Between Hashed DH and Computational DH: Compact Encryption from Weaker Assumption

Goichiro Hanaoka[*] Kaoru Kurosawa[†]

Abstract

In this paper, we introduce the *intermediate hashed Diffie-Hellman* (IHDH) assumption which is weaker than the hashed DH (HDH) assumption (and thus the decisional DH assumption), and is stronger than the computational DH assumption. We then present two public key encryption schemes with short ciphertexts which are both chosen-ciphertext secure under this assumption. The short-message scheme has smaller size of ciphertexts than Kurosawa-Desmedt (KD) scheme, and the long-message scheme is a KD-size scheme (with arbitrary plaintext length) which is based on a weaker assumption than the HDH assumption.

Key words: public key encryption, chosen-ciphertext security, Diffie-Hellman assumption

1 Introduction

1.1 Background

The design of public-key encryption (PKE) schemes without random oracles is a central subject in modern cryptography. It should satisfy both high *security* and high *efficiency* at the same time. High security means chosen ciphertext (CCA) security [36, 21] under a reasonable assumption which is as weak as possible.

The decisional Diffie-Hellman (DDH) assumption, the hashed Diffie-Hellman (HDH) assumption and the computational Diffie-Hellman (CDH) assumption are well known such assumptions, where the HDH assumption is weaker than the DDH assumption and stronger than the CDH assumption. That is, $DDH < HDH < CDH$, where "$X < Y$" denotes that assumption Y always holds if assumption X holds.

Cramer and Shoup [19] showed the first practical CCA-secure encryption scheme under the DDH assumption. They extended it to a hybrid encryption scheme in which a message can be any bit string of arbitrary length [38, 20]. We call their hybrid encryption scheme CS scheme. They also formalized a notion of KEM-DEM framework to prove the security of hybrid encryption schemes.

Kurosawa and Desmedt next constructed a more efficient scheme (which we call KD scheme) than CS scheme under the DDH assumption. This scheme violated a then commonly held belief such that both the KEM part and the DEM part must be CCA-secure. (Indeed, the KEM part of KD scheme is not CCA-secure as shown by [24].)

CS scheme and KD scheme are often used as reference schemes to measure the efficiency of other hybrid encryption schemes. For evaluating efficiency, ciphertext size is one of the most important evaluation items, and in this paper, we mainly discuss PKE schemes with short ciphertext length. In what follows,

[*] National Institute of Advanced Industrial Science and Technology (AIST). hanaoka-goichiro@aist.go.jp

[†] Ibaraki University. kurosawa@mx.ibaraki.ac.jp

CS-size scheme denotes a scheme whose size of ciphertexts is the same as that of CS scheme (i.e. three group elements of ciphertext overhead), and KD-size scheme denotes a scheme whose size of ciphertexts is the same as that of KD scheme (i.e. two group elements and one MAC of ciphertext overhead).

In [18], Cash, Kiltz and Shoup demonstrated a CCA-secure encryption scheme under the CDH assumption by utilizing the hardcore bits. However, the size of ciphertexts is very large. The first CS-size scheme under the CDH assumption was given by Hanaoka and Kurosawa [23]. KD-size schemes under the HDH assumption were shown by Hofheinz and Kiltz [25] and Hanaoka and Kurosawa [23] independently.

Abe, Gennaro and Kurosawa [2] introduced the tag-KEM paradigm. Hofheinz and Kiltz [25] relaxed the notion of CCA-security of KEM to constrained CCA (CCCA) security. KD scheme can be explained in each of these frameworks.

1.2 Our Contribution

In this paper, we first introduce a new intractability assumption which we call the intermediate hashed Diffie-Hellman (IHDH) assumption. The IHDH assumption states that it is hard to compute $g^{\alpha\beta}$ from g, g^{α}, g^{β} and $h(g^{\alpha\beta})$, where g is a generator of an Abelian group $\mathbb{G}$ with prime order p, and h is a function whose output length is $\frac{1}{2}\log_2 p$ bits. We then show that the IHDH assumption is weaker than the HDH assumption, and is stronger than the CDH assumption. That is, $DDH < HDH < IHDH < CDH$.

Next we present two compact encryption schemes, a short-message scheme and a long-message scheme, which are both CCA-secure under the IHDH assumption.

- The short-message scheme has smaller size ciphertexts than KD scheme. It can encrypt only short messages with logarithmic length in the security parameter (say at most $7 \sim 50$ bits long), but still be useful to encrypt passwords, for example. This is the first scheme which simultaneously yields both shorter ciphertexts than that of KD schemes and CCA-security from a weaker assumption than the HDH assumption.

- The long-message scheme can encrypt any bit string of arbitrary length. It is a KD-size scheme which is based on a weaker assumption than the HDH assumption.

We stress that in this paper, we mainly discuss ciphertext length, and computational cost is not taken into account for efficiency evaluation. In fact, our proposed schemes are less efficient than KD scheme in terms of computational cost.

As by-products, two generic techniques for designing efficient PKE schemes can also be derived from our proposed schemes. These techniques are as follows: (1)A method for constructing PKE schemes from any key encapsulation mechanism (KEM) *without using DEMs*, (2)A computationally cost-free conversion from any CCCA-secure KEM [25] into a CCA-secure KEM.

1.3 Related Works

In the literatures, methods for constructing practical CCA-secure PKE schemes with better efficiency and/or stronger security have been actively studied. Here, we briefly review previous results.

As mentioned above, the first practical CCA-secure PKE was proposed by Cramer and Shoup [19, 20], and this scheme was further improved by Kurosawa and Desmedt [30]. Hofheinz and Kiltz [25], and Hanaoka and Kurosawa [23] independently proposed variants of KD scheme whose security can be proven under the HDH assumption.

After the proposal of KD scheme, there are mainly two directions for improving it: (1) One direction is to improve efficiency by using stronger assumptions, e.g. [14, 29], and (2) the other is to relax the underlying assumption, e.g. [37, 25, 18, 23].

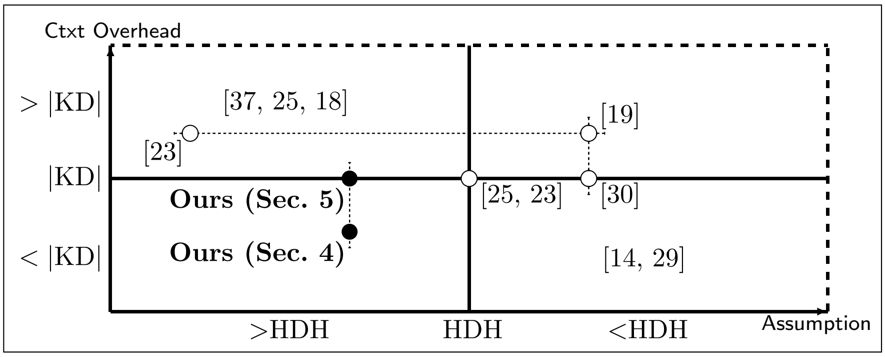

Figure 1: Relation among our proposed and existing schemes, where $|KD|$, $> |KD|$, and $< |KD|$ denote ciphertext overhead of KD scheme, larger ciphertext overhead than that of KD scheme, and smaller ciphertext overhead than that of KD scheme, respectively, and $> HDH$ and $< HDH$ are weaker and stronger assumptions than the HDH assumption, respectively.

For the first direction (i.e. direction (1)), Boyen, Mei, and Waters [14] proposed more efficient PKE schemes from the bilinear DH (BDH) assumption [9, 10, 16], and this scheme is constructed by using the Canetti-Halevi-Katz transform [17] along with specific algebraic properties of certain efficient identity-based encryption schemes [7, 39]. In [29], Kiltz presented an efficient PKE scheme whose ciphertext length is the same as that of [14], and its CCA-security is proven under the Gap HDH (GHDH) assumption. [1, 31, 13, 18, 3] yield signicantly shorther ciphertexts than the above mentioned schemes. However, these schemes require random oracles which do not exist in the real world [15].

For the second direction (i.e. direction (2)), Shacham [37] and Hofheinz and Kiltz [25] independently proposed CCA-secure PKE scheme from the n-linear assumption [8], where DDH = 1-linear < 2-linear < $\cdots < CDH$. However, the size of ciphertexts is about n times larger than that of KD scheme. Under the CDH assumption, Cash, Kiltz, and Shoup [18] proposed a CCA-secure PKE scheme. However, its ciphertext length is also significantly larger than that of CS scheme (and thus KD scheme). Hanaoka and Kurosawa presented another CCA-secure PKE scheme from the CDH assumption whose ciphertext length is the same as that of CS scheme. In [26] and [27], Hofheinz and Kiltz proposed a CCA-secure PKE scheme from the factoring assumption and the higher order residuosity assumption which are considerably weak assumptions. However, their ciphertext length are longer than that of KD scheme (over elliptic curves), and it is not known whether factoring is harder than Diffie-Hellman problems.

In summary, we see that all existing schemes are either less secure or less efficient in terms of ciphertext length than the (hashed variants of) KD scheme. Fig. 1 depicts relations among these schemes and our proposed schemes.

2 Preliminaries

2.1 PKE and CCA Security

In this subsection, we review definitions of the model and CCA-security of PKE schemes. A PKE scheme consists of the following three algorithms: **Setup**(1^k) takes as input the security parameter 1^k and outputs a decryption key dk and a public key PK. **Encrypt**(PK, M) takes as input a public key PK and a plaintext $M \in \mathcal{M}$, and outputs a ciphertext C. **Decrypt**(dk, C, PK) takes as input the private key dk, a ciphertext C, and the public key PK, and outputs M. We require that if $(dk, PK) \overset{R}{\leftarrow}$ **Setup**(1^k) and $C \overset{R}{\leftarrow}$ **Encrypt**(PK, M) then **Decrypt**$(dk, C, PK) = M$. CCA-security of a PKE scheme is defined using the following game between an attack algorithm A and a challenger. Both the challenger and A are given 1^k as input.

Setup. The challenger runs **Setup**(1^k) to obtain a decryption key dk and a public key PK, and gives PK to A.

Query I. Algorithm A adaptively issues decryption queries $C_1, \ldots, C_m$. For query C_i, the challenger responds with **Decrypt**(dk, C_i, PK).

Challenge. At some point, A submits a pair of plaintexts $(M_0, M_1) \in \mathcal{M}^2$. Then, the challenger picks a random $b \in \{0,1\}$, runs algorithm **Encrypt** to obtain the challenge ciphertext $C^\star \overset{R}{\leftarrow}$ **Encrypt**(PK, M_b), and give $C^\star$ to A.

Query II. Algorithm A continues to adaptively issue decryption queries $C_{m+1}, \ldots, C_{q_D}$. For query $C_i (\neq C^\star)$, the challenger responds as **Query I**.

Guess. Algorithm A outputs its guess $b' \in \{0,1\}$ for b and wins the game if $b = b'$.

Let $\mathsf{AdvPKE}_\mathsf{A}$ denote the probability that A wins the game.

Definition 1. We say that a PKE scheme is (τ, ϵ, q_D) *CCA-secure* if for all τ-time algorithms A who make a total of q_D decryption queries, we have that $|\mathsf{AdvPKE}_\mathsf{A} - 1/2| < \epsilon$. If a PKE scheme is $(\tau(k), \epsilon(k), q_D(k))$ CCA-secure for all polynomials $\tau(k)$ and $q_D(k)$ and some negligible function $\epsilon(k)$, we simply say that it is *CCA-secure.*

2.2 KEM and (C)CCA Security

Here, we further give a review on definitions of the model and (C)CCA-security of KEMs. A KEM consists of the following three algorithms: **Setup**(1^k) takes as input the security parameter 1^k and outputs a decryption key dk and a public key PK. **Encrypt**(PK) takes as input a public key PK and outputs a pair (ψ, K) where ψ is a ciphertext and $K \in \mathcal{K}$ is a data encryption key. **Decrypt**(dk, ψ, PK) takes as input the private key dk, a ciphertext ψ, and the public key PK, and outputs $K \in \mathcal{K}$ which will be used for decrypting the DEM part of hybrid encryption. We require that if $(dk, PK) \overset{R}{\leftarrow}$ **Setup**(1^k) and $(\psi, K) \overset{R}{\leftarrow}$ **Encrypt**(PK) then **Decrypt**$(dk, \psi, PK) = K$. In [25], Hofheinz and Kiltz proposed the notion of CCCA-security for KEMs, and showed that it is generally possible to construct a CCA-secure PKE scheme from any CCCA-secure KEM if authenticated encryption [5] is used as DEM. CCCA-security for KEM is defined as follows: Both the challenger and A are given 1^k as input.

Setup. The challenger runs **Setup**(1^k) to obtain a decryption key dk and a public key PK. The challenger also runs algorithm **Encrypt** to obtain $(\psi^\star, K^\star) \overset{R}{\leftarrow}$ **Encrypt**(PK) where $K^\star \in \mathcal{K}$. Next, the challenger picks a random $b \in \{0,1\}$. It sets $K_0 = K^\star$ and picks a random K_1 in $\mathcal{K}$. It then gives the public key PK and the challenge ciphertext $(\psi^\star, K_b)$ to algorithm A.

Query. Algorithm A adaptively issues decryption queries $(\psi_1, pred_1(\cdot)), ..., (\psi_{q_D}, pred_{q_D}(\cdot))$. For query $(\psi_i (\neq \psi^\star), pred_i(\cdot))$, the challenger responds with K(or "$\perp$")= **Decrypt**(dk, ψ_i, PK) if $pred_i(K) = 1$. It returns "$\perp$" otherwise.

Guess. Algorithm A outputs its guess $b' \in \{0,1\}$ for b and wins the game if $b = b'$.

Here, function $pred_i : \mathcal{K} \to \{0,1\}$ is called predicate, and according to $pred_1, ..., pred_{q_D}$, uncertainty $uncert_\mathsf{A}$ is estimated as $uncert_\mathsf{A} = \max_\mathsf{E} \frac{1}{q_D} \sum_{1 \le i \le q_D} \Pr_{K \in \mathcal{K}}[pred_i(K) = 1 \text{ when } \mathsf{A} \text{ runs with } \mathsf{E}]$, where E is an environment which interacts with A. See also [25] for some remarks on the definition. Let $\mathsf{AdvKEM}_\mathsf{A}$ denote the probability that A wins the game.

Definition 2. We say that a KEM is $(\tau, \epsilon, q_D, \mu)$-*CCCA-secure* if for all τ-time algorithms A who make a total of q_D decryption queries with $uncert_{\mathsf{A}} \leq \mu$, we have that $|\mathsf{AdvKEM}_{\mathsf{A}} - 1/2| < \epsilon$. Especially, we say that a KEM is (τ, ϵ, q_D)-*CCA-secure if it is $(\tau, \epsilon, q_D, 1)$-CCCA-secure. If a KEM is $(\tau(k), \epsilon(k), q_D(k)(, \mu))$-(C)CCA-secure for all polynomials $\tau(k)$ and $q_D(k)$ and some negligible function $\epsilon(k)$ (and $\mu > 0$), we simply say that it is (C)CCA-secure .*

As mentioned above, a hybrid encryption scheme from a CCCA-secure KEM and an authenticated symmetric key encryption (AE) scheme [5] becomes a CCA-secure PKE scheme [25], and that from a CCA-secure KEM and a CCA-secure data encapsulation mechanism (DEM) becomes also a CCA-secure PKE scheme [38]. It is known that an AE scheme yields a (at least) k-bit longer ciphertext than a plaintext, for the security parameter k, while ciphertext length of a CCA-secure DEM can be determined to be the same as the plaintext length if we use a *strong pseudorandom permutation* [32, 34] as a DEM.

2.3 Conventional Diffie-Hellman Assumptions

Let $\mathbb{G}$ be a multiplicative group with prime order p. Then, the CDH problem on $\mathbb{G}$ is stated as follows. Let A be an algorithm, and we say that A has advantage ϵ in solving the CDH problem on $\mathbb{G}$ if $\Pr[\mathsf{A}(g, g^{\alpha}, g^{\beta}) = g^{\alpha\beta}] \geq \epsilon$, where the probability is over the random choice of generators g in $\mathbb{G}$, the random choice of α and β in $\mathbb{Z}_p$, and the random bits consumed by A.

Definition 3. We say that the (τ, ϵ)-*CDH assumption* holds in $\mathbb{G}$ if no τ-time algorithm has advantage ϵ in solving the CDH problem on $\mathbb{G}$. If the $(\tau(k), \epsilon(k))$-CDH assumption holds for all polynomial $\tau(k)$ and some negligible function $\epsilon(k)$, we simply say that the *CDH assumption* holds.

The *hashed Diffie-Hellman* (HDH) problem on $\mathbb{G}$ and function $H : \mathbb{G} \to \mathcal{S}$ is stated as follows. Let A be an algorithm, and we say that A has advantage ϵ in solving the HDH problem on $\mathbb{G}$ and H if

$$1/2 \cdot |\Pr[\mathsf{A}(g, g^{\alpha}, g^{\beta}, H(g^{\alpha\beta})) = 0] - \Pr[\mathsf{A}(g, g^{\alpha}, g^{\beta}, T) = 0]| \geq \epsilon,$$

where the probability is over the random choice of generators g in $\mathbb{G}$, the random choice of α and β in $\mathbb{Z}_p$, the random choice of $T \in \mathcal{S}$, and the random bits consumed by A.

Definition 4. We say that the (τ, ϵ)-*HDH assumption* holds in $\mathbb{G}$ and H if no τ-time algorithm has advantage ϵ in solving the HDH problem on $\mathbb{G}$ and h. Especially, we say that the (τ, ϵ)-*DDH assumption* holds in $\mathbb{G}$ if (τ, ϵ)-HDH assumption holds in $\mathbb{G}$ and H, where H is the identity function. If the $(\tau(k), \epsilon(k))$-HDH (resp. DDH) assumption holds for all polynomial $\tau(k)$ and some negligible function $\epsilon(k)$, we simply say that the *HDH (resp. DDH) assumption* holds.

It is known that the DDH assumption implies the HDH assumption if H is a key derivation function (KDF) [38], but not vice versa.

2.4 Other Tools

Target Collision Resistant Hash Functions. Let $\mathsf{TCR} : \mathcal{X} \to \mathcal{Y}$ be a hash function, A be an algorithm, and A's advantage be $\Pr[\mathsf{TCR}(x') = \mathsf{TCR}(x) \in \mathcal{Y} \wedge x' \neq x |\ x \overset{R}{\leftarrow} \mathcal{X};\ x' \overset{R}{\leftarrow} \mathsf{A}(x)]$, where the probability is over the random choice of x in $\mathbb{G}$ and the random bits consumed by A.

Definition 5. We say that TCR is a (τ, ϵ)-*target collision resistant hash function* (TCRHF) if no τ-time algorithm has advantage ϵ. If it is a $(\tau(k), \epsilon(k))$-TCRHF for all polynomial $\tau(k)$ and some negligible function $\epsilon(k)$, we simply say that it is a *TCRHF*.

Key Derivation Functions. Let $H : \mathcal{X} \to \mathcal{Y}$ be a function, A be an algorithm, and A's advantage be $1/2 \cdot |\Pr[\mathsf{A}(H(x)) = 0] - \Pr[\mathsf{A}(T) = 0]|$, where the probability is over the random choice of x in $\mathcal{X}$ and the random choice of $T \in \mathcal{Y}$, and the random bits consumed by A.

Definition 6. We say that H is a (τ, ϵ)-*key derivation function* (KDF) if no τ-time algorithm has advantage ϵ. If it is a $(\tau(k), \epsilon(k))$-KDF for all polynomial $\tau(k)$ and some negligible function $\epsilon(k)$, we simply say that it is a *KDF.*

3 New Assumptions between CDH and DDH

3.1 IDH Assumption and IHDH Assumption

Let $\mathbb{G}$ be a cyclic group of prime order p, and let g be a generator. It is known that the discrete log problem is solved in time $O(\sqrt{p})$ by using Pollard rho algorithm. Hence we say that $\mathbb{G}$ has k-bit security when $\sqrt{p}$ is k-bit long. In what follows, assume that $\mathbb{G}$ has k-bit security, that is, p is $2k$-bit long. The relationship between KD scheme and Pollard rho algorithm is also discussed at Appendix E. Now we introduce the intermediate DH assumption (IDH assumption) and the intermediate hashed DH assumption (IHDH assumption). Each assumption states that it is hard to compute $g^{\alpha\beta}$ from g, g^{α}, g^{β} and K, where K is some k-bit information of $g^{\alpha\beta}$. In the IDH assumption, K is the k most significant bits of $g^{\alpha\beta}$. In the IHDH assumption, $K = h(g^{\alpha\beta})$ for some hash function $h : \mathbb{G} \to \{0,1\}^k$. The IHDH assumption is defined more formally as follows. Let A be an algorithm, and we say that A has advantage ϵ in solving the IHDH problem on $\mathbb{G}$ and $h : \mathbb{G} \to \{0,1\}^k$ if

$$\Pr[\mathsf{A}(g, g^{\alpha}, g^{\beta}, h(g^{\alpha\beta})) = g^{\alpha\beta}] \geq \epsilon$$

where the probability is over the random choice of generator g in $\mathbb{G}$, the random choice of α and β in $\mathbb{Z}_p$, and the random bits consumed by A.

Definition 7. We say that the (τ, ϵ)-*IHDH assumption* holds in $\mathbb{G}$ and h if no τ-time algorithm has advantage ϵ in solving the IHDH problem on $\mathbb{G}$ and h. If the $(\tau(k), \epsilon(k))$-IHDH assumption holds for all polynomial $\tau(k)$ and some negligible function $\epsilon(k)$, we simply say that the *IHDH assumption* holds.

It is easy to prove that for k-bit security, the IHDH assumption on $\mathbb{G}$ and $h : \mathbb{G} \to \{0,1\}^k$ is implied by the HDH assumption in $\mathbb{G}$ and $H : \mathbb{G} \to \{0,1\}^{2k}$ (and thus the DDH assumption if H is a KDF) if $h(x)$ maps to k most significant bits of $H(x)$ for all $x \in \mathbb{G}$. For simplicity, throughout this paper k-$\mathsf{msb}(x)$ and k-$\mathsf{lsb}(x)$ denote k most and least significant bits of x, respectively. We note that for typical cases H does not need to yield a larger range than its domain since an element of $\mathbb{G}$ is already $2k$-bit long for the same security level. The IDH assumption is directly implied by the DDH assumption. More formally, these implications are addressed as follows:

Theorem 1. *If the (τ, ϵ_{hdh})-HDH assumption on $\mathbb{G}$ and $H : \mathbb{G} \to \{0,1\}^{2k}$ holds, then the $(\tau, 2\epsilon_{hdh} + 1/2^{-k})$-IHDH assumption on $\mathbb{G}$ and h also holds, where $h(x) = k$-$\mathsf{msb}(H(x))$ for all $x \in \mathbb{G}$.*

Proof. For proving the theorem, by using an algorithm A which for given $(g, g^{\alpha}, g^{\beta}, h(g^{\alpha\beta}))$ computes $g^{\alpha\beta}$ with non-negligible probability, we construct another algorithm B which for given $(g, g^{\alpha}, g^{\beta}, Z)$ distinguishes whether $Z = H(g^{\alpha\beta})$ or not with non-negligible advantage by using A.

Algorithm B is constructed as follows: For given $(g, g^{\alpha}, g^{\beta}, Z)$, B computes $\omega = k$-$\mathsf{msb}(Z)$ and inputs $(g, g^{\alpha}, g^{\beta}, \omega)$ to A. Let Z' be A's output. Then, B outputs "$Z = H(g^{\alpha\beta})$" if and only if $H(Z') = Z$. We note that $(g, g^{\alpha}, g^{\beta}, \omega)$ is perfectly indistinguishable from a randomly chosen instance of the IHDH

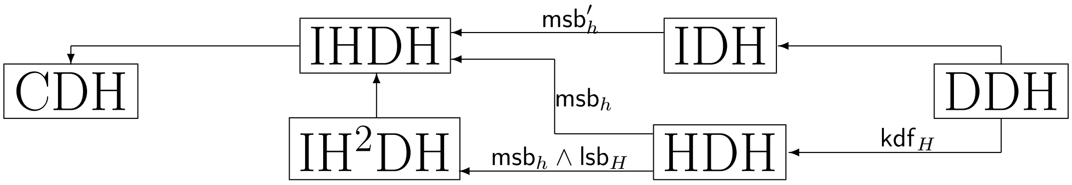

Figure 2: Implications among variants of the Diffie-Hellman assumptions, where "$A \overset{\mathsf{xxx}}{\longleftarrow} B$" denotes that assumption A is implied by assumption B under condition xxx for h and/or H. Conditions for h and H are as follows: kdf_H denotes that H is a KDF, msb_h and lsb_H denote that $h(x) = k\text{-}\mathsf{msb}(H'(x))$ and $H(x) = k\text{-}\mathsf{lsb}(H'(x))$, respectively, for all $x \in \mathbb{G}$ assuming that the HDH assumption holds in $\mathbb{G}$ and $H' : \mathbb{G} \to \{0,1\}^{2k}$, msb'_h denotes that for some function $h' : \{0,1\}^k \to \{0,1\}^k$, $h(x) = h'(k\text{-}\mathsf{msb}(x))$ for all $x \in \mathbb{G}$.

problem if $Z = H(g^{\alpha\beta})$, and therefore, A will output $g^{\alpha\beta}$. On the other hand, A may notice that $(g, g^\alpha, g^\beta, \omega)$ is not in a valid form of an instance of the IHDH problem. In this case, B cannot know A's behavior, and A may output whatever it likes. However, Z is picked uniformly random from $\{0,1\}^{2k}$, and consequently even if k-$\mathsf{msb}(Z)$ is exposed to A, the remained part of Z is still information-theoretically hidden to A. Hence, $H(Z') = Z$ happens with probability $1/2^k$. □

Since the DDH assumption implies the HDH assumption, we also have the following corollary.

Corollary 1. *If the (τ, ϵ_{ddh})-DDH assumption on $\mathbb{G}$ holds and $H : \mathbb{G} \to \{0,1\}^{2k}$ is a KDF, then the $(\tau, 2\epsilon_{ddh} + 2\epsilon_{kdf} + 1/2^k)$-IHDH assumption on $\mathbb{G}$ and h holds, where $h(x) = k\text{-}\mathsf{msb}(H(x))$ for all $x \in \mathbb{G}$.*

3.2 Decisional Variants

As usual, we can define a decisional (and hashed) version of the IHDH assumption. The *hashed* IHDH (IH^{2}DH) problem on $\mathbb{G}$, function $H : \mathbb{G} \to \mathcal{S}$, and function $h : \mathbb{G} \to \{0,1\}^k$ is stated as follows. Let A be an algorithm, and we say that A has advantage ϵ in solving the IH^{2}DH problem on $\mathbb{G}$, H, and h if

$$1/2 \cdot |\Pr[\mathsf{A}(g, g^\alpha, g^\beta, h(g^{\alpha\beta}), H(g^{\alpha\beta})) = 0] - \Pr[\mathsf{A}(g, g^\alpha, g^\beta, h(g^{\alpha\beta}), T) = 0]| \geq \epsilon,$$

where the probability is over the random choice of generators g in $\mathbb{G}$, the random choice of α and β in $\mathbb{Z}_p$, the random choice of $T \in \mathcal{S}$, and the random bits consumed by A.

Definition 8. We say that the (τ, ϵ)-*IH^{2}DH assumption* holds in $\mathbb{G}$, H, and h if no τ-time algorithm has advantage ϵ in solving the IH^{2}DH problem on $\mathbb{G}$, H, and h. If the $(\tau(k), \epsilon(k))$-IH^{2}DH assumption holds for all polynomial $\tau(k)$ and some negligible function $\epsilon(k)$, we simply say the *IH^{2}DH assumption* holds.

Fig. 2 illustrates implications among the above security notions, and we see that the IHDH assumption is implied by the other ones except for the CDH assumption under natural conditions. Proofs for the implications can be straightforwardly done. We stress that requirements for hash function h in the IHDH assumption is considerably mild and for example, least significant bits of the output of a KDF is sufficient.

3.3 Relationship between IH^{2}DH and Hardcore of IHDH

The IHDH assumption implies the IH^{2}DH assumption if H is a hardcore function of the IHDH problem. To be more precise, we define a hardcore function for the IHDH problem as follows. Let A be a τ-time algorithm which has advantage ϵ in solving the IH^{2}DH problem on $\mathbb{G}$, H and h.

Definition 9. We say that function H is a *hardcore function for the IHDH problem* on $\mathbb{G}$ and h if there exists a $p_1(\tau)$-time algorithm B^{A} which can solve the IHDH problem on $\mathbb{G}$, H and h with advantage $p_2(\epsilon)$ for some polynomials p_1 and p_2 for any A.

By using Goldreich-Levin (GL) function [22], hardcore functions can be generically constructed for any one-way function. The construction of the GL function H is as follows: For a given function f, pick a random binary string R where $|R|$ is the same as the input length of f, and define $H(x) = \langle R, x\rangle$ where $\langle R, x\rangle$ denotes the innerproduct of R and the binary representation of x. Then, if there exists an algorithm which for given $f(x)$, distinguishes $H(x) = 0$ or 1 with a non-negligibly better probability than 1/2, there always exists another algorithm which for given $f(x)$, outputs x with a non-negligible probability. For constructing a hardcore function for the IHDH problem, we set $f(g^{\alpha}, g^{\beta}, g^{\alpha\beta}) = (g^{\alpha}, g^{\beta}, h(g^{\alpha\beta}))$ (if an input is not in a valid form, its corresponding output is "$\perp$"). For this particular one-way function, H can be set as $H(g^{\alpha}, g^{\beta}, g^{\alpha\beta}) = \langle R, g^{\alpha\beta}\rangle$ where $|R| = |g^{\alpha\beta}|$. This technique can be straightforwardly extend for extracting $c \log k$ hardcore bits from one DH key, where c is a constant value. See also Appendix of [12] for the GL function for the Diffie-Hellman type keys. On a well-designed elliptic curve, an element of a group can be represented by the x-coordinate of its corresponding point. Therefore, throughout this paper, we assume that $g^{\alpha\beta}$ is encoded as an element in $\mathbb{Z}_q$ where q is close to p (i.e. $|q| = |p|$), and that this representation is used for computing innerproducts. We call a hardcore function which is constructed in such a way the *GL hardcore function for the IHDH problem.* For constructing hardcore functions for the IHDH problem, there are also some other candidates which can be used for generating hardcore bits of Diffie-Hellman type keys [12, 11, 28].

4 Short Message Encryption Scheme from IHDH

In this section, we propose a new PKE scheme from the IHDH assumption. This is the first PKE scheme which simultaneously yields (i) CCA-security under a weaker assumption than the HDH assumption, and (ii) shorter ciphertext length than the KD hybrid encryption scheme. Our scheme is constructed by non-straightforwardly extending the Hanaoka-Kurosawa KEM which is CCCA-secure under the HDH assumption. Some techniques in this extension can be generically applied to other CCCA (or CCA) secure KEMs for enhancing their efficiency, and these generic techniques are discussed in Sec. 6.

The Construction. Let $\mathbb{G}$ be a multiplicative group with prime order p, and $g \in \mathbb{G}$ be a generator, where $|p| = 2k$ for the security parameter k. Then, the construction of our PKE scheme is as follows:

Setup(1^k)**:** Generate a random polynomial $f(x) = a_0 + a_1 x + a_2 x^2$ over $\mathbb{Z}_p$, and compute $y_j = g^{a_j}$ for $0 \le j \le 2$. The decryption key is $f(x)$, and the public key is $PK = (\mathbb{G}, g, y_0, y_1, y_2, \mathsf{TCR}, h, H)$, where $\mathsf{TCR} : \mathbb{G} \to \mathbb{Z}_p^*$ is a TCRHF, $h : \mathbb{G} \to \{0,1\}^k$ is a KDF, and $H : \mathbb{G} \to \{0,1\}^{c \cdot \log k}$ is the GL hardcore function for the IHDH problem on $\mathbb{G}$ and h (see Sec. 3.3).

Encrypt(M, PK)**:** For a plaintext $M \in \{0,1\}^{c \cdot \log k}$, pick a random $r \overset{R}{\leftarrow} \mathbb{Z}_p$, and check whether $M \overset{?}{=} H(y_0^r)$. Then, compute $C = (g^r, (y_0 y_1^i y_2^{i^2})^r, h(y_0^r))$ if the equality holds, where $i = \mathsf{TCR}(g^r)$, or repeat the same procedure with another r otherwise. (Computational cost for finding such r is almost equivalent to that for $\frac{\log 2k + k^c}{\log 2k + 1}$ exponentiations [35]. See also Claim 1 which guarantees uniformity of $H(y_0^r)$ for randomly chosen r.)

Decrypt(dk, C, PK)**:** For a ciphertext $C = (C_0, C_1, C_2)$, check whether $C_0^{f(i)} \overset{?}{=} C_1 \wedge h(C_0^{a_0}) \overset{?}{=} C_2$, where $i = \mathsf{TCR}(C_0)$. If not, output $\perp$. Otherwise, output $H(C_0^{a_0}) = M$.

We see that a ciphertext of the above scheme consists of two group elements and k-bit binary string which results in a $(c \cdot \log k)$-bit shorter ciphertext than that of the KD hybrid encryption scheme. Drawbacks of the proposed scheme are that the size of a plaintext is only $(c \cdot \log k)$-bit long and that its computational cost for encryption is expensive. As mentioned, CCA-security of our scheme can be proven under the IHDH assumption which is weaker than the HDH assumption (and thus the DDH assumption). Security of our scheme is formally addressed as follows:

Theorem 2. *Let $\mathbb{G}$ be a multiplicative group with prime order p, TCR be a TCRHF, h be a KDF, and H be the GL hardcore function for the IHDH problem on $\mathbb{G}$ and h. Then, the above PKE scheme is CCA-secure under the IHDH assumption on $\mathbb{G}$ and h.*

In the above theorem, we address the security of the proposed scheme without explicitly giving the reduction cost since it is loose and complicated due to the use of the GL hardcore function. More precise reduction cost can be estimated from Lemma 2. An immediate corollary of Theorem 2 is that the above scheme is also secure under the IDH assumption (this is clear since the IDH assumption is stronger than the IHDH assumption if its underlying hash functions are appropriately chosen).

The proof of Theorem 2 can be trivially obtained by proving Lemmas 1 and 2, where Lemma 1 guarantees that the GL hardcore function H has a specific property (besides the hardcoreness) which is required for the proof of Lemma 2, and Lemma 2 addresses CCA-security of the above scheme under that specific property and the IH^{2}DH assumption.

Lemma 1. *Suppose that a random variable x is uniformly distributed over $\mathbb{Z}_p$. Then, for the GL hardcore function $H(x)$, the most and least likely values of $H(x) \in \{0,1\}^{c \log k}$ are known a priori. More specifically, $M = (0,0,...,0)$ and $(1,1,...,1)$ maximizes and minimizes $\Pr_x[H(x) = M]$, respectively. (Proof of this lemma is given in Appendix A.)*

Lemma 2. *Let $\mathbb{G}$ be a multiplicative group with prime order p, TCR be a (τ, ϵ_{tcr})-TCRHF, and h be a (τ, ϵ_{kdf})-KDF. Also, assume that most and least likely values of $H(x)$ for randomly chosen $x \in \mathbb{G}$ is known a priori. Then, the above scheme is $(\tau - o(\tau), \epsilon_{cca}, q_D)$-CCA-secure under the $(\tau, \epsilon_{ih^2dh})$-IH^{2}DH assumption*[1] *on $\mathbb{G}$, H and h, where*

$$\epsilon_{cca} = \frac{6\epsilon_{ih^2dh} + \epsilon_{tcr} + q_D(\epsilon_{kdf} + 1/2^k + 3/(p-3))}{k^{-c} - 4\epsilon_{ih^2dh}}.$$

Proof. Assume we are given an adversary A which breaks CCA-security of the above PKE scheme with running time τ, advantage ϵ, and q_D decryption queries. We use A to construct another adversary B which solves the IH^{2}DH problem on $\mathbb{G}$, H and h. Define adversary B as follows:

1. For a given IH^{2}DH instance $(g, g^\alpha, g^\beta, h(g^{\alpha\beta}), Z)$, B picks a TCRHF TCR, and computes $i^\star = \mathsf{TCR}(g^\beta)$. ($Z$ is $H(g^{\alpha\beta})$ or a random k-bit string.)
2. B sets $y_0 = g^\alpha$, and picks a random rnd from $\mathbb{Z}_p^* \backslash \{i^\star\}$. B also picks randoms $u_{i^\star}$ and u_{rnd} from $\mathbb{Z}_p$.
3. Let $f(x) = \alpha + a_1 x + a_2 x^2$ be a polynomial over $\mathbb{Z}_p$ such that $f(i^\star) = u_{i^\star}, f(rnd) = u_{rnd}$. Note that each a_i can be expressed as a linear combination of $\alpha, u_{i^\star}$ and u_{rnd} by using Lagrange formula. B then computes $y_i = g^{a_i}$ for $i = 1, 2$ by using $y_0 = g^\alpha$.
4. B inputs public key $PK = (\mathbb{G}, g, y_0, y_1, y_2, \mathsf{TCR}, h, H)$ to A.

[1] We consider only the case where $k^{-c} \gg 4\epsilon_{ih^2dh}$, and this is sufficient for all interesting parameter settings.

5. When A makes decryption query $C = (C_0, C_1, C_2)$, B proceeds as follows:
   (a) If $C_0 = g^\beta$, then B responds $\perp$.
   (b) If $C_0 \neq g^\beta$ and $\mathsf{TCR}(C_0) \in \{i^\star, rnd\}$, then B aborts and outputs a random bit.
   (c) If $C_0 \neq g^\beta$ and $\mathsf{TCR}(C_0) \notin \{i^\star, rnd\}$, B computes $C_0^{u_{i^\star}}$ and $C_0^{u_{rnd}}$. Let $\mathsf{TCR}(C_0) = i$ and f' be polynomials over $\mathbb{Z}_p$ with degree two, such that $(f'(i), f'(i^\star), f'(rnd)) = (\log_{C_0} C_1, u_{i^\star}, u_{rnd})$. Then, B calculates ${C_0}^{f'(0)}$ by using the Lagrange interpolation from $(C_1, C_0^{u_{i^\star}}, C_0^{u_{rnd}})$. B responds $H({C_0}^{f'(0)})$ if $h({C_0}^{f'(0)}) = C_2$, or "$\perp$" otherwise.

6. At some point, A queries a pair of plaintexts $M_0, M_1 \in \{0,1\}^{c \cdot \log k}$. Then, B picks a random bit b. If M_b is identical to Z, then B generates a challenge ciphertext $C^\star = (g^\beta, (g^\beta)^{u_{i^\star}}, h(g^{\alpha\beta}))$, and sends it to A. Otherwise, it aborts and outputs a random bit. We note that $C^\star$ is a correct challenge ciphertext if $Z = H(g^{\alpha\beta})$.

7. Finally, A outputs his guess, and B outputs 0 (i.e. "$Z = H(g^{\alpha\beta})$") if and only if A's output is identical to b.

Let Win denote the event that A correctly outputs the underlying bit of the challenge ciphertext in the real world, Abort denote the event that for the challenge ciphertext $C^\star = (C_0^\star, C_1^\star, C_2^\star)$ and a random $rnd \in \mathbb{Z}_p^* \backslash \{\mathsf{TCR}(C_0^\star)\}$, A submits a ciphertext $C = (C_0, C_1, C_2)$ such that $C_0 \neq C_0^\star$ and $\mathsf{TCR}(C_0) \in \{\mathsf{TCR}(C_0^\star), rnd\}$, and $\mathsf{Invalid}$ denote the event that A submits a ciphertext $C = (C_0, C_1, C_2)$ which is rejected in the real world, but not in the above simulation. More precisely, C is a ciphertext such that $\mathsf{TCR}(C_0) \notin \{\mathsf{TCR}(C_0^\star), rnd\}$, $h(C_0^{f'(0)}) = C_2$, but $C_1 \neq C_0^{f(i)}$. Let Embed denote the event that the plaintext of the challenge ciphertext is $H(x) \in \{0,1\}^{c \cdot \log k}$ for randomly chosen $x \in \mathbb{G}$. In other words, Embed is the event that M_b happens to be $H(g^{\alpha\beta})$ in the simulation. Then, B's advantage in solving the IH^{2}DH problem is estimated as follows:

$$
\begin{aligned}
&\frac{1}{2} \cdot |\Pr[\mathsf{B}(g, g^\alpha, g^\beta, h(g^{\alpha\beta}), H(g^{\alpha\beta})) = 0] - \Pr[\mathsf{B}(g, g^\alpha, g^\beta, h(g^{\alpha\beta}), T) = 0]| \\
&\qquad\qquad \geq \frac{1}{2} \cdot |\Pr[\mathsf{Win} \wedge \mathsf{Embed} \wedge \overline{\mathsf{Abort}} \wedge \overline{\mathsf{Invalid}}] + \frac{1}{2}\Pr[\overline{\mathsf{Embed}}] - \frac{1}{2}| \\
&\qquad\qquad \geq \frac{1}{2} \cdot |\Pr[\mathsf{Win} \wedge \mathsf{Embed}] - \Pr[\mathsf{Abort}] - \Pr[\mathsf{Invalid}] - \frac{1}{2}\Pr[\mathsf{Embed}]|.
\end{aligned}
$$

We note that if $Z \neq H(g^{\alpha\beta})$, then A may notice that the given environment is a simulation. However, it cannot carry out anything except for outputting a random bit since it is information-theoretically impossible to distinguish whether $Z = M_0$ or M_1. Therefore, $\Pr[\mathsf{B}(g, g^\alpha, g^\beta, h(g^{\alpha\beta}), T) = 0] = 1/2$. The proof completes by proving following claims.

Claim 1. *For any $M \in \{0,1\}^{c \cdot \log k}$, $k^{-c} - 4\epsilon_{ih^2dh} \leq \Pr_{x \in \mathbb{G}}[H(x) = M] \leq k^{-c} + 4\epsilon_{ih^2dh}$ assuming the $(\tau, \epsilon_{ih^2dh})$-IH^{2}DH assumption holds.*

Proof. Let M' be the least likely value of $H(x)$ for randomly chosen $x \in \mathbb{G}$. (Recall that most and least likely values of $H(x)$ are assumed to be known a priori, and due to Lemma 1 this assumption always holds for the GL hardcore function.) Then, we can construct an algorithm B' which immediately solves the IH^{2}DH problem as follows: For a given IH^{2}DH instance $(g, g^\alpha, g^\beta, h(g^{\alpha\beta}), Z)$, B' outputs 1 if $Z = M'$, or a random bit otherwise.

B''s advantage is estimated as

$$
\frac{1}{2} \cdot |\Pr[\mathsf{B}'(g, g^\alpha, g^\beta, h(g^{\alpha\beta}), H(g^{\alpha\beta})) = 0] - \Pr[\mathsf{B}'(g, g^\alpha, g^\beta, h(g^{\alpha\beta}), T) = 0]|
$$

$$= \frac{1}{2} \cdot |\frac{1}{2} \cdot (1 - \Pr[H(g^{\alpha\beta}) = M']) - \frac{1}{2} \cdot (1 - k^{-c})| = \frac{1}{4} \cdot |k^{-c} - \Pr[H(g^{\alpha\beta}) = M']|.$$

Since the distribution of $g^{\alpha\beta}$ is uniform over $\mathbb{G}$, $\Pr[H(g^{\alpha\beta}) = M'] = \Pr_{x \in \mathbb{G}}[H(x) = M']$. Hence, we have that $\frac{1}{4}(k^{-c} - \Pr_{x\in\mathbb{G}}[H(x) = M']) \le \epsilon_{ih^2dh}$, and thus $\Pr_{x\in\mathbb{G}}[H(x) = M'] \le k^{-c} - 4\epsilon_{ih^2dh}$. Similarly to this, by using the most likely value of $H(x)$, we can also prove $\Pr_{x\in\mathbb{G}}[H(x) = M] \ge k^{-c} + 4\epsilon_{ih^2dh}$. □

Claim 2. $\Pr[\mathsf{Win} \wedge \mathsf{Embed}] \ge (\frac{1}{2} + \epsilon) \cdot (k^{-c} - 4\epsilon_{ih^2dh})$.

Proof. Since there is a possibility that events Win and Embed are dependent, it is not straightforward to estimate $\Pr[\mathsf{Win} \wedge \mathsf{Embed}]$ from $\Pr[\mathsf{Win}]$ and $\Pr[\mathsf{Embed}]$. This is one of the main non-trivial parts in the security proof of the proposed scheme. For individually dealing with these two dependent events, we discuss them by using conditional probabilities under the condition that M_b is fixed. Interestingly, Embed depends on only the values of M_b and Z, and therefore, under that condition Embed is independent to Win under the same condition. Namely, we have $\Pr[\mathsf{Win} \wedge \mathsf{Embed}] = \sum_{M\in\{0,1\}^{c\cdot\log k}} \Pr[\mathsf{Win}|M_b = M] \Pr[\mathsf{Embed}|M_b = M] \Pr[M_b = M]$. Since $\Pr[\mathsf{Embed}|M_b = M] = \Pr_{x\in\mathbb{G}}[H(x) = M]$, from Claim 1 we have $\Pr[\mathsf{Embed}|M_b = M] \ge k^{-c} - 4\epsilon_{ih^2dh}$. Consequently, we also have

$$\begin{aligned}\Pr[\mathsf{Win} \wedge \mathsf{Embed}] &\ge (\sum_{M\in\{0,1\}^{c\cdot\log k}} \Pr[\mathsf{Win}|M_b = M] \Pr[M_b = M]) \cdot (k^{-c} - 4\epsilon_{ih^2dh}) \\ &= \Pr[\mathsf{Win}] \cdot (k^{-c} - 4\epsilon_{ih^2dh}) = (\frac{1}{2} + \epsilon) \cdot (k^{-c} - 4\epsilon_{ih^2dh}),\end{aligned}$$

which proves the claim. □

Claim 3. $\Pr[\mathsf{Abort}] \le \epsilon_{tcr} + \frac{q_D}{p-2}$. *(Proof of this claim is given in Appendix B.)*

Claim 4. $\Pr[\mathsf{Invalid}] \le q_D \cdot (\epsilon_{kdf} + \frac{1}{2^k} + \frac{2}{p-2})$.

Proof. Suppose $C = (C_0, C_1, C_2)$ is a ciphertext such that B does not abort and $C_1 \ne C_0^{f(i)}$. Then, we notice that for any $f(x)$, $i^\star$, and i, the value $f'(0)$ takes $p-2$ different values according to $p-2$ different values for rnd. This can be easily proved by a contradiction as follows: Fix $f(x), i^\star, i$ and $u \ne f(i)$. For $rnd \in (\mathbb{Z}_p^* \backslash \{i^\star\})$, let $f_{rnd}(x)$ be a polynomial of degree at most two such that $f_{rnd}(i^\star) = f(i^\star), f_{rnd}(i) = u, f_{rnd}(rnd) = f(rnd)$. Then, we will show that for any $(rnd_1, rnd_2) \in (\mathbb{Z}_p^* \backslash \{i^\star\})^2$, $f_{rnd_1}(0) \ne f_{rnd_2}(0)$ if $rnd_1 \ne rnd_2$. Suppose that $f_{rnd_1}(0) = f_{rnd_2}(0)$. Then $f_{rnd_1}(x) = f_{rnd_2}(x)$ because they intersect at three points, $x = 0, i^\star$ and i. In this case, $f_{rnd_1}(x) = f(x)$ because they intersect at three points, $x = i^\star, rnd_1$ and rnd_2. But this is a contradiction because $f_{rnd_1}(i) = u \ne f(i)$.

Therefore, from the viewpoint of A, the distribution of $h(C_0^{f'(0)})$ is computationally indistinguishable from the uniform distribution over $\{0,1\}^k$ with advantage at least $\epsilon_{kdf} + 2/p (\le \epsilon_{kdf} + 2/(p-2))$ where $2/p$ comes from statistical distance between the distribution of $C_0^{f'(0)}$ and the uniform distribution over $\mathbb{G}$. Notice that if the distribution of $h(C_0^{f'(0)})$ is uniform over $\{0,1\}^k$, then it is information-theoretically impossible to guess the value of $h(C_0^{f'(0)})$ with probability more than $1/2^k$. Hence, $h(C_0^{f'(0)}) = C_2$ happens with probability at most $1/2^k + \epsilon_{kdf} + 2/(p-2)$. □

Claim 5. $\Pr[\mathsf{Embed}] \le k^{-c} + 4\epsilon_{ih^2dh}$. *(The claim is straightforwardly proved by Claim 1.)*

From Claims 2, 3, 4, and 5, we have

$$\begin{aligned}\epsilon_{ih^2dh} &\geq \frac{1}{2}\left((\frac{1}{2}+\epsilon)(\frac{1}{k^c}-4\epsilon_{ih^2dh})-\epsilon_{tcr}-q_D(\epsilon_{kdf}+\frac{1}{2^k}+\frac{3}{p-3})-\frac{1}{2}(\frac{1}{k^c}+4\epsilon_{ih^2dh})\right)\\ &= \frac{1}{2}\left(\epsilon(\frac{1}{k^c}-4\epsilon_{ih^2dh})-4\epsilon_{ih^2dh}-\epsilon_{tcr}-q_D(\epsilon_{kdf}+\frac{1}{2^k}+\frac{3}{p-3})\right).\end{aligned}$$

Then, we also have $(6\epsilon_{ih^2dh}+\epsilon_{tcr}+q_D(\epsilon_{kdf}+\frac{1}{2^k}+\frac{3}{p-3}))(\frac{1}{k^c}-4\epsilon_{ih^2dh})^{-1}\geq\epsilon$, which proves the lemma. □

5 Long Message Encryption Scheme from IHDH

In this section, we propose a new KEM which is CCA-secure under the IHDH assumption. By using this KEM along with any redundancy free DEM, we can construct a hybrid encryption scheme whose ciphertext length is the same as that of the KD hybrid encryption scheme, and this is a PKE scheme (with arbitrary plaintext length) that achieves such ciphertext length from a weaker assumption than the HDH assumption.

The Construction. Let $\mathbb{G}$ be a multiplicative group with prime order p, and $g\in\mathbb{G}$ be a generator. Let $h:\mathbb{G}\rightarrow\{0,1\}^k$ be a KDF. (Based on the leftover hash lemma, we can also use a universal hash function as h.) Let $H:\mathbb{G}\rightarrow\{0,1\}$ be the hardcore function for the IHDH problem on $\mathbb{G}$ and h.

Setup(1^k)**:** Choose a KDF $h:\mathbb{G}\rightarrow\{0,1\}^k$ and a TCRHF $\mathsf{TCR}:\mathbb{G}\rightarrow\mathbb{Z}_p^*$. Generate a random polynomial $f(x)=a_0+a_1x+\cdots+a_{k+1}x^{k+1}$ over $\mathbb{Z}_p$, and compute $y_i=g^{a_i}$ for $0\leq i\leq k+1$. The decryption key is $f(x)$, and the public key is $PK=(\mathbb{G},g,y_0,y_1,...,y_{k+1},\mathsf{TCR},H,h)$.

Encrypt(PK)**:** Pick a random $r\overset{R}{\leftarrow}\mathbb{Z}_p$, and compute $s=\mathsf{TCR}(g^r)$. Let $\psi=(g^r,g^{r\cdot f(s)},h(y_1^r)\oplus\cdots\oplus h(y_k^r))$, $K=(H(y_1^r)||...||H(y_k^r))$. (One can easily compute $g^{f(x)}$ as $g^{f(x)}=\prod_{0\leq i\leq k+1}y_i^{x^i}$. Also, notice that y_0 and y_{k+1} are used for only computing $g^{r\cdot f(s)}$.) The final output is (ψ,K).

Decrypt(dk,ψ,PK)**:** For a ciphertext $\psi=(C_0,C_1,C_2)$, if $C_1=C_0^{f(s)}$ for $s=\mathsf{TCR}(C_0)$ and $C_2=h(C_0^{a_1})\oplus\cdots\oplus h(C_0^{a_k})$, then output $K=(H(C_0^{a_1})||...||H(C_0^{a_k}))$. Otherwise output $\perp$.

The security of the above scheme is addressed as follows:

Theorem 3. *Let $\mathbb{G}$ be a multiplicative group with prime order p, TCR be a TCRHF, h be a KDF, and H be a hardcore function for the IHDH problem on $\mathbb{G}$ and h. Then, the above KEM is CCA-secure under the IHDH assumption on $\mathbb{G}$ and h.*

The hardcore function H does not need to be the GL function, which is required in the previous section. Theorem 3 is straightforwardly proven by the following lemma.

Lemma 3. *Let $\mathbb{G}$ be a multiplicative group with prime order p, TCR be a (τ,ϵ_{tcr})-TCRHF, and h be a (τ,ϵ_{kdf})-KDF. Then, the above scheme is $(\tau-o(\tau),\epsilon_{cca},q_D)$-CCA-secure under the (τ,ϵ_{ih^2dh})-IH^{2}DH assumption on $\mathbb{G}$, H and h, where $\epsilon_{cca}=k\cdot(\epsilon_{ih^2dh}+\epsilon_{tcr}+q_D(\epsilon_{kdf}+\frac{1}{2^k}+\frac{3}{p-2}))$. (Proof of this lemma is given in Appendix C.)*

Table 1: Ciphertext overhead and security of PKE schemes (with short plaintexts) which are directly derived from existing KEMs, where k is the security parameter, and we assume that size of a group element is $2k$. ROM, GDH, CBDH, and DBDH denote the random oracle model [6], the gap DH assumption [33], the computational BDH assumption [9, 10], and the decisional BDH assumption [16], respectively.

KEM	Ctxt Overhead	Assumption	$\Rightarrow$	Ctxt Overhead	Assumption
[1]	$2k$	GDH (ROM)	$\Rightarrow$	$2k - c\log k$	GDH (ROM)
[18]	$2k$	CDH (ROM)		$2k - c\log k$	CDH (ROM)
[14]	$4k$	DBDH		$4k - c\log k$	CBDH
[29]	$4k$	GHDH		$4k - c\log k$	GDH
[18, 23]	$6k$	CDH		$6k - c\log k$	CDH

6 Generic Techniques

In this section, we explain two generic techniques for constructing efficient PKE schemes which are extracted from the above proposed schemes.

6.1 Direct Use of KEM as PKE and its Subtle Point

The first technique is to generically convert any KEM into a PKE scheme *without using a DEM*. This method is as follows: Let M be a plaintext which the sender wants to confidentially send to the receiver. Then, the sender runs the encryption algorithm of a KEM to obtain a ciphertext ψ (of the KEM) and a data encryption key K. If $K = M$, then the sender transmits only $C = \psi$ as the whole ciphertext of M. The receiver recovers $M(= K)$ by using the decryption algorithm of the KEM.

Intuitively, the above method always seems to work for converting any CCA-secure KEM with $|K| = O(\mathsf{poly}(k))$ into a CCA-secure PKE scheme with $|M| \leq |K|$. However, its security proof is not straightforward since the precise distribution of K is not generally known, and furthermore it is not necessary to be uniform. Therefore, it is not easy to *individually* handle (1) the probability of succeeding in the simulation and (2) the conditional advantage of the adversary under the condition that the simulation is succeeded. Thus, one of our contributions is a method to *simultaneously* cope with them.[2]

Our proof technique can also be extended to other schemes, and we can construct new PKE schemes with shorter ciphertext length from various KEMs. Examples are summarized in Table 1. Interestingly, it is observed that *ciphertext overhead of a CCA-secure PKE scheme can be less than one group element in the random oracle model under the CDH assumption, and furthermore, it can be less than two group elements in the standard model under the Gap DH (GDH) or computational BDH (CBDH) assumptions.* PKE schemes with such short ciphertext length have not been known before.[3]

It should be also noticed that for short plaintexts (such that $|M| < k$), redundancy free DEMs are not available in the standard KEM/DEM framework. Therefore, our scheme is advantageous to the standard KEM/DEM framework in terms of ciphertext length when plaintexts are very short.

Remark. For any PKE scheme, it is also possible to save $c\log k$ bits in ciphertext length if we repeatedly run the encryption algorithm until the first $c\log k$ bits in the generated ciphertext become all zero (and do not transmit the first $c\log k$ bits of the ciphertext). However, since in this modified scheme, the ciphertext space becomes significantly restricted, the security proof of the original PKE scheme cannot be immediately applied.

[2] It is not difficult to show that the distribution of K is *computationally* indistinguishable from the uniform distribution. However, this is not sufficient for simultaneously handling (1) and (2).

[3] Due to the use of the GL function, underlying assumptions are also weakened to be computational ones.

6.2 A Practical Conversion from Any CCCA-Secure KEM into CCA-Secure One

Here, we explain the other generic technique which is a computationally cost-free method to convert any CCCA-secure KEM into a CCA-secure KEM. It should be noticed that our proposed CCA-secure PKE scheme is constructed from the CCCA-secure KEMs due to [25, 23] without using authenticated encryption. Here, we generalize this technique. More specifically, we give a generic conversion which provides a CCA-secure KEM with essentially the same efficiency as its underlying CCCA-secure KEM. For simplicity, we assume that length of a data encryption key of the underlying CCCA-secure KEM and the resulting CCA-secure KEM are $2k$ bits and k bits, respectively, where k is the security parameter.[4]

The proposed conversion is as follows: For a given (CCCA-secure) KEM $\Pi' = (\mathbf{Setup}', \mathbf{Encrypt}', \mathbf{Decrypt}')$, we construct another (CCA-secure) KEM $\Pi = (\mathbf{Setup}, \mathbf{Encrypt}, \mathbf{Decrypt})$ where $\mathbf{Setup}(1^k)$: Run $\mathbf{Setup}'(1^k)$ to obtain (dk, PK). The decryption key is dk and the public key is PK. $\mathbf{Encrypt}(PK)$: Run $(\psi, K) \leftarrow \mathbf{Encrypt}'(PK)$, and split K into K_a and K_s such that $K = (K_a \| K_s)$ and $|K_a| = |K_s| = k$. The final output is $((\psi, K_a), K_s)$ where (ψ, K_a) is the ciphertext and K_s is the data encryption key. $\mathbf{Decrypt}(dk, (\psi, K_a), PK)$: For a ciphertext (ψ, K_a), compute $K' \leftarrow \mathbf{Decrypt}'(dk, \psi, PK)$, and split K' into K'_a and K'_s as above. Then, output K'_s if $K_a = K'_a$, or "$\perp$" otherwise.

Theorem 4. Π *is* $(\tau, 2\epsilon, q_D)$*-CCA-secure if* Π' *is* $(\tau, \epsilon, q_D, \mu)$*-CCCA-secure with* $\frac{1}{2^k} \leq \mu \leq 1$*. (Proof of this theorem is given in Appendix D.)*

Baek, Galindo, Susilo, and Zhou [4] proposed another simple construction of CCA-secure KEMs from CCCA-secure KEMs. However, our method is more efficient than their construction since the method in [4] requires one additional MAC computation while ours does not need any additional computation. It should be noticed that [4] also requires that two k-bit keys can be extracted from one KEM, and therefore, cipheretext sizes of both their method and ours are same. An instantiation of our conversion based on KD scheme is given in Appendix E.

[4] With the help of appropriately chosen KDFs, it is also possible to flexibly determine these values as various length. However, this assumption significantly simplifies the description of the construction, and furthermore it holds in many interesting instantiations.

References

[1] M. Abdalla, M. Bellare, and P. Rogaway, “The oracle Diffie-Hellman assumptions and an analysis of DHIES,” Proc. of CT-RSA’01, LNCS 2020, pp.143-158, 2001.

[2] M. Abe, R. Gennaro, and K. Kurosawa, “Tag-KEM/DEM: A new framework for hybrid encryption” J. of Cryptology, 21(1), pp.97-130, 2008.

[3] M. Abe, E. Kiltz, and T. Okamoto, “Compact CCA-Secure encryption for messages of arbitrary length,” Proc. of PKC’09, LNCS 5443, pp.377-392, 2009.

[4] J. Baek, D. Galindo, W. Susilo, and J. Zhou, “Constructing strong KEM from weak KEM (or how to revive the KEM/DEM framework),” Proc. of SCN’08, LNCS 5229, pp.358-374, 2008.

[5] M. Bellare and C. Namprempre, “Authenticated encryption: relations among notions and analysis of the generic composition paradigm,” Proc. of Asiacrypt’00, LNCS 1976, pp.531-545, 2000.

[6] M. Bellare and P. Rogaway, “Random oracles are practical: a paradigm for designing efficient protocols,” Proc. of CCS’93, pp.62-73, 1993.

[7] D. Boneh and X. Boyen, “Efficient selective-ID secure identity-based encryption without random oracles,” Proc. of Eurocrypt’04, LNCS 3027, pp.223-238, 2004.

[8] D. Boneh, X. Boyen, and H. Shacham, “Short group signatures,” Proc. of Crypto’04, LNCS 3152, pp.41-55, 2004.

[9] D. Boneh and M.K. Franklin, “Identity-based encryption from the Weil pairing,” Proc. of Crypto’01, LNCS 2139, pp.213-229, 2001.

[10] D. Boneh and M. Franklin, “Identity-based encryption from the Weil pairing,” SIAM J. Comput. 32(3), pp.585-615, 2003.

[11] D. Boneh and I. Shparlinski, “On the unpredictability of bits of the elliptic curve Diffie-Hellman scheme,” Proc. of Crypto’01, LNCS 2139, pp.201-212, 2001.

[12] D. Boneh and R. Venkatesan, “Hardness of computing the most significant bits of secret keys in Diffie-Hellman and related schemes,” Proc. of Crypto’96, LNCS 1109, pp.129-142, 1996.

[13] X. Boyen, “Miniature CCA2 PK encryption,” Proc. of Asiacrypt’07, LNCS 4833, pp.485-501, 2007.

[14] X. Boyen, Q. Mei, and B. Waters, “Direct chosen ciphertext security from identity-based techniques,” Proc. of CCS’05, pp.320-329, 2005.

[15] R. Canetti, O. Goldreich, and S. Halevi, “The random oracle methodology, revisited,” Proc. of STOC’98, pp.209-218, 1998.

[16] R. Canetti, S. Halevi, and J. Katz, “A forward-secure public-key encryption scheme,” Proc. of Eurocrypt’03, LNCS 2656, pp.255-271, 2003.

[17] R. Canetti, S. Halevi, and J. Katz, “Chosen-ciphertext security from identity-based encryption,” Proc. of Eurocrypt’04, LNCS 3027, pp.207-222, 2004.

[18] D. Cash, E. Kiltz, and V. Shoup, “The twin Diffie-Hellman problem and applications,” Proc. of Eurocrypt’08, LNCS 4965, pp.127-145, 2008.

[19] R. Cramer and V. Shoup, “A practical public key cryptosystem provably secure against adaptive chosen ciphertext attack,” Proc. of Crypto’98, LNCS 1462, pp.13-25, 1998.

[20] R. Cramer and V. Shoup, “Design and analysis of practical public-key encryption schemes secure against adaptive chosen ciphertext attack,” SIAM J. of Comput., 33(1), pp.167-226, 2003.

[21] D. Dolev, C. Dwork, and M. Naor, "Non-malleable cryptography," Proc. of STOC'91, pp. 542-552, 1991.

[22] O. Goldreich and L.A. Levin, "A hard-core predicate for all one-way functions," Proc. of STOC'89, pp.25-32, 1989.

[23] G. Hanaoka and K. Kurosawa, "Efficient chosen ciphertext secure public key encryption under the computational Diffie-Hellman assumption", Proc. of Asiacrypt'08, LNCS 5350, pp. 308-325, 2008.

[24] J. Herranz, D. Hofheinz, and E. Kiltz, "The Kurosawa-Desmedt key encapsulation is not chosen-ciphertext secure," Cryptology ePrint Archive, 2006/207, 2006.

[25] D. Hofheinz and E. Kiltz, "Secure hybrid encryption from weakened key encapsulation," Proc. of Crypto'07, LNCS 4622, pp.553-571, 2007.

[26] D. Hofheinz and E. Kiltz, "Practical chosen ciphertext secure encryption from factoring," Proc. of Eurocrypt'09, LNCS 5479, pp.313-332, 2009.

[27] D. Hofheinz and E. Kiltz, "The group of signed quadratic residues and applications," Proc. of Crypt'09, LNCS 5677, pp.637-653, 2009.

[28] E. Kiltz, "A primitive for proving the security of every bit and about universal hash functions & hard core bits," Proc. of FCT'01, LNCS 2138, pp.388-391, 2001.

[29] E. Kiltz, "Chosen-ciphertext secure key-encapsulation based on gap hashed Diffie-Hellman," Proc. of PKC'07, LNCS 4450, pp.282-297, 2007. Full version is available at `http://eprint.iacr.org/2007/036`.

[30] K. Kurosawa and Y. Desmedt, "A new paradigm of hybrid encryption scheme," Proc. of Crypto'04, LNCS 3152, pp.426-442, 2004.

[31] K. Kurosawa and T. Matsuo, "How to remove MAC from DHIES," Proc. of ACISP'04, LNCS 3108, pp.236-247, 2004.

[32] M. Luby and C. Rackoff, "How to construct pseudorandom permutations from pseudorandom functions," SIAM J. Comput., 17(2), pp.373-386, 1988.

[33] T. Okamoto and D. Pointcheval, "The gap-problems: a new class of problems for the security of cryptographic schemes," Proc. of PKC'01, LNCS 1992, pp.104-118, 2001.

[34] D.H. Phan and D. Pointcheval, "About the security of ciphers (semantic security and pseudo-random permutations)," Proc. of SAC'04, LNCS 3357, pp.182-197, 2004.

[35] N. Pippenger, "On the evaluation of powers and related problems," Proc. of FOCS'76, pp.258-263, 1976.

[36] C. Rackoff and D.R. Simon, "Non-interactive zero-knowledge proof of knowledge and chosen ciphertext attack," Proc. of Crypto'91, LNCS 576, pp.433-444, 1991.

[37] H. Shacham, "A Cramer-Shoup encryption scheme from the linear assumption and from progressively weaker linear variants," Cryptology ePrint Archive, 2007/074, 2007.

[38] V. Shoup, "Using hash functions as a hedge against chosen ciphertext attack," Eurocrypt'00, LNCS 1807, pp.275-288, 2000.

[39] B. Waters, "Efficient identity based encryption without random oracles," Proc. of Eurocrypt'05, LNCS 3494, pp.114-127, 2005.

A Proof of Lemma 1

For proving the lemma, it is sufficient to show that for any $2k$-dimensional vector $R = (r_1, ..., r_{2k}) \in \{0,1\}^{2k}$, $\Pr_x[\langle x, R\rangle = 0] \geq 1/2$, where $\langle x, R\rangle$ denotes the innerproduct of the binary representation of $x \in \mathbb{Z}_q$ and R. We note that vector R is not required to be randomly picked, and even for biased ones, e.g. all zero or all one vectors, it is possible to prove that this always holds.

Then, the proof completes by proving the following claims: (i) Lemma 1 is true if for all $i \in \{1, 2, ..., 2k\}$, $\Pr_x[x_i = 0] \geq 1/2$ where x_i is i-th bit of the binary representation of x (x_1 is the most significant bit of x), and (ii) For all $i \in \{1, 2, ..., 2k\}$, $\Pr_x[x_i = 0] \geq 1/2$ holds. These are formally addressed as Claims 6 and 7.

Claim 6. *For any $2k$-dimensional vector $R = (r_1, ..., r_{2k}) \in \{0,1\}^{2k}$, $\Pr_x[\langle x, R\rangle = 0] \geq 1/2$ if $\Pr_x[x_i = 0] \geq 1/2$ always holds for all $i \in \{1, 2, ..., 2k\}$.*

Proof. Let p_i be the probability that the innerproduct of vectors $(x_1, ..., x_i)$ and $(r_1, ..., r_i)$ is zero, i.e. $p_i = \Pr_x[\oplus_{j=1,...,i} x_j r_j = 0]$, and let q_i be the probability that x_i is zero, i.e. $q_i = \Pr_x[x_i = 0]$. Then, we have the following equation:

$$p_{i+1} = \begin{cases} p_i & \text{if } r_{i+1} = 0, \\ p_i q_{i+1} + (1 - p_i)(1 - q_{i+1}) & \text{if } r_{i+1} = 1. \end{cases}$$

Therefore, by a straightforward modification of the above equation, we have

$$p_{i+1} - \frac{1}{2} = \begin{cases} p_i - \frac{1}{2} & \text{if } r_{i+1} = 0, \\ (p_i - \frac{1}{2})(2q_{i+1} - 1) & \text{if } r_{i+1} = 1. \end{cases}$$

It should be noticed that due to the assumption, $q_i = \Pr_x[x_i = 0] \geq 1/2$ for all $i \in \{1, 2, ..., 2k\}$, and consequently, if $(p_i - 1/2) \geq 0$, then $(p_{i+1} - 1/2) \geq 0$ also holds. Since $p_1 \geq q_1 = \Pr_x[x_1 = 0] = 2^{2k-1}/q \geq 1/2$, by the mathematical induction $(p_i - 1/2) \geq 0$ always holds. Hence, we have $(p_{2k} - 1/2) \geq 0$, and therefore $\Pr_x[\langle x, R\rangle = 0] - 1/2 \geq 0$. □

Claim 7. *For all $i \in \{1, 2, ..., 2k\}$, $\Pr_x[x_i = 0] \geq 1/2$.*

Proof. Let q_i be $\Pr_x[x_i = 0]$ for $i \in \{1, 2, ..., 2k\}$ (as above), and let π_i be the i-th bit of the binary representation of q (π_1 is the most significant bit of q). Let E_i be an event that $[(x_1, ..., x_i) = (\pi_1, ..., \pi_i)]$ is true. Then, we have

$$\begin{aligned} q_i &= \Pr_x[x_i = 0 | \mathsf{E}_{i-1}, \pi_i = 1] \Pr[\mathsf{E}_{i-1}, \pi_i = 1] + \Pr_x[x_i = 0 | \mathsf{E}_{i-1}, \pi_i = 0] \Pr[\mathsf{E}_{i-1}, \pi_i = 0] \\ &\quad + \Pr_x[x_i = 0 | \overline{\mathsf{E}_{i-1}}] \Pr[\overline{\mathsf{E}_{i-1}}] \\ &\geq \Pr_x[x_i = 0 | \mathsf{E}_{i-1}, \pi_i = 1] \Pr[\mathsf{E}_{i-1}, \pi_i = 1] + \Pr[\mathsf{E}_{i-1}, \pi_i = 0] + \frac{1}{2} \Pr[\overline{\mathsf{E}_{i-1}}]. \end{aligned}$$

Therefore, for proving the claim, it is sufficient to show that $\Pr_x[x_i = 0 | \mathsf{E}_{i-1}, \pi_i = 1] \geq 1/2$. Let q'_i be $\Pr_x[x_i = 0 | \mathsf{E}_{i-1}, \pi_i = 1]$. Then, q'_i is estimated as

$$q'_i = \frac{2^{2k-i}}{p - \sum_{j \in \{1,...,i-1\}} \pi_j \cdot 2^{2k-j}} \geq \frac{2^{2k-i}}{2^{2k-i+1}} = \frac{1}{2},$$

which proves the claim. □

From Claims 6 and 7, the lemma is proved.

B Proof of Claim 3

Since rnd is information-theoretically hidden to A, for a query $C = (C_0, C_1, C_2)$ $\mathsf{TCR}(C_0) = rnd$ happens with probability at most $1/(p-2)$. On the other hand, the probability of succeeding in generating a ciphertext $C = (C_0, C_1, C_2)$ such that $C_0 \neq C_0^\star$ and $\mathsf{TCR}(C_0) = \mathsf{TCR}(C_0^\star)$ is bounded by ϵ_{tcr}. Therefore, $\mathsf{Abort} = $ true happens with probability at most $\epsilon_{tcr} + q_D/q - 2$.

C Proof of Lemma 3

Assume we are given an adversary A which breaks CCA-security of the above KEM with running time τ, advantage ϵ, and q_D decryption queries. We use A to construct another adversary B which solves the IH^{2}DH problem on $\mathbb{G}$, H and h.

Assume that there exists an adversary A' which can break the CCA-security of the above KEM with running time τ, advantage ϵ, and q_D decryption queries. That is, A' is given a challenge ciphertext

$$(g^\beta, g^{\beta \cdot f(s^\star)}, h(g^{a_1\beta}) \oplus \cdots \oplus h(g^{a_k\beta})),$$

where $s^\star = \mathsf{TCR}(g^\beta)$, and then A' can distinguish $(H(y_1^\beta)||...||H(y_k^\beta))$ from a random k-bit string. By applying a standard hybrid argument, we can construct another adversary A which distinguishes

$$(H(y_1^\beta)||...||H(y_j^\beta)||random_{k-j})$$

from

$$(H(y_1^\beta)||...||H(y_{j-1}^\beta)||random_{k-j+1})$$

for some j with $1 \leq j \leq k$, where $random_\ell$ denotes an ℓ-bit random string, with running time τ, advantage $\frac{1}{k}\epsilon$, and q_D decryption queries.

We use A to construct a distinguisher B for our IH^{2}DH problem. That is, for given $(g, g^\alpha, g^\beta, h(g^{\alpha\beta}))$, h and γ, B decides if $\gamma = H(g^{\alpha\beta})$ or a random bit as follows.

1. Let $a_j = \alpha$. B chooses $a_1, \cdots, a_{j-1}, a_{j+1}, \cdots, a_k$ randomly from $\mathbb{Z}_p$. Let $y_i = g^{a_i}$ for $i = 1, \cdots, k$.
2. B picks a target collision resistant hash function TCR, and computes $s^\star = \mathsf{TCR}(g^\beta)$.
3. B chooses $t \in \mathbb{Z}_p^* \setminus \{s^\star\}$ and $u^\star, u_t \in \mathbb{Z}_p$ randomly.
4. Define a_0 and a_{k+1} and $f(x)$ in such a way that $f(x) = a_0 + a_1 x + \cdots + a_{k+1}x^{k+1}$ satisfies $f(s^\star) = u^\star$ and $f(t) = u_t$. B computes $y_0 = g^{a_0}$ and $y_{k+1} = g^{a_{k+1}}$ by using g^α.
5. B runs A on input a public key $PK = (\mathbb{G}, g, y_0, y_1, ..., y_{k+1}, \mathsf{TCR}, H, h)$.
6. B computes a challenge ciphertext

   $$\psi^\star = (g^\beta, (g^\beta)^{u^\star}, h((g^\beta)^{a_1}) \oplus \cdots \oplus h(g^{\alpha\beta}) \oplus \cdots \oplus h((g^\beta)^{a_k})$$

   and a challenge key

   $$K^\star = (H((g^\beta)^{a_1})||...||H((g^\beta)^{a_{j-1}})||\gamma||random_{k-j}),$$

   where $\gamma = H(g^{\alpha\beta})$ or a random bit.
7. When A makes a decryption query $\psi = (C_0, C_1, C_2)$, B proceeds as follows:

(a) If $C_0 = g^\beta$, then B responds $\perp$.

(b) If $C_0 \neq g^\beta$ and $\mathsf{TCR}(C_0) \in \{s^\star, t\}$, then B aborts and outputs a random bit.

(c) Otherwise let $\mathsf{TCR}(C_0) = s$. Define $a_{0,t}, a_{j,t}, a_{k+1,t}$ and $f_t(x)$ in such a way that

$$f_t(x) = a_{0,t} + a_1 x + \cdots + a_{j,t}x^j + \cdots a_k x^k + a_{k+1,t}x^{k+1}$$

satisfies

$$f_t(s) = \log_{C_0} C_1,\ f_t(s^\star) = u^\star,\ f_t(t) = u_t.$$

B computes $z_t = C_0^{a_{j,t}}$ by using Lagrange formula. If

$$C_2 = h(C_0^{a_1}) \oplus \cdots \oplus h(z_t) \oplus \cdots \oplus h(C_0^{a_k}), \tag{1}$$

then B returns $(H(C_0^{a_1})||...||H(z_t)||...||H(C_0^{a_k}))$. Otherwise it returns "$\perp$".

8. Finally, A outputs a bit b as his guess, and B outputs the same bit b as his own guess for $h(g^{\alpha\beta})$.

Let Win denote the event that for the challenge ciphertext in the real world, A correctly distinguishes $(H(y_1^\beta)||...||H(y_j^\beta)||random_{k-j})$ from $(H(y_1^\beta)||...||H(y_{j-1}^\beta)||random_{k-j+1})$, Abort denote the event that for the challenge ciphertext $\psi^\star = (C_0^\star, C_1^\star, C_2^\star)$ and a random $t \in \mathbb{Z}_p^* \backslash \{\mathsf{TCR}(C_0^\star)\}$, A submits a ciphertext $\psi = (C_0, C_1, C_2)$ such that $C_0 \neq C_0^\star$ and $\mathsf{TCR}(C_0) \in \{\mathsf{TCR}(C_0^\star), t\}$, and $\mathsf{Invalid}$ denote the event that A submits a ciphertext $\psi = (C_0, C_1, C_2)$ which is rejected in the real world, but not in the above simulation. More precisely, ψ is a ciphertext such that $\mathsf{TCR}(C_0) \notin \{\mathsf{TCR}(C_0^\star), t\}$ and Eq. 1 holds, but $C_1 \neq C_0^{f(s)}$. Then, B's advantage in distinguishing $H(g^{\alpha\beta})$ is estimated as follows:

$$\begin{aligned}
&\frac{1}{2} \cdot |\Pr[\mathsf{B}(g, g^\alpha, g^\beta, h(g^{\alpha\beta}), H(g^{\alpha\beta})) = 0] - \Pr[\mathsf{B}(g, g^\alpha, g^\beta, h(g^{\alpha\beta}), T) = 0]| \\
&\qquad\qquad\qquad \geq\ |\Pr[\mathsf{Win} \wedge \overline{\mathsf{Abort}} \wedge \overline{\mathsf{Invalid}}] - \frac{1}{2}| \\
&\qquad\qquad\qquad \geq\ |\Pr[\mathsf{Win}] - \Pr[\mathsf{Abort}] - \Pr[\mathsf{Invalid}] - \frac{1}{2}]|.
\end{aligned}$$

The proof is completed by estimating bounds on $\Pr[\mathsf{Abort}]$ and $\Pr[\mathsf{Invalid}]$.

Claim 8. $\Pr[\mathsf{Abort}] \leq \epsilon_{tcr} + \frac{q_D}{p-2}$.

Proof. Since t is information-theoretically hidden to A, for a query $\psi = (C_0, C_1, C_2)$ $\mathsf{TCR}(C_0) = t$ happens with probability at most $1/(p-2)$. On the other hand, the probability of succeeding in generating a ciphertext $\psi = (C_0, C_1, C_2)$ such that $C_0 \neq C_0^\star$ and $\mathsf{TCR}(C_0) = \mathsf{TCR}(C_0^\star)$ is bounded by ϵ_{tcr}. Therefore, $\mathsf{Abort} =$ true happens with probability at most $\epsilon_{tcr} + q_D/q - 2$. □

Before estimating $\Pr[\mathsf{Invalid}]$, we address a useful claim for it.

Claim 9. *Suppose that $(C_0, C_1) \neq (g^r, g^{rf(s)})$ at step 7. For $t_1, t_2 \in \mathbb{Z}_p^* \setminus \{s^\star\}$, let z_{t_1} and z_{t_2} be z_t with $t = t_1$ and t_2, respectively. If $t_1 \neq t_2$, then $z_{t_1} \neq z_{t_2}$.*

Proof. Suppose that $z_{t_1} = z_{t_2}$. Then we have

$$\begin{aligned}
\Delta(x) &= f_{t_1}(x) - f_{t_2}(x) \\
&= (a_{0,t_1} - a_{0,t_2}) + (a_{k+1,t_1} - a_{k+1,t_2})x^{k+1}.
\end{aligned}$$

Further it is clear that

$$\Delta(s) = \Delta(s^\star) = 0.$$

Hence we obtain that $f_{t_1}(x) = f_{t_2}(x)$. In this case, let

$$\begin{array}{rcl} \Delta'(x) & = & f_{t_1}(x) - f(x) \\ & = & (a_{0,t_1} - a_0) + (a_{j,t_1} - \alpha)x^j + (a_{k+1,t_1} - a_{k+1})x^{k+1}. \end{array}$$

Then it is easy to see that

$$\Delta'(s^\star) = \Delta'(t_1) = \Delta'(t_2) = 0.$$

Therefore we obtain that $f_{t_1}(x) = f(x)$. However this is a contradiction because $f_{t_1}(s) \neq f(s)$. □

Now, we are ready to estimate an upper bound on $\Pr[\mathsf{Invalid}]$.

Claim 10. $\Pr[\mathsf{Invalid}] \leq q_D \cdot (\epsilon_{kdf} + \frac{1}{2^k} + \frac{2}{p-2})$.

Proof. It is obvious that Eq. 1 holds with probability at most $1/2^k$ if no information on $h(z_t)$ is given. From Claim 9, we have that if $C_1 \neq C_0^{f(s)}$, then z_t takes $p-2$ different values according to $p-2$ different values for t. Since h is a KDF, the distribution of $h(z_t)$ is computationally indistinguishable from the uniform distribution with advantage at most $\epsilon_{kdf} + 2/p$, where $2/p$ comes from statistical distance between the uniform distributions over $\mathbb{Z}_p$ and $\mathbb{Z}_p^*\backslash\{t\}$. Hence, for an invalid query Eq. 1 holds with probability at most $1/2^k + \epsilon_{kdf} + 2/p$. □

From Claims 8 and 10, we have

$$\epsilon_{ih^2dh} \geq \frac{1}{k}\epsilon - \epsilon_{tcr} - q_D\left(\epsilon_{kdf} + \frac{1}{2^k} + \frac{3}{p-2}\right),$$

which proves the lemma.

D Proof of Theorem 4

Assume we are given an adversary A which breaks CCA-security of Π with advantage 2ϵ. We use A to construct another adversary B which breaks CCCA-security of Π' with advantage ϵ and uncertainty $\mu(\geq 1/2^k)$. Specifically, when A submits a decryption query (ψ, K_a) to B, B also submits $(\psi, \mathit{pred}(\cdot))$ to the CCCA challenger where pred is defined as $\mathit{pred}(K) = 1$ iff *the first k bits of K is identical to K_a*. Obviously, this does not contradict to the restriction that $\mu \geq 1/2^k$. Therefore, B can perfectly respond to A's queries by using the challenger's answer to A. For the CCCA challenge ciphertext $(\psi^\star, \hat{K})$ for B, B inputs $((\psi^\star, \hat{K}_a), \hat{K}_{s,b})$ to A where b is a random bit, $\hat{K} = (\hat{K}_a \| \hat{K}_{s,0})$, and $\hat{K}_{s,1}$ is a random string with $|\hat{K}_{s,1}| = k$. It is clear that this is a valid CCA challenge ciphertext if $\hat{K}$ is the real key for $\psi^\star$, and therefore, in this case A will output b' such that $b' = b$ with advantage 2ϵ. On the other hand, if $\hat{K}$ is random, then A may notice that it is a simulation, but it can not do anything except for randomly guessing b. Hence, if Π' is CCCA-secure, then Π is CCA-secure.

E Instantiation of Our Conversion

Here, we demonstrate to construct a new CCA-secure KEM which is derived from the (CCCA-secure) KD KEM via our conversion, and address its efficiency. The construction of the original KD KEM is as follows: Let $\mathbb{G}$ be a multiplicative group with prime order p, and $g, h \in \mathbb{G}$ be generators. The decryption key is $dk = (x_0, x_1, y_0, y_1) \stackrel{R}{\leftarrow} \mathbb{Z}_p^4$, and the public key is $PK = (\mathbb{G}, g, h, X, Y, \mathsf{TCR})$ where $X = g^{x_0}h^{x_1}$, $Y = g^{y_0}h^{y_1}$, and TCR is a target collision resistant hash function. A ciphertext and its corresponding data encryption key are generated as $\psi = (g^r, h^r)$ and $K = (XY^\alpha)^r$, respectively, where $r \overset{R}{\leftarrow} \mathbb{Z}_p$ and $\alpha = \mathsf{TCR}(g^r, h^r)$. Decryption can be straightforwardly done by using dk. It is known that the KD KEM is not CCA-secure [24], but CCCA-secure [25]. By applying our conversion, the KD KEM is transformed to be a CCA-secure KEM which is as follows: The decryption key and the publc key are the same as the original KD KEM. A ciphertext and its corresponding data encryption key is generated as $\psi = (g^r, h^r, K_a)$ and $K = K_s$, respectively, where $(K_a \| K_s) = (XY^\alpha)^r$. We note that due to the existence of the Pollard rho algorithm, $|p|$ is determined to be at least $2k$ bits for k-bit security, and therefore, we have $|(XY^\alpha)^r| = 2k$ and $|K_a| = |K_s| = k$.[5] Decryption is straightforward (it is the same as that of the KD KEM with consistency check of K_a). Obviously, computational cost of the resulting KEM is completely the same as the original KD KEM. On the other hand, length of a ciphertext becomes k-bit longer than that of the KD KEM. However, due to strengthened security (i.e. CCA-security) it also becomes possible to use a standard CCA-secure DEM (instead of authenticated encryption) for hybrid encryption, and consequently total length of a ciphertext of hybrid encryption from the new KEM is also completely the same as that from the KD KEM.

[5] Rigorously speaking, since the distribution of $(XY^\alpha)^r$ is not uniform over $\{0,1\}^{2k}$ but over $\mathbb{G}$, we need to convert $(XY^\alpha)^r$ into a $2k$-bit binary string in an appropriate manner, and then split it into k-bit strings K_a and K_s.