

A Lightweight Authenticated Key Exchange Protocol that Can Resist to Continual Leakage Attacks for Wireless Body Area Network

Author: Wenjun Zeng

Abstract - As a wireless network of the wearable computing device, the wireless body area network (WBAN) plays an increasing role in healthcare and medical research. However, security remains the major challenge for WBAN, and many cryptographic primitives for WBAN are vulnerable to side-channel attacks, which poses serious threats on data security. In addition, the wide application of computationally limited sensors in WBAN also hampers the utilization of high-level cryptographic primitives. In this paper, a leakage-resilient and lightweight (LRL) ECDH-based authenticated key exchange (AKE) protocol is proposed. It is found that the protocol remains secure under continual leakage attacks by employing the blinding and refreshing technique for long-term secret keys. Moreover, it alleviates the computational burden on the computationally limited sensor node by transferring some computation tasks to the more powerful hub (or coordinator). A prototype of our proposed protocol is constructed, on which several experiments are performed to demonstrate that the proposed protocol is more efficient than similar protocols in mainstream communication standards for WBAN.

Keywords-leakage-resilient; authenticated key exchange; wireless body area network; elliptic-curve Diffie-Hellman

1. INTRODUCTION

Recent years have witnessed the rapid development of the wireless body area network (WBAN). The WBAN, also referred to as the Medical Sensor Network (MSN), is a wireless network of wearable sensing and actuating devices that can offer real-time and continuous monitoring of vital physical parameters [1] [2]. To date, WBAN has been widely applied in healthcare, sports, multimedia, military, security and many other sectors [3] [4] [5]. A typical WBAN architecture for remote healthcare is presented in Fig. 1. It consists of a number of sensors, which are usually situated on clothes, attached to human body or implanted inside the skin, and are used to monitor physical activities or the status of vital organs functionalities. The raw data collected from the sensors are transmitted to a coordinator or personal server which processes and aggregate the data which are then sent to a centralized server for permanent storage and real-time access.

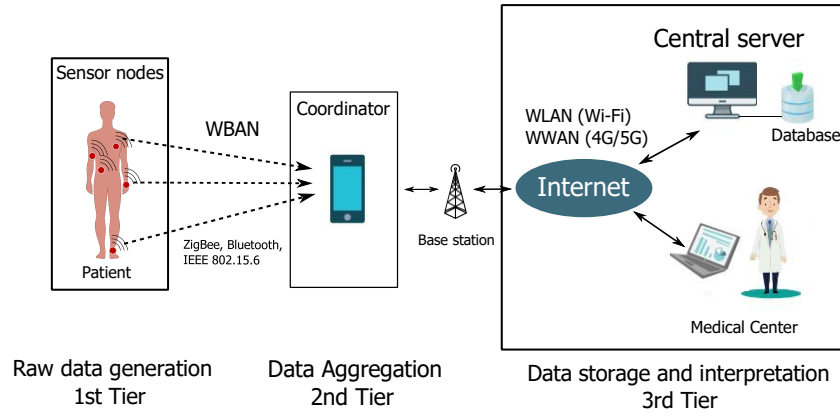


Fig. 1. Schematic architecture of WBAN

Data security remains one of the most significant concerns in developing and implementing many WBAN applications since the data collected by the sensors often contains sensitive information [6]. If such information is leaked to or distorted by illegal parties, serious issues might be elicited.

The Bluetooth Low Energy and IEEE 802.15.6 are two mainstreaming networking technologies for WBAN application. Although these techniques provide basic security solutions [7] [8] [9], the data security of current WBAN is still challenging due to the following factors:

- 1) Wide deployment of computationally limited devices in WBAN. The wearable sensors in a WBAN are designed to be low-power, lightweight and miniature in order to minimize the load of a person, thus they are inevitably less powerful in computation and memory storage than normal mobile devices [10]. As advanced cryptographic protocols that require larger computation power might overburden these sensors, the current communication standards deployed in WBAN only guarantee the very basic level of data security.
- 2) The increasing emergence of side-channel attacks. In recent years, side-channel attacks have received increasing attentions by researchers. In contrast to those standard adversarial models aiming at the weaknesses of cryptographic algorithms, side-channel attacks exploit unintentional information revealed by the physical behavior of a computing device that executes the cryptosystem [11] such as time consuming [12], power consumption [13], electromagnetic radiation [14] [15], thermal fluctuation [16] and so forth. However, the cryptographic schemes in most communication standards fail to fend off side channel attacks, making the WBAN vulnerable to data leakage.

Authenticated Key Exchange (AKE) is an important cryptographic primitive for two or more entities communicating over an open network [17] [18] [19]. It secures communication channels by establishing a shared secret key among each entity. A series of AKE protocols have been widely utilized in the WBAN networks such as protocols in Bluetooth Low Energy [20], Zigbee [21] and IEEE 802.15.6 [22] standards, but they fail to defend side-channel attacks. Although a number of leakage-resilient AKE protocols have been proposed [23, 24, 25, 26, 27, 28], few of them has been applied in the international communication as well as in safety standards mentioned above due to some constraints. Firstly, there is a paucity of reliable leakage-resilient AKE protocols based on the Elliptic-curve Diffie–Hellman (ECDH), which is a popular method for authenticated key exchange. Secondly, the majority of existing leakage-resilient AKE protocols

lack details of implementation, resulting in gaps between theoretical achievements and real-world applications.

Herein we propose a leakage-resilient and lightweight (LRL) ECDH-based AKE protocol employing the low-bandwidth out-of-band (OOB) channels to tackle the aforementioned issues. The main contributions of our work include:

- In our protocol, an unbalanced computation method is designed for low-power wireless networks. This design reduces the computational burden on the computationally limited sensor nodes and improves the protocol's efficiency. Compared to the Bluetooth Display and IEEE Display protocols in international standards, our protocol significantly reduces the computational time and CPU usage on the sensor nodes.
- A leakage-resilient ECDH-based protocol in the continual leakage model is proposed for the first time. The continual leakage model allows the adversary to know partial information of the permanent and ephemeral secret keys of two participants. In this protocol, the permanent secret key is split into two components and the two components are refreshed by employing the blinding technique [29, 24, 27]. Therefore, it remains secure in such leakage model.
- A concert construction and a use case of our protocol are given. Both the implementation details and the use case can narrow down the gaps between theories and practical applications and depict the further improvement of existing protocols for WBAN.

The rest of this paper is organized as follows: Section 2 reviews the relevant literatures on ECDH and leakage-resilient cryptography; Section 3 introduces the method of unbalanced ECDH and the blinding scheme for key refreshing. In Section 4, the design of the LRL AKE protocol and its security features are presented. While in Section 5, the prototype of the proposed protocol is implemented and its performance is studied. In Section 6, the application of the LRL AKE protocol in RPM system is illustrated through a use case. Finally, Section 7 summarizes our work and discusses future work.

2. RELATED WORK

This section reviews the previous work of ECDH and leakage-resilient key exchange protocols.

2.1 ECDH-based AKE

The ECDH-based AKE protocol [30] is an improved Diffie-Hellman protocol based on the Elliptic Curve Cryptography (ECC), which secures secret communications over a public network by generating a shared secret between two (or more) participants. The elliptic curve discrete logarithm problem (ECDLP) and elliptic curve Diffie-Hellman problem (ECEHP), and makes attackers difficult to obtain the secret keys, thus providing the security for the ECDH-based protocol. Compared to other protocols such as RSA and DH, ECDH key exchanges protocol dramatically reduces the size of encryption keys but maintain the same level of security [31]. Due to this advantage, ECDH-based AKE protocol is widely utilized in many international standards.

- *Bluetooth Low Energy*

The Bluetooth Low Energy (Bluetooth LE) is a short-range, low power-consumption network technology designed for applications in healthcare, home entertainment and so on. Compared with classic Bluetooth, Bluetooth LE is optimized for medical sensors which are required to be

low-power to suit small-sized device and long battery life. The Bluetooth 5.1 specification [20] supports devices to implement Bluetooth LE. There are five distinct security features in Bluetooth security mode, including pairing, bonding, device authentication, encryption and message integrity, among which the pairing process constitutes an AKE protocol to establish shared keys for devices. In Bluetooth 5.1, an enhanced pairing method called LE Secure Connections is introduced to pair Bluetooth LE device. It applies ECDH-based AKE protocols for key generation in the following four association models: Numeric Comparison, Just Works, Passkey Entry and Out of Band (OOB), among which the Numeric Comparison association, denoted as Bluetooth Display hereinafter, is designed for scenarios in which two devices are capable of displaying a 6-digit number and allowing user to enter ‘yes’ or ‘no’ [20]. The display is a type of low bandwidth OOB channel. If a third party is launching a man-in-the-middle attack, the two parties would not display the same number, thereby preventing such attack.

- *IEEE 802.15.6*

IEEE 802.15.6 is the latest standard supporting communication in WBAN. Several security association protocols in this standard are based on ECDH key exchange protocol using P-256 elliptic curve for authenticated shared keys generation [22]. These protocols include the Public key hidden association, Password authenticated association and Display authenticated association, among which the display authenticated association, denoted as IEEE Display in this paper, requires the node and hub each to display a 5-digit decimal number [32]. Only when the user confirms that the display numbers match can the protocol proceed. Similar to the Bluetooth Display, the IEEE Display also belongs to the low bandwidth OOB channel that can thwart man-in-the-middle attacks.

We choose the Bluetooth Display and IEEE Display protocols as the benchmarks for our protocol since the two establish low-bandwidth OOB channels for authentication and are suitable for sensors connected with a display such as wrist heart rate monitors.

2.2 Leakage-resilient cryptography

2.2.1 Leakage-resilient models

In this section, we generalize previous work on leakage-resilient models against side-channel attacks. The early works on side-channel attacks countermeasures focus on specific side-channel attacks such as the secret-sharing schemes against power-analysis attacks [33, 34]. The first model protecting against side-channel attacks on a general level was proposed by Micali and Reyzin [35] based on the *Only Computation Leaks Information* assumption. However, some later-proposed attacks such as the “cold-boot” attack [36] indicate that information can also be leaked from the memory when no computation is taking place. Akavia et al. [37] thereby introduced the relative leakage model (RLM), which assumes that adversaries can reveal a bounded amount of the secret key in a leakage attack. Dziembowski [38] and Di Crescenzo et al. [39] generalized the relative leakage model as the bounded retrieval model (BRM), in which there is an independent leakage parameter λ on the overall leakage. The BRM requires the key size must be larger than the leakage parameter to maintain security. Another stronger leakage model, Auxiliary Input Memory Leakage Model, was formulated by Dodis et al. [40] shortly after the formalization of the RLM. In this model, the leakage is unbounded in length, but it is required to be computationally hard to find the secret key from the leakage. In addition, Dodis et al. [41] and Brakerski et al. [42] considered the continuous leakage of information from all parts of the memory instead of the previous

assumption that only the memory involved in computation leaks information and therefore introduced the continual leakage model which allows adversaries to reveal a bounded leakage of the secret key but processes an unbounded property of the total leakage. To achieve the security goal, the secret key is required to be updated periodically while the public key is remained unchanged.

However, the aforementioned models only assume that leakage happens before the challenge/test session, which results in Halevi and Lin's study [43] in the attack model where leakage happens after the challenge/test session (a.k.a. after-the-fact leakage model). Afterwards, Alawatugoda et al. [44, 45] presented the after-the-fact leakage e-CK model for AKE protocols.

2.2.2. LR-AKE protocols

The authenticated key exchange (AKE) is an important cryptographic primitive. Traditional AKE security model encompasses Bellare-Rogaway (B-R) model [46], Canetti-Krawczyk (CK) model [47] and e-CK model [48], which is applied in many widely-used AKE protocols to prove security. However, the models mentioned above cannot ward off side-channel attacks.

To protect against side-channel attacks, a number of leakage-resilient AKE (LR-AKE) protocols based on different leakage-resilient models are proposed in recent years. Since 2009, there have been a number of LR-AKE schemes constructed in the BRM model, including the authenticated key agreement schemes, identification schemes, signature schemes and public key encryption schemes proposed in [23, 25]. In 2013, Yang et al. [26] proposed an LR-AKE protocol based on the auxiliary input model by employing a digital signature scheme. Followed by the proposal of the continual memory leakage model, Alawatugoda et al. [28] presented the first LR-AKE protocol in the continual leakage model by combining the after-the-fact leakage e-CK model [44] with continual leakage feature. Inspired by Alawatugoda's work, Wu et al. proposed an enhanced LR-AKE protocol [24] by employing the multiplicative blinding technique, and later propose an identity-based LR-AKE protocol under the continual leakage model [27].

2.3. Summary

We summarize the ECDH-based AKE protocols in international communication standards. Among these protocols, the Bluetooth Display and IEEE Display protocols are chosen as benchmarks of our protocol and their properties and limitations are listed in Table 1. The aforementioned LR-AKE protocols are compared in Table 2 as well.

According to the tables, we find that:

- 1) Although the Bluetooth Display and IEEE Display protocols can prevent men-in-the-middle attacks, neither do they alleviate the computational burden on the sensor nodes nor take the side-channel attacks into consideration.
- 2) The prior LR-AKE schemes in the related work either lack concert constructions or do not provide ECDH-based designs, which might be the reason that most of them are not approved in communication and security standards mentioned in Section 2.2.

Table 1. Properties and limitations of Bluetooth Display and IEEE Display protocols

Protocol	Men-in-the-middle attacks	Side-channel attacks	Reduce computation costs on sensor nodes	Additional requirement
----------	---------------------------	----------------------	--	------------------------

Bluetooth Display	+	-	-	Display on both parties
IEEE Display	+	-	-	Display on both parties

Table 2. Comparison of LR-AEK protocols in related work

LRL-AKE protocol	AKE model	Leakage model	cryptographic primitive	Implementation details
Protocol in [23, 25].	CK	Bounded retrieval	Diffie-Hellman	-
Protocol in [26]	CK	Auxiliary input memory	Diffie-Hellman	+
Protocol in [28]	e-CK	Continual after-the-fact leakage	Diffie-Hellman	+
Protocol in [24]	e-CK	Continual	Diffie-Hellman	+
Protocol in [27]	e-CK	Continual	Diffie-Hellman	+

3. PRELIMINARIES AND NOTATIONAL CONVENTIONS

This section introduces the underlying cryptographic knowledge related to our work. The notations used in this paper are listed in Table 3.

Table 3. Summary of Notations

Notation	Meaning
\times	The scalar multiplication
$+$	The point addition
\mathbb{Z}_p	The prime finite field
SK	The secret key (an integer)
PK	The public key (a pair of x-coordinate and y-coordinate values)
$x \xleftarrow{R} \mathbb{Z}$	x is picked uniformly and randomly from the set \mathbb{Z}
$ $	The concatenation of bit strings
MAC	The message authentication code
MAC_{16}	The MAC algorithm with 16-digit outputs
$F(\cdot)$	One-way hash function

3.1 Elliptic Curve Cryptography

3.1.1 Definition

An elliptic curve defined over the field Z_p ($p > 3$) is the set of solutions to the equation:

$$y^2 = x^3 + ax + b \mod p \quad \text{with } a, b \in Z_p, 4a^3 + 27b^2 \neq 0$$

where Z_p is the Galois Field or prime finite field with the prime modulus p .

3.1.2. Group Operations on Elliptic Curves

- **Point Addition ($P+Q$).** Let P and Q be two distinctive points on the elliptic curve E . The result of the point addition between P and Q is another point on E .
- **Point Doubling ($P+P$ / $2 \times P$).** Let P be a point on the elliptic curve E . Point doubling is denoted as $P+P$ or $2P$ and the result is also a point on E .
- **Scalar Multiplication ($n \times P$).** Scalar Multiplication is derived from **Point Addition** and **Doubling**. Let n be an integer and P be a point on the elliptic curve E . Then the scalar multiplication between n and P is defined by:

$$n \times P = \underbrace{P + P + \dots + P}_n$$

When n is a large integer, computation of the scalar multiplication becomes significantly more time-consuming than that of a point addition.

3.1.3. ECC in FIPS Standard

The Federal Information Processing Standards (FIPS) issued by The National Institute of Standards and Technology (NIST) [49] selects the following equation of elliptic curves over prime field for cryptographic usage:

$$y^2 = x^3 - 3x + b \mod p$$

Some other domain parameters are also given to determine a specific curve:

- The prime modulus p ;
- The coefficient b ;
- The order of the base point n ;
- The base point $G = (G_x, G_y)$;

Based on the standard, the NIST specifies five elliptic curves, which are listed along with their prime moduli in Table 4. More information about these curves and their corresponding domain parameters are given in [49]

Table 4. NIST curves

Curve name	Prime modulus p
P-192	$2^{192} - 2^{64} - 1$
P-224	$2^{224} - 2^{96} + 1$
P-256	$2^{256} - 2^{224} + 2^{192} + 2^{96} - 1$
P-384	$2^{384} - 2^{128} - 2^{96} + 2^{32} - 1$
P-521	$2^{521} - 1$

3.1.4. Difficult Problems

The elliptic curve discrete logarithm problem (ECDLP) and elliptic curve Diffie-Hellman problem (ECDHP) are two difficult problems of the elliptic curve. Let E be an elliptic curve defined over Z_p , G be the base point of E , and A, B and C be the points on E such that $A = a \times G$, $B = b \times G$ and $C = c \times G$, where the unknown numbers $a, b, c \in [0, n-1]$. The two problems are described as follows:

- Elliptic Curve Discrete Logarithm Problem (ECDLP): Given the points G and A on E , the ECDLP is to find the integer a , if exist, such that $A = a \times G$.
- Elliptic Curve Diffie-Hellman Problem (ECDHP): Given the points G, A, B and C , the ECDHP is to compute the point C such that $C = ab \times G$.

Due to these difficult problems, it is computationally difficult and time-consuming for adversaries to obtain the secret key using mathematical approaches.

3.2 Unbalanced ECDH

In practice, two devices between a WBAN (i.e. a sensor and a coordinator) have different computation capabilities. The limited device should undertake less computation burdens when implementing cryptosystems due to its less computational power and poor power supply. A typical solution is to transfer computation tasks from the sensor node to the server during key establishment process. In previous work, such design is utilized in Diffie-Hellman AKE protocol to transfer one exponentiation from the sensor to the server terminal. Zhang et al. [50, 51, 52, 53, 54] applied the similar design to ECDH protocol and proposed the unbalanced ECDH (UECDH) scheme. Compared to traditional ECDH protocol, the UECDH scheme transfers one scalar multiplication from the limited device A to the powerful device B in the key establishment process, which reduces the computation costs on A (as shown in Table 5.).

Table 5. Computation costs on A and B

AKE protocol	Number of scalar multiplications on	
	A	B
ECDH	2	2
UECDH	1	3

3.3 Continual Leakage Model

The LRL-AKE protocol is constructed in the continual leakage model. This model generalizes the settings in which all the parameters of the scheme are bounded while the total leakage shows an unbounded property [55, 27, 24]. The properties of the continual leakage model are briefly introduced in this section:

- *Only computation leakage*: The information of permanent/ephemeral key leaks to side-channel adversaries only when computation occurs. Other parts of memory not involved in computation are not subject to leakage during a computation round.
- *Bounded leakage in single computation round*: The amount of leakage in a single computation round is bounded to a security parameter, say λ . This means an adversary can only reveal λ bits of information in each time period.
- *Independence between computation rounds*: The leakage information in one computation

round is independent with that in other rounds.

- *Overall unbounded leakage*: After executing a computation round, the secret key is refreshed by the system. Thus, the total leakage is unbounded.

3.4 Blinding Technique

To satisfy properties of the continual leakage model, the blinding scheme [29, 24, 27] is applied in the LRL-AKE protocol. The Blinding technique is one of the countermeasures against side-channel attacks which update or refresh the permanent secret key after each computation round, without changing the associated public key. It comprises the following two algorithms:

- *Encode(SK)*: This algorithm takes the secret key SK as the parameter. It generates a random number $r_0 \xleftarrow{R} Z_p$ and splits SK into two components $L_0 = r_0$ and $R_0 = SK - r_0$. After partition, it eliminates SK and stores L_0 and R_0 in different parts of the memory. The **encode** algorithm is called at the beginning of the first computation round.
- *Refresh(L_{i-1}, R_{i-1})*: This algorithm takes two parameters L_{i-1} and R_{i-1} . It generates $r_i \xleftarrow{R} Z_p$ and refreshes L_{i-1} and R_{i-1} into L_i and R_i , where $L_i = L_{i-1} + r_i$ and $R_i = R_{i-1} - r_i$. Then, L_{i-1} and R_{i-1} are eliminated and L_i and R_i are stored separately. The **refresh** algorithm is called at the beginning of i^{th} computation round, where $i \geq 1$.

Note that L_i , R_i and SK yield the following relationship:

$$L_i + R_i = (L_{i-1} + r_i) + (R_{i-1} - r_i) = L_{i-1} + R_{i-1} = \dots = L_0 + R_0 = SK$$

4. PROTOCOL

4.1. Overview

4.1.1. Communication Model

The communication model of the LRL-AKE protocol is specified as follows:

- *Participants*. In this protocol, there are two participants A and B . A denotes the computationally limited sensor node; while B the person server such as a personal smartphone.
- *Communication Channels*. There are two types of communication channels accessible in this protocol: normal channel and out-of-band (OOB) channels.
 - *Normal Channels*. The normal channels belong to the Dolev-Yao model [56], which indicates that the messages transmitted through these channels can be eavesdropped, blocked or modified by the attackers. Example of the normal channels includes the Internet, Wi-Fi, Bluetooth and ZigBee. The normal channels in the LRL-AKE protocol is denoted as \leftrightarrow .
 - *OOB Channels*. The OOB channels require each device to independently generate a confirmation value based on the cryptographic information. The confirmation values may be provided to an operator for authentication in the form of numbers, images, QR code and so on. Since the OOB channels are established via human interactions, the messages in these channels are difficult to be spoofed and blocked. Examples of OOB channels used in

WBAN include the Numeric Comparison association in Bluetooth LE and the display authenticated association in IEEE 802.15.6 standard, which use short strings as the confirmation value. The OOB channels in the LRL-AKE protocol is denoted as \Leftrightarrow .

4.1.2. Attack Model

The adversary's power in continual leakage model is specified in section 3.3. In addition to that, this section generalizes the abilities and limitations of an adversary in the LRL-AKE protocol.

- **Assumption 1:** The adversary cannot alter, delay and delete any message transmitted through the OOB channels.
- **Assumption 2:** The MAC algorithms are unforgeable for the adversary.
- **Basic Ability:** The adversary is able to query, alter and delete all the messages transmitted through the normal channels.
- **Stronger Ability 1:** The adversary is able to access previous session keys.
- **Stronger Ability 2:** The adversary is able to compromise the permanent secrets of A or B .
- **Stronger Ability 3:** The adversary performing side-channel attack can recover λ bits of the secret key in a single computation round, where λ is less than the size of the secret key.

4.1.3. Security Model.

Based on the above attack model, the LRL-AKE protocol aims at achieving the following security goals:

- **Key authentication.** The session key produced by the protocol should be shared only between A and B other than any other adversary with the **Basic Ability**.
- **Key confidentiality.** The session key produced by the protocol cannot be computed by any adversary with the **Basic Ability**.
- **Key integrity.** After the completed run of the protocol, the session key computed by A should be equivalent to that computed by B under the attack model of an adversary with the **Basic Ability**.
- **Key confirmation.** After a computation run of the LRL-AKE protocol, both A and B receive confirmation that other party knows the secret keys under the attack model of an adversary with the **Basic Ability**.
- **Key freshness.** The session key produced in each computation run of the protocol between A and B should be unique under the attack model of an adversary with the **Basic Ability** and the **Stronger Ability 1**.
- **Forward Secrecy.** Under the attack model of an adversary with the **Basic Ability** and the Stronger Ability 2, if the permanent secret keys of A and B are compromised, the secrecy of the previous session keys established by the private keys should not be affected.
- **Resistance to combinatorial attacks.** Under the attack model of an adversary with the **Basic Ability**, the LRL-AKE protocol that generates authentication information by short hash functions should prevent general or multiple-shot attacks providing the adversary advantage over guess.

- **Resistance to continual leakage attacks.** The protocol should be secure under the attack model of an adversary with the **Stronger Ability 3**.

4.2 Protocol Design

Leakage-resilient AKE: LR-AKE

The protocol is designed with the following four procedures:

- **Initialization procedure.** This procedure firstly generates the following parameters
 - Common and public parameters shared by parties A and B: $\text{comm} = \{E, G, Z_p\}$;
 - Values held by A:

$$SK_A \xleftarrow{R} Z_p \text{ and } PK_A = SK_A \times G;$$

- Values held by B:

$$SK_B \xleftarrow{R} Z_p \text{ and } PK_B = SK_B \times G.$$

Then, the private keys are hided using the blind storage scheme, i.e.,

$$\text{– A: } L_A \xleftarrow{R} Z_p \text{ and } R_A = SK_A - L_A;$$

$$\text{– B: } L_B \xleftarrow{R} Z_p \text{ and } R_B = SK_B - L_B;$$

- **Authentication and Key Exchange procedure.**

- A generates $\text{rand}_A \xleftarrow{R} Z_p$ and computes

$$U_{AL} = R_A + L_i,$$

$$U_A = U_{AL} + R_i,$$

$$T_A = U_A \times G,$$

$$\text{commit}_A = \text{MAC}(T_A, A || PK_A)$$

Then A sends B with $(A, PK_A, \text{commit}_A)$.

- B generates $\text{rand}_B \xleftarrow{R} Z_p$ and computes

$$U_{BL} = R_B + L_i,$$

$$U_B = U_{BL} + R_i,$$

$$T_B = U_B \times G,$$

$$\text{commit}_B = \text{MAC}(T_B, B || PK_B)$$

Then B sends A with $(B, PK_B, \text{commit}_B)$.

- Upon receiving $(B, PK_B, \text{commit}_B)$, A sends T_A to B.
- Upon receiving T_A , B firstly checks the validity of commit_A by computing $\text{MAC}(T_A, A || PK_A)$ and comparing it with the received commit_A . If the verification succeeds, B sends T_B to A and computes the shared secrets as follows:

$$K_B = rand_B \times (T_A - PK_A).$$

Finally, B computes and display the following digest:

$$digest_B = MAC_{16}(K_{Bx}, A || B || PK_A || PK_B || T_A || T_B)$$

- Upon receiving T_B , A firstly checks the validity of *commitB* by computing $MAC(T_B, B || PK_B)$ and comparing it with the received *commitB*. If the verification succeeds, A computes the shared secrets as follows:

$$K_A = rand_A \times (T_B - PK_B).$$

Finally, A computes and display the following digest:

$$digest_A = MAC_{16}(K_{Ax}, A || B || PK_A || PK_B || T_A || T_B)$$

- The human user compares the two digests displayed on the screen of A and B. If they are identical, then proceed; otherwise, quit.
- **Session Key Computation procedure.** A and B use the share secret to derive the session keys K_{ENC} and K_{MAC} which are used in encryption and message authenticate code algorithms respectively:

$$- A: K_{ENC} = F(K_{Ay}, 1), K_{MAC} = F(K_{Ay}, 2)$$

$$- B: K_{ENC} = F(K_{By}, 1), K_{MAC} = F(K_{By}, 2)$$

- **Refreshing procedure.**

$$- A: rand_1 \xleftarrow{R} Z_P, L_A \leftarrow L_A + rand_1 \text{ and } R_A \leftarrow R_A - rand_1.$$

$$- B: rand_2 \xleftarrow{R} Z_P, L_B \leftarrow L_B + rand_2 \text{ and } R_B \leftarrow R_B - rand_2.$$

Lightweight Version: LLR-AKE

Suppose that the capability of A is lower than B. The lightweight version protocol is designed by shifting one elliptic curve scalar multiplication from A to B.

- **Initialization procedure.** This procedure is the same as that of the original one.
- **Authentication and Key Exchange procedure.**

– A generates $rand_A \xleftarrow{R} Z_P$ and computes

$$U_{AL} = R_A + L_i,$$

$$U_A = U_{AL} + R_i,$$

$$commit_A = MAC(U_A, A || PK_A)$$

Then A sends B with $(A, PK_A, commit_A)$.

– B generates $rand_B \xleftarrow{R} Z_P$ and computes

$$U_{BL} = R_B + L_i,$$

$$U_B = U_{BL} + R_i,$$

$$T_B = U_B \times G,$$

$$commit_B = \text{MAC}(T_B, B || PK_B)$$

Then B sends A with $(B, PK_B, commit_B)$.

- Upon receiving $(B, PK_B, commit_B)$, A sends U_A to B .
- Upon receiving U_A , B firstly checks the validity of $commit_A$ by computing $\text{MAC}(U_A, A || PK_A)$ and comparing it with the received $commit_A$. If the verification succeeds, B sends T_B to A and computes the shared secrets as follows:

$$T_A = U_A \times G,$$

$$K_B = rand_B \times (T_A - PK_A).$$

Finally, B computes and displays the following digest:

$$digest_B = \text{MAC}_{16}(K_B, A || B || PK_A || PK_B || U_A || T_B)$$

- Upon receiving T_B , A firstly verifies $commit_B$ by computing $\text{MAC}(T_B, B || PK_B)$ and comparing it with $commit_B$. If $commit_B$ is valid, A then computes the shared secret:

$$K_A = R_A \times (T_B - PK_B)$$

Finally, A computes and displays the digest:

$$digest_A = \text{MAC}_{16}(K_A, A || B || PK_A || PK_B || U_A || T_B)$$

- The rest steps are the same as that of the original one.
- **Session Key Computation procedure.** This procedure is that same as that of the original one.
- **Refreshing procedure.** This procedure is the same as that of the original one.

In the lightweight version, B computes $T_A = U_A \times G$ one behalf of A . As a result, A computes one scalar multiplication while B computes three.

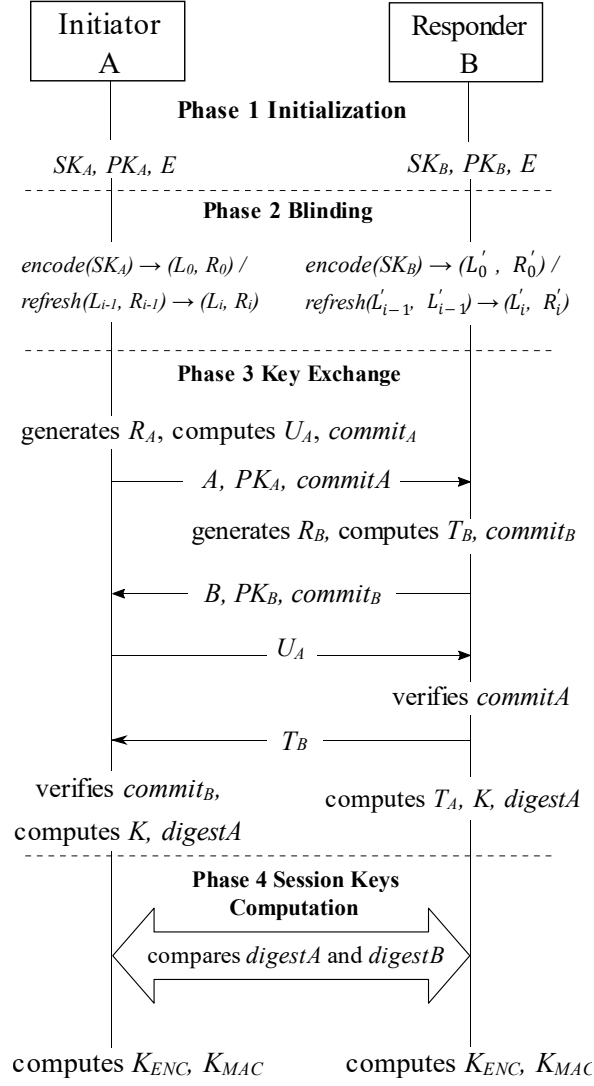


Fig. 2. LRL-AKE protocol.

4.3 Security Features

In this section, the security analysis of the LRL-AKE protocol is presented. We demonstrate that our LRL-AKE protocol remains secure under the attack models in Section 4.1.

Proposition 1 (Key authentication): *Assume that an adversary C is endowed with the Basic Ability, the secret is only shared between A and B other than any party after a completed run of the protocol.*

Proof: (1) A completed run of the protocol is marked by the equality of $digest_A$ and $digest_B$; and the prerequisite of comparing two digests is the validation of $commit_A$ and $commit_B$. Therefore, a completed run of the LRL-AKE protocol means that $commit_A$ and $commit_B$ are both valid.

(2) Since $commit_A = \text{MAC}(U_A, A || PK_A)$, the validation of $commit_A$ ensures the authenticity of U_A and PK_A . Likewise, the validation of $commit_B$ ensures the authenticity of T_B and PK_B . Therefore, a completed run of the LRL-AKE protocol guarantees the authenticity of U_A , T_B , PK_A and PK_B .

(3) Given that $K_A = R_A \times (T_B - PK_B)$, where R_A is generated by A and (T_B, PK_B) are authenticated in (2), A thereby believes that the secret key K_A is shared with B other than other party. Similarly, given that $K_B = R_B \times (U_A \times G - PK_A)$, where R_B is generated by B and (U_A, PK_A) are authenticated according to (2), B thereby believes that the secret key K_B is shared with A other than other party.

Therefore, Proposition 1 is proved correct in the LRL-AKE protocol.

Proposition 2 (Key confidentiality): *Suppose that an adversary C has the Basic Ability. After a completed run of the protocol between A and B, C cannot derive the secret keys of A and B*

Proof: (1) The shared keys of A and B can be computed in the following equations:

$$K_A = R_A \times (T_B - PK_B) \quad (1)$$

$$K_B = R_B \times (G \times U_A - PK_A) \quad (2)$$

Since $L_i + R_i = SK_A$ and $L'_i + R'_i = SK_B$, equation (1) and (2) can be converted as follows:

$$\begin{aligned} K_A &= R_A \times (T_B - PK_B) \\ &= R_A \times (G \times U_B - PK_B) \\ &= R_A \times (G \times (R_B + L'_i + R'_i) - PK_B) \\ &= R_A \times (G \times (R_B + SK_B) - PK_B) \\ &= R_B \times (G \times R_B + PK_B - PK_B) \\ &= R_A R_B \times G \end{aligned}$$

$$\begin{aligned} K_B &= R_B \times (G \times U_A - PK_A) \\ &= R_A \times (G \times (R_B + L_i + R_i) - PK_B) \\ &= R_B \times (G \times (R_A + SK_A) - PK_A) \\ &= R_B \times (G \times R_A + PK_A - PK_A) \\ &= R_B R_A \times G \end{aligned}$$

Therefore, $K_A = K_B = R_A R_B \times G$ and C should compute R_A and R_B in order to derive the shared key.

(2) Since R_A is hidden by $U_A = R_A + SK_A$ and R_B is hidden by $T_B = (R_B + SK_B) \times G$, C should firstly compute SK_A and SK_B in order to find R_A and R_B .

(3) According to the attack model, C is unable to obtain either SK_A or SK_B since they are not transmitted through the normal channel. As a result, C cannot derive R_A and R_B , and cannot derive K_A and K_B further. Therefore, the LRL-AKE is proved to support Proposition 2.

Proposition 3 (Key Integrity): *Suppose that the adversary C has the Basic Ability, A and B compute identical secrets after a completed run of the protocol.*

Proof: As mentioned in the proof of Proposition 3, the shared secrets of A and B yield the following equation:

$$K_A = K_B = R_A R_B \times G$$

Therefore, the LRL-AKE protocol can maintain key integrity even under the attack model that C has the Basic Ability.

Proposition 4 (Key confirmation): *Suppose that the adversary C has the Basic Ability, and after a completed run of the protocol, A and B can confirm that the other party knows the shared secret.*

Proof: As stated in the proof of Proposition 1, a completed run of the LRL-AKE protocol is marked by the equality of $digest_A$ and $digest_B$. Since $digest_A$ is computed by taking K_A as one of the inputs, it is evident that A knows the shared secret; similarly, $digest_B$ is computed by taking K_B as one of the inputs, which indicates that B knows the shared secret.

Consequently, after verifying the equality of $digest_A$ and $digest_B$, A and B can confirm that the

other party knows the shared secret. The LRL-AKE protocol is proved to support Proposition 4.

Proposition 5 (Key freshness): *Suppose that the adversary C has the Basic Ability and Stronger Ability 1, C cannot derive the shared secret from previous session keys.*

Proof: As mentioned in the proof of Proposition 2, the shared secret is computed by taking R_A and R_B as the inputs. Since R_A and R_B are ephemeral keys and are updated in each computation round, the shared secret is also fresh in each computation round. Therefore, proposition 5 holds true in the LRL-AKE protocol.

Proposition 6 (Forward secrecy): *Suppose that the adversary C has the Basic Ability and Stronger Ability 2, C is unable to derive the previous session keys yet.*

Proof: Due to the Stronger Ability 2, C can obtain the following long-term values in the LRL-AKE protocol:

$$(E, G, Z_P^*, A, B, PK_A, PK_B, U_A, T_B).$$

As mentioned in the proof of Proposition 2, C should know R_A and R_B in order to compute the shared key. Since R_A and R_B are ephemeral keys which C cannot obtain according to the *Basic Ability 2*. Therefore, the LRL-AKE protocol is proved to provide forward secrecy under the aforementioned attack model.

Proposition 7 (Resistance to combinatorial attacks): *Suppose that the adversary C has the Basic Ability, C is unable to decipher the short MAC algorithm (i.e. the digests in this protocol) via the combinatorial attacks.*

Proof: In the key exchange procedure of the LRL-AKE protocol, the values of $digest_A$ and $digest_B$ are determined in the first step. To compute $digest_A$ and $digest_B$, C needs to know U_A and T_B . Since the value of U_A is kept secret until step 3 and the value of T_B is kept secret until step 4, C cannot break $digest_A$ and $digest_B$ through combinatorial attacks before they are displayed in step 4. Therefore, the LRL-AKE protocol is proved to resist the combinatorial attacks under the above attack models.

Proposition 8 (Resistance to continual leakage attacks): *Suppose that the adversary C has the Stronger Ability 3, C is unable to recover the secret keys through the continual leakage attacks.*

Proof: Let $f(SK)$ denote a leakage function on the secret key SK performed by C . According to the properties of the continual leakage model mentioned in Section 3.3, the output bit-length of $f(SK)$ is bounded to λ (λ is smaller than the size of SK). By employing the blinding technique, the secret key of the system is refreshed in each round and satisfies the equality: $SK = L_{i-1} + R_{i-1} = L_i + R_i$. In addition, the leaked information of both L_i and R_i is independent of that of both L_{i-1} and R_{i-1} in previous round. Therefore, C can only recover at most λ bit of (L_{i-1}, R_{i-1}) or (L_i, R_i) in a signal run of the protocol, which prevents C obtaining the completed secret keys by repeatedly performing side-channel attacks. Consequently, the LRL-AKE protocol is proved secure under the continual leakage attacks.

5. PROTOTYPE AND PERFORMANCE

In this section, the performance of the LRL-AKE protocol is evaluated. The Bluetooth Display protocol and the IEEE Display protocol are chosen as the benchmarks. We first theoretically analyze the overall performances of our proposed protocol and the two benchmarks; and then we

construct a prototype of the LRL-AKE protocol and carry out two sets of experiment to test its performance.

5.1 Evaluation

Denote the cost of a scalar multiplication operation by \mathcal{S} and cost of computing a MAC algorithm by \mathcal{H} . To measure the computational cost, the number of scalar multiplication and MAC is counted. The overall performances of the three protocols are listed in Table 6.

Table 6. Overall Performance

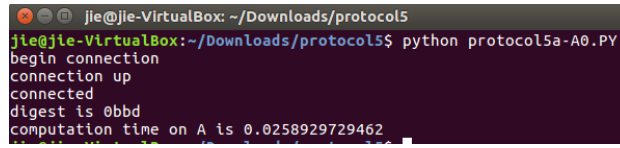
Protocol	Computational cost on		Leakage-resilient
	A	B	
LRL-AKE	$3\mathcal{H}+\mathcal{S}$	$3\mathcal{H}+3\mathcal{S}$	+
Bluetooth	$3\mathcal{H}+2\mathcal{S}$	$3\mathcal{H}+2\mathcal{S}$	-
Display			
IEEE Display	$3\mathcal{H}+2\mathcal{S}$	$3\mathcal{H}+2\mathcal{S}$	-

As shown in Table 6, the LRL-AKE protocol provides the lowest computational cost on the limited device A and possesses leakage-resilient property. Therefore, the overall performance of the LRL-AKE protocol is better than that of the Bluetooth Display protocol and the IEEE Display protocol.

5.2 Prototype and implementation details

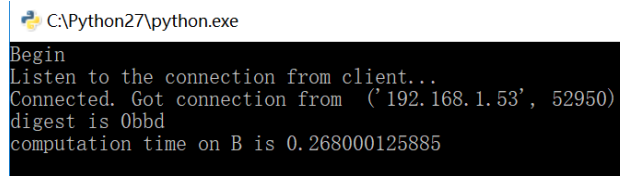
1) Setup

The prototypes of the LRL-AKE protocol, Bluetooth Display protocol and IEEE Display protocol are built in Python programming language, in which the MAC algorithm is realized by the HMAC in conjunction with SHA-256 and the network communication is implemented through the socket connection based on the Transmission Control Protocol (TCP). Fig. 3 and Fig. 4 respectively show the output results of the prototype on A and B .



```
jle@jle-VirtualBox: ~/Downloads/protocol5
jle@jle-VirtualBox:~/Downloads/protocol5$ python protocol5a-A0.PY
begin connection
connection up
connected
digest is 0bbd
computation time on A is 0.0258929729462
```

Fig. 3. The output display of the prototype on A , in which the digest and computational time would be displayed when connected successfully.



```
C:\Python27\python.exe
Begin
Listen to the connection from client...
Connected. Got connection from ('192.168.1.53', 52950)
digest is 0bbd
computation time on B is 0.268000125885
```

Fig. 4. The output display of the prototype on B , in which the digest, IP address of the initiator and computational time would be displayed when connected successfully.

In order to test the above prototypes, two sets of experiment are carried out. In Experiment I, the prototypes of the three protocols are implemented between two identical virtual machines; in

Experiment II, each prototype is executed between a Raspberry Pi and a laptop.

Table 7. Environment of Experiment I

Party	Operating System	Memory	Storage
A	Ubuntu 64-bit	5010 MB	10 GB
B	Ubuntu 64-bit	5010 MB	10 GB

Table 8. Environment of Experiment II

Device	CPU	Memory	Hard Disk
Raspberry Pi (A)	BCM2837 x64 1.2GHz (4 cores)	0.91 GB	32 GB
Laptop (B)	Core i5-7200U x64 2.50GHz (2 cores)	7.88 GB	237 GB



Left to right:



Fig. 5. Hardware platform of Experiment II.

2) Experiment I

In Experiment I, both the initiator A and responder B are deployed on two virtual machines with the same configurations to verify whether the LRL-AKE protocol satisfies the requirement of unbalance computation. The detailed information of the two virtual machines is listed in Table 7.

The prototype of the LRL-AKE protocol was run ten times respectively on the five FIPS-recommended elliptic curves, i.e. P-194, P-224, P-256, P-384 and P-521, and the prototypes of the Bluetooth Display and IEEE Display protocols were also run ten times on P-256 in order to find the average computational time and the average increment of CPU usage on A and B during execution. The computational time of the LRL-AKE protocol on the five elliptic curves are illustrated in Fig. 6 and those of the three protocols on P-256 are illustrated in Fig. 7. While the average increments of CPU usage of the LRL-AKE protocol on the five elliptic curves are illustrated in Fig. 8, and those of the three protocols on P-256 are illustrated in Fig. 9.

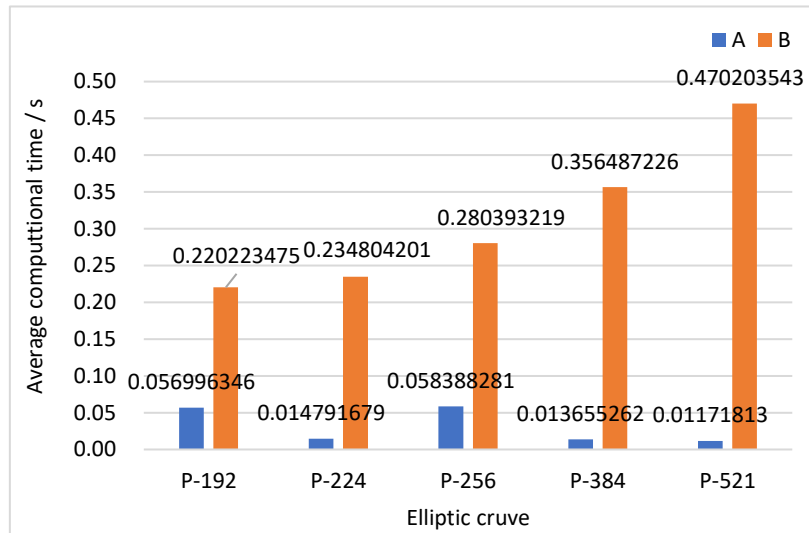


Fig. 6. Average computational time on A and B in the LRL-AKE protocol.

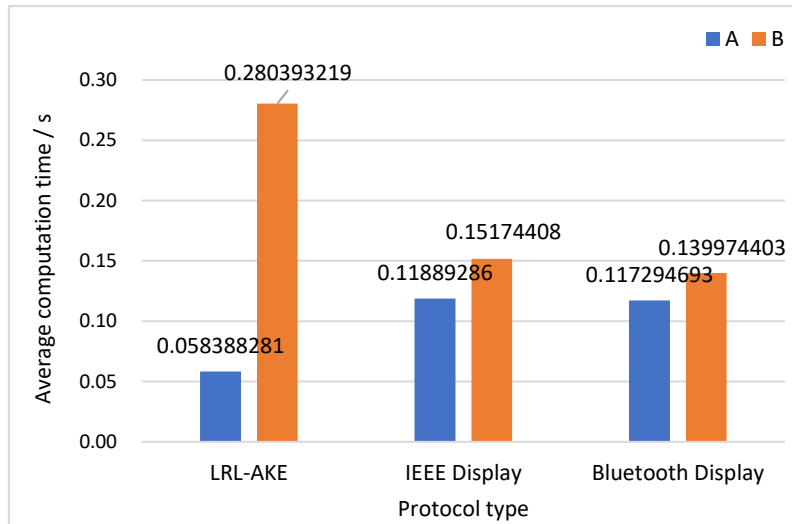


Fig. 7. Average computational time for the LRL-AKE protocol, Bluetooth Display protocol and IEEE Display protocol on P-256

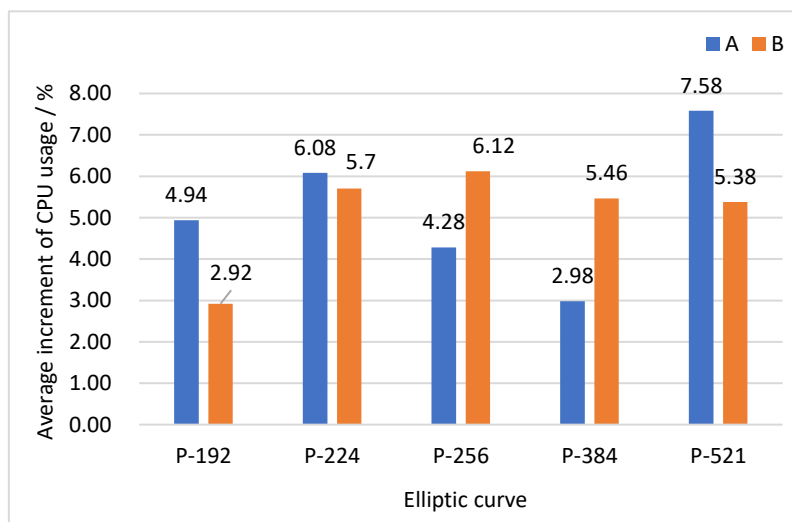


Fig. 8. Average increment of CPU usage on A and B in the LRL-AKE protocol.

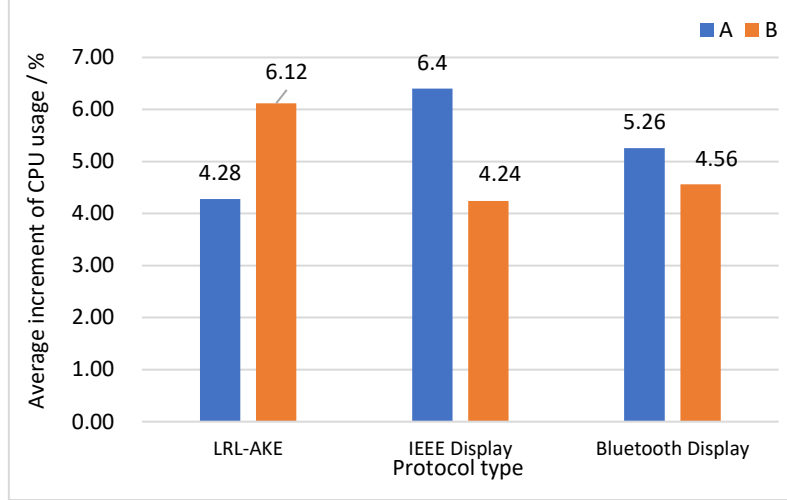


Fig. 9. Average increment of CPU usage in the LRL-AKE protocol, Bluetooth Display protocol and IEEE Display protocol on P-256

The results of Experiment I are discussed as follows:

According to Fig. 6, in the LLK-AKE protocol, the average computation time on A is less than that on B for all of the five curves. Note that the code number for the curves increases from left to right, and larger code numbers correspond to larger EC domain parameters. As the digits of the domain parameters increase, the computational time on B also increases remarkably, whereas the computational time on A stays at low levels.

Fig. 7 reveals that the average computational time on A in the LRL-AKE protocol is much less than those in the Bluetooth Display and IEEE Display protocols, which coincides with the evaluation in Table 6.

As shown in Fig. 8 and 9, the differences in CPU usage between A and B under the LRL-AKE protocol are not significant, which deviates from our expectation. We suppose this is because the computation powers of the two virtual machines are close and relatively low. Hence, even one scalar multiplication might cause significant computation burdens on both devices.

3) Experiment II

In Experiment II, the initiator A and the responder B are deployed respectively on a Raspberry Pi and a laptop in order to simulate two nodes with unbalanced computation powers in a WBAN and evaluate the extra burden on the computationally limited device. It is obvious that the laptop is significantly more powerful than the Raspberry Pi. The detailed information of the two devices is displayed in Table 8 and the hardware platform is demonstrated in Fig. 5.

Before running the prototypes, the file size of each prototype is measured, and results are listed in Table 9. The prototype of the LRL-AKE protocol was run on the five FIPS-recommended elliptic curves ten times and the prototypes of the other two benchmarks were also run on P-256 ten times. During each computation round, the computational time, increment of CPU usage on A and B , and the temperature variation on the Raspberry Pi's CPU are measured in order to obtain the average computational time, the average increment of CPU usage and the average temperature variation in each protocol. The average computational time and the average increment of CPU

usage of the LRL-AKE protocol on the five curves are respectively displayed in Fig. 10 and Fig. 11. While the average computational time and the average increment of CPU usage of the LRL-AKE protocol and the two benchmarks on P-256 are respectively displayed in Fig. 12 and Fig. 13, and the results of temperature variation are shown in Fig. 14. Moreover, the representative infrared images reflecting the CPU temperature of the initiator before (*A*) and during (*B*) execution of LRL-AKE protocol are given in Fig. 15.

Table 9. File size comparison

Protocol	File size on <i>A</i>	File size on <i>B</i>
LRL-AKE	9.3 KB + Blind storage	9.7 KB + Blind storage
IEEE Display	3.8 KB	3.8 KB
Bluetooth Display	3.0 KB	2.8 KB

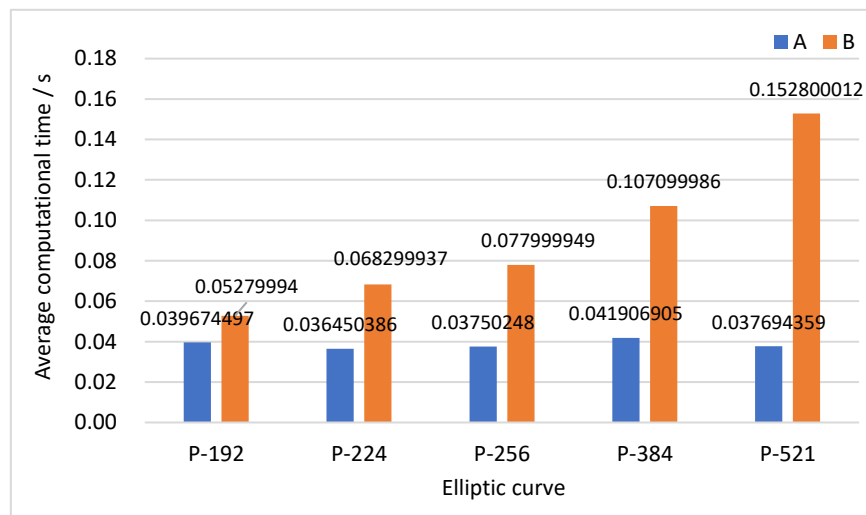


Fig. 10. Average computational time on *A* and *B* in the LRL-AKE protocol

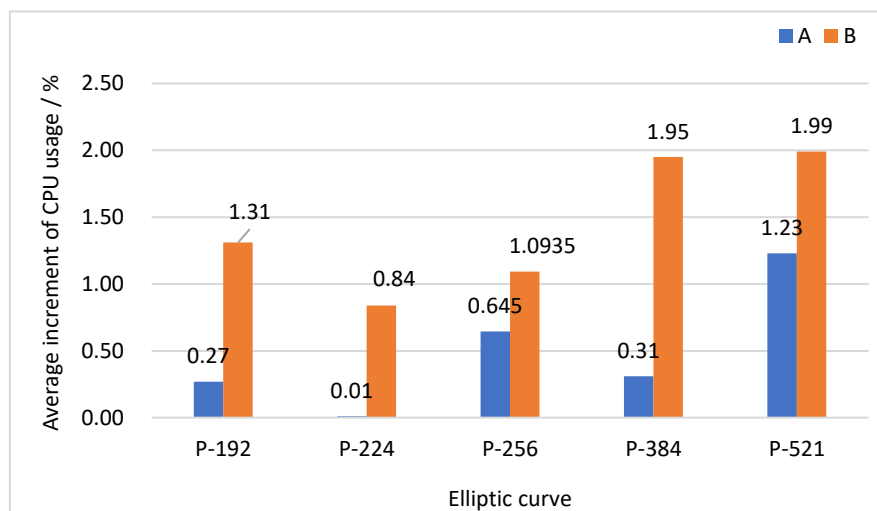


Fig. 11. Average increment of CPU usage on *A* and *B* in the LRL-AKE protocol

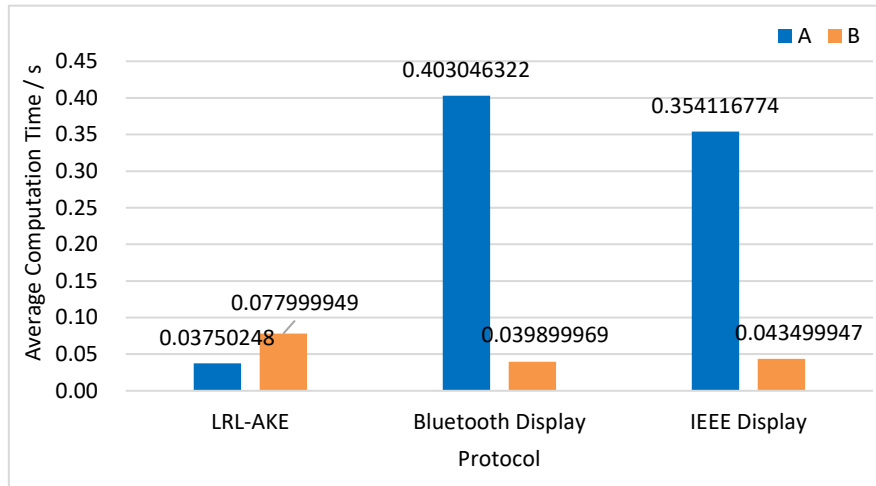


Fig. 12. Average computational time of the LRL-AKE protocol, Bluetooth Display protocol and IEEE Display protocol on P-256

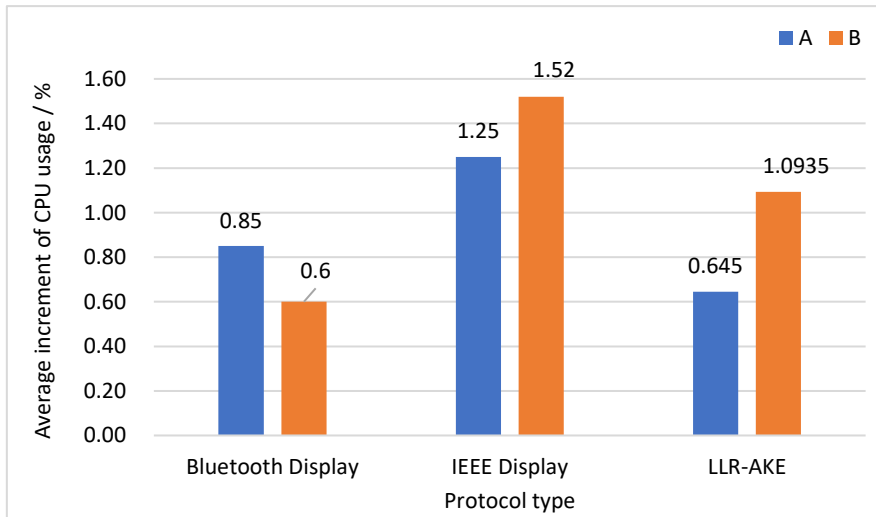


Fig. 13. Average increment of CPU usage in the LRL-AKE protocol, Bluetooth Display protocol and IEEE Display protocol on P-256

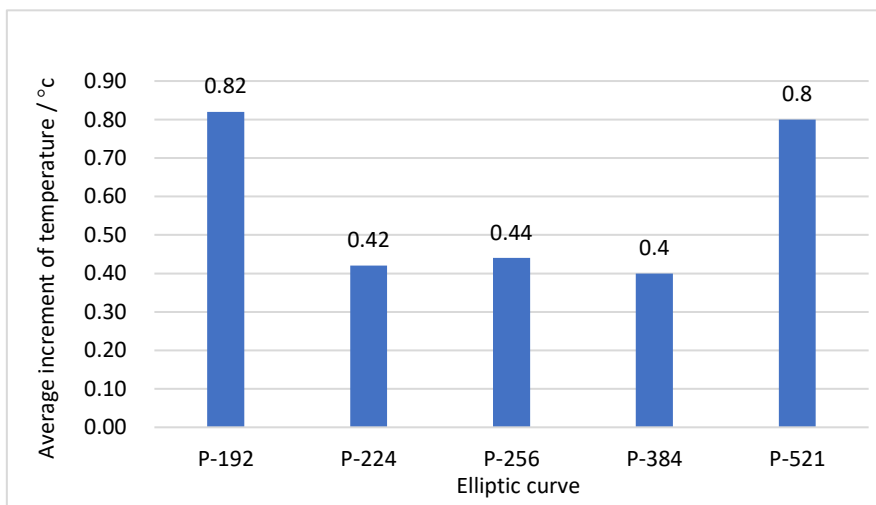


Fig. 14. Average temperature increment on A in the LRL-AKE protocol

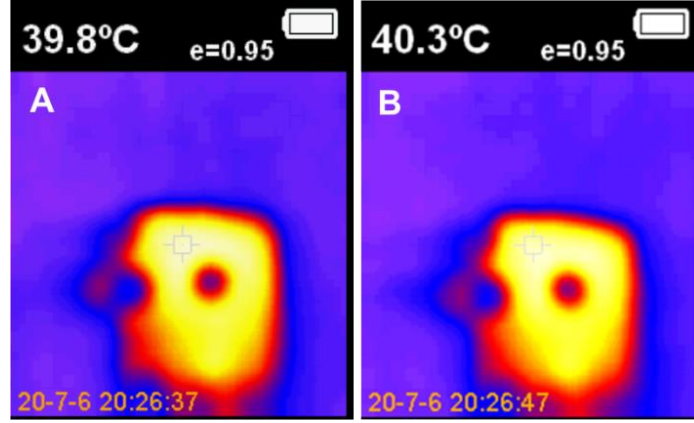


Fig. 15. Representative infrared thermal images reflecting the CPU temperature of the initiator B before (A) and during (B) execution of LRL-AKE protocol

The results of Experiment II are discussed as follows:

Table 9 compares the file size of the three protocols. We can find that the file size of the prototype of LRL-AKE protocol is larger than that of the two other protocols. Although the larger file size means more storage space, but that for the LRL-AKE protocol is less than 10 KB and is still acceptable for a limited device. In addition, the LRL-AKE prototype also needs extra space to store updated secret keys (a.k.a. blind storage) due to the application of blinding technique, but these secret keys only contain a few hundred bits of characters, and the extra space can therefore be neglected.

Fig. 10 reveals that A consumes less computational time than B when running the LRL-AKE protocol. The result coincides with the evaluation in Table 6. Moreover, with the increasing domain parameters, the computational time of A increases steadily, while that of B barely increases. This clearly indicates that A can take over B 's computation and keep B 's computational time at a low value.

The average increment of CPU usage on A and B in the LRL-AKE protocol is presented in Fig. 11, which shows that the CPU usage on A is less than that on B in the LRL-AKE protocol. Compared with the results in Fig. 8, the CPU usage on A is significantly reduced when the difference of computation power between A and B becomes larger.

As shown in Fig. 12, the average computational time for A and B in LRL-AKE protocol on P-256 are much shorter than that for A and B in Bluetooth Display protocol and IEEE Display protocol. More importantly, in LRL-AKE protocol, the computational time for A is even shorter than that for B , in great contrast to the two benchmarks in which computational time of A is much longer than B . In addition, the CPU usage for A in LRL-AKE protocol is found lower than that in the other two protocols, as demonstrated in Fig. 13. These experiments all prove that the LRL-AKE protocol is friendlier to a computationally limited device.

To further verify the advantage of LRL-AKE protocol, the CPU temperature for A before and during execution of LRL-AKE protocol were also measured using an infrared thermometer/imager; and the recorded temperature increment and the representative infrared thermal images are given in Figure 5.2.11 and 5.2.12, respectively. It is found that the temperature increments for A are all less than 1.0 °C. Clearly, running the LRL-AKE protocol would not cause the limited device (B)

overheating.

6. DISCUSSION

As shown in Fig. 2, except the blinding technique and unbalanced computation scheme, the LRL-AKE protocol is constructed by incorporating the initialization, key exchange and session key computation procedures, which are also the basis of the Bluetooth display and IEEE display protocols [20] [22]. Therefore, it allows full reuse of the existing protocols and reduces complexity of modification.

In addition, the protocol proposed herein do not alter the display authentication via low-bandwidth OOB channels in the above two protocols. That is to say, the user can still authenticate by comparing short digests. Hence, our new protocol has a strong interoperability and consistency with the existing protocols and would not cause any difficulties to users accustomed to the existing protocols.

7. CONCLUSION

A leakage-resilient and lightweight authenticated key exchange protocol (LRL-AKE) for capability-constrained devices in WBAN was constructed and tested. By adopting the blinding technique, the LRL-AKE protocol could resist continual leakage attacks and provided stronger security than similar protocols in some international communicating standards. A prototype of this protocol was installed to evaluate the performance of the protocol. The results show that, this protocol allows the limited device to transfer its computational burden to the powerful device; and during the tests, the limited device exhibited short computational time and low increment of CPU usage, which is in great contrast to those of the two benchmarks.

In future, we plan to carry out more experiments using several popular types of medical sensors to evaluate the performance of LRR-AKE. In addition, we will utilize our proposed approach to improve some widely used AKE protocols against side-channels attacks.

REFERENCE

- [1] S. Ullah, H. Higgins, B. Braem, B. Latre, C. Blondia, I. Moerman and S. Saleem, A Comprehensive Survey of Wireless Body Area Networks on PHY, MAC, and Network Layers Solutions, *Journal of Medical Systems*, 36 (3) (2010) 1065-1094. <https://doi.10.1007/s10916-010-9571-3>.
- [2] M. Salayma, A. Y. Al-Dubai, I. Romdhani and Y. Nasser, Wireless Body Area Network (WBAN): A Survey on Reliability, Fault Tolerance, and Technologies Coexistence, *ACM Computing Surveys*, 50 (1) (2017) 1-38.
- [3] F. Wu, X. Li, A.K. Sangaiah, L. Xu, S. Kumari, L. Wu, J. Shen, A lightweight and robust two-factor authentication scheme for personalized healthcare systems using wireless medical sensor networks. *Future Generation Computer Systems*, 82 (2018) 727-737. <https://doi.10.1016/j.future.2017.08.042>.
- [4] P. Raković and B. Lutovac, A Cloud Computing Architecture with Wireless Body Area Network for Professional Athletes Health Monitoring in Sports Organizations – case study of Montenegro, in: 4th Mediterranean Conference on Embedded Computing, 2015, Budva.

- [5] S. Jakub, et al. FlexiGuard: Modular Biotelemetry System for Military Applications, in: International Conference on Military Applications, 2015, Brno. <https://doi.org/10.1109/miltechs.2015.7153712>.
- [6] J. Liu, X. Huang, J. K. Liu, Secure sharing of Personal Health Records in cloud computing: Ciphertext-Policy Attribute-Based Signcryption, *Future Generation Computer Sciences*, 52 (2015) 67-76. <https://doi.org/10.1016/j.future.2014.10.014>.
- [7] A. DSouza, H. Viittala and M. Hämäläinen, Performance Comparison Between ETSI SmartBAN and Bluetooth, in: 2018 12th International Symposium on Medical Information and Communication Technology (ISMICT), 2018, Sydney. <https://doi.org/10.1109/ismict.2018.8573708>.
- [8] A. C. Wong, et al. A 1 V 5 mA Multimode IEEE 802.15.6/Bluetooth Low-Energy WBAN Transceiver for Biotelemetry Applications, *IEEE Journal of Solid-State Circuits*, 48 (1) (2013) 186-198. <https://doi.org/10.1109/jssc.2012.2221215>.
- [9] P. Baronti, P. Pillai, V. W. Chook, S. Chessa, A. Gotta and Y. F. Hu, Wireless sensor networks: A survey on the state of the art and the 802.15.4 and ZigBee standards, *Computer Communications*, 30 (7) (2007) 1655-1695. <https://doi.org/10.1016/j.comcom.2006.12.020>.
- [10] M. Ambigavathi and D. Sridharan, Energy efficient and load balanced priority queue algorithm for Wireless Body Area Network, *Future Generation Computer Systems*, 88 (2018) 586–593. <https://doi.org/10.1016/j.future.2018.05.044>.
- [11] Z. Yu, M. H. Au, Q. Xu, R. Yang, J. Han, Towards leakage-resilient fine-grained access control in fog computing, *Future Generation Computer Systems*, 78 (2) (2018) 763-777. <https://doi.org/10.1016/j.future.2017.01.025>.
- [12] P. C. Kocher, Timing Attacks on Implementations of Diffie-Hellman, RSA, DSS, and Other Systems, *Lecture Notes in Computer Science*, (1996) 104-113. https://doi.org/10.1007/3-540-68697-5_9.
- [13] P. Kocher, J. Joshua and B. Jun, Differential Power Analysis, *Lecture Notes in Computer Science*, (1999) 388-397. https://doi.org/10.1007/3-540-48405-1_25.
- [14] D. Genkin, L. Pachmanov and I. Pipman, ECDH Key-Extraction via Low-Bandwidth Electromagnetic Attacks on PCs, *Lecture Notes in Computer Science*, (2016) 219-235. https://doi.org/10.1007/978-3-319-29485-8_13.
- [15] D. Agrawal, B. Archambeault, J. R. Rao and P. Rohatgi, The EM side-channel(s), *Lecture Notes in Computer Science*, (2003) 29–45. https://doi.org/10.1007/3-540-36400-5_4.
- [16] P. Gu, D. Stow, E. Kursun, Y. Xie and R. Barnes, Thermal-aware 3D design for side-channel information leakage, in: IEEE 34th International Conference on Computer Design (ICCD), 2016, Phoenix. <https://doi.org/10.1109/iccd.2016.7753336>.
- [17] M. Ma, D. He, H. Wang, N. Kumar and K.-K. R. Choo, An Efficient and Provably-Secure Authenticated Key Agreement Protocol for Fog-Based Vehicular Ad-Hoc Networks. *IEEE Internet of Things Journal*, (2019) 1–1. <https://doi.org/10.1109/IIOT.2019.2902840>.
- [18] S. Challa, A. K. Das, V. Odelu, N. Kumar, S. Kumari, M. K. Khan and V. A. Vasilakos, An efficient ECC-based provably secure three-factor user authentication and key agreement protocol for wireless healthcare sensor networks, *Computers & Electrical Engineering*, 69 (2018) 534-554. <https://doi.org/10.1016/j.compeleceng.2017.08.003>.
- [19] A. Ostad-Sharif, H. Arshad, M. Nikooghadam and D. Abbasinezhad-Mood, Three party secure data transmission in IoT networks through design of a lightweight authenticated key agreement scheme, *Future Generation Computer Systems*, 100 (2019) 882-892.

- <https://doi.10.1016/j.future.2019.04.019>.
- [20] Bluetooth SIG Proprietary, Bluetooth Core Specification v5.1. <https://www.bluetooth.com/specifications/adopted-specifications>, 2019 (Accessed 15 July 2019).
 - [21] J. Zhang, S. Rajendran, Z. Sun, R. Woods and L. Hanzo, Physical Layer Security for the Internet of Things: Authentication and Key Generation, IEEE Wireless Communications, (2019) 1-7. <https://doi.10.1109/mwc.2019.1800455>.
 - [22] IEEE Computer Society, IEEE Standard for Local and metropolitan area networks— Part 15.6: Wireless Body Area Networks. <https://ieeexplore.ieee.org/document/6161600>, 2012 (Accessed 15 July 2019).
 - [23] J. Alwen, Y. Dodis and D. Wichs, Leakage-resilient public-key cryptography in the bounded-retrieval model, Advances in Cryptology— CRYPTO, 5677 (2009) 36-54. <https://doi.10.1007/978-3-642-03356-8>.
 - [24] J.-D. Wu, Y.-M. Tseng and S.-S. Huang, Efficient Leakage-Resilient Authenticated Key Agreement Protocol in the Continual Leakage eCK Model, IEEE Access, 6 (2018) 17130-17142. <https://doi.10.1109/access.2018.2799298>.
 - [25] Y. Dodis, K. Haralambiev, A. Lopez-Alt and D. Wichs, Efficient public-key cryptography in the presence of key leakage, Advances in Cryptology—ASIACRYPT, 6477 (2010) 613-631. <https://doi.10.1007/978-3-642-17373-8>.
 - [26] G. Yang, Y. Mu, W. Susilo and D. S. Wong, Leakage resilient authenticated key exchange secure in the auxiliary input model, Proc.Int.Conf. Inf. Secur. Pract. Exper. (ISPEC), (2013) 204-217. https://doi.10.1007/978-3-642-38033-4_15.
 - [27] J.-D. Wu, Y.-M. Tseng and S.-S. Huang, An Identity-Based Authenticated Key Exchange Protocol Resilient to Continuous Key Leakage, IEEE Systems Journal, (2019) 1-12. <https://doi.10.1109/jsyst.2019.2896132>.
 - [28] A. Janaka, S. Douglas and B. Colin, Continuous after-the-fact leakage-resilient eCK-secure key exchange, in: Proc. IMA Int. Conf. Cryptogr. Coding, 2015, Oxford. <https://doi.10.1007/978-3-319-27239-9>.
 - [29] E. Kiltz and K. Pietrzak, Leakage Resilient ElGamal Encryption, Lecture Notes in Computer Science, 6477 (2010) 634-646. <https://doi.10.1007/978-3-642-17373-8>.
 - [30] K. Neal, Elliptic curve cryptosystems, Mathematics of Computation, 48 (177) (1987) 203-209. <https://doi.10.2307/2007884>.
 - [31] K. Mahmood, S. A. Chaudhry, H. Naqvi, S. Kumari, X. Li, A. K. Sangaiah, An elliptic curve cryptography based lightweight authentication scheme for smart grid communication. Future Generation Computer Systems, 81 (2018) 557–565. <https://doi.10.1016/j.future.2017.05.002>.
 - [32] J.-M. Ho, Display Authenticated Security Association. US Patent, US 8,644,515 B2, (2014).
 - [33] S. Chari, C. S. Jutla, J. R. Rao and P. Rohatgi, Towards sound approaches to counteract power-analysis attacks, Lecture Notes in Computer Science, (1999) 398-412, 1999. https://doi.10.1007/3-540-48405-1_26.
 - [34] C. Clavier and M. Joye, Universal Exponentiation Algorithm A First Step towards Provable SPA-Resistance, Lecture Notes in Computer Science, (2001) 300-308. https://doi.10.1007/3-540-44709-1_25.
 - [35] S. Micali and L. Reyzin, Physically Observable Cryptography, Lecture Notes in Computer Science, 2951 (2004) 278-296. <https://doi.10.1007/b95566>.
 - [36] J. A. Halderman, et al. Lest We Remember: Cold-Boot Attacks on Encryption Keys,

Communications of the ACM, (2009).

- [37] J. Alwen, Y. Dodis and D. Wichs, Leakage-resilient public-key cryptography in the bounded-retrieval model, Lecture Notes in Computer Science, 5677, (2009) 36-54. <https://doi.10.1007/978-3-642-03356-8>.
- [38] S. Dziembowski, Intrusion-resilience via the bounded-storage model, Lecture Notes in Computer Science, 3876, (2006) 207-224.
- [39] S. Dziembowski, K. Pietrzak and D. Wichs, Non-malleable codes, ICS 2010: 1st Innovations in Computer Science, (2010) 434-452.
- [40] Y. Dodis, Y. T. Kalai and S. Lovett, On cryptography with auxiliary input, 41st Annual ACM Symposium on Theory of Computing, (2009) 621-630. <https://doi.10.1145/1536414.1536498>.
- [41] Y. Dodis, K. Haralambiev, L.-A. Adriana and W. Daniel, Cryptography Against Continuous Memory Attacks, in: IEEE 51st Annual Symposium on Foundations of Computer Science, 2010, Los Alamitos. <https://doi.10.1109/focs.2010.56>.
- [42] B. Zvika, T. K. Yael, K. Jonathan and V. Vinod, Overcoming the Hole in the Bucket: Public-Key Cryptography Resilient to Continual Memory Leakage, in: IEEE 51st Annual Symposium on Foundations of Computer Science, 2010, Los Alamitos. <https://doi.10.1109/focs.2010.55>.
- [43] H. Shai and L. Huijia, After-the-fact leakage in public-key encryption, Lecture Notes in Computer Science, 6597, (2011) 107-124. <https://doi.10.1007/978-3-642-19571-6>.
- [44] A. Janaka, S. Douglas and B. Colin, Modelling after-the-fact leakage for key exchange, in: Proceedings of the 9th ACM Symposium on Information, Computer and Communications Security - ASIA CCS '14, 2014, Kyoto. <https://doi.10.1145/2590296.2590317>.
- [45] A. Janaka, B. Colin and S. Douglas, Continuous after-the-fact leakage-resilient key exchange, in: Proc. Austral. Conf. Inf. Secur. Privacy (ACISP), 2014, Heidelberg. <https://doi.10.1007/978-3-319-08344-5>.
- [46] M. Bellare and P. Rogaway, Entity Authentication and Key Distribution, Lecture Notes in Computer Science, 773, (1993) 110-125. https://doi.10.1007/3-540-48329-2_21.
- [47] C. Ran and K. Hugo, Analysis of Key-Exchange Protocols and Their Use for Building Secure Channels, Lecture Notes in Computer Science, 2045, (2001) 453-474. https://doi.10.1007/3-540-44987-6_28.
- [48] B. LaMacchia, K. Lauter and A. Mityagin, Stronger Security of Authenticated Key Exchange, Lecture Notes in Computer Science, (2006) 1-16. https://doi.10.1007/978-3-540-75670-5_1.
- [49] NIST, U.S. Department of Commerce, Federal Information Processing Standards Publication Digital Signature Standard (DSS). <https://csrc.nist.gov/publications/>, 2013 (Accessed 18 July 2019).
- [50] J. Zhang, Authenticated key exchange protocols with unbalanced computational requirements, Ph.D. Thesis, University of Liverpool, 2018.
- [51] J. Zhang, X. Huang, W. Wang and Y. Yue, Unbalancing Pairing-Free Identity-Based Authenticated Key Exchange Protocols for Disaster Scenarios, IEEE Internet of Things Journal, 6 (1), (2019) 878-890. <https://doi.10.1109/jiot.2018.2864219>.
- [52] J. Cai et al., A Handshake Protocol with Unbalanced Cost for Wireless Updating, IEEE Access, 6, (2018) 18570-18581. <https://doi.10.1109/access.2018.2820086>.
- [53] J. Zhang, N. Xue and X. Huang, A Secure System for Pervasive Social Network-Based Healthcare, IEEE Access, 4 (2016) 9239-9250. <https://doi.10.1109/access.2016.2645904>.
- [54] J. Zhang et al., An Improved Protocol for the Password Authenticated Association of IEEE

- 802.15.6 Standard That Alleviates Computational Burden on the Node, *Symmetry*, 8 (2016) 131. <https://doi.10.3390/sym8110131>.
- [55] S. Li, Y. Mu, M. Zhang, F. Zhang, Continuous Leakage Resilient Lossy Trapdoor Functions, *Information*, 8 (2) (2017) 38. <https://doi.10.3390/info8020038>.
- [56] D. Dolev and A. Yao, On the security of public key protocols, *IEEE Transactions on Information Theory*, 29 (2) (1993) 198–208. <https://doi.10.1109/tit.1983.1056650>.

Highlights

- A leakage-resilient and lightweight AKE (LRL-AKE) protocol is proposed for WBAN.
- The LRL-AKE exhibits higher security level than protocols in mainstreaming standards.
- The LRL-AKE protocol is secure under the continual leakage model.
- The protocol reduces computation burdens on computationally limited devices.

Wenjun Zeng is currently pursuing the B.Eng. degree in Xi'an Jiaotong-Liverpool University and University of Liverpool. He majors in computer science and his current research interests include public key cryptography, Internet of Things, and Blockchain.

Jie Zhang received her Ph.D. degree from University of Liverpool in 2018. She is currently a lecturer with Xi'an Jiaotong-Liverpool University, Suzhou, China. Her research interests include public key cryptography, key management, and Internet of Things.



Wenjun Zeng



Jie Zhang