Leakage-resilient Identity-based Encryption in Bounded Retrieval Model with Nearly Optimal Leakage-Ratio

Ryo Nishimaki and Takashi Yamakawa

NTT Secure Platform Laboratories, Tokyo, Japan
{ryo.nishimaki.zk, takashi.yamakawa.ga}@hco.ntt.co.jp

Abstract. We propose new constructions of leakage-resilient public-key encryption (PKE) and identity-based encryption (IBE) schemes in the bounded retrieval model (BRM). In the BRM, adversaries are allowed to obtain at most ℓ-bit leakage from a secret key and we can increase ℓ only by increasing the size of secret keys without losing efficiency in any other performance measure. We call $\ell/|\mathsf{sk}|$ leakage-ratio where $|\mathsf{sk}|$ denotes a bit-length of a secret key. Several PKE/IBE schemes in the BRM are known. However, none of these constructions achieve a constant leakage-ratio under a standard assumption in the standard model. Our PKE/IBE schemes are the first schemes in the BRM that achieve leakage-ratio $1-\epsilon$ for any constant $\epsilon > 0$ under standard assumptions in the standard model.
As previous works, we use identity-based hash proof systems (IB-HPS) to construct IBE schemes in the BRM. It is known that a parameter for IB-HPS called the universality-ratio is translated into the leakage-ratio of the resulting IBE scheme in the BRM. We construct an IB-HPS with universality-ratio $1-\epsilon$ for any constant $\epsilon > 0$ based on any inner-product predicate encryption (IPE) scheme with compact secret keys. Such IPE schemes exist under the d-linear, subgroup decision, learning with errors, or computational bilinear Diffie-Hellman assumptions. As a result, we obtain IBE schemes in the BRM with leakage-ratio $1-\epsilon$ under any of these assumptions. Our PKE schemes are immediately obtained from our IBE schemes.

1 Introduction

1.1 Background

Modern cryptography have been placing much importance on provable security. In a traditional theory of provable security, we often assume that secret values (e.g., secret key, randomness etc.) are perfectly hidden from an adversary, and give a security proof in such models. On the other hand, developments of side channel attacks have discovered that an adversary may obtain partial information of these secret values, and some cryptographic schemes can be broken due to the leakage even though they are provably secure in the model where secret values are perfectly

hidden. To withstand these attacks, Akavia et al. [AGV09] initiated the study of leakage resilient cryptography, where leakages from secret values are captured in a security model, and their security is proven even if a certain amount of secret values is leaked to an adversary. There have been vast amount of studies on leakage resilient cryptography including public key encryption, identity-based encryption, attribute-based encryption, digital signatures, identification, zero-knowledge proofs etc. [NS12,ADN+09,HLWW16,LRW11,KP17,KV09,BSW13,ADW09,GJS11].

Relative-leakage and Absolute-leakage. If a whole secret key is leaked, then no security remains. Thus we have to bound an amount of leakages an adversary can obtain to prove security in the presence of leakages. There are two possible choices for the way to bound an amount of leakage. In the first choice called the *relative-leakage model*, we bound a *leakage-ratio* $0 < \alpha < 1$, and we allow an adversary to obtain $\alpha \cdot |\mathsf{sk}|$-bit leakage from a secret key sk, where $|\mathsf{sk}|$ denotes a bit-length of sk. In the second choice called the *absolute-leakage model*, we bound an absolute amount ℓ of leakage (which we call a *absolute-leakage-bound*), and we allow an adversary to obtain ℓ-bit leakage from a secret key. This model is especially useful when considering security against malware attacks, where an adversary persistently obtains some parts of secret key remotely. If ℓ is set to be very large (say, many gigabytes), it is difficult for such an adversary to obtain more than ℓ bits of a secret key. We note that any scheme in the relative-leakage model can be also seen as one in the absolute-leakage model. Suppose that one has a scheme resilient to leakage of leakage-ratio α in the relative-leakage model. We can obtain a scheme resilient to absolute-leakage-bound ℓ by simply increasing the security parameter so that $|\mathsf{sk}| > \alpha\ell$.

Bounded Retrieval Model. As seen above, a scheme in the relative-leakage model can also be seen as one in the absolute-leakage model by increasing the security parameter. However, this does not serve as a satisfactory solution considering efficiency. To increase an absolute-leakage-bound ℓ, we have to increase the security parameter, which means that the efficiency of the whole system becomes less efficient when ℓ is set larger. Considering a situation where we set ℓ to be extremely large, it is desirable that we can increase ℓ by just increasing the secret key size without affecting efficiencies of other parts (e.g., public key size, encryption-time, decryption-time in the case of PKE). This goal is usually referred to as the *bounded retrieval model* (BRM) [DLW06,Dzi06].

PKE and IBE in BRM. All known constructions of PKE and IBE schemes in the BRM follow the same template proposed by Alwen et al. [ADN+09]. Specifically, they introduced a primitive called *identity-based hash proof system* (IB-HPS), which is a generalization of a hash proof system [CS02], and gave a generic construction of PKE and IBE schemes in the BRM based on that. Moreover, they gave three concrete constructions of IB-HPS based on (1) truncated augmented bilinear Diffie-Hellman exponent (TABDHE) assumption, (2) learning with errors (LWE) assumption, and (3) quadratic residuosity (QR) assumption,

where the second and the third constructions are in the random oracle model.[1] As a result, they obtained PKE and IBE schemes in the BRM based on any of these assumptions. Leakage-ratios of these schemes are $1/2 - \epsilon$, $O(\frac{1}{\mathrm{poly}(\lambda)})$, and $1 - \epsilon$, respectively, where ϵ is an arbitrary constant. Subsequently, Chen et al. [CZLC16] constructed IB-HPSs based on the decisional bilinear Diffie-Hellman (DBDH) and the decisional square bilinear Diffie-Hellman (DSBDH) assumptions in the random oracle model. Based on them, one can construct PKE and IBE schemes in the BRM with leakage-ratio $1/2 - \epsilon$ for an arbitrary constant ϵ.

Hazay et al. [HLWW16] showed that, in fact, an IB-HPS is generically constructed from any IBE scheme.[2] As a result, one can construct PKE and IBE schemes in the BRM from any IBE scheme. However, one drawback of their construction is a poor leakage-ratio. Namely, the leakage-ratio of their scheme is $O(\frac{\log(\lambda)}{\mathrm{poly}(\lambda)})$. In that case, if one wants to set an absolute-leakage-bound to be ℓ, then a secret key size is $O(\frac{\mathrm{poly}(\lambda)}{\log(\lambda)} \cdot \ell)$, which is significantly larger than ℓ. Hopefully, we want to make the leakage-ratio close to 1 so that we can set a secret key size to be almost equal to ℓ for an absolute-leakage-bound ℓ. However, the only known construction of PKE and IBE schemes in the BRM that achieve such high leakage-ratio is the one based on the LWE assumption in the random oracle model. If one only relies on a standard assumption in the standard model, then the only known way to construct PKE and IBE schemes in the BRM is just instantiating the generic construction by Hazay et al. [HLWW16], which results in poor leakage-ratio $O(\frac{\log(\lambda)}{\mathrm{poly}(\lambda)})$. Thus the following problem remains open:

Is it possible to construct PKE and IBE schemes in the BRM whose leakage-ratio is almost equal to 1 *based on a standard assumption in the standard model?*

1.2 Our Contribution

We give a generic construction of IB-HPS based on any inner product encryption (IPE) scheme. As a result, we obtain PKE and IBE schemes in the BRM based on any IPE scheme. The leakage-ratio of our constructions is $\frac{n}{n+|\mathsf{sk}_{\mathsf{IPE}}(n)|}$ where n is an arbitrary integer and $|\mathsf{sk}_{\mathsf{IPE}}(n)|$ denotes a length of secret key of an underlying IPE scheme associated with an n-dimensional vector. In particular, if an underlying IPE scheme is fully key-compact (i.e., $|\mathsf{sk}_{\mathsf{IPE}}(n)|$ does not depend on n), then leakage-ratio can be made arbitrarily close to 1 by increasing n. For example, there are some known constructions of fully key-compact IPE schemes based on the d-linear (d-Lin) assumption [CGW15] and the subgroup-decision assumption on composite order pairing [Wee14] with adaptive security, and the learning-with-errors (LWE) assumption [AFV11] with selective security. Moreover,

[1] They can be proven secure in the standard model if one assumes non-standard interactive versions of these assumptions.

[2] In [HLWW16], IB-HPS is called *identity-based weak hash proof system* (IB-wHPS) for compatibility to their notion of weak hash proof system. We stress that IB-HPS in [ADN+09] and IB-wHPS in [HLWW16] mean completely the identical primitive.

Table 1. The "$|\mathsf{ct}|$" column shows ciphertext-length of IBE schemes in the BRM, Sel and Ad denote selective and adaptive securities, ϵ and δ are arbitrary constants larger than 0, λ denotes the security parameter, N denotes a composite number in underlying hard problems, $|\mathsf{sk}_{\mathsf{IBE}}|$ and $|\mathsf{ct}_{\mathsf{IBE}}|$ denote the length of a secret key and a ciphertext of an underlying IBE scheme, n denotes an arbitrary parameter supposed to be a dimension of vectors in IPE, $|\mathsf{sk}_{\mathsf{IPE}}|(n)$ and $|\mathsf{ct}_{\mathsf{IPE}}|(n)$ denotes the length of a secret key and a ciphertext of an underlying IPE scheme with dimension n, and ROM means the random oracle model.

Reference	leakage-ratio	$\lvert\mathsf{ct}\rvert$	Sel/Ad	Assumption
[ADN+09]	$\frac{1}{2}-\epsilon$	$O(\lambda^2)$	Ad	TABDHE
[ADN+09]	$\frac{1}{O(N)}$	$O(N)$	Ad	QR (ROM)
[ADN+09]	$1-\epsilon$	$O(\lambda^4)$	Ad	LWE (ROM)
[CZLC16]	$\frac{1}{2}-\epsilon$	$O(\lambda^2)$	Ad	DBDH (ROM)
[CZLC16]	$\frac{1}{2}-\epsilon$	$O(\lambda^2)$	Ad	DSBDH (ROM)
[HLWW16]	$(1-\epsilon)\frac{\log(\lambda)}{\lvert\mathsf{sk}_{\mathsf{IBE}}\rvert}$	$O(\lambda^2\lvert\mathsf{ct}_{\mathsf{IBE}}\rvert)$	Sel/Ad	Sel/Ad IBE
Ours	$(1-\epsilon)\frac{n}{n+\lvert\mathsf{sk}_{\mathsf{IPE}}(n)\rvert}$	$O(n\lambda\lvert\mathsf{ct}_{\mathsf{IPE}}(n)\rvert)$	Sel/Ad	Sel/Ad IPE
Ours + [CGW15]	$1-\epsilon$	$O(d^3\lambda^4)$	Ad	d-Lin
Ours + [Wee14]	$1-\epsilon$	$O(N^3\lambda)$	Ad	SD
Ours + [AFV11]	$1-\epsilon$	$\tilde{O}(\lambda^{4+\delta})$	Sel	LWE
Ours + Appendix A	$1-\epsilon$	$O(\lambda^4)$	Sel	CBDH

we give a construction of a fully key-compact selectively secure IPE scheme based on the computational bilinear Diffie-Hellman (CBDH) assumption. Each of these schemes gives new PKE and IBE schemes in the BRM model. In particular,

- We obtain the first PKE and selective/adaptive IBE schemes in the BRM whose leakage-ratio is arbitrarily close to 1 based on standard assumptions including d-Lin, LWE and CBDH assumptions in the standard model.
- Our CBDH-based construction is the first selectively secure IBE scheme whose leakage-ratio is arbitrarily close to 1 based on a search assumption on pairing groups even in the relative-leakage model where we allow the efficiency of a scheme to depend on the amount of leakage.

A comparison of IBE schemes in the BRM among known and our constructions is given in Table 1. We omit the comparison among PKE schemes in the BRM since all known constructions of PKE in the BRM are just degenerations of IBE in the BRM. We note that the selective security suffices for this degeneration.

1.3 Technical Overview

IB-HPS. We first roughly explain the definition of IB-HPS. An IB-HPS can be seen as an identity-based key encapsulation mechanism (IB-KEM) with a special "invalid encapsulation algorithm". It consists of a setup algorithm $\mathsf{Setup}(1^\lambda) \xrightarrow{\mathsf{R}} (\mathsf{pp}, \mathsf{msk})$, a key generation algorithm $\mathsf{KeyGen}(\mathsf{msk}, \mathsf{id}) \xrightarrow{\mathsf{R}} \mathsf{sk}_{\mathsf{id}}$, a valid encapsulation algorithm $\mathsf{Encap}(\mathsf{id}) \xrightarrow{\mathsf{R}} (\mathsf{ct}, \mathsf{k})$, an invalid encapsulation algorithm $\mathsf{Encap}^*(\mathsf{id}) \xrightarrow{\mathsf{R}} \mathsf{ct}$, and a decapsulation algorithm $\mathsf{Decap}(\mathsf{sk}_{\mathsf{id}}, \mathsf{id}, \mathsf{ct}) \xrightarrow{\mathsf{R}} \mathsf{k}$.

The correctness requires that a ciphertext generated by Encap is correctly decapsulated to the corresponding encapsulated key. A special feature of IB-HPS is that if we decapsulate an invalid ciphertext ct^* generated by Encap^* by a secret key $\mathsf{sk}_{\mathsf{id}}$, then the resulting key $\mathsf{k} \xleftarrow{R} \mathsf{Decap}(\mathsf{sk}_{\mathsf{id}}, \mathsf{id}, \mathsf{ct}^*)$ has a certain entropy given any fixed pp, id and ct^*. That is, there are many possible values of secret keys $\mathsf{sk}_{\mathsf{id}}$ for each id, and the value of $\mathsf{k} \xleftarrow{R} \mathsf{Decap}(\mathsf{sk}_{\mathsf{id}}, \mathsf{id}, \mathsf{ct}^*)$ depends on which $\mathsf{sk}_{\mathsf{id}}$ was used for the decapsulation. As security, we require that valid and invalid ciphertexts are computationally indistinguishable even if an adversary can obtain one secret key per identity for all identities including the challenge identity used for generating the ciphertext to distinguish.

IBE in BRM from IB-HPS. Alwen et al. [ADN+09] proved that we can construct a leakage resilient IBE scheme in the BRM based on any IB-HPS. The leakage-ratio of the resulting IBE scheme depends on the parameter called the *universality-ratio* of the underlying IB-HPS. Roughly speaking, the universality-ratio is defined to be $\frac{n}{|\mathsf{sk}_{\mathsf{id}}|}$ where 2^n is the number of possible $\mathsf{sk}_{\mathsf{id}}$ for each id and $|\mathsf{sk}_{\mathsf{id}}|$ denotes the bit-length of $\mathsf{sk}_{\mathsf{id}}$. They proved that the leakage-ratio of the resulting IBE scheme could be made arbitrarily close to the universality-ratio of the underlying IB-HPS. Thus, the problem of constructing IBE schemes in the BRM with high leakage-ratio is translated into the problem of constructing IB-HPS with high universality-ratio.

IB-HPS from any IBE. Here, we explain the idea of the work by Hazay et al. [HLWW16] that constructed an IB-HPS based on any IBE scheme. The setup algorithm of the IB-HPS (denoted by HPS) is the same as that of the IBE scheme and uses the same pp and msk. Let $\mathsf{Enc}_{\mathsf{IBE}}$ and $\mathsf{KeyGen}_{\mathsf{IBE}}$ denote the encryption and key generation algorithms of the underlying IBE scheme. Then, the key generation algorithm $\mathsf{KeyGen}_{\mathsf{HPS}}$, valid encapsulation algorithm $\mathsf{Enc}_{\mathsf{HPS}}$, and invalid encapsulation algorithm $\mathsf{Enc}^*_{\mathsf{HPS}}$ of HPS work as follows. In the description of $\mathsf{Enc}^*_{\mathsf{HPS}}$, differences from $\mathsf{Enc}_{\mathsf{HPS}}$ are highlighted in red letters.

$\mathsf{KeyGen}_{\mathsf{HPS}}(\mathsf{msk}, \mathsf{id})$: It picks $r \xleftarrow{R} \{0,1\}$, computes $\mathsf{sk}'_{\mathsf{id}} \xleftarrow{R} \mathsf{KeyGen}_{\mathsf{IBE}}(\mathsf{id}\|r)$, and sets $\mathsf{sk}_{\mathsf{id}} := (\mathsf{sk}'_{\mathsf{id}}, r)$. That is, $\mathsf{sk}_{\mathsf{id}}$ consists of secret keys for identities that are either $\mathsf{id}\|0$ or $\mathsf{id}\|1$, plus the random bit r that represents which identities were chosen.

$\mathsf{Enc}_{\mathsf{HPS}}(\mathsf{id})$: It picks $\mathsf{k} \in \{0,1\}$, computes $\mathsf{ct}_b \xleftarrow{R} \mathsf{Enc}_{\mathsf{IBE}}(\mathsf{id}\|b, \mathsf{k})$ for $b \in \{0,1\}$, and outputs a ciphertext $\mathsf{ct} := (\mathsf{ct}_0, \mathsf{ct}_1)$ and an encapsulated key k. That is, ct_0 and ct_1 encrypt the same value k under identities $\mathsf{id}\|0$ and $\mathsf{id}\|1$, respectively. The encapsulated key is defined to be k.

$\mathsf{Enc}^*_{\mathsf{HPS}}(\mathsf{id})$: It picks $\mathsf{k}_0, \mathsf{k}_1 \in \{0,1\}$ for $b \in \{0,1\}$, computes $\mathsf{ct}_b \xleftarrow{R} \mathsf{Enc}_{\mathsf{IBE}}(\mathsf{id}\|b, \mathsf{k}_b)$ for $b \in \{0,1\}$, and outputs a ciphertext $\mathsf{ct} := (\mathsf{ct}_0, \mathsf{ct}_1)$. That is, ct_0 and ct_1 encrypt independently random values k_0 and k_1 under identities $\mathsf{id}\|0$ and $\mathsf{id}\|1$, respectively. We note that this algorithm does not output an encapsulated key.

It is easy to see that the indistinguishability of valid and invalid ciphertexts can be reduced to the security of the underlying IBE scheme because an adversary

never obtains secret keys for identities $\mathsf{id}\|0$ and $\mathsf{id}\|1$ simultaneously.[3] A valid ciphertext generated by $\mathsf{Enc}_{\mathsf{HPS}}$ can be correctly decapsulated because either ct_0 or ct_1, both of which encapsulate the same key k, can be decrypted with $\mathsf{sk}_{\mathsf{id}}$. On the other hand, for an invalid ciphertext, ct_0 and ct_1 encapsulate independent keys k_0 and k_1. Therefore the decapsulation result depends on r that was used as randomness to generate a secret key. This means that the above IB-HPS has 2 different $\mathsf{sk}_{\mathsf{id}}$ for each id, and each of them decapsulate an invalid ciphertext to a different value.[4] However, since the size of $\mathsf{sk}_{\mathsf{id}}$ is $\mathrm{poly}(\lambda)$, the universality-ratio of the above IB-HPS is $\frac{1}{\mathrm{poly}(\lambda)}$, which is far from 1. They also showed that the universality-ratio can be improved to $O(\frac{\log(\lambda)}{\mathrm{poly}(\lambda)})$ by modifying the above scheme to choose r from $[\mathrm{poly}(\lambda)]$ instead of $\{0,1\}$ and modifying other algorithms accordingly. However, this is still far from optimal.

First Step: Parallel Repetition. As a first step to achieve higher universality-ratio, we consider a variant of the above IB-HPS via parallel repetition. Let $n \in \mathbb{N}$ be an arbitrarily chosen parameter and $\mathsf{bin}(i)$ denote a binary representation of i. The setup algorithm of the "n-parallel variant" (denoted by n-HPS) is the same as that of the IBE scheme, and use the same pp and msk. Then, the key generation algorithm $\mathsf{KeyGen}_{n\text{-}\mathsf{HPS}}$, valid encapsulation algorithm $\mathsf{Enc}_{n\text{-}\mathsf{HPS}}$, and invalid encapsulation algorithm $\mathsf{Enc}^*_{n\text{-}\mathsf{HPS}}$ of n-HPS as follows. In the description of $\mathsf{Enc}^*_{n\text{-}\mathsf{HPS}}$, differences from $\mathsf{Enc}_{n\text{-}\mathsf{HPS}}$ are highlighted in red letters.

$\mathsf{KeyGen}_{n\text{-}\mathsf{HPS}}(\mathsf{msk}, \mathsf{id})$: It picks $r_1, ..., r_n \xleftarrow{R} \{0,1\}$, computes $\mathsf{sk}'_{\mathsf{id},i} \xleftarrow{R} \mathsf{KeyGen}_{\mathsf{IBE}}(\mathsf{id}\|\mathsf{bin}(i)\|r_i)$ for $i \in [n]$, and outputs a secret key $\mathsf{sk}_{\mathsf{id}} := (\{\mathsf{sk}'_{\mathsf{id},i}\}_{i\in[n]}, \{r_i\}_{i\in[n]})$. That is, $\mathsf{sk}_{\mathsf{id}}$ consists of secret keys for identities that are either $\mathsf{id}\|\mathsf{bin}(i)\|0$ or $\mathsf{id}\|\mathsf{bin}(i)\|1$, plus random bits $\{r_i\}_{i\in[n]}$ that represent which identities were chosen.

$\mathsf{Enc}_{n\text{-}\mathsf{HPS}}(\mathsf{id})$: It picks $\mathsf{k}_1, ..., \mathsf{k}_n \in \{0,1\}$, computes $\mathsf{ct}_{i,b} \xleftarrow{R} \mathsf{Enc}_{\mathsf{IBE}}(\mathsf{id}\|\mathsf{bin}(i)\|b, \mathsf{k}_i)$ for $i \in [n]$ and $b \in \{0,1\}$, and outputs a ciphertext $\mathsf{ct} := \{\mathsf{ct}_{i,b}\}_{i\in[n],b\in\{0,1\}}$ and an encapsulated key $\mathsf{k} := \bigoplus_{i\in[n]} \mathsf{k}_i$. That is, $\mathsf{ct}_{i,0}$ and $\mathsf{ct}_{i,1}$ encrypt the same value k_i under identities $\mathsf{id}\|\mathsf{bin}(i)\|0$ and $\mathsf{id}\|\mathsf{bin}(i)\|1$, respectively, for each $i \in [n]$. The encapsulated key is defined to be $\mathsf{k} := \bigoplus_{i\in[n]} \mathsf{k}_i$.

$\mathsf{Enc}^*_{n\text{-}\mathsf{HPS}}(\mathsf{id})$: It picks $\mathsf{k}_{1,b}, ..., \mathsf{k}_{n,b} \in \{0,1\}$ for $b \in \{0,1\}$, computes $\mathsf{ct}_{i,b} \xleftarrow{R} \mathsf{Enc}_{\mathsf{IBE}}(\mathsf{id}\|\mathsf{bin}(i)\|b, \mathsf{k}_{i,b})$ for $i \in [n]$ and $b \in \{0,1\}$, and outputs a ciphertext $\mathsf{ct} := \{\mathsf{ct}_{i,b}\}_{i\in[n],b\in\{0,1\}}$. That is, $\mathsf{ct}_{i,0}$ and $\mathsf{ct}_{i,1}$ encrypt independently random values $\mathsf{k}_{i,0}$ and $\mathsf{k}_{i,1}$ under identities $\mathsf{id}\|\mathsf{bin}(i)\|0$ and $\mathsf{id}\|\mathsf{bin}(i)\|1$, respectively, for each $i \in [n]$. We note that this algorithm does not output an encapsulated key.

The indistinguishability of valid and invalid ciphertexts can be reduced to the security of the underlying IBE scheme similarly to the case for HPS. Next, we

[3] Here, it is crucial that an adversary obtains at most one secret key for each identity in the security model of IB-HPS.

[4] Here we assumed that $\mathsf{KeyGen}_{\mathsf{IBE}}$ is deterministic so that $\mathsf{sk}'_{\mathsf{id}}$ is determined by id. This can be assumed without loss of generality since we can derandomize $\mathsf{KeyGen}_{\mathsf{IBE}}$ by using a pseudorandom function.

calculate the universality-ratio of n-HPS. For each id, the number of possible $\mathsf{sk}_{\mathsf{id}}$ is 2^n since different $\{r_i\}_{i\in[n]}$ give different $\mathsf{sk}_{\mathsf{id}}$. On the other hand, $\mathsf{sk}_{\mathsf{id}}$ contains n secret keys of the underlying IBE scheme, each of them has a size of $\mathrm{poly}(\lambda)$. As a result, the universality-ratio of n-HPS is still $\frac{1}{\mathrm{poly}(\lambda)}$, which is even not better than that of HPS. Hence, to achieve better universality-ratio, we need an additional idea.

Our Idea: Compressing Secret Keys. As seen above, the reason for the poor universality-ratio of n-HPS is that a secret key of the scheme contains many secret keys of the underlying IBE scheme. Our idea is to compress them. Towards this goal, we introduce a notion called multi-identity-based encryption (MIBE). MIBE works similarly to IBE except that a secret key is associated with multiple identities, and the key can be used to decrypt a ciphertext that is encrypted under any of these identities. If we do not care about the size of a secret key, then it is trivial to construct an MIBE scheme from any IBE scheme: we can just let a secret key of the MIBE consist of a tuple of those of the IBE. The crucial property for our purpose is *key-compactness*, which means that the size of a secret key does not depend on the number of identities the key is associated with. With such a key-compact MIBE, the universality-ratio of n-HPS is dramatically improved because a secret key of the IB-HPS consists of a single secret key of MIBE whose size is $\mathrm{poly}(\lambda)$ that does not depend on n. Then, the universality-ratio is $\frac{n}{n+\mathrm{poly}(\lambda)}$. By increasing n, we can make it arbitrarily close to 1.

MIBE from IPE. The final challenge is to construct a key-compact MIBE. We show that a key-compact MIBE scheme can be constructed from any key-compact IPE scheme where key-compactness of an IPE scheme means that its secret key size does not depend on the dimension of the vector space. In an IPE scheme, a ciphertext and a secret key are associated with vectors $\boldsymbol{x}$ and $\boldsymbol{y}$ respectively, and the ciphertext is decryptable by the secret key if and only if $\boldsymbol{x}^T\boldsymbol{y} = 0$. Suppose that we have a key-compact IPE scheme with vector space $\mathbb{Z}_q^{n+1}$. We construct a key-compact MIBE scheme whose identity space is $\mathbb{Z}_q$ and secret key can be associated with n different identities as follows. To generate a secret key $\mathsf{sk}_{\mathsf{id}_1,...,\mathsf{id}_n}$ associated with a set $(\mathsf{id}_1, ..., \mathsf{id}_n)$ of identities, we first compute a vector $\boldsymbol{y} = (y_0, ..., y_n) \in \mathbb{Z}_q^{n+1}$ such that $\prod_{i=1}^{n}(X - \mathsf{id}_n) = \sum_{i=0}^{n} y_i X^i$ as a polynomial in the indeterminate X. The secret key $\mathsf{sk}_{\mathsf{id}_1,...,\mathsf{id}_n}$ is set to be a secret key associated with $\boldsymbol{y}$ of the underlying IPE scheme. To encrypt a message under an identity id^*, we encrypt the message under the vector $\boldsymbol{x} = (1, \mathsf{id}^*, (\mathsf{id}^*)^2, ..., (\mathsf{id}^*)^n)$ by the encryption algorithm of the underlying IPE. Since we have $\boldsymbol{x}^T\boldsymbol{y} = \sum_{i=0}^{n} y_i(\mathsf{id}^*)^i = \prod_{i=1}^{n}(\mathsf{id}^* - \mathsf{id}_n)$, we have $\boldsymbol{x}^T\boldsymbol{y} = 0$ if and only if $\mathsf{id}^* \in \{\mathsf{id}_1, ..., \mathsf{id}_n\}$. Therefore, this gives a construction of an MIBE scheme. We note that this construction is implicit in the work by Katz, Sahai and Waters [KSW08]. In the above construction, a secret key of the MIBE scheme consists of one secret key of the underlying IPE scheme. Therefore, if the underlying IPE scheme is key-compact, then the resulting MIBE is also key-compact. Finally, we note that IPE schemes with desirable key-compactness are known to exist based

on various standard assumptions. Putting everything together, we can construct a leakage resilient IBE scheme in the BRM with leakage-ratio arbitrarily close to 1 based on these standard assumptions.

1.4 Discussion

Notes on Efficiency of Our IBE. One may think that our scheme does not satisfy the definition of IBE in the BRM since the efficiency of our IBE scheme (including encryption time, decryption time, ciphertext size etc.) depends on the parameter n, which is a dimension of a vector space in the underlying IPE. However, our scheme actually satisfies the definition. This is because we *do not directly use* our IB-HPS itself as an IBE scheme, and we use the compiler by Alwen et al. [ADN+09] to *convert* our IB-HPS to IBE scheme (in the BRM). Since their compiler is general and applicable to any IB-HPS, we obtain an IBE scheme in the BRM. To explain this in more detail, we briefly recall their compiler. In their construction of an IBE scheme in the BRM, a "key-size parameter" m and a "locality-parameter" t are set appropriately,[5] and the public parameter is exactly the same as that of the underlying IB-HPS, a secret key for an identity id consists of secret keys for identities $\mathsf{id}\|\mathsf{bin}(1)...\mathsf{id}\|\mathsf{bin}(m)$ generated by the key generation algorithm of the underlying IB-HPS, and an encryption algorithm given a message m, randomly picks $\{r_1, ..., r_t\} \xleftarrow{R} [m]$, runs the encapsulation algorithm of the underlying IB-HPS under identities $\mathsf{id}\|\mathsf{bin}(r_1)...\mathsf{id}\|\mathsf{bin}(r_t)$ to obtain $(\mathsf{ct}_1, \mathsf{k}_1), ..., (\mathsf{ct}_t, \mathsf{k}_t)$, and outputs a ciphertext $(r_1, ..., r_t, \mathsf{ct}_1, ..., \mathsf{ct}_t, \mathsf{m} \oplus g(\mathsf{k}_1, ..., \mathsf{k}_t))$ where g is a universal hash function. We remark that the efficiency of the scheme (except the secret key size) just depends on t and *does not depend on* m. Their main theorem [ADN+09, Theorem 5.1] shows that we can increase an absolute-leakage-bound ℓ of the scheme just by increasing m and without increasing t, and the leakage-ratio of the IBE scheme is almost the same as the universality-ratio of the underlying IB-HPS. Thus, when we plug our IB-HPS from IPE (with a fixed dimension n) into their construction, we can arbitrarily increase an absolute-leakage-ratio ℓ just by increasing m *neither increasing* n *nor* t. Since what affect the efficiency of the IBE scheme is only n and t, and *not* m, we can increase an absolute-leakage-ratio ℓ without sacrificing the efficiency.

On Further Improving the Leakage-Ratio. In this paper, we propose IBE and PKE schemes in the BRM with leakage-ratio $1 - \epsilon$ for arbitrary constant $\epsilon > 0$. A natural question is if we can further achieve leakage-ratio $1 - \frac{1}{\mathrm{poly}(\lambda)}$ for any polynomial poly, which is optimal. The reason why we cannot achieve such a leakage-ratio is that we rely on Alwen et al.'s theorem [ADN+09] (Theorem 1), which gives an IBE scheme in the BRM with leakage-ratio $\beta(1 - \epsilon)$ where β is the universality-ratio of the underlying IB-HPS and $\epsilon > 0$ is an arbitrary constant. As long as we rely on this theorem, the resulting leakage-ratio cannot be better than $1 - \epsilon$ for constant $\epsilon > 0$. Though it seems that it is possible to

[5] In their paper, they use "n" instead of "m" for representing a "key-size" parameter. We use m for avoiding confusion with the dimension for IPE.

achieve leakage-ratio $1 - \frac{1}{\mathrm{poly}(\lambda)}$, by extending the theorem to treat the case of sub-constant ϵ, the analysis is rather complicated, and thus we simply rely on their theorem as a black-box to make the presentation of our results simpler. We note that if we consider schemes in the relative-leakage model where the efficiency of a scheme can depend on a leakage bound ℓ, then our constructions easily yield schemes with leakage-ratio $1 - \frac{1}{\mathrm{poly}(\lambda)}$.

1.5 Related Work

Here, we review existing works on leakage-resilient PKE and IBE schemes in other models. We remark that in all these models, the efficiency of schemes degrades with the leakage bound unlike ones in the BRM.

Leakage Resilient PKE/IBE in the Relative-leakage Model. We review existing works on leakage resilient PKE and IBE schemes in the relative-leakage model. Naor and Segev [NS12] proposed the first PKE scheme whose leakage resilience can be reduced to standard assumptions. Namely, they gave a generic construction of leakage-resilient PKE scheme based on a hash proof system. Subsequently, various constructions of leakage-resilient PKE schemes have been proposed [DHLW10b,BG10,HLWW16,BLSV18,QL13,QL14].

Chow et al. [CDRW10] proposed a leakage resilient IBE scheme based on the DBDH assumption with leakage-ratio $1/3 - o(1)$. Kurosawa and Phong [KP17] proposed leakage resilient IBE and IPE schemes based on the DLIN (2-Lin) and SXDH (1-Lin) assumptions with optimal leakage-ratio $1 - o(1)$ (they also constructed IBE and IPE schemes in an extended leakage model explained below, but its leakage-ratio is not optimal).

Continual Leakage Model. Brakerski et al. [BKKV10] and Dodis et al. [DHLW10a] concurrently introduced the notion of continual leakage model (CLM), where there is a notion of time periods and secret information is updated at the end of each time period. Adversaries are allowed to obtain a limited amount of secret information in each time period, but there is no limitation on the total amount of information that they obtained in all time periods. Brakerski et al. constructed PKE, IBE, and signature schemes from the DLIN or SXDH assumptions in the CLM. Dodis et al. constructed signature and identification schemes and authenticated key agreement protocols from the d-Lin assumption in the CLM.

Subsequently, Lewko et al. [LRW11] constructed adaptively secure IBE and attribute-based encryption (ABE) schemes based on the subgroup decision assumption in the CLM. In their scheme, adversaries are allowed to obtain leakage even from *master-secret keys*. Yu et al. [YAX+16] constructed adaptively secure ABE schemes for wider classes of functionality based on composite-order pairing groups in the CLM. Zhang et al. [ZCG+18] constructed adaptively secure ABE schemes for wider classes of functionality based on prime-order pairing groups (the d-Lin assumption) in the CLM.

Hard-to-invert Leakage. Dodis et al. [DKL09] introduced the notion of cryptography with hard-to-invert auxiliary inputs, where adversaries are given auxiliary input $h(s)$ such that it is computationally hard to find s from $h(s)$ (s is secret information). Dodis et al. [DKL09] constructed symmetric encryption schemes from a non-standard variant of the learning parity with noise assumption in that model. Dodis et al. [DGK+10] constructed PKE schemes from the DDH or LWE assumption in that model. Yuen et al. [YCZY12] considered IBE schemes in an extended leakage model that is a combination of the CLM and hard-to-invert auxiliary input model.

2 Preliminaries

2.1 Notations

For any natural number n, $[n]$ denotes the set $\{1, \ldots, n\}$. $x \xleftarrow{R} S$ denotes x is randomly chosen from a finite set S, and $y \xleftarrow{R} \mathcal{A}(x; r)$ denotes that y is an output of a randomized algorithm $\mathcal{A}$ with input x and randomness r. We say that a function $f(\cdot) : \mathbb{N} \to [0, 1]$ is negligible if for all positive polynomials $p(\cdot)$ and all sufficiently large $\lambda \in \mathbb{N}$, we have $f(\lambda) < 1/p(\lambda)$. We say that an algorithm $\mathcal{A}$ is probabilistic polynomial time (PPT) if there exists a polynomial p such that a running time of $\mathcal{A}$ with input length λ is less than $p(\lambda)$. For a bit string x, $|x|$ denotes the bit-length of x. The min-entropy of a random variable X is $\mathsf{H}_\infty(X) \coloneqq -\log(\max_x \Pr[X = x])$. We often denote poly to mean an unspecified polynomial and negl to mean an unspecified negligible function.

2.2 Pseudorandom Function

Definition 1. *An deterministic function* $\mathsf{PRF} : \mathcal{K} \times \mathcal{D} \to \mathcal{R}$ *computable in polynomial time is said to be a pseudorandom function (PRF) if for any PPT adversary* $\mathcal{A}$*,*

$$\mathsf{Adv}_{\mathsf{PRF},\mathcal{A}}(\lambda) \coloneqq |\Pr[1 \leftarrow \mathcal{A}^{\mathsf{PRF}(K\cdot)}(1^\lambda)] - \Pr[1 \leftarrow \mathcal{A}^{\mathsf{Rand}(\cdot)}(1^\lambda)]|$$

is negligible where $K \xleftarrow{R} \mathcal{K}$ *and* $\mathsf{Rand} \xleftarrow{R} \mathcal{F}(\mathcal{D}, \mathcal{R})$ *where* $\mathcal{F}(\mathcal{D}, \mathcal{R})$ *denotes the set of all functions from* $\mathcal{D}$ *to* $\mathcal{R}$*.*

2.3 Identity-based Encryption

We define IBE, and its leakage-resilient security (in the bounded retrieval model). An IBE scheme consists of the following algorithms.

$\mathsf{Setup}(1^\lambda, 1^\ell) \xrightarrow{R} (\mathsf{pp}, \mathsf{msk})$: This is the setup algorithm that takes the security parameter 1^λ and the leakage parameter 1^ℓ as input [6] and outputs a public parameter pp and a master secret key msk. All other algorithms implicitly include pp as an input.

[6] Since we consider a leakage resilient IBE, we give the leakage parameter 1^ℓ as input, which means a maximum amount of leakage bits the scheme tolerates.

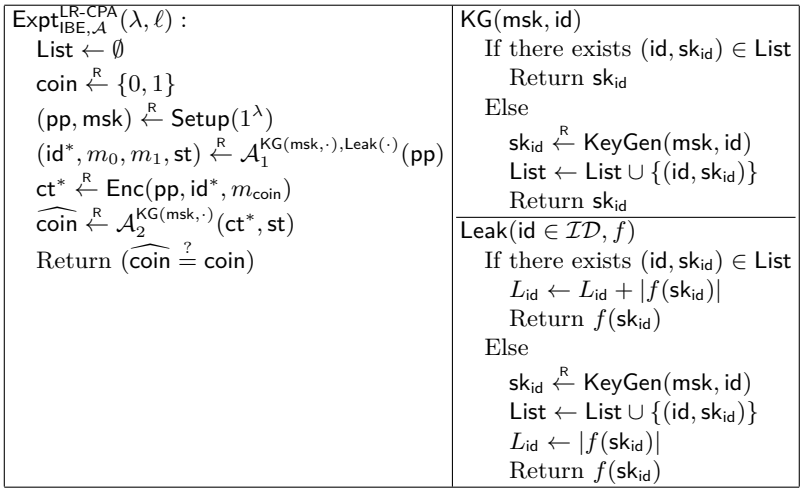

Fig. 1. The experiment for defining the leakage-resilience for IBE

$\mathsf{KeyGen}(\mathsf{msk},\mathsf{id}) \xrightarrow{R} \mathsf{sk}_{\mathsf{id}}$**:** This is the key generation algorithm that takes a master secret key msk and an identity id as input, and outputs a secret key $\mathsf{sk}_{\mathsf{id}}$ associated with the identity id.

$\mathsf{Enc}(\mathsf{id},\mathsf{m}) \xrightarrow{R} \mathsf{ct}$**:** This is the encryption algorithm that takes an identity id and a message m, and outputs a ciphertext ct.

$\mathsf{Dec}(\mathsf{sk}_{\mathsf{id}},\mathsf{id},\mathsf{ct}) \to \mathsf{m}$**:** This is the decryption algorithm that takes a secret key $\mathsf{sk}_{\mathsf{id}}$, an identity id and a ciphertext ct as input, and outputs a message m.

Remark 1. In our definition, we explicitly give id to Dec as an input, which differs from a commonly-used definition. We define in this way because id need not be hidden, and thus it is natural to separate it from a secret key. We note that this modification does not lose any generality because we can simply include id in $\mathsf{sk}_{\mathsf{id}}$. This modification slightly affects the leakage-ratio defined below, but the difference is negligible when $\ell = \omega(|\mathsf{id}|)$.

Correctness. For any $(\mathsf{pp},\mathsf{msk})$ produced by $\mathsf{Setup}(1^\lambda, 1^\ell)$, any $\mathsf{id} \in \mathcal{ID}$, any $\mathsf{m} \in \mathcal{M}$, we have

$$\Pr\left[\mathsf{m} \neq \mathsf{m}' \,\middle|\, \begin{array}{l} \mathsf{sk}_{\mathsf{id}} \xleftarrow{R} \mathsf{KeyGen}(\mathsf{msk},\mathsf{id}), \\ \mathsf{ct} \xleftarrow{R} \mathsf{Enc}(\mathsf{id},\mathsf{m}), \mathsf{m}' := \mathsf{Dec}(\mathsf{sk}_{\mathsf{id}},\mathsf{id},\mathsf{ct}) \end{array}\right] = \mathrm{negl}(\lambda)$$

Leakage-resilience. Leakage resilience of an IBE scheme IBE is defined by the experiment $\mathsf{Expt}^{\mathsf{LR\text{-}CPA}}_{\mathsf{IBE},\mathcal{A}}(\lambda,\ell)$ for an adversary $\mathcal{A} = (\mathcal{A}_1,\mathcal{A}_2)$ described in Figure 1. We say that a PPT adversary $\mathcal{A}$ is admissible if it does not query id^* to $\mathsf{KG}(\mathsf{msk},\cdot)$, and at the end of the experiment, we have $L_{\mathsf{id}^*} \leq \ell$ or L_{id^*} is undefined (i.e., $\mathcal{A}$ never queries id^* to Leak). We say that an PPT adversary $\mathcal{A}$ is selectively admissible if in addition to the above, $\mathcal{A}_1$ can be divided into two stages $\mathcal{A}_{1\text{-}1}$ and $\mathcal{A}_{1\text{-}2}$: $\mathcal{A}_{1\text{-}1}$ is given 1^λ and not allowed to access to any oracle, and returns

$(\mathsf{id}^*, \mathsf{st}_{\mathsf{pre}})$, and $\mathcal{A}_{1\text{-}2}$ is given $(\mathsf{pp}, \mathsf{st}_{\mathsf{pre}})$ and allowed to access to oracles $\mathsf{KG}(\mathsf{msk}, \cdot)$ and $\mathsf{Leak}(\cdot, \cdot)$, and returns $(\mathsf{m}_0, \mathsf{m}_1, \mathsf{st})$.

Definition 2. *We say that an IBE scheme* IBE *is adaptively leakage resilient if for any polynomial* $\ell(\lambda)$*, any admissible adversary* $\mathcal{A}$*, if the advantage*

$$\mathsf{Adv}^{\mathsf{LR\text{-}CPA}}_{\mathsf{IBE},\mathcal{A}}(\lambda, \ell) := 2 \cdot |\Pr[\mathsf{Expt}^{\mathsf{LR\text{-}CPA}}_{\mathsf{IBE},\mathcal{A}}(\lambda, \ell) = 1] - 1/2|$$

is negligible in λ*. We define selective leakage resilience of IBE analogously by replacing "any admissible adversary* $\mathcal{A}$*" in the above definition with "any selectively admissible adversary* $\mathcal{A}$*". We define leakage-ratio* α *of the scheme to be minimal value of* $\frac{\ell}{|\mathsf{sk}_{\mathsf{id}}|}$ *where* $(\mathsf{pp}, \mathsf{msk}) \xleftarrow{R} \mathsf{Setup}(1^\lambda, 1^\ell)$ *and* $\mathsf{sk}_{\mathsf{id}} \xleftarrow{R} \mathsf{KeyGen}(\mathsf{msk}, \mathsf{id})$*.*

Remark 2. In our security model, we assume that an adversary obtains a leakage from one decryption key per one identity, and cannot obtain a leakage from a master secret key. This is the same model as the ones in [CDRW10,ADN+09,KP17]. Some works (e.g., [LRW11]) consider stronger security models where an adversary obtains leakages from many secret keys of the same identity and leakages from a master secret key.

Bounded Retrieval Model.

Next, we define leakage resilient IBE in the BRM [ADN+09][7].

Definition 3. *([ADN+09, Def. 6.2]) We say that an IBE scheme is adaptively (resp. selectively) leakage-resilient in the bounded retrieval model (BRM), if the scheme is adaptively (resp. selectively) leakage-resilient, and the public parameter size, master secret key size, ciphertext size, encryption time, and decryption time (and the number of secret key bits read by decryption) are independent of the leakage bound* ℓ*. More formally, there exist polynomials* ppsize*,* $\mathsf{msksize}$*,* ctsize*,* encT*,* decT*, such that, for any polynomial* ℓ *and any* $(\mathsf{pp}, \mathsf{msk}) \xleftarrow{R} \mathsf{KeyGen}(1^\lambda, 1^\ell)$*,* $\mathsf{id} \in \mathcal{ID}$*,* $\mathsf{m} \in \mathcal{M}$*,* $\mathsf{ct} \xleftarrow{R} \mathsf{Enc}(\mathsf{id}, m)$*, the scheme satisfies:*

1. *Public parameter size is* $|\mathsf{pp}| \leq O(\mathsf{ppsize}(\lambda))$*, master secret key size is* $|\mathsf{msk}| \leq O(\mathsf{msksize}(\lambda))$*, ciphertext size is* $|\mathsf{ct}| \leq O(\mathsf{ctsize}(\lambda, |\mathsf{m}|))$*.*
2. *Run-time of* $\mathsf{Enc}(\mathsf{id}, \mathsf{m})$ *is* $\leq O(\mathsf{encT}(\lambda, |\mathsf{m}|))$*.*
3. *Run-time of* $\mathsf{Dec}(\mathsf{ct}, \mathsf{sk})$*, and the number of bits of* sk *accessed, is* $\leq O(\mathsf{decT}(\lambda, |\mathsf{m}|))$*.*

2.4 Inner Product Encryption

We define inner product encryption (IPE) and its security. We remark that we do not define the leakage resilience for IPE because we do not construct a leakage resilient IPE scheme, and we just use a (non-leakage resilient) IPE scheme as a building block to construct a leakage resilient IBE scheme in the BRM. An IPE scheme consists of the following algorithms.

[7] In [ADN+09], they only consider the adaptive security. We also define the selective security similarly.

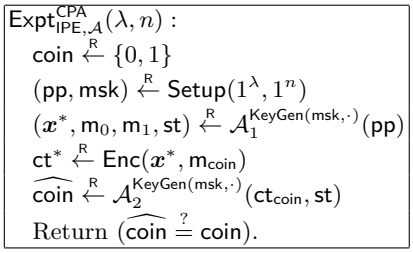

Fig. 2. The experiment for defining the security for IPE

$\mathsf{Setup}(1^\lambda, 1^n) \xrightarrow{R} (\mathsf{pp}, \mathsf{msk})$: This is the setup algorithm that takes the security parameter 1^λ and the vector-dimension 1^n as input and outputs a public parameter pp and a master secret key msk. The public parameter pp specifies a vector space $\mathbb{Z}_q^n$. All other algorithms implicitly include pp as an input.

$\mathsf{KeyGen}(\mathsf{msk}, \boldsymbol{y} \in \mathbb{Z}_q^n) \xrightarrow{R} \mathsf{sk}_{\boldsymbol{y}}$**:** This is the key generation algorithm that takes a master secret key msk and a vector $\boldsymbol{y} \in \mathbb{Z}_q^n$ as input, and outputs a secret key $\mathsf{sk}_{\boldsymbol{y}}$ associated with the vector $\boldsymbol{y}$.

$\mathsf{Enc}(\boldsymbol{x}, \mathsf{m}) \xrightarrow{R} \mathsf{ct}$**:** This is the encryption algorithm that takes a vector $\boldsymbol{x} \in \mathbb{Z}_q^n$ and a message m, and outputs a ciphertext ct.

$\mathsf{Dec}(\mathsf{sk}_{\boldsymbol{y}}, \boldsymbol{y}, \mathsf{ct}) \to \mathsf{m}$**:** This is the decryption algorithm that takes a secret key $\mathsf{sk}_{\boldsymbol{y}}$, a vector $\boldsymbol{y}$ and a ciphertext ct as input, and outputs a message m.

Correctness. For any $(\mathsf{pp}, \mathsf{msk})$ produced by $\mathsf{Setup}(1^\lambda, 1^n)$, any $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{Z}_q^n$ such that $\boldsymbol{x}^T \cdot \boldsymbol{y} = 0$, any $\mathsf{m} \in \mathcal{M}$, we have

$$\Pr\left[\mathsf{m} \neq \mathsf{m}' \,\middle|\, \begin{array}{l} \mathsf{sk}_{\boldsymbol{y}} \xleftarrow{R} \mathsf{KeyGen}(\mathsf{msk}, \boldsymbol{y}), \\ \mathsf{ct} \xleftarrow{R} \mathsf{Enc}(\boldsymbol{x}, \mathsf{m}), \mathsf{m}' := \mathsf{Dec}(\mathsf{sk}_{\boldsymbol{y}}, \boldsymbol{y}, \mathsf{ct}) \end{array}\right] = \mathrm{negl}(\lambda)$$

Security. Security of an IPE scheme IPE is defined by the experiment $\mathsf{Expt}^{\mathsf{CPA}}_{\mathsf{IPE},\mathcal{A}}(\lambda, n)$ for an adversary $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$ described in Figure 2. We say that a PPT adversary $\mathcal{A}$ is admissible if it does not query $\boldsymbol{y}$ satisfying $(\boldsymbol{x}^*)^T \cdot \boldsymbol{y} = 0$ to $\mathsf{KeyGen}(\mathsf{msk}, \cdot)$. We say that a PPT adversary $\mathcal{A}$ is selectively admissible if in addition to the above, $\mathcal{A}_1$ can be divided into two stages $\mathcal{A}_{1\text{-}1}$ and $\mathcal{A}_{1\text{-}2}$: $\mathcal{A}_{1\text{-}1}$ is given $(1^\lambda, 1^n)$ and not allowed to access to any oracle, and returns $(\boldsymbol{x}^*, \mathsf{st}_{\mathsf{pre}})$, and $\mathcal{A}_{1\text{-}2}$ is given $(\mathsf{pp}, \mathsf{st}_{\mathsf{pre}})$ and allowed to access to oracles $\mathsf{KeyGen}(\mathsf{msk}, \cdot)$, and returns $(\mathsf{m}_0, \mathsf{m}_1, \mathsf{st})$.

Definition 4. *We say that an IPE scheme* IPE *is adaptively secure if for any polynomial* $n(\lambda)$*, any admissible adversary* $\mathcal{A}$*, if the advantage*

$$\mathsf{Adv}^{\mathsf{CPA}}_{\mathsf{IPE},\mathcal{A}}(\lambda, n) := 2 \cdot |\Pr[\mathsf{Expt}^{\mathsf{CPA}}_{\mathsf{IBE},\mathcal{A}}(\lambda, n) = 1] - 1/2|$$

is negligible in λ*. We define selective security of IPE analogously by replacing "any admissible adversary* $\mathcal{A}$*" in the above definition with "any selectively admissible adversary* $\mathcal{A}$*".*

Key-compactness. We say that an IPE scheme is fully key-compact if for any polynomial $n = n(\lambda)$, any $(\mathsf{pp}, \mathsf{msk})$ produced by $\mathsf{Setup}(1^\lambda, 1^n)$, any $\boldsymbol{y} \in \mathbb{Z}_q^n$, and any $\mathsf{sk}_{\boldsymbol{y}}$ produced by $\mathsf{KeyGen}(\mathsf{msk}, \boldsymbol{y})$, we have

$$|\mathsf{sk}_{\boldsymbol{y}}| = \mathrm{poly}(\lambda)$$

where poly is a fixed polynomial that does not depend on n.

2.5 Identity-based Hash Proof System (IB-HPS)

An identity-based hash proof system (IB-HPS) [ADN+09] consists of five PPT algorithms $\Pi = (\mathsf{Setup}, \mathsf{KeyGen}, \mathsf{Encap}, \mathsf{Encap}^*, \mathsf{Decap})$.

$\mathsf{Setup}(1^\lambda)$: This is the setup algorithm that takes the security parameter 1^λ as an input, and outputs a public parameter pp and a master secret key msk. All other algorithms implicitly include pp as an input.

$\mathsf{KeyGen}(\mathsf{msk}, \mathsf{id})$: This is the key generation algorithm that takes a master secret key msk and an identity id as inputs, and outputs a identity secret key $\mathsf{sk}_{\mathsf{id}}$.

$\mathsf{Encap}(\mathsf{id})$: This is the valid encapsulation algorithm that takes an identity id as an input and outputs a valid ciphertext ct and a encapsulated key k.

$\mathsf{Encap}^*(\mathsf{id})$: This is the invalid encapsulation algorithm that takes an identity id as an input and outputs an invalid ciphertext ct'.

$\mathsf{Decap}(\mathsf{sk}_{\mathsf{id}}, \mathsf{id}, \mathsf{ct})$: This is the decapsulation algorithm that takes an identity secret key $\mathsf{sk}_{\mathsf{id}}$, an identity id and a ciphertext ct as inputs, and outputs an encapsulated key k.

We require that an IB-HPS satisfies the following properties.

Correctness. For any $(\mathsf{pp}, \mathsf{msk})$ produced by $\mathsf{Setup}(1^\lambda)$, any $\mathsf{id} \in \mathcal{ID}$, we have

$$\Pr\left[\mathsf{k} \neq \mathsf{k}' \;\middle|\; \begin{array}{l} \mathsf{sk}_{\mathsf{id}} \xleftarrow{R} \mathsf{KeyGen}(\mathsf{msk}, \mathsf{id}) \\ (\mathsf{ct}, \mathsf{k}) \xleftarrow{R} \mathsf{Encap}(\mathsf{id}), \mathsf{k}' := \mathsf{Decap}(\mathsf{sk}_{\mathsf{id}}, \mathsf{id}, \mathsf{ct}) \end{array}\right] = \mathrm{negl}(\lambda)$$

Valid/invalid ciphertext indistinguishability. The valid ciphertexts generated by Encap and the invalid ciphertexts generated by Encap^* should be indistinguishable even given a secret key of a challenge identity. In particular, we define an experiment $\mathsf{Expt}^{\mathsf{ind}}_{\Pi,\mathcal{A}}$ for an IB-HPS Π and an adversary $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$ as described in Figure 3.

We say that a PPT adversary $\mathcal{A}$ is admissible if it does not makes the same query to $\mathsf{KeyGen}(\mathsf{msk}, \cdot)$ twice. We say that a PPT adversary $\mathcal{A}$ is selectively admissible if in addition to that, $\mathcal{A}_1$ is given 1^λ instead of pp and not allowed to access to $\mathsf{KeyGen}(\mathsf{msk}, \cdot)$.

$\mathcal{A}_1$ is only given 1^λ instead of pp and does not make any query. We note that we do not prohibit an adversary from querying id^*. That is, valid and invalid ciphertexts under an identity id^* are indistinguishable even if an adversary is given one secret key that corresponds to id^*.

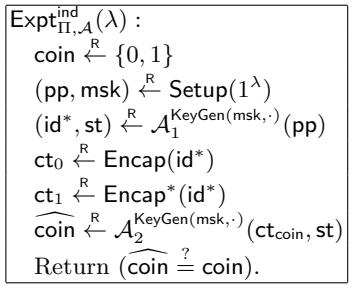

Fig. 3. The experiment for defining valid/invalid ciphertext indistinguishability for IB-HPS

Definition 5. *We say that an IB-HPS* Π *is adaptively secure if for any admissible adversary* $\mathcal{A}$*, the advantage* $\mathsf{Adv}^{\mathsf{ind}}_{\Pi,\mathcal{A}}(\lambda) := 2 \cdot |\Pr[\mathsf{Expt}^{\mathsf{ind}}_{\Pi,\mathcal{A}}(\lambda) = 1] - 1/2|$ *is negligible. We define selective security of IB-HPS analogously by replacing "any PPT adversary* $\mathcal{A}$*" in the above definition with "any selectively admissible adversary* $\mathcal{A}$*".*

Universality. Another property of IB-HPS is *universality*. An IB-HPS is said to be (n, ρ)-universal if the number of possible values of $\mathsf{sk}_{\mathsf{id}} \xleftarrow{R} \mathsf{KeyGen}(\mathsf{msk}, \mathsf{id})$ is larger than 2^n, and any distinct pair of them decrypts a randomly generated invalid ciphertext to the same message with probability at most ρ. In other words, $\{\mathsf{Decap}(\mathsf{ct}, \mathsf{id}, \cdot) : \mathsf{ct} \xleftarrow{R} \mathsf{Enc}^*(\mathsf{pp})\}$ is a family of ρ-universal functions.

Definition 6. *[ADN+09, Def. 3.1] We say that an IB-HPS* Π *is* (n, ρ)*-universal if for any fixed values of* $(\mathsf{pp}, \mathsf{msk})$ *produced by* $\mathsf{Setup}(1^\lambda)$*,* $\mathsf{id} \in \mathcal{ID}$*, the following hold:*

1. $\mathsf{H}_\infty(\mathsf{sk}_{\mathsf{id}}) \geq n$ *where* $\mathsf{sk}_{\mathsf{id}} \xleftarrow{R} \mathsf{KeyGen}(\mathsf{msk}, \mathsf{id})$.
2. *For any fixed distinct* $\mathsf{sk}_{\mathsf{id}} \neq \overline{\mathsf{sk}}_{\mathsf{id}}$ *produced by* $\mathsf{KeyGen}(\mathsf{msk}, \mathsf{id})$*,*

$$\Pr_{\mathsf{ct} \xleftarrow{R} \mathsf{Encap}^*(\mathsf{id})} [\mathsf{Decap}(\mathsf{sk}_{\mathsf{id}}, \mathsf{id}, \mathsf{ct}) = \mathsf{Decap}(\overline{\mathsf{sk}}_{\mathsf{id}}, \mathsf{id}, \mathsf{ct})] \leq \rho.$$

We say that Π *has a universality-ratio* β *if there exists* n *and a constant* $\rho < 1$ *such that* Π *is* (n, ρ)*-universal and we have* $\beta < \frac{n}{|\mathsf{sk}_{\mathsf{id}}|}$ *for any* $(\mathsf{pp}, \mathsf{msk})$ *produced by* $\mathsf{Setup}(1^\lambda)$*, any* id *and any* $\mathsf{sk}_{\mathsf{id}}$ *produced by* $\mathsf{KeyGen}(\mathsf{msk}, \mathsf{id})$*.*

Alwen et al. [ADN+09] gave a construction of an IBE scheme in the BRM based on an (n, ρ)-universal IB-HPS, and prove that the leakage-ratio α of their IBE scheme can be arbitrarily close to the universality-ratio β of an underlying IB-HPS. More formally, they proved the following theorem [8].

Theorem 1. *([ADN+09, Theorem 6.1].) If there exists an adaptively (resp. selectively) secure IB-HPS with universality-ratio* $\beta > c$ *for some constant* c*,*

[8] Though Alwen et al. [ADN+09] only gave a proof for the case of the adaptive security, the proof can be straightforwardly extended to the selective case.

then for any constant $\epsilon > 0$ and any polynomial v, there exists an adaptively (resp. selectively) leakage-resilient IBE scheme in the BRM with message space $\mathcal{M} = \{0,1\}^v$ and:

1. *Master public/secret key size is the same as that of the underlying IB-HPS.*
2. *Ciphertext-size/encryption-time/decryption-time are $t = O(v + \lambda)$ times larger than that of the underlying IB-HPS.*
3. *Leakage-ratio is $\alpha \geq \beta(1 - \epsilon)$ for sufficiently large values of the leakage parameter ℓ.*

3 Generic Construction of IB-HPS from IPE

In this section, we give a generic construction of IB-HPS based on any IPE scheme. Interestingly, universality-ratio of a resulting IB-HPS is related to key-compactness of an underlying IPE scheme. Especially, if an underlying IPE is fully key-compact, then the universality-ratio of the resulting IB-HPS can be arbitrarily close to 1. We give our construction through an intermediate primitive called *multi-identity-based encryption* (MIBE).

3.1 Multi-Identity-based Encryption

Here, we introduce a notion of MIBE, which is a variant of IBE such that a secret key is associated with multiple identities. Then we show a key-compactness-preserving conversion from IPE to MIBE. An MIBE scheme consists of four PPT algorithms $(\mathsf{Setup}, \mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$.

$\mathsf{Setup}(1^\lambda, 1^n) \xrightarrow{R} (\mathsf{pp}, \mathsf{msk})$: This is the setup algorithm that takes the security parameter 1^λ and the identity-multiplicity 1^n as inputs and outputs a public parameter pp and a master secret key msk. All other algorithms implicitly include pp as an input.

$\mathsf{KeyGen}(\mathsf{msk}, (\mathsf{id}_1, ..., \mathsf{id}_n)) \xrightarrow{R} \mathsf{sk}_{(\mathsf{id}_1,...,\mathsf{id}_n)}$**:** This is the key generation algorithm that takes a master secret key msk and identities $\mathsf{id}_1, ..., \mathsf{id}_n$ as inputs, and outputs secret key $\mathsf{sk}_{(\mathsf{id}_1,...,\mathsf{id}_n)}$ associated with the set $\{\mathsf{id}_1, ..., \mathsf{id}_n\}$.

$\mathsf{Enc}(\mathsf{id}, \mathsf{m}) \xrightarrow{R} \mathsf{ct}$**:** This is the encryption algorithm that takes an identity id and a message m as inputs, and outputs a ciphertext ct.

$\mathsf{Dec}(\mathsf{sk}_{\mathsf{id}_1,...,\mathsf{id}_n}, (\mathsf{id}_1, ..., \mathsf{id}_n), \mathsf{ct}) \to \mathsf{m}$**:** This is the decryption algorithm that takes a secret key $\mathsf{sk}_{(\mathsf{id}_1,...,\mathsf{id}_n)}$, a set of identities $(\mathsf{id}_1, ..., \mathsf{id}_n)$ and a ciphertext ct as inputs, and outputs a message m.

Correctness. For any $(\mathsf{pp}, \mathsf{msk})$ produced by $\mathsf{Setup}(1^\lambda)$, any $n \in \mathbb{N}$, any $\mathsf{id}_1, ..., \mathsf{id}_n \in \mathcal{ID}^n$, any $i \in [n]$, and any message m, we have

$$\Pr\left[\mathsf{m} \neq \mathsf{m}' \;\middle|\; \begin{array}{l} \mathsf{sk}_{(\mathsf{id}_1,...,\mathsf{id}_n)} \xleftarrow{R} \mathsf{KeyGen}(\mathsf{msk}, (\mathsf{id}_1, ..., \mathsf{id}_n)) \\ (\mathsf{ct}, \mathsf{k}) \xleftarrow{R} \mathsf{Enc}(\mathsf{id}_i, \mathsf{m}), \mathsf{m}' := \mathsf{Dec}(\mathsf{sk}_{(\mathsf{id}_1,...,\mathsf{id}_n)}, (\mathsf{id}_1, ..., \mathsf{id}_n), \mathsf{ct}) \end{array}\right] = \mathrm{negl}(\lambda)$$

Security. The security of an MIBE scheme MIBE is defined by the experiment

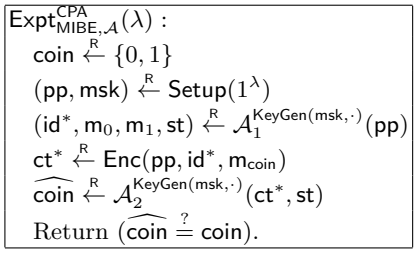

Fig. 4. The experiment for defining the security for MIBE

$\mathsf{Expt}^{\mathsf{CPA}}_{\mathsf{MIBE},\mathcal{A}}(\lambda)$ for an adversary $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$ described in Figure 4. We say that a PPT adversary $\mathcal{A}$ is admissible if for any query $(\mathsf{id}_1, ..., \mathsf{id}_n)$ made by $\mathcal{A}$, we have $\mathsf{id}^* \notin \{\mathsf{id}_1, ..., \mathsf{id}_n\}$. We say that $\mathcal{A}$ is selectively admissible if in addition to the above, $\mathcal{A}_1$ can be divided into two stages $\mathcal{A}_{1\text{-}1}$ and $\mathcal{A}_{1\text{-}2}$: $\mathcal{A}_{1\text{-}1}$ is given 1^λ and not allowed to access to any oracle, and returns $(\mathsf{id}^*, \mathsf{st}_{\mathsf{pre}})$, and $\mathcal{A}_{1\text{-}2}$ is given $(\mathsf{pp}, \mathsf{st}_{\mathsf{pre}})$ and allowed to access to oracles $\mathsf{KeyGen}(\mathsf{msk}, \cdot)$, and returns $(\mathsf{m}_0, \mathsf{m}_1, \mathsf{st})$.

Definition 7. *We say that a MIBE scheme* MIBE *is adaptively secure if for any admissible adversary $\mathcal{A}$, if the advantage*

$$\mathsf{Adv}^{\mathsf{CPA}}_{\mathsf{MIBE},\mathcal{A}}(\lambda) \coloneqq 2 \cdot |\Pr[\mathsf{Expt}^{\mathsf{CPA}}_{\mathsf{MIBE},\mathcal{A}}(\lambda) = 1] - 1/2|$$

is negligible. We define selective security of MIBE analogously by replacing "any admissible adversary $\mathcal{A}$" in the above definition with "any selectively admissible adversary $\mathcal{A}$".

Key-compactness. We say that an MIBE scheme is fully key-compact if for any polynomial $n = n(\lambda)$, any $(\mathsf{pp}, \mathsf{msk})$ produced by $\mathsf{Setup}(1^\lambda, 1^n)$, any $\mathsf{id}_1, ..., \mathsf{id}_n$, and $\mathsf{sk}_{\mathsf{id}_1,...,\mathsf{id}_n}$ produced by $\mathsf{KeyGen}(\mathsf{msk}, (\mathsf{id}_1, ..., \mathsf{id}_n))$, we have

$$|\mathsf{sk}_{\mathsf{id}_1,...,\mathsf{id}_n}| = \mathrm{poly}(\lambda)$$

where poly is a fixed polynomial that does not depend on n.

Remark 3. If we do not require the key-compactness, it is trivial to construct an MIBE scheme from any IBE scheme.

3.2 MIBE from IPE

Here, we give a key-compactness-preserving construction of an MIBE scheme based on an IPE scheme. Actually, this construction is implicit in the work by Katz, Sahai and Waters [KSW08]. We give the full description for completeness. Let $\mathsf{IPE} = (\mathsf{Setup}_{\mathsf{IPE}}, \mathsf{KeyGen}_{\mathsf{IPE}}, \mathsf{Enc}_{\mathsf{IPE}}, \mathsf{Dec}_{\mathsf{IPE}})$ be an IPE scheme. We construct an MIBE scheme $\mathsf{MIBE} = (\mathsf{Setup}_{\mathsf{MIBE}}, \mathsf{KeyGen}_{\mathsf{MIBE}}, \mathsf{Enc}_{\mathsf{MIBE}}, \mathsf{Dec}_{\mathsf{MIBE}})$ as follows.

$\mathsf{Setup}_{\mathsf{MIBE}}(1^\lambda, 1^n)$**:** This algorithm runs $(\mathsf{pp}, \mathsf{msk}) \xleftarrow{R} \mathsf{Setup}_{\mathsf{IPE}}(1^\lambda, 1^{n+1})$ and outputs $(\mathsf{pp}, \mathsf{msk})$. If pp specifies vector space $\mathbb{Z}_q^n$ as an IPE scheme, an identity-space of MIBE is specified to be $\mathbb{Z}_q$.

$\mathsf{KeyGen}_{\mathsf{MIBE}}(\mathsf{msk}, (\mathsf{id}_1, ..., \mathsf{id}_n) \in \mathbb{Z}_q^n)$**:** This algorithm computes $\{y_i \in \mathbb{Z}_q\}_{i\in\{0,...,n\}}$ such that $\prod_{i=1}^n (X - \mathsf{id}_n) = \sum_{i=0}^n y_i X^i$ as a polynomial in the indeterminate X, sets $\boldsymbol{y} := (y_0, ..., y_n)$, runs $\mathsf{sk}_{\boldsymbol{y}} \xleftarrow{R} \mathsf{KeyGen}_{\mathsf{IPE}}(\mathsf{msk}, \boldsymbol{y})$ and outputs $\mathsf{sk}_{(\mathsf{id}_1,...,\mathsf{id}_n)} := \mathsf{sk}_{\boldsymbol{y}}$.

$\mathsf{Enc}_{\mathsf{MIBE}}(\mathsf{id}, \mathsf{m})$**:** This algorithm sets $\boldsymbol{x} := (1, \mathsf{id}, \mathsf{id}^2, ..., \mathsf{id}^n)$, where id^i denotes the i-th power of id on $\mathbb{Z}_q$, runs $\mathsf{ct} \xleftarrow{R} \mathsf{Enc}_{\mathsf{IPE}}(\boldsymbol{x}, \mathsf{m})$ and outputs ct.

$\mathsf{Dec}_{\mathsf{MIBE}}(\mathsf{sk}_{(\mathsf{id}_1,...,\mathsf{id}_n)}, (\mathsf{id}_1, ..., \mathsf{id}_n), \mathsf{ct})$**:** This algorithm computes $\boldsymbol{y}$ similarly to in $\mathsf{KeyGen}_{\mathsf{MIBE}}$, runs $\mathsf{m} \xleftarrow{R} \mathsf{Dec}_{\mathsf{IPE}}(\mathsf{sk}_{(\mathsf{id}_1,...,\mathsf{id}_n)}, \boldsymbol{y}, \mathsf{ct})$ and outputs m.

Correctness. Suppose that we have $\mathsf{id} \in \{\mathsf{id}_1, ..., \mathsf{id}_n\}$. Let $\boldsymbol{x} := (1, \mathsf{id}, \mathsf{id}^2, ..., \mathsf{id}^n)$ and $\boldsymbol{y}$ be the vector associated with $\{\mathsf{id}_1, ..., \mathsf{id}_n\}$ specified as in the description of $\mathsf{KeyGen}_{\mathsf{MIBE}}$. Then we have $\boldsymbol{x}^T\boldsymbol{y} = \sum_{i=0}^n y_i \mathsf{id}^i = \prod_{i=1}^n (\mathsf{id} - \mathsf{id}_n) = 0$. Therefore the correcrtness of MIBE follows from the correctness of IPE.

Security.

Theorem 2. *If* IPE *is adaptively (resp. selectively) secure, then* MIBE *is adaptively (resp. selectively) secure. Moreover, if* IPE *is fully key-compact, then* MIBE *is fully key-compact.*

Proof. First, it is easy to see that MIBE is fully key-compact if IPE is fully key-compact since a decryption key $\mathsf{sk}_{(\mathsf{id}_1,...,\mathsf{id}_n)}$ of MIBE consists of a single secret key $\mathsf{sk}_{\boldsymbol{y}}$ of IPE whose size is polynomial in λ due to the full key-compactness of IPE. Then, we reduce the security of MIBE to IPE. Here, we only give the proof for the adaptive case because the selective case can be proven similarly. Let $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$ be an admissible adversary against the adaptive security of MIBE. Then we construct an adversary $\mathcal{B} = (\mathcal{B}_1, \mathcal{B}_2)$ against the adaptive security of IPE as follows.

$\mathcal{B}_1^{\mathsf{KeyGen}_{\mathsf{IPE}}(\mathsf{msk},\cdot)}(\mathsf{pp})$**:** This algorithm runs $\mathcal{A}_1^{\mathsf{KeyGen}_{\mathsf{MIBE}}(\mathsf{msk},\cdot)}(\mathsf{pp})$. When $\mathcal{A}_1$ queries $(\mathsf{id}_1, ..., \mathsf{id}_n)$ to $\mathsf{KeyGen}_{\mathsf{MIBE}}(\mathsf{msk}, \cdot)$, $\mathcal{B}_1$ computes $\{y_i \in \mathbb{Z}_q\}_{i\in\{0,...,n\}}$ such that $\prod_{i=1}^n (X - \mathsf{id}_n) = \sum_{i=0}^n y_i X^i$, sets $\boldsymbol{y} := (y_0, ..., y_n)$, queries $\boldsymbol{y}$ to its own oracle $\mathsf{KeyGen}_{\mathsf{IPE}}(\mathsf{msk}, \cdot)$ to obtain $\mathsf{sk}_{\boldsymbol{y}}$, sets $\mathsf{sk}_{(\mathsf{id}_1,...,\mathsf{id}_n)} := \mathsf{sk}_{\boldsymbol{y}}$, and returns $\mathsf{sk}_{(\mathsf{id}_1,...,\mathsf{id}_n)}$ as a response by the oracle $\mathsf{KeyGen}_{\mathsf{MIBE}}(\mathsf{msk}, \cdot)$. When $\mathcal{A}_1$ outputs $(\mathsf{id}^*, \mathsf{m}_0, \mathsf{m}_1, \mathsf{st})$, $\mathcal{B}_1$ sets $\boldsymbol{x}^* := (1, \mathsf{id}^*, (\mathsf{id}^*)^2, ..., (\mathsf{id}^*)^n)$ and outputs $(\boldsymbol{x}^*, \mathsf{m}_0, \mathsf{m}_1, \mathsf{st})$.

$\mathcal{B}_2^{\mathsf{KeyGen}_{\mathsf{IPE}}(\mathsf{msk},\cdot)}(\mathsf{ct}^*, \mathsf{st})$**:** This algorithms runs $\mathcal{A}_2^{\mathsf{KeyGen}_{\mathsf{MIBE}}(\mathsf{msk},\cdot)}(\mathsf{ct}^*, \mathsf{st})$. When $\mathcal{A}_2$ queries $(\mathsf{id}_1, ..., \mathsf{id}_n)$ to $\mathsf{KeyGen}_{\mathsf{MIBE}}(\mathsf{msk}, \cdot)$, $\mathcal{B}_2$ computes $\{y_i \in \mathbb{Z}_q\}_{i\in\{0,...,n\}}$ such that $\prod_{i=1}^n (X - \mathsf{id}_n) = \sum_{i=0}^n y_i X^i$, sets $\boldsymbol{y} := (y_0, ..., y_n)$, queries $\boldsymbol{y}$ to its own oracle $\mathsf{KeyGen}_{\mathsf{IPE}}(\mathsf{msk}, \cdot)$ to obtain $\mathsf{sk}_{\boldsymbol{y}}$, sets $\mathsf{sk}_{(\mathsf{id}_1,...,\mathsf{id}_n)} := \mathsf{sk}_{\boldsymbol{y}}$, and returns $\mathsf{sk}_{(\mathsf{id}_1,...,\mathsf{id}_n)}$ as a response by the oracle $\mathsf{KeyGen}_{\mathsf{MIBE}}(\mathsf{msk}, \cdot)$. When $\mathcal{A}_2$ outputs $\widehat{\mathsf{coin}}$, $\mathcal{B}_2$ outputs $\widehat{\mathsf{coin}}$.

It is easy to see that $\mathcal{B}$ perfectly simulates $\mathsf{KeyGen}_{\mathsf{MIBE}}(\mathsf{msk}, \cdot)$ for $\mathcal{A}$, and the challenge ciphertext simulated by $\mathcal{B}$ is a correct encryption of $\mathsf{m}_{\mathsf{coin}}$ where coin is the random coin chosen by the challenger in the experiment $\mathcal{B}$ is involved. Therefore, we have $\mathsf{Adv}^{\mathsf{CPA}}_{\mathsf{MIBE},\mathcal{A}}(\lambda) = \mathsf{Adv}^{\mathsf{CPA}}_{\mathsf{IPE},\mathcal{B}}(\lambda)$. What is left is to prove that $\mathcal{B}$ is admissible if $\mathcal{A}$ is admissible. Let $\boldsymbol{y} = (y_0, ..., y_n)$ be the corresponding vector to a queried set of identities $(\mathsf{id}_1, ..., \mathsf{id}_n)$ by $\mathcal{A}$, i.e., $\boldsymbol{y}$ satisfies $\prod_{i=1}^{n}(X - \mathsf{id}_n) = \sum_{i=0}^{n} y_i X^i$, and $\boldsymbol{x}^* := (1, \mathsf{id}^*, (\mathsf{id}^*)^2, ..., (\mathsf{id}^*)^n)$. Then we have $(\boldsymbol{x}^*)^T \boldsymbol{y} = \sum_{i=0}^{n} y_i (\mathsf{id}^*)^i = \prod_{i=1}^{n}(\mathsf{id}^* - \mathsf{id}_n) \neq 0$ where the last inequality holds because we have $\mathsf{id}^* \notin \{\mathsf{id}_1, ..., \mathsf{id}_n\}$ due to the admissibility of $\mathcal{A}$. This completes the proof. ■

3.3 IB-HPS from MIBE

Here, we give a construction of an IB-HPS based on any MIBE scheme. Moreover, we show that if the underlying MIBE scheme is fully key-compact, then the universality-ratio of the resulting IB-HPS can be made arbitrarily close to 1.

Let $\mathsf{MIBE} = (\mathsf{Setup}_{\mathsf{MIBE}}, \mathsf{KeyGen}_{\mathsf{MIBE}}, \mathsf{Enc}_{\mathsf{MIBE}}, \mathsf{Dec}_{\mathsf{MIBE}})$ be an MIBE scheme with a message space $\mathcal{M}$ and an identity space $\{0,1\}^{\ell_{\mathsf{id}}}$. We assume that there exists a positive integer ℓ_{k} such that $\{0,1\}^{\ell_{\mathsf{k}}}$ can be embedded into $\mathcal{M}$, i.e., there exists an efficiently computable injective function $\sigma : \{0,1\}^{\ell_{\mathsf{k}}} \to \mathcal{M}$. In the following, we often identify $\mathsf{k} \in \{0,1\}^{\ell_{\mathsf{k}}}$ with $\sigma(\mathsf{k})$ and treat k as an element of $\mathcal{M}$. Let $\mathsf{PRF} : \mathcal{K} \times \{0,1\}^{\ell_{\mathsf{id}}} \to \mathcal{R}$ where $\mathcal{R}$ denotes the randomness space for $\mathsf{KeyGen}_{\mathsf{MIBE}}$. Then for any positive integer n, we construct an IB-HPS $\Pi_n = (\mathsf{Setup}_{\mathsf{HPS}}, \mathsf{KeyGen}_{\mathsf{HPS}}, \mathsf{Encap}_{\mathsf{HPS}}, \mathsf{Encap}^*_{\mathsf{HPS}}, \mathsf{Decap}_{\mathsf{HPS}})$ with identity space $\{0,1\}^{\ell_{\mathsf{id}} - \lceil \log n \rceil - 1}$ and key space $\{0,1\}^{\ell_{\mathsf{k}}}$ as follows, where $\mathsf{bin}(m)$ denotes a binary representation of an integer m.

$\mathsf{Setup}_{\mathsf{HPS}}(1^\lambda)$**:** This algorithm generates $(\mathsf{pp}, \mathsf{msk}) \xleftarrow{\mathsf{R}} \mathsf{Setup}_{\mathsf{MIBE}}(1^\lambda, 1^n)$, chooses a PRF key $K \xleftarrow{\mathsf{R}} \mathcal{K}$, and outputs pp and (msk, K) as its public parameter and master secret key.

$\mathsf{KeyGen}_{\mathsf{HPS}}((\mathsf{msk}, K), \mathsf{id})$**:** This algorithm picks $r_i \xleftarrow{\mathsf{R}} \{0,1\}$ for $i \in [n]$, generates $\mathsf{sk}'_{\mathsf{id}} \xleftarrow{\mathsf{R}} \mathsf{KeyGen}_{\mathsf{MIBE}}(\mathsf{msk}, (\mathsf{id} \| \mathsf{bin}(1) \| r_1, ..., \mathsf{id} \| \mathsf{bin}(n) \| r_n); \mathsf{PRF}(K, \mathsf{id}))$, and outputs $\mathsf{sk}_{\mathsf{id}} := (\mathsf{sk}'_{\mathsf{id}}, \{r_i\}_{i \in [n]})$.

$\mathsf{Encap}_{\mathsf{HPS}}(\mathsf{id})$**:** This algorithm picks $\mathsf{k}_i \in \{0,1\}^{\ell_{\mathsf{k}}}$ for $i \in [n]$, generates $\mathsf{ct}_{i,b} \xleftarrow{\mathsf{R}} \mathsf{Enc}_{\mathsf{MIBE}}(\mathsf{id} || \mathsf{bin}(i) || b, \mathsf{k}_i)$ for $i \in [n]$ and $b \in \{0,1\}$, sets $\mathsf{ct} := \{\mathsf{ct}_{i,b}\}_{i \in [n], b \in \{0,1\}}$ and $\mathsf{k} := \bigoplus_{i \in [n]} \mathsf{k}_i$, and outputs $(\mathsf{ct}, \mathsf{k})$.

$\mathsf{Encap}^*_{\mathsf{HPS}}(\mathsf{id})$**:** This algorithm picks $\mathsf{k}_{i,b} \in \{0,1\}^{\ell_{\mathsf{k}}}$ for $i \in [n]$ and $b \in \{0,1\}$, generates $\mathsf{ct}_{i,b} \xleftarrow{\mathsf{R}} \mathsf{Enc}_{\mathsf{MIBE}}(\mathsf{id} || \mathsf{bin}(i) || b, \mathsf{k}_{i,b})$ for all $i \in [n]$, $b \in \{0,1\}$, sets $\mathsf{ct} := \{\mathsf{ct}_{i,b}\}_{i \in [n], b \in \{0,1\}}$, and outputs ct.

$\mathsf{Decap}_{\mathsf{HPS}}(\mathsf{sk}_{\mathsf{id}}, \mathsf{id}, \mathsf{ct})$**:** This algorithm parses $(\mathsf{sk}'_{\mathsf{id}}, \{r_i\}_{i \in [n]}) \leftarrow \mathsf{sk}_{\mathsf{id}}$ and $\{\mathsf{ct}_{i,b}\}_{i \in [n], b \in \{0,1\}} \leftarrow \mathsf{ct}$, runs $\mathsf{k}'_i \leftarrow \mathsf{Dec}_{\mathsf{MIBE}}(\mathsf{sk}'_{\mathsf{id}}, (\mathsf{id} || \mathsf{bin}(1) || r_1, ..., \mathsf{id} || \mathsf{bin}(n) || r_n), \mathsf{ct}_{i,r_i})$, and outputs $\mathsf{k} := \bigoplus_{i \in [n]} \mathsf{k}'_i$

Correctness. Correctness of Π_n is easy to see given the correctness of MIBE.

Security.

Theorem 3. *If* MIBE *is adaptively (resp. selectively) secure MIBE scheme, then* Π_n *is an adaptively (resp. selectively) secure and* $(n, 2^{-\ell_k})$*-universal IB-HPS with the universality-ratio* $\frac{n}{n+\ell_{\mathsf{sk}}(n)}$ *where* $\ell_{\mathsf{sk}}(n)$ *denotes the maximum length of* sk *generated by* $\mathsf{KeyGen}_{\mathsf{MIBE}}(\mathsf{msk}, \cdot)$ *where* $(\mathsf{pp}, \mathsf{msk}) \xleftarrow{R} \mathsf{Setup}_{\mathsf{MIBE}}(1^\lambda, 1^n)$*. Especially, if* MIBE *is fully key-compact, then we can make the universality-ratio arbitrarily close to* 1 *by increasing* n*.*

Proof.
Valid/invalid ciphertext indistinguishability. First, we prove that Π_n is adaptively (resp. selectively) secure if MIBE is adaptively (resp. selectively) secure. Here, we only give the proof for the adaptive case because the selective case can be proven similarly. We assume that there exists a PPT adversary $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$ that breaks the adaptive security of Π_n. We consider the following sequence of hybrid games.

Game_0 This game simulates the environment of $\mathsf{Expt}^{\mathsf{ind}}_{\Pi_n,\mathcal{A}}(\lambda)$ for the case of $\mathsf{coin} = 0$ (where $\mathcal{A}$ is always given a valid ciphertext) to $\mathcal{A}$. $\widehat{\mathsf{coin}}$ output by $\mathcal{A}$ is treated as the output of this game.

Game'_0 This game is the same as Game_0 except that the challenger uses a fresh randomness instead of $\mathsf{PRF}(K, \mathsf{id})$ when responding to $\mathcal{A}$'s key generation queries. We denote this modified key generation oracle by $\mathsf{KeyGen}'_{\mathsf{HPS}}(\mathsf{msk}, \cdot)$. We note that K is not needed for simulating $\mathsf{KeyGen}'_{\mathsf{HPS}}(\mathsf{msk}, \cdot)$.

Game'_x**:** For $x = 0, ..., n$, we consider the following games. We remark that the definitions of Game'_0 given above and below is consistent.

Game'_x :
$(\mathsf{pp}, \mathsf{msk}) \xleftarrow{R} \mathsf{Setup}_{\mathsf{MIBE}}(1^\lambda)$
$(\mathsf{id}^*, \mathsf{st}_{\mathcal{A}}) \xleftarrow{R} \mathcal{A}_1^{\mathsf{KeyGen}'_{\mathsf{HPS}}(\mathsf{msk},\cdot)}(\mathsf{pp})$
For $i = 1$ to x
  $\mathsf{k}_{i,0}, \mathsf{k}_{i,1} \xleftarrow{R} \{0,1\}^{\ell_k}$
For $i = x+1$ to n
  $\mathsf{k}_{i,0} \xleftarrow{R} \{0,1\}^{\ell_k}$
  $\mathsf{k}_{i,1} \coloneqq \mathsf{k}_{i,0}$
For $i \in [n]$, $b \in \{0,1\}$
  $\mathsf{ct}^*_{i,b} \xleftarrow{R} \mathsf{Enc}_{\mathsf{MIBE}}(\mathsf{id}^* || \mathsf{bin}(i) || b, \mathsf{k}_{i,b})$
  $\mathsf{ct}^* \coloneqq \{\mathsf{ct}^*_{i,b}\}_{i\in[n],b\in\{0,1\}}$
$\widehat{\mathsf{coin}} \xleftarrow{R} \mathcal{A}_2^{\mathsf{KeyGen}'_{\mathsf{HPS}}(\mathsf{msk},\cdot)}(\mathsf{ct}^*, \mathsf{st}_{\mathcal{A}})$
Return $\widehat{\mathsf{coin}}$.

It is easy to see that Game'_n simulates the environment of $\mathsf{Expt}^{\mathsf{ind}}_{\Pi_n,\mathcal{A}}(\lambda)$ given $\mathsf{coin} = 1$ (where $\mathcal{A}$ is always given an invalid ciphertext) to $\mathcal{A}$. Therefore, we have $|\Pr[1 \xleftarrow{R} \mathsf{Game}_0] - \Pr[1 \xleftarrow{R} \mathsf{Game}'_n]| = \mathsf{Adv}^{\mathsf{ind}}_{\Pi_n,\mathcal{A}}(\lambda)$. We prove that this is negligible by showing the following lemmas.

Lemma 1. *There exists a PPT adversary* $\mathcal{B}$ *against* PRF *such that* $|\Pr[\mathsf{Game}_0 = 1] - \Pr[\mathsf{Game}'_0 = 1]| = \mathsf{Adv}_{\mathsf{PRF},\mathcal{B}}(\lambda)$.

Proof. The PRF key K is used only when simulating the key generation oracle, and evaluations of the PRF on the same input id is not repeated more than once since $\mathcal{A}$ is not allowed to query the same identity more than once. Therefore, it is

straightforward to reduce the distinguishing advantage between these two games to the security of PRF. ■

Lemma 2. *For $x \in [n]$, there exists an admissible adversary $\mathcal{B}$ against* MIBE *such that* $|\Pr[\mathsf{Game}'_{x-1} = 1] - \Pr[\mathsf{Game}'_x = 1]| = \mathsf{Adv}^{\mathsf{CPA}}_{\mathsf{MIBE},\mathcal{B}}(\lambda)$.

Proof. We assume that $\mathcal{A}$ distinguishes Game'_x and Game'_{x+1}, and construct a PPT adversary $\mathcal{B} = (\mathcal{B}_1, \mathcal{B}_2)$ that breaks the adaptive security of MIBE. We describe $\mathcal{B}$ below.

$\mathcal{B}_1^{\mathsf{KeyGen}_{\mathsf{MIBE}}(\mathsf{msk},\cdot)}(\mathsf{pp})$**:** It runs $(\mathsf{id}^*, \mathsf{st}_{\mathcal{A}}) \xleftarrow{\mathsf{R}} \mathcal{A}_1^{\mathsf{KeyGen}'_{\mathsf{HPS}}(\mathsf{msk},\cdot)}(\mathsf{pp})$ where $\mathcal{B}$ simulates $\mathsf{KeyGen}'_{\mathsf{HPS}}$ to $\mathcal{A}_1$ as follows. When $\mathcal{A}_1$ makes its j-th query $\mathsf{id}^{(j)}$ to $\mathsf{KeyGen}'_{\mathsf{HPS}}$, it randomly picks $r_i^{(j)} \xleftarrow{\mathsf{R}} \{0,1\}$ for $i \in [n]$, queries $(\mathsf{id}^{(j)}\|\mathsf{bin}(1)\|r_1^{(j)}, ..., \mathsf{id}^{(j)}\|\mathsf{bin}(n)\|r_n^{(j)})$ to its own oracle $\mathsf{KeyGen}_{\mathsf{MIBE}}$ to obtain $\mathsf{sk}_{\mathsf{id}}^{(j)'}$, and gives $(\mathsf{sk}_{\mathsf{id}}^{(j)'}, \{r_i^{(j)}\}_{i\in[n]})$ to $\mathcal{A}_1$ as a response from the oracle $\mathsf{KeyGen}'_{\mathsf{HPS}}$. If there exists $j \in [Q]$ such that $\mathsf{id}^* = \mathsf{id}^{(j)}$, then it sets $r_x^* \coloneqq r_x^{(j)}$. Otherwise it picks $r_x^* \xleftarrow{\mathsf{R}} \{0,1\}$. It picks $\mathsf{k}_{x,0}, \mathsf{k}_{x,1} \xleftarrow{\mathsf{R}} \{0,1\}^{\ell_{\mathsf{k}}}$ and sets $\mathsf{st}_{\mathcal{B}} \coloneqq (\mathsf{st}_{\mathcal{A}}, r_x^*, \mathsf{k}_{x,0}, \mathsf{k}_{x,1})$. Then, $\mathcal{B}_1$ outputs $(\mathsf{id}\|\mathsf{bin}(x)\|(1-r_x^*), \mathsf{k}_{x,r_x^*}, \mathsf{k}_{x,1-r_x^*}, \mathsf{st}_{\mathcal{B}})$.

$\mathcal{B}_2^{\mathsf{KeyGen}_{\mathsf{MIBE}}(\mathsf{msk},\cdot)}(\mathsf{ct}^*_{\mathsf{MIBE}}, \mathsf{st}_{\mathcal{B}})$**:** It parses $\{\mathsf{ct}_{i,b}\}_{i\in[n],b\in\{0,1\}} \leftarrow \mathsf{ct}^*$ and $(\mathsf{st}_{\mathcal{A}}, r_x^*) \leftarrow \mathsf{st}_{\mathcal{B}}$. It picks $\mathsf{k}_{i,0}, \mathsf{k}_{i,1} \xleftarrow{\mathsf{R}} \{0,1\}^{\ell_{\mathsf{k}}}$ for $i = 1, ..., x-1$. It picks $\mathsf{k}_{i,0} \xleftarrow{\mathsf{R}} \{0,1\}^{\ell_{\mathsf{k}}}$ and sets $\mathsf{k}_{i,1} \coloneqq \mathsf{k}_{i,0}$ for $i = x+1, ..., n$. It computes $\mathsf{ct}^*_{i,b} \xleftarrow{\mathsf{R}} \mathsf{Enc}_{\mathsf{MIBE}}(\mathsf{id}\|\mathsf{bin}(i)\|b, \mathsf{k}_{i,b})$ for all $(i,b) \in ([n]\times\{0,1\}) \setminus \{(x, 1-r_x^*)\}$, and sets $\mathsf{ct}^*_{x,1-r_x^*} \coloneqq \mathsf{ct}^*_{\mathsf{MIBE}}$. Then, it sets $\mathsf{ct}^*_{\mathsf{HPS}} \coloneqq \{\mathsf{ct}^*_{i,b}\}_{i\in[n],b\in\{0,1\}}$ and runs $\widehat{\mathsf{coin}} \xleftarrow{\mathsf{R}} \mathcal{A}_2^{\mathsf{KeyGen}'_{\mathsf{HPS}}(\mathsf{msk},\cdot)}(\mathsf{ct}^*_{\mathsf{HPS}}, \mathsf{st}_{\mathcal{A}})$ where $\mathcal{B}_2$ simulates $\mathsf{KeyGen}'_{\mathsf{HPS}}(\mathsf{msk},\cdot)$ as follows. When $\mathcal{A}_2$ makes a j-th query $\mathsf{id}^{(j)}$ to $\mathsf{KeyGen}'_{\mathsf{HPS}}(\mathsf{msk},\cdot)$ (where the number of query is counted through $\mathcal{A}_1$ and $\mathcal{A}_2$), if $\mathsf{id}^{(j)} \neq \mathsf{id}^*$, then $\mathcal{B}_2$ randomly picks $r_i^{(j)} \xleftarrow{\mathsf{R}} \{0,1\}$ for $i \in [n]$, and otherwise it randomly picks $r_i^{(j)} \xleftarrow{\mathsf{R}} \{0,1\}$ for $i \in [n] \setminus \{x\}$ and sets $r_x^{(j)} \coloneqq r_x^*$. Then, $\mathcal{B}_2$ queries $(\mathsf{id}^{(j)}\|\mathsf{bin}(1)\|r_1^{(j)}, ..., \mathsf{id}^{(j)}\|\mathsf{bin}(n)\|r_n^{(j)})$ to its own oracle $\mathsf{KeyGen}_{\mathsf{MIBE}}(\mathsf{msk},\cdot)$ to obtain $\mathsf{sk}_{\mathsf{id}}^{(j)'}$, and gives $(\mathsf{sk}_{\mathsf{id}}^{(j)'}, \{r_i^{(j)}\}_{i\in[n]})$ to $\mathcal{A}_2$ as a response from the oracle $\mathsf{KeyGen}'_{\mathsf{HPS}}(\mathsf{msk},\cdot)$. Finally, $\mathcal{B}_2$ outputs coin'.

This completes the description of $\mathcal{B}$. First, we can see that $\mathcal{B}$ is admissible because $\mathcal{B}$'s query to its oracle never contains $\mathsf{id}^*\|\mathsf{bin}(x)\|(1-r_x^*)$. If the random coin chosen by the challenger of $\mathsf{Expt}^{\mathsf{CPA}}_{\mathsf{MIBE},\mathcal{B}}$, which is the experiment $\mathcal{B}$ is involved in, is 0, then $\mathcal{B}$ perfectly simulates Game'_{x-1} to $\mathcal{A}$, and if the coin is 1, then $\mathcal{B}$ perfectly simulates Game'_x to $\mathcal{A}$. Therefore we have $|\Pr[\mathsf{Game}'_{x-1} = 1] - \Pr[\mathsf{Game}'_x = 1]| = \mathsf{Adv}^{\mathsf{CPA}}_{\mathsf{MIBE},\mathcal{B}}(\lambda)$ as desired. ■

Due to the above lemmas and the triangle inequality, if PRF is a secure PRF and MIBE is adaptively secure, then $|\Pr[1 \xleftarrow{\mathsf{R}} \mathsf{Game}_0] - \Pr[1 \xleftarrow{\mathsf{R}} \mathsf{Game}'_n]| = \mathsf{Adv}^{\mathsf{ind}}_{\Pi,\mathcal{A}}(\lambda)$ is negligible, and thus Π_n is adaptively secure.

Universality. We prove that Π_n is $(n, 2^{-\ell_k})$-universal. First, for any fixed $(\mathsf{msk}, \mathsf{pp})$ and $\mathsf{id} \in \{0,1\}^{\ell_{\mathsf{id}}}$, we have $\mathsf{H}_\infty(\mathsf{sk}_{\mathsf{id}}) = n$ where $\mathsf{sk}_{\mathsf{id}} \xleftarrow{R} \mathsf{KeyGen}_{\mathsf{HPS}}(\mathsf{msk}, \mathsf{id})$ because a different choice of $\{r_i\}_{i\in[n]} \in \{0,1\}^n$ gives a different value of $\mathsf{sk}_{\mathsf{id}}$. For any fixed $(\mathsf{msk}, \mathsf{pp})$ and $\mathsf{id} \in \{0,1\}^{\ell_{\mathsf{id}}}$, let $\mathsf{sk}_{\mathsf{id}} = (\mathsf{sk}'_{\mathsf{id}}, \{r_i\}_{i\in[n]})$ and $\overline{\mathsf{sk}}_{\mathsf{id}} = (\overline{\mathsf{sk}'}_{\mathsf{id}}, \{\overline{r}_i\}_{i\in[n]})$ be distinct secret keys produced by $\mathsf{KeyGen}_{\mathsf{HPS}}(\mathsf{msk}, \mathsf{id})$. Since the first component $\mathsf{sk}'_{\mathsf{id}}$ in a secret key $\mathsf{sk}_{\mathsf{id}}$ is deterministically derived from msk, id, and $\{r_i\}_{i\in[n]}$, we must have $\{r_i\}_{i\in[n]} \neq \{\overline{r}_i\}_{i\in[n]}$. Let ct be an invalid ciphertext generated by $\mathsf{Encap}^*(\mathsf{id})$, i.e., we pick $\mathsf{k}_{i,b} \in \{0,1\}^{\ell_k}$ for $i \in [n]$ and $b \in \{0,1\}$, generate $\mathsf{ct}_{i,b} \xleftarrow{R} \mathsf{Enc}_{\mathsf{MIBE}}(\mathsf{id}||\mathsf{bin}(i)||b, \mathsf{k}_{i,b})$ for all $i \in [n]$, $b \in \{0,1\}$ and set $\mathsf{ct} := \{\mathsf{ct}_{i,b}\}_{i\in[n],b\in\{0,1\}}$. Then, we have $\mathsf{Decap}(\mathsf{sk}_{\mathsf{id}}, \mathsf{id}, \mathsf{ct}) = \bigoplus_{i=1}^n \mathsf{k}_{i,r_i}$ and $\mathsf{Decap}(\overline{\mathsf{sk}}_{\mathsf{id}}, \mathsf{id}, \mathsf{ct}) = \bigoplus_{i=1}^n \mathsf{k}_{i,\overline{r}_i}$ by the correctness of MIBE. Since there exists $i^* \in [n]$ such that $r_{i^*} \neq \overline{r}_{i^*}$ and $\mathsf{k}_{i^*,0}$ and $\mathsf{k}_{i^*,1}$ are independently random, $\mathsf{Decap}(\mathsf{sk}_{\mathsf{id}}, \mathsf{id}, \mathsf{ct})$ and $\mathsf{Decap}(\overline{\mathsf{sk}}_{\mathsf{id}}, \mathsf{id}, \mathsf{ct})$ are independently random. Therefore, we have

$$\Pr_{\mathsf{ct}\xleftarrow{R}\mathsf{Encap}^*(\mathsf{id})}[\mathsf{Decap}(\mathsf{sk}_{\mathsf{id}}, \mathsf{id}, \mathsf{ct}) = \mathsf{Decap}(\overline{\mathsf{sk}}_{\mathsf{id}}, \mathsf{id}, \mathsf{ct})] \leq 2^{-\ell_k}.$$

Therefore Π_n is $(n, 2^{-\ell_k})$-universal. Since a secret key $\mathsf{sk}_{\mathsf{id}}$ of Π_n consists of a secret key $\mathsf{sk}'_{\mathsf{id}}$ of MIBE and an n-bit string $\{r_i\}_{i\in[n]}$, the secret key size of Π_n is $n + \ell_{\mathsf{sk}}(n)$. Therefore, the universality-ratio of Π_n is $\frac{n}{n+\ell_{\mathsf{sk}}(n)}$. Especially, if MIBE is fully key-compact, then $\ell_{\mathsf{sk}}(n)$ is a fixed polynomial in λ that does not depend on n, and thus we can make the universality-ratio arbitrarily close to 1 by increasing n. ■

4 Leakage resilient IBE in BRM

Here, we first observe that combining Theorem 1, 2, and 3, we can construct a leakage resilient IBE scheme in the BRM based on any IPE scheme, and the leakage-ratio of the resulting IBE scheme can be made arbitrary close to 1 if the underlying IPE is fully key-compact. Then we give some instantiations for it.

4.1 Construction from IPE

Combining Theorem 1, 2, and 3, we obtain the following corollary.

Corollary 1. *Suppose we have an adaptively (resp. selectively) secure fully key-compact IPE scheme with vector space $\mathbb{Z}_q^n$ whose secret key size is $|\mathsf{sk}_{\mathsf{IPE}}(n)|$ where $q = \lambda^{\omega(1)}$ and the dimension $n \in \mathbb{N}$ can be flexibly chosen by the setup algorithm. Then for any $n = \mathrm{poly}(\lambda)$ and constant $\epsilon > 0$, we can construct an adaptively (resp. selectively) secure leakage resilient IBE scheme in the BRM with identity-space $\{0,1\}^{\lfloor \frac{\log q}{2} \rfloor}$ and message space $\{0,1\}^v$ such that*

1. *Public parameter/master secret key size is almost the same as that of the underlying IPE scheme.*

2. *Ciphertext-size/encryption-time/decryption-time are* $O(n(v+\lambda))$ *times larger than that of the underlying IPE with dimension* n.
3. *leakage-ratio is* $(1-\epsilon)(\frac{n}{n+|\mathsf{sk}_{\mathsf{IPE}}(n)|})$ *for sufficiently large values of the leakage parameter* ℓ.

Especially, by choosing sufficiently large $n = O(|\mathsf{sk}_{\mathsf{IPE}}(n)|)$*, we can make the leakage-ratio* $1-\epsilon$ *for any constant* $\epsilon > 0$.

Proof. Suppose we have an adaptively (resp. selectively) secure fully key-compact IPE scheme with vector space $\mathbb{Z}_q^n$. By Theorem 2, we can construct an adaptively (resp. selectively) secure fully key-compact MIBE scheme whose identity-space is $\{0,1\}^{\lfloor \log q \rfloor}$ and public parameter/master secret key size, ciphertext-size/encryption-time/decryption-time, are almost the same as those of the underlying IPE. Then by Theorem 3, for any $n \in \mathbb{N}$, we can construct an adaptively (resp. selectively) secure IB-HPS whose identity-space is $\{0,1\}^{\lfloor \log q \rfloor - \lfloor \log(n) \rfloor - 1}$, master public/secret key size is the same as that of the underlying IPE scheme, ciphertext-size/encryption-time/decryption-time differ by a factor of $O(n)$ from those of the underlying IPE with dimension n, and universality-ratio $\frac{n}{n+|\mathsf{sk}_{\mathsf{IPE}}(n)|}$. Here, for sufficiently large λ, we have $\lfloor \log q \rfloor - \lfloor \log n \rfloor - 1 > \lfloor \frac{\log q}{2} \rfloor$ since we have $q = \lambda^{\omega(1)}$ and $n = \mathrm{poly}(\lambda)$. Therefore the identity-space of IB-HPS can be restricted to $\{0,1\}^{\lfloor \frac{\log q}{2} \rfloor}$. Finally, by applying Theorem 1 to this IB-HPS, we obtain Corollary 1. Especially, for any constant $\epsilon' > 0$, if we set $n > \frac{|\mathsf{sk}_{\mathsf{IPE}}(n)|}{\epsilon'}$ then we have $\frac{n}{n+|\mathsf{sk}_{\mathsf{IPE}}(n)|} > \frac{1}{1+\epsilon'}$. Thus we can make the leakage-ratio arbitrarily close to 1. ■

4.2 Instantiations

By Corollary 1, we can construct a leakage resilient IBE scheme in the BRM whose leakage-ratio is arbitrarily close to 1 based on any fully key-compact IPE scheme. We give a list of possible instantiations below. Note that all constructions are secure in the standard model. In the following, $|\mathsf{sk}_{\mathsf{IPE}}|$ and $|\mathsf{ct}_{\mathsf{IPE}}(n)|$ denotes the size of a secret key and a ciphertext when the dimension is set to be n. (Remark that since these schemes are fully-key-compact, $|\mathsf{sk}_{\mathsf{IPE}}|$ does not depend on n.)

1. Wee constructed an adaptively secure IPE scheme from the subgroup decision assumption on composite-order pairing groups [Wee14]. The construction is fully key-compact since a secret key for vector $\boldsymbol{y} \in \mathbb{Z}_N^n$ consists of 2 group elements where N is the order of a group and consists of three distinct primes. A ciphertext of the scheme consists of n elements of the group. Namely, we have $|\mathsf{sk}_{\mathsf{IPE}}| = O(N)$ and $|\mathsf{ct}_{\mathsf{IPE}}(n)| = O(nN)$. For any constant $\epsilon > 0$, we can set $n = O(N)$ to achieve the leakage-ratio $1-\epsilon$. In this case, the ciphertext size of the resulting IBE scheme is $O(N^3\lambda)$.
2. Chen et al. constructed an adaptively secure IPE scheme from the d-Lin assumption on prime-order pairing groups [CGW15]. The construction is fully key-compact since a secret key for vector $\boldsymbol{y} \in \mathbb{Z}_q^n$ consists of $2(d+1)$ group

elements where q is the order of a group. If we use the 1-Lin (i.e., SXDH) assumption, only 4 group elements. A ciphertext of the scheme consists of $(n+1)(d+1)$ group elements and a message masking part. Namely, we have $|\mathsf{sk}_{\mathsf{IPE}}| = O(d\lambda)$ and $|\mathsf{ct}_{\mathsf{IPE}}(n)| = O(nd\lambda)$. For any constant $\epsilon > 0$, we can set $n = O(d\lambda)$ to achieve the leakage-ratio $1-\epsilon$. In this case, the ciphertext size of the resulting IBE scheme is $O(d^3\lambda^4)$.

3. Agrawal et al. constructed a selectively secure IPE scheme from the LWE assumption [AFV11]. The construction is fully key-compact since a secret key for a vector $\boldsymbol{y} \in \mathbb{Z}_q^n$ is a vector of small length in $\mathbb{Z}^m$ where m does not depend on n. More precisely, a secret key consists of a vector of length of $O(\sigma\sqrt{m})$ (with overwhelming probability) in $\mathbb{Z}^{2m}$, and a ciphertext consists of $O(n \log q)$ vectors in $\mathbb{Z}_q^m$ where we can set $q = \mathrm{poly}(\lambda, n)$, $m = O(\lambda^{1+\delta})$ and $\sigma = \mathrm{poly}(\lambda, n)$ where $\delta > 0$ is an arbitrary constant. Namely, we have $|\mathsf{sk}_{\mathsf{IPE}}| = \tilde{O}(\lambda^{1+\delta})$ and $|\mathsf{ct}_{\mathsf{IPE}}(n)| = O(n\lambda^{1+\delta})$. For any constant $\epsilon > 0$, we can set $n = \tilde{O}(\lambda^{1+\delta})$ to achieve the leakage-ratio $1-\epsilon$. In this case, the ciphertext size of the resulting IBE scheme is $\tilde{O}(\lambda^{4+3\delta})$.
4. We constructed a selectively secure IPE scheme from the CBDH assumption. This construction is an extension of Boneh-Boyen selectively secure IBE [BB04] and can be seen as a selectively secure variant of the scheme proposed by Chen et al. [CGW15] (which is an adaptively secure IPE scheme under the d-Lin assumption). The construction is fully key compact since a secret key for vector $\boldsymbol{y} \in \mathbb{Z}_q^n$ consists of 2 group elements where q is the order of a group. A ciphertext consists of $n+1$ group elements and a message masking part. Namely, we have $|\mathsf{sk}_{\mathsf{IPE}}| = O(\lambda)$ and $|\mathsf{ct}_{\mathsf{IPE}}(n)| = O(n\lambda)$. For any constant $\epsilon > 0$, we can set $n = O(\lambda)$ to achieve the leakage-ratio $1-\epsilon$. In this case, the ciphertext size of the resulting IBE scheme is $O(\lambda^4)$.

Acknowledgments

We thank Daniel Wichs for helpful comments on the presentation.

References

ADN+09. Joël Alwen, Yevgeniy Dodis, Moni Naor, Gil Segev, Shabsi Walfish, and Daniel Wichs. Public-key encryption in the bounded-retrieval model. *IACR Cryptology ePrint Archive*, 2009:512, 2009. Version 20091028:202321. This paper appeared in EUROCRYPT 2010, volume 6110 of LNCS. 2, 3, 4, 5, 8, 12, 14, 15

ADW09. Joël Alwen, Yevgeniy Dodis, and Daniel Wichs. Leakage-resilient public-key cryptography in the bounded-retrieval model. In Shai Halevi, editor, *CRYPTO 2009*, volume 5677 of *LNCS*, pages 36–54. Springer, Heidelberg, August 2009. 2

AFV11. Shweta Agrawal, David Mandell Freeman, and Vinod Vaikuntanathan. Functional encryption for inner product predicates from learning with errors. In Dong Hoon Lee and Xiaoyun Wang, editors, *ASIACRYPT 2011*, volume 7073 of *LNCS*, pages 21–40. Springer, Heidelberg, December 2011. 3, 4, 24

AGV09. Adi Akavia, Shafi Goldwasser, and Vinod Vaikuntanathan. Simultaneous hardcore bits and cryptography against memory attacks. In Omer Reingold, editor, *TCC 2009*, volume 5444 of *LNCS*, pages 474–495. Springer, Heidelberg, March 2009. 2

BB04. Dan Boneh and Xavier Boyen. Efficient selective-ID secure identity based encryption without random oracles. In Christian Cachin and Jan Camenisch, editors, *EUROCRYPT 2004*, volume 3027 of *LNCS*, pages 223–238. Springer, Heidelberg, May 2004. 24, 27

BG10. Zvika Brakerski and Shafi Goldwasser. Circular and leakage resilient public-key encryption under subgroup indistinguishability - (or: Quadratic residuosity strikes back). In Tal Rabin, editor, *CRYPTO 2010*, volume 6223 of *LNCS*, pages 1–20. Springer, Heidelberg, August 2010. 9

BKKV10. Zvika Brakerski, Yael Tauman Kalai, Jonathan Katz, and Vinod Vaikuntanathan. Overcoming the hole in the bucket: Public-key cryptography resilient to continual memory leakage. In *51st FOCS*, pages 501–510. IEEE Computer Society Press, October 2010. 9

BLSV18. Zvika Brakerski, Alex Lombardi, Gil Segev, and Vinod Vaikuntanathan. Anonymous IBE, leakage resilience and circular security from new assumptions. In Jesper Buus Nielsen and Vincent Rijmen, editors, *EUROCRYPT 2018, Part I*, volume 10820 of *LNCS*, pages 535–564. Springer, Heidelberg, April / May 2018. 9

BSW13. Elette Boyle, Gil Segev, and Daniel Wichs. Fully leakage-resilient signatures. *Journal of Cryptology*, 26(3):513–558, July 2013. 2

CDRW10. Sherman S. M. Chow, Yevgeniy Dodis, Yannis Rouselakis, and Brent Waters. Practical leakage-resilient identity-based encryption from simple assumptions. In Ehab Al-Shaer, Angelos D. Keromytis, and Vitaly Shmatikov, editors, *ACM CCS 10*, pages 152–161. ACM Press, October 2010. 9, 12

CGW15. Jie Chen, Romain Gay, and Hoeteck Wee. Improved dual system ABE in prime-order groups via predicate encodings. In Elisabeth Oswald and Marc Fischlin, editors, *EUROCRYPT 2015, Part II*, volume 9057 of *LNCS*, pages 595–624. Springer, Heidelberg, April 2015. 3, 4, 23, 24, 27

CS02. Ronald Cramer and Victor Shoup. Universal hash proofs and a paradigm for adaptive chosen ciphertext secure public-key encryption. In Lars R. Knudsen, editor, *EUROCRYPT 2002*, volume 2332 of *LNCS*, pages 45–64. Springer, Heidelberg, April / May 2002. 2

CZLC16. Yu Chen, Zongyang Zhang, Dongdai Lin, and Zhenfu Cao. Generalized (identity-based) hash proof system and its applications. *Security and Communication Networks*, 9(12):1698–1716, 2016. 3, 4

DGK+10. Yevgeniy Dodis, Shafi Goldwasser, Yael Tauman Kalai, Chris Peikert, and Vinod Vaikuntanathan. Public-key encryption schemes with auxiliary inputs. In Daniele Micciancio, editor, *TCC 2010*, volume 5978 of *LNCS*, pages 361–381. Springer, Heidelberg, February 2010. 10

DHLW10a. Yevgeniy Dodis, Kristiyan Haralambiev, Adriana López-Alt, and Daniel Wichs. Cryptography against continuous memory attacks. In *51st FOCS*, pages 511–520. IEEE Computer Society Press, October 2010. 9

DHLW10b. Yevgeniy Dodis, Kristiyan Haralambiev, Adriana López-Alt, and Daniel Wichs. Efficient public-key cryptography in the presence of key leakage. In Masayuki Abe, editor, *ASIACRYPT 2010*, volume 6477 of *LNCS*, pages 613–631. Springer, Heidelberg, December 2010. 9

DKL09. Yevgeniy Dodis, Yael Tauman Kalai, and Shachar Lovett. On cryptography with auxiliary input. In Michael Mitzenmacher, editor, *41st ACM STOC*, pages 621–630. ACM Press, May / June 2009. 10

DLW06. Giovanni Di Crescenzo, Richard J. Lipton, and Shabsi Walfish. Perfectly secure password protocols in the bounded retrieval model. In Shai Halevi and Tal Rabin, editors, *TCC 2006*, volume 3876 of *LNCS*, pages 225–244. Springer, Heidelberg, March 2006. 2

Dzi06. Stefan Dziembowski. Intrusion-resilience via the bounded-storage model. In Shai Halevi and Tal Rabin, editors, *TCC 2006*, volume 3876 of *LNCS*, pages 207–224. Springer, Heidelberg, March 2006. 2

GJS11. Sanjam Garg, Abhishek Jain, and Amit Sahai. Leakage-resilient zero knowledge. In Phillip Rogaway, editor, *CRYPTO 2011*, volume 6841 of *LNCS*, pages 297–315. Springer, Heidelberg, August 2011. 2

GL89. Oded Goldreich and Leonid A. Levin. A hard-core predicate for all one-way functions. In *21st ACM STOC*, pages 25–32. ACM Press, May 1989. 27

HLWW16. Carmit Hazay, Adriana López-Alt, Hoeteck Wee, and Daniel Wichs. Leakage-resilient cryptography from minimal assumptions. *Journal of Cryptology*, 29(3):514–551, July 2016. 2, 3, 4, 5, 9

KP17. Kaoru Kurosawa and Le Trieu Phong. Anonymous and leakage resilient IBE and IPE. *Des. Codes Cryptography*, 85(2):273–298, 2017. 2, 9, 12

KSW08. Jonathan Katz, Amit Sahai, and Brent Waters. Predicate encryption supporting disjunctions, polynomial equations, and inner products. In Nigel P. Smart, editor, *EUROCRYPT 2008*, volume 4965 of *LNCS*, pages 146–162. Springer, Heidelberg, April 2008. 7, 17

KV09. Jonathan Katz and Vinod Vaikuntanathan. Signature schemes with bounded leakage resilience. In Mitsuru Matsui, editor, *ASIACRYPT 2009*, volume 5912 of *LNCS*, pages 703–720. Springer, Heidelberg, December 2009. 2

LRW11. Allison B. Lewko, Yannis Rouselakis, and Brent Waters. Achieving leakage resilience through dual system encryption. In Yuval Ishai, editor, *TCC 2011*, volume 6597 of *LNCS*, pages 70–88. Springer, Heidelberg, March 2011. 2, 9, 12

NS12. Moni Naor and Gil Segev. Public-key cryptosystems resilient to key leakage. *SIAM J. Comput.*, 41(4):772–814, 2012. 2, 9

QL13. Baodong Qin and Shengli Liu. Leakage-resilient chosen-ciphertext secure public-key encryption from hash proof system and one-time lossy filter. In Kazue Sako and Palash Sarkar, editors, *ASIACRYPT 2013, Part II*, volume 8270 of *LNCS*, pages 381–400. Springer, Heidelberg, December 2013. 9

QL14. Baodong Qin and Shengli Liu. Leakage-flexible CCA-secure public-key encryption: Simple construction and free of pairing. In Hugo Krawczyk, editor, *PKC 2014*, volume 8383 of *LNCS*, pages 19–36. Springer, Heidelberg, March 2014. 9

Wee14. Hoeteck Wee. Dual system encryption via predicate encodings. In Yehuda Lindell, editor, *TCC 2014*, volume 8349 of *LNCS*, pages 616–637. Springer, Heidelberg, February 2014. 3, 4, 23

YAX+16. Zuoxia Yu, Man Ho Au, Qiuliang Xu, Rupeng Yang, and Jinguang Han. Leakage-resilient functional encryption via pair encodings. In Joseph K. Liu and Ron Steinfeld, editors, *ACISP 16, Part I*, volume 9722 of *LNCS*, pages 443–460. Springer, Heidelberg, July 2016. 9

YCZY12. Tsz Hon Yuen, Sherman S. M. Chow, Ye Zhang, and Siu Ming Yiu. Identity-based encryption resilient to continual auxiliary leakage. In David Pointcheval and Thomas Johansson, editors, *EUROCRYPT 2012*, volume 7237 of *LNCS*, pages 117–134. Springer, Heidelberg, April 2012. 10

ZCG+18. Jie Zhang, Jie Chen, Junqing Gong, Aijun Ge, and Chuangui Ma. Leakage-resilient attribute based encryption in prime-order groups via predicate encodings. *Des. Codes Cryptography*, 86(6):1339–1366, 2018. 9

A Key-Compact IPE from CBDH or DBDH

Here, we give constructions of a fully key-compact selectively secure IPE scheme based on the CBDH or DBDH assumptions. The constructions are simple extensions of the Boneh-Boyen IBE [BB04] and can be seen as selectively secure variants of the adaptively secure short secret key IPE scheme by Chen, Gay, and Wee [CGW15].

A.1 Definitions

First, we define pairing groups and CBDH and DBDH assumptions for it. Let $\mathbb{G}_1$, $\mathbb{G}_2$ and $\mathbb{G}_T$ be groups of prime order q associated with a pairing $e : \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T$. We require e to satisfy the following two properties.

Bilinearity For all $g_1 \in \mathbb{G}_1$, $g_2 \in \mathbb{G}_2$ and $a, b \in \mathbb{Z}_q$, it holds that $e(g_1^a, g_2^b) = e(g_1, g_2)^{ab}$.

Non-degeneracy If g_1 and g_2 generate $\mathbb{G}_1$ and $\mathbb{G}_2$ respectively, then $e(g_1, g_2) \neq 1$.

Definition 8. *(Computational Bilinear Diffie-Hellman Assumption.) We say that the computational bilinear Diffie-Hellman (CBDH) assumption holds if for any PPT adversary $\mathcal{A}$, we have*

$$\mathsf{Adv}^{\mathsf{cbdh}}_{\mathcal{A}}(\lambda) \coloneqq \Pr[e(g_1, g_2)^{xyz} \xleftarrow{\mathsf{R}} \mathcal{A}(g_1, g_1^\alpha, g_1^\beta, g_1^\gamma, g_2, g_2^\alpha, g_2^\beta, g_2^\gamma)] = \mathrm{negl}(\lambda)$$

where $g_1 \xleftarrow{\mathsf{R}} \mathbb{G}_1$, $g_2 \xleftarrow{\mathsf{R}} \mathbb{G}_2$ and $\alpha, \beta, \gamma \xleftarrow{\mathsf{R}} \mathbb{Z}_q$.

Definition 9. *(Decisional Bilinear Diffie-Hellman Assumption.) We say that the decisional bilinear Diffie-Hellman (DBDH) assumption holds if for any PPT adversary $\mathcal{A}$, we have*

$$\begin{aligned} &|\Pr[\mathcal{A}(g_1, g_1^\alpha, g_1^\beta, g_1^\gamma, g_2, g_2^\alpha, g_2^\beta, g_2^\gamma, T_0) = 1] \\ &- \Pr[\mathcal{A}(g_1, g_1^\alpha, g_1^\beta, g_1^\gamma, g_2, g_2^\alpha, g_2^\beta, g_2^\gamma, T_1) = 1]| = \mathrm{negl}(\lambda) \end{aligned}$$

where $g_1 \xleftarrow{\mathsf{R}} \mathbb{G}_1$, $g_2 \xleftarrow{\mathsf{R}} \mathbb{G}_2$, $\alpha, \beta, \gamma \xleftarrow{\mathsf{R}} \mathbb{Z}_q$, $T_0 \coloneqq e(g_1, g_2)^{\alpha\beta\gamma}$, and $T_1 \xleftarrow{\mathsf{R}} \mathbb{G}_T$.

By the Goldreich-Levin theorem [GL89], the following lemma holds.

Lemma 3. *(Hardcore security of CBDH.) If the CBDH assumption holds, then there exists a family $\mathcal{GL}$ of functions $\mathsf{hc} : \mathbb{G}_T \to \{0,1\}$ such that*

$$\begin{aligned}&|\Pr[\mathcal{A}(g_1, g_1^\alpha, g_1^\beta, g_1^\gamma, g_2, g_2^\alpha, g_2^\beta, g_2^\gamma, \mathsf{hc}, T_0) = 1]\\ &- \Pr[\mathcal{A}(g_1, g_1^\alpha, g_1^\beta, g_1^\gamma, g_2, g_2^\alpha, g_2^\beta, g_2^\gamma, \mathsf{hc}, T_1) = 1]| = \mathrm{negl}(\lambda)\end{aligned}$$

where $g_1 \xleftarrow{R} \mathbb{G}_1$, $g_2 \xleftarrow{R} \mathbb{G}_2$, $\alpha, \beta, \gamma \xleftarrow{R} \mathbb{Z}_q$, $\mathsf{hc} \xleftarrow{R} \mathcal{GL}$, $T_0 := \mathsf{hc}(e(g_1, g_2)^{\alpha\beta\gamma})$, *and* $T_1 \xleftarrow{R} \{0,1\}$.

A.2 Construction

We first describe our IPE scheme based on the CBDH assumption.

$\mathsf{Setup}(1^\lambda, 1^n)$: It generates parameters of a pairing group $\mathsf{pp_{bm}} := (q, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, e, g_1, g_2)$, chooses $\mathsf{hc} \xleftarrow{R} \mathcal{GL}$, $\alpha, \beta \xleftarrow{R} \mathbb{Z}_q$ and $r_i \xleftarrow{R} \mathbb{Z}_q$ for $i \in [n]$, sets $v := g_1^\alpha$, $w := e(g_1, g_2)^{\alpha\beta}$ and $u_i := g_1^{r_i}$ for $i \in [n]$, and outputs $\mathsf{pp} := (\mathsf{pp_{bm}}, v, w, u_1, ..., u_n)$ and $\mathsf{msk} := (g_2^{\alpha\beta}, r_1, ..., r_n)$. All other algorithms implicitly include pp as an input. The message space is $\{0,1\}$ and the vector space $\mathbb{Z}_q^n$.

$\mathsf{KeyGen}(\mathsf{msk}, \boldsymbol{y} = (y_1, ..., y_n))$**:** It chooses $s \xleftarrow{R} \mathbb{Z}_q$, sets $k_0 := g_2^s$, $k_1 := g_2^{\alpha\beta} \cdot (g_2^{\sum_{i=1}^n y_i r_i})^s$, and outputs $\mathsf{sk}_{\boldsymbol{y}} := (k_0, k_1)$.

$\mathsf{Enc}(\boldsymbol{x} = (x_1, ..., x_n), \mathsf{m} \in \mathbb{G}_T)$**:** It chooses $\gamma \xleftarrow{R} \mathbb{Z}_q$, computes $C_0 := g_1^\gamma$, $C_i := (v^{x_i} u_i)^\gamma$ for $i \in [n]$, and $C_\mathsf{m} := \mathsf{m} \oplus \mathsf{hc}(w^\gamma)$, and outputs $\mathsf{ct}_{\boldsymbol{x}} := (C_0, C_1, ..., C_n, C_\mathsf{m})$.

$\mathsf{Dec}(\mathsf{sk}_{\boldsymbol{y}}, \boldsymbol{y} = (k_0, k_1), \mathsf{ct}_x = (C_0, C_1, ..., C_n, C_\mathsf{m}))$**:** It outputs $\mathsf{m} := C_\mathsf{m} \oplus \mathsf{hc}(e(C_0, k_1) e(\prod_{i=1}^n (C_i^{y_i}), k_0)^{-1})$.

Correctness. Let $\boldsymbol{x} \in \mathbb{Z}_q^n$ and $\boldsymbol{y} \in \mathbb{Z}_q^n$ be vectors such that $\boldsymbol{x}^T \cdot \boldsymbol{y} = 0$ and $\mathsf{m} \in \{0,1\}$ be any message. Suppose that $\mathsf{ct}_{\boldsymbol{x}} = (C_0, C_1, ..., C_n, C_\mathsf{m})$ and $\mathsf{sk}_{\boldsymbol{y}} = (k_1, k_2)$ are generated as $(\mathsf{msk}, \mathsf{pp}) \xleftarrow{R} \mathsf{Setup}(1^\lambda, 1^n)$, $\mathsf{ct}_{\boldsymbol{x}} \xleftarrow{R} \mathsf{Enc}(\boldsymbol{x}, \mathsf{m})$, and $\mathsf{sk}_{\boldsymbol{y}} \xleftarrow{R} \mathsf{KeyGen}(\mathsf{msk}, \boldsymbol{y} = (y_1, ..., y_n))$. Then we have

$$\begin{aligned}&e(C_0, k_1) \cdot e(\prod_{i=1}^n (C_i^{y_i}), k_0)^{-1}\\ &= e(g_1^\gamma, g_2^{\alpha\beta + s\sum_{i=1}^n y_i r_i}) \cdot e(g_1^{\sum_{i=1}^n y_i \gamma(\alpha x_i + r_i)}, g_2^s)^{-1}\\ &= e(g_1, g_2)^{\alpha\beta\gamma + \gamma s \sum_{i=1}^n y_i r_i} \cdot e(g_1, g_2)^{-\gamma s(\alpha \sum_{i=1}^n x_i y_i + \sum_{i=1}^n y_i r_i)}\\ &= e(g_1, g_2)^{\alpha\beta\gamma}.\end{aligned}$$

Thus, the decryption correctly works since $w^\gamma = e(g_1, g_2)^{\alpha\beta\gamma}$.

Key-compactness. A secret key $\mathsf{sk}_{\boldsymbol{y}}$ for a vector $\boldsymbol{y}$ consists of two group elements of $\mathbb{G}_2$, and its size is independent from the demension n. Therefore the scheme is fully key-compact.

Security.

Theorem 4. *If the CBDH assumption holds, then the above scheme is selectively secure.*

Proof. Suppose that there exists a PPT adversary $\mathcal{A} = ((\mathcal{A}_{1\text{-}1}, \mathcal{A}_{1\text{-}2}), \mathcal{A}_2)$ that breaks the selective security of the above IPE scheme. We construct a PPT algorithm $\mathcal{B}$ that breaks the hardcore security of CBDH as follows.

$\mathcal{B}(g_1, g_1^\alpha, g_1^\beta, g_1^\gamma, g_2, g_2^\alpha, g_2^\beta, g_2^\gamma, \mathsf{hc}, T)$**:** The goal of $\mathcal{B}$ is to distinguish if $T = \mathsf{hc}(e(g_1, g_2)^{\alpha\beta\gamma})$ or $T \xleftarrow{R} \{0,1\}$. It first runs $(\boldsymbol{x}^*, \mathsf{st}_{\mathcal{A},\mathsf{pre}}) \xleftarrow{R} \mathcal{A}_{1\text{-}1}(1^\lambda, 1^n)$. Then it picks $r_i' \xleftarrow{R} \mathbb{Z}_q$ for $i \in [n]$, sets $v \xleftarrow{R} g_1^\alpha$, $w \coloneqq e(g_1^\alpha, g_2^\beta)$, $u_i \coloneqq g_1^{r_i'} \cdot (g_1^\alpha)^{-x_i^*}$ (this implicitly sets $r_i \coloneqq r_i' - \alpha x_i^* \mod q$), and $\mathsf{pp} \coloneqq (v, w, u_1, ..., u_n)$, and runs $(\mathsf{m}_0, \mathsf{m}_1, \mathsf{st}) \xleftarrow{R} \mathcal{A}_{1\text{-}2}^{\mathsf{KeyGen}(\mathsf{msk},\cdot)}(\mathsf{pp}, \mathsf{st}_{\mathcal{A},\mathsf{pre}})$ where the way to simulate the oracle $\mathsf{KeyGen}(\mathsf{msk}, \cdot)$ is described below. Then $\mathcal{B}$ picks $\mathsf{coin} \xleftarrow{R} \{0,1\}$, sets $C_0^* \coloneqq g^\gamma$, $C_i^* \coloneqq (g^\gamma)^{r_i'}$ for $i \in [n]$, $C_\mathsf{m}^* \coloneqq \mathsf{m}_\mathsf{coin} \oplus T$, and $\mathsf{ct}^* \coloneqq (C_0^*, C_1^*, ..., C_n^*, C_\mathsf{m}^*)$, and runs $\widehat{\mathsf{coin}} \xleftarrow{R} \mathcal{A}_2^{\mathsf{KeyGen}(\mathsf{msk},\cdot)}(\mathsf{ct}^*, \mathsf{st})$ where the way to simulate the oracle $\mathsf{KeyGen}(\mathsf{msk}, \cdot)$ is described below. Finally, $\mathcal{B}$ outputs $(\widehat{\mathsf{coin}} \stackrel{?}{=} \mathsf{coin})$.

$\mathsf{KeyGen}(\mathsf{msk}, \cdot)$**:** Here, we describe the way to simulate $\mathsf{KeyGen}(\mathsf{msk}, \cdot)$ by $\mathcal{B}$. Given a key query $\boldsymbol{y} = (y_1, ..., y_n)$, it first computes $\eta \coloneqq (\boldsymbol{x}^*)^T \cdot \boldsymbol{y}$. If $\eta = 0$, then it aborts. Otherwise it picks $s' \xleftarrow{R} \mathbb{Z}_q$, sets $k_0 \coloneqq g_2^{s'} \cdot (g_2^\beta)^{1/\eta}$ and $k_1 \coloneqq (g^{\sum_{i=1}^n y_i r_i'} \cdot (g^\alpha)^{-\eta})^{s'} \cdot (g^\beta)^{\sum_{i=1}^n y_i r_i'/\eta}$, and returns $\mathsf{sk}_{\boldsymbol{y}} \coloneqq (k_0, k_1)$. We omit sub/super-script of $\sum_{i=1}^n$ below for ease of notation. Now, we set $s \coloneqq s' + \beta/\eta$, then we can rewrite

$$\begin{aligned} k_1 &= g^{s' \sum y_i(r_i + \alpha x_i^*) - s'\alpha\eta + \beta/\eta \sum y_i(r_i + \alpha x_i^*)} \\ &= g^{(s' + \beta/\eta) \sum y_i r_i + s'\alpha(\sum y_i x_i^* - \eta) + \alpha\beta/\eta \sum y_i x_i^*} \\ &= g^{s \sum y_i r_i + \alpha\beta} \quad (\because (\boldsymbol{x}^*)^T \cdot \boldsymbol{y} = \sum y_i x_i^* = \eta) \end{aligned}$$

This perfectly simulate secret keys.
For the target ciphertext, $C_0^* = g^\gamma$, and for $i = 1, ..., n$, we have

$$\begin{aligned} C_i^* &= (g^{r_i'})^\gamma \\ &= (g_1^{\alpha x_i^*} \cdot g_1^{r_i' - \alpha x_i^*})^\gamma \\ &= (v^{x_i^*} u_i)^\gamma \end{aligned}$$

If $T = \mathsf{hc}(e(g_1, g_2)^{\alpha\beta\gamma})$, then C_m^* is also simulated correctly. On the othere hand, if $T \xleftarrow{R} \{0,1\}$, no information of coin is given to $\mathcal{A}$, and thus the probability that $\mathcal{B}$ outputs 1 is 1/2. Therefore we have

$$\Pr[1 \xleftarrow{R} \mathcal{B} | T = \mathsf{hc}(e(g_1, g_2)^{\alpha\beta\gamma})] - \Pr[1 \xleftarrow{R} \mathcal{B} | T \xleftarrow{R} \{0,1\}] = \frac{\mathsf{Adv}_{\mathsf{IPE},\mathcal{A}}^{\mathsf{CPA}}(\lambda)}{2}.$$

i Thus, $\mathcal{B}$ can break the hardcore security of CBDH if $\mathcal{A}$ breaks the selective security of the IPE scheme. This immediately implies that if the CBDH assumption holds, then the scheme is selectively secure by Lemma 3.

■

If we use the DBDH assumption, we can set the message space of the scheme to $\mathbb{G}_T$.