Continuous Leakage-resilient and Hierarchical Identity-based Online/Offline Encryption

Qihong Yu[1*], Jian Shen[2], Jin-Feng Lai[3], Sai Ji[1]

[1] College of Information Engineering, Suqian University, China
[2] School of Computer and Software, Nanjing University of Information Science and Technology, China
[3] Department of Engineering Science, National Cheng Kung University, Taiwan
yuqhsqu@163.com, s_shenjian@126.com, cinfon@ieee.com, jisai@squ.edu.cn

Abstract

By dividing encryption as online and offline stages, the online/offline encryption schemes are very suitable to lightweight equipment. For the offline stage, high-performance equipment is used for complex preprocessing calculation, and the online stage the lightweight devices only make some simple calculations. In addition, side channel attacks can disclose some secret information of the cryptosystem, which leads to the destruction of the security of the cryptography schemes. Most of the online/offline identity-based encryption schemes cannot resist side channel attacks. The paper proposes a concrete hierarchical identity-based and online/offline encryption scheme that can resist continuous leakage of secret key. By the dual system encryption technology, we prove that the given scheme is fully secure. Through key updation technology, our proposed scheme resists continual leakage of private key. The relative leakage rate of the private key can reach 1/3. In addition, the presented scheme has the hierarchical function which effectively solves the problem of heavy load in a single key generation center. The given scheme is suitable for applications in distributed environment.

Keywords: Continuous leakage attacks, Hierarchical encryption, Online/offline encryption, Key updation technology

1 Introduction

1.1 Related Work

Shamir [1] first presented the concept of identity-based encryption (IBE). IBE removes the certificate verification process for the traditional public key encryption mechanism and improves encryption efficiency. In IBE, users can express their identity information by using a string (for example, ID number, email address, etc.). Key generation center (KGC) uses this identity information and system master key to produce user's secret key. An encryptor uses a receiver's identity information and system public information to encrypt the plaintext.

For improving encryption efficiency in IBE, Guo et al. [2] gave an identity-based and online/offline encryption scheme (IBOOE). Guo et al. divided encryption operations as two parts: offline part and online part. For offline stage, most encryption operations are preprocessed to generate offline ciphertext. For online stage, very few simple operations are performed by using offline ciphertext to generate final ciphertext, which improves the efficiency of actual encryption. Subsequently, a series of efficient IBOOE [3-5] were proposed. For the attribute-based encryption schemes which are the extended ones of identity-based encryptions, Chen et al. [6] used an integrated access tree to improve the efficiency of ciphertext policy attribute-based encryption (CP-ABE) scheme. Li et al. [7] proposed a white-box efficient and traceable CP-ABE scheme with accountability for CloudIoT. Online and offline encryption technology is also widely used [8-9]. In order to decrease computation costs, Zhang et al. [10] proposed that most of decryption operations should be executed by the decryption cloud server provider (D-CSP). Online/offline technology is also used in the blockchain technology [11]. The existing IBOOE schemes do not consider the problem of key leakage.

In recent years, the side channel attacks [12-15] enable the enemy to get secret information by means of observing the timing and other characteristics of the operations of the cryptosystem, which bring about the leakage of relevant secret information of the cryptosystem. The leakage information undermines the security of the cryptosystem. Side channel attacks provide favorable conditions for adversaries to obtain private key information. Thus, the previous security model can not be used to solve the new problem. A new model must be provided. Leakage-resilient cryptography has come being because it succeeds in catching side channel attack.

In 2004, the paper [16] proposed the "only calculation leaks (OCL)" model which places restrictions that leakage only occurs in the visited part of the calculation process. The accessed part for each step of calculation is called active. The attacker selects a polynomial time function which is called leakage function and applies this function to these active states. He can obtain the bounded output of the function. It is assumed that the inaccessible part in the memory in the current calculation will not leak information. In this model, many practical schemes are designed, such as the leakage-resilient (LR) stream cipher [17] and the LR signature scheme [18]. The OCL model does not capture leakage in inactive parts of memory. In light of this problem, the paper [19] introduces the "bounded leakage-resilient (BLR) model", which is more applicable than the OCL model. In BLR settings, the inactive parts are allowed to give away information. In the BLR model, a large number of security schemes are constructed. For

*Corresponding Author: Qihong Yu; E-mail: yuqhsqu@163.com
DOI: 10.53106/160792642022112306011

example, the works [20-21] construct leakage-resilient encryption schemes.

As time goes on, the leakage amount will increase. The leakage may exceed the specified limit, and undermines the system safety. The BLR model cannot solve this problem. The "continuous leakage model (CLM) " solves this problem. The paper [22] proposed the "continuous leakage model". For CLM, the private keys are periodically updated. What's more, the leakage information between two updations cannot exceed the given upper bound. In another word, the amount about revealed key information for a period is limited, while the amount about revealed key information is unlimited for the whole execution about the system. References [23-24] give the security schemes under this model.

1.2 Motivation and Contribution

Waters [25] proposed dual system technology in which private key and ciphertext present two outward appearance: semi-functional appearance and normal appearance. A normal private key decrypts two kinds of ciphertext rightly. The semi-functional private key only decrypts the normal ciphertext. For the real scheme, ciphertext and private key present normal appearance. The security proof is finished by several games. For the first game, the ciphertext has semi-functional form. For those next games the private key presents a semi-functional form step by step. It must be proved that the attacker cannot detect this change. For the last game, every private key and ciphertext have semi-functional appearance. The attacker does not have the ability to decrypt them correctly. Reference [26] constructed identity based encryption schemes against leakage attacks. Zheng et al. [27] proposed a signature scheme through dual system technology. Chen et al. [28] presented a novel attribute based signature scheme by using the attribute tree as access policy and utilized server-aid technique to help the verifier to verify signatures and reduce the computation burden for resource-limited devices. For devices of Internet of Things, Li et al. [29] proposed a decentralized attribute-based server-aid signature scheme in which a server can help users execute heavy computation in the signature and verification algorithms. Shen and Yang et al. [30-32] propose several privacy protection methods for cloud data, and further point out that side channel attacks should be prevented. To increase the security and efficiency for cloud storage, Li et al. [33] gave an efficient identity-based provable multi-copy data possession in multi-cloud storage. Wang et al. [34-37] emphasize that efficiency is also critical for lightweight devices.

We present a fully secure hierarchical identity-based online/offline encryption (HIBOOE) scheme against continual leakage attacks which is very suitable for lightweight devices. The encryption operations are divided into two parts: offline part and online part. Offline encryption needs neither the plaintext nor the receiver's identity vector, performs the complex operation in the encryption operation, and stores some information as the offline ciphertext which needs to be kept secretly. Then, the online encryption can quickly generate ciphertext by performing some simple operation. For offline stage, the offline ciphertext is got from a high-performance external device and transmitted to the lightweight device. The online phase can be performed by lightweight devices. A private key is updated continuously, which ensures the continual leakage-resilience. The dual system encryption technology is used to prove the security. This presented scheme plays a good role for lightweight devices with weak computing power, and can meet the needs of security in practical applications.

2 Preliminaries

We give some notations in Table 1 and give the preliminaries used in our paper.

2.1 Bilinear Group with Composite Order

Bilinear group with composite order is introduced by Waters [25]. For an algorithm Φ, it inputs safety parameter λ and produces a description $\Omega = \{N = n_1 n_2 n_3, G, G^*, e\}$ about bilinear group with composite order, where n_1, n_2 and n_3 are different primes with λ bits length. G and G^* are cyclic groups with order N. Bilinear mapping e satisfies the flowing conditions.

(1) Bilinearity:$\forall h, p \in G, c, d \in Z_N, e(h^c, p^d) = e(h, p)^{cd}$.

(2) Non-degeneracy: We can find an element h in G such that $e(h, h) \neq 1$.

Table 1. Notations

Notation	Description
Φ	A bilinear group generation algorithm
Ω	Bilinear group description
G,G^*	Two cyclic groups with order N
$N = n_1 n_2 n_3$	The order of G
e	Bilinear mapping
G_{n_1},G_{n_2},G_{n_3}	Subgroups of G about order n_1, n_2 and n_3
λ	The safety parameter
X_1	Random value of G_{n_1}
X_2, Y_2, Z_2	Random values of G_{n_2}
X_3, Y_3	Random values of G_{n_3}
PP	The public parameters
MK	The master key
$PK_{\vec{I}}$	The private key of identity vector $\vec{I}$
$\widehat{PK}_{\vec{I}}$	The updated private key
CTM	The offline ciphertext
CTF	The final ciphertext
$\mathscr{B}$	A challenger
$\mathcal{A}$	An attacker
LK_{PK}	The upper bound about leakage of a private key

The operations about G and G^* are polynomial time efficient and computable with respect to safety parameters λ. G_{n_1}, G_{n_2} and G_{n_3} are used to represent subgroups of G about order n_1, n_2 and n_3 respectively. In particular, when $p_i \in G_{n_i}$and $p_j \in G_{n_j}$ $(i \neq j)$, $e(p_i, p_j)$ is the identity of G^*. For example, suppose $p_1 \in G_{n_1}$, $p_2 \in G_{n_2}$ and h is a generator about G, $h^{n_1 n_2}$ may generate G_{n_3}, $h^{n_1 n_3}$ may generate G_{n_2}, and $h^{n_2 n_3}$ may generate G_{n_1}. Therefore, there exists α_1, α_2 such that if $p_1 = (h^{n_2 n_3})^{\alpha_1}$ and $p_2 = (h^{n_1 n_3})^{\alpha_2}$, we can get that $e(p_1, p_2) = e(h^{n_2 n_3 \alpha_1}, h^{n_1 n_3 \alpha_2}) = e(h^{\alpha_1}, h^{n_3 \alpha_2})^{n_1 n_2 n_3} = 1$. G_{n_1}, G_{n_2} and $G_{n_3}$are orthogonal.

2.2 Difficult Assumptions

Three assumptions will be used to prove the safety of the proposed continual leakage-resilient and hierarchical identity-based online/offline encryption scheme (CLR-HIBOOE). The given assumption is static assumption on which the number of levels is independent of the number about private key inquiries from the attacker. The order of G is $N = n_1 n_2 n_3$, where n_1, n_2 and n_3 are different primes with λ bits length. For these following assumptions, we let $G_{n_1 n_2}$ represent subgroups of G with order $n_1 n_2$, and other situations are similar.

Assumption 1. It is also called subgroup decision problem of three primes. For the algorithm Φ which can generate bilinear group with composite order, the following distributions are given:

$$\Omega = (N = n_1 n_2 n_3, G, G^*, e) \xleftarrow{R} \Phi,$$
$$h \xleftarrow{R} G_{n_1}, X_3 \xleftarrow{R} G_{n_3},$$
$$D = (\Omega, h, X_3),\ T_1 \xleftarrow{R} G_{n_1 n_2}, T_2 \xleftarrow{R} G_{n_1}.$$

The advantages that algorithm $\mathcal{A}$ distinguishes T_1 from T_2 is denoted by $Adv1_{\Phi,\mathcal{A}}(\lambda) = |P[\mathcal{A}(D, T_1) = 1] - P[\mathcal{A}(D, T_2) = 1]|$. If the advantages $Adv1_{\Phi,\mathcal{A}}(\lambda)$ achieved by any algorithm $\mathcal{A}$ is ignorable, assumption 1 is valid.

If $T_1 = p_1 p_2$ where $p_i \in G_{n_i}$ $(i \in \{1,2\})$, we call p_1 and p_2 as the part in G_{n_1} and the part in G_{n_2} respectively.

Assumption 2. For the algorithm Φ which can generate bilinear group with composite order, the following distributions are given:

$$\Omega = (N = n_1 n_2 n_3, G, G^*, e) \xleftarrow{R} \Phi,$$
$$h, X_1 \xleftarrow{R} G_{n_1}, X_2, Y_2 \xleftarrow{R} G_{n_2}, X_3, Y_3 \xleftarrow{R} G_{n_3},$$
$$D = (\Omega, h, X_1 X_2, X_3, Y_2 Y_3),$$
$$T_1 \xleftarrow{R} G, T_2 \xleftarrow{R} G_{n_1 n_3}.$$

The advantages that algorithm $\mathcal{A}$ distinguishes T_1 from T_2 is denoted by $Adv2_{\Phi,\mathcal{A}}(\lambda) = |P[\mathcal{A}(D, T_1) = 1] - P[\mathcal{A}(D, T_2) = 1]|$. If the advantages $Adv2_{\Phi,\mathcal{A}}(\lambda)$ achieved by any algorithm $\mathcal{A}$ is ignorable, assumption 2 is valid.

Assumption 3. For the algorithm Φ which can generate bilinear group with composite order, the following distributions are given:

$$\Omega = (N = n_1 n_2 n_3, G, G^*, e) \xleftarrow{R} \Phi, \alpha, s \xleftarrow{R} Z_N,$$
$$h \xleftarrow{R} G_{n_1}, X_2, Y_2, Z_2 \xleftarrow{R} G_{n_2}, X_3 \xleftarrow{R} G_{n_3},$$
$$D = (\Omega, h, h^\alpha X_2, X_3, h^s Y_2, Z_2),$$
$$T_1 \xleftarrow{R} e(h,h)^{\alpha s}, T_2 \xleftarrow{R} G^*.$$

The advantages that algorithm $\mathcal{A}$ distinguishes T_1 from T_2 is denoted by $Adv3_{\Phi,\mathcal{A}}(\lambda) = |P[\mathcal{A}(D, T_1) = 1] - P[\mathcal{A}(D, T_2) = 1]|$. If the advantages $Adv3_{\Phi,\mathcal{A}}(\lambda)$ achieved by any algorithm $\mathcal{A}$ is ignorable, assumption 3 is valid.

3 Formal Description of CLR-HIBOOE

The proposed continual leakage-resilient and hierarchical identity-based online/offline encryption scheme is composed of these algorithms.

Start: This algorithm takes λ as input, and gives a master key $MK$and public parameter PP. $Start(\lambda) \to (PP, MK)$.
KeyG: This algorithm inputs public parameters PP, master key MK and identity vector $\vec{I}$, and outputs the private key $PK_{\vec{I}}$ for identity vector $\vec{I}$. $KeyG(PP, MK, \vec{I}) \to PK_{\vec{I}}$.
Delegation: This algorithm inputs a private key $PK_{\vec{I}}$ of identity vector $\vec{I}$ as well as identity ID_{j+1}, and outputs a private key $PK_{\vec{I}'}$ about identity vector $\vec{I}' = \vec{I}: ID_{j+1}$. $Delegation(PP, PK_{\vec{I}}, ID_{j+1}) \to PK_{\vec{I}'}$.
Updation: The algorithm inputs PP and $PK_{\vec{I}}$. It outputs the updated private key $\widehat{PK}_{\vec{I}}$. $Updation(PK, PK_{\vec{I}}) \to \widehat{PK}_{\vec{I}}$.
OfflineE: It inputs λ and PP. The algorithm produces offline ciphertext CTM. $OfflineE(\lambda, PP) \to CTM$.
OnlineE: The algorithm inputs λ, PP, plaintext M, offline ciphertext CTM and identity vector $\vec{I}$ and generates final ciphertext CTF. $OnlineE(\lambda, PP, M, CTM, \vec{I}) \to CTF$.
Decryption: The decryptor inputs final ciphertext CTF and private key $PK_{\vec{I}}$. It obtains the plaintext M. $Decryption(CTF, PK_{\vec{I}}) \to M$.

4 Security Semantics about Our CLR-HIBOOE

The security semantics about our **CLR-HIBOOE** is described with the help of this game Game_{Real} which is played between the attacker $\mathcal{A}$ and the challenger $\mathscr{B}$.

The proposed scheme **CLR-HIBOOE** is semantically secure against chosen plaintext attack. An attacker $\mathcal{A}$ may inquiry public parameters, private key and some leakage information with private key.

In the game Game_{Real}, the challenger $\mathscr{B}$ has one list $\mathscr{L} = \{(\mathcal{JN}, \mathcal{JD}, \mathcal{PK}, \mathscr{LK})\}$ which is composed of indicia, set of identity vector, private key and amount of leakage. $\mathcal{JN}$ is indicia's space. $\mathcal{PK}$ is private key's space. $\mathcal{JD}$ is space about identity vector. $\mathscr{LK}$ is the space for leakage amount. Suppose that $\mathcal{JN} = \mathbb{N}$ and $\mathscr{LK} = \mathbb{N}$. $\mathscr{B}$ has another list $\mathscr{R}$. This revealed identity vector will be recorded in $\mathscr{R}$.

A challenger $\mathscr{B}$ and an attacker $\mathcal{A}$ play the game Game_{Real}.

Game_{Real}:
Initialize: The challenger runs **Start** to get MK and PP: $Start(\lambda) \to$ （PP, MK）. $\mathscr{B}$ keeps this master key secretly and sends this public parameter to the attacker $\mathcal{A}$. An element $(0,0,0,0)$ is added in $\mathscr{L}$. The indicia in is 0.
Phase 1: The attacker $\mathcal{A}$ may do the queries as follows.

$O - Create(\vec{I})$: Given an identity vector $\vec{I}$, $\mathscr{B}$ looks up it within $\mathscr{L}$. If $\vec{I}$ exists within $\mathscr{L}$, it stops. If not, the challenger runs **KeyG** to obtain this secret key $PK_{\vec{I}}$ ($KeyG(PP, MK, \vec{I}) \to PK_{\vec{I}}$). Furthermore, $\mathscr{B}$ puts the element $(in + 1, \vec{I}, PK_{\vec{I}}, 0)$ in this list $\mathscr{L}$.

$O - Leak(in, fn)$: The attacker inquiries some leakage information of one private key for this indicia in. The attacker uses one function fn on this secret key and gets the outputs. The function is polynomial time computable.

If $(in, \vec{I}, PK_{\vec{I}}, LK)$ is not in $\mathscr{L}$, the challenger $\mathscr{B}$ does nothing. If $(in, \vec{I}, PK_{\vec{I}}, LK)$ is in $\mathscr{L}$, the challenger $\mathscr{B}$ judges whether $LK + |fn(PK_{\vec{I}})| \leq LK_{PK}$. LK_{PK} is the upper bound of leakage for this private key. If that's true, $\mathscr{B}$ sends $fn(PK_{\vec{I}})$ to the attacker and updates $(in, \vec{I}, PK_{\vec{I}}, LK)$ with $(in, \vec{I}, PK_{\vec{I}}, LK + |fn(PK_{\vec{I}})|)$ in $\mathscr{L}$. Otherwise, $\mathscr{B}$ returns $\perp$.

$O - Reveal(in)$: The attacker queries this private key for indicia in. If the element $(in, \vec{I}, PK_{\vec{I}}, LK)$ exists within $\mathscr{L}$, this challenger sends $PK_{\vec{I}}$ to the attacker and adds this identity vector $\vec{I}$ in $\mathscr{R}$. If it doesn't, $\mathscr{B}$ returns $\perp$.

$O - Updation(in)$: The attacker inquires about the updated private key about indicia in. This challenger judges if $(in, \vec{I}, PK_{\vec{I}}, LK)$ of indicia in belongs to $\mathscr{L}$. If it does, the challenger calls the algorithm **Updation** $Updation(PK, PK_{\vec{I}}) \rightarrow \widehat{PK}_{\vec{I}}$. The challenger gives the attacker the updated key $\widehat{PK}_{\vec{I}}$. Then, the challenger updates the item $(in, \vec{I}, PK_{\vec{I}}, LK)$ with $(in, \vec{I}, \widehat{PK}_{\vec{I}}, 0)$.

$O - Delegation(in, PK_{\vec{I}}, \vec{I})$: The attacker gives a private key $PK_{\vec{I}}$ of identity vector $\vec{I}$ with depth j as well as identity ID_{j+1}. The challenger finds whether the item $(in, \vec{I}, PK_{\vec{I}}, LK)$ is in $\mathscr{L}$. If it is true, the challenger runs **Delegation** to produce the private key $PK_{\vec{I}:ID_{j+1}}$ for $\vec{I}:ID_{j+1}$. An item $(in+1, \vec{I}:ID_{j+1}, PK_{\vec{I}:ID_{j+1}}, 0)$ is added in the list $\mathscr{L}$.

Challenge: The attacker $\mathcal{A}$ gives two messages M_0, M_1 and an identity vector $\vec{I}^*$. The constraint is that $\vec{I}^*$ is not in $\mathscr{R}$. $\mathscr{B}$ chooses randomly $\beta \leftarrow \{0,1\}$ and generates ciphertext CTF about M_β. $\mathscr{B}$ sends CTF to that attacker.

Phase 2: $\mathcal{A}$ may query the oracles $O - Create(\vec{I})$, $O - Delegation(in, PK_{\vec{I}}, \vec{I})$ and $O - Reveal(in)$. As with Phase 1, these same limitations are required. What is more, the delegated identity vectors are not in $\mathscr{R}$.

Guess: The attacker $\mathcal{A}$ gives the guess $\beta' \in \{0,1\}$. If $\beta' = \beta$, the attacker $\mathcal{A}$ wins Game$_{Real}$. The attacker $\mathcal{A}$ obtains these advantages $Adv_{\mathcal{A}}(LK_{PK}) = \left|P[\beta' = \beta] - \frac{1}{2}\right|$.

If the advantage that any PPT attacker $\mathcal{A}$ can win in Game$_{Real}$ is very little (ignorable), our **CLR-HIBOOE** is LK_{PK} leakage-resilient.

5 Construction of CLR-HIBOOE

Our scheme is constructed by bilinear group with composite order. The private key is randomized by G_{n_3}. G_{n_2} is not used in real systems, but it is used in a semi-functional form.

Setup: The algorithm run bilinear group generation algorithm Φ to obtain a bilinear group G with order $N = n_1 n_2 n_3$. Suppose that ℓ symbols the maximum depth for our **CLR-HIBOOE**. This algorithm randomly selects $g_1, h_1, u_1, \ldots, u_l \in G_{n_1}$, $X_3 \in G_{n_3}, \alpha \in Z_N$ and $x_1, x_2, \ldots, x_n \in Z_N$. It issues the public key $PP = \{N, g_1, h_1, u_1, \ldots, u_l, X_3, e(g_1, g_1)^\alpha, g_1^{x_1}, g_1^{x_2}, \ldots, g_1^{x_n}\}$ and keeps this master key $MK = \{\alpha\}$ as secret.

KeyG: The private key generator takes PP, MK and identity vector $\vec{I} = (ID_1, \ldots, ID_j)$ as input. It randomly selects $r, y_1, \ldots, y_n \in Z_N$ and $R_{0,1}, \ldots, R_{0,n}, R_3, R_3', R_{j+1}, \ldots, R_l$ of G_{n_3}. It generates the private key as follows.

$$\overrightarrow{K_0} = (g_1^{y_1} R_{0,1}, g_1^{y_2} R_{0,2}, \ldots, g_1^{y_n} R_{0,n}),$$
$$K_1 = g_1^r R_3,$$
$$K_2 = g_1^\alpha \prod_{i=1}^n g_1^{-x_i y_i} (u_1^{ID_1} \ldots u_j^{ID_j} h_1)^r R_3',$$
$$E_{j+1} = u_{j+1}^r R_{j+1}, \ldots, E_l = u_l^r R_l.$$

Delegation: On input the private key $PK_{\vec{I}}$ of an identity vector $\vec{I}$ about j^{th} level and one identity ID_{j+1}, the algorithm produces this private key $PK_{\vec{I'}}$ for the identity vector $\vec{I'}(\vec{I}:ID_{j+1})$ of $(j+1)^{th}$ level. It randomly selects $r_1 \in Z_N$ and $R'_{0,1}, \ldots, R'_{0,n}, R_{3,1}, R'_{3,1}, R_{j+2,1}, \ldots, R_{l,1}$ of G_{n_3}. It generates the delegated private key as follows.

$$\overrightarrow{K_0} = \left(g_1^{y_1} R'_{0,1}, g_1^{y_2} R'_{0,2}, \ldots, g_1^{y_n} R'_{0,n}\right),$$
$$K'_1 = K_1 g_1^{r_1} R_{3,1},$$
$$K'_2 = K_2 E_{j+1}^{ID_{j+1}} (u_1^{ID_1} \ldots u_j^{ID_j} u_{j+1}^{ID_{j+1}} h_1)^{r_1} R'_{3,1},$$
$$E'_{j+2} = E_{j+2} u_{j+2}^{r_1} R_{j+2,1}, \ldots, E'_l = E_l u_l^{r_1} R_{l,1}.$$

The new key is completely randomized and the only connection with the previous key is the identity vector $\vec{I} = (ID_1, \ldots, ID_j)$.

Updation: The algorithm inputs PP and the private key $PK_{\vec{I}}$ for an identity vector $\vec{I}$ and outputs a new private key $\widehat{PK}_{\vec{I}}$ for $\vec{I}$.

Given $PK_{\vec{I}} = (\overrightarrow{K_0}, K_1, K_2, E_{j+1}, \ldots, E_l)$, the algorithm selects randomly $\Delta r, \Delta y_1, \ldots, \Delta y_n \in Z_N$ and $\Delta R_{0,1}, \ldots, \Delta R_{0,n}, \Delta R_3, \Delta R_3', \Delta R_{j+1}, \ldots, \Delta R_l \in G_{n_3}$ and computes

$$\widehat{\overrightarrow{K_0}} = (g_1^{y_1 + \Delta y_1}(R_{0,1} + \Delta R_{0,1}),$$
$$g_1^{y_2 + \Delta y_2} R_{0,2}, \ldots, g_1^{y_n + \Delta y_n}(R_{0,n} + \Delta R_{0,n})),$$
$$\widehat{K_1} = g_1^{r + \Delta r}(R_3 + \Delta R_3),$$
$$\widehat{K_2} = g_1^\alpha \prod_{i=1}^n g_1^{-x_i(y_i + \Delta y_i)} (u_1^{ID_1} \ldots u_j^{ID_j} h_1)^{r + \Delta r}(R_3' + \Delta R_3'),$$
$$\widehat{E_{j+1}} = u_{j+1}^{r+\Delta r}(R_{j+1} + \Delta R_{j+1}), \ldots, \widehat{E_l} = u_l^{r+\Delta r}(R_l + \Delta R_l).$$

The new private key is $\widehat{PK}_{\vec{I}} = (\widehat{\overrightarrow{K_0}}, \widehat{K_1}, \widehat{K_2}, \widehat{E_{j+1}}, \ldots, \widehat{E_l})$. Because $\Delta r, \Delta y_1, \ldots, \Delta y_n \in Z_N$, $t_j \in Z_N$, $\Delta t_j \in Z_N$ and $\Delta R_{0,1}, \ldots, \Delta R_{0,n}, \Delta R_3, \Delta R_3', \Delta R_{j+1}, \ldots, \Delta R_l \in G_{n_3}$ are random, $y_i + \Delta y_i (i = 1, \ldots, n)$, $r + \Delta r$, $R_{0,i} + \Delta R_{0,i} (i = 1, \ldots, n)$, $R_3 + \Delta R_3$, $R_3' + \Delta R_3'$ and $R_{j+1} + \Delta R_{j+1}, \ldots, R_l + \Delta R_l$ are all random. Essentially, we add only extra random values to the old ones in the private key. So, the private key $\widehat{PK}_{\vec{I}}$ and $PK_{\vec{I}}$ have the same distribution**.**

OfflineE: On input λ and PP, the algorithm gives the indirect ciphertext CTM. It randomly selects $z_1, z_2, \ldots z_l, s, t \in Z_N$, and calculates

$$(\overrightarrow{C_0}) = ((g_1^{x1})^s, (g_1^{x2})^s, \ldots, (g_1^{xn})^s$$
$$R = e(g_1, g_1)^{\alpha s}), C_1 = (u_1^{z1} \ldots u_l^{zl} h_1)^s,$$

$$C_2 = g_1^s, C_{3,1} = u_1^{st}, C_{3,2} = u_2^{st}, \dots, C_{3,l} = u_l^{st},$$

The indirect ciphertext is $CTM = (\overrightarrow{C_0}, R, C_1, C_2, C_{3,1}, \dots, C_{3,l}, z_1, \dots, z_l, t)$.

OnlineE: On input λ, M, PP, CTM and $\vec{I}$, the algorithm computes as follows.

$$t_1 = t^{-1}(ID_1 - z_1) \bmod N,$$
$$t_2 = t^{-1}(ID_2 - z_2) \bmod N,$$
$$\dots, t_j = t^{-1}(ID_j - z_j) \bmod N,$$
$$t_{j+1} = -t^{-1}z_{j+1} \bmod N,$$
$$\dots, t_l = -t^{-1}z_l \bmod N,$$
$$C_4 = R \oplus M$$

The ultimate ciphertext is $CTF = (\overrightarrow{C_0}, C_1, C_2, C_{3,1}, \dots, C_{3,l}, C_4, t_1, \dots, t_l)$.

Decryption: On input CTF and $PK_{\vec{I}}$, the algorithm obtains the message M when this identity vector of CTF and that of $PK_{\vec{I}}$ are same.

First, the decryption algorithm obtains the blinding factor.

$$e(\overrightarrow{K_0}, \overrightarrow{C_0})$$
$$= e(< g_1^{y_1}R_{0,1}, g_1^{y_2}R_{0,2}, \dots, g_1^{y_n}R_{0,n} >,$$
$$< (g_1^{x_1})^s, (g_1^{x_2})^s, \dots, (g_1^{x_n})^s >)$$
$$e(K_2, C_2)$$
$$= e(g_1^{\alpha} \prod_{i=1}^n g_1^{-x_i y_i} \left(u_1^{ID_1} \dots u_j^{ID_j} h_1\right)^r R_3', g_1^s\right)$$
$$e(K_1, C_1 C_{3,1}^{t_1} \dots C_{3,l}^{t_l})$$
$$= e(g_1^r R_3, (u_1^{z_1} \dots u_l^{z_l} h_1)^s (u_1^{st})^{t^{-1}(ID_1 - z_1)}$$
$$\dots (u_j^{st})^{t^{-1}(ID_j - z_j)} (u_{j+1}^{st})^{-t^{-1}z_{j+1}} \dots (u_l^{st})^{-t^{-1}z_l})$$
$$\frac{e(\overrightarrow{K_0}, \overrightarrow{C_0}) e(K_2, C_2)}{e(K_1, C_1 C_{3,1}^{t_1} \dots C_{3,l}^{t_l})} = e(g_1, g_1)^{\alpha s} = R$$

Then, he calculates $R \oplus C_4 = R \oplus (R \oplus M) = M$.

6 Security of CLR-HIBOOE

The proof depends on three static assumptions given in subsection 2.2. For proving this security, we employ the method introduced in reference [25] to construct additional semi-functional ciphertext and key which are only used for proof.

Semi-functional ciphertext. Based on a generator g_2 of G_{n_2} and the normal ciphertext $CTF = (\overrightarrow{C_0}, C_1, C_2, C_{3,1}, \dots, C_{3,l}, C_4, t_1, \dots, t_l)$, the algorithm randomly selects $v, \gamma_1, \dots, \gamma_n, \eta_1, \dots, \eta_l, z_c \in Z_N$ and further generates the semi-functional ciphertext.

$$\widetilde{\overrightarrow{C_0}} = ((g_1^{x_1})^s (g_2^v)^{\gamma_1}, (g_1^{x_2})^s (g_2^v)^{\gamma_2}, \dots, (g_1^{x_n})^s (g_2^v)^{\gamma_n})$$
$$R = e(g_1, g_1)^{\alpha s},$$
$$\widetilde{C_1} = (u_1^{z_1} \dots u_l^{z_l} h_1)^s (g_2^v)^{z_c},$$
$$\widetilde{C_2} = g_1^s g_2^v,$$
$$\widetilde{C_{3,1}} = u_1^{st} (g_2^v)^{\eta_1},$$
$$\widetilde{C_{3,2}} = u_2^{st} (g_2^v)^{\eta_2}, \dots, \widetilde{C_{3,l}} = u_l^{st} (g_2^v)^{\eta_l},$$
$$t_1 = t^{-1}(ID_1 - z_1) \bmod N,$$
$$t_2 = t^{-1}(ID_2 - z_2) \bmod N, \dots$$
$$t_j = t^{-1}(ID_j - z_j) \bmod N,$$
$$t_{j+1} = -t^{-1}z_{j+1} \bmod N, \dots$$
$$t_l = -t^{-1}z_l \bmod N,$$
$$C_4 = R \oplus M$$

Semi-functional private key. Based on the normal private key $\overrightarrow{K_0}, K_1, K_2, E_{j+1}, \dots, E_l$, the algorithm randomly selects $w, \xi_1, \dots, \xi_n, \zeta_{j+1}, \dots, \zeta_l, z_k \in Z_N$ and further generates the semi-functional private key.

$$\widetilde{\overrightarrow{K_0}} = (g_1^{y_1}.g_2^{w\xi_1}.R_{0,1}, g_1^{y_2}.g_2^{w\xi_2}.R_{0,2}, \dots, g_1^{y_n}.g_2^{w\xi_n}.R_{0,n}),$$
$$\widetilde{K_1} = K_1.g_2^w = g_1^r.g_2^w.R_3,$$
$$\widetilde{K_2} = K_2.g_2^{wz_k} = g_1^{\alpha} \prod_{i=1}^n g_1^{-x_i y_i} (u_1^{ID_1} \dots u_j^{ID_j} h_1)^r.g_2^{wz_k}.R_3',$$
$$\widetilde{E_{j+1}} = E_{j+1}.g_2^{w\zeta_{j+1}} = u_{j+1}^r.g_2^{w\zeta_{j+1}}.R_{j+1}, \dots,$$
$$\widetilde{E_l} = E_l.g_2^{w\zeta_l} = u_l^r.g_2^{w\zeta_l}.R_l.$$

The normal key decrypts correctly not only the normal ciphertexts but also the semi-functional ones. The semi-functional key decrypts only the normal ciphertexts correctly. If a semi-functional key decrypts a semi-functional ciphertext, we have

$$e(\widetilde{\overrightarrow{K_0}}, \widetilde{\overrightarrow{C_0}})$$
$$= e(< g_1^{y_1}.g_2^{w\xi_1}.R_{0,1}, g_1^{y_2}.g_2^{w\xi_2}.R_{0,2}, \dots, g_1^{y_n}.g_2^{w\xi_n}.R_{0,n} >,$$
$$< (g_1^{x_1})^s (g_2^v)^{\gamma_1}, (g_1^{x_2})^s (g_2^v)^{\gamma_2}, \dots, (g_1^{x_n})^s (g_2^v)^{\gamma_n} >)$$
$$= \prod_{i=1}^n e(g_2^{w\xi_i}, g_2^{v\eta_i}) \prod_{i=1}^n e(g_1^{y_1}, (g_1^{x_1})^s)$$
$$e(\widetilde{K_2}, \widetilde{C_2})$$
$$= e(g_1^{\alpha} \prod_{i=1}^n g_1^{-x_i y_i} (u_1^{ID_1} \dots u_j^{ID_j} h_1)^r.g_2^{wz_k}.R_3', g_1^s g_2^v)$$
$$= e(\prod_{i=1}^n g_1^{-x_i y_i}, g_1^s) e((u_1^{ID_1} \dots u_j^{ID_j} h_1)^r, g_1^s).$$
$$e(g_1^{\alpha}, g_1^s) e(g_2^{wz_k}, g_2^v)$$
$$\frac{e(\widetilde{\overrightarrow{K_0}}, \widetilde{\overrightarrow{C_0}}) e(\widetilde{K_2}, \widetilde{C_2})}{e(\widetilde{K_1}, \widetilde{C_1} (\widetilde{C_{3,1}})^{t_1} \dots (\widetilde{C_{3,l}})^{t_l})}$$
$$= R.e(g_2, g_2)^{\varpi}$$

where,

$$\varpi = (\textstyle\sum_{i=1}^n wv\eta_i\xi_i) + wvz_k - wvz_c - wv\eta_1(ID_1 - z_1) - \dots - wv\eta_j(ID_j - z_j) + wv\eta_{j+1}z_{j+1} + \dots + wv\eta_l z_l.$$

There will be an additional item $e(g_2, g_2)^{\varpi}$. If

$$wvz_k = wvz_c + wv\eta_1(ID_1 - z_1) + \dots + wv\eta_j(ID_j - z_j) - wv\eta_{j+1}z_{j+1} - \dots - wv\eta_l z_l - (\textstyle\sum_{i=1}^n wv\eta_i\xi_i),$$

we call the private key as nominal semi-functional private key.

The security proof of our scheme is obtained by constructing several games. We use q to indicate the number of private key queries.

Game$_{Real}$. This game is played by the challenger $\mathscr{B}$ and the attacker $\mathcal{A}$, in which these private keys are produced through this delegation algorithm.

Game$_{Real'}$. The only difference from Game$_{Real}$ is that the private is generated by private key generation algorithm.

Game$_{Restricted}$. Compared with the Game$_{Real'}$, the attacker cannot inquiry the identity vectors that are the prefix of this given challenge identity modulo p_2.

$Game_i$ ($i \in [0, q]$): Similar to $\mathrm{Game}_{Restricted}$, the ciphertext is semi-functional. For these i key enquiries at the head the challenger produces semi-functional ones. For the remaining key enquiries the challenger gives the normal ones.

$Game_{Final}$: Compared with $Game_q$, the only difference is that this ciphertext is obtained by encrypting a random message.

Six upcoming lemmas are contributed to the completion of Theorem 1.

Lemma 1. The total leakage amount of our proposed scheme is near to $LK_{PK} = (n - 2\vartheta - 1)\lambda$.

Proof. A conclusion in the work [22] helps to complete this lemma 1.

Conclusion 1. For a prime p , $d_1 \geq d_2 \geq 2$ $(d_1, d_2 \in N)$, $X \leftarrow Z_p^{d_1 \times d_2}$, $Y \leftarrow Rk_1(Z_p^{d_2 \times 1})$ and $\Gamma \leftarrow Z_p^{d_1}$, if fn is leakage function over $Z_p^{d_1}$ to W $(fn: Z_p^{d_1} \rightarrow W)$ where $|W| \leq 4 \cdot (1 - \frac{1}{p}) \cdot p^{d_2 - 1} \cdot \varepsilon^2$, this statistical distance $SD((X, fn(X \cdot Y)), (X, fn(\Gamma)) \leq \varepsilon$. ε is negligible.

From conclusion 1, the following Corollary 1 is obtained easily.

Corollary 1. For $d_1 \geq 3$ and a prime p, we choose $\vec{\delta} \leftarrow Z_p^{d_1}$, $\vec{\tau} \leftarrow Z_p^{d_1}$ and $\vec{\tau}' \leftarrow Z_p^{d_1}$ on the condition that $\vec{\tau}'$ is orthogonal to $\vec{\delta}$ modulo p . For leakage function $f: Z_p^{d_1} \rightarrow W$, if $|W| \leq 4 \cdot (1 - \frac{1}{p}) \cdot p^{d_1 - 2} \cdot \varepsilon^2$, $SD((\vec{\delta}, fn(\vec{\tau}')), (\vec{\delta}, fn(\vec{\tau}))) \leq \varepsilon$.

Proof. Based on conclusion 1, let $d_2 = d_1 - 1$. Therefore, $\vec{\tau}$ matches Γ. This basis for that orthogonal space about $\vec{\delta}$ matches X . Thus, $\vec{\tau}'$ and $X \cdot Y$ have the same distributions where $Y \leftarrow Rk_1(Z_p^{(d_1 - 1) \times 1})$ and $X \leftarrow Z_p^{d_1 \times (d_1 - 1)}$. So, $SD((\vec{\delta}, fn(\vec{\tau}')), (\vec{\delta}, fn(\vec{\tau}))) = SD((X, fn(X \cdot T)), (X, fn(\Gamma))$.

In consideration that $n = d_1 - 1$, $n_2 = p$ and $\varepsilon = n_2^{-\vartheta}$, it is concluded that the leakage information amounts to $log|W| \leq (n-1)\, logn_2 - 2\vartheta logn_2 = (n - 2\vartheta - 1) logn_2$ = $(n - 2\vartheta - 1)\lambda$, where $logn_2 = \lambda$. Consequently, the leakage adds up to $LK_{PK} = (n - 2\vartheta - 1)\lambda$.

Lemma 2. Given $LK_{PK} = (n - 2\vartheta - 1)\lambda$, any attacker $\mathcal{A}$ can only gain the same advantages in Game_{Real} or $\mathrm{Game}_{Real'}$. That is to say, $Game_{Real} Adv_{\mathcal{A}} = Game_{Real'} Adv_{\mathcal{A}}$.

The attacker $\mathcal{A}$ wins the advantage $Game_{Real} Adv_{\mathcal{A}}$ in Game_{Real} . The attacker $\mathcal{A}$ wins the advantage $Game_{Real'} Adv_{\mathcal{A}}$ in $\mathrm{Game}_{Real'}$.

Proof. No matter whether the key is generated by **KeyG** or by **Delegation**, their distributions are exactly same. For the attacker, they are not fundamentally different.

Lemma 3. Given $LK_{PK} = (n - 2\vartheta - 1)\lambda$, if an attacker $\mathcal{A}$ may make a distinction between $\mathrm{Game}_{Restricted}$ and $\mathrm{Game}_{Real'}$ in advantage ε , i.e. $Game_{Real'} Adv_{\mathcal{A}} - \mathrm{Game}_{Restricted} Adv_{\mathcal{A}} = \varepsilon$, there exists an algorithm $\mathscr{B}$ who can broke assumption 2 with advantage ε.

Proof. In consideration of g_1, X_1X_2, X_3, Y_2Y_3 and T , $\mathscr{B}$ plays the game $\mathrm{Game}_{Real'}$ with $\mathcal{A}$. $\mathcal{A}$ may give identity vector $\vec{I} = (ID_1, ID_2, \ldots, ID_j)$ and $\vec{I}^* = (ID_1^*, ID_2^*, \ldots, ID_j^*)$ in probability ε such that for any $k \leq j$, $ID_k \neq ID_k^* \mod N$. $\mathscr{B}$ computes $a = gcd(ID_1 - ID_1^*, N)$ and obtains a nontrivial factor $b = \frac{N}{a}$ of N . Three possibilities are considered.

① $a = n_1$ and $b = n_2 n_3$, or vice versa.
② $a = n_3$ and $b = n_1 n_2$, or vice versa.
③ $a = n_2$ and $b = n_1 n_3$, or vice versa.

Case 1. $\mathscr{B}$ can determine that either of a and b is n_3 by testing that either of $(Y_2Y_3)^a$ and $(Y_2Y_3)^b$ is equal to identity. It may be assumed that $a = n_1$ and $b = n_2 n_3$. Afterwards, $\mathscr{B}$ can determine whether T contains some component of G_{n_2} by differentiating whether $e(T^a, X_1X_2)$ is equivalent to identity. If it is not, T contains the component of G_{n_2}. Otherwise, $T \in G_{n_1}$.

Case 2. $\mathscr{B}$ can determine that either of a and b is n_3 by testing that either of $(X_1X_2)^a$ and $(X_1X_2)^b$ is equal to identity. It may be assumed that $a = n_3$ and $b = n_1 n_2$. Afterwards, $\mathscr{B}$ can determine whether T contains some component of G_{n_2} by judging whether $e(T^a, X_1X_2)$ is equivalent to identity. If it is true, $T \in G_{n_1}$. Otherwise, T contains the component of G_{n_2}.

If it does not satisfy case 1 or case 2, it satisfies case 3. We can judge that either of a and b is n_2 by judging that either of X_3^a and X_3^b is the identity element. It may be assumed that $a = n_2$ and $b = n_1 n_3$. Afterwards, $\mathscr{B}$ can determine whether T contains some component of G_{n_2} by judging whether T^a is an identity element. If not, T contains the component of G_{n_2}. Otherwise, $T \in G_{n_1}$.

Therefore, if algorithm $\mathcal{A}$ can distinguish $\mathrm{Game}_{Restricted}$ from $\mathrm{Game}_{Real'}$ by advantage ε, algorithm $\mathscr{B}$ destroys assumption 2 by advantage ε.

Lemma 4. Given $LK_{PK} = (n - 2\vartheta - 1)\lambda$, if an attacker $\mathcal{A}$ may make a distinction between $\mathrm{Game}_{Restricted}$ and Game_0 in advantage ε , i.e. $\mathrm{Game}_{Restricted} Adv_{\mathcal{A}} - Game_0\ Adv_{\mathcal{A}} = \varepsilon$, there exists an algorithm $\mathscr{B}$ who can broke assumption 1 with advantage ε .

Proof. In consideration of g_1, X_3 and T, $\mathscr{B}$ plays the game $\mathrm{Game}_{Restricted}$ or $Game_0$ with $\mathcal{A}$. $\mathscr{B}$ randomly selects $a, a_1, \ldots, a_l, b, x_1, \ldots, x_n \in Z_N$ and sets $u_1 = g_1^{a_1}, \ldots, u_l = g_l^{a_l}$, $h_1 = g_1^b$ and $g_1^{x_1}, \ldots, g_1^{x_n}$. $\mathscr{B}$ transmits public parameter $\{N, g_1, h_1, u_1, \ldots, u_l, e(g_1, g_1)^\alpha, g_1^{x_1}, \ldots, g_1^{x_n}\}$ to $\mathcal{A}$. $\mathscr{B}$ keeps the master key $MK = \{a\}$ as a secret.

When $\mathcal{A}$ provides an identity vector $\vec{I} = (ID_1, \ldots, ID_j)$, $\mathscr{B}$ randomly generates $r, y_1, \ldots, y_n \in Z_N$ and $R_{0,1}, \ldots, R_{0,n}, R_3, R_3', R_{j+1}, \ldots, R_l$ in G_{n_3} . $\mathscr{B}$ produces the private key:

$$\overrightarrow{K_0} = \left(g_1^{y_1} R_{0,1}, g_1^{y_2} R_{0,2}, \ldots, g_1^{y_n} R_{0,n}\right),$$
$$K_1 = g_1^r R_3,$$
$$K_2 = g_1^\alpha \prod_{i=1}^n g_1^{-x_i y_i} (u_1^{ID_1} \ldots u_j^{ID_j} h_1)^r R_3',$$
$$E_{j+1} = u_{j+1}^r R_{j+1}, \ldots, E_l = u_l^r R_l.$$

Challenge phrase: The attacker $\mathcal{A}$ gives two plaintexts M_0, M_1 and an identity vector $\vec{I}^*$. The constraint is that $\vec{I}^*$ is not in $\mathscr{R}$. $\mathscr{B}$ randomly chooses $\beta \leftarrow \{0,1\}$ and encrypts M_β. $\mathscr{B}$ sends the ciphertext CTF to the attacker.

$\mathscr{B}$ chooses $z_1, z_2, \ldots z_l, t \in Z_N$ randomly, and computes:

$$\overrightarrow{C_0} = (T^{x_1}, T^{x_2}, \ldots, T^{x_n})$$
$$R = e(g_1, T)^{\alpha},$$
$$C_1 = T^{a_1 z_1 + \ldots + a_l z_l + b},$$
$$C_2 = T, C_{3,1} = T^{a_1 t},$$
$$C_{3,2} = T^{a_2 t}, \ldots, C_{3,l} = T^{a_l t},$$

Then, $\mathscr{B}$ calculates

$$t_1 = t^{-1}(ID_1^* - z_1) \bmod N,$$
$$t_2 = t^{-1}(ID_2^* - z_2) \bmod N, \ldots,$$
$$t_j = t^{-1}(ID_j^* - z_j) \bmod N,$$
$$t_{j+1} = -t^{-1} z_{j+1} \bmod N, \ldots,$$
$$t_l = -t^{-1} z_l \bmod N,$$
$$C_4 = R \oplus M_\beta$$
$$CTF = (\overrightarrow{C_0}, C_1, C_2, C_{3,1}, \ldots, C_{3,l}, C_4, t_1, \ldots, t_l).$$

Phase 2: This phase and phase 1 are similar.

If $T \in G_{n_1}$, we suppose that $T = g_1^s$. The ciphertext $CTF = (\overrightarrow{C_0}, C_1, C_2, C_{3,1}, \ldots, C_{3,l}, C_4, t_1, \ldots, t_l)$ has the following form.

$$\overrightarrow{C_0} = ((g_1^{x_1})^s, (g_1^{x_2})^s, \ldots, (g_1^{x_n})^s)$$
$$R = e(g_1, g_1)^{\alpha s},$$
$$C_1 = (u_1^{z_1} \ldots u_l^{z_l} h_1)^s,$$
$$C_2 = g_1^s, C_{3,1} = u_1^{st},$$
$$C_{3,l} = u_2^{st}, \ldots, C_{3,l} = u_l^{st},$$

where,

$$t_1 = t^{-1}(ID_1^* - z_1) \bmod N,$$
$$t_2 = t^{-1}(ID_2^* - z_2) \bmod N, \ldots,$$
$$t_j = t^{-1}(ID_j^* - z_j) \bmod N,$$
$$t_{j+1} = -t^{-1} z_{j+1} \bmod N, \ldots,$$
$$t_l = -t^{-1} z_l \bmod N,$$
$$C_4 = R \oplus M_\beta.$$

Obviously, no the component of G_{n_2} is in T. This is a normal ciphertext. $\mathcal{A}$ simulates $\text{Game}_{Restricted}$.

If $T \in G_{n_1 n_2}$, we suppose that $T = g_1^s g_2^v$. The ciphertext $CTF = (\overrightarrow{C_0}, C_1, C_2, C_{3,1}, \ldots, C_{3,l}, C_4, t_1, \ldots, t_l)$ has the following form.

$$\widetilde{\overrightarrow{C_0}} = (g_1^{x_1})^s (g_2^v)^{x_1}, (g_1^{x_2})^s (g_2^v)^{x_2}, \ldots, (g_1^{x_n})^s (g_2^v)^{x_n}$$
$$R = e(g_1, g_1)^{\alpha s},$$
$$\widetilde{C_1} = (g_1^s g_2^v)^{a_1 z_1 + \ldots + a_l z_l + b} = (u_1^{z_1} \ldots u_l^{z_l} h_1)^s (g_2^v)^{z_c},$$
$$\widetilde{C_2} = g_1^s g_2^v,$$
$$\widetilde{C_{3,1}} = (g_1^s g_2^v)^{a_1 t} = u_1^{st} (g_2^v)^{\eta_1},$$
$$\widetilde{C_{3,2}} = (g_1^s g_2^v)^{a_2 t} = u_2^{st} (g_2^v)^{\eta_2}, \ldots,$$
$$\widetilde{C_{3,l}} = (g_1^s g_2^v)^{a_l t} = u_l^{st} (g_2^v)^{\eta_l},$$
$$t_1 = t^{-1}(ID_1 - z_1) \bmod N,$$
$$t_2 = t^{-1}(ID_2 - z_2) \bmod N, \ldots,$$
$$t_j = t^{-1}(ID_j - z_j) \bmod N,$$
$$t_{j+1} = -t^{-1} z_{j+1} \bmod N, \ldots,$$
$$t_l = -t^{-1} z_l \bmod N,$$
$$C_4 = R \oplus M.$$

Where $z_c = a_1 z_1 + \ldots + a_l z_l + b, \eta_1 = a_1 t, \eta_2 = a_2 t, \ldots, \eta_l = a_l t$. $\mathcal{A}$ simulates $Game_0$. Hence, algorithm $\mathscr{B}$ destroys assumption 1 by advantage $\mathcal{E}$.

Lemma 5: Given $LK_{PK} = (n - 2\vartheta - 1)\lambda$, if an attacker $\mathcal{A}$ may make a distinction between Game_{k-1} and Game_k in advantage $\mathcal{E}$, i.e. $\text{Game}_{k-1} Adv_{\mathcal{A}} - Game_k\, Adv_{\mathcal{A}} = \varepsilon$, there exists an algorithm $\mathscr{B}$ who can broke assumption 2 with advantage ε.

Proof. In consideration of $g_1, X_1 X_2, X_3, Y_2 Y_3$ and T, $\mathscr{B}$ gets $a, a_1, \ldots, a_l, b, x_1, \ldots, x_n \in Z_N$ through a random process and sets $u_1 = g_1^{a_1}, \ldots, u_l = g_l^{a_l}$, $h_1 = g_1^b$ and $g_1^{x_1}, \ldots, g_1^{x_n}$. $\mathscr{B}$ transmits public parameter $\{N, g_1, h_1, u_1, \ldots, u_l, e(g_1, g_1)^{\alpha}, g_1^{x_1}, \ldots, g_1^{x_n}\}$ to $\mathcal{A}$. $\mathscr{B}$ keeps the master key $MK = \{a\}$ as a secret.

When $\mathcal{A}$ inquiries this private key about the p^{th} $(p < k)$ identity vector $\vec{I} = (ID_1, \ldots, ID_j)$, $\mathscr{B}$ randomly generates $r, y_1, \ldots, y_n \in Z_N$ and $w, \xi_1, \ldots, \xi_n, \zeta_{j+1}, \ldots, \zeta_l, z_k \in Z_N$. $\mathscr{B}$ generates a semi-functional private key.

$$\widetilde{\overrightarrow{K_0}} = g_1^{y_1}.(Y_2 Y_3)^{\xi_1}, g_1^{y_2}.(Y_2 Y_3)^{\xi_2}, \ldots, g_1^{y_n}.(Y_2 Y_3)^{\xi_n}),$$
$$\widetilde{K_1} = g_1^r (Y_2 Y_3),$$
$$\widetilde{K_2} = g_1^{\alpha} \prod_{i=1}^{n} g_1^{-x_i y_i} (u_1^{ID_1} \ldots u_j^{ID_j} h_1)^r.(Y_2 Y_3)^{z_k},$$
$$\widetilde{E_{j+1}} = u_{j+1}^r.(Y_2 Y_3)^{\xi_{j+1}}, \ldots, \widetilde{E_l} = u_l^r.(Y_2 Y_3)^{\xi_l}.$$

This is the correctly distributed semi-functional key, which implies that $g_2^w = Y_2$.

When $\mathcal{A}$ inquiries this private key about the p^{th} $(p > k)$ identity vector $\vec{I} = (ID_1, \ldots, ID_j)$, $\mathscr{B}$ randomly generates $r, y_1, \ldots, y_n \in Z_N$ and $R_{0,1}, \ldots, R_{0,n}, R_3, R_3', R_{j+1}, \ldots, R_l$ in G_{p_3}. $\mathscr{B}$ produces the normal private key.

$$\overrightarrow{K_0} = (g_1^{y_1} R_{0,1}, g_1^{y_2} R_{0,2}, \ldots, g_1^{y_n} R_{0,n}),$$
$$K_1 = g_1^r R_3,$$
$$K_2 = g_1^{\alpha} \prod_{i=1}^{n} g_1^{-x_i y_i} (u_1^{ID_1} \ldots u_j^{ID_j} h_1)^r R_3',$$
$$E_{j+1} = u_{j+1}^r R_{j+1}, \ldots, E_l = u_l^r R_l.$$

When $\mathcal{A}$ inquiries this private key about the p^{th} $(p = k)$ identity vector $\vec{I} = (ID_1, \ldots, ID_j)$, $\mathscr{B}$ randomly generates $r, y_1, \ldots, y_n \in Z_N$ and $w, \xi_1, \ldots, \xi_n, \zeta_{j+1}, \ldots, \zeta_l, z_k \in Z_N$. Then, $\mathscr{B}$ sets $z_k = a_1 ID_1 + \ldots + a_j ID_j + b$. $\mathscr{B}$ produces the private key.

$$\overrightarrow{K_0} = (T^{y_1}, \ldots, T^{y_n}),$$
$$K_1 = T,$$
$$K_2 = g_1^{\alpha} \prod_{i=1}^{n} g_1^{-x_i y_i} T^{z_k} X_3^w,$$
$$E_{j+1} = T^{a_{j+1}} X_3^{\xi_{j+1}}, \ldots, E_l = T^{a_l} X_3^{\zeta_l}.$$

If $T \in G_{n_1 n_3}$, this key is normal, where g_1^r is equivalent to the part in T. Otherwise, $T \in G$. This key is semi functional.

Challenge: The attacker $\mathcal{A}$ gives two plaintexts M_0, M_1 and an identity vector $\vec{I}^*$. The constraint is that $\vec{I}^*$ is not in $\mathscr{R}$. The challenger $\mathscr{B}$ selects randomly $\beta \leftarrow \{0,1\}$ and encrypts M_β. $\mathscr{B}$ calculates the ciphertext CTF as follows.

$\mathscr{B}$ randomly chooses $z_1, z_2, \ldots z_l, t \in Z_N$, and produces the ciphertext.

$\overrightarrow{C_0}=((X_1X_2)^{x_1},(X_1X_2)^{x_2},\ldots,(X_1X_2)^{x_n})$
$R\ =e(g_1,X_1X_2)^{\alpha}$,
$C_1=(X_1X_2)^{a_1z_1+\ldots+a_lz_l+b}$,
$C_2=X_1X_2, C_{3,1}=(X_1X_2)^{a_1t}$,
$C_{3,2}=(X_1X_2)^{a_2t},\ldots,C_{3,l}=(X_1X_2)^{a_lt}$.

Then, $\mathscr{B}$ computes.

$t_1=t^{-1}(ID_1^*-z_1)\ mod\ N$,
$t_2=t^{-1}(ID_2^*-z_2)\ mod\ N$, ...,
$t_j=t^{-1}\left(ID_j^*-z_j\right)mod\ N$,
$t_{j+1}=-t^{-1}z_{j+1}\ mod\ N,\ldots$,
$t_l=-t^{-1}z_l\ mod\ N$,
$C_4=\ R\oplus M_\beta$.

The ciphertext is
$CTF=(\overrightarrow{C_0},C_1,C_2,C_{3,1},\ldots,C_{3,l},C_4,t_1,\ldots,t_l)$.
The ciphertext implies that $g_1^sg_2^v=X_1X_2$ and $z_c=a_1z_1+\ldots+a_lz_l+b$.

If $T\in G_{n_1n_3}$, $\mathscr{B}$ runs Game$_{k-1}$. Otherwise, $T\in G$. $\mathscr{B}$ simulates Game$_k$. Hence, algorithm $\mathscr{B}$ destroys assumption 2 by advantage ε.

Lemma 6: Given $LK_{PK}=(n-2\vartheta-1)\lambda$, if an attacker $\mathcal{A}$ may make a distinction between Game$_q$ and Game$_{Final}$ in advantage ε, i.e. Game$_q\ Adv_{\mathcal{A}}-$Game$_{Final}\ Adv_{\mathcal{A}}=\varepsilon$, there exists an algorithm $\mathscr{B}$ who can broke assumption 3 with advantage ε.

Proof. In consideration of $g_1,g_1^\alpha X_2,X_3,g_1^sY_2,Z_2$ and T, $\mathscr{B}$ chooses $a,a_1,\ldots,a_l,b,x_1,\ldots,x_n\in Z_N$ randomly and sets $u_1=g_1^{a_1},\ldots,u_l=g_l^{a_l}$, $e(g_1,g_1)^\alpha=e(g_1^\alpha X_2,g_1)$, $h_1=g_1^b$ and $g_1^{x_1},\ldots,g_1^{x_n}$. $\mathscr{B}$ transmits public parameter $\{N,g_1,h_1,u_1,\ldots,u_l,e(g_1,g_1)^\alpha,g_1^{x_1},\ldots,g_1^{x_n}\}$ to $\mathcal{A}$. $\mathscr{B}$ keeps the master key $MK=\{a\}$ as secret.

When $\mathcal{A}$ inquiries this private key about the identity vector $\vec{I}=(ID_1,\ldots,ID_j)$, $\mathscr{B}$ randomly generates $c,r,d,w,z,z_{j+1},\ldots,z_l,w_{j+1},\ldots,w_l,y_1,\ldots,y_n\in Z_N$. $\mathscr{B}$ produces the semi-functional private key.

$\widetilde{\overrightarrow{K_0}}\ =\ ((g_1^\alpha X_2)^{y_1},(g_1^\alpha X_2)^{y_2},\ldots,(g_1^\alpha X_2)^{y_n})$,
$\widetilde{K_1}\ =\ g_1^rZ_2^zX_3^d$,
$\widetilde{K_2}\ =\ g_1^\alpha X_2\prod_{i=1}^n g_1^{-x_iy_i}(u_1^{ID_1}\ldots u_j^{ID_j}h_1)^r.X_3^wZ_2^c$,
$\widetilde{E_{j+1}}\ =\ u_{j+1}^r.Z_2^{z_{j+1}}X_3^{w_{j+1}},\ldots,\widetilde{E_l}=u_l^r.Z_2^{z_j}X_3^{w_l}$.

The attacker $\mathcal{A}$ gives two plaintexts M_0, M_1 and an identity vector $\vec{I}^*=(ID_1^*,\ldots,ID_j^*)$. The constraint is that $\vec{I}^*$ is not in $\mathscr{R}$. The challenger $\mathscr{B}$ gets $\beta\leftarrow\{0,1\}$ in a random selection and encrypts M_β. $\mathscr{B}$ calculates the ciphertext CTF as follows.

$\mathscr{B}$ selects $z_1,z_2,\ldots z_l,t\in Z_N$ randomly, and calculates:
$\overrightarrow{C_0}\ =\ ((g_1^sY_2)^{x_1},(g_1^sY_2)^{x_2},\ldots,(g_1^sY_2)^{x_n})$
$R\ =T$,
$C_1\ =(g_1^sY_2)^{a_1z_1+\ldots+a_lz_l+b}$,
$C_2\ =\ g_1^sY_2, C_{3,1}=(g_1^sY_2)^{a_1t}$,
$C_{3,2}\ =\ (g_1^sY_2)^{a_2t},\ldots,C_{3,l}=(g_1^sY_2)^{a_lt}$.

Then, $\mathscr{B}$ computes:

$t_1\ =\ t^{-1}(ID_1^*-z_1)\ mod\ N$,
$t_2\ =\ t^{-1}(ID_2^*-z_2)\ mod\ N$, ...,
$t_j\ =\ t^{-1}\left(ID_j^*-z_j\right)mod\ N$,
$t_{j+1}\ =\ -t^{-1}z_{j+1}\ mod\ N,\ldots$,
$t_l\ =\ -t^{-1}z_l\ mod\ N$,
$C_4\ =\ R\oplus M_\beta$.
$CTF\ =\ (\overrightarrow{C_0},C_1,C_2,C_{3,1},\ldots,C_{3,l},C_4,t_1,\ldots,t_l)$

The ciphertext implies that $z_c=a_1z_1+\ldots+a_lz_l+b$.

Note that z_c is only related to module n_2 and $u_1=g_1^{a1},\ldots,u_l=g_1^{al},h_1=g_1^b$ are only the elements of subgroup G_{n_1}. So, when $a,a_1,\ldots,a_l,b\in Z_N$ are randomly selected, $a,a_1,\ldots,a_l,b$ modulo N is independent of that module n_2.

In case $T=e(g_1,g_1)^{as}$, the ciphertext is semi-functional about the message M_β. Besides, supposing that T is a random element in G^*, the ciphertext is semi-functional about a random message. Hence, algorithm $\mathscr{B}$ destroys assumption 3 by advantage ε.

Theorem 1. If assumption 1, 2, and 3 are true, our presented scheme is resistant to private key leakage. The total leakage amount for private key may come to $LK_{PK}=(n-2\vartheta-1)\lambda$, where $n\geq 2$ is an integer and $\lambda=\log n_2$.

Proof. We denote that the advantages which the attackers can obtain in assumption 1, assumption 2 and assumption 3 are $\varepsilon_1,\varepsilon_2$ and ε_3 respectively. According to the above six lemmas, the difference advantages that attacker $\mathcal{A}$ wins in the above different games are as follows.

$Game_{Real}\ Adv_{\mathcal{A}}=Game_{Real'}\ Adv_{\mathcal{A}}$
$Game_{Real'}\ Adv_{\mathcal{A}}-\text{Game}_{Restricted}Adv_{\mathcal{A}}\ =\varepsilon$
$Game_{Restricted}Adv_{\mathcal{A}}-Game_0\ Adv_{\mathcal{A}}\ =\varepsilon$
$Game_{k-1}Adv_{\mathcal{A}}-Game_k\ Adv_{\mathcal{A}}\ =\varepsilon$
$Game_qAdv_{\mathcal{A}}-Game_{Final}Adv_{\mathcal{A}}\ =\varepsilon$

So, we get
$Game_{Real'}\ Adv_{\mathcal{A}}-Game_{Final}\ Adv_{\mathcal{A}}\leq\varepsilon_2+\varepsilon_1+q\varepsilon_2+\varepsilon_3$
Because q is a definite finite number, the advantage obtained by any attacker can be ignored. Thus, theorem 1 holds.

7 Continual Leakage Resilience

Theorem 2. The given **CLR-HIBOOE** can resist continual leakage attack.

Proof. By **Updation,** the private keys are updated periodically. This algorithm has PP and $PK_{\vec{I}}$ as input and produces a new private key $\widehat{PK}_{\vec{I}}$ for $\vec{I}$. In essence, for **KeyUpd**, some extra random values are added to their original ones in this private key. Therefore, a new private key and its old private key have the same distribution.

By running **KeyUpd**, we get a new private key. Because the private keys are updated periodically, the proposed **CLR-HIBOOE** scheme resists continual leakage.

8 Leakage Ratio

In **CLR-HIBOOE**, n_1,n_2,n_3 are all λ-bits primes. A private key has $3(n+2+l-j)\lambda$ bits. For a private key, the total leakage can reach $(n-2\vartheta-1)\lambda$. What is more, ϑ and

n are positive constants. The relative leakage ratio of the private key is $\frac{(n-2\vartheta-1)\lambda}{3(n+2+l-j)\lambda}=\frac{(n-2\vartheta-1)\lambda}{3(n+2+l-j)}\approx\frac{1}{3}$.

9 Conclusions

The proposed **CLR-HIBOOE** resists the continuous leakage attack about private key. On account of dual system technology the security of our presented scheme is obtained. The leakage ratio about private key reaches 1/3.

The advantages of this scheme mainly embody the three aspects. First, it is suitable for lightweight equipment and it has high encryption efficiency. Second, our scheme resists the continual leakage for private key. Third, our scheme achieves fully security for standard model. Therefore, our scheme is very suitable for lightweight devices against side channel attacks.

Acknowledgements

This research was supported by the National Natural Science Foundation of China (62172292, 62072104, 61972095, U21A20465), the Natural Science Foundation of the Jiangsu Higher Education Institutions of China (17KJB520042, 20KJB413003), Suqian Sci&Tech Program (S201820, Z2019109), the cloud computing and big data security research team of Suqian University, and sponsored by Qing Lan Project. This work was also supported by the Natural Science Foundation of the Fujian Province, China (2020J01159).

References

[1] A. Shamir, Identity-Based Cryptosystems and Signature Schemes, *Workshop on the theory and application of cryptographic techniques*, Santa Barbara, California, USA, 1984, pp. 47-53.

[2] F. Guo, Y. Mu, Z. Chen, Identity-based Online/Offline Encryption, *Financial Cryptography and Data Security 2008*, Cozumel, Mexico, 2008, PP. 247-261.

[3] S. Chow, J. K. Liu, J. Y. Zhou, Identity-Based Online/Offline Key Encapsulation and Encryption, *Proceedings of the 6th ACM symposium on information, computer and communications security*, Hong Kong, China, 2011, pp. 52-60.

[4] J. Lai, Y. Mu, F. Guo, Efficient Identity-Based Online/Offline Encryption and Signcryption with Short Ciphertext, *International Journal of Information Security*, Vol. 16, No. 3, pp. 299-311, June, 2017.

[5] J. Xu, X. Wu, X. Xie, Efficient Identity-Based Offline/Online Encryption Scheme for Lightweight Devices, *IEEE Third International Conference on Data Science in Cyberspace*, Guangzhou, China, 2018, pp. 569-575.

[6] N. Chen, J. Li, Y. Zhang, Y. Guo, Efficient CP-ABE Scheme with Shared Decryption in Cloud Storage, *IEEE Transactions on Computers*, Vol. 71, No. 1, pp. 175-184, January, 2022.

[7] J. Li, Y. Zhang, J. Ning, X. Huang, G. S. Poh, D. Wang, Attribute Based Encryption with Privacy Protection and Accountability for CloudIoT, *IEEE Transactions on Cloud Computing*, Vol. 10, No. 2, pp. 762-773, April, 2022.

[8] N. Eltayieb, R. Elhabob, A. Hassan, F. Li, An Efficient Attribute-Based OnlineOffline Searchable Encryption and Its Application in Cloud-Based Reliable Smart Grid, *Journal of Systems Architecture*, Vol. 98, pp. 165-172, September, 2019.

[9] M. Ali, M. R. Sadeghi, X. Liu, Y. Miao, A. V. Vasilakos, Verifiable Online/Offline Multi-Keyword Search for Cloud-Assisted Industrial Internet of Things, *Journal of Information Security and Applications*, Vol. 65, Article No. 103101, March, 2022.

[10] R. Zhang, J. Li, Y. Lu, J. Han, Y. Zhang, Key Escrow-free Attribute Based Encryption with User Revocation, *Information Sciences*, Vol. 600, pp. 59-72, July, 2022.

[11] A. Elkhalil, J. Zhang, R. Elhabob, An Efficient Heterogeneous Block Chain-Based Online/Offline Signcryption Systems for Internet of Vehicles, *Cluster Computing*, Vol. 24, No. 3, pp. 2051-2068, September, 2021.

[12] P. Kocher, J. Horn, A. Fogh, D. Genkin, D. Gruss, W. Haas, M. Hamburg, M. Lipp, S. Mangard, T. Prescher, M. Schwarz, Y. Yarom, Spectre Attacks: Exploiting Speculative Execution, *2019 IEEE Symposium on Security and Privacy (SP)*, San Francisco, USA, 2019. pp. 1-19.

[13] M. Lipp, M. Schwarz, D. Gruss, T. Prescher, W. Haas, A. Fogh, J. Horn, S. Mangard, P. Kocher, D. Genkin, Y. Yarom, M. Hamburg, Meltdown: Reading Kernel Memory from User Space, *27th USENIX Security Symposium*, Baltimore, MD, USA, 2018, pp. 973-990.

[14] C. S. Chen, T. Wang, J. Tian, Improving Timing Attack on RSA-CRT via Error Detection and Correction Strategy, *Information Sciences*, Vol. 232, pp. 464-474, May, 2013.

[15] O. Acıiçmez, W. Schindler, C. K. Koç, Cache Based Remote Timing Attack on The AES, *Cryptographers' track at the RSA conference*, San Francisco, CA, USA, 2007, pp. 271-286.

[16] S. Micali, L. Reyzin, Physically Observable Cryptography, *Theory of Cryptography Conference*, Cambridge, MA, USA, 2004, pp. 278-296.

[17] K. Pietrzak, A Leakage-Resilient Mode of Operation, *Annual International Conference on the Theory and Applications of Cryptographic Techniques (EUROCRYPT 2009)*, Cologne, Germany, 2009, pp. 462-482.

[18] S. Faust, E. Kiltz, K. Pietrzak, G. N. Rothblum, Leakage-Resilient Signatures, *Theory of Cryptography Conference*, Zurich, Switzerland, 2010, pp. 343-360.

[19] A. Akavia, S. Goldwasser, V. Vaikuntanathan, Simultaneous Hardcore Bits and Cryptography against Memory Attacks, *Theory of Cryptography Conference*, San Francisco, CA, USA, 2009, pp. 474-495.

[20] S. S. M. Chow, Y. Dodis, Y. Rouselakis, B. Waters, Practical Leakage-Resilient Identity-Based Encryption from Simple Assumptions, *the 17th ACM conference on Computer and communications security*, Chicago Illinois, USA, 2010, pp. 152-161.

[21] M. Naor, G. Segev, Public Key Cryptosystems Resilient to Key Leakage, *SIAM Journal on Computing*, Vol. 41, No. 4, pp. 772-814, August, 2012.

[22] Z. Brakerski, Y. T. Kalai, J. Katz, V. Vaikuntanathan,

Overcoming The Hole in The Bucket: Public-key Cryptography Resilient to Continual Memory Leakage, *IEEE 51st Annual Symposium on Foundations of Computer Science*, Las Vegas, Nevada, USA, 2010, pp. 501-510.

[23] Y. Zhou, B. Yang, Y. Mu, T. Wang, X. Wang, Identity-Based Encryption Resilient to Continuous Key Leakage, *IET Information Security*, Vol 13, No. 5, pp. 426-434, September, 2019.

[24] Q. Yu, J. Li, S. Ji, Fully Secure ID-Based Signature Scheme with Continuous Leakage-Resilience, *Security and Communication Networks*, Vol. 2022, Article No. 8220259, January, 2022.

[25] B. Waters, Dual System Encryption: Realizing Fully Secure IBE and HIBE under Simple Assumptions, *29th Annual International Cryptology Conference*, Santa Barbara, CA, USA, 2009, pp. 619-636.

[26] H. Hou, B. Yang, M. Zhang, Y. Zhou, M. Huang, Fully Secure Wicked Identity-Based Encryption Resilient to Continual Auxiliary-Inputs Leakage, *Journal of Information Security and Applications*, Vol. 53, Article No. 102521, August, 2020.

[27] X. Zheng, F. Zheng, X. Liu, D. Wang, B. Meng, A Secure and Policy-Controlled Signature Scheme with Strong Expressiveness and Privacy-Preserving Policy, *IEEE Access*, Vol. 9, pp. 14945-14957, January, 2021.

[28] Y. Chen, J. Li, C. Liu, J. Han, Y. Zhang, P. Yi, Efficient Attribute Based Server-Aided Verification Signature, *IEEE Transactions on Service Computing*, DOI: 10.1109/TSC.2021.3096420.

[29] J. Li, Y. Chen, J. Han, C. Liu, Y. Zhang, H. Wang, Decentralized Attribute-based Server-aid Signature in The Internet of Things, *IEEE Internet of Things Journal*, Vol. 9, No. 6, pp. 4573-4583, March, 2022.

[30] J. Shen, T. Zhou, Z. Cao, Protection Methods for Cloud Data Security, *Journal of Computer Research and Development*, Vol. 58, No. 10, pp. 2079-2098, October, 2021.

[31] H. Yang, J. Shen, J. Lu, T. Zhou, X. Xia, S. Ji, A Privacy-Preserving Data Transmission Scheme Based on Oblivious Transfer and Blockchain Technology in the Smart Healthcare, *Security and Communication Networks*, Article No. 5781354, September, 2021.

[32] J. Shen, T. Zhou, X. Chen, J. Li, W. Susilo, Anonymous and Traceable Group Data Sharing in Cloud Computing, *IEEE Transactions on Information Forensics and Security*, Vol. 13, No. 4, pp. 912-925, April, 2018.

[33] J. Li, H. Yan, Y. Zhang, Efficient Identity-based Provable Multi-Copy Data Possession in Multi-Cloud Storage, *IEEE Transactions on Cloud Computing*, Vol. 10, No. 1, pp. 356-365, March, 2022.

[34] C. Wang, R. Huang, J. Shen, J. Liu, P. Vijayakumar, N. Kumar, A Novel Lightweight Authentication Protocol for Emergency Vehicle Avoidance in VANETs, *IEEE Internet of Things Journal*, Vol. 8, No. 18, pp. 14248-14257, September, 2021.

[35] J. Shen, Z. Gui, X. Chen, J. Zhang, Y. Xiang, Lightweight and Certificateless Multi-Receiver Secure Data Transmission Protocol for Wireless Body Area Networks, *IEEE Transactions on Dependable and Secure Computing*, Vol. 19, No. 3, pp. 1464-1475, May, 2022.

[36] C. Wang, J. Shen, J. Lai, J. Liu, B-TSCA: Blockchain assisted Trustworthiness Scalable Computation for V2I Authentication in VANETs, *IEEE Transactions on Emerging Topics in Computing*, Vol. 9, No. 3, pp. 1386-1396, July, 2021.

[37] S. Ji, Y. Yuan, J. Shen, C. F. Lai, B. Chen, An Efficient Three-Party Authentication and Key Agreement Protocol for Privacy-Preserving of IoT Devices in Mobile Edge Computing, *Journal of Internet Technology*, Vol. 23, No. 3, pp. 437-448, May, 2022.

Biographies

Qihong Yu received his M.S. degree in computer science from the Yangzhou University, China, in 2006. He received his Ph.D. degree in computer science from Hohai University, China, in 2016. He is an associate professor in Suqian University, China. His research interests include cryptography, network security, etc.

Jian Shen received the M.E. and Ph.D. degrees in computer science from Chosun University, South Korea, in 2009 and 2012, respectively. Since 2012, he has been a Professor at Nanjing University of Information Science and Technology, China. His research interests include public cryptography, cloud computing and security, etc.

Jin-Feng Lai received the Ph.D. degree in the Department of Engineering Science from National Cheng Kung University, Taiwan, in 2008. He is currently a Professor with the Department of Engineering Science, National Cheng Kung University. His research interests include multimedia communications, sensor-based healthcare, and embedded systems, etc.

Sai Ji received his MS and PhD degrees from Nanjing University of Aeronautics and Astronautics, China in 2006 and 2014. Now, he is a professor in Nanjing University of Information Science & Technology and in Suqian University. His research interests mainly include wireless sensor networks, secure computing, etc.