Timed-Release Secret Sharing Scheme with Information Theoretic Security

Yohei Watanabe
Graduate School of Environment and Information Sciences
Yokohama National University, Japan
Email: watanabe-yohei-xs@ynu.jp

Junji Shikata
Graduate School of Environment and Information Sciences
Yokohama National University, Japan
Email: shikata@ynu.ac.jp

Abstract—In modern cryptography, the secret sharing scheme is an important cryptographic primitive and it is used in various situations. In this paper, a timed-release secret sharing scheme (TR-SS) with information-theoretic security is first studied. TR-SS is a secret sharing scheme with the property that participants more than a threshold number can reconstruct a secret by using their shares only when the time specified by a dealer has come. Specifically, in this paper we first introduce a model and formalization of security for TR-SS based on the traditional secret sharing scheme and information-theoretic timed-release security. We also derive tight lower bounds on the sizes of shares, time-signals, and entities’ secret-keys required for TR-SS. In addition, we propose a direct construction for TR-SS. Our direct construction is optimal in the sense that the construction meets equality in each of our bounds. As a result, it is shown that the timed-release security can be realized without any additional redundancy on the share-size.

I. INTRODUCTION

Secret sharing schemes were proposed independently by Shamir [1] and Blakley [2] in (k, n)-threshold secret sharing (for example, see [1]), a dealer shares a secret among all participants, and then, k participants can reconstruct the secret while any k − 1 participants obtain no information on the secret. Since Shamir and Blakley proposed secret sharing schemes, various research on them have been reported. Recently, in a real world setting, secret sharing schemes have been considered as applications, especially for cloud computing (e.g., secure data storage services).

On the other hand, “time” is intimately related to our lives. We get up, eat something, do a job, and get asleep at a time of our (or someone’s) choice. From the above reason, it appears that cryptographic protocols associated with “time” are useful and meaningful. Actually, as those protocols, timed-release cryptographic protocols introduced in [3] are well-known.

From the above discussion, we study secret sharing schemes with timed-release security in this paper.

Timed-Release Security. Informally, the goal of timed-release cryptography is to securely send a 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 [3] in 1993, and after that, Rivest et al. [4] developed it in a systematic and formal way. Since Rivest et al. gave a formal definition of timed-release encryption (TRE) in [4], various research on timed-release cryptography including timed-release signatures (e.g., [5], [6]) and timed-release encryption have been done based on computational security. In particular, timed-release public key encryption (TR-PKE for short) has been recently researched on intensively (e.g., [7], [8], [9]). Recently, information-theoretically (or unconditionally) secure timed-release cryptography was proposed by Watanabe et al. [10]. In addition, they investigated not only an encryption but also a key-agreement and an authentication code with information-theoretic timed-release security. To the best of our knowledge, however, there is no paper which reports on the study of secret sharing schemes with (information-theoretic) timed-release security.

Our Contribution. In this paper, we study a timed-release secret sharing scheme (TR-SS) with information-theoretic security. We begin by formalizing a model and a security notion of (k1, k2, n)-TR-SS based on those of (k, n)-threshold secret sharing scheme (e.g., [1], [11]) and those of information-theoretically secure timed-release schemes [10], where k1 and k2 are threshold values, n is the number of participants. In addition, we also show lower bounds on the sizes of shares, time-signals, and entities’ secret-keys required for (k1, k2, n)-TR-SS. Moreover, we provide a direct construction of (k1, k2, 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. In particular, it is shown that the timed-release security can be realized without any additional redundancy on the share-size.

Applications of TR-SS. TR-SS is useful when one wants to add the application of secret sharing schemes to timed-release property. As examples, we introduce three types of applications of TR-SS as follows.

As one of applications of secret sharing schemes, dispersion of risk on sending a data by physical mail is known. When we send a data by physical mail, we generate shares from the data, and then send each share by physical mail. If some
(subthreshold) shares are leaked in the mails, no information on the data will be leaked from the leaked shares. This application is useful when we send a data to one place. However, when we send a data to multitude places (e.g. the head office sends a sensitive data to branch offices all over the world), dates of reconstructing data vary for each place, since arrival dates of data vary for each place (we call this problem the reconstruction delay). By applying secret sharing schemes to the property that participants can reconstruct a secret after the specified time has come, we can provide a solution to the problem of the reconstruction delay: For example, the head office specifies appropriate future time when branch offices can reconstruct a data, generates shares from the data, and as a result, every branch office can reconstruct the data only after all branch offices have received shares.

Also, we can apply TR-SS to cloud data storage services in such a way that multiple users will share common data at some future time via cloud data storage services. For example, suppose that a teacher wants to share lecture notes or homework among students at noon next Friday, however, he/she will not be able to do it at that time for some reason (e.g. he/she is on board an airplane at that time). Then, he/she generates shares from the data beforehand by using TR-SS, and uploads the shares to the storage. Consequently, students can get lecture notes or homework only when the time has come (i.e., at noon next Friday).

In addition to this, TR-SS is useful when an authority (e.g., a government or a company) makes some sensitive announcement through mass media. Specifically, the authority specifies an announcement date and generates shares from the data including the announcement by using TR-SS, then the authority sends shares to mass media in advance. As a result, mass media can report the announcement at the specified date, whereas they cannot get any information on the announcement before the date comes.

Note that for the above applications it is better to utilize TR-SS than (information-theoretically secure) TRE, since all users have to share own secret-keys in advance in TRE whereas only a dealer and a time server need secret-keys in TR-SS. In addition, the use of TRE would require more communication complexity than the use of TR-SS in some applications (e.g., the third application above). Therefore, we can consider various applications in which TR-SS works more effectively than TRE. If TRE is applied to the above third application, since the announcement will be finally disclosed at the announcement date, it is okay to send the decryption-key along with the ciphertext including the announcement instead of shares. However, its communication complexity is increased as compared to it for applying TR-SS.

The rest of this paper is organized as follows. In Section III we introduce a model of \((k_1, k_2, n)\)-TR-SS based on the ideas according to [1], [11], [10], and formalize a security notion of \((k_1, k_2, n)\)-TR-SS. In Section III we show lower bounds on share-size, time-signal-size, and key-size required for \((k_1, k_2, n)\)-TR-SS. In Section IV we propose a direct construction and show the construction is provably secure and optimal. Finally, in Section V we give concluding remarks of this paper.

Throughout this paper, we use the following notation. Generally speaking, \(X\) indicates a random variable which takes values in \(\mathcal{X}\) (e.g., \(A, B,\) and \(C\) are random variables which take values in \(A, B,\) and \(C\), respectively). For any finite set \(Z\) and arbitrary non-negative integers \(z_1, z_2\), let \(\mathcal{P}(Z, z_1, z_2) := \{Z \subset Z | z_1 \leq |Z| \leq z_2\}\) be the family of all subsets of \(Z\) whose cardinality is at least \(z_1\) but no more than \(z_2\).

II. MODEL AND SECURITY DEFINITION

In this section, we propose a model and a security definition of \((k_1, k_2, n)\)-timed-release secret sharing schemes (TR-SS). First, we introduce a model of \((k_1, k_2, n)\)-TR-SS. Unlike traditional secret sharing schemes [2], [1], we assume that \((k_1, k_2, n)\)-TR-SS, there are \(n + 3\) entities, a dealer \(D\), \(n\) participants \(P_1, P_2, \ldots, P_n\), a time-server \(T S\) for broadcasting time-signals at most \(\tau\) times and a trusted initializer \(TA\) where \(n\) and \(\tau\) are positive integers. \(D\) can specify two kinds of threshold values, \(k_1\) and \(k_2\) with \(k_1 \leq k_2\): \(k_1\) 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. In this paper, we assume that the identity of each user \(P_i\) is also denoted by \(P_i\).

Informally, \((k_1, k_2, n)\)-TR-SS is executed as follows. First, \(TA\) generates secret-keys on behalf of \(D\) and \(T S\). After distributing these keys via secure channels, \(TA\) deletes it in his memory. Next, \(D\) specifies future time, as \(D\) wants, when a secret can be reconstructed by at least \(k_1\) 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 \(T S\) periodically broadcasts a time-signal which is generated by using his secret-key. When the specified time has come, at least \(k_1\) participants can compute the secret by using their shares and the time-signal of the specified time. On the other hand, at anytime, no less than \(k_2\) participants can compute the secret by using only their shares (without any time-signals).

Formally, we give the definition of \((k_1, k_2, n)\)-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. \(T := \{1, 2, \ldots, \tau\}\) is a set of time. Let \(U_t^{(i)}\) be the set of possible \(P_i\)’s shares at the time \(t \in T\). Also, \(U_t := \bigcup_{i=1}^{n} U_t^{(i)}\) is a set of possible \(P_i\)’s shares for \(i \in \{1, 2, \ldots, n\}\), and let \(U := \bigcup_{i=1}^{n} U_t\). In addition, \(TT_t^{(i)}\) is a set of time-signals at time \(t\), and let \(TT_t := \bigcup_{i=1}^{n} TT_t^{(i)}\). Furthermore, for any subset of participants \(J = \{P_1, \ldots, P_i\} \subset \mathcal{P},\)
\( U_{J}^{(t)} := U_{i_{1}}^{(t)} \times \cdots \times U_{i_{r}}^{(t)} \) denotes the set of possible shares held by \( J \).

**Definition 1 (TR-SS).** A \((k_{1}, k_{2}, n)\)-timed-release secret sharing scheme \((k_{1}, k_{2}, n)\)-TR-SS involves \( n + 3 \) entities, \( T_{A}, D, P_{1}, \ldots, P_{n} \), and \( T_{S} \), and consists of five phases, Initialize, Extract, Share, Reconstruct with time-signals and Reconstruct without time-signals, and five finite spaces, \( S, SK, U, T, \) and \( T_{T} \). \( \Pi \) is executed based on the following phases as follows.

1. **Initialize.** \( T_{A} \) generates a secret-key \( sk \in SK \) for \( T_{S} \) and \( D \). These keys are distributed to corresponding entities via secure channels. After distributing these keys, \( T_{A} \) deletes them from his memory. And, \( D \) and \( T_{S} \) keep their keys secret, respectively.

2. **Share.** A dealer \( D \) selects a secret \( s \in S \) according to \( P_{S} \). Then, \( D \) specifies \( k_{1}, k_{2} \in \{1, \ldots, n\}\left( k_{1} \leq k_{2} \right) \) and future time \( t \in T \) when at least \( k_{1} \) participants can reconstruct \( s \). Then, on input the secret \( s \in S \), the specified time \( t \in T \) and a secret-key \( sk \), \( D \) computes a share \( u_{i}^{(t)} \in U_{i}^{(t)} \) 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, respectively.

3. **Extract.** For broadcasting a time-signal at each time \( t \), \( T_{S} \) generates a time-signal \( ts^{(t)} \in T_{T} \) by using his secret-key \( sk \) and time \( t \in T \), where for simplicity we assume that \( ts^{(t)} \) is deterministically computed by \( t \) and \( sk \). Then, \( T_{S} \) broadcasts it to all participants via a (authenticated) broadcast channel.

4. **Reconstruct with time-signals.** At the specified time \( t \), any set of participants \( \mathcal{A} = \{P_{i_{1}}, \ldots, P_{i_{r}}\} \in \mathcal{P}S(P, k_{1}, k_{2} - 1) \) can reconstruct the secret \( s \) by using their shares \((u_{i_{1}}^{(t)}, \ldots, u_{i_{r}}^{(t)})(k_{1} \leq j < k_{2}) \) and a time-signal of the specified time \( ts^{(t)} \).

5. **Reconstruct without time-signals.** At anytime, any set of participants \( \mathcal{A} = \{P_{i_{1}}, \ldots, P_{i_{r}}\} \in \mathcal{P}S(P, k_{2}, n) \) can reconstruct the secret \( s \) by using only their shares \((u_{i_{1}}^{(t)}, \ldots, u_{i_{r}}^{(t)})(k_{2} \leq j \leq n) \).

In the above model, we assume that \( \Pi \) meets the following correctness properties:

(a) If \( D \) correctly completes the phase Share and \( T_{S} \) correctly completes the phase Extract, then, for all possible \( i \in \{1, 2, \ldots, n\}, t \in T, s \in S, u_{i}^{(t)} \in U_{i}^{(t)}, \) and \( ts^{(t)} \in T_{T}^{(t)} \), it holds that any \( \mathcal{A} \in \mathcal{P}S(P, k_{1}, k_{2} - 1) \) will correctly reconstruct the secret \( s \) at the end of phase Reconstruct with time-signals, namely,
\[
H(S \mid U_{\mathcal{A}}^{(t)}, T_{S}^{(t)}) = 0.
\]

(b) If \( D \) correctly completes the phase Share, then, for all possible \( i \in \{1, 2, \ldots, n\}, t \in T, s \in S, \) and \( u_{i}^{(t)} \in U_{i}^{(t)} \), it holds that any \( \mathcal{A} \in \mathcal{P}S(P, k_{2}, n) \) will correctly reconstruct the secret \( s \) at the end of phase Reconstruct without time-signals, namely,
\[
H(S \mid U_{\mathcal{A}}^{(t)}, T_{S}^{(t)}) = 0.
\]

Next, we formalize a security definition of \((k_{1}, k_{2}, n)\)-TR-SS based on the idea of the information-theoretic timed-release security [10] and secret sharing schemes (e.g., see [11]). In \((k_{1}, k_{2}, n)\)-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 secret sharing: less than \( k_{1} \) participants cannot obtain any information on a secret. In addition to this, as the second security we want to require that even at least \( k_{2} \) but no more than \( k_{2} \) 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), since we consider timed-release security in this paper. Therefore, we formally define the security of \((k_{1}, k_{2}, n)\)-TR-SS as follows.

**Definition 2 (Security).** Let \( \Pi \) be \((k_{1}, k_{2}, n)\)-TR-SS. \( \Pi \) is said to be secure if the following conditions are satisfied:

(i) For any \( F \in \mathcal{P}S(P, 1, k_{1} - 1) \) and for any \( t \in T \), it holds that
\[
H(S \mid U_{F}^{(t)}, T_{S}^{(1)}, \ldots, T_{S}^{(r)}) = H(S).
\]

(ii) For any \( F \in \mathcal{P}S(P, k_{1}, k_{2} - 1) \) and \( t \in T \), it holds that
\[
H(S \mid U_{F}^{(t)}, T_{S}^{(1)}, \ldots, T_{S}^{(t-1)}, T_{S}^{(t+1)}, \ldots, T_{S}^{(r)}) = H(S).
\]

Intuitively, the meaning of two conditions (i) and (ii) in Definition 2 is explained as follows. (i) No information on a secret is obtained by any set of less than \( k_{1} \) participants, even if they obtain time-signals at all the time; (ii) 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\(^{2}\).

**Remark 1.** We can also consider the following security definition (the condition (iii)) instead of (i): No information on a secret is obtained by collusion of \( T_{S} \) and any set of less than \( k \) participants, namely this is defined as follows.

(iii) For any \( F \in \mathcal{P}S(P, 1, k_{1} - 1) \) and for any \( t \in T \), it holds that
\[
H(S \mid U_{F}^{(t)}, SK) = H(S).
\]

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 \( k_{1} \) participants colludes with \( T_{S} \) in the real world;

\(^{2}\) In this sense, we have formalized the security notion stronger than the security that any set of participants cannot obtain any information on a secret before the specified time, as is the same approach considered in [10].
and secondly, our lower bounds in Theorem 1 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 1 since Definition 2 is weaker. Interestingly, our direct construction in Subsection IV-B 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 secret-key is generated independently of a secret.

Remark 2. In the case of \( k = k_1 = k_2 \), \( \mathcal{T} = \emptyset \) (i.e. the dealer does not specify future time), and \( \text{sk} \) is empty string, the model and security definition of secure \((k, k, n)\)-TR-SS (Definitions 1 and 2) are the same as those of traditional \((k, n)\)-threshold secret sharing schemes. Namely, our model of TR-SS includes the model of traditional secret sharing schemes.

III. LOWER BOUNDS

In this section, we show lower bounds on share-size, time-signal-size, and key-size required for secure \((k_1, k_2, n)\)-TR-SS as follows.

Theorem 1. Let \( \Pi \) be any secure \((k_1, k_2, n)\)-TR-SS. Then, for any \( i \in \{1, 2, \ldots, n\} \) and for any \( t \in \mathcal{T} \), we have

\[
\begin{align*}
(i) \quad & H(U_i^{(t)}) \geq H(S), \\
(ii) \quad & H(T(1)) \geq H(S), \\
(iii) \quad & H(SK) \geq \tau H(S).
\end{align*}
\]

The proof follows from the following lemmas.

Lemma 1. \( H(U_i^{(t)}) \geq H(S) \) for any \( i \in \{1, 2, \ldots, n\} \) and any \( t \in \mathcal{T} \).

Proof. The proof of this lemma can be proved in a way similar to the proof in [1] Theorem 1. For arbitrary \( i \in \{1, 2, \ldots, n\} \), we take a subset \( \mathcal{B} := \{P_{i_1}, P_{i_2}, \ldots, P_{i_{k_1-1}}\} \subset \mathcal{P} \) of participants such that \( P_i \notin \mathcal{B} \). Then, for any \( t \in \mathcal{T} \), we have

\[
\begin{align*}
& H(U_i^{(t)}) \geq H(U_i^{(t)} | U_B^{(t)}, T(1)) \\
& \geq I(S; U_i^{(t)} | U_B^{(t)}, T(1)) \\
& = H(S | U_B^{(t)}, T(1)) \geq H(S)
\end{align*}
\]

where (1) follows from the correctness of \((k_1, k_2, n)\)-TR-SS and (2) follows from the condition (i) in Definition 2.

Lemma 2. \( H(T(1) | T(1), \ldots, T(t-1)) \geq H(S) \) for any \( t \in \mathcal{T} \). In particular, \( H(T(1)) \geq H(S) \) for any \( t \in \mathcal{T} \).

Proof. For any \( A \in \mathcal{PS}(\mathcal{P}, k_1, k_2 - 1) \) and any \( t \in \mathcal{T} \), we have

\[
\begin{align*}
& H(T(1)) \geq H(T(1) | T(1), \ldots, T(t-1)) \\
& \geq H(T(1) | U_A^{(t)}, T(1), \ldots, T(t-1)) \\
& \geq I(S; T(1) | U_A^{(t)}, T(1), \ldots, T(t-1)) \\
& = H(S | U_A^{(t)}, T(1), \ldots, T(t-1)) \geq H(S)
\end{align*}
\]

where (3) follows from the correctness of \((k_1, k_2, n)\)-TR-SS and (4) follows from the condition (ii) in Definition 2.

Lemma 3. \( H(SK) \geq \tau H(S) \).

Proof. We have

\[
\begin{align*}
& H(SK) \geq I(T(1), \ldots, T(\tau); SK) \\
& = H(T(1), \ldots, T(\tau)) - H(T(1), \ldots, T(\tau) | SK) \\
& = H(T(1), \ldots, T(\tau)) \\
& = \sum_{i=1}^{\tau} H(T(1), \ldots, T(t-1)) \\
& \geq \tau H(S),
\end{align*}
\]

where the last inequality follows from Lemma 2.

Proof of Theorem 4. From Lemmas 1, 2, the proof of Theorem 1 is completed.

As we will see in Section IV, 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 \((k_1, k_2, n)\)-TR-SS as follows.

Definition 3. 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 1.

Remark 3. The secret sharing scheme such that the size of each participant’s share is equal to that of the secret is often called the ideal secret sharing. The construction of \((k_1, k_2, n)\)-TR-SS in Subsection IV-B is optimal, hence, in this sense we achieve ideal TR-SS. In terms of 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 share-size, our results are also regarded as the extension of traditional secret sharing schemes.

IV. CONSTRUCTION

We propose a direct construction of \((k_1, k_2, n)\)-TR-SS. In addition, it is shown that our construction is optimal. First, we show a naive construction based on two \((k, n)\)-threshold secret sharing schemes, which is not optimal.

A. Naive Construction

1) Initialize. Let \( q \) be a \( \lambda \)-bit prime power, where \( q > \max(n, \tau) \), and \( \mathbb{F}_q \) be the finite field with \( q \) elements. We assume that the identity of each participant \( P_i \) is encoded as \( P_i \in \mathbb{F}_q \setminus \{0\} \). Also, we assume \( T = \{1, 2, \ldots, \tau\} \subset \mathbb{F}_q \setminus \{0\} \) by using appropriate encoding. First, \( TA \) chooses a polynomial \( sk^*(y) := \sum_{i=0}^{\tau-1} b_i y^i \) over \( \mathbb{F}_q \), each coefficient \( b_i \) is randomly and uniformly chosen from \( \mathbb{F}_q \). \( TA \) sends a secret-key \( sk := sk^*(y) \) to \( TS \) and \( D \) via secure channels, respectively.

2) Share. First, \( D \) chooses a secret \( s \in \mathbb{F}_q \). Also, \( D \) specifies the time \( t \) when at least \( k_1 \) participants can reconstruct the secret and computes \( sk^*(t) \). Next, \( D \) randomly chooses two polynomials \( f_1(x) := s + \sum_{i=1}^{k_1-1} a_{11} x^i \) and \( f_2(x) := s + \sum_{i=1}^{k_2-1} a_{22} x^i \)
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 $ts^*$ and time $t \in T$, $TS$ computes a time-signal at time $t$, $ts^{(t)} := sk^*(t)$. Then, $TS$ broadcasts $ts^{(t)}$ to all participants via a (authenticated) broadcast channel.

4) **Reconstruct with time-signals.** First, $\mathcal{A} = \{P_{i_1}, P_{i_2}, \ldots, P_{i_{k_1}}\} \in \mathcal{PS}(P, k_1, 1)$ computes $s + sk^*(t)$ by Lagrange interpolation:

$$s + sk^*(t) = \sum_{j=1}^{k_1} \left( \prod_{l \neq j} \frac{P_{i_l} - P_{i_l}}{P_{i_j} - P_{i_l}} \right) f_1(P_{i_j}),$$

from $(f_1(P_{i_1}), \ldots, f_1(P_{i_{k_1}}))$. After receiving $ts^{(t)} = sk^*(t)$, they can compute and get $s = s + sk^*(t) - ts^{(t)}$.

5) **Reconstruct without time-signals.** any $\mathcal{A} = \{P_{i_1}, P_{i_2}, \ldots, P_{i_{k_2}}\} \in \mathcal{PS}(P, k_2, 2)$ computes

$$s = \sum_{j=1}^{k_2} \left( \prod_{l \neq j} \frac{P_{i_l} - P_{i_l}}{P_{i_j} - P_{i_l}} \right) f_2(P_{i_j}),$$

by Lagrange interpolation from $(f_2(P_{i_1}), \ldots, f_2(P_{i_{k_2}}))$.

The above construction is secure and simple, however not optimal since the resulting share-size is twice as large as that of secrets.

**B. Optimal Construction**

To achieve an optimal construction, we use the technique as in [13]: In the phase **Share**, the dealer computes the public parameters, and the public parameters are broadcasted to users or else stored on a publicly accessible authenticated bulletin board. The detail of our construction is given as follows.

1) **Initialize.** This phase follows the same procedure as that of the naive construction.

2) **Share.** First, $D$ chooses a secret $s \in \mathbb{F}_q$. 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, $D$ divides $f(x)$ into $h(x)$ and $g(x)$ such that $h(x) := s + \sum_{i=1}^{k_1-1} a_i x^i$ and $g(x) := \sum_{i=k_1}^{k_2-1} a_i x^i$ (i.e., $f(x) = h(x) + g(x)$). Then, $D$ computes a share $u_i^{(t)} := f(P_i)$ and a public parameter $r_i^{(t)} := sk^*(t) - g(P_i) (i = 1, 2, \ldots, n)$. Finally, $D$ sends $(u_i^{(t)}, t)$ to $P_i (i = 1, 2, \ldots, n)$ via a secure channel and discloses $(r_1^{(t)}, \ldots, r_n^{(t)})$.

3) **Extract.** This phase follows the same procedure as that of the naive construction.

4) **Reconstruct with time-signals.** Suppose that all participants receive $ts^{(t)} = sk^*(t)$. Let $\mathcal{A} = \{P_{i_1}, P_{i_2}, \ldots, P_{i_{k_1}}\} \in \mathcal{PS}(P, k_1, 1)$ be a set of any $k_1$ participants. First, each $P_{i_j} \in \mathcal{A}$ computes $h(P_{i_j}) := f(P_{i_j}) - (ts^{(t)} - r_i^{(t)} (j = 1, \ldots, k_1))$. Then, $A$ computes

$$s = \sum_{j=1}^{k_1} \left( \prod_{l \neq j} \frac{P_{i_l} - P_{i_l}}{P_{i_l} - P_{i_l}} \right) h(P_{i_j}),$$

by Lagrange interpolation from $(h(P_{i_1}), \ldots, h(P_{i_{k_1}}))$.

**Theorem 2.** The resulting $(k_1, k_2, n)$-TR-SS II by the above construction is secure and optimal.

**Proof.** First, we show the proof of (i) in Definition 2. Assume that $k_1 - 1$ participants $\mathcal{F} = \{P_{i_1}, \ldots, P_{i_{k_1-1}}\}$ try to guess $s$ by using their shares, public parameters, and all time-signals. $\mathcal{F}$ can compute $h(P_{i_l}) = f(P_{i_l}) - (sk^*(t) - r_i^{(t)}) (l = 1, \ldots, k_1 - 1)$, however, cannot guess the information on $s$ from $h(P_{i_1}), \ldots, h(P_{i_{k_1-1}})$ with probability larger than random guessing since the degree of $h(x)$ is at most $k_1 - 1$. Therefore, for any $\mathcal{F} \in \mathcal{PS}(P, 1, k_1 - 1)$ and any $t \in T$, we have $H(S | U_{\mathcal{F}}^{(t)}, T^{(1)}, \ldots, T^{(t)}) = H(S)$.

Next, we show the proof of (ii) in Definition 2. Without loss of generality, we suppose that $\tau$ is a specified time, and that $k_2 - 1$ participants try to guess $h(P_{i_l}) = f(P_{i_l}) - g(P_{i_l}) = f(P_{i_l}) - (sk^*(\tau) - r_i^{(\tau)}) (l = 1, \ldots, k_2 - 1)$ by using their shares, public parameters, and time-signals at all the time except the time $\tau$. Namely, they attempt to guess $ts^*(\tau)$.

They know $\tau - 1$ time-signals, however, they cannot guess at least one coefficient of $sk^*(y)$ with probability larger than $1/q$ since the degree of $sk^*(y)$ is at most $\tau - 1$. Therefore, we have $H(S | U_{\mathcal{F}}^{(\tau)}, U_{\mathcal{F}}^{(\tau+1)}, T^{(1)}, \ldots, T^{(\tau)}) = H(S)$.

Hence, for any $\mathcal{A} \in \mathcal{PS}(P, k_1, k_2 - 1)$ and any $t \in T$, we have $H(S | U_{\mathcal{A}}^{(t)}, T^{(1)}, \ldots, T^{(t-1)}, T^{(t+1)}, \ldots, T^{(\tau)}) = H(S)$.

Finally, it is straightforward to see that the construction satisfies all the equalities of lower bounds in Theorem 1.

Therefore, the above construction is optimal. □

**V. CONCLUDING REMARKS**

In this paper, we studied a secret sharing scheme with timerelease security. Specifically, we first proposed a model and formalization of security for $(k_1, k_2, n)$-timed-release secret-sharing schemes ($(k_1, k_2, n)$-TR-SS). In addition, we derived tight lower bounds on share-size, time-signal-size, and key-size required for TR-SS, and we proposed the optimal direct construction.

It would be interesting to extend our results to timed-release verifiable secret sharing schemes, and furthermore, to multiparty computation schemes with timed-release security.
REFERENCES

[1] A. Shamir, “How to share a secret,” Commun. ACM, vol. 22, no. 11, pp. 612–613, Nov. 1979.

[2] G. Blakley, “Safeguarding cryptographic keys,” in Proceedings of the 1979 AFIPS National Computer Conference. Montval, NJ, USA: AFIPS Press, 1979, pp. 313–317.

[3] T. May, “Timed-release crypto,” 1993.

[4] R. L. Rivest, A. Shamir, and D. A. Wagner, “Time-lock puzzles and timed-release crypto,” MIT Laboratory for Computer Science, Tech. Rep. Technical memo MIT/LCS/TR-684, 1996. (Revision 3/10/96).

[5] J. Garay and M. Jakobsson, “Timed release of standard digital signatures,” in Financial Cryptography, M. Blaze, Ed., vol. 2357. Springer Berlin Heidelberg, 2003, pp. 168–182.

[6] J. Garay and C. Pomerance, “Timed fair exchange of standard signatures,” in Financial Cryptography, R. Wright, Ed., vol. 2742. Springer Berlin Heidelberg, 2003, pp. 190–207.

[7] A.-F. Chan and I. Blake, “Scalable, server-passive, user-anonymous timed release cryptography,” in Distributed Computing Systems, 2005. ICDCS 2005. Proceedings. 25th IEEE International Conference on, 2005, pp. 504–513.

[8] J. Cathalo, B. Libert, and J.-J. Quisquater, “Efficient and non-interactive timed-release encryption,” in Information and Communications Security, S. Qing, W. Mao, J. López, and G. Wang, Eds., vol. 3783. Springer Berlin Heidelberg, 2005, pp. 291–303.

[9] K. Chalkias, D. Hristu-Varsakelis, and G. Stephanides, “Improved anonymous timed-release encryption,” in Computer Security – ESORICS 2007, J. Biskup and J. López, Eds., vol. 4734. Springer Berlin Heidelberg, 2007, pp. 311–326.

[10] Y. Watanabe, T. Seito, and J. Shikata, “Information-theoretic timed-release security: Key-agreement, encryption, and authentication codes,” in Information Theoretic Security, A. Smith, Ed., vol. 7412. Springer Berlin Heidelberg, 2012, pp. 167–186.

[11] E. Karnin, J. Greene, and M. Hellman, “On secret sharing systems,” Information Theory, IEEE Transactions on, vol. 29, no. 1, pp. 35–41, 1983.

[12] R. L. Rivest, “Unconditionally secure commitment and oblivious transfer schemes using private channels and a trusted initializer;” 1999.

[13] M. Jhanwar and R. Safavi-Naini, “Unconditionally-secure robust secret sharing with minimum share size,” in Financial Cryptography and Data Security, A.-R. Sadeghi, Ed., vol. 7859. Springer Berlin Heidelberg, 2013, pp. 96–110.