On the Equivalence of Several Security Notions of Key Encapsulation Mechanism

Waka Nagao[1] Yoshifumi Manabe[1,2] Tatsuaki Okamoto[1,2]

[1] Graduate School of Informatics, Kyoto University
Yoshida-honmachi, Kyoto, 606-8501 Japan
w-nagao@ai.soc.i.kyoto-u.ac.jp
[2] NTT Labs, Nippon Telegraph and Telephone Corporation
1-1 Hikari-no-oka, Yokosuka, 239-0847 Japan
{manabe.yoshifumi,okamoto.tatsuaki}@lab.ntt.co.jp

August 12, 2006

Abstract. KEM (Key Encapsulation Mechanism) was introduced by Shoup to formalize the asymmetric encryption specified for key distribution in ISO standards on public-key encryption. Shoup defined the “semantic security (IND) against adaptively chosen ciphertext attacks (CCA2)” as a desirable security notion of KEM. This paper introduces ”non- malleability (NM)” of KEM, a stronger security notion than IND. We provide three definitions of NM, and show that these three definitions are equivalent. We then show that NM-CCA2 KEM is equivalent to IND-CCA2 KEM. That is, we show that NM is equivalent to IND under CCA2 attacks, although NM is stronger than IND in the definition (or under some attacks like CCA1). In addition, this paper defines the universally composable (UC) security of KEM and shows that NM-CCA2 KEM is equivalent to UC KEM.

1 Introduction

The Key Encapsulation Mechanism (KEM), a key distribution mechanism in public-key cryptosystems, was proposed by Shoup for ISO standards on public-key encryption [8]. The difference between KEM and public-key encryption (PKE) is as follows: PKE’s encryption procedure, on input plaintext M and receiver R’s public-key PK_R, outputs ciphertext C, while KEM’s encryption procedure, on input receiver R’s public-key PK_R, outputs ciphertext C and key K, where C is sent to R, and K is kept secret inside the sender, and employed in the subsequent process of data encryption. PKE’s decryption procedure, on input C and secret-key SK_R, outputs plaintext M, while KEM’s decryption procedure, on input C and secret-key SK_R, outputs key K. Although KEM is a mechanism for key distribution and the applications of KEM are not specified, the most typical application is hybrid encryption, where a key shared via KEM is employed for symmetric-key encryption.

Shoup defined the security, “indistinguishable (or semantically secure) (IND) against adaptively chosen-ciphertext attacks (CCA2),” for KEM. Although this security notion is considered to be feasible for KEM, we may define a stronger security notion than Shoup’s, and such a stronger security notion could be more feasible for KEM.

In this paper, we investigate two stronger security notions for KEM. One is “non- malleability (NM)” and the other is “universal composability (UC)”.

NM was introduced for PKE [4, 1, 2] as a stronger security notion than IND, but a straightforwardly analogous definition of NM for KEM is not successful [3], since the message space of PKE can be flexibly specified, while the key space of KEM is, in general, hard to specify flexibly (i.e., it may be hard to restrict the output of the encryption function of KEM into a small key space).

This paper gives the first feasible (three) definitions of NM for KEM; they are not so straightforwardly analogous to those of NM for PKE, and no key space is treated explicitly in our definitions. We then show that these three definitions are equivalent.

[3] A straightforwardly analogous definition of NM for KEM in [6, 7] has a problem in the treatment of the key space.

It is easily obtained from one of the definitions of NM that NM-CCA2 KEM is equivalent to IND-CCA2 KEM. That is, we can now realize that Shoup's definition, IND-CCA2, is as feasible as NM-CCA2, whereas NM is stronger than IND in the definition.

In addition, this paper investigates another stronger definition, the universally composable (UC) of KEM. The framework of UC was introduced by Canetti [3] and it guarantees very strong security, i.e., preserves stand-alone security in any type of composition with other primitives and protocols.

This paper defines the UC security of KEM, i.e., the ideal functionality of KEM. We then show that NM-CCA2 (i.e., IND-CCA2) KEM is equivalent to UC KEM.

Remark: Very recently, a weaker security notion of non-malleability than our NM definitions has been introduced and investigated in [5].

2 Preliminaries

2.1 Notations

$\mathbb{N}$ is the set of natural numbers and $\mathbb{R}$ is the set of real numbers. $\perp$ denotes the null string.

A function $f : \mathbb{N} \rightarrow \mathbb{R}$ is negligible in k, if for every constant $c > 0$, there exists integer k_c such that $f(k) < k^{-c}$ for all $k > k_c$. Hereafter, we often use $f < \epsilon(k)$ to mean that f is negligible in k. On the other hand, we use $f > \mu(k)$ to mean that f is not negligible in k. i.e., function $f : \mathbb{N} \rightarrow \mathbb{R}$ is not negligible in k, if there exists a constant $c > 0$ such that for every integer k_c, there exists $k > k_c$ such that $f(k) > k^{-c}$.

When A is a probabilistic machine or algorithm, $A(x)$ denotes the random variable of A's output on input x. Then, $y \overset{\mathsf{R}}{\leftarrow} A(x)$ denotes that y is randomly selected from $A(x)$ according to its distribution. When A is a set, $y \overset{\mathsf{U}}{\leftarrow} A$ denotes that y is uniformly selected from A. When A is a value, $y \leftarrow A$ denotes that y is set as A.

We write vectors in boldface, as in $\boldsymbol{x}$. We also denote the number of components in $\boldsymbol{x}$ by $|\boldsymbol{x}|$, and the i-th component by $\boldsymbol{x}[i]$, so that $\boldsymbol{x}$ = ($\boldsymbol{x}[1], \cdots, \boldsymbol{x}[|\boldsymbol{x}|]$). We denote a component of a vector as x $\in \boldsymbol{x}$ or x $\notin \boldsymbol{x}$, which means, respectively, that x is in or is not in the set $\{ \boldsymbol{x}[i] : 1 \leq i \leq |\boldsymbol{x}|\}$. We can simply write $\boldsymbol{x} \leftarrow \mathcal{D}(\boldsymbol{y})$ as the shorthand form of $1 \leq i \leq |\ \boldsymbol{y}\ |\ \boldsymbol{x}[i] \leftarrow \mathcal{D}(\boldsymbol{y}[i])$. We will consider a relation, Rel, of t variables. Rather than writing $Rel(x_1, \cdots, x_t)$, we write $Rel(x, \boldsymbol{x})$, meaning the first argument is special and the rest are bunched into vector $\boldsymbol{x}$ with $|\boldsymbol{x}|$ = $t - 1$.

2.2 Key Encapsulation Mechanism

Definition of Key Encapsulation Mechanism We recall the standard notion of key encapsulation mechanism, KEM, which was formalized by Shoup in [8]. A KEM scheme is the triple of algorithms, $\Sigma = (\mathcal{G}, \mathcal{E}, \mathcal{D})$, where

1. $\mathcal{G}$, the key generation algorithm, is a probabilistic polynomial time (PPT) algorithm that takes a security parameter $k \in \mathbb{N}$ (provided in unary) and returns a pair (pk, sk) of matching public and secret keys.
2. $\mathcal{E}$, the key encryption algorithm, is a PPT algorithm that takes as input public key pk and outputs a key/ciphertext pair (K^*, C^*).
3. $\mathcal{D}$, the decryption algorithm, is a deterministic polynomial time algorithm that takes as input secret key sk and ciphertext C^*, and outputs key K^* or $\perp$ ($\perp$ implies that the ciphertext is invalid).

We require that for all (pk, sk) output by key generation algorithm $\mathcal{G}$ and for all (K^*, C^*) output by key encryption algorithm $\mathcal{E}(pk)$, $\mathcal{D}(sk, C^*) = K^*$ holds. Here, the length of the key, $|K^*|$, is specified by $l(k)$, where k is the security parameter.

Attack types of KEM From the standard notion of attack types, we consider the following three attack types of KEM; CPA, CCA1, and CCA2. CPA means "Chosen Plaintext Attacks," where an adversary is allowed to access only an encryption oracle, not any decryption oracle. CCA1 means "Chosen Ciphertext Attacks," where an adversary is allowed to access both encryption and decryption oracles, but the adversary cannot access the decryption oracle after getting the target ciphertext. CCA2 means "Adaptive Chosen Ciphertext Attacks," where an adversary is allowed to access both encryption and decryption oracles even after the adversary is given the target ciphertext.

Definition of Indistinguishability for KEM The indistinguishability (IND) of KEM was defined by Shoup [8]. We use “IND-ATK-KEM” to describe the security notion of indistinguishability for KEM against ATK $\in \{CPA, CCA1, CCA2\}$. “IND-KEM” is used to focus on the indistinguishability of KEM without regard to attack type. If it is clear from the context that IND-ATK-KEM (and IND-KEM) is used for KEM, we will call it IND-ATK (and IND) for simplicity.

To clarify the indistinguishability of public key encryption (PKE), we may use IND-ATK-PKE and IND-PKE.

$$\mathtt{Adv}^{\text{IND-ATK}}_{A,\Sigma}(k) \equiv \Pr[\mathtt{Expt}^{\text{IND-ATK}}_{A,\Sigma}(k) = 1] - \frac{1}{2},$$

where
$\mathtt{Expt}^{\text{IND-ATK}}_{A,\Sigma}(k)$

$$\begin{array}{c} (pk, sk) \overset{\mathsf{R}}{\leftarrow} \mathcal{G}(1^k);\ s \overset{\mathsf{R}}{\leftarrow} A_1^{O_1}(pk); \\ (K^*, C^*) \overset{\mathsf{R}}{\leftarrow} \mathcal{E}(pk);\ R \overset{\mathsf{U}}{\leftarrow} \{0,1\}^{l(k)};\ b \overset{\mathsf{U}}{\leftarrow} \{0,1\}; \\ X \leftarrow \begin{cases} K^*, \text{ if } \ b = 0 \\ R, \text{ if } \ b = 1 \end{cases} \\ g \overset{\mathsf{R}}{\leftarrow} A_2^{O_2}(s, X, C^*) \\ \text{return 1, iff} \quad g = b \end{array}$$

and
If ATK = CPA, then $O_1 = \perp$ and $O_2 = \perp$.
If ATK = CCA1, then $O_1 = \mathcal{D}(sk, \cdot)$ and $O_2 = \perp$.
If ATK = CCA2, then $O_1 = \mathcal{D}(sk, \cdot)$ and $O_2 = \mathcal{D}(sk, \cdot)$.

Fig. 1. Advantage of IND-ATK-KEM

Definition 1. *Let Σ be a* KEM, $A = (A_1, A_2)$ *be an adversary, and $k \in \mathbb{N}$ be a security parameter. For ATK $\in \{CPA, CCA1, CCA2\}$, $\mathtt{Adv}^{\text{IND-ATK}}_{A,\Sigma}(k)$ is defined in Fig. 1. We say that Σ is IND-ATK-KEM, if for any adversary $A \in \mathcal{P}$, $\mathtt{Adv}^{\text{IND-ATK}}_{A,\Sigma}(k)$ is negligible in k, where ATK $\in \{CPA, CCA1, CCA2\}$, and $\mathcal{P}$ denotes a class of polynomial-time bounded machines.*

2.3 Universal Composability

The notion of universal composability (UC) was introduced by Canetti [3]. In this notion, we consider the real life world and the ideal process world. In the real life world, there are an adversary A and a protocol π which realizes a functionality among some parties. On the other hand, in the ideal process world, there are a simulator S that simulates the real life world, an ideal functionality $\mathcal{F}$, and dummy parties. We consider an environment Z which tries to distinguish the real life world from the ideal process world.

Informally, we describe the universally composable security notion as follows: (For more details, see [3].)

The Real Life World Let $\mathtt{REAL}_{\pi,A,Z}(k, z, \boldsymbol{r})$ denote the output of environment Z when interacting with adversary A and parties P_1, …, P_n running protocol π on security parameter k, input z and random input $\boldsymbol{r}$ = $(r_Z$, r_A, $r_1 \ldots r_n)$ (z and r_Z for Z, r_A for A, r_i for party P_i). Let $\mathtt{REAL}_{\pi,A,Z}(k, z)$ denote the random variable describing $\mathtt{REAL}_{\pi,A,Z}(k, z, \boldsymbol{r})$ when $\boldsymbol{r}$ is uniformly chosen.

The Ideal Process World Let $\mathtt{IDEAL}_{\mathcal{F},S,Z}(k, z, \boldsymbol{r})$ denote the output of environment Z after interacting in the ideal process world with adversary S and ideal functionality $\mathcal{F}$, on security parameter k, input z, and random input $\boldsymbol{r}$ = (r_Z, r_S, r_F) (z and r_Z for Z, r_S for S, r_F for $\mathcal{F}$). Let $\mathtt{IDEAL}_{\mathcal{F},S,Z}(k, z)$ denote the random variable describing $\mathtt{IDEAL}_{\mathcal{F},S,Z}(k, z, \boldsymbol{r})$ when $\boldsymbol{r}$ is uniformly chosen.

The Security Framework of UC Let $\mathcal{F}$ be an ideal functionality and let π be a protocol. We say that π UC-realizes $\mathcal{F}$, if for any adversary $A \in \mathcal{P}$ there exists a simulator $S \in \mathcal{P}$ such that for any environment Z $\in \mathcal{P}$,

$$\mathtt{IDEAL}_{\mathcal{F},S,Z}(k, z) \approx \mathtt{REAL}_{\pi,A,Z}(k, z),$$

where $\approx$ denotes statistically indistinguishable in k and $\mathcal{P}$ denotes a class of polynomial-time bounded machines.

3 Three Non-Malleability Definitions of KEM

3.1 Definition of SNM-ATK-KEM

KEM Σ is called "SNM-ATK-KEM" in the sense that Σ is secure in the *simulation based non-malleability* (SNM) for each attack type ATK $\in \{CPA, CCA1, CCA2\}$.

Definition 2. *Let Σ be* KEM*, Rel be a relation, $A = (A_1, A_2)$ be an adversary, $S = (S_1, S_2)$ be an algorithm (the "simulator"), and $k \in \mathbb{N}$ be a security parameter. For ATK $\in \{CPA, CCA1, CCA2\}$, we define $\mathtt{Adv}^{\mathrm{SNM\text{-}ATK}}_{A,S,\Sigma}(Rel, k)$ in Fig. 2. We say that Σ is SNM-ATK-KEM, if for any adversary $A \in \mathcal{P}$ and all relations Rel computable in $\mathcal{P}$, there exists simulator $S \in \mathcal{P}$ such that $\mathtt{Adv}^{\mathrm{SNM\text{-}ATK}}_{A,S,\Sigma}(Rel, k)$ is negligible in k , where ATK $\in \{CPA, CCA1, CCA2\}$, and $\mathcal{P}$ denotes a class of polynomial-time bounded machines.*

Note that adversary A_2 is not allowed to pose the challenge ciphertext C^* to its decryption oracle in the case of CCA2.

In the attack scenario of SNM for public key encryption (PKE), SNM-PKE, the adversary can decide the message space [2]. Note that such a message space in the scenario is introduced to make SNM-PKE to be compatible with IND-PKE (i.e., to make SNM-PKE to imply IND-PKE), in whose attack scenario the adversary can decide a pair of messages (a message space).

In contrast, in the attack scenario of IND-KEM, a correct key or a random value along with the target ciphertext is given to the adversary. To make SNM-KEM compatible with IND-KEM, our SNM-KEM's attack scenario gives the adversary a randomly-ordered pair of a correct key and a random value.

Two additional minor differences between SNM-KEM and SNM-PKE are:

1. Simulator S also gets access to the decryption oracle when ATK allows to do so.
2. Relation R takes state information s calculated not by A_1 or S_1 but by A_2 or S_2 in SNM-KEM.

3.2 CNM-ATK-KEM

A KEM Σ is called "CNM-ATK-KEM" in the sense that Σ is secure in the *comparison based non-malleability* (CNM) for each attack type ATK $\in \{CPA, CCA1, CCA2\}$.

Definition 3. *Let Σ be* KEM*, $A = (A_1, A_2)$ be an adversary, and $k \in \mathbb{N}$ be a security parameter. For ATK $\in \{CPA, CCA1, CCA2\}$, we define $\mathtt{Adv}^{\mathrm{CNM\text{-}ATK}}_{A,\Sigma}(k)$ in Fig. 3. We say that Σ is CNM-ATK-KEM, if for any adversary $A \in \mathcal{P}$, $\mathtt{Adv}^{\mathrm{CNM\text{-}ATK}}_{A,\Sigma}(k)$ is negligible in k, where ATK $\in \{CPA, CCA1, CCA2\}$, and $\mathcal{P}$ denotes a class of polynomial-time bounded machines.*

Note that adversary A_2 is not allowed to ask its oracle to decrypt the challenge ciphertext C^* in the case of CCA2.

Similar to SNM-KEM, our CNM-KEM's attack scenario gives the adversary a randomly-ordered pair of a correct key and a random value to make CNM-KEM compatible with IND-KEM.

$$\mathtt{Adv}^{\text{SNM-ATK}}_{A,S,\Sigma}(Rel,k) \equiv$$

$$\Pr[\mathtt{Expt}^{\text{SNM-ATK}}_{A,\Sigma}(Rel,k)=1] - \Pr[\mathtt{Expt}^{\text{SNM-ATK}}_{S,\Sigma}(Rel,k)=1],$$

where

```
$\mathtt{Expt}^{\text{SNM-ATK}}_{A,\Sigma}(Rel,k)$

$(pk,sk) \xleftarrow{\mathsf{R}} \mathcal{G}(1^k)$
$s_1 \xleftarrow{\mathsf{R}} A_1^{O_1}(pk)$
$(K^*,C^*) \xleftarrow{\mathsf{R}} \mathcal{E}(pk)$
$R \xleftarrow{\mathsf{U}} \{0,1\}^{l(k)}$
$b \xleftarrow{\mathsf{U}} \{0,1\}$
$X \leftarrow (r_0,r_1)$, where
    if $b=0$, then $r_0 \leftarrow K^*$ and $r_1 \leftarrow R$
    if $b=1$, then $r_0 \leftarrow R$ and $r_1 \leftarrow K^*$
$(s_2,\boldsymbol{C}) \xleftarrow{\mathsf{R}} A_2^{O_2}(X,s_1,C^*)$
$\boldsymbol{K} \leftarrow \mathcal{D}(sk,\boldsymbol{C})$
return 1, iff $(C^* \notin \boldsymbol{C}) \wedge Rel(K^*,\boldsymbol{K},s_2)$
```

```
$\mathtt{Expt}^{\text{SNM-ATK}}_{S,\Sigma}(Rel,k)$

$(pk,sk) \xleftarrow{\mathsf{R}} \mathcal{G}(1^k)$
$s_1 \xleftarrow{\mathsf{R}} S_1^{O_1}(pk)$
$R^* \xleftarrow{\mathsf{U}} \{0,1\}^{l(k)}$
$R \xleftarrow{\mathsf{U}} \{0,1\}^{l(k)}$
$b \xleftarrow{\mathsf{U}} \{0,1\}$
$X \leftarrow (r_0,r_1)$, where
    if $b=0$, then $r_0 \leftarrow R^*$ and $r_1 \leftarrow R$
    if $b=1$, then $r_0 \leftarrow R$ and $r_1 \leftarrow R^*$
$(s_2,\boldsymbol{C}) \xleftarrow{\mathsf{R}} S_2^{O_2}(X,s_1)$
$\boldsymbol{K} \leftarrow \mathcal{D}(sk,\boldsymbol{C})$
return 1, iff $Rel(R^*,\boldsymbol{K},s_2)$
```

and
If ATK = CPA, then $O_1 = \perp$ and $O_2 = \perp$.
If ATK = CCA1, then $O_1 = \mathcal{D}(sk,\cdot)$ and $O_2 = \perp$.
If ATK = CCA2, then $O_1 = \mathcal{D}(sk,\cdot)$ and $O_2 = \mathcal{D}(sk,\cdot)$.

Fig. 2. Advantage of SNM-ATK-KEM

3.3 PNM-ATK-KEM

KEM Σ is called "PNM-ATK-KEM" in the sense that Σ is secure in the *parallel chosen-ciphertext attack based non-malleability* (PNM) for each attack type ATK $\in \{CPA, CCA1, CCA2\}$.

Definition 4. *Let Σ be a* KEM, *$A = (A_1, A_2, A_3)$ be an adversary, and $k \in \mathbb{N}$ be a security parameter. For ATK $\in \{CPA, CCA1, CCA2\}$, we define $\mathtt{Adv}^{\text{PNM-ATK}}_{A,\Sigma}(k)$ in Fig. 4. We say that Σ is PNM-ATK-KEM, if for any adversary $A \in \mathcal{P}$, $\mathtt{Adv}^{\text{PNM-ATK}}_{A,\Sigma}(k)$ is negligible in k, where k is a security parameter, ATK $\in \{CPA, CCA1, CCA2\}$, and $\mathcal{P}$ denotes a class of polynomial-time bounded machines.*

Note that adversary A_2 is not allowed to ask its oracle to decrypt the challenge ciphertext C^* in the case of CCA2.

In the PNM defintion, the non-malleability property is captured by the indistinguishability under the parallel chosen-ciphertext attack such that A_2 outputs a vector of ciphertext $\boldsymbol{C}$ and its decryption result $\boldsymbol{K}$ is given to A_3.

4 Equivalence of the Three Non-Malleability Definitions

Here, we prove the equivalence of the three non-malleability definitions.

Theorem 1 *For any ATK $\in \{CPA, CCA1, CCA2\}$, if* KEM *Σ is CNM-ATK-KEM, then Σ is SNM-ATK-KEM.*

Theorem 2 *For any ATK $\in \{CPA, CCA1, CCA2\}$, if* KEM *Σ is SNM-ATK-KEM, then Σ is PNM-ATK-KEM.*

Theorem 3 *For any ATK $\in \{CPA, CCA1, CCA2\}$, if* KEM *Σ is PNM-ATK-KEM, then Σ is CNM-ATK-KEM.*

$$\mathtt{Adv}^{\text{CNM-ATK}}_{A,\Sigma}(k) \equiv \Pr[\mathtt{Expt}^{\text{CNM-ATK}}_{A,\Sigma}(k) = 1] - \Pr[\widetilde{\mathtt{Expt}}^{\text{CNM-ATK}}_{A,\Sigma}(k) = 1],$$

where

```
$\mathtt{Expt}^{\text{CNM-ATK}}_{A,\Sigma}(k)$

$(pk, sk) \overset{\mathsf{R}}{\leftarrow} \mathcal{G}(1^k)$
$s \overset{\mathsf{R}}{\leftarrow} A_1^{O_1}(pk)$
$(K^*, C^*) \overset{\mathsf{R}}{\leftarrow} \mathcal{E}(pk)$

$R \overset{\mathsf{U}}{\leftarrow} \{0,1\}^{l(k)}$
$b \overset{\mathsf{U}}{\leftarrow} \{0,1\}$
$X \leftarrow (r_0, r_1)$, where
    if $b = 0$, then $r_0 \leftarrow K^*$ and $r_1 \leftarrow R$
    if $b = 1$, then $r_0 \leftarrow R$ and $r_1 \leftarrow K^*$
$(Rel, \boldsymbol{C}) \overset{\mathsf{R}}{\leftarrow} A_2^{O_2}(X, s, C^*)$
$\boldsymbol{K} \leftarrow \mathcal{D}(sk, \boldsymbol{C})$
return 1, iff $(C^* \notin \boldsymbol{C}) \wedge Rel(K^*, \boldsymbol{K})$
```

```
$\widetilde{\mathtt{Expt}}^{\text{CNM-ATK}}_{A,\Sigma}(k)$

$(pk, sk) \overset{\mathsf{R}}{\leftarrow} \mathcal{G}(1^k)$
$s \overset{\mathsf{R}}{\leftarrow} A_1^{O_1}(pk)$
$(K^*, C^*) \overset{\mathsf{R}}{\leftarrow} \mathcal{E}(pk)$
$R^* \overset{\mathsf{U}}{\leftarrow} \{0,1\}^{l(k)}$
$R \overset{\mathsf{U}}{\leftarrow} \{0,1\}^{l(k)}$
$b \overset{\mathsf{U}}{\leftarrow} \{0,1\}$
$X \leftarrow (r_0, r_1)$, where
    if $b = 0$, then $r_0 \leftarrow R^*$ and $r_1 \leftarrow R$
    if $b = 1$, then $r_0 \leftarrow R$ and $r_1 \leftarrow R^*$
$(Rel, \boldsymbol{C}) \overset{\mathsf{R}}{\leftarrow} A_2^{O_2}(X, s, C^*)$
$\boldsymbol{K} \leftarrow \mathcal{D}(sk, \boldsymbol{C})$
return 1, iff $(C^* \notin \boldsymbol{C}) \wedge Rel(R^*, \boldsymbol{K})$
```

and
If ATK = CPA, then $O_1 = \perp$ and $O_2 = \perp$.
If ATK = CCA1, then $O_1 = \mathcal{D}(sk, \cdot)$ and $O_2 = \perp$.
If ATK = CCA2, then $O_1 = \mathcal{D}(sk, \cdot)$ and $O_2 = \mathcal{D}(sk, \cdot)$.

Fig. 3. Advantage of CNM-ATK-KEM

4.1 Proof of Theorem 1:

Proof. We prove that KEM Σ is not CNM-ATK-KEM if Σ is not SNM-ATK-KEM. More precisely, we show that if there exist adversary A and relation Rel such that $\mathtt{Adv}^{\text{SNM-ATK}}_{A,S,\Sigma}(Rel, k)$ is not negligible in k for any simulator S, then there exists adversary B such that $\mathtt{Adv}^{\text{CNM-ATK}}_{B,\Sigma}(k)$ is not negligible in k, where k is a security parameter and ATK $\in \{CPA, CCA1, CCA2\}$.

Let $A = (A_1, A_2)$ be an adversary for SNM-ATK. First, we construct a CNM-ATK adversary $B = (B_1, B_2)$ using SNM-ATK adversary A in Fig. 5. From the construction of B, we obtain the following equivalence for all $k \in \mathbb{N}$:

$$\Pr[\mathtt{Expt}^{\text{SNM-ATK}}_{A,\Sigma}(Rel, k) = 1] = \Pr[\mathtt{Expt}^{\text{CNM-ATK}}_{B,\Sigma}(k) = 1]. \qquad (4.1)$$

We then construct SNM-ATK simulator $\hat{S} = (\hat{S}_1, \hat{S}_2)$ using SNM-ATK adversary A as shown in Fig. 6.

From the construction of B using A, and the construction of $\hat{S}$, we obtain the following equivalence for all $k \in \mathbb{N}$:

$$\Pr[\mathtt{Expt}^{\text{SNM-ATK}}_{\hat{S},\Sigma}(Rel, k) = 1] = \Pr[\widetilde{\mathtt{Expt}}^{\text{CNM-ATK}}_{B,\Sigma}(k) = 1]. \qquad (4.2)$$

The assumption (for contradiction) that, for any S, $\mathtt{Adv}^{\text{SNM-ATK}}_{A,S,\Sigma}(Rel, k) > \mu(k)$ implies $\mathtt{Adv}^{\text{SNM-ATK}}_{A,\hat{S},\Sigma}(Rel, k) > \mu(k)$ (for specific $\hat{S}$). From this inequality and Eqs.(4.1) and (4.2), we obtain

$$\begin{aligned}
\mathtt{Adv}^{\text{CNM-ATK}}_{B,\Sigma}(k) &= \Pr[\mathtt{Expt}^{\text{CNM-ATK}}_{B,\Sigma}(k) = 1] - \Pr[\widetilde{\mathtt{Expt}}^{\text{CNM-ATK}}_{B,\Sigma}(k) = 1] \\
&= \Pr[\mathtt{Expt}^{\text{SNM-ATK}}_{A,\Sigma}(Rel, k) = 1] - \Pr[\mathtt{Expt}^{\text{SNM-ATK}}_{\hat{S},\Sigma}(Rel, k) = 1] \\
&= \mathtt{Adv}^{\text{SNM-ATK}}_{A,\hat{S},\Sigma}(Rel, k) > \mu(k).
\end{aligned}$$

$$\mathtt{Adv}^{\mathrm{PNM\text{-}ATK}}_{A,\Sigma}(k) \equiv \Pr[\mathtt{Expt}^{\mathrm{PNM\text{-}ATK}}_{A,\Sigma}(k) = 1] - \frac{1}{2},$$

where
$\mathtt{Expt}^{\mathrm{PNM\text{-}ATK}}_{A,\Sigma}(k)$

$$(pk, sk) \xleftarrow{\mathsf{R}} \mathcal{G}(1^k);\ s_1 \xleftarrow{\mathsf{R}} A_1^{O_1}(pk);$$
$$(K^*, C^*) \xleftarrow{\mathsf{R}} \mathcal{E}(pk);\ R \xleftarrow{\mathsf{U}} \{0,1\}^{l(k)};\ b \xleftarrow{\mathsf{U}} \{0,1\};$$
$$X \leftarrow \begin{cases} K^*, \text{ if } \ b = 0 \\ R, \text{ if } \ b = 1 \end{cases}$$
$$(s_2, \boldsymbol{C}) \xleftarrow{\mathsf{R}} A_2^{O_2}(X, s_1, C^*)$$
$$\boldsymbol{K} \leftarrow \mathcal{D}(sk, \boldsymbol{C})$$
$$g \xleftarrow{\mathsf{R}} A_3(s_2, \boldsymbol{K})$$
$$\text{return 1, iff } \ (C^* \notin \boldsymbol{C}) \ \wedge \ (g = b)$$

and
If ATK = CPA, then $O_1 = \perp$ and $O_2 = \perp$.
If ATK = CCA1, then $O_1 = \mathcal{D}(sk, \cdot)$ and $O_2 = \perp$.
If ATK = CCA2, then $O_1 = \mathcal{D}(sk, \cdot)$ and $O_2 = \mathcal{D}(sk, \cdot)$.

Fig. 4. Advantage of PNM-ATK-KEM

$B_1^{O_1}(pk)$	$B_2^{O_2}(X, s, C^*),$
$t_1 \xleftarrow{\mathsf{R}} A_1^{O_1}(pk)$	$(s_2, \boldsymbol{C}) \xleftarrow{\mathsf{R}} A_2^{O_2}(X, s, C^*)$
$s \leftarrow t_1$	Define Rel' by $Rel'(a, \boldsymbol{b}) = 1,$
return s	iff $Rel(a, \boldsymbol{b}, s_2) = 1$
	return $(Rel', \boldsymbol{C})$

Fig. 5. CNM-ATK adversary B using SNM-ATK adversary A.

□

4.2 Proof of Theorem 2:

Proof. We prove that KEM Σ is not SNM-ATK-KEM if Σ is not PNM-ATK-KEM. More precisely, we show that if there exists adversary A such that $\mathtt{Adv}^{\mathrm{PNM\text{-}ATK}}_{A,\Sigma}(k)$ is not negligible in k, then there exist adversary B and relation Rel for any simulator S such that $\mathtt{Adv}^{\mathrm{SNM\text{-}ATK}}_{B,S,\Sigma}(Rel, k)$ is not negligible in k, where k is a security parameter and ATK $\in \{CPA, CCA1, CCA2\}$.

Let $A = (A_1, A_2, A_3)$ be an adversary for PNM-ATK. First, we construct SNM-ATK adversary $B = (B_1, B_2)$ and relation Rel using PNM-ATK adversary A as shown in Fig. 7. Here, we say event $\mathtt{Bad}$ occurs iff Y is not an element of X. From the construction of B, we obtain the following equivalence for all $k \in \mathbb{N}$:

$$\Pr[\mathtt{Expt}^{\mathrm{PNM\text{-}ATK}}_{A,\Sigma}(k) = 1] = \Pr[\mathtt{Expt}^{\mathrm{SNM\text{-}ATK}}_{B,\Sigma}(Rel, k) = 1] \tag{4.3}$$

By Eq.(4.5), we show that, given relation Rel, for any simulator S, the success probability of $\mathtt{Expt}^{\mathrm{SNM\text{-}ATK}}_{S,\Sigma}(Rel, k)$ is at most $\frac{1}{2}$.

```
$\hat{S}_1^{O_1}(pk)$

    $t_1 \overset{R}{\leftarrow} A_1^{O_1}(pk)$
    $s_1 \leftarrow t_1$
    return $s_1$

$\hat{S}_2^{O_2}(X, s_1)$

    $(K^*, C^*) \overset{R}{\leftarrow} \mathcal{E}(pk)$
    $(s_2, \boldsymbol{C}) \overset{R}{\leftarrow} A_2^{O_2}(X, s_1, C^*)$
    return $(s_2, \boldsymbol{C})$
```

Fig. 6. SNM-ATK simulator $\hat{S}$ using SNM-ATK adversary A.

```
$B_1^{O_1}(pk)$

    $t_1 \overset{R}{\leftarrow} A_1^{O_1}(pk)$
    $s_1 \leftarrow t_1$
    return $s_1$

$B_2^{O_2}(X, s_1, C^*)$, where $s_1 = t_1$ and $X = (r_0, r_1)$

    $(t_2, \boldsymbol{C}) \overset{R}{\leftarrow} A_2^{O_2}(r_0, t_1, C^*)$
    Choose random coins $\sigma$ for $A_3$.
    $s_2 \leftarrow (t_2, \sigma, X)$
    return$(s_2, \boldsymbol{C})$

$Rel(Y, \boldsymbol{K}, s_2)$, where $s_2 = (t_2, \sigma, X)$

    If $Y$ is not an element of $X$, return 0.
    If $Y = r_0$, then $b = 0$. Otherwise, $b = 1$.
    $g \leftarrow A_3(t_2, \boldsymbol{K}; \sigma)$
    return 1, iff $b = g$
```

Fig. 7. SNM-ATK adversary B and Relation Rel using PNM-ATK adversary A.

$$\begin{aligned}
\Pr[\mathtt{Expt}_{S,\Sigma}^{\text{SNM-ATK}}(Rel, k) = 1] &= \Pr[g = b \wedge \neg\mathtt{Bad}] \\
&= \Pr[b = 0 \wedge g = 0 \wedge \neg\mathtt{Bad}] + \Pr[b = 1 \wedge g = 1 \wedge \neg\mathtt{Bad}] \\
&= \Pr[b = 0 \wedge \neg\mathtt{Bad}] \times \Pr[g = 0 | b = 0 \wedge \neg\mathtt{Bad}] \\
&\quad + \Pr[b = 1 \wedge \neg\mathtt{Bad}] \times \Pr[g = 1 | b = 1 \wedge \neg\mathtt{Bad}] \\
&\leq \frac{1}{2} \times \Pr[g = 0 | b = 0 \wedge \neg\mathtt{Bad}] + \frac{1}{2} \times \Pr[g = 1 | b = 1 \wedge \neg\mathtt{Bad}] \\
&= \frac{1}{2} \times (\Pr[g = 0] + \Pr[g = 1]) \qquad (4.4) \\
&\qquad \text{(because } b \text{ and } \mathtt{Bad} \text{ are independent of } g\text{)} \\
&= \frac{1}{2} \qquad (4.5)
\end{aligned}$$

By applying Eqs. (4.3) and (4.5) in the above-mentioned assumption that $\mathtt{Adv}_{A,\Sigma}^{\text{PNM-ATK}}(k) > \mu(k)$, we obtain:

$$\begin{aligned}\mathtt{Adv}_{B,S,\Sigma}^{\text{SNM-ATK}}(Rel,k) &= \Pr[\mathtt{Expt}_{B,\Sigma}^{\text{SNM-ATK}}(Rel,k)=1] - \Pr[\mathtt{Expt}_{S,\Sigma}^{\text{SNM-ATK}}(Rel,k)=1] \\ &\geq \Pr[\mathtt{Expt}_{A,\Sigma}^{\text{PNM-ATK}}(k)=1] - \frac{1}{2} \\ &= \mathtt{Adv}_{A,\Sigma}^{\text{PNM-ATK}}(k) \ > \mu(k).\end{aligned}$$

□

4.3 Proof of Theorem 3:

Proof. We prove that KEM Σ is not PNM-ATK-KEM if Σ is not CNM-ATK-KEM. More precisely, we show that if there exists adversary A such that $\mathtt{Adv}_{A,\Sigma}^{\text{CNM-ATK}}(k)$ is not negligible in k, then there exists adversary B such that $\mathtt{Adv}_{B,\Sigma}^{\text{PNM-ATK}}(k)$ is not negligible in k, where k is a security parameter and ATK $\in \{CPA, CCA1, CCA2\}$.

Let $A = (A_1, A_2)$ be an adversary for CNM-ATK. We construct PNM-ATK adversary $B = (B_1, B_2, B_3)$ using CNM-ATK adversary A as shown in Fig. 8. From the construction of B, we obtain

```
$B_1^{O_1}(pk)$

        $t \xleftarrow{R} A_1^{O_1}(pk)$
        $s_1 \leftarrow t$
        return $s_1$

$B_2^{O_2}(X, s_1, C^*)$, where $s_1 = t$ and $X = K^*$ or $R$

        $R' \xleftarrow{U} \{0,1\}^{l(k)}$
        $c \xleftarrow{U} \{0,1\}$
        $X' \leftarrow \begin{cases} (R', X), \text{ if } c = 0 \\ (X, R'), \text{ if } c = 1 \end{cases}$
        $(Rel, \boldsymbol{C}) \xleftarrow{R} A_2^{O_2}(X', s_1, C^*)$
        $s_2 \leftarrow (Rel, X)$
        return$(s_2, \boldsymbol{C})$

$B_3(s_2, \boldsymbol{K})$, where $s_2 = (Rel, X)$

        If $Rel(X, \boldsymbol{K})$, then $g \leftarrow 0$,
            otherwise $g \leftarrow 1$
        return $g$
```

Fig. 8. PNM-ATK adversary B using CNM-ATK adversary A.

$$
\begin{aligned}
&\Pr[\mathtt{Expt}^{\text{PNM-ATK}}_{B,\Sigma}(k)=1]\\
&\qquad= \Pr[b=g]\\
&\qquad= \Pr[b=0\wedge g=0]+\Pr[b=1\wedge g=1])\\
&\qquad= \Pr[b=0]\times\Pr[g=0|b=0]+\Pr[b=1]\times\Pr[g=1|b=1]\\
&\qquad= \frac{1}{2}\Pr[\mathtt{Expt}^{\text{CNM-ATK}}_{A,\Sigma}(k)=1]+\frac{1}{2}(1-\Pr[\widetilde{\mathtt{Expt}}^{\text{CNM-ATK}}_{A,\Sigma}(k)=1])\\
&\qquad= \frac{1}{2}(\Pr[\mathtt{Expt}^{\text{CNM-ATK}}_{A,\Sigma}(k)=1]-\Pr[\widetilde{\mathtt{Expt}}^{\text{CNM-ATK}}_{A,\Sigma}(k)=1])+\frac{1}{2}.
\end{aligned}
$$

That is,

$$
\begin{aligned}
&\Pr[\mathtt{Expt}^{\text{PNM-ATK}}_{B,\Sigma}(k)=1]-\frac{1}{2}\\
&\quad= \frac{1}{2}(\Pr[\mathtt{Expt}^{\text{CNM-ATK}}_{A,\Sigma}(k)=1]-\Pr[\widetilde{\mathtt{Expt}}^{\text{CNM-ATK}}_{A,\Sigma}(k)=1])\\
&\quad= \frac{1}{2}\mathtt{Adv}^{\text{CNM-ATK}}_{A,\Sigma}(k). \qquad (4.6)
\end{aligned}
$$

By applying Eq.(4.6) in the above-mentioned assumption that $\mathtt{Adv}^{\text{CNM-ATK}}_{A,\Sigma}(k)>\mu(k)$, we obtain

$$\mathtt{Adv}^{\text{PNM-ATK}}_{B,\Sigma}(k)=\frac{1}{2}\mathtt{Adv}^{\text{CNM-ATK}}_{A,\Sigma}(k)>\mu(k)/2.$$

□

4.4 Equivalence of the Three Non-Malleability Definitions

From Theorems 1, 2 and 3, we immediately obtain the equivalence of the three non-malleable definitions, SNM-ATK-KEM, CNM-ATK-KEM and PNM-ATK-KEM. Hereafter, we use NM-ATK-KEM for the three non-malleable definitions. If it is clear that NM-ATK-KEM is used for KEM, we will just call it NM-ATK.

5 IND-CCA2 KEM Is Equivalent to NM-CCA2 KEM

This section shows that non-malleability is equivalent to indistinguishability for KEM against adaptive chosen ciphertext attacks (CCA2). For public-key encryption (PKE), it has been already proven that non-malleability is equivalent to indistinguishability against CCA2 [1].

Theorem 4 KEM Σ *is NM-CCA2-KEM, if and only if* Σ *is IND-CCA2-KEM.*

Proof. To prove this theorem, it is enough to show that PNM-CCA2-KEM is equivalent to IND-CCA2-KEM. It is trivial from the definition that KEM Σ is not IND-CCA2-KEM if Σ is not PNM-CCA2-KEM. The opposite direction, that Σ is not PNM-CCA2-KEM if Σ is not IND-CCA2-KEM, is also easy as follows: Let $A=(A_1,A_2)$ be an attacker for IND-CCA2-KEM. We then construct an attacker $B=(B_1,B_2,B_3)$ for PNM-CCA2-KEM using A such that B_1 executes A_1, and B_2 executes A_2 which outputs g and outputs (s_2, $\boldsymbol{C}$) such that $s_2\leftarrow g$ and $\boldsymbol{C}$ is an arbitrary ciphertext. B_3 outputs $s_2(=g)$ regardless of the value of $\boldsymbol{K}$. Clearly, B is an attacker for PNM-CCA2-KEM with the same advantage as that of A for IND-CCA2-KEM. □

6 UC KEM

Let $\Sigma=(\mathcal{G},\mathcal{E},\mathcal{D})$ be a key encapsulation mechanism (KEM). We define the key encapsulation mechanism functionality $\mathcal{F}_{\text{KEM}}$ and protocol π_Σ that is constructed from KEM Σ and has the same interface with the environment as $\mathcal{F}_{\text{KEM}}$.

Definition 5. *Let* $\mathcal{F}_{\text{KEM}}$ *be the key encapsulation mechanism functionality shown in Fig.9, and let* π_Σ *be the key encapsulation mechanism protocol in Fig.10.*

Here, note that there is no functionality of data transmission between parties in $\mathcal{F}_{\text{KEM}}$.

Functionality $\mathcal{F}_{\mathrm{KEM}}$

$\mathcal{F}_{\mathrm{KEM}}$ which runs with adversary S proceeds as follows:

Key Generation: Upon receiving (`KEM.KeyGen`, sid) from some party D, verify that sid=(D, sid') for some sid'. If not, then ignore the request. Else, hand (`KEM.KeyGen`, sid) to adversary S. Upon receiving (`Algorithms`, sid, e, d) from S, where e, d are descriptions of PPT ITMs, output (`Encryption Algorithm`, sid, e) to D.

Encryption: Upon receiving (`KEM.Encrypt`, sid, e') from any party E, do: If $e' \neq e$, or decryptor D is corrupted, then execute e' and obtain (K^*, C_0^*). Let $(key, cip) \leftarrow (K^*, C_0^*)$. Else, obtain (K^*, C_0^*) by e' and $R \overset{\mathsf{U}}{\leftarrow} \{0,1\}^{l(k)}$, then let $(key, cip) \leftarrow (R, C_0^*)$ and record (key, cip). Output (`Key and Ciphertext`, sid, key, cip) to E.

Decryption: Upon receiving a value (`KEM.Decrypt`, sid, C_0^*) from D (and D only), do: If there is a recorded entry (K, C_0^*) for some K then return (`Shared Key`, sid, K) to D. Else, return (`Shared Key`, sid, $d(C_0^*)$) to D. (If threre are more than one K recorded for C_0^*, then output an error message.)

Fig. 9. Key Encapsulation Mechanism Functionality $\mathcal{F}_{\mathrm{KEM}}$

Protocol π_Σ

π_Σ proceeds with parties E and D as follows:

Key Generation: Upon input (`KEM.KeyGen`, sid), party D verifies that sid=(D, sid') for some sid'. If not, then ignore the request. Else, D obtains public key pk and secret key sk by running the algorithm $\mathcal{G}$, and generates $e \leftarrow \mathcal{E}(pk, \cdot)$ and $d \leftarrow \mathcal{D}(sk, \cdot)$, then outputs (`Encryption Algorithm`, sid, e).

Encryption: Upon input (`KEM.Encrypt`, sid, e), party E obtains pair $(key, cip) \leftarrow (K^*, {C_0}^*)$ of a key and a ciphertext by running algorithm e and outputs (`Key and Ciphertext`, sid, key, cip).

Decryption: Upon input (`KEM.Decrypt`, sid, ${C_0}^*$), party D (that has d) obtains $K^* \leftarrow d(C_0^*)$ and outputs (`Shared Key`, sid, K^*).

Fig. 10. Key Encapsulation Mechanism Protocol π_Σ

7 UC KEM Is Equivalent to IND-CCA2 KEM

This section shows that KEM Σ is UC secure if and only if Σ is IND-CCA2 (or NM-CCA2).

Theorem 5 *Let Σ be a KEM scheme, and $\mathcal{F}_{\mathrm{KEM}}$ and π_Σ be as described in Definition 5. Protocol π_Σ UC-realizes $\mathcal{F}_{\mathrm{KEM}}$ with respect to non-adaptive adversaries, if and only if Σ is IND-CCA2-KEM.*

Proof.
("Only if" part)

Let $\Sigma = (\mathcal{G}, \mathcal{E}, \mathcal{D})$ be a KEM scheme. We prove that if Σ is not IND-CCA2-KEM, then π_Σ does not UC-realize $\mathcal{F}_{\mathrm{KEM}}$. In more detail, we can construct environment Z such that, for any ideal process world adversary (simulator) S, Z can tell whether it is interacting with A and π_Σ or with S and the ideal protocol for $\mathcal{F}_{\mathrm{KEM}}$, by using adversary G that breaks Σ in the sense of IND-CCA2-KEM with not negligible advantage (i.e., $\mathtt{Adv}^{\mathrm{IND-CCA2}}_{G,\Sigma}(k) > \mu(k)$).

Z activates parties E and D, and uses adversary G as follows:

1. Activates key receiver D with (KEM.KeyGen, sid) for sid=$(D, 0)$, obtains encryption algorithm e and hands e to G.
2. Activates E with (KEM.Encrypt, sid, e), and obtains (key, cip). Z chooses $b \overset{\cup}{\leftarrow} \{0,1\}$ and $R \overset{\cup}{\leftarrow} \{0,1\}^{l(k)}$. If $b = 0$, then $key' \leftarrow key$. If $b = 1$, then $key' \leftarrow R$. Z hands (key', cip) to G as a target pair of key and ciphertext in the IND-CCA2 game shown in Fig. 1.
3. When G asks its decryption oracle to decrypt ciphertext $C^{\dagger} \neq cip$, Z activates D with input (KEM.Decrypt, $sid, C^{\dagger}$), obtains key $K^{\dagger}$, and hands $K^{\dagger}$ to G.
4. When G outputs $g \in \{0,1\}$, Z outputs $g \oplus b$ and halts.

Here note that Z corrupts no party and interacts with no advesary.

When Z interacts with π_{Σ}, the view of G interacting with Z is exactly the same as that behaving in the real IND-CCA2 game in Fig. 1. Therefore, in this case (say Real), $g = b$ with probability $> \frac{1}{2} + \mu(k)$.

In contrast, when Z interacts with the ideal process world for $\mathcal{F}_{\text{KEM}}$, the view of G interacting with Z is independent of b, since b is independent of (key', cip) generated by Z in step 2 and is independent of the decryption result $K^{\dagger}$ in step 3 (as key' and $K^{\dagger}$ are random strings independent of b). Hence, in this case (say Ideal), $g = b$ with probability of exactly $\frac{1}{2}$.

Thus, $|\Pr[Z \rightarrow 0 \mid \texttt{Real}] - |\Pr[Z \rightarrow 0 \mid \texttt{Ideal}]| \; > \; \mu(k)$.

("If" part)

We show that if π_{Σ} does not UC-realize $\mathcal{F}_{\text{KEM}}$, then Σ is not IND-CCA2-KEM. To do so, we first assume that for any simulator S there exist a real world adversary A and an environment Z that distinguishes with probability $> \frac{1}{2} + \mu(k)$ whether it interacts with S and the ideal process for $\mathcal{F}_{\text{KEM}}$ or with A and π_{Σ}. We then show that there exists an IND-CCA2 attacker G against Σ using Z.

First we show that Z can distinguish (A, π_{Σ}) and $(S, \mathcal{F}_{\text{KEM}})$ only when no party is corrupted. Since we are dealing with non-adaptive adversaries, there are three cases; Case 1: Sender E is corrupted (throughout the protocol), Case 2: Receiver R is corrupted (throughout the protocol), Case 3: E and D are uncorrupted.

In Case 1, we can construct simulator S such that no Z can distinguish (A, π_{Σ}) and $(S, \mathcal{F}_{\text{KEM}})$ as follows:

1. When Z sends (KEM.KeyGen, sid) to D, D forwards it to $\mathcal{F}_{\text{KEM}}$. $\mathcal{F}_{\text{KEM}}$ sends (KEM.KeyGen, sid) to S, S computes (pk,sk) by running algorithm $\mathcal{G}$, and generates e and d, where $e \leftarrow \mathcal{E}(pk, \cdot)$ and $d \leftarrow \mathcal{D}(sk, \cdot)$. S returns (Algorithms, sid, e, d) to $\mathcal{F}_{\text{KEM}}$.
2. When Z sends (KEM.Encrypt, sid, e) to the corrupted party E (i.e., S), S receives the message and sends it to the simulated copy of A, which replies to S. S then returns A's reply (that may be $\perp$) to Z.
3. When Z sends (KEM.Decrypt, sid, C^*) to D, D forwards it to $\mathcal{F}_{\text{KEM}}$. $\mathcal{F}_{\text{KEM}}$ then returns (Shared Key, sid, $d(C^*)$), since E (i.e., S) sends no (KEM.Encrypt, sid, e) to $\mathcal{F}_{\text{KEM}}$, which records nothing as (key, cip). Note that, S does not receive any message in this step.

In this case, Z cannot distinguish (A, π_{Σ}) and $(S, \mathcal{F}_{\text{KEM}})$, because the message returned by S (using A) as E in the ideal world is the same as that returned by A as E in the real world, and (Shared Key, sid, $d(C^*)$) returned by $\mathcal{F}_{\text{KEM}}$ is exactly the same as that returned by D in the real world.

In Case 2, we can also construct simulator S such that no Z can distinguish (A, π_{Σ}) and $(S, \mathcal{F}_{\text{KEM}})$ as follows:

1. When Z sends (KEM.KeyGen, sid) to the corrupted party D (i.e., S), S receives the message and sends it to the simulated copy of A, which returns a reply message (that may be $\perp$) to S. S sends it to Z.
2. When Z sends (KEM.Encrypt, sid, e) to E, E forwards it to $\mathcal{F}_{\text{KEM}}$. $\mathcal{F}_{\text{KEM}}$ generates a corresponding pair (K^*, C^*) by executing e, sets $(key, cip) \leftarrow (K^*, C^*)$ and returns (Key and Ciphertext, sid, key, cip) to E, since D (i.e., S) sends no (KEM.KeyGen, sid) to $\mathcal{F}_{\text{KEM}}$, which records nothing as encryption algorithm e.
3. When Z sends (KEM.Decrypt, sid, C^*) to D (i.e., S), S sends (KEM.Decrypt, sid, C^*) to A. A returns a reply (that may be $\perp$) to S, which forwards A's reply to Z.

In this case, Z cannot distinguish (A, π_{Σ}) and $(S, \mathcal{F}_{\text{KEM}})$, because the message returned by S (using A) as D in the ideal world is the same as that returned by A as D in the real world, and (Key and Ciphertext, sid, key, cip) returned by $\mathcal{F}_{\text{KEM}}$ is exactly the same as that returned by E in the real world.

Thus, Z cannot distinguish the real/ideal worlds in Cases 1 and 2. Hereafter, we consider only Case 3: E and D are uncorrupted.

Referring to the UC framework, three types of messages are sent from Z to A. The first message type is to corrupt either party, the second message type is to report on message sending, and the third message type is to deliver some message. In our protocol π_Σ, parties don't send messages to each other over the network. In addition, we consider the case that no party is corrupted. Therefore, there are no messages from Z to A (and S).

Since there exists at least one environment Z that can distinguish the real life world from the ideal process world for any simulator S, we consider the following special simulator S:

When S receives message (KEM.KeyGen, sid) from $\mathcal{F}_{\mathrm{KEM}}$, S runs key generation algorithm $\mathcal{G}$, obtains public key pk and secret key sk. S sets $e \leftarrow \mathcal{E}(pk, \cdot)$ and $d \leftarrow \mathcal{D}(sk, \cdot)$, and returns (Algorithms, sid, e, d) to $\mathcal{F}_{\mathrm{KEM}}$.

We now show that we can construct adversary G that breaks IND-CCA2-KEM by using the simulated copy of Z which distinguishes real/ideal worlds. To do so, we assume that there is an environment Z such that

$$|\mathtt{IDEAL}_{\mathcal{F}_{\mathrm{KEM}},S,Z}(k,z) - \mathtt{REAL}_{\pi_\Sigma,A,Z}(k,z)| > \mu(k).$$

We then show that G using Z correctly guesses b in the IND-CCA2 game in Fig. 1 with probability of at least $\frac{1}{2} + \mu(k)/2\ell$, where ℓ is the total number of times the encryption oracle is invoked.

In the IND-CCA2 game, G, given a target public-key (encryption algorithm) e and a target pair (key, cip) from the encryption oracle with private random bit b, is allowed to query the decryption oracle, and finally outputs g, which is G's guess of b. G runs Z with the following simulated interaction as protocol $\pi_\Sigma/\mathcal{F}_{\mathrm{KEM}}$.

G acts as follows, where K_i^*, C_i^* and R_i denote the i-th key, ciphertext and random value of the length $l(k)$, respectively:

1. When Z activates some party D with (KEM.KeyGen, sid), G lets D output (Encryption Algorithms, sid, e), where e is the target public-key (encryption algorithm) for G in the IND-CCA2 game.
2. For the first $h-1$ times that Z asks some party E to generate (key, cip) with sid, G lets E return $(key, cip) \leftarrow (K_i^*, C_i^*)$ by using algorithm e.
3. The h-th time that Z asks to generate (key, cip) with sid, G queries its encryption oracle in the IND-CCA2 game, and obtains corresponding pair $(key, cip) \leftarrow (K_h^*, C_h^*)$ (when $b = 0$) or non-corresponding pair $(key, cip) \leftarrow (R_h, C_h^*)$ (when $b = 1$), where $R_h \overset{\mathsf{U}}{\leftarrow} \{0,1\}^{l(k)}$. Accordingly, G hands the pair of (key, cip) to Z.
4. For the remaining $\ell - h$ times that Z asks E to generate (key, cip) with sid, G lets E return $(key, cip) \leftarrow (R_i, C_i^*)$, where $R_i \overset{\mathsf{U}}{\leftarrow} \{0,1\}^{l(k)}$.
5. Whenever Z activates decryptor D with (KEM.Decrypt, sid, C^*), where $C^* = C_i^*$ for some i, G lets D return the corresponding key K_i^* for any i. If C^* is different from all C_i^*'s, then G poses C^* to its decryption oracle, obtains value v, and lets D return v to Z.
6. When Z halts, G outputs whatever Z outputs and halts.

We use a standard hybrid argument to analyze the success probability of G in the IND-CCA2 game.

For $h \in \{0, \ldots, \ell\}$, let $\mathtt{Env}_h$ be an event that for the first h times that Z asks some party E to generate (key, cip) with sid, E returns $(key, cip) \leftarrow (K_i^*, C_i^*)$ by using algorithm e and for the remaining $\ell - h$ times that Z asks E to generate (key, cip) with sid, E returns $(key, cip) \leftarrow (R_i, C_i^*)$, where $R_i \overset{\mathsf{U}}{\leftarrow} \{0,1\}^{l(k)}$. The replies to Z from decryptor D are the same as those shown in step 5 above.

Let H_h be $\Pr[Z \rightarrow 1 | \mathtt{Env}_h]$. We then obtain the following inequality.

$$\sum_{h=1}^{\ell} |H_h - H_{h-1}| \geq |H_\ell - H_0|. \tag{7.1}$$

Here, from the construction of H_h it is clear that

$$H_0 = \mathtt{IDEAL}_{\mathcal{F}_{\mathrm{KEM}},S,Z}(k,z), \tag{7.2}$$

$$H_\ell = \mathtt{REAL}_{\pi_\Sigma,A,Z}(k,z). \tag{7.3}$$

Therefore,

$$\sum_{h=1}^{\ell} |H_h - H_{h-1}| \geq |H_\ell - H_0| \\ = |\mathtt{REAL}_{\pi_\Sigma,A,Z}(k,z) - \mathtt{IDEAL}_{\mathcal{F}_{\mathrm{KEM}},S,Z}(k,z)| \\ > \mu(k). \tag{7.4}$$

Then there exists some $h \in \{1, \cdots \ell\}$ that satisfies

$$|H_h - H_{h-1}| > \mu(k)/\ell. \tag{7.5}$$

Here, w.l.o.g., let $H_{h-1} - H_h > \mu(k)/\ell$, since if $H_h - H_{h-1} > \mu(k)/\ell$ for Z, we can obtain $H_{h-1} - H_h > \mu(k)/\ell$ for Z^*, where Z^* outputs the opposite of Z's output bit.

In step 3 of G's construction, if G gets the corresponding pair of (K_h^*, C_h^*) (when $b = 0$), then the probability that Z outputs 1 is identical to H_h. If, on the other hand, G gets the non-corresponding pair of (R_h, C_h^*) (when $b = 1$), then the probability that Z outputs 1 is identical to H_{h-1}.

Since G's output follows Z's output,

$$H_h = \Pr[g = 1|b = 0], \tag{7.6}$$

$$H_{h-1} = \Pr[g = 1|b = 1], \tag{7.7}$$

where b is the private random bit of the encryption oracle in the IND-CCA2 game and g is G's output (G's guess of b).

Since $\Pr[g = 1|b = 0] + \Pr[g = 0|b = 0] = 1$, we obtain $\Pr[g = 0|b = 0] = 1 - \Pr[g = 1|b = 0]$.

Therefore, we obtain G's success probability, $\Pr[\mathtt{Expt}_{G,\Sigma}^{\mathrm{IND-CCA2}}(k) = 1]$, as follows:

$$\begin{aligned}
&\Pr[\mathtt{Expt}_{G,\Sigma}^{\mathrm{IND-CCA2}}(k) = 1] \\
&\quad = \Pr[b = g] \\
&\quad = \Pr[b = 0] \times \Pr[g = 0|b = 0] + \Pr[b = 1] \times \Pr[g = 1|b = 1]) \\
&\quad = \frac{1}{2} \times (\Pr[g = 0|b = 0] + \Pr[g = 1|b = 1]) \\
&\quad = \frac{1}{2} \times (1 - \Pr[g = 1|b = 0] + \Pr[g = 1|b = 1]) \\
&\quad = \frac{1}{2} \times (1 - H_h + H_{h-1}) \\
&\quad > \frac{1}{2} + \mu(k)/2\ell.
\end{aligned}$$

That is, $\mathtt{Adv}_{G,\Sigma}^{\mathrm{IND-CCA2}}(k) > \mu(k)/2\ell$, which is not negligible in k since ℓ is polynomially bounded in k. □

Acknowledgments

The authors would like to thank Masayuki Abe for his comment on a problem of our non-malleability formulation in [7].

References

1. M. Bellare, A. Desai, D. Pointcheval and P. Rogaway, “Relations Among Notions of Security for Public-Key Encryption Schemes,” CRYPTO’98, LNCS, vol.1462, pp.26-45, Springer Verlag, 1998.
2. M. Bellare and A. Sahai, “Non-Malleable Encryption: Equivalence between Two Notions, and an Indistinguishability-Based Characterisation,” CRYPTO’99, LNCS, vol.1666, pp.519-536, Springer Verlag, 1999.

3. R. Canetti, “Universally Composable Security: A New paradigm for Cryptographic Protocols,” 42nd FOCS, 2001. IACR ePrint Archive 2000/067, http://eprint.iacr.org.
4. D. Dolev, C. Dwork and M. Naor, “Non-malleable Cryptography,” 23rd ACM annual ACM symposium on Theory of computing, pp.542-552, ACM, 1991.
5. J. Herranz, D. Hofheinz and E. Kiltz, “KEM/DEM: Necessary and Sufficient Conditions for Secure Hybrid Encryption,” IACR ePrint Archive 2006/265, http://eprint.iacr.org.
6. W. Nagao, Y. Manabe and T. Okamoto, “A Universally Composable Secure Channel Based on the KEM-DEM Framework,” TCC’05, LNCS, vol.3378, pp.426-444, Springer Verlag, 2005.
7. W. Nagao, Y. Manabe and T. Okamoto, “A Universally Composable Secure Channel Based on the KEM-DEM Framework,” IEICE Transactions on Fundamentals, Vol.E89-A, pp.28-38, 2006 Jan.
8. V.Shoup, “A Proposal for an ISO Standard for Public Key Encryption (version 2.1),” ISO/IEC JTC1/SC27, N2563, 2001 Dec. http://shoup.net/papers/.