CCA-Secure IB-KEM from Identity-Based Extractable Hash Proof Systems

Yu Chen[1], Zongyang Zhang[2], Dongdai Lin[1], Zhenfu Cao[2]

[1]State Key Laboratory of Information Security (SKLOIS),
Institute of Information Engineering, Chinese Academy of Sciences, China
{chenyu,ddlin}@iie.ac.cn
[2]Department of Computer Science and Engineering,
Shanghai Jiao Tong University, China
{zongyangzhang,zhenfucao}@sjtu.edu.cn

Abstract. In this paper, we introduce a general paradigm called identity-based extractable hash proof system (IB-EHPS), which is an extension of extractable hash proof system (EHPS) proposed by Wee (CRYPTO '10). We show how to construct identity-based key encapsulation mechanism (IB-KEM) from IB-EHPS in a simple and modular fashion. Our construction provides a generic method of building and interpreting CCA-secure IB-KEMs based on computational assumptions. As instantiations, we realize IB-EHPS from the bilinear Diffie-Hellman assumption and the modified bilinear Diffie-Hellman assumption, respectively.

Key words: identity-based extractable hash proof, identity-based key encapsulation mechanism, CCA security, BDH assumption

1 Introduction

Security against adaptive chosen-ciphertext attack (CCA-security) [RS91] is now accepted as the standard security notion for public-key encryption (PKE) schemes as well as identity-based encryption (IBE) schemes. In contrast to security against adaptive chosen-plaintext attack (CPA-security) [MRS88], CCA-security captures the immunity against an active adversary who is given access to a decryption oracle that allows it to obtain the decryptions of ciphertexts of its choice. Instead of providing the full functionality of PKE/IBE, in many applications it is sufficient to allow sender and receiver to agree on a common random session key. This can be accomplished by *(identity-based) key encapsulation mechanism* as formalized in [BFMLS08]. It is also well-known that a CCA-secure KEM/IB-KEM combined with a CCA-secure data encapsulation mechanism (DEM) yields a full-fledged CCA-secure PKE/IBE scheme. Considering the above reasons, the research community pay more attention to the contruction of CCA-secure KEM/IB-KEM.

On the other hand, in most cases related to cryptography, decisional assumptions form a much stronger class of assumptions than the corresponding search (computational) assumptions[1]. For instance, deciding if a given integer has a modular square root or not may be much easier than actually computing a square root (or, equivalently, factoring the modulus); in groups equipped with efficient pairings, the decisional Diffie-Hellman (DDH) problem is easy, but the computational Diffie-Hellman (CDH) problem still appears to be hard. As such, cryptosystems based on computational assumptions are generally preferred to those based on decisional assumptions. From now on, we will use the term *computational* and *search* interchangeably.

Up to now, only a handful of IB-KEMs [HJKS10, Gal10, CCZ11] were known to be CCA-secure based on computational assumptions in the standard model. Besides, there seems no overarching concept explaining these constructions. Inspired by the notion of extractable hash

[1] Unless the decisional assumption can be proved equivalent to its computational counterpart, as it is the case with cryptosystems based on the problem of "leaning with error" (LWE) [PW08].

proof system [Wee10] in the public key setting, we introduce a new notion named identity-based extractable hash proof system and show how to use it to construct CCA-secure IB-KEMs based on computational assumptions.

1.1 Background

The concept of identity-based encryption (IBE) was first introduced by Shamir [Sha84] in 1984. IBE can be viewed as a special type of PKE in which the public key of a user can be publicly derived from arbitrary strings such as an email address or any other user identifier. This distinguishing characteristic minimizes the need to distribute public key certificates — which is one of the main technical difficulties when implementing public-key infrastructure.

Boneh and Franklin [BF01] defined formal security notions for IBE and designed the first practical IBE scheme based on the computational bilinear Diffie-Hellman (CBDH) assumption. Cocks [Coc01] proposed an IBE scheme based on the decisional quadratic residues (DQR) assumption. Sakai and Kasahara [SK03] presented another IBE scheme based on the q bilinear Diffie-Hellman inversion (q-BDHI) assumption. All of them are proven secure in the random oracle model [BR95]. However, a proof in the random oracle model can only serve as a heuristic argument and possibly lead to insecure schemes in the standard model [CGH98]. This posed an interesting problem of constructing IB-KEMs in the standard model.

First, Canetti, Halevi, and Katz [CHK04] made the breakthrough by giving a solution in the standard model, but under a weaker notion named "selective-identity" where the attacker must declare the target identity id^* before seeing mpk. Boneh and Boyen [BB04a] then provided two efficient selective-identity CPA-secure IBE schemes known as BB_1-IBE and BB_2-IBE. The former is based on the decisional bilinear Diffie-Hellman (DBDH) assumption while the latter is based the decisional q-BDHI assumption. Subsequently, Boneh and Boyen [BB04b] put forwarded a coding-theoretic extension to BB_1-IBE, which is adaptive-identity CPA-secure in the standard model. However, their scheme serves mainly as a proof of theoretical feasibility rather than practical utility due to its low efficiency. Waters [Wat05] then created an efficient and adaptive-identity CPA-secure IBE scheme (Waters-IBE) also based on the DBDH assumption in the standard model by employing Waters hash in place of Boneh-Boyen hash in BB_1-IBE. One drawback is that it suffers from large public parameter size. Gentry [Gen06] proposed an IBE scheme (Gentry-IBE) which enjoys short public parameters and tight security reduction without random oracles. Although Gentry's IBE achieves adaptive-identity CCA-security in the standard model, it did so at the cost of relying on a non-standard and non-static assumption called the decisional q-ABHDE assumption. Waters [Wat09] then introduced dual system encryption methodology and proposed an adaptive-identity CPA-secure IBE scheme based on the DBDH assumption and the decisional Linear (DLIN) assumption in the standard model. It is worth to note that, except Cocks' IBE [Coc01], all the aforementioned IBE schemes use pairing as a primitive. We refer to them as pairing-based IBE. Recently, Gentry *et al.* [GPV08] proposed an IBE scheme based on the LWE assumption in the random oracle model. Cash *et al.* [CHKP10] and Agrawal *et al.* [ABB10] showed how to construct IBE schemes based on the LWE assumption in the standard model.

As stated before, CCA-security is the *de facto* level of security required for IBE used in practice. In the random oracle model, achieving CCA-security is relatively easy. One can apply generic CPA-to-CCA transformation (e.g. Fujisaki-Okamoto transformation [FO99]) to a CPA-secure IBE scheme. The CCA-secure version of Boneh-Franklin IBE [BF03] and Sakai-Kasahara IBE [CC05] are exactly obtained in this way. However, constructing CCA-secure IBE schemes in the standard model turns out to be difficult. Boneh, Canetti, Halevi, and Katz [BCHK07] proposed a generic method (known as the BCHK transformation) from any CPA-secure 2-level

HIBE scheme to a CCA-secure IBE scheme, which is the only generic approach known for constructing efficient CCA-secure IBE in the standard model.

1.2 Motivation

As we have already stated, a decisional assumption is generally stronger than its computational counterpart. From both theoretical and practical perspective, it is more desirable to reduce the security of cryptographic schemes to computational assumptions. Considering an IBE scheme obtained from the BCHK transformation, its CCA-security relies on the CPA-security of the underlying 2-level HIBE scheme and the security of one-time signature or MAC. Hence its assumption cannot be directly counted as computational or decisional assumption. However, the indistinguishability against CPA-attack is of decisional flavor, thus it is arguably closer to decisional assumptions.

Haralambiev *et al.* [HJKS10] proposed several efficient CCA-secure KEMs in the standard model. They also sketched that one of their KEMs can be extended to an IB-KEM. Galindo [Gal10] gave an IB-KEM from the KEM due to Hanaoka and Kurosawa [HK08]. Chen *et al.* [CCZ11] proposed another IB-KEM. Based on their basic 1-bit IB-KEM, they constructed two generalized n-bit IB-KEMs, which compare favorably to [HJKS10] and [Gal10]. The aforementioned IB-KEMs are proven to be selective-identity CCA-secure based on the CBDH assumption in the standard model. All of them fall outside of the BCHK [BCHK07] methodology. While the IB-KEMs due to [HJKS10] and [CCZ11] are similar, it seems that the IB-KEM [Gal10] relies on different techniques to achieve CCA-security. So far, there is no overarching framework explaining these constructions.

Recently, several CCA-secure KEMs from various computational assumptions emerged, such as [CKS08, HK08, HK09, HJKS10]. Inspired in part by hash proof system (HPS) [CS02], Wee [Wee10] introduced the notion of extractable hash proof system (EHPS) and showed how to derive efficient CCA-secure PKE via EHPS. Roughly speaking, EHPS resembles hash proof system (HPS) [CS02] in that both of them are essentially a special kind of non-interactive zero-knowledge proof (designated-verifier NIZK), except that EHPS replaces the soundness requirement with a *proof of knowledge property* [RS91]. The framework of EHPS does not only encompass a series of CCA-secure PKE schemes [BMW05, CHK04, Kil06, Kil07] based on decisional assumptions, but also can explain a serie of CCA-secure PKE schemes [HK09, HJKS10] based on computational assumptions in a unified way, which is the most appealing advantage of EHPS.

Although the realm of IBE and PKE are inherently different, the techniques are sometimes interchangeable. Motivated by the above discussion, we find the following intriguing question:

> *Does there exist a general framework for the construction of identity-based encryption from computational assumptions in the standard model?*

1.3 Our Contributions

EHPSs and their benefits are confined to the realm of public-key setting. In this paper we bring them to the identity-based setting, defining identity-based extractable hash proof system (IB-EHPS). Using IB-EHPS, we obtain new insights into the construction of CCA-secure IB-KEMs. In particular, we show that this notion unifies many seemingly unrelated IB-KEMs based on computational assumptions under a single framework. We summarize our main contributions as follows.

Identity-Based Extractable Hash Proof System. We introduce the notion of IB-EHPS by tailoring EHPS to the identity-based setting. We show that IB-EHPS instantly yields adaptive-identity CPA-secure IB-KEM. However, the basic IB-EHPS is too generic to encompass more applications. To resolve this problem, we further propose the notion of all-but-one (ABO) IB-EHPS, which can in turn be used to construct adaptive-identity CCA-secure IB-KEM. Specially, we investigate the relation between IB-EHPS and IB-HPS/EHPS, which indicates IB-EHPS is not a straightforward extension of previous work. We also put forward the notion of selective ABO IB-EHPS, which can be viewed as a special case of ABO IB-EHPS. We show that selective ABO IB-EHPS turns out to be a useful paradigm for constructing selective-identity CCA-secure IB-KEM.

Practical CCA-secure IB-KEM from IB-EHPS. We present two instantiations of ABO IB-EHPS from the CBDH assumption and the modified CBDH assumption, respectively. As a result, we obtain two efficient adaptive-identity CCA-secure IB-KEMs based on computational assumptions in the standard model. One is a variant of [KG06], while the other is a variant of [KV08]. We also carefully review all the known selective-identity CCA-secure IB-KEMs [HJKS10, Gal10, CCZ11] based on computational assumptions and figure out their relations, which are not clear prior to this work. We find that the instantiation of selective ABO IB-EHPS from the BDH relation serves as a clarification and unification of all these constructions. It is also worth to note that ABO IB-EHPS also encompasses a series of CCA-secure IBE schemes in [KG06, KV08] whose security are based on decisional assumptions.

As an independent of interest, we show the KEM [HK08] due to Hanaoka and Kurosawa can be greatly simplified by resorting to a slightly stronger assumption. This observation not only clarifies the relations between the KEM [HK08] and other CCA-secure KEMs [HJKS10, Wee10], but also leads to a significant simplification of the IB-KEM [Gal10].

1.4 Organization

In the following section, we provide the definitions and all related cryptographic notions. In Section 3 we propose the notion of (all-but-one) IB-EHPS. In particular, we elaborate the differences between IB-EHPS and IB-HPS/EHPS in Section 3.5 and Section 3.6. In Section 4 we present a generic construction of CCA-secure IB-KEM from IB-EHPS. In Section 5 we instantiate two IB-EHPSs from the CBDH assumption and the modified CBDH assumption, then show two adaptive-identity CCA-secure IB-KEMs from them. In Section 6 we propose the notion of selective IB-EHPS, and show its application in Section 6.3. Appendix A recalls all the known CCA-secure IB-KEMs from computational assumptions in the standard model, then investigates their relations carefully. Appendix B gives an intensive observation on the relations between the KEM [HK08] and other KEMs [HJKS10, Wee10].

2 Preliminaries

2.1 Definitions

For a finite set X, we use $x \xleftarrow{R} X$ to denote that x is sampled from X uniformly at random. The main security parameter through this paper is κ, and all algorithms are implicitly given κ as input. We use standard asymptotic notation O and o to denote the growth of functions. Let $\mathsf{poly}(\kappa)$ denote an unspecified function $f(\kappa) = O(\kappa^c)$ for some constant c. Let $\mathsf{negl}(\kappa)$ denote an unspecified function $f(\kappa)$ such that $f = o(\kappa^{-c})$ for every constant c. A probability parametrized by κ is said to be overwhelming if it is $1 - \mathsf{negl}(\kappa)$, and said to be noticeable if it is $1/\mathsf{poly}(\kappa)$. A probabilistic polynomial-time (PPT) algorithm is a randomized algorithm that runs in time $\mathsf{poly}(\kappa)$. If $\mathcal{A}$ is a randomized algorithm, we write $z \leftarrow \mathcal{A}(x_1, \ldots, x_n; r)$ to indicate that $\mathcal{A}$ outputs z on inputs $(x_1, \ldots, x_n)$ and random coins r. We will omit r and write $z \leftarrow \mathcal{A}(x_1, \ldots, x_n)$ when it is not necessary to make explicit the randomness $\mathcal{A}$ uses. We assume that an algorithm returns $\perp$ if any of its inputs is $\perp$.

2.2 Identity-Based Key Encapsulation Mechanism

An IB-KEM consists of four PPT algorithms as follows:

- $\mathsf{Setup}(\kappa)$: take as input a security parameter κ, output the master public key mpk and the master secret key msk. mpk may be used as an implicit input by algorithms KeyGen, Encap, Decap. Let I be the identity space, C be the ciphertext space, and K be the DEM key space.
- $\mathsf{KeyGen}(msk, id)$: take as input msk and an identity $id \in I$, output a private key sk_{id} of id.
- $\mathsf{Encap}(id)$: take as input mpk and an identity $id \in I$, output a ciphertext c and a DEM key $k \in K$.
- $\mathsf{Decap}(sk_{id}, c)$: take as input a private key sk_{id} for identity id and a ciphertext $c \in C$, output a DEM key $k \in K$ or a special reject symbol $\perp$ (which is not in K) indicating that c is not consistent under id.

Definition 2.1 (Correctness) *For correctness, we require that for all $\kappa \in \mathbb{N}$, all identities $id \in I$, and all $(c, k) \leftarrow \mathsf{Encap}(mpk, id)$, $\mathsf{Decap}(\mathsf{KeyGen}(msk, id), c) = k$ holds overwhelmingly, where the probability is taken over the choice of $(mpk, msk) \leftarrow \mathsf{Setup}(\kappa)$, and the random coins of all the algorithms in the expression above.*

Definition 2.2 (Consistency) *A ciphertext c is said to be consistent (or well-formed or valid) under identity id if $c \in C_{id}$, where C_{id} the set of all possible first output of $\mathsf{Encap}(mpk, id)$.*

Definition 2.3 (Verifiability) *There are two flavors of verifiability for IB-KEM. The public verifiability means that anyone can do the "consistency check", i.e., for any identity $id \in I$ and ciphertext $c \in C$, there exists a PPT algorithm Verify which can judge id $c \in C_{id}$. The private verifiability means that only the private key owner can do the corresponding "consistency check", i.e., for any identity $id \in I$ and ciphertext $c \in C$, there exists a PPT algorithm Verify' which can judge id $c \in C_{id}$ by taking sk_{id} as an additional input.*

Chosen-Ciphertext Security. The adaptive chosen-ciphertext security (CCA-security) for IB-KEM is defined by the following game between an adversary $\mathcal{A}$ and a challenger $\mathcal{CH}$.
Setup: $\mathcal{CH}$ runs $\mathsf{Setup}(\kappa)$ to generate (mpk, msk). It gives mpk to $\mathcal{A}$ and keeps msk to itself.
Phase 1: $\mathcal{A}$ may adaptively make polynomially-many the following two types queries:

- Private key queries $\langle id \rangle$: $\mathcal{CH}$ responds with $sk_{id} \leftarrow \mathsf{KeyGen}(msk, id)$.
- Decapsulation queries $\langle id, c \rangle$: $\mathcal{CH}$ first extracts $sk_{id} \leftarrow \mathsf{KeyGen}(msk, id)$ and then responds with $\mathsf{Decap}(sk_{id}, c)$.

Challenge: Once $\mathcal{A}$ decides that Phase 1 is over it outputs the target identity id^* on which it wishes to be challenged. The only constraint is that id^* did not appear in any private key query in Phase 1. $\mathcal{CH}$ computes $(c^*, k_0^*) \leftarrow \mathsf{Encap}(mpk, id^*)$, and samples $k_1^* \xleftarrow{R} K$. Finally, $\mathcal{CH}$ picks $\beta \xleftarrow{R} \{0, 1\}$, and sends (c^*, k_β^*) as the challenge to $\mathcal{A}$.
Phase 2: $\mathcal{A}$ issues more private key queries and decapsulation queries:

- Private key queries $\langle id \rangle$: $\mathcal{CH}$ responds as in Phase 1. The query $\langle id^* \rangle$ is not allowed.
- Decapsulation queries $\langle id, c \rangle$: $\mathcal{CH}$ responds as in Phase 1. The query $\langle id^*, c^* \rangle$ is not allowed.

Guess: Finally, $\mathcal{A}$ outputs a guess $\beta' \in \{0, 1\}$ and wins if $\beta = \beta'$.

We refer to such an adversary $\mathcal{A}$ as an IND-ID-CCA adversary. We define adversary $\mathcal{A}$'s advantage over an IB-KEM by $\mathsf{Adv}_{\mathcal{A}}^{\mathrm{CCA}}(\kappa) = \left|\Pr[\beta = \beta'] - \frac{1}{2}\right|$. The probability is over the random coins used by $\mathcal{A}$ and $\mathcal{CH}$.

Definition 2.4 *An IB-KEM is said to be IND-ID-CCA secure if for any PPT IND-ID-CCA adversary $\mathcal{A}$ its advantage $\mathsf{Adv}_{\mathcal{A}}^{\mathrm{CCA}}(\kappa)$ is negligible in κ.*

The above CCA-security definition is given with respect to "adaptive-identity" attack. The CCA-security with respect to "selective-identity" [CHK04] can be defined similarly, except that $\mathcal{A}$ has to declare the target identity id^* in advance, even before seeing mpk.

Chosen-Plaintext Security. The adaptive chosen-plaintext security (CPA-security) for IB-KEM is defined as in the CCA-security game, except that the adversary is not allowed to issue decapsulation queries.

2.3 Target Collision Resistant Hash Function

Let X and Y be finite, non-empty sets, and ℓ be a non-negative integer. $\mathsf{TCR} = (\mathsf{TCR}_k)_{k\in\{0,1\}^\ell}$ is a family of keyed hash functions. For each ℓ-bits key k, TCR_k is a hash function from X into Y. The target collision resistant (TCR) security is captured by defining the tcr-advantage of an adversary $\mathcal{A}$ as:

$$\mathsf{Adv}_{\mathcal{A}}^{\mathrm{TCR}}(\ell) = \Pr[\mathsf{TCR}_k(x) = \mathsf{TCR}_k(x^*) \ \wedge \ x \neq x^* : k \xleftarrow{R} \{0,1\}^\ell; x^* \xleftarrow{R} X; x \leftarrow \mathcal{A}(k, x^*)]$$

The above notion of TCR [CS03, KG06] is slightly different from the conventional TCR hash function (also known as the universal one-way hashing [NY89, BR97]), where in the security experiment of the latter the target value x^* is chosen by the adversary (but before seeing the hash key k). Please refer to [KG06, Section 2.3] for more details about the implementations of this type of TCR hash functions. To simplify notation we will drop the superscript k and simply use TCR hereafter.

2.4 Diffie-Hellman Assumption

Let $\mathsf{GroupGen}(\kappa)$ be a PPT algorithm that takes as input a security parameter κ and outputs $(p, \mathbb{G})$, where p is a κ-bit prime, and $\mathbb{G}$ is a group of order p. Let g be a generator of $\mathbb{G}$. Define the Diffie-Hellman predicate:

$$\mathrm{dh}(g_1, g_2) := h, \text{ where } g_1 = g^a, g_2 = g^b, a, b, \in \mathbb{Z}_p \text{ and } h = g^{ab}$$

The computational Diffie-Hellman (CDH) assumption with respect to $(p, \mathbb{G}) \leftarrow \mathsf{GroupGen}(\kappa)$ is that $\Pr[\mathcal{A}(g_1, g_2) = \mathrm{dh}(g_1, g_2)] \leq \mathsf{negl}(\kappa)$ for any PPT algorithm $\mathcal{A}$, where the probability is taken over the random choices of g_1 and g_2. Define the DH predicate as:

$$\mathrm{dhp}(g_1, g_2, h) := \mathrm{dh}(g_1, g_2) \stackrel{?}{=} h$$

The Strong DH assumption with respect to $(p, \mathbb{G}) \leftarrow \mathsf{GroupGen}(\kappa)$ is that the CDH assumption still holds even $\mathcal{A}$ can access to a fixed based *decision oracle* for the predicate $\mathrm{dhp}(g_1, \cdot, \cdot)$, which on input $(\hat{g_2}, \hat{h})$, returns $\mathrm{dhp}(g_1, \hat{g_2}, \hat{h})$.

2.5 Bilinear Diffie-Hellman Assumption

Let $\mathsf{BLGroupGen}(\kappa)$ be a PPT algorithm that takes as input a security parameter κ and outputs $(p, \mathbb{G}, \mathbb{G}_T, e)$, where p is a κ-bit prime, $\mathbb{G}$ and $\mathbb{G}_T$ are two groups of order p, and e is a bilinear map from $\mathbb{G} \times \mathbb{G}$ to $\mathbb{G}_T$. Let g be a generator of $\mathbb{G}$. Define the bilinear Diffie-Hellman predicate:

$$\mathrm{bdh}(g_1, g_2, g_3) := h, \text{ where } g_1 = g^a, g_2 = g^b, g_3 = g^c, a, b, c \in \mathbb{Z}_p, \text{ and } h = e(g, g)^{abc}$$

The computational bilinear Diffie-Hellman (CBDH) assumption with respect to $(p, \mathbb{G}, \mathbb{G}_T, e) \leftarrow \mathsf{BLGroupGen}(\kappa)$ is that $\Pr[\mathcal{A}(g_1, g_2, g_3) = \mathrm{bdh}(g_1, g_2, g_3)] \leq \mathsf{negl}(\kappa)$ for any PPT algorithm $\mathcal{A}$, where the probability is taken over the random choices of A, B, C.

In the bilinear setting, the Goldreich-Levin theorem [GL89] gives us the following lemma for Goldreich-Levin hardcore predicate $f_{\mathrm{gl}} : \mathbb{G}_T \times \{0,1\}^u \to \{0,1\}$, where u is an appropriate integer that specifies the size of seed space.

Lemma 2.5 *If the CBDH assumption holds with respect to* $(p, \mathbb{G}, \mathbb{G}_T, e) \leftarrow \mathsf{BLGroupGen}(\kappa)$, *then the distributions* $\Delta_{\mathrm{bdh}} = (g, g_1, g_2, g_3, s, k)$ *and* $\Delta_{\mathrm{rand}} = (g, g_1, g_2, g_3, s, r)$ *are computational indistinguishable, where* $g, g_1, g_2, g_3 \xleftarrow{R} \mathbb{G}$, $s \xleftarrow{R} \{0,1\}^u$, $k \leftarrow f_{\mathrm{gl}}(\mathrm{bdh}(g_1, g_2, g_3), s)$, $r \xleftarrow{R} \{0,1\}$.

The modified computational bilinear Diffie-Hellman (mCBDH) assumption [KV08] is similar to the CBDH assumption except that an additional point $B' = g^{b^2}$ is given to the adversary. We can prove a similar lemma regarding mCBDH assumption as Lemma 2.5.

3 Identity-Based Extractable Hash Proof System

3.1 Binary Relations for Search Problems

A search problem $\mathbf{S} = (S_\kappa)_{\kappa \geq 0}$ is a collection of distributions. For each $\kappa \in \mathbb{N}$, S_κ is a probability distribution over problem instance descriptions. Each instance description Γ specifies:

- Finite non-empty sets X, W.
- A family of binary relations R (indexed by pp) defined over $X \times W$.

We write $\Gamma = (X, W, \mathsf{R})$ to indicate that the instance Γ specifies X, W, and R as above. $\mathbf{S}$ also provides the following two efficient sampling algorithms:

- $\mathsf{SampInst}(\kappa)$: take as input a security parameter κ, output an instance description Γ according to the distribution S_κ, public parameter pp, and secret parameter sp. sp is usually the random coins that used to generate pp.
- $\mathsf{SampR}(pp; r)$: take as input pp, output a tuple $(x, w) \in \mathsf{R}_{pp}$.

Different to the requirement in EHPS [Wee10], we do not require that R can be efficiently verifiable. For binary relations R, we require that with overwhelming probability over pp, for any $x \in X$, there exists at most one $w \in W$ such that $(x, w) \in \mathsf{R}_{pp}$ (we say that w is a *witness* for x). We say R is one-way if:

- there is an efficiently computable function F from W to $\{0,1\}^l$ for some positive integer l such that given pp and x, $\mathsf{F}(w)$ is pseudo-random over $\{0,1\}^l$ where $(x, w) \leftarrow \mathsf{SampR}(pp; r)$. The probability is taken over the random coins used by $\mathsf{SampInst}$ and SampR.

For relations where computing w from x is hard on average, we may derive a function GL with a one-bit output via the Goldreich-Levin hardcore predicate f_{gl}. Note that GL is an instantiation of the above function F.

Next, we consider a search problem from the BDH assumption.

A Search Problem from the BDH Assumption. $\mathsf{SampInst}(\kappa)$ runs $\mathsf{BLGroupGen}(\kappa)$ to generate common public parameter $(p, \mathbb{G}, \mathbb{G}_T, e)$, picks $g \xleftarrow{R} \mathbb{G}$, $a, b \xleftarrow{R} \mathbb{Z}_p$, outputs $pp = (g, g^a, g^b)$, $sp = (a, b)$, and an instance $\Gamma = (X, W, \mathsf{R})$, where $X = \mathbb{G}$, $W = \mathbb{G}_T$, and R is defined as:

$$\mathsf{R}_{pp}^{\mathsf{bdh}} = \left\{(x, w) \in \mathbb{G} \times \mathbb{G}_T : w = e(g, x)^{ab}\right\}$$

The associated SampR picks $r \xleftarrow{R} \mathbb{Z}_p$ and outputs $(g^r, e(g^a, g^b)^r)$. Lemma 2.5 shows that we can extract a single hardcore bit from w using $f_{\mathrm{gl}}(w)$ for relation $\mathsf{R}_{pp}^{\mathsf{bdh}}$.

The modified BDH relation $\mathsf{R}_{pp}^{\mathsf{mbdh}}$ can be defined in an analogous way as the BDH relation.

3.2 Hash Family

To interact with a search problem $\mathbf{S}$, we consider a hash family $\mathbf{H} = (\mathsf{H}, MPK, I, X, Y)$, where MPK, I, X, and Y are finite, non-empty sets, $\mathsf{H} = (\mathsf{H}_{mpk})_{mpk \in MPK}$ is a collection of functions indexed by MPK, so that for every $mpk \in MPK$, H_{mpk} is a function from $I \times X$ into Y.

3.3 The Paradigm of Identity-Based Hash Proof System

An IB-EHPS $\mathbf{P}$ for a search problem $\mathbf{S}$ contains a tuple of algorithms (Setup, KeyGen, Pub, Ext, Setup', KeyGen', Priv). $\mathbf{P}$ can behave in one of two modes, namely the extraction mode and the hashing mode. Loosely speaking,

- In the extraction mode, there is an algorithm Pub that can evaluate $\mathsf{H}_{mpk}(id, x)$ with the knowledge of randomness r that used to sample (x, w). Moreover, for a correctly computed hash value $y = \mathsf{H}_{mpk}(id, x)$, there is an algorithm Ext that can extract the witness w of x by taking a private key sk_{id} for id, x, y as input.
- In the hashing mode, there is an algorithm Priv that can compute $\mathsf{H}_{mpk}(id, x)$ without knowing the randomness used to sample (x, w).

Looking ahead, we rely on the extraction mode for the normal functionality of the resulting IB-KEM, and on the hashing mode for the arguement of security.

Extraction Mode

- $\mathsf{Setup}(\kappa)$: run $\mathsf{SampInst}(\kappa)$ to generate an instance $\Gamma = (X, W, \mathsf{R})$, pp, sp, pick a corresponding hash family $\mathbf{H} = (\mathsf{H}, MPK, I, X, Y)$; output master public/secret key (mpk, msk). Generally, we have $pp \subseteq mpk$ and $msk \subseteq sp$.
- $\mathsf{KeyGen}(msk, id)$: take as input msk and $id \in I$, output a private key sk_{id} for id.
- $\mathsf{Pub}(mpk, id, r)$: take as input mpk, $id \in I$, and randomness r that used to sample (x, w), output $y \in Y$ such that $y = \mathsf{H}_{mpk}(id, x)$. This is the *public evaluation algorithm*.
- $\mathsf{Ext}(sk_{id}, x, y)$: take as input a private key sk_{id} for $id \in I$, $x \in X$ and $y \in Y$, output $w \in W$ such that $(x, w) \in \mathsf{R}_{pp}$ if $y = \mathsf{H}_{mpk}(id, x)$.

Hashing Mode

- $\mathsf{Setup}'(\kappa)$: run $\mathsf{SampInst}(\kappa)$ to generate an instance $\Gamma = (X, W, \mathsf{R})$, pp, sp, pick a corresponding hash family $\mathbf{H} = (\mathsf{H}, MPK, I, X, Y)$; output master public/secret key (mpk, msk') (the generation of the master key pair can be done without the knowledge of sp). msk' implicitly splits set I into two disjoint subsets I_0 and I_1 such that $I = I_0 \cup I_1$ and $I_0 \cap I_1 = \emptyset$.
- $\mathsf{KeyGen}'(msk', id)$: take as input msk' and $id \in I$, if $id \in I_0$ output a private key sk_{id} for id, else output $\perp$.
- $\mathsf{Priv}(msk', id, x)$: take as input msk' and $id \in I$, if $id \in I_1$ output $y \in Y$ such that $y = \mathsf{H}_{mpk}(id, x)$, else output $\perp$. This is the *private evaluation algorithm*.

Well-Partition. We now introduce a property which is very useful in transforming IB-EHPS into CPA-secure IB-KEM. Intuitively, this property guarantees that no PPT adversary $\mathcal{A}$ can distinguish the following two games simulated by $\mathcal{CH}$ operating IB-EHPS in the extraction mode and the hashing mode with noticeable probability.

Game Ext. $\mathcal{CH}$ interacts with $\mathcal{A}$ by operating IB-EHPS in the extraction mode.
Setup: $\mathcal{CH}$ samples $(x^*, w^*) \leftarrow \mathsf{SampR}(r^*)$, runs $\mathsf{Setup}(\kappa)$ to generate (mpk, msk) and gives mpk to $\mathcal{A}$.

Phase 1 - Private key queries: When $\mathcal{A}$ submits a private key query $\langle id \rangle$, $\mathcal{CH}$ responds with $\mathsf{KeyGen}(msk, id)$.
Phase Middle: $\mathcal{A}$ submits $id^* \in I$ which has not been asked for private key in Phase 1. $\mathcal{CH}$ computes $y^* = \mathsf{H}_{mpk}(id^*, x^*)$ by evaluating $\mathsf{Pub}(mpk, id^*, r^*)$, then sets $k_0^* \leftarrow \mathsf{F}(w^*)$ and $k_1^* \xleftarrow{R} \{0,1\}^l$. $\mathcal{CH}$ picks a random bit $\beta \in \{0,1\}$ and returns (x^*, y^*, k_β^*) to $\mathcal{A}$.
Phase 2 - Private key queries: Same as in Phase 1 except that the private key query $\langle id^* \rangle$ is not allowed.

Game Hash. $\mathcal{CH}$ interacts with $\mathcal{A}$ in this game by operating IB-EHPS in the hashing mode.
Setup: $\mathcal{CH}$ samples $(x^*, w^*) \leftarrow \mathsf{SampR}(r^*)$, runs $\mathsf{Setup}'(\kappa)$ to generate (mpk, msk') and gives mpk to $\mathcal{A}$.
Phase 1 - Private key queries: When $\mathcal{A}$ submits a private key query $\langle id \rangle$, $\mathcal{CH}$ responds with $\mathsf{KeyGen}'(msk', id)$.
Phase Middle: $\mathcal{A}$ submits $id^* \in I$ which has not been asked for private key in Phase 1. $\mathcal{CH}$ computes $y^* = \mathsf{H}_{mpk}(id, x^*)$ by evaluating $\mathsf{Priv}(msk', id^*, x^*)$, then sets $k_0^* \leftarrow \mathsf{F}(w^*)$ and $k_1^* \xleftarrow{R} \{0,1\}^l$. $\mathcal{CH}$ picks a random bit $\beta \in \{0,1\}$ and returns (x^*, y^*, k_β^*) to $\mathcal{A}$.
Phase 2 - Private key queries: Same as in Phase 1 except that the private key query $\langle id^* \rangle$ is not allowed.

Let Q_e be the number of private key queries. If for any PPT adversary $\mathcal{A}$, its view in Game Hash is identical to Game Ext with at least a noticeable probability δ, we say such an IB-EHPS is (Q_e, δ)-well-partition.

Remark 1. In the case of IB-EHPS, the output of Ext is unspecified when $y \neq \mathsf{H}_{mpk}(id, x)$. In the hashing mode, msk' usually contains the trapdoor information for $mpk \backslash pp$. Since msk' only allows one to extract private keys for a subset of I, we may regard it as semi-functional master secret key.

3.4 All-But-One Identity-Based Extractable Hash Proof System

For our application, it is convenient to work with a richer abstraction, named all-but-one (ABO) IB-EHPS. ABO IB-EHPS can behave in one of two modes, namely the extraction mode and the ABO hashing mode. More precisely, it contains a tuple of algorithms (Setup, KeyGen, Pub, Ext, Setup', KeyGen', Priv, Ext'). The meaning of the term "all-but-one" is: Priv only works when $x = x^*$.

Extraction Mode

- The algorithms Setup, Pub, and KeyGen related to the extraction mode are identical to that in IB-EHPS.
- $\mathsf{Ext}(sk_{id}, x, y)$: take as input a private key sk_{id} for $id \in I$, $x \in X$ (suppose $(x, w) \in \mathsf{R}_{pp}$) and $y \in Y$, if $y = \mathsf{H}_{mpk}(id, x)$ output $w \in W$, else output a value from $W \cup \perp$ which is independent of w.

ABO Hashing Mode

- $\mathsf{Setup}'(\kappa, x^*)$: similar to Setup' in IB-EHPS except taking an extra input $x^* \in X$.
- $\mathsf{KeyGen}'(msk', id)$: same as KeyGen' in IB-EHPS.
- $\mathsf{Priv}(msk', id, x)$: take as input msk', $id \in I$, and $x \in X$, if $id \in I_1$ and $x = x^*$ output $y \in Y$ such that $y = \mathsf{H}_{mpk}(id, x^*)$, else output $\perp$.

– $\mathsf{Ext}'(msk', id, x, y)$: take as input msk', $id \in I$, $x \in X$ (suppose $(x, w) \in \mathsf{R}_{pp}$), and $y \in Y$, if $y = \mathsf{H}_{mpk}(id, x)$ output $\perp$ if $x = x^* \wedge id \notin I_0$ and w otherwise, else output a value from $W \cup \perp$ which is independent of w.

Well Partition. This property for ABO IB-EHPS is defined analogously as that for IB-EHPS. We formally defined it as below.

Game Ext. $\mathcal{CH}$ interacts with $\mathcal{A}$ by operating ABO IB-EHPS in the extraction mode.
Setup: $\mathcal{CH}$ samples $(x^*, w^*) \leftarrow \mathsf{SampR}(r^*)$, runs $\mathsf{Setup}(\kappa)$ to generate (mpk, msk) and gives mpk to $\mathcal{A}$.
Phase 1 - Private key queries: Same as Game Ext in IB-EHPS.
Phase 1 - Decapsulation queries: When $\mathcal{A}$ submits a query $\langle id, x, y\rangle$, $\mathcal{CH}$ extracts $sk_{id} \leftarrow \mathsf{KeyGen}(msk, id)$ and responds with $\mathsf{F}(\mathsf{Ext}(sk_{id}, x, y))$.
Phase Middle: Same as Game Ext in IB-EHPS.
Phase 2 - Private key queries: Same as Game Ext in IB-EHPS.
Phase 2 - Decapsulation queries: Same as Phase 1 except that the decapsulation query $\langle id^*, x^*, y^*\rangle$ is not allowed.

Game ABO. $\mathcal{CH}$ interacts with $\mathcal{A}$ by operating ABO IB-EHPS in the ABO hashing mode.
Setup: $\mathcal{CH}$ samples $(x^*, w^*) \leftarrow \mathsf{SampR}(r^*)$, runs $\mathsf{Setup}'(\kappa, x^*)$ to generate (mpk, msk') and gives mpk to $\mathcal{A}$.
Phase 1 - Private key queries: Same as Game Hash in IB-EHPS.
Phase 1 - Decapsulation queries: When $\mathcal{A}$ submits a query $\langle id, x, y\rangle$, $\mathcal{CH}$ responds with $\mathsf{F}(\mathsf{Ext}'(msk', id, x, y))$.
Phase Middle: Same as Game Hash in IB-EHPS.
Phase 2 - Private key queries: Same as Game Hash in IB-EHPS.
Phase 2 - Decapsulation queries: Same as in Phase 1 except that the decapsulation query $\langle id^*, x^*, y^*\rangle$ is not allowed.

Let Q_e and Q_d be the number of private key queries and decapsulation queries. If for any PPT adversary $\mathcal{A}$, its view in Game ABO is identical to that in Game Ext with at least a noticeable probability δ, we say such an ABO IB-EHPS is (Q_e, Q_d, δ)-well-partition.

Remark 2. In IB-EHPS, for algorithm Ext we require that:

$$y = \mathsf{H}_{mpk}(id, x) \Longrightarrow (x, \mathsf{Ext}(sk_{id}, x, y)) \in \mathsf{R}_{pp} \tag{1}$$

Combining with the one-wayness of R, such requirement is sufficient to yield CPA-secure IB-KEM (this requirement ensures the correctness, while the one-wayness implies CPA-security). However, in ABO IB-EHPS the same correctness requirement (1) for algorithm Ext is not sufficient to yield CCA-secure IB-KEM. This is because, to achieve CCA-security we have to make sure that the decryption oracle does not help the adversary to distinguish w^* from random. Particularly, the decryption algorithm must reveal no knowledge of w^* corresponding to x^* when the input ciphertext (x^*, y) is not consistent. In line of this intuition, for the purpose of CCA-secure IB-KEM, algorithm Ext of ABO IB-EHPS must also satisfy the following requirement:

$$y \neq \mathsf{H}_{mpk}(id, x) \Longrightarrow \mathsf{Ext}(sk_{id}, x, y)) \text{ is independent of } w \tag{2}$$

Generally, there are two approaches to achieve the above requirement. One is explicit check, that is equipping algorithm Ext with algorithm Verify, which can determine if $y = \mathsf{H}_{mpk}(id, x)$ or not. The other is implicit check, that is designing algorithm Ext smartly using the "implicit rejection" idea [Kil06, KV08], namely when $y \neq \mathsf{H}_{mpk}(id, x)$, $\mathsf{Ext}(sk_{id}, x, y)$ returns a random value from W which is independent of w.

We also note that the requirement for Ext in ABO EHPS [Wee10] is:

$$y = \mathsf{H}_{pk}(x) \Longleftrightarrow (x, \mathsf{Ext}(sk, x, y)) \in \mathsf{R}_{pp} \tag{3}$$

However, this requirement is not sufficient to lead to CCA-security. An counterexample is that Ext returns $f(w)$ when $y \neq \mathsf{H}_{pk}(x)$, where f is an invertible function and $f(w) \neq w$ holds overwhelmingly. Clearly, such Ext satisfies the above requirement but can not ensure CCA-security, since the adversary can easily recover w by issuing a ill-formed decryption query. In fact, all the ABO EHPS constructions presented in [Wee10] ensure that $\mathsf{Ext}(sk, x, y) = \perp$ when $y \neq \mathsf{H}_{pk}(x)$, which satisfies the requirement similar to (2) and strictly stronger than (3).

Remark 3. The well-partition property may also rely on the one-wayness of R.

3.5 Relation to the Identity-Based Hash Proof System

Boneh *et al.* [BGH07], Alwen *et al.* [ADN+10], and Chen *et al.* [CZLC12] generalized the notion of HPS due to Cramer and Shoup [CS02] into the identity-based setting by defining identity-based hash proof system (IB-HPS). IB-HPS turns out to be a useful primitive to construct leakage-resilient IBE schemes.

Similar to the difference between HPS and EHPS, IB-EHPS departs from IB-HPS in that they establish the security of the resulting IB-KEMs in different ways. To argue the real DEM key is indistinguishable from a random one, in IB-EHPS paradigm we directly reduce the indistinguishability to the one-wayness of relation R associated with the underlying search problem, while in IB-HPS paradigm we first prove that the real DEM key is indistinguishable from a fake DEM key encapsulated by invalid ciphertext (relying on the underlying subset membership problem), then prove the fake DEM key is indistinguishable from a random one (relying on the smoothness property).

3.6 Relation to Extractable Hash Proof System

IB-EHPS is the corresponding notion of EHPS in the identity-based setting. However, we stress that the extension is not straightforward for the following main differences.

- In EHPS the hash function H is indexed by public key set PK with a single element $x \in X$ as input, while in IB-EHPS the hash function H is indexed by the master public key set MPK with an element $id \in I$ and an element $x \in X$ as input. Note the transformation from IBE to PKE, identity-based EHPS can be viewed as a generalization of EHPS.
- In EHPS, the first output of Setup and Setup' is explicitly required to be statistically indistinguishable. In IB-EHPS, we essentially relax the indistinguishability to parametrized indistinguishable (may be computational) and embody it in the well-partition property.
- IB-EHPS is introduced as a general framework to encompass IB-KEMs. In line of this, two additional algorithms KeyGen and KeyGen' are included in IB-EHPS.
- The (ABO) hashing mode for (ABO) IB-EHPS is defined in partitioning flavor. More precisely, the algorithm Setup' generates (mpk, msk') and implicitly splits the whole identity space I into two disjoint subspaces, — 1) I_0: identities for which KeyGen' can generate private keys; and 2) I_1: identities for which Priv can evaluate the hash value. We note that (ABO) IB-EHPS inherently relies on the partitioning strategy. Suppose that there is an identity id that belongs to the intersection of I_0 and I_1, then given (pp, x) one can compute the corresponding w such that $(x, w) \in \mathsf{R}_{\mathrm{pp}}$ by itself as follows: first computes $y = \mathsf{H}_{mpk}(id, x)$

via $\mathsf{Priv}(msk', id, x)$, then extracts $sk_{id} \leftarrow \mathsf{KeyGen}(msk', id)$ and uses it to recover w via $\mathsf{Ext}(sk_{id}, x, y)$. This contradicts the one-wayness of R_{pp}. This feature of IB-EHPS makes it particularly well-suited to yield IB-KEM whose provable security follows the partitioning strategy.
- In ABO EHPS, the ABO hashing mode is defined with respect to a tag t^*, which in turn is the hash value of x^* for some target collision resistant (TCR) hash function. In our case, we define the ABO hashing mode directly with respect to x^*. We do so out of two reasons. One is that towards utmost generality for an abstract paradigm, it is more preferable to minimize the dependence on other primitives, while the other is that the proof for the transformation from IB-EHPS to CCA-secure IB-KEM would be more clean and simple. Nevertheless, TCR hash function turns out to be a useful tool when instantiating EHPS/IB-EHPS from concrete number-theoretic assumptions.

4 Generic Constructions from Identity-Based Extractable Hash Proofs

In this section, we present generic constructions of IB-KEM from (ABO) IB-EHPS. As a warm up, we first show the transformation from IB-EHPS to adaptive-identity CPA-secure IB-KEM, then the transformation from ABO IB-EHPS to adaptive-identity CCA-secure IB-KEM. Before going into details, we first give an intuitive explanation of the constructions from IB-EHPS to IB-KEM with respect to the underlying relation. Suppose that the binary relation of an IB-EHPS is R_{pp} and (x, w) is a tuple that belongs to R_{pp}. The overall construction is: first encrypt (or commit to) a fresh DEM key (the corresponding witness is w) which is in turn used to encrypt the actual message, and then provide an identity-based extractable hash proof $y = \mathsf{H}_{mpk}(id, x)$ (which is also zero-knowledge) for it. The overall ciphertext is of the form (x, y). In fact, such an approach was used implicitly in the PKE schemes based on computational assumptions and its connection to the Rackoff-Simon paradigm [RS91] was made explicitly in [Wee10]. Here we make its link to the underlying relation R clear. It is useful to note the distinguished feature in the construction from IB-EHPS to IBE that the value w (used to compute the session key) is uniquely determined by pp and the random coins used by SampR. This explains why IB-EHPS cannot encompass the IBE schemes whose session keys are related to the identity, such as Boneh-Franklin IBE [BF03] and Sakai-Kasahara IBE [SK03].

4.1 Generic Construction of CPA-secure IB-KEM

Starting from an IB-EHPS (Setup, Setup', Pub, Priv, Ext, KeyGen, KeyGen') for a search problem $\mathbf{S}$, we construct an IB-KEM as follows:

- $\mathsf{Setup}(\kappa)$: same as $\mathsf{Setup}(\kappa)$ in IB-EHPS. The identity space is I, the ciphertext space is $X \times W$, and the DEM key space is $\{0,1\}^l$.
- $\mathsf{KeyGen}(msk, id)$: same as $\mathsf{KeyGen}(msk, id)$ in IB-EHPS.
- $\mathsf{Encap}(mpk, id)$: sample $(x, w) \leftarrow \mathsf{SampR}(r)$, compute $y \leftarrow \mathsf{Pub}(mpk, id, r)$, and output a ciphertext $c = (x, y)$ and a DEM key $k \leftarrow \mathsf{F}(w)$,
- $\mathsf{Decap}(sk_{id}, c)$: parse c as (x, y), and output $\mathsf{F}(\mathsf{Ext}(sk_{id}, x, y))$.

The functionality of the above IB-KEM follows readily from the correctness of the extraction mode. For the security, we have the following theorem.

Theorem 4.1 *If R is one-way and the IB-EHPS is (Q_e, δ)-well-partition, then the above IB-KEM is IND-ID-CPA secure.*

Proof. To establish the IND-ID-CPA security based on the one-wayness of R, we proceed via a sequence of games. Let S be the event that $\mathcal{A}$ wins in Game CPA, and S_i be the event that $\mathcal{A}$ wins in Game i.

Game CPA. $\mathcal{CH}$ plays with $\mathcal{A}$ in the following game.
Setup: $\mathcal{CH}$ runs $\mathsf{Setup}(\kappa)$ to generate (mpk, msk), and gives mpk to $\mathcal{A}$.
Phase 1 - Private key queries: When $\mathcal{A}$ submits a private key query $\langle id \rangle$, $\mathcal{CH}$ responds with $\mathsf{KeyGen}(msk, id)$.
Challenge: $\mathcal{A}$ submits a target identity id^* on the condition that id^* has not been asked for private key in Phase 1. $\mathcal{CH}$ samples $(x^*, w^*) \leftarrow \mathsf{SampR}(r^*)$ and computes $y^* = \mathsf{H}_{mpk}(id^*, x^*)$ by evaluating $\mathsf{Pub}(mpk, id^*, r^*)$, sets $k_0^* \leftarrow \mathsf{F}(w^*)$ and $k_1^* \xleftarrow{R} \{0,1\}^l$. $\mathcal{CH}$ then picks a random bit $\beta \in \{0,1\}$ and returns (x^*, y^*, k_β^*) to $\mathcal{A}$.
Phase 2 - Private key queries: Same as in Phase 1 except that the query $\langle id^* \rangle$ is not allowed.
Guess: $\mathcal{A}$ outputs its guess β' for β and wins if $\beta = \beta'$.

It is easy to see that $\mathcal{A}$'s view in Game CPA is identical to the standard IND-ID-CPA game, thus we have:

$$\Pr[S] = \frac{1}{2} + \mathsf{Adv}_{\mathcal{A}}^{\text{CPA}}(\kappa) \tag{4}$$

Game 0. $\mathcal{CH}$ plays with $\mathcal{A}$ in the following game.
Setup: Same as in Game Ext for IB-EHPS.
Phase 1 - Private key queries: Same as Game Ext for IB-EHPS.
Challenge: Same as the Phase Middle in Game Ext for IB-EHPS.
Phase 2 - Private key queries: Same as Game Ext for IB-EHPS.
Guess: $\mathcal{A}$ outputs its guess β' for β and wins if $\beta = \beta'$.

The only difference between Game 0 and Game CPA is that in Game 0 $\mathcal{CH}$ samples (x^*, w^*) at the setup phase while in Game CPA $\mathcal{CH}$ samples (x^*, w^*) at the challenge phase. Note that the sampling operation is independent of phase 1, $\mathcal{A}$'s view in Game 0 is identical to Game CPA. Thus we have:

$$\Pr[S_0] = \Pr[S] \tag{5}$$

We claim that $\mathsf{Adv}_{\mathcal{A}}^{\text{CPA}}$ is negligible in κ based on the one-wayness of R. Suppose there exists an algorithm $\mathcal{A}$ who has non-negligible advantage against the CPA-security of the IB-KEM, then we can construct an adversary $\mathcal{B}$ breaking the pseudo-randomness of F with non-negligible advantage, which is sufficient to prove CPA-security based on the one-wayness of R.

Game 1. $\mathcal{B}$ receives a challenge instance (pp, x^*, k^*) of R, where x^* is picked from the tuple $(x^*, w^*) \in \mathsf{R}_{pp}$ generated by $\mathsf{SampR}(pp, r^*)$ and k^* is either $\mathsf{F}(w^*)$ or randomly picked from $\{0,1\}^l$. $\mathcal{B}$ is asked to determine $k^* = \mathsf{F}(w^*)$ or $k \xleftarrow{R} \{0,1\}^l$. $\mathcal{B}$ plays with $\mathcal{A}$ in the following game.
Setup: $\mathcal{B}$ generates (mpk, msk') from pp as $\mathcal{CH}$ does in Game Hash. $\mathcal{B}$ sends mpk to $\mathcal{A}$.
Phase 1 - Private key queries: $\mathcal{B}$ operates as $\mathcal{CH}$ does in Game Hash.
Challenge: $\mathcal{A}$ submits a target identity id^* on the condition that id^* did not appear in any private key query in Phase 1, $\mathcal{B}$ computes $y^* = \mathsf{H}_{mpk}(id^*, x^*)$ via $\mathsf{Priv}(msk', id^*, x^*)$, then instead of creating the challenge by explicitly generating a random bit β, it sends (x^*, y^*, k^*) to $\mathcal{A}$ as the challenge.
Phase 2 - Private key queries: $\mathcal{B}$ operates as $\mathcal{CH}$ does in Game Hash.
Guess: $\mathcal{A}$ outputs its guess β' for β and $\mathcal{B}$ forwards β' to its own challenger.

Observe that $\mathcal{A}$'s view in Game 1 (resp. Game 0) is identical to Game Hash (resp. Game Ext), and the underlying IB-EHPS is (Q_e, δ)-well-partition, we conclude that $\mathcal{B}$ can break the pseudo-randomness of F with advantage:

$$\mathsf{Adv}_{\mathcal{B}} = \left|(1-\delta)\cdot\frac{1}{2} + \delta\cdot\Pr[S_0] - \frac{1}{2}\right| = \delta\cdot\mathsf{Adv}_{\mathcal{A}}^{\mathrm{CPA}}(\kappa) \tag{6}$$

Since δ is noticeable, $\mathcal{B}$'s advantage is non-negligible in κ, which contradicts to the one-wayness of R. This proves the theorem. □

4.2 Generic Construction of CCA-secure IB-KEM

Starting from an ABO IB-EHPS (Setup, KeyGen, Pub, Ext, Setup', KeyGen' Priv, Ext') for a search problem $\mathbf{S}$, we can construct an IB-KEM (Setup, KeyGen, Encap, Decap) exactly the same way as we did in Section 4.1. The functionality of the above IB-KEM follows readily from the correctness of the extraction mode. For the security, we have the following theorem.

Theorem 4.2 *If R is one-way and the ABO IB-EHPS is (Q_e, Q_d, δ)-well-partition, then the above IB-KEM is IND-ID-CCA secure.*

Proof. To establish the IND-ID-CCA security based on the one-wayness of relation R, we proceed via a sequence of games. Let S be the event that $\mathcal{A}$ wins in Game CCA, and S_i be the event that $\mathcal{A}$ wins in Game i.

Game CCA. $\mathcal{CH}$ plays with $\mathcal{A}$ in the following game.
Setup: $\mathcal{CH}$ runs $\mathsf{Setup}(\kappa)$ to generate (mpk, msk) and gives mpk to $\mathcal{A}$.
Phase 1 - Private key queries: When $\mathcal{A}$ submits a private key query $\langle id\rangle$, $\mathcal{CH}$ responds with $sk_{id} \leftarrow \mathsf{KeyGen}(msk, id)$.
Phase 1 - Decapsulation queries: When $\mathcal{A}$ submits a decapsulation query $\langle id, c = (x, y)\rangle$, $\mathcal{CH}$ extracts $sk_{id} \leftarrow \mathsf{KeyGen}(msk, id)$ and responds with $\mathsf{F}(\mathsf{Ext}(sk_{id}, x, y))$.
Challenge: When $\mathcal{A}$ submits a target identity id^* that did not appear in any private key query in Phase 1, $\mathcal{CH}$ samples $(x^*, w^*) \leftarrow \mathsf{SampR}(r^*)$ and computes $y^* = \mathsf{H}_{mpk}(id^*, x^*)$ via $\mathsf{Pub}(mpk, id^*, r^*)$, sets $c^* = (x^*, y^*)$, then sets $k_0^* \leftarrow \mathsf{F}(w^*)$ and computes $k_1^* \xleftarrow{R} \{0,1\}^l$. $\mathcal{CH}$ picks $\beta \xleftarrow{R} \{0,1\}$ and returns (c^*, k_β^*) to $\mathcal{A}$ as the challenge.
Phase 2 - Private key queries: Same as in Phase 1 except that the query $\langle id^*\rangle$ is not allowed.
Phase 2 - Decapsulation queries: Same as in Phase 1 except that the query $\langle id^*, c^*\rangle$ is not allowed.
Guess: $\mathcal{A}$ outputs its guess β' for β and wins if $\beta' = \beta$.

It is easy to see that $\mathcal{A}$'s view in Game CCA is identical to the standard IND-ID-CCA game for IB-KEM. According to the definition, we have:

$$\Pr[S] = \frac{1}{2} + \mathsf{Adv}_{\mathcal{A}}^{\mathrm{CCA}}(\kappa) \tag{7}$$

Game 0. $\mathcal{CH}$ plays with $\mathcal{A}$ in the following game.
Setup: Same as in Game Ext for ABO IB-EHPS.
Phase 1 - Private key queries: Same as in Game Ext of ABO IB-EHPS.
Phase 1 - Decapsulation queries: Same as in Game Ext of ABO IB-EHPS.
Challenge. Same as the Phase Middle in Game Ext of ABO IB-EHPS.
Phase 2 - Private key queries: Same as in Game Ext of ABO IB-EHPS.
Phase 2 - Decapsulation queries: Same as in Game Ext of ABO IB-EHPS.

Guess: $\mathcal{A}$ outputs its guess β' for β and wins if $\beta = \beta'$.

The only difference between Game 0 and Game CCA is that in Game 0 $\mathcal{CH}$ samples (x^*, w^*) at the setup phase while in Game CCA $\mathcal{CH}$ samples (x^*, w^*) at the challenge phase. It is easy to see that this difference is invisible in $\mathcal{A}$'s view. Thus we have:

$$\Pr[S_0] = \Pr[S] \tag{8}$$

We claim that $\mathsf{Adv}_{\mathcal{A}}^{\mathrm{CCA}}$ is negligible in κ assuming the one-wayness of R. Suppose there exists an algorithm $\mathcal{A}$ whose advantage against the CCA-security of IB-KEM is not negligible in κ, then we can construct an adversary $\mathcal{B}$ breaking the pseudo-randomness of F also with non-negligible advantage, which is sufficient to prove CCA-security under the one-wayness of R.

Game 1. $\mathcal{B}$ receives a challenge instance (pp, x^*, k^*), where x^* is picked from the tuple $(x^*, w^*) \in \mathsf{R}_{pp}$ generated by $\mathsf{SampR}(r^*)$ and k^* is either $\mathsf{F}(w^*)$ or randomly picked from $\{0,1\}^l$. $\mathcal{B}$ is asked to determine $k^* = \mathsf{F}(w^*)$ or $k^* \xleftarrow{R} \{0,1\}^l$. $\mathcal{B}$ plays with $\mathcal{A}$ in the following game.
Setup: $\mathcal{B}$ generates (mpk, msk') from (pp, x^*) as $\mathcal{CH}$ does in Game ABO for ABO IB-EHPS. $\mathcal{B}$ sends mpk to $\mathcal{A}$.
Phase 1 - Private key queries: $\mathcal{B}$ operates as $\mathcal{CH}$ does in Game ABO of ABO IB-EHPS.
Phase 1 - Decapsulation queries: $\mathcal{B}$ operates as $\mathcal{CH}$ does in Game ABO of ABO IB-EHPS.
Challenge: When $\mathcal{A}$ submits a target identity id^* that did not appear in any private key query in Phase 1, $\mathcal{B}$ computes $y^* = \mathsf{H}_{mpk}(id^*, x^*)$ via $\mathsf{Priv}(msk', id^*, x^*)$, then instead of creating the challenge by explicitly generating a random bit β, it sends (x^*, y^*, k^*) to $\mathcal{A}$ as the challenge.
Phase 2 - Private key queries: $\mathcal{B}$ operates as $\mathcal{CH}$ does in Game ABO of ABO IB-EHPS.
Phase 2 - Decapsulation queries: $\mathcal{B}$ operates as $\mathcal{CH}$ processes the decapsulation queries in Game ABO of ABO IB-EHPS.
Guess: $\mathcal{A}$ outputs its guess β' for β and $\mathcal{B}$ forwards β' to its own challenger.

Observe that $\mathcal{A}$'s view in Game 1 (resp. Game 0) is essentially the same as Game ABO (resp. Game Ext), and the underlying ABO IB-EHPS is (Q_e, Q_d, δ)-well-partition, we conclude that $\mathcal{B}$ can break the pseudo-randomness of F with advantage:

$$\mathsf{Adv}_{\mathcal{B}} = \left|(1-\delta)\cdot\frac{1}{2} + \delta\cdot\Pr[S_0] - \frac{1}{2}\right| = \delta\cdot\left|\Pr[S] - \frac{1}{2}\right| = \delta\cdot\mathsf{Adv}_{\mathcal{A}}^{\mathrm{CCA}}$$

Since δ is noticeable, $\mathcal{B}$'s advantage is also non-negligible, which contradicts the one-wayness of R. This proves the theorem. □

5 Instantiations of IB-EHPS

We present an ABO IB-EHPS for the bilinear Diffie-Hellman relation from Section 2.5, namely $\mathsf{R}_{pp}^{\mathsf{bdh}} = \{(x, y) \in \mathbb{G} \times \mathbb{G}_T : y = e(g, x)^{ab}\}$. Applying the transformation in Section 4.2 to this ABO IB-EHPS, we obtain an adaptive-identity CCA-secure IB-KEM based on the CBDH assumption (see Fig 1), which can be viewed as a variant of the IB-KEM in [KG06].

5.1 ABO IB-EHPS for the BDH Relation

We first run $\mathsf{SampInst}(\kappa)$ to generate $pp = (g, g^a, g^b)$, $sp = (a, b)$, and an instance $\Gamma = (X, W, \mathsf{R})$ of the BDH relation with respect to $(p, \mathbb{G}, \mathbb{G}_T, e) \leftarrow \mathsf{BLGroupGen}(\kappa)$, where $X = \mathbb{G}$, $W = \mathbb{G}_T$, R is defined as in Section 3.1. For the choice of $\mathbf{H} = (\mathsf{H}, MPK, I, X, Y)$, let $MPK = \mathbb{G}^{5+n}$ for some integer n, $I = \{0,1\}^n$, $Y = \mathbb{G}^2$. We write $\overline{u}$ for a n-length vector $(u_1, \ldots, u_n)$ hereafter. We also need a target collision resistant hash function TCR from $\mathbb{G}$ to $\mathbb{Z}_p$. For $mpk = (g, g_1', g_1, g_2, u_0, \overline{u}) \in MPK$, we define:

$$\mathsf{H}_{mpk}(id, x) = (y_1, y_2) := ((g_1^t g_1')^r, \mathsf{IHF}(id)^r)$$

Here $x = g^r$, $t \leftarrow \mathsf{TCR}(x)$, and $\mathsf{IHF}(id) = u_0 \prod_{i=1}^n u_i^{id_i}$ (id_i denotes the i-th bit of identity id) is known as Waters-hash.

Extraction Mode

- $\mathsf{Setup}(\kappa)$: run $\mathsf{SampInst}(\kappa)$ to generate $\Gamma = (X, W, \mathsf{R})$, pp, sp, choose $\mathbf{H} = (\mathsf{H}, MPK, I, X, Y)$ as above; pick $g_1', u_0 \xleftarrow{R} \mathbb{G}$, $\overline{u} \xleftarrow{R} \mathbb{G}^n$, output $mpk = (g, g_1', g_1 = g^a, g_2 = g^b, u_0, \overline{u})$, $msk = a$.
- $\mathsf{KeyGen}(msk, id)$: pick $s \xleftarrow{R} \mathbb{Z}_p$, output $sk_{id} \leftarrow (g_2^a \mathsf{IHF}(id)^s, g^s)$.
- $\mathsf{Pub}(mpk, id, r)$: compute $t \leftarrow \mathsf{TCR}(g^r)$, set $y_1 = (g_1^t g_1')^r$, $y_2 = \mathsf{IHF}(id)^r$, output $y = (y_1, y_2)$.
- $\mathsf{Ext}(sk_{id}, x, y)$: parse sk_{id} as (sk_1, sk_2) and y as (y_1, y_2), compute $t \leftarrow \mathsf{TCR}(x)$; if $e(x, g_1^t g_1') = e(g, y_1)$ and $e(x, \mathsf{IHF}(id)) = e(g, y_2)$ then output $e(x, sk_1)/e(y_2, sk_2)$, else output $\perp$.

The correctness of the extraction mode follows from the following facts:

1. $y = ((g_1^t g_1')^r, \mathsf{IHF}(id)^r) = \mathsf{H}_{mpk}(id, x) \Longrightarrow e(x, g_2^a \mathsf{IHF}(id)^s)/e(y_2, g^s) = e(g_1, g_2)^r$.
2. $y \stackrel{?}{=} \mathsf{H}_{mpk}(id, x)$ is publicly verifiable, and $\mathsf{Ext}(sk_{id}, x, y)$ outputs $\perp$ when $y \neq \mathsf{H}_{mpk}(id, x)$.

ABO Hashing Mode

- $\mathsf{Setup}'(\kappa, x^*)$: run $\mathsf{SampInst}(\kappa)$ to generate $\Gamma = (X, W, \mathsf{R})$, pp, sp, pick $\mathbf{H} = (\mathsf{H}, MPK, I, X, Y)$ as above; pick $d \xleftarrow{R} \mathbb{Z}_p$, compute $t^* \leftarrow \mathsf{TCR}(x^*)$, set $g_1' = g_1^{-t^*} g^d$; set $m = 2(Q_e + Q_d)$, and choose $k \xleftarrow{R} [n+1]$; pick $\alpha' \xleftarrow{R} \mathbb{Z}_m$, $\overline{\alpha} \xleftarrow{R} \mathbb{Z}_m^n$, $\beta' \xleftarrow{R} \mathbb{Z}_p$, $\overline{\beta} \xleftarrow{R} \mathbb{Z}_p^n$, set $u_0 = g_2^{p-km+\alpha'} g^{\beta'}$ and $u_i = g_2^{\alpha_i} g^{\beta_i}$ for $1 \leq i \leq n$; output $mpk = (g, g_1', g_1 = g^a, g_2 = g^b, u_0, \overline{u})$, $msk^* = (t^*, d, \alpha', \overline{\alpha}, \beta', \overline{\beta})$. For ease of narration we define two functions, namely $\mathsf{K}(id) = (p - mk) + \alpha' + \sum id_i \alpha_i$ and $\mathsf{L}(id) = \beta' + \sum id_i \beta_i$. Hence $\mathsf{IHF}(id)$ is essentially of the form $g_2^{\mathsf{K}(id)} g^{\mathsf{L}(id)}$. The structure of mpk implicitly splits set I into I_0 and I_1. For $id \in I$, if $\mathsf{K}(id) \neq p$ it belongs to I_0, otherwise it belongs to I_1.
- $\mathsf{KeyGen}'(msk', id)$: if $id \notin I_0$ output $\perp$, else pick $s \xleftarrow{R} \mathbb{Z}_p$ and output

$$sk_{id} = (sk_1, sk_2) = \left(g_1^{\frac{-\mathsf{L}(id)}{\mathsf{K}(id)}} \mathsf{IHF}(id)^s, g_1^{\frac{-1}{\mathsf{K}(id)}} g^s \right)$$

- $\mathsf{Priv}(msk', id, x)$: if $id \in I_1$ and $x = x^*$ output $y = (y_1, y_2) = ((x^*)^d, (x^*)^{\mathsf{L}(id)})$, else output $\perp$.
- $\mathsf{Ext}'(msk', id, x, y)$: parse y as (y_1, y_2), compute $t \leftarrow \mathsf{TCR}(x)$. If $e(x, g_1^t g_1') = e(g, y_1)$ and $e(x, \mathsf{IHF}(id)) = e(g, y_2)$, if $t = t^* \wedge id \notin I_0$ output $\perp$, otherwise either output $e((y_1/x^d)^{1/(t-t^*)}, g_2)$ or extract the private key sk_{id} and output $\mathsf{Ext}(sk_{id}, x, y)$. Else output $\perp$.

The correctness of the ABO hashing mode follows from the following two facts:

1. If $id \in I_1$ and $x = x^*$, $\mathsf{Priv}(msk', id, x^*) = ((x^*)^d, (x^*)^{K(id)}) = ((g^d)^{r^*}, (g^{K(id)})^{r^*}) = ((g_1^{t^*} g_1')^{r^*}, F(id)^{r^*}) = \mathsf{H}_{mpk}(id, x^*)$.
2. If $y = ((g_1^t g_1')^r, \mathsf{IHF}(id)^r) = \mathsf{H}_{mpk}(id, x)$ where $t = \mathsf{TCR}(x)$, then ensured by the property of TCR, Ext' outputs $\perp$ if $x = x^* \wedge id \notin I_0$ and outputs the correct w otherwise with overwhelming probability.
3. Same to the case of the extraction mode, $y \stackrel{?}{=} \mathsf{H}_{mpk}(id, x)$ is publicly verifiable, $\mathsf{Ext}(sk_{id}, x, y)$ outputs $\perp$ when $y \neq \mathsf{H}_{mpk}(id, x)$.

The well-partition property is related to the following facts:

1. The distribution of mpk in both modes are identical.
2. For any mpk and any $id \in I_1$, the output of $\mathsf{KeyGen}(msk, id)$ and $\mathsf{KeyGen}'(msk', id)$ are statistically indistinguishable. To see this, let $\tilde{s} = s - a/\mathsf{K}(id)$, we have:

$$\begin{aligned} sk_1 &= g_1^{\frac{-\mathsf{L}(id)}{\mathsf{K}(id)}} \mathsf{IHF}(id)^s = g_1^{\frac{-\mathsf{L}(id)}{\mathsf{K}(id)}} (g_2^{\mathsf{K}(id)} g^{\mathsf{L}(id)})^s \\ &= g_2^a (g_2^{\mathsf{K}(id)} g^{\mathsf{L}(id)})^{-\frac{a}{\mathsf{K}(id)}} (g_2^{\mathsf{K}(id)} g^{\mathsf{L}(id)})^s = g_2^a \mathsf{IHF}(id)^{s-\frac{a}{\mathsf{K}(id)}} = g_2^a \mathsf{IHF}(id)^{\tilde{s}} \\ sk_2 &= g_1^{\frac{-1}{\mathsf{K}(id)}} g^s = g^{s-\frac{a}{\mathsf{K}(id)}} = g^{\tilde{s}} \end{aligned}$$

   Since s is uniform in $\mathbb{Z}_p$, then $\tilde{s}$ is also uniform in $\mathbb{Z}_p$. Thus the distribution of $\mathsf{KeyGen}(msk, id)$ and $\mathsf{KeyGen}'(msk', id)$ are identical.

Follow the same analysis in [KG06], the above IB-EHPS is (Q_e, Q_d, δ)-well-partition, where $\delta \geq \frac{1}{8(n+1)(Q_e+Q_d)}$. Applying the transformation in Section 4.2 to this ABO IB-EHPS, we obtain an IB-KEM (see Figure 1), which can be viewed as a variant of the IB-KEM in [KG06]. Combining Theorem 4.2, we conclude that this IB-KEM is IND-ID-CCA secure based the CBDH assumption.

$\mathsf{Setup}(\kappa)$:	$\mathsf{Extract}(msk, id)$
$g, g_1', g_2, u_0 \xleftarrow{R} \mathbb{G}, \overline{u} \xleftarrow{R} \mathbb{G}^n$; $a \xleftarrow{R} \mathbb{Z}_p$	$s \xleftarrow{R} \mathbb{Z}_p$
$\mathsf{IHF}(id) = u_0 \prod_{i=1}^n u_i^{id_i}$	$sk = (g_2^a \mathsf{IHF}(id)^s, g^s)$
$mpk = (g, g_1 = g^a, g_1', g_2, u_0, \overline{u})$; $msk = a$	output sk
output (mpk, msk)	
$\mathsf{Encap}(mpk, id)$	$\mathsf{Decap}(sk_{id}, c)$
$r \xleftarrow{R} \mathbb{Z}_p$, $x \leftarrow g^r$	parse sk_{id} as (sk_1, sk_2), c as (x, y_1, y_2)
$t \leftarrow \mathsf{TCR}(x)$	$t \leftarrow \mathsf{TCR}(x)$
$y_1 = (g_1^t g_1')^r$	If $e(x, g_1^t g_1') \neq e(g, y_1)$ or
$y_2 = \mathsf{IHF}(id)^r$	$e(x, \mathsf{IHF}(id)) \neq e(g, y_2)$, then output $\perp$
output $c = (x, y_1, y_2)$ and $k \leftarrow \mathsf{GL}(e(g_1, g_2)^r)$	else output $\mathsf{GL}(e(x, sk_1)/e(y_2, sk_2))$

Fig. 1. An IND-ID-CCA secure IB-KEM based on BDH (variant of [KG06])

5.2 ABO IB-EHPS for the mBDH Relation

Based on the modified bilinear Diffie-Hellman relation $\mathsf{R}_{pp}^{\mathsf{mbdh}}$, we can create an ABO IB-EHPS whose Ext and Ext' algorithms implement the "implicit rejection" idea. Applying the transformation from Section 4.2 to this ABO IB-EHPS, we obtain a CCA-secure IB-KEM based on the mBDH assumption (see Figure 2), which is a variant of the IB-KEM in [KV08].

6 Selective Identity-Based Extractable Hash Proof System

In this section, we introduce the notion of selective IB-EHPS which is a natural but useful extension of IB-EHPS.

Setup(κ):	Extract(msk, id)
$g, g_2, u_0 \xleftarrow{R} \mathbb{G}, \overline{u} \xleftarrow{R} \mathbb{G}^n$; $a \xleftarrow{R} \mathbb{Z}_p$	$s \xleftarrow{R} \mathbb{Z}_p$
$\mathsf{IHF}(id) = u_0 \prod_{i=1}^n u_i^{id_i}$	$sk = (g_2^a \mathsf{IHF}(id)^s, g^{-s}, g_2^s)$
$mpk = (g, g_1 = g^a, g_2, u_0, \overline{u})$; $msk = a$	output sk
output (mpk, msk)	
Encap(mpk, id)	Decap(sk_{id}, c)
$r \xleftarrow{R} \mathbb{Z}_p$, $x \leftarrow g^r$, $t \leftarrow \mathsf{TCR}(x)$	parse sk_{id} as (sk_1, sk_2, sk_3), c as (x, y)
$y = (\mathsf{IHF}(id) g_2^t)^r$	$t \leftarrow \mathsf{TCR}(x)$
output $c = (x, y)$ and $k \leftarrow \mathsf{GL}(e(g_1, g_2)^r)$	output $\mathsf{GL}(e(x, sk_1 \cdot sk_3^t) \cdot e(y, sk_2))$

Fig. 2. An IND-ID-CCA secure IB-KEM based on mBDH (variant of [KV08])

6.1 Selective IB-EHPS

Same as IB-EHPS, a selective IB-EHPS consists of a tuple of algorithms (Setup, KeyGen, Pub, Ext, Setup′, KeyGen′, Priv). It can also behave in one of two modes, namely the extraction mode and the hashing mode. The extraction mode of selective IB-EHPS is identical to that of IB-EHPS. The hashing mode is defined as below:

- Setup′(κ, id^*): similar to Setup′(κ) in IB-EHPS except taking an extra input $id^* \in I$.
- KeyGen′(msk', id): take as input msk' and $id \in I$, if $id \neq id^*$ output a private key sk_{id} for id, else output $\perp$.
- Priv(msk', id, x): take msk', $id \in I$, and $x \in X$ as input, if $id = id^*$ output $y \in Y$ such that $y = \mathsf{H}_{mpk}(id^*, x)$, else output $\perp$.

Well-Partition. This property for selective IB-EHPS can be defined similarly as that for IB-EHPS except the following two differences: 1) in both Game Ext and Game Hash, $\mathcal{A}$ declares $id^* \in I$ before seeing mpk; 2) the private key query for id^* is not allowed.

Starting from a selective IB-EHPS for a one-way relation R, we can derive an IND-sID-CPA secure IB-KEM exactly the same way as we did in Section 4.1. The security proof is similar to that for theorem 4.1. We omit it here to avoid repetition.

6.2 Selective ABO IB-EHPS

Same as ABO IB-EHPS, a selective ABO IB-EHPS consists of a tuple of algorithms (Setup, KeyGen, Pub, Ext, Setup′, KeyGen′, Priv, Ext′). It can behave in one of two modes, namely the extraction mode and the hashing mode. The extraction mode of selective ABO IB-EHPS is identical to that of ABO IB-EHPS. The selective ABO hashing mode is defined as below:

- Setup′(κ, id^*, x^*): similar to Setup′(κ, x^*) in ABO IB-EHPS except taking an extra input $id^* \in I$.
- KeyGen′(msk', id): take as input msk' and $id \in I$, if $id \neq id^*$ output a private key sk_{id} for id, else output $\perp$.
- Priv(msk', id, x): take msk', $id \in I$, and $x \in X$ as input, if $id = id^*$ and $x = x^*$ output $y \in Y$ such that $y = \mathsf{H}_{mpk}(id^*, x^*)$, else output $\perp$.
- Ext′(msk', id, x, y): take msk', $id \in I$, $x \in X$ (suppose $(x, w) \in \mathsf{R}_{pp}$), and $y \in Y$ as input, if $y = \mathsf{H}_{mpk}(id, x)$ output $\perp$ if $id = id^* \wedge x = x^*$ and w otherwise, else output a value from $W \cup \perp$ which is independent of w.

Well-Partition. This property for selective ABO IB-EHPS is defined similarly as that for ABO IB-EHPS except the same two differences as above. Particularly, if 1) the first output of $\mathsf{Setup}(\kappa)$ and $\mathsf{Setup}(\kappa, id^*, x^*)$ are indistinguishable; 2) for $id \neq id^*$, the output of $\mathsf{KeyGen}(msk, id)$ and $\mathsf{Keygen}'(msk', id)$ are indistinguishable, then $\delta = 1$.

Selective ABO IB-EHPS can be viewed as a special case of ABO IB-EHPS that I_1 shrinks to a single point id^*. Starting from a selective ABO IB-EHPS (Setup, KeyGen, Pub, Ext, Setup', KeyGen', Priv, Ext') for a one-way relation R, we can derive an IND-sID-CCA secure IB-KEM (Setup, KeyGen, Encap, Decap) exactly the same way as we did in Section 4.1. The security proof is similar to that for theorem 4.2. We omit it here to avoid repetition.

6.3 Selective ABO IB-EHPS for the BDH Relation

The BDH relation is defined the same way as in Section 5. For the choice of $\mathbf{H} = (\mathsf{H}, MPK, I, X, Y)$, let $MPK = \mathbb{G}^5$, $I = \mathbb{Z}_p$, $Y = \mathbb{G}^2$. For $mpk = (g, g_1, g_1', g_2, h) \in MPK$, we define H_{mpk} as

$$\mathsf{H}_{mpk}(id, x) = (y_1, y_2) = ((g_1^t g_1')^r, \mathsf{IHF}(id)^r)$$

Here $x = g^r$, $t \leftarrow \mathsf{TCR}(x)$, $\mathsf{IHF}(id) = g_1^{id} h$ is known as Boneh-Boyen hash [BB04a].

Extraction Mode

- $\mathsf{Setup}(\kappa)$: run $\mathsf{SampInst}(\kappa)$ to generate $\Gamma = (X, W, \mathsf{R})$, pp, sp, choose $\mathbf{H} = (\mathsf{H}, MPK, I, X, Y)$ as above; pick $g_1', h \xleftarrow{R} \mathbb{G}$, output $mpk = (g, g_1 = g^a, g_1', g_2 = g^b, h)$, $msk = a$.
- $\mathsf{Pub}(mpk, id, r)$: compute $t \leftarrow \mathsf{TCR}(g^r)$, output $((g_1^t g_1')^r, \mathsf{IHF}(id)^r)$.
- $\mathsf{KeyGen}(msk, id)$: pick $s \xleftarrow{R} \mathbb{Z}_p$, output $sk_{id} = (g_2^a F(id)^s, g^s)$.
- $\mathsf{Ext}(sk, x, y)$: parse sk as (sk_1, sk_2) and y as (y_1, y_2), compute $t \leftarrow \mathsf{TCR}(x)$, if $e(x, g_1^t g_1') = e(g, y_1)$ and $e(x, \mathsf{IHF}(id)) = e(g, y_2)$, return $e(x, sk_1)/e(y_2, sk_2)$, else return $\perp$.

The correctness of the extraction mode follows from the following facts:

1. $y = ((g_1^t g_1')^r, \mathsf{IHF}(id)^r) = \mathsf{H}_{mpk}(id, x) \Longrightarrow e(x, g_2^a \mathsf{IHF}(id)^s)/e(y_2, g^s) = e(g_1, g_2)^r$.
2. $y \stackrel{?}{=} \mathsf{H}_{mpk}(id, x)$ is publicly verifiable, $\mathsf{Ext}(sk_{id}, x, y)$ outputs $\perp$ when $y \neq \mathsf{H}_{mpk}(id, x)$.

Selective ABO Hashing Mode

- $\mathsf{Setup}'(\kappa, id^*, x^*)$: run $\mathsf{SampInst}(\kappa)$ to generate $\Gamma = (X, W, \mathsf{R})$, pp, sp, and choose $\mathbf{H} = (\mathsf{H}, MPK, I, X, Y)$ as above; pick $d, z \xleftarrow{R} \mathbb{Z}_p$, compute $t^* \leftarrow \mathsf{TCR}(x^*)$ and set $g_1' = g_1^{-t^*} g^d$, $h = g_1^{-id^*} g^z$; output $mpk = (g, g_1', g_1 = g^a, g_2 = g^b, h)$, $msk' = (d, z)$. The identity hashing function $\mathsf{IHF}(id)$ is essentially of the form $g_1^{id - id^*} g^z$. Particularly, $\mathsf{IHF}(id^*) = g^z$.
- $\mathsf{KeyGen}'(msk', id)$: pick $s \xleftarrow{R} \mathbb{Z}_p$, output $sk_{id} = \left(g_2^{\frac{-z}{id - id^*}} (g_1^{id - id^*} g^z)^s, g^s g_2^{\frac{-1}{id - id^*}} \right)$.
- $\mathsf{Priv}(msk^*, id, x)$: if $id = id^*$ and $x = x^*$ output (x^d, x^z), else output $\perp$.
- $\mathsf{Ext}'(msk', id, x, y)$: compute $t \leftarrow \mathsf{TCR}(x)$. If $e(x, g_1^t g_1') = e(g, y_1)$ and $e(x, \mathsf{IHF}(id)) = e(g, y_2)$, if $t = t^* \wedge id = id^*$ output $\perp$, otherwise either output $e((\tau_1/u^d)^{1/(t - t^*)}, g_2)$ or extract the private key sk_{id} and return $\mathsf{Ext}(sk_{id}, x, y)$. Else output $\perp$.

The correctness of the selective ABO hashing mode follows from the following facts:

1. If $y = ((g_1^t g_1')^r, \mathsf{IHF}(id)^r) = \mathsf{H}_{mpk}(id, x)$ where $t \leftarrow \mathsf{TCR}(x)$, then ensured by the property of TCR, Ext' outputs $\perp$ if $x = x^* \wedge id = id^*$ and outputs the correct w otherwise with overwhelming probability.

2. Same to the case of the extraction mode, $y \stackrel{?}{=} \mathsf{H}_{mpk}(id, x)$ is publicly verifiable, $\mathsf{Ext}(sk_{id}, x, y)$ outputs $\perp$ when $y \neq \mathsf{H}_{mpk}(id, x)$.

The $(Q_e, Q_d, 1)$-well-partition property is established from the following two facts:

- For any $id^* \in I$ and $x^* \in X$, the distribution of mpk in both modes are identical.
- For any mpk and any identity $id \neq id^*$, the output of $\mathsf{KeyGen}(msk, id)$ and $\mathsf{KeyGen}^*(msk^*, id)$ are statistically indistinguishable. To see this, let $\tilde{s} = s - b/(id - id^*)$, we have:

$$sk_1 = g_2^{\frac{-z}{id-id^*}}(g_1^{id-id^*}g^z)^s = g_2^a(g_1^{id-id^*}g^z)^{s-\frac{b}{id-id^*}} = g_2^a \mathsf{IHF}(id)^{\tilde{s}}$$
$$sk_2 = g^s g_2^{\frac{-1}{id-id^*}} = g^{\tilde{s}}$$

Since s is uniform in $\mathbb{Z}_p$, then $\tilde{s}$ is also uniform in $\mathbb{Z}_p$. Thus the output of $\mathsf{KeyGen}(msk, id)$ and $\mathsf{KeyGen}'(msk', id)$ are statistically indistinguishable.

Applying the transformation from Section 4.1 to the above selective ABO IB-EHPS, we obtain an IND-sID-CCA secure IB-KEM based on the CBDH assumption, the IB-KEM due to Haralambiev *et al.*[HJKS10] and Galindo [Gal10] can be simplified to the IB-KEM in [CCZ11, Section 3]. Thus our selective ABO IB-EHPS based on BDH relation can explain the IB-KEMs in [HJKS10, Gal10] as well.

7 Further Discussions

Inspired by [Wee11], our notion of identity-based extractable hash proof systems can be further generalized to threshold setting easily. Its applications may include threshold identity-based encryption, identity-based broadcast encryption, and identity-based encryption with non-interactive opening, etc.

Although EHPS can be constructed from various assumptions and provide us a unifying framework to explain many CCA-secure PKE schemes from search problems, IB-EHPS is currently only known to be based on BDH-style assumptions. However, we think this contrast is understandable from the following discussion.

- Why not factoring or RSA?
  In general, constructing an IBE scheme is harder than constructing a PKE scheme. Although CCA-secure PKE schemes from factoring have been proposed recently, constructing IBE schemes based on factoring assumption or RSA-type assumption in the standard model is still a longstanding problem. As soon as we obtain an IB-EHPS based on such kind of assumptions, the open problem will be immediately solved.
- Why not lattice?
  Lattices have recently emerged as a powerful mathematical platform on which to build a rich variety of cryptographic primitives. It is compelling to know if we can instantiate IB-EHPS from relations related to lattices. As we already mentioned in Section 4, IB-EHPS inherently relies on the Rackoff-Simon paradigm. However, all the known encryption schemes based on lattice [GPV08, ABB10, CHKP10] are falling out of the this paradigm. Instead, they fall into the paradigm of HPS or IB-HPS [CZLC13]. This explains why it turns out to be hard to instantiate IB-EHPS based on lattice. As soon as we are able to construct an IB-EHPS from assumptions related to lattice, we will find a new approach to use lattice to construct encryption schemes. We left this as an open problem.

8 Acknowledgment

The authors greatly thank the anonymous ACNS 2012 reviewers for many helpful and insightful comments. The work of the first author and the third author is supported by the Foundation of SKLOIS of Chinese Academy of Sciences, the Strategic Priority Research Program of Chinese Academy of Sciences under Grant No. XDA06010701, the National 973 Program of China under Grant No. 2011CB302400, and the National Natural Science Foundation of China under Grant No. 60970152. The work of the second and the fourth authors is partially supported by the National Natural Science Foundation of China under Grant Nos. 61033014, 60970110, 60972034 and Asia 3 Foresight Program under Grant No. 61161140320.

References

[ABB10] Shweta Agrawal, Dan Boneh, and Xavier Boyen. Efficient lattice (H)IBE in the standard model. In *Advances in Cryptology - EUROCRYPT 2010*, volume 6110 of *Lecture Notes in Computer Science*, pages 553–572. Springer, 2010.

[ABR01] Michel Abdalla, Mihir Bellare, and Phillip Rogaway. The oracle diffie-hellman assumptions and an analysis of dhies. In *Topics in Cryptology - CT-RSA 2001*, volume 2020 of *LNCS*, pages 143–158, 2001.

[ADN+10] Joël Alwen, Yevgeniy Dodis, Moni Naor, Gil Segev, Shabsi Walfish, and Daniel Wichs. Public-key encryption in the bounded-retrieval model. In *Advances in Cryptology - EUROCRYPT 2010*, volume 6110 of *LNCS*, pages 113–134. Springer, 2010.

[BB04a] Dan Boneh and Xavier Boyen. Efficient selective-id secure identity based encryption without random oracles. In *Advances in Cryptology - EUROCRYPT 2004*, volume 3027 of *LNCS*, pages 223–238, 2004.

[BB04b] Dan Boneh and Xavier Boyen. Secure identity based encryption without random oracles. In *Advances in Cryptology - CRYPTO 2004*, volume 3152 of *LNCS*, pages 443–459, 2004.

[BCHK07] Dan Boneh, Ran Canetti, Shai Halevi, and Jonathan Katz. Chosen-ciphertext security from identity-based encryption. *SIAM J. Comput.*, 36(5):1301–1328, 2007.

[BF01] Dan Boneh and Matthew Franklin. Identity-based encryption from the weil pairing. In *Advances in Cryptology - CRYPTO 2001*, volume 2139 of *LNCS*, pages 213–229, 2001.

[BF03] Dan Boneh and Matthew K. Franklin. Identity-based encryption from the weil pairing. *SIAM Journal on Computation*, 32:586–615, 2003.

[BFMLS08] Kamel Bentahar, Pooya Farshim, John Malone-Lee, and Nigel P. Smart. Generic constructions of identity-based and certificateless kems. *Journal of Cryptology*, 21(2):178–199, 2008.

[BGH07] Dan Boneh, Craig Gentry, and Michael Hamburg. Space-efficient identity based encryption without pairings. In *48th Annual IEEE Symposium on Foundations of Computer Science (FOCS 2007)*, pages 647–657. IEEE Computer Society, 2007.

[BMW05] Xavier Boyen, Qixiang Mei, and Brent Waters. Direct chosen ciphertext security from identity-based techniques. *ACM CCS 2005*, pages 320–329, 2005.

[BR95] Mihir Bellare and Phillip Rogaway. Random oracles are practical: A paradigm for designing efficient protocols. *ACM conference on Computers and Communication Security*, pages 62–73, 1995.

[BR97] Mihir Bellare and Phillip Rogaway. Collision-resistant hashing: Towards making uowhfs practical. In *Advances in Cryptology - CRYPTO '97*, volume 1294 of *Lecture Notes in Computer Science*, pages 470–484. Springer, 1997.

[CC05] Liqun Chen and Zhaohui Cheng. Security proof of sakai-kasahara's identity-based encryption scheme. In *Cryptography and Coding, 10th IMA International Conference*, volume 3796 of *LNCS*, pages 442–459, 2005.

[CCZ11] Yu Chen, Liqun Chen, and Zongyang Zhang. CCA secure IB-KEM from the computational bilinear diffie-hellman assumption in the standard model. Cryptology ePrint Archive, Report 2011/593, 2011. http://eprint.iacr.org/2011/593.

[CGH98] Ran Canetti, Oded Goldreich, and Shai Halevi. The random oracle methodology, revisited (preliminary version). In *STOC*, pages 209–218, 1998.

[CHK04] Ran Canetti, Shai Halevi, and Jonathan Katz. Chosen-ciphertext security from identity based encryption. In *Advances in Cryptology - EUROCRYPT 2004*, volume 3027 of *LNCS*, pages 207–222, 2004.

[CHK10] Ronald Cramer, Dennis Hofheinz, and Eike Kiltz. A twist on the naor-yung paradigm and its application to efficient cca-secure encryption from hard search problems. In *Theory of Cryptography, 7th Theory of Cryptography Conference, TCC 2010*, volume 5978 of *LNCS*, pages 146–164. Springer, 2010.

[CHKP10] David Cash, Dennis Hofheinz, Eike Kiltz, and Chris Peikert. Bonsai trees, or how to delegate a lattice basis. In *Advances in Cryptology - EUROCRYPT 2010*, volume 6110 of *Lecture Notes in Computer Science*, pages 523–552. Springer, 2010.

[CKS08] David Cash, Eike Kiltz, and Victor Shoup. The twin diffie-hellman problem and applications. In *Advances in Cryptology - EUROCRYPT 2008*, volume 4965 of *LNCS*, pages 127–145, 2008.

[Coc01] Clifford Cocks. An indentity based encryption scheme based on quadratic residues. In *Institute of Mathematics and Its Applications International Conference on Cryptigraphy and Coding Proceedings of IMA 2001*, volume 2260 of *LNCS*, pages 360–363, 2001.

[CS02] Ronald Cramer and Victor Shoup. Universal hash proofs and a paradigm for adaptive chosen ciphertext secure public-key encryption. In *Advances in Cryptology - EUROCRYPT 2002*, volume 2332 of *LNCS*, pages 45–64. Springer, 2002.

[CS03] Ronald Cramer and Victor Shoup. Design and analysis of practical public-key encryption schemes secure against adaptive chosen ciphertext attack. *SIAM Journal on Computing*, 33:167–226, 2003.

[CZLC12] Yu Chen, Zongyang Zhang, Dongdai Lin, and Zhenfu Cao. Anonymous identity-based hash proof system and its applications. In *Provable Security - 6th International Conference, ProvSec 2012*, volume 7496 of *LNCS*, pages 143–160. Springer, 2012.

[CZLC13] Yu Chen, Zongyang Zhang, Dongdai Lin, and Zhenfu Cao. Generalized (identity-based) hash proof system and its applications. Cryptology ePrint Archive, Report 2013/002, 2013. http://eprint.iacr.org/.

[FO99] Eiichiro Fujisaki and Tatsuaki Okamoto. Secure integration of asymmetric and symmetric encryption schemes. In *Advances in Cryptology - CRYPTO 1999*, volume 1666 of *LNCS*, pages 537–554, 1999.

[Gal10] David Galindo. Chosen-ciphertext secure identity-based encryption from computational bilinear diffie-hellman. In *Pairing-Based Cryptography - Pairing 2010*, volume 6487 of *LNCS*, pages 367–376. Springer, 2010.

[Gen06] Craig Gentry. Practical identity-based encryption without random oracles. In *Advances in Cryptology - EUROCRYPT 2006*, volume 4004 of *LNCS*, pages 445–464, 2006.

[GL89] Oded Goldreich and Leonid A. Levin. A hard-core predicate for all one-way functions. In *Proceedings of the Twenty-First Annual ACM Symposium on Theory of Computing, STOC*, pages 25–32. ACM, 1989.

[GPV08] Craig Gentry, Chris Peikert, and Vinod Vaikuntanathan. Trapdoors for hard lattices and new cryptographic constructions. In *Proceedings of the 40th Annual ACM Symposium on Theory of Computing, STOC*, pages 197–206. ACM, 2008.

[HJKS10] Kristiyan Haralambiev, Tibor Jager, Eike Kiltz, and Victor Shoup. Simple and efficient public-key encryption from computational diffie-hellman in the standard model. In *Public Key Cryptography - PKC 2010*, volume 6056 of *LNCS*, pages 1–18. Springer, 2010.

[HK08] Goichiro Hanaoka and Kaoru Kurosawa. Efficient chosen ciphertext secure public key encryption under the computational diffie-hellman assumption. In *Advances in Cryptology - ASIACRYPT 2008*, volume 5350 of *LNCS*, pages 308–325. Springer, 2008.

[HK09] Dennis Hofheinz and Eike Kiltz. Practical chosen ciphertext secure encryption from factoring. In *Advances in Cryptology - EUROCRYPT 2009*, volume 5479 of *LNCS*, pages 313–332. Springer, 2009.

[KG06] Eike Kiltz and David Galindo. Direct Chosen-Ciphertext Secure Identity-Based Key Encapsulation Without Random Oracles. In *Information Security and Privacy, 11th Australasian Conference, ACISP 2006*, volume 4058 of *LNCS*, pages 336–347, 2006.

[Kil06] Eike Kiltz. Chosen-ciphertext security from tag-based encryption. In *Theory of Cryptography, TCC 2006*, volume 3876 of *LNCS*, pages 581–600. Springer, 2006.

[Kil07] Eike Kiltz. Chosen-ciphertext secure key-encapsulation based on gap hashed diffie-hellman. In *Public Key Cryptography - PKC 2007*, volume 4450 of *LNCS*, pages 282–297. Springer, 2007. Full version is avaiable at ePrint Archive: Report 2007/036.

[KV08] Eike Kiltz and Yevgeniy Vahlis. CCA2 secure IBE: Standard model efficiency through authenticated symmetric encryption. In *CT-RSA*, volume 4964 of *LNCS*, pages 221–238. Springer, 2008.

[MRS88] Silvio Micali, Charles Rackoff, and Bob Sloan. The notion of security for probabilistic cryptosystems. *SIAM J. Comput.*, 17(2):412–426, 1988.

[NY89] Moni Naor and Moti Yung. Universal one-way hash functions and their cryptographic applications. In *Proceedings of the 21st Annual ACM Symposium on Theory of Computing, STOC 1989*, pages 33–43. ACM, 1989.

[PW08] Chris Peikert and Brent Waters. Lossy trapdoor functions and their applications. In *STOC 2008*, pages 187–196, 2008.

[RS91] Charles Rackoff and Daniel R. Simon. Non-interactive zero-knowledge proof of knowledge and chosen ciphertext attack. In *Advances in Cryptology - CRYPTO 1991*, volume 576 of *LNCS*, pages 433–444, 1991.

[Sha84] Adi Shamir. Identity-based cryptosystems and signatures schemes. In *Advances in Cryptology - CRYPTO 1984*, volume 196 of *LNCS*, pages 47–53, 1984.

[SK03] Ryuichi Sakai and Masao Kasahara. Id based cryptosystems with pairing on elliptic curve. Cryptology ePrint Archive, Report 2003/054, 2003. http://eprint.iacr.org/2003/054.

[Wat05] Brent Waters. Efficient identity-based encryption without random oracles. In *Advances in Cryptology - EUROCRYPT 2005*, volume 3494 of *LNCS*, pages 114–127, 2005.

[Wat09] Brent Waters. Dual system encryption: Realizing fully secure ibe and hibe under simple assumptions. In *Advances in Cryptology - CRYPTO 2009*, volume 5677 of *LNCS*, pages 619–636. Springer, 2009.

[Wee10] Hoeteck Wee. Efficient chosen-ciphertext security via extractable hash proofs. In *Advances in Cryptology - CRYPTO 2010*, volume 6223 of *LNCS*, pages 314–332. Springer, 2010.

[Wee11] Hoeteck Wee. Threshold and revocation cryptosystems via extractable hash proofs. In *Advances in Cryptology - EUROCRYPT 2011*, volume 6632 of *Lecture Notes in Computer Science*, pages 589–609. Springer, 2011.

A The Known Selective-Identity CCA-secure IB-KEM Constructions

In this section, we recall all the three existing IB-KEMs [HJKS10, Gal10, CCZ11] that were proven to be CCA-secure based on computational assumptions without random oracles. To get a clear understanding of their essence, we describe the one-bit version of these IB-KEMs. [CCZ11] has already shows that all the one-bit IB-KEMs [HJKS10, Gal10, CCZ11] can extend to n-bit IB-KEM via several different approaches.

A.1 Haralambiev et al.'s Scheme

Haralambiev *et al* [HJKS10] mentioned that one of their PKE schemes can extends to a BB_1-style IB-KEM which is selective-identity CCA-secure based on the CBDH assumption. However, the concrete construction is not given. For completeness, we provide the construction according to our understanding.

$\mathsf{Setup}(\kappa)$: run $\mathsf{BLGroupGen}(\kappa)$ to generate $(p, \mathbb{G}, \mathbb{G}_T, e)$, pick $a, b \xleftarrow{R} \mathbb{Z}_p$, $h, g_1' \xleftarrow{R} \mathbb{G}$, set $mpk = (g, g_1 = g^a, g_1', g_2 = g^b, h)$ while $msk = (a, b)$. Choose a TCR hash function $\mathsf{TCR} : \mathbb{G} \to \mathbb{Z}_p^*$, set the identity space $I = \mathbb{Z}_p$, define the identity hashing function $\mathsf{IHF} : \mathbb{Z}_p \to \mathbb{G}$ as $\mathsf{IHF}(id) = g_1^{id} h$.
$\mathsf{KeyGen}(msk, id)$: pick $s \xleftarrow{R} \mathbb{Z}_p$ and output $sk_{id} = (g^{ab}\mathsf{IHF}(id)^s, g^s)$.
$\mathsf{Encap}(mpk, id)$: pick $r \xleftarrow{R} \mathbb{Z}_p$, compute $t \leftarrow \mathsf{TCR}(x)$, set $x = g^r$, $y_1 = (g_1^t g_1')^r$, and $y_2 = \mathsf{IHF}(id)^r$; output a ciphertext $c = (x, y_1, y_2)$ and a DEM key $k \leftarrow \mathsf{GL}(e(g_1, g_2)^r)$.
$\mathsf{Decap}(sk_{id}, c)$: parse sk_{id} as (sk_1, sk_2) and c as (x, y_1, y_2), compute $t \leftarrow \mathsf{TCR}(x)$; if $e(x, g_1^t g_1') = e(g, y_1) \wedge e(x, \mathsf{IHF}(id)) = e(g, y_2)$ return $k \leftarrow \mathsf{GL}(e(x, sk_1)/e(y_2, sk_2))$, else return $\perp$.

A.2 Galindo's Scheme

Galindo [Gal10] proposed a selective-identity CCA-secure IB-KEM by extending the PKE scheme [HK08]. For a clear comparison to other schemes, we use symmetric pairing in place of asymmetric pairing in the original scheme.

$\mathsf{Setup}(\kappa)$: run $\mathsf{BLGroupGen}(\kappa)$ to generate $(p, \mathbb{G}, \mathbb{G}_T, e)$, pick $a_0, a_1, a_2, a_3 \xleftarrow{R} \mathbb{Z}_p^*$ and define the polynomial $f(t) = a_0 + a_1 t + a_2 t^2 + a_3 t^3$, pick $b \xleftarrow{R} \mathbb{Z}_p$, pick $h \xleftarrow{R} \mathbb{G}$, $mpk = (g, g_0 = g^{a_0}, g_1 = g^{a_1}, g_2 = g^{a_2}, g_3 = g^{a_3}, \bar{g}_0 = g^b, h)$, $msk = (a_0, a_1, a_2, a_3, b)$. Choose a TCR hash function $\mathsf{TCR} : \mathbb{G} \times \{0, 1\} \to \mathbb{Z}_p^*$, set the identity space $I = \mathbb{Z}_p$, define the identity hashing function $\mathsf{IHF} : \mathbb{Z}_p \to \mathbb{G}$ as $\mathsf{IHF}(id) = g_0^{id} h$.
$\mathsf{KeyGen}(msk, id)$: pick $s \xleftarrow{R} \mathbb{Z}_p$, output $sk_i d = (g^{a_0 b}\mathsf{IHF}(id)^s, g^s)$.
$\mathsf{Encap}(mpk, id)$: pick $r \xleftarrow{R} \mathbb{Z}_p$, compute $t_0 \leftarrow \mathsf{TCR}(x, 0)$, and $t_1 \leftarrow \mathsf{TCR}(x, 1)$, set $x = g^r$, $y_0 = g^{rf(t_0)}$, $y_1 = g^{rf(t_1)}$, and $y_2 = \mathsf{IHF}(id)^r$, output a ciphertext $c = (x, y_0, y_1, y_2)$ and a DEM key $k \leftarrow \mathsf{GL}(e(g_0, \bar{g}_0)^r)$.
$\mathsf{Decap}(sk_{id}, c)$: parse sk_{id} as (sk_1, sk_2) and c as (x, y_0, y_1, y_2), compute $t_0 \leftarrow \mathsf{TCR}(x, 0)$ and $t_1 \leftarrow \mathsf{TCR}(x, 1)$. If $e(x, g^{f(t_0)}) \neq e(g, y_0)$ or $e(x, g^{f(t_1)}) \neq e(g, y_1)$ or $e(x, \mathsf{IHF}(id)) \neq e(g, y_2)$ then return $\perp$. Otherwise parse sk_{id} as (sk_1, sk_2) and return $k \leftarrow \mathsf{GL}(e(x, sk_1)/e(y_2, sk_2))$.

A.3 Chen et al.'s Scheme

Chen *et al.* [CCZ11] proposed a one-bit IB-KEM which is similar to that of [HJKS10]. We re-write is as follows.

$\mathsf{Setup}(\kappa)$: run $\mathsf{BLGroupGen}(\kappa)$ to generate $(p, \mathbb{G}, \mathbb{G}_T, e)$, pick $a \xleftarrow{R} \mathbb{Z}_p$, $h, g_1', g_2 \xleftarrow{R} \mathbb{G}$, set $mpk = (g, g_1 = g^a, g_1', g_2, h)$, $msk = a$. Choose a TCR hash function $\mathsf{TCR} : \mathbb{G} \to \mathbb{Z}_p^*$, set the identity space $I = \mathbb{Z}_p$, define the identity hashing function $\mathsf{IHF} : \mathbb{Z}_p \to \mathbb{G}$ as $\mathsf{IHF}(id) = g_1^{id} h$.
$\mathsf{KeyGen}(msk, id)$: pick $s \xleftarrow{R} \mathbb{Z}_p$, output $sk = (g_2^a \mathsf{IHF}(id)^s, g^s)$.
$\mathsf{Encap}(mpk, id)$: same as the IB-KEM presented in Section A.1.
$\mathsf{Decap}(sk, c)$: same as the IB-KEM presented in Section A.1.

A.4 A Unified Interpretation

As we claimed in Section 6.3, the selective ABO IB-EHPS from the CBDH assumption can encompass all the known selective-identity CCA-secure IBE schemes [HJKS10, Gal10, CCZ11] based on the CBDH assumption. Next we explain the reason by identifying the connections among these schemes. Galindo's IB-KEM scheme may be viewed as a natural extension of the PKE scheme [HK08] combining with the Boneh-Boyen hash [BB04a]. Judging from the appearance, it differs much from the constructions of [HJKS10] and [CCZ11]. However, in Section B we show that the KEM scheme [HK08] can be greatly simplified by relying on a slightly stronger assumption or applying the twinning framework. The intuition of simplification is providing the simulator a strategy to check the consistency of the ciphertext. Particularly, the KEM [HK08] will become publicly verifiable when it builds upon groups with pairing. In line of this observation, Galindo's IB-KEM scheme [Gal10] can be significantly simplified without changing the underlying assumption. The resulting scheme is exactly the IB-KEM scheme implicitly mentioned in [HJKS10] and detailed in Section A.1. Moreover, Haralambiev *et al.*'s scheme does not have to include element b in msk. Hence the IB-KEM constructions of [HJKS10] and [Gal10] can be finally simplified to the IB-KEM proposed by Chen *et al.* [CCZ11].

B Observations on HK2008

Hanaoka and Kurosawa [HK08] proposed a novel CCA-secure KEM based on the CDH assumption. We will refer to it as HK-KEM. Compared to related works [CKS08, CHK10, HJKS10, Wee10], HK-KEM adopts a different approach to resist chosen-ciphertext attack, that is, achieving CCA-security from Broadcast Encryption (BE) with verifiability. Based on the CDH assumption, HK-PKE is not publicly verifiable. In the security reduction, the simulator checks the consistency of the ciphertext by comparing the "session key" computed from different combinations of ciphertext components. Note that if the simulator can check the consistency of the ciphertext without doing redundant computation, then HK-KEM can be significantly simplified. Generally there are two approaches can achieve this goal. One approach is resorting to a slightly stronger assumption such as strong Diffie-Hellman (SDH) assumption [ABR01]. The other approach is applying the twin Diffie-Hellman framework [CKS08].

Next we show a simplification of HK-KEM [HK08] via the first approach.
$\mathsf{Gen}(\kappa)$: pick $a_0, a_1 \xleftarrow{R} \mathbb{Z}_p^*$ and define a degree one polynomial $f(t) = a_0 + a_1 t$, set $pk = (g, g_0 = g^{a_0}, g_1 = g^{a_1})$ and $sk = (a_0, a_1)$. Choose a TCR function $\mathsf{TCR} : \mathbb{G} \to \mathbb{Z}_p$.
$\mathsf{Encap}(id)$: pick $r \xleftarrow{R} \mathbb{Z}_p$, compute $t \leftarrow \mathsf{TCR}(g^r)$, set $c_0 = g^r$ and $c_1 = g^{rf(t)}$ (note that $g^{f(t)} = g^{a_0 + a_1 t} = g_0 g_1^t$, thus one can easily compute c_1 from g_0, g_1), return a ciphertext $c = (c_0, c_1)$ and a corresponding DEM key $k \leftarrow \mathsf{GL}(g_0^r)$.

$\mathsf{Decap}(sk, c)$: parse sk as (sk_1, sk_2) and c as (c_0, c_1), check if $c_1 = c_0^{f(t)}$ for $t \leftarrow \mathsf{TCR}(c_0)$ (note that $f(t)$ is computable with $sk = (a_0, a_1)$). If so output $k \leftarrow \mathsf{GL}(c_0^{a_0})$, else output $\perp$.

Theorem 2.1 *The above scheme is IND-CCA secure if the SDH assumption holds in $\mathbb{G}$ and* TCR *is a target collision resistant hash function.*

Proof. In the following, let $c^* = (c_0^*, c_1^*)$ denote the challenge ciphertext, k^* denote the corresponding DEM key encapsulated in c^*, and let $t^* = \mathsf{TCR}(c_0^*)$. To establish IND-CCA security, we proceed via a sequence of games. We start with Game 0 where the challenger proceeds like the standard IND-CCA game and end up with a game where $k^* \xleftarrow{R} \{0, 1\}$. Let S be the event that $\mathcal{A}$ wins Game CCA, and S_i be the event that $\mathcal{A}$ wins Game i.

Game CCA. This is the standard IND-CCA game for KEM. By definition we have:

$$\Pr[S] = \frac{1}{2} + \mathsf{Adv}_{\mathcal{A}}^{\mathrm{CCA}}(\kappa) \tag{9}$$

Game 0. Let E_0 be the event that the adversary issues a decapsulation query $\langle c_0, c_1 \rangle$ with $c_0 = c_0^*$ in Phase 1. Note that the probability that the adversary submits such a decapsulation query before seeing the challenge ciphertext is bounded by Q_d/p, where Q_d is the number of decapsulation queries issued by $\mathcal{A}$. Since $Q_d = \mathsf{poly}(\kappa)$, we have $\Pr[E_{01}] \leq Q_d/p \leq \mathsf{negl}(\kappa)$. We define Game 0 exactly the same as Game CCA except assuming that E_0 never occurs in Game 0. It follows that:

$$|\Pr[S_0] - \Pr[S]| \leq \mathsf{negl}(\kappa) \tag{10}$$

Game 1. Let E_1 be the event that the adversary issues a decapsulation query $\langle c_0, c_1 \rangle$ with $c_0 \neq c_0^*$ but $\mathsf{TCR}(c_0) = \mathsf{TCR}(c_0^*)$. By the target collision resistance of TCR, we have $\Pr[E_{12}] \leq \mathsf{negl}(\kappa)$. We define Game 1 exactly the same as Game 0 except assuming that E_1 never occurs in Game 1. It follows that:

$$|\Pr[S_1] - \Pr[S_0]| \leq \mathsf{negl}(\kappa) \tag{11}$$

We claim that:

$$|\Pr[S_1] - \frac{1}{2}| \leq \mathsf{negl}(\kappa) \tag{12}$$

assuming the SDH assumption holds. We prove this statement as follows. Suppose there exists an algorithm $\mathcal{A}$ such that $|\Pr[S_1] - 1/2| = \mathsf{poly}(\kappa)$, then we can construct an algorithm $\mathcal{B}$ distinguishing $\mathsf{GL}(\mathrm{dh}(g^a, g^b))$ from a random bit with access to $\mathrm{sdh}(g^a, \cdot, \cdot)$ with non-negligible advantage, which is sufficient to prove the security based on the SDH assumption. $\mathcal{B}$ receives a challenge instance (g, g^a, g^b, L) of SDH assumption, where L is either $\mathsf{GL}(\mathrm{dh}(g^a, g^b))$ or a random bit. $\mathcal{B}$ plays Game 1 with $\mathcal{A}$ as follows:

Setup: $\mathcal{B}$ picks a TCR function TCR and computes $t^* \leftarrow \mathsf{TCR}(g^b)$; sets $g_0 = g^a$, and picks $z^* \xleftarrow{R} \mathbb{Z}_p$. Let $f(t) = a_0 + a_1 t$ be a polynomial over $\mathbb{Z}_p$ such that $f(0) = a$ and $f(t^*) = z^*$. Here it is straightforward that $a_0 = a$, $a_1 = (z^* - a)/t^*$. Note that both a_0 and a_1 are unknown to $\mathcal{B}$. $\mathcal{B}$ computes $g_1 = g^{a_1} = (g^{z^*}/g_0)^{1/t^*}$. Finally, $\mathcal{B}$ sends to $\mathcal{A}$ the public key $pk = (g, g_0, g_1)$. It is easy to see that pk has the identical distribution as the real one.

Phase 1 - Decapsulation Queries: When $\mathcal{A}$ issues a decapsulation query $\langle c_0, c_1 \rangle$, $\mathcal{B}$ first computes $t \leftarrow \mathsf{TCR}(c_0)$. Suppose that $c_1 = c_0^z$ for some integer z and $f'(t) = a_0' + a_1' t$ is a one-degree polynomial such that $f'(t) = z$ and $f'(t^*) = z^*$. From two distinct points $(t, f'(t) = z)$ and $(t^*, f'(t^*) = z^*)$ we can write a_0' as

$$a_0' = \frac{tf'(t^*) - t^* f'(t)}{t - t^*}$$

Thus $\mathcal{B}$ can compute $c_0^{f'(0)}$ as $(c_0^{tf'(t^*)-t^*f'(t)})^{\frac{1}{t-t^*}} = (c_0^{tz^*}/c_1^{t^*})^{\frac{1}{t-t^*}}$. $\mathcal{B}$ tests the consistency of ciphertexts by querying $\text{dhp}(g_0, c_0, c_0^{f'(0)})$, which returns 1 if and only if $c_0^{f'(0)} = \text{dh}(g_0, c_0)$. (The equation holds implies $f'(0) = f(0)$ and thus f' is identical to f, thereby the ciphertext is valid.) If this test is passed, $\mathcal{B}$ returns $\mathsf{GL}(c_0^{f(0)})$. Otherwise, $\mathcal{B}$ returns $\perp$.

Challenge: $\mathcal{B}$ creates $c^* = (c_0^*, c_1^*)$, where $c_0^* = g^b$, $c_1^* = (c_0^*)^{z^*}$. It is easy to see that c^* is a valid ciphertext. $\mathcal{B}$ returns c^* combined with L as the challenge.

Phase 2 - Decapsulation Queries: When $\mathcal{A}$ issues a decapsulation query $c = \langle c_0, c_1 \rangle$, $\mathcal{B}$ responds as follows:

- If $c_0 = c_0^*$, then $\mathcal{B}$ responds with $\perp$. In this case, c is either illegal (equal to c^*) or invalid because c_0 uniquely determines c_1 (i.e., $c_1 \neq c_1^*$).
- If $c_0 \neq c_0^*$, $\mathcal{B}$ responds as it did in Phase 1.

Guess: $\mathcal{A}$ outputs its guess β' for β, $\mathcal{B}$ forwards β' to its own challenger.

According to the definition of Game 1, $\mathcal{B}$'s simulation is perfect. Therefore if $\mathcal{A}$'s advantage is non-negligible, $\mathcal{B}$ has non-negligible advantage against the SDH assumption. This proves the statement. The desired security immediately follows. □

Surprisingly, we find that the above simplification bears a close resemblance to the KEM scheme presented in [Kil07, Remark 4.2] and the "toy" KEM scheme present in [HJKS10] from the SDH assumption. All the three schemes use the same trick in security reduction, that is, hiding a degree one polynomial $f(t) = a_0 + a_1 t$ in the second element c_1 of the ciphertext (equal to $g^{f(t)}$). The only difference is that in our simplification the exponent a is embedded in the first coefficient a_0 while in [Kil07] and [HJKS10] the exponent a is embedded in the second coefficient a_1. As mentioned before, we can also avoid the need of resorting to a stronger assumption by using the twin Diffie-Hellman framework. The resulting scheme is exactly the one-bit version KEM in [HJKS10, Section 3] and [Wee10, Section 5.2].