2007－CSEC－37 (2)
2007／5／25

# 送信者認証機能付きブロードキャスト暗号の改良と安全性に関する考察

金沢 史明† 大川 直人‡ 土井 洋‡ 岡本 健† 岡本 栄司†

†筑波大学大学院 システム情報工学研究科
305-8573 茨城県つくば市天王台 1-1-1

‡情報セキュリティ大学院大学
221-0835 神奈川県横浜市神奈川区鶴屋町 2-14-1

kanazawa@cipher.risk.tsukuba.ac.jp
{ken,okamoto}@risk.tsukuba.ac.jp

{mgs053103,doi}@iisec.ac.jp

**あらまし** ブロードキャスト暗号とは，多数のユーザが存在する中で，送信者が選択したユーザのみに対し，ブロードキャストチャネルを通して安全かつ効率的にデータを配布する技術である．2005 年，Boneh らによって，秘密鍵と暗号文のサイズが小さい方式が提案された．金沢らは，Boneh らの方式を基に，送信者認証機能付きブロードキャスト暗号を提案したが，大川らによって欠陥が指摘された．本稿では，金沢らの方式を改良し，大川らの攻撃に耐えうる方式を提案する．

# Improvement of Broadcast Encryption with Sender Authentication and its Security

Fumiaki Kanazawa† Naoto Ohkawa‡ Hiroshi Doi‡ Takeshi Okamoto†
Eiji Okamoto†

†Graduate School of Systems and Information Engineering, University of Tsukuba
1-1-1 Tennodai, Tsukuba, Ibaraki 305-8573, Japan

kanazawa@cipher.risk.tsukuba.ac.jp
{ken,okamoto}@risk.tsukuba.ac.jp

‡Institute of Information Security
2-14-1 Tsuruya-cho, Kanagawa-ku, Yokohama, Kanagawa 221-0835, Japan

{mgs053103,doi}@iisec.ac.jp

**Abstract** Broadcast encryption allows a sender to distribute digital data securely and efficiently, through a broadcast channel to selected users. In 2005, Boneh et al. proposed the broadcast encryption with short private key and ciphertext. Kanazawa et al. proposed the broadcast encryption with sender authentication based on Boneh's scheme. However the weakness of Kanazawa's scheme was pointed out by Ohkawa et al. In this paper, we propose the secure scheme against Ohkawa's attack.

## 1 はじめに

近年，デジタル情報はコピーが容易であることから，インターネットなども含め，放送されたデジタルコンテンツの著作権が侵害されるという事件が多発しており，深刻な社会問題になっている．有料放送では，商業的な理由から料金を支払う者のみが視聴などのサービス利用を許され，料金を支払わない者はサービスを利用できないといったシステムが求められる．ただし，消費者は自由に契約，解約，再契約できる必要があるため，サービス利用の権限をどのようにして動的に，かつ効率よく与えるかが，消費者の利便性に深くかかわり，結果としてシステムにおける大切な要件となる．

ブロードキャスト暗号 [1,5] は，こうしたサービスに適した暗号方式であり，Berkovits や Fiat らによって提案された．この暗号方式では，送信者が指定したメンバは暗号文を復号することができるが，それ以外のメンバは復号することができない．従来の暗号（データ守秘）方式は送信者と受信者が一対一であるが，この方式は一対多の暗号方式である点が異なっている．

ブロードキャスト暗号は，秘密鍵のサイズ，(暗号文の) ヘッダのサイズ，暗号化・復号時における計算コストの 3 点から性能評価が行われる．ただし，秘密鍵サイズとヘッダサイズは互いにトレードオフの関係にあり，そのバランスが考慮された方式が数多く提案されている．

2005 年，Boneh らによって，stateless receiver（秘密

鍵の更新機能をもたない受信者）に適したブロードキャスト暗号 [2] が提案された．(以降，本方式を BGW 方式と称す．) BGW 方式は，秘密鍵サイズとヘッダサイズに関し，優れた性能を有する．

前述の有料放送では，送信者の身元を示すため，送信メッセージに対し署名機能を付加することが望ましい．しかしながら，BGW 方式は暗号方式であるため，署名機能を有していない．このことは，送信者と送信データの正当性が保証されない（相手認証およびメッセージ認証の不備）ということを意味する．また，単純に署名機能を付加しただけでは，鍵管理や伝送量などの点で，送信時におけるオーバヘッドが増大する．

そういった問題を解決するため，金沢らは，送信者認証機能を付加したブロードキャスト暗号 [7] を提案した．(以降，本方式を KOIO 方式と称す．)

しかし，大川によって KOIO 方式に対する攻撃法 [10] が提案された．この攻撃法は，ある者が他人に成りすました認証子生成を可能とする方法であり，KOIO 方式の送信者認証機能の欠陥を示すものである．

本稿では，KOIO 方式を改良し，大川らの攻撃法に耐えうる方式を提案する．提案方式は，KOIO 方式同様，送信者が認証子（署名）生成時に秘密鍵を明示的に使用し，署名生成と暗号化処理を同時に行う方式である．このため，BGW 方式に対し単純に署名機能を付加した場合に比べ，鍵管理や伝送量という点で優れている．また，提案方式の安全性と性能について考察を行い，送信者認証機能の安全性を証明した．

以下，本稿の構成を述べる．第 2 章では，本稿で必要な知識を述べる．第 3 章では，本稿の提案方式であるブロードキャスト暗号について述べる．第 4 章では，提案方式の安全性，性能などを考察する．最後に第 5 章で本稿をまとめる．

# 2 準備

本章では，各表記と安全性の根拠となる問題を定義する．

## 2.1 表記

各表記を以下のように定義する．

$\mathcal{N}$ ：システム内の全ユーザの集合 ($|\mathcal{N}| = N$)
$\mathcal{S}$ ：復号を許可するユーザの集合 ($|\mathcal{S}| = n$)
$p, q$ ：$q$ が $p-1$ を割り切るような大きな素数
$\mathbb{G}_1, \mathbb{G}_2$ ：素数 $q$ を位数とする有限加法群
$\mathbb{G}_T$ ：素数 $q$ を位数とする有限乗法群
$\mathcal{H}(\cdot)$ ：ハッシュ関数 ($\{0,1\}^* \to \mathbb{Z}_q$)
$P$ ：$\mathbb{G}_1$ の生成元

## 2.2 ペアリング

ペアリング関数 $\hat{e}$ を以下のように定義する．

**定義 1** (ペアリング)**.** 以下の二つの性質をもつ写像 $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T$ をペアリングという．

(双線形性)
$\forall a, b \in \mathbb{Z}_p, \forall P \in \mathbb{G}_1, \forall Q \in \mathbb{G}_2;$
$\hat{e}(aP, bQ) = \hat{e}(P, Q)^{ab}$

(非退化性)
$\forall P \in \mathbb{G}_1; \quad \hat{e}(P, Q) = 1 \Rightarrow Q = 0_{\mathbb{G}_2}$
$\forall Q \in \mathbb{G}_2; \quad \hat{e}(P, Q) = 1 \Rightarrow P = 0_{\mathbb{G}_1}$

本稿では，群 $\mathbb{G}_1$ と群 $\mathbb{G}_2$ が同一の群であるものとし，それに対応したペアリングを用いることとする．すなわち，本稿で取り扱うペアリングは写像 $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{G}_T$ である．

## 2.3 問題と仮定

提案方式の安全性を考察するために，以下のような問題と仮定を定義する．

**定義 2** (Diffie-Hellman 問題)**.** $\alpha, \beta \in \mathbb{Z}_q$, $P \in \mathbb{G}_1$ とする．DH 問題とは，任意のベクトル $(P, \alpha P, \beta P)$ が与えられたとき，$\alpha\beta P$ を求める問題である．

**定義 3** (拡張 Bilinear Diffie-Hellman Exponent 問題)**.** $\alpha \in \mathbb{Z}_q$, $N \in \mathbb{N}$, $T \in \mathbb{G}_1$ とする．拡張 $N$-BDHE 問題とは，任意のベクトル $(T, \alpha^{-(N-1)}P, \cdots, \alpha^{-1}P, P, \alpha P, \alpha^2 P, \cdots, \alpha^N P, \alpha^{N+2}P, \cdots, \alpha^{2N}P)$ が与えられたとき，$e(P,T)^{\alpha^{N+1}}$ を求める問題である．[1]

**仮定 1.** DH 問題や拡張 $N$-BDHE 問題を解く確率的多項式時間アルゴリズムは存在しない．

# 3 提案方式

提案方式は，KOIO 方式を改良した方式であり，大川らの攻撃法 [10] に耐えうる方式である．

## 3.1 改良手法

大川らは，KOIO 方式の安全性に関する問題点として，以下の 3 点を挙げている．

1. セッション鍵 $K$ の流用による暗号文生成が可能である．
2. 任意の送信者になりすまし，一人のユーザ $i \in \mathcal{S}$ が受理する暗号文作成が可能である．
3. 任意の送信者になりすまし，復号を許可するユーザ全員が受理する暗号文生成が可能である

問題点 1 と 3 を解決するために，暗号化時に生成する認証子に改良を加える必要がある．具体的には，コミットメントを生成するハッシュ関数の入力に，暗号化メッセージ（問題 1 のため）とセッション鍵（問題 3 のため）を加えることとする．

問題 2 を解決するために，送信者 $a$ が秘密鍵 $D_a$ を使用して暗号化したことを，受信者が検証する必要がある．具体的には，受信者がセッション鍵を導出することとともに，ヘッダの型を検証することとする．

[1] 与えられたベクトルに $\alpha^{N+1}P$ が抜けていることに注意されたい．

### 3.2 モデル

提案方式は，送信者の認証機能を備えているため，以下の様になる．

初期化・鍵生成フェーズ 初期段階において，システム管理者はセキュリティパラメータからユーザの秘密鍵と公開情報を生成する．生成した秘密鍵を各ユーザに配布し，公開情報を各ユーザが常に取得可能な状態にする．

具体的には，システム内のユーザ数 $N$ を考慮し，各ユーザの秘密鍵 $\{D_1, \cdots, D_N\}$，公開情報 $PK$ を生成する．生成した秘密鍵 $D_i$ をユーザ $i$ にそれぞれ配布する．

暗号化フェーズ 送信者は，復号を許可するユーザを選択し，初期に配布された秘密鍵を用いてメッセージを暗号化する．更に暗号鍵に関する情報（ヘッダ）を生成する．暗号化されたメッセージは，復号を許可されたユーザのみが導出可能な鍵で復号可能である．

具体的には，送信者 $a \in \mathcal{N}$ は復号を許可するユーザ集合 $\mathcal{S}$ を選択し，公開情報 $PK$，秘密鍵 $D_a$ を用いてメッセージ $M$ を暗号化し，暗号化メッセージ $C_M$ と認証子を含んだヘッダ $Hdr$ を生成する．

検証・復号フェーズ 復号を許可されたユーザは，初期に配布された秘密鍵とヘッダから送信者の検証を行う．検証に成功した場合，秘密鍵とヘッダから復号鍵（セッション鍵）を導出し，暗号化メッセージを復号する．

具体的には，ユーザ $i \in \mathcal{S}$ は，復号許可ユーザ集合 $\mathcal{S}$，公開情報 $PK$，秘密鍵 $D_i$ を用いて，ヘッダ $Hdr$ から，送信者が $a$ であるかを検証する．検証に成功した場合，$\mathcal{S}$，$PK$，$D_i$ を用いて，$Hdr$ から復号鍵を導出し，暗号化メッセージ $C_M$ を復号する．

### 3.3 プロトコル

初期化・鍵生成

1. 乱数 $\alpha \in \mathbb{Z}_q$ を生成し，$P_i = \alpha^i P$ を計算する．($i = -(N-1), \cdots, -2, -1, 1, 2, \cdots, N, N+2, \cdots, 2N$)
2. 乱数 $\gamma \in \mathbb{Z}_q$ を生成し，$Q = \gamma P$ と，ユーザ $i \in \{1, \cdots, N\}$ の署名鍵 $D_i = \gamma P_i$ を計算する．
3. 各ユーザの署名鍵 $\{D_1, \cdots, D_N\}$ と，公開情報 $PK = (P, P_{-(N-1)}, \cdots, P_{-1}, P_1, \cdots, P_N, P_{N+2}, \cdots, P_{2N}, Q) \in \mathbb{G}_1^{3N}$ を出力する．

各ユーザは，随時，公開情報 $PK$ より $g = \hat{e}(P_N, P_1)$ を計算する．(各自の記憶領域に保存してもよい．)

暗号化

1. 送信者 $a$（秘密鍵 $D_a$ を所持）は，乱数 $t \in \mathbb{Z}_q$ を生成し，セッション鍵 $K = g^t$ を計算する．
2. メッセージ $M$ を鍵 $K$ で暗号化し，暗号化メッセージ $C_M$ を生成する（任意の共通鍵暗号系を利用）．
3. 乱数 $r \in \mathbb{Z}_q$ を生成し，$e = \mathcal{H}(C_M, K, g^r)$ を計算する．
4. $y = r - et$ を計算する．
5. ヘッダ $Hdr$ を以下のように計算する．($e$ と $y$ は送信者とメッセージの認証子)

$$Hdr = \left( tP\,,\ t(D_a + \sum_{j \in \mathcal{S}} P_{N+1+a-j}), e, y \right).$$

6. $Hdr$ と $C_M$ を出力する．

検証・復号

$Hdr = (C_0, C_1, e, y)$ とする．

1. 以下の式が成り立つか検証する．成り立たなければ暗号文を破棄する．

$$\hat{e}(P_{-a},\, C_1) = \hat{e}(Q + \sum_{j \in \mathcal{S}} P_{N+1-j}\,,\, C_0).$$

2. ユーザ $i \in \mathcal{S}$ は秘密鍵 $D_i \in \mathbb{G}_1$ を利用し，$C_M$ の復号鍵 $K'$ を以下のように導出する．

$$K' = \frac{\hat{e}(P_{i-a},\, C_1)}{\hat{e}(D_i + \sum_{j \in \mathcal{S},\, j \neq i} P_{N+1-j+i}\,,\, C_0)}.$$

3. $e = \mathcal{H}(C_M, K', g^y K'^e)$ が成り立つか検証する．成立しなければ，暗号文を破棄する．
4. 鍵 $K'$ で $C_M$ を復号し，復号されたメッセージ $M$ を出力．

検証・復号アルゴリズムの手順 1 において，正しく生成された $C_0, C_1$ について検証式が成り立つことを以下に示す．

$$\begin{aligned}
\hat{e}(P_{-a},\, C_1) &= \hat{e}(P_{-a},\, C_1) \\
&= \hat{e}(P_{-a},\, t(D_a + \sum_{j \in \mathcal{S}} P_{N+1+a-j})) \\
&= \hat{e}(\alpha^{-a} P,\, t\alpha^a (Q + \sum_{j \in \mathcal{S}} P_{N+1-j})) \\
&= \hat{e}(Q + \sum_{j \in \mathcal{S}} P_{N+1-j},\, tP) \\
&= \hat{e}(Q + \sum_{j \in \mathcal{S}} P_{N+1-j},\, C_0).
\end{aligned}$$

$K$ が正確に導出されるかどうかは，KOIO 方式と同じであるため，省略する．

## 4 考察

本章では，第 3 章の提案方式に関して，送信者認証機能の安全性と，性能を考察する．

### 4.1 安全性

提案方式は，KOIO 方式の検証・復号アルゴリズムに検証手順 1 と 3 を加えたものである．

手順 3 は Schnorr 署名と同じであり，$K' = g^t$ となる $t$ の知識の証明である．従って，手順 3 の検証式が成り立つ $e, y$ を生成した者（署名生成者）は，$K' = g^t$ となる $t$ の知識を有している．つまり，$t$ を知っていることになる．また，提案方式は，暗号化アルゴリズムに従って生成した $C_0, C_1$ ならば成り立つはずの検証手順 1 を加えている．

実は，手順 1 と 3 の検証式が成り立つ $(C_0, C_1, e, y)$ の生成者は，$D_a$ を計算できることが証明可能である．

表 1: 提案方式の性能評価

| | (a) | (b) | KOIO 方式 | 提案方式 |
|---|---|---|---|---|
| 秘密鍵サイズ (bit) | 502 | 502 | 342 | 342 |
| 暗号文サイズ (bit) | 1004 | 855 | 1004 | 1004 |
| 公開情報サイズ (bit) | $1710N + 342$ | $1026N + 342$ | $1026N$ | $1026N$ |
| 暗号化 ($M$) | $645.8 + 0.3n$ | $571.6 + 0.3n$ | $645.8 + 0.3n$ | $645.8 + 0.3n$ |
| 検証・復号 ($M$) | $1289.7 + 0.3n$ | $1694.5 + 0.3n$ | $1289.7 + 0.3n$ | $1456.1 + 0.6n$ |

(a) BGW 方式 + Schnorr 署名方式 [11]
(b) BGW 方式 + short signature 方式 [3]
暗号文サイズはヘッダ部分のみ

**定理 1.** 仮定 1 の下に，3.3 節の検証・復号で受理される $(C_0, C_1, e, y)$ の生成者は，$D_a$ を計算可能である．

定理 1 の証明は付録 A を参照されたい．なお，定理 1 によれば，提案方式に対するなりすましを行うことは，送信者の秘密鍵を求めることと同程度に困難であることが分かる．

## 4.2 性能

本節では，提案方式の性能を考察する

### 4.2.1 計算コストの削減

3.3 節の検証・復号アルゴリズムに対して，計算コスト削減を考える．

本節では，暗号文の検証と復号鍵 $K'$ の導出を同時に行うことを提案する．具体的には，手順 1 と 2 の代わりに，付録 B の代替手順 1′ を実行する．すなわち，手順 1 → 手順 2 → 手順 3 → 手順 4 の順で実行するところを，代替手順 1′ → 手順 3 → 手順 4 の順で実行することとなる．この代替により，ペアリング演算を 2 回を，$\mathbb{G}_1$ 上のべき倍算 2 回と加算 2 回へ削減することが可能となる．

3.3 節の検証・復号アルゴリズムにおいて，手順 3 の検証式が成り立たずに破棄される暗号文は，代替手順 1′ を採用した場合においても手順 3 で破棄される．

また，手順 1 で破棄される暗号文は，代替手順 1′ が暗号文を破棄する機能を持たないため，手順 3 で破棄される．これは証明可能であり付録 B に記している．受信者によって生成される乱数 $k$ を，送信者が予知することが困難であることに基づいている．

以上より，送信者認証機能に関して，3.3 節の検証・復号アルゴリズムと同等の安全性を持つといえる．

### 4.2.2 評価

提案方式の性能評価を行う．ただし，評価対象は 4.2.1 節のコスト削減を施したものとする．

なお，表 1 は以下の条件における値である．$|p| = 1026$, $|q| = 160$, $\mathbb{G}_1$ は楕円曲線上に定義される加法群とする．$(x, y) \in \mathbb{G}_1$ のサイズは 342bit($|x| = |y| = 171$) とし，加算とべき倍算は Jacobian 座標系 [4] で計算する．$\mathbb{G}_1$ 上のべき倍算と $\mathbb{G}_T$ 上のべき乗算は binary 法 [8] を利用し，ペアリング演算は Kobayashi らの sliding window Miller 法 [9] を利用する．さらに，$1M$ を $\mathbb{Z}_p^*$ 上における乗算 1 回分のコストと定義する．

提案方式は，KOIO 方式と比較して，検証・復号にかかる計算量のみが多い．具体的には，$\mathbb{G}_1$ 上の演算に関して，KOIO 方式では加算が $n - 1$ 回のみとなるが，提案方式では加算が $2n + 1$ 回とべき倍算が 2 回となる．なお，$\mathbb{G}_T$ 上の演算に関しては同量となる．

## 4.3 検証者指定型 1-out-of-$n$ 署名

本節では，KOIO 方式を変形である検証者指定型 1-out-of-$n$ 署名 [7] について考察する．

本方式は，大川らの攻撃法を応用することで，署名者の成りすましが可能である．これは，検証者が導出した検証鍵 $K'$ が $\hat{e}(tP, P_{N+1})$ でない場合においても受諾するためである．

署名者なりすましの対応策として，第 3 章と同様に，コミットメントに検証鍵を加えることが挙げられる．

# 5 まとめ

本稿では，KOIO 方式を改良し，送信者認証機能付きブロードキャスト暗号方式を提案した．送信者認証機能に関して安全性を証明し，大川らの攻撃に耐えうる方式となった．効率の観点から考慮すると，検証・復号時の計算コストを除き，KOIO 方式の利点を継承している．また，KOIO 方式の変形版である検証者指定型 1-out-of-$n$ 署名方式の改良方式を提案した．改良により安全性が向上したが，効率は変わっていない．今後は，安全性を証明する必要がある．

# 参考文献


[1] Berkovits, S.: How to Broadcast a Secret, *Advances in Cryptology — Eurocrypt '91*, LNCS, Vol.548, pp.535–541, Springer-Verlag (1991).

[2] Boneh, D., Gentry, C. and Waters, B.: Collusion Resistant Broadcast Encryption with Short Ciphertexts and Private Keys, *Advances in Cryptology — CRYPTO 2005*, LNCS, Vol.3621, pp.258–275, Springer-Verlag (2005).

[3] Boneh, D., Lynn, B. and Shacham, H.: Short signatures from the Weil pairing, *Advances in Cryptology — Asiacrypt 2001*, LNCS, Vol.2248, pp.514–532, Springer-Verlag (2001).

[4] Cohen, H., Miyaji, A., Ono, T.: Efficient elliptic curve exponentiation using mixed coordinates, *Advances in Cryptology — Asiacrypt '98*, LNCS, Vol.1514, pp.51–65, Springer-Verlag (1998).

[5] Fiat, A., and Naor, M.: Broadcast Encryption, *Advances in Cryptology — CRYPTO '93*, LNCS, Vol.773, pp.480–491, Springer-Verlag (1994).

[6] 金沢史明, 岡本健, 猪俣敦夫, 岡本栄司: 送信者に認証機能を付加したブロードキャスト暗号, コンピュータセキュリティシンポジウム (CSS2005) 論文集, pp.349–354 (2005).

[7] 金沢史明, 岡本健, 猪俣敦夫, 岡本栄司: 送信者に認証機能を付加したブロードキャスト暗号とその応用, 情報処理学会論文誌, Vol.47, No.11, pp.2992–3004 (2006).

[8] Knuth, D.E.: Seminumerical Algorithms, The Art of Computer Programming, Vol.2, 3rd ed., Addison-Wesley (1998).

[9] Kobayashi, T., Aoki, K., Imai, H.: Efficient Algorithms for Tate Pairing, IEICE Trans. Fandamentals, Vol.E89-A, No.1, pp.134–143 (2006)

[10] 大川直人, 土井洋: 送信者に認証機能を付加したブロードキャスト暗号の安全性に関する一考察, 情報処理学会研究報告, 2007-CSEC-36, pp.31–35 (2007).

[11] Schnorr, C. P.: Efficient Signature Generation by Smart Cards, *J. Cryptology*, Vol.4, No.3, pp.161–174 (1991).

# 付録 A

以下は 4.1 節の定理 1 の証明である.

暗号文 $(C_0, C_1, e, y)$ の生成者は $D_a$ を計算可能であることを示す.

まず, 検証手順 3 が成り立つことから, $(C_0, C_1, e, y)$ の生成者は $K' = g^t$ となる $t$ を知っていることが分かる. 事実, 手順 3 は Schnorr 署名 (実際には, $t$ の知識の証明) の検証式であり, 手順 3 が成り立つ $e, y$ を生成した者 (署名生成者) は, $t$ を知っていることになる.

よって, $K'$ の導出式より,

$$\frac{\hat{e}(P_{i-a},\, C_1)}{\hat{e}(D_i + \sum_{\substack{j\in\mathcal{S}\\ j\neq i}} P_{N+1-j+i}\,,\, C_0)} = g^t,$$

であり, 式変形すると,

$$\hat{e}(P_{i-a},\, C_1) = g^t \cdot \hat{e}(D_i + \sum_{\substack{j\in\mathcal{S}\\ j\neq i}} P_{N+1-j+i}\,,\, C_0),$$

となる. ここで右辺を式変形すると,

$$\begin{aligned} & g^t \cdot \hat{e}(D_i + \sum_{\substack{j\in\mathcal{S}\\ j\neq i}} P_{N+1-j+i}\,,\, C_0) \\ = \;& \hat{e}(P_{i-a}, tP_{N+1-i+a}) \\ & \quad \cdot \hat{e}(\alpha^a(\gamma + \sum_{\substack{j\in\mathcal{S}\\ j\neq i}} \alpha^{N+1-j})P_{i-a}\,,\, C_0) \\ = \;& \hat{e}(P_{i-a}, tP_{N+1-i+a} + \alpha^a(\gamma + \sum_{\substack{j\in\mathcal{S}\\ j\neq i}} \alpha^{N+1-j})C_0), \end{aligned}$$

となる. ここで, $\hat{e}$ の第 2 成分に注目すると,

$$C_1 = tP_{N+1-i+a} + \alpha^a(\gamma + \sum_{\substack{j\in\mathcal{S}\\ j\neq i}} \alpha^{N+1-j})C_0,$$

であることが分かる. ここで, $C_1$ を手順 1 の検証式

$$\hat{e}(P_{-a},\, C_1) = \hat{e}(Q + \sum_{j\in\mathcal{S}} P_{N+1-j}\,,\, C_0),$$

へ代入し, 式変形を行うと,

$$\begin{aligned} & \hat{e}(P_{-a}, tP_{N+1-i+a} + \alpha^a(\gamma + \sum_{\substack{j\in\mathcal{S}\\ j\neq i}} \alpha^{N+1-j})C_0) \\ & \qquad = \hat{e}(Q + \sum_{j\in\mathcal{S}} P_{N+1-j}\,,\, C_0), \end{aligned}$$

$$\begin{aligned} & \hat{e}(P, tP_{N+1-i} + (\gamma + \sum_{\substack{j\in\mathcal{S}\\ j\neq i}} \alpha^{N+1-j})C_0) \\ & \quad = \hat{e}((\gamma + \sum_{\substack{j\in\mathcal{S}\\ j\neq i}} \alpha^{N+1-j})C_0 + \alpha^{N+1-i}C_0\,,\, P), \end{aligned}$$

$$\begin{aligned} \hat{e}(P, tP_{N+1-i}) &= \hat{e}(\alpha^{N+1-i}C_0\,,\, P), \\ \hat{e}(\alpha^{N+1-i}tP, P) &= \hat{e}(\alpha^{N+1-i}C_0\,,\, P), \end{aligned}$$

となる. ここで, $\hat{e}$ の第 1 成分に注目すると,

$$C_0 = tP,$$

であることが分かる. ここで, $C_0$ を手順 1 の検証式へ代入し, 式変形を行うと,

$$\begin{aligned} \hat{e}(P_{-a},\, C_1) &= \hat{e}(Q + \sum_{j\in\mathcal{S}} P_{N+1-j}\,,\, tP), \\ \hat{e}(C_1,\, P_{-a}) &= \hat{e}(t(D_a + \sum_{j\in\mathcal{S}} P_{N+1+a-j})\,,\, P_{-a}), \end{aligned}$$

となり, $\hat{e}$ の第 1 成分に注目すると,

$$C_1 = t(D_a + \sum_{j\in\mathcal{S}} P_{N+1+a-j}),$$

であることが分かる. $(C_0, C_1, e, y)$ の生成者は, $C_1$ や $P_{N+1+a-j}$ を知っているので,

$$D_a = \frac{C_1}{t} - \sum_{j\in\mathcal{S}} P_{N+1+a-j},$$

と, $D_a$ を計算可能である.

ゆえに定理 1 は証明された. □

# 付録B

4.2.1節で述べた計算コスト削減のために，3.3節の検証・復号アルゴリズムの手順1と2の代替手順1′を示す．代替手順1′は，手順1の検証と手順2の鍵導出を同時に行うものである．

1′. ユーザ $i \in \mathcal{S}$（秘密鍵 $D_i$ を所持）は，乱数 $k \in \mathbb{Z}_q$ を生成し，$X_0$ と $X_1$ を以下のように計算する．

$$\begin{aligned} X_0 &= P_{i-a} + kP_{-a}, \\ X_1 &= D_i + \sum_{\substack{j \in \mathcal{S} \\ j \neq i}} P_{N+1-j+i} \\ &\qquad + k(Q + \sum_{j \in \mathcal{S}} P_{N+1-j}). \end{aligned}$$

鍵 $K'$ を以下のように導出する．

$$K' = \frac{\hat{e}(X_0, C_1)}{\hat{e}(X_1, C_0)}.$$

代替手順1′で導出された鍵 $K'$ は，次の手順3の検証に用いられる．

暗号文 $(C_0, C_1, e, y)$ について，手順1の検証式が成り立つ場合，代替手順1′で導出された鍵は，手順2で導出される鍵と同一となる．

他方，元の手順1の検証式を満たさない場合を考える．手順1′は鍵導出のみを行うため，暗号文を破棄する機能を持たない．手順1の検証式を満たさない暗号文は，手順1′の代わりに手順3の検証で破棄されることとなる．

以下，手順1と2の代わりに代替手順1′を実行した場合，手順1の検証式を満たさない暗号文が手順3の検証で受諾される確率は，高々$1/2^q$であることを示す．

まず，暗号文 $(C_0, C_1, e, y)$ について手順1の検証式が成り立たないと仮定する．すなわち，

$$\hat{e}(P_{-a}, C_1) \neq \hat{e}(Q + \sum_{j \in \mathcal{S}} P_{N+1-j}\,,\, C_0),$$

であり，式変形すると，

$$\frac{\hat{e}(P_{-a}, C_1)}{\hat{e}(Q + \sum_{j \in \mathcal{S}} P_{N+1-j}\,,\, C_0)} \neq 1, \qquad (1)$$

となる．また，$X_0$ と $X_1$ は，それぞれ，

$$\begin{aligned} X_0 &= P_{i-a} + kP_{-a} \\ &= (\alpha^i + k)P_{-a}, \\ X_1 &= D_i + \sum_{\substack{j \in \mathcal{S} \\ j \neq i}} P_{N+1-j+i} + k(Q + \sum_{j \in \mathcal{S}} P_{N+1-j}) \\ &= (\alpha^i + k)(Q + \sum_{j \in \mathcal{S}} P_{N+1-j}) - P_{N+1}, \end{aligned}$$

であることから，導出される鍵 $K'$ は

$$\begin{aligned} K' &= \frac{\hat{e}((\alpha^i + k)P_{-a}, C_1)}{\hat{e}((\alpha^i + k)(Q + \sum_{j \in \mathcal{S}} P_{N+1-j}) - P_{N+1}, C_0)} \\ &= \frac{\hat{e}(P_{-a}, C_1)^{\alpha^i + k} \cdot \hat{e}(P_{N+1}, C_0)}{\hat{e}(Q + \sum_{j \in \mathcal{S}} P_{N+1-j}, C_0)^{\alpha^i + k}} \\ &= \left(\frac{\hat{e}(P_{-a}, C_1)}{\hat{e}(Q + \sum_{j \in \mathcal{S}} P_{N+1-j}, C_0)}\right)^{\alpha^i + k} \cdot \hat{e}(P_{N+1}, C_0), \end{aligned}$$

となる．式1より，鍵 $K'$ は受信者の乱数 $k$ によって変化することが分かる．よって，手順3の検証式，

$$e = \mathcal{H}(C_M, K', g^y K'^e),$$

の右辺は鍵 $K'$ を引数とするため，乱数 $k$ に依存する．また，左辺の $e$ は送信者が暗号化の際に生成したものである．この検証式が成り立つためには，少なくとも，$e$ を生成する前に，送信者が受信者によって生成される乱数 $k$ を予知する必要がある．送信者の予想と受信者が生成した乱数 $k$ が一致する確率は，$1/2^{|k|} = 1/2^q$ である．

以上より，受諾される確率が高々$1/2^q$であることは証明された．