IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 16 July 2023, accepted 30 July 2023, date of publication 7 August 2023, date of current version 15 August 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3302901

RESEARCH ARTICLE

Cryptanalysis of Zhu et al.'s Identity-Based Encryption With Equality Test Without Random Oracles

HYUNG TAE LEE
School of Computer Science and Engineering, Chung-Ang University, Seoul 06974, Republic of Korea
e-mail: hyungtaelee@cau.ac.kr

This work was supported in part by the Chung-Ang University Research Grants in 2021; and in part by the National Research Foundation of Korea (NRF) funded by the Korean Government (MSIT) under Grant NRF-2021R1A2C1007484.

ABSTRACT Recently, Zhu et al. proposed a new identity-based encryption with equality test (IBEwET) in the standard model (IEEE Access, 2023). According to the authors, it was claimed that their proposed construction achieves the indistinguishability against adaptive identity and adaptive chosen ciphertext attacks (IND-ID-CCA) by adversaries who do not have trapdoors for equality tests and the one-wayness against adaptive identity and adaptive chosen ciphertext attacks (OW-ID-CCA) by adversaries who have trapdoors. In this paper, we propose adaptive chosen ciphertext attacks against Zhu et al.'s construction that break the IND-ID-CCA security and the OW-ID-CCA security. Subsequently, we discuss how to fix their scheme so that it achieves the security requirements, as claimed in the original paper, however we confirm that a simple modification is no longer superior to the currently existing IBEwET schemes over bilinear groups in the standard model, obtained by generic constructions for IBEwET. Furthermore, we point out another issue that some operations in the original encryption algorithm are incompatible.

INDEX TERMS Chosen ciphertext attacks, identity-based encryption with equality test, indistinguishability, one-wayness, standard model.

I. INTRODUCTION

Encryption with equality test allows testers to check if two ciphertexts contain the same message or not, regardless of owners of ciphertexts. It can be applied to various practical scenarios, e.g., secure data management on the cloud [1], spam filtering on the encrypted email system [2], secure communication in the Internet of Vehicles [3], and secure telemedicine system [4]. Thus, since Yang et al. firstly proposed its concept with concrete instantiation [1], there have been proposed many encryption schemes with equality test under diverse settings.

Very recently, Zhu et al. [5] proposed a new identity-based encryption with equality test (IBEwET) in the standard model, based on Waters' identity-based encryption (IBE) [6]. In [5], the authors claimed that their proposed scheme achieves the indistinguishability against adaptive identity and adaptive chosen ciphertext attacks (IND-ID-CCA) by adversaries who do not have trapdoors for equality tests as well as the one-wayness against adaptive identity and adaptive chosen ciphertext attacks (OW-ID-CCA) by adversaries who have trapdoors.

In this paper, we propose adaptive chosen ciphertext attacks on Zhu et al.'s IBEwET scheme that break not only the IND-ID-CCA security but also the OW-ID-CCA security. Our attacks are critical in the sense that they break both the IND-ID-CCA security against adversaries who do not have trapdoors and the OW-ID-CCA security against adversaries who have trapdoors. Moreover, they are simple in the sense that they require only 1 pairing computation, 3 exponentiations, and 1 decryption oracle query each.

Thereafter, we discuss how to remedy their scheme so that it achieves the security requirements, as they claimed. To this end, we may replace the underlying IBE scheme with its security-enhanced version. However, unfortunately, this modification is no longer superior to the currently existing

The associate editor coordinating the review of this manuscript and approving it for publication was Jun Wang.

schemes over bilinear groups, which can be obtained by applying generic constructions for IBEwET [7], [8]. Finally, we point out an additional issue that some operations are not compatible in the original encryption algorithm.

Outline of the Paper: Section II provides the formal definitions for IBEwET and presents the related work on IBEwET. We review Zhu et al.'s IBEwET in Section III and provide our attack algorithms against their IBEwET with the analysis in Section IV. We discuss about some issues on their IBEwET in Section V.

II. BACKGROUNDS ON IDENTITY-BASED ENCRYPTION WITH EQUALITY TEST

In this section, we first review the formal definitions of IBEwET including its system model, correctness, and security. Then, we investigate existing IBEwET schemes in the standard model, which are closely related to the work in this paper.

Notation: Throughout the paper, $a \leftarrow A$ denotes that algorithm A outputs a.

A. IDENTITY-BASED ENCRYPTION WITH EQUALITY TEST

1) SYSTEM MODEL FOR IBEwET

An IBEwET can be understood as an identity-based counterpart of public key encryption with equality test. It has various application scenarios in practice. For example, let us elaborate a fundamental application scenario of IBEwET for secure and efficient data management on the cloud. The system for IBEwET is composed of the key generation center (KGC), users including a sender and a receiver, and a tester who can access to the cloud. Once a user would like to join the system, he/she forwards an identity to the KGC and it issues a user's secret key using user's identity. The sender encrypts a message using receiver's identity and sends it to the receiver. Then, the receiver may decrypt it using his/her secret key and stores it to the cloud. When the receiver wants to delegate a right for equality tests of his/her all ciphertexts to the tester, he/she issues a trapdoor and passes it to the tester. Thereafter, the tester can perform equality tests on ciphertexts under the identity of the receiver who passed the trapdoor to the tester.

Beyond the above fundamental application scenario, there are several applications of IBEwET in smart city applications [9], wireless body area network [10], mobile social networking [11], and so on. See [9], [10], and [11] for the detailed application scenarios.

We now recall the formal definition of IBEwET below.

Definition 1 (Identity-Based Encryption With Equality Test): An identity-based encryption with equality test (IBEwET) consists of the following 6 polynomial-time algorithms:

- **Setup**(λ): It takes a security parameter λ as an input and returns a public parameter pp and a master secret key msk.
- **KeyGen**(pp, ID, msk): It takes the public parameter pp, an identity ID and the master secret key msk as inputs, and returns a secret key sk_{ID} for identity ID.
- **Enc**(pp, ID, M): It takes the public parameter pp, an identity ID and a message M as inputs, and returns a ciphertext CT.
- **Dec**(sk_{ID}, CT): It takes the secret key sk_{ID} for identity ID and a ciphertext CT as inputs, and returns a message M.
- **Auth**(sk_{ID}): It takes the secret key sk_{ID} as an input and returns a trapdoor td_{ID} for identity ID.
- **Test**(CT_A, td_A, CT_B, td_B): It takes two pairs of ciphertext and trapdoor as inputs, and returns 1 indicating that CT_A and CT_B contain the same message or 0 indicating that they contain different messages.

2) CORRECTNESS OF IBEwET

Next, we recall the correctness definition of IBEwET. It is composed of three conditions: On the one hand, the first condition guarantees the correctness of recovering the message correctly in the decryption algorithm of IBEwET. On the other hand, the second and last conditions guarantee the correctness of results of the test algorithm of IBEwET.

Definition 2 (Correctness of IBEwET): An IBEwET (**Setup**, **KeyGen**, **Enc**, **Dec**, **Auth**, **Test**) is *correct* if for any security parameter λ, any identities ID_i, ID_j, all $(pp, msk) \leftarrow$ **Setup**(λ), $sk_{\mathsf{ID}_i} \leftarrow$ **KeyGen**(pp, ID_i, msk), and $sk_{\mathsf{ID}_j} \leftarrow$ **KeyGen**(pp, ID_j, msk), the following conditions hold:

1) For any message M, it always holds that

$$\mathbf{Dec}(sk_{\mathsf{ID}_i}, \mathbf{Enc}(pp, \mathsf{ID}_i, M)) = M.$$

2) For any ciphertexts $\mathsf{CT}_{\mathsf{ID}_i}$ and $\mathsf{CT}_{\mathsf{ID}_j}$, if $\mathbf{Dec}(sk_{\mathsf{ID}_i}, \mathsf{CT}_{\mathsf{ID}_i}) = \mathbf{Dec}(sk_{\mathsf{ID}_j}, \mathsf{CT}_{\mathsf{ID}_j}) \neq \perp$, then

$$\Pr[\mathbf{Test}(\mathsf{CT}_{\mathsf{ID}_i}, td_{\mathsf{ID}_i}, \mathsf{CT}_{\mathsf{ID}_j}, td_{\mathsf{ID}_j})] = 1,$$

where $td_{\mathsf{ID}_i} \leftarrow \mathbf{Auth}(sk_{\mathsf{ID}_i})$ and $td_{\mathsf{ID}_j} \leftarrow \mathbf{Auth}(sk_{\mathsf{ID}_j})$.

3) For any ciphertexts $\mathsf{CT}_{\mathsf{ID}_i}$ and $\mathsf{CT}_{\mathsf{ID}_j}$, if $\mathbf{Dec}(sk_{\mathsf{ID}_i}, \mathsf{CT}_{\mathsf{ID}_i}) \neq \mathbf{Dec}(sk_{\mathsf{ID}_j}, \mathsf{CT}_{\mathsf{ID}_j})$, then

$$\Pr[\mathbf{Test}(\mathsf{CT}_{\mathsf{ID}_i}, td_{\mathsf{ID}_i}, \mathsf{CT}_{\mathsf{ID}_j}, td_{\mathsf{ID}_j})] \leq \mathsf{negl}(\lambda),$$

where $td_{\mathsf{ID}_i} \leftarrow \mathbf{Auth}(sk_{\mathsf{ID}_i})$, $td_{\mathsf{ID}_j} \leftarrow \mathbf{Auth}(sk_{\mathsf{ID}_j})$, and $\mathsf{negl}(\lambda)$ is a negligible function in λ.

B. SECURITY MODELS FOR IBEwET

For encryption schemes that support equality tests, we consider two types of adversaries with respect to whether they possess trapdoors for equality test or not.

- Type-I adversary: It is assumed that this type of adversaries can have trapdoors for equality test. So, it can perform equality test on the target ciphertext and distinguish whether which message is contained in the target ciphertext between two candidates. Thus, it is impossible to achieve the indistinguishability against this type of adversaries and we assume that the aim of this type of adversaries is to recover a message in the target ciphertext, i.e., to break the one-wayness of the scheme.

- Type-II adversary: It is assumed that this type of adversaries does not have trapdoors for equality test. So, conditions given to this type of adversaries are the same as those of traditional encryption schemes. Thus, we assume that the aim of this type of adversaries is to distinguish a message in the target ciphertext between two candidates, as in traditional encryption schemes.

Below we formalize security definitions for the above two types of adversaries, respectively.

Definition 3 (OW-ID-CCA Security for Type-I Adversaries): An IBEwET is OW-ID-CCA secure if for any probabilistic polynomial time (PPT) adversary $\mathcal{A}$ its advantage is negligible in the security parameter λ in the following game played with the challenger $\mathcal{C}$:

1) **Setup**: $\mathcal{C}$ runs the setup algorithm $\mathbf{Setup}(\lambda)$ to obtain the public parameter pp and the master secret key msk. $\mathcal{C}$ passes pp to $\mathcal{A}$.
2) **Phase 1**: $\mathcal{A}$ may request the following queries to the oracles polynomially many times in any order:
   - $\mathcal{O}^{\text{KeyGen}}$: On $\mathcal{A}$'s request of a query on identity ID_i, it responds a secret key sk_{ID_i} for ID_i.
   - $\mathcal{O}^{\text{Dec}}$: On $\mathcal{A}$'s request of a query on a pair of identity ID_i and ciphertext CT_i, it runs the decryption algorithm and returns the resulting message M_i to $\mathcal{A}$.
   - $\mathcal{O}^{\text{Auth}}$: On $\mathcal{A}$'s request of a query on identity ID_i, it responds a trapdoor td_{ID_i} for ID_i.
3) **Challenge**: $\mathcal{A}$ submits an identity ID^* which was never queried to $\mathcal{O}^{\text{KeyGen}}$. $\mathcal{C}$ picks a random message M from the message space and runs $\mathbf{Enc}(pp, \mathsf{ID}^*, M)$ to get $\mathsf{CT}^*_{\mathsf{ID}^*}$. $\mathcal{C}$ passes $\mathsf{CT}^*_{\mathsf{ID}^*}$ as the challenge ciphertext to $\mathcal{A}$.
4) **Phase 2**: $\mathcal{A}$ may request queries to $\mathcal{O}^{\text{KeyGen}}$, $\mathcal{O}^{\text{Dec}}$, and $\mathcal{O}^{\text{Auth}}$ polynomially many times in any order. $\mathcal{C}$ responds as the same way as in **Phase 1**. There are some constraints for $\mathcal{A}$ that
   - ID^* should not be requested to $\mathcal{O}^{\text{KeyGen}}$.
   - $(\mathsf{ID}^*, \mathsf{CT}^*_{\mathsf{ID}^*})$ should not be requested to $\mathcal{O}^{\text{Dec}}$.
5) **Guess**: $\mathcal{A}$ returns M'.

The advantage of $\mathcal{A}$ in the above game is defined as

$$Adv^{\text{OW-ID-CCA}}_{\mathcal{A},\text{IBEwET}}(\lambda) = \Pr[M = M'].$$

Definition 4 (IND-ID-CCA Security for Type-II Adversaries): An IBEwET is IND-ID-CCA secure if for any PPT adversary $\mathcal{A}$ its advantage is negligible in the security parameter λ in the following game played with the challenger $\mathcal{C}$:

1) **Setup**: This phase is exactly the same as that of the OW-ID-CCA security game.
2) **Phase 1**: This phase is also exactly the same as that of the OW-ID-CCA security game.
3) **Challenge**: $\mathcal{A}$ submits an identity ID^* and two messages M_0, M_1 to $\mathcal{C}$ where ID^* was never queried to $\mathcal{O}^{\text{KeyGen}}$ and $\mathcal{O}^{\text{Auth}}$. $\mathcal{C}$ tosses a fair coin $b \in \{0, 1\}$ and runs $\mathbf{Enc}(pp, \mathsf{ID}^*, M_b)$ to get $\mathsf{CT}^*_{\mathsf{ID}^*,b}$. $\mathcal{C}$ passes $\mathsf{CT}^*_{\mathsf{ID}^*,b}$ as the challenge ciphertext to $\mathcal{A}$.
4) **Phase 2**: This phase is almost the same as that of the OW-ID-CCA security game, except that there is an additional constraint that ID^* should not be queried to $\mathcal{O}^{\text{Auth}}$.
5) **Guess**: $\mathcal{A}$ returns b'.

The advantage of $\mathcal{A}$ in the above game is defined as

$$Adv^{\text{IND-ID-CCA}}_{\mathcal{A},\text{IBEwET}}(\lambda) = |\Pr[b = b'] - 1/2|.$$

C. RELATED WORK ON IBEwET IN THE STANDARD MODEL

Since Yang et al. [1] firstly proposed the concept of public key encryption with equality test, there have been proposed various types of encryption schemes supporting equality tests. Among them, Ma [12] firstly proposed the concept of IBEwET which is an encryption scheme supporting equality tests under the identity-based setting. She also proposed an instantiation of IBEwET over bilinear groups in the random oracle model. Following her first proposal, there were presented numerous IBEwET constructions in the random oracle model or the standard model. In this subsection, we briefly investigate IBEwET schemes in the standard model, which are closely related to our work.

To avoid random oracle heuristics, there have been proposed several IBEwET schemes in the standard model. First, Lee et al. [7] proposed a generic construction for IBEwET that achieves CCA security from a 3-level hierarchical IBE (HIBE) and a one-time signature scheme by using the Canetti-Halevi-Katz (CHK) transformation [13]. Duong et al. [14] presented a concrete lattice-based instantiation of IBEwET which outperforms the lattice-based outcome obtained by Lee et al.'s generic construction. However, their construction is secure against chosen plaintext attacks (CPA) only. Later, Susilo et al. [15] provided a security enhanced version of lattice-based IBEwET that achieves CCA security. However, their construction achieves the selective identity security only, where the challenge identity should be chosen before seeing the public parameter in the security game. Nguyen et al. [16] extended types of equality tests of lattice-based IBEwET in the standard model. Wu et al. [17] presented an efficient IBEwET from lattices that achieves IND-ID-CPA security. They remarked that one can easily obtain CCA secure version by applying the CHK transformation to their construction, but the detailed description was omitted. Recently, Qu et al. [18] proposed an efficient lattice-based IBEwET construction, but it is secure under the selective identity setting against Type-I adversaries who have trapdoors for equality test, while being secure under the adaptive identity setting against Type-II adversaries who do not have. Independently, Asono et al. [8] proposed a generic construction for IBEwET which outperforms Lee et al.'s one. While Lee et al.'s approach employs a CCA secure 3-level HIBE scheme which is adaptive identity secure for all levels, Asono et al.'s approach requires a CCA secure 3-level HIBE scheme that is adaptive identity secure for the first level and selective identity secure for other levels. Very recently, Zhu et al. [5] proposed an IBEwET scheme over bilinear groups. To the best of our knowledge, it is the first scheme constructed over bilinear groups in the standard model, except instantiations obtained by generic constructions for IBEwET. However, throughout the paper, we demonstrate that their construction is not secure, contrary to the authors claimed.

III. ZHU ET AL.'S IDENTITY-BASED ENCRYPTION WITH EQUALITY TEST

In this section, we review the description of Zhu et al.'s IBEwET [5] below, which is composed of 6 polynomial-time algorithms.

- **Setup**(λ): Given a security parameter λ,
  1) Generate a bilinear group parameter that includes
     - two cyclic groups $\mathbb{G}, \mathbb{G}_T$ of prime order p,
     - a generator g of $\mathbb{G}$, and
     - a bilinear map $e : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_T$.
  2) Select a random element α from $\mathbb{Z}_p$.
  3) Compute and set $g_1 = g^\alpha$.
  4) Pick a random element g_2 from $\mathbb{G}$.
  5) Compute g_1^α and g_2^α as the master secret key.
  6) Choose a random element u' from $\mathbb{G}$ and a random vector $U = (u_i)_{i=1}^n$ where each element u_i is selected at random from $\mathbb{G}$.
  7) Generate a collision-resistant hash function $H : \{0,1\}^* \to \{0,1\}^n$.
  8) Output the public parameter pp and the master secret key msk:
  $$pp = \langle \mathbb{G}, \mathbb{G}_T, e, H, p, g, g_1, g_2, u', U \rangle,$$
  $$msk = (g_1^\alpha, g_2^\alpha).$$

- **KeyGen**(pp, ID, msk): On input the public parameter pp, an identity ID, and the master secret key $msk = (g_1^\alpha, g_2^\alpha)$,
  1) Compute $H(\mathsf{ID}) = (v_1, \cdots, v_n) \in \{0,1\}^n$ and let $v = \{i \in \{1, 2, \cdots, n\} \mid v_i = 1\}$.
  2) Pick two random numbers s, s' from $\mathbb{Z}_p$.
  3) Compute and output $sk_{\mathsf{ID}} = (d, d')$ where
  $$d = \big(g_1^\alpha \big(u' \prod_{i \in v} u_i\big)^s, g^s\big) \text{ and}$$
  $$d' = \big(g_2^\alpha \big(u' \prod_{i \in v} u_i\big)^{s'}, g^{s'}\big).$$

- **Enc**(pp, ID, M): Given the public parameter pp, an identity ID, and a message $M \in \mathbb{G}_T$,
  1) Select three random numbers r_1, r_2, r_3 from $\mathbb{Z}_p$.
  2) Compute $H(\mathsf{ID}) = (v_1, \cdots, v_n) \in \{0,1\}^n$ and let $v = \{i \in \{1, 2, \cdots, n\} \mid v_i = 1\}$.
  3) Compute and output $\mathsf{CT} = (C_1, C_2, C_3)$ where
  $$C_1 = g^{r_1},$$
  $$C_2 = \Big(M^{r_1} e(g_1, g_2)^{r_2}, g^{r_2}, \big(u' \prod_{i \in v} u_i\big)^{r_2}\Big), \text{ and}$$
  $$C_3 = \Big((M \| r_1) e(g_1, g_2)^{r_3}, g^{r_3}, \big(u' \prod_{i \in v} u_i\big)^{r_3}\Big).$$

- **Dec**($sk_{\mathsf{ID}}, \mathsf{CT}$): On input the secret key $sk_{\mathsf{ID}} = (d, d')$ for identity ID and the ciphertext $\mathsf{CT} = (C_1, C_2, C_3)$ where $d = (d_1, d_2), d' = (d_1', d_2'), C_2 = (C_{2,1}, C_{2,2}, C_{2,3})$, and $C_3 = (C_{3,1}, C_{3,2}, C_{3,3})$, it performs as follows:
  1) Compute $M \| r_1 = C_{3,1} \dfrac{e(d_2', C_{3,3})}{e(d_1', C_{3,2})}$.
  2) Check if the following relations hold:
  $$C_1 = g^{r_1} \quad \text{and} \quad C_{2,1} \frac{e(d_2, C_{2,3})}{e(d_1, C_{2,2})} = M^{r_1}.$$
  If both hold, then output M. Otherwise, output $\perp$.

- **Auth**(sk_{ID}): On input the secret key $sk_{\mathsf{ID}} = (d, d')$ for identity ID, it returns $td_{\mathsf{ID}} = d$.

- **Test**($\mathsf{CT}_A, td_A, \mathsf{CT}_B, td_B$): Given two pairs of ciphertext and trapdoor, (CT_A, td_A) and (CT_B, td_B) for ID_A and ID_B, respectively, where $\mathsf{CT}_A = (C_{A,1}, C_{A,2}, C_{A,3})$, $\mathsf{CT}_B = (C_{B,1}, C_{B,2}, C_{B,3})$, $C_{A,2} = (C_{A,2,1}, C_{A,2,2}, C_{A,3,3})$, $C_{B,2} = (C_{B,2,1}, C_{B,2,2}, C_{B,3,3})$, $td_A = (d_{A,1}, d_{A,2})$, and $td_B = (d_{B,1}, d_{B,2})$, it performs as follows:
  1) Compute
  $$X_A = C_{A,2,1} \frac{e(d_{A,2}, C_{A,2,3})}{e(d_{A,1}, C_{A,2,2})} \text{ and}$$
  $$X_B = C_{B,2,1} \frac{e(d_{B,2}, C_{B,2,3})}{e(d_{B,1}, C_{B,2,2})}.$$
  2) Check if
  $$e(C_{B,1}, X_A) = e(C_{A,1}, X_B)$$
  If it holds, return 1. Otherwise, return 0.

IV. OUR ATTACKS ON ZHU ET AL.'S CONSTRUCTION

In this section, we present our adaptive chosen ciphertext attacks that break the IND-ID-CCA security and the OW-ID-CCA security of Zhu et al.'s construction.

A. DESCRIPTION OF OUR ATTACK

We first provide an adaptive chosen ciphertext attack against Zhu et al.'s scheme that breaks the IND-ID-CCA security. The flowchart of our attack algorithm is given in Figure 1.

Let $\mathcal{A}$ be a Type-II adversary and consider the IND-ID-CCA security game between $\mathcal{A}$ and the challenger $\mathcal{C}$. At the **Challenge** phase of the security game, suppose that the challenge ciphertext $\mathsf{CT}^*_{\mathsf{ID}^*,b} = (C_1^*, C_2^*, C_3^*)$ is given to $\mathcal{A}$. In particular, let $C_2^* = (C_{2,1}^*, C_{2,2}^*, C_{2,3}^*)$. Then, $\mathcal{A}$ selects a random element $\overline{r}_2$ from $\mathbb{Z}_p$ and calculates

$$\overline{C}_{2,1} = C_{2,1}^* \cdot e(g_1, g_2)^{\overline{r}_2},$$

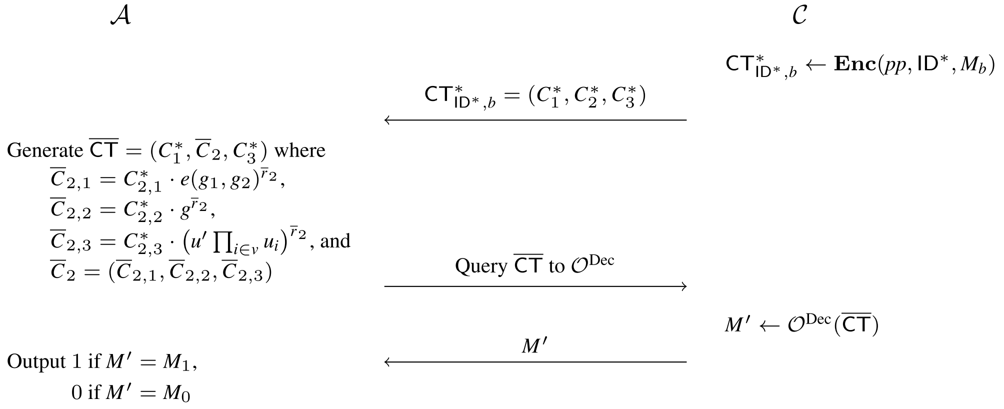
$\mathcal{A}$
$\mathcal{C}$
$\mathsf{CT}^*_{\mathsf{ID}^*,b} \leftarrow \mathbf{Enc}(pp, \mathsf{ID}^*, M_b)$
$\mathsf{CT}^*_{\mathsf{ID}^*,b} = (C_1^*, C_2^*, C_3^*)$
Generate $\overline{\mathsf{CT}} = (C_1^*, \overline{C}_2, C_3^*)$ where
$\overline{C}_{2,1} = C^*_{2,1} \cdot e(g_1, g_2)^{\overline{r}_2}$,
$\overline{C}_{2,2} = C^*_{2,2} \cdot g^{\overline{r}_2}$,
$\overline{C}_{2,3} = C^*_{2,3} \cdot \left(u' \prod_{i\in\mathcal{V}} u_i\right)^{\overline{r}_2}$, and
$\overline{C}_2 = (\overline{C}_{2,1}, \overline{C}_{2,2}, \overline{C}_{2,3})$
Query $\overline{\mathsf{CT}}$ to $\mathcal{O}^{\mathrm{Dec}}$
$M' \leftarrow \mathcal{O}^{\mathrm{Dec}}(\overline{\mathsf{CT}})$
M'
Output 1 if $M' = M_1$,
0 if $M' = M_0$

FIGURE 1. Our attack executed by the adversary $\mathcal{A}$ interacting with the challenger $\mathcal{C}$; For an algorithm A, $a \leftarrow A$ indicates that A outputs a.

Algorithm 1 Our Attack Algorithm for Breaking the IND-ID-CCA Security

```
Input: The challenge ciphertext $\mathsf{CT}^*_{\mathsf{ID}^*,b} = (C_1^*, C_2^*, C_3^*)$
    and challenge messages $M_0, M_1$
Output: A bit $b'$ that indicates the challenge message $M_{b'}$
  $\overline{\mathsf{CT}} \leftarrow \mathbf{MC}(\mathsf{CT}^*_{\mathsf{ID}^*,b})$
  $M' \leftarrow \mathcal{O}^{\mathrm{Dec}}(\overline{\mathsf{CT}})$  /* A decryption oracle query to $\mathcal{O}^{\mathrm{Dec}}$ */
  $b' \leftarrow \mathbf{DB}(M', M_0, M_1)$
  Return $b'$
```

Algorithm 2 Manufacturing a Ciphertext: $\mathbf{MC}(\mathsf{CT}^*_{\mathsf{ID}^*,b})$

```
Input: The challenge ciphertext $\mathsf{CT}^*_{\mathsf{ID}^*,b} = (C_1^*, C_2^*, C_3^*)$
Output: A manufactured ciphertext $\overline{\mathsf{CT}}$
  Select a random number $\overline{r}_2$ from $\mathbb{Z}_p$
  Parse $C_2^*$ as $(C^*_{2,1}, C^*_{2,2}, C^*_{2,3})$
  Compute $\overline{C}_{2,1} = C^*_{2,1} \cdot e(g_1, g_2)^{\overline{r}_2}$
  Compute $\overline{C}_{2,2} = C^*_{2,2} \cdot g^{\overline{r}_2}$
  Compute $\overline{C}_{2,3} = C^*_{2,3} \cdot \left(u' \prod_{i\in\mathcal{V}} u_i\right)^{\overline{r}_2}$
  Set $\overline{C}_2 = (\overline{C}_{2,1}, \overline{C}_{2,2}, \overline{C}_{2,3})$
  Set $\overline{\mathsf{CT}} = (C_1^*, \overline{C}_2, C_3^*)$
  Return $\overline{\mathsf{CT}}$
```

$$\overline{C}_{2,2} = C^*_{2,2} \cdot g^{\overline{r}_2}, \text{ and}$$
$$\overline{C}_{2,3} = C^*_{2,3} \cdot \left(u' \prod_{i\in\mathcal{V}} u_i\right)^{\overline{r}_2}.$$

Thereafter, $\mathcal{A}$ sets $\overline{C}_2 = (\overline{C}_{2,1}, \overline{C}_{2,2}, \overline{C}_{2,3})$ and then $\overline{\mathsf{CT}} = (C_1^*, \overline{C}_2, C_3^*)$.

Later, at **Phase 2** of the security game, $\mathcal{A}$ requests a decryption query on $\overline{\mathsf{CT}}$ and receives M' from the decryption oracle. Then, $\mathcal{A}$ confirms that which one is the same as M' between M_0 and M_1 which are challenge messages submitted to $\mathcal{C}$ at the **Challenge** phase of the security game. Finally, $\mathcal{A}$ returns 1 if $M' = M_1$ and 0 if $M' = M_0$.

Algorithm 3 Determine a Bit: $\mathbf{DB}(M', M_0, M_1)$

```
Input: A message $M'$ and two challenge message $M_0, M_1$
Output: A bit $b'$
  if $M' = M_1$ then
      Set $b' = 1$
  else
      Set $b' = 0$
  Return $b'$
```

For better understanding of readers, we also present the pseudo-code of the proposed attack by focusing on $\mathcal{A}$'s behaviours. Our main attack algorithm, presented in Algorithm 1, calls two sub-algorithms, **MC** and **DB**. The former algorithm **MC**, described in Algorithm 2, takes the challenge ciphertext $\mathsf{CT}_{\mathsf{ID}^*,b}$ as an input and returns a manufactured chiphertext $\overline{\mathsf{CT}}$ which contains the same message as the challenge ciphertext. The latter algorithm **DB** takes a message M' and two challenge messages M_0, M_1 as inputs, and returns a bit b' that indicates $M' = M_{b'}$. Between executions of two sub-algorithms, our attack algorithm requests a decryption query on the resulting ciphertext $\overline{\mathsf{CT}}$ of the algorithm **MC** to the decryption oracle $\mathcal{O}^{\mathrm{Dec}}$.

B. ANALYSIS OF OUR ATTACK

1) CORRECTNESS OF OUR ATTACK ALGORITHM

We first check the correctness of our attack described in the previous subsection. Suppose that $\mathsf{CT}^*_{\mathsf{ID}^*,b} = (C_1^*, C_2^*, C_3^*)$ is the challenge ciphertext of message M_b where b is a bit randomly selected by the challenger $\mathcal{C}$ at the **Challenge** phase. That is, $\mathsf{CT}^*_{\mathsf{ID}^*,b}$ is the form of

$$C_1^* = g^{r_1^*},$$
$$C_2^* = \left(M_b^{r_1^*} e(g_1, g_2)^{r_2^*}, g^{r_2^*}, \left(u' \prod_{i\in\mathcal{V}} u_i\right)^{r_2^*}\right), \text{ and}$$
$$C_3^* = \left((M_b \| r_1^*) e(g_1, g_2)^{r_3^*}, g^{r_3^*}, \left(u' \prod_{i\in\mathcal{V}} u_i\right)^{r_3^*}\right)$$

where r_1^*, r_2^*, r_3^* are random numbers selected from $\mathbb{Z}_p$ by the encryption algorithm. Thus, after the execution of the algorithm for manufacturing a ciphertext, $\mathbf{MC}(\mathsf{CT}^*_{\mathsf{ID}^*,b})$, it holds that

$$\begin{aligned}\overline{C}_{2,1} &= C^*_{2,1} \cdot e(g_1, g_2)^{\overline{r}_2} = M_b^{r_1^*} e(g_1, g_2)^{r_2^*+\overline{r}_2},\\ \overline{C}_{2,2} &= C^*_{2,2} \cdot g^{\overline{r}_2} = g^{r_2^*+\overline{r}_2},\\ \overline{C}_{2,3} &= C^*_{2,3} \cdot \big(u' \prod_{i\in v} u_i\big)^{\overline{r}_2} = \big(u' \prod_{i\in v} u_i\big)^{r_2^*+\overline{r}_2}\end{aligned}$$

and so $\overline{\mathsf{CT}} = (C_1^*, \overline{C}_2, C_3^*)$ is the form of

$$\begin{aligned}C_1^* &= g^{r_1^*},\\ \overline{C}_2 &= \Big(M_b^{r_1^*} e(g_1, g_2)^{r_2^*+\overline{r}_2}, g^{r_2^*+\overline{r}_2}, \big(u' \prod_{i\in v} u_i\big)^{r_2^*+\overline{r}_2}\Big), \text{ and}\\ C_3^* &= \Big((M_b \| r_1^*) e(g_1, g_2)^{r_3^*}, g^{r_3^*}, \big(u' \prod_{i\in v} u_i\big)^{r_3^*}\Big)\end{aligned}$$

which is still a valid ciphertext of message M_b where only the second randomness r_2^* is replaced by $r_2^* + \overline{r}_2$. Therefore, the decryption oracle $\mathcal{O}^{\mathsf{Dec}}$ returns $M' = M_b$ and $\mathcal{A}$ always outputs $b' = b$ correctly after the execution of the algorithm $\mathbf{DB}(M', M_0, M_1)$.

2) COMPLEXITY OF OUR ATTACK ALGORITHM

In the viewpoint of efficiency, our attack algorithm is very simple: It calls algorithms **MC**, **DB**, and a decryption oracle once each. For **MC**, it requires

- 1 pairing computation and 1 exponentiation for calculating $e(g_1, g_2)$ and then $e(g_1, g_2)^{\overline{r}_2}$ at Step 3,
- 1 exponentiation for calculating $g^{\overline{r}_2}$ at Step 4, and
- n multiplications and 1 exponentiation for calculating $\big(u' \prod_{i\in v} u_i\big)^{\overline{r}_2}$ at Step 5.

where n is the output size of hash function H. For **DB**, it just requires one comparison of two messages.

Remark 1: We may break Zhu et al.'s construction by modifying C_3^*, instead of C_2^*. Similarly, in Algorithm 2, it selects $\overline{r}_3$ and calculates

$$\begin{aligned}\overline{C}_{3,1} &= C^*_{3,1} \cdot e(g_1, g_2)^{\overline{r}_3},\\ \overline{C}_{3,2} &= C^*_{3,2} \cdot g^{\overline{r}_3},\\ \overline{C}_{3,3} &= C^*_{3,3} \cdot \big(u' \prod_{i\in v} u_i\big)^{\overline{r}_3},\end{aligned}$$

instead of $\overline{C}_{2,1}, \overline{C}_{2,2}, \overline{C}_{2,3}$, respectively, where $\mathsf{CT}^*_{\mathsf{ID}^*,b} = (C_1^*, C_2^*, C_3^*)$ and $C_3^* = (C^*_{3,1}, C^*_{3,2}, C^*_{3,3})$. Then, $\mathcal{A}$ obtains M_b by requesting a decryption query on $\overline{\mathsf{CT}}' = (C_1^*, C_2^*, \overline{C}_3)$ where $\overline{C}_3 = (\overline{C}_{3,1}, \overline{C}_{3,2}, \overline{C}_{3,3})$. Of course, we may also break the scheme by modifying C_2^* and C_3^* simultaneously.

C. BREAKING THE OW-ID-CCA SECURITY

So far, we have presented an adaptive chosen ciphertext attack against Zhu et al.'s IBEwET and have shown that it breaks the IND-ID-CCA security of Zhu et al.'s scheme. We remark that, in fact, our proposed attack can also break the OW-ID-CCA security of Zhu et al.'s construction by slightly modifying $\mathcal{A}$'s output.

Let us further elaborate the detailed process of our attack that breaks the OW-ID-CCA security. At the **Challenge** phase of the OW-ID-CCA security game, suppose that $\mathcal{A}$ receives the challenge ciphertext $\mathsf{CT}^*_{\mathsf{ID}^*}$ from $\mathcal{C}$. Then, $\mathcal{A}$ manipulates $\mathsf{CT}^*_{\mathsf{ID}^*}$ as $\mathsf{CT}^*_{\mathsf{ID}^*,b}$ of our proposed attack described in Section IV-A, by running the algorithm $\mathbf{MC}(\mathsf{CT}^*_{\mathsf{ID}^*})$. Let us denote the output of $\mathbf{MC}(\mathsf{CT}^*_{\mathsf{ID}^*})$ by $\overline{\mathsf{CT}}$. Then, $\overline{\mathsf{CT}}$ is a valid ciphertext of message M which is selected by $\mathcal{C}$ at the **Challenge** phase. Thus, $\mathcal{A}$ can obtain the message M by requesting the decryption query on $\overline{\mathsf{CT}}$ to the decryption oracle. Therefore, our attack breaks the OW-ID-CCA security of Zhu et al.'s construction by outputting the resulting message M.

We also provide the pseudo-code of our attack algorithm for breaking the OW-ID-CCA security of Zhu et al.'s construction in Algorithm 4. The correctness of Algorithm 4 is straightforward from the above and the efficiency of Algorithm 4 is almost the same as that of Algorithm 1.

Algorithm 4 Our Attack Algorithm for Breaking the OW-ID-CCA Security

```
Input: The challenge ciphertext $\mathsf{CT}^*_{\mathsf{ID}^*} = (C_1^*, C_2^*, C_3^*)$
Output: A message $M'$
  $\overline{\mathsf{CT}} \leftarrow \mathbf{MC}(\mathsf{CT}^*_{\mathsf{ID}^*})$
  $M' \leftarrow \mathcal{O}^{\mathsf{Dec}}(\overline{\mathsf{CT}})$  /* A decryption oracle query to $\mathcal{O}^{\mathsf{Dec}}$ */
  Return $M'$
```

V. DISCUSSIONS

In this section, we discuss additional issues on Zhu et al.'s IBEwET. We first consider a potential method to fix their scheme so that it is secure against chosen ciphertext attacks (CCA) and show that it is hard to obtain an IBEwET scheme over bilinear groups that significantly outperforms existing IBEwET constructions in the standard model from a naive modification. Next, we point out that some operations in their encryption algorithm are incompatible.

A. ACHIEVING CCA SECURITY

Zhu et al.'s construction was designed based on a weaker version of Waters' IBE scheme [6] which satisfies the indistinguishability against adaptive identity and chosen plaintext attacks (IND-ID-CPA), not IND-ID-CCA security. Informally, the encryption algorithm of Waters' IBE works as follows: Given the public parameter $pp = \langle \mathbb{G}, \mathbb{G}_T, e, H, p, g, g_1, g_2, u', U\rangle$, an identity ID, and a message $M \in \mathbb{G}_T$, it selects a random integer r and computes

$$\mathbf{W}.\mathbf{Enc}(pp, \mathsf{ID}, M; r) := (M \cdot e(g_1, g_2)^r, g^r, \big(u' \prod_{i\in v} u_i\big)^r)$$

where $H(\mathsf{ID}) = (v_1, \cdots, v_n) \in \{0, 1\}^n$ and $v = \{i \in \{1, 2, \cdots, n\} \mid v_i = 1\}$. We observe that the encryption algorithm of Zhu et al.'s IBEwET can be understood as a combination of two encryption algorithms of Waters' IBE with messages $M\|r_1$ and M^{r_1}, respectively. That is, a ciphertext $\mathsf{CT} = (C_1, C_2, C_3)$ of Zhu et al.'s IBEwET can be understood as

$$C_1 = g^{r_1},$$
$$C_2 = \mathbf{W}.\mathbf{Enc}(pp, \mathsf{ID}, M^{r_1}; r_2), \text{ and}$$
$$C_3 = \mathbf{W}.\mathbf{Enc}(pp, \mathsf{ID}, M\|r_1; r_3).$$

So, when we regard that M^{r_1} is a kind of hash value of M, we can view that C_3 and C_2 are ciphertexts of message and its hash value, respectively. Then, a link between C_2 and C_3 is given by using r_1 and C_1 so that it is hard for adversaries to manipulate ciphertexts in the security game. However, the exploited Waters' IBE scheme achieves the IND-ID-CPA security only, thus the adversary can modify C_3 and C_2 so that they are valid ciphertexts of $M\|r_1$ and M^{r_1}, respectively, without knowing M and r_1. Therefore, our attack succeeds.

In order to avoid our attack, we may replace the current underlying IBE scheme with an IND-ID-CCA version of Waters' IBE. According to [6], we can obtain it by applying the generic transformation, e.g., CHK transformation [13] or its improved version by Boneh and Katz [19]. However, such transformations require a 2-level hierarchical identity-based encryption (HIBE) scheme and a strongly unforgeable (SUF) one-time signature scheme. Thus, each C_2 and C_3 in a ciphertext of Zhu et al.'s scheme would be replaced by a pair of a ciphertext of 2-level HIBE scheme and a signature of SUF one-time signature scheme.

Let us take a deep look at the naive modification of Zhu et al.'s IBEwET. To avoid readers' misunderstanding, we first emphasize that our final modification is no longer superior to the instantiations obtained by generic constructions [7], [8] for IBEwET, in terms of efficiency. The main purpose to visit this modification is to understand why it is hard to obtain IBEwET schemes over bilinear groups in the standard model that (significantly) outperform outcomes of generic constructions.

Suppose we have a 2-level IND-ID-CPA secure Waters' HIBE scheme $\mathbf{W}^{(2)} = (\mathbf{Setup}, \mathbf{KeyGen}, \mathbf{Enc}, \mathbf{Dec})$ that is obtained by combining IND-ID-CPA secure Waters' IBE at the first level and Boneh and Boyen's HIBE [20] at the second level, as introduced in [6]. To apply the CHK transformation for obtaining CCA security, we employ Boneh, Shen, and Waters (BSW)'s SUF signature scheme [21], denoted by $\Sigma = (\mathbf{KG}, \mathbf{S}, \mathbf{V})$, where $\mathbf{KG}$, $\mathbf{S}$, and $\mathbf{V}$ indicate the key generation, sign, and verify algorithms, respectively, which run as follows:

- $\mathbf{KG}(\lambda)$: It takes the security parameter λ as an input, and returns a verification key vk and a secret key sk.
- $\mathbf{S}(sk, M)$: It takes the secret key sk and a message M as inputs, and returns a signature σ.
- $\mathbf{V}(vk, \sigma)$: It takes the verification key vk and the signature σ as inputs, and returns 1 that indicates σ is valid or 0 that indicates σ is not valid.

To simplify the explanation of our modification, we try to avoid the use of explicit algorithm descriptions of Waters' HIBE and BSW signature schemes as much as possible, and exploit the forms of algorithms only. Refer to [6] and [21] for the details of algorithms of those schemes. Then, we can obtain the simple modification of Zhu et al.'s IBEwET as follows.

- $\mathbf{Setup}'(\lambda)$: Given a security parameter λ, it performs as follows:
  1) Generate a BSW signature $\Sigma = (\mathbf{KG}, \mathbf{S}, \mathbf{V})$.
  2) Generate two pairs of public parameter and master secret key of Waters' 2-level HIBE scheme as follows:
     a) Run $(pp_1, msk_1) \leftarrow \mathbf{W}^{(2)}.\mathbf{Setup}(\lambda)$
     b) Run $(pp_2, msk_2) \leftarrow \mathbf{W}^{(2)}.\mathbf{Setup}(\lambda)$
  3) Set and output the public parameter pp and the master secret key msk,
  $$pp = (\Sigma, pp_1, pp_2) \text{ and } msk = (msk_1, msk_2).$$
- $\mathbf{KeyGen}'(pp, \mathsf{ID}, msk)$: Given the public parameter pp, an identity ID, and the master secret key msk, it performs as follows:
  1) Run $sk_{1,\mathsf{ID}} \leftarrow \mathbf{W}^{(2)}.\mathbf{KeyGen}(pp_1, \mathsf{ID}, msk_1)$
  2) Run $sk_{2,\mathsf{ID}} \leftarrow \mathbf{W}^{(2)}.\mathbf{KeyGen}(pp_2, \mathsf{ID}, msk_2)$
  3) Set and output $sk_{\mathsf{ID}} = (sk_{1,\mathsf{ID}}, sk_{2,\mathsf{ID}})$.
- $\mathbf{Enc}'(pp, \mathsf{ID}, M)$: Given the public parameter pp, an identity ID, and a message M, it performs as follows:
  1) Select a random element r_1 from $\mathbb{Z}_p$.
  2) Compute $C_1 = g^{r_1}$ where g is in pp_1.
  3) Run
     a) $(vk_1, sk_1) \leftarrow \Sigma.\mathbf{KG}(\lambda)$.
     b) $\mathsf{CT}_1 \leftarrow \mathbf{W}^{(2)}.\mathbf{Enc}(pp_1, [\mathsf{ID}, vk_1], M^{r_1})$
     c) $\sigma_1 \leftarrow \Sigma.\mathbf{S}(sk_1, \mathsf{CT}_1)$.
  4) Run
     a) $(vk_2, sk_2) \leftarrow \Sigma.\mathbf{KG}(\lambda)$.
     b) $\mathsf{CT}_2 \leftarrow \mathbf{W}^{(2)}.\mathbf{Enc}(pp_2, [\mathsf{ID}, vk_2], M\|r_1)$
     c) $\sigma_2 \leftarrow \Sigma.\mathbf{S}(sk_2, \mathsf{CT}_2)$.
  5) Output $\mathsf{CT} = (C_1, (\mathsf{CT}_1, vk_1, \sigma_1), (\mathsf{CT}_2, vk_2, \sigma_2))$.

  Note that a symbol $[\mathsf{ID}_1, \mathsf{ID}_2]$ denotes a 2-level identity where each ID_i is the i-th level identity, throughout this section.
- $\mathbf{Dec}'(sk_{\mathsf{ID}}, \mathsf{CT})$: Given the secret key $sk_{\mathsf{ID}} = (sk_{1,\mathsf{ID}}, sk_{2,\mathsf{ID}})$ for identity ID and a ciphertext $\mathsf{CT} = (C_1, (\mathsf{CT}_1, vk_1, \sigma_1), (\mathsf{CT}_2, vk_2, \sigma_2))$, it performs as follows:
  1) Run $\Sigma.\mathbf{V}(vk_1, \sigma_1)$ and $\Sigma.\mathbf{V}(vk_2, \sigma_2)$. If both outputs are 1, then proceed the next step. Otherwise, output $\perp$.
  2) Run
     a) $sk_{1,[\mathsf{ID},vk_1]} \leftarrow \mathbf{W}^{(2)}.\mathbf{KeyGen}(pp_1, [\mathsf{ID}, vk_1], sk_{1,\mathsf{ID}})$
     b) $M' \leftarrow \mathbf{W}^{(2)}.\mathbf{Dec}(sk_{1,[\mathsf{ID},vk_1]}, \mathsf{CT}_1)$.

3) Run
   a) $sk_{2,[\mathsf{ID},vk_2]} \leftarrow \mathbf{W^{(2)}.KeyGen}(pp_2, [\mathsf{ID}, vk_2], sk_{2,\mathsf{ID}})$.
   b) $M\|r_1 \leftarrow \mathbf{W^{(2)}.Dec}(sk_{2,[\mathsf{ID},vk_2]}, \mathsf{CT}_2)$.

4) Check if $C_1 = g^{r_1}$ and $M' = M^{r_1}$. If both hold, output M. Otherwise, output $\perp$.

- **Auth**$'(sk_{\mathsf{ID}})$: On input the secret key $sk_{\mathsf{ID}} = (sk_{1,\mathsf{ID}}, sk_{2,\mathsf{ID}})$ for identity ID, it returns $td_{\mathsf{ID}} = sk_{1,\mathsf{ID}}$.
- **Test**$'(\mathsf{CT}_{\mathsf{ID}_A}, td_{\mathsf{ID}_A}, \mathsf{CT}_{\mathsf{ID}_B}, td_{\mathsf{ID}_B})$: On input

$$\begin{aligned}\mathsf{CT}_{\mathsf{ID}_A} =&(C_{A,1}, (\mathsf{CT}_{A,1}, vk_{A,1}, \sigma_{A,1}),\\ &(\mathsf{CT}_{A,2}, vk_{A,2}, \sigma_{A,2})),\\ \mathsf{CT}_{\mathsf{ID}_B} =&(C_{B,1}, (\mathsf{CT}_{B,1}, vk_{B,1}, \sigma_{B,1}),\\ &(\mathsf{CT}_{B,2}, vk_{B,2}, \sigma_{B,2})),\end{aligned}$$

$td_{\mathsf{ID}_A} = sk_{1,\mathsf{ID}_A}$, and $td_{\mathsf{ID}_B} = sk_{1,\mathsf{ID}_B}$, it runs as follows:

1) Run $\Sigma.\mathbf{V}(vk_{A,1}, \sigma_{A,1})$ and $\Sigma.\mathbf{V}(vk_{B,1}, \sigma_{B,1})$. If both are 1, then proceed the next step. Otherwise, output 0.

2) Run
   a) $sk_{1,[\mathsf{ID}_A,vk_{A,1}]} \leftarrow \mathbf{W^{(2)}.KeyGen}(pp_1, [\mathsf{ID}_A, vk_{A,1}], td_{\mathsf{ID}_A})$
   b) $M'_{\mathsf{ID}_A} \leftarrow \mathbf{W^{(2)}.Dec}(sk_1, [\mathsf{ID}_A, vk_{A,1}], \mathsf{CT}_{A,1})$.

3) Run
   a) $sk_{1,[\mathsf{ID}_B,vk_{B,1}]} \leftarrow \mathbf{W^{(2)}.KeyGen}(pp_1, [\mathsf{ID}_B, vk_{B,1}], td_{\mathsf{ID}_B})$
   b) $M'_{\mathsf{ID}_B} \leftarrow \mathbf{W^{(2)}.Dec}(sk_{1,[\mathsf{ID}_B,vk_{B,1}]}, \mathsf{CT}_{B,1})$.

4) Check if

$$e(C_{A,1}, M'_{\mathsf{ID}_B}) = e(C_{B,1}, M'_{\mathsf{ID}_A}) \tag{1}$$

where e is a bilinear map in pp_1. If it holds, return 1. Otherwise, return 0.

Remark 2: While it is assumed that an identity at each level in Waters' HIBE belongs to $\mathbb{Z}_p$, a verification key of BSW signature scheme consists of 7 elements in $\mathbb{G}$. For compatibility, we assume that a verification key is mapped to an element in $\mathbb{Z}_p$ by using a function from $\mathbb{G}^7$ to $\mathbb{Z}_p$, once it is used as an identity.

As aforementioned, the above modification is not more efficient than the instantiations obtained by generic constructions. So, we omit the formal analysis about correctness and security of the modification, but they are quite straightforward. In fact, the modification is designed by following the well-known design strategy in the area of encryption with equality test: (1) Generate two secure ciphertexts of message and its hash value, and (2) give a link between them. By following this strategy, a ciphertext of the modification consists of three parts:

- $C_1 = g^{r_1}$: a component to give a link between the next two parts,
- $(\mathsf{CT}_1, vk_1, \sigma_1)$: a ciphertext of message M^{r_1} of 2-level IND-ID-CCA secure Waters' HIBE under identity $[\mathsf{ID}, vk_1]$, and
- $(\mathsf{CT}_2, vk_2, \sigma_2)$: a ciphertext of message $M\|r_1$ of 2-level IND-ID-CCA secure Waters' HIBE under identity $[\mathsf{ID}, vk_2]$.

That is, the last two parts of ciphertexts are generated by IND-ID-CCA secure HIBE schemes. Thus, contrary to Zhu et al.'s IBEwET, it is difficult to generate a new valid ciphertext by modifying either $(\mathsf{CT}_1, vk_1, \sigma_1)$ or $(\mathsf{CT}_2, vk_2, \sigma_2)$ itself. In addition, the attacker may try to replace $(\mathsf{CT}_1, vk_1, \sigma_1)$ or $(\mathsf{CT}_2, vk_2, \sigma_2)$ with other valid ciphertexts by generating itself, but it should know r_1 in C_1 for this case to generate a valid ciphetext. Therefore, informally, the above modification seems to be CCA secure.

For correctness, if the underlying IND-ID-CCA secure Waters' IBE scheme is correct, then the decryption algorithm can recover M^{r_1} and $M\|r_1$ from $(\mathsf{CT}_1, vk_1, \sigma_1)$ and $(\mathsf{CT}_2, vk_2, \sigma_2)$, respectively. So, the first condition of the correctness holds, as Zhu et al.'s construction. Furthermore, the test algorithm can recover M^{r_1} parts of input ciphertexts since the secret key for $(\mathsf{CT}_1, vk_1, \sigma_1)$ can be generated by the trapdoor which is the secret key of ID at the first level of HIBE by using the secret key delegation property of HIBE. Then, by checking the relation (1), we can confirm the equality of two messages in ciphertexts, and so the second and last conditions of the correctness also hold.

On the other hand, the ciphertext of IBEwET instantiations obtained by generic constructions [7], [8] consists of two ciphertexts of a 3-level IND-ID-CPA secure HIBE scheme[1] and one signature of SUF one-time signature scheme. For fair comparison, let us assume that we exploit the same HIBE and signature schemes as our modification, but the HIBE scheme in the generic construction should support one more level. That is, we employ the 3-level Waters' HIBE scheme. Then, a ciphertext of those instantiations consists of

$$\begin{aligned}\mathsf{CT}_1 &= \mathbf{W^{(3)}.Enc}(pp_1, [\mathsf{ID}, 0, vk], M)\\ \mathsf{CT}_2 &= \mathbf{W^{(3)}.Enc}(pp_1, [\mathsf{ID}, 1, vk], H(M))\\ \sigma &= \Sigma.\mathbf{S}(sk, \mathsf{CT}_1\|\mathsf{CT}_2), \text{ and } vk,\end{aligned}$$

where H is an appropriate hash function and $[\mathsf{ID}_1, \mathsf{ID}_2, \mathsf{ID}_3]$ denotes a 3-level identity with ID_1 for the first level, ID_2 for the second level, and ID_3 for the last level. Refer to [7] and [8] for the details about generic constructions.

To compare the efficiency, we first look into the efficiency of Waters' HIBE and BSW signature schemes. First, in the IND-ID-CPA secure ℓ-level Waters' HIBE scheme, the parameter sizes and computational costs are as follows: Let $|\mathbb{G}|$, $|\mathbb{G}_T|$, and $|\mathbb{Z}_p|$ denote bit sizes required to represent elements in $\mathbb{G}$, $\mathbb{G}_T$, and $\mathbb{Z}_p$, respectively. BM, $\mathsf{E}_{\mathbb{G}}$, and $\mathsf{E}_{\mathbb{G}_T}$ denote costs for bilinear map computation, exponentiation in $\mathbb{G}$, and exponentiation in $\mathbb{G}_T$, respectively.

- the public parameter size: $(2\lambda + 4)|\mathbb{G}|$

[1] One may consider that our modification also uses IND-ID-CPA secure HIBE schemes and then apply the CHK transformation simultaneously as in generic constructions. However, it just comes to resemble the instantiations of generic constructions and is still not efficient due to the redundant part of checking relations about C_1. So, we avoid to apply this approach.

TABLE 1. Efficiency comparison of our modification and instantiation from generic constructions.

		Our Modification	Instantiation by [6]+ [21]
Comm	$\lvert\mathsf{CT}\rvert$	$25\lvert\mathbb{G}\rvert + 2\lvert\mathbb{G}_T\rvert + 2\lvert\mathbb{Z}_p\rvert$	$17\lvert\mathbb{G}\rvert + 2\lvert\mathbb{G}_T\rvert + \lvert\mathbb{Z}_p\rvert$
	$\lvert sk_{\mathsf{ID}}\rvert$	$4\lvert\mathbb{G}\rvert$	$2\lvert\mathbb{G}\rvert$
	$\lvert td_{\mathsf{ID}}\rvert$	$2\lvert\mathbb{G}\rvert$	$3\lvert\mathbb{G}\rvert$
Comp	Key Generation	$4\mathsf{E}_{\mathbb{G}}$	$2\mathsf{E}_{\mathbb{G}}$
	Encryption	$2\mathsf{BM} + 19\mathsf{E}_{\mathbb{G}} + 2\mathsf{E}_{\mathbb{G}_T}$	$2\mathsf{BM} + 14\mathsf{E}_{\mathbb{G}} + 2\mathsf{E}_{\mathbb{G}_T}$
	Decryption	$12\mathsf{BM} + 12\mathsf{E}_{\mathbb{G}}$	$11\mathsf{BM} + 12\mathsf{E}_{\mathbb{G}}$
	Authorization	–	$2\mathsf{E}_{\mathbb{G}}$
	Test	$14\mathsf{BM} + 12\mathsf{E}_{\mathbb{G}}$	$8\mathsf{BM} + 4\mathsf{E}_{\mathbb{G}}$

$\lvert\mathbb{G}\rvert, \lvert\mathbb{G}_T\rvert, \lvert\mathbb{Z}_p\rvert$: the bit sizes to represent elements in $\mathbb{G}, \mathbb{G}_T, \mathbb{Z}_p$, respectively
$\mathsf{BM}, \mathsf{E}_{\mathbb{G}}, \mathsf{E}_{\mathbb{G}_T}$: the costs for bilinear map computation, exponentiation in $\mathbb{G}$ and exponentiation in $\mathbb{G}_T$, respectively

- the master secret key size: $1\lvert\mathbb{G}\rvert$
- the user's secret key size for level-j: $(j+1)\lvert\mathbb{G}\rvert$
- the ciphertext size for level-j: $(j+1)\lvert\mathbb{G}\rvert + 1\lvert\mathbb{G}_T\rvert$
- the setup cost: $2\mathsf{E}_{\mathbb{G}}$
- the key generation cost from level $j-1$ to level j: $2\mathsf{E}_{\mathbb{G}}$
- the encryption cost for level j: $1\mathsf{BM} + (j+1)\mathsf{E}_{\mathbb{G}} + 1\mathsf{E}_{\mathbb{G}_T}$
- the decryption cost for level j: $(j+1)\mathsf{BM}$

Similarly, in the BSW signature scheme, the parameter sizes and computational costs are as follows:

- the verification key size: $7\lvert\mathbb{G}\rvert$
- the secret key size: $1\lvert\mathbb{G}\rvert$
- the signature size: $2\lvert\mathbb{G}\rvert + 1\lvert\mathbb{Z}_p\rvert$
- the key generation cost: $2\mathsf{E}_{\mathbb{G}}$
- the signing cost: $4\mathsf{E}_{\mathbb{G}}$
- the verification cost: $3\mathsf{BM} + 4\mathsf{E}_{\mathbb{G}}$

By applying the above, we provide an efficiency comparison of our modification and the outcome obtained by generic constructions in Table 1. It shows that our modification is almost worse than instantiations of generic constructions in terms of all, except for trapdoor size and authorization cost.

Finally, we again remark that the main purpose for our modification is to explore how hard to design IBEwET schemes over bilinear groups that (significantly) outperform outcomes obtained by generic constructions in the standard model, not to improve existing IBEwET schemes. To the best of our knowledge, there is no known way to design efficient IBE schemes over bilinear groups in the standard model without employing the CHK transformation or its variants. On the other hand, all generic constructions for IBEwET in the standard model already exploit the CHK transformation or its variants. So, it seems hard to design IBEwET schemes over bilinear groups that significantly outperform instantiations obtained by generic constructions in the standard model.

B. INCOMPATIBILITY OF OPERATIONS IN THE ENCRYPTION ALGORITHM

In the previous subsection, we discuss how to fix Zhu et al.'s scheme so that it achieves the CCA security. Though we may fix it to be CCA secure, it still remains another issue in the encryption algorithm.

Let me recall the encryption algorithm of Zhu et al.'s original IBEwET. It takes the public parameter pp, an identity ID, and a message M as inputs, and returns a ciphertext $\mathsf{CT} = (C_1, C_2, C_3)$ such that

$$\begin{aligned} C_1 &= g^{r_1}, \\ C_2 &= \Big(M^{r_1} e(g_1, g_2)^{r_2}, g^{r_2}, \big(u' \prod_{i \in v} u_i\big)^{r_2}\Big), \text{ and} \\ C_3 &= \Big((M \| r_1) e(g_1, g_2)^{r_3}, g^{r_3}, \big(u' \prod_{i \in v} u_i\big)^{r_3}\Big) \end{aligned}$$

where r_1, r_2, r_3 are random elements in $\mathbb{Z}_p$ selected by the encryption algorithm, $H(\mathsf{ID}) = (v_1, \cdots, v_n) \in \{0, 1\}^n$ and $v = \{i \in \{1, 2, \cdots, n\} \mid v_i = 1\}$. Here, let us focus on the third component C_3. In order to obtain C_3, the encryption algorithm computes

$$(M \| r_1) e(g_1, g_2)^{r_3}$$

where M and $e(g_1, g_2)^{r_3}$ belong to $\mathbb{G}_T$, but r_1 belongs to $\mathbb{Z}_p$. Thus, it is not well-defined to multiply $M \| r_1 \in \mathbb{G}_T \times \mathbb{Z}_p$ with $e(g_1, g_2)^{r_3} \in \mathbb{G}_T$.

To resolve this issue, we need to find an embedding ι from $\mathbb{G}_T \times \mathbb{Z}_p$ into $\mathbb{G}_T$. Simultaneously, such an embedding should satisfy a condition that the pre-image $M \| r_1$ can be easily recovered from the image $\iota(M \| r_1)$ to check if

$$C_1 = g^{r_1} \quad \text{and} \quad C_{2,1} \frac{e(d_2, C_{2,3})}{e(d_1, C_{2,2})} = M^{r_1}$$

in the decryption algorithm. However, it seems hard to develop such an embedding directly and we need other ways to circumvent this issue, e.g., restricting the message space from $\mathbb{G}_T$ to a subset of $\mathbb{G}_T$. We leave it as an open problem to construct concrete embedding functions.

VI. CONCLUSION

In this paper, we have provided adaptive chosen ciphertext attacks against Zhu et al.'s IBEwET. As a result, contrary to their security claim, we demonstrate that their scheme fails to achieve the IND-ID-CCA security against adversaries without trapdoors for equality tests and the OW-ID-CCA security against adversaries with trapdoors. Furthermore, we have attempted to address their scheme through simple modifications, but the corresponding result is no longer superior to other existing IBEwET schemes. So, to the best of our knowledge, there is no known IBEwET scheme over bilinear groups in the standard model that outperforms instantiations obtained by generic constructions for IBEwET. Considering the wide range of applications for IBEwET, it would be worthwhile to enhance the efficiency of IBEwET over bilinear groups in the standard model.

ACKNOWLEDGMENT

The author would like to thank the anonymous reviewers for their helpful comments.

REFERENCES

[1] G. Yang, C. Tan, Q. Huang, and D. S. Wong, "Probabilistic public key encryption with equality test," in *Topics in Cryptology—CT-RSA* (Lecture Notes in Computer Science), vol. 5985. San Francisco, CA, USA: Springer, Mar. 2010, pp. 119–131.

[2] S. Ma, Q. Huang, M. Zhang, and B. Yang, "Efficient public key encryption with equality test supporting flexible authorization," *IEEE Trans. Inf. Forensics Security*, vol. 10, no. 3, pp. 458–470, Mar. 2015.

[3] Y. Hou, Y. Cao, H. Xiong, Y. Song, and L. Xu, "An efficient online/offline heterogeneous signcryption scheme with equality test for IoVs," *IEEE Trans. Veh. Technol.*, early access, Apr. 5, 2023, doi: 10.1109/TVT.2023.3264672.

[4] M. Ramadan and S. Raza, "Secure equality test technique using identity based signcryption for telemedicine systems," *IEEE Internet Things J.*, early access, May 11, 2023, doi: 10.1109/JIOT.2023.3269222.

[5] H. Zhu, H. Ahmad, Q. Xue, T. Li, Z. Liu, and A. Liu, "New constructions of equality test scheme without random oracles," *IEEE Access*, vol. 11, pp. 49519–49529, 2023.

[6] B. Waters, "Efficient identity-based encryption without random oracles," in *Advances in Cryptology—EUROCRYPT* (Lecture Notes in Computer Science), vol. 3494, R. Cramer, Ed. Heidelberg, Germany: Springer, 2005, pp. 114–127.

[7] H. T. Lee, S. Ling, J. H. Seo, H. Wang, and T.-Y. Youn, "Public key encryption with equality test in the standard model," *Inf. Sci.*, vol. 516, pp. 89–108, Apr. 2020.

[8] K. Asano, K. Emura, and A. Takayasu, "More efficient adaptively secure lattice-based IBE with equality test in the standard model," in *Proc. 25th Int. Conf. Inf. Secur. (ISC)* (Lecture Notes in Computer Science), vol. 13640, W. Susilo, X. Chen, F. Guo, Y. Zhang, and R. Intan, Eds. Cham, Switzerland: Springer, 2022, pp. 75–83.

[9] Y. Ming and E. Wang, "Identity-based encryption with filtered equality test for smart city applications," *Sensors*, vol. 19, no. 14, p. 3046, Jul. 2019.

[10] M. Ramadan, Y. Liao, F. Li, S. Zhou, and H. Abdalla, "IBEET-RSA: Identity-based encryption with equality test over RSA for wireless body area networks," *Mobile Netw. Appl.*, vol. 25, no. 1, pp. 223–233, Feb. 2020.

[11] A. Hassan, R. Elhabob, N. Eltayieb, and Y. Wang, "An authorized equality test on identity-based cryptosystem for mobile social networking applications," *Trans. Emerg. Telecommun. Technol.*, vol. 32, no. 12, p. e4361, Dec. 2021.

[12] S. Ma, "Identity-based encryption with outsourced equality test in cloud computing," *Inf. Sci.*, vol. 328, pp. 389–402, Jan. 2016.

[13] R. Canetti, S. Halevi, and J. Katz, "Chosen-ciphertext security from identity-based encryption," in *Advances in Cryptology—EUROCRYPT* (Lecture Notes in Computer Science), vol. 3027, C. Cachin and J. Camenisch, Eds. Interlaken, Switzerland: Springer, 2004, pp. 207–222.

[14] D. H. Duong, H. Q. Le, P. S. Roy, and W. Susilo, "Lattice-based IBE with equality test in standard model," in *Proc. 13th Int. Conf. Provable Secur. (ProvSec)*, (Lecture Notes in Computer Science), vol. 11821, R. Steinfeld and T. H. Yuen, Eds. Cham, Switzerland: Springer, 2019, pp. 19–40.

[15] W. Susilo, D. H. Duong, and H. Q. Le, "Efficient post-quantum identity-based encryption with equality test," in *Proc. IEEE 26th Int. Conf. Parallel Distrib. Syst. (ICPADS)*, Dec. 2020, pp. 633–640.

[16] G. L. D. Nguyen, W. Susilo, D. H. Duong, H. Q. Le, and F. Guo, "Lattice-based IBE with equality test supporting flexible authorization in the standard model," in *Progress in Cryptology—INDOCRYPT* (Lecture Notes in Computer Science), vol. 12578, K. Bhargavan, E. Oswald, and M. Prabhakaran, Eds. Cham, Switzerland: Springer, 2020, pp. 624–643.

[17] Z. Wu, J. Weng, A. Yang, L. Yao, X. Liang, Z. Jiang, and J. Wen, "Efficient and fully secure lattice-based IBE with equality test," in *Proc. 23rd Int. Conf. Inf. Commun. Secur. (ICICS)* (Lecture Notes in Computer Science), vol. 12919, D. Gao, Q. Li, X. Guan, and X. Liao, Eds. Cham, Switzerland: Springer, 2021, pp. 301–318.

[18] Q. Qu, B. Wang, L. Wang, Y. Wang, and Y. Yan, "More efficient tightly-secure lattice-based IBE with equality test," *Comput. Standards Interfaces*, vol. 86, Aug. 2023, Art. no. 103736.

[19] D. Boneh and J. Katz, "Improved efficiency for CCA-secure cryptosystems built using identity-based encryption," in *Topics in Cryptology—CT-RSA* (Lecture Notes in Computer Science), vol. 3376, A. Menezes, Ed. Berlin, Germany: Springer, 2005, pp. 87–103.

[20] D. Boneh and X. Boyen, "Efficient selective-id secure identity-based encryption without random oracles," in *Advances in Cryptology—EUROCRYPT* (Lecture Notes in Computer Science), vol. 3027, C. Cachin and J. Camenisch, Eds. Berlin, Germany: Springer, May 2004, pp. 223–238.

[21] D. Boneh, E. Shen, and B. Waters, "Strongly unforgeable signatures based on computational Diffie–Hellman," in *Public Key Cryptography—PKC* (Lecture Notes in Computer Science), vol. 3958, M. Yung, Y. Dodis, A. Kiayias, and T. Malkin, Eds. Berlin, Germany: Springer, 2006, pp. 229–240.

HYUNG TAE LEE received the B.Sc., M.Sc., and Ph.D. degrees in mathematics from Seoul National University, Republic of Korea, in 2006, 2008, and 2013, respectively. He was a Research Fellow with Nanyang Technological University, Singapore, and an Assistant Professor with Jeonbuk National University, Republic of Korea. He is currently an Assistant Professor with the School of Computer Science and Engineering, Chung-Ang University, Republic of Korea. His research interests include computational number theory, cryptography, and information security.