フォワード安全暗号を用いたタイムリリース暗号の一般的構成の安全性証明

笠松 宏平* 松田 隆宏 † 江村 恵太 ‡ 花岡 悟一郎 † 今井 秀樹*†

*中央大学理工学研究科〒 112-8551 東京都文京区春日 1-13-7
kasamatsu-kouhei@imailab.jp

† 産業技術総合研究所 情報セキュリティ研究センター,
〒 305-8568 茨城県つくば市梅園 1-1-1 つくば中央第 2 事業所 つくば本部・情報技術共同研究棟

‡ 北陸先端科学技術大学院大学 高信頼性組み込みシステムセンター,
〒 923-1292 石川県能美市旭台 1-1,

あらまし タイムリリース暗号 (Timed-Release Encryption,TRE) とは, 送信者が受信者の復号できる時間を指定して暗号文を生成する方式であり, 信頼できる時報局が時刻鍵と呼ばれる暗号文の復号に必要な情報を一定時刻ごとに配信することで実現できることが知られている. しかしながらこの時報局を用いたアプローチでは, 受信者が復号に必要な時刻鍵を紛失した時にその暗号文を復号できなくなるという問題が生じる. その解決法の一つとして, 時刻鍵作成時に指定された時刻以前に作成された暗号文であれば復号可能となるように時刻鍵を更新していく手法が知られており, またこのような時刻鍵の更新がフォワード安全暗号 (Forward Secure Encryption, FSE) を利用して実現可能であることが Boneh ら, Chan らにより示唆されている. しかしながらその厳密な定義と安全性の証明は為されておらず, さらに時刻鍵の紛失問題を解決する既存研究との比較検討すら行われていない. そこで本稿では, FSE から時刻鍵が更新可能な TRE を構成する際の厳密な定義及び安全性証明を与え, さらに既存研究との効率性比較を行うことで FSE を用いた TRE 構成の優位性を示す.

A Generic Construction of Timed-Release Encryption from Forward Secure Encryption

Kouhei Kasamatsu* Takahiro Matsuda † Keita Emura‡
Goichiro Hanaoka† Hideki Imai†

*Faculty of Science and Engineering, Chuo University 1-13-27 Kasuga, Bunkyo-ku Tokyo 112-8551,

†Research Center for Information Security, National Institute of Advanced Industrial Science and Technology, Tsukuba Headquarters' building Tsukuba Central 2 1-1-1 Umezono, Tsukuba-shi, Ibaraki 305-8568 JAPAN

‡Center for Highly Dependable Embedded Systems Technology,
Japan Advanced Institute Science and Technology (JAIST), 1-1, Asahidai, Nomi, Ishikawa

Abstract In timed-release encryption (TRE), a central timer server will, in each time interval, release a timed-release key which enables a receiver to decrypt any ciphertext encrypted for that time interval. However, this approach suffers from the drawback that if the receiver is temporary off-line or otherwise unavailable in a given time interval, and therefore does not receive the timed-release key, he will not be able to decrypt. One solution to this problem is to ensure that a timed-release key can be used to decrypt ciphertexts encrypted for any previous time interval. It has been suggested in the literature that a scheme with this property can be constructed from forward secure encryption (FSE), but such a construction has neither been formally defined nor proved secure. In this paper, we address this by proposing a generic construction of a TRE scheme from FSE, and formally proving this construction secure.

1 はじめに

電子入札，電子投票，そしてオンライン試験のようにある時刻になると中身を閲覧できる機能が重要な役割を果たすアプリケーションが数多く存在する．しかし，このような機能を実現するためには一般的な公開鍵暗号では不十分である．そこで 1993 年に May [14] により送信者が受信者の復号できる時間を指定して暗号文を生成する方式としてタイムリリース暗号 (Timed-Release Encryption,TRE) 方式が初

めて議論され，以降多く研究が行われている．

TRE 方式の実現方法にはいくつかの方法があるが，本稿ではその中でも一定時刻ごとに時刻鍵と呼ばれる暗号文の復号に必要な情報をユーザと対話せず配信する信頼できる時報局の存在を仮定した方式について取り上げる．なお TRE 方式の安全性は以下の 3 つが主に研究されているが，本稿では 3. の開封時刻前の受信者に対する秘匿のみを考える．

1. 部外者に対する秘匿
2. 時報局に対する秘匿
3. 開封時刻前の受信者に対する秘匿

その理由は 1. の部外者に対する秘匿は 2. の時報局に対する秘匿が満たされれば自動的に満たされるため 2. と 3. の安全性について考えれば十分であり，2. の安全性は送信者と受信者が公開鍵暗号を用いてなど安全な通信路を確立すれば達成できる安全性だからである [10][13].

この時報局を用いたアプローチの際に生じる問題の一つは，もし時報局が配信した時刻鍵を受信者が受け取れない場合，その暗号文を復号できないことである．例えば ID ベース暗号を利用した TRE の一般的構成法 [15][13] では時刻を ID とみなして時刻鍵を生成するために, 暗号化時に指定された時刻と時刻鍵生成時に指定された時刻とが一致している必要があり 時刻鍵の紛失が問題となる．この問題は時報局が時刻鍵を逐次保存/公開することで解決可能ではあるが, その場合全ての時刻に対する鍵を時報局が公開する必要があり, とても効率的とはいえない．この問題を解決するため, [8],[18],[9] などの研究が知られている．

[8] では，ハッシュ関数を複数回用いて生成したワンタイムパスワードによる認証システム S/Key システム [12] と双線形写像を組み合わせることで時報局から配信された最新の鍵で失くした鍵を生成できる方式を提案している．[17] では受信者が時刻鍵を紛失した場合，送信者から復元信号をもらうことで，失くした時刻鍵に対応した暗号文も復号できる方式を提案している．[8][17] の方式は本稿で考える方式より暗号文サイズ，秘密鍵サイズは小さくなるが [8] は送信者，受信者，時報局それぞれが公開鍵と秘密鍵を持たなければならず，[17] では鍵の復元に送信者の復元信号が必要であるという欠点がある．

[18] では, 暗号文に対応する時刻以降に対応する鍵でその暗号文を復号可能である TRE 方式の一般的構成方法を提案しており，その具体例として大小比較を行えるアクセス構造 [4] を関数型暗号 [16] に実装することで構成している．またこのような “ある時刻以降であれば復号に使用可能” な時刻鍵がフォワード安全暗号 (Forward Secure Encryption, FSE) から構成可能であることが [9],[5] にて示唆されている．FSE は階層型 ID ベース暗号 (HIBE) より構成可能なことが示されており [7], 例えば [5] の HIBE を用いることで関数型暗号を用いた構成方法よりも小さいコストで TRE が構成できる (比較結果は 4 章参照のこと). また，[1] を用いれば格子に基づいた TRE も構成可能である. ここで ID ベース暗号を利用した TRE の一般的構成法 [13],[10] の構成要素として格子に基づいた ID ベース暗号 [11] を利用したとしても, 時刻鍵の紛失問題が解決されるわけではないことに注意されたい. すなわち FSE を介することで, 初めて時刻鍵の紛失問題を解決した格子ベースの TRE が構成できるといえる．

本研究の貢献　上記のように FSE を用いた TRE の構成は多くの利点がある一方で, [9],[5] ではその厳密な定義と安全性証明はされていない．そこで本稿ではその定義と安全性の証明を与え，さらに [5] の HIBE から構成した TRE と既存研究との比較を行い FSE から TRE を構成することの優位性を示す．

2　準備

本論文で用いる TRE と FSE の性質に関し簡単に紹介し，また厳密な定義を与える．

2.1　タイムリリース暗号

TRE 方式は時間の概念を取り入れ，送信者が受信者の復号できる時間を指定して暗号文を送ることができる方式である．本稿では信頼できる時報局が周期的に現在時刻に対応した秘密鍵を受信者に配信し，受信者は送信者の指定した時間に対応した秘密鍵を用いて暗号文を復号する実現方法を考える．

2.1.1　機能的要件

TRE 方式では送信者は受信者の復号できる時刻 t_r を指定してメッセージを暗号化する．従来の TRE 方式では受信者は送信者が指定した時刻 t_r に対応した秘密鍵 k_{t_r} でしか暗号文を復号できなかったが，本稿では時刻 t_r 以降の時刻鍵ならば暗号文を復号できるように正当性を修正した定義 [18] を以下に与える．

定義 2.1. *TRE* は平文空間 M, そして次のような４つの確率的多項式時間アルゴリズム (TRE.Setup, TRE.Ext, TRE.Enc, TRE.Dec) より定義される．

TRE.Setup$(1^k, T)$**:** セットアップアルゴリズムはセキュリティパラメータ 1^k とシステムの時間空間 T を入力として，公開パラメータ MPK とマスター鍵 MSK を出力する．

TRE.Ext(MSK, t)**:** 鍵生成アルゴリズムは MSK と時刻 t を入力として，時刻 t に対応する時刻鍵 k_t を出力する．

TRE.Enc(MPK, t_r, m)**:** 暗号化アルゴリズムは MPK と開封時刻 t_r とメッセージ m を入力として，生成した暗号文を CT とし $c = (t_r, CT)$ を出力する．

TRE.Dec(k_t, c): 復号アルゴリズムは k_t と c を入力として，復号結果 m または暗号文が無効であることを示すシンボル $\perp$ を出力する．

もし $c \leftarrow$ TRE.Enc(MPK, t_r, m) かつ k_t が TRE.Ext $(MPK, t(\geq t_r))$ によって出力された値ならば，TRE.Dec $(k_t, c) \rightarrow m$ であるとき， TRE 方式は正当性を満たす．

2.1.2 安全性の要件

1 章で説明したように本稿では TRE 方式の安全性のとして開封時刻前の受信者に対する秘匿のみを考える．その 3. の厳密な定義を次に与える．

定義 2.2. *TRE* 方式が *IND-TR-CCA* 安全であるとはすべての多項式時間攻撃者が次のゲームにおいて高々無視可能なアドバンテージしか持たないときである。

Setup チャレンジャー C は TRE.Setup$(1^k, T(k))$ を起動して公開パラメータ MPK とマスター秘密鍵 MSK を得て，攻撃者 A に MPK を与える．

Phase1 攻撃者 A は適応的に次のタイプのオラクルに複数回クエリする．

- **鍵生成クエリ** 時刻 $t_i \in [0, T-1]$ を入力として TRE.Ext(MSK, t_i) 時間 t_i に対応した秘密鍵 k_{t_i} を返す．
- **復号クエリ** $c_h = (t_h, CT_h)$ の組を入力として、TRE.Dec(k_{t_h}, CT_h) の計算結果であるメッセージ m_h を返す．ただし k_{t_h} は時間 t_h に対応した秘密鍵とする．

Challenge 攻撃者 A は 2 つのメッセージ $m_0, m_1 \in TRE.MSP$ と開封時刻 $t_r \in T$ を選ぶ。ただし、t_i を Phase1 でクエリしたときの値としたときに $t_r > t_i$ を満たすものとする．攻撃者 A は (m_0, m_1, t_r) をチャレンジャー C に渡す．チャレンジャー C はランダムビット $b \leftarrow \{0, 1\}$ を選び $CT^* =$ TRE.Enc(MPK, t_r, m_b) を計算して $c^* = (t_r, CT^*)$ を攻撃者 A に渡す。

Phase2 攻撃者 A は Phase1 と同じように適応的に鍵生成オラクルと復号オラクルへクエリし続ける．ただし，鍵生成オラクルへは $t \geq t_r$ なる時刻 t はクエリできず，復号オラクルへは $CT = CT^*$ かつ $t \geq t_r$ を満たす c はクエリできないものとする．

Guess 攻撃者 A は b を予測した値 b' を出力する．

上記のゲームにおける攻撃者のアドバンテージを $Adv_A(k) = |Pr[b' = b] - \frac{1}{2}|$ と定義する．

IND-TR-CPA ゲームの定義は上記のゲームから両方の復号オラクルを除いたものである．

2.2 フォワード安全暗号

ここでは，TRE の構成に用いる FSE の簡単な説明と厳密な定義を与える．

2.2.1 機能的要件

この方式は時間の経過によって使用する秘密鍵を変えることで，秘密鍵の漏洩に対する影響をある時刻のみに限定する．送信者は暗号化の際にメッセージとともに現在の時間 i を入力し暗号文を出力する．受信者はその時間に対応する秘密鍵を用いてその暗号文を復号する．ある時間が経過した後, 受信者は秘密鍵を次の時間用の秘密鍵へと更新し，古い鍵は削除する．なお受信者は外部とのやり取りなしに秘密鍵を更新できることに注意されたい. 秘密鍵からその秘密鍵の時間より前の鍵を計算することは困難な設計になっているため，鍵を漏洩してもその鍵に対応した時間以前の暗号文を読むことはできない. よって秘密鍵が漏洩しても対応した時間以外の暗号文を守ることができる．以下に FSE の厳密な定義を与える [7].

定義 2.3. フォワード安全暗号は平文空間 M と次のような 4 つの確率的多項式時間アルゴリズム組 $(FSE.Gen, FSE.Upd, FSE.Enc, FSE.Dec)$ より定義される*:*

FSE.Gen$(1^k, N)$: 鍵生成アルゴリズムはセキュリティパラメータ 1^k と時間の合計 N を入力として、公開鍵 PK と最初の秘密鍵 SK_0 を出力する．

FSE.Upd(PK, i, SK_i): 鍵更新アルゴリズム $FSE.Upd$ は PK, 時間 $i < N$, そして時間 i に対応する秘密鍵 SK_i を入力として次の時間に対応する秘密鍵 SK_{i+1} を出力する．

FSE.Enc(PK, i, m): 暗号化アルゴリズム $FSE.Enc$ は PK,$i \leq N$, そしてメッセージ m を入力として暗号文を CT とし，$c = (i, CT)$ を出力する．

FSE.Dec$(PK, SK_{i'}, c)$: 復号アルゴリズム $FSE.Dec$ は PK,$i \leq N$,SK_i, そして c を入力とし，復号結果 m または暗号文が無効であることを示すシンボル $\perp$ を出力する．

我々は次のように正当性を定める．正当性は FSE.Gen $(1^k, N)$ によって出力された任意の (PK, SK_0), 時刻 $i \in [0, N-1]$, 時刻 i' に対して正しい秘密鍵 $SK_{i'}$, そして任意のメッセージ m に対してもし $i' \leq i$ ならば $M =$ FSE.Dec$(PK, i, SK_i,$ FSE.Enc$(PK, i, m))$ が成り立つことである．

2.2.2 安全性の要件

次に FSE の安全性の厳密な定義を与える．

定義 2.4. フォワード安全暗号が暗号文選択攻撃に対して *fs-CCA* 安全であるとはすべての多項式関数 $N(\cdot)$ に対して次のゲームにおける任意の確率的多項式時間アルゴリズムのアドバンテージがセキュリティパラメータに対して無視可能関数であることである.

Setup チャレンジャー C は $\mathtt{FSE.Gen}(1^k, N(k))$ を起動し (PK, SK_0) を得て，攻撃者に PK を与える.

Phase1 攻撃者 A は適応的に break-in クエリを一度だけ行い，復号クエリを複数回行う.

- **break-in クエリ** $i \in [0, N-1]$ を入力として，チャレンジャー C は $SK_1 \leftarrow \mathtt{FSE.Upd}(PK, 0, SK_0), SK_2 \leftarrow \mathtt{FSE.Upd}(PK, 1, SK_1), \ldots, SK_i \leftarrow \mathtt{FSE.Upd}(PK, i-1, SK_{i-1})$ より得た時間 i に対応する秘密鍵 SK_i を返す
- **復号クエリ** $c_h = (i_h \in [0, N-1], CT_h)$ の組を入力として，チャレンジャー C は $\mathtt{FSE.Dec}(PK, SK_{i_h}, CT_h)$ の計算結果であるメッセージ m_h を返す．ただし SK_{i_h} は時間 i_h に対応する秘密鍵とする.

Challenge 攻撃者 A は 2 つのメッセージ m_0, m_1 と時刻 j を選び一度だけクエリする．チャレンジャー C はランダムビット b を得て，$CT^* = \mathtt{FSE.Enc}(PK, j, m_b)$ を計算し，攻撃者 A にチャレンジ暗号文 $c^* = (j, CT^*)$ を返す。ただし $0 \leq j < i < N$.

Phase2 攻撃者は適応的に Phase1 と同じような復号オラクルへクエリを行う．ただし，$CT = CT^*$ かつ $t \leq j$ であるクエリ $c = (t, CT)$ を復号オラクルに渡すことはできない.

Guess 攻撃者は推測したビット $b' \in \{0, 1\}$ を返す．もし $b' = b$ ならば攻撃者の勝利である.

上記のゲームにおける攻撃者のアドバンテージは $Adv_A^{fs-CCA}(k) = |Pr[b' = b] - \frac{1}{2}|$ のように定義する.

fs-CPA ゲームの定義は上記のゲームから両方の復号オラクルを除いたものである.

3 フォワード安全暗号を用いたタイムリリース暗号の構成

本章では FSE を用いた TRE の一般的構成方法を説明する．この構成方法については [5],[9] でその可能性が示唆されているが，その厳密な定義とこの構成方法において，fs-CCA 安全な FSE を用いて TRE を構成すれば IND-TR-CCA 安全な TRE が構成できることの証明を与える.

3.1 構成方法

FSE 方式を用いた TRE 方式の一般的構成法について説明する．図 3.1 の上の図は FSE 方式，下の図は TRE 方式の構成の概念図である．T はそのシステムの時間を，矢印は FSE 方式における鍵の更新を意味し，例えば上の図では SK_3 を持っていればその鍵を用いて矢印の先 SK_4 を作ることができ，一方矢印の逆方向への更新に当たる SK_2 への更新は困難であることを表している．図 3.1 からわかるように FSE 方式はある時間に対応する秘密鍵を持っていれば，鍵の更新を行うことによってそれ以降の時間に対応する暗号文を復号できる．この性質は 2.1 小節で紹介した TRE のある時間の鍵を持っていればそれ以降の暗号文を復号できる要件と同じである．したがって図 3.1 の下の図のように時報局がシステムの時間に対して FSE の秘密鍵を降順に配布し，それに対応した暗号化復号を送信者と受信者が行えば TRE を構成できる．以下にその構成方法と安全性の証明の詳細を示す.

FSE 方式 $(\mathtt{FSE.Gen}, \mathtt{FSE.Upd}, \mathtt{FSE.Enc}, \mathtt{FSE.Dec})$ が与えられたら，我々は TRE 方式 $(\mathtt{TRE.Setup}, \mathtt{TRE.KeyGen}, \mathtt{TRE.Enc}, \mathtt{TRE.Dec})$ を次のように構成できる．ただし両者の平文空間と時間空間は等しいものとし，時間空間の大きさはセキュリティパラメータの多項式とする:

$\mathtt{TRE.Setup}(1^k, N)$: $\mathtt{FSE.Gen}(1^k, N)$ を起動し (PK', SK_0) を得る．そしてそれぞれ $MPK = PK'$,$MSK = SK_0$ として返す

$\mathtt{TRE.KeyGen}(MSK, t)$: $SK_1 \leftarrow \mathtt{FSE.Upd}(MPK, 0, MSK)$, $SK_2 \leftarrow \mathtt{FSE.Upd}(MPK, 1, SK_1), \ldots, SK_{N-t-1} \leftarrow \mathtt{FSE.Upd}(PK, N-t-2, SK_{N-t-2})$ で SK_{N-t-1} を計算し $k_t = SK_{N-t-1}$ として返す．ただし，もし $t = N-1$ ならば $k_{N-1} = MSK$ とする.

$\mathtt{TRE.Enc}(MPK, t_r, m)$: $\mathtt{FSE.Enc}(MPK, N-t_r-1, m)$ を起動して暗号文 $c' = (t_r, CT)$ を得る．$c = c'$ として返す.

$\mathtt{TRE.Dec}(MPK, k_t, c)$: $\mathtt{FSE.Dec}(MPK, k_t, c)$ を起動してメッセージ m をそれを返す.

この方式の正当性について考える．もし $c \leftarrow \mathtt{TRE.Enc}(MPK, t_r, m)$ かつ k_t が $\mathtt{TRE.Ext}(MPK, t(\geq t_r))$ によって出力された値ならば，$\mathtt{TRE.Dec}(k_t, c) \rightarrow m$ であることである．$c \leftarrow \mathtt{FSE.Enc}(MPK, N-t_r-1, m)$ かつ k_t が $SK_1 \leftarrow \mathtt{FSE.Upd}(MPK, 0, MSK), SK_2 \leftarrow \mathtt{FSE.Upd}(MPK, 1, SK_1), \ldots, SK_{N-t-1} \leftarrow \mathtt{FSE.Upd}(PK, N-t-2, SK_{N-t-2})$ により $k_t = SK_{N-t-1}$ ならば，FSE の正当性から $\mathtt{FSE.Dec}(MPK, k_t, c) \rightarrow m$ を満たすのでこの方式は正当性を満たす.

3.2 安全性の証明

上記の構成方法において，fs-CCA 安全な FSE を用いて TRE を構成すれば IND-TR-CCA 安全な TRE が構成できることを証明する.

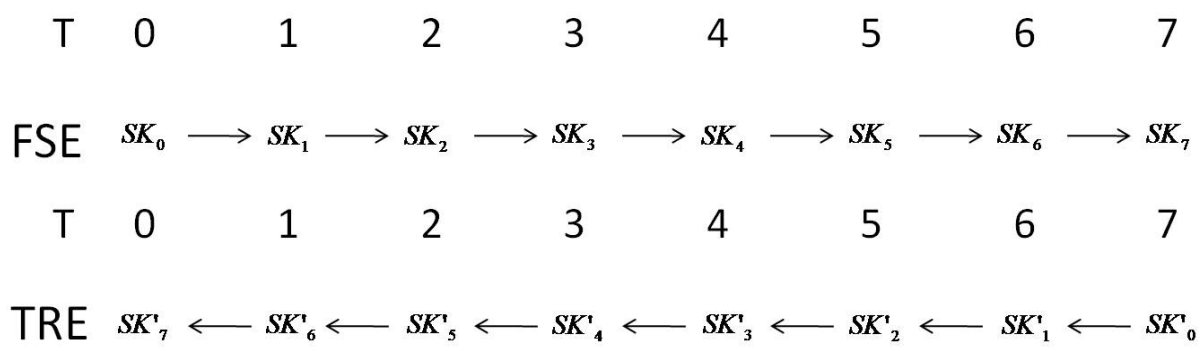

図 3.1: FSE を用いた TRE の構成の概念図

定理 3.1. もし *FSE* 方式 *fs-CCA(fs-CPA)* 安全ならば，*TRE* 方式は *IND-TR-CCA(IND-TR-CPA)* 安全である．

証明. 背理法を用いて証明する．もし TRE 方式に対して IND-TR-CCA をアドバンテージ $Adv_{A'}^{IND-TR-CCA}(k)$ で破る攻撃者 A' が存在するならば，FSE 方式に対して fs-CCA をアドバンテージ $\frac{1}{N} \cdot Adv_{A'}^{IND-TR-CCA}(k)$ で破る攻撃者 A が存在することを示す．我々は攻撃者 A を次のように定義する．

1. チャレンジャー C は $\mathtt{FSE.Setup}(1^k, N)$ を起動して (PK, SK_0) を得る．PK は攻撃者 A に与えられる．攻撃者 A は A' を起動して，PK を与える．
2. 攻撃者 A はランダムに時間 $j \in [0, N-1]$ を選び，j が A' のチャレンジの時にクエリされる時間 t_r であると予測する．A は break-in$(N-j)$ をクエリして，SK_{N-j} を得る．ただし，$j = 0$ ならば break-in クエリをしない．A' が鍵生成オラクルに t_i をクエリしたとき，もし $j < t_i$ ならば A はランダムビットを出力して停止する．さもなくば A は適切な $\{t_i\}$ に対応した秘密鍵 SK'_{t_i} を計算して，それらを A' に与える．
3. A' が復号オラクルに $c_h = (t_h, CT_h)$ をクエリしたとき，A は復号オラクルに $(N - t_h - 1, c_h)$ の組をクエリし， その結果を A' に返す．
4. A' が challenge(m_0, m_1, t_r) をクエリしたとき，もし $t_r \neq j$ ならば A はランダムビットを出力して停止する．さもなくば A は $CT^* \leftarrow$challenge $(m_0, m_1, N - t_r - 1)$ を計算して暗号文 $c^* = (t_r, CT^*)$ を A' に与える．
5. A' が鍵生成オラクルに t_i をクエリしたとき，もし $j < t_i$ ならば A はランダムビットを出力して停止する．さもなくば A は適切な t_i に対応した秘密鍵 SK'_{t_i} を計算して，それらを A' に与える．
6. A' が復号オラクルに (t_h, CT_h) をクエリしたとき，A は復号オラクルに $(N - t_h - 1, CT_h)$ をクエリし， その結果を A' に返す．
7. A' が b' を出力した時，A も b' を出力して停止する．

もし $t_r = j$ ならば A は A' の IND-TR-CPA ゲームを完全にシミュレートしている．さもなくば A はランダムビットを出力する．したがって $Pr[b = b' | t_r = j] = Adv_{A'}^{IND-TR-CCA}(k) + \frac{1}{2}$ かつ $Pr[b = b' | t_r \neq j] = \frac{1}{2}$ である．だから攻撃者 A が fs-CCA ゲームにおいてアドバンテージ $Adv_A^{fs-CCA}(k)$ は次の通りである．

$$
\begin{aligned}
Adv_A^{fs-CCA}(k) &= |Pr[b = b'] - \frac{1}{2}| \\
&= |Pr[t_r = j]Pr[b = b' | t_r = j] \\
&\quad + Pr[t_r \neq j]Pr[b = b' | t_r \neq j] - \frac{1}{2}| \\
Pr[t_r = j] &= \frac{1}{N} \\
Adv_A^{fs-CCA}(k) &= |\frac{1}{N} \cdot Pr[b = b' | t_r = j] - \frac{1}{2}(\frac{1}{N})| \\
&= \frac{1}{N} \cdot Adv_{A'}^{IND-TR-CCA}(k)
\end{aligned}
$$

つまり $Adv_{A'}^{IND-TR-CCA}(k)$ が無視できない値ならば FSE の攻撃者のアドバンテージ $Adv_A^{fs-CCA}(k)$ も無視できない値となり矛盾が生じる．定理 3.1 が成立する．もし FSE 方式が fs-CPA の意味で安全ならば，TRE 方式は IND-TR-CPA の意味で安全であることも上記の証明の復号オラクルを除くだけで証明可能である．

□

4 比較とまとめ

本稿の方式と既存方式の性能を比較したものを表 1 に示す。表 1 において，CPA 安全な方式と CCA 安全な方式ではオーダーで見たときの性能に違いがないためまとめて記述してある．提案方式の構成に用いる FSE は [5] の HIBE に [7] の構成方法を用いて得た FSE 方式とする．CCA 安全性について

表 1: TRE の既存研究との性能比較

方式	暗号文サイズ	秘密鍵サイズ	TRE.Enc の計算コスト	TRE.Dec の計算コスト
本稿の TRE	$2\|g\|$	$O((\log T)^2)$	$[0, O(\frac{\log T}{\log(\log T)+\log(\log k)}]$	$[2,\ 0\ ]$
[18] に [16] を用いた TRE	$O(\log T)\|g\|$	$O(\log T)$	$[0, O(\log T)]$	$[O(\log T), O(\log T)]$
[18] に [2] を用いた TRE	$3\|g\|$	$O((\log T)^2)$	$[0, O(\frac{\log T}{\log(\log T)+\log(\log k)})]$	$[3, O(\frac{(\log T)^2}{2\log(\log T)+\log(\log k)})]$
[2] を用いた TRE	$3\|g\|$	$O(T^2)$	$[0, O(\frac{T}{\log T+\log(\log k)})]$	$[3, O(\frac{T^2}{2\log T+\log(\log k)}]$

T はシステムが扱える合計時間
g は symmetric pairing が可能な群の要素の大きさ
[a,b] は計算コスト．それぞれ a はペアリング，b は [3] のアルゴリズムにおける指数乗の回数で正規化した値とする．[3] のアルゴリズムにより g_0^x の計算量は $O(\log k)$,$g_1^{a_1} \cdot g_2^{a_2} \cdot .... \cdot g_q^{a_q}$ の計算量は $\frac{q \cdot \log k}{\log q+\log(\log k)}$ と表される．ただし $g_0, g_1, ..., g_q$ は群の元であり，k はセキュリティパラメータ.

は，IND-CPA 安全な HIBE に BK 変換 [6] を用いて IND-CCA 安全な HIBE を得て，その方式に [7] のテクニックを用いて fs-CCA 安全な FSE を構成し本稿の提案方式に使用し，その性能は表 1 の 1 段目に示す.

表 1 の 2 段目は [18] で提案されているように，[4] の大小比較の回路の構成方法を [16] の関数暗号に適用して得た TRE である．3 段目は [4] の大小比較回路の構成方法を鍵ポリシー型属性ベース暗号 (key-Policy Attribute-Based Encryption, KP-ABE) に適用して得た TRE の性能評価である．本稿では KP-ABE として， 暗号文サイズが属性数に依存しない方式 [2] を採用した．その他 KP-ABE を用いた TRE の構成方法として，属性集合を $\{t_0, \ldots, T-1\}$, 時刻 α に対する時刻鍵をアクセス構造 $\{t_1 \vee \cdots \vee t_\alpha\}$ とし，属性 $\{t_r\}$ に対する KP-ABE の暗号文を時刻 t_r における TRE 暗号文とすることが考えられる．本構成法を KP-ABE[2] に提要した TRE の性能評価を 4 段目に示す.

表 1 より本稿の方式は他の方式に比べ，暗号文サイズと Dec の計算コストが小さく，Enc の計算コストも少なくとも他の方式と同等の性能であることがわかる．また，2 段目の方式と比べた場合, 秘密鍵サイズは大きいが他のサイズとコストはいずれも小さい.

本稿では FSE から時刻鍵の紛失の問題を解決する TRE の構成方法について，その厳密な定義と安全性の証明を与え，TRE と既存研究との比較を行いこの構成の優位性を示した．なお [18] では本稿で取り上げた時刻鍵の更新以外にも多くの機能を提案しており，それらの機能が FSE から構成できるかどうかの精査を今後の課題とする.

参考文献

[1] S. Agrawal, D. Boneh, and X. Boyen. Lattice basis delegation in fixed dimension and shorter-ciphertext hierarchical ibe. *Advances in Cryptology–CRYPTO 2010*, pages 98–115, 2010.

[2] N. Attrapadung, B. Libert, and E. De Panafieu. Expressive key-policy attribute-based encryption with constant-size ciphertexts. *Public Key Cryptography–PKC 2011*, pages 90–108, 2011.

[3] D.J. Bernstein. Pippenger's exponentiation algorithm. *Preprint. Available from* `http://cr.yp.to/papers.html`.

[4] J. Bethencourt, A. Sahai, and B. Waters. Ciphertext-policy attribute-based encryption. 2007.

[5] D. Boneh, X. Boyen, and E.J. Goh. Hierarchical identity based encryption with constant size ciphertext. *Advances in Cryptology–EUROCRYPT 2005*, pages 440–456, 2005.

[6] D. Boneh and J. Katz. Improved efficiency for cca-secure cryptosystems built using identity-based encryption. *Topics in Cryptology–CT-RSA 2005*, pages 87–103, 2005.

[7] R. Canetti, S. Halevi, and J. Katz. A forward-secure public-key encryption scheme. *Advances in Cryptology-Eurocrypt 2003*, pages 646–646, 2003.

[8] K. Chalkias and G. Stephanides. Timed release cryptography from bilinear pairings using hash chains. In *Communications and Multimedia Security*, pages 130–140. Springer, 2006.

[9] A.C.F. Chan and I.F. Blake. Scalable, server-passive, user-anonymous timed release cryptography. In *Distributed Computing Systems, 2005. ICDCS 2005. Proceedings. 25th IEEE International Conference on*, pages 504–513. IEEE, 2005.

[10] A. Dent and Q. Tang. Revisiting the security model for timed-release encryption with pre-open capability. *Information Security*, pages 158–174, 2007.

[11] C. Gentry, C. Peikert, and V. Vaikuntanathan. Trapdoors for hard lattices and new cryptographic constructions. In *Proceedings of the 40th annual ACM symposium on Theory of computing*, pages 197–206. ACM, 2008.

[12] N.M. Haller. The s/key$^{\text{tm}}$ one-time password system.

[13] T. Matsuda, Y. Nakai, and K. Matsuura. Efficient generic constructions of timed-release encryption with pre-open capability. *Pairing-Based Cryptography-Pairing 2010*, pages 225–245, 2010.

[14] T. May. Time-release crypto. *Manuscript, available at* `http://www.cyphernet.org/cyphernomicon/chapter14/14.5.html`, 1993.

[15] Y. Nakai, T. Matsuda, W. Kitada, and K. Matsuura. A generic construction of timed-release encryption with pre-open capability. *Advances in Information and Computer Security*, pages 53–70, 2009.

[16] T. Okamoto and K. Takashima. Fully secure functional encryption with general relations from the decisional linear assumption. *Advances in Cryptology–CRYPTO 2010*, pages 191–208, 2010.

[17] 岡本 義明 斎藤 泰一. 復元信号を用いた公開鍵型 timed-release 暗号. *CSS*, 2008.

[18] 藤岡 淳 星野 文学. 関数型暗号の応用: 検証可能時限式暗号. *SCIS*, 2011.