping [6], [8], where the eavesdroppers record all the transmissions and try to crack the system. According to communication theory, when the eavesdropper is located sufficiently far from the legitimate users, the eavesdropping channel is uncorrelated with the legitimate channel. The seminal work in [20], [21] has laid the information-theoretical foundation for key generation, which formulates the secret key capacity as

$$C_{sk} \geq I(X_A; X_B) - \min[I(X_A; X_E), I(X_B; X_E)], \quad (3)$$

where X_A, X_B, and X_E are the observations of Alice, Bob, and Eve, respectively. When $C_{sk} > 0$, key generation can be carried out securely. Key generation security under passive eavesdropping has been validated in [8] through extensive ZigBee-based experiments. In particular, the work in [10] demonstrates that legitimate users can still generate keys securely by using LoRa, when the eavesdropper is located only 0.15 m or 2 m away.

VII. Conclusions and Future Work

This paper applies key generation with the LoRa technology and investigates automatic key generation performance in a long-range environment. Because of the large variation of the channel conditions and RSSI values, we employed differential quantization to extract the channel's randomness. We carried out experiments both in an outdoor urban environment and in an indoor environment to demonstrate that LoRa-based key generation has a good performance. Key generation application in LoRaWAN was also discussed and was shown to be feasible by leveraging the uplink confirmed data packet and downlink ACK packet.

References

[1] J. Zhang, T. Q. Duong, A. Marshall, and R. Woods, "Key generation from wireless channels: A review," *IEEE Access*, vol. 4, pp. 614–626, Mar. 2016.

[2] C. T. Zenger, M. Pietersz, J. Zimmer, J.-F. Posielek, T. Lenze, and C. Paar, "Authenticated key establishment for low-resource devices exploiting correlated random channels," *Computer Networks*, vol. 109, pp. 105–123, 2016.

[3] S. Jana, S. N. Premnath, M. Clark, S. K. Kasera, N. Patwari, and S. V. Krishnamurthy, "On the effectiveness of secret key extraction from wireless signal strength in real environments," in *Proc. 15th Annu. Int. Conf. Mobile Computing and Networking (MobiCom)*, Beijing, China, Sep. 2009, pp. 321–332.

[4] B. Zan, M. Gruteser, and F. Hu, "Key agreement algorithms for vehicular communication networks based on reciprocity and diversity theorems," *IEEE Trans. Veh. Technol.*, vol. 62, no. 8, pp. 4020–4027, 2013.

[5] J. Zhang, R. Woods, T. Q. Duong, A. Marshall, and Y. Ding, "Experimental study on channel reciprocity in wireless key generation," in *Proc. 17th IEEE Int. Workshop Signal Process. Advances in Wireless Commun. (SPAWC)*, Edinburgh, UK, Jul. 2016, pp. 1–5.

[6] J. Zhang, R. Woods, T. Q. Duong, A. Marshall, Y. Ding, Y. Huang, and Q. Xu, "Experimental study on key generation for physical layer security in wireless communications," *IEEE Access*, vol. 4, pp. 4464–4477, 2016.

[7] O. Gungor, F. Chen, and C. Koksal, "Secret key generation via localization and mobility," *IEEE Trans. Veh. Technol.*, vol. 64, no. 6, pp. 2214–2230, Jun. 2015.

[8] C. Zenger, H. Vogt, J. Zimmer, A. Sezgin, and C. Paar, "The passive eavesdropper affects my channel: Secret-key rates under real-world conditions," in *Proc. IEEE Globecom TCPLS Workshops*, Washington DC, USA, Dec. 2016, pp. 1–6.

[9] S. N. Premnath, P. L. Gowda, S. K. Kasera, N. Patwari, and R. Ricci, "Secret key extraction using Bluetooth wireless signal strength measurements," in *Proc. 11th Annu. IEEE Int. Conf. Sensing, Commun. Networking (SECON)*, Singapore, Jun. 2014, pp. 293–301.

[10] H. Ruotsalainen and S. Grebeniuk, "Towards wireless secret key agreement with lora physical layer," in *Proc. Int. Conf. Availability, Reliability and Security*, Hamburg, Germany, Aug. 2018, p. 23.

[11] *LoRaWAN™ 1.1 Specification*, Std., accessed on 3rd June, 2018. [Online]. Available: https://lora-alliance.org/resource-hub/lorawantm-specification-v11

[12] A. Augustin, J. Yi, T. Clausen, and W. M. Townsley, "A study of LoRa: Long range & low power networks for the Internet of Things," *Sensors*, vol. 16, no. 9, p. 1466, 2016.

[13] Y. Dodis, R. Ostrovsky, L. Reyzin, and A. Smith, "Fuzzy extractors: How to generate strong keys from biometrics and other noisy data," *SIAM J. Comput.*, vol. 38, no. 1, pp. 97–139, 2008.

[14] L. Hanzo, T. H. Liew, and B. L. Yeap, *Turbo Coding, Turbo Equalisation and Space-Time Coding*. John Wiley & Sons, 2002.

[15] RadioHead Packet Radio library for embedded microprocessors. Accessed on 3rd June, 2018. [Online]. Available: http://www.airspayce.com/mikem/arduino/RadioHead/classRH_RF95.html

[16] J. Zhang, A. Marshall, R. Woods, and T. Q. Duong, "Efficient key generation by exploiting randomness from channel responses of individual OFDM subcarriers," *IEEE Trans. Commun.*, vol. 64, no. 6, pp. 2578–2588, Jun. 2016.

[17] A. Rukhin *et al.*, "A statistical test suite for random and pseudorandom number generators for cryptographic applications," National Institute of Standards and Technology, Tech. Rep. Special Publication 800-22 Revision 1a, Apr. 2010.

[18] LoRaWAN frequencies overview. Accessed on 3rd June, 2018. [Online]. Available: https://www.thethingsnetwork.org/docs/lorawan/frequency-plans.html

[19] P. Huang and X. Wang, "Fast secret key generation in static wireless networks: A virtual channel approach," in *Proc. IEEE Int. Conf. Computer Communications*, Turin, Italy, Apr. 2013, pp. 2292–2300.

[20] R. Ahlswede and I. Csiszar, "Common randomness in information theory and cryptography – Part I: secret sharing," *IEEE Trans. Inf. Theory*, vol. 39, no. 4, pp. 1121–1132, 1993.

[21] U. M. Maurer, "Secret key agreement by public discussion from common information," *IEEE Trans. Inf. Theory*, vol. 39, no. 3, pp. 733–742, May 1993.