Cryptography with Timed Access Control

(時刻制御暗号技術)

A dissertation

by

Yohei Watanabe

Supervisor: Asso. Prof. Dr. Junji Shikata

Graduate School of Environment and Information Sciences
Yokohama National University

March 2016

Abstract

"Time" is intimately related to our lives. We get up, eat something, do a job, and get asleep at time of our (or someone's) choice. Moreover, many industrial systems have been automated according to schedule (i.e. time), and such automation have accelerated progress on the modern industrial society. For the above reason, it appears that cryptographic techniques associated with "time", which we call *cryptographically timed access control*, are useful and meaningful. In this thesis, we deal with cryptography with timed access control, where an entity can specify when other entity's functionality is activated.

Specifically, we consider two types of cryptographically timed access controls in terms of available periods of target functionality. One is the *timed-release cryptography*, which is well known and has been investigated by many researchers so far. The goal of timed-release cryptography is to send certain information *into the future*. More precisely, a sender can control when receiver's functionality is available. For example, in timed-release encryption, a sender can encrypt a plaintext by designating time when a receiver can decrypt the encrypted plaintext, and even the legitimate receiver cannot decrypt the ciphertext until the designated time comes. The other is *timed-revocable cryptography* in which an entity (e.g, a sender or a third party) can revoke (i.e. inactivate) other entity's functionality (e.g., decryption) in the middle of the protocol. In the general setting, such functionality cannot be realized since a receiver has both his secret key and ciphertexts (i.e., the sender directly sends ciphertexts to the receiver). Therefore, we consider such functionality in the cloud environment setting (i.e., the receiver has only his secret key, and ciphertexts are stored in cloud storage). Recent progress of cloud technologies has been remarkable, and thus it is highly significant to consider the timed-revocable cryptography.

We consider the above cryptographic properties from the perspectives of two major security criteria, *information-theoretic security* and *computational security*. Information-theoretic security provides the strongest security, namely, information-theoretically secure protocols are secure even if an adversary has *infinite computational power*. In other words, realization of quantum computers and development of computational algorithms do not affect security of information-theoretically secure protocols. Hence, cryptographic protocols with information-theoretic security can provide long-term security.

On the other hand, although we have to assume that all adversaries are *computationally bounded ones* (i.e., polynomial time Turing machines) has to be assumed, computational security can realize useful and practical protocols such as public-key mechanisms, and efficient protocols in terms of secret-information sizes. In fact, most of cryptographic protocols used in the real world are computationally secure. As can be seen, the above two security criteria have both merits and demerits, and thus it is important to investigate cryptographic protocols from both perspectives.

Specifically, contributions of this thesis are as follows.

- In the computational security setting, many research papers on timed-release security have been reported so far. Toward developing computational timed-release cryptography, we first propose a timed-release computational secret sharing (TR-CSS) scheme as a new timed-release fundamental primitive. We also show the TR-CSS scheme can provide new timed-release protocols, timed-release multiple encryption and threshold encryption.

- We first introduce information-theoretic timed-release cryptography. We show how we realize information-theoretically secure fundamental cryptographic primitives with timed-release functionalities. We succeed in adding the functionalities to the fundamental primitives, and propose timed-release key-agreement (TR-KA), encryption (TRE), authentication codes (TRA-codes), and secret sharing (TR-SS). If a sender wants to transmit a message far into the future, information-theoretic security will be helpful in constructing timed-release mechanisms, since its security can provide the long-term security.

- Finally, we first design and analyze the timed-revocable cryptography with information-theoretic security. We consider broadcast encryption (BE), which provides flexible access control and is suited to the cloud environment, with timed-revocable security by assuming that ciphertexts are stored in cloud storage. Therefore, we propose revocable-storage BE (RS-BE) with information-theoretic security. Although it is known that there are trade-offs between secret-key sizes and ciphertext sizes in BE schemes, the RS-BE scheme only captures the case of the smallest ciphertext size since ciphertexts are stored in cloud storage for a long time. Hence, as a step toward an RS-BE scheme with more general ciphertext sizes, we first propose BE schemes with general ciphertext sizes.

Acknowledgments

This thesis would have been impossible without the support and mentoring of my adviser, Asso. Prof. Junji Shikata. Since I started doing research on modern cryptography, he has given me lots of things. For example, he showed me how to do research, write papers, and give talks. He always devoted time for discussions with me and often helped me clear my garbled ideas and explanations, and consequently I was able to open an avenue of my research. Outside of work, he took me several great restaurants and izakayas (Japanese-style pub), and I had a wonderful time every time. His door has been always open, and I would often ask him for some advice about my personal life.

I am very thankful to Prof. Tsutomu Matsumoto and Asso. Prof. Katsunari Yoshioka for supporting me despite their busy schedules. Prof. Matsumoto always saw my research from a higher perspective, and gave me fruitful comments. Further, he constantly cared about me, in particular, my research progress and my future post. I would like to appreciate him for the concern. Looking back now, I would never choose information security as my research area without attending Asso. Prof. Yoshioka's class during my undergraduate course. I can recall that I wanted to be assigned to Yoshioka laboratory because of his personality and his attractive presentation, though I did not choose network security but cryptology as my research interest.

I am also indebted to Prof. Tatsunori Mori and Prof. Tomoharu Nagao for reviewing this thesis. Despite their different research areas, their insightful comments have significantly improved this thesis.

I thank my collaborators on works both in this thesis and outside (titles are omitted): Keita Emura, Shogo Hajime, Goichiro Hanaoka, Junichi Ida, Yuu Ishida, Marina Kasai, Kazuyuki Kinose, Asato Kubai, Le Trieu Phong, Takenobu Seito, Noriyasu Takei, Shinichiro Tomita, and Takahiro Yoshizawa.

I have had helpful discussions and received comments and suggestions from all members in Shikata laboratory. Dr. Takenobu Seito (now at Bank of Japan) has been particularly helpful in terms of optimizing my own performance. He gave me lots of advice about the contents of the research. During my master's course, I received lots of comments and support from Daisuke Inoue (now at NEC Corporation), Hirokazu Tagai (now at Toshiba Solutions Corporation), and Masato Hata (now at Ericsson Japan K.K.), who were my colleagues. The discussion has led to significant performance optimizations.

I am thankful to Technical Professional, Mr. Yasuyuki Mori, and Secretaries, Ms. Mio Narimatsu, Ms. Tomoko Ishidate, Ms. Kumiko Nakayama, and Ms. Satono Yoshitani who had great assistance offered in smoothly researching. Especially, Ms. Narimatsu and Ms. Ishidate would often visit my office to have a coffee break, and chatting with them refreshed me.

I was very fortunate to work as a short-term intern at Secure Computing Lab. in Fujitsu Laboratories ltd. during November and December 2011. I would like to thank Dr. Tetsuya Izu for giving me such an opportunity for the intern, and Mr. Ikuya Morikawa for gently mentoring me. I would also like to thank everyone in Secure Computing Lab.

I also thank all members of "Shin-Akarui-Angou-Benkyou-Kai" for the colorful discussion. In particular, I am highly thankful to Atsushi Takayasu and Satsuya Ohata for amusing conversations about the state-of-the-art research, daily life, and Japanese idol groups such as ℃-ute and Nogizaka 46.

Music is indispensable to me not only during my private life but also research life. I have been (and will be) refreshed and provided the acceleration of my research by listening my favorite music and playing the drums with my (ex-)bandmates. I have been encouraged by many artists including (but not exclusively): +44, Adele, Chat Monchy, Clammbon, Domannaka Zun, the Feeling, Fishmans, Fleet Foxes, Galileo Galilei, Gen Hoshino, Gingnang Boyz, Hanuman, the Hives, Indigo Jam Unit, Johnny Foreigner, Kings of Leon, the La's, the Libertines, Local Natives, Mando Diao, Motohiro Hata, Muse, Mute Math, Mono, the Naked & Famous, October Fall, OK Go, Paramore, the Reign of Kindo, Rooster, Sakerock, Sigur Rós, Special Others, the Strokes, Tama, Toe, Ukadan, Vampire Weekend, the Who, Yakozen, and Yuko Ando. Especially, I would have only half of publications without healing from all groups of Hello! Project such as ℃-ute and Morning Musume and Nogizaka 46, in particular, Mai Hagiwara in ℃-ute and Mai Fukagawa in Nogizaka 46 (a.k.a. Mai-Mai, respectively). Both Mai-Mai have been (and will be) ones of my healing lights. I would also like to express my gratitude to my (ex-)bandmates of the Stick Sisters, the Fundamentals, the Voxies, Monterosa Groove, Hyy, and Tokeru for composing and performing pieces with me. Jamming with them would make me happy, and give me energy to spend my research life.

I would like to thank all my friends for their support and friendships. In particular, I am deeply grateful to Naosuke Yamaichi, who is a photographer. He has an extraordinary color sense, and so photographs of his work are highly attractive to my (perhaps many people's) sensibility. He was also my ex-roommate. I shared a room with him for about one year and a half, and he always worried about me. Thanks, Yama-chan, best wishes for your marriage!

Last but not least, I greatly appreciate my dearest Anri Mabuchi and my family including my parents, aunt, and grandmother for their unconditional support and constant encouragement. Especially, I cannot thank my parents, Hiromichi Watanabe and Kumiko Watanabe, enough. Without their understanding, I would not choose a career as a researcher.

Contents

Chapter 1

Introduction

1.1 Two Major Kinds of Security in Cryptology

The history of modern cryptography (or cryptology) starts with Shannon's seminal paper [129] in 1949. He first considered cryptography from the standpoint of information theory, and contributed greatly to research areas in modern cryptography. Later, in 1976, Diffie and Hellman [52] opened up new avenues for the public-key cryptography. These two celebrated papers form the basis for current two major security criteria, the so-called *information-theoretic security* (a.k.a. information-theoretic security) and *computational security* (a.k.a. complexity-theoretic security). Until now, many researchers have developed the theory of cryptography. Taking into account developments in modern technologies such as cloud technologies and cloud-based applications, we can say this area is still in the developing stage.

As explained above, there are two major kinds of security criteria in this area, information-theoretic security and computational security. Generally, information-theoretic security is formalized in the information-theoretic sense (e.g., by Shannon entropy) or in the probability-theoretic sense (by considering success probability of adversary's guessing). On the other hand, computational security is proved by making a reduction from a protocol to some computational assumptions such as the difficulty of the integer factoring problem and that of the discrete logarithm problem. Namely, we guarantee the security by showing that "if there exists an algorithm that can break the protocol, then we can solve the problem by using the algorithm," (and hence, by showing that "if the computational problem is hard to solve, then the protocol is secure," by contraposition). Hence, in the computational security setting, we have to assume computationally-bounded adversaries (i.e., polynomial time Turing machines), whereas we can guarantee the security of information-theoretically secure protocols even if adversaries' computational power is unlimited (i.e., infinite). This means that information-theoretic security is completely unaffected by the realization of quantum computers and progress of computa-

tional algorithms, and hence it can provide long-term security (e.g., more than decades). Thus, information-theoretic security has the advantage of superior security than computational security. However, information-theoretic security also has some drawbacks regarding the practicality and efficiency as follows. (1) From the aspect of the practicality, it is impossible to realize any public-key protocols if there is no assumption. Namely, entities have to share some secret information. Therefore, although information-theoretic security does not rely on any computational assumption, it requires some non-computational assumptions (e.g., the existence of a trusted third party, or the existence of quantum channels). (2) From the aspect of the efficiency, information-theoretically secure protocols usually require long secret keys. Therefore, it is important to show the minimal key size (i.e., to analyze relationships between security and efficiency).

Thus, there exists a trade-off between these two security criteria, and hence, it is important to investigate cryptographic protocols from both perspectives.

1.2 Background

In this thesis, we consider *cryptography with timed access control*. This "timed access control" means that an entity can specify when other entity's functionality (e.g., decryption) is allowed to be activated. "Time" is intimately related to our lives, and many industrial systems are automated according to schedule (i.e. time). We can consider such automation as timed access control, and therefore, we believe that it is useful and meaningful to consider cryptographic protocols with timed access control.

Actually, as such protocols, *timed-release cryptographic protocols* introduced in [102] are well known. Informally, the goal of timed-release cryptography is *to securely send certain information into the future*. For instance, in timed-release encryption, a sender transmits a ciphertext so that a receiver can decrypt it when the time which the sender specified has come, and the receiver cannot decrypt it before the time. The timed-release cryptography was first proposed by May [102] in 1993, and after that, Rivest et al. [121] developed it in a systematic and formal way. Since Rivest et al. gave a formal definition of timed-release encryption in [121], various research on timed-release cryptography have been done based on computational security. However, the fundamental research on computational timed-release security does not seem enough. Namely, it is still not clear whether and how timed-release functionality is added to all computationally secure cryptographic primitives. Furthermore, there is no papers which report on the study of information-theoretic timed-release security.

Again, the timed-release cryptography allows a sender to specify when receiver's functionality (e.g., decryption) is activated. On the one hand, we

can also consider another concept of cryptographically timed access control arising from the following natural question:

Can we realize cryptographic protocols which allow a sender to specify when receiver's functionality is inactivated in the middle of the protocols?

Since we can regard "inactivate" as "activate inactivation", such a protocol can be considered as a kind of cryptographic protocols with timed access control. In other words, we consider protocols such that a receiver who possesses the ability to decrypt ciphertexts will get less able to decrypt them at some point. For convenience, we call this property "timed-revocable" in this thesis. Although it seems difficult to realize such functionality in the traditional setting (i.e., the receiver has both a ciphertext and a secret key), we consider such functionality in the cloud environment setting (i.e. the receive has only a secret key, and ciphertexts are stored in the cloud). In fact, Sahai et al. [125] proposed an attribute-based encryption scheme with such functionality in the cloud environment setting. The scheme is called *revocable-storage attribute-based encryption* (RS-ABE). In RS-ABE, ciphertexts in a cloud storage system can be periodically updated according to changing users who are permitted to decrypt the ciphertext, and hence some users lose his ability to decrypt ciphertexts. However, this scheme is computationally secure and does not guarantee security against future powerful adversaries.

1.3 Overview of Contribution

In this thesis, we aim to develop cryptography with timed access control.

First, we develop the timed-release cryptography from the viewpoints of two security criteria, and ultimately complete the fundamental research on the timed-release cryptography. Specifically, we consider how fundamental cryptographic protocols can achieve timed-release security in respective security criteria, (i) computational security and (ii) information-theoretic security.

We also consider (iii) the timed-revocable cryptography in the information-theoretic security setting. Namely, we consider information-theoretically secure timed-revocable protocols which can revoke receiver's functionality in the middle of the protocols in the cloud environment setting.

The overview of the contributions is as follows. The detailed contributions will be given at the beginning of each chapter.

(i) Computational Timed-Release Cryptography (in Chapter 3). So far, timed-release encryption [51, 93] and timed-release signatures [64, 65] (or so-called time-capsule signatures [56, 89]) have been proposed. However, there exist no papers which report on computational secret sharing (CSS), which is also known as one of the fundamental computational cryptographic protocols, with timed-release functionality. Com-

pared to information-theoretically secure secret sharing schemes, the advantage of CSS schemes is that the secret-information size can be significantly reduced. Therefore, we propose a *timed-release computational secret sharing* (TR-CSS) scheme, which is a CSS scheme with timed-release functionality, from the aspect of efficiency. Specifically, we formalize a model and security notions of TR-CSS, and present two kinds of constructions: One is a generic construction, and another is not generic but more efficient one than the former. Our TR-CSS scheme finally achieves the almost same secret-information size as Krawczyk's CSS scheme [86], which is the well-known construction of CSS schemes, when some predefined parameter (to be precise, a threshold value) is sufficiently large. We also show that not only this proposal is theoretically-interesting, but also we can consider many applications of our TR-CSS scheme. Specifically, we show that our TR-CSS scheme can more efficiently provide multiple encryption and threshold encryption with timed-release functionality than timed-release public-key encryption (TR-PKE), which is the major timed-release primitive.

(ii) Information-Theoretic Timed-Release Cryptography (in Chapter 4). As fundamental cryptographic primitives with information-theoretic security, we can consider information-theoretically secure key-agreement, encryption, authentication codes, and secret sharing. Therefore, we propose all those primitives with timed-release security. If a sender wants to transmit a message far into the future, information-theoretic security will be helpful in constructing timed-release mechanism, since its security can provide the long-term security. Specifically, we propose the following four fundamental timed-release protocols.

1. Timed-release key-agreement (TR-KA), where any two users in a user set can share a common key at certain time specified by one user.
2. Timed-release encryption (TRE), where a receiver cannot decrypt a ciphertext until certain time specified by a sender comes.
3. Timed-release authentication codes (TRA-codes), where a receiver cannot check the validity of a message until certain time specified by a sender comes.
4. Timed-release secret sharing (TR-SS), where participants cannot recover the secret information until certain time specified by a sender (called a dealer in the secret sharing context) comes.

Specifically, we give a model and security definition of each scheme, derive (tight) lower bounds on sizes of secret information required for each scheme, and propose the most efficient construction of each scheme in the sense of secret-information sizes (i.e., the construction attains the

lower bounds). As explained earlier, information-theoretically secure protocols generally have the drawback of long secret keys. Therefore, it is important to show the minimal key size by deriving tight bounds on the key sizes.

(iii) Information-Theoretic Timed-Revocable Cryptography (in Chapter 5). We consider a timed-revocable cryptographic protocol in the cloud environment setting. Considering affinity for cloud storage, we focus on *broadcast encryption* (BE) schemes, which allow a sender to choose arbitrary receivers who are eligible to decrypt a ciphertext (such receivers are called *privileged users*) when generating the ciphertext. In the BE context, the timed-revocable property means that a sender (or a third party) can arbitrarily revoke the decryption ability of some privileged users. However, the privileged users cannot be dynamically changed without decrypting and re-encrypting the ciphertext in traditional BE schemes. Namely, it means that BE schemes does not have the timed-revocable property, though the property must be desired in the cloud environment since there are potentially many users and their access privileges for stored data is subject to change. By assuming that ciphertexts are stored in cloud storage, we propose a new concept of BE, which we call information-theoretically secure *revocable-storage broadcast encryption* (RS-BE). In RS-BE, a sender can choose arbitrary privileged users and encrypt a plaintext as in the traditional BE schemes. Moreover, the privileged users can be dynamically changed without decrypting the corresponding ciphertext. Since the ability to decrypt ciphertext stored in the storage can be dynamically revoked, this BE scheme is named "revocable-storage." As mentioned earlier, Sahai et al. [125] proposed an RS-ABE scheme, which has the same concept as our RS-BE scheme. However, their scheme is only computationally secure, and we note that there are no papers which report how timed-revocable protocols can be realized in the information-theoretic security setting.

It is known that BE schemes have trade-offs between the secret-key sizes and ciphertext sizes. Our RS-BE scheme only captures the case of the smallest ciphertext size since many ciphertexts are stored in the storage for a long time (or permanently). However, RS-BE schemes with more general ciphertext sizes might be suited to some (cloud-based) applications. Therefore, as a basic step toward RS-BE schemes with general ciphertext sizes, we consider how traditional BE schemes with any ciphertext sizes can be realized, and thus, we propose a generic construction of the BE scheme when its ciphertext size is equal to integer multiple of the plaintext size. Actually, deriving a tight bound on the secret-key size required for traditional BE schemes with general ciphertext sizes is an open problem. This result is also the first step to solve the above open problem.

Chapter 2

Preliminaries

In Section 2.1, we prepare the notation used throughout of this thesis. In Section 2.2, we describe several information-theoretic tools such as Shannon entropy. In Sections 2.4, 2.5, and 2.6, we give definitions of key agreement, encryption, and authentication schemes, respectively. In Section 2.7, we also give the definition of secret sharing schemes, which are one of the fundamental cryptographic protocols.

2.1 Notation

In this thesis, we use the following notations. If we write $(y_1, y_2, \ldots, y_m) \leftarrow A(x_1, x_2, \ldots, x_n)$ for an algorithm A having n inputs and m outputs, it means to input $x_1, x_2, \ldots, x_n$ into A and to get the resulting output $y_1, y_2, \ldots, y_m$. If $\mathcal{X}$ is a set, we write $x \xleftarrow{\$} \mathcal{X}$ to mean the operation of picking an element x of $\mathcal{X}$ uniformly at random, and $|\mathcal{X}|$ denotes its cardinality. If x is a string, then $|x|$ denotes its bit-length. For any finite set $\mathcal{Z}$ and arbitrary non-negative integers z_1, z_2, let $\mathcal{PS}(\mathcal{Z}, z_1, z_2) := \{Z \subset \mathcal{Z} \mid z_1 \leq |Z| \leq z_2\}$ be the family of all subsets of $\mathcal{Z}$ whose cardinality is at least z_1 but no more than z_2. In particular, if $z_1 = 0$, then we simply write $\mathcal{PS}(\mathcal{Z}, z_2) := \mathcal{PS}(\mathcal{Z}, 0, z_2)$. For some positive integer n, let $[n] := \{1, 2, \ldots, n\}$. All the base of logarithm is assumed to be 2. Note that $0 \log \frac{b}{0} = 0$ $(b \geq 0)$, and $a \log \frac{a}{0} = +\infty$ $(a > 0)$. We use κ as a security parameter. When we write negligible ϵ in κ, it means a function $\epsilon : \mathbb{N} \to [0, 1]$ where $\epsilon(\kappa) < 1/g(\kappa)$ for any polynomial g and sufficiently large κ. Furthermore, in this paper "probabilistic polynomial-time" is abbreviated as "PPT."

2.2 Information-Theoretic Tools

We describe several information-theoretic results. For details, see [41] for the excellent instruction.

Definition 2.1 (Shannon Entropy [128]). *The entropy $H(X)$ of a random variable X is defined by*

$$H(X) := -\sum_{x \in \mathcal{X}} \Pr[X = x] \log \Pr[X = x].$$

The following fact is well known.

Proposition 2.1. *For a random variable X, it holds that*

$$\log |\mathcal{X}| \geq H(X) \geq 0,$$

where the left-side equality holds if and only if a probability distribution of $\mathcal{X}$ is uniform, and the right-side equality holds if and only if there exists some $x \in \mathcal{X}$ such that $\Pr[X = x] = 1$.

Definition 2.2 (Joint Entropy). *The joint entropy $H(X, Y)$ of a pair of random variables (X, Y) with a joint probability distribution P_{XY} is defined by*

$$H(X, Y) := -\sum_{x \in \mathcal{X}} \sum_{y \in \mathcal{Y}} \Pr[X = x, Y = y] \log \Pr[X = x, Y = y].$$

Definition 2.3 (Conditional Entropy). *The conditional entropy $H(X \mid Y)$ of a pair of random variables (X, Y) with a joint probability distribution P_{XY} is defined by*

$$H(X \mid Y) := \sum_{y \in \mathcal{Y}} \Pr[Y = y] H(X \mid Y = y).$$

Proposition 2.2 (Chain Rule). *It holds that $H(X, Y) = H(X) + H(Y \mid X) = H(Y) + H(X \mid Y)$. More generally, it holds that $H(X_1, X_2, \ldots, X_n) = \sum_{i=1}^{n} H(X_i \mid X_1, \ldots, X_{i-1})$.*

Proposition 2.3 (Conditioning Reduces Entropy). *For any two random variables X and Y, it hold that $H(X) \geq H(X \mid Y)$, where equality holds if and only if X and Y are independent.*

Definition 2.4 (Mutual Information). *For two random variables X and Y with a joint probability distribution P_{XY} and marginal probability distributions $P_{X|Y}$ and $P_{Y|X}$, the mutual information $I(X; Y)$ is defined by*

$$I(X; Y) := \sum_{(x,y) \in \mathcal{X} \times \mathcal{Y}} \Pr[X = x, Y = y] \log \frac{\Pr[X = x, Y = y]}{\Pr[X = x] \Pr[Y = y]}.$$

Proposition 2.4 (Mutual Information and Entropy). *For any two random variables X and Y, it hold that*

$$I(X; Y) = H(X) - H(X \mid Y),$$

$$
\begin{aligned}
I(X;Y) =& H(Y) - H(Y \mid X),\\
I(X;Y) =& H(X) + H(Y) - H(X,Y),\\
I(X;Y) =& I(Y;X),\\
I(X;X) =& H(X).
\end{aligned}
$$

Corollary 2.1 (Nonnegativity of Mutual Information). *For any two random variables X and Y, it hold that $I(X;Y) \geq 0$, where equality holds if and only if X and Y are independent.*

2.3 Premises and Assumptions

We clarify what premises and assumptions we consider in this thesis. Specifically, we describe what kinds of channels we suppose, what kinds of physical and computational assumptions there are, and why we consider information-theoretically secure protocols in the one-time model.

2.3.1 Channels

Based on [100, 101], we can classify channel types into four types in terms of *confidentiality* and *authenticity*.

1. A *secure channel* is a channel that provides both confidentiality and authenticity. Namely, the secure channel leaks no information on the transmitted message, and only allows adversaries to forward the message. We assume that this channel is used when distributing secret keys by a trusted authority or a sender.

2. A *confidential channel* is a channel that provides confidentiality. Namely, the confidential channel leaks no information on the transmitted message, however allows the adversary to change (i.e., modify, insert, etc.) the message before a receiver receives it. We do not consider this channel in this thesis.

3. An *authenticated channel* is a channel that provides authenticity. Namely, the authenticated channel leaks the transmitted message, however only allows adversaries to forward the message (i.e., the adversary cannot change the message). We assume that this channel is used in all protocols except for authentication/signature ones (i.e., except for protocols that aim to provide authenticity).

4. An *insecure channel* is a channel that provides neither confidentiality nor authenticity. Namely, the insecure channel leaks the transmitted message, and allows the adversary to change (i.e., modify, insert, etc.) the message before a receiver receives it. We assume that this channel is used in authentication/signature protocols.

We consider that adversaries can do any permitted attacks in each channel, however, for simplicity we assume that the behaviors of adversaries are restricted as follows: Adversaries do not abort the channel, and do not perform denial-of-service (DoS) attacks. In addition, adversaries also never try to delay and delete transmitted messages in insecure (and confidential) channels.

Further, we also consider a *broadcast channel*, which is a channel used when a single entity sends information to multiple entities. We assume that all broadcast channels that appear in this thesis are *authenticated*, and broadcast information are received by all entities *simultaneously*. If we write "a sender broadcasts a message to receivers," it means that "a sender sends a message to receivers through authenticated broadcast channel."

2.3.2 Physical Assumptions vs. Computational Assumptions

In information-theoretically secure protocols, we often use physical assumptions, though the protocols require no computational assumptions. Major physical assumptions are as follows (for details, see [130], which is a comprehensive survey paper).

Trusted initializer (TI) model. We assume that there exists a trusted authority, which is sometimes called *a trusted initializer* (TI), whose role is to generate secret keys behalf of all entities, and to distribute secret keys to corresponding entities through secure channels. A model in which a sender performs as TI is also considered as the TI model. Although the TI model had been used in various researches implicitly, it was explicitly introduced by Rivest [120]. *All information-theoretically secure protocols that appear in this thesis are considered in this model.*

Bounded storage model (BSM). We assume there exists an information source (e.g., a satellite) whose role is to generate and broadcast a long random string, and an adversary has a limited storage capacity such that he cannot fully store the random string. However, the adversary has an unbounded computational power and is allowed to have an unlimited storage *after the random string is broadcasted.* We also assume that legitimate entities have limited storage capacity. This model was introduced by Maurer [98].

Noisy channel model (NCM). We assume that entities transmit their information over channels with noise, though we usually assume to use error-free channels in modern cryptography. If a channel between a sender and a receiver has less noise than a channel between the sender and an adversary, the receiver can correctly get the transmitted information while the adversary cannot get the information correctly. This model was introduced by Wyner [147], and later Maurer [99] improved Wyner's result. Specifically, he showed that by allowing the receiver to

send feedback to the sender through a noiseless channel, they can succeed in key agreement securely even if the channel between them has much noise than the channel between the sender and the adversary.

Quantum channel model (QCM). We assume ideal quantum channels, and consider security based on quantum information theory. The most famous (quantum key distribution) protocol is so-called BB84 protocol [12], which was proposed by Bennet and Brassard.

Manual channel model (MCM). We assume that there exist interactive insecure channels and a one-way, narrow-band, weakly-authenticated channel, which is called *a manual channel*. Users interacts with each other through insecure channels, and finally a sender sends some significantly small message through the *manual channel*. In the manual channel, the adversary cannot modify the small message, however he can read it, delay it, and remove it. Intuitively, the manual channel is an intermediate type between the authenticated channel and insecure channel. This model was introduced by Naor et al. [106, 107].

In addition, there are other physical assumptions such as *a bounded quantum storage model* [47] and *a hybrid BSM* [59, 74]. Again, all information-theoretically secure protocols that appear in this thesis are considered in the TI models.

On the other hand, computationally secure protocols require no physical assumptions but some computational assumptions. There are many computational problems such as the integer factoring, discrete logarithm, and Diffie–Hellman problem [52]. We describe a computational assumption used in this thesis. Specifically, we define a bilinear group and the decisional Diffie–Hellman problem over the bilinear group as follows.

Bilinear Group. A bilinear group generator $\mathcal{G}$ is an algorithm that takes a security parameter κ as input and outputs a bilinear group $(p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, g_1, g_2, e)$, where p is a prime, $\mathbb{G}_1$, $\mathbb{G}_2$, and $\mathbb{G}_T$ are multiplicative cyclic groups of order p, g_1 and g_2 are (random) generators of $\mathbb{G}_1$ and $\mathbb{G}_2$, respectively, and e is an efficiently computable and non-degenerate bilinear map $e : \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T$ with the following bilinear property: For any $u, u' \in \mathbb{G}_1$ and $v, v' \in \mathbb{G}_2$, $e(uu', v) = e(u, v)e(u', v)$ and $e(u, vv') = e(u, v)e(u, v')$, and for any $u \in \mathbb{G}_1$ and $v \in \mathbb{G}_2$ and any $a \in \mathbb{Z}_p$, $e(u^a, v) = e(u, v^a) = e(u, v)^a$.

A bilinear map e is called symmetric or a "Type-1" pairing if $\mathbb{G}_1 = \mathbb{G}_2$. Otherwise, it is called asymmetric. In the asymmetric setting, e is called a "Type-2" pairing if there is an efficiently computable isomorphism either from $\mathbb{G}_1$ to $\mathbb{G}_2$ or from $\mathbb{G}_2$ to $\mathbb{G}_1$. If no efficiently computable isomorphisms are known, then it is called a "Type-3" pairing. In this thesis, we assume the Type-1 pairing (i.e., $\mathbb{G} := \mathbb{G}_1 = \mathbb{G}_2$).

Decisional Bilinear Diffie–Hellman (DBDH) Assumption. Let $\mathcal{A}$ be a PPT adversary and we consider $\mathcal{A}$'s advantage against the DBDH problem as follows.

$$Adv^{\mathsf{DBDH}}_{\mathcal{G},\mathcal{A}}(\kappa) := \left| \Pr \left[b' = b \;\middle|\; \begin{array}{l} (p, \mathbb{G}, \mathbb{G}_T, g, e) \leftarrow \mathcal{G}(1^\kappa), \\ c_1, c_2, c_3 \xleftarrow{\$} \mathbb{Z}_p, \\ b \xleftarrow{\$} \{0,1\}, \\ \text{if } b = 1 \text{ then } W := \hat{e}(g,g)^{c_1 c_2 c_3}, \\ \text{else } W \xleftarrow{\$} \mathbb{G}_T, \\ b' \leftarrow \mathcal{A}(\kappa, g, g^{c_1}, g^{c_2}, g^{c_3}, W) \end{array} \right] - \frac{1}{2} \right|.$$

Definition 2.5. *For* $^{\exists}\kappa_0 \in \mathbb{N}$ *and* $^{\forall}\kappa \geq \kappa_0$*, the DBDH assumption relative to a generator* $\mathcal{G}$ *holds if there exists a negligible* ϵ *in* κ *such that* $Adv^{DBDH}_{\mathcal{G},\mathcal{A}}(\kappa) < \epsilon$ *holds for any PPT adversary* $\mathcal{A}$*.*

2.3.3 One-time Model vs. Multiple-time Model

In this thesis, all of information-theoretically secure protocols are considered in one-time models, where a sender runs his algorithm (e.g., an encryption algorithm) only once. The reason why we deal with the one-time model is to simplify the analysis. Generally, formalization of models and often become complicated in multiple-time information-theoretically secure cryptographic protocols. Actually, several recent works such as oblivious polynomial evaluation [138], key distribution [123], and authentication codes [139, 107] dealt with one-time protocols. In information-theoretic cryptography, researchers usually start to consider a protocol in the one-time model since it makes the analysis simple, and then the protocol is extended to the multiple-time model. Therefore, in this thesis we consider protocols in the one-time model. We believe our results will be bases for proposals of the protocols in the multiple-time setting.

Note that we consider all computationally secure protocols in multiple-time models.

2.4 Key Agreement

We here describe an information-theoretically secure non-interactive key agreement protocol with initial secret keys, which is called a *key predistribution system* (KPS). At the beginning of the KPS, a trusted authority TA generates secret keys $uk_1, \ldots, uk_n$ of n users $\mathcal{U} := \{U_1, \ldots, U_n\}$, and distributes them to the corresponding users via secure channels. Then, each user U_i can choose any subset $\mathcal{S} \in \mathcal{U}$ such that $U_i \in \mathcal{S}$ and generate a common key $k_{\mathcal{S}}$ for $\mathcal{S}$ without any interaction.

Formally, the definition of KPS is given as follows. For any subset $\mathcal{J} := \{U_{i_1}, \ldots, U_{i_j}\} \subset \mathcal{U}$, let $\mathcal{K}_{\mathcal{J}}$ be a set of all possible session keys for the privileged

set $\mathcal{J}$, and let $\mathcal{K} := \bigcup_{\mathcal{J}\subset\mathcal{U}} \mathcal{K}_{\mathcal{J}}$. Let $\mathcal{UK}_i$ be a set of possible secret keys for U_i, and $\mathcal{UK} := \bigcup_{i=1}^{n} \mathcal{UK}_i$. A KPS Π_{KPS} consists of the following two-tuple of algorithms (Init, Der) with finite two spaces, $\mathcal{K}$ and $\mathcal{UK}$.

- $(uk_1, \ldots, uk_n) \leftarrow \mathsf{Init}(n)$: A probabilistic algorithm for initial key generation. It takes the number of users n as input, and outputs n secret keys $(uk_1, \ldots, uk_n) \in \prod_{i=1}^{n} \mathcal{UK}_i$. Here, we also define uk, which is called *a master key*, as a randomness required for determining $(uk_1, \ldots, uk_n)$.[1]

- $k_{\mathcal{S}} \leftarrow \mathsf{Der}(uk_i, \mathcal{S})$: A deterministic algorithm for key derivation. It takes a secret key uk_i of a user U_i, and a privileged set $\mathcal{S} \subset \mathcal{U}$ as input, and outputs a session key $k_{\mathcal{S}}$ for $\mathcal{S}$.

In the above model, there is the following correctness requirement: For all $n \in \mathbb{N}$, all $(uk_1, \ldots, uk_n) \leftarrow \mathsf{Init}(n)$, and all $\mathcal{S} := \{U_{i_1}, \ldots, U_{i_j}\} \subset \mathcal{U}$, $\mathsf{Der}(uk_{i_1}, \mathcal{S}) = \cdots = \mathsf{Der}(uk_{i_j}, \mathcal{S})$, or equivalently it holds $H(K_{\mathcal{S}} \mid UK_i) = 0$ for any $U_i \in \mathcal{S}$.

We consider perfect secrecy, which means that an adversary cannot get any information on the session key, against at most ω colluders. Namely, a set $\mathcal{W}$ of at most ω colluders cannot obtain any information on a session key for any set $\mathcal{S}$ such that $\mathcal{S} \cap \mathcal{W} = \emptyset$ from their secret keys. For any $\mathcal{J} := \{U_{i_1}, \ldots, U_{i_j}\} \subset \mathcal{U}$, let $\mathcal{UK}_{\mathcal{J}} := \mathcal{UK}_{i_1} \times \cdots \times \mathcal{UK}_{i_j}$ be a set of possible secret keys of $\mathcal{J}$. $K_{\mathcal{J}}$, UK, UK_i, and $UK_{\mathcal{J}}$ denote random variables which take values in $\mathcal{K}_{\mathcal{J}}$, $\mathcal{UK}$, $\mathcal{UK}_i$, and $\mathcal{UK}_{\mathcal{J}}$, respectively. Formally, security of a KPS is defined as follows.

Definition 2.6 (Security of KPS). *Let Π_{KPS} be a KPS. Π_{KPS} is said to be an $(\leq n, \leq \omega)$-KPS if the following conditions are satisfied: For any privileged set $\mathcal{S} \subset \mathcal{U}$, and any set of colluders $\mathcal{W} \subset \mathcal{U}$ such that $\mathcal{S} \cap \mathcal{W} = \emptyset$ and $|\mathcal{W}| \leq \omega$, it holds that*

$$H(K_{\mathcal{S}} \mid UK_{\mathcal{W}}) = H(K_{\mathcal{S}}).$$

We describe some known results on $(\leq n, \leq \omega)$-KPSs. First, we describe tight lower bounds on the secret-key size required for $(\leq n, \leq \omega)$-KPSs. In [19], these bound were first derived in the context of *zero-message BESs*, which are the same as KPSs. In the following proposition, for all $\mathcal{S}, \mathcal{S}' \subset \mathcal{U}$ it is assumed that $H(K_{\mathcal{S}}) = H(K_{\mathcal{S}'})$ for simplicity. This common entropy is denoted by $H(K)$.

[1] Although uk is not explicitly described in several papers on KPSs [18, 22, 23, 87, 97], we introduce uk for measuring actual sizes of secret keys which TA has to generate. It is reasonable to consider uk since Blundo and Cresti [19] also dealt with uk in another context. We note that uk is actually not used in the scheme (however we use it in our construction in Section 5.5.1), and hence we do not explicitly describe it in output. We can also see uk as a deterministic function for deriving secret keys $(uk_1, \ldots, uk_n)$.

Proposition 2.5 ([19]). *Let Π_{KPS} be an $(\leq n, \leq \omega)$-KPS. Then, the following lower bounds hold:*

$$(i)\ H(UK) \geq \sum_{j=0}^{\omega} \binom{n}{j} H(K),$$

$$(ii)\ H(UK_i) \geq \sum_{j=0}^{\omega} \binom{n-1}{j} H(K) \text{ for any } i \in [n].$$

Next, we describe an optimal construction of an $(\leq n, \leq \omega)$-KPS. This *optimal* means the construction attains the lower bounds in Proposition 2.5 with equalities. Before that, we define the following families of sets:

$$\begin{aligned}
&\mathscr{W} := \{\mathcal{W} \subset \mathcal{U} \mid |\mathcal{W}| \leq \omega\},\\
&\mathscr{W}^{(i)} := \{\mathcal{W} \subset \mathcal{U} \setminus \{U_i\} \mid |\mathcal{W}| \leq \omega\},\\
&\mathscr{W}(\mathcal{S}) := \{\mathcal{W} \in \mathscr{W} \mid \mathcal{W} \cap \mathcal{S} = \emptyset \wedge |\mathcal{W}| = \min(\omega, n - |\mathcal{S}|)\}.
\end{aligned}$$

The optimal construction of an $(\leq n, \leq \omega)$-KPS is as follows. This construction, which we call the Fiat–Naor KPS, can be easily obtained from an $(\leq n, \leq \omega)$-one-time secure BES proposed by Fiat and Naor [61]. We show the somewhat fine-tuned Fiat–Naor KPS as follows, since a session key is created by redundant operation in the original Fiat–Naor KPS, though the sizes of secret keys in the original scheme are optimal.

1. $(uk_1, \ldots, uk_n) \leftarrow \mathsf{Init}(n)$: Let q be a prime power such that $q > n$, and $\mathbb{F}_q$ be a finite field with q elements. For every $\mathcal{W} \in \mathscr{W}$, it chooses $r_{\mathcal{W}} \in \mathbb{F}_q$ uniformly at random. Then, it outputs $uk_i := \{r_{\mathcal{W}} \mid \mathcal{W} \in \mathscr{W}^{(i)}\}$ $(1 \leq i \leq n)$. Also, $uk := \{r_{\mathcal{W}} \mid \mathcal{W} \in \mathscr{W}\}$.

2. $k_{\mathcal{S}} \leftarrow \mathsf{Der}(uk_i, \mathcal{S})$: For any subset $\mathcal{S}$, it computes and outputs a session key $k_{\mathcal{S}} := \sum_{\mathcal{W} \in \mathscr{W}(\mathcal{S})} r_{\mathcal{W}}$,

Proposition 2.6. *Π_{KPS} given by the above construction is an $(\leq n, \leq \omega)$-KPS and optimal.*

2.5 Encryption

Encryption schemes (or cryptosystems) achieve *confidentiality* (or *privacy*). Namely, the aim of encryption schemes is to keep information secret from adversaries. We can classify encryption schemes into two types, *secret key encryption* and *public key encryption* schemes. Literally, the former requires that users must share and keep some secret information, while the latter can disclose some information (e.g., keys). As explained in Section 1.1, the public key mechanism is useful from the practical point of view, however it cannot be realized in the information-theoretic security setting.

2.5.1 Secret Key (Symmetric Key) Encryption

Secret key encryption (SKE), which is also known as symmetric key encryption, is defined as follows.[2] $\mathcal{M}_{\mathrm{SKE}}$ is a set of plaintexts, $\mathcal{K}$ is a set of secret keys, and $\mathcal{C}$ is a set of ciphertexts. a SKE scheme Π_{SKE} consists of three-tuple algorithms $(\mathsf{G}, \mathsf{E}, \mathsf{D})$ defined as follows:

- $k \leftarrow \mathsf{G}(1^{\kappa})$: A probabilistic algorithm for key generation. It takes a security parameter κ as input and outputs a secret key $k \in \mathcal{K}$.
- $c \leftarrow \mathsf{E}(k, m)$: An algorithm for encryption. It takes a secret key k and a plaintext $m \in \mathcal{M}_{\mathrm{SKE}}$ as input and then outputs a ciphertext $c \in \mathcal{C}$.
- m or $\perp \leftarrow \mathsf{D}(k, c)$: A deterministic algorithm for decryption. It takes a secret key k and a ciphertext c as inputs and then outputs a plaintext $m \in \mathcal{M}_{\mathrm{SKE}}$ or $\perp \notin \mathcal{M}_{\mathrm{SKE}}$.

In the above model, we assume that Π_{SKE} meets the following *correctness* property: For all $\kappa \in \mathbb{N}$, all $k \in \mathcal{K}$, and all $m \in \mathcal{M}_{\mathrm{SKE}}$, it holds that $m \leftarrow \mathsf{D}(k, \mathsf{E}(k, m))$.

We here give security definition of SKE schemes in the information-theoretic security sense. Note that we here assume a *one-time model*, where a sender encrypts and sends a plaintexts only once. We define the formalization of the security notion of this scheme written with Shannon entropy as follows. M, K, and C denote random variables which take values in $\mathcal{M}_{\mathrm{SKE}}$, $\mathcal{K}$, and $\mathcal{C}$, respectively.

Definition 2.7. *(Perfect Secrecy [129]) An SKE scheme Π_{SKE} is said to be perfectly secure if it holds that*

$$H(M \mid C) = H(M).$$

Intuitively, this formalization says that any computationally-unbounded adversary can obtain no information on the underlying plaintext from the ciphertext which the adversary can observe.

In the information-theoretic cryptography, we can derive a lower bound on the size of secret information (i.e., a lower bound on entities' memory-sizes for information-theoretically secure cryptographic schemes). Shannon [129] derived a lower bound on the secret-key sizes required for perfectly secure SKE schemes as follows.

Proposition 2.7 (Shannon's bound [129])**.** *Let Π_{SKE} be a perfectly secure SKE scheme. Then, we have*

$$H(K) \geq H(M).$$

[2] In some contexts, a SKE scheme is also called *a data encapsulation mechanism (DEM)*. In this thesis, we sometimes use the term "a DEM" instead of a SKE scheme.

In particular, if a probability distribution of $\mathcal{M}_{\mathrm{SKE}}$ *is uniform, then we have*

$$|\mathcal{K}| \geq |\mathcal{M}_{\mathrm{SKE}}|.$$

Proof. We have

$$\begin{aligned}
\log|\mathcal{K}| \geq H(K) \geq H(K \mid C) \geq I(K;M \mid C) &= H(M \mid C) - H(M \mid C,K) \\
&= H(M \mid C) && (2.1)\\
&= H(M) && (2.2)\\
&= \log|\mathcal{M}_{\mathrm{SKE}}|, && (2.3)
\end{aligned}$$

where Eq. (2.1) follows from that the D algorithm is deterministic, Eq. (2.2) follows from perfect secrecy (Definition 2.7), and Eq. (2.3) follows from the above assumption, $\log|\mathcal{M}_{\mathrm{SKE}}| = H(M)$. □

The one-time pad (or the vernam cipher) [140], which is the most famous construction of information-theoretically secure SKE schemes, meets the lower bound in Proposition 2.7 with equality. In other words, the one-time pad is optimal in the sense of the most efficient construction in terms of secret-key sizes.

We also define the perfect secrecy in the computational security sense. Namely, we target only *computationally-bounded* adversaries (i.e. polynomial time algorithms). Instead of such a security restriction, computationally secure SKE schemes can significantly reduce its secret-key size and reuse the secret key for encryption (i.e., remove the restriction on the one-time model). Specifically, in this paper we consider the notion of find-then-guess indistinguishability against chosen plaintext attack (FTG-CPA). Let $\mathcal{A}$ be a PPT adversary, and $\mathcal{A}$'s advantage against the FTG-CPA security is defined by

$$Adv^{\mathsf{FTG\text{-}CPA}}_{\Pi_{\mathrm{SKE}},\mathcal{A}}(\kappa) := \left| \Pr\left[b' = b \;\middle|\; \begin{array}{l} b \xleftarrow{\$} \{0,1\},\ k \leftarrow \mathsf{G}(1^\kappa), \\ (m_0^*, m_1^*, st) \leftarrow \mathcal{A}^{E(\cdot)}(\mathsf{find}, \kappa), \\ c^* \leftarrow \mathsf{E}(k, m_b^*), \\ b' \leftarrow \mathcal{A}^{E(\cdot)}(\mathsf{guess}, c^*, st) \end{array} \right] - \frac{1}{2} \right|.$$

Here, we require $|m_0^*| = |m_1^*|$, and st is state information. In addition, $E(\cdot)$ is an *encryption oracle* which takes a plaintext m as input, and then returns $\mathsf{E}(k,m)$. $\mathcal{A}$ is allowed to issue arbitrary queries to the above oracle q_e times, where q_e is polynomial in κ.

Definition 2.8 (FTG-CPA [7]). *For* $^\exists\kappa_0 \in \mathbb{N}$ *and* $^\forall\kappa \geq \kappa_0$*, an SKE (or, DEM*[3]*)* Π_{SKE} *is said to be* (q_e, ϵ)*-FTG-CPA secure if there exists a negligible* ϵ *in* κ *such that* $Adv^{\mathsf{FTG\text{-}CPA}}_{\Pi_{\mathrm{SKE}},\mathcal{A}}(\kappa) < \epsilon$ *holds for any PPT adversary* $\mathcal{A}$*, where* q_e

[3] A computationally-secure SKE scheme is often called a *data encapsulation mechanism* (DEM) in the context of hybrid encryption [45].

is the number of queries that $\mathcal{A}$ can issue to the oracle in the FTG-CPA game. In particular, if $q_e = 0$ (i.e. in the case that $\mathcal{A}$ never queries to the encryption oracle), then we just write ϵ-FTG-CPA.

As mentioned in [7], FTG-CPA is not a strong security notion. This is because several symmetric encryption schemes based on finite pseudorandom functions or permutations (e.g., AES in practice) meet a notion of left-or-right indistinguishability against chosen plaintext attack (LOR-CPA), which is a stronger notion than FTG-CPA (for details, see [3, 7, 136]).

2.5.2 Public Key Encryption

In this thesis, we consider public key encryption (PKE) with labels as in [131]. Let $\mathcal{M}_{\mathrm{PKE}}$ be a set of plaintexts determined by a security parameter κ. A PKE scheme Π_{PKE} consists of three-tuple algorithms (Gen, Enc, Dec) defined as follows.

- $(pk, sk) \leftarrow \mathsf{Gen}(1^\kappa)$: A probabilistic algorithm for key generation. It takes a security parameter κ as input and outputs a pair of a public key and a secret key (pk, sk).
- $c \leftarrow \mathsf{Enc}_L(pk, m)$: An algorithm for encryption. It takes the public key pk, a label L, and a plaintext $m \in \mathcal{M}_{\mathrm{PKE}}$ as input, and outputs a ciphertext c.
- m or $\perp \leftarrow \mathsf{Dec}_L(sk, c)$: A deterministic algorithm for decryption. It takes the secret key sk, a label L, and a ciphertext C, and outputs a plaintext $m \in \mathcal{M}_{\mathrm{PKE}}$ or $\perp \notin \mathcal{M}_{\mathrm{PKE}}$.

We assume that Π_{PKE} meets the following *correctness* property: For all $\kappa \in \mathbb{N}$, all labels L, all $(pk, sk) \leftarrow \mathsf{Gen}(1^\kappa)$, and all $m \in \mathcal{M}_{\mathrm{PKE}}$, it holds that $m \leftarrow \mathsf{Dec}_L(sk, \mathsf{Enc}_L(pk, m))$.

We describe the notion of indistinguishability against chosen plaintext attack (IND-CPA). This security notion (a.k.a. *semantic security*) was introduced by Goldwasser and Micali [70]. Let $\mathcal{A}$ be a PPT adversary, and $\mathcal{A}$'s advantage against the IND-CPA security is defined by

$$Adv^{\mathsf{IND\text{-}CPA}}_{\Pi_{\mathrm{PKE}},\mathcal{A}}(\kappa) := \left| \Pr\left[b' = b \;\middle|\; \begin{array}{l} (pk, sk) \leftarrow \mathsf{Gen}(1^\kappa), \\ (m_0^*, m_1^*, L^*, st) \leftarrow \mathcal{A}(\mathsf{chal}, pk), \\ b \xleftarrow{\$} \{0,1\}; c^* \leftarrow \mathsf{Enc}_{L^*}(pk, m_b^*), \\ b' \leftarrow \mathcal{A}(\mathsf{guess}, c^*, st) \end{array} \right] - \frac{1}{2} \right|.$$

Here, we require $|m_0^*| = |m_1^*|$, and st is state information.

Definition 2.9 (IND-CPA). *For $^\exists\kappa_0 \in \mathbb{N}$ and $^\forall\kappa \geq \kappa_0$, a PKE scheme Π_{PKE} is said to be ϵ-IND-CPA secure if there exists a negligible ϵ in κ such that $Adv^{\mathsf{IND\text{-}CPA}}_{\Pi_{\mathrm{PKE}},\mathcal{A}}(\kappa) < \epsilon$ holds for any PPT adversary $\mathcal{A}$.*

Moreover, by taking a *decryption oracle* into account we can also consider the notion of indistinguishability against chosen ciphertext attack (IND-CCA). This security notion was first introduced by Rackoff ad Simon [119]. Actually, it is important to consider IND-CCA secure PKE schemes since Bleichenbacher [17] showed practical chosen ciphertext attack against protocols following the encryption standard PKCS #1. Let $\mathcal{A}$ be a PPT adversary, and $\mathcal{A}$'s advantage against the IND-CCA security is defined by

$$Adv^{\mathsf{IND\text{-}CCA}}_{\Pi_{\mathrm{PKE}},\mathcal{A}}(\kappa) := \left| \Pr \left[b' = b \;\middle|\; \begin{array}{l} (pk, sk) \leftarrow \mathsf{Gen}(1^\kappa), \\ (m_0^*, m_1^*, L^*, st) \leftarrow \mathcal{A}^{Dec(\cdot,\cdot)}(\mathsf{chal}, pk), \\ b \xleftarrow{\$} \{0,1\}; c^* \leftarrow \mathsf{Enc}_{L^*}(pk, m_b^*), \\ b' \leftarrow \mathcal{A}^{Dec(\cdot,\cdot)}(\mathsf{guess}, c^*, st) \end{array} \right] - \frac{1}{2} \right|.$$

Here, we require $|m_0^*| = |m_1^*|$, and st is state information. In addition, $Dec(\cdot,\cdot)$ is a *decryption oracle* which takes a pair of a ciphertext and a label c as input, and then returns $\mathsf{Dec}_L(sk, c)$. $\mathcal{A}$ is allowed to issue arbitrary queries to the above oracle in the chal stage, however, $\mathcal{A}$ cannot submit c^* to the oracle in the guess stage.

Definition 2.10 (IND-CCA). *For $^\exists\kappa_0 \in \mathbb{N}$ and $^\forall\kappa \geq \kappa_0$, a PKE scheme Π_{PKE} is said to be (q_c, ϵ)-IND-CCA secure if there exists a negligible ϵ in κ such that $Adv^{\mathsf{IND\text{-}CCA}}_{\Pi_{\mathrm{PKE}},\mathcal{A}}(\kappa) < \epsilon$ holds for any PPT adversary $\mathcal{A}$, where q_c is the number of queries that $\mathcal{A}$ can issue to the oracle in the IND-CCA game.*

There are several IND-CPA secure PKE schemes such as Goldwasser–Micali encryption [70], Elgamal encryption [60], and Paillier encryption [112]. Dolev et al. [57, 58] first realized the IND-CCA secure PKE scheme based on the work by Naor and Yung [108], though the scheme is impractical since it relies on expensive non-interactive zero-knowledge proofs. Later, several practical IND-CCA secure PKE schemes such as Cramer–Shoup encryption [44] and Hofheinz–Kiltz encryption [77]. By assuming ideal hash functions (called *random oracles* [8]), Bellare and Rogaway [9], Fujisaki and Okamoto [63], and Okamoto and Pointcheval [109] also proposed IND-CCA secure PKE schemes. In addition, many other constructions of IND-CPA and IND-CCA secure PKE schemes have been proposed (e.g., [26, 32, 73, 116]).

2.5.3 Identity-Based Key Encapsulation Mechanism

A key encapsulation mechanism (KEM) is a special case of PKE schemes. Specifically, the KEM is a PKE scheme in which probability distribution of $\mathcal{M}_{\mathrm{PKE}}$ is uniform. In other words, in the KEM a sender encrypts a random string such as a secret key of a SKE scheme. The KEM is mainly used for constructing efficient PKE schemes. On the other hand, identity-based encryption (IBE) is a class of PKE which allows users to use arbitrary strings as their

public keys. Recently, many applications of IBE has been proposed, and hence IBE has been one of the important fundamental cryptographic protocols.

An identity-based key encryption mechanism (IB-KEM), which is a KEM in the identity-based setting, was proposed in [13, 84, 85]. $\mathcal{ID}$ is a set of IDs and $\mathcal{K}_{\mathrm{KEM}}$ is a set of session keys determined by a security parameter κ. An IB-KEM Π_{KEM} consists of four-tuple algorithms (IB.Setup, IB.Gen, IB.Encaps, IB.Decaps) defined as follows.

- $(prm, mk) \leftarrow \mathsf{IB.Setup}(1^\kappa)$: A probabilistic algorithm for setup. It takes a security parameter κ as input, and outputs a public parameter prm and a master secret key mk.
- $sk_{ID} \leftarrow \mathsf{IB.Gen}(prm, mk, ID)$: An algorithm for key derivation. It takes the master secret key mk and an identity $ID \in \mathcal{ID}$ as input, and outputs a secret key sk_{ID} for ID.
- $(k, c_{ID}) \leftarrow \mathsf{IB.Encaps}(prm, ID)$: A probabilistic algorithm for encapsulation. It takes the public parameter prm and an identity ID as input, and outputs a pair of a session key and a corresponding ciphertext $(k \in \mathcal{K}_{\mathrm{KEM}}, c_{ID})$.
- $k \leftarrow \mathsf{IB.Decaps}(prm, sk_{ID}, c_{ID})$: A deterministic algorithm for decapsulation. It takes a secret key sk_{ID} for ID and a ciphertext c_{ID} as input, and then outputs a session key $k \in \mathcal{K}_{\mathrm{KEM}}$.

In the above model, we assume that Π_{KEM} meets the following *correctness* property: For all $\kappa \in \mathbb{N}$, all $(prm, mk) \leftarrow \mathsf{IB.Setup}(1^\kappa)$, all $ID \in \mathcal{ID}$, and all $(k, c_{ID}) \leftarrow \mathsf{IB.Encaps}(prm, ID)$, it holds $\mathsf{IB.Decaps}(prm, sk_{ID}, c_{ID}) \rightarrow k$, where $sk_{ID} \leftarrow \mathsf{IB.Gen}(mk, ID)$.

We describe the notion of indistinguishability against adaptive-identity chosen plaintext attack (IND-ID-CPA). Let $\mathcal{A}$ be a PPT adversary, and $\mathcal{A}$'s advantage against the IND-ID-CPA security is defined by

$$Adv^{\mathsf{IND\text{-}ID\text{-}CPA}}_{\Pi_{\mathrm{KEM}},\mathcal{A}}(\kappa) := \left| \Pr \left[b' = b \;\middle|\; \begin{array}{l} (prm, mk) \leftarrow \mathsf{IB.Setup}(1^\kappa), \\ (ID^*, st) \leftarrow \mathcal{A}^{Extract(\cdot)}(\mathsf{chal}, prm), \\ (k_1, c^*_{ID^*}) \leftarrow \mathsf{IB.Encaps}(prm, ID^*), \\ b \xleftarrow{\$} \{0,1\}; k_0 \xleftarrow{\$} \mathcal{K}, \\ b' \leftarrow \mathcal{A}^{Extract(\cdot)}(\mathsf{guess}, k_b, c^*_{ID^*}, st) \end{array} \right] - \frac{1}{2} \right|.$$

Here, st is state information. In addition, $Extract(\cdot)$ is a *key generation oracle* which takes an identity $ID \in \mathcal{ID}$ as input, and then returns $\mathsf{IB.Gen}(prm, mk, ID)$. $\mathcal{A}$ is allowed to issue arbitrary queries to the above oracle in the chal stage, however, $\mathcal{A}$ cannot submit ID^* to the oracle after deciding the challenge identity ID^*.

Definition 2.11 (IND-ID-CPA). *For* $^{\exists}\kappa_0 \in \mathbb{N}$ *and* $^{\forall}\kappa \geq \kappa_0$*, an IB-KEM* Π_{KEM} *is said to be* (q_{ID}, ϵ)*-IND-ID-CPA secure if there exists a negligible* ϵ *in* κ *such that* $Adv^{\mathsf{IND\text{-}ID\text{-}CPA}}_{\Pi_{\mathrm{KEM}},\mathcal{A}}(\kappa) < \epsilon$ *holds for any PPT adversary* $\mathcal{A}$*, where* q_{ID} *is the number of queries that* $\mathcal{A}$ *can issue to the oracle in the IND-ID-CPA game.*

2.6 Authentication

Authentication protocols achieve *integrity* (or *authenticity*). Namely, the aim of authentication protocols is to prevent information from being altered or substituted by an adversary. Roughly speaking, in the secret-key setting, information-theoretically secure authentication protocols are traditionally called *authentication codes* (A-codes), and computationally secure ones are called *message authentication codes* (MAC). On the other hand, authentication protocols in the public-key setting are called *digital signature* (DS) schemes. To be precise, authentication protocols which allow a third party to check the validity of signatures are called DS schemes. Therefore, we can actually consider DS schemes not only in the computational security setting but also in the information-theoretic security setting. We here describe A-codes and computationally secure DS schemes, which are relative to our contribution or used in this thesis, as follows.

2.6.1 Authentication Codes

The traditional A-codes for two communicating parties was introduced by Gilbert et al. [69], and later Simmons [132] developed the theory of A-codes. The authentication model in [132] contains a sender and a receiver, who share common secret keys. By usage of authentication codes, they can protect the transmission of a piece of information against an adversary, who can either impersonate the sender and insert a message on the channel, or replace a transmitted massage with another. In 1989, Soete et al. [133] formalized cartesian (or tag-based) authentication codes. Here, we describe cartesian authentication codes.

$\mathcal{M}_{\mathrm{A}}$, $\mathcal{A}$, and $\mathcal{E}$ are a set of possible messages, a set of possible authenticators (or tags), and a finite set of possible common keys, respectively. An A-code Π_{A} consists of three-tuple algorithms (KGen, Auth, Ver) defined as follows.

- $e \leftarrow \mathsf{KGen}(1^{\kappa})$: It takes a security parameter κ and outputs a common key $e \in \mathcal{E}$.
- $\alpha \leftarrow \mathsf{Auth}(e, m)$: It takes the common key e and a message $m \in \mathcal{M}_{\mathrm{A}}$ and outputs an authenticator (or a tag) $\alpha \in \mathcal{A}$. A pair of a message and an authenticator (m, α) is often called an authenticated message.
- *true* or *false* $\leftarrow \mathsf{Ver}(e, m, \alpha)$: It takes the common key e and a pair of a message and an authenticator (m, α), and then outputs *true* if it accepts

them, otherwise it outputs *false*.

We assume that Π_{A} meets the following *correctness* property: For all $\kappa \in \mathbb{N}$, all $e \leftarrow \mathsf{KG}(1^{\kappa})$, and all $m \in \mathcal{M}_{\mathrm{A}}$, it holds that $1 \leftarrow \mathsf{Ver}(e, m, \mathsf{Auth}(e, m))$.

In A-codes, two kinds of attacks are considered: *impersonation attacks* and *substitution attacks*. We describe the traditional security definition of A-codes as follows.

Definition 2.12. *Let Π_{A} be an A-code. Π_{A} is said to be ε-secure if $\max\{P_I, P_S\} \leq \epsilon$, where P_I and P_S are defined as follows.*

1) Impersonation attacks: The adversary tries to generate a fraudulent authenticated message (m, α) that has not been legally generated by the sender but will be accepted by the receiver. The success probability of this attack denoted by P_I is defined by

$$P_I := \max_{(m,\alpha)} \Pr[\mathsf{Ver}(e, m, \alpha) = \mathit{true}],$$

where the probability is taken over random choice of KGen, and the maximum is taken over all possible authenticated messages $(m, \alpha) \in \mathcal{M}_{\mathrm{A}} \times \mathcal{A}$.

2) Substitution attacks: The adversary tries to generate a fraudulent authenticated message (m', α'), that has not been legally generated by the sender but will be accepted by the receiver, after observing a valid authenticated message, (m, α) such that $(m, \alpha) \neq (m', \alpha')$. The success probability of this attack denoted by P_S is defined by

$$P_S := \max_{(m',\alpha')} \max_{(m,\alpha)\neq(m',\alpha')} \Pr[\mathsf{Ver}(e, m', \alpha') = \mathit{true} \mid (m, \alpha)],$$

where the probability is taken over random choice of KGen, and the maximum is taken over all possible authenticated messages $(m, \alpha), (m', \alpha') \in \mathcal{M}_{\mathrm{A}} \times \mathcal{A}$ such that $(m, \alpha) \neq (m', \alpha')$.

Next, we show lower bounds on success probabilities of attacks. For details on the proof of this proposition, please see [132].

Proposition 2.8 ([132])**.** *Let Π_{A} be an ε-secure A-code. Then, we have the following inequality:*

$$\max\{P_I, P_S\} \geq \frac{1}{\sqrt{|\mathcal{E}|}}.$$

And we call Π_{A} to be perfect when it meets the equality in the above inequality.

From the above proposition, we obtain a lower bound on a common key in A-codes.

Corollary 2.2. *Let Π_{A} be an ε-secure A-code. Let $q := \varepsilon^{-1}$. Then, we have*

$$|\mathcal{E}| \geq q^2.$$

Proof. Since $q^{-1} \geq \max\{P_I, P_S\} \geq 1/\sqrt{|\mathcal{E}|}$, we have $|\mathcal{E}| \geq q^2$. □

Several constructions of ϵ-secure A-codes are known. We here describe the most famous construction as follows. Let $\mathbb{F}_q$ be a finite field whose cardinality is q. KG outputs $e := (a, b) \in \mathbb{F}_q^2$. Auth takes e and $m \in \mathbb{F}_q$ as input and outputs $\alpha := am + b$. Ver takes e and (m', α') as input checks whether or not it holds $\alpha' = am' + b$. This construction is ϵ-secure, where $\epsilon = 1/q$, and optimal in the sense that the construction attains the lower bound with equality.

2.6.2 Digital Signature and One-Time Signature

Let $\mathcal{M}_{\mathrm{DS}}$ be a set of messages determined by a security parameter κ. A DS scheme Π_{DS} consists of three-tuple algorithms (KG, Sign, Vrfy) defined as follows.

- $(vk, sigk) \leftarrow \mathsf{KG}(1^\kappa)$: It takes a security parameter κ and outputs a pair of a verification key and a signing key $(vk, sigk)$.
- $\sigma \leftarrow \mathsf{Sign}(sigk, m)$: It takes the signing key $sigk$ and a message $m \in \mathcal{M}_{\mathrm{DS}}$ and outputs a signature σ.
- 1 or $0 \leftarrow \mathsf{Vrfy}(vk, m, \sigma)$: It takes the verification key vk and a pair of a message and a signature (m, σ), and then outputs 1 if it accepts them, otherwise it outputs 0.

We assume that Π_{DS} meets the following *correctness* property: For all $\kappa \in \mathbb{N}$, all $(vk, sigk) \leftarrow \mathsf{KG}(1^\kappa)$, and all $m \in \mathcal{M}_{\mathrm{DS}}$, it holds that $1 \leftarrow \mathsf{Vrfy}(vk, m, \mathsf{Sign}(sigk, m))$.

We describe the notion of unforgeability against chosen message attacks (UF-CMA). Let $\mathcal{A}$ be a PPT adversary, and $\mathcal{A}$'s advantage against the UF-CMA security is defined by

$$Adv^{\mathsf{UF\text{-}CMA}}_{\Pi_{\mathrm{DS}},\mathcal{A}}(\kappa) := \Pr\left[\begin{array}{l|l} \mathsf{Vrfy}(vk, m^*, \sigma^*) \rightarrow 1 & (vk, sigk) \leftarrow \mathsf{KG}(1^\kappa), \\ \wedge\ m^* \notin \{m_i\}_{i=1}^q & (m^*, \sigma^*) \leftarrow \mathcal{A}^{Sign(\cdot)}(vk) \end{array} \right],$$

where *Sign* is a *signing oracle* which takes a message $m \in \mathcal{M}_{\mathrm{DS}}$ as input and returns $\mathsf{Sign}(sigk, m)$, and m_i is the i-th query to *Sign*. $\mathcal{A}$ is allowed to issue arbitrary queries to the above oracle.

Definition 2.13 (UF-CMA). *For $^\exists\kappa_0 \in \mathbb{N}$ and $^\forall\kappa \geq \kappa_0$, a DS scheme Π_{DS} is said to be (q_s, ϵ)-UF-CMA secure if there exists a negligible ϵ in κ such that $Adv^{\mathsf{UF\text{-}CMA}}_{\Pi_{\mathrm{DS}},\mathcal{A}}(\kappa) < \epsilon$ holds for any PPT adversary $\mathcal{A}$, where q_s is the number of queries that $\mathcal{A}$ can issue to the oracle in the UF-CMA game.*

We also define a notion of *strong* UF-CMA (sUF-CMA). Intuitively, $\mathcal{A}$ is allowed to use messages issued to the *Sign* oracle for a forgery. Let $\mathcal{A}$ be a PPT adversary, and $\mathcal{A}$'s advantage against the sUF-CMA security is defined by

$$Adv^{\mathsf{sUF\text{-}CMA}}_{\Pi_{\mathrm{DS}},\mathcal{A}}(\kappa) := \Pr\left[\begin{array}{l|l} \mathsf{Vrfy}(vk, m^*, \sigma^*) \rightarrow 1 \wedge & (vk, sigk) \leftarrow \mathsf{KG}(1^\kappa), \\ (m^*, \sigma^*) \notin \{(m_i, \sigma_i)\}_{i=1}^{q} & (m^*, \sigma^*) \leftarrow \mathcal{A}^{Sign(\cdot)}(vk) \end{array} \right],$$

where (m_i, σ_i) is a pair of the i-th query to the *Sign* oracle and its response.

Definition 2.14 (sUF-CMA). *For* $^\exists\kappa_0 \in \mathbb{N}$ *and* $^\forall\kappa \geq \kappa_0$*, a DS scheme* Π_{DS} *is said to be* (q_s, ϵ)*-sUF-CMA secure if there exists a negligible* ϵ *in* κ *such that* $Adv^{\mathsf{sUF\text{-}CMA}}_{\Pi_{\mathrm{DS}},\mathcal{A}}(\kappa) < \epsilon$ *holds for any PPT adversary* $\mathcal{A}$*, where* q_s *is the number of queries that* $\mathcal{A}$ *can issue to the oracle in the sUF-CMA game.*

In particular, if $q_s = 1$ (i.e., $\mathcal{A}$ is allowed to access the *Sign* oracle only once), then sUF-CMA is also called strong unforgeability against a one-time attack (sUF-OT). DS schemes which meet sUF-OT is often called *one-time signature* (OTS) schemes. Formally, the OTS scheme is defined as follows.

Definition 2.15 (sUF-OT). *If a DS scheme* Π_{DS} *is* $(1, \epsilon)$*-sUF-CMA secure, then it is also* ϵ*-sUF-OT secure.*

So far, many kinds of UF-CMA secure DS schemes (e.g., [27, 46, 71, 76, 146]), and also many kinds of sUF-CMA secure DS schemes including OTS schemes have also been proposed (e.g., [10, 25, 30, 67, 88]).

2.7 Secret Sharing and Information Dispersal

Secret sharing schemes were proposed independently by Shamir [127] and Blakley [16]. The aim of secret sharing (SS) schemes is to disperse the risk of exposure of certain secret information *without secret keys*. For example, in a (k, n)-threshold secret sharing ((k, n)-SS, for short) scheme (e.g. see [127]), a dealer shares a secret among all n participants, and then, at least k participants can reconstruct the secret while any at most $k-1$ participants obtain no information on the secret. SS schemes are also known as one of the important cryptographic primitives. Since Shamir and Blakley proposed secret sharing schemes, various research on them have been reported.

On the other hand, the aim of information dispersal algorithms (IDAs), which were proposed in [117], is to disperse some information so that the original information can be efficiently reconstructed from the dispersed information. In a (k, n)-threshold IDA ((k, n)-IDA, for short), a piece of information (called *fragments*) is distributed among n participants, and then, at least k ($\leq n$) participants can recover the information. The IDA can be regarded as an SS scheme without its security.

2.7.1 Secret Sharing Schemes

Let $\mathcal{P} := \{P_1, P_2, \ldots, P_n\}$ be a set of IDs of all participants. $\mathcal{S}$ is a set of possible secrets with a probability distribution P_S, and we assume $|\mathcal{S}| = 2^\lambda$ for simplicity (i.e. the length of a secret is λ bit), where λ is a polynomial in κ (i.e. $\lambda = p(\kappa)$ for some polynomial p in κ). For every $P_i \in \mathcal{P}$, let $\mathcal{U}_i$ be the set of possible P_i's shares, and let $\mathcal{U} := \bigcup_{i=1}^{n} \mathcal{U}_i$. For any subset of participants $\mathcal{J} = \{P_{i_1}, \ldots, P_{i_j}\} \subset \mathcal{P}$, $\mathcal{U}_\mathcal{J} := \mathcal{U}_{i_1} \times \cdots \times \mathcal{U}_{i_j}$ denotes the set of possible shares held by $\mathcal{J}$, and $u_\mathcal{J} := (u_{i_1}, \ldots, u_{i_j})$. In addition, we consider a (k, n)-*threshold access structure* $\Gamma := (\mathscr{Q}, \mathscr{F})$, where $\mathscr{Q} := \{\mathcal{Q} \subset \mathcal{P} \mid |\mathcal{Q}| \geq k\}$ and $\mathscr{F} := \{\mathcal{F} \subset \mathcal{P} \mid |\mathcal{F}| \leq k-1\}$.[4] An SS scheme with a (k, n)-threshold access structure ((k, n)-SS scheme, for short) Π_{SS} consists of two-tuple algorithms, (SS.Share, SS.Recon), as follows.

- $(u_1, \ldots, u_n) \leftarrow \mathsf{SS.Share}(1^\kappa, \Gamma, s)$: A probabilistic algorithm for generating n shares. It takes a (k, n)-threshold access structure $\Gamma = (\mathscr{Q}, \mathscr{F})$ and a secret $s \in \mathcal{S}$ as input and then outputs n shares $(u_1, \ldots, u_n)$.
- $s \leftarrow \mathsf{SS.Recon}(u_\mathcal{Q})$: A deterministic algorithm for reconstructing a secret. It takes at least k shares $u_\mathcal{Q}$ for $\mathcal{Q} \in \mathscr{Q}$ as inputs and outputs a secret s.

We say that Π_{SS} has the *perfect correctness* property if it meets the following condition: For all $\kappa \in \mathbb{N}$, all Γ, all $s \in \mathcal{S}$, and for all $(u_1, \ldots, u_n) \leftarrow \mathsf{SS.Share}(1^\kappa, \Gamma, s)$, it holds that $s \leftarrow \mathsf{SS.Recon}(u_\mathcal{Q})$ for any $\mathcal{Q} \in \mathscr{Q}$.

To give security formalization of (k, n)-SS, we consider the following notion of Privacy as in [122]. Let $\mathcal{A}$ be a PPT adversary, and $\mathcal{A}$'s advantage against Privacy is defined by

$$Adv^{\mathsf{Privacy}}_{\Pi_{\mathrm{SS}},\mathcal{A}}(\kappa) := \left| \Pr\left[b' = b \;\middle|\; \begin{array}{l} \mathcal{W} \leftarrow \emptyset, \\ (s^{(0)}, s^{(1)}, st) \leftarrow \mathcal{A}(\mathsf{chal}),\ b \xleftarrow{\$} \{0,1\}, \\ (u_1, \ldots, u_n) \leftarrow \mathsf{SS.Share}(1^\kappa, \Gamma, s^{(b)}), \\ b' \leftarrow \mathcal{A}^{Corrupt(\cdot)}(\mathsf{guess}, st) \end{array} \right] - \frac{1}{2} \right|.$$

Here, we require $|s^{(0)}| = |s^{(1)}| = \lambda$, and st is state information. $\mathcal{W}$ is a set of corrupted participants. In addition, *Corrupt* is a *corrupt oracle* which takes an ID P_i as input, and then $\mathcal{W} \leftarrow \mathcal{W} \cup \{P_i\}$ and returns u_i. $\mathcal{A}$ can query to $Corrupt(\cdot)$ until $|\mathcal{W}| = k - 1$.

Based on the above game, we give two security formalization for (k, n)-SS: privacy in the sense of information-theoretic security; and privacy in the sense of computational security. First, we define *perfect privacy*, which means that *no* information is leaked from subthreshold shares in the information-theoretic security sense. Hereafter, a (k, n)-SS scheme with perfect privacy is called a (k, n)-*threshold perfect SS* ((k, n)-PSS, for short) scheme.

[4] SS schemes with the (k, n)-threshold access structure is traditionally called (k, n)-threshold SS schemes.

Definition 2.16 ((k,n)-PSS [122]). *For any $\kappa \in \mathbb{N}$, and any (k,n)-threshold access structure, a (k,n)-SS scheme Π_{SS} is said to be a (k,n)-PSS scheme if it has perfect correctness and $Adv^{\mathsf{Privacy}}_{\Pi_{\mathrm{SS}},\Gamma,\mathcal{A}}(\kappa) = 0$ for any computationally-unbounded adversary $\mathcal{A}$.*

Note that the traditional (k,n)-SS schemes such as [16, 82, 127] are equivalent to (k,n)-PSS, so perfect privacy can be defined by using Shannon entropy in a traditional manner as follows. Let S and $U_{\mathcal{J}}$ be random variables which take values on $\mathcal{S}$ and $\mathcal{U}_{\mathcal{J}}$, respectively.

Definition 2.17 (Another definition of (k,n)-PSS). *Let Π_{SS} be a (k,n)-SS scheme. Π_{SS} is said to be secure if for any $\mathcal{F} \in \mathcal{PS}(\mathcal{P}, k-1)$, it holds that*

$$H(S \mid U_{\mathcal{F}}) = H(S).$$

Next, we define *computational privacy*, which means that no information is leaked in the computational security sense. Hereafter, a (k,n)-SS scheme with computational privacy is called a *(k,n)-threshold computational SS* ((k,n)-CSS, for short) scheme.

Definition 2.18 ((k,n)-CSS [86, 122]). *For $^{\exists}\kappa_0 \in \mathbb{N}$, $^{\forall}\kappa \geq \kappa_0$, and any (k,n)-threshold access structure, a (k,n)-SS scheme Π_{SS} is said to be an ϵ-(k,n)-CSS scheme if it has perfect correctness and there exists a negligible ϵ in κ such that $Adv^{\mathsf{Privacy}}_{\Pi_{\mathrm{SS}},\Gamma,\mathcal{A}}(\kappa) < \epsilon$ for any PPT adversary $\mathcal{A}$.*

Next, we describe an well-known lower bound on share size for (k,n)-PSS schemes.

Proposition 2.9 ([82]). *Let Π_{SS} be any (k,n)-PSS scheme. Then, for any $i \in [n]$, it holds that $|u_i| \geq \lambda$, where λ is the bit-length of the secret.*

This lower bound indicates that each share size must be larger than or equal to the underlying secret size on (k,n)-PSS schemes. Actually, this lower bound is tight since there exist several constructions that attain the lower bound with equality. We briefly describe the most famous construction, Shamir's scheme [127]. Let $\mathbb{F}_q$ be a finite field whose cardinality is q such that $q > n$. $\mathsf{SS.Share}$ takes a secret s as input, and chooses a polynomial $f(x) := s + \sum_{i=1}^{k-1} a_i x^i$ over $\mathbb{F}_q$ with a variable x uniformly at random. It outputs $(u_1, \ldots, u_n) := (f(P_1), \ldots, f(P_n))$, where every P_i is encoded as $P_i \in \mathbb{F}_q \setminus \{0\}$. $\mathsf{SS.Recon}$ takes k shares $u_{i_1}, \ldots, u_{i_k}$ as input, and outputs $s = \sum_{j=1}^{k} (\prod_{l \neq j} \frac{P_{i_j}}{P_{i_j} - P_{i_l}}) f(P_{i_j})$ by Lagrange interpolation.

On the other hand, Krawczyk successfully reduced the share size by restricting adversary's computational power. Krawczyk's scheme is as follows: In the share algorithm $\mathsf{CSS.Share}$, a secret s is encrypted by an SKE scheme, namely $c \leftarrow \mathsf{E}(k, s)$, and shares of the secret key k is generated by using

a (k,n)-PSS scheme, namely $(\hat{u}_1,\ldots,\hat{u}_n) \leftarrow \mathsf{SS.Share}(1^\kappa,\Gamma,k)$. Further, the resulting ciphertext c is split into n fragments by using a (k,n)-IDA, which will appear in the next subsection, namely $(\tilde{u}_1,\ldots,\tilde{u}_n) \leftarrow \mathsf{IDA.Share}(1^\kappa,\Gamma,c)$. $\mathsf{CSS.Share}$ finally outputs $(u_1,\ldots,u_n)$, where each $u_i := (\hat{u}_i,\tilde{u}_i)$ $(1 \le i \le n)$. In the reconstruct algorithm $\mathsf{CSS.Recon}$, c and k are reconstructed by using $\mathsf{SS.Recon}$ and $\mathsf{IDA.Recon}$, respectively, with at least k shares, and s is recovered from k and c, namely $s \leftarrow \mathsf{D}(k,c)$. The share size in an ϵ-(k,n)-CSS scheme [86] (i.e., Krawczyk's scheme) is $\frac{\lambda}{k} + |K| + COH_{\mathrm{DEM}}(\kappa)$, where $|K|$ and $COH_{\mathrm{DEM}}(\kappa)$ mean the key size and the ciphertext-overhead (i.e. the ciphertext-length excluding the plaintext-length, which depends on a security parameter κ) of the underlying ϵ-$\mathsf{FTG\text{-}CPA}$ secure DEM, which is used in the construction, respectively.

Remark 2.1. *As in [86], for simplicity, we also assume that $COH_{\mathrm{DEM}}(\kappa) = 0$ in the following, since it can be achieved by carefully selecting a symmetric encryption scheme (e.g., when we use an appropriate block cipher as the underlying symmetric encryption scheme and a secret size $\lambda = c \cdot |K|$, where c is constant). Then, we can rewrite the share size as $|u_i| = \frac{\lambda}{k} + |K|$.*

2.7.2 Information Dispersal Algorithm

We describe an IDA with a (k,n)-threshold access structure ((k,n)-IDA, for short). As one may think, IDAs are quite similar to SS schemes. However, IDAs differ from SS schemes as there are no restriction whatsoever about the sets which are not in $\mathscr{Q}$. Formally, a (k,n)-IDA Π_{IDA} consists of two-tuple algorithms ($\mathsf{IDA.Share}$, $\mathsf{IDA.Recon}$) defined as follows:

- $(u_1,\ldots,u_n) \leftarrow \mathsf{IDA.Share}(\Gamma,s)$: A probabilistic algorithm for generating n fragments. It takes a (k,n)-threshold access structure $\Gamma = (\mathscr{Q},\mathscr{F})$ and a file $s \in \mathcal{S}$ as input and then outputs n fragments $(s_1,\ldots,s_n)$.
- $s \leftarrow \mathsf{IDA.Recon}(u_{\mathcal{Q}})$: A deterministic algorithm for reconstructing a file. It takes at least k fragments $u_{\mathcal{Q}}$ for $\mathcal{Q} \in \mathscr{Q}$ as inputs and outputs a file s.

Definition 2.19 ((k,n)-IDA [115, 117])**.** *Π_{IDA} is a (k,n)-IDA if it meets the following perfect correctness property: For all possible $s \in \mathcal{S}$, and for all possible $(u_1,\ldots,u_n) \leftarrow \mathsf{IDA.Share}(\Gamma,s)$, it holds that $s \leftarrow \mathsf{IDA.Recon}(u_{\mathcal{Q}})$ for any $\mathcal{Q} \in \mathscr{Q}$.*

A lower bound on fragment size required for (k,n)-IDAs is as follows.

Proposition 2.10 ([115, 117])**.** *Let Π_{IDA} be any (k,n)-IDA. Then, for any $i \in [n]$, it holds that $|u_i| \ge \frac{\lambda}{k}$, where λ is the bit-length of the file.*

We can construct a (k,n)-IDA in a way similar to Shamir's scheme. Let $\mathbb{F}_q$ be a finite field whose cardinality is q such that $q > n$. $\mathsf{IDA.Share}$ takes a file $s =$

$(s_0, \ldots, s_{k-1}) \in \mathbb{F}_q^k$ as input, and constructs a polynomial $f(x) := \sum_{i=0}^{k-1} s_i x^i$ over $\mathbb{F}_q$ with a variable x. It outputs $(u_1, \ldots, u_n) := (f(P_1), \ldots, f(P_n))$, where every P_i is encoded as $P_i \in \mathbb{F}_q \setminus \{0\}$. IDA.Recon reconstructs the polynomial $f(x) = \sum_{j=1}^{k} (\prod_{l \neq j} \frac{x - P_{i_l}}{P_{i_j} - P_{i_l}}) f(P_{i_j})$ from k fragments $u_{i_1}, \ldots, u_{i_k}$ by Lagrange interpolation, and hence gets $s = (s_0, \ldots, s_{k-1})$.

Chapter 3

Computational Timed-Release Cryptography

3.1 Contribution in This Chapter

In this chapter, we aim to develop computational timed-release cryptography by newly introducing cryptographic primitives in addition to TR-PKE (e.g., [33, 35, 39]), timed-release and time capsule signatures (e.g., [56, 64, 65, 89]), and timed-commitments [29]. Our main purpose in this chapter is *to realize a CSS scheme with timed-release functionality in a generic and efficient way* in terms of the share size. Specifically, in Section 3.3, we begin with newly formalizing a model and a security notion of (k, n)-TR-CSS based on those of (k, n)-SS (e.g., [127, 16, 82]) and those of TR-PKE. In addition, we propose two kinds of constructions of (k, n)-TR-CSS schemes, starting from an IB-KEM, and we finally succeed to add timed-release functionality to a traditional CSS scheme—especially for Krawczyk's scheme [86]—with small overhead, which seems to be *almost optimal*. In particular, our two kinds of constructions have the following features:

- The first one is a *generic* construction of a (k, n)-TR-CSS scheme based on Krawczyk's (k, n)-CSS scheme, which is a simple, elegant construction. In addition to using the idea of Krawczyk's scheme, since all currently-known generic constructions of TR-PKE schemes [39, 104, 96] are proposed starting from identity-based cryptographic schemes such as IBE [27], we also use an IB-KEM [13, 84, 85] as an building block in our construction.

- The second one is a *more efficient* construction than the first one. In this construction, we use a specific IB-KEM as an building block. However, to the best of our knowledge, all currently-known IB-KEMs are IND-ID-CCA secure, and there is no IND-ID-CPA secure IB-KEM, which suits the construction of our (k, n)-TR-CSS scheme. In particular, Kiltz

and Galindo [84, 85] proposed one of the most efficient (but IND-ID-CCA secure) IB-KEM in terms of the ciphertext size,[1] however, it guarantees too much security for constructing TR-CSS schemes. Therefore, to reduce the ciphertext size, we construct a IND-ID-CPA secure IB-KEM based on the IND-ID-CCA secure IB-KEM [84, 85], and then we construct a (k, n)-TR-CSS scheme by using the specific IB-KEM. In this sense, though this construction is not fully generic, we achieve more efficiency (i.e., shorter share size) than the first construction. Importantly, this construction can realize efficient (k, n)-TR-CSS scheme in the sense that the share size is close to that of Krawczyk's (k, n)-CSS scheme when k is sufficiently large.

Therefore, our study on TR-CSS can be regarded as a natural extension of CSS, in particular, Krawczyk's CSS in terms of both a model and constructions.

Next, in Section 3.4, we consider realizing threshold encryption [50] with timed-release functionality (TR-TE) from TR-CSS in a generic and efficient way, since it is natural to consider constructing threshold encryption from secret sharing in general. By Dodis and Katz [54], it is shown that: (i) a threshold encryption scheme can be constructed from a multiple encryption scheme in a generic and simple way; and (ii) a multiple encryption scheme can be constructed from PKE schemes, an OTS scheme and a CSS scheme in a generic and simple way.

By effectively applying their paradigm even in the context of timed-release security, we show that: (i') a TR-TE scheme can be constructed from a multiple encryption scheme with timed-release functionality (TR-ME) by using the same way as in [54]; (ii') a TR-ME scheme can be constructed from PKE schemes, an OTS scheme, and a TR-CSS scheme by using the same way as in [54], which is much *more efficient* than the construction from TR-PKE schemes, an OTS scheme, and a CSS scheme. From these results above, our proposal of TR-CSS is important for constructing a TR-ME scheme and a TR-TE scheme in a generic and efficient way. Moreover, multiple encryption has many other applications, and hence, it is expected that TR-CSS can provide these applications with timed-release functionality, e.g., broadcast encryption with timed-release functionality.

Related work on (computational) timed-release cryptography. Since Rivest et al. gave a formal definition of timed-release encryption in [121], various research on timed-release cryptography including signatures (e.g., [64, 65]), commitments (e.g., [29]), and encryption (e.g., [51, 93]) have been done based on computational security. In particular, TR-PKE has been particularly investigated. Di Crescenzo et al. [51] proposed the first TR-PKE scheme. How-

[1] The ciphertext size in the underlying IB-KEM leads directly to the share size of the resulting TR-CSS scheme.

ever, in their scheme the receiver needs to interact with a time-server, which broadcasts information related to time (called *time-signals* in this thesis) for decrypting ciphertexts. It is desirable that the time-server only broadcast time-signals (without any interaction with receivers) since there are many receivers in the real systems. Chan and Blake [36] proposed the first non-interactive TR-PKE scheme, but did not present a formal security definition. Cathalo et al. [33] and Chalkias et al. [35] proposed direct constructions of TR-PKE schemes based on number-theoretic assumptions in the random oracle model. Independently, Cheon et al. [39] proposed a generic construction of TR-PKE which is provably secure in the standard model. Nakai et al. [104] and Matsuda et al. [96] proposed more efficient generic constructions of TR-PKE schemes. Also, Fujioka et al. [62] and Kikuchi et al. [83] proposed generic constructions of TR-PKE schemes that guarantee strong security in the random oracle model. In the strong security setting, it is assumed that there are multiple receivers, since all receivers' public-keys are closely related to each other in TR-PKE. Specifically, an adversary can query not only the specified time and a ciphertext but also a public-key to the decryption oracle, and it can get a result decrypted by a secret-key corresponding to the public-key. Recently, time-specific encryption (TSE for short) was proposed by Paterson and Quagila [113] as an extension of TR-PKE. In TSE, a sender can encrypt a plaintext and specify any *time interval* such that a receiver can only decrypt the ciphertext within this time interval. It is also known that TR-PKE has close relations to other cryptographic protocols. Actually, Choen et al. [38] recently shows relationships between TR-PKE and key-insulated public-key encryption [55], and Chow et al. [40] also shows relationships between TR-PKE and certificateless encryption [2].

Related work on secret sharing with similar functionalities. There are many related works, e.g., fully dynamic secret sharing schemes [20], on-line secret sharing schemes [31], and secret sharing schemes with disenrollment capability [15]. In a nutshell, in such schemes, a dealer generates and distributes shares securely, and later on, the dealer can generate and publicly broadcast information for changing the shared secret or qualified sets. Our scheme differs from such schemes in that broadcasted information is generated independently of a shared secret (i.e. the broadcasted information can be generated by a third party).

The other type of related works dealing with the concept of time is a proactive secret sharing scheme [75]. In our TR-CSS scheme (and TR-SS schemes which will appear in Chapter 4), broadcast channels among all participants are assumed. Each participant generates and broadcasts updating information to other participants, and then, they refresh their shares by using the updating information. Hence, shares leaked before that time become irrelevant. Namely, proactive secret sharing schemes realize *share-updating functionality*. In our

scheme, such broadcast channels are not assumed and the concept of both schemes is completely different, though both schemes deal with the concept of time.

3.2 Timed-Release Public-Key Encryption

3.2.1 Model and Security Definition

In timed-release cryptography including TR-PKE, we consider that there exists a time-server whose role is to periodically generate and broadcast information regarding each time (or, time-period). In this thesis, we call this information *time-signals*. The time-server do not have any interaction with any other entities, namely, it independently generates time-signals and only broadcasts them.

In TR-PKE, a time-server TS generates a public parameter and his master secret key. Then, a receiver generates his public/secret keys, and discloses his public key. A sender S specifies future time when he wants the receiver R to decrypt the ciphertext, and encrypts a plaintext with the public key. R cannot decrypt the ciphertext until the specified time comes. TS periodically generates and broadcasts a time-signal at every time-period. After receiving the time-signal at the specified time, then R finally gets the plaintext by decrypting the ciphertext.

$\mathcal{T}$ is a set of time and $\mathcal{M}_{\mathrm{TRE}}$ is a set of plaintexts determined by a security parameter κ. A TR-PKE scheme Π_{TPKE} consists of five-tuple algorithms (TR.Init, TRKeyGen, TR.Release, TR.Enc, TR.Dec) defined as follows:

- $(params, tsk) \leftarrow \mathsf{TR.Init}(1^{\kappa})$: A probabilistic algorithm for setup. It takes a security parameter κ as input and outputs a public parameter $params$ and time-server's master secret key tsk.
- $(upk, usk) \leftarrow \mathsf{TR.KeyGen}(params)$: An algorithm for key generation. It takes the public parameter $params$ as input and outputs a pair of public key upk and a secret key usk.
- $s_t \leftarrow \mathsf{TR.Release}(tsk, t)$: An algorithm for time-signal generation. It takes the master secret key tsk and time $t \in \mathcal{T}$ as input and outputs a time-signal s_t at time t.
- $ct_t \leftarrow \mathsf{TR.Enc}(params, upk, m, t)$: A probabilistic algorithm for encryption. It takes the public parameter $params$, a public key upk, time t, and a plaintext $m \in \mathcal{M}_{\mathrm{TRE}}$ as input and then outputs a ciphertext ct_t.
- m or $\perp \leftarrow \mathsf{TR.Dec}(params, usk, s_t, ct_t, t)$: A deterministic algorithm for decryption. It takes the public parameter $params$, a secret key usk, a time-signal s_t at the specified time t, a ciphertext ct_t, and the specified time t as input, and then outputs a plaintext $m \in \mathcal{M}_{\mathrm{TRE}}$ or $\perp \notin \mathcal{M}_{\mathrm{TRE}}$.

In the above model, we assume that Π_{TPKE} meets the following *correctness* property: For all $\kappa \in \mathbb{N}$, all $(params, tsk) \leftarrow \mathsf{TR.Init}(1^\kappa)$, all $(upk, usk) \leftarrow \mathsf{TR.KeyGen}(params)$, all $t \in \mathcal{T}$, and all $m \in \mathcal{M}_{\mathrm{TRE}}$, it holds that $m \leftarrow \mathsf{TR.Dec}(params, usk, \mathsf{TR.Release}(tsk, t), \mathsf{TR.Enc}(params, upk, m, t), t)$.

We describe two security notions: Indistinguishability against chosen plaintext attack (IND-CPA) and indistinguishability against chosen time and plaintext attack (IND-CTPA). The former is a security notion against a *curious* time-server.[2] In fact, formalization of this notion is almost the same as the IND-CPA security of PKE. In the latter, we consider malicious receivers attempting to obtain information on the plaintext before its specified time. Let $\mathcal{A}$ be a PPT adversary, and $\mathcal{A}$'s advantage against the IND-CPA security is defined by

$$Adv^{\mathsf{IND\text{-}CPA}}_{\Pi_{\mathrm{TPKE}},\mathcal{A}}(\kappa) := \left| \Pr\left[b' = b \;\middle|\; \begin{array}{l} (params, tsk) \leftarrow \mathsf{TR.Init}(1^\kappa), \\ (upk, usk) \leftarrow \mathsf{TR.KeyGen}(params), \\ (m_0^*, m_1^*, t^*, st) \leftarrow \mathcal{A}(\mathsf{chal}, params, tsk, upk), \\ b \xleftarrow{\$} \{0,1\},\ ct_{t^*}^* \leftarrow \mathsf{TR.Enc}(params, upk, m_b, t^*), \\ b' \leftarrow \mathcal{A}(\mathsf{guess}, c_{t^*}^*, st) \end{array} \right] - \frac{1}{2} \right|.$$

Here, we require $|m_0^*| = |m_1^*|$, and st is state information.

Definition 3.1 (IND-CPA). *For $^\exists\kappa_0 \in \mathbb{N}$ and $^\forall\kappa \geq \kappa_0$, a TR-PKE scheme Π_{tre} is said to be ϵ-IND-CPA secure if there exists a negligible ϵ in κ such that $Adv^{\mathsf{IND\text{-}CPA}}_{\Pi_{tre},\mathcal{A}}(\kappa) < \epsilon$ holds for any PPT adversary $\mathcal{A}$.*

Next, we define the IND-CTPA security. Let $\mathcal{A}$ be a PPT adversary, and $\mathcal{A}$'s advantage against the IND-CTPA security is defined by

$$Adv^{\mathsf{IND\text{-}CTPA}}_{\Pi_{\mathrm{TPKE}},\mathcal{A}}(\kappa) := \left| \Pr\left[b' = b \;\middle|\; \begin{array}{l} (params, tsk) \leftarrow \mathsf{TR.Init}(1^\kappa), \\ (upk, usk) \leftarrow \mathsf{TR.KeyGen}(params), \\ (m_0^*, m_1^*, t^*, st) \leftarrow \mathcal{A}^{\mathcal{R}(\cdot)}(\mathsf{chal}, params, upk, usk), \\ b \xleftarrow{\$} \{0,1\},\ ct_{t^*}^* \leftarrow \mathsf{TR.Enc}(params, upk, m_b, t^*), \\ b' \leftarrow \mathcal{A}^{\mathcal{R}(\cdot)}(\mathsf{guess}, c_{t^*}^*, st) \end{array} \right] - \frac{1}{2} \right|.$$

Here, we require $|m_0^*| = |m_1^*|$, and st is state information. In addition, $\mathcal{R}(\cdot)$ is a *time-signal generation oracle* which takes time t as input, and then returns $\mathsf{TR.Release}(tsk, t)$. $\mathcal{A}$ is allowed to issue arbitrary queries to the above oracle in the chal stage, however, $\mathcal{A}$ cannot submit t^* to the oracle after deciding t^*.

[2] In timed-release cryptography, it is usually assumed that the time-server always generates and broadcasts time-signals correctly, however tries to get some information on the underlying plaintexts from transmitted ciphertexts. Such an adversary is often called the curious one.

Definition 3.2 (IND-CTPA). *For* $^{\exists}\kappa_0 \in \mathbb{N}$ *and* $^{\forall}\kappa \geq \kappa_0$*, a TR-PKE scheme* Π_{TPKE} *is said to be* (q_{ts}, ϵ)*-IND-CTPA secure if there exists a negligible* ϵ *in* κ *such that* $Adv^{\mathsf{IND\text{-}CTPA}}_{\Pi_{\text{TPKE}},\mathcal{A}}(\kappa) < \epsilon$ *holds for any PPT adversary* $\mathcal{A}$*, where* q_{ts} *is the number of queries that* $\mathcal{A}$ *can issue to the oracle in the IND-CTPA game.*

3.3 Timed-Release Computational Secret Sharing

We consider the model of TR-CSS so as to become an extension of CSS, and the basic idea comes from based on models of (C)SS and TR-PKE. As in TR-PKE, we consider the presence of a *time-server*, whose role is to periodically generate and broadcast *time-signals*. The time-server executes a setup algorithm and a time-signal generation algorithm in TR-PKE, hence, it is natural and reasonable that we assume these algorithms in TR-CSS. Note that we assume that the time-server is curious. Namely, the curious time-server always generates time-signals correctly, however, it may attempt to collude some participants and guess the secret.

In TR-CSS, not only shares of any qualified set but also a time-signal at time specified by a dealer are required for reconstructing a secret. Namely, even all participants cannot obtain any information on the secret from their shares without the time-signal at the specified time.

3.3.1 The Model of (k, n)-TR-CSS

We consider a TR-CSS scheme with a (k, n)-threshold access structure (a (k, n)-TR-CSS scheme for short). Informally, (k, n)-TR-CSS is executed as follows. First, a time-server TS generates a master public key and a master secret key. Next, a dealer D specifies future time, as D wants, when a secret can be reconstructed by at least k participants (we call the time *the specified time*), and he generates n shares from the secret by using the master public key. And, D sends shares to corresponding participants, respectively, via secure channels. The time-server TS periodically broadcasts a time-signal which is generated by using his secret key. When the specified time has come, at least k participants can compute the secret by using both their shares and the time-signal of the specified time. Let $\mathcal{T}$ be a set of time. For any subset of participants $\mathcal{J} = \{P_{i_1}, \ldots, P_{i_j}\} \subset \mathcal{P}$, $u^{(t)}_{\mathcal{J}} := (u^{(t)}_{i_1}, \ldots, u^{(t)}_{i_j})$, where $u^{(t)}_i$ is P_i's share at the specified time t.

A (k, n)-TR-CSS scheme Π_{TCSS} consists of four-tuple algorithms (Setup, Release, Share, Recon) defined as follows:

- $(mpk, msk) \leftarrow \mathsf{Setup}(1^\kappa)$: A probabilistic algorithm for key generation. It takes a security parameter κ as input and outputs a master public key mpk and a master secret key msk.

- $ts^{(t)} \leftarrow \mathsf{Ext}(msk, t)$: An algorithm for generating time-signals. It takes

the master secret key msk and time $t \in \mathcal{T}$ as input and outputs a time-signal $ts^{(t)}$ at time t.

- $(u_1^{(t)}, \ldots, u_n^{(t)}) \leftarrow \mathsf{Share}(\Gamma, mpk, s, t)$: A probabilistic algorithm for generating n shares. It takes a (k, n)-threshold access structure $\Gamma = (\mathscr{Q}, \mathscr{F})$, a master public key mpk, a secret $s \in \mathcal{S}$ and a specified time t as input, and then outputs n shares $(u_1^{(t)}, \ldots, u_n^{(t)})$ at time t .
- $s \leftarrow \mathsf{Recon}(u_{\mathcal{Q}}^{(t)}, ts^{(t)})$: A deterministic algorithm for reconstructing a secret. It takes at least k shares $u_{\mathcal{Q}}^{(t)}$ for $\mathcal{Q} \in \mathscr{Q}$ and a time-signal $ts^{(t)}$ at specified time t as inputs, and outputs a secret s.

We say that Π_{TCSS} has the *perfect correctness* property if it meets the following condition: For all $\kappa \in \mathbb{N}$, all $s \in \mathcal{S}$, $(mpk, msk) \leftarrow \mathsf{Setup}(1^\kappa)$, all $t \in \mathcal{T}$, all $ts^{(t)} \leftarrow \mathsf{Ext}(msk, t)$, and all $(u_1^{(t)}, \ldots, u_n^{(t)}) \leftarrow \mathsf{Share}(\Gamma, mpk, s, t)$, it holds that $s \leftarrow \mathsf{Recon}(u_{\mathcal{Q}}^{(t)}, ts^{(t)})$ for any $\mathcal{Q} \in \mathscr{Q}$.

Remark 3.1. *In the case that a time-server does not exist (i.e., $\mathcal{T} = \emptyset$, mpk is a security parameter κ, and msk is an empty string), the model of (k, n)-TR-CSS can be regarded as that of traditional (k, n)-SS. Namely, our model of TR-CSS includes the model of traditional secret sharing schemes.*

3.3.2 Security Definition of (k, n)-TR-CSS

To discuss security, we convert security notions of TR-PKE into those of CSS: Even a curious time-server who colludes with at most $k - 1$ participants can obtain no information on the secret; and all participants can obtain no information on the secret without a time-signal at the specified time. Hence, we consider the following two notions: privacy against a curious time-server (Type-I Privacy) and privacy against participants (Type-II Privacy). Note that if a time-server does not exist, Type-I Privacy is the same as Privacy of (k, n)-CSS (Definition 2.18).

First, we formalize Type-I Privacy. Let $\mathcal{A}$ be a PPT adversary, and $\mathcal{A}$'s advantage against Type-I Privacy is defined by

$$Adv^{\mathsf{Type\text{-}I}}_{\Pi_{\mathrm{TCSS}},\Gamma,\mathcal{A}}(\kappa) := \left| \Pr\left[b' = b \;\middle|\; \begin{array}{l} \mathcal{W} \leftarrow \emptyset,\ (mpk, msk) \leftarrow \mathsf{Setup}(1^\kappa), \\ (s^{(0)}, s^{(1)}, t^*, st) \leftarrow \mathcal{A}(\mathsf{chal}, mpk, msk), \\ b \xleftarrow{\$} \{0,1\}, \\ (u_1^{(t^*)}, \ldots, u_n^{(t^*)}) \leftarrow \mathsf{Share}(\Gamma, mpk, s^{(b)}, t^*), \\ b' \leftarrow \mathcal{A}^{Corrupt(\cdot)}(\mathsf{guess}, st) \end{array} \right] - \frac{1}{2} \right|.$$

Here, we require $|s^{(0)}| = |s^{(1)}| = \lambda$, and st is state information including $s^{(0)}$, $s^{(1)}$ and t^*. $\mathcal{W}$ is a set of corrupted participants and $Corrupt(\cdot)$ is a *corrupt*

oracle which takes an ID P_i as input, and then $\mathcal{W} \leftarrow \mathcal{W} \cup \{P_i\}$ and returns $u_i^{(t^*)}$. A can query to *Corrupt*$(\cdot)$ until $|\mathcal{W}| = k - 1$. Note that we do not consider the *share oracle* such that it takes any s, t, and Γ as input and outputs $\mathsf{Share}(\Gamma, mpk, s, t)$, since $\mathcal{A}$ computes this by himself. Therefore, $\mathcal{A}$ can even get shares of the challenge secrets $s^{(0)}$ and $s^{(1)}$ at any t.

Definition 3.3 (Type-I Privacy). *For* $^{\exists}\kappa_0 \in \mathbb{N}$, $^{\forall}\kappa \geq \kappa_0$, *and any* (k, n)*-threshold access structure, a* (k, n)*-TR-CSS scheme* Π_{TCSS} *meets* ϵ*-Type-I Privacy if there exists a negligible* ϵ *in* κ *such that* $Adv^{\textsf{Type-I}}_{\Pi_{\mathrm{TCSS}},\Gamma,\mathcal{A}}(\kappa) < \epsilon$ *for any PPT adversary* $\mathcal{A}$.

We next formalize Type-II Privacy. Let $\mathcal{A}$ be a PPT adversary, and $\mathcal{A}$'s advantage against Type-II Privacy is defined by

$$Adv^{\textsf{Type-II}}_{\Pi_{\mathrm{TCSS}},\Gamma,\mathcal{A}}(\kappa) := \left| \Pr\left[b' = b \;\middle|\; \begin{array}{l} \mathcal{W} = \mathcal{P},\ (mpk, msk) \leftarrow \mathsf{Setup}(1^\kappa), \\ (s^{(0)}, s^{(1)}, t^*, st) \leftarrow \mathcal{A}^{Release(msk,\cdot)}(\mathsf{chal}, mpk), \\ b \xleftarrow{\$} \{0,1\}, \\ (u_1^{(t^*)}, \ldots, u_n^{(t^*)}) \leftarrow \mathsf{Share}(\Gamma, mpk, s^{(b)}, t^*), \\ b' \leftarrow \mathcal{A}^{Release(msk,\cdot)}(\mathsf{guess}, u_1^{(t^*)}, \ldots, u_n^{(t^*)}, st) \end{array} \right] - \frac{1}{2} \right|.$$

Here, we require $|s^{(0)}| = |s^{(1)}| = \lambda$, and st is state information including $s^{(0)}$, $s^{(1)}$ and t^*. $\mathcal{W}$ is the set of all (corrupted) participants.[3] *Release*$(msk, \cdot)$ is a *time-signals generation oracle* which takes time t as input, and returns $\mathsf{Ext}(msk, t)$. $\mathcal{A}$ is allowed to access the above oracle at most q_t times at any time, where q_t is polynomial in κ. However, $\mathcal{A}$ cannot submit the target time t^* to *Release*$(msk, \cdot)$ after the chal stage. We do not consider the share oracle by the same reason as in Type-I Privacy game.

Definition 3.4 (Type-II Privacy). *For* $^{\exists}\kappa_0 \in \mathbb{N}$, $^{\forall}\kappa \geq \kappa_0$, *and any* (k, n)*-threshold access structure, a* (k, n)*-TR-CSS scheme* Π_{TCSS} *meets* (q_t, ϵ)*-Type-II Privacy if there exists a negligible* ϵ *in* κ *such that* $Adv^{\textsf{Type-II}}_{\Pi_{\mathrm{TCSS}},\Gamma,\mathcal{A}}(\kappa) < \epsilon$ *for any PPT adversary* $\mathcal{A}$*, where* q_t *is the number of queries that* $\mathcal{A}$ *can issue to the oracle in the Type-II Privacy game.*

Definition 3.5 (Security of (k, n)-TR-CSS). *A* (k, n)*-TR-CSS scheme* Π_{TCSS} *is said to be a* $(q_t, \epsilon_1, \epsilon_2)$*-*$(k, n)$*-TR-CSS scheme if it has perfect correctness,* ϵ_1*-Type-I Privacy and* (q_t, ϵ_2)*-Type-II Privacy.*

3.3.3 Generic Construction of a (k, n)-TR-CSS scheme

We propose a generic construction of a (k, n)-TR-CSS scheme. Our generic construction can be regarded as extension of Krawczyk's CSS scheme [86].

[3] In this game, we consider $\mathcal{W} = \mathcal{P}$ (not a proper subset of $\mathcal{P}$), since we want to focus on the strongest security.

The idea of our construction is to combine Krawczyk's CSS scheme and an IB-KEM. It is natural and reasonable to use an IB-KEM as an building block in our constructions, since all currently-known TR-PKE schemes [39, 104, 96] are also constructed from an IBE scheme or an IB-KEM (and a PKE scheme).

Let Π_{SS}=(SS.Share, SS.Recon) be a (k,n)-SS scheme, Π_{IDA}=(IDA.Share, IDA.Recon) be a (k,n)-IDA, Π_{KEM}=(IB.Setup, IB.Gen, IB.Encaps, IB.Decaps) be an IB-KEM, and let Π_{DEM}=(E, D) be a DEM. Suppose that $\mathcal{T} \subset \mathcal{ID}_{\mathrm{KEM}}$. Then, a (k,n)-TR-CSS scheme Π_{TCSS}=(Setup, Ext, Share, Recon) is constructed as follows.

- $(mpk, msk) \leftarrow \mathsf{Setup}(1^\kappa)$:
  1. Output $(mpk, msk) := (prm, mk) \leftarrow \mathsf{IB.Setup}(1^\kappa)$.
- $ts^{(t)} \leftarrow \mathsf{Ext}(msk, t)$:
  1. Output $ts^{(t)} := sk_t \leftarrow \mathsf{IB.Gen}(prm, mk, t)$.
- $(u_1^{(t)}, \ldots, u_n^{(t)}) \leftarrow \mathsf{Share}(\Gamma, mpk, s, t)$:
  1. $(K, c_t) \leftarrow \mathsf{IB.Encaps}(prm, t)$.
  2. $C \leftarrow \mathsf{E}(K, s)$.
  3. $(\tilde{u}_1, \ldots, \tilde{u}_n) \leftarrow \mathsf{IDA.Share}(1^\kappa, \Gamma, C)$.
  4. $(\hat{u}_1, \ldots, \hat{u}_n) \leftarrow \mathsf{SS.Share}(1^\kappa, \Gamma, c_t)$.
  5. Output $u_i^{(t)} := (\tilde{u}_i, \hat{u}_i)$ $(1 \leq i \leq n)$.
- $s \leftarrow \mathsf{Recon}(\hat{u}_{\mathcal{Q}}, ts^{(t)})$:
  1. $c_t \leftarrow \mathsf{SS.Recon}(1^\kappa, \hat{u}_{\mathcal{Q}})$.
  2. $C \leftarrow \mathsf{IDA.Recon}(1^\kappa, \tilde{u}_{\mathcal{Q}})$.
  3. $K \leftarrow \mathsf{IB.Decaps}(prm, sk_t, c_t)$.
  4. Output $s \leftarrow \mathsf{D}(K, C)$.

We can show that the resulting (k,n)-TR-CSS scheme in the above construction is secure, if a given (k,n)-SS scheme is a (k,n)-PSS scheme, a given IB-KEM meets IND-ID-CPA, and a given DEM meets FTG-CPA, as follows.

Theorem 3.1. *If Π_{DEM} is ϵ_1-FTG-CPA secure, Π_{SS} is a (k,n)-PSS scheme, and Π_{KEM} is (q_{ID}, ϵ_2)-IND-ID-CPA secure, then the resulting (k,n)-TR-CSS scheme Π_{TCSS} in the above construction is a $(q_t, \delta_1, \delta_2)$-$(k,n)$-TR-CSS scheme, where $q_t = q_{ID}$, $\delta_1 \leq \epsilon_1$ and $\delta_2 \leq \epsilon_1 + 2\epsilon_2$.*

Proof. It is straightforward that the above construction meets perfect correctness. The rest of the proof follows from the following lemmas.

Lemma 3.1. *If Π_{DEM} is ϵ_1-FTG-CPA secure and Π_{SS} is a (k,n)-PSS scheme, then the resulting (k,n)-TR-CSS scheme Π_{TCSS} in the above construction meets δ-Type-I Privacy, where $\delta \leq \epsilon_1$.*

Proof. Let $\mathcal{A}$ be a Type-I Privacy adversary against the proposed (k,n)-TR-CSS scheme Π_{TCSS}. We define the following two games:

Game$_0$: This is the original Type-I Privacy game for $\mathcal{A}$ as follows.

Step 1. $\mathcal{W} \leftarrow \emptyset$, $(prm, mk) \leftarrow \mathsf{IB.Setup}(1^\kappa)$.

Step 2. $(s^{(0)}, s^{(1)}, t^*, st) \leftarrow \mathcal{A}(\mathsf{chal}, prm, mk)$.

Step 3. $b \xleftarrow{\$} \{0,1\}$, $(K, c_{t^*}) \leftarrow \mathsf{IB.Encaps}(prm, t^*)$,
$C \leftarrow \mathsf{E}(K, s^{(b)})$,
$(\tilde{u}_1, \ldots, \tilde{u}_n) \leftarrow \mathsf{IDA.Share}(1^\kappa, \Gamma, C)$,
$(\hat{u}_1, \ldots, \hat{u}_n) \leftarrow \mathsf{SS.Share}(1^\kappa, \Gamma, c_{t^*})$,
$u_i^{(t^*)} := (\tilde{u}_i, \hat{u}_i)$ $(1 \leq i \leq n)$.

Step 4. $b' \leftarrow \mathcal{A}^{Corrupt(\cdot)}(\mathsf{guess}, st)$.

$Corrupt(\cdot)$ takes P_i as input, and computes $\mathcal{W} \leftarrow \mathcal{W} \cup \{P_i\}$ and outputs $u_i^{(t^*)} = (\tilde{u}_i, \hat{u}_i)$.

Game$_1$: This is the same as Game$_0$ except for the following modification: The underlying session key $\tilde{K}$ of $\tilde{c}_{t^*}$ is independent from K, which is used for encrypting the challenge secret $s^{(b)}$. Namely, Game$_1$ is defined by modifying Step 3 in Game$_0$ as follows.

$b \xleftarrow{\$} \{0,1\}$, $(K, c_{t^*}) \leftarrow \mathsf{IB.Encaps}(prm, t^*)$,
$C \leftarrow \mathsf{E}(K, s^{(b)})$,
$(\tilde{K}, \tilde{c}_{t^*}) \leftarrow \mathsf{IB.Encaps}(prm, t^*)$,
$(\tilde{u}_1, \ldots, \tilde{u}_n) \leftarrow \mathsf{IDA.Share}(1^\kappa, \Gamma, C)$,
$(\hat{u}_1, \ldots, \hat{u}_n) \leftarrow \mathsf{SS.Share}(1^\kappa, \Gamma, \tilde{c}_{t^*})$,
$u_i^{(t^*)} := (\tilde{u}_i, \hat{u}_i)$ $(1 \leq i \leq n)$.

Let X_0 and X_1 be events that $b' = b$ in Game$_0$ and Game$_1$, respectively. Then, we have $Adv^{\mathsf{Type\text{-}I}}_{\Pi_{\mathrm{TCSS}},\Gamma,A}(\kappa) = |\Pr[X_0] - \frac{1}{2}| \leq |\Pr[X_0] - \Pr[X_1]| + |\Pr[X_1] - \frac{1}{2}|$.

We construct a PPT adversary $\mathcal{B}$ against Privacy of Π_{SS} by using an adversary $\mathcal{A}$ against Type-I Privacy as follows.

$\mathcal{B}$ computes $(mpk, msk) := (prm, mk) \leftarrow \mathsf{IB.Setup}(1^\kappa)$, and $\mathcal{B}$ sends (mpk, msk) to $\mathcal{A}$. $\mathcal{A}$ then outputs $(s^{(0)}, s^{(1)}, t^*)$. $\mathcal{B}$ picks the random bit δ. $\mathcal{B}$ computes $(K, c_{t^*}) \leftarrow \mathsf{IB.Encaps}(prm, t^*)$, $C \leftarrow \mathsf{E}(K, s^{(\delta)})$, and $(\tilde{u}_1, \ldots, \tilde{u}_n) \leftarrow \mathsf{IDA.Share}(1^\kappa, \Gamma, C)$. Then, B runs $(\tilde{K}, \tilde{c}_{t^*}) \leftarrow \mathsf{IB.Encaps}(prm, t^*)$, and sends

$(c_{t^*}^{(0)}, c_{t^*}^{(1)})$ to the challenger of Privacy. When $\mathcal{A}$ makes a corrupt query P_i, $\mathcal{B}$ issues the query P_i to the corrupt oracle, and obtains $\hat{u}_i$ which is generated as follows: $(\hat{u}_1, \ldots, \hat{u}_n) \leftarrow \mathsf{SS.Share}(1^\kappa, \Gamma, c_{t^*}^{(b)})$, where b is the random bit in the Privacy game. $\mathcal{B}$ then returns $(\tilde{u}_i, \hat{u}_i)$ to $\mathcal{A}$. Finally, $\mathcal{A}$ outputs a guessing bit δ', and $\mathcal{B}$ outputs $b' = 0$ if $\delta = \delta'$, or $b' = 1$ otherwise.

In the case of $b = 0$, $\mathcal{B}$ can perfectly simulate Game_0. Otherwise, $\mathcal{B}$ can perfectly simulate Game_1. Hence, we can evaluate the difference between X_0 and X_1 by using $\mathcal{B}$. Thus, we have $|\Pr[X_0] - \Pr[X_1]| = 2 \cdot Adv^{\mathsf{Privacy}}_{\Pi_{\mathrm{ss}},\Gamma,\mathcal{B}}(\kappa) = 0$.

Finally, we construct a PPT adversary $\mathcal{B}$ against FTG-CPA of Π_{DEM} by using an adversary $\mathcal{A}$ against Type-I Privacy as follows. $\mathcal{B}$ computes $(mpk, msk) := (prm, mk) \leftarrow \mathsf{IB.Setup}(1^\kappa)$, and $\mathcal{B}$ sends (mpk, msk) to $\mathcal{A}$. After $\mathcal{A}$ outputs $(s^{(0)}, s^{(1)}, t^*)$, $\mathcal{B}$ outputs $(s^{(0)}, s^{(1)})$ as well. In return, $\mathcal{B}$ is given a challenge ciphertext $C^* \leftarrow \mathsf{E}(K, s^{(b)})$, then $\mathcal{B}$ computes $(\tilde{u}_1, \ldots, \tilde{u}_n) \leftarrow \mathsf{IDA.Share}(1^\kappa, \Gamma, C^*)$. $\mathcal{B}$ runs $(\tilde{K}, \tilde{c}_{t^*}) \leftarrow \mathsf{IB.Encaps}(prm, t^*)$, and computes $(\hat{u}_1, \ldots, \hat{u}_n) \leftarrow \mathsf{SS.Share}(1^\kappa, \Gamma, \tilde{c}_{t^*})$. When $\mathcal{A}$ makes a corrupt query P_i, B' returns $(\tilde{u}_i, \hat{u}_i)$ to $\mathcal{A}$. Finally, $\mathcal{A}$ outputs a guessing bit b', and $\mathcal{B}$ outputs the same bit.

Therefore, $\mathcal{B}$ provides perfect simulation for $\mathcal{A}$. It is easy see to that $|\Pr[X_1] - \frac{1}{2}| \leq \epsilon_1$.

From the above discussion, the proof is completed. □

Lemma 3.2. *If Π_{DEM} is ϵ_1-FTG-CPA secure and Π_{KEM} meets (q_{ID}, ϵ_2)-IND-ID-CPA secure, then the resulting (k, n)-TR-CSS scheme Π_{TCSS} in the above construction is (q_t, δ)-Type-II Privacy, where $q_t = q_{ID}$ and $\delta \leq \epsilon_1 + 2\epsilon_2$.*

Proof. Let $\mathcal{A}$ be a Type-II Privacy adversary against the proposed (k, n)-TR-CSS scheme Π_{TCSS}. We define the following two games:

Game_0: This is the original Type-II Privacy game for $\mathcal{A}$ as follows.

Step 1. $\mathcal{W} = \mathcal{P}$, $(prm, mk) \leftarrow \mathsf{IB.Setup}(1^\kappa)$.

Step 2. $(s^{(0)}, s^{(1)}, t^*, st) \leftarrow \mathcal{A}^{Release(mk,\cdot)}(\mathsf{chal}, prm)$.

Step 3. $b \xleftarrow{\$} \{0, 1\}$, $(K, c_{t^*}) \leftarrow \mathsf{IB.Encaps}(prm, t^*)$,
$C \leftarrow \mathsf{E}(K, s^{(b)})$,
$(\tilde{u}_1, \ldots, \tilde{u}_n) \leftarrow \mathsf{IDA.Share}(1^\kappa, \Gamma, C)$,
$(\hat{u}_1, \ldots, \hat{u}_n) \leftarrow \mathsf{SS.Share}(1^\kappa, \Gamma, c_{t^*})$,
$u_i^{(t^*)} := (\tilde{u}_i, \hat{u}_i)$ $(1 \leq i \leq n)$.

Step 4. $b' \leftarrow \mathcal{A}^{Release(mk,\cdot)}(\mathsf{guess}, u_1^{(t^*)}, \ldots, u_n^{(t^*)}, st)$.

$Release(mk, \cdot)$ takes t as input, and outputs $\mathsf{IB.Gen}(prm, mk, t)$. Note that $\mathcal{A}$ cannot issue the target time t^* after the chal stage.

Game_1: This is the same as Game_0 except for the following modification of Step 3 as follows.

$$
\begin{aligned}
&b \xleftarrow{\$} \{0,1\},\ (K, c_{t^*}) \leftarrow \mathsf{IB.Encaps}(prm, t^*),\\
&C \leftarrow \mathsf{E}(K, s^{(b)}),\\
&(\tilde{K}, \tilde{c}_{t^*}) \leftarrow \mathsf{IB.Encaps}(prm, t^*),\\
&(\tilde{u}_1, \ldots, \tilde{u}_n) \leftarrow \mathsf{IDA.Share}(1^\kappa, \Gamma, C),\\
&(\hat{u}_1, \ldots, \hat{u}_n) \leftarrow \mathsf{SS.Share}(1^\kappa, \Gamma, \tilde{c}_{t^*}),\\
&u_i^{(t^*)} := (\tilde{u}_i, \hat{u}_i)\ (1 \le i \le n).
\end{aligned}
$$

Let X_0 and X_1 be events that $b' = b$ in Game_0 and Game_1, respectively. Then, we have $Adv^{\mathsf{Type\text{-}II}}_{\Pi_{\mathrm{TCSS}},\Gamma,A}(\kappa) = |\Pr[X_0] - \frac{1}{2}| \le |\Pr[X_0] - \Pr[X_1]| + |\Pr[X_1] - \frac{1}{2}|$.

We construct a PPT adversary $\mathcal{B}$ against the IND-ID-CPA security of Π_{KEM} by using an adversary $\mathcal{A}$ against Type-II Privacy as follows.

$\mathcal{B}$ is given $mpk := prm$, where $(prm, mk) \leftarrow \mathsf{IB.Setup}(1^\kappa)$, and $\mathcal{B}$ transfers mpk to $\mathcal{A}$. When $\mathcal{A}$ makes a release query t, $\mathcal{B}$ issues the query t to the extract oracle, and obtains sk_t and returns it to $\mathcal{A}$. At some point, $\mathcal{A}$ outputs $(s^{(0)}, s^{(1)}, t^*)$. $\mathcal{B}$ picks the random bit δ. Then, $\mathcal{B}$ outputs t^*. In return, $\mathcal{B}$ is given a pair of a session key and a challenge ciphertext (K_b, c_{t^*}), where b is a random bit in the IND-ID-CPA game, $K_0 \xleftarrow{\$} \mathcal{K}_{\mathrm{KEM}}$, and $(K_1, c_{t^*}) \leftarrow \mathsf{IB.Encaps}(prm, ID)$. $\mathcal{B}$ then computes $C \leftarrow \mathsf{E}(K_b, s^{(\delta)})$, and $(\tilde{u}_1, \ldots, \tilde{u}_n) \leftarrow \mathsf{IDA.Share}(1^\kappa, \Gamma, C)$. $\mathcal{B}$ computes $(\hat{u}_1, \ldots, \hat{u}_n) \leftarrow \mathsf{SS.Share}(1^\kappa, \Gamma, c_{t^*})$. $\mathcal{B}$ finally returns $(u_1^{(t^*)}, \ldots, u_n^{(t^*)})$, where $u_i^{(t^*)} := (\tilde{u}_i, \hat{u}_i)$, to $\mathcal{A}$. $\mathcal{A}$ may continue to make release queries, and these are answered by $\mathcal{B}$ as before. Finally, $\mathcal{A}$ outputs a guessing bit δ', and $\mathcal{B}$ outputs $b' = 0$ if $\delta = \delta'$, or $b' = 1$ otherwise.

In the case of $b = 1$, $\mathcal{B}$ can perfectly simulate Game_0. Otherwise, $\mathcal{B}$ can perfectly simulate Game_1. Hence, we can evaluate the difference between X_0 and X_1 by using $\mathcal{B}$. Thus, we have $|\Pr[X_0] - \Pr[X_1]| \le 2 \cdot Adv^{\mathsf{IND\text{-}ID\text{-}CPA}}_{\Pi_{\mathrm{KEM}},\mathcal{B}}(\kappa)$.

Finally, we can construct a PPT adversary $\mathcal{B}$ against the FTG-CPA security of Π_{DEM} by using an adversary $\mathcal{A}$ against Type-II Privacy in a similar way to the proof of Lemma 3.1. Hence, we have $|\Pr[X_1] - \frac{1}{2}| \le \epsilon_1$.

From the above discussion, the proof is completed. □

Proof of Theorem 3.1. Now, the proof is completed. □

We evaluate the sizes of shares and time-signals in the above (k, n)-TR-CSS scheme.

Proposition 3.1. *The sizes of shares and time-signals required in our (k, n)-TR-CSS scheme are given by*

$$|u_i^{(t)}| = \frac{\lambda}{k} + |K| + COH_{\mathrm{KEM}}(\kappa) + COH_{\mathrm{DEM}}(\kappa),$$

$$\textit{and } |ts^{(t)}| = |sk_t|,$$

where $COH_{\mathrm{KEM}}(\kappa)$ *and* $COH_{\mathrm{DEM}}(\kappa)$ *are ciphertext-overhead in* Π_{KEM} *and* Π_{DEM}*, respectively. Here, ciphertext-overhead means ciphertext-length minus plaintext-length, which depends on a security parameter* κ *in general. In particular, under the assumption of Remark 2.1,* $|u_i^{(t)}| = \frac{\lambda}{k} + |K| + COH_{\mathrm{KEM}}(\kappa)$.

Note that the share size in our construction is only $\mathrm{COH}_{\mathrm{KEM}}(\kappa)$-bits longer than that in Krawczyk's CSS scheme. It means that we successfully added the timed-release functionality to a CSS scheme with only the underlying IB-KEM's ciphertext-overhead. In addition, in our construction, this ciphertext-overhead depends on only the security parameter κ, and not dependent on the secret size, which is advantage in our construction. Moreover, the size of time-signals is the same as that of secret keys, and not dependent on other ones such as size of secrets λ, n, or k. Namely, the resulting TR-CSS is scalable.

Although we have assumed that $\mathrm{COH}_{\mathrm{DEM}}(\kappa) = 0$ as in [86] (see Remark 2.1), we cannot assume $\mathrm{COH}_{\mathrm{KEM}}(\kappa) = 0$ since we do not know an IB-KEM (and an IBE scheme) with this property. Therefore, it is plausible and natural to assume $\mathrm{COH}_{\mathrm{KEM}}(\kappa) > 0$ and improve the above construction, which is the topic in the next section, by focusing on a concrete IND-ID-CPA secure IB-KEM based on currently known efficient (but IND-ID-CCA secure) IB-KEM [84, 85].

3.3.4 More Efficient Construction of (k, n)-TR-CSS

Our generic construction shown in the previous section is very simple and can be regarded as a natural extension of Krawczyk's CSS scheme. In this section, we give a more efficient construction of a (k, n)-TR-CSS scheme based on our generic construction. To reduce the share size, we give a specific construction to the underlying IB-KEM in this construction, though this construction is not fully generic construction. Namely, we want to generate not shares of an *entire* ciphertext of the IB-KEM, but shares of *part of* the ciphertext. To do so, we have to give a specific IB-KEM, particularly, as efficient as possible in terms of the ciphertext size.

A roadmap of this construction is as follows. We first describe a specific IND-ID-CPA secure IB-KEM based on [84, 85] since there are no explicit constructions of IND-ID-CPA secure IB-KEMs. Consequently, we realize an IB-KEM in which the ciphertext consists of only two components. Then, we apply this construction to the underlying IB-KEM in our generic construction shown in the previous section, and we generate shares of only one component of the ciphertext by using a (k, n)-PSS scheme since it is enough to prove its security. Another component is split by using a (k, n)-IDA to reduce the share size.

We construct an efficient IND-ID-CPA secure IB-KEM Π_{KEM} in terms of the ciphertext size as follows. We assume that each identity $ID := (id_1, id_2, \ldots, id_\ell) \in \{0,1\}^\ell$ is an ℓ bit string, where id_i $(1 \leq i \leq \ell)$ is i-th bit of ID.

– $(prm, mk) \leftarrow \mathsf{IB.Setup}(1^\kappa)$:

1. Choose a random generator $g \in \mathbb{G}$ and $\alpha, u, h_0, h_1, \ldots, h_\ell \xleftarrow{\$} \mathbb{G}$.
2. Compute $z := e(g, \alpha)$.
3. Output $prm := (g, u, h_0, h_1, \ldots, h_\ell, z)$ and $mk := \alpha$.

– $sk_{ID} \leftarrow \mathsf{IB.Gen}(prm, mk, ID)$: For $ID := (id_1, id_2, \ldots, id_\ell) \in \{0,1\}^\ell$,

1. Choose $\gamma \xleftarrow{\$} \mathbb{Z}_p$.
2. Output $sk_{ID} := (sk_{ID}^{(1)}, sk_{ID}^{(2)}) = (\alpha(h_0 \prod_{i=1}^{\ell} h_i^{id_i})^\gamma, g^\gamma)$.

– $(K, c_{ID}) \leftarrow \mathsf{IB.Encaps}(prm, ID)$: For $ID := (id_1, id_2, \ldots, id_\ell) \in \{0,1\}^\ell$,

1. Choose $r \xleftarrow{\$} \mathbb{Z}_p$.
2. Compute $c_{ID}^{(1)} := g^r$ and $c_{ID}^{(2)} := (h_0 \prod_{i=1}^{\ell} h_i^{id_i})^r$.
3. Output $K := z^r$ and $c_{ID} := (c_{ID}^{(1)}, c_{ID}^{(2)})$.

– $K \leftarrow \mathsf{IB.Decaps}(prm, sk_{ID}, c_{ID})$: For $sk_{ID} = (sk_{ID}^{(1)}, sk_{ID}^{(2)})$ and $c_{ID} = (c_{ID}^{(1)}, c_{ID}^{(2)})$,

1. Output $K = e(sk_{ID}^{(2)}, c_{ID}^{(2)}) / e(sk_{ID}^{(1)}, c_{ID}^{(1)})$.

It is easy to see the above construction satisfies the correctness property:

$$\begin{aligned}
\frac{e(sk_{ID}^{(1)}, c_{ID}^{(1)})}{e(sk_{ID}^{(2)}, c_{ID}^{(2)})} &= \frac{e(\alpha(h_0 \prod_{i=1}^{\ell} h_i^{id_i})^\gamma, g^r)}{e(g^\gamma, (h_0 \prod_{i=1}^{\ell} h_i^{id_i})^r)} \\
&= \frac{e(\alpha, g^r) e((h_0 \prod_{i=1}^{\ell} h_i^{id_i})^\gamma, g^r)}{e(g^\gamma, (h_0 \prod_{i=1}^{\ell} h_i^{id_i})^r)} \\
&= \frac{e(\alpha, g)^r e((h_0 \prod_{i=1}^{\ell} h_i^{id_i}), g)^{r\gamma}}{e(g, (h_0 \prod_{i=1}^{\ell} h_i^{id_i}))^{r\gamma}} \\
&= e(\alpha, g)^r = K.
\end{aligned}$$

We can prove security of the above IB-KEM under the DBDH assumption in a similar way to the proof of [85, Thm. 1].

Theorem 3.2. *Let δ be an upper bound of the advantage of an adversary in the DBDH problem, and let q_{ID} ($< \frac{p}{2(n+1)}$) be an upper bound on the number of key-derivation queries made by an adversary in the IND-ID-CPA game of the above IB-KEM Π_{KEM}. If the DBDH assumption holds, then the resulting IB-KEM Π_{KEM} in the above construction is (q_{ID}, ϵ)-IND-ID-CPA secure, where $\epsilon \leq 8(\ell+1) q_{ID} \delta$.*

Next, we propose an efficient construction of a (k, n)-TR-CSS scheme by using the above IB-KEM Π_{KEM}. We consider a set of identities $\mathcal{ID}$ in the underlying IB-KEM as a set of time $\mathcal{T}$ as in our generic construction described in the previous section. Namely, each time $t := (t_1, t_2, \ldots, t_\ell) \in \{0,1\}^\ell$ is an ℓ bit string, where t_i $(1 \leq i \leq \ell)$ is i-th bit of t. Π_{SS}, Π_{IDA} and Π_{DEM} are the same as those in our generic construction shown in the previous section. Although we write output of every algorithm in the underlying IB-KEM in a generic manner, each algorithm is executed as above.

- $(mpk, msk) \leftarrow \mathsf{Setup}(1^\kappa)$:
  1. Output $(mpk, msk) := (prm, mk) \leftarrow \mathsf{IB.Setup}(1^\kappa)$.
- $ts^{(t)} \leftarrow \mathsf{Ext}(msk, t)$:
  1. $ts^{(t)} := (sk_t^{(1)}, sk_t^{(2)}) \leftarrow \mathsf{IB.Gen}(prm, msk, t)$.
- $(u_1^{(t)}, \ldots, u_n^{(t)}) \leftarrow \mathsf{Share}(\Gamma, mpk, s, t)$:
  1. $(K, (c_t^{(1)}, c_t^{(2)})) \leftarrow \mathsf{IB.Encaps}(prm, t)$.
  2. $C \leftarrow \mathsf{E}(K, s)$.
  3. $(\hat{u}_1^{(1)}, \ldots, \hat{u}_n^{(1)}) \leftarrow \mathsf{IDA.Share}(1^\kappa, \Gamma, c_t^{(1)})$
  4. $(\hat{u}_1^{(2)}, \ldots, \hat{u}_n^{(2)}) \leftarrow \mathsf{SS.Share}(1^\kappa, \Gamma, c_t^{(2)})$.
  5. $(\tilde{u}_1, \ldots, \tilde{u}_n) \leftarrow \mathsf{IDA.Share}(1^\kappa, \Gamma, C)$.
  6. Output $u_i^{(t)} := (\tilde{u}_i, \hat{u}_i^{(1)}, \hat{u}_i^{(2)})$ $(1 \leq i \leq n)$.
- $s \leftarrow \mathsf{Recon}(u_{\mathcal{Q}}, ts^{(t)})$:]
  1. $C \leftarrow \mathsf{IDA.Recon}(1^\kappa, \tilde{u}_{\mathcal{Q}})$.
  2. $c_t^{(1)} \leftarrow \mathsf{IDA.Recon}(1^\kappa, \hat{u}_{\mathcal{Q}}^{(1)})$.
  3. $c_t^{(2)} \leftarrow \mathsf{SS.Recon}(1^\kappa, \hat{u}_{\mathcal{Q}}^{(2)})$.
  4. After receiving a time-signal $ts^{(t)} = (sk_t^{(1)}, sk_t^{(2)})$ at the specified time t, $K \leftarrow \mathsf{IB.Decaps}(prm, ts^{(t)}, c_t)$.
  5. $s = \mathsf{D}(K, C)$.

We obtain the following theorem and give the proof sketch of it since we can prove it in a similar way to the proof of Theorem 3.1.

Theorem 3.3. *Let q be the number of queries and ϵ be an upper bound of the advantage of an adversary in IND-ID-CPA game of the IB-KEM Π_{KEM}. If Π_{DEM} is ϵ'-FTG-CPA secure and Π_{SS} is a (k, n)-PSS scheme, then the resulting (k, n)-TR-CSS scheme Π_{TCSS} in the above construction is a $(q_t, \delta_1, \delta_2)$-$(k, n)$-TR-CSS scheme, where $q_t = q$, $\delta_1 \leq \epsilon'$ and $\delta_2 \leq \epsilon' + 2\epsilon$.*

Proof Sketch. The IB-KEM Π_{KEM} meets IND-ID-CPA security. Hence, the difference between the generic construction shown in the previous section and the above construction is that we generate shares of the *entire* ciphertext of K or *part of* the ciphertext of K by using a (k, n)-PSS scheme. Therefore, we do not have to reconsider Type-II Privacy of this construction, since it is the same situation as our generic construction in the Type-II Privacy game setting (i.e., the adversary can reconstruct the ciphertext of K in both cases). Therefore, we can obtain $\delta_2 \leq \epsilon' + 2\epsilon$ as in the proof of Lemma 3.2. Moreover, in the Type-I Privacy game setting, even if the adversary obtains $c_t^{(1)}$, the adversary cannot get any information on K since the adversary cannot distinguish $c_t^{(1)}$ and a random element in $\mathbb{G}$. Actually, a form of $c_t^{(1)}$ is g^r. Therefore, we can obtain $\delta_1 \leq \epsilon'$ as in the proof of Lemma 3.1. □

We evaluate the sizes of shares and time-signals in the above (k, n)-TR-CSS scheme.

Proposition 3.2. *The sizes of shares and time-signals required in the above (k, n)-TR-CSS scheme are given by*

$$|u_i^{(t)}| = \frac{\lambda}{k} + |\mathbb{G}| + \frac{|\mathbb{G}|}{k} + COH_{\mathrm{DEM}}(\kappa), \text{ and } |ts^{(t)}| = 2|\mathbb{G}|,$$

where $|\mathbb{G}|$ denotes the length of the element of $\mathbb{G}$. In particular, under the assumption of Remark 2.1, we have $|u_i^{(t)}| = \frac{\lambda}{k} + |\mathbb{G}| + |\mathbb{G}|/k$.

Since $|\mathbb{G}| = |\mathbb{G}_T| = |K|$ in the above construction, this proposition says that we can achieve the share size which is close to that of Krawczyk's CSS scheme when k is sufficiently large. Moreover, to the best of our knowledge, there is no IB-KEM (and IBE scheme) with non-redundant ciphertexts so far. Hence, in the sense of the share-overhead compared with Krawczyk's CSS scheme, we can say that this construction is *almost optimal*.

3.3.5 Discussion

We discuss several points of TR-CSS to clarify our contributions of this chapter. Specifically, we discuss adequacy, needs and extensions of TR-CSS.

Our scheme is a natural extension of Krawczyk's CSS scheme. In the case that $\mathcal{T} = \emptyset$, mpk is a security parameter κ, and msk is an empty string (i.e., if we remove the concept of time from TR-CSS), the model and security definition of (k, n)-TR-CSS are the same as those of traditional (k, n)-SS (see Remark 3.1). Namely, our model of TR-CSS includes the model of traditional secret sharing schemes.

In terms of a generic construction, Krawczyk's CSS scheme is the special case of our TR-CSS scheme. This is because our construction can be regarded

as Krawczyk's CSS scheme in the above situation: Since there is no concept of time (i.e. $\mathcal{T} = \mathcal{ID} = \emptyset$), it need not execute an encapsulation algorithm and a decapsulation algorithm of the underlying IB-KEM, and it is natural and reasonable to regard c_t as K. Namely, our construction can be considered as the extension of Krawczyk's CSS scheme.

Alternative construction and its limitation. One may consider a construction starting from a TR-PKE scheme. However, a TR-PKE scheme is insufficient for efficiently constructing a TR-CSS scheme in the sense of the achievable share size. Specifically, we construct a (k, n)-TR-CSS scheme by using a TR-PKE scheme as follows. Let Π_{SS}=(SS.Share, SS.Recon) be a (k, n)-SS scheme, Π_{IDA}=(IDA.Share, IDA.Recon) be a (k, n)-IDA, Π_{TPKE}=(TR.Init, TRKeyGen, TR.Release, TR.Enc, TR.Dec) be a TR-PKE scheme, and let Π_{DEM}=(E, D) be a DEM. Suppose that $\mathcal{T} \subset \hat{\mathcal{T}}$. Then, a (k, n)-TR-CSS scheme Π_{TCSS}=(Setup, Ext, Share, Recon) is constructed as follows.

- $(mpk, msk) \leftarrow \mathsf{Setup}(1^\kappa)$:
  1. Output $(mpk, msk) := (params, tsk) \leftarrow \mathsf{TR.Init}(1^\kappa)$.
- $ts^{(t)} \leftarrow \mathsf{Ext}(msk, t)$:
  1. Output $ts^{(t)} \leftarrow \mathsf{TR.Release}(tsk, t)$.
- $(u_1^{(t)}, \ldots, u_n^{(t)}) \leftarrow \mathsf{Share}(\Gamma, mpk, s, t)$:
  1. Choose $K \xleftarrow{\$} \mathcal{K}_{\mathsf{DEM}}$.
  2. $C \leftarrow \mathsf{E}(K, s)$.
  3. $(upk, usk) \leftarrow \mathsf{TRKeyGen}(params)$.
  4. $ct_t \leftarrow \mathsf{TR.Enc}(params, upk, K, t)$.
  5. $(\tilde{u}_1, \ldots, \tilde{u}_n) \leftarrow \mathsf{IDA.Share}(1^\kappa, \Gamma, C)$.
  6. $(\hat{u}_1, \ldots, \hat{u}_n) \leftarrow \mathsf{SS.Share}(1^\kappa, \Gamma, ct_t)$.
  7. $(\acute{u}_1, \ldots, \acute{u}_n) \leftarrow \mathsf{SS.Share}(1^\kappa, \Gamma, usk)$.
  8. Output $u_i^{(t)} := (\tilde{u}_i, \hat{u}_i, \acute{u}_i)$ $(1 \leq i \leq n)$.
- $s \leftarrow \mathsf{Recon}(\hat{u}_{\mathcal{Q}}, ts^{(t)})$:
  1. $ct_t \leftarrow \mathsf{SS.Recon}(1^\kappa, \hat{u}_{\mathcal{Q}})$.
  2. $usk \leftarrow \mathsf{SS.Recon}(1^\kappa, \acute{u}_{\mathcal{Q}})$.
  3. $C \leftarrow \mathsf{IDA.Recon}(1^\kappa, \tilde{u}_{\mathcal{Q}})$.
  4. $K \leftarrow \mathsf{TR.Dec}(params, usk, ts^{(t)}, ct_t, t)$.
  5. Output $s \leftarrow \mathsf{D}(K, C)$.

The resulting (k, n)-TR-CSS scheme in the above construction is secure, if a given (k, n)-SS scheme is a (k, n)-PSS scheme, a given TR-PKE scheme meets IND-CTPA, and a given DEM meets FTG-CPA, as follows.

Proposition 3.3. *If Π_{DEM} is ϵ_1-FTG-CPA secure, Π_{SS} is (k, n)-PSS, and Π_{TPKE} is (q_{ts}, ϵ_2)-IND-CTPA secure, then the resulting (k, n)-TR-CSS scheme Π_{TCSS} in the above construction is a $(q_t, \delta_1, \delta_2)$-$(k, n)$-TR-CSS scheme, where $q_t = q_{ts}$, $\delta_1 \leq \epsilon_1$ and $\delta_2 \leq \epsilon_1 + 2\epsilon_2$.*

We omit the proof of Proposition 3.3 since we can prove it in a similar way to the proof of Theorem 3.1.

We show that the share size of the alternative construction as follows.

Proposition 3.4. *The share size required in a (k, n)-TR-CSS scheme is*

$$|u_i^{(t)}| = \frac{\lambda}{k} + |K| + COH_{\mathrm{TRE}}(\kappa) + COH_{\mathrm{DEM}}(\kappa) + |usk|,$$

where $COH_{\mathrm{TRE}}(\kappa)$ and $COH_{\mathrm{DEM}}(\kappa)$ are ciphertext-overhead used in the underlying TR-PKE scheme and the underlying DEM, respectively, and $|usk|$ is the length of secret keys of the TR-PKE scheme. Even under the assumption of Remark 2.1, $|u_i^{(t)}| = \frac{\lambda}{k} + |K| + COH_{\mathrm{TRE}}(\kappa) + |usk|$.

As seen above, a secret key of the TR-PKE scheme must be sent as a part of the share in this construction, since a dealer generates a pair of public and secret keys. Note that our aim is *to realize a TR-CSS scheme with the smallest possible share size*. Incidentally, KEMs in the timed-release security setting have not been proposed so far.

Moreover, one may consider another approach from a TR-PKE scheme: each participants generates a pair of public and secret keys, and a dealer encrypts each share by the corresponding public key of the TR-PKE scheme. This approach can avoid sending a secret key of the TR-PKE scheme as a part of the share, however, this is a stronger situation than the traditional secret sharing schemes setting. That is, each participant must generate a pair of public and secret keys in the approach (in addition, all public keys must be certified by a public key infrastructure).

Robust TR-CSS schemes. (k, n)-TR-CSS schemes can easily achieve the *robustness*, which means the property that participants can definitely reconstruct the correct secret from any submitted shares including at least k correct shares, otherwise the reconstruct algorithm always outputs $\perp$ (i.e., it does not output any value which is different from the original secret). There is a simple method to transform any CSS scheme into a robust CSS scheme by using DS schemes (without taking into account efficiency). Specifically, the dealer generates a signature for each share by using his secret key, and then participants reconstruct the secret from shares whose signatures are correct (see [54, 122] for similar discussions). Hence, we can achieve the *robust* TR-CSS scheme in the same way.

TR-CSS schemes with a general access structure Γ. A concept of a *general access structure* $\Gamma := (\mathscr{Q}, \mathscr{F})$ is considered to generalize (k, n)-threshold schemes [11, 78]. A qualified set $\mathcal{Q} \in \mathscr{Q}$ is the set of participants that can reconstruct the secret, whereas any participants of a forbidden set $\mathcal{F} \in \mathscr{F}$ cannot get any information on the secret, where $\mathscr{Q} \subset 2^{\mathcal{P}}$ and $\mathscr{F} \subset 2^{\mathcal{P}}$ are families of qualified sets and forbidden sets, respectively. And we assume that $\mathscr{Q} \cup \mathscr{F} = 2^{\mathcal{P}}$ and $\mathscr{Q} \cap \mathscr{F} = \emptyset$.

In [6], Béguin and Cresti proposed an IDA with a general access structure Γ (called a *general* IDA) and then constructed a CSS scheme with a general access structure Γ (called a *general* CSS scheme) by using the general IDA. By applying their idea, we can easily realize a general TR-CSS scheme by replacing a (k, n)-PSS scheme and a (k, n)-IDA with the general PSS scheme and the general IDA, respectively, in our constructions.

3.4 Multiple Encryption and Threshold Encryption with Timed-Release Functionality

In this section, we show that a TR-CSS scheme can provide a threshold encryption scheme [50] with the timed-release functionality. Dodis and Katz [54] showed that a multiple encryption scheme can be transformed to a threshold encryption scheme. Therefore, we first consider multiple encryption with timed-release functionality, since multiple encryption has many other applications.

3.4.1 The Model of Multiple Encryption

In multiple encryption (ME), a plaintext is encrypted by using independent secret keys or distinct PKE schemes based on different computational assumptions. An ME scheme provides better security than a PKE scheme, since even if underlying assumptions of some component PKE schemes are broken or some of secret keys are leaked, the confidentiality can still be maintained by the remaining PKE schemes (for details, see [54, 149, 103]).

We adapt the model of multiple encryption shown by Dodis and Katz [54]. An ME scheme Π_{ME} consists of five-tuple algorithms (M.Gen, M.Enc, Split, M.Dec, M.Combine) defined as follows, where $\mathcal{M}_{\mathrm{ME}}$ is a set of plaintexts determined by a security parameter κ.

- $(EK, \boldsymbol{DK}) \leftarrow \mathsf{M.Gen}(1^{\kappa})$: A probabilistic algorithm for key generation. It takes a security parameter κ as input, and outputs a public key EK and secret keys $\boldsymbol{DK} := (DK_1, \ldots, DK_n)$.

- $C \leftarrow \mathsf{M.Enc}_L(EK, M)$: A probabilistic algorithm for encryption. It takes a public key EK, a label L, and a plaintext $M \in \mathcal{M}_{\mathrm{ME}}$ as input, and then outputs a ciphertext C.

- $\boldsymbol{C} \leftarrow \mathsf{Split}_L(EK, C)$: A deterministic algorithm for splitting a ciphertext. It takes a public key EK, a label L, and a ciphertext C as input, and outputs n ciphertext shares $\boldsymbol{C} := (C_1, \ldots, C_n)$.

- M_i or $\perp \leftarrow \mathsf{M.Dec}(DK_i, C_i)$: A deterministic algorithm for decryption. It takes a secret key DK_i and a ciphertext share C_i as inputs, and outputs a plaintext share M_i or $\perp$.

- M or $\perp \leftarrow \mathsf{M.Combine}(\boldsymbol{M})$: A deterministic algorithm for combining plaintext shares. It takes all plaintext shares $\boldsymbol{M} := (M_1, \ldots, M_n)$ as input, and outputs the plaintext $M \in \mathcal{M}_{\mathrm{ME}}$ or $\perp \notin \mathcal{M}_{\mathrm{ME}}$.

In the above model, we assume that Π_{ME} meets the following *correctness* property: For all $\kappa \in \mathbb{N}$, all labels L, all $(EK, \boldsymbol{DK}) \leftarrow \mathsf{M.Gen}(1^\kappa)$ and all $M \in \mathcal{M}_{\mathrm{ME}}$, it holds that

$$M \leftarrow \mathsf{M.Combine}(\boldsymbol{DEC}(\mathsf{Split}_L(EK, \mathsf{M.Enc}_L(EK, M)))),$$

where $\boldsymbol{DEC}(\boldsymbol{C}) := (\mathsf{M.Dec}(DK_1, C_1), \ldots, \mathsf{M.Dec}(DK_n, C_n))$ for $(C_1, \ldots, C_n) \leftarrow \mathsf{Split}_L(EK, \mathsf{M.Enc}_L(EK, M))$.

We describe notions of indistinguishability against the following four attacks [54]: multiple chosen plaintext attack (IND-MCPA), weak multiple chosen ciphertext attack (IND-wMCCA), multiple chosen ciphertext attack (IND-MCCA), and strong multiple chosen ciphertext attack (IND-sMCCA). Let $\mathcal{A}$ be a PPT adversary, and let X$\in$\{MCPA,wMCCA,MCCA,sMCCA\}. Let $\mathcal{A}$ be a PPT adversary, and $\mathcal{A}$'s advantage against the IND-X security is defined by

$$Adv^{\mathsf{IND\text{-}X}}_{\Pi_{\mathrm{ME}},\mathcal{A}}(\kappa) := \left| \Pr\left[b' = b \;\middle|\; \begin{array}{l} \mathcal{W} \leftarrow \emptyset,\ (EK, \boldsymbol{DK}) \leftarrow \mathsf{M.Gen}(1^\kappa), \\ (M_0^*, M_1^*, L^*, st) \leftarrow \mathcal{A}^{Corrupt(\cdot), \mathcal{O}}(\mathsf{chal}, EK), \\ b \xleftarrow{\$} \{0,1\},\ C^* \leftarrow \mathsf{M.Enc}_{L^*}(EK, M_b), \\ b' \leftarrow \mathcal{A}^{Corrupt(\cdot), \mathcal{O}}(\mathsf{guess}, C^*, st) \end{array} \right] - \frac{1}{2} \right|.$$

Here, we require $|M_0^*| = |M_1^*|$, and st is state information. $Corrupt(\cdot)$ is a *corrupt oracle* which takes an ID i as input, and then $\mathcal{W} \leftarrow \mathcal{W} \cup \{P_i\}$ and returns DK_i. $\mathcal{A}$ can query to $Corrupt(\cdot)$ until $|\mathcal{W}| = d$. In addition, $\mathcal{O}$ is an oracle corresponding to attacks (see Table 3.1). $\mathcal{A}$ is allowed to access the above oracle at most q_c times at any time, however it cannot submit the target ciphertext C^* to $\mathcal{O}$. In addition to this, we need to add two restrictions in the IND-sMCCA game. First, $\mathcal{A}$ cannot submit (i, C_i^*), where $C_i^* \in \boldsymbol{C^*}$ and $(\boldsymbol{C^*}, aux) \leftarrow \mathsf{Split}_{L^*}(EK, C^*)$. Second, we need the weakly collision-resistant property, which means that any PPT adversary $\mathcal{A}$ cannot find C' such that $\mathsf{Split}_L^{(i)}(EK, C) = \mathsf{Split}_L^{(i)}(EK, C')$ and $C' \neq C$, where $\mathsf{Split}_L^{(i)}(EK, C)$ denotes the i-th output of $\mathsf{Split}_L(EK, C)$ (see [54] for the formal definition).

X	MCPA	wMCCA	MCCA	sMCCA
$\mathcal{O}$	*empty*	M.Combine($\boldsymbol{DEC}$(Split$_{(\cdot)}(\cdot)$))	$\boldsymbol{DEC}$(Split$_{(\cdot)}(\cdot)$)	Dec($DK_{(\cdot)}, \cdot$)

Table 3.1: Oracles in the IND-X game.

Definition 3.6 (Security of ME [54]). *For* $^{\exists}\kappa_0 \in \mathbb{N}$ *and* $^{\forall}\kappa \geq \kappa_0$*, a multiple encryption scheme* Π_{ME} *is said to be* (d, q_c, ϵ)*-IND-X secure if there exists a negligible* ϵ *in* κ *such that* $Adv^{\mathsf{IND\text{-}X}}_{\Pi_{\mathrm{ME}},\mathcal{A}}(\kappa) < \epsilon$ *holds for any PPT adversary* $\mathcal{A}$*, where* d *is the maximum number of secret keys that* $\mathcal{A}$ *can obtain and* q_c *is the number of queries that* $\mathcal{A}$ *can issue to the oracle* $\mathcal{O}$ *in the IND-X game.*

3.4.2 Advantages of timed-release functionality from TR-CSS rather than TR-PKE

Dodis and Katz [54] proposed a generic construction of an IND-MCCA secure ME scheme by using n IND-CCA secure PKE schemes, an OTS scheme and a (k, n)-CSS scheme, which is simple and elegant. We want to extend their construction so that the resulting ME scheme has the timed-release functionality. To add the timed-release functionality to ME based on the construction, we can naturally consider the following two approaches: Either replacing the underlying IND-CCA secure PKE scheme with an IND-CCA secure TR-PKE scheme, or replacing the underlying (k, n)-CSS scheme with a (k, n)-TR-CSS scheme.[4] The former seems to be a natural solution, however, this approach has two disadvantages as follows.

- The former needs to replace at least $n - k + 1$ PKE schemes with TR-PKE schemes to guarantee its security. Actually, the most efficient IND-CCA secure TR-PKE scheme is constructed from an IND-CCA secure PKE scheme[5] and an IND-ID-CPA secure IB-KEM [96]. Therefore, the former leads to combining the Dodis–Katz construction with $n - k + 1$ IB-KEMs. One the other hand, the latter only needs to replace a (k, n)-CSS scheme with a (k, n)-TR-CSS scheme. Namely, by using our (k, n)-TR-CSS scheme in Section 3.3.3, the latter only needs to combine the Dodis–Katz construction with only one IB-KEM.

- As mentioned in [54], multiple encryption might be used in different scenarios: All secret keys $\boldsymbol{DK}$ might be co-located in a single place (i.e. a local server of a receiver); or each secret key DK_i might be stored in different locations (i.e. on different servers). To respond to both scenarios, we argue that the latter is better than the former. Specifically,

[4] Our goal is to keep confidentiality until the specified time comes. Therefore, we do not consider replacing OTS schemes.

[5] To be precise, an IND-CCA secure tag-KEM [1] and a DEM. However, we roughly describe here for the easy-to-understand explanation since an IND-CCA secure PKE scheme can be constructed from these primitives.

the former approach can realize only a *partial decryption algorithm with the timed-release functionality*. Namely, when the specified time comes, a receiver gets plaintext shares, and then obtains a plaintext. It means that a receiver must access each server and get plaintext shares *after* the specified time. On the other hand, the latter approach can realize a *combining algorithm with the timed-release functionality*. Namely, a receiver can get plaintext shares before specified time comes, and then, he obtains a plaintext when the specified time comes. It means that a receiver can access each server and get plaintext shares *in advance*. Of course, no information on the plaintext is leaked from plaintext shares before the specified time. When the specified time comes, the receiver can get a plaintext without communicating with servers (i.e., the receiver can get a plaintext with only local computation at that time).

3.4.3 Timed-Release Multiple Encryption

Based on discussion in the previous section, we consider multiple encryption with the *suitable* timed-release functionality, which we call *timed-release multiple encryption* (TR-ME).

The model

A TR-ME scheme Π_{TRME} =(TM.Setup, TM.KeyGen, TM.Ext, TM.Enc, TM.Split, TM.Dec, TM.Combine) defined as follows, where $\mathcal{M}_{\text{TME}}$ is a set of plaintexts determined by a security parameter κ.

- $(Par, MK) \leftarrow \mathsf{TM.Setup}(1^{\kappa})$: A probabilistic algorithm for setup. It takes a security parameter κ as input and outputs a public parameter Par and a master secret key MK.
- $(MEK, \boldsymbol{MDK}) \leftarrow \mathsf{TM.KeyGen}(Par)$: A probabilistic algorithm for key generation. It takes a public parameter Par as input and outputs a public key MEK and secret keys $\boldsymbol{MDK} := (MDK_1, \ldots, MDK_n)$.
- $ts^{(t)} \leftarrow \mathsf{TM.Ext}(MK, t)$: A probabilistic algorithm for generating time-signals. It takes the master secret key MK and time $t \in \mathcal{T}$ as input and outputs a time-signal $ts^{(t)}$ at time t.
- $C^{(t)} \leftarrow \mathsf{TM.Enc}_L(Par, MEK, M, t)$: A probabilistic algorithm for encryption. It takes the public parameter Par, the public key MEK, a label L, a plaintext $M \in \mathcal{M}_{\text{TME}}$, and a specified time t as input, and then outputs a ciphertext $C^{(t)}$ at time t.
- $\boldsymbol{C}^{(t)}$ or $\perp \leftarrow \mathsf{TM.Split}_L(Par, MEK, C^{(t)})$: A deterministic algorithm for splitting a ciphertext. It takes the public parameter Par, the public key MEK, a label L, and a ciphertext $C^{(t)}$ as input, and then outputs n ciphertext shares $\boldsymbol{C}^{(t)} := (C_1^{(t)}, \ldots, C_n^{(t)})$ at time t or $\perp$.

- $M_i^{(t)}$ or $\perp \leftarrow \mathsf{TM.Dec}(MDK_i, C_i^{(t)})$: A deterministic algorithm for decryption. It takes a secret key MDK_i and a ciphertext share $C_i^{(t)}$ as inputs and then outputs a plaintext share $M_i^{(t)}$ or $\perp$.
- M or $\perp \leftarrow TM.Combine(\boldsymbol{M}^{(t)}, ts^{(t)})$: A deterministic algorithm for combining plaintext shares. It takes all plaintext shares $\boldsymbol{M}^{(t)} := (M_1^{(t)}, \ldots, M_n^{(t)})$ and the time-signal $ts^{(t)}$ at the specified time t as input, and then outputs the plaintext M or $\perp$.

In the above model, we assume that Π_{TRME} meets the following *correctness* property: For all $\kappa \in \mathbb{N}$, all labels L, all $(Par, MK) \leftarrow \mathsf{TM.Setup}(1^\kappa)$, all $(MEK, \boldsymbol{MDK}) \leftarrow \mathsf{TM.KeyGen}(Par, \Gamma)$, all t, all $ts^{(t)} \leftarrow \mathsf{TM.Ext}(MK, t)$, and all $M \in \mathcal{M}_{\mathrm{TME}}$, it holds

$$M \leftarrow \mathsf{TM.Combine}(\boldsymbol{DEC}(\mathsf{TM.Split}_L(Par, MEK, \mathsf{TM.Enc}_L(Par, MEK, M, t))), ts^{(t)}),$$

where $\boldsymbol{DEC}(\boldsymbol{C}^{(t)}) := (\mathsf{TM.Dec}(MDK_1, C_1^{(t)}), \ldots, \mathsf{TM.Dec}(MDK_n, C_n^{(t)}))$ for $(C_1^{(t)}, \ldots, C_n^{(t)}) \leftarrow \mathsf{TM.Split}_L(Par, MEK, \mathsf{TM.Enc}_L(Par, MEK, M, t))), ts^{(t)}))$.

Security

As in TR-PKE [33], we need to consider security against a curious time-server (time-server security) and security against a receiver (insider security). Therefore, we define the notion of indistinguishability against multiple chosen ciphertext attack (IND-MCCA) for time-server security and the notion of indistinguishability against multiple chosen time and plaintext attack (IND-MCTPA) for insider security[6]. The former notion means that even a time-server cannot obtain any information on the underlying plaintext from the target ciphertext, and the latter notion means even a legitimate receiver cannot obtain any information on the underlying plaintext without the time-signal at the specified time.

We formalize the former notion, IND-MCCA security. Let $\mathcal{A}$ be a PPT adversary, and $\mathcal{A}$'s advantage against the IND-MCCA security is defined by

$$Adv^{\mathsf{IND\text{-}MCCA}}_{\Pi_{\mathrm{TRME}},\mathcal{A}}(\kappa) := \left| \Pr\left[b' = b \;\middle|\; \begin{array}{l} \mathcal{W} \leftarrow \emptyset,\ (Par, MS) \leftarrow \mathsf{TM.Setup}(1^\kappa), \\ (MEK, \boldsymbol{MDK}) \leftarrow \mathsf{TM.KeyGen}(Par), \\ (M_0^*, M_1^*, L^*, t^*, st) \\ \qquad \leftarrow \mathcal{A}^{Corrupt(\cdot), \mathcal{O}}(\mathsf{chal}, MEK, Par, MK), \\ b \xleftarrow{\$} \{0,1\},\ C^{(t^*)} \leftarrow \mathsf{TM.Enc}_{L^*}(Par, MEK, M_b, t^*), \\ b' \leftarrow \mathcal{A}^{Corrupt(\cdot), \mathcal{O}}(\mathsf{guess}, C^{(t^*)}, st) \end{array} \right] - \frac{1}{2} \right|.$$

[6] As mentioned in [49], it is only necessary to consider a notion of "CPA" for insider security, since an adversary has all secret keys and hence he can decrypt ciphertexts by access to a time-signals generation oracle.

Here, we require $|M_0^*| = |M_1^*|$, and st is state information. *Corrupt*$(\cdot)$ is a *corrupt oracle* which takes an ID i as input, and then $\mathcal{W} \leftarrow \mathcal{W} \cup \{i\}$ and returns MDK_i. $\mathcal{A}$ can query to *Corrupt*$(\cdot)$ until $|\mathcal{W}| = d$. In addition, $\mathcal{O}$ is a decryption oracle which takes $(C^{(t)}, L)$ as input and outputs $\boldsymbol{DEC}(\mathsf{TM.Split}_{(\cdot)}(Par, MEK, \cdot))$. $\mathcal{A}$ is allowed to access the decryption oracle at most q_c times at any time, however it cannot submit the target ciphertext and label $(C^{(t^*)}, L^*)$ to the oracle.

Definition 3.7 (IND-MCCA). *For* $^{\exists}\kappa_0 \in \mathbb{N}$ *and* $^{\forall}\kappa \geq \kappa_0$*, a TR-ME scheme* Π_{TRME} *is said to be* (d, q_c, ϵ)*-IND-MCCA secure if there exists a negligible* ϵ *in* κ *such that* $Adv^{\mathsf{IND\text{-}MCCA}}_{\Pi_{\mathrm{TRME}},\mathcal{A}}(\kappa) < \epsilon$ *holds for any PPT adversary* $\mathcal{A}$*, where* d *is the maximum number of secret keys that* $\mathcal{A}$ *can obtain and* q_c *is the number of queries that* $\mathcal{A}$ *can issue to the decryption oracle in the IND-MCCA game.*

Next, we formalize the latter notion. Let $\mathcal{A}$ be a PPT adversary, and $\mathcal{A}$'s advantage against the IND-MCTPA security is defined by

$$Adv^{\mathsf{IND\text{-}MCTPA}}_{\Pi_{\mathrm{TRME}},\mathcal{A}}(\kappa) := \left| \Pr\left[b' = b \;\middle|\; \begin{array}{l} (Par, MK) \leftarrow \mathsf{TM.Setup}(1^\kappa), \\ (MEK, \boldsymbol{MDK}) \leftarrow \mathsf{TM.KeyGen}(Par), \\ (M_0^*, M_1^*, L^*, t^*, st) \\ \qquad \leftarrow \mathcal{A}^{Release(MK,\cdot)}(\mathsf{chal}, MEK, \boldsymbol{MDK}, Par), \\ b \xleftarrow{\$} \{0,1\},\ C^{(t^*)} \leftarrow \mathsf{TM.Enc}_{L^*}(Par, MEK, M_b, t^*), \\ b' \leftarrow \mathcal{A}^{Release(MK,\cdot)}(\mathsf{guess}, C^{(t^*)}, st) \end{array} \right] - \frac{1}{2} \right|.$$

Here, we require $|M_0^*| = |M_1^*|$, and st is state information. *Release*$(MK, \cdot)$ is a *time-signals generation oracle* which takes time t as input, and returns $\mathsf{TM.Ext}(MK, t)$. $\mathcal{A}$ can query to $\mathsf{Release}(MK, \cdot)$ at most q_t times, however, it cannot submit the target time t^* to *Release*$(MK, \cdot)$ after the chal stage.

Definition 3.8 (IND-MCTPA). *For* $^{\exists}\kappa_0 \in \mathbb{N}$ *and* $^{\forall}\kappa \geq \kappa_0$*, a TR-ME scheme* Π_{TRME} *is said to be* (q_t, ϵ)*-IND-MCTPA secure if there exists a negligible* ϵ *in* κ *such that* $Adv^{\mathsf{IND\text{-}MCTPA}}_{\Pi_{\mathrm{TRME}},\mathcal{A}}(\kappa) < \epsilon$ *holds for any PPT adversary* $\mathcal{A}$*, where* q_t *is the number of queries that* $\mathcal{A}$ *can issue to the time-signal generation oracle in the IND-MCTPA game.*

Construction

As mentioned in Section 3.4.2, we can realize a secure TR-ME scheme in the above sense by replacing a (k, n)-CSS scheme in the Dodis–Katz construction with a (k, n)-TR-CSS scheme. Let Π_{PKE} =(Gen, Enc, Dec) be a PKE scheme, Π_{OTS} =(KG, Sign, Vrfy) be an OTS scheme, and Π_{TCSS} =(Setup, Ext, Share, Recon) be a (k, n)-TR-CSS scheme. A TR-ME scheme Π_{TRME} is constructed from $\Pi_{\mathrm{PKE}}, \Pi_{\mathrm{OTS}}$ and Π_{TCSS} as follows.

- $(Par, MK) \leftarrow \mathsf{TM.Setup}(1^\kappa)$:
  1. Output $(Par, MK) := (mpk, msk) \leftarrow \mathsf{Setup}(1^\kappa)$.
- $(MEK, \boldsymbol{MDK}) \leftarrow \mathsf{TM.KeyGen}(Par, \Gamma)$:
  1. Compute $(pk_i, sk_i) \leftarrow \mathsf{Gen}(1^\kappa)$ for every $i \in [n]$.
  2. Output $MEK := (pk_1, \ldots, pk_n, \Gamma)$ and $\boldsymbol{MDK} := (dk_1, \ldots, dk_n)$ where $MDK_i := dk_i$ $(1 \leq i \leq n)$.
- $ts^{(t)} \leftarrow \mathsf{TM.Ext}(MK, t)$:
  1. Output $ts^{(t)} := \tilde{ts}^{(t)} \leftarrow \mathsf{Ext}(msk, t)$.
- $C^{(t)} \leftarrow \mathsf{TM.Enc}_L(Par, MEK, M, t)$:
  1. Compute $(VK, SK) \leftarrow \mathsf{KG}(1^\kappa)$ and $(u_1^{(t)}, \ldots, u_n^{(t)}) \leftarrow \mathsf{Share}(\Gamma, mpk, M, t)$.
  2. Set $C_i \leftarrow \mathsf{Enc}_{VK}(pk_i, u_i^{(t)})$ $(i \in [n])$.
  3. Compute $\sigma \leftarrow \mathsf{Sign}(SK, (C_1, \ldots, C_n, L))$.
  4. Output $C^{(t)} := (C_1, \ldots, C_n, VK, \sigma)$.
- $\boldsymbol{C}^{(t)}$ or $\perp \leftarrow \mathsf{TM.Split}_L(Par, MEK, C^{(t)})$:
  1. If $0 \leftarrow \mathsf{Vrfy}(VK, (C_1, \ldots, C_n, L), \sigma)$, then output $\perp$.
  2. Otherwise, output $\boldsymbol{C}^{(t)} := (C_1^{(t)}, \ldots, C_n^{(t)}, VK)$, where $C_i^{(t)} := C_i$ $(i \in [n])$.
- $M_i^{(t)}$ or $\perp \leftarrow \mathsf{TM.Dec}(MDK_i, C_i^{(t)})$:
  1. Output $M_i^{(t)} := \tilde{u}_i^{(t)} \leftarrow \mathsf{Dec}_{VK}(sk_i, C_i)$.
- M or $\perp \leftarrow \mathsf{TM.Combine}(\boldsymbol{M}^{(t)}, ts^{(t)})$:
  1. Output $M \leftarrow \mathsf{Recon}((\tilde{u}_1^{(t)}, \ldots, \tilde{u}_n^{(t)}), \tilde{ts}^{(t)})$.

Theorem 3.4. *If* Π_{PKE} *is* (q, ϵ_1)*-IND-CCA secure,* Π_{OTS} *is* ϵ_2*-sUF-OT secure and* Π_{TCSS} *meets* ϵ_3*-Type-I Privacy, then the resulting TR-ME scheme* Π_{TRME} *in the above construction is* (d, q_c, δ)*-IND-MCCA secure, where* $d = k - 1$, $q_c = (n - k + 1)q$ *and* $\delta \leq 2n\epsilon_1 + \epsilon_2 + \epsilon_3$.

Proof. We can prove this theorem in a similar way to the proof of [54, Theorem 1]. Let $\mathcal{A}$ be an IND-MCCA adversary against the proposed TR-ME scheme Π_{TRME}. We suppose $\mathsf{Game}_{\mathsf{Real}}$ is the original *IND-MCCA* game for $\mathcal{A}$ as follows.

Step 1. $\mathcal{W} \leftarrow \emptyset$, $(mpk, msk) \leftarrow \mathsf{Setup}(1^\kappa)$,
$(pk_i, sk_i) \leftarrow \mathsf{Gen}(1^\kappa)$ $(i \in [n])$,

Step 2. $(M_0, M_1, L^*, t^*, st) \leftarrow \mathcal{A}^{Corrupt(\cdot),\mathcal{O}}(\mathsf{chal}, mpk, msk, pk_1, \ldots, pk_n)$.

Step 3. $b \xleftarrow{\$} \{0,1\}$, $(VK^*, SK^*) \leftarrow \mathsf{KG}(1^\kappa)$,
$(u_1^{(t^*)}, \ldots, u_n^{(t^*)}) \leftarrow \mathsf{Share}(\Gamma, mpk, M_b, t^*)$,
$C_i^* \leftarrow \mathsf{Enc}_{VK^*}(pk_i, u_i^{(t^*)})$ $(i \in \{1, \ldots, n\})$,
$\sigma^* \leftarrow \mathsf{Sign}(SK^*, (C_1^*, \ldots, C_n^*, L^*))$,

Step 4. $b' \leftarrow \mathcal{A}^{Corrupt(\cdot),\mathcal{O}}(\mathsf{guess}, VK^*, C_1^*, \ldots, C_n^*, \sigma^*, st)$.

$Corrupt(\cdot)$ takes P_i as input, and computes $\mathcal{W} \leftarrow \mathcal{W} \cup \{P_i\}$ and outputs $DK_i = sk_i$. $\mathcal{O}$ takes $(C, L) = ((C_1, \ldots, C_n, VK, \sigma), L)$ as input, and outputs $(\mathsf{Dec}_{VK}(sk_1, C_1), \ldots, \mathsf{Dec}_{VK}(sk_n, C_n))$ if $1 \leftarrow \mathsf{Vrfy}(VK, (C_1, \ldots, C_n, L), \sigma)$. Note that $\mathcal{A}$ is not allowed to issue (C, L) such that $(C, L) \neq (C^*, L^*)$.

We say that a pair of a ciphertext and a label $(C = (C_0, \cdots, C_4, VK, \sigma), L)$ is *valid* if $\mathsf{Vrfy}(vk, (C_1, \cdots, C_n, L), \sigma) \rightarrow 1$. Suppose that $(VK^*, SK^*) \leftarrow \mathsf{KG}(1^\kappa)$ is generated in Step 1 (not in Step 3). This modification does not affect adversary's view. We define the following event:

Forge: $\mathcal{A}$ issues a query $(C = (C_1, \ldots, C_n, VK^*, \sigma), L)$ such that (C, L) is valid to $\mathcal{O}$.

We also define the following games:

Game_0: This is the same as $\mathsf{Game}_{\mathsf{Real}}$ except that Forge never occurs.

Game_k $(1 \leq k \leq n - k + 1)$**:** This is the same as Game_{k-1} except for the following modification: The k-th share $u_i^{(t)}$ of the challenge plaintext M_b is $0^{|u_i^{(t)}|}$.

$\widetilde{\mathsf{Game}}_k$ $(1 \leq k \leq n - k + 1)$**:** This is the same as Game_k except for the following modification: $\mathcal{A}$ never issues a query k to *Corrupt*.

Let X_{Real}, X_k $(0 \leq k \leq n - k + 1)$, $\tilde{X}_k$ $(0 \leq k \leq n - k + 1)$, and X_{Final} be events that $b' = b$ in $\mathsf{Game}_{\mathsf{Real}}$, Game_k, $\widetilde{\mathsf{Game}}_k$, and $\mathsf{Game}_{\mathsf{Final}}$, respectively.

The rest of the proof follows from the following lemmas.

Lemma 3.3. $\Pr[Forge] \leq \epsilon_2$.

Proof. We construct a PPT adversary $\mathcal{F}$ against the sUF-OT security by using a PPT adversary $\mathcal{A}$ against the IND-MCCA security. $\mathcal{F}$ keeps VK^* received from the challenger of the sUF-OT game. $\mathcal{F}$ generates $(mpk, msk) \leftarrow \mathsf{Setup}(1^\kappa)$ and $(pk_i, sk_i) \leftarrow \mathsf{Gen}(1^\kappa)$ for every $i \in [n]$, and sends $(mpk, msk, pk_1, \ldots, pk_n)$ to $\mathcal{A}$. $\mathcal{F}$ can answer any queries for all oracles. After receiving (M_0^*, M_1^*, L^*, t^*) from $\mathcal{A}$, $\mathcal{F}$ creates $C_1^*, \ldots, C_n^*$, issues $(C_1^*, \ldots, C_n^*, L^*)$ to the *Sign* oracle, and gets σ^*. $\mathcal{F}$ sends $(C_1^*, \ldots, C_n^*, VK^*, \sigma^*)$ as a challenge ciphertext. At some point, $\mathcal{F}$ receives a query $((C_1, \ldots, C_n, VK^*, \sigma), L)$. Then, $\mathcal{F}$ simply outputs $((C_1, \ldots, C_n, L), \sigma)$ as a forgery since $((C_1, \ldots, C_n, L), \sigma) \neq ((C_1^*, \ldots, C_n^*, L^*), \sigma^*)$. □

Lemma 3.4. $\frac{n-k+1}{n}\Pr[X_j] = \Pr[\tilde{X}_j]$ *for every* $j \in [n-k+1]$.

Proof. Since each secret key $MDK_j = sk_j$ is generated with the same procedure, we can consider that the probability that $\mathcal{A}$ chooses an identity j for the corrupt query is uniform. Then, the probability that j is not included a set of the $k-1$ identities chosen by $\mathcal{A}$ is

$$\frac{\binom{n-1}{k-1}}{\binom{n}{k-1}} = \frac{n-k+1}{n}.$$

Therefore, we have $\frac{n-k+1}{n}\Pr[X_j] = \Pr[\tilde{X}_j]$ for every $j \in [n-k+1]$. □

Lemma 3.5. $|\Pr[\tilde{X}_j] - \Pr[\tilde{X}_{j-1}]| \leq 2\epsilon_1$ *for every* $j \in [n-k+1]$.

Proof. We construct a PPT adversary $\mathcal{B}$ against the IND-CCA security by using a PPT adversary $\mathcal{A}$ against the IND-MCCA security. First, $\mathcal{B}$ receives pk^* from the challenger of the IND-CCA game and sets $pk_j := pk^*$. $\mathcal{B}$ generates $(pk_i, sk_i) \leftarrow \mathsf{Gen}(1^\kappa)$ for every $i \in [n] \setminus \{j\}$, $(mpk, msk) \leftarrow \mathsf{Setup}(1^\kappa)$, and $(VK^*, SK^*) \leftarrow \mathsf{KG}(1^\kappa)$. Then, $\mathcal{B}$ sends $(mpk, msk, pk_1, \ldots, pk_n)$ to $\mathcal{A}$. When $\mathcal{A}$ issues a corrupt query i, then $\mathcal{B}$ returns sk_i. Note that $\mathcal{A}$ never issues j. When receiving $(C^{(t)}, L) = (C_1, \ldots, C_n, VK, \sigma, L)$, $\mathcal{B}$ runs $\mathsf{Vrfy}(VK, (C_1, \ldots, C_n), \sigma)$. If it outputs 0, then $\mathcal{B}$ returns $\perp$ to $\mathcal{A}$. Otherwise, $\mathcal{B}$ computes $\tilde{u}_i^{(t)} \leftarrow \mathsf{Dec}_{VK}(sk_i, C_i)$ for every $i \in [n] \setminus \{j\}$ and gets $\tilde{u}_j^{(t)}$ by issuing a query (C_j, VK) to the decryption oracle of the IND-CCA game. Then, $\mathcal{B}$ returns $(\tilde{u}_1^{(t)}, \ldots, \tilde{u}_n^{(t)})$ to $\mathcal{A}$. In the challenge phase, suppose that $\mathcal{B}$ receives (M_0^*, M_1^*, L^*, t^*) from $\mathcal{A}$. Then, $\mathcal{B}$ chooses $\beta \xleftarrow{\$} \{0,1\}$, and computes $(u_1^{(t)}, \ldots, u_n^{(t)}) \leftarrow \mathsf{Share}(\Gamma, mpk, M_\beta^*, t^*)$. Then, $\mathcal{B}$ computes $C_i^* \leftarrow \mathsf{Enc}_{VK^*}(pk_i, 0^{|u_i^{(t)}|})$ for every $i \in [j-1]$, and $C_i^* \leftarrow \mathsf{Enc}_{VK^*}(pk_i, u_i^{(t)})$ for every $i \in \{j+1, \ldots, n\}$. $\mathcal{B}$ issues $(m_0^*, m_1^*, L^*) := (u_j^{(t)}, 0^{|u_j^{(t)}|}, VK^*)$ to the challenger of the IND-CCA game. After receiving $C_j^* \leftarrow \mathsf{Enc}_{VK^*}(pk_j, m_b^*)$, where $b \xleftarrow{\$} \{0,1\}$, $\mathcal{B}$ computes $\sigma^* \leftarrow \mathsf{Sign}(SK^*, (C_1^*, \ldots, C_n^*))$, and returns $C^{t^*} = (C_1^*, \ldots, C_n^*, VK^*, \sigma^*)$ to $\mathcal{A}$. Finally, when receiving β' from $\mathcal{A}$, $\mathcal{B}$ sends $b' := 0$ if $\beta' = \beta$ to the challenger. Otherwise, $\mathcal{B}$ sends $b' := 1$ to the challenger. □

Lemma 3.6. $|\Pr[\tilde{X}_{n-k+1}] - \frac{1}{2}| = \epsilon_3$.

Proof. We construct a PPT adversary $\mathcal{B}$ against the Type-I Privacy by using a PPT adversary $\mathcal{A}$ against the IND-MCCA security. First, $\mathcal{B}$ receives (mpk, msk, Γ) from the challenger of the Type-I Privacy game, and sets $Par := mpk$ and $MK := msk$. $\mathcal{B}$ generates $(pk_i, sk_i) \leftarrow \mathsf{Gen}(1^\kappa)$ for every $i \in [n]$ and $(VK^*, SK^*) \leftarrow \mathsf{KG}(1^\kappa)$, and sets $MEK := (pk_1, \ldots, pk_n, \Gamma)$. When $\mathcal{A}$ issues a corrupt query i, then $\mathcal{B}$ returns sk_i. When receiving $(C^{(t)}, L) = (C_1, \ldots, C_n, VK, \sigma, L)$, $\mathcal{B}$ runs $\mathsf{Vrfy}(VK, (C_1, \ldots, C_n), \sigma)$. If it outputs 0, then

$\mathcal{B}$ returns $\perp$ to $\mathcal{A}$. Otherwise, $\mathcal{B}$ computes $\tilde{u}_i^{(t)} \leftarrow \mathsf{Dec}_{VK}(sk_i, C_i)$ for every $i \in [n]$, and returns $(\tilde{u}_1^{(t)}, \ldots, \tilde{u}_n^{(t)})$ to $\mathcal{A}$. In the challenge phase, suppose that $\mathcal{B}$ receives (M_0^*, M_1^*, L^*, t^*) from $\mathcal{A}$. Then, $\mathcal{B}$ sends to (M_0^*, M_1^*, t^*) to the challenger of the Type-I Privacy, and gets $(u_1^{(t)}, \ldots, u_n^{(t)}) \leftarrow \mathsf{Share}(\Gamma, mpk, M_b^*, t^*)$, where $b \xleftarrow{\$} \{0,1\}$. Then, $\mathcal{B}$ computes $C_i^* \leftarrow \mathsf{Enc}_{VK^*}(pk_i, 0^{|u_i^{(t)}|})$ for every $i \in [n-k+1]$, and $C_i^* \leftarrow \mathsf{Enc}_{VK^*}(pk_i, u_i^{(t)})$ for every $i \in \{n-k+2, \ldots, n\}$. $\mathcal{B}$ computes $\sigma^* \leftarrow \mathsf{Sign}(SK^*, (C_1^*, \ldots, C_n^*))$, and returns $C^{t^*} = (C_1^*, \ldots, C_n^*, VK^*, \sigma^*)$ to $\mathcal{A}$. Finally, after receiving b' from $\mathcal{A}$, $\mathcal{B}$ transfers b' to the challenger. □

From Lemmas 3.3–3.6, we have

$$
\begin{aligned}
&Adv^{IND\text{-}MCCA}_{\Pi_{\mathrm{TRME}},\mathcal{A}} \\
&= \left|\Pr[X_{\mathsf{Real}}] - \frac{1}{2}\right| \\
&\leq \Pr[X_{\mathsf{Real}} \wedge \mathsf{Forge}] + \left|\Pr[X_{\mathsf{Real}} \wedge \overline{\mathsf{Forge}}] - \frac{1}{2}\right| \\
&\leq \Pr[\mathsf{Forge}] + \left|\Pr[X_0] - \frac{1}{2}\right| \\
&\leq \Pr[\mathsf{Forge}] + \sum_{i=1}^{n-k+1} |\Pr[X_i] - \Pr[X_{i-1}]| + \left|\Pr[X_{n-k+1}] - \frac{1}{2}\right| \\
&= \Pr[\mathsf{Forge}] + \frac{n}{n-k+1} \sum_{i=1}^{n-k+1} \left|\Pr[\tilde{X}_i] - \Pr[\tilde{X}_{i-1}]\right| + \left|\Pr[X_{n-k+1}] - \frac{1}{2}\right| \\
&\leq \epsilon_2 + 2n\epsilon_1 + \epsilon_3.
\end{aligned}
$$

Thus, the proof is completed. □

Theorem 3.5. *If Π_{TCSS} meets (q, ϵ)-Type-II Privacy, then the resulting TR-ME scheme Π_{TRME} in the above construction is (q_t, δ)-IND-MCTPA secure, where $q_t = q$ and $\delta \leq \epsilon$.*

The proof is omitted since it is straightforward.

Remark 3.2. *We have not focused on decryption robustness discussed in [54], since this property can be easily achieved if the underlying SS scheme meets robustness (see [54] for the similar discussion). As mentioned in Section 3.3.5, TR-CSS schemes can be modified to meet robustness by using DSs. Therefore, a TR-ME scheme with decryption robustness can be achieved by using our robust (k, n)-TR-CSS scheme.*

3.4.4 Timed-Release Threshold Encryption

In this section, we consider threshold encryption (TE) in the timed-release security setting. Dodis and Katz [54] showed an outline of a transformation

from a ME scheme to a TE scheme, which we call a *Dodis–Katz transformation*. Actually, the Dodis–Katz transformation can provide only TE schemes with restricted threshold access structures, namely, (n, n)-TE schemes. Nonetheless, we can also realize a (n, n)-TE scheme with timed-release functionality (a TR-TE scheme) from a TR-ME scheme, since timed-release functionality have little or no effect on traditional security of ME and TE.

The model

We can also consider *suitable* TR-TE for the similar reason as TR-ME. A TR-TE scheme Π_{TRTE} consists of six-tuple algorithms ($\mathsf{TT.Setup}$, $\mathsf{TT.KeyGen}$, $\mathsf{TT.Ext}$, $\mathsf{TT.Enc}$, $\mathsf{TT.ShareDec}$, $\mathsf{TT.Combine}$) defined as follows, where $\mathcal{M}_{\text{TTE}}$ is a set of plaintexts determined by a security parameter κ.

- $(PP, MSK) \leftarrow \mathsf{TT.Setup}(1^\kappa)$: A probabilistic algorithm for setup. It takes a security parameter as input and outputs a public parameter PP and a master secret key MSK.
- $(TEK, \boldsymbol{TDK}) \leftarrow \mathsf{TT.KeyGen}(PP, \Gamma)$: A probabilistic algorithm for key generation. It takes the public parameter PP and a (k, n)-threshold access structure $\Gamma = (\mathscr{Q}, \mathscr{F})$ as input and outputs a public encryption key TEK and secret keys $\boldsymbol{TDK} := (TDK_1, \ldots, TDK_n)$.
- $ts^{(t)} \leftarrow \mathsf{TT.Ext}(MSK, t)$: A probabilistic algorithm for generating time-signals. It takes the master secret key MK and time $t \in \mathcal{T}$ as input and outputs a time-signal $ts^{(t)}$ at time t.
- $C^{(t)} \leftarrow \mathsf{TT.Enc}(PP, TEK, M, t)$: A probabilistic algorithm for encryption. It takes the public parameter PP, the public encryption key TEK, a plaintext $M \in \mathcal{M}_{\text{TTE}}$ and a specified time t as input, and then outputs a ciphertext $C^{(t)}$ at time t.
- $M_i^{(t)}$ or $\perp \leftarrow \mathsf{TT.ShareDec}(TDK_i, C^{(t)})$: A deterministic algorithm for decryption. It takes the secret key TDK_i and a ciphertext $C^{(t)}$ as inputs and then outputs a decryption share $M_i^{(t)}$ or $\perp$.
- M or $\perp \leftarrow \mathsf{TT.Combine}(C^{(t)}, \boldsymbol{M}^{(t)}, ts^{(t)})$: A deterministic algorithm for combining plaintext shares. It takes a ciphertext $C^{(t)}$, at least k decryption shares $\boldsymbol{M}^{(t)} := (M_{i_1}^{(t)}, \ldots, M_{i_j}^{(t)})$ $(k \leq j)$ of $C^{(t)}$, and the time-signal $ts^{(t)}$ at the specified time t as input, and then outputs the plaintext M or $\perp$.

In the above model, we assume that Π_{TRTE} meets the following *correctness* property: For all $\kappa \in \mathbb{N}$, all $(PP, MSK) \leftarrow \mathsf{TT.Setup}(1^\kappa)$, all $\Gamma \subset 2^{\mathcal{P}}$ where $|\mathcal{P}| = n = poly(\kappa)$, all $(TEK, \boldsymbol{TDK}) \leftarrow \mathsf{TT.KeyGen}(PP, \Gamma)$, all t, all $ts^{(t)} \leftarrow$

$\mathsf{TT.Ext}(MSK, t)$, all $M \in \mathcal{M}_{\mathrm{TTE}}$, and all $C^{(t)} \leftarrow \mathsf{TT.Enc}(PP, TEK, M, t)$, it holds and

$$M \leftarrow \mathsf{TT.Combine}(C^{(t)}, \boldsymbol{DEC}(C^{(t)}), ts^{(t)})$$

where $\boldsymbol{DEC}(C^{(t)}) := (\mathsf{TT.ShareDec}(TDK_{i_1}, C^{(t)}), \ldots, \mathsf{TT.ShareDec}(TDK_{i_j}, C^{(t)}))$ $(k \leq j \leq n)$.

Security

We consider similar security notions as those of TR-ME. Therefore, we define the notion of indistinguishability against threshold chosen ciphertext attack (IND-TCCA) for time-server security and the notion of Indistinguishability against threshold chosen time and plaintext attack (IND-TCTPA) for insider security.

To formalize the former notion, we consider the following IND-TCCA game:

$$Adv^{\mathsf{IND\text{-}TCCA}}_{\Pi_{\mathrm{TRTE}},\mathcal{A}}(\kappa) := \left| \Pr\left[b' = b \;\middle|\; \begin{array}{l} \mathcal{W} \leftarrow \emptyset,\ (PP, MSK) \leftarrow \mathsf{TT.Setup}(1^\kappa), \\ (TEK, \boldsymbol{TDK}) \leftarrow \mathsf{TT.KeyGen}(PP, \Gamma), \\ (M_0^*, M_1^*, t^*, st) \\ \qquad \leftarrow \mathcal{A}^{Corrupt(\cdot),\mathcal{O}}(\mathsf{chal}, TEK, PP, MSK), \\ b \xleftarrow{\$} \{0,1\},\ C^{(t^*)} \leftarrow \mathsf{TT.Enc}(PP, TEK, M_b, t^*), \\ b' \leftarrow \mathcal{A}^{Corrupt(\cdot),\mathcal{O}}(\mathsf{guess}, C^{(t^*)}, st) \end{array} \right] - \frac{1}{2} \right|.$$

Here, we require $|M_0^*| = |M_1^*|$, and st is state information. $Corrupt(\cdot)$ is a *corrupt oracle* which takes an ID i as input, and then $\mathcal{W} \leftarrow \mathcal{W} \cup \{i\}$ and returns TDK_i. $\mathcal{A}$ can query to $Corrupt(\cdot)$ until $|\mathcal{W}| = d$ $(< k)$. In addition, $\mathcal{O}$ is a decryption oracle which takes $(C^{(t)}, i)$ as input and outputs $\mathsf{TT.ShareDec}(TDK_{(\cdot)}, \cdot)$. $\mathcal{A}$ is allowed to access the decryption oracle at most q_c times at any time, however it cannot submit the target ciphertext $C^{(t^*)}$ to the oracle.

Definition 3.9 (IND-TCCA). *For* $^{\exists}\kappa_0 \in \mathbb{N}$, $^{\forall}\kappa \geq \kappa_0$, *and any* (k, n)*-threshold access structure, a TR-TE scheme* Π_{TRTE} *is said to be* (d, q_c, ϵ)*-IND-TCCA secure if there exists a negligible* ϵ *in* κ *such that* $Adv^{\mathsf{IND\text{-}TCCA}}_{\Pi_{\mathrm{TRTE}},\Gamma,\mathcal{A}}(\kappa) < \epsilon$ *holds for any PPT adversary* $\mathcal{A}$*, where* d *is the maximum number of secret keys that* $\mathcal{A}$ *can obtain and* q_c *is the number of queries that* $\mathcal{A}$ *can issue to the decryption oracle in the IND-TCCA game.*

Next, we formalize the latter notion.

$$Adv^{\mathsf{IND\text{-}TCTPA}}_{\Pi_{\mathrm{TRTE}},\mathcal{A}}(\kappa) :=$$

$$\left| \Pr\left[b' = b \;\middle|\; \begin{array}{l} \mathcal{W} = \mathcal{P},\ (PP, MSK) \leftarrow \mathsf{TT.Setup}(1^\kappa), \\ (TEK, \boldsymbol{TDK}) \leftarrow \mathsf{TT.KeyGen}(PP, \Gamma), \\ (M_0^*, M_1^*, t^*, st) \\ \qquad \leftarrow \mathcal{A}^{Release(MSK,\cdot)}(\mathsf{chal}, TEK, \boldsymbol{TDK}, PP), \\ b \xleftarrow{\$} \{0,1\},\ C^{(t^*)} \leftarrow \mathsf{TT.Enc}(PP, TEK, M_b, t^*), \\ b' \leftarrow \mathcal{A}^{Release(MSK,\cdot)}(\mathsf{guess}, C^{(t^*)}, st) \end{array} \right] - \frac{1}{2} \right|.$$

Here, we require $|M_0^*| = |M_1^*|$, and st is state information. $Release(MSK, \cdot)$ is a *time-signals generation oracle* which takes time t as input, and returns $\mathsf{TT.Ext}(MSK, t)$. $\mathcal{A}$ can query to $Release(MSK, \cdot)$ at most q_t times, however, it cannot submit the target time t^* to $Release(MSK, \cdot)$ after the chal stage.

Definition 3.10 (IND-TCTPA). *For $^\exists\kappa_0 \in \mathbb{N}$, $^\forall\kappa \geq \kappa_0$, and any (k, n)-threshold access structure, a TR-TE scheme Π_{TRTE} is said to be (q_t, ϵ)-IND-TCTPA secure if there exists a negligible ϵ in κ such that $Adv^{\mathsf{IND\text{-}TCTPA}}_{\Pi_{\mathrm{TRTE}},\Gamma,\mathcal{A}}(\kappa) < \epsilon$ holds for any PPT adversary $\mathcal{A}$, where q_t is the number of queries that $\mathcal{A}$ can issue to the time-signal generation oracle in the IND-TCTPA game.*

Construction

To begin with, we note that we can obtain only (n, n)-threshold encryption schemes from multiple encryption schemes by the Dodis–Katz transformation, though this fact was not mentioned in [54]. The reason why the Dodis–Katz transformation can provide only (n, n)-threshold encryption schemes is that a combine algorithm in multiple encryption schemes requires *all* decryption shares (i.e., n decryption shares), whereas a combine algorithm in threshold encryption schemes requires *at least* k decryption shares. Namely, the combine algorithm in a threshold encryption scheme cannot execute the combine algorithm in a multiple encryption scheme with less than n decryption shares. Therefore, we can construct only a TR-TE scheme with an (n, n)-threshold access structure from a TR-ME scheme by the Dodis–Katz transformation, and further, we can also construct a TR-TE scheme with a (k, n)-threshold access structure in a similar way to a construction of a TR-ME scheme. Note that for the similar reason as TR-ME, we can effectively construct a TR-TE scheme by using a TR-CSS scheme. Here, we show how to apply the Dodis–Katz transformation to a TR-ME scheme, namely, a construction of a TR-TE scheme with an (n, n)-threshold access structure.

A TR-TE scheme Π_{TRTE} is constructed from a TR-ME scheme Π_{TRME} as follows.

- $(PP, MSK) \leftarrow \mathsf{TT.Setup}(1^\kappa)$:
  1. Output $(PP, MSK) := (Par, MK) \leftarrow \mathsf{TM.Setup}(1^\kappa)$.
- $(TEK, \boldsymbol{TDK}) \leftarrow \mathsf{TT.KeyGen}(PP, \Gamma)$:

1. Compute $(MEK, \boldsymbol{MDK}) \leftarrow \mathsf{TM.KeyGen}(Par, \Gamma)$ where Γ is an (n,n)-threshold access structure.
2. Output $TEK := MEK$ and $\boldsymbol{TDK} := \boldsymbol{MDK}$ where $TDK_i := MDK_i$ $(1 \leq i \leq n)$.

– $ts^{(t)} \leftarrow \mathsf{TT.Ext}(MSK, t)$:

1. Output $ts^{(t)} := \tilde{ts}^{(t)} \leftarrow \mathsf{TM.Ext}(MK, t)$.

– $C^{(t)} \leftarrow \mathsf{TT.Enc}(PP, TEK, M, t)$:

1. Compute $\widehat{C}^{(t)} \leftarrow \mathsf{TM.Enc}_{\varepsilon}(Par, MEK, M, t)$, where ε is an empty string (or, any string L can be also used).
2. Output $C^{(t)} := \widehat{\boldsymbol{C}}^{(t)} = (\widehat{C}_1^{(t)}, \ldots, \widehat{C}_n^{(t)}) \leftarrow \mathsf{TM.Split}_{\varepsilon}(Par, MEK, \widehat{C}^{(t)})$.

– $M_i^{(t)}$ or $\perp \leftarrow \mathsf{TT.ShareDec}(TDK_i, C^{(t)})$:

1. Output $M_i^{(t)} := \widehat{M}_i^{(t)} \leftarrow \mathsf{TM.Dec}(MDK_i, \widehat{C}_i^{(t)})$.

– M or $\perp \leftarrow \mathsf{TT.Combine}(VK, C^{(t)}, \boldsymbol{M}^{(t)}, ts^{(t)})$:

1. Output $M \leftarrow \mathsf{TM.Combine}(\widehat{M}_1^{(t)}, \ldots, \widehat{M}_n^{(t)}, ts^{(t)})$.

We have the following theorems. We omit the proofs since it is straightforward.

Theorem 3.6. *If given Π_{TRME} is (d', q, ϵ)-IND-MCCA secure, then the resulting (n,n)-TR-TE scheme Π_{TRTE} in the above construction is (d, q_c, δ)-IND-TCCA secure, where $d = d'$, $q_c = q$ and $\delta \leq \epsilon$.*

Theorem 3.7. *If given Π_{TRME} is (q, ϵ)-IND-MCTPA secure, then the resulting (n,n)-TR-TE scheme Π_{TRME} in the above construction is (q_t, δ)-IND-TCTPA secure, where $q_t = q$ and $\delta \leq \epsilon$.*

Moreover, if the underlying TR-ME scheme has decryption robustness, then the resulting (n,n)-TR-TE scheme has decryption robustness.

Chapter 4

Information-Theoretic Timed-Release Cryptography

4.1 Contribution in This Chapter

In this chapter, we study timed-release cryptography with information-theoretic security. As fundamental cryptographic primitives with information-theoretic security, we can consider information-theoretically secure key-agreement, encryption, authentication codes, and secret sharing. Therefore, in this chapter, we deal with and first realize information-theoretic timed-release security for all those primitives. Specifically, the contributions of this chapter are as follows.

- TR-KA (in Section 4.2). We propose a model and formalization of security for timed-release key-agreement (TR-KA) in information-theoretic security setting. We also derive tight lower bounds on size of secrets for entities required for TR-KA. In addition, we propose an optimal direct construction of TR-KA based on polynomials over finite fields.

- TRE (in Section 4.3). We propose a model and formalization of security for timed-release encryption (TRE) in information-theoretic security setting. In addition, we derive tight lower bounds on size of secrets for entities required for TRE. Furthermore, we present a simple generic construction of TRE: TRE can be constructed from TR-KA and the one-time pad. In particular, the application of our optimal direct construction of TR-KA in the generic construction leads to an optimal direct construction of TRE.

- TRA-code (in Section 4.4). We propose a model and formalization of security for timed-release authentication codes (TRA-codes) in information-theoretic security setting. We also derive tight lower bounds on size of secrets for entities required for TRA-codes. In addition, we present two

kinds of constructions, generic and direct ones. Our generic construction of TRA-codes is simple: TRA-codes can be constructed from TR-KA and traditional A-codes. Since the generic construction does not lead to an optimal construction of TRA-codes, we also propose an optimal direct construction by using polynomials over finite fields.

- TR-SS (in Section 4.5). We add timed-release functionality to SS schemes. Specifically, we conceive the following two types of schemes. We propose models and formalization of security for timed-release secret sharing (TR-SS) schemes in information-theoretic security setting. We also derive tight lower bounds on size of secrets for entities required for each TR-SS scheme, respectively. In addition, for each type of TR-SS scheme, we propose an optimal direct construction based on polynomials over finite fields.

4.2 Timed-Release Key-Agreement

4.2.1 Model and Security Definition

We show a model and a security definition of TR-KA with information-theoretic security. This is done based on those of timed-release schemes with computational security and those of traditional key-agreement with information-theoretic security.

In TR-KA, there are $n+2$ entities, n users $U_1, U_2, \ldots, U_n$ with $n \geq 2$, a time-server TS for broadcasting *time-signals* and a trusted authority TA. We assume that the identity of each user U_i is also denoted by U_i. In addition, when any two users communicate each other in a timed-release scheme (i.e., not only TR-KA but also TRE and TRA-codes in the following sections) under consideration in this chapter, we call a user who specifies the time a *sender* and the other a *receiver* for convenience.

Informally, TR-KA is executed as follows. In the initial phase, TA generates secret keys on behalf of U_i $(1 \leq i \leq n)$ and the time-server TS. After distributing these keys via secure channels, TA deletes them in his memory. Any user U_{i_1} can specify future time when U_{i_1} wants to share a common key with a user U_{i_2}, and he computes a common key in advance by using U_{i_1}'s secret key and the identity U_{i_2}. And U_{i_1} tells U_{i_2} the future time which U_{i_1} specified. The time-server TS periodically broadcasts a time-signal at each time which is generated by using TS's master key. When the specified time has come, U_{i_2} can compute a common key shared with U_{i_1} by using U_{i_2}'s secret key, the identity U_{i_1} and a time-signal of the specified time. Note that each user has two kinds of secret keys: one is used for generating a common key when he is a sender; and the other is used for deriving a common key when he is a receiver. In TR-KA, we consider a non-interactive model where any two users can share a common key without interactive communications.

Formally, we give the definition of TR-KA as follows.[1]

Definition 4.1 (TR-KA). *A TR-KA* Π_{KA} *involves* $n+2$ *entities,* TA, $U_1, U_2, \ldots, U_n$ *and* TS, *and consists of a four-tuple of algorithms (*Setup, Ext, KeyGen, KeyDer*) with five spaces,* $\mathcal{TCK}, \mathcal{TUK}, \mathcal{TMK}, \mathcal{T}$, *and* $\mathcal{TI}$, *where all of the above algorithms except* Setup *are deterministic and all of the above spaces are finite. In addition,* Π_{KA} *is executed with four phases as follows.*

- ***Notation:***
  - *Entities:* TA *is a trusted initializer,* U_i $(1 \leq i \leq n)$ *is a user and* TS *is a time-server which broadcasts time-signals. Let* $\mathcal{U} := \{U_1, U_2, \ldots, U_n\}$ *be the set of all users.*
  - *Spaces:* $\mathcal{TCK}$ *is a set of possible common keys, and* $\mathcal{TMK}$ *is a set of possible master keys with associated probability distribution* P_{TMK}. $\mathcal{T} := [\tau]$ *is a set of time.* $\mathcal{TI}^{(t)}$ *is a set of possible time-signals at time* $t \in \mathcal{T}$. *Let* $\mathcal{TI} := \bigcup_{t=1}^{\tau} \mathcal{TI}^{(t)}$. *Also,* $\mathcal{TUK}_i^{(S)}$ *is a set of possible* U_i*'s secret keys for common key generation. And also,* $\mathcal{TUK}_i^{(R)}$ *is a set of possible* U_i*'s secret keys for common key derivation. Then,* $\mathcal{TUK}_i := \mathcal{TUK}_i^{(S)} \times \mathcal{TUK}_i^{(R)}$ *is the set of possible secret keys for* U_i. *Let* $\mathcal{TUK}^{(S)} := \bigcup_{i=1}^{n} \mathcal{TUK}_i^{(S)}$, $\mathcal{TUK}^{(R)} := \bigcup_{i=1}^{n} \mathcal{TUK}_i^{(R)}$, *and* $\mathcal{TUK} := \bigcup_{i=1}^{n} \mathcal{TUK}_i$.
  - *Algorithms:* Setup *is a key generation algorithm which on input a security parameter* 1^{κ}, *outputs users' secret keys and a time-server's master key,* Ext: $\mathcal{TMK} \times \mathcal{T} \rightarrow \mathcal{TI}$ *is a time-signal generation algorithm for* TS, KeyGen: $\mathcal{TUK}^{(S)} \times \mathcal{T} \times \mathcal{U} \rightarrow \mathcal{TCK}$ *is a common key generation algorithm and* KeyDer: $\mathcal{TUK}^{(R)} \times \mathcal{TI} \times \mathcal{U} \rightarrow \mathcal{TCK}$ *is a common key derivation algorithm.*

1. ***Key Generation and Distribution.*** *In the initial phase,* TA *generates the following keys by using* Setup*: a master key* $tmk^* \in \mathcal{TMK}$ *for* TS*; and a secret key* $tuk_i = (tuk_i^{(S)}, tuk_i^{(R)}) \in \mathcal{TUK}_i$ *for* U_i $(i = 1, 2, \ldots, n)$. *These keys are distributed to corresponding entities via secure channels. After distributing these keys,* TA *deletes them from his memory. And,* TS *and* U_i *keep their keys secret, respectively.*

2. ***Time-signal Generation.*** *For broadcasting a time-signal at each time,* TS *generates a time-signal* $tmk^{(t)} = \mathsf{Ext}(tmk^*, t) \in \mathcal{TI}^{(t)}$ *by using a master key* tmk^* *and time* $t \in \mathcal{T}$. *Then,* TS *broadcasts it to all users via a broadcast channel.*

[1] Note that our models of information-theoretically secure timed-release schemes (i.e., Definitions 4.1, 4.4 and 4.7) are almost the same as those of computationally secure timed-release schemes (e.g., [36, 33, 39, 62, 38]) except for considering the trusted initializer in our models.

3. ***Common key Generation.*** *If U_{i_1} wants to share a common key with U_{i_2} at future time t, U_{i_1} computes a common key to be shared with U_{i_2} in advance, $tck_{i_1,i_2}^{(t)} = \mathsf{KeyGen}(tuk_{i_1}^{(S)}, t, U_{i_2}) \in \mathcal{TCK}$, by using his secret key $tuk_{i_1}^{(S)}$, time t, and the receiver's identity U_{i_2}. And, U_{i_1} tells U_{i_2} the specified time t via an authenticated channel.*

4. ***Common key Derivation.*** *On receiving the specified time t from U_{i_1}, and if the time t has come, U_{i_2} computes a common key $tck_{i_1,i_2}^{(t)} = \mathsf{KeyDer}(tuk_{i_2}^{(R)}, tmk^{(t)}, U_{i_1})$ by using his secret key $tuk_{i_2}^{(R)}$, a time-signal $tmk^{(t)}$ at time t, and the sender's identity U_{i_1}.*

In the model of TR-KA, we require the following equation holds: For all possible $t \in \mathcal{T}$, $i_1, i_2 \in [n]$, $tuk_{i_1}^{(S)} \in \mathcal{TUK}_{i_1}^{(S)}$, $tuk_{i_2}^{(R)} \in \mathcal{TUK}_{i_2}^{(R)}$, $tmk^{(t)} \in \mathcal{TI}^{(t)}$, we have $\mathsf{KeyGen}(tuk_{i_1}^{(S)}, t, U_{i_2}) = \mathsf{KeyDer}(tuk_{i_2}^{(R)}, tmk^{(t)}, U_{i_1})$. The above requirement implies that any two users can share a common key at the specified time without any error if they correctly follow the specification of TR-KA. In addition, $tck_{i_1,i_2}^{(t)}$ means a shared key between U_{i_1} and U_{i_2} at time t when U_{i_1} is the sender and U_{i_2} is the receiver, and we note that $tck_{i_1,i_2}^{(t)} \neq tck_{i_2,i_1}^{(t)}$ in general.

We now define several notation to formalize security of TR-KA as follows. Let ω $(< n)$ be the maximum number of possible colluders. For a set of colluders $\mathcal{W} = \{U_{l_1}, U_{l_2}, \ldots, U_{l_j}\} \in \mathcal{PS}(\mathcal{U}, \omega)$, $\mathcal{TUK}_{\mathcal{W}}^{(S)} := \mathcal{TUK}_{l_1}^{(S)} \times \mathcal{TUK}_{l_2}^{(S)} \times \cdots \times \mathcal{TUK}_{l_j}^{(S)}$ denotes the set of possible $\mathcal{W}$'s secret keys for common key generation, and $\mathcal{TUK}_{\mathcal{W}}^{(R)} := \mathcal{TUK}_{l_1}^{(R)} \times \mathcal{TUK}_{l_2}^{(R)} \times \cdots \times \mathcal{TUK}_{l_j}^{(R)}$ denotes the set of possible $\mathcal{W}$'s secret keys for common key derivation. And, let $\mathcal{TCK}_{i_1,i_2}^{(t)}$ be the set of possible common keys shared between U_{i_1} and U_{i_2} at the time $t \in \mathcal{T}$. Furthermore, let $TCK_{i_1,i_2}^{(t)}, TMK, TUK_{\mathcal{W}}^{(S)}, TUK_{\mathcal{W}}^{(R)}$, and $TI^{(1)}, \ldots, TI^{(\tau)}$ be random variables which take values in $\mathcal{TCK}_{i_1,i_2}^{(t)}, \mathcal{TMK}, \mathcal{TUK}_{\mathcal{W}}^{(S)}, \mathcal{TUK}_{\mathcal{W}}^{(R)}$, and $\mathcal{TI}^{(1)}, \ldots, \mathcal{TI}^{(\tau)}$, respectively.

Next, we formalize a security definition of TR-KA based on the idea of computational timed-release security and traditional key-agreement with information-theoretic security. In TR-KA, we consider the following security goal and attacking model. First, the security goal which we consider is basically the same as that of the traditional key-agreement: an adversary (or a dishonest entity) cannot obtain any information on a common key shared between two honest users. In addition to this, we want to require that even a legitimate receiver cannot obtain any information on a common key to be shared before the specified time comes (i.e., before a time-signal at the specified time is received), since we consider timed-release security in this paper. Secondly, as an attacking model we consider the following three types of attacks: (1) an attack by a dishonest time-server; (2) an attack by colluders

(i.e., dishonest users) not including a receiver; and (3) an attack by colluders including a receiver. By combining the security goal and attacks mentioned above, we formally define security of TR-KA as follows.

Definition 4.2. *Let Π_{KA} be TR-KA. Π_{KA} is said to be (n,ω,τ)-secure if the following conditions are satisfied.*

(1) For any $U_{i_1}, U_{i_2} \in \mathcal{U}$ and $t \in \mathcal{T}$, it holds that

$$H(TCK_{i_1,i_2}^{(t)} \mid TMK) = H(TCK_{i_1,i_2}^{(t)}).$$

(2) For any $\mathcal{W} \in \mathcal{PS}(\mathcal{U},\omega)$, $U_{i_1}, U_{i_2} \in \mathcal{U}$ such that $U_{i_1}, U_{i_2} \notin \mathcal{W}$, and for any $t \in \mathcal{T}$, it holds that

$$H(TCK_{i_1,i_2}^{(t)} \mid TUK_{\mathcal{W}}^{(S)}, TUK_{\mathcal{W}}^{(R)}, TI^{(1)}, \ldots, TI^{(\tau)}) = H(TCK_{i_1,i_2}^{(t)}).$$

(3) For any $\mathcal{W} \in \mathcal{PS}(\mathcal{U},\omega)$, $U_{i_1}, U_{i_2} \in \mathcal{U}$ such that $U_{i_1} \notin \mathcal{W}$ and $U_{i_2} \in \mathcal{W}$, for any $t \in \mathcal{T}$, it holds that

$$H(TCK_{i_1,i_2}^{(t)} \mid TUK_{\mathcal{W}}^{(S)}, TUK_{\mathcal{W}}^{(R)}, TI^{(1)}, \ldots, TI^{(t-1)}, TI^{(t+1)}, \ldots, TI^{(\tau)}) = H(TCK_{i_1,i_2}^{(t)}).$$

Intuitively, the meaning of formalizations (1)–(3) in Definition 4.2 is explained as follows: (1) a dishonest time-server cannot obtain any information on a common key shared between two honest users. However, we assume that the time-server correctly broadcasts a time-signal at each time; (2) No information on a common key shared between two honest users is obtained by any colluding group $\mathcal{W}$ not including a legitimate receiver, even if $\mathcal{W}$ obtains time-signals at all the time; (3) No information on a common key between two users at the specified time is obtained by any colluding group $\mathcal{W}$ including a legitimate (but dishonest) receiver, even if $\mathcal{W}$ obtains time-signals at all the time except the specified time.[2]

4.2.2 Lower Bounds

We derive lower bounds on size of secrets for entities and size of time-signals required for secure TR-KA as follows.

Theorem 4.1. *Let Π_{KA} be (n,ω,τ)-secure TR-KA, and we assume that all entropies on common keys are equal, namely $H(TCK) = H(TCK_{i_1,i_2}^{(t)})$ for any $i_1, i_2 \in [n]$ and $t \in \mathcal{T}$. Then, for any $i \in [n]$ and $t \in \mathcal{T}$, we have*

$$(i)\ H(TUK_i^{(R)}) \geq (\omega+1)H(TCK), \quad (ii)\ H(TUK_i^{(S)}) \geq (\tau+\omega)H(TCK),$$
$$(iii)\ H(TI^{(t)}) \geq (\omega+1)H(TCK), \quad (iv)\ H(TMK) \geq \tau(\omega+1)H(TCK).$$

[2] In this sense, we have formalized the security notion stronger than the security that a dishonest receiver cannot obtain any information on a common key to be shared *before* the specified time comes.

Proof. The proof follows from the following lemmas.

Lemma 4.1. $H(TUK_i^{(R)}) \geq (\omega+1)H(TCK)$ *for any* $i \in [n]$.

Proof. For arbitrary $i \in [n]$, we take a subset $B := \{l_1, l_2, \ldots, l_{\omega+1}\} \subset [n]$ of indices of users such that $i \notin B$. Let $D_k := (l_k, i)$ and $\mathcal{W}_k := \{l_1, l_2, \ldots, l_k\}$ for each k with $1 \leq k \leq \omega+1$. Then, we have

$$\begin{aligned}
H(TUK_i^{(R)}) &\geq H(TUK_i^{(R)} \mid TI^{(t)}) \\
&\geq I(TCK_{D_1}^{(t)}, TCK_{D_2}^{(t)}, \ldots, TCK_{D_{\omega+1}}^{(t)}; TUK_i^{(R)} \mid TI^{(t)}) \\
&= H(TCK_{D_1}^{(t)}, TCK_{D_2}^{(t)}, \ldots, TCK_{D_{\omega+1}}^{(t)} \mid TI^{(t)}) \\
&\qquad - H(TCK_{D_1}^{(t)}, TCK_{D_2}^{(t)}, \ldots, TCK_{D_{\omega+1}}^{(t)} \mid TI^{(t)}, TUK_i^{(R)}) \\
&= H(TCK_{D_1}^{(t)}, TCK_{D_2}^{(t)}, \ldots, TCK_{D_{\omega+1}}^{(t)} \mid TI^{(t)}) \\
&= \sum_{k=1}^{\omega+1} H(TCK_{D_k}^{(t)} \mid TI^{(t)}, TCK_{D_1}^{(t)}, TCK_{D_2}^{(t)}, \ldots, TCK_{D_{k-1}}^{(t)}) \\
&\geq \sum_{k=1}^{\omega+1} H(TCK_{D_k}^{(t)} \mid TUK_{\mathcal{W}_{k-1}}^{(S)}, TI^{(t)}) \\
&= \sum_{k=1}^{\omega+1} H(TCK_{D_k}^{(t)}) \qquad (4.1) \\
&= (\omega+1)H(TCK),
\end{aligned}$$

where Eq. (4.1) follows from the condition (2) in Definition 4.2. □

Lemma 4.2. $H(TUK_i^{(S)}) \geq (\tau+\omega)H(TCK)$ *for any* $i \in [n]$.

Proof. For arbitrary $i \in [n]$, we take a subset $B := \{l_1, l_2, \ldots, l_{\omega+1}\} \subset [n]$ of indices of users such that $i \notin B$. Let $D_k := (i, l_k)$ and $\mathcal{W}_k := \{l_1, l_2, \ldots, l_k\}$ for each k with $1 \leq k \leq \omega+1$. Also, let $F_k^{(t)} := (TCK_{D_k}^{(1)}, TCK_{D_k}^{(2)}, \ldots, TCK_{D_k}^{(t)})$ and $G_k^{(t)} := (TCK_{D_1}^{(t)}, TCK_{D_2}^{(t)}, \ldots, TCK_{D_k}^{(t)})$ for $1 \leq k \leq \omega+1$ and $1 \leq t \leq \tau$. Then, we have

$$\begin{aligned}
&H(TUK_i^{(S)}) \qquad (4.2) \\
&\geq H(F_1^{(\tau)}, G_{\omega+1}^{(t)}) \\
&= H(F_1^{(\tau)}) + H(G_{\omega+1}^{(t)} \mid F_1^{(\tau)}) \\
&= \sum_{t=1}^{\tau} H(TCK_{D_1}^{(t)} \mid F_1^{(t-1)}) + \sum_{k=2}^{\omega+1} H(TCK_{D_k}^{(t)} \mid F_1^{(\tau)}, TCK_{D_2}^{(t)}, \ldots, TCK_{D_{k-1}}^{(t)})
\end{aligned}$$

$$
\begin{aligned}
&\geq \sum_{t=1}^{\tau} H(TCK_{D_1}^{(t)} \mid TUK_{D_1}^{(R)}, TI^{(1)}, \ldots, TI^{(t-1)}) \\
&\qquad\qquad + \sum_{k=2}^{\omega+1} H(TCK_{D_k}^{(t)} \mid TUK_{\mathcal{W}_{k-1}}^{(R)}, TI^{(1)}, \ldots, TI^{(\tau)}) \\
&= \sum_{t=1}^{\tau} H(TCK_{D_1}^{(t)}) + \sum_{k=2}^{\omega+1} H(TCK_{D_k}^{(t)}) \qquad (4.3) \\
&= (\tau+\omega) H(TCK),
\end{aligned}
$$

where Eq. (4.3) follows from the conditions (2) and (3) in Definition 4.2. □

Lemma 4.3. *$H(TI^{(t)} \mid TI^{(1)}, \ldots, TI^{(t-1)}) \geq (\omega+1)H(TCK)$ for any $t \in \mathcal{T}$. In particular, $H(TI^{(t)}) \geq (\omega+1)H(TCK)$ for any $t \in \mathcal{T}$.*

Proof. For arbitrary $i \in [n]$, we take a subset $B := \{l_1, l_2, \ldots, l_{\omega+1}\} \subset [n]$ of indices of users such that $i = l_1$. Let $D_k := (l_k, i)$ and $\mathcal{W}_k := \{l_1, l_2, \ldots, l_k\}$ for each k with $1 \leq k \leq \omega+1$. Then, we have

$$
\begin{aligned}
&H(TI^{(t)} \mid TI^{(1)}, \ldots, TI^{(t-1)}) \qquad (4.4) \\
&\geq H(TI^{(t)} \mid TUK_i^{(R)}, TI^{(1)}, \ldots, TI^{(t-1)}) \\
&\geq I(TCK_{D_1}^{(t)}, TCK_{D_2}^{(t)}, \ldots, TCK_{D_{\omega+1}}^{(t)}; TI^{(t)} \mid TUK_i^{(R)}, TI^{(1)}, \ldots, TI^{(t-1)}) \\
&= H(TCK_{D_1}^{(t)}, TCK_{D_2}^{(t)}, \ldots, TCK_{D_{\omega+1}}^{(t)} \mid TUK_i^{(R)}, TI^{(1)}, \ldots, TI^{(t-1)}) \\
&\quad - H(TCK_{D_1}^{(t)}, \ldots, TCK_{D_{\omega+1}}^{(t)} \mid TUK_i^{(R)}, TI^{(1)}, \ldots, TI^{(t)}) \\
&= H(TCK_{D_1}^{(t)}, TCK_{D_2}^{(t)}, \ldots, TCK_{D_{\omega+1}}^{(t)} \mid TUK_i^{(R)}, TI^{(1)}, \ldots, TI^{(t-1)}) \\
&= \sum_{k=1}^{\omega+1} H(TCK_{D_k}^{(t)} \mid TUK_i^{(R)}, TI^{(1)}, \ldots, TI^{(t-1)}, TCK_{D_1}^{(t)}, \ldots, TCK_{D_{k-1}}^{(t)}) \\
&\geq \sum_{k=1}^{\omega+1} H(TCK_{D_k}^{(t)} \mid TUK_{\mathcal{W}_{k-1}}^{(S)}, TUK_i^{(R)}, TI^{(1)}, \ldots, TI^{(t-1)}) \\
&= \sum_{k=1}^{\omega+1} H(TCK_{D_k}^{(t)}) \qquad (4.5) \\
&= (\omega+1) H(TCK),
\end{aligned}
$$

where Eq. (4.5) follows from the condition (3) in Definition 4.2. □

Lemma 4.4. *$H(TMK) \geq \tau(\omega+1)H(TCK)$.*

Proof. We have

$$\begin{aligned} H(TMK) &\geq I(TI^{(1)},\ldots,TI^{(\tau)};TMK) \\ &= H(TI^{(1)},\ldots,TI^{(\tau)}) - H(TI^{(1)},\ldots,TI^{(\tau)} \mid TMK) \\ &= H(TI^{(1)},\ldots,TI^{(\tau)}) \\ &= \sum_{t=1}^{\tau} H(TI^{(t)} \mid TI^{(1)},\ldots,TI^{(t-1)}) \\ &= \tau(\omega+1)H(TCK), \end{aligned}$$

where the last equality follows from Lemma 4.3. □

Proof of Theorem 4.1. From Lemmas 4.1–4.4, the proof of Theorem 4.1 is completed. □

As we will see in Section 4.2.3, the above lower bounds are tight since our construction will meet all the above bounds with equalities. Therefore, we define optimality of constructions of TR-KA as follows.

Definition 4.3. *A construction of (n,ω,τ)-secure TR-KA is said to be optimal if it meets equality in every lower bound of (i)–(iv) in Theorem 4.1.*

4.2.3 Construction

We present a construction, which is provably secure TR-KA in our model, by using multivariate polynomials over finite fields. In addition, it is shown that the construction is optimal. The detail of our construction of TR-KA Π_{KA}=(Setup, Ext, KeyGen, KeyDer) is given as follows.

1. $(tmk^*.tuk_1,\ldots,tuk_n) \leftarrow \mathsf{Setup}(1^\kappa)$: For a security parameter 1^κ, Setup outputs matching secret keys tuk_i and tmk^* for U_i $(1 \leq i \leq n)$ and T, respectively, as follows. Setup picks a k-bit prime power q, where $q > \max n,\tau$, and constructs the finite field $\mathbb{F}_q$ with q elements. We assume that the identity of each user U_i is encoded as $U_i \in \mathbb{F}_q\backslash\{0\}$. Also, we assume $\mathcal{T} = [\tau] \subset \mathbb{F}_q\backslash\{0\}$ by using appropriate encoding. And, Setup chooses uniformly at random $f(x,y) := \sum_{i=0}^{\omega}\sum_{j=0}^{\omega} a_{ij}x^iy^j$, $tmk^*(x,z) := \sum_{i=0}^{\omega}\sum_{k=0}^{\tau-1} b_{ik}x^iz^k$ over $\mathbb{F}_q$ with three variables x, y and z in which each degree of x and y is at most ω, and the degree of z is at most $\tau-1$. Setup also computes $tuk_i^{(S)}(y,z) := f(U_i,y) + tmk^*(U_i,z)$ and $tuk_i^{(R)}(x) := f(x,U_i)$ $(1 \leq i \leq n)$. Then, Setup outputs secret keys $tuk_i := (tuk_i^{(S)}(y,z),\ tuk_i^{(R)}(x))$ $(1 \leq i \leq n)$ and $tmk^* := tmk^*(x,z)$ for U_i $(1 \leq i \leq n)$ and TS, respectively.

2. $tmk^{(t)} \leftarrow \mathsf{Ext}(tmk^*,t)$: For $tmk^* = tmk^*(x,z)$ and time $t \in \mathcal{T}$, Ext outputs a time-signal at time t, $tmk^{(t)}(x) := tmk^*(x,t)$.

3. $tck_{i_1,i_2}^{(t)} \leftarrow \mathsf{KeyGen}(tuk_{i_1}^{(S)}, t, U_{i_2})$: For a secret key $tuk_{i_1}^{(S)}$, the specified time t and an identity U_{i_2}, KeyGen generates a common key shared between U_{i_1} and U_{i_2}, $tck_{i_1,i_2}^{(t)} := tuk_{i_1}^{(S)}(U_{i_2}, t)$, and outputs it.

4. $tck_{i_1,i_2}^{(t)} \leftarrow \mathsf{KeyDer}(tuk_{i_2}^{(R)}, tmk^{(t)}, U_{i_1})$: For a secret key $tuk_{i_2}^{(R)}$, a time-signal $tmk^{(t)}$ at the specified time t and an identity U_{i_1}, KeyDer outputs a common key shared between U_{i_1} and U_{i_2}, $tck_{i_1,i_2}^{(t)} := tuk_{i_2}^{(R)}(U_{i_1}) + tmk^{(t)}(U_{i_1})$.

The security and optimality of the above construction is stated as follows.

Theorem 4.2. *The resulting TR-KA Π_{KA} by the above construction is (n, ω, τ)-secure and optimal.*

Proof. In this proof, we can write $f(x, y)$ and $tmk^*(x, z)$ in the form of

$$f(x,y) := (1, x, \ldots, x^{\omega})A \begin{pmatrix} 1 \\ y \\ \vdots \\ y^{\omega} \end{pmatrix} \text{ and } tmk^*(x,z) := (1, x, \ldots, x^{\omega})B \begin{pmatrix} 1 \\ z \\ \vdots \\ z^{\tau-1} \end{pmatrix},$$

respectively, where A is an $(\omega+1) \times (\omega+1)$ matrix and B is an $(\omega+1) \times \tau$ matrix, respectively. To complete the proof of Theorem 4.2, we show the following lemmas.

Lemma 4.5. *Let X be an $h \times i$ matrix, A be an $i \times j$ matrix, Y be a $j \times k$ matrix, W be an $h \times j$ matrix, and Z be an $i \times k$ matrix, respectively, where all entries of the matrices are elements in $\mathbb{F}_q$. When X, Y, W and Z are given, there are at least q solutions of A for the simultaneous linear equations, $W = XA$ and $Z = AY$, if $i > h$ and $j > k$.*

Proof. First, let X, A, and Y be

$$X = \begin{pmatrix} x_{1,1} & x_{1,2} & \cdots & x_{1,i} \\ x_{2,1} & x_{2,2} & \cdots & x_{2,i} \\ \vdots & \vdots & \ddots & \vdots \\ x_{h,1} & x_{h,2} & \cdots & x_{h,i} \end{pmatrix}, \quad A = \begin{pmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,j} \\ a_{2,1} & a_{2,2} & \cdots & a_{2,j} \\ \vdots & \vdots & \ddots & \vdots \\ a_{i,1} & a_{i,2} & \cdots & a_{i,j} \end{pmatrix}, \text{ and}$$

$$Y = \begin{pmatrix} y_{1,1} & y_{1,2} & \cdots & y_{1,k} \\ y_{2,1} & y_{2,2} & \cdots & y_{2,k} \\ \vdots & \vdots & \ddots & \vdots \\ y_{j,1} & y_{j,2} & \cdots & y_{j,k} \end{pmatrix},$$

respectively. Then, we can write

$$W = \begin{pmatrix} w_{1,1} & w_{1,2} & \cdots & w_{1,j} \\ w_{2,1} & w_{2,2} & \cdots & w_{2,j} \\ \vdots & \vdots & \ddots & \vdots \\ w_{h,1} & w_{h,2} & \cdots & w_{h,j} \end{pmatrix} \text{ and } Z = \begin{pmatrix} z_{1,1} & z_{1,2} & \cdots & z_{1,k} \\ z_{2,1} & z_{2,2} & \cdots & z_{2,k} \\ \vdots & \vdots & \ddots & \vdots \\ z_{i,1} & z_{i,2} & \cdots & z_{i,k} \end{pmatrix},$$

such that

$$\begin{aligned} w_{\ell,m} &= x_{\ell,1}a_{1,m} + x_{\ell,2}a_{2,m} + \cdots + x_{\ell,i}a_{i,m} \quad \text{for } 1 \le \ell \le h,\ 1 \le m \le j, \\ z_{\ell,m} &= a_{\ell,1}y_{1,m} + a_{\ell,2}y_{2,m} + \cdots + a_{\ell,j}y_{j,m} \quad \text{for } 1 \le \ell \le i,\ 1 \le m \le k. \end{aligned}$$

Since we have the equation $WY = XZ$, it holds that, for any α, β with $1 \le \alpha \le h$ and $1 \le \beta \le k$,

$$w_{\alpha,1}y_{1,\beta} + w_{\alpha,2}y_{2,\beta} + \cdots + w_{\alpha,j}y_{j,\beta} = x_{\alpha,1}z_{1,\beta} + x_{\alpha,2}z_{2,\beta} + \cdots + x_{\alpha,i}z_{i,\beta}.$$

Thus, with respect to unknowns $a_{s,t}$ $(1 \le s \le i,\ 1 \le t \le j)$, we have at most $hj + ik - hk$ linearly independent equations. Therefore, the number of unknowns not uniquely determined is at least

$$ij - (hj + ik - hk) = (i - h)(j - k),$$

and it is positive if $i > h$ and $j > k$. From this, it follows that A has at least q solutions. □

Lemma 4.6. *Let $i > h$ and $j > k$, and suppose that X, A, Y, W, and Z are the same as those in Lemma 4.5. Let $\boldsymbol{x} := (x_1, x_2, \ldots, x_i) \in (\mathbb{F}_q)^i$ and $\boldsymbol{y} := {}^t(y_1, y_2, \ldots, y_j) \in (\mathbb{F}_q)^j$ be vectors such that: $\boldsymbol{x}$ is not contained in the $\mathbb{F}_q$-vector space generated by row vectors of X; and $\boldsymbol{y}$ is not contained in the $\mathbb{F}_q$-vector space generated by column vectors of Y. Suppose that such X, Y, W, Z, $\boldsymbol{x}$, and $\boldsymbol{y}$ are arbitrarily given, and each entry of A is chosen from $\mathbb{F}_q$ uniformly at random such that $W = XA$ and $Z = AY$. Then, an element $\boldsymbol{x}A\boldsymbol{y} \in \mathbb{F}_q$ cannot be guessed with probability larger than $1/q$.*

Proof. Let $\chi := \{A \mid XA = W, AY = Z\}$ be the set of solutions of A for the simultaneous linear equations, $W = XA$ and $Z = AY$. First, we show the following claims.

Claim 4.1. *Define $\chi_0 := \{A \mid XA = O, AY = O\}$, and let A_1 be a solution in χ. Then, χ_0 is a linear space over $\mathbb{F}_q$ with $\dim \chi_0 \ge 1$, and $\chi = \{A_0 + A_1 \mid A_0 \in \chi_0\}$.*

Proof. It is straightforward to see that χ_0 is a linear space over $\mathbb{F}_q$, and $\dim \chi_0 \geq 1$ follows from the special case of $W = Z = O$ in Proposition 4.5.

For generally given W and Z, let A_1 be an element in χ. For any $A \in \chi$, it holds that $XA = W$ and $AY = Z$, and hence $X(A - A_1) = O$ and $(A - A_1)Y = O$, which implies $A - A_1 \in \chi_0$. Thus, we have $\chi = \{A_0 + A_1 \mid A_0 \in \chi_0\}$. □

Claim 4.2. *Let X and Y be an $h \times i$ matrix and a $j \times k$ matrix, respectively, with $i > h$ and $j > k$. Then, the $\mathbb{F}_q$-linear mapping $f : \chi_0 \to \mathbb{F}_q$ defined by $f(A) := \boldsymbol{x}A\boldsymbol{y}$ is surjective.*

Proof. First, we assume that X and Y are $(i-1) \times i$ matrix and $j \times (j-1)$ matrix, respectively, such that rank $X = i - 1$ and rank $Y = j - 1$. It is obvious that the mapping f is $\mathbb{F}_q$-linear. In addition, since f is $\mathbb{F}_q$-linear, $\mathrm{Im} f$ is a linear subspace of $\mathbb{F}_q$. Therefore, by Claim 4.1, $\dim(\mathrm{Im}\ f)$ is 0 or 1. We will show that $\dim(\mathrm{Im}\ f) = 1$ (i.e., $\mathrm{Im}\ f = \mathbb{F}_q$). To prove this, it is sufficient to show that, for $A, A' \in \chi_0$ with $A \neq A'$, we have $\boldsymbol{x}A\boldsymbol{y} \neq \boldsymbol{x}A'\boldsymbol{y}$. Suppose on the contrary that $\boldsymbol{x}A\boldsymbol{y} = \boldsymbol{x}A'\boldsymbol{y}$. Let $\hat{X} := \begin{pmatrix} X \\ \boldsymbol{x} \end{pmatrix}$ and $\hat{Y} := (Y, \boldsymbol{y})$. Then, since $XA = XA' = O$ and $AY = A'Y = O$, we obtain $\hat{X}A\hat{Y} = \hat{X}A'\hat{Y}$. Since $\hat{X}$ and $\hat{Y}$ are invertible, we have $A = A'$, which implies contradiction. Therefore, f is surjective.

Next, we consider a general case that X and Y are $h \times i$ matrix and $j \times k$ matrix, respectively, with $i > h$ and $j > k$. Let $\tilde{X}$ be an $(i-1) \times i$ matrix such that: rank $\tilde{X} = i - 1$; $\boldsymbol{x}$ is not contained in the $\mathbb{F}_q$-vector space generated by row vectors of $\tilde{X}$; and the $\mathbb{F}_q$-vector space generated by row vectors of $\tilde{X}$ contains the vector space generated by row vectors of X. Similarly, let $\tilde{Y}$ be an $j \times (j-1)$ matrix such that: rank $\tilde{Y} = j - 1$; $\boldsymbol{y}$ is not contained in the $\mathbb{F}_q$-vector space generated by column vectors of $\tilde{Y}$; and the $\mathbb{F}_q$-vector space generated by column vectors of $\tilde{Y}$ contains the vector space generated by column vectors of Y. Letting $\tilde{\chi}_0 := \{A \mid \tilde{X}A = O,\ A\tilde{Y} = O\}$, and we have $\tilde{\chi}_0 \subset \chi_0$. Therefore, it holds that $f : \chi_0 \to \mathbb{F}_q$ defined by $f(A) := \boldsymbol{x}A\boldsymbol{y}$ is surjective, since $f \mid \tilde{\chi}_0$ is surjective as shown by the above paragraph. □

Proof of Lemma 4.6. We show that, if A is chosen from χ uniformly at random, a value of $\boldsymbol{x}A\boldsymbol{y}$ cannot be guessed with probability larger than $1/q$. For proving it, it is sufficient to show that, for every $t \in \mathbb{F}_q$, $\Pr[t = \boldsymbol{x}A\boldsymbol{y}] = 1/q$ if A is chosen from χ uniformly at random. Define $\hat{f} : \chi \to \mathbb{F}_q$ by $\hat{f}(A) := \boldsymbol{x}A\boldsymbol{y}$, and fix some $A_1 \in \chi$. Then, arbitrary $A \in \chi$ is expressed by $A = A_0 + A_1$ ($A_0 \in \chi_0$) by Claim 4.1, and then, $\hat{f}(A) = \boldsymbol{x}A_0\boldsymbol{y} + \boldsymbol{x}A_1\boldsymbol{y} = f(A_0) + \boldsymbol{x}A_1\boldsymbol{y}$. Note that A being chosen from χ uniformly at random is equivalent to that A_0 being chosen from χ_0 uniformly at random. If A_0 is chosen from χ_0 uniformly at random, we have $\Pr[t = f(A_0)] = 1/q$ for every $t \in \mathbb{F}_q$ since f is $\mathbb{F}_q$-linear

and surjective by Claim 4.2. Therefore, since $f(A_0)$ takes every value of $\mathbb{F}_q$ with equal probability and $\boldsymbol{x}A_1\boldsymbol{y}$ is fixed, $\hat{f}(A) = f(A_0) + \boldsymbol{x}A_1\boldsymbol{y}$ takes every value of $\mathbb{F}_q$ with equal probability. □

Lemma 4.7. *The above construction meets* $H(TCK^{(t)}_{i_1,i_2} \mid TMK) = H(TCK^{(t)}_{i_1,i_2})$ *for any* $U_{i_1}, U_{i_2} \in \mathcal{U}$ *and* $t \in \mathcal{T}$.

Proof. Consider the case that TS will guess $tck^{(t)}_{i_1,i_2} = f(U_{i_1}, U_{i_2}) + tmk^*(U_{i_1}, t)$ by using his master key. Since TS knows tmk^*, he can compute $tmk^*(U_{i_1}, t)$. Therefore, he has to guess $f(U_{i_1}, U_{i_2})$. However, by applying $X := O$, $A := A$ and $Y := O$ in Lemma 4.5, there are at least q candidates of A. Then, by applying $\boldsymbol{x} := (1, U_{i_1}, U_{i_1}^2, \ldots, U_{i_1}^\omega)$, $A := A$ and $\boldsymbol{y} := {}^t(1, U_{i_2}, U_{i_2}^2, \ldots, U_{i_2}^\omega)$ in Lemma 4.6, TS cannot guess $f(U_{i_1}, U_{i_2}) = \boldsymbol{x}A\boldsymbol{y}$ with probability larger than $1/q$. On the other hand, it is clear that $H(TCK^{(t)}_{i_1,i_2}) = \log_2 q$. Hence, for any $U_{i_1}, U_{i_2} \in \mathcal{U}$ and $t \in \mathcal{T}$, $H(TCK^{(t)}_{i_1,i_2} \mid TMK) = H(TCK^{(t)}_{i_1,i_2}) = \log_2 q$. □

Lemma 4.8. *The above construction meets* $H(TCK^{(t)}_{i_1,i_2} \mid TUK^{(S)}_{\mathcal{W}}, TUK^{(R)}_{\mathcal{W}}, TI^{(1)}, \ldots, TI^{(\tau)}) = H(TCK^{(t)}_{i_1,i_2})$ *for any* $U_{i_1}, U_{i_2} \in \mathcal{U}$ *and* $\mathcal{W} \in \mathcal{PS}(\mathcal{U}, \omega)$ *such that* $U_{i_1}, U_{i_2} \notin \mathcal{W}$, *and for any* $t \in \mathcal{T}$.

Proof. Without loss of generality, we consider that $\mathcal{W} := \{U_1, \ldots, U_\omega\}$ is a set of colluders such that $U_{i_1}, U_{i_2} \notin \mathcal{W}$, and we write $\boldsymbol{x_i} := (1, U_i, U_i^2, \ldots, U_i^\omega)$ $(1 \leq i \leq n)$. Consider the case that a group of colluders $\mathcal{W}$ not including a targeted receiver will guess $tck^{(t)}_{i_1,i_2} = f(U_{i_1}, U_{i_2}) + tmk^*(U_{i_1}, t)$ by using their secret keys and all time-signals. Since $\mathcal{W}$ can compute tmk^* by all time-signals, $\mathcal{W}$ can correctly obtain $tmk^*(U_{i_1}, t)$. Therefore, the purpose of $\mathcal{W}$ is to guess $f(U_{i_1}, U_{i_2})$. Since $\mathcal{W}$ can calculate $tmk^*(U_l, z)$ $(1 \leq l \leq \omega)$ and hence $tuk^{(S)}_l(y, z) - tmk^*(U_l, z) = f(U_l, y)$ $(1 \leq l \leq \omega)$, $\mathcal{W}$ gets

$$f(U_l, y) = \boldsymbol{x}_l A \begin{pmatrix} 1 \\ y \\ \vdots \\ y^\omega \end{pmatrix},$$

for $1 \leq l \leq \omega$. Thus, $\mathcal{W}$ can know the following matrix:

$$X_U A := \begin{pmatrix} \boldsymbol{x}_1 \\ \boldsymbol{x}_2 \\ \vdots \\ \boldsymbol{x}_\omega \end{pmatrix} A.$$

In addition, $\mathcal{W}$ knows

$$f(x, U_l) = (1, x, \ldots, x^\omega)\ A\ {}^t\boldsymbol{x}_l,$$

for $1 \leq l \leq \omega$ by their secret keys $tuk_{\mathcal{W}}^{(R)}$. Thus, $\mathcal{W}$ can know the following matrix:

$$A\ {}^tX_U = A\ ({}^t\boldsymbol{x}_1, {}^t\boldsymbol{x}_2, \ldots, {}^t\boldsymbol{x}_\omega).$$

By applying $X := X_U$, $A := A$ and $Y := {}^tX_U$ in Lemma 4.5, there are at least q candidates of A. In addition, $\{\boldsymbol{x}_{i_1}, \boldsymbol{x}_1, \ldots, \boldsymbol{x}_\omega\}$ and $\{\boldsymbol{x}_{i_2}, \boldsymbol{x}_1, \ldots, \boldsymbol{x}_\omega\}$ are linearly independent, respectively, since $U_{i_1}, U_{i_2} \notin \mathcal{W}$. Therefore, $\mathcal{W}$ cannot guess $f(U_{i_1}, U_{i_2}) = \boldsymbol{x}_{i_1} A\ {}^t\boldsymbol{x}_{i_2}$ with probability larger than $1/q$ by Lemma 4.6. Thus, we have $H(TCK_{i_1,i_2}^{(t)} \mid TUK_{\mathcal{W}}^{(S)},\ TUK_{\mathcal{W}}^{(R)},\ TI^{(1)}, \ldots,\ TI^{(\tau)}) = \log_2 q$. Hence, for any $U_{i_1},\ U_{i_2} \in \mathcal{U}$ such that $U_{i_1}, U_{i_2} \notin \mathcal{W}$, and for any $t \in \mathcal{T}$, $H(TCK_{i_1,i_2}^{(t)} \mid TUK_{\mathcal{W}}^{(S)}, TUK_{\mathcal{W}}^{(R)}, TI^{(1)},\ \ldots, TI^{(\tau)}) = H(TCK_{i_1,i_2}^{(t)}) = \log_2 q$. □

Lemma 4.9. *The above construction meets* $H(TCK_{i_1,i_2}^{(t)} \mid TUK_{\mathcal{W}}^{(S)}, TUK_{\mathcal{W}}^{(R)}, TI^{(1)}, \ldots, TI^{(t-1)}, TI^{(t+1)}, \ldots, TI^{(\tau)}) = H(TCK_{i_1,i_2}^{(t)})$ *for any* $U_{i_1}, U_{i_2} \in \mathcal{U}$ *and* $\mathcal{W} \in \mathcal{PS}(\mathcal{U}, \omega)$ *such that* $U_{i_1} \notin \mathcal{W}$ *and* $U_{i_2} \in \mathcal{W}$, *and for any* $t \in \mathcal{T}$.

Proof. Without loss of generality, we suppose that $\mathcal{W} := \{U_1, \ldots, U_\omega\}$ is a set of colluders such that $U_{i_1} \notin \mathcal{W}$, U_{i_1} is a targeted sender, U_ω is a targeted receiver, and τ is a specified time. In addition, we write $\boldsymbol{x_i} := (1, U_i, U_i^2, \ldots, U_i^\omega)$ $(1 \leq i \leq n)$ and $\boldsymbol{y_i} := {}^t(1, i, i^2, \ldots, i^{\tau-1})$ $(1 \leq i \leq \tau)$. Consider the case that a group of colluders $\mathcal{W}$ will guess $tck_{i_1,\omega}^{(\tau)} = f(U_{i_1}, U_\omega) + tmk^*(U_{i_1}, \tau)$ by using their secret keys and time-signals at all the time except the specified time. Note that $\mathcal{W}$ can get $f(U_{i_1}, U_\omega)$ since $U_\omega \in \mathcal{W}$. Thus, $\mathcal{W}$ tries to obtain $tmk^*(x, z)$ to know $tmk^*(U_{i_1}, \tau)$. $\mathcal{W}$ can compute $tuk_l^{(S)}(y, z) - f(U_l, z) = tmk^*(U_l, z)$ $(1 \leq l \leq \omega)$, and hence $\mathcal{W}$ gets

$$tmk^*(U_l, z) = \boldsymbol{x}_l B \begin{pmatrix} 1 \\ z \\ \vdots \\ z^{\tau-1} \end{pmatrix},$$

for $1 \leq l \leq \omega$. Thus, $\mathcal{W}$ can know the following matrix:

$$X_U B := \begin{pmatrix} \boldsymbol{x}_1 \\ \boldsymbol{x}_2 \\ \vdots \\ \boldsymbol{x}_\omega \end{pmatrix} B.$$

In addition, $\mathcal{W}$ obtains $tmk^*(x,t) = (1, x, \ldots, x^\omega)B\boldsymbol{y}_t$ for $1 \leq t \leq \tau - 1$ by time-signals at all except the time τ. Thus, $\mathcal{W}$ can know the following matrix:

$$BY_T := B(\boldsymbol{y}_1, \boldsymbol{y}_2, \ldots, \boldsymbol{y}_{\tau-1}).$$

By applying $X := X_U$, $A := B$ and $Y := Y_T$ in Lemma 4.5, there are at least q candidates of B. In addition, $\{\boldsymbol{x}_{i_1}, \boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_\omega\}$ and $\{\boldsymbol{y}_1, \boldsymbol{y}_2, \ldots, \boldsymbol{y}_\tau\}$ are linearly independent, respectively, since $U_{i_1} \notin \mathcal{W}$. Therefore, $\mathcal{W}$ cannot guess $tmk^*(U_{i_1}, \tau) = \boldsymbol{x}_{i_1} B \boldsymbol{y}_\tau$ with probability larger than $1/q$ from Lemma 4.6. Thus, we have $H(TCK^{(\tau)}_{i_1,\omega} \mid TUK^{(S)}_{\mathcal{W}}, TUK^{(R)}_{\mathcal{W}}, TI^{(1)}, TI^{(2)}, \ldots, TI^{(\tau-1)}) = \log_2 q$. Hence, in general, for any $U_{i_1}, U_{i_2} \in \mathcal{U}$ such that $U_{i_1} \notin \mathcal{W}$ and $U_{i_2} \in \mathcal{W}$, and for any $t \in \mathcal{T}$, it holds that

$$\begin{aligned} &H(TCK^{(t)}_{i_1,i_2} \mid TUK^{(S)}_{\mathcal{W}}, TUK^{(R)}_{\mathcal{W}}, TI^{(1)}, \ldots, TI^{(t-1)}, TI^{(t+1)}, \ldots, TI^{(\tau)}) \\ &\quad = H(TCK^{(t)}_{i_1,i_2}) = \log_2 q. \end{aligned}$$

□

Proof of Theorem 4.2. It follows that our construction satisfies the conditions (1)–(3) in Definition 4.2 from the above lemmas. Finally, it is straightforward to see that the construction satisfies all the lower bounds in Theorem 4.1 with equalities. Therefore, the above construction is optimal. □

4.3 Timed-Release Encryption

In this section, we show a model and a security formalization of TRE with information-theoretic security. We also show that TRE can be constructed from TR-KA and the one-time pad in a generic and simple way. In addition, we derive tight lower bounds on size of secrets for entities and size of time-signals required for TRE.

4.3.1 Model and Security Definition

We propose a model and a security definition of TRE, based on that of TR-PKE with computational security and that of SKE with information-theoretic security. Formally, we give a definition of TRE in the TI model as in the case of TR-KA.

Definition 4.4 (TRE). *A TRE Π_{TRE} involves $n+2$ entities, $TA, U_1, U_2, \ldots, U_n$ and TS, and consists of a four-tuple of algorithms (EGen, EExt, Enc, Dec) with six spaces, $\mathcal{C}, \mathcal{M}_{\mathrm{E}}, \mathcal{USK}, \mathcal{EMK}, \mathcal{T}$, and $\mathcal{ETI}$, where all of the above algorithms except EGen are deterministic and all of the above spaces are finite. In addition, Π_{TRE} is executed with four phases as follows.*

- ***Notation:***

- *Entities: TA, U_i $(1 \leq i \leq n)$, TS, and $\mathcal{U}$ are the same as those in Definition 4.1.*
- *Spaces: $\mathcal{T}$ is the same as that in Definition 4.1. $\mathcal{C}$ is a set of possible ciphertexts, $\mathcal{M}_{\mathrm{E}}$ is a set of possible plaintexts with a probability distribution P_M, $\mathcal{EMK}$ is a set of possible master keys. $\mathcal{ETI}^{(t)}$ is a set of possible time-signals at time $t \in \mathcal{T}$. Let $\mathcal{ETI} := \bigcup_{t=1}^{\tau} \mathcal{ETI}^{(t)}$. Also, $\mathcal{EK}_i$ is a set of possible encryption keys for U_i, $\mathcal{DK}_i$ is a set of possible decryption keys for U_i, and $\mathcal{USK}_i := \mathcal{EK}_i \times \mathcal{DK}_i$ is a set of possible secret keys for U_i. Let $\mathcal{EK} := \bigcup_{i=1}^{n} \mathcal{EK}_i$, $\mathcal{DK} := \bigcup_{i=1}^{n} \mathcal{DK}_i$ and $\mathcal{USK} := \bigcup_{i=1}^{n} \mathcal{USK}_i$.*
- *Algorithms: EGen is a key generation algorithm which on input a security parameter 1^{κ}, outputs each user's secret key and a server's master key, EExt: $\mathcal{EMK} \times \mathcal{T} \to \mathcal{ETI}$ is a time-signal generation algorithm for TS, Enc: $\mathcal{M}_{\mathrm{E}} \times \mathcal{EK} \times \mathcal{T} \times \mathcal{U} \to \mathcal{C}$ is an encryption algorithm, and Dec: $\mathcal{C} \times \mathcal{DK} \times \mathcal{ETI} \times \mathcal{U} \to \mathcal{M}_{\mathrm{E}}$ is a decryption algorithm.*

1. ***Key Generation and Distribution.*** *In the initial phase, TA generates the following keys by using EGen: a master key $emk^* \in \mathcal{EMK}$ for TS; a secret key $usk_i = (ek_i, dk_i) \in \mathcal{USK}_i$ for U_i $(i = 1, 2, \ldots, n)$. These keys are distributed to corresponding entities via secure channels. After distributing these keys, TA deletes them from his memory. And, TS and U_i keep their keys secret, respectively.*

2. ***Time-signal Generation.*** *For broadcasting a time-signal at each time, TS generates a time-signal $emk^{(t)} \leftarrow \mathsf{EExt}(emk^*, t) \in \mathcal{ETI}^{(t)}$ by using a master key $emk^* \in \mathcal{EMK}$ and time $t \in \mathcal{T}$. Then, TS broadcasts it to all users via a broadcast channel.*

3. ***Encryption.*** *U_{i_1} specifies time t when U_{i_2} can decrypt a ciphertext, and then U_{i_1} computes a ciphertext, $c_{i_1,i_2}^{(t)} \leftarrow \mathsf{Enc}(m, ek_{i_1}, t, U_{i_2}) \in \mathcal{C}$, by a plaintext $m \in \mathcal{M}_{\mathrm{E}}$, an encryption key $ek_{i_1} \in \mathcal{EK}_i$, the specified time t and the identity U_{i_2}. And, U_{i_1} sends a pair of the ciphertext and the specified time, $(c_{i_1,i_2}^{(t)}, t)$, to U_{i_2} via an authenticated channel.*

4. ***Decryption.*** *Suppose that U_{i_2} has received $(c_{i_1,i_2}^{(t)}, t)$ from U_{i_1}. After receiving a time-signal $emk^{(t)}$ at the specified time t, U_{i_2} recovers $m \leftarrow \mathsf{Dec}(c_{i_1,i_2}^{(t)}, dk_{i_2}, emk^{(t)}, U_{i_1})$ by a ciphertext $c_{i_1,i_2}^{(t)}$, a decryption key $dk_{i_2} \in \mathcal{DK}_i$, a time-signal $emk^{(t)}$, and the identity U_{i_1}.*

In the model of TRE, we require the following equation holds: For all possible $\kappa \in \mathbb{N}$, $((ek_1, dk_1), \ldots, (ek_n, dk_n), emk^*) \leftarrow \mathsf{Setup}(1^{\kappa}) \in \prod_{i=1}^{n} (\mathcal{EK}_i \times \mathcal{DK}_i) \times \mathcal{EMK}$, and $t \in \mathcal{T}$, $i_1, i_2 \in [n]$, we have

$$m \leftarrow \mathsf{Dec}(\mathsf{Enc}(m, ek_{i_1}, t, U_{i_2}), dk_{i_2}, \mathsf{EExt}(emk^*, t), U_{i_1}).$$

The above requirement means correctness of TRE.

Next, we provide a security definition of TRE based on the idea of timed-release security and the traditional encryption with information-theoretic security. The choice of possible colluders $\mathcal{W} \in \mathcal{PS}(\mathcal{U}, \omega)$ is the same as that in TR-KA. For a set of colluders $\mathcal{W} = \{U_{l_1}, U_{l_2}, \ldots, U_{l_j}\} \in \mathcal{PS}(\mathcal{U}, \omega)$, $\mathcal{EK}_{\mathcal{W}} := \mathcal{EK}_{l_1} \times \mathcal{EK}_{l_2} \times \cdots \times \mathcal{EK}_{l_j}$ is a set of $\mathcal{W}$'s encryption keys, and $\mathcal{DK}_{\mathcal{W}} := \mathcal{DK}_{l_1} \times \mathcal{DK}_{l_2} \times \cdots \times \mathcal{DK}_{l_j}$ is a set of $\mathcal{W}$'s decryption keys. Also, let $\mathcal{C}^{(t)}_{i_1,i_2}$ be a finite set of possible ciphertexts sent from U_{i_1} to U_{i_2} such that it can be decrypted at the time t. Furthermore, let M, $C^{(t)}_{i_1,i_2}$, EMK, $EK_{\mathcal{W}}$, $DK_{\mathcal{W}}$, and $ETI^{(1)}, \ldots, ETI^{(\tau)}$ be random variables which take values in $\mathcal{M}_{\mathrm{E}}$, $\mathcal{C}^{(t)}_{i_1,i_2}$, $\mathcal{EMK}$, $\mathcal{EK}_{\mathcal{W}}$, $\mathcal{DK}_{\mathcal{W}}$, and $\mathcal{ETI}^{(1)}, \ldots, \mathcal{ETI}^{(\tau)}$, respectively.

Similarly as in Definition 4.2 we consider the following three types of security notions for TRE: (1) A dishonest time-server cannot obtain any information on an underlying plaintext from a target ciphertext transmitted on the channel; (2) No information on an underlying plaintext from a target ciphertext is obtained by any colluding group $\mathcal{W}$ not including a legitimate receiver, even if $\mathcal{W}$ obtains time-signals at all the time; (3) No information on an underlying plaintext from a target ciphertext is obtained by any colluding group $\mathcal{W}$ including a legitimate (but dishonest) receiver, even if $\mathcal{W}$ obtains time-signals at all the time except the specified time.

The formalizations of the above security notions for TRE are given as follows.

Definition 4.5. *Let Π_{TRE} be TRE. Π_{TRE} is said to be (n, ω, τ)-secure if the following conditions are satisfied:*

(1) For any $U_{i_1}, U_{i_2} \in \mathcal{U}$ and any $t \in \mathcal{T}$, it holds that

$$H(M \mid C^{(t)}_{i_1,i_2}, EMK) = H(M).$$

(2) For any $\mathcal{W} \in \mathcal{PS}(\mathcal{U}, \omega)$, $U_{i_1}, U_{i_2} \in \mathcal{U}$ such that $U_{i_1}, U_{i_2} \notin \mathcal{W}$, and for any $t \in \mathcal{T}$, it holds that

$$H(M \mid C^{(t)}_{i_1,i_2}, EK_{\mathcal{W}}, DK_{\mathcal{W}}, ETI^{(1)}, \ldots, ETI^{(\tau)}) = H(M).$$

(3) For any $\mathcal{W} \in \mathcal{PS}(\mathcal{U}, \omega)$, $U_{i_1}, U_{i_2} \in \mathcal{U}$ such that $U_{i_1} \notin \mathcal{W}$ and $U_{i_2} \in \mathcal{W}$, for any $t \in \mathcal{T}$, it holds that

$$H(M \mid C^{(t)}_{i_1,i_2}, EK_{\mathcal{W}}, DK_{\mathcal{W}}, ETI^{(1)}, \ldots, ETI^{(t-1)}, ETI^{(t+1)}, \ldots, ETI^{(\tau)}) = H(M).$$

4.3.2 Lower Bounds

We derive lower bounds on size of secrets for entities and size of time-signals required for secure TRE as follows.

Theorem 4.3. *Let Π_{TRE} be an (n,ω,τ)-secure TRE. Then, for any $i \in [n]$ and $t \in \mathcal{T}$, we have*

$$(i)\ H(DK_i) \geq (\omega+1)H(M), \qquad (ii)\ H(EK_i) \geq (\tau+\omega)H(M),$$
$$(iii)\ H(ETI^{(t)}) \geq (\omega+1)H(M), \qquad (iv)\ H(EMK) \geq \tau(\omega+1)H(M).$$

Proof. Although the proof is similar to that of Theorem 4.1, there are several points in the proof which are more technically complicated than that of Theorem 4.1. The proof of Theorem 4.3 follows following lemmas. In this proof, for any $i,j \in [n]$ and any $t \in [\tau]$, $M_{i,j}^{(t)}$ denotes the random variable which takes plaintexts to be sent from U_i to U_j at time t, and $M_{i,j}^{(t)}$ is i.i.d. according to P_M.

Lemma 4.10. *$H(DK_i) \geq (\omega+1)H(M)$ for any $i \in [n]$.*

Proof. For arbitrary $i \in [n]$, we take a subset $B := \{l_1, l_2, \ldots, l_{\omega+1}\} \subset [n]$ of indices of users such that $i \notin B$. Let $D_k := (l_k, i)$ with $1 \leq k \leq \omega+1$. Then, we have

$$\begin{aligned}
H(DK_i) \geq & H(DK_i \mid ETI^{(t)}, C_{D_1}^{(t)}, C_{D_2}^{(t)}, \ldots, C_{D_{\omega+1}}^{(t)}) \\
\geq & I(M_{D_1}^{(t)}, M_{D_2}^{(t)}, \ldots, M_{D_{\omega+1}}^{(t)}; DK_i \mid ETI^{(t)}, C_{D_1}^{(t)}, C_{D_2}^{(t)}, \ldots, C_{D_{\omega+1}}^{(t)}) \\
= & H(M_{D_1}^{(t)}, M_{D_2}^{(t)}, \ldots, M_{D_{\omega+1}}^{(t)} \mid ETI^{(t)}, C_{D_1}^{(t)}, C_{D_2}^{(t)}, \ldots, C_{D_{\omega+1}}^{(t)}) \\
& - H(M_{D_1}^{(t)}, M_{D_2}^{(t)}, \ldots, M_{D_{\omega+1}}^{(t)} \mid ETI^{(t)}, DK_i, C_{D_1}^{(t)}, C_{D_2}^{(t)}, \ldots, C_{D_{\omega+1}}^{(t)}) \\
= & H(M_{D_1}^{(t)}, M_{D_2}^{(t)}, \ldots, M_{D_{\omega+1}}^{(t)} \mid ETI^{(t)}, C_{D_1}^{(t)}, C_{D_2}^{(t)}, \ldots, C_{D_{\omega+1}}^{(t)}) \\
= & \sum_{k=1}^{\omega+1} H(M_{D_k}^{(t)} \mid ETI^{(t)}, M_{D_1}^{(t)}, M_{D_2}^{(t)}, \ldots, M_{D_{k-1}}^{(t)}, C_{D_1}^{(t)}, C_{D_2}^{(t)}, \ldots, C_{D_{\omega+1}}^{(t)}) \\
= & \sum_{k=1}^{\omega+1} H(M_{D_k}^{(t)}) \qquad (4.6) \\
= & (\omega+1)H(M),
\end{aligned}$$

where Eq. (4.6) is shown by following: Let $\mathcal{W}_k := \{l_1, l_2, \ldots, l_{k-1}, l_{k+1}, \ldots, l_{\omega+1}\}$ for each k with $1 \leq k \leq \omega+1$. Then, we have

$$\begin{aligned}
& H(M_{D_k}^{(t)} \mid C_{D_k}^{(t)}, EK_{\mathcal{W}_k}, ETI^{(t)}) \qquad (4.7) \\
= & H(M_{D_k}^{(t)} \mid C_{D_k}^{(t)}, EK_{\mathcal{W}_k}, ETI^{(t)}, M_{D_1}^{(t)}, \ldots, M_{D_{k-1}}^{(t)}, M_{D_{k+1}}^{(t)}, \ldots, M_{D_{\omega+1}}^{(t)})
\end{aligned}$$

$$\begin{aligned}
&\leq H(M_{D_k}^{(t)} \mid M_{D_1}^{(t)}, \ldots, M_{D_{k-1}}^{(t)}, C_{D_1}^{(t)}, \ldots, C_{D_{\omega+1}}^{(t)}, ETI^{(t)}) \\
&\leq H(M_{D_k}^{(t)}).
\end{aligned}$$

And, we have $H(M_{D_k}^{(t)} \mid C_{D_k}^{(t)}, EK_{\mathcal{W}_k}, ETI^{(t)}) = H(M_{D_k}^{(t)})$ from the condition (2) in Definition 4.5. Therefore, we have $H(M_{D_k}^{(t)} \mid M_{D_1}^{(t)}, \ldots, M_{D_{k-1}}^{(t)}, C_{D_1}^{(t)}, \ldots, C_{D_{\omega+1}}^{(t)}, ETI^{(t)}) = H(M_{D_k}^{(t)})$. □

Lemma 4.11. *$H(EK_i) \geq (\tau + \omega)H(M)$ for any $i \in [n]$.*

Proof. For arbitrary $i \in [n]$, we take a subset $B := \{l_1, l_2, \ldots, l_{\omega+1}\} \subset [n]$ of indices of users such that $i \notin B$. Let $D_k := (i, l_k)$ with $1 \leq k \leq \omega + 1$. Also, let $F_k^{(t)} := (M_{D_k}^{(1)}, M_{D_k}^{(2)}, \ldots, M_{D_k}^{(t)})$, $G_k^{(t)} := (M_{D_1}^{(t)}, M_{D_2}^{(t)}, \ldots, M_{D_k}^{(t)})$, $FC_k^{(t)} := (C_{D_k}^{(1)}, C_{D_k}^{(2)}, \ldots, C_{D_k}^{(t)})$, and $GC_k^{(t)} := (C_{D_1}^{(t)}, C_{D_2}^{(t)}, \ldots, C_{D_k}^{(t)})$ for $1 \leq k \leq \omega + 1$ and $1 \leq t \leq \tau$. Then, we have

$$\begin{aligned}
&H(EK_i) \\
=&H(EK_i \mid F_1^{(\tau-1)}, G_{\omega+1}^{(\tau)}) \\
\geq&I(EK_i; FC_1^{(\tau-1)}, GC_{\omega+1}^{(\tau)} \mid F_1^{(\tau-1)}, G_{\omega+1}^{(\tau)}) \\
=&H(FC_1^{(\tau-1)}, GC_{\omega+1}^{(\tau)} \mid F_1^{(\tau-1)}, G_{\omega+1}^{(\tau)}) \\
&\qquad - H(FC_1^{(\tau-1)}, GC_{\omega+1}^{(\tau)} \mid F_1^{(\tau-1)}, G_{\omega+1}^{(\tau)}, EK_i) \\
=&H(FC_1^{(\tau-1)}, GC_{\omega+1}^{(\tau)} \mid F_1^{(\tau-1)}, G_{\omega+1}^{(\tau)}) \qquad (4.8) \\
=&H(FC_1^{(\tau-1)} \mid F_1^{(\tau-1)}, G_{\omega+1}^{(\tau)}) + H(GC_{\omega+1}^{(\tau)} \mid F_1^{(\tau-1)}, G_{\omega+1}^{(\tau)}, FC_1^{(\tau-1)}) \\
=&\sum_{t=1}^{\tau-1} H(C_{D_1}^{(t)} \mid F_1^{(\tau-1)}, G_{\omega+1}^{(\tau)}, C_{D_1}^{(1)}, \ldots, C_{D_1}^{(t-1)}) \\
&\qquad + \sum_{j=1}^{\omega+1} H(C_{D_j}^{(\tau)} \mid F_1^{(\tau-1)}, G_{\omega+1}^{(\tau)}, FC_1^{(\tau-1)}, C_{D_1}^{(\tau)}, \ldots, C_{D_{j-1}}^{(\tau)}) \\
\geq&(\tau + \omega)H(M), \qquad (4.9)
\end{aligned}$$

where Eq. (4.8) follows from Enc algorithm (i.e., $H(FC_1^{(\tau-1)}, GC_{\omega+1}^{(\tau)} \mid F_1^{(\tau-1)}, G_{\omega+1}^{(\tau)}, EK_i) = 0$), and Eq. (4.9) follows from the following claims:

Claim 4.3. *$H(C_{D_1}^{(t)} \mid F_1^{(\tau-1)}, G_{\omega+1}^{(\tau)}, C_{D_1}^{(1)}, \ldots, C_{D_1}^{(t-1)}) \geq H(M_{D_1}^{(\tau)})$ for $1 \leq t \leq \tau - 1$.*

Proof. Let $\tilde{F}_1^{(t,\tau-1)} := (M_{D_1}^{(1)}, M_{D_1}^{(2)}, \ldots, M_{D_1}^{(t-1)}, M_{D_1}^{(t+1)}, \ldots, M_{D_1}^{(\tau-1)})$.

First, since $M_{D_1}^{(t)}$ is independent of $(\tilde{F}_1^{(t,\tau-1)}, G_{\omega+1}^{(\tau)}, C_{D_1}^{(1)}, \ldots, C_{D_1}^{(t-1)})$ and $C_{D_1}^{(t)}$ (see Definition 4.5), we have

$$\begin{aligned} &H(C_{D_1}^{(t)} \mid F_1^{(\tau-1)}, G_{\omega+1}^{(\tau)}, C_{D_1}^{(1)}, \ldots, C_{D_1}^{(t-1)}) \\ &\quad = H(C_{D_1}^{(t)} \mid \tilde{F}_1^{(t,\tau-1)}, G_{\omega+1}^{(\tau)}, C_{D_1}^{(1)}, \ldots, C_{D_1}^{(t-1)}). \end{aligned} \tag{4.10}$$

Next, we have

$$\begin{aligned} &H(C_{D_1}^{(t)}, M_{D_1}^{(t)}, EK_i, DK_{l_1}, ETI^{(t)} \mid \tilde{F}_1^{(t,\tau-1)}, G_{\omega+1}^{(\tau)}, C_{D_1}^{(1)}, \ldots, C_{D_1}^{(t-1)}) \\ =&H(M_{D_1}^{(t)}, EK_i, DK_{l_1}, ETI^{(t)} \mid \tilde{F}_1^{(t,\tau-1)}, G_{\omega+1}^{(\tau)}, C_{D_1}^{(1)}, \ldots, C_{D_1}^{(t-1)}) \\ &\quad + H(C_{D_1}^{(t)} \mid F_1^{(\tau-1)}, G_{\omega+1}^{(\tau)}, C_{D_1}^{(1)}, \ldots, C_{D_1}^{(t-1)}, EK_i, DK_{l_1}, ETI^{(t)}) \\ =&H(M_{D_1}^{(t)}, EK_i, DK_{l_1}, ETI^{(t)} \mid \tilde{F}_1^{(t,\tau-1)}, G_{\omega+1}^{(\tau)}, C_{D_1}^{(1)}, \ldots, C_{D_1}^{(t-1)}), \end{aligned} \tag{4.11}$$

where Eq. (4.11) follows from Enc algorithm in Definition 4.4 (i.e., $H(C_{i,l_1}^{(t)} \mid M_{i,l_1}^{(t)}, EK_i) = 0$).

On the other hand, we have

$$\begin{aligned} &H(C_{D_1}^{(t)}, M_{D_1}^{(t)}, EK_i, DK_{l_1}, ETI^{(t)} \mid \tilde{F}_1^{(t,\tau-1)}, G_{\omega+1}^{(\tau)}, C_{D_1}^{(1)}, \ldots, C_{D_1}^{(t-1)}) \\ =&H(C_{D_1}^{(t)}, EK_i, DK_{l_1}, ETI^{(t)} \mid \tilde{F}_1^{(t,\tau-1)}, G_{\omega+1}^{(\tau)}, C_{D_1}^{(1)}, \ldots, C_{D_1}^{(t-1)}) \\ &\quad + H(M_{D_1}^{(t)} \mid \tilde{F}_1^{(t,\tau-1)}, G_{\omega+1}^{(\tau)}, C_{D_1}^{(1)}, \ldots, C_{D_1}^{(t)}, EK_i, DK_{l_1}, ETI^{(t)}) \\ =&H(C_{D_1}^{(t)}, EK_i, DK_{l_1}, ETI^{(t)} \mid \tilde{F}_1^{(t,\tau-1)}, G_{\omega+1}^{(\tau)}, C_{D_1}^{(1)}, \ldots, C_{D_1}^{(t-1)}), \end{aligned} \tag{4.12}$$

where Eq. (4.12) follows from Dec algorithm in Definition 4.4 (i.e., $H(M_{i,l_1}^{(t)} \mid C_{i,l_1}^{(t)}, DK_{l_1}, ETI^{(t)}) = 0$).

Therefore, we have

$$\begin{aligned} &H(C_{D_1}^{(t)} \mid \tilde{F}_1^{(t,\tau-1)}, G_{\omega+1}^{(\tau)}, C_{D_1}^{(1)}, \ldots, C_{D_1}^{(t-1)}) \\ &\quad + H(EK_i, DK_{l_1}, ETI^{(t)} \mid \tilde{F}_1^{(t,\tau-1)}, G_{\omega+1}^{(\tau)}, C_{D_1}^{(1)}, \ldots, C_{D_1}^{(t-1)}) \\ \geq&H(C_{D_1}^{(t)}, EK_i, DK_{l_1}, ETI^{(t)} \mid \tilde{F}_1^{(t,\tau-1)}, G_{\omega+1}^{(\tau)}, C_{D_1}^{(1)}, \ldots, C_{D_1}^{(t-1)}) \\ =&H(M_{D_1}^{(t)}, EK_i, DK_{l_1}, ETI^{(t)} \mid \tilde{F}_1^{(t,\tau-1)}, G_{\omega+1}^{(\tau)}, C_{D_1}^{(1)}, \ldots, C_{D_1}^{(t-1)}) \qquad (4.13) \\ =&H(M_{D_1}^{(t)}) + H(EK_i, DK_{l_1}, ETI^{(t)} \mid \tilde{F}_1^{(t,\tau-1)}, G_{\omega+1}^{(\tau)}, C_{D_1}^{(1)}, \ldots, C_{D_1}^{(t-1)}), \qquad (4.14) \end{aligned}$$

where Eq. (4.13) follows from Eqs. (4.11) and (4.12), and Eq. (4.14) follows from that $M_{D_1}^{(t)}$ is independent of $(EK_i, DK_{l_1}, ETI^{(t)}, \tilde{F}_1^{(t,\tau-1)}, G_{\omega+1}^{(\tau)}, C_{D_1}^{(1)}, \ldots, C_{D_1}^{(t-1)})$.

Hence, we have

$$H(C_{D_1}^{(t)} \mid \tilde{F}_1^{(t,\tau-1)}, G_{\omega+1}^{(\tau)}, C_{D_1}^{(1)}, \ldots, C_{D_1}^{(t-1)}) \geq H(M_{D_1}^{(t)}). \tag{4.15}$$

Finally, from Eqs. (4.10) and (4.15), we have $H(C_{D_1}^{(t)} \mid F_1^{(\tau-1)}, G_{\omega+1}^{(\tau)}, C_{D_1}^{(1)}, \ldots, C_{D_1}^{(t-1)}) \geq H(M_{D_1}^{(t)})$ for $1 \leq t \leq \tau - 1$. □

Claim 4.4. $H(C_{D_j}^{(\tau)} \mid F_1^{(\tau-1)}, G_{\omega+1}^{(\tau)}, FC_1^{(\tau-1)}, C_{D_1}^{(\tau)}, \ldots, C_{D_{j-1}}^{(\tau)}) \geq H(M_{D_j}^{(\tau)})$ *for* $1 \leq j \leq \omega+1$.

Proof. We can prove this lemma in a similar way to the proof of Claim 4.3. Let $\tilde{G}_{j,\omega+1}^{(\tau)} := (M_{D_1}^{(\tau)}, M_{D_2}^{(\tau)}, \ldots, M_{D_{j-1}}^{(\tau)}, M_{D_{j+1}}^{(\tau)}, \ldots, M_{D_{\omega+1}}^{(\tau)})$.

First, since $M_{D_j}^{(\tau)}$ is independent of $(F_1^{(\tau-1)}, \tilde{G}_{j,\omega+1}^{(\tau)}, FC_1^{(\tau-1)}, C_{D_1}^{(\tau)}, \ldots, C_{D_{j-1}}^{(\tau)})$ and $C_{D_j}^{(\tau)}$ (see Definition 4.5), we have

$$\begin{aligned} &H(C_{D_j}^{(\tau)} \mid F_1^{(\tau-1)}, G_{\omega+1}^{(\tau)}, FC_1^{(\tau-1)}, C_{D_1}^{(\tau)}, \ldots, C_{D_{j-1}}^{(\tau)}) \\ &\quad = H(C_{D_j}^{(\tau)} \mid F_1^{(\tau-1)}, \tilde{G}_{j,\omega+1}^{(\tau)}, FC_1^{(\tau-1)}, C_{D_1}^{(\tau)}, \ldots, C_{D_{j-1}}^{(\tau)}). \end{aligned} \tag{4.16}$$

Next, we have

$$\begin{aligned} &H(C_{D_j}^{(\tau)}, M_{D_j}^{(\tau)}, EK_i, DK_{l_j}, ETI^{(\tau)} \mid F_1^{(\tau-1)}, \tilde{G}_{j,\omega+1}^{(\tau)}, FC_1^{(\tau-1)}, C_{D_1}^{(\tau)}, \ldots, C_{D_{j-1}}^{(\tau)}) \\ =&H(M_{D_j}^{(\tau)}, EK_i, DK_{l_j}, ETI^{(\tau)} \mid F_1^{(\tau-1)}, \tilde{G}_{j,\omega+1}^{(\tau)}, FC_1^{(\tau-1)}, C_{D_1}^{(\tau)}, \ldots, C_{D_{j-1}}^{(\tau)}) \\ &+ H(C_{D_j}^{(\tau)} \mid F_1^{(\tau-1)}, G_{\omega+1}^{(\tau)}, FC_1^{(\tau-1)}, C_{D_1}^{(\tau)}, \ldots, C_{D_{j-1}}^{(\tau)}, M_{D_j}^{(t)}, EK_i, DK_{l_j}, ETI^{(\tau)}) \\ =&H(M_{D_j}^{(\tau)}, EK_i, DK_{l_j}, ETI^{(\tau)} \mid F_1^{(\tau-1)}, \tilde{G}_{j,\omega+1}^{(\tau)}, FC_1^{(\tau-1)}, C_{D_1}^{(\tau)}, \ldots, C_{D_{j-1}}^{(\tau)}), \end{aligned} \tag{4.17}$$

where Eq. (4.17) follows from Enc algorithm in Definition 4.4 (i.e., $H(C_{i,l_j}^{(\tau)} \mid M_{i,l_j}^{(\tau)}, EK_i) = 0$).

On the other hand, we have

$$\begin{aligned} &H(C_{D_j}^{(\tau)}, M_{D_j}^{(\tau)}, EK_i, DK_{l_j}, ETI^{(\tau)} \mid F_1^{(\tau-1)}, \tilde{G}_{j,\omega+1}^{(\tau)}, FC_1^{(\tau-1)}, C_{D_1}^{(\tau)}, \ldots, C_{D_{j-1}}^{(\tau)}) \\ =&H(C_{D_j}^{(\tau)}, EK_i, DK_{l_j}, ETI^{(\tau)} \mid F_1^{(\tau-1)}, \tilde{G}_{j,\omega+1}^{(\tau)}, FC_1^{(\tau-1)}, C_{D_1}^{(\tau)}, \ldots, C_{D_{j-1}}^{(\tau)}) \\ &+ H(M_{D_j}^{(\tau)} \mid F_1^{(\tau-1)}, \tilde{G}_{j,\omega+1}^{(\tau)}, FC_1^{(\tau-1)}, C_{D_1}^{(\tau)}, \ldots, C_{D_j}^{(\tau)}, EK_i, DK_{l_j}, ETI^{(\tau)}) \\ =&H(C_{D_j}^{(\tau)}, EK_i, DK_{l_j}, ETI^{(\tau)} \mid F_1^{(\tau-1)}, \tilde{G}_{j,\omega+1}^{(\tau)}, FC_1^{(\tau-1)}, C_{D_1}^{(\tau)}, \ldots, C_{D_{j-1}}^{(\tau)}), \end{aligned} \tag{4.18}$$

where Eq. (4.18) follows from Dec algorithm in Definition 4.4 (i.e., $H(M_{i,l_j}^{(\tau)} \mid C_{i,l_j}^{(\tau)}, DK_{l_j}, ETI^{(\tau)}) = 0$).

Therefore, we have

$$\begin{aligned} &H(C_{D_j}^{(\tau)} \mid F_1^{(\tau-1)}, \tilde{G}_{j,\omega+1}^{(\tau)}, FC_1^{(\tau-1)}, C_{D_1}^{(\tau)}, \ldots, C_{D_{j-1}}^{(\tau)}) \\ &\quad + H(EK_i, DK_{l_j}, ETI^{(\tau)} \mid F_1^{(\tau-1)}, \tilde{G}_{j,\omega+1}^{(\tau)}, FC_1^{(\tau-1)}, C_{D_1}^{(\tau)}, \ldots, C_{D_{j-1}}^{(\tau)}) \\ \geq&H(C_{D_j}^{(\tau)}, EK_i, DK_{l_j}, ETI^{(\tau)} \mid F_1^{(\tau-1)}, \tilde{G}_{j,\omega+1}^{(\tau)}, FC_1^{(\tau-1)}, C_{D_1}^{(\tau)}, \ldots, C_{D_{j-1}}^{(\tau)}) \end{aligned}$$

$$=H(M_{D_j}^{(\tau)}, EK_i, DK_{l_j}, ETI^{(\tau)} \mid F_1^{(\tau-1)}, \tilde{G}_{j,\omega+1}^{(\tau)}, FC_1^{(\tau-1)}, C_{D_1}^{(\tau)}, \ldots, C_{D_{j-1}}^{(\tau)}) \tag{4.19}$$

$$=H(M_{D_j}^{(\tau)}) + H(EK_i, DK_{l_j}, ETI^{(\tau)} \mid F_1^{(\tau-1)}, \tilde{G}_{j,\omega+1}^{(\tau)}, FC_1^{(\tau-1)}, C_{D_1}^{(\tau)}, \ldots, C_{D_{j-1}}^{(\tau)}), \tag{4.20}$$

where Eq. (4.19) follows from Eqs. (4.18) and (4.17), and Eq. (4.20) follows from that $M_{D_j}^{(\tau)}$ is independent of $(EK_i, DK_{l_j}, ETI^{(\tau)}, F_1^{(\tau-1)}, \tilde{G}_{j,\omega+1}^{(\tau)}, FC_1^{(\tau-1)}, C_{D_1}^{(\tau)}, \ldots, C_{D_{i-1}}^{(\tau)})$.

Hence, we have

$$H(C_{D_j}^{(\tau)} \mid F_1^{(\tau-1)}, \tilde{G}_{j,\omega+1}^{(\tau)}, FC_1^{(\tau-1)}, C_{D_1}^{(\tau)}, \ldots, C_{D_{j-1}}^{(\tau)}) \geq H(M_{D_j}^{(\tau)}). \tag{4.21}$$

Finally, from Eqs. (4.16) and (4.21), we have $H(C_{D_j}^{(\tau)} \mid F_1^{(\tau-1)}, G_{\omega+1}^{(\tau)}, FC_1^{(\tau-1)}, C_{D_1}^{(\tau)}, \ldots, C_{D_{j-1}}^{(\tau)}) \geq H(M_{D_j}^{(\tau)})$ for $1 \leq j \leq \omega+1$. □

Proof of Lemma 4.11: Now, the proof of Lemma 4.11 is completed. □

Lemma 4.12. *$H(ETI^{(t)} \mid ETI^{(1)}, \ldots, ETI^{(t-1)}) \geq (\omega+1)H(M)$ for any $t \in \mathcal{T}$. In particular, $H(ETI^{(t)}) \geq (\omega+1)H(M)$ for any $t \in \mathcal{T}$.*

Proof. For arbitrary $i \in [n]$, we take a subset $B := \{l_1, l_2, \ldots, l_{\omega+1}\} \subset [n]$ of indices of users such that $i = l_1$. Let $D_k := (l_k, i)$ with $1 \leq k \leq \omega+1$. Then, we have

$$\begin{aligned}
&H(ETI^{(t)} \mid ETI^{(1)}, \ldots, ETI^{(t-1)}) \\
&\geq H(ETI^{(t)} \mid C_{D_1}^{(t)}, \ldots, C_{D_{\omega+1}}^{(t)}, DK_i, ETI^{(1)}, \ldots, ETI^{(t-1)}) \\
&\geq I(M_{D_1}^{(t)}, \ldots, M_{D_{\omega+1}}^{(t)}; ETI^{(t)} \mid C_{D_1}^{(t)}, \ldots, C_{D_{\omega+1}}^{(t)}, DK_i, ETI^{(1)}, \ldots, ETI^{(t-1)}) \\
&= H(M_{D_1}^{(t)}, \ldots, M_{D_{\omega+1}}^{(t)} \mid C_{D_1}^{(t)}, \ldots, C_{D_{\omega+1}}^{(t)}, DK_i, ETI^{(1)}, \ldots, ETI^{(t-1)}) \\
&\quad - H(M_{D_1}^{(t)}, \ldots, M_{D_{\omega+1}}^{(t)} \mid C_{D_1}^{(t)}, \ldots, C_{D_{\omega+1}}^{(t)}, DK_i, ETI^{(1)}, \ldots, ETI^{(t)}) \\
&= H(M_{D_1}^{(t)}, \ldots, M_{D_{\omega+1}}^{(t)} \mid C_{D_1}^{(t)}, \ldots, C_{D_{\omega+1}}^{(t)}, DK_i, ETI^{(1)}, \ldots, ETI^{(t-1)}) \\
&= \sum_{k=1}^{\omega+1} H(M_{D_k}^{(t)} \mid C_{D_1}^{(t)}, \ldots, C_{D_{\omega+1}}^{(t)}, DK_i, ETI^{(1)}, \ldots, ETI^{(t-1)}, M_{D_1}^{(t)}, \ldots, M_{D_{k-1}}^{(t)}) \\
&= \sum_{k=1}^{\omega+1} H(M_{D_k}^{(t)}) \qquad (4.22) \\
&= (\omega+1)H(M),
\end{aligned}$$

where Eq. (4.22) is shown by following. Let $\mathcal{W}_k := \{l_1, l_2, \ldots, l_{k-1}, l_{k+1}, \ldots, l_{\omega+1}\}$ for each k with $1 \leq k \leq \omega + 1$. Then, we have

$$\begin{aligned}
&H(M_{D_k}^{(t)} \mid C_{D_k}^{(t)}, EK_{\mathcal{W}_k}, DK_i, ETI^{(1)}, \ldots, ETI^{(t-1)}) \\
=&H(M_{D_k}^{(t)} \mid C_{D_k}^{(t)}, EK_{\mathcal{W}_k}, DK_i, ETI^{(1)}, \ldots, ETI^{(t-1)}, \\
&\qquad\qquad M_{D_1}^{(t)}, \ldots, M_{D_{k-1}}^{(t)}, M_{D_{k+1}}^{(t)}, \ldots, M_{D_{\omega+1}}^{(t)}) \\
\leq&H(M_{D_k}^{(t)} \mid C_{D_1}^{(t)}, C_{D_2}^{(t)}, \ldots, C_{D_{\omega+1}}^{(t)}, DK_i, ETI^{(1)}, \ldots, ETI^{(t-1)}, M_{D_1}^{(t)}, \ldots, M_{D_{k-1}}^{(t)}) \\
\leq&H(M_{D_k}^{(t)}).
\end{aligned}$$

And, we have $H(M_{D_k}^{(t)} \mid C_{D_k}^{(t)}, EK_{\mathcal{W}_k}, DK_i, ETI^{(1)}, \ldots, ETI^{(t-1)}) = H(M_{D_k}^{(t)})$ from the condition (3) in Definition 4.5. Hence, $H(M_{D_k}^{(t)} \mid C_{D_1}^{(t)}, C_{D_2}^{(t)}, \ldots, C_{D_{\omega+1}}^{(t)}, DK_i, ETI^{(1)}, \ldots, ETI^{(t-1)}, M_{D_1}^{(t)}, \ldots, M_{D_{k-1}}^{(t)}) = H(M_{D_k}^{(t)})$. □

Lemma 4.13. $H(EMK) \geq \tau(\omega + 1)H(M)$.

Proof. The proof can be shown by the same way as in the proof of Lemma 4.4. □

Proof of Theorem 4.3: From Lemmas 4.10–4.13, the proof of Theorem 4.3 is completed. □

As we will see in Section 4.3.3, the above lower bounds are tight since an instantiation of our generic construction will meet all the above bounds with equalities. Therefore, we define optimality of constructions of TRE as follows.

Definition 4.6. *A construction of (n, ω, τ)-secure TRE is said to be optimal, if it meets equality in every lower bound of (i)–(iv) in Theorem 4.3.*

4.3.3 Construction of TRE from TR-KA and One-time Pad

We present a generic construction of TRE Π_{TRE}=(EGen, EExt, Enc, Dec) starting from TR-KA Π_{KA}=(Setup, Ext, KeyGen, KeyDer) and the one-time pad. In our construction, Π_{KA} and Π_{TRE} satisfy the following conditions: $\mathcal{EMK} = \mathcal{TMK}$; $\mathcal{ETI} = \mathcal{TI}$; $\mathcal{EK} = \mathcal{TUK}^{(S)}$; and $\mathcal{DK} = \mathcal{TUK}^{(R)}$.

1. $(usk_1, \ldots, usk_n, emk^*) \leftarrow$ EGen: For a security parameter 1^κ, EGen outputs matching secret keys $usk_i = (ek_i, dk_i)$ and emk^* for U_i $(1 \leq i \leq n)$ and TS, respectively, as follows. EGen calls Setup with input 1^κ. Suppose $((tuk_1^{(S)}, tuk_1^{(R)}), (tuk_2^{(S)}, tuk_2^{(R)}), \ldots, (tuk_n^{(S)}, tuk_n^{(R)}), tmk^*) \leftarrow$ Setup(1^κ). Then, EGen outputs secret keys $ek_i := tuk_i^{(S)}$, $dk_i := tuk_i^{(R)}$, and $emk^* := tmk^*$ for U_i $(1 \leq i \leq n)$ and TS, respectively.

2. $emk^{(t)} \leftarrow \mathsf{EExt}(emk^*, t)$: For a master key $emk^* = tmk^*$ and time t, EExt calls Ext, and let $tmk^{(t)} \leftarrow \mathsf{Ext}(tmk^*, t)$. Then, EExt outputs a time-signal at the time t, $emk^{(t)} := tmk^{(t)}$.

3. $c^{(t)}_{i_1,i_2} \leftarrow \mathsf{Enc}(m, ek_{i_1}, t, U_{i_2})$: For a plaintext m, an encryption key $ek_{i_1} = tuk^{(S)}_{i_1}$, the specified time t and an identity U_{i_2}, Enc calls KeyGen, and suppose $tck^{(t)}_{i_1,i_2} \leftarrow \mathsf{KeyGen}(tuk^{(S)}_{i_1}, t, U_{i_2})$. Then, Enc outputs a ciphertext $c^{(t)}_{i_1,i_2} := m \oplus tck^{(t)}_{i_1,i_2}$.

4. $m \leftarrow \mathsf{Dec}(dk_{i_2}, emk^{(t)}, U_{i_1})$: For a ciphertext $c^{(t)}_{i_1,i_2}$, a decryption key $dk_{i_2} = tuk^{(R)}_{i_2}$, a time-signal $emk^{(t)} = tmk^{(t)}$ at the specified time t and an identity U_{i_1}, Dec calls KeyDer, and suppose $tck^{(t)}_{i_1,i_2} \leftarrow \mathsf{KeyDer}(tuk^{(R)}_{i_2}, tmk^{(t)}, U_{i_1})$. Then, Dec outputs a plaintext $m := c^{(t)}_{i_1,i_2} \oplus tck^{(t)}_{i_1,i_2}$.

The security of the above construction is shown as follows.

Theorem 4.4. *Given (n, ω, τ)-secure TR-KA Π_{KA} in which common keys are uniformly distributed over $\mathcal{TCK}$ (i.e., $H(TCK^{(t)}_{i,j}) = \log_2 |\mathcal{TCK}|$ for any i, j, and t), then the TRE Π_{TRE} formed by the above construction based on Π_{KA} is (n, ω, τ)-secure.*

Proof. Let Z be a random variable such that: (1) $Z = EMK$; or (2) $Z = (EK_{\mathcal{W}}, DK_{\mathcal{W}}, ETI^{(1)}, \ldots, ETI^{(\tau)})$ with $U_{i_1}, U_{i_2} \notin \mathcal{W}$; or (3) $Z = (EK_{\mathcal{W}}, DK_{\mathcal{W}}, ETI^{(1)}, \ldots, ETI^{(t-1)}, ETI^{(t+1)}, \ldots, ETI^{(\tau)})$ with $U_{i_1} \notin \mathcal{W}$ and $U_{i_2} \in \mathcal{W}$.

Then, for any random variable Z of (1)–(3) mentioned above, we have

$$H(M \mid C, Z) = H(M), \tag{4.23}$$

where (4.23) follows from Definition 4.2 and perfect secrecy of one-time pad $c = m \oplus tck^{(t)}_{i_1,i_2}$ (i.e., each pair of M, TCK, Z is independent). Therefore, the above construction satisfies the conditions (1)–(3). □

Remark 4.1. *For (n, ω, τ)-secure TR-KA Π_{KA} in which common keys are uniformly distributed over $\mathcal{TCK}$, we can prove Theorem 4.1 by using Theorem 4.3 (i.e., the lower bounds in TRE) and the above generic construction in which uniformly distributed plaintexts are taken.*

Remark 4.2. *In the above generic construction, we suppose P_M to be uniform (i.e., uniformly distributed plaintexts) and apply the direct (and optimal) construction of TR-KA in Section 4.2.3. Then, the resulting direct construction of TRE meets equality in every bound of (i)–(iv) in Theorem 4.3. Therefore, the resulting direct construction is optimal and the lower bounds in Theorem 4.3 are tight.*

4.4 Timed-Release Authentication Code

In this section, we show a model and a security definition of timed-release authentication codes (TRA-codes for short). We also derive tight lower bounds on size of secrets for entities and size of time-signals required for TRA-codes. In addition, we present two kinds of constructions of TRA-codes, generic and direct ones. Our generic construction is simple, while our direct construction is optimal.

4.4.1 Model and Security Definition

We newly propose a model and a security definition of TRA-codes, based on that of timed-release signatures with computational security and that of A-codes.

Formally, we give a definition of TRA-codes in the TI model as in the case of TR-KA.

Definition 4.7 (TRA-code). *A TRA-code Π_{TRA} involves $n+2$ entities, TA, $U_1, U_2, \ldots, U_n$ and TS, and consists of a four-tuple of algorithms (TAGen, AExt, TAuth, TVer) with six spaces, $\mathcal{M}_{\mathrm{A}}$, $\mathcal{A}$, $\mathcal{E}$, $\mathcal{AMK}$, $\mathcal{T}$ and $\mathcal{ATI}$, where all of the above algorithms except TAGen are deterministic and all of the above spaces are finite. In addition, Π_{TRA} is executed with four phases as follows.*

- ***Notation:***
  - *Entities: TA, U_i $(1 \leq i \leq n)$, TS, and $\mathcal{U}$ are the same as those in Definition 4.1.*
  - *Spaces: $\mathcal{T}$ is the same as that in Definition 4.1. $\mathcal{A}$ is a set of possible authenticators (or tags), $\mathcal{M}_{\mathrm{A}}$ is a set of possible messages, $\mathcal{AMK}$ is a set of possible master keys. $\mathcal{ATI}^{(t)}$ is a set of possible time-signals at time $t \in \mathcal{T}$. Let $\mathcal{ATI} := \bigcup_{t=1}^{\tau} \mathcal{ATI}^{(t)}$. Also, $\mathcal{E}_i^{(S)}$ is a set of possible U_i's authentication keys, $\mathcal{E}_i^{(R)}$ is a set of possible U_i's verification keys, and $\mathcal{E}_i := \mathcal{E}_i^{(S)} \times \mathcal{E}_i^{(R)}$ is a set of possible secret keys for U_i. Let $\mathcal{E}^{(S)} := \bigcup_{i=1}^{n} \mathcal{E}_i^{(S)}$, $\mathcal{E}^{(R)} := \bigcup_{i=1}^{n} \mathcal{E}_i^{(R)}$, and $\mathcal{E} := \bigcup_{i=1}^{n} \mathcal{E}_i$.*
  - *Algorithms: TAGen is a key generation algorithm which on input a security parameter 1^{κ}, outputs each user's secret key and a time-server's master key, AExt: $\mathcal{AMK} \times \mathcal{T} \rightarrow \mathcal{ATI}$ is a time-signal generation algorithm for TS, TAuth: $\mathcal{M}_{\mathrm{A}} \times \mathcal{E}^{(S)} \times \mathcal{T} \times \mathcal{U} \rightarrow \mathcal{A}$ is an authentication algorithm, and TVer: $\mathcal{M}_{\mathrm{A}} \times \mathcal{A} \times \mathcal{T} \times \mathcal{E}^{(R)} \times \mathcal{ATI} \times \mathcal{U} \rightarrow \{true, false\}$ is a verification algorithm.*

1. ***Key Generation and Distribution.*** *In the initial phase, TA generates the following keys by using TAGen: a master key $amk^* \in \mathcal{AMK}$ for*

TS; a secret key $e_i = (e_i^{(S)}, e_i^{(R)}) \in \mathcal{E}_i$ for U_i $(i = 1, 2, \ldots, n)$. These keys are distributed to corresponding entities via secure channels. After distributing these keys, TA deletes them from his memory. And, TS and U_i keep their keys secret, respectively.

2. ***Time-signal Generation.*** *For broadcasting a time-signal at each time, TS generates a time-signal $amk^{(t)} \leftarrow \mathsf{AExt}(amk^*, t) \in \mathcal{ATI}^{(t)}$ by using a master key $amk^* \in \mathcal{AMK}$ and time $t \in \mathcal{T}$. Then, TS broadcasts it to all users via a broadcast channel.*

3. ***Authentication.*** *U_{i_1} specifies time t when U_{i_2} can verify validity of a message m, and then U_{i_1} computes an authenticator, $\alpha_{i_1,i_2}^{(t)} \leftarrow \mathsf{TAuth}(m, e_{i_1}^{(S)}, t, U_{i_2}) \in \mathcal{A}$, by the message $m \in \mathcal{M}_{\mathrm{A}}$, an authentication key $e_{i_1}^{(S)}$, the specified time t and the identity U_{i_2}. And, U_{i_1} sends $(m, \alpha_{i_1,i_2}^{(t)}, t)$ to U_{i_2} via an insecure channel.*

4. ***Verification.*** *Suppose that U_{i_2} has received $(m, \alpha_{i_1,i_2}^{(t)}, t)$ from U_{i_1}. After receiving a time-signal $amk^{(t)}$ at the specified time t, U_{i_2} checks the validity of $\alpha_{i_1,i_2}^{(t)}$ by a verification key $e_{i_2}^{(R)}$, a time-signal $amk^{(t)}$ and the identity U_{i_1}: If $\mathsf{TVer}(m, \alpha_{i_1,i_2}^{(t)}, t, e_{i_2}^{(R)}, amk^{(t)}, U_{i_1}) \rightarrow true$, then U_{i_2} accepts $(m, \alpha_{i_1,i_2}^{(t)}, t)$ as valid, and rejects it otherwise.*

In the model of TRA-codes, we require the following equation holds: for all possible $\kappa \in \mathbb{N}$, $(e_1, \ldots, e_n, amk^*) \leftarrow \mathsf{TAGen}(1^\kappa)$, $t \in \mathcal{T}$, and $i_1, i_2 \in [n]$, we have

$$\mathsf{TVer}(m, \mathsf{TAuth}(m, e_{i_1}^{(S)}, t, U_{i_2}), t, e_{i_2}^{(R)}, \mathsf{AExt}(amk^*, t), U_{i_1}) \rightarrow true.$$

The above requirement means correctness of TRA-codes.

Next, we provide a security notion and its formalization for TRA-codes based on the idea of timed-release security and the traditional authentication code with information-theoretic security. The choice of possible colluders $\mathcal{W} \in \mathcal{PS}(\mathcal{U}, \omega)$ is the same as that in TR-KA. For a set of colluders $\mathcal{W} := \{U_{l_1}, U_{l_2}, \ldots, U_{l_j}\} \in \mathcal{PS}(\mathcal{U}, \omega)$, $\mathcal{E}_{\mathcal{W}}^{(S)} := \mathcal{E}_{l_1}^{(S)} \times \mathcal{E}_{l_2}^{(S)} \times \cdots \times \mathcal{E}_{l_j}^{(S)}$ is a set of $\mathcal{W}$'s authentication keys, and $\mathcal{E}_{\mathcal{W}}^{(R)} := \mathcal{E}_{l_1}^{(R)} \times \mathcal{E}_{l_2}^{(R)} \times \cdots \times \mathcal{E}_{l_j}^{(R)}$ is a set of $\mathcal{W}$'s verification keys. In TRA-codes, we consider *impersonation attacks* and *substitution attacks* as follows. (a) *Impersonation attacks:* an adversary (or a dishonest entity) tries to generate a fraudulent authenticated message at time t, $(m, \alpha_{i_1,i_2}^{(t)}, t)$, that has not been legally generated by a sender U_{i_1} but will be accepted by a receiver U_{i_2}. (b) *Substitution attacks:* an adversary (or a dishonest entity) tries to generate a fraudulent authenticated message at time t_2, $(m', \alpha_{i_1,i_2}^{(t_2)}, t_2)$, that has not been legally generated by a sender U_{i_1} but will be accepted by a receiver U_{i_2}, after observing a valid authenticated message

at time t_1, $(m, \alpha_{i_1,i_2}^{(t_1)}, t_1)$ with $(m, \alpha_{i_1,i_2}^{(t_1)}, t_1) \neq (m', \alpha_{i_1,i_2}^{(t_2)}, t_2)$. Similarly as in Definition 4.2 we consider the following three types of security notions for TRA-codes: (1) The probability that a dishonest time-server succeeds in each of the *impersonation attack* and *substitution attack* is small; (2) The probability that any colluding group $\mathcal{W}$ not including a legitimate receiver succeeds in each of the *impersonation attack* and *substitution attack* is small, even if $\mathcal{W}$ obtains time-signals at all the time; (3) Any colluding group $\mathcal{W}$ including a legitimate (but dishonest) receiver cannot check the validity of a target authenticated message without a time-signal at the specified time, even if $\mathcal{W}$ obtains time-signals at all the time except the specified time. To formalize this security notion, we consider it to be a kind of security against impersonation attacks at the future specified time: The probability that any colluding group $\mathcal{W}$ including a receiver succeeds in impersonation attacks at the future specified time is small, even if $\mathcal{W}$ obtains time-signals at all the time except the specified time.

The formalizations of the above three types of security notions for TRA-codes are given as follows.

Definition 4.8. *Let Π_{TRA} be a TRA-code. Π_{TRA} is said to be $(n, \omega, \tau; \epsilon)$-secure, if $\max\{P_{Server}, P_1, P_2\} \leq \epsilon$, where P_{Server}, P_1 and P_2 are defined as follows.*

(1) Success probability of attacks by a dishonest time-server. Let $P_{Server} := \max\{P_{I_S}, P_{S_S}\}$, where P_{I_S} and P_{S_S} are given as follows.

1-1) Success probability of impersonation attacks. For any $U_{i_1}, U_{i_2} \in \mathcal{U}$ and any $t \in \mathcal{T}$, we define $P_{I_S}(U_{i_1}, U_{i_2}, t)$ by

$$P_{I_S}(U_{i_1}, U_{i_2}, t) := \max_{(m, \alpha_{i_1,i_2}^{(t)}, t)} \max_{amk^*} \max_{amk^{(t)}} \Pr[\mathsf{TVer}(m, \alpha_{i_1,i_2}^{(t)}, t, e_{i_2}^{(R)}, amk^{(t)}, U_{i_1}) = true \mid amk^*].$$

The probability P_{I_S} is defined as $P_{I_S} := \max_{U_{i_1}, U_{i_2}, t} P_{I_S}(U_{i_1}, U_{i_2}, t)$.

1-2) Success probability of substitution attacks. For any $U_{i_1}, U_{i_2} \in \mathcal{U}$ and any $t_1, t_2 \in \mathcal{T}$, we define $P_{S_S}(U_{i_1}, U_{i_2}, t_1, t_2)$ by

$$P_{S_S}(U_{i_1}, U_{i_2}, t_1, t_2) := \max_{(m', \alpha_{i_1,i_2}^{(t_2)}, t_2)} \max_{(m, \alpha_{i_1,i_2}^{(t_1)}, t_1) \neq (m', \alpha_{i_1,i_2}^{(t_2)}, t_2)} \max_{amk^*} \max_{amk^{(t_2)}} \Pr[\mathsf{TVer}(m', \alpha_{i_1,i_2}^{(t_2)}, t_2, e_{i_2}^{(R)}, amk^{(t_2)}, U_{i_1}) = true \mid (m, \alpha_{i_1,i_2}^{(t_1)}, t_1), amk^*].$$

The probability P_{S_S} is defined as $P_{S_S} := \max_{U_{i_1}, U_{i_2}, t_1, t_2} P_{S_S}(U_{i_1}, U_{i_2}, t_1, t_2)$.

(2) Success probability of attacks by colluders not including a legitimate receiver. Let $P_1 := \max\{P_{I_1}, P_{S_1}\}$, where P_{I_1} and P_{S_1} are given as follows.

2-1) Success probability of impersonation attacks. For any set of colluders $\mathcal{W} \in \mathcal{PS}(\mathcal{U}, \omega)$*, any* $U_{i_1}, U_{i_2} \in \mathcal{U}$ *such that* $U_{i_1}, U_{i_2} \notin \mathcal{W}$ *and for any* $t \in \mathcal{T}$*, we define* $P_{I_1}(U_{i_1}, U_{i_2}, \mathcal{W}, t)$ *by*

$$P_{I_1}(U_{i_1}, U_{i_2}, \mathcal{W}, t) := \max_{(m, \alpha^{(t)}_{i_1,i_2}, t)} \max_{e^{(S)}_{\mathcal{W}}} \max_{e^{(R)}_{\mathcal{W}}} \max_{amk^{(1)}, \ldots, amk^{(\tau)}} \Pr[\mathit{TVer}(m, \alpha^{(t)}_{i_1,i_2}, t, e^{(R)}_{i_2}, amk^{(t)}, U_{i_1}) = true \mid e^{(S)}_{\mathcal{W}}, e^{(R)}_{\mathcal{W}}, amk^{(1)}, \ldots, amk^{(\tau)}].$$

The probability P_{I_1} *is defined as* $P_{I_1} := \max_{U_{i_1}, U_{i_2}, \mathcal{W}, t} P_{I_1}(U_{i_1}, U_{i_2}, \mathcal{W}, t)$.

2-2) Success probability of substitution attacks. For any set of colluders $\mathcal{W} \in \mathcal{PS}(\mathcal{U}, \omega)$*, any* $U_{i_1}, U_{i_2} \in \mathcal{U}$ *such that* $U_{i_1}, U_{i_2} \notin \mathcal{W}$ *and for any* $t_1, t_2 \in \mathcal{T}$*, we define* $P_{S_1}(U_{i_1}, U_{i_2}, \mathcal{W}, t_1, t_2)$ *by*

$$P_{S_1}(U_{i_1}, U_{i_2}, \mathcal{W}, t_1, t_2) := \max_{(m', \alpha^{(t_2)}_{i_1,i_2}, t_2)} \max_{(m, \alpha^{(t_1)}_{i_1,i_2}, t_1) \neq (m', \alpha^{(t_2)}_{i_1,i_2}, t_2)} \max_{e^{(S)}_{\mathcal{W}}} \max_{e^{(R)}_{\mathcal{W}}} \max_{amk^{(1)}, \ldots, amk^{(\tau)}} \Pr[\mathit{TVer}(m', \alpha^{(t_2)}_{i_1,i_2}, t_2, e^{(R)}_{i_2}, amk^{(t_2)}, U_{i_1}) = true \mid (m, \alpha^{(t_1)}_{i_1,i_2}, t_1), e^{(S)}_{\mathcal{W}}, e^{(R)}_{\mathcal{W}}, amk^{(1)}, \ldots, amk^{(\tau)}].$$

And, P_{S_1} *is defined as* $P_{S_1} := \max_{U_{i_1}, U_{i_2}, \mathcal{W}, t_1, t_2} P_{S_1}(U_{i_1}, U_{i_2}, \mathcal{W}, t_1, t_2)$.

(3) Success probability of attacks by colluders including a legitimate (but dishonest) receiver. For any set of colluders $\mathcal{W} \in \mathcal{PS}(\mathcal{U}, \omega)$*, any* $U_{i_1}, U_{i_2} \in \mathcal{U}$ *such that* $U_{i_1} \notin \mathcal{W}$ *and* $U_{i_2} \in \mathcal{W}$*, and for any* $t \in \mathcal{T}$*, we define* $P_2(U_{i_1}, U_{i_2}, \mathcal{W}, t)$ *by*

$$P_2(U_{i_1}, U_{i_2}, \mathcal{W}, t) := \max_{(m, \alpha^{(t)}_{i_1,i_2}, t)} \max_{e^{(S)}_{\mathcal{W}}} \max_{e^{(R)}_{\mathcal{W}}} \max_{amk^{(1)}, \ldots, amk^{(t-1)}, amk^{(t+1)}, \ldots, amk^{(\tau)}} \Pr[\mathit{TVer}(m, \alpha^{(t)}_{i_1,i_2}, t, e^{(R)}_{i_2}, amk^{(t)}, U_{i_1}) = true \mid e^{(S)}_{\mathcal{W}}, e^{(R)}_{\mathcal{W}} amk^{(1)}, \ldots, amk^{(t-1)}, amk^{(t+1)}, \ldots, amk^{(\tau)}].$$

The probability P_2 *is defined as* $P_2 := \max_{U_{i_1}, U_{i_2}, \mathcal{W}, t} P_2(U_{i_1}, U_{i_2}, \mathcal{W}, t)$.

4.4.2 Lower Bounds

We derive lower bounds on success probabilities of attacks and size of secrets required for $(n, \omega, \tau; \epsilon)$-secure TRA-codes. Let $\mathcal{MA}^{(t)}_{i_1,i_2} := \{(m, \alpha^{(t)}_{i_1,i_2}) \in \mathcal{M}_{\mathrm{A}} \times \mathcal{A} \mid \mathit{TAuth}(m, e^{(S)}_{i_1}, t, U_{i_2}) = \alpha^{(t)}_{i_1,i_2}$ for some $e^{(S)}_{i_1} \in \mathcal{E}^{(S)}_{i_1}\}$ be a set of

possible pairs of messages and authenticators such that each element of the set can be generated by the sender U_{i_1} to send it to U_{i_2} at specified future time t. Furthermore, let $MA_{i_1,i_2}^{(t)}$, AMK, $E_{\mathcal{W}}^{(S)}$, $E_{\mathcal{W}}^{(R)}$, $ATI^{(1)}$, $\ldots$, $ATI^{(\tau)}$ be random variables which take values in $\mathcal{MA}_{i_1,i_2}^{(t)}$, $\mathcal{AMK}$, $\mathcal{E}_{\mathcal{W}}^{(S)}$, $\mathcal{E}_{\mathcal{W}}^{(R)}$, $\mathcal{ATI}^{(1)}$, $\ldots$, $\mathcal{ATI}^{(\tau)}$, respectively.

We assume that there exist the following mappings in the model of TRA-codes: for every $i, j \in [n]$ and every $t \in [\tau]$,

$$\begin{aligned}
\lambda_i &: \mathcal{E}_i^{(S)} \to \mathcal{E}_{i,1}^{(S)} \times \cdots \times \mathcal{E}_{i,n}^{(S)}, \\
\pi_j &: \mathcal{E}_j^{(R)} \to \mathcal{E}_{1,j}^{(R)} \times \cdots \times \mathcal{E}_{n,j}^{(R)}, \\
f^{(t)} &: \mathcal{ATI}^{(t)} \to \mathcal{ATI}_1^{(t)} \times \cdots \times \mathcal{ATI}_n^{(t)}, \\
g &: \mathcal{AMK} \to \mathcal{AMK}_1 \times \cdots \times \mathcal{AMK}_n, \\
g_i &: \mathcal{AMK}_i \to \mathcal{ATI}_i^{(1)} \times \cdots \times \mathcal{ATI}_i^{(\tau)}, \\
\rho_{i,j} &: \mathcal{E}_{i,j}^{(S)} \to \mathcal{E}_{i,j}^{(R)} \times \mathcal{AMK}_i,
\end{aligned}$$

where $\mathcal{E}_{i,j}^{(S)}$ is a set of possible U_i's authentication keys which are actually used to communicate with a receiver U_j; $\mathcal{E}_{i,j}^{(R)}$ is a set of possible U_j's verification keys which are actually used to communicate with a sender U_i; $\mathcal{ATI}_i^{(t)}$ is a set of possible information on time-signals at time t when U_i becomes a sender; $\mathcal{AMK}_i$ is a set of possible partial information about master keys when U_i becomes a sender.[3] Note that each user has the potential to become an adversary, but each user is honest when he is a sender. Hence, if a sender U_i is fixed and $amk_i^{(t)} \in \mathcal{ATI}_i^{(t)}$ is given, TRA-codes look like MRA-codes [124]. From this, it would be natural to assume a mapping $\mathcal{E}_{i,j}^{(S)} \to \mathcal{E}_{i,j}^{(R)}$, if $amk_i^{(t)} \in \mathcal{ATI}_i^{(t)}$ is given, in TRA-codes as in the model of MRA-codes (see Definition 3.1 in [124]). In addition, from the footnote of this page, we have assumed the above mapping $\rho_{i,j} : \mathcal{E}_{i,j}^{(S)} \to \mathcal{E}_{i,j}^{(R)} \times \mathcal{AMK}_i$. From the explanation, we consider that the assumption of existence of the above mappings is not so strange, rather natural, and we will show that these mappings actually exist in our simple direct construction in Section 4.4.4.

Then, we can derive lower bounds on success probabilities of attacks as follows.

Theorem 4.5. *For any $i_1, i_2 \in [n]$, any time $t \in \mathcal{T}$, any colluding group $\mathcal{W}$ with $U_{i_1}, U_{i_2} \notin \mathcal{W}$, and $\tilde{\mathcal{W}}$ with $U_{i_1} \notin \tilde{\mathcal{W}}$ and $U_{i_2} \in \tilde{\mathcal{W}}$, it holds that*

1. $\log P_{I_S}(U_{i_1}, U_{i_2}, t) \geq -I(MA_{i_1,i_2}^{(t)}; E_{i_1,i_2}^{(R)} \mid AMK)$.

[3] We assume that each user U_i potentially has partial information on a master key, since a sender U_i can specify any time t (i.e., the sender U_i can generate $amk_i^{(t)} \in \mathcal{ATI}_i^{(t)}$ for $1 \leq \forall t \leq \tau$) but he cannot generate a time-signal $amk^{(t)} \in \mathcal{ATI}^{(t)}$.

2. $\log P_{S_S}(U_{i_1}, U_{i_2}, t_1, t_2) \geq -I(\tilde{MA}^{(t_2)}_{i_1,i_2}; E^{(R)}_{i_1,i_2} \mid AMK, MA^{(t_1)}_{i_1,i_2})$.
3. $\log P_{I_1}(U_{i_1}, U_{i_2}, \mathcal{W}, t) \geq -I(MA^{(t)}_{i_1,i_2}; E^{(R)}_{i_1,i_2} \mid E^{(S)}_{\mathcal{W}}, E^{(R)}_{\mathcal{W}}, ATI^{(1)}, \ldots, ATI^{(\tau)})$.
4. $\log P_{S_1}(U_{i_1}, U_{i_2}, \mathcal{W}, t_1, t_2)$
$\geq -I(\tilde{MA}^{(t_2)}_{i_1,i_2}; E^{(R)}_{i_1,i_2} \mid E^{(S)}_{\mathcal{W}}, E^{(R)}_{\mathcal{W}}, ATI^{(1)}, \ldots, ATI^{(\tau)}, MA^{(t_1)}_{i_1,i_2})$.
5. $\log P_2(U_{i_1}, U_{i_2}, \tilde{\mathcal{W}}, t)$
$\geq -I(MA^{(t)}_{i_1,i_2}; ATI^{(t)}_{i_1}$
$\mid E^{(S)}_{\tilde{\mathcal{W}}}, E^{(R)}_{\tilde{\mathcal{W}}}, ATI^{(1)}, \ldots, ATI^{(t-1)}, ATI^{(t+1)}, \ldots, ATI^{(\tau)})$.

The proof can be shown in a way similar to [124, Theorem 3.2], and we omit it.

We next show lower bounds on sizes of secret keys and time-signals required for secure TRA-codes.

Theorem 4.6. *Let Π_{TRA} be an $(n, \omega, \tau; \epsilon)$-secure TRA-code. Let $q := \epsilon^{-1}$. Then, for any $i_1, i_2 \in [n]$ and $t \in [\tau]$, we have*

$$(i)\ \ |\mathcal{E}^{(R)}_{i_2}| \geq q^{2(\omega+1)},\quad (ii)\ \ |\mathcal{E}^{(S)}_{i_1}| \geq q^{2\omega+\tau+1},$$
$$(iii)\ \ |\mathcal{ATI}^{(t)}| \geq q^{\omega+1},\quad (iv)\ \ |\mathcal{AMK}| \geq q^{\tau(\omega+1)},\quad (v)\ \ |\mathcal{A}^{(t)}_{i_1,i_2}| \geq q.$$

Proof. In order to complete the proof, we show the following lemmas.

Lemma 4.14. *$|\mathcal{E}^{(R)}_{i_2}| \geq q^{2(\omega+1)}$ for any $i_2 \in [n]$.*

Proof. For arbitrary $i_1, i_2 \in [n]$, let $\mathcal{W}_{i_1} := \{U_1, \ldots, U_{i_1-1}, U_{i_1+1}, \ldots, U_{\omega+1}\}$ such that $U_{i_2} \notin \mathcal{W}_{i_1}$. Then, for any $t_1, t_2 \in \mathcal{T}$, we have

$$\begin{aligned}
\left(\frac{1}{q}\right)^{2(\omega+1)} &\geq \prod_{i_1=1}^{\omega+1} P_{I_1}(U_{i_1}, U_{i_2}, \mathcal{W}_{i_1}, t_1) P_{S_1}(U_{i_1}, U_{i_2}, \mathcal{W}_{i_1}, t_1, t_2) \\
&\geq 2^{-\sum_{i_1=1}^{\omega+1} H(E^{(R)}_{i_1,i_2} | E^{(S)}_{\mathcal{W}_{i_1}})} && (4.24)\\
&\geq 2^{-\sum_{i_1=1}^{\omega+1} H(E^{(R)}_{i_1,i_2} | E^{(S)}_{1}, \ldots, E^{(S)}_{i_1-1})} \\
&\geq 2^{-\sum_{i_1=1}^{\omega+1} H(E^{(R)}_{i_1,i_2} | E^{(S)}_{1,i_2}, \ldots, E^{(S)}_{i_1-1,i_2})} && (4.25)\\
&\geq 2^{-\sum_{i_1=1}^{\omega+1} H(E^{(R)}_{i_1,i_2} | E^{(R)}_{1,i_2}, \ldots, E^{(R)}_{i_1-1,i_2})} && (4.26)\\
&= 2^{-H(E^{(R)}_{1,i_2}, \ldots, E^{(R)}_{\omega+1,i_2})} \\
&\geq 2^{-H(E^{(R)}_{i_2})} && (4.27)\\
&\geq 2^{-\log |\mathcal{E}^{(R)}_{i_2}|} = \frac{1}{|\mathcal{E}^{(R)}_{i_2}|},
\end{aligned}$$

where Eq. (4.24) follows from Theorem 4.5, and Eqs. (4.25), (4.26), and (4.27) follow from the mappings, λ_{i_1} for $1 \leq i_1 \leq \omega$, ρ_{i_1,i_2} for $1 \leq i_1 \leq \omega$, and π_{i_2}, respectively. Therefore, we have $|\mathcal{E}_{i_2}^{(R)}| \geq q^{2(\omega+1)}$. □

Lemma 4.15. *$|\mathcal{ATI}^{(t)}| \geq q^{\omega+1}$ for any $t \in \mathcal{T}$.*

Proof. For arbitrary $i_1, i_2 \in [n]$, let $\tilde{\mathcal{W}}_{i_1} := \{U_1, \ldots, U_{i_1-1}, U_{i_1+1}, \ldots, U_{\omega+1}\}$ such that $U_{i_2} \in \tilde{\mathcal{W}}_{i_1}$. Then, for any $t \in \mathcal{T}$, we have

$$\begin{aligned}
\left(\frac{1}{q}\right)^{\omega+1} &\geq \prod_{i_1=1}^{\omega+1} P_2(U_{i_1}, U_{i_2}, \tilde{\mathcal{W}}_{i_1}, t) \\
&\geq 2^{-\sum_{i_1=1}^{\omega+1} H(ATI_{i_1}^{(t)} | E_{\tilde{\mathcal{W}}_{i_1}}^{(S)})} && (4.28) \\
&\geq 2^{-\sum_{i_1=1}^{\omega+1} H(ATI_{i_1}^{(t)} | E_1^{(S)}, \ldots, E_{i_1-1}^{(S)})} \\
&\geq 2^{-\sum_{i_1=1}^{\omega+1} H(ATI_{i_1}^{(t)} | AMK_1, \ldots, AMK_{i_1-1})} && (4.29) \\
&\geq 2^{-\sum_{i_1=1}^{\omega+1} H(ATI_{i_1}^{(t)} | ATI_1^{(t)}, \ldots, ATI_{i_1-1}^{(t)})} && (4.30) \\
&= 2^{-H(ATI_1^{(t)}, \ldots, ATI_{\omega+1}^{(t)})} \\
&\geq 2^{-H(ATI^{(t)})} && (4.31) \\
&\geq 2^{-\log |\mathcal{ATI}^{(t)}|} = \frac{1}{|\mathcal{ATI}^{(t)}|},
\end{aligned}$$

where Eq. (4.28) follows from Theorem 4.5; Eq. (4.29) follow from the mappings λ_{i_1} and ρ_{i_1,i_2} for $1 \leq i_1 \leq \omega$; Eqs. (4.30) and (4.31) follow from the mappings g_{i_1} for $1 \leq i_1 \leq \omega$, and $f^{(t)}$, respectively. Therefore, we have $|\mathcal{ATI}^{(t)}| \geq q^{\omega+1}$. □

Lemma 4.16. *$|\mathcal{AMK}| \geq q^{\tau(\omega+1)}$.*

Proof. For arbitrary $i_1, i_2 \in [n]$, let $\tilde{\mathcal{W}}_{i_1} := \{U_1, \ldots, U_{i_1-1}, U_{i_1+1}, \ldots, U_{\omega+1}\}$ such that $U_{i_2} \in \tilde{\mathcal{W}}_{i_1}$. Then, for any $t \in \mathcal{T}$, we have

$$\begin{aligned}
\left(\frac{1}{q}\right)^{\tau(\omega+1)} &\geq \prod_{t=1}^{\tau} \prod_{i_1=1}^{\omega+1} P_2(U_{i_1}, U_{i_2}, \tilde{\mathcal{W}}_{i_1}, t) \\
&\geq 2^{-\sum_{t=1}^{\tau} \sum_{i_1=1}^{\omega+1} H(ATI_{i_1}^{(t)} | E_{\tilde{\mathcal{W}}_{i_1}}^{(S)}, ATI^{(1)}, \ldots, ATI^{(t-1)})} && (4.32) \\
&\geq 2^{-\sum_{t=1}^{\tau} \sum_{i_1=1}^{\omega+1} H(ATI_{i_1}^{(t)} | E_1^{(S)}, \ldots, E_{i_1-1}^{(S)}, ATI^{(1)}, \ldots, ATI^{(t-1)})} \\
&\geq 2^{-\sum_{t=1}^{\tau} \sum_{i_1=1}^{\omega+1} H(ATI_{i_1}^{(t)} | AMK_1, \ldots, AMK_{i_1-1}, ATI^{(1)}, \ldots, ATI^{(t-1)})} && (4.33) \\
&\geq 2^{-\sum_{t=1}^{\tau} \sum_{i_1=1}^{\omega+1} H(ATI_{i_1}^{(t)} | ATI_1^{(t)}, \ldots, ATI_{i_1-1}^{(t)}, ATI^{(1)}, \ldots, ATI^{(t-1)})} && (4.34)
\end{aligned}$$

$$
\begin{aligned}
&= 2^{-\sum_{t=1}^{\tau} H(ATI_1^{(t)},\ldots,ATI_{\omega+1}^{(t)} | ATI^{(1)},\ldots,ATI^{(t-1)})} \\
&\geq 2^{-\sum_{t=1}^{\tau} H(ATI^{(t)} | ATI^{(1)},\ldots,ATI^{(t-1)})} && (4.35) \\
&= 2^{-H(ATI^{(1)},\ldots,ATI^{(\tau)})} \\
&\geq 2^{-H(AMK)} && (4.36) \\
&\geq 2^{-\log|\mathcal{AMK}|} = \frac{1}{|\mathcal{AMK}|},
\end{aligned}
$$

where Eq. (4.32) follows from Theorem 4.5; Eq. (4.33) follow from the mappings λ_{i_1} and ρ_{i_1,i_2} for $1 \leq i_1 \leq \omega$; Eqs. (4.34) and (4.35) follow from the mappings g_{i_1} for $1 \leq i_1 \leq \omega$ and $f^{(t)}$, respectively; Eq. (4.36) follows from the deterministic algorithm (i.e., mapping) AExt: $\mathcal{AMK} \times \mathcal{T} \to \mathcal{ATI}$. Therefore, we have $|\mathcal{AMK}| \geq q^{\tau(\omega+1)}$. □

Lemma 4.17. $|\mathcal{E}_{i_1}^{(S)}| \geq q^{2\omega+\tau+1}$ *for any* $i_1 \in [n]$.

Proof. For arbitrary $i_1, i_2 \in [n]$, let $\mathcal{W}_{i_2} := \{U_1, \ldots, U_{i_2-1}, U_{i_2+1}, \ldots, U_{\omega+1}\}$ such that $U_{i_1} \notin \mathcal{W}_{i_2}$, and $\tilde{\mathcal{W}} \in \mathcal{PS}(\mathcal{U}, \omega)$ such that $U_{i_1} \notin \tilde{\mathcal{W}}$ and $U_{i_2} \in \tilde{\mathcal{W}}$. Then, for any $t, t_1, t_2 \in \mathcal{T}$, we have

$$
\begin{aligned}
&\log\left(\frac{1}{q}\right)^{2\omega+\tau+1} && (4.37) \\
&\geq \log\left(\prod_{t=2}^{\tau} P_2(U_{i_1}, U_{i_2}, \tilde{\mathcal{W}}, t) \prod_{i_2=1}^{\omega+1} P_{I_1}(U_{i_1}, U_{i_2}, \mathcal{W}_{i_2}, t_1) P_{S_1}(U_{i_1}, U_{i_2}, \mathcal{W}_{i_2}, t_1, t_2)\right) \\
&\geq -\sum_{t=2}^{\tau} H(ATI_{i_1}^{(t)} \mid E_{\tilde{\mathcal{W}}}^{(S)}, E_{\tilde{\mathcal{W}}}^{(R)}, ATI^{(1)}, \ldots, ATI^{(t-1)}, ATI^{(t+1)}, \ldots, ATI^{(\tau)}) \\
&\qquad - \sum_{i_2=1}^{\omega+1} H(E_{i_1,i_2}^{(t)} \mid E_{\mathcal{W}_{i_2}}^{(S)}, E_{\mathcal{W}_{i_2}}^{(R)}, ATI^{(1)}, \ldots, ATI^{(\tau)}) && (4.38) \\
&\geq -\sum_{t=2}^{\tau} H(ATI_{i_1}^{(t)} \mid ATI^{(1)}, ATI^{(2)}, \ldots, ATI^{(t-1)}) \\
&\qquad - \sum_{i_2=1}^{\omega+1} H(E_{i_1,i_2}^{(t)} \mid E_{\mathcal{W}_{i_2}}^{(R)}, ATI^{(1)}, \ldots, ATI^{(\tau)}) \\
&\geq -\sum_{t=2}^{\tau} H(ATI_{i_1}^{(t)} \mid ATI^{(2)}, ATI^{(3)}, \ldots, ATI^{(t-1)}) \\
&\qquad - \sum_{i_2=1}^{\omega+1} H(E_{i_1,i_2}^{(t)} \mid E_1^{(R)}, \ldots, E_{i_2-1}^{(R)} ATI^{(2)}, \ldots, ATI^{(\tau)}) \\
&\geq -\sum_{t=2}^{\tau} H(ATI_{i_1}^{(t)} \mid ATI_{i_1}^{(2)}, ATI_{i_1}^{(3)}, \ldots, ATI_{i_1}^{(t-1)})
\end{aligned}
$$

$$- \sum_{i_2=1}^{\omega+1} H(E_{i_1,i_2}^{(t)} \mid E_{i_1,1}^{(R)}, \ldots, E_{i_1,i_2-1}^{(R)}, ATI_{i_1}^{(2)}, \ldots, ATI_{i_1}^{(\tau)}) \tag{4.39}$$

$$= - H(ATI_{i_1}^{(2)}, ATI_{i_1}^{(3)}, \ldots, ATI_{i_1}^{(\tau)}, E_{i_1,1}^{(R)}, \ldots, E_{i_1,\omega+1}^{(R)})$$

$$\geq - H(AMK_{i_1}, E_{i_1,1}^{(R)}, \ldots, E_{i_1,\omega+1}^{(R)}) \tag{4.40}$$

$$\geq - H(E_{i_1,1}^{(S)}, \ldots, E_{i_1,\omega+1}^{(S)}) \tag{4.41}$$

$$\geq - H(E_{i_1}^{(S)}) \tag{4.42}$$

$$\geq - \log |\mathcal{E}_{i_1}^{(S)}|,$$

where Eq. (4.38) follows from Theorem 4.5; Eq. (4.39) follows from the mappings π_{i_2} for $1 \leq i_2 \leq \omega$ and $f^{(t)}$ for $2 \leq t \leq \tau$; Eqs. (4.40), (4.41), and (4.42) follow from the mappings, g_{i_1}, ρ_{i_1,i_2} for $1 \leq i_2 \leq \omega+1$, and λ_{i_1}, respectively. Therefore, $|\mathcal{E}_{i_1}^{(S)}| \geq q^{2\omega+\tau+1}$. □

Lemma 4.18. *$|\mathcal{A}_{i_1,i_2}^{(t)}| \geq q$ for any $i_1, i_2 \in [n]$ and $t \in \mathcal{T}$.*

Proof. Let $\mathcal{W} = \emptyset$. Then, we have

$$\begin{aligned}
\frac{1}{q} \geq & P_{I_1}(U_{i_1}, U_{i_2}, \mathcal{W}, t) \\
\geq & 2^{-I(MA_{i_1,i_2}^{(t)}; E_{i_1,i_2}^{(R)} | ATI^{(1)}, \ldots, ATI^{(\tau)})} \\
= & 2^{-I(M; E_{i_1,i_2}^{(R)} | ATI^{(1)}, \ldots, ATI^{(\tau)}) - I(A_{i_1,i_2}^{(t)}; E_{i_1,i_2}^{(R)} | ATI^{(1)}, \ldots, ATI^{(\tau)}, M)} \\
= & 2^{-I(A_{i_1,i_2}^{(t)}; E_{i_1,i_2}^{(R)} | ATI^{(1)}, \ldots, ATI^{(\tau)}, M)} \\
\geq & 2^{-H(A_{i_1,i_2}^{(t)})} \geq \frac{1}{|\mathcal{A}_{i_1,i_2}^{(t)}|},
\end{aligned} \tag{4.43}$$

where Eq. (4.43) follows from Theorem 4.5. Therefore, we have $|\mathcal{A}_{i_1,i_2}^{(t)}| \geq q$. □

Proof of Theorem 4.6: From Lemmas 4.14–4.18, the proof of Theorem 4.6 is completed. □

As we will see in Section 4.4.4, the above lower bounds are all tight since our direct construction will meet all the above bounds with equalities. Therefore, we define optimality of constructions of TRA-codes as follows.

Definition 4.9. A construction of $(n, \omega, \tau; \epsilon)$-secure TRA-codes is said to be *optimal* if it meets equality in every lower bound of (i)–(v) in Theorem 4.6.

4.4.3 Generic Construction of TRA-codes from TR-KA and A-codes

We propose a generic construction of $(n, \omega, \tau; \epsilon)$-secure TRA-codes from TR-KA and the traditional A-codes.

The detail of our generic construction of TRA-codes Π_{TRA}=(TAGen, AExt, TAuth, TVer) by using TR-KA Π_{KA}=(Setup, Ext, KeyGen, KeyDer) and A-codes Π_{A}=(KGen, Auth, Ver) is given as follows.

1. $(e_1, \ldots, e_n, amk^*) \leftarrow \mathsf{TAGen}(1^\kappa)$: For a security parameter 1^κ, TAGen outputs matching secret keys $e_i = (e_i^{(S)}, e_i^{(R)})$ and amk^* for U_i $(1 \leq i \leq n)$ and TS, respectively, as follows. TAGen calls Setup with input 1^κ, and suppose $(tuk_1^{(S)}, tuk_1^{(R)}, \ldots, tuk_n^{(S)}, tuk_n^{(R)}, tmk^*) \leftarrow \mathsf{Setup}(1^\kappa)$. Then, TAGen outputs secret keys $e_i^{(S)} := tuk_i^{(S)}, e_i^{(R)} := tuk_i^{(R)}$ and $amk^* := tmk^*$ for U_i $(1 \leq i \leq n)$ and TS, respectively.

2. $amk^{(t)} \leftarrow \mathsf{AExt}(amk^*, t)$: For a master key $amk^* = tmk^*$ and time t, AExt calls Ext, and suppose $tmk^{(t)} \leftarrow \mathsf{Ext}(tmk^*, t)$. Then, AExt outputs a time-signal at time t, $amk^{(t)} := tmk^{(t)}$.

3. $\alpha_{i_1,i_2}^{(t)} \leftarrow \mathsf{TAuth}(m, e_{i_1}^{(S)}, t, U_{i_2})$: For a message m, an authentication key $e_{i_1}^{(S)} = tuk_{i_1}^{(S)}$, the specified time t and an identity U_{i_2}, TAuth calls KeyGen, and suppose $tck_{i_1,i_2}^{(t)} \leftarrow \mathsf{KeyGen}(tuk_{i_1}^{(S)}, t, U_{i_2})$. Then, TAuth calls Auth, and it computes an authenticator $\alpha \leftarrow \mathsf{Auth}(tck_{i_1,i_2}^{(t)}, (m, t))$. Finally, TAuth outputs an authenticator at time t, $\alpha_{i_1,i_2}^{(t)} := \alpha$.

4. $true/false \leftarrow \mathsf{TVer}(m, \alpha_{i_1,i_2}^{(t)}, t, e_{i_2}^{(R)}, amk^{(t)}, U_{i_1})$: For a message m, the specified time t, an authenticator $\alpha_{i_1,i_2}^{(t)}$, a verification key $e_{i_2}^{(R)} = tuk_{i_2}^{(R)}$, a time-signal $amk^{(t)} = tmk^{(t)}$ at the specified time t and an identity U_{i_1}, TVer calls KeyDer, and suppose $tck_{i_1,i_2}^{(t)} \leftarrow \mathsf{KeyDer}(tuk_{i_2}^{(R)}, tmk^{(t)}, U_{i_1})$. Then, TVer outputs *true* if $\mathsf{Ver}(tck_{i_1,i_2}^{(t)}, (m, t), \alpha_{i_1,i_2}^{(t)}) \rightarrow true$, and outputs *false* otherwise.

The security of the above construction is shown as follows.

Theorem 4.7. *Given an ϵ-secure A-code Π_{A} and (n, ω, τ)-secure TR-KA Π_{KA} in which common keys are uniformly distributed over $\mathcal{TCK}$, then the TRA-code Π_{TRA} formed by the above construction based on Π_{A} and Π_{KA} is $(n, \omega, \tau; \epsilon)$-secure.*

Proof. First, we show the proof of $P_{S_S} \leq \epsilon$ to prove $P_{Server} \leq \epsilon$ (i.e., condition (1)). Assume that TS tries to generate a fraudulent authenticated message at

time t_2, (m', α', t_2), that will be accepted by a receiver U_{i_2}, after observing a valid authenticated message at time t_1, (m, α, t_1). Then, we have

$$
\begin{aligned}
&\max_{(m',\alpha_{i_1,i_2}^{(t_2)},t_2)} \max_{(m,\alpha_{i_1,i_2}^{(t_1)},t_1)\neq(m',\alpha_{i_1,i_2}^{(t_2)},t_2)} \max_{amk^*} \max_{amk^{(t_2)}} \\
&\quad \Pr[\mathsf{TVer}(m', \alpha_{i_1,i_2}^{(t_2)}, t_2, e_{i_2}^{(R)}, amk^{(t_2)}, U_{i_1}) = true \mid (m, \alpha_{i_1,i_2}^{(t_1)}, t_1), amk^*] \\
&= \max_{(m',\alpha',t_2)} \max_{(m,\alpha,t_1)\neq(m',\alpha',t_2)} \max_{tmk^{(t_2)}} \max_{tmk^*} \\
&\quad \Pr[\mathsf{KeyDer}(tuk_{i_2}^{(R)}, tmk^{(t_2)}, U_{i_1}) = tck_{i_1,i_2}^{(t_2)} \\
&\qquad\qquad \wedge \mathsf{Ver}((m', t_2), \alpha', tck_{i_1,i_2}^{(t_2)}) = true \mid (m, \alpha, t_1), tmk^*] \\
&= \max_{(m',\alpha',t_2)} \max_{(m,\alpha,t_1)\neq(m',\alpha',t_2)} \max_{tmk^*} \\
&\quad \Pr[\mathsf{Ver}((m', t_2), \alpha', tck_{i_1,i_2}^{(t_2)}) = true \mid (m, \alpha, t_1), tmk^*] \qquad (4.44) \\
&= \max_{(m',\alpha',t_2)} \max_{(m,\alpha,t_1)\neq(m',\alpha',t_2)} \Pr[\mathsf{Ver}((m', t_2), \alpha', C) = true \mid (m, \alpha, t_1)] \\
&\leq P_S \leq \epsilon, \qquad (4.45)
\end{aligned}
$$

where Eq. (4.44) follows from the correctness of TR-KA, Eq. (4.45) follows from Definition 4.2 (i.e., tmk^* is unhelpful to guess $tck_{i_1,i_2}^{(t_2)}$), and P_S is the success probability of substitution attacks in ϵ-secure A-codes. Thus, we have $P_{S_S} \leq \epsilon$. In a manner similar to this, we can prove $P_{I_S} \leq \epsilon$. Therefore, we have $P_{Server} = \max\{P_{I_S}, P_{S_S}\} \leq \epsilon$.

Next, we show the proof of $P_{S_1} \leq \epsilon$ to prove $P_1 \leq \epsilon$ (i.e., condition (2)). Assume that any colluding group $\mathcal{W}$ not including a targeted receiver tries to generate a fraudulent authenticated message at time t_2, (m', α', t_2), that will be accepted by a receiver U_{i_2}, after observing a valid authenticated message at time t_1, (m, α, t_1). Let $\mathrm{Info}(\mathcal{W}) := (e_{\mathcal{W}}^{(S)}, e_{\mathcal{W}}^{(R)}, amk^{(1)}, \ldots, amk^{(\tau)}) = (tuk_{\mathcal{W}}^{(S)}, tuk_{\mathcal{W}}^{(R)}, tmk^{(1)}, \ldots, tmk^{(\tau)})$. Then, we have

$$
\begin{aligned}
&\max_{(m',\alpha_{i_1,i_2}^{(t_2)},t_2)} \max_{(m,\alpha_{i_1,i_2}^{(t_1)},t_1)\neq(m',\alpha_{i_1,i_2}^{(t_2)},t_2)} \max_{\mathrm{Info}(\mathcal{W})} \\
&\quad \Pr[\mathsf{TVer}(m', \alpha_{i_1,i_2}^{(t_2)}, t_2, e_{i_2}^{(R)}, amk^{(t_2)}, U_{i_1}) = true \mid (m, \alpha_{i_1,i_2}^{(t_1)}, t_1), \mathrm{Info}(\mathcal{W})] \\
&= \max_{(m',\alpha',t_2)} \max_{(m,\alpha,t_1)\neq(m',\alpha',t_2)} \max_{\mathrm{Info}(\mathcal{W})} \\
&\quad \Pr[\mathsf{KeyDer}(tuk_{i_2}^{(R)}, tmk^{(t_2)}, U_{i_1}) = tck_{i_1,i_2}^{(t_2)} \\
&\qquad\qquad \wedge \mathsf{Ver}((m', t_2), \alpha', tck_{i_1,i_2}^{(t_2)}) = true \mid (m, \alpha, t_1), \mathrm{Info}(\mathcal{W})] \\
&= \max_{((m',t_2),\alpha',t_2)} \max_{((m,t_1),\alpha,t_1)\neq((m',t_2),\alpha',t_2)} \max_{\mathrm{Info}(\mathcal{W})} \\
&\quad \Pr[\mathsf{Ver}((m', t_2), \alpha', tck_{i_1,i_2}^{(t_2)}) = true \mid ((m, t_1), \alpha, t_1), \mathrm{Info}(\mathcal{W})] \qquad (4.46) \\
&= \max_{((m',t_2),\alpha',t_2)} \max_{((m,t_1),\alpha,t_1)\neq((m',t_2),\alpha',t_2)}
\end{aligned}
$$

$$\Pr[\mathsf{Ver}((m', t_2), \alpha', tck_{i_1,i_2}^{(t_2)}) = true \mid ((m, t_1), \alpha, t_1)] \tag{4.47}$$
$$\leq P_S \leq \epsilon.$$

where Eq. (4.46) follows from the correctness of TR-KA, Eq. (4.47) follows from Definition 4.2 (i.e., $\mathrm{Info}(\mathcal{W})$ is unhelpful to guess $tck_{i_1,i_2}^{(t_2)}$), and P_S is the success probability of substitution attacks in ϵ-secure A-codes. Thus, we have $P_{S_1} \leq \epsilon$. In a manner similar to this, we can prove $P_{I_1} \leq \epsilon$. Therefore, we have $P_1 = \max\{P_{I_1}, P_{S_1}\} \leq \epsilon$.

Finally, we show the proof of $P_2 \leq \epsilon$ (i.e., condition (3)). Assume that any colluding group $\mathcal{W}$ including a legitimate (but dishonest) receiver tries to check the validity of a target authenticated message without a time-signal at the specified time, even if $\mathcal{W}$ obtains time-signals at all the time except the specified time. Let

$$\begin{aligned}\mathrm{Info}(\mathcal{W}) &:= (e_{\mathcal{W}}^{(S)}, e_{\mathcal{W}}^{(R)}, amk^{(1)}, \ldots, amk^{(t-1)}, amk^{(t+1)}, \ldots, amk^{(\tau)}) \\ &= (tuk_{\mathcal{W}}^{(S)}, tuk_{\mathcal{W}}^{(R)}, tmk^{(1)}, \ldots, tmk^{(t-1)}, tmk^{(t+1)}, \ldots, tmk^{(\tau)}).\end{aligned}$$

Then, we have

$$\begin{aligned}&\max_{(m,\alpha_{i_1,i_2}^{(t)},t)} \max_{\mathrm{Info}(\mathcal{W})} \Pr[\mathsf{TVer}(m, \alpha_{i_1,i_2}^{(t)}, t, e_{i_2}^{(R)}, amk^{(t)}, U_{i_1}) = true \mid \mathrm{Info}(\mathcal{W})] \\ &= \max_{(m,\alpha_{i_1,i_2}^{(t)},t)} \max_{\mathrm{Info}(\mathcal{W})} \Pr[\mathsf{KeyDer}(tuk_{i_2}^{(R)}, tmk^{(t)}, U_{i_1}) = tck_{i_1,i_2}^{(t)} \\ &\qquad\qquad \wedge \mathsf{Ver}((m,t), \alpha, tck_{i_1,i_2}^{(t)}) = true \mid \mathrm{Info}(\mathcal{W})] \\ &\leq \max_{(m,\alpha_{i_1,i_2}^{(t)},t)} \max_{\mathrm{Info}(\mathcal{W})} \Pr[\mathsf{Ver}((m,t), \alpha, tck_{i_1,i_2}^{(t)}) = true \mid \mathrm{Info}(\mathcal{W})] \\ &= \max_{(m,\alpha_{i_1,i_2}^{(t)},t)} \Pr[\mathsf{Ver}((m,t), \alpha, tck_{i_1,i_2}^{(t)}) = true] \qquad (4.48) \\ &\leq P_I \leq \epsilon.\end{aligned}$$

where Eq. (4.48) follows from Definition 4.2 (i.e., $\mathrm{Info}(\mathcal{W})$ is unhelpful to guess $tck_{i_1,i_2}^{(t_2)}$) and P_I is the success probability of impersonation attacks in ϵ-secure A-codes. Thus, we have $P_2 \leq \epsilon$. □

Remark 4.3. *Even if we apply optimal constructions of TR-KA and A-codes in the above generic construction, we cannot obtain an optimal construction of TRA-codes. Actually, consider the optimal construction of TR-KA in Section 4.2.3 and the well-known optimal construction of A-codes (see Section 2.6.1). We can quite smoothly combine these constructions in our generic construction since both ones are given based on polynomials over $\mathbb{F}_q$. However, the resulting construction of TRA-codes is not optimal. Therefore, in Section 4.4.4 we will show that there exists a direct construction (i.e., a construction from scratch) which satisfies Definition 4.9.*

4.4.4 Direct Construction of TRA-codes by Polynomials over Finite Fields

We propose a direct construction of $(n, \omega, \tau; \epsilon)$-secure TRA-codes. In addition, it is shown that the construction is optimal. The detail of our construction of TRA-codes Π_{TRA}=(AGen, AExt, TAuth, TVer) is given as follows.

1. $(e_1, \ldots, e_n, amk^*) \leftarrow \mathsf{TAGen}(1^\kappa)$: For a security parameter 1^κ, Setup outputs matching secret keys tuk_i and tmk^* for U_i $(1 \le i \le n)$ and TS, respectively, as follows. AGen picks a k-bit prime power q, where $q > \max\{n, \tau\}$, and constructs the finite field $\mathbb{F}_q$ with q elements. We assume that the identity of each user U_i is encoded as $U_i \in \mathbb{F}_q \backslash \{0\}$. Also, we assume $\mathcal{T} = [\tau] \subset \mathbb{F}_q \backslash \{0\}$ by using appropriate encoding. And, AGen chooses uniformly at random $f(x,y) := \sum_{i=0}^{\omega}\sum_{j=0}^{\omega} a_{ij}x^i y^j$, $g(x,y) := \sum_{i=0}^{\omega}\sum_{j=0}^{\omega} b_{ij}x^i y^j$, $amk^*(x,z) := \sum_{i=0}^{\omega}\sum_{k=0}^{\tau-1} c_{ik}x^i z^k$ over $\mathbb{F}_q$ with three variables x, y and z in which each degree of x and y is at most ω, and the degree of z is at most $\tau - 1$. AGen also computes $e_i^{(S)} := (g_i^{(S)}(y) := g(U_i, y), F_i^{(S)}(y,z) := f(U_i, y) + amk^*(U_i, z))$ and $e_i^{(R)} := (g_i^{(R)}(x) := g(x, U_i), f_i^{(R)}(x) := f(x, U_i))$ $(1 \le i \le n)$. Then, AGen outputs secret keys $e_i := (e_i^{(S)}, e_i^{(R)})$ $(1 \le i \le n)$ and $amk^* := amk^*(x,z)$ for U_i $(1 \le i \le n)$ and TS, respectively.

2. $amk^{(t)} \leftarrow \mathsf{AExt}(amk^*, t)$: For $amk^* = amk^*(x,z)$ and time $t \in \mathcal{T}$, AExt outputs a time-signal at time t, $amk^{(t)}(x) := amk^*(x,t)$.

3. $\alpha_{i_1,i_2}^{(t)} \leftarrow \mathsf{TAuth}(m, e_{i_1}^{(S)}, t, U_{i_2})$: For a message m, a secret key $e_{i_1}^{(S)}$, the specified time t and an identity U_{i_2}, TAuth generates an authenticator, $\alpha_{i_1,i_2}^{(t)} := g_{i_1}^{(S)}(U_{i_2})m + F_{i_1}^{(S)}(U_{i_2}, t)$, and outputs it.

4. $true/false \leftarrow \mathsf{TVer}(m, \alpha_{i_1,i_2}^{(t)}, t, e_{i_2}^{(R)}, amk^{(t)}, U_{i_1})$: For the message m, the authenticator $\alpha_{i_1,i_2}^{(t)}$, the specified time t, a secret key $e_{i_2}^{(R)}$, a time-signal $amk^{(t)}$ at the specified time t and an identity U_{i_1}, TVer outputs *true* if $\alpha_{i_1,i_2}^{(t)} = g_{i_2}^{(R)}(U_{i_1})m + f_{i_2}^{(R)}(U_{i_1}) + amk^{(t)}(U_{i_1})$ holds, and otherwise outputs *false*.

The security and optimality of the above construction is stated as follows.

Theorem 4.8. *The resulting TRA-code* Π_{TRA} *by the above construction is* $(n, \omega, \tau; 1/q)$*-secure and optimal.*

Proof. In this proof, we can write $f(x,y)$, $g(x,y)$ and $tmk^*(x,z)$ in the form of

$$f(x,y) := (1, x, \ldots, x^\omega)A \begin{pmatrix} 1 \\ y \\ \vdots \\ y^\omega \end{pmatrix}, \; g(x,y) := (1, x, \ldots, x^\omega)B \begin{pmatrix} 1 \\ y \\ \vdots \\ y^\omega \end{pmatrix}, \text{ and}$$

$$amk^*(x,z) := (1, x, \ldots, x^\omega)C \begin{pmatrix} 1 \\ z \\ \vdots \\ z^{\tau-1} \end{pmatrix},$$

respectively, where A and B are $(\omega+1) \times (\omega+1)$ matrices and C is an $(\omega+1) \times \tau$ matrix, respectively. To complete the proof of Theorem 4.8, we show the following lemmas.

Lemma 4.19. *The above construction satisfies* $P_{Server} \leq \frac{1}{q}$.

Proof. First, we show the proof of $P_{S_S} \leq 1/q$. Assume that TS will generate a fraudulent authenticated message at time t_2 $(m', \alpha_{i_1,i_2}^{(t_2)}, t_2)$, under the following conditions: TS can obtain a valid authenticated message $(m, \alpha_{i_1,i_2}^{(t_1)}, t_1)$ where $m \neq m'$ and knows his master key amk^*. To begin with, since TS knows amk^*, he can compute $amk^*(U_{i_1}, t_2)$. Therefore, he tries to generate $g(U_{i_1}, U_{i_2})m' + f(U_{i_1}, U_{i_2})$. Moreover, TS can obtain $g(U_{i_1}, U_{i_2})m + f(U_{i_1}, U_{i_2})$ by calculating $\alpha_{i_1,i_2}^{(t_1)} - amk^*(U_{i_1}, t_1)$. However, by applying $X := O$, $A := A$ and $Y := O$ in Lemma 4.5, there are at least q candidates of A. Then, by applying $\boldsymbol{x} := (1, U_{i_1}, U_{i_1}^2, \ldots, U_{i_1}^\omega)$, $A := A$ and $\boldsymbol{y} := {}^t(1, U_{i_2}, U_{i_2}^2, \ldots, U_{i_2}^\omega)$ in Lemma 4.6, TS cannot guess $f(U_{i_1}, U_{i_2}) = \boldsymbol{x}A\boldsymbol{y}$ with probability larger than $1/q$. In a similar way, we can prove that TS cannot guess $g(U_{i_1}, U_{i_2})$ with probability larger than $1/q$. Hence, $P_{S_S} \leq 1/q$. We can also prove $P_{I_S} \leq 1/q$. Thus, we have $P_{Server} = \max(P_{I_S}, P_{S_S}) \leq 1/q$. □

Lemma 4.20. *The above construction satisfies* $P_1 \leq \frac{1}{q}$.

Proof. First, we show the proof of $P_{S_1} \leq 1/q$. Without loss of generality, we consider that $\mathcal{W} := \{U_1, \ldots, U_\omega\}$ is a set of colluders such that $U_{i_1}, U_{i_2} \notin \mathcal{W}$, and we write $\boldsymbol{x_i} := (1, U_i, U_i^2, \ldots, U_i^\omega)$ $(1 \leq i \leq n)$. Assume that $\mathcal{W}$ will generate a fraudulent authenticated message at time t_2 $(m', \alpha_{i_1,i_2}^{(t_2)}, t_2)$, under the following conditions: $\mathcal{W}$ can obtain ω user's secret keys, all time-signals, and a valid authenticated message $(m, \alpha_{i_1,i_2}^{(t_1)}, t_1)$ where $m \neq m'$. Note that $\mathcal{W}$ can compute amk^* by all time-signals and calculate $amk^*(U_{i_1}, t_1)$ and $amk^*(U_{i_1}, t_2)$. Therefore, $\mathcal{W}$ tries to generate $g(U_{i_1}, U_{i_2})(m - m')$ to succeed in this substitution attack, since $\alpha_{i_1,i_2}^{(t_2)} = \alpha_{i_1,i_2}^{(t_1)} - g(U_{i_1}, U_{i_2})(m - m') - amk^*(U_{i_1}, t_1) + amk^*(U_{i_1}, t_2)$. $\mathcal{W}$ can compute $g(U_l, y)$ $(1 \leq l \leq \omega)$ by using $e_l^{(S)}(y, z)$ and $amk^*(U_l, z)$. Hence, $\mathcal{W}$ gets

$$g(U_l, y) = \boldsymbol{x}_l B \begin{pmatrix} 1 \\ y \\ \vdots \\ y^\omega \end{pmatrix},$$

for $1 \leq l \leq \omega$. Thus, $\mathcal{W}$ can know the following matrix:

$$X_U B := \begin{pmatrix} \boldsymbol{x}_1 \\ \boldsymbol{x}_2 \\ \vdots \\ \boldsymbol{x}_\omega \end{pmatrix} B.$$

In addition, $\mathcal{W}$ knows

$$g(x, U_l) = (1, x, \ldots, x^\omega) B \, {}^t\boldsymbol{x}_l,$$

for $1 \leq l \leq \omega$ by their verification keys $f(x, U_l)$ $(1 \leq l \leq \omega)$. Thus, $\mathcal{W}$ can know the following matrix:

$$B \, {}^t X_U := B \, ({}^t\boldsymbol{x}_1, {}^t\boldsymbol{x}_2, \cdots, {}^t\boldsymbol{x}_\omega).$$

By applying $X := X_U$, $A := B$ and $Y := {}^t X_U$ in Lemma 4.5, there are at least q candidates of B. In addition, $\{\boldsymbol{x}_{i_1}, \boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_\omega\}$ and $\{\boldsymbol{x}_{i_2}, \boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_\omega\}$ are linearly independent, respectively, since $U_{i_1}, U_{i_2} \notin \mathcal{W}$. Therefore, $\mathcal{W}$ cannot guess $g(U_{i_1}, U_{i_2}) = \boldsymbol{x}_{i_1} B \, {}^t\boldsymbol{x}_{i_2}$ with probability larger than $1/q$ by Lemma 4.6. Hence, $P_{S_1} \leq 1/q$. We can also prove $P_{I_1} \leq 1/q$. Thus, we have $P_1 = \max(P_{I_1}, P_{S_1}) \leq 1/q$. □

Lemma 4.21. *The above construction satisfies* $P_2 \leq \frac{1}{q}$.

Proof. Without loss of generality, we suppose that $\mathcal{W} := \{U_1, \ldots, U_\omega\}$ is a set of colluders such that $U_{i_1} \notin \mathcal{W}$, U_{i_1} is a targeted sender, U_ω is a targeted receiver, and τ is a specified time. In addition, we write $\boldsymbol{x_i} := (1, U_i, U_i^2, \ldots, U_i^\omega)$ $(1 \leq i \leq n)$ and $\boldsymbol{y_i} := {}^t(1, i, i^2, \ldots, i^{\tau-1})$ $(1 \leq i \leq \tau)$. To succeed in the substitution attack by a group of colluders $\mathcal{W}$, $\mathcal{W}$ will try to check the validity of a target authenticated message without a time-signal at the specified time under the following conditions: $\mathcal{W}$ can obtain ω user's secret keys, time-signals at all the time except the specified time τ, and a valid authenticated message $(m, \alpha_{i_1,\omega}^{(t_1)}, t)$. Note that $\mathcal{W}$ can get $f(U_{i_1}, U_\omega)$ and $g(U_{i_1}, U_\omega)$ since $U_\omega \in \mathcal{W}$. Thus, $\mathcal{W}$ tries to obtain $amk^*(x, z)$ to know $f(U_{i_1}, U_\omega) + amk^*(U_{i_1}, \tau)$. $\mathcal{W}$ can compute $amk^*(U_l, z)$ $(1 \leq l \leq \omega)$ by using $e_l^{(S)}(y, z)$ and $f(U_l, z)$. Hence, $\mathcal{W}$ gets

$$amk^*(U_l, z) = \boldsymbol{x}_l C \begin{pmatrix} 1 \\ z \\ \vdots \\ z^{\tau-1} \end{pmatrix},$$

for $1 \leq l \leq \omega$. Thus, $\mathcal{W}$ can know the following matrix:

$$X_U C := \begin{pmatrix} \boldsymbol{x}_1 \\ \boldsymbol{x}_2 \\ \vdots \\ \boldsymbol{x}_\omega \end{pmatrix} C.$$

In addition, $\mathcal{W}$ obtains $amk^*(x,t) = (1, x, \ldots, x^\omega)C\boldsymbol{y}_t$ for $1 \leq t \leq \tau - 1$ by time-signals at all except the time τ. Thus, $\mathcal{W}$ can know the following matrix:

$$CY_T := C(\boldsymbol{y}_1, \boldsymbol{y}_2, \cdots, \boldsymbol{y}_{\tau-1}).$$

By applying $X := X_U$, $A := C$ and $Y := Y_T$ in Lemma 4.5, there are at least q candidates of C. In addition, $\{\boldsymbol{x}_{i_1}, \boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_\omega\}$ and $\{\boldsymbol{y}_1, \boldsymbol{y}_2, \ldots, \boldsymbol{y}_\tau\}$ are linearly independent, respectively, since $U_{i_1} \notin \mathcal{W}$. Therefore, $\mathcal{W}$ cannot guess $amk^*(U_{i_1}, z) = \boldsymbol{x}_{i_1} C \boldsymbol{y}_\tau$ with probability larger than $1/q$ by Lemma 4.6. Thus, we have $P_2 \leq 1/q$. □

Proof of Theorem 4.8. It follows that $\max\{P_{Server}, P_1, P_2\} \leq 1/q$ from the above lemmas. Finally, it is straightforward to see that the construction satisfies all the lower bounds in Theorem 4.6 with equalities. Therefore, the above construction is optimal. □

4.5 Timed-Release Secret Sharing

As explained in Section 4.1, we conceive the following two types of schemes.

One is a SS scheme such that information associated with time (called *time-signals*) is required whenever a secret is reconstructed, which means a SS scheme with a simple combination of traditional secret sharing functionality and timed-release functionality. For realizing it, we propose (k,n)-TR-SS. In (k,n)-TR-SS, a dealer can specify positive integers k, n with $k \leq n$, where n is the number of participants and k is a threshold value, and future time when a secret can be recovered; and the secret can be reconstructed from at least k shares and a time-signal at the specified time. On the other hand, participants cannot reconstruct the secret without the time-signal even if they can obtain all shares. Specifically, we define a model and security notions of (k,n)-TR-SS, and we derive lower bounds on the sizes of shares, time-signals, and entities' secret keys required for (k,n)-TR-SS. Moreover, we provide a direct construction of (k,n)-TR-SS, which is constructed by using polynomials over finite fields and provably secure in our security definition. In addition, we show that the direct construction meets the lower bounds on the sizes of shares, time-signals, and entities' secret keys with equalities. Therefore, it turns out that our lower bounds are tight, and that the direct construction is optimal.

Another one is a *hybrid* TR-SS, which means a SS scheme in which traditional secret sharing functionality and timed-release functionality are simultaneously realized. In our hybrid TR-SS, a secret can be reconstructed, if one of the following condition is satisfied: a secret can be reconstructed from k_1 shares and a time-signal at a specified time as in the (k_1, n)-TR-SS; or a secret can be reconstructed from k_2 shares as in the traditional (k_2, n)-SS. Hence, we consider two threshold values k_1, k_2 to define a model of the hybrid TR-SS, and we propose (k_1, k_2, n)-TR-SS as such a model, where $k_1 \leq k_2 \leq n$. Specifically, in (k_1, k_2, n)-TR-SS, a dealer can specify future time, and arbitrarily chooses k_1, k_2 and n. At least k_1 (and less than k_2) participants can reconstruct a secret with a time-signal at the specified time, and at least k_2 participants can reconstruct a secret *without* any time-signal (i.e. they can reconstruct from *only* their shares). Specifically, we define a model and security notions of (k_1, k_2, n)-TR-SS, and we derive *tight* lower bounds on the sizes of shares, time-signals, and entities' secret keys required for (k_1, k_2, n)-TR-SS. Moreover, we provide a direct constructions of (k_1, k_2, n)-TR-SS, which is an *optimal* construction, which meets the above lower bounds with equalities. To achieve its optimality, we use a public parameter, which is needed to reconstruct a secret, in our construction. This technique is reasonable since public parameters are sometimes used in the context of SS schemes such as [80].

In particular, a theoretically-interesting point in our results includes that the timed-release security can be realized without any additional redundancy on the share size in both schemes.

4.5.1 The Model and Security Definition of (k, n)-TR-SS

In this section, we propose a model and a security definition of (k, n)-TR-SS. In (k, n)-TR-SS, a time-signal at the specified time is always required when a secret is reconstructed. In other words, a secret cannot be reconstructed without a time-signal at the specified time even if there are all shares.

First, we introduce the model of (k, n)-TR-SS. Unlike traditional SS schemes, we assume that there is a trusted initializer. In (k, n)-TR-SS, there are $n + 3$ entities, a dealer D, n participants $P_1, P_2, \ldots, P_n$, a time-server TS for broadcasting time-signals at most τ times and a trusted initializer TA, where k, n and τ are positive integers. We assume that the identity of each user P_i is also denoted by P_i.

Informally, (k, n)-TR-SS is executed as follows. First, TA generates secret keys on behalf of D and TS. After distributing these keys via secure channels, TA deletes it in his memory. Next, D specifies future time, as D wants, when a secret is reconstructed by participants, and he generates n shares from the secret by using his secret key. And, D sends each share to each participant respectively via secure channels. The time-server TS periodically broadcasts a time-signal which is generated by using his secret key. Note that there is no interaction between TS and D, hence TS may not know when the specified

time is. When the specified time has come, at least k participants can compute the secret by using their shares and the time-signal of the specified time.

Formally, we give the definition of (k, n)-TR-SS as follows. In this model, let $\mathcal{P} := \{P_1, P_2, \ldots, P_n\}$ be a set of all participants. And also, $\mathcal{S}$ is a set of possible secrets with a probability distribution P_S, and $\mathcal{SK}$ is a set of possible secret keys. $\mathcal{T} := [\tau]$ is a set of time. Let $\mathcal{U}_i^{(t)}$ be the set of possible P_i's shares at the time $t \in \mathcal{T}$. Also, $\mathcal{U}_i := \bigcup_{t=1}^{\tau} \mathcal{U}_i^{(t)}$ is a set of possible P_i's shares for every $i \in [n]$, and let $\mathcal{U} := \bigcup_{i=1}^{n} \mathcal{U}_i$. In addition, $\mathcal{TI}^{(t)}$ is a set of time-signals at time t, and let $\mathcal{TI} := \bigcup_{t=1}^{\tau} \mathcal{TI}^{(t)}$. Furthermore, for any subset of participants $\mathcal{J} = \{P_{i_1}, \ldots, P_{i_j}\} \subset \mathcal{P}$, $\mathcal{U}_{\mathcal{J}}^{(t)} := \mathcal{U}_{i_1}^{(t)} \times \cdots \times \mathcal{U}_{i_j}^{(t)}$ denotes the set of possible shares held by $\mathcal{J}$.

Definition 4.10 ((k, n)-TR-SS). *A (k, n)-TR-SS scheme Π_{TSS} involves $n + 3$ entities, $TA, D, P_1, \ldots, P_n$, and TS, and consists of four phases, Initialize, Extract, Share and Reconstruct, and five finite spaces, $\mathcal{S}, \mathcal{SK}, \mathcal{U}, \mathcal{T}$, and $\mathcal{TI}$. Π_{TSS} is executed based on the above phases as follows.*

1. ***Initialize.*** *TA generates a secret key $sk \in \mathcal{SK}$ for TS and D. This key is distributed to corresponding entities via secure channels. After distributing the secret key, TA deletes it from his memory. And, D and TS keep their keys secret, respectively.*[4]

2. ***Share.*** *A dealer D randomly selects a secret $s \in \mathcal{S}$ according to P_S, and chooses k and n. If D wants the secret s to be reconstructed by participants at future time $t \in \mathcal{T}$, on input the secret $s \in \mathcal{S}$, specified time $t \in \mathcal{T}$ and a secret key sk, D computes a share $u_i^{(t)} \in \mathcal{U}_i^{(t)}$ for every P_i $(i = 1, 2, \ldots, n)$. And then, D sends a pair of the share and specified time, $(u_i^{(t)}, t)$, to P_i $(i = 1, 2, \ldots, n)$ via a secure channel.*[5]

3. ***Extract.*** *For broadcasting a time-signal at each time t, TS generates a time-signal $ts^{(t)} \in \mathcal{TI}^{(t)}$ by using his secret key sk and time $t \in \mathcal{T}$, where for simplicity we assume that $ts^{(t)}$ is deterministically computed by t and sk.*

4. ***Reconstruct.*** *At the specified time t, any set of at least k participants $\mathcal{A} = \{P_{i_1}, \ldots, P_{i_j}\} \in \mathcal{PS}(\mathcal{P}, k, n)$ can reconstruct the secret s by using their shares $u_{i_1}^{(t)}, \ldots, u_{i_j}^{(t)}$ $(k \leq j \leq n)$ and a time-signal $ts^{(t)}$ at the specified time.*

[4] If we consider a situation in which TS is trusted and has functionality of generating keys and distributing them to participants by secure private channels, we can identify TA with TS in the situation. However, there may be a situation in which the roles of TA and TS are quite different (e.g., TA is a provider of secure data storage service and TS is a time-signal broadcasting server). Therefore, we assume two entities TA and TS in our model to capture various situations.

[5] More precisely, there is no need to keep the specified time confidential (D only has to send shares via secure channels).

In the above model, we assume that Π_{TSS} meets the following *correctness* property: If D correctly completes the phase *Share* and TS correctly completes the phase *Extract*, then, for all possible $i \in [n]$, $t \in \mathcal{T}$, $s \in \mathcal{S}$, $u_i^{(t)} \in \mathcal{U}_i$, and $ts^{(t)} \in \mathcal{TI}^{(t)}$, it hold that any $\mathcal{A} \in \mathcal{PS}(\mathcal{P}, k, n)$ will correctly reconstruct the secret s at the end of phase *Reconstruct*, namely, $H(S \mid U_{\mathcal{A}}^{(t)}, TI^{(t)}) = 0$, where S, $U_{\mathcal{A}}^{(t)}$, and $TI^{(t)}$ are random variables which take values in $\mathcal{S}$, $\mathcal{U}_{\mathcal{A}}^{(t)}$, and $\mathcal{TI}^{(t)}$, respectively.

Next, we formalize a security definition of (k, n)-TR-SS based on the idea of the timed-release security and SS schemes. Although the concept of TR-SS is similar to that of TR-CSS, we traditionally formalize the security of TR-SS by Shannon entropy. In (k, n)-TR-SS, we consider the following two kinds of security. The first security which we consider is basically the same as that of the traditional (k, n)-SS: less than k participants cannot obtain any information on a secret. In addition to this, as the second security we want to require that even at least k participants cannot obtain any information on a secret before the specified time comes (i.e., before a time-signal at the specified time is received). Let S, $U_{\mathcal{J}}^{(t)}$ $(\mathcal{J} \subset \mathcal{P})$, and $TI^{(t)}$ are random variables which take values in $\mathcal{S}$, $\mathcal{U}_{\mathcal{J}}^{(t)}$ $(\mathcal{J} \subset \mathcal{P})$, and $\mathcal{TI}^{(t)}$, respectively. Therefore, we formally define secure (k, n)-TR-SS as follows.

Definition 4.11 (Security of (k, n)-TR-SS). *Let Π_{TSS} be a (k, n)-TR-SS scheme. Π_{TSS} is said to be secure if the following conditions are satisfied:*

(i) For any $\mathcal{F} \in \mathcal{PS}(\mathcal{P}, 1, k-1)$ and any $t \in \mathcal{T}$, it holds that

$$H(S \mid U_{\mathcal{F}}^{(t)}, TI^{(1)}, \ldots, TI^{(\tau)}) = H(S).$$

(ii) For any $\mathcal{A} \in \mathcal{PS}(\mathcal{P}, k, n)$ and any $t \in \mathcal{T}$, it holds that

$$H(S \mid U_{\mathcal{A}}^{(t)}, TI^{(1)}, \ldots, TI^{(t-1)}, TI^{(t+1)}, \ldots, TI^{(\tau)}) = H(S).$$

Intuitively, the meaning of two conditions (i) and (ii) in Definition 4.11 is explained as follows. (i) No information on a secret is obtained by any set of less than k participants, even if they obtain time-signals at all the time; (ii) No information on a secret is obtained by any set of more than $k-1$ participants, even if they obtain time-signals at all the time except the specified time.

Remark 4.4. *We can also consider the following security definition (the condition (iii)) instead of (i): No information on a secret is obtained by collusion of TS and any set of less than k participants, namely, this is defined as follows.*

(iii) For any $\mathcal{F} \in \mathcal{PS}(\mathcal{P}, 1, k-1)$ and for any $t \in \mathcal{T}$, it holds that $H(S \mid U_{\mathcal{F}}^{(t)}, SK) = H(S)$.

Note that the condition (iii) is stronger than (i). However, we do not consider (iii) in this paper because of the following two reasons: first, the condition (i) is more natural than (iii), since it does not seem natural to consider the situation that any set of less than k participants colludes with TS in the real world; and secondly, our lower bounds in Theorem 4.9 are still valid even under the conditions (ii) and (iii), in other words, even if we consider the conditions (ii) and (iii), we can derive the same lower bounds in Theorem 4.9 since Definition 4.11 is weaker. Interestingly, our direct construction in Section 4.5.3 also satisfies (iii), and tightness of our lower bounds and optimality of our direct construction will be valid not depending on the choice of the condition (i) or (iii). Furthermore, we do not have to consider an attack by dishonest TS only, since TS's master key is generated independently of a secret.

4.5.2 Lower Bounds Required for (k, n)-TR-SS

In this section, we show lower bounds on sizes of shares, time-signals, and secret keys required for secure (k, n)-TR-SS as follows.

Theorem 4.9. *Let Π_{TSS} be any secure (k, n)-TR-SS scheme. Then, for any $i \in [n]$ and for any $t \in \mathcal{T}$, we have*

$$(I)\ H(U_i^{(t)}) \geq H(S), \qquad (II)\ H(TI^{(t)}) \geq H(S),$$
$$(III)\ H(SK) \geq \tau H(S).$$

Proof. The proof of Theorem 4.9 follows from the following lemmas.

Lemma 4.22. *$H(U_i^{(t)}) \geq H(S)$ for any $i \in [n]$ and any $t \in \mathcal{T}$.*

Proof. The proof can be proved in a way similar to the proof in [82, Theorem 1]. For arbitrary $i \in [n]$, we take a subset $\mathcal{B}_i \in \mathcal{PS}(\mathcal{P} \setminus \{P_i\}, k-1, k-1)$ of participants. Then, for any $t \in \mathcal{T}$, we have

$$\begin{aligned} H(U_i^{(t)}) &\geq H(U_i^{(t)} \mid U_{\mathcal{B}_i}^{(t)}, TI^{(t)}) \geq I(S; U_i^{(t)} \mid U_{\mathcal{B}_i}^{(t)}, TI^{(t)}) \\ &= H(S \mid U_{\mathcal{B}_i}^{(t)}, TI^{(t)}) \qquad (4.49) \\ &= H(S), \qquad (4.50) \end{aligned}$$

where Eq. (4.49) follows from the correctness of (k, n)-TR-SS and Eq. (4.50) follows from the condition (i) in Definition 4.11. □

Lemma 4.23. *$H(TI^{(t)} \mid TI^{(1)}, \ldots, TI^{(t-1)}) \geq H(S)$ for any $t \in \mathcal{T}$. In particular, $H(TI^{(t)}) \geq H(S)$ for any $t \in \mathcal{T}$.*

Proof. For any $\mathcal{A} \in \mathcal{PS}(\mathcal{P}, k, n)$ and any $t \in \mathcal{T}$, we have

$$\begin{aligned} H(TI^{(t)}) \geq & H(TI^{(t)} \mid TI^{(1)}, \ldots, TI^{(t-1)}) \\ \geq & H(TI^{(t)} \mid U_{\mathcal{A}}^{(t)}, TI^{(1)}, \ldots, TI^{(t-1)}) \end{aligned}$$

$$
\begin{aligned}
&\geq I(S; TI^{(t)} \mid U_{\mathcal{A}}^{(t)}, TI^{(1)}, \ldots, TI^{(t-1)}) \\
&= H(S \mid U_{\mathcal{A}}^{(t)}, TI^{(1)}, \ldots, TI^{(t-1)}) \qquad (4.51) \\
&= H(S), \qquad (4.52)
\end{aligned}
$$

where Eq. (4.51) follows from the correctness of (k, n)-TR-SS and Eq. (4.52) follows from the condition (ii) in Definition 4.11. □

Lemma 4.24. $H(SK) \geq \tau H(S)$.

Proof. We have

$$
\begin{aligned}
H(SK) &\geq I(TI^{(1)}, \ldots, TI^{(\tau)}; SK) \\
&= H(TI^{(1)}, \ldots, TI^{(\tau)}) - H(TI^{(1)}, \ldots, TI^{(\tau)} \mid SK) \\
&= H(TI^{(1)}, \ldots, TI^{(\tau)}) \\
&= \sum_{t=1}^{\tau} H(TI^{(t)} \mid TI^{(1)}, \ldots, TI^{(t-1)}) \geq \tau H(S),
\end{aligned}
$$

where the last inequality follows from Lemma 4.23. □

Proof of Theorem 4.9: From Lemmas 4.22–4.24, the proof of Theorem 4.9 is completed. □

As we will see in Section 4.5.3, the above lower bounds are tight since our construction will meet all the above lower bounds with equalities. Therefore, we define optimality of constructions of (k, n)-TR-SS as follows.

Definition 4.12. *A construction of secure (k, n)-TR-SS is said to be optimal if it meets equality in every bound of (i)–(iii) in Theorem 4.9.*

Remark 4.5. *The secret sharing scheme such that the size of each participant's share is equal to that of the secret is often called an ideal secret sharing scheme. The construction of (k, n)-TR-SS in Section 4.5.3 is optimal, hence, in this sense we achieve ideal (k, n)-TR-SS. In terms of the share size, an interesting point is that the timed-release property can be realized without any additional redundancy on the share size. Therefore in the sense of the bound on the share size, our results are also regarded as the extension of traditional secret sharing schemes.*

4.5.3 Direct Construction of (k, n)-TR-SS

We propose a direct construction of (k, n)-TR-SS. In addition, it is shown that our construction is optimal. The detail of our construction of a (k, n)-TR-SS scheme Π_{TSS} is given as follows.

1. *Initialize.* Let q be a prime power, where $q > \max\{n, \tau\}$, and let $\mathbb{F}_q$ be the finite field with q elements. We assume that the identity of each participant P_i is encoded as $P_i \in \mathbb{F}_q \backslash \{0\}$. Also, we assume $\mathcal{T} = \{1, 2, \ldots, \tau\} \subset \mathbb{F}_q \backslash \{0\}$ by using appropriate encoding. First, TA chooses uniformly at random τ numbers $r^{(j)} (j = 1, \ldots, \tau)$ from $\mathbb{F}_q$. TA sends a secret key $sk := (r^{(1)}, \ldots, r^{(\tau)})$ to TS and D via secure channels, respectively.

2. *Share.* First, D randomly chooses a secret $s \in \mathbb{F}_q$ according to a distribution P_S over $\mathbb{F}_q$. Also, D specifies the time t at which participants can reconstruct the secret. Next, D randomly chooses a polynomial $f(x) := c^{(t)} + \sum_{i=1}^{k-1} a_i x^i$ over $\mathbb{F}_q$, where $c^{(t)}$ is computed by $c^{(t)} := s + r^{(t)}$ and each coefficient a_i is randomly and uniformly chosen from $\mathbb{F}_q$. Finally, D computes $u_i^{(t)} := f(P_i) (i = 1, 2, \ldots, n)$ and sends $(u_i^{(t)}, t)$ to $P_i (i = 1, 2, \ldots, n)$ via a secure channel.

3. *Extract.* For sk and time $t \in \mathcal{T}$, TS broadcasts t-th key $r^{(t)}$ as a time-signal at time t to all participants via a (authenticated) broadcast channel.

4. *Reconstruct.* First, a set of at least k participants $\mathcal{A} = \{P_{i_1}, P_{i_2}, \ldots, P_{i_k}\} \in \mathcal{PS}(\mathcal{P}, k, k)$ computes $c^{(t)}$ by Lagrange interpolation from their k shares:

$$c^{(t)} = \sum_{j=1}^{k} \left(\prod_{l \neq j} \frac{P_{i_j}}{P_{i_j} - P_{i_l}} \right) f(P_{i_j}).$$

   After receiving $ts^{(t)} = r^{(t)}$, they can compute and get $s = c^{(t)} - r^{(t)}$.

The security and optimality of the above construction is stated as follows.

Theorem 4.10. *The resulting (k, n)-TR-SS Π_{TSS} by the above construction is secure and optimal.*

Proof. First, we show the proof of (i) in Definition 4.11. Assume that any $k-1$ participants $\mathcal{F} = \{P_{i_1}, \ldots, P_{i_{k-1}}\} \in \mathcal{PS}(\mathcal{P}, k-1, k-1)$ try to guess $c^{(t)}$ by using their shares. Note that they know $r^{(t)} = c^{(t)} - s$ and

$$f(P_{i_j}) = (1, P_{i_j}, \ldots, P_{i_j}^{k-1}) \begin{pmatrix} c^{(t)} \\ a_1 \\ \vdots \\ a_{k-1} \end{pmatrix},$$

for $j = 1, \ldots, k-1$. Thus, they can know the following matrix:

$$\begin{pmatrix} 1 & P_{i_1} & \cdots & P_{i_1}^{k-1} \\ 1 & P_{i_2} & \cdots & P_{i_2}^{k-1} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & P_{i_{k-1}} & \cdots & P_{i_{k-1}}^{k-1} \end{pmatrix} \begin{pmatrix} c^{(t)} \\ a_1 \\ \vdots \\ a_{k-1} \end{pmatrix}. \tag{4.53}$$

However, from Eq. (4.53), they cannot guess at least one element of $(c^{(t)}, a_1, \ldots, a_{k-1})$ with probability larger than $1/q$. Therefore, $H(S \mid U_{\mathcal{F}}^{(t)}, TI^{(1)}, \ldots, TI^{(\tau)}) = H(S)$ for any $\mathcal{F} \in \mathcal{PS}(\mathcal{P}, 1, k-1)$ and any $t \in \mathcal{T}$.

Next, we show the proof of (ii) in Definition 4.11. Without loss of generality, we suppose that τ is a specified time, and that all participants try to guess $r^{(\tau)}$ by using $c^{(\tau)}$ and time-signals at all the time except the time τ, since they obtain $c^{(\tau)} = s + r^{(\tau)}$ from their shares. They get $\tau - 1$ time-signals $r^{(1)}, \ldots, r^{(\tau-1)}$. However, since each time-signal is chosen uniformly at random from $\mathbb{F}_q$, they can guess $r^{(\tau)}$ only with probability $1/q$. By the security of one-time pad, we have $H(S \mid U_1^{(\tau)}, \ldots, U_n^{(\tau)}, TI^{(1)}, \ldots, TI^{(\tau-1)}) = H(S)$. Hence, for any $\mathcal{A} \in \mathcal{PS}(\mathcal{P}, k, n)$ and for any $t \in \mathcal{T}$, we have $H(S \mid U_{\mathcal{A}}^{(t)}, TI^{(1)}, \ldots, TI^{(t-1)}, TI^{(t+1)}, \ldots, TI^{(\tau)}) = H(S)$.

Finally, it is straightforward to see that the construction satisfies all the equalities of lower bounds in Theorem 4.9. Therefore, the above construction is optimal. □

4.5.4 Model and Security Definition of (k_1, k_2, n)-TR-SS

In this section, we consider the following problem,

> "Can we realize traditional secret sharing functionality and timed-release secret sharing functionality simultaneously?"

Therefore, we propose (k_1, k_2, n)-TR-SS, where k_1 and k_2 are threshold values with $1 \leq k_1 \leq k_2 \leq n$. (k_1, k_2, n)-TR-SS can realize timed-release functionality —a secret can be reconstructed from at least k_1 shares and a time-signal at the specified time— and traditional secret sharing functionality —a secret can be also reconstructed from only at least k_2 shares— simultaneously. In the case that $k = k_1 = k_2$, (k, k, n)-TR-SS can be considered as the traditional (k, n)-SS (for details, see Remark 4.6).

We propose a model and a security definition of (k_1, k_2, n)-TR-SS. First, we introduce a model of (k_1, k_2, n)-TR-SS. In (k_1, k_2, n)-TR-SS, there are same entities and sets as those of (k, n)-TR-SS. The main difference from (k, n)-TR-SS is that a dealer D can specify two kinds of threshold values, k_1 and k_2 with $k_1 \leq k_2 \leq n$: k_1 indicates the number of participants who can reconstruct a secret s with the time-signal at the time specified by the dealer; and k_2 indicates the number of participants who can reconstruct s without any time-signals. We give the definition of (k_1, k_2, n)-TR-SS as follows.

Definition 4.13 ((k_1, k_2, n)-TR-SS). *A (k_1, k_2, n)-TR-SS scheme Π_{hTSS} involves $n + 3$ entities, $TA, D, P_1, \ldots, P_n$, and TS, and consists of five phases, Initialize, Extract, Share, Reconstruct with time-signals and Reconstruct without time-signals, and five finite spaces, $\mathcal{S}, \mathcal{SK}, \mathcal{U}, \mathcal{T}$, and $\mathcal{TI}$. Π_{hTSS} is executed based on the following phases as follows.*

1. ***Initialize.*** *This phase follows the same procedure as that of (k, n)-TR-SS (see Definition 4.10).*

2. ***Share.*** *A dealer D randomly selects a secret $s \in \mathcal{S}$ according to P_S. Then, D chooses k_1, k_2 and n, and specifies future time $t \in \mathcal{T}$ when at least k_1 participants can reconstruct s. Then, on input the secret s, the specified time t and a secret key $sk \in \mathcal{SK}$, D computes a share $u_i^{(t)} \in \mathcal{U}_i^{(t)}$ for every P_i $(i = 1, 2, \ldots, n)$ and a public parameter $pp \in \mathcal{PP}$.[6] And then, D discloses pp and sends a pair of the share and specified time, $(u_i^{(t)}, t)$, to P_i $(i = 1, 2, \ldots, n)$ via a secure channel, respectively.*

3. ***Extract.*** *This phase follows the same procedure as that of (k, n)-TR-SS (see Definition 4.10).*

4. ***Reconstruct with time-signals.*** *At the specified time t, any set of participants $\mathcal{A} = \{P_{i_1}, \ldots, P_{i_j}\} \in \mathcal{PS}(\mathcal{P}, k_1, k_2 - 1)$ can reconstruct the secret s by using their shares $(u_{i_1}^{(t)}, \ldots, u_{i_j}^{(t)})$ $(k_1 \leq j < k_2)$ and a time-signal of the specified time $ts^{(t)}$.*

5. ***Reconstruct without time-signals.*** *At any time (even before the specified time), any set of participants $\hat{\mathcal{A}} = \{P_{i_1}, \ldots, P_{i_j}\} \in \mathcal{PS}(\mathcal{P}, k_2, n)$ can reconstruct the secret s by using only their shares $(u_{i_1}^{(t)}, \ldots, u_{i_j}^{(t)})$ $(k_2 \leq j \leq n)$.*

In the above model, we assume that $\Pi_{h\text{TSS}}$ meets the following *correctness* properties:

a) If D correctly completes the phase *Share* and TS correctly completes the phase *Extract*, then, for all possible $i \in [n]$, $t \in \mathcal{T}$, $s \in \mathcal{S}$, $u_i^{(t)} \in \mathcal{U}_i^{(t)}$, and $ts^{(t)} \in \mathcal{TI}^{(t)}$, it holds that any $\mathcal{A} \in \mathcal{PS}(\mathcal{P}, k_1, k_2 - 1)$ will correctly reconstruct the secret s at the end of phase *Reconstruct with time-signals*, namely, $H(S \mid U_{\mathcal{A}}^{(t)}, TI^{(t)}) = 0$.

b) If D correctly completes the phase *Share*, then, for all possible $i \in \{1, 2, \ldots, n\}$, $t \in \mathcal{T}$, $s \in \mathcal{S}$, and $u_i^{(t)} \in \mathcal{U}_i^{(t)}$, it holds that any $\hat{\mathcal{A}} \in \mathcal{PS}(\mathcal{P}, k_2, n)$ will correctly reconstruct the secret s at the end of phase *Reconstruct without time-signals*, namely, $H(S \mid U_{\hat{\mathcal{A}}}^{(t)}) = 0$.

Next, we formalize a security definition of (k_1, k_2, n)-TR-SS in a similar way to that of (k, n)-TR-SS as follows. Note that the description of (the random variable of) the public parameters is omitted below since existing works using public parameters such as [80] do not explicitly describe the public parameter in the security definition.

[6] Although not used in the previous TR-SS scheme, we here introduce a public parameter *pp* since we will need it in our construction in Section 4.5.6.

Definition 4.14 (Security of (k_1, k_2, n)-TR-SS). *Let $\Pi_{h\mathrm{TSS}}$ be a (k_1, k_2, n)-TR-SS scheme. $\Pi_{h\mathrm{TSS}}$ is said to be secure if the following conditions are satisfied:*

(i) For any $\mathcal{F} \in \mathcal{PS}(\mathcal{P}, 1, k_1 - 1)$ and any $t \in \mathcal{T}$, it holds that

$$H(S \mid U_{\mathcal{F}}^{(t)}, TI^{(1)}, \ldots, TI^{(\tau)}) = H(S).$$

(ii) For any $\hat{\mathcal{F}} \in \mathcal{PS}(\mathcal{P}, k_1, k_2 - 1)$ and any $t \in \mathcal{T}$, it holds that

$$H(S \mid U_{\hat{\mathcal{F}}}^{(t)}, TI^{(1)}, \ldots, TI^{(t-1)}, TI^{(t+1)}, \ldots, TI^{(\tau)}) = H(S).$$

In Definition 4.14, intuitively, the meaning of (i) is the same as that of (k, n)-TR-SS (Definition 4.11), and the meaning of the condition (ii) implies that no information on a secret is obtained by any set of at least k_1 but *no more than* k_2 participants, even if they obtain time-signals at all the time except the specified time. We can also consider a more strong security notion as discussed in (k, n)-TR-SS, however, we do not consider such a strong notion by the same reason as in the case of (k, n)-TR-SS.

Remark 4.6. *In the case of $k = k_1 = k_2$, the model and security definition of secure (k, k, n)-TR-SS (Definitions 4.10 and 4.11) are the same as those of traditional (k, n)-SS. Therefore, the model and security definition of (k_1, k_2, n)-TR-SS can be regarded as the natural extension of those of traditional secret sharing schemes.*

4.5.5 Lower Bounds Required for (k_1, k_2, n)-TR-SS

In this section, we show lower bounds on sizes of shares, time-signals, and secret keys required for secure (k_1, k_2, n)-TR-SS as follows. Note that in the proof, there are several technical points which are more complicated than that of Theorem 4.9.

Theorem 4.11. *Let $\Pi_{h\mathrm{TSS}}$ be any secure (k_1, k_2, n)-TR-SS scheme. Then, for any $i \in [n]$ and for any $t \in \mathcal{T}$, we have*

$$(I)\ H(U_i^{(t)}) \geq H(S).$$

Moreover, if the above lower bound holds with equality (i.e. $H(U_i^{(t)}) = H(S)$ for any i and t), we have

$$(II)\ H(TI^{(t)}) \geq (k_2 - k_1)H(S), \quad (III)\ H(SK) \geq \tau(k_2 - k_1)H(S).$$

Proof. The proof of Theorem 4.11 follows from the following lemmas.

Lemma 4.25. *$H(U_i^{(t)}) \geq H(S)$ for any $i \in [n]$ and any $t \in \mathcal{T}$.*

Proof. The proof of this lemma can be proved in a way similar to the proof of Lemma 4.22. For arbitrary $i \in [n]$, we take a subset $\mathcal{B}_i \in \mathcal{PS}(\mathcal{P} \setminus \{P_i\}, k_2 - 1, k_2 - 1)$ of participants. Then, for any $t \in \mathcal{T}$, we have

$$\begin{aligned} H(U_i^{(t)}) &\geq H(U_i^{(t)} \mid U_{\mathcal{B}_i}^{(t)}, TI^{(1)}, \ldots, TI^{(t-1)}) && (4.54)\\ &\geq I(S; U_i^{(t)} \mid U_{\mathcal{B}_i}^{(t)}, TI^{(1)}, \ldots, TI^{(t-1)}) \\ &= H(S \mid U_{\mathcal{B}_i}^{(t)}, TI^{(1)}, \ldots, TI^{(t-1)}) && (4.55)\\ &= H(S), && (4.56) \end{aligned}$$

where Eq. (4.55) follows from the correctness of (k_1, k_2, n)-TR-SS and Eq. (4.56) follows from the condition (ii) in Definition 4.14. □

Lemma 4.26. *If $H(U_i^{(t)}) = H(S)$ for any $i \in [n]$ and $t \in \mathcal{T}$, $H(TI^{(t)}) \geq H(TI^{(t)} \mid TI^{(1)}, \ldots, TI^{(t-1)}) \geq (k_2 - k_1)H(S)$ for any $t \in \mathcal{T}$.*

Proof. The statement is true in the case that $k_1 = k_2$, since Shannon entropy is non-negative. Therefore, in the following, we assume $k_1 < k_2$. For arbitrary $i \in [n]$, we take a subset $\mathcal{B}_i \in \mathcal{PS}(\mathcal{P} \setminus \{P_i\}, k_2 - 1, k_2 - 1)$ of participants. For any $t \in \mathcal{T}$, we have

$$\begin{aligned} &H(TI^{(t)}) && (4.57)\\ &\geq H(TI^{(t)} \mid TI^{(1)}, \ldots, TI^{(t-1)}) \\ &\geq I(TI^{(t)}; U_1^{(t)}, U_2^{(t)}, \ldots, U_n^{(t)} \mid TI^{(1)}, \ldots, TI^{(t-1)}) \\ &= H(U_1^{(t)}, U_2^{(t)}, \ldots, U_n^{(t)} \mid TI^{(1)}, \ldots, TI^{(t-1)}) \\ &\qquad - H(U_1^{(t)}, U_2^{(t)}, \ldots, U_n^{(t)} \mid TI^{(1)}, \ldots, TI^{(t)}) \\ &= H(U_1^{(t)}, \ldots, U_{k_1}^{(t)} \mid TI^{(1)}, \ldots, TI^{(t-1)}) \\ &\qquad + H(U_{k_1+1}^{(t)}, \ldots, U_{k_2}^{(t)} \mid TI^{(1)}, \ldots, TI^{(t-1)}, U_1^{(t)}, \ldots, U_{k_1}^{(t)}) \\ &\qquad\quad + H(U_{k_2+1}^{(t)}, \ldots, U_n^{(t)} \mid TI^{(1)}, \ldots, TI^{(t-1)}, U_1^{(t)}, \ldots, U_{k_2}^{(t)}) \\ &\qquad - H(U_1^{(t)}, \ldots, U_{k_1}^{(t)} \mid TI^{(1)}, \ldots, TI^{(t)}) \\ &\qquad\quad - H(U_{k_1+1}^{(t)}, \ldots, U_{k_2}^{(t)} \mid TI^{(1)}, \ldots, TI^{(t)}, U_1^{(t)}, \ldots, U_{k_1}^{(t)}) \\ &\qquad\quad\quad - H(U_{k_2+1}^{(t)}, \ldots, U_n^{(t)} \mid TI^{(1)}, \ldots, TI^{(t)}, U_1^{(t)}, \ldots, U_{k_2}^{(t)}) \\ &\geq H(U_1^{(t)}, \ldots, U_{k_1}^{(t)} \mid TI^{(1)}, \ldots, TI^{(t)}) \\ &\qquad + H(U_{k_1+1}^{(t)}, \ldots, U_{k_2}^{(t)} \mid TI^{(1)}, \ldots, TI^{(t-1)}, U_1^{(t)}, \ldots, U_{k_1}^{(t)}) \\ &\qquad\quad + H(U_{k_2+1}^{(t)}, \ldots, U_n^{(t)} \mid TI^{(1)}, \ldots, TI^{(t)}, U_1^{(t)}, \ldots, U_{k_2}^{(t)}) \\ &\qquad - H(U_1^{(t)}, \ldots, U_{k_1}^{(t)} \mid TI^{(1)}, \ldots, TI^{(t)}) \\ &\qquad\quad - H(U_{k_1+1}^{(t)}, \ldots, U_{k_2}^{(t)} \mid TI^{(1)}, \ldots, TI^{(t)}, U_1^{(t)}, \ldots, U_{k_1}^{(t)}) \end{aligned}$$

$$
\begin{aligned}
&\quad - H(U_{k_2+1}^{(t)},\ldots,U_n^{(t)} \mid TI^{(1)},\ldots,TI^{(t)},U_1^{(t)},\ldots,U_{k_2}^{(t)}) \\
&= H(U_{k_1+1}^{(t)},\ldots,U_{k_2}^{(t)} \mid TI^{(1)},\ldots,TI^{(t-1)},U_1^{(t)},\ldots,U_{k_1}^{(t)}) \\
&\quad - H(U_{k_1+1}^{(t)},\ldots,U_{k_2}^{(t)} \mid TI^{(1)},\ldots,TI^{(t)},U_1^{(t)},\ldots,U_{k_1}^{(t)}) \\
&\geq \sum_{i=k_1+1}^{k_2} H(U_i^{(t)} \mid TI^{(1)},\ldots,TI^{(t-1)},U_{\mathcal{B}_i}^{(t)}) \\
&\quad - \sum_{i=k_1+1}^{k_2} H(U_i^{(t)} \mid TI^{(1)},\ldots,TI^{(t)},U_1^{(t)},\ldots,U_{i-1}^{(t)}) \\
&= (k_2-k_1)H(S), \qquad (4.58)
\end{aligned}
$$

where Eq. (4.58) follows from Eq. (4.54) in the proof of Lemma 4.25, the assumption of $H(U_i^{(t)}) = H(S)$, and the following claim. □

Claim 4.5. *If $k_1 < k_2$ and $H(U_i^{(t)}) = H(S)$ for any $i \in [n]$ and $t \in \mathcal{T}$, $H(U_i^{(t)} \mid U_{\mathcal{A}_i}, TI^{(t)}) = 0$ for any $i \in [n]$, any $\mathcal{A}_i \in \mathcal{PS}(\mathcal{P} \setminus \{P_i\}, k_1, k_2 - 1)$, and any $t \in \mathcal{T}$.*

Proof. First, for arbitrary $i \in \{1, 2, \ldots, n\}$, we take subsets $\mathcal{B}_i := \mathcal{PS}(\mathcal{P} \setminus \{P_i\}, k_1 - 1, k_1 - 1)$ and $\mathcal{A}_i := \mathcal{PS}(\mathcal{P} \setminus \{P_i\}, k_1, k_2 - 1)$ of participants such that $\mathcal{B}_i \subset \mathcal{A}_i$. Then, for any $t \in \mathcal{T}$, we have

$$
\begin{aligned}
H(U_i^{(t)}) &\geq H(U_i^{(t)} \mid U_{\mathcal{B}_i}^{(t)}, TI^{(t)}) \\
&\geq H(U_i^{(t)} \mid U_{\mathcal{B}_i}^{(t)}, TI^{(t)}) - H(U_i^{(t)} \mid U_{\mathcal{B}_i}^{(t)}, TI^{(t)}, S) \qquad (4.59) \\
&= I(U_i^{(t)}; S \mid U_{\mathcal{B}_i}^{(t)}, TI^{(t)}) \\
&= H(S \mid U_{\mathcal{B}_i}^{(t)}, TI^{(t)}) - H(S \mid U_{\mathcal{B}_i}^{(t)}, U_i^{(t)}, TI^{(t)}) \\
&= H(S \mid U_{\mathcal{B}_i}^{(t)}, TI^{(t)}) \qquad (4.60) \\
&= H(S), \qquad (4.61)
\end{aligned}
$$

where Eq. (4.60) follows form the correctness of (k_1, k_2, n)-TR-SS and Eq. (4.61) follows from the condition (i) in Definition 4.14.

From Eq. (4.59) and the assumption of $H(U_i^{(t)}) = H(S)$, we have

$$
\begin{aligned}
&H(U_i^{(t)} \mid U_{\mathcal{B}_i}^{(t)}, TI^{(t)}) \\
&\quad = H(U_i^{(t)} \mid U_{\mathcal{B}_i}^{(t)}, TI^{(t)}) - H(U_i^{(t)} \mid U_{\mathcal{B}_i}^{(t)}, TI^{(t)}, S).
\end{aligned}
$$

Therefore, we have

$$
H(U_i^{(t)} \mid U_{\mathcal{B}_i}^{(t)}, TI^{(t)}, S) = 0.
$$

Hence, we have

$$\begin{aligned} H(U_i^{(t)} \mid U_{\mathcal{A}_i}^{(t)}, TI^{(t)}) &= H(U_i^{(t)} \mid U_{\mathcal{A}_i}^{(t)}, TI^{(t)}, S) \\ &\leq H(U_i^{(t)} \mid U_{\mathcal{B}_i}^{(t)}, TI^{(t)}, S) = 0. \end{aligned}$$

Since $H(U_i^{(t)} \mid U_{\mathcal{A}_i}^{(t)}, TI^{(t)}) \geq 0$, we have $H(U_i^{(t)} \mid U_{\mathcal{A}_i}^{(t)}, TI^{(t)}) = 0$. □

Lemma 4.27. *If $H(U_i^{(t)}) = H(S)$ for any $i \in [n]$ and $t \in \mathcal{T}$, $H(SK) \geq \tau(k_2 - k_1)H(S)$.*

Proof. We have

$$\begin{aligned} H(SK) &\geq I(TI^{(1)}, \ldots, TI^{(\tau)}; SK) \\ &= H(TI^{(1)}, \ldots, TI^{(\tau)}) - H(TI^{(1)}, \ldots, TI^{(\tau)} \mid SK) \\ &= H(TI^{(1)}, \ldots, TI^{(\tau)}) \\ &= \sum_{t=1}^{\tau} H(TI^{(t)} \mid TI^{(1)}, \ldots, TI^{(t-1)}) \\ &\geq \tau(k_2 - k_1)H(S), \end{aligned}$$

where the last inequality follows from Lemma 4.26. □

Proof of Theorem 4.11: From Lemmas 4.25–4.27, the proof of Theorem 4.11 is completed. □

As we will see in Section 4.5.6, the lower bounds in Theorem 4.11 are tight since our construction will meet all the above lower bounds with equalities. Therefore, we define optimality of constructions of (k_1, k_2, n)-TR-SS as follows.

Definition 4.15. *A construction of secure (k_1, k_2, n)-TR-SS is said to be optimal if it meets equality in every bound of (i)–(iii) in Theorem 4.11.*

4.5.6 Optimal (but Restricted) Construction of (k_1, k_2, n)-TR-SS

We can consider a naive construction based on (k_1, n)-TR-SS and (k_2, n)-SS, however, this naive construction is not optimal since the share size is twice as large as the underlying secret size. Before describing our construction, we show such a naive construction as follows.

1. *Initialize.* Let q be a prime power, where $q > \max\{n, \tau\}$, and let $\mathbb{F}_q$ be the finite field with q elements. We assume that the identity of each participant P_i is encoded as $P_i \in \mathbb{F}_q \backslash \{0\}$. Also, we assume $\mathcal{T} = [\tau] \subset \mathbb{F}_q \backslash \{0\}$ by using appropriate encoding. First, TA chooses uniformly at random τ numbers $r^{(j)} (1 \leq j \leq \tau)$ from $\mathbb{F}_q$. TA sends a secret key $sk := (r^{(1)}, \ldots, r^{(\tau)})$ to TS and D via secure channels, respectively.

2. *Share.* First, D randomly chooses a secret $s \in \mathbb{F}_q$ according to a distribution P_S over $\mathbb{F}_q$. Also, D specifies the time t when at least k_1 participants can reconstruct the secret and chooses t-th key $r^{(t)}$. Next, D randomly chooses two polynomials $f_1(x) := s + r^{(t)} + \sum_{i=1}^{k_1-1} a_{1i}x^i$ and $f_2(x) := s + \sum_{i=1}^{k_2-1} a_{2i}x^i$ over $\mathbb{F}_q$, where each coefficient is randomly and uniformly chosen from $\mathbb{F}_q$. Then, D computes $u_i^{(t)} := (f_1(P_i), f_2(P_i))$. Finally, D sends $(u_i^{(t)}, t)$ to $P_i (i = 1, 2, \ldots, n)$ via a secure channel.

3. *Extract.* For sk and time $t \in \mathcal{T}$, TS broadcasts t-th key $r^{(t)}$ as a time-signal at time t to all participants via a (authenticated) broadcast channel.

4. *Reconstruct with time-signals.* First, $\mathcal{A} = \{P_{i_1}, P_{i_2}, \ldots, P_{i_{k_1}}\} \in \mathcal{PS}(\mathcal{P}, k_1, k_1)$ computes $s + r^{(t)}$ by Lagrange interpolation:

$$s + r^{(t)} = \sum_{j=1}^{k_1} \left(\prod_{l \neq j} \frac{P_{i_j}}{P_{i_j} - P_{i_l}} \right) f_1(P_{i_j}),$$

   from $(f_1(P_{i_1}), \ldots, f_1(P_{i_{k_1}}))$. After receiving $ts^{(t)} = r^{(t)}$, they can compute and get $s = s + r^{(t)} - ts^{(t)}$.

5. *Reconstruct without time-signals.* any $\hat{\mathcal{A}} = \{P_{i_1}, P_{i_2}, \ldots, P_{i_{k_2}}\} \in \mathcal{PS}(\mathcal{P}, k_2, k_2)$ computes

$$s = \sum_{j=1}^{k_2} \left(\prod_{l \neq j} \frac{P_{i_j}}{P_{i_j} - P_{i_l}} \right) f_2(P_{i_j}),$$

   by Lagrange interpolation from $(f_2(P_{i_1}), \ldots, f_2(P_{i_{k_2}}))$.

It is easy to see that the above construction is secure, since this construction is a simple combination of (k_1, n)-TR-SS and Shamir's (k_2, n)-SS. Also, the above construction is simple, however not optimal since the resulting share size is twice as large as that of secrets.

To achieve an optimal construction, we use the technique in [80]: In the phase *Share*, the dealer computes public parameters, and the public parameters are broadcasted to participants or else stored on a publicly accessible authenticated bulletin board. Although we have to disclose $k_2 - k_1$ elements in a finite field as a public parameter, each share can consist of only one element. In [80], Jhanwar and Safavi-Naini used this technique for reducing share sizes, and consequently they succeeded in constructing optimal share sizes. We note that although similar techniques that the dealer broadcasts several coefficients of the polynomial such as [20, 21, 94, 95] are known, the aim of their techniques differs from our aim. Specifically, it is to realize the

functionality, whereas our aim is to reduce the share sizes, and consequently, to achieve an optimal construction. However, in this optimal construction a dealer is only allowed to choose k_1 and k_2 such that $k_2 - k_1 \leq \ell$, where ℓ is determined by TA in the phase *Initialize*. In this sense, this construction is restricted. The detail of our construction is given as follows.

1. *Initialize.* Let q be a prime power, where $q > \max\{n, \tau\}$, and let $\mathbb{F}_q$ be the finite field with q elements. We assume that the identity of each participant P_i is encoded as $P_i \in \mathbb{F}_q \backslash \{0\}$. Also, we assume $\mathcal{T} = [\tau] \subset \mathbb{F}_q \backslash \{0\}$ by using appropriate encoding. First, TA chooses ℓ, which is the maximum difference between k_2 and k_1. Note that k_1 and k_2 will be determined by a dealer D in the phase *Share*. Then, TA chooses $\tau\ell$ numbers $r_i^{(t)}$ $(1 \leq i \leq \ell$, and $1 \leq t \leq \tau)$ from $\mathbb{F}_q$ uniformly at random. TA sends a secret key $sk := \{(r_1^{(t)}, r_2^{(t)}, \ldots, r_\ell^{(t)})\}_{1 \leq t \leq \tau}$ to TS and D via secure channels, respectively.

2. *Share.* First, D randomly selects a secret $s \in \mathbb{F}_q$ according to a distribution P_S over $\mathbb{F}_q$, and chooses k_1, k_2 and n such that $k_2 - k_1 \leq \ell$. Also, D specifies the time t when at least k_1 participants can reconstruct the secret. Next, D randomly chooses a polynomial $f(x) := s + \sum_{i=1}^{k_2-1} a_i x^i$ over $\mathbb{F}_q$, where each coefficient a_i is randomly and uniformly chosen from $\mathbb{F}_q$. Then, D computes a share $u_i^{(t)} := f(P_i)$ and a public parameter $p_i^{(t)} := a_{k_1-1+i} + r_i^{(t)}$ $(i = 1, 2, \ldots, k_2 - k_1)$. Finally, D sends $(u_i^{(t)}, t)$ to $P_i (i = 1, 2, \ldots, n)$ via a secure channel and discloses $pp := (p_1^{(t)}, \ldots, p_{k_2-k_1}^{(t)})$.

3. *Extract.* For sk and time $t \in \mathcal{T}$, TS broadcasts a time-signal at time t, $ts^{(t)} := (r_1^{(t)}, r_2^{(t)}, \ldots, r_\ell^{(t)})$ to all participants via a (authenticated) broadcast channel.

4. *Reconstruct with time-signals.* Suppose that all participants receive $ts^{(t)} = (r_1^{(t)}, r_2^{(t)}, \ldots, r_\ell^{(t)})$. Let $\mathcal{A} = \{P_{i_1}, P_{i_2}, \ldots, P_{i_{k_1}}\} \in \mathcal{PS}(\mathcal{P}, k_1, k_1)$ be a set of any k_1 participants. First, each $P_{i_j} \in \mathcal{A}$ computes $a_{k_1-1+i} = p_i^{(t)} - r_i^{(t)}$ $(i = 1, 2, \ldots, k_2 - k_1)$ and constructs $g(x) := \sum_{k_1}^{k_2-1} a_i x^i$. Then, each P_{i_j} computes $h(P_{i_j}) := f(P_{i_j}) - g(P_{i_j})$ $(j = 1, \ldots, k_1)$ such that $h(x) := s + \sum_{i=1}^{k_1-1} a_i x^i$. Then, they compute

$$s = \sum_{j=1}^{k_1} \left(\prod_{l \neq j} \frac{P_{i_j}}{P_{i_j} - P_{i_l}} \right) h(P_{i_j}),$$

by Lagrange interpolation from $(h(P_{i_1}), \ldots, h(P_{i_{k_1}}))$.

5. *Reconstruct without time-signals.* any $\hat{\mathcal{A}} = \{P_{i_1}, P_{i_2}, \ldots, P_{i_{k_2}}\} \in \mathcal{PS}(\mathcal{P}, k_2, k_2)$ computes

$$s = \sum_{j=1}^{k_2} \left(\prod_{l \neq j} \frac{P_{i_j}}{P_{i_j} - P_{i_l}} \right) f(P_{i_j}),$$

by Lagrange interpolation from their k_2 shares.

The security and optimality of the above construction is stated as follows.

Theorem 4.12. *The resulting (k_1, k_2, n)-TR-SS Π_{hTSS} by the above construction is secure. Moreover, it is optimal if $k_2 - k_1 = \ell$.*

Proof. First, we show the proof of (i) in Definition 4.14. Assume that $k_1 - 1$ participants $\mathcal{F} = \{P_{i_1}, \ldots, P_{i_{k_1-1}}\} \in \mathcal{PS}(\mathcal{P}, k_1 - 1, k_1 - 1)$ try to guess s by using their shares, public parameters, and all time-signals. $\mathcal{F}$ can compute $g(x)$ from public parameters and the time-signal at the specified time, hence they can get $h(P_{i_l}) = f(P_{i_l}) - g(P_{i_l})$ $(l = 1, \ldots, k_1 - 1)$. Thus, they can know the following matrix:

$$\begin{pmatrix} 1 & P_{i_1} & \cdots & P_{i_1}^{k_1-1} \\ 1 & P_{i_2} & \cdots & P_{i_2}^{k_1-1} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & P_{i_{k_1-1}} & \cdots & P_{i_{k_1-1}}^{k_1-1} \end{pmatrix} \begin{pmatrix} s \\ a_1 \\ \vdots \\ a_{k_1-1} \end{pmatrix}. \tag{4.62}$$

However, from Eq. (4.62), they cannot guess at least one element of $(a_1, \ldots, a_{k_1-1})$ with probability larger than $1/q$. Therefore, from the property of the one-time pad, we have $H(S \mid U_{\mathcal{F}}^{(t)}, TI^{(1)}, \ldots, TI^{(\tau)}) = H(S)$ for any $\mathcal{F} \in \mathcal{PS}(\mathcal{P}, 1, k_1 - 1)$ and any $t \in \mathcal{T}$.

Next, we show the proof of (ii) in Definition 4.14. Without loss of generality, we suppose that τ is a specified time, and $k_2 - k_1 = \ell$. Suppose that $k_2 - 1$ participants try to guess s by using their shares, public parameters, and time-signals at all the time except the time τ. First, they cannot guess at least one coefficient of $f(x)$ with probability larger than $1/q$ since the degree of $f(x)$ is at most $k_2 - 1$ as in Shamir's secret sharing scheme [127]. Therefore, they attempt to guess one of $a_{k_1}, \ldots, a_{k_2-1}$ by using their $k_2 - 1$ shares, public parameters and $\tau - 1$ time-signals, since if they obtain any one of these coefficient, they can get $f^*(P_{i_l})$ $(l = 1, \ldots, k_2 - 1)$ such that the degree of $f^*(x)$ is $k_2 - 2$ and reconstruct s by Lagrange interpolation. They know $\tau - 1$ time-signals, however, these time-signals $\{(r_1^{(j)}, \ldots, r_\ell^{(j)})\}_{1 \leq j \leq \tau - 1}$ are independent of the time-signal $(r_1^{(\tau)}, \ldots, r_\ell^{(\tau)})$ at τ. Hence, by the security of one-time pad, they cannot guess each a_{k_1-1+i} $(= p_i^{(\tau)} - r_i^{(\tau)})$ $(1 \leq i \leq k_2 - k_1)$ with probability larger than $1/q$ since each $r_i^{(\tau)}$ is chosen from $\mathbb{F}_q$ uniformly

at random. Therefore, we have $H(S \mid U_{l_1}^{(\tau)}, \ldots, U_{l_{k_2-1}}^{(\tau)}, TI^{(1)}, \ldots, TI^{(\tau-1)}) = H(S)$. Hence, for any $\mathcal{A} \in \mathcal{PS}(\mathcal{P}, k_1, k_2 - 1)$ and any $t \in \mathcal{T}$, we have $H(S \mid U_{\mathcal{A}}^{(t)}, TI^{(1)}, \ldots, TI^{(t-1)}, TI^{(t+1)}, \ldots, TI^{(\tau)}) = H(S)$.

Finally, if $k_2 - k_1 = \ell$, it is straightforward to see that the construction satisfies all the equalities of lower bounds in Theorem 4.11. Therefore, the above construction is optimal if $k_2 - k_1 = \ell$. □

4.5.7 Extensions of TR-SS

In this section, we discuss the following extensions of our results in the previous sections.

Timed-release secret sharing with general access structures. In [78, 79], a generic construction of secret sharing schemes for any general access structure by using threshold secret sharing schemes was proposed, and later such a technique was improved in terms of efficiency on share sizes in [11, 137]. We can realize a timed-release secret sharing scheme for any general access structures from (k, n)-TR-SS schemes based on the techniques [11, 137].

Robust timed-release secret sharing schemes. Robust secret sharing schemes [118, 42, 34, 80] are secret sharing schemes secure against malicious modification of shares. Technically, suppose that at most ω ($< n/2$) participants are allowed to modify their own shares so that a reconstructor recovers a secret s', which is different from the original secret s, from all n shares. Then, the secret sharing scheme is said to be (ω, δ)-robust if the success probability of the attack is at most δ. We can construct a (ω, δ)-robust TR-SS scheme by using $(\omega + 1, n)$-TR-SS schemes via two existing approaches: (1) the Rabin–Ben-Or (RB) approach [118, 34]; and (2) the Cramer–Damgård–Fehr (CDF) approach [42, 80]. Since each scheme can be easily constructed and security of each scheme can be proved by a slight modification to the original proof, we here briefly explain the two approaches below.

(1) The RB approach: We can construct a (ω, δ)-robust TR-SS scheme from a $(\omega + 1, n)$-TR-SS scheme and an information-theoretically secure authentication code (A-code for short) [132]. First, a dealer specifies time t, and generates n shares $u_1^{(t)}, \ldots, u_n^{(t)}$ of a secret s by using the $(\omega+1, n)$-TR-SS scheme. Next, he generates n^2 keys of the A-code, $k_i^{(j)}$ $(1 \leq i, j \leq n)$, and generates tags $tag_i^{(j)}$ by using $u_j^{(t)}$ and $k_i^{(j)}$. Then, P_i's share is $(u_i^{(t)}, k_i^{(1)}, \ldots, k_i^{(n)}, tag_1^{(i)}, \ldots, tag_n^{(i)})$. In the reconstruction phase, a reconstructor checks the validity of $u_i^{(t)}$ by using $k_j^{(i)}$ and $tag_j^{(i)}$. If the validity of $u_i^{(t)}$ is guaranteed by at least $\omega + 1$ pairs of $k_j^{(i)}$ and $tag_j^{(i)}$, then the share is considered as the *valid* share. After receiving a time-signal at t, then the secret s can be recovered from at least $\omega + 1$ valid shares and the time-signal.

(2) The CDF approach: We can also construct a (ω, δ)-robust TR-SS scheme

from a $(\omega+1,n)$-TR-SS scheme and a traditional $(\omega+1,n)$-SS scheme. For simplicity, let $\mathcal{S} := \mathbb{F}_q$. First, as in the RB approach, a dealer specifies time t, and generates n shares $u_1^{(t)},\ldots,u_n^{(t)}$ of a secret s by using the $(\omega+1,n)$-TR-SS scheme. Then, he chooses $r \in \mathbb{F}_q$ uniformly at random, and computes $tag := s \cdot r$. He generates n shares of r and tag by using the $(\omega+1,n)$-SS scheme, respectively. Let $\tilde{u}_1^{(t)},\ldots,\tilde{u}_n^{(t)}$ be shares of r, and $\hat{u}_1^{(t)},\ldots,\hat{u}_n^{(t)}$ be shares of tag, respectively. Then, P_i's share is $(u_i^{(t)},\tilde{u}_i^{(t)},\hat{u}_i^{(t)})$. In the reconstruction phase, a reconstructor chooses a subset of $\omega+1$ participants, and reconstructs s', r', and tag' from their shares and a time-signal at t. Then, he checks whether it holds $s' \cdot r' = tag'$ or not. If so, he accepts s' as the original secret. Otherwise, he chooses different $\omega+1$ participants and performs the above operation again.

Chapter 5

Information-Theoretic Timed-Revocable Cryptography

5.1 Contribution in This Chapter

There are potentially many users in the cloud, and their access privileges for stored data is subject to change. BE schemes can handle such access privileges by designating privileged users and encrypting a plaintext so that only the privileged users can decrypt the encrypted plaintext. Thus, BE schemes seems well-suited to the cloud environment. In this chapter, we aim to add timed-revocable functionality to BE schemes in the cloud environment setting. Namely, we realize a BE scheme which allows to remove arbitrary users from a set of privileged users (i.e., revoke the users' decryption ability) without decrypting the corresponding ciphertext. (To be precise, the BE schemes also allows to arbitrarily add users to the set of privileged users without decryption.) We name such a scheme *a revocable-storage broadcast encryption (RS-BE) scheme*, and propose the notion of information-theoretically secure RS-BE. In a RS-BE scheme, similarly to traditional BE, the sender chooses a set of (initial) privileged users and encrypts a plaintext so that only these users can decrypt the ciphertext. Moreover, the *storage manager* can update the ciphertext to reflect changes in the set of privileged users. Here, the update procedure is carried out without revealing the plaintext, and thus, the storage manager cannot learn anything about the encrypted plaintext. We show tight lower bounds on the sizes of ciphertexts and secret keys in the smallest ciphertext setting. Note that this is an important and desired property since ciphertexts are stored in the cloud permanently or for a long time, and therefore, compactness of ciphertexts is one of the most important aspects to consider in the design of a RS-BE scheme. We then present an optimal construction which achieves these bounds. We furthermore propose a

collusion-resistant construction which is secure against collusion of a storage manager and illegitimate users and a robust construction which is resilient to a maliciously behaving storage manager.

As mentioned above, the proposed RS-BE scheme only captures the case of the smallest ciphertext size. Generally, BE schemes have trade-offs between the secret key sizes and ciphertext sizes, and RS-BE schemes must also have such trade-offs. Considering potential applications of RS-BE schemes, RS-BE schemes should be considered in the case of more general cipertext sizes. As a step toward RS-BE schemes with more general ciphertext sizes, we propose a generic construction of a traditional BE scheme with general ciphertext sizes.

Details of the first contribution (i.e., a proposal of RS-BE) in Sections 5.3 and 5.4 are as follows. Firstly, in Sections 5.3, we give a formal model and security definitions of information-theoretically secure RS-BE. Then, we clarify that it is possible to construct an information-theoretically secure RS-BE scheme in which the ciphertext length is the same as the plaintext length. We then investigate lower bounds on the sizes of decryption keys, encryption keys, and the storage manager's keys under the condition that the ciphertext size is the same as the plaintext size. These bounds can also be seen as a generalization of the bounds for (traditional) BE, and furthermore imply a tight bound on the size of encryption keys in BE which, to the best of our knowledge, has not been clarified before our work. For instance, Kurosawa et al. [87] showed tight lower bounds on the size of decryption keys for BE schemes through equivalence between BE schemes and KPS, however, they did not mention lower bounds on encryption keys in their paper. In contrast, we derive tight lower bounds on both of the sizes of encryption keys and decryption keys for BE schemes without using such equivalence, and it turns out that the tight lower bound on the size of decryption keys in [87] is a special case of our results (Theorem 5.2). We show an information-theoretically secure RS-BE scheme which meets all of these bounds with equalities. This means that these bounds are *tight* and the proposed construction is *optimal*. In Section 5.4, we consider two extensions of RS-BE. First, we consider a stronger security notion of RS-BE. Specifically, we formalize the security notion against collusion of a storage manager and receivers who are not included in a set of privileged users, and show a construction which meets the stronger security. Next, we furthermore consider a scenario in which a maliciously behaving storage manager can try to modify the encrypted plaintext. This is related to *non-malleability* in the context of ordinary encryption. In a RS-BE scheme, malleability may cause a serious problem since the ciphertext is periodically updated, but an improper update carried out by a malicious storage manager may not be immediately detectable by the users. Then, we present a concrete robust construction, which is provably secure against this type of attacks, from an ordinary RS-BE scheme and *algebraic manipulation detection codes* (AMD-code) [43].

Details of the second contribution (i.e., a proposal of BE schemes with

general ciphertext sizes) in Section 5.5 are as follows. There are two types of BE schemes: A $(t, \leq \omega)$-one-time secure BE scheme and an $(\leq n, \leq \omega)$-one-time secure BE scheme, where n is the number of users and ω is the maximum number of colluders (t will be explained in the next sentence). In the former BE scheme, a sender encrypts a plaintext for some privileged set $\mathcal{S}$ such that the cardinality of the subset $\mathcal{S}$ is exactly t (i.e. $|\mathcal{S}| = t$). On the other hand, in the latter BE scheme a sender can encrypt a plaintext for *any* privileged set. Namely, the latter scheme provides more flexible functionality than the former scheme. However, only two constructions of an $(\leq n, \leq \omega)$-one-time secure BE scheme are known so far: One is the Fiat–Naor construction [61] when the ciphertext size is equal to the plaintext size; and another is a trivial construction from the one-time pad when the maximum ciphertext size is n times larger than the plaintext size. These two constructions can be constructed KPSs (the one-time pad can be regarded as the special case of KPSs). Therefore, we propose a generic construction of an $(\leq n, \leq \omega)$-one-time secure BE scheme from KPSs with more general ciphertext sizes. Our generic construction includes the two above constructions, namely, it can be regarded as a natural extension of them. Specifically, we show that, if the maximum ciphertext size is δ times larger than the plaintext size, an $(\leq n, \leq \omega)$-one-time secure BE scheme can be constructed from δ KPSs. Our construction is simple, however, for fixed n, ω, and δ, there are many possible combinations of the KPSs to realize the $(\leq n, \leq \omega)$-one-time secure BE scheme in our construction methodology. Thus, we analyze parameters of all possible combinations in our construction methodology, and we show which combination is the best one in the sense that secret-key size can be minimized. Specifically, the analysis, which is our main contribution, is as follows. Each parameter consists of δ sub-parameters which each sub-parameter is input to each KPS. Again, we show which parameter gives such an optimal construction. To do so, we capture the analysis as a certain type of optimization problems, and we show the optimal parameter by solving the problem by using finite combinatorics. Consequently, we show that the parameter that consists of *even* sub-parameters makes the construction optimal. At first glance, the optimal parameter might be intuitively-plausible. However, it is important to show the optimal parameter *for any* n, ω, *and* δ in a mathematical way.

As explained above, deriving a tight bound on the secret-key size required for $(\leq n, \leq \omega)$-one-time secure BE schemes for any ciphertext size is an open problem. Solving the open problem is important from not only the academic aspect but also the practical aspect since it means that we can construct the most efficient BE scheme in terms of its secret-key sizes (i.e. a construction that attains every bound with equality) for any δ, and therefore we can adjust the secret-key sizes by arbitrarily choosing δ based on restrictions on channels such as channel capacity and channel bandwidth. In this sense, although our construction may not be most efficient, our result is meaningful since such *flexible* parameter setup is first realized by our construction (and parameter

optimization).

Generally, tight bounds on the secret-key size required for information-theoretically secure protocols are shown in the following manner: (i) we derive both lower bounds (by often using non-constructive proofs) and upper bounds (by often using constructive proofs); and (ii) we show the optimality by matching both of them. Hence, it is important to show not only lower bounds but also upper bounds, and this result (i.e. deriving and reducing upper bounds) is the first step to solve the open problem.

Related work on secure cloud storage and timed-revocable cryptography. Recently, many researchers have investigated how we can *securely* use cloud data storage for various purposes [4, 72, 81, 90, 91, 125, 126, 134, 148]. Sahai, Seyalioglu, and Waters [125] first dealt with the concept of a revocable storage, and proposed revocable-storage attribute-based encryption (RS-ABE). They assume ciphertexts are stored in an external storage, such as cloud storage, and considered revocable attribute-based encryption [5, 24] with ciphertext updatable functionality (to be precise, [24] in the context of identity-based encryption). However, RS-ABE is only computationally secure, and hence cannot guarantee long-term security.

Related work on broadcast encryption. Berkovits [14] first considered the concept of broadcast encryption, and Fiat and Naor [61] developed a formal and systematic approach to the construction of broadcast encryption schemes. Since then, broadcast encryption schemes have been improved both in the computationally secure setting [28, 53, 68, 105, 114] and in the information-theoretically secure setting [14, 19, 22, 37, 48, 61, 66, 87, 92, 110, 111, 135], and used in various situations such as copyright protection in the real world.

As explained earlier, roughly speaking in the information-theoretic security setting, there are two types of BE schemes, a $(t, \leq \omega)$-one-time secure BE scheme [19, 22, 87, 92, 110] and an $(\leq n, \leq \omega)$-one-time secure BE scheme [19, 61]. In the former BE scheme, a sender can encrypt a plaintext for a privileged set $\mathcal{S}$ such that $|\mathcal{S}| = t$, whereas in the latter BE scheme a sender can encrypt a plaintext for any privileged set $\mathcal{S}$. The latter can be applied to more flexible applications than the former, however, Blundo and Cresti [19] showed the secret-key size of the latter BE scheme is significantly larger than that of the former BE scheme by deriving tight lower bounds on the sizes of secret keys of both BE schemes in the context of *zero-message BE schemes*, which are the same as *key predistribution systems* (KPSs) [18, 97]. In other words, their lower bound holds only when the ciphertext size is equal to the plaintext size, and therefore, deriving a tight lower bound on the secret-key size for any ciphertext size still remains an open problem.[1] Furthermore, in

[1] Information-theoretically secure protocols usually require long secret keys, and therefore it is important to show the minimal key size (i.e. derive a tight lower bound on the secret-key

the same setting (i.e. the case that the ciphertext size is equal to the plaintext size), Kurosawa et al. [87] showed tight lower bounds on the size of decryption keys for BE schemes through equivalence between BE schemes and KPSs.

Blundo et al. [22] showed that there is a trade-off between the secret-key size and the ciphertext size in $(t, \leq \omega)$-one-time secure BE schemes. Namely, they derived lower bounds on the secret-key size of $(t, \leq \omega)$-one-time secure BE schemes when the ciphertext size is equal to an integer multiple of the plaintext size. Later, a trade-off between sizes of ciphertexts and secret keys in $(t, \leq \omega)$-one-time secure BE schemes was improved by Padró et al. [110]. There is no doubt that such a trade-off in $(\leq n, \leq \omega)$-one-time secure BE schemes exists, however there is no concrete analysis of the trade-off.

5.2 One-time Secure Broadcast Encryption Scheme

As explained in the introduction, we deal with a BE scheme where *any* subset of $\mathcal{U}$, where $\mathcal{U} := \{U_1, \ldots, U_n\}$ is a user set, can be chosen as a privileged set, and we call such a BE scheme simply "BE scheme". In BE schemes, a sender E generates an encryption key ek and n decryption keys $dk_1, \ldots, dk_n$, and sends dk_i to a user U_i via a secure channel $(1 \leq i \leq n)$, respectively. Then, E chooses a subset $\mathcal{S} \in \mathcal{U}$ (called *a privileged set*) and encrypts a plaintext m with his encryption key ek. After broadcasting a ciphertext $c_{\mathcal{S}}$, a user $U_i \in \mathcal{S}$ can decrypt it with his decryption key dk_i, while $U_j \notin \mathcal{S}$ cannot decrypt it.

Formally, the definition of a BE scheme is as follows. Let $\mathcal{M}$ be a set of possible plaintexts. For any subset $\mathcal{J} := \{U_{i_1}, \ldots, U_{i_j}\} \subset \mathcal{U}$, let $\mathcal{C}_{\mathcal{J}}$ be a set of all possible ciphertexts for the privileged set $\mathcal{J}$, and let $\mathcal{C} := \bigcup_{\mathcal{J} \subset \mathcal{U}} \mathcal{C}_{\mathcal{J}}$. Let $\mathcal{EK}$ be a set of possible encryption keys, and let $\mathcal{DK}_i$ be a set of possible decryption keys for U_i. Let $\mathcal{DK} := \bigcup_{i=1}^{n} \mathcal{DK}_i$.

Definition 5.1 (BE). *A BE scheme Π_{BE} involves $n+1$ entities, a sender E and n users $\mathcal{U}$, and consists of the following three-tuple of algorithms (Setup, Enc, Dec) with four spaces, $\mathcal{M}$, $\mathcal{C}$, $\mathcal{EK}$, and $\mathcal{DK}$, where all of the above algorithms except Setup are deterministic and all of the above spaces are finite.*

1. $(ek, dk_1, \ldots, dk_n) \leftarrow \mathsf{Setup}(n)$*: It takes the number of users n as input, and outputs an encryption key $ek \in \mathcal{EK}$, and n decryption keys $(dk_1, \ldots, dk_n) \in \prod_{i=1}^{n} \mathcal{DK}_i$.*

2. $c_{\mathcal{S}} \leftarrow \mathsf{Enc}(ek, m, \mathcal{S})$*: It takes an encryption key ek, a plaintext $m \in \mathcal{M}$, and a privileged set $\mathcal{S} \subset \mathcal{U}$ as input, and outputs a ciphertext $c_{\mathcal{S}}$.*

3. *m or $\perp$* $\leftarrow \mathsf{Dec}(dk_i, c_{\mathcal{S}}, \mathcal{S}, U_i)$*: It takes a decryption key dk_i of a user U_i, the ciphertext $c_{\mathcal{S}}$, the privileged set $\mathcal{S}$, and the identity U_i as input, and outputs m if $U_i \in \mathcal{S}$ or $\perp$ if $U_i \notin \mathcal{S}$.*

size).

In the above model, there is the following correctness requirement: For all $n \in \mathbb{N}$, all $(ek, dk_1, \ldots, dk_n) \leftarrow \mathsf{Setup}(n)$, all $m \in \mathcal{M}$, all $\mathcal{S} \subset \mathcal{U}$, and all $U_i \in \mathcal{S}$, $m \leftarrow \mathsf{Dec}(dk_i, \mathsf{Enc}(ek, m, \mathcal{S}), \mathcal{S}, U_i)$, or equivalently it holds $H(M \mid DK_i, C_{\mathcal{S}}) = 0$ for any $U_i \in \mathcal{S}$.

We consider perfect secrecy against at most ω colluders. Namely, at most ω colluders who are not included in the privileged set cannot get any information on the underlying plaintext from the ciphertext. For any $\mathcal{J} := \{U_{i_1}, \ldots, U_{i_j}\} \subset \mathcal{U}$, let $\mathcal{DK}_{\mathcal{J}} := \mathcal{DK}_{i_1} \times \cdots \times \mathcal{DK}_{i_j}$ be a set of possible secret keys of $\mathcal{J}$. Let M, $C_{\mathcal{S}}$, DK_i $(1 \leq i \leq n)$, and $DK_{\mathcal{J}}$ $(\mathcal{J} \subset \mathcal{U})$ be random variables which takes values on $\mathcal{M}$, $\mathcal{C}_{\mathcal{S}}$, $\mathcal{DK}_i$ $(1 \leq i \leq n)$, and $\mathcal{DK}_{\mathcal{J}}$ $(\mathcal{J} \subset \mathcal{U})$, respectively. Formally, security of a BE scheme is defined as follows.

Definition 5.2 (Security of BE). *Let Π_{BE} be a BE scheme. Π_{BE} is said to be $(\leq n, \leq \omega)$-one-time secure if the following conditions are satisfied: For any privileged set $\mathcal{S} \subset \mathcal{U}$, and any set of colluders $\mathcal{W} \in \mathcal{PS}(\mathcal{U}, \omega)$ such that $\mathcal{S} \cap \mathcal{W} = \emptyset$, it holds that $H(M \mid C_{\mathcal{S}}, DK_{\mathcal{W}}) = H(M)$.*

5.3 $(\leq n, \leq \omega)$-one-time Secure Revocable-Storage Broadcast Encryption

5.3.1 Model and Security Definition

In RS-BE, there are $n + 2$ entities, a sender E, n users $U_1, \ldots, U_n$, and a storage manager SM. As in BE schemes, let $\mathcal{U} := \{U_1, \ldots, U_n\}$ be a set of all users. First, E generates own encryption key ek, also generates n decryption keys $dk_1, \ldots, dk_n$ and a maintenance key mk behalf of $U_1, \ldots, U_n, SM$, and distributes them securely. E can specify a privileged set $\mathcal{S}$ of $\mathcal{U}$ such that $\mathcal{S} \neq \emptyset$, and encrypt a plaintext by using his encryption key ek so that only users in the privileged set can decrypt the resulting ciphertext. The ciphertext is stored and disclosed in an external storage such as cloud storage. A user U_i in the privileged set $\mathcal{S}$ takes the ciphertext from the storage himself, then he decrypts the ciphertext by using his decryption key dk_i. The storage manager SM can change *any* privileged set $\mathcal{S}$ of the ciphertext into *any* privileged set $\mathcal{S}'$ (even if *not* $\mathcal{S}' \subset \mathcal{S}$) by using his maintenance key mk without decryption (i.e., without revealing the underlying plaintext). At sender's request or by some kind of rule, the storage manager SM changes the privileged set of the ciphertext, and then SM replaces the old one with the new one.

Formally, we consider BE with the updating algorithm Upd as RS-BE. $\mathcal{M}$, $\mathcal{C}_{\mathcal{J}}$, $\mathcal{C}$, $\mathcal{EK}$, $\mathcal{DK}_i$, and $\mathcal{DK}$ are the same as those of BE. In addition, let $\mathcal{MK}$ be a set of maintenance keys.

Definition 5.3 (RS-BE). *An RS-BE scheme Π_{RSBE} involves $n+2$ entities, E, $U_1, U_2, \ldots, U_n$ and SM, and consists of the following four-tuple of algorithms (Setup, Enc, Dec, Upd) with five spaces, $\mathcal{M}, \mathcal{C}, \mathcal{EK}, \mathcal{DK}$, and $\mathcal{MK}$, where all of*

the above algorithms except Setup *are deterministic and all of the above spaces are finite.*

1. $(ek, mk, dk_1, \ldots, dk_n) \leftarrow \mathsf{Setup}(n)$*: It takes the number of users* n *as input, and outputs an encryption key* $ek \in \mathcal{EK}$*,* n *decryption keys* $(dk_1, \ldots, dk_n) \in \prod_{i=1}^{n} \mathcal{DK}_i$*, and a maintenance key* $mk \in \mathcal{MK}$*.*

2. $c_{\mathcal{S}} \leftarrow \mathsf{Enc}(ek, m, \mathcal{S})$*: It takes an encryption key* ek*, a plaintext* $m \in \mathcal{M}$*, and an initial privileged set* $\mathcal{S} \subset \mathcal{U}$ *as input, and outputs a ciphertext* $c_{\mathcal{S}}$*.*

3. m *or* $\perp \leftarrow \mathsf{Dec}(dk_i, c_{\mathcal{S}}, \mathcal{S}, U_i)$*: It takes a decryption key* dk_i *of a user* U_i*, the ciphertext* $c_{\mathcal{S}}$*, the privileged set* $\mathcal{S}$*, and the identity* U_i *as input, and outputs* m *or* $\perp$*.*

4. $c_{\mathcal{S}'}$ *or* $\perp \leftarrow \mathsf{Upd}(mk, c_{\mathcal{S}}, \mathcal{S}, \mathcal{S}')$*: It takes a maintenance key* mk*, the ciphertext* $c_{\mathcal{S}}$*, its privileged set* $\mathcal{S}$*, and a new privileged set* $\mathcal{S}'$ *as input, and outputs a ciphertext* $c_{\mathcal{S}'}$ *for* $\mathcal{S}'$ *or* $\perp$*.*

In RS-BE Π_{RSBE}, we require the following correctness holds: (a) For all $n \in \mathbb{N}$, all $(ek, mk, dk_1, \ldots, dk_n) \leftarrow \mathsf{Setup}(n)$, all $m \in \mathcal{M}$, all $\mathcal{S} \subset \mathcal{U}$, and all $U_i \in \mathcal{S}$, $m \leftarrow \mathsf{Dec}(dk_i, \mathsf{Enc}(ek, m, \mathcal{S}), \mathcal{S}, U_i)$. (b) For all $n \in \mathbb{N}$, all $(ek, mk, dk_1, \ldots, dk_n) \leftarrow \mathsf{Setup}(n)$, all $m \in \mathcal{M}$, all $\mathcal{S}, \mathcal{S}' \subset \mathcal{U}$, $\mathsf{Upd}(mk, \mathsf{Enc}(ek, m, \mathcal{S}), \mathcal{S}') = \mathsf{Enc}(ek, m, \mathcal{S}')$. (a) means the *decryption correctness* and (b) means the *updating correctness*.

In RS-BE, it is unrestricted for the storage manager to execute the algorithm Upd (i.e. the ciphertext can be updated unboundedly).

We consider perfect secrecy against at most ω colluders and the storage manager. Here, we note that in principle, it is impossible to guarantee security against collusion of them since the storage manager can change any privileged set of a ciphertext into any privileged set. Therefore, we consider security in the case that at most ω colluders and the storage manager try to attack separately.[2] Namely, we consider the following two kinds of security notions: (1) At most ω colluders who are not included in the privileged set cannot get any information on the underlying plaintext from the ciphertext (a traditional security notion for BE). (2) The storage manager cannot get any information on the underlying plaintext from the ciphertext. The reason why we consider the second one is that if the storage manager can obtain the underlying plaintext or some information on it, it is only necessary to encrypt the same plaintext with a new privileged set and replace an old ciphertext with the new one by a sender to change privileged sets. Hence, we require the storage manager can update the ciphertext without decryption (without leaking any information on the underlying plaintext). For any $\mathcal{J} := \{U_{i_1}, \ldots, U_{i_j}\} \subset \mathcal{U}$,

[2]We also discuss a RS-BE scheme secure against collusion of at most ω colluders and the storage manager under a restricted transformation rule of the storage manager's key in Section 5.4.1.

let $\mathcal{DK}_{\mathcal{J}} := \mathcal{DK}_{i_1} \times \cdots \times \mathcal{DK}_{i_j}$ be a set of possible secret keys of $\mathcal{J}$. Let M, $C_{\mathcal{S}}$, EK, DK_i $(1 \leq i \leq n)$, $DK_{\mathcal{J}}$ $(\mathcal{J} \subset \mathcal{U})$, and MK be random variables which takes values on $\mathcal{M}$, $\mathcal{C}_{\mathcal{S}}$, $\mathcal{EK}$, $\mathcal{DK}_i$ $(1 \leq i \leq n)$, $\mathcal{DK}_{\mathcal{J}}$ $(\mathcal{J} \subset \mathcal{U})$, and $\mathcal{MK}$, respectively. Formally, security of RS-BE is defined as follows.

Definition 5.4 (Security of RS-BE). *Let Π_{RSBE} be an RS-BE scheme. Π_{RSBE} is said to be $(\leq n, \leq \omega)$-one-time secure if the following conditions are satisfied:*

(1) For any privileged set $\mathcal{S} \subset \mathcal{U}$, and any set of colluders $\mathcal{W} \in \mathcal{PS}(\mathcal{U}, \omega)$ such that $\mathcal{S} \cap \mathcal{W} = \emptyset$, it holds that $H(M \mid C_{\mathcal{S}}, DK_{\mathcal{W}}) = H(M)$.

(2) For any privileged set $\mathcal{S} \subset \mathcal{U}$, it holds that $H(M \mid C_{\mathcal{S}}, MK) = H(M)$.

Remark 5.1. *In the model of RS-BE (Definition 5.3), if SM does not exist (i.e., mk is empty string and we do not consider the algorithm Upd), and we therefore do not consider the condition (2) in Definition 5.4, then Definitions 5.3 and 5.4 are the same as those of ($(\leq n, \leq \omega)$-one-time secure) traditional BE schemes (Definitions 5.1 and 5.2). Hence, we can say our scheme is natural extension of the BE schemes.*

Remark 5.2. *The condition (1) in Definition 5.4 implies that the number of ciphertexts taken by $\mathcal{W}$ from the storage is at most one. However, it is natural to think that $\mathcal{W}$ can access the storage multiple time and take ciphertexts for various privileged sets. Namely, for more realistic definition, we should consider the following security condition (1') instead of (1):*

(1') For any privileged sets $\mathcal{S}_1, \ldots, \mathcal{S}_k \subset \mathcal{U}$ $(1 \leq k \leq 2^n)$, and any set of colluders $\mathcal{W} \in \mathcal{PS}(\mathcal{U}, \omega)$ such that $\left(\bigcup_{i=1}^{k} \mathcal{S}_i\right) \cap \mathcal{W} = \emptyset$, it holds that $H(M \mid C_{\mathcal{S}_1}, \ldots, C_{\mathcal{S}_k}, DK_{\mathcal{W}}) = H(M)$.

For convenience, we call Π_{RSBE} a strongly secure RS-BE scheme if it satisfies the conditions (1') and (2), and just call Π_{RSBE} a secure RS-BE scheme if it satisfies Definition 5.4 (the conditions (1) and (2)). Actually, tight lower bounds on secret keys required for such a strongly secure RS-BE scheme are the same as those required for the secure RS-BE scheme (the bounds will appear in Theorem 5.2). Therefore, we can obtain the same optimal construction, in the sense that the construction meets equality in every lower bound, which will be proposed in Section 5.3.3. In addition to this, to deal with RS-BE as natural extension of traditional BE, we consider the above weaker security definition (Definition 5.4).

5.3.2 Tight Lower Bounds on Sizes of Ciphertexts and Secret Keys

In this section, we show lower bounds on the sizes of ciphertexts and secret keys required for a $(\leq n, \leq \omega)$-one-time secure RS-BE scheme. As mentioned

in [22, 92, 110, 111], in traditional BE schemes, there is a trade-off between the ciphertext size and the secret key size. RS-BE schemes also have such a trade-off. Actually, if we ignore the size of a ciphertext, it is not difficult to construct an ($\leq n, \leq \omega$)-one-time secure RS-BE scheme which is fairly efficient in other aspects, and the concrete construction is as follows. A sender E has n secret keys $k_1, \ldots, k_n$ and a common key K shared among E and all users $U_1, \ldots, U_n$ as ek, each user U_i has k_i and K as dk_i, and a storage manager SM has $k_1, \ldots, k_n$ as mk. E encrypts a plaintext m by $ct_{i_j} := m + k_{i_j} + K$ for every $U_{i_j} \in \mathcal{S}$ ($1 \leq j \leq |\mathcal{S}|$), and outputs $c_{\mathcal{S}} := (ct_{i_1}, \ldots, ct_{i_{|\mathcal{S}|}})$. For updating the ciphertext, SM computes $ct = ct_\ell - k_\ell = m + K$ for $U_\ell \in \mathcal{S}$ and $ct_{i_j} := ct + k_{i_j}$ for every $U_{i_j} \in \mathcal{S}'$ ($1 \leq j \leq |\mathcal{S}'|$), and then SM outputs $c_{\mathcal{S}'} := (ct_{i_1}, \ldots, ct_{i_{|\mathcal{S}'|}})$. Then, we have $|c_{\mathcal{S}}| = |\mathcal{S}| \cdot |m|$ for every $\mathcal{S} \in \mathcal{U}$, $|ek| = (n+1)|m|$, $|dk_i| = 2|m|$ ($1 \leq i \leq n$), and $|mk| = n|m|$. The sizes of secret keys of this scheme are significantly smaller than those of our construction which will be proposed in Section 5.3.3 though the ciphertext length is proportional to the cardinality of the privileged set; on the other hand, that of the proposed scheme is equal to the plaintext length for any privileged set.

However, when we consider applying RS-BE to a cloud storage, compactness of a ciphertext is one of the most important factors to be taken into account, since in such a scenario, a ciphertext is stored in cloud permanently or for a long-time, and thus, the ciphertext length should be as small as possible. For the above reason, we first investigate the *tight* lower bound on the size of ciphertexts, and then, derive lower bounds on sizes of secret keys under the condition that the ciphertext length is optimal.

Theorem 5.1. *Let Π_{RSBE} be an ($\leq n, \leq \omega$)-one-time secure RS-BE scheme. Then, for any $\mathcal{S} \subset \mathcal{U}$, $H(C_{\mathcal{S}}) \geq H(M)$ and there exists a concrete construction which meets this bound with equality.*

Proof. For any $\mathcal{S} \subset \mathcal{U}$ and $U_i \in \mathcal{S}$, we have

$$\begin{aligned} H(C_{\mathcal{S}}) \geq & H(C_{\mathcal{S}} \mid DK_i) && (5.1)\\ \geq & H(C_{\mathcal{S}} \mid DK_i) - H(C_{\mathcal{S}} \mid DK_i, M) && (5.2)\\ = & I(C_{\mathcal{S}}; M \mid DK_i) = H(M \mid DK_i) - H(M \mid DK_i, C_{\mathcal{S}}) = H(M), \end{aligned}$$

where the last equality follows from independence of M and DK_i and the decryption correctness.

Then, we show a construction which meets this bound with equality. A secret key of the one-time pad is assigned for every possible $\mathcal{S} \subset \mathcal{U}$. Namely, $ek := \{k_{\mathcal{S}} \mid \mathcal{S} \subset \mathcal{U}\}$, $dk_i := (k_\emptyset, \{k_{\mathcal{S}} \mid \mathcal{S} \subset \mathcal{U} \wedge U_i \in \mathcal{S}\})$ ($1 \leq i \leq n$), and $mk := \{k_{\mathcal{S}} \mid \mathcal{S} \subset \mathcal{U} \wedge \mathcal{S} \neq \emptyset\}$, where each $k_{\mathcal{S}}$ is chosen from a finite field uniformly at random. In Enc, for any $\mathcal{S}$, it outputs $c_{\mathcal{S}} := m + k_\emptyset + k_{\mathcal{S}}$. In Dec, if $U_i \in \mathcal{S}$, it can output $m = c_{\mathcal{S}} - k_\emptyset - k_{\mathcal{S}}$. In Upd, for any $\mathcal{S}$ and $\mathcal{S}'$, it outputs $c_{\mathcal{S}'} := c_{\mathcal{S}} - k_{\mathcal{S}} + k_{\mathcal{S}'}$. This construction is ($\leq n, \leq \omega$)-one-time secure

since any $\mathcal{W}$ such that $\mathcal{S} \cap \mathcal{W} = \emptyset$ does not have $k_{\mathcal{S}}$ and SM does not have $k_{\emptyset}$. □

Next, we derive lower bounds on sizes of secret keys when the ciphertext size is optimal (i.e. the ciphertext length is equal to the plaintext length).

Theorem 5.2. *Let Π_{RSBE} be an $(\leq n, \leq \omega)$-one-time secure RS-BE scheme. Then, the following lower bounds hold under the condition $H(C_{\mathcal{S}}) = H(M)$ for any $\mathcal{S} \subset \mathcal{U}$:*

$$\textit{(i)}\ H(EK) \geq \sum_{j=0}^{\omega} \binom{n}{j} H(M),$$

$$\textit{(ii)}\ H(DK_i) \geq \sum_{j=0}^{\omega} \binom{n-1}{j} H(M) \textit{ for any } i \in [n],$$

$$\textit{(iii)}\ H(MK) \geq \left(\sum_{j=0}^{\omega} \binom{n}{j} - 1 \right) H(M).$$

Proof. The proof follows from the following lemmas.

Lemma 5.1. *For any $\mathcal{S} \subset \mathcal{U}$ and any $\mathcal{W} \in \mathcal{PS}(\mathcal{U}, \omega)$ such that $\mathcal{W} \cap \mathcal{S} = \emptyset$, let Y_i $(1 \leq i \leq k)$ be a privileged set such that $Y_i \cap \mathcal{W} \neq \emptyset$. Then, we have $H(C_{\mathcal{S}} \mid M, C_{Y_1}, \ldots, C_{Y_k}, DK_{\mathcal{W}}) \geq H(M)$ under the condition $H(C_{\mathcal{S}}) = H(M)$ for any $\mathcal{S} \subset \mathcal{U}$.*

Proof. From Eqs. (5.1) and (5.2) in Theorem 5.1, and the condition $H(C_{\mathcal{S}}) = H(M)$, we have $H(C_{\mathcal{S}} \mid DK_i) = H(C_{\mathcal{S}} \mid DK_i) - H(C_{\mathcal{S}} \mid DK_i, M)$ for any $\mathcal{S} \subset \mathcal{U}$ and $U_i \in \mathcal{S}$. Therefore, we have

$$H(C_{\mathcal{S}} \mid DK_i, M) = 0. \tag{5.3}$$

For $H(M, C_{\mathcal{S}}, C_{Y_1}, \ldots, C_{Y_k} \mid DK_{\mathcal{W}})$, we have

$$\begin{aligned} &H(M, C_{\mathcal{S}}, C_{Y_1}, \ldots, C_{Y_k} \mid DK_{\mathcal{W}}) \\ &\quad = H(C_{\mathcal{S}} \mid DK_{\mathcal{W}}) + H(M \mid DK_{\mathcal{W}}, C_{\mathcal{S}}) + H(C_{Y_1}, \ldots, C_{Y_k} \mid DK_{\mathcal{W}}, C_{\mathcal{S}}, M) \\ &\quad = H(C_{\mathcal{S}} \mid DK_{\mathcal{W}}) + H(M) + H(C_{Y_1}, \ldots, C_{Y_k} \mid DK_{\mathcal{W}}, C_{\mathcal{S}}, M) \qquad (5.4) \\ &\quad = H(C_{\mathcal{S}} \mid DK_{\mathcal{W}}) + H(M), \qquad (5.5) \end{aligned}$$

where Eq. (5.4) follows from the condition (1) of Definition 5.4, and Eq. (5.5) follows from Eq. (5.3) (i.e. $H(C_{Y_j} \mid DK_{\mathcal{W}}, M) = 0$) since $Y_j \cap \mathcal{W} \neq \emptyset$ for any Y_j $(1 \leq j \leq k)$.

On the other hand, for $H(M, C_{\mathcal{S}}, C_{Y_1}, \ldots, C_{Y_k} \mid DK_{\mathcal{W}})$, we have

$$H(M, C_{\mathcal{S}}, C_{Y_1}, \ldots, C_{Y_k} \mid DK_{\mathcal{W}})$$

$$
\begin{aligned}
&= H(M \mid DK_{\mathcal{W}}) + H(C_{Y_1}, \ldots, C_{Y_k} \mid DK_{\mathcal{W}}, M) \\
&\qquad + H(C_{\mathcal{S}} \mid DK_{\mathcal{W}}, M, C_{Y_1}, \ldots, C_{Y_k}) \\
&= H(M) + H(C_{\mathcal{S}} \mid DK_{\mathcal{W}}, M, C_{Y_1}, \ldots, C_{Y_k}), \qquad (5.6)
\end{aligned}
$$

where Eq. (5.6) follows from independence of M and $DK_{\mathcal{W}}$ and the same reason for Eq. (5.5).

Hence, from (5.5) and (5.6), we have

$$
H(C_{\mathcal{S}} \mid DK_{\mathcal{W}}, M, C_{Y_1}, \ldots, C_{Y_k}) = H(C_{\mathcal{S}} \mid DK_{\mathcal{W}}). \qquad (5.7)
$$

In the following, we show $H(C_{\mathcal{S}} \mid DK_{\mathcal{W}}) \geq H(M)$.

For $H(M, C_{\mathcal{S}} \mid DK_{\mathcal{S}}, DK_{\mathcal{W}}, EK)$, we have

$$
\begin{aligned}
&H(M, C_{\mathcal{S}} \mid DK_{\mathcal{S}}, DK_{\mathcal{W}}, EK) \qquad (5.8) \\
&\quad = H(C_{\mathcal{S}} \mid DK_{\mathcal{S}}, DK_{\mathcal{W}}, EK) + H(M \mid DK_{\mathcal{S}}, DK_{\mathcal{W}}, EK, C_{\mathcal{S}}) \\
&\quad = H(C_{\mathcal{S}} \mid DK_{\mathcal{S}}, DK_{\mathcal{W}}, EK), \qquad (5.9)
\end{aligned}
$$

where Eq. (5.9) follows from the decryption correctness (i.e. $H(M \mid DK_{\mathcal{S}}, C_{\mathcal{S}}) = 0$).

On the other hand, for $H(M, C_{\mathcal{S}} \mid DK_{\mathcal{S}}, DK_{\mathcal{W}}, EK)$, we have

$$
\begin{aligned}
&H(M, C_{\mathcal{S}} \mid DK_{\mathcal{S}}, DK_{\mathcal{W}}, EK) \\
&\quad = H(M \mid DK_{\mathcal{S}}, DK_{\mathcal{W}}, EK) + H(C_{\mathcal{S}} \mid DK_{\mathcal{S}}, DK_{\mathcal{W}}, EK, M) \\
&\quad = H(M \mid DK_{\mathcal{S}}, DK_{\mathcal{W}}, EK), \qquad (5.10)
\end{aligned}
$$

where Eq. (5.10) follows from the algorithm Enc (i.e. $H(C_{\mathcal{S}} \mid EK, M) = 0$).

Hence, we have

$$
\begin{aligned}
H(C_{\mathcal{S}} | DK_{\mathcal{W}}) &\geq H(C_{\mathcal{S}} \mid DK_{\mathcal{S}}, DK_{\mathcal{W}}, EK) \\
&= H(M \mid DK_{\mathcal{S}}, DK_{\mathcal{W}}, EK) \qquad (5.11) \\
&= H(M), \qquad (5.12)
\end{aligned}
$$

where Eq. (5.11) follows from Eqs. (5.9) and (5.10), and (5.12) follows from independence of M and $(EK, DK_1, \ldots, DK_n)$.

From Eqs. (5.7) and (5.12), we have $H(C_{\mathcal{S}} \mid M, C_{Y_1}, \ldots, C_{Y_k}, DK_{\mathcal{W}}) \geq H(M)$. □

Lemma 5.2. *We have* $H(EK) \geq \sum_{j=0}^{\omega} \binom{n}{j} H(M)$ *under the condition* $H(C_{\mathcal{S}}) = H(M)$ *for any* $\mathcal{S} \subset \mathcal{U}$.

Proof. Let $\mathscr{W} := \{\mathcal{W} \subset \mathcal{U} \mid |\mathcal{W}| \leq \omega\} = \{\mathcal{W}_1, \ldots, \mathcal{W}_t\}$ be the family of all possible sets of colluders, where $t = \sum_{j=0}^{\omega} \binom{n}{j}$. Moreover, let $\mathscr{S}(\mathscr{W}) := \{\mathcal{S}_1, \ldots, \mathcal{S}_t\}$, where $\mathcal{S}_i = \mathcal{U} \setminus \mathcal{W}_i$ such that $\mathcal{W}_i \in \mathscr{W}$ $(1 \leq i \leq t)$. Without loss of generality, $|\mathcal{S}_1| \geq \cdots \geq |\mathcal{S}_t|$. Then, we have

$$
\begin{aligned}
H(EK) =& H(EK \mid M) && (5.13)\\
\geq& I(EK; C_{\mathcal{S}_1}, \ldots, C_{\mathcal{S}_t} \mid M)\\
=& H(C_{\mathcal{S}_1}, \ldots, C_{\mathcal{S}_t} \mid M) - H(C_{\mathcal{S}_1}, \ldots, C_{\mathcal{S}_t} \mid M, EK)\\
=& H(C_{\mathcal{S}_1}, \ldots, C_{\mathcal{S}_t} \mid M) && (5.14)\\
=& \sum_{j=1}^{t} H(C_{\mathcal{S}_j} \mid M, C_{\mathcal{S}_1}, \ldots, C_{\mathcal{S}_{j-1}})\\
\geq& \sum_{j=1}^{t} H(C_{\mathcal{S}_j} \mid M, C_{\mathcal{S}_1}, \ldots, C_{\mathcal{S}_{j-1}}, DK_{\mathcal{W}_j})\\
\geq& \sum_{j=0}^{\omega} \binom{n}{j} H(M), && (5.15)
\end{aligned}
$$

where Eq. (5.13) follows from independence of M and EK, Eq. (5.14) follows from the algorithm Enc (i.e. $H(C_{\mathcal{S}_i} \mid EK, M) = 0$ $(1 \leq i \leq t)$), and Eq. (5.15) follows from Lemma 5.1. □

Lemma 5.3. *For any $i \in [n]$, we have $H(DK_i) \geq \sum_{j=0}^{\omega} \binom{n-1}{j} H(M)$ under the condition $H(C_{\mathcal{S}}) = H(M)$ for any $\mathcal{S} \subset \mathcal{U}$.*

Proof. Let $\mathscr{W}^{(i)} := \{\mathcal{W} \subset \mathcal{U} \setminus \{U_i\} \mid |\mathcal{W}| \leq \omega\} = \{\mathcal{W}_1, \ldots, \mathcal{W}_\ell\}$ be the family of all possible sets of colluders except for sets of colluders containing U_i, where $\ell = \sum_{j=0}^{\omega} \binom{n-1}{j}$. Moreover, let $\mathscr{S}(\mathscr{W}^{(i)}) := \{\mathcal{S}_1, \ldots, \mathcal{S}_\ell\}$, where $\mathcal{S}_i = \mathcal{U} \setminus \mathcal{W}_i$ such that $\mathcal{W}_i \in \mathscr{W}^{(i)}$ $(1 \leq i \leq \ell)$. Without loss of generality, $|\mathcal{S}_1| \geq \cdots \geq |\mathcal{S}_\ell|$. We note $U_i \in \mathcal{S}$ for any $\mathcal{S} \in \mathscr{S}(\mathscr{W}^{(i)})$. Then, we have

$$
\begin{aligned}
H(DK_i) =& H(DK_i \mid M) && (5.16)\\
\geq& I(DK_i; C_{\mathcal{S}_1}, \ldots, C_{\mathcal{S}_\ell} \mid M)\\
=& H(C_{\mathcal{S}_1}, \ldots, C_{\mathcal{S}_\ell} \mid M) - H(C_{\mathcal{S}_1}, \ldots, C_{\mathcal{S}_\ell} \mid M, DK_i)\\
=& H(C_{\mathcal{S}_1}, \ldots, C_{\mathcal{S}_\ell} \mid M) && (5.17)\\
=& \sum_{j=1}^{\ell} H(C_{\mathcal{S}_j} \mid M, C_{\mathcal{S}_1}, \ldots, C_{\mathcal{S}_{j-1}})\\
\geq& \sum_{j=1}^{\ell} H(C_{\mathcal{S}_j} \mid M, C_{\mathcal{S}_1}, \ldots, C_{\mathcal{S}_{j-1}}, DK_{\mathcal{W}_j})\\
\geq& \sum_{j=0}^{\omega} \binom{n-1}{j} H(M), && (5.18)
\end{aligned}
$$

where Eq. (5.16) follows from independence of M and DK_i, Eq. (5.17) follows from Eq. (5.3) in Lemma 5.1 (i.e. $H(C_{\mathcal{S}_j} \mid DK_i, M) = 0$ $(1 \leq j \leq \ell)$), and Eq. (5.18) follows from Lemma 5.1. □

Lemma 5.4. *We have* $H(MK) \geq \left(\sum_{j=0}^{\omega} \binom{n}{j} - 1\right) H(M)$ *under the condition* $H(C_{\mathcal{S}}) = H(M)$ *for any* $\mathcal{S} \subset \mathcal{U}$.

Proof. Let $\mathscr{W}$ and $\mathscr{S}(\mathscr{W})$ be the same as those in Lemma 5.2. Then, we have

$$
\begin{aligned}
H(MK) \geq & H(MK \mid C_{\mathcal{S}_1}) \\
\geq & I(MK; C_{\mathcal{S}_2}, \ldots, C_{\mathcal{S}_t} \mid C_{\mathcal{S}_1}) \\
= & H(C_{\mathcal{S}_2}, \ldots, C_{\mathcal{S}_t} \mid C_{\mathcal{S}_1}) - H(C_{\mathcal{S}_2}, \ldots, C_{\mathcal{S}_t} \mid C_{\mathcal{S}_1}, MK) \\
= & H(C_{\mathcal{S}_2}, \ldots, C_{\mathcal{S}_t} \mid C_{\mathcal{S}_1}) \qquad (5.19) \\
= & \sum_{j=2}^{t} H(C_{\mathcal{S}_j} \mid C_{\mathcal{S}_1}, \ldots, C_{\mathcal{S}_{j-1}}) \\
\geq & \sum_{j=2}^{t} H(C_{\mathcal{S}_j} \mid M, C_{\mathcal{S}_1}, \ldots, C_{\mathcal{S}_{j-1}}, DK_{\mathcal{W}_j}) \\
\geq & \left(\sum_{j=0}^{\omega} \binom{n}{j} - 1\right) H(M), \qquad (5.20)
\end{aligned}
$$

where Eq. (5.19) follows from the algorithm Upd (i.e. $H(C_{\mathcal{S}_i} \mid C_{\mathcal{S}_1}, MK) = 0$ $(2 \leq i \leq t)$), and Eq. (5.20) follows from Lemma 5.1. □

Proof of Theorem 5.2: Now, the proof is completed. □

As we will see in the next section, the above lower bounds are tight since our construction will meet all the above bounds with equalities. Therefore, we define optimality of constructions of RS-BE as follows.

Definition 5.5 (Optimality). *A construction of an* ($\leq n, \leq \omega$)*-one-time secure RS-BE scheme is said to be* optimal *if it meets equality in every bound of (i)–(iii) in Theorem 5.2.*

In a similar way, we can also derive tight lower bounds on secret keys required for another class of RS-BE schemes, called ($t, \leq \omega$)-one-time secure RS-BE schemes, in which the number of privileged users is constant in all time periods, and show an optimal construction under this condition (see Section 5.3.4 for details).

5.3.3 Optimal Construction

In this section, we propose an optimal construction of $(\leq n, \leq \omega)$-one-time secure RS-BE based on the Fiat–Naor KPS[3] [61] (see Section 2.4). We define the following families of sets:

$$\begin{aligned}
\mathscr{W} &:= \{\mathcal{W} \subset \mathcal{U} \mid |\mathcal{W}| \leq \omega\}, \\
\mathscr{W}^{(i)} &:= \{\mathcal{W} \subset \mathcal{U} \setminus \{U_i\} \mid |\mathcal{W}| \leq \omega\}, \\
\mathscr{W}(\mathcal{S}) &:= \{\mathcal{W} \in \mathscr{W} \mid (\mathcal{W} \cap \mathcal{S} = \emptyset \wedge |\mathcal{W}| = \min(\omega, n - |\mathcal{S}|)) \vee \mathcal{W} = \emptyset\}.
\end{aligned}$$

Our construction is as follows.

1. $(ek, mk, dk_1, \ldots, dk_n) \leftarrow \mathsf{Setup}(n)$: Let q be a prime power such that $q > n$, and $\mathbb{F}_q$ be a finite field with q elements. For every $\mathcal{W} \in \mathscr{W}$, it chooses $r_{\mathcal{W}} \in \mathbb{F}_q$ uniformly at random. Then, it outputs $ek := \{r_{\mathcal{W}} \mid \mathcal{W} \in \mathscr{W}\}$, $dk_i := \{r_{\mathcal{W}} \mid \mathcal{W} \in \mathscr{W}^{(i)}\}$ $(1 \leq i \leq n)$, and $mk := \{r_{\mathcal{W}} \mid \mathcal{W} \in \mathscr{W} \setminus \{\emptyset\}\}$.

2. $c_{\mathcal{S}} \leftarrow \mathsf{Enc}(ek, m, \mathcal{S})$: For any privileged set $\mathcal{S}$, it computes a session key $k_{\mathcal{S}} := \sum_{\mathcal{W} \in \mathscr{W}(\mathcal{S})} r_{\mathcal{W}}$, and then outputs $c_{\mathcal{S}} := m + k_{\mathcal{S}}$.

3. m or $\perp \leftarrow \mathsf{Dec}(dk_i, c_{\mathcal{S}}, \mathcal{S}, U_i)$: If $U_i \in \mathcal{S}$, then it computes $k_{\mathcal{S}}$ as in the algorithm Enc and outputs $m = c_{\mathcal{S}} - k_{\mathcal{S}}$. Otherwise, it outputs $\perp$.

4. $c_{\mathcal{S}'}$ or $\perp \leftarrow \mathsf{Upd}(mk, c_{\mathcal{S}}, \mathcal{S}, \mathcal{S}')$: For any privileged sets $\mathcal{S}$ and $\mathcal{S}'$, it computes an updating key $uk_{\mathcal{S} \to \mathcal{S}'} := \sum_{\mathcal{W} \in \mathscr{W}(\mathcal{S}') \setminus \{\emptyset\}} r_{\mathcal{W}} - \sum_{\mathcal{W} \in \mathscr{W}(\mathcal{S}) \setminus \{\emptyset\}} r_{\mathcal{W}}$, and outputs $c_{\mathcal{S}'} := c_{\mathcal{S}} + uk_{\mathcal{S} \to \mathcal{S}'}$.

Theorem 5.3. *The resulting RS-BE scheme Π_{RSBE} by the above construction is $(\leq n, \leq \omega)$-one-time secure and optimal.*

Proof. First, we show the above construction meets the condition (1) in Definition 5.4. Without loss of generality, we consider that $\mathcal{W} := \{U_1, \ldots, U_\omega\}$ is a set of colluders and $\mathcal{S} := \{U_{\omega+1} \ldots, U_n\}$ is a privileged set. Consider the case that the set of colluders $\mathcal{W}$ will guess $k_{\mathcal{S}}$ to obtain $m = c_{\mathcal{S}} - k_{\mathcal{S}}$ by using their decryption keys. However, $\mathcal{W}$ cannot compute $k_{\mathcal{S}}$ since they do not have $r_{\mathcal{W}}$. Therefore, the best strategy of $\mathcal{W}$ is to make a random guess at m as in the one-time pad. Thus, we have $H(M \mid C_{\mathcal{S}}, DK_{\mathcal{W}}) = H(M)$. Similarly, for any privileged set $\mathcal{S} \subset \mathcal{U}$, any set of colluders $\mathcal{W} \subset \mathcal{U}$ such that $\mathcal{S} \cap \mathcal{W} = \emptyset$ and $|\mathcal{W}| \leq \omega$ does not have $r_{\mathcal{W}}$, though $r_{\mathcal{W}}$ is used for computing $k_{\mathcal{S}}$. Hence, for any $\mathcal{S} \subset \mathcal{U}$, and any $\mathcal{W} \subset \mathcal{U}$ such that $\mathcal{S} \cap \mathcal{W} = \emptyset$ and $|\mathcal{W}| \leq \omega$, $H(M \mid C_{\mathcal{S}}, DK_{\mathcal{W}}) = H(M)$.

Next, we show the above construction meets the condition (2) in Definition 5.4. Since $1 \leq |\mathcal{S}| \leq n$, $r_\emptyset$ is always used for computing $k_{\mathcal{S}}$ for any $\mathcal{S} \subset \mathcal{U}$,

[3] If we define a construction which meets equality in every bound of (i) and (ii) in Theorem 5.2 as an optimal construction of $(\leq n, \leq \omega)$-one-time secure BE, then we can obtain such an optimal construction from the Fiat–Naor KPS scheme and the one-time pad.

whereas SM does not have $r_\emptyset$. Hence, he can only guess m randomly as in the one-time pad. Thus, for any $\mathcal{S} \subset \mathcal{U}$, $H(M \mid C_\mathcal{S}, MK) = H(M)$.

Moreover, it is straightforward to see that the above construction is optimal. □

5.3.4 $(t, \leq \omega)$-one-time Secure RS-BE

As in traditional BE schemes [37, 87, 92, 110], we can also consider another class of RS-BE schemes, which is called $(t, \leq \omega)$-one-time secure RS-BE schemes, where $t + \omega \leq n$. A model and security of such a scheme are almost the same as that described in Section 5.3.1, and the only difference from those in Section 5.3.1 is that a sender can specify only a privileged set whose cardinality is exactly t (i.e., $|\mathcal{S}| = t$).

Then, we can derive lower bounds on secret keys in a similar way to Section 5.3.2, and these bounds can also be applied to traditional $(t, \leq \omega)$-one-time secure BE schemes.

Theorem 5.4. *Let Π_{RSBE} be a $(t, \leq \omega)$-one-time secure RS-BE scheme. Then, for any $\mathcal{S} \subset \mathcal{U}$, the following lower bounds hold under the condition $H(C_\mathcal{S}) = H(M)$:*

$$\begin{aligned}
&\textit{(i)}\ H(EK) \geq \binom{t+\omega}{t} H(M),\\
&\textit{(ii)}\ H(DK_i) \geq \binom{t+\omega-1}{t-1} H(M)\ \textit{for any}\ i \in [n],\\
&\textit{(iii)}\ H(MK) \geq \left(\binom{t+\omega}{t} - 1\right) H(M).
\end{aligned}$$

Proof. The proof follows from the following lemmas.

Lemma 5.5. *We have $H(EK) \geq \binom{t+\omega}{t} H(M)$ under the condition $H(C_\mathcal{S}) = H(M)$ for any $\mathcal{S} \subset \mathcal{U}$.*

Proof. Without loss of generality, let $\mathcal{I} := \{U_1, \ldots, U_{t+\omega}\}$. Let $\mathscr{W} := \{\mathcal{W} \subset \mathcal{I} \mid |\mathcal{W}| = \omega\} = \{\mathcal{W}_1, \ldots, \mathcal{W}_\ell\}$ be the family of all possible set of colluders, where $\ell = \binom{t+\omega}{\omega} = \binom{t+\omega}{t}$. Moreover, let $\mathscr{S}(\mathscr{W}) := \{\mathcal{S}_1, \ldots, \mathcal{S}_\ell\}$, where $\mathcal{S}_i = \mathcal{I} \setminus \mathcal{W}_i$ such that $\mathcal{W}_i \in \mathscr{W}$ $(1 \leq i \leq \ell)$. Then, we have

$$\begin{aligned}
H(EK) =& H(EK \mid M) && (5.21)\\
\geq& I(EK; C_{\mathcal{S}_1}, \ldots, C_{\mathcal{S}_\ell} \mid M)\\
=& H(C_{\mathcal{S}_1}, \ldots, C_{\mathcal{S}_\ell} \mid M) - H(C_{\mathcal{S}_1}, \ldots, C_{\mathcal{S}_\ell} \mid M, EK)\\
=& H(C_{\mathcal{S}_1}, \ldots, C_{\mathcal{S}_\ell} \mid M) && (5.22)\\
=& \sum_{j=1}^{\ell} H(C_{\mathcal{S}_j} \mid M, C_{\mathcal{S}_1}, \ldots, C_{\mathcal{S}_{j-1}})
\end{aligned}$$

$$\geq \sum_{j=1}^{\ell} H(C_{\mathcal{S}_j} \mid M, C_{\mathcal{S}_1}, \ldots, C_{\mathcal{S}_{j-1}}, DK_{\mathcal{W}_j})$$
$$\geq \binom{t+\omega}{t} H(M), \tag{5.23}$$

where Eq. (5.21) follows from independence of M and EK, Eq. (5.22) follows from the algorithm Enc (i.e. $H(C_{\mathcal{S}_i} \mid EK, M) = 0$ $(1 \leq i \leq \ell)$), and Eq. (5.23) follows from Lemma 5.1. □

Lemma 5.6. *For any $i \in [n]$, then we have $H(DK_i) \geq \sum_{j=0}^{\omega} \binom{n-1}{j} H(M)$ under the condition $H(C_{\mathcal{S}}) = H(M)$ for any $\mathcal{S} \subset \mathcal{U}$.*

Proof. Without loss of generality, let $\mathcal{I} := \{U_1, \ldots, U_i, \ldots, U_{t+\omega}\}$. Let $\mathscr{W}^{(i)} := \{\mathcal{W} \subset \mathcal{I} \setminus \{U_i\} \mid |\mathcal{W}| = \omega\} = \{\mathcal{W}_1, \ldots, \mathcal{W}_\ell\}$ be the family of all possible set of colluders except for sets of colluders containing U_i, where $\ell = \binom{t+\omega-1}{\omega} = \binom{t+\omega-1}{t-1}$. Let $\mathscr{S}(\mathscr{W}^{(i)}) := \{\mathcal{S}_1, \ldots, \mathcal{S}_\ell\}$, where $\mathcal{S}_i = \mathcal{I} \setminus \mathcal{W}_i$ such that $\mathcal{W}_i \in \mathscr{W}^{(i)}$ $(1 \leq i \leq \ell)$ We note $U_i \in \mathcal{S}$ for any $\mathcal{S} \in \mathscr{S}(\mathscr{W}^{(i)})$. Then, we have

$$\begin{aligned}
H(DK_i) =& H(DK_i \mid M) && (5.24)\\
\geq& I(DK_i; C_{\mathcal{S}_1}, \ldots, C_{\mathcal{S}_\ell} \mid M)\\
=& H(C_{\mathcal{S}_1}, \ldots, C_{\mathcal{S}_\ell} \mid M) - H(C_{\mathcal{S}_1}, \ldots, C_{\mathcal{S}_\ell} \mid M, DK_i)\\
=& H(C_{\mathcal{S}_1}, \ldots, C_{\mathcal{S}_\ell} \mid M) && (5.25)\\
=& \sum_{j=1}^{\ell} H(C_{\mathcal{S}_j} \mid M, C_{\mathcal{S}_1}, \ldots, C_{\mathcal{S}_{j-1}})\\
\geq& \sum_{j=1}^{\ell} H(C_{\mathcal{S}_j} \mid M, C_{\mathcal{S}_1}, \ldots, C_{\mathcal{S}_{j-1}}, DK_{\mathcal{W}_j})\\
\geq& \binom{t+\omega-1}{t-1} H(M), && (5.26)
\end{aligned}$$

where Eq. (5.24) follows from independence of M and DK_i, Eq. (5.25) follows from Eq. (5.3) in Lemma 5.1 (i.e. $H(C_{\mathcal{S}_j} \mid DK_i, M) = 0$ $(1 \leq j \leq \ell)$), and Eq. (5.26) follows from Lemma 5.1. □

Lemma 5.7. *We have $H(MK) \geq \left(\binom{t+\omega}{t} - 1\right) H(M)$ under the condition $H(C_{\mathcal{S}}) = H(M)$ for any $\mathcal{S} \subset \mathcal{U}$.*

Proof. Let $\mathcal{I}$, $\mathscr{W}$ and $\mathscr{S}(\mathscr{W})$ be the same as those in Lemma 5.5. Then, we have

$$\begin{aligned}
H(MK) \geq& H(MK \mid C_{\mathcal{S}_1})\\
\geq& I(MK; C_{\mathcal{S}_2}, \ldots, C_{\mathcal{S}_\ell} \mid C_{\mathcal{S}_1})\\
=& H(C_{\mathcal{S}_2}, \ldots, C_{\mathcal{S}_\ell} \mid C_{\mathcal{S}_1}) - H(C_{\mathcal{S}_2}, \ldots, C_{\mathcal{S}_\ell} \mid C_{\mathcal{S}_1}, MK)
\end{aligned}$$

$$
\begin{aligned}
&= H(C_{\mathcal{S}_2}, \ldots, C_{\mathcal{S}_\ell} \mid C_{\mathcal{S}_1}) && (5.27)\\
&= \sum_{j=2}^{\ell} H(C_{\mathcal{S}_j} \mid C_{\mathcal{S}_1}, \ldots, C_{\mathcal{S}_{j-1}}) \\
&\geq \sum_{j=2}^{\ell} H(C_{\mathcal{S}_j} \mid M, C_{\mathcal{S}_1}, \ldots, C_{\mathcal{S}_{j-1}}, DK_{\mathcal{W}_j}) \\
&\geq \left(\binom{t+\omega}{t} - 1 \right) H(M), && (5.28)
\end{aligned}
$$

where Eq. (5.27) follows from the algorithm Upd (i.e. $H(C_{\mathcal{S}_i} \mid C_{\mathcal{S}_1}, MK) = 0$ $(2 \leq i \leq \ell)$), and Eq. (5.28) follows from Lemma 5.1. □

The proof of Theorem 5.2: Now, the proof is completed. □

We can construct a $(t, \leq \omega)$-one-time secure RS-BE scheme based on the idea of our construction described in Section 5.3.3 and an ω-secure non-interactive t-conference KPS (or, the so-called $(t, \leq \omega)$-KPS) [23] as follows. We omit the security proof since it is easy to prove in a similar manner as the proof of Theorem 5.3.

1. $(ek, mk, dk_1, \ldots, dk_n) \leftarrow \mathsf{Setup}(n)$: Let $\mathbb{F}_q$ be a finite field with q $(> n)$ elements, where q is a prime power. It chooses a symmetric polynomial $f(x_1, \ldots, x_t) := \sum_{i_1=0}^{\omega} \cdots \sum_{i_t=0}^{\omega} a_{i_1 i_2 \cdots i_t} x_1^{i_1} \cdots x_t^{i_t}$ over $\mathbb{F}_q$, where $a_{i_1 i_2 \cdots i_t} = a_{\sigma(i_1)\sigma(i_2)\cdots\sigma(i_t)}$ for all permutations $\sigma = (\sigma(i_1), \sigma(i_2), \ldots, \sigma(i_t))$. Also, it computes $g(x_1, x_2, \ldots, x_t) := f(x_1, x_2, \ldots, x_t) - a_{00\cdots0}$. Then, it outputs $ek := f(x_1, x_2, \ldots, x_t)$, $dk_i := f(i, x_2, \ldots, x_t)$ $(1 \leq i \leq n)$, and $mk := g(x_1, x_2, \ldots, x_t)$.

2. $c_{\mathcal{S}} \leftarrow \mathsf{Enc}(ek, m, \mathcal{S})$: For any privileged set $\mathcal{S} := \{U_{i_1}, \ldots, U_{i_t}\}$, it computes a session key $k_{\mathcal{S}} := f(i_1, \ldots, i_t)$, and then outputs $c_{\mathcal{S}} := m + k_{\mathcal{S}}$.

3. m or $\perp \leftarrow \mathsf{Dec}(dk_i, c_{\mathcal{S}}, \mathcal{S}, U_i)$: If $U_i \in \mathcal{S}$, then it computes $k_{\mathcal{S}}$ as in the algorithm Enc and outputs $m = c_{\mathcal{S}} - k_{\mathcal{S}}$. Otherwise, it outputs $\perp$.

4. $c_{\mathcal{S}'}$ or $\perp \leftarrow \mathsf{Upd}(mk, c_{\mathcal{S}}, \mathcal{S}, \mathcal{S}')$: For any pair of privileged sets $\mathcal{S} := \{U_{i_1}, \ldots, U_{i_t}\}$ and $\mathcal{S}' := \{U_{j_1}, \ldots, U_{j_t}\}$, it computes and outputs $c_{\mathcal{S}'} := c_{\mathcal{S}} + g(j_1, \ldots, j_t) - g(i_1, \ldots, i_t)$.

Theorem 5.5. *The resulting RS-BE scheme Π_{RSBE} by the above construction is $(t, \leq \omega)$-one-time secure and meets equality in every bound of (i)–(iii) in Theorem 5.4.*

5.4 Extensions of RS-BE

5.4.1 Collusion Resistant Scheme

We consider security against collusion of at most ω colluders and a storage manager. Intuitively, if a storage manager can change any privileged set of a ciphertext into any privileged set by using his maintenance key mk, we cannot achieve RS-BE secure against collusion of a set of colluders and the storage manager. Therefore, here we simply set the following transformation rule for mk: For any $\mathcal{S}, \mathcal{S}' \subset \mathcal{U}$, $\mathsf{Upd}(mk, c_{\mathcal{S}}, \mathcal{S}, \mathcal{S}')$ outputs an updated ciphertext $c_{\mathcal{S}'}$ if $\mathcal{S}' \subset \mathcal{S}$ holds, otherwise it outputs $\perp$. Therefore, we consider only an $(\leq n, \leq \omega)$-one-time secure scheme.

We define collusion resistant security as follows.

Definition 5.6 (Collusion Resistant RS-BE). *Let Π_{RSBE} be an RS-BE scheme. Π_{RSBE} is said to be collusion-resistantly $(\leq n, \leq \omega)$-one-time secure if the following conditions are satisfied: For any privileged set $\mathcal{S} \subset \mathcal{U}$, and any set of colluders $\mathcal{W} \in \mathcal{PS}(\mathcal{U}, \omega)$ such that $\mathcal{S} \cap \mathcal{W} = \emptyset$, it holds that*

$$H(M \mid C_{\mathcal{S}}, DK_{\mathcal{W}}, MK) = H(M).$$

A construction which satisfies Definition 5.6 is as follows.

1. $(ek, mk, dk_1, \ldots, dk_n) \leftarrow \mathsf{Setup}(n)$: Let q be a prime power such that $q > n$, and $\mathbb{F}_q$ be a finite field with q elements. It chooses n polynomials $f^{(h)}(x) := \sum_{i=0}^{\omega} a_i x^i$ $(h = 1, \ldots, n)$ over $\mathbb{F}_q$ uniformly at random, and computes $n-1$ polynomials $g^{(\ell)}(x) := f^{(\ell)}(x) - f^{(\ell-1)}(x)$ $(2 \leq \ell \leq n)$. Then, it outputs $ek := f^{(1)}(x)$, $dk_i := (f^{(1)}(i), \ldots, f^{(n)}(i))$ $(1 \leq i \leq n)$, and $mk := (g^{(2)}(x), \ldots, g^{(n)}(x))$.

2. $c_{\mathcal{S}} \leftarrow \mathsf{Enc}(ek, m, \mathcal{S})$: Let $\mathcal{S} = \{U_{i_1}, \ldots, U_{i_k}\}$ $(1 \leq k \leq n)$ be a privileged set. For every U_{i_j}, it computes $c_{i_j}^{(1)} := m + f^{(1)}(i_j)$, and sets a counter $t := 1$. Finally, it outputs $c_{\mathcal{S}} := (t, c_{i_1}^{(t)}, \ldots, c_{i_k}^{(t)})$.

3. m or $\perp \leftarrow \mathsf{Dec}(dk_i, c_{\mathcal{S}}, \mathcal{S}, U_i)$: If $U_i \in \mathcal{S}$, it computes $m = c_i^{(t)} - f^{(t)}(i)$ and outputs it. Otherwise, it outputs $\perp$.

4. $c_{\mathcal{S}'}$ or $\perp \leftarrow \mathsf{Upd}(mk, c_{\mathcal{S}}, \mathcal{S}, \mathcal{S}')$: Let $\mathcal{S}' = \{U_{i_1}, \ldots, U_{i_k}\}$. If $\mathcal{S}' \subset \mathcal{S}$ does not hold, it outputs $\perp$. Otherwise, for every $U_{i_j} \in \mathcal{S}' \subset \mathcal{S}$, it computes $c_i^{(t+1)} := c_{i_j}^{(t)} + g^{(t+1)}(i_j)$ $(1 \leq j \leq k)$. Finally, it sets $t := t + 1$ and outputs $c_{\mathcal{S}'} := (t, c_{i_1}^{(t)}, \ldots, c_{i_k}^{(t)})$.

Theorem 5.6. *The resulting RS-BE scheme Π_{RSBE} by the above construction is collusion-resistantly $(\leq n, \leq \omega)$-one-time secure.*

Proof. It is not so difficult to prove this theorem. Without loss of generality, we consider that $\mathcal{W} := \{U_1, \ldots, U_\omega\}$ is a set of colluders and $\mathcal{S} := \{U_{\omega+1} \ldots, U_n\}$ is a privileged set. Consider the case that the set of colluders $\mathcal{W}$ and the storage manager will guess $k_{\mathcal{S}}$ to obtain the plaintext m by the using their secret keys. Since each degree of x of $f^{(h)}(x)$ $(1 \leq h \leq n)$ is at most ω, at most ω colluders cannot obtain $f^{(h)}(x)$ from $f^{(h)}(1), \ldots, f^{(h)}(\omega)$ $(1 \leq h \leq n)$. Hence, they cannot obtain any information on $f^{(h)}(x)$ $(1 \leq h \leq n)$ even if they have $g^{(\ell)}(x)$ $(2 \leq \ell \leq n)$. Hence, for any $\mathcal{S} \subset \mathcal{U}$, and any $\mathcal{W} \subset \mathcal{U}$ such that $\mathcal{S} \cap \mathcal{W} = \emptyset$ and $|\mathcal{W}| \leq \omega$, $H(M \mid C_{\mathcal{S}}, DK_{\mathcal{W}}, MK) = H(M)$. □

5.4.2 Robust Scheme

We now consider a scenario in which a maliciously behaving storage manager can try to modify the encrypted plaintext. This is related to *non-malleability* in the context of ordinary encryption. In a RS-BE scheme, malleability may cause a serious problem since the ciphertext is periodically updated, but an improper update carried out by a malicious storage manager may not be immediately detectable by the users. More specifically, we consider security against a storage manager who tries to modify a ciphertext so that a user in the privileged set obtains a modified plaintext which differs from an original plaintext encrypted by the sender. In addition to this, since ciphertexts of RS-BE schemes are stored in external storage such as cloud storage (in other words, the ciphertexts are accessible at anytime), we should also consider security against such a modification attack by colluders. Formally, we consider two types of adversaries as in Definition 5.4, and define the robustness of RS-BE as follows. We here consider only an $(\leq n, \leq \omega)$-one-time secure scheme, though we can also consider a robust $(t, \leq \omega)$-one-time secure scheme in the same manner.

Definition 5.7 (Robust RS-BE). *Let Π_{RSBE} be an $(\leq n, \leq \omega)$-one-time secure RS-BE scheme. Π_{RSBE} is said to be δ-robust if $\max\{P_1, P_2\} \leq 1 - \delta$, where P_1 and P_2 are defined as follows:*

(3) For any $\mathcal{S}_1, \ldots, \mathcal{S}_k \subset \mathcal{U}$ $(1 \leq k \leq 2^n)$, any $U_i \in \mathcal{S}_k$, and any $\mathcal{W} \in \mathcal{PS}(\mathcal{U}, \omega)$ such that $\left(\bigcup_{i=1}^{k} \mathcal{S}_i\right) \cap \mathcal{W} = \emptyset$, we define $P_1(\mathcal{S}_1, \ldots, \mathcal{S}_k, U_i, \mathcal{W})$ as:

$$P_1(\mathcal{S}_1, \ldots, \mathcal{S}_k, U_i, \mathcal{W}) := \max_{c'_{\mathcal{S}_k}} \max_{c_{\mathcal{S}_1}, \ldots, c_{\mathcal{S}_k}} \max_{dk_{\mathcal{W}}} \Pr[m' \leftarrow \mathsf{Dec}(dk_i, c'_{\mathcal{S}_k}, \mathcal{S}_k, U_i) \mid \{\mathsf{Enc}(ek, m, \mathcal{S}_j)\}_{1 \leq j \leq k}, dk_{\mathcal{W}}],$$

where $m' \notin \{m, \perp\}$ and $c_{\mathcal{S}_j} = \mathsf{Enc}(ek, m, \mathcal{S}_j)$ $(1 \leq j \leq k)$. Note that $\mathsf{Enc}(ek, m, \mathcal{S}_{j+1}) = \mathsf{Upd}(mk, \mathsf{Enc}(ek, m, \mathcal{S}_j), \mathcal{S}_j, \mathcal{S}_{j+1})$ for any $\mathcal{S}_j, \mathcal{S}_{j+1}$ $(1 \leq j \leq k-1)$ (the updating correctness). Then, P_1 is defined as $P_1 := \max_{\mathcal{S}_1, \ldots, \mathcal{S}_k, U_i, \mathcal{W}} P_1(\mathcal{S}_1, \ldots, \mathcal{S}_k, U_i, \mathcal{W})$.

(4) For any $\mathcal{S}, \mathcal{S}' \subset \mathcal{U}$ and any $U_i \in \mathcal{S}'$, we define $P_2(\mathcal{S}, \mathcal{S}', U_i)$ as:

$$P_2(\mathcal{S}, \mathcal{S}', U_i) := \max_{c'_{\mathcal{S}'}} \max_{c_{\mathcal{S}}} \max_{mk} \Pr[m' \leftarrow \mathsf{Dec}(dk_i, c'_{\mathcal{S}'}, \mathcal{S}', U_i) \mid \mathsf{Enc}(ek, m, \mathcal{S}), mk],$$

where $m' \notin \{m, \perp\}$ and $c_{\mathcal{S}} = \mathsf{Enc}(ek, m, \mathcal{S})$. Then, P_2 is defined as $P_2 := \max_{\mathcal{S}, \mathcal{S}', U_i} P_2(\mathcal{S}, \mathcal{S}', U_i)$.

We can construct a robust scheme by using an *algebraic manipulation detection code* (AMD-code), which is defined as follows.

Definition 5.8 (AMD-code [43])**.** *Let $\mathcal{M}_{\mathrm{AMD}}$ be a set of messages such that $|\mathcal{M}_{\mathrm{AMD}}| = \eta$, and $\mathbb{G}$ be a commutative group of order γ. An algebraic manipulation detection code (AMD-code) Π_{AMD} consists of the following two-tuple algorithms (Encode, Decode), where Encode is a probabilistic encoding map $\mathsf{Encode} : \mathcal{M}_{\mathrm{AMD}} \to \mathbb{G}$ and a deterministic decoding map $\mathsf{Decode} : \mathbb{G} \to \mathcal{M}_{\mathrm{AMD}} \cup \{\perp\}$ such that $\mathsf{Decode}(\mathsf{Encode}(m)) = m$ with probability one for every $m \in \mathcal{M}_{\mathrm{AMD}}$. Π_{AMD} is an $(\eta, \gamma, \varepsilon)$-AMD-code if for every $m \in \mathcal{M}_{\mathrm{AMD}}$ and for every $\delta \in \mathbb{G}$, the probability that $\mathsf{Decode}(\mathsf{Encode}(m) + \delta) \notin \{m, \perp\}$ is at most ε.*

A robust RS-BE scheme is constructed by modifying the construction proposed in Section 5.3.3 as follows: Before encrypting a plaintext $m \in \mathbb{F}_q$, the Enc algorithm runs $\hat{m} \leftarrow \mathsf{Encode}(m)$; and after decrypting a ciphertext, then the Dec algorithm runs $m \leftarrow \mathsf{Decode}(\tilde{m})$, where $\tilde{m}$ is the decryption result.

We obtain the following theorem, and omit the proof since it is straightforward.

Theorem 5.7. *If Π_{AMD} is an (q, q, ε)-AMD-code, then the resulting RS-BE scheme Π_{RSBE} by the above construction is $(\leq n, \leq \omega)$-one-time secure and ε-robust.*

5.5 Broadcast Encryption with Trade-offs between Communication and Storage

Toward RS-BE schemes with general ciphertext sizes, we propose an efficient generic construction of a BE scheme from KPSs. In the following, we focus on an $(\leq n, \leq \omega)$-one-time secure BE scheme when the maximum ciphertext size is an integer multiple of the plaintext size, whereas ciphertexts in most of the previous $(\leq n, \leq \omega)$-one-time secure BE schemes were assumed to be the same size as plaintexts.

Definition 5.9. *For an $(\leq n, \leq \omega)$-one-time secure BE scheme Π_{BE}, we define*

$$\delta := \frac{\max_{\mathcal{S} \subset \mathcal{U}} \log |\mathcal{C}_{\mathcal{S}}|}{\log |\mathcal{M}|}.$$

Then, Π_{BE} *is said to be* $(\leq n, \leq \omega; \delta)$*-one-time secure.*

We briefly describe two known constructions of $(\leq n, \leq \omega)$-one-time secure BE schemes from KPSs when $\delta = 1$ and $\delta = n$, respectively.

First, for the case $\delta = 1$ we describe an $(\leq n, \leq \omega; 1)$-one-time secure BE scheme, which is proposed by Fiat and Naor [61], from the $(\leq n, \leq \omega)$-KPS described in Section 2.4 and the one-time pad. Specifically, a sender obtains a secret key for a privileged set from the $(\leq n, \leq \omega)$-KPS, and then encrypts a plaintext with the secret key by the one-time pad. Blundo and Cresti [19] showed this Fiat–Naor construction is *optimal*.

The second one is an $(\leq n, \leq \omega; n)$-one-time secure BE scheme from n $(\leq 1, \leq 0)$-KPSs (i.e. from n one-time pads), which we call the trivial construction, for the case $\delta = n$. Namely, a sender generates n independent secret keys of the one-time pad, and each decryption key is one of the secret keys. The sender encrypts a plaintext with secret keys of the users in a privileged set by using the one-time pad, and an entire ciphertext is a concatenation of the resulting ciphertexts. This trivial construction is obviously optimal.

As can be seen in related works [19, 61, 87], it is meaningful to consider the case $\delta \geq 1$, and the case $\delta > n$ is not interesting since secret-key sizes cannot become shorter than those in the case $\delta = n$ due to a tight lower bound on secret-key sizes required for the one-time pad. Therefore, in the next section we propose a generic construction of an $(\leq n, \leq \omega; \delta)$-one-time secure BE scheme, which is an intermediate construction between the above two constructions, for arbitrary $\delta \in [n]$.

5.5.1 Generic Construction of $(\leq n, \leq \omega; \delta)$-one-time Secure BE scheme

We propose a generic construction of an $(\leq n, \leq \omega; \delta)$-one-time secure BE scheme from KPSs. Then, we show its instantiation such that the secret-key size can be minimized in it.

A basic idea is simple. First, we split a user set $\mathcal{U}$ into δ disjoint subsets $\mathcal{U}_1, \ldots, \mathcal{U}_\delta$ of $\mathcal{U}$. Then, we assign an $(\leq |\mathcal{U}_i|, \leq \omega_i)$-KPS with each subset $\mathcal{U}_i$, where $\omega_i := \min\{\omega, |\mathcal{U}_i| - 1\}$ since an $(\leq |\mathcal{U}_i|, \leq \omega)$-KPS such that $\omega \geq |\mathcal{U}_i|$ is meaningless. For a privileged set $\mathcal{S}$, let $\mathcal{S}_i := \mathcal{U}_i \cap \mathcal{S}$. Then, a session key $k_{\mathcal{S}_i}$ is generated by $(\leq |\mathcal{U}_i|, \leq \omega)$-KPS, and an entire ciphertext $c_{\mathcal{S}}$ is $c_{\mathcal{S}} := (m \oplus k_{\mathcal{S}_1}, \ldots, m \oplus k_{\mathcal{S}_\delta})$. However, there are various combinations of δ natural numbers such that the sum of the numbers is n. Formally, for any natural number $n \in \mathbb{N}$ and $\delta \in [n]$, we define the following set:

$$\mathcal{L}(n, \delta) := \left\{ (\ell_1, \ldots, \ell_\delta) \in \mathbb{N}^\delta \;\middle|\; (\ell_1 \geq \cdots \geq \ell_\delta) \wedge \sum_{i=1}^{\delta} \ell_i = n \right\}.$$

We often write $\boldsymbol{L} := (\ell_1, \ldots, \ell_\delta) \in \mathcal{L}(n, \delta)$. $\mathcal{L}(n, \delta)$ means a set of δ natural numbers such that the sum of the numbers is n, therefore we have to choose $\boldsymbol{L}$

such that secret-key sizes are minimized for fixed n, ω, and δ. Hence, we will clarify such an optimal condition for minimizing secret-key sizes in the next subsection.

Formally, our generic construction of a BE scheme $\Pi_{\mathrm{BE}} = (\mathsf{Setup}, \mathsf{Enc}, \mathsf{Dec})$ from KPSs $\Pi_{\mathrm{KPS}}^{(i)} = (\mathsf{Init}_i, \mathsf{Der}_i)$ $(1 \leq i \leq \delta)$ is given as follows.

- $(ek, dk_1, \ldots, dk_n) \leftarrow \mathsf{Setup}(n)$: Choose $\boldsymbol{L} \in \mathcal{L}(n, \delta)$. Without loss of generality, let $\mathcal{U}_i := \{U_{\sum_{j=1}^{i-1} \ell_j + 1}, \ldots, U_{\sum_{j=1}^{i} \ell_j}\}$.[4] Run $\mathsf{Init}_i(\ell_i) \rightarrow (uk_{\sum_{j=1}^{i-1} \ell_j + 1}, \ldots, uk_{\sum_{j=1}^{i} \ell_j})$ $(1 \leq i \leq \delta)$, and let $uk^{(i)}$ be the corresponding master key. Set and output $ek := (uk^{(1)}, \ldots, uk^{(\delta)})$ and $dk_i := uk_i$.

- $c_{\mathcal{S}} \leftarrow \mathsf{Enc}(ek, m, \mathcal{S})$: Choose $\mathcal{S} \subset \mathcal{U}$ and let $\mathcal{S}_i := \mathcal{S} \cap \mathcal{U}_i$ $(1 \leq i \leq \delta)$. If $\mathcal{S}_i \neq \emptyset$, then for some $U_j \in \mathcal{S}_i$, run $\mathsf{Der}_i(uk_j, \mathcal{S}_i) \rightarrow k_{\mathcal{S}_i}$ $(1 \leq i \leq \delta)$. Note that any uk_j can be derived from $uk^{(i)}$. Set and output $c_{\mathcal{S}} := (m \oplus k_{\mathcal{S}_i})_{\mathcal{S}_i \neq \emptyset}$.

- m or $\perp \leftarrow \mathsf{Dec}(dk_i, c_{\mathcal{S}}, \mathcal{S}, U_i)$: Parse $c_{\mathcal{S}}$ as $(c_{\mathcal{S}_{k_1}}, \ldots, c_{\mathcal{S}_{k_l}})$ and suppose $U_i \in \mathcal{U}_{k_j}$. If $U_i \in \mathcal{S}_{k_j}$, compute $k_{\mathcal{S}_{k_j}} \leftarrow \mathsf{Der}(uk_i, \mathcal{S}_{k_j})$ and then output $m = c_{\mathcal{S}_{k_j}} \oplus k_{\mathcal{S}_{k_j}}$. Otherwise, output $\perp$.

Theorem 5.8. *The above construction of* Π_{BE} *given by* $(\leq \ell_i, \leq \omega_i)$*-one-time secure KPSs* $\Pi_{\mathrm{KPS}}^{(i)}$ $(1 \leq i \leq \delta)$ *is* $(\leq n, \leq \omega; \delta)$*-one-time secure.*

Proof. It is not difficult to see the above scheme is $(\leq n, \leq \omega; \delta)$-one-time secure. The maximum ciphertext size of the above scheme is obviously $\delta \log |\mathcal{M}|$ since if each $\mathcal{S}_i \neq \emptyset$, then a ciphertext size is δ times longer than the underlying plaintext size. Without loss of generality, suppose $\mathcal{S}$ such that $|\mathcal{S}| = n - \omega$, and $c_{\mathcal{S}} := (c_{S_1}, \ldots, c_{S_j})$ (i.e. $S_{j+1} = \cdots = S_\delta = \emptyset$). Let $\mathcal{W} := \mathcal{U} \setminus \mathcal{S}$ and $\mathcal{W}_i := \mathcal{U}_i \setminus \mathcal{S}_i$. Note that $\mathcal{W}_k = \mathcal{U}_k$ $(j+1 \leq k \leq \delta)$. Then, we have $H(M \mid C_{\mathcal{S}}, DK_{\mathcal{W}}) = H(M \mid M \oplus K_{\mathcal{S}_1}, \ldots, M \oplus K_{\mathcal{S}_j}, DK_{\mathcal{W}})$. In each $(\leq \ell_i, \leq \omega_i)$-KPS $(1 \leq i \leq j)$, $\mathcal{W}_i$ $(|\mathcal{W}_i| \leq \omega_i)$ cannot get any information on a session key $k_{\mathcal{S}_i}$ from security definition of the KPS (Definition 2.6). In addition, $\mathcal{W}_i$ $(1 \leq i \leq j)$ cannot obtain any information on session keys $k_{\mathcal{S}_1}, \ldots, k_{\mathcal{S}_{i-1}}, k_{\mathcal{S}_{i+1}}, \ldots, k_{\mathcal{S}_j}$ since $\mathcal{W}_i$'s secret keys are independent of them. For the same reason, $\mathcal{W}_i$ $(j+1 \leq i \leq \delta)$ cannot also obtain any information on session keys $k_{\mathcal{S}_1}, \ldots, k_{\mathcal{S}_j}$ from their secret keys. Namely, for $i \in [\delta]$, we have $H(K_{\mathcal{S}_i} \mid DK_{\mathcal{W}_1}, \ldots, DK_{\mathcal{W}_\delta}) = H(K_{\mathcal{S}_i})$. Therefore, we have $H(M \mid C_{\mathcal{S}}, DK_{\mathcal{W}}) = H(M \mid C_{\mathcal{S}})$ since $DK_{\mathcal{W}}$ is independent of M and $\{M \oplus K_{\mathcal{S}_i}\}_{1 \leq i \leq j}$. Moreover, from security of the one-time pad, we have $H(M \mid C_{\mathcal{S}}) = H(M)$. □

[4] For example, when $n = 9$, $\delta = 3$, and $\ell_i = 3$ $(i = 1, 2, 3)$, then $\mathcal{U}_1 := \{U_1, U_2, U_3\}$, $\mathcal{U}_2 := \{U_4, U_5, U_6\}$, and $\mathcal{U}_3 := \{U_7, U_8, U_9\}$.

5.5.2 Optimal Parameters for Minimal Keys

To obtain the most efficient scheme in terms of the secret-key size, we have to carefully choose a combination of $(\leq \ell_i, \leq \omega)$-KPSs. We obtain the following theorem.

Theorem 5.9. *When we apply an optimal construction of each KPS* $\Pi_{\mathrm{KPS}}^{(i)}$ $(1 \leq i \leq \delta)$ *to the resulting* $(\leq n, \leq \omega; \delta)$*-one-time secure BE scheme* Π_{BE}*, then sizes of the secret keys required in the above construction are given by*

$$(i)\ \log|\mathcal{EK}| = \sum_{i=1}^{\delta}\sum_{j=0}^{\omega_i}\binom{\ell_i}{j}\log q,$$

$$(ii)\ \sum_{i=1}^{n}\log|\mathcal{DK}_i| = \sum_{i=1}^{\delta}\left(\ell_i\sum_{j=0}^{\omega_i}\binom{\ell_i-1}{j}\right)\log q.$$

Moreover, $\boldsymbol{L} \in \mathcal{L}(n,\delta)$ *minimizes the sizes of the encryption keys if it satisfies the following conditions:*

$$\begin{cases} \textit{arbitrary } \boldsymbol{L} & \textit{if } \omega = 0, \\ \boldsymbol{L} = (n-(\delta-1), 1, \ldots, 1) & \textit{if } \omega = 1, \\ \ell_1 - \ell_\delta = 0 & \textit{if } \omega \geq 2 \wedge n/\delta \in \mathbb{N}, \\ \ell_1 - \ell_\delta = 1 & \textit{otherwise.} \end{cases}$$

On the other hand, $\boldsymbol{L} \in \mathcal{L}(n,\delta)$ *minimizes the sizes of the decryption keys if it satisfies the following conditions:*

$$\begin{cases} \textit{arbitrary } \boldsymbol{L} & \textit{if } \omega = 0, \\ \ell_1 - \ell_\delta = 0 & \textit{if } \omega \geq 1 \wedge n/\delta \in \mathbb{N}, \\ \ell_1 - \ell_\delta = 1 & \textit{otherwise.} \end{cases}$$

We prove the above theorem by solving a certain type of optimization problems through the following approach. To prove that *even*[5] $\boldsymbol{L}$ minimizes secret-key sizes, we change how to sum terms as follows: For the encryption-key size, we have

$$\begin{aligned}\sum_{i=1}^{\delta}\sum_{j=0}^{\omega_i}\binom{\ell_i}{j} &= \sum_{j=1}^{\widetilde{\omega}}\sum_{i=1}^{k_j}\binom{\ell_i}{\widetilde{\omega}-(j-1)} + \sum_{i=1}^{\delta}\binom{\ell_i}{0} \\ &= \sum_{i=1}^{k_1}\binom{\ell_i}{\widetilde{\omega}} + \sum_{i=1}^{k_2}\binom{\ell_i}{\widetilde{\omega}-1} + \cdots + \sum_{i=1}^{k_{\widetilde{\omega}}}\binom{\ell_i}{1} + \sum_{i=1}^{\delta}\binom{\ell_i}{0},\end{aligned}$$

where $\widetilde{\omega} := \min\{\omega, \ell_1 - 1(=\omega_1)\}$ and $k_\alpha := \beta$ such that $\ell_\beta > \widetilde{\omega} - (\alpha-1) \geq \ell_{\beta+1}$ $(1 \leq \alpha \leq \widetilde{\omega})$. Then, we prove a lower bound for each $\sum_{i=1}^{k_j}\binom{\ell_i}{\widetilde{\omega}-(j-1)}$

[5] $\boldsymbol{L}$ is said to be *even* when $\ell_1 - \ell_\delta = 0$ if $n/\delta \in \mathbb{N}$ or $\ell_1 - \ell_\delta = 1$ if $n/\delta \notin \mathbb{N}$.

$(1 \leq j \leq \widetilde{\omega})$ in the case $k_j = \delta$ (Lemma 5.8) and in the case $k_j < \delta$ (Lemma 5.9), respectively. Specifically, in proofs of these lemmas we show that $\sum_{i=1}^{k_j} \binom{\ell_i}{\widetilde{\omega}-(j-1)}$ given by any "not even" $\boldsymbol{L}$ is always larger than or equal to that given by "even" $\boldsymbol{L}$. (To be precise, except for the case $k_1 < \delta$ (Corollary 5.1).) We also prove the case of the decryption-key sizes in a similar way (Lemma 5.10). The formal proof is as follows.

Proof of Theorem 5.9. For $\boldsymbol{L} \in \mathcal{L}(n, \delta)$, we define

$$F(\boldsymbol{L}, \omega) := \sum_{i=1}^{\delta} \sum_{j=0}^{\omega_i} \binom{\ell_i}{j} \text{ and } G(\boldsymbol{L}, \omega) := \sum_{i=1}^{\delta} \left(\ell_i \sum_{j=0}^{\omega_i} \binom{\ell_i - 1}{j} \right).$$

Obviously, $F(\boldsymbol{L}, 0) = \delta$ and $G(\boldsymbol{L}, 0) = n$ for any $\boldsymbol{L} \in \mathcal{L}(n, \delta)$.

First, we show the case of $F(\boldsymbol{L}, \omega)$ when $\omega > 0$ and $n/\delta \in \mathbb{N}$. To prove this, we show three lemmas. The first one is as follows.

Lemma 5.8. *For any $a, j \in \mathbb{N}$ and any $r \in [a]$, choose any $b_i \in \mathbb{Z}$ $(1 \leq i \leq j)$ such that $b_1 \geq \cdots \geq b_j \geq -(a - r)$, and $\sum_{i=1}^{j} b_i = 0$. Then, it holds that*

$$j \binom{a}{r} \leq \binom{a + b_1}{r} + \binom{a + b_2}{r} + \cdots + \binom{a + b_j}{r}.$$

The equality holds if and only if $r = 1$.

Proof. Let $b_1 \geq \cdots \geq b_\ell \geq 0 \geq b_{\ell+1} \geq \cdots \geq b_j > r - a$. Then, by applying Pascal's rule several times, for b_i $(1 \leq i \leq \ell)$, we have

$$\binom{a + b_i}{r} = \sum_{m=0}^{b_i - 1} \binom{a + m}{r - 1} + \binom{a}{r}. \tag{5.29}$$

On the other hand, for b_i $(\ell + 1 \leq i \leq j)$, we similarly have

$$\binom{a}{r} = \sum_{m=1}^{-b_i} \binom{a - m}{r - 1} + \binom{a + b_i}{r}.$$

Therefore, we have

$$\binom{a + b_i}{r} = \binom{a}{r} - \sum_{m=1}^{-b_i} \binom{a - m}{r - 1}. \tag{5.30}$$

From Eqs. (5.29) and (5.30), we have

$$\sum_{i=1}^{j} \binom{a + b_i}{r} = j \binom{a}{r} + \underbrace{\sum_{m=0}^{b_1 - 1} \binom{a + m}{r - 1} + \cdots + \sum_{m=0}^{b_\ell - 1} \binom{a + m}{r - 1}}_{\sum_{i=1}^{\ell} b_i \text{ terms}}$$

$$\underbrace{-\sum_{m=1}^{-b_{\ell+1}}\binom{a-m}{r-1}-\cdots-\sum_{m=1}^{-b_j}\binom{a-m}{r-1}}_{-\sum_{i=\ell+1}^{j} b_i \text{ terms}}.$$

Let $\tau := \sum_{i=1}^{\ell} b_i = -\sum_{i=\ell+1}^{j} b_i$. Then, for convenience we rewrite the above terms as follows:

$$\sum_{m=0}^{b_1-1}\binom{a+m}{r-1}+\cdots+\sum_{m=0}^{b_\ell-1}\binom{a+m}{r-1}=\sum_{m=1}^{\tau}\binom{a+\alpha_m}{r-1},$$

where $\alpha_m \in \mathbb{N}$ $(1 \le m \le \tau)$, and

$$\sum_{m=1}^{-b_{\ell+1}}\binom{a-m}{r-1}+\cdots+\sum_{m=1}^{-b_j}\binom{a-m}{r-1}=\sum_{m=1}^{\tau}\binom{a-\beta_m}{r-1},$$

where $\beta_m \in \mathbb{N}$ $(1 \le m \le \tau)$.

Then, we have

$$\sum_{i=1}^{j}\binom{a+b_i}{r}=j\binom{a}{r}+\sum_{m=1}^{\tau}\left(\binom{a+\alpha_m}{r-1}-\binom{a-\beta_m}{r-1}\right).$$

Note that for any $a, b, r \in \mathbb{N}$ such that $a \ge b \ge r$, it obviously holds $\binom{a}{r} \ge \binom{b}{r}$. Hence, it holds that $\binom{a+\alpha_m}{r-1} - \binom{a-\beta_m}{r-1} \ge 0$ $(1 \le m \le \tau)$.

Thus, it holds that $j\binom{a}{r} \le \binom{a+b_1}{r}+\binom{a+b_2}{r}+\cdots+\binom{a+b_j}{r}$ for any $a, j \in \mathbb{N}$, any $r \in [a]$, and any $b_i \in \mathbb{Z}$ such that $b_1 \ge \cdots \ge b_j \ge -(a-r)$ and $\sum_{i=1}^{j} b_i = 0$. The equality holds if and only if $r = 1$. □

We have the following corollary.

Corollary 5.1. *For any $a, j \in \mathbb{N}$, choose any $b_i \in \mathbb{Z}$ such that $b_1 \ge \cdots \ge b_k > -(a-1) = b_{k+1} = \cdots = b_j$ and $\sum_{i=1}^{j} b_i = 0$. Then, it holds that*

$$j\binom{a}{1} > \binom{a+b_1}{1}+\binom{a+b_2}{1}+\cdots+\binom{a+b_k}{1}.$$

Obviously, the value is minimized when $k = 1$, namely it holds

$$j\binom{a}{1} > \binom{a+b_1}{1} = \binom{ja-(j-1)}{1}.$$

This corollary means that if $r = 1$ and some terms are removed from the right side in Lemma 5.8, then the inequality sign is reversed.

Then, we can prove the case $\omega = 1$ from Corollary 5.1. Specifically, if $\omega = 1$, then we have

$$F(\boldsymbol{L}, 1) = \sum_{i=1}^{k} \binom{\ell_i}{1} + \sum_{i=0}^{\delta} \binom{\ell_i}{0} = \sum_{i=1}^{k} \binom{\ell_i}{1} + \delta,$$

where $\ell_1 \geq \cdots \geq \ell_k > 1 = \ell_{k+1} = \cdots = \ell_\delta$. Therefore, the minimum value of $F(\boldsymbol{L}, 1)$ is given when $\boldsymbol{L} = (n-(\delta-1), 1, \ldots, 1)$ by setting $j := \delta$ and $a := n/\delta$ in Corollary 5.1.

Next, we show the second lemma, which is a variant of Corollary 5.1. The lemma gives a contrary result to Corollary 5.1 when $r \geq 2$.

Lemma 5.9. *For any $a, j \in \mathbb{N}$ and any $r \in \{2, \ldots, a\}$, choose any $b_i \in \mathbb{Z}$ such that $b_1 \geq \cdots \geq b_k > -(a-r) \geq b_{k+1} \geq \cdots \geq b_j > -a$ and $\sum_{i=1}^{j} b_i = 0$. Then, it holds that*

$$j\binom{a}{r} < \binom{a+b_1}{r} + \binom{a+b_2}{r} + \cdots + \binom{a+b_k}{r}.$$

Further, it holds that

$$\binom{a+b_1}{r} + \binom{a+b_2}{r} + \cdots + \binom{a+b_k}{r} - j\binom{a}{r} > \sum_{m=1}^{\lambda} \binom{a+\alpha_m}{r-1},$$

where $\lambda := \sum_{i=1}^{k} b_i - (j-k)(a-r)$, and $\alpha_m \in \mathbb{N}$ $(1 \leq m \leq \lambda)$ such that $1 \leq \alpha_m \leq b_1$.

Proof. As in the proof of Lemma 5.8, let $b_1 \geq \cdots \geq b_\ell \geq 0 \geq b_{\ell+1} \geq \cdots \geq b_k > r - a \geq b_{k+1} \geq \cdots \geq b_j > -a$. Then, we can write

$$\sum_{i=1}^{k} \binom{a+b_i}{r} = \sum_{i=1}^{k} \binom{a+b_i}{r} + (j-k)\binom{r}{r} - (j-k).$$

In a similar way to Eq. (5.30), we have

$$\binom{r}{r} = \binom{a}{r} - \sum_{m=r}^{a-1} \binom{m}{r-1}. \tag{5.31}$$

From Eqs. (5.29), (5.30) and (5.31), we have

$$\begin{aligned}
&\sum_{i=1}^{k} \binom{a+b_i}{r} + (j-k)\binom{r}{r} - (j-k) \\
&= j\binom{a}{r} + \underbrace{\sum_{m=0}^{b_1-1} \binom{a+m}{r-1} + \cdots + \sum_{m=0}^{b_\ell-1} \binom{a+m}{r-1}}_{\sum_{i=1}^{\ell} b_i \text{ terms}}
\end{aligned}$$

$$\underbrace{-\sum_{m=1}^{-b_{\ell+1}}\binom{a-m}{r-1}-\cdots-\sum_{m=1}^{-b_k}\binom{a-m}{r-1}-(j-k)\sum_{m=r}^{a-1}\binom{m}{r-1}}_{-\sum_{i=\ell+1}^{k}b_i+(j-k)(a-r)\text{ terms}}$$
$$-(j-k).$$

Let $\tau := \sum_{i=1}^{\ell} b_i$ and $\eta := -\sum_{i=\ell+1}^{k} b_i + (j-k)(a-r)$. Since $(j-k)(a-1) \geq \sum_{i=1}^{k} b_i \geq (j-k)(a-r)$, it always holds $\tau \geq \eta$ and $(j-k)(r-1) \geq \tau - \eta \geq 0$. As in the proof of Lemma 5.8, for convenience we rewrite the above terms as follows:

$$\sum_{m=0}^{b_1-1}\binom{a+m}{r-1}+\cdots+\sum_{m=0}^{b_\ell-1}\binom{a+m}{r-1}=\sum_{m=1}^{\tau}\binom{a+\alpha_m}{r-1},$$

where $\alpha_m \in \mathbb{N}$ $(1 \leq m \leq \tau)$, and

$$\sum_{m=1}^{-b_{\ell+1}}\binom{a-m}{r-1}+\cdots+\sum_{m=1}^{-b_k}\binom{a-m}{r-1}+(j-k)\sum_{m=r}^{a-1}\binom{m}{r-1}=\sum_{m=1}^{\eta}\binom{a-\beta_m}{r-1},$$

where $\beta_m \in \mathbb{N}$ $(1 \leq m \leq \eta)$.

Then, we have

$$\sum_{i=1}^{k}\binom{a+b_i}{r}=j\binom{a}{r}+\sum_{m=1}^{\eta-(j-k)}\left(\binom{a+\alpha_m}{r-1}-\binom{a-\beta_m}{r-1}\right)$$
$$+\sum_{m=\eta-(j-k)+1}^{\eta}\left(\binom{a+\alpha_m}{r-1}-\binom{a-\beta_m}{r-1}-1\right)$$
$$+\sum_{m=\eta+1}^{\tau}\binom{a+\alpha_m}{r-1}.$$

Since any $(a+\alpha_m)-(a-\beta_m)=\alpha_m+\beta_m \geq 2$ and $r \geq 2$, $\binom{a+\alpha_m}{r-1}-\binom{a-\beta_m}{r-1}-1 > 0$. Therefore, $j\binom{a}{r} < \binom{a+b_1}{r}+\binom{a+b_2}{r}+\cdots+\binom{a+b_k}{r}$ for any $a, j \in \mathbb{N}$, any $r \in \{2,\ldots,a\}$, and any $b_i \in \mathbb{Z}$ such that $b_1 \geq \cdots \geq b_k \geq -(a-r) > b_{k+1} \geq \cdots \geq b_j > -a$ and $\sum_{i=1}^{j} b_i = 0$.

Further, let $\lambda := \tau - \eta = \sum_{i=1}^{k} b_i - (j-k)(a-r)$. Then, we have

$$\sum_{i=1}^{k}\binom{a+b_i}{r}-j\binom{a}{r}>\sum_{i=1}^{\lambda}\binom{a+\alpha_m}{r-1},$$

where $\alpha_m \in \mathbb{N}$ such that $1 \leq \alpha_m \leq b_1$. □

We then prove the case $\omega \geq 2$ and $n/\delta \in \mathbb{N}$. Now, let $a := n/\delta$. Then, for $\widehat{\boldsymbol{L}} = (a, \ldots, a)$, we express $F(\widehat{\boldsymbol{L}}, \omega)$ as

$$F(\widehat{\boldsymbol{L}}, \omega) = \delta\binom{a}{\hat{\omega}} + \delta\binom{a}{\hat{\omega}-1} + \cdots + \delta\binom{a}{1} + \delta\binom{a}{0},$$

where $\hat{\omega} := \min\{\omega, a-1\}$.

In the same manner, for any $\widetilde{\boldsymbol{L}} \in \mathcal{L}(n,\delta) \setminus \{\widehat{\boldsymbol{L}}\}$, $F(\widetilde{\boldsymbol{L}}, \omega)$ can be expressed as

$$F(\widetilde{\boldsymbol{L}}, \omega) = \sum_{i=1}^{k_1}\binom{\ell_i}{\tilde{\omega}} + \sum_{i=1}^{k_2}\binom{\ell_i}{\tilde{\omega}-1} + \cdots + \sum_{i=1}^{k_{\tilde{\omega}}}\binom{\ell_i}{1} + \sum_{i=1}^{\delta}\binom{\ell_i}{0},$$

where $\tilde{\omega} := \min\{\omega, \ell_1 - 1\}$, $k_m := i$ such that $\ell_i > \tilde{\omega} - (m-1) \geq \ell_{i+1}$ $(1 \leq m \leq \tilde{\omega})$. Note that $k_1 \leq \cdots \leq k_{\tilde{\omega}}$.

Then, we consider the following two cases: (i) $\hat{\omega} = \tilde{\omega}$; and (ii) $\hat{\omega} < \tilde{\omega}$. Note that the case $\hat{\omega} > \tilde{\omega}$ would never occur since $\ell_1 > a$.

We first prove the case (i). Then, we have

$$\begin{aligned}
&F(\widetilde{\boldsymbol{L}}, \omega) - F(\widehat{\boldsymbol{L}}, \omega) \\
&= \left(\sum_{i=1}^{k_1}\binom{\ell_i}{\hat{\omega}} - \delta\binom{a}{\hat{\omega}}\right) + \cdots + \left(\sum_{i=1}^{k_{\hat{\omega}}}\binom{\ell_i}{1} - \delta\binom{a}{1}\right) \\
&= \left(\sum_{i=1}^{k_1}\binom{\ell_i}{\hat{\omega}} - \delta\binom{a}{\hat{\omega}}\right) + \cdots + \left(\sum_{i=1}^{k_{\hat{\omega}}}\binom{\ell_i}{1} - \sum_{i=1}^{\delta}\binom{\ell_i}{1}\right) && (5.32) \\
&> \sum_{m=1}^{\lambda_1}\binom{a+\alpha_m^{(1)}}{\hat{\omega}-1} + \cdots + \sum_{m=1}^{\lambda_{\hat{\omega}}\ 1}\binom{a+\alpha_m^{(\hat{\omega}-1)}}{1} + \left(\sum_{i=1}^{k_{\hat{\omega}}}\binom{\ell_i}{1} - \sum_{i=1}^{\delta}\binom{\ell_i}{1}\right) && (5.33) \\
&> 0,
\end{aligned}$$

where $\lambda_i := \sum_{j=1}^{k_i}\ell_j - (\delta - k_i)(a - (\hat{\omega} - i))$, and $\alpha_m^{(k)} \in \mathbb{N}$ $(1 \leq k \leq \hat{\omega}-1)$ such that $1 \leq \alpha_m^{(k)} \leq \ell_1 - a$, Eq. (5.32) follows from Lemma 5.8, and Eq. (5.33) follows from Lemma 5.9.

In the case (ii), we have

$$\begin{aligned}
&F(\widetilde{\boldsymbol{L}}, \omega) - F(\widehat{\boldsymbol{L}}, \omega) \\
&= \sum_{i=1}^{k_1}\binom{\ell_i}{\tilde{\omega}} + \cdots + \left(\sum_{i=1}^{k_{\tilde{\omega}}\ \hat{\omega}+1}\binom{\ell_i}{\hat{\omega}} - \delta\binom{a}{\hat{\omega}}\right) + \cdots + \left(\sum_{i=1}^{k_{\tilde{\omega}}}\binom{\ell_i}{1} - \delta\binom{a}{1}\right).
\end{aligned}$$

Therefore, we can also prove $F(\widetilde{\boldsymbol{L}}, \omega) - F(\widehat{\boldsymbol{L}}, \omega) > 0$ in a similar way to the case (i). Hence, if $\omega \geq 2 \wedge n/\delta \in \mathbb{N}$, the minimum value of $F(\boldsymbol{L}, \omega)$ is given when $\ell_1 - \ell_\delta = 0$.

Similarly, we can prove the case $\omega = 1 \wedge n/\delta \notin \mathbb{N}$ and $\omega \geq 2 \wedge n/\delta \notin \mathbb{N}$, and then the minimum value of $F(\boldsymbol{L}, \omega)$ is given when $\boldsymbol{L} = (n - (\delta - 1), 1, \ldots, 1)$

and $\ell_1 - \ell_\delta = 1$, respectively, since we can express $n = \delta a + \delta_1 = \delta_1(a+1) + \delta_2 a$ for $a := \lfloor n/\delta \rfloor$, $\delta_1 := n \bmod \delta$, and $\delta_2 := \delta - \delta_1$.

We can show the case of $G(\boldsymbol{L}, \omega)$ in a similar way to the case of $F(\boldsymbol{L}, \omega)$ by the third lemma.

Lemma 5.10. *For any $a, j \in \mathbb{N}$ and any $r \in [a]$, choose any $b_i \in \mathbb{Z}$ such that $b_1 \geq \cdots \geq b_k > -(a-r) \geq b_{k+1} \geq \cdots \geq b_j > -a$ and $\sum_{i=1}^{j} b_i = 0$. Then, it holds that*

$$aj\binom{a-1}{r} < \sum_{i=1}^{k}(a+b_i)\binom{a+b_i-1}{r}.$$

Proof. As in the proof of Lemma 5.9, let $b_1 \geq \cdots \geq b_\ell \geq 0 \geq b_{\ell+1} \geq \cdots \geq b_k > r - a \geq b_{k+1} \geq \cdots \geq b_j > -a$. Let $c := a - 1$ for simplicity.

First, we consider the case $r \geq 2$. Then, we have

$$\begin{aligned}
&\sum_{i=1}^{k}(c+1+b_i)\binom{c+b_i}{r} \\
&\quad = (c+1)\left(\sum_{i=1}^{k}\binom{c+b_i}{r}\right) + \sum_{i=1}^{k} b_i\binom{c+b_i}{r} \\
&\quad > (c+1)\left(j\binom{c}{r} + \sum_{i=1}^{\lambda}\binom{c+\gamma_m}{r-1}\right) + \sum_{i=1}^{k} b_i\binom{c+b_i}{r}
\end{aligned}$$

where the last inequality follows from Lemma 5.9, $\lambda := \sum_{i=1}^{k} b_i - (j-k)(c-r)$, and $\gamma_m \in \mathbb{N}$ $(1 \leq m \leq \lambda)$.

As in the proof of Lemma 5.9, for convenience we rewrite the above terms as follows:

$$\sum_{i=1}^{\ell} b_i\binom{c+b_i}{r-1} = \sum_{m=1}^{\tau}\binom{c+\alpha_m}{r-1},$$

where $\tau := \sum_{i=1}^{\ell} b_i$, and $\alpha_m \in \mathbb{N}$ $(1 \leq m \leq \tau)$, and

$$\sum_{i=\ell+1}^{k} b_i\binom{c+b_i}{r-1} = -\sum_{m=1}^{\eta}\binom{c-\beta_m}{r-1},$$

where $\eta := -\sum_{i=\ell+1}^{k} b_i$, and $\beta_m \in \mathbb{N}$ $(1 \leq m \leq \eta)$.

Therefore, we have

$$\sum_{i=1}^{k}(c+1+b_i)\binom{c+b_i}{r}$$

$$
\begin{aligned}
&> (c+1)\left(j\binom{c}{r} + \sum_{i=1}^{\lambda}\binom{c+\gamma_m}{r-1}\right) + \sum_{m=1}^{\tau}\binom{c+\alpha_m}{r-1} - \sum_{m=1}^{\eta}\binom{c-\beta_m}{r-1} \\
&= (c+1)\left(j\binom{c}{r} + \sum_{i=1}^{\lambda}\binom{c+\gamma_m}{r-1}\right) \\
&\qquad + \left(\sum_{m=1}^{\eta}\binom{c+\alpha_m}{r-1} - \binom{c-\beta_m}{r-1}\right) + \sum_{m=\eta+1}^{\tau}\binom{c+\alpha_m}{r-1}.
\end{aligned}
$$

Since each $\binom{c+\alpha_m}{r-1} - \binom{c-\beta_m}{r-1} > 0$, it holds $aj\binom{a-1}{r} < (a+b_1)\binom{a+b_1-1}{r} + (a+b_2)\binom{a+b_2-1}{r} + \cdots + (a+b_k)\binom{a+b_k-1}{r}$ for any $a, j \in \mathbb{N}$, any $r \in \{2, \ldots, a\}$, and any $b_i \in \mathbb{Z}$ such that $b_1 \geq \cdots \geq b_k \geq -(a-r) > b_{k+1} \geq \cdots \geq b_j$ and $\sum_{i=1}^{j} b_i = 0$.

Finally, we prove the case $r = 1$. Then, we have

$$
\begin{aligned}
&\sum_{i=1}^{k}(c+1+b_i)\binom{c+b_i}{1} \\
&\quad = (c+1)\left(\sum_{i=1}^{k}\binom{c+b_i}{1}\right) + \sum_{i=1}^{k} b_i\binom{c+b_i}{1} \\
&\quad = (c+1)\left(k\binom{c}{1} + \sum_{i=1}^{k} b_i + (j-k)\binom{1}{1} - (j-k)\right) + c\sum_{i=1}^{k} b_i + \sum_{i=1}^{k} b_i^2 \\
&\quad = (c+1)j\binom{c}{1} + c^2(j-k) + \sum_{i=1}^{k} b_i^2 \qquad (5.34) \\
&\quad > (c+1)j\binom{c}{1},
\end{aligned}
$$

where Eq. (5.34) follows from Eq. (5.31) and the following fact: Since $b_{k+1} = \cdots = b_j = -c$ and $\sum_{i=1}^{j} b_i = 0$, we have $\sum_{i=1}^{k} b_i = c(j-k)$.

Therefore, it holds that $aj\binom{a-1}{1} < (a+b_1)\binom{a+b_1-1}{1} + (a+b_2)\binom{a+b_2-1}{1} + \cdots + (a+b_k)\binom{a+b_k-1}{1}$ for any $a, j \in \mathbb{N}$ and any $b_i \in \mathbb{Z}$ such that $b_1 \geq \cdots \geq b_k \geq -(a-1) > b_{k+1} \geq \cdots \geq b_j$ and $\sum_{i=1}^{j} b_i = 0$. □

Note that the above lemma holds even if $r = 1$, whereas Lemma 5.9 holds if $r \geq 2$. Hence, we can prove that if $\omega \geq 1 \wedge n/\delta \in \mathbb{N}$, then the minimum value of $G(\boldsymbol{L}, \omega)$ is given when $\ell_1 - \ell_\delta = 0$. Similarly, we can prove that if $\omega \geq 1 \wedge n/\delta \notin \mathbb{N}$, then the minimum value of $G(\boldsymbol{L}, \omega)$ is given when $\ell_1 - \ell_\delta = 1$.

Thus, the proof of Theorem 5.9 is completed. □

Hence, we can obtain the following minimal sizes of secret keys by applying the optimal parameter obtained from Theorem 5.9 to our generic construction.

Corollary 5.2. *Let Π_{BE} be an $(\leq n, \leq \omega; \delta)$-one-time secure BE scheme from our generic construction. Let $a := \lfloor n/\delta \rfloor$, $\delta_1 := n \bmod \delta$, and $\delta_2 := \delta - \delta_1$. Then, the sizes of the secret keys (in particular, decryption keys) can be minimized when we apply δ_1 $(\leq a+1, \leq \omega_1)$-KPSs $\Pi_{\mathrm{KPS}}^{(i)}$ $(1 \leq i \leq \delta_1)$ and δ_2 $(\leq a, \leq \omega_2)$-KPSs $\Pi_{\mathrm{KPS}}^{(i)}$ $(\delta_1 + 1 \leq i \leq \delta)$ to Π_{BE}, where $\omega_1 := \min\{a, \omega\}$ and $\omega_2 := \min\{a-1, \omega\}$. Namely, we have*

$$\textit{(i)}\ \log|\mathcal{EK}| = \left(\delta_1 \sum_{j=0}^{\omega_1} \binom{a+1}{j} + \delta_2 \sum_{j=0}^{\omega_2} \binom{a}{j}\right) \log q,$$

$$\textit{(ii)}\ \sum_{i=1}^{n} \log|\mathcal{DK}_i| = \left(\delta_1(a+1) \sum_{j=0}^{\omega_1} \binom{a}{j} + \delta_2 a \sum_{j=0}^{\omega_2} \binom{a-1}{j}\right) \log q.$$

As the result, the above is an upper bound on sizes of secret keys required for $(\leq n, \leq \omega; \delta)$-one-time secure BE schemes for any $\delta \in [n]$.

Remark 5.3. *As can be seen in Theorem 5.9, we cannot always minimize both of the encryption-key size and the decryption-key size for any n, ω, and δ. Specifically, the above size of encryption keys is not minimal one if $\omega = 1$, however in that case, the overhead of the encryption key is small. Therefore, in Corollary 5.2, we chose a parameter to always minimize the decryption-key size and to make the encryption-key size as small as possible. The reason why we focus specifically on the decryption-key size is that it is generally considered that the decryption-key size is more important than the encryption-key size in the context of BE schemes. Actually, most of previous works (e.g., [19, 22, 87, 92, 110]) dealt with only a lower bound on the decryption-key size.*

Remark 5.4. *The $(\leq n, \leq \omega; \delta)$-one-time secure BE scheme with the above optimal parameter includes known two constructions: When $\delta = 1$, then the above key size is equal to that of the Fiat–Naor construction; and when $\delta = n$, then the above key size is equal to that of the trivial construction. Therefore, we can say our construction is a natural extension of these known constructions.*

Chapter 6

Concluding Remarks

In this thesis, we dealt with cryptographically timed access controls. Specifically, we considered two types of them. One is so-called timed-release cryptography, where a sender can specify when receiver's functionality (e.g., decryption) is activated; and the other is so-called timed-revocable cryptography, which can inactivate receiver's functionality in the middle of the protocol.

First, we improved timed-release cryptography in the computational security setting. In addition to existing protocols such as timed-release public-key encryption (TR-PKE), we first realized TR-SS in the computational security setting (TR-CSS). Our TR-CSS scheme is almost as efficient as Krawczyk's CSS scheme, which is one of the most efficient CSS schemes, in terms of the share size. We showed that timed-release multiple encryption (TR-ME) and threshold encryption (TR-TE) can be more efficiently constructed from TR-CSS rather than from TR-PKE. We expect that our TR-CSS scheme may provide other cryptographic protocols with timed-release security such as timed-release broadcast encryption.

Second, we considered how we realized timed-release cryptography in the information-theoretic setting. We first realized information-theoretically secure cryptographic primitives with timed-release functionality, timed-release key-agreement (TR-KA), encryption (TRE), authentication codes (TRA-codes), and secret sharing (TR-SS). In each scheme, we formalized the model and security notions, derived lower bounds on sizes of secret information, and proposed the most efficient construction in the sense that it meets each inequality with equality. We can say that we finally succeeded in completing the fundamental research on information-theoretic timed-release security.

Third, we considered timed-revocable cryptography in the information-theoretic security setting. We assumed that ciphertexts are stored in cloud storage, we proposed a new concept of broadcast encryption (BE), revocable-storage broadcast encryption (RS-BE). This "revocable-storage" property provides BE with functionality suited to the cloud environment. We furthermore analyzed a trade-off between the secret-key sizes and ciphertext sizes in tradi-

tional BE schemes, and presented a generic construction of the BE scheme with general ciphertext sizes. We expected this analysis will be basis of proposals of RS-BE schemes with general ciphertext sizes.

In timed-release and timed-revocable cryptography, we assumed the existence of the time-server and storage manager, respectively.[1] More specifically, we assume that (i) the time-server broadcasts time-signals properly, though it may try to get some information on the underlying plaintext from ciphertexts, and that (ii) the storage manager updates ciphertexts properly, though the manager may try to get some information on the underlying plaintext from ciphertexts. Actually, we can regard authorities such as NICT and NIST, which broadcast time calibration signals for atomic clocks and provide network time protocol servers, as time-servers. Authorities such as NICT and NIST cannot afford to lose the public trust since those are public institutions. Therefore, it is reasonable to regard NICT and NIST as the time-servers, which may try to attack but generate and broadcast time-signals properly. The same can be said of the case of the storage manager. Namely, we can also regard providers of cloud storage services such as Dropbox and Amazon S3 as storage managers since they must want to keep user trust. However, they might maliciously update ciphertexts (i.e., modify ciphertexts) to go after profits. Therefore, we also considered such a case (i.e., a scheme secure against such a modification attack) in Section 5.4.2. Thus, we conclude that it is reasonable to consider providers of cloud data storage as the storage managers.

Essentially, the cryptographically timed access control including our proposals and previous works are realized by the following mechanisms: Timed-release cryptography realizes its functionality by (i) modifying some traditional algorithm of the basic scheme so that it cannot be executed without the additional information (i.e., time-signals) and (ii) periodically broadcasting the information by a time-server. On the other hand, timed-revocable cryptography realizes its functionality by (i') adding a new algorithm (i.e., the update algorithm) to the basic scheme (ii') and executing the algorithm by a storage manager. Hence, although periodic broadcast of time-signals and ciphertexts update in cloud storage are adopted for cryptographically timed access control, such mechanisms can be captured in other types of access control by changing (ii) and (ii'). In particular, timed-revocable cryptography (i.e., our RS-BE and RS-ABE [125]) can be seen in a broader context since other than the update algorithm, all algorithms are the same as those in the basic schemes (i.e., BE and ABE). We give some examples as follows.

Location-based access control: Consider a set of time in timed-release cryptography as a set of location-dependent information. Therefore, each time-signal are generated from location-dependent information of each fairly-limited area,

[1] Note that these entities are commonly assumed in not only our work but also the previous works (e.g., [121, 35] for timed-release cryptography and [125] for timed-revocable cryptography).

and available only at each area. A receiver can run the target algorithm only when given the information at the area specified by a sender. For example, in timed-release encryption only receivers at the area can decrypt ciphertexts.

Leakage-resilient access control: Let a user set in RS-BE be a set of time-periods, and suppose that there are a sender, a receiver who has secret keys of all time-periods, and a storage manager. The sender encrypts a plaintext with some time-period t, and stores a ciphertext in cloud storage. The storage manager updates ciphertexts to a new time-period t' if a receiver's secret key at t is exposed. As a result, the exposed secret key leaks no information on the underlying plaintexts of stored ciphertexts.

As seen above, we may consider that such a mechanism can be captured in a more general framework. Thus, it would be interesting to generalize the mechanism for realizing more new cryptographic protocols and unifying existing protocols.

Nowadays, many industrial systems are automated, and such automation has been one of the main factors in developing the modern society. The cryptographically timed access control provides automation of cryptographic protocols. We expect that our results on cryptographically timed access control will also develop the modern society and modern cryptography.

Bibliography

[1] M. Abe, R. Gennaro, K. Kurosawa, and V. Shoup. Tag-KEM/DEM: A new framework for hybrid encryption and a new analysis of kurosawa-desmedt KEM. In R. Cramer, editor, *Advances in Cryptology – EUROCRYPT 2005*, volume 3494, pages 128–146. Springer Berlin Heidelberg, 2005.

[2] S. Al-Riyami and K. Paterson. Certificateless public key cryptography. In C.-S. Laih, editor, *Advances in Cryptology – ASIACRYPT 2003*, volume 2894 of *Lecture Notes in Computer Science*, pages 452–473. Springer Berlin Heidelberg, 2003.

[3] A. Alkassar, A. Geraldy, B. Pfitzmann, and A.-R. Sadeghi. Optimized self-synchronizing mode of operation. In M. Matsui, editor, *Fast Software Encryption 2001*, volume 2355, pages 78–91. Springer Berlin Heidelberg, 2002.

[4] G. Ateniese, R. Burns, R. Curtmola, J. Herring, O. Khan, L. Kissner, Z. Peterson, and D. Song. Remote data checking using provable data possession. *ACM Trans. Inf. Syst. Secur.*, 14(1):12:1–12:34, June 2011.

[5] N. Attrapadung and H. Imai. Attribute-based encryption supporting direct/indirect revocation modes. In M. Parker, editor, *Cryptography and Coding*, volume 5921, pages 278–300. Springer Berlin Heidelberg, 2009.

[6] P. Béguin and A. Cresti. General short computational secret sharing schemes. In L. Guillou and J.-J. Quisquater, editors, *Advances in Cryptology – EUROCRYPT '95*, volume 921, pages 194–208. Springer Berlin Heidelberg, 1995.

[7] M. Bellare, A. Desai, E. Jokipii, and P. Rogaway. A concrete security treatment of symmetric encryption. In *38th Annual Symposium on Foundations of Computer Science*, pages 394–403, 1997.

[8] M. Bellare and P. Rogaway. Random oracles are practical: A paradigm for designing efficient protocols. In *Proceedings of the 1st ACM Confer-*

ence on Computer and Communications Security, CCS '93, pages 62–73, New York, NY, USA, 1993. ACM.

[9] M. Bellare and P. Rogaway. Optimal asymmetric encryption. In A. De Santis, editor, *Advances in Cryptology – EUROCRYPT'94*, volume 950, pages 92–111. Springer Berlin Heidelberg, 1995.

[10] M. Bellare and S. Shoup. Two-tier signatures, strongly unforgeable signatures, and Fiat-Shamir without random oracles. In T. Okamoto and X. Wang, editors, *Public Key Cryptography – PKC 2007*, volume 4450, pages 201–216. Springer Berlin Heidelberg, 2007.

[11] J. Benaloh and J. Leichter. Generalized secret sharing and monotone functions. In S. Goldwasser, editor, *Advances in Cryptology – CRYPTO' 88*, volume 403, pages 27–35. Springer New York, 1990.

[12] C. H. Bennett and G. Brassard. Quantum cryptography: Public key distribution and coin tossing. In *Proceedings of IEEE International Conference on Computers, Systems and Signal Processing*, volume 175. New York, 1984.

[13] K. Bentahar, P. Farshim, J. Malone-Lee, and N. Smart. Generic constructions of identity-based and certificateless KEMs. *Journal of Cryptology*, 21(2):178–199, 2008.

[14] S. Berkovits. How to broadcast a secret. In D. Davies, editor, *Advances in Cryptology – EUROCRYPT '91*, volume 547, pages 535–541. Springer Berlin Heidelberg, 1991.

[15] B. Blakley, G. Blakley, A. Chan, and J. Massey. Threshold schemes with disenrollment. In E. Brickell, editor, *Advances in Cryptology – CRYPTO' 92*, volume 740, pages 540–548. Springer Berlin Heidelberg, 1993.

[16] G. Blakley. Safeguarding cryptographic keys. In *Proceedings of the 1979 AFIPS National Computer Conference*, pages 313–317, Monval, NJ, USA, 1979. AFIPS Press.

[17] D. Bleichenbacher. Chosen ciphertext attacks against protocols based on the RSA encryption standard PKCS #1. In H. Krawczyk, editor, *Advances in Cryptology – CRYPTO'98*, volume 1462, pages 1–12. Springer Berlin Heidelberg, 1998.

[18] R. Blom. An optimal class of symmetric key generation systems. In T. Beth, N. Cot, and I. Ingemarsson, editors, *Advances in Cryptology – EUROCRYPT'84*, volume 209, pages 335–338. Springer Berlin Heidelberg, 1985.

[19] C. Blundo and A. Cresti. Space requirements for broadcast encryption. In A. Santis, editor, *Advances in Cryptology – EUROCRYPT'94*, volume 950, pages 287–298. Springer Berlin Heidelberg, 1995.

[20] C. Blundo, A. Cresti, A. Santis, and U. Vaccaro. Fully dynamic secret sharing schemes. In D. Stinson, editor, *Advances in Cryptology – CRYPTO' 93*, volume 773, pages 110–125. Springer Berlin Heidelberg, 1994.

[21] C. Blundo, A. Cresti, A. D. Santis, and U. Vaccaro. Fully dynamic secret sharing schemes. *Theoretical Computer Science*, 165(2):407–440, 1996.

[22] C. Blundo, L. Mattos, and D. Stinson. Trade-offs between communication and storage in unconditionally secure schemes for broadcast encryption and interactive key distribution. In N. Koblitz, editor, *Advances in Cryptology – CRYPTO '96*, volume 1109, pages 387–400. Springer Berlin Heidelberg, 1996.

[23] C. Blundo, A. Santis, A. Herzberg, S. Kutten, U. Vaccaro, and M. Yung. Perfectly-secure key distribution for dynamic conferences. In E. Brickell, editor, *Advances in Cryptology – CRYPTO' 92*, volume 740, pages 471–486. Springer Berlin Heidelberg, 1993.

[24] A. Boldyreva, V. Goyal, and V. Kumar. Identity-based encryption with efficient revocation. In *Proceedings of the 15th ACM conference on Computer and communications security*, CCS '08, pages 417–426, New York, NY, USA, 2008. ACM.

[25] D. Boneh and X. Boyen. Short signatures without random oracles. In C. Cachin and J. Camenisch, editors, *Advances in Cryptology – EUROCRYPT 2004*, volume 3027, pages 56–73. Springer Berlin Heidelberg, 2004.

[26] D. Boneh, R. Canetti, S. Halevi, and J. Katz. Chosen ciphertext security from identity based encryption. *SIAM Journal on Computing*, 36(5):1301–1328, 2007.

[27] D. Boneh and M. Franklin. Identity-based encryption from the Weil pairing. In J. Kilian, editor, *Advances in Cryptology – CRYPTO 2001*, volume 2139, pages 213–229. Springer Berlin Heidelberg, 2001.

[28] D. Boneh, C. Gentry, and B. Waters. Collusion resistant broadcast encryption with short ciphertexts and private keys. In V. Shoup, editor, *Advances in Cryptology – CRYPTO 2005*, volume 3621, pages 258–275. Springer Berlin Heidelberg, 2005.

[29] D. Boneh and M. Naor. Timed commitments. In M. Bellare, editor, *Advances in Cryptology – CRYPTO 2000*, volume 1880, pages 236–254. Springer Berlin Heidelberg, 2000.

[30] D. Boneh, E. Shen, and B. Waters. Strongly unforgeable signatures based on computational Diffie–Hellman. In M. Yung, Y. Dodis, A. Kiayias, and T. Malkin, editors, *Public Key Cryptography – PKC 2006*, volume 3958 of *Lecture Notes in Computer Science*, pages 229–240. Springer Berlin Heidelberg, 2006.

[31] C. Cachin. On-line secret sharing. In C. Boyd, editor, *Cryptography and Coding*, volume 1025, pages 190–198. Springer Berlin Heidelberg, 1995.

[32] D. Cash, E. Kiltz, and V. Shoup. The twin Diffie–Hellman problem and applications. In N. Smart, editor, *Advances in Cryptology – EUROCRYPT 2008*, volume 4965, pages 127–145. Springer Berlin Heidelberg, 2008.

[33] J. Cathalo, B. Libert, and J.-J. Quisquater. Efficient and non-interactive timed-release encryption. In S. Qing, W. Mao, J. López, and G. Wang, editors, *Information and Communications Security*, volume 3783, pages 291–303. Springer Berlin Heidelberg, 2005.

[34] A. Cevallos, S. Fehr, R. Ostrovsky, and Y. Rabani. Unconditionally-secure robust secret sharing with compact shares. In D. Pointcheval and T. Johansson, editors, *Advances in Cryptology – EUROCRYPT 2012*, volume 7237, pages 195–208. Springer Berlin Heidelberg, 2012.

[35] K. Chalkias, D. Hristu-Varsakelis, and G. Stephanides. Improved anonymous timed-release encryption. In J. Biskup and J. López, editors, *Computer Security – ESORICS 2007*, volume 4734, pages 311–326. Springer Berlin Heidelberg, 2007.

[36] A.-F. Chan and I. Blake. Scalable, server-passive, user-anonymous timed release cryptography. In *the 25th IEEE International Conference on Distributed Computing Systems*, ICDCS 2015, pages 504–513, 2005.

[37] H. Chen, S. Ling, C. Padró, H. Wang, and C. Xing. Key predistribution schemes and one-time broadcast encryption schemes from algebraic geometry codes. In M. Parker, editor, *Cryptography and Coding*, volume 5921, pages 263–277. Springer Berlin Heidelberg, 2009.

[38] J. Cheon, N. Hopper, Y. Kim, and I. Osipkov. Timed-release and key-insulated public key encryption. In G. Crescenzo and A. Rubin, editors, *Financial Cryptography and Data Security*, volume 4107, pages 191–205. Springer Berlin Heidelberg, 2006.

[39] J. H. Cheon, N. Hopper, Y. Kim, and I. Osipkov. Provably secure timed-release public key encryption. *ACM Trans. Inf. Syst. Secur.*, 11(2):4:1–4:44, May 2008.

[40] S. Chow, V. Roth, and E. Rieffel. General certificateless encryption and timed-release encryption. In R. Ostrovsky, R. Prisco, and I. Visconti, editors, *Security and Cryptography for Networks*, volume 5229, pages 126–143. Springer Berlin Heidelberg, 2008.

[41] T. M. Cover and J. A. Thomas. *Elements of Information Theory*. Wiley-Interscience, 2nd edition, July 2006.

[42] R. Cramer, I. Damgård, and S. Fehr. On the cost of reconstructing a secret, or VSS with optimal reconstruction phase. In J. Kilian, editor, *Advances in Cryptology – CRYPTO 2001*, volume 2139, pages 503–523. Springer Berlin Heidelberg, 2001.

[43] R. Cramer, Y. Dodis, S. Fehr, C. Padró, and D. Wichs. Detection of algebraic manipulation with applications to robust secret sharing and fuzzy extractors. In N. Smart, editor, *Advances in Cryptology – EUROCRYPT 2008*, volume 4965, pages 471–488. Springer Berlin Heidelberg, 2008.

[44] R. Cramer and V. Shoup. A practical public key cryptosystem provably secure against adaptive chosen ciphertext attack. In H. Krawczyk, editor, *Advances in Cryptology – CRYPTO '98*, volume 1462, pages 13–25. Springer Berlin Heidelberg, 1998.

[45] R. Cramer and V. Shoup. Design and analysis of practical public-key encryption schemes secure against adaptive chosen ciphertext attack. *SIAM Journal on Computing*, 33(1):167–226, 2003.

[46] Y. Cui, E. Fujisaki, G. Hanaoka, H. Imai, and R. Zhang. Formal security treatments for IBE-to-signature transformation: Relations among security notions. *IEICE Transactions*, 92-A(1):53–66, 2009.

[47] I. Damgard, S. Fehr, L. Salvail, and C. Schaffner. Cryptography in the bounded quantum-storage model. In *46th Annual IEEE Symposium on Foundations of Computer Science*, pages 449–458, Oct 2005.

[48] P. D'Arco and D. Stinson. Fault tolerant and distributed broadcast encryption. In M. Joye, editor, *Topics in Cryptology — CT-RSA 2003*, volume 2612, pages 263–280. Springer Berlin Heidelberg, 2003.

[49] A. Dent and Q. Tang. Revisiting the security model for timed-release encryption with pre-open capability. In J. Garay, A. Lenstra, M. Mambo, and R. Peralta, editors, *Information Security*, volume 4779, pages 158–174. Springer Berlin Heidelberg, 2007.

[50] Y. Desmedt and Y. Frankel. Threshold cryptosystems. In G. Brassard, editor, *Advances in Cryptology – CRYPTO' 89*, volume 435, pages 307–315. Springer New York, 1990.

[51] G. Di Crescenzo, R. Ostrovsky, and S. Rajagopalan. Conditional oblivious transfer and timed-release encryption. In J. Stern, editor, *Advances in Cryptology – EUROCRYPT'99*, volume 1592, pages 74–89. Springer Berlin Heidelberg, 1999.

[52] W. Diffie and M. Hellman. New directions in cryptography. *IEEE Transactions on Information Theory*, 22(6):644–654, 1976.

[53] Y. Dodis and N. Fazio. Public key broadcast encryption for stateless receivers. In J. Feigenbaum, editor, *Digital Rights Management*, volume 2696, pages 61–80. Springer Berlin Heidelberg, 2003.

[54] Y. Dodis and J. Katz. Chosen-ciphertext security of multiple encryption. In J. Kilian, editor, *Theory of Cryptography*, volume 3378, pages 188–209. Springer Berlin Heidelberg, 2005.

[55] Y. Dodis, J. Katz, S. Xu, and M. Yung. Key-insulated public key cryptosystems. In L. Knudsen, editor, *Advances in Cryptology – EUROCRYPT 2002*, volume 2332, pages 65–82. Springer Berlin Heidelberg, 2002.

[56] Y. Dodis and D. Yum. Time capsule signature. In A. Patrick and M. Yung, editors, *Financial Cryptography and Data Security*, volume 3570, pages 57–71. Springer Berlin Heidelberg, 2005.

[57] D. Dolev, C. Dwork, and M. Naor. Non-malleable cryptography. In *Proceedings of the Twenty-third Annual ACM Symposium on Theory of Computing*, STOC '91, pages 542–552, New York, NY, USA, 1991. ACM.

[58] D. Dolev, C. Dwork, and M. Naor. Non-malleable cryptography. In *SIAM Journal on Computing*, pages 542–552, 1998.

[59] S. Dziembowski and U. Maurer. On generating the initial key in the bounded-storage model. In C. Cachin and J. Camenisch, editors, *Advances in Cryptology – EUROCRYPT 2004*, volume 3027 of *Lecture Notes in Computer Science*, pages 126–137. Springer Berlin Heidelberg, 2004.

[60] T. ElGamal. A public key cryptosystem and a signature scheme based on discrete logarithms. In G. Blakley and D. Chaum, editors, *Advances in Cryptology – CRYPTO'84*, volume 196, pages 10–18. Springer Berlin Heidelberg, 1985.

[61] A. Fiat and M. Naor. Broadcast encryption. In D. Stinson, editor, *Advances in Cryptology – CRYPTO' 93*, volume 773, pages 480–491. Springer Berlin Heidelberg, 1994.

[62] A. Fujioka, Y. Okamoto, and T. Saito. Generic construction of strongly secure timed-release public-key encryption. *IEICE Transactions*, 96-A(1):76–91, 2013.

[63] E. Fujisaki and T. Okamoto. A practical and provably secure scheme for publicly verifiable secret sharing and its applications. In K. Nyberg, editor, *Advances in Cryptology – EUROCRYPT'98*, volume 1403, pages 32–46. Springer Berlin Heidelberg, 1998.

[64] J. Garay and M. Jakobsson. Timed release of standard digital signatures. In M. Blaze, editor, *Financial Cryptography*, volume 2357, pages 168–182. Springer Berlin Heidelberg, 2003.

[65] J. Garay and C. Pomerance. Timed fair exchange of standard signatures. In R. Wright, editor, *Financial Cryptography*, volume 2742, pages 190–207. Springer Berlin Heidelberg, 2003.

[66] J. Garay, J. Staddon, and A. Wool. Long-lived broadcast encryption. In M. Bellare, editor, *Advances in Cryptology – CRYPTO 2000*, volume 1880, pages 333–352. Springer Berlin Heidelberg, 2000.

[67] R. Gennaro, S. Halevi, and T. Rabin. Secure hash-and-sign signatures without the random oracle. In J. Stern, editor, *Advances in Cryptology – EUROCRYPT'99*, volume 1592, pages 123–139. Springer Berlin Heidelberg, 1999.

[68] C. Gentry and B. Waters. Adaptive security in broadcast encryption systems (with short ciphertexts). In A. Joux, editor, *Advances in Cryptology – EUROCRYPT 2009*, volume 5479, pages 171–188. Springer Berlin Heidelberg, 2009.

[69] E. N. Gilbert, F. J. MacWilliams, and N. J. A. Sloane. Codes which detect deception. *Bell System Technical Journal*, 53(3):405–424, 1974.

[70] S. Goldwasser and S. Micali. Probabilistic encryption. *Journal of Computer and System Sciences*, 28(2):270 – 299, 1984.

[71] S. Goldwasser, S. Micali, and R. L. Rivest. A digital signature scheme secure against adaptive chosen-message attacks. *SIAM Journal on Computing*, 17(2):281–308, 1988.

[72] S. Halevi, D. Harnik, B. Pinkas, and A. Shulman-Peleg. Proofs of ownership in remote storage systems. In *Proceedings of the 18th ACM conference on Computer and communications security*, CCS '11, pages 491–500, New York, NY, USA, 2011. ACM.

[73] G. Hanaoka and K. Kurosawa. Efficient chosen ciphertext secure public key encryption under the computational Diffie-Hellman assumption. In J. Pieprzyk, editor, *Advances in Cryptology – ASIACRYPT 2008*, volume 5350, pages 308–325. Springer Berlin Heidelberg, 2008.

[74] D. Harnik and M. Naor. On everlasting security in the hybrid bounded storage model. In M. Bugliesi, B. Preneel, V. Sassone, and I. Wegener, editors, *Automata, Languages and Programming*, volume 4052, pages 192–203. Springer Berlin Heidelberg, 2006.

[75] A. Herzberg, S. Jarecki, H. Krawczyk, and M. Yung. Proactive secret sharing or: How to cope with perpetual leakage. In D. Coppersmith, editor, *Advances in Cryptology – CRYPTO' 95*, volume 963, pages 339–352. Springer Berlin Heidelberg, 1995.

[76] D. Hofheinz, T. Jager, and E. Kiltz. Short signatures from weaker assumptions. In D. Lee and X. Wang, editors, *Advances in Cryptology – ASIACRYPT 2011*, volume 7073, pages 647–666. Springer Berlin Heidelberg, 2011.

[77] D. Hofheinz and E. Kiltz. Practical chosen ciphertext secure encryption from factoring. In A. Joux, editor, *Advances in Cryptology – EUROCRYPT 2009*, volume 5479, pages 313–332. Springer Berlin Heidelberg, 2009.

[78] M. Ito, A. Saito, and T. Nishizeki. Secret sharing scheme realizing general access structure. In *IEEE Globecom'87*, pages 99–102, 1987.

[79] M. Ito, A. Saito, and T. Nishizeki. Secret sharing scheme realizing general access structure. *Electronics and Communications in Japan (Part III: Fundamental Electronic Science)*, 72(9):56–64, 1989.

[80] M. Jhanwar and R. Safavi-Naini. Unconditionally-secure robust secret sharing with minimum share size. In A.-R. Sadeghi, editor, *Financial Cryptography and Data Security*, volume 7859, pages 96–110. Springer Berlin Heidelberg, 2013.

[81] S. Kamara and K. Lauter. Cryptographic cloud storage. In R. Sion, R. Curtmola, S. Dietrich, A. Kiayias, J. Miret, K. Sako, and F. Sebé, editors, *Financial Cryptography and Data Security*, volume 6054, pages 136–149. Springer Berlin Heidelberg, 2010.

[82] E. Karnin, J. Greene, and M. Hellman. On secret sharing systems. *IEEE Transactions on Information Theory*, 29(1):35–41, 1983.

[83] R. Kikuchi, A. Fujioka, Y. Okamoto, and T. Saito. Strong security notions for timed-release public-key encryption revisited. In H. Kim,

editor, *Information Security and Cryptology - ICISC 2011*, volume 7259, pages 88–108. Springer Berlin Heidelberg, 2012.

[84] E. Kiltz and D. Galindo. Direct chosen-ciphertext secure identity-based key encapsulation without random oracles. In L. Batten and R. Safavi-Naini, editors, *Information Security and Privacy*, volume 4058, pages 336–347. Springer Berlin Heidelberg, 2006.

[85] E. Kiltz and D. Galindo. Direct chosen-ciphertext secure identity-based key encapsulation without random oracles. *Theoretical Computer Science*, 410(47–49):5093–5111, 2009.

[86] H. Krawczyk. Secret sharing made short. In D. Stinson, editor, *Advances in Cryptology – CRYPTO' 93*, volume 773, pages 136–146. Springer Berlin Heidelberg, 1994.

[87] K. Kurosawa, T. Yoshida, Y. Desmedt, and M. Burmester. Some bounds and a construction for secure broadcast encryption. In K. Ohta and D. Pei, editors, *Advances in Cryptology – ASIACRYPT'98*, volume 1514, pages 420–433. Springer Berlin Heidelberg, 1998.

[88] L. Lamport. Constructing digital signatures from a one-way function. Csl-98, SRI International, Palo Alto, Oct. 1979.

[89] B. Libert and J.-J. Quisquater. Practical time capsule signatures in the standard model from bilinear maps. In T. Takagi, T. Okamoto, E. Okamoto, and T. Okamoto, editors, *Pairing-Based Cryptography – Pairing 2007*, volume 4575, pages 23–38. Springer Berlin Heidelberg, 2007.

[90] J. Liu, H. Wang, M. Xian, and K. Huang. A secure and efficient scheme for cloud storage against eavesdropper. In S. Qing, J. Zhou, and D. Liu, editors, *Information and Communications Security*, volume 8233, pages 75–89. Springer International Publishing, 2013.

[91] Z. Liu, J. Li, X. Chen, J. Yang, and C. Jia. TMDS: Thin-model data sharing scheme supporting keyword search in cloud storage. In W. Susilo and Y. Mu, editors, *Information Security and Privacy*, volume 8544, pages 115–130. Springer International Publishing, 2014.

[92] M. Luby and J. Staddon. Combinatorial bounds for broadcast encryption. In K. Nyberg, editor, *Advances in Cryptology – EUROCRYPT'98*, volume 1403, pages 512–526. Springer Berlin Heidelberg, 1998.

[93] W. Mao. Timed-release cryptography. In *Selected Areas in Cryptography VIII (SAC'01)*, pages 342–357. Prentice Hall, 2001.

[94] K. Martin, J. Pieprzyk, R. Safavi-Naini, and H. Wang. Changing thresholds in the absence of secure channels. In J. Pieprzyk, R. Safavi-Naini, and J. Seberry, editors, *Information Security and Privacy*, volume 1587, pages 177–191. Springer Berlin Heidelberg, 1999.

[95] K. M. Martin, R. Safavi-Naini, and H. Wang. Bounds and techniques for efficient redistribution of secret shares to new access structures. *The Computer Journal*, 42(8):638–649, 1999.

[96] T. Matsuda, Y. Nakai, and K. Matsuura. Efficient generic constructions of timed-release encryption with pre-open capability. In M. Joye, A. Miyaji, and A. Otsuka, editors, *Pairing-Based Cryptography - Pairing 2010*, volume 6487, pages 225–245. Springer Berlin Heidelberg, 2010.

[97] T. Matsumoto and H. Imai. On the key predistribution system: A practical solution to the key distribution problem. In C. Pomerance, editor, *Advances in Cryptology – CRYPTO '87*, volume 293, pages 185–193. Springer Berlin Heidelberg, 1988.

[98] U. Maurer. Conditionally-perfect secrecy and a provably-secure randomized cipher. *Journal of Cryptology*, 5(1):53–66, 1992.

[99] U. Maurer. Secret key agreement by public discussion from common information. *IEEE Transactions on Information Theory*, 39(3):733–742, 1993.

[100] U. Maurer and P. Schmid. A calculus for secure channel establishment in open networks. In D. Gollmann, editor, *Computer Security – ESORICS 94*, volume 875, pages 173–192. Springer Berlin Heidelberg, 1994.

[101] U. M. Maurer and P. E. Schmid. A calculus for security boots trapping in distributed systems. *Journal of Computer Security*, 4(1):55–80, Jan. 1996.

[102] T. May. Timed-release crypto. manuscript, 1993.

[103] R. C. Merkle and M. E. Hellman. On the security of multiple encryption. *Commun. ACM*, 24(7):465–467, July 1981.

[104] Y. Nakai, T. Matsuda, W. Kitada, and K. Matsuura. A generic construction of timed-release encryption with pre-open capability. In T. Takagi and M. Mambo, editors, *Advances in Information and Computer Security*, volume 5824, pages 53–70. Springer Berlin Heidelberg, 2009.

[105] D. Naor, M. Naor, and J. Lotspiech. Revocation and tracing schemes for stateless receivers. In J. Kilian, editor, *Advances in Cryptology – CRYPTO 2001*, volume 2139, pages 41–62. Springer Berlin Heidelberg, 2001.

[106] M. Naor, G. Segev, and A. Smith. Tight bounds for unconditional authentication protocols in the manual channel and shared key models. In C. Dwork, editor, *Advances in Cryptology – CRYPTO 2006*, volume 4117, pages 214–231. Springer Berlin Heidelberg, 2006.

[107] M. Naor, G. Segev, and A. Smith. Tight bounds for unconditional authentication protocols in the manual channel and shared key models. *IEEE Transactions on Information Theory*, 54(6):2408–2425, 2008.

[108] M. Naor and M. Yung. Public-key cryptosystems provably secure against chosen ciphertext attacks. In *Proceedings of the Twenty-second Annual ACM Symposium on Theory of Computing*, STOC '90, pages 427–437, New York, NY, USA, 1990. ACM.

[109] T. Okamoto and D. Pointcheval. REACT: Rapid enhanced-security asymmetric cryptosystem transform. In D. Naccache, editor, *Topics in Cryptology – CT-RSA 2001*, volume 2020, pages 159–174. Springer Berlin Heidelberg, 2001.

[110] C. Padró, I. Gracia, and S. Martín. Improving the trade-off between storage and communication in broadcast encryption schemes. *Discrete Applied Mathematics*, 143(1-3):213–220, 2004.

[111] C. Padró, I. Gracia, S. Martín, and P. Morillo. Linear broadcast encryption schemes. *Discrete Applied Mathematics*, 128(1):223–238, 2003.

[112] P. Paillier. Public-key cryptosystems based on composite degree residuosity classes. In J. Stern, editor, *Advances in Cryptology – EUROCRYPT'99*, volume 1592, pages 223–238. Springer Berlin Heidelberg, 1999.

[113] K. Paterson and E. Quaglia. Time-specific encryption. In J. Garay and R. Prisco, editors, *Security and Cryptography for Networks*, volume 6280, pages 1–16. Springer Berlin Heidelberg, 2010.

[114] D. Phan, D. Pointcheval, and M. Strefler. Security notions for broadcast encryption. In J. Lopez and G. Tsudik, editors, *Applied Cryptography and Network Security*, volume 6715, pages 377–394. Springer Berlin Heidelberg, 2011.

[115] M. Rabin. The information dispersal algorithm and its applications. In R. Capocelli, editor, *Sequences*, pages 406–419. Springer New York, 1990.

[116] M. O. Rabin. Digitalized signatures and public-key functions as intractable as factorization. Technical report, Massachusetts Institute of Technology Laboratory for Computer Science, Cambridge, MA, USA, 1979.

[117] M. O. Rabin. Efficient dispersal of information for security, load balancing, and fault tolerance. *J. ACM*, 36(2):335–348, Apr. 1989.

[118] T. Rabin and M. Ben-Or. Verifiable secret sharing and multiparty protocols with honest majority. In *Proceedings of the twenty-first annual ACM symposium on Theory of computing*, STOC '89, pages 73–85, New York, NY, USA, 1989. ACM.

[119] C. Rackoff and D. Simon. Non-interactive zero-knowledge proof of knowledge and chosen ciphertext attack. In J. Feigenbaum, editor, *Advances in Cryptology – CRYPTO'91*, volume 576, pages 433–444. Springer Berlin Heidelberg, 1992.

[120] R. L. Rivest. Unconditionally secure commitment and oblivious transfer schemes using private channels and a trusted initializer. manuscript, 1999.

[121] R. L. Rivest, A. Shamir, and D. A. Wagner. Time-lock puzzles and timed-release crypto. Technical Report Technical memo MIT/LCS/TR-684, MIT Laboratory for Computer Science, 1996. (Revision 3/10/96).

[122] P. Rogaway and M. Bellare. Robust computational secret sharing and a unified account of classical secret-sharing goals. In *Proceedings of the 14th ACM Conference on Computer and Communications Security*, CCS '07, pages 172–184, New York, NY, USA, 2007. ACM.

[123] R. Safavi-Naini and S. Jiang. Unconditionally secure conference key distribution: Security notions, bounds and constructions. *International Journal of Foundations of Computer Science*, 22(06):1369–1393, 2011.

[124] R. Safavi-Naini and H. Wang. Multireceiver authentication codes: Models, bounds, constructions and extensions. *Information and Computation*, 151:148–172, 1998.

[125] A. Sahai, H. Seyalioglu, and B. Waters. Dynamic credentials and ciphertext delegation for attribute-based encryption. In R. Safavi-Naini and R. Canetti, editors, *Advances in Cryptology – CRYPTO 2012*, volume 7417, pages 199–217. Springer Berlin Heidelberg, 2012.

[126] H. Shacham and B. Waters. Compact proofs of retrievability. *Journal of Cryptology*, 26(3):442–483, 2013.

[127] A. Shamir. How to share a secret. *Commun. ACM*, 22(11):612–613, Nov. 1979.

[128] C. E. Shannon. A mathematical theory of communication. *The Bell System Technical Journal*, 27:379–423, 623–656, July, October 1948.

[129] C. E. Shannon. Communication theory of secrecy systems. *The Bell System Technical Journal*, 28:656–715, October 1949.

[130] J. Shikata. Trends and development of information-theoretic cryptography. *IEICE Transactions*, 98-A(1):16–25, 2015.

[131] V. Shoup. A proposal for an ISO standard for public key encryption. Cryptology ePrint Archive, Report 2001/112, 2001. `http://eprint.iacr.org/`.

[132] G. Simmons. Authentication theory/coding theory. In G. Blakley and D. Chaum, editors, *Advances in Cryptology – CRYPTO'84*, volume 196, pages 411–431. Springer Berlin Heidelberg, 1985.

[133] M. Soete, K. Vedder, and M. Walker. Cartesian authentication schemes. In J.-J. Quisquater and J. Vandewalle, editors, *Advances in Cryptology – EUROCRYPT '89*, volume 434, pages 476–490. Springer Berlin Heidelberg, 1990.

[134] J. Stanek, A. Sorniotti, E. Androulaki, and L. Kencl. A secure data deduplication scheme for cloud storage. In N. Christin and R. Safavi-Naini, editors, *Financial Cryptography and Data Security*, Lecture Notes in Computer Science, pages 99–118. Springer Berlin Heidelberg, 2014.

[135] D. Stinson. On some methods for unconditionally secure key distribution and broadcast encryption. *Designs, Codes and Cryptography*, 12(3):215–243, 1997.

[136] J. Sung, S. Lee, J. Lim, W. Lee, and O. Yi. Concrete security analysis of CTR-OFB and CTR-CFB modes of operation. In K. Kim, editor, *Information Security and Cryptology – ICISC 2001*, volume 2288, pages 103–113. Springer Berlin Heidelberg, 2002.

[137] K. Tochikubo, T. Uyematsu, and R. Matsumoto. Efficient secret sharing schemes based on authorized subsets. *IEICE Transactions*, 88-A(1):322–326, 2005.

[138] R. Tonicelli, A. Nascimento, R. Dowsley, J. Műller-Quade, H. Imai, G. Hanaoka, and A. Otsuka. Information-theoretically secure oblivious polynomial evaluation in the commodity-based model. *International Journal of Information Security*, 14(1):73–84, 2015.

[139] D. Tonien, R. Safavi-Naini, P. Nickolas, and Y. Desmedt. Unconditionally secure approximate message authentication. In Y. Chee, C. Li, S. Ling, H. Wang, and C. Xing, editors, *Coding and Cryptology*, volume 5557, pages 233–247. Springer Berlin Heidelberg, 2009.

[140] G. Vernam. Cipher printing telegraph systems: For secret wire and radio telegraphic communications. *Journal of the A.I.E.E.*, 45(2):109–115, Feb 1926.

[141] Y. Watanabe, T. Seito, and J. Shikata. Information-theoretic timed-release security: Key-agreement, encryption, and authentication codes. In A. Smith, editor, *Information Theoretic Security*, volume 7412, pages 167–186. Springer Berlin Heidelberg, 2012.

[142] Y. Watanabe and J. Shikata. Timed-release computational secret sharing scheme and its applications. In S. Chow, J. Liu, L. Hui, and S. Yiu, editors, *Provable Security*, volume 8782, pages 326–333. Springer International Publishing, 2014.

[143] Y. Watanabe and J. Shikata. Constructions of unconditionally secure broadcast encryption from key predistribution systems with trade-offs between communication and storage. In M.-H. Au and A. Miyaji, editors, *Provable Security*, volume 9451, pages 489–502. Springer International Publishing, 2015.

[144] Y. Watanabe and J. Shikata. Timed-release secret sharing schemes with information theoretic security. In B. Ors and B. Preneel, editors, *Cryptography and Information Security in the Balkans*, volume 9024, pages 219–236. Springer International Publishing, 2015.

[145] Y. Watanabe and J. Shikata. Unconditionally secure broadcast encryption schemes with trade-offs between communication and storage. *IEICE Transactions*, 99-A(6), 2016. (to appear).

[146] B. Waters. Efficient identity-based encryption without random oracles. In R. Cramer, editor, *Advances in Cryptology – EUROCRYPT 2005*, volume 3494, pages 114–127. Springer Berlin Heidelberg, 2005.

[147] A. D. Wyner. The wire-tap channel. *Bell System Technical Journal*, 54:1355–1387, 1975.

[148] K. Yang, X. Jia, and K. Ren. Attribute-based fine-grained access control with efficient revocation in cloud storage systems. In *Proceedings of the 8th ACM SIGSAC Symposium on Information, Computer and Communications Security*, ASIA CCS '13, pages 523–528, New York, NY, USA, 2013. ACM.

[149] R. Zhang, G. Hanaoka, J. Shikata, and H. Imai. On the security of multiple encryption or CCA-security+CCA-security=CCA-security? In F. Bao, R. Deng, and J. Zhou, editors, *Public Key Cryptography – PKC 2004*, volume 2947, pages 360–374. Springer Berlin Heidelberg, 2004.

List of Publications

Peer-Reviewed Journal Article and Conference Papers

Related to The Thesis:

1. Y. Watanabe and J. Shikata, "Unconditionally Secure Broadcast Encryption Schemes with Trade-offs between Communication and Storage," IEICE Transactions, vol.99-A, no.6, 2016 (see also [145]). (to appear)
2. Y. Watanabe and J. Shikata, "Constructions of Unconditionally Secure Broadcast Encryption from Key Predistribution Systems with Trade-offs between Communication and Storage," Provable Security, LNCS 9451, pp.489–502, Springer, 2015 (see also [143]).
3. Y. Watanabe and J. Shikata, "Timed-Release Secret Sharing Schemes with Information Theoretic Security," Cryptography and Information Security in the Balkans, LNCS 9024, pp.219–236, Springer, 2014 (see also [144]).
4. Y. Watanabe and J. Shikata, "Timed-Release Computational Secret Sharing Scheme and Its Applications," Provable Security, LNCS 8782, pp.326–333, Springer, 2014 (see also [142]).
5. Y. Watanabe, T. Seito, and J. Shikata, "Information-Theoretic Timed-Release Security: Key-Agreement, Encryption and Authentication Codes" Information Theoretic Security, LNCS 7412, pp.167–186, Springer, 2012 (see also [141]).

Other Publications:

6. T. Shinichiro, Y. Watanabe, and J. Shikata, "Sequential Aggregate Authentication Codes with Information Theoretic Security," 50th Annual Conference on Information Sciences and Systems (CISS 2016), 2016. (to appear)
7. Y. Watanabe and J. Shikata, "Identity-based Hierarchical Key-insulated Encryption without Random Oracles," Public-Key Cryptography – PKC 2016, LNCS 9614, Springer, 2016. (to appear)

8. K. Emura, L. T. Phong, and Y. Watanabe, "Keyword Revocable Searchable Encryption with Trapdoor Exposure Resistance and Re-generateability, " 2015 IEEE Trustcom/BigDataSE/ISPA, vol.1, pp.167–174, 2015.

9. Y. Ishida, Y. Watanabe, and J. Shikata, "Constructions of CCA-secure Revocable Identity-based Encryption," Information Security and Privacy, LNCS 9144, pp.174–191, Springer, 2015.

10. N. Takei, Y. Watanabe, and J. Shikata, "Information-Theoretically Secure Blind Authentication Codes without Verifier's Secret Keys," Josai Mathematical Monograph 8, pp.115–133, Graduate School of Sciences, Josai University, 2015.

11. T. Seito, Y. Watanabe, K. Kinose, and J. Shikata, "Information-Theoretically Secure Anonymous Group Authentication with Arbitration: Formal Definition and Construction," Josai Mathematical Monograph 7, pp.85–110, Graduate School of Sciences, Josai University, 2014.

12. S. Hajime, Y. Watanabe, and J. Shikata, " Information-Theoretically Secure Entity Authentication in the Multi-user Setting, " Information Security and Cryptology, LNCS 8565, pp.400–417, Springer, 2013.

13. N. Takei, Y. Watanabe, and J. Shikata, " Unconditionally Secure Blind Authentication Codes in the Manual Channel Model, " Proc. of The 3rd International Symposium on Engineering, Energy and Environment (3rd ISEEE), pp.297–302, 2013.

14. T. Seito, Y. Watanabe, K. Kinose, and J. Shikata, " Unconditionally Secure Anonymous Group Authentication with an Arbiter," Proc. of The 3rd International Symposium on Engineering, Energy and Environment (3rd ISEEE), pp.291–296, 2013.

15. A. Kubai, J. Shikata, and Y. Watanabe, " Information-Theoretically Secure Aggregate Authentication Code: Model, Bounds, and Constructions, " Security Engineering and Intelligence Informatics, LNCS 8128, pp.16–28, Springer, 2013.

Non Peer-Reviewed Conference Papers

Related to The Thesis:

16. Y. Watanabe and J. Shikata, "Unconditionally Secure Broadcast Encryption Schemes with Trade-offs between Communication and Storage," Proc. of Computer Security Symposium 2015 (CSS 2015), 2015 (in Japanese).

17. Y. Watanabe, G. Hanaoka, and J. Shikata, "Unconditionally Secure Robust Revocable-Storage Broadcast Encryption," Proc. of The 32nd Symposium on Cryptography and Information Security (SCIS 2015), 2D1-2, 2015 (in Japanese). [**SCIS Award**]

18. Y. Watanabe and J. Shikata, "Information-Theoretically Secure Revocable-Storage Broadcast Encryption," Proc. of Computer Security Symposium 2014 (CSS 2014), 3E1-1, 2014 (in Japanese). [**CSS Student Paper Prize**]

19. Y. Watanabe and J. Shikata, "Timed-Release Secret Sharing Scheme with Computatinal Security," Proc. of The 31st Symposium on Cryptography and Information Security (SCIS 2014), 3F1-5, 2014 (in Japanese).

20. Y. Watanabe and J. Shikata, "Information-Theoretically Secure Timed-Release Secret Sharing Schemes," Proc. of Computer Security Symposium 2013 (CSS 2013), 2C2-4, 2013 (in Japanese).

21. Y. Watanabe, T. Seito, and J. Shikata, "Tight Lower Bounds in Information-Theoretically Secure Timed-Release Encryption and Authentication Codes," Proc. of Computer Security Symposium 2012 (CSS 2012), 2C4-2, 2012 (in Japanese).

22. Y. Watanabe, T. Seito, and J. Shikata, "Applications of Information-Theoretically Secure Timed-Release Key-Agreement," Proc. of The 29th Symposium on Cryptography and Information Security (SCIS 2012), 4B2-2, 2012 (in Japanese).

23. Y. Watanabe, T. Seito, and J. Shikata, "Information-Theoretically Secure Timed-Release Key-Agreement," Proc. of Computer Security Symposium 2011 (CSS 2011), 3C3-2, 2011 (in Japanese).

Other Publications:

24. Y. Watanabe and J. Shikata, "Identity-based Encryption with Hierarchical Key-insulation in the Standard Model," Proc. of The 33rd Symposium on Cryptography and Information Security (SCIS 2016), 2E3-2, 2016 (in Japanese).

25. J. Ida, Y. Watanabe, and J. Shikata, "Security Notions and Constructions of Interactive Signcryption in the Multi-User Setting," Proc. of The 33rd Symposium on Cryptography and Information Security (SCIS 2016), 2C3-3, 2016 (in Japanese).

26. T. Yoshizawa, Y. Watanabe, and J. Shikata, "A General Model and Construction of Unconditionally Secure Searchable Encryption," Proc. of The 33rd Symposium on Cryptography and Information Security (SCIS 2016), 2C2-1, 2016 (in Japanese).

27. J. Ida, Y. Watanabe, and J. Shikata, "Interactive Signcryption," Proc. of Computer Security Symposium 2015 (CSS 2015), 2C3-3, 2015 (in Japanese).

28. T. Yoshizawa, Y. Watanabe, and J. Shikata, "Information-Theoretically Secure Searchable Encryption," Proc. of Computer Security Symposium 2015 (CSS 2015), 3C3-4, 2015 (in Japanese).

29. Y. Ishida, Y. Watanabe, and J. Shikata, "CCA-secure Revocable Identity-based Encryption with Short Ciphertext-size," Proc. of The 32nd Symposium on Cryptography and Information Security (SCIS 2015), 2D3-4, 2015 (in Japanese).

30. M. Kasai, T. Seito, Y. Watanabe, and J. Shikata, "A Generic Construction of Proxy Re-Encryption by Canetti–Halevi–Katz Transform," Proc. of The 32nd Symposium on Cryptography and Information Security (SCIS 2015), 2F2-3, 2015 (in Japanese).

31. S. Tomita, Y. Watanabe, and J. Shikata, "Sequential Multi-Authentication Code with Information Theoretic Security," Proc. of The 32nd Symposium on Cryptography and Information Security (SCIS 2015), 2D1-3, 2015 (in Japanese).

32. Y. Ishida, Y. Watanabe, and J. Shikata, "Revocable Identity-based Encryption Secure against Chosen Ciphertext Attack," Proc. of Computer Security Symposium 2014 (CSS 2014), 1E4-4, 2014 (in Japanese).

33. N. Takei, Y. Watanabe, and J. Shikata, "Unconditionally Secure Blind Authentication Codes without Verifier's Secret Keys," Proc. of Computer Security Symposium 2013 (CSS 2013), 2C3-3, 2013 (in Japanese).

34. S. Hajime, Y. Watanabe, and J. Shikata, "Unconditionally Secure Entity Authentication in a Group," Proc. of Computer Security Symposium 2012 (CSS 2012), 2C4-1, 2012 (in Japanese).

35. T. Seito, Y. Watanabe, and J. Shikata, "Relationships between Key-Insulated and Timed-Release Key-Agreement with Information-Theoretic Security," Proc. of The 29th Symposium on Cryptography and Information Security (SCIS 2012), 4B2-1, 2012 (in Japanese).

Non Peer-Reviewed Posters

36. Y. Watanabe, G. Hanaoka, and J. Shikata, "How to Provide Long-term Security and Required Functionality for Cloud Storage," Yokohama Environment and Information Sciences (YEIS) International Forum, Yokohama, Japan, 2015.

37. Y. Ishida, Y. Watanabe, and J. Shikata, "Constructions of Strongly Secure Revocable Identity-based Encryption," Yokohama Environment and Information Sciences (YEIS) International Forum, Yokohama, Japan, 2015.

38. Y. Watanabe, G. Hanaoka, and J. Shikata, "How to Provide Long-term Security and Required Functionality for Cloud Storage," Privacy-aware Computational Genomics 2015 (PRIVAGEN 2015), Japan, 2015.

39. Y. Watanabe and J. Shikata, "Information-Theoretically Secure Revocable-Storage Broadcast Encryption," the 9th International Workshop on Security (IWSEC 2014), Japan, 2014. [**Best Poster Award**]

Invited Talks

40. "Unconditionally Secure Revocable Storage," the 10th International Workshop on Security (IWSEC 2015), Japan, 2015.

41. "Timed-Release Cryptography -Two Theoretical Approaches to Achieve Security," JSPS-DST Asian Academic Seminar 2013 (AAS 2013), Japan, 2013.

Invited Paper

42. 四方順司, 渡邉洋平, "情報理論的暗号技術について," 情報処理, Vol. 55, No. 3, pp.260–267, 2014 年 3 月号, 情報処理学会, 2014 (in Japanese).

Awards and Honers

- **JSPS Research Fellowship for Young Scientists (PD)**, 2016–2019.
- **SCIS Paper Award** for 17 at SCIS 2015, 2016.
- **CSS Student Paper Prize** for 18 at CSS 2014, 2014.
- **Best Poster Award** for 39 at IWSEC 2014, 2014.
- **JSPS Research Fellowship for Young Scientists (DC1)**, 2013–2016.