Security Notions and Generic Constructions of Chosen Ciphertext Secure Public Key Encryption Schemes

選択暗号文攻撃に対して安全な公開鍵暗号の安全性定義と一般的構成法

Takahiro Matsuda

松田 隆宏

Acknowledgement

Firstly, I would like to express my gratitude to my supervisor, Associate Professor Kanta Matsuura, for his guidance and encouragement during my studies. I have learned a lot from him, including how to do research activities, “where”, “when”, “what”, and “how” to focus on things, and the attitude towards a research activity and education. I would also like to thank my avisors, Professor Tohru Asami and Associate Professor Kaoru Sezaki for their invaluable comments and feedbacks on the results in this thesis and the presentation of them.

I would like to thank the present and past Matsuura-lab members, Takuro Hosoi, Takashi Kitagawa, Jun Furukawa, Nuttapong Attrapadung, Yang Cui, Peng Yang, Jacob Schuldt, Wataru Kitada, Vadim Zendejas, Phan Thi Lan Anh, Yu Watanabe, Yasumasa Nakai, Yi Shi, Daiki Chiba, Bongkot Jenjarrussakul, Ken Ichikawa, Young Seok Choi, Kayoko Ogura, Naoko Hashizume, Sae Nakano, and Yoko Tsuruyama, for many interesting discussions on the topics related and not related to my research area. Especially, I would like to thank Jacob Schuldt. Whenever I had some ideas, he always took the time for listening to and commenting on my ideas. The discussion with him was always fun and fruitful, and made it possible to achieve the results in and not in this thesis. I would like to thank external Matsuura-lab meeting visitors, Chiaki Okada, Kazuto Ogawa, Makoto Sugita, Ryo Nojima, Hitoshi Tanuma, and Kouhei Kasamatsu, who gave me helpful suggestions and comments on my works.

I would like to thank the researchers at RCIS, AIST, Prof. Hideki Imai, Goichiro Hanaoka, Rui Zhang, Nuttapong Attrapadung, Yang Cui (these two people graduated from the Univ. of Tokyo, and then joined RCIS), and many other participants of the study group (Akarui Angou Benkyo-Kai). Especially, I would like to thank Goichiro Hanaoka for his sincere guidance, suggestions, comments, and discussion with me about research topics not only my own but also various topics, which was always insightful and helpful, and made it possible to achieve the results in this thesis, especially in the first part.

Finally, I would like to thank my parents for their unconditional supports and loves in all stages of my life.

Abstract

Public key encryption (PKE) is a fundamental cryptographic primitive with which we can communicate securely over possibly insecure network without shared secret information in advance. For PKE schemes, security against chosen ciphertext attacks (CCA security) is nowadays considered as a standard security notion needed in most practical applications/situations where PKE schemes are used. Roughly, CCA security captures security against "active" adversaries that can access to an imaginary machine called decryption oracle which on input a ciphertext returns a decryption result of it, and has been shown to imply important strong security notions such as non-malleability and universal composability. Therefore, studies on constructing and understanding CCA secure PKE schemes are important research topics in the area of cryptography. In this thesis, we focus on "generic constructions" of CCA secure PKE schemes from other cryptographic primitives, and make several contributions both from practical and theoretical points of view.

Firstly, aiming at generic constructions that lead to CCA secure PKE schemes with practical efficiency, we focus on the so-called "IBE-to-PKE" transformation paradigm, where IBE stands for identity-based encryption and is a kind of PKE scheme where any string can be used as a public key. This is a methodology that transforms an IBE scheme which only satisfies security against chosen plaintext attacks (CPA security), the least requirement as an encryption scheme, into a CCA secure PKE scheme, and is the only known generic methodology with which we can construct CCA secure PKE schemes with practical efficiency. The biggest problem of this methodology is that the constructed PKE scheme has large ciphertext size, even if we use a practical IBE scheme as a building block. We propose two approaches to overcome this problem. The first approach is to require non-malleability, slightly stronger security than CPA security, for the underlying IBE scheme, and develop a new very simple IBE-to-PKE transformation where we only use one-way function, the weakest primitive used in the area of cryptography, as an additional building block. The second approach is to develop a new efficient encapsulation scheme, which is a special kind of commitment scheme and is a primitive used in one of the previous IBE-to-PKE transformations, from a special kind of pseudorandom generator. Both approaches do not need strong cryptographic primitives as additional building blocks, and lead to CCA secure PKE schemes with smaller ciphertext size than the previous IBE-to-PKE transformations.

Secondly, we focus on the problem of whether it is possible to construct a CCA secure PKE scheme only from a CPA secure one. This is an important fundamental open problem that leads to clarifying a necessary and sufficient condition to realize a CCA secure PKE scheme. Regarding this problem, the best known positive results are the constructions of so-called bounded CCA secure schemes from any CPA secure PKE scheme, where bounded CCA security is security against adversaries that make at most the predetermined number of decryption queries, and thus is weaker than ordinary CCA security. Since we can achieve

the best possible security in the bounded CCA security notions, in order to further tackle the fundamental problem, we need new security notions that capture intermediate security notions that lie between CPA and CCA security in a different sense from bounded CCA security. Motivated by this situation, in order to provide a theoretical foundation for further tackling the above problem, we focus on parallel decryption queries for an extension of bounded CCA security, and introduce a new security notion which we call "mixed CCA" security. It captures security against adversaries that make single and parallel decryption queries in a predetermined order, where each parallel query can contain unboundedly many ciphertexts. Moreover, how the decryption oracle is available before and after the challenge is also taken into account in this new security definition, which enables us to capture existing major security notions that lie between CPA and CCA security, including a complex notion like non-malleability against bounded CCA, in a unified security notion. We investigate the relations among mixed CCA security notions, and show a necessary and sufficient condition regarding implications/separations between any two notions in mixed CCA security. We then show two black-box constructions of PKE schemes from CPA secure ones, one of which satisfies a strictly stronger security notion than the security notions achieved by the existing constructions of PKE schemes constructed only from a CPA secure one. We also discuss the consequences of our results regarding security with parallel decryption queries and give several observations.

Contents

Acknowledgement **i**

Abstract **iii**

1 Introduction **1**
1.1 Overview and Motivation 1
1.2 Outline and Summary of Contributions 2

2 Basic Definitions **5**
2.1 Basic Notations 5
2.2 Public Key Encryption 6
2.3 Key Encapsulation Mechanism 8
2.4 Data Encapsulation Mechanism 9
2.5 Identity-Based Encryption 9
2.6 Tag-Based Encryption 11
2.7 Encapsulation Scheme 12
2.8 Pseudorandom Generator 13
2.9 Target Collision Resistant Hash Function 14
2.10 Pseudorandom Function 14
2.11 Signature 15
2.12 Message Authentication Code 15
2.13 One-Way Function 16

3 Practical Constructions: IBE-to-PKE Transformations **17**
3.1 Introduction 17
3.1.1 Background and Motivation 17
3.1.2 Our Contribution 18
3.1.3 Related Works 20
3.1.4 Organization of This Chapter 21
3.2 The Boneh-Katz Transformation 21
3.3 Proposed Transformation using Non-malleable IBE 23
3.3.1 Construction 23
3.3.2 Security 23
3.3.3 Applying the Transformation to Tag-Based Encryption 32
3.4 Proposed Encapsulation Scheme 34
3.4.1 Construction 35
3.4.2 Security 35

3.4.3 Concrete Instantiation of PRG with NCR Property . . . 37
3.4.4 PRG with Near Collision Resistance from Any One-Way Permutation 38
3.5 Comparison . . . 39
3.6 Conclusion . . . 41

4 Towards CCA Security from CPA Security 43
4.1 Introduction . . . 43
4.1.1 Background . . . 43
4.1.2 Our Contribution . . . 45
4.1.3 Related Works . . . 46
4.1.4 Organization of This Chapter . . . 47
4.2 Extending Bounded CCA Security: Mixed CCA Security . . . 47
4.2.1 Mixed CCA Security . . . 48
4.2.2 General Properties of Mixed CCA Security . . . 50
4.3 Relations among Security Notions for Mixed CCA Security . . . 54
4.3.1 "is-Simulatable-by" Relation for Query Sequences . . . 55
4.3.2 Useful Tool for Separation: Backdoor-Sequence Scheme . . . 56
4.3.3 Separation Results . . . 62
4.3.4 Implication Results . . . 78
4.3.5 Necessary and Sufficient Conditions for Implications/Separations . . . 81
4.4 Black-box Feasibility Results from `IND-CPA` Secure PKE Schemes . . . 82
4.5 Open Problems . . . 86
4.6 Conclusion . . . 87

5 Conclusion 90

A Publication List 101

B The Existing PKE Constructions 104
B.1 The CDMW Construction . . . 104
B.2 The CHH+ Construction . . . 104

Chapter 1

Introduction

Contents

1.1 Overview and Motivation 1
1.2 Outline and Summary of Contributions 2

1.1 Overview and Motivation

Public key encryption (PKE) is a fundamental cryptographic primitive with which we can communicate securely over possibly insecure network without shared secret information in advance. The most fundamental security requirement, which is nowadays considered as a least security requirement as a PKE scheme, is *semantic security* [60], also called `IND-CPA` security (or just CPA security), which guarantees that a ciphertext does not leak any information (even one-bit) of the corresponding plaintext. However, nowadays, *security against chosen ciphertext attacks* (CCA security) [84, 93] is considered as a "standard" security notion that is required in most practical applications/situations where PKE schemes are used. Roughly, CCA security captures security against "active" adversaries that can access to an imaginary machine called decryption oracle which on input a ciphertext returns a decryption result of it, and has been shown to imply important strong security notions such as non-malleability [47, 7] and universal composability [30, 35]. Therefore, studies on constructing and understanding CCA secure PKE schemes are important research topics in the area of cryptography.

We can roughly categorize the approaches for constructing CCA secure PKE schemes into two types: Constructions from specific number-theoretic assumptions and constructions from general assumptions. (From now on, we write `IND-CCA1` to denote non-adaptive CCA security [84] and `IND-CCA2` to denote adaptive CCA security [93])

The approaches of the first type have been successful so far from both theoretical and practical points of view. After the first novel practical scheme based on the decisional Diffie-Hellman (DDH) assumption by Cramer and Shoup [43], many practical `IND-CCA2` secure PKE schemes that pursue smaller ciphertext size, have small computation costs, and/or base security on weaker assumptions have been constructed so far, e.g. [75, 27, 70, 98, 65, 37, 61, 67, 73, 66, 41, 62]. Especially, the scheme by Cash et al. [37] (and the schemes in recent papers [61, 41, 62]) is based on the computational DH (CDH) assumption, while the scheme by Hofheinz and Kiltz [67] is based on the factoring assumption, and both assumptions are very fundamental in the area of cryptography.

The approaches of the second type, which we call *generic constructions*, have also been successful, mainly from a theoretical point of view. Naor and Yung [84] proposed a generic construction of IND-CCA1 secure PKE schemes from semantically secure (IND-CPA) PKE schemes, using non-interactive zero-knowledge (NIZK) proofs [17]. It is known that if enhanced trapdoor permutations exist, then NIZK proofs for any NP language is possible [15, 56][1]. Based on the Naor-Yung paradigm, several constructions of IND-CCA2 secure PKE schemes were also proposed [47, 96, 77]. Since the existence of enhanced trapdoor permutations implies the existence of IND-CPA secure PKE schemes, these results suggest that we can construct IND-CCA2 secure PKE schemes from any enhanced trapdoor permutation. (We review other generic constructions of IND-CCA2 secure PKE schemes in Appendix 4.1.3.)

However, to the best of our knowledge, the following two questions have not been solved before:

- *Is there a generic construction that lead to CCA secure PKE scheme with practical efficiency?*
- *Is it possible to generically construct a CCA (*IND-CCA1 or IND-CCA2*) secure PKE scheme from any* IND-CPA *secure one?*

The first question is impotant, (of course as it indicates) from the practical point of view, while the second question is a famous fundamental problem that lead to clarifying a necessary and sufficient condition of a CCA secure PKE scheme. In this thesis, we make progress towards these problems.

1.2 Outline and Summary of Contributions

In this thesis, we make contributions to the generic constructions of CCA secure PKE schemes. Roughly, our contributions can be classified into two parts: practical aspects and theoretical aspects. The contruction for each of aspects are summarized below. (All the technical terms that appear below will either be defined in Chapter 2 or in the chapter where the results are presented.)

- In Chapter 3, aiming at generic constructions that lead to CCA secure PKE schemes with practical efficiency, we focus on the so-called *IBE-to-PKE transformation paradigm* [34, 26], which is the only known generic methodology with which we can construct CCA secure PKE schemes with practical efficiency. As the name indicates, this methodology transforms an identity-based encryption scheme [99, 24], a kind of PKE scheme in which we can use any string as a public key, into a CCA secure PKE scheme, possibly using some other cryptographic primitives as additional building blocks. To improve the large ciphertext size that all the previous methods sufferred from, we propose two approaches. The first approach is to require non-malleability [47, 13, 14], slightly stronger security than CPA security, for the underlying IBE scheme, and develop a new very simple IBE-to-PKE transformation where we only use one-way function, the weakest primitive used in the area of cryptography, as an additional building block. The second approach is to develop a new efficient encapsulation scheme [26], which is a special kind of commitment scheme and is a primitive used in one of the previous IBE-to-PKE transformations,

[1] It was shown in [57] that we actually need the so-called *doubly-enhanced* trapdoor permutations.

from a special kind of pseudorandom generator. Both approaches do not need strong cryptographic primitives as additional building blocks, and lead to CCA secure PKE schemes with smaller ciphertext size than the previous IBE-to-PKE transformations.

- In Chapter 4, We focused on the problem of whether it is possible to construct a CCA secure PKE scheme only from a CPA secure one is one of the most important fundamental open problems, which leads to clarifying a necessary and sufficient condition to realize a CCA secure PKE scheme. Regarding this problem, the best known positive results are the constructions of so-called bounded CCA secure schemes from any CPA secure PKE scheme [40, 38], where bounded CCA security is security against adversaries that make at most the predetermined number of decryption queries, and thus is weaker than ordinary CCA security. Since we can achieve the best possible security in the bounded CCA security notions, in order to further tackle the fundamental problem, we need new security notions that capture intermediate security notions that lie between CPA and CCA security in a different sense from bounded CCA security. Motivated by this situation, in order to provide a theoretical foundation for further tackling the above problem, we focused on parallel decryption queries for the extension of bounded CCA security, and introduce a new security notion which we call *mixed CCA* security. It captures security against adversaries that make single and parallel decryption queries in a predetermined order, where each parallel query can contain unboundedly many ciphertexts. Moreover, how the decryption oracle is available before and after the challenge is also taken into account in this new security definition, which enables us to capture existing major security notions that lie between CPA and CCA security, including complex notion like non-malleability against bounded CCA, in a unified security notion. We investigated the relations among mixed CCA security notions, and show a necessary and sufficient condition regarding implications/separations between any two notions in mixed CCA security. We then showed two black-box constructions of PKE schemes from CPA secure ones. The first scheme satisfies a strictly stronger security notion than the security notions achieved by the existing constructions of PKE schemes constructed only from a CPA secure one, while the second one achieves yet another security notion that has not been achieved by the previous constructions. We also discussed the consequences of our results regarding security with parallel decryption queries, and give several observations as well as some open problems.

Chapter 2

Basic Definitions

Contents

2.1 Basic Notations **5**
2.2 Public Key Encryption **6**
2.3 Key Encapsulation Mechanism **8**
2.4 Data Encapsulation Mechanism **9**
2.5 Identity-Based Encryption **9**
2.6 Tag-Based Encryption **11**
2.7 Encapsulation Scheme **12**
2.8 Pseudorandom Generator **13**
2.9 Target Collision Resistant Hash Function **14**
2.10 Pseudorandom Function **14**
2.11 Signature **15**
2.12 Message Authentication Code **15**
2.13 One-Way Function **16**

In this Chapter, we review the basic terminology, notation, and definitions of primitives that appear throughout the thesis.

2.1 Basic Notations

We use the following notations: $\mathbb{N}$ denotes the set of all integers, and if $q \in \mathbb{N}$ then $[q] = \{1, \ldots, q\}$. "$x \leftarrow y$" denotes that x is chosen uniformly at random from y if y is a finite set, x is output from y if y is a function or an algorithm, or y is assigned to x otherwise. "$x||y$" denotes a concatenation of x and y. "$|x|$" denotes the size of the set if x is a finite set or bit length of x if x is an element of some set. "PPTA" denotes a *probabilistic polynomial time algorithm*. Unless otherwise stated, algorithms considered in this thesis are PPTAs. If $\mathcal{A}$ is a probabilistic algorithm then $y \leftarrow \mathcal{A}(x; r)$ denotes that $\mathcal{A}$ computes y as output by taking x as input and using r as randomness. $\mathcal{A}^{\mathcal{O}}$ denotes an algorithm $\mathcal{A}$ with oracle access to $\mathcal{O}$. A function $f(k)$ is said to be *negligible* if for any positive polynomial $p(k)$ and for all sufficiently large k, we have $f(k) < \frac{1}{p(k)}$. A function $g(k)$ is said to be *overwhelming* if $1 - g(k)$ is negligible.

$\mathsf{Expt}^{\texttt{IND-ATK}}_{\Pi,\mathcal{A}}(k)$:
$(pk, sk) \leftarrow \mathsf{PKG}(1^k)$;
$(m_0, m_1, \mathsf{st}_{\mathcal{A}}) \leftarrow \mathcal{A}_1^{\mathcal{O}_1}(pk)$;
$b \leftarrow \{0,1\}$;
$c^* \leftarrow \mathsf{PEnc}(pk, m_b)$;
$b' \leftarrow \mathcal{A}_2^{\mathcal{O}_2}(c^*, \mathsf{st}_{\mathcal{A}})$;
`If` $b' = b$ `then return` 1
`else return` 0

$\mathsf{Expt}^{\texttt{NM-ATK}}_{\Pi,\mathcal{A}}(k)$:
$(pk, sk) \leftarrow \mathsf{PKG}(1^k)$;
$(m_0, m_1, \mathsf{st}_{\mathcal{A}}) \leftarrow \mathcal{A}_1^{\mathcal{O}_1}(pk)$;
$b \leftarrow \{0,1\}$;
$c^* \leftarrow \mathsf{PEnc}(pk, m_b)$;
$(\overrightarrow{c'}, \mathsf{st}'_{\mathcal{A}}) \leftarrow \mathcal{A}_2^{\mathcal{O}_2}(c^*, \mathsf{st}_{\mathcal{A}})$
where $\overrightarrow{c'} = (c'_1, c'_2, \ldots)$;
$m'_i \leftarrow \mathsf{PDec}(sk, c'_i)$ `for` $1 \le i \le |\overrightarrow{c'}|$;
$\overrightarrow{m'} \leftarrow (m'_1, m'_2, \ldots, m'_{|\overrightarrow{c'}|})$;
$b' \leftarrow \mathcal{A}_3(\overrightarrow{m'}, \mathsf{st}'_{\mathcal{A}})$;
`If` $b' = b$ `then return` 1
`else return` 0

Available oracles

ATK	$\mathcal{O}_1(\cdot)$	$\mathcal{O}_2(\cdot)$
CPA	$\perp$	$\perp$
CCA1	$\mathsf{PDec}(sk, \cdot)$	$\perp$
CCA2	$\mathsf{PDec}(sk, \cdot)$	$\mathsf{PDec}(sk, \cdot)$
q-CCA2	$\mathsf{PDec}(sk, \cdot)$	$\mathsf{PDec}(sk, \cdot)$

($\perp$ means the oracle is unavailable.)

Figure 2.1: Experiments for defining security notions of PKE schemes

2.2 Public Key Encryption

A public key encryption (PKE) scheme Π consists of the following three PPTAs:

PKG: A key generation algorithm that takes 1^k (security parameter k) as input, and outputs a public/secret key pair (pk, sk). We write: $(pk, sk) \leftarrow \mathsf{PKG}(1^k)$.

PEnc: An encryption algorithm that takes pk and a plaintext $m \in \mathcal{M}_\Pi$ as input, and outputs a ciphertext c. We write: $c \leftarrow \mathsf{PEnc}(pk, m)$.

PDec: A deterministic decryption algorithm that takes sk and c as input, and outputs a plaintext m or an error symbol $\perp$. We write: $m \leftarrow \mathsf{PDec}(sk, c)$.

where $\mathcal{M}_\Pi$ is a plaintext space of Π.

We require $\mathsf{PDec}(sk, \mathsf{PEnc}(pk, m)) = m$ for all (pk, sk) output from PKG and all $m \in \mathcal{M}$.

Security Notions for PKE Schemes. The security notions for PKE schemes are expressed by a combination of a goal and an attack type of an adversary. As conventional security notions for PKE schemes, here we recall *indistinguishability* (IND) and *non-malleability* (NM) for security goals and *chosen plaintext attacks* (CPA), *non-adaptive chosen ciphertext attacks* (CCA1), *adaptive chosen ciphertext attacks* (CCA2), and *q-bounded chosen ciphertext attacks* (q-CCA2) [40] for attack types of an adversary. Non-malleability for PKE schemes we use in this thesis is the so-called *parallel chosen-ciphertext attack* based definition [13, 14], which is equivalent to the indistinguishability based definition used in [89, 90][1].

Formally, we define the security notions IND-ATK and NM-ATK of a PKE scheme $\Pi = (\mathsf{PKG}, \mathsf{PEnc}, \mathsf{PDec})$ for ATK $\in \{\texttt{CPA}, \texttt{CCA1}, \texttt{CCA2}, q\text{-}\texttt{CCA2}\}$ (with $q \ge 0$) via the experiments $\mathsf{Expt}^{\texttt{IND-ATK}}_{\Pi,\mathcal{A}}$ and $\mathsf{Expt}^{\texttt{NM-ATK}}_{\Pi,\mathcal{A}}$ in Figure 2.1 that an adversary $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2(, \mathcal{A}_3))$ runs in, respectively. We refer to the vector $\overrightarrow{c'}$ of ciphertexts that the NM-ATK adversary of the second stage outputs as the *final parallel query* (though it is not a query to an oracle), and the vector $\overrightarrow{m'}$ as its answer.

[1] Pass et al. [90] prove that *many-message* (indistinguishability-based) non-malleability, which considers multiple challenge messages, and single-message non-malleability, adopted in this thesis, are equivalent.

We make several restrictions: If $\mathtt{ATK} = q\text{-}\mathtt{CCA2}$, then the total number of $\mathcal{A}$'s queries to the oracles $\mathcal{O}_1$ and $\mathcal{O}_2$ must be less than or equal to q. In both types of experiments, $\mathcal{A}_2$ is not allowed to issue c^* to $\mathcal{O}_2$ if $\mathtt{ATK} \in \{\mathtt{CCA2}, q\text{-}\mathtt{CCA2}\}$. Besides, in the $\mathtt{NM\text{-}ATK}$ experiments, $\mathcal{A}_2$ is not allowed to include c^* into $\overrightarrow{c'}$.

We define the advantage of an adversary $\mathcal{A}$ in the $\mathtt{GOAL\text{-}ATK}$ experiment by the following function of the security parameter k: $\mathsf{Adv}^{\mathtt{GOAL\text{-}ATK}}_{\Pi,\mathcal{A}}(k) = |\Pr[\mathsf{Expt}^{\mathtt{GOAL\text{-}ATK}}_{\Pi,\mathcal{A}}(k) = 1] - \frac{1}{2}|$.

Definition 1. *Let* $\mathtt{GOAL} \in \{\mathtt{IND}, \mathtt{NM}\}$ *and* $\mathtt{ATK} \in \{\mathtt{CPA}, \mathtt{CCA1}, \mathtt{CCA2}, q\text{-}\mathtt{CCA2}\}$. *We say that a PKE scheme* Π *is* (t, ϵ)-$\mathtt{GOAL\text{-}ATK}$ *secure if we have* $\mathsf{Adv}^{\mathtt{GOAL\text{-}ATK}}_{\Pi,\mathcal{A}}(k) \leq \epsilon$ *for any algorithm* $\mathcal{A}$ *running in time less than* t. *Furthermore, we simply say that* Π *is* $\mathtt{GOAL\text{-}ATK}$ *secure if* $\mathsf{Adv}^{\mathtt{GOAL\text{-}ATK}}_{\Pi,\mathcal{A}}(k)$ *is negligible for any PPTA* $\mathcal{A}$.

Implications and Separations of Security Notions. In this thesis, we will show several implications and separations of security notions. We follow the methodology used in several papers [7, 92, 64, 40, 82]. Though we write only the definition for PKE schemes, the same is defined for any primitive.

Definition 2. *Let* $\mathtt{X}$ *and* $\mathtt{Y}$ *be security notions for PKE schemes. We say that* $\mathtt{X}$ *security implies* $\mathtt{Y}$ *security for PKE schemes if any* $\mathtt{X}$ *secure PKE scheme is also* $\mathtt{Y}$ *secure. We say that* $\mathtt{X}$ *security does not imply* $\mathtt{Y}$ *security for PKE schemes if, under the assumption that* $\mathtt{X}$ *secure PKE schemes exist, there exists a PKE scheme which is* $\mathtt{X}$ *secure but is not* $\mathtt{Y}$ *secure.*

Smoothness. The notion of smoothness of PKE schemes was recently formalized by Bellare et al. [8]. The smoothness of a PKE scheme $\Pi = (\mathsf{PKG}, \mathsf{PEnc}, \mathsf{PDec})$ with plaintext space $\mathcal{M}_\Pi$, is defined as follows:

$$\mathsf{Smth}_\Pi(k) = \mathop{\mathbf{E}}_{(pk,sk)\leftarrow\mathsf{PKG}(1^k)} \left[\max_{(m,c)\in\mathcal{M}_\Pi\times\{0,1\}^*} \Pr_{c'\leftarrow\mathsf{PEnc}(pk,m)}[c' = c] \right].$$

Definition 3. *We say that a PKE scheme* Π *is* ϵ-smooth *if we have* $\mathsf{Smth}_\Pi(k) \leq \epsilon$. *Furthermore, we simply say that* Π *is* smooth *if* $\mathsf{Smth}_\Pi(k)$ *is negligible.*

Shielding Black-Box Constructions. We briefly recall the definition of a shielding black-box construction of a PKE scheme that is secure in the sense of $\mathtt{X}$ from a PKE scheme that is secure in the sense of $\mathtt{Y}$. The notion of black-box constructions we mention in this thesis is classified as *fully-black-box* ones [94], but specified for PKE-to-PKE constructions. (for details, see [94]). The notion of the *shielding* constructions is from [55].

Definition 4. *Let* $\mathtt{X}$ *and* $\mathtt{Y}$ *be security notions for PKE schemes. We say that there exists a shielding black-box construction of an* $\mathtt{X}$ *secure PKE scheme from a* $\mathtt{Y}$ *secure PKE scheme, if there exist oracle PPTAs* $\Pi = (\mathsf{PKG}, \mathsf{PEnc}, \mathsf{PDec})$ *and* $\mathcal{B}$ *with the following properties: For all algorithms* $\pi = (\mathsf{pkg}, \mathsf{penc}, \mathsf{pdec})$ *and* $\mathcal{A}$ *(each algorithm can be of arbitrary complexity),*

Correctness: *If* π *satisfies correctness as a PKE scheme, so does* $\Pi^{\mathsf{pkg},\mathsf{penc},\mathsf{pdec}} = (\mathsf{PKG}^{\mathsf{pkg},\mathsf{penc},\mathsf{pdec}}, \mathsf{PEnc}^{\mathsf{pkg},\mathsf{penc},\mathsf{pdec}}, \mathsf{PDec}^{\mathsf{pkg},\mathsf{pdec}})$.

Security: *If* $\mathcal{A}$ *breaks* $\mathtt{X}$ *security of* $\Pi^{\mathsf{pkg},\mathsf{penc},\mathsf{pdec}} = (\mathsf{PKG}^{\mathsf{pkg},\mathsf{penc},\mathsf{pdec}}, \mathsf{PEnc}^{\mathsf{pkg},\mathsf{penc},\mathsf{pdec}}, \mathsf{PDec}^{\mathsf{pkg},\mathsf{pdec}})$ *then* $\mathcal{B}^{\mathcal{A},\mathsf{pkg},\mathsf{penc},\mathsf{pdec}}$ *breaks* $\mathtt{Y}$ *security of* π.

(Note that PDec *does not have access to* penc.*)*

$\mathsf{Expt}^{\mathtt{IND\text{-}ATK}}_{\Gamma,\mathcal{A}}(k)$:
$(pk, sk) \leftarrow \mathsf{KKG}(1^k)$;
$\mathsf{st}_{\mathcal{A}} \leftarrow \mathcal{A}_1^{\mathcal{O}_1}(pk)$;
$b \leftarrow \{0,1\}$; $K_0^* \leftarrow \mathcal{K}$;
$(c^*, K_1^*) \leftarrow \mathsf{KEnc}(pk)$;
$b' \leftarrow \mathcal{A}_2^{\mathcal{O}_2}(c^*, K_b^*, \mathsf{st}_{\mathcal{A}})$;
If $b' = b$ then return 1
else return 0

$\mathsf{Expt}^{\mathtt{NM\text{-}ATK}}_{\Gamma,\mathcal{A}}(k)$:
$(pk, sk) \leftarrow \mathsf{KKG}(1^k)$;
$\mathsf{st}_{\mathcal{A}} \leftarrow \mathcal{A}_1^{\mathcal{O}_1}(pk)$;
$b \leftarrow \{0,1\}$; $K_0^* \leftarrow \mathcal{K}$;
$(c^*, K_1^*) \leftarrow \mathsf{KEnc}(pk)$;
$(\overrightarrow{c'}, \mathsf{st}'_{\mathcal{A}}) \leftarrow \mathcal{A}_2^{\mathcal{O}_2}(c^*, K_b^*, \mathsf{st}_{\mathcal{A}})$;
$K'_i \leftarrow \mathsf{KDec}(sk, c'_i)$ for $1 \le i \le |\overrightarrow{c'}|$;
$\overrightarrow{K'} \leftarrow (K'_1, \ldots, K'_{|\overrightarrow{c'}|})$;
$b' \leftarrow \mathcal{A}_3(\overrightarrow{K'}, \mathsf{st}'_{\mathcal{A}})$;
If $b' = b$ then return 1
else return 0

Available oracles

ATK	$\mathcal{O}_1(\cdot)$	$\mathcal{O}_2(\cdot)$
CPA	$\perp$	$\perp$
CCA1	$\mathsf{KDec}(sk, \cdot)$	$\perp$
CCA2	$\mathsf{KDec}(sk, \cdot)$	$\mathsf{KDec}(sk, \cdot)$
q-CCA2	$\mathsf{KDec}(sk, \cdot)$	$\mathsf{KDec}(sk, \cdot)$

($\perp$ means the oracle is unavailable.)

Figure 2.2: Experiments for defining security notions KEMs

2.3 Key Encapsulation Mechanism

A key encapsulation mechanism (KEM) Γ consists of the following three PPTAs:

KKG: A key generation algorithm that takes 1^k (security parameter κ) as input, and outputs a public/secret key pair (pk, sk). We write: $(pk, sk) \leftarrow \mathsf{KKG}(1^k)$.

KEnc: An encapsulation algorithm that takes pk as input, and outputs a ciphertext c and a session-key $K \in \mathcal{K}$. We write: $(c, K) \leftarrow \mathsf{KEnc}(pk, m)$.

KDec: A deterministic decapsulation algorithm that takes sk and c as input, and outputs a session-key K or an error symbol $\perp$. We write: $K \leftarrow \mathsf{KDec}(sk, c)$.

where $\mathcal{K}$ is a session-key space of Γ.

We require $\mathsf{KDec}(sk, c) = K$ for all (pk, sk) output from KKG and all (c, K) output from $\mathsf{KEnc}(pk)$.

Security Notions for KEMs. In an analogous way to the security definitions for PKE schemes, the security notions for KEMs are expressed by the combination of a goal and an adversary's attack type. We recall IND and NM for security goals and CPA, CCA1, CCA2, and q-CCA2 for attack types of an adversary. Like the definition for PKE schemes, non-malleability for KEMs we use in this thesis is the parallel chosen-ciphertext attack based definition [82, 64].

Formally, we define the security notions IND-ATK and NM-ATK of a KEM $\Gamma = (\mathsf{KKG}, \mathsf{KEnc}, \mathsf{KDec})$ (with the session-key space $\mathcal{K}$) for $\mathtt{ATK} \in \{\mathtt{CPA}, \mathtt{CCA1}, \mathtt{CCA2}, q\text{-}\mathtt{CCA2}\}$ (with $q \ge 0$) via the experiments $\mathsf{Exp}^{\mathtt{IND\text{-}ATK}}_{\Gamma,\mathcal{A}}$ and $\mathsf{Exp}^{\mathtt{NM\text{-}ATK}}_{\Gamma,\mathcal{A}}$ in Figure 2.2 (bottom) that an adversary $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2(, \mathcal{A}_3))$ runs in, respectively. The restrictions we need to make are exactly the same as those for PKE schemes.

We define the advantage of an adversary $\mathcal{A}$ in the GOAL-ATK experiment by the following function of the security parameter k: $\mathsf{Adv}^{\mathtt{GOAL\text{-}ATK}}_{\Gamma,\mathcal{A}}(k) = |\Pr[\mathsf{Expt}^{\mathtt{GOAL\text{-}ATK}}_{\Gamma,\mathcal{A}}(k) = 1] - \frac{1}{2}|$.

Definition 5. *Let* $\mathtt{GOAL} \in \{\mathtt{IND}, \mathtt{NM}\}$ *and* $\mathtt{ATK} \in \{\mathtt{CPA}, \mathtt{CCA1}, \mathtt{CCA2}, q\text{-}\mathtt{CCA2}\}$. *We say that a KEM* Γ *is* (t, ϵ)-GOAL-ATK *secure if we have* $\mathsf{Adv}^{\mathtt{GOAL\text{-}ATK}}_{\Gamma,\mathcal{A}}(k) \le \epsilon$ *for any algorithm* $\mathcal{A}$ *running in time less than* t. *Furthermore, we simply say that* Γ *is* GOAL-ATK *secure if* $\mathsf{Adv}^{\mathtt{GOAL\text{-}ATK}}_{\Gamma,\mathcal{A}}(k)$ *is negligible for any PPTA* $\mathcal{A}$.

Smoothness. The notion of smoothness of KEMs was recently formalized by Bellare et al. [8]. The smoothness of a KEM $\Gamma = (\mathsf{KKG}, \mathsf{KEnc}, \mathsf{KDec})$, denoted by Smth_Γ, is defined as follows:

$$\mathsf{Smth}_\Gamma(k) = \mathop{\mathbf{E}}_{(pk,sk)\leftarrow\mathsf{KKG}(1^k)} \left[\max_{c\in\{0,1\}^*} \Pr_{(c',K)\leftarrow\mathsf{KEnc}(pk)}[c' = c] \right].$$

Definition 6. *We say that a KEM Γ is* ϵ-smooth *if we have $\mathsf{Smth}_\Gamma(k) \leq \epsilon$. Furthermore, we simply say that Γ is* smooth *if $\mathsf{Smth}_\Gamma(k)$ is negligible.*

For our results, we will utilize the following result shown by Bellare et al. [8][2].

Lemma 1. *[8] If a KEM Γ is* `IND-CPA` *secure, then Γ is smooth.*

2.4 Data Encapsulation Mechanism

A data encapsulation mechanism (DEM) D consists of the following two PPTAs:

DEnc: An encryption algorithm that takes a session-key $K \in \mathcal{K}$ and a plaintext $m \in \mathcal{M}$ as input, and outputs a ciphertext c. We write: $c \leftarrow \mathsf{DEnc}(K, m)$.

DDec: A deterministic decryption algorithm that takes K and c as input, and outputs a plaintext m or an error symbol $\perp$. We write: $m \leftarrow \mathsf{DDec}(K, c)$.

where $\mathcal{K}$ and $\mathcal{M}$ are a session-key space and a plaintext space of D, respectively.

We require $\mathsf{DDec}(K, \mathsf{DEnc}(K, m)) = m$ for all $K \in \mathcal{K}$ and all $m \in \mathcal{M}$.

We define the `IND-CCA2` advantage of an adversary $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$ against a DEM $D = (\mathsf{DEnc}, \mathsf{DDec})$ as follows:

$$\mathsf{Adv}^{\texttt{IND-CCA2}}_{D,\mathcal{A}} = |\Pr[K \leftarrow \mathcal{K}; (m_0, m_1, \mathsf{st}_\mathcal{A}) \leftarrow \mathcal{A}_1^{\mathcal{O}_1}; b \leftarrow \{0,1\}; c^* \leftarrow \mathsf{DEnc}(K, m_b); b' \leftarrow \mathcal{A}_2^{\mathcal{O}_2}(c^*, \mathsf{st}_\mathcal{A}) : b' = b] - \frac{1}{2}|,$$

where $\mathcal{O}_1(\cdot) = \mathcal{O}_2(\cdot) = \mathsf{DDec}(K, \cdot)$ is the decryption oracle, and $\mathcal{A}_2$ is not allowed to issue the challenge ciphertext c^* to $\mathcal{O}_2$.

Definition 7. *We say that a DEM D is (t, ϵ)-*`IND-CCA2` *secure if we have $\mathsf{Adv}^{\texttt{IND-CCA2}}_{D,\mathcal{A}} \leq \epsilon$ for any algorithm $\mathcal{A}$ running in time less than t. Furthermore, we simply say that D is* `IND-CCA2` *secure if $\mathsf{Adv}^{\texttt{IND-CCA2}}_{D,\mathcal{A}}$ is negligible for any PPTA $\mathcal{A}$.*

2.5 Identity-Based Encryption

An identity-based encryption (IBE) scheme Π consists of the following four (probabilistic) algorithms.

ISetup: A setup algorithm that takes 1^k (security parameter k) as input, and outputs a pair of global parameters prm and a master secret key msk. We write: $(\mathsf{prm}, \mathsf{msk}) \leftarrow \mathsf{ISetup}(1^\kappa)$.

[2]Strictly speaking, the authors of [8] show that if a KEM satisfies `IND-CCA2` security, then the KEM is smooth. However, in the proof they do not use any decryption query and thus their proof carries over to the case of `IND-CPA` secure KEMs.

IExt: A key extraction algorithm that takes prm, msk, and an identity $\mathsf{ID} \in \mathcal{I}_\Pi$ as input, and outputs a decryption key dk_{ID} corresponding to ID. We write: $dk_{\mathsf{ID}} \leftarrow \mathsf{IExt}(\mathsf{prm}, \mathsf{msk}, \mathsf{ID})$.

IEnc: An encryption algorithm that takes prm, $\mathsf{ID} \in \mathcal{I}_\Pi$, and a plaintext $m \in \mathcal{M}_\Pi$ as input, and outputs a ciphertext χ. We write: $\chi \leftarrow \mathsf{IEnc}(\mathsf{prm}, \mathsf{ID}, m)$.

IDec: A (deterministic) decryption algorithm that takes dk_{ID} and χ as input, and outputs a plaintext m or an error symbol $\perp$. We write: $m \leftarrow \mathsf{IDec}(dk_{\mathsf{ID}}, \chi)$.

where $\mathcal{I}_\Pi$ and $\mathcal{M}_\Pi$ are an identity space and a plaintext space of Π, respectively.

We require $\mathsf{IDec}(\mathsf{IExt}(\mathsf{prm}, \mathsf{msk}, \mathsf{ID}), \mathsf{IEnc}(\mathsf{prm}, \mathsf{ID}, m)) = m$ hold for all $(\mathsf{prm}, \mathsf{msk})$ output from ISetup, all $\mathsf{ID} \in \mathcal{I}_\Pi$, and all $m \in \mathcal{M}_\Pi$.

NM-sID-CPA Security. Non-malleability against selective identity, chosen plaintext attacks (NM-sID-CPA) of an IBE scheme Π is defined using the following NM-sID-CPA game between an adversary $\mathcal{A}$ and the NM-sID-CPA challenger $\mathcal{C}$:[3]

Init. Given 1^k, $\mathcal{A}$ commits the target identity ID^*.

Setup. $\mathcal{C}$ runs $(\mathsf{prm}, \mathsf{msk}) \leftarrow \mathsf{ISetup}(1^\kappa)$. Then $\mathcal{C}$ gives prm to $\mathcal{A}$ and keeps msk to itself.

Phase 1. $\mathcal{A}$ can adaptively issue extraction queries ID to $\mathcal{A}$, except that $\mathcal{A}$ is not allowed to issue the target identity ID^*. $\mathcal{C}$ responds to each query ID by running $dk_{\mathsf{ID}} \leftarrow \mathsf{IExt}(\mathsf{prm}, \mathsf{msk}, \mathsf{ID}_i)$ and returning dk_{ID} to $\mathcal{A}$.

Challenge. $\mathcal{A}$ specifies a probabilistic algorithm $\mathcal{M}^*_\Pi$ that outputs some element in the plaintext space $\mathcal{M}_\Pi$ where all the possible values output by $\mathcal{M}^*_\Pi$ are of equal length, and sends the description of $\mathcal{M}^*_\Pi$ to $\mathcal{C}$. $\mathcal{C}$ obtains m^* and $m^{\bar{*}}$ by running $\mathcal{M}^*_\Pi$ twice, computes a challenge ciphertext $\chi^* \leftarrow \mathsf{IEnc}(\mathsf{prm}, \mathsf{ID}^*, m^*)$, sends χ^* to $\mathcal{A}$, and keeps $m^{\bar{*}}$ to itself.

Phase 2. $\mathcal{A}$ can issue extraction queries in the same way as Phase 1.

Output. $\mathcal{A}$ outputs a vector of ciphertexts $\overrightarrow{\chi}' = (\chi'_1, \chi'_2, \ldots, \chi'_\ell)$, and a description of a relation $R(\cdot, \cdot)$ of arity $(\ell + 1)$, where the first input is a scalar and the second input is a vector of length ℓ.

$\mathcal{C}$ runs $dk_{\mathsf{ID}^*} \leftarrow \mathsf{IExt}(\mathsf{prm}, \mathsf{msk}, \mathsf{ID}^*)$, decrypts all elements in $\overrightarrow{\chi}'$ by running $m'_i \leftarrow \mathsf{IDec}(dk_{\mathsf{ID}^*}, \chi'_i)$ for $1 \le i \le \ell$, and obtains $\overrightarrow{m}' = (m'_1, m'_2, \ldots, m'_\ell)$.

We define $\mathcal{R}^*$ as an event that $[\chi^* \notin \overrightarrow{\chi}' \wedge \perp \notin \overrightarrow{m}' \wedge R(m^*, \overrightarrow{m}') = \mathtt{true}]$. We also define $\mathcal{R}^{\bar{*}}$ in the same way as $\mathcal{R}^*$ except that m^* is replaced with $m^{\bar{*}}$. We then define the NM-sID-CPA advantage of $\mathcal{A}$ attacking Π as follows:

$$\mathsf{Adv}^{\text{NM-sID-CPA}}_{\Pi,\mathcal{A}} = \Pr[\mathcal{R}^*] - \Pr[\mathcal{R}^{\bar{*}}].$$

[3]Here, we choose to write this security via a game between an adversary and a challenger, and not in the "experiment" style that we did for PKE schemes, for readability of the proof in Section 3.3.

Definition 8. *We say that an IBE scheme* Π *is* (t, ℓ, q_E, ϵ)*-NM-sID-CPA secure if we have* $\mathsf{Adv}^{\text{NM-sID-CPA}}_{\Pi,\mathcal{A}} \leq \epsilon$ *for any algorithm* $\mathcal{A}$ *that outputs at most* ℓ *ciphertexts and makes at most* q_E *extraction queries, and such that the total of* $\mathcal{A}$*'s running time, the running time of* $\mathcal{M}^*_\Pi$*, and the time needed to evaluate the relation* R *output by* $\mathcal{A}$ *is less than* t*. Furthermore, we simply say that* Π *is* ℓ*-NM-sID-CPA secure if* $\mathsf{Adv}^{\text{NM-sID-CPA}}_{\Pi,\mathcal{A}}$ *is negligible for any PPTA* $\mathcal{A}$ *that outputs at most* ℓ *ciphertexts.*

Remark. NM-sID-CPA security of an IBE scheme we use in this thesis is from [50, 5]. This type of non-malleability is called *comparison-based* non-malleability [13, 14], which was first introduced in [7] for PKE schemes. Note that in our definition of the NM-sID-CPA game, an adversary cannot gain the advantage if it outputs invalid ciphertexts (i.e., ciphertexts that decrypt to $\perp$). It was shown in [14, 90] that this type of non-malleability is, depending on attacks, equivalent to or weaker than the one where the adversary may gain the advantage even if it outputs invalid ciphertexts. In the original definition, the number l of ciphertexts need not be predetermined and can be dependent only on the adversary. (Asymtotically, ℓ can be any polynomial in the security parameter k.) If ℓ is bounded to be some predetermined value independently of an adversary, then it is weaker than the original definition. Myers and shelat [81] recently defined this weaker form of non-malleability for PKE schemes and call it ℓ-wise non-malleability. The relation between this ℓ-wise definition and the original definition is similar to the relation between the *bounded* CCA security [40] and the ordinary (unbounded) CCA security. In this thesis, we will need an IBE scheme non-malleability for $\ell = 1$ (i.e. 1-wise non-malleability). Moreover, it was shown in [50] that the selective identity security is strictly weaker than adaptive identity security for IBE schemes. Therefore, in summary, what we actually need is a very weak form of non-malleability for IBE schemes.

2.6 Tag-Based Encryption

A tag-based encryption (TBE) [78, 69] is an extension of a PKE scheme so that the encryption and decryption algorithms take an arbitrary string called *tag* as an additional input. TBE has also been called "PKE with labels" in several previous papers e.g. [45]. A TBE scheme Π consists of the following three (probabilistic) algorithms:

TKG**:** A key generation algorithm that takes 1^κ (security parameter κ) as input, and outputs a pair of a public key pk and a secret key sk. We write: $(pk, sk) \leftarrow \mathsf{TKG}(1^\kappa)$.

TEnc**:** An encryption algorithm that takes pk, a tag $\mathsf{tag} \in \mathcal{T}_\Pi$, and a plaintext $m \in \mathcal{M}_\Pi$ as input, and outputs a ciphertext χ. We write: $\chi \leftarrow \mathsf{TEnc}(pk, \mathsf{tag}, m)$.

TDec**:** A (deterministic) decryption algorithm that takes sk, tag, and χ as input, and outputs a plaintext m or an error symbol $\perp$. We write: $m \leftarrow \mathsf{TDec}(sk, \mathsf{tag}, \chi)$.

where $\mathcal{T}_\Pi$ and $\mathcal{M}_\Pi$ are a tag space and a plaintext space of Π, respectively.

We require $\mathsf{TDec}(sk, \mathsf{tag}, \mathsf{TEnc}(pk, \mathsf{tag}, m)) = m$ hold for all (pk, sk) output from TKG, all $\mathsf{tag} \in \mathcal{T}_\Pi$, and all $m \in \mathcal{M}_\Pi$.

NM-stag-wCCA Security. Here, we define non-malleability for TBE scheme. We note that non-malleability defined here is slightly different from and stronger than the one defined in [79]. For more details, see Section 3.3.3.

Non-malleability against selective tag, *weak* chosen ciphertext attacks (NM-stag-wCCA) of a TBE scheme Π is defined using the following NM-stag-wCCA game between an adversary $\mathcal{A}$ and the NM-stag-wCCA challenger $\mathcal{C}$:

Init. Given 1^k, $\mathcal{A}$ commits the target tag tag^*.

Setup. $\mathcal{C}$ runs $(pk, sk) \leftarrow \mathsf{TKG}(1^\kappa)$. Then $\mathcal{C}$ gives pk to $\mathcal{A}$ and keeps sk to itself.

Phase 1. $\mathcal{A}$ can adaptively issue decryption queries of the form (tag, χ), except that $\mathcal{A}$ is not allowed to issue the tag-ciphertext pair with $\mathsf{tag} = \mathsf{tag}^*$. $\mathcal{C}$ responds to each query (tag, χ) by running $m \leftarrow \mathsf{TDec}(sk, \mathsf{tag}, \chi)$ and returning m to $\mathcal{A}$.

Challenge. $\mathcal{A}$ specifies a probabilistic algorithm $\mathcal{M}^*_\Pi$ that outputs some element in the plaintext space $\mathcal{M}_\Pi$ where all the possible values output by $\mathcal{M}^*$ are of equal length, and sends the description of $\mathcal{M}^*$ to $\mathcal{C}$. $\mathcal{C}$ chooses m^* and $m^{\bar{*}}$ by running $\mathcal{M}^*$ twice, computes a challenge ciphertext $\chi^* \leftarrow \mathsf{TEnc}(pk, \mathsf{tag}^*, m^*)$, sends χ^* to $\mathcal{A}$, and keeps $m^{\bar{*}}$ to itself.

Phase 2. $\mathcal{A}$ can issue decryption queries in the same way as Phase 1.

Output. $\mathcal{A}$ outputs a vector of ciphertexts $\overrightarrow{\chi}' = (\chi'_1, \chi'_2, \ldots, \chi'_\ell)$, and a description of a relation $R(\cdot, \cdot)$ of arity $(\ell + 1)$, where the first input is a scalar and the second input is a vector of length ℓ.

$\mathcal{C}$ decrypts all elements in $\overrightarrow{\chi}'$ by running $m'_i \leftarrow \mathsf{TDec}(sk, \mathsf{tag}^*, \chi'_i)$ for $1 \le i \le \ell$, and obtains $\overrightarrow{m}' = (m'_1, m'_2, \ldots, m'_\ell)$.

We define the NM-stag-wCCA advantage of $\mathcal{A}$ attacking Π as follows:

$$\mathsf{Adv}^{\text{NM-stag-CCA}}_{\Pi,\mathcal{A}} = \Pr[\mathcal{R}^*] - \Pr[\mathcal{R}^{\bar{*}}],$$

where $\mathcal{R}^*$ and $\mathcal{R}^{\bar{*}}$ are defined in the same way as the NM-sID-CPA game (see Section 2.5).

Definition 9. *We say that a TBE scheme Π is (t, ℓ, q_D, ϵ)-NM-stag-wCCA secure if we have $\mathsf{Adv}^{\text{NM-stag-CCA}}_{\Pi,\mathcal{A}} \le \epsilon$ for any algorithm $\mathcal{A}$ that outputs at most ℓ ciphertexts and makes at most q_D decryption queries, and such that the total of $\mathcal{A}$'s running time, the running time of $\mathcal{M}^*_\Pi$, and the time needed to evaluate the relation R output by $\mathcal{A}$ is less than t. Furthermore, we simply say Π is ℓ-NM-stag-wCCA secure if $\mathsf{Adv}^{\text{NM-stag-CCA}}_{\Pi,\mathcal{A}}$ is negligible for any PPTA $\mathcal{A}$ that outputs at most ℓ ciphertexts.*

Remark. It is trivial to see that an NM-sID-CPA secure IBE scheme can be seen as a NM-stag-wCCA secure TBE scheme if we regard an identity for the IBE scheme as a tag for the TBE scheme, and as a decryption algorithm of the TBE scheme, an extraction and a decryption algorithms of the IBE scheme are combined in a natural way.

2.7 Encapsulation Scheme

Boneh and Katz [26] introduced the notion of an encapsulation scheme, which works as the main building block in the BK transformation. Roughly speaking, an encapsulation scheme is a kind of commitment scheme that commits a random value, so that it can be later recovered by using a decommitment. Formally, an encapsulation scheme E consists of the following three (probabilistic) algorithms:

ESetup: A setup algorithm that takes 1^κ (security parameter κ) as input, and outputs a public parameter prm. We write: $\mathsf{prm} \leftarrow \mathsf{ESetup}(1^k)$.

ECom: A commitment algorithm that takes a public parameter prm as input, and outputs a committed value $r \in \mathcal{V}$, a commitment $c \in \mathcal{C}$, and a decommitment $d \in \mathcal{D}$. We write: $(r, c, d) \leftarrow \mathsf{ECom}(\mathsf{prm})$.

ERec: A deterministic recovery algorithm that takes a public parameter prm, a commitment $c \in \mathcal{C}$, and a decommitment $d \in \mathcal{D}$ as input, and outputs a committed value $r \in \mathcal{V} \cup \{\perp\}$, We write: $r \leftarrow \mathsf{ERec}(\mathsf{prm}, c, d)$.

where $\mathcal{V}$, $\mathcal{C}$, and $\mathcal{D}$ are a committed value space, a commitment space, and a decommitment space of E, respectively. We require that $\mathsf{ERec}(\mathsf{prm}, c, d) = r$ hold for all prm output from ESetup and all $(r, c, d) \in \mathcal{V} \times \mathcal{C} \times \mathcal{D}$ output from $\mathsf{ECom}(\mathsf{prm})$.

Hiding Property. We define the advantage of an adversary $\mathcal{A}$ against hiding property of an encapsulation scheme $E = (\mathsf{ESetup}, \mathsf{ECom}, \mathsf{ERec})$ as follows:

$$\mathsf{Adv}^{\mathtt{Hiding}}_{E,\mathcal{A}}(k) = \left| \Pr \left[\begin{array}{ll} b \leftarrow \{0,1\}; & \mathsf{prm} \leftarrow \mathsf{ESetup}(1^k); \\ (r_1^*, c^*, d^*) \leftarrow \mathsf{ECom}(\mathsf{prm}); & : b' = b \\ r_0^* \leftarrow \mathcal{V}; & b' \leftarrow \mathcal{A}(\mathsf{prm}, r_b^*, c^*) \end{array} \right] - \frac{1}{2} \right|.$$

Definition 10. *We say that an encapsulation scheme E is (t, ϵ)-hiding if we have $\mathsf{Adv}^{\mathtt{Hiding}}_{E,\mathcal{A}}(k) \leq \epsilon$ for any algorithm $\mathcal{A}$ running in time less than t. Furthermore, we simply say that E is hiding if $\mathsf{Adv}^{\mathtt{Hiding}}_{E,\mathcal{A}}(k)$ is negligible for any PPTA $\mathcal{A}$.*

Binding Property. We define the advantage of an adversary $\mathcal{A}$ against binding property of an encapsulation scheme $E = (\mathsf{ESetup}, \mathsf{ECom}, \mathsf{ERec})$ as follows:

$$\mathsf{Adv}^{\mathtt{Binding}}_{E,\mathcal{A}}(k) = \Pr \left[\begin{array}{ll} & \mathsf{prm} \leftarrow \mathsf{ESetup}(1^k); \\ (r^*, c^*, d^*) \leftarrow \mathsf{ECom}(\mathsf{prm}); & : \mathsf{ERec}(\mathsf{prm}, c^*, d') \notin \{\perp, r^*\} \wedge d' \neq d^* \\ d' \leftarrow \mathcal{A}(\mathsf{prm}, r^*, c^*, d^*) & \end{array} \right].$$

Definition 11. *We say that an encapsulation scheme E is (t, ϵ)-binding if we have $\mathsf{Adv}^{\mathtt{Binding}}_{E,\mathcal{A}}(k) \leq \epsilon$ for any algorithm $\mathcal{A}$ running in time less than t. Furthermore, we simply say that E is binding if $\mathsf{Adv}^{\mathtt{Binding}}_{E,\mathcal{A}}(k)$ is negligible for any PPTA $\mathcal{A}$.*

2.8 Pseudorandom Generator

Let $G : \mathcal{D}_k \to \mathcal{R}_k$ be a function with $|\mathcal{D}_k| \leq |\mathcal{R}_k|$. We define the advantage of an adversary $\mathcal{A}$ against pseudorandomness of G as follows:

$$\mathsf{Adv}^{\mathtt{PRG}}_{G,\mathcal{A}}(k) = |\Pr[b \leftarrow \{0,1\}; x^* \leftarrow \mathcal{D}_k; y_1^* \leftarrow G(x^*); y_0^* \leftarrow \mathcal{R}_k; b' \leftarrow \mathcal{A}(1^k, y_b^*) : b' = b] - \frac{1}{2}|.$$

Definition 12. *We say that a function G is a (t, ϵ)-pseudorandom generator (PRG) if we have $\mathsf{Adv}^{\mathtt{PRG}}_{G,\mathcal{A}}(k) \leq \epsilon$ for any algorithm $\mathcal{A}$ running in time less than t. Furthermore, we simply say that G is a PRG if $\mathsf{Adv}^{\mathtt{PRG}}_{G,\mathcal{A}}(k)$ is negligible for any PPTA $\mathcal{A}$.*

Near Collision Resistance (for Predetermined Parts of Output). Boldyreva and Fischlin [19] introduced the notion of near collision resistance (NCR) for predetermined parts of output of a PRG. Roughly speaking, NCR property ensures that given a randomly chosen input $x \in \mathcal{D}$, no adversary can efficiently find another input $x'(\neq x) \in \mathcal{D}$ such that the predetermined parts of output becomes identical. Since an adversary cannot have a control over one of the inputs, it is more related to target collision resistance [83, 12] than ordinary (any) collision resistance [44]. According to the authors of [19] "near collision resistance" is named after [16].

In this thesis, we will only use k-least significant bits of output of G as the predetermined parts for NCR property, where κ is the security parameter. Formally, we define the advantage of an adversary $\mathcal{A}$ against NCR for k-least significant bits of output of G as follows:

$$\mathsf{Adv}^{\mathtt{NCR}\text{-}k\text{-}\mathtt{LSB}}_{G,\mathcal{A}}(k) = \Pr[x^* \leftarrow \mathcal{D}; x' \leftarrow \mathcal{A}(1^k, x^*) : k\text{-}\mathsf{LSB}(G(x')) = k\text{-}\mathsf{LSB}(G(x^*)) \wedge x' \neq x^*].$$

Definition 13. *We say that a function (or PRG) G is (t, ϵ)-near collision resistant for k-least significant bits of output (NCR-k-LSB) if we have $\mathsf{Adv}^{\mathtt{NCR}\text{-}k\text{-}\mathtt{LSB}}_{G,\mathcal{A}}(k) \leq \epsilon$ for any algorithm $\mathcal{A}$ running in time less than t. Furthermore, we simply say that G is NCR-k-LSB if $\mathsf{Adv}^{\mathtt{NCR}\text{-}k\text{-}\mathtt{LSB}}_{G,\mathcal{A}}(k)$ is negligible for any PPTA $\mathcal{A}$.*

2.9 Target Collision Resistant Hash Function

Let $H : \mathcal{D}_k \rightarrow \mathcal{R}_k$ be a hash function with $|\mathcal{D}_k| \geq |\mathcal{R}_k|$. We define the advantage of an adversary $\mathcal{A}$ against target collision resistance of H as follows:

$$\mathsf{Adv}^{\mathtt{TCR}}_{H,\mathcal{A}}(k) = \Pr[x^* \leftarrow \mathcal{D}_k; x' \leftarrow \mathcal{A}(x^*) : H(x') = H(x^*) \wedge x' \neq x^*].$$

Definition 14. *We say that H is a (t, ϵ)-target collision resistant hash function (TCRHF) if we have $\mathsf{Adv}^{\mathtt{TCR}}_{H,\mathcal{A}}(k) \leq \epsilon$ for any algorithm $\mathcal{A}$ running in time less than t. Furthermore, we simply say that H is a TCRHF if $\mathsf{Adv}^{\mathtt{TCR}}_{H,\mathcal{A}}(k)$ is negligible for any PPTA $\mathcal{A}$.*

2.10 Pseudorandom Function

Let $F : \{0,1\}^k \times \mathcal{D}_k \rightarrow \mathcal{R}_k$ be an efficiently computable keyed-function, where the first argument of F is regarded as a key (also called a *seed* or an "index"). We write $F_K(\cdot)$ to mean $F(K, \cdot)$. We define the advantage of an adversary $\mathcal{A}$ against the pseudorandomness of F as follows:

$$\mathsf{Adv}^{\mathtt{PRF}}_{F,\mathcal{A}}(k) = |\Pr[K \leftarrow \{0,1\}^k : 1 \leftarrow \mathcal{A}^{F_K(\cdot)}(1^k)] - \Pr[RF \leftarrow \mathsf{FUNK}_{\mathcal{D}_k \rightarrow \mathcal{R}_k} : 1 \leftarrow \mathcal{A}^{RF(\cdot)}(1^k)]|,$$

where $\mathsf{FUNK}_{\mathcal{D}_k \rightarrow \mathcal{R}_k}$ is a set of all functions whose domain and range are $\mathcal{D}_k$ and $\mathcal{R}_k$, respectively.

Definition 15. *We say that F is (t, ϵ)-pseudorandom function (PRF) if we have $\mathsf{Adv}^{\mathtt{PRF}}_{F,\mathcal{A}}(k) \leq \epsilon$ for any algorithm $\mathcal{A}$ running in time less than t. Furthermore, we simply say that F is a PRF if $\mathsf{Adv}^{\mathtt{PRF}}_{F,\mathcal{A}}(k)$ is negligible for any PPTA $\mathcal{A}$.*

2.11 Signature

A signature scheme Σ consists of the following three PPTAs:

SKG**:** A key generation algorithm that takes 1^k (security parameter k) as input, and outputs a verification/signing key pair $(vk, sigk)$. We write: $(vk, sigk) \leftarrow \mathsf{SKG}(1^k)$.

Sign**:** A signing algorithm that takes $sigk$ and a message $m \in \mathcal{M}$ as input, and outputs a signature σ. We write: $\sigma \leftarrow \mathsf{Sign}(sigk, m)$.

SVrfy**:** A deterministic verification algorithm that takes vk, m, and σ as input, and outputs $\top$ if σ is a valid message on m under the verification key vk, or $\bot$ otherwise. We write: $\top/\bot \leftarrow \mathsf{SVrfy}(vk, m, \sigma)$.

where $\mathcal{M}$ is a message space of Σ.

We require $\mathsf{SVrfy}(vk, m, \mathsf{Sign}(sigk, m)) = \top$ for all $(vk, sigk)$ output from SKG and all $m \in \mathcal{M}$.

We define the advantage of an adversary $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$ against strong one-time (SOT) security of a signature scheme $\Sigma = (\mathsf{SKG}, \mathsf{Sign}, \mathsf{SVrfy})$ as follows:

$$\mathsf{Adv}^{\mathtt{SOT}}_{\Sigma,\mathcal{A}}(k) = \Pr[(vk, sigk) \leftarrow \mathsf{SKG}(1^k); (m, \mathsf{st}_{\mathcal{A}}) \leftarrow \mathcal{A}_1(vk); \sigma \leftarrow \mathsf{Sign}(sigk, m); \\ (m', \sigma') \leftarrow \mathcal{A}_2(\sigma, \mathsf{st}_{\mathcal{A}}) : \mathsf{SVrfy}(vk, m', \sigma') = \top \wedge (m', \sigma') \neq (m, \sigma)]$$

Definition 16. *We say that a signature scheme Σ is (t, ϵ)-strongly one-time secure if we have $\mathsf{Adv}^{\mathtt{SOT}}_{\Sigma,\mathcal{A}}(k) \leq \epsilon$ for any algorithm $\mathcal{A}$ running in time less than t. Furthermore, we simply say that Σ is strongly one-time secure if $\mathsf{Adv}^{\mathtt{SOT}}_{\Sigma,\mathcal{A}}(k)$ is negligible for any PPTA $\mathcal{A}$.*

2.12 Message Authentication Code

A message authentication code (MAC) scheme M consists of the following two algorithms:

Mac**:** A MAC tag generation algorithm that takes a key $K \in \{0,1\}^k$ and a message $m \in \mathcal{M}$ as input, and outputs a MAC tag τ. We write: $\tau \leftarrow \mathsf{Mac}(K, m)$.

MVrfy**:** A deterministic verification algorithm that takes a key $K \in \{0,1\}^k$, m and τ as input, and outputs $\top$ if σ is a valid message on m under the verification key K, or $\bot$ otherwise. We write: $\top/\bot \leftarrow \mathsf{MVrfy}(K, m, \sigma)$.

where $\mathcal{M}$ is a message space of M.

We require $\mathsf{MVrfy}(K, m, \mathsf{Mac}(K, m)) = \top$ for all $K \in \{0,1\}^k$ and all $m \in \mathcal{M}$.

We define the advantage of an adversary $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$ against strong one-time (SOT) security of a MAC $M = (\mathsf{Mac}, \mathsf{MVrfy})$ as follows:

$$\mathsf{Adv}^{\mathtt{SOT}}_{M,\mathcal{A}}(k) = \Pr[K \leftarrow \{0,1\}^k; (m, \mathsf{st}_{\mathcal{A}}) \leftarrow \mathcal{A}_1(1^k); \tau \leftarrow \mathsf{Mac}(K, m); \\ (m', \tau') \leftarrow \mathcal{A}_2(\tau, \mathsf{st}_{\mathcal{A}}) : \mathsf{MVrfy}(Km', \tau') = \top \wedge (m', \tau') \neq (m, \tau)]$$

Definition 17. *We say that a MAC scheme M is (t, ϵ)-strongly one-time secure if we have $\mathsf{Adv}^{\mathtt{SOT}}_{M,\mathcal{A}}(k) \leq \epsilon$ for any algorithm $\mathcal{A}$ running in time less than t. Furthermore, we simply say that M is strongly one-time secure if $\mathsf{Adv}^{\mathtt{SOT}}_{M,\mathcal{A}}(k)$ is negligible for any PPTA $\mathcal{A}$.*

2.13 One-Way Function

Let $f : \mathcal{D}_k \to \mathcal{R}_k$ be a function where we can efficiently sample a uniformly random element from $\mathcal{D}_k$. We define the advantage of an adversary $\mathcal{A}$ against one-wayness of f as follows:

$$\mathsf{Adv}^{\mathtt{OW}}_{f,\mathcal{A}}(k) = \Pr[x \leftarrow \mathcal{D}_k; y \leftarrow f(x); x' \leftarrow \mathcal{A}(1^k, y) : f(x') = y].$$

Definition 18. *We say that f is a (t, ϵ)-one-way function (OWF) if we have $\mathsf{Adv}^{\mathtt{OW}}_{f,\mathcal{A}}(k) \leq \epsilon$ for any algorithm $\mathcal{A}$ running in time less than t. Furthermore, we simply say that f is a OWF if $\mathsf{Adv}^{\mathtt{OW}}_{f,\mathcal{A}}(k)$ is negligible for any PPTA $\mathcal{A}$.*

Chapter 3

Practical Constructions: IBE-to-PKE Transformations

Contents

3.1 Introduction 17
3.1.1 Background and Motivation 17
3.1.2 Our Contribution 18
3.1.3 Related Works 20
3.1.4 Organization of This Chapter 21
3.2 The Boneh-Katz Transformation 21
3.3 Proposed Transformation using Non-malleable IBE 23
3.3.1 Construction 23
3.3.2 Security 23
3.3.3 Applying the Transformation to Tag-Based Encryption 32
3.4 Proposed Encapsulation Scheme 34
3.4.1 Construction 35
3.4.2 Security 35
3.4.3 Concrete Instantiation of PRG with NCR Property 37
3.4.4 PRG with Near Collision Resistance from Any One-Way Permutation 38
3.5 Comparison 39
3.6 Conclusion 41

3.1 Introduction

3.1.1 Background and Motivation

Studies on constructing and understanding efficient public key encryption (PKE) schemes secure against chosen ciphertext attacks (CCA) [93, 47] are important research topics in the area of cryptography. Among several approaches towards CCA secure PKE schemes, one of the promising approaches is the "IBE-to-PKE" transformation paradigm [34], which is a method to obtain CCA secure PKE schemes from identity-based encryption (IBE) schemes.

In [34], Canetti, Halevi, and Katz showed a generic construction of CCA secure PKE schemes from any semantically secure IBE scheme and one-time signature scheme (we call this IBE-to-PKE transformation the *CHK transformation*). This construction is fairly simple, and specifically, its ciphertext consists of (χ, vk, σ) where χ is a ciphertext of the underlying IBE scheme (under identity "vk"), vk is a verification key of a one-time signature scheme, and σ is a valid signature of χ (under verification key vk). However, due to the use of a one-time signature, ciphertext length of the resulting scheme becomes longer than that of the underlying IBE scheme for $|vk|$ and $|\sigma|$, which might result in significantly large ciphertexts.

This method was later improved by Boneh and Katz [26] (we call the *BK transformation*) by replacing a one-time signature in the CHK transformation with an *encapsulation* scheme and a message authentication code (MAC) scheme, where an encapsulation scheme (the notion of which is introduced in the same paper [26]) is a special kind of commitment scheme that commits a random value. This method has a possibility of drastically reducing computation costs for encryption and decryption algorithms and ciphertext size of the transformed PKE scheme, compared to the CHK transformation. However, its ciphertext size directly depends on the size of parameters (commitment, decommitment, and the committed value) of the underlying encapsulation scheme, and thus an encapsulation scheme with large parameters still yields a large ciphertext for a transformed PKE scheme. Since the concrete encapsulation scheme that Boneh and Katz presented in [26] (we call the *BK encapsulation scheme*) had somewhat large parameters, PKE schemes transformed via the BK transformation could not be as size-efficient as existing practical CCA secure PKE schemes, e.g. [42, 75].

There are some IBE-to-PKE transformations which can be applied to IBE schemes with specific properties or structures [3, 106]. Although these transformations achieve PKE schemes with shorter ciphertext than those obtained from CHK and BK transformations, these transformations sacrifice the generality of IBE schemes.

Hence, it is still desired to further achieve IBE-to-PKE transformations that achieve PKE with shorter ciphertext length, without loosing generality. Moreover, there is still a room for further improvement for the BK transformation in terms of ciphertext size, by designing an encapsulation scheme with small parameter sizes.

3.1.2 Our Contribution

We have developed two appoaches regarding this topic, each of them are explained in detail in the following subsections.

New IBE-to-PKE Transformation using Non-malleable IBE

We present a very *simple* IBE-to-PKE transformation which is fairly generic and practical. In contrast to the previous transformations [34, 26] which require semantic security [60] for the underlying IBE scheme, our proposed method requires *non-malleability* [47].

Informally, for a given IBE scheme IBE, we generate a ciphertext χ of a PKE scheme which is converted from IBE via our method as follows:

$$\chi = (\ f(r), \mathsf{IBE.Enc}(\mathsf{prm}, \text{“}f(r)\text{”}, (m||r))\ ),$$

where m is a plaintext to be encrypted, r is a randomness, the second component of χ is an encryption of $(m||r)$ with the encryption algorithm of a given IBE scheme IBE under the identity "$f(r)$", and f is a one-way function (OWF). It should be noticed that only a

OWF is directly used as an additional building block and thus fairly simple while in [34, 26] more complicated tools, e.g. one-time signatures, are required (though these tools can be obtained from OWFs in theory). As seen in the above construction, ciphertext overhead of our construction is that of IBE plus $|r| + |f(r)| =$ (256-bit) for 128-bit security, and this is fairly efficient compared to the Boneh-Katz (BK) construction [26].

An obvious and crucial disadvantage of our proposed transformation is that it requires a stronger assumption for the underlying IBE scheme, non-malleability. It is well known that non-malleability is a significantly stronger notion of security than semantic security, and in fact, except for CCA secure IBE schemes, no practical non-malleable IBE scheme is currently known. [1] Thus, we have to honestly remark that our proposal cannot be seen as a direct improvement of the previous generic IBE-to-PKE transformations [34, 26]. However, once we have an IBE scheme which is proved (or can be assumed) to be non-malleable, an efficient CCA secure PKE scheme can be immediately obtained via our transformation. Also, we believe that the simpleness of our transformation itself is theoretically interesting.

Our proof technique for the proposed method will be of another theoretical interest. Since in the security proof, there exists a non-trivial issue which cannot be treated by straightforward application of known techniques, we have to concurrently carry out a totally different proof strategy. Hence, we develop a dedicated proof technique for handling two different strategies simultaneously.

Though there are several definitions for non-malleability so far [47, 7, 13, 89, 5], the non-malleability for IBE our transformation requires is a very weak one. See the remark in the Section 2.5 that is given after the formal definition of the non-malleability for IBE scheme.

Improving Boneh-Katz Transformation with Efficient Encapsualtion Scheme

Focusing on the size-efficiency of the BK transformation, we present an efficient encapsulation scheme. Specifically, for 128-bit security, the ciphertext overhead (the difference of size between the whole ciphertext and its plaintext) of a PKE scheme obtained via the BK transformation with our encapsulation scheme can be that of the underlying IBE scheme plus 384-bit, while that of a PKE scheme via the BK transformation with their encapsulation scheme needs to be that of the underlying IBE scheme plus at least 704-bit.

The main building block used in the proposed encapsulation scheme is a pseudorandom generator (PRG) with a special property called *near collision resistance for predetermined parts of output* (NCR for short), which was first introduced and used by Boldyreva and Fischlin in [19]. Roughly speaking, NCR property is target collision resistance [83, 12] for some part of output. We only consider κ-least significant bits of output as the predetermined parts of NCR property, where κ is the security parameter. See Section 2.8 for more details.

We also show concrete instantiations of a PRG with NCR property. One construction is a slight modification of a practical PRG [1] used in practice which is based on cryptographic hash functions such as SHA-1. If we can assume that the hash functions used in the PRG satisfy target collision resistance, we immediately obtain a PRG with NCR property. Though we can provide only a heuristic analysis for this construction, we believe that it is fairly reasonable to assume that this practical PRG satisfies NCR property and we can use it in practical scenarios.

[1] In theory, it is possible to construct non-malleable IBE schemes generically from any semantically secure IBE schemes using the techniques shown by Pass et al. [89] and Choi et al. [38] (while it is not known how to generically construct CCA secure IBE schemes).

In order to confirm that a PRG with NCR property, though seemingly strong, is actually a fairly weak primitive, we also address how to generically construct such a PRG from any one-way permutation. Interestingly, the construction is the well-known one by Blum and Micali [18] and Yao [105] itself. Namely, the Blum-Micali-Yao PRG has NCR property as it is.

3.1.3 Related Works

Identity-Based Encryption. Here, we briefly review IBE schemes. The concept of the identity-based encryption was introduced by Shamir [99]. Roughly speaking, an IBE scheme is a PKE scheme where one can use an arbitrary string (e.g., an email address) as one's public key. Boneh and Franklin [24] proposed a first efficient scheme (in the random oracle model [10]) under the computational bilinear Diffie-Hellman (CBDH) assumption. They also defined the security models for IBE schemes. Sakai, Ohgishi, and Kasahara [97] independently proposed an IBE scheme with basically the same structure as the Boneh-Franklin IBE scheme (without proper security discussions). In the same year, Cocks [39] also proposed an IBE scheme secure in the random oracle model based on the decisional quadratic residuosity (QR) assumption. Horwitz and Lynn [68] introduced a notion of the hierarchical IBE (HIBE) which supports hierarchical structure of identities and Gentry and Silverberg [54] achieved the first scheme secure in the random oracle model under the CBDH assumption. Canetti, Halevi, and Katz [33] introduced a weaker security model called *selective identity security*, and proposed an IBE scheme with this security without using random oracles under the decisional bilinear Diffie-Hellman (DBDH) assumption. Boneh and Boyen [20] proposed two efficient IBE schemes which are selective identity secure in the standard model, and in the following this works, Boneh and Boyen [21] and Waters [102] proposed fully secure (H)IBE schemes (under the DBDH assumption). Boneh, Boyen, and Goh [22] proposed an HIBE scheme with constant ciphertext size, but was only selective identity secure in the standard model (can be fully secure in the randon oracle model) under a relatively complex q-type assumption. Gentry [51] proposed a practical IBE scheme which has short parameters, tight security reduction, and *anonymity* of identities, but required a q-type assumption. Boneh, Gentry, and Hamberg [25] constructed efficient variants of the Cocks scheme [39] which can be proven to be fully secure in the random oracle model based on the QR assumption, or in the standard model based on the "interactive" QR assumption. Gentry and Halevi [52] constructed a first fully secure HIBE scheme which allows a polynomially-many level hierarchy of identities under a q-type assumption. Waters [103] proposed a first (H)IBE scheme with short parameters under simple assumptions (the decisional Linear and the DBDH assumptions), using a new concept called *dual system encryption*. Using the same technique, Lewko and Waters [76] constructed an HIBE scheme with shorter parameters. Gentry, Peikert, and Vaikuntanathan [53] constructed an IBE scheme which is based on the worst case hardness of standard lattice problems.

Other IBE-to-PKE Transformations and Tag-Based Encryption. As mentioned above, one promising approach for constructing CCA secure PKE schemes is to transform an IBE scheme via the IBE-to-PKE transformation paradigm. We review them here. Canetti, Halevi, and Katz [34] proposed a generic method for obtaining CCA secure PKE schemes. Following [34], there have been some attempts to construct practical CCA secure PKE schemes by using specific algebraic properties of underlying IBE schemes, and especially, based on this approach Boyen, Mei, and Waters [27] proposed the currently best known CCA secure

PKE schemes in terms of ciphertext length by using certain specific IBE schemes [20, 102]. Boneh and Katz [26] improved the efficiency of [34] by replacing a one-time signature with a combined use of MAC and a new primitive called an *encapsulation* scheme, which is essentially a (non-interactive) commitment scheme where we can only commit to random messages and can be realized by a combination of a pairwise-independent hash function and a target collision resistant hash function. In order to further improve the efficiency of [26], Matsuda et al. [80] showed efficient constructions of encapsulation schemes, which require either a one-way permutation or a hash function which satisfies some practical assumptions.

Using chameleon hash functions [74], Abe et al. [3] proposed several IBE-to-PKE transformations for *partitioned* identity-based key encapsulation mechanisms and constructed several CCA secure PKE schemes via the *Tag-KEM/DEM* paradigm [4]. Zhang [106] independently proposed two transformations that also use chameleon hash functions, where the first transformation is applicable to schemes with *separable* property which are similar to the *partitioned* property [3] and the second transformation is applicable generically but requires stronger security for the used chameleon hash function. In this thesis, we do not aim at a size-efficient IBE-to-PKE transformation at the cost of "generality" for the underlying IBE, so that the transformation is widely applicable. Moreover, a chameleon hash function usually yields a computation of exponentiations, which is heavier compared to computation of "symmetric-key" primitives such as block ciphers.

Kiltz [69] showed that the IBE-to-PKE transformation paradigm can be generically applied to *tag-based encryption* (TBE) schemes [78] of appropriate security, which are weaker primitives than IBE schemes. Similar results can be obtained from our result (see Section 3.3.3).

3.1.4 Organization of This Chapter

In Section 3.2, we review the BK transformation [26] and its mechanism, as well as the encapsulation scheme presented in [26], and discuss the efficiency.

In Section 3.3, we show our first approach regarding the IBE-to-PKE transformation paradigm. In particular, we propose a new generic IBE-to-PKE transformation which can be applied to any non-malleable IBE schemes, and discuss its security. There, we also discuss non-malleability for TBE schemes and apply our transformation to non-malleable TBE schemes.

In Section 3.4, we present our proposed encapsulation scheme from a PRG with NCR property and prove its security. We also show a practical instantiation of such PRG from a cryptographic hash function, as well as how to construct it from any one-way permutation.

Then in Section 3.5, we compare our result with previous generic IBE-to-PKE transformations. Finally in Section 3.5, we compare our result with other generic IBE-to-PKE transformations. Section 3.6 is the conclusion of this chapter.

Publication Information. The results shown in this chapter were presented as [b] and [c] (see Appendix A). In particular, the results in Section 3.3 were shown in [c], and those in Section 3.4 were shown in [b].

3.2 The Boneh-Katz Transformation

In this section, we briefly review the IBE-to-PKE transformation by Boneh and Katz [26]. Let $\Pi = (\mathsf{ISetup}, \mathsf{IExt}, \mathsf{IEnc}, \mathsf{IDec})$ be an IBE scheme, $E = (\mathsf{ESetup}, \mathsf{ECom}, \mathsf{ERec})$ be an

$\mathsf{PKG}(1^k)$:	$\mathsf{PEnc}(PK, m)$:
$(\mathsf{msk}, \mathsf{prm}_I) \leftarrow \mathsf{ISetup}(1^k)$	$(r, c, d) \leftarrow \mathsf{ECom}(\mathsf{prm}_E)$
$\mathsf{prm}_E \leftarrow \mathsf{ESetup}(1^k)$	$y \leftarrow \mathsf{IEnc}(\mathsf{prm}_I, c, (m\|\|d))$
$SK \leftarrow \mathsf{msk}$; $PK \leftarrow (\mathsf{prm}_I, \mathsf{prm}_E)$	$\mathsf{tag} \leftarrow \mathsf{Mac}(r, y)$
Output (SK, PK).	$\chi \leftarrow \langle c, y, \mathsf{tag}\rangle$
	Output χ.
$\mathsf{PDec}(SK, \chi)$:	
Parse χ as $\langle c, y, \mathsf{tag}\rangle$.; $dk_c \leftarrow \mathsf{IExt}(\mathsf{prm}_I, \mathsf{msk}, c)$	
$(m\|\|d) \leftarrow \mathsf{IDec}(dk_c, y)$ (if this returns $\perp$ then output $\perp$ and stop.)	
$r \leftarrow \mathsf{ERec}(\mathsf{prm}_E, c, d)$ (if this returns $\perp$ then output $\perp$ and stop.)	
Output m if $\mathsf{MVrfy}(r, y, \mathsf{tag}) = \top$. Otherwise output $\perp$.	

Figure 3.1: The Boneh-Katz Transformation

encapsulation scheme, and $\Sigma = (\mathsf{Mac}, \mathsf{MVrfy})$ be a MAC scheme. Then, a PKE scheme $\Pi' = (\mathsf{PKG}, \mathsf{PEnc}, \mathsf{PDec})$ obtained via the BK transformation is as shown in Fig. 3.1.

The following states the security of the PKE scheme obtained via the BK transformation.

Theorem 1. *([26, 23]) If the underlying IBE scheme* Π *is* (t, q, ϵ_{ibe})*-IND-sID-CPA secure, the encapsulation scheme* E *is* (t, ϵ_{hide})*-hiding and* (t, ϵ_{bind})*-binding, and the MAC scheme* Σ *is* (t, ϵ_{mac})*-strongly one-time secure, then the PKE scheme* Π' *in Fig. 3.1 is* $(t - o(t), q, 4\epsilon_{ibe} + 2\epsilon_{hide} + \epsilon_{bind} + q\epsilon_{mac})$-IND-CCA2 *secure.*

Notice that the overhead of ciphertext size from that of the underlying IBE scheme is caused by a commitment c, a decommitment d, and a MAC tag tag. Since the size of a MAC tag can be k-bit for k-bit security and is optimal, designing an encapsulation scheme such that the sizes of parameters (c, d) are small is desirable for obtaining a PKE scheme with a small ciphertext overhead.

In [26], the authors also showed a concrete construction of an encapsulation scheme. Here, we briefly review their encapsulation scheme. $\mathsf{ESetup}(1^k)$ picks a target collision resistant hash function (TCRHF) TCR and a pairwise-independent hash function (PIHF) h, and outputs $\mathsf{prm} \leftarrow (\mathsf{TCR}, h)$. $\mathsf{ECom}(\mathsf{prm})$ picks a decommitment d randomly, computes $c \leftarrow \mathsf{TCR}(d)$ and $r \leftarrow h(d)$, then outputs (r, c, d). $\mathsf{ERec}(\mathsf{prm}, c, d)$ checks whether $\mathsf{TCR}(d) = c$ or not, and outputs $r \leftarrow h(d)$ if this holds or $\perp$ otherwise.

Their scheme only uses a TCRHF and a PIHF for both the commitment and the recovery algorithms and thus is fairly efficient in terms of computation cost. However, due to the leftover hash lemma [63] used to show hiding property, we need to set d to be at least 448-bit for 128-bit security (it achieves hiding property in a statistical sense). Thus, even though we use an efficient IBE scheme such as [20] as the underlying IBE scheme in the BK transformation, it results in a PKE scheme with somewhat large ciphertext because of the size of d. However, as the authors of [26] pointed out, it is important to note that we do not need "statistical security" for neither hiding nor binding properties. We only need "computational security" for both. (Our proposed encapsulation scheme in the next section actually achieves them in computational sense.)

As we have seen in this section, designing a size-efficient encapsulation scheme directly leads to the improvement for ciphertext overhead of a PKE scheme obtained via the BK transformation. Thus, in the next section we present a new efficient encapsulation scheme.

$\mathsf{PKG}(1^\kappa)$:	$\mathsf{PEnc}(PK, m)$:
$(\mathsf{prm}, \mathsf{msk}) \leftarrow \mathsf{ISetup}(1^\kappa)$	$r \leftarrow \{0,1\}^\gamma$; $\mathsf{ID} \leftarrow f(r)$
Pick a OWF f.	$y \leftarrow \mathsf{IEnc}(\mathsf{prm}, \mathsf{ID}, (m\|\|r))$
$PK \leftarrow (\mathsf{prm}, f)$; $SK \leftarrow \mathsf{msk}$	$\chi \leftarrow (\mathsf{ID}, y)$
Output (PK, SK).	Output χ.
$\mathsf{PDec}(SK, \chi)$:	
Parse χ as (ID, y).; $dk_{\mathsf{ID}} \leftarrow \mathsf{IExt}(\mathsf{prm}, \mathsf{msk}, \mathsf{ID})$	
$(m\|\|r)$ / $\perp \leftarrow \mathsf{IDec}(dk_{\mathsf{ID}}, y)$ (if $\perp$ then output $\perp$ and stop.)	
Output m if $f(r) = \mathsf{ID}$. Otherwise output $\perp$.	

Figure 3.2: The Proposed IBE-to-PKE Transformation

3.3 Proposed Transformation using Non-malleable IBE

In this section, we give the details of the construction of our simple IBE-to-PKE transformation from any NM-sID-CPA secure IBE scheme.

The idea behind the construction is as follows. Suppose f is a OWF. In our construction, a randomness r is encrypted as a part of a plaintext of the underlying non-malleable IBE scheme using $f(r)$ as an identity. In the decryption, the relation between r and $f(r)$ is then used to check the validity of the ciphertext. Constructed like this, it seems hard to make a valid ciphertext without knowing the exact value of r. Moreover, due to non-malleability of the IBE scheme and one-wayness of f, an adversary given a target ciphertext cannot make any alternation on it with keeping the consistency of r and $f(r)$.

3.3.1 Construction

Let $\Pi = (\mathsf{ISetup}, \mathsf{IExt}, \mathsf{IEnc}, \mathsf{IDec})$ be a non-malleable IBE scheme and $f : \{0,1\}^\gamma \to \mathcal{I}_\Pi$ be a OWF, where $\mathcal{I}_\Pi$ is the identity space of Π. Then we construct a PKE scheme $\Pi' = (\mathsf{PKG}, \mathsf{PEnc}, \mathsf{PDec})$ as in Fig. 3.4. Suppose the plaintext space of Π' is $\mathcal{M}_{\Pi'}$, then we require that the plaintext space $\mathcal{M}_\Pi$ of the underlying IBE scheme Π satisfy $\mathcal{M}_{\Pi'} \times \{0,1\}^\gamma \subseteq \mathcal{M}_\Pi$. We also require that length of all elements in $\mathcal{I}_\Pi$, the output space of f as well as the identity space of Π, be of equal length and fixed. Typically, length γ of the randomness will be the security parameter κ.

In terms of the construction of the transformation, ours is fairly simpler compared to other generic IBE-to-PKE transformations [34, 26], since only a OWF f, the weakest primitive, is directly used as an additional building block.

3.3.2 Security

Before going into a formal security proof, we give an intuitive explanation on how CCA security is proved. In the security proof, we construct an adversary $\mathcal{B}$ which breaks NM-sID-CPA security using an IND-CCA adversary $\mathcal{A}$ attacking the proposed PKE scheme Π'. The adversary $\mathcal{B}$'s task is to output a ciphertext y' and a relation R such that R holds between the plaintext of y' and that of $\mathcal{B}$'s challenge ciphertext y^*.

Roughly, the proof strategy of the previous generic IBE-to-PKE transformations [34, 26] is that $\mathcal{A}$'s decryption queries encrypted under identities different from the target identity of the adversary $\mathcal{B}$ are responded perfectly using $\mathcal{B}$'s own extraction queries, and the probability

that $\mathcal{A}$ issues a valid ciphertext under the target identity as a decryption query is bounded due to the properties of the underlying building blocks.

This "previous strategy" seems to work in our proof. But it is not sufficient because there seems to be a chance for the adversary $\mathcal{A}$ to confuse the NM-sID-CPA adversary $\mathcal{B}$ by submitting a decryption query of the form $(f(r^*), y)$ where f is a OWF, $f(r^*)$ is submitted as a simulator's target identity, and $y \neq y^*$. Seeing such a query, $\mathcal{B}$ cannot tell whether it is a valid ciphertext or not and only it can do is to return "$\perp$" to $\mathcal{A}$. If this query is a valid ciphertext, then $\mathcal{B}$'s simulation for $\mathcal{A}$ becomes imperfect by the improper response $\perp$ (if this is not the case, then the simulation is still perfect). However, notice that if the ciphertext of the form $(f(r^*), y)$ where $y \neq y^*$ is valid, then $\mathcal{B}$ can use it to gain NM-sID-CPA advantage by outputting y with a relation such that "the $|r^*|$-significant bits are mapped to the same value by f." Namely, suppose y is an encryption of $(m_A||r_A)$ under the target identity "$f(r^*)$", then a valid ciphertext satisfies $f(r_A) = f(r^*)$, which can be used for the relation R. (We call this "new strategy".)

The difficult point is that the NM-sID-CPA adversary $\mathcal{B}$ cannot know whether $\mathcal{A}$'s decryption query under the target identity is a valid ciphertext or not when $\mathcal{A}$ issues such a query. Therefore, we further show how to handle both the "previous" and "new" strategies so that the NM-sID-CPA adversary $\mathcal{B}$ can *always* gain the advantage of breaking NM-sID-CPA security from $\mathcal{A}$'s IND-CCA advantage.

Theorem 2. *If the underlying IBE scheme* Π *is* $(t_{nm}, 1, q, \epsilon_{nm})$*-NM-sID-CPA secure and* f *is a* (t_{ow}, ϵ_{ow})*-OWF, then the proposed PKE scheme* Π' *is* $(t, q, 4q\epsilon_{nm} + 2q\epsilon_{ow})$*-IND-CCA secure, where* $t = \min\{t_{nm}, t_{ow}\} - O(q)$.

Proof. Suppose $\mathcal{A}$ is an adversary that breaks (t_A, q, ϵ_{cca})-IND-CCA security of Π', which means that $\mathcal{A}$ with running time t makes at most q decryption queries and wins the IND-CCA game with probability $\frac{1}{2} + \epsilon_{cca}$. Then we construct another adversary $\mathcal{B}$ who can break $(t_A + O(q),\ 1,\ q,\ \frac{1}{4q}\epsilon_{cca} - \frac{1}{2}\epsilon_{ow})$-NM-sID-CPA security of the underlying IBE scheme Π using $\mathcal{A}$ and the $(t_A + O(q), \epsilon_{ow})$-OWF f. Note that we use the weakest case of NM-sID-CPA security where an attacker outputs a binary relation R and only a single ciphertext y' in Output phase (i.e. $l = 1$ in Defnition 8), because it is sufficient for our proof. Without loss of generality, we assume $q > 0$. The adversary $\mathcal{B}$, simulating the IND-CCA game for $\mathcal{A}$, plays the NM-sID-CPA game with the NM-sID-CPA challenger $\mathcal{C}$ as follows.

Setup. $\mathcal{B}$ generates a public key for $\mathcal{A}$ as follows. Pick a OWF f. Choose $r^* \in \{0,1\}^\gamma$ uniformly at random and compute $\mathsf{ID}^* \leftarrow f(r^*)$. Commit ID^* as $\mathcal{B}$'s target identity in the NM-sID-CPA game and obtain prm from $\mathcal{C}$. Give $PK = (\mathsf{prm}, f)$ to $\mathcal{A}$.

Phase 1. $\mathcal{B}$ responds to $\mathcal{A}$'s decryption queries $\chi = (\mathsf{ID}, y)$ by returning m to $\mathcal{A}$, where m is generated as follows.

If $\mathsf{ID} = \mathsf{ID}^*$**:** Set $m = \perp$.

Otherwise: Issue ID as an extraction query to $\mathcal{C}$ and obtain dk_{ID}. Compute $\mathsf{IDec}(dk_{\mathsf{ID}}, y)$ and set $m = \perp$ if the decryption result is $\perp$. Otherwise, check whether $f(r) = \mathsf{ID}$ holds or not for the decryption result $(m||r)$. If this is the case, then this m is used as a response to $\mathcal{A}$, otherwise, set $m = \perp$.

Challenge. When $\mathcal{A}$ submits (m_0, m_1) to $\mathcal{B}$, $\mathcal{B}$ returns the challenge ciphertext χ^* to $\mathcal{A}$ where χ^* is generated as follows. Flip a coin $\beta \in \{0,1\}$ uniformly at random. Choose

a random message $m' \in \mathcal{M}_{\Pi'}$ (equal length to m_β). Choose $r' \in \{0,1\}^\gamma$ uniformly at random. Set $M_0 = (m_\beta || r^*)$ and $M_1 = (m' || r') \in \mathcal{M}_\Pi$. Define a probabilistic machine $\mathcal{M}^*_\Pi$ that outputs one of $\{M_0, M_1\}$ uniformly at random. Submit the description of $\mathcal{M}^*_\Pi$ to $\mathcal{C}$ as $\mathcal{B}$'s challenge and obtain y^* from $\mathcal{C}$. Give $\chi^* = (\mathsf{ID}^*, y^*)$ to $\mathcal{A}$.

Phase 2. $\mathcal{B}$ responds to $\mathcal{A}$'s decryption queries in the same way as Phase 1.

Guess. $\mathcal{A}$ outputs β_A. $\mathcal{B}$ outputs a ciphertext y' and a description of a relation R which are generated as follows.

If $\beta_A = \beta$: Set a binary relation $R(\cdot,\cdot)$ as "$R(a,b) = \mathtt{true}$ iff γ-$\mathsf{LSB}(a) = \gamma$-$\mathsf{LSB}(b)$." Pick $m'' \in \mathcal{M}_\Pi$ (equal length to m_β) randomly. Choose $r'' \in \{0,1\}^\gamma$ uniformly at random. Flip a biased coin $b_\alpha \in \{0,1\}$ where $b_\alpha = 1$ holds with probability α. If $b_\alpha = 1$, compute $y' \leftarrow \mathsf{IEnc}(\mathsf{prm}, \mathsf{ID}^*, (m''||r^*))$, otherwise compute $y' \leftarrow \mathsf{IEnc}(\mathsf{prm}, \mathsf{ID}^*, (m''||r''))$.

Otherwise: Set a binary relation $R(\cdot,\cdot)$ as "$R(a,b) = \mathtt{true}$ iff $f(\gamma$-$\mathsf{LSB}(a)) = f(\gamma$-$\mathsf{LSB}(b))$." Pick uniformly one ciphertext $\chi_j = (\mathsf{ID}_j, y_j)$ from $\mathcal{A}$'s decryption queries $\{\chi_i = (\mathsf{ID}_i, y_i)\}_{i \in \{1,\ldots,q\}}$ and set $y' = y_j$.

Note that $\mathcal{B}$ makes at most the same number of extraction queries as $\mathcal{A}$' decryption queries, i.e. q times. Note also that $\mathcal{B}$ needs to run IDec at most once for each decryption query from $\mathcal{A}$, as well as other computations of constant steps, which causes additional running time $O(q)$. (In particular, the computation of $\mathcal{M}^*_\Pi$ and the evaluation of R can be done in constant steps.)

We remain probability α unknown here, and discuss later in this proof. Note that $\Pr[b_\alpha = 1] = \alpha$ and $\Pr[b_\alpha = 0] = 1 - \alpha$, according to our definition. Note also that the description of the relation R that $\mathcal{B}$ uses is different depending on whether $\mathcal{A}$'s guess bit β_A is equal to the bit β chosen by $\mathcal{B}$.

Next, we estimate $\mathcal{B}$'s NM-sID-CPA advantage $\mathsf{Adv}^{\text{NM-sID-CPA}}_{\Pi,\mathcal{B}}$. In our construction of $\mathcal{B}$, The plaintext chooser $\mathcal{M}^*_\Pi$ submitted by $\mathcal{B}$ is always a uniform distribution over two messages $\{M_0, M_1\}$. Thus, for convenience, we assume that the NM-sID-CPA challenger $\mathcal{C}$ flips two coins $c^* \in \{0,1\}$ and $c^{\bar{*}} \in \{0,1\}$ uniformly at random and sets M_{c^*} as a challenge message M^* and $M_{c^{\bar{*}}}$ as $M^{\bar{*}}$ in Challenge phase. Note that $\mathcal{B}$'s simulation for $\mathcal{A}$ becomes imperfect if $c^* = 1$ occurs, since with overwhelming probability the challenge ciphertext given to $\mathcal{A}$ is not an encryption of either of (m_0, m_1) submitted by $\mathcal{A}$.

We say that a ciphertext $\chi = (\mathsf{ID}, y)$ is *valid* if χ decrypts to an element in the plaintext space $\mathcal{M}_{\Pi'}$ (i.e., not $\perp$) according to the decryption process of Π'. Let Valid be an event that $\mathcal{A}$ issues at least one decryption query of the form $\chi = (\mathsf{ID}^*, y)$ that is *valid*. Note that $\mathcal{B}$'s simulation for $\mathcal{A}$ becomes imperfect if Valid occurs, because in this case $\mathcal{B}$ cannot return an appropriate plaintext to $\mathcal{A}$.

We also note that throughout the simulation $\mathcal{B}$ cannot explicitly know whether $c^* = 1$ and Valid have occurred or not.

In the following, we consider the following seven cases depending on c^*, $c^{\bar{*}}$, β, β_A, and Valid:

- Case 1: $c^* = c^{\bar{*}}$
- Case 2: $c^* = 0 \wedge c^{\bar{*}} = 1 \wedge \beta_A = \beta \wedge \overline{\mathsf{Valid}}$

- Case 3: $c^* = 0 \wedge \bar{c^*} = 1 \wedge \beta_A = \beta \wedge \mathsf{Valid}$
- Case 4: $c^* = 0 \wedge \bar{c^*} = 1 \wedge \beta_A \neq \beta \wedge \overline{\mathsf{Valid}}$
- Case 5: $c^* = 0 \wedge \bar{c^*} = 1 \wedge \beta_A \neq \beta \wedge \mathsf{Valid}$
- Case 6: $c^* = 1 \wedge \bar{c^*} = 0 \wedge \beta_A = \beta$
- Case 7: $c^* = 1 \wedge \bar{c^*} = 0 \wedge \beta_A \neq \beta$

Note that these cases cover all possibilities. Let $[\textcircled{i}]$ denotes an event that Case i occurs. We denote $\mathcal{B}$'s advantage in Case i by Adv_i and define it as:

$$\mathsf{Adv}_i = \Pr[\mathcal{R}^* \wedge \textcircled{i}] - \Pr[\mathcal{R}^{\bar{*}} \wedge \textcircled{i}] = (\Pr[\mathcal{R}^*|\textcircled{i}] - \Pr[\mathcal{R}^{\bar{*}}|\textcircled{i}]) \cdot \Pr[\textcircled{i}],$$

where $\mathcal{R}^*$ and $\mathcal{R}^{\bar{*}}$ are defined in Section 2.5. According to the definition of the NM-sID-CPA advantage, we obviously have $\mathsf{Adv}^{\text{NM-sID-CPA}}_{\Pi,\mathcal{B}} = \sum_{i=1}^{7} \mathsf{Adv}_i$.

Now, we introduce the following lemmas.[2] In the following, just for notational convenience, we define two conditional probabilities $P_v = \Pr[\mathsf{Valid}|c^* = 0 \wedge \bar{c^*} = 1]$ and $P_k = \Pr[\beta_A = \beta|c^* = 0 \wedge \bar{c^*} = 1 \wedge \mathsf{Valid}]$, and use them for describing the lemmas.

Lemma 2. $\mathsf{Adv}_1 = 0$.

Lemma 3. $\mathsf{Adv}_2 \geq \frac{1}{4}\alpha(\frac{1}{2} + \epsilon_{cca})(1 - P_v) - \frac{1}{2^\gamma}\Pr[②]$.

Lemma 4. $\mathsf{Adv}_3 \geq \frac{1}{4}\alpha P_k P_v - \frac{1}{2^\gamma}\Pr[③]$.

Lemma 5. $\mathsf{Adv}_4 \geq -\epsilon_{ow}\Pr[④]$.

Lemma 6. $\mathsf{Adv}_5 \geq -\frac{1}{4q}(1 - P_k)P_v - \epsilon_{ow}\Pr[⑤]$.

Lemma 7. $\mathsf{Adv}_6 \geq -\frac{1}{8}\alpha - \frac{1}{2^\gamma}\Pr[⑥]$.

Lemma 8. $\mathsf{Adv}_7 \geq -\epsilon_{ow}\Pr[⑦]$.

Then, before proving the lemmas, we first calculate $\mathsf{Adv}^{\text{NM-sID-CPA}}_{\Pi,\mathcal{B}}$.

$$\begin{aligned}
\mathsf{Adv}^{\text{NM-sID-CPA}}_{\Pi,\mathcal{B}} &= \sum_{i=1}^{7} \mathsf{Adv}_i \\
&\geq \frac{1}{4}\alpha(\frac{1}{2} + \epsilon_{cca})(1 - P_v) + \frac{1}{4}\alpha P_k P_v + \frac{1}{4q}(1 - P_k)P_v - \frac{1}{8}\alpha \\
&\quad - \frac{1}{2^\gamma}(\Pr[②] + \Pr[③] + \Pr[⑥]) - \epsilon_{ow}(\Pr[④] + \Pr[⑤] + \Pr[⑦]) \\
&\geq \frac{1}{4}\alpha(\frac{1}{2} + \epsilon_{cca})(1 - P_v) + \frac{1}{4}\alpha P_k P_v + \frac{1}{4q}(1 - P_k)P_v - \frac{1}{8}\alpha - \frac{1}{2}\epsilon_{ow},
\end{aligned}$$

where, in order to sum up the terms regarding $\Pr[\textcircled{i}]$ into one term $\frac{1}{2}\epsilon_{ow}$ in the last inequality, we used the fact that $\sum_{i=2}^{7}\Pr[\textcircled{i}] = 1 - \Pr[①] = 1 - \Pr[c^* = \bar{c^*}] = \frac{1}{2}$ and the following claim.

Claim 1. $\frac{1}{2^\gamma} \leq \epsilon_{ow}$.

[2] Here, we purposely remain each $\Pr[\textcircled{i}]$ as it is for the later calculation of $\mathsf{Adv}^{\text{NM-sID-CPA}}_{\Pi,\mathcal{B}}$.

Proof of Claim 1. Consider the adversary $\mathcal{A}'$ against one-wayness of f who on input $f(r)$, where r is chosen uniformly from $\{0,1\}^\gamma$, runs as follows. Choose r' uniformly at random from $\{0,1\}^\gamma$ and output r' as the solution of the one-way experiment. Since r and r' are independent, we have

$$\mathsf{Adv}^{\mathrm{OW}}_{f,\mathcal{A}'} = \Pr[f(r) = f(r')] \geq \Pr[r = r'] = \frac{1}{2^\gamma}.$$

According to the definition of ϵ_{ow}, we have $\mathsf{Adv}^{\mathrm{OW}}_{f,\mathcal{A}} \leq \epsilon_{ow}$ for any adversary $\mathcal{A}$ (including $\mathcal{A}'$). Thus, we have $\frac{1}{2^\gamma} \leq \epsilon_{ow}$. □

Now, focusing on the second and the third terms of the right side member of the above inequality, we can define $\alpha = \frac{1}{q}$, which will cancel out all the terms regarding P_k. Using this α, we have

$$\mathsf{Adv}^{\text{NM-sID-CPA}}_{\Pi,\mathcal{B}} \geq \frac{1}{4q}(\epsilon_{cca} + P_v(\frac{1}{2} - \epsilon_{cca})) - \frac{1}{2}\epsilon_{ow} \geq \frac{1}{4q}\epsilon_{cca} - \frac{1}{2}\epsilon_{ow},$$

where the right side inequality is obtained due to $0 \leq \epsilon_{cca} \leq \frac{1}{2}$ (see Definition 1) and the fact that $P_v \geq 0$.

Consequently, assuming $\mathcal{A}$ has advantage ϵ_{cca} in breaking IND-CCA security of the proposed PKE scheme Π' and f is a $(t_A + O(q), \epsilon_{ow})$-OWF, $\mathcal{B}$ can break NM-sID-CPA security of the underlying IBE scheme Π with the above advantage, using above α.

To complete the proof of Theorem 2, we prove Lemmas 2 to 8 in order.

Proof of Lemma 2: In Case 1, $c^* = \bar{c^*}$ occurs. This means $M^* = \bar{M^*}$, and thus, the events $\mathcal{R}^*$ and $\bar{\mathcal{R}^*}$ become identical. Therefore, $\mathcal{B}$'s advantage in Case 1 is

$$\mathsf{Adv}_1 = (\Pr[\mathcal{R}^*|①] - \Pr[\bar{\mathcal{R}^*}|①]) \cdot \Pr[①] = 0,$$

which completes the proof of Lemma 2. □

Proof of Lemma 3: In Case 2, an event $[②] = [c^* = 0 \wedge \bar{c^*} = 1 \wedge \beta_A = \beta \wedge \overline{\mathsf{Valid}}]$ occurs. In this case, we have $M^* = (m_\beta || r^*)$ and $\bar{M^*} = (m' || r')$, and since $\beta_A = \beta$, the relation R output by $\mathcal{B}$ tests the equality of the γ-least significant bits.

First, we estimate $\Pr[\mathcal{R}^*|②]$.

$$\Pr[\mathcal{R}^*|②] \geq \Pr[\mathcal{R}^* \wedge b_\alpha = 1|②] = \Pr[b_\alpha = 1] \cdot \Pr[\mathcal{R}^*|② \wedge b_\alpha = 1] = \alpha,$$

where we used the following.

Claim 2. $\Pr[\mathcal{R}^*|② \wedge b_\alpha = 1] = 1$.

Proof of Claim 2. $c^* = 0$ implies that the plaintext of $\mathcal{B}$'s challenge ciphertext y^* is $M^* = (m_\beta || r^*)$, and thus γ-least significant bits of M^* is r^*. And when $[\beta_A = \beta \wedge b_\alpha = 1]$ occurs, $\mathcal{B}$ outputs y' which is an encryption of $(m'' || r^*)$. Thus, conditioned on $[② \wedge b_\alpha = 1]$, $\mathcal{B}$ always outputs a ciphertext y' such that $\mathcal{R}^*$ occurs and we have $\Pr[\mathcal{R}^*|② \wedge b_\alpha = 1] = 1$. □

Next, we estimate $\Pr[\bar{\mathcal{R}^*}|②]$.

Claim 3. $\Pr[\bar{\mathcal{R}^*}|②] = \frac{1}{2^\gamma}$.

Proof of Claim 3. Recall that $\gamma\text{-}\mathsf{LSB}(M^{\bar{*}}) = r'$. Recall also that γ-least significant bits of the plaintext of y' is either r^* or r'', depending on the value b_α. Since each of r', r^* and r'' is uniformly and independently chosen by $\mathcal{B}$, we have

$$\Pr[\mathcal{R}^{\bar{*}}|②] = \Pr[r' = r^* \wedge b_\alpha = 1] + \Pr[r' = r'' \wedge b_\alpha = 0] = \frac{1}{2^\gamma} \cdot \alpha + \frac{1}{2^\gamma} \cdot (1 - \alpha) = \frac{1}{2^\gamma}.$$

□

Finally, we estimate $\Pr[②]$. We have

$$\begin{aligned}\Pr[②] &= \Pr[c^* = 0 \wedge c^{\bar{*}} = 1 \wedge \beta_A = \beta \wedge \overline{\mathsf{Valid}}] \\ &= \frac{1}{4}\Pr[\beta_A = \beta | c^* = 0 \wedge c^{\bar{*}} = 1 \wedge \overline{\mathsf{Valid}}] \cdot \Pr[\overline{\mathsf{Valid}} | c^* = 0 \wedge c^{\bar{*}} = 1] \\ &= \frac{1}{4}(\frac{1}{2} + \epsilon_{cca})(1 - P_v),\end{aligned}$$

where we used $\Pr[c^* = 0 \wedge c^{\bar{*}} = 1] = \frac{1}{4}$ and the following.

Claim 4. $\Pr[\beta_A = \beta | c^* = 0 \wedge c^{\bar{*}} = 1 \wedge \overline{\mathsf{Valid}}] = \frac{1}{2} + \epsilon_{cca}$.

Proof of Claim 4. Note that when Valid does not occur, then $\mathcal{B}$'s response to each of $\mathcal{A}$'s decryption queries is perfect. Concretely, if the decryption query is of the form $\chi = (\mathsf{ID}, y)$ with $\mathsf{ID} \neq \mathsf{ID}^*$, then $\mathcal{B}$ can correctly decrypt χ because it can obtain the decryption keys dk_{ID} by the use of extraction queries to $\mathcal{B}$'s challenger. And if the decryption query is of the form $\chi = (\mathsf{ID}^*, y)$, then this χ is not a valid ciphertext unless Valid occurs, and thus $\mathcal{B}$'s response $\perp$ for this type of query is also a correct answer. Moreover, note also that if $c^* = 0$ occurs, then the challenge ciphertext given from $\mathcal{B}$ to $\mathcal{A}$ is a correct encryption of m_β. Therefore, $\mathcal{B}$ perfectly simulates the IND-CCA game for $\mathcal{A}$, and the view of $\mathcal{A}$ is identical to that when it is attacking the proposed scheme Π' where the challenge bit for $\mathcal{A}$ is β. Then, the event $\beta_A = \beta$ corresponds to the event that $\mathcal{A}$ succeeds in guessing in the IND-CCA game, which occurs with probability $\frac{1}{2} + \epsilon_{cca}$. □

Consequently, $\mathcal{B}$'s advantage in Case 2 is estimated as:

$$\mathsf{Adv}_2 = (\Pr[\mathcal{R}^*|②] - \Pr[\mathcal{R}^{\bar{*}}|②]) \cdot \Pr[②] \geq \frac{1}{4}\alpha(\frac{1}{2} + \epsilon_{cca})(1 - P_v) - \frac{1}{2^\gamma}\Pr[②],$$

which completes the proof of Lemma 3. □

Proof of Lemma 4: In Case 3, an event $[③] = [c^* = 0 \wedge c^{\bar{*}} = 1 \wedge \beta_A = \beta \wedge \mathsf{Valid}]$ occurs. In this case, we have $M^* = (m_\beta || r^*)$ and $M^{\bar{*}} = (m' || r')$, and since $\beta_A = \beta$, the relation R output by $\mathcal{B}$ tests the equality of the γ-least significant bits. With the same discussion in the proof of Lemma 3, we have $\Pr[\mathcal{R}^*|③] \geq \alpha$ and $\Pr[\mathcal{R}^{\bar{*}}|②] = \frac{1}{2^\gamma}$. As for $\Pr[③]$, we have

$$\begin{aligned}\Pr[③] &= \Pr[c^* = 0 \wedge c^{\bar{*}} = 1 \wedge \beta_A = \beta \wedge \mathsf{Valid}] \\ &= \frac{1}{4}\Pr[\beta_A = \beta | c^* = 0 \wedge c^{\bar{*}} = 1 \wedge \mathsf{Valid}] \cdot \Pr[\mathsf{Valid} | c^* = 0 \wedge c^{\bar{*}} = 1] \\ &= \frac{1}{4}P_k P_v.\end{aligned}$$

Consequently, $\mathcal{B}$'s advantage in Case 3 is estimated as:

$$\mathsf{Adv}_3 = (\Pr[\mathcal{R}^*|③] - \Pr[\mathcal{R}^{\bar{*}}|③]) \cdot \Pr[③] \geq \frac{1}{4}\alpha P_k P_v - \frac{1}{2^\gamma}\Pr[③],$$

which completes the proof of Lemma 4. □

Proof of Lemma 5: In Case 4, an event $[④] = [c^* = 0 \wedge c^{\bar{*}} = 1 \wedge \beta_A \neq \beta \wedge \overline{\mathsf{Valid}}]$ occurs. In this case, we have $M^* = (m_\beta || r^*)$ and $M^{\bar{*}} = (m' || r')$, and since $\beta_A \neq \beta$, the relation R output by $\mathcal{B}$ tests the equality of "f of" the γ-least significant bits. According to the description of $\mathcal{B}$, when $\beta_A \neq \beta$ occurs, the ciphertext y' which is finally output by $\mathcal{B}$ is chosen uniformly from $\mathcal{A}$'s decryption queries.

We estimate $\Pr[\mathcal{R}^{\bar{*}}|④]$ in the following.

Claim 5. $\Pr[\mathcal{R}^{\bar{*}}|④] \leq \epsilon_{ow}$.

Proof of Claim 5. Since in this case r' is information-theoretically hidden from $\mathcal{A}$, the value $f(r')$ is also information-theoretically hidden from $\mathcal{A}$. Thus, the probability that $\mathcal{A}$, without seeing $f(r')$, happens to issue some decryption query such that the image with f of γ-least significant bits of the plaintext becomes identical to $f(r')$ is at most ϵ_{ow}, because f is assumed to be a $(t_A + O(q), \epsilon_{ow})$-OWF. □

Consequently, $\mathcal{B}$'s advantage in Case 4 is estimated as:

$$\mathsf{Adv}_4 \geq -\Pr[\mathcal{R}^{\bar{*}}|④] \cdot \Pr[④] \geq -\epsilon_{ow} \Pr[④],$$

which completes the proof of Lemma 5. □

Proof of Lemma 6: In Case 5, an event $[⑤] = [c^* = 0 \wedge c^{\bar{*}} = 1 \wedge \beta_A \neq \beta \wedge \mathsf{Valid}]$ occurs. In this case, we have $M^* = (m_\beta || r^*)$ and $M^{\bar{*}} = (m' || r')$. Since $\beta_A \neq \beta$, the relation R output by $\mathcal{B}$ tests the equality of "f of" the γ-least significant bits, and the ciphertext y' output by $\mathcal{B}$ is chosen uniformly from $\mathcal{A}$'s decryption queries.

First, we estimate $\Pr[\mathcal{R}^*|⑤]$.

Claim 6. $\Pr[\mathcal{R}^*|⑤] \geq \frac{1}{q}$.

Proof of Claim 6. Since Valid occurs, in this case $\mathcal{A}$ issues at least one *valid* ciphertext of the form $\chi = (\mathsf{ID}^*, y)$. But $\mathcal{B}$ returns $\perp$ as an answer to this query, which is not a correct response. Therefore $\mathcal{B}$'s simulation for $\mathcal{A}$ becomes imperfect from the point $\mathcal{A}$ receives the response. However, a valid ciphertext $\chi = (\mathsf{ID}^*, y)$ satisfies $\mathsf{IDec}(dk_{\mathsf{ID}^*}, y) = (m_A || r_A) \neq \perp$ and $f(r_A) = \mathsf{ID}^* = f(r^*)$, where dk_{ID^*} is a decryption key corresponding to ID^*. Since $\mathcal{B}$ picks one ciphertext y' from $\mathcal{A}$'s decryption queries uniformly and outputs it, if $\mathcal{B}$'s choice is a ciphertext that causes the event Valid, $\mathcal{R}^*$ occurs. Since the number of $\mathcal{A}$'s query is at most $q > 0$, the probability that $\mathcal{B}$ picks a valid ciphertext that causes the event Valid is at least $\frac{1}{q}$. □

With the same discussion in the proof of Lemma 5, we have $\Pr[\mathcal{R}^{\bar{*}}|⑤] \leq \epsilon_{ow}$. With a similar calculation to $\Pr[③]$, we also have

$$\Pr[⑤] = \Pr[c^* = 0 \wedge c^{\bar{*}} = 1 \wedge \beta_A \neq \beta \wedge \mathsf{Valid}] = \frac{1}{4}(1 - P_k)P_v.$$

Consequently, $\mathcal{B}$'s advantage in Case 5 is estimated as:

$$\mathsf{Adv}_5 = (\Pr[\mathcal{R}^*|⑤] - \Pr[\mathcal{R}^{\bar{*}}|⑤]) \cdot \Pr[⑤] \geq \frac{1}{4q}(1 - P_k)P_v - \epsilon_{ow} \Pr[⑤],$$

which completes the proof of Lemma 6. □

Proof of Lemma 7: In Case 6, an event $[⑥] = [c^* = 1 \wedge c^{\bar{*}} = 0 \wedge \beta_A = \beta]$ occurs. In this case, we have $M^* = (m'||r')$ and $M^{\bar{*}} = (m_\beta||r^*)$, and since $\beta_A = \beta$, the relation R output by $\mathcal{B}$ tests the equality of the γ-least significant bits.

We first estimate $\Pr[\mathcal{R}^{\bar{*}}|⑥]$.

$$\begin{aligned}\Pr[\mathcal{R}^{\bar{*}}|⑥] &= \Pr[\mathcal{R}^{\bar{*}} \wedge b_\alpha = 1|⑥] + \Pr[\mathcal{R}^{\bar{*}} \wedge b_\alpha = 0|⑥] \\ &= \Pr[b_\alpha = 1] \cdot \Pr[\mathcal{R}^{\bar{*}}|⑥ \wedge b_\alpha = 1] + \Pr[b_\alpha = 0] \cdot \Pr[\mathcal{R}^{\bar{*}}|⑥ \wedge b_\alpha = 0] \\ &\leq \Pr[b_\alpha = 1] + \Pr[\mathcal{R}^{\bar{*}}|⑥ \wedge b_\alpha = 0] = \alpha + \frac{1}{2^\gamma},\end{aligned}$$

where we used the following.

Claim 7. $\Pr[\mathcal{R}^{\bar{*}}|⑥ \wedge b_\alpha = 0] = \frac{1}{2^\gamma}$.

Proof of Claim 7. When $[b_\alpha = 0]$ occurs, the ciphertext y' output by $\mathcal{B}$ is an encryption of $(m''||r'')$ where r'' is chosen uniformly from $\{0,1\}^\gamma$, independently of r^*. Therefore, the probability that $\mathcal{R}^{\bar{*}}$ occurs in this case is identical to the probability that $r'' = r^*$ occurs, which is exactly $\frac{1}{2^\gamma}$. □

As for $\Pr[⑥]$, we have

$$\Pr[⑥] = \Pr[c^* = 1 \wedge c^{\bar{*}} = 0 \wedge \beta_A = \beta] = \frac{1}{4}\Pr[\beta_A = \beta|c^* = 1 \wedge c^{\bar{*}} = 0] = \frac{1}{8},$$

where we used the following.

Claim 8. $\Pr[\beta_A = \beta|c^* = 1 \wedge c^{\bar{*}} = 0] = \frac{1}{2}$.

Proof of Claim 8. If $c^* = 1$, then the challenge ciphertext given to $\mathcal{A}$ is of the form $\chi^* = (\mathsf{ID}^*, y^*) = (f(r^*), \mathsf{IEnc}(\mathsf{prm}, \mathsf{ID}^*, (m'||r'))$. This χ^* is, with overwhelming probability, not a legitimate challenge ciphertext for $\mathcal{A}$ and thus $\mathcal{B}$'s simulation for $\mathcal{A}$ becomes imperfect very likely. However, since in this case β is information-theoretically hidden from $\mathcal{A}$'s view ($\mathcal{A}$ cannot see $M^{\bar{*}} = (m_\beta||r^*)$) and $\beta \in \{0,1\}$ is chosen uniformly, the probability that $\beta_A = \beta$ occurs is exactly $\frac{1}{2}$. □

Consequently, $\mathcal{B}$'s advantage in Case 6 is estimated as:

$$\mathsf{Adv}_6 \geq -\Pr[\mathcal{R}^{\bar{*}}|⑥] \cdot \Pr[⑥] \geq -\frac{1}{8}\alpha - \frac{1}{2^\gamma}\Pr[⑧],$$

which completes the proof of Lemma 7. □

Proof of Lemma 8: In Case 7, an event $[⑦] = [c^* = 1 \wedge c^{\bar{*}} = 0 \wedge \beta_A \neq \beta]$ occurs. In this case, we have $M^* = (m'||r')$ and $M^{\bar{*}} = (m_\beta||r^*)$. Since $\beta_A \neq \beta$, the relation R output by $\mathcal{B}$ tests the equality of "f of" the γ-least significant bits, and the ciphertext y' output by $\mathcal{B}$ is chosen uniformly from $\mathcal{A}$'s decryption queries.

We estimate $\Pr[\mathcal{R}^{\bar{*}}|⑥]$.

Claim 9. $\Pr[\mathcal{R}^{\bar{*}}|⑥] \leq \epsilon_{ow}$.

Proof of Claim 9. Recall that, when $[\beta_A \neq \beta]$ occurs, $\mathcal{B}$ outputs the relation R such that $R(a,b)$ tests whether $f(\gamma\text{-}\mathsf{LSB}(a)) = f(\gamma\text{-}\mathsf{LSB}(b))$ holds. Thus, the event $\mathcal{R}^{\bar{*}}$ in this case consists of the following two events: (1) $\mathcal{A}$ issues at least one decryption query (ID^*, y) which satisfies the conditions $\mathsf{IDec}(dk_{\mathsf{ID}^*}, y) = (m||r) \neq \perp$ and $\mathsf{ID}^* = f(r^*) = f(r)$, and (2) $\mathcal{B}$ chooses such a query. Note that the first event above is exactly the same event as Valid. For

notational convenience, we denote by Choice the second event above. Moreover, according to our construction of $\mathcal{B}$, when $c^* = 1$ occurs, the challenge ciphertext given to $\mathcal{A}$ is a "garbage" ciphertext which is of the form $\chi^* = (\mathsf{ID}^*, y^*) = (f(r^*), \mathsf{IEnc}(\mathsf{prm}, f(r^*), (m'||r')))$ where r^* and r' are chosen independently and uniformly from $\{0,1\}^\gamma$ and m' is also chosen randomly. Also for notational convenience, we denote by $\mathsf{Garbage}$ an event that $\mathcal{A}$ is given a garbage challenge ciphertext of the above form. Suppose that, as $\mathcal{B}$'s final output, one ciphertext (ID, y) is chosen from $\mathcal{A}$'s decryption queries. Using these notations, we have

$$\begin{aligned}
\Pr[\mathcal{R}^{\bar{*}}|\textcircled{7}] &= \Pr[\mathsf{Choice} \wedge \mathsf{IDec}(dk_{\mathsf{ID}^*}, y) = (m||r) \neq \perp \wedge R((m_\beta||r^*), (m||r)) \mid \\
&\qquad\qquad c^* = 1 \wedge c^{\bar{*}} = 0 \wedge \beta_A \neq \beta] \\
&= \Pr[\mathsf{Choice} \wedge \mathsf{IDec}(dk_{\mathsf{ID}^*}, y) = (m||r) \neq \perp \wedge f(r^*) = f(r) | \mathsf{Garbage}] \\
&= \Pr[\mathsf{Choice} \wedge \mathsf{Valid} | \mathsf{Garbage}] \\
&\leq \Pr[\mathsf{Valid}|\mathsf{Garbage}].
\end{aligned}$$

Thus, all we have to do is to show $\Pr[\mathsf{Valid}|\mathsf{Garbage}] \leq \epsilon_{ow}$.

Towards a contradiction, we assume $\Pr[\mathsf{Valid}|\mathsf{Garbage}] > \epsilon_{ow}$. We construct another adversary $\mathcal{B}'$ which, using $\mathcal{A}$, breaks one-wayness of f with the OWF advantage greater than ϵ_{ow} and runs in time $t_A + O(q)$. The description of $\mathcal{B}'$ is as follows.

Given $f(r^*)$ (where r^* is uniformly chosen from $\{0,1\}^\gamma$ and unknown to $\mathcal{B}'$), $\mathcal{B}'$ first sets $\mathsf{ID}^* = f(r^*)$, then runs $(\mathsf{prm}, \mathsf{msk}) \leftarrow \mathsf{ISetup}$ and $dk_{\mathsf{ID}^*} \leftarrow \mathsf{IExt}(\mathsf{prm}, \mathsf{msk}, \mathsf{ID}^*)$. It gives $PK = (\mathsf{prm}, f)$ to $\mathcal{A}$. Since $\mathcal{B}'$ possesses $SK = \mathsf{msk}$, it can perfectly respond to the decryption queries. When $\mathcal{A}$ submits two plaintexts as a challenge, $\mathcal{B}'$ ignores them and generates a "garbage" challenge ciphertext $\chi^* = (\mathsf{ID}^*, y^*) = (f(r^*), \mathsf{IEnc}(\mathsf{prm}, f(r^*), (m'||r')))$ where r' is uniformly chosen from $\{0,1\}^\gamma$ and m' is also chosen randomly. Then $\mathcal{B}'$ gives χ^* to $\mathcal{A}$. After $\mathcal{A}$ outputs a guess bit and terminates, from $\mathcal{A}$'s decryption queries $\{\chi_i\}_{i\in\{1,\ldots,q\}}$ $\mathcal{B}'$ finds a ciphertext (ID, y) whose second component y satisfies $\mathsf{IDec}(dk_{\mathsf{ID}^*}, y) = (m||r) \neq \perp$ and $f(r^*) = f(r)$, and outputs such r (if no such query is found then $\mathcal{B}'$ simply gives up and aborts).

Note that $\mathcal{B}'$ needs to run IExt and IDec for each of decryption query from $\mathcal{A}$, needs to run IDec at most q times for finding a valid ciphertext after $\mathcal{A}$ terminates, and needs to run other computations of constant steps, which canses addtional running time $O(q)$. It is easy to see that $\mathcal{B}'$ perfectly simulates the scenario $\mathsf{Garbage}$ for $\mathcal{A}$. Moreover, whenever Valid occurs, $\mathcal{B}'$ can find a preimage of $f(r^*)$ and thus breaks the one-wayness of f. Therefore, we have

$$\mathsf{Adv}^{\mathrm{OW}}_{f,\mathcal{B}'} = \Pr[\mathsf{Valid}|\mathsf{Garbage}] > \epsilon_{ow},$$

which contradicts that f is a $(t_A + O(q), \epsilon_{ow})$-OWF, and thus we must have $\Pr[\mathsf{Valid}|\mathsf{Garbage}] \leq \epsilon_{ow}$. This completes the proof of Claim 9. □

Consequently, $\mathcal{B}$'s advantage in Case 7 is estimated as:

$$\mathsf{Adv}_7 \geq -\Pr[\mathcal{R}^{\bar{*}}|\textcircled{7}] \cdot \Pr[\textcircled{7}] \geq -\epsilon_{ow} \Pr[\textcircled{7}],$$

which completes the proof of Lemma 8. □

Above completes the proof of Theorem 2. □

Extensions As is the same with the previous generic IBE-to-PKE transformations [34, 26], our transformation can be applied to TBE schemes if we appropriately define non-malleability for TBE schemes. We discuss this in Section 3.3.3.

Moreover, if we consider non-malleability for HIBE schemes in the same way as in Section 2.5, then our transformation can be used to obtain adaptive (resp., selective) identity CCA-secure t-level HIBE from $(t+1)$-level HIBE that is non-malleable against adaptive (resp., selective) identity, chosen plaintext attacks.

A Generic Construction of NM-sID-CPA Secure IBE Schemes So far, no concrete construction of NM-sID-CPA secure IBE scheme (other than CCA secure ones) is known. In theory, however, it is possible to construct an NM-sID-CPA secure IBE scheme from any IND-sID-CPA secure IBE scheme by adopting the methodology by Choi et al. [38]. They showed a construction of a NM-CPA secure PKE scheme from any IND-CPA secure PKE scheme. It is straightforward to see that the IBE analogue construction of [38] trivially works. We note that if we use an adaptive-ID secure (i.e. IND-ID-CPA secure) IBE scheme as a building block scheme of the IBE analogue of the Choi et al. construction [38], then an adaptive-ID, non-malleable (i.e. NM-ID-CPA secure) IBE scheme can be obtained. Moreover, non-malleability for IBE schemes achieved by their construction is a stronger than the one treated in this chapter, in the sense that it captures an adversary who may output invalid ciphertexts.

However, the IBE scheme obtained via their methodology will not be a practical scheme. (To encrypt with the resulting IBE scheme, we have to run the encryption algorithm of the building block IBE scheme $O(\kappa^2)$ times, where κ is the security parameter.)

3.3.3 Applying the Transformation to Tag-Based Encryption

In this subsection, we refer to the paradigm to obtain IND-CCA secure PKE schemes from TBE schemes [78, 79]. A TBE scheme is a PKE scheme whose encryption and decryption algorithms take an arbitrary string called "tag" as an addiotional input (see Section 2.6 for definition). It is well-known that every IBE scheme can be viewed as a TBE scheme if an extraction algorithm of the IBE scheme is combined with a decryption algorithm and identities in the IBE scheme are used as tags of the TBE scheme.

Kiltz [69] showed that the CHK transformation [34] can be applied to TBE schemes. More specifically, if the underlying TBE scheme is indistinguishable against selective tag, *weak* chosen ciphertext attacks (IND-stag-wCCA) [69], then the resulting PKE transformed by the CHK transformation is IND-CCA secure. This paradigm also applies to our IBE-to-PKE transformation if non-malleability for TBE schemes is appropriately defined.

NM-stag-wCCA security and Other Security Notions for TBE We define NM-stag-wCCA security, which is required for TBE schemes when applying our transformation, as in Section 2.6. In this subsection, we refer to the relation between our non-malleability for TBE and other security notions for TBE.

When we consider the security notions of TBE schemes, we should care about when the target tag is chosen by adversaries (i.e., selective tag attacks or adaptive tag attacks), how the decryption queries are allowed to adversaries, and what the goals of adversaries are.

Shoup [100] first implicitly introduced indistinguishability against adaptive tag, CCA

adversaries for TBE.[3] In his definition, the target tag tag^* is chosen in Challenge phase (adaptive tag attack), and adversaries are allowed to issue decryption queries of the form (tag, χ) as long as $(\mathsf{tag}, \chi) \neq (\mathsf{tag}^*, \chi^*)$, where χ^* is the challenge ciphertext. (In particular, a query of the form (tag^*, χ) with $\chi \neq \chi^*$ and a query of the form (tag, χ^*) with $\mathsf{tag} \neq \mathsf{tag}^*$ are allowed.) It was discussed by Kiltz [69] that a TBE scheme with this security notion is equivalent to an IND-CCA secure PKE scheme, if we see a PKE scheme as an instant TBE scheme (e.g., encrypting a tag concatenated with a plaintext) and vice versa (e.g., including one fixed tag in a public key and using it always).

MacKenzie, Reiter, and Yang [79] defined indistinguishability and non-malleability against adaptive tag, CPA and CCA adversaries. Especially, for CCA adversaries, the decryption queries are somewhat restricted compared to Shoup's definition [100], where adversaries are allowed to issue decryption queries (tag, χ) such that $\mathsf{tag} \neq \mathsf{tag}^*$. We refer to this restricted type of CCA attacks as "*weak CCA*", in order not to mix up with Shoup's definition. This name *weak CCA* is due to Kiltz [69]. As for non-malleability defined by MacKenzie et al. [79] (which we call *MRY non-malleability*), the goal of adversaries and the definition of the advantage of adversaries are slightly different from ours (and [7]).[4] In [79], in Output phase, an adversary outputs a description of a relation R and a pair of l-length vectors $(\overrightarrow{\mathsf{tag}}', \overrightarrow{\chi}')$ where $\overrightarrow{\mathsf{tag}}' = (\mathsf{tag}'_1, \ldots, \mathsf{tag}'_l)$ and $\overrightarrow{\chi}' = (\chi'_1, \ldots, \chi'_l)$, and each χ'_i is regarded as an encryption of some plaintext m'_i under tag'_i (may not be tag^*). Then the event $\mathcal{R}'^*$ is defined such that $[\mathsf{tag}^* \notin \overrightarrow{\mathsf{tag}}' \wedge R(m^*, \overrightarrow{m}')]$ occurs. $\mathcal{R}'^{\bar{*}}$ is similarly defined by replacing m^* with $m^{\bar{*}}$ in $\mathcal{R}'^*$. The advantage is defined as the difference between $\Pr[\mathcal{R}'^*]$ and $\Pr[\mathcal{R}'^{\bar{*}}]$. Informally, MRY non-malleability captures the property that given a ciphertext of the target tag tag^*, it is difficult to make meaningfully related ciphertexts encrypted under another tags $\mathsf{tag}' \neq \mathsf{tag}^*$. But this definition does not rule out the possibility of making meaningfully related ciphertexts encrypted under tag^*. It was shown in [79] that this non-malleability and indistinguishability are equivalent under the adaptive tag, weak CCA.

Kiltz [69] defined indistinguishability against selective tag, CCA adversaries, which is the selective tag version of MacKenzie et al.'s indistinguishability definition, where adversaries must choose the target tag tag^* before a public key is given. It is easy to see that an IND-sID-CPA secure IBE scheme can immediately be seen as a TBE scheme satisfying this Kiltz' definition by combining an extraction and a decryption algorithms of IBE schemes appropriately.

We can consider the selective tag version of MRY non-malleability (we call it *selective* MRY non-malleability) and prove that selective MRY non-malleability is equivalent to Kiltz' indistinguishability under weak CCA in the same way as in [79]. However, our definition of non-malleability for TBE schemes (NM-stag-CCA security) in Section 2.6 is stronger than selective MRY non-malleability. Informally, our non-malleability for TBE schemes provides NM-CPA security for the target tag tag^*, and IND-CCA security for tags other than the target tag. Selective MRY non-malleability only ensures IND-CPA security for the target tag.

Galindo [50] showed for IBE schemes that the selective identity security is strictly weaker than adaptive identity security. This relation applies to TBE schemes with a similar discussion in [50]. Namely, selective tag security is strictly weaker than adaptive tag security.

In summary, if restricted to security notions against selective tag, CCA adversaries, Kiltz'

[3] The author refer to tags as "*labels*".

[4] Here, we only consider the *comparison-based* non-malleability [7, 13, 14].

$\mathsf{PKG}(1^\kappa)$:	$\mathsf{PEnc}(PK, m)$:
$(pk, sk) \leftarrow \mathsf{TKG}(1^\kappa)$	$r \leftarrow \{0,1\}^\gamma$; $\mathsf{tag} \leftarrow f(r)$
Pick a OWF f.	$y \leftarrow \mathsf{TEnc}(pk, \mathsf{tag}, (m\|\|r))$
$PK \leftarrow (pk, f)$; $SK \leftarrow sk$	$\chi \leftarrow (\mathsf{tag}, y)$
Output (PK, SK).	Output χ.
$\mathsf{PDec}(SK, \chi)$:	
Parse χ as (tag, y)	
$(m\|\|r)$ / $\perp \leftarrow \mathsf{TDec}(sk, \mathsf{tag}, y)$ (if $\perp$ then output $\perp$ and stop.)	
Output m if $f(r) = \mathsf{tag}$. Otherwise output $\perp$.	

Figure 3.3: The Transformation from TBE to PKE

indistinguishability [69] and selective MRY non-malleability [79] are equivalently the weakest. Our definition of NM-stag-wCCA security is stronger than them, and yet weaker than the selective tag version of Shoup's definition of indistinguishability under CCA [100].

Construction Let $\Pi = (\mathsf{TKG}, \mathsf{TEnc}, \mathsf{TDec})$ be a NM-stag-wCCA secure TBE scheme and $f : \{0,1\}^\gamma \to \mathcal{T}_\Pi$ be a OWF, where $\mathcal{T}_\Pi$ is the tag space of Π. Then we construct a PKE scheme $\Pi' = (\mathsf{PKG}, \mathsf{PEnc}, \mathsf{PDec})$ as in Fig. 3.3. Suppose the plaintext space of Π' is $\mathcal{M}_{\Pi'}$, then we require that the plaintext space $\mathcal{M}_\Pi$ of the underlying TBE scheme Π satisfy $\mathcal{M}_{\Pi'} \times \{0,1\}^\gamma \subseteq \mathcal{M}_\Pi$. We also require that length of all elements of $\mathcal{T}_\Pi$, the output space of f as well as the tag space of Π, be of equal length and fixed. Typically, length γ of the randomness will be the security parameter κ.

Security The security of the PKE scheme obtained via the "TBE-to-PKE" transformation in Fig. 3.3 is guaranteed by the following theorem.

Theorem 3. *If the underlying TBE scheme* Π *is* $(t_{nm}, 1, q, \epsilon_{nm})$*-NM-stag-wCCA secure and* f *is a* (t_{ow}, ϵ_{ow})*-OWF, then the PKE scheme* Π' *in Fig. 3.3 is* $(t, q, 4q\epsilon_{nm} + 2q\epsilon_{ow})$*-IND-CCA secure, where* $t = \min\{t_{nm}, t_{ow}\} - O(q)$.

We omit the proof of this theorem, because it is almost the same as the proof of Theorem 2.

Note that our transformation cannot be applied if the underlying TBE scheme satisfies only selective MRY non-malleability [79]. The non-malleability adversary (that uses the IND-CCA adversary internally) essentially uses the ability of being the NM-stag-wCCA adversary in the definition in Section 2.6: the final output can be (a relation and) a ciphertext which is encrypted under the target tag. If selective MRY non-malleability is used, the non-malleability adversary with our strategy cannot appropriately provide a ciphertext encrypted under tags that are different from the target tag.

3.4 Proposed Encapsulation Scheme

As we have seen in Section 3.2, designing an encapsulation scheme with small parameter size is important for the size-efficiency of the BK transformation. In this section, we present an efficient encapsulation scheme using a PRG with NCR property and prove its security. We also show a concrete instantiation of the PRG with NCR property.

$\mathsf{ESetup}(1^k)$:	$\mathsf{ECom}(\mathsf{prm})$:
Pick a (NCR-k-LSB) PRG G.	$d \leftarrow \{0,1\}^\kappa$
$\mathsf{prm} \leftarrow G$	$(r\|\|c) \leftarrow G(d)$ (s.t. $\|r\| = \|c\| = k$)
Output prm	Output (r, c, d).
$\mathsf{ERec}(\mathsf{prm}, c, d)$:	
$(r\|\|c') \leftarrow G(d)$ (s.t. $\|r\| = \|c'\| = k$)	
Output r if $c' = c$. Otherwise output $\perp$.	

Figure 3.4: Proposed Encapsulation Scheme

3.4.1 Construction

Let $G : \{0,1\}^k \to \{0,1\}^{2k}$ be a PRG (with NCR-k-LSB property). Then we construct an encapsulation scheme $E = (\mathsf{ESetup}, \mathsf{ECom}, \mathsf{ERec})$ as in Fig. 3.4.

3.4.2 Security

In this subsection, we prove hiding and binding properties of the proposed scheme. The proofs for both properties are fairly intuitive and easy to understand. Specifically, pseudorandomness of G provides hiding property and NCR-k-LSB of G provides binding property of the proposed encapsulation scheme E.

Theorem 4. *If G is a (t, ϵ_{prg})-PRG, then the proposed encapsulation scheme E is $(t, 2\epsilon_{prg})$-hiding.*

Proof. Suppose $\mathcal{A}$ is an adversary that breaks (t, q, ϵ_{hide})-hiding property of E, which means that $\mathcal{A}$ with running time t wins the hiding game with probability $\frac{1}{2}+\epsilon_{hide}$. Then we construct a simulator $\mathcal{S}$ who can break $(t, \frac{1}{2}\epsilon_{hide})$-pseudorandomness of G. Our simulator $\mathcal{S}$, simulating the hiding game for $\mathcal{A}$, plays the PRG game with the PRG challenger $\mathcal{C}$ as follows.

Given a $2k$-bit string $y^*_{b_C}$, first $\mathcal{S}$ sets $\mathsf{prm} \leftarrow G$ and $(r^*_1||c^*) \leftarrow y^*_{b_C}$ such that $|r^*_1| = |c^*| = k$, and then picks $b_S \in \{0,1\}$ and $r^*_0 \in \{0,1\}^\kappa$ uniformly at random. $\mathcal{S}$ gives $(\mathsf{prm}, r^*_{b_S}, c^*)$ to $\mathcal{A}$. After $\mathcal{A}$ outputs his guess b_A, $\mathcal{S}$ sets $b'_S \leftarrow 1$ if $b_A = b_S$ or $b'_S \leftarrow 0$ otherwise. Then $\mathcal{S}$ outputs b'_S as its guess.

Next, we estimate the advantage of $\mathcal{S}$. We have

$$\begin{aligned}
\mathsf{Adv}^{\mathrm{PR}}_{G,\mathcal{S}} &= |\Pr[b'_S = b_C] - \frac{1}{2}| \\
&= \frac{1}{2}|\Pr[b'_S = 1|b_C = 1] - \Pr[b'_S = 1|b_C = 0]| \\
&= \frac{1}{2}|\Pr[b_A = b_S|b_C = 1] - \Pr[b_A = b_S|b_C = 0]|.
\end{aligned}$$

To complete the proof, we prove the following claims.

Claim 1. $\Pr[b_A = b_S|b_C = 1] = \frac{1}{2} + \epsilon_{hide}$

Proof of Claim 1. In the case $b_C = 1$, c^* and r^*_1 are computed with G with a uniformly chosen input $d^* \in \{0,1\}^\kappa$ (i.e. $(r^*_1||c^*) = y^*_1 = G(d^*)$). On the other hand, r^*_0 is chosen

uniformly by $\mathcal{S}$. Thus, the view of $\mathcal{A}$ is exactly the same as that in the hiding game (with the challenger's bit is b_S). Therefore, the probability that $b_A = b_S$ occurs is exactly the same as the probability that $\mathcal{A}$ succeeds in guessing in the hiding game, i.e., $\frac{1}{2} + \epsilon_{hide}$. □

Claim 2. $\Pr[b_A = b_S | b_C = 0] = \frac{1}{2}$

Proof of Claim 2. In the case $b_C = 0$, since y_0^* given to $\mathcal{S}$ is a uniformly chosen 2κ-bit string, c^* and r_1^* are both uniformly and independently distributed in $\{0,1\}^\kappa$. Therefore, c^* may not necessarily be in the range of G and thus $\mathcal{S}$'s simulation for $\mathcal{A}$ may be imperfect. Thus, $\mathcal{A}$ may notice that he is in the simulated game and act unfavorably for $\mathcal{S}$. However, r_0^* is also uniformly chosen from $\{0,1\}^\kappa$ by $\mathcal{S}$. Since the distribution of the uniformly distributed value r_1^* and the distribution of a uniformly chosen value r_0^* are perfectly indistinguishable, it is information-theoretically impossible for $\mathcal{A}$ to distinguish r_1^* and r_0^*. Therefore, the probability that $b_A = b_S$ occurs is exactly $\frac{1}{2}$. □

Above shows that if $\mathcal{A}$ wins the hiding game of E with advantage greater than ϵ_{hide}, then $\mathcal{S}$ breaks pseudorandomness of G with advantage greater than $\frac{1}{2}\epsilon_{hide}$, which completes the proof of Theorem 4. □

Theorem 5. *If G is (t, ϵ_{ncr})-NCR-k-LSB, then the proposed encapsulation scheme E is (t, ϵ_{ncr})-binding.*

Proof. Suppose $\mathcal{A}$ is an adversary that breaks (t, ϵ_{bind})-binding property of E, which means that $\mathcal{A}$ with running time t wins the binding game with probability ϵ_{bind}. Then we construct a simulator $\mathcal{S}$ who can break (t, ϵ_{bind})-NCR-κ-LSB property of G. The description of $\mathcal{S}$ is as follows.

Given $d^* \in \{0,1\}^\kappa$ which is chosen uniformly, first $\mathcal{S}$ sets $\mathsf{prm} \leftarrow G$ and computes $(r^*||c^*) \leftarrow G(d^*)$ such that $|r^*| = |c^*| = k$. Then $\mathcal{S}$ gives $(\mathsf{prm}, r^*, c^*, d^*)$ to $\mathcal{A}$. After $\mathcal{A}$ outputs d', $\mathcal{S}$ simply outputs it as its output.

Note that $\mathcal{S}$'s simulation for $\mathcal{A}$ is perfect. Next, we estimate the advantage of $\mathcal{S}$. Let r' and c' be defined as $(r'||c') = G(d')$ such that $|r'| = |c'| = \kappa$. We have

$$\begin{aligned}
\mathsf{Adv}_{G,\mathcal{S}}^{\text{NCR-}\kappa\text{-LSB}} &= \Pr[\kappa\text{-}\mathsf{LSB}(G(d')) = \kappa\text{-}\mathsf{LSB}(G(d^*)) \wedge d' \neq d^*] \\
&= \Pr[\kappa\text{-}\mathsf{LSB}\left(r'||c'\right) = \kappa\text{-}\mathsf{LSB}\left(r^*||c^*\right) \wedge d' \neq d^*] \\
&= \Pr[c' = c^* \wedge d' \neq d^*] \\
&= \Pr[\mathsf{ERec}(\mathsf{prm}, c^*, d') \neq \perp \wedge d' \neq d^*] \\
&\geq \Pr[\mathsf{ERec}(\mathsf{prm}, c^*, d') \notin \{r^*, \perp\} \wedge d' \neq d^*] \\
&= \epsilon_{bind},
\end{aligned}$$

where the transition from the third to the fourth equalities is due to the definition of the recovery algorithm ERec of our encapsulation scheme E in this section. Above means that if $\mathcal{A}$ succeeds in breaking binding property of E with advantage greater than ϵ_{bind}, $\mathcal{S}$ also succeeds in breaking NCR-k-LSB property with advantage greater than ϵ_{bind}, which completes the proof of Theorem 5. □

3.4.3 Concrete Instantiation of PRG with NCR Property

In this subsection, we show a concrete construction of a PRG that has NCR property for practical scenarios. Specifically, we discuss that it is reasonable to assume that (a slight modification of) the PRG currently described in FIPS 186-2, Revised Appendix 3.1 [1] satisfies NCR property. Here, we briefly review the essential construction of the PRG in [1].

Let $H : \{0,1\}^* \to \{0,1\}^m$ be a cryptographic hash function. The construction of a PRG $\mathsf{FIPSPRG}_c^H : \{0,1\}^m \to \{0,1\}^{cm}$ for $c \geq 1$ is as follows:

Step 1. On input $x \in \{0,1\}^m$, set $x_0 \leftarrow x$.
Step 2. Compute $w_i \leftarrow H(x_{i-1})$ and $x_i \leftarrow (1 + x_{i-1} + w_i) \bmod 2^m$ for $1 \leq i \leq c$.
Step 3. Output $(w_1||w_2|| \ldots ||w_c)$.

Then, we define our PRG G^H by interchanging the first and the second m-bit blocks of $\mathsf{FIPSPRG}_2^H$, i.e.,

$$G^H(x) = (\ H(\ (1 + x + H(x)) \bmod 2^m\ ) \ ||\ H(x)\ ).$$

Note that G^H is a PRG as long as $\mathsf{FIPSPRG}_2^H$ is. Moreover, since m-least significant bits of $G^H(x)$ is $H(x)$ itself, if we can assume that H satisfies target collision resistance [83, 12], then we will obviously obtain a PRG with NCR-m-LSB.

Below, we address the above in a more formal manner.

Definition 19. *(FIPS186-2-PRG Assumption) We say that the (t, ϵ)-FIPS186-2-PRG assumption with regard to $\mathsf{FIPSPRG}_c^H$ holds if we can assume that the PRG $\mathsf{FIPSPRG}_c^H$ constructed using a hash function H as above is a (t, ϵ)-PRG.*

Theorem 6. *If the (t, ϵ_{fips})-FIPS186-2-PRG assumption with regard to $\mathsf{FIPSPRG}_2^H$ holds, then G^H constructed as above is a (t, ϵ_{fips})-PRG.*

Proof. Suppose $\mathcal{A}$ is an adversary that breaks the (t, ϵ_{pr})-pseudorandomness of G^H. Then we construct a simulator $\mathcal{S}$ who can break (t, ϵ_{pr})-FIPS186-2-PRG assumption with regard to $\mathsf{FIPSPRG}_2^H$, which means that $\mathcal{S}$ can break the (t, ϵ_{pr})-pseudorandomness of $\mathsf{FIPSPRG}_2^H$. The description of $\mathcal{S}$ is as follows.

Given a $2m$-bit string $y^*_{b_C}$, $\mathcal{S}$ sets z^* as a $2m$-bit string such that the first and the second m-bit blocks of $y^*_{b_C}$ are interchanged. Then $\mathcal{S}$ gives z^* to $\mathcal{A}$. After $\mathcal{A}$ outputs its guess b_A, $\mathcal{S}$ sets $b_S \leftarrow b_A$ and output b_S as its guess.

Notice that $\mathcal{S}$ simulates the experiment of attacking pseudorandomness of G^H perfectly for $\mathcal{A}$. Namely, if $b_C = 1$, i.e., $y^*_{b_C} = \mathsf{FIPSPRG}_2^H(x)$ where $x \in \{0,1\}^m$ is chosen uniformly at random, then z^* given to $\mathcal{A}$ is a $2m$-bit string that is $G^H(x)$ for the uniformly random value x. On the other hand, if $b_C = 0$, i.e., $y^*_{b_C}$ is a uniformly chosen $2m$-bit string, then z^* given to $\mathcal{A}$ is also a uniformly random $2m$-bit string. Therefore, we have

$$\mathsf{Adv}^{\mathrm{PR}}_{\mathsf{FIPSPRG}_2^H,\mathcal{S}} = |\Pr[b_S = b_C] - \frac{1}{2}| = |\Pr[b_A = b_C] - \frac{1}{2}| = \epsilon_{pr}.$$

Above shows that if $\mathcal{A}$ breaks pseudorandomness of G^H with advantage greater than ϵ_{pr}, then $\mathcal{S}$ breaks pseudorandomness of $\mathsf{FIPSPRG}_2^H$ with advantage greater than ϵ_{pr}. This completes the proof of Theorem 6. □

Theorem 7. *If a hash function H that is a building block of G^H is a (t, ϵ_{tcr})-TCRHF, then G^H is (t, ϵ_{tcr})-NCR-m-LSB.*

Proof. Suppose $\mathcal{A}$ is an adversary that breaks the (t, ϵ_{ncr})-NCR-m-LSB of G^H. Then we construct a simulator $\mathcal{S}$ who can break (t, ϵ_{ncr})-target collision resistance of H. The description of $\mathcal{S}$ is as follows.

Given $x^* \in \{0,1\}^m$ which is chosen uniformly at random, $\mathcal{S}$ gives x^* to $\mathcal{A}$. After $\mathcal{A}$ outputs x', $\mathcal{S}$ outputs x' as its own output.

It is easy to see that the $\mathcal{S}$'s simulation of the experiment attacking NCR-m-LSB of G^H for $\mathcal{A}$ is perfect. $\mathcal{S}$'s advantage is estimated as

$$\mathsf{Adv}^{\mathrm{TCR}}_{H,\mathcal{S}} = \Pr[H(x') = H(x^*) \wedge x' \neq x^*] = \Pr[m\text{-}\mathsf{LSB}(G^H(x')) = m\text{-}\mathsf{LSB}(G^H(x^*)) \wedge x' \neq x^*] = \epsilon_{ncr}.$$

Above shows that if $\mathcal{A}$ breaks NCR-m-LSB property of G^H with advantage greater than ϵ_{ncr}, then $\mathcal{S}$ breaks target collision resistance of H with advantage greater than ϵ_{ncr}. This completes the proof of Theorem 7. □

As shown above, since we do not need full power of collision resistance [44] but target collision resistance, we can set $m = \kappa$ for κ-bit security. In practice, (an appropriate modification of) SHA-1 may be used as H. (Though SHA-1 is known to be already broken as a collision resistant hash function [101], it is still reasonable to assume that SHA-1 is target collision resistant.)

Although the FIPS186-2-PRG assumption with regard to $\mathsf{FIPSPRG}_2^H$ is somewhat heuristic (note that the FIPS186-2-PRG assumption with regard to $\mathsf{FIPSPRG}_1^H$ is the same assumption that H with m-bit input space is a PRG), we note that the PRG $\mathsf{FIPSPRG}_c^H$ we introduced here is used (recommended) for generating randomness for Digital Signature Standard (DSS) and is also listed in Recommended techniques of CRYPTREC [2], and thus, using the PRG G^H we presented above as a PRG with NCR-κ-LSB in our encapsulation scheme is fairly reasonable.

One might still think that a PRG with NCR-κ-LSB is a somewhat strong primitive. However, we can actually show that a PRG with NCR-κ-LSB can be constructed from a fairly weak assumption. As addressed in the next section, existence of a PRG with NCR property is generically implied by existence of a one-way permutation which is one of the most fundamental cryptographic primitives. This fact means that a PRG with NCR property is also considered as a sufficiently weak primitive, and therefore, it is not very unreasonable to assume that a carefully designed PRG (like the above example) has NCR property as well.

3.4.4 PRG with Near Collision Resistance from Any One-Way Permutation

The security of the PRG we show in Section 3.4.3 is somewhat heuristic (though we believe it to be fairly reasonable to use in practical scenarios). Here, we show an evidence that a PRG with NCR-κ-LSB is actually a very weak primitive. Specifically, we address that a PRG with NCR-κ-LSB can be generically constructed based on any one-way permutation, which is a fundamental and weak assumption in the area of cryptography. Actually, the construction we show here is the well-known and well-studied PRG by Blum and Micali [18] and Yao [105] (we call the *BMY-PRG*) itself. Namely, the BMY-PRG construction satisfies NCR-κ-LSB property as it is. We briefly review the construction below.

Let $g : \{0,1\}^\kappa \to \{0,1\}^\kappa$ be a one-way permutation and $h : \{0,1\}^\kappa \to \{0,1\}$ be a hardcore bit function of g (e.g. the Goldreich-Levin bit [58]). Then the BMY-PRG $G : \{0,1\}^\kappa \to$

$\{0,1\}^{\kappa+l}$ for $l > 0$ is defined as follows:

$$G(x) = \Big(\; h(x) \;||\; h(g(x)) \;||\; h(g^{(2)}(x)) \;||\; \ldots \;||\; h(g^{(l-1)}(x)) \;||\; g^{(l)}(x) \; \Big),$$

where $g^{(i)}(x) = g(g^{(i-1)}(x))$ and $g^{(1)}(x) = g(x)$. Pseudorandomness of G constructed as above was proved assuming the one-wayness of the permutation g. See [18, 105] for details.

As for NCR-κ-LSB property, it was already mentioned by Boldyreva and Fischlin in [19] that the BMY-PRG has the property. Here, however, we prove for completeness. The following shows that a PRG with NCR-κ-LSB can be actually constructed only from a one-way permutation.

Theorem 8. *([19]) If G is constructed as above, then G is $(t, 0)$-NCR-κ-LSB for any t.*

Proof. According to the definition of the NCR-κ-LSB advantage, for an adversary $\mathcal{A}$, we have

$$\begin{aligned}\mathsf{Adv}^{\text{NCR-}\kappa\text{-LSB}}_{G,\mathcal{A}} &= \Pr[x^* \leftarrow \{0,1\}^\kappa; x' \leftarrow \mathcal{A}(G, x^*) : \kappa\text{-}\mathsf{LSB}(G(x')) = \kappa\text{-}\mathsf{LSB}(G(x^*)) \wedge x' \neq x^*] \\ &= \Pr[x^* \leftarrow \{0,1\}^\kappa; x' \leftarrow \mathcal{A}(G, x^*) : g^{(l)}(x') = g^{(l)}(x^*) \wedge x' \neq x^*].\end{aligned}$$

Since g is a permutation, for any $x, x'(\neq x) \in \{0,1\}^\kappa$ and any $i \geq 1$, we have $g^{(i)}(x) \neq g^{(i)}(x')$. Therefore, of course we have $g^{(l)}(x) \neq g^{(l)}(x')$ for any $x, x'(\neq x) \in \{0,1\}^\kappa$ and any $l > 0$, and thus the above probability equals to zero for any adversary $\mathcal{A}$ with any running time. □

3.5 Comparison

Table 3.1 compares IBE-to-PKE transformations. Our transformation in Section 3.3 is denoted by "Ours (§3.3)". The CHK transformation [34] is denoted by "CHK"), and the original BK transformation [26] is denoted by "BK") [26]. The BK transformation where the encapsulation scheme is instantiated with the original encapsulation scheme by Boneh and Katz (as reviewed in Section 3.2) is denoted by "BK w. BK-encap.", and the BK transformation where the encapsulation scheme used in BK is instantiated by our encapsulation scheme proposed in Section 3.4 is denoted by "BK w. Our-encap.(§3.4)". In Table 3.1, the column "IBE" denotes the security requirement for the underlying IBE schemes ("-sID-CPA" is omitted), the column "Overhead by Transformation" denotes how much the ciphertext size increases from that of the underlying IBE scheme (typical sizes for 128-bit security are given as numerical examples), the column "Required Size for $\mathcal{M}_{IBE}$" denotes how much size is necessary for the plaintext space of the underlying IBE scheme, and the column "Reduction" denotes the ratios of the advantage of breaking the transformed PKE schemes and that of the underlying IBE schemes (i.e. reduction cost)

Ciphertext Overhead by Transformations. In "BK w. BK-encap." scheme, the overhead is caused by a TCRHF (TCR), a MAC, and a large randomness r' (because of the use of the Leftover Hash Lemma [63] with the use of a PIHF in order to get an almost uniformly distributed value for a MAC key). Because of r', though size of $\mathsf{TCR}(r')$ and the tag from MAC can be 128-bit, we need at least 448-bit for the randomness r', and the overhead in total needs to be 704-bit.

In "Ours(§3.3)", the size overhead from the ciphertext of the underlying IBE scheme is caused by a randomness r and its image $f(r)$ with a OWF f. If we require 128-bit security, we can set each to be 128-bit, and thus we have 256-bit overhead in total.

Table 3.1: Comparison among Generic IBE-to-PKE Transformations

	IBE	Overhead by Transformation † (Numerical Example (bit) ‡)		Required Size * for $\mathcal{M}_{IBE}$	Reduction
CHK	IND	$\|\mathsf{vk}\| + \|\mathsf{sig}\|$	(—)	$\|m_{PKE}\|$	tight
BK	IND	$\|\mathsf{com}\| + \|\mathsf{dec}\| + \|\mathsf{MAC}\|$	(—)	$\|m_{PKE}\| + \|\mathsf{dec}\|$	tight
BK w. BK-encap.	IND	$\|\mathsf{TCR}(r')\| + \|r'\| + \|\mathsf{MAC}\|$	(704)	$\|m_{PKE}\| + \|r'\|$	tight
BK w. **Our-encap.(§3.4)**	IND	$2\kappa + \|\mathsf{MAC}\|$	(384)	$\|m_{PKE}\| + \|r'\|$	tight
Ours(§3.3)	NM	$\|f(r)\| + \|r\|$	(256)	$\|m_{PKE}\| + \|r\|$	$1/4q$

† vk and sig denote a verification key and a signature of the one-time signature in the CHK transformation, com and dec denote a commitment and a decommitment of the encapsulation scheme [26], r' denotes a randomness used in the BK transformation when the encapsulation proposed in [26] is used, κ denotes the security parameter, and r denotes a randomness used in our proposed transformation.
‡ We consider 128-bit security for the numerical examples. We used $\kappa = |\mathsf{TCR}(\mathsf{r}')| = |\mathsf{MAC}| = |f(r)| = |r| = 128$ and $|r'| = 448$. We do not consider the cases of the CHK and the BK transformations, because we need to specify the instantiations of one-time signature in CHK and the encapsulation scheme in BK for the size comparison.
* $|m_{PKE}|$ denotes a plaintext size of a PKE scheme obtained via each transformation.

In "BK w. Our-encap.(§3.4)", the overhead is caused by a MAC and an output of a pseudorandom generator (PRG) which has a special security property called *near collision resistance*, which will in total be at least 384-bits. Moreover, unlike the building blocks (e.g. one-time signatures, TCRHFs, and MACs) used in the other transformations including ours, whether a PRG with near collision resistance required in "BK w. Our-encap.(§3.4)", is implied by the existence of IND-sID-CPA secure IBE schemes is not known so far, which means that we have to additionally assume the existence of it. To instantiate a PRG with near collision resistance, we need either a one-way permutation, or a cryptographic hash function that has target collision resistance as well as some pseudorandomness property.

Computation Overhead. We see that, for "Ours (§3.3)" and "BK w. BK-encap.", the overhead of computation costs caused by additional building blocks in the transformations can be considered to be practically the same, because the computation of hash functions are far cheap compared to the computation costs for encryption and decryption algorithms of the used IBE schemes that usually include computations of exponentiations and pairings.

If we use the PRG in Section 3.4.3 for the encapsulation scheme in "BK w. Our-encap.(§3.4)", then the essential efficiency of computations (two computations of a cryptographic hash function) of the encapsulation scheme is comparable to the BK encapsulation scheme (one computation of a cryptographic hash function and one computation of a PIHF, the latter of which is usually a cheap arithmetic computation over some finite field). If we use the PRG in Section 3.4.4 for our encapsulation scheme, then, because of the computation of the BMY-PRG, our encapsulation scheme requires heavier computations for both commitment and recovery algorithms compared to the BK encapsulation scheme. Specifically, for obtaining a 2κ-bit pseudorandom string from κ-bit string with the BMY-PRG, we have to compute a one-way permutation κ times (though this can be reduced to $\kappa/(\log \kappa)$ times by taking not just one bit but $\log \kappa$ bits of hardcore bits in each iteration of the computation of a one-way permutation in the BMY-PRG, this is still far worse than the BK encapsulation

scheme). However, since the computations in these encapsulation schemes are all "symmetric-key" computations, we believe that in most cases they are not so significant compared to the computations done in encryption and decryption algorithms of the used IBE scheme.

Observation: IND vs. NM. As we can see, there exists a trade-off between assumptions on security of the underlying IBE schemes and ciphertext overhead. Roughly speaking, if we see the OWF in our transformation as a hash function, then "Ours (§3.3)" is obtained by getting rid of the PIHF and the MAC from "BK w. BK-encap.". And the lost power is supplied by the property of non-malleability of the underlying IBE scheme. But it is not easy with a brief consideration to come up with an efficient NM-sID-CPA secure IBE scheme from a combination of an IND-sID-CPA secure IBE scheme, a PIHF, and a MAC. Thus, this relation between "Ours (§3.3)" and "BK w. BK-encap.". could be seen as a concrete (but qualitative) evidence that shows a huge gap between what indistinguishability (semantic security) provides and what non-malleability provides (at least, for selective identity, chosen plaintext attacks for IBE schemes).

3.6 Conclusion

We have described a new *simple* and generic IBE-to-PKE transformation that transforms any NM-sID-CPA secure IBE scheme into a CCA secure PKE scheme. Our proof technique for security of the proposed method is not a straightforward application of previous techniques for the security proof of generic IBE-to-PKE transformations [34, 26], and we believe that our technique, together with the simpleness of our transformation itself, is theoretically interesting. Though non-malleability is somewhat a strong requirement and we have no concrete practical non-malleable IBE scheme (other than CCA secure ones) so far, once we in the future have an efficient IBE scheme which are proved (or can be assumed) to be non-malleable, we will immediately have an efficient PKE scheme via our simple and generic transformation.

Moreover, we have also described how to construct an efficient encapsulation scheme from a PRG with near collision resistance, with which we can drastically improve the ciphertext size of PKE schemes obtained from the Boneh-Katz transfomation [26]. From this result, we can construct a more size efficient PKE scheme from any CPA secure IBE scheme.

Chapter 4

Towards CCA Security from CPA Security

Contents

4.1 Introduction 43
4.1.1 Background 43
4.1.2 Our Contribution 45
4.1.3 Related Works 46
4.1.4 Organization of This Chapter 47
4.2 Extending Bounded CCA Security: Mixed CCA Security 47
4.2.1 Mixed CCA Security 48
4.2.2 General Properties of Mixed CCA Security 50
4.3 Relations among Security Notions for Mixed CCA Security . . . 54
4.3.1 "is-Simulatable-by" Relation for Query Sequences 55
4.3.2 Useful Tool for Separation: Backdoor-Sequence Scheme 56
4.3.3 Separation Results 62
4.3.4 Implication Results 78
4.3.5 Necessary and Sufficient Conditions for Implications/Separations . . 81
4.4 Black-box Feasibility Results from IND-CPA Secure PKE Schemes 82
4.5 Open Problems 86
4.6 Conclusion 87

4.1 Introduction

4.1.1 Background

Studies on constructing and understanding public key encryption (PKE) schemes secure against chosen ciphertext attacks (CCA) [84, 93], which is nowadays considered as a standard security notion needed in most practical applications/situations where PKE schemes are used, are important research topics in the area of cryptography. We can roughly categorize the approaches for constructing CCA secure PKE schemes into two types: Constructions from

specific number-theoretic assumptions and constructions from general assumptions. (From now on, we write IND-CCA1 to denote non-adaptive CCA security [84] and IND-CCA2 to denote adaptive CCA security [93])

The approaches of the first type have been successful so far from both a theoretical and practical point of view. After the first novel practical scheme based on the decisional Diffie-Hellman (DDH) assumption by Cramer and Shoup [43], many practical IND-CCA2 secure PKE schemes that pursue smaller ciphertext size and/or base security on weaker assumptions have been constructed so far, e.g. [75, 27, 70, 98, 65, 37, 61, 67, 73, 66, 41, 62]. Especially, the scheme by Cash et al. [37] (and the schemes in recent papers [61, 41, 62]) is based on the computational DH (CDH) assumption, while the scheme by Hofheinz and Kiltz [67] is based on the factoring assumption, and both assumptions are very fundamental in the area of cryptography.

The approaches of the second type have also been successful, mainly from a theoretical point of view. Naor and Yung [84] proposed a generic construction of IND-CCA1 secure PKE schemes from semantically secure (IND-CPA) PKE schemes, using non-interactive zero-knowledge (NIZK) proofs [17]. It is known that if enhanced trapdoor permutations exist, then NIZK proofs for any NP language is possible [15, 56][1]. Based on the Naor-Yung paradigm, several constructions of IND-CCA2 secure PKE schemes were also proposed [47, 96, 77]. Since the existence of enhanced trapdoor permutations implies the existence of IND-CPA secure PKE schemes, these results suggest that we can construct IND-CCA2 secure PKE schemes from any enhanced trapdoor permutation. (We review other generic constructions of IND-CCA2 secure PKE schemes in Section 4.1.3.)

However, one of the most fundamental problems still remains open: *Is it possible to generically construct a CCA (*IND-CCA1 or IND-CCA2*) secure PKE scheme from any* IND-CPA *secure one?*

So far, there are several negative and positive results related to this problem. Gertner et al. [55] showed that constructing an IND-CCA1 secure PKE scheme only from IND-CPA secure PKE schemes in a black-box manner is impossible, if the construction satisfies the property called *shielding*, where a construction of a PKE scheme from another PKE scheme is said to be shielding if the decryption algorithm of the construction does not call the encryption algorithm of the underlying PKE scheme.

Pass et al. [89] showed how to construct a PKE scheme that is non-malleable against chosen plaintext attacks (NM-CPA) from any IND-CPA secure PKE scheme. Their construction uses a certain class of NIZK proofs and thus was non-black-box.

Cramer et al. [40] introduced the notion of *bounded CCA* security which is defined in exactly the same way as ordinary IND-CCA2 security, except that the number of decryption oracle queries that an adversary can ask is bounded by some predetermined value (say, q) that is known a priori (we denote this notion by q-CCA2). Then they showed that for any polynomial q it is possible to construct an IND-q-CCA2 secure PKE scheme from any IND-CPA secure one in a black-box and shielding manner. They furthermore showed that for any polynomial q it is possible to construct a PKE scheme that satisfies non-malleability against q-bounded CCA (NM-q-CCA2) in a non-black-box manner.

Recently, Choi et al. [38] showed the constructions of PKE schemes from any IND-CPA secure scheme both in a black-box and shielding manner. Their first construction achieves NM-CPA security, and their second construction, which is essentially the same as the first

[1] It was shown in [57] that we actually need the so-called *doubly-enhanced* trapdoor permutations.

construction but needs larger parameters, can achieve NM-q-CCA2 security.

These previous results show that we can achieve the best possible security notion (NM-q-CCA2) in the bounded CCA framework. This suggests that in order to proceed from the current situation, we would need new security notions which are intermediate between CPA and CCA security in a different sense from bounded CCA security. The motivation of this chapter is to introduce and study such intermediate security notions as an extension of the bounded CCA security as a foundation for tackling the above fundamental problem.

Extending Bounded CCA Security with Parallel Decryption Queries. As we mentioned, we would need intermediate security notions that can capture the notions between CPA and CCA security, in order to make further progress on the fundamental problem. For this purpose, we focus on and use the concept of the *parallel chosen ciphertext attacks* which is originally introduced by Bellare and Sahai [13, 14] in the context of non-malleability [47] for PKE schemes[2], and considers the parallel queries in the bounded CCA security framework. More specifically, as an extension of bounded CCA security, we introduce a new security notion, which we call *mixed CCA security*, that captures security against adversaries that make single (i.e. ordinary) decryption queries and parallel decryption queries in a predetermined order, where each parallel query can contain *unboundedly* many ciphertexts. (The name "mixed" is because we consider a mix of single and parallel queries.) Moreover, the difference among decryption queries that are only allowed to make before/after the challenge and those that are allowed to make both before and after the challenge (an adversary can decide "flexibly" how to issue queries as long as it follows the predetermined order of queries and types) is also taken into account in this security definition, which enables us to capture existing major security notions that lie between CPA and CCA security, including slightly complex notions such as non-malleability against bounded CCA (NM-q-CCA2) that considers "stage-specific" decryption queries, in a unified security notion. As a natural and interesting special class of mixed CCA security, we also introduce the notion of *bounded parallel CCA* security. For more details, see Section 4.2. We believe that the mixed CCA security notions provides a theoretical foundation for discussion of the problem of whether constructing (unbounded) CCA secure PKE schemes from any CPA secure PKE schemes is possible or not, and for intermediate results towards the problem.

4.1.2 Our Contribution

Our contributions are summarized as follows:

Relations among Mixed CCA Security Notions. We investigate the relations among mixed CCA security notions for PKE schemes and for key encapsulation mechanisms (KEMs), and show a necessary and sufficient condition for implications/separations between any given two notions in mixed CCA security (which includes major existing security notions that lie between CPA and CCA security). Interestingly and perhaps somewhat surprisingly, *the relations for PKE schemes differ depending on its plaintext space size.* More specifically, the relations among security notions for PKE schemes with superpolynomially large plaintext

[2]They used the notion of parallel CCA (in which a "parallel decryption query" is available only after all decryption queries are done in the second stage) in order to show the equivalence of among several type of non-malleability definitions.

space size and those with polynomially bounded plaintext space size are different. Therefore, this difference suggests that when we consider the relations among security notions for PKE schemes, we have to be also careful about the plaintext space size, though seemingly unrelated. The relations for KEMs are the same as those of PKE schemes with polynomially bounded plaintext space size. See Section 4.3 for details. There, as a corollary of the above general result of a necessary and sufficient condition, we also fully clarifies the relations among bounded parallel CCA security and other existing security notions. We believe that the relations among security notions clarified here will be useful for further studying mixed CCA security.

Black-Box Feasibility Results from CPA-Security. Using the notion of mixed CCA security, in Section 4.4, we will show two new black-box constructions of PKE scheme (which can encrypt plaintexts of polynomial length) from an IND-CPA secure PKE scheme. The first one is constructed based on the construction by Choi et al. [38] which is NM-q-CCA2 secure, and achieves slightly but strictly stronger security notion than NM-q-CCA2. Our approach for the first construction is to use the Choi et al. scheme as a KEM and combine it with an IND-CCA2 secure data encapsulation mechanism (DEM), and thus is a very simple extension. In order for this simple approach to work, we show some implication result for mixed CCA security of KEMs (and PKE schemes with polynomially bounded plaintext space size). The second one is constructed based on the above result and the construction of PKE scheme by Cramer et al. [40], and achieves yet another security notion which cannot be directly compared with the security notion achieved by our other constructions and with NM-q-CCA2 security. See Section 4.4 for details.

As will be explained later, one of the important and interesting observations that our results suggest, combined with previously known results, is that *the difficulty of constructing an* IND-CCA1 *secure PKE scheme only from* IND-CPA *secure one lies not in whether the number of decryption results that the adversary can see is bounded or not, but in whether the number of an adversary's "adaptive" decryption queries is bounded.* To the best of our knowledge, this observation has not been explicitly stated before.

4.1.3 Related Works

Here, we review several generic constructions of IND-CCA2 secure PKE schemes that are not mentioned in Section 4.1.1. We note that assuming the existence of the building blocks for the constructions below is strictly stronger than assuming the existence of IND-CPA secure PKE schemes.

Canetti et al. [34] proposed a novel methodology for achieving IND-CCA2 security from any semantically secure identity-based encryption (IBE) scheme [99]. Kiltz [69] showed that the IBE-to-PKE transformation paradigm is applicable to tag-based encryption [78] of appropriate security, the existence of which is implied by the existence of semantically secure IBE schemes.

Peikert and Waters [91] proposed a methodology to obtain IND-CCA2 secure PKE schemes using a primitive called *lossy trapdoor function*. Rosen and Segev [95] proposed a generic paradigm for obtaining IND-CCA2 secure PKE schemes from an injective trapdoor function that is one-way under *correlated products*, which is a strictly weaker primitive than a lossy trapdoor function. Kiltz et al. [71] showed that an *adaptive trapdoor function*, which is a strictly weaker primitive than the above two special trapdoor functions, is sufficient to

construct IND-CCA2 secure PKE schemes. Very recently, Wee [104] further relaxed the requirement of adaptive trapdoor functions and showed that *adaptive trapdoor relations* is sufficient.

Hanaoka and Kurosawa [61] proposed yet another paradigm to achieve IND-CCA2 secure PKE schemes from any semantically secure *broadcast encryption with verifiability*. Dowsley et al. [48] proposed the generic construction of IND-CCA2 secure PKE schemes from IND-CPA secure PKE schemes that are *verifiable under k-repetition* which is currently only known to be achieved by some schemes such as the randomized McEliece encryption scheme [86].

Pandey et al. [88] introduced the notion of *adaptive security* for one-way functions and related primitives, and showed that if there exist an adaptively secure perfectly one-way hash function [29, 36] and a trapdoor permutation of a special kind, then the PKE scheme by Bellare and Rogaway [10] can be shown to be IND-CCA2 secure in the standard model, although the original scheme requires random oracles in order to show its CCA security. However, the relations among existence of such adaptively secure cryptographic primitives and those with non-adaptive property are not known well.

Canetti and Dakdouk [31] introduced a new primitive called an *extractable perfectly one-way function*. They showed that using a trapdoor permutation and an extractable perfectly one-way function with *dependent auxiliary information and public randomness*, then the PKE scheme by Bellare and Rogaway [10] can be shown to be IND-CCA2 secure in the standard model. However, how to construct an extractable perfectly one-way function with such properties is not known yet.

In the random oracle methodology [10], several generic methodologies (e.g., [11, 49, 87]) are known. However, since the results from several papers [32, 85, 59, 6, 46, 72] have shown that this methodology has some problems, in this thesis we focus only on the constructions in the standard model.

Myers and Shelat [81] recently showed that we can construct IND-CCA2 secure PKE scheme which can encrypt multi-bit messages from IND-CCA2 secure PKE schemes which can encrypt only 1-bit in a black-box manner.

4.1.4 Organization of This Chapter

In Section 4.2, we define the notion of mixed CCA security, and show some important properties of mixed CCA security notions. Then, in Section 4.3 we investigate the relations among mixed CCA security notions. We show the results on black-box constructions of PKE schemes from CPA secure ones in Section 4.4. We leave several open problems regarding mixed CCA security in Section 4.5. Section 4.6 is the conclusion of this chapter.

Publication Information. The results shown in this chapter will be presented as [a] (see Appendix A).

4.2 Extending Bounded CCA Security: Mixed CCA Security

In order to deal with and discuss existing security notions for PKE schemes and KEMs that lie between IND-CPA and IND-CCA2 security in a unified way, in this section we introduce an extension of conventional bounded CCA security, which we call security against *mixed*

chosen ciphertext attacks (mixed CCA security), where the decryption oracle in the security experiment accepts both single decryption queries and *parallel* decryption queries in a predetermined order, and "how" the decryption oracle is available is also predetermined.

Preliminary Definitions. We first formally define the notion of a parallel query to an oracle.

Definition 20. *Let $\mathcal{O} : \{0,1\}^* \to \{0,1\}^*$ be an oracle which on input x outputs $y = \mathcal{O}(x)$. A parallel query to $\mathcal{O}$ is a vector $\overrightarrow{x} = (x_1, x_2, \dots)$ of inputs for $\mathcal{O}$, where the size of the vector $\overrightarrow{x}$ is not predetermined, and a response to the parallel query $\overrightarrow{x}$ is a vector of outputs $\overrightarrow{y} = (y_1, y_2, \dots)$ where $y_i = \mathcal{O}(x_i)$ for every $1 \leq i \leq |\overrightarrow{x}|$.*

Here, we stress that we make no restriction on the size of each parallel query. That is, the number of inputs in each parallel query $\overrightarrow{x}$ is unbounded and can be dependent only on an algorithm that uses the oracle.

To define mixed CCA security, we need to introduce several notations.

The symbols "s" and "p" denote one *single query* and one *parallel query*, respectively. Let $q \geq 0$ be an integer. "s^q" and "p^q" denote q single queries and q parallel queries, respectively. We define $\mathsf{s}^0 = \mathsf{p}^0 = \emptyset$.

If we write "$(\mathsf{s}^{q_1}\mathsf{p}^{q_2}\dots)$" with some integers $q_1, q_2, \dots \geq 0$, then it denotes a *query sequence*. This query sequence will define how the decryption oracle in the mixed CCA experiment accepts the queries. For example, $(\mathsf{s}^2\mathsf{p}^3)$ denotes two single decryption queries followed by three parallel decryption queries. We denote by "$\mathsf{unbound}$" a special sequence that indicates "unboundedly" many single queries, i.e. $\mathsf{unbound} = \mathsf{s}^\infty$.

"$\mathcal{QS}$" denotes a set consisting of all possible query sequences with the restriction that the total number of queries in each sequence is bounded to be polynomial (in the security parameter). We furthermore define $\mathcal{QS}^* = \mathcal{QS} \cup \{\mathsf{unbound}\}$. We refer to queries following the query sequence $\mathsf{seq} \in \mathcal{QS}^*$ as "seq-queries".

If $\mathsf{seq} \in \mathcal{QS}$, then we denote by "$|\mathsf{seq}|$" the length of the query sequence. For example, if $\mathsf{seq} = (\mathsf{s}^2\mathsf{p})$ then $|\mathsf{seq}| = 3$. We define $|\mathsf{unbound}| = \infty$.

We define a concatenation operation "$||$" for query sequences naturally. For example, if $\mathsf{seq}_1 = (\mathsf{s}^2\mathsf{p})$ and $\mathsf{seq}_2 = (\mathsf{p}^2\mathsf{s}^3)$, then $(\mathsf{seq}_1||\mathsf{seq}_2) = (\mathsf{s}^2\mathsf{p}\mathsf{p}^2\mathsf{s}^3) = (\mathsf{s}^2\mathsf{p}^3\mathsf{s}^3)$. For any $\mathsf{seq} \in \mathcal{QS}^*$, we define $(\mathsf{seq}||\emptyset) = (\emptyset||\mathsf{seq}) = \mathsf{seq}$ and $(\mathsf{seq}||\mathsf{unbound}) = (\mathsf{unbound}||\mathsf{seq}) = \mathsf{unbound}$.

4.2.1 Mixed CCA Security

Now we define mixed CCA security as a special type of `IND-ATK` security parameterized by three query sequences $\mathsf{B}, \mathsf{F}, \mathsf{A} \in \mathcal{QS}^*$, denoted by $\langle \mathsf{B} : \mathsf{F} : \mathsf{A} \rangle$-`mCCA` security, via the "$\langle \mathsf{B} : \mathsf{F} : \mathsf{A} \rangle$-`mCCA` experiment". In the $\langle \mathsf{B} : \mathsf{F} : \mathsf{A} \rangle$-`mCCA` experiment, an adversary $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$ can issue the queries following the sequences firstly B, secondly F, and lastly A, in exactly this order with the following restriction: B-queries are available only before the challenge ("B" is used to denote "before" the challenge). F-queries are available only after all the B-queries are completed. However, as long as the order and the number of queries are maintained, the queries can be issued before and after the challenge ("F" is used to denote "flexible" in the sense that $\mathcal{A}$ can "flexibly" decide how it issues queries before and after the challenge). A-queries are available only after the challenge and only after all the F-queries are completed ("A" is used to denote "after" the challenge). In other words, an adversary in the $\langle \mathsf{B} : \mathsf{F} : \mathsf{A} \rangle$-`mCCA` experiment can issue $(\mathsf{B}||\mathsf{F}_1)$-queries before the challenge and $(\mathsf{F}_2||\mathsf{A})$-queries after the

Table 4.1: Compatibility with Existing Security Notions.

Existing Notions	Notation in Mixed CCA Security
IND-CPA	$\langle \emptyset :: \emptyset \rangle$-mCCA
NM-CPA	$\langle \emptyset :: \mathsf{p} \rangle$-mCCA
IND-q-CCA2	$\langle \emptyset : \mathsf{s}^q : \emptyset \rangle$-mCCA
NM-q-CCA2	$\langle \emptyset : \mathsf{s}^q : \mathsf{p} \rangle$-mCCA
IND-q-pCCA1	$\langle \mathsf{p}^q :: \emptyset \rangle$-mCCA
NM-q-pCCA1	$\langle \mathsf{p}^q :: \mathsf{p} \rangle$-mCCA
IND-q-pCCA2	$\langle \emptyset : \mathsf{p}^q : \emptyset \rangle$-mCCA
NM-q-pCCA2	$\langle \emptyset : \mathsf{p}^q : \mathsf{p} \rangle$-mCCA
IND-CCA1	$\langle \mathsf{unbound} :: \emptyset \rangle$-mCCA
NM-CCA1	$\langle \mathsf{unbound} :: \mathsf{p} \rangle$-mCCA
IND-CCA2	$\langle \mathsf{unbound} :: \mathsf{unbound} \rangle$-mCCA

challenge, for any pair of query sequences $(\mathsf{F}_1, \mathsf{F}_2)$ satisfying $(\mathsf{F}_1||\mathsf{F}_2) = \mathsf{F}$, and how F is split into F_1 and F_2 can be decided adaptively by an adversary in the experiment.

We refer to B, F, and A as "before-challenge" queries, "flexible" queries, and "after-challenge" queries, respectively. Just for notational convenience, if $\mathsf{F} = \emptyset$ then we write $\langle \mathsf{B} :: \mathsf{A} \rangle$-mCCA, instead of $\langle \mathsf{B} : \emptyset : \mathsf{A} \rangle$-mCCA. (We do not omit if $\mathsf{B} = \emptyset$ or $\mathsf{A} = \emptyset$.)

The advantage of an adversary $\mathcal{A}$ in the $\langle \mathsf{B} : \mathsf{F} : \mathsf{A} \rangle$-mCCA experiment regarding a PKE scheme Π, denoted by $\mathsf{Adv}^{\langle \mathsf{B:F:A} \rangle\text{-mCCA}}_{\Pi,\mathcal{A}}(k)$, is defined similarly to the other security notions (which are defined in Section 2.2).

Definition 21. *Let* $\mathsf{B}, \mathsf{F}, \mathsf{A} \in \mathcal{QS}^*$. *We say that a PKE scheme* Π *is* $\langle \mathsf{B} : \mathsf{F} : \mathsf{A} \rangle$-mCCA *secure if* $\mathsf{Adv}^{\langle \mathsf{B:F:A} \rangle\text{-mCCA}}_{\Pi,\mathcal{A}}(k)$ *is negligible for any PPTA* $\mathcal{A}$.

We define mixed CCA security for KEMs in exactly the same way as above.

With the mixed CCA security notation, we can express all the existing security notions reviewed in Section 2.2. These are summarized in Table 4.1. For non-malleability, we adopt the characterization using a parallel query by Bellare and Sahai [13, 14]. We also include the bounded parallel CCA security notions defined in the following paragraph.

We also included bounded parallel CCA security notions defined in Section 4.2.1 in the table.

We remark that we can also define a parallel decryption query in mixed CCA security experiment (i.e. the $\langle \mathsf{B} : \mathsf{F} : \mathsf{A} \rangle$-mCCA experiment) so that the number of ciphertexts contained in each parallel query is also bounded to be some predetermined value (say, t). However, such security definition is implied by $(|(\mathsf{B}||\mathsf{F}||\mathsf{A})| \cdot t)$-Bounded CCA security, which is already achieved by the existing PKE schemes that are constructed only from IND-CPA secure schemes by the previous results [40, 38]. Therefore, we think that studying security with such "bounded parallel" queries is less interesting than studying mixed CCA security defined in this section, and is not treated in this thesis.

Previously to our work, Phan and Pointcheval [92] defined a similar notion which they call (i, j)-IND security and (i, j)-NM security, which are equivalent to $\langle \mathsf{s}^i :: \mathsf{s}^j \rangle$-mCCA security and $\langle \mathsf{s}^i :: \mathsf{s}^j \mathsf{p} \rangle$-mCCA security in our definition, respectively (for NM, we interpret it with parallel CCA-based characterization in [13]). They did not consider the "flexible" F-queries.

Bounded Parallel CCA Security. Here, we define a natural and interesting special class of mixed CCA security which we call *bounded parallel CCA* security. This is the security against adversaries whose decryption queries are always parallel, and is also a natural extension from the bounded CCA security defined by Cramer et al. [40].

Depending on how the oracle is available for an adversary, we define pCCA1 and pCCA2 as natural bounded parallel CCA analogue of CCA1 and CCA2. Moreover, as is similar to the existing security notions, we define indistinguishability (IND) and non-malleability (NM) as well.

Definition 22. *Let $q \geq 0$ be an integer. We say that a PKE scheme is* IND-q-pCCA1 *(resp.* IND-q-pCCA2*,* NM-q-pCCA1*, and* NM-q-pCCA2*) secure if it is* $\langle \mathsf{p}^q :: \emptyset \rangle$-mCCA *(resp.* $\langle \emptyset : \mathsf{p}^q : \emptyset \rangle$-mCCA*,* $\langle \mathsf{p}^q :: \mathsf{p} \rangle$-mCCA*, and* $\langle \emptyset : \mathsf{p}^q : \mathsf{p} \rangle$-mCCA*) secure.*

We define the bounded parallel CCA security notions for KEMs in the same way.

We remark that the above definition is interesting only if the number of the adversary's oracle queries is predetermined as in the conventional bounded CCA security [40]. This is because if it is unbounded, then such security is implied by the ordinary CCA security which allows unbounded queries for the adversary. That is, for $\mathtt{GOAL} \in \{\mathtt{IND}, \mathtt{NM}\}$, "GOAL-$q$-pCCA2 security for any polynomial q" (resp. "GOAL-q-pCCA1 security for any polynomial q") implies GOAL-CCA2 (resp. GOAL-CCA1) security. (This is also true for ordinary (i.e. not parallel) bounded CCA case.)

4.2.2 General Properties of Mixed CCA Security

Here, we show two general implication results about the mixed CCA security notions. (In the following, we always assume $\mathsf{B}, \mathsf{F}, \mathsf{A} \in \mathcal{QS}^*$, and do not write it explicitly.)

Firstly, by noticing the property of the "flexible" queries F, we show the following.

Theorem 9. *For both PKE schemes and KEMs, a combination of all security notions* $\langle (\mathsf{B}||\mathsf{F}_1) :: (\mathsf{F}_2||\mathsf{A}) \rangle$-mCCA *satisfying* $(\mathsf{F}_1||\mathsf{F}_2) = \mathsf{F}$ *implies* $\langle \mathsf{B} : \mathsf{F} : \mathsf{A} \rangle$-mCCA *security.*

Since this theorem is almost trivial, we omit the proof and only mention the intuition using the simplest case $\mathsf{F} = \mathsf{s}$. It is easy to see that $\langle \mathsf{B} : \mathsf{s} : \mathsf{A} \rangle$-mCCA adversary can be divided into two types: The first type adversary who makes $(\mathsf{B}||\mathsf{s})$-queries before the challenge, and A-queries after the challenge, and the second type who makes B-queries before, and $(\mathsf{s}||\mathsf{A})$-queries after the challenge. Then, the experiment for the first type can be simulated by a $\langle (\mathsf{B}||\mathsf{s}) :: \mathsf{A} \rangle$-mCCA adversary while the experiment for the second type can be simulated by a $\langle \mathsf{B} :: (\mathsf{s}||\mathsf{A}) \rangle$-mCCA adversary. This is easily extended to any $\mathsf{F} \in \mathcal{QS}$ case. Note that if $\mathsf{F} = \mathsf{unbound}$, then the statement is again trivial because we can have $\mathsf{F}_1 = \mathsf{F}_2 = \mathsf{unbound}$ (since $\mathsf{unbound} = (\mathsf{unbound}||\mathsf{unbound})$), and thus in this case $\langle (\mathsf{B}||\mathsf{F}_1) :: (\mathsf{F}_2||\mathsf{A}) \rangle$-mCCA security is $\langle \mathsf{unbound} :: \mathsf{unbound} \rangle$-mCCA security, which is equivalent to IND-CCA2 security that implies all the mixed CCA security notions.

Next, we show that for PKE schemes with polynomially bounded plaintext space size and for KEMs, the A-queries in the $\langle \mathsf{B} : \mathsf{F} : \mathsf{A} \rangle$-mCCA experiment, which is intended to be only available after the challenge, can be actually issued "flexibly", i.e., can be combined into the "flexible" F-queries with maintaining its order.

Theorem 10. *For PKE schemes with polynomially bounded plaintext space size and for KEMs,* $\langle \mathsf{B} : \mathsf{F} : \mathsf{A} \rangle$-mCCA *security implies* $\langle \mathsf{B} : (\mathsf{F}||\mathsf{A}) : \emptyset \rangle$-mCCA *security.*

Intuitively, showing this theorem is possible because the challenge ciphertext can be made "in advance" for PKE schemes with polynomially bounded plaintext space size and for KEMs.
Proof. Since the proof for the KEM case is easily inferred from that of the PKE case, here we only show the PKE case (which we think is more interesting). Firstly, we have use the following results on PKE schemes with polynomially bounded plaintext space size:

Lemma 9. *If a PKE scheme* Π *whose plaintext space size* $|\mathcal{M}_\Pi|$ *is polynomially bounded is* `IND-CPA` *secure, then* Π *is smooth.*

The proof is given after the proof of this theorem, The KEM-analogue of this lemma was shown in [8].

Let $\Pi = (\mathsf{PKG}, \mathsf{PEnc}, \mathsf{PDec})$ be a $\langle \mathsf{B} : \mathsf{F} : \mathsf{A} \rangle$-`mCCA` secure PKE scheme whose plaintext space size $|\mathcal{M}_\Pi|$ is polynomially bounded. First of all, since $\langle \mathsf{B} : \mathsf{F} : \mathsf{A} \rangle$-`mCCA` security implies `IND-CPA` security, and the size of the plaintext space $\mathcal{M}_\Pi$ of Π is polynomially bounded, Π is smooth by Lemma 9. (Recall the definition of smoothness for PKE schemes in Definition 3 in Section 2.2.) Therefore, Smth_Π is negligible.

Now, assume towards a contradiction that there exists a PPTA adversary $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$ that succeeds in guessing the challenge bit, in the $\langle \mathsf{B} : (\mathsf{F}||\mathsf{A}) : \emptyset \rangle$-`mCCA` experiment regarding Π, with probability $\frac{1}{2} + \mathsf{Adv}^{\langle \mathsf{B}:(\mathsf{F}||\mathsf{A}):\emptyset\rangle\text{-mCCA}}_{\Pi,\mathcal{A}} = \frac{1}{2} + \delta$ and δ is not negligible. Then we show that we can construct another adversary $\mathcal{B} = (\mathcal{B}_1, \mathcal{B}_2)$ that has non-negligible advantage in the $\langle \mathsf{B} : \mathsf{F} : \mathsf{A} \rangle$-`mCCA` experiment regarding the same Π. The description of $\mathcal{B}$ is as follows.

$\mathcal{B}_1$: On input pk, $\mathcal{B}_1$ runs $\mathcal{A}_1$ with input pk. $\mathcal{B}_1$ answers to $\mathcal{A}_1$'s B-queries by the access to $\mathcal{B}$'s own decryption oracle in the $\langle \mathsf{B} : \mathsf{F} : \mathsf{A} \rangle$-`mCCA` experiment. Then $\mathcal{B}_1$ picks m'_0 and m'_1 uniformly at random from the plaintext space $\mathcal{M}_\Pi$, sets the state information $\mathsf{st}_\mathcal{B}$ that consists of all the values known to $\mathcal{B}_1$, and terminates with output $(m'_0, m'_1, \mathsf{st}_\mathcal{B})$.

$\mathcal{B}_2$: On input $(c^*, \mathsf{st}_\mathcal{B})$ where c^* is the challenge ciphertext for $\mathcal{B}$, $\mathcal{B}_2$ waits until $\mathcal{A}_1$ makes further decryption queries. At this point, $\mathcal{A}_1$ is allowed to make decryption queries following the query sequence F, and $\mathcal{B}_2$ is allowed to make decryption queries following the query sequence $(\mathsf{F}||\mathsf{A})$. Here, we suppose that after this point $\mathcal{A}_1$ will issue F_1-queries before $\mathcal{A}$'s challenge and $\mathcal{A}_2$ will issue $(\mathsf{F}_2||\mathsf{A})$-queries after $\mathcal{A}$'s challenge, such that $(\mathsf{F}_1||\mathsf{F}_2) = \mathsf{F}$ holds. We stress that this is just for notational convenience for the description of $\mathcal{B}$. How F_1 and F_2 are split is dependent only on $\mathcal{A}$ and need not be known to $\mathcal{B}$ in advance. $\mathcal{B}_2$ answers to $\mathcal{A}_1$'s F_1-queries by using $\mathcal{B}$'s own decryption oracle access, except that if some query from $\mathcal{A}_1$ contains c^*, then $\mathcal{B}_2$ gives up and aborts. When $\mathcal{A}_1$ terminates with output $(m_0, m_1, \mathsf{st}_\mathcal{A})$, $\mathcal{B}_2$ checks if both $m_0 = m'_0$ and $m_1 = m'_1$ hold. If this is not the case, then $\mathcal{B}_2$ picks a bit $b' \leftarrow \{0, 1\}$ uniformly at random and terminates with output b'. Otherwise (i.e. $m_0 = m'_0$ and $m_1 = m'_1$ both hold), $\mathcal{B}_2$ runs $\mathcal{A}_2$ with input $(c^*, \mathsf{st}_\mathcal{A})$. $\mathcal{B}_2$ answers to $\mathcal{A}_2$'s $(\mathsf{F}_2||\mathsf{A})$-queries again by using $\mathcal{B}$'s own decryption oracle access. When $\mathcal{A}_2$ terminates with output b', $\mathcal{B}_2$ outputs this b' as its guess and terminates.

Note that according to our description of $\mathcal{B}$, the challenge ciphertext c^* for $\mathcal{B}$ is never submitted to the decryption oracle in the second stage of $\mathcal{B}$'s own $\langle \mathsf{B} : \mathsf{F} : \mathsf{A} \rangle$-`mCCA` experiment.

Let Succ be the event that $\mathcal{B}$ succeeds in guessing the challenge bit b, and let Coll be the event that $\mathcal{A}_1$ issues a (single or parallel) decryption query that makes $\mathcal{B}_2$ abort, i.e. some of $\mathcal{A}_1$'s decryption queries contain the challenge ciphertext c^* of $\mathcal{B}$. In other words, Coll corresponds to the event that $\mathcal{A}_1$ makes a collision of the challenge ciphertext c^* without

seeing c^*. Moreover, let MsgOK be the event that both $m_0 = m'_0$ and $m_1 = m'_1$ hold. If MsgOK occurs, then $\mathcal{B}_2$ does not terminates when $\mathcal{B}_2$ receives the challenge plaintexts from $\mathcal{A}_1$.

We have the following:

$$
\begin{aligned}
&\Pr[\mathsf{Succ}] \\
&\geq \Pr[\mathsf{Succ} \wedge \overline{\mathsf{Coll}}] \\
&= \Pr[\mathsf{Succ}|\overline{\mathsf{Coll}}] \cdot (1 - \Pr[\mathsf{Coll}]) \\
&\geq \Pr[\mathsf{Succ}|\overline{\mathsf{Coll}}] - \Pr[\mathsf{Coll}] \\
&= \Pr[\mathsf{Succ} \wedge \mathsf{MsgOK}|\overline{\mathsf{Coll}}] + \Pr[\mathsf{Succ} \wedge \overline{\mathsf{MsgOK}}|\overline{\mathsf{Coll}}] - \Pr[\mathsf{Coll}] \\
&= \Pr[\mathsf{Succ}|\mathsf{MsgOK} \wedge \overline{\mathsf{Coll}}] \cdot \Pr[\mathsf{MsgOK}|\overline{\mathsf{Coll}}] + \Pr[\mathsf{Succ}|\overline{\mathsf{MsgOK}} \wedge \overline{\mathsf{Coll}}] \cdot \Pr[\overline{\mathsf{MsgOK}}|\overline{\mathsf{Coll}}] - \Pr[\mathsf{Coll}]
\end{aligned}
$$

Here, note that $\Pr[\mathsf{MsgOK}|\overline{\mathsf{Coll}}] = \Pr[\mathsf{MsgOK}] = \frac{1}{|\mathcal{M}_\Pi|^2}$ (and thus $\Pr[\overline{\mathsf{MsgOK}}|\overline{\mathsf{Coll}}] = 1 - \frac{1}{|\mathcal{M}_\Pi|^2}$). This is because m'_0 and m'_1 are chosen uniformly at random from $\mathcal{M}_\Pi$ and therefore for any massage pair (m_0, m_1), the probability that $(m'_0, m'_1) = (m_0, m_1)$ occurs is exactly $\frac{1}{|\mathcal{M}_\Pi|^2}$.

Note also that we have $\Pr[\mathsf{Succ}|\overline{\mathsf{MsgOK}} \wedge \overline{\mathsf{Coll}}] = \frac{1}{2}$, because if $\overline{\mathsf{MsgOK}}$ occurs, $\mathcal{B}_2$ outputs a random bit b'.

On the other hand, we have $\Pr[\mathsf{Succ}|\mathsf{MsgOK} \wedge \overline{\mathsf{Coll}}] = \frac{1}{2} + \delta$. This is because if both $\overline{\mathsf{Coll}}$ and MsgOK occur, $\mathcal{B}$ perfectly simulates the $\langle \mathsf{B} : (\mathsf{F}||\mathsf{A}) : \emptyset \rangle$-$\mathtt{mCCA}$ experiment for $\mathcal{A}$. Specifically, unless Coll occurs, the answers to all the decryption queries from $\mathcal{A}$ are also perfect due to the use of $\mathcal{B}$'s own decryption oracle. (Recall that the order of the queries (i.e., the order of B, F, and A) for $\mathcal{B}$ and $\mathcal{A}$ are exactly the same.) Moreover, if MsgOK occurs, then the challenge ciphertext c^* given to $\mathcal{A}$ is a "correct" challenge ciphertext for $\mathcal{A}$'s own experiment. Therefore, if both $\overline{\mathsf{Coll}}$ and MsgOK occur, $\mathcal{B}$ succeeds with exactly the same probability as $\mathcal{A}$ succeeds in the real $\langle \mathsf{B} : (\mathsf{F}||\mathsf{A}) : \emptyset \rangle$-$\mathtt{mCCA}$ experiment.

Let Q be a total number of ciphertexts contained in $\mathcal{A}_1$'s F_1-queries. Since $\mathcal{A}$ is a PPTA adversary, Q is a polynomial. Then by the definition of the smoothness and considering the union bound over all the ciphertext in $\mathcal{A}_1$'s F_1-queries, we have $\Pr[\mathsf{Coll}] \leq Q \cdot \mathsf{Smth}_\Pi$. Hence, since Smth_Π is negligible, $\Pr[\mathsf{Coll}]$ is upperbounded to be negligible.

Using the above, we now have,

$$
\begin{aligned}
\Pr[\mathsf{Succ}] &\geq (\frac{1}{2} + \delta) \cdot \frac{1}{|\mathcal{M}_\Pi|^2} + \frac{1}{2} \cdot (1 - \frac{1}{|\mathcal{M}_\Pi|^2}) - Q \cdot \mathsf{Smth}_\Pi \\
&= \frac{1}{2} + \frac{1}{|\mathcal{M}_\Pi|^2} \cdot \delta - Q \cdot \mathsf{Smth}_\Pi
\end{aligned}
$$

Therefore, $\mathsf{Adv}^{\langle \mathsf{B:F:A} \rangle\text{-}\mathtt{mCCA}}_{\Pi,\mathcal{B}} = |\Pr[\mathsf{Succ}] - \frac{1}{2}| = |\frac{1}{|\mathcal{M}_\Pi|^2} \cdot \delta - Q \cdot \mathsf{Smth}_\Pi|$, which is not negligible due to the facts that $|\mathcal{M}_\Pi|$ and Q are polynomial and Smth_Π is negligible, and by the assumption we made at the beginning of the proof of this theorem that δ is not negligible. Since this contradicts the $\langle \mathsf{B} : \mathsf{F} : \mathsf{A} \rangle$-$\mathtt{mCCA}$ security of Π, $\mathsf{Adv}^{\langle \mathsf{B:(F||A):\emptyset} \rangle\text{-}\mathtt{mCCA}}_{\Pi,\mathcal{A}}$ must be negligible for any PPTA adversary $\mathcal{A}$. This completes the proof of Theorem 10. □

Now, we prove Lemma 9 in the following.

Lemma 9 (Restated). *If a PKE Π whose plaintext space size $|\mathcal{M}_\Pi|$ is polynomially bounded is* $\mathtt{IND\text{-}CPA}$ *secure, then Π is smooth.*

Proof. The proof of this lemma is almost the same as the proof in [8] that shows that an IND-CCA2 secure KEM is smooth. Some notations are borrowed from [8].

Assume towards a contradiction that there exists an IND-CPA secure PKE scheme $\Pi = (\mathsf{PKG}, \mathsf{PEnc}, \mathsf{PDec})$ whose plaintext space size $|\mathcal{M}_\Pi|$ is polynomially bounded, and the smoothness Smth_Π (as defined in Definition 3 in Section 2.2) is not negligible. Then we show that we can construct a PPTA adversary $\mathcal{B} = (\mathcal{B}_1, \mathcal{B}_2)$ that breaks IND-CPA security of Π with non-negligible advantage. The description of $\mathcal{B}$ is as follows.

$\mathcal{B}_1$: On input pk, $\mathcal{B}_1$ picks $m_1 \in \mathcal{M}_\Pi$ uniformly at random and also picks some m_0 satisfying $m_0 \neq m_1$. Then $\mathcal{B}_1$ sets $\mathsf{st}_\mathcal{B}$ that consists of all the values known to $\mathcal{B}_1$, and terminates with output $(m_0, m_1, \mathsf{st}_\mathcal{B})$.

$\mathcal{B}_2$: On input $(c^*, \mathsf{st}_\mathcal{B})$, $\mathcal{B}_2$ computes $c' \leftarrow \mathsf{PEnc}(pk, m_1)$. If $c' = c^*$ then $\mathcal{B}_2$ sets $b' \leftarrow 1$, otherwise $\mathcal{B}_2$ sets $b' \leftarrow \{0,1\}$ uniformly at random. Then $\mathcal{B}_2$ terminates with output b' as its guess.

Let b be the challenge bit of the IND-CPA experiment that $\mathcal{B}$ has to guess. According to our description of $\mathcal{B}$, it is easy to see that $\Pr[b' = 1 | c' = c^* \wedge b = 1] = 1$ and $\Pr[b' = 1 | c' \neq c^* \wedge b = 1] = \Pr[b' = 0 | c' \neq c^* \wedge b = 0] = \frac{1}{2}$. Moreover, $\Pr[c' = c^* | b = 0] = 0$ (and hence $\Pr[c' \neq c^* | b = 0] = 1$) also holds. This is because if $b = 0$ then c^* is an encryption of m_0 while c' is always an encryption of m_1, and thus c' and c^* never collides due to correctness of the PKE scheme Π.

Using the above, we estimate the IND-CPA advantage of the adversary $\mathcal{B}$.

$$
\begin{aligned}
&\mathsf{Adv}^{\text{IND-CPA}}_{\Pi,\mathcal{B}} = |\Pr[b' = b] - \frac{1}{2}| \\
&= \frac{1}{2}|\Pr[b' = 1 | b = 1] + \Pr[b' = 0 | b = 0] - 1| \\
&= \frac{1}{2}|\Pr[b' = 1 | c' = c^* \wedge b = 1] \cdot \Pr[c' = c^* | b = 1] + \Pr[b' = 1 | c' \neq c^* \wedge b = 1] \cdot \Pr[c' \neq c^* | b = 1] \\
&\quad + \Pr[b' = 0 | c' = c^* \wedge b = 0] \cdot \Pr[c' = c^* | b = 0] + \Pr[b' = 0 | c' \neq c^* \wedge b = 0] \cdot \Pr[c' \neq c^* | b = 0] - 1| \\
&= \frac{1}{2}|1 \cdot \Pr[c' = c^* | b = 1] + \frac{1}{2} \cdot \Pr[c' \neq c^* | b = 1] + 0 + \frac{1}{2} \cdot 1 - 1| \\
&= \frac{1}{4} \Pr[c' = c^* | b = 1]
\end{aligned}
$$

In order to proceed, we now need some notation. For any pk, and any pair (m, c) where $m \in \mathcal{M}_\Pi$ and $c \in \{0,1\}^*$, let

$$\nu(pk, m, c) = \Pr_{c' \leftarrow \mathsf{PEnc}(pk,m)}[c' = c].$$

Let $(m_{\max}(pk), c_{\max}(pk)) \in \mathcal{M}_\Pi \times \{0,1\}^*$ be an arbitrary pair such that $\nu(pk, m_{\max}(pk), c_{\max}(pk)) \geq \nu(pk, m, c)$ for any pair $(m, c) \in \mathcal{M}_\Pi \times \{0,1\}^*$.

Using the above, we can proceed as follows.

$$\begin{aligned}
\mathsf{Adv}^{\texttt{IND-CPA}}_{\Pi,\mathcal{B}} &= \frac{1}{4}\Pr[c' = c^* | b = 1] \\
&\geq \frac{1}{4}\Pr[c' = c^* \wedge m_1 = m_{\max}(pk) | b = 1] \\
&= \frac{1}{4}\Pr[c' = c^* | m_1 = m_{\max}(pk) \wedge b = 1] \cdot \Pr[m_1 = m_{\max}(pk) | b = 1] \\
&= \frac{1}{4|\mathcal{M}_\Pi|} \cdot \Pr[c' = c^* | m_1 = m_{\max}(pk) \wedge b = 1] \\
&\geq \frac{1}{4|\mathcal{M}_\Pi|} \cdot \Pr[c' = c_{\max}(pk) \wedge c^* = c_{\max}(pk) | m_1 = m_{\max}(pk) \wedge b = 1],
\end{aligned}$$

where we used $\Pr[m_1 = m_{\max}(pk) | b = 1] = \Pr[m_1 = m_{\max}(pk)] = \frac{1}{|\mathcal{M}_\Pi|}$, which is because $\mathcal{B}$ picks m_1 uniformly at random from $\mathcal{M}_\Pi$.

Here, recall that due to our description of $\mathcal{B}$, $\Pr[c' = c_{\max}(pk) \wedge c^* = c_{\max}(pk) | m_1 = m_{\max}(pk) \wedge b = 1]$ in the $\texttt{IND-CPA}$ experiment is equivalent to $\Pr[(pk, sk) \leftarrow \mathsf{PKG}; c^* \leftarrow \mathsf{PEnc}(pk, m_{\max}(pk)); c' \leftarrow \mathsf{PEnc}(pk, m_{\max}(pk)) : c' = c_{\max}(pk) \wedge c^* = c_{\max}(pk)]$.

Moreover, let

$$X(pk) = \Pr_{c \leftarrow \mathsf{PEnc}(pk, m_{\max}(pk))}[c = c_{\max}(pk)].$$

We will regard X as a random variable over the choice of pk, which is given by $(pk, sk) \leftarrow \mathsf{PKG}(1^k)$. By definition, we have $\mathbf{E}_{(pk,sk)\leftarrow\mathsf{PKG}(1^k)}[X(pk)] \geq \mathsf{Smth}_\Pi(k)$. (Below, we omit the security parameter k.)

Using these, we have

$$\begin{aligned}
&\Pr\left[\begin{array}{l}(pk, sk) \leftarrow \mathsf{PKG}; \\ c^* \leftarrow \mathsf{PEnc}(pk, m_{\max}(pk)); \\ c' \leftarrow \mathsf{PEnc}(pk, m_{\max}(pk))\end{array} : c' = c_{\max}(pk) \wedge c^* = c_{\max}(pk)\right] \\
&= \mathop{\mathbf{E}}_{(pk,sk)\leftarrow\mathsf{PKG}}\left[\Pr\left[\begin{array}{l} c^* \leftarrow \mathsf{PEnc}(pk, m_{\max}(pk)); \\ c' \leftarrow \mathsf{PEnc}(pk, m_{\max}(pk))\end{array} : c' = c_{\max}(pk) \wedge c^* = c_{\max}(pk)\right]\right] \\
&= \mathop{\mathbf{E}}_{(pk,sk)\leftarrow\mathsf{PKG}}\left[\ (\ \Pr[c \leftarrow \mathsf{PEnc}(pk, m_{\max}(pk)) : c = c_{\max}(pk)]\ )^2\ \right] \\
&= \mathop{\mathbf{E}}_{(pk,sk)\leftarrow\mathsf{PKG}}\left[X(pk)^2\right] \geq \left(\mathop{\mathbf{E}}_{(pk,sk)\leftarrow\mathsf{PKG}}[X(pk)]\right)^2 \geq (\mathsf{Smth}_\Pi)^2,
\end{aligned}$$

where in the first inequality we used the Jensen's inequality.

In summary, we have $\mathsf{Adv}^{\texttt{IND-CPA}}_{\Pi,\mathcal{B}} \geq \frac{1}{4|\mathcal{M}_\Pi|} \cdot (\mathsf{Smth}_\Pi)^2$, which is non-negligible because $|\mathcal{M}_\Pi|$ is polynomial and we made an assumption at the beginning of the proof of this lemma that Smth_Π is non-negligible. Since this contradicts the $\texttt{IND-CPA}$ security of Π, Smth_Π must be negligible. This completes the proof of Lemma 9. □

4.3 Relations among Security Notions for Mixed CCA Security

In this section, we investigate the relations among mixed CCA security notions.

Due to its generality (i.e. choices of the three query sequences $\mathsf{B}, \mathsf{F}, \mathsf{A} \in \mathcal{QS}^*$, the difference between single and parallel queries), given two mixed CCA security notions, it is not always easy to tell if one mixed CCA security notion implies the other. Therefore, a natural and yet non-trivial question is: *given two mixed CCA security notions* $\langle \mathsf{B} : \mathsf{F} : \mathsf{A} \rangle$-$\mathtt{mCCA}$ *and* $\langle \widetilde{\mathsf{B}} : \widetilde{\mathsf{F}} : \widetilde{\mathsf{A}} \rangle$-$\mathtt{mCCA}$*, under what conditions on* $\mathsf{B}, \mathsf{F}, \mathsf{A}, \widetilde{\mathsf{B}}, \widetilde{\mathsf{F}}, \widetilde{\mathsf{A}}$ *are there implications/separations?*

We fully answer this question and show a necessary and sufficient condition for implications/separations between any two mixed CCA security notions. Interestingly, it turns out that *for PKE schemes, the relations among security notions are different depending on its plaintext space size.* The relations among mixed CCA security notions for PKE schemes with polynomially bounded plaintext space size and those for the KEMs are always the same.

The rest of this section is organized as follows. In Section 4.3.1 we first introduce a relation over query sequences which plays a key role for our results. Then in Sections 4.3.3 and 4.3.4 we show separation results and implication results, resprctively, that lead to the necessary and sufficient condition, which will be shown in Section 4.3.5. For notational convenience, in the following we always assume $\mathsf{B}, \mathsf{F}, \mathsf{A}, \widetilde{\mathsf{B}}, \widetilde{\mathsf{F}}, \widetilde{\mathsf{A}} \in \mathcal{QS}^*$, and do not write it explicitly.

4.3.1 "is-Simulatable-by" Relation for Query Sequences

In order to state our results on relations among mixed CCA security notions, we need to introduce the "is-simulatable-by" relation for query sequences. But before defining it, we introduce the following relation over the symbols s and p.

Definition 23. *We define the partial order "*$\subseteq_1$*" over symbols* $\{\mathsf{s}, \mathsf{p}\}$ *by* $\mathsf{s} \subseteq_1 \mathsf{s}$*,* $\mathsf{s} \subseteq_1 \mathsf{p}$*, and* $\mathsf{p} \subseteq_1 \mathsf{p}$*.*

Intuitively, the meaning of "$\subseteq_1$" is that the former type oracle query "is-simulatable-by" the latter type of oracle query. For example, in the mixed CCA security experiment a single decryption query is simulatable if a single or parallel query can be made, and a parallel query is simulatable by a parallel query. The subscript "1" of "$\subseteq_1$" denotes that it is a relation for one symbol, and it should not be mixed up with the relation for query sequences below.

Now, we extend the "is-simulatable-by" relation to query sequences $\mathcal{QS}^*$, which we denote by "$\subseteq_{qs}$" (the subscript "qs" stands for *query sequence*).

Definition 24. *Let* $\mathsf{seq}, \widetilde{\mathsf{seq}} \in \mathcal{QS}^*$*. We define the binary relation "*$\subseteq_{qs}$*" over* $\mathcal{QS}^*$ *as follows. "*$\widetilde{\mathsf{seq}} \subseteq_{qs} \mathsf{seq}$*" if and only if one of the following is satisfied:*

- $\mathsf{seq} = \mathsf{unbound}$ *or* $\widetilde{\mathsf{seq}} = \emptyset$
- $\mathsf{seq} = (a_1 \dots a_m), \mathsf{seq} = (b_1 \dots b_n) \in \mathcal{QS} \backslash \{\emptyset\}$ *where* $a_i, b_j \in \{\mathsf{s}, \mathsf{p}\}$ *for each* $i \in [m]$*,* $j \in [n]$*, and there exists a strictly increasing function* $f : [n] \to [m]$ *such that* $b_j \subseteq_1 a_{f(j)}$ *holds for all* $j \in [n]$*.*

If seq *and* $\widetilde{\mathsf{seq}}$ *do not satisfy the above, we write "*$\widetilde{\mathsf{seq}} \not\subseteq_{qs} \mathsf{seq}$*".*

It is easy to see that the above relation "$\subseteq_{qs}$" is a natural extension from $\subseteq_1$ i.e. "is-simulatable-by" relation over $\{\mathsf{s}, \mathsf{p}\}$. Suppose $\widetilde{\mathsf{seq}} \subseteq_{qs} \mathsf{seq}$. Consider two adversaries $\mathcal{A}$ and $\mathcal{B}$ attacking a same PKE scheme, where $\mathcal{A}$ makes seq-queries and $\mathcal{B}$ makes $\widetilde{\mathsf{seq}}$-queries, and a situation in which $\mathcal{A}$ simulates the experiment for $\mathcal{B}$. If $\widetilde{\mathsf{seq}} = \emptyset$, then $\mathcal{B}$ makes no query. If $\mathsf{seq} = \mathsf{unbound}$, then $\mathcal{A}$ can use unbounded oracle access, and thus $\mathcal{B}$'s decryption oracle can be simulated. Otherwise, (i.e. $\mathsf{seq}, \widetilde{\mathsf{seq}} \in \mathcal{QS} \backslash \{\emptyset\}$), then i-th query from $\mathcal{B}$ can be simulated by

$\mathcal{A}$'s $f(i)$-th query (where f is a strictly increasing function guaranteed to exist by definition) for all $i \in [n]$.

Now, given any two query sequences $\mathsf{seq}, \widetilde{\mathsf{seq}} \in \mathcal{QS}^*$ we can tell if $\widetilde{\mathsf{seq}} \subseteq_{qs} \mathsf{seq}$ or $\widetilde{\mathsf{seq}} \not\subseteq_{qs} \mathsf{seq}$.[3] For example, if $\mathsf{seq} = \mathsf{s}^q$ and $\widetilde{\mathsf{seq}} = \mathsf{s}^r$, then $\widetilde{\mathsf{seq}} \subseteq_{qs} \mathsf{seq}$ if and only if $q \geq r$. If $\mathsf{seq} = (\mathsf{psps})$ and $\widetilde{\mathsf{seq}} = (\mathsf{ssp})$ then $\widetilde{\mathsf{seq}} \subseteq_{qs} \mathsf{seq}$, while if $\mathsf{seq} = (\mathsf{sspss})$ and $\widetilde{\mathsf{seq}} = (\mathsf{spp})$ then $\widetilde{\mathsf{seq}} \not\subseteq_{qs} \mathsf{seq}$.

We stress that our definition of "$\subseteq_{qs}$" (and "$\not\subseteq_{qs}$") is different from the notion of "subset" ("$\subseteq$"), because the "ordering" in sequences matters.

4.3.2 Useful Tool for Separation: Backdoor-Sequence Scheme

A common approach for showing a separation of a security notion $\mathtt{X}$ from a security notion $\mathtt{Y}$ for PKE schemes is to implement some "backdoor" mechanism, which leads to some "critical information" v for breaking $\mathtt{Y}$ security, into a decryption algorithm (and possibly into other algorithms) of an $\mathtt{X}$-secure PKE scheme, so that $\mathtt{Y}$-adversary can, by using a decryption oracle, reach for v and break $\mathtt{Y}$-security of the scheme, while an $\mathtt{X}$-adversary cannot reach for v or simply v is useless for breaking $\mathtt{X}$-security of the scheme. We also follow this approach.

We wish to implement a backdoor mechanism so that given two sequences $\mathsf{seq}, \widetilde{\mathsf{seq}}$ satisfying $\widetilde{\mathsf{seq}} \not\subseteq_{qs} \mathsf{seq}$, the mechanism exploits the essential difference between the information available for an adversary making $\widetilde{\mathsf{seq}}$-queries and that for an adversary making seq-queries. Basically, we implement such backdoor mechanism as a sequence of backdoor information $(u_1, \ldots, u_{|\widetilde{\mathsf{seq}}|+1})$ and a strategy for "how to release next backdoor information", based on $\mathsf{seq}, \widetilde{\mathsf{seq}} \in \mathcal{QS}$ and the critical information v. Specifically, let $\widetilde{\mathsf{seq}} = (b_1 \ldots b_n)$ such that $b_i \in \{\mathsf{s}, \mathsf{p}\}$ for $i \in [n]$.

- The sequence of backdoor information $(u_1, \ldots, u_{n+1})$ is set up so that $u_1 = 1^k$ (any publicly known value will do), $u_2, \ldots, u_n$ are random values, and u_{n+1} is the critical information v.

- The strategy for "how to release next backdoor information", depending on $\widetilde{\mathsf{seq}} = (b_1 \ldots b_n)$, is set up so that: If $b_i = \mathsf{s}$, then this "release" strategy on input u_i outputs u_{i+1} itself; If $b_i = \mathsf{p}$, then this "release" strategy on input u_i outputs a "secret-share" of u_{i+1}, so that if we collect the shares more than a threshold which is set to be a value greater than $|\mathsf{seq}|$, we can reconstruct u_{i+1}.

Constructed like this, an adversary making $\widetilde{\mathsf{seq}}$-queries to the release strategy can finally obtain u_{n+1} which is the critical information v. In particular, if $b_i = \mathsf{p}$ then an adversary can make a parallel query to the release strategy to obtain all the share of u_{i+1} at once, and thus can reconstruct u_{i+1}. A key point is that if $\widetilde{\mathsf{seq}} \not\subseteq_{qs} \mathsf{seq}$, then we can show that no adversary who is only allowed to make seq-queries can reach for $u_{n+1} = v$, and thus we can make a difference in the information available for an adversary making $\widetilde{\mathsf{seq}}$-queries and that making seq-queries.

In order to make it easier to analyze PKE schemes used to show separations, we formalize this "backdoor mechanism" as a "stand alone" primitive, independently of decryption algorithms of PKE schemes. We name it a *backdoor-sequence scheme*, and use it as a key tool for establishing the separations.

[3] Note that "$\subseteq_{qs}$" forms a partial order over $\mathcal{QS}^*$. However, it is not a total order. For example, if $\mathsf{seq}_1 = (\mathsf{sp})$ and $\mathsf{seq}_2 = (\mathsf{ps})$, then we have both $\mathsf{seq}_1 \not\subseteq_{qs} \mathsf{seq}_2$ and $\mathsf{seq}_2 \not\subseteq_{qs} \mathsf{seq}_1$.

Algorithms of Backdoor-Sequence Schemes

Formally, a backdoor-sequence scheme consists of the following three PPTA algorithms (BSGen, $\mathsf{Release}$, Recon). As a common input, these algorithms take a public parameter pub consisting of 1^k, a query sequence $\mathsf{seq} \in \mathcal{QS}\backslash\{\emptyset, \mathsf{unbound}\}$, and an integer $q \in \mathbb{N}$. Let $\mathsf{seq} = (b_1 \dots b_n)$ where $b_i \in \{\mathsf{s}, \mathsf{p}\}$ for each $i \in [n]$.

BSGen**:** A "backdoor-sequence generation" algorithm which takes pub and a secret value v as input. It then internally generates a sequence of backdoors $(u_1, \dots, u_{n+1}) \in (\{0,1\}^k)^{n+1}$ and auxiliary information aux. Finally it outputs a private parameter $\mathsf{pri} = (u_1, \dots, u_{n+1}, \mathsf{aux})$.

$\mathsf{Release}$**:** A deterministic "release" algorithm which takes pub, pri, two indices $\alpha \in [n]$ and $\beta \in [q]$, and a string $w \in \{0,1\}^k$ as input, and outputs some value $y \in \{0,1\}^k \cup \{\perp\}$.

Recon**:** A deterministic "reconstruction" algorithm which pub and a set of strings $(y_1, \dots, y_q) \in (\{0,1\}^k)^q$ as input, and outputs some value $z \in \{0,1\}^k \cup \{\perp\}$.

As a correctness requirement for a backdoor-sequence algorithm $BS = (\mathsf{BSGen}, \mathsf{Release}, \mathsf{Recon})$, for all $k \in \mathbb{N}$, all query sequences $\mathsf{seq} = (b_1 \dots b_n) \in \mathcal{QS}\backslash\{\emptyset\}$ (where each $b_i \in \{\mathsf{s}, \mathsf{p}\}$), all integers $q \in \mathbb{N}$, all values $v \in \{0,1\}^k$, and all $\mathsf{pri} \leftarrow \mathsf{BSGen}(\mathsf{pub} = (1^k, \mathsf{seq}, q), v)$, we require the following:

1. $u_1 = 1^k$ and $u_{n+1} = v$
2. For all $1 \le i \le n$:
   - If $b_i = \mathsf{s}$, then $\mathsf{Release}(\mathsf{pub}, \mathsf{pri}, i, 1, u_i) = u_{i+1}$
   - If $b_i = \mathsf{p}$ and $\mathsf{Release}(\mathsf{pub}, \mathsf{pri}, i, j, u_i) = y_i$ for all $j \in [q]$, then $\mathsf{Recon}(1^k, \mathsf{seq}, q, (y_1, \dots, y_q)) = u_{i+1}$
3. If $\alpha \notin [n]$ or $\beta \notin [q]$, then $\mathsf{Release}(\mathsf{pub}, \mathsf{pri}, \alpha, \beta, w) = \perp$ for any $w \in \{0,1\}^k$

Security Requirement of Backdoor-Sequence Scheme

We consider the security of a backdoor-sequence scheme. We would like the security property of a backdoor-sequence scheme to ensure that if $\widetilde{\mathsf{seq}} \not\subseteq_{qs} \mathsf{seq}$, then no PPTA adversary who issues "release" queries following the sequence seq can reach for the secret value v, while any adversary who can issue "release" queries following the sequence $\widetilde{\mathsf{seq}}$ can reach for v. To capture this, we consider the following $(\mathsf{seq}, \widetilde{\mathsf{seq}})$-backdoor-sequence experiment that an adversary $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$ runs in. Let $\widetilde{\mathsf{seq}} = (b_1 \dots b_n) \in \mathcal{QS}\backslash\{\emptyset\}$ where $b_i \in \{\mathsf{s}, \mathsf{p}\}$ for each $i \in [n]$, and let $q = |\mathsf{seq}| + 1$.

$$\mathsf{Expt}^{(\mathsf{seq},\widetilde{\mathsf{seq}})}_{BS,\mathcal{A}}(k) : \mathsf{pub} \leftarrow (1^k, \widetilde{\mathsf{seq}}, q); (v_0^*, v_1^*, \mathsf{st}_{\mathcal{A}}) \leftarrow \mathcal{A}_1(\mathsf{pub}); b \leftarrow \{0,1\};$$
$$\mathsf{pri} \leftarrow \mathsf{BSGen}(\mathsf{pub}, v_b^*); b' \leftarrow \mathcal{A}_2^{\mathcal{O}}(\mathsf{st}_{\mathcal{A}}); \ \texttt{If } b' = b \texttt{ then return 1 else return 0}$$

Here, we define the oracle $\mathcal{O}$ which is given to $\mathcal{A}_2$ by $\mathcal{O}(\alpha, \beta, w) = \mathsf{Release}(\mathsf{pub}, \mathsf{pri}, \alpha, \beta, w)$, where $\alpha \in [n]$, $\beta \in [q]$, and $w \in \{0,1\}^k$. However, this oracle is available according to the query sequence $\mathsf{seq} = (b_1 \dots b_n)$. That is, if $b_i = \mathsf{s}$ then $\mathcal{A}_2$ can issue a single query of the form (α_i, β_i, w_i) as i-th query and is given $\mathsf{Release}(\mathsf{pub}, \mathsf{pri}, \alpha_i, \beta_i, w_i)$ as an answer, and if $b_i = \mathsf{p}$ then

Common Input: $\mathsf{pub} = (1^k, \mathsf{seq}, q)$ where $\mathsf{seq} = (b_1 \dots b_n) \in \mathcal{QS}\backslash\{\emptyset\}$ (i.e., each $b_i \in \{\mathsf{s}, \mathsf{p}\}$), and $q \in \mathbb{N}$.

$\mathsf{BSGen}(\mathsf{pub}, v)$ where $v \in \{0,1\}^k$:
Set $u_1 \leftarrow 1^k$ and $u_{n+1} \leftarrow v$.
If $n \geq 2$ then pick $u_2, \dots, u_n \in \{0,1\}^k$ uniformly at random.
For $1 \leq i \leq n$ Do:
If $b_i = \mathsf{s}$ then
Set $p_{i,1}(x) \leftarrow u_{i+1}$ and $p_{i,j} \leftarrow \perp$ for $2 \leq j \leq q$.
Else ($b_i = \mathsf{p}$)
Pick randomly $p_{i,1}, \dots, p_{i,q} \in \{0,1\}^k$ such that $u_{i+1} = \bigoplus_{j=1}^{q} p_{i,j}$.
End For
Set $\mathsf{aux} \leftarrow \{p_{i,j}\}_{i\in[n],j\in[q]}$.
Return $\mathsf{pri} = (u_1, \dots, u_n, v, \mathsf{aux})$.

$\mathsf{Release}(\mathsf{pub}, \mathsf{pri}, \alpha, \beta, w)$ where $\alpha \in [n]$, $\beta \in [q]$, and $w \in \{0,1\}^k$: (If $\alpha \notin [n]$ or $\beta \notin [q]$ then return $\perp$) Parse pri as $(u_1, \dots, u_{n+1}, \mathsf{aux})$ and then parse aux as $\{p_{i,j}\}_{i\in[n],j\in[q]}$. If $u_\alpha = w$ then return $y \leftarrow p_{\alpha,\beta}$, else return $\perp$.
$\mathsf{Recon}(\mathsf{pub}, (y_1, \dots, y_q))$ where each $y_i \in \{0,1\}^k$ Return $z \leftarrow \bigoplus_{j=1}^{q} y_i$.

Figure 4.1: A concrete instantiation of a backdoor-sequence scheme BS.

$\mathcal{A}_2$ can issue a parallel query of the form $((\alpha_{i,1}, \beta_{i,1}, w_{i,1}), (\alpha_{i,2}, \beta_{i,2}, w_{i,2}), \dots, (\alpha_{i,\ell}, \beta_{i,\ell}, w_{i,\ell}), \dots)$ as i-th query and is given $(y_1, y_2, \dots, y_\ell, \dots)$, where $y_\ell = \mathsf{Release}(\mathsf{pub}, \mathsf{pri}, \alpha_{i,\ell}, \beta_{i,\ell}, w_{i,\ell})$ for every $\ell \in \mathbb{N}$, as an answer. In the parallel query, the number of inputs to the oracle is unbounded and thus can be dependent only on the adversary $\mathcal{A}$.

We define the advantage of an adversary $\mathcal{A}$ in the $(\mathsf{seq}, \widetilde{\mathsf{seq}})$-backdoor-sequence experiment as:

$$\mathsf{Adv}^{(\mathsf{seq},\widetilde{\mathsf{seq}})}_{BS,\mathcal{A}}(k) = |\Pr[\mathsf{Expt}^{(\mathsf{seq},\widetilde{\mathsf{seq}})}_{BS,\mathcal{A}}(k) = 1] - \frac{1}{2}|$$

Definition 25. *Let* $\mathsf{seq} \in \mathcal{QS}$ *and* $\widetilde{\mathsf{seq}} \in \mathcal{QS}\backslash\{\emptyset\}$. *We say that a backdoor-sequence scheme is* $(\mathsf{seq}, \widetilde{\mathsf{seq}})$*-secure if* $\mathsf{Adv}^{(\mathsf{seq},\widetilde{\mathsf{seq}})}_{BS,\mathcal{A}}(k)$ *is negligible for any PPTA* $\mathcal{A}$. *Furthermore, we say that a backdoor sequence scheme* BS *is secure if it is* $(\mathsf{seq}, \widetilde{\mathsf{seq}})$*-secure for any* $\mathsf{seq}, \widetilde{\mathsf{seq}} \in \mathcal{QS}$ *satisfying* $\widetilde{\mathsf{seq}} \not\sqsubseteq_{qs} \mathsf{seq}$.

Concrete Instantiation

We concretely instantiate a backdoor-sequence scheme $BS = (\mathsf{BSGen}, \mathsf{Release}, \mathsf{Recon})$ as in Fig. 4.1, which is based on a secrete sharing. It is straightforward to see that our scheme BS satisfies the correctness requirement.

Now, we prove the security of BS.

Lemma 10. *The backdoor-sequence scheme* BS *in Fig. 4.1 is secure.*

Proof. Fix $\mathsf{seq} \in \mathcal{QS}$ and $\widetilde{\mathsf{seq}} \in \mathcal{QS}\backslash\{\emptyset\}$ satisfying $\widetilde{\mathsf{seq}} \not\sqsubseteq_{qs} \mathsf{seq}$. Let $\widetilde{\mathsf{seq}} = (b_1 b_2 \dots, b_n)$ where $b_i \in \{\mathsf{s}, \mathsf{p}\}$ for each $i \in [n]$, and let $q = |\mathsf{seq}| + 1$. (Here, $\widetilde{\mathsf{seq}} \neq \emptyset$ guarantees that such $n \geq 1$ exists.)

Let $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$ be any PPTA adversary which runs in the $(\mathsf{seq}, \widetilde{\mathsf{seq}})$-backdoor-sequence experiment regarding BS.

The key observation for proving this lemma is that due to our design of the backdoor sequence scheme, the only way that the adversary $\mathcal{A}$ can learn the information on b is limited to the following cases (which differ depending on b_n):

- If $b_n = \mathsf{s}$, then $\mathcal{A}_2$ has to make a query of the form $(\alpha, \beta, w) = (n, 1, u_n)$.
- If $b_p = \mathsf{p}$, then since $u_{n+1} = v_b^*$ is divided into q shares with a q-out-of-q secret sharing with perfect secrecy, $\mathcal{A}_2$ has to make a query of the form $(\alpha, \beta, w) = (n, j, u_n)$ for every $1 \leq j \leq q$, and reconstruct u_{n+1} from the shares returned from the oracle. (This can be done in one parallel query or in the combination of some single and/or parallel queries.)

If $\mathcal{A}$ fails the above, the information on b is information-theoretically hidden from $\mathcal{A}$'s view point. We will use this observation later.

Let us introduce some notation. For a while, we assume $n \geq 2$. We consider the "splitting" of seq according to $\widetilde{\mathsf{seq}}$ so that $\mathsf{seq} = (\mathsf{seq}_1 || \ldots || \mathsf{seq}_{j^*-1} || \mathsf{seq}_{j^*})$ for some integer $j^* \geq 2$, and

- (1) for $1 \leq i \leq j^* - 1$, if $b_i = \mathsf{s}$ then $\mathsf{seq}_i \in \{\mathsf{s}, \mathsf{p}\}$, and if $b_i = \mathsf{p}$ then $\mathsf{seq}_i = (\mathsf{s}^{q_i}\mathsf{p})$ for some $0 \leq q_i < q$, and
- (2) if $b_{j^*} = \mathsf{s}$ then $\mathsf{seq}_{j^*} = \emptyset$, and if $b_{j^*} = \mathsf{p}$ then $\mathsf{seq}_{j^*} = (\mathsf{s}^{q_{j^*}})$ for some $0 \leq q_{j^*} < q$.

We call j^* the *separating index*.

For example, if $\mathsf{seq} = (\mathsf{pssps})$ and $\widetilde{\mathsf{seq}} = (\mathsf{spps})$, then we have $\mathsf{seq}_1 = \mathsf{p}$, $\mathsf{seq}_2 = (\mathsf{ssp})$, and $\mathsf{seq}_3 = \mathsf{s}$. Note that $(\mathsf{seq}_1 || \mathsf{seq}_2 || \mathsf{seq}_3) = \mathsf{seq}$. Therefore, in this example, $j^* = 3$.

If we split seq as above, we have $b_i \subseteq_{qs} \mathsf{seq}_i$ for $1 \leq i \leq j^* - 1$ and $b_{j^*} \not\subseteq_{qs} \mathsf{seq}_{j^*}$. We note that if $\mathsf{seq}' \not\subseteq_{qs} \mathsf{seq}$ and $n \geq 2$, then there must exist such a position $j^* \in \{2, \ldots, n\}$. This is because if no such j^* exists, i.e., if we have $b_i \subseteq_{qs} \mathsf{seq}_i$ for all $i \in [n]$, then we must have $\widetilde{\mathsf{seq}} \subseteq_{qs} \mathsf{seq}$.

We refer to the queries following seq_i as seq_i-queries. By the definition of $(\mathsf{seq}, \widetilde{\mathsf{seq}})$-backdoor-sequence experiment, an adversary can make seq_i-queries in the game only after it has completed $(\mathsf{seq}_1 || \ldots || \mathsf{seq}_{i-1})$-queries.

For convenience, we define the separating index j^* for the case $n = 1$ by $j^* = 1$. This guarantees that $b_{j^*} = b_1 = \widetilde{\mathsf{seq}} \not\subseteq_{qs} \mathsf{seq} = \mathsf{seq}_j^*$.[4]

Let Succ denote the event that $\mathcal{A}$ succeeds in guessing the bit b (i.e. $b' = b$ occurs). For $i \in \{1, \ldots, j^*\}$, let Jump_i denote the event that until the point $\mathcal{A}$ completes seq_i-queries, $\mathcal{A}$ has made at least one query that contains an input (α, β, w) such that $w = u_\alpha$ holds for some $\alpha \in \{i + 1, \ldots, n\}$. For convenience, we define the event Jump_i also for $i = 0$ and $i > j^*$ as follows: Jump_0 is set to be always false (and thus $\overline{\mathsf{Jump}_0}$ is set to be always true).; the truth-value of Jump_i for $i > j^*$ to be that of Jump_{j^*}, and thus for $i > j^*$ and for any event E defined in the $(\mathsf{seq}, \widetilde{\mathsf{seq}})$-backdoor-sequence experiment, $\Pr[\mathsf{Jump}_i] = \Pr[\mathsf{Jump}_{j^*}]$, $\Pr[\mathsf{E}|\mathsf{Jump}_i] = \Pr[\mathsf{E}|\mathsf{Jump}_{j^*}]$, and $\Pr[\mathsf{E}|\overline{\mathsf{Jump}_i}] = \Pr[\mathsf{E}|\overline{\mathsf{Jump}_{j^*}}]$.

We will use the above notations to analyze the advantage of the adversary $\mathcal{A}$ in the $(\mathsf{seq}, \widetilde{\mathsf{seq}})$-backdoor-sequence experiment. In order to upperbound the advantage of $\mathcal{A}$, we use the following two claims, which will be proven later:

Claim 3. $\Pr[\mathsf{Succ}|\overline{\mathsf{Jump}_{n-1}}] = \frac{1}{2}$

[4]Note that if $\widetilde{\mathsf{seq}} \not\subseteq_{qs} \mathsf{seq}$ and $|\widetilde{\mathsf{seq}}| = n = 1$ then $\widetilde{\mathsf{seq}} = b_1 \in \{\mathsf{s}, \mathsf{p}\}$. If $\widetilde{\mathsf{seq}} = \mathsf{s}$ then seq must be $\emptyset$. If $\widetilde{\mathsf{seq}} = \mathsf{p}$ then seq must be of the form s^q for some integer $q \geq 0$.

Claim 4. $\Pr[\mathsf{Jump}_{n-1}]$ *is negligible.*

Using these, the advantage is shown to be negligible as follows:

$$\begin{aligned}
\mathsf{Adv}^{(\mathsf{seq},\widetilde{\mathsf{seq}})}_{BS,\mathcal{A}} &= |\Pr[\mathsf{Succ}] - \frac{1}{2}| \\
&= |\Pr[\mathsf{Succ}|\mathsf{Jump}_{n-1}] \cdot \Pr[\mathsf{Jump}_{n-1}] + \Pr[\mathsf{Succ}|\overline{\mathsf{Jump}_{n-1}}] \cdot \Pr[\overline{\mathsf{Jump}_{n-1}}] - \frac{1}{2}| \\
&= |\Pr[\mathsf{Succ}|\mathsf{Jump}_{n-1}] \cdot \Pr[\mathsf{Jump}_{n-1}] + \frac{1}{2}\Pr[\overline{\mathsf{Jump}_{n-1}}] - \frac{1}{2}| \\
&= |\Pr[\mathsf{Succ}|\mathsf{Jump}_{n-1}] \cdot \Pr[\mathsf{Jump}_{n-1}] - \frac{1}{2}\Pr[\mathsf{Jump}_{n-1}]| \\
&= |\Pr[\mathsf{Succ}|\mathsf{Jump}_{n-1}] - \frac{1}{2}| \cdot \Pr[\mathsf{Jump}_{n-1}] \\
&\leq \frac{1}{2}\Pr[\mathsf{Jump}_{n-1}] \qquad (4.1)
\end{aligned}$$

Therefore, it remains to prove Claims 3 and 4.

Proof of Claim 3. Recall that $\mathcal{A}$ can make seq-queries in the $(\mathsf{seq}, \widetilde{\mathsf{seq}})$-backdoor sequence experiment, and we have split seq as $\mathsf{seq} = (\mathsf{seq}_1||\ldots||\mathsf{seq}_{j^*})$. We consider two cases separately: Case (i) $j^* < n$ and Case (ii) $j^* = n$:

Case (i) $j^* < n$**:** $j^* < n$ implies $j^* \leq n-1$, and thus $\Pr[\mathsf{Succ}|\overline{\mathsf{Jump}_{n-1}}] = \Pr[\mathsf{Succ}|\overline{\mathsf{Jump}_{j^*}}]$. Note that if $\overline{\mathsf{Jump}_{j^*}}$ occurs, then $\mathcal{A}$ has not made any query that contains an input (α, β, w) satisfying $w = u_\alpha$ for some $\alpha \in \{j^*+1, \ldots, n\}$. Under this situation, since after seq_{j^*}-queries $\mathcal{A}$ can issue no further query, by our key observation, the information on b is information-theoretically hidden from $\mathcal{A}$'s view point. This implies $\Pr[\mathsf{Succ}|\overline{\mathsf{Jump}_{j^*}}] = \Pr[\mathsf{Succ}|\overline{\mathsf{Jump}_{n-1}}] = \frac{1}{2}$.

Case (ii) $j^* = n$**:** Recall that if $\overline{\mathsf{Jump}_{n-1}}$ has occurred, then, after the point $\mathcal{A}_2$ has completed the seq_{n-1}-queries, $\mathcal{A}_2$ has issued no query which contains the input of the form $(\alpha, \beta, w) = (n, j, u_n)$. If $b_n = \mathsf{s}$, then, according to our splitting of seq, $\mathsf{seq}_n = \emptyset$. Thus, if $b_n = \mathsf{s}$ then $\mathcal{A}_2$ cannot make any further query to the oracle $\mathcal{O}$. In this case, the information on b is information-theoretically hidden from $\mathcal{A}$'s view point. On the other hand, if $b_n = \mathsf{p}$, then we have $\mathsf{seq}_n = (\mathsf{s}^{q_n})$ for some $0 \leq q_n < q$. That is, $\mathcal{A}$ can make single queries to $\mathcal{O}$ less than q times. By our key observation above, even if $\mathcal{A}_2$ makes further single queries less than q times after the point $\mathcal{A}$ has completed seq_{n-1}-queries, the information on b is again information-theoretically hidden from $\mathcal{A}$. Therefore, regardless of whether $b_n = \mathsf{s}$ or $b_n = \mathsf{p}$, the information of b is information-theoretically hidden from $\mathcal{A}$, which implies $\Pr[\mathsf{Succ}|\overline{\mathsf{Jump}_{n-1}}] = \frac{1}{2}$.

The above completes the proof of Claim 3. □

Proof of Claim 4. The following holds for $1 \leq i \leq n-1$,

$$\begin{aligned}
\Pr[\mathsf{Jump}_i] &= \Pr[\mathsf{Jump}_i \wedge \mathsf{Jump}_{i-1}] + \Pr[\mathsf{Jump}_i \wedge \overline{\mathsf{Jump}_{i-1}}] \\
&= \Pr[\mathsf{Jump}_i|\mathsf{Jump}_{i-1}] \cdot \Pr[\mathsf{Jump}_{i-1}] + \Pr[\mathsf{Jump}_i|\overline{\mathsf{Jump}_{i-1}}] \cdot \Pr[\overline{\mathsf{Jump}_{i-1}}] \\
&\leq \Pr[\mathsf{Jump}_{i-1}] + \Pr[\mathsf{Jump}_i|\overline{\mathsf{Jump}_{i-1}}]
\end{aligned}$$

Thus we have

$$\Pr[\mathsf{Jump}_i] - \Pr[\mathsf{Jump}_{i-1}] \leq \Pr[\mathsf{Jump}_i | \overline{\mathsf{Jump}_{i-1}}]$$

Then, taking the summation regarding i, we have

$$\sum_{i=1}^{n-1}(\Pr[\mathsf{Jump}_i] - \Pr[\mathsf{Jump}_{i-1}]) = \Pr[\mathsf{Jump}_{n-1}] \leq \sum_{i=1}^{n-1} \Pr[\mathsf{Jump}_i | \overline{\mathsf{Jump}_{i-1}}]$$

where we used $\Pr[\mathsf{Jump}_0] = 0$. Therefore,

$$\Pr[\mathsf{Jump}_{n-1}] \leq \sum_{i=1}^{n-1} \Pr[\mathsf{Jump}_i | \overline{\mathsf{Jump}_{i-1}}] \tag{4.2}$$

It remains to show the upperbound of each $\Pr[\mathsf{Jump}_i | \overline{\mathsf{Jump}_{i-1}}]$. Let Q_{max} be the maximum number of inputs in a parallel query. Since $\mathcal{A}$ is a PPTA, Q_{max} is some polynomial (this value can be dependent only on the adversary $\mathcal{A}$). Without loss of generality we assume $Q_{max} \geq q$. Let $(u_1, \ldots, u_{n+1})$ be the sequence of backdoors generated from BSGen algorithm in the $(\mathsf{seq}, \widetilde{\mathsf{seq}})$-backdoor-sequence experiment. For $0 \leq i < n$ we define $\mathcal{U}_i = \{u_{i+1}, \ldots, u_n\}$. (For example, $\mathcal{U}_1 = \{u_2, \ldots, u_n\}$, and $\mathcal{U}_{n-1} = \{u_n\}$.)

Recall that $[\overline{\mathsf{Jump}_i} | \overline{\mathsf{Jump}_{i-1}}]$ is the event that, given that none of $\mathcal{A}_2$'s queries previous to seq_{i-1}-queries contains an input (α, β, w) (for $\mathcal{O}$) satisfying $w \in \mathcal{U}_{i-1}$, none of the inputs (α, β, w) contained in $\mathcal{A}_2$'s seq_i-queries satisfies $w \in \mathcal{U}_i$. Recall that $\mathsf{seq}_i \in \{\mathsf{s}, \mathsf{p}\}$ if $b_i = \mathsf{s}$, or $\mathsf{seq}_i = (s^{q_i}\mathsf{p})$ with some $q_j < q$ if $b_i = \mathsf{p}$, and we are assuming $Q_{max} \geq q$. Therefore, seq_i-queries may contain at most $q + Q_{max} \leq 2Q_{max}$ inputs for $\mathcal{O}$. At the point when $\mathcal{A}$ receives the answer to seq_{i-1}-queries, $\mathcal{A}$ may know at most $(i-1) \cdot 2Q_{max}$ k-bit strings w that does not satisfy $w \in \mathcal{U}_i$. Since the size of $\mathcal{U}_i$ (i.e. the number of u_α's unknown to $\mathcal{A}_2$ except u_{n+1}) is $n - i$, we have

$$\begin{aligned}\Pr[\overline{\mathsf{Jump}_i} | \overline{\mathsf{Jump}_{i-1}}] \geq \prod_{j=1}^{2Q_{max}} \left(1 - \frac{n-i}{2^k - 2(i-1)Q_{max} - j + 1}\right) &\geq \left(1 - \frac{n-i}{2^k - 2iQ_{max}}\right)^{2Q_{max}} \\ &\geq 1 - \frac{(n-i) \cdot 2Q_{max}}{2^k - 2iQ_{max}} \geq 1 - \frac{2nQ_{max}}{2^k},\end{aligned}$$

where the last inequality is obtained by considering sufficiently large security parameter k. Therefore we have

$$\Pr[\mathsf{Jump}_i | \overline{\mathsf{Jump}_{i-1}}] = 1 - \Pr[\overline{\mathsf{Jump}_i} | \overline{\mathsf{Jump}_{i-1}}] \leq \frac{2nQ_{max}}{2^k},$$

for sufficiently large k.

Using the above in inequality 4.2, for sufficiently large k, we have:

$$\Pr[\mathsf{Jump}_{n-1}] \leq \sum_{i=1}^{n-1} \Pr[\mathsf{Jump}_i | \overline{\mathsf{Jump}_{i-1}}] \leq \sum_{i=1}^{n-1} \left(\frac{2nQ_{max}}{2^k}\right) = \frac{2n(n-1)Q_{max}}{2^k}$$

which is negligible. This completes the proof of Claim 4. □

The inequation (4.1) and Claims 3 and 4 imply that the backdoor-sequence scheme BS is $(\mathsf{seq}, \widetilde{\mathsf{seq}})$-secure. Note that the above proof works for any $\mathsf{seq}, \widetilde{\mathsf{seq}} \in \mathcal{QS}$ satisfying $\widetilde{\mathsf{seq}} \not\subseteq_{qs} \mathsf{seq}$. This completes the proof of Lemma 10. □

$\mathsf{PKG}_{\mathtt{sep1}}(1^k)$:	$\mathsf{PEnc}_{\mathtt{sep1}}(PK, m)$:
$(pk, sk) \leftarrow \mathsf{PKG}(1^k)$ $q \leftarrow \lvert(\mathsf{B}\lvert\lvert\mathsf{F}\lvert\lvert\mathsf{A})\rvert + 1$ $\mathsf{pub} \leftarrow (1^k, (\widetilde{\mathsf{B}}\lvert\lvert\widetilde{\mathsf{F}}\lvert\lvert\widetilde{\mathsf{A}}), q)$ $\mathsf{pri} \leftarrow \mathsf{BSGen}(\mathsf{pub}, sk)$ $PK \leftarrow (pk, \mathsf{pub})$ $SK \leftarrow (sk, \mathsf{pri})$ Return (PK, SK).	Parse PK as (pk, pub). $c \leftarrow \mathsf{PEnc}(pk, m)$ Return $C \leftarrow (0^k\lvert\lvert 0^k\lvert\lvert c)$.
$\mathsf{PDec}_{\mathtt{sep1}}(SK, C)$: Parse SK as (sk, pri) and C as $(\alpha\lvert\lvert\beta\lvert\lvert c)$ such that $\lvert\alpha\rvert = \lvert\beta\rvert = k$. If $(\alpha\lvert\lvert\beta) = (0^k\lvert\lvert 0^k)$ then return $\mathsf{PDec}(sk, c)$. Return $\mathsf{Release}(\mathsf{pub}, \mathsf{pri}, \alpha, \beta, c)$ (If c is longer than k-bit, then use the k-most significant bits of c.)	

Figure 4.2: The PKE scheme $\Pi_{\mathtt{sep1}}$ that separates $\langle\widetilde{\mathsf{B}} : \widetilde{\mathsf{F}} : \widetilde{\mathsf{A}}\rangle$-mCCA from $\langle\mathsf{B} : \mathsf{F} : \mathsf{A}\rangle$-mCCA in case $(\widetilde{\mathsf{B}}\lvert\lvert\widetilde{\mathsf{F}}\lvert\lvert\widetilde{\mathsf{A}}) \not\preceq_{qs} (\mathsf{B}\lvert\lvert\mathsf{F}\lvert\lvert\mathsf{A})$.

4.3.3 Separation Results

Here, we show the separations among mixed CCA security notions.

Firstly, by focusing on the difference in the total query sequences, we show the following separation.

Theorem 11. *For both PKE schemes and KEMs, if* $(\widetilde{\mathsf{B}}\lvert\lvert\widetilde{\mathsf{F}}\lvert\lvert\widetilde{\mathsf{A}}) \not\preceq_{qs} (\mathsf{B}\lvert\lvert\mathsf{F}\lvert\lvert\mathsf{A})$*, then* $\langle\mathsf{B} : \mathsf{F} : \mathsf{A}\rangle$-mCCA *security does not imply* $\langle\widetilde{\mathsf{B}} : \widetilde{\mathsf{F}} : \widetilde{\mathsf{A}}\rangle$-mCCA *security.*

Intuition. The idea for building the separating PKE scheme is straightforward. We use the secret key sk for the underlying PKE as a critical information together with a backdoor-sequence scheme. Then, a $\langle\widetilde{\mathsf{B}} : \widetilde{\mathsf{F}} : \widetilde{\mathsf{A}}\rangle$-mCCA adversary who can (in total) make $(\widetilde{\mathsf{B}}\lvert\lvert\widetilde{\mathsf{F}}\lvert\lvert\widetilde{\mathsf{A}})$-queries can reach for sk and decrypt the challenge ciphertext, while since $(\widetilde{\mathsf{B}}\lvert\lvert\widetilde{\mathsf{F}}\lvert\lvert\widetilde{\mathsf{A}}) \not\preceq_{qs} (\mathsf{B}\lvert\lvert\mathsf{F}\lvert\lvert\mathsf{A})$, a $\langle\mathsf{B} : \mathsf{F} : \mathsf{A}\rangle$-mCCA adversary who is only allowed to make $(\mathsf{B}\lvert\lvert\mathsf{F}\lvert\lvert\mathsf{A})$-queries in total cannot reach for it.

Proof. Since the proof is essentially the same for both PKE schemes and KEMs, we only show the PKE case below.

In order to show the statement, we will show that if there exists a $\langle\mathsf{B} : \mathsf{F} : \mathsf{A}\rangle$-mCCA secure PKE scheme $\Pi = (\mathsf{PKG}, \mathsf{PEnc}, \mathsf{PDec})$, then there exists a PKE scheme $\Pi_{\mathtt{sep1}} = (\mathsf{PKG}_{\mathtt{sep1}}, \mathsf{PEnc}_{\mathtt{sep1}}, \mathsf{PDec}_{\mathtt{sep1}})$ which is $\langle\mathsf{B} : \mathsf{F} : \mathsf{A}\rangle$-mCCA secure but is not $\langle\widetilde{\mathsf{B}} : \widetilde{\mathsf{F}} : \widetilde{\mathsf{A}}\rangle$-mCCA secure.

Specifically, let $q = \lvert(\mathsf{B}\lvert\lvert\mathsf{F}\lvert\lvert\mathsf{A})\rvert + 1$. We use the backdoor-sequence scheme $BS = (\mathsf{BSGen}, \mathsf{Release}, \mathsf{Recon})$ as a building block and construct the separating PKE scheme $\Pi_{\mathtt{sep1}}$ as in Fig. 4.2. Recall that a backdoor-sequence scheme can be constructed without any computational intractability assumption.

Without loss of generality, we assume that all the integers that appear in this proof have k-bit representation. In order to clarify that we are treating an integer as a k-bit string, we will use "hat". For example, $\widehat{1}$ is a k-bit representation of 1.

In the following, we show two lemmas that imply Theorem 11.

Lemma 11. *The PKE scheme* $\Pi_{\mathtt{sep1}}$ *is not* $\langle\widetilde{\mathsf{B}} : \widetilde{\mathsf{F}} : \widetilde{\mathsf{A}}\rangle$-mCCA *secure.*

Proof of Lemma 11. We construct a PPTA adversary $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$ as follows:

$\mathcal{A}_1$**:** Let $(\widetilde{\mathsf{B}}||\widetilde{\mathsf{F}}) = (b_1 \dots b_s)$ such that $b_i \in \{\mathsf{s}, \mathsf{p}\}$ for $i \in [s]$. On input $PK = (pk, \mathsf{pub})$, $\mathcal{A}_1$ sets $u_1 \leftarrow 1^k$ and then repeats the following for $1 \le i \le s$:

- if $b_i = \mathsf{s}$, then $\mathcal{A}_1$ issues $C = (\widehat{i}||\widehat{1}||u_i)$ as i-th (single) decryption query and is given $u_{i+1} = \mathsf{Release}(\mathsf{pub}, \mathsf{pri}, i, 1, u_i)$ as an answer.
- if $b_i = \mathsf{p}$, then $\mathcal{A}_1$ issues, as its i-th query, a parallel query $(C_1, C_2, \dots, C_q)$ such that $C_j = (\widehat{i}||\widehat{j}||u_i)$ for each $1 \le j \le q$, and is given $(y_1, \dots, y_q)$ as an answer, where $y_j = \mathsf{Release}(\mathsf{pub}, \mathsf{pri}, i, j, u_i)$ for every $1 \le j \le q$. Then $\mathcal{A}_1$ runs $u_{i+1} \leftarrow \mathsf{Recon}(\mathsf{pub}, (y_1, \dots, y_q))$.

Note that in both cases, the returned value(s) from the oracle is always the output of $\mathsf{Release}$, due to the design of the PKE scheme $\Pi_{\mathtt{sep1}}$. Moreover, since u_{i+1} can be always obtained from the response to i-th query, $\mathcal{A}_1$ can continue the above.

When the above repetition is completed, $\mathcal{A}_1$ picks two plaintexts $m_0, m_1 \in \mathcal{M}_\Pi$ such that $m_0 \neq m_1$, and sets $\mathsf{st}_{\mathcal{A}}$ that consists of all the values known to $\mathcal{A}_1$. Finally, $\mathcal{A}_1$ terminates with output $(m_0, m_1, \mathsf{st}_{\mathcal{A}})$.

$\mathcal{A}_2$**:** On input $(C^*, \mathsf{st}_{\mathcal{A}})$ where $C^* = (0^k||0^k||c^*)$, $\mathcal{A}_2$ makes queries in the same as $\mathcal{A}_1$ does (but this time $\mathcal{A}_2$ follows the query sequence $\widetilde{\mathsf{A}}$). When all the queries are completed, according to the definition of $\mathsf{PDec}_{\mathtt{sep1}}$ (and the definition of the backdoor-sequence scheme), $\mathcal{A}_2$ finally obtains $u_{n+1} = sk$ as an answer to $\mathcal{A}_2$'s final query if the final query was a single query or as an output from Recon if the final query was a parallel query. $\mathcal{A}_2$ can now tell the challenge bit by decrypting c^* by itself.

It is clear that $\mathcal{A}$ succeeds in guessing the challenge bit b with probability 1, and thus always has $\langle\widetilde{\mathsf{B}} : \widetilde{\mathsf{F}} : \widetilde{\mathsf{A}}\rangle$-$\mathtt{mCCA}$ advantage $\frac{1}{2}$, which is non-negligible. This completes the proof of Lemma 11. □

Lemma 12. *If the underlying PKE scheme* Π *is* $\langle\mathsf{B} : \mathsf{F} : \mathsf{A}\rangle$-$\mathtt{mCCA}$ *secure and the backdoor-sequence scheme BS is secure, then the PKE scheme* $\Pi_{\mathtt{sep1}}$ *is* $\langle\mathsf{B} : \mathsf{F} : \mathsf{A}\rangle$-$\mathtt{mCCA}$ *secure.*

Proof of Lemma 12. Let $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$ be any PPTA adversary that attacks the PKE scheme $\Pi_{\mathtt{sep1}}$ in the sense of $\langle\mathsf{B} : \mathsf{F} : \mathsf{A}\rangle$-$\mathtt{mCCA}$ security. Consider the following sequence of games.

Game 1 This is the ordinary $\langle\mathsf{B} : \mathsf{F} : \mathsf{A}\rangle$-$\mathtt{mCCA}$ experiment that $\mathcal{A}$ runs in.

Game 2 Same as Game 1, except that the input sk to the BSGen algorithm of the backdoor-sequence scheme run in $\mathsf{PKG}_{\mathtt{sep1}}$ is replaced with 0^k.

Let Succ_i denote the event that $\mathcal{A}$ succeeds in guessing the challenge bit in Game i. Then, the advantage of an adversary $\mathcal{A}$ is calculated as:

$$\mathsf{Adv}^{\langle\mathsf{B:F:A}\rangle\text{-}\mathtt{mCCA}}_{\Pi_{\mathtt{sep1}},\mathcal{A}} = |\Pr[\mathsf{Succ}_1] - \frac{1}{2}| \le |\Pr[\mathsf{Succ}_1] - \Pr[\mathsf{Succ}_2]| + |\Pr[\mathsf{Succ}_2] - \frac{1}{2}| \tag{4.3}$$

To upperbound the above advantage, we prove the following claims.

Claim 5. $|\Pr[\mathsf{Succ}_1] - \Pr[\mathsf{Succ}_2]|$ *is negligible.*

Proof of Claim 5. Assume towards a contradiction that $|\Pr[\mathsf{Succ}_1] - \Pr[\mathsf{Succ}_2]|$ is not negligible. Then we show that we can construct another PPTA adversary $\mathcal{B} = (\mathcal{B}_1, \mathcal{B}_2)$ that has non-negligible advantage in the $((\mathsf{B}||\mathsf{F}||\mathsf{A}), (\widetilde{\mathsf{B}}||\widetilde{\mathsf{F}}||\widetilde{\mathsf{A}}))$-backdoor-sequence experiment regarding $BS = (\mathsf{BSGen}, \mathsf{Release}, \mathsf{Recon})$.

The description of $\mathcal{B}$ is as follows.

$\mathcal{B}_1$: On input $\mathsf{pub} = (1^k, (\widetilde{\mathsf{B}}||\widetilde{\mathsf{F}}||\widetilde{\mathsf{A}}), q)$, $\mathcal{B}_1$ runs $(pk, sk) \leftarrow \mathsf{PKG}(1^k)$, and sets $v_0^* = 0^k$ and $v_1^* = sk$. Then $\mathcal{B}_1$ sets $\mathsf{st}_{\mathcal{B}}$ that consists of all the values known to $\mathcal{B}_1$, and terminates with output $(v_0^*, v_1^*, \mathsf{st}_{\mathcal{B}})$.

$\mathcal{B}_2$: On input $\mathsf{st}_{\mathcal{B}}$, $\mathcal{B}_2$ runs $\mathcal{A}_1$ with input $PK = (pk, \mathsf{pub})$. When $\mathcal{A}_1$ issues decryption queries, $\mathcal{B}_2$ responds as follows.

- If $\mathcal{A}_1$'s query is a single query $C = (\alpha||\beta||c)$, $\mathcal{B}_2$ first runs $m \leftarrow \mathsf{PDec}(sk, c)$. Next, $\mathcal{B}_2$ issues a single query (α, β, c) to $\mathcal{B}$'s own oracle and receives the result y. $\mathcal{B}_1$ sets $m \leftarrow y$ if $(\alpha||\beta) \neq (0^k||0^k)$. Finally, $\mathcal{B}_1$ sends m back to $\mathcal{A}_1$.
- If $\mathcal{A}_1$'s query is a parallel query $\overrightarrow{C} = (C_1, C_2, \dots)$, where $C_i = (\alpha_i||\beta_i||c_i)$ for each $1 \leq i \in |\overrightarrow{C}|$, $\mathcal{B}_2$ first computes $m_i \leftarrow \mathsf{PDec}(sk, c_i)$ for every $1 \leq i \leq |\overrightarrow{C}|$. Then $\mathcal{B}_2$ also issues $((\alpha_1, \beta_1, c_1), (\alpha_2, \beta_2, c_2), \dots)$ as a parallel query to $\mathcal{B}$'s own oracle and receives the result $(y_1, y_2, \dots)$. Then for each $1 \leq i \leq |\overrightarrow{C}|$, $\mathcal{B}_2$ sets $m_i \leftarrow y_i$ if $(\alpha||\beta) \neq (0^k||0^k)$. Finally, $\mathcal{B}_2$ sends $(m_1, m_2, \dots)$ back to $\mathcal{A}_1$.

When $\mathcal{A}_1$ terminates with output two plaintexts (m_0, m_1) of equal length and state information $\mathsf{st}_{\mathcal{A}}$, $\mathcal{B}_2$ picks a bit $\gamma \in \{0, 1\}$ uniformly at random (which will play a role of the challenge bit for $\mathcal{A}$), computes $c^* \leftarrow \mathsf{PEnc}(pk, m_\gamma)$, and sets $C^* \leftarrow (0^k||0^k||c^*)$. Then $\mathcal{B}_2$ runs $\mathcal{A}_2$ with input $(C^*, \mathsf{st}_{\mathcal{A}})$. The queries from $\mathcal{A}_2$ are answered in exactly the same way as the queries from $\mathcal{A}_1$. When $\mathcal{A}_2$ terminates with output γ', $\mathcal{B}_2$ sets $b' \leftarrow 1$ if $\gamma' = \gamma$ or $b' \leftarrow 0$ otherwise. Finally, $\mathcal{B}_2$ terminates with output b'.

Let b be a bit that the adversary $\mathcal{B}$ has to guess in the $((\mathsf{B}||\mathsf{F}||\mathsf{A}), (\widetilde{\mathsf{B}}||\widetilde{\mathsf{F}}||\widetilde{\mathsf{A}}))$-backdoor-sequence experiment.

Note that $\mathcal{B}$ makes exactly the same type of sequence of queries as $\mathcal{A}$, which is $(\mathsf{B}||\mathsf{F}||\mathsf{A})$. The advantage of $\mathcal{B}$ is calculated as:

$$\begin{aligned}
\mathsf{Adv}_{BS,\mathcal{B}}^{((\mathsf{B}||\mathsf{F}||\mathsf{A}),(\widetilde{\mathsf{B}}||\widetilde{\mathsf{F}}||\widetilde{\mathsf{A}}))} &= |\Pr[b' = b] - \frac{1}{2}| \\
&= \frac{1}{2}|\Pr[b' = 1|b = 1] - \Pr[b' = 1|b = 0]| \\
&= \frac{1}{2}|\Pr[\gamma' = \gamma|b = 1] - \Pr[\gamma' = \gamma|b = 0]|
\end{aligned}$$

It is easy to see that when $b = 1$, $\mathcal{B}$ perfectly simulates Game 1 for $\mathcal{A}$ in which the challenge bit for $\mathcal{A}$ is γ. In particular, the secret value input to BSGen in $\mathcal{B}$'s experiment is $v_b^* = v_1^* = sk$, which is exactly the procedure done in $\mathsf{PKG}_{\mathsf{sep1}}$ in Game 1. Under this situation, the event $\gamma' = \gamma$ corresponds to the event Succ_1, i.e. $\Pr[\gamma' = \gamma|b = 1] = \Pr[\mathsf{Succ}_1]$.

When $b = 0$, on the other hand, the secret value input to BSGen in $\mathcal{B}$'s experiment is $v_b^* = v_0^* = 0^k$. It is again easy to see that $\mathcal{B}$ does the perfect simulation of Game 2 for $\mathcal{A}$ in which the challenge bit for $\mathcal{A}$ is γ, and with a similar argument to the above, we have $\Pr[\gamma' = \gamma|b = 0] = \Pr[\mathsf{Succ}_2]$.

In summary, we have $\mathsf{Adv}_{BS,\mathcal{B}}^{((\mathsf{B}||\mathsf{F}||\mathsf{A}),(\widetilde{\mathsf{B}}||\widetilde{\mathsf{F}}||\widetilde{\mathsf{A}}))} = \frac{1}{2}|\Pr[\mathsf{Succ}_1] - \Pr[\mathsf{Succ}_2]|$, which is not negligible by the assumption we made at the beginning of the proof of this claim. However, since $(\widetilde{\mathsf{B}}||\widetilde{\mathsf{F}}||\widetilde{\mathsf{A}}) \not\subseteq_{qs} (\mathsf{B}||\mathsf{F}||\mathsf{A})$, the existence of such $\mathcal{B}$ contradicts the security of the backdoor-sequence scheme BS. Therefore, $|\Pr[\mathsf{Succ}_1] - \Pr[\mathsf{Succ}_2]|$ must be negligible. This completes the proof of Claim 5. □

Claim 6. $|\Pr[\mathsf{Succ}_2] - \frac{1}{2}|$ *is negligible.*

Proof of Claim 6. Assume towards a contradiction that $|\Pr[\mathsf{Succ}_2] - \frac{1}{2}|$ is not negligible. Then we show that we can construct another PPTA adversary $\mathcal{B} = (\mathcal{B}_1, \mathcal{B}_2)$ that can break the $\langle \mathsf{B} : \mathsf{F} : \mathsf{A} \rangle$-$\mathtt{mCCA}$ security of the underlying PKE scheme Π with non-negligible advantage. The description of $\mathcal{B}$ is as follows.

$\mathcal{B}_1$: On input pk, $\mathcal{B}_1$ sets $\mathsf{pub} = (1^k, (\widetilde{\mathsf{B}}||\widetilde{\mathsf{F}}||\widetilde{\mathsf{A}}), q)$, and runs $\mathsf{pri} \leftarrow \mathsf{BSGen}(\mathsf{pub}, 0^k)$. Then $\mathcal{B}_1$ runs $\mathcal{A}_1$ with input $PK = (pk, \mathsf{pub})$.

When $\mathcal{A}_1$ issues decryption queries, $\mathcal{B}_1$ responds as follows.

- If $\mathcal{A}_1$'s query is a single query $C = (\alpha||\beta||c)$, then $\mathcal{B}_1$ issues c to its decryption oracle and obtains m. Then if $(\alpha||\beta) \neq (0^k||0^k)$, $\mathcal{B}_1$ sets $m \leftarrow \mathsf{Release}(\mathsf{pub}, \mathsf{pri}, \alpha, \beta, c)$. Finally $\mathcal{B}_1$ sends m back to $\mathcal{A}_1$.
- If $\mathcal{A}_1$'s query is a parallel query $\overrightarrow{C} = (C_1, C_2, \dots)$, where $C_i = (\alpha_i||\beta_i||c_i)$, $\mathcal{B}_1$ makes a parallel query $\overrightarrow{c} = (c_1, c_2, \dots)$ to $\mathcal{B}$'s decryption oracle and receives the answer $(m_1, m_2, \dots)$. Then for each $1 \leq i \leq |\overrightarrow{C}|$, $\mathcal{B}_1$ sets $m_i \leftarrow \mathsf{Release}(\mathsf{pub}, \mathsf{pri}, \alpha_i, \beta_i, c_i)$ if $(\alpha_i||\beta_i) \neq (0^k||0^k)$. Finally $\mathcal{B}_1$ sends $(m_1, m_2, \dots)$ back to $\mathcal{A}_1$.

When $\mathcal{A}_1$ terminates with output two plaintexts (m_0, m_1) of equal length and state information st_A, $\mathcal{B}_1$ sets its own state information $\mathsf{st}_{\mathcal{B}}$ that consists of all the values known to $\mathcal{B}_1$ and terminates with output $(m_0, m_1, \mathsf{st}_{\mathcal{B}})$.

$\mathcal{B}_2$: On input $(c^*, \mathsf{st}_{\mathcal{B}})$, $\mathcal{B}_2$ sets $C^* \leftarrow (0^k||0^k||c^*)$ and runs $\mathcal{A}_2$ with input $(C^*, \mathsf{st}_{\mathcal{A}})$. The queries from $\mathcal{A}_2$ are answered in exactly the same way as the queries from $\mathcal{A}_1$, except that if $\mathcal{A}_2$'s query contains a ciphertext of the form $C = (\alpha||\beta||c^*)$, $\mathcal{B}_2$ does not submit it to the decryption oracle and $\mathcal{B}_2$ directly computes $m \leftarrow \mathsf{Release}(\mathsf{pub}, \mathsf{pri}, \alpha, \beta, c^*)$ as a decryption result of this C. Note that such ciphertext $C = (\alpha||\beta||c^*)$ cannot satisfy $(\alpha||\beta) = (0^k||0^k)$ by the definition of the $(\mathsf{B} : \mathsf{F} : \mathsf{A})$-$\mathtt{mCCA}$ experiment, and setting $\mathsf{Release}(\mathsf{pub}, \mathsf{pri}, \alpha, \beta, c^*)$ as a decryption result for this C is a legitimate response for $\mathcal{A}$. When $\mathcal{A}_2$ terminates with output b', $\mathcal{B}_2$ outputs this b' and terminates.

Note that $\mathcal{B}$ makes exactly the same type of the query sequence as $\mathcal{A}$, and thus $\mathcal{B}$ is a legitimate $\langle \mathsf{B} : \mathsf{F} : \mathsf{A} \rangle$-$\mathtt{mCCA}$ adversary regarding Π. Moreover, it is easy to see that $\mathcal{B}$ does the perfect simulation of Game 2 for $\mathcal{A}$, and in particular, the challenge bit for $\mathcal{B}$ is that for $\mathcal{A}$. Therefore, $\mathcal{B}$ succeeds in guessing the challenge bit whenever $\mathcal{A}$ does so, and we have $\mathsf{Adv}_{\Pi,\mathcal{B}}^{\langle \mathsf{B}:\mathsf{F}:\mathsf{A} \rangle\text{-}\mathtt{mCCA}} = |\Pr[\mathsf{Succ}_2] - \frac{1}{2}|$, but this is not negligible by the assumption we made at the beginning of the proof of this claim. Since this contradicts the $\langle \mathsf{B} : \mathsf{F} : \mathsf{A} \rangle$-$\mathtt{mCCA}$ security of the underlying PKE scheme Π, $|\Pr[\mathsf{Succ}_2] - \frac{1}{2}|$ must be negligible. This completes the proof of Claim 6. □

According to the inequality (4.3) and Claims 5 and 6, we can upperbound $\mathsf{Adv}_{\Pi_{\mathsf{sep1}},\mathcal{A}}^{\langle \mathsf{B}:\mathsf{F}:\mathsf{A} \rangle\text{-}\mathtt{mCCA}}$ to be negligible for any PPTA adversary $\mathcal{A}$. This completes the proof of Lemma 12. □

Lemmas 11 and 12 imply Theorem 11. □

Next, by focusing on the "after-challenge" queries, we show the following separation.

Theorem 12. *For both PKE schemes and KEMs, if* $\widetilde{\mathsf{A}} \not\subseteq_{qs} \mathsf{A}$*, then* $\langle \mathsf{unbound} :: \mathsf{A} \rangle$-$\mathtt{mCCA}$ *security does not imply* $\langle \emptyset :: \widetilde{\mathsf{A}} \rangle$-$\mathtt{mCCA}$ *security.*

Intuition. Note that a $\langle \mathsf{unbound} :: \mathsf{A} \rangle$-$\mathtt{mCCA}$ adversary can make unbounded single queries before the challenge while a $\langle \emptyset :: \widetilde{\mathsf{A}} \rangle$-$\mathtt{mCCA}$ adversary can make no query. Therefore, the critical information for breaking $\langle \emptyset :: \widetilde{\mathsf{A}} \rangle$-$\mathtt{mCCA}$ security must be something that is useful and available only after the challenge. We set the critical information to be the decryption of the challenge ciphertext itself. This means that the backdoor mechanism cannot be set up in the key generation algorithm (because the sequence of backdoors are determined only after the critical information v is determined), and must be set up in a decryption algorithm, while keeping the decryption algorithm stateless and deterministic.

We use a pseudorandom function to realize the separating PKE scheme that has a "ciphertext-specific" backdoor mechanism, meaning that the backdoor mechanism is not the same for all ciphertext, but is set up differently depending on the input ciphertext each time the decryption (by the decryption oracle) is performed. More specifically, a seed K for a pseudorandom function F is picked as a part of a secret key of the separating PKE scheme. The decryption algorithm of the separating scheme, on input a ciphertext c togather with backdoor and some information that indicates "backdoor mode", derives a pseudorandom value $R = F_K(c)$ and use this R as a randomness for deriving the sequence of backdoors, and then outputs a "next backdoor" or "reject" depending on the backdoor that is input with c. If the separating PKE scheme is constructed as above, unbounded single queries before the challenge is meaningless because the challenge ciphertext is not available before the challange.

To use a pseudorandom function as above is the idea first used by Bellare et al. [7] who used it to show the separation between `NM-CCA1` security and `NM-CCA2` security (i.e. `IND-CCA2` security).

Proof. Since the proof is essentially the same for both PKE schemes and KEMs, we only show the PKE case below.

In order to show the statement, we will show that if there exists a $\langle \mathsf{unbound} :: \mathsf{A} \rangle$-$\mathtt{mCCA}$ secure PKE scheme $\Pi = (\mathsf{PKG}, \mathsf{PEnc}, \mathsf{PDec})$, then there exists a PKE scheme $\Pi_{\mathtt{sep2}} = (\mathsf{PKG}_{\mathtt{sep2}}, \mathsf{PEnc}_{\mathtt{sep2}}, \mathsf{PDec}_{\mathtt{sep2}})$ which is $\langle \mathsf{unbound} :: \mathsf{A} \rangle$-$\mathtt{mCCA}$ secure but is not $\langle \emptyset :: \widetilde{\mathsf{A}} \rangle$-$\mathtt{mCCA}$ secure.

Specifically, let $q = |\mathsf{A}| + 1$. We use the backdoor-sequence scheme $BS = (\mathsf{BSGen}, \mathsf{Release}, \mathsf{Recon})$ and a PRF F_K (where $K \in \{0,1\}^k$ is a seed for F) as building blocks to construct the separating PKE scheme $\Pi_{\mathtt{sep2}}$ as in Fig. 4.3. We assume that the length of ciphertext in the underlying PKE scheme Π is $\ell = \ell(k)$ if generated under a correctly generated public key pk that is output from $\mathsf{PKG}(1^k)$. Let $\mathcal{R}_{BS}$ be the randomness space of BSGen. We also require that for any $K \in \{0,1\}^k$ PRF F_K is of the form $F_K : \{0,1\}^\ell \to \mathcal{R}_{BS}$, and such PRF can be constructed by only assuming the existence of a $\langle \mathsf{unbound} :: \mathsf{A} \rangle$-$\mathtt{mCCA}$ secure PKE scheme. We note that a backdoor-sequence scheme can be constructed without any computational intractability assumption.

Without loss of generality, we assume that all the integers that appear in this proof have k-bit representation. In order to clarify that we are treating an integer as a k-bit string, we will use "hat". For example, $\widehat{1}$ is a k-bit representation of 1.

$\mathsf{PKG}_{\mathtt{sep2}}(1^k)$:	$\mathsf{PEnc}_{\mathtt{sep2}}(PK, m)$:
$(pk, sk) \leftarrow \mathsf{PKG}(1^k)$ $q \leftarrow \lvert\mathsf{A}\rvert + 1$ $\mathsf{pub} \leftarrow (1^k, \widetilde{\mathsf{A}}, q)$ $K \leftarrow \{0,1\}^k$ $PK \leftarrow (pk, \mathsf{pub})$ $SK \leftarrow (sk, K)$ Return (PK, SK).	Parse PK as (pk, pub). $c \leftarrow \mathsf{PEnc}(pk, m)$ Return $C \leftarrow (0^k \lvert\lvert 0^k \lvert\lvert 0^k \lvert\lvert c)$.
$\mathsf{PDec}_{\mathtt{sep2}}(SK, C)$: Parse SK as (sk, K) and C as $(\alpha \lvert\lvert \beta \lvert\lvert w \lvert\lvert c)$ such that $\lvert\alpha\rvert = \lvert\beta\rvert = \lvert w\rvert = k$. $m \leftarrow \mathsf{PDec}(sk, c)$ If $(\alpha \lvert\lvert \beta \lvert\lvert w) = (0^k \lvert\lvert 0^k \lvert\lvert 0^k)$ then return m. $R \leftarrow F_K(c)$ $\mathsf{pri} \leftarrow \mathsf{BSGen}(\mathsf{pub}, m; R)$ Return $\mathsf{Release}(\mathsf{pub}, \mathsf{pri}, \alpha, \beta, w)$.	

Figure 4.3: The PKE scheme $\Pi_{\mathtt{sep2}}$ that separates $\langle \emptyset :: \widetilde{\mathsf{A}} \rangle$-mCCA from $\langle \mathsf{unbound} :: \mathsf{A} \rangle$-mCCA in case $\widetilde{\mathsf{A}} \not\subseteq_{qs} \mathsf{A}$.

In the following, we show two lemmas that imply Theorem 12.

Lemma 13. *The PKE scheme $\Pi_{\mathtt{sep2}}$ is not $\langle \emptyset :: \widetilde{\mathsf{A}} \rangle$-mCCA secure.*

Proof of Lemma 13. We construct a PPTA adversary $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$ as follows:

$\mathcal{A}_1$: On input $PK = (pk, \mathsf{pub})$, $\mathcal{A}_1$ picks two plaintexts m_0, m_1 such that $m_0 \neq m_1$ and $\mathsf{st}_{\mathcal{A}}$ that consists of all the values known to $\mathcal{A}_1$. Then $\mathcal{A}_1$ terminates with output $(m_0, m_1, \mathsf{st}_{\mathcal{A}})$.

$\mathcal{A}_2$: Let $\widetilde{\mathsf{A}} = (b_1 \dots b_n)$ such that $b_i \in \{\mathsf{s}, \mathsf{p}\}$ for $i \in [n]$. On input $(C^*, \mathsf{st}_{\mathcal{A}})$ where $C^* = (0^k \lvert\lvert 0^k \lvert\lvert 0^k \lvert\lvert c^*)$, $\mathcal{A}_2$ sets $u_1 \leftarrow 1^k$ and then repeats the following for $1 \leq i \leq n$:

- if $b_i = \mathsf{s}$, then $\mathcal{A}_2$ issues $C = (\widehat{i} \lvert\lvert \widehat{1} \lvert\lvert u_i \lvert\lvert c^*)$ as i-th (single) decryption query and is given $u_{i+1} = \mathsf{Release}(\mathsf{pub}, \mathsf{pri}, i, 1, u_i)$ as an answer.
- if $b_i = \mathsf{p}$, then $\mathcal{A}_2$ issues, as its i-th query, a parallel query $(C_1, C_2, \dots, C_q)$ such that $C_j = (\widehat{i} \lvert\lvert \widehat{j} \lvert\lvert u_i \lvert\lvert c^*)$ for each $1 \leq j \leq q$, and is given $(y_1, \dots, y_q)$ as an answer, where $y_j = \mathsf{Release}(\mathsf{pub}, \mathsf{pri}, i, j, u_i)$ for every $1 \leq j \leq q$. Then $\mathcal{A}_2$ runs $u_{i+1} \leftarrow \mathsf{Recon}(\mathsf{pub}, (y_1, \dots, y_q))$.

Note that in both cases, the returned value(s) from the oracle is always the output of $\mathsf{Release}(\mathsf{pub}, \mathsf{pri}, \cdot, \cdot, \cdot)$, where $\mathsf{pri} = \mathsf{BSGen}(\mathsf{pub}, \mathsf{PDec}(c^*); F_K(c^*))$, due to the design of the PKE scheme $\Pi_{\mathtt{sep2}}$. In particular, pri is always the same as long as c^* is used. Moreover, since u_{i+1} can be always obtained from the response to i-th query, $\mathcal{A}_2$ can continue the above.

When all the queries are completed, according to the definition of $\mathsf{PDec}_{\mathtt{sep2}}$ (and the definition of the backdoor-sequence scheme), $\mathcal{A}_2$ finally obtains $u_{n+1} = \mathsf{PDec}(sk, c^*) = m_b$ as an answer to $\mathcal{A}_2$'s final query if the final query was a single query or as an output from Recon if the final query was a parallel query. Since $\mathcal{A}_2$ has now obtained the decryption result of the challenge ciphertext c^*, $\mathcal{A}_2$ can now outputs b and terminates.

It is clear that $\mathcal{A}$ succeeds in guessing the challenge bit b with probability 1, and thus always has $\langle \emptyset :: \widetilde{\mathsf{A}} \rangle$-$\mathtt{mCCA}$ advantage $\frac{1}{2}$, which is non-negligible. This completes the proof of Lemma 13. □

Lemma 14. *If the underlying PKE scheme* Π *is* $\langle \mathsf{unbound} :: \mathsf{A} \rangle$-$\mathtt{mCCA}$ *secure, the backdoor-sequence scheme BS is secure, and F is a PRF, then the PKE scheme* $\Pi_{\mathtt{sep2}}$ *is* $\langle \mathsf{unbound} :: \mathsf{A} \rangle$-$\mathtt{mCCA}$ *secure.*

Proof of Lemma 14. Let $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$ be any PPTA adversary that attacks the PKE scheme $\Pi_{\mathtt{sep2}}$ in the sense of $\langle \mathsf{unbound} :: \mathsf{A} \rangle$-$\mathtt{mCCA}$ security. Consider the following sequence of games. (The values with asterisk (*) are the ones that appear when generating the challenge ciphertext $C^* = (0^k||0^k||0^k||c^*)$.)

Game 1 This is the ordinary $\langle \mathsf{unbound} :: \mathsf{A} \rangle$-$\mathtt{mCCA}$ experiment that $\mathcal{A}$ runs in.

Game 2 Same as Game 1, except that after the challenge ciphertext $C^* = (0^k||0^k||0^k||c^*)$ is generated, if $\mathcal{A}_1$ has submitted at least one ciphertext of the form $C = (\alpha||\beta||w||c^*)$, then all the decryption queries from $\mathcal{A}_2$ (i.e. queries after the challenge) are answered with $\perp$.

Game 3 Same as Game 2, except that the PRF F_K used in the decryption oracle is replaced with a truly random function $RF : \{0,1\}^\ell \to \mathcal{R}_{BS}$.

Game 4 Same as Game 3, except that if the decryption oracle receives a ciphertext (via a single query or a parallel query) of the form $C = (\alpha||\beta||w||c^*)$ from $\mathcal{A}_2$, when running BSGen, the secret value $m_b = \mathsf{PDec}(sk, c^*)$ input into BSGen is replaced with 0^k. That is, the decryption of $C = (\alpha||\beta||w||c^*)$ by the decryption oracle is replaced with $m \leftarrow \mathsf{Release}(\mathsf{pub}, \mathsf{pri}, \alpha, \beta, w)$ where $\mathsf{pri} \leftarrow \mathsf{BSGen}(\mathsf{pub}, 0^k; RF(c^*))$.

Let Succ_i denote the event that $\mathcal{A}$ succeeds in guessing the challenge bit in Game i, and let Coll_i be the event that after the challenge ciphertext $C^* = (0^k||0^k||0^k||c^*)$ is generated, $\mathcal{A}_1$ has submitted at least one ciphertext $C = (\alpha||\beta||w||c^*)$ to the decryption oracle.

Then, the advantage of an adversary $\mathcal{A}$ is calculated as:

$$\begin{aligned} \mathsf{Adv}^{\langle \mathsf{unbound::A} \rangle\text{-}\mathtt{mCCA}}_{\Pi_{\mathtt{sep2}},\mathcal{A}} &= |\Pr[\mathsf{Succ}_1] - \frac{1}{2}| \\ &\leq \sum_{i \in \{1,2,3\}} |\Pr[\mathsf{Succ}_i] - \Pr[\mathsf{Succ}_{i+1}]| + |\Pr[\mathsf{Succ}_4] - \frac{1}{2}| \end{aligned} \tag{4.4}$$

To upperbound the above advantage, we show the following claims.

Claim 7. $|\Pr[\mathsf{Succ}_1] - \Pr[\mathsf{Succ}_2]|$ *is negligible.*

Proof of Claim 7. Note that the Game 1 and Game 2 proceed identically until the event $\mathsf{Coll}_1 = \mathsf{Coll}_2$ happens, and thus we have

$$|\Pr[\mathsf{Succ}_1] - \Pr[\mathsf{Succ}_2]| \leq \Pr[\mathsf{Coll}_1] = \Pr[\mathsf{Coll}_2]$$

Therefore, we show that the upperbound of $\Pr[\mathsf{Coll}_1]$ is negligible. Assume towards a contradiction that $\Pr[\mathsf{Coll}_i]$ is not negligible. Then we show that we can construct another PPTA adversary $\mathcal{B} = (\mathcal{B}_1, \mathcal{B}_2)$ that has non-negligible advantage in the $\mathtt{IND\text{-}CCA1}$ experiment regarding the underlying PKE scheme Π. The description of $\mathcal{B}$ is as follows.

$\mathcal{B}_1$**:** On input pk, $\mathcal{B}_1$ sets $\mathsf{pub} = (1^k, \widetilde{\mathsf{A}}, q)$, and picks a PRF key $K \leftarrow \{0,1\}^k$. and then runs $\mathcal{A}_1$ with input $PK = (pk, \mathsf{pub})$. All the (single) decryption queries C from $\mathcal{A}_1$ are answered by faithfully following the procedure of $\mathsf{PDec}_{\mathsf{sep2}}$, except that if $\mathcal{B}_1$ needs to run $\mathsf{PDec}(sk, c)$, $\mathcal{B}_1$ uses its own decryption oracle. When $\mathcal{A}_1$ terminates with output two plaintexts (m_0, m_1) of equal length and state information st_A, $\mathcal{B}_1$ sets its own state information $\mathsf{st}_{\mathcal{B}}$ that consists of all the values known to $\mathcal{B}_1$. Finally, $\mathcal{B}_1$ terminates with output $(m_0, m_1, \mathsf{st}_{\mathcal{B}})$.

$\mathcal{B}_2$**:** On input $(c^*, \mathsf{st}_{\mathcal{B}})$, $\mathcal{B}_2$ checks if $\mathcal{A}_1$ has submitted a decryption query of the form $C = (\alpha||\beta||w||c^*)$. If no such query has been made, $\mathcal{B}_2$ picks a random bit $b' \leftarrow \{0,1\}$ and terminates with output b'. If such query has been made, then, according to our design of $\mathcal{B}_1$, c^* has been also submitted to $\mathcal{B}$'s own decryption oracle. Let m be the response of $\mathcal{B}$'s decryption oracle for the query c^*. $\mathcal{B}_2$ sets $b' \leftarrow 0$ if $m = m_0$, or sets $b' \leftarrow 1$ otherwise. (Here, $m = \mathsf{PDec}(sk, c^*)$ must be either m_0 or m_1, due to correctness of the PKE scheme Π.) Finally, $\mathcal{B}_2$ outputs this b' and terminates.

Let $\mathsf{Succ}_{\mathcal{B}}$ be the event that $\mathcal{B}$ succeeds in guessing the challenge bit, and $\mathsf{Find}_{\mathcal{B}}$ be the event that in $\mathcal{B}$'s experiment, $\mathcal{B}_2$ finds a ciphertext $C = (\alpha||\beta||w||c^*)$ from $\mathcal{A}_1$'s decryption oracle query. Due to the description of $\mathcal{B}$, we have $\Pr[\mathsf{Succ}_{\mathcal{B}}|\mathsf{Find}_{\mathcal{B}}] = 1$ and $\Pr[\mathsf{Succ}_{\mathcal{B}}|\overline{\mathsf{Find}_{\mathcal{B}}}] = \frac{1}{2}$. Moreover, it is clear that $\mathcal{B}$ perfectly simulates Game 1 for $\mathcal{A}_1$. Under this situation, the event $\mathsf{Find}_{\mathcal{B}}$ occurs if and only if Coll_1 occurs, i.e., $\Pr[\mathsf{Find}_{\mathcal{B}}] = \Pr[\mathsf{Coll}_1]$.

Using the above, $\mathcal{B}$'s `IND-CCA1` advantage is estimated as:

$$\begin{aligned}
\mathsf{Adv}^{\texttt{IND-CCA1}}_{\Pi,\mathcal{B}} &= |\Pr[\mathsf{Succ}_{\mathcal{B}}] - \frac{1}{2}| \\
&= |\Pr[\mathsf{Succ}_{\mathcal{B}}|\mathsf{Find}_{\mathcal{B}}] \cdot \Pr[\mathsf{Find}_{\mathcal{B}}] + \Pr[\mathsf{Succ}_{\mathcal{B}}|\overline{\mathsf{Find}_{\mathcal{B}}}] \cdot \Pr[\overline{\mathsf{Find}_{\mathcal{B}}}] - \frac{1}{2}| \\
&= |\Pr[\mathsf{Coll}_1] + \frac{1}{2}(1 - \Pr[\mathsf{Coll}_1]) - \frac{1}{2}| \\
&= \frac{1}{2}\Pr[\mathsf{Coll}_1]
\end{aligned}$$

which is non-negligible by the assumption we made above. Therefore, $\mathcal{B}$ breaks an `IND-CCA1` security of the underlying PKE scheme Π. However, the existence of such $\mathcal{B}$ is a contradiction because Π is $\langle \mathsf{unbound} :: \mathsf{A} \rangle$-`mCCA` secure, which is trivially `IND-CCA1` secure. Therefore, $\Pr[\mathsf{Coll}_1]$ must be negligible, which also upperbounds $|\Pr[\mathsf{Succ}_1] - \Pr[\mathsf{Succ}_2]|$ to be negligible. This completes the proof of Claim 7. □

Claim 8. $|\Pr[\mathsf{Succ}_2] - \Pr[\mathsf{Succ}_3]|$ *is negligible.*

We omit the proof of this claim, since it is almost trivial to see due to the security of the PRF F. Note that F is only used in the decryption algorithm. If $|\Pr[\mathsf{Succ}_1] - \Pr[\mathsf{Succ}_2]|$ is not negligible, we can use $\mathcal{A}$ to distinguish a PRF F_K (where K is randomly chosen) from a truly random function RF.

Claim 9. $|\Pr[\mathsf{Succ}_3] - \Pr[\mathsf{Succ}_4]|$ *is negligible.*

Proof of Claim 9. Assume towards a contradiction that $|\Pr[\mathsf{Succ}_3] - \Pr[\mathsf{Succ}_4]|$ is not negligible. Then we show that we can construct another PPTA adversary $\mathcal{B} = (\mathcal{B}_1, \mathcal{B}_2)$ that has non-negligible advantage in the $(\mathsf{A}, \widetilde{\mathsf{A}})$-backdoor-sequence experiment regarding $BS = (\mathsf{BSGen}, \mathsf{Release}, \mathsf{Recon})$.

The description of $\mathcal{B}$ is as follows.

$\mathcal{B}_1$**:** On input $\mathsf{pub} = (1^k, \widetilde{\mathsf{A}}, q)$, $\mathcal{B}_1$ runs $(pk, sk) \leftarrow \mathsf{PKG}(1^k)$. $\mathcal{B}_1$ then generates an empty list L into which pairs of the form $(c, R) \in \{0,1\}^\ell \times \mathcal{R}_{BS}$ will be stored for simulating a truly random function RF in a lazy sampling manner.

Then $\mathcal{B}_1$ runs $\mathcal{A}_1$ with input $PK = (pk, \mathsf{pub})$. All the (single) queries C from $\mathcal{A}_1$ are answered by following the decryption procedure in Game 3 (and in Game 4), except that when $\mathcal{B}_1$ has to compute a random function $RF(c)$, $\mathcal{B}_1$ simulates it by a lazy sampling[5] using the list L. When $\mathcal{A}_1$ terminates with output two plaintexts (m_0, m_1) of equal length and state information st_A, $\mathcal{B}_1$ flips a random coin $\gamma \in \{0,1\}$, and sets $v_1^* \leftarrow m_\gamma$ and $v_0^* \leftarrow 0^k$. $\mathcal{B}_1$ then sets its own state information $\mathsf{st}_\mathcal{B}$ that consists of all the values known to $\mathcal{B}_1$. Finally, $\mathcal{B}_1$ terminates with output $(v_0^*, v_1^*, \mathsf{st}_\mathcal{B})$.

$\mathcal{B}_2$**:** On input $\mathsf{st}_\mathcal{B}$, $\mathcal{B}_2$ runs $c^* \leftarrow \mathsf{PEnc}(pk, m_\gamma)$, sets $C^* \leftarrow (0^k||0^k||0^k||c^*)$, and runs $\mathcal{A}_2$ with input $PK = (C^*, \mathsf{st}_\mathcal{A})$. If $\mathcal{A}_1$ has made at least one (single) decryption query of the form $C = (\alpha||\beta||w||c^*)$, then $\mathcal{B}_2$ responds to all the decryption queries from $\mathcal{A}_2$ with $\perp$. Otherwise, $\mathcal{B}_2$ responds to the decryption queries from $\mathcal{A}_2$ as follows.

- If $\mathcal{A}_2$'s query is a single query $C = (\alpha||\beta||w||c)$, $\mathcal{B}_2$ first runs $m \leftarrow \mathsf{PDec}(sk, c)$. If $(\alpha||\beta||w) = (0^k||0^k||0^k)$, then $\mathcal{B}_2$ returns m to $\mathcal{A}$. If $c \neq c^*$, then $\mathcal{B}_2$ simulates $R \leftarrow RF(c)$ using L as above, computes $\mathsf{pri} \leftarrow \mathsf{BSGen}(\mathsf{pub}, m; R)$ and returns $y \leftarrow \mathsf{Release}(\mathsf{pub}, \mathsf{pri}, \alpha, \beta, w)$ to $\mathcal{A}_2$. Otherwise (i.e. $c = c^*$), $\mathcal{B}_2$ issues a single query (α, β, w) to $\mathcal{B}$'s own oracle, receives the result y from the oracle, and returns y to $\mathcal{A}_2$. (In this process, if $\mathcal{B}_2$ has not used its own oracle, $\mathcal{B}_2$ makes some query to its own oracle to ensure that the query sequence of $\mathcal{A}$ and that of $\mathcal{B}$ remain the same.)
- If $\mathcal{A}_2$'s query is a parallel query $\overrightarrow{C} = (C_1, C_2, \dots)$, where $C_i = (\alpha_i||\beta_i||w_i||c_i)$ for each $1 \leq i \leq |\overrightarrow{C}|$, $\mathcal{B}_2$ first computes $m_i \leftarrow \mathsf{PDec}(sk, c_i)$ for every $1 \leq i \leq |\overrightarrow{C}|$. Then $\mathcal{B}_2$ also issues $((\alpha_1, \beta_1, w_1), (\alpha_2, \beta_2, w_2), \dots)$ as a parallel query to $\mathcal{B}$'s own oracle and receives the result $(y_1, y_2, \dots)$. Then for each $1 \leq i \leq |\overrightarrow{C}|$, $\mathcal{B}_2$ sets $m_i \leftarrow y_i$ if $(\alpha_i||\beta_i||w_i) \neq (0^k||0^k||0^k)$ and $c_i = c^*$. Next, for each $1 \leq i \leq |\overrightarrow{C}|$, $\mathcal{B}_2$ sets $m_i \leftarrow y_i'$ if $(\alpha_i||\beta_i||w_i) \neq (0^k||0^k||0^k)$ and $c_i \neq c^*$, where $y_i' = \mathsf{Release}(\mathsf{pub}, \mathsf{pri}_i, \alpha_i, \beta_i, w_i)$, $\mathsf{pri}_i \leftarrow \mathsf{BSGen}(\mathsf{pub}, m_i; R_i)$, and $R_i \leftarrow RF(c_i)$ (this is simulated by a lazy sampling using L as above). Finally, $\mathcal{B}_2$ sends $(m_1, m_2, \dots)$ back to $\mathcal{A}_1$.

When $\mathcal{A}_2$ terminates with output γ', $\mathcal{B}_2$ sets $b' \leftarrow 1$ if $\gamma' = \gamma$ or $b' \leftarrow 0$ otherwise. Finally, $\mathcal{B}_2$ terminates with output b'.

Let b be a bit that the adversary $\mathcal{B}$ has to guess in the $(\mathsf{A}, \widetilde{\mathsf{A}})$-backdoor-sequence experiment.

Note that $\mathcal{B}$ makes exactly the same type of sequence of queries as $\mathcal{A}$, which is A. The advantage of $\mathcal{B}$ is calculated as:

$$\begin{aligned}
\mathsf{Adv}_{BS,\mathcal{B}}^{(\mathsf{A},\widetilde{\mathsf{A}})} &= |\Pr[b' = b] - \frac{1}{2}| \\
&= \frac{1}{2}|\Pr[b' = 1 | b = 1] - \Pr[b' = 1 | b = 0]| \\
&= \frac{1}{2}|\Pr[\gamma' = \gamma | b = 1] - \Pr[\gamma' = \gamma | b = 0]|
\end{aligned}$$

[5]That is, when c is input, if there exists an entry of the form (c, R) in the list L, this R is used as $RF(c)$. Otherwise, a uniformly random value $R \in \mathcal{R}_{BS}$ is picked and (c, R) is added into L for a future reference, and this R is returned as $RF(c)$.

It is easy to see that when $b = 1$ (i.e. $v_b^* = v_1^* = m_\gamma$) $\mathcal{B}$ perfectly simulates Game 3 for $\mathcal{A}$ in which the challenge bit for $\mathcal{A}$ is γ. Specifically, $\mathcal{B}$ is simulating so that the decryption of a ciphertext of the form $C = (\alpha||\beta||w||c^*)$ is answered with $\mathsf{Release}(\mathsf{pub}, \mathsf{pri}, \alpha, \beta)$ where $\mathsf{pri} = \mathsf{BSGen}(\mathsf{pub}, m_\gamma; RF(c^*))$, which is an legitimate response in Game 3. The queries with other types are also answered perfectly by appropriately using sk and the list L. Under this situation, the event $\gamma' = \gamma$ corresponds to the event Succ_3, i.e. $\Pr[\gamma' = \gamma | b = 1] = \Pr[\mathsf{Succ}_3]$.

When $b = 0$ (i.e. $v_b^* = v_0^* = 0^k$), on the other hand, $\mathcal{B}$ perfectly simulates Game 4 for $\mathcal{A}$ in which the challenge bit for $\mathcal{A}$ is γ. With a similar argument to the above, we have $\Pr[\gamma' = \gamma | b = 0] = \Pr[\mathsf{Succ}_4]$.

In summary, we have $\mathsf{Adv}_{BS,\mathcal{B}}^{(\mathsf{A},\widetilde{\mathsf{A}})} = \frac{1}{2}|\Pr[\mathsf{Succ}_3] - \Pr[\mathsf{Succ}_4]|$, which is not negligible by the assumption we made at the beginning of the proof of this claim. However, since $\widetilde{\mathsf{A}} \not\subseteq_{qs} \mathsf{A}$, the existence of such $\mathcal{B}$ contradicts the security of the backdoor-sequence scheme BS. Therefore, $|\Pr[\mathsf{Succ}_3] - \Pr[\mathsf{Succ}_4]|$ must be negligible. This completes the proof of Claim 9. □

Claim 10. $|\Pr[\mathsf{Succ}_4] - \frac{1}{2}|$ *is negligible.*

Proof of Claim 10. Assume towards a contradiction that $|\Pr[\mathsf{Succ}_4] - \frac{1}{2}|$ is not negligible. Then we show that we can construct another PPTA adversary $\mathcal{B} = (\mathcal{B}_1, \mathcal{B}_2)$ that can break the $\langle \mathsf{unbound} :: \mathsf{A} \rangle$-$\mathtt{mCCA}$ security of the underlying PKE scheme Π with non-negligible advantage. The description of $\mathcal{B}$ is as follows.

$\mathcal{B}_1$: On input pk, $\mathcal{B}_1$ sets $\mathsf{pub} = (1^k, \widetilde{\mathsf{A}}, q)$. $\mathcal{B}_1$ then generates an empty list L into which pairs of the form $(c, R_c) \in \{0,1\}^\ell \times \mathcal{R}_{BS}$ will be stored for simulating a truly random function RF in a lazy sampling manner. Then $\mathcal{B}_1$ runs $\mathcal{A}_1$ with input $PK = (pk, \mathsf{pub})$. All the (single) queries C from $\mathcal{A}_1$ are answered by following the decryption procedure in Game 3 (which is identical to that in Game 4), except that when $\mathcal{B}_1$ has to run $\mathsf{PDec}(sk, c)$, $\mathcal{B}_1$ use its own (single) decryption oracle, and to compute a random function $RF(c)$, $\mathcal{B}_1$ simulates it by a lazy sampling using the list L (as is done in the proof of Claim 9).

When $\mathcal{A}_1$ terminates with output two plaintexts (m_0, m_1) of equal length and state information st_A, $\mathcal{B}_1$ sets its own state information $\mathsf{st}_\mathcal{B}$ that consists of all the values known to $\mathcal{B}_1$, and terminates with output $(m_0, m_1, \mathsf{st}_\mathcal{B})$.

$\mathcal{B}_2$: On input $(c^*, \mathsf{st}_\mathcal{B})$, $\mathcal{B}_2$ sets $C^* \leftarrow (0^k||0^k||0^k||c^*)$ and runs $\mathcal{A}_2$ with input $(C^*, \mathsf{st}_\mathcal{A})$. If $\mathcal{A}_1$ has made at least one (single) decryption query of the form $C = (\alpha||\beta||w||c^*)$, then $\mathcal{B}_2$ responds to all the decryption queries from $\mathcal{A}_2$ with $\perp$.

Otherwise, $\mathcal{B}_2$ responds to the decryption queries from $\mathcal{A}_2$ as follows.

- If $\mathcal{A}_2$'s query is a single query $C = (\alpha||\beta||w||c)$, $\mathcal{B}_2$ submits c to its own decryption oracle and receives an answer m from the oracle only if $c \neq c^*$. If $(\alpha||\beta||w) = (0^k||0^k||0^k)$, then $\mathcal{B}_2$ returns m to $\mathcal{A}$. If $c = c^*$, then set $m \leftarrow 0^k$. Finally, $\mathcal{B}_2$ simulates $R \leftarrow RF(c)$ as above, computes $\mathsf{pri} \leftarrow \mathsf{BSGen}(\mathsf{pub}, m; R)$ and returns $y \leftarrow \mathsf{Release}(\mathsf{pub}, \mathsf{pri}, \alpha, \beta, w)$ to $\mathcal{A}_2$. (In this process, if $\mathcal{B}_2$ has not used its own oracle, $\mathcal{B}_2$ makes some query to its own oracle to ensure that the query sequence of $\mathcal{A}$ and that of $\mathcal{B}$ remain the same.)
- If $\mathcal{A}_2$'s query is a parallel query $\overrightarrow{C} = (C_1, C_2, \dots)$, where $C_i = (\alpha_i||\beta_i||w_i||c_i)$ for each $1 \leq i \leq |\overrightarrow{C}|$, $\mathcal{B}_2$ sets $\overrightarrow{c} \leftarrow (c_1, c_2, \dots)$ where if $c_i = c^*$ for some $i \in [n]$, c_i

is replaced with $\perp$. Next, $\mathcal{B}_2$ submits a parallel query $\overrightarrow{c}$ to its decryption oracle and receives an answer $\overrightarrow{m} = (m_1, m_2, \dots)$. Here, if $c_i = c^*$ for some $i \in [n]$, the corresponding plaintext m_i in $\overrightarrow{m}$ is replaced with 0^k. Then, for each $1 \leq i \leq |\overrightarrow{C}|$, $\mathcal{B}_2$ sets $m_i \leftarrow y'_i$ if $(\alpha_i||\beta_i||w_i) \neq (0^k||0^k||0^k)$, where $y'_i \leftarrow \mathsf{Release}(\mathsf{pub}, \mathsf{pri}_i, \alpha_i, \beta_i, w_i)$, $\mathsf{pri}_i \leftarrow \mathsf{BSGen}(\mathsf{pub}, m_i; R_i)$, and $R_i \leftarrow RF(c_i)$ (this is simulated by a lazy sampling using L as above). Finally, $\mathcal{B}_2$ sends $(m_1, m_2, \dots)$ back to $\mathcal{A}_2$.

When $\mathcal{A}_2$ terminates with output b', $\mathcal{B}_2$ outputs this b' and terminates.

Note that $\mathcal{B}$ makes exactly the same type of the query sequence as $\mathcal{A}$, and thus $\mathcal{B}$ is a legitimate $\langle \mathsf{unbound} :: \mathsf{A} \rangle$-mCCA adversary regarding Π. Moreover, it is easy to see that $\mathcal{B}$ does the perfect simulation of Game 4 for $\mathcal{A}$, and in particular, the challenge bit for $\mathcal{B}$ is that for $\mathcal{A}$. Therefore, $\mathcal{B}$ succeeds in guessing the challenge bit whenever $\mathcal{A}$ does so, and we have $\mathsf{Adv}^{\langle \mathsf{unbound}::\mathsf{A} \rangle\text{-mCCA}}_{\Pi,\mathcal{B}} = |\Pr[\mathsf{Succ}_4] - \frac{1}{2}|$, but this is not negligible by the assumption we made at the beginning of the proof of this claim. Since this contradicts the $\langle \mathsf{unbound} :: \mathsf{A} \rangle$-mCCA security of the underlying PKE scheme Π, $|\Pr[\mathsf{Succ}_4] - \frac{1}{2}|$ must be negligible. This completes the proof of Claim 10. □

According to the inequality (4.4) and Claims 7 to 10, we can upperbound $\mathsf{Adv}^{\langle \mathsf{unbound}::\mathsf{A} \rangle\text{-mCCA}}_{\Pi_{\mathsf{sep2}},\mathcal{A}}$ to be negligible for any PPTA adversary $\mathcal{A}$. This completes the proof of Lemma 14. □

Lemmas 13 and 14 imply Theorem 12. □

An important corollary of Theorem 12 is the following.

Corollary 1. *For both PKE schemes and KEMs, if* $(\widetilde{\mathsf{F}}||\widetilde{\mathsf{A}}) \not\subseteq_{qs} (\mathsf{F}||\mathsf{A})$*, then* $\langle \mathsf{B} : \mathsf{F} : \mathsf{A} \rangle$-mCCA *security does not imply* $\langle \widetilde{\mathsf{B}} : \widetilde{\mathsf{F}} : \widetilde{\mathsf{A}} \rangle$-mCCA *security.*

Proof. Let $\mathsf{seq} = (\mathsf{F}||\mathsf{A})$ and $\widetilde{\mathsf{seq}} = (\widetilde{\mathsf{F}}||\widetilde{\mathsf{A}})$, and consider $\langle \mathsf{unbound} :: \mathsf{seq} \rangle$-mCCA security and $\langle \emptyset :: \widetilde{\mathsf{seq}} \rangle$-mCCA security. By Theorem 12 and the given condition $\widetilde{\mathsf{seq}} \not\subseteq_{qs} \mathsf{seq}$, $\langle \mathsf{unbound} :: \mathsf{seq} \rangle$-mCCA security does not imply $\langle \emptyset :: \widetilde{\mathsf{seq}} \rangle$-mCCA security. However, $\langle \mathsf{unbound} :: \mathsf{seq} \rangle$-mCCA security implies $\langle \mathsf{B} : \mathsf{F} : \mathsf{A} \rangle$-mCCA security while $\langle \widetilde{\mathsf{B}} : \widetilde{\mathsf{F}} : \widetilde{\mathsf{A}} \rangle$-mCCA security implies $\langle \emptyset :: \widetilde{\mathsf{seq}} \rangle$-mCCA security. Since an implication of a security notion from another is a transitive relation, $\langle \mathsf{B} : \mathsf{F} : \mathsf{A} \rangle$-mCCA security cannot imply $\langle \widetilde{\mathsf{B}} : \widetilde{\mathsf{F}} : \widetilde{\mathsf{A}} \rangle$-mCCA security. □

Finally, by noticing the difference in the "before-challenge" queries, we show the following separation which is true only for PKE schemes with superpolynomially large plaintext space size.

Theorem 13. *For PKE schemes with superpolynomially large plaintext space size, if* $\widetilde{\mathsf{B}} \not\subseteq_{qs} \mathsf{B}$*, then* $\langle \mathsf{B} :: \mathsf{unbound} \rangle$-mCCA *security does not imply* $\langle \widetilde{\mathsf{B}} :: \emptyset \rangle$-mCCA *security.*

Intuition. This time, the critical information that can be used to break $\langle \widetilde{\mathsf{B}} :: \emptyset \rangle$-mCCA security must be something that is useful only before the challenge, because $\langle \widetilde{\mathsf{B}} :: \emptyset \rangle$-mCCA adversary can make no query after the challenge. We design the separating PKE scheme so that it has "weak" plaintexts, which are not encrypted at all by the encryption algorithm of the separating PKE scheme. Then, we set the critical information to be one of these weak plaintexts, which can be used as one of two challenge plaintexts. However, such weak plaintexts must be hard to find, because otherwise $\langle \mathsf{B} :: \mathsf{unbound} \rangle$-mCCA adversary can also find such a weak

$\mathsf{PKG}_{\mathsf{sep3}}(1^k)$:	$\mathsf{PEnc}_{\mathsf{sep3}}(PK, m)$:
$(pk, sk) \leftarrow \mathsf{PKG}(1^k)$	Parse PK as (pk, pub, V).
$q \leftarrow \lvert\mathsf{B}\rvert + 1$	If $f(m) = V$ then return $C \leftarrow (1 \lvert\rvert 0^k \lvert\rvert 0^k \lvert\rvert m)$.
$\mathsf{pub} \leftarrow (1^k, \widetilde{\mathsf{B}}, q)$	$c \leftarrow \mathsf{PEnc}(pk, m)$
$v \leftarrow \mathcal{M}_\Pi$	Return $C \leftarrow (0 \lvert\rvert 0^k \lvert\rvert 0^k \lvert\rvert c)$.
$V \leftarrow f(v)$	
$\mathsf{pri} \leftarrow \mathsf{BSGen}(\mathsf{pub}, v)$	
$PK \leftarrow (pk, \mathsf{pub}, V)$	
$SK \leftarrow (sk, \mathsf{pri})$	
Return (PK, SK).	

$\mathsf{PDec}_{\mathsf{sep3}}(SK, C)$:
Parse SK as (sk, pri) and C as $(\gamma \lvert\rvert \alpha \lvert\rvert \beta \lvert\rvert c)$ such that $\lvert\gamma\rvert = 1$ and $\lvert\alpha\rvert = \lvert\beta\rvert = k$.
If $(\gamma \lvert\rvert \alpha \lvert\rvert \beta) = (0 \lvert\rvert 0^k \lvert\rvert 0^k)$ then return $\mathsf{PDec}(sk, c)$.
If $(\gamma \lvert\rvert \alpha \lvert\rvert \beta) = (1 \lvert\rvert 0^k \lvert\rvert 0^k)$ and $f(c) = V$ then return c.
Return $\mathsf{Release}(\mathsf{pub}, \mathsf{pri}, \alpha, \beta, c)$
(If c is longer than k-bit, then use the k-most significant bits of c.)

Figure 4.4: The PKE scheme Π_{sep3} that separates $\langle\widetilde{\mathsf{B}} :: \emptyset\rangle$-mCCA from $\langle\mathsf{B} :: \mathsf{unbound}\rangle$-mCCA in case $\widetilde{\mathsf{B}} \not\subseteq_{qs} \mathsf{B}$ and a plaintext space size is superpolynomially large.

plaintext before the challenge. Moreover, such weak plaintexts must be easy to recognize without any secret information so that the encryption algorithm of the separating scheme can tell if a given plaintext is a weak plaintext or not.

In order to deal with such a situation, we use a one-way function f. A public key of the separating PKE scheme contains $V = f(m^*)$ for some random element m in the plaintext space of the underlying PKE scheme, and weak plaintexts m are the ones satisfying $f(m) = V$. We set the critical information to be m^* itself, and implement a backdoor mechanism so that m^* is accessible before the challenge only by a $\langle\widetilde{\mathsf{B}} :: \emptyset\rangle$-mCCA adversary. Then, this weak plaintext m^* cannot be found by a $\langle\mathsf{B} :: \mathsf{unbound}\rangle$-mCCA adversary before the challenge, and is a useless value even if it is found after the challenge.

Proof. In order to show the statement, we will show that if there exists a $\langle\mathsf{B} :: \mathsf{unbound}\rangle$-mCCA secure PKE scheme $\Pi = (\mathsf{PKG}, \mathsf{PEnc}, \mathsf{PDec})$, then there exists a PKE scheme $\Pi_{\mathsf{sep3}} = (\mathsf{PKG}_{\mathsf{sep3}}, \mathsf{PEnc}_{\mathsf{sep3}}, \mathsf{PDec}_{\mathsf{sep3}})$ which is $\langle\mathsf{B} :: \mathsf{unbound}\rangle$-mCCA secure but is not $\langle\widetilde{\mathsf{B}} :: \emptyset\rangle$-mCCA secure.

Specifically, let $\mathcal{M}_\Pi$ be the plaintext space of Π, and let $q = \lvert\mathsf{B}\rvert + 1$. We use the backdoor-sequence scheme $BS = (\mathsf{BSGen}, \mathsf{Release}, \mathsf{Recon})$ and a one-way function $f : \mathcal{M}_\Pi \to \{0,1\}^*$ as building blocks and construct the separating PKE scheme Π_{sep3} as in Fig. 4.4. We note that a backdoor-sequence scheme can be constructed without any computational intractability assumption, the existence of a one-way function is implied by the existence of a $\langle\mathsf{B} :: \mathsf{unbound}\rangle$-mCCA secure PKE scheme.

Without loss of generality, we assume that all the integers that appear in this proof have k-bit representation. In order to clarify that we are treating an integer as a k-bit string, we will use "hat". For example, $\widehat{1}$ is a k-bit representation of 1.

In the following, we show two lemmas that imply Theorem 13.

Lemma 15. *The PKE scheme Π_{sep3} is not $\langle\widetilde{\mathsf{B}} :: \emptyset\rangle$-mCCA secure.*

Proof of Lemma 15. We construct a PPTA adversary $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$ as follows:

$\mathcal{A}_1$**:** Let $\widetilde{\mathsf{B}} = (b_1 \dots b_n)$ where $b_i \in \{\mathsf{s}, \mathsf{p}\}$ for each $i \in [n]$. On input $PK = (pk, \mathsf{pub}, V)$, $\mathcal{A}_1$ sets $u_1 \leftarrow 1^k$ and then repeats the following for $1 \leq i \leq n$:

- if $b_i = \mathsf{s}$, then $\mathcal{A}_1$ issues $C = (0||\widehat{i}||\widehat{1}||u_i)$ as i-th (single) decryption query and is given $u_{i+1} = \mathsf{Release}(\mathsf{pub}, \mathsf{pri}, i, 1, u_i)$ as an answer.
- if $b_i = \mathsf{p}$, then $\mathcal{A}_1$ issues, as its i-th query, a parallel query $(C_1, C_2, \dots, C_q)$ such that $C_j = (0||\widehat{i}||\widehat{j}||u_i)$ for each $1 \leq j \leq q$, and is given $(y_1, \dots, y_q)$ as an answer, where $y_j = \mathsf{Release}(\mathsf{pub}, \mathsf{pri}, i, j, u_i)$ for every $1 \leq j \leq q$. Then $\mathcal{A}_1$ runs $u_{i+1} \leftarrow \mathsf{Recon}(\mathsf{pub}, (y_1, \dots, y_q))$.

Note that in both cases, the returned value(s) from the oracle is always the output of $\mathsf{Release}$, due to the design of the PKE scheme $\Pi_{\mathtt{sep3}}$. Moreover, since u_{i+1} can be always obtained from the response to i-th query, $\mathcal{A}_1$ can continue the above. Therefore, after $\mathcal{A}_1$ completes all the queries following $\widetilde{\mathsf{B}}$ as above, $\mathcal{A}_1$ obtains (or can compute) $u_{n+1} = v$. Then, $\mathcal{A}_1$ sets $m_0 = v$ and pick some value $m_1 \in \mathcal{M}_\Pi$ such that $m_1 \neq v$, and sets $\mathsf{st}_{\mathcal{A}}$ that consists of all the values known to $\mathcal{A}_1$. Finally, $\mathcal{A}_1$ terminates with output $(m_0, m_1, \mathsf{st}_{\mathcal{A}})$.

$\mathcal{A}_2$**:** On input $(C^*, \mathsf{st}_{\mathcal{A}})$ where $C^* = (\gamma||0^k||0^k||c^*)$ and $\gamma \in \{0, 1\}$, $\mathcal{A}_2$ sets $b' \leftarrow 0$ if $\gamma = 1$ and $c^* = v$ hold, or sets $b' \leftarrow 1$ otherwise. Then $\mathcal{A}_2$ outputs b' as its guess and terminates.

It is clear that $\mathcal{A}$ succeeds in guessing the challenge bit b with probability 1, and thus always has $\langle \widetilde{\mathsf{B}} :: \emptyset \rangle$-$\mathtt{mCCA}$ advantage $\frac{1}{2}$, which is non-negligible. This completes the proof of Lemma 15. □

Lemma 16. *If the underlying PKE scheme* Π *is* $\langle \mathsf{B} :: \mathsf{unbound} \rangle$-$\mathtt{mCCA}$ *secure, the backdoor-sequence scheme BS is secure, and f is a one-way function, then the PKE scheme* $\Pi_{\mathtt{sep3}}$ *is* $\langle \mathsf{B} :: \mathsf{unbound} \rangle$-$\mathtt{mCCA}$ *secure.*

Proof of Lemma 16. Let $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$ be any PPTA adversary that attacks the PKE scheme $\Pi_{\mathtt{sep3}}$ in the sense of $\langle \mathsf{B} :: \mathsf{unbound} \rangle$-$\mathtt{mCCA}$ security. Consider the following sequence of games.

Game 1 This is the ordinary $\langle \mathsf{B} :: \mathsf{unbound} \rangle$-$\mathtt{mCCA}$ experiment that $\mathcal{A}$ runs in.

Game 2 Same as Game 1, except that the input v to the BSGen algorithm of the backdoor-sequence scheme run in $\mathsf{PKG}_{\mathtt{sep3}}$ is replaced with 0^k.

Let Succ_i denote the event that $\mathcal{A}$ succeeds in guessing the challenge bit in Game i, and let Invert_i be the event that $\mathcal{A}_1$ outputs two plaintexts (m_0, m_1) such that $f(m_0) = V$ or

$f(m_1) = V$. Then, the advantage of an adversary $\mathcal{A}$ is calculated as:

$$
\begin{aligned}
&\mathsf{Adv}^{\langle \mathsf{B}::\mathsf{unbound}\rangle\text{-}\mathtt{mCCA}}_{\Pi_{\mathsf{sep3}},\mathcal{A}} = |\Pr[\mathsf{Succ}_1] - \frac{1}{2}| \\
&= |\Pr[\mathsf{Succ}_1 \wedge \overline{\mathsf{Invert}_1}] + \Pr[\mathsf{Succ}_1|\mathsf{Invert}_1] \cdot \Pr[\mathsf{Invert}_1] - \frac{1}{2}| \\
&= |\Pr[\mathsf{Succ}_1 \wedge \overline{\mathsf{Invert}_1}] + \Pr[\mathsf{Succ}_1|\mathsf{Invert}_1] \cdot \Pr[\mathsf{Invert}_1] - \frac{1}{2} + \frac{1}{2}\Pr[\mathsf{Invert}_1] - \frac{1}{2}\Pr[\mathsf{Invert}_1]| \\
&= |\Pr[\mathsf{Succ}_1 \wedge \overline{\mathsf{Invert}_1}] + \frac{1}{2}\Pr[\mathsf{Invert}_1] - \frac{1}{2} + (\Pr[\mathsf{Succ}_1|\mathsf{Invert}_1] - \frac{1}{2}) \cdot \Pr[\mathsf{Invert}_1]| \\
&\leq |\Pr[\mathsf{Succ}_1 \wedge \overline{\mathsf{Invert}_1}] + \frac{1}{2}\Pr[\mathsf{Invert}_1] - \frac{1}{2}| + |\Pr[\mathsf{Succ}_1|\mathsf{Invert}_1] - \frac{1}{2}| \cdot \Pr[\mathsf{Invert}_1] \\
&\leq |\Pr[\mathsf{Succ}_1 \wedge \overline{\mathsf{Invert}_1}] + \frac{1}{2}\Pr[\mathsf{Invert}_1] - \frac{1}{2}| + \frac{1}{2}\Pr[\mathsf{Invert}_1] \\
&\leq |\Pr[\mathsf{Succ}_1 \wedge \overline{\mathsf{Invert}_1}] + \frac{1}{2}\Pr[\mathsf{Invert}_1] - \frac{1}{2}| + \frac{1}{2}|\Pr[\mathsf{Invert}_1] - \Pr[\mathsf{Invert}_2]| + \frac{1}{2}\Pr[\mathsf{Invert}_2]
\end{aligned}
\tag{4.5}
$$

To upperbound the above advantage, we show the following claims.

Claim 11. $|\Pr[\mathsf{Succ}_1 \wedge \overline{\mathsf{Invert}_1}] + \frac{1}{2}\Pr[\mathsf{Invert}_1] - \frac{1}{2}]|$ *is negligible.*

Proof of Claim 11. Assume towards a contradiction that $|\Pr[\mathsf{Succ}_1 \wedge \overline{\mathsf{Invert}_1}] + \frac{1}{2}\Pr[\mathsf{Invert}_1] - \frac{1}{2}]|$ is not negligible. Then we show that we can construct another PPTA adversary $\mathcal{B} = (\mathcal{B}_1, \mathcal{B}_2)$ that can break the $\langle \mathsf{B} :: \mathsf{unbound}\rangle$-$\mathtt{mCCA}$ security of the underlying PKE scheme Π with non-negligible advantage. The description of $\mathcal{B}$ is as follows.

$\mathcal{B}_1$**:** On input pk, $\mathcal{B}_1$ picks $v \in \mathcal{M}_\Pi$ uniformly at random and computes $f(v) = V$. $\mathcal{B}_1$ next sets $\mathsf{pub} = (1^k, \widetilde{\mathsf{B}}, q)$, and runs $\mathsf{pri} \leftarrow \mathsf{BSGen}(\mathsf{pub}, v)$. Then $\mathcal{B}_1$ runs $\mathcal{A}_1$ with input $PK = (pk, \mathsf{pub}, V)$.

When $\mathcal{A}_1$ issues decryption queries, $\mathcal{B}_1$ responds as follows.

- If $\mathcal{A}_1$'s query is a single query $C = (\gamma||\alpha||\beta||c)$, then $\mathcal{B}_1$ issues c to its decryption oracle and obtains m. Next, if $(\gamma||\alpha||\beta) = (1||0^k||0^k)$ and $f(c) = V$, then $\mathcal{B}_1$ sets $m \leftarrow c$. Furthermore, if $(\alpha||\beta) \neq (0^k||0^k)$, $\mathcal{B}_1$ sets $m \leftarrow \mathsf{Release}(\mathsf{pub}, \mathsf{pri}, \alpha, \beta, c)$. Finally $\mathcal{B}_1$ sends m back to $\mathcal{A}_1$.
- If $\mathcal{A}_1$'s query is a parallel query $\overrightarrow{C} = (C_1, C_2, \dots)$, where $C_i = (\gamma_i||\alpha_i||\beta_i||c_i)$ for each $1 \leq i \leq |\overrightarrow{C}|$, $\mathcal{B}_1$ makes a parallel query $\overrightarrow{c} = (c_1, c_2, \dots)$ to $\mathcal{B}$'s decryption oracle and receives the answer $(m_1, m_2, \dots)$. Then for each $1 \leq i \leq |\overrightarrow{C}|$, $\mathcal{B}_1$ sets $m_i \leftarrow c_i$ if $(\gamma_i||\alpha_i||\beta_i) = (1||0^k||0^k)$ and $f(c_i) = V$. Furthermore, for each $1 \leq i \leq |\overrightarrow{C}|$, $\mathcal{B}_1$ sets $m_i \leftarrow \mathsf{Release}(\mathsf{pub}, \mathsf{pri}, \alpha_i, \beta_i, c_i)$ if $(\alpha_i||\beta_i) \neq (0^k||0^k)$. Finally $\mathcal{B}_2$ sends $(m_1, m_2, \dots)$ back to $\mathcal{A}_1$.

When $\mathcal{A}_1$ terminates with output two plaintexts (m_0, m_1) of equal length and state information st_A, If $f(m_0) = V$ or $f(m_1) = V$, then $\mathcal{B}_1$ sets a state information $\mathsf{st}_\mathcal{B}$ that will tell $\mathcal{B}_2$ that $\mathcal{B}_1$ has given up. Otherwise $\mathcal{B}_1$ sets its own state information $\mathsf{st}_\mathcal{B}$ that consists of all the values known to $\mathcal{B}_1$. Finally, $\mathcal{B}_1$ terminates with output $(m_0, m_1, \mathsf{st}_\mathcal{B})$.

$\mathcal{B}_2$**:** On input $(c^*, \mathsf{st}_{\mathcal{B}})$, $\mathcal{B}_2$ first checks if $\mathcal{B}_1$ has given up by looking at $\mathsf{st}_{\mathcal{B}}$. If this is the case, $\mathcal{B}_2$ picks a random coin $b' \leftarrow \{0,1\}$ and terminates with output b'. Otherwise, $\mathcal{B}_2$ sets $C^* \leftarrow (0||0^k||0^k||c^*)$ and runs $\mathcal{A}_2$ with input $(C^*, \mathsf{st}_{\mathcal{A}})$. By definition, $\mathcal{A}_2$'s queries are always single queries, and are answered in exactly the same way as the queries from $\mathcal{A}_1$, except that if $\mathcal{A}_2$'s query contains a ciphertext of the form $C = (\gamma||\alpha||\beta||c^*)$, $\mathcal{B}_2$ does not submit it to the decryption oracle. Note that such ciphertext $C = (\gamma||\alpha||\beta||c^*)$ cannot satisfy $(\gamma||\alpha||\beta) = (0||0^k||0^k)$ by the definition of the $(\mathsf{B} :: \mathsf{unbound})$-$\mathtt{mCCA}$ experiment, and thus to decrypt C as a ciphertext of $\Pi_{\mathtt{sep3}}$, $\mathsf{PDec}(sk, c^*)$ need not be run, and thus $\mathcal{B}_2$ can appropriately compute the decryption of C with the knowledge about pri. When $\mathcal{A}_2$ terminates with output b', $\mathcal{B}_2$ outputs this b' and terminates.

Let $\mathsf{Succ}_{\mathcal{B}}$ be the event that $\mathcal{B}$ succeeds in guessing the challenge bit, and $\mathsf{Abort}_{\mathcal{B}}$ be the event that in $\mathcal{B}$'s experiment, $\mathcal{B}_1$ gives up and $\mathcal{B}_2$ outputs a random bit.

Note that $\mathcal{B}$ makes exactly the same type of the query sequence as $\mathcal{A}$, and thus $\mathcal{B}$ is a legitimate $\langle \mathsf{B} :: \mathsf{unbound} \rangle$-$\mathtt{mCCA}$ adversary regarding Π. Moreover, it is easy to see that $\mathcal{B}$ does the perfect simulation of Game 1 for $\mathcal{A}$ in which the challenge bit for $\mathcal{A}$ is that of $\mathcal{B}$ unless $\mathcal{A}_1$'s challenge plaintexts (m_0, m_1) satisfies $f(m_0) = V$ or $f(m_1) = V$. This implies $\Pr[\mathsf{Succ}_{\mathcal{B}} \wedge \overline{\mathsf{Abort}_{\mathcal{B}}}] = \Pr[\mathsf{Succ}_1 \wedge \overline{\mathsf{Invert}_1}]$ and $\Pr[\mathsf{Abort}_{\mathcal{B}}] = \Pr[\mathsf{Invert}_1]$. Furthermore, if $\mathcal{A}_1$ outputs such challenge plaintexts, then $\mathsf{Invert}_{\mathcal{B}}$ occurs, and thus $\mathcal{B}$ gives up and outputs a random bit, which implies $\Pr[\mathsf{Succ}_{\mathcal{B}} | \mathsf{Abort}_{\mathcal{B}}] = \frac{1}{2}$.

Using the above, we can now estimate $\mathcal{B}$'s advantage as:

$$\begin{aligned}
\mathsf{Adv}^{\langle \mathsf{B}::\mathsf{unbound}\rangle\text{-}\mathtt{mCCA}}_{\Pi,\mathcal{B}} &= |\Pr[\mathsf{Succ}_{\mathcal{B}}] - \frac{1}{2}| \\
&= |\Pr[\mathsf{Succ}_{\mathcal{B}} \wedge \overline{\mathsf{Abort}_{\mathcal{B}}}] + \Pr[\mathsf{Succ}_{\mathcal{B}} | \mathsf{Abort}_{\mathcal{B}}] \cdot \Pr[\mathsf{Abort}_{\mathcal{B}}] - \frac{1}{2}| \\
&= |\Pr[\mathsf{Succ}_1 \wedge \overline{\mathsf{Invert}_1}] + \frac{1}{2}\Pr[\mathsf{Invert}_1] - \frac{1}{2}|
\end{aligned}$$

This means that $\mathsf{Adv}^{\langle \mathsf{B}::\mathsf{unbound}\rangle\text{-}\mathtt{mCCA}}_{\Pi,\mathcal{B}}$ is non-negligible by the assumption we made at the beginning of the proof of this claim. Since this contradicts the $\langle \mathsf{B} :: \mathsf{unbound} \rangle$-$\mathtt{mCCA}$ security of the underlying PKE scheme Π, $|\Pr[\mathsf{Succ}_1 \wedge \overline{\mathsf{Invert}_1}] + \frac{1}{2}\Pr[\mathsf{Invert}_1] - \frac{1}{2}|$ must be negligible. This completes the proof of Claim 11. □

Claim 12. $|\Pr[\mathsf{Invert}_1] - \Pr[\mathsf{Invert}_2]|$ *is negligible.*

Proof of Claim 12. Assume towards a contradiction that $|\Pr[\mathsf{Invert}_1] - \Pr[\mathsf{Invert}_2]|$ is not negligible. Then we show that we can construct another PPTA adversary $\mathcal{B} = (\mathcal{B}_1, \mathcal{B}_2)$ that has non-negligible advantage in the $(\mathsf{B}, \widetilde{\mathsf{B}})$-backdoor-sequence experiment regarding $BS = (\mathsf{BSGen}, \mathsf{Release}, \mathsf{Recon})$.

The description of $\mathcal{B}$ is as follows.

$\mathcal{B}_1$**:** On input $\mathsf{pub} = (1^k, \widetilde{\mathsf{B}}, q)$, $\mathcal{B}_1$ runs $(pk, sk) \leftarrow \mathsf{PKG}(1^k)$. Next, $\mathcal{B}_1$ next picks $v \leftarrow \mathcal{M}_{\Pi}$ uniformly at random and computes $V \leftarrow f(v)$. Then, $\mathcal{B}_1$ sets $v_0^* = 0^k$, $v_1^* = v$, and state information $\mathsf{st}_{\mathcal{B}}$ that consists of all the values known to $\mathcal{B}_1$, and terminates with output $(v_0^*, v_1^*, \mathsf{st}_{\mathcal{B}})$.

$\mathcal{B}_2$**:** On input $\mathsf{st}_{\mathcal{B}}$, $\mathcal{B}_2$ runs $\mathcal{A}_1$ with input $PK = (pk, \mathsf{pub}, V)$. When $\mathcal{A}_1$ issues decryption queries, $\mathcal{B}_2$ responds as follows.

- If $\mathcal{A}_1$'s query is a single query $C = (\gamma||\alpha||\beta||c)$, $\mathcal{B}_2$ first runs $m \leftarrow \mathsf{PDec}(sk, c)$. Next, $\mathcal{B}_2$ issues a single query (α, β, c) to $\mathcal{B}$'s own oracle and receives the result y. Then, $\mathcal{B}_1$ sets $m \leftarrow c$ if $(\gamma||\alpha||\beta) = (1||0^k||0^k)$ and $f(c) = V$. Furthermore, $\mathcal{B}_1$ sets $m \leftarrow y$ if $(\alpha||\beta) \neq (0^k||0^k)$. Finally, $\mathcal{B}_1$ sends m back to $\mathcal{A}_1$.
- If $\mathcal{A}_1$'s query is a parallel query $\overrightarrow{C} = (C_1, C_2, \dots)$, where $C_i = (\gamma_i||\alpha_i||\beta_i||c_i)$ for each $1 \leq i \leq |\overrightarrow{C}|$, $\mathcal{B}_2$ first computes $m_i \leftarrow \mathsf{PDec}(sk, c_i)$ for every $1 \leq i \leq |\overrightarrow{C}|$. Then $\mathcal{B}_2$ also issues $((\alpha_1, \beta_1, c_1), (\alpha_2, \beta_2, c_2), \dots)$ as a parallel query to $\mathcal{B}$'s own oracle and receives the result $(y_1, y_2, \dots)$. Then, for each $1 \leq i \leq |\overrightarrow{C}|$, $\mathcal{B}_2$ sets $m_i \leftarrow c_i$ if $(\gamma_i||\alpha_i||\beta_i) = (1||0^k||0^k)$ and $f(c_i) = V$. Furthermore, for each $1 \leq i \leq |\overrightarrow{C}|$, $\mathcal{B}_2$ sets $m_i \leftarrow y_i$ if $(\alpha||\beta) \neq (0^k||0^k)$. Finally, $\mathcal{B}_2$ sends $(m_1, m_2, \dots)$ back to $\mathcal{A}_1$.

When $\mathcal{A}_1$ terminates with output two plaintexts (m_0, m_1) of equal length and state information $\mathsf{st}_{\mathcal{A}}$, $\mathcal{B}_2$ checks if $f(m_0) = V$ or $f(m_1) = V$ holds. If this is the case, $\mathcal{B}_2$ sets $b' \leftarrow 1$ or sets $b' \leftarrow 0$ otherwise. Finally, $\mathcal{B}_2$ terminates with output b'.

Let b be a bit that the adversary $\mathcal{B}$ has to guess in the $(\mathsf{B}, \widetilde{\mathsf{B}})$-backdoor-sequence experiment.

Note that $\mathcal{B}$ makes exactly the same type of sequence of queries as $\mathcal{A}$, which is B. The advantage of $\mathcal{B}$ is calculated as:

$$\mathsf{Adv}^{(\mathsf{B},\widetilde{\mathsf{B}})}_{BS,\mathcal{B}} = |\Pr[b' = b] - \frac{1}{2}| = \frac{1}{2}|\Pr[b' = 1|b = 1] - \Pr[b' = 1|b = 0]|$$

It is easy to see that when $b = 1$, $\mathcal{B}$ perfectly simulates Game 1 for $\mathcal{A}_1$ (i.e. before the challenge). In particular, the secret value input to BSGen in $\mathcal{B}$'s experiment is $v^*_b = v^*_1 = v$ such that $f(v) = V$, which is exactly the procedure done in $\mathsf{PKG}_{\mathsf{sep3}}$ in Game 1. Under this situation, the event that $\mathcal{A}_1$ outputs two plaintexts (m_0, m_1) satisfying $f(m_0) = V$ or $f(m_1) = V$ exactly corresponds to the event Invert_1, i.e. $\Pr[b' = 1|b = 1] = \Pr[\mathsf{Invert}_1]$.

When $b = 0$, on the other hand, the secret value input to BSGen in $\mathcal{B}$'s experiment is $v^*_b = v^*_0 = 0^k$, and $\mathcal{B}$ perfectly simulates Game 2 for $\mathcal{A}_1$. With a similar argument to the above, we have $\Pr[b' = 1|b = 0] = \Pr[\mathsf{Succ}_2]$.

In summary, we have $\mathsf{Adv}^{(\mathsf{B},\widetilde{\mathsf{B}})}_{BS,\mathcal{B}} = \frac{1}{2}|\Pr[\mathsf{Invert}_1] - \Pr[\mathsf{Invert}_2]|$, which is not negligible by the assumption we made at the beginning of the proof of this claim. However, since $\widetilde{\mathsf{B}} \not\subseteq_{qs} \mathsf{B}$, the existence of such $\mathcal{B}$ contradicts the security of the backdoor-sequence scheme BS. Therefore, $|\Pr[\mathsf{Invert}_1] - \Pr[\mathsf{Invert}_2]|$ must be negligible. This completes the proof of Claim 12. □

Claim 13. $\Pr[\mathsf{Invert}_2]$ *is negligible.*

Proof of Claim 13. Assume towards a contradiction that $\Pr[\mathsf{Invert}_2]$ is not negligible. Then we show that we can construct another PPTA adversary $\mathcal{B}$ that can break one-wayness of f with non-negligible advantage. The description of $\mathcal{B}$ is as follows.

$\mathcal{B}$**:** On input 1^k and $V = f(v)$ where $v \in \mathcal{M}_\Pi$ is a randomly chosen value and unknown to $\mathcal{B}$, $\mathcal{B}$ runs $(pk, sk) \leftarrow \mathsf{PKG}(1^k)$. Moreover, $\mathcal{B}$ sets $\mathsf{pub} \leftarrow (1^k, \widetilde{\mathsf{B}}, q)$ and runs $\mathsf{pri} \leftarrow \mathsf{BSGen}(\mathsf{pub}, 0^k)$. Then $\mathcal{B}$ runs $\mathcal{A}_1$ with input $PK = (pk, \mathsf{pub}, V)$. Since $\mathcal{B}$ owns sk and pri, all the decryption queries from $\mathcal{A}_1$ can be answered perfectly as in Game 2. When $\mathcal{A}_1$ terminates with two plaintexts (m_0, m_1) of equal length and state information $\mathsf{st}_{\mathcal{B}}$, $\mathcal{B}$ checks if $f(m_b) = V$ for some $b \in \{0, 1\}$. If such b exists, $\mathcal{B}$ outputs m_b and terminates. Otherwise, $\mathcal{B}$ simply gives up and aborts.

It is easy to see that $\mathcal{B}$ perfectly simulates Game 2 for $\mathcal{A}_1$ and can output the preimage of V under f whenever Invert_2 occurs. Therefore, the advantage of $\mathcal{B}$ is exactly $\Pr[\mathsf{Invert}_2]$, which is non-negligible by the assumption we made at the beginning of the proof of this claim. Since this contradicts one-wayness of f, $\Pr[\mathsf{Invert}_2]$ must be negligible. This completes the proof of Claim 13. □

According to the inequality (4.5) and Claims 11 to 13, we can upperbound $\mathsf{Adv}^{\langle \mathsf{B}::\mathsf{unbound}\rangle\text{-}\mathtt{mCCA}}_{\Pi_{\mathsf{sep3}},\mathcal{A}}$ to be negligible for any PPTA adversary $\mathcal{A}$. This completes the proof of Lemma 16. □

Lemmas 15 and 16 imply Theorem 13. □

An important corollary of Theorem 13 is the following.

Corollary 2. *For PKE schemes with superpolynomially large plaintext space size, if* $(\widetilde{\mathsf{B}}||\widetilde{\mathsf{F}}) \not\subseteq_{qs} (\mathsf{B}||\mathsf{F})$*, then* $\langle \mathsf{B} : \mathsf{F} : \mathsf{A}\rangle$-mCCA *security does not imply* $\langle \widetilde{\mathsf{B}} : \widetilde{\mathsf{F}} : \widetilde{\mathsf{A}}\rangle$-mCCA *security.*

Proof. Let $\mathsf{seq} = (\mathsf{B}||\mathsf{F})$ and $\widetilde{\mathsf{seq}} = (\widetilde{\mathsf{B}}||\widetilde{\mathsf{F}})$, and consider $\langle \mathsf{seq} :: \mathsf{unbound}\rangle$-mCCA security and $\langle \widetilde{\mathsf{seq}} :: \emptyset\rangle$-mCCA security. By Theorem 13 and the given condition $\widetilde{\mathsf{seq}} \not\subseteq_{qs} \mathsf{seq}$, $\langle \mathsf{seq} :: \mathsf{unbound}\rangle$-mCCA security does not imply $\langle \widetilde{\mathsf{seq}} :: \emptyset\rangle$-mCCA security. However, $\langle \mathsf{seq} :: \mathsf{unbound}\rangle$-mCCA security implies $\langle \mathsf{B} : \mathsf{F} : \mathsf{A}\rangle$-mCCA security while $\langle \widetilde{\mathsf{B}} : \widetilde{\mathsf{F}} : \widetilde{\mathsf{A}}\rangle$-mCCA security implies $\langle \widetilde{\mathsf{seq}} :: \emptyset\rangle$-mCCA security. Since an implication of a security notion from another is a transitive relation, $\langle \mathsf{B} : \mathsf{F} : \mathsf{A}\rangle$-mCCA security cannot imply $\langle \widetilde{\mathsf{B}} : \widetilde{\mathsf{F}} : \widetilde{\mathsf{A}}\rangle$-mCCA security. □

4.3.4 Implication Results

Here, we show the implications among mixed CCA security notions.

A combination of Theorem 11 and Corollaries 1 and 2 shows that given two mixed CCA security notions $\langle \mathsf{B} : \mathsf{F} : \mathsf{A}\rangle$-mCCA and $\langle \widetilde{\mathsf{B}} : \widetilde{\mathsf{F}} : \widetilde{\mathsf{A}}\rangle$-mCCA, the latter notion is separated from the former notion if $(\widetilde{\mathsf{B}}||\widetilde{\mathsf{F}}||\widetilde{\mathsf{A}}) \not\subseteq_{qs} (\mathsf{B}||\mathsf{F}||\mathsf{A})$, $(\widetilde{\mathsf{B}}||\widetilde{\mathsf{F}}) \not\subseteq_{qs} (\mathsf{B}||\mathsf{F})$, or $(\widetilde{\mathsf{F}}||\widetilde{\mathsf{A}}) \not\subseteq_{qs} (\mathsf{F}||\mathsf{A})$ holds for PKE schemes with superpolynomially large plaintext space. We show that if none of the above conditions are satisfied, then we actually have implication from the former notion to the latter notion, where this implication is also true for PKE schemes with polynomially bounded plaintext space size and for KEMs.

Theorem 14. *For both PKE schemes and KEMs, if* $(\widetilde{\mathsf{B}}||\widetilde{\mathsf{F}}||\widetilde{\mathsf{A}}) \subseteq_{qs} (\mathsf{B}||\mathsf{F}||\mathsf{A})$, $(\widetilde{\mathsf{B}}||\widetilde{\mathsf{F}}) \subseteq_{qs} (\mathsf{B}||\mathsf{F})$, *and* $(\widetilde{\mathsf{F}}||\widetilde{\mathsf{A}}) \subseteq_{qs} (\mathsf{F}||\mathsf{A})$ *hold simultaneously, then* $\langle \mathsf{B} : \mathsf{F} : \mathsf{A}\rangle$-mCCA *security implies* $\langle \widetilde{\mathsf{B}} : \widetilde{\mathsf{F}} : \widetilde{\mathsf{A}}\rangle$-mCCA *security.*

Intuition. The key point is that if the three conditions regarding query sequences are satisfied, then whatever strategy regarding the "flexible" queries an $\langle \widetilde{\mathsf{B}} : \widetilde{\mathsf{F}} : \widetilde{\mathsf{A}}\rangle$-mCCA adversary may take, the $\langle \widetilde{\mathsf{B}} : \widetilde{\mathsf{F}} : \widetilde{\mathsf{A}}\rangle$-mCCA experiment can be perfectly simulated by an $\langle \mathsf{B} : \mathsf{F} : \mathsf{A}\rangle$-mCCA adversary

Proof. Since the proof is essentially the same for both PKE schemes and KEMs, we only show the PKE case.

Let $\Pi = (\mathsf{PKG}, \mathsf{PEnc}, \mathsf{PKG})$ be a PKE scheme which is $\langle \mathsf{B} : \mathsf{F} : \mathsf{A}\rangle$-mCCA secure. We consider the following cases:

- **Case 1:** $(\widetilde{\mathsf{B}}||\widetilde{\mathsf{F}}||\widetilde{\mathsf{A}}) = \emptyset$

- **Case 2:** $\mathsf{F} = \mathsf{unbound}$
- **Case 3:** $\mathsf{B} = \mathsf{unbound}$
- **Case 4:** $\mathsf{A} = \mathsf{unbound}$
- **Case 5:** None of $(\mathsf{B}, \mathsf{F}, \mathsf{A})$ is equal to $\mathsf{unbound}$ and $(\widetilde{\mathsf{B}}||\widetilde{\mathsf{F}}||\widetilde{\mathsf{A}}) \neq \emptyset$

Note that these cases cover all the possibilities. We show that under the given conditions, $\langle \mathsf{B} : \mathsf{F} : \mathsf{A} \rangle$-mCCA security implies $\langle \widetilde{\mathsf{B}} : \widetilde{\mathsf{F}} : \widetilde{\mathsf{A}} \rangle$-mCCA security for every case. (Since first four cases are (almost) trivial, we only show the sketch for them.)

Case 1: $(\widetilde{\mathsf{B}}||\widetilde{\mathsf{F}}||\widetilde{\mathsf{A}}) = \emptyset$**.** In this case, we have $\widetilde{\mathsf{B}} = \widetilde{\mathsf{F}} = \widetilde{\mathsf{A}} = \emptyset$, and thus $\langle \widetilde{\mathsf{B}} : \widetilde{\mathsf{F}} : \widetilde{\mathsf{A}} \rangle$-mCCA security is equivalent to IND-CPA security. Therefore, for any $\mathsf{B}, \mathsf{F}, \mathsf{A} \in \mathcal{QS}^*$, $\langle \mathsf{B} : \mathsf{F} : \mathsf{A} \rangle$-mCCA security implies $\langle \widetilde{\mathsf{B}} : \widetilde{\mathsf{F}} : \widetilde{\mathsf{A}} \rangle$-mCCA security.

Case 2: $\mathsf{F} = \mathsf{unbound}$**.** In this case, $\langle \mathsf{B} : \mathsf{F} : \mathsf{A} \rangle$-mCCA security implies $\langle \mathsf{unbound} :: \mathsf{unbound} \rangle$-mCCA security (i.e. IND-CCA2 security), which trivially implies $\langle \widetilde{\mathsf{B}} : \widetilde{\mathsf{F}} : \widetilde{\mathsf{A}} \rangle$-mCCA security for any $\widetilde{\mathsf{B}}, \widetilde{\mathsf{F}}, \widetilde{\mathsf{A}} \in \mathcal{QS}^*$.

Case 3: $\mathsf{B} = \mathsf{unbound}$**.** In this case, $\langle \mathsf{B} : \mathsf{F} : \mathsf{A} \rangle$-mCCA security implies $\langle \mathsf{unbound} :: (\mathsf{F}||\mathsf{A}) \rangle$-mCCA security. Now, consider any $\langle \widetilde{\mathsf{B}} : \widetilde{\mathsf{F}} : \widetilde{\mathsf{A}} \rangle$-mCCA PPTA adversary $\mathcal{A}$. Then, $\langle \mathsf{unbound} :: (\mathsf{F}||\mathsf{A}) \rangle$-mCCA PPTA adversary $\mathcal{B}$ can perfectly simulate the $\langle \widetilde{\mathsf{B}} : \widetilde{\mathsf{F}} : \widetilde{\mathsf{A}} \rangle$-mCCA experiment for $\mathcal{A}$. Concretely, the decryption queries from $\mathcal{A}$ before the challenge can be answered by $\mathcal{B}$ by unbounded access to the (single) decryption oracle available for $\mathcal{B}$ before the challenge. Let $\widetilde{\mathsf{F}}_1$ be a part of $\mathcal{A}$'s $\widetilde{\mathsf{F}}$-queries made before the challenge, and let $\widetilde{\mathsf{F}}_2$ be the remaining $\widetilde{\mathsf{F}}$-queries such that $(\widetilde{\mathsf{F}}_1||\widetilde{\mathsf{F}}_2) = \widetilde{\mathsf{F}}$. $\mathcal{A}$ can make $(\widetilde{\mathsf{F}}_2||\widetilde{\mathsf{A}})$-queries after the challenge. However, since $(\widetilde{\mathsf{F}}_2||\widetilde{\mathsf{A}}) \subseteq_{qs} (\widetilde{\mathsf{F}}||\widetilde{\mathsf{A}}) \subseteq_{qs} (\mathsf{F}||\mathsf{A})$ due to the given condition, the queries made by $\mathcal{A}$ after the challenge can be perfectly responded by $\mathcal{B}$ by $(\mathsf{F}||\mathsf{A})$-queries available for $\mathcal{B}$ after the challenge. Therefore, $\mathcal{B}$ can perfectly simulate the experiment for $\mathcal{A}$, which means that $\mathcal{A}$ and $\mathcal{B}$ have the same advantage, and if $\mathcal{A}$ has non-negligible advantage, so does $\mathcal{B}$. Therefore, in Case 3, $\langle \mathsf{B} : \mathsf{F} : \mathsf{A} \rangle$-mCCA security implies $\langle \widetilde{\mathsf{B}} : \widetilde{\mathsf{F}} : \widetilde{\mathsf{A}} \rangle$-mCCA security.

Case 4: $\mathsf{A} = \mathsf{unbound}$**.** In this case, $\langle \mathsf{B} : \mathsf{F} : \mathsf{A} \rangle$-mCCA security implies $\langle (\mathsf{B}||\mathsf{F}) :: \mathsf{unbound} \rangle$-mCCA security. Now, consider any $\langle \widetilde{\mathsf{B}} : \widetilde{\mathsf{F}} : \widetilde{\mathsf{A}} \rangle$-mCCA PPTA adversary $\mathcal{A}$. Then, $\langle (\mathsf{B}||\mathsf{F}) :: \mathsf{unbound} \rangle$-mCCA PPTA adversary $\mathcal{B}$ can perfectly simulate the $\langle \widetilde{\mathsf{B}} : \widetilde{\mathsf{F}} : \widetilde{\mathsf{A}} \rangle$-mCCA experiment for $\mathcal{A}$. Concretely, due to the given condition $(\widetilde{\mathsf{B}}||\widetilde{\mathsf{F}}) \subseteq_{qs} (\mathsf{B}||\mathsf{F})$, all the queries from $\mathcal{A}$ before the challenge (i.e. $(\widetilde{\mathsf{B}}||\widetilde{\mathsf{F}})$-queries) can be perfectly answered by $\mathcal{B}$ by $(\mathsf{B}||\mathsf{F})$-queries available for $\mathcal{B}$ before the challenge. Moreover, any query from $\mathcal{A}$ after the challenge can be answered by $\mathcal{B}$ by unbounded access to the (single) decryption oracle available for $\mathcal{B}$ after the challenge. Therefore, $\mathcal{B}$ can perfectly simulate the experiment for $\mathcal{A}$, which means that $\mathcal{A}$ and $\mathcal{B}$ have the same advantage, and if $\mathcal{A}$ has non-negligible advantage, so does $\mathcal{B}$. Therefore, in Case 4, $\langle \mathsf{B} : \mathsf{F} : \mathsf{A} \rangle$-mCCA security implies $\langle \widetilde{\mathsf{B}} : \widetilde{\mathsf{F}} : \widetilde{\mathsf{A}} \rangle$-mCCA security.

Case 5: None of $(\mathsf{B}, \mathsf{F}, \mathsf{A})$ **is equal to** $\mathsf{unbound}$ **and** $(\widetilde{\mathsf{B}}||\widetilde{\mathsf{F}}||\widetilde{\mathsf{A}}) \neq \emptyset$**.** In this case, none of $(\widetilde{\mathsf{B}}, \widetilde{\mathsf{F}}, \widetilde{\mathsf{A}})$ is equal to $\mathsf{unbound}$ as well, due to the given condition $(\widetilde{\mathsf{B}}||\widetilde{\mathsf{F}}||\widetilde{\mathsf{A}}) \subseteq_{qs} (\mathsf{B}||\mathsf{F}||\mathsf{A})$. Now, consider any $\langle \widetilde{\mathsf{B}} : \widetilde{\mathsf{F}} : \widetilde{\mathsf{A}} \rangle$-mCCA PPTA adversary $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$ attacking the PKE

scheme Π. We show that there exists a PPTA adversary $\mathcal{B} = (\mathcal{B}_1, \mathcal{B}_2)$ that can perfectly simulate the $\langle \widetilde{\mathsf{B}} : \widetilde{\mathsf{F}} : \widetilde{\mathsf{A}} \rangle$-mCCA experiment for $\mathcal{A}$ and has the same advantage as $\mathcal{A}$ in breaking $\langle \mathsf{B} : \mathsf{F} : \mathsf{A} \rangle$-mCCA security of the same PKE scheme Π.

For notational convenience, we write $\mathsf{seq} = (\mathsf{B}||\mathsf{F}||\mathsf{A})$ and $\widetilde{\mathsf{seq}} = (\widetilde{\mathsf{B}}||\widetilde{\mathsf{F}}||\widetilde{\mathsf{A}})$, and moreover let us write $\widetilde{\mathsf{seq}} = (b_1 \dots b_n)$ so that $b_i \in \{\mathsf{s}, \mathsf{p}\}$ for every $i \in [n]$.

We consider the "splitting" of seq according to $\widetilde{\mathsf{seq}} = (b_1, \dots, b_n)$ so that $\mathsf{seq} = (\mathsf{seq}_1|| \dots ||\mathsf{seq}_n||\mathsf{seq}_{n+1})$, and for every $1 \le i \le n$:

- if $b_i = \mathsf{s}$ then $\mathsf{seq}_i = \mathsf{s}$ or $\mathsf{seq}_i = \mathsf{p}$
- if $b_i = \mathsf{p}$ then $\mathsf{seq}_i = (\mathsf{s}^{q_i}, \mathsf{p})$ for some integer $q_i \ge 0$.

We put no condition on seq_{n+1}, which could be $\emptyset$. This splitting guarantees $b_i \subseteq_{qs} \mathsf{seq}_i$ for every $1 \le i \le n$. We stress that such splitting of seq (according to $\widetilde{\mathsf{seq}}$) is always possible as long as $\widetilde{\mathsf{seq}} \subseteq_{qs} \mathsf{seq}$, which is guaranteed by the given condition $(\widetilde{\mathsf{B}}||\widetilde{\mathsf{F}}||\widetilde{\mathsf{A}}) \subseteq_{qs} (\mathsf{B}||\mathsf{F}||\mathsf{A})$.

Now, the description of the $\langle \mathsf{B} : \mathsf{F} : \mathsf{A} \rangle$-mCCA adversary $\mathcal{B} = (\mathcal{B}_1, \mathcal{B}_2)$ is as follows.

$\mathcal{B}_1$: On input pk, $\mathcal{B}_1$ runs $\mathcal{A}_1$ with input pk. When $\mathcal{A}_1$ makes i-th query (which is a single query if $b_i = \mathsf{s}$ and is a parallel query if $b_i = \mathsf{p}$), $\mathcal{B}_1$ answers by following seq_i-queries. (If $\mathcal{A}_1$ makes no decryption query before the challenge, this process is skipped.) Since we have $b_i \subseteq_{qs} \mathsf{seq}_i$ for every $1 \le i \le n$ by definition and we also have $(\widetilde{\mathsf{B}}||\widetilde{\mathsf{F}}) \subseteq_{qs} (\mathsf{B}||\mathsf{F})$ due to the given condition, $\mathcal{B}_1$ can always answer $\mathcal{A}_1$'s queries, whatever strategy $\mathcal{A}_1$ may take for making the decryption queries (i.e. how to make $\widetilde{\mathsf{F}}$-queries before and after the challenge). When $\mathcal{A}_1$ terminates with output two plaintexts (m_0, m_1) and state information $\mathsf{st}_{\mathcal{A}}$, $\mathcal{B}_1$ sets its own state information $\mathsf{st}_{\mathcal{B}}$ that consists of all the values known to $\mathcal{B}_1$ and terminates with output two plaintexts (m_0, m_1) and $\mathsf{st}_{\mathcal{B}}$.

$\mathcal{B}_2$: On input $(c^*, \mathsf{st}_{\mathcal{B}})$ where c^* is the challenge ciphertext for $\mathcal{B}$, $\mathcal{B}_2$ runs $\mathcal{A}_2$ with input $(c^*, \mathsf{st}_{\mathcal{A}})$. Let $\widetilde{\mathsf{F}}_1$ be the $\widetilde{\mathsf{F}}$-queries $\mathcal{A}_1$ made ($\widetilde{\mathsf{F}}_1$ might be $\emptyset$). That is, $\mathcal{A}_1$ has made $(\widetilde{\mathsf{B}}||\widetilde{\mathsf{F}}_1)$-queries before the challenge, and $\mathcal{A}_2$ can make $(\widetilde{\mathsf{F}}_2||\widetilde{\mathsf{A}})$-queries after the challenge, where $(\widetilde{\mathsf{F}}_1||\widetilde{\mathsf{F}}_2) = \widetilde{\mathsf{F}}$.

Depending on the type of $\mathcal{A}_1$, $\mathcal{B}_2$ works differently in the following ways:

- *Type 1:* $(\widetilde{\mathsf{B}}||\widetilde{\mathsf{F}}_1) \subseteq_{qs} \mathsf{B}$. (This type includes the case in which $\mathcal{A}_1$ has made no query.) In this case, $\mathcal{B}_1$ has not made any F-queries, because $\mathcal{B}_1$ could have answered $\mathcal{A}_1$'s $(\widetilde{\mathsf{B}}||\widetilde{\mathsf{F}}_1)$-queries by only making B-queries. Therefore, $\mathcal{B}_2$ is allowed to make $(\mathsf{F}||\mathsf{A})$-queries fully. Since $(\widetilde{\mathsf{F}}_2||\widetilde{\mathsf{A}}) \subseteq_{qs} (\widetilde{\mathsf{F}}||\widetilde{\mathsf{A}})$, due to the given condition $(\widetilde{\mathsf{F}}||\widetilde{\mathsf{A}}) \subseteq_{qs} (\mathsf{F}||\mathsf{A})$, we have $(\widetilde{\mathsf{F}}_2||\widetilde{\mathsf{A}}) \subseteq_{qs} (\mathsf{F}||\mathsf{A})$. Therefore, $\mathcal{B}_2$ can perfectly answer $(\widetilde{\mathsf{F}}_2||\widetilde{\mathsf{A}})$-queries from $\mathcal{A}_2$ by $(\mathsf{F}||\mathsf{A})$-queries available for $\mathcal{B}_2$.
- *Type 2:* $(\widetilde{\mathsf{B}}||\widetilde{\mathsf{F}}_1) \not\subseteq_{qs} \mathsf{B}$. Let $(\widetilde{\mathsf{B}}||\widetilde{\mathsf{F}}_1) = (b_1 \dots b_j)$ for some $1 \le j \le n$. That is, $\mathcal{A}_1$ has made j queries. (Note that $(\widetilde{\mathsf{B}}||\widetilde{\mathsf{F}}_1) \neq \emptyset$ and thus such integer $j > 0$ exists.) We further consider two sub-types $j = n$ and $j < n$.

  If $j = n$, then no further query is allowed for $\mathcal{A}_2$, and thus $\mathcal{B}_2$ need not use its own decryption oracle any further.

  If $j < n$, then according to the description of $\mathcal{B}_1$, $\mathcal{B}_1$ has made $(\mathsf{seq}_1|| \dots ||\mathsf{seq}_j)$-queries. $\mathcal{B}_2$ is thus allowed to make $(\mathsf{seq}_{j+1}|| \dots ||\mathsf{seq}_n||\mathsf{seq}_{n+1})$-queries. On the other hand, by definition, $\mathcal{A}_2$ is only allowed to make $(\widetilde{\mathsf{F}}_2||\widetilde{\mathsf{A}})$-queries where $(\widetilde{\mathsf{F}}_2||\widetilde{\mathsf{A}}) = (b_{j+1} \dots b_n)$. Recall that $b_i \subseteq_{qs} \mathsf{seq}_i$ for every $j+1 \le i \le n$. Therefore, all queries

from $\mathcal{A}_2$ can be perfectly answered by using $\mathcal{B}_2$'s decryption oracle which is available for $\mathcal{B}$ after the challenge.

When $\mathcal{A}_2$ terminated with output the guess bit b', $\mathcal{B}_2$ also output b' and terminates.

The above completes the description of $\mathcal{B}$. It is straightforward to see that $\mathcal{B}$ perfectly simulates $\langle\widetilde{\mathsf{B}} : \widetilde{\mathsf{F}} : \widetilde{\mathsf{A}}\rangle$-mCCA experiment for $\mathcal{A}$, and $\mathcal{B}$ succeeds in guessing the challenge bit whenever $\mathcal{A}$ does so, which means that $\mathcal{B}$ and $\mathcal{A}$ have exactly the same advantage. If $\mathcal{A}$ has non-negligible advantage, so does $\mathcal{B}$. Since it contradicts that the PKE scheme Π is $\langle\mathsf{B} : \mathsf{F} : \mathsf{A}\rangle$-mCCA secure, such PPTA adversary $\mathcal{A}$ must not exist. This shows that in Case 5, $\langle\mathsf{B} : \mathsf{F} : \mathsf{A}\rangle$-mCCA security implies $\langle\widetilde{\mathsf{B}} : \widetilde{\mathsf{F}} : \widetilde{\mathsf{A}}\rangle$-mCCA security.

We have shown that $\langle\mathsf{B} : \mathsf{F} : \mathsf{A}\rangle$-mCCA security implies $\langle\widetilde{\mathsf{B}} : \widetilde{\mathsf{F}} : \widetilde{\mathsf{A}}\rangle$-mCCA security for all cases (Cases 1 to 5). This completes the proof of Theorem 14. □

Combining Theorem 14 with Theorem 10 in Section 4.2.2, we immediately obtain the following corollary.

Corollary 3. *For PKE schemes with polynomially bounded plaintext space size and for KEMs, if* $(\widetilde{\mathsf{B}}||\widetilde{\mathsf{F}}||\widetilde{\mathsf{A}}) \subseteq_{qs} (\mathsf{B}||\mathsf{F}||\mathsf{A})$ *and* $(\widetilde{\mathsf{F}}||\widetilde{\mathsf{A}}) \subseteq_{qs} (\mathsf{F}||\mathsf{A})$ *hold simultaneously, then* $\langle\mathsf{B} : \mathsf{F} : \mathsf{A}\rangle$-mCCA *security implies* $\langle\widetilde{\mathsf{B}} : \widetilde{\mathsf{F}} : \widetilde{\mathsf{A}}\rangle$-mCCA *security.*

Proof. Let $\mathsf{seq} = (\mathsf{F}||\mathsf{A})$ and $\widetilde{\mathsf{seq}} = (\widetilde{\mathsf{F}}||\widetilde{\mathsf{A}})$. By Theorem 10, for PKE schemes with polynomially bounded plaintext space size and for KEMs, $\langle\mathsf{B} : \mathsf{F} : \mathsf{A}\rangle$-mCCA (resp. $\langle\widetilde{\mathsf{B}} : \widetilde{\mathsf{F}} : \widetilde{\mathsf{A}}\rangle$-mCCA) security and $\langle\mathsf{B} : \mathsf{seq} : \emptyset\rangle$-mCCA (resp. $\langle\widetilde{\mathsf{B}} : \widetilde{\mathsf{seq}} : \emptyset\rangle$-mCCA) security is equivalent. Therefore, $\langle\mathsf{B} : \mathsf{F} : \mathsf{A}\rangle$-mCCA security implies $\langle\widetilde{\mathsf{B}} : \widetilde{\mathsf{F}} : \widetilde{\mathsf{A}}\rangle$-mCCA security if and only if $\langle\mathsf{B} : \mathsf{seq} : \emptyset\rangle$-mCCA security implies $\langle\widetilde{\mathsf{B}} : \widetilde{\mathsf{seq}} : \emptyset\rangle$-mCCA security.

Now, consider $\langle\mathsf{B} : \mathsf{seq} : \emptyset\rangle$-mCCA security and $\langle\widetilde{\mathsf{B}} : \widetilde{\mathsf{seq}} : \emptyset\rangle$-mCCA security. Then, by Theorem 14, the former implies the latter if $(\widetilde{\mathsf{B}}||\widetilde{\mathsf{seq}}||\emptyset) \subseteq_{qs} (\mathsf{B}||\mathsf{seq}||\emptyset)$, $(\widetilde{\mathsf{B}}||\widetilde{\mathsf{seq}}) \subseteq_{qs} (\mathsf{B}||\mathsf{seq})$, and $(\widetilde{\mathsf{seq}}||\emptyset) \subseteq_{qs} (\mathsf{seq}||\emptyset)$ hold simultaneously. Note that these three conditions are equivalent to the two conditions $(\widetilde{\mathsf{B}}||\widetilde{\mathsf{F}}||\widetilde{\mathsf{A}}) \subseteq_{qs} (\mathsf{B}||\mathsf{F}||\mathsf{A})$ and $(\widetilde{\mathsf{F}}||\widetilde{\mathsf{A}}) \subseteq_{qs} (\mathsf{F}||\mathsf{A})$. In summary, for PKE schemes with polynomially bounded plaintext space size and for KEMs, $\langle\mathsf{B} : \mathsf{F} : \mathsf{A}\rangle$-mCCA security implies $\langle\widetilde{\mathsf{B}} : \widetilde{\mathsf{F}} : \widetilde{\mathsf{A}}\rangle$-mCCA security if both $(\widetilde{\mathsf{B}}||\widetilde{\mathsf{F}}||\widetilde{\mathsf{A}}) \subseteq_{qs} (\mathsf{B}||\mathsf{F}||\mathsf{A})$ and $(\widetilde{\mathsf{F}}||\widetilde{\mathsf{A}}) \subseteq_{qs} (\mathsf{F}||\mathsf{A})$ hold simultaneously. □

4.3.5 Necessary and Sufficient Conditions for Implications/Separations

As a summarization of the results in this section, we show the following necessary and sufficient conditions for implications/separations among mixed CCA security notions, where the results for PKE schemes differ depending on the size of a plaintext space.

Theorem 15. *For PKE schemes with superpolynomially large plaintext space size,* $\langle\mathsf{B} : \mathsf{F} : \mathsf{A}\rangle$-mCCA *security implies* $\langle\widetilde{\mathsf{B}} : \widetilde{\mathsf{F}} : \widetilde{\mathsf{A}}\rangle$-mCCA *security if and only if* $(\widetilde{\mathsf{B}}||\widetilde{\mathsf{F}}||\widetilde{\mathsf{A}}) \subseteq_{qs} (\mathsf{B}||\mathsf{F}||\mathsf{A})$, $(\widetilde{\mathsf{B}}||\widetilde{\mathsf{F}}) \subseteq_{qs} (\mathsf{B}||\mathsf{F})$, *and* $(\widetilde{\mathsf{F}}||\widetilde{\mathsf{A}}) \subseteq_{qs} (\mathsf{F}||\mathsf{A})$ *hold simultaneously.*

Proof. This follows from a combination of Theorem 11, Corollary 1, Corollary 2, and Theorem 14. □

Theorem 16. *For PKE schemes with polynomially bounded plaintext space size and for KEMs,* $\langle\mathsf{B} : \mathsf{F} : \mathsf{A}\rangle$-mCCA *security implies* $\langle\widetilde{\mathsf{B}} : \widetilde{\mathsf{F}} : \widetilde{\mathsf{A}}\rangle$-mCCA *security if and only if* $(\widetilde{\mathsf{B}}||\widetilde{\mathsf{F}}||\widetilde{\mathsf{A}}) \subseteq_{qs} (\mathsf{B}||\mathsf{F}||\mathsf{A})$ *and* $(\widetilde{\mathsf{F}}||\widetilde{\mathsf{A}}) \subseteq_{qs} (\mathsf{F}||\mathsf{A})$ *hold simultaneously.*

Proof. This follows from a combination of Theorem 11, Corollary 1, and Corollary 3. □

We believe the relations among security notions shown in this section are useful for future studies on PKE schemes and KEMs whose security notions can be expressed in mixed CCA security notions. As a concrete evidence of usefulness, by utilizing the above theorems, we fully establish the relations among bounded parallel CCA security and other exising security notions, which we summarize in the figures. The results for PKE schemes with superpolynomially large plaintext space size is summarized in Figure 4.5, and those for PKE schemes with polynomially bounded plaintext space size and for KEMs is summarized in Figure 4.6. In the figures, the known implication/separation results among PKE schemes and KEMs from previous results [7, 64, 40] are reflected.

All the implications/separations in the figures can be derived from Theorems 15 and 16 in Section 4.3.5.

We note that the previously established relations among security notions [7, 64, 40] can be re-proved as corollaries from the above theorems.

Importance of Plaintext Space Size in Relations among Security Notions for PKE Schemes. As our implications/separations in this section clarifies, it is important to care about the size of the plaintext space size when considering relations among security notions for PKE schemes. A natural question would be "when" we should care about it. Theorems 15 and 16 tell us that given $\langle \mathsf{B} : \mathsf{F} : \mathsf{A} \rangle$-mCCA and $\langle \widetilde{\mathsf{B}} : \widetilde{\mathsf{F}} : \widetilde{\mathsf{A}} \rangle$-mCCA security notions, the implication/separation from the former notion to the latter notion differs if $(\widetilde{\mathsf{B}}||\widetilde{\mathsf{F}}||\widetilde{\mathsf{A}}) \subseteq_{qs} (\mathsf{B}||\mathsf{F}||\mathsf{A})$, $(\widetilde{\mathsf{F}}||\widetilde{\mathsf{A}}) \subseteq_{qs} (\mathsf{F}||\mathsf{A})$, and $(\widetilde{\mathsf{B}}||\widetilde{\mathsf{F}}) \not\subseteq_{qs} (\mathsf{B}||\mathsf{F})$ hold simultaneously.

4.4 Black-box Feasibility Results from IND-CPA Secure PKE Schemes

By adopting the notion of mixed CCA security, we show two black-box construction of PKE schemes, which can encrypt plaintexts of polynomial length (thus, exponentially large plaintext space), from IND-CPA secure PKE schemes. The first construction achieves slightly but strictly stronger security than NM-q-CCA2 security and thus achieves the currently strongest security notion among the security notions achieved by other PKE constructions that uses only IND-CPA secure PKE schemes as building blocks. The second construction achieves yet another security notion which cannot be directly compared with the notion achieved by our first construction (or cannot be compared even with NM-CPA security).

The first result is the following.

Theorem 17. *For any polynomial $q \geq 0$, there exists a shielding black-box construction of a $\langle \emptyset : \mathsf{s}^q\mathsf{p} : \emptyset \rangle$-mCCA secure PKE scheme which can encrypt plaintexts of polynomial length from an* IND-CPA *secure PKE scheme.*

Proof. This theorem is proved by combining the existing results and Theorem 10 in Section 4.2.2. Our construction is fairly simple: Using the construction of an NM-q-CCA2 secure PKE scheme by Choi et al. [38] as a KEM, and combining it with an IND-CCA2 secure DEM.

The following statement is due to the result by Choi et al. [38].

Lemma 17. *[38] For any polynomial $q \geq 0$, there exists a shielding black-box construction of an* NM-q-CCA2 *secure PKE scheme which can encrypt plaintexts of polynomial length from an* IND-CPA *secure PKE scheme.*

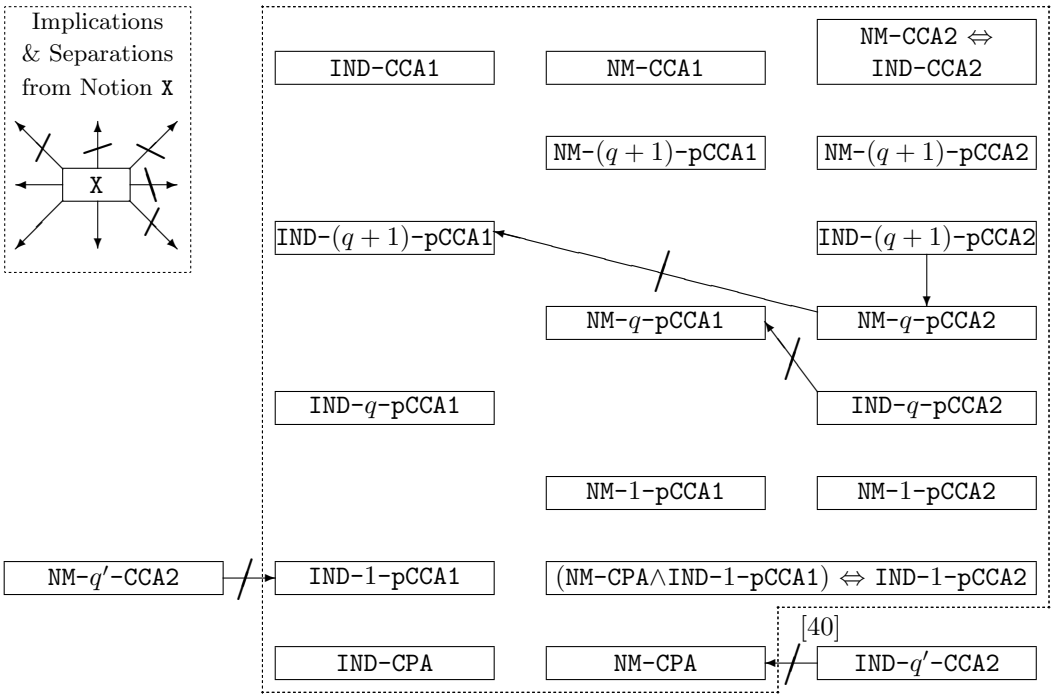

Figure 4.5: Relations among bounded parallel CCA security and other existing security notions for PKE with superpolynomially large plaintext space size. In the figure, $q \geq 2$ and $q' \geq 0$ are polynomials. For any security notion X inside the area enclosed by the dotted line, X implies any of the notions written to the left and below X. However, X does not imply any of the notions written to the right or above X. The arrows for implications and separations without any reference and equivalence except for IND-CCA2 $\Leftrightarrow$ NM-CCA2 are the relations that are not known before.

We write the Choi et al. NM-q-CCA2 secure PKE scheme that is constructed from any IND-CPA secure PKE scheme as the *CDMW PKE scheme*. (We recall the Choi et al. [38] construction in Appendix B.1.)

For any security goal GOAL and any attack type ATK of an adversary considered in Section 2.2, a GOAL-ATK secure PKE scheme can be trivially used as a GOAL-ATK secure KEM by encrypting a uniformly random string K and using it as a session-key, if the PKE scheme has sufficiently large plaintext space, say k-bits for k-bit security. Since the CDMW PKE will have k-bit plaintext space if we use IND-CPA secure PKE scheme with k-bit plaintext space (which is possible to achieve from any 1-bit PKE scheme by a simple concatenation of ciphertexts) as the underlying PKE scheme of the CDMW construction, from the CDMW PKE scheme we can, for any polynomial $q \geq 0$, obtain NM-q-CCA2 $= \langle \emptyset : \mathsf{s}^q : \mathsf{p} \rangle$-mCCA secure KEM (we call it the *CDMW KEM*).

Then, by the special case of Theorem 10 for KEMs in which $\mathsf{B} = \emptyset$, $\mathsf{F} = \mathsf{s}^q$, and $\mathsf{A} = \mathsf{p}$, we can immediately say that the CDMW KEM is $\langle \emptyset : \mathsf{s}^q, \mathsf{p} : \emptyset \rangle$-mCCA secure.

Finally, by the following composition result of a KEM and a DEM which is implicit from the work by Herranz et al. [64], we can show that an appropriate combination of a KEM and a DEM will result in a PKE scheme with the claimed security.

Lemma 18. *(Implicit from [64].) Let* $\mathsf{B}, \mathsf{F}, \mathsf{A} \in \mathcal{QS}^*$. *If a KEM* Γ *is* $\langle \mathsf{B} : \mathsf{F} : \mathsf{A} \rangle$-mCCA *secure and a DEM* D *is* IND-CCA2 *secure, then a PKE scheme constructed from the KEM* Γ *and the*

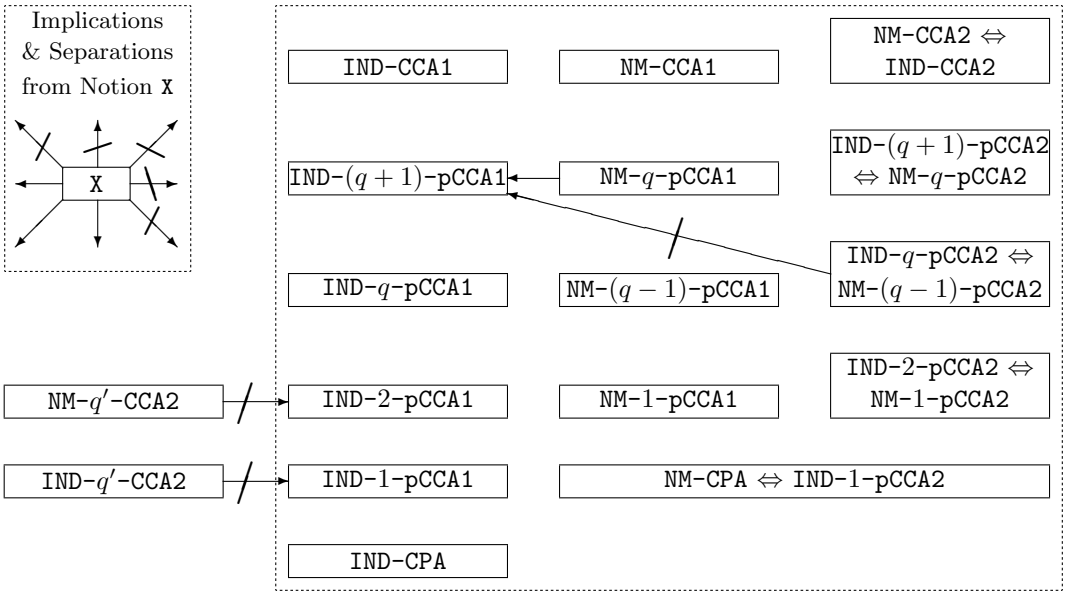

Figure 4.6: Relations among bounded parallel CCA security and other existing security notions for PKE schemes with polynomially bounded plaintext space size and those for KEMs. In the figure, $q \geq 3$ and $q' \geq 0$ are polynomials. The notations are the same as those used in Figure 4.5.

DEM D in a straightforward manner satisfies $\langle \mathsf{B} : \mathsf{F} : \mathsf{A} \rangle$-mCCA *security.*

Thus, according to Lemma 18, if we construct a new PKE scheme by combining the CDMW KEM and an IND-CCA2 secure DEM, the resulting scheme is $\langle \emptyset : \mathsf{s}^q, \mathsf{p} : \emptyset \rangle$-mCCA secure. Since we can construct an IND-CCA2 secure DEM without any computational assumption[6] here we have shown how to construct, for any polynomial $q \geq 0$, a $\langle \emptyset : \mathsf{s}^q\mathsf{p} : \emptyset \rangle$-mCCA secure PKE scheme from any IND-CPA secure one.

Since the CDMW PKE scheme is a shielding black-box construction, and these "shielding" and "black-box" properties are trivially preserved by our construction, we conclude that our construction is also shielding and black-box. This completes the proof of Theorem 17. □

Though in the above we have shown how to enhance the CDMW PKE scheme which is NM-q-CCA2 $= \langle \emptyset : \mathsf{s}^q : \mathsf{p} \rangle$-mCCA secure to be $\langle \emptyset : \mathsf{s}^q\mathsf{p} : \emptyset \rangle$-mCCA secure, we remark that the original CDMW PKE scheme might be shown to be $\langle \emptyset : \mathsf{s}^q\mathsf{p} : \emptyset \rangle$-mCCA secure as it is (i.e. without using the arguments shown in the above theorem), under the same assumptions used to show its NM-q-CCA2 security. However, our main purpose here is to show the improved feasibility rather than the concrete construction and efficiency, and thus we did not try proving directly that the CDMW PKE is $\langle \emptyset : \mathsf{s}^q\mathsf{p} : \emptyset \rangle$-mCCA secure.

We remark that $\langle \emptyset : \mathsf{s}^q\mathsf{p} : \emptyset \rangle$-mCCA security trivially implies NM-q-CCA2 $= \langle \emptyset : s^q : \mathsf{p} \rangle$-mCCA security, while by Theorem 15 we know that for PKE schemes with superpolynomially large plaintext space size, $\langle \emptyset : \mathsf{s}^q : \mathsf{p} \rangle$-mCCA security does not imply $\langle \emptyset : \mathsf{s}^q\mathsf{p} : \emptyset \rangle$-mCCA security.

[6] One can use a combination of a one-time pad with a one-time secure message authentication code (MAC), both of which are possible without any computational assumption, in an *encrypt-then-MAC* manner [9] to achieve an IND-CCA2 secure DEM.

Therefore, for these types of PKE schemes, $\langle \emptyset : \mathsf{s}^q\mathsf{p} : \emptyset \rangle$-mCCA security is strictly stronger than NM-q-CCA2 security.

We also remark that due to Theorem 10, the Choi et al. result [38] (i.e. Lemma 17 in this section) already achieves the shielding black-box construction of $\langle \emptyset : \mathsf{s}^q\mathsf{p} : \emptyset \rangle$-mCCA secure PKE schemes for polynomially bounded plaintext space size. However, the Choi et al. result itself does not imply Theorem 17, because how to construct PKE schemes which can encrypt plaintexts of polynomially length from PKE schemes which has polynomial size plaintext space in a black-box and shielding manner is not known so far[7].

A corollary of Theorem 17 is the following.

Corollary 4. *There exists a shielding black-box construction of an* IND-1-pCCA2 *secure PKE scheme which can encrypt plaintexts of polynomial length from an* IND-CPA *secure PKE scheme.*

Our second result on black-box constructions is the following.

Theorem 18. *For any polynomials $q, q' \geq 0$, there exists a shielding black-box construction of a $\langle \mathsf{s}^q\mathsf{p} : \mathsf{s}^{q'} : \emptyset \rangle$-mCCA secure PKE scheme which can encrypt plaintexts of polynomial length from an* IND-CPA *secure PKE scheme.*

Proof. To prove this theorem, we combine the result from Theorem 17 and the construction by Cramer et al. [40] which, for any polynomial q, constructs an IND-q-CCA2 secure PKE scheme from any IND-CPA secure PKE scheme. (We call the construction by Cramer et al. [40] the *CHH+ PKE scheme*, and recall it in Appendix B.2.)

We will use the following generalized version of Lemma 1 in [40][8].

Lemma 19. *For any $\mathsf{B} \in \mathcal{QS}^*$, if the underlying PKE scheme Π in the CHH+ construction is $\langle \mathsf{B} :: \emptyset \rangle$-mCCA secure and the underlying signature scheme Σ is strongly one-time secure, then the CHH+ PKE scheme Π_{CHH+} (Fig. B.2) satisfies $\langle \mathsf{B} : \mathsf{s}^q : \emptyset \rangle$-mCCA security.*

The proof of this lemma is essentially the same as that of Lemma 1 in [40], and thus we omit it.

Due to Theorem 17 above, for any polynomial $q \geq 0$, we can construct a $\langle \emptyset : \mathsf{s}^q\mathsf{p} : \emptyset \rangle$-mCCA secure PKE scheme, which is also $\langle \mathsf{s}^q\mathsf{p} :: \emptyset \rangle$-mCCA secure, from any IND-CPA secure PKE scheme. Then, by using this PKE scheme as a building block of the CHH+ PKE scheme, due to Lemma 19, we have a PKE scheme which satisfies the claimed security. Moreover, the CHH+ PKE construction is shielding and black-box. Since the construction of the PKE scheme in Theorem 17 is also shielding and black-box, so is the construction here as a whole. The size of the plaintext space is maintained as well. This completes the proof of Theorem 18. □

We note that by Theorem 15, for PKE schemes with superpolynomially large plaintext space size, the security notion $\langle \mathsf{s}^q\mathsf{p} : \mathsf{s}^{q'} : \emptyset \rangle$-mCCA achieved in Theorem 18 cannot be directly compared even with NM-CPA= $\langle \emptyset :: \mathsf{p} \rangle$-mCCA security. That is, the security notion $\langle \mathsf{s}^q\mathsf{p} : \mathsf{s}^{q'} :$

[7]Recently, Myers and Shelat [81] showed a black-box construction of multi-bit IND-CCA2 secure PKE schemes from 1-bit IND-CCA2 secure PKE schemes. However, whether their results extend to mixed CCA security is not known so far. Moreover, their construction is non-shielding.

[8]The original statement of Lemma 1 in [40] is a special case of Lemma 19 in which $\mathsf{B} = \emptyset$. Moreover, the special case of Lemma 19 in which $\mathsf{B} = \mathsf{unbound}$ is also mentioned in [40]. See Remark 2 after the proof of Lemma 1 in [40].

$\emptyset\rangle$-mCCA does not imply NM-CPA security, and vice versa. Therefore, for PKE schemes with superpolynomially large plaintext space size, the achieved security in Theorem 18 also cannot be directly compared with those achieved in Theorem 17. However, the security achieved in Theorem 18 allows the bounded number of "flexible" single queries before and after the challenge, after the parallel query in the first stage, while the security achieved by Theorem 17 does not allow any query after one parallel query for an adversary. Moreover, at least the notion that can be achieved by Theorem 18 is stronger than IND-q-CCA2 type security, and thus we believe that Theorem 18 is also meaningful and interesting as a feasibility result.

Handling Decryption of Unboundedly Many Ciphertexts before the Challenge. Previous to our work, none of the constructions of PKE schemes that use only IND-CPA secure schemes have achieved the security notion against adversaries that can observe unboundedly many decryption results (via the decryption oracle) in the first stage, i.e., before choosing two challenge plaintexts, regardless of whether the construction is black-box or non-black-box. On the other hand, Theorems 17 and 18 (and also the combination of [38] and Theorem 10) clarified that it is possible to construct a PKE scheme which is secure even though an adversary can observe unboundedly many decryption results by one parallel decryption query before the challenge.

Thus, our results in this section clarified that *the difficulty of constructing an* IND-CCA1 *secure PKE scheme only from* IND-CPA *secure ones lies not in whether the number of decryption results that the adversary can see before the challenge is bounded or not, but in whether the number of the adversary's "adaptive" decryption queries is bounded.* We believe that this observation is important and interesting towards fully answering the problem of whether a CCA secure PKE schemes can be constructed only from IND-CPA secure ones, and can be seen as a concrete evidence that studying mixed CCA security is useful.

4.5 Open Problems

Two or More Parallel Queries? None of our feasibility results achieves parallel (or mixed) CCA security in which we can handle more than one parallel decryption query, and whether we can construct a PKE scheme with such security only using IND-CPA secure schemes is still unclear. Therefore, we would like to leave it as an open problem. Since any (unbounded) CCA secure PKE construction from IND-CPA secure ones must first be secure against adversaries who make two or more parallel decryption queries, we believe that overcoming this barrier of "two parallel queries" is worth tackling.

We notice that if it is generically possible to construct an NM-q-pCCA1 (resp. NM-q-pCCA2) secure PKE scheme from any IND-q-pCCA1 (resp. IND-q-pCCA2) secure one, by combining such a statement with Theorem 10 and taking the same KEM-DEM approach as done in the proof of Theorem 17, we will be able to construct an NM-q-pCCA1 (resp. NM-q-pCCA2) secure PKE schemes for $q \geq 1$ only from IND-CPA secure ones. Moreover, we also notice that if we can construct a strong DV-NIZK proof system with q-bounded "*parallel*" strong soundness, which is a natural extension of a strong DV-NIZK with q-bounded strong soundness [40] in the soundness experiment of which an adversary can ask verification of many theorem/proof pairs in a parallel manner, only from the existence of IND-CPA secure PKE schemes, then by using it in the NY construction [84] (resp. the DDN construction [47]) we will be able to construct an IND-$(q+1)$-pCCA1 (resp. IND-$(q+1)$-pCCA2) secure PKE scheme. However,

how to construct such a DV-NIZK proof system only from IND-CPA secure PKE schemes is not known so far. This might be worth looking at towards the next step from our results.

Efficient Instantiations of PKE with Bounded Parallel CCA Security? Seeking for bounded parallel CCA secure PKE schemes with practical efficiency from specific computational hardness assumptions may also be interesting. Cramer et al. [40] show that under the DDH assumption, for any polynomial q we can construct an IND-q-CCA2 secure PKE scheme which has redundancy-free ciphertext size (plaintext size plus randomness size used for encryption), which is not achieved by any IND-CCA2 secure PKE scheme in the standard model known so far. Of course tackling this problem is interesting only if we seek for schemes which have some properties that are not achieved by the known IND-CCA2 secure schemes such as [27, 70, 98, 65, 37, 61, 67, 73, 66, 62] (smaller ciphertext, smaller parameter size, smaller computational costs, and/or basing security on weaker assumptions, etc.)

On Black-Box Impossibility Results. Gertner et al. [55] show that there exists no shielding black-box construction of an IND-CCA1 secure PKE scheme from IND-CPA secure PKE schemes. Since the constructions in Theorems 17 and 18 are both shielding and black-box, according to the impossibility result of [55], we have that there exists no shielding black-box construction of an IND-CCA1 secure PKE scheme from PKE schemes which satisfy any security notion achieved in Theorems 17 and 18.

It is currently not known if we can strengthen the impossibility result of [55]. Thus, it may also be interesting to clarify if we can show a stronger impossibility result so that constructing IND-q-pCCA1 secure PKE schemes in a shielding and black-box manner for some $q > 1$ is impossible. (Or more generally, we can also consider the impossibility of some of mixed CCA security notion.) Note that this strengthening of the impossibility result of [55] is meaningful only if we consider parallel decryption queries, because the result by Choi et al. [38] already shows that it is possible to achieve the strongest form of (ordinary) bounded CCA security, i.e., for any polynomial $q' \geq 0$ achieving NM-q'-CCA2 secure PKE scheme from any IND-CPA secure one in a shielding and black-box manner is possible.

4.6 Conclusion

In this chapter, in order to address the further possibility of constructions of CCA secure PKE schemes only from CPA secure ones, we first introduced the notion of bounded parallel CCA security, which is an extension of the conventional bounded CCA security. We then investigated the implications and separations among bounded parallel CCA security notions and the conventional security notions for PKE schemes and KEMs. As a feasibility result, we showed a shielding black-box construction of an IND-1-pCCA2 secure PKE scheme from an IND-CPA secure PKE scheme. Moreover, in order to precisely describe further feasibility results, we introduced the notion of mixed CCA security, which is a generalization of the conventional bounded CCA security and bounded parallel CCA security, and then we showed two shielding black-box constructions of PKE schemes that satisfy stronger security notions than the security notions achieved by the existing shielding black-box constructions of PKE schemes from any IND-CPA secure PKE schemes. We furthermore discussed the consequences of our feasibility results, made several observations, and left some open problems. We believe that studying bounded parallel CCA security and mixed CCA security further will be good

intermediate steps towards solving the problem of whether constructing (unbounded) CCA secure PKE schemes from any CPA secure PKE schemes is possible or not.

Chapter 5

Conclusion

In this thesis, we have focused on generic constructions of CCA secure PKE schemes, and made the following contributions:

- Aiming at generic constructions that lead to CCA secure PKE schemes with practical efficiency, we focused on the IBE-to-PKE transformation paradigm, which is the only known generic methodology with which we can construct CCA secure PKE schemes with practical efficiency. To improve the large ciphertext size that all the previous methods sufferred from, we proposed two approaches. The first approach is to require non-malleability, slightly stronger security than CPA security, for the underlying IBE scheme, and we developed a new very simple IBE-to-PKE transformation where we only used a one-way function, the weakest primitive used in the area of cryptography, as an additional building block. The second approach is to develop a new efficient encapsulation scheme, which is a special kind of commitment scheme and is a primitive used in one of the previous IBE-to-PKE transformations, from a pseudorandom generator that satisfies near collision resistance for predetermined parts of output, and we use the new encapsulation scheme in the transformation. Both approaches do not need strong cryptographic primitives as additional building blocks, and lead to CCA secure PKE schemes with smaller ciphertext size than the previous IBE-to-PKE transformations.
- We focused on the problem of whether it is possible to construct a CCA secure PKE scheme only from a CPA secure one is one of the most important fundamental open problems, which leads to clarifying a necessary and sufficient condition to realize a CCA secure PKE scheme. Since we can achieve the best possible security in the bounded CCA security notions that capture security notions that lie between CPA and CCA, in order to further tackle the fundamental problem, we need new security notions that capture intermediate security notions that lie between CPA and CCA security in a different sense from the existing bounded CCA security definition. Motivated by this situation, in order to provide a theoretical foundation for further tackling the above problem, we focused on parallel decryption queries for the extension of bounded CCA security, and introduced a new security notion which we call *mixed CCA* security. It captures security against adversaries that make single and parallel decryption queries in a predetermined order, where each parallel query can contain unboundedly many ciphertexts. Moreover, how the decryption oracle is available before and after the challenge is also taken into account in this new security definition, which enables us to capture existing major security notions that lie between CPA and CCA security, including complex notion like

non-malleability against bounded CCA, in a unified security notion. We investigated the relations among mixed CCA security notions, and show necessary and sufficient conditions regarding implications/separations between any two notions in mixed CCA security for PKE schemes and KEMs. We then showed two black-box constructions of PKE schemes from a CPA secure scheme. The first one satisfies a strictly stronger security notion than the security notions achieved by the existing constructions of PKE schemes constructed only from a CPA secure one, while the second one achieves yet another security notion that has not been achieved by the previously known constructions. We also discussed the consequences of our results regarding security with parallel decryption queries and gave several observations, as well as several open problems.

From the former part of our results, the IBE-to-PKE transformations, we expect that more and more practical concrete PKE schemes which are CCA secure based on some concrete intractability assumptions will be constructed, and our transformations will give insights to such constructions that will be proposed in the future. In fact, previously, based on the construction idea of the first IBE-to-PKE transformation by Canetti et al. [34], one of the most efficient CCA secure PKE scheme was proposed by Boyen et al. [27, 28].

Regarding the latter part of our results, we believe that our way of formalizing and analysing the mixed CCA security notions will be useful, and can be done in the same way as ours, for any security notion of any primitive that involves an adversary's oracle queries. Moreover, we also believe that studying such security notions of primitives will lead to better understanding of the security notions and the primitives themseleves as well as concrete constructions of them, and will ultimately lead to systematizing the theoretical foundations in the area of cryptography.

Bibliography

[1] Digital Signature Standard (DSS). FIPS 186-2 (+ Change Notice), Revised Appendix 3.1, 2000. http://csrc.nist.gov/publications/fips/fips186-2/fips186-2-change1.pdf.

[2] CRYPTREC Report 2007 (in Japanese), page 31. http://www.cryptrec.go.jp/report/c07_wat_final.pdf. Older but English version is also available. CRYPTREC Report 2002, page 23. http://www.ipa.go.jp/security/enc/CRYPTREC/fy15/doc/c02e_report2.pdf.

[3] M. Abe, Y. Cui, H. Imai, and E. Kiltz. Efficient hybrid encryption from ID-based encryption. *Designs, Codes and Cryptography*, 54(3):205–240, 2010.

[4] M. Abe, R. Gennaro, K. Kurosawa, and V. Shoup. Tag-KEM/DEM: A new framework for hybrid encryption and a new analysis of Kurosawa-Desmedt KEM. In *Proc. of EUROCRYPT 2005*, volume 3494 of *LNCS*, pages 128–146. Springer, 2005.

[5] N. Attrapadung, Y. Cui, D. Galindo, G. Hanaoka, I. Hasuo, H. Imai, K. Matsuura, P. Yang, and R. Zhang. Relations among notions of security for identity based encryption schemes. In *Proc. of LATIN 2006*, volume 3887 of *LNCS*, pages 130–141. Springer, 2006.

[6] M. Bellare, A. Boldyreva, and A. Palacio. An uninstantiable random-oracle-model scheme for a hybrid-encryption problem. In *Proc. of ASIACRYPT 2004*, volume 3027 of *LNCS*, pages 171–188. Springer, 2004.

[7] M. Bellare, A. Desai, D. Pointcheval, and P. Rogaway. Relations among notions of security for public-key encryption schemes. In *Proc. of CRYPTO 1998*, volume 1462 of *LNCS*, pages 26–45. Springer, 1998.

[8] M. Bellare, D. Hofheinz, and E. Kiltz. Subtleties in the definition of IND-CCA: When and how should challenge-decryption be disallowed?, 2009. Available at eprint.iacr.org/2009/418.

[9] M. Bellare and C. Namprempre. Authenticated encryption: Relations among notions and analysis of the generic composition paradigm. In *Proc. of ASIACRYPT 2000*, volume 1976 of *LNCS*, pages 531–545. Springer, 2000.

[10] M. Bellare and P. Rogaway. Random oracles are practical: A paradigm for designing efficient protocols. In *Proc. of CCS 1993*, pages 62–73. ACM, 1993.

[11] M. Bellare and P. Rogaway. Optimal asymmetric encryption. In *Proc. of EUROCRYPT 1994*, volume 950 of *LNCS*, pages 92–111. Springer, 1995.

[12] M. Bellare and P. Rogaway. Collision-resistant hashing: Towards making UOWHFs practical. In *Proc. of CRYPTO 1997*, volume 1294 of *LNCS*, pages 320–335. Springer, 1997.

[13] M. Bellare and A. Sahai. Non-malleable encryption: Equivalence between two notions, and indistinguishability-based characterization. In *Proc. of CRYPTO 1999*, volume 1666 of *LNCS*, pages 519–536. Springer, 1999.

[14] M. Bellare and A. Sahai. Non-malleable encryption: Equivalence between two notions, and indistinguishability-based characterization, 2006. Full version of [13]. Available at eprint.iacr.org/2006/228.

[15] M. Bellare and M. Yung. Certifying permutations: Noninteractive zero-knowledge based on any trapdoor permutation. *J. of Cryptology*, 9(3):149–166, 1996.

[16] E. Biham and R. Chen. Near-collisions of SHA-0. In *Proc. of CRYPTO 2004*, volume 3152 of *LNCS*, pages 290–305. Springer, 2004.

[17] M. Blum, P. Feldman, and S. Micali. Non-interactive zero-knowledge and its applications. In *Proc. of STOC 1988*, pages 103–112. ACM, 1988.

[18] M. Blum and S. Micali. How to generate cryptographically strong sequences of pseudo-random bits. *SIAM J. Computing*, 13(4):850–864, 1984.

[19] A. Boldyreva and M. Fischlin. On the security of OAEP. In *Proc. of ASIACRYPT 2006*, volume 4284 of *LNCS*, pages 210–225. Springer, 2006.

[20] D. Boneh and X. Boyen. Efficient selective-ID secure identity-based encryption without random oracles. In *Proc. of EUROCRYPT 2004*, volume 3027 of *LNCS*, pages 223–238. Springer, 2004.

[21] D. Boneh and X. Boyen. Secure identity based encryption without random oracles. In *Proc. of CRYPTO 2004*, volume 3152 of *LNCS*, pages 443–459. Springer, 2004.

[22] D. Boneh, X. Boyen, and E.-J. Goh. Hierarchical identity based encryption with constant size ciphertext. In *Proc. of EUROCRYPT 2005*, volume 3494 of *LNCS*, pages 440–456. Springer, 2005.

[23] D. Boneh, R. Canetti, S. Halevi, and J. Katz. Chosen-ciphertext security from identity-based encryption. *SIAM J. Computing*, 36(5):1301–1328, 2007.

[24] D. Boneh and M. Franklin. Identity-based encryption from the Weil pairing. In *Proc. of CRYPTO 2001*, volume 2139 of *LNCS*, pages 213–229. Springer, 2001.

[25] D. Boneh, C. Gentry, and M. Hamberg. Space-efficient identity based encryption without pairings. In *Proc. of FOCS 2007*, pages 647–657. IEEE Computer Society Press, 2007.

[26] D. Boneh and J. Katz. Improved efficiency for CCA-secure cryptosystems built using identity-based encryption. In *Proc. of CT-RSA 2005*, volume 3376 of *LNCS*, pages 87–103. Springer, 2005.

[27] X. Boyen, Q. Mei, and B. Waters. Direct chosen ciphertext security from identity-based techniques. In *Proc. of CCS 2005*, pages 320–329. ACM, 2005.

[28] X. Boyen, Q. Mei, and B. Waters. Direct chosen ciphertext security from identity-based techniques, 2005. Updated version of [27]. Cryptology ePrint Archive: Report 2005/288. http://eprint.iacr.org/2005/288/.

[29] R. Canetti. Towards realizing random oracles: Hash functions that hide all partial information. In *Proc. of CRYPTO 1997*, volume 1294 of *LNCS*, pages 455–469. Springer, 1997.

[30] R. Canetti. Universally composable security: A new paradigm for cryptographic protocols. In *Proc. of FOCS 2001*, pages 136–145. IEEE Computer Society Press, 2001. Full version available at eprint.iacr.org/2000/067.

[31] R. Canetti and R.R. Dakdouk. Extractable perfectly one-way functions. In *Proc. of ICALP 2008 Part 2*, volume 5126 of *LNCS*, pages 449–460. Springer, 2008.

[32] R. Canetti, O. Goldreich, and S. Halevi. The random oracle methodology, revisited. In *Proc. of STOC 1998*, pages 209–218. ACM, 1998.

[33] R. Canetti, S. Halevi, and J. Katz. A forward-secure public-key encryption scheme. In *Proc. of EUROCRYPT 2003*, volume 2656 of *LNCS*, pages 255–271. Springer, 2003.

[34] R. Canetti, S. Halevi, and J. Katz. Chosen-ciphertext security from identity-based encryption. In *Proc. of EUROCRYPT 2004*, volume 3027 of *LNCS*, pages 207–222. Springer, 2004.

[35] R. Canetti, H. Krawczyk, and J.B. Nielsen. Relaxing chosen-ciphertext security. In *Proc. of CRYPTO 2003*, volume 2729 of *LNCS*, pages 565–582. Springer, 2003.

[36] R. Canetti, D. Micciancio, and O. Reingold. Perfectly one-way probabilistic hash functions (preliminary version). In *Proc. of STOC 1998*, pages 131–140. ACM, 1998.

[37] D. Cash, E. Kiltz, and V. Shoup. The twin Diffie-Hellman problem and applications. In *Proc. of EUROCRYPT 2008*, volume 4965 of *LNCS*, pages 127–145. Springer, 2008.

[38] S.G. Choi, D. Dachman-Soled, T. Malkin, and H. Wee. Black-box construction of a non-malleable encryption scheme from any semantically secure one. In *Proc. of TCC 2008*, volume 4948 of *LNCS*, pages 427–444. Springer, 2008.

[39] C. Cocks. An identity based encryption scheme based on quadratic residues. In *Proc. of Cryptography and Coding 2001*, volume 2260 of *LNCS*, pages 360–363. Springer, 2001.

[40] R. Cramer, G. Hanaoka, D. Hofheinz, H. Imai, E. Kiltz, R. Pass, a. shelat, and V. Vaikuntanathan. Bounded CCA2-secure encryption. In *Proc. of ASIACRYPT 2007*, volume 4833 of *LNCS*, pages 502–518. Springer, 2007.

[41] R. Cramer, D. Hofheinz, and E. Kiltz. A twist on the Naor-Yung paradigm and its application to efficient CCA-secure encryption from hard search problems. In *Proc. of TCC 2010*, volume 5978 of *LNCS*, pages 146–164. Springer, 2010.

[42] R. Cramer and V. Shoup. A practical public key cryptosystem provably secure against adaptive chosen ciphertext attack. In *Proc. of CRYPTO 1998*, volume 1462 of *LNCS*, pages 13–25. Springer, 1998.

[43] R. Cramer and V. Shoup. Universal hash proofs and a paradigm for adaptive chosen ciphertext secure public-key encryption. In *Proc. of EUROCRYPT 2002*, volume 2332 of *LNCS*, pages 45–64. Springer, 2002.

[44] I. Damgård. Collision free hash functions and public key signature schemes. In *Proc. of EUROCRYPT 1987*, volume 304 of *LNCS*, pages 203–216. Springer, 1988.

[45] Y. Dodis and J. Katz. Chosen-ciphertext security of multiple encryption. In *Proc. of TCC 2005*, volume 3378 of *LNCS*, pages 189–209. Springer, 2005.

[46] Y. Dodis, R. Oliveira, and K. Pietrzak. On the generic insecurity of the full domain hash. In *Proc. of CRYPTO 2005*, volume 3621 of *LNCS*, pages 449–466. Springer, 2005.

[47] D. Dolev, C. Dwork, and M. Naor. Non-malleable cryptography. In *Proc. of STOC 1991*, pages 542–552. ACM, 1991.

[48] R. Dowsley, J. Müller-Quade, and A.C.A. Nascimento. A CCA2 secure public key encryption scheme based on the McEliece assumptions in the standard model. In *Proc. of CT-RSA 2009*, volume 5473 of *LNCS*, pages 240–251. Springer, 2009.

[49] E. Fujisaki and T. Okamoto. How to enhance the security of public-key encryption at minimum cost. In *Proc. of PKC 1999*, volume 1560 of *LNCS*, pages 53–68. Springer, 1999.

[50] D. Galindo. A separation between selective and full-identity security notions for identity-based encryption. In *Proc. of ICCSA 2006*, volume 3982 of *LNCS*, pages 318–326. Springer, 2006.

[51] C. Gentry. Practical identity-based encryption without random oracles. In *Proc. of EUROCRYPT 2006*, volume 4004 of *LNCS*, pages 445–464. Springer, 2006.

[52] C. Gentry and S. Halevi. Herarchical identity based encryption with polynomially many levels. In *Proc. of TCC 2009*, volume 5444 of *LNCS*, pages 437–456. Springer, 2009.

[53] C. Gentry, C. Peikert, and V. Vaikuntanathan. Trapdoors for hard lattices and new cryptographic constructions. In *Proc. of STOC 2008*, pages 197–206. ACM, 2008.

[54] C. Gentry and A. Silverberg. Hierarchical ID-based cryptography. In *Proc. of ASI-ACRYPT 2002*, volume 2501 of *LNCS*, pages 548–566. Springer, 2002.

[55] Y. Gertner, T. Malkin, and S. Myers. Towards a separation of semantic and CCA security for public key encryption. In *Proc. of TCC 2007*, volume 4392 of *LNCS*, pages 434–455. Springer, 2007.

[56] O. Goldreich. *Foundations of Cryptography - Volume 2*. Cambridge University Press, 2004.

[57] O. Goldreich. Basing non-interactive zero-knowledge on (enhanced) trapdoor permutations: The state of the art, 2008. Available at www.wisdom.weizmann.ac.il/~oded/foc-vol2.html.

[58] O. Goldreich and L.A. Levin. Hardcore predicate for all one-way functions. In *Proc. of STOC 1989*, pages 25–32. ACM, 1989.

[59] S. Goldwasser and Y.T. Kalai. On the (in)security of the Fiat-Shamir paradigm. In *Proc. of FOCS 2003*, pages 102–113. IEEE Computer Society Press, 2003.

[60] S. Goldwasser and S. Micali. Probabilistic encryption. *J. of Computer and System Sciences*, 28(2):270–299, 1984.

[61] G. Hanaoka and K. Kurosawa. Efficient chosen ciphertext secure public key encryption under the computational Diffie-Hellman assumption. In *Proc. of ASIACRYPT 2008*, volume 5350 of *LNCS*, pages 308–325. Springer, 2008.

[62] K. Haralambiev, T. Jager, E. Kiltz, and V. Shoup. Simple and efficient public-key encryption from computational Diffie-Hellman in the standard model. In *Proc. of PKC 2010*, volume 6056 of *LNCS*, pages 279–295. Springer, 2010.

[63] J. Håstad, R. Impagliazzo, L. Levin, and M. Luby. Construction of a pseudorandom generator from any one-way function. *SIAM J. Computing*, 28(4):1364–1396, 1999.

[64] J. Herranz, D. Hofheinz, and E. Kiltz. KEM/DEM: Necessary and sufficient conditions for secure hybrid encryption, 2006. Available at eprint.iacr.org/2006/265.

[65] D. Hofheinz and E. Kiltz. Secure hybrid encryption from weakened key encapsulation. In *Proc. of CRYPTO 2007*, volume 4622 of *LNCS*, pages 553–571. Springer, 2007.

[66] D. Hofheinz and E. Kiltz. The group of signed quadratic residues and applications. In *Proc. of CRYPTO 2009*, volume 5677 of *LNCS*, pages 637–653. Springer, 2009.

[67] D. Hofheinz and E. Kiltz. Practical chosen ciphertext secure encryption from factoring. In *Proc. of EUROCRYPT 2009*, volume 5479 of *LNCS*, pages 313–332. Springer, 2009.

[68] J. Horwitz and B. Lynn. Toward hierarchical identity-based encryption. In *Proc. of EUROCRYPT 2002*, volume 2332 of *LNCS*, pages 466–481. Springer, 2002.

[69] E. Kiltz. Chosen-ciphertext security from tag-based encryption. In *Proc. of TCC 2006*, volume 3876 of *LNCS*, pages 581–600. Springer, 2006.

[70] E. Kiltz. Chosen-ciphertext secure key-encapsulation based on gap hashed Diffie-Hellman. In *Proc. of PKC 2007*, volume 4450 of *LNCS*, pages 282–297. Springer, 2007.

[71] E. Kiltz, P. Mohassel, and A. O'Neill. Adaptive trapdoor functions and chosen-ciphertext security. In *Proc. of EUROCRYPT 2010*, volume 6110 of *LNCS*, pages 673–692. Springer, 2010.

[72] E. Kiltz and K. Pietrzak. On the security of padding-based encryption schemes - or - why we cannot prove OAEP secure in the standard model. In *Proc. of EUROCRYPT 2009*, volume 5479 of *LNCS*, pages 389–406. Springer, 2009.

[73] E. Kiltz, K. Pietrzak, M. Stam, and M. Yung. A new randomness extraction paradigm for hybrid encryption. In *Proc. of EUROCRYPT 2009*, volume 5479 of *LNCS*, pages 590–609. Springer, 2009.

[74] H. Krawczyk and T. Rabin. Chameleon hashing and signatures. In *Proc. of NDSS 2000*. Internet Society, 2000.

[75] K. Kurosawa and Y. Desmedt. A new paradigm of hybrid encryption scheme. In *Proc. of CRYPTO 2004*, volume 3152 of *LNCS*, pages 426–442. Springer, 2004.

[76] A.B. Lewko and B. Waters. New techniques for dual system encryption and fully secure HIBE with short ciphertexts. In *Proc. of TCC 2010*, volume 5978 of *LNCS*, pages 455–479. Springer, 2010.

[77] Y. Lindell. A simpler construction of CCA2-secure public-key encryption under general assumptions. *J. of Cryptology*, 19(3):359–377, 2006.

[78] P. MacKenzie, M.K. Reiter, and K. Yang. Alternatives to non-malleability: Definitions, constructions and applications. In *Proc. of TCC 2004*, volume 2951 of *LNCS*, pages 171–190. Springer, 2004.

[79] P. MacKenzie, M.K. Reiter, and K. Yang. Alternatives to non-malleability: Definitions, constructions and applications, 2004. Full version of [78]. Available at www.cs.cmu.edu/˜yangke/papers/nm.pdf.

[80] T. Matsuda, G. Hanaoka, K. Matsuura, and H. Imai. An efficient encapsulation scheme from near collision resistant pseudorandom generators and its application to IBE-to-PKE transformations. In *Proc. of CT-RSA 2009*, volume 5473 of *LNCS*, pages 16–31. Springer, 2009.

[81] S. Myers and a. shelat. Bit encryption is complete. In *FOCS 2009*, pages 607–616. IEEE Computer Society Press, 2009.

[82] W. Nagao, Y. Manabe, and T. Okamoto. On the equivalence of several security notions of KEM and DEM. *IEICE Transactions*, E91-A(1):283–297, 2008.

[83] M. Naor and M. Yung. Universal one-way hash functions and their cryptographic applications. In *Proc. of STOC 1989*, pages 33–43. ACM, 1989.

[84] M. Naor and M. Yung. Public-key cryptosystems provably secure against chosen ciphertext attacks. In *Proc. of STOC 1990*, pages 427–437. ACM, 1990.

[85] J.B. Nielsen. Separating random oracle proofs from complexity theoretic proofs: The non-committing encryption case. In *Proc. of CRYPTO 2002*, volume 2442 of *LNCS*, pages 111–126. Springer, 2002.

[86] R. Nojima, H. Imai, K. Kobara, and K. Morozov. Semantic security for the McEliece cryptosystem without random oracles. *Designs, Codes and Cryptography*, 49(1-3):289–305, 2008.

[87] T. Okamoto and D. Pointcheval. REACT: Rapid enhanced-security asymmetric cryptosystem transform. In *Proc. of CT-RSA 2001*, volume 2020 of *LNCS*, pages 159–174. Springer, 2001.

[88] O. Pandey, R. Pass, and V. Vaikuntanathan. Adaptive one-way functions and applications. In *Proc. of CRYPTO 2008*, volume 5157 of *LNCS*, pages 57–74. Springer, 2008.

[89] R. Pass, a. shelat, and V. Vaikuntanathan. Construction of a non-malleable encryption scheme from any semantically secure one. In *Proc. of CRYPTO 2006*, volume 4117 of *LNCS*, pages 271–289. Springer, 2006.

[90] R. Pass, a. shelat, and V. Vaikuntanathan. Relations among notions of non-malleability for encryption. In *Proc. of ASIACRYPT 2007*, volume 4833 of *LNCS*, pages 519–535. Springer, 2007.

[91] C. Peikert and B. Waters. Lossy trapdoor functions and their applications. In *Proc. of STOC 2008*, pages 187–196. ACM, 2008.

[92] D.H. Phan and D. Pointcheval. On the security notions for public-key encryption schemes. In *Proc. of SCN 2004*, volume 3352 of *LNCS*, pages 33–46. Springer, 2005.

[93] C. Rackoff and D.R. Simon. Non-interactive zero-knowledge proof of knowledge and chosen ciphertext attack. In *Proc. of CRYPTO 1991*, volume 576 of *LNCS*, pages 433–444. Springer, 1992.

[94] O. Reingold, L. Trevisan, and S. Vadhan. Notions of reducibility between cryptographic primitives. In *Proc. of TCC 2004*, volume 2951 of *LNCS*, pages 1–20. Springer, 2004.

[95] A. Rosen and G. Segev. Chosen-ciphertext security via correlated products. In *Proc. of TCC 2009*, volume 5444 of *LNCS*, pages 419–436. Springer, 2009.

[96] A. Sahai. Non-malleable non-interactive zero knowledge and adaptive chosen-ciphertext security. In *Proc. of FOCS 1999*, pages 543–553. IEEE Computer Society Press, 1999.

[97] R. Sakai, K. Ohgishi, and M. Kasahara. Cryptosystems based on pairing over elliptic curve (in japanese). In *Proc. of SCIS 2001*, 2001.

[98] H. Shacham. A Cramer-Shoup encryption scheme from the linear assumption and from progressively weaker linear variants, 2007. Available at eprint.iacr.org/2007/074/.

[99] A. Shamir. Identity-based cryptosystems and signature schemes. In *Proc. of CRYPTO 1984*, volume 196 of *LNCS*, pages 47–53. Springer, 1985.

[100] V. Shoup. A proposal for an ISO standard for public key encryption (version 2.1), 2001. Available at shoup.net/papers/.

[101] X. Wang, Y.L. Yin, and H. Yu. Finding collisions in the full SHA-1. In *Proc. of CRYPTO 2005*, volume 3621 of *LNCS*, pages 12–36. Springer, 2005.

[102] B. Waters. Efficient identity-based encryption without random oracles. In *Proc. of EUROCRYPT 2005*, volume 3494 of *LNCS*, pages 114–127. Springer, 2005.

[103] B. Waters. Dual system encryption: Realizing fully secure IBE and HIBE under simple assumptions. In *Proc. of EUROCRYPT 2009*, volume 5677 of *LNCS*, pages 619–636. Springer, 2009.

[104] H. Wee. Efficient chosen-ciphertext security via extractable hash proofs. In *Proc. of CRYPTO 2010*, volume 6223 of *LNCS*, pages 314–332. Springer, 2010.

[105] A.C. Yao. Theory and application of trapdoor functions. In *Proc. of FOCS 1982*, pages 80–91. IEEE Computer Society Press, 1982.

[106] R. Zhang. Tweaking TBE/IBE to PKE transforms with chameleon hash functions. In *Proc. of ACNS 2007*, volume 4521 of *LNCS*, pages 323–339. Springer, 2007.

Appendix A

Publication List

International Conference Papers (included in this thesis):

a. Takahiro Matsuda, Kanta Matsuura. "Parallel Decryption Queries in Bounded Chosen Ciphertext Attacks," To appear in the 14th IACR International Conference on Practice and Theory in Public Key Cryptography (PKC 2011).

b. Takahiro Matsuda, Goichiro Hanaoka, Kanta Matsuura, Hideki Imai. "An Efficient Encapsulation Scheme from Near Collision Resistant Pseudorandom Generators and Its Application to IBE-to-PKE Transformations," Topics in Cryptology - CT-RSA 2009, Lecture Notes in Computer Science, vol. 5473, Springer, pp. 16-31, 2009.

c. Takahiro Matsuda, Goichiro Hanaoka, Kanta Matsuura, Hideki Imai. "Simple CCA-Secure Public Key Encryption from Any Non-Malleable Identity-Based Encryption," Information Security and Cryptology - ICISC 2008, Seoul, Lecture Notes in Computer Science, vol. 5461, Springer, pp. 1-19, 2009.

Journal and International Conference Papers (not included in this thesis):

- Takahiro Matsuda, Kanta Matsuura. "On Black-Box Separations among Injective One-Way Functions," To appear in the 8th Theory of Cryptography Conference (TCC 2011).

- Takahiro Matsuda, Yasumasa Nakai, Kanta Matsuura. "Efficient Generic Constructions of Timed-Release Encryption with Pre-open Capability," Pairing-Based Cryptography - Pairing 2010, Lecture Notes in Computer Science, vol. 6487, Springer, pp. 225-245, 2010.

- Takahiro Matsuda, Kanta Matsuura, Jacob C.N. Schuldt. "Efficient Constructions of Signcryption Schemes and Signcryption Composability," Progress in Cryptology - INDOCRYPT 2009, Lecture Notes in Computer Science, vol. 5922, Springer, pp. 321-342, 2009.

- Yasumasa Nakai, Takahiro Matsuda, Wataru Kitada, Kanta Matsuura. "A Generic Construction of Timed-Release Encryption with Pre-open Capability," Advances in Information and Computer Security, Lecture Notes in Computer Science, vol. 5824, Springer, pp. 53-70, 2009.

- Takahiro Matsuda, Nuttapong Attrapadung, Goichiro Hanaoka, Kanta Matsuura, Hideki Imai. "A Strongly Unforgeable Signature under the CDH Assumption without Collision Resistant Hash Functions," IEICE Trans. on Information and Systems, vol. E91-D, no. 5, pp. 1466-1476, May, 2008.

- Takahiro Matsuda, Nuttapong Attrapadung, Goichiro Hanaoka, Kanta Matsuura, Hideki Imai. "A CDH-Based Strongly Unforgeable Signature Without Collision Resistant Hash Function," Provable Security, Lecture Notes in Computer Science, vol. 4784, Springer, pp. 68-84, 2007.

- Takahiro Matsuda, Goichiro Hanaoka, Kanta Matsuura, Hideki Imai. "A Practical Provider Authentication System for Bidirectional Broadcast Service," Knowledge-Based Interlligent Information and Engineering Systems, Lecture Notes in Computer Science, vol. 4694, Springer, pp. 967-974, 2007

Publications at Domestic Conferences:

- 江村恵太, 花岡悟一郎, 川合豊, 松田隆宏, 面和成, 坂井祐介. "メッセージ依存開示可能グループ署名と匿名掲示板への応用," 2011 年暗号と情報セキュリティシンポジウム (SCIS 2011), 3A1-4, 福岡. 2011 年・1 月.

- 千葉大輝, 松田隆宏, シュルツヤコブ, 松浦幹太. "多人数モデルで内部者安全な Signcryption の一般的構成法," 2011 年暗号と情報セキュリティシンポジウム (SCIS 2011), 2A2-4, 福岡. 2011 年・1 月.

- 松田隆宏, 花岡悟一郎, 松浦幹太. "KEM の Constrained CCA 安全性と回数制限付き CCA 安全性の関係," 2011 年暗号と情報セキュリティシンポジウム (SCIS 2011), 2A2-3, 福岡. 2011 年・1 月.

- 松田隆宏, 松浦幹太. "単一型と並行型の復号クエリを考慮した回数制限付き選択暗号文攻撃に対する安全性定義間の関係," 2011 年暗号と情報セキュリティシンポジウム (SCIS 2011), 2A1-1, 福岡. 2011 年・1 月.

- 松田隆宏, 松浦幹太. "単写の一方向関数のブラックボックス構成の不可能性について," 2011 年暗号と情報セキュリティシンポジウム (SCIS 2011), 1A1-2, 福岡. 2011 年・1 月.

- 松田隆宏, 松浦幹太. "開封時刻の秘匿性を持つ事前開封機能付きタイムリリース暗号の一般的な構成法," コンピュータセキュリティシンポジウム 2010 (CSS 2010), 3B1-3, 岡山. 2010 年・10 月.

- 小田哲, 永井彰, 山本剛, 小林鉄太郎, 冨士仁, 中井泰雅, 松田隆宏, 松浦幹太. "汎用 IBE 向けシステムの構成法とその実装," 2010 年暗号と情報セキュリティシンポジウム (SCIS 2010). 4A2-5. 香川. 2010 年・1 月.

- 松田隆宏, 松浦幹太. "Mixed CCA 安全性: より強い安全性を持つ公開鍵暗号方式の CPA 安全な方式のみを用いた構成," 2010 年暗号と情報セキュリティシンポジウム (SCIS 2010). 4A1-2. 香川. 2010 年・1 月.

- 松田隆宏, 松浦幹太. "公開鍵暗号の回数制限付き並行型選択暗号文攻撃に対する安全性," 2010 年暗号と情報セキュリティシンポジウム (SCIS 2010). 4A1-1. 香川. 2010 年・1 月.

- 千葉大輝, 松田隆宏, 松浦幹太. “タグベース KEM の選択的タグ安全性から適応的タグ安全性へのカメレオンハッシュを用いた強化手法と Signcryption への応用,” 2010 年暗号と情報セキュリティシンポジウム (SCIS 2010). 3A2-1. 香川. 2010 年・1 月.
- 中井泰雅, 松田隆宏, 松浦幹太. “時間前復号機能付き時限式暗号の効率的な一般的構成法,” 2010 年暗号と情報セキュリティシンポジウム (SCIS 2010). 2C3-1. 香川. 2010 年・1 月.
- 松田隆宏, シュルツヤコブ, 松浦幹太. “多人数環境を考慮した Signcryption の簡潔な一般的構成法” コンピュータセキュリティシンポジウム 2009(CSS 2009). B4-2. 富山. 2009 年・10 月.
- 中井泰雅, 松田隆宏, 北田亘, 松浦幹太. “時間前開封機能付き時限式暗号の一般的構成法,” 2009 年暗号と情報セキュリティシンポジウム (SCIS 2009). 3F2-3. 滋賀. 2009 年・1 月.
- 松田隆宏, 花岡悟一郎, 松浦幹太, 今井秀樹. “効率の良い Encapsulation 方式と IBE-to-PKE 変換への応用,” 2009 年暗号と情報セキュリティシンポジウム (SCIS 2009). 2B2-5. 滋賀. 2009 年・1 月.
- 松田隆宏, 花岡悟一郎, 松浦幹太, 今井秀樹. “任意の頑強な ID ベース暗号に基づく CCA 安全な公開鍵暗号の効率的構成方法,” 2008 年暗号と情報セキュリティシンポジウム (SCIS 2008). 2F1-1. 宮崎, 2008 年・1 月.
- 松田隆宏, アッタラパドゥンナッタポン, 花岡悟一郎, 松浦幹太, 今井秀樹.” “スタンダードモデルでの CDH 仮定に基づく衝突困難ハッシュ関数を用いない強偽造不可能性を持つ署名方式,” 2007 年暗号と情報セキュリティシンポジウム (SCIS 2007). 3C4-4. 長崎, 2007 年・1 月.

Appendix B

The Existing PKE Constructions

Here, we recall the existing PKE constructions from IND-CPA secure PKE schemes we mention in this paper.

B.1 The CDMW Construction

Here, we recall the black-box construction by Choi et al. [38] of an NM-CPA secure PKE scheme from any IND-CPA secure PKE scheme.

Let $\Pi = (\mathsf{PKG}, \mathsf{PEnc}, \mathsf{PDec})$ be a PKE scheme, $\Sigma = (\mathsf{SKG}, \mathsf{Sign}, \mathsf{SVrfy})$ be a signature scheme in which we assume that the length of a verification key vk is k, when generated from $\mathsf{SKG}(1^k)$. Define a code $\mathcal{W}$ over the alphabet $\{0,1\}^n$ as follows:

$$\mathcal{W} = \{\ (p(1), \ldots, p(10k)) \mid p \text{ is a degree-}k \text{ polynomial }\},$$

which is the Reed-Solomon code with minimum relative distance 0.9, and thus we can correct up to 0.45 fraction errors. (It is known that there exists an efficient decoding algorithm.)

Then the PKE scheme $\Pi_{\mathtt{CDMW'}} = (\mathsf{PKG}_{\mathtt{CDMW'}}, \mathsf{PEnc}_{\mathtt{CDMW'}}, \mathsf{PDec}_{\mathtt{CDMW'}})$ by Choi et al. [38] is constructed as shown in Figure B.1. (We call this construction the *CDMW' PKE* scheme, in order to distinguish from the enhanced version of this scheme below.) It was shown in [38] that if the underlying PKE scheme Π is IND-CPA secure, and the underlying signature scheme Σ is strongly one-time secure, then the PKE scheme $\Pi_{\mathtt{CDMW'}}$ is NM-CPA secure.

Moreover, it was also stated in [38] that if we change the size of the set S and the degree of the polynomial p from k to $8(k+q)$, and the number of columns in C_{mat} from $10k$ to $80(k+q)$, then the resulting PKE scheme achieves NM-q-CCA2 security (see [38] for details). We call this enhanced scheme the *CDMW PKE* scheme.

B.2 The CHH+ Construction

Here, we recall the black-box construction by Cramer et al. [40] of an IND-q-CCA2 secure PKE scheme from any IND-CPA secure PKE scheme.

First we briefly review the definition of a *cover free family* which is used as a building box of their construction. Let $F = \{F_i\}_{1 \le i \le s}$ be subsets over the indices $\{1, \ldots, d\}$ such that $|F_i| = l$ for all $1 \le i \le s$ (such family F is called l-uniform). We say that F is q-cover-free over $\{1, \ldots, d\}$ if $F_i \not\subseteq \bigcup_{j \in S_i} F_j$, for all $i \in \{1, \ldots, s\}$ and for all $S_i \subseteq \{1, \ldots, s\} \backslash \{i\}$ such

$\mathsf{PKG}_{\mathtt{CDMW'}}(1^k)$:	$\mathsf{PEnc}_{\mathtt{CDMW'}}(PK, m)$:
$(pk_{i,j}^{(b)}, sk_{i,j}^{(b)}) \leftarrow \mathsf{PKG}(1^k)$ for $1 \leq i \leq k$, $1 \leq j \leq 10k$, $b \in \{0,1\}$ Pick a set of indices $S \subset \{1, \ldots, 10k\}$ uniformly such that $\lvert S \rvert = k$. $PK \leftarrow (\{pk_{i,j}^{(b)}\})$ $SK \leftarrow (\{sk_{i,j}^{(b)}\}, S)$ Return (PK, SK).	Parse PK as $\{pk_{i,j}^{(b)}\}$. $a_1, \ldots a_k \leftarrow \{0,1\}^k$ Define $p(x) = a_k x^k + \cdots + a_1 x + m$ over $GF(2^k)$. $s_j \leftarrow p(j)$ for $1 \leq j \leq 10k$ $(vk, sigk) \leftarrow \mathsf{SKG}(1^k)$ View vk as a k-bit string $(v_1 \| \ldots \| v_k)$. $c_{i,j} \leftarrow \mathsf{PEnc}(pk_{i,j}^{v_i}, s_j)$ for $1 \leq i \leq k$, $1 \leq j \leq 10k$ $C_{mat} \leftarrow \{c_{i,j}\}$ $\sigma \leftarrow \mathsf{Sign}(sigk, C_{mat})$ Return $C \leftarrow (vk, C_{mat}, \sigma)$.
$\mathsf{PDec}_{\mathtt{CDMW'}}(SK, C)$: Parse SK as $(\{sk_{i,j}^{(b)}\}, S)$ and C as (vk, C_{mat}, σ). (1) Check if $\mathsf{SVrfy}(vk, C_{mat}, \sigma) = \mathtt{accept}$. Parse C_{mat} as $\{c_{i,j}\}$ and view vk as a k-bit string $(v_1 \| \ldots \| v_k)$. $s_j \leftarrow \mathsf{PDec}(sk_{1,j}^{(v_1)}, c_{1,j})$ for $1 \leq j \leq 10k$ Find a codeword $w = (w_1, \ldots, w_{10k}) \in \mathcal{W}$ which agrees with $(s_1, \ldots, s_{10k})$ in at least $9k$ positions. (If no such codeword is found, return $\perp$.) (2) Check if $\mathsf{PDec}(sk_{1,j}^{(v_1)}, c_{1,j}) = \cdots = \mathsf{PDec}(sk_{k,j}^{(v_k)}, c_{k,j})$ holds for all $j \in S$. (3) Check if $s_j = w_j$ for all $j \in S$. If the checks (1) to (3) all accept, return m that corresponds to the codeword w else return $\perp$.	

Figure B.1: The CDMW' PKE construction $\Pi_{\mathtt{CDMW'}}$.

that $|S_i| \leq q$. It is known that there is a deterministic polynomial time algorithm which on input s and q returns (l, d, F), where $F = \{F_i\}_{1 \leq i \leq s}$ is a l-uniform q-cover-free family over $\{1, \ldots, d\}$. Moreover, let SUB denote the resulting deterministic polynomial-time algorithm that on input s, q, and i where $i \in \{1, \ldots, s\}$ returns F_i. See [40] and references therein for details of such cover free families. We set $s = 2^k$, $d = 16kq^2$, and $l = 4kq$, where k is a security parameter and q is the upperbound of decryption queries we expect the construction to be resistant against.

Using the above tool, we now recall the construction of a PKE scheme by Cramer et al. [40] (which we call the *CHH+ PKE* scheme). Let $\Pi = (\mathsf{PKG}, \mathsf{PEnc}, \mathsf{PDec})$ be a PKE scheme, $\Sigma = (\mathsf{SKG}, \mathsf{Sign}, \mathsf{SVrfy})$ be a signature scheme in which we assume that the length of a verification key vk is k, when generated from $\mathsf{SKG}(1^k)$. Then the CHH+ PKE scheme $\Pi_{\mathtt{CHH+}} = (\mathsf{PKG}_{\mathtt{CHH+}}, \mathsf{PEnc}_{\mathtt{CHH+}}, \mathsf{PDec}_{\mathtt{CHH+}})$ is constructed as shown in Figure B.2. It was shown in [40] that if the underlying PKE scheme Π is `IND-CPA` secure, the underlying signature scheme Σ is strongly one-time secure, and the parameters (d, q, l) are as explained above, then the PKE scheme $\Pi_{\mathtt{CHH+}}$ is `IND-`q`-CCA2` secure.

Moreover, it was also stated in [40] (Remark 2 after the proof of Lemma 1 in [40]) that if we replace the underlying `IND-CPA` secure PKE scheme with `IND-CCA1` secure one, then the resulting PKE scheme achieves $\langle \mathsf{unbound} :: \mathsf{s}^q \rangle$`-mCCA` security (the notations used here are defined in Section 4.2).

$\mathsf{PKG}_{\mathtt{CHH+}}(1^k)$:	$\mathsf{PEnc}_{\mathtt{CHH+}}(PK, m)$:	$\mathsf{PDec}_{\mathtt{CHH+}}(SK, C)$:
$(pk_i, sk_i) \leftarrow \mathsf{PKG}(1^k)$ for $1 \le i \le d$	Parse PK as $\{pk_i\}$.	Parse SK as $\{sk_i\}$.
$PK \leftarrow \{pk_i\}$	$(vk, sigk) \leftarrow \mathsf{SKG}(1^k)$	Parse C as (vk, C_{vec}, σ).
$SK \leftarrow \{sk_i\}$	$F_{vk} \leftarrow \mathsf{SUB}(2^k, q, vk)$	If $\mathsf{SVrfy}(vk, C_{vec}, \sigma) = \mathtt{reject}$
Return (PK, SK).	Let $F_{vk} = (s_1, \ldots, s_l)$.	then return $\perp$.
	Pick $m_1, \ldots, m_l$ randomly	Parse C_{vec} as $(c_1, \ldots, c_l)$.
	such that $m = \bigoplus_{i=1}^{l} m_i$.	$F_{vk} \leftarrow \mathsf{SUB}(2^k, q, vk)$
	$c_i \leftarrow \mathsf{PEnc}(pk_{s_i}, m_i)$ for $1 \le i \le l$	Let $F_{vk} = (s_1, \ldots, s_l)$.
	$C_{vec} \leftarrow (c_1, \ldots, c_l)$	$m_i \leftarrow \mathsf{PDec}(sk_{s_i}, c_i)$ for $1 \le i \le l$
	$\sigma \leftarrow \mathsf{Sign}(sigk, C_{vec})$	If $\exists j \in \{1, \ldots, l\}$ such that $m_j = \perp$
	Return $C \leftarrow (vk, C_{vec}, \sigma)$.	then return $\perp$.
		Return $m \leftarrow \bigoplus_{i=1}^{l} m_i$.

Figure B.2: The CHH+ PKE construction $\Pi_{\mathtt{CHH+}}$.