Practical Dual-Receiver Encryption

Soundness, Complete Non-Malleability, and Applications

Sherman S.M. Chow[1], Matthew Franklin[2], and Haibin Zhang[2]

[1] Department of Information Engineering, Chinese University of Hong Kong
sherman@ie.cuhk.edu.hk
[2] Department of Computer Science, University of California Davis
{franklin,hbzhang}@cs.ucdavis.edu

Abstract. We reformalize and recast dual-receiver encryption (DRE) proposed in CCS '04, a public-key encryption (PKE) scheme for encrypting to two *independent* recipients in one shot. We start by defining the crucial *soundness* property for DRE, which ensures that two recipients will get the same decryption result. While conceptually simple, DRE with soundness turns out to be a powerful primitive for various goals for PKE, such as complete non-malleability (CNM) and plaintext-awareness (PA). We then construct *practical* DRE schemes without random oracles under the Bilinear Decisional Diffie-Hellman assumption, while prior approaches rely on random oracles or inefficient non-interactive zero-knowledge proofs. Finally, we investigate further applications or extensions of DRE, including DRE with CNM, combined use of DRE and PKE, strengthening two types of PKE schemes with plaintext equality test, off-the-record messaging with a stronger notion of deniability, etc.

Key words: Dual receiver encryption, soundness, complete non-malleability, plaintext-awareness, combined encryption, public plaintext equality test, off-the-record messaging.

1 Introduction

Dual-receiver encryption (DRE), introduced by Diament, Lee, Keromytis, and Yung [17] (DLKY), is a special kind of public-key encryption (PKE) which allows a ciphertext to be decrypted into the same plaintext by two *independent* users. More concretely, the DRE encryption algorithm produces a ciphertext by taking as input a message and two independently generated public keys pk_1 and pk_2. Both receivers (owners of pk_1 and pk_2) will get the same decryption result.

DRE is a handy tool when sensitive information (may it be political, financial, or medical) should be backed up, and potentially decryptable by some other party (or a threshold number of designated parties which further requires the DRE to support threshold decryption).

On the other hand, while it appears to be conceptually simple, DRE turns out to be a valuable tool in many cryptographic applications. For example, DLKY show how to construct security puzzles for rate-limiting remote users, e.g., in the TLS protocol [17]. Dodis, Katz, Smith, and Walfish describe the use of DRE to address the deniable authentication problem [18].

Soundness. A crucial requirement for the above applications is that "the ciphertext will be decrypted to the same message by either private key." Unfortunately, the original formulation due to DLKY only ensures the correctness property for honestly generated ciphertexts. As our first contribution, we strengthen the definition by introducing new soundness security notions which formalize the intuition that "two receivers will get the same plaintext and they do *know* this fact."[3] The importance of soundness can be seen when we discuss our second contribution on various applications or extensions of DRE with soundness[4].

Complete Non-Malleability. Complete non-malleability (CNM) [19, 42] prohibits adversaries from computing encryptions of related plaintexts even under adversarially generated public keys.

[3] Yet, we can show that the DLKY construction (in the ROM) satisfies our soundness.

[4] When there is no ambiguity from context, we omit "with soundness".

This notion is useful both in theory and practice. One can transform our DRE scheme which is secure against chosen-ciphertext attacks (CCA) to a CNM-PKE scheme in the common reference string (CRS) model. Namely, given a DRE scheme, one of the public keys of DRE is added to the CRS, whereas the other serves as the public key of the new scheme; encryption algorithm remains the same as the one for DRE for either public key, while the decryption algorithm is simply the one for DRE decryption scheme with respect to the secret key of the other receiver.

We also study CNM-DRE which remains secure for dynamically generated public keys. It also leads to *dual receiver non-malleable commitment*, a new primitive of independent interests.

Plaintext-Awareness. Roughly, *plaintext-awareness* captures the property that an adversary can decrypt any ciphertext it creates. Assuming *key registration*, Herzog, Liskov, and Micali [25] build a plaintext-aware PKE from general zero-knowledge proof of knowledge and non-malleable non-interactive zero-knowledge (NIZK) proof, which is rather inefficient. We show that one can simply use our DRE schemes which leads to efficient registration-based plaintext-aware PKE.

More Applications. Some (new) applications of DRE will be outlined in Section 7, which include PKE with plaintext equality test (PET), PKE with non-interactive opening, off-the-record messaging protocol with stronger deniability, and useful security puzzle without random oracles.

On Constructing DREs. In this paper, we propose an efficient construction of DRE, a useful primitive that helps achieving various goals as we described. Indeed, the known DRE constructions in the literature are either in the random oracle model (ROM), or rely on CRS to realize the idea of Naor-Yung two-key encryption [32]. The DLKY-DRE scheme [17] uses REACT transformation [34] to achieve CCA-security under the Gap-Bilinear Diffie-Hellman (GBDH) assumption [33].[5] Building DRE using (the most efficient instantiation of) Groth-Sahai proof system [23] will take nearly one hundred group elements [41].[6] From another perspective, given the difficulties, Zhang, Hanaoka, and Imai [47] rely on identity-based encryption [5] to solve the problems that DRE would (in constructing useful security puzzles [17, 47]). All these are suggesting that constructing an efficient DRE without random oracles is *non-trivial.*

Broadcast Encryption. While encryption schemes for multiple recipients exist such as broadcast encryption, the group manager needs to prepare the decryption keys for users and can thus decrypt any ciphertexts. The key generation may also be stateful, and the decryption algorithm may also create different intermediate variables for different users. On the other hand, it is natural for DRE to satisfy *independence* of receivers (except, they may share the same security parameter and cryptographic group, and of course, they can be certified by a trusted party). In general, broadcast encryption is more expensive than dual-receiver encryption.

Properties for DRE. In light of these discussion, a *good* DRE should satisfy:

Security under standard assumptions, yet with practical efficiency. As other primitives, a DRE preferably should avoid the use of the less-studied cryptographic assumptions, and its security proof should avoid the use of the ROM. At the same time, it should be efficient so it can be used directly in practice, or as a building block, without introducing much overhead.

Symmetry. Naturally, the role of two receivers should be "symmetric" with respect to all DRE algorithms. This means that the same key generation algorithm will be executed by any user, and the resulting key can be used as the "first" receiver or the "second" receiver, up to the wish of the encryptor. Otherwise, if a DRE user is required to use keys in two different formats for different "positions" in the receivers list, that means each user should generate both kinds of keys, and two implementations (either as software or as circuit) for the same functionality

[5] In Appendix B, we also present a more efficient, redundancy-free DRE in the ROM.

[6] The scheme was first studied by Smith and Youn in an unpublished manuscript [41] and we review it in Appendix C.

(e.g., decryption) are required. It is also somewhat counter-intuitive to have different decryption algorithms when they can take the CRS as an input.

Symmetry is also useful for applications when the message sender takes the role as one of the receivers of the DRE as well. Section 6 will discuss registration-based plaintext-aware encryption from DRE which benefits from this property.

Public verifiability. Verifying the validity of a ciphertext might be done without decrypting. If a scheme satisfies this requirement we call it publicly verifiable. It is one of the most common cryptographic tasks to prove that two ciphertexts (or commitments) are well-formed and encrypting (or committing to) the same plaintext. In particular, it is useful to achieve threshold decryption [10].

Our Proposed Construction. We provide a practical solution of DRE based on the well-known Bilinear Decisional Diffie-Hellman (BDDH) assumption without random oracles. By analogy with the well-known notions of key encapsulation mechanism (KEM) and hybrid encryption, we also introduce the notions of dual-receiver KEM (DKEM) and hybrid DRE. It is followed by an efficient construction of DKEM secure under the BDDH assumption. Both of our DRE and DKEM constructions are symmetric, publicly verifiable, and competitive with the most efficient PKE schemes. Also, both can be easily extended to support threshold decryption, which is desirable for the backup application. Both constructions require a trusted setup to acquire a common bilinear group, but all receivers can create their own secret keys, in contrast to the broadcast encryption approach where the users are either assigned with a secret keys or they need to interact with each other before deriving their own secret keys.

Combining DRE and PKE without Key Separation. DRE is of limited use *per se.* For conventional usages, one expects a *combined encryption scheme* which can *securely* provide the functionalities of both DRE and regular PKE simultaneously. This enables users to employ the same key to achieve both functions, and minimizes the risk of misuse and the times of registration with the trusted party. Of course, the combination makes sense only if the schemes retain their efficiency. We first define the security requirements formally, and then give a construction without random oracles from the BDDH assumption which is nearly as efficient as the DRE scheme proposed.

DRE with Complete Non-Malleability. It is proven that non-interactive CNM-PKE does not exist with simulation-based black-box simulation in the standard model [19]. A similar impossibility result applies to DRE. We thus instead explore how to design CNM-DRE in the CRS model, just like the study of CNM-PKE in the literature [42, 30]. This not only provides a stronger notion for DRE but also for two kinds of PETs (as illustrated later), which apply to settings where on-line authorities are available and adversaries might dynamically and maliciously generate public keys. However, it does not seem an easy task to build CNM-DRE either. Intuitively, this new primitive requires three trapdoors, two of which must be symmetric. We provide two different paradigms in the CRS model. One is to combine Naor-Yung [32] and Rackoff-Simon [37] paradigms, while the other relies on lossy trapdoor functions [36]. Both of the general paradigms can give reasonably efficient instantiations based on a number of assumptions.

Additional Contributions. Besides the contributions mentioned above (i.e., connections among DRE with soundness and other primitives, formalization of definitions, efficient constructions from standard assumptions, and novel applications), we stress that our paper makes a number of (other) definitional contributions, including a refined syntax of DRE and a simplified CCA-security for it, dual receiver KEM (DKEM) and its CCA-security, combined encryption scheme and its CCA-security, and complete non-malleability (CNM) for DRE.

Further Related Work. We define the soundness property of DRE by strengthening consistency condition via an experiment involving an adversary. The approach echoes Bellare *et al.* [1], which likewise formulates consistency more as a security condition. On the other hand, securely combining PKE and PKES (public-key encryption with keyword search) has been studied [4, 48]. Rogaway [38] studied securely combining CPA encryption and CCA encryption in the context of length-preserving ciphers. Finally, a related concept of ad hoc broadcast encryption was proposed by Wu, Qin, Zhang, and Domingo-Ferrer [44], in which anyone can encrypt messages to a group of ad hoc and independent receivers without central trusted dealer. However, their construction is less efficient and only considers IND-CPA security.

Outline. The rest of the paper is organized as follows. Section 2 refines the security of DRE. Section 3 presents a practical DRE followed a practical DKEM. Section 4 studies how to securely combine PKE and DRE. Section 5 models the CNM notion for DRE and then provides two general paradigms both followed by efficient instantiations. Section 6 explores how to use DRE to construct registration-based plaintext-aware encryption, and Section 7 goes on to illustrate more applications of DRE. Preliminaries, more DRE schemes in the ROM and CRS model, and proofs of theorems can be found in the Appendix.

2 Refining the Security Model of DRE

All the definitions and security experiments to be described are in the common reference string (CRS) model, where there is a trusted CRS generation algorithm that takes as input the security parameter, and outputs a CRS, which will be part of the inputs of the other algorithms. However, they can be easily adopted for the standard model where the CRS is simply the common security parameter.

Public-Key Dual Receiver Encryption. A public-key *dual receiver encryption* scheme $\mathcal{DRE} = (\mathsf{CGen}_{\mathrm{DRE}}, \mathsf{Gen}_{\mathrm{DRE}}, \mathsf{Enc}_{\mathrm{DRE}}, \mathsf{Dec}_{\mathrm{DRE}})$ consists of algorithms:

- $\mathsf{CGen}_{\mathrm{DRE}}(1^k)$: The randomized *CRS generation* algorithm takes as input a security parameter k and outputs a CRS crs; we write $\mathsf{crs} \overset{\$}{\leftarrow} \mathsf{CGen}_{\mathrm{DRE}}(1^k)$.
- $\mathsf{Gen}_{\mathrm{DRE}}(\mathsf{crs})$: The randomized *key generation* algorithm takes as input crs and outputs a public/secret key pair (pk, sk); we write (pk_1, sk_1) and (pk_2, sk_2) for the key pairs of two independent users. Without loss of generality, for the rest of the paper, we assume $pk_1 <^d pk_2$, where $<^d$ is a "less-than" operator based on lexicographic order.
- $\mathsf{Enc}_{\mathrm{DRE}}(\mathsf{crs}, pk_1, pk_2, M)$: The randomized *encryption algorithm* takes as input crs, two public keys pk_1 and pk_2 (such that $pk_1 <^d pk_2$) and message M, and outputs a ciphertext C; we write $C \overset{\$}{\leftarrow} \mathsf{Enc}_{\mathrm{DRE}}(\mathsf{crs}, pk_1, pk_2, M)$.
- $\mathsf{Dec}_{\mathrm{DRE}}(\mathsf{crs}, pk_1, pk_2, sk_i, C)$: The deterministic *decryption algorithm* takes two public keys pk_1 and pk_2 ($pk_1 <^d pk_2$), one of the secret keys sk_i ($i \in \{0, 1\}$), and a ciphertext C as input, and outputs a message M_i (which may be the special symbol $\perp$); we write $M_i \leftarrow \mathsf{Dec}_{\mathrm{DRE}}(\mathsf{crs}, pk_1, pk_2, sk_i, C)$. We may simply write $M_i \leftarrow \mathsf{Dec}_{\mathrm{DRE}}(sk_i, C)$ when there is no ambiguity.

For consistency, we require that, if $\mathsf{crs} \overset{\$}{\leftarrow} \mathsf{CGen}_{\mathrm{DRE}}(1^k)$, $(pk_1, sk_1) \overset{\$}{\leftarrow} \mathsf{Gen}_{\mathrm{DRE}}(\mathsf{crs})$ and $(pk_2, sk_2) \overset{\$}{\leftarrow} \mathsf{Gen}_{\mathrm{DRE}}(\mathsf{crs})$ where $pk_1 <^d pk_2$, and $C \overset{\$}{\leftarrow} \mathsf{Enc}_{\mathrm{DRE}}(\mathsf{crs}, pk_1, pk_2, M)$, then we have the probability $\Pr[\mathsf{Dec}_{\mathrm{DRE}}(\mathsf{crs}, pk_1, pk_2, sk_1, C) = \mathsf{Dec}_{\mathrm{DRE}}(\mathsf{crs}, pk_1, pk_2, sk_2, C) = M] = 1$ for all integers k and messages M, where the probability is taken over the coins of all the algorithms above. We omit the inclusion of crs when context is clear. Our syntax is slightly different from the initially proposed one [17] for the sake of clarity. We explicitly regard DRE encryption and decryption algorithms as functions of the public keys of two *independent* receivers. As far as we are concerned, the notational changes better capture the spirits of DRE, as discussed in the introduction.

Extending the DLKY Notion—Soundness. In DLKY [17], only the *basic* correctness property is taken into account, which ensures that if the sender honestly follows the protocol then the two receivers will get the same plaintext. However, it is fairly weak or even problematic since there exist solutions satisfying the basic correctness requirement yet failing to provide the functionality of DRE required by its applications. For instance, one can pick a conventional PKE scheme to encrypt the same message using two independent users' public keys as a potential solution of DRE with correctness for a honest sender, but a cheating sender can simply encrypt different messages.

We thus need to formalize the intuition of this rather basic property that any adversary cannot "cheat" by creating a ciphertext which can be decrypted to two different plaintexts. It is also not allowed that one party decrypts it to a message m, while the other decrypts it to $\perp$, i.e., it is a valid ciphertext for one but invalid for another. Besides, there is an additional goal of DRE — both receivers "know" that the ciphertext can be decrypted to the same result. Formally, we consider the following experiment that is associated to an adversary $\mathcal{A}$:

```
Experiment $\mathbf{Exp}^{\mathrm{sound}}_{\mathcal{DRE},\mathcal{A}}(k)$
 $\mathsf{crs} \xleftarrow{\$} \mathsf{CGen}_{\mathrm{DRE}}(1^k)$
 $(pk_1, sk_1) \xleftarrow{\$} \mathsf{Gen}_{\mathrm{DRE}}(\mathsf{crs}); (pk_2, sk_2) \xleftarrow{\$} \mathsf{Gen}_{\mathrm{DRE}}(\mathsf{crs})\ (pk_1 <^d pk_2)$
 $C \xleftarrow{\$} \mathcal{A}(\mathsf{crs}, pk_1, sk_1, pk_2, sk_2)$
 if $\mathsf{Dec}_{\mathrm{DRE}}(sk_1, C) \neq \mathsf{Dec}_{\mathrm{DRE}}(sk_2, C)$ then
 return 1 else return 0
```

We define the advantage of $\mathcal{A}$ in the above experiment as

$$\mathbf{Adv}^{\mathrm{sound}}_{\mathcal{DRE},\mathcal{A}}(k) = \Pr[\mathbf{Exp}^{\mathrm{sound}}_{\mathcal{DRE},\mathcal{A}}(k) = 1].$$

$\mathcal{DRE}$ satisfies *soundness*, if for any adversary $\mathcal{A}$, we have that $\mathbf{Adv}^{\mathrm{sound}}_{\mathcal{DRE},\mathcal{A}}(k)$ is negligible in the security parameter k, where the probability is taken over the choice of $\mathsf{crs} \xleftarrow{\$} \mathsf{CGen}_{\mathrm{DRE}}(1^k)$, $(pk_1, sk_1) \xleftarrow{\$} \mathsf{Gen}_{\mathrm{DRE}}(\mathsf{crs})$, $(pk_2, sk_2) \xleftarrow{\$} \mathsf{Gen}_{\mathrm{DRE}}(\mathsf{crs})$, and coins of $\mathcal{A}$. The adversary can be either computationally bounded or unbounded. If the advantage is always equal to 0, we say that $\mathcal{DRE}$ has *perfect soundness.*

DLKY-DRE is Perfectly Sound. Though DLKY does not formalize any soundness notion, we show that the CCA-secure DRE [17] remains a sound and non-trivial DRE which underscores their wisdom in designing DRE.

We start by rephrasing the DLKY-DRE as follows. It builds on a symmetric bilinear group $\mathcal{BG} = (q, \mathbb{G}, \mathbb{G}_T, e, g)$ where $\mathbb{G}$ and $\mathbb{G}_T$ are cyclic groups of prime order q, g generates $\mathbb{G}$, and $e\colon \mathbb{G} \times \mathbb{G} \to \mathbb{G}_T$ is an efficiently computable bilinear map. The idea is quite easy to understand. The BDH assumption implies a one-way function with "double" trapdoors, and the BDDH assumption implies an IND-CPA secure DRE. They use a general conversion REACT [34] to achieve IND-CCA-security. The system fixes three hash functions (all modeled as random oracles) $\mathsf{H}\colon \mathbb{G}_T \to \{0,1\}^n$, $\mathsf{G}\colon \{0,1\}^n \to \{0,1\}^n$, and $\mathsf{F}\colon \{0,1\}^{4n+|e(g,g)|} \to \{0,1\}^{n'}$. One first selects two receivers having public keys (g, X) and (g, Y) where $X = g^x$ and $Y = g^y$ with corresponding secret keys x and y, respectively. To encrypt a message $M \in \{0,1\}^n$, sender selects $r \xleftarrow{\$} \mathbb{Z}_q$ and $\rho \in \{0,1\}^n$ and computes

$$u_1 \leftarrow g^r, \quad u_2 \leftarrow \rho \oplus \mathsf{H}(e(X,Y)^r), \quad u_3 \leftarrow M \oplus \mathsf{G}(\rho), \quad u_4 \leftarrow \mathsf{F}(\rho, M, u_2, u_3, e(X,Y)^r).$$

The ciphertext is $C = (u_1, u_2, u_3, u_4)$. To decrypt, one having x computes $\rho \leftarrow u_2 \oplus \mathsf{H}(e(u_1, Y)^x)$, and $M \leftarrow \mathsf{G}(\rho) \oplus u_3$. Then, she checks whether $u_4 = \mathsf{F}(\rho, M, u_2, u_3, e(u_1, Y)^x)$. If this is the case then she outputs M. Likewise, the other who owns y can decrypt C by first computing $\rho' \leftarrow u_2 \oplus \mathsf{H}(e(u_1, X)^y)$ and $M' \leftarrow \mathsf{G}(\rho') \oplus u_3$ and then checking if $u_4 = \mathsf{F}(\rho', M', u_2, u_3, e(u_1, X)^y)$. This completes the description of the DLKY construction.

Given any ciphertext $(u_1, u_2, u_3, u_4, X, Y)$ where $X = g^x$ and $Y = g^y$ are the public keys of two recipients, we show that either the two recipients both output the same message, or they output $\perp$. Assume that one who owns x can decrypt it to a message M. This means that the intermediate value ρ and M must satisfy $u_4 = \mathsf{F}(\rho, M, u_2, u_3, e(u_1, Y)^x)$. It remains to show that the other who has y should also decrypt to the same M. Indeed, the second receiver will get the same $\rho' = \rho$ since $e(u_1, Y)^x = e(u_1, X)^y$. It is thus clear that last verification equation must satisfy and $M = M'$. It is also clear that this argument remains valid if the first user outputs the special symbol $\perp$. Thus, the DLKY construction is perfectly sound.

Weakening/Strengthening Soundness Notions. The above soundness notion allows the adversary to know the secret keys of targeted receivers. One could weaken it by providing the adversary with the full decryption oracles instead of secret keys. Given two honestly generated public keys pk_1 and pk_2, we define the *weak-soundness* advantage $\mathbf{Adv}^{\text{w-sound}}_{\mathcal{DRE},\mathcal{A}}(k)$ of adversary $\mathcal{A}$ as

$$\Pr[\mathsf{crs} \xleftarrow{\$} \mathsf{CGen}_{\mathrm{DRE}}(1^k); C \xleftarrow{\$} \mathcal{A}^{\mathsf{Dec}_{\mathrm{DRE}}(sk_1,\cdot),\mathsf{Dec}_{\mathrm{DRE}}(sk_2,\cdot)}(\mathsf{crs}, pk_1, pk_2) \\ :\ \mathsf{Dec}_{\mathrm{DRE}}(sk_1, C) \neq \mathsf{Dec}_{\mathrm{DRE}}(sk_2, C)].$$

$\mathcal{DRE}$ satisfies *weak soundness*, if for any adversary $\mathcal{A}$, $\mathbf{Adv}^{\text{w-sound}}_{\mathcal{DRE},\mathcal{A}}(k)$ is negligible in the security parameter k, where the probability is taken over the random choices of $\mathsf{crs} \xleftarrow{\$} \mathsf{CGen}_{\mathrm{DRE}}(1^k)$, $(pk_1, sk_1) \xleftarrow{\$} \mathsf{Gen}_{\mathrm{DRE}}(\mathsf{crs})$, $(pk_2, sk_2) \xleftarrow{\$} \mathsf{Gen}_{\mathrm{DRE}}(\mathsf{crs})$, and $\mathcal{A}$.

On the other hand, we can give a stronger soundness notion by allowing the adversary to adversarially choose public keys where it does not even know the corresponding secret keys. We have to be a little careful here. Some encryption scheme might support valid-looking keys such that the adversary might produce ciphertexts that can be decrypted in different ways. We call an encryption scheme *admissible* if there is an efficient public verification algorithm such that any valid public key pk that passes the verification algorithm must only correspond to one unique secret key sk. For instance, the basic ElGamal encryption scheme is admissible. In the context of DRE, we *only* consider admissible encryption schemes. If $\mathcal{DRE}$ is an admissible dual receiver encryption scheme and $\mathcal{A}$ is an adversary, we define the *strong-soundness* advantage $\mathbf{Adv}^{\text{s-sound}}_{\mathcal{DRE},\mathcal{A}}(k)$ of $\mathcal{A}$ as

$$\Pr[\mathsf{crs} \xleftarrow{\$} \mathsf{CGen}_{\mathrm{DRE}}(1^k); (C, pk_1, pk_2) \xleftarrow{\$} \mathcal{A}(\mathsf{crs}) :\ \mathsf{Dec}_{\mathrm{DRE}}(sk_1, C) \neq \mathsf{Dec}_{\mathrm{DRE}}(sk_2, C)],$$

where, above, sk_1 and sk_2 are the unique secret keys of pk_1 and pk_2, respectively, and both the public and secret keys can be chosen by the adversary. $\mathcal{DRE}$ satisfies the *strong soundness* property if for any adversary $\mathcal{A}$, we have that $\mathbf{Adv}^{\text{s-sound}}_{\mathcal{DRE},\mathcal{A}}(k)$ is negligible in the security parameter k, where the probability is taken over coins of $\mathcal{A}$. Jumping ahead, we stress that the above notion is useful when speaking of completely non-malleable DRE (CNM-DRE).

Security of DRE against Chosen-Ciphertext Attacks. DRE's soundness makes one of the two decryption oracles redundant. To simplify the experiment modelling CCA-security without loss of generality, we assume that the adversary is only given the decryption oracle of the first receiver.

```
Experiment $\mathbf{Exp}^{\mathrm{cca}}_{\mathcal{DRE},\mathcal{A}}(k)$
  $\mathsf{crs} \xleftarrow{\$} \mathsf{CGen}_{\mathrm{DRE}}(1^k)$
  $(pk_1, sk_1) \xleftarrow{\$} \mathsf{Gen}_{\mathrm{DRE}}(\mathsf{crs}); (pk_2, sk_2) \xleftarrow{\$} \mathsf{Gen}_{\mathrm{DRE}}(\mathsf{crs})\ (pk_1 <^d pk_2)$
  $(M_0, M_1, \mathsf{s}) \xleftarrow{\$} \mathcal{A}^{\mathsf{Dec}_{\mathrm{DRE}}(sk_1,\cdot)}(\mathsf{find}, \mathsf{crs}, pk_1, pk_2)$
  $b \xleftarrow{\$} \{0,1\}; C^* \xleftarrow{\$} \mathsf{Enc}_{\mathrm{DRE}}(\mathsf{crs}, pk_1, pk_2, M_b)$
  $b' \xleftarrow{\$} \mathcal{A}^{\mathsf{Dec}_{\mathrm{DRE}}(sk_1,\cdot)}(\mathsf{guess}, C^*, \mathsf{s})$
  if $b' = b$ then return 1 else return 0
```

In the find stage, it is required that $|M_0|=|M_1|$. In the guess stage, adversary $\mathcal{A}$ is not allowed to query $\mathsf{Dec}_{\text{DRE}}(sk_1, \cdot)$ or $\mathsf{Dec}_{\text{DRE}}(sk_2, \cdot)$ on the challenge ciphertext C^*. We define the advantage of $\mathcal{A}$ in the above experiment as

$$\mathbf{Adv}^{\text{cca}}_{\mathcal{DRE},\mathcal{A}}(k) = \Pr[\mathbf{Exp}^{\text{cca}}_{\mathcal{DRE},\mathcal{A}}(k) = 1] - 1/2.$$

A DRE is said to be *indistinguishable against chosen-ciphertext attacks* (IND-CCA) if for any polynomial-time adversary $\mathcal{A}$, $\mathbf{Adv}^{\text{cca}}_{\mathcal{DRE},\mathcal{A}}(k)$ is negligible in k, where the probability is taken over the choice of $\mathsf{crs} \xleftarrow{\$} \mathsf{CGen}_{\text{DRE}}(1^k)$, $(pk_1, sk_1) \xleftarrow{\$} \mathsf{Gen}_{\text{DRE}}(\mathsf{crs})$, $(pk_2, sk_2) \xleftarrow{\$} \mathsf{Gen}_{\text{DRE}}(\mathsf{crs})$, and coins of $\mathcal{A}$. From a standard hybrid argument, we can show that, similar to PKE [3], single-user, single-query DRE security implies multi-user, multi-query DRE security.

3 Practical DRE and DKEM from BDDH Assumption

We build our efficient CCA-secure dual-receiver schemes in the CRS model. The CRS generation algorithm, which takes an input of security parameter k, will output the description of a symmetric bilinear group $\mathcal{BG} = (q, \mathbb{G}, \mathbb{G}_T, e, g)$ where q is a k-bit integer, $\mathbb{G}$ and $\mathbb{G}_T$ are cyclic groups of prime order q, g generates $\mathbb{G}$, and e: $\mathbb{G} \times \mathbb{G} \to \mathbb{G}_T$ is an efficiently computable bilinear map. This definition of $\mathcal{BG}$ will be used throughout the rest of the paper. Note that the public keys of the two receivers should satisfy the "weak separability" property [11], i.e., they should choose their keys from the same bilinear group. For instance, this can be achieved by going through a standard key-setup procedure.

3.1 DRE from BDDH Assumption

Our scheme, detailed in Fig. 1, is symmetric and publicly verifiable. The starting point is a selective-tag weakly CCA-secure tag-based DRE, which can be transformed to a fully secure one by using a strong one-time signature scheme (OTS) $\mathcal{OT} = (\mathsf{Gen}_{\text{OT}}, \mathsf{Sig}_{\text{OT}}, \mathsf{Vrf}_{\text{OT}})$ [27].

Correctness. A ciphertext $(\mathsf{vk}, c, \pi_1, \pi_2, \phi, \sigma)$ is *consistent*, if $\mathsf{Vrf}_{\text{OT}}(\mathsf{vk}, \sigma, (c, \pi_1, \pi_2, \phi)) = 1$, and $e(g, \pi_1) = e(c, u_1^{\mathsf{vk}} v_1)$, and $e(g, \pi_2) = e(c, u_2^{\mathsf{vk}} v_2)$. It is clear that all above can be checked publicly, and in particular, the pairing equations hold if and only if $\pi_1 = c^{x_1 \mathsf{vk} + y_1}$ and $\pi_2 = c^{x_2 \mathsf{vk} + y_2}$. If the ciphertext is consistent then the plaintext can be recovered by either of the two receivers. The receiver obtain the plaintext either via $\phi \cdot e(c, u_2)^{-x_1}$ or $\phi \cdot e(c, u_2)^{-x_2}$. The correctness thus follows from the fact that $e(c, u_2)^{x_1} = e(c, u_1)^{x_2} = e(u_1, u_2)^r$.

Efficiency. The public key for either receiver includes two group elements in $\mathbb{G}$, and the secret key has one element in $\mathbb{Z}_q$. Encryption requires one exponentiation, two multi-exponentiations, one pairing, and a one-time signature computation. Decryption takes five pairings, three exponentiations, and one signature verification. The scheme does not rely on random oracles, having efficiency comparable to the scheme by Kiltz [27] which our scheme relies on.

Security. For soundness, the key point is that the consistency of any ciphertext can be publicly verifiable. If the ciphertext is not consistent then the decryption algorithm for either receiver will reject it (i.e., return $\perp$). Otherwise, any consistent ciphertext $(\mathsf{vk}, c, \pi_1, \pi_2, \phi, \sigma)$ will be decrypted by either of two receivers to the same message, since for any c we have that $e(c, u_2)^{x_1} = e(c, u_1)^{x_2} = e(u_1, u_2)^r$. Therefore, for any ciphertext C (whether consistent or not), we always have that $\mathsf{Dec}_{\text{DRE}}(sk_1, C) = \mathsf{Dec}_{\text{DRE}}(sk_2, C)$. The soundness security thus follows. For its CCA security, we have the following theorem:

Theorem 1. *If $\mathcal{OT}$ is a strongly-unforgeable OTS scheme and the BDDH assumption holds, then the scheme $\mathcal{DRE}$ described in Fig. 1 is a secure DRE against chosen-ciphertext attacks.* ▮

We omit the proof since we will be proving, by analogous but more involved means, essentially a stronger result for a combined encryption scheme in Section 4.

```
$\mathsf{CGen}_{\mathrm{DRE}}(1^k)$
    return $\mathcal{BG}$
$\mathsf{Gen}_{\mathrm{DRE}}(1^k, \mathcal{BG})$
    $x_i, y_i \xleftarrow{\$} \mathbb{Z}_q^*$
    $u_i \leftarrow g^{x_i}; v_i \leftarrow g^{y_i}$
    $pk_i \leftarrow (u_i, v_i)$
    $sk_i \leftarrow x_i$
    return $(pk_i, sk_i)$
```

```
$\mathsf{Enc}_{\mathrm{DRE}}(\mathcal{BG}, pk_1, pk_2, M)$
    $(\mathsf{vk}, \mathsf{sk}) \xleftarrow{\$} \mathsf{Gen}_{\mathrm{OT}}(1^k)$
    $r \xleftarrow{\$} \mathbb{Z}_q^*;\ c \leftarrow g^r$
    $\pi_1 \leftarrow (u_1^{\mathsf{vk}} v_1)^r$
    $\pi_2 \leftarrow (u_2^{\mathsf{vk}} v_2)^r$
    $\phi \leftarrow e(u_1, u_2)^r \cdot M$
    $\sigma \xleftarrow{\$} \mathsf{Sig}_{\mathrm{OT}}(\mathsf{sk}, (c, \pi_1, \pi_2, \phi))$
    return $C \leftarrow (\mathsf{vk}, c, \pi_1, \pi_2, \phi, \sigma)$
```

```
$\mathsf{Dec}_{\mathrm{DRE}}(\mathcal{BG}, pk_1, pk_2, sk_1, C)$
    parse $C$ as $(\mathsf{vk}, c, \pi_1, \pi_2, \phi, \sigma)$
    if $\mathsf{Vrf}_{\mathrm{OT}}(\mathsf{vk}, \sigma, (c, \pi_1, \pi_2, \phi)) \neq 1$ or
        $e(g, \pi_1) \neq e(c, u_1^{\mathsf{vk}} v_1)$ or
        $e(g, \pi_2) \neq e(c, u_2^{\mathsf{vk}} v_2)$
    return $\perp$
    $M \leftarrow \phi \cdot e(c, u_2)^{-x_1}$
    return $M$
```

Fig. 1. DRE from the BDDH assumption: The CRS generation algorithm takes as input the security parameter k and outputs $\mathcal{BG} = (q, \mathbb{G}, \mathbb{G}_T, e, g)$. The key generation algorithms are run independently for user $i \in \{1, 2\}$. The decryption algorithm is specified for user 1, and the decryption algorithm is similar for user 2. The schemes in Section 3.2 and Section 4 have similar formulations.

3.2 DKEM from BDDH Assumption

We extend the concept of dual-receiver encryption to the KEM setting by defining *dual-receiver KEM* $\mathcal{DKEM} = (\mathsf{CGen}_{\mathrm{DKEM}}, \mathsf{Gen}_{\mathrm{DKEM}}, \mathsf{Enc}_{\mathrm{DKEM}}, \mathsf{Dec})$:

- $\mathsf{CGen}_{\mathrm{DKEM}}(1^k)$: The randomized *CRS generation* algorithm takes as input a security parameter k and outputs a CRS crs; we write $\mathsf{crs} \xleftarrow{\$} \mathsf{CGen}_{\mathrm{DKEM}}(1^k)$.
- $\mathsf{Gen}_{\mathrm{DKEM}}(\mathsf{crs})$: The randomized *key generation* algorithm takes as input crs and outputs a public/secret key pair (pk, sk); we write (pk_1, sk_1) and (pk_2, sk_2) for the key pairs of two independent users.
- $\mathsf{Enc}_{\mathrm{DKEM}}(\mathsf{crs}, pk_1, pk_2)$: The randomized *encapsulation algorithm* takes as input the CRS crs and the public keys pk_1 and pk_2 of two users, and outputs a pair (K, C) where $K \in \mathsf{KeySp}$ (i.e, the *encapsulation key space*) is a session key and C is a ciphertext; we write $(K, C) \xleftarrow{\$} \mathsf{Enc}_{\mathrm{DKEM}}(\mathsf{crs}, pk_1, pk_2)$.
- $\mathsf{Dec}_{\mathrm{DKEM}}(\mathsf{crs}, pk_1, pk_2, sk_i, C)$: The deterministic *decapsulation algorithm* takes the CRS crs, the public keys pk_1 and pk_2 of two users, one of the secret keys sk_i ($i \in \{0, 1\}$) and a ciphertext C as input, and outputs either a session key K (which may be the special symbol $\perp$); we write $K_i \leftarrow \mathsf{Dec}_{\mathrm{DKEM}}(\mathsf{crs}, pk_1, pk_2, sk_i, C)$ (or simply $K_i \leftarrow \mathsf{Dec}_{\mathrm{DKEM}}(sk_i, C)$).

As before, we require that the public keys for Enc and Dec are ordered by their lexicographic ordering. Conventional consistency is required. Soundness can be defined as that of DRE, i.e., we require that no (polynomial-time) adversary can, with noticeable probability, produce a ciphertext C (whether consistent or not) such that $\mathsf{Dec}_{\mathrm{DKEM}}(sk_1, C) \neq \mathsf{Dec}_{\mathrm{DKEM}}(sk_2, C)$.

DKEM is a useful building block for *dual-receiver hybrid encryption.* One can easily prove that a hybrid usage of DKEM and a symmetric-key encryption gives a secure and efficient DRE scheme.

Our DKEM $\mathcal{DKEM} = (\mathsf{CGen}_{\mathrm{DKEM}}, \mathsf{Gen}_{\mathrm{DKEM}}, \mathsf{Enc}_{\mathrm{DKEM}}, \mathsf{Dec}_{\mathrm{DKEM}})$ is depicted in Fig. 2. It uses a target collision resistant hash function [13] TCR. Such a hash function is usually "keyed," but this raises problems. First, it does not make sense to let either of the receivers choose the key, since this would immediately damage the symmetry property of DRE. Even if we neglect the symmetry property and allow one of them to choose the key, one has to choose multiple keys (each for per pair of receivers) in the multi-recipient setting, which is clearly prohibitive. Last, it does not make sense to let them jointly choose the hash key, as this would violate the key independence requirement of DRE and be less efficient. For our scheme, we can circumvent the problems in using a *non-keyed* TCR by choosing a bijective encoding function from $\mathbb{G}$ to $\mathbb{Z}_q$, as discussed in the literature [10, 28]. Correspondingly, the hash function is perfectly collision resistant.

```
$\mathsf{CGen}_{\mathrm{DKEM}}(1^k)$
  return $\mathcal{BG}$
$\mathsf{Gen}_{\mathrm{DKEM}}(1^k, \mathcal{BG})$ $i \in \{1,2\}$
  $x_i, y_i \xleftarrow{\$} \mathbb{Z}_q^*$
  $u_i \leftarrow g^{x_i}; v_i \leftarrow g^{y_i}$
  $pk_i \leftarrow (u_i, v_i)$
  $sk_i \leftarrow x_i$
  return $(pk_i, sk_i)$
```

```
$\mathsf{Enc}_{\mathrm{DKEM}}(\mathcal{BG}, pk_1, pk_2)$
  $r \xleftarrow{\$} \mathbb{Z}_q^*$; $c \leftarrow g^r$
  $t \leftarrow \mathsf{TCR}(c)$
  $\pi_1 \leftarrow (u_1^t v_1)^r$
  $\pi_2 \leftarrow (u_2^t v_2)^r$
  $K \leftarrow e(u_1, u_2)^r$
  $C \leftarrow (c, \pi_1, \pi_2)$
  return $(C, K)$
```

```
$\mathsf{Dec}_{\mathrm{DKEM}}(\mathcal{BG}, pk_1, pk_2, sk_1, C)$
  parse $C$ as $(c, \pi_1, \pi_2)$
  $t \leftarrow \mathsf{TCR}(c)$
  if $e(g, \pi_1) \neq e(c, u_1^t v_1)$ or
   $e(g, \pi_2) \neq e(c, u_2^t v_2)$
  return $\perp$
  $K \leftarrow e(c, u_2)^{x_1}$
  return $K$
```

Fig. 2. DKEM from the BDDH assumption

Our DKEM is publicly verifiable, and its correctness easily follows. The perfect soundness is also satisfied similar to the one for the above DRE. The following theorem establishes the chosen-ciphertext security of $\mathcal{DKEM}$:

Theorem 2. *If* TCR *is a target collision resistant hash function and the BDDH assumption holds, then the scheme $\mathcal{DKEM}$ described in Fig. 2 is a secure DKEM against chosen-ciphertext attacks.* ▮

Discussion. At the heart of our schemes is an ElGamal-like encryption in bilinear groups, also used in DLKY-DRE [17]. We also borrow ideas from "identity-based technique" due to Boneh and Boyen [5], and our constructions are similar to that of Kiltz's tag-based encryption [27], and KEMs due to Kiltz [27, 28], and Boyen, Mei, and Waters [10], respectively. Further optimizations and simplifications are applied on our schemes and symmetry has been taken into account.

For typical usage of DRE, the message transmitted is often very sensitive, and one receiver is usually the "supervision party" which is more desirable to have its secret key distributed across multiple parties. It is thus desirable to have a threshold decryptable DRE. We can modify the dual decryption algorithm of the second receiver such that its private key is distributed among n decryption servers which at least k servers are needed for decryption. From the public verifiability, both of our DRE and DKEM proposed can be modified to support threshold decryption [6, 10].

4 Combined Encryption Scheme

A combined encryption scheme $\mathcal{CE}$ consists of the following algorithms ($\mathsf{CGen}_{\mathrm{COM}}$, $\mathsf{Gen}_{\mathrm{COM}}$, $\mathsf{Enc}_{\mathrm{DRE}}$, $\mathsf{Dec}_{\mathrm{DRE}}$, $\mathsf{Enc}_{\mathrm{PKE}}$, $\mathsf{Dec}_{\mathrm{PKE}}$):

- $\mathsf{CGen}_{\mathrm{COM}}(1^k)$: The randomized *CRS generation* algorithm takes as input a security parameter k and outputs a CRS crs; we write $\mathsf{crs} \xleftarrow{\$} \mathsf{CGen}_{\mathrm{COM}}(1^k)$.
- $\mathsf{Gen}_{\mathrm{COM}}(\mathsf{crs})$: The randomized *key generation* algorithm takes as input crs and outputs a public/secret key pair (pk, sk); we write (pk_1, sk_1) and (pk_2, sk_2) for the key pairs of two independent users.
- $\mathsf{Enc}_{\mathrm{DRE}}(\mathsf{crs}, pk_1, pk_2, M)$: The randomized *DRE encryption algorithm* takes as input the CRS crs, the public keys pk_1, pk_2, and a message M, and outputs a ciphertext C; we write $C \xleftarrow{\$} \mathsf{Enc}_{\mathrm{DRE}}(\mathsf{crs}, pk_1, pk_2, M)$.
- $\mathsf{Dec}_{\mathrm{DRE}}(\mathsf{crs}, pk_1, pk_2, sk_i, C)$: The deterministic *DRE decryption algorithm* takes as input two public keys pk_1 and pk_2, one of the secret keys sk_i ($i \in \{0, 1\}$), and a ciphertext C, and outputs a message M_i (which may be the special symbol $\perp$); we write $M_i \leftarrow \mathsf{Dec}_{\mathrm{DRE}}(\mathsf{crs}, pk_1, pk_2, sk_i, C)$ or simply $M_i \leftarrow \mathsf{Dec}_{\mathrm{DRE}}(pk_1, pk_2, sk_i, C)$.
- $\mathsf{Enc}_{\mathrm{PKE}}(\mathsf{crs}, pk_1, M')$: The randomized *PKE encryption algorithm* takes as input the CRS crs and the public key pk_1 and a message M', and outputs a ciphertext C'; we write $C' \xleftarrow{\$} \mathsf{Enc}_{\mathrm{PKE}}(\mathsf{crs}, pk_1, M')$.

- $\mathsf{Dec}_{\mathrm{PKE}}(\mathsf{crs}, pk_1, sk_1, C')$: The deterministic *PKE decryption algorithm* takes as input the CRS crs, the public/secret key pair (pk_1, sk_1), and a ciphertext C', and outputs a message M' (which may be the special symbol $\perp$); we write $M' \leftarrow \mathsf{Dec}_{\mathrm{PKE}}(\mathsf{crs}, pk_1, sk_1, C')$ or simply $M' \leftarrow \mathsf{Dec}_{\mathrm{PKE}}(pk_1, sk_1, C')$.

We require that the public keys in the DRE encryption and decryption algorithms respect their lexicographic ordering. Both the PKE consistency and DRE consistency are required.

```
Experiment $\mathbf{Exp}^{\mathrm{cca}}_{\mathcal{CE},1,\mathcal{A}_1}(k)$
 $\mathsf{crs} \xleftarrow{\$} \mathsf{CGen}_{\mathrm{COM}}(1^k)$
 $(pk_i, sk_i) \xleftarrow{\$} \mathsf{Gen}_{\mathrm{COM}}(\mathsf{crs}) \quad i \in \{1,2\}$
 $(M_0, M_1, \mathsf{s}) \xleftarrow{\$} \mathcal{A}^{\mathcal{O}_1,\mathcal{O}_2,\mathcal{O}_3}(\mathsf{find}, \mathsf{crs}, pk_1, pk_2)$
 $b \xleftarrow{\$} \{0,1\};\ C^* \xleftarrow{\$} \mathsf{Enc}_{\mathrm{DRE}}(\mathsf{crs}, pk_1, pk_2, M_b)$
 $b' \xleftarrow{\$} \mathcal{A}^{\mathcal{O}_1,\mathcal{O}_2,\mathcal{O}_3}(\mathsf{guess}, C^*, \mathsf{s})$
 if $b' = b$ then return 1 else return 0
```

```
Experiment $\mathbf{Exp}^{\mathrm{cca}}_{\mathcal{CE},2,\mathcal{A}_2}(k)$
 $\mathsf{crs} \xleftarrow{\$} \mathsf{CGen}_{\mathrm{COM}}(1^k)$
 $(pk_1, sk_1) \xleftarrow{\$} \mathsf{Gen}_{\mathrm{COM}}(\mathsf{crs})$
 $(M_0, M_1, \mathsf{s}) \xleftarrow{\$} \mathcal{A}^{\mathcal{Q}_1,\mathcal{Q}_2,\mathcal{Q}_3}(\mathsf{find}, \mathsf{crs}, pk_1)$
 $b \xleftarrow{\$} \{0,1\};\ C^* \xleftarrow{\$} \mathsf{Enc}_{\mathrm{PKE}}(\mathsf{crs}, pk_1, M_b)$
 $b' \xleftarrow{\$} \mathcal{A}^{\mathcal{Q}_1,\mathcal{Q}_2,\mathcal{Q}_3}(\mathsf{guess}, C^*, \mathsf{s})$
 if $b' = b$ then return 1 else return 0
```

Fig. 3. (**Left**:) DRE security with PKE decryption oracle (**Right**:) PKE security with DRE decryption oracle

The formalization of the security of combined encryption schemes is more involved than those of previous combined encryption schemes (e.g. [4, 48]), due to the more-than-one receivers nature of DRE. We establish the security of combined encryption scheme by defining *DRE security with PKE decryption oracle* and *PKE security with DRE decryption oracle*. The former captures the security of DRE even with *unrestricted* PKE decryption oracles of the two receivers, while the latter formalizes the security of PKE even with *unrestricted* DRE decryption oracles regarding the target public key and some *arbitrary* (valid) public key even if it does *not* know the corresponding secret key.

Fig. 3 (**Left**) depicts DRE security with PKE decryption oracle, where $\mathcal{O}_1 = \mathsf{Dec}_{\mathrm{DRE}}(sk_1, pk_1, pk_2, \cdot)$, $\mathcal{O}_2 = \mathsf{Dec}_{\mathrm{PKE}}(sk_1, \cdot)$, and $\mathcal{O}_3 = \mathsf{Dec}_{\mathrm{PKE}}(sk_2, \cdot)$. In its guess stage, $\mathcal{A}_1$ is not allowed to query the oracles $\mathsf{Dec}_{\mathsf{DRE}}(sk_1, \cdot)$ on the challenge ciphertext C^*. Note that we do not impose any restrictions on $\mathsf{Dec}_{\mathsf{PKE}}(sk_1, \cdot)$ and $\mathsf{Dec}_{\mathsf{PKE}}(sk_2, \cdot)$ oracles. Fig. 3 (**Right**) describes P-KE security with DRE decryption, where $\mathcal{Q}_1 = \mathsf{Dec}_{\mathrm{PKE}}(sk_1, \cdot)$, $\mathcal{Q}_2 = \mathsf{Dec}_{\mathrm{DRE}}(sk_1, pk_1, \cdot, \cdot)$, and $\mathcal{Q}_3 = \mathsf{Dec}_{\mathrm{DRE}}(sk_1, \cdot, pk_1, \cdot)$. In the guess stage, $\mathcal{A}_2$ is not allowed to query the oracle $\mathsf{Dec}_{\mathrm{PKE}}(sk_1, \cdot)$ with the challenge ciphertext C^*. The query $\mathsf{Dec}_{\mathrm{DRE}}(sk_1, pk_1, \cdot, \cdot)$ on (pk, C) such that $pk_1 <^d pk$ returns $M \leftarrow \mathsf{Dec}_{\mathrm{DRE}}(sk_1, pk_1, pk, C)$, and the oracle query $\mathsf{Dec}_{\mathrm{DRE}}(sk_1, \cdot, pk_1, \cdot)$ on (pk', C) such that $pk' <^d pk_1$ returns $M \leftarrow \mathsf{Dec}_{\mathrm{DRE}}(sk_1, pk', pk_1, C)$. We levy no restrictions except the validity of keys pk and pk' (i.e., output by $\mathsf{Gen}_{\mathrm{COM}}(\mathsf{crs})$).

In the find stages of both experiments, it is required that $|M_0| = |M_1|$. We define the advantage of $\mathcal{A}_i$ in experiment $\mathbf{Exp}^{\mathrm{cca}}_{\mathcal{CE},i,\mathcal{A}_i}(k)$ $(i \in \{1,2\})$ as

$$\mathbf{Adv}^{\mathrm{cca}}_{\mathcal{CE},i,\mathcal{A}_i}(k) = \Pr[\mathbf{Exp}^{\mathrm{cca}}_{\mathcal{CE},i,\mathcal{A}_i}(k) = 1] - 1/2.$$

The soundness for the DRE functionality is identical to that of a regular DRE.

An Efficient Construction. We describe an efficient combined scheme $\mathcal{CE}$, depicted in Fig. 4, which combines our scheme $\mathcal{DRE}$ in Section 3.1 and a PKE scheme adapted from the one based on the BDDH assumption due to Kiltz [27]. The scheme exploits the "identity-based technique" in a *symmetric* manner, where it can be used to simulate all the unrestricted decryption oracles. It is easy to see that the ciphertext consistency of the combined scheme is also publicly verifiable. Theorem 3 below asserts the security of our combined scheme $\mathcal{CE}$.

Theorem 3. *Our $\mathcal{CE}$ is a combined encryption scheme satisfying DRE security with PKE decryption oracle and PKE security with DRE decryption oracle.* ∎

```
$\mathsf{CGen}(1^k)$
  return $\mathcal{BG}$
$\mathsf{Gen}_{\mathrm{COM}}(1^k, \mathcal{BG})$
  $x_i, y_i \xleftarrow{\$} \mathbb{Z}_q^*$
  $u_i \leftarrow g^{x_i}$; $v_i \leftarrow g^{y_i}$
  $w_i \leftarrow g^{z_i}$
  $pk_i \leftarrow (u_i, v_i, w_i)$
  $sk_i \leftarrow x_i$
  return $(pk_i, sk_i)$
```

```
$\mathsf{Enc}_{\mathrm{DRE}}(\mathcal{BG}, pk_1, pk_2, M)$
  $(\mathsf{vk}, \mathsf{sk}) \xleftarrow{\$} \mathsf{Gen}_{\mathrm{OT}}(1^k)$
  $r \xleftarrow{\$} \mathbb{Z}_q^*$
  $c \leftarrow g^r$
  $\pi_1 \leftarrow (u_1^{\mathsf{vk}} v_1)^r$
  $\pi_2 \leftarrow (u_2^{\mathsf{vk}} v_2)^r$
  $\phi \leftarrow e(u_1, u_2)^r \cdot M$
  $\sigma \xleftarrow{\$} \mathsf{Sig}_{\mathrm{OT}}(\mathsf{sk}, (c, \pi_1, \pi_2, \phi))$
  return $C \leftarrow (\mathsf{vk}, c, \pi_1, \pi_2, \phi, \sigma)$
```

```
$\mathsf{Dec}_{\mathrm{DRE}}(\mathcal{BG}, pk_1, pk_2, sk_1, C)$
  parse $C$ as $(\mathsf{vk}, c, \pi_1, \pi_2, \phi, \sigma)$
  if $\mathsf{Vrf}_{\mathrm{OT}}(\mathsf{vk}, \sigma, (c, \pi_1, \pi_2, \phi)) \neq 1$ or
    $e(g, \pi_1) \neq e(c, u_1^{\mathsf{vk}} v_1)$ or
    $e(g, \pi_2) \neq e(c, u_2^{\mathsf{vk}} v_2)$
  return $\perp$
  $M \leftarrow \phi \cdot e(c, u_2)^{-x_1}$
  return $M$
```

```
$\mathsf{Enc}_{\mathrm{PKE}}(\mathcal{BG}, pk_1, M)$
  $(\mathsf{vk}, \mathsf{sk}) \xleftarrow{\$} \mathsf{Gen}_{\mathrm{OT}}(1^k)$
  $r \xleftarrow{\$} \mathbb{Z}_q^*$; $c \leftarrow g^r$
  $\pi \leftarrow (u_1^{\mathsf{vk}} v_1)^r$
  $\phi \leftarrow e(u_1, w_1)^r \cdot M$
  $\sigma \xleftarrow{\$} \mathsf{Sig}_{\mathrm{OT}}(\mathsf{sk}, (c, \pi, \phi))$
  return $C \leftarrow (\mathsf{vk}, c, \pi, \phi, \sigma)$
```

```
$\mathsf{Dec}_{\mathrm{PKE}}(\mathcal{BG}, pk_1, sk_1, C)$
  parse $C$ as $(\mathsf{vk}, c, \pi, \phi, \sigma)$
  if $\mathsf{Vrf}_{\mathrm{OT}}(\mathsf{vk}, \sigma, (c, \pi, \phi) \neq 1$ or
    $e(g, \pi) \neq e(c, u_1^{\mathsf{vk}} v_1)$ then
  return $\perp$
  $M \leftarrow \phi \cdot e(c, w_1)^{-x_1}$
  return $M$
```

Fig. 4. A combined encryption scheme from the BDDH assumption

5 Completely Non-Malleable DRE

Completely non-malleable DRE provides a stronger notion for DRE, which can apply to settings where on-line authorities are available and adversaries might dynamically and maliciously generate public keys. CNM notion of DRE ensures ciphertext non-malleability even in such settings. This section is also motivated by acquiring stronger notions for plaintext equality test as discussed in Section 7, and by dual-receiver non-malleable commitments to be illustrated.

As argued in the introduction, we need to resort to CRS for constructing CNM-DRE. We will propose two *general* approaches to constructing CNM-DRE followed by *efficient* instantiations. We start with a model of CNM-DRE.

5.1 Modeling Completely Non-Malleable DRE

Fischlin [19] gave a simulation-based definition of CNM extending the original definition of non-malleability, and later Ventre and Visconti [42] introduced the game-based definition. We extend the game-based definition of complete non-malleability [42] to the DRE setting and formalize the definition of CNM for DRE. In this setting, we consider a complete relation $\mathtt{R}$ that outputs a boolean variable, and takes as input a plaintext m, two public keys pk_1 and pk_2 for two receivers, two (possibly adversarially generated) public keys pk_1^* and pk_2^*, a vector $\mathbf{m}^*$ of plaintexts, and a vector of DRE ciphertext $\mathbf{c}^*$ encrypting $\mathbf{m}^*$ under pk_1^* and pk_2^*. Consider an experiment with adversary $\mathcal{A}$, as depicted in Fig. 5.

In the experiment, it is mandated that adversary will not query $\mathsf{Dec}_{\mathrm{DRE}}(sk_1, \cdot)$ with c. We also require the chosen distribution M to be valid such that $|m| = |m'|$ for any m and m' having non-zero probability of being sampled. By $\mathbf{m}^* \neq \perp$, we mean that at least one of the elements of $\mathbf{c}^*$ is a valid ciphertext. We define the advantage of $\mathcal{A}$, $\mathbf{Adv}_{\mathcal{DRE},\mathcal{A}}^{\text{cnm-cca}}(k)$, in the above experiments as

$$\Pr[\mathbf{Exp}_{\mathcal{DRE},\mathcal{A}}^{\text{cnm-cca-0}}(k) = 1] - \Pr[\mathbf{Exp}_{\mathcal{DRE},\mathcal{A}}^{\text{cnm-cca-1}}(k) = 1].$$

To thwart a trivial attack, we require the public keys output from the adversary to be in the lexicographic ordering among bit strings (i.e., $pk_1^* <^d pk_2^*$).

When we restrict the public keys (pk_1^*, pk_2^*) output from $\mathcal{A}$ to be exactly (pk_1, pk_2), our definition is equivalent to NM-CCA (and IND-CCA).

```
Experiment $\mathbf{Exp}^{\text{cnm-cca-0}}_{\mathcal{DRE},\mathcal{A}}(k)$
  $\mathsf{crs} \xleftarrow{\$} \mathsf{CGen}_{\mathrm{DRE}}(1^k)$
  $(pk_i, sk_i) \xleftarrow{\$} \mathsf{Gen}_{\mathrm{DRE}}(\mathsf{crs}) \quad i \in \{1,2\}$
  $(\mathsf{M},\mathsf{s}) \xleftarrow{\$} \mathcal{A}^{\mathsf{Dec}_{\mathrm{DRE}}(sk_1,\cdot)}(\mathsf{crs}, pk_1, pk_2)$
  $m \xleftarrow{\$} \mathsf{M}$
  $c \leftarrow \mathsf{Enc}_{\mathrm{DRE}}(\mathsf{crs}, pk_1, pk_2, m, r)$
  $(\mathtt{R}, pk_1^*, pk_2^*, \mathbf{c}^*) \leftarrow \mathcal{A}^{\mathsf{Dec}_{\mathrm{DRE}}(sk_1,\cdot)}(\mathsf{s}, c)$
  return 1 iff $\exists(\mathbf{m}^*, \mathbf{r}^*)$ such that
      $(\mathbf{c}^* = \mathsf{Enc}_{\mathrm{DRE}}(\mathsf{crs}, pk_1^*, pk_2^*, \mathbf{m}^*, \mathbf{r}^*))$ and
      $(c \notin \mathbf{c}^*$ or $(pk_1, pk_2) \neq (pk_1^*, pk_2^*))$ and
      $(\mathbf{m}^* \neq \perp)$ and
      $(\mathtt{R}(m, \mathbf{m}^*, \mathsf{crs}, pk_1, pk_2, pk_1^*, pk_2^*, \mathbf{c}^*) = 1)$
```

```
Experiment $\mathbf{Exp}^{\text{cnm-cca-1}}_{\mathcal{DRE},\mathcal{A}}(k)$
  $\mathsf{crs} \xleftarrow{\$} \mathsf{CGen}_{\mathrm{DRE}}(1^k)$
  $(pk_i, sk_i) \xleftarrow{\$} \mathsf{Gen}_{\mathrm{DRE}}(\mathsf{crs}) \quad i \in \{1,2\}$
  $(\mathsf{M},\mathsf{s}) \xleftarrow{\$} \mathcal{A}^{\mathsf{Dec}_{\mathrm{DRE}}(sk_1,\cdot)}(\mathsf{crs}, pk_1, pk_2)$
  $m, \tilde{m} \xleftarrow{\$} \mathsf{M}$
  $c \leftarrow \mathsf{Enc}_{\mathrm{DRE}}(\mathsf{crs}, pk_1, pk_2, m, r)$
  $(\mathtt{R}, pk_1^*, pk_2^*, c^*) \leftarrow \mathcal{A}^{\mathsf{Dec}_{\mathrm{DRE}}(sk_1,\cdot)}(\mathsf{s}, c)$
  return 1 iff $\exists(\mathbf{m}^*, \mathbf{r}^*)$ such that
      $(\mathbf{c}^* = \mathsf{Enc}_{\mathrm{DRE}}(\mathsf{crs}, pk_1^*, pk_2^*, \mathbf{m}^*, \mathbf{r}^*))$ and
      $(c \notin \mathbf{c}^*$ or $(pk_1, pk_2) \neq (pk_1^*, pk_2^*))$ and
      $(\mathbf{m}^* \neq \perp)$ and
      $(\mathtt{R}(\tilde{m}, \mathbf{m}^*, \mathsf{crs}, pk_1, pk_2, pk_1^*, pk_2^*, \mathbf{c}^*) = 1)$
```

Fig. 5. Modeling the Security of CNM-DRE

We require for CNM-DRE the *strong soundness* property defined in Section 2 since in the setting of CNM the adversary can choose keys adversarially. (Accordingly, we require the encryption scheme to be admissible.) This also ensures that any final output $\mathbf{c}^*$ will give the same plaintext after decrypting.

5.2 CNM-DRE from Groth-Sahai Proof System

It is known that Naor-Yung "two-key" paradigm [32], where the well-formedness of a ciphertext is ensured by the soundness property of a non-interactive zero-knowledge (NIZK) proof, allows dual encryption and decryption but only achieves IND-CCA1 security. It is later shown by Sahai [39] that one can replace the underlying NIZK proof system with a (one-time) simulation-sound NIZK proof system to get IND-CCA security. To achieve complete non-malleability, we employ an even stronger notion of *simulation-sound and simulation-sound extractable NIZK proof of knowledge* [16, 22], which, loosely speaking, requires that the extraction can be achieved even in the simulation setting. The stronger property is needed because it allows the decryption for the forged ciphertext output by the adversary in the simulation setting even if one does not have the corresponding secret key.

An NIZK proof of knowledge system $\mathcal{SSPK}$ is a *proof system* $(\mathsf{CGen}, \mathsf{P}, \mathsf{V})$ together with *knowledge extraction algorithms* $(\mathsf{E}_1, \mathsf{E}_2)$ and *simulation algorithms* $(\mathsf{S}_1, \mathsf{S}_2)$. The proof system satisfies *completeness, soundness, zero-knowledge, simulation soundness, and simulation sound extractability* properties. We assume some familiarity with NIZK and only recall the latter two.

We say a NIZK proof system *simulation sound* if no adversary can prove any false and new statement even with a simulation oracle. Formally, we define the ss-*advantage* against a polynomial-time adversary $\mathcal{A}$, $\mathbf{Adv}^{\mathrm{ss}}_{\mathcal{SSPK},\mathcal{A}}(k)$, for an efficiently computable relation R and a corresponding language $\mathcal{L}$ as $\Pr[(\mathsf{crs}, \tau) \xleftarrow{\$} \mathsf{S}_1(1^k); (x, \pi) \xleftarrow{\$} \mathcal{A}^{\mathsf{S}_2(\mathsf{crs}, \tau, \cdot)}(\mathsf{crs}) \colon (x, \pi)$ is new $\wedge\, x \notin \mathcal{L} \wedge \mathsf{V}(\mathsf{crs}, x, \pi) = 1]$.

We say a NIZK proof is *simulation sound extractable* if one can always extract a witness, in the simulated setting, whenever the adversary with a simulation oracle makes a new proof. Namely, we define the sse-*advantage* against an adversary $\mathcal{A}$, $\mathbf{Adv}^{\mathrm{sse}}_{\mathcal{SSPK},\mathcal{A}}(k)$, for a relation R and a language $\mathcal{L}$ as the probability $\Pr[(\mathsf{crs}, \tau, \mathsf{ek}) \xleftarrow{\$} \mathsf{Gen}_{\mathrm{unite}}(1^k); (x, \pi) \xleftarrow{\$} \mathcal{A}^{\mathsf{S}_2(\mathsf{crs}, \tau, \cdot)}(\mathsf{crs}, \mathsf{ek}); \omega \xleftarrow{\$} \mathsf{E}_2(\mathsf{crs}, \mathsf{ek}, x, \pi) \colon (x, \pi)$ is new $\wedge\, (x, \omega) \notin R \wedge \mathsf{V}(\mathsf{crs}, x, \pi) = 1]$, where $\mathsf{Gen}_{\mathrm{unite}}(1^k)$ is a generation algorithm unifying extraction algorithm E_1 and simulation algorithm S_1 such that they share the same simulated common reference string crs.

Our scheme $\mathcal{CDRE}_1 = (\mathsf{CGen}_{\mathrm{DRE}}, \mathsf{Gen}_{\mathrm{DRE}}, \mathsf{Enc}_{\mathrm{DRE}}, \mathsf{Dec}_{\mathrm{DRE}})$ is detailed in Fig. 6. It employs any *admissible* encryption $\mathcal{PKE} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ and a simulation-sound and simulation-sound

```
$\mathsf{CGen}_{\mathrm{DRE}}(1^k)$
  return $\mathsf{crs} \xleftarrow{\$} \mathsf{CGen}(1^k)$

$\mathsf{Gen}_{\mathrm{DRE}}(1^k)$  $i \in \{1,2\}$
  $(pk_i, sk_i) \xleftarrow{\$} \mathsf{Gen}(1^k)$
  return $(pk_i, sk_i)$
```

```
$\mathsf{Enc}_{\mathrm{DRE}}(\mathsf{crs}, pk_1, pk_2, m)$
  $c_1 \leftarrow \mathsf{Enc}(pk_1, m; r_1)$
  $c_2 \leftarrow \mathsf{Enc}(pk_2, m; r_2)$
  $\pi \xleftarrow{\$} \mathsf{P}(\mathsf{crs}, (c_1, c_2, pk_1, pk_2), (m, r_1, r_2))$
  $c \leftarrow (c_1, c_2, \pi)$
  return $c$
```

```
$\mathsf{Dec}_{\mathrm{DRE}}(\mathsf{crs}, pk_1, pk_2, sk_1, C)$
  parse $C$ as $(c_1, c_2, \pi)$
  if $\mathsf{V}(\mathsf{crs}, c_1, c_2, pk_1, pk_2, \pi) \neq 1$
    return $\perp$
  $m \leftarrow \mathsf{Dec}(c_1, pk_1, sk_1)$
  return $m$
```

Fig. 6. General CNM-DRE from Naor-Yung Paradigm

extractable NIZK argument of knowledge proof system $\mathcal{SSPK} = (\mathsf{CGen}, \mathsf{P}, \mathsf{V}, \mathsf{E}_1, \mathsf{E}_2, \mathsf{S}_1, \mathsf{S}_2)$ for the language $\mathcal{L}_1 := \{(c_1, c_2, pk_1, pk_2) | \exists (m, r_1, r_2)\ [c_1 = \mathsf{Enc}(pk_1, m; r_1) \wedge c_2 = \mathsf{Enc}(pk_2, m; r_2)]\}$, where r_1 and r_2 denote the randomness used by Enc.

Theorem 4. *If encryption $\mathcal{PKE}$ is admissible and indistinguishable under chosen-plaintext attack (IND-CPA), and $\mathcal{SSPK}$ is a simulation-sound and simulation-sound extractable NIZK argument of knowledge proof system, then the scheme $\mathcal{CDRE}_1$ described in Fig. 6 is a secure CNM-DRE against chosen-ciphertext attacks.* ▮

Efficient instantiations. The general construction from simulation-sound NIZK argument of knowledge can be instantiated with reasonable efficiency. In particular, the simulation-sound NIZK *argument* of knowledge can be achieved by using Groth-Sahai proof system which can be realized based on a number of standard assumptions.

To instantiate our construction $\mathcal{CDRE}_1$ depicted in Fig. 6, one can either choose ElGamal encryption as the underlying encryption and use Groth-Sahai proof from the SXDH assumption in asymmetric bilinear groups, or employ linear encryption (from the DLIN assumption) [7] and Groth-Sahai proof from the DLIN assumption in either symmetric or asymmetric bilinear groups. It remains to be shown how to obtain an efficient simulation-sound NIZK argument of knowledge.

Here, we take for example ElGamal encryption and SXDH based Groth-Sahai proof system in an asymmetric bilinear group $\mathcal{BG} = (q, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, e, g, h)$. The CRS contains a verification key vk (for a structure-preserving signature [2] scheme $\mathcal{DS} = (\mathsf{Gen}, \mathsf{Sig}, \mathsf{Vrf})$ and Groth-Sahai reference string for SXDH and pairing product equation setup. Let (y_i, x_i) be the public and secret key of the two receivers $i = 1, 2$ respectively such that $y_i = g^{x_i}$. To show that two ciphertexts $c_1 = (c_{11}, c_{12}) = (g^{r_1}, m_1 y_1^{r_1})$ and $c_2 = (c_{21}, c_{22}) = (g^{r_2}, m_2 y_2^{r_2})$ for two receivers have the same plaintext (i.e., $m_1 = m_2$) is equivalent to prove the satisfiability of a set of pairing product equations by a witness (a_1, b_1, a_2, b_2): $e(c_{11}, h) = e(g, b_1), e(a_1, h) = e(y_1, b_1), e(c_{21}, h) = e(g, b_2), e(a_2, h) = e(y_2, b_2)$, and $e(c_{12} a_1^{-1}, h) = e(c_{22} a_2^{-1}, h)$.

To do so, one makes a proof π that it either knows a witness satisfying the above five pairing product equations, or a structure-preserving signature on the verification key vk of the strong one-time signatures (which can be also verified by a set of pairing product equations). Finally, one computes a signature under vk on $(c_1, c_2, \pi, pk_1, pk_2)$.

The signing key for the structure-preserving signature is the simulation trapdoor. The extraction key of Groth-Sahai proof system is the extraction key.

Discussion. The method that we use to construct simulation-sound NIZK argument of knowledge originates from Groth [22] and follows Boyen, Chevalier, Fuchsbauer, and Pointcheval [9]. The one by Groth [22] does not explicitly use Groth-Sahai proof [23] and thus is less efficient, while the one by Boyen *et al.* [9] only asks for proof of *membership* rather than proof of knowledge.

From another perspective, our CNM-DRE scheme can be viewed as combining the Naor-Yung paradigm [32] (and its descendants, e.g., [39]) and the Rackoff and Simon paradigm [37]. Both the paradigms can lead to (various) CCA-secure PKE schemes, and it is interesting to note that combining them gives rise to stronger CNM-DRE (which clearly implies CCA-secure PKE).

Since the public key for our scheme is admissible, we do not have to worry about adversarially-chosen key issues. That is, we do not need to further add a NIZKPoK for the public keys. Also note that the simulation-sound NIZK argument proof of knowledge is in the "same-string" model, i.e., the honest prover and simulator both use the same reference string. (It is known that in this model NIZK proof does not exist.)

5.3 CNM-DRE from Lossy Trapdoor Functions

This construction follows the CNM-PKE due to Libert and Yung [30] that modifies the PKE from lossy trapdoor functions by Peikert and Waters [36]. In their scheme the family of all-but-one functions is put in the CRS, rather than being generated by the user key generation. We extend this idea to achieve CNM-DRE. Concretely, in our encryption algorithm, the *same* randomness is used as input to two independent lossy trapdoor functions generated by two receivers and the rest of the encryption remains as in Peikert and Waters [36]. To achieve the soundness of the scheme, we ask that *both* of the receivers should check the consistency of *both* of the lossy trapdoor functions. Note that it is easy for them to do so, since the decryption algorithm is *witness-recovering.* We now recall the definitions of lossy and all-but-one trapdoor functions [36] in the following.

Lossy Trapdoor Functions. A collection of (n, l)-lossy trapdoor functions $\mathcal{LTF} = (\mathcal{S}, \mathcal{F}, \mathcal{F}^{-1})$: $\mathcal{S}(1^k, 1)$ is the *injective function sampling* algorithm which outputs (s, t) where s is a function index and t is the trapdoor; $\mathcal{F}(s, \cdot)$ computes an *injective function* over the domain $\{0,1\}^n$, while $\mathcal{F}^{-1}(t, \cdot)$ computes the inverse of the injective function; the *lossy function sampling* algorithm $\mathcal{S}(1^k, 0)$ outputs $(s, \perp)$ where s is a function index; $\mathcal{F}(s, \cdot)$ then computes a deterministic function over $\{0,1\}^n$ such that its image size is at most 2^{n-l}. The first outputs of $\mathcal{S}(1^k, 1)$ and $\mathcal{S}(1^k, 0)$ are computationally indistinguishable.

All-but-one Trapdoor Functions. Let $\mathcal{B} = \{B_k\}_{k \in \mathbb{N}}$ be a collection of sets whose elements represent the *branches.* A collection of (n, l)-all-but-one trapdoor functions $\mathcal{ABO} = (\mathcal{S}_{abo}, \mathcal{G}_{abo}, \mathcal{G}_{abo}^{-1})$ with branch collection $\mathcal{B}_k$ consists of the following algorithms: With a given lossy branch b^*, the *trapdoor function sampling* algorithm $\mathcal{S}_{abo}(1^k, b^*)$ outputs (s, t) where s is a function index and t is the trapdoor. For any $b \in \mathcal{B}$ such that $b \neq b^*$, $\mathcal{G}_{abo}(s, b, \cdot)$ computes an *injective function* over the domain $\{0,1\}^n$, while $\mathcal{G}^{-1}(t, b, \cdot)$ computes the inverse of the injective function. $\mathcal{G}^{-1}(t, b^*, \cdot)$ instead computes a deterministic function such that its image size is at most 2^{n-l}. It is required that, for any $b_0^*, b_1^* \in \mathcal{B}$, the first outputs of $\mathcal{S}_{abo}(1^k, b_0^*)$ and $\mathcal{S}_{abo}(1^k, b_1^*)$ are computationally indistinguishable.

Our Construction. We present $\mathcal{CDRE}_2 = (\mathsf{CGen}_{\mathrm{DRE}}, \mathsf{Gen}_{\mathrm{DRE}}, \mathsf{Enc}_{\mathrm{DRE}}, \mathsf{Dec}_{\mathrm{DRE}})$ in Fig. 7, from a collection of (n, l) lossy trapdoor functions $\mathcal{LTF} = (\mathcal{S}, \mathcal{F}, \mathcal{F}^{-1})$, a collection of (n, l') all-but-one trapdoor functions $\mathcal{ABO} = (\mathcal{S}_{abo}, \mathcal{G}_{abo}, \mathcal{G}_{abo}^{-1})$, and a collection of pairwise independent hash functions [45] H: $\mathcal{H} \times \{0,1\}^n \to \{0,1\}^d$. We require that $2l + l' \geq n + \lambda$, where $\lambda = \omega(\log n)$ and $\lambda > d + \log(1/\epsilon)$ for some negligible function ϵ. This is the only scheme in our paper that is *not* publicly verifiable, i.e., only the receivers can check if a ciphertext is sound.

Theorem 5. *The scheme $\mathcal{CDRE}_2$ adapted from lossy trapdoor functions as described in Fig. 7 is a secure CNM-DRE against chosen-ciphertext attacks.* ▮

We omit the proof which resembles the existing ones [36, 30] but is more involved.

Discussion. The two paradigms (the one based on simulation-sound NIZK argument of knowledge and the one from lossy trapdoor functions) are both general and can be instantiated in

```
$\mathsf{CGen}(1^k)$
    $b_0 \xleftarrow{\$} \{0,1\}^n$
    $(s_0, t_0) \xleftarrow{\$} \mathcal{S}_{abo}(1^k, b_0)$
    $h \xleftarrow{\$} \mathcal{H}$
    return $\mathsf{crs} \leftarrow (s_0, h)$

$\mathsf{Gen}_{\mathrm{DRE}}(1^k) \quad i \in \{1,2\}$
    $(s_i, t_i) \xleftarrow{\$} \mathcal{S}(1^k, 1)$
    return $(s_i, t_i)$

$\mathsf{Enc}_{\mathrm{DRE}}(\mathsf{crs}, s_1, s_2, m; r)$
    $(\mathsf{vk}, \mathsf{sk}) \xleftarrow{\$} \mathsf{Gen}_{\mathrm{OT}}(1^k)$
    $r \xleftarrow{\$} \{0,1\}^n$
    $C_1 \leftarrow \mathcal{F}(s_1, r)$
    $C_2 \leftarrow \mathcal{F}(s_2, r)$
    $C_3 \leftarrow \mathcal{G}_{abo}(s_0, \mathsf{vk}, r)$
    $C_4 \leftarrow M \oplus \mathsf{H}_h(r)$
    $\sigma \xleftarrow{\$} \mathsf{Sig}_{\mathrm{OT}}(\mathsf{sk}, (C_1, C_2, C_3, C_4, pk_1, pk_2))$
    return $C \leftarrow (\mathsf{vk}, C_1, C_2, C_3, C_4, \sigma)$

$\mathsf{Dec}_{\mathrm{DRE}}(s_1, s_2, t_1, C)$
    parse $C$ as $(C_1, C_2, C_3, C_4, pk_1, pk_2, \sigma)$
    if $\mathsf{Vrf}_{\mathrm{OT}}(\mathsf{vk}, \sigma, (C_1, C_2, C_3, C_4, pk_1, pk_2)) \neq 1$ then
        return $\perp$
    $r \leftarrow \mathcal{F}^{-1}(t_1, C_1)$
    if $C_2 \neq \mathcal{F}(s_2, r)$ or $C_3 \neq \mathcal{F}(s_0, r)$ then
        return $\perp$
    $m \leftarrow C_4 \oplus \mathsf{H}_h(r)$
    return $m$
```

Fig. 7. General CNM-DRE from Lossy Trapdoor Functions

a reasonably efficient way. The former can be realized from the SXDH and DLIN assumptions in bilinear groups. It allows short (constant) public keys and constant ciphertext size (more than a hundred group elements though). The latter paradigm has longer ciphertexts, and can be achieved via a number of simpler and more elementary assumptions such as DDH, LWE (learning with errors) [36], and Composite Residuosity [20].

One primary interest in studying completely non-malleable encryption schemes springs from non-malleable commitments. Our CNM-DREs, correspondingly, lead to *dual-receiver non-malleable commitments*, generalizing regular non-malleable commitments, which one can use to commit to the message in a non-malleable sense for two independent receivers with double trapdoors, where they both know that the de-committed messages will be the same. This is a useful property, and it might find other interesting applications.

It would be interesting to propose more efficient CNM-DRE based on specific assumptions. But this seems very challenging, since, roughly speaking, CNM-DRE "requires" *three* trapdoors, two of which must be symmetric.

6 Plaintext-aware Encryption via Registration from DRE

The notion of *plaintext-awareness via key registration*, due to Herzog, Liskov, and Micali [25], requires the sender to go through a key registration step with the authority. Roughly, it captures that an adversary can decrypt any ciphertext that it creates, as long as the adversary registered its sending key. Their construction [25] relies on general zero-knowledge proof of knowledge and non-malleable NIZK, and thus is rather inefficient. We show that our DRE schemes lead to very efficient registration-based plaintext-aware PKE schemes.

Definitions. A *registration-based plaintext-aware encryption* (RPA) scheme consists of the following algorithms: $\mathcal{RPA} = (\mathsf{CGen}_{\mathrm{RPA}}, \mathsf{Gen}_{\mathrm{RPA}}, \mathsf{Enc}_{\mathrm{RPA}}, \mathsf{Dec}_{\mathrm{RPA}}, \mathsf{RU}, \mathsf{RA})$. $\mathsf{CGen}_{\mathrm{RPA}}$ generates the CRS crs which serves as part of the inputs of the following algorithms. RU and RA are two interactive algorithms (i.e., registration protocol) run by the sender and the key registration authority (KRA), respectively. Each takes as input an incoming message and a state, and outputs an outgoing message, an (updated) state, and a decision (accept, reject, or cont). If the sender accepts, its final state output is a sender key pair (pk_s, sk_s). If the KRA accepts, its final state output is the sender public key pk'_s, where $pk_s = pk'_s$ with overwhelming probability. $\mathsf{Gen}_{\mathrm{RPA}}$ generates a key pair (pk_r, sk_r) for the receiver. $\mathsf{Enc}_{\mathrm{RPA}}$ takes as input a message M, the public key of the receiver pk_r, and the public key of the sender pk_s, and outputs a ciphertext C. $\mathsf{Dec}_{\mathrm{RPA}}$ takes as input a ciphertext C, the public key of the receiver pk_r, the public key of the sender pk_s, and the secret key of the receiver sk_r, and outputs a message M.

Apart from the conventional encryption consistency, we expect *honest security*, which ensures that if the receiver and the sender are *both* honest, the scheme should satisfy the (conventional) CCA-security even if the adversary fully controls the KRA. Furthermore, we expect *plaintext-awareness* which guarantees the registered adversary can decrypt any ciphertexts it sends to an receiver, for an honest KRA. We define registration-based plaintext-awareness via the following experiment involving an adversary $\mathcal{A}$ and a simulator $\mathsf{S}_{\mathcal{A}}$:

```
Experiment $\mathbf{Exp}^{\mathrm{rpa}}_{\mathcal{RPA},\mathsf{S}_{\mathcal{A}},\mathcal{A}}(k)$
    $\mathsf{crs} \xleftarrow{\$} \mathsf{CGen}_{\mathrm{DKEM}}(1^k)$
    $(pk_r, sk_r) \xleftarrow{\$} \mathsf{Gen}_{\mathrm{RPA}}(\mathsf{crs})$
    $(pk_{\mathcal{A}}, \mathsf{s}) \xleftarrow{\$} \mathcal{A}^{\mathsf{RA}}(\mathsf{crs}, pk_r)$
    $C \xleftarrow{\$} \mathcal{A}^{\mathsf{Dec}_{\mathrm{RPA}}(\mathsf{crs},\cdot,pk_r,sk_r,\cdot)}(pk_r, pk_{\mathcal{A}}, \mathsf{s})$
    if $\mathsf{S}_{\mathcal{A}}(\mathsf{s},C,pk_r,pk_{\mathcal{A}}) = \mathsf{Dec}_{\mathrm{RPA}}(\mathsf{crs}, C,pk_r,sk_r,pk_{\mathcal{A}})$ then
    return 1 else return 0
```

We define the advantage of $\mathcal{A}$ in the above experiment as $\mathbf{Adv}^{\mathrm{rpa}}_{\mathcal{RPA},\mathsf{S}_{\mathcal{A}},\mathcal{A}}(k) = \Pr[\mathbf{Exp}^{\mathrm{rpa}}_{\mathcal{RPA},\mathsf{S}_{\mathcal{A}},\mathcal{A}}(k) = 0]$. An RPA scheme is registration-based plaintext-aware if for any polynomial-time adversary $\mathcal{A}$ there exists $S_{\mathcal{A}}$ such that the advantage $\mathbf{Adv}^{\mathrm{rpa}}_{\mathcal{RPA},\mathsf{S}_{\mathcal{A}},\mathcal{A}}(k)$ is negligible in the security parameter k.

The Construction of Herzog, Liskov, and Micali. We briefly recall the Herzog, Liskov, and Micali (crypto/HerzogLM03) scheme [25]: Given a CRS crs, the receiver generates two key pairs (pk_1, sk_1), (pk_2, sk_2) of a PKE scheme $(\mathsf{Gen}_{\mathrm{PKE}}, \mathsf{Enc}_{\mathrm{PKE}}, \mathsf{Dec}_{\mathrm{PKE}})$ which is indistinguishable against chosen-plaintext attack (IND-CPA). The public key is $pk_r = (pk_1, pk_2, \mathsf{crs})$ and the secret key $sk_r = sk_1$. The sender generates another pair of public/secret key pair (pk_3, sk_3) for the same encryption scheme. The sender run a zero-knowledge proof of knowledge protocol with the KRA to prove the knowledge of sk_3. The RPA encryption algorithm computes $C = (c_1 = \mathsf{Enc}_{\mathrm{PKE}}(pk_1, m), c_2 = \mathsf{Enc}_{\mathrm{PKE}}(pk_2, m), c_3 = \mathsf{Enc}_{\mathrm{PKE}}(pk_3, m), \pi)$ where π a non-malleable NIZK proof that c_1, c_2, and c_3 encrypt the same message with respect to pk_1, pk_2, and pk_3, respectively. Authenticated channel is needed to make sure the ciphertext was indeed sent by the entity that registered pk_3. The benefit of the above construction is its generality, but it relies on general non-malleable NIZK proofs, which does not seem to have immediate practical instantiations. Another potential drawback is that it is not symmetric, but in real applications the sender might be also the receiver in another instance.

DRE-based Plaintext-Awareness. We show that in general our refined DRE naturally leads to a secure RPA overcoming the drawbacks of crypto/HerzogLM03. The transformation is a simple one. Given a DRE scheme, the sender and the receiver correspond to the two receivers of DRE, and the sender further runs a zero-knowledge proof of knowledge of its secret key protocol with the KRA. The RPA encryption is the same as the DRE encryption relative to the public keys of the sender and the receiver, while the decryption algorithm is just the DRE decryption algorithm relative to the receiver. It is easy to see if the receiver and the sender are both honest, the honest security is implied by the DRE CCA-security. Registration-based plaintext-awareness is also simple to see — for any adversary registered its public key, given a ciphertext, we first rewind the adversary to extract its secret key using the proof of knowledge extractor, then decrypt the given ciphertext with this secret key to obtain a plaintext. Via the soundness of DRE, the obtained plaintext is the same as that by decrypting the ciphertext with the secret key of the receiver. It is also easy to see that the conventional formulation of DRE without the soundness requirement is not adequate, since the ciphertext output by the adversary can be maliciously generated.

The general transformation does not rely on NIZK proof (except for the sender registration process). One can instantiate our DRE based PRA schemes with those in Section 3. The key registration protocol simply runs two (well-known and standard) four-round protocol of zero-knowledge proof of knowledge for discrete logarithm or uses more efficient concurrently secure protocol [14] in the auxiliary string model.

7 More Applications

We investigate further applications or extensions of DRE, which include two types of PKE schemes with plaintext equality test [46, 31, 26, 12, 49], deniable authentication for off-the-record messaging [35, 18], practical PKE with non-interactive opening [15], and useful security puzzle [17, 47].

DRE for Public Plaintext Equality Test. Probabilistic public-key encryption with equality test (PET), first introduced by Yang, Tan, Huang, and Wong [46], allows anyone to be able to check, via a public function Test, whether two independent ciphertexts are encryptions of the same message. They consider one-wayness definition of security for the primitive. Lu, Zhang, and Lin [31] propose and study stronger notions for PET (that loosely speaking are stronger than those for deterministic encryption but weaker than those for regular PKE).

A different concept of PET has also been widely used in many applications such as e-voting schemes (e.g., [26, 12]) and even reliable distributed computing (under a different name, *verifiable dual encryption* [49]). The functionality of the primitive is to provide a NIZK proof that two ciphertexts (encrypted via semantic encryption) have the same underlying plaintext. The existing constructions are only in the ROM.

Our DRE (with refined formulation) generally and naturally handles as well as *strengthens* two different kinds of public-key encryptions with plaintext equality test. In particular, DRE with public verifiability enables PET with public test. The security notion that we can achieve is still IND-CCA, which is the *de facto* notion that one usually considers for a regular PKE. The reason we can do so is that we deal with the ciphertexts in their entirety. Constructing completely non-malleable DRE can be viewed as a further attempt to achieve a stronger notion than IND-CCA for PET. Our combined encryption schemes allow securely combining regular PKE and DRE with IND-CCA-security for both.

Deniable Authentication for Off-the-Record Messaging. Our DRE and DKEM can be useful as an update to the widely used *off-the-record messaging* (OTR) protocol originally due to Borisov, Goldberg, and Brewer (see [35] and references therein), with the stronger deniability notion of Dodis, Katz, Smith, and Walfish (DKSW) [18].

Concretely, DKSW use dual receiver encryption and non-committing encryption to attain an authentication protocol with stronger deniability (i.e., deniability with incriminating abort). The bottleneck for this protocol is just DRE, which harms its genuine utility in practice. Indeed, this is the primary motivation of Smith and Youn [41], where they resort to reasonably efficient Groth-Sahai proofs to achieve the goal. As discussed earlier, their construction is less efficient. With our efficient scheme, DKSW can be now realized efficiently.

Practical PKE with Non-Interactive Opening. Public-key encryption schemes with non-interactive opening (PKENO) [15] allow a receiver to prove to a third party that a ciphertext decrypts to a given plaintext or that the ciphertext is invalid. We note that DRE implies a one-time PKENO. That is, the public key is exactly the same as the public keys of two receivers of DRE, and the encryption and decryption algorithms remain the same as those for DRE. The *proving* algorithm is to simply output the secret key of one of the two receivers. The above method leads to a natural one-time PKENO.

Useful Security Puzzle from Our DRE. Security puzzle is a computational task for the client to solve with its solution relatively efficient for a server to verify. For useful security puzzle, the solution given by the client is associated with what is useful for the server, e.g., the puzzle can be created by another client. Furthermore if the client failed to provide a valid solution, the server can always solve the puzzle at the same cost as a client.

In the original paper due to DLKY [17], DRE is leveraged to create a useful security puzzle. The idea is to let a DRE ciphertext to serve as a puzzle. Due to the public key nature of the encryption, a puzzle can be created by any client. (For higher security, the puzzle can be further post-processed by the server in a “privacy-preservation” stage [17].) The message being encrypted can be something useful for establishing a secure connection between the server and a client. In other words, solving the puzzle requires decrypting the message. The dual-receiver involved in DRE will be set as the server and another client, i.e., the puzzle provided by client A is an DRE encryption of some session information for client A, under the public key of the server and another client B chosen by the server. Finally, REACT transformation [34] is leveraged to provide the efficient validity checking.

Apparently, not only the instantiation provided by DLKY but also the technique for checking the puzzle, are tightly coupled with the random oracle model. Zhang, Hanaoka, and Imai [47] tried to realize the notion of useful security puzzle from identity-based encryption (e.g., [5]) without relying on DRE. With our pairing-based DRE construction, it is easy to build a useful security puzzle without random oracles. The idea is to delegate the pairing computation involved to the client, in a way that the verification of the consistency of the pairing computations can be checked much faster than performing the pairing computations themselves. This is possible, e.g., by using a batch pairing delegation protocol [43].

Acknowledgement

Many thanks for the valuable comments from the anonymous referees and Jens Groth that largely improved the paper. We would also like to thank Adam Smith for his very helpful discussion and in particular pointing us the OTR protocols, and Rui Zhang for his perceptive comments. Sherman S.M. Chow is supported by the Early Career Scheme and the Early Career Award of the Research Grants Council, Hong Kong SAR (CUHK 439713), and grants (4055018, 4930034) from Chinese University of Hong Kong. Haibin Zhang is supported by NSF grants CNS 0904380 and CNS 1228828. Haibin Zhang also received support for this project under NSFC 61272035 while visiting Shandong University.

References

1. M. Abdalla, M. Bellare, D. Catalano, E. Kiltz, T. Kohno, T. Lange, J. Malone-Lee, G. Neven, P. Paillier, and H. Shi. Searchable encryption revisited: consistency properties, relation to anonymous IBE, and extensions. *J. Cryptology* 21(3): 350–391, 2008.
2. M. Abe, G. Fuchsbauer, J. Groth, K. Haralambiev, and M. Ohkubo. Structure-preserving signatures and commitments to group elements. *CRYPTO 2010*, LNCS vol. 6223, Springer, pp. 209–236, 2010.
3. M. Bellare, A. Boldyreva, and S. Micali. Public-key encryption in a multi-user setting: security proofs and improvements. *EUROCRYPT 2000*, LNCS vol. 1807, Springer, pp. 259–274, 2000.
4. J. Baek, R. Safavi-Naini, and W. Susilo. On the integration of public-key data encryption and public-key encryption with keyword search. *ISC 2006*, LNCS vol. 4176, Springer, pp. 217–232, 2006.
5. D. Boneh and X. Boyen. Efficient selective-ID secure identity based encryption without random oracles. *EUROCRYPT 2004*, LNCS vol. 3027, Springer, pp. 223–238, 2004.
6. D. Boneh, X. Boyen, and S. Halevi. Chosen ciphertext secure public-key threshold encryption without random oracles. *CT-RSA 2006*, LNCS vol. 3860, Springer, pp. 226–243, 2006.
7. D. Boneh, X. Boyen, and H. Shacham. Short group signatures. *CRYPTO 2004*, LNCS vol. 3152, Springer, pp. 41–55, 2004.
8. D. Boneh and M. Franklin. Identity-based encryption from the Weil pairing. *CRYPTO 2001*, LNCS vol. 2139, Springer, pp. 213–229, 2001.
9. X. Boyen, C. Chevalier, G. Fuchsbauer, and D. Pointcheval. Strong cryptography from weak secrets. *AFRICACRYPT 2010*, LNCS vol. 6055, Springer, pp. 297–315, 2010.
10. X. Boyen, Q. Mei, and B. Waters. Direct chosen ciphertext security from identity-based techniques. *ACM CCS 2005*, ACM Press, pp. 320–329, 2005.
11. J. Camenisch and M. Michels. Separability and efficiency for generic group signature schemes. *CRYPTO 1999*, LNCS vol. 1666, 1998.

12. M. Clarkson, S. Chong, and A. Myers. Civitas: Toward a secure voting system. *IEEE Symposium on Security and Privacy*, pp. 354–368, 2008.
13. R. Cramer and V. Shoup. Design and analysis of practical public-key encryption schemes secure against adaptive chosen ciphertext attack. *SIAM Journal on Computing*, 33(1):167–226, 2003.
14. I. Damgård. Efficient concurrent zero-knowledge in the auxiliary string model. *EUROCRYPT 2000*, LNCS vol. 1807, Springer, pp. 431–444, 2000.
15. I. Damgård, D. Hofheinz, E. Kiltz, and R. Thorbek. Public-key encryption with non-interactive opening. *CT-RSA 2008*, LNCS vol. 4964, Springer, pp. 239–255, 2008.
16. A. De Santis, G. Di Crescenzo, R. Ostrovsky, G. Persiano, and A. Sahai. Robust non-interactive zero knowledge. *CRYPTO 2001*, LNCS vol. 2139, Springer, pp. 566–598, 2001.
17. T. Diament, H. Lee, A. Keromytis, and M. Yung. The dual receiver cryptosystem and its applications. *CCS 2004*, ACM press, pp. 330–343, 2004.
18. Y. Dodis, J. Katz, A. Smith, and S. Walfish. Composability and on-line deniability of authentication. *TCC 2009*, LNCS vol. 5444, Springer, pp. 146–162, 2009.
19. M. Fischlin. Completely non-malleable schemes. *ICALP 2005*, LNCS vol. 3580, Springer, pp. 779–790, 2005.
20. D. Freeman, O. Goldreich, E. Kiltz, and G. Segev. More constructions of lossy and correlation-secure trapdoor functions. *PKC 2010*, LNCS vol. 6056, Springer, pp. 279–295, 2010.
21. S. Goldwasser, S. Micali, and R. Rivest. A digital signature scheme secure against adaptive chosen-message attacks. *SIAM Journal on Computing*, 17(2):281–308, April 1988.
22. J. Groth. Simulation-sound NIZK proofs for a practical language and constant size group signatures. *ASIACRYPT 2006*, LNCS vol. 4284, Springer, pp. 444–459, 2006.
23. J. Groth and A. Sahai. Efficient non-interactive proof systems for bilinear groups. *EUROCRYPT 2008*, LNCS vol. 4965, Springer, pp. 415–432, 2008.
24. S. Halevi. EME*: Extending EME to handle arbitrary-length messages with associated data. In *INDOCRYPT 2004*, LNCS vol. 3348, Springer, pp. 315–327, 2004.
25. J. Herzog, M. Liskov, and S. Micali. Plaintext awareness via key registration. *CRYPTO 2003*, LNCS vol. 2729, Springer, pp. 548–564, 2003.
26. M. Jakobsson and A. Juels. Mix and match: secure function evaluation via ciphertexts. *ASIACRYPT 2000*, LNCS vol. 1976, Springer, pp. 162–177, 2000.
27. E. Kiltz. Chosen-ciphertext security from tag-based encryption. *TCC 2006*, LNCS vol. 3876, Springer, pp. 581–600, 2006.
28. E. Kiltz. Chosen-ciphertext secure key encapsulation based on hashed gap decisional Diffie-Hellman. *PKC 2007*, LNCS vol. 4450, Springer, pp. 282–297, 2007.
29. B. Libert and J. Quisquater. Identity based encryption without redundancy. *ACNS 2005*, LNCS vol. 3531, Springer, pp. 285–300, 2005.
30. B. Libert and M. Yung. Efficient completely non-malleable public-key encryption. *ICALP 2010*, LNCS vol. 6198, Springer, pp. 127–139, 2010.
31. Y. Lu, R. Zhang, and D. Lin. Stronger security model for public key encryption with equality test. *Pairing 2012*, LNCS vol. 7708, Springer, pp. 65–82, 2012.
32. M. Naor and M. Yung. Public-key cryptosystems provably secure against chosen ciphertext attacks. *STOC 1990*, ACM press, pp. 427–437, 1990.
33. T. Okamoto and D. Pointcheval. The gap-problems: A new class of problems for the security of cryptographic schemes. *PKC 2001*, LNCS vol. 1992, Springer, pp. 104–118, 2001.
34. T. Okamoto and D. Pointcheval. REACT: Rapid enhanced-security asymmetric cryptosystem transform. *CT-RSA 2002*, LNCS vol. 2271, Springer, pp. 159–175, 2002.
35. N. Borisov, I. Goldberg, and E. A. Brewer. Off-the-record communication, or, why not to use PGP. *WPES*, ACM press, pp. 77–84, 2004. More information can be found: http://www.cypherpunks.ca/otr/
36. C. Peikert and B. Waters. Lossy trapdoor functions and their applications. *STOC 2008*, ACM Press, pp. 187–196, 2008.
37. C. Rackoff and D. Simon. Non-interactive zero-knowledge proof of knowledge and chosen ciphertext attack. *CRYPTO 1991*, pp. 433–444, 1991.
38. P. Rogaway. Efficient instantiations of tweakable blockciphers and refinements to modes OCB and PMAC. *ASIACRYPT 2004*, LNCS vol. 3329, Springer, pp. 16–31, 2004.
39. A. Sahai. Non-malleable non-interactive zero knowledge and adaptive chosen-ciphertext security. *FOCS 1999*, IEEE Computer Society, pp. 543–553, 1999.
40. M. Scott. Authenticated id-based key exchange and remote log-in with simple token and pin number. Cryptology ePrint Archive, Report 2002/164, 2002. http://eprint.iacr.org/.
41. A. Smith and Y. Youn. An efficient construction of dual-receiver encryption. Unpublished manuscripts, 2008.
42. C. Ventre and I. Visconti. Completely non-malleable encryption revisited. *PKC 2008*, LNCS vol. 4939, Springer, pp. 65–84, 2008.
43. P. Tsang, S. Chow, and S. Smith. Batch pairing delegation. *IWSEC 2007*, LNCS vol. 4752, Springer, pp. 74–90, 2007.

44. Q. Wu, B. Qin, L. Zhang, and J. Domingo-Ferrer. Ad hoc broadcast encryption. *CCS 2010*, poster, ACM press, pp. 741–743, 2010.
45. M. Wegman and L. Carter. New hash functions and their use in authentication and set equality. *J. Comput. Syst. Sci.*, 22(3): 265–279, 1981.
46. G. Yang, C. Tan, Q. Huang, and D. Wong. Probabilistic public-key encryption with equality test. *CT-RSA 2010*, LNCS vol. 5985, Springer, pp. 119–131, 2010.
47. R. Zhang, G. Hanaoka, and H. Imai. A generic construction of useful client puzzles. *ASIACCS 2009*, ACM press, pp. 70–79, 2009.
48. R. Zhang and H. Imai. Generic combination of public-key encryption with keyword search and public-key encryption. *CANS 2007*, LNCS vol. 4856, Springer, pp. 159–174, 2007.
49. L. Zhou, M. Marsh, F. Schneider, and A. Redz. Distributed blinding for distributed ElGamal re-encryption. *ICDCS 2005*, pp. 815–824, 2005.

A Preliminaries

Notations. If x is a string then $|x|$ denotes its length. If S is a set then $|S|$ denotes its size and $s \overset{\$}{\leftarrow} S$ denotes the operation of selecting an element s of S uniformly at random. If $\mathcal{A}$ is a randomized algorithm then we write $z \overset{\$}{\leftarrow} \mathcal{A}(x, y, \cdots)$ to indicate the operation that runs $\mathcal{A}$ on inputs $x, y, \cdots$ and a uniformly selected r from an appropriately required domain and outputs z. We write $z \overset{\$}{\leftarrow} \mathcal{A}^{\mathcal{O}_1, \mathcal{O}_2, \cdots}(x, y, \cdots)$ to indicate the operation that runs $\mathcal{A}$ having access to oracles $\mathcal{O}_1, \mathcal{O}_2, \cdots$ on inputs $x, y, \cdots$ and outputs z. A function $\epsilon(k)$: $\mathbb{N} \rightarrow \mathbb{R}$ is *negligible* if, for any positive number d, there exists some constant $k_0 \in \mathbb{N}$ such that $\epsilon(k) < (1/k)^d$ for any $k > k_0$.

Complexity Assumptions. We recall the definition of a *bilinear group* $\mathcal{BG} = (q, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, e, g, h)$ where $\mathbb{G}_1$, $\mathbb{G}_2$, and $\mathbb{G}_T$ are cyclic groups of prime order q, g and h generate $\mathbb{G}_1$ and $\mathbb{G}_2$, respectively, and e: $\mathbb{G}_1 \times \mathbb{G}_2 \rightarrow \mathbb{G}_T$ is an efficiently computable bilinear map. We call a bilinear group *symmetric* if $\mathbb{G}_1 = \mathbb{G}_2$, otherwise we call it *asymmetric*. We write $(q, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, e, g, h) \overset{\$}{\leftarrow} \mathsf{BGen}(1^k)$ to denote the algorithm to generate the bilinear groups.

Bilinear Diffie-Hellman Related Assumptions. Given a bilinear group of prime order as above, *the bilinear Diffie-Hellman (BDH) assumption* is that, given (g^a, g^b, g^c), it is hard to compute $e(g, g)^{abc}$. The bilinear decisional Diffie-Hellman (BDDH) assumption is that, given $(g^a, g^b, g^c, e(g, g)^t)$, it is hard to tell whether $t = abc$ or t is random. The Gap BDH (GBDH) assumption, by analogy with the Gap Diffie-Hellman assumption [33], asserts that the BDH assumption holds even given a BDDH oracle.

Symmetric eXternal Diffie-Hellman Assumption (SXDH) [40]. Given a bilinear group of prime order described, *the symmetric external Diffie-Hellman (SXDH) assumption* [40] is that the DDH problem is hard in both $\mathbb{G}_1$ and $\mathbb{G}_2$. This setting implies that there are no efficiently computable homomorphisms between $\mathbb{G}_1$ and $\mathbb{G}_2$.

Decisional Linear Assumption (DLIN) [7]. The *Decisional linear (DLIN) assumption* [7] is first proposed in the setting of symmetric bilinear groups of prime order: Given $(g, g^\alpha, g^\beta, g^{r\alpha}, g^{s\beta}, g^t)$, it is computationally hard to distinguish whether $t = r + s$ or t is random.

Building Blocks. We use the following standard building blocks. They are described and used in the standard model.

Public-Key Encryption. Syntactically, a *public-key encryption* (PKE) scheme $\mathcal{PKE} = (\mathsf{Gen}_{\mathrm{PKE}}, \mathsf{Enc}_{\mathrm{PKE}}, \mathsf{Dec}_{\mathrm{PKE}})$ consists of three algorithms.

- $\mathsf{Gen}_{\mathrm{PKE}}(1^k)$: The randomized *key generation* algorithm takes as input a security parameter k and outputs a public/secret key pair (pk, sk); we write $(pk, sk) \overset{\$}{\leftarrow} \mathsf{Gen}_{\mathrm{PKE}}(1^k)$.
- $\mathsf{Enc}_{\mathrm{PKE}}(pk, M)$: The randomized *encryption algorithm* takes as input the message M and the public key pk and outputs a ciphertext C; we write $C \overset{\$}{\leftarrow} \mathsf{Enc}_{\mathrm{PKE}}(pk, M)$.

- $\mathsf{Dec}_{\mathrm{PKE}}(sk, C)$: The deterministic *decryption algorithm* takes as input a ciphertext C and the secret key sk and outputs a message M (which may be the special symbol $\perp$); we write $M \leftarrow \mathsf{Dec}_{\mathrm{PKE}}(sk, C)$.

It is required that if $(pk, sk) \stackrel{\$}{\leftarrow} \mathsf{Gen}_{\mathrm{PKE}}(1^k)$ and $C \stackrel{\$}{\leftarrow} \mathsf{Enc}_{\mathrm{PKE}}(pk, M)$ then we have $\mathsf{Dec}_{\mathrm{PKE}}(sk, C) = M$ for all the message M from the PKE message space. We recall the IND-CCA-security of PKE scheme $\mathcal{PKE}$ by considering the following experiment that is associated to an adversary $\mathcal{A}$:

```
Experiment $\mathbf{Exp}^{\mathrm{cca}}_{\mathcal{PKE},\mathcal{A}}(k)$
    $(pk, sk) \stackrel{\$}{\leftarrow} \mathsf{Gen}_{\mathrm{PKE}}(1^k)$
    $(M_0, M_1, \mathsf{s}) \stackrel{\$}{\leftarrow} \mathcal{A}^{\mathsf{Dec}_{\mathrm{PKE}}(sk,\cdot)}(\mathsf{find}, pk)$
    $b \stackrel{\$}{\leftarrow} \{0,1\}$; $C^* \stackrel{\$}{\leftarrow} \mathsf{Enc}_{\mathrm{PKE}}(pk, M_b)$
    $b' \stackrel{\$}{\leftarrow} \mathcal{A}^{\mathsf{Dec}_{\mathrm{PKE}}(sk,\cdot)}(\mathsf{guess}, C^*, \mathsf{s})$
    if $b' = b$ then return $1$ else return $0$
```

In the find stage, it is required that $|M_0| = |M_1|$. In the guess stage $\mathcal{A}$ is not allowed to query the oracle $\mathsf{Dec}(sk, \cdot)$ on the challenge ciphertext C^*. We define the advantage of $\mathcal{A}$ in the above experiment as

$$\mathbf{Adv}^{\mathrm{cca}}_{\mathcal{PKE},\mathcal{A}}(k) = \Pr[\mathbf{Exp}^{\mathrm{cca}}_{\mathcal{PKE},\mathcal{A}}(k) = 1] - 1/2.$$

The PKE scheme $\mathcal{PKE}$ is IND-CCA-secure if $\mathbf{Adv}^{\mathrm{cca}}_{\mathcal{PKE},\mathcal{A}}(k)$ is negligible for any polynomial-time adversary $\mathcal{A}$.

We also study PKE (and DRE) in the CRS model, where all the algorithms additionally take as input a common reference string by a trusted party.

Tag Based Encryption. Syntactically, a *tag-based encryption* [27] takes as input an additional "tag" t in both the encryption and decryption algorithms. The security that we require is *selective-tag weakly CCA-security.* Given a tag-based encryption $\mathcal{TE} = (\mathsf{Gen}_{\mathrm{TE}}, \mathsf{Enc}_{\mathrm{TE}}, \mathsf{Dec}_{\mathrm{TE}})$, we associate to an adversary $\mathcal{A}$ the following experiment:

```
Experiment $\mathbf{Exp}^{\text{stag-cca}}_{\mathcal{TE},\mathcal{A}}(k)$
    $(t^*, \mathsf{s}) \stackrel{\$}{\leftarrow} \mathcal{A}(1^k)$
    $(pk, sk) \stackrel{\$}{\leftarrow} \mathsf{Gen}_{\mathrm{TE}}(1^k)$
    $(M_0, M_1, \mathsf{s}) \stackrel{\$}{\leftarrow} \mathcal{A}^{\mathsf{Dec}_{\mathrm{TE}}(sk,\cdot,\cdot)}(\mathsf{find}, pk, \mathsf{s})$
    $b \stackrel{\$}{\leftarrow} \{0,1\}$; $C^* \stackrel{\$}{\leftarrow} \mathsf{Enc}_{\mathrm{TE}}(pk, t, M_b)$
    $b' \stackrel{\$}{\leftarrow} \mathcal{A}^{\mathsf{Dec}_{\mathrm{TE}}(sk,\cdot,\cdot)}(\mathsf{guess}, C^*, \mathsf{s})$
    if $b' = b$ then return $0$ else return $1$
```

In the find stage, it is required that $|M_0| = |M_1|$. In the guess stage $\mathcal{A}$ is not allowed to query the oracle $\mathsf{Dec}_{\mathrm{TE}}(sk, \cdot, \cdot)$ for the target tag t^*. We define the advantage of $\mathcal{A}$ in the above experiment as

$$\mathbf{Adv}^{\text{stag-cca}}_{\mathcal{TE},\mathcal{A}}(k) = \Pr[\mathbf{Exp}^{\text{stag-cca}}_{\mathcal{TE},\mathcal{A}}(k) = 1] - 1/2.$$

The tag-based encryption $\mathcal{TE}$ is selective-tag weakly CCA-secure if $\mathbf{Adv}^{\text{stag-cca}}_{\mathcal{TE},\mathcal{A}}(k)$ is negligible for any polynomial-time adversary $\mathcal{A}$.

We now recall one of the Kiltz's tag-based encryption schemes [27] in the context of symmetric bilinear groups $\mathcal{BG} = (q, \mathbb{G}, \mathbb{G}_T, e, g)$. The public key is $(G, X, Y, Z, W) \in \mathbb{G}^5$ and the secret key is $(x, y) \in \mathbb{Z}_q^2$ such that $X = G^x$ and $Y = G^y$. To encrypt a message M with a tag $t \in \mathbb{Z}_q$, one computes ciphertext $C = (X^r, Y^s, (G^t Z)^r, (G^t W)^s, G^{r+s} M)$ where $(r, s) \in \mathbb{Z}_q^2$ is the randomness used. Given a ciphertext (R, S, U, V, T), the validity can be publicly verified by checking if $e(X, U) = e(R, G^t Z)$ and $e(Y, V) = e(S, G^t W)$. If this is the case, the receiver (having the secret key) computes $M = TR^{-1/x}S^{-1/y}$. The scheme is selective-tag weakly CCA-secure under DLIN assumption.

Key Encapsulation Mechanisms. A *key encapsulation mechanism* $\mathcal{KEM} = (\mathsf{Gen}_{\mathrm{KEM}}, \mathsf{Enc}_{\mathrm{KEM}}, \mathsf{Dec}_{\mathrm{KEM}})$ consists of three algorithms:

- $\mathsf{Gen}_{\mathrm{KEM}}$: The randomized *key generation* algorithm takes as input a security parameter k outputs a public/secret pair (pk, sk); we write $(pk, sk) \xleftarrow{\$} \mathsf{Gen}_{\mathrm{KEM}}(1^k)$.
- $\mathsf{Enc}_{\mathrm{KEM}}$: The randomized *encapsulation algorithm* takes as input the public key pk, and outputs a pair (K, C) where $K \in \mathsf{KeySp}$ is a session key and C is a ciphertext; we write $(K, C) \xleftarrow{\$} \mathsf{Enc}_{\mathrm{KEM}}(pk)$.
- $\mathsf{Dec}_{\mathrm{KEM}}$: The deterministic *decapsulation algorithm* takes as input a ciphertext C and the secret key sk, and outputs either a session key K (which may be the special symbol $\perp$); we write $K \leftarrow \mathsf{Dec}_{\mathrm{KEM}}(sk, C)$.

We require for consistency that for any security parameter k, and all $(K, C) \xleftarrow{\$} \mathsf{Enc}_{\mathrm{KEM}}(pk)$, we have $\Pr[\mathsf{Dec}_{\mathrm{KEM}}(sk) = K] = 1$. To an adversary $\mathcal{A}$ we associate the following experiment:

```
**Experiment** $\mathbf{Exp}^{\text{kem-cca}}_{\mathcal{KEM},\mathcal{A}}(k)$
    $(pk, sk) \xleftarrow{\$} \mathsf{Gen}_{\mathrm{KEM}}(1^k)$
    $K_0^* \xleftarrow{\$} \mathsf{KeySp}$; $(K_1^*, C^*) \xleftarrow{\$} \mathsf{Enc}_{\mathrm{KEM}}(pk)$
    $b \xleftarrow{\$} \{0,1\}$
    $b' \xleftarrow{\$} \mathcal{A}^{\mathsf{Dec}_{\mathrm{KEM}}(sk,\cdot)}(pk, K_b^*, C^*)$
    **if** $b' \neq b$ **then return** 0
    **else return** 1
```

where $\mathcal{A}$ is not allowed to query the oracle $\mathsf{Dec}_{\mathrm{KEM}}(sk, \cdot)$ on the challenge ciphertext C^*. We define the advantage of $\mathcal{A}$ in the above experiment as

$$\mathbf{Adv}^{\text{kem-cca}}_{\mathcal{KEM},\mathcal{A}}(k) = \Pr[\mathbf{Exp}^{\text{kem-cca}}_{\mathcal{KEM},\mathcal{A}}(k) = 1] - 1/2.$$

A KEM is IND-CCA-secure if $\mathbf{Adv}^{\text{cca}}_{\mathcal{KEM},\mathcal{A}}(k)$ is negligible in the security parameter k for any polynomial-time adversary $\mathcal{A}$.

Digital Signature. A *digital signature* $\mathcal{DS}$ consists of three algorithms $(\mathsf{Gen}, \mathsf{Sig}, \mathsf{Vrf})$. A *key generation* algorithm Gen takes the security parameter k and generates a *verification key* vk and a *signing key* sk. A *signing* algorithm Sig computes a signature σ for input message m using the signing key sk. A *verification* algorithm Vrf takes as input vk and a message/signature pair (m, σ) and outputs a single bit b. It is required that for all the messages m it holds that $\Pr[\mathsf{Vrf}(\mathsf{vk}, m, \mathsf{Sig}(\mathsf{sk}, m)) = 1] = 1$. The standard security notion of a digital signature is *existential unforgeability against adaptive chosen message attacks* [21]. Formally, given a signature scheme $\mathcal{DS}$, we associate to an adversary $\mathcal{A}$ the following experiment:

```
**Experiment** $\mathbf{Exp}^{\text{uf}}_{\mathcal{DS},\mathcal{A}}(k)$
    $(\mathsf{vk}, \mathsf{sk}) \xleftarrow{\$} \mathcal{DS}.\mathsf{Gen}(1^k)$
    $(m, \sigma) \xleftarrow{\$} \mathcal{A}^{\mathsf{Sig}(\mathsf{sk},\cdot)}(vk)$
    **if** $\mathsf{Vrf}(\mathsf{vk}, m, \sigma) = 0$ **then return** 0
    **else return** 1
```

where m was not a query of $\mathcal{A}$. We define the advantage of $\mathcal{A}$ in the above experiment as

$$\mathbf{Adv}^{\text{uf}}_{\mathcal{DS},\mathcal{A}}(k) = \Pr[\mathbf{Exp}^{\text{uf}}_{\mathcal{DS},\mathcal{A}}(k) = 1].$$

In particular, a signature scheme is *structure-preserving* [2] if its verification keys, messages, and signatures are group elements and verification algorithm is a set of pairing product equations.

We also use a notion of *strong one-time signature* where it is secure if no probabilistic polynomial-time adversary that has access to a single chosen message attack oracle can create a new message/signature pair (m, σ).

Pairwise Independent Hash Functions. A hash function $\mathsf{H}\colon \mathcal{H} \times \mathcal{D} \to \mathcal{R}$ is called *pairwise independent hash* [45] with *key space* $\mathcal{H}$, if for every distinct $x, x' \in \mathcal{D}$ and every $y, y' \in \mathcal{R}$, it holds:

$$\Pr[h \xleftarrow{\$} \mathcal{H}\colon \mathsf{H}_h(x) = y \wedge \mathsf{H}_h(x') = y'] = 1/|\mathcal{R}|^2.$$

We will use the pairwise independent hash function as a *strong randomness extractor*.

Target Collision Resistant Hash Functions. Let $\mathsf{TCR}\colon \mathcal{S} \times \mathbb{G} \to \mathbb{Z}_q$ be a *target collision resistant hash function* [13] with *key space* $\mathcal{S} = \{0,1\}^k$. We define the tcr-*advantage* of adversary $\mathcal{A}$ for TCR as $\mathbf{Adv}^{\text{tcr}}_{\mathcal{TCR},\mathcal{A}}(k) = \Pr[s \xleftarrow{\$} \{0,1\}^k; c^* \xleftarrow{\$} \mathbb{G}; c \xleftarrow{\$} \mathcal{A}(s, c^*)\colon c \neq c^* \wedge \mathsf{TCR}(s,c) = \mathsf{TCR}(s,c^*)]$.

Groth-Sahai Proof System. Groth-Sahai proof system [23] provides efficient (composable) NIWI proofs and NIZK proofs in the *common reference string model* for a large set of statements involving bilinear groups, including *pairing product equations*, *multi-scalar multiplication equations*, and *quadratic equations*. This system can be instantiated under three assumptions: the SXDH assumption (in asymmetric bilinear groups), the DLIN assumption (in symmetric bilinear groups), and the subgroup decision assumption (in composite order bilinear groups). There are two types of common reference strings (which are computationally indistinguishable) yielding perfect (co-)soundness and perfect witness-indistinguishability (or zero-knowledge) respectively. A Groth-Sahai proof system consists of four algorithms $(\mathsf{Gen}_{\text{GS}}, \mathsf{P}_{\text{GS}}, \mathsf{V}_{\text{GS}}, \mathsf{Extr}_{\text{GS}})$. The *key generation* algorithm Gen_{GS} takes a security parameter and outputs a common reference string crs together with an *extraction key xk*. The *prover* P_{GS} takes as input crs and witnesses of equations and outputs a proof π. The *verifier* V_{GS} takes as input crs and π and outputs a bit b with respect to a set of equations. The $\mathsf{Extr}_{\text{GS}}$ algorithm taking as input the extraction key ek can extract the *group elements witnesses*. Therefore, for the equations whose witnesses are group elements the above proofs as well provide *proofs of knowledge (PoK)*.

B DRE without Redundancy in the ROM

In the ROM, one can design more efficient DRE schemes. In particular, the DRE scheme (in fact, a DKEM) that we will describe has the most compact ciphertext, i.e., without redundancy. Our construction is similar to existing PKE without redundancy [13, 29].

The system will specify a hash function H (which will be modeled as a random oracle in the security proof), a secure cipher $(\mathsf{SE}, \mathsf{SD})$ which is a strong pseudo-random permutation (SPRP) (e.g., [24]). The public key is $X = g^x$ where x is the corresponding secret key. Given two independent public keys X_1 and X_2 with corresponding secret keys x_1 and x_2, the encryption scheme proceeds as follows: one first selects a random element r, and sets

$$Y \leftarrow g^r, \quad T \leftarrow e(X_1, X_2)^r, \quad k \leftarrow \mathsf{H}(Y, T), \quad c \leftarrow \mathsf{SE}_k(m).$$

The ciphertext is (Y, c). Given such a ciphertext, one can decrypt it using either x_1 or x_2. For instance, the first receiver computes $T \leftarrow e(Y, X_2)^{x_1}$, recovers $k \leftarrow \mathsf{H}(Y, T)$, and gets $m \leftarrow \mathsf{SD}_k(c)$.

One can show that, with a similar argument as existing proofs [13, 29], the above scheme is a secure DRE (a hybrid DRE) under the GBDH assumption in the ROM.

C Efficient DRE from Groth-Sahai Proof System

Naor-Yung paradigm is perhaps the most natural method to obtain an efficient DRE. It is tempting to use Groth-Sahai proof system to realize this method in an efficient way. The idea of the following scheme is first described in an unpublished manuscript by Smith and Youn [41]. The scheme uses the Kiltz's tag-based encryption and Groth-Sahai NIZK proof. Compared to the schemes in Section 3, the scheme is efficient but not practical — containing nearly a hundred group elements.

We first pick Kiltz's tag-based encryption $\mathcal{TE}$ (based on DLIN assumption) [27], and select a strong one-time signature scheme $\mathcal{OT} = (\mathsf{Gen}_{\mathrm{OT}}, \mathsf{Sig}_{\mathrm{OT}}, \mathsf{Vrf}_{\mathrm{OT}})$. A Groth-Sahai proof system common reference string crs is selected. The public key of each user is just the one from $\mathcal{TE}$ which is $pk_i = (G_i, X_i, Y_i, Z_i, W_i)$ where $i \in \{1, 2\}$. To encrypt m, one first generates a strong one-time signature key pair $(\mathsf{vk}, \mathsf{sk}) \xleftarrow{\$} \mathsf{Gen}_{\mathrm{OT}}$, and uses vk as the tag to run $\mathcal{TE}$ for receiver 1 and receiver 2, respectively, to get $C_i = (X_i^{r_i}, Y_i^{s_i}, (G_i^{\mathsf{vk}} Z_i)^{r_i}, (G_i^{\mathsf{vk}} W_i)^{s_i}, G_i^{r_i+s_i} m)$ for $i \in \{1, 2\}$. It then adds a Groth-Sahai NIZK proof π such that they encrypt the same plaintext. Finally, it uses sk to get a signature σ for (C_1, C_2, π) and outputs $(\mathsf{vk}, C_1, C_2, \pi, \sigma)$ as the DRE ciphertext. The correctness of DRE scheme follows from the correctness of Kiltz's tag-based encryption, perfect completeness of Groth-Sahai proof system, and correctness of strong one-time signature.

```
$\mathsf{CGen}_{\mathrm{DRE}}(\mathsf{crs})$
  $\mathsf{crs} \xleftarrow{\$} \mathsf{Gen}_{\mathrm{GS}}(1^k)$

$\mathsf{Gen}_{\mathrm{DRE}}(\mathsf{crs})\ i \in \{1, 2\}$
  $x_i, y_i \xleftarrow{\$} \mathbb{Z}_q^*$
  $X_i \leftarrow g^{x_i}$
  $Y_i \leftarrow g^{y_i}$
  $G_i, Z_i, W_i \xleftarrow{\$} \mathbb{G}$
  $pk_i \leftarrow (G_i, X_i, Y_i, Z_i, W_i)$
  $sk_i \leftarrow (x_i, y_i)$
  return $(pk_i, sk_i)$

$\mathsf{Enc}_{\mathrm{DRE}}(\mathsf{crs}, pk_1, pk_2, M)$
  $(\mathsf{vk}, \mathsf{sk}) \xleftarrow{\$} \mathsf{Gen}_{\mathrm{OT}}(1^k)$
  for $i = 1, 2$ do
    $r_i, s_i \xleftarrow{\$} \mathbb{Z}_q^*$
    $R_i \leftarrow X_i^{r_i};\ S_i \leftarrow Y_i^{s_i}$
    $U_i \leftarrow (G_i^{\mathsf{vk}} Z_i)^{r_i}; V_i \leftarrow (G_i^{\mathsf{vk}} W_i)^{s_i}$
    $T_i \leftarrow G_i^{r_i+s_i} M$
    $C_i \leftarrow (R_i, S_i, U_i, V_i, T_i)$
  $\pi \xleftarrow{\$} \mathsf{P}_{\mathrm{GS}}(\mathsf{crs}, C_1, C_2)$
  $\sigma \xleftarrow{\$} \mathsf{Sig}_{\mathrm{OT}}(\mathsf{sk}, (C_1, C_2, \pi, pk_1, pk_2))$
  return $C \leftarrow (\mathsf{vk}, C_1, C_2, \pi, \sigma)$

$\mathsf{Dec}_{\mathrm{DRE}}(\mathsf{crs}, sk_1, pk_1, pk_2, C)$
  parse $C$ as $(\mathsf{vk}, C_1, C_2, \pi, \sigma)$
  if $\mathsf{Vrf}_{\mathrm{OT}}(\mathsf{vk}, \sigma, (C_1, C_2, \pi, pk_1, pk_2) \neq 1$ or
    $e(X_1, U_1) \neq e(R_1, G_1^{\mathsf{vk}} Z_1)$ or
    $e(Y_1, V_1) \neq e(S_1, G_1^{\mathsf{vk}} W_1)$ or
    $e(X_2, U_2) \neq e(R_2, G_2^{\mathsf{vk}} Z_2)$ or
    $e(Y_2, V_2) \neq e(S_2, G_2^{\mathsf{vk}} W_2)$ or
    $\mathsf{V}_{\mathrm{GS}}(\mathsf{crs}, C_1, C_2, \pi) \neq 1$ then
    return $\perp$
  $M \leftarrow T_1 R_1^{-1/x_1} S_1^{-1/y_1}$
  return $M$
```

Fig. 8. Efficient DRE from Groth-Sahai Proof System. The common reference string crs contains the bilinear map parameter $\mathcal{BG} = (q, \mathbb{G}, \mathbb{G}_T, e, g)$ besides the Groth-Sahai proof parameter. Here we use DLIN setup of Groth-Sahai proof system.

The most direct implementation method for the NIZK is to use Groth-Sahai *multi-exponentiation equations*. Concretely, for this construction, we need to first commit to the randomness used for the encryption (i.e., (r_1, s_1, r_2, s_2), which are Groth-Sahai variables) and then give a NIZK proof π such that two ciphertexts $(R_1, S_1, U_1, V_1, T_1)$ and $(R_2, S_2, U_2, V_2, T_2)$ encrypted the same plaintext. Specifically, there should exist (r'_1, s'_1, r'_2, s'_2) such that $R_1 = X_1^{r'_1}$, $S_1 = Y_1^{s'_1}$, $R_2 = X_2^{r'_2}$, $S_2 = Y_2^{s'_2}$, and $T_1/T_2 = G_1^{r'_1+s'_1}/G_2^{r'_2+s'_2}$.

One can also choose to use Groth-Sahai pairing product equations. Given two consistent ciphertexts $C_1 = (R_1, S_1, U_1, V_1, T_1)$ and $C_2 = (R_2, S_2, U_2, V_2, T_2)$ where the consistency can be publicly verified by checking the pairing equations, C_1 and C_2 encrypted the same plaintext M if and only if we give a set of pairing product equations that are satisfiable with a witness (A_1, A_2, A_3, A_4) such that: $e(R_1, G_1) = e(X_1, A_1)$, $e(S_1, G_1) = e(Y_1, A_2)$, $e(R_2, G_2) = e(X_2, A_3)$, $e(S_2, G_2) = e(Y_2, A_4)$, and $e(T_1 A_1^{-1} A_2^{-1}, G_1) = e(T_2 A_3^{-1} A_4^{-1}, G_1)$. The witness satisfying the first four equations (A_1, A_2, A_3, A_4) is thus $(G_1^{r_1}, G_1^{s_1}, G_2^{r_2}, G_2^{s_2})$. The last equation implies C_1 and C_2 encrypted the same plaintext M. The Groth-Sahai witness indistinguishable proof for the above set of equations can be easily adapted to be zero-knowledge. The first four equations are "homogeneous equations" which can be simulated directly since they have trivial witnesses; the last one can also be simulated by adding one multi-scalar multiplication equation.

D Proof of Theorem 2

Before proving Theorem 2, we first define chosen-ciphertext attacks for DKEM.

DKEM under Chosen-Ciphertext Attacks. We associate the following experiment to an adversary $\mathcal{A}$:

```
$\mathbf{Experiment}\ \mathbf{Exp}^{\text{kem-cca}}_{\mathcal{DKEM},\mathcal{A}}(k)$
    $\mathsf{crs} \xleftarrow{\$} \mathsf{CGen}_{\mathrm{DRE}}(1^k)$
    $(pk_1, sk_1) \xleftarrow{\$} \mathsf{Gen}_{\mathrm{DRE}}(\mathsf{crs}), (pk_2, sk_2) \xleftarrow{\$} \mathsf{Gen}_{\mathrm{DRE}}(\mathsf{crs})$
    $K_0^* \xleftarrow{\$} \mathsf{KeySp};\ (K_1^*, C^*) \xleftarrow{\$} \mathsf{Enc}_{\mathrm{DKEM}}(\mathsf{crs}, pk_1, pk_2)$
    $b \xleftarrow{\$} \{0,1\}$
    $b' \xleftarrow{\$} \mathcal{A}^{\mathsf{Dec}_{\mathrm{DKEM}}(sk_1,\cdot)}(pk_1, pk_2, K_b^*, C^*)$
    $\mathbf{if}\ b' \neq b\ \mathbf{then\ return}\ 0\ \mathbf{else\ return}\ 1$
```

where $\mathcal{A}$ is not allowed to query the oracles $\mathsf{Dec}_{\mathrm{DKEM}}(sk_1, \cdot)$ on the challenge ciphertext C^*. We define the advantage of $\mathcal{A}$ in the above experiment as

$$\mathbf{Adv}^{\text{kem-cca}}_{\mathcal{DKEM},\mathcal{A}}(k) = \Pr[\mathbf{Exp}^{\text{kem-cca}}_{\mathcal{DKEM},\mathcal{A}}(k) = 1] - 1/2.$$

Proof: Assume there exists a polynomial time adversary $\mathcal{A}$ that breaks the chosen-ciphertext security of the DKEM $\mathcal{DKEM}$ with (non-negligible) advantage $\mathbf{Adv}^{\text{kem-cca}}_{\mathcal{DKEM},\mathcal{A}}(k)$. We show that there exist adversaries $\mathcal{B}$ against a random instance of the BDDH problem and $\mathcal{C}$ against a target collision resistant hash function such that

$$\mathbf{Adv}^{\text{bddh}}_{\mathcal{BG},\mathcal{B}}(k) \geq \mathbf{Adv}^{\text{cca}}_{\mathcal{DRE},\mathcal{A}}(k) - \mathbf{Adv}^{\text{tcr}}_{\mathsf{TCR},\mathcal{C}}(k).$$

Now we give the description of $\mathcal{B}$ taking as input a random BDDH instance $(g^{a_1}, g^{a_2}, g^r, K)$ for a symmetric bilinear group $\mathcal{BG} = (q, \mathbb{G}, \mathbb{G}_T, e, g)$. $\mathcal{B}$ would like to determine whether $K = e(g,g)^{a_1 a_2 r}$ or K is a random element in $\mathbb{G}_T$. Adversary $\mathcal{B}$ simulates adversary $\mathcal{A}$'s view as follows: Adversary $\mathcal{B}$ randomly selects d_1 and d_2 from $\mathbb{Z}_q^*$ and computes a challenge ciphertext

$$C^* = (c^*, \pi_1^*, \pi_2^*) = (g^r, (g^r)^{d_1}, (g^r)^{d_2}).$$

with the corresponding challenge key being K. Let t^* be $\mathsf{TCR}(c^*)$. The public key (u_i, v_i) for either receiver $i \in \{0,1\}$ is defined respectively as

$$(u_i = g^{a_i}, \quad v_i = u_i^{-t^*} \cdot g^{d_i}).$$

It is easy to see that the public keys are distributed just as in the DKEM experiment. Since $(u_i^{t^*} v_i)^r = (g^r)^{d_i}$ for $i \in \{0,1\}$ and where $t^* = \mathsf{TCR}(g^r)$, the challenge ciphertext $C^* = (c^*, \pi_1^*, \pi_2^*) = (g^r, (g^r)^{d_1}, (g^r)^{d_2})$ is distributed correctly (for the unknown randomness r). If $K = e(g,g)^{a_1 a_2 r}$ then $K = e(u_1, u_2)^r$ is a valid session key for the challenge ciphertext. Otherwise, K is independently and uniformly distributed in $\mathbb{G}_T$.

We then show how adversary $\mathcal{B}$ can simulate the decapsulation oracle by adversary $\mathcal{A}$. Suppose $C = (c, \pi_1, \pi_2)$ be any ciphertext to the decapsulation oracle. Adversary $\mathcal{B}$ first checks if C is consistent (i.e., whether the two pairing equations are satisfied). If C is not consistent then $\mathcal{B}$ simply returns $\perp$, just as in the original experiment; otherwise, adversary $\mathcal{B}$ computes $t = \mathsf{TCR}(c)$ and we distinguish three cases:

- Case 1. If $t = t^*$ and $c = c^*$ then we have that $\pi_1 = c^{d_1} = (c^*)^{d_1} = \pi_1^*$ and $\pi_2 = c^{d_2} = (c^*)^{d_2} = \pi_2^*$. Adversary $\mathcal{B}$ rejects the query, since now $C = C^*$.
- Case 2. If $t = t^*$ and $c \neq c^*$ then adversary outputs c and c^* and aborts.
- Case 3. If $t \neq t^*$ then adversary $\mathcal{B}$ computes $(\pi_1/c^{d_1})^{(t-t^*)^{-1}}$ and returns the session key $e((\pi_1/c^{d_1})^{(t-t^*)^{-1}}, u_2)$. This is a valid session key since we have $\pi_1 = (u_1^t v_1)^{r'} = (u_1^{r'})^{t-t^*}$ for some r' where $r' = \log_g c$.

Finally, adversary $\mathcal{A}$ outputs a guess b'; adversary $\mathcal{B}$ outputs the same guess.

It is easily seen that as long as $\mathcal{B}$ did not find a collision (Case 2) the simulation of the decapsulation oracle is perfect. We can also easily prove that the probability that $\mathcal{B}$ finds a collision in the hash function TCR is bounded by $\mathbf{Adv}^{\mathrm{tcr}}_{\mathsf{TCR},\mathcal{C}}(k)$, where $\mathcal{C}$ is an adversary against the target collision resistance of hash function TCR. For our specific scheme, the probability is equal to zero — recall that we use a bijective encoding function. Following a standard argument, we can show that

$$\mathbf{Adv}^{\mathrm{bddh}}_{\mathcal{BG},\mathcal{B}}(k) \geq \mathbf{Adv}^{\mathrm{cca}}_{\mathcal{DRE},\mathcal{A}}(k).$$

This completes the proof of the theorem. ▮

Remarks. Note that (in Case 3) one recovers the key by first computing $(\pi_1/c^{d_1})^{(t-t^*)^{-1}}$. It is equally feasible to first compute $(\pi_2/c^{d_2})^{(t-t^*)^{-1}}$ and then return $e(u_1, (\pi_2/c^{d_2})^{(t-t^*)^{-1}})$. In this sense, the proof is also "symmetric", which seems to be naturally required by the symmetry property of DRE. The simulation "redundancy", on the other hand, turns out to be crucial for proving the security of combined encryption scheme in Section 4 (for Theorem 3).

E Proof of Theorem 3

Proof: We first prove the *DRE security with PKE decryption oracle*. The proof resembles that for Theorem 2, and the crux is to show how to simulate the unrestricted PKE oracle.

Suppose there exists a polynomial time adversary $\mathcal{A}$ that breaks, for $\mathcal{CE}$, the DRE security with PKE decryption oracle with (non-negligible) advantage $\mathbf{Adv}^{\mathrm{cca}}_{\mathcal{CE},1,\mathcal{A}}(k)$. We show that there exist adversaries $\mathcal{B}$ against a random instance of the BDDH problem and adversary $\mathcal{C}$ against the strong unforgeability of one-time signature $\mathcal{OT}$ such that

$$\mathbf{Adv}^{\mathrm{bddh}}_{\mathcal{BG},\mathcal{B}}(k) \geq \mathbf{Adv}^{\mathrm{cca}}_{\mathcal{CE},1,\mathcal{A}}(k) - \mathbf{Adv}^{\mathrm{suf}}_{\mathcal{OT},\mathcal{C}}(k).$$

We give the description of $\mathcal{B}$ taking as input a random BDDH instance $(g^{a_1}, g^{a_2}, g^r, K)$ for a symmetric bilinear group $\mathcal{BG} = (q, \mathbb{G}, \mathbb{G}_T, e, g)$. $\mathcal{B}$ would like to determine whether $K = e(g,g)^{a_1 a_2 r}$ or K is a random element in $\mathbb{G}_T$. Adversary $\mathcal{B}$ simulates adversary $\mathcal{A}$'s view as follows:

The public keys are prepared similarly as those of Theorem 2. Adversary $\mathcal{B}$ runs $\mathsf{Gen}_{\mathrm{OT}}(1^k)$ to return a verification/signing key pair $(\mathsf{vk}^*, \mathsf{sk}^*)$. It then randomly selects d_1 and d_2 from $\mathbb{Z}_q^*$ and defines the public key (u_i, v_i) for either receiver $i \in \{0,1\}$ respectively as

$$(u_i = g^{a_i}, \quad v_i = u_i^{-\mathsf{vk}^*} \cdot g^{d_i}).$$

It is clear that the public keys are distributed just as in the DKEM experiment.

We then show how adversary $\mathcal{B}$ can simulate the decryption oracles by adversary $\mathcal{A}$. Adversary $\mathcal{B}$ rejects all the DRE and PKE ciphertexts that are not consistency, just as in the original experiment. Otherwise, we show how adversary $\mathcal{B}$ to answer all three types of decryption oracles:

For any DRE ciphertext $C = (\mathsf{vk}, c, \pi_1, \pi_2, \phi, \sigma)$ to $\mathsf{Dec}_{\mathrm{DRE}}(sk_1, \cdot)$, if $\mathsf{vk} = \mathsf{vk}^*$ then it simply aborts; otherwise, it computes $(\pi_1/c^{d_1})^{(t-t^*)^{-1}}$ and returns $\phi/e((\pi_1/c^{d_1})^{(t-t^*)^{-1}}, u_2)$.

For any PKE ciphertext $C = (\mathsf{vk}, c, \pi, \phi, \sigma)$ to $\mathsf{Dec}_{\mathrm{PKE}}(sk_1, \cdot)$, if $\mathsf{vk} = \mathsf{vk}^*$ then it simply aborts; otherwise, it computes $(\pi_1/c^{d_1})^{(t-t^*)^{-1}}$ and returns $\phi/e((\pi_1/c^{d_1})^{(t-t^*)^{-1}}, u_2)$.

For any PKE ciphertext $C = (\mathsf{vk}, c, \pi, \phi, \sigma)$ to $\mathsf{Dec}_{\mathrm{PKE}}(sk_2, \cdot)$, if $\mathsf{vk} = \mathsf{vk}^*$ then it simply aborts; otherwise, it computes $(\pi_2/c^{d_2})^{(t-t^*)^{-1}}$ and returns $\phi/e(u_1, (\pi_2/c^{d_2})^{(t-t^*)^{-1}})$.

At some point, adversary $\mathcal{A}$ outputs two messages M_0 and M_1. Adversary $\mathcal{B}$ provides $\mathcal{A}$ with the following challenge ciphertext:

$$C^* = (\mathsf{vk}^*,\ c^*,\ \pi_1^*,\ \pi_2^*,\ \phi^*,\ \sigma^*)$$

where, above, $c^* = g^r, \pi_1^* = (g^r)^{d_1}, \pi_2^* = (g^r)^{d_2}$, $\phi^* = K \cdot M_b$, and $\sigma^* = \mathsf{Sig}_{\mathsf{sk}^*}(c^*, \pi_1^*, \pi_2^*)$.

Since for $i = 1, 2$, $(u_i^{\mathsf{vk}^*} v_i)^r = (g^r)^{d_i}$, the ciphertext is a valid challenge ciphertext for unknown randomness r.

Adversary $\mathcal{A}$ may further make decryption oracle queries with the only restriction that it may not query $\mathsf{Dec}_{\mathrm{DRE}}(sk_1, C^*)$ after receiving the challenge ciphertext C^*.

Finally, adversary $\mathcal{A}$ outputs a guess b'; adversary $\mathcal{B}$ outputs the same guess.

Let Forge be the event that $\mathcal{A}$ submits a valid ciphertext $(\mathsf{vk}^*, c, \pi_1, \pi_2, \phi, \sigma)$ to the $\mathsf{Dec}_{\mathrm{DRE}}(sk_1, \cdot)$ oracle (recall that $\mathcal{A}$ is not allowed from submitting the challenge ciphertext after receiving it), or a valid ciphertext $(\mathsf{vk}^*, c, \pi, \phi, \sigma)$ to the $\mathsf{Dec}_{\mathrm{PKE}}(sk_1, \cdot)$ oracle, or a valid ciphertext $(\mathsf{vk}^*, c, \pi, \phi, \sigma)$ to the $\mathsf{Dec}_{\mathrm{PKE}}(sk_2, \cdot)$ oracle. It is easy to check that as long as Forge did not occur then the simulation of the decapsulation oracle is perfect. The probability that the above event occurs can be shown to be bounded by $\mathbf{Adv}^{\mathrm{suf}}_{\mathcal{OT},\mathcal{C}}(k)$, where $\mathcal{C}$ is an adversary against the strong unforgeability of one-time signature $\mathcal{OT}$. Following a standard argument, we can show that

$$\mathbf{Adv}^{\mathrm{bddh}}_{\mathcal{BG},\mathcal{B}}(k) \geq \mathbf{Adv}^{\mathrm{cca}}_{\mathcal{CE},1,\mathcal{A}}(k) - \mathbf{Adv}^{\mathrm{suf}}_{\mathcal{OT},\mathcal{C}}(k).$$

We now consider *PKE security with DRE decryption oracle*. Similar to the above one, the point is to show how to simulate the unrestricted DRE oracle for any ciphertext.

Suppose there exists a polynomial time adversary $\mathcal{A}$ that breaks for the $\mathcal{CE}$ the PKE security with DRE decryption oracle with (non-negligible) advantage $\mathbf{Adv}^{\mathrm{cca}}_{\mathcal{CE},2,\mathcal{A}}(k)$. We show that there exist adversaries $\mathcal{B}$ against a random instance of the BDDH problem and $\mathcal{C}$ against the strong unforgeability of one-time signature $\mathcal{OT}$ such that

$$\mathbf{Adv}^{\mathrm{bddh}}_{\mathcal{BG},\mathcal{B}}(k) \geq \mathbf{Adv}^{\mathrm{cca}}_{\mathcal{CE},1,\mathcal{A}}(k) - \mathbf{Adv}^{\mathrm{suf}}_{\mathcal{OT},\mathcal{C}}(k).$$

Adversary $\mathcal{B}$ is given as input a random BDDH instance $(g^{a_1}, g^{a_2}, g^r, K)$ for a symmetric bilinear group $\mathcal{BG} = (q, \mathbb{G}, \mathbb{G}_T, e, g)$. Again, the goal of adversary $\mathcal{B}$ is to determine whether $K = e(g,g)^{a_1 a_2 r}$ or K is a random element in $\mathbb{G}_T$. Adversary $\mathcal{B}$ simulates adversary $\mathcal{A}$'s view as follows:

Adversary $\mathcal{B}$ runs $\mathsf{Gen}_{\mathrm{OT}}(1^k)$ to return a verification/signing key pair $(\mathsf{vk}^*, \mathsf{sk}^*)$. It then randomly selects d_1 from $\mathbb{Z}_q^*$ and defines the public key for user 1

$$(u_1 = g^{a_1}, \quad v_1 = u_1^{-\mathsf{vk}^*} \cdot g^{d_1}).$$

The public key is distributed just as in the DKEM experiment.

We show how adversary $\mathcal{B}$ can simulate the decryption oracles by adversary $\mathcal{A}$. Adversary $\mathcal{B}$ rejects all the DRE and PKE ciphertexts that are not consistency, just as in the original experiment. Otherwise, we show how adversary $\mathcal{B}$ to answer all three types of decryption oracles:

For any PKE ciphertext $C = (\mathsf{vk}, c, \pi, \phi, \sigma)$ to $\mathsf{Dec}_{\mathrm{PKE}}(sk_1, \cdot)$, if $\mathsf{vk} = \mathsf{vk}^*$ then it simply aborts; otherwise, it computes $(\pi_1/c^{d_1})^{(t-t^*)^{-1}}$ and returns $\phi/e((\pi_1/c^{d_1})^{(t-t^*)^{-1}}, g^{a_2})$.

For any DRE ciphertext $C = (\mathsf{vk}, c, \pi_1, \pi_2, \phi, \sigma)$ to $\mathsf{Dec}_{\mathrm{DRE}}(sk_1, pk_1, pk, \cdot)$ for some valid public key $pk = (u, v)$, if $\mathsf{vk} = \mathsf{vk}^*$ then it simply aborts; otherwise, it computes $(\pi_1/c^{d_1})^{(t-t^*)^{-1}}$ and returns $\phi/e((\pi_1/c^{d_1})^{(t-t^*)^{-1}}, u)$.

For any DRE ciphertext $C = (\mathsf{vk}, c, \pi_1, \pi_2, \phi, \sigma)$ to $\mathsf{Dec}_{\mathrm{DRE}}(sk_1, pk', pk_1, \cdot)$ for some valid public key $pk' = (u', v')$, if $\mathsf{vk} = \mathsf{vk}^*$ then it simply aborts; otherwise, it computes $(\pi_1/c^{d_1})^{(t-t^*)^{-1}}$ and returns $\phi/e((\pi_1/c^{d_1})^{(t-t^*)^{-1}}, u')$.

At some point, adversary $\mathcal{A}$ outputs two messages M_0 and M_1. Adversary $\mathcal{B}$ provides $\mathcal{A}$ with the following challenge ciphertext:

$$C^* = (\mathsf{vk}^*,\ c^*,\ \pi^*,\ \phi^*,\ \sigma^*).$$

where, above, $c^* = g^r, \pi^* = (g^r)^{d_1}, \phi^* = K \cdot M_b$, and $\mathsf{Sig}_{\mathsf{sk}^*}(c^*, \pi^*)$.

Since $(u_1^{\mathsf{vk}^*} v_1)^r = (g^r)^{d_1}$, the ciphertext is a valid challenge ciphertext for unknown randomness r.

Adversary $\mathcal{A}$ may further make decryption oracle queries with the only restriction that it may not query $\mathsf{Dec}_{\mathrm{PKE}}(sk_1, C^*)$ after receiving the challenge ciphertext C^*.

Finally, adversary $\mathcal{A}$ outputs a guess b'; adversary $\mathcal{B}$ outputs the same guess.

Let Forge be the event that $\mathcal{A}$ submits a valid ciphertext $(\mathsf{vk}^*, c, \pi, \phi, \sigma)$ to the $\mathsf{Dec}_{\mathrm{PKE}}(sk_1, \cdot)$ oracle (recall that $\mathcal{A}$ is not allowed from submitting the challenge ciphertext after receiving it), or a valid ciphertext $(\mathsf{vk}^*, c, \pi_1, \pi, \phi, \sigma)$ to the $\mathsf{Dec}_{\mathrm{DRE}}(sk_1, pk_1, pk, \cdot)$ oracle, where pk is valid and $pk_1 <^d pk$, or a valid ciphertext $(\mathsf{vk}^*, c, \pi_1, \pi, \phi, \sigma)$ to the $\mathsf{Dec}_{\mathrm{DRE}}(sk_1, pk', pk_1, \cdot)$ oracle, where pk is valid and $pk' <^d pk_1$. It is easy to check that as long as Forge did not occur then the simulation of the decapsulation oracle is perfect. The probability that the above event occurs can be shown to be bounded by $\mathbf{Adv}^{\mathrm{suf}}_{\mathcal{OT},\mathcal{C}}(k)$, where $\mathcal{C}$ is an adversary against the strong unforgeability of one-time signature $\mathcal{OT}$. Following a standard argument, we can show that

$$\mathbf{Adv}^{\mathrm{bddh}}_{\mathcal{BG},\mathcal{B}}(k) \geq \mathbf{Adv}^{\mathrm{cca}}_{\mathcal{CE},2,\mathcal{A}}(k) - \mathbf{Adv}^{\mathrm{suf}}_{\mathcal{OT},\mathcal{C}}(k).$$

∎

F Proof of Theorem 4

Proof: We sketch the proof via a sequence of games. The games involve the challenger and an adversary $\mathcal{A}$. Let T_i be the event that the challenger outputs 1 in Game i.

Game 0. Let Game 0 be the **Experiment** $\mathbf{Exp}^{\text{cnm-cca-0}}_{\mathcal{DRE},\mathcal{A}}(k)$. The adversary $\mathcal{A}$ is given the common reference string crs and the public keys pk_1 and pk_2. The challenger answers the decryption oracles for adversary $\mathcal{A}$. After adversary $\mathcal{A}$ choosing a distribution M, the challenger selects $m_0 \xleftarrow{\$} \mathsf{M}$, and returns (c_1, c_2, π), where $c_1 \leftarrow \mathsf{Enc}(pk_1, m_0; r_1)$, $c_2 \leftarrow \mathsf{Enc}(pk_2, m_0; r_2)$, and $\pi \xleftarrow{\$} \mathsf{P}(\mathsf{crs}, (c_1, c_2, pk_1, pk_2), (m_0, r_1, r_2))$ for some random r_1 and r_2. Finally adversary $\mathcal{A}$ outputs $(\mathtt{R}, pk_1^*, pk_2^*, \mathbf{c}^*)$. The challenger checks whether the public keys pk_1^* and pk_2^* are valid. (Note that we ask the encryption to be admissible.) The challenger then computes the underlying secret key of either pk_1^* or pk_2^* and decrypts $\mathbf{c}^*$ to $\mathbf{m}^*$. The challenger returns 1 if $\mathtt{R}(m_0, \mathbf{m}^*, \mathsf{crs}, pk_1, pk_2, pk_1^*, pk_2^*, \mathbf{c}^*) = 1$ and returns 0 otherwise.

Game 1. Let Game 1 be as Game 0, except that after adversary $\mathcal{A}$ chooses the distribution the challenger selects $m_0, m_1 \xleftarrow{\$} \mathsf{M}$ (instead of m_0 only). Note that challenger still encrypts m_0 as the challenge ciphertext. Since m_1 is never given to the adversary $\mathcal{A}$, the difference is only conceptual from the perspective of the adversary.

Game 2. Let Game 2 be as Game 1, except that when responding to the challenge plaintext, the challenger uses a simulated proof. More formally, the challenger runs $(\mathsf{crs}, \tau, \mathsf{ek}) \xleftarrow{\$} \mathsf{Gen}_{\mathrm{unite}}(1^k)$ to prepare the common reference string crs, and keeps the simulation trapdoor τ and extraction key ek. It then computes $c_1 \leftarrow \mathsf{Enc}(pk_1, m_0; r_1)$ and $c_2 \leftarrow \mathsf{Enc}(pk_2, m_0; r_2)$ for some random r_1 and r_2, but computes a simulation proof π using the simulation trapdoor τ such that $(c_1, c_2, pk_1, pk_2) \in \mathcal{L}_1$.

It follows from the zero-knowledge property that except with negligible probability the adversary $\mathcal{A}$ cannot distinguish Game 2 from Game 1.

Game 3. Let Game 2 be as Game 1 with the following difference. For final output $(\mathbf{c}^*, pk_1^*, pk_2^*)$ by the adversary $\mathcal{A}$ where $\mathbf{c}^* = (c_1^*, c_2^*, \pi^*)$, if $\mathsf{V}(c_1^*, c_2^*, \pi^*) = 1$ then the challenger uses the extraction key ek to extract the witness $\mathbf{m}^*$. Note that the challenger still generates the public/secret key pairs (pk_1, sk_2) and (pk_2, sk_2). For any decryption oracles, it simply uses sk_1 to decrypt the ciphertexts.

It follows from the simulation sound extractability property that except with negligible probability the adversary $\mathcal{A}$ cannot distinguish Game 3 from Game 2.

Game 4. Let Game 4 be as Game 3 with the following difference. When answering the encryption query, the challenger computes $c_1 \leftarrow \mathsf{Enc}(pk_1, m_0; r_1)$ and $c_2 \leftarrow \mathsf{Enc}(pk_2, m_1; r_2)$ for some random r_1 and r_2, and simulates the proof π using the simulation trapdoor τ.

It follows from the IND-CPA security of encryption for the second receiver with public key pk_2 that the adversary $\mathcal{A}$ cannot distinguish Game 4 from Game 3 except with negligible probability.

Game 5. Let Game 5 be as Game 4 with the following difference. When answering the decryption query, the challenger uses sk_2 to decrypt the ciphertext. The adversary $\mathcal{A}$ would not notice the difference if the adversary could not query the decryption oracle with an *invalid* ciphertext (i.e., ciphertext that contains encryptions of different messages).

It follows from the simulation soundness property for the proof system that adversary $\mathcal{A}$ cannot produce invalid ciphertext except with negligible probability. Therefore, adversary $\mathcal{A}$ cannot distinguish Game 5 from Game 4 with noticeable probability.

Game 6. Let Game 6 be as Game 5 with the following difference. When answering the encryption query, the challenger computes $c_1 \leftarrow \mathsf{Enc}(pk_1, m_1; r_1)$ and $c_2 \leftarrow \mathsf{Enc}(pk_2, m_1; r_2)$ for some random r_1 and r_2, and simulates the proof π using the simulation trapdoor τ.

It follows from the IND-CPA security of encryption for the first receiver with public key pk_1 that the adversary $\mathcal{A}$ cannot distinguish Game 6 from Game 5 except with negligible probability.

Game 7. Let Game 7 be as Game 6, except that the challenger computes the underlying secret keys of the target public keys to answer the challenge ciphertext.

It follows from the simulation sound extractability property that except with negligible probability the adversary $\mathcal{A}$ cannot distinguish Game 7 from Game 6.

Game 8. Let Game 8 be as Game 7, except that when responding to the challenge plaintext, the challenger uses a real proof.

It follows from the zero-knowledge property that except with negligible probability the adversary $\mathcal{A}$ cannot distinguish Game 8 from Game 7.

Game 9. Let Game 9 be as Game 8 with the following difference. When answering the decryption query, the challenger uses sk_1 to decrypt the ciphertext.

It follows from the soundness property for the proof system that adversary $\mathcal{A}$ cannot distinguish Game 9 from Game 8 with noticeable probability.

One can check that Game 9 behaves exactly as **Experiment** $\mathbf{Exp}^{\text{cnm-cca-1}}_{\mathcal{DRE},\mathcal{A}}(k)$ involving the challenger and adversary $\mathcal{A}$.

Combining the probability results, we essentially get that $\Pr[T_9] - \Pr[T_0]$ is negligible. The theorem now follows. ▮