選択暗号文攻撃に対して安全な鍵失効機能付き ID ベース暗号

石田 優 †　　渡邉 洋平 †　　四方 順司 †

† 横浜国立大学大学院環境情報学府/研究院
240-8501 神奈川県横浜市保土ヶ谷区常盤台 79-7
{ishida-yuu-xg, watanabe-yohei-xs}@ynu.jp, shikata@ynu.ac.jp

あらまし 近年，鍵失効機能付き ID ベース暗号（RIBE）が数多く研究されている．PKC2013 にて，Seo，Emura は選択平文攻撃に対して安全な（CPA 安全な）復号鍵漏洩耐性を持つ RIBE を初めて提案した．彼らの CPA 安全な RIBE の構成は，既存の CPA 安全な IBE の構成を拡張することで達成している．さらに，構成内の CPA 安全な IBE の部分を階層的 IBE（HIBE）に拡張し，ワンタイム署名と組み合わせることにより選択暗号文攻撃に対して安全な（CCA 安全な）RIBE が構成できるであろうと言及されているが，厳密には示されていない．本稿では，上記のアプローチで CCA 安全な RIBE を実現するにあたり，内在する HIBE の構成の仕方が重要であることを示し，適切に CCA 安全な RIBE を構成するための方法を示す．

Revocable Identity-based Encryption Secure against Chosen Ciphertext Attack

Yuu Ishida†　　Yohei Watanabe†　　Junji Shikata†

†Graduate School of Environment and Information Sciences, Yokohama National University,
79-7 Tokiwadai, Hodogaya-ku, Yokohama 240-8501, Japan
{ishida-yuu-xg, watanabe-yohei-xs}@ynu.jp, shikata@ynu.ac.jp

Abstract Recently, revocable IBE (RIBE) has been many studied. In PKC 2013, Seo and Emura first proposed RIBE with decryption key leakage resistance. Their RIBE scheme is secure against chosen plaintext attacks (CPA), since it is based on a specific CPA-secure IBE scheme. Further, although no details were given, they mentioned the following approach to extend the RIBE scheme to an RIBE scheme secure against chosen ciphertext attacks (CCA): (1) The underlying CPA-secure IBE can be extended to CPA-secure hirarchical IBE (HIBE), and then; (2) CCA-secure RIBE can be realized from the CPA-secure HIBE with a combinaton of one-time signitures. In this paper, we show how CCA-secure RIBE should be constructed based on the above approach, and consequently, we first propose a concrete scheme of CCA-secure RIBE with decryption key leakage resistance.

1　はじめに

ID ベース暗号（Identity-Based Encryption, IBE）[1] は，ID を公開鍵として利用可能な公開鍵暗号（Public Key Encryption, PKE）の一種である．IBE に関してこれまでに盛んな研究が進められており，重要な暗号要素技術の一つに位置付けられている．IBE では，通常の PKE に対する公開鍵暗号基盤（Public Key Infrastructure, PKI）による認証を必要としない代わ

りに，PKI における鍵失効機能を PKI なしで実現する必要があるが，[1] に代表される通常の IBE で鍵失効機能を実現するためには，各ユーザの秘密鍵を生成する機関（Key Generation Center, KGC）に大きな負担がかかることが知られている．しかし，ユーザを動的に管理するためにも，鍵失効機能は PKE においてだけでなく IBE においても非常に重要な機能である．

鍵失効機能をより効率的に実現できる IBE として，鍵失効機能付き ID ベース暗号（Revocable Identity-Based Encryption, RIBE）が Boldyreva らによって提案された [4]．RIBE では，通常の IBE 同様，各 ID に対して秘密鍵 sk_{ID} が KGC から発行される．更に，KGC は各期間 T において鍵更新用の情報 ku_T を放送する．期間 T で削除されていないユーザのみ，sk_{ID} と ku_T から復号鍵 $dk_{ID,T}$ を計算することができる．

関連研究． これまでに RIBE に関する様々な方式が提案されてきた [4, 5, 7]. Boneh，Flanklin の IBE [1] における KGC の鍵失効時の計算コストは，各期間ごとに $O(N-R)$（N，R はそれぞれ全ユーザ数と削除ユーザ数）であるのに対し，Boldyreva らの RIBE は計算コストが各期間ごとに $O(R\log(N/R))$ まで削減に成功している．これは放送暗号のフレームワークのひとつである Complete Subtree（CS）法 [6] を用いていることによる．また，Boldyreva らの方式 [4] は Selective-ID 安全と呼ばれる弱い安全性のみ実現していたが，Libert と Vergnaud により，より強い安全性である Adaptive-ID 安全な RIBE も提案されている [5]．更に，Seo と Emura は復号鍵漏洩耐性をもつ Adaptive-ID 安全な RIBE を提案している [7]．[7] における復号鍵漏洩耐性を考慮した安全性定義は，上記の [4, 5] における定義よりも強い安全性定義となっており，更に，彼らの提案した RIBE[7] は，同じ Adaptive-ID 安全性をみたす [5] の RIBE に比べて鍵長等の面で，より効率的なものとなっている．また，上記の方式はいずれも選択平文攻撃に対して安全（CPA 安全）な方式である．

本論文の貢献． 先に述べたように，[7] において，復号鍵漏洩耐性を持ち，Adaptive-ID 安全かつ CPA 安全な RIBE が初めて示された．彼らの CPA 安全な RIBE の構成は，既存の CPA 安全な IBE [8] の構成を拡張することで達成している．さらに，構成内の CPA 安全な IBE の部分を 2 階層の階層的 IBE（HIBE）に拡張し，ワンタイム署名（One-Time Signature, OTS）と組み合わせることにより，選択暗号文攻撃に対して安全な（CCA 安全な）RIBE が構成できるであろうと言及されているが，厳密には示されていない．本稿では，まず上記のアプローチで CCA 安全な RIBE を実現するにあたり，内在する HIBE の構成の仕方が重要であることを示す．すなわち，内部の CPA 安全な 2 階層の HIBE から CCA 安全な RIBE に拡張する際に，その HIBE の階層構造を工夫しないと，CCA 安全かつ復号鍵漏洩耐性をもつ RIBE が実現できないことを示す．更に，適切に階層構造を工夫することによって，CCA 安全な RIBE を構成できることを示す．

2 準備

本節では提案する RIBE の構成，またその安全性証明に用いる要素技術について述べる．

2.1 KUNode アルゴリズム

KGC の計算コストを抑えるために，既存の RIBE [4, 5, 7]，そして本方式においても，KUNode アルゴリズムを利用している．

定義 1 (KUNode アルゴリズム). 二分木 BT，削除リスト RL，期間 T を入力とし，ノードの集合 Y を出力する．以下，葉ではないノード x に対し，x_{left} を左の子ノード，x_{right} を右の子ノードとする．各ユーザは葉ノードに割り当てられる．(葉ノード η に割り当てられた）ユーザが期間 T で削除される場合，$(\eta, T) \in RL$ とする．η から root までの最短経路上に現れるノードの集合を $Path(\eta)$ と記述する．

$KUNode(BT, RL, T)$:
$X, Y \leftarrow \emptyset$;

$\forall(\eta_i, T_i) \in RL$;

If $T_i \leq T$ then add $Path(\eta_i)$ to X;

$\forall x \in X$;

If $x_{left} \notin X$ then add x_{left} to Y;

If $x_{right} \notin X$ then add x_{right} to Y;

If $Y = \emptyset$ then add root to Y;

Return Y.

2.2 階層的 ID ベース暗号 (HIBE)

ℓ-level HIBE とは，ユーザ（ID）を階層的に配置し，自身の下の階層の ID に対して秘密鍵を生成できる IBE である．ℓ-level HIBE は以下の 4 つのアルゴリズムからなる．以下では，$\mathcal{M}$ を平文空間，$\mathcal{ID}$ を ID 空間とし，$ID_i = (id_1, \ldots, id_i) \in \prod_{j=1}^{i} \mathcal{ID}$ $(1 \leq i \leq \ell)$ を ID ベクトルとする．ID_0 を空の文字列とする．

- $(PK, SK_{ID_0}) \leftarrow HIBE.Setup(\lambda)$．セキュリティパラメータ λ を入力し，公開鍵 PK と初期秘密鍵 SK_{ID_0} を出力する．以降 SK_{ID_0} を SK とかく．
- $SK_{ID_i} \leftarrow HIBE.Der\left(ID_{i-1}, SK_{ID_{i-1}}, id_i\right)$．$ID_{i-1}$，$ID_{i-1}$ の秘密鍵 $SK_{ID_{i-1}}$，id_i を入力として，下の階層の秘密鍵 SK_{ID_i} を出力する．ただし，$ID_i = (ID_{i-1}, id_i)$ とする．
- $c \leftarrow HIBE.Enc(PK, ID_i, m)$．$PK$，$ID_i$，平文 $m \in \mathcal{M}$ を入力とし，暗号文 c を出力する．
- m or $\perp \leftarrow HIBE.Dec(SK_{ID_i}, ID_i, c)$．$ID_i$ の秘密鍵 SK_{ID_i}，ID_i，c を入力として，m または $\perp$ を出力する．

全ての $(PK, SK) \leftarrow HIBE.Setup(\lambda)$，$ID_i \in \prod_{j=1}^{i} \mathcal{ID}$，$SK_{ID_i} \leftarrow HIBE.Der(SK_{ID_{i-1}}, ID_{i-1}, id_i)$, $m \in \mathcal{M}$ に対して，$m \leftarrow HIBE.Dec(SK_{ID_i}, ID_i, HIBE.Enc(PK, ID_i, m))$ をみたすとする．

ℓ-level HIBE の安全性は攻撃者 A とチャレンジャー CH 間の以下のゲームを用いて定義される．

Step 1. CH は *HIBE.Setup* を実行し PK を A に渡す．

Step 2. 攻撃者は適応的に ID に対応する秘密鍵を得ることができる．A は任意の ID を鍵導出オラクルに問い合わせ，問い合わせた ID に対応する SK_{ID} を受け取る．

Step 3. A は 2 つの長さの等しい平文 m_0，m_1，そしてターゲット ID ベクトル ID^* を CH に送る．CH はランダムに $b \in \{0, 1\}$ を選び，$c^* = HIBE.Enc(PK, ID^*, m_b)$ を A に送る．

Step 4. A は Step 3 と同様にオラクルへの問い合わせを繰り返す．ただし，ID^* や ID^* が接頭辞であるような ID ベクトルについて問い合わせることはできない．

Step 5. A は b の推測値 b' を出力する．$b' = b$ であれば攻撃成功となる．

定義 2 (IND-ID-CPA). すべての多項式時間アルゴリズム A に対し，$Adv_{A,HIBE}^{IND-ID-CPA} = |\Pr[b' = b] - 1/2|$ が λ 無視できるほど小さい時，HIBE は IND-ID-CPA 安全であるという．

2.3 ワンタイム署名 (OTS)

ワンタイム署名（One-Time Signature, OTS）は以下 3 つのアルゴリズムからなる．$\mathcal{M}$ をメッセージ空間とする．

- $(vk, sk) \leftarrow \partial(\lambda)$．セキュリティパラメータ λ を入力として，検証鍵と署名鍵のペア (vk, sk) を出力する．
- $s \leftarrow Sign(sk, m)$．sk，メッセージ $m \in \mathcal{M}$ を入力として，署名 σ を出力する．
- 1 or 0 $\leftarrow Verfy(vk, m, \sigma)$．$vk$，$m$，$\sigma$ を入力として，1（受理）または 0（拒否）を出力する．

全ての $(vk, sk) \leftarrow \partial(\lambda)$，$m \in \mathcal{M}$ に対して，$Verfy(vk, m, Sign(sk, m)) = 1$ をみたすとする．

OTS の安全性は攻撃者 A とチャレンジャー CH 間の以下のゲームを用いて定義される．

Step 1. CH は ∂ を実行し vk を A に渡し，sk を保持しておく．

Step 2. A は任意のメッセージ m を署名オラクルに高々1 回問い合わせ，問い合わせた m に対応する署名 σ を受け取る．

Step 3. Aは正当なメッセージと署名の組$(m^*, \sigma^*) \neq (m, \sigma)$を生成することができれば攻撃成功となる.

定義 3 (OT-sEUF-CMA). すべての多項式時間アルゴリズムAに対し，$Adv^{OT-sEUF-CMA}_{A,OTS} = \Pr\big[Verfy(vk, m^*, \sigma^*) = 1 \wedge (m^*, \sigma^*) \neq (m, \sigma)\big]$が$\lambda$に関して無視できるほど小さい時，OTSはOT-sEUF-CMA安全であるという.

3 復号鍵漏洩耐性を持つRIBE

本節では, 復号鍵漏洩耐性を考慮したRIBE[7]のモデル，安全性定義を与える[1]．以下, $\mathcal{M}$を平文空間, $\mathcal{ID}$をID空間, $\mathcal{T}$を期間空間とする.

3.1 モデル

セットアップアルゴリズム *Setup*: セキュリティパラメータλおよび最大ユーザ数Nを入力とし, マスター公開鍵mpk, マスター秘密鍵msk, 初期削除リスト$RL = \phi$, およびステートstを出力する.

秘密鍵生成アルゴリズム *PKG*: $mpk, msk, ID \in \mathcal{ID}$, および$st$を入力とし, 秘密鍵$sk_{ID}$を出力し, stを更新する.

鍵更新アルゴリズム *KeyUp*: mpk, msk, 復号鍵更新期間$T \in \mathcal{T}$, 現在の削除リストRL, およびstを入力とし, 鍵更新用情報ku_Tを出力する.

復号鍵生成アルゴリズム *DKG*: mpk, sk_{ID}, およびku_Tを入力とし, 復号鍵$dk_{ID,T}$を出力する. もしIDが削除されていた場合, $\perp$を出力する.

暗号化アルゴリズム *Enc*: mpk, $ID \in \mathcal{ID}, T \in \mathcal{T}$, および平文$M \in \mathcal{M}$を入力とし, 暗号文CTを出力する.

復号アルゴリズム *Dec*: mpk, $dk_{ID,T}$, およびCTを入力とし, Mまたは$\perp$を出力する.

鍵失効アルゴリズム *Revoke*: 削除する$ID \in \mathcal{ID}$, $T \in \mathcal{T}$, RL, およびstを入力とし, RLを更新する.

すべての$(mpk, msk) \leftarrow Setup(\lambda), M \in \mathcal{M}$, stおよびRLに対し，もし$ID \in \mathcal{ID}$が期間$T \in \mathcal{T}$で削除されていない場合，$(sk_{ID}, st) \leftarrow PKG(mpk, msk, ID, st)$，$ku_T \leftarrow KeyUp(mpk, msk, T, RL, st)$，および$dk_{ID,T} \leftarrow DKG(mpk, sk_{ID}, ku_T)$に対し，$Dec(mpk, dk_{ID,T}, Enc(mpk, ID, T, M)) = M$が成り立つ.

3.2 安全性定義

RIBEの安全性は攻撃者AとチャレンジャーCH間の以下のゲームを用いて定義される.
Step 1. CHは*Setup*を実行しmpkをAに渡す.
Step 2. 制限$(*)$のもとAはオラクルアクセスを行う.
Step 3. Aは2つの長さの等しい平文M_0，M_1とターゲットであるID^*, T^*をCHに送る. CHはランダムに$b \in \{0, 1\}$を選び$CT^* = Enc(mpk, ID^*, T^*, M_b)$を$A$に送る.
Step 4. AはStep 2と同様にオラクルへの問い合わせを繰り返す.
Step 5. Aはbの推測値b'を出力する．$b' = b$であれば攻撃成功となる.
Aが利用可能なオラクルを$PKG(\cdot)$，$KeyUp(\cdot)$，$Revoke(\cdot, \cdot)$，$DKG(\cdot, \cdot)$とし，各オラクルは以下で定義される.

- $PKG(\cdot)$: $ID \in \mathcal{ID}$に対して，$sk_{ID} \leftarrow PKG(mpk, msk, ID, st)$を実行し，$sk_{ID}$を返す.
- $KeyUp(\cdot)$: $T \in \mathcal{T}$に対し，$ku_T \leftarrow KeyUp(mpk, msk, T, RL, st)$を実行し，$ku_T$を返す.
- $Revoke(\cdot, \cdot)$: $ID \in \mathcal{ID}$，$T \in \mathcal{T}$に対し，$RL \leftarrow Revoke(mpk, ID, T, RL, st)$を実行し，更新後の$RL$を返す.
- $DKG(\cdot, \cdot)$: $ID \in \mathcal{ID}$，$T \in \mathcal{T}$に対し，$sk_{ID} \leftarrow PKG(mpk, msk, ID, st)$，および

[1]以降，単にRIBEと呼ぶときは復号鍵漏洩耐性をもつRIBEを指すものとする.

$dk_{ID,T} \leftarrow DKG(mpk, sk_{ID}, ku_T)$ を実行し，$dk_{ID,T}$ を返す．

- $Dec(\cdot,\cdot,\cdot)$ $ID \in \mathcal{ID}$，$T \in \mathcal{T}$, CT に対し，$sk_{ID} \leftarrow PKG(mpk, msk, ID, st)$，$dk_{ID,T} \leftarrow DKG(mpk, sk_{ID}, ku_T)$，$M \leftarrow Dec(mpk, dk_{ID,T}, CT)$ を実行し，M を返す．

制限 $(*)$：
攻撃者 A は上記オラクルに以下の制限下でのアクセスが行われる．

1. すべての既に発行されたクエリ期間と同じかそれ以降の期間に対して $KeyUp(\cdot)$ および $Revoke(\cdot,\cdot)$ に対するクエリが可能．
2. もし期間 T で $KeyUp(\cdot)$ がクエリされていた場合，$Revoke(\cdot,\cdot)$ に対するクエリはできない．
3. もし $PKG(ID^*)$ に対するクエリがされていた場合，期間 $T \leq T^*$ に対し $Revoke(ID^*, T)$ に対するクエリがされていなければいけない．
4. 期間 T で $KeyUp(\cdot)$ する前に $DKG(\cdot,\cdot)$ および $Dec(\cdot,\cdot,\cdot)$ はクエリできない．
5. $DKG(ID^*, T^*)$ はクエリできない．
6. $Dec(ID^*, T^*, CT^*)$ はクエリできない．

定義 4 (IND-RID-CCA). すべての多項式時間アルゴリズム A に対し，$Adv^{IND-RID-CCA}_{A,RIBE} = |\Pr[b' = b] - 1/2|$ が無視できるほど小さい時，RIBE は IND-RID-CCA 安全であるという．

なお，上記ゲームから復号オラクルを取り除くと IND-RID-CPA 安全性の定義と一致する．この意味で [7] からの IND-RID-CCA 安全性への拡張とみなせる．

4 CCA 安全な RIBE の構成法

本節では，CCA 安全な RIBE の構成法を提案する．本方式は，[7] で述べられている CCA 安全な方式への拡張アプローチを基にしている．以下では，まず [7] で述べられている CCA 安全な方式への拡張アプローチについてをまとめ，その後，本方式がとった具体的なアプローチについてまとめる．

[7] で述べられている拡張アプローチ. [7] における構成法は，[8] における CPA 安全な IBE の構成法（以下，Waters's IBE とよぶ）をベースとしており，その CPA 安全性も Waters's IBE に帰着させている．Waters's IBE は比較的簡単に任意の階層の CPA 安全な HIBE に拡張でき [8]，また CPA 安全な 2-level HIBE と OTS から CCA 安全な IBE が構成できることが知られている [2] ことから， 構成法内の Waters's IBE に相当する部分を 2 階層の HIBE に拡張し（これを 2-level Waters's HIBE とよぶ），OTS と組み合わせることで，CCA 安全性な RIBE が構成できるだろうと言及されている．

本方式における具体的アプローチ. 上記アプローチを基に CCA 安全な RIBE を構成するためには，実は内部で拡張した HIBE の階層構造の作り方が重要である．本方式ではその点を考慮し，CCA 安全な RIBE を提案する．一見上記のアプローチによって自明に CCA 安全な RIBE が構成できそうに思えるが，実は単純には構成することができない．工夫が必要な理由として，復号鍵生成アルゴリズム DKG の存在が挙げられる．[2] の CCA 安全な IBE を構成するテクニックは以下の通りである: 暗号化の際に，OTS の鍵ペアを生成し，その検証鍵 vk を 2-level HIBE の 2 階層目の ID とみなし，すなわち (ID, vk) を暗号化鍵として暗号化する．更に，暗号文に対する署名を生成し，その検証鍵 vk も暗号文の一部とする．そして復号の際に $sk_{(ID,vk)}$ を生成し，その鍵で復号する．しかし，DKG は入力に暗号文がとれないため，$sk_{(ID,vk)}$ が生成できず，従って $dk_{(ID,vk),T}$ も生成できない．従って，$dk_{ID,T} = (sk_{ID}, ku_T)$ として出力し，復号アルゴリズム Dec で $sk_{ID} \rightarrow sk_{(ID,vk)} \rightarrow dk_{(ID,vk),T}$ の手順で復号鍵を生成し，復号を行うことになる．しかし，これでは復号鍵漏洩耐性をみたさないことは明らかである（IND-RID-CCA ゲームにおいて $dk_{ID^*,T}$，dk_{ID,T^*} を DKG オラクルにクエリすることで sk_{ID^*}，ku_{T^*} を入手することができるため）．従って，[7] で述べられているア

プローチから CCA 安全な RIBE を自明には構成できないことがわかる[2]．本構成法のアイディアとして，上記の流れを，DKG アルゴリズムで先に $dk_{ID,T}$ を生成しておき，Dec アルゴリズムで $dk_{ID,T}$ から $dk_{(ID,vk),T}$ を生成するように修正することで，CCA 安全性を達成する．

以下，本方式の構成法を示す．$ID = (b_1, \ldots, b_n) \in \{0,1\}^n$，$T \in \mathbb{Z}_p$，$\{u'_1, u'_2, u_1^{(1)}, \ldots, u_n^{(1)}, u_1^{(2)}, \ldots, u_n^{(2)}, v', v\} \in \mathbb{G}$ に対して，$F_W^{(t)}(ID) = u'_1 \prod_{i=1}^{n} u_i^{(t)^{b_i}}$，$F_{BB}(T) = v'v^T$ とする．

- $Setup(\lambda)$:
  $\alpha \in \mathbb{Z}_p$，$\{g, g_2, u'_1, u'_2, u_1^{(1)}, \ldots, u_n^{(1)}, u_1^{(2)}, \ldots, u_n^{(2)}, v', v\} \in \mathbb{G}$ をランダムに選択し，$g_1 = g^\alpha$ とする．$mpk = \{g, g_1, g_2, u'_1, u'_2, u_1^{(1)}, \ldots, u_n^{(1)}, u_1^{(2)}, \ldots, u_n^{(2)}, v', v\}$，$msk = g_2^\alpha$，$RL = \phi$，$st = BT$ とする．なお BT は葉ノードが N 個の Binary Tree である．

- $PKG(mpk, msk, ID)$:
  $ID = (b_1, \cdots, b_n) \in \{0,1\}^n$ とする．BT から割り当てられてない葉 η をランダムに選ぶ．ID を η に割り当てる．$\theta \in Path(\eta)$ に対し，次の処理を行う．

  1. g_θ が定義されていたらその値を使用する．そうでなければ，$g_\theta \in \mathbb{G}$ をランダムに選び，ノード θ に $(g_\theta, \tilde{g}_\theta = g_2/g_\theta)$ を保存する．
  2. $r_\theta \in \mathbb{Z}_p$ をランダムに選び，$(D_{\theta,0}, D_{\theta,1}) = (g_\theta^\alpha F_W^{(1)}(ID)^{r_\theta}, g^{r_\theta})$ を計算する．

  $SK_{ID} = (\theta, D_{\theta,0}, D_{\theta,1})_{\theta \in Path(\eta)}$ を返す．

- $KeyUp(mpk, msk, T, RL, st)$:
  $st = BT$ とし，ノード $\theta \in KUNode(BT, RL, T)$ に対して以下の処理を行う．

  1. $\tilde{g}_\theta$ を取得する（$\tilde{g}_\theta$ は常に PKG で事前に定義されている）．
  2. $s_\theta \in \mathbb{Z}_p$ をランダムに選択する．
  3. $(\tilde{D}_{\theta,0}, \tilde{D}_\theta) = (\tilde{g}_\theta^\alpha F_{BB}(T)^{s_\theta}, g^{s_\theta})$

  $ku_\tau = (\theta, \tilde{D}_{\theta,0}, \tilde{D}_{\theta,1})_{\theta \in KUNode(BT,RL,T)}$ を返す．

- $DKG(mpk, SK_{ID}, ku_\tau)$:
  $SK_{ID} = \{(\theta, D_{\theta,0}, D_{\theta,1})\}_{\theta \in I}$，$ku_\tau = \{(\theta, \tilde{D}_{\theta,0}, \tilde{D}_{\theta,1})\}_{\theta \in J}$ とする．$I \cap J = \emptyset$ なら $\perp$ を返し，そうでないなら $\theta \in I \cap J$ をひとつ選び，$r, s \in \mathbb{Z}_p$ をランダムに選択し，$dk_{ID,T} = (D_{\theta,0} \cdot \tilde{D}_{\theta,0} \cdot F_W^{(1)}(ID)^r \cdot F_{BB}(T)^s, D_{\theta,1} \cdot g^r, \tilde{D}_{\theta,1} \cdot g^s) = (g_2^\alpha F_W^{(1)}(ID)^{r+r_\theta} \cdot F_{BB}(T)^{s+s_\theta}, g^{r+r_\theta}, g^{s+s_\theta})$ を返す．

- $Enc(mpk, ID, T, M)$:
  $\partial(\lambda) \rightarrow (vk, sk)$ を実行する．$t \in \mathbb{Z}_p$ をランダムに選択し，$C = (M \cdot e(g_1, g_2)^t, g^{-t}, F_W^{(1)}(ID)^t, F_W^{(2)}(vk)^t, F_{BB}(T)^t)$ を計算する．さらに，$Sign(sk, C) \rightarrow \sigma$ を実行し，暗号文として $CT =< vk, C, \sigma >$ を返す．

- $Dec(mpk, dk_{ID,T}, CT)$:
  $dk_{ID,T} = (K_1, K_2, K_3)$ とする．$Verfy(C, \sigma) = 0$ ならば $\perp$ を返す．そうでなければ，$\tilde{r} \in \mathbb{Z}_p$ をランダムに選択し，$dk_{(ID,vk),T} = (K_1 \cdot F_W^{(2)}(vk)^{\tilde{r}} \cdot, K_2, g^{\tilde{r}}, K_3) = (g_2^\alpha F_W^{(1)}(ID)^{r+r_\theta} \cdot F_W^{(2)}(vk)^{\tilde{r}} \cdot F_{BB}(T)^{s+s_\theta}, g^{r+r_\theta}, g^{\tilde{r}}, g^{s+s_\theta})$ を計算する．$dk_{(ID,vk),T} = (D_1, D_2, D_3, D_4)$，$CT =< vk, C_0, C_1, C_2, C_3, C_4, \sigma >$ とし，$M = C_0 \cdot \prod_{i=1}^{4} e(C_i, D_i)$ を返す．

- $Revoke(mpk, ID, T, RL, st)$:
  η を ID が割り当てられた葉ノードとする．$RL \leftarrow RL \cup (\eta, T)$ によって revoke リストを更新し，更新された新しい RL を返す．

5 安全性証明

本節では，本構成法が IND-RID-CCA 安全性を満たすことを示す．元々[7] の RIBE は Waters's IBE [8] に対する帰着を示すことで安全性を示していた．本構成法の安全性も同様のアプローチで示す．具体的には，本構成法は 2-level Waters's HIBE の拡張であることと，[2] の証明手法を利用して証明する．まず，2-level Waters's HIBE [8] を下記に示す．

[2] DKG アルゴリズムは復号鍵漏洩耐性を考えるために必要不可欠なアルゴリズムであるため，[4] のように復号アルゴリズムの一部として考えることはできない．

- $(PK, SK) \leftarrow$ *HIBE.Setup*(λ): $\alpha \in \mathbb{Z}_p$, $\{g, g_2, u'_1, u'_2, u_1^{(1)}, \cdots, u_n^{(1)}, u_1^{(2)}, \ldots, u_n^{(2)}\} \in \mathbb{G}$ をランダムに選択し，$g_1 = g^\alpha$ とする．$PK = \{g, g_1, g_2, u'_1, u'_2, u_1^{(1)}, \ldots, u_n^{(1)}, u_1^{(2)}, \ldots, u_n^{(2)}\}$，$SK = g_2^\alpha$ を返す．

- $SK_{ID_i} \leftarrow$ *HIBE.Der*$(SK_{ID_{i-1}}, ID_{i-1}, id_i)$: $i = 1, 2$ に対して，$SK_{ID_{i-1}} := (D_1, \ldots, D_i)$ とする．$id_i = (b_1, \ldots, b_n) \in \{0,1\}^n$ とし，$\tilde{r} \in \mathbb{Z}_p$ をランダムに選択し，$SK_{ID_i} = (d_1 \cdot F_W^{(i)}(id_i)^{\tilde{r}}, D_2, \ldots, g^{\tilde{r}})$ を返す．

- $c \leftarrow$ *HIBE.Enc*(PK, ID_i, m): $i = 1, 2$ に対して，$t \in \mathbb{Z}_p$ をランダムに選択し，$C = (m \cdot e(g_1, g_2)^t, g^{-t}, F_W^{(1)}(id_1)^t, \ldots, F_W^{(i)}(id_i)^t)$ として返す．

- m or $\perp \leftarrow$ *HIBE.Dec*(SK_{ID}, ID_i, c): $i = 1, 2$ に対して，$c = (C_0, C_1, \ldots, C_{i+2})$，$SK_{ID_i} = (D_1, \ldots, D_{i+2})$ に対し，$m = C_0 \cdot \prod_{j=1}^{i+2} e(C_j, D_j)$ を返す．

上記の構成が IND-ID-CPA 安全性をみたすことは，Waters's IBE [8] の安全性証明と同様の流れで示すことが可能である．紙面の都合上，具体的な証明は省略する．

提案した RIBE の構成法の安全性に関して，以下の定理を示すことができる．

定理 1. 上記の 2-level Waters's HIBE が IND-ID-CPA 安全かつ OTS が OT-sEUF-CMA 安全ならば，提案する RIBE は IND-RID-CCA 安全である．

証明. IND-RID-CCA 安全性に対する攻撃者 A について，暗号文 (vk, C, σ) が $Verfy(vk, C, \sigma) = 1$ を満たす場合，この暗号文を正当な暗号文と呼ぶ．(vk^*, C^*, σ^*) を IND-RID-CCA ゲームにおけるチャレンジ暗号文としたとき，以下の事象を定義する．

- Forge: IND-RID-CCA ゲームにおいて，A が $ID = ID^*$ かつ $Verfy(vk, C, \sigma) = 1$ となるような $(ID, T, < vk, C, \sigma >)$ を復号オラクルにクエリする事象．

- Succ: IND-RID-CCA ゲームにおいて，A が勝利する（$b' = b$ となるような b' を出力する）事象．

このとき，

$$
\begin{aligned}
&Adv_{A,RIBE}^{IND-RID-CCA} \\
&= |\Pr[\mathsf{Succ}] - \frac{1}{2}| \\
&\leq |\Pr[\mathsf{Succ} \wedge \mathsf{Forge}] - \frac{1}{2}\Pr[\mathsf{Forge}]| \\
&\qquad + |\Pr[\mathsf{Succ} \wedge \overline{\mathsf{Forge}}] + \frac{1}{2}\Pr[\mathsf{Forge}] - \frac{1}{2}| \\
&\leq \frac{1}{2}\Pr[\mathsf{Forge}] + \\
&\qquad |\Pr[\mathsf{Succ} \wedge \overline{\mathsf{Forge}}] + \frac{1}{2}\Pr[\mathsf{Forge}] - \frac{1}{2}|.
\end{aligned}
$$

この時，以下の 2 つの補題が成立する．

補題 1. $\Pr[\mathsf{Forge}]$ は無視できるほど小さい．

証明. 事象 Forge が起こったと仮定すると，IND-RID-CCA 安全性を破る攻撃者 A を用いて OT-sUF-CMA 安全性を破る署名偽造者 F を構成することができることを示す．F はチャレンジャー CH より vk^* を受け取る．F は RIBE の攻撃者 A に対して IND-RID-CCA ゲームをシミュレーションするために，*Setup*(λ) を実行し，攻撃者 A に $mpk = \{g, g_1, g_2, u'_1, u'_2, u_1^{(1)}, \ldots, u_n^{(1)}, u_1^{(2)}, \ldots, u_n^{(2)}, v', v\}$ を与える．F は $msk = g_2^\alpha$ を持っているので，A の復号クエリにすべて答えることができる．IND-RID-CCA ゲームの Step 2 の時点で A が復号クエリ $(ID^*, T, < vk^*, C, \sigma >)$ を出力した場合，F は (C, σ) を偽造署名として出力する．そうでなければ，A が $ID^*, T^*, (M_0, M_1)$ を出力したときに，$b \in \{0,1\}$ をランダムに選び，$C^* = (M_b \cdot e(g_1, g_2)^t, g^{-t}, F_W^{(1)}(ID^*)^t, F_W^{(2)}(vk^*)^t, F_{BB}(T^*)^t)$ を計算して，これを署名オラクルに問い合わせて，C^* に対する署名 σ^* を得る．その後，A が Step 4 にて復号クエリ $(ID^*, T, < vk^*, C, \sigma >)$ を復号オラクルに質問してきた場合，$(C, \sigma) \neq (C^*, \sigma^*)$ が成立していなければならないので，F は (C, σ) を偽造署名として出力する．従って，事象 Forge が起こる確率は無視できることが証明できた．
□

補題 2. $\Pr[\mathsf{Succ} \wedge \overline{\mathsf{Forge}}] + \frac{1}{2}\Pr[\mathsf{Forge}] - \frac{1}{2}$ は無視できるほど小さい．

証明. RIBE の IND-RID-CCA 安全性を破る攻撃者 A を用いて，2-level Waters's HIBE の

IND-ID-CPA 安全性を破る攻撃者 B を構成する．基本的に [7] と同様の流れで証明可能であるが，差分として復号オラクルをシミュレートする必要がある．そこで本稿では紙面の都合上，復号オラクルのシミュレートについてのみ記す．また，攻撃者 A の振る舞いを (1) sk_{ID^*} をクエリできるが ID^* は T^* の前に削除されている場合，(2) sk_{ID^*} はクエリできないが $dk_{ID^*,T}$ はクエリできる場合に分けて考える．復号オラクルは以下のようにシミュレートされる．

まず，(1)(2) の両方の場合において，次のように構成する．A が復号オラクルに $(ID, T, < vk, C, \sigma >)$ をクエリしてきた時，$Verfy(vk, C, \sigma) = 0$ であれば $\perp$ を返し，そうでない場合は次のようにシミュレートする．

- $(ID, vk) = (ID^*, vk^*)$ であれば，終了してランダムなビットを出力する．
- $(ID, vk) \neq (ID^*, vk^*)$ の時は，(1)(2) で分けて考える．

(1) まず，[7] における証明と同様に，この時点で既に鍵更新オラクルに T に対するクエリがなされており，各ノード $\theta \in KUNode(BT, RL, T)$ には乱数 S_θ が割り当てられている．B は S_θ を用いて，この秘密鍵生成オラクルや鍵更新オラクルに対する答えを行っていることに留意されたい（詳細は [7] を参照）．

A から受け取ったクエリ $(ID, T, < vk, C, \sigma >)$ に対して，B は (ID, vk) を秘密鍵生成オラクルにクエリし，$sk_{(ID,vk)} = (g_2^\alpha F_W^{(1)}(ID)^r F_W^{(2)}(vk)^{\tilde{r}}, g^r, g^{\tilde{r}})$ を得る．いま，η を ID が割り当てられている葉ノードとする．$\theta \in Path(\eta)$ に対して，θ に S_θ が保存されていればその S_θ を用い，そうでなければ $S_\theta \in \mathbb{G}$ をランダムに選び，θ に保存する．$ku_T = (I_1, I_2)$，$sk_{(ID,vk)} = (d_1, d_2, d_3)$ とすると，$r_\theta, s \in \mathbb{Z}_p$ をランダムに選ぶことで，$dk_{(ID,vk),T} = (S_\theta \cdot d_1 \cdot I_1 \cdot F_W^{(1)}(ID)^{r_\theta} \cdot F_{BB}(T)^s, d_2 g^{r_\theta}, d_3, I_2 g^s)$ を計算できるので，正しい復号結果 M を返すことができる[3]．

[3]鍵更新オラクルにおいて，ターゲットのユーザ ID が割り当てられた葉ノードを η^* とすると，各ノード $\theta \in KUNode(BT, RL, T)$ に対して，$\theta \notin Path(\eta^*)$ か $\theta \in Path(\eta^*)$ かで ku_T の作り方が異なる．しかし，どちらの場合でも，このように計算することで正しい復号結果を得ることができる．

(2) (1) と同様に，B は (ID, vk) を秘密鍵生成オラクルにクエリし，$sk_{(ID,vk)} = (d_1, d_2, d_3)$ を得，$ku_T = (I_1, I_2)$ に対して，$dk_{(ID,vk),T} = (S_\theta \cdot d_1 \cdot I_1 \cdot F_W^{(1)}(ID)^{r_\theta} \cdot F_{BB}(T)^s, d_2 g^{r_\theta}, d_3, I_2 g^s)$ を計算し，M を返すことができる．

上記の議論より，B は復号オラクルのシミュレーションを実行できており，さらに，Forge が起こらない限り完全なシミュレーションとなっている．Forge が発生した時には，B はランダムビットを出力しているだけなので，事象 Succ^{HIBE} を B が 2-level Waters's HIBE の IND-ID-CPA 安全性を破る事象とすれば，$\left|\Pr[\mathsf{Succ}^{HIBE}] - \frac{1}{2}\right| = \left|2 \cdot (\Pr\left[\mathsf{Succ} \wedge \overline{\mathsf{Forge}}\right] + \frac{1}{2}\Pr[\mathsf{Forge}] - \frac{1}{2}\right|$ が成立する．従って，証明が完成した． □

以上により定理 1 が証明された． □

参考文献

[1] Boneh, D., and Franklin, M.: Identity-based encryption from the Weil pairing. In CRYPTO 2001 (pp. 213-229). Springer, 2001.

[2] Boneh, D., Canetti, R., Halevi, S., and Katz, J.: Chosen-ciphertext security from identity-based encryption. SIAM Journal on Computing, 36(5), pp.1301-1328, 2006.

[3] Boneh, D., and Boyen, X.: Efficient selective identity-based encryption without random oracles. Journal of Cryptology, 24(4), pp.659-693, 2011.

[4] Boldyreva, A., Goyal, V., and Kumar, V.: Identity-based encryption with efficient revocation. In Proceedings of the 15th ACM conference on Computer and communications security (pp. 417-426). ACM, 2008.

[5] Libert, B., and Vergnaud, D.: Adaptive-ID secure revocable identity-based encryption. In CT-RSA 2009 (pp. 1-15). Springer, 2009.

[6] Naor, D., Naor, M., and Lotspiech, J.: Revocation and tracing schemes for stateless receivers. In CRYPTO 2001 (pp. 41-62). Springer, 2001.

[7] Seo, J. H., and Emura, K.: Revocable identity-based encryption revisited: Security model and construction. In PKC 2013 (pp. 216-234). Springer, 2013.

[8] Waters, B: Efficient identity-based encryption without random oracles. In EUROCRYPT 2005 (pp. 114-127). Springer, 2005.