Contents lists available at ScienceDirect

Future Generation Computer Systems

journal homepage: www.elsevier.com/locate/fgcs

An over-the-air key establishment protocol using keyless cryptography

Yuexin Zhang [a,b], Yang Xiang [a,b], Tao Wang [c], Wei Wu [d,*], Jian Shen [e]

[a] Centre for Cyber Security Research, Deakin University, Geelong, VIC 3220, Australia
[b] The State Key Laboratory of Integrated Services Networks, Xidian University, China
[c] University of South Florida, Tampa, FL 33620, USA
[d] Fujian Provincial Key Laboratory of Network Security and Cryptology, School of Mathematics and Computer Science, Fujian Normal University, Fuzhou, China
[e] School of Computer and Software, Nanjing University of Information Science and Technology, Nanjing, China

HIGHLIGHTS

- We present a key establishment protocol using keyless cryptography.
- The protocol is design for two nearby wireless devices.
- Two devices can establish a secret key by sending random signals to each other.
- The analysis shows that our protocol is a low cost key establishment protocol.

ARTICLE INFO

Article history:
Received 27 August 2016
Received in revised form
22 November 2016
Accepted 9 December 2016
Available online xxxx

Keywords:
Key establishment
Security
Anonymous channel
Source indistinguishability

ABSTRACT

Today, an increasing number of devices wirelessly communicate with each other. However, due to the nature of wireless transmission, the communications are vulnerable to many adversarial attacks such as eavesdropping. Key establishment is one of the fundamental and widely studied countermeasures for securing the communications. In certain applications, the wireless devices may be energy-constrained, such as sensor nodes. Thus, energy intensive asymmetric key establishment protocols are infeasible. Additionally, in some scenarios, it is not practical to assume that all the devices pre-share certain secrets. Motivated by these observations, this paper presents an over-the-air key establishment protocol using keyless cryptography. Specifically, the proposed protocol is designed without using asymmetric key cryptography and pre-shared secrets. More specifically, our protocol provides a concrete construction to transform the wireless channel into an anonymous channel, and two wireless devices can establish a secret key by directly sending random signals to each other. The performance analysis shows that the energy consumption of our protocol is around 176 times cheaper than that of the Diffie–Hellman key exchange protocol. Additionally, it takes only 159.04 *ms* to establish a key with 112 secret bits.

1. Introduction

In recent years, an increasing number of devices are equipped with wireless interfaces and microprocessors. Using these devices, we can access the Internet and keep connected with others. In certain applications, a device needs to directly communicate with other nearby devices. Thus, many protocols are designed for nearby devices' communications, such as the Device-to-Device (D2D) communication, Near Field Communication (NFC), and the IEEE standard 802.15.4. Specifically, in these protocols, messages are directly transmitted between two nearby devices (without employing forwarders and routers). Due to the nature of wireless transmission, the communications are vulnerable to many adversarial attacks. For instance, the adversary can eavesdrop the communications and conduct malicious attacks. Recently, a new class of attack, the Advanced Persistent Threat (APT), has emerged. Specifically, the APT is defined by the US National Institute of Standards and Technology (NIST) as: "An adversary that possesses

[*] Corresponding author.
E-mail addresses: yuexinz@deakin.edu.au (Y. Zhang), yang.xiang@deakin.edu.au (Y. Xiang), taow@mail.usf.edu (T. Wang), weiwu@fjnu.edu.cn (W. Wu), s_shenjian@126.com (J. Shen).

http://dx.doi.org/10.1016/j.future.2016.12.013
0167-739X/© 2016 Published by Elsevier B.V.

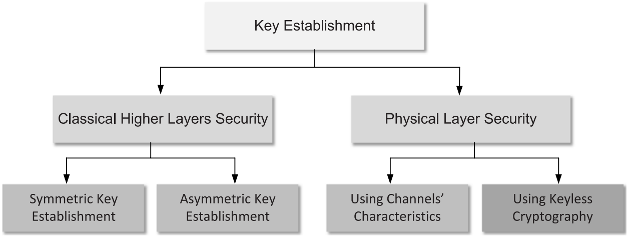

Key Establishment
Classical Higher Layers Security
Physical Layer Security
Symmetric Key Establishment
Asymmetric Key Establishment
Using Channels' Characteristics
Using Keyless Cryptography

Fig. 1. Overview of existing key establishment.

sophisticated levels of expertise and significant resources which allow it to create opportunities to achieve its objectives by using multiple attack vectors (e.g., cyber, physical, and deception)" [1,2].

To ensure the security and privacy of communications, cryptographic keys need to be established. Until now, key establishment protocols have been extensively and intensively studied, and classical designs can be classified into two main types, namely, asymmetric key establishment protocols and symmetric key establishment protocols (as shown in Fig. 1). Specifically, in asymmetric key establishment protocols (e.g., the Diffie–Hellman key exchange protocol [3]), costly computation operations, such as the exponentiation operations, need to be executed. In symmetric key establishment protocols (e.g., the key pre-distribution protocols [4,5]), however, considerable memory spaces are used to pre-load secrets.

In certain applications, the wireless devices may be energy-constrained devices (i.e., powered by batteries), such as sensor nodes. Thus, the energy intensive asymmetric key establishment protocols are excluded. Besides, the wireless devices (such as sensor nodes, smart phones, tablets, and laptops) are produced by different factories, and they are integrated with different technologies. Thus, it is not a practical assumption that all these devices are pre-loaded with certain secrets when they leave factories. Motivated by these observations, in this paper, we aim to design a key establishment protocol without using asymmetric key cryptography and pre-shared secrets.

It is a challenging topic to establish secret keys without using energy intensive asymmetric key cryptography and pre-shared secrets, and the topic has been undertaken in two ways, namely, (a) extracting keys by taking advantage of the wireless channels' characteristics, such as the received signal strength (RSS) and channel impulse response (CIR) [6,7]; and (b) establishing keys using keyless cryptography [8].

For those key establishment protocols using characteristics of the wireless channels, some issues still remain unsatisfactory. For instance, asymmetric effects introduced by the multipath fading, the key generation rate needs to be improved, and a dynamic environment is needed to provide sufficient entropy. For those key establishment protocols using keyless cryptography, two devices can establish a secret key with light consumptions. Specifically, an anonymous channel is needed to guarantee that the adversaries cannot identify the source of the eavesdropped messages. Namely, the anonymous channel achieves source indistinguishability (please refer to Section 3.2 for details). However, these key establishment protocols are designed based on human assistance (e.g., shaking the devices), or they are designed without giving a concrete construction to transform the wireless channel into an anonymous channel.

Our contribution. In this paper, we present an over-the-air key establishment protocol using keyless cryptography. Specifically, our key establishment protocol possesses the following properties:

1. Our key establishment protocol is specifically designed for assisting users, who do not pre-share any secrets and have no access to the on-line trusted third party, to establish secret keys. Specifically, in order to establish a secret key, two users in our protocol show off their wireless devices, and directly send analog signals to each other.
2. The protocol provides a concrete construction to transform the wireless channel into an anonymous channel. Specifically, to achieve source indistinguishability, users move into proximity and introduce randomness to the signals (in order to achieve spatial indistinguishability). Besides, in each round, signals are sent at randomly chosen times (in order to achieve temporal indistinguishability).
3. Our protocol is a low-cost key establishment protocol. The performance analysis shows that energy consumption of our protocol is about 176 times cheaper than that of the Diffie–Hellman key exchange protocol [3], and it only takes around 159.04 ms to establish a key with 112 secret bits.

Organization of the paper. The remainder of this paper is organized as follows. In the next section, we present a brief overview on the related work. Section 3 reviews the preliminaries required in this paper. Then, the proposed protocol is described in Section 4, and its security and performance analysis are provided in Section 5 and Section 6, respectively. In Section 7, we conclude this paper.

2. Related work

In this section, we review those closely related key establishment protocols. Namely, key establishment protocols using keyless cryptography, and key establishment protocols for full-duplex NFC.

2.1. *Key establishment protocols using keyless cryptography*

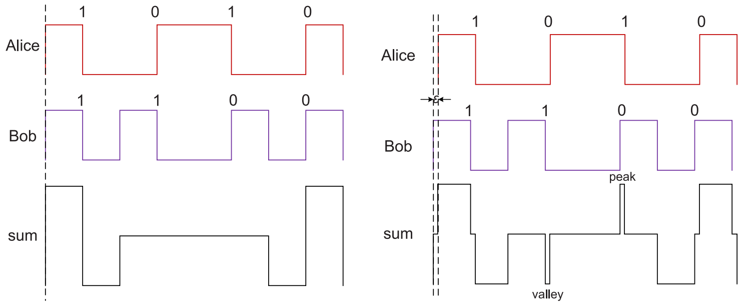
The key establishment protocol using keyless cryptography is introduced for the first time in [8], and it is optimized by [9–12]. Specifically, these protocols are designed based on the anonymous channels. A broadcast channel is said to be an anonymous channel if it achieves source indistinguishability. Namely, the adversary can eavesdrop the transmitted messages over the channel, but she cannot identify the source of the messages (please refer to Section 3.2 for details).

For instance, users in [10] can establish a key with k secret bits by executing following operations:

- Alice randomly chooses $\frac{k}{2}$ bits $C_a = [C_a^1, C_a^2, \ldots, C_a^{\frac{k}{2}}]$. Similarly, Bob randomly chooses $\frac{k}{2}$ bits $C_b = [C_b^1, C_b^2, \ldots, C_b^{\frac{k}{2}}]$;
- Alice builds $\frac{k}{2}$ messages $m_A^1, m_A^2, \ldots, m_A^{\frac{k}{2}}$ using C_a. For instance, the message m_A^i is built by following the rule that, the source identifier of m_A^i is set to be *Alice* if $C_a^i = 1$. Otherwise, it is set to be *Bob*. Following the same rule, Bob builds $\frac{k}{2}$ messages $m_B^1, m_B^2, \ldots, m_B^{\frac{k}{2}}$ using C_b; and
- In the ith round, either Alice or Bob (with equal probability) sends an empty packet m_A^i or m_B^i at time t_i, where t_i is chosen uniformly at random in the interval $[(i-1)T_r, iT_r]$ (T_r is a constant parameter).

The secret bits are represented by identifying the correct or incorrect identifiers of the messages. For example, in the ith round, Alice and Bob set the ith bit of secret key K to be 1, if the sender and recipient address of m^i is correct. Otherwise, it is set to be 0. The security of [10] relies on the source indistinguishability, and the source indistinguishability requires that the exchanged messages are temporal indistinguishability and spatial indistinguishability.

Please cite this article in press as: Y. Zhang, et al., An over-the-air key establishment protocol using keyless cryptography, Future Generation Computer Systems (2016), http://dx.doi.org/10.1016/j.future.2016.12.013



(a) Perfect synchronization.

(b) Out of synchronization.

Fig. 2. Key agreement protocols presented in [13,14].

To achieve temporal indistinguishability, the order of the transmitted packets are randomized in [10]. As a result, the adversary cannot predict who is going to transmit next. To achieve spatial indistinguishability, users in [10] shake the devices during the key establishment phase. All these countermeasures ensure that the adversary cannot identify the source of the eavesdropped messages.

In order to minimize human intervention, a crypto-less key establishment protocol is presented in [11]. The difference between the protocol in [11] and the protocol in [10] is that, users in [11] have no need to shake the devices. To achieve spatial indistinguishability, in [11], the transmission powers are randomly chosen by each user' device. Besides, the authors extend the idea of [11] and present a peer-to-peer key establishment protocol in [12].

Based on the idea of [10–12], in this paper, we provide a concrete construction to transform the wireless channel into an anonymous channel. Specifically, in order to achieve spatial indistinguishability, users add user-introduced randomness to the amplitudes, phase, and angular frequencies of the transmitted signals, simultaneously. Thus, our protocol is an optimized version of [10–12].

2.2. Key establishment protocols for full-duplex NFC

Low-cost key establishment protocols are presented in [13,14] by taking advantage of the full-duplex capability of NFC devices.[1] In these protocols, two users (say Alice and Bob) can establish a secret key by sending random bits simultaneously. To facilitate understanding, Fig. 2(a) abstracts the basic ideas of [13,14]. In Fig. 2(a), we assume that the random bits C_a (chosen by Alice) are $C_a = [1, 0, 1, 0]$, and the random bits C_b (chosen by Bob) are $C_b = [1, 1, 0, 0]$. When two devices are in proximity (e.g., $d < 10$ cm) and send the random bits C_a and C_b simultaneously, the adversary can only eavesdrop the superposed signals of C_a and C_b. Specifically, the adversary cannot identify the source of "1" and the source of "0" when she eavesdrops a flat voltage (as shown in Fig. 2(a)). However, both Alice and Bob know the bit sent by themselves. Thus, they can establish a secret key by taking advantage of this "knowledge".

As pointed out in [17], these protocols are impractical due to the requirement of perfect synchronization. In Fig. 2(b), we show an example when two devices are out of synchronization. Specifically, we assume that the signals sent by Bob are ε ahead of the signals sent by Alice. From Fig. 2(b) we can see that a small out of synchronization introduces valleys and peaks. Observing the valleys and peaks, the adversary can identify the bit sent by Alice and Bob. Thus, in this scenarios, the adversary can break the security of the protocols. Additionally, Jin et al. in [17] investigate that when the digital baseband signals are converted to analog signals, there are slight mismatches in the transmitted signals (due to the impairment of the devices [18]). Taking advantage of these mismatches, the adversary can violate the protocols.

To enhance the security of protocols [13,14], Jin et al. in [17] present an optimized key establishment protocol. Specifically, in [17], user-introduced randomness are employed to mask the slight synchronization offset and the mismatches. More specifically, in order to achieve the source indistinguishability, Jin et al. introduce (a) random time shifting against out of synchronization; (b) random amplitude scaling against amplitude mismatch; and (c) random phase shifting against phase mismatch. The security strength of [17] is expected to be maintained [19], however, protocol [17] is designed for the scenarios when devices work at full-duplex mode. In other words, it cannot be directly applied to the scenarios when devices work at half-duplex mode.[2] In this paper, we aim to present a protocol without this limitation. Namely, our protocol can be applied to the half-duplex mode and the full-duplex mode.

3. Preliminaries

Before presenting our protocol, in this section, we introduce the preliminaries required in this paper.

3.1. System model

This subsection reviews the system model of our protocol, and it follows the system model of [10–12,17]. Specifically, we consider the scenarios that two legitimate users, Alice and Bob, need to establish a secret key, but they do not have any pre-shared secret. To establish a secret key, wireless devices, such as PDAs, smart phones, tablets, and laptops, are used to obtain secret bits. More specifically, in order to establish a secret key, Alice and Bob move their devices into proximity and directly send analog signals to each other. Namely, the analog signals are transmitted

[1] A duplex communication system is a point-to-point system, and it composes of two devices that can communicate with each other in both directions. Specifically, in the full-duplex mode, two devices can transmit and receive signals simultaneously. In the half-duplex mode, however, the transmission and reception of signals must happen alternately. Namely, while one device is transmitting, the other one must only receive. Many existing wireless communication systems are designed based on the half-duplex mode (for example, the IEEE 802.11 families). The problem to achieve full-duplex capability is self-interference, and it is estimated that self-interference is billions of times stronger than the received signals. In [15,16], researchers from Stanford University built full-duplex radio prototypes.

[2] Recall that in the half-duplex mode, the transmission and reception of signals must happen alternately. In this scenario, the signals sent by Alice and Bob cannot superimpose with each other. Thus, the adversary can easily identify the source of the signals when protocol [17] is applied to the half-duplex mode.

Please cite this article in press as: Y. Zhang, et al., An over-the-air key establishment protocol using keyless cryptography, Future Generation Computer Systems (2016), http://dx.doi.org/10.1016/j.future.2016.12.013

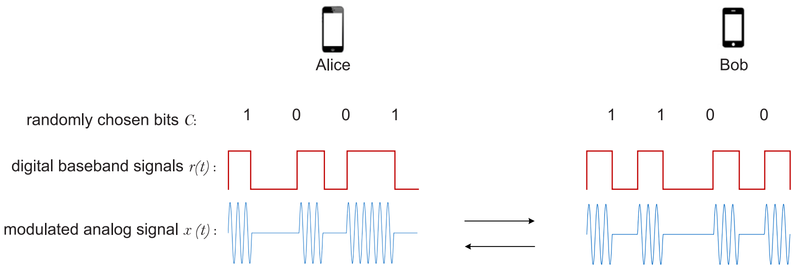

Alice
Bob
randomly chosen bits C:
1 0 0 1
1 1 0 0
digital baseband signals r(t) :
modulated analog signal x (t) :

Fig. 3. Alice and Bob try to establish a secret key over the wireless channel.

without employing forwarders and routers (e.g., by running the IEEE standard 802.15.4 protocol).

Adversarial model. We consider the adversary who aims to compromise the key established between two legitimate users. Specifically, we assume that the adversary possesses a high quality wireless channel to eavesdrop the communications between the users. Additionally, we assume that the adversary can capture the transmitted signals with high sensitivity and sampling rate. As a result, she can sense slight mismatches of signals' amplitudes, phases, and frequencies. Besides, the adversary can store all the transmitted signals and conduct sophisticated signal processing or data analysis. However, the adversary cannot tamper and delete the transmitted signals. It is due to the reason that two users are in proximity, and they send wireless signals directly to each other (without employing any forwarders and routers). Thus, in this scenarios, the adversary cannot tamper and delete the transmitted signals.

3.2. *Transforming the wireless channel into an anonymous channel*

This subsection reviews the characteristics of anonymous channels, and provides a concrete construction to transform the wireless channel into an anonymous channel.

As reviewed in Section 2 that the anonymous channels can be employed to establish secret keys. Specifically, these key establishment protocols do not rely on complex cryptographic computations and pre-shared secrets, users in these protocols can establish secret keys by exchanging a few "plain texts" over an anonymous channel. A broadcast channel is an anonymous channel if it hides the source of the messages. Namely, the adversary can eavesdrop the transmitted messages over the channel, but she cannot identify the source of the messages. Thus, anonymous channels need to achieve *source indistinguishability*. As defined in [10], a wireless channel between two users Alice and Bob achieves source indistinguishability, if the difference between the probability that the signals were sent by Alice and the probability that the signals were sent by Bob is negligible. Namely,

$$\mathrm{P}[Source(C) = \text{Alice}] - \mathrm{P}[Source(C) = \text{Bob}] < \varepsilon,$$

where ε is a negligible probability.

In order to illustrate the method to transform the wireless channel into an anonymous channel, we show an example in Fig. 3. In this example, we assume that the randomly chosen bits at Alice side are $C_a = [1, 0, 0, 1]$, and the randomly chosen bits at Bob side are $C_b = [1, 1, 0, 0]$. The random bits can be represented using the digital baseband signals $r(t)$. Specifically, as shown in Fig. 3, the bit "1" is represented using a failing edge, and the bit "0" is represented using a rising edge. In practice, to transmit the bits, devices need to convert the digital baseband signals $r(t)$ to the analog signals $x(t)$ using the equipped digital-to-analog converters. In Fig. 3, the amplitude shift keying (ASK) modulation is employed. Specifically, the analog signals $x_a(t)$ and $x_b(t)$ can be written as:

$$x_a(t) = h(t) \cdot r_a(t),$$
$$x_b(t) = h(t) \cdot r_b(t).$$

Here, the $h(t)$ is the radio frequency (RF) waveform at carrier frequency, $x_a(t)$ ($x_a(t) = X_A^1(t) + X_A^2(t) + X_A^3(t) + X_A^4(t)$) is the analog signals at Alice side, and $x_b(t)$ ($x_b(t) = X_B^1(t) + X_B^2(t) + X_B^3(t) + X_B^4(t)$) is the analog signals at Bob side. In the ith round ($i = 1, 2, 3, 4$), Alice and Bob send the ith signals $X^i(t)$ in a random order. Then, Alice and Bob can obtain the ith secret bit by following the predefined rule. For instance, the second bit of the key K is set to be 1 (i.e., $K = *1**$) if the first signals in the second round were sent by Alice. Otherwise, it is set to be 0 (i.e., $K = *0**$).

The above wireless channel is an anonymous channel, if it achieves source indistinguishability (i.e., the adversary cannot identify the source of the transmitted signals). In order to achieve source indistinguishability, the signals (transmitted over the channel) should be *temporally* and *spatially* indistinguishable [10].

3.2.1. *Temporal indistinguishability*

As pointed out in [10], temporal indistinguishability should ensure that the adversary cannot use *timing analysis* technology to identify the source of the transmitted signals. Namely, in each round, the adversary can correctly identify the source of the signals with probability at most $\frac{1}{2} + \varepsilon_1$ (using timing analysis), where ε_1 is a negligible probability.

In order to achieve temporal indistinguishability, each user in above example sends analog signals at time t_i, where t_i is chosen uniformly at random in the interval $[(i-1)T_r, iT_r]$. For instance, in the second round, Alice needs to send $C_a^2 = 0$ and Bob needs to send $C_b^2 = 1$. Thus, in this round, Alice and Bob send the corresponding analog signals $X_A^2(t)$ and $X_B^2(t)$ at time t_2^A and t_2^B, where t_2^A and t_2^B are chosen uniformly at random in the interval $[T_r, 2T_r]$. Besides, certain collision avoidance mechanism should be employed when devices work at half-duplex mode. The random medium access control (MAC) protocols, such as carrier sense multiple access with collision avoidance (CSMA/CA) MAC employed in the IEEE 802.15.4, are available to achieve temporal indistinguishability. In these protocols, a user needs to listen to the shared channel before transmitting signals. If the channel is busy, the user waits for a random time and listens again. If the channel is identified as idle, the user transmits its packets. It is easy to see that a CSMA/CA based system can achieve temporal indistinguishability.

As a contrast, the time division multiple access (TDMA) based MAC protocols cannot be employed, it is due to the reason that TDMA based systems cannot provide temporal indistinguishability. More specifically, a user in a TDMA based system is given one or several time slots, and the packets of that user are transmitted only during its slots. Thus, the adversary can easily identify the source of the transmitted packets.

3.2.2. Spatial indistinguishability

Spatial indistinguishability should ensure that the adversary cannot use *spatial analysis* technology to identify the source of the transmitted signals. Namely, in each round, the adversary can identify the source of the signals with probability at most $\frac{1}{2} + \varepsilon_2$ (using spatial analysis), where ε_2 is a negligible probability. To achieve spatial indistinguishability, frequency division multiple access (FDMA) based systems and code division multiple access (CDMA) based systems cannot be employed (the reason is similar to that of the TDMA based systems). Besides, *the spatial decorrelation property of the wireless channel* and *the characteristics of the signals transmitted by different devices* should be considered.

According to *the spatial decorrelation property of the wireless channel*, the signals sent by Alice and Bob undergo different fading channels when two users are far away from each other. More specifically, as long as the distance d between Alice and Bob is larger than $\frac{\lambda}{2}$ (where λ is the wavelength of the transmitted signals), the adversary can identify the source of the transmitted signals, e.g., by making use of the temporal link signatures [20,21]. Thus, in order to achieve spatial indistinguishability, two users Alice and Bob need to move into proximity (i.e., $d < \frac{\lambda}{2}$).

Besides, *the characteristics of the signals transmitted by different devices* can also be used to identify the source of the signals. It is proved that due to hardware and manufacturing inconsistencies, minute and unique variations are caused in signals transmitted by electronic devices [22,23]. Specifically, the amplitude, phase, and frequency features of the transmitted signals can be used to identify the wireless devices [24]. Taking Fig. 3 as an example, in the first round ($0 \le t \le T$), both Alice and Bob need to send bit 1. The transmitted analog signals are $X_A^1(t)$ and $X_B^1(t)$, and they can be written as:

$$\begin{cases} X_A^1(t) = A_a \cos(\omega_a t + \varphi_a) \cdot r^1(t), \\ X_B^1(t) = A_b \cos(\omega_b t + \varphi_b) \cdot r^1(t), \end{cases} \quad (0 \le t \le T) \tag{1}$$

where A_a and A_b are the amplitudes, φ_a and φ_b are the phases, ω_a and ω_b are the angular frequency of the transmitted signals.

Theoretically, the transmitted analog signals are the same (recall that both Alice and Bob send "1" in the first round), i.e., $A_a = A_b$, $\varphi_a = \varphi_b$, and $\omega_a = \omega_b$. In practice, however, there are slight mismatch in amplitudes A_a and A_b, such that $A_a \neq A_b$ (due to hardware and manufacturing inconsistencies). Similarly, the phases $\varphi_a \neq \varphi_b$, and the angular frequencies $\omega_a \neq \omega_b$. Making use of these mismatch, the adversary can identify the wireless devices (i.e., identify the sources of the signals). For example, we assume that the amplitudes of the transmitted signals $A_a = (1 + \alpha)A_b$, where α is a slight mismatch. Thus, the adversary can identify the sources of the transmitted signals by comparing the amplitudes of them (we assume that Alice and Bob are in proximity, i.e., $d < \frac{\lambda}{2}$).

Since the mismatches of amplitudes, phases, and angular frequencies are unavoidable, users can introduce randomness to mask these mismatches [17]. To introduce randomness to the amplitudes of the transmitted signals, in each round, a user randomly chooses coefficients Λ from $[\Lambda_{\min}, \Lambda_{\max}]$, and adds it to the amplitudes. For example, in the first round of Fig. 3, Alice and Bob randomly choose coefficients $\Lambda_a \in [\Lambda^1_{\min}, \Lambda^1_{\max}]$ and $\Lambda_b \in [\Lambda^2_{\min}, \Lambda^2_{\max}]$, and modulate the amplitudes of the signals as:

$$\begin{cases} X_A^1(t) = (\Lambda_a + A_a) \cos(\omega_a t + \varphi_a) \cdot r^1(t), \\ X_B^1(t) = (\Lambda_b + A_b) \cos(\omega_b t + \varphi_b) \cdot r^1(t). \end{cases} \tag{2}$$

Similarly, users can introduce randomness to the phases and angular frequencies of the transmitted signals. Introducing randomness to amplitudes, phases, and angular frequencies simultaneously, the modulated signals can be written as (in the first round of Fig. 3):

$$\begin{cases} X_A^1(t) = (\Lambda_a + A_a) \cos[(\varpi_a + \omega_a)t + (\psi_a + \varphi_a)] \cdot r^1(t), \\ X_B^1(t) = (\Lambda_b + A_b) \cos[(\varpi_b + \omega_b)t + (\psi_a + \varphi_a)] \cdot r^1(t). \end{cases} \tag{3}$$

Thus, in order to achieve source indistinguishability, users need to move into proximity and add user-introduced randomness to amplitudes, phases, and angular frequencies simultaneously (in order to achieve spatial indistinguishability). Besides, in each round, each user sends the corresponding signals at a randomly chosen time (in order to achieve temporal indistinguishability). Executing these operations, the wireless channel can be transformed into an anonymous channel, and two users can establish a secret key. In the next section, we present the details of our key establishment protocol.

4. A key establishment protocol using keyless cryptography

This section presents an over-the-air key establishment protocol using keyless cryptography. Specifically, our protocol is designed for the following scenarios. Two nearby users, for instance Alice and Bob, need to directly share data with each other. To ensure the security and privacy of these data, a cryptographic key is needed. However, they do not have any pre-shared secrets, and there is no on-line trusted third party or infrastructure available. Fortunately, both Alice and Bob have wireless devices, such as PDAs, smart phones, tablets, and laptops. Making use of these devices, Alice and Bob can establish a secret key by implementing our protocol.

4.1. Overview

Our key establishment protocol consists of four phases:

- **Initialization**. In this phase, a trusted system authority generates public parameters, and the operations can be completed when it is off-line.
- **Training**. In this phase, two users move into proximity and exchange the public training signals. The training signals of our protocol is similar to the prelude of a song. Making use of the "prelude", two users evaluate the mismatch values of amplitudes, phases, and angular frequencies.
- **Signal transmission**. In this phase, users send analog signals. Specifically, in each round, in order to mask the mismatches and achieve spatial indistinguishability, Alice and Bob introduce randomness to the analog signals; Additionally, in order to achieve temporal indistinguishability, Alice and Bob send analog signals at randomly chosen times. At the end of each round, Alice and Bob obtain a secret bit.
- **Key establishment**. In this phase, two users establish a secret key.

4.2. Our key establishment protocol

This subsection presents our key establishment protocol.

Initialization. Before implementing our protocol, the initialization phase is activated, and public parameters are generated. For an input security parameter k, the trusted system authority chooses hash functions $H_1(\cdot)$ and $H_2(\cdot)$ from a collision-resistant hash family $\mathcal{H}$. The hash functions $H_1(\cdot)$ and $H_2(\cdot)$ are used to map an arbitrary finite input $\{0, 1\}^*$ to $\{0, 1\}^k$. Additionally, based on the security requirements, the trusted system authority chooses security parameter n (in practice, $n \ge k$). We denote by n the number of communication rounds. Then, the system authority generates training signals $X(t) = A\cos(\omega_c t + \varphi_0) \cdot r(t)$, the

Please cite this article in press as: Y. Zhang, et al., An over-the-air key establishment protocol using keyless cryptography, Future Generation Computer Systems (2016), http://dx.doi.org/10.1016/j.future.2016.12.013

system parameters T_r (i.e., the time interval) and T_s (i.e., the time duration of each round), and decides the modulation ranges of amplitude (α), phase (φ), and angular frequency (w). At the end of this phase, the system authority publishes public parameters $<k, n, H_1(\cdot), H_2(\cdot), X(t), T_r, T_s, \alpha, \varphi, w>$.

Training. In this phase, two users move into proximity and exchange the training signals $X(t)$ publicly. Specifically, in this phase:

- Two users Alice and Bob (we assume that Alice is the initiator, and Bob is the responder) show off their wireless devices, and move their devices close to each other. Typically, they should ensure that the distance d of their devices is no more than $\frac{\lambda}{2}$.[3]
- Alice sends the training signals $X_A(t) = A_a \cos(\omega_a t + \varphi_a) \cdot r(t)$ to Bob.[4] Recall that the training signals are $X(t) = A\cos(\omega_c t + \varphi_0) \cdot r(t)$. Due to hardware and manufacturing inconsistencies, the amplitude A_a, phase φ_a, and angular frequency ω_a have slight variations. Namely, $A_a \neq A$, $\varphi_a \neq \varphi_0$ and $\omega_a \neq \omega_c$.
- Receiving the training signals $X_A(t)$, Bob makes the response by sending the training signals $X_B(t) = A_b \cos(\omega_b t + \varphi_b) \cdot r(t)$ to Alice.
- Completing these transmissions, both Alice and Bob evaluate the mismatch values of the transmitted signals. We denote by $\triangle A = |A_a - A_b|$, $\triangle\varphi = |\varphi_a - \varphi_b|$, and $\triangle\omega = |\omega_a - \omega_b|$ the mismatch values of amplitudes, phases, and angular frequencies. Without loss of generality, we assume that $A_a \geq A_b$, $\varphi_a \geq \varphi_b$, and $\omega_a \geq \omega_b$. Thus, we have $\triangle A = A_a - A_b$, $\triangle\psi = \varphi_a - \varphi_b$, and $\triangle\omega = \omega_a - \omega_b$.

Signal transmission. In this phase, Alice and Bob randomly choose n bits $C_a \in \{0, 1\}^n$ and $C_b \in \{0, 1\}^n$, respectively. Then, in the ith round (where $i = 1, 2, \ldots, n$):

- Alice randomly chooses $\varLambda_a \in [-\alpha, \alpha]$, $\psi_a \in [-\varphi, \varphi]$, and $\varpi_a \in [-w, w]$, and converts the ith bit of C_a to the corresponding analog signals $X_A^i(t) = (\varLambda_a + A_a)\cos[(\varpi_a + \omega_a)t + (\psi_a + \varphi_a)] \cdot r^i(t)$. Then, Alice sends the ith signals $X_A^i(t)$ at time t_i^A, where t_i^A is chosen uniformly at random in the interval $[(i-1)T_r, iT_r]$.
- Similarly, Bob randomly chooses $\varLambda_b \in [-\triangle A - \alpha, \triangle A + \alpha]$, $\psi_b \in [-\triangle\varphi - \varphi, \triangle\varphi + \varphi]$, and $\varpi_b \in [-\triangle\omega - w, \triangle\omega + w]$, and converts the ith bit of C_b to the corresponding analog signals $X_B^i(t) = (\varLambda_b + A_b)\cos[(\varpi_b + \omega_b)t + (\psi_b + \varphi_b)] \cdot r^i(t)$. Then, Bob sends the ith signals $X_B^i(t)$ at time t_i^B, where t_i^B is chosen uniformly at random in the interval $[(i-1)T_r, iT_r]$.
- Completing the aforementioned transmissions, Alice and Bob: (a) obtain the ith secret bit, or (b) discard it. Specifically, Alice (and Bob) discards the ith bit, if the number of transmitted analog signals in the ith round is not equal to 2. Else, Alice (and Bob) sets the ith bit of the key K_a (and K_b) to be "1", if the first signals in this round are sent by Alice; Otherwise, it is set to be "0".

Executing the **Signal Transmission** n rounds, Alice obtains the key K_a, and Bob obtains the key K_b.

Key establishment. In this phase, Alice and Bob establish a secret key K_{AB} (K_{BA}). Specifically, Alice and Bob execute following operations:

- Alice computes $V_a = H_1(K_a \parallel 1)$, and sends the V_a to Bob. Here, the "$\parallel$" is the string concatenation.
- Receiving the V_a, Bob computes $V_a' = H_1(K_b \parallel 1)$, and verifies if $V_a' = V_a$. Bob computes $V_b = H_1(K_b \parallel 0)$ and the secret key $K_{BA} = H_2(K_b)$, and sends the V_b to Alice, if the verification succeeds; Otherwise, Bob terminates the executions immediately.
- Similarly, receiving the V_b, Alice computes $V_b' = H_1(K_a \parallel 0)$, and verifies if $V_b' = V_b$. Alice computes the secret key $K_{AB} = H_2(K_a)$, if the verification succeeds; Otherwise, Alice terminates the executions immediately.

In an honest execution of the protocol, we have $K_a = K_b$. Thus, the established secret key $K_{AB} = H_2(K_a) = H_2(K_b) = K_{BA}$. This completes the description of our protocol.

Remark 1. A limitation of our key establishment protocol is that, it can only be applied to the scenarios when two users are "neighbors" or they can move into proximity. To be more exact, the distance (d) between two wireless devices should no more than half a wavelength ($\frac{\lambda}{2}$) when they establish a secret key by implementing our protocol.[5]. However, the limitation provides a benefit. Namely, our protocol naturally provides authentication. It is due to the reason that, two users can authenticate each other when they are physically close to one another.

5. Security analysis

This section analyzes the security of our key establishment protocol. Specifically, for a passive adversary $\mathcal{A}$, she eavesdrops the transmitted signals and conducts sophisticated signal processing. The adversary $\mathcal{A}$ succeeds if she can identify the source of the eavesdropped signals.

As analyzed in [10] that, according to *the law of free space path loss*, the strength of the transmitted signals is modulated as:

$$S_r = \frac{S_t G_t G_r E}{d^2}. \tag{4}$$

Here, S_r is the reception power, $S_t = \frac{A^2}{2}$ is the power of transmitted signal,[6] G_t and G_r are the antenna gains of the transmitter and receiver, E is a constant coefficient (it depends on the frequency of the transmitted signal), and d is the distance between the transmitter and the receiver. Other factors, such as reflection and diffraction caused by obstacles, are not considered [10]. To simplify the investigation, we assume that the wireless devices are equipped with the same types of antenna. Thus, the above equation can be rewritten as:

$$S_r = \frac{kA^2}{2d^2}, \tag{5}$$

where $k = G_t G_r E$. If the powers of the transmitted signals (sent by Alice) are uniformly distributed between $[\frac{A^2}{2}, \frac{A^2}{2} + \delta]$, the powers of the received signals at the adversary side are uniformly distributed between $[\frac{kA^2}{2d_a^2}, \frac{k(A^2+2\delta)}{2d_a^2}]$; Similarly, the powers of the received signals at the adversary side are uniformly distributed between $[\frac{kA^2}{2d_b^2}, \frac{k(A^2+2\delta)}{2d_b^2}]$, if the powers of the transmitted signals (sent by Bob) are uniformly distributed between $[\frac{A^2}{2}, \frac{A^2}{2} + \delta]$, where d_a is the distance between Alice and the adversary, and d_b is the distance between Bob and the adversary. Without loss of generality, we assume $d_a \leq d_b$. Fig. 4 roughly shows the received powers of

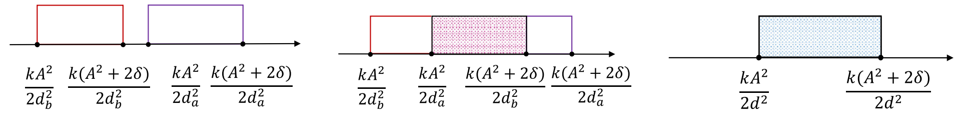
[3] Take the IEEE standard 802.15.4 as an example. The 802.15.4 specifies the frequency bands of the physical layer [25], i.e., 868 MHz, 915 MHz, and 2400 MHz. Thus, we can evaluate that $\frac{\lambda}{2} \approx 17.28$ cm when the frequency band is 868 MHz; $\frac{\lambda}{2} \approx 16.39$ cm when the frequency band is 915 MHz; and $\frac{\lambda}{2} \approx 6.25$ cm when the frequency band is 2400 MHz.

[4] Note, that our protocol is designed for the scenarios that two nearby users need to directly share data with each other. Thus, in our protocol, all signals are directly transmitted between Alice and Bob, without employing any forwarders and routers.

[5] Namely, $d \leq \frac{\lambda}{2} \approx 17.28$ cm when the frequency band is 868 MHz; $d \leq \frac{\lambda}{2} \approx 16.39$ cm when the frequency band is 915 MHz; and $d \leq \frac{\lambda}{2} \approx 6.25$ cm when the frequency band is 2400 MHz.

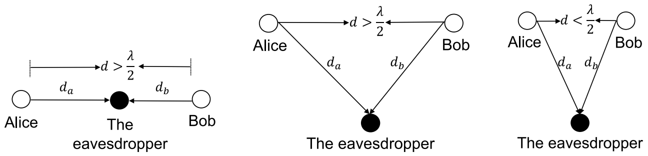
[6] Recall that the transmitted analog signals are $X^i(t) = A\cos(\omega t + \varphi) \cdot r^i(t)$. Thus, we have $S_t = \frac{A^2}{2}$.

Please cite this article in press as: Y. Zhang, et al., An over-the-air key establishment protocol using keyless cryptography, Future Generation Computer Systems (2016), http://dx.doi.org/10.1016/j.future.2016.12.013


$\frac{kA^2}{2d_b^2}$
$\frac{k(A^2+2\delta)}{2d_b^2}$
$\frac{kA^2}{2d_a^2}$
$\frac{k(A^2+2\delta)}{2d_a^2}$
$\frac{kA^2}{2d_b^2}$
$\frac{kA^2}{2d_a^2}$
$\frac{k(A^2+2\delta)}{2d_b^2}$
$\frac{k(A^2+2\delta)}{2d_a^2}$
$\frac{kA^2}{2d^2}$
$\frac{k(A^2+2\delta)}{2d^2}$

(a) Condition 1. (b) Condition 2. (c) Condition 3.

Fig. 4. The received powers of the signals at the adversary side, where $d_a \leq d_b$.



(a) Scenario 1. (b) Scenario 2. (c) Scenario 3.

Fig. 5. Different scenarios in practice when $d_a = d_b$.

the transmitted signals. From a mathematical point of view, at the adversary side, the signals sent by Alice are statistically indistinguishable from the signals sent by Bob when condition 3 ($d_a = d_b$) happens. However, the adversary can identify parts of these signals when condition 2 ($d_a < d_b \leq \sqrt{\frac{A^2+2\delta}{A^2}}d_a$) happens. Additionally, the adversary can identify the source of all the signals when condition 1 ($d_b > \sqrt{\frac{A^2+2\delta}{A^2}}d_a$) happens.

It is analyzed in [10] that the protocol achieves spatial indistinguishability if (1). condition 3 ($d_a = d_b$) happens, or (2). users shake those two devices during the key establishment phase (such that d_a and d_b are statistically indistinguishable). In order to achieve spatial indistinguishability, our protocol tries to ensure that the condition 3 happens. In practice, however, different scenarios exist when $d_a = d_b$ (as shown in Fig. 5). Specifically, in scenarios 1 and 2, the transmitted signals (sent by Alice and Bob) undergo different wireless fading channels. According to the spatial decorrelation property of the wireless channel, the adversary can identify the source of the signals in these scenarios. More specifically, when the distance d between Alice and Bob is larger than $\frac{\lambda}{2}$ (recall that λ is the wavelength of the transmitted signals), the adversary can identify the source of the transmitted signals, e.g., by making use of the temporal link signatures [20,21]. Thus, in order to ensure condition 3 happens and achieve spatial indistinguishability, two users Alice and Bob in our protocol need to move into proximity, i.e., $d \leq \frac{\lambda}{2}$.

Additionally, due to hardware and manufacturing inconsistencies, slight variations are introduced in the amplitudes, phases, and frequencies of the transmitted signals. To mask these variations and achieve spatial indistinguishability, users in our protocol introduce randomness to amplitudes, phases, and angular frequencies simultaneously. More specifically, Alice and Bob send analog signals, and the signals can be written as Eq. (1). Without loss of generality, in our paper we assume that $A_a \geq A_b$, $\varphi_a \geq \varphi_b$, and $\omega_a \geq \omega_b$. Thus, slight amplitudes mismatch $\triangle A = A_a - A_b$, slight phases mismatch $\triangle\psi = \varphi_a - \varphi_b$, and slight angular frequencies mismatch $\triangle\omega = \omega_a - \omega_b$. In order to mask these mismatches, Alice randomly chooses $\Lambda_a \in [-\alpha, \alpha]$, $\psi_a \in [-\varphi, \varphi]$, and $\varpi_a \in [-w, w]$, and modulates the transmitted analog signals $X_A^i = (\Lambda_a + A_a)\cos[(\varpi_a + \omega_a)t + (\psi_a + \varphi_a)] \cdot r^i(t)$. Similarly, Bob randomly chooses $\Lambda_b \in [-\triangle A - \alpha, \triangle A + \alpha]$, $\psi_b \in [-\triangle\varphi - \varphi, \triangle\varphi + \varphi]$, and $\varpi_b \in [-\triangle\omega - w, \triangle\omega + w]$, and modulates the analog signals $X_B^i = (\Lambda_b + A_b)\cos[(\varpi_b + \omega_b)t + (\psi_b + \varphi_b)] \cdot r^i(t)$. All these operations ensure that the adversary cannot identify the source of the eavesdropped signals.

In Fig. 6, we take the amplitudes as an example. Specifically, we assume $A_a \geq A_b$, and $\triangle A = A_a - A_b = 0.3A_b$. Additionally, the parameter $\alpha = 0.5A_b$. Then, Alice and Bob run the random selection algorithm $\Lambda_a \in [-\alpha, \alpha]$ and $\Lambda_b \in [-\triangle A - \alpha, \triangle A + \alpha]$. Namely, $\Lambda_a + A_a \in [-A_a - \alpha, A_a + \alpha]$ and $\Lambda_b + A_b \in [-A_b - \triangle A - \alpha, A_b + \triangle A + \alpha]$. From Fig. 6 we can see that, the amplitudes of the signals (sent by Alice and Bob) are mixed. As a result, the adversary cannot identify the source of the signals by eavesdropping amplitudes of the transmitted signals. It is easy to see that the adversary cannot identify the source of the signals, when Alice and Bob introduce randomness to the amplitudes, phases, and frequencies, simultaneously. Thus, our protocol is secure against the passive eavesdropping attacks.

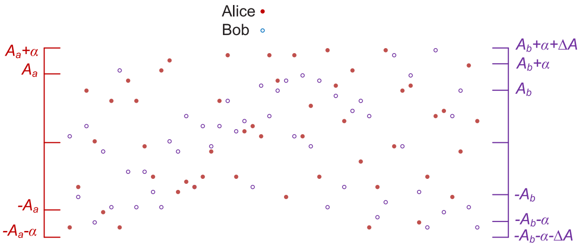


Fig. 6. To achieve indistinguishability, Alice and Bob randomly chooses $\Lambda_a \in [-\alpha, \alpha]$ and $\Lambda_b \in [-\triangle A - \alpha, \triangle A + \alpha]$ (take the amplitudes as an example). In this example, $\alpha = 0.5A_b$, and $\triangle A = 0.3A_b$.

In certain scenarios, the variations $\triangle A \ll A_a$ (A_b), $\triangle\psi \ll \varphi_a$ (φ_b), and $\triangle\omega \ll \omega_a$ (ω_b). As a result, users' devices cannot sense the minor variations. Thus, both Alice and Bob randomly choose $\Lambda \in [-\alpha, \alpha]$, $\psi \in [-\varphi, \varphi]$, and $\varpi \in [-w, w]$, and modulates the transmitted analog signals. However, it is assumed that the adversary can capture the transmitted signals with high sensitivity, i.e., the adversary can sense the variations. Now, we investigate the security of our protocol in this scenarios.

To simplify the analysis, we assume $\triangle A \ll A_a$ (A_b), $\triangle\psi = 0$, and $\triangle\omega = 0$. Namely, minor variations only exist in the amplitudes. Similarly, we assume $A_a > A_b$. Thus, $\triangle A = A_a - A_b$. For honest executions, the amplitudes of the transmitted signals (sent by Alice) are uniformly distributed in $[-A_a - \alpha, A_a + \alpha]$ (i.e., $[-A_b - \alpha - \triangle A, A_b + \alpha + \triangle A]$). Similarly, the amplitudes of the transmitted signals (sent by Bob) are uniformly distributed in $[-A_b - \alpha, A_b + \alpha]$. In this scenarios, the adversary can identify the source of the signals when the amplitudes of the transmitted signals are distributed in $[-A_b - \alpha - \triangle A, -A_b - \alpha] \cup [A_b + \alpha, A_b + \alpha + \triangle A]$. Thus, in any round, the adversary can correctly identify the source of the signals with probability:

$$\begin{aligned} \mathrm{P} &= p_1 + p_2 \\ &= \frac{1}{2} \times \frac{A_b + \alpha}{A_b + \alpha + \triangle A} + 1 \times \frac{\triangle A}{A_b + \alpha + \triangle A} \\ &= \frac{A_b + \alpha + 2\triangle A}{2(A_b + \alpha + \triangle A)} \end{aligned} \tag{6}$$

Table 1
Difference scenarios when $\triangle\psi = 0$, $\triangle\omega = 0$, and $|k| = 112$.

	s1	s2	s3	s4	s5	s6	s7
α	$0.5A_b$	$0.5A_b$	$0.5A_b$	$0.5A_b$	$0.5A_b$	$0.5A_b$	$0.5A_b$
$\triangle A$	$0.01A_b$	$0.03A_b$	$0.05A_b$	$0.07A_b$	$0.1A_b$	$0.2A_b$	$0.3A_b$
p_2	0.66%	1.96%	3.23%	4.46%	6.25%	11.77%	16.67%
n	113	115	116	118	120	127	135

where $p_1 = \frac{1}{2} \times \frac{A_b+\alpha}{A_b+\alpha+\triangle A}$ (i.e., the adversary successfully guess the source of the signals), and $p_2 = 1 \times \frac{\triangle A}{A_b+\alpha+\triangle A}$ (i.e., the adversary successfully identify the source of the signals when the amplitudes of the transmitted signals are distributed in $[-A_b-\alpha-\triangle A, -A_b-\alpha] \cup [A_b+\alpha, A_b+\alpha+\triangle A]$). Therefore, in any round, the advantage of the adversary, namely $Adv(\mathcal{A})$, in identifying the source of the signals is:

$$Adv(\mathcal{A}) = 2 \times \mathrm{P} - 1 = \frac{\triangle A}{A_b + \alpha + \triangle A}. \tag{7}$$

Table 1 shows the probabilities in different scenarios. Recall that in the **Initialization** phase of our protocol, the system authority chooses security parameter n (n is the number of round) based on the security parameter k and security requirements. To show the way to alleviate the above security problems, we provide examples when $|k| = 112$. For instance, to establish a key with 112 secret bits, i.e., $n \times (1\%–0.66\%) = 112$ when $\triangle A = 0.01A_b$, $\triangle\psi = 0$, and $\triangle\omega = 0$, we have $n = 113$. In other words, to counteract the problem (i.e., users' devices cannot sense the minor variations $\triangle A$), Alice and Bob need to send 113 signals, respectively. Similarly, we can get $n = 120$ when $\triangle A = 0.1A_b$. From the analysis we can see that, in order to alleviate the security problems (caused due to the reason that users' devices cannot sense the minor mismatches of the transmitted signals), each user needs to transmit extra $n - k$ signals. This introduces extra communication consumptions, and we will analysis the energy consumptions in Section 6. The above analysis illustrates that our protocol is secure against the passive eavesdropping attacks.

Recall that our protocol is designed for the scenarios that two users are in proximity, and they send wireless signals directly to each other (without employing any forwarders and routers). Thus, the adversary cannot tamper and delete the transmitted signals (please refer to the **Adversarial model** of our protocol for details). For an active adversary, she can insert bogus signals $X_{\mathcal{A}}(t)$ in certain round. When this happens, the number of received signals (at Alice and Bob) in that round is not equal to 2. According to our protocol, the corresponding bit will be discarded (please refer to the **Signal Transmission** phase of our protocol for details). Thus, our protocol is secure against such kind of attack. Besides, in our analysis, other active attacks, such as wireless denial of service (WDoS) attacks [26], are not discussed. Specifically, in the WDoS attacks, the active adversary $\mathcal{A}$ continually transmits signals. Such attacks do prevent legitimate users from establishing secret keys. However, in our protocol, the adversary aims to compromise the established keys (please refer to Section 3.1 for details), not to prevent legitimate users from establishing keys. Thus, these active attacks are not considered in our analysis.

Remark 2. The device fingerprinting may be extracted by making use of the features, such as clock-skew deviations, turn-on transients, and spectral transformations, and it might help the adversary to identify the source. However, the effectiveness of the identification procedures is still an open issue [12,24,27]. Additionally, it is reported in [23] that impersonation attacks on the modulation-based and transient-based fingerprinting techniques can be performed successfully. Thus, the device fingerprinting is not considered in our security analysis.

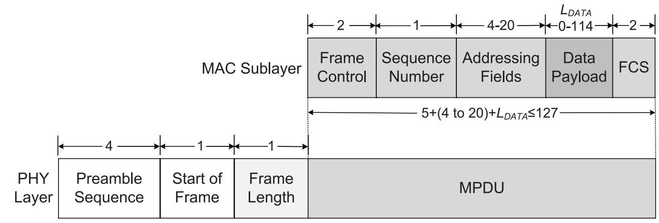


Fig. 7. Schematic view of packet structure.

6. Performance analysis and comparisons

In this section, we evaluate the performance of our key establishment protocol, and compare it with other protocols [3,10–12,17].

Recall that our protocol is designed based on protocols [10–12,17]. Specifically, to achieve source indistinguishability, users in [10] need to shake the devices. Additionally, to achieve source indistinguishability, random transmission powers are chosen in [11,12]. In our protocol, users have no need to shake the devices. However, in order to achieve source indistinguishability, two users in our protocol need to move into proximity and add user-introduced randomness to the transmitted signals (as suggested in [17]). Besides, our protocol enhanced the security of protocols [10–12]. It is pointed out in [10] that, the protocol is vulnerable to the key poisoning attacks if a single message is transmitted in each round. The key poisoning attacks is that, in each round, the adversary insert a bogus packet. Receiving the inserted packet, two legitimate users generate different "secret" bits (the "secret" bits are predictable at the adversary side). To defeat the key poisoning attack, each user in our protocol needs to transmit a bit in each round (as suggested in [10]). Namely, for the honest executions of our protocol, two analog signals are transmitted in each round. One is sent by Alice, the other is sent by Bob. Thus, Alice and Bob can identify the key poisoning attacks once the number of received signals in certain round is not equal to 2. Obviously, the energy and time consumptions of our protocol are the same with that of protocols [10–12], when the security of [10–12] are enhanced.

Thus, in the following paragraphs, we compare the consumptions of our protocol with that of the protocol [17] and the Diffie–Hellman key exchange protocol [3]. We assume that two devices transmit data by running the unslotted CSMA–CA MAC (in the IEEE 802.15.4). Fig. 7 reviews the packet structure of IEEE 802.15.4 [25,28].

Recall that in each round, each user in our protocol sends one bit. In practical applications, other messages, such as Preamble Sequence and Start of Frame, should be included in each packet (as shown in Fig. 7). Thus, in order to achieve a security level of 112 bits, each user in our protocol needs to send and receive 112 packets. Namely, the total number of bits that each device sent and received is 10 752 bits[7]: $112 \times (4 + 1 + 1 + 5 + 1) \times 8$. Besides, transmitting one bit consumes around as much power as executing 800–1000 instructions. Additionally, receiving one bit consumes about half as much power as sending one bit [10,29]. Thus, in our protocol, each device consumes as much energy as executing $10\,752 \times 1000 + 10\,752 \times 500 \approx 1.61 \times 10^7$ instructions.

Similarly, to achieve a security level of 112 bits in [17], each user needs to send and receive 312 bits: $1 \times (4 + 1 + 1 + 5 + 28) \times 8$ (considering 50% probability that a bit is effective in [17]). Thus, the energy consumption at each device is as much as executing $312 \times 1000 + 312 \times 500 = 4.68 \times 10^5$ instructions.

[7] To achieve source indistinguishability, a. the addressing fields should be set empty; and b. the randomness should be introduced to the transmitted signals (which are converted from the packet).

Please cite this article in press as: Y. Zhang, et al., An over-the-air key establishment protocol using keyless cryptography, Future Generation Computer Systems (2016), http://dx.doi.org/10.1016/j.future.2016.12.013

Besides, a security equivalent to 112 bits requires to select a modulus of 2048 bits and an exponent of 224 bits. Thus, each device needs to send 2048 bits when implementing the Diffie–Hellman key exchange protocol. Recall that in a IEEE 802.15.4 packet, the maximum data payload is $114 \times 8 = 912$ bits. Namely, each device needs to send and receive 3 packets. Thus, we can obtain the total number of bits that each device sent and received is 2504 bits: $2 \times (4+1+1+127) \times 8 + [4+1+1+(127-114+28)] \times 8$. These consumes as much energy as executing $2504 \times 1000 + 2504 \times 500 \approx 3.76 \times 10^6$ instructions. Receiving other user's packets (i.e., g^b), each user needs to exponentiate it using its Diffie–Hellman private key (i.e., computes $(g^b)^a$). It requires $3 \times l \times (l+1) \times (t+1)$ single-precision multiplications, when exponentiating using the Montgomery algorithm [30,10]. Namely, each device needs to perform 2.83×10^9 single-precision multiplications when $l = 2048$ and $t = 224$. Thus, the total consumption at each device is equivalent to execute $3.76 \times 10^6 + 2.83 \times 10^9$ instructions.

From the above analysis we can see that, in our protocol, the energy consumption at each device is around 35 times more expensive than the consumption in [17], but energy consumption of our protocol is about 176 times cheaper than that of the Diffie–Hellman key exchange protocol [3].

Now, we analyze the time consumptions of our protocol and protocols [17]. In protocol [17], each device only needs to send 1 packet. While in our protocol, each device needs to send 112 packets. More specifically, it takes 312 bit/250 kbps $\approx$ 1.25 ms to send the packet in [17], and it takes 96 bit/250 kbps $\approx$ 0.38 ms to send one packet in our protocol, where the bit rate is assumed to be 250 kbps.[8] However, the time consumption of our protocol depends on the time duration of each round (recall that each device sends a single packet in each round). We denote by T_s the time duration of each round. Thus, in our protocol, it takes $T = 112 \times T_s$ ms to establish a key with 112 secret bits. Now, we evaluate the duration T_s of each round.

Recall that in the ith round, each device sends a packet at time t_i, where t_i is chosen uniformly at random in the interval $[(i-1)T_r, iT_r]$. Besides, in our protocol, it takes $T_p = 0.38$ ms to send a packet. When devices work at half-duplex mode, certain collision avoidance mechanism, such as CSMA/CA, should be employed. Let $T_r = 0.40$ ms. Thus, the time duration is $T_s = t_{II} + T_p$, when $t_I + 0.38 < t_{II} \leq 0.4$ ms (i.e., there is no back-off happens in this round). Here, t_I is the time when the first packet is sent, and t_{II} is the time when the second packet is sent. Similarly, the time duration is $T_s = t_{II} + T_{bp} + T_p$, when $t_I + 0.06 < t_{II} \leq t_I + 0.38$ ms (i.e., one back-off happens in this round). Here, we denote by T_{bp} the back-off time. In the unslotted CSMA/CA mechanism of the IEEE 802.15.4, one back-off time $T_{bp} = 0.32$ ms [28]. When two back-off happens, the time duration is $T_s = t_{II} + 2T_{bp} + T_p$, when $t_I < t_{II} \leq t_I + 0.06$ ms. Thus, we have Eq. (8).

$$T_s = \begin{cases} t_{II} + 2T_{bp} + T_p = t_{II} + 1.02 & t_{II} \in (t_I, t_I + 0.06 \text{ ms}] \\ t_{II} + T_{bp} + T_p = t_{II} + 0.70 & t_{II} \in (t_I + 0.06 \text{ ms}, \\ \quad t_I + 0.38 \text{ ms}] & \\ t_{II} + T_p = t_{II} + 0.38 & t_{II} \in (t_I + 0.38 \text{ ms}, 0.4 \text{ ms}]. \end{cases} \qquad (8)$$

In Fig. 8, we investigate the expected time duration T_s when $T_r = 0.40$ ms. In this scenarios, it can be estimated that in each round, the probability of no back-off is about 5%, one back-off happens with probability about 80%, and two back-off happens with probability about 15%. From Fig. 8 and Eq. (8) we can see that, in order to ensure all signals are transmitted successfully, the T_s should not less than 1.42 ms (when $T_r = 0.40$ ms). Thus, $T = 112 \times T_s = 159.04$ ms when $T_s = 1.42$ ms. Namely, in our protocol, it takes about 159.04 ms to establish a key with 112 secret bits, when devices work at half-duplex mode.

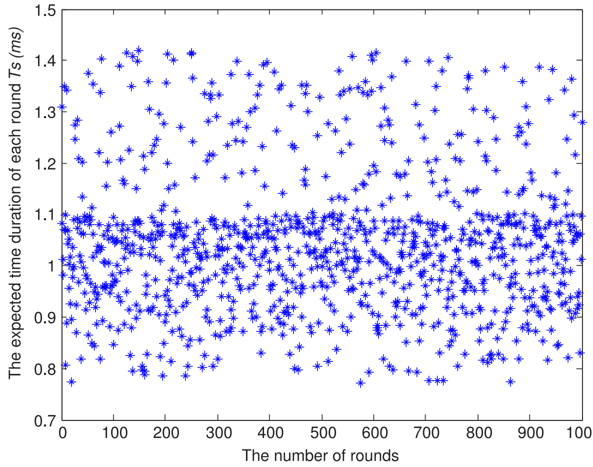


Fig. 8. The expected time duration T_s of each round. Specifically, in this example, Alice and Bob run the random number generation algorithm 1000 times.

When devices work at full-duplex mode, there is no need to employ collision avoidance mechanisms. Thus, we get $T_s = 0.78$ ms, when $T_r = 0.4$ ms. In this scenarios, it takes around $T = 112 \times T_s = 87.36$ ms to establish a key with 112 secret bits.

From the analysis we can see that, the energy and time consumptions of our protocol and that of protocols [10–12] are in the same order of magnitude. Besides, the energy consumption of our protocol is about 176 times cheaper than that of the Diffie–Hellman key exchange protocol [3], and our protocol only takes around 159.04 ms (when working at half-duplex mode) or 87.36 ms (when working at full-duplex mode) to establish a key with 112 secret bits. Comparing with protocol [17], however, the energy consumption of our protocol is about 35 times more expensive. Furthermore, in [17], it takes about 1.25 ms to establish a key with 112 secret bits. Thus, in terms of energy and time consumptions, the protocol [17] achieves a better performance than our protocol. This is due to the reason that in [17], 112 bits are sent using a single packet. In our protocol, however, 112 bits are sent using 112 packets. This incurs extra energy and time consumptions.

However, comparing with [17], our protocol can be applied to broader scenarios. Recall that protocol [17] is designed for the full-duplex communications. Namely, protocol [17] only can be applied to the scenarios when devices work at full-duplex mode (please refer to Section 2.2 for details). However, our protocol do not have this limitation. Namely, our protocol can be applied to scenarios when devices work at full-duplex mode or half-duplex mode.

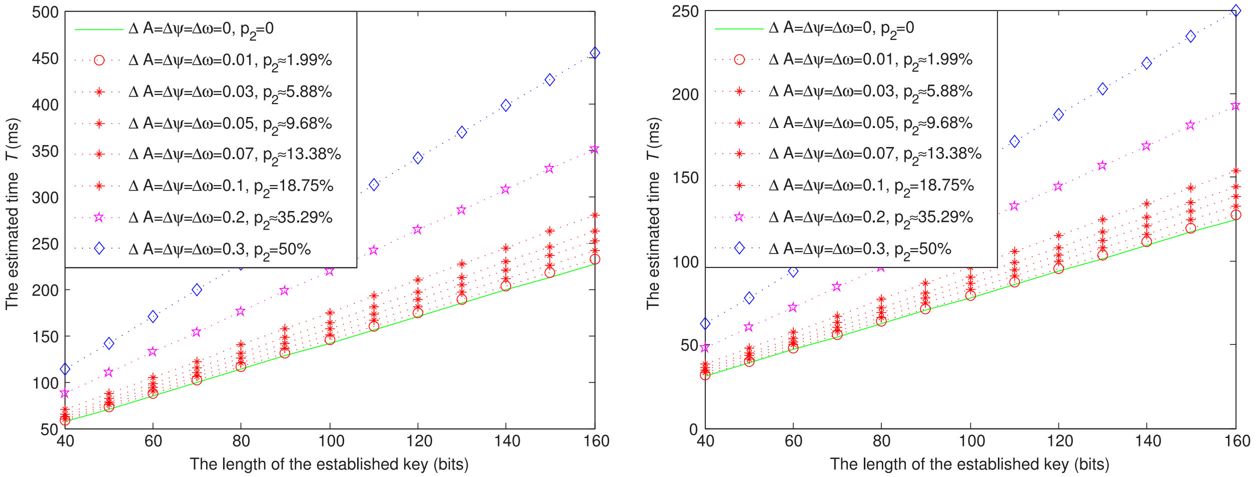
We conclude the time and energy consumptions of our protocol in Table 2 and Fig. 9. Recall that the bit rate is 250 kbps and $T_r = 0.4$ ms. In order to ensure that all signals are transmitted successfully, we get $T_s = 1.42$ ms when devices work at half-duplex mode, and $T_s = 0.78$ ms when devices work at full-duplex mode. Thus, we get the estimated time (our protocol consumes) is $T = \frac{k}{1-p_2} \times 1.42$ ms, when devices work at half-duplex mode, and the estimated time is $T = \frac{k}{1-p_2} \times 0.78$ ms, when devices work at full-duplex mode. Here, k is the length of the established key, p_2 is the probability that the adversary successfully identify the source of the signals, due to the reason that users' devices cannot sense the minor variations. Please refer to Eq. (6) and Table 1 for details. Specifically, when $p_2 = 0$, it implies that the users' devices can sense the variations. Besides, we can get that each device consumes as much energy as executing $\frac{k}{1-p_2} \times (4+1+1+5+1) \times 8 \times 1000 + \frac{k}{1-p_2} \times (4+1+1+5+1) \times 8 \times 500 = \frac{k}{1-p_2} \times 14\,400$ instructions. In Fig. 9, we plot the estimated time consumptions, without plotting the estimated energy consumptions. It is easy to

[8] IEEE 802.15.4 supports the over-the-air data rates of 20 kbps, 40 kbps, and 250 kbps.

Table 2
Time and energy consumptions of our protocol when it works at half-duplex mode and full-duplex mode.

	$T\|k, p_2 = 0$[a]	$T\|k, p_2 \neq 0$	$E\|k, p_2 = 0$[b]	$E\|k, p_2 \neq 0$
Half-duplex mode	$k \times 1.42$	$\frac{k}{1-p_2} \times 1.42$	$k \times 144\,000$	$\frac{k}{1-p_2} \times 144\,000$
Full-duplex mode	$k \times 0.78$	$\frac{k}{1-p_2} \times 0.78$	$k \times 144\,000$	$\frac{k}{1-p_2} \times 144\,000$

[a] $T|k, p_2 = 0$ is the time used to establish a secret key when the key length is k bits and $p_2 = 0$.
[b] $E|k, p_2 = 0$ is used to evaluate the consumed energies (by counting the executed instructions) when the key length is k bits and $p_2 = 0$.



(a) Devices work at half-duplex mode.

(b) Device work at full-duplex mode.

Fig. 9. The estimated time that our protocol consumes, when devices work at half-duplex mode and full-duplex mode. To simplify the analysis, we let $\triangle A = \triangle\psi = \triangle\omega$.

see that the trend of energy consumptions is similar to that of the time consumptions.

From Table 2 and Fig. 9 we can see that, our protocol consumes more times and energies when longer keys need to be established. For instance, when devices work at half-duplex mode, it takes $T = k \times 1.42$ ms $= 112 \times 1.42$ ms $= 159.04$ ms and consumes as much energy as executing $E = k \times 144\,000 = 112 \times 144\,000 = 1.61 \times 10^7$ instructions, when the input security parameter $k = 112$ (i.e., the length of the established key is 112 bits), and $p_2 = 0$. It takes $T = k \times 1.42$ ms $= 128 \times 1.42$ ms $= 181.76$ ms and consumes as much energy as executing $E = k \times 144\,000 = 128 \times 144\,000 = 1.84 \times 10^7$ instructions, when the input security parameter $k = 128$, and $p_2 = 0$.

Besides, from Fig. 9 we can see that, the proposed protocol consumes more times when the minor variations increased. In order to simplify the analysis, we assume that $\triangle A = \triangle\psi = \triangle\omega$ and $\alpha = 0.5$. For example, it takes $T = 159.04$ ms to establish a secret key, when devices work at half-duplex mode (as shown in Fig. 9(a)), $k = 112$ bits, and $p_2 = 0$ (i.e., $\triangle A = \triangle\psi = \triangle\omega = 0$). When minor variations $\triangle A = \triangle\psi = \triangle\omega = 0.01$, i.e., $\triangle A = 0.01A_b$, $\triangle\psi = 0.01\psi_b$, and $\triangle\omega = 0.01\omega_b$, we have $p_2 = 3 \times \frac{\triangle A}{A_b+\alpha+\triangle A} \approx 1.99\%$. Thus, the estimated time is $T = \frac{k}{1-p_2} \times 1.42$ ms ≈ 162.27 ms when $p_2 = 1.99\%$. Similarly, we can get that the estimated time is $T \approx 195.74$ ms when $p_2 = 18.75\%$ (i.e., $\triangle A = \triangle\psi = \triangle\omega = 0.1$). From the above analysis we can see that the time and energy consumptions of our protocol linearly increase with the increase of the key length and the minor variations (when the minor variations cannot be sensed by users' devices).[9]

7. Conclusion

This paper provides a concrete construction to transform the wireless channel into an anonymous channel, and presents an over-the-air key establishment protocol using keyless cryptography. Specifically, the protocol is designed without using energy intensive asymmetric key cryptography and pre-shared secrets. To establish a secret key, two users in our protocol need to move into proximity and directly send random analog signals to each other. The analysis shows that our protocol is a low cost key establishment protocol, and it only takes around 159.04 ms (when working at half-duplex mode) or 87.36 ms (when working at full-duplex mode) to establish a key with 112 secret bits.

Acknowledgments

This work is supported by the Distinguished Young Scholars Fund of Fujian (2016J06013). Wei Wu is supported by the National Natural Science Foundation of China (61472083, 61402110), the CICAEET fund and the PAPD fund. Jian Shen is supported by the National Natural Science Foundation of China under Grant No. 61672295.

References

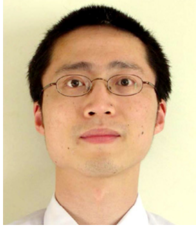
[1] P. Chen, L. Desmet, C. Huygens, A study on advanced persistent threats, in: B.D. Decker, A. Zúquete (Eds.), Communications and Multimedia Security - 15th IFIP TC 6/TC 11 International Conference, CMS 2014, Aveiro, Portugal, September 25–26, 2014. Proceedings, in: Lecture Notes in Computer Science, vol. 8735, Springer, 2014, pp. 63–72.

[9] From Table 2 we can see that, when $p_2 \neq 0$, $\frac{k}{1-p_2}$ is introduced when estimating the time and energy consumptions. Specifically, $p_2 = \frac{\triangle A}{A_b+\alpha+\triangle A}$ (we take the amplitude as an example, please refer to Eq. (6) for details). Then, we get $\frac{k}{1-p_2} = \frac{k(A_b+\alpha+\triangle A)}{A_b+\alpha}$. Thus, when the key length k and the modulation range of amplitude α are chosen (in practical applications, these parameters are chosen before the implementation of the protocol), the time and energy consumptions linearly increase with the increase of the minor variation $\triangle A$.

Please cite this article in press as: Y. Zhang, et al., An over-the-air key establishment protocol using keyless cryptography, Future Generation Computer Systems (2016), http://dx.doi.org/10.1016/j.future.2016.12.013

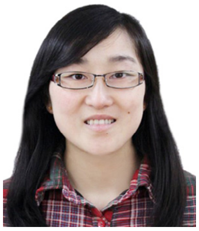
[2] R. Brewer, Advanced persistent threats: Minimising the damage, Netw. Secur. 2014 (2014) 5–9.
[3] W. Diffie, M.E. Hellman, New directions in cryptography, IEEE Trans. Inf. Theory 22 (1976) 644–654.
[4] L. Eschenauer, V.D. Gligor, A key-management scheme for distributed sensor networks, in: V. Atluri (Ed.), Proceedings of the 9th ACM Conference on Computer and Communications Security, CCS 2002, Washington, DC, USA, November 18–22, 2002, ACM, 2002, pp. 41–47.
[5] H. Chan, A. Perrig, D.X. Song, Random key predistribution schemes for sensor networks, in: 2003 IEEE Symposium on Security and Privacy (S&P 2003), 11–14 May 2003, Berkeley, CA, USA, IEEE Computer Society, 2003, p. 197.
[6] T. Wang, Y. Liu, A.V. Vasilakos, Survey on channel reciprocity based key establishment techniques for wireless systems, Wirel. Netw. 21 (2015) 1835–1846.
[7] J. Zhang, T.Q. Duong, A.J. Marshall, R.F. Woods, Key generation from wireless channels: A review, IEEE Access 4 (2016) 614–626.
[8] B. Alpern, F.B. Schneider, Key exchange using 'keyless cryptography', Inform. Process. Lett. 16 (1983) 79–81.
[9] M.M. Yung, A secure and useful keyless cryptosystem, Inform. Process. Lett. 21 (1985) 35–38.
[10] C. Castelluccia, P. Mutaf, Shake them up!: A movement-based pairing protocol for CPU-constrained devices, in: K.G. Shin, D. Kotz, B.D. Noble (Eds.), Proceedings of the 3rd International Conference on Mobile Systems, Applications, and Services, MobiSys 2005, Seattle, Washington, USA, June 6–8, 2005, ACM, 2005, pp. 51–64.
[11] R.D. Pietro, G. Oligeri, COKE crypto-less over-the-air key establishment, IEEE Trans. Inf. Forensics Secur. 8 (2013) 163–173.
[12] R.D. Pietro, G. Oligeri, ESC: An efficient, scalable, and crypto-less solution to secure wireless networks, Comput. Netw. 84 (2015) 46–63.
[13] E. Haselsteiner, K. Breitfuß, Security in near field communication (NFC), in: Workshop on RFID Security RFIDSec.
[14] M.M.A. Allah, Strengths and weaknesses of near field communication (NFC) technology, Glob. J. Comput. Sci. Technol. 11 (2011).
[15] J.I. Choi, M. Jain, K. Srinivasan, P. Levis, S. Katti, Achieving single channel, full duplex wireless communication, in: N.H. Vaidya, S. Banerjee, D. Katabi (Eds.), Proceedings of the 16th Annual International Conference on Mobile Computing and Networking, MOBICOM 2010, Chicago, Illinois, USA, September 20–24, 2010, ACM, 2010, pp. 1–12.
[16] M. Jain, J.I. Choi, T. Kim, D. Bharadia, S. Seth, K. Srinivasan, P. Levis, S. Katti, P. Sinha, Practical, real-time, full duplex wireless, in: P. Ramanathan, T. Nandagopal, B.N. Levine (Eds.), Proceedings of the 17th Annual International Conference on Mobile Computing and Networking, MOBICOM 2011, Las Vegas, Nevada, USA, September 19–23, 2011, ACM, 2011, pp. 301–312.
[17] R. Jin, X. Du, Z. Deng, K. Zeng, J. Xu, Practical secret key agreement for full-duplex near field communications, in: S. Moriai, T. Jaeger, K. Sakurai (Eds.), 9th ACM Symposium on Information, Computer and Communications Security, ASIA CCS '14, Kyoto, Japan - June 03 – 06, 2014, ACM, 2014, pp. 217–228.
[18] D. Bharadia, E. McMilin, S. Katti, Full duplex radios, in: D.M. Chiu, J. Wang, P. Barford, S. Seshan (Eds.), ACM SIGCOMM 2013 Conference, SIGCOMM'13, Hong Kong, China, August 12–16, 2013, ACM, 2013, pp. 375–386.
[19] K. Zeng, Physical layer key generation in wireless networks: Challenges and opportunities, IEEE Commun. Mag. 53 (2015) 33–39.
[20] N. Patwari, S.K. Kasera, Robust location distinction using temporal link signatures, in: E. Kranakis, J.C. Hou, R. Ramanathan (Eds.), Proceedings of the 13th Annual International Conference on Mobile Computing and Networking, MOBICOM 2007, Montréal, Québec, Canada, September 9–14, 2007, ACM, 2007, pp. 111–122.
[21] X. He, H. Dai, W. Shen, P. Ning, Is link signature dependable for wireless security? in: Proceedings of the IEEE INFOCOM 2013, Turin, Italy, April 14–19, 2013, IEEE, 2013, pp. 200–204.
[22] R.M. Gerdes, T.E. Daniels, M. Mina, S. Russell, Device identification via analog signal fingerprinting: A matched filter approach, in: Proceedings of the Network and Distributed System Security Symposium, NDSS 2006, The Internet Society, San Diego, California, USA, 2006.
[23] B. Danev, H. Luecken, S. Capkun, K.M.E. Defrawy, Attacks on physical-layer identification, in: S. Wetzel, C. Nita-Rotaru, F. Stajano (Eds.), Proceedings of the Third ACM Conference on Wireless Network Security, WISEC 2010, Hoboken, New Jersey, USA, March 22–24, 2010, ACM, 2010, pp. 89–98.
[24] B. Danev, D. Zanetti, S. Capkun, On physical-layer identification of wireless devices, ACM Comput. Surv. 45 (2012) 6.
[25] IEEE standard for local and metropolitan area networks–part 15.4: Low-rate wireless personal area networks (LR-WPANs), IEEE Std 802.15.4-2011 (Revision of IEEE Std 802.15.4-2006), 2011, pp. 1–314.
[26] K. Pelechrinis, M. Iliofotou, S.V. Krishnamurthy, Denial of service attacks in wireless networks: The case of jammers, IEEE Commun. Surv. Tutor. 13 (2011) 245–257.
[27] K.B. Rasmussen, S. Capkun, Implications of radio fingerprinting on the security of sensor networks, in: Third International Conference on Security and Privacy in Communication Networks and the Workshops, SecureComm 2007, Nice, France, 17–21 September, 2007, IEEE, 2007, pp. 331–340.
[28] B. Lauwens, B. Scheers, A.V. de Capelle, Performance analysis of unslotted CSMA/CA in wireless networks, Telecommun. Syst. 44 (2010) 109–123.
[29] C. Karlof, N. Sastry, D. Wagner, Tinysec: A link layer security architecture for wireless sensor networks, in: J.A. Stankovic, A. Arora, R. Govindan (Eds.), Proceedings of the 2nd International Conference on Embedded Networked Sensor Systems, SenSys 2004, Baltimore, MD, USA, November 3–5, 2004, ACM, 2004, pp. 162–175.
[30] A. Menezes, P.C. van Oorschot, S.A. Vanstone, Handbook of Applied Cryptography, CRC Press, 1996.

Yuexin Zhang received his B.S. degree from the Department of Physics and Electronic Information Engineering, Inner Mongolia Normal University, China, in 2010. and the M.S. degree from the School of Mathematics and Computer Science, Fujian Normal University, China, in 2013. He is currently pursuing the Ph.D. degree in Computer Science at Deakin University, Melbourne, Australia. His research interests include network security.



Yang Xiang received his Ph.D. in Computer Science from Deakin University, Australia. He is currently a full professor at School of Information Technology, Deakin University. He is the Director of Centre for Cyber Security Research, Deakin University. His research interests include network and system security, distributed systems, and networking. In particular, he is currently leading his team developing active defense systems against large-scale distributed network attacks. He is the Chief Investigator of several projects in network and system security, funded by the Australian Research Council (ARC). He has published more than 130 research papers in many international journals and conferences, such as IEEE Transactions on Computers, IEEE Transactions on Parallel and Distributed Systems, IEEE Transactions on Information Security and Forensics, and IEEE Journal on Selected Areas in Communications. Two of his papers were selected as the featured articles in the April 2009 and the July 2013 issues of IEEE Transactions on Parallel and Distributed Systems. He has published two books, Software Similarity and Classification (Springer) and Dynamic and Advanced Data Mining for Progressing Technological Development (IGI-Global). He has served as the Program/General Chair for many international conferences such as ICA3PP 12/11, IEEE/IFIP EUC 11, IEEE TrustCom 13/11, IEEE HPCC 10/09, IEEE ICPADS 08, NSS 11/10/09/08/07. He has been the PC member for more than 60 international conferences in distributed systems, networking, and security. He serves as the Associate Editor of IEEE Transactions on Computers, IEEE Transactions on Parallel and Distributed Systems, Security and Communication Networks (Wiley), and the Editor of Journal of Network and Computer Applications. He is the Coordinator, Asia for IEEE Computer Society Technical Committee on Distributed Processing (TCDP). He is a Senior Member of the IEEE.

Tao Wang is currently a third-year Ph.D. student in the Department of Computer Science and Engineering, University of South Florida, Tampa, Florida, USA. His research is related to wireless network, mobile security and cyber–physical system security. Currently, his research mostly focuses on securing the wireless communication by exploring the physical-layer features of the wireless channel.



Wei Wu received her Ph.D. degree from the School of Computer Science and Software Engineering, University of Wollongong, Australia, in 2011. She is currently an Associate Professor with the Fujian Provincial Key Laboratory of Network Security and Cryptology, School of Mathematics and Computer Science, Fujian Normal University, China. Her research focus is on public key cryptography and its applications. She has published more than 40 papers in refereed international journals and conferences.

Jian Shen received the B.E. degree from Nanjing University of Information Science and Technology, Nanjing, China, in 2007 and the M.E. and Ph.D. degrees in Computer Science from Chosun University, Gwangju, Korea, in 2009 and 2012, respectively. Since late 2012, he has been a full professor in the School of Computer and Software at Nanjing University of Information Science and Technology, Nanjing, China. His research interests include information security, public-key cryptography, security systems, network security, and mobile computing and networking.

Please cite this article in press as: Y. Zhang, et al., An over-the-air key establishment protocol using keyless cryptography, Future Generation Computer Systems (2016), http://dx.doi.org/10.1016/j.future.2016.12.013