Provable secure identity-based online/offline encryption scheme with continual leakage resilience for wireless sensor network

Xiujie Zhang¹, Xingbing Fu²,³, Lu Hong¹, Yu Liu¹ and Liangliang Wang⁴

Abstract
As a potential technology, the identity-based online/offline encryption scheme is split into two phases (the offline phase and the online phase) which is especially suitable for sensor nodes with limited computation resources in that most of the works can be executed offline. However, a challenging issue is the well-known identity-based online/offline encryption schemes unable to resist continual key leakage attacks of the secret keys. To address the above security challenge, we put forth the first continual leakage-resilient identity-based online/offline encryption scheme which is suitable for ensuring secure communications in wireless sensor networks. More specifically, our formal security proofs analysis indicates that the proposed scheme can guarantee security even if partial information of the secret key is continually leaked due to side-channel attacks or fault injection attacks. Above all, compared to the existing identity-based online/offline encryption schemes, an identity-based online/offline encryption scheme with continual leakage resilient meets wireless sensor networks with strong security.

Keywords
Identity-based encryption, online/offline encryption, continual leakage resilience, wireless sensor networks, provable security

Date received: 9 January 2020; accepted: 2 April 2020

Handling Editor: Zheng Chang

Introduction
Wireless sensor networks (WSNs) as a major technique is an academic hotspot that possesses the advantages of building the Internet of things (IoT). Due to the demand for sensing, gathering, and monitoring data, individual users and enterprises make better management control and decision-making. For example, individual users equipped with tiny sensor nodes continuously monitor their sensitive data such as health records and so on which is necessary to be encrypted before transmitting to the base stations (BSs). However, the security and efficiency challenges need to design access control systems in WSNs. For one thing, the sensor nodes have the limited capabilities. For another, most of the existing access control schemes...
were proposed in a weak security model. In order to solve the above problems, researchers directly utilized typical identity-based encryption (IBE) schemes to design access control systems. IBE\textsuperscript{1–4} is one of the powerful cryptographic mechanisms to realize the above functionality. The encryption procedure of IBE simply employs the user’s identity which was composed of an arbitrary string. However, the IBE system always requires the user to execute expensive computations, such as pairings over points on the elliptic curve and exponentiations in groups which is not desirable for weak users with limited computation resources. Especially in WSNs, the nodes have very limited computation resources, memory, and storage capabilities. These make many cryptographic primitives impractical. A good solution to reduce the computational overhead of encryption procedure is the model of identity-based online/offline encryption (IBOOE) which splits the encryption process into two phases: the offline phase and the online phase. In the offline phase, the vast majority of the work is to be done and it requires neither the information of the message to be encrypted nor the receiver’s identity. This division of computational tasks makes most of the works executed offline. However, the security and efficiency challenges have appeared when traditional IBOOE schemes are directly utilized in the WSN environment. In such applications, it is usually assumed that the adversary cannot get any information about the internal secret states. In contrast, this assumption no longer holds in WSNs in that a malicious adversary can obtain partial information from side-channel attacks\textsuperscript{5} which will lead to the leakage of important sensitive information. Then, a natural question arises: Can we design a provable secure IBOOE scheme resilient to the continual leakage of secret keys? A plausible solution to the above problem might be dual system encryption which has the potential to address the problem of continual key leakage for traditional IBOOE schemes.

Related work

In this subsection, we summarize leakage-resilient IBE (lr-IBE) cryptography and the online/offline encryption cryptography.

Leakage resilience was first studied in public key setting by Akavia et al.\textsuperscript{9} It is a powerful tool to ensure security-protecting data confidential against side-channel attacks. Naor and Segev\textsuperscript{5} extended leakage resilience to the public key encryption. More broadly, Alwen et al.\textsuperscript{7} first introduced leakage-resilient security to identity-based setting and constructed lr-IBE scheme using the identity-based hash proof system (IB-HPS). Unfortunately, only the third scheme was proved in the standard model based on Gentry’s IBE system.\textsuperscript{2} So far, two main models have been proposed: bound leakage model (BLM) and the continual leakage model (CLM). The first line of work constructs lr-IBE schemes that are proven secure in the BLM. In CCS, Chow et al.\textsuperscript{8} proposed three lr-IBE schemes assuming that each secret-key holder had two randomness that applied to secret keys and master secret keys. The first two lr-IBE schemes based on BB-IBE\textsuperscript{3} and Waters-IBE\textsuperscript{2}, respectively, which had selective security and the third scheme was fully secure and short public parameters using Lewko and Waters’s dual system encryption.\textsuperscript{4} Namely, they showed how to go from IB-HPS to lr-IBE scheme. Li et al.\textsuperscript{9} investigated how to accomplish chosen ciphertext attack (CCA) security of the lr-IBE scheme by applying a hash proof technique into Gentry’s IBE scheme.\textsuperscript{3} However, the lr-IBE schemes presented in Alwen et al.,\textsuperscript{7} Chow et al.,\textsuperscript{8} and Li et al.\textsuperscript{9} cannot resistent to continual key leakage attack. Another line of work is so-called the CLM. It divides the lifetime of the cryptosystem into some periods, and updates the secret keys at the end of each period. Lewko et al.\textsuperscript{10} first proposed the IBE scheme under continual leakage resilient which allows continual leakage of user’s key and master key. Subsequently, Li et al.\textsuperscript{11} developed the first identity-based broadcast encryption scheme under continual leakage of secret keys of any identity. To improve the security level, Zhou et al.\textsuperscript{12–14} gave exciting new continual leakage-resilient constructions for IBE from decisional bilinear Diffie–Hellman (DBDH), truncated augmented bilinear Diffie–Hellman exponent (q-TABDHE) assumptions, and a new primitive called updatable identity-based hash proof system (U-IBHPS). Recently, many BLMs and CLMs were surveyed in Zhang et al.\textsuperscript{15} and Yuel et al.\textsuperscript{16}

Online/offline encryption technology has been widely accepted as an effective method to protect data security and privacy in online/offline scenarios and becomes a useful tool to achieve effective encryption such that only the authorized users can access the data. Online/offline encryption technology mainly focuses on efficient encryption which is different from efficient decryption.\textsuperscript{17} Guo et al.\textsuperscript{18} first introduced encryption schemes which split the encryption procedure into two phases: the offline phase and the online phase. In the offline phase without input of plaintext and user’s identity, it outputs the offline ciphertext. Then, in the online phase, it efficiently generates the online ciphertext concerning plaintext and the identity of the user. Subsequently, Liu et al.\textsuperscript{19} presented a more efficient IBOOE scheme in the random oracles which is especially suitable for embedded devices of limited computation power such as smart cards or wireless sensors. Later, Chu et al.\textsuperscript{20} proposed a practical IBOOE scheme in the selective-ID model, which is suitable for WSN which removed the elements in the group $G_T$ from the offline storage. The notion of ID-based online/offline key encapsulation mechanism (IBOOKEM) was
introduced by Chow et al.\textsuperscript{21} and a generic construction of the IBOOE scheme to achieve CCA security in the ROM. However, the above scheme has the following property: the security, storage, and computation have relied on symmetric encryption mechanism. By utilizing a novel notion to obtain an efficient semi-generic transformation from IBOOKEM\textsuperscript{22} scheme to IBOOE scheme in the standard model. Recently, to address the problem of large ciphertext, efficient IBOOE and signature with short ciphertext were proposed by Lai et al.\textsuperscript{23} Nevertheless, according to the technologies of side-channel attacks and new attack methods are appearing, the conventional IBOOE scheme will be threatened. Thus, those schemes mentioned above will be broken by key leakage attacks. As an attractive technique for naturally leading to leakage resilience, dual system encryption holds very strong security proofs based on three static assumptions. In this article, our goal is to propose an IBOOE scheme in the continual key leakage security model.

Motivation and our contribution

The primary challenge for the IBOOE schemes is to achieve leakage resilience. In this article, we consider the continual leakage resilience in IBOOE schemes. Specifically, the contributions of this work are elaborated as follows:

- Motivated by the aforementioned application to leakage tolerance in WSNs, we study IBOOE resilient to a stronger form of leakage. We first introduce the notion of the continual leakage-resilient IBOOE (clr-IBOOE) and formalize its framework. In clr-IBOOE, the encryption procedure is split into two phases, that is, an offline phase and an online phase. To support continual leakage functionality, we expand the space of secret key and ciphertext to $n + 2$ and $n + 5$ dimensional vectors, respectively. Here, the parameter $n(n \geq 3)$ will determine the leakage tolerance.
- We utilize the semi-functional structures and a hybrid argument which was used in the security proof by a sequence of games. The main difficulty in constructing our scheme is to implement nominally semi-functional secret keys that can decrypt semi-functional ciphertext without the adversary awareness of. Nominality permits the simulator to produce nominal semi-functional keys which is decrypt successfully the interrelated challenge ciphertext while the adversary was unaware of the above process.
- We provide provable security of clr-IBOOE, including indistinguishability of normal secret keys and semi-function secret keys, indistinguishability of normal ciphertext and semi-function ciphertext, and continual leakage of secret keys. In addition, we conduct a comprehensive performance evaluation compared with the existing IBOOE schemes, our scheme not only enjoys continual leakage resilience but also avoids the additional online computation. Specifically, clr-IBOOE dramatically realizes the strongest features of continual leakage. This is because that in update process, the secret key is updated over time.

The rest of the article is organized as follows: in section “Preliminaries,” we describe the notation used in this article, composite order bilinear and complexity assumptions. In section “System architecture and security model,” we first present a continual leakage-resilient security model and a formal definition for IBOOE, respectively. Then, we propose a clr-IBOOE scheme based on three assumptions in which encryption can be afforded by mobile devices with limited computation power. We elaborate on formal security proof in the standard model based on the new security model in section “System architecture and security model.” And in section “Provable secure IBOOE with continual leakage resilience in WSNs,” we give efficiency results and security of the new scheme compared to the existing IBOOE schemes. At last, we conclude this article in section “Security proof and efficiency comparisons.”

Preliminaries

In this section, we introduce the preliminary knowledge on some cryptographic background used throughout this article.

Notations

Assuming that all the algorithms take as input the security parameter represented in unary, that is, $1^\lambda$. Let $N$ be a positive integer. For a set $R$, $|R|$ represents its cardinality and means that the variable $s$ is chosen uniformly at random from $R$. Let symbols $\cdot$ and $*$ denote dot product of two vectors and component-wise product of two elements, respectively. In addition, the useful notations for IBOOE scheme are listed in Table 1. These notations are employed throughout this literature.

Composite order bilinear groups

Three assumptions based on the pairing over composite order groups\textsuperscript{4,10} are as follows. Let the parameters be $N = p_1p_2p_3$, where $p_1$, $p_2$, $p_3$ are three numbers. The group generator algorithm $I$ takes as input a security parameter $1^\lambda$ and outputs the description of a bilinear
group \((N = p_1p_2p_3, G, G_T, e)\). In the group description, \(p_1, p_2, p_3\) are three distinct numbers; \(G\) and \(G_T\) are cyclic groups of order \(N\); and \(e: G \times G \rightarrow G_T\) is a bilinear map. We assume that the generator algorithm outputs the values of \(p_1, p_2, p_3\) and \(g_i\) is generator of subgroup \(G_i\) of order \(p_i\). Moreover, let \(G_{ij}\) denote the subgroup of order \(p_ip_j\), which implicitly have \(e(h_i, h_j) = 1_{G_{ij}}, h_i \in G_i\) and \(h_j \in G_j\) for \(i \neq j\).

### Complexity assumptions

**Assumption 1.** Given \(D^1 = (N, G, G_T, e, g_1, g_3)\), it is hard for an adversary to distinguish \(T_0^1 = g_1^z\) from \(T_1^1 = g_1^zg_2^z\) where \(z, v \in \mathbb{Z}_N\).

**Assumption 2.** Given \(D^2 = (N, G, G_T, e, g_1, g_3, g_1g_2, g_1g_3, g_2g_3)\), it is hard for an adversary to distinguish \(T_0^2 = g_1^w g_3^k\) from \(T_1^2 = g_1^w g_2^k g_3^w\) where \(w, k, \sigma \in \mathbb{Z}_N\).

**Assumption 3.** Given \(D^3 = (N, G, G_T, e, g_1, g_2, g_3, g_1g_2, g_1g_3, g_2g_3)\), it is hard for an adversary to distinguish \(T_0^3 = e(g_1, g_1)^{z_1}\) from \(T_1^3 \in G_T\).

The advantage of an algorithm \(A\) in Assumption \(i\) where \(i \in \{1, 2, 3\}\) is defined as

\[
\text{Adv}_{\mathcal{A}}^i(\lambda) = \left| \Pr[A(D', T_0') = 1] - \Pr[A(D', T_1) = 1] \right|
\]

### Security theorem

In the context of Lewko and Waters,\(^\text{10}\) it expands the semi-functional space to \(n + 2\) dimensional vectors, where \(n \geq 3\). Notice that, the leakage depends only on the size of the \(G_2\) subgroup, and not on the size of \(p_1\) or \(p_3\). They give the Lemma 1 as follows.

**Lemma 1.** Under Assumptions 1, 2, 3 and for \(l_{SK} = (n - 1 - 2c)\log(p_2), l_{MK} = (n - 1 - 2c)\log(p_2)\) where \(c > 0\) is a fixed positive constant, our dual system IBE scheme is \((l_{MK}, l_{SK})\)-master-leakage secure.

### System architecture and security model

Before giving the formalized security model, we first lay out the system architecture and specification of an IBOOE scheme with the key update algorithm which is the underlying technique of the proposed secure identity-based online/offline scheme with continual leakage resilience for WSNs.

#### Network model

In WSNs, we focus on single-SP (service provider) multi-user sensor network. The overview of the network model is shown in the following Figure 1. There are four distinct entities in the system, that is, the SP, the sensor node, the gateway, and the network user.

- **SP** is an authority who deploys a sensor network composed of multiple sensors, such as the registration of sensor nodes and users. The SP is a trusted third party.
- **Sensor nodes** are battery powered which is a big challenge for network designers. Note that sensor nodes can offline.
- **Gateway:** compared with sensor nodes, it has much higher storage and processing capability.
- **Users:** a user can join the system by registering with the SP to join the WSN.

It is assumed that all the entities except SP are “honest-but-curious.” Sensor nodes generate offline ciphertexts, encrypt a file, and outsource the final ciphertext to SP. When users use the encrypted data, he downloads a ciphertext from SP. Then, he decrypts it based on the secret keys if the ciphertext is legitimate.

### Specification of clr-IBOOE

An IBOOE scheme typically consists of six algorithms: Setup, Ext, UpdateSK, Enc\(^{\text{off}}\), Enc\(^{\text{on}}\), and Dec. In the continual leakage setting, we require an additional algorithm UpdateSK which updates the secret key of users. We define a continual leakage resilience IBOOE scheme, clr-IBOOE in short. Note that the public key remains unchanged. The definition below summarizes the above.

**Definition 1 (clr-IBOOE).**

IBOOE schemes are defined by a tuple of algorithms and protocols \(\Pi = (\text{Setup}, \text{Ext}, \text{UpdateSK}, \text{Enc}^{\text{off}}, \text{Enc}^{\text{on}}, \text{Dec})\) where
Setup(1) takes the security parameter λ to return the master public key mpk and the master secret key msk.

Ext(mpk, I, msk) takes as input the master public key mpk, master secret key msk, an identity I ∈ S and returns a secret key sk_I for I.

UpdateSK(sk_I, I, mpk) takes in a secret key sk_I and outputs a re-randomized key sk'_I, such that the distribution of a secret key sk'_I is indistinguishable from the distribution of a secret key sk_I.

Enc^off(mpk) takes mpk to return an offline ciphertext coff.

Enc(m, mpk, I, coff) takes as input the message m, the master public key mpk, an identity I ∈ S, and an offline ciphertext coff and returns the ciphertext c.

Dec(mpk, sk_I, c) takes a secret key sk_I and ciphertext c to return a message m or a special reject symbol ⊥ indicating c is invalid.

Correctness. With probability 1 over the randomness of (Setup, Ext, UpdateSK, Enc^off, Enc^on, Dec), we have that Dec(mpk, sk_I, Enc^on(m, mpk, I, Enc^off(mpk))) = m where (mpk, msk) ← Setup(λ) and sk_I ← Ext(mpk, I, msk).

Security model

In our surroundings, there exist two adversaries either an external intruder or a registered network user who intended to obtain other users’ data without the authorization of access according to work.24 Based on the adversary’s ability and the specification of IBOOE scheme, we formalize the security model by specifying the ability of adversaries. To define continual leakage security, we extend the CLM of Li and Zhang11 to online/offline identity-based setting. The security model is defined through a game played by a probabilistic polynomial-time (PPT) adversary A and a challenger (also called simulator). Assuming A cannot win the following security game cLeak with probability greater than (1/2) + negl(λ). All the queries in the following security can happen adaptively, that is, they can depend on previous ones.

Setup: The simulator B runs the Setup(1^λ) algorithm, obtains the public secret key mpk and the master secret key msk. The master secret key is kept secret. The challenger gives mpk to the adversary A. B creates a set L = Ø to keep track of Ext queries and key leakage queries. Moreover, another set K = N corresponds with Reveal queries. The set L of triples of handles, identities, secret keys, and a counter, that is, (h, I, SK, Cntr) ∈ H × I × SK × N. Note that a handle counter h is set to 0.

Phase 1: The adversary A makes Ext, Leak, Reveal, UpdateSK, and Decrypt queries as follows:

Ext(h, I): h is a handle that has to refer to a secret key. Given an identity I, the challenger B looks up the item with the identity I in the list L. If it exists, B terminates. Otherwise, it creates a key for this identity I by calling the key extraction algorithm Ext(I, MSK) → SK_I, updates h ← h + 1 and adds the item (h, I, SK_I, 0) into the list L.
**Leak(h, f):** Given a handle h, the challenger looks up the item with the handle h in L. A selects a polynomial-time computable function f of constant output size acting on the set of keys. If the handle h exists, B checks if \( \text{Cntr} + f(SK_h) \leq l_{SK} \) where \( l_{SK} \) is the leakage bound for the secret key. If this is true, it responds with \( f(SK_h) \) and updates the Cntr with \( \text{Cntr} + f(SK_h) \).

**Reveal(h):** The attacker A requests the entire secret key for handle h. The simulator looks up in the list L. If it finds the item, it denotes the tuple by \((h, I, SK_f, \text{Cntr})\). B returns this item and adds the identity I to the set K.

**UpdateSK(I):** The simulator queries a new secret key of I with a polynomial-time computable function \( f \). B looks up the item with the handle h in L. Then, it runs algorithm UpdateSK(SK_h) to get the updated secret key \( SK'_h \). B forwards the updated secret key \( SK'_h \) and refreshes the set L with \((h, I, SK', 0)\).

**Decrypt(I, c):** The adversary queries a message for \((I, c)\). The simulator looks up the list L to get the secret key \( SK_h \). The simulator calls the algorithm Dec\((sk_l, c)\) to recover the corresponding plaintext \( m \) and sends it to the adversary.

**Challenge:** Once A decides that Phase 1 is over, it outputs two equal-length messages \( m_0 \) and \( m_1 \). The simulator B chooses \( b \in \{0, 1\} \) randomly and encrypts \( m_b \) under \( I^* \) by calling algorithm Enc\(^\text{off}(m_b, mpk, I^*, \text{Enc}\(^\text{off}(mpk)\)) \to c^* \). It sends \( c^* \) to A.

**Phase 2:** The same as Phase 1, except that \( I^* \) may not be submitted for all the oracles in Phase 1.

**Guess:** Finally, A outputs a guess bit \( b^* \in \{0, 1\} \). If \( b^* = b \), A wins the game. The advantage of an adversary A in attacking the IBOOE scheme \( II \) with security parameter \( \lambda \) is defined as \( \text{Adv}^{\text{Leak}}(\lambda, I) = |\Pr[b^* = b] - 1/2| \).

**Definition 2.** If for all PPT adversaries A, the advantage \( \text{Adv}^{\text{Leak}}(\lambda, I) \) in the above game \( c\text{Leak} \) is negligible, that is, \( \text{Adv}^{\text{Leak}}(\lambda, I) \leq \text{negl}(\lambda) \), we say that a clr-IBOOE scheme \( II \) is (Setup, Ext, UpdateSK, Enc\(^\text{off}, \text{Enc}\(^\text{on}, \text{Dec}\)) is fully secure against \( l\)-continual leakage attacks, the total amount of leakage on each secret key of user has to be bounded by \( \lambda \). Note that in all definitions and lemmas we will not write the dependence of \( \text{Adv} \) in \( \lambda, I \) for easiness of notation.

**Provable secure IBOOE with continual leakage resilience in WSNs**

In this section, we first present the proposed IBOOE with continual leakage resilience in WSNs and then show its correctness. Our clr-IBOOE scheme \( II \) comprises six algorithms detailed as follows.

### Details of the proposed IBOOE scheme

Our construction of clr-IBOOE scheme employs a composite order bilinear groups \( I = (N = p_1p_2p_3, G, G_T, e) \) where \( p_1, p_2, p_3 \) are three A-bits length primes; \( G, G_T \) are cyclic groups with order \( N = p_1p_2p_3 \); and \( e \) is such a bilinear map \( e: G \times G \to G_T \). Assume that each identity is an element in \( \mathbb{Z}_N \) and every message is an element in group \( G_T \). In our scheme, the secret key is randomized by subgroup \( G_3 \). The elements of subgroup \( G_2 \) will not be used in the real scheme, but it supplies semi-functional property.

- **Setup(\( I^*: \)** The setup algorithm picks \( (\alpha, x_1, \ldots, x_n) \sim Z_n^+ \), \( u_1, g_1, h_1 \in G_1 \) at random. It computes \( R_1 = e(g_1, g_1)^\alpha \). Finally, it outputs the master secret key \( msk = \alpha \) and the master public key \( mpk = (u_1, g_1, h_1, R_1, g_1^i, \ldots, g_1^i) \).
- **Ext(I, mpk, mkp):** On input the master public key \( mpk \), the master secret key \( msk \), and an identity \( I \in \mathbb{S} \), the algorithm picks \( n + 1 \) random exponents \( (r, r_1, \ldots, r_n) \sim Z_n^+ \) and returns the secret key \( sk_I = (k_1, k_2, k_3) = (g_1^{r_1}, \ldots, g_1^{r_1}, g_1^{r_1} \cdot (u_1^{k_1})^r \cdot \prod_{i=1}^{n} g_i^{k_1}, g_i^{r_1}) \).
- **Enc\(^\text{off}(mpk):** On input the master public key \( mpk \), the algorithm chooses \( x, s, w \in \mathbb{Z}_N \) and computes the offline ciphertext \( c_{\text{off}} = (c_1, c_2, c_3, c_4, c_5, w) = ((g_1^s)^{c_1}, \ldots, (g_1^s)^{c_1}, g_1^{w_1}, (u_1^{k_1})^{c_1}, u_1^{w_1}, R_1, w) \).
- **Enc\(^\text{on}(mpk, I, c_{\text{off}}):** On input message \( m \), master public key \( mpk \), an identity \( I \in \mathbb{S} \), and offline ciphertext \( c_{\text{off}} \), the algorithm picks \( x, s, w \in \mathbb{Z}_N \) and encrypts the message by computing \( c_0 = m \cdot c_5 \) and \( w_1 = w - 1 \mod N \). Finally, it outputs the final ciphertext \( c = (c_0, c_1, c_2, c_3, c_4, w_1) \).
- **Dec\((sk_l, c, mpk):** Let the ciphertext \( c \) be parsed as \((c_0, c_1, c_2, c_3, c_4, w_1)\). The decryption algorithm calculates the binding factor \( e(g_1, g_1)^{s_0} \) using the secret key \( sk_l \)

\[
\begin{align*}
&= e(k_1, c_1) \cdot e(k_2, c_2) \cdot e(k_3, c_3) \cdot e(k_n, c_n) \\
&= e(g_1^{c_1}, (g_1^{s_1})^{c_1}) \ldots e(g_1^{c_1}, (g_1^{s_1})^{c_1})e \\
&\quad \cdot (g_1^{s_1} \cdot (u_1^{k_1})^{r} \cdot \prod_{i=1}^{n} g_i^{k_1}, g_i^{r_1}) \\
&\quad \cdot e(g_1^{s_1}, (u_1^{k_1})^{r} \cdot (u_1^{k_1})^{w_1 - 1})^* \\
&= e(g_1, g_1)^{s_0}
\end{align*}
\]

Finally, the message is computed as \( m = c_0/(e(g_1, g_1)^{s_0}) \). It proves correctness of decryption procedure.
Continual leakage

To implement re-randomization, we give here the update algorithm for the secret keys.

- **UpdateSK**($sk_i$, $I$, $mpk$): The update algorithm picks $n + 1$ exponents $\langle \bar{r}, z_1, \ldots, z_n \rangle \overset{\$}{\leftarrow} Z_N^{n+1}$ and a random vector $\vec{p} \overset{\$}{\leftarrow} Z_N^{n+2}$. It outputs the new secret key as follows

$$sk'_i = (k_1, k_2, k_3) = sk_i \cdot \left\langle g_1^{z_1}, \ldots, g_n^{z_n}, (w_i h_i)^{-1} \right\rangle \prod_{i=1}^{n} g_1^{-r_i z_i}, g_1'$$

Semi-functionality

All the secret keys and ciphertext generated by the above algorithms are normal which has no group $G_2$ parts. To get semi-functional key and ciphertext, we add a factor of $G_2$ into the normal key and normal ciphertext as follows:

- **KeygenSF**($I$, $msk$): Let $sk_i$ be a normal secret key of user $I$ which was generated by calling key extraction algorithm $Ext(I, msk, mpk)$ and $g_2$ be a generator of group $G_2$. Then picks random $\tilde{\theta} \overset{\$}{\leftarrow} Z_N^n, \theta_n + 1, \theta_n + 2 \in Z_N$ and calculates the semi-functional secret key $sk_i^\text{semi} = (k_1 \cdot g_2^{\bar{r}}, k_2 \cdot g_2^{\theta_n + 1}, k_3 \cdot g_2^{\theta_n + 2})$.

- **EncryptSF**($sk_i$, $I$, $mpk$): It first calls the online encryption algorithm $Enc_{\text{on}}(m, mpk, I, Enc_{\text{of}}(mpk))$ to obtain a normal ciphertext $c = (c_0, \tilde{c}_1, c_2, c_3, c_4, w_1)$. Then, it picks random $\tilde{\delta} \overset{\$}{\leftarrow} Z_N^n, \delta_n + 1, \delta_n + 2 \in Z_N$ and calculates a semi-functional ciphertext $c_{\text{semi}} = (c_0, \tilde{c}_1 \cdot g_2^{\bar{r}}, c_2 \cdot g_2^{\theta_n + 1}, c_3 \cdot g_2^{\theta_n + 2}, c_4 \cdot g_2^{\delta_n + 1}, w_1)$. Here, we contract the $n + 2$ factors of semi-functional coefficient by $\delta_n + 2 = \delta'_n + \tilde{\delta}'$.

Procedure of decryption

First, we contract the two terms $\bar{\theta}, \tilde{\delta}$ the semi-functional parameters of the secret key and the ciphertext, respectively. The algorithm $Dec(sk_i, c, mpk)$ shows the detailed procedure on how to decrypt the normal ciphertext using the normal secret key. That is, it concludes $e(g_1, g_1)^{\alpha z}$ by computing $e(k_1, c_1) \cdot e(k_2, c_2)$ $e(k_3, c_3 \cdot c_4^{\tilde{\delta}'})$. Second, we elaborate on the whole derivation of decryption procedure about two types of secret keys and ciphtertexts.

1. The normal secret key decrypts the semi-functional ciphertext as follows

$$e(k_1, \tilde{c}_1 + g_2^{\bar{r}}) \cdot e(k_2, c_2 \cdot g_2^{\theta_n + 1})$$

$$e(k_3, c_3 \cdot g_2^{\theta_n + 1}, c_4 \cdot g_2^{\delta_n + 1}) = e(g_1, g_1)^{\alpha z}$$

2. The semi-functional secret key decrypts the normal ciphertext as follows

$$e(k_1, \tilde{c}_1 + g_2^{\bar{r}}) \cdot e(k_2, g_2^{\theta_n + 1}, c_2)$$

$$e(k_3, g_2^{\theta_n + 1}, c_3 \cdot c_4^{\tilde{\delta}'}) = e(g_1, g_1)^{\alpha z}$$

3. The semi-functional secret key decrypts the semi-functional ciphertext as follows

$$e(k_1, \tilde{c}_1 + g_2^{\bar{r}}) \cdot e(k_2, g_2^{\theta_n + 1}, c_2)$$

$$e(k_3, g_2^{\theta_n + 1}, c_3 \cdot c_4^{\tilde{\delta}'}) = e(g_1, g_1)^{\alpha z}$$

Here, we write $e(g_2, g_2)^{\tilde{\theta}} \cdot e(g_2, g_2)^{\theta_n + 1}$. $e(g_2, g_2)^{\theta_n + 1} \cdot (\delta'_n + \tilde{\delta}')$ where implicitly include $\delta_n + 2 = \delta'_n + \tilde{\delta}'$.

Through the above analysis, we obtain an additional entry $e(g_2, g_2)^{\tilde{\theta}} \cdot (\delta'_n + \tilde{\delta}')$ by the pairing. That is, a semi-functional secret key of identity $I$ with parameter $\tilde{\theta}$ is nominal relative to a semi-functional ciphertext for identity $I$, with parameter $\tilde{\delta}$ if and only if $\tilde{\theta} \cdot \tilde{\delta} = 0 \mod p_2$ for $I_k = I_e$.

Security proof and efficiency comparisons

Security proof

**Theorem 1.** If Assumptions 1, 2, and 3 hold in $G$, $G_T$, then the above proposal II is $l_{SK}$-fully secure against key leakage attacks.

**Proof.** Each secret key $sk_i$ is a $n + 2$ dimensional vector where $n \equiv 3$ is a parameter determining the leakage tolerance. Based on lemma 1 in section “Security theorem,” we have leakage parameters $l_{SK} = (n - 1 - 2c)\lambda$. To prove theorem, we will build a PPT simulator $B$ that breaks Assumption 1, 2, 3 with the help of a PPT adversary $A$ that breaks the adaptive security of our proposal II. First, we consider the following sequence of hybrid games.

- **cLeak.** This game is the original game as defined in the section “Security model.”

- **LeakI.** This game has the following differences from cLeak: first, the simulator responds a semi-functional ciphertext as the challenge ciphertext. Second, the first $i$ keys generated by $Ext$ queries are semi-functional for $i = 0, \ldots, q$. Remark that the simulator returns normal
secret keys for key extraction queries and a semi-functional challenge ciphertext for challenge phase in game Leak. Moreover, the simulator returns semi-functional secret keys for key queries in game Leak.

**Lemma 2.** Suppose there exists a PPT algorithm $A$ such that $Adv^{Leak_1}_{A, \Pi} - Adv^{Leak_0}_{A, \Pi} \geq \varepsilon$. Then, we can build a PPT algorithm $B$ with advantage at least $\varepsilon/2$ in breaking Assumption 1.

**Proof.** The algorithm $B$ simulates fully the encryption scheme $\Pi$ as follows. $B$ receives $D^i = (N, G, G_T, e, g_1, g_3)$ and $T$ where $T = g_1^{1}$ or $g_1^{*g_2}$. Then, $B$ randomly chooses $u_1 = g_1^{h_1}$ and calculates $h_2 = g_1^{u_1}, \ldots, h_{i-1}^{u_{i-1}}$. $R_i = e(g_1, g_i)^{\psi}$ using group element $g_i$ in Assumption 1. $B$ sets $msk = \alpha$ as the master secret key and has ability to provide two types of secret keys for all identities to answer all key extraction queries and key leakage queries. Moreover, $B$ makes normal ciphertext as well as semi-functional one for any message in the challenge phase.

Both games, $B$ always calculates normal secret keys for all identities as follows. $B$ randomly chooses $(a, b, \alpha, x_1, \ldots, x_y) \sim \mathbb{Z}_N^+$ and a random vector $\vec{\rho} \sim \mathbb{Z}_N^{+2}$, creates the secret key of identity $I$.

$$s_{k_I} = g_1^{y_1}, \ldots, g_1^{y_y}, g_1^{-r(a+b)}, \prod_{i=1}^{y} g_1^{-x_i}, g_1^\vec{\rho}$$

In the challenge phase, $B$ gets two messages $m_0, m_1$ and the challenge identity $I$ from $A$. Then, it randomly chooses $b \in \{0, 1\}, x, w \in \mathbb{Z}_N$ and creates the following semi-functional challenge ciphertext using the challenge term $T$ from Assumption 1.

$$c^* = (m_b \cdot e(g_1, T^a, T^{x_1}, \ldots, T^{x_y}, T, T^{x+y+b}, T^w, w^1)$$

where $w^1 = w^{-1}(I - x) \mod N$.

**Phase 2: $B$ works in the same way as Phase 1.**

If $T = g_1^1$, then $c^* = (m_b \cdot e(g_1, g_1)^{\psi} , (g_1^{y_1})^{\psi}, \ldots , (g_1^{y_y})^{\psi}, g_1^{x_1}, \ldots, g_1^{x_y}, g_1^{-r(a+b)}, g_1^\vec{\rho})$ which has no $G_2$ components. Hence, it is a normal ciphertext. In other words, $B$ has properly simulated game $\text{Leak}$. Because it set $u_1 = g_1^1$ and $h_1 = g_1^1$ at the begin, that is, it has $(g_1^{x_1})^{a+b} = u_1^h h_1$.

If $T = g_1^{g_2}$, the challenge ciphertext generated as follows.

$$c^* = (m_b \cdot e(g_1, g_1)^{\psi}, (g_1^{g_2})^{\psi}, \ldots , (g_1^{g_2})^{\psi}, g_1^{g_2}, (g_1^{x_1})^{a+b}, (g_1^{g_2})^\vec{\rho}, w^1)$$

This implicitly has $s = z$. We get the semi-functional parameter $\tilde{\delta} = v(x_1, \ldots, x_y, (ax + b + av))$ according to decryption. $A$ plays game $\text{Leak}_0$. Hence, if $B$ answers $b = 0$ when $A$ succeeds, he has advantage in breaking Assumption 1. Thus, if Assumption 1 holds, these two successive games $\text{Leak}$ and $\text{Leak}_0$ are indistinguishable.

**Lemma 3.** Suppose there exists a PPT algorithm $A$ such that $Adv^{Leak_1}_{A, \Pi} - Adv^{Leak_0}_{A, \Pi} \geq \varepsilon$. Then, we can build a PPT algorithm $B$ with advantage at least $\varepsilon/2$ in breaking Assumption 2.

**Proof.** If the challenge instance $B$ simulates fully, the encryption scheme $\Pi$ as follows. $B$ receives $D^2 = (N, G, G_T, e, g_1, g_3, g_1^{g_2}, g_2^{g_3})$ and $T$ where $T = g_1^{g_2}$ or $g_1^{g_2} g_2^{g_3}$. Then, $B$ constructs the public parameters and the secret parameters as lemma 1. At some point, the adversary $A$ sends identity $I$ to simulator $B$. $B$ simulates the secret key as below.

If $k < i$, $B$ generates the semi-functional secret key for identity $I$ as follows

$$s_{k_I} = (g_1^{g_2})^{y_1}, \ldots, (g_1^{g_2})^{y_y}, g_1^\rho \cdot (g_1^{g_2})^{-(a+b)}$$

$$\prod_{i=1}^{y} g_1^{-x_i}, g_1^\vec{\rho}$$

If $k = i$, $B$ simulates the challenge secret key for identity $I$ as follows

$$s_{k_I} = T^{a}, \ldots, T^{x_y}, T, T^{a+b}, T^w, w^1$$

If $k > i$, $B$ creates the secret key of identity $k_I$ like lemma 1.

In challenge phase, $B$ selects two distinct and random messages $m_0, m_1$ with equal length and the challenge identity $I$. It sends the messages to $A$. Then, it randomly chooses $b \in \{0, 1\}, x, w \in \mathbb{Z}_N$ and returns the challenge ciphertext as follows

$$c^* = m_b \cdot e(g_1^{g_2}, (g_1^{g_2})^{y_1}, \ldots, (g_1^{g_2})^{y_y}, g_1^{x_1}, g_1^{x+b}, (g_1^{g_2})^w, w^1)$$

Here, it has $w^1 = w^{-1}(I - x) \mod N$.

This implicitly has $s = z$ and $\delta = v(x_1, \ldots, x_y, 1, a+b)$. Hence, if $B$ answers $b = 0$ when $A$ succeeds, he has advantage in breaking Assumption 2. Thus, if Assumption 2 holds, $\text{Leak}_{i-1}$ and $\text{Leak}_i$ successive games are indistinguishable.

**Lemma 4.** Suppose there exists a PPT algorithm $A$ such that $Adv^{Leak_{i-1}}_{A, \Pi} - Adv^{Leak_{i}}_{A, \Pi} \geq \varepsilon$. Then, we can build a PPT algorithm $B$ with advantage at least $\varepsilon/2$ in breaking Assumption 3.
Table 2. Comparisons of the existing IBOOE and clr-IBOOE scheme.

|     | Off-storage | On-encrypt | Assumption | Security model | LR | CLR |
|-----|-------------|------------|------------|----------------|----|-----|
| CLZ\(^{20}\) | 4[\(G_1\)] | \(e + 2m_1(Z_\emptyset)\) | DBDH | Standard | \(\times\) | \(\times\) |
| LMG\(^{19}\) | 3[\(G_1\)] + [\(G_1\)] | 2\(m_1(Z_\emptyset)\) | k-CAA1 | ROM | \(\times\) | \(\times\) |
| Ours | \((n + 4)[\(G_1\)] + [\(G_1\)]\) | 1\(m_1(Z_\emptyset)\) | 1, 2, 3 | Standard | \(\checkmark\) | \(\checkmark\) |

IBOOE: identity-based online/offline encryption; clr-IBOOE: continual leakage-resilient IBOOE; LR: leakage resilience; CLR: continual leakage resilience; DBDH: decisional bilinear Diffie–Hellman.

**Proof.** B simulates fully the encryption scheme II as follows. B receives \(D^3 = (N, G, G_T, e, g_1, g_2, g_3, g_1Tg_2, g_1g_3T)\) and \(T\) where \(T = e(g_1, g_1)^{xz}\) or \(T \in G_T\). The public parameter \(mpk\) and the master secret key \(msk\) as lemma 1. When B receives secret key queries, he constructs the semi-functional secret key

\[
sk_I = \left( (g_1^Tg_2)^{y_1}, \ldots, (g_1^Tg_2)^{y_n}, g_1^{\alpha}, (g_1^Tg_2)^{-\alpha(af + b)} \right)
\]

\[
= \prod_{i=1}^{n} g_i^{-x_iy_i}g_1^{\alpha}g_2^{b} + g_3^{b}
\]

In the challenge phase, the challenge ciphertext generates as follows

\[
c^* = m_I \cdot ((g_1^Tg_2)^{-\alpha}, \ldots, (g_1^Tg_2)^{-\alpha}, g_1^{-\alpha} = \alpha + h, (g_1^Tg_2)^{w_1})
\]

Here, it has coefficient \(w_1^* = w^{-1}(I^* - x)\)modN.

If \(T = e(g_1, g_1)^{xz}\), then this is a semi-functional ciphertext of message \(m_I\). This is because the semi-functional parameters \(s = z\) and \(T = e(x, y, u, u(\alpha I^* + b)) \) implicitly \(\delta_{n+2} = \delta_{n+2} + \delta_{n+1}w_1\). It is easy to verify vector \(\delta\) is properly distributed since all terms \(\{x_1, \ldots, x_n, 1(\alpha I^* + b)\}\) are random modulo \(p_2\).

Finally, in game \(\text{Leak}\_F\) value of \(b\) is information theoretically hidden from the adversary A. Consequently, A has no advantage in winning \(\text{Leak}\_F\). Hence, \(c^\text{Leak}\) is indistinguishable from \(\text{Leak}\_F\) if Assumption 1, 2, and 3 hold. As a result, the adversary has negligible advantage in winning game \(\text{Leak}\). Lemma 2, Lemma 3, and Lemma 4 conclude the proof of Theorem 1. Thus, we finish the proof of Theorem 1.

**Theorem 2.** If Assumptions 1, 2, and 3 hold in \(G, GT\), our proposed scheme II has continual leakage resilience in the standard model.

**Proof.** According to the algorithm UpdateSK given in section “Continual leakage,” we get continual leakage resilience. The algorithm UpdateSK inputs the secret key \(sk_I\), identity I, and master public key \(mpk\) and then outputs a new secret key \(sk'_I\). In our update algorithm, it picks \(n + 1\) values from \(Z_N\), adds extra \(G_1\) part and \(G_3\) part to all the \(n + 2\) parts of the secret key \(sk_I\) in turn. Hence, the generated secret key has the same distribution with the old one. As a result, we achieve continual leakage resilience by calling UpdateSK periodically.

**Leakage resilience and performance analysis**

In this section, we draw a comparison between the proposed scheme and the existing IBOOE schemes in Table 2. The computation efficiency and offline-storage cost are determined by the computation costs of the encryption algorithm which constitutes two subroutines \(\text{Enc}^{\text{old}}(mpk)\) and \(\text{Enc}^{\text{old}}(m, mpk, i, e^{\text{old}})\). When calculate the computation efficiency, we omit minor factors such as hash functions. We calculate only the dominant operations which are the exponentiations in \(G_1\) or \(G_2\) denoted as \(E\) and the multi-exponentiations in \(G_1\) or \(G_T\) denoted as \(ME\). For modular computation in \(Z_n\) and \(Z_N\), we denote by \(m_i(Z_n)\) and \(m_i(Z_N)\). Without loss of generality, we assume that one multi-exponentiation is equal to two exponentiations in group \(G_1\) or \(G_T\). Moreover, the size of element in the group \(G_1\) and \(G_T\) is denoted by \(|G_1|\) and \(|G_T|\), the size of element in \(Z_N\) is denoted by \(n\). Furthermore, to avoid a large number of calculations at the online encryption phase, we place it in the offline encryption phase. According to Table 2, we can obtain that our proposal have the minimum computation of \(\text{Enc}^{\text{old}}(\cdot, \cdot, \cdot)\) algorithm and achieve leakage resilience. This is suitable for WSNs which were interconnected sensor nodes that communicate wirelessly to collect data about the surrounding circumstance. Moreover, LR and CLR denote the security of leakage resilience and continual leakage resilience, respectively. Even though the offline storage has a larger size than other two traditional IBOOE schemes. From Table 2, it is clear that our scheme resolves the problem of continual leakage resilience which can be obtained naturally using the techniques of dual system encryption.

**Conclusion**

In this article, we propose the first clr-IBOOE scheme which is suitable for WSNs. Especially, sensor nodes in WSN with limited computation ability, but it also brings some security challenges. To avoid the problems of continual key leakage attacks in the existing IBOOE
schemes, we formalize the notion and the security model for clr-IBOOE. Furthermore, we propose the first clr-secure IBOOE scheme based on three static assumptions in composite order groups. To the best of our knowledge, this is the first IBOOE resilient to continual key leakage. It achieves fully secure and continual leakage resilience against chosen plaintext attacks in the standard model. Compared with the schemes, performance analysis shows the proposed scheme has low computation overhead in the online encryption phase which is suitable for WSN with limited sensor nodes. As a direction of future work, designing secure IBOOE resilient to master secret key leakage would be another interesting work.

Author note
The author Xingbing Fu is now affiliated to Guangxi Key Laboratory of Cryptography and Information Security.

Acknowledgements
The authors would like to thank the reviewers for their detailed reviews and constructive comments, which have helped improve the quality of this article.

Declaration of conflicting interests
The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding
The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article.

References
1. Boneh D and Boyen X. Efficient selective-ID secure identity-based encryption without random oracles. In: Cachin C and Camenisch JL (eds) Advances in cryptology—EUROCRYPT, LNCS 3027. Berlin: Springer, 2004, pp.223–238.
2. Waters B. Efficient identity-based encryption without random oracles. In: Cramer R (ed.) Advances in Cryptology—EUROCRYPT, LNCS 3494. Berlin: Springer, 2005, pp.114–127.
3. Gentry C. Practical identity-based encryption without random oracles. In: Vaudenay S (ed.) Advances in Cryptology—EUROCRYPT. Berlin: Springer, 2006, pp.445–464.
4. Lewko A and Waters B. New techniques for dual system encryption and fully secure HIBE with short ciphertexts. In: Lewko A and Waters B (eds) Proceedings of the 7th theory of cryptography conference, LNCS 5978. Berlin: Springer, 2010, pp.455–479.
5. Naor M and Segev G. Public-key cryptosystems resilient to key leakage. SIAM J Comput 2012; 41(4): 772–814.
6. Akavia A, Goldwasser S and Vaikuntanathan V. Simultaneous hardcore bits and cryptography against memory attacks. In: Dwork C (ed.) Proceedings of the 6th theory of cryptography conference, LNCS 5444. Berlin: Springer, 2009, pp.474–495.
7. Alwen J, Dodis Y, Naor M, et al. Public-key encryption in the bounded-retrieval model. In: Advances in Cryptology—EUROCRYPT 2010, 29th annual international conference on the theory and applications of cryptographic techniques, French Riviera, 30 May–3 June 2010, pp.113–134. Berlin, Heidelberg: Springer.
8. Chow SSM, Dodis Y, Rouselakis Y, et al. Practical leakage-resilient identity-based encryption from simple assumptions. In: Proceedings of the 17th ACM conference on computer and communications security, Chicago, IL, 4–8 October 2010, pp.152–161. New York: ACM.
9. Li JG, Teng M, Zhang YC, et al. A leakage-resilient CCA-secure identity-based encryption scheme. Comput. J 2016; 59(7): 1066–1075.
10. Lewko A and Waters B. Achieving leakage resilience through dual system encryption. In: Proceedings of the 8th theory of cryptography conference. Providence, RI, 28–30 March 2011, pp.70–88. Berlin, Heidelberg: Springer.
11. Li JGYQH and ZhangYC. Identity-based broadcast encryption with continuous leakage resilience. Inf Sci 2018; 429: 177–193.
12. Zhou YW, Yang B and Mu Y. Continuous leakage-resilient identity-based encryption without random oracles. Comput J 2018; 61(4): 586–600.
13. Zhou YW, Yang B, Mu Y, et al. Continuous leakage-resilient access control for wireless sensor networks. Ad Hoc Netw 2018; 80: 41–53.
14. Zhou YW, Yang B and Mu Y. Continuous leakage-resilient identity-based encryption with leakage amplification. Des Codes Cryptogr 2019; 87(9): 2061–2090.
15. Zhang XJ, Wang HXX and Xu CX. Identity-based key-exposure resilient cloud storage public auditing scheme from lattices. Inform Sci 2019; 472: 223–234.
16. Yael TK and Leonid R. A survey of leakage-resilient cryptography. Prog Sound Found Cryptogr 2019; 727–794: 1–55.
Zhang et al.

17. Fu XB, Nie XY, Wu T, et al. Large universe attribute based access control with efficient decryption in cloud storage system. *J Syst Softw* 2018; 135: 157–164.

18. Guo FC, Mu Y and Chen ZD. Identity-based online/offline encryption. In: Tsudik G (ed.) *Financial cryptography and data security, LNCS 5143*. Berlin: Springer, 2008, pp.247–261.

19. Liu JK and Zhou JY. An efficient identity-based online/offline encryption scheme. In: Abdalla M, Pointcheval D, Fouque PA, et al. (eds) *Applied cryptography and network security, LNCS 5536*. Berlin: Springer, 2009, pp.156–167.

20. Chu CK, Liu JK, Zhou J, et al. Practical ID-based encryption for wireless sensor network. In: *Proceedings of the 5th ACM symposium on information, computer and communications security, ASIACCS'10*, Beijing, China, 13–16 April 2010, pp.337–340. New York: ACM.

21. Chow SSM, Liu JK and Zhou JY. Identity-based online/offline key encapsulation and encryption. In: *Proceedings of the 6th ACM symposium on information, computer and communications security*, Hong Kong, China, 22–24 March 2011, pp. 52–60. New York: ACM, 2011.

22. Lai JC, Mu Y, Guo FC, et al. Improved identity-based online/offline encryption. In: Foo E and Stebila D (eds) *Information security and privacy (20th Australasian Conference, LNCS 9144)*. Berlin: Springer, 2015, pp. 160–173.

23. Lai JC, Mu Y and Guo FC. Efficient identity-based online/offline encryption and signcryption with short ciphertext. *Int J Inform Secur* 2017; 16(3): 299–311.

24. Ma C, Xue K and Hong P. Distributed access control with adaptive privacy preserving property for wireless sensor networks. *Secur Commun Netw* 2014; 7(4): 759–773.