A New Frontier for IoT Security Emerging From Three Decades of Key Generation Relying on Wireless Channels

Junqing Zhang, Guyue Li, Member, IEEE, Alan Marshall, Senior Member, IEEE, Aiqun Hu, Member, IEEE, and Lajos Hanzo, Fellow, IEEE

Abstract—The Internet of Things (IoT) is a transformative technology, which is revolutionizing our everyday life by connecting everyone and everything together. The massive number of devices are preferably connected wirelessly because of the easy installment and flexible deployment. However, the broadcast nature of the wireless medium makes the information accessible to everyone including malicious users, which should hence be protected by encryption. Unfortunately the secure and efficient provision of cryptographic keys for low-cost IoT devices is challenging; weak keys have resulted in severe security breaches, as evidenced by numerous notorious cyberattacks. This paper provides a comprehensive survey of lightweight security solutions conceived for IoT, relying on key generation from wireless channels. We first introduce the key generation fundamentals and protocols. We then examine how to apply this emerging technique to secure IoT and demonstrate that key generation relying on the randomness of wireless channels is eminently suitable for IoT. This paper reviews the extensive research efforts in the areas of theoretical modelling, simulation based validation and experimental exploration. We finally discuss the hurdles and challenges that key generation is facing and suggest future work to make key generation a reliable and secure solution to safeguard the IoT.

Index Terms—Internet of Things, wireless security, physical layer security, key generation

I. INTRODUCTION

A. Motivation

The Internet of Things (IoT) integrates people, things and the environment. As illustrated in Fig. 1, IoT will transform our daily life with the aid of exciting new applications, including smart homes, e-commerce, connected healthcare and smart cities, to name but a few [1], [2]. Hence the IoT has attracted massive research and development interests from both academia and industry, given its significant impact on the economy and society. McKinsey estimated that by 2025 there would be 25 billion to 50 billion devices and the potential economic impact would be in the range of $3.9 to $11.1 trillion per year [3].

There are many tiny low-cost devices in IoT applications, e.g. sensor nodes, Fitbit and implantable medical devices. They are usually powered by batteries, which may be difficult to replace. For example, many Long Range (LoRa) sensor nodes are designed to work for ten years with two AAA batteries. The limited size and power supply facilitate to provide “just” sufficient computational resources and storage spaces. IoT design has hence been mainly focused on reducing energy and computational cost as well as improving the hardware efficiency. However, the security issues of the IoT have often been overlooked, treated as something “nice to have” rather than “must to have”.

On the other hand, the data transmitted in IoT applications can be sensitive, private and confidential, hence IoT security has significant societal and economic impacts. Healthcare devices and the data they generate are vital and private. For example, implantable devices such as a pacemaker are vital to patients’ life and their health related data such as their heart rate is very private. Financial data should also be protected to the highest possible standard. Therefore, IoT security has been brought to the spotlight and has stimulated substantial research efforts [4]–[6].

Despite these efforts, there are still numerous security flaws and vulnerabilities [7], as evidenced by many notorious cyberattacks. Researchers have successfully hacked the latest
generation of implantable medical devices using the widespread wireless devices referred to as the universal software radio peripheral (USRP) [8]. The automotive remote keyless entry system has also been cracked by very low-cost wireless modules ($40), which exposes millions of cars to risk [9]. An implementation bug was found in the WiFi protected access (WPA) 2 [10], namely in the well-known WiFi encryption protocol, which affected almost everyone using smartphones and laptops.

In summary, the IoT is far from being secure, which is a major bottleneck on the road to trustworthy IoT applications. Numerous challenges arise because of the limited computational resources and energy supply. Hence, more research efforts should be invested into designing optimized security primitives, which are capable of providing tailored security for IoT applications.

B. Wireless Security

1) A Brief History of Information Security: Information security can be mainly achieved by two approaches, namely modern cryptography and physical layer security. Cryptography protects information using mathematical algorithms and protocols. On the other hand, physical layer security techniques achieve information-theoretical security by exploiting the unpredictable features of the fading channel. These techniques are summarized in Table I and will be introduced in detail.

Providing information security dates back to as early as 1919, when Venman proposed the “one-time pad” encryption of each message bit, by performing exclusive-OR operation with different and truly random key bits [11]. In 1949, Shannon established the concept of perfect secrecy [12]. When the amount of information conveyed by the key sequence is higher than the information carried by the message, $M$, the message can be encoded into a codeword, $C$, which does not reveal any information about the message. This is formulated as

$$\mathbb{H}(M|C) = \mathbb{H}(M),$$

(1)

where $\mathbb{H}(\cdot)$ represents the entropy. However, because the keys cannot be reused at all, it is extremely challenging to provide a sufficiently high number of keys in an efficient manner.

Physical layer security research is pioneered by Wyner who presented his seminal work by designing the wiretap channel model in 1975 [13]. It is capable of achieving perfect secrecy without encrypting messages for transmission over a discrete memoryless channel, when the channel capacity of the legitimate channel is higher than that of the eavesdropping channel. His theory was then extended to the Gaussian wiretap channel in 1978 and the notion of secrecy capacity was defined [14]. Because no encryption is involved, these techniques are termed as keyless security in [15] and not affected by the computational capability of attackers. Wyner’s seminal work has inspired significant research efforts, dedicated to ensuring that the quality of the legitimate channel remains better than that of the eavesdropping channel (see [15]–[20] and references therein). This can be achieved for example by using artificial noise [21]–[24], beamforming [25]–[28] and on-off secure transmission [29]. However, keyless secure transmission usually requires complex code design and accurate channel state information (CSI) that may not be available. Additionally, having a better legitimate channel cannot always be guaranteed. Hence its practical applications remain rather limited at the time of writing.

As another design alternative, computational security achieved by modern cryptography has been one of the dominant information security solutions since the conception of the famous Diffie Hellman key exchange protocol in 1976 [30]. Cryptography does not achieve perfect secrecy, but it is capable of securing the information against attacks by using complex mathematical manipulations. Hence it is also often termed as computational security. Since cryptography imposes moderate complexity, it has become the de facto solution of securing information transmission. Depending on whether the two users have a pair of different keys or the same key, computational security based schemes are termed as asymmetric and symmetric encryption [31]. In asymmetric encryption schemes, the parties have a pair of different public and private keys. The associated protocols are also known as public key cryptography (PKC). Relying on concepts inherited from number theory, such as discrete logarithm and integer factorization, PKC is eminently suitable for encryption such as Rivest–Shamir–Adleman (RSA) algorithm, key distribution such as Diffie Hellman key exchange and digital signatures such as ElGamal cryptosystem [31]. On the other hand, symmetric encryption schemes require the same key at both parties for encryption and decryption. The popular symmetric schemes include the RC4 (Section 7.5 of [31]), Data Encryption Standard (DES) [32] and the Advanced Encryption Standard (AES) [33], etc.

A classic encryption system is illustrated in Fig. 2, which includes key distribution by PKC and symmetric encryption. The public key infrastructure (PKI) first distributes the same public key to a pair of legitimate users, Alice and Bob. Alice and Bob have different private keys and they will be able to get the same session key based on some complex mathematical operations. The key is then used for the symmetric encryption to secure the transmissions.

1RC4 and DES have been cracked, hence they are not used any more.
TABLE I

| Security Route                      | Technique                          | Feature                                                                 | Algorithm |
|-------------------------------------|------------------------------------|-------------------------------------------------------------------------|-----------|
| Cryptography                        | Asymmetric Encryption              | Use same public key but different private keys; based on mathematical problems | Encryption: RSA  
Key distribution: Diffie Hellman key exchange  
Digital signature: ElGamal cryptosystem |
| Symmetric Encryption                |                                    | use the identical key at both users.                                    | DES, RC4, AES |
| Physical Layer Security             | Keyless Security                   | Confidential wireless transmission by employing the channel advantage    | Beamforming, artificial noise |
| Transmission                        |                                    | Automatic key generation by leveraging unpredictable wireless fading      | A four-stage protocol |

Fig. 3. Security mechanism in the LoRaWAN protocol. AES is used to encrypt the network and application sessions. However, how to distribute the root keys, namely AppKey and NwkKey, is missing.

Fig. 4. The number of IoT malwares. Data source: Kaspersky Lab [45]

2) Challenges for IoT Security: Since there is a huge number of IoT devices, wireless links are preferred for connecting these devices because of their convenient installation. There are many wireless IoT techniques, relying on cellular, IEEE 802.11/WiFi [34], IEEE 802.15.4 [35], Bluetooth [36], LoRa/LoRaWAN [37], Narrowband IoT (NB-IoT) [38], Sigfox [39] solutions, to name but a few.

The broadcast nature of wireless communications however exposes the information to all users within the communication range. Encryption is thus vital for ensuring the message confidentiality and integrity. In particular, AES has been included in many IoT standards such as WiFi, IEEE 802.15.4, Bluetooth and LoRaWAN.

Taking LoRaWAN as an example. The latest LoRaWAN specification v1.1 [37] has defined a rigid security mechanism, as portrayed in Fig. 3. The end devices will be configured with the same network key, NwkKey and the same application key, AppKey, for the network and applications servers, respectively. These keys are used for generating the network session key and application session key to encrypt the payload using AES. While the LoRaWAN specification has explicitly defined the encryption mechanisms, unfortunately it does not specify how to securely provide the cryptographic keys, namely NwkKey and AppKey. The LoRaWAN 1.1 specification states “secure provisioning, storage and usage of root keys NwkKey and AppKey on the end-device and the backend are intrinsic to the overall security of the solution. These are left to implementation and out of scope of this document.” (page 48 of [37])

Similar to LoRaWAN, other IoT standards also refrain from specifying how to distribute keys for encryption to the legitimate users. PKC is widely used in the Internet but may be challenged in the IoT context. Even though there exist lightweight implementations of PKC [40], e.g., TinyECC [41], NanoECC [42], some IoT devices still cannot afford the complexity. Many IoT devices have very limited computational resources and are powered by battery. Additionally, PKI may not be readily available in device-to-device IoT communications. Finally, the security of cryptographic schemes, both symmetric encryption and PKC, are threatened by the emerging quantum computers [43]. Symmetric encryption can be enhanced by increasing the length of the keys; but PKC relies on complex mathematical algorithms that are not scalable, which will be broken by quantum computers [43].

Although neither secure nor efficient, pre-shared key is actually quite a common method for deploying keys for the IoT devices, as exemplified by programming keys into a device from a PC through a USB cable. However, it is challenging to update the keys for IoT devices once they are configured and deployed, given their huge population and typical locations.

Having a weak key/password will expose the entire network to risk and has indeed already resulted in serious cyberattacks [44]. As shown in Fig. 4, the Kaspersky Lab reported that there were more than 120,000 malware modifications during the first half of 2018, which is more than triple the amount in 2017 [45]. It is further revealed in the report that 93% of the attacks were caused by weak passwords. For example, “admin” is often used as the default password for many devices.

Many IoT devices are eventually connected to the Internet, but they have become the “Achilles’ Heel” of the broad Internet network. The Dyn cyberattack is such a sad example, which occurred in the USA in October 2016 and affected millions of Internet users [44]. As illustrated in Fig. 5, the malware simply scanned all the connected IoT devices,
including web cameras, building gateways, baby monitors, and tried a password for access. Massive number of devices were unfortunately configured with the default password and hacked. The Mirai malware then initialized a series of severe distributed denial of service (DDoS) attacks, which broke down the Dyn, the domain name system (DNS) provider in the USA. Internet services were thus disrupted and millions of Internet users were affected. Similarly, another DDoS attack was applied to the Philips smart lamps, which use ZigBee as the communication protocol [46]. The authors developed a novel side channel attack to deduce the global key that was used for each device type. Thus, the worm can be spread easily from an infected node to a bulb of the same type, because the same key was used.

In summary, the IoT and also the associated Internet are significantly threatened by the weak passwords of connected devices. An efficient and lightweight key distribution scheme is urgently required for the low-cost IoT devices.

C. Key Generation

Apart from the above-mentioned security solutions, there is another popular technique of agreeing on a key extracted from wireless channels, which is termed as secret key agreement. Together with keyless security aided transmission, secret key agreement also falls under the umbrella of physical layer security, which achieves information-theoretical security by exploiting the unpredictable features of the random channel fading. Depending on the specific realization, secret key agreement has two models, namely the channel model and source model (Chapter 4 of [47]). The channel model-based key agreement operates in a similar manner to the wiretap channel model, which intends to securely transmit keys from Alice to Bob, and agree on the same key via a two-way public channel [48]–[50]. However, it also faces the same challenges as keyless security in terms of its practical implementation.

The source model of secret key agreement works in a different manner, namely by generating the keys from the wireless channel between Alice and Bob, rather than transmitting the keys, which is termed as key generation from wireless channels. The timeline is given in Fig. 6. The key generation philosophy dates back to 1993, when Ahlswede et al. [51] and Maurer [52] laid down its information-theoretical foundations. Since then, the past three decades have witnessed the ever more sophisticated exploration of this promising technique. A practical key generation protocol was proposed in 1995 [53] and in 1996 [54]. There have been extensive interests on theoretical exploration [55], [56], modelling [57]–[61] and protocol design [62]–[65]. Thanks to the rapid development of the semiconductor industry and wireless technologies, wirelessless attacks have become pervasive and lead to fruitful key generation prototyping and ultimately to its practical exploration. Key generation has then been applied to numerous wireless techniques, including IEEE 802.15.4/ZigBee (since 2005) [66]–[68], IEEE 802.11/WiFi (since 2008) [69]–[75], Bluetooth (since 2014) [76], LoRa/LoRaWAN (since 2018) [77]–[79].

The generated key can be used for one-time pad to achieve perfect secrecy, as explored in [80]. However, the key generation rate is not sufficiently high to support data communications. Hence, a more common application is constructing a hybrid cryptosystem using key generation and symmetric encryption, as shown in Fig. 7. Alice and Bob can generate the keys directly from their common wireless channel, without assistance from a third party, such as a PKI. Additionally, key generation is information-theoretically secure, hence it is not threatened by the emerging quantum computers. Finally, this technique is of lightweight nature, therefore it is eminently suitable for low-cost IoT devices. Therefore, key generation is an ideal alternative to PKC for the establishment of secure keys for the IoT.

This paper provides a comprehensive survey of random key generation from wireless channels. We introduce the key generation fundamentals, including the system modelling techniques and evaluation metrics. A full key generation protocol is proposed to exploit the common randomness of wireless channels between a pair of legitimate users. We then carefully review the associated design considerations of pairwise key generation by examining the channel parameters, signal domains, duplex modes and implementation aspects. We further extend the discussions from pairwise key generation to multiple players, which covers the multi-user and cooperative key generation scenarios, as well as the associated security analysis. Finally, we review the scientific debate on this technique and identify a number of promising research directions. In a nutshell, we survey the entire suite of practical protocol designs and applications suitable for different wireless techniques and scenarios.

There have been a number of key generation surveys and tutorials published in [81]–[87]. A summary and comparison is given in Table II. The most similar survey is the one that the authors published in 2016 [84]. This article significantly extends previous work by reviewing the exciting advances in the area since then.

D. Organization

The rest of the article is organized as follows. Section II introduces the wireless IoT techniques. Key generation fundamentals, including random sources, principles, information-theoretic models and metrics, are covered in Section III. Section IV, Section V and Section VI review the family of key generation protocols, design considerations as well
Fig. 6. Timeline of key generation from wireless channels

Fig. 7. A key generation-based hybrid cryptosystem. Key generation establishes the same session key for Alice and Bob. They then use the key for symmetric encryption.

as their implementation and applications, respectively. Key generation designed for multiple parties/nodes are discussed in Section VII. Section VIII briefly introduces device authentica-

tion, conceived for ascertaining the identity of key generation parties. Section IX suggests future research while Section X concludes the article. The paper’s structure is given in Fig. 8 for the convenience of readers. The abbreviations used in this article are listed in Table III.

II. WIRELESS IOT TECHNIQUES

Wireless connectivity has been widely used in the IoT, as a benefit of its convenient installation and flexible deployment [2]. Wireless networks can be divided into wireless local area networks (WLANs), wireless personal area networks (WPANs) and low power wide area networks (LPWANs). The same taxonomy is used in this paper as well. Naturally, the different wireless techniques have different communication ranges, data rates and energy consumption, since they are designed and optimized for particular applications. For example, WiFi has a high data rate in the order of 100 Mbps, but it
TABLE II
COMPARISON WITH AVAILABLE SURVEYS AND TUTORIALS

| Paper                     | Year | Type   | Principle | Protocol | Channel Parameter | Signal Domain | Duplex Mode | Applications | Attack | Relay | Group/Multi-User Key Generation | 5G related | Device Authentication |
|---------------------------|------|--------|-----------|----------|-------------------|---------------|-------------|--------------|--------|-------|---------------------------------|-------------|------------------------|
| Ren et al. [81]           | 2011 | Tutorial | ✓         | ✓        | ✓                 |               |             |              |        |       |                                 |             |                        |
| Zeng et al. [82]          | 2015 | Tutorial | ✓         | ✓        | ✓                 |               |             |              |        |       |                                 |             |                        |
| Wang et al. [83]          | 2015 | Survey  | ✓         | ✓        | ✓                 | ✓             | ✓           |              |        | ✓     |                                 |             |                        |
| Zhang et al. [84]         | 2016 | Survey  | ✓         | ✓        | ✓                 | ✓             | ✓           |              |        | ✓     |                                 |             |                        |
| Zhang et al. [85]         | 2017 | Tutorial | ✓         | ✓        | ✓                 | ✓             | ✓           |              |        | ✓     |                                 |             |                        |
| Li et al. [86]            | 2019 | Survey  | ✓         | ✓        | ✓                 | ✓             | ✓           |              |        | ✓     |                                 |             |                        |
| Zhang et al. [87]         | 2019 | Tutorial | ✓         | ✓        | ✓                 | ✓             | ✓           |              |        | ✓     |                                 |             |                        |
| This work                 |      | Survey  | ✓         | ✓        | ✓                 | ✓             | ✓           |              |        | ✓     |                                 |             | ✓                      |

is energy hungry, which is widely used in smartphones and laptops. On the other hand, LoRa can only achieve a rate up to 50 kbps, but can operate for years using a battery, which is suitable for sensor nodes. A summary and comparison of several popular wireless techniques is provided in Table IV and Fig. 9.

A. Wireless Local Area Networks

IEEE 802.11 is the most popular and successful WLAN technique, which is used in laptops, smartphones, tablets, etc [34]. WiFi is the industrial alliance that adopts and promotes IEEE 802.11 technology. In this paper we use IEEE 802.11 and WiFi interchangeably.

Since its conception in 1997, the WiFi family has evolved quickly with the advances of wireless and semiconductor technologies. It has also had a number of successful amendments, including a/b/g/n/ac/ah/ax, as summarized in Table V. Orthogonal frequency-division multiplexing (OFDM) is the main physical layer modulation scheme of WiFi, which was first used in IEEE 802.11a in 1999 and later adopted by IEEE 802.11g/n/ac/ah/ax. OFDM exploits the available spectrum efficiently by transmitting on orthogonal subcarriers/frequencies and improves the communication rate. Following the introduction of IEEE 802.11n (2009), WiFi has been further enhanced by multi-antenna techniques for exploiting spatial diversity. Finally, because of the increased user density, IEEE 802.11ax employs multi-user access techniques for enabling simultaneous transmission between an access point (AP) and multiple stations.

IEEE 802.11 can be used in the smart home and diverse indoor applications, where large amounts of data transfer is required, as in residential camera-based monitoring. However, the communication range remains limited within 100 meters, but the IEEE 802.11ah amendment, also known as WiFi Halow, supports a longer range with a coverage of one kilometer radius.

B. Wireless Personal Area Networks

IEEE 802.15.4 defines the physical layer and medium access control (MAC) layer protocols [35] and serves as the basis for ZigBee, 6LoWPAN, WirelessHART, etc. It uses direct-sequence spread spectrum-based transmission of signals. It is particularly suitable for low-power, low-rate (up to 250 kbps) and short-distance (up to 100 meters) communications. It is the main technique used for WPANs and it has been widely used in wireless sensor networks (WSNs) (e.g., for environment monitoring), smart home, industrial automation, etc.

Bluetooth is a low-energy wireless technique for short range communications (50 to 100 m), with a data-rate of up to 1 Mbps. Bluetooth was conceived in 1989 and the latest version is v5.1 (January 2019) [36], which has been widely used in smartphones, laptops and Fitbits. Bluetooth also operates at 2.4 GHz but employs adaptive frequency hopping to avoid channel collision. In contrast to IEEE 802.15.4, Bluetooth uses a single-hop solution, which is suitable for healthcare devices and consumer electronics [88].

Ultra-wideband (UWB) is a low-energy technique conceived for short-range, high-bandwidth (> 500 MHz) communications [89]. It has been included in the IEEE 802.15.4-2015 standard for WPANs and IEEE 802.15.6-2012 standard for wireless body area networks (WBAN). Numerous solutions have been proposed for UWB systems, relying on impulse radio [89], OFDM [90] and multi-stage frequency hopping [91], just to name a few.

C. Low Power Wide Area Networks

Many IoT applications rely on distributed devices in a wide area, thus will have to use long range communications, for example, for environmental monitoring and smart cities. These IoT devices should also be low power to support long operation. The wireless connection techniques are thus termed as low power wide area networks (LPWAN) [92]. LPWAN techniques have become prevalent, including LoRa/LoRaWAN,
Sigfox, and NB-IoT. Both LoRa/LoRaWAN and Sigfox operate in the unlicensed ISM bands, while NB-IoT is a cellular technique [38].

LoRa is a physical layer modulation technique patented by
TABLE IV
WIRELESS TECHNIQUES FOR THE IOT

| Wireless Technique       | Frequency       | Range              | Data Rate    | Security Mechanism                                      |
|--------------------------|-----------------|--------------------|--------------|---------------------------------------------------------|
| IEEE 802.15.4            | 2.4 GHz         | 10 to 100 m        | 250 kbps     | AES in MAC layer                                        |
| Bluetooth Low Power      | 2.4 GHz         | 50 to 150 m        | 1 Mbps       | AES in link layer                                       |
| IEEE 802.11 a/b/g/n/ac/ax| 2.4 or 5 GHz    | 50 m               | > 100 Mbps   | WPA in MAC layer (with AES implemented)                 |
| IEEE 802.11 ah           | sub GHz         | 1 km               | 150 kbps     | WPA in MAC layer (with AES implemented)                 |
| LoRa/LoRaWAN             | sub GHz         | 15 km              | 0.3 kbps to 50 kbps | AES 128 Encryption at network and application layer |
| SigFox                   | sub GHz         | 30-50 km in rural areas or 3-10 km in urban area | 200 bps | AES-based cryptography                                 |
| NB-IoT                   | sub GHz, licensed band | 18 - 21 km | 200 kbps | Symmetric key cryptography, e.g., UEA2 and UIA2, AKA, etc. |

Fig. 9. Wireless techniques for the IoT.

TABLE V
IEEE 802.11 PHYSICAL LAYER EVOLVEMENT

| Amendment     | Release Year | Frequency (GHz) | Modulation |
|---------------|--------------|-----------------|------------|
| IEEE 802.11   | 1997         | 2.4             | DSS, FHSS  |
| IEEE 802.11b  | 1999         | 2.4             | DSS        |
| IEEE 802.11a  | 1999         | 5               | OFDM       |
| IEEE 802.11g  | 2003         | 2.4             | OFDM       |
| IEEE 802.11n  | 2009         | 2.4/5           | MIMO OFDM  |
| IEEE 802.11ac | 2013         | 5               | MIMO OFDM  |
| IEEE 802.11ah | Est. 2020    | sub GHz         | MIMO OFDM  |
| IEEE 802.11ax |             | 2.4/5           | MIMO OFDM, multi user |

Semtech, which employs chirp spread spectrum transmission for distances as high as 15 km. LoRaWAN is proposed by the LoRa Alliance [37], which defines the MAC layer protocol and network architecture. The LoRaWAN scheme operates at sub-GHz carriers but the specific frequency plans of different countries vary [93]. SigFox uses an ultra-narrow band technique for supporting extremely long-range transmissions, namely up to 30 to 50 km in rural areas or 3 to 10 km in urban environments. Again, NB-IoT is a cellular technique operating in a licensed band. It works in the classic frequency division duplex (FDD) mode, which poses challenges for the key generation process, because the uplink and downlink channels are not necessarily similar at different frequencies.

D. Security Mechanism

IoT security has attracted extensive research interests in diverse fields, such as Internet of Vehicles [94], [95], smart homes [96], healthcare [97]–[99], etc. As summarized in Table IV, AES-based encryption is widely used for achieving data confidentiality and integrity in the IoT. AES can be implemented in a hardware-friendly manner, which is very suitable for low-cost IoT devices. For example, AES has been integrated in the popular Texas Instruments (TI) ZigBee chipset, cc253x [100].

While the IoT standards have defined the encryption mechanisms, a secure and efficient key distribution scheme is still currently missing. Key generation is an ideal candidate technique for establishing cryptographic keys for legitimate users in a lightweight and secure manner.

III. FUNDAMENTALS

This section will cover the randomness source, key generation principles, information-theoretical fundamentals and metrics. These aspects will be linked to each other.

A. Randomness Source

Wireless communications undergo large-scale fading, including the path loss and shadow fading, as well as small scale multipath fading [101]. The path loss represents the power decay over the transmission path, which is a direct function of the transmission distance, whilst its steepness depends both on the carrier frequency and on the building patterns for example. However, the path loss is rather deterministic and thus is not secure for key generation [102]. On the other hand, shadow fading is a correlated random process caused by large obstacles in the environment, such as buildings, which has been used for key generation in [102]. However, shadow fading is changing relatively slowly, which limits the key generation performance. An experimental validation of key generation based on shadow fading is not available at the time of writing.
Hence, the majority of key generation contributions focus on the small-scale fading. As shown in Fig. 10, the electromagnetic wave undergoes reflections, refraction and scattering in the environment. These effects are unpredictable and can be used as the random source of key generation.

Channel modelling is essential for designing reliable and efficient key generation. A detailed channel model\footnote{From now on, channel model represents the modelling of wireless channel, but is not related to the channel model of secret key agreement.} for both narrowband and wideband channels can be found in Chapter 3 of [101]. This section will provide a brief introduction to the relevant channel effects. The multipath channel can be modelled by several resolvable path components. The corresponding channel impulse response (CIR), $h_{uv}(\tau, t)$, between the transmitter $u$ and receiver $v$ can be mathematically expressed as

$$h_{uv}(\tau, t) = \sum_{l=0}^{L_{uv}(t)} \alpha_{l}^{uv}(t)e^{-j\phi_{l}^{uv}(t)}\delta(\tau - \tau_{l}^{uv}(t)), \quad (2)$$

where $\alpha_{l}^{uv}(t)$, $\phi_{l}^{uv}(t)$ and $\tau_{l}^{uv}(t)$ are the amplitude attenuation, phase shift and time delay of the $l^{th}$ tap, respectively, $L_{uv}(t)$ is the total number of paths and $\delta(\cdot)$ is the Dirac function.

When a signal $s(t)$ is transmitted via a multipath channel, the received signal is given by the convolution of

$$y(t) = \int_{0}^{\tau_{\text{max}}} h_{uv}(\tau, t)s(t - \tau)d\tau + n^{v}(t), \quad (3)$$

where $n^{v}(t)$ is the noise at receiver $v$ and $\tau_{\text{max}}$ is the maximum channel delay. The received power of a packet having a duration of $T_{\text{pkt}}$ is given as

$$P(t) = \frac{1}{T_{\text{pkt}}} \int_{t}^{t+T_{\text{pkt}}} |y(t')|^{2}dt'. \quad (4)$$

The received signal can be converted to the frequency domain, which is given by

$$Y(f, t) = H_{uv}(f, t)S(f, t) + w^{v}(f, t), \quad (5)$$

where $H_{uv}(f, t)$ is the corresponding channel frequency response (CFR) given by

$$H_{uv}(f, t) = \int_{0}^{\tau_{\text{max}}} h_{uv}(\tau, t)e^{-j2\pi f \tau}d\tau. \quad (6)$$

The CIR $h_{uv}(\tau, t)$ includes the intrinsic randomness source, which can be represented by the CFR $H_{uv}(f, t)$ and received power $P(t)$ as well. A detailed introduction to these parameters will be given in Section V-A. We use $X^{u}$ to denote the channel observation of user $u$, which can be one of the above parameters. The channel effects are determined by the specific environment (indoor or outdoor), the reflector and scatterer material and distribution, which leads to unpredictable fading of the wireless channels. Depending on the wireless technique adopted, key generation exploits these features by measuring $X^{u}$ and extracts the common randomness as the key.

### B. Information-Theoretic Fundamental

The source model of key generation is given in Fig. 11, which involves two legitimate users, Alice and Bob, and a passive eavesdropper, Eve [47]. Alice, Bob and Eve acquire the channel observations $X^{A} = [x^{A}(1), x^{A}(2), \ldots, x^{A}(n)]$, $X^{B} = [x^{B}(1), x^{B}(2), \ldots, x^{B}(n)]$, and $X^{E} = [x^{E}(1), x^{E}(2), \ldots, x^{E}(n)]$, respectively.

Key generation is information-theoretically secure, which has been shown in the pair of seminal papers [51], [52]. In order to agree on using the same key, Alice and Bob will have to exchange some information $s$ over the public channel, which can be overhead by Eve as well. For any $\epsilon$ and sufficiently large $n$, there exists a key generation protocol, $K_{ir}^{A} = g_{A}(X^{A})$ and $K_{ir}^{B} = g_{B}(X^{B}, s)$, which satisfies

$$\Pr(K_{ir}^{A} \neq K_{ir}^{B}) < \epsilon, \quad (7)$$

$$\frac{1}{n}I(K_{ir}^{A}, s; X^{E}) < \epsilon, \quad (8)$$

$$\frac{1}{n}H(K_{ir}^{A}) > R - \epsilon, \quad (9)$$

$$\frac{1}{n}\log |\mathcal{K}| < \frac{1}{n}H(K_{ir}^{A}) + \epsilon, \quad (10)$$

where $I(\cdot, \cdot)$ denotes mutual information and $\mathcal{K}$ represents the key’s alphabet. (7) is about the channel reciprocity, which indicates that Alice and Bob can get the same keys with a high probability. Furthermore, (8) is based on the spatial decorrelation, which means that Eve cannot infer the keys based on her own observation and the public discussion $s$. Finally, (10) describes the temporal variation, which ensures having a uniformly distributed key.

A detailed introduction of the information-theoretical model of key generation can be found in Chapter 4 of [47].
C. Principle

The above information-theoretical modelling can be described by three key generation principles.

1) Channel Reciprocity: Channel reciprocity indicates that the channel gains and phases are the same at both ends of the link. As seen in (7), Alice and Bob can then generate the same keys, \( K^u_A \) and \( K^v_B \), from their channel observations, namely \( X^A \) and \( X^B \), respectively. However, the channel reciprocity is impacted in practice by the specific duplex mode used, the hardware imbalance, interference and noise, which will be further discussed in Section IV-A1 and Section V-C.

2) Spatial Decorrelation: According to Jakes’ Doppler model [103], the correlation function is represented by the Bessel function of zeroth order and first kind in a rich multipath environment with infinite and uniformly distributed scatterers. Thus the eavesdroppers will experience uncorrelated fading, when they are located at least 0.4\( \lambda \) (approximately half wavelength) away from the legitimate users [101]. This feature is termed as spatial decorrelation, which is essential to the security of key generation. As seen in (8), based on the uncorrelated channel observation and the public messages received, Eve is unable to extract the key. However, the condition is quite rigid, which may not hold in a real environment. More detailed discussions will be presented in Section VII-C1.

3) Temporal Variation: Temporal variation describes the channel variation over the time, which can be caused by the movement of the transmitter, receiver and any objects within the environment. Having temporal variations is essential for generating random uniformly distributed keys, as seen in (10), which are desired by cryptographic applications. A detailed study will be given in Section V-B1.

D. Evaluation Metric

There are a number of metrics in the key generation area for evaluating the quality of the keys generated.

1) Cross-Correlation: The signal similarity can be quantified by the cross-correlation coefficient between the measurements of user \( u \) and user \( v \), i.e. \( X^u \) and \( X^v \), which is formulated as

\[
\rho^{uv} = \frac{E\{X^u X^v\} - E\{X^u\}E\{X^v\}}{\sigma_u \sigma_v},
\]

where \( E\{\cdot\} \) denotes the expectation operation and \( \sigma_u \) is the standard deviation of \( X^u \).

2) Key Disagreement Rate (KDR): The KDR quantifies the difference between the raw keys generated at user \( u \) and user \( v \) after the quantization, i.e., \( K^u_q \) and \( K^v_q \), which is mathematically expressed as

\[
\text{KDR}^{uv} = \frac{\sum_{i=1}^{n_k} |K^u_q(i) - K^v_q(i)|}{n_k},
\]

where \( n_k \) is the length of keys. A KDR modelling technique was formulated for OFDM systems and was also validated by measurements in [104].

3) Secret Key Rate (SKR): SKR is the upper bound on the number of bits per channel observation that Alice and Bob can generate, about which Eve cannot obtain any useful information based on her own observation. Maurer proved the following lower bound and upper bound on the key rate in [52], which are given as

\[
R(X^A, X^B||X^E) \geq \max\{ I(X^A; X^B) - I(X^A; X^E), \\
I(X^A; X^B) - I(X^B; X^E) \},
\]

and

\[
R(X^A, X^B||X^E) \leq \min\{ I(X^A; X^B), I(X^A; X^B||X^E) \},
\]

respectively. The maximum attainable SKR of a Nakagami \( m \) fading channel was quantified in [105].

4) Key Generation Rate (KGR): KGR describes the number of key bits generated in each unit time interval, e.g., bit per second or bit per measurement. Note that KGR represents the actual rate of the key produced by a key generation system, while the SKR indicates the theoretical maximum rate that the system can achieve. Alice and Bob can get a KGR approaching the SKR with the aid of well-designed protocols.

5) Autocorrelation Function: The signal variation can be quantified by the autocorrelation function (ACF) of the signal, which is given by

\[
r^u(t, \Delta t) = \frac{E\{(X^u(t) - \mu^u)(X^u(t + \Delta t) - \mu^u)\}}{\sigma^u_u},
\]

where \( \mu^u \) represents the mean value of the random variable \( X^u \). The ACF of the channel responses is theoretically modelled in [60], for a wide sense stationary uncorrelated scattering (WSSUS) channel.

6) Randomness: Because the keys generated are used for cryptographic applications, they are exposed to the risk of brute-force attack, unless the key is truly random. The National Institute of Standards and Technology (NIST) random test suite is widely used to evaluate the randomness for random number generators (RNG) and pseudo random number generators (PRNG) [106]. Key generation is a RNG therefore this test suite can also be used for this purpose.

The suite includes a total of 15 tests, each evaluating a specific feature, as shown in Table VI. Each test returns a \( P \)-value, which is compared to a statistical significance level \( \alpha \), typically in the range of \([0.001 \ 0.01]\). When the \( P \)-value > \( \alpha \), the sequence passes the test. Some tests require a long sequence with e.g., \( 10^6 \) bits, which cannot be readily gleaned from key generation simulations and experiments. Therefore, only a subset of tests are used for evaluating whether the keys generated possess these features.

An official C implementation is provided for download at [107] and a Python implementation is also available at github [108].

E. Summary

The key generation principles, information-theoretic fundamentals and the evaluation metrics are intrinsically linked to each other, which is summarized in Table VII. The metrics
TABLE VI
NIST Random Test Suite [106]

| Test                                      | Purpose                                                                 | Recommended key size $n_k$ |
|-------------------------------------------|------------------------------------------------------------------------|----------------------------|
| Frequency (monobit) test                  | Proportion of zeros and ones for the entire sequence                   | 100                        |
| Frequency test within a block             | To determine whether the frequency of ones in an $M$-bit block is approximately $M/2$ | 100                        |
| Runs test                                 | Total number of runs in the sequence, where a run is an uninterrupted sequence of identical bits | 100                        |
| Longest run of ones in a block test       | Longest run of ones within $M$-bit blocks                              | $n_k=128$, $M=8$            |
| Binary matrix rank test                   | Check linear dependence among fixed length substrings of the original sequence | 38,912                     |
| Discrete fourier transform (Spectral) test| To detect periodic features (i.e., repetitive patterns that are near each other) | 1000                       |
| Non-overlapping template matching test    | To detect generators that produce too many occurrences of a given non-periodic (aperiodic) pattern | Not specified               |
| Overlapping template matching test        | To detect generators that produce too many occurrences of a given non-periodic (aperiodic) pattern | $10^6$                     |
| Maurer’s universal statistical test       | To determine whether or not the sequence can be significantly compressed without loss of information | 387,840                    |
| Linear complexity test                    | To determine whether or not the sequence is complex enough to be considered random | $10^6$                     |
| Serial test                               | The frequency of all possible overlapping $m$-bit patterns across the entire sequence | Choose $m$ and $n$ such that $m < \lfloor (\log_2 n_k - 2) \rfloor$ |
| Approximate entropy test                  | To compare the frequency of overlapping blocks of two consecutive/adjacent lengths $(m$ and $m+1$) against the expected result for a random sequence | Choose $m$ and $n$ such that $m < \lfloor (\log_2 n_k - 5) \rfloor$ |
| Cumulative Sums test                      | The maximal excursion (from zero) of the random walk defined by the cumulative sum of adjusted (-1, +1) digits in the sequence | 100                        |
| Random excursions test                    | The number of cycles having exactly $K$ visits in a cumulative sum random walk | $10^6$                     |
| Random excursions variant test            | To detect deviations from the expected number of visits to various states in the random walk | $10^6$                     |

evaluate both the quality of analog measurements (cross-correlation and ACF, SKR) and the performance metrics such as the KDR, KGR and randomness.

IV. KEY GENERATION PROTOCOL

A key generation protocol typically relies on four stages, including channel probing, quantization, information reconciliation and privacy amplification, which are portrayed in Fig. 12 and will be detailed later in this Section. Alice and Bob first carry out channel probing, which involves bidirectional measurements, and will obtain the measurements $X^A$ and $X^B$, respectively. They then convert the analog measurements into digital binaries, namely $K^A$ and $K^B$. There will probably be a mismatch between $K^A$ and $K^B$, hence information reconciliation has to be adopted to correct the mismatch; Alice and Bob will then obtain $K^{A}_{\text{IR}}$ and $K^{B}_{\text{IR}}$, respectively. Finally, privacy amplification is employed and the legitimate users acquire $K^A$ and $K^B$. Again, this section will introduce each of these stages in detail.

A. Channel Probing

Channel probing is the most essential step of key generation from wireless channels. The users will sample the channel via packet transmissions, which may be subject to all typical channel effects, such as sampling delay, interference and noise. Signal preprocessing can thus be adopted for improving the measurement quality.

1) Channel Sampling: Key generation requires bidirectional measurements, so that both users can glean the reciprocal channel information. Here, we describe the channel sampling process of time division duplex (TDD) systems as an example, while channel sampling associated with other duplex modes will be discussed in Section V-C.

The timing of the TDD-based channel sampling is illustrated in Fig. 13. At the $i^{th}$ sampling instant $t_q(i)$, Alice sends a request packet to Bob, who will obtain the measurement $X^B(i)$. After a time delay $\Delta t$, Bob replies with a packet to Alice, who will also measure the same parameter and get $X^A(i)$. Because in TDD schemes both directions use the same carrier frequency, unless strong frequency-selective
fading and different co-channel interference is encountered, the complex-valued channel envelope remains near-constant during the coherence time $T_c$ (defined in (23)). Hence, Alice and Bob can get highly correlated measurements. Alice and Bob will repeat the above sampling every $T_s$ time interval, where $T_s > T_c$, in order to avoid having correlated samples. Figs. 14(a) and 14(b) show the received power sampled by using WiFi in an indoor environment and using LoRa in an urban environment, respectively.

It is worth noting that at this early stage of their communications, Alice and Bob do not intend to decrypt the received messages, they simply aim to measure the channel using these sampling pilots and public links. Additionally, the legitimate users may also rely on payload data packets to sample the channel, if extra packet transmissions have to be avoided [68]. For example, each DATA packet will be confirmed by an Acknowledgement packet in classic WiFi transmissions, which jointly constitute a perfect pair for key generation. Therefore key generation does not impose additional energy consumption, which is beneficial for IoT devices.

2) Signal Preprocessing: Signal preprocessing mainly deals with two problems of the raw channel measurements, i.e., the channel reciprocity impairment and sample autocorrelation.

Channel reciprocity impairments are caused by hardware imbalance, fading, interference imbalance and noise.

- Different duplex modes have different impact, e.g., sampling delay in TDD systems, independent fading caused by frequency separation of the uplink and downlink carriers in FDD systems and self interference in in-band full-duplex (IBFD) systems.
- Hardware imbalance implies that the transmit and receive radio frequency chains in transceivers are not identical.
- Both inter-symbol and multi-user interference may be inflicted by the network. Fading is caused by mobility, while thermal noise is owing to the Brownian motion of electrons in the receiver.

As observed from the results in Figs. 14(a) and 14(b), the received powers of Alice and Bob are highly correlated, but not exactly the same in both scenarios. The received power variation is as high as 70 dBm in the LoRa-based large scale experiments, compared to the more moderate 25 dBm variation in the WiFi-based indoor environment.

The undesired autocorrelation manifests itself between the adjacent measurements, when the two probes are within the same coherence time and/or coherence bandwidth, which will introduce redundancy. This correlation may be introduced in the temporal, frequency and spatial domains, when employing for example OFDM techniques [60], [73], [109] or multiple antennas [58], [72], [110].

Various signal preprocessing algorithms have been proposed to address the above issues, which are summarized as follows.

- The countermeasures of mitigating the correlation of duplex modes will be given in Section V-C.
- Hardware asymmetry can be mitigated by calibration in advance [111]. As another innovative technique, a real-time transform based on the time-invariant nature of hardware imbalance was proposed for time-varying TDD channels without involving any calibration [112].

As observed from the results in Figs. 14(a) and 14(b), the received powers of Alice and Bob are highly correlated, but not exactly the same in both scenarios. The received power variation is as high as 70 dBm in the LoRa-based large scale experiments, compared to the more moderate 25 dBm variation in the WiFi-based indoor environment.

The undesired autocorrelation manifests itself between the adjacent measurements, when the two probes are within the same coherence time and/or coherence bandwidth, which will introduce redundancy. This correlation may be introduced in the temporal, frequency and spatial domains, when employing for example OFDM techniques [60], [73], [109] or multiple antennas [58], [72], [110].

Various signal preprocessing algorithms have been proposed to address the above issues, which are summarized as follows.

- The countermeasures of mitigating the correlation of duplex modes will be given in Section V-C.
- Hardware asymmetry can be mitigated by calibration in advance [111]. As another innovative technique, a real-time transform based on the time-invariant nature of hardware imbalance was proposed for time-varying TDD channels without involving any calibration [112].

As observed from the results in Figs. 14(a) and 14(b), the received powers of Alice and Bob are highly correlated, but not exactly the same in both scenarios. The received power variation is as high as 70 dBm in the LoRa-based large scale experiments, compared to the more moderate 25 dBm variation in the WiFi-based indoor environment.

The undesired autocorrelation manifests itself between the adjacent measurements, when the two probes are within the same coherence time and/or coherence bandwidth, which will introduce redundancy. This correlation may be introduced in the temporal, frequency and spatial domains, when employing for example OFDM techniques [60], [73], [109] or multiple antennas [58], [72], [110].

Various signal preprocessing algorithms have been proposed to address the above issues, which are summarized as follows.

- The countermeasures of mitigating the correlation of duplex modes will be given in Section V-C.
- Hardware asymmetry can be mitigated by calibration in advance [111]. As another innovative technique, a real-time transform based on the time-invariant nature of hardware imbalance was proposed for time-varying TDD channels without involving any calibration [112].

B. Quantization

Following the above channel probing process, Alice and Bob obtain a series of analog channel measurements, $X^A$
output: preliminary key material. Quantization can be categorized into
with $\alpha$ key design parameters include the threshold value selection
sentations by comparing the measurements to thresholds. The
based quantizer converts the analog values into binary repre-
and $X$ and $Y$ respectively, but binary keys are required for crypto-
graphic schemes. The quantization stage converts the analog channel measurements into digital binary sequences, $K_q^A$ and $K_q^B$. We refer to the quantized binary sequence as the preliminary key material. Quantization can be categorized into absolute value-based and difference value-based quantization, which will be introduced.

1) **Absolute Value-based Quantization**: An absolute value-based quantizer converts the analog values into binary representations by comparing the measurements to thresholds. The key design parameters include the threshold value selection and the number of quantization levels.

Mean and standard deviation-based quantization is the most popular one, which is summarized by the pseudo code given in Algorithm 1. An example is shown in Fig. 15, in conjunction with $\alpha = 0$, which corresponds to mean value-based quantization. The quantizer is simple to implement, since it only requires the mean and variance of the samples for calculating the threshold. However, it is not robust to burst errors, which are quite common in wireless communications. Explicitly, the burst errors may significantly affect the threshold and result in unbalance between the proportions of 1s and 0s.

Cumulative distribution function (CDF)-based quantization operates differently from the above quantizer in terms of its threshold selection procedure [62], as detailed in Algorithm 2. The threshold is calculated based on the distribution of measurements, and as a benefit an even proportion between 1s and 0s can be ensured. It can also be designed for multi-bit quantization by assigning more quantization levels and thresholds [62], [73]. Usually a Gray code is adopted for ensuring that similar samples result in similar binary strings having only a single different bit position, hence yielding a Hamming distance of one. However, CDF-based quantizers are more complex, requiring more resources.

2) **Differential Value-based Quantization**: In contrast to absolute value-based quantizers, a differential value-based quantizer generates keys by comparing a pair of the adjacent measurements [118], as seen in Algorithm 3. The difference threshold of $\epsilon$ is introduced to ensure that minor fluctuations caused by hardware noise are ignored. An example is given in Fig. 16 in conjunction with $\epsilon = 0$.

This quantizer is eminently suitable for large-scale outdoor environments, where the channel variation is high but changing slowly. A case study can be found in [77], where LoRa-based key generation experiments were carried out in an urban environment. As shown in Fig. 16, the mean value-based quantizer may result in large chunks of 1s (or 0s), because the signal variation is not high enough compared to the global mean value. This can be improved by block-wise quantization, i.e. by partitioning the measurements into small blocks and quantizing individual blocks [70]. However, the block-based quantizer has to learn the environment in order to determine and adjust the length of the blocks.

3) **Summary**: Quantization determines the KGR, as it directly controls the number of key bits that can be generated per measurement. To this end, a number of quantizer variants of the above two main approaches have been designed and tested. A comparison among different quantizers can be found in [119], [120].

Different from the above quantizers, the work in [121] employed the machine learning clustering algorithms, namely the k-means, for quantization. The authors used the real and

| Algorithm 1: Mean and standard deviation-based quantization |
|-----------------------------------------------------------|
| **INPUT:** $X^u$ % Channel measurement                      |
| **INPUT:** $\alpha$ % Tuning parameter                     |
| **OUTPUT:** $K^u_q$ % Generated key sequence of user $u$   |
| 1: $\eta^u_+ = \mu_x + \alpha \times \sigma_x$ $\eta^u_+$ is the positive threshold. |
| 2: $\eta^u_- = \mu_x - \alpha \times \sigma_x$ $\eta^u_-$ is the negative threshold. |
| 3: for $i \leftarrow 1$ to $N_y$, do                       |
| 4: if $X^u(i) > \eta^u_+$ then                             |
| 5: $K_q^A(i) = 1$                                           |
| 6: else if $X^u(i) < \eta^u_-$ then                        |
| 7: $K_q^B(i) = 0$                                           |
| 8: else                                                    |
| 9: $X^u(i)$ dropped                                        |
| 10: end if                                                 |
| 11: end for                                                |

| Algorithm 2: CDF-based quantization                        |
|-----------------------------------------------------------|
| **INPUT:** $X^u$ % Channel measurement                     |
| **INPUT:** $QL$ % Quantization level                       |
| **OUTPUT:** $K^u_q$ % Generated key sequence of user $u$   |
| 1: $F(x) = Pr(X^u < x)$ % CDF calculation                 |
| 2: $\eta^u_0 = -\infty$ % Threshold                      |
| 3: for $j \leftarrow 1$ to $2^{QL} - 1$, do              |
| 4: $\eta^u_j = F^{-1}\left(\frac{j}{2^{QL}}\right)$ % Threshold |
| 5: end for                                                |
| 6: $\eta^u_{\text{int}} = \infty$                        |
| 7: Construct Gray code $b_j$ and assign them to different intervals $[\eta_{j-1}, \eta_j]$ |
| 8: for $i \leftarrow 1$ to $N_y$, do                      |
| 9: if $\eta_{j-1} < X^u(i) < \eta_j$ then                |
| 10: $K_q^A(i) = b_j$                                      |
| 11: end if                                                |
| 12: end for                                               |

![Fig. 15. Mean value-based quantization with received power sampled by using WiFi in an indoor office environment. The mean values are calculated based on all the received power in Fig. 14(a).](image)
Algorithm 3 Differential-based quantization

INPUT: $X^u$ % Channel measurement
INPUT: $\epsilon$ % Parameter resolution
OUTPUT: $K^u_q$ % Generated key sequence of user $u$
1: for $i \leftarrow 1$ to $N_p - 1$ do
2:  if $X^u(i + 1) > X^u(i) + \epsilon$ then
3:    $K^u_q(i) = 1$
4:  else if $X^u(i + 1) < X^u(i) - \epsilon$ then
5:    $K^u_q(i) = 0$
6:  else
7:    $X^u(i)$ dropped
8:  end if
9: end for

Fig. 16. Differential-based quantization with received power sampled by using LoRa in an urban environment. The mean value-based quantizer does not work in this case. The mean values are calculated based on all the received power in Fig. 14(b).

imaginary parts of the channel coefficients are the clustering features, calculated a number of cluster centers, and assigned gray codes to these centers.

C. Information Reconciliation

The objective of key generation is to generate a pair of identical symmetric keys at Alice and Bob for cryptographic applications. Even a single bit difference would result in decryption failure, due to the avalanche-like effects. As shown in Fig. 15 and Fig. 16, even when the absolute values of the received power of Alice and Bob are very close, Alice and Bob may still quantize them differently.

To address this issue, information reconciliation has to be used for detecting and correcting the errors in the preliminary key material between a pair of legitimate parties, i.e., $K^A_q$ and $K^B_q$. A survey of the information reconciliation techniques can be found in [122]. Information reconciliation tends to rely on a pair of approaches, i.e., error detection protocol based approaches (EDPA) and error correction code based approaches (ECCA).

It is worth mentioning that many of the information reconciliation and privacy amplification methods used in wireless key generation are borrowed from the field of quantum key distribution (QKD) [123].

1) EDPA: As described in Fig. 17, Alice first partitions the preliminary key material gleaned from the signals received from Bob into small blocks and sends parity information of each block to Bob. Similarly, Bob also partitions his key material in the same way, derives parity check bits and checks for mismatches between his own parity bits and those received from Alice. For each mismatch, Bob performs a binary search right across the block to find a correction vector, which may fix the errors. These steps may be repeated a number of times to eliminate mismatches and to obtain a high probability of success.

Specific examples of EDPA schemes include BBBSS [124], Cascade [125] and Winnow [126]. To elaborate a little further, Bennett et al. proposed the permute-and-bisect method for the first implementation of QKD in [124]. As a further advance conceived for reducing the information leakage, Brassard and Salvail proposed an improved scheme termed as Cascade in [125], which exploits the information gleaned from the preceding iterations for correcting errors during the current pass. A more efficient implementation of Cascade exploits some inherent information already available in the protocol, such as exactly known bits and/or already known parities [127]. In contrast to BBBSS and Cascade, Butler et al. [126] proposed to correct the errors in the block using syndrome-based error correction in the context of Hamming codes. The parity bits and syndromes can be calculated and exchanged in parallel. However, Winnow may introduce new errors if the error count per block is more than two. A modified one-way error reconciliation protocol using a Hamming code-based concatenated scheme was proposed to study the relationship between the error correction capability and the key generation efficiency in [128].

2) ECCA: Information reconciliation may also be viewed as a special case of channel coding and correction. Therefore, literally the entire family of forward error correction (FEC) codes can be adapted for reconciliation. Hence numerous error correction codes have been used for information reconciliation, including BCH codes [129]–[131], Reed-Solomon codes [132], Golay codes [133]–[135], turbo [136], polar [137] and low density parity check (LDPC) codes [16], [59], [71], [138].

The ECCA algorithm is described in Fig. 18. Again, Alice and Bob first partition the preliminary key material into blocks. Then, by relying on an error correction code, Alice encodes the key materials, $K^A_q$, calculates and sends the syndrome to Bob. Bob applies the corresponding decoder, whereby the required codeword is composed of Bob’s key, $K^B_q$ and the
received syndrome. When the number of bit disagreements is smaller than the code’s error correcting capability, having synchronized key material is guaranteed by this single-round interaction. Following the error correction procedure, the key agreement can be confirmed by employing CRC. If the check values of Alice and Bob match, i.e., \( p^A = p^B \), Alice and Bob generate the same keys and they will proceed to the privacy amplification stage. Otherwise, they will have to start over from the channel probing stage.

To elaborate a little further, secure sketch is a widely used ECCA information reconciliation protocol [129], which is described in Algorithm 4 and illustrated in Fig. 19. We use BCH coding as an example. A BCH \((n, k, t)\) code has an \(n\)-bit codeword and \(k\)-bit message; it can correct up to \(t\)-bit errors. As shown in Fig. 19, Alice first randomly selects a codeword \(c\) from the BCH code set \(C\). Alice then calculates the syndrome based on the exclusive-OR operation, given as \(s = \text{XOR}(K_q^A, c)\). It should be noted that the syndrome calculation here is different from that of classic FEC. After that, she transmits the syndrome \(s\) to Bob. Assuming Bob receives the syndrome correctly, he calculates a codeword as \(c^B = \text{XOR}(K_q^B, s)\). When the errors are correctable, Bob can get \(c'\) by decoding \(c^B\), and arrives at \(c' = c\). Finally, he will get a new key by exclusive-OR operation, namely \(K_{ir} = \text{XOR}(c', s)\). Fig. 19(b) exemplifies the error correction process by using the BCH (7,4,1) code as an example, which has a codeword length of \(n = 7\) and can correct \(t = 1\) bit error. Let us consider \(K_q^A = [1010011]\) and \(K_q^B = [1000001]\) as an example, where there is a single bit difference between them. This will result in one bit difference between \(c^B\) and \(c\), which is within the code’s correction capacity.

There are also other FEC-based information reconciliation techniques. Treeviriyanupab et al. used the syndromes of a BCH code for error correction and a one-bit feedback to report successful decoding [130]. An information reconciliation protocol based on a rate compatible LDPC code construction was proposed in [139].

3) Summary: Reconciliation efficiency is one of the most commonly used metrics, which is inversely proportional to the bit leakage rate. However, there is a paucity of literature on the interaction delay and computational complexity, which should be considered, in particular in case of IoT devices having limited resources.

Li et al. proposed a new hybrid information reconciliation protocol integrating the BBBSS protocol and BCH codes [140]. Their objective was that of maximizing the proportion of corrected bits per unit time, whilst making a trade-off amongst the conflicting performance indicators of information leakage, interaction delay and computational complexity. Future work should take into consideration these metrics and improve the reconciliation performance.

On the other hand, the work in [141], [142] designed key
generation protocols without using information reconciliation. Alice encrypted the information bits with the unreconciled keys using XOR operation. The encrypted bits received by Bob may be affected by the transmissions errors and channel coding and decoding are usually used to achieve successful transmission. The concept is inspired that the FEC is used to correct the transmission errors anyway and the key mismatch between Alice and Bob can be corrected together with the transmission errors. However, this approach is not applicable when a non-XOR encryption is used.

D. Privacy Amplification

Alice and Bob have to exchange information over a public channel during the previous steps, including preprocessing, quantization and information reconciliation. Unfortunately, Eve may be able to infer the secret key from these interactions. For example, a 2-bit syndrome leaked during the information reconciliation phase will narrow the search space to be explored by Eve by a factor of four. Hence Eve may find the key much quicker. As a countermeasure, privacy amplification allows Alice and Bob to distill a shorter but almost completely secret key from a common random variable about which Eve has acquired partial information [143]. This process is commonly implemented by using so-called universal hash families, which can be used for compressing keys, such as the leftover hash lemma of [70], [144], the cryptographic hash functions (e.g., secure hash algorithm) of [132], [145] and the Merkle-Damgard hash function [63].

According to the leftover hash lemma [70], [144], when an adversary know $t_k$ bits of a $n_k$-bit sequence, Alice and Bob can produce a key of length $L = n_k - t_k$ bits, over which Eve has almost no knowledge [146]. Considering the MD5 protocol as an example that maps a data string of arbitrary length to a data string of $L = 128$ bits, we have

$$\Phi : \{0,1\}^{n_k} \rightarrow \{0,1\}^L.$$  

(16)

In order to apply the MD5 hash function $\Phi$, Alice and Bob have to calculate the input sequence length $n_k$. Assuming that the information leakage ratio is $\eta$, the length of the secret key, $L$, is given by definition as

$$L = n_k(1 - \eta).$$  

(17)

Thus, in order to produce a secret key having a length of $L$ bits, Alice and Bob should generate at least

$$n_k = \left\lfloor \frac{L}{1 - \eta} \right\rfloor$$  

(18)

bits as their common random sequence, where $\lfloor \cdot \rfloor$ represents the floor operation.

The input sequence of the privacy amplification should have a uniform random distribution, otherwise, it will result in a weak key. Considering again MD5 as an example, the output of the MD5 function may pass the random test even when there are long runs of 0s and 1s in the input. However, this property leaves MD5 vulnerable to so-called dictionary attack. This can be enhanced by randomness extractors, i.e. by transforming biased probability distributions representing weak random sources into near-uniform probability distributions [147]–[149].

For high-speed real-time key generation systems, the imposed delay by privacy amplification is one of the limitations, which may be reduced by resorting to the techniques advocated in [150]–[152].

E. Summary

Having completed the four stages in Fig. 12, Alice and Bob will generate the same key. The key generated can then be used, wherever a common session key/password is required. Some applications of these techniques have been reported, including physical layer encryption [153], building a so-called 1-out-of-2 oblivious transfer [154], a cross-layer password-authenticated group key exchange protocol [155], [156], the design of spreading codes for spread spectrum communications [157], assisting the preloading of 6LoWPAN nodes wirelessly [158], [159] and a hybrid Merkle Puzzle-based key agreement scheme conceived for smart home applications [160].

V. DESIGN CONSIDERATIONS AND PERFORMANCE OPTIMIZATION

This section will introduce the relevant design considerations and possible methods to optimize key generation performance. We will first introduce the pertinent channel parameters, including the received signal strength indicator (RSSI) and CSI. We then review the different signal domains, namely the temporal, frequency and spatial domains. Finally, the duplex modes such as TDD, FDD and IBFD modes will be discussed.

A. Channel Parameters

The channel parameters are the most important characteristics used for key generation, since they represent the channel’s randomness. The most popular parameters are the RSSI and CSI. The latter can also be further divided into CIR and CFR.

1) Received Power/RSS/RSSI: These three terms, namely the received power, received signal strength (RSS) and RSSI, are used interchangeably in this paper. RSSI is used in almost all the wireless techniques to represent the link quality and it is also made public to the users, for example in the IEEE 802.11, IEEE 802.15.4, Bluetooth and LoRa, etc. This section will reveal the technical details of the RSSI-based solutions in different standards and their calculation in the real transceivers. The received power is mathematically defined in (4), but its calculation is more complicated in real transceivers. For example, the IEEE 802.16 standard specifies RSSI as (Section 8.3.9.2, [161])

$$\text{RSSI} = 10^{-\frac{G_{rf}}{10}} \frac{1.2567 \times 10^4 V_c^2}{(22B)R_i} \left( \frac{1}{N} \sum_{n=0}^{N-1} \frac{|y[t]|}{|r[t]|} \right)^2,$$  

(19)

where $B$, $R_i$ and $V_c$ are the ADC precision, input resistance and input clip level, respectively. Furthermore, $G_{rf}$ is the analog gain between the antenna connector and the ADC input,
IEEE 802.11 defines RSSI as a relative measure of the received power, with a range spanning from 0 to RSSI maximum (Section 18.2.3.3, [34]). However, different manufacturers may interpret the RSSI in different manners. For example, the RSSI maximum values of Cisco and Atheros are 100 and 60, respectively. Additionally, MAX2829, a WiFi transceiver, reports the RSSI in voltage [162]. It is very common in practice that the transmitter and receiver use different NICs and transceivers. However, because Alice and Bob quantize the measurements individually and independently, their heterogeneous devices are unlikely to have an impact on their key generation [163].

The RSSI is also available in the IEEE 802.15.4 standard. The CC253x, a TI ZigBee radio, calculates the RSSI by averaging the power received over eight symbol periods (128 µs) [100]. The RSSI reflects the signal strength, but not necessarily the link quality, since both the interference and noise will increase the signal strength. Therefore, IEEE 802.15.4 defines the link quality indicator (LQI), which characterizes both the signal strength and signal quality [35]. The CC253x calculates the LQI by

$$LQI = (CORR - a) \times b,$$  \hspace{1cm} (20)

where CORR is calculated by correlating the incoming frame with the first eight symbols following the start of the frame delimiter field, which ranges from 50 (lowest quality) to 110 (best quality), while $a$ and $b$ is chosen empirically. It would be interesting to explore how the LQI parameter may be exploited for improving the key generation performance.

The LoRa standard specifies a very high receiving sensitivity level of -148 dBm. The Semtech LoRa family of sx127x exploits both the instantaneous RSSI value in the register RegRssiValue ($Rssi$) and the packet RSSI value in the register RegPktRssiValue ($PktRssi$) (Section 5.5.5, [164]). The latter is an averaged version of the former. Additionally, the RegRssiValue is usually smarter than the RegPktRssiValue. When LoRa operates above 779 MHz, the RSSI is calculated as [164]

$$RSSI = \begin{cases} 
-157 + Rssi, & SNR \geq 0; \\
-157 + PktRssi + PktSnr \times 0.25, & SNR < 0.
\end{cases} \hspace{1cm} (21)$$

As discussed above, the RSSI is available in almost all the wireless techniques and it is provided by the COTS transceivers. However, it is left to the vendors to decide about its specific calculation method. Additionally, since the RSSI is averaged over the entire packet, it is a coarse-grained parameter. Hence the resultant KGR is usually limited. Finally, Jana et al. [70] found that the RSSI-based key generation is vulnerable to predictable channel attacks.

2) CSI: Compared to the RSSI, CSI is a fine-grained parameter, which can provide more valuable channel information. The CSI can be categorized into CIR, $h(\tau, t)$, and CFR, $H(f, t)$. Both are complex-valued, hence they have phase and amplitude, or real and imaginary parts. A multipath channel modelling technique was proposed for key generation in [59], which demonstrates that both CIR and CFR are beneficial source of randomness. An entropy extraction technique based on the CIR was conceived in [165].

The amplitude of the complex-valued CIR is exploited by the UWB systems [166]–[171]. By contrast, the channel phase was used for key generation both in wideband systems [172], [173] and in narrowband systems [133], [174], [175]. Compared to the amplitude, the phase has an extra pair of more attractive key generation features. Firstly, the phase is accumulative, which has inspired interesting applications, such as group and cooperative key generation [174], [175]. Secondly, the phases of all the channel paths, namely $\phi_i$ in (2), are distributed uniformly across $[0, 2\pi]$, regardless of the power. Yet, the phase is vulnerable to noise, carrier frequency offset and asynchronous clocks/clock drift at the receiver, hence it is less suitable for practical applications [176]. The study in [133] is the only one on a practical phase-based key generation system, which is implemented on the USRP [177].

As shown in (6), the CFR represents the channel response in the frequency domain, which can be readily estimated by OFDM systems. An example is given in Fig. 20, and the CFR is generated based on the configuration of the IEEE 802.11 OFDM system with 20 MHz channel spacing. Based on (5), the channel estimation can be formulated as

$$\hat{H}^{uv}(f, t) = \frac{Y(f, t)}{S(f, t)} = H^{uv}(f, t) + \hat{\omega}^{uv}(f, t).$$ \hspace{1cm} (22)

As mentioned earlier, OFDM has been widely used in the IEEE 802.11a/g/n/ac/ax standard family. Taking IEEE 802.11 OFDM with 20 MHz channel spacing as an example, there are 52 subcarriers out of 64 subcarriers in the long training symbol; the training symbols use publicly known pilot sequences, thus the receiver can use them for channel estimation. The channel responses of individual subcarriers are modelled in [60], which analyzes its autocorrelation and cross-correlation relationship.

The CSI represents fine-grained channel information, which can significantly improve the KGR [73], [178].
immune to the predictable channel attacks. However, the majority of the COTS NICs do not make the CSI publicly available, which limits its current adoption. There are two exceptions, however, namely the Linux CSI tools for the Intel 5300 NIC [179] and the Atheros NICs [180]. Alternatively, specialized hardware platforms can be used, such as USRP, WARP, etc. However, these platforms are expensive, therefore they are only used for prototyping and experimenting.

Apart from the above OFDM-based applications, a chaotic signal-based key generation was proposed in [181] for transmissions over frequency selective fading channels. The channel effects are characterized by the difference between the spectrum of the received signal and that of the transmitted chaotic signal. After the initial synchronization, both users can indeed generate the same transmitted signal, albeit it is not clear how to share the initial value for the first time.

3) Summary: A summary of RSSI-based and CSI-based key generation techniques is given in Table VIII, including the channel parameters, the related wireless techniques and testbeds, as well as the representative contributions, advantages and disadvantages. Generally speaking, the RSSI is usually readily available, but it tends to result in a low KGR due to its coarse-grained nature. On the other hand, solutions relying on the CSI typically have a better performance, but the application is usually limited to a few NICs and specialized devices.

B. Signal Domain

As shown in Fig. 21, the characterization of the wireless channel relies on three domains, namely the temporal, frequency and spatial domains. Each domain tends to exhibit randomness, which can be exploited for key generation.

1) Temporal Domain: The movement of objects and any reflectors as well as scatterers in the environment will affect the propagation path, which will cause unpredictable channel variation. The coherence time is defined as the duration over

\[
T_c = \frac{0.423}{f_d} = \frac{0.423c}{v f_c},
\]

where \(f_d\) is the Doppler spread, \(c\) is the speed of light and \(v\) is the moving speed. When the coherence time is longer than the symbol period, the channel undergoes slow fading; otherwise, fast fading will occur.

Slow Fading: Considering a pedestrian scenario at a walking speed of \(v = 1\) m/s, and a 20 MHz WiFi system operating at \(f_c = 2.4\) GHz, the coherence time is \(T_c = 52.6\) ms, while the symbol length of is \(\frac{1}{20 \times 10^6} = 0.05\) µs. Clearly, the symbol length is much lower than the coherence time, which indicates a slow fading channel. This is often the case for many WiFi-based and IEEE 802.15.4-based key generation applications.

When the channel is fluctuating at a near-constant rate, it obeys a wide sense stationary (WSS) random process. Zhang et al. modelled the autocorrelation function of the CIR and CFR based on a WSSUS channel model [60], and found that the frequency response of individual OFDM subcarriers is also a WSS random process. This indicates that a fixed sampling interval can be used for both the CIR and CFR in WSS channels. Their findings were experimentally validated in different environments in [74].

However, the channel is not necessarily fluctuating at a fixed rate, hence a constant probing rate tends to result in inefficiency. Therefore, adaptive probing was proposed for addressing this issue by adjusting the channel probing rate for accommodating the channel variations in real-time [63]. Explicitly, a proportional-integral-derivative-based algorithm was designed for exploiting the RSSI variation. Channel probing is first mathematically modelled in [63] and it is then validated by experiments conducted at different speeds, mobility types and sites, using COTS WiFi hardware.

Fast Fading: Key generation requires correlated two-way measurements, which will be adversely impacted by fast fading. Hence research efforts have to be invested in conceiving key generation techniques for fast fading environments, in particular vehicular communications [183]–[185]. Considering a vehicle driving at \(v = 60\) km/h and \(f_c = 5.9\) GHz as an example, the coherence time is 1.3 ms. The shortest airtime of a 20 MHz channel spacing IEEE 802.11 packet is 34 µs. The sampling interval, consisting of the packet airtime and short interframe space (SIFS) (more details can be found in Section VI-A1), is not negligible any more compared to the coherence time, which adversely affects the cross-correlation of measurements.

Zhu et al. tested key generation in vehicular scenarios at speeds up to 80 km/h using a WiFi Atheros chipset-based testbed [184]. They found the RSSI measurements very noisy, therefore, smoothing and level-crossing algorithms were used. They furthermore proposed an online parameter learning mechanism for adjusting the level crossing to the channel conditions. A KGR of 5 bps was finally achieved.

Static Environment: Another extreme case is the static environment, where the channel remains near-constant over time and no randomness can be provided. The limited randomness

---

3PCI-e interface is required for these NICs.
renders key generation challenging, hence innovative solutions have been proposed for introducing artificial randomness or using reconfigurable antennas.

Artificial randomness can be introduced by either the keying parties or by helpers [131], [186], [187]. A virtual channel is created in [186]. Alice is equipped with two antennas and controls the amplitude and phase of each symbol on each antenna. A helper node is introduced to broadcast jamming signals for varying the channel status in a static environment, but the jamming information is shared with Alice through a secure channel [131].

Using a reconfigurable antenna is another potential solution [66], [188]. An electronically steerable parasitic array radiator (ESPAR) antenna was designed having \( N_a = 7 \) elements [66]. The number of available beam patterns is \( 2^8 N_a^{-1} = 2^{18} \). The RSSI profile will change when a beam pattern is randomly selected to provide suitable randomness even in static environments.

However, the above solutions are not entirely general, because either helpers or additional reconfigurable resources or multiple antennas are required.

Gollakota et al. [189] designed a friendly jamming-based key exchange system, termed as iJam. The transmitter generates a random sequence referred to as a salt, which is modulated onto OFDM symbols. The transmitter will send two copies of the OFDM symbol back-to-back. The receiver will randomly jam one of the symbols, namely either the original one or its repetition. Because the receiver knows which symbol it has deliberately jammed, it can still decode the salt, but eavesdroppers cannot. The system has achieved 3 - 18 kbps KGR at a low KDR. However, the iJam system is different from the key generation concept, as it is not generating keys from the channel any more.

2) Frequency Domain: In a multipath environment, the signals undergo frequency selective fading. The coherence bandwidth, \( B_c \), is defined as [190]

\[
B_c \approx \frac{1}{\sigma_\tau},
\]

where \( \sigma_\tau \) is the root mean squared (RMS) delay spread imposed by the multipath propagation. When the signal bandwidth, \( B_s \), is higher than \( B_c \), it is a frequency selective fading channel. Otherwise, it is a frequency flat fading channel. For example, experimental results indicate that the RMS delay is above 100 ns in the 2.4 GHz indoor environment [191], hence the coherence bandwidth is

\[
B_c \approx \frac{1}{\sigma_\tau} < 10 \text{ MHz}.
\]

For an IEEE 802.11 20 MHz channel spacing OFDM system, the signal bandwidth of \( B_s = 20 \text{ MHz} \) is wider than \( B_c \), and thus the channel is frequency selective. Frequency selective channels exhibit increased randomness, which is desirable for key generation.

The randomness of the frequency domain can be exploited by wideband systems. A number of OFDM-based key generation systems have been reported in [59], [60], [73], [109], [178]. The multipath channel is modelled in [59], [60] while the frequency domain autocorrelation is also modelled in [60], where nine out of 52 subcarriers can be used for producing random keys. Liu et al. [73] designed an IEEE 802.11n-based key generation technique and achieved a substantial KGR, namely 90 bits per packet.

The frequency domain randomness can also be exploited in narrowband systems by channel hopping [67], [192], [193]. For example, the bandwidth of IEEE 802.15.4 is much narrower than that of IEEE 802.11 OFDM. However, it has 16 channels in the 2.4 GHz band, with 5 MHz channel spacing between adjacent channels. Wilhelm et al. generated 50 bits from 16 IEEE 802.15.4 channels in a static but frequency selective channel [67]. They proved that a 160-bit key can be generated if the number of IEEE 802.15.4 channels is increased to 40.

| Parameter | Technique | Testbed | Representative Contribution | Advantage | Disadvantage |
|-----------|-----------|---------|-----------------------------|-----------|--------------|
| RSS       | IEEE 802.11 | All COTS WiFi NICs | [63], [69], [70], [72], [144], [162] | Various interpretation by manufacturers | Vulnerable to predictable channel attacks |
|           | IEEE 802.15.4 | Sensor nodes such as MICAz, TelosB | [65], [69], [68], [134], [135] | Availability in most standards and transceivers | Coarse-grained channel information |
|           | LoRa | LoRa transceivers | [77]-[79] | | |
|           | Bluetooth | Smartphone | [76] | | |

| Parameter | Technique | Testbed | Representative Contribution | Advantage | Disadvantage |
|-----------|-----------|---------|-----------------------------|-----------|--------------|
| CIR, amplitude | UWB | Constructed by oscilloscope and waveform generator | [166]-[170] | Immune to the predictable channel attack | NOT currently available in most commercial transceivers |
| CIR, phase | Simulation | NA | Wideband systems [172], [173]; Narrowband systems [174], [175] | | |
|           | FM radio signals | Customized platforms, USRP | [133] | | |
| CFR, amplitude | IEEE 802.11 OFDM | Customized platforms, USRP and WARP | [73], [178] | | |
3) **Spatial Domain:** Multiple antenna techniques exploit the spatial diversity and have significantly improved the attainable key generation performance.

The family of MIMO schemes may be used for improving the KGR by exploiting the channel randomness in the spatial domain. Wallace *et al.* [58] derived the SKR for MIMO-based key generation schemes and evaluated their attainable performance both by simulation and indoor measurements. Chen *et al.* [110] investigated the performance of decorrelation techniques in eliminating the temporal and spatial correlation in MIMO systems. Quist and Jensen [194]–[197] conducted a systematic study of SKR maximization for MIMO-based key generation by optimizing both the beamforming vectors and the power allocation of the antenna elements.

MIMO-based key generation has been prototyped for the IEEE 802.11n standard. Zeng *et al.* [72] specifically designed a RSS and MIMO-based key generation system, which achieved four times higher KGR with the aid of three antennas compared to a single antenna protocol. Liu *et al.* [73] exploited both the frequency and spatial domain diversities simultaneously by using MIMO OFDM.

Multiple antennas can also be used for creating directional beams for randomizing the channel directions to mitigate the temporal correlations in static environments [198]. An ESPAR antenna can also be used for beamforming [66]. Precoding is another method of randomizing the signal and assisting key generation [199].

It is worth mentioning that MIMO solutions can also be used for multi-user access. IEEE 802.11ac supports downlink multi-user access, while IEEE 802.11ax enables both uplink and downlink multi-user access. Zhang *et al.* [200] demonstrated that the CSI in multi-user MIMO systems can be inferred either using explicit or implicit feedback. However, special techniques are required for multi-user MIMO-based key generation.

### C. Duplex Mode

There are three basic duplex modes for wireless communication systems, namely TDD, FDD and IBFD, as illustrated in Fig. 22. Key generation meets different challenges, when it is applied to practical communication systems operating in these modes.

1) **TDD Mode:** TDD refers to duplex communication links, where the uplink is separated from the downlink by the allocation of different time slots in the same frequency band. It is widely used in many systems, including WiFi, ZigBee, Bluetooth, LoRa, Long Term Evolution (LTE)-Advanced, and in the emerging 5G new radio mobile networks. In TDD systems, channel reciprocity is exploited for facilitating adaptive transmission to improve the system performance without any feedback overhead.

Fig. 22(a) illustrates the TDD-based channel sampling procedure, where Alice and Bob are allocated different time slots namely $t_a$ and $t_b$ for uplink and downlink channel probing at the same frequency $f_1$. Eve observes the channel between her and Alice at time $t_a$ over frequency $f_1$ and observes the channel between her and Bob at time $t_b$ over frequency $f_1$. The time delay $\Delta t$ includes the time of packet transmission and the time of switching from transmitting to receiving.

The sampling delay affects the cross-correlation of the measurements in TDD systems. Zhang *et al.* [61] systematically study a practical scenario by taking into account all relevant parameters including the sampling delay, the eavesdroppers’ location, the qualities of the legitimate and eavesdropping channels, the Doppler spread and pilot length. Their findings indicate that it is possible to tune the SKR by carefully designing the sampling delay and pilot length. For fixed sampling delays, interpolation filters can be employed to interpolate the value of a signal at unobserved points that lie in between two known samples [62], [134]. The reciprocity of interpolated measurements is typically improved and the effect of the normalized Doppler frequency on the correlation coefficient is also reduced.

The effect of sampling delay imposed on the channel correlation relies on whether the sampling time delay $\Delta t$ is smaller than the coherence time $T_c$. In a slow fading channel with pedestrian walking, the coherence time is about 50 ms. By contrast, $\Delta t$ can be configured to be on the order of $\mu$s. For example, $\Delta t = 60 \mu s$ is achieved in a WiFi system [60], [201], hence the channel’s cross-correlation is only modestly impacted by non-simultaneous measurements in this case. Thanks to the channel reciprocity of TDD, most of the existing key generation implementations are realized in the TDD mode, as exemplified by employing WiFi [201]–[203], ZigBee [204], Bluetooth [76], UWB [166], and LoRa [77]–[79].

However, the sampling delay may have a significant impact, when it becomes comparable to the coherence time. In fast fading channels associated with high mobility objects, the coherence time becomes very short. For example, the terminals move fast in scenarios associated with moving robots, vehicles, high speed trains, drones, etc. Additionally, since the sampling time delay increases with the number of antennas and users, it might become longer in multi-antenna and multi-user scenarios. Finally, this could occur even in slow fading channels. For example, LoRaWAN specifies a one second delay between the uplink and downlink transmissions, which is much longer than the coherence time in slow fading channels (about 50 ms). Further theoretical and experimental investigations are required to address this issue.

2) **FDD Mode:** In the FDD mode, the uplink and downlink transmissions operate at different carrier frequencies simultaneously. Fig. 22(b) illustrates the FDD-based key generation, where Alice and Bob probe the channel at the same time at the carrier frequencies of $f_1$ and $f_2$. Eve can observe both transmissions and then estimate the channel between her and Alice over frequency $f_1$ and the channel between her and Bob over frequency $f_2$. In contrast to TDD systems, FDD systems are not affected by non-simultaneous sampling, hence they are eminently suitable for supporting high mobility communications.

However, the frequency separation between the uplink and downlink results in non-reciprocal channels in FDD systems. Most of the reciprocal channel parameters used in TDD systems, such as the RSSI, channel gains, envelope and phase, can be quite different in FDD systems, depending on the
channel’s coherence bandwidth.

The existing FDD-based key generation solutions can be broadly classified into two categories: loopback-based protocols and frequency-invariant parameter-based approaches. Loopback-based protocols establish combinatorial channels with reciprocal channel gains with the aid of an additional reverse channel training phase [205]–[207]. Alice and Bob use combinatorial observations, such as $X^A X^B$, to generate secret keys. However, these protocols may complicate the channel sounding process and have potential security issues, since passive eavesdroppers might succeed in capturing the entire transmissions [208].

Another family of solutions relies on frequency-invariant parameters, including the eigenvalue of the channel’s covariance matrices [209], the multipath angle and delay [210], and the reconstructed CFR [211]. The channel’s covariance matrices represent second-order statistics, which differ by a fixed constant for the uplink and downlink [209]. However, they change slowly and the KGR is rather limited. The other two algorithms provide instantaneous reciprocal channel parameters, which are inspired by the fact that the propagation paths in the uplink and downlink are reciprocal in most FDD systems. The frequency spacing between the uplink and downlink sub-bands of LTE systems is much lower than the center frequency. For example, the center frequency of Band 1 (IMT) is 2100 MHz, while the duplex spacing is 190 MHz; the center frequency of Band 30 (WCS) used by AT&T in the United States is 2300 MHz, while the frequency spacing is only 45 MHz [212]. Field measurements disseminated in the literature have shown that the uplink and downlink transmissions travel along the same propagation paths and experience similar multipath clusters [213], [214]. If the channel parameters, such as the complex path gain, path delay and the angle of each individual path are accurately estimated from the pilot signals in one frequency band, the channel state in another frequency band can be calculated from these parameters based on the FDD channel model provided in [211].

However, the estimation accuracy is quite critical, because even small estimation errors may be magnified by the multiplication of the frequency difference between the bands. The required accuracy is not readily achievable for narrow bands and for single antenna systems. Nevertheless, both the operational and future wireless systems rely on increasingly higher bandwidths and more antennas, hence it becomes more suitable for key generation in FDD systems. But given the plethora of open issues, further studies are still required for accurately modelling and prototyping FDD key generation schemes.

3) IBFD Mode: IBFD has emerged as an attractive technique of increasing the throughput of next-generation wireless communication systems. Upon using IBFD, a wireless device is allowed to transmit and receive simultaneously in the same frequency band. Fig. 22(c) illustrates the key generation relying on the IBFD mode, in which Alice and Bob probe the channel at the same time at the same carrier frequency. The self-interferences (SI) of Alice and Bob are denoted as $SI_A$ and $SI_B$, respectively, which can be reduced close to the noise level using multi-domain SI suppression techniques [215]. Eve can only observe the superposition of messages between Alice and Bob.

The IBFD mode brings about some advantages for key generation. Firstly, it is not restricted by encountering the aforementioned non-simultaneous sampling in TDD systems and frequency separation in FDD modes. Secondly, it may provide a higher KGR given the same time- and frequency-domain resources. Finally, IBFD provides additional protection against Eve, because she will be confused by observing the superposition of simultaneous transmissions from Alice and Bob.

Several authors have studied key generation in the IBFD mode. Theoretical key generation approaches have been proposed for IBFD mode in [216], [217]. Practical key generation testbeds relying on the IBFD capability of USRP devices and near field communication (NFC) devices are demonstrated in [218] and [219], respectively.

4) Summary: In TDD systems, the channel reciprocity is adversely impacted by the non-simultaneous sampling. Under FDD operation, the channel responses are generally not similar, due to encountering different propagation paths. Although IBFD enables wireless users to transmit and receive
simultaneously over the same frequency band, it imposes new challenges due to the excessive self-interference. Table IX lists the factors influencing the reciprocity and their countermeasures, the representative contributions, advantages and disadvantages of the TDD, FDD and IBFD modes for key generation.

VI. IMPLEMENTATION AND APPLICATION SCENARIOS

A number of key generation prototypes have been implemented in the context of IEEE 802.11, IEEE 802.15.4, Bluetooth, UWB, LoRa, etc. This section will review these key generation applications. Three case studies are then used for exemplifying the key generation resource requirement and its implementation details.

A. Application Scenarios

As discussed, the wireless transceivers measure the RSSI and SNR, which can be readily used for key generation. Hence, many key generation prototypes have been built for IEEE 802.11, IEEE 802.15.4 and LoRa. Some of the most representative contributions are summarized in Table X.

1) IEEE 802.11: IEEE 802.11 is the most popular technique adopted for characterizing key generation. According to the IEEE 802.11 distributed coordination function-based MAC protocol, when the receiver successfully receives a DATA packet, it should reply with an ACKnowledgement (ACK) packet after waiting for a SIFS time interval. This interval is 10 $\mu$s in the 2.4 GHz band and 16 $\mu$s in the 5 GHz band. The DATA and ACK packets are thus perfect for probing, since their transmission time interval is on the order of $\mu$s [74], which is very desirable to get a high measurements correlation between these instances.

All the WiFi features, namely the frequency diversity of OFDM [60], spatial diversity of MIMO [72] and multi-user access capability of OFDMA [75], have been leveraged to enhance the key generation performance. Explicitly, [69], [70] represent the seminal key generation research, which uses IEEE 802.11a/g and extracts keys from the RSSI. However, the KGR is rather limited, on the order of 1 bit per second (bps). Because the RSSI is coarse-grained, the KGR can be improved by exploiting diversity both in the frequency and spatial domains. Zeng et al. [72] designed a three-antenna key generation system based on IEEE 802.11n, which achieves four times higher KGR than a single-antenna system. The KGR can be further enhanced by using OFDM for exploiting the frequency diversity [73], [178]; Liu et al. achieved a KGR as high as 360 bit/pkt employing a $2 \times 2$ MIMO OFDM system (3-bit quantization is used) [73]. Finally, Zhang et al. [75] leveraged the multi-user access feature in the latest IEEE 802.11ax amendment, which enables the AP to simultaneously establish keys with multiple users.

2) IEEE 802.15.4: Similarly to the IEEE 802.11, IEEE 802.15.4 also uses acknowledgement frame to confirm a successful reception, after waiting for the duration of an acknowledgement interframe spacing (AIFS). The length of AIFS is specified as 12 symbols in the standard (Section 10.1.3, [35]). For a data rate of 250 kps, each symbol contains 4 bits, and lasts $\frac{1}{512} \times 4 = 16 \mu$s; AIFS thus lasts 192 $\mu$s. The length of a typical IEEE 802.15.4 payload is between 30 to 60 bytes. The time interval is thus in the order of milliseconds and a high measurement correlation can be expected in a slow fading channel. Therefore, there are a number of prototypes and experiments relying on IEEE 802.15.4 [221].

WSNs can be used for industrial and environmental monitoring. The sensors remain at the same place once deployed, hence the channel variation is very limited. Kreiser et al. [204] investigated key generation in an industrial environment associated with two movable robot arms and a milling machine. Based on the experiments, the authors concluded that key generation does not work well in this kind of demanding scenarios. However, their conclusion is not entirely convincing, as it does not exploit the frequency selectivity of the channel at all. IEEE 802.15.4 is capable of operating across 16 channels at 2.4 GHz and legitimate users can switch their channel for exploiting randomness in the frequency domain [67], [222], as discussed in Section V-B2.

IEEE 802.15.4 is also widely used for body area networks (BAN) [223], [224], where the sensor nodes are mounted on the body. Hence in contrast to WSNs, usually sensor mobility is introduced by the host human. Hanlen et al. [225] demonstrated the randomness incurred by human activities, such as their movement in an office or running on a treadmill, which is sufficiently random for key generation. They achieved a KGR of 4 bps in theory and 2 bps by simulation. Ali et al. [68] evaluated the key generation performance in different scenarios. They considered

- high activity, where the host is working and walking,
- low activity, where the host is mainly sitting but occasionally moving,
- dynamic environment, where the devices are stationary but the surrounding channel is changing due to people walking around.

Their experiments demonstrate that key generation is feasible in all these three scenarios. They also show that it takes 15 to 35 minutes to generate a 128 bit key, when channel sampling is combined with the regular data transmissions, but does not require any dedicated communications. Li et al. [226] investigated the issues of group key generation in BANs. Four group models were proposed and then one of them was selected as an example for experimental evaluation.

3) Bluetooth: Bluetooth is operating at 2.4 GHz, which is an ISM band crowded by WiFi, ZigBee, etc. Therefore, Bluetooth divides the 2.4 GHz band into 79 channels and uses adaptive channel hopping (AFH) to avoid access collision. According to the specification, the slave will have to respond to the master on the same RF channel that is used by the master-to-slave transmission, which is known as the same channel mechanism of AFH (page 401, [36]). Additionally, the specification divides the physical channel into time slots, each with 625 $\mu$s and each packet can occupy up to five time slots, namely 3.125 ms (page 387, [36]); thus the maximum transmission delay between the master-to-slave and slave-to-master phases is 3.125 ms. These two features are desirable and beneficial for key generation, as the bidirectional transmissions operate at the same carrier frequency and the sampling
TABLE IX

| Duplex Mode | Reciprocity Influence Factor | Countermeasures | Representative Contribution | Advantage | Disadvantage |
|-------------|------------------------------|-----------------|----------------------------|-----------|--------------|
| TDD         | Non-simultaneous sampling    | Probing design, interpolation filter | [61], [62], [134] | Most of existing key generation implementations are realized in TDD | Low performance for faster moving scenarios |
| FDD         | Frequency separation         | Loopback based protocols, frequency-invariant parameters based approaches | [205]–[207], [209]–[211] | Support high speed scenarios | No experimental results |
| IBFD        | Self-interference            | Multi-domain SI suppression techniques | [216]–[219] | Full duplex, Higher key generation rate, Security | Potential debilitating effects of self interference |

TABLE X

| Paper | Year | Wireless Technique | Testbed | Parameter | Contribution |
|-------|------|--------------------|---------|-----------|--------------|
| [69]  | 2008 | IEEE 802.11a       | FPGA-based platform and Atheros NIC | CIR, RSS | One of the first key papers to apply key generation with IEEE 802.11 |
| [70]  | 2009 | IEEE 802.11g       | Intel 3945ABG NIC | RSS | Extensive experiments in different environments and mobility modes |
| [72]  | 2010 | IEEE 802.11a       | Intel 5300 NIC | RSS | Improving KGR by employing spatial diversity with multi-antenna |
| [73]  | 2013 | IEEE 802.11n       | Intel 5300 NIC | CSI | Improving key generation performance by using MIMO OFDM |
| [74]  | 2016 | IEEE 802.11g       | WARP | CSI, RSS | Evaluating key generation principles in different multipath environments including anechoic chamber, reverberation chamber and indoor office |
| [75]  | 2019 | IEEE 802.11ax      | Simulation only | CSI | Improving the efficiency of multi-user key generation by employing multi-user OFDMA |
| [66]  | 2005 | IEEE 802.15.4      | CC2420 | RSS | One of the first key papers to apply key generation with IEEE 802.15.4; used a reconfigurable antenna (ESPAR) |
| [62]  | 2010 | IEEE 802.15.4      | TelosB | RSS | Investigating multi-bit quantization and signal preprocessing algorithms (filtering and decorrelation) |
| [67]  | 2013 | IEEE 802.15.4      | MicaZ sensor mote | RSS | Exploiting channel responses of a frequency-selective environment in a static wireless sensor network |
| [68]  | 2014 | IEEE 802.15.4      | MicaZ sensor mote | RSS | Evaluating key generation feasibility in body area network with different channel variations |
| [76]  | 2014 | Bluetooth          | Smartphones with Broadcom chips | RSS | The first paper to apply key generation with Bluetooth; Investigating key generation performance with Bluetooth under heavy WiFi traffic by using a very wide bandwidth and random frequency hopping |
| [166] | 2007 | UWB                | Simulation | CIR | The first key paper to apply key generation with UWB; investigating the feasibility such as reciprocity and secret key rate |
| [169] | 2010 | UWB                | Waveform generator and oscilloscope | CIR | Evaluating channel reciprocity and spatial decorrelation of UWB-based key generation |
| [171] | 2015 | UWB                | Vector network analyzer | CIR | Evaluating key generation performance in indoor environment with LOS/NLOS |
| [220] | 2016 | UWB                | Integrated IR-UWB devices | CIR | A UWB-based key generation prototype on a platform with the full protocol stack; Performance evaluation in the static, occupied and mobile scenarios |
| [77]  | 2018 | LoRa               | LoRa/GPS Shield with sx1276 | RSS | Experiments in urban environment and deep in-building penetration; Using differential-based quantization to automatically adjust the channel variation |
| [78]  | 2019 | LoRa               | Multitech mDot with sx1272 | RSS | Extensive experiments with different setups, such as mobile and static scenarios, indoor and outdoor environment |
| [79]  | 2019 | LoRa               | SX1276RF1JAS evaluation boards with sx1276 | RSS | Experimentally evaluating effects of different LoRa parameters, such as spreading factor and bandwidth. Key generation validation with LoRaWAN protocol |

delay is small. A high correlation of the measurements can thus be obtained.

Surprisingly, there are very few papers that design key generation for Bluetooth and [76] is the first one. In this work, Premnath et al. considered a three-node scenario where Alice and Bob are exchanging information using WiFi and a node C wants to generate key with Alice. When the key generation probing is using Bluetooth, Node C first estimates the channel usage and then generates the frequency hopping sequence. Frequency hopping is beneficial, because in a wideband fading scenario, the different carriers may be deemed to fluctuate independently; hence a faded channel is
followed by an unfaded one. The keying parties, Alice and the node C, will then exchange probing packets based on the hopping sequence. The authors also compared that of WiFi-based probing. They implemented both key extraction schemes on typical smartphones and carried out extensive experiments. Their results demonstrated that under heavy WiFi traffic Bluetooth key generation outperforms WiFi key generation, when Alice conveys heavy WiFi traffic.

4) UWB: Wilson et al. [166] are the first authors to apply key generation for UWB systems and they derived the SKR. The majority of the practical UWB-based key generation solutions in the literature relied on a system consisting of a waveform generator and an oscilloscope [167]–[170] or a vector network analyzer [171]. Nevertheless, these sophisticated facilities are quite expensive, thus they are only suitable for experimental verification. Researchers have carried out extensive experiments both in indoor and outdoor LOS and non-line-of-sight (NLOS) scenarios for validating the channel reciprocity and spatial decorrelation characteristics of UWB systems [167]–[171]. However, to avoid practical pitfalls, it is important to note that typically the higher the bandwidth, the less valid the reciprocity becomes.

There is an exception that uses an integrated IR-UWB device [220]. The device operates in the band of [4.25, 4.75] GHz and can provide CIR estimation (real part) in the resolution of 1 ns. The device adopts the classical slotted ALOHA MAC protocol. In particular, the sampling delay between the pair of bidirectional measurements is 7.5 ms and the sampling interval can be 150.7 ms. The authors used a quantization algorithm for representing the CIR. Based on their evaluation in the static, occupied and mobile scenarios, the authors demonstrate that their system achieves a high grade of reciprocity as well as randomness and an acceptable KGR of 18 bps.

5) LoRa/LoRaWAN: In this specific context, key generation was only applied for short range communications, because the ranges of WiFi, ZigBee and Bluetooth are below 100 meters. Long range key generation was first reported in 2018 [77]–[79], [227] using LoRa, even though LoRa was standardized in 2015.

In contrast to WiFi or IEEE 802.15.4, there are several special issues affecting key generation in LoRa/LoRaWAN, which are listed as follows.

- The packet duration of LoRa is much longer than that of WiFi, ZigBee and Bluetooth, ranging from milliseconds to seconds. Additionally, LoRaWAN specifies a Receive Delay parameter between the uplink and downlink, which is one or two seconds. These two factors result in a high sampling delay between the bidirectional measurements, which degrades the measurements’ correlation.

- Because LoRaWAN uses the classic ALOHA MAC protocol without any channel sensing, there is usually an unavoidable duty cycle limitation for the LoRaWAN band. For example, the European Telecommunications Standards Institute regulates the ISM band’s channel utilization and hence the duty cycle of the LoRaWAN band in Europe is limited to 1%. This will significantly decrease the number of exchanged packets, hence the average KGR is extremely limited.

For example, considering a 10-byte payload and a LoRa configuration associated with spreading factor of 7, bandwidth of 125 kHz and code rate of 4/5, the packet airtime is 41.22 ms [228]. Therefore, the (minimum) sampling delay is 1.04122 ms; the LoRaWAN end devices can only transmit a single packet every 4.122 seconds, in order to meet the duty cycle regulation.

LoRa/LoRaWAN-based key generation is still in its early stage of development. In order to evaluate the LoRa key generation performance, Xu et al. [78] carried out extensive experiments in different mobility modes (static or mobile), environments (indoor or outdoor), distances (up to 4 km), data rates and motion types (walking, biking, or driving). Their results demonstrated the feasibility of LoRa-based key generation.

Ruotsalainen et al. [79] evaluated the attainable key generation performance for different LoRa modulation parameters, including spreading factors and bandwidths. These parameters will determine the airtime of the LoRa packets, which is directly related to the measurement correlation. They found that the KDR will be too high for SF > 10. Additionally, both the instantaneous RSSI and packet RSSI are used and the former is found to have a better performance in terms of its cross-correlation and KDR. They have then further extended their work by implementing key generation for the LoRaWAN protocol. They experimentally demonstrated that LoRaWAN-based key generation is indeed feasible, even when Alice and Bob are located seven km away from each other with both devices static, hence only experiencing environmental variation.

Zhang et al. [77] applied differential value-based quantization to capture the channel variation both in urban environment and deep in-building penetration. More explicitly, as shown in Fig. 14(b), the channel envelope may be varying from -123 dBm to -49 dBm in an urban test, but the consecutive samples are likely to be similar. Hence differential value-based quantization may be adopted for producing key sequences with high randomness and low KDR.

6) Summary: Key generation benefits from TDD-based techniques because of the channel reciprocity. Fortunately, the majority of the wireless techniques support the TDD mode, including IEEE 802.11/WiFi, IEEE 802.15.4, LoRa, etc. There are also some wireless techniques with no or very few key generation applications reported at the time of writing.

- **Cellular Networks**: There is one paper reporting LTE-based key generation with some preliminary results [229]. This is partly because there are fewer open platforms supporting cellular networks.

- **FDD Systems**: FDD LTE and NB-IoT operate in FDD mode. As discussed in Section V-C2, the channel reciprocity in such systems is challenged and correlated measurements are difficult to obtain.

- **SigFox**: Key generation requires bidirectional measurements between Alice and Bob. However, SigFox, for example, only allows up to 140 messages per day, which results in very inefficient sampling; there will also be a delay of 20 seconds between the uplink and downlink
messages [230], which significantly degrades the channel's correlation.

B. Case Study

1) Resource and Energy Analysis of a ZigBee-based Key Generation Protocol: The key generation protocol of Zenger et al. [64] only involves low-complexity operations, leading to low energy consumption. Explicitly, Zenger et al. implemented their full ZigBee-based key generation protocol on both a 32-bit ARM Cortex-M3 platform (EFM32GG-STK3700) as well as an 8-bit Intel MCS-51 (CC2531) chip, and calculated the resource and energy consumption. Additionally, a 32-bit [231] and an 8-bit [232] reference implementation of the elliptic curve Diffie Hellman (ECDH) key exchange, known as one of the most efficient PKC, were also realized for comparison.

The resources and energy consumption results are given in Table XI, where their key generation is seen to outperform ECDH. In particular, key generation requires much less computational resources than ECDH and it is much more energy-efficient. For example, when they are implemented in an 8-bit platform, ECDH requires about 8 times more code size, imposes 1289 times higher complexity, and consumes 98 times more energy than that of the key generation procedure of [64]. Since the key generation design is not optimized, it is expected that its resource and energy consumption can even be further reduced. Key generation is hence eminently suitable for IoT devices, constrained by their computational capability and battery power.

2) A WiFi-based Demo with Specialized WARP Hardware: A key generation demonstration and testbed has been created at the University of Liverpool, UK. A demonstration video is included as the multimedia supplement material for this paper. The experimental setup and the associated graphical user interface are shown in Fig. 23 and Fig. 24, respectively. The demo is based on a specialized hardware platform, namely WARP boards [233], but it may also be readily ported to other wireless testbeds with the necessary changes made to the channel probing part. The protocol is implemented using the Python language.

The protocol implementation is detailed as follows.

- Channel Sampling: The WARP 802.11 Reference Design, which is compatible with commercial WiFi, is used for accessing the WARP hardware [234]. The DATA packet and its corresponding ACK packet, which are standard WiFi packets, are used for bidirectional probing. The sampling delay between the DATA and ACK packets is configured as 64 $\mu$s, therefore highly correlated received power measurements can be obtained.
- Packet Match: Because the testbeds will receive all the WiFi broadcast transmissions in the air, such as the Beacon frames of other WiFi networks, reliable packet selection is required for capturing the packets having the correct receiver MAC address. In addition, there may be packet loss events during the transmissions, resulting in inconsistency between the packets received by Alice and Bob. The difference between the timestamps of the paired DATA and ACK packets is 64 $\mu$s in this demo. The packets are further refined by comparing their timestamps of the packets at Alice and Bob.
- Quantization: Mean value-based quantization is used as an example of converting the analog measurements into a binary sequence.
- Information Reconciliation & Privacy amplification: The BCH-based secure sketch [129] is adopted. The SHA256 hash function [235] is used for privacy amplification.
- Randomness Test: A python-based implementation of the NIST randomness test suite is used [108].

3) A WiFi-based Implementation Using COTS Hardware: Prophylaxe [236] is a German project aiming for creating practical wireless physical layer security for IoT, which was completed with great success.

Zenger et al. [237] created a key generation implementation using COTS WiFi platforms, namely a WRT54GL WiFi router and a Nexus 4 smartphone. The WRT54GL router is an open source hardware platform and SoftMAC [238] is used for enabling channel measurements on a per frame basis. The NIC should also support a virtual monitor mode and raw packet injection, which will allow devices to perform communications even when they are not associated to a particular network. Radiotap header [239] is a particular header that is designed for some WiFi NICs to report the characteristics of the frames, including timestamps, channel, RSSI, etc.

Nexus 4 is a partly open source hardware, which is produced by LG and Google. Root access to the file systems is required. An open-source WiFi FullMAC driver [240] is supported, but it is very complex. Fortunately, there is an experimental open-source project based on the SoftMAC driver, WCN36xx [241]. This is beneficial, since it allows the developer to integrate their manipulation in the same manner as for the router.

A full implementation is then performed. The channel measurements are carried out by using IEEE 802.11 management frames, namely the probe request and probe response frames [242]. Graphical user interfaces are created for both the router and the smartphone. It is also integrated into the WiFi WPA/WPS protocol. Experiments have been carried out both in stationary and mobile environments.

4) Summary: The results and implementation aspects portrayed in this case study section are generally applicable to
TABLE XI
ENERGY AND RESOURCES REQUIREMENTS OF KEY GENERATION PROTOCOLS AND ECDH

| Protocol          | Platform            | Resources | Energy                              |
|-------------------|---------------------|-----------|-------------------------------------|
|                   |                     | Code Size (kb) | # of Cycles | Computation (mJ) | Communication (mJ) | Total (mJ) |
| Key generation    | 32-bit ARM Cortex-M3 | 1.033     | 302,297    | 2.24             | 0.18              | 2.43       |
| Key generation    | 8-bit Intel MCS-51  | 1.137     | 1,345,205  | 5.20             | 0.18              | 5.39       |
| ECDH              | 32-bit ARM Cortex-M3 | 5.918     | 38,774,000 | 100.96           | 0.06              | 101.02     |
| ECDH              | 8-bit Intel MCS-51  | 8.749     | 1,734,400,000 | 528.45          | 0.06              | 528.51     |

Fig. 24. The graphical user interface of the key generation demonstration at the University of Liverpool, UK.

all the key generation protocols. However, when applied in different wireless techniques, such as WiFi or ZigBee, the channel probing will differ. By contrast, the remaining three steps, namely the quantization, information reconciliation and privacy amplification can be the same.

VII. KEY GENERATION FOR MULTIPLE PLAYERS

The previous sections only involved a pair of legitimate users, Alice and Bob. This section will extend these concepts to scenarios with multiple players, involving Alice, Bob and third parties. The third parties may act as
- keying parties that wish to establish a common group key,
- relays that assist the key generation process;
- attackers that passively eavesdrop or actively disrupt the key generation process;

This section will cover all the above three scenarios.

A. Multi-User/Group Key Generation

Key generation usually works between a pair of users by establishing a pairwise key between them. However, there is a clear need to establish keys among multiple nodes in some scenarios, where a number of users have to exchange confidential information.

1) Star Topology: Star topology-based networks are the most well investigated scenarios of multi-user/group key generation. Ye et al. [243] studied the SKR of group key generation using graph theory. The existing protocols can be categorized into reference-channel based schemes and OFDMA-based schemes, as portrayed in Fig. 25.

As shown in Fig. 25(a), Liu et al. [135] proposed a reference-channel based scheme, which first randomly selects a central node, \( n_c \), and then a reference node, \( n_{ref} \). The RSS between the central and reference nodes, \( p_{ref} \), is calculated and defined as the reference channel. By performing the bidirectional probings, the RSS of the downlink and uplink
channels between the central node and the \( u^{th} \) node can be measured, which are denoted as \( p_{dl}^u \) and \( p_{ul}^u \), respectively. After completing all the probing, the central node will calculate the difference of the signal strengths (DOSS) between the \( u^{th} \) uplink channel and the reference channel, namely

\[
\Delta p^u = p_{ul}^u - p_{ref}
\]  

and then transmits \( \Delta p^u \) to the node. The \( u^{th} \) node then calculates

\[
p_{dl}^u - \Delta p^u = p_{dl}^u - p_{ul}^u + p_{ref} \approx p_{ref}.
\]

Thus, all the participating nodes will extract the common secret, namely \( p_{ref} \). Xiao et al. [244] designed a similar scheme. They converted all the RSS values to binary keys first, \( k_{ul}^u \) and \( k_{dl}^u \). Instead of calculating their DOSS, the central node then calculates

\[
\Delta k^u = k_{ul}^u \oplus p_{ref}.
\]

The other operations are the same as those of the scheme in [135].

The reference-channel based scheme has to carry out pairwise channel probing between two users, which was found inefficient by the study of Jin et al. in pairwise-based multi-user key generation [245]. A pair of scheduling algorithms were discussed, namely serial and parallel probing.

Inspired by the desire of conceiving secure multi-user access, Zhang et al. [75] designed an efficient OFDMA-based multi-user key generation protocol and applied it to the latest IEEE 802.11ax standard as a case study. As shown in Fig. 25(b), the central controller and the nodes will share the subcarrier allocation information in advance. The central controller first broadcasts a downlink packet to all the stations, which carry out channel estimation. All the nodes will then commence their uplink transmissions simultaneously on their pre-allocated subcarriers, which will not cause inter-user interference. The central controller can then carry out uplink channel estimation for each user. A common key, \( k^u \), can be generated between the AP and the \( u^{th} \) node. This scheme intelligently exploits the multi-user access technique and significantly reduces the channel probing overhead.

Once an individual key has been setup in multi-user key generation, Wei et al. [246] designed a group key distribution algorithm. The AP will generate the group key as

\[
k_G = k_1 \oplus ... \oplus k_N.
\]

It will then mask the group key as \( k_G \oplus k^u \) and transmit it to the \( u^{th} \) user. Finally, the \( u^{th} \) user extracts the group key by the exclusive-OR operation.

2) Other Topologies: Group key generation protocols have also been conceived for other network topologies. Thai et al. [247] proposed a protocol for mesh topologies, but a pairwise channel probing was performed between different nodes. Wang et al. [174] designed a roundtrip-based protocol, where the nodes form a circle. Channel sounding was again carried out on a pairwise basis. However, the protocol relied on employing the channel phase, which limited its practical application, because accurate phase estimation is rather challenging. Xu et al. [248] maximized the group key rate for a ring network by studying the time required for channel estimation among all the users.

B. Relay-Based/Cooperative Key Generation

Key generation usually supports the interactions of a pair of users and thus it can only exploit the randomness of the link between them, which limits the amount of randomness and the communication range. In this scenario, the attainable performance can be improved by employing relaying/cooperating nodes for reaping the randomness between the legitimate users and relay, as shown in Fig. 26. For example, there is typically a dominant LOS link between a pair of unmanned aerial vehicles (UAVs), resulting in near-constant channel. A ground station can act as the relay and whilst still LOS-oriented, this relay-UAV channel usually has higher entropy than the direct LOS UAV-UAV channel, which can be exploited for key generation [249].

1) Trusted Relay: Trusted relays will actively participate in the key generation process and share the randomness with the legitimate users, which can thus significantly improve the key generation performance.

Lai et al. [250] proposed a cooperative key generation solution, where Alice, Bob and the relay node exchange packets with each other and separate keys can be established.
between each pair of users, namely $k^{ab}$, $k^{ar}$ and $k^{br}$. The relay will then broadcast $k^{ar} \oplus k^{br}$; Alice and Bob can then get $k^{br}$ and $k^{ar}$, respectively. Since Eve also gets $k^{ar} \oplus k^{br}$, Alice and Bob will use either $(k^{ab}, k^{ar})$ or $(k^{ab}, k^{br})$ as their key. The channels between the relay and the legitimate users are then exploited. The scheme was then also further extended to multiple relays.

Based on the same cooperative model, Wang et al. [251] derived the upper and lower bounds of the SKR in the face of a passive eavesdropper. The authors investigated a practical system relying on classic modulation schemes, such as PSK or QAM. Finally, they optimized the protocol for achieving tight bound of the SKR.

Shimizu et al. [252] designed relaying-aided schemes, namely an amplify-and-forward scheme, a signal-combining amplify-and-forward scheme, a multiple-access amplify-and-forward (MA-AF) scheme, and an amplify-and-forward with artificial noise scheme. They showed by their simulations that the MA-AF scheme has the best performance of SKR.

However, the above body of literature was based on single antenna systems. The authors of [253] and [254] further extended these ideas to MIMO relays and investigated the power sharing amongst the antennas. In particular, Chen et al. [254] found that their proposed power allocation scheme improves the SKR from 15% to 30% at low power, when compared to equal power allocation.

In some special cases, the node is not directly participating but only assisting in the key generation process, for example, by transmitting artificial interference for improving the channel’s randomness. This is particularly helpful in static environments [131], as mentioned in Section V-B1.

2) Untrusted Relay: On the other hand, security concerns arise when the relay is untrusted. An untrusted relay will help the key generation process, for example by forwarding messages, but potentially only owing to its desire to reveal the keys generated. Special measures should thus be taken.

Thai et al. [255] investigated scenarios with non-colluding, partially colluding and fully colluding relays, where all the users are equipped with multiple antennas. They concluded that key generation is feasible even in the face of fully colluding relays. They also found that there exists an optimal number of antennas for the untrusted relays.

Waqas et al. [256] borrowed the concept of social relationships to model the relay nodes. In particular, they modelled the actions of untrusted relay as the social reciprocity relationship, where the user cooperation should be based on the mutual benefit. Coalition game theory was used to select the optimal relays.

In order to tackle the malicious actions of untrusted relays, a retrodirective array (RDA) was used by the relay nodes in [257]. Since the RDA acts in a similar manner to a mirror, it will reflect the incoming signal by appropriately adjusting the phase conjugation, but it will not be able to store or decode the signal. Key generation will therefore be enhanced by the RDA owing to forwarding the messages, but imposing no threat.

C. Attacks and Countermeasures

Similarly to classic wireless communications systems, key generation is also vulnerable both to passive eavesdropping and to active attacks. Passive eavesdroppers listen to all the key generation transmissions and endeavor to generate the same keys as the legitimate users. On the other hand, active attackers aim for disrupting the key generation process. Some attacks are summarized in [82], but key generation attacks have received relatively limited research attention. This section will review the known attacks and their countermeasures.

1) Passive Eavesdropping: The spatial decorrelation of received signals is based on Jakes’ model, which indicates that the channel will be uncorrelated when a third party is located half-wavelength away [101]. The key generation performance under the Jakes’ model can serve as a benchmark [61]. However, this model requires infinite and uniformly distributed scatterers around the user, which may not be the case in real environments.

Substantial research efforts have been invested into evaluating key generation security against passive eavesdropping both by simulation and experimental studies [74], [258]–[262]. He et al. [258] carried out comprehensive investigations on the link signature (LS)-based security, which mainly includes secret key generation and physical layer authentication [263]. They first investigated different channel correlation models, including one-ring model, two-ring model, elliptical ring model and a far scatterer-ring model. These models were then evaluated by simulations. They have also carried out the experimental verification of the simulation results both in indoor and outdoor environments. Based on the simulation and experimental results, it was found that half-wavelength distance decorrelation is only valid in rich scattering environments.

Zenger et al. [260] created an automated antenna positioning platform for repeatable experiments, in order to evaluate both the cross-correlation and the mutual information of the legitimate users and eavesdroppers. Testbeds of the IEEE 802.15.4 standard operating at 2.4 GHz were used and the RSS was relied upon as the keying parameter. The authors found that cross-correlation between Alice and Bob is affected by Eve’s antenna position when Eve is located within three wavelengths. This will help the legitimate users to detect the presence of eavesdroppers by evaluating their channel correlation.

To expound further, Zhang et al. [74] carried out extensive IEEE 802.11 OFDM-based experiments at 2.4 GHz and at different multipath levels, including those conducted in an anechoic chamber (no multipath), in an indoor office (typical multipath) and in a reverberation chamber (very strong multipath). They found that neither CSI-based nor RSS-based key
generation is secure, when there is no multipath propagation. On the other hand, key generation is quite secure in the face of strong multipath, as the eavesdroppers experience a channel that is uncorrelated with the legitimate link even when they are only a few centimeters away. Furthermore, it was observed that the eavesdropper's channel response varies significantly versus the distance, when they are within about two wavelength from the legitimate users in an environment having strong LOS (anechoic chamber). This observation indicates a limited validity of Jakes’ model, which may due to the mutual coupling [264] and near field effects. Similar effects were also observed in UWB measurements [169]. Their experimental results indicated that it may not be optimal for eavesdroppers to locate too close to the legitimate users.

2) Active Attack: In contrast to passive eavesdropping, active attackers aim for interrupting the key generation process by injecting jamming signals [265], man-in-the-middle (MITM) attack [266] and manipulative attack [267], [268].

Zafer et al. [265] introduced both a simple jammer transmitting at a fixed power and a smart jammer that can estimate channel. They defined a new efficiency metric that quantifies the minimum number of messages to be exchanged per secret key bit. They found that the key generation efficiency is dramatically reduced as a function of the jamming power. In terms of countermeasure, Belmega et al. [269] proposed to use channel hopping or power spreading; they also equipped the key generation parties with energy harvesting capabilities, which will harvest energy from the malicious jamming power.

Ebert et al. [266] designed a MITM attack poisoning the quantization stage and carried out the experimental validation of their solution using off-the-shelf hardware. They demonstrated that an intentional sabotage attack may indeed result in a high KDR and that Eve may acquire up to 47% of the generated key bits.

Jin et al. [267], [268] took a further step by conceiving a manipulative attack, which aims for forcing legitimate users agree on some manipulated keys. More particularly, they designed a signal inject attack and a channel control attack. The authors later proposed a practical countermeasure, namely the PHYsical layer key agreement with User Introduced Randomness (PHY-UIR). The effectiveness of the method was validated both by simulations and experiments. However, the protocol was later found vulnerable in [270] to a session hijacking attack.

VIII. DEVICE AUTHENTICATION

A complete security system should meet the requirements of authentication, confidentiality and integrity. Confidentiality and integrity can be handled by encryption, which is assisted by the key generation process. However, key generation itself usually cannot be used for the authentication, hence existing key generation research simply assumes that both Alice and Bob are legitimate users.

Research attempts have been made to achieve both device authentication and key generation simultaneously in [271], [272]. However, the scheme proposed is only applicable to wireless BANs, where the devices should be mounted on the same person, which is not generally applicable.

Therefore, authentication techniques are necessary and this section introduces a complete wireless security architecture, which will achieve both device authentication and confidential transmission, as portrayed in Fig. 27. Some candidate techniques in this context are physical layer authentication and radio frequency fingerprinting (RFF)-based identification, as illustrated in Fig. 28. The former relies on the channel variations, while the latter is based on the random hardware features of wireless transceivers.

A. Physical Layer Authentication

Physical layer authentication constitutes another branch of physical layer security, which identifies the wireless devices based on the channel characteristics [273].

Fig. 28(a) considers a scenario, where Alice is the transmitter and Bob is the receiver, who tries to authenticate if the signal is transmitted by Alice. Alice will transmit at a rate lower than coherence time, while Bob will continuously estimate the channel attributes, and compare their values to his previous records. When a pair of consecutive channel estimates are similar to each other, Bob concludes that the signal is indeed transmitted from Alice [274]. A spoofer, Eve, may impersonate Alice. According to the spatial decorrelation, the Eve-Bob link will have different channel features from the Alice-Bob link, when Eve is located at a certain distance away from Alice. Therefore, when Bob detects any anomaly of the received signal, it declares a potential hijack [275].

Similar to the key generation process, physical layer authentication has also been designed for exploiting different channel parameters, including the RSS [276], the CIR [277], the CFR [263]. Again, the CIR and CFR usually provide better authentication reliability, because they are fine-grained. Since the channel fluctuates in an unpredictable manner, machine learning was introduced for adaptively learning and processing the complex-valued time-varying channel [278].

Although substantial research advances have been made, there are still numerous challenges preventing physical layer authentication from practical deployment [273], some of which are listed below:

- Low reliability. Frequent and continuous sampling of the channel attributes are required for physical layer authentication. This may be difficult for many IoT devices, since
sensor nodes may turn into sleep mode and the wireless connection is lost. The channel will have changed significantly over the dormant period and the reliability of this technique will be significantly impacted [279].

- Integration with upper-layer authentication schemes and network infrastructure. The principle of upper-layer authentication is quite different from its PHY counterparts. Additionally, physical layer authentication mainly operates in device-to-device mode, but wireless networks are usually large scale, with many devices not directly connected.

- Complex heterogeneous networks. The mobile devices will roam across the coverage area of different base stations, which requires a frequent handover. This will introduce additional complexity and latency, which may not meet the timing requirements.

B. RFF Identification

RFF identification authenticates the wireless devices based on their hardware imperfections resulting from the manufacturing process (see [280]–[282] and references therein). These hardware features are unique, permanent and cannot be tampered with, which are ideal for device authentication.

As shown in Fig. 28(b), RFF identification consists of two stages, namely training and classification. During the training stage, the authenticator, Bob, will collect wireless signals from a device, extracts some features and saves them in a database. When the device wishes to join the network again, Bob will extract the same features from the received wireless signals, compare them against the database, and then classify the device identity.

RFF can be categorized into transient features and modulation features [280].

- Transient features represent the turn-on/off transient or signal variation, such as the envelope of the transient signals [283]. However, it is very sensitive to both the device position and to the antenna polarization.

- The modulation features are stable and extracted from the baseband signal, such as the amplifier’s non-linear characteristics [284], the carrier frequency offset [285], etc. These features can be captured by SDR platforms, such as USRP.

The classifier is designed for differentiating the devices based on the features extracted. The classification performance can be enhanced by combining multiple features [285]–[287]. Machine learning algorithms, such as support vector machine (SVM), may also be readily exploited [286]. Sometimes it is challenging to identify and extract the best feature. Hence, deep learning may be adopted to directly process the raw I/Q samples without using a particular feature [282], [288], [289].

RFF identification has been prototyped in conjunction with a number of wireless techniques, such as WiFi [285], [290], ZigBee [287], [291], Bluetooth [292] and LoRa [293]–[295], just to name a few. Because RFF identification exploits the features of wireless transceivers, it is a perfect candidate for key generation in an integrated security framework [87]. However, there are also some challenges to be tackled, when designing a reliable and robust classification system.

- Rigorous modelling. The transceiver hardware chain has many hardware components, such as the oscillator, mixer, power amplifier, analog-to-digital converter, filter, etc. Many of them may exhibit nonlinear characteristics. Despite some research attempts [296], a rigorous RFF modelling is challenging. On the other hand, it is desirable to gain a comprehensive understanding of the hardware effects.

- Channel effect. The RFF is extracted from wireless signals, which are affected by the channel fading. Since the training and classification usually do not occur at the same place, the classification performance is impaired by the different multipath fading.

- Expensive authenticator. RFF identification requires raw I/Q samples to extract fingerprint, which are usually not available in the COTS devices. Therefore, expensive devices such as oscilloscopes and spectrum analyzers are used in the testbed, but unfortunately they cannot be used in operational networks. SDR platforms such as USRP are also often used, which still cost hundreds or thousands of US dollars.

- Classification capacity. A single gateway of IoT networks may serve thousands of end devices. Intuitively, the more devices have to be authenticated, the more complex classification algorithm and the higher requirements on the authenticator hardware specification. The capacity of the RFF identification thus requires more research [297].
IX. POTENTIAL PITFALLS AND FUTURE RESEARCH

This section first covers the ongoing debate on how attractive key generation is as a practical security solution. We then provide a number of future research directions in order to bridge the gaps.

A. Is Key Generation An Attractive Security Solution?

Although there have been a number of key generation prototypes relying on various wireless techniques, a natural question arises, is key generation really an attractive security solution?

Trappe [279] discussed a number of challenges that physical layer security is facing before it can be adopted to protect operational communications systems. In terms of key generation, he identified the following hurdles:

- **Weak adversary model.** The key generation research community often only considers passive eavesdropping but underestimates the capabilities of active attackers. These weak models are not recognized by the cryptographic community.

- **Idea assumption of wireless channels.** The assumption of the WSSUS and Jakes’s model may not be valid in real scenarios. A sufficiently random and dynamic channel may not be available.

- **Transceiver imperfection.** Practical impairments of the transceivers will impact channel reciprocity, such as the amplifier discrepancies and transceiver burn-in and frequency drift.

Robyns et al. [298] discredit physical layer security, including keyless transmission, key generation and physical layer identification. In particular, the authors criticize that key generation requires an uncorrelated eavesdropping channel and the public discussion leaks information to eavesdroppers. Indeed, we concur that many of these issues have to be resolved when we consider realistic systems.

As a counter-argument, it was argued by Trappe [279], [299] that physical layer security/key generation will be indeed an ideal candidate to complement the classical cryptography for securing low-cost IoT devices. This is because IoT devices use the majority of their resources for supporting their core functions and there is very few of them left for security, which makes them vulnerable to attacks. On the other hand, key generation aims for exploiting existing radio resources and communications without imposing substantial additional energy consumption [299]. In addition, as demonstrated in Section VI-B1, key generation implementation costs very few computational resources. Therefore, it is deemed to be suitable for the low cost IoT devices with limited energy and computational resources.

B. Vision for Future Directions

Despite this promise, there are still numerous research challenges to be addressed for adopting key generation as a practical and reliable security solution. Some suggestions for future research are given below.

**Key generation for 5G.** 5G has adopted numerous physical layer techniques, such as massive MIMO and mmwave communications. These technologies provide more flexibility for supporting multiple users. However, the research of multi-user key generation in mmwave massive MIMO wireless communications is fairly open. Since the pilot overhead scales linearly with the number of antennas, it becomes impractical for Alice and Bob to complete their channel probing within the coherence time of massive MIMO TDD systems [86]. Furthermore, when the base station generates secret keys with multiple users in sequence, the complexity escalates with the number of users. Jiao et al. [300] proposed a key generation scheme for single user mmwave massive MIMO systems. Explicitly, they exploited the virtual angle of arrival (AoA) and angle of departure (AoD) characteristics of the channel to reduce both the probing time and the complexity. They imposed a small perturbation angle on the AoA as the common randomness for improving the SKR [301]. However, both the theoretical analysis of the SKR and the design of practical protocols require further investigations for multi-user key generation in mmwave massive MIMO systems.

**Key generation with non-reciprocal channel.** Key generation is particularly challenging in scenarios where channel reciprocity and randomness may not be readily achieved, for example in FDD systems, static environments, and vehicular communications, etc. These scenarios, however, are very common in the IoT. For example, NB-IoT is a popular IoT standard operating in FDD mode. Many IoT devices are stationary and the environment is usually static or quasi-static. Although there are some research attempts to circumvent this problem, unfortunately the existing solutions are not general and they all need additional hardware or other resources.

**Key generation in large scale fading channels.** As discussed in Section II-C the communication ranges of LPWAN are on the order of km, and the channel is subject to large scale fading. Different from small scale multipath fading, large scale fading changes much slower, which limits the randomness. While there is some preliminary work on key generation for LoRa presented in Section VI-A5 and some theoretical exploration on key generation in large scale fading channels [102], it requires more investigation. It will be quite important as numerous IoT applications operate in such environments.

**Key generation security analysis.** As mentioned above, the attack model is weak. It is strongly recommended to enhance the security analysis and the investigation of both passive and active attacks. Since the keys generated support the cryptographic schemes, it is necessary to carry out the associated crypto-analysis, rather than pure wireless-based attack analysis.

**Bridging cryptography and wireless communities.** The concepts of classical cryptography and key generation are rather different, resulting in different evaluation metrics for their security levels. A common language bridging both community is extremely desirable for unveiling the pros and cons of these techniques [299]. The hybrid cryptosystem relying on the amalgam of key generation and symmetric encryption may be deemed to be an intriguing starting point.

A final positive perspective offered by the authors of this treatise is that the Chinese Micius experiment demonstrated QKD over satellites across a distance of 1200 km [123],
Hence key distillation in the classical domain is also a promising frontier research area.

X. Conclusions

This article provided a comprehensive survey of random key generation from wireless channels, systematically reviewing the topics of key generation fundamentals, protocol design considerations, implementational case studies, multiplexer key generation and device authentication. We first introduced the fundamentals, including the random sources, principles, and followed by information-theoretic modelling and the pertinent evaluation metrics. A four-stage protocol was then proposed, including channel probing, quantization, information reconciliation and privacy amplification. We then examined the relevant design aspects, such as the channel parameter selection, the temporal, frequency and spatial signal domains, as well as duplex mode including the TDD, FDD and IBFD modes. Efforts dedicated to implementing and prototyping key generation protocols were also included. The key generation was then extended to multi-player scenarios where the third parties act as keying parties, attackers, or relays. Device authentication was briefly introduced, which can assist in identifying the keying parties in key generation. The article concluded with suggestions for future research and a list of potential pitfalls as well as scientific arguments concerning the pros and cons of this alluring frontier research subject.

Acknowledgements

We would like to thank Mr Yan Wang for his hard work implementing the key generation demonstration system during his final year project.

References

[1] A. Al-Fuqahe, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, “Internet of things: A survey on enabling technologies, protocols, and applications,” IEEE Commun. Surveys Tuts., vol. 17, no. 4, pp. 2342–2376, 2015.

[2] A. Burg, A. Chattopadhyay, and K.-Y. Lam, “Wireless communication and security issues for cyber–physical systems and the internet-of-things,” Proc. IEEE, vol. 106, no. 1, pp. 38–60, 2018.

[3] J. Manyika, M. Chui, P. Bisson, J. Woetzel, R. Dobbs, J. Bughin, and D. Aharon, The Internet of Things: Mapping the value beyond the hype. McKinsey Global Institute, 2015.

[4] J. Granjal, E. Monteiro, and J. S. Silva, “Security for the internet of things: A survey of existing protocols and open research issues,” IEEE Commun. Surveys Tuts., vol. 17, no. 3, pp. 1294–1312, 2015.

[5] A. Mukherjee, “Physical-layer security in the internet of things: Sensing and communication confidentiality under resource constraints,” Proc. IEEE, vol. 103, no. 10, pp. 1747–1761, 2015.

[6] Y. Zou, J. Zhu, X. Wang, and L. Hanzo, “A survey on wireless security: Technical challenges, recent advances and future trends,” Proc. IEEE, vol. 104, no. 9, pp. 1727–1765, Sep. 2016.

[7] A. Grau, “How to build a safer Internet of Things,” Feb. 2015, accessed on 21 April, 2020. [Online]. Available: http://spectrum.ieee.org/telecom/security/how-to-build-a-safer-internet-of-things

[8] E. Marin, D. Singelée, F. D. García, T. Chothia, R. Willems, and B. Preneel, “On the (in) security of the latest generation implantable cardiac defibrillators and how to secure them,” in Proc. 32nd Annual Conf. Computer Security Applications, Los Angeles, CA, USA, Dec. 2016, pp. 226–236.

[9] F. D. García, D. Oswald, T. Kasper, and P. Pavlidès, “Lock it and still lose it—the (in) security of automotive remote keyless entry systems,” in Proc. 25th USENIX Security Symposium, Austin, TX, USA, Aug. 2016, pp. 929–944.

[10] M. Vanhoef and F. Piessens, “Key reinstallation attacks: Forcing nonce reuse in WPA2,” in Proc. ACM SIGSAC Conf. Computer Communications Security, Dallas, USA, Oct/Nov 2017, pp. 1313–1328.

[11] G. S. Vernam, “Secret signaling system,” U.S. Patent 1310719, Jul. 12, 1919.

[12] C. E. Shannon, “Communication theory of secrct systems,” Bell System Tech. J., vol. 28, no. 4, pp. 656–715, 1949.

[13] A. Wyner, “The wire-tap channel,” Bell System Tech. J., vol. 54, no. 8, pp. 1355–1387, 1975.

[14] S. Leung-Yan-Cheong and M. Hellman, “The gaussian wire-tap channel,” IEEE Trans. Inf. Theory, vol. 24, no. 4, pp. 451–456, 1978.

[15] A. Mukherjee, S. Fakoorian, J. Huang, and A. Swindlehurst, “Principles of physical layer security in multiuser wireless networks: A survey,” IEEE Commun. Surveys Tuts., vol. 16, no. 3, pp. 1550–1573, Third Quarter 2014.

[16] M. Bloch, J. Barros, M. R. Rodrigues, and S. W. McLaughlin, “Wireless information-theoretic security,” IEEE Trans. Inf. Theory, vol. 54, no. 6, pp. 2515–2534, 2008.

[17] Y. Liu, H.-H. Chen, and L. Wang, “Physical layer security for next generation wireless networks: Theories, technologies, and challenges,” IEEE Commun. Surveys Tuts., vol. 19, no. 1, pp. 347–376, 2016.

[18] H. V. Poor and R. F. Schaefer, “Wireless physical layer security,” Proc. National Academy of Sciences, vol. 114, no. 1, pp. 19–26, 2017.

[19] Y. Wu, A. Khisti, C. Xiao, G. Caire, K.-K. Wong, and X. Gao, “A survey of physical layer security techniques for 5G wireless networks and challenges ahead,” IEEE J. Sel. Areas Commun., vol. 36, no. 4, pp. 679–695, 2018.

[20] J. M. Hamamreh, H. M. Furqan, and H. Arslan, “Classifications and applications of physical layer security techniques for confidentiality: A comprehensive survey,” IEEE Commun. Surveys Tuts., vol. 21, no. 2, pp. 1773 – 1828, 2019.

[21] S. Goel and R. Negi, “Guaranteeing secrecy using artificial noise,” IEEE Trans. Wireless Commun., vol. 7, no. 6, pp. 2180–2189, 2008.

[22] X. Ding, T. Song, Y. Zou, X. Chen, and L. Hanzo, “Security-reliability tradeoff analysis of artificial noise aided two-way opportunistic relay selection,” IEEE Trans. Veh. Technol., vol. 66, no. 5, pp. 3930–3941, 2016.

[23] F. Zhou, Z. Chu, H. Sun, R. Q. Hu, and L. Hanzo, “Artificial noise aided secure cognitive beamforming for cooperative MISO-NOMA using SWIPT,” IEEE J. Sel. Areas Commun., vol. 36, no. 4, pp. 918–931, 2018.

[24] Y. Wén, M. Yoshida, J. Zhang, Z. Chu, P. Xiao, and R. Tafazolli, “Machine learning based attack against artificial noise-aided secure communication,” in Proc. IEEE ICC, Shanghai, China, May 2019, pp. 1–6.

[25] A. Mukherjee and A. L. Swindlehurst, “Robust beamforming for security in MIMO wiretap channels with imperfect CSI,” IEEE Trans. Signal Process., vol. 59, no. 1, pp. 351–361, 2010.

[26] Y. Zhang, Y. Ko, R. Woods, and A. Marshall, “Defining spatial secrecy outage probability for exposure region-based beamforming,” IEEE Trans. Wireless Commun., vol. 16, no. 2, pp. 900–912, 2016.

[27] Y. Zhang, R. Woods, Y. Ko, A. Marshall, and J. Zhang, “Security optimization of exposure region-based beamforming with a uniform circular array,” IEEE Trans. Commun., vol. 66, no. 6, pp. 2630–2641, 2017.

[28] Z. Kong, S. Yang, D. Wang, and L. Hanzo, “Robust beamforming and jamming for enhancing the physical layer security of full duplex radios,” IEEE Trans. Inf. Forensics Security, vol. 14, no. 12, pp. 3151–3159, 2019.

[29] B. He and X. Zhou, “Secure on-off transmission design with channel estimation errors,” IEEE Trans. Inf. Forensics Security, vol. 8, no. 12, pp. 1923–1936, 2013.

[30] W. Diffie and M. E. Hellman, “New directions in cryptography,” IEEE Trans. Inf. Theory, vol. 22, no. 6, pp. 644–654, 1976.

[31] W. Stallings, Cryptography and network security: Principles and practice, 5th ed. Pearson Upper Saddle River, 2017.

[32] Data Encryption Standard (DES), Federal Information Processing Standards Publication Std. FIPS PUB 46-3, 1999, accessed on 21 April, 2020. [Online]. Available: http://csrc.nist.gov/publications/fips/fips46-3.pdf

[33] Advanced Encryption Standard, Federal Information Processing Standards Publication Std. FIPS PUB 197, 2001, accessed on 21 April, 2020. [Online]. Available: http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf

[34] Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specification, IEEE Std. 802.11, 2012.
New IoT-malware grew three-fold in H1 2018. Accessed on 21 April 2020.

J. W. Wallace and R. K. Sharma, “Automatic secret keys from recipro-

C. Ye, A. Reznik, and Y. Shah, “Extracting secrecy from jointly

Y. Liu, S. C. Draper, and A. M. Sayeed, “Exploiting channel diversity in

Sigfox. Accessed on 21 April 2020. [Online]. Available: https://

K.-A. Shim, “A survey of public-key cryptographic primitives in

Y. Liu, S. C. Draper, and A. M. Sayeed, “Exploiting channel diversity in

U. Maurer and S. Wolf, “Secret-key agreement by public discussion from

M. Bloch and J. Barros, Physical-Logical Security: From Information

S. Tomasin and A. D’All’Arche, “Resource allocation for secret key

J. E. Hershey, A. A. Hassan, and R. Yarlagadda, “Unconventional

A. A. Hassan, W. E. Stark, J. E. Hershey, and S. Chennakeshu, “Crypt-

R. Ahlswede and I. Csiszar, “Common randomness in information

S. N. Premnath, P. L. Gowda, S. K. Kasera, N. Patwari, and S. V.

S. Jana, S. N. Premnath, M. Clark, S. K. Kasera, N. Patwari, and S. V.

J. Zhang, R. Woods, T. Q. Duong, and R. Woods, “On the key generation from correlated wireless channels,” IEEE Commun. Lett., vol. 21, no. 4, pp. 961–964, 2017.

N. Patwari, J. Croft, S. Jana, and S. K. Kasera, “High-rate uncorrelated bit extraction for shared secret key generation from channel measurements,” IEEE Trans. Mobile Comput., vol. 9, no. 1, pp. 17–30, Jan. 2010.

Y. Wei, K. Zeng, and P. Mohapatra, “Adaptive wireless channel probing for shared key generation based on PID controller,” IEEE Trans. Mobile Comput., vol. 12, no. 5, pp. 1842–1852, 2013.

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.

G. Li, A. Hu, J. Zhang, L. Peng, C. Sun, and D. Cao, “High-agreement uncorrelated secret key generation based on principal component analysis pre-processing,” IEEE Trans. Commun., vol. 66, no. 7, pp. 3022–3034, 2018.

T. Aono, K. Higuchi, T. Ohira, B. Komiyama, and H. Sasaoka, “Wireless secret key generation exploiting reactance-domain scalar response of multipath fading channels,” IEEE Trans. Antennas Propag., vol. 53, no. 11, pp. 3776–3784, Nov. 2005.

M. Wilhelm, I. Martinovic, and J. B. Schmitt, “Secure key generation in sensor networks based on frequency-selective channels,” IEEE J. Sel. Areas Commun., vol. 31, no. 9, pp. 1779–1790, 2013.

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

S. Mathur, W. Trappe, N. Mandayam, C. Ye, and A. Reznik, “Radio-telepathy: extracting a secret key from an unauthenticated wireless channel,” in Proc. 14th Annu. Int. Conf. Mobile Computing and Networking (MobiCom), San Francisco, California, USA, Sep. 2008, pp. 128–139.

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.

C. Ye, S. Mathur, A. Reznik, Y. Shah, W. Trappe, and N. B. Mandayam, “Information-theoretically secret key generation for fading wireless channels,” IEEE Trans. Inf. Forensics Security, vol. 5, no. 2, pp. 240–254, 2010.

K. Zeng, D. Wu, A. Chan, and P. Mohapatra, “Exploiting multiple-antenna diversity for shared secret key generation in wireless networks,” in Proc. 29th IEEE Int. Conf. Comput. Commun. (INFOCOM), Phoenix, Arizona, USA, Mar. 2010, pp. 1–9.

H. Liu, Y. Wang, J. Yang, and Y. Chen, “Fast and practical secret key extraction by exploiting channel response,” in Proc. 32nd IEEE Int. Conf. Comput. Commun. (INFOCOM), Turin, Italy, Apr. 2013, pp. 3048–3056.

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, 2016.

J. Zhang, B. He, T. Q. Duong, and R. Woods, “On the key generation from correlated wireless channels,” IEEE Commun. Lett., vol. 21, no. 4, pp. 961–964, 2017.

Y. Hu, S. Jha, and W. Hu, “LoRa-Key: Secure key generation system for LoRa-based network,” IEEE Internet Things J., vol. 6, no. 4, pp. 381–391, 2019.

H. Ruotsalainen, J. Zhang, and S. Grebennik, “Experimental investigation of wireless key generation for low-power wide area networks,” IEEE Internet Things J., vol. 7, no. 3, pp. 1745 – 1755, Mar. 2020.
[80] G. Li, Z. Zhang, J. Zhang, and A. Hu, “Encrypting wireless communications on the fly using one time pad and key generation,” IEEE Internet Things J., 2020, accepted.

[81] R. Ren, H. Su, and Q. Wang, “Secret key generation exploiting channel characteristics in wireless communications,” IEEE Wireless Commun., vol. 18, no. 4, pp. 6–12, 2011.

[82] S. P. Peng, “Physical layer key generation in wireless networks: Challenges and opportunities,” IEEE Commun. Mag., vol. 53, no. 6, pp. 33–39, 2015.

[83] T. Wang, Y. Liu, and A. V. Vasilakos, “Survey on channel reciprocity based key establishment techniques for wireless systems,” Wireless Networks, pp. 1–12, Jan. 2015.

[84] 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.

[85] J. Zhang, T. Duong, R. Woods, and A. Marshall, “Securing wireless communications of the internet of things from the physical layer, an overview,” Entropy, vol. 19, no. 8, p. 420, 2017.

[86] G. Li, C. Sun, J. Zhang, E. Jorswieck, B. Xiao, and A. Hu, “Physical layer key generation in 5G and beyond wireless communications: Challenges and opportunities,” Entropy, vol. 21, p. 497, 2019.

[87] J. Zhang, S. R. Kundran, Z. Sun, R. Woods, and L. Hanzo, “Physical layer security for the Internet of Things: Authentication and key generation,” IEEE Wireless Commun., pp. 92–98, Oct. 2019.

[88] C. Gomez, J. Oller, and J. Paradells, “Overview and evaluation of Bluetooth low energy: An emerging low-power wireless technology,” Sensors, vol. 12, no. 9, pp. 11 734–11 753, 2012.

[89] V. Niemelä, J. Haapola, M. Hämäläinen, and J. Iinatti, “An ultra wideband survey: Global regulations and impulse radio research based on standards,” IEEE Commun. Surveys Tuts., vol. 19, no. 2, pp. 874–890, 2017.

[90] Y. Li, H. Minn, T. Jacobs, and M. Z. Win, “Frequency offset estimation for MB-OFDM-based UWB systems,” IEEE Trans. Commun., vol. 56, no. 6, pp. 968–979, 2008.

[91] L.-L. Yang and L. Hanzo, “Residue number system assisted fast-frequency-chirped synchronous ultra-wideband spread-spectrum multiple-access: a design alternative to impulse radio,” IEEE J. Sel. Areas Commun., vol. 20, no. 9, pp. 1652–1663, 2002.

[92] U. Raza, P. Kulkarni, and M. Soorijabandara, “Low power wide area networks: An overview,” IEEE Commun. Surveys Tuts., vol. 19, no. 2, pp. 855–873, 2017.

[93] LoRaWAN 1.1 Regional Parameters. Accessed on 21 April, 2020. [Online]. Available: https://lorawan-alliance.org/resource-hub/lorawantm-regional-parameters-v11rb

[94] A. H. Sodhro, J. J. Rodrigues, S. Piribhulal, N. Zahid, A. R. L. de Macedo, and V. H. C. de Albuquerque, “Mobility enabled security for optimizing IoT based automation system using a wireless sensor network,” Sensors, vol. 17, no. 1, p. 69, 2017.

[95] M. Almulhim and N. Zaman, “A lightweight and secure authentication scheme for IoT based e-health applications,” in Proc. IEEE Int. Conf. Commun. (ICC), London, UK, Jun. 2015, pp. 4949–4953.

[96] G. Li, A. Hu, Y. Zou, L. Peng, and M. Valkama, “A novel transform for secret key generation in time-varying TDD channel under hardware fingerprint deviation,” in Proc. IEEE 82nd Veh. Technol. Conf. (VTC 2015-Fall), Boston, MA, USA, Sep. 2015, pp. 1–5.

[97] S. Yasakiwara, H. Iwai, and H. Sasaoka, “Adaptive key generation in secret key agreement scheme based on the channel characteristics in OFDM,” in Proc. Int. Symp. Information Theory and its Applications, Auckland, New Zealand, Dec. 2008, pp. 1–6.

[98] M. Almulhim, G. Oikonomou, R. Piechocki, T. Tryfonas, and P. Thomas, “Physical layer secret-key generation with discrete cosine transform for the Internet of Things,” in Proc. IEEE Int. Conf. Commun. (ICC), Paris, France, May 2017.

[99] Y. Wu, Y. Sun, L. Zhan, and Y. Ji, “Low mismatch key agreement based on wavelet-transform trend and fuzzy vault in body area network,” Int. J. Distributed Sensor Networks, vol. 2013, pp. 1–16, 2013.

[100] F. Zhan and N. Yao, “On the using of discrete wavelet transform for physical layer key generation,” Ad Hoc Networks, vol. 64, pp. 22–32, Sep. 2017.

[101] S. Gopinath, R. Guillaume, P. Duplys, and A. Czyliw, “Reciprocity enhancement and decorrelation schemes for phy-based key generation,” in Proc. IEEE Globecom Workshop Trusted Commun. with Physical Layer Security (TCPLS), Austin, Texas, USA, Dec. 2014, pp. 1367–1372.

[102] 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.

[103] R. Guillaume, A. Mueller, C. T. Zenger, C. Paar, and A. Czyliw, “Fair comparison and evaluation of quantization schemes for PHY-based key generation,” in Proc. 18th Int. OFDM Workshop (InfOw14), Essen, Germany, Aug. 2014, pp. 1–5.

[104] C. T. Zenger, J. Zimmer, and C. Paar, “Security analysis of quantization schemes for channel-based key extraction,” in Workshop Wireless Commun. Security at the Physical Layer, Coimbra, Portugal, Jul. 2015, pp. 1–6.

[105] Q. Han, J. Liu, Z. Shen, J. Liu, and F. Gong, “Vector partitioning quantization utilizing k-means clustering for physical layer secret key generation,” Information Sciences, vol. 512, pp. 137–160, 2020.

[106] C. Huth, R. Guillaume, T. Strohm, P. Duplys, I. A. Samuel, and T. Güneysu, “Information reconciliation schemes in physical-layer security: A survey,” Computer Networks, vol. 109, pp. 84–104, 2016.

[107] N. Vázquezinidehaj, Z. Wang, S. N. Ng, and L. Hanzo, “Satellite-based continuous-variable quantum communications: State-of-the-art and a predictive outlook,” IEEE Commun. Surveys Tuts., vol. 21, no. 1, pp. 881–919, 2018.
X. Zhu, F. Xu, E. Novak, C. C. Tan, Q. Li, and G. Chen, “Using O. Gungor, F. Chen, and C. Koksal, “Secret key generation via Y. Xie, Z. Li, and M. Li, “Precise power delay profiling with com-
D. Halperin, W. Hu, A. Sheth, and D. Wetherall, “Tool release: gather-
Wireless Communications: Principles and Practice π H. MacLeod, C. Loadman, and Z. Chen, “Experimental studies of the 2.4-GHz ISM wireless indoor channel,” in Proc. IEEE 3rd Annual

Communication Networks and Services Research Conference, 2005, pp. 63–68.

L. Yao, S. T. Ali, V. Sivaraman, and D. Ostry, “Decorrelating secret bit extraction via channel hopping in body area networks,” in Proc. 23rd IEEE Int. Symp. Personal Indoor Mobile Radio Commun. (PIMRC), Sydney, Australia, Sep. 2012, pp. 1454–1459.

N. Premnath, J. Croft, N. Patwari, and K. K. R. Kasera, “Efficient high-rate secret key extraction in wireless sensor networks using collaboration,” ACM Trans. Sensor Netw., vol. 11, no. 1, p. 2, 2014.

B. T. Quist and M. A. Jensen, “Maximizing the secret key rate for informed radios under different channel conditions,” IEEE Trans. Wireless Commun., vol. 12, no. 10, pp. 5146–5153, 2013.

— Bound on the key establishment rate for multi-antenna reciprocal electromagnetic channels,” IEEE Trans. Antennas Propag., vol. 62, no. 3, pp. 1378–1385, 2014.

Optimal channel estimation in beamformed systems for common-randomness-based secret key establishment,” IEEE Trans. Inf. Forensics Security, vol. 8, no. 7, pp. 1211–1220, 2013.

— Maximization of the channel-based key establishment rate in MIMO systems,” IEEE Trans. Wireless Commun., vol. 14, no. 10, pp. 5577–5583, 2015.

M. G. Madiseh, S. W. Neville, and M. L. McGuire, “Applying beamforming to address temporal correlation in wireless channel characterization-based secret key generation,” IEEE Trans. Inf. Forensics Security, vol. 7, no. 4, pp. 1278–1287, 2012.

S. Bakši and D. C. Popescu, “Secret key generation with precoding and role reversion in MIMO wireless systems,” IEEE Trans. Wireless Commun., vol. 18, no. 6, pp. 3104–3112, 2019.

X. Zhang, and E. W. Knightly, “Csisnoop: Inferring channel state information in multi-user MIMO WLANs,” IEEE/ACM Trans. Netw., vol. 27, no. 1, pp. 231–244, 2019.

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, July 2016, pp. 1–5.

A. J. Pierrot, R. A. Chou, and M. R. Bloch, “Experimental aspects of secret key generation in indoor wireless environments,” in Proc. 14th IEEE Int. Workshop Signal Process. Advances in Wireless Commun. (SPAWC), Darmstadt, Germany, 2013, pp. 669 – 673.

C. L. K. Ngassa, R. Molière, F. Delaveau, A. Sibille, and N. Shapiro, “Secret key generation scheme from WiFi and LTE reference signals,” Analog Integrated Circuits and Signal Processing, vol. 91, no. 2, pp. 359–372, 2019.

D. Kreiser, Z. Dyka, S. Kornemann, C. Witke, I. Kabin, O. Stecklina, and P. Langendoerfer, “On wireless channel parameters for key generation in industrial environments,” IEEE Access, vol. 5, pp. 79010–79025, 2017.

A. M. Allam, “Channel-based secret key establishment for FDD wire-
less communication systems,” Communications on Applied Electronics, vol. 7, no. 9, pp. 2399–2412, 2017.

X. Wu, Y. Peng, C. Hu, H. Zhao, and L. Shu, “A secret key generation method based on CSI in OFDM-FDD system,” in Proc. IEEE GLOBECOM Workshop Trusted Commun. with Physical Layer Security (TCPLS), Atlanta, GA, USA, Dec. 2013, pp. 1297–1302.

S. J. Goldberg, Y. C. Shah, and A. Reznik, “Method and apparatus for performing JRNSO in FDD, TDD and MIMO communications,” U.S. Patent 8,601,196, Mar. 19, 2013, accessed on 21 April, 2020. [Online]. Available: https://www.google.com/patents/US8601196

L. Peng, G. Li, J. Zhang, R. Woods, M. Liu, and A. Hu, “An investigation of using loop-back mechanism for channel reciprocity enhancement in secret key generation,” IEEE Trans. Mobile Comput., vol. 18, no. 3, pp. 507–519, 2019.

B. Liu, A. Hu, and G. Li, “Secret key generation scheme based on the channel covariance matrix eigenvalues in FDD systems,” IEEE Commun. Lett., vol. 23, no. 9, pp. 1493–1496, 2019.

W. J. Wang, H. Y. Jiang, X. G. Xia, P. C. Mu, and Q. Y. Yin, “A wireless secret key generation method based on chinese remainder theorem in FDD systems,” Science China (Information Sciences), vol. 55, no. 7, pp. 1605–1616, 2012.

G. Li, A. Hu, C. Sun, and J. Zhang, “Constructing reciprocal channel coefficients for secret key generation in fdd systems,” IEEE Commun. Lett., vol. 22, no. 12, pp. 2479–2483, Dec. 2018.

Evolved universal terrestrial radio access (E-UTRA): User equipment (UE) radio transmission and reception, 3GPP Std. Technical Specifi-
cation 36.101 Release 10.
Guyue Li received the B.S. degree in Information Science and Technology and the Ph.D. degree in Information Security from Southeast University, Nanjing, China, in 2011 and 2017, respectively. She is currently a Lecturer at Southeast University. Her research interests include physical layer security, secret key generation, radio frequency fingerprint and link signature.

Alan Marshall (M’88-SM’00) has spent over 25 years working in the telecommunications and defense Industries, and in Academia. He currently holds the chair in Communications Networks at the University of Liverpool where he is also Head of Department of Electrical Engineering and Electronics, and director of the Advanced Networks Group. He has published over 250 scientific papers and holds a number of joint patents in the areas of communications and network security. He formed a successful spin-out company Traffic Observation & Management (TOM) Ltd specializing in intrusion detection & prevention for wireless networks. His research interests include Network architectures and protocols; Mobile and Wireless networks; Network Security; machine-to-machine communications and Trust systems; high-speed packet switching, Quality of Service & Experience (QoS/QoE); distributed haptics and multisensory communications. He is Section Editor (section B: Computer and Communications Networks and Systems) for the Computer Journal of the British Computer Society, and sits on the program committees of a number of IEEE conferences. He is the UK lead for the UK-Jiangsu 20+20 world class universities initiative.

Aiqun Hu received the B.Sc.(Eng.), the M.Eng.Sc. and Ph.D. degrees from Southeast University in 1987, 1990, and 1993 respectively. He was invited as a post-doc research fellow in The University of Hong Kong from 1997 to 1998, and TCF fellow in Nanyang Technological University in 2006. His research interests include data transmission and secure communication technology. He has published two books and over 100 technical papers in wireless communications field.

Lajos Hanzo (http://www-mobile.ecs.soton.ac.uk, https://en.wikipedia.org/wiki/Lajos_Hanzo) FREng, FIEEE, FIET, Fellow of EURASIP, DSc has received his Master degree and Doctorate in 1976 and 1983, respectively from the Technical University (TU) of Budapest. He was also awarded Honorary Doctorates by the TU of Budapest (2009) and by the University of Edinburgh (2015). He is a Foreign Member of the Hungarian Academy of Sciences and a former Editor-in-Chief of the IEEE Press. He has served as Governor of both IEEE ComSoc and of VTS. He has published 1900+ contributions at IEEE Xplore, 19 Wiley-IEEE Press books and has helped the fast-track career of 119 PhD students. Over 40 of them are Professors at various stages of their careers in academia and many of them are leading scientists in the wireless industry.