Journal of Universal Computer Science, vol. 25, no. 3 (2019), 245-269
submitted: 28/3/18, accepted: 30/12/18, appeared: 28/3/19

CCA-Secure Deterministic Identity-Based Encryption Scheme

Meijuan Huang
(School of Computer Science, Shaanxi Normal University, Xi'an 710062
School of Mathematics and Information Science
Baoji University of Arts and Sciences, Baoji 721013, China
mjhuang335@foxmail.com)

Bo Yang*
(School of Computer Science, Shaanxi Normal University, Xi'an 710062
State Key Laboratory of Information Security
Institute of Information Engineering, Chinese Academy of Sciences
Beijing 100093, China
byang@snnu.edu.cn
*Corresponding author)

Yi Zhao
(Department of Computer Science, University of Surrey, Guildford, Surrey
GU2 7XH, United Kingdom
yizhaore@hotmail.com)

Kaitai Liang
(Department of Computer Science, University of Surrey, Guildford, Surrey
GU2 7XH, United Kingdom
k.liang@surrey.ac.uk)

Liang Xue
(Department of Electrical and Computer Engineering
University of Waterloo, Waterloo, ON N2L 3G1, Canada
l34xue@uwaterloo.ca)

Xiaoyi Yang
(Department of Computing, The Hong Kong Polytechnic University
Hong Kong, China
aiyooho@sina.com)

Abstract: Deterministic public-key encryption, encrypting a plaintext into a unique ciphertext without involving any randomness, was introduced by Bellare, Boldyreva, and O'Neill (CRYPTO 2007) as a realistic alternative to some inherent drawbacks in randomized public-key encryption. Bellare, Kiltz, Peikert and Waters (EUROCRYPT 2012) bring deterministic public-key encryption to the identity-based setting, and propose deterministic identity-based encryption scheme (DIBE). Although the constructions of chosen plaintext attack (CPA) secure DIBE scheme have been studied intensively, the construction of chosen ciphertext attack (CCA) secure DIBE scheme is still challenging problems. In this paper, we introduce the notion of identity-based all-but-one trapdoor functions (IB-ABO-TDF), which is an extension version of all-but-one lossy trapdoor function in the public-key setting. We give a instantiation of IB-ABO-TDF under decisional linear assumption. Based on an identity-based lossy trapdoor function and our IB-ABO-TDF, we present a generic construction of CCA-secure DIBE scheme.

Key Words: deterministic identity-based encryption, identity-based lossy trapdoor functions, identity-based all-but-one trapdoor functions, chosen ciphertext security
Category: C.2.0, D.4.6, E.3

1 Introduction

The semantic security for public key encryption requires that the encryption algorithm must be a random process. This creates a significant performance bottleneck if, for example, one wants to perform fast search over many encrypted data items. To address this issue, Bellare, Boldyreva, and O'Neill [Bellare et al. 2007] first introduced the notion of deterministic public-key encryption (DPKE), in which the encryption algorithm does not use randomness, i.e., its encryption algorithm is required to be a deterministic function of the message. The motivating application of deterministic PKE is to perform fast search over many encrypted data items. The technique is more effective in scenarios where frequent search queries are performed over a huge database of unpredictable data items. Deterministic encryption permits logarithmic time search on encrypted data, while randomized encryption only allows linear time search, meaning a search requires scanning the whole database. Moreover, since deterministic encryption does not use randomness, it is an important class of PKE dealing with the subsequently revealed problem of randomness subversion [Bellare et al. 2009]. The DPKE is used as a building block of hedged PKE [Bellare et al. 2015b, Boldyreva et al. 2017] and nonce-based PKE [Huang et al. 2018], which achieve best possible security in the face of bad randomness.

Because the encryption algorithm of DPKE is a deterministic process, of course deterministic public key encryption cannot satisfy the meaningful notion of security of randomized public key encryption. Bellare et al. [Bellare et al. 2007] provided the "strongest possible" notion of security for this primitive, called PRIV, which can be realized for relatively high-entropy plaintext distributions. Constructions of DPKE schemes satisfying the notions of security were proposed in the random oracle model by Bellare et al. [Bellare et al. 2007]. Later, Bellare et al. [Bellare et al. 2008] and Boldyreva et al. [Boldyreva et al. 2008] refined and extended the security notion and presented constructions in the standard model. Especially, Boldyreva et al. [Boldyreva et al. 2008] gave general constructions of both CPA and CCA secure deterministic public key encryption schemes which are based on lossy trapdoor functions (LTDF) [Peikert et al. 2008]. They showed that any LTDF is a deterministic PKE scheme which is PRIV-secure for high min-entropy block-sources (namely, each message to be encrypted has high min-entropy given the other messages) as long as the lossy mode acts as a universal hash function. Emerging as a practically-motivated notion of theoretical depth and interest, several significant foundational works then further investigated security for deterministic encryption and presented standard model constructions [Brakerski et al. 2011, Mironov et al. 2012, Raghunathan et al.2013, O'Neill 2010, Cui et al. 2014, Zhang et al. 2014, Wee 2012, Bellare et al. 2015a, Fuller et al. 2012, Koppula et al. 2016].

Identity-based encryption (IBE) is a public key encryption that enables one to encrypt a message using a recipient's identity, rather than its public key. It simplifies public key and certificate distribution and management and thus has a wide range of applications [Yu et al. 2017, Li et al. 2019]. Due to the inherent advantage of IBE, Bellare, Kiltz, Peikert and Waters [Bellare et al. 2012] extended the notion of deterministic encryption into the identity-based setting. They proposed a CPA-secure deterministic identity-based encryption (DIBE) scheme by first constructing identity-based lossy trapdoor function (IB-LTDF). The DIBE allows quickly logarithmic-time searchable identity-based encryption of database entries while maintaining the maximal possible privacy. Later, Escala et al. [Escala et al. 2014] provided an alternative definition of partial lossiness of IB-LTDF and constructed a hierarchical identity-based lossy trapdoor functions (HIB-LTDF), based on which they achieved DHIBE scheme for block-sources, this DHIBE scheme is secure against chosen plaintext attack (CPA). After that, several follow-up works [Xie et al. 2012, Fang et al. 2016, Zhang et al. 2017] further investigated security of DIBE and presented CPA-secure DIBE schemes from the hardness of learning with error (LWE) problem. So far, existing DIBE schemes only achieve chosen-plaintext security.

Security against adaptive chosen-ciphertext attack (CCA) is a de facto security notion for public-key encryption in practice. Active adversary might also obtain the decryption of ciphertexts under any identity of its choice. Thus it is necessary to consider the stronger security notion of DIBE, i.e., PRIV-ID-CCA security (we will explain it in section 2). Inspired by CHK transformation approach [Canetti et al. 2004] in randomized PKE, which converts IND-ID-CPA secure 2-level hierarchical IBE to IND-ID-CCA secure IBE using a strongly unforgeable one-time signature. The natural idea is to adapt this approach to the deterministic encryption. That is to say, one can construct PRIV-CCA-secure deterministic IBE based on a PRIV-CPA-secure 2-level deterministic HIBE. Unfortunately, we observe that it is not natural to do so. Since in the deterministic setting, the strongly unforgeable one-time signature is replaced by a target collision-resistant hash function of the plaintext. Similarly, following CHK transformation, the hope is that the proof tries to reduce the security of deterministic IBE against adaptive chose-ciphertext attacks to that of the 2-level selective-id secure deterministic HIBE against chose-plaintext attacks. That is, the reduction attempts to determine which message was actually encrypted in deterministic IBE with the help of an adversary who breaks the deterministic HIBE. The analysis from CHK transformation, however, does not quite work since it crucially relies on the fact that the plaintexts corresponding to the challenge ciphertexts are chosen by the adversary in randomized encryption setting. While, in the deterministic encryption setting, the plaintexts corresponding to the challenge ciphertexts are not chosen by the adversary, such that the challenger cannot perform the simulations. For the above reasons, in this paper, we attempt to construct a deterministic IBE scheme which can achieve CCA security.

1.1 Our contributions

Identity-Based All-But-One TDFs. In STOC'08 [Peikert et al. 2008], Peikert and Waters introduced a new powerful primitive called lossy trapdoor functions (LTDF) and a richer abstraction called all-but-one trapdoor functions (ABO-TDF). LTDF operates in one of two possible "mode", an injective one and an un-invertible lossy one, for which the outputs are indistinguishable. ABO-TDF is a generalization of the LTDF whose first input is drawn from a set of branches, one of which is lossy. Freeman et al. [Freeman et al. 2010] generalized the definition of ABO-TDF by allowing possibly many lossy branches (other than one).

We introduce the notion of identity-based all-but-one trapdoor functions (IB-ABO-TDF), which is an extension of all-but-one trapdoor functions (ABO-TDF) in the public key setting. As for identity-based ABO-TDF, which is essentially specific ABO-TDF whose identity set can be viewed as the set of branches, each function has many lossy branches just as the generalized definition in [Freeman et al. 2010], but each branch is now represented by a pair of (id, b). The first component id is the user's identity, and the second component b is the tag. That is to say, the IB-ABO-TDF can be viewed as a specific ABO-TDF with the set of branch IDSp $\times$ TagSp, where IDSp is identity space and TagSp is tag space. The lossy identity id and the lossy tag b determine together the lossy branch. Lossy identities are determined by a auxiliary input which is hidden in the public parameters. The security requires that it is computationally indistinguishable to tell a lossy branch from an injective branch. Meanwhile, given a lossy branch, it is hard to find one-more lossy branches without the trapdoor.

Based on the basic IBE scheme of Bellare et al. [Bellare et al. 2012], we present a concrete construction of IB-ABO-TDF, and its security is proved in the selective-id security model based on the hardness of decisional linear Diffie-Hellman assumption (DLIN assumption). In our construction, each identity function takes as input (mpk, id, b, x), where mpk is public parameter, id is identity, b is branch and x is the input value, and outputs a ciphertext, which is a matrix encryption of the basic IBE scheme.

CCA-secure deterministic IBE. Based on an IB-LTDF and our IB-ABO-TDF, we present a generic construction of CCA-secure deterministic identity-based encryption scheme in the standard model. Its security is proved under the selective-id security model. In this paper, we stick to the original setting of Bellare et al. [Bellare et al. 2007] and require that the plaintext distributions can not depend on the master public key of the system. In the case of block-sources [Boldyreva et al. 2008], Boldyreva et al. proved that PRIV1-security (i.e., for single-message challenge security) is equivalent to PRIV-security (i.e., for multi-message challenge security) in the sense of indistinguishability-based definitions. In this work, we follow the simplified indistinguishability-based notion, called PRIV1-IND as introduced in [Boldyreva et al. 2008].

Our construction of DIBE scheme builds on the framework of Boldyreva et al. [Boldyreva et al. 2008] for constructing CCA-secure deterministic PKE scheme in the standard model. In our construction, the deterministic encryption algorithm $E(id, b, m)$, where id is identity, b is branch and m is the message, requires to find a deterministic method to sample the branch. Following [Boldyreva et al. 2008], let $b = H_{tcr}(m)$, where H_{tcr} is a universal and target collision-resistant hash function. If message m has sufficient entropy, the branch looks random due to the Leftover Hash Lemma [Dodis et al. 2004]. The differences are, in the identity-based setting, the simulator must answer the decryption queries for any identity (including the challenge identity) from adversary. Because the branch is a pair (id, b), let (id^*, b^*) be the lossy branch, only if the branch $(id, b) \neq (id^*, b^*)$, with overwhelming probability, the identity-based ABO-TDF works as an injective trapdoor function. Therefore, the simulator can answer the adversary's decryption queries corresponding to the challenge identity as long as $b \neq b^*$.

1.2 Related work

Bellare et al. introduced the notion of deterministic identity-based encryption in [Bellare et al. 2012]. They constructed an identity-based lossy trapdoor functions (IB-LTDF) from pairing and built a DIBE scheme with selective-id security as an application of IB-LTDF. Soon afterwards, Escala et al. [Escala et al. 2014] introduced the notion of hierarchical identity-based lossy trapdoor functions (HIB-LTDF), based on which they constructed deterministic hierarchical identity-based encryption scheme (DHIBE) from pairings. In [Xie et al. 2012], Xie et al. considered deterministic identity-based public key encryption in the auxiliary-input setting and proposed a DIBE scheme from lattices that is adaptively secure. Fang et al. [Fang et al. 2016] constructed a selective-id secure DHIBE scheme based on the hardness of learning with rounding over small modulus [Bogdanov et al. 2016]. In fact, a selective-id secure DHIBE implies a DIBE with a selective security. Recently, Zhang et al. [Zhang et al. 2017] constructed an adaptive secure DIBE scheme with shorter public parameters from partitioning function [Yamada et al. 2017] under the learning with error assumption.

Below we compare our construction with the related works [Xie et al. 2012, Fang et al. 2016, Zhang et al. 2017] in terms of property and security in Table 1. The second column shows whether the scheme is constructed in a generic way. The third to fifth column show the security, the adversary type and the underlying assumption for guaranteeing the security. It can be learnt from Table 1 that only our scheme achieves chosen ciphertext security.

Scheme	Generic	security level	Adversary type	Assumption
[Bellare et al. 2012]	$\surd$	CPA	selective-id	DLIN
[Xie et al. 2012]	$\times$	CPA	adaptive-id	LWE
[Escala et al. 2014]	$\surd$	CPA	adaptive-id	DBDH+DDH
[Fang et al. 2016]	$\times$	CPA	selective-id	LWR
[Zhang et al. 2017]	$\times$	CPA	adaptive-id	LWE
Our scheme	$\surd$	CCA	selective-id	DLIN

Table 1: Comparison of performance

1.3 Organization

The rest of the paper is organized as follows. In section 2, we review some standard notions and cryptographic definitions. We introduce the notion of IB-ABO-TDF and present a concrete construction of IB-ABO-TDF in section 3. In section 4, we propose a generic construction of CCA-secure DIBE scheme based on our IB-ABO-TDF and an IB-LTDF. Finally, we state our conclusion in section 5.

2 Preliminaries

2.1 Notation

We use uppercase Roman letters A, B,... to represent sets, lowercase Roman letters to elements of a set $x \in X$, and bold to vectors $\mathbf{x} \in X^n$. Bold uppercase letters $\mathbf{A} = [a_{ij}]$ represents matrices of scalars. If $\mathbf{x}$ is a vector, then $|\mathbf{x}|$ denotes the number of its coordinates and $\mathbf{x}[i]$ denotes its i-th coordinate. If X is a set, X^n denotes the set of n dimensional vector over X. $\mathbf{X}^{a\times b}$ denotes the set of a by b matrices with entries in X. The (i,j)-th entry of a 2 dimensional matrix $\mathbf{X}$ is denoted by $\mathbf{X}[i,j]$. If S is a set, then $s \xleftarrow{\$} S$ denotes the operation of sampling an elements uniformly at random from S. Notation $\langle \mathbf{a}, \mathbf{b}\rangle$ represents standard scalar product of vector $\mathbf{a}$ and $\mathbf{b}$ with equal-length.

The security parameter is denoted by λ throughout the paper. We let negl(λ) denote some unspecified function such that it approaches zero faster than reciprocal of every polynomial $f(\lambda)$, saying that such a function is negligible.

2.2 Hashing

A family of hash functions $\mathcal{H} = \{H_i : \{0,1\}^n \to R\}$ is *universal* if for all $x_1 \neq x_2 \in \{0,1\}^n$, $\Pr[H(x_1) = H(x_2) : H \xleftarrow{\$} \mathcal{H}] \leq \frac{1}{|R|}$.

A hash function $\mathcal{H} = (K, H)$ is a *target collision-resistant* (tcr) if for every polynomial time adversary $\mathcal{A}$, the tcr-advantage

$$Adv_{\mathcal{H}}^{tcr}(\mathcal{A}) = \Pr[H(k,x_1) = H(k,x_2) : (x_1, st) \xleftarrow{\$} \mathcal{A}; k \leftarrow K; x_2 \xleftarrow{\$} \mathcal{A}(k, st)]$$

of $\mathcal{A}$ against $\mathcal{H}$ is negligible [Boldyreva et al. 2008].

2.3 Randomness extractor

Here we review a few concepts related probability distributions and extracting uniform bits from weak random sources.

The statistical distance between two probability distributions X and Y over the same domain D is $\Delta(X,Y) = \frac{1}{2}\Sigma_{a\in D}\ |\Pr[X = a] - \Pr[Y = a]|$. The min-entropy of a random variable X is $H_\infty(X) = -\log(\max_x \Pr[X = x])$. A distribution X over $\{0,1\}^l$ is called a (t,l)-source if $H_\infty(X) \geq t$. A distribution X is ϵ-*close* to a t-source if there exists a t-source Y such that $\Delta(X,Y) \leq \epsilon$. Average min-entropy, which captures the remaining unpredictability of X conditioned on the value of Y, is $\tilde{H}_\infty(X|Y) = -\log(E_{y\leftarrow Y}[2^{-H_\infty(X|Y=y)}])$.

Lemma 2.1 (Generalized leftover hash lemma(LHL) [Dodis et al. 2004]). Let H be a family of universal hash functions with range R. Let X and Y be random variables such that $X \in \{0,1\}^n$ and $\tilde{H}_\infty(X|Y) \geq \log|R| + 2\log(\frac{1}{\epsilon})$. Then for $h \xleftarrow{\$} H$, we have $\Delta((Y, h, h(X)), (Y, h, U)) \leq \epsilon$, where U is the uniform distribution over the range R.

Lemma 2.2 (Chain Rule [Dodis et al. 2004]). If Y has 2^r values and Z is any random variable, then $\tilde{H}_\infty(X|(Y,Z)) \geq \tilde{H}_\infty(X|Z) - r$.

2.4 Identity-based lossy trapdoor functions

The notion of IB-LTDF, introduced by Bellare et al. [Bellare et al. 2012], is an extension of LTDF in the identity-based setting. In an identity-based lossy trapdoor functions collection, which mode (injective or lossy) the function operates depends on the identity. To properly define lossiness in the identity-based setting, Bellare et al. added an auxiliary input from auxiliary input space AuxSp when generating the parameters. Depending on the value of this auxiliary input, it can obtain the injective trapdoor function or lossy function. $aux \in$ AuxSp denotes a particular auxiliary input independent of any identity, which results in an injective setup. That is, under the injective setup, the evaluation function is injective for any identity. Aux$(\cdot)$ (called auxiliary input generator in [Bellare et al. 2012]) denotes an algorithm that takes as input an identity from identity space IDSp and returns an auxiliary input in AuxSp. $aux(id)$ denotes an auxiliary input produced by an auxiliary input generator Aux(id) taking as input special identity id, which results in a lossy setup. That is, under the lossy setup, the identity id lead to lossy evaluation functions, used in the security proof. The requirement is that it is hard to distinguish lossy identities from injective ones. Next, we review the notion of identity-based lossy trapdoor functions proposed by Bellare et al. [Bellare et al. 2012] in the selective-id case.

Definition 1 (IB-LTDF). A collection of identity-based (n, k)-lossy trapdoor functions with the identity space IDSp, input space InSp, and auxiliary input space AuxSp is a tuple of (possibly probabilistic) polynomial time algorithms (Setup, KG, Eval, Inv) with the following specifications:

Setup$(1^\lambda, aux)$. For fixed particular auxiliary input $aux \in$ AuxSp, the algorithm outputs (mpk, msk), where mpk is public parameters and msk is its master secret key.

KG(mpk, msk, id). Given mpk, msk, identity $id \in$ IDSp the probabilistic algorithm outputs a private key sk_{id} with respect to the given id.

Eval(mpk, id, x). The deterministic algorithm takes as input mpk, identity $id \in$ IDSp, and $x \in \{0,1\}^n$, outputs a value y.

Inv(mpk, sk_{id}, y). The deterministic algorithm takes as input mpk , private key sk_{id}, and a value y, outputs either $x \in \{0,1\}^n$ or $\perp$.

We require that the following correctness and lossiness requirements hold:

- Injective correctness and invertibility. For any $id \in$ IDSp, the evaluation algorithm computes a deterministic injective function over the input space $\{0,1\}^n$, which can be inverted using the private key sk_{id} corresponding to the given id. Formally, for $aux \in$ AuxSp, $(mpk, msk) \leftarrow$ Setup$(1^\lambda, aux)$, $sk_{id} \leftarrow$ KG(mpk, msk, id), and $x \in \{0,1\}^n$,

$$\Pr[\mathrm{Inv}(mpk, sk_{id}, \mathrm{Eval}(mpk, id, x)) \neq x] \leq negl(\lambda)$$

- Lossiness. Sample an identity $id \xleftarrow{\$}$ IDSp, for auxiliary input $aux(id) \leftarrow$ Aux(id), $(mpk, msk) \leftarrow$ Setup$(1^\lambda, aux(id))$, the image of the algorithm Eval$(mpk, id, \cdot)$ has size at most 2^{n-k}. That is $|\text{Eval}(mpk, id, \cdot)| \leq 2^{n-k}$.
- Indistinguishability of lossy identities and injective identities. For the auxiliary inputs aux, $aux(id) \in$ AuxSp, the first output mpk_0 of Setup$(1^\lambda, aux)$ and the first output mpk_1 of Setup$(1^\lambda, aux(id))$ are computationally indistinguishable. That is, for every probabilistic polynomial time adversary $\mathcal{A}$

$$Adv^{\text{IND}}_{\text{IB-LTDF},\mathcal{A}}(\lambda) =$$
$$|\Pr[\mathcal{A}(mpk_0, 1^\lambda)^{\text{KG}(\cdot)} = 1] - \Pr[\mathcal{A}(mpk_1, 1^\lambda)^{\text{KG}(\cdot)} = 1]|$$

is negligible, where KG$(\cdot)$ denotes that $\mathcal{A}$ can make private key query on identity id by calling KG algorithm. For $d \in \{0,1\}$, $\mathcal{A}(mpk_d, 1^\lambda)$ is defined as follows:

(1) The adversary $\mathcal{A}$ outputs an identity id^* as the target identity.

(2) The challenger $\mathcal{C}$ samples $aux \in$ AuxSp and $aux(id^*) \leftarrow$ Aux(id^*), and computes $(mpk_0, msk_0) \leftarrow$ Setup$(1^\lambda, aux)$, $(mpk_1, msk_1) \leftarrow$ Setup$(1^\lambda, aux(id^*))$. The challenger then sends mpk_d to $\mathcal{A}$.

(3) The adversary $\mathcal{A}$ makes private secret query for identity id with restriction that $id \neq id^*$. $\mathcal{C}$ returns sk_{id} to $\mathcal{A}$ by calling the algorithm KG.

(4) $\mathcal{A}$ outputs a guess $d' \in \{0,1\}$.

2.5 Deterministic identity-based encryption

An deterministic identity-based encryption (DIBE) scheme DE_{IB} is a tuple of polynomial time algorithm (DIB.Setup, DIB.Der, DIB.Enc, DIB.Dec). The probability algorithm DIB.Setup takes as input a security parameter 1^λ, and outputs a master key pair (mpk,msk), where mpk is master public key and msk is master secret key. The key derivation algorithm DIB.Der takes as input an identity id and master secret key. It returns the private key sk_{id} associated with the identity id. The deterministic encryption algorithm DIB.Enc takes as input the master public key mpk, and identity id and a message m. It outputs a ciphertext C. The decryption algorithm DIB.D takes as input identity id, its associated private key sk_{id}, and a ciphertext C. It returns a message m or the symbol $\perp$.

PRIV1-IND-ID-CCA security. An DIBE scheme DE_{IB} is PRIV1 selective-id secure against chosen-ciphertext attack for (t,n)-source M_0 and M_1 and all polynomial time adversary $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$, the advantage

$$Adv^{\text{PRIV1-IND-ID-CCA}}_{\text{DE}_{\text{IB}},\mathcal{A}}(\lambda) =$$
$$|\Pr[Guess^{\text{CCA}}_{\text{DE}_{\text{IB}},\mathcal{A}}(M_0) = 1] - \Pr[Guess^{\text{CCA}}_{\text{DE}_{\text{IB}},\mathcal{A}}(M_1) = 1]|$$

of $\mathcal{A}$ against DE_{IB} is negligible. Where $Guess^{\text{CCA}}_{\text{DE}_{\text{IB}},\mathcal{A}}(M_b)$ for $b \in \{0,1\}$ is defined as follows:

- The adversary $\mathcal{A}$ outputs an identity id^* as the target identity.
- The challenger runs $(mpk, msk) \leftarrow$ DIB.Setup(1^λ), and sends mpk to $\mathcal{A}_2$.
- $\mathcal{A}_2$ is allowed to make a number of private key queries and decryption queries for identity id:
  Private key query. The adversary $\mathcal{A}_2$ asks for the private key corresponding to any identity id as long as $id \neq id^*$. The challenger correctly generates private key sk_{id} for id and returns to $\mathcal{A}_2$.
  Decryption query. $\mathcal{A}_2$ issues decryption queries C for identity id. The challenger responds with DIB.Dec$(C, id,\ sk_{id})$ using private key sk_{id} correctly generated for id.
- The challenger samples m from distribution M_b $((M_0, M_1, state) \xleftarrow{\$} \mathcal{A}_1(1^\lambda))$, and computes $C^* =$ DIB.Enc(mpk, id^*, m), and then sends C^* to $\mathcal{A}_2$.
- $\mathcal{A}_2$ outputs its guess $b' \in \{0,1\}$.

3 Identity-based ABO-TDF and its construction

In this section, we first introduce the notion of identity-based ABO-TDF. Then based on the basic IBE scheme of Bellare et al. [Bellare et al. 2012], we propose a concrete construction of IB-ABO-TDF under decisional linear (DLIN) assumption. Let G be a finite cyclic group of prime order p specified by a randomly chosen generator g. The DLIN assumption says that $g_{d+1}^{r_1+\cdots+r_d}$ is pseudorandom given $g_1, \ldots, g_{d+1}, g_1^{r_1}, \ldots, g_d^{r_d}$ where $g_1, \ldots, g_{d+1} \xleftarrow{\$} G$; $r_1, \ldots, r_d \xleftarrow{\$} \mathbb{Z}_p$.

3.1 Identity-based ABO-TDF

Identity-based ABO-TDF can be viewed as a specific kind of ABO-TDF with two variable (id, tag) as a branch. The first component id of branch (id, tag) is user's identity, and the second component tag is the label. If and only if id is lossy identity and tag is lossy label, the branch (id, tag) is lossy branch. In our construction, similarly, we follow the method of Bellare et al. [Bellare et al. 2012] for constructing IB-LTDF. In setup phrase, algorithm takes an additional auxiliary input from an auxiliary input space AuxSp. Lossy identities are determined by an auxiliary input which is hidden in the master public key. The definition of identity-based all-but-one trapdoor functions is described as follows.

Definition 2 (IB-ABO-TDF). A collection of (n, k)-identity-based all-but-one trapdoor functions with the identity space IDSp, auxiliary input space AuxSp and label space TagSp, is a tuple of polynomial time algorithms (Setup$_{abo}$, KG$_{abo}$, Eval$_{abo}$, Inv$_{abo}$) with the following specifications:

Setup$_{abo}(1^\lambda, aux, tag^*)$. For $aux \xleftarrow{\$}$ AuxSp, $tag^* \xleftarrow{\$}$ TagSp, the algorithm outputs $(mpk, msk, \tilde{B})$, where mpk is master public key and msk is master secret key, and $\tilde{B} \subset \text{IDSp} \times \text{TagSp}$ is a set of lossy branches.

KG$_{abo}(mpk, msk, id)$. Given mpk, msk, identity $id \in \text{IDSp}$, the probabilistic algorithm outputs a private key sk_{id} with respect to the identity id.

Eval$_{abo}(mpk, id, tag, x)$. For any $tag \in \text{TagSp}$, $id \in \text{IDSp}$, the algorithm takes as input mpk, id, tag, and $x \in \{0,1\}^n$, outputs a value C.

Inv$_{abo}(mpk, sk_{id}, C)$. The deterministic algorithm takes as input mpk, private key sk_{id}, and a value C, outputs either $x \in \{0,1\}^n$ or $\perp$.

We require that the following properties hold:

– Injective correctness and invertibility. For any $(id, tag) \in \text{IDSp} \times \text{TagSp}$, if $(id, tag) \notin \tilde{B}$, where $(mpk, msk, \tilde{B}) \leftarrow \text{Setup}_{abo}(1^\lambda, aux, tag^*)$, the algorithm $\text{Eval}_{abo}(mpk,\ id,\ tag, \cdot)$ computes a deterministic injective function over the domain $\{0,1\}^n$, which can be inverted using the private key sk_{id} corresponding to the given id. Formally, $(mpk, msk, \tilde{B}) \leftarrow \text{Setup}_{abo}(1^\lambda,\ aux, tag^*)$, $sk_{id} \leftarrow \text{KG}_{abo}(mpk, msk, id)$, $(id, tag) \notin \tilde{B}$, and $x \in \{0,1\}^n$

$$\Pr[\text{Inv}_{abo}(mpk, sk_{id}, \text{Eval}_{abo}(mpk, id, tag, x)) \neq x] \leq negl(\lambda)$$

– Lossiness. For any $(id, tag) \in \text{IDSp} \times \text{TagSp}$, if $(id, tag) \in \tilde{B}$, the algorithm $\text{Eval}_{abo}(mpk,\ id,\ tag, \cdot)$ computes a deterministic function over the domain $\{0,1\}^n$ whose image has size at most 2^{n-k}.

– Indistinguishability of lossy branch. For every probabilistic polynomial time algorithm $\mathcal{A}$, the first output mpk_1 of $\text{Setup}_{abo}(1^\lambda, aux_1, tag_1)$ and the first output mpk_2 of $\text{Setup}_{abo}(1^\lambda, aux_2, tag_2)$ are computationally indistinguishable. Formally, the advantage

$$Adv^{\text{IND}}_{\text{IB-ABO-TDF},\mathcal{A}}(\lambda) =$$

$$|\Pr[\mathcal{A}(mpk_1, 1^\lambda)^{\text{KG}_{abo}(\cdot)} = 1] - \Pr[\mathcal{A}(mpk_2, 1^\lambda)^{\text{KG}_{abo}(\cdot)} = 1]|$$

of $\mathcal{A}$ is negligible, where $(mpk_d, msk, \tilde{B}) \leftarrow \text{Setup}_{abo}(1^\lambda, aux_d, tag_d)$ for $d \in \{1,2\}$. $\text{KG}_{abo}(\cdot)$ denotes that $\mathcal{A}$ can make private key query on identity id by calling KG_{abo} algorithm.

– Hard to find one-more lossy branch. Any probabilistic polynomial time algorithm $\mathcal{A}$ that receives (mpk, id, tag) as input, where $(id, tag) \in \tilde{B}$, has only a negligible probability of outputing a pair $(id', tag') \in \tilde{B} \backslash \{id, tag\}$.

3.2 The construction of identity-based ABO-TDF

Fix a bilinear map $e : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_T$, where $\mathbb{G}, \mathbb{G}_T$ are groups of prime order p. By $\mathbf{1}, \mathbf{1}_T$ we denote the identity elements of $\mathbb{G}, \mathbb{G}_T$, respectively. By $\mathbb{G}^* =$

$\mathbb{G}-\{\mathbf{1}\}$ we denote the set of generators of $\mathbb{G}$. For vectors $\mathbf{y}=(y_0,y_1)\in\mathbb{Z}_p^2$, $\mathbf{tag}=(tag_0,tag_1)$ and $id\in\mathbb{Z}_p$. We let $f_{id}(\mathbf{y})=(y_0+y_1 id)\bmod p$. For any integer n and any identity space IDSp $\subseteq\mathbb{Z}_p$, message space $\{0,1\}^n$ and auxiliary input space AuxSp $\subseteq\mathbb{Z}_p^2$, the algorithms of IB-ABO-TDF are as follows.

Setup$_{abo}(1^\lambda,\mathbf{y},\mathbf{tag}^*)$. Given auxiliary input $\mathbf{y}=(1,0)\in\mathbb{Z}_p^2$, $\mathbf{tag}^*=(tag_0^*, tag_1^*)\in\mathbb{Z}_p^2$, let $g\xleftarrow{\$}\mathbb{G}^*$, $t\xleftarrow{\$}\mathbb{Z}_p^*$, $\hat{g}=g^t$. Then let $U\xleftarrow{\$}\mathbb{G}$, $\mathbf{s}\xleftarrow{\$}(\mathbb{Z}_p^*)^n$, $\hat{\mathbf{s}}\xleftarrow{\$}\mathbb{Z}_p^n$, $\mathbf{H},\hat{\mathbf{H}},\mathbf{V}_0,\mathbf{V}_1,\hat{\mathbf{V}}_0,\hat{\mathbf{V}}_1\xleftarrow{\$}\mathbb{G}^n$. It returns master public key $mpk=(g,\hat{g},\mathbf{G},\hat{\mathbf{G}}, \mathbf{J},\mathbf{W}_0,\mathbf{W}_1,\mathbf{H},\hat{\mathbf{H}},\ \mathbf{V}_0,\mathbf{V}_1,\hat{\mathbf{V}}_0,\hat{\mathbf{V}}_1,U)$, where for $1\le i,j\le n$,

$$\mathbf{G}[i]=g^{\mathbf{s}[i]},\ \hat{\mathbf{G}}[i]=\hat{g}^{\hat{\mathbf{s}}[i]},\ \mathbf{J}[i,j]=\mathbf{H}[j]^{\mathbf{s}[i]}\hat{\mathbf{H}}[j]^{\hat{\mathbf{s}}[i]},$$

$$\mathbf{W}_0[i,j]=\mathbf{V}_0[j]^{\mathbf{s}[i]}\hat{\mathbf{V}}_0[j]^{\hat{\mathbf{s}}[i]}(U^{y_0}g^{tag_0^*})^{\mathbf{s}[i]\Delta(i,j)}$$

$$\mathbf{W}_1[i,j]=\mathbf{V}_1[j]^{\mathbf{s}[i]}\hat{\mathbf{V}}_1[j]^{\hat{\mathbf{s}}[i]}(U^{y_1}g^{tag_1^*})^{\mathbf{s}[i]\Delta(i,j)}$$

Where $\Delta(i,j)=1$ if $i=j$ and 0 otherwise. The master secret key $msk=t$ and the set of lossy branches $\tilde{B}=\{(id,\mathbf{tag})|f_{id}(\mathbf{y})=0\bmod p\wedge\mathbf{tag}=\mathbf{tag}^*\}$.

KG$_{abo}(mpk,msk,id)$. Given mpk,msk, identity $id\in\mathbb{Z}_p$, the algorithm computes decryption key $sk_{id}=(\mathbf{D}_1,\mathbf{D}_2,\mathbf{D}_3,\mathbf{D}_4)$, where $\mathbf{r}\xleftarrow{\$}(\mathbb{Z}_p^*)^n$, $\hat{\mathbf{r}}\xleftarrow{\$}\mathbb{Z}_p^n$, and for all $1\le i\le n$

$$\mathbf{D}_1[i]=(\mathbf{V}_0[i]\cdot\mathbf{V}_1[i]^{id})^{t\mathbf{r}[i]}\cdot\mathbf{H}[i]^{t\hat{\mathbf{r}}[i]},\ \mathbf{D}_2[i]=(\hat{\mathbf{V}}_0[i]\cdot\hat{\mathbf{V}}_1[i]^{id})^{\mathbf{r}[i]}\cdot\hat{\mathbf{H}}[i]^{\hat{\mathbf{r}}[i]}$$

$$\mathbf{D}_3[i]=g^{-t\mathbf{r}[i]},\ \ \mathbf{D}_4[i]=g^{-t\hat{\mathbf{r}}[i]}$$

Eval$_{abo}(mpk,id,tag,x)$. Given mpk, identity $id\in\mathbb{Z}_p$, $\mathbf{tag}=(tag_0,tag_1)\in\mathbb{Z}_p^2$, input $x\in\{0,1\}^n$, the algorithm computes the value $\mathbf{C}=(C_1,C_2,\mathbf{C}_3,\mathbf{C}_4)$ where for $1\le j\le n$, let $\mathbf{S}[j]=\mathbf{G}[j]^{-x[j]f_{id}(\mathbf{tag})}$

$$C_1=\prod_{i=1}^n\mathbf{G}[i]^{x[i]},\quad C_2=\prod_{i=1}^n\hat{\mathbf{G}}[i]^{x[i]}$$

$$\mathbf{C}_3[j]=\mathbf{S}[j]\prod_{i=1}^n(\mathbf{W}_0[i,j]\mathbf{W}_1[i,j]^{id})^{x[i]},\ \mathbf{C}_4[j]=\prod_{i=1}^n\mathbf{J}[i,j]^{x[i]}$$

Inv$_{abo}(mpk,sk_{id},\mathbf{C})$. Given mpk, ciphertext $\mathbf{C}=(C_1,C_2,\mathbf{C}_3,\mathbf{C}_4)$ for identity id, the algorithm returns $x\in\{0,1\}^n$ where for $1\le j\le n$, it sets $x[j]=0$ if

$$e(C_1,\mathbf{D}_1[j])e(C_2,\mathbf{D}_2[j])e(\mathbf{C}_3[j],\mathbf{D}_3[j])e(\mathbf{C}_4[j],\mathbf{D}_4[j])=\mathbf{1}_T$$

and 1 otherwise.

Correctness and invertibility. For $1 \leq j \leq n$, let $\mathbf{I} = (U^{f_{id}(\mathbf{y})} g^{f_{id}(\mathbf{tag}^*)})$

$$C_1 = \prod_{i=1}^{n} \mathbf{G}[i]^{x[i]} = g^{\langle \mathbf{s}, x \rangle} \qquad C_2 = \prod_{i=1}^{n} \hat{\mathbf{G}}[i]^{x[i]} = \hat{g}^{\langle \hat{\mathbf{s}}, x \rangle}$$

$$\begin{aligned}
\mathbf{C}_3[j] &= \mathbf{S}[j] \prod_{i=1}^{n} (\mathbf{W}_0[i,j] \mathbf{W}_1[i,j]^{id})^{x[i]} \\
&= \mathbf{S}[j] \prod_{i=1}^{n} (\mathbf{V}_0[j] \mathbf{V}_1[j]^{id})^{\mathbf{s}[i]x[i]} (\hat{\mathbf{V}}_0[j] \hat{\mathbf{V}}_1[j]^{id})^{\hat{\mathbf{s}}[i]x[i]} \mathbf{I}^{\mathbf{s}[i]x[i]\Delta(i,j)} \\
&= (\mathbf{V}_0[j] \cdot \mathbf{V}_1[j]^{id})^{\langle \mathbf{s}, x \rangle} (\hat{\mathbf{V}}_0[j] \cdot \hat{\mathbf{V}}_1[j]^{id})^{\langle \hat{\mathbf{s}}, x \rangle} (U^{f_{id}(\mathbf{y})} g^{f_{id}(\mathbf{tag}^* - \mathbf{tag})})^{\mathbf{s}[j]x[j]} \\
\mathbf{C}_4[j] &= \prod_{i=1}^{n} \mathbf{J}[i,j]^{x[i]} = \prod_{i=1}^{n} \mathbf{H}[j]^{\mathbf{s}[i]x[i]} \hat{\mathbf{H}}[j]^{\hat{\mathbf{s}}[i]x[i]} = \mathbf{H}[j]^{\langle \mathbf{s}, x \rangle} \hat{\mathbf{H}}[j]^{\langle \hat{\mathbf{s}}, x \rangle}
\end{aligned}$$

Thus

$$\begin{aligned}
&e(C_1, \mathbf{D}_1[j]) e(C_2, \mathbf{D}_2[j]) e(\mathbf{C}_3[j], \mathbf{D}_3[j]) e(\mathbf{C}_4[j], \mathbf{D}_4[j]) \\
&\quad = e((U^{f_{id}(\mathbf{y})} g^{f_{id}(\mathbf{tag}^* - \mathbf{tag})})^{\mathbf{s}[j]x[j]}, \mathbf{D}_3[j])
\end{aligned}$$

Because we chose $s[i]$ to be non-zero modulo p, $f_{id}(\mathbf{y}) = 1 mod p \neq 0$, therefore, $(id, \mathbf{tag}) \notin \tilde{B}$, if

$$e(C_1, \mathbf{D}_1[j]) e(C_2, \mathbf{D}_2[j]) e(\mathbf{C}_3[j], \mathbf{D}_3[j]) e(\mathbf{C}_4[j], \mathbf{D}_4[j]) = \mathbf{1}_T,$$

then $x[j] = 0$, if the result of the pairing is never $\mathbf{1}_T$, then the inversion algorithm will correctly recover $x[j] = 1$.

Lossiness. On input a selective identity $id^* \in$ IDSp, the algorithm Aux(id^*) returns $\mathbf{y} = (-id^*, 1)$. Obviously, $f_{id^*}(\mathbf{y}) = 0 mod p$, when $\mathbf{tag} = \mathbf{tag}^*$, we have $(id^*, \mathbf{tag}) \in \tilde{B}$. We show that if $(id, \mathbf{tag}) \in \tilde{B}$, then algorithm Eval$_{abo}(mpk, id, \mathbf{tag}, \cdot)$ evaluates a lossy function. Due to $f_{id}(\mathbf{y}) = 0 \bmod p$ and $\mathbf{tag} = \mathbf{tag}^*$, then the dependency of $\mathbf{C}_3[j]$ on $x[j]$ vanishes. Examing $(C_1, C_2, \mathbf{C}_3, \mathbf{C}_4)$, we see that with mpk fixed, the values $\langle \mathbf{s}, x \rangle, \langle \hat{\mathbf{s}}, x \rangle$ determine the ciphertext C. Thus there are at most p^2 possible ciphertexts when $(id, \mathbf{tag}) \in \tilde{B}$. This means that $R = |\text{Eval}_{abo}(mpk, id, \mathbf{tag}, \cdot)| \leq p^2$. Moreover, the lossy branch of IB-ABO-TDF is universal. That is, if $x_1 \neq x_2$, for $\mathbf{s}, \hat{\mathbf{s}} \xleftarrow{\$} \mathbb{Z}_p^n$, $\Pr[\text{Eval}_{abo}(mpk, id, \mathbf{tag}, x_1) = \text{Eval}_{abo}(mpk, id, \mathbf{tag}, x_2)] = \Pr[(\langle \mathbf{s}, x_1 \rangle, \langle \hat{\mathbf{s}}, x_1 \rangle) = (\langle \mathbf{s}, x_2 \rangle, \langle \hat{\mathbf{s}}, x_2 \rangle)] = \frac{1}{p^2} \leq \frac{1}{|R|}$.

Indistinguishability of lossy branches. From Theorem 1, we can see that, under the decision linear (DLIN) assumption, it is hard to distinguish the master public key and random group elements. Therefore, the first output mpk_1 of $Setup(1^\lambda, \mathbf{y}_1, \mathbf{tag}_1)$ and the first output mpk_2 of $Setup(1^\lambda, \mathbf{y}_2, \mathbf{tag}_2)$ are computationally indistinguishable.

Hard to find one-more lossy branch. We show that any probabilistic polynomial time adversary $\mathcal{A}$ that receives $(mpk, id, \mathbf{tag})$ as input, where $(id, \mathbf{tag}) \in \tilde{B}$, outputs a pair $(id', \mathbf{tag}')$ satisfying $(id', \mathbf{tag}') \neq (id, \mathbf{tag})$ and $(id', \mathbf{tag}') \in \tilde{B}$

with negligible probability. To see this, observe that the value $\mathbf{y}$ and $\mathbf{tag}^*$ are initially hidden by the public parameter $W_0[i,j], W_1[i,j]$, from Theorem 1, we know that $W_0[i,j], W_1[i,j]$ are indistinguishable from a random element in the group $\mathbb{G}$ to any probabilistic polynomial time adversary $\mathcal{A}$. However, $\mathcal{A}$ could obtain the information that $uf_{id}(\mathbf{y})+f_{id}(\mathbf{tag}^*-\mathbf{tag}) = 0$ (let $U = g^u$). There are exactly p^2 pairs which satisfy this equation and each of them are equally likely. Therefore, we can conclude that the adversary has only negligible probability of outputing a pair $(id', tag') \in \tilde{B}\backslash\{id, tag\}$ without master secret key.

In order to prove the indistinguishability of lossy branches (i.e Theorem 1), we first prove the following lemma. That is, the basic ciphertext which contain certain 'atoms" from which, given an identity, one can reconstruct ciphertext of the resemble Bellar et al.'s basic IBE scheme, is indistinguishable from random group elements. The concrete games are described in table 2.

Lemma 3.1. Game G_{real} and game G_{random} are computationally indistinguishable under DLIN assumption.

Proof. This proof via a series of games G_0, G_1, G_2, G_3, where game G_0 is the game G_{real} and game G_3 is the game G_{random}. Game G_1 is the same as game G_0 except $S \xleftarrow{\$} \mathbb{G}$ in challenge phase, Game G_2 is the same as game G_1 except $W_0 \xleftarrow{\$} \mathbb{G}$ in challenge phase. Game G_3 is the same as game G_2 except $W_1 \xleftarrow{\$} \mathbb{G}$ in challenge phase. We then show that for $i = 0,1,2$, G_i and G_{i+1} are computationally indistinguishable under DLIN assumption. It follows that the Lemma holds. The advantage of a distinguisher $\mathcal{B}$ attacking DLIN assumption is denoted by $Adv_{\mathcal{B}}^{\text{DLIN}}(\lambda)$.

Claim 1. $\Pr[G_0 \Rightarrow 1] - \Pr[G_1 \Rightarrow 1] \leq Adv_{\mathcal{B}_1}^{\text{DLIN}}(\lambda)$.

Proof. We prove this claim by describing a distinguisher $\mathcal{B}_1$ is given $(g, \hat{g}, g^s, \hat{g}^{\hat{s}}, H, T)$ where T is either $H^{s+\hat{s}}$ or random. $\mathcal{B}_1$ runs adversary $\mathcal{A}$ responding to its queries as follows. When $\mathcal{A}$ makes query to Initialize$(i^\lambda, \mathbf{y}, \mathbf{tag})$, $\mathcal{B}_1$ chooses $\mathbf{y} \in \mathbb{Z}_p^2$, $\mathbf{tag} \in \mathbb{Z}_p^2$, $u, v \xleftarrow{\$} \mathbb{Z}_p$, $\mathbf{v} = (v_0, v_1), \hat{\mathbf{v}} = (\hat{v}_0, \hat{v}_1) \xleftarrow{\$} \mathbb{Z}_p^2$ and computes $\hat{H} = H\hat{g}^v$, $U = \hat{g}^u$, for $k = 0,1$, $V_k = U^{-y_k} g^{tag_k} g^{v_k}$, $\hat{V}_k = \hat{g}^{\hat{v}_k}$. $\mathcal{B}_1$ returns $(g, \hat{g}, H, \hat{H}, V_0, V_1, \hat{V}_0, \hat{V}_1, U)$ to $\mathcal{A}$.

When $\mathcal{A}$ makes private-key query to Getsk(id), $\mathcal{B}_1$ does the following: If $f_{id}(\mathbf{y}) = 0$ then return $\perp$. Else $\mathcal{B}_1$ chooses $r', \hat{r}' \xleftarrow{\$} \mathbb{Z}_p$, and computes

$$D_1 = U^{-f_{id}(\mathbf{y})r'} g^{(f_{id}(\mathbf{tag})+f_{id}(\mathbf{v}))r'} H^{\frac{\hat{r}'(f_{id}(\mathbf{v})+f_{id}(\mathbf{tag}))}{f_{id}(\mathbf{y})}}$$

$$D_2 = g^{r' f_{id}(\hat{\mathbf{v}})} H^{-\frac{\hat{r}' f_{id}(\hat{\mathbf{v}})}{f_{id}(\mathbf{y})}} H^{u\hat{r}'},\ D_3 = g^{-r'} H^{\frac{\hat{r}'}{f_{id}(\mathbf{y})}},\ D_4 = \hat{g}^{-u\hat{r}'}$$

Then $\mathcal{B}_1$ returns $sk_{id} = (D_1, D_2, D_3, D_4)$ to $\mathcal{A}$.

We claim that sk_{id} is a correctly distributed and valid random private key for the identity id. To see this, let h be such that $H = g^h$ and let $r = \frac{r'}{t} - \frac{h\hat{r}'}{tf_{id}(\mathbf{y})}$

G_{real}:	G_{random}:
Initialize$(1^\lambda, \mathbf{y}, \mathbf{tag})$	**Initialize**$(1^\lambda, \mathbf{y}, \mathbf{tag})$
$g \xleftarrow{\$} \mathbb{G}^*$; $t \xleftarrow{\$} \mathbb{Z}_p^*$; $H, \hat{H} \xleftarrow{\$} \mathbb{G}$	$g \xleftarrow{\$} \mathbb{G}^*$; $t \xleftarrow{\$} \mathbb{Z}_p^*$; $H, \hat{H} \xleftarrow{\$} \mathbb{G}$
$U \xleftarrow{\$} \mathbb{G}^*$; $\mathbf{V} = (V_0, V_1) \xleftarrow{\$} \mathbb{G}^2$	$U \xleftarrow{\$} \mathbb{G}^*$ $\mathbf{V} = (V_0, V_1) \xleftarrow{\$} \mathbb{G}^2$
$\hat{\mathbf{V}} = (\hat{V}_0, \hat{V}_1) \xleftarrow{\$} \mathbb{G}^2$; $msk \leftarrow t$	$\hat{\mathbf{V}} = (\hat{V}_0, \hat{V}_1) \xleftarrow{\$} \mathbb{G}^2$; $msk \leftarrow t$
$mpk \leftarrow (g, \hat{g}, H, \hat{H}, \mathbf{V}, \hat{\mathbf{V}}, U)$	$mpk \leftarrow (g, \hat{g}, H, \hat{H}, \mathbf{V}, \hat{\mathbf{V}}, U)$
Return mpk	Return mpk
Getsk(id)	**Getsk**(id)
If $f_{id}(\mathbf{y}) = 0$ then $sk_{id} \leftarrow \perp$	If $f_{id}(\mathbf{y}) = 0$ then $sk_{id} \leftarrow \perp$
Else $sk_{id} \leftarrow (D_1, D_2, D_3, D_4)$	Else $sk_{id} \leftarrow (D_1, D_2, D_3, D_4)$
$r, \hat{r} \xleftarrow{\$} \mathbb{Z}_p^*$; $D_1 \leftarrow (V_0 V_1^{id})^{tr} H^{t\hat{r}}$	$r, \hat{r} \xleftarrow{\$} \mathbb{Z}_p^*$; $D_1 \leftarrow (V_0 V_1^{id})^{tr} H^{t\hat{r}}$
$D_2 \leftarrow (\hat{V}_0 \hat{V}_1^{id})^r \hat{H}^{\hat{r}}$	$D_2 \leftarrow (\hat{V}_0 \hat{V}_1^{id})^r \hat{H}^{\hat{r}}$
$D_3 \leftarrow g^{-tr}$, $D_4 \leftarrow g^{-t\hat{r}}$	$D_3 \leftarrow g^{-tr}$, $D_4 \leftarrow g^{-t\hat{r}}$
Challenge (it has no identity input)	**Challenge** (it has no identity input)
$s \xleftarrow{\$} \mathbb{Z}_p^*$; $\hat{s} \xleftarrow{\$} \mathbb{Z}_p$; $G \leftarrow g^s$; $\hat{G} \leftarrow \hat{g}^{\hat{s}}$	$\boxed{G \xleftarrow{\$} \mathbb{G};\ \hat{G} \xleftarrow{\$} \mathbb{G}}$
$S \leftarrow H^s \hat{H}^{\hat{s}}$	$\boxed{S \xleftarrow{\$} \mathbb{G}}$
$W_0 \leftarrow (U^{y_0} g^{tag_0} V_0)^s \hat{V}_0^{\hat{s}}$	$\boxed{W_0 \xleftarrow{\$} \mathbb{G}}$
$W_1 \leftarrow (U^{y_1} g^{tag_1} V_1)^s \hat{V}_1^{\hat{s}}$	$\boxed{W_1 \xleftarrow{\$} \mathbb{G}}$
Return $(G, \hat{G}, S, W_0, W_1)$	Return $(G, \hat{G}, S, W_0, W_1)$
Finalize$((d'))$	**Finalize**$((d'))$
Return $d' = 1$	Return $d' = 1$

Table 2: Games for the proof of Lemma 3.1. Border areas indicate the difference between the games

and $\hat{r} = u\hat{r}'$. Then we have

$$\begin{aligned}
D_1 = (V_0 V_1^{id})^{tr} H^{t\hat{r}} &= (U^{-y_0} g^{tag_0} g^{v_0} (U^{-y_1} g^{tag_1} g^{v_1})^{id})^{tr} H^{t\hat{r}} \\
&= (U^{-f_{id}(\mathbf{y})} g^{f_{id}(\mathbf{tag})} g^{f_{id}(\mathbf{v})})^{t(\frac{r'}{t} - \frac{h\hat{r}'}{t f_{id}(\mathbf{y})})} H^{tu\hat{r}'} \\
&= U^{-f_{id}(\mathbf{y})r'} U^{h\hat{r}'} g^{(f_{id}(\mathbf{tag})+f_{id}(\mathbf{v}))r'} g^{\frac{-(f_{id}(\mathbf{tag})+f_{id}(\mathbf{v}))h\hat{r}'}{f_{id}(\mathbf{y})}} g^{htu\hat{r}'} \\
&= U^{-f_{id}(\mathbf{y})r'} \hat{g}^{uh\hat{r}'} g^{(f_{id}(\mathbf{tag})+f_{id}(\mathbf{v}))r'} H^{\frac{-(f_{id}(\mathbf{tag})+f_{id}(\mathbf{v}))\hat{r}'}{f_{id}(\mathbf{y})}} \hat{g}^{hu\hat{r}'} \\
&= U^{-f_{id}(\mathbf{y})r'} g^{(f_{id}(\mathbf{tag})+f_{id}(\mathbf{v}))r'} H^{\frac{\hat{r}'(f_{id}(\mathbf{v})+f_{id}(\mathbf{tag}))}{f_{id}(\mathbf{y})}}
\end{aligned}$$

$$\begin{aligned} D_2 = (\hat{V}_0\hat{V}_1^{id})^r\hat{H}^{\hat{r}} = (\hat{g}^{v_0}\hat{g}^{v_1 id})^r\hat{H}^{\hat{r}} &= g^{tf_{id}(\hat{\mathbf{v}})(\frac{r'}{t}-\frac{h\hat{r}'}{tf_{id}(\mathbf{y})})}\hat{H}^{u\hat{r}'} \\ &= g^{f_{id}(\hat{\mathbf{v}})r'}g^{-\frac{h\hat{r}' f_{id}(\hat{\mathbf{v}})}{f_{id}(\mathbf{y})}}\hat{H}^{u\hat{r}'} = g^{f_{id}(\hat{\mathbf{v}})r'}H^{-\frac{\hat{r}' f_{id}(\hat{\mathbf{v}})}{f_{id}(\mathbf{y})}}\hat{H}^{u\hat{r}'} \\ D_3 = g^{-tr} &= g^{-t(\frac{r'}{t}-\frac{h\hat{r}'}{tf_{id}(\mathbf{y})})} = g^{-r'}H^{\frac{\hat{r}'}{f_{id}(\mathbf{y})}} \\ D_4 = g^{-t\hat{r}} &= g^{-tu\hat{r}'} = \hat{g}^{-u\hat{r}'} \end{aligned}$$

Since t, $f_{id}(\mathbf{y})$ are non-zero module p and $r', \hat{r}'$ are uniform and independent in $\mathbb{Z}_p$, $r, \hat{r}$ are uniform as well. This matches the distribution of private key of identity id generated by Getsk. Thus, sk_{id} is a valid private key of the identity id.

When $\mathcal{A}$ makes its Challenge query, $\mathcal{B}_1$ computes $S = T\hat{g}^{v\hat{s}}$, for $k = 0, 1$, do $W_k = g^{sv_k}\hat{g}^{\hat{s}\hat{v}_k}$ and returns $(g, \hat{g}^{\hat{s}}, S, W_0, W_1)$ to $\mathcal{A}$. We can see that for $k = 0, 1$,

$$W_k = g^{sv_k}\hat{g}^{\hat{s}\hat{v}_k} = (U^{y_k}g^{tag_k}U^{-y_k}g^{-tag_k}g^{v_k})^s(\hat{g}^{\hat{v}_k})^{\hat{s}} = (U^{y_k}g^{tag_k}V_k)^s\hat{V}_k^{\hat{s}}$$

If $\mathcal{B}_1$ was given a DLIN instance, that is $T = H^{s+\hat{s}}$, then we have

$$S = T\hat{g}^{v\hat{s}} = H^{s+\hat{s}}\hat{g}^{v\hat{s}} = H^s(H\hat{g}^v)^{\hat{s}} = H^s\hat{H}^{\hat{s}}$$

We see that $\mathcal{B}_1$ simulates game G_0. Otherwise, $\mathcal{B}_1$ was given a non-DLIN instance, i.e $T \xleftarrow{\$} \mathbb{G}$, then $S \xleftarrow{\$} \mathbb{G}$, $\mathcal{B}_1$ simulates game G_1. Finally, $\mathcal{A}$ outputs d', $\mathcal{B}_1$ also outputs d'. So this completes the claim.

Claim 2. $\Pr[G_1 \Rightarrow 1] - \Pr[G_2 \Rightarrow 1] \leq Adv_{\mathcal{B}_2}^{\mathrm{DLIN}}(\lambda)$.

Proof. Similar to claim 1, we design a simulator $\mathcal{B}_2$ such that it is given $(g, \hat{g}, g^s, \hat{g}^{\hat{s}}, \hat{U}, T)$ where T is either $\hat{U}^{s+\hat{s}}$ or random. $\mathcal{B}_2$ runs adversary $\mathcal{A}$ responding to its queries as follows. When $\mathcal{A}$ makes query Initialize($i^\lambda, \mathbf{y}, \mathbf{tag}$), $\mathcal{B}_2$ chooses $\mathbf{y} \in$ AuxSp, $\mathbf{tag} \in$ TagSp, $u, h, \hat{h} \xleftarrow{\$} \mathbb{Z}_p$, $\mathbf{v} = (v_0, v_1), \hat{\mathbf{v}} = (\hat{v}_0, \hat{v}_1) \xleftarrow{\$} \mathbb{Z}_p^2$, and computes $H \leftarrow \hat{g}^h$, $\hat{H} = \hat{g}^{\hat{h}}$, $U = g^u$, $V_0 = \hat{U}g^{v_0}$, $V_1 = g^{v_1}$, $\hat{V}_0 = \hat{U}\hat{g}^{\hat{v}_0}$, $\hat{V}_1 = \hat{g}^{\hat{v}_1}$. $\mathcal{B}_2$ returns $(g, \hat{g}, H, \hat{H}, V_0, V_1, \hat{V}_0, \hat{V}_1, U)$ to adversary $\mathcal{A}$.

When $\mathcal{A}$ makes query private-key Getsk(id), $\mathcal{B}_2$ does the following: If $f_{id}(\mathbf{y}) = 0$ then return $\perp$. Else $\mathcal{B}_2$ chooses $r, \hat{r}' \xleftarrow{\$} \mathbb{Z}_p$, and computes

$$D_1 = \hat{g}^{f_{id}(\mathbf{v})r}H^{\hat{r}'}, \quad D_2 = \hat{U}^r\hat{g}^{rf_{id}(\hat{\mathbf{v}})}H^{\hat{r}'}\hat{U}^{\frac{-\hat{h}r}{h}}$$

$$D_3 = \hat{g}^{-r}, \quad D_4 = g^{-\hat{r}'}\hat{U}^{\frac{r}{h}}$$

Then $\mathcal{B}_2$ returns $sk_{id} = (D_1, D_2, D_3, D_4)$ to $\mathcal{A}$.

We claim that sk_{id} is a correctly distributed and valid random private key for the identity id. To see this, let $\hat{u}$ be such that $\hat{U} = g^{\hat{u}}$ and let $\hat{r} = \frac{\hat{r}'}{t} - \frac{\hat{u}r}{ht}$.

Then we have

$$
\begin{aligned}
D_1 = (V_0 V_1^{id})^{tr} H^{t\hat{r}} &= (\hat{U} g^{v_0} g^{v_1})^{tr} H^{t\hat{r}} \\
&= (\hat{U}^{tr} g^{f_{id}(\mathbf{v})})^{tr} \hat{g}^{ht\hat{r}} \\
&= (\hat{U}^{tr} g^{f_{id}(\mathbf{v})})^{tr} \hat{g}^{ht(\frac{\hat{r}'}{t} - \frac{\hat{u}r}{ht})} \\
&= (g^{\hat{u}tr} g^{f_{id}(\mathbf{v})})^{tr} \hat{g}^{h\hat{r}'} \hat{g}^{-\hat{u}r} \\
&= (g^{f_{id}(\mathbf{v})})^{tr} H^{\hat{r}'} \\
D_2 = (\hat{V}_0 \hat{V}_1^{id})^{r} \hat{H}^{\hat{r}} &= (\hat{U} \hat{g}^{\hat{v}_0} \hat{g}^{\hat{v}_1 id})^{r} \hat{H}^{\hat{r}} \\
&= \hat{U}^{r} \hat{g}^{r f_{id}(\hat{\mathbf{v}})} \hat{H}^{(\frac{\hat{r}'}{t} - \frac{\hat{u}r}{ht})} \\
&= \hat{U}^{r} \hat{g}^{r f_{id}(\hat{\mathbf{v}})} \hat{g}^{\hat{h}(\frac{\hat{r}'}{t} - \frac{\hat{u}r}{ht})} \\
&= \hat{U}^{r} \hat{g}^{r f_{id}(\hat{\mathbf{v}})} g^{\hat{h}\hat{r}'} g^{\frac{-\hat{h}\hat{u}r}{h}} \\
&= \hat{U}^{r} \hat{g}^{r f_{id}(\hat{\mathbf{v}})} H^{\hat{r}'} \hat{U}^{\frac{-\hat{h}r}{h}} \\
D_3 = g^{-tr} &= \hat{g}^{-r} \\
D_4 = g^{-t\hat{r}} &= g^{-t(\frac{\hat{r}'}{t} - \frac{\hat{u}r}{ht})} = g^{-\hat{r}'} g^{\frac{\hat{u}r}{h}} = g^{-\hat{r}'} \hat{U}^{\frac{r}{h}}
\end{aligned}
$$

Since t, $f_{id}(\mathbf{y})$ are non-zero module p and $\hat{r}'$ are uniform and independent in $\mathbb{Z}_p$, $\hat{r}$ are uniform as well. This matches the distribution of private key of identity id generated by Getsk. Thus, sk_{id} is a valid private key of the identity id.

When $\mathcal{A}$ makes its Challenge query, $\mathcal{B}_2$ computes $W_0 = (g^s)^{uy_0+tag_0+v_0}(\hat{g}^{\hat{s}})^{\hat{v}_0} \cdot T$, $W_1 = (g^s)^{uy_1+tag_1+v_1}(\hat{g}^{\hat{s}})^{\hat{v}_1}$ and returns $(g, \hat{g}^{\hat{s}}, S, W_0, W_1)$ to $\mathcal{A}$. We can see that

$$W_1 = (g^s)^{uy_1+tag_1+v_1}(\hat{g}^{\hat{s}})^{\hat{v}_1} = (U^{y_1} g^{tag_1} V_1)^s \hat{V}_1^{\hat{s}}$$

If $\mathcal{B}_2$ was given a DLIN instance, that is $T = \hat{U}^{s+\hat{s}}$, then we have

$$
\begin{aligned}
W_0 &= (g^s)^{uy_0+tag_0+v_0}(\hat{g}^{\hat{s}})^{\hat{v}_0} \cdot T \\
&= (U^{y_0} g^{tag_0} \hat{U}^{-1} V_0)^s (\hat{U}^{-1} \hat{V}_0)^{\hat{s}} \hat{U}^{s+\hat{s}} \\
&= (U^{y_0} g^{tag_0} V_0)^s \hat{U}^{-s} \hat{U}^{-\hat{s}} \hat{V}_0^{\hat{s}} \hat{U}^{s+\hat{s}} \\
&= (U^{y_0} g^{tag_0} V_0)^s \hat{V}_0^{\hat{s}}
\end{aligned}
$$

We see that $\mathcal{B}_2$ simulates game G_1 in this case. Otherwise, $\mathcal{B}_2$ was given a non-DLIN instance, i.e $T \xleftarrow{\$} \mathbb{G}$, then $W_0 \xleftarrow{\$} \mathbb{G}$, $\mathcal{B}_2$ simulates game G_2. Finally, $\mathcal{A}$ outputs d', $\mathcal{B}_2$ also outputs d'. So this completes the claim.

Claim 3. $\Pr[G_2 \Rightarrow 1] - \Pr[G_3 \Rightarrow 1] \leq Adv_{\mathcal{B}_2}^{\mathrm{DLIN}}(\lambda)$.

Proof. The proof is identical to that of Claim 2. The only differences are, in the Initialize($i^\lambda, \mathbf{y}, \mathbf{tag}$) query phase, the simulator $\mathcal{B}_2$ lets $V_0 = g^{v_0}$, $V_1 = \hat{U} g^{v_1}$, $\hat{V}_0 = \hat{g}^{\hat{v}_0}$, $\hat{V}_1 = \hat{U} \hat{g}^{\hat{v}_1}$, in the private key query phase, lets $\hat{r} = \frac{\hat{r}'}{t} - \frac{id\hat{u}r}{ht}$, and in the challenge phrase, the simulator $\mathcal{B}_2$ computes $W_0 = (g^s)^{uy_0+tag_0+v_0}(\hat{g}^{\hat{s}})^{\hat{v}_0}$ and $W_1 = (g^s)^{uy_1+tag_1+v_1}(\hat{g}^{\hat{s}})^{\hat{v}_1} \cdot T$.

Theorem 1. The first output mpk_1 of $\text{Setup}_{abo}(1^\lambda, \mathbf{y}_1, \mathbf{tag}_1)$ and the first output mpk_2 of $\text{Setup}_{abo}(1^\lambda, \mathbf{y}_2, \mathbf{tag}_2)$ are computationally indistinguishable.

Proof. We first prove that mpk_b for $b \in \{0,1\}$ and random group element matrices are computationally indistinguishable. It is obvious that the theorem follows. To prove mpk_b generated by $\text{Setup}_{abo}(1^\lambda, \mathbf{y}_b, \mathbf{tag}_b)$ is indistinguishable from random group element matrices, we define a set of hybrid games $R_1, \cdots, R_n$. In R_l $(1 \le l \le n)$, where game R_1 produces a real public parameter matrices and R_n produces a random public parameter matrices. Below we argue that for every $l \in \{1, \ldots, n\}$, any distinguisher $\mathcal{A}$ of the two games R_{l-1} and R_l can be used to distinguish G_{real} from G_{random}. From Lemma 3.1, R_{l-1} and R_l are indistinguishable, therefore, this theorem holds.

Let adversary $\mathcal{A}$ be a distinguisher of R_{l-1} and R_l, the simulator $\mathcal{B}$ is an adversary distinguishing G_{real} from G_{random}.

Simulating public parameter. The simulator $\mathcal{B}$ is given $(g, \hat{g}, H, \hat{H}, V_0, V_1, \hat{V}_0, \hat{V}_1, U) \leftarrow \text{Inaitialize}(1^\lambda, \mathbf{y}, \mathbf{tag})$ and $(G, \hat{G}, S, W_0, W_1) \leftarrow$ Challenge where $(G, \hat{G}, S, W_0, W_1)$ is either real or random. $\mathcal{B}$ chooses $\mathbf{h}, \hat{\mathbf{h}} \xleftarrow{\$} (\mathbb{Z}_p^*)^n$, $\mathbf{v}_0$, $\mathbf{v}_1$, $\hat{\mathbf{v}}_0$, $\hat{\mathbf{v}}_1 \xleftarrow{\$} \mathbb{Z}_p^n$, $\mathbf{s} \xleftarrow{\$} (\mathbb{Z}_p^*)^n$, $\hat{\mathbf{s}} \xleftarrow{\$} \mathbb{Z}_p^n$. For $i = 1, \ldots, n$

- If $i \neq l$ then $\mathbf{H}[i] \leftarrow g^{\mathbf{h}[i]}$, $\hat{\mathbf{H}}[i] \leftarrow g^{\hat{\mathbf{h}}[i]}$, $\mathbf{V}_k[i] \leftarrow g^{\mathbf{v}_k[i]}$, $\hat{\mathbf{V}}_k[i] \leftarrow g^{\hat{\mathbf{v}}_k[i]}$ $(k = 0, 1)$, $\mathbf{G}[i] \leftarrow g^{\mathbf{s}[i]}$, $\hat{\mathbf{G}}[i] \leftarrow g^{\hat{\mathbf{s}}[i]}$. For $j = 1, \ldots, n$, $\mathbf{J}[i,j] \leftarrow \mathbf{H}[j]^{\mathbf{s}[i]}\hat{\mathbf{H}}[j]^{\hat{\mathbf{s}}[i]}$.
- If $i = l$ then $\mathbf{H}[i] \leftarrow H$, $\hat{\mathbf{H}}[i] \leftarrow \hat{H}$, $\mathbf{V}_k[i] \leftarrow V_k$, $\hat{\mathbf{V}}_k[i] \leftarrow \hat{V}_k$ $(k = 0, 1)$, $\mathbf{G}[i] \leftarrow G$, $\hat{\mathbf{G}}[i] \leftarrow \hat{G}$. For $j = 1, \ldots, n$, if $j \neq i$ then $\mathbf{J}[i,j] \leftarrow G^{\mathbf{h}[j]}\hat{G}^{\hat{\mathbf{h}}[j]}$, if $j = i$ then $\mathbf{J}[i,j] \leftarrow S$.
- For $j = 1, \ldots, n$, $k = 0, 1$, if $i = j$ and $i \le l - 1$ then $\mathbf{W}_k[i,j] \xleftarrow{\$} \mathbb{G}$; if $i = j$ and $i = l$ then $\mathbf{W}_k[i,j] \leftarrow W_k$; otherwise

$$\mathbf{W}_k[i,j] \leftarrow \mathbf{V}_k[j]^{\mathbf{s}[i]}\hat{\mathbf{V}}_k[j]^{\hat{\mathbf{s}}[i]}(U^{y_k}g^{tag_k})^{s[i]\Delta(i,j)}$$

$\mathcal{B}$ then sends $paras = (g, \hat{g}, \mathbf{H}, \hat{\mathbf{H}}, \mathbf{V}_0, \mathbf{V}_1, \hat{\mathbf{V}}_0, \hat{\mathbf{V}}_1, \mathbf{G}, \hat{\mathbf{G}}, \mathbf{J}, \mathbf{W}_0, \mathbf{W}_1, U)$ to $\mathcal{A}$.

Simulating private key. When the simulator $\mathcal{B}$ simulates the private key sk_{id} for identity id which is chosen by the adversary $\mathcal{A}$, $\mathcal{B}$ first queries its own Getsk oracle and is given $(D_1, D_2, D_3, D_4) \leftarrow \text{Getsk}(id)$. If $f_{id}(\mathbf{y}) = 0$ then $\mathcal{B}$ returns $\perp$. Otherwise, $\mathcal{B}$ chooses $\mathbf{r}' \xleftarrow{\$} (\mathbb{Z}_p^*)^n$, $\hat{\mathbf{r}}' \xleftarrow{\$} (\mathbb{Z}_p)^n$. For $i = 1, \ldots, n$

- If $i \neq l$ then $\mathbf{D}_1[i] \leftarrow (V_0[i]V_1[i]^{id})^{\mathbf{r}'[i]}H[i]^{\hat{\mathbf{r}}'[i]}$, $\mathbf{D}_2[i] \leftarrow g^{f_{id}(\hat{\mathbf{v}})\mathbf{r}'[i]}g^{\hat{h}[i]\hat{\mathbf{r}}'[i]}$, $\mathbf{D}_3[i] \leftarrow g^{-\mathbf{r}'[i]}$, $\mathbf{D}_4[i] \leftarrow g^{-\hat{\mathbf{r}}'[i]}$.
- If $i = l$ then $(\mathbf{D}_1[i], \mathbf{D}_2[i], \mathbf{D}_3[i], \mathbf{D}_4[i]) \leftarrow (D_1, D_2, D_3, D_4)$.

Where $\mathbf{r}[i] = \frac{\mathbf{r}'[i]}{t}$, $\hat{\mathbf{r}}[i] = \frac{\hat{\mathbf{r}}'[i]}{t}$ for $i \neq l$. Here t is the master secret key, so that $\hat{g} = g^t$. Because $\mathbf{r}'[i], \hat{\mathbf{r}}'[i]$ are random, so the above simulation matches the distribution of the private key.

Note that if $(G, \hat{G}, S, W_0, W_1)$ is real, $\mathcal{B}$ perfectly simulates the game R_{l-1}, but if $(G, \hat{G}, S, W_0, W_1)$ is random, $\mathcal{B}$ perfectly simulates the game R_l. Finally, $\mathcal{B}$ outputs what $\mathcal{A}$ outputs. Since $\mathcal{B}$ perfectly simulates game R_{l-1} or game R_l depending on the $(G, \hat{G}, S, W_0, W_1)$. This completes the proof of the theorem.

4 CCA-secure DIBE scheme

Given the identity space IDSp and auxiliary input space AuxSp, let $\Pi_{\mathrm{LF}} =$ (LF.Setup, LF.KG, LF.Eval, LF.Inv) be an identity-based lossy trapdoor function with $2^{r_{\mathrm{LF}}}$ -bounded lossy function range R_{LF} (i.e., in the lossy mode, the image of Π_{LF} has size at most $2^{r_{\mathrm{LF}}}$), let $\Pi_{abo} = (\mathrm{Setup}_{abo}, \mathrm{KG}_{abo}, \mathrm{Eval}_{abo}, \mathrm{Inv}_{abo})$ be an identity-based all-but-one trapdoor function with branches set $B = \mathrm{IDSp} \times$ TagSp (TagSp is tag space) and with $2^{r_{abo}}$ -bounded lossy function range R_{abo}, and let $\mathcal{H}_{tcr} = (K_{tcr}, H_{tcr})$ be a target collision-resistant hash function with $2^{r_{tcr}}$ -bounded hash range $R_{tcr} \subseteq \mathrm{TagSp} \backslash \{tag^*\}$. We assume that the DIBE scheme has message space $\{0,1\}^l$. Our deterministic identity-based encryption scheme $\mathrm{DE_{IB}} = (\mathrm{DIB.Setup}, \mathrm{DIB.Der}, \mathrm{DIB.Enc}, \mathrm{DIB.Dec})$ is defined as follows.

DIB.Setup(1^λ). $aux_0, aux_1 \xleftarrow{\$} \mathrm{AuxSp}$, $(mpk_{\mathrm{LF}}, msk_{\mathrm{LF}}) \leftarrow \mathrm{LF.Setup}(1^\lambda, aux_0)$, $(mpk_{abo}, msk_{abo}) \leftarrow \mathrm{Setup}_{abo}(1^\lambda, aux_1, tag^*)$, $k_{tcr} \xleftarrow{\$} K_{tcr}$. Return $mpk = (mpk_{\mathrm{LF}}, mpk_{abo}, k_{tcr})$, $msk = msk_{\mathrm{LF}}$.

DIB.Der(mpk, msk, id). $sk_{id} \leftarrow \mathrm{LF.KG}(mpk_{\mathrm{LF}}, msk_{\mathrm{LF}}, id)$. Return sk_{id}.

DIB.Enc(mpk, id, m). $h \leftarrow H_{tcr}(m)$, $c_1 \leftarrow \mathrm{LF.Eval}(mpk_{\mathrm{LF}}, id, m)$, $c_2 \leftarrow \mathrm{Eval}_{abo}$ (mpk_{abo}, id, h, m). Return $C = h \| c_1 \| c_2$.

DIB.Dec(mpk, id, sk_{id}, C). Parse C as $C = h \| c_1 \| c_2$, compute $m' \leftarrow \mathrm{LF.Inv}$ $(mpk_{\mathrm{LF}}, sk_{id}, c_1)$, $C' \leftarrow \mathrm{DIB.Enc}(mpk, id, m')$. If $C' = C$ then return m', otherwise return $\perp$.

Note that consistency of the above scheme follows from the fact that the particular auxiliary input aux_0 is a constant and independent to the identity, and the range of the tcr hash function does not include the lossy branch of the identity-based ABO-TDF. So both $\mathrm{LF.Eval}(mpk_{\mathrm{LF}}, id, \cdot)$ and $\mathrm{Eval}_{abo}(mpk_{abo}, id, h, \cdot)$ are injective trapdoor functions. For all output (mpk, msk) by DIB.Setup and all $m \in \{0,1\}^l$, there is exactly one string C such that DIB.Dec outputs m. We now turn to security.

Theorem 2. If Π_{LF} is a selective-id secure identity-based lossy trapdoor functions with universal lossy mode, Π_{abo} is a universal identity-based all-but-one trapdoor functions, and $\mathcal{H}_{tcr}$ is universal TCR hash function, if for (t, l)-sources M_0, M_1, and any $\epsilon > 0$ such that $t \geq r_{\mathrm{LF}} + r_{abo} + r_{tcr} + 2\log(\frac{1}{\epsilon})$, then the deterministic identity-based encryption scheme DIBE described above is selective-id PRIV1-secure against chosen ciphertext attacks.

Proof. The proof proceeds via a sequence of games $G_0, G_1, \ldots, G_6$, where game G_0 is the original PRIV1-IND-ID-CCA game, G_6 will be independent of the any underly distribution imposed by the adversary. Then we show that for all $i = 0, \ldots, 5$, game G_i and G_{i+1} are (computationally or statistically) indistinguishable. It follows that the deterministic IBE scheme is PRIV1-IND-ID-CCA secure.

G_1: This game is the same as the game G_0 except that the auxiliary input and label (aux_1, tag^*) of IB-ABO-TDF is replaced by $(aux_1(id^*), h^*)$, where $h^* = H_{tcr}(m^*)$, id^* is the target identity chosen in advance by the adversary, m^* is chosen from (t, l)-source M_b $(b \in \{0, 1\})$ by the simulator, and $aux_1(id^*)$ is generated by an auxiliary input generator $\mathrm{Aux}_1(id^*)$ which takes input an identity in IDSp and returns an auxiliary input associated with the identity.

G_2: This game is identical to game G_1 except that the decryption oracle rejects all the ciphertext $C = h\|c_1\|c_2$ such that $h = h^*$.

G_3: This game is the same as the game G_2, the only change is to decryption oracle, in which if the adversary submits a ciphertext $C = h\|c_1\|c_2$ for decryption, such that (id, h) is a lossy branch, then the decryption oracle immediately outputs reject and halts.

G_4: This game is the same as game G_3 except that the decryption oracle decrypted using the master secret key msk_{abo} of IB-ABO-TDF. That is to say, when the adversary submits a ciphertext $C = h\|c_1\|c_2$ for decryption, the challenger computes $m' \leftarrow \mathrm{Inv}_{abo}(mpk_{abo}, sk_{id}, c_2)$ (Note that the challenger operates the algorithm $\mathrm{KG}_{abo}(msk_{abo}, id)$ and generates sk_{id} for identity id.), and $C' \leftarrow \mathrm{DIB.Enc}(mpk, id, m')$. Then it checks whether $C' = C$. If not, it outputs $\perp$, otherwise outputs m'.

G_5: This game is the same as the game G_4, the only change is to the algorithm DIB.Setup, in which we replace the injective function with a lossy one. Formally, in the algorithm DIB.Setup, we replace $(mpk_{\mathrm{LF}}, msk_{\mathrm{LF}}) \leftarrow \mathrm{LF.Setup}(1^\lambda, aux_0)$ with $(mpk_{\mathrm{LF}}, msk_{\mathrm{LF}}) \leftarrow \mathrm{LF.Setup}(1^\lambda, aux_0(id^*))$, where $aux_0(id^*)$ is generated by an auxiliary input generator $\mathrm{Aux}_0(id^*)$ and id^* is the target identity chosen in advance by the adversary.

G_6: This game is the same as the game G_5 except that the challenge ciphertext is sampled uniformly from the ciphertext space instead of encrypting the message m sampled from the (t, l)-source M_b. Formally, in the challenge phase, the challenger chooses randomly $h^* \stackrel{\$}{\leftarrow} R_{tcr}$, $c_1^* \stackrel{\$}{\leftarrow} R_{\mathrm{LF}}$, $c_2^* \stackrel{\$}{\leftarrow} R_{abo,h^*}$ as the challenge ciphertext $C^* = h_*\|c_1^*\|c_2^*$.

Claim 4. Game G_0 and game G_1 are computationally indistinguishable, given the indistinguishability of the lossy branch of IB-ABO-TDF.

Proof. We prove this claim by describing an IB-ABO-TDF distinguisher algorithm $\mathcal{B}$ that receives mpk_{abo} as input where mpk_{abo} is either (mpk_{abo}, msk_{abo}) $\leftarrow$ Setup$_{abo}$ (1^λ, aux_1, tag^*) or (mpk_{abo}, msk_{abo}) $\leftarrow$ Setup$_{abo}$(1^λ, $aux_1(id^*)$, h^*), where $h^* = H_{tcr}(m^*)$, id^* is the target identity chosen in advance by the adversary and m^* is chosen from (t, l)-source M_b ($b \in \{0, 1\}$) by $\mathcal{B}$.

The distinguisher $\mathcal{B}$ operates by implementing DIB.Setup, DIB.Der, DIB.Dec and challenge. In the DIB.Setup phase, $\mathcal{B}$ runs $(mpk_{\mathrm{LF}}, msk_{\mathrm{LF}}) \leftarrow$ LF.Setup $(1^\lambda, aux_0)$ and chooses $k_{tcr} \xleftarrow{\$} K_{tcr}$. The public key is output as $mpk = (mpk_{\mathrm{LF}}$, $mpk_{abo}, k_{tcr})$. We point out that $\mathcal{B}$ knows the injective trapdoor msk_{LF}, but does not know the trapdoor msk_{abo}. DIB.Der, DIB.Dec are implemented just as game G_0 and game G_1. Note that the only secret information DIB.Der and DIB.Dec need to operate is msk_{LF}, which the distinguisher knows. Likewise, Challenge is implemented just as in all the games. Therefore, any difference in behavior between game G_0 and game G_1 immediately breaks the hardness of distinguishing a lossy branch from an injective branch of the identity-based ABO trapdoor functions collection.

Claim 5. Game G_1 and game G_2 are computationally indistinguishable, given the target collision-resistant property of the hash function $\mathcal{H}_{tcr}$.

Proof. We begin by observing that game G_1 or game G_2 behave equivalently unless an event E happens, which is that the adversary makes a query $C = h\|c_1\|c_2$ to its decryption oracle, where $h = h^*$. We then show that event E happens with negligible probability. There are two possibilities to consider for its decryption. The first is that C is the ciphertext corresponding to m^*. But by the unique encryption property of deterministic encryption that m^* has only one valid ciphertext, namely C^*, which the adversary is not allowed to query to its decryption oracle. So in fact this possibility cannot occur. The second possibility is that C decrypts to some $m \neq m^*$. In this case, we can find a valid target-collision (m, m^*) of hash function $\mathcal{H}_{tcr}$. By the collision-resistant property of $\mathcal{H}_{tcr}$, we conclude that event E happens with negligible probability.

Claim 6. Game G_2 and game G_3 are computationally indistinguishable, given the hardness of finding one-more lossy branch of IB-ABO-TDF.

Proof. Let F be the event that the adversary makes a legal decryption query of the form $C = h\|c_1\|c_2$, such that (id, h) is lossy branch. It is clear that game G_2 and game G_3 proceed identically until the event F happens. We then show that the event F happens with negligible probability.

Note that, if $h = h^*$, the decryption oracle rejects the ciphertext in both games. Therefore, the (id, h) is a new lossy branch of IB-ABO-TDF. By the hardness of finding one-more lossy branch property of IB-ABO-TDF, the event F happens with negligible probability, and hence the claim follows.

Claim 7. Game G_3 and game G_4 are equivalent.

Proof. The only difference between game G_3 and game G_4 is in the implementation of decryption oracle. We show that decryption oracle is equivalent in the two games. In both games, when the challenger receives a legal decryption query of the form $C = h\|c_1\|c_2$ from the adversary. It checks that $c_1 \leftarrow \text{LF.Eval}(mpk_{\text{LF}}, id, x)$, $c_2 \leftarrow \text{Eval}_{abo}(mpk_{abo}, id, h, x)$ for some x that they compute (in different ways), and outputs $\perp$ if not. It suffices to show that such x is unique. Note that, if $h = h^*$ or (id, h) is lossy branch, the decryption oracle outputs reject. Therefore, Π_{LF} and Π_{abo} are both injective, and there is a unique x such that $(c_1, c_2) = (\text{LF.Eval}(mpk_{\text{LF}}, id, x), \text{Eval}_{abo}(mpk_{abo}, id, h, x))$. The both implementations of decryption oracle find the x.

Claim 8. Game G_4 and game G_5 are computationally indistinguishable, given the indistinguishability of the injective and lossy functions of IB-LTDF.

Proof. We prove this claim by describing an IB-LTDF distinguisher $\mathcal{B}$ that receives mpk_{LF} as input where mpk_{LF} was either generated by $\text{LF.Setup}(1^\lambda, aux_0)$ or generated by $\text{LF.Setup}(1^\lambda, aux_0(id^*))$. Note that the distinguisher $\mathcal{B}$ knows the trapdoor msk_{abo} of IB-ABO-TDF, but does not know the trapdoor msk_{LF} corresponding to mpk_{LF}. $\mathcal{B}$ interacts with the adversary as follows.

In the setup phase, $\mathcal{B}$ runs $(mpk_{abo}, msk_{abo}) \leftarrow \text{Setup}_{abo}(1^\lambda, aux_1(id^*), h^* = H_{tcr}(m^*))$, $k_{tcr} \xleftarrow{\$} K_{tcr}$ and outputs public key $mpk = (mpk_{\text{LF}}, mpk_{abo}, k_{tcr})$. When the adversary makes a legal private key query for identity id (i.e., $id \neq id^*$), $\mathcal{B}$ obtains sk_{id} by querying its own private key extracting oracle on the identity id, then forwards to the adversary. When the adversary makes a legal decryption query $C = h\|c_1\|c_2$ for any identity id, $\mathcal{B}$ can compute the private key for identity id using the trapdoor msk_{abo}, then runs the algorithm Inv_{abo} and responds the adversary. Challenge phase is implemented just as in both games.

It is easy to see that the distinguisher $\mathcal{B}$ perfectly simulates game G_4 or game G_5 depending on whether mpk_{LF} results in an injective or lossy function (respectively). By the indistinguishability of injective and lossy functions of IB-LTDF, the claim holds.

Claim 9. Game G_5 and game G_6 are 3ε-close.

Proof. We proceed via two sub-games $G_{5,1}$,$G_{5,2}$. In sub-game $G_{5,1}$, we modify the challenge ciphertext of game G_5 so that $h^* \leftarrow H_{tcr}(m^*)$, $c_1^* \leftarrow \text{LF.Eval}$ $(mpk_{\text{LF}},\ id, m^*)$, $c_2^* \xleftarrow{\$} R_{abo}$. The sub-game $G_{5,2}$ is identical to sub-game $G_{5,1}$ except that $c_1^* \xleftarrow{\$} R_{\text{LF}}$. Blow we will show that game G_5 and sub-game $G_{5,1}$, sub-game $G_{5,1}$ and sub-game $G_{5,2}$, sub-game $G_{5,2}$ and game G_6 are statistically indistinguishable respectively.

In game G_5, let $X = m$ and $Z = h^*\|c_1^*$, then we have $|Z| \leq 2^{r_{tcr}+r_{\text{LF}}}$ due to the universal TCR hash function $\mathcal{H}_{tcr}$ and the IB-LTDF with the lossy mode. By the hypothesis that $t \geq r_{\text{LF}} + r_{abo} + r_{tcr} + 2\log(\frac{1}{\epsilon})$ and Chain Rule, we have $\tilde{H}_\infty(X|Z) = \tilde{H}_\infty(m^*|h^*\|c_1^*) \geq r_{abo} + 2\log(\frac{1}{\epsilon})$, generalized LHL shows that c_2^* is ε-close to uniform on the range of Π_{abo} given $h^*\|c_1^*$. That is to say, G_5 and sub-game $G_{5,1}$ are ε-close. Similarly, in sub-game $G_{5,1}$, we take $X = m$ and $Z = h^*$. According to the Chain Rule and the Generalized LHL, we can show that c_1^* is

ε-close to uniform on the range R_{LF}. In sub-game $G_{5,2}$, we take $X = m$, and according to the standard LHL (i.e., the Generalized LHL with empty Z), we can conclude that h^* is ε-close to uniform on its range as well. The claim holds.

Claim 10. In game G_6, adversary has no advantage to win the game.

Proof. Obviously, when executed in game G_6, h^*, c_1^* and c_2^* are chosen uniformly and independent of all other variables, including b. It is easy to see that the adversary has no advantage in the game. This claim follows.

5 Conclusions

In this paper, we introduced a notion of identity-based all-but-one trapdoor functions, which is an extension of all-but-one trapdoor functions in the identity-based setting. Based on the Bellare et al.'s identity-based lossy trapdoor functions [Bellare et al. 2012], we gave a concrete construction of IB-ABO-TDF and proved its security under DLIN assumption. Based on an IB-LTDF and our IB-ABO-TDF, we proposed a CCA-secure deterministic IBE scheme in the selective-id attack model. A future direction is to construct CCA-secure DIBE scheme in the adaptive case.

Acknowledgements

This work is supported by the National Natural Science Foundation of China (61802241, 61572303, 61772326, 61402015, 61802242), National Key R and D Program of China (No.2017YFB0802000), the National Cryptography Development Fund during the 13th Five-year Plan Period (MMJJ20180217), the Foundation of State Key Laboratory of Information Security (2017-MS-03), the Fundamental Research Funds for the Central Universities (GK201702004), the Natural Science Basic Research plan in Shannxi Province of China (2017JM6048), the project of science and technology in Baoji City (15RKX-1-5-8), Key project of Baoji University of Arts and Sciences (ZK15027).

References

[Bellare et al. 2007] Bellare, M., Boldyreva, A., ONeill, A.: " Deterministic and Efficiently Searchable Encryption"; In: Advances in Cryptology-CRYPTO 2007. Lect. Notes in Comp. Sci., vol 4622. Springer, Berlin, 2007, 535-552.

[Bellare et al. 2009] Bellare, M., Brakerski, Z., Naor, M., Ristenpart, T., Segev, G., Shacham, H., Yilek, S.: " Hedged Public-Key Encryption: How to Protect against Bad Randomness"; In: Advances in Cryptology-ASIACRYPT 2009. Lect. Notes in Comp. Sci., vol 5912. Springer, Berlin, 2009, 232-249.

[Bellare et al. 2015a] Bellare, M., Dowsley, R., Keelveedhi, S.: " How Secure is Deterministic Encryption?"; In: Katz J. (eds) Public-Key Cryptography-PKC 2015. Lect. Notes in Comp. Sci., vol 9020. Springer, Berlin, 2015, 52-73.

[Bellare et al. 2008] Bellare, M., Fischlin, M., ONeill, A., Ristenpart, T.: " Deterministic Encryption: Definitional Equivalences and Constructions without Random Oracles"; In: Advances in Cryptology-CRYPTO 2008. Lect. Notes in Comp. Sci., vol 5157. Springer, Berlin, 2008, 360-378.

[Bellare et al. 2015b] Bellare, M., Hoang, V. T.: " Resisting Randomness Subversion: Fast Deterministic and Hedged Public-Key Encryption in the Standard Model"; In: Advances in Cryptology-EUROCRYPT 2015. Lect. Notes in Comp. Sci., vol 9057. Springer, Berlin, 2015, 627-656.

[Bellare et al. 2012] Bellare, M., Kiltz, E., Peikert, C., Waters, B.: " Identity-Based (Lossy) Trapdoor Functions and Applications"; In: Advances in Cryptology-EUROCRYPT 2012. Lect. Notes in Comp. Sci., vol 7237. Springer, Berlin, 2012, 228-245.

[Bogdanov et al. 2016] Bogdanov, A., Guo, S., Masny, D., Richelson, S., Rosen, A.: " On the hardness of learning with rounding over small modulus"; In Theory of Cryptography, Springer, Berlin, 2016, 209-224.

[Boldyreva et al. 2008] Boldyreva, A., Fehr, S., ONeill, A.: " On Notions of Security for Deterministic Encryption, and Efficient Constructions without Random Oracles"; In: Advances in Cryptology-CRYPTO 2008. Lect. Notes in Comp. Sci., vol 5157. Springer, Berlin, 2008, 335-359.

[Boldyreva et al. 2017] Boldyreva, A., Patton, C., Shrimpton, T.: " Hedging Public-Key Encryption in the Real World"; In: Advances in Cryptology-CRYPTO 2017. Lect. Notes in Comp. Sci., vol 10403. Springer, Cham, 2017, 462-494.

[Brakerski et al. 2011] Brakerski, Z., Segev, G.: " Better Security for Deterministic Public-Key Encryption: The Auxiliary-Input Setting"; In: Advances in Cryptology-CRYPTO 2011. Lect. Notes in Comp. Sci., vol 6841. Springer, Berlin, 2011, 543-560.

[Canetti et al. 2004] Canetti, R., Halevi, S., Katz, J.: " Chosen-Ciphertext Security from Identity-Based Encryption"; In: Advances in Cryptology-EUROCRYPT 2004. Lect. Notes in Comp. Sci., vol 3027. Springer, Berlin, 2004, 207-222.

[Cui et al. 2014] Cui, Y., Morozov, K., Kobara, K., et al.: " Efficient Constructions of Deterministic Encryption from Hybrid Encryption and Code-Based PKE"; International Journal of Network Security, 16(1), 2014, 19-28.

[Dodis et al. 2004] Dodis, Y., Reyzin, L., Smith, A.: " Fuzzy Extractors: How to Generate Strong Keys from Biometrics and Other Noisy Data"; In: Advances in Cryptology-EUROCRYPT 2004. Lect. Notes in Comp. Sci., vol 3027. Springer, Berlin, 2004, 523-540.

[Escala et al. 2014] Escala, A., Herranz, J., Liber,t B., Rfols, C.: " Identity-Based Lossy Trapdoor Functions: New Definitions, Hierarchical Extensions, and Implications"; In: Public-Key Cryptography-PKC 2014. Lect. Notes in Comp. Sci., vol 8383. Springer, Berlin, 2014, 239-256.

[Fang et al. 2016] Fang, F., Li, B., Lu., Liu, Y., Jia, D., Xue, H.: " (Deterministic) Hierarchical identity-based encryption from learning with rounding over small modulus"; In: Proceedings of the 11th ACM on Asia Conference on Computer and Communications Security, 2016, 907-912.

[Freeman et al. 2010] Freeman, D.M., Goldreich, O., Kiltz, E., Rosen, A., Segev, G.: " More constructions of lossy and correlation-secure trapdoor functions"; In: Nguyen, P.Q., Pointcheval, D. (eds.) PKC 2010. LNCS, vol. 6056, 2010, 279-295.

[Fuller et al. 2012] Fuller, B., O'Neill, A., Reyzin, L.: " A Unified Approach to Deterministic Encryption: New Constructions and a Connection to Computational Entropy"; In: Theory of Cryptography. Lect. Notes in Comp. Sci., vol 7194. Springer, Berlin, 2012, 582-599.

[Huang et al. 2018] Huang, Z., Lai, J., Chen, W., Au, M. H., Peng, Z., Li, J.: " Hedged Nonce-Based Public-Key Encryption: Adaptive Security Under Randomness Failures"; In: Public-Key Cryptography-PKC 2018. Lect. Notes in Comp. Sci., vol 10769. Springer, Cham. 2018, 253-279.

[Koppula et al. 2016] Koppula, V., Pandey, O., Rouselakis, Y., et al.: " Deterministic Public-Key Encryption Under Continual Leakage"; In: Applied Cryptography and Network Security 2016. Springer International Publishing, LNCS 9696, 2016, 304-323.

[Li et al. 2019] Li, Y., Yu, Y., Susilo, W., Min, G., Ni, J., Choo, R.: " Fuzzy Identity-Based Data Integrity Auditing for Reliable Cloud Storage Systems"; IEEE Trans. on Dependable and Secure Computing, 16(1), 2019, 72-83.

[Mironov et al. 2012] Mironov, I., Pandey, O., Reingold, O., Segev, G.: " Incremental Deterministic Public-Key Encryption"; In: Pointcheval D., Johansson T. (eds) Advances in Cryptology-EUROCRYPT 2012. Lect. Notes in Comp. Sci., vol 7237. Springer, Berlin, 2012, 628-644.

[O'Neill 2010] O'Neill, A.: " Deterministic Public-Key Encryption Revisited"; Online available from http:// eprint.iacr.org/2010/533.

[Peikert et al. 2008] Peikert, C., Waters, B.: " Lossy trapdoor functions and their applications"; In: Proceedings of the fortieth annual ACM symposium on Theory of computing, STOC '08, 2008, 187-196.

[Raghunathan et al.2013] Raghunathan, A., Segev, G., Vadhan, S.: " Deterministic Public-Key Encryption for Adaptively Chosen Plaintext Distributions"; In: Advances in Cryptology-EUROCRYPT 2013. Lect. Notes in Comp. Sci., vol 7881. Springer, Berlin, 2013, 93-110.

[Wee 2012] Wee, H.: " Dual Projective Hashing and Its Applications Lossy Trapdoor Functions and More"; In: Advances in Cryptology-EUROCRYPT 2012. Lect. Notes in Comp. Sci., vol 7237. Springer, Berlin, 2012, 246-262.

[Xie et al. 2012] Xie, X., Xue, R., Zhang, R.: " Deterministic Public Key Encryption and Identity-Based Encryption from Lattices in the Auxiliary-Input Setting"; In: Security and Cryptography for Networks. SCN 2012. Lect. Notes in Comp. Sci., vol 7485. Springer, Berlin, 2012, 1-18.

[Yamada et al. 2017] Yamada, S.: " Asymptotically compact adaptively secure lattice ibes and verifiable random functions via generalized partitioning techniques"; Cryptology ePrint Archive, 2017 Report 2017/096.

[Yu et al. 2017] Yu, Y., Au, M. H., Ateniese, G., Huang, X., Susilo, W., Dai, Y., Min, G.: " Identity-Based Remote Data Integrity Checking with Perfect Data Privacy Preserving for Cloud Storage"; IEEE Trans. Information Forensics and Security, 12(4), 2017, 767-778.

[Zhang et al. 2014] Zhang, Z., Chen, Y., Chow, S. S. M., Hanaoka, G., Cao, Z., Zhao, Y.: " All-but-One Dual Projective Hashing and Its Applications"; In: Applied Cryptography and Network Security. ACNS 2014. Lect. Notes in Comp. Sci., vol 8479. Springer, Cham, 2014, 181-198.

[Zhang et al. 2017] Zhang, D., Fang, F., Li, B., Wang, X.: " Deterministic Identity-Based Encryption from Lattices with More Compact Public Parameters"; In: Advances in Information and Computer Security. IWSEC 2017. Lect. Notes in Comp. Sci., vol 10418. Springer, Cham, 2017, 215-230.