Efficient Chosen Ciphertext Secure Public Key Encryption under the Computational Diffie-Hellman Assumption

Goichiro Hanaoka* Kaoru Kurosawa†

Abstract

Recently Cash, Kiltz, and Shoup [20] showed a variant of the Cramer-Shoup (CS) public key encryption (PKE) scheme [21] whose chosen-ciphertext (CCA) security relies on the *computational Diffie-Hellman* (CDH) assumption. The cost for this high security is that the size of ciphertexts is much longer than the CS scheme. In this paper, we show how to achieve CCA-security under the CDH assumption without increasing the size of ciphertexts. We further show a more efficient scheme under the *hashed Diffie-Hellman* (HDH) assumption such that the size of ciphertexts is the same as that of the Kurosawa-Desmedt (KD) scheme [42]. Note that the CDH and HDH assumptions are weaker than the decisional Diffie-Hellman assumption which the CS and KD schemes rely on.

Both of our schemes are based on a certain broadcast encryption (BE) scheme while the Cash-Kiltz-Shoup scheme is based on a different paradigm which is called the *Twin DH problem*. As an independent interest, we also show a generic method of constructing CCA-secure PKE schemes from BE schemes such that the existing CCA-secure constructions can be viewed as special cases.

Key words: public key encryption, CCA security, the CDH assumption

1 Introduction

1.1 Background

Chosen-ciphertext security (CCA-security, for short) [52, 27] is considered as a standard notion of security for public key encryption (PKE) in practice. Furthermore, this security also implies universally composable security [16]. So far, many CCA-secure PKE schemes have been proposed for both theoretical ones [48, 27, 53, 43] and practical ones [21, 55, 22, 18, 42, 15, 1, 41, 38], and their security are proven under existence of enhanced trapdoor permutations (for theoretical schemes) or various number theoretic assumptions (for practical schemes). One of the most important research topics in this field is to design CCA-secure PKE schemes with weaker assumptions and/or better efficiency. Especially, there have been no CCA-secure PKE scheme under the *computational Diffie-Hellman* (CDH) assumption nor even the *hashed Diffie-Hellman* (HDH) assumption, except for a

*National Institute of Advanced Industrial Science and Technology (AIST). hanaoka-goichiro@aist.go.jp
†Ibaraki University. kurosawa@mx.ibaraki.ac.jp

recent work by Cash, Kiltz, and Shoup [20] (see Appendix K) which is an independent work to ours.[1]

The main motivation of this work is to construct *practical* CCA-secure PKE schemes under the CDH or HDH assumption (with significantly better efficiency than the schemes in [20]). Note that the CDH and HDH assumptions are weaker than the decisional Diffie-Hellman (DDH) assumption which the Cramer-Shoup (CS) scheme [21] and the Kurosawa-Desmedt (KD) scheme [42] rely on.

1.2 Our Contribution

In this paper, we present a practical CCA-secure PKE scheme under the CDH assumption such that the size of a ciphertext is much smaller than that of the scheme (independently) proposed by Cash, Kiltz, and Shoup [20]. Specifically, ciphertext overhead of our CDH-based scheme is only three group elements for arbitrary plaintext length, while that of the CDH-based Cash-Kiltz-Shoup (CKS) scheme is $k/\log k + 2$ group elements where k is the security parameter. Indeed, the ciphertext length of our scheme is the same as that of the CS scheme.

Under the HDH assumption, we also present another practical CCA-secure PKE scheme which is as efficient as the KD scheme [42] in terms of both computational costs and data sizes, where the KD scheme is based on the DDH assumption only.[2] Surprisingly, our HDH-based scheme is more or equally advantageous to both the KD scheme and the HDH-based CKS scheme in all aspects which are mentioned in Table 1 (see Sec. 7). More specifically, our HDH-based scheme provides the same efficiency as the KD scheme with *stronger security*, and the same security as the HDH-based CKS scheme with *better efficiency*.[3]

We construct our schemes from the Naor-Pinkas (NP) broadcast encryption (BE) scheme based on an observation that the Dolev-Dwork-Naor paradigm [27] can be generalized by using BE schemes with verifiability (see below), while the CKS scheme is based on a different paradigm which is called the *Twin DH* problem. As an independent interest, we show that a CCA-secure key encapsulation mechanism (KEM) can be constructed from any selectively chosen plaintext (CPA) secure *verifiable* BE scheme (see Appendix B.3.3) at a slight cost, where we say that a BE scheme is verifiable if any one of valid receivers can verify whether decryption results of all valid receivers are identical or not. Interestingly, almost all of existing methods for achieving CCA-security, e.g. [27, 21, 18], can be also explained from this viewpoint with different verifiable BE schemes. See Sec. 8.5 for details. Furthermore, it is also possible to construct a new PKE scheme from the Boneh-Gentry-Waters (BGW) BE scheme [11] (see Sec. 9). This is a further evidence which implies that BE with verifiability is a powerful tool for obtaining CCA-secure PKE schemes. Moreover, we can generically convert any CPA-secure BE with verifiability into CCA-secure BE at a slight cost (see Sec. 10).

[1] The authors of [20] also cite an earlier version of this paper as an independent work.

[2] The HDH assumption does not imply the DDH assumption, but the DDH assumption implies the HDH assumption assuming the underlying hash function is a secure *key derivation function* (KDF) [55].

[3] Independently to our work, the authors of [20] discovered that a variant of the KD scheme in [38] can be transformed into another one with the HDH assumption [19] by extending the proof technique in [38] but not the twin DH problem in [20]. This result is presented in the unpublished full(er) version of [20].

1.3 Related Works

Chosen-Ciphertext Security. The notion of CCA-security was introduced by Naor and Yung [48], and this was further extended by Rackoff and Simon [52] and Dolev, Dwork, and Naor [27]. Naor and Yung proposed a generic construction of non-adaptively CCA-secure cryptosystems from any semantically secure encryption [34] and *non-interactive zero knowledge* (NIZK) proof [6]. Dolev, Dwork, and Naor [27] and Sahai [53] later improved this idea and proposed adaptively CCA-secure constructions. However, it is not known if it is possible to generically construct an NIZK proof from any semantically secure encryption. Recently, Gertner, Malkin, and Myers [31] showed that for a large non-trivial class of constructions, it is impossible to construct a CCA-secure scheme from a CPA-secure scheme in a black box manner.

Cramer and Shoup [21] proposed the first practical CCA-secure scheme under the DDH assumption. Cramer and Shoup [22] further applied their methodology to [34] and [49].

Shoup [55] proposed a general KEM/DEM framework, and extended the CS scheme to be a hybrid encryption scheme. Kurosawa and Desmedt [42] improved efficiency of the hybrid version of CS. Abe, Gennaro, Kurosawa, and Shoup [1] established the Tag-KEM/DEM framework, and explained the security of Kurosawa-Desmedt in this framework. Hofheinz and Kiltz [38] presented another paradigm for constructing hybrid encryption with strictly weakened KEM. The DDH assumption is still required for these extensions except for one of Hofheinz and Kiltz's schemes which depends on the *n-linear* DDH assumption.

Canetti, Halevi, and Katz [18] proposed a generic method for converting a selectively secure identity-based encryption (IBE) scheme [54, 10] into a CCA-secure PKE scheme, and Boneh and Katz [12] improved its efficiency. Kiltz [40] discussed a more relaxed condition for achieving CCA-security. Boyen, Mei, and Waters [15] proposed practical CCA-secure schemes by using the basic idea of the Canetti-Halevi-Katz (CHK) paradigm and specific properties of [56] and [7]. Security of these schemes are proven under the *bilinear Diffie-Hellman* (BDH) assumption. Kiltz [41] also proposed another practical CCA-secure scheme whose security is proven under the *gap hashed Diffie-Hellman* (GHDH) assumption. With a slight modification (by using hardcore bits), this scheme can be also provably secure under the *gap Diffie-Hellman* (GDH) assumption which is equivalent to the CDH assumption over specific cyclic groups such as *bilinear groups*.

All of the above-mentioned number theoretic assumptions, i.e. DDH, n-linear DDH, BDH, GHDH, and GDH assumptions, are strictly stronger than the CDH assumption. In other words, the CDH assumption is implied by any of these assumptions. See also Appendix A for understanding the difficulty for constructing CDH-based PKE schemes.

Under the random oracle methodology [4], there exist many practical schemes, e.g. [5, 30]. However, this methodology is known to be problematic [17], and hence, in this paper we do not consider it.

Very recently, as an independent work to ours (see the footnote in Sec. 1.1), Cash, Kiltz, and Shoup [20] also proposed CCA-secure PKE schemes (see Appendix K) under the CDH or HDH assumption by using the *Twin DH problem* (which is also applicable to a wide range of cryptographic primitives). However, our proposed schemes are more efficient than theirs in various aspects.

Broadcast Encryption. Broadcast encryption (BE) is a class of encryption schemes where there exist multiple receivers. For each message transmission, the sender can generate a ciphertext which can be decrypted by only privileged receivers. Fiat and Naor [29] proposed the first non-trivial

construction of BE. Naor, Naor, and Lotspiech [46] presented a significantly more efficient scheme. Dodis and Fazio [24] extended [46] to be a public key scheme. Naor and Pinkas [47] proposed another public key BE scheme by using ElGamal-like constructions, and Dodis and Fazio [25] improved it to be secure against adaptive adversaries as well as chosen ciphertext attacks. However, the efficiency of all these schemes depend on the number of colluders, and are not advantageous to the trivial scheme if full collusion resistance is required. Boneh, Gentry, and Waters [11] proposed the first fully collusion resistant (public key) BE scheme whose ciphertext and user decryption keys are of constant size, and they also showed its CCA-secure version.

1.4 Organization of the Rest of the Paper

In the rest of this paper, we first give definitions for our proposed schemes in Sec. 2 (and Appendix B), give the overview of our strategy for obtaining a CDH-based CCA-secure PKE scheme in Sec. 3, show the basic construction of our CDH-based CCA-secure PKE scheme in Sec. 4, propose our full scheme with the CDH assumption and an efficient scheme with the HDH assumption in Secs. 5 and 6, respectively, give a comparison of our proposed schemes with existing practical CCA-secure PKE schemes in Sec. 7, observe the relationship between BE with verifiability and CCA-secure PKE in Sec. 8, demonstrate to construct another new CCA-secure PKE scheme from the BGW BE scheme in Sec. 9 (and Appendix J), and show a generic construction of CCA-secure BE schemes in Sec. 10 (and Appendix L).

2 Definitions

Throughout this paper, we use definitions which are given in Appendix B. For simplicity, we define PKE schemes as key encapsulation mechanisms (KEM). It is well-known that by combining a CCA-secure KEM and a CCA-secure data encryption mechanism (DEM), a CCA-secure PKE scheme is generically obtained [55], and furthermore, there exist some other flexible methods for hybrid encryption as well [1, 38]. It is also known that a CCA-secure DEM can be generically constructed from any pseudorandom functions without redundancy [44]. Therefore, we concentrate on constructions of CCA-secure KEMs.

3 Our Strategy for CCA-Security from the CDH Assumption

Before going into details, here we give an intuitive explanation of our strategy behind the proposed construction. By carefully looking into the classical Dolev-Dwork-Naor construction, we notice that this method can be generalized by using BE schemes with a special property which we call *verifiability*. Roughly speaking, we say that a BE scheme has verifiability if any valid receiver can check equality of decryption results of all valid receivers. See Sec. 8 for details on this observation.

Hence, BE with the CDH assumption would be an appropriate start point for achieving CCA-security from the CDH assumption. Fortunately, it is known that the NP BE scheme [47] has one-wayness against chosen plaintext attacks under the CDH assumption (without verifiability).

3.1 The Naor-Pinkas Broadcast Encryption Scheme

The NP scheme [47], which is constructed based on [2], is as follows. Let $\mathbb{G}$ be a multiplicative group with prime order p, and $g \in \mathbb{G}$ be a generator. For at most $p-1$ potential users and at most t revoked users, in the setup phase, a polynomial $f(x) = \sum_{0 \le i \le t} a_i x^i$ is generated where $a_i \stackrel{R}{\leftarrow} \mathbb{Z}_p$ for $0 \le i \le t$, and $y_i = g^{a_i}$ is computed for $0 \le i \le t$. The public key is $PK = (\mathbb{G}, g, y_0, ..., y_t)$. The center keeps $f(x)$ as the master key, and for each user $i \in \{1, ..., p-1\}$ he has $d_i = f(i)$ as his decryption key.

Assuming that users $i_1, ..., i_t$ are revoked, the sender generates a ciphertext $\psi = (g^r, (g^{f(i_1)})^r, ..., (g^{f(i_t)})^r)$ and a key $K = y_0^r$ where $r \stackrel{R}{\leftarrow} \mathbb{Z}_p$. Notice that $g^{f(i)}$ can be computed as $\prod_{0 \le j \le t} y_j^{ij}$ for any $i \in \{1, ..., p-1\}$. On receiving $\psi = (C_0, ..., C_t)$, user $i (\notin \{i_1, ..., i_t\})$ computes $C_i = C_0^{d_i}$ and recovers the session key as $K = C_i^{\lambda(i)} \prod_{1 \le j \le t} C_j^{\lambda(i_j)}$ where $\lambda(x)$ is the Lagrange coefficient such that $\lambda(x) = \prod_{i' \in \{i, i_1, ..., i_t\} \setminus \{x\}} i' \cdot (i' - x)^{-1}$ over $\mathbb{Z}_p$.

3.2 Main Difficulty

Since the NP scheme does not have verifiability, we cannot straightforwardly convert it into CCA-secure PKE. *This is the main non-trivial part in our work.* In Appendix H.2, two methods for converting the NP scheme to be BE with verifiability are given, however both methods are not sufficient to obtain CDH-based CCA-secure PKE.

3.3 Our Solution: Triple Decryption with Three Keys

As mentioned above, the main difficulty of this work is to add verifiability to the NP scheme. Here, we give a new approach for it which is as follows: Consider a modification of the NP scheme such that user i is given $(f(i), f(rnd), rnd)$ as his decryption key, where $rnd \stackrel{R}{\leftarrow} \mathbb{Z}_p$. We note that a legitimate user i can decrypt a ciphertext in two different ways according to two different keys, i.e. $f(i)$ and $f(rnd)$. If these decryption results are not identical, then the user can detect that the ciphertext is in an invalid form. Notice that since rnd is random and not known to other users, it is difficult to generate an invalid ciphertext whose decryption results under $f(i)$ and $f(rnd)$ are identical.

Unfortunately, the above idea is faulty. Namely, even if user i is revoked and $f(i)$ does not work for decryption, he still has $f(rnd)$ and can decrypt a ciphertext by using it. Hence, the modified scheme is not BE any more. Therefore, we further modify the NP scheme as follows: For at most t revoked users, in the setup phase, a polynomial $f(x) = \sum_{0 \le i \le 2t+1} a_i x^i$ is generated in the same manner as the original NP scheme except that its degree is changed to be $2t+1$. The public key is $PK = (\mathbb{G}, g, y_0, ..., y_{2t+1})$. We assume that a user i has two unique identities $\mathbf{i}$ and i, where we denote $i = (\mathbf{i}, \mathsf{i}) \in \{1, ..., p-1\}^2$. The center keeps $f(x)$ as the master key, and for user $i = (\mathbf{i}, \mathsf{i}) \in \{1, ..., p-1\}^2$ he publishes $d_i = (f(\mathbf{i}), f(\mathsf{i}), f(rnd), rnd)$ as i's decryption key, where $rnd \stackrel{R}{\leftarrow} \mathbb{Z}_p$. Assuming that users $i_1 (= (\mathbf{i}_1, \mathsf{i}_1)), ..., i_t (= (\mathbf{i}_t, \mathsf{i}_t))$ are revoked, the sender generates $\psi = (g^r, (g^{f(\mathbf{i}_1)})^r, ..., (g^{f(\mathbf{i}_t)})^r, (g^{f(\mathsf{i}_1)})^r, ..., (g^{f(\mathsf{i}_t)})^r)$ and $K = y_0^r$ where $r \stackrel{R}{\leftarrow} \mathbb{Z}_p$.

On receiving $\psi = (C_0, ..., C_{2t})$, a user $i = (\mathbf{i}, \mathsf{i}) (\notin \{i_1, ..., i_t\})$ computes $C_{\mathbf{i}} = C_0^{f(\mathbf{i})}$, $C_{\mathsf{i}} = C_0^{f(\mathsf{i})}$, and $C_{rnd} = C_0^{f(rnd)}$. We notice that ψ can be decrypted by using any two of $C_{\mathbf{i}}$, C_{i}, and C_{rnd} with the Lagrange interpolation (for example, by using $(C_{\mathbf{i}}, C_{\mathsf{i}})$, the session key is recovered

as $K = C_{\mathbf{i}}^{\lambda(\mathbf{i})} C_{\mathsf{i}}^{\lambda(\mathsf{i})} \prod_{1 \le j \le t} (C_j^{\lambda(\mathbf{i}_j)} C_{j+t}^{\lambda(\mathsf{i}_j)})$ where $\lambda(x)$ is the Lagrange coefficient such that $\lambda(x) = \prod_{i' \in \{\mathbf{i}, \mathsf{i}, \mathbf{i}_1, \ldots, \mathbf{i}_t, \mathsf{i}_1, \ldots, \mathsf{i}_t\} \setminus \{x\}} i' \cdot (i' - x)^{-1}$ over $\mathbb{Z}_p$). Then, user i carries out decryption in three different ways according to the three different choices of $(C_{\mathbf{i}}, C_{\mathsf{i}})$, $(C_{\mathbf{i}}, C_{rnd})$, and $(C_{\mathsf{i}}, C_{rnd})$. Then, user i can be convinced of the equality of decryption results for all legitimate subscribers if i's three decryption results are identical. Furthermore, when i is revoked, he cannot decrypt a ciphertext at all even though he still has $f(rnd)$. Now, we obtain a new BE scheme with verifiability from NP BE, and are ready to convert it into a CCA-secure PKE scheme.

4 The Basic Scheme from the CDH Assumption

In this section, we give the basic construction of our CDH-based CCA-secure KEM by using the strategy in Sec. 3. This construction yields only one-bit DEM-key space, and in Sec. 5, we modify it to admit arbitrary DEM-key length without increasing ciphertext overhead. Furthermore, in Sec. 6 we give another modification of the basic scheme which is as efficient as the KD scheme under the HDH assumption. Here, for clarifying the essential part of our basic idea, we mainly discuss our construction of a CDH-based one-way KEM (see Sec. B.1.3) which can be easily converted into a CDH-based CCA-secure KEM with one-bit DEM keys by using a hardcore bit (see, for examples, [33, 14, 13, 39]). Our one-way KEM is also CCA-secure as it is, under the DDH assumption with a better reduction cost than the CS scheme [21].

4.1 The One-Way KEM against CCA under the CDH Assumption

Let $\mathbb{G}$ be a multiplicative group with prime order p, and $g \in \mathbb{G}$ be a generator. Then, the construction of our CDH-based one-way KEM is as follows:

Setup(1^k)**:** Generate a random polynomial $f(x) = a_0 + a_1 x + a_2 x^2 + a_3 x^3$ over $\mathbb{Z}_p$, and compute $y_i = g^{a_i}$ for $0 \le i \le 3$. The decryption key is $f(x)$, and the public key is $PK = (\mathbb{G}, g, y_0, y_1, y_2, y_3, \mathsf{TCR})$, where $\mathsf{TCR} : \mathbb{G} \times \{0,1\} \to \mathbb{Z}_p^*$ is a target collision resistant hash function.

Encrypt(PK)**:** Pick a random $r \xleftarrow{R} \mathbb{Z}_p$, and compute

$$\psi = (g^r, g^{r \cdot f(\mathbf{i})}, g^{r \cdot f(\mathsf{i})}), \qquad K = y_0^r$$

where $\mathbf{i} = \mathsf{TCR}(g^r, 0)$ and $\mathsf{i} = \mathsf{TCR}(g^r, 1)$. The final output is (ψ, K). (Notice that one can easily compute $g^{f(x)}$ as $g^{f(x)} = \prod_{0 \le i \le 3} y_i^{x^i}$.)

Decrypt(dk, ψ, PK)**:** For a ciphertext $\psi = (C_0, C_1, C_2)$, check whether $(C_0^{f(\mathbf{i})}, C_0^{f(\mathsf{i})}) \stackrel{?}{=} (C_1, C_2)$, where $\mathbf{i} = \mathsf{TCR}(C_0, 0)$ and $\mathsf{i} = \mathsf{TCR}(C_0, 1)$. If not, output $\perp$. Otherwise, output $K = C_0^{a_0}$.

Theorem 1. *Let $\mathbb{G}$ be a multiplicative group with prime order p, and TCR be a (τ, ϵ_{tcr}) target collision resistant hash function. Then, the above scheme is $(\tau - o(\tau), \epsilon_{cdh} + 2\epsilon_{tcr} + 3q_D/(p-3), q_D)$ one-way under the (τ, ϵ_{cdh}) CDH assumption on $\mathbb{G}$, and is $(\tau - o(\tau), \epsilon_{ddh} + 2\epsilon_{tcr} + 3q_D/(p-3), q_D)$ CCA-secure under the (τ, ϵ_{ddh}) DDH assumption on $\mathbb{G}$.*

Proof. Here, we mainly give the proof for one-wayness of the proposed KEM under the CDH assumption. The proof for CCA-security under the DDH assumption can be straightforwardly done in the almost same manner.

Assume we are given an adversary A which breaks one-wayness 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 CDH problem. Define adversary B as follows:

1. For a given CDH instance (g, g^α, g^β), B picks a target collision resistant hash function TCR randomly, and computes $\mathbf{i}^\star = \mathsf{TCR}(g^\beta, 0)$ and $\mathsf{i}^\star = \mathsf{TCR}(g^\beta, 1)$.

2. B sets $y_0 = g^\alpha$, and picks a random rnd from $\mathbb{Z}_p^* \backslash \{\mathbf{i}^\star, \mathsf{i}^\star\}$. B also picks randoms $u_{\mathbf{i}^\star}$, $u_{\mathsf{i}^\star}$, and u_{rnd} from $\mathbb{Z}_p$.

3. Let $f(x) = \alpha + \sum_{i=1}^{3} a_i x^i$ be a polynomial over $\mathbb{Z}_p$ such that

$$f(\mathbf{i}^\star) = u_{\mathbf{i}^\star}, \quad f(\mathsf{i}^\star) = u_{\mathsf{i}^\star}, \quad f(rnd) = u_{rnd}.$$

   Note that each a_i can be expressed as a linear combination of $\alpha, u_{\mathbf{i}^\star}, u_{\mathsf{i}^\star}$ and u_{rnd} by using Lagrange formula. B then computes $y_i = g^{a_i}$ for $i = 1, 2, 3$ by using $y_0 = g^\alpha$ (see Appdendix B.5).

4. B inputs public key $PK = (\mathbb{G}, g, y_0, y_1, y_2, y_3, \mathsf{TCR})$ and challenge ciphertext $\psi^\star = (g^\beta, (g^\beta)^{u_{\mathbf{i}^\star}}, (g^\beta)^{u_{\mathsf{i}^\star}})$ to A. We note that this is a correct ciphertext and its corresponding encapsulated key is $K^\star = y_0^\beta = g^{\alpha\beta}$.

5. When A makes 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 $\mathsf{TCR}(C_0, b) = [\mathbf{i}^\star, \mathsf{i}^\star$ or $rnd]$ for $b = 0$ or 1, then B aborts and outputs a random element in $\mathbb{G}$.

   (c) If $C_0 \neq g^\beta$ and $\mathsf{TCR}(C_0, b) \neq [\mathbf{i}^\star, \mathsf{i}^\star$ nor $rnd]$ for both $b = 0$ and 1, B computes $C_0^{u_{\mathbf{i}^\star}}$, $C_0^{u_{\mathsf{i}^\star}}$ and $C_0^{u_{rnd}}$. Let $\mathsf{TCR}(C_0, 0) = \mathbf{i}$ and $\mathsf{TCR}(C_0, 1) = \mathsf{i}$, and f_1, f_2, and f_3 be polynomials over $\mathbb{Z}_p$ with degree three, such that

$$\begin{aligned}
(f_1(\mathbf{i}), f_1(\mathsf{i}), f_1(\mathbf{i}^\star), f_1(\mathsf{i}^\star)) &= (\log_{C_0} C_1, \log_{C_0} C_2, u_{\mathbf{i}^\star}, u_{\mathsf{i}^\star}) \\
(f_2(\mathbf{i}), f_2(\mathsf{i}), f_2(\mathbf{i}^\star), f_2(rnd)) &= (\log_{C_0} C_1, \log_{C_0} C_2, u_{\mathbf{i}^\star}, u_{rnd}) \\
(f_3(\mathbf{i}), f_3(\mathsf{i}), f_3(rnd), f_3(\mathsf{i}^\star)) &= (\log_{C_0} C_1, \log_{C_0} C_2, u_{rnd}, u_{\mathsf{i}^\star}).
\end{aligned}$$

   Then, B calculates ${C_0}^{f_1(0)}$ by using the Lagrange interpolation from $(C_1, C_2, C_0^{u_{\mathbf{i}^\star}}, C_0^{u_{\mathsf{i}^\star}})$ (see Appdendix B.5). Similarly, B computes ${C_0}^{f_2(0)}$ and ${C_0}^{f_3(0)}$ from $(C_1, C_2, C_0^{u_{\mathbf{i}^\star}}, C_0^{u_{rnd}})$ and $(C_1, C_2, C_0^{u_{rnd}}, C_0^{u_{\mathsf{i}^\star}})$, respectively. B responds ${C_0}^{f_1(0)}$ if ${C_0}^{f_1(0)} = {C_0}^{f_2(0)} = {C_0}^{f_3(0)}$, or "$\perp$" otherwise.

6. Finally, A outputs a data encryption key K', and B outputs the same value as the solution of the given CDH instance.

Let Win denote the event that $K' = g^{\alpha\beta}$, Abort denote the event that A submits a ciphertext $\psi = (C_0, C_1, C_2)$ such that $C_0 \neq g^{\beta}$ and $\mathsf{TCR}(C_0, b) = [\mathbf{i}^\star, \mathsf{i}^\star \text{ or } rnd]$ for $b = 0$ or 1, and $\mathsf{Invalid}$ denote the event that A submits a ciphertext $\psi = (C_0, C_1, C_2)$ such that B does not abort, ${C_0}^{f_1(0)} = {C_0}^{f_2(0)} = {C_0}^{f_3(0)}$, but $(C_1, C_2) \neq (C_0^{f(\mathbf{i})}, C_0^{f(\mathsf{i})})$ where $f(x) = \sum_{0 \leq i \leq 3} a_i x^i$.

Then, B's success probability in solving the CDH problem is estimated as follows:

$$\begin{aligned} \Pr[\mathsf{B}(g, g^\alpha, g^\beta) \rightarrow g^{\alpha\beta}] &\geq \Pr[\mathsf{Win} | \overline{\mathsf{Abort}} \wedge \overline{\mathsf{Invalid}}] \Pr[\overline{\mathsf{Abort}} \wedge \overline{\mathsf{Invalid}}] \\ &\geq \Pr[\mathsf{Win}] - \Pr[\mathsf{Abort}] - \Pr[\mathsf{Invalid}]. \end{aligned}$$

The proof completes by proving following lemmas.

Lemma 1. $\Pr[\mathsf{Abort}] \leq 2\epsilon_{tcr} + \frac{2q_D}{p-3}$.

Proof. Assume we are given an adversary A with $\Pr[\mathsf{Abort}] = p_A$. Then, we can construct another adversary B' which finds a collision in TCR as follows. For a randomly given TCR instance (C, b), B' generates decryption key $f(x)$ and public key $PK = (\mathbb{G}, g, y_0, y_1, y_2, y_3, \mathsf{TCR})$, and computes challenge ciphertext $\psi^\star = (C, C^{u_{\mathbf{i}^\star}}, C^{u_{\mathsf{i}^\star}})$, where $u_{\mathbf{i}^\star} = f(\mathsf{TCR}(C, 0))$ and $u_{\mathsf{i}^\star} = f(\mathsf{TCR}(C, 1))$. B' also picks a random rnd from $\mathbb{Z}_p^* \backslash \{\mathbf{i}^\star, \mathsf{i}^\star\}$, and gives PK and $\psi^\star$ to A.

Since rnd is information-theoretically hidden to A, for a query $\psi = (C_0, C_1, C_2)$ $[\mathsf{TCR}(C_0, 0)$ or $\mathsf{TCR}(C_0, 1)] = rnd$ happens with probability at most $2/(p-3)$. Therefore, the probability that A submits a ciphertext $\psi = (C_0, C_1, C_2)$ such that $C_0 \neq C$ and $[\mathsf{TCR}(C_0, 0)$ or $\mathsf{TCR}(C_0, 1)] = [\mathbf{i}^\star$ or $\mathsf{i}^\star]$ is at least $p_A - 2q_D/(p-3)$. Since b is also information-theoretically indistinguishable, $[\mathsf{TCR}(C_0, 0)$ or $\mathsf{TCR}(C_0, 1)] = \mathsf{TCR}(C, b)$ happens with probability at least $1/2(p_A - 2q_D/(p-3))$. Hence, we have $\epsilon_{tcr} \geq 1/2(p_A - 2q_D/(p-3))$. □

Lemma 2. $\Pr[\mathsf{Invalid}] \leq \frac{q_D}{p-3}$.

Proof. Suppose $\psi = (C_0, C_1, C_2)$ is a ciphertext such that B does not abort, ${C_0}^{f_1(0)} = {C_0}^{f_2(0)} = {C_0}^{f_3(0)}$, but $(C_1, C_2) \neq (C_0^{f(\mathbf{i})}, C_0^{f(\mathsf{i})})$. Then, we notice that f_1 and f_2 which are polynomials with degree three have four intersections, and consequently they have to be identical. Similarly, we have that $f_1 = f_2 = f_3$. This implies that for $[\mathsf{Invalid} = \text{true}]$, A has to generate a polynomial $f_1 (\neq f)$ with degree three (without knowing rnd) such that

$$(f_1(\mathbf{i}), f_1(\mathsf{i}), f_1(\mathbf{i}^\star), f_1(\mathsf{i}^\star), f_1(rnd)) = (\log_{C_0} C_1, \log_{C_0} C_2, f(\mathbf{i}^\star), f(\mathsf{i}^\star), f(rnd)).$$

Since f_1 and f have at most three intersections and two of them are $(\mathbf{i}^\star, f(\mathbf{i}^\star))$ and $(\mathsf{i}^\star, f(\mathsf{i}^\star))$, there is only one intersection other than these two points. Further rnd is randomly chosen from $\mathbb{Z}_p^* \backslash \{\mathbf{i}^\star, \mathsf{i}^\star\}$. Therefore for any fixed f_1, $\Pr_{rnd}[f_1(rnd) = f(rnd)] = 1/(p-3)$. This means that $\Pr[\mathsf{Invalid}] \leq q_D/(p-3)$. □

□

4.2 The Construction of the CCA-Secure KEM

Now, we give the concrete construction of our CCA-secure KEM under the CDH or HDH assumption. This is a simple modification from the above one-way KEM, and the difference is that in the modified scheme $h(K)$ for some function h is used as the data encryption key instead of K. If h is a hardcore bit function with $h : \mathbb{G} \to \{0,1\}$, then the resulting scheme is a CCA-secure KEM under the CDH assumption with a *single-bit* key. If h is a hash function with $h : \mathbb{G} \to \{0,1\}^\nu$, it then becomes a CCA-secure KEM under the HDH assumption with ν-bit data encryption keys. Then, the security of the modified scheme is addressed as follows:

Theorem 2. *Let $\mathbb{G}$ be a multiplicative group with prime order p, and TCR be a (τ, ϵ_{tcr}) target collision resistant hash function. Then, the above scheme is $(p_1^{-1}(\tau - o(\tau)), p_2^{-1}(\epsilon_{cdh} + 2\epsilon_{tcr} + 3q_D/(p-3)), q_D)$ CCA-secure under the (τ, ϵ_{cdh}) CDH assumption on $\mathbb{G}$ and h is a (p_1, p_2) hardcore bit function in $\mathbb{G}$, and is $(\tau - o(\tau), \epsilon_{hdh} + 2\epsilon_{tcr} + 3q_D/(p-3), q_D)$ CCA-secure under the (τ, ϵ_{hdh}) HDH assumption on $\mathbb{G}$ and h.*

The proof of the theorem is almost the same as that of Theorem 1 (see also Appendix C). We can also directly construct a CCA-secure PKE scheme (with one-bit plaintexts) under the CDH assumption without using the KEM/DEM framework. The full description of the scheme is given in Appendix C.

4.3 Improved Reduction Cost with the DDH Assumption

As addressed in Theorem 1, our proposed KEM in Sec. 4.1 is already CCA-secure under the DDH assumption. It is remarkable that the reduction cost (with respect to the DDH assumption) of our basic scheme is significantly better (more precisely, by a factor of two) than of conventional schemes with the DDH assumption, i.e. the CS scheme [21] and its variants [55, 42].

5 The Full Scheme from the CDH Assumption

Our basic scheme in Sec. 4 yields only a one-bit DEM key. To obtain a k-bit DEM key, we must use k independent copies of the basic scheme which results in k times larger ciphertext size.

In this section, we show our full scheme which yields a k-bit DEM key without increasing the size of ciphertexts. It is obtained by using a random polynomial $f(x)$ of degree $k+2$. Remeber that $\deg f(x) = 3$ in the basic scheme. Let $\mathbb{G}$ be a multiplicative group with prime order p, and $g \in \mathbb{G}$ be a generator. Then, the construction of the scheme is as follows:

Setup(1^k)**:** Generate a random polynomial $f(x) = a_0 + a_1 x + \cdots + a_{k+2}x^{k+2}$ over $\mathbb{Z}_p$, and compute $y_i = g^{a_i}$ for $0 \le i \le k+2$. The decryption key is $f(x)$, and the public key is $PK = (\mathbb{G}, g, y_0, y_1, ..., y_{k+2}, \mathsf{TCR}, h)$, where $\mathsf{TCR} : \mathbb{G} \times \{0,1\} \to \mathbb{Z}_p^*$ is a target collision resistant hash function, and $h : \mathbb{G} \to \{0,1\}$ is a hardcore bit function for the Diffie-Hellman key in $\mathbb{G}$.[4]

Encrypt(PK)**:** Pick a random $r \stackrel{R}{\leftarrow} \mathbb{Z}_p$, and compute

$$\psi = (g^r, g^{r \cdot f(\mathbf{i})}, g^{r \cdot f(\mathrm{i})}), \qquad K = (h(y_0^r)||h(y_1^r)||...||h(y_{k-1}^r))$$

[4] h is a random string R if it is the Goldreich-Levin bit [33], where the size of R is equal to that of a group element.

where $\mathbf{i} = \mathsf{TCR}(g^r, 0)$ and $\mathsf{i} = \mathsf{TCR}(g^r, 1)$. The final output is (ψ, K). (Notice that one can easily compute $g^{f(x)}$ as $g^{f(x)} = \prod_{0 \le i \le k+2} y_i^{x^i}$.)

Decrypt(dk, ψ, PK)**:** For a ciphertext $\psi = (C_0, C_1, C_2)$, check whether $(C_0^{f(\mathbf{i})}, C_0^{f(\mathsf{i})}) \stackrel{?}{=} (C_1, C_2)$, where $\mathbf{i} = \mathsf{TCR}(C_0, 0)$ and $\mathsf{i} = \mathsf{TCR}(C_0, 1)$. If not, output $\perp$. Otherwise, output $K = (h(C_0^{a_0})||h(C_0^{a_1})||...||h(C_0^{a_{k-1}}))$.

Theorem 3. *Let $\mathbb{G}$ be a multiplicative group with prime order p, TCR be a (τ, ϵ_{tcr}) target collision resistant hash function, and h be a (p_1, p_2) hardcore bit function for the Diffie-Hellman key in $\mathbb{G}$. Then, the above scheme is $(p_1^{-1}(\tau) - o(p_1^{-1}(\tau)), k \cdot p_2^{-1}(\epsilon_{cdh}) + 2\epsilon_{tcr} + q_D(2k/(p-3) + 1/(p-k-2)), q_D)$ CCA-secure under the (τ, ϵ_{cdh}) CDH assumption on $\mathbb{G}$.*

The proof of the above theorem is a natural extension of that of Theorem 1 along with a standard hybrid argument. See Appendix D for the full description of the proof.

6 Efficient CCCA-Secure KEM from the HDH Assumption

In this section, we give another modification of our basic scheme which is CCCA-secure [38] under the HDH assumption. (See Appendix B.1.4 for the definition of CCCA security.) This scheme is comparably efficient to the KD scheme [42] with both a weaker assumption and a better reduction cost. As shown in [38], a CCA-secure PKE scheme can be constructed by combining any CCCA-secure KEM and authenticated symmetric key encryption [3] as a DEM. Let $\mathbb{G}$ be a multiplicative group with prime order p, and $g \in \mathbb{G}$ be a generator. Then, the construction of our CCCA-secure KEM 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 $PK = (\mathbb{G}, g, y_0, y_1, y_2, \mathsf{TCR}, h)$, where $\mathsf{TCR} : \mathbb{G} \to \mathbb{Z}_p^*$ is a target collision resistant hash function and $h : \mathbb{G} \to \{0,1\}^\nu$ is a hash function.

Encrypt(PK)**:** Pick a random $r \stackrel{R}{\leftarrow} \mathbb{Z}_p$, and compute

$$\psi = (g^r, g^{r \cdot f(i)}), \qquad K = h(y_0^r)$$

where $i = \mathsf{TCR}(g^r)$. The final output is (ψ, K). (Notice that one can easily compute $g^{f(x)}$ as $g^{f(x)} = \prod_{0 \le j \le 2} y_j^{x^j}$.)

Decrypt(dk, ψ, PK)**:** For a ciphertext $\psi = (C_0, C_1)$, check whether $C_0^{f(i)} \stackrel{?}{=} C_1$, where $i = \mathsf{TCR}(C_0)$. If not, output $\perp$. Otherwise, output $K = h(C_0^{a_0})$.

Theorem 4. *Let $\mathbb{G}$ be a multiplicative group with prime order p, and TCR be a (τ, ϵ_{tcr}) target collision resistant hash function. Then, the above scheme is $(\tau - o(\tau), \epsilon_{hdh} + \epsilon_{tcr} + q_D(\mu + 4/(p-2)), q_D, \mu)$ CCCA-secure under the (τ, ϵ_{hdh}) HDH assumption on $\mathbb{G}$ and h.*

The proof of the above theorem is given in Appendix E.

Table 1: Efficiency comparison for CCA-secure PKE schemes. Some figures are borrowed from [15, 41]. For efficiency, we count the number of pairings, multi(or sequential)-exponentiations [51], regular-exponentiations, and other group operations ("ops" denotes group operations) used for encryption and decryption. All symmetric operations (such as hash function/MAC/KDF) are ignored. Ciphertext overhead represents the difference between ciphertext and plaintext length, and $|g|$ and $|mac|$ are the length of a group element and an authentication tag, respectively. The key size is measured in two parameters: the size of the system parameters (which are fixed for every public key) plus the size of public key pk, and the size of decryption key dk. Here we only take into account the number of group elements for pk, and the number of elements in $\mathbb{Z}_p$ or $\mathbb{G}_1$ for dk. In the table, we let $k' = k/\log k$ where k is the security parameter, i.e. DEM-key length.

	Security Assumption	Ciphertext Overhead	Encryption	Decryption	Key size (pk/dk)	Redection Cost
			#pairings + #[multi,regular]-exp (+ #ops)			
CS [21]	DDH	$3\|g\|$	$0 + [1, 3]$	$0 + [1, 1]$	$5/5$	$\frac{1}{2}$
KD [42]	DDH	$2\|g\| + \|mac\|$	$0 + [1, 2]$	$0 + [1, 0]$	$4/4$	$\frac{1}{2}$
BMW [15]	BDH	$2\|g\|$	$0 + [1, 2]$	$1 + [0, 1]$	$4/3$	1
	↓	↓	↓	$3 + [0, 1]$	$4/1$	1
Kiltz [41]	GHDH	$2\|g\|$	$0 + [1, 2]$	$0 + [1, 0]$	$3/2$	1
CKS [20]	CDH	$(k' + 2)\|g\|$	$0 + [k' + 1, k' + 1]$	$0 + [1^{\ddagger}, 0]$	$2k' + 3/2k' + 2$	$-^{\dagger}$
	HDH	$3\|g\|$	$0 + [2, 2]$	$0 + [1, 0]$	$5/4$	1
Ours §5	CDH	$3\|g\|$	$0 + [2^{\ddagger}, k' + 1]$	$0 + [1^{\ddagger}, 0]$	$k' + 4/k' + 3$	$-^{\dagger}$
Ours §6	HDH	$2\|g\| + \|mac\|$	$0 + [1, 2]$	$0 + [1, 0]$	$4/3$	1
Ours §9	2ℓ-BDHE	$2\|g\|$	$0 + [0, 3] + \ell$	$3 + [0, 0] + \ell$	$4\ell + 1/1$	1

† This depends on the underlying hardcore bit function. ‡ Relatively more expensive computation is needed for each exponentiation.

7 Comparison

Table 1 shows a comparison of our schemes with other CCA-secure schemes, i.e. Cramer-Shoup (CS) [23], Kurosawa-Desmedt (KD) [42], Boyen-Mei-Waters (BMW) [15], Kiltz [41], and Cash-Kiltz-Shoup (CKS) [20]. In the comparison, we utilize a redundancy-free CCA-secure DEM [36, 37, 35, 50] for constructing a CCA-secure hybrid encryption scheme from a CCA-secure KEM.

As seen in Table 1, our proposed scheme in Sec. 5 yields both provable security under the CDH assumption and short ciphertext length which is comparable to other practical schemes. Comparing with the CDH-based CKS scheme, our scheme in Sec. 5 is superior in both computational costs and data sizes, and especially, the ciphertext overhead of our scheme, i.e. three group elements, is much shorter than that of the CKS scheme, i.e. $k/\log k + 2$ group elements, since $k/\log k \simeq 18$ for 128-bit security. See also Appendix K for the detailed description of the CKS scheme. In the comparison, we assume that $\log k$ hardcore bits can be extracted from a single DH key [33]. Furthermore, the ciphertext overhead of our scheme is the same as that of the CS scheme. Our scheme in Sec. 6 is as efficient as the KD scheme with a weaker underlying assumption and a better reduction cost, and is as secure as the HDH-based CSK scheme with a shorter ciphertext size and a cheaper computational cost for encryption.

8 CCA-Security from BE with Verifiability

In this section, we observe that it is possible to construct a CCA-secure PKE scheme from an arbitrary verifiable BE scheme where we say that a BE is *verifiable* if any one of valid receivers can verify whether decryption results of all valid receivers are identical or not, and that security of many existing CCA-secure PKE schemes can also be explained from this viewpoint. This observa-

tion implies that a promising approach for achieving CCA-security is to concentrate on designing verifiable BE schemes. In fact, our proposed schemes are constructed based on this approach.

8.1 The Generic Conversion

Given a verifiable BE scheme $\Pi' = (\mathbf{Setup}', \mathbf{Encrypt}', \mathbf{Decrypt}')$ which is CPA-secure against selective adversaries, we construct a CCA-secure KEM $\Pi = (\mathbf{Setup}, \mathbf{Encrypt}, \mathbf{Decrypt})$. In the construction, we use a strong one-time signature scheme $\Sigma = (\mathbf{Gen}, \mathbf{Sign}, \mathbf{Verify})$ in which the verification key generated by $\mathbf{Gen}(1^k)$ has length k. We assume that the maximum number of potential users in Π' is n, and a sender can revoke t users where there exists an injective mapping (or a target collision resistant hash function) $\mathsf{INJ} : \{0,1\}^k \to \mathcal{P}$ and $\mathcal{P}$ is the set of all subsets $\mathcal{S} \subseteq \{1, ..., n\}$ with $|\mathcal{S}| = n - t$. Notice that for existence of such an injective mapping, it is necessary that ${}_nC_t \geq 2^k$. A more detailed discussion on typical parameter choice is given in Sec. 8.3. The construction of Π is as follows:

$\mathbf{Setup}(1^k)$: Choose n and t (which is a possible parameter choice for Π') such that ${}_nC_t \geq 2^k$. Run $\mathbf{Setup}'(1^k, n, t)$ to obtain $(d_1, ..., d_n, PK)$, and pick an injective mapping $\mathsf{INJ} : \{0,1\}^k \to \mathcal{P}$. The decryption key is $dk = (d_1, ..., d_n)$ and the public key is $\overline{PK} = (PK, \mathsf{INJ})$.

$\mathbf{Encrypt}(\overline{PK})$: Run $\mathbf{Gen}(1^k)$ to obtain verification key vk and signing key sk (with $|vk| = k$), and compute $\mathcal{S}_{vk} = \mathsf{INJ}(vk)$, $(\psi, K) \leftarrow \mathbf{Encrypt}'(\mathcal{S}_{vk}, PK)$ and $\sigma \leftarrow \mathbf{Sign}(sk, \psi)$. The final output is $((\psi, vk, \sigma), K)$.

$\mathbf{Decrypt}(dk, \psi, \overline{PK})$: For a ciphertext (ψ, vk, σ), check whether $\mathbf{Verify}(vk, \psi, \sigma) \stackrel{?}{=} 1$. If not, output $\perp$. Otherwise, compute $\mathcal{S}_{vk} = \mathsf{INJ}(vk)$ and output $K \leftarrow \mathbf{Decrypt}'(\mathcal{S}_{vk}, i, d_i, \psi, PK)$ where $i \in \mathcal{S}_{vk}$.

CCA-security of the above construction can be proven in a similar manner to [18]. We give an intuitive explanation for the security. Let A be an algorithm which can break CCA-security of Π. Then, it is possible to construct another algorithm B which can break Π' by using A as follows: B runs $(vk^\star, sk^\star) \leftarrow \mathbf{Gen}(1^k)$, and commits $\mathcal{S}^\star = \mathsf{INJ}(vk^\star)$ as the subset of users which will be attacked. For given public key PK of Π', B passes (PK, INJ) to A as a public key of Π. When A submits decryption query (ψ, vk, σ), B responds to it by simply decrypting the ciphertext with decryption key d_i such that $i \in \mathsf{INJ}(vk) \backslash \mathcal{S}^\star \subseteq \{1, ..., n\}$. We note that there always exists at least one such a decryption key unless $vk = vk^\star$, and $vk \neq vk^\star$ holds with an overwhelming probability if σ is a valid signature. Let $(\psi^\star, K^\star)$ be a challenge ciphertext of Π' from the challenger. Then, B gives $((\psi^\star, vk^\star, \sigma^\star), K^\star)$ to A as a challenge ciphertext of Π where $\sigma^\star \leftarrow \mathsf{Sign}(sk^\star, \psi^\star)$. A formal security proof is given in Appendix F.

Theorem 5. *If Π' is a $(\tau, \epsilon_{cpa}, n, t)$ semantically secure and $(\tau, \epsilon_{vfy}, n, t(, q_D))$ publicly (or privately) verifiable broadcast encryption scheme such that ${}_nC_t \geq 2^k$, and Σ is a (τ, ϵ_{uf}) strongly unforgeable one-time signature scheme, then Π is a $(\tau - o(\tau), \epsilon_{cpa} + \epsilon_{vfy} + \frac{1}{2}\epsilon_{uf}, q_D)$ CCA-secure key encapsulation mechanism.*

8.2 Extended Constructions

Here, we address extensions of the above basic construction.

Ext. 1: Removing One-Time Signatures. It is possible to remove the one-time signature, which may be a large ciphertext overhead, from the above basic construction by similar methods as [12] and [15]. Especially, by using the method of [15], ciphertext length can be significantly reduced if the underlying BE scheme has a specific form in which a ciphertext consists of two components $\psi = (C_0, C_1)$ and C_0 is independent to the subset of privileged users. By applying this method, a ciphertext of the modified construction becomes only ψ where $(\psi, K) \leftarrow \mathbf{Encrypt}'(\mathcal{S}_{C_0}, PK)$, $\mathcal{S}_{C_0} = \mathsf{INJ}(C_0)$, and INJ is an injective mapping (or a target collision resistant hash function). More precisely, $\mathbf{Encrypt}'$ first picks random C_0 and then decides the privileged users $\mathcal{S}_{C_0}$.

Ext. 2: Extension for Broadcast Encryption with Dynamic Join. It is trivial to extend the basic construction for BE with dynamic join (see Appendix G). If the underlying scheme is such a BE scheme, the decryption key $dk = (d_1, .., d_n)$ is replaced with master key mst, and decryption for ciphertext (ψ, vk, σ) is carried out by using d_i $(i \in \mathcal{S}_{vk})$ where $d_i \leftarrow \mathbf{Setup2}(i, PK, mst)$. This technique can be effective for compressing a decryption key.

Ext. 3: Compressing Decryption Key. As another method for compressing a decryption key, we can modify the setup algorithm of the basic construction as follows:

Setup(1^k)**:** Run $\mathbf{Setup}'(1^k, n+1, t)$ to obtain $(d_1, ..., d_{n+1}, PK)$, and pick an injective mapping $\mathsf{INJ} : \{0,1\}^k \to \mathcal{P}$ where $\mathcal{P}$ is the set of all $\mathcal{S} \subseteq \{1, ..., n+1\}$ with $|\mathcal{S}| = n - t + 1$ and $1 \in \mathcal{S}$. The decryption key is $dk = d_1$ and the public key is $\overline{PK} = (PK, \mathsf{INJ})$.

The encryption algorithm is as it is, and any ciphertext can be decrypted by using only d_1 since $1 \in \mathsf{INJ}(vk)$ always holds.

8.3 Typical Parameter Choices for n and t

For the existence of injective mapping $\mathsf{INJ} : \{0,1\}^k \to \mathcal{P}$, ${}_nC_t \geq 2^k$ is necessary. A typical parameter choice for this is $n = 2k$ and $t = k$, and in this case we have that ${}_{2k}C_k = \prod_{1 \leq i \leq k}(k \cdot i^{-1} + 1) \geq 2^k$. By using Stirling's formula,[5] a tighter estimation can be obtained: ${}_{2k}C_k \simeq (\pi \cdot k)^{\frac{-1}{2}} 2^{2k}$.

If $n \geq 2^k$, then we can set $t = 1$ or $t = n - 1$. This parameter choice is possible only when the underlying BE scheme allows dynamic join.

8.4 Relation to the IBE-to-PKE (TBE-to-PKE) Transform

We notice that the above generic conversion is identical to the CHK paradigm [18] except that the underlying primitive of CHK, i.e. IBE, is replaced with BE in our construction, and IBE can be viewed as a special case of BE with a single receiver and exponentially many potential users. This means that CHK is even applicable to weaker primitives than IBE, and can provide a variety of constructions of CCA-secure schemes. Kiltz [40] also showed that IBE is not always necessary for CHK and a weaker primitive which is called *tag-based encryption* (TBE) [45] is sufficient, and demonstrated to construct a concrete TBE scheme without using IBE-related techniques. There are also other CCA-secure schemes whose security can be explained via the TBE framework, e.g. [21, 15, 41]. Our proposed method is considered as a generic construction of TBE with a weaker primitive, i.e. BE with verifiability, than IBE.

[5] $n! \simeq (2\pi \cdot n)^{\frac{1}{2}} (\frac{n}{e})^n$, where e is the base of natural logarithm.

Table 2: Relation among broadcast encryption and public key encryption schemes. The column "(n, t)" denote a possible and typical parameter setting for each underlying broadcast encryption scheme, and $\mathsf{poly}(k)$ and $\mathsf{exp}(k)$ denote polynomial and exponential functions for the security parameter k, respectively. For verifiability, related cryptographic tools are described, and $\surd$ means that the underlying broadcast encryption has verifiability as it is. Extensions which can be used for the conversion are addressed in the next column (see Sec. 8.2).

BE Scheme	(n, t)	Verifiability	Extensions	$\Rightarrow$	PKE Scheme
Trivial BE	$(\mathsf{poly}(k), n/2)$	NIZK	–		DDN [27]
NP [47]	$(\mathsf{exp}(k), 1)$	DDH	**Ext. 1, 2**		a variant of CS [21]
		GHDH	**Ext. 1, 2**	$\Rightarrow$	Kiltz [41]
		Sec. 3.3	**Ext. 1, 2**		Ours §4
IBE	$(\mathsf{exp}(k), n-1)$	$\surd$	**Ext. 2**		CHK [18]
BGW [11]	$(\mathsf{poly}(k), n/2)$	$\surd$	**Ext. 1, 2**		Ours §9

8.5 Relations among Existing BE and PKE Schemes

Many existing CCA-secure PKE schemes can be explained via our observation in Sec. 8.1 with different underlying BE schemes, and relations among existing BE and CCA-secure PKE schemes are summarized in Table 2. We give more detailed explanations for this in Appendix H.

9 Another New CCA-Secure Scheme from Boneh-Gentry-Waters

Based on the methodology in Sec. 8, we can construct yet another new practical CCA-secure KEM from the BGW BE scheme [11]. This can be a further evidence that BE with verifiability is a powerful tool for constructing CCA-secure PKE. The proposed scheme yields tight security reduction to the 2ℓ-BDHE problem [8, 11] for relatively small ℓ, short ciphertexts and short decryption keys. The full description of the scheme is given in Appendix J. Unfortunately, this scheme is not significantly advantageous to other schemes, but it is still comparably efficient to other practical schemes (see Table 1). For a practical implementation, we set $\ell = 67$ which implies that $|\mathcal{P}| > 2^{128}$ (see Sec. 8.3).

10 A Generic Construction of CCA-Secure Broadcast Encryption

By using our methodology (with **Ext. 3**), it is also generically possible to construct a CCA-secure BE scheme from CPA-secure one with public verifiability. The conversion is fairly simple, and the resulting CCA-secure scheme can be practical. When applying this to the BGW BE scheme, we can have a new CCA-secure BE scheme with verifiability whose computational cost is slightly better than the previous scheme [11]. More detailed explanation is given in Appendix L.

Acknowledgement

The authors would like to thank Nuttapong Attrapadung and Takahiro Matsuda for their helpful comments and suggestions. The authors also would like to thank David Cash and Eike Kiltz for their invaluable comments.

References

[1] M. Abe, R. Gennaro, K. Kurosawa, and V. Shoup, "Tag-KEM/DEM: a new framework for hybrid encryption and a new analysis of Kurosawa-Desmedt KEM," Proc. of Eurocrypt'05, pp.128-146, 2005.

[2] J. Anzai, N. Matsuzaki, and T. Matsumoto, "A quick group key distribution scheme with "entity revocation"," Proc. of Asiacrypt'99, pp.333-347, 1999.

[3] M. Bellare and C. Namprempre, "Authenticated encryption: relations among notions and analysis of the generic composition paradigm," Proc. of Asiacrypt'00, pp.531-545, 2000.

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

[5] M. Bellare and P. Rogaway, "Optimal asymmetric encryption," Proc. of Eurocrypt'94, pp.92-111, 1994.

[6] M. Blum, P. Feldman, and S. Micali, "Non-interactive zero-knowledge and its applications," Proc. of STOC'88, pp.103-112, 1988.

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

[8] D. Boneh, X. Boyen, and E.-J. Goh, "Hierarchical identity based encryption with constant size ciphertext," Proc. of Eurocrypt'05, pp.440-456, 2005.

[9] D. Boneh, R. Canetti, S. Halevi, and J. Katz, "Chosen-ciphertext security from identity-based encryption," SIAM J. Comput. 36(5), pp.1301-1328, 2007.

[10] D. Boneh and M.K. Franklin, "Identity-based encryption from the Weil pairing," Proc. of Crypto'01, pp.213-229, 2001.

[11] D. Boneh, C. Gentry, and B. Waters, "Collusion resistant broadcast encryption with short ciphertexts and private keys," Proc. of Crypto'05, pp.258-275, 2005.

[12] D. Boneh and J. Katz, "Improved efficiency for CCA-secure cryptosystems built using identity-based encryption," Proc. of CT-RSA'05, pp.87-103, 2005.

[13] D. Boneh and I. Shparlinski, "On the unpredictability of bits of the elliptic curve Diffie-Hellman scheme," Proc. of Crypto'01, pp.201-212, 2001.

[14] 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, pp.129-142, 1996.

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

[16] R. Canetti, "Universally composable security: a new paradigm for cryptographic protocols," Proc. of FOCS'01, pp.136-145, 2001.

[17] R. Canetti, O. Goldreich, and S. Halevi, "The random oracle methodology, revisited," Proc. of STOC'98, pp.209-218, 1998.

[18] R. Canetti, S. Halevi, and J. Katz, "Chosen-ciphertext security from identity-based encryption," Proc. of Eurocrypt'04, pp.207-222, 2004.

[19] D. Cash and E. Kiltz, *personal communication*, 2008.

[20] D. Cash, E. Kiltz, and V. Shoup, "The twin Diffie-Hellman problem and applications," Proc. of Eurocrypt'08, pp.127-145, 2008. The full version is available from IACR ePrint 2008/067.

[21] R. Cramer and V. Shoup, "A practical public key cryptosystem provably secure against adaptive chosen ciphertext attack," Proc. of Crypto'98, pp.13-25, 1998.

[22] R. Cramer and V. Shoup, "Universal hash proofs and a paradigm for adaptive chosen ciphertext secure public-key encryption," Proc. of Eurocrypt'02, pp.45-64, 2002.

[23] R. Cramer and V. Shoup, "Design and analysis of practical public-key encryption schemes secure against adaptive chosen ciphertext attack," SIAM Journal of Computing 33:167-226, 2003.

[24] Y. Dodis and N. Fazio, "Public key broadcast encryption for stateless receivers," Proc. of ACM-DRM'02, pp.61-80, 2002.

[25] Y. Dodis and N. Fazio, "Public key trace and revoke scheme secure against adaptive chosen ciphertext attack," Proc. of PKC'03, pp.100-115, 2003.

[26] Y. Dodis, N. Fazio, A. Kiayias, and M. Yung, "Scalable public-key tracing and revoking," Proc. of PODC'03, pp.190-199, 2003.

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

[28] T. ElGamal, "A public key cryptosystem and a signature scheme based on discrete logarithms," IEEE Trans. on Inform. Theory, 31(4), pp.469-472, 1985.

[29] A. Fiat and M. Naor, "Broadcast encryption," Proc. of Crypto'93, pp.480-491, 1993.

[30] E. Fujisaki and T. Okamoto, "Secure integration of asymmetric and symmetric encryption schemes," Proc. of Crypto'99, pp.537-554, 1999.

[31] Y. Gertner, T. Malkin, and S. Myers, "Towards a separation of semantic and CCA security for public key encryption," Proc. of TCC'07.

[32] O. Goldreich, Foundations of Cryptography: Volume 2–Basic Applications, 2004.

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

[34] S. Goldwasser and S. Micali, "Probabilistic encryption," J. Comput. Syst. Sci., 28(2), pp.270-299, 1984.

[35] S. Halevi, "EME*: extending EME to handle arbitrary-length messages with associated data," Proc. of Indocrypt'04, pp.315-327, 2004.

[36] S. Halevi and P. Rogaway, "A tweakable enciphering mode," Proc. of Crypto'03, pp.482-499, 2003.

[37] S. Halevi and P. Rogaway, "A parallelizable enciphering mode," Proc. of CT-RSA'04, pp.292-304, 2004.

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

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

[40] E. Kiltz, "Chosen-ciphertext security from tag-based encryption," Proc. of TCC'06, pp.581-600, 2006.

[41] E. Kiltz, "Chosen-ciphertext secure key-encapsulation based on gap hashed Diffie-Hellman," Proc. of PKC'07, pp.282-297, 2007.

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

[43] Y. Lindell, "A simpler construction of CCA2-secure public-key encryption under general assumptions," Proc. of Eurocrypt'03, pp.241-254, 2003.

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

[45] P.D. MacKenzie, M.K. Reiter, and K. Yang, "Alternatives to non-malleability: Definitions, constructions, and applications," Proc. of TCC'04, pp. 171-190, 2004.

[46] D. Naor, M. Naor, and J. Lotspiech, "Revocation and tracing schemes for stateless receivers," Proc. of Crypto'01, pp.41-62, 2001.

[47] M. Naor and B. Pinkas, "Efficient trace and revoke schemes," Proc. of FC'00, pp. 1-20, 2000.

[48] M. Naor and M. Yung, "Public-key cryptosystems provably secure against chosen ciphertext attacks," Proc. of STOC'90, pp.427-437, 1990.

[49] P. Paillier, "Public-key cryptosystems based on composite degree residuosity classes," Proc. of Eurocrypt'99, pp.223-238, 1999.

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

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

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

[53] A. Sahai, "Non-malleable non-interactive zero knowledge and adaptive chosen-ciphertext security," Proc. of FOCS'99, pp.543-553, 1999.

[54] A. Shamir, "Identity-based cryptosystems and signature schemes," Proc. of Crypto'84, LNCS 196, Springer-Verlag, pp.47-53, 1985.

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

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

A Two Naive and Faulty Ideas for CDH-Based PKE

Here, we give two naive (and faulty) ideas for achieving CDH-based CCA-secure PKE, and show that these ideas do not work. These discussions would be helpful for understanding the technical hurdles of designing a CDH-based CCA-secure PKE scheme.

Cramer-Shoup + Hardcore Bits. The first naive idea is to modify the CS scheme by using a hardcore bit. More specifically, if it is possible to prove the one-wayness of CS encryption under the CDH assumption, then by "condensing" it into one-bit (i.e. a hardcore bit) we can obtain CCA-security from the CDH assumption. However, unfortunately it is not easy to prove the one-wayness of CS encryption under the CDH assumption. Recall the proof of CCA-security of the CS scheme under the DDH assumption. In the security proof, the simulator generates a complete decryption key by himself, and embeds the given DDH instance into only the challenge ciphertext. If the given instance is a Diffie-Hellman (DH) tuple, the adversary will correctly guess the plaintext of the challenge ciphertext. Otherwise, the adversary can output only a random bit. Therefore, by observing the adversary's behavior, the simulator can distinguish if the given instance is a DH-tuple or not. Then, we notice that it is difficult to modify this security proof to be another one under the CDH assumption even for one-wayness. Namely, for constructing a challenge ciphertext, all of four components of the DDH instance are required, and it is hard to generate it from the CDH instance since it lacks one component of the DDH instance. Therefore, this proof strategy cannot be straightforwardly extended for proving the one-wayness of CS encryption under the CDH assumption.

ElGamal + Naor-Yung. The second idea is to enhance a CDH-based semantically secure PKE to be CCA-secure one by using the Naor-Yung paradigm [48, 27, 53]. Since unlike the CS scheme, one-wayness of the ElGamal scheme [28] can be proven under the CDH assumption (against only chosen plaintext attacks), we can easily modify it to be a CDH-based semantically secure PKE scheme by using a hardcore bit. Then, it seems that we can have a CDH-based CCA-secure PKE scheme from this semantically secure PKE and the Naor-Yung paradigm. However, this is not true. Namely, it is not known if it is possible to construct an NIZK proof under the CDH assumption, and consequently, the Naor-Yung paradigm is not applicable under only this assumption. Therefore, this idea does not work either.

B Definitions

Here, we give definitions for our proposed schemes.

B.1 Key Encapsulation Mechanisms

B.1.1 The Model

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 the data encryption key $K \in \mathcal{K}$. The key K can then be used to decrypt the DEM part of hybrid encryption.

We require that if $(dk, PK) \stackrel{R}{\leftarrow}$ **Setup**(1^k) and $(\psi, K) \stackrel{R}{\leftarrow}$ **Encrypt**(PK) then **Decrypt**$(dk, \psi, PK) = K$.

B.1.2 Chosen-Ciphertext Security

CCA-security of a KEM 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. The challenger also runs algorithm **Encrypt** to obtain $(\psi^\star, K^\star) \stackrel{R}{\leftarrow}$ **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 $\psi_1, ..., \psi_{q_D}$. For query $\psi_i (\neq \psi^\star)$, the challenger responds with **Decrypt**(dk, ψ_i, PK).

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

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

Definition 1. We say that a KEM 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 $|\mathsf{AdvKEM}_\mathsf{A} - 1/2| < \epsilon$.

B.1.3 One-Wayness against CCA

One-wayness of a KEM is defined in a similar manner. 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 $(\psi^\star, K^\star) \stackrel{R}{\leftarrow}$ **Encrypt**(PK) where $K^\star \in \mathcal{K}$. It then gives the public key PK and the challenge ciphertext $\psi^\star$ to algorithm A.

Query. Algorithm A adaptively issues decryption queries $\psi_1, ..., \psi_{q_D}$. For query $\psi_i (\neq \psi^\star)$, the challenger responds with $\mathbf{Decrypt}(dk, \psi_i, PK)$.

Guess. Algorithm A outputs its guess $K' \in \mathcal{K}$ for $K^\star$ and wins the game if $K^\star = K'$.

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

Definition 2. We say that a KEM is (τ, ϵ, q_D) *one-way* if for all τ-time algorithms A who make a total of q_D decryption queries, we have that $\mathsf{AdvOW}_\mathsf{A} < \epsilon$.

B.1.4 Constrained Chosen-Ciphertext Security

In [38], Hofheinz and Kiltz proposed a relaxed notion of CCA security, which is called *Constrained CCA* (CCCA) security. According to their new composition theorem for hybrid encryption, CCCA security for KEM is sufficient for constructing a CCA-secure PKE scheme if authenticated encryption [3] is used as DEM [38]. CCCA-security for KEM is defined as follows: Both the challenger and A are given 1^k as input.

Setup. The challenger runs $\mathbf{Setup}(1^k)$ to obtain a decryption key dk and a public key PK. The challenger also runs algorithm $\mathbf{Encrypt}$ to obtain $(\psi^\star, K^\star) \xleftarrow{R} \mathbf{Encrypt}(PK)$ 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 "$\perp$")$= \mathbf{Decrypt}(dk, \psi_i, PK)$ if $pred_i(K) = 1$. It returns "$\perp$" ohtherwise.

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. There are also some small restrictions for A and E, but these are omitted here. More rigorous definition is given in [38]. Let $\mathsf{AdvKEM'}_\mathsf{A}$ denote the probability that A wins the game.

Definition 3. 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 $uncert_\mathsf{A} \le \mu$, we have that $|\mathsf{AdvKEM'}_\mathsf{A} - 1/2| < \epsilon$.

B.2 Number Theoretic Assumptions

B.2.1 The CDH, HDH, and DDH 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

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

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 4. *We say that the (τ, ϵ)-CDH assumption holds on $\mathbb{G}$ if no τ-time algorithm has advantage at least ϵ in solving the CDH problem on $\mathbb{G}$.*

Occasionally we drop the τ and ϵ and refer to the CDH in $\mathbb{G}$.

The *hashed Diffie-Hellman* (HDH) problem on $\mathbb{G}$ and function $h : \mathbb{G} \to \mathcal{D}$ 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

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

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 $T \in \mathcal{D}$, and the random bits consumed by A.

Definition 5. *We say that the (τ, ϵ)-HDH assumption holds on $\mathbb{G}$ and h if no τ-time algorithm has advantage at least ϵ in solving the HDH problem on $\mathbb{G}$ and h. Especially, we say that the (τ, ϵ)-DDH assumption holds on $\mathbb{G}$ if (τ, ϵ)-HDH assumption holds on $\mathbb{G}$ and h and $h : \mathbb{G} \to \mathbb{G}$ is the identity function.*

Occasionally we drop the τ and ϵ and refer to the HDH in $\mathbb{G}$ and h (or the DDH in $\mathbb{G}$).

Important Implications. It is important to note that the HDH assumption is strictly weaker than the DDH assumption for appropriately chosen h. If h is a *key derivation function* [23], then the DDH assumption immediately implies the HDH assumption (but not vice versa). Furthermore, if h is a hardcore bit for the Diffie-Hellman key [33, 14, 13, 39], then the CDH assumption is equivalent to the HDH assumption. Obviously, the CDH assumption is weaker than both the HDH and DDH assumptions.

B.2.2 Hardcore Bits for the Diffie-Hellman Key

Let A be a τ-time algorithm which has advantage ϵ in solving the HDH problem on $\mathbb{G}$ and $h : \mathbb{G} \to \{0, 1\}$.

Definition 6. *We say that function $h : \mathbb{G} \to \{0, 1\}$ is a (p_1, p_2) hardcore bit function in $\mathbb{G}$ if there exists a $p_1(\tau)$-time algorithm B which for any given A, can solve the CDH problem with advantage $p_2(\epsilon)$ for some polynomials p_1 and p_2.*

See [33, 14, 13, 39] for examples of hardcore bit functions for the Diffie-Hellman key.

B.2.3 The Bilinear Diffie-Hellman Exponent (BDHE) assumption

Let $\mathbb{G}_1$ (with prime order p) and $\mathbb{G}_2$ be bilinear groups where there exists a bilinear mapping $e : \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{G}_2$ such that $e(g^a, g^b) = e(g, g)^{ab}$ for all integer a and b. Then, the decisional n-bilinear Diffie-Hellman exponent (n-BDHE) problem on $\mathbb{G}_1$ is stated as follows. Let A be an

algorithm that outputs $\{0,1\}$, and we say that A has advantage ϵ in solving decision n-BDHE in $\mathbb{G}_1$ if

$$\frac{1}{2} \cdot |\Pr[\mathsf{A}(h, g, g^{\alpha}, g^{(\alpha^2)}, ..., g^{(\alpha^n)}, g^{(\alpha^{n+2})}, ..., g^{(\alpha^{2n})}, e(g^{(\alpha^{n+1})}, h)) = 0] - \Pr[\mathsf{A}(h, g, g^{\alpha}, g^{(\alpha^2)}, ..., g^{(\alpha^n)}, g^{(\alpha^{n+2})}, ..., g^{(\alpha^{2n})}, T) = 0]| \geq \epsilon,$$

where the probability is over the random choice of generators g, h in $\mathbb{G}_1$, the random choice of α in $\mathbb{Z}_p$, the random choice of $T \in \mathbb{G}_2$, and the random bits consumed by A.

Definition 7. *We say that the (decision) (τ, ϵ, n)-BDHE assumption holds on $\mathbb{G}_1$ if no τ-time algorithm has advantage at least at least ϵ in solving the (decision) n-BDHE problem on $\mathbb{G}_1$.*

Occasionally we drop the τ and ϵ and refer to the (decision) n-BDHE in $\mathbb{G}_1$.

B.3 Public Key Broadcast Encryption Schemes

Here, we review definitions for public key BE schemes. For simplicity, we define encryption schemes as key encapsulation mechanisms, and borrow the same notations as [11] with some slight modifications.

B.3.1 The Model

A BE scheme consists of the following three algorithms:

Setup$(1^k, n, t)$ Takes as input the security parameter 1^k, the number of receivers n, and the maximum number of revoked users t ($t < n$). It outputs n private keys $d_1, ..., d_n$ and a public key PK.

Encrypt$(\mathcal{S}, PK)$ Takes as input a subset $\mathcal{S} \subseteq \{1, ..., n\}$ with $|\mathcal{S}| \geq n - t$, and a public key PK. It outputs a pair (ψ, K) where ψ is called the header and $K \in \mathcal{K}$ is a message encryption key.

Let M be a message to be broadcast to the set $\mathcal{S}$ and let C_M be the encryption of M under the symmetric key K. The broadcast to users in $\mathcal{S}$ consists of $(\mathcal{S}, \psi, C_M)$. The pair $(\mathcal{S}, \psi)$ is often called the full header and C_M is often called the broadcast body.

Decrypt$(\mathcal{S}, i, d_i, \psi, PK)$ Takes as input a subset $\mathcal{S} \subseteq \{1, ..., n\}$, a user index $i \in \{1, ..., n\}$ and the private key d_i for user i, a header ψ, and the public key PK. If $i \in \mathcal{S}$ and $|\mathcal{S}| \geq n - t$, then the algorithm outputs the message encryption key $K \in \mathcal{K}$. The key K can then be used to decrypt the broadcast body C_M and obtain the message body M.

The standard key encapsulation mechanism (KEM) is a special case of BE with $n = 1$ and $t = 0$.

As usual, we require that the scheme be correct, namely that for all $\mathcal{S} \subseteq \{1, ..., n\}$ and all $i \in \mathcal{S}$, if $((d_1, ..., d_n), PK) \xleftarrow{R} \textbf{Setup}(1^k, n, t)$ and $(\psi, K) \xleftarrow{R} \textbf{Encrypt}(\mathcal{S}, PK)$ then $\textbf{Decrypt}(\mathcal{S}, i, d_i, \psi, PK) = K$.

B.3.2 Chosen-Ciphertext Security

We define CCA-security of a BE scheme against a static adversary. Security is defined using the following game between an attack algorithm A and a challenger. Both the challenger and A are given 1^k, n and t, the total number of potential users and the maximum number of revoked users, respectively, as inputs.

Init. Algorithm A begins by outputting a set $\mathcal{S}^\star \subseteq \{1, ..., n\}$ of receivers that it wants to attack, where $|\mathcal{S}^\star| \geq n - t$.

Setup. The challenger runs $\mathbf{Setup}(1^k, n, t)$ to obtain private keys $d_1, ..., d_n$ and a public key PK. The challenger also runs algorithm **Encrypt** to obtain $(\psi^\star, K^\star) \stackrel{R}{\leftarrow} \mathbf{Encrypt}(\mathcal{S}^\star, PK)$ 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 $(\psi^\star, K_b)$ to algorithm A.

Query. Algorithm A adaptively issues decryption queries $q_1, ..., q_D$ where a decryption query consists of the triple $(u, \mathcal{S}, \psi)$ where $\psi \neq \psi^\star$, $\mathcal{S} \subseteq \mathcal{S}^\star$ and $u \in \mathcal{S}$. The challenger responds with K (or $\perp$) $= \mathbf{Decrypt}(\mathcal{S}, u, d_u, \psi, PK)$.

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

Let $\mathsf{AdvBr}_{\mathsf{A},n,t}$ denote the probability that A wins the game when the challenger is given n and t.

Definition 8. We say that a broadcast encryption scheme is $(\tau, \epsilon, n, t, q_D)$ *CCA-secure* if for all τ-time algorithms A who make a total of q_D decryption queries, we have that $|\mathsf{AdvBr}_{\mathsf{A},n,t} - 1/2| < \epsilon$. Especially, we say that a broadcast encryption scheme is (τ, ϵ, n, t) *semantically secure* if it is $(\tau, \epsilon, n, t, 0)$ CCA-secure.

The above models an attack where at most t users not in set $\mathcal{S}^\star$ collude to try and expose a broadcast intended for users in $\mathcal{S}^\star$ only. The set $\mathcal{S}^\star$ is chosen by the adversary before it sees the public key PK. The parameter t is often called the collusion threshold, and we say a BE scheme is *fully collusion resistant* if $t = n - 1$. Notice that there are many useful schemes which are not fully collusion resistant [47, 25, 26], and our conversion method for obtaining CCA-security can be applied to even these schemes. In other words, full collusion resistance of BE is not necessary for providing CCA-security, while in the original CHK requires fully collusion resistant IBE.

It is possible to extend the above model for BE with *dynamic join*, where the number of actual users is not fixed at the setup phase, but only its upper bound n is known. See Appendix G for the formal definition of BE with dynamic join.

B.3.3 Verifiability

For achieving CCA-security, we need an important property for underlying BE, which we call *verifiability*. Roughly speaking, we say that a BE scheme has verifiability if a valid receiver of a broadcasted message can verify if his decryption result is the same as that for any other receiver. Verifiability is a useful property for preventing a dishonest sender from distributing invalid data (e.g. video with worse quality) for a specific subset of privileged users. We can define two flavors of verifiability: *public* verifiability and *private* verifiability. Their difference is that in a publicly verifiable BE scheme, a receiver can verify equality of keys without using his decryption key, and

on the other hand, it is necessary in a privately verifiable scheme. Both notions of verifiability are sufficient for our requirement.

Though BE schemes do not generally have verifiability, in principle it is not hard to add this property. Namely, it is generically possible to add verifiability by using NIZK proofs, assuming only existence of enhanced trap-door permutations. Notice that this NIZK proof does not need simulation-soundness which is required for a general construction of CCA-secure public key encryption [48, 53]. Namely, an NIZK proof which is used for providing verifiability may be malleable. Furthermore, BGW already has verifiability as it is, and NP BE can be efficiently modified to have this property in at least three different ways (see Sec. 3.3 and Appendix H.2). However, we should also honestly mention that it turns hard to construct a BE with verifiability when using only BE in a black-box manner.

Public Verifiablity. For public verifiability, we define adversary A's advantage $\mathsf{AdvVfy}_{\mathsf{A},n,t}$ as

$$\mathsf{AdvVfy}_{\mathsf{A},n,t} = \Pr[\exists i,j \in \mathcal{S}^\star,\ \mathbf{Decrypt}(\mathcal{S}^\star, i, d_i, \psi^\star, PK) \neq \mathbf{Decrypt}(\mathcal{S}^\star, j, d_j, \psi^\star, PK) | \\ ((d_1,...,d_n), PK) \stackrel{R}{\leftarrow} \mathbf{Setup}(1^k, n, t);\ (\mathcal{S}^\star, \psi^\star) \stackrel{R}{\leftarrow} \mathsf{A}((d_1,...,d_n), PK)].$$

Definition 9. We say that a broadcast encryption scheme is (τ, ϵ, n, t) *publicly verifiable* if for all τ-time algorithms A, we have that $\mathsf{AdvVfy}_{\mathsf{A},n,t} < \epsilon$.

Private Verifiablity. We can also define private verifiability in a similar manner, and this is formally addressed as follows: Private verifiability of BE is defined using the following game between an attack algorithm A and a challenger. Both the challenger and A are given 1^k, n and t, the total number of potential users and the maximum number of revoked users, respectively, as inputs.

Setup. The challenger runs $\mathbf{Setup}(1^k, n, t)$ to obtain private keys $d_1, ..., d_n$ and a public key PK, and gives A the public key PK.

Query. Algorithm A adaptively issues decryption queries $q_1, ..., q_{q_D}$ where a decryption query consists of the triple $(u, \mathcal{S}, \psi)$ where $|\mathcal{S}| \geq n - t$ and $u \in \mathcal{S}$. The challenger responds with $\mathbf{Decrypt}(\mathcal{S}, u, d_u, \psi, PK)$.

Forge. Algorithm A outputs an encapsulated key $(\mathcal{S}^\star, \psi^\star)$ with $|\mathcal{S}^\star| \geq n - t$ and wins the game if there is a pair of users $i, j \in \mathcal{S}^\star$ such that $\mathbf{Decrypt}(\mathcal{S}^\star, i, d_i, \psi^\star, PK) \neq \mathbf{Decrypt}(\mathcal{S}^\star, j, d_j, \psi^\star, PK)$.

Let $\mathsf{AdvVfy}'_{\mathsf{A},n,t}$ denote the probability that A wins the game when the challenger is given n and t.

Definition 10. We say that a broadcast encryption scheme is $(\tau, \epsilon, n, t, q_D)$ *privately verifiable* if for all τ-time algorithms A who make a total of q_D decryption queries, we have that $\mathsf{AdvVfy}'_{\mathsf{A},n,t} < \epsilon$.

One may think that the notion of private verifiability is too weak in practice since this does not model collusion attacks of valid users in $\{1, ..., n\}$. However, for achieving CCA-security by our proposed method, this notion of verifiability is sufficient. In other words, full verifiability, which can be required for practical applications, is not necessary for the proposed method. Obviously, public verifiability implies private verifiability.

B.4 Other Cryptographic Tools

B.4.1 One-Time Signatures

A signature scheme consists of the following three algorithms:

Gen(1^k) Takes as input the security parameter 1^k, and outputs a verification key vk and a signing key sk.

Sign(sk, m) Takes as input a signing key sk and a message m, and outputs a signature σ.

Verify(vk, m, σ) Takes as input a verification key vk, a message m, and a signature σ, and outputs a bit $b \in \{0, 1\}$.

We require that for all sk, all m in the message space, and all σ output by **Sign**(sk, m), we have **Verify**$(vk, m, \sigma) = 1$.

Next, we define strong unforgery of a (one-time) signature scheme against chosen message attacks. Security 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 **Gen**(1^k) to obtain a verification key vk and a signing key sk. It gives A the verification key vk.

Query. Algorithm A may issue at most one signing query m. The challenger responds with $\sigma \xleftarrow{R}$ **Sign**(sk, m).

Forge. Algorithm A outputs $(m^\star, \sigma^\star)$ such that $(m^\star, \sigma^\star) \neq (m, \sigma)$.

Let $\mathsf{AdvOTS}_\mathsf{A}$ denote the probability that **Verify**$(vk, m^\star, \sigma^\star) = 1$.

Definition 11. We say that a signature scheme is (τ, ϵ) *strongly unforgeable* if for all τ-time algorithm A, we have that $\mathsf{AdvOTS}_\mathsf{A} < \epsilon$.

B.4.2 Target Collision Resistant Hash Functions

Let $\mathsf{TCR} : \mathcal{X} \to \mathcal{Y}$ be a hash function (we individually define the range and domain of TCR for each scheme), A be an algorithm, and A's advantage $\mathsf{AdvTCR}_\mathsf{A}$ be

$$\mathsf{AdvTCR}_\mathsf{A} = \Pr[\mathsf{TCR}(x') = \mathsf{TCR}(x) \in \mathcal{Y} \wedge x' \neq x |\ x \xleftarrow{R} \mathcal{X};\ x' \xleftarrow{R} \mathsf{A}(x)].$$

Definition 12. We say that TCR is a (τ, ϵ) *target collision resistant hash function* if for all τ-time algorithm A, we have that $\mathsf{AdvTCR}_\mathsf{A} < \epsilon$.

It is obvious that any injective mapping can be used as a perfectly secure target collision resistant hash function.

B.5 The Lagrange Interpolation and a Remark

Here, we give a concrete description of the Lagrange interpolation as well as an important remark. Let $f(x) = \sum_{0 \le j \le t} a_j x^j$ be a polynomial over $\mathbb{Z}_p$ with degree t where p is a prime, and $(x_0, f(x_0)), ..., (x_t, f(x_t))$ be $t+1$ distinct points over $f(x)$. Then, for given $(x_0, f(x_0)), ..., (x_t, f(x_t))$ one can reconstruct $f(x)$ as

$$f(x) = f(x_0)\lambda_{x_0}(x) + \cdots + f(x_t)\lambda_{x_t}(x),$$

where for $0 \le j \le t$,

$$\lambda_{x_j}(x) = \frac{(x-x_0)(x-x_1)\cdots(x-x_{j-1})(x-x_{j+1})\cdots(x-x_t)}{(x_j-x_0)(x_j-x_1)\cdots(x_j-x_{j-1})(x_j-x_{j+1})\cdots(x_j-x_t)}.$$

By a careful calculation, we notice that for a multiplicative group $\mathbb{G}$ of prime order p, it is also possible to reconstruct any g^{a_j} for $0 \le j \le t$ from $(g, (x_0, g^{f(x_0)}), ..., (x_t, g^{f(x_t)}))$, where g is any element of $\mathbb{G}$. In this paper, we often use this fact for security proofs.

Similarly to this, for given $(g, g^{a_0}, ..., g^{a_{\ell-1}}, (x_0, g^{f(x_0)}), ..., (x_{t-\ell}, g^{f(x_{t-\ell})}))$, one can reconstruct any g^{a_j} for $\ell \le j \le t$ as well.

C Direct CCA-Secure PKE from the CDH assumption

Here, we give a direct construction of a CCA-secure PKE scheme under the CDH assumption without depending on the KEM/DEM framework. The difference between the proposed PKE scheme and the proposed KEM is similar to that between the CS PKE scheme [21] and the Shoup KEM [55]. See also [1, page 136].

The proposed PKE scheme is the same as the proposed one-way KEM in Sec. 4.1 except that

Setup(1^k)**:** Pick also a function $h : \mathbb{G} \to \{0,1\}$, which is a hardcore bit function for the Diffie-Hellman key in $\mathbb{G}$. The decryption key is $f(x)$, and the public key is $PK = (\mathbb{G}, g, y_0, y_1, y_2, y_3, \mathsf{TCR}, h)$, where $\mathsf{TCR} : \mathbb{G} \times \{0,1\} \times \{0,1\} \to \mathbb{Z}_p^*$. The message space is $\{0,1\}$.

Encrypt(PK, M)**:** For encrypting a plaintext $M \in \{0,1\}$, pick a random $r \xleftarrow{R} \mathbb{Z}_p$, and compute

$$\psi = (g^r, g^{r \cdot f(\mathbf{i})}, g^{r \cdot f(\mathsf{i})}, h(y_0^r) \oplus M)$$

where $\mathbf{i} = \mathsf{TCR}(g^r, h(y_0^r) \oplus M, 0)$ and $\mathsf{i} = \mathsf{TCR}(g^r, h(y_0^r) \oplus M, 1)$. The ciphertext is ψ.

Decrypt(dk, ψ, PK)**:** For a ciphertext $\psi = (C_0, C_1, C_2, C_3)$, check whether $(C_0^{f(\mathbf{i})}, C_0^{f(\mathsf{i})}) \stackrel{?}{=} (C_1, C_2)$, where $\mathbf{i} = \mathsf{TCR}(C_0, C_3, 0)$ and $\mathsf{i} = \mathsf{TCR}(C_0, C_3, 1)$. If not, output $\perp$. Otherwise, output $M = C_3 \oplus h(C_0^{a_0})$.

The above scheme is CCA-secure PKE [52, 27] under the CDH assumption. A sketch of the security proof is as follows: Assume we are given an adversary A which breaks CCA-security of the above PKE scheme. We use A to construct another adversary B which for given $(g, g^\alpha, g^\beta) \in \mathbb{G}^3$ distinguishes $h(g^{\alpha\beta})$ from a random bit. It should be noticed that existence of such B immediately implies existence of yet another adversary which solves the CDH problem because h is a hardcore bit function. For given (g, g^α, g^β), B picks a random bit c and a target collision resistant hash

function TCR, and computes $\mathbf{i}^\star = \mathsf{TCR}(g^\beta, c, 0)$ and $\mathsf{i}^\star = \mathsf{TCR}(g^\beta, c, 1)$. Similarly to the proof of Theorem 1, B picks rnd, $u_{\mathbf{i}^\star}$, $u_{\mathsf{i}^\star}$, and u_{rnd}, and computes $y_0, ..., y_3$. Then, B inputs public key $PK = (\mathbb{G}, g, y_0, y_1, y_2, y_3, \mathsf{TCR})$ to A. For any decryption query from A, B can correctly respond to it by using $(u_{\mathbf{i}^\star}, u_{\mathsf{i}^\star}, u_{rnd}, rnd)$ with an overwhelming probability. As the challenge ciphertext, B sets $\psi^\star = (g^\beta, (g^\beta)^{u_{\mathbf{i}^\star}}, (g^\beta)^{u_{\mathsf{i}^\star}}, c)$. Recall that since the message space is $\{0,1\}$, the message pair which will be challenged is always $m_0 = 0$ and $m_1 = 1$. When A outputs his guess b', B outputs $c \oplus b'$ as his guess for $h(g^{\alpha\beta})$. B's advantage is negligibly close to A's advantage.

D Proof of Theorem 3

Assume that for challenge ciphertext $(g^\beta, g^{\beta \cdot f(\mathbf{i}^\star)}, g^{\beta \cdot f(\mathsf{i}^\star)})$ such that $\mathbf{i}^\star = \mathsf{TCR}(g^\beta, 0)$ and $\mathsf{i}^\star = \mathsf{TCR}(g^\beta, 1)$, there exists an adversary A' which distinguishes $(h(y_0^\beta)||h(y_1^\beta)||...||h(y_{k-1}^\beta))$ from a random k-bit string. Then, by a standard hybrid argument, there also exists another adversary A which for some j such that $0 \leq j \leq k-1$ distinguishes

$$(h(y_0^\beta)||h(y_1^\beta)||...||h(y_j^\beta)||random_{k-j-1})$$

from

$$(h(y_0^\beta)||h(y_1^\beta)||...||h(y_{j-1}^\beta)||random_{k-j})$$

where $random_\ell$ denotes an ℓ-bit random string.

Now, assume we are given such an adversary A which with running time τ, advantage ϵ, and q_D decryption queries. We use A to construct another adversary B which for given (g, g^α, g^β) distinguishes $h(g^{\alpha\beta})$ from a random bit. Define adversary B as follows:

1. For given (g, g^α, g^β), B picks a target collision resistant hash function TCR, and computes $\mathbf{i}^\star = \mathsf{TCR}(g^\beta, 0)$ and $\mathsf{i}^\star = \mathsf{TCR}(g^\beta, 1)$.

2. B sets $y_j = g^\alpha$, and picks distinct randoms $rnd_j, ..., rnd_{k-1}$ from $\mathbb{Z}_p^* \backslash \{\mathbf{i}^\star, \mathsf{i}^\star\}$. B also picks randoms $u_{\mathbf{i}^\star}$, $u_{\mathsf{i}^\star}$, $a_0, ..., a_{j-1}$, and $u_j, ..., u_{k-1}$ from $\mathbb{Z}_p$.

3. B calculates $y_l = g^{a_l}$ for $0 \leq l \leq j-1$.

4. Then, by using the Lagrange interpolation, B calculates $y_{j+1}, ..., y_{k+2}$ such that for a function $F(x) = \prod_{0 \leq j \leq k+2} y_j^{x^j}$, $F(\mathbf{i}^\star) = g^{u_{\mathbf{i}^\star}}$, $F(\mathsf{i}^\star) = g^{u_{\mathsf{i}^\star}}$, and $F(rnd_j) = g^{u_j}, ..., F(rnd_{k-1}) = g^{u_{k-1}}$ hold.

5. B inputs public key $PK = (\mathbb{G}, g, y_0, y_1, ..., y_{k+2}, \mathsf{TCR}, h)$ and challenge ciphertext $\psi^\star = (g^\beta, (g^\beta)^{u_{\mathbf{i}^\star}}, (g^\beta)^{u_{\mathsf{i}^\star}})$ and $K^\star = (h((g^\beta)^{a_0})||h((g^\beta)^{a_1})||...||h((g^\beta)^{a_{j-1}})||\gamma||random_{k-j-1})$ to A for a random bit γ.

6. When A makes 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 $\mathsf{TCR}(C_0, b) = [\mathbf{i}^\star, \mathsf{i}^\star, rnd_j, ..., rnd_{k-2}$ or $rnd_{k-1}]$ for $b = 0$ or 1, then B aborts and outputs a random bit.

(c) If $C_0 \neq g^\beta$ and $\mathsf{TCR}(C_0, b) \neq [\mathbf{i}^\star, \mathsf{i}^\star, rnd_j, ..., rnd_{k-2}$ nor $rnd_{k-1}]$ for both $b = 0$ and 1, B computes $C_0^{u_{\mathbf{i}^\star}}, C_0^{u_{\mathsf{i}^\star}}, C_0^{u_j}, ..., C_0^{u_{k-2}}$, and $C_0^{u_{k-1}}$. Let $\mathsf{TCR}(C_0, 0) = \mathbf{i}$ and $\mathsf{TCR}(C_0, 1) = \mathsf{i}$, and f_1, f_2, and f_3 be polynomials over $\mathbb{Z}_p$ with degree $k+2$ whose coefficient for x^l term is a_l for $0 \leq l \leq j-1$, such that

$$\begin{aligned}
(f_1(\mathbf{i}), f_1(\mathsf{i}), f_1(\mathbf{i}^\star), f_1(\mathsf{i}^\star), f_1(rnd_{j+1}), ..., f_1(rnd_{k-1})) &= (\log_{C_0} C_1, \log_{C_0} C_2, u_{\mathbf{i}^\star}, u_{\mathsf{i}^\star}, u_{j+1}, ..., u_{k-1}) \\
(f_2(\mathbf{i}), f_2(\mathsf{i}), f_2(\mathbf{i}^\star), f_2(rnd_j), ..., f_2(rnd_{k-1})) &= (\log_{C_0} C_1, \log_{C_0} C_2, u_{\mathbf{i}^\star}, u_j, ..., u_{k-1}) \\
(f_3(\mathbf{i}), f_3(\mathsf{i}), f_3(\mathsf{i}^\star), f_3(rnd_j), ..., f_3(rnd_{k-1})) &= (\log_{C_0} C_1, \log_{C_0} C_2, u_{\mathsf{i}^\star}, u_j, ..., u_{k-1}).
\end{aligned}$$

Then, B calculates ${C_0}^{a_{1,l}}, {C_0}^{a_{2,l}}, {C_0}^{a_{3,l}}$ by using the Lagrange interpolation where $a_{1,l}$, $a_{2,l}$, and $a_{3,l}$ denote the coefficients of x^l term of f_1, f_2, and f_3 for $j \leq l \leq k-1$, respectively, and responds $(h(C_0^{a_0})||...||h(C_0^{a_{j-1}})||h(C_0^{a_{1,j}})||...||h(C_0^{a_{1,k-1}}))$ if ${C_0}^{a_{1,j}} = {C_0}^{a_{2,j}} = {C_0}^{a_{3,j}}$, or "$\perp$" otherwise.

7. 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 A correctly distinguishes the key, Abort denote the event that A submits a ciphertext $\psi = (C_0, C_1, C_2)$ such that $C_0 \neq g^\beta$ and $\mathsf{TCR}(C_0, b) = [\mathbf{i}^\star, \mathsf{i}^\star, rnd_j, ..., rnd_{k-2}$ or $rnd_{k-1}]$ for $b = 0$ or 1, and $\mathsf{Invalid}$ denote the event that A submits a ciphertext $\psi = (C_0, C_1, C_2)$ such that B does not abort, ${C_0}^{a_{1,j}} = {C_0}^{a_{2,j}} = {C_0}^{a_{3,j}}$, but $(C_1, C_2) \neq (C_0^{f(\mathbf{i})}, C_0^{f(\mathsf{i})})$ where $f(x) = \sum_{0 \leq i \leq k+2} a_i x^i$.

Then, B's advantage for guessing $h(g^{\alpha\beta})$ is estimated as follows:

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

Now, we prove following lemmas.

Lemma 3. $\Pr[\mathsf{Abort}] \leq 2\epsilon_{tcr} + \frac{2q_D k}{p-3}$.

Proof. Assume we are given an adversary A with $\Pr[\mathsf{Abort}] = p_A$. Then, we can construct another adversary B' which finds a collision in TCR as follows. For a given TCR instance (C, b), B' generates decryption key $f(x)$ and public key $PK = (\mathbb{G}, g, y_0, y_1, ..., y_{k+2}, \mathsf{TCR}, h)$, and computes challenge ciphertext $\psi^\star = (C, C^{u_{\mathbf{i}^\star}}, C^{u_{\mathsf{i}^\star}})$, where $u_{\mathbf{i}^\star} = f(\mathsf{TCR}(C, 0))$ and $u_{\mathsf{i}^\star} = f(\mathsf{TCR}(C, 1))$. B' also picks distinct randoms $rnd_j, ..., rnd_{k-1}$ from $\mathbb{Z}_p^* \backslash \{\mathbf{i}^\star, \mathsf{i}^\star\}$, and gives PK and $(\psi^\star, K^\star)$ to A, where $K^\star$ is a correct key under $f(x)$ or a random element of $\mathbb{G}$ with probability 1/2.

Since $rnd_j, ..., rnd_{k-1}$ are information-theoretically hidden to A, for a query $\psi = (C_0, C_1, C_2)$, $[\mathsf{TCR}(C_0, 0)$ or $\mathsf{TCR}(C_0, 1)] = [rnd_j, ..., rnd_{k-2}$, or $rnd_{k-1}]$ happens with probability at most $2(k-j)/(p-3)$. Therefore, the probability that A submits a ciphertext $\psi = (C_0, C_1, C_2)$ such that $C_0 \neq C$ and $[\mathsf{TCR}(C_0, 0)$ or $\mathsf{TCR}(C_0, 1)] = [\mathbf{i}^\star$ or $\mathsf{i}^\star]$ is at least $p_A - 2q_D(k-j)/(p-3)$. Since b is also information-theoretically indistinguishable, $[\mathsf{TCR}(C_0, 0)$ or $\mathsf{TCR}(C_0, 1)] = \mathsf{TCR}(C, b)$ happens with probability at least $1/2(p_A - 2q_D(k-j)/(p-3))$. Hence, we have $\epsilon_{tcr} \geq 1/2(p_A - 2q_D(k-j)/(p-3)) \geq 1/2(p_A - 2q_D k/(p-3))$. □

Lemma 4. $\Pr[\mathsf{Invalid}] \leq \frac{q_D}{p-k-2}$.

Proof. Let $f_0(x) = \sum_{0 \leq l \leq j-1} a_l x^l$, and $f'_1(x), f'_2(x)$, and $f'_3(x)$ be polynomials such that $f_l(x) = f_0(x) + x^j \cdot f'_l(x)$ for $l = 1, 2, 3$. Let $f'(x)$ be a polynomial such that $f(x) = f_0(x) + x^j \cdot f'(x)$. Suppose $\psi = (C_0, C_1, C_2)$ is a ciphertext such that B does not abort, ${C_0}^{f'_1(0)} = {C_0}^{f'_2(0)} = {C_0}^{f'_3(0)}$, but $(C_1, C_2) \neq (C_0^{f(\mathbf{i})}, C_0^{f(\mathsf{i})})$. Then, we notice that f'_1 and f'_2 which are polynomials with degree $k-j+2$ have $k-j+3$ intersections, and consequently they have to be identical. Similarly, we have that $f'_1 = f'_2 = f'_3$. This implies that for $[\mathsf{Invalid} = \text{true}]$, A has to choose C_1 and C_2 (without knowing $rnd_j, ..., rnd_{k-1}$) such that f'_1 (with degree $k-j+2$) satisfies

1. $(f'_1(\mathbf{i}), f'_1(\mathsf{i}), f'_1(\mathbf{i}^\star), f'_1(\mathsf{i}^\star), f'_1(rnd_j), ..., f'_1(rnd_{k-1}))$
   $= ((\log_{C_0} C_1 - f_0(\mathbf{i})) \cdot \mathbf{i}^{-j}, (\log_{C_0} C_2 - f_0(\mathsf{i})) \cdot \mathsf{i}^{-j}, f'(\mathbf{i}^\star), f'(\mathsf{i}^\star), f'(rnd_j), ..., f'(rnd_{k-1}))$,
2. $f'_1 \neq f'$.

Since f'_1 and f' have at most $k-j+2$ intersections and $k-j+1$ of them are $(\mathbf{i}^\star, f'(\mathbf{i}^\star))$, $(\mathsf{i}^\star, f'(\mathsf{i}^\star))$, $(rnd_{j+1}, f'(rnd_{j+1})), ..., (rnd_{k-1}, f'(rnd_{k-1}))$, there is only one remained intersection which must be $(rnd_j, f'(rnd_j))$. Therefore, $[\mathsf{Invalid} = \text{true}]$ happens only when A correctly guesses the value of rnd_j (even if A is given $rnd_{j+1}, ..., rnd_{k-1}$). Hence, for any invalid query ψ, the probability that B does not respond "$\perp$" is at most $1/(p-k+j-2) (\leq 1/(p-k-2))$. □

A's advantage is estimated as at least $1/k$ times A''s advantage due to the hybrid argument.

E Proof of Theorem 4

Assume we are given an adversary A which breaks CCCA security of the above KEM with running time τ, advantage ϵ, and q_D decryption queries with $uncert_{\mathsf{A}} = \mu$. We use A to construct another adversary B which solves the HDH problem. Define adversary B as follows:

1. For a given HDH instance $(g, g^\alpha, g^\beta, K^\star)$, B picks a target collision resistant hash function TCR, and computes $i^\star = \mathsf{TCR}(g^\beta)$.
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. Then, by using the Lagrange interpolation, B calculates y_1 and y_2 such that for a function $F(x) = \prod_{0 \leq j \leq 2} y_j^{x^j}$, $F(i^\star) = g^{u_{i^\star}}$ and $F(rnd) = g^{u_{rnd}}$ hold. Notice that letting $y_j = g^{a_j}$ for $0 \leq j \leq 2$, $F(x)$ is rephrased as $F(x) = g^{f(x)}$ where $f(x) = \sum_{0 \leq i \leq 2} a_j x^j$, and therefore, one can easily computes g^{a_j} for $0 \leq j \leq 2$ by using the Lagrange coefficients for $f(x)$.
4. B inputs public key $PK = (\mathbb{G}, g, y_0, y_1, y_2, \mathsf{TCR}, h)$ and challenge ciphertext $(\psi^\star (= (g^\beta, (g^\beta)^{u_{i^\star}})), K^\star)$ to A. We note that this is a correct ciphertext and its corresponding data encryption key is $h(g^{\alpha\beta})$.
5. When A makes decryption query $(\psi (= (C_0, C_1)), pred(\cdot))$, B proceeds as follows:
   (a) If $C_0 = g^\beta$, then B responds $\perp$.
   (b) If $C_0 \neq g^\beta$ and $\mathsf{TCR}(C_0) = [i^\star \text{ or } rnd]$, then B aborts and outputs a random bit.

(c) If $C_0 \neq g^\beta$ and $\mathsf{TCR}(C_0) \neq [i^\star \text{ nor } rnd]$, 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 $K = h({C_0}^{f'(0)})$ if $pred(K) = 1$, or "$\perp$" otherwise.

6. Finally, A outputs a guess b, and B outputs the same value as his own guess for the given HDH instance.

Let Win denote the event that A correctly distinguishes the key, Abort denote the event that A submits a ciphertext $\psi = (C_0, C_1)$ such that $C_0 \neq g^\beta$ and $\mathsf{TCR}(C_0) = [i^\star \text{ or } rnd]$, and $\mathsf{Invalid}$ denote the event that A submits a ciphertext $\psi = (C_0, C_1)$ and predicate $pred$ such that B does not abort, $pred(h(C_0^{f'(0)})) = 1$, and $\psi \neq (C_0, C_0^{f(i)})$ where $f(x) = \sum_{0 \leq j \leq 2} a_j x^j$.

Then, B's advantage in solving the HDH problem is estimated as follows:

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

The proof completes by proving following lemmas.

Lemma 5. $\Pr[\mathsf{Abort}] \leq \epsilon_{tcr} + \frac{q_D}{p-2}$.

Proof. Assume we are given an adversary A with $\Pr[\mathsf{Abort}] = p_A$. Then, we can construct another adversary B' which finds a collision in TCR as follows. For a given TCR instance C, B' generates decryption key $f(x)$ and public key $PK = (\mathbb{G}, g, y_0, y_1, y_2, \mathsf{TCR}, h)$ of the above KEM, and computes challenge ciphertext $\psi^\star = (C, C^{f(u_{i^\star})})$, where $u_{i^\star} = \mathsf{TCR}(C)$. B' also picks a random rnd from $\mathbb{Z}_p^* \backslash \{i^\star\}$, and gives PK and $(\psi^\star, K^\star)$ to A, where $K^\star = h(C^{f(0)})$ or a random ν-bit string with probability $1/2$.

Since rnd is information-theoretically hidden to A, for a query $\psi = (C_0, C_1)$ $\mathsf{TCR}(C_0) = rnd$ happens with probability at most $1/(p-2)$. Therefore, the probability that A submits a ciphertext $\psi = (C_0, C_1)$ such that $C_0 \neq C$ and $\mathsf{TCR}(C_0) = i^\star$ is at least $p_A - q_D/(p-2)$. Hence, we have $\epsilon_{tcr} \geq p_A - q_D/(p-2)$. □

Lemma 6. $\Pr[\mathsf{Invalid}] \leq q_D(\mu + \frac{3}{p-2})$.

Proof. Suppose $(\psi(= (C_0, C_1)), pred)$ is a ciphertext such that B does not abort, $pred(h(C_0^{f'(0)})) = 1$, but $C_1 \neq C_0^{f(i)}$. 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 $u \neq f(i)$. For $rnd \in (\mathbb{Z}_p^* \backslash \{i^\star\})$, let $f_{rnd}(x)$ be a polynomial of degree at most two such that

$$f_{rnd}(i^\star) = f(i^\star), \quad f_{rnd}(i) = u, \quad f_{rnd}(rnd) = f(rnd).$$

Then, we will show that for any $(rnd_1, rnd_2) \in (\mathbb{Z}_p^* \backslash \{i^\star\})^2$,

$$f_{rnd_1}(0) \neq f_{rnd_2}(0)$$

if $rnd_1 \neq rnd_2$. 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 $f_{rnd_1}(i) = u \neq f(i)$.

Hence, even if A has unlimited computational power, $[pred(h(C_0^{f'(0)})) = 1]$ happens only when A's guess for rnd is correct or it accidentally occurs according to the probability $uncert_{\mathsf{A}}(= \mu)$ (more precisely, $\mu + 2/p (\leq \mu + 2/(p-2))$ where $2/p$ comes from statistical distance between the distribution of possible values for rnd and the uniform distribution over $\mathbb{Z}_p$). □

F Proof of Theorem 5

Assume we are given an adversary A which breaks CCA-security of Π with running time τ, advantage ϵ, and q_D decryption queries. Let $(\psi^\star, vk^\star, \sigma^\star)$ denote the challenge ciphertext received by A during a particular run of the experiment, and let Win denote the event that A wins the game, $\mathsf{Invalid}$ denote the event that A submits a ciphertext (ψ, vk, σ) such that $[\exists i, j \in \mathcal{S}_{vk},\ \mathbf{Decrypt}'(\mathcal{S}_{vk}, i, d_i, \psi, PK) \neq \mathbf{Decrypt}'(\mathcal{S}_{vk}, j, d_j, \psi, PK)]$, and Forge denote the event that A submits a ciphertext (ψ, vk, σ) to the decryption oracle such that $[vk = vk^\star$ and $\mathbf{Verify}(vk, \psi, \sigma) = 1]$.

We use A to construct another adversary B which breaks semantic security of the underlying BE scheme Π' (in the sense of Def. 8). Define adversary B as follows:

1. B runs $\mathbf{Gen}(1^k)$ to generate $(vk^\star, sk^\star)$, and outputs the target users $\mathcal{S}^\star = \mathsf{INJ}(vk^\star)$, where $\mathsf{INJ} : \{0,1\}^k \to \mathcal{P}$ is an injective mapping.
2. B is given PK and $(d_i)_{i \in \{1,\ldots,n\} \backslash \mathcal{S}^\star}$, where PK is a public key of Π' and d_i is a decryption key of user i of Π', respectively. B is also given a challenge ciphertext $(\psi^\star, K^\star)$. Then, B inputs (PK, INJ) and $(\psi^\star, vk^\star, \sigma^\star)$ to A as a public key and a challenge ciphertext of Π, respectively, where $\sigma^\star \leftarrow \mathsf{Sign}(sk^\star, \psi^\star)$.
3. When A makes decryption query (ψ, vk, σ), B proceeds as follows:
   (a) If $vk = vk^\star$, then B checks whether $\mathbf{Verify}(vk, \psi, \sigma) = 1$. If so, B aborts and outputs a random bit. Otherwise, it simply responds $\perp$.
   (b) If $vk \neq vk^\star$ and $\mathbf{Verify}(vk, \psi, \sigma) = 0$, then B responds with $\perp$.
   (c) If $vk \neq vk^\star$ and $\mathbf{Verify}(vk, \psi, \sigma) = 1$, then B computes $\mathbf{Decrypt}'(\mathcal{S}_{vk}, i, d_i, \psi, PK)$ for some $i \in \mathcal{S}_{vk}$. Notice that $\mathcal{S}_{vk} \backslash \mathcal{S}_{vk^\star}$ is not an empty set if $vk \neq vk^\star$.
4. Finally, A outputs a guess b', and B outputs the same bit as his guess.

Then, B's success probability $\mathsf{AdvBr}_{\mathsf{B},n,t}$ is estimated as follows:

$$\begin{aligned}
\mathsf{AdvBr}_{\mathsf{B},n,t} &\geq \Pr[\mathsf{Win} | \overline{\mathsf{Forge}} \wedge \overline{\mathsf{Invalid}}] \Pr[\overline{\mathsf{Forge}} \wedge \overline{\mathsf{Invalid}}] + \frac{1}{2} \Pr[\mathsf{Forge}] \\
&\geq (\frac{1}{2} + \epsilon - \Pr[\mathsf{Invalid}] - \Pr[\mathsf{Forge}]) + \frac{1}{2} \Pr[\mathsf{Forge}] = \frac{1}{2} + \epsilon - \Pr[\mathsf{Invalid}] - \frac{1}{2} \Pr[\mathsf{Forge}].
\end{aligned}$$

Consequently we have that $\epsilon \leq \epsilon_{cpa} + \epsilon_{vfy} + \frac{1}{2}\epsilon_{uf}$.

G Broadcast Encryption with Dynamic Join

A BE scheme with dynamic join consists of four algorithms: **Setup1**, **Setup2**, **Encrypt**, and **Decrypt** which are the same as those for BE (without dynamic join) except that the algorithm **Setup** is separated into two independent algorithms: **Setup1** and **Setup2**, such that $(PK, mst) \stackrel{R}{\leftarrow}$ **Setup1**$(1^k, n, t)$ and $d_i \stackrel{R}{\leftarrow}$ **Setup2**(i, PK, mst), where mst is a "master key". We can simply set $mst = (d_1, ..., d_n)$ for any given n where $n = O(\mathsf{poly}(k))$, but for a (polynomially) unbounded number of potential users, mst can not be this form. This extension is useful for capturing full functionality of some BE schemes, e.g. [47], which can deal with an exponentially many number of potential users.

According to this extension, the attack model is slightly changed as follows. In **Init.** phase, A also chooses a set of colluders $\mathcal{S}_{col} \subseteq \{1, ..., n\} \backslash \mathcal{S}^\star$ where $|\mathcal{S}_{col}| \leq t$. If t is a polynomial of k, in **Setup.** phase the challenger runs **Setup1**$(1^k, n, t)$ and **Setup2**(i, PK, mst) to obtain (PK, mst) and $(d_i)_{i \in \mathcal{S}_{col}}$, respectively, and A is given $(d_i)_{i \in \mathcal{S}_{col}}$. If t is a super-polynomial of k, A is given an oracle which A can adaptively ask d_i for any $i \in \mathcal{S}_{col}$ in any order. This oracle can be accessed in **Query**. phase. Let $\mathsf{AdvBr}'_{\mathsf{A},n,t}$ denote the probability that A wins the game when the challenger is given n and t.

Definition 13. We say that a broadcast encryption scheme with dynamic join is $(\tau, \epsilon, n, t, q_D)$ *CCA-secure* if for all τ-time algorithms A who make a total of q_D decryption queries, we have that $|\mathsf{AdvBr}'_{\mathsf{A},n,t} - 1/2| < \epsilon$. Especially, we say that a broadcast encryption scheme with dynamic join is (τ, ϵ, n, t) *semantically secure* if it is $(\tau, \epsilon, n, t, 0)$ CCA-secure.

It is obvious that a $(\tau, \epsilon, n, t, q_D)$ CCA-secure BE scheme (without dynamic join) implies a $(\tau, \epsilon, n, t, q_D)$ CCA-secure BE scheme with dynamic join.

H Relations among Existing BE and PKE Schemes

Here, we discuss constructions of existing CCA-secure PKE schemes from the viewpoint of the methodology in Sec. 8.1. Table 2 summarizes relations among existing BE and CCA-secure PKE schemes. We explain individual cases in more detail below.

H.1 Dolev-Dwork-Naor from Trivial Broadcast Encryption

A trivial construction of BE with verifiability is as follows. In the setup phase, n key pairs $(d_i, PK_i)_{1 \leq i \leq n}$ of a semantically secure public key encryption scheme are generated, where n is the number of potential users. User i is given d_i as his decryption key, and a sender generates a ciphertext for privileged receivers $\mathcal{S} \subseteq \{1, ..., n\}$ by using public keys $(PK_i)_{i \in \mathcal{S}}$. For verifiability, we add an NIZK proof for equivalence of the decryption results.

By our construction, the above trivial BE (with $n = 2k$) is transformed into a CCA-secure public key encryption scheme as follows. Let $dk = (d_1, ..., d_{2k})$ be the decryption key, and $PK = (PK_1, ..., PK_{2k}, \mathsf{INJ}, r)$ be the public key, where INJ is the same injective mapping as the basic construction with $n = 2k$ and $t = k$, and r is a common random string for NIZK proofs. For generating a ciphertext, the sender generates verification key vk and signing key sk of a one-time signature scheme, computes $\mathcal{S}_{vk} = \mathsf{INJ}(vk) \subseteq \{1, ..., 2k\}$, and generates component ciphertexts $(C_i)_{i \in \mathcal{S}_{vk}}$ by encrypting the same session key K with $(PK_i)_{i \in \mathcal{S}_{vk}}$, respectively. The sender also

generates an NIZK proof π for equivalence of the decryption results of $(C_i)_{i\in\mathcal{S}_{vk}}$ by using random string r, and adds a signature σ for all other components by using signing key sk. The ciphertext consists of $((C_i)_{i\in\mathcal{S}_{vk}}, vk, \pi, \sigma)$. The receiver first checks validity of σ and π, and if invalid, he then outputs $\perp$. Otherwise, the receiver recovers K by using d_i where $i \in \mathcal{S}_{vk}$. This construction is strikingly the same as the Dolev-Dwork-Naor scheme.

H.2 Cramer-Shoup and Kiltz from Naor-Pinkas

The description of the NP BE scheme is given in Sec. 3.1. A remarkable property of NP BE is that it allows exponentially many number of potential users, i.e. $n = p - 1$, and therefore, we can set $t = 1$ for transforming this scheme into CCA-secure public key encryption since ${}_{p-1}C_1 = p - 1$.

Since NP BE is not verifiable as it is, it is necessary to add a functionality for checking whether (g, g^r, y_0, K) is a DH-tuple or not. Here, we mention two different methods to add this functionality (without using NIZK proofs) with different number theoretic assumptions, and we can have two different CCA-secure KEMs, i.e. (a variant of) Cramer-Shoup [21] and Kiltz [41], according to these two methods. However, we stress that it is difficult to prove the security of these schemes under the CDH assumption. We explain these two schemes in more detail below.

A Variant of Cramer-Shoup from Naor-Pinkas. Here, we first explain a method to add (private) verifiability to NP BE with the same underlying assumption as the original scheme, i.e. the DDH assumption.

For this method, in addition to the keys for the original NP BE scheme, the center also generates $(b_0, b_1, c_0, c_1) \xleftarrow{R} \mathbb{Z}_p^4$, and compute $z = g^{a_0 b_0 + b_1}$ and $w = g^{a_0 c_0 + c_1}$. The public key is $PK = (\mathbb{G}, g, y_0, ..., y_t, z, w)$. The center keeps $f(x)$ and (b_0, b_1, c_0, c_1) as the master key, and user i's decryption key is $(d_i, b_0, b_1, c_0, c_1)$ where $d_i = f(i)$. Assuming that users $i_1, ..., i_t$ are revoked, the sender generates $\psi = (g^r, (g^{f(i_1)})^r, ..., (g^{f(i_t)})^r, z^r)$ and $K = w^r$. On receiving $\psi = (C_0, ..., C_t, D)$, user i recovers $\overline{K} = y_0^r$ as the original scheme, and checks if $D \stackrel{?}{=} \overline{K}^{b_0} \cdot C_0^{b_1}$, and if not, user i outputs $\perp$. Otherwise, he outputs $K = \overline{K}^{c_0} \cdot C_0^{c_1}$ as the session key (this rerandomization is necessary for the security proof).

We next demonstrate a CCA-secure KEM which is derived from the above BE. Extensions **Ext. 1** and **2** in Sec. 8.2 are applied in this transformation.

Setup(1^k)**:** Generate a polynomial $f(x) = a_0 + a_1 x$, and $(b_0, b_1, c_0, c_1) \xleftarrow{R} \mathbb{Z}_p^4$, and compute $y_0 = g^{a_0}$, $y_1 = g^{a_1}$, $z = g^{a_0 b_0 + b_1}$, and $w = g^{a_0 c_0 + c_1}$. The decryption key is $dk = (f(x), b_0, b_1, c_0, c_1)$, and the public key is $PK = (\mathbb{G}, g, y_0, y_1, z, w, \mathsf{TCR})$, where $\mathsf{TCR} : \mathbb{G} \to \mathbb{Z}_p^*$ is a target collision resistant hash function.

Encrypt(PK)**:** Pick a random $r \xleftarrow{R} \mathbb{Z}_p$, and compute $\psi = (g^r, y_0^r y_1^{i \cdot r}, z^r)$ where $i = \mathsf{TCR}(g^r)$, and $K = w^r$. The final output is (ψ, K). (Notice that when viewing it as a ciphertext of BE, this is a ciphertext for all users except for user i.)

Decrypt(dk, ψ, PK)**:** For a ciphertext $\psi = (C_0, C_1, D)$, check whether $(C_0^{f(i)}, C_0^{a_0 b_0 + b_1}) \stackrel{?}{=} (C_1, D)$, where $i = \mathsf{TCR}(C_0)$. If not, output $\perp$. Otherwise, output $K = C_0^{a_0 c_0 + c_1}$.

We notice that the above construction is very similar to Cramer-Shoup (especially, "$y_0^r y_1^{i \cdot r}$" in the ciphertext is the same as a component of Cramer-Shoup's ciphertext) with slight differences, and

CCA-security of it can be proven under the DDH assumption in the same manner as Theorem 5 with similar proof techniques to Cramer-Shoup. Security of the above scheme is discussed in Appendix I in more detail.

In the above, a user can keep $(f(x), a_0b_0 + b_1, a_0c_0 + c_1)$ as his decryption key instead of $(f(x), b_0, b_1, c_0, c_1)$, and this results in shorter size of a decryption key.

Kiltz from Naor-Pinkas. There is yet another method for adding verifiability to NP BE by using bilinear groups $\mathbb{G}_1$ and $\mathbb{G}_2$ where there exists a bilinear mapping $e : \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{G}_2$ such that $e(g^a, g^b) = e(g, g)^{ab}$ for all integer a and b. By replacing the underlying group $\mathbb{G}$ in the original NP BE with $\mathbb{G}_1$ (with order p), one can publicly verify whether (g, g^r, y_0, K) is a DH-tuple or not by checking $e(g, K) \stackrel{?}{=} e(g^r, y_0)$. However, in this modified scheme, the DDH assumption does not hold any more, and it is necessary to introduce another number theoretic assumption, i.e. the Gap Hashed Diffie-Hellman (GHDH) assumption [41]. Then, from this modified NP BE scheme, we can construct a simple CCA-secure KEM as follows:

Setup(1^k)**:** Generate a polynomial $f(x) = a_0 + a_1x$, and compute $y_0 = g^{a_0}$ and $y_1 = g^{a_1}$. The decryption key is $dk = f(x)$, and the public key is $PK = (\mathbb{G}_1, g, y_0, y_1, \mathsf{TCR}, H)$, where $\mathsf{TCR} : \mathbb{G}_1 \to \mathbb{Z}_p^*$ is a target collision resistant hash function.

Encrypt(PK)**:** Pick a random $r \stackrel{R}{\leftarrow} \mathbb{Z}_p$, and compute $\psi = (g^r, y_0^r y_1^{i \cdot r})$ where $i = \mathsf{TCR}(g^r)$, and $K = h(y_0^r)$. The final output is (ψ, K).

Decrypt(dk, ψ, PK)**:** For a ciphertext $\psi = (C_0, C_1)$, check whether $C_0^{f(i)} \stackrel{?}{=} C_1$, where $i = \mathsf{TCR}(C_0)$. If not, output $\perp$. Otherwise, output $K = h(C_0^{a_0})$.

Surprisingly, the above construction is completely the same as Kiltz [41].

H.3 Canetti-Halevi-Katz from Identity-Based Encryption

As already mentioned in Sec. 8.4, our proposed method is an extension of CHK [18]. Namely, an IBE scheme can also be viewed as a BE scheme (with dynamic join) with $n = |\mathcal{ID}|$ and $t = |\mathcal{ID}| - 1$ (i.e. ${}_nC_t = |\mathcal{ID}|$), where $\mathcal{ID}$ is the identity space, and furthermore verifiability always holds since there is only one receiver. Hence, it is possible to apply our construction to IBE if $|\mathcal{ID}| \geq 2^k$, and this construction is identical to the original CHK.

I Security of the Variant of Cramer-Shoup in Appendix H.2

Here, we give an explanation for constructing a DDH adversary B from another adversary A which breaks CCA-security of the proposed CS-variant in Appendix H.2. For a given instance $(g, g_1, g_2, \overline{K^\star}) \in \mathbb{G}_1^4$ of the DDH problem on $\mathbb{G}$, B picks a target collision resistant hash function TCR, and computes $i^\star = \mathsf{TCR}(g_1)$. B also picks randoms $(d, b_0, b_1, c_0, c_1) \in \mathbb{Z}_p^5$ and sets $y_0 = g_2$, $y_1 = (g^d/y_0)^{1/i^\star}$, $z = y_0^{b_0} g^{b_1}$, and $w = y_0^{c_0} g^{c_1}$. The public key $PK = (g, y_0, y_1, z, w, \mathsf{TCR})$ is given to A. For any decryption query (C_0, C_1, D) from A, B computes as

$$i \leftarrow \mathsf{TCR}(C_0), \quad \overline{K} \leftarrow (C_0^d)^{\frac{i}{i - i^\star}} \cdot (C_1)^{\frac{i^\star}{i^\star - i}},$$

and if $D \neq \overline{K}^{b_0} C_0^{b_1}$, returns "$\perp$". Otherwise, B returns $K = \overline{K}^{c_0} C_0^{c_1}$.

It should be noticed that $\mathsf{TCR}(C_0) \neq i^\star$ holds with an overwhelming probability due to the use of target collision resistant hash function TCR.

The challenge ciphertext which is given to A is $((g_1, g_1^d, (\overline{K}^\star)^{b_0} g_1^{b_1}), K_b)$ where $K_0 = (\overline{K}^\star)^{c_0} g_1^{c_1}$ and $K_1 \stackrel{R}{\leftarrow} \mathbb{G}$ for random $b \in \{0,1\}$. B outputs "$(g, g_1, g_2, \overline{K}^\star)$ is a DH-tuple" if (and only if) A's output is identical to b. We notice that if $(g, g_1, g_2, \overline{K}^\star)$ is a DH-tuple, A's view is perfectly the same as the real attack, and therefore, A will correctly guess b with a non-negligible advantage. If it is a random tuple, the distributions $(\overline{K}^\star)^{c_0} g_1^{c_1}$ is indistinguishable from a random element in $\mathbb{G}$ from A's viewpoint, and therefore, A can guess b with only negligible advantage.

J The Proposed CCA-Secure PKE from the BGW BE Scheme

Here, we give the concrete construction of our proposed CCA-secure PKE scheme from the BGW BE scheme.

J.1 Brief Review of Boneh-Gentry-Waters [11]

Let $\mathbb{G}_1$ and $\mathbb{G}_2$ be multiplicative cyclic groups with prime order p, and $e : \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{G}_2$ be a bilinear mapping such that for all $u, v \in \mathbb{G}_1$ and $a, b \in \mathbb{Z}$, we have $e(u^a, v^b) = e(u, v)^{ab}$ and $e(g, g) \neq 1$ where $g \in \mathbb{G}_1$ is a generator of $\mathbb{G}_1$. The BGW BE scheme is constructed as follows:

Setup$(1^k, n)$**:** Let $\mathbb{G}_1$ be a bilinear group with prime order p. Pick a random generator $g \in \mathbb{G}_1$ and random $\alpha \in \mathbb{Z}_p$. Compute $g_i = g^{(\alpha^i)} \in \mathbb{G}_1$ for $i = 1, 2, ..., n, n+2, ..., 2n$. Pick a random $\gamma \in \mathbb{Z}_p$ and set $v = g^\gamma \in \mathbb{G}_1$. The public key is $PK = (g, g_1, ..., g_n, g_{n+2}, ..., g_{2n}, v) \in \mathbb{G}_1^{2n+1}$, and the decryption keys for user $i \in \{1, ..., n\}$ is set as $d_i = g_i^\gamma \in \mathbb{G}_1$. Output $(d_1, ..., d_n, PK)$. Notice that the maximum number of revoked users t $(t < n)$ is arbitrary.

Encrypt$(\mathcal{S}, PK)$**:** Pick a random $r \in \mathbb{Z}_p$, and set $K = e(g_{n+1}, g)^r \in \mathbb{G}_2$. Output (ψ, K) where $\psi = (g^r, (v \cdot \prod_{j \in \mathcal{S}} g_{n+1-j})^r) \in \mathbb{G}_1^2$.

Decrypt$(\mathcal{S}, i, d_i, \psi, PK)$**:** Letting $\psi = (C_0, C_1)$, check whether $e(g, C_1) \stackrel{?}{=} e(v \cdot \prod_{j \in \mathcal{S}} g_{n+1-j}, C_0)$, and if not, output $\perp$. Otherwise, output $K = e(g_i, C_1) / e(d_i \cdot \prod_{j \in \mathcal{S} \setminus \{i\}} g_{n+1-j+i}, C_0)$.

The decryption algorithm is slightly modified from the original BGW to add verifiability. The security of this scheme is addressed as follows:

Proposition 1 ([11])**.** *Let $\mathbb{G}_1$ be bilinear group with prime order p. For any positive integers n and t $(t < n)$, the above scheme is (τ, ϵ, n, t) semantically secure under the decision (τ, ϵ, n) BDHE assumption [8, 11] (see Appendix B.2.3) on $\mathbb{G}_1$.*

We also notice that this BE scheme is unconditionally and publicly verifiable.

J.2 A New CCA-Secure KEM from Boneh-Gentry-Waters

By applying our construction (with extensions **Ext. 1** and **2** in Sec. 8.2) to BGW BE, we have the following CCA-secure KEM:

Setup(1^k)**:** Choose $\ell \in \mathbb{N}$ such that ${}_{2\ell}C_\ell \geq 2^k$. Generate $g, \alpha, g_1, ..., g_{2\ell}, g_{2\ell+2}, ..., g_{4\ell}, v$ as the setup algorithm of Boneh-Gentry-Waters with $n = 2\ell$, and compute $Z = e(g_{2\ell+1}, g)$ where $g_{2\ell+1} = g^{\alpha^{2\ell+1}}$. The decryption key is $dk = g^{\alpha^{2\ell+1}}$, and the public key is $PK = (g, g_1, ..., g_{2\ell}, g_{2\ell+2}, ..., g_{4\ell}, v, Z, \mathsf{TCR})$, where $\mathsf{TCR} : \mathbb{G}_1 \to \mathcal{P}$ is a target collision resistant hash function (see Appendix B.4.2) and $\mathcal{P}$ is the set of all $\mathcal{S} \subseteq \{1, ..., 2\ell\}$ with $|\mathcal{S}| = \ell$.

Encrypt(PK)**:** Pick a random $r \in \mathbb{Z}_p$, and set $K = Z^r \in \mathbb{G}_2$. Compute $\mathcal{S} = \mathsf{TCR}(g^r)$, and output (ψ, K) where $\psi = (g^r, (v \cdot \prod_{j \in \mathcal{S}} g_{2\ell+1-j})^r) \in \mathbb{G}_1^2$.

Decrypt(dk, ψ, PK)**:** Letting $\psi = (C_0, C_1)$, compute $\mathcal{S} = \mathsf{TCR}(C_0)$, and check whether $e(g, C_1) \stackrel{?}{=} e(v \cdot \prod_{j \in \mathcal{S}} g_{2\ell+1-j}, C_0)$, and if not, output $\perp$. Otherwise, output $K = e(dk, C_0)$.

The security of the above scheme is addressed as follows:

Theorem 6. *Let $\mathbb{G}_1$ be bilinear group with prime order p, and TCR be a (τ, ϵ_{tcr}) target collision resistant hash function. Then, the above scheme is $(\tau - o(\tau), \epsilon_{bdhe} + \epsilon_{tcr}, q_D)$ CCA-secure under the decision $(\tau, \epsilon_{bdhe}, 2\ell)$ BDHE assumption on $\mathbb{G}_1$ such that ${}_{2\ell}C_\ell \geq 2^k$.*

This theorem can be proven by a straightforward combination of the proofs of Theorem 5 of this paper and Theorem 3.1 of [11]. Here, we give an intuitive explanation for constructing a BDHE adversary B from another adversary A which breaks CCA-security of the above KEM. For a given instance $(h, g, g^\alpha, ..., g^{(\alpha^{2\ell})}, g^{(\alpha^{2\ell+2})}, ..., g^{(\alpha^{4\ell})}, K^\star) \in \mathbb{G}_1^{4\ell+1} \times \mathbb{G}_2$ of the 2ℓ-BDHE problem on $\mathbb{G}_1$, B picks a target collision resistant hash function TCR, and computes $\mathcal{S}^\star = \mathsf{TCR}(h)$. B also picks a random $u \in \mathbb{Z}_p$ and computes $v = g^u \cdot (\prod_{j \in \mathcal{S}^\star} g_{n+1-j})^{-1}$. The public key $PK = (g, g_1, ..., g_{2\ell}, g_{2\ell+2}, ..., g_{4\ell}, v, \mathsf{TCR})$ is given to A where $g_i = g^{\alpha^i}$ $(1 \leq i \leq 4\ell)$. For any valid decryption query (C_0, C_1) from A, B answers it by using a decryption key $d_i = g_i^u \cdot (\prod_{j \in \mathcal{S}^\star} g_{n+1-j+i})^{-1} (= v^{(\alpha^i)})$ where $i \in \mathsf{TCR}(C_0) \backslash \mathcal{S}^\star$. It should be noticed that $\mathsf{TCR}(C_0) \neq \mathsf{TCR}(h)$ holds with an overwhelming probability due to the use of target collision resistant hash function TCR. The challenge ciphertext which is given to A is $(\psi^\star, K^\star)$ where $\psi^\star = (h, h^u)$. B outputs "$K^\star = e(g_{2\ell+1}, h)$" if (and only if) A's output is 0.

K The Cash-Kiltz-Shoup KEM from the CDH Assumption

Let $\mathbb{G}$ be a multiplicative group with prime order p, and $g \in \mathbb{G}$ be a generator. Then, the construction of the CKS KEM [20] is as follows:

Setup(1^k)**:** Pick $a_{l,j}$ from $\mathbb{Z}_p$, and compute $y_{l,j} = g^{a_{l,j}}$ for $1 \leq l \leq k+1$ and $j = 0, 1$, where k is DEM-key length. The decryption key is $(a_{l,j})_{(1 \leq l \leq k+1,\ j=0,1)}$, and the public key is $PK = (\mathbb{G}, g, (y_{l,j})_{(1 \leq l \leq k+1,\ j=0,1)}, \mathsf{TCR}, h)$, where $\mathsf{TCR} : \mathbb{G} \to \mathbb{Z}_p^*$ is a target collision resistant hash function, and $h : \mathbb{G} \to \{0, 1\}$ is a hardcore bit function for the Diffie-Hellman key in $\mathbb{G}$.

Encrypt(PK)**:** Pick a random $r \stackrel{R}{\leftarrow} \mathbb{Z}_p$, and compute

$$\psi = (g^r, ((y_{l,0}{}^i y_{l,1})^r)_{1 \leq l \leq k+1}), \qquad K = (h(y_{1,0}^r) || h(y_{2,0}^r) || ... || h(y_{k,0}^r))$$

where $i = \mathsf{TCR}(g^r)$. The final output is (ψ, K).

Decrypt(dk, ψ, PK)**:** For a ciphertext $\psi = (C_0, C_1, ..., C_{k+1})$, check whether $C_0^{i \cdot a_{l,0}} C_0^{a_{l,1}} \stackrel{?}{=} C_l$ for all l such that $1 \leq l \leq k+1$, where $i = \mathsf{TCR}(C_0)$. If not, output $\perp$. Otherwise, output $K = (h(C_0^{a_{1,0}})||h(C_0^{a_{2,0}})||...||h(C_0^{a_{k,0}}))$.

The above KEM is CCA-secure under the CDH assumption, and furthermore, assuming that it is possible to extract $\log k$ hardcore bits from a single DH key, sizes for ciphertexts and keys can be compressed by a factor of approximately $1/\log k$ (as mentioned in [20]). The above construction is a natural extension of another scheme with the HDH assumption in the same paper [20], and basically is constructed by k (or $k/\log k$) copies of their HDH-based scheme. Since a straightforward use of k independent encryption with a ciphertext overhead of three group elements results in a ciphertext overhead with $3k$ group elements in total, the authors also give an improvement for reducing it. Namely, in their CDH-based scheme (as described above), two components of one ciphertext is commonly used by all other encryption operations, and the ciphetext length of the resulting scheme becomes $k+2$ (or $k/\log k + 2$) group elements. Its security can be proved by a hybrid argument.

L A Generic Construction of CCA-Secure Broadcast Encryption

Here, we give a generic construction of CCA-secure BE schemes from any CPA-secure one with public verifiability. We also apply this to the BGW BE scheme, and demonstrate a new CCA-secure BE scheme with verifiability whose computational cost is slightly better than the previous scheme [11].

L.1 The Construction

Given a BE scheme $\Pi' = (\mathbf{Setup}', \mathbf{Encrypt}', \mathbf{Decrypt}')$ which is CPA-secure against selective adversaries and verifiable, we construct a CCA-secure BE scheme $\Pi = (\mathbf{Setup}, \mathbf{Encrypt}, \mathbf{Decrypt})$. Similarly to the construction in Sec. 8.1, we use a strong-one time signature scheme Σ with the same notation. We assume that the maximum number of potential users in Π' is $n + \delta_n$, and a sender can revoke $t + \delta_t$ users where there exists an injective mapping (or a target collision resistant hash function) $\mathsf{INJ} : \{0,1\}^k \to \mathcal{P}$ and $\mathcal{P}$ is the set of all $\Delta\mathcal{S} \subseteq \{n+1, ..., n+\delta_n\}$ with $|\Delta\mathcal{S}| = \delta_n - \delta_t$. The construction of Π is as follows:

Setup$(1^k, n, t)$**:** Choose δ_n and δ_t (where $(n + \delta_n, t + \delta_t)$ is a possible parameter choice for Π') such that ${}_{\delta_n}C_{\delta_t} \geq 2^k$. Run $\mathbf{Setup}'(1^k, n + \delta_n, t + \delta_t)$ to obtain $(d_1, ..., d_{n+\delta_n}, PK)$, and pick an injective mapping $\mathsf{INJ} : \{0,1\}^k \to \mathcal{P}$. The decryption key is $dk = (d_1, ..., d_n)$ and the public key is $\overline{PK} = (PK, \mathsf{INJ})$.

Encrypt$(\mathcal{S}, \overline{PK})$**:** Run $\mathbf{Gen}(1^k)$ to obtain verification key vk and signing key sk (with $|vk| = k$), and compute $\Delta\mathcal{S}_{vk} = \mathsf{INJ}(vk)$, $(\psi, K) \leftarrow \mathbf{Encrypt}'(\mathcal{S} \cup \Delta\mathcal{S}_{vk}, PK)$ and $\sigma \leftarrow \mathbf{Sign}(sk, \psi)$. The final output is $((\psi, vk, \sigma), K)$.

Decrypt$(\mathcal{S}, i, d_i, \psi, \overline{PK})$**:** For a ciphertext (ψ, vk, σ), check whether $\mathbf{Verify}(vk, \psi, \sigma) \stackrel{?}{=} 1$. If not, output $\perp$. Otherwise, compute $\Delta\mathcal{S}_{vk} = \mathsf{INJ}(vk)$ and output $K \leftarrow \mathbf{Decrypt}'(\mathcal{S} \cup \Delta\mathcal{S}_{vk}, i, d_i, \psi, PK)$.

The security of the above construction is addressed as follows.

Theorem 7. *If Π' is a $(\tau, \epsilon_{cpa}, n+\delta_n, t+\delta_t)$ semantically secure and $(\tau, \epsilon_{vfy}, n+\delta_n, t+\delta_t)$ publicly verifiable broadcast encryption scheme such that ${}_{\delta_n}C_{\delta_t} \geq 2^k$, and Σ is a (τ, ϵ_{uf}) strongly unforgeable one-time signature scheme, then Π is $(\tau - o(\tau), \epsilon_{cpa} + \epsilon_{vfy} + \frac{1}{2}\epsilon_{uf}, n, t, q_D)$ CCA-secure broadcast encryption scheme.*

The proof of the theorem is similar to that of Theorem 5, and omitted here. Extension **Ext. 1** in Sec. 8.2 is also applicable to the above construction.

L.2 An Instantiation from Boneh-Gentry-Waters

We give a concrete construction (with **Ext. 1**) of a CCA-secure BE scheme by using CPA-secure BGW BE as the underlying scheme. Notations are the same as Appendix J.1.

Setup$(1^k, n)$**:** Choose $\ell \in \mathbb{N}$ such that ${}_{2\ell}C_\ell \geq 2^k$. Let $\mathbb{G}_1$ be a bilinear group with prime order p. Pick a random generator $g \in \mathbb{G}_1$ and random $\alpha \in \mathbb{Z}_p$. Compute $g_i = g^{(\alpha^i)} \in \mathbb{G}_1$ for $i = 1, 2, ..., n+2\ell, n+2\ell+2, ..., 2(n+2\ell)$. Pick a target collision resistant hash function $\mathsf{TCR} : \mathbb{G}_1 \to \mathcal{P}$, where $\mathcal{P}$ is the set of all $\Delta\mathcal{S} \subseteq \{n+1, ..., n+2\ell\}$ with $|\Delta\mathcal{S}| = \ell$. Pick a random $\gamma \in \mathbb{Z}_p$ and set $v = g^\gamma \in \mathbb{G}_1$. Set $Z = e(g_{n+2\ell+1}, g)$ where $g_{n+2\ell+1} = g^{\alpha^{n+2\ell+1}}$. The public key is $PK = (g, g_1, ..., g_{n+2\ell}, g_{n+2\ell+2}, ..., g_{2(n+2\ell)}, v, Z, \mathsf{TCR})$, and the decryption keys for user $i \in \{1, ..., n\}$ is set as $d_i = g_i^\gamma \in \mathbb{G}_1$. Output $(d_1, ..., d_n, PK)$.

Encrypt$(\mathcal{S}, PK)$**:** Pick a random $r \in \mathbb{Z}_p$, and set $K = Z^r \in \mathbb{G}_2$. Compute $\Delta\mathcal{S} = \mathsf{TCR}(g^r)$, and output (ψ, K) where $\psi = (g^r, (v \cdot \prod_{j \in \mathcal{S} \cup \Delta\mathcal{S}} g_{n+2\ell+1-j})^r) \in \mathbb{G}_1^2$.

Decrypt$(\mathcal{S}, i, d_i, \psi, PK)$**:** Letting $\psi = (C_0, C_1)$, compute $\Delta\mathcal{S} = f(C_0)$, and check whether $e(g, C_1) \stackrel{?}{=} e(v \cdot \prod_{j \in \mathcal{S} \cup \Delta\mathcal{S}} g_{n+2\ell+1-j}, C_0)$, and if not, output $\perp$. Otherwise, output $K = e(g_i, C_1)/e(d_i \cdot \prod_{j \in \mathcal{S} \cup \Delta\mathcal{S} \setminus \{i\}} g_{n+2\ell+1-j+i}, C_0)$.

The security of the above scheme is addressed as follows:

Theorem 8. *Let $\mathbb{G}_1$ be bilinear group with prime order p, and TCR be a (τ, ϵ_{tcr}) target collision resistant hash function. Then, for any positive integers n and t $(t < n)$, the above scheme is $(\tau - o(\tau), \epsilon_{bdhe} + \epsilon_{tcr}, n, t, q_D)$ CCA-secure under the $(\tau, \epsilon_{bdhe}, n+2\ell)$ decision BDHE assumption on $\mathbb{G}_1$ such that ${}_{2\ell}C_\ell \geq 2^k$.*

The proof of the theorem is similar to that of Theorem 5, and omitted here.

L.3 Comparison

Table 3 summarizes an efficiency comparison among variants of BGW BE [11]. For concreteness, we can set $\ell = 67$ for 128-bit security, and therefore, can assume that $n \geq |\mathcal{S}| \gg \ell$ holds for many typical applications. We note that it is possible to add verifiability to both CPA-secure BGW (CPA-BGW) and its CCA-secure version (CCA-BGW)[6] by checking the validity of the ciphertext (see Appendix J.1), and efficiency of their verifiable schemes is also mentioned in Table 3. We notice that computational costs for our proposed scheme is slightly better than CCA-BGW with

[6] In the comparison, we apply BMW-like technique [15] to enhance the efficiency of the original CCA-secure BGW which is presented in [11].

Table 3: Efficiency comparison for variants of the Boneh-Gentry-Waters broadcast encryption schemes, assuming that the maximum number of potential users is n. Notations are basically the same as Table 1.

	Security Assumption	Ciphertext Overhead	Encryption	Decryption	Key size (pk/dk)	Verifiability	Security Level
			#pairings + #[multi,regular]-exp + #ops				
CPA-BGW [11]	n-BDHE	$2\|g\|$	$0 + [0,3] + \|\mathcal{S}\|$	$2 + [0,0] + \|\mathcal{S}\|$	$2n + 1/1$	–	CPA
	↓	↓	↓	$4 + [0,0] + 2\|\mathcal{S}\|$	↓	√	CPA
CCA-BGW [11]	$n+1$-BDHE	$2\|g\|$	$0 + [1,2] + \|\mathcal{S}\|$	$2 + [1,2] + 2\|\mathcal{S}\|$	$2n + 3/1$	–	CCA
	↓	↓	↓	$4 + [1,1] + 2\|\mathcal{S}\|$	↓	√	CCA
Ours	$n + 2\ell$-BDHE	$2\|g\|$	$0 + [0,3] + (\|\mathcal{S}\| + \ell)$	$4 + [0,0] + 2(\|\mathcal{S}\| + \ell)$	$2n + 4\ell + 1/1$	√	CCA

verifiability, and comparable to that for other less functional schemes, i.e. CPA-secure and/or non-verifiable schemes. The sizes of the ciphertext and the decryption key of our proposed scheme are the same as other schemes.