A Fully Secure Identity Based Encryption Scheme with Equality Test in The Standard Model in Cloud Computing

Zijun Zhou, Yongjian Liao, Ganglin Zhang, Tingyun Gan, and Shijie Zhou
(Corresponding author: Yongjian Liao, Zijun Zhou)

School of Information and Software Engineering, University of Electronic Science and Technology of China
No.4, Section 2, North Jianshe Road, Chengdu, Sichuan 610054, China
Email: liaoyj@uestc.edu.cn, 1364041927@qq.com

(Received Dec. 29, 2021; Revised and Accepted Apr. 28, 2022; First Online May 1, 2022)

Abstract

Traditional encryption schemes cannot realize encrypted data searching in cloud computing. To solve this issue, several notions have been proposed. One of the concepts is identity-based encryption with an equality test (IBEET). Although an IBEET scheme in the standard model was proposed by Lee *et al.*, their approach is general and inefficient. In this paper, we construct a concrete IBEET scheme that can achieve full security in the standard model, and our scheme is more efficient than Lee *et al.*'s scheme under the same conditions. More specifically, our scheme improves by about 75.1% in the encryption algorithm compared with their scheme.

Keywords: Dual System Encryption; Equality Test; Full Security; Identity-based Encryption; Standard Model

1 Introduction

In recent years, the application of cloud computing has become more and more widespread. A growing number of files are stored in cloud servers, and to prevent these data leaks, they have been encrypted. But in addition to storing them, cloud servers need to process these encrypted data so that users can use them in the future.

To calculate encrypted data, many new issues have arisen, such as auditing [4], encrypted data searching [6], and so on. To realize encrypted data search, a new concept – public key encryption with keyword search (PKEKS) was presented by Boneh *et al.* [6]. PKEKS schemes can search for encrypted data, but can't decrypt it. Later, to realize the decryption function, a new method – public key encryption scheme with equality test (PKEET) was presented by Yang *et al.* [27]. This kind of scheme combines the public key encryption (PKE) and searchable encryption (SE), so they can decrypt ciphertexts and judge whether the messages corresponding to the ciphertexts are the same even if the public keys to encrypt the messages are different. Another PKEET scheme was presented by Zhang *et al.*, and the scheme was more efficient and achieved security in the standard model (SM) [28]. But PKEET has the problem of certificate management. Later, to solve this problem, another new method – identity based encryption with equality test (IBEET) was proposed by Ma [19]. And she gave the first concrete IBEET scheme that achieved one-way security under chosen-ciphertext attack (OW-CCA). Many people have conducted further research on IBEET based on the concept of Ma. A semi-generic approach for IBEET schemes and PKEET schemes was presented by Lee *et al.* [10]. Also, a general approach for PKEET schemes was presented by Lin *et al.* [17] and their construction could be easily extended to IBEET schemes. An efficient IBEET scheme was presented by Wu *et al.* [24]. Because the HashToPoint function is time-consuming, to reduce its use to improve efficiency, the scheme uses bilinear pairing. Later this scheme was applied to the smart grid of smart city [25]. In the equality test schemes, it is a difficult task to solve the security problem of internal attacks. To solve the problem, a new equality test scheme was presented by Wu *et al.* [26] by using identity based cryptography. But by using an attack, Lee *et al.* [12] proved that Wu's scheme didn't achieve the security they required, and they gave a modification method. Later, to resist insider attack in cloud computing, Seth Alornyo *et al.* [2] proposed a new scheme by using a witness based cryptographic primitive with an added pairing operation. And they refered to their scheme as identity based public key cryptographic primitive with delegated equality test against insider attack in cloud computing (IB-PKC-DETIA). A new IBEET scheme presented by Seth Alornyo *et al.* [1] was used to detect malware and verify encrypted data. An efficient identity based privacy information sharing scheme was presented by Wu *et al.* [23]. The scheme uses a similarity test to search for data similar to the target data on data that has been

encrypted in the cloud environment. IBEET solves the problem of PKEET, but it has the problem of key escrowing. To solve this problem, a new concept – certificateless PKEET (CL-PKEET) was presented by Qu *et al.* [20]. Later, the IBEET scheme of Ma was proved by Liao *et al.* that isn't OW-CCA secure [15], and they improved the scheme. A new concept – IBEET supporting flexible authorization (IBEET-FA) was presented by Li *et al.* [14]. Using the RSA assumption, Ramadan *et al.* [21] presented an efficient IBEET scheme. Later, by introducing group mechanism into IBEET, a novel concept – group IBEET (G-IBEET) was presented by Ling *et al.* [18]. Seth Alornyo *et al.* [3] combined the concepts of key-insulated encryption (KIE) and identity-based encryption with the equality test (IBEET) to obtain identity-based key-insulated encryption with equality test (IB-KIEET). And their scheme reduced the possibility of key exposure by adding the key-insulated mechanism. Recently, a general approach for PKEET which achieved security in the SM was presented by Lee *et al.* [11] and the approach can also be extended to the IBEET scheme.

As far as we know, for IBEET systems, except a general approach presented by Lee *et al.* [11] which can achieve security in the SM, all others achieve security in the random oracle model (ROM). Lee *et al.* constructed the generic IBEET scheme by combining a hierarchical identity based encryption scheme (HIBE), a strongly unforgeable signature scheme, and a cryptographic hash function and they require that the HIBE scheme is 3-level and the signature is one-time. And to ensure the scheme's equality test function, security in the SM, and the validity of ciphertexts, the generic approach needs to use the HIBE scheme to encrypt twice and the signature scheme to sign once. So the efficiency of their scheme is not high.

In this paper, we present a novel IBEET scheme which achieves full security in the SM. Firstly, we give the definitions of the IBEET model and the security model. Secondly, we construct our IBEET scheme based on composite order bilinear groups and prove it achieves one-way and indistinguishability security under chosen-identity and chosen-ciphertext attack (OW/IND-ID-CCA). Finally, the scheme is compared with the existing IBEET schemes. As far as we know, we present the first concrete IBEET scheme for full security in the SM. The contributions of our paper are as follows.

- We present a fully secure IBEET scheme by using the Lewko and Waters' IBE scheme [13] and a variant of their scheme, so the bilinear group on which our scheme is based is of composite order and our scheme's basic theory is the dual system encryption technology studied by Lewko and Waters. Based on the subgroup decision problem, our scheme is proven to achieve OW/IND-ID-CCA security. It can be used for encrypted search or encrypted classification of information with high-security requirements.
- Our scheme doesn't need additional calculations to enhance security, such as the one-time signature scheme [11] and the interpolating polynomial with degree two [17]. While our scheme directly uses the bilinear Diffie-Hellman (BDH) problem to ensure the ciphertext is valid and cannot be tampered with. It means that our scheme only needs to encrypt a message and an "authentication code" to realize the encryption of messages, the validation of ciphertexts, and the equality test on ciphertexts. Thus, the structure of our proposed scheme is simpler, and our scheme is more efficient. More specifically, compared with the only existing IBEET scheme that is secure in the SM [11], our scheme improves by about 75.1% in the encryption algorithm. Thus, our scheme is more suitable for sensor networks.

Now, we give the organization of the rest of our paper. Section 2 mainly gives the application scenario of IBEET schemes, the basic concepts which will be used, and the definitions of the IBEET model and the security model. Then in section 3, we give the concrete structure of our scheme and prove that it achieves OW/IND-ID-CCA security in the SM. And in section 4, we improve our scheme and analysis its security. Next, in section 5, we compare the scheme in section 3 with other IBEET schemes in terms of computational costs, security, and parameter sizes. In the end, there is a summary of this paper in section 6.

2 Preliminary

Here, we will give the application scenario of IBEET schemes and the definitions of the IBEET scheme and the security model and review some basic knowledge.

2.1 Application Scenario

Ma [19] first proposed the concept of IBEET and introduced the application scenario of the IBEET algorithm in the same paper. In simple terms, the IBEET algorithm can search and classify encrypted files using encrypted keywords without revealing any information. For example, a hospital information management system, as shown in Figure 1. Alice and Bob are doctors in the same hospital. They will upload the patient information to the cloud server, and to facilitate searching, they will add some keywords after each case. To protect the patient information, they will use their ID to encrypt the cases and keywords, so the server cannot know the uploaded case information and its keywords. Then the ordinary algorithms cannot complete the search without decryption, while the IBEET algorithm can. It uses trapdoors to match the keyword ciphertext of one case with the keyword ciphertext of another case to judge whether their corresponding keywords are equal so that the encrypted cases can be searched by using their encrypted keywords and they are classified according to keywords. And no patient information will be disclosed during the whole process.

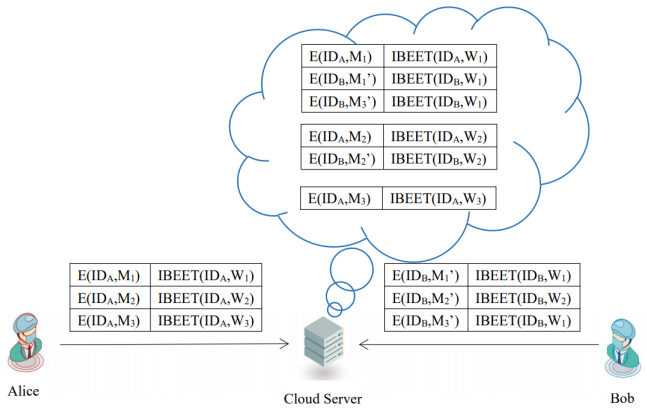

Figure 1: An example of IBEET

2.2 Composite Order Bilinear Groups

Boneh, Goh, and Nissim [7] first gave the definition of composite order bilinear group. We use a group generator $\mathcal{G}$ to define it. $\mathcal{G}$ is an algorithm. The input of it is a security parameter k and the outputs are G, G_T, $N = p_1p_2p_3$, e, where the order of cyclic groups G and G_T is N, p_1, p_2, p_3 are three different primes, and the map $e : G \times G \to G_T$ has the following properties:

1) (Bilinear) $\forall h_1, h_2 \in G, \alpha, \beta \in \mathbb{Z}_N, e({h_1}^{\alpha}, {h_2}^{\beta}) = e(h_1, h_2)^{\alpha\beta}$.

2) (Non-degenerate) $\exists h \in G$, so that $e(h, h)$ has order N in G_T.

We suppose the group descriptions of G and G_T contain generators for each cyclic group. The subgroup of order p_i in G is denoted by G_{p_i}, where $i = 1, 2, 3$. It is worth noting that if g_i is an element of G_{p_i}, g_j is an element of G_{p_j}, and $i \neq j$, then $e(g_i, g_j)$ is the identity element (denoted by 1) of G_T. For example, we assume that $i = 1$, $j = 2$, and the generator of G is denoted by g. Then the generator of G_{p_3} is denoted by $g^{p_1p_2}$, the generator of G_{p_2} is denoted by $g^{p_1p_3}$ and the generator of G_{p_1} is denoted by $g^{p_2p_3}$. Thus, for some α, β, $g_1 = (g^{p_2p_3})^{\alpha}$ and $g_2 = (g^{p_1p_3})^{\beta}$. We find:

$$e(g_1, g_2) = e(g^{p_2p_3\alpha}, g^{p_1p_3\beta}) = e(g^{\alpha}, g^{p_3\beta})^{p_2p_3p_1} = 1.$$

This is the orthogonality property of the subgroups of order p_i in G where $i = 1, 2, 3$ and we will use it in our construction and proof.

2.3 Dual System Encryption

Waters proposes a new methodology – dual system encryption [22], and its function is to prove the security of encryption systems, but it has a shortcoming, that is, it is not enough to realize short ciphertexts if it is directly applied. To solve this problem, Lewko and Waters [13] designed a new method to realize dual system encryption. A dual system has two types of ciphertexts and keys: normal form and semi-functional form. But the semi-functional form is only used for security proof. Normal ciphertexts can be decrypted by two kinds of keys. And normal keys can decrypt two kinds of ciphertexts. However, semi-functional keys aren't able to decrypt semi-functional ciphertexts. And the dual system uses a sequence of indistinguishable games to prove itself is secure. The specific proof method will be introduced in the scheme proof later.

2.4 Model of IBEET

There are six algorithms in an IBEET scheme, including Setup, Extract, Encrypt, Decrypt, Trapdoor, and Test [15].

Setup (k). The input is a security parameter k, and the output is public parameters PK and a master key msk.

Extract (ID, msk). The inputs are an identity $ID \in \{0,1\}^*$ and msk, and the output is the corresponding private key sk_{ID} of ID.

Encrypt (M, ID). The inputs are a message M and an identity $ID \in \{0,1\}^*$, and the output is a ciphertext C.

Decrypt (sk_{ID}, C). The inputs are a ciphertext C encrypted with an identity $ID \in \{0,1\}^*$ and the private key sk_{ID} of ID, and the output is a message M.

Trapdoor (ID, sk_{ID}). The inputs are an identity $ID \in \{0,1\}^*$ and sk_{ID}, and the output is a trapdoor td_{ID} of ID.

Test (C_A, td_A, C_B, td_B). The inputs of the algorithm are a ciphertext C_A encrypted with an identity $ID_A \in \{0,1\}^*$, the trapdoor td_A of the identity ID_A, and a ciphertext C_B encrypted with an identity $ID_B \in \{0,1\}^*$, the trapdoor td_B of the identity ID_B, and the output is "1" if C_A and C_B correspond to the same message, otherwise the output is "0".

Correctness. To be correct, these algorithms need to satisfy two conditions:

1) When sk_{ID} is the private key generated by *Extract* algorithm given ID, then

$$\forall M : Decrypt(sk_{ID}, C) = M$$

where $C = Encrypt(M, ID)$.

2) When td_A and td_B are trapdoors generated by *Trapdoor* algorithm given ID_A and ID_B, then

$$\forall M, M' : Test(C_A, td_A, C_B, td_B) = 1,$$

$$if\ and\ only\ if\ M = M'$$

where $C_A = Encrypt(M, ID_A)$ and $C_B = Encrypt(M', ID_B)$.

2.5 Security Model

Our IBEET security models are mainly defined for two types of adversaries: Type-I adversary that has a trapdoor of the target identity and Type-II adversary that has no trapdoor of the target identity.

Next, we give specific definitions of the two security models. The first one is the definition of the one-way security under chosen-identity and chosen-ciphertext attack (OW-ID-CCA) against Type-I adversaries, and the second one is the definition of the indistinguishability security under chosen-identity and chosen-ciphertext attack (IND-ID-CCA) against the Type-II adversaries.

Definition 1 (OW-ID-CCA against Type-I Adversaries)**.** *An IBEET scheme achieves OW-ID-CCA security if the probability of any PPT adversary $\mathcal{A}$ winning in the following game with the challenger $\mathcal{C}$ is negligible in the security parameter k:*

Setup. $\mathcal{C}$ executes *Setup(k)* and sends the public parameters PK to $\mathcal{A}$.

Phase 1. The following oracles can be adaptively queried by the adversary $\mathcal{A}$ polynomially many times.

$\mathcal{O}^{Ext}$: The input of this oracle is an identity ID_i, and the output is a private key sk_{ID_i} of the identity ID_i.

$\mathcal{O}^{Dec}$: The inputs are an identity ID_i and a ciphertext C_i encrypted with ID_i, and the output is the corresponding message M_i of the ciphertext C_i.

$\mathcal{O}^{Td}$: The input of this oracle is an identity ID_i, and the output is a trapdoor td_{ID_i} of ID_i.

Challenge. $\mathcal{A}$ chooses a target identity ID^* that has never been asked in $\mathcal{O}^{Ext}$ in Phase 1, and then sends ID^* to $\mathcal{C}$. $\mathcal{C}$ chooses a random message M, computes a challenge ciphertext $C^*_{ID^*}$ by executing *Encrypt(ID^*, M)*, and sends $C^*_{ID^*}$ to $\mathcal{A}$.

Phase 2. This phase is the same as Phase 1, but queries of $\mathcal{A}$ has the following limitations:

(a) $\mathcal{A}$ cannot query the oracle $\mathcal{O}^{Ext}$ for ID^*.

(b) $\mathcal{A}$ cannot query the oracle $\mathcal{O}^{Dec}$ for the pair of ID^* and $C^*_{ID^*}$.

Guess. $\mathcal{A}$ outputs M'.

If M' is equal to M, then the adversary $\mathcal{A}$ wins and the advantage is defined as

$$Adv^{OW-ID-CCA}_{\mathcal{A},IBEET}(k) := Pr[M = M'].$$

Definition 2 (IND-ID-CCA against Type-II Adversaries)**.** *An IBEET scheme achieves IND-ID-CCA security if the probability of any PPT adversary $\mathcal{A}$ winning in the following game with the challenger $\mathcal{C}$ is negligible in the security parameter k:*

Setup. $\mathcal{C}$ executes *Setup(k)* and sends the public parameters PK to $\mathcal{A}$.

Phase 1. The following oracles can be adaptively queried by the adversary $\mathcal{A}$ polynomially many times.

$\mathcal{O}^{Ext}$: The input of this oracle is an identity ID_i, and the output is the private key sk_{ID_i} of ID_i.

$\mathcal{O}^{Dec}$: The inputs are an identity ID_i and a ciphertext C_i encrypted with ID_i, and the output is the corresponding message M_i of the ciphertext C_i.

$\mathcal{O}^{Td}$: The input of this oracle is an identity ID_i, and the output is a trapdoor td_{ID_i} of the identity ID_i.

Challenge. $\mathcal{A}$ chooses a target identity ID^* that has never been asked in $\mathcal{O}^{Ext}$ and $\mathcal{O}^{Td}$ in Phase 1, and two messages M_0, M_1, which have the same length, and then sends ID^*, M_0, M_1 to $\mathcal{C}$. $\mathcal{C}$ computes a challenge ciphertext $C^*_{ID^*,b}$ by executing *Encrypt(ID^*, M_b)*, and sends $C^*_{ID^*,b}$ to $\mathcal{A}$, where b is randomly chosen from $\{0, 1\}$.

Phase 2. This phase is the same as Phase 1, but the query of $\mathcal{A}$ has the following limitations:

(a) $\mathcal{A}$ cannot query the oracle $\mathcal{O}^{Ext}$ and the oracle $\mathcal{O}^{Td}$ for the target identity ID^*;

(b) $\mathcal{A}$ cannot query the oracle $\mathcal{O}^{Dec}$ for the pair of ID^* and $C^*_{ID^*,b}$.

Guess. $\mathcal{A}$ outputs b', where b' is equal to 0 or 1.

If $b' = b$, then $\mathcal{A}$ wins, and the advantage is defined as

$$Adv^{IND-ID-CCA}_{\mathcal{A},IBEET}(k) := \mid Pr[b = b'] - \frac{1}{2} \mid$$

3 Our IBEET Scheme

Here, we first give the concrete structure of our IBEET scheme which is based on the IBE scheme presented by Lewko and Waters [13]. The order of the composite order bilinear group we use is $N = p_1p_2p_3$, and the identity is the element of $\mathbb{Z}_N$. Then our scheme is proved to achieve full security in the SM.

3.1 Construction

The existing generic IBEET scheme [11] encrypts twice to implement the encryption and testing function and uses the one-time signature scheme to enhance the security of their scheme. But our scheme only needs to use the same identity ID to encrypt twice. The first encryption is to encrypt the message M directly, which achieves the encryption function, and the other is to encrypt $u^{rH_1(M)}$, which is mainly used to achieve the correctness verification of decrypted message and the function of equality test. And because it is hard to solve the discrete logarithm problem, we can't get $rH_1(M)$ even if we know $u^{rH_1(M)}$ and u, thus, the second encryption does not leak M. Thus, compared with the scheme in [11], our scheme doesn't need additional calculations to enhance security.

The specific scheme design is as follows:

Setup. The input is a security parameter $k \in \mathbb{Z}^+$, and the output is the public parameter $PK = (N, u, g, h, e, e(g,g)^{\alpha_1}, e(g,g)^{\alpha_2}, H_1, H_2)$.

The specific meaning is as follow:

- G is a multiplicative group, G_T is a cyclic group and their order is N. G_{p_i} represents the subgroup of group G of order p_i.
- $N = p_1p_2p_3$, where p_1, p_2, p_3 denote three different prime numbers.
- $e : G \times G \to G_T$ is a bilinear map.
- $H_1 : G_T \to \mathbb{Z}_N$ and $H_2 : G_T \to G_{p_1}$ are two collision-resistant hash functions[1].
- $u, g, h \in G_{p_1}$, $\alpha_1, \alpha_2 \in \mathbb{Z}_N$, and they are random.
- The master key msk consists of α_1, α_2, and the generator of group G_{p_3}.

Extract. The inputs are an identity ID and msk, and the output is the following private key $sk_{ID} = (sk_{ID,1}, sk_{ID,2}, sk_{ID,3}, sk_{ID,4})$ for ID where $s_1, s_2 \in \mathbb{Z}_N$, $P_3, P_3' \in G_{p_3}$ are chosen at random:

$$\begin{aligned} sk_{ID,1} &= g^{s_1}P_3, \\ sk_{ID,2} &= g^{\alpha_1}(u^{ID}h)^{s_1}P_3', \\ sk_{ID,3} &= g^{s_2}P_3, \\ sk_{ID,4} &= g^{\alpha_2}(u^{ID}h)^{s_2}P_3'. \end{aligned}$$

Encrypt. The inputs are ID and a message M, and the output is the following ciphertext $C = (C_0, C_1, C_2, C_3)$ where $r \in \mathbb{Z}_N$ is chosen at random:

$$\begin{aligned} C_0 &= Me(g,g)^{\alpha_1 r}, \\ C_1 &= (u^{ID}h)^r, \\ C_2 &= g^r, \\ C_3 &= u^{rH_1(M)}H_2(e(g,g)^{\alpha_2 r}). \end{aligned}$$

Decrypt. The algorithm takes sk_{ID} and C encrypted with ID as inputs, and then computes a message M' by using the orthogonality of subgroups of group G and the bilinearity of the bilinear map, and finally outputs it. The calculation process is as follows:

$$\begin{aligned} \frac{e(sk_{ID,2}, C_2)}{e(sk_{ID,1}, C_1)} &= \frac{e(g,g)^{\alpha_1 r}e(u^{ID}h, g)^{s_1 r}}{e(u^{ID}h, g)^{s_1 r}} \\ &= e(g,g)^{\alpha_1 r}, \\ \frac{C_0}{e(g,g)^{\alpha_1 r}} &= M'. \end{aligned}$$

It's worth noting that we need to verify the validity of the message M', the process is as follows:

$$\begin{aligned} X &= \frac{e(sk_{ID,4}, C_2)}{e(sk_{ID,3}, C_1)} \\ U &= \frac{C_3}{H_2(X)} \\ e(U, g) &\stackrel{?}{=} e(u, C_2)^{H_1(M')} \end{aligned}$$

If $e(U,g) = e(u, C_2)^{H_1(M')}$, then output M', otherwise output $\perp$.

Trapdoor. The inputs are ID and sk_{ID} and the output is a trapdoor $td_{ID} = (td_{ID,1}, td_{ID,2})$ that is formed as:

$$\begin{aligned} td_{ID,1} &= sk_{ID,3} = g^{s_2}P_3, \\ td_{ID,2} &= sk_{ID,4} = g^{\alpha_2}(u^{ID}h)^{s_2}P_3''. \end{aligned}$$

Test. The algorithm takes a ciphertext C_A encrypted with an identity ID_A, the trapdoor td_A for the identity ID_A, and a ciphertext C_B encrypted with an identity ID_B, the trapdoor td_B for the identity ID_B as input, verifies whether the corresponding message M_A of C_A is equal to the corresponding message M_B of C_B and outputs the result. The calculation process is as follows:

First, the algorithm computes the parameters as follows:

$$\begin{aligned} E_A &= \frac{e(td_{ID_{A,2}}, C_{A,2})}{e(td_{ID_{A,1}}, C_{A,1})} = e(g,g)^{\alpha_2 r_A}, \\ X_A &= \frac{C_{A,3}}{H_2(E_A)} = u^{r_A H_1(M_A)}, \\ E_B &= \frac{e(td_{ID_{B,2}}, C_{B,2})}{e(td_{ID_{B,1}}, C_{B,1})} = e(g,g)^{\alpha_2 r_B}, \\ X_B &= \frac{C_{B,3}}{H_2(E_B)} = u^{r_B H_1(M_B)}. \end{aligned}$$

Then it verifies if $e(C_{A,2}, X_B) = e(C_{B,2}, X_A)$ is true. If it is true, M_A is equal to M_B, otherwise, they are not equal.

3.2 Correctness

Here, we verify the correctness.

Correctness of Decryption Algorithm.

$$\begin{aligned} X &= \frac{e(sk_{ID,4}, C_2)}{e(sk_{ID,3}, C_1)} \\ &= \frac{e(g,g)^{\alpha_2 r}e(u^{ID}h, g)^{s_2 r}}{e(u^{ID}h, g)^{s_2 r}} \\ &= e(g,g)^{\alpha_2 r} \\ U &= \frac{C_3}{H_2(e(g,g)^{\alpha_2 r})} = u^{rH_1(M)} \\ e(U, g) &= e(u^{rH_1(M)}, g) = e(u, g)^{rH_1(M)} \\ e(u, C_2)^{H_1(M')} &= e(u, g^r)^{H_1(M')} = e(u, g)^{rH_1(M')}. \end{aligned}$$

[1] Based on the difficulty of integer factoring, when H_1 is collision-resistant, it is also collision-resistant in the case of modulo p_1.

So if $e(U,g) = e(u, C_2)^{H_1(M')}$, then $M' = M$.

Correctness of Test Algorithm.

$$\begin{aligned} e(C_{A,2}, X_B) &= e(g^{r_A}, u^{r_B H_1(M_B)}) \\ &= e(g,u)^{r_A r_B H_1(M_B)} \\ e(C_{B,2}, X_A) &= e(g^{r_B}, u^{r_A H_1(M_A)}) \\ &= e(g,u)^{r_B r_A H_1(M_A)}. \end{aligned}$$

So if $e(C_{A,2}, X_B) = e(C_{B,2}, X_A)$, then $M_A = M_B$, otherwise $M_A \neq M_B$.

3.3 Security

We firstly give the complexity assumptions that we need to use in the proof. These assumptions are similar to the assumptions of Lewko and Waters [13]. These assumptions don't depend on how many times an attacker asks, so they are static. Assumption 1 is the subgroup decision problem and the order of this group is the product of three different prime numbers. And in Appendix A of [13], by using the theorems in [9], Lewko and Waters proved that if computing a nontrivial factor for the group order is difficult, these assumptions are valid in the general group model.

Assumption 1 (Subgroup decision problem for three primes). We give the following definition of the distribution, where $\mathcal{G}$ is a group generator:

$$\begin{aligned} \mathbb{G} &= (N = p_1p_2p_3, G, G_T, e) \xleftarrow{R} \mathcal{G}, g \xleftarrow{R} G_{p_1}, \\ &\quad Z_1 \xleftarrow{R} G_{p_3}, \\ F &= (\mathbb{G}, g, Z_1), T_1 \xleftarrow{R} G_{p_1p_2},\ T_2 \xleftarrow{R} G_{p_1}. \end{aligned}$$

The definition of the advantage that Assumption 1 is broken by an algorithm $\mathcal{A}$ is as follows:

$$Adv1_{\mathcal{G},\mathcal{A}}(k) := |Pr[\mathcal{A}(F, T_1) = 1] - Pr[\mathcal{A}(F, T_2) = 1]|.$$

It can be noticed that T_1 is an element of $G_{p_1p_2}$, so it can be seen as the product of the elements in G_{p_1} and G_{p_2}. And these elements are called the "G_{p_1} part of T_1" and the "G_{p_2} part of T_2" respectively. This nomenclature is going to be used in the proof.

Definition 3. *We define $\mathcal{G}$ satisfies Assumption 1 if for any polynomial time algorithm $\mathcal{A}$, $Adv1_{\mathcal{G},\mathcal{A}}(k)$ is negligible.*

Assumption 2. We give the following definition of the distribution, where $\mathcal{G}$ is a group generator:

$$\begin{aligned} \mathbb{G} &= (N = p_1p_2p_3, G, G_T, e) \xleftarrow{R} \mathcal{G}, g, X_1 \xleftarrow{R} G_{p_1}, \\ &\quad Y_1, Y_2 \xleftarrow{R} G_{p_2},\ Z_1, Z_2 \xleftarrow{R} G_{p_3}, \\ F &= (\mathbb{G}, g, X_1Y_1, Z_1, Y_2Z_2), T_1 \xleftarrow{R} G,\ T_2 \xleftarrow{R} G_{p_1p_3}. \end{aligned}$$

The definition of the advantage that Assumption 2 is broken by an algorithm $\mathcal{A}$ is as follows:

$$Adv2_{\mathcal{G},\mathcal{A}}(k) := |Pr[\mathcal{A}(F, T_1) = 1] - Pr[\mathcal{A}(F, T_2) = 1]|.$$

Definition 4. *We define $\mathcal{G}$ satisfies Assumption 2 if for any polynomial time algorithm $\mathcal{A}$, $Adv2_{\mathcal{G},\mathcal{A}}(k)$ is negligible.*

Assumption 3. We give the following definition of the distribution, where $\mathcal{G}$ is a group generator:

$$\begin{aligned} \mathbb{G} &= (N = p_1p_2p_3, G, G_T, e) \xleftarrow{R} \mathcal{G},\ \alpha, r \xleftarrow{R} \mathbb{Z}_N, \\ &\quad g \xleftarrow{R} G_{p_1},\ Y_1, Y_2, Y_3 \xleftarrow{R} G_{p_2},\ Z_1 \xleftarrow{R} G_{p_3}, \\ F &= (\mathbb{G}, g, g^{\alpha}Y_1, Z_1, g^{r}Y_2, Y_3), T_1 = e(g,g)^{\alpha r},\ T_2 \xleftarrow{R} G_T. \end{aligned}$$

The definition of the advantage that Assumption 3 is broken by an algorithm $\mathcal{A}$ is as follows:

$$Adv3_{\mathcal{G},\mathcal{A}}(k) := |Pr[\mathcal{A}(F, T_1) = 1] - Pr[\mathcal{A}(F, T_2) = 1]|.$$

Definition 5. *We define $\mathcal{G}$ satisfies Assumption 3 if for any polynomial time algorithm $\mathcal{A}$, $Adv3_{\mathcal{G},\mathcal{A}}(k)$ is negligible.*

Then because we need to use semi-functional ciphertexts and keys in our proof, we now give the definitions of them. And the generator of the subgroup G_{p_2} is denoted by g_2.

Semi-functional Ciphertext. We create the following semi-functional ciphertext $C' = (C'_0, C'_1, C'_2, C'_3)$:

$$\begin{aligned} C'_0 &= C_0, \\ C'_1 &= g_2^{cz_{c_1}} C_1, \\ C'_2 &= g_2^{c} C_2, \\ C'_3 &= g_2^{cz_{c_2}} C_3. \end{aligned}$$

where a normal ciphertext $C = (C_0, C_1, C_2, C_3)$ is generated by the encryption algorithm and $c, z_{c_1}, z_{c_2} \in \mathbb{Z}_N$ are random exponents.

Semi-functional Key. We create the following semi-functional key $sk'_{ID} = (sk'_{ID,1}, sk'_{ID,2}, sk'_{ID,3}, sk'_{ID,4})$:

$$\begin{aligned} sk'_{ID,1} &= g_2^{k} sk_{ID,1}, \\ sk'_{ID,2} &= g_2^{kz_k} sk_{ID,2}, \\ sk'_{ID,3} &= g_2^{k} sk_{ID,3}, \\ sk'_{ID,4} &= g_2^{kz_k} sk_{ID,4}. \end{aligned}$$

where $sk_{ID} = (sk_{ID,1}, sk_{ID,2}, sk_{ID,3}, sk_{ID,4})$ is a normal key and $k, z_k \in \mathbb{Z}_N$ are random exponents.

It is worth noting that an extra factor of $e(g_2, g_2)^{ck(z_k - z_{c_1})}$ will obscure the blinding factor if a semi-functional ciphertext is decrypted by a semi-functional key. And a semi-functional key can still decrypt a semi-functional ciphertext if z_{c_1} is equal to z_k. Although these semi-functional keys have terms in G_{p_2}, they can still perform the decryption function, so we call they are nominally semi-functional keys.

Now, we begin to prove our scheme achieves IND-ID-CCA security.

We will use a sequence of games to prove the scheme is IND-ID-CCA secure. $Game_{Real}$ is the first game which

is the real security game. $Game_{Restricted}$ is the second game and it is like the first game, but when the value of an identity equals the value of the challenge identity modulo p_2, the attacker can't query a private key for it, so there is a stronger restriction that the identities must be unequal modulo N in $Game_{Restricted}$, and the restriction will also be maintained in the following games. We set the number of times the attacker queries the private key is denoted by q. We set $Game_k$ as follows, where k is from 0 to q:

$Game_k$. This is similar to the second game, but the form of ciphertexts and keys received by the attacker is different: ciphertexts and the first k keys are semi-functional and the rest keys are normal. Therefore, there are only normal keys and semi-functional ciphertexts in $Game_0$, and there are only semi-functional keys and ciphertexts in $Game_q$.

$Game_{Final}$ is the last game. It is similar to $Game_q$, but the semi-functional ciphertext is the result of encrypting a random message rather than one of the two messages chosen by the attacker.

We use the following four lemmas to prove that each of these games is indistinguishable. And it should be noted that in the following proof, the private keys created by the algorithm $\mathcal{B}$ and the private keys created by the real system are identically distributed, so $\mathcal{B}$ is able to respond to the decryption queries, the private key queries, and so on, which are asked by the algorithm $\mathcal{A}$.

Lemma 1. *Assume an algorithm $\mathcal{A}$ exists which makes $Game_{Real}Adv_{\mathcal{A}} - Game_{Restricted}Adv_{\mathcal{A}} = \varepsilon$. We are able to construct an algorithm $\mathcal{B}$ which has the advantage $\geq \frac{\varepsilon}{2}$ to break Assumption 1 or 2.*

Proof. $\mathcal{B}$ receives g, Z_1, then it and $\mathcal{A}$ can simulate $Game_{Real}$ together. $\mathcal{A}$ produces two identities: ID and ID^*. $ID-ID^*$ is divided by p_2 and $ID \neq ID^*$ modulo N. The probability that $\mathcal{A}$ produces these two identities is ε. $\mathcal{B}$ produces a nontrivial factor $n = gcd(ID - ID^*, N)$ by using ID and ID^*. We set $m = \frac{N}{n}$, so $N = nm = p_1p_2p_3$. Therefore, n is divided by p_2. Let's think about two cases: the first one is that m is divided by p_1 and the second is that $n = p_1p_2$ and $m = p_3$. The probability that at least one of these two cases must occur is $\geq \frac{\varepsilon}{2}$.

In the first case, $\mathcal{B}$ breaks the first assumption. $\mathcal{B}$ receives g, Z_1 and T, and then it verifies that g^m is the identity element to prove that m is divided by p_1. And $\mathcal{B}$ tests whether T^m is the identity element. If it is, T is an element in G_{p_1}, otherwise, T is an element in $G_{p_1p_2}$.

In the second case, $\mathcal{B}$ breaks the second assumption. $\mathcal{B}$ receives g, X_1Y_1, Z_1, Y_2Z_2 and T, and then it verifies that $(X_1Y_1)^n$ is the identity element to prove $n = p_1p_2$ is true. And $\mathcal{B}$ tests whether $e((Y_2Z_2)^m, T)$ is the identity element. Because of $n = p_1p_2$ and $N = nm = p_1p_2p_3$, $m = p_3$ is true. If $T \in G_{p_1p_3}$, we can find that $e((Y_2Z_2)^m, T)$ is the identity element by using the orthogonality of subgroups. However, when T is an element of all the other subgroups of the group G, we can find that none of $e((Y_2Z_2)^m, T)$ is the identity element. And T is an element in a subgroup of G, so $T \in G$. Therefore, if $e((Y_2Z_2)^m, T)$ is the identity element, $T \in G_{p_1p_3}$, otherwise, $T \in G$. □

Lemma 2. *Assume an algorithm $\mathcal{A}$ exists which makes $Game_{Restricted}Adv_{\mathcal{A}} - Game_0Adv_{\mathcal{A}} = \varepsilon$. We are able to construct an algorithm $\mathcal{B}$ which has the advantage ε to break Assumption 1.*

Proof. Given g, Z_1, and T, $\mathcal{B}$ and $\mathcal{A}$ can simulate $Game_{Restricted}$ or $Game_0$. First, $\mathcal{B}$ sets the public parameters $\{N, u, g, h, e, e(g,g)^{\alpha_1}, e(g,g)^{\alpha_2}, H_1, H_2\}$, in which H_1, H_2 are two collision-resistant hash functions, $\alpha_1, \alpha_2, x, y \in \mathbb{Z}_N$ are random exponents chosen by $\mathcal{B}$ and $g = g$, $u = g^x$, $h = g^y$. And then $\mathcal{B}$ sends the parameters to $\mathcal{A}$. When $\mathcal{A}$ asks $\mathcal{B}$ for a private key of an identity ID_i, $\mathcal{B}$ sets the private key as follows, where $a_i, b_i, s_{i,1}, s_{i,2} \in \mathbb{Z}_N$ are random exponents chosen by $\mathcal{B}$:

$$
\begin{aligned}
sk_{ID,1} &= g^{s_{i,1}} {Z_1}^{a_i}, \\
sk_{ID,2} &= g^{\alpha_1} (u^{ID_i} h)^{s_{i,1}} {Z_1}^{b_i}, \\
sk_{ID,3} &= g^{s_{i,2}} {Z_1}^{a_i}, \\
sk_{ID,4} &= g^{\alpha_2} (u^{ID_i} h)^{s_{i,2}} {Z_1}^{b_i}.
\end{aligned}
$$

$\mathcal{B}$ receives a challenge identity ID^* and two messages: M_0, M_1 from $\mathcal{A}$. Then it sets the following ciphertext, where β randomly chosen by $\mathcal{B}$ is 0 or 1:

$$
\begin{aligned}
C_0 &= M_\beta e(T,g)^{\alpha_1}, \\
C_1 &= T^{xID^*+y}, \\
C_2 &= T, \\
C_3 &= T^{xH_1(M_\beta)} H_2(e(T,g)^{\alpha_2}).
\end{aligned}
$$

We can find that the G_{p_1} part of T is g^r in this ciphertext. And when $T \in G_{p_1}$, the ciphertext is normal. When $T \in G_{p_1p_2}$, it is semi-functional, and $z_{c_1} = xID^* + y$ and $z_{c_2} = xH_1(M_\beta)$. And since z_{c_1} $(mod\ p_2)$ isn't related to x $(mod\ p_1)$ and y $(mod\ p_1)$, its distribution is correct. Therefore, by using the output of $\mathcal{A}$, the possibilities of T can be distinguished by $\mathcal{B}$. □

Lemma 3. *Assume an algorithm $\mathcal{A}$ exists which makes $Game_{k-1}Adv_{\mathcal{A}} - Game_kAdv_{\mathcal{A}} = \varepsilon$. We are able to construct an algorithm $\mathcal{B}$ that has the advantage ε to break Assumption 2.*

Proof. Given g, X_1Y_1, Z_1, Y_2Z_2, T, $\mathcal{B}$ sets the public parameters $\{N, u, g, h, e, e(g,g)^{\alpha_1}, e(g,g)^{\alpha_2}, H_1, H_2\}$, in which H_1, H_2 are two collision-resistant hash functions, $\alpha_1, \alpha_2, x, y \in \mathbb{Z}_N$ are random exponents chosen by $\mathcal{B}$ and $g = g$, $u = g^x$, $h = g^y$. And then it sends these parameters to $\mathcal{A}$. When $\mathcal{B}$ responds to the i^{th} key for ID_i, the value of i influences the form of the private key which $\mathcal{B}$ responds.

If $i < k$, the private key it creates is semi-functional. And the key is formed as follows, where $a_i, c_i, s_{i,1}, s_{i,2} \in$

$\mathbb{Z}_N$ are random exponents chosen by $\mathcal{B}$:

$$sk_{ID,1} = g^{s_{i,1}}(Y_2Z_2)^{a_i},$$
$$sk_{ID,2} = g^{\alpha_1}(u^{ID_i}h)^{s_{i,1}}(Y_2Z_2)^{c_i},$$
$$sk_{ID,3} = g^{s_{i,2}}(Y_2Z_2)^{a_i},$$
$$sk_{ID,4} = g^{\alpha_2}(u^{ID_i}h)^{s_{i,2}}(Y_2Z_2)^{c_i}.$$

We find that a_i $(mod\ p_2)$ isn't related to a_i $(mod\ p_3)$ and c_i $(mod\ p_2)$ is also not related to c_i $(mod\ p_3)$, so the semi-functional key is correctly distributed and g_2^k is equal to $Y_2^{a_i}$.

If $i > k$, the private key that $\mathcal{B}$ creates is normal. And the key is formed as follows, where $a_i, b_i, s_{i,1}, s_{i,2} \in \mathbb{Z}_N$ are random exponents chosen by $\mathcal{B}$:

$$sk_{ID,1} = g^{s_{i,1}}{Z_1}^{a_i},$$
$$sk_{ID,2} = g^{\alpha_1}(u^{ID_i}h)^{s_{i,1}}{Z_1}^{b_i},$$
$$sk_{ID,3} = g^{s_{i,2}}{Z_1}^{a_i},$$
$$sk_{ID,4} = g^{\alpha_2}(u^{ID_i}h)^{s_{i,2}}{Z_1}^{b_i}.$$

If $i = k$, the following private key is created, where $z_k = xID_k + y$ and $a_k, b_k \in \mathbb{Z}_N$ are random exponents chosen by $\mathcal{B}$:

$$sk_{ID,1} = T,$$
$$sk_{ID,2} = g^{\alpha_1}T^{z_k}{Z_1}^{b_k},$$
$$sk_{ID,3} = T^{a_k},$$
$$sk_{ID,4} = g^{\alpha_2}(T^{a_k})^{z_k}{Z_1}^{b_k}.$$

At some point, $\mathcal{B}$ receives a challenge identity ID^* and two messages: M_0, M_1 from $\mathcal{A}$. Then it sets the following ciphertext, where β chosen by $\mathcal{B}$ is randomly 0 or 1:

$$C_0 = M_\beta e(X_1Y_1, g)^{\alpha_1},$$
$$C_1 = (X_1Y_1)^{xID^*+y},$$
$$C_2 = X_1Y_1,$$
$$C_3 = (X_1Y_1)^{xH_1(M_\beta)}H_2(e(X_1Y_1, g)^{\alpha_2}).$$

We can find that this makes $g^r = X_1$, $z_{c_1} = xID^* + y$ and $z_{c_2} = xH_1(M_\beta)$. Because $f(ID^*) = xID^* + y$ $(mod\ p_2)$ is a pairwise independent function, unless $ID_k \neq ID^*(mod\ p_2)$ is true, z_{c_1} and z_k don't seem to be randomly distributed to $\mathcal{A}$. x $(mod\ p_1)$ isn't related to x $(mod\ p_2)$ and y $(mod\ p_1)$ is also not related to y $(mod\ p_2)$. Moreover, if $ID_k \equiv ID^*(mod\ p_2)$, the key request made by $\mathcal{A}$ is invalid. This is where the extra modular constraint is used.

The relationship between z_{c_1} and z_k is hidden from $\mathcal{A}$, but it is very important: if $\mathcal{B}$ creates a semi-functional ciphertext encrypted with ID_k and decrypts it to test whether key k is a semi-functional key, then because z_{c_1} is equal to z_k, no matter whether k is semi-functional or normal, $\mathcal{B}$ will decrypt the ciphertext successfully. In other word, the semi-functional key k created by $\mathcal{B}$ is only nominal.

If $T \in G_{p_1p_3}$, then $Game_{k-1}$ has been simulated by $\mathcal{B}$ correctly. And if $T \in G$, then $Game_k$ has been simulated by $\mathcal{B}$ correctly. Therefore, $\mathcal{B}$ is able to distinguish the possibilities of T by using the result that $\mathcal{A}$ outputs. □

Lemma 4. *Assume an algorithm $\mathcal{A}$ exists which makes $Game_qAdv_{\mathcal{A}} - Game_{Final}Adv_{\mathcal{A}} = \varepsilon$. We are able to construct an algorithm $\mathcal{B}$ that has the advantage ε to break Assumption 3.*

Proof. Given $g, g^\alpha Y_1, Z_1, g^r Y_2, Z_2$, and T, $\mathcal{B}$ sets the public parameters $\{N, u, g, h, e, e(g,g)^{\alpha_1}, e(g,g)^{\alpha_2}, H_1, H_2\}$, in which H_1, H_2 are two collision-resistant hash functions, $x, y \in \mathbb{Z}_N$ are random exponents chosen by $\mathcal{B}$ and $g = g$, $u = g^x$, $h = g^y$, $e(g,g)^{\alpha_1} = e(g^\alpha Y_1, g)$, $e(g,g)^{\alpha_2} = e(g,g)^{\alpha_1}$. And then $\mathcal{B}$ sends the parameters to $\mathcal{A}$. When $\mathcal{B}$ responses a semi-functional private key for an identity ID_i, it sets the key as follows, where $s_{i,1}, s_{i,2}, z_i, b_i, a_i, k_i \in \mathbb{Z}_N$ are random exponents chosen by $\mathcal{B}$:

$$sk_{ID,1} = g^{s_{i,1}}{Y_3}^{k_i}{Z_1}^{a_i},$$
$$sk_{ID,2} = g^{\alpha_1}Y_1(u^{ID_i}h)^{s_{i,1}}{Y_3}^{z_i}{Z_2}^{b_i},$$
$$sk_{ID,3} = g^{s_{i,2}}{Y_3}^{k_i}{Z_1}^{a_i},$$
$$sk_{ID,4} = g^{\alpha_2}Y_1(u^{ID_i}h)^{s_{i,2}}{Y_3}^{z_i}{Z_2}^{b_i}.$$

$\mathcal{B}$ receives a challenge identity ID^* and two messages: M_0, M_1 from $\mathcal{A}$. Then it sets the following ciphertext, where β chosen by $\mathcal{B}$ is randomly 0 or 1:

$$C_0 = M_\beta T,$$
$$C_1 = (g^r Y_2)^{xID^*+y},$$
$$C_2 = g^r Y_2,$$
$$C_3 = (g^r Y_2)^{xH_1(M_\beta)}H_2(T).$$

We can find that this makes $z_{c_1} = xID^* + y$ and $z_{c_2} = xH_1(M_\beta)$. And since z_{c_1} is only modulo p_2, and $u = g^x$ and $h = g^y$ are elements of the subgroup G_{p_1}, when $\mathcal{B}$ randomly chooses x and y modulo N, x $(mod\ p_1)$ and y $(mod\ p_1)$ aren't related to $z_{c_1} = xID^* + y$ $(mod\ p_2)$.

If $T = e(g,g)^{\alpha r}$, the correctly distributed semi-functional ciphertext is obtained by encrypting M_β. If T is a random element in the group G_T, the semi-functional ciphertext is obtained by encrypting a random message. Therefore, by using the output of $\mathcal{A}$, the possibilities of T can be distinguished by $\mathcal{B}$. □

Theorem 1. *If Assumption 1, 2, 3 hold and the hash functions are collision-resistant, our IBEET system achieves IND-ID-CCA security in the SM.*

Proof. If Assumption 1, 2, 3 hold and the hash functions are collision-resistant, through these four lemmas, we have proved that $Game_{Real}$ and $Game_{Final}$ are indistinguishable. And because to the attacker, in $Game_{Final}$, the value of β is information-theoretically hidden, it cannot gain any advantage to break our IBBET scheme. Therefore, the IBEET scheme achieves IND-ID-CCA security in the SM. □

Next, we begin to prove that our scheme achieves OW-ID-CCA security.

We also use a sequence of games to prove security. And the definitions of $Game_{Real}$, $Game_{Restricted}$ and $Game_k$ in this proof are the same as the definitions of them in the IND-ID-CCA security proof, thus, so are the definitions of $Game_0$ and $Game_q$. But The definition of $Game_{Final}$ is different. In this security proof, $Game_{Final}$ is the same as $Game_q$.

We use the following three lemmas to prove each of these games is indistinguishable. And it should be noted that in the following proof, the private keys created by the algorithm $\mathcal{B}$ and the private keys created by the real system are identically distributed, so $\mathcal{B}$ is able to respond to the decryption queries, the private key queries, and so on, which are asked by the algorithm $\mathcal{A}$.

Lemma 5. *Assume an algorithm $\mathcal{A}$ exists which makes $Game_{Real}Adv_{\mathcal{A}} - Game_{Restricted}Adv_{\mathcal{A}} = \varepsilon$. We are able to construct an algorithm $\mathcal{B}$ which has the advantage $\geq \frac{\varepsilon}{2}$ to break Assumption 1 or 2.*

Proof. $\mathcal{B}$ receives g, Z_1, then it and $\mathcal{A}$ can simulate $Game_{Real}$ together. $\mathcal{A}$ produces two identities: ID and ID^*. $ID-ID^*$ is divided by p_2 and $ID \neq ID^*$ modulo N. The probability that $\mathcal{A}$ produces these two identities is ε. $\mathcal{B}$ produces a nontrivial factor $n = gcd(ID - ID^*, N)$ by using ID and ID^*. We set $m = \frac{N}{n}$, so $N = nm = p_1p_2p_3$. Therefore, n is divided by p_2. Let's think about two cases: the first one is that m is divided by p_1 and the second is that $n = p_1p_2$ and $m = p_3$. The probability that at least one of these two cases must occur is $\geq \frac{\varepsilon}{2}$.

In the first case, $\mathcal{B}$ breaks the first assumption. $\mathcal{B}$ receives g, Z_1 and T, and then it verifies that g^m is the identity element to prove that m is divided by p_1. And $\mathcal{B}$ tests whether T^m is the identity element. If it is, T is an element in G_{p_1}, otherwise, T is an element in $G_{p_1p_2}$.

In the second case, $\mathcal{B}$ breaks the second assumption. $\mathcal{B}$ receives g, X_1Y_1, Z_1, Y_2Z_2 and T, and then it verifies that $(X_1Y_1)^n$ is the identity element to prove $n = p_1p_2$ is true. And $\mathcal{B}$ tests whether $e((Y_2Z_2)^m, T)$ is the identity element. Because of $n = p_1p_2$ and $N = nm = p_1p_2p_3$, $m = p_3$ is true. If $T \in G_{p_1p_3}$, we can find that $e((Y_2Z_2)^m, T)$ is the identity element by using the orthogonality of subgroups. However, when T is an element of all the other subgroups of the group G, we can find that none of $e((Y_2Z_2)^m, T)$ is the identity element. And T is an element in a subgroup of G, so $T \in G$. Therefore, if $e((Y_2Z_2)^m, T)$ is the identity element, $T \in G_{p_1p_3}$, otherwise, $T \in G$. □

Lemma 6. *Assume an algorithm $\mathcal{A}$ exists which makes $Game_{Restricted}Adv_{\mathcal{A}} - Game_0Adv_{\mathcal{A}} = \varepsilon$. We are able to construct an algorithm $\mathcal{B}$ that has the advantage ε to break Assumption 1.*

Proof. Given g, Z_1, and T, $\mathcal{B}$ and $\mathcal{A}$ can simulate $Game_{Restricted}$ or $Game_0$. First, $\mathcal{B}$ sets the public parameters $\{N, u, g, h, e, e(g,g)^{\alpha_1}, e(g,g)^{\alpha_2}, H_1, H_2\}$, in which H_1, H_2 are two collision-resistant hash functions, $\alpha_1, \alpha_2, x, y \in \mathbb{Z}_N$ are random exponents chosen by $\mathcal{B}$ and $g = g$, $u = g^x$, $h = g^y$. And then $\mathcal{B}$ sends the parameters to $\mathcal{A}$. When $\mathcal{A}$ asks $\mathcal{B}$ for a private key of an identity ID_i, $\mathcal{B}$ sets the private key as follows, where $a_i, b_i, s_{i,1}, s_{i,2} \in \mathbb{Z}_N$ are random exponents chosen by $\mathcal{B}$:

$$
\begin{aligned}
sk_{ID,1} &= g^{s_{i,1}} {Z_1}^{a_i}, \\
sk_{ID,2} &= g^{\alpha_1}(u^{ID_i}h)^{s_{i,1}} {Z_1}^{b_i}, \\
sk_{ID,3} &= g^{s_{i,2}} {Z_1}^{a_i}, \\
sk_{ID,4} &= g^{\alpha_2}(u^{ID_i}h)^{s_{i,2}} {Z_1}^{b_i}.
\end{aligned}
$$

$\mathcal{B}$ receives a challenge identity ID^* from $\mathcal{A}$. Then it sets the following ciphertext, where M is a random message chosen by $\mathcal{B}$:

$$
\begin{aligned}
C_0 &= Me(T,g)^{\alpha_1}, \\
C_1 &= T^{xID^*+y}, \\
C_2 &= T, \\
C_3 &= T^{xH_1(M)}H_2(e(T,g)^{\alpha_2}).
\end{aligned}
$$

We can find that the G_{p_1} part of T is g^r in this ciphertext. And when $T \in G_{p_1}$, the ciphertext is normal. When $T \in G_{p_1p_2}$, it is semi-functional, and $z_{c_1} = xID^* + y$ and $z_{c_2} = xH_1(M)$. And since z_{c_1} $(mod\ p_2)$ isn't related to x $(mod\ p_1)$ and y $(mod\ p_1)$, its distribution is correct. Therefore, by using the output of $\mathcal{A}$, the possibilities of T can be distinguished by $\mathcal{B}$. □

Lemma 7. *Assume an algorithm $\mathcal{A}$ exists which makes $Game_{k-1}Adv_{\mathcal{A}} - Game_kAdv_{\mathcal{A}} = \varepsilon$. We are able to construct an algorithm $\mathcal{B}$ that has the advantage ε to break Assumption 2.*

Proof. Given g, X_1Y_1, Z_1, Y_2Z_2 and T, $\mathcal{B}$ sets the public parameters $\{N, u, g, h, e, e(g,g)^{\alpha_1}, e(g,g)^{\alpha_2}, H_1, H_2\}$, in which H_1, H_2 are two collision-resistant hash functions, $\alpha_1, \alpha_2, x, y \in \mathbb{Z}_N$ are random exponents chosen by $\mathcal{B}$ and $g = g$, $u = g^x$, $h = g^y$. And then it sends these parameters to $\mathcal{A}$. When $\mathcal{B}$ responds to the i^{th} key for ID_i, the value of i influences the form of the private key which $\mathcal{B}$ responds.

If $i < k$, the private key it creates is semi-functional. And the key is formed as follows, where $a_i, c_i, s_{i,1}, s_{i,2} \in \mathbb{Z}_N$ are random exponents chosen by $\mathcal{B}$:

$$
\begin{aligned}
sk_{ID,1} &= g^{s_{i,1}}(Y_2Z_2)^{a_i}, \\
sk_{ID,2} &= g^{\alpha_1}(u^{ID_i}h)^{s_{i,1}}(Y_2Z_2)^{c_i}, \\
sk_{ID,3} &= g^{s_{i,2}}(Y_2Z_2)^{a_i}, \\
sk_{ID,4} &= g^{\alpha_2}(u^{ID_i}h)^{s_{i,2}}(Y_2Z_2)^{c_i}.
\end{aligned}
$$

We find that a_i $(mod\ p_2)$ isn't related to a_i $(mod\ p_3)$ and c_i $(mod\ p_2)$ is also not related to c_i $(mod\ p_3)$, so the semi-functional key is correctly distributed and g_2^k is equal to $Y_2^{a_i}$.

If $i > k$, the private key that $\mathcal{B}$ creates is normal. And the key is formed as follows, where $a_i, b_i, s_{i,1}, s_{i,2} \in \mathbb{Z}_N$

are random exponents chosen by $\mathcal{B}$:

$$sk_{ID,1} = g^{s_{i,1}} {Z_1}^{a_i},$$
$$sk_{ID,2} = g^{\alpha_1}(u^{ID_i}h)^{s_{i,1}} {Z_1}^{b_i},$$
$$sk_{ID,3} = g^{s_{i,2}} {Z_1}^{a_i},$$
$$sk_{ID,4} = g^{\alpha_2}(u^{ID_i}h)^{s_{i,2}} {Z_1}^{b_i}.$$

If $i = k$, the private key is created as follows. $\mathcal{B}$ first sets $z_k = xID_k + y$ and chooses two random exponents $a_k, b_k \in \mathbb{Z}_N$. And then the specific construction of the key is:

$$sk_{ID,1} = T,$$
$$sk_{ID,2} = g^{\alpha_1} T^{z_k} {Z_1}^{b_k},$$
$$sk_{ID,1} = T^{a_k},$$
$$sk_{ID,3} = g^{\alpha_2}(T^{a_k})^{z_k} {Z_1}^{b_k}.$$

At some point, $\mathcal{B}$ receives a challenge identity ID^* from $\mathcal{A}$. Then it sets the following ciphertext, where M is a random message chosen by $\mathcal{B}$:

$$C_0 = Me(X_1Y_1, g)^{\alpha_1},$$
$$C_1 = (X_1Y_1)^{xID^*+y},$$
$$C_2 = X_1Y_1,$$
$$C_3 = (X_1Y_1)^{xH_1(M)} H_2(e(X_1Y_1, g)^{\alpha_2}).$$

We can find that this makes $g^r = X_1$, $z_{c_1} = xID^* + y$ and $z_{c_2} = xH_1(M)$. Because $f(ID^*) = xID^* + y \ (mod\ p_2)$ is a pairwise independent function, unless $ID_k \neq ID^* (mod\ p_2)$ is true, z_{c_1} and z_k don't seem to be randomly distributed to $\mathcal{A}$. $x\ (mod\ p_1)$ isn't related to $x\ (mod\ p_2)$ and $y\ (mod\ p_1)$ is also not related to $y\ (mod\ p_2)$. Moreover, if $ID_k \equiv ID^* (mod\ p_2)$, the key request made by $\mathcal{A}$ is invalid. This is where the extra modular constraint is used.

The relationship between z_{c_1} and z_k is hidden from $\mathcal{A}$, but it is very important: if $\mathcal{B}$ creates a semi-functional ciphertext encrypted with ID_k and decrypts it to test whether key k is a semi-functional key, then because z_{c_1} is equal to z_k, no matter whether k is semi-functional or normal, $\mathcal{B}$ will decrypt the ciphertext successfully. In other word, the semi-functional key k created by $\mathcal{B}$ is only nominal.

If $T \in G_{p_1p_3}$, then $Game_{k-1}$ has been simulated by $\mathcal{B}$ correctly. And if $T \in G$, then $Game_k$ has been simulated by $\mathcal{B}$ correctly. Thus, $\mathcal{B}$ is able to distinguish the possibilities of T by using the result that $\mathcal{A}$ outputs. □

Theorem 2. *If Assumption 1, 2 hold and the hash functions are collision-resistant, the IBEET scheme is OW-ID-CCA secure in the SM.*

Proof. If Assumption 1, 2 hold and the hash functions are collision-resistant, through the three lemmas, we have proved that $Game_{Real}$ and $Game_{Final}$ are indistinguishable. In $Game_{Final}$, both ciphertexts and private keys are semi-functional, and semi-functional keys cannot decrypt semi-functional ciphertexts, so each message is equally likely to be encrypted. Thus, the probability of the attacker guessing the message M correctly is $\frac{1}{N}$. This probability is negligible, so to the attacker, the value of M is hidden with an overwhelming probability, and the advantage of the attacker to break the IBBET system is negligible. Therefore, our IBEET scheme is OW-ID-CCA secure in the SM. □

4 Improved Scheme

In the previous scheme, the trapdoor is only related to the private key, which means that as long as the cloud server obtains the trapdoor of the user, it can match all his ciphertexts. This may lead to the abuse of the trapdoor, which is not conducive to the user's control of his information. Therefore, this section improves the previous scheme. And in the improved scheme, the trapdoor is related to the private key and the ciphertext (or message).

4.1 Construction

Compared with the previous scheme, the improved scheme only improves the trapdoor algorithm and the test algorithm. The details of these two improved algorithms are as follows.

Trapdoor: The inputs are the private key sk_{ID} corresponding to ID and a ciphertext $C = (C_0, C_1, C_2, C_3)$, and the output is a trapdoor td_{ID} that is formed as:

$$td_{ID} = \frac{e(sk_{ID,4}, C_2)}{e(sk_{ID,3}, C_1)} = e(g, g)^{\alpha_2 r},$$

Test: The algorithm takes a ciphertext C_A encrypted with an identity ID_A, the trapdoor td_A for the identity ID_A, and a ciphertext C_B encrypted with an identity ID_B, the trapdoor td_B for the identity ID_B as input, verifies whether the corresponding message M_A of C_A is equal to the corresponding message M_B of C_B and outputs the result. The calculation process is as follows:

First, the algorithm computes the parameters as follows:

$$X_A = \frac{C_{A,3}}{H_2(td_A)} = u^{r_A H_1(M_A)},$$
$$X_B = \frac{C_{B,3}}{H_2(td_B)} = u^{r_B H_1(M_B)}.$$

Then it verifies if $e(C_{A,2}, X_B) = e(C_{B,2}, X_A)$ is true. If it is true, M_A is equal to M_B, otherwise, they are not equal.

In addition, the correctness verification of the improved scheme is the same as that of the previous scheme.

4.2 Security

We can find that compared with the improved scheme, the previous scheme leaks part of the private key, which means that the security of the previous scheme is stronger than that of the improved scheme. Thus, as long as the previous scheme is secure, the improved scheme is secure. And Theorem 1 and Theorem 2 have been proved that the previous scheme can achieve OW/IND-ID-CCA security, so the improved scheme can also achieve OW/IND-ID-CCA security.

In addition, according to the definition of the IND-ID-CCA security model, we can find that the cloud server cannot match the ciphertext without obtaining the trapdoor of the ciphertext. Otherwise, the attacker in the security model can obtain two ciphertexts by encrypting the challenge messages using non-target identity, and then match them with the challenge ciphertext $C^*_{ID^*,b}$ to obtain the message corresponding to $C^*_{ID^*,b}$ without knowing the trapdoor of $C^*_{ID^*,b}$. Thus, as long as the scheme is IND-ID-CCA secure, the cloud server must obtain the trapdoor of the ciphertext before matching it. According to the analysis, the improved scheme can realize IND-ID-CCA security, so the cloud server can only match the ciphertexts which it has known the trapdoors. Therefore, the improved scheme ensures that the trapdoor will not be abused.

5 Analytical Evaluation

Here, we compare our scheme in section 3 and the schemes mentioned in [10, 11, 16, 19, 24] for performance, and we mainly compare in terms of computational costs, security, and parameter sizes.

5.1 Symbol Definition

In Table 1, we will define some symbols that will be used in this section.

Table 1: Meaning of each symbol

Symbol	Meaning
Enc	encryption algorithm
Dec	decryption algorithm
Test	test algorithm
E_G	the computational cost of an exponentiation operation in group $\mathbb{G}$
E_T	the computational cost of an exponentiation operation in group $\mathbb{G}_T$
BP	the computational cost of a bilinear map evaluation
CT	ciphertext
TD	trapdoor
$\lvert\mathbb{Z}_p\rvert$	bit-lengths of elements in group $\mathbb{Z}_p$
$\lvert\mathbb{G}\rvert$	bit-lengths of elements in group $\mathbb{G}$
$\lvert\mathbb{G}_T\rvert$	bit-lengths of elements in group $\mathbb{G}_T$
H	hash function
k	security parameter
Comp	computational costs
Order	the order type of the group on which the algorithm is based

5.2 Comparison Conclusion

According to Table 2, we can find only our scheme and the scheme in [11] achieve security in the SM, the others achieve security in the ROM. And we note that only the bilinear group on which our scheme is based is of composite order, the bilinear groups on which the other schemes are based are of prime order. Compared with the schemes in [10, 16, 19, 24], our scheme enhance security, but our scheme is less efficient than them. Furthermore, Ramadan *et al.* [21] proposed a highly efficient IBEET system by using the RSA assumption, but in the same security conditions, its ciphertext length is much longer than that of the IBEET scheme which is based on the assumptions on the elliptic curve. And the IBEET systems of Ma *et al.* and Wu *et al.* are proved insecure by Liao *et al.* in their security model, and Liao *et al.* have improved their scheme [15, 16].

Although our scheme and the generic IBEET scheme of Lee *et al.* [11] both achieve security in the SM, they are very different. Firstly, in Table 2, to compare efficiency, we use the 3-level HIBE scheme in [5] and the signature scheme in [8] to implement the scheme in [11]. Because the encryption algorithm of our scheme is calculated on the subgroup G_{p_1}, our scheme's encryption algorithm can be regarded as realized on the prime order group. Thus, we can find that under the same conditions, our scheme is more efficient. More specifically, according to our test data in Table 3, we can find that compared with the scheme in [11], our scheme improves by about 75.1% in the encryption algorithm. Secondly, the scheme in [11] is a generic IBEET scheme by combining a HIBE scheme, a strongly unforgeable signature scheme, and a cryptographic hash function and they require that the HIBE scheme is 3-level and the signature is one-time. While our scheme is the first concrete IBEET scheme of which the basic theory is the dual system encryption technology studied by Lewko and Waters, and the bilinear groups on which our scheme is based is of composite order. Thirdly, to ensure the scheme's equality test function and security in the SM, their scheme encrypts a message and an "authentication code" by using the same encryption scheme with different identities and signs the ciphertext once. While our scheme presented by us just needs to encrypt a message and an "authentication code" by using two different encryption schemes with the same identity. Fourthly, to ensure that ciphertexts are valid and can't be tampered with, they use strongly unforgeable signatures, while we directly use the BDH problem. Finally, they use the standard hybrid argument to prove their scheme achieves security, while based on the subgroup decision problem, we prove our scheme achieves full security by using the dual system encryption technology. Therefore, the scheme presented by us and the scheme in [11] use dif-

Table 2: Comparison of our IBEET with existing schemes

		[19]	[10]	[24]	[16]	[11]	Our
Comp of	**Enc**	$4E_G + 2E_T + 2BP$	$3E_G + 3E_T$	$2E_T$	$2E_T$	$14E_G + 2E_T + 1BP$	$4E_G + 2E_T$
	Dec	$2E_G + 2BP$	$2E_G + 3BP$	$2BP$	$2BP$	$14E_G + 11BP$	$6BP$
	Test	$4BP$	$2E_G + 2BP$	$2E_T + 2BP$	$4BP$	$12E_G + 8BP$	$6BP$
Size of	**CT**	$4\|\mathbb{G}\| + \|k\|$	$2\|\mathbb{G}\| + 5\|H\|$	$3\|\mathbb{G}\| + \|H\|$	$3\|\mathbb{G}\| + \|H\|$	$(2k+15)\|\mathbb{G}\| + 2\|\mathbb{G}_T\| + \|\mathbb{Z}_p\|$	$3\|\mathbb{G}\| + \|\mathbb{G}_T\|$
	TD	$\|\mathbb{G}\|$	$\|\mathbb{G}\|$	$\|\mathbb{G}\|$	$2\|\mathbb{G}\|$	$4\|\mathbb{G}\|$	$2\|\mathbb{G}\|$
Security		OW-CCA ROM	OW/IND-ID-CCA2 ROM	OW-ID-CCA ROM	OW-ID-CCA ROM	OW/IND-ID-CCA SM	OW/IND-ID-CCA SM
Order		prime	prime	prime	prime	prime	composite

Table 3: Computation time in prime order group

Symbol	E_G	E_T	BP
time (ms)	1.40	0.14	3.71

ferent basic theories, our scheme construction is simpler.

6 Conclusion

IBEET is an important cryptographic scheme for searching encrypted data in cloud computing. It can decrypt ciphertexts and compare ciphertexts to determine whether the corresponding messages are the same or not. There is only a generic IBBET scheme [11] which achieves security in the SM, but the efficiency of it isn't high. In this paper, the first concrete IBEET scheme proposed by us achieves full security in the SM. And to prevent the abuse of trapdoors, we improve our scheme so that each ciphertext corresponds to a different trapdoor. Our schemes don't need to use additional calculations to enhance security, so in the same conditions, the schemes presented by us are more efficient than the generic scheme. More specifically, compared with their scheme, the encryption algorithm is improved by about 75.1%. Based on the subgroup decision problem, we prove that our schemes achieve OW/IND-ID-CCA security in the SM. But by using dual system encryption technology presented by Lewko and Water, the bilinear group on which our schemes are based is of composite order, so they are not very efficient. Thus, our future work is to improve the efficiency of our schemes.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (62002049); and the Sichuan Science and Technology Program (2017GZDZX0002, 2019YFG0506, 2019YFG0503, 2020YFG0292).

References

[1] S. Alornyo, M. Asante, X. Hu, and K. K. Mireku, "Encrypted traffic analytic using identity based encryption with equality test for cloud computing," in *2018 IEEE 7th International Conference on Adaptive Science & Technology (ICAST)*, pp. 1–4. IEEE, 2018.

[2] Seth Alornyo, Acheampong Edward Mensah, and Abraham Opanfo Abbam, "Identity-based public key cryptographic primitive with delegated equality test against insider attack in cloud computing," *International Journal of Network Security*, vol. 22, no. 5, pp. 743–751, 2020.

[3] Seth Alornyo, Yanan Zhao, Guobin Zhu, and Hu Xiong, "Identity based key-insulated encryption with outsourced equality test.," *Int. J. Netw. Secur.*, vol. 22, no. 2, pp. 257–264, 2020.

[4] G. Ateniese, R. Burns, R. Curtmola, J. Herring, L. Kissner, Z. Peterson, and D. Song, "Provable data possession at untrusted stores," in *Proceedings of the 14th ACM conference on Computer and communications security*, pp. 598–609, 2007.

[5] D. Boneh and X. Boyen, "Efficient selective-id secure identity-based encryption without random oracles,"

in *International conference on the theory and applications of cryptographic techniques*, pp. 223–238. Springer, 2004.

[6] D. Boneh, G. D. Crescenzo, R. Ostrovsky, and G. Persiano, "Public key encryption with keyword search," in *International conference on the theory and applications of cryptographic techniques*, pp. 506–522. Springer, 2004.

[7] D. Boneh, E. J. Goh, and K. Nissim, "Evaluating 2-dnf formulas on ciphertexts," in *Theory of cryptography conference*, pp. 325–341. Springer, 2005.

[8] D. Boneh, E. Shen, and B. Waters, "Strongly unforgeable signatures based on computational diffie-hellman," in *International Workshop on Public Key Cryptography*, pp. 229–240. Springer, 2006.

[9] J. Katz, A. Sahai, and B. Waters, "Predicate encryption supporting disjunctions, polynomial equations, and inner products," in *annual international conference on the theory and applications of cryptographic techniques*, pp. 146–162. Springer, 2008.

[10] H. T. Lee, S. Ling, J. H. Seo, and H. X. Wang, "Semi-generic construction of public key encryption and identity-based encryption with equality test," *Information Sciences*, vol. 373, pp. 419–440, 2016.

[11] H. T. Lee, S. Ling, J. H. Seo, H. X. Wang, and T. Y. Youn, "Public key encryption with equality test in the standard model," *Information Sciences*, vol. 516, pp. 89–108, 2020.

[12] H. T. Lee, H. X. Wang, and K. Zhang, "Security analysis and modification of id-based encryption with equality test from acisp 2017," in *Australasian Conference on Information Security and Privacy*, pp. 780–786. Springer, 2018.

[13] A. Lewko and B. Waters, "New techniques for dual system encryption and fully secure hibe with short ciphertexts," in *Theory of Cryptography Conference*, pp. 455–479. Springer, 2010.

[14] H. B. Li, Q. Huang, S. Ma, J. Shen, and W. Susilo, "Authorized equality test on identity-based ciphertexts for secret data sharing via cloud storage," *IEEE Access*, vol. 7, pp. 25409–25421, 2019.

[15] Y. J. Liao, H. J. Chen, W. Huang, R. Mohammed, H. T. Pan, and S. J. Zhou, "Insecurity of an ibeet scheme and an abeet scheme," *IEEE Access*, vol. 7, pp. 25087–25094, 2019.

[16] Y. J. Liao, Y. Fan, Y. K. Liang, Y. L. Liu, and R. Mohammed, "Cryptanalysis of an identity-based encryption scheme with equality test and improvement," *IEEE Access*, vol. 7, pp. 75067–75072, 2019.

[17] X. J. Lin, L. Sun, and H. P. Qu, "Generic construction of public key encryption, identity-based encryption and signcryption with equality test," *Information Sciences*, vol. 453, pp. 111–126, 2018.

[18] Y. H. Ling, S. Ma, Q. Huang, R. Xiang, and X. M. Li, "Group id-based encryption with equality test," in *Australasian Conference on Information Security and Privacy*, pp. 39–57. Springer, 2019.

[19] S. Ma, "Identity-based encryption with outsourced equality test in cloud computing," *Information Sciences*, vol. 328, pp. 389–402, 2016.

[20] H. P. Qu, Z. Yan, X. J. Lin, Q. Zhang, and L. Sun, "Certificateless public key encryption with equality test," *Information Sciences*, vol. 462, pp. 76–92, 2018.

[21] M. Ramadan, Y. J. Liao, F. G. Li, S. J. Zhou, and H. Abdalla, "Ibeet-rsa: Identity-based encryption with equality test over rsa for wireless body area networks," *Mobile Networks and Applications*, vol. 25, no. 1, pp. 223–233, 2020.

[22] B. Waters, "Dual system encryption: Realizing fully secure ibe and hibe under simple assumptions," in *Annual International Cryptology Conference*, pp. 619–636. Springer, 2009.

[23] F. Wu, W. Yao, X. Zhang, Z. M. Zheng, and W. H. Wang, "Identity based privacy information sharing with similarity test in cloud environment," in *International Conference on Cloud Computing and Security*, pp. 69–78. Springer, 2018.

[24] L. B. Wu, Y. B. Zhang, K. R. Choo, and D. B. He, "Efficient and secure identity-based encryption scheme with equality test in cloud computing," *Future Generation Computer Systems*, vol. 73, pp. 22–31, 2017.

[25] L. B. Wu, Y. B. Zhang, K. R. Choo, and D. B. He, "Efficient identity-based encryption scheme with equality test in smart city," *IEEE Transactions on Sustainable Computing*, vol. 3, no. 1, pp. 44–55, 2017.

[26] T. Wu, S. Ma, Y. Mu, and S. K. Zeng, "Id-based encryption with equality test against insider attack," in *Australasian Conference on Information Security and Privacy*, pp. 168–183. Springer, 2017.

[27] G. Yang, C. H. Tan, Q. Huang, and D. S. Wong, "Probabilistic public key encryption with equality test," in *Cryptographers' Track at the RSA Conference*, pp. 119–131. Springer, 2010.

[28] K. Zhang, J. Chen, H. T. Lee, H. F. Qian, and H. x. Wang, "Efficient public key encryption with equality test in the standard model," *Theoretical Computer Science*, vol. 755, pp. 65–80, 2019.

Biography

Zijun Zhou biography. She is currently studying for a Master degree in School of Information and Software Engineering, University of Electronic Science and Technology of China. Her main research interests include cryptography and keyword encryption search.

Yongjian Liao biography. He is currently an associate professor of School of Information and Software Engineering, University of Electronic Science and Technology of China. He received his Ph.D. degree in applied electronic science and technology from College of Information Science and Electronic Engineering, Zhejiang University in 2007. His main research interests include public key

cryptography and information security, in particular, cryptographic protocols.

Ganglin Zhang biography. He is currently pursuing a Master degree in School of Information and Software Engineering, University of Electronic Science and Technology of China. His main research interests include cryptography and artificial intelligence security.

Tingyun Gan is currently studying for a Master degree in School of Information and Software Engineering, University of Electronic Science and Technology of China. His main research interests include security in 5G, security in Internet of Things and cryptography.

Shijie Zhou is currently an associate professor of School of Information and Software Engineering, University of Electronic Science and Technology of China. He received his Ph.D. degree in applied electronic science and technology from College of Information Science and Electronic Engineering, Zhejiang University in 2007. His main research interests include public key cryptography and information security, in particular, cryptographic protocols.