情報漏洩耐性を有する暗号系に関する研究

Kaneko, Kiminori / 金子, 公徳

(出版者 / Publisher)
法政大学大学院情報科学研究科
(雑誌名 / Journal or Publication Title)
法政大学大学院紀要. 情報科学研究科編
(巻 / Volume)
19
(開始ページ / Start Page)
1
(終了ページ / End Page)
6
(発行年 / Year)
2024-03-24
(URL)
https://doi.org/10.15002/00030609

情報漏洩耐性を有する暗号系に関する研究
A Study on Cryptosystems with Resilience against Information Leakage

金子公徳 (Kiminori Kaneko)*

法政大学大学院 情報科学研究科 情報科学専攻
kiminori.kaneko.6m@stu.hosei.ac.jp

Abstract

The contribution of the paper is twofold: (1) I show two attacks against a cheating detectable secret sharing scheme (CDSS for short) from NISS 2019. (2) I propose a certificate issuance system of public key infrastructure with intrusion-resilient signature and Intel SGX. Since the limited space available for writing, this abstract will focus on the first. The second contribution will be detailed in the full paper.

A CDSS is a variant of secret sharing scheme with the extra functionality to detect forgery of shares when some of shares are submitted maliciously. In NISS 2019, Becerra and Vega proposed a CDSS which employs addition over $\mathbb{Z}_{2^m}$ as the verification function whereas the secret is distributed and reconstructed with computation over $\mathbb{F}_{2^m}$. They proved that the scheme is secure and the bit length of the share meets the lower bound with equality. In this paper, I show that the scheme is insecure by showing that only a single cheater can forge the resulting secret without being detected by the reconstruction algorithm. I present two types of attack. The probability of successful cheating of the first attack is 1 but the pattern of forgery is limited. In the second attack, the successful cheating probability is less than the first one but the pattern of forgery is more flexible. The successful cheating probabilities of the proposed attacks are evaluated with the careful analysis about structures of additions over $\mathbb{Z}_{2^m}$ and $\mathbb{F}_{2^m}$, which will be of independent interest.

1 背景

本研究では、NISS 2019 で提案された不正検出可能な秘密分散方式に対する暗号解析と侵入耐性署名を用いた証明書発行アプリケーションの提案を行った。以降では、不正検出可能な秘密分散方式に対する暗号解析について述べる。秘密分散方式は秘密を複数の参加者に分散し、認可された参加集合からのみ秘密を復元できる方法である。認可されていない参加者集合は秘密の部分情報も得られない。分散された秘密情報のそれぞれをシェアと呼ぶ。

秘密分散方式は Shamir [7] と Blakley [2] によってそれぞれ独立に提案された。Tompa と Woll [8] は Shamir が提案した方式が不正に対して安全でないことを示し、不正検出可能な秘密分散方式 (以下 CDSS とする) を提案した。

復元時に、共謀した悪意のある参加者が改竄されたシェアを提出することで、他の参加者が間違った秘密を復元することがある。さらに、参加者は秘密が改竄されたことに気づくことができない。このようなシナリオは二つの方法でモデル化されている [3,6]。尾形、黒沢、Stinson は不正者が秘密を知らないモデルを提案し、Carpentieri、De Santis、Vaccaro は不正者が秘密を知っているモデルを提案した。これらのモデルはそれぞれ OKS モデルと CDV モデルと呼ばれる。

Becerra と Vega [1] は NISS 2019 で CDSS を提案した (以下 BV 方式 とする)。既存の多くの CDSS は復元された秘密の正当性を検証するためのチェックディジットの計算に乗算を用いるが、BV 方式は乗算を用いずに $\mathbb{Z}_{2^m}$ 上の加算を用いる。彼らは、加算は乗算よりも高速であるため、BV 方式が既存の CDSS [4,5] よりも計算効率が良いことを示している。また、BV 方式が OKS モデルにおいて安全であることを証明した。しかし、安全性証明で使われている証明手法は標準的なものでなく、彼らの証明から BV 方式が安全であることを保証するのは容易でない。

本研究では、BV 方式に対して二つの攻撃方法を示す。どちらの攻撃も一人の不正者がいれば実行できる。一つ目の攻撃では、不正者が復元アルゴリズムに対して確率 1 で不正できる。しかし、改竄パターン (改竄された秘密と元々の秘密との差分) が限られている。二つ目の攻撃は、不正者の不正成功確率が一つ目の攻撃以下となるが、改竄パターンをより柔軟に変更できる。$\mathbb{Z}_{2^m}$ と $\mathbb{F}_{2^m}$ 上の加算の類似性を慎重に分析し、提案する二つの攻撃の不正成功確率を評価する。

2 準備

2.1 記号の説明

本論文では、以下の記号を用いる。

* 指導教員：尾花賢 教授

$\mathbb{Z}$: 整数の集合
$\mathbb{N}$: 自然数の集合
$\mathbb{F}_p$: 位数 p の有限体の集合
$\mathbb{F}_{2^q}^{\leq \ell}$: ビット長が最大 ℓ のビット列の集合
$\oplus$: $\mathbb{F}_{2^q}$ 上の加法演算子 (q ビットの XOR と等価)
$\boxplus$: $\mathbb{Z}_{2^q}$ 上の加法演算子
$\|$: ビット列の結合
$|\mathcal{X}|$: 集合 $\mathcal{X}$ の濃度
$x_{[i]}$: x の i 番目のビット
$x_{[i:j]}$: x の i 番目から j 番目のビットを抽出したビット列
$w_H(x)$: x のハミング重み

以降では、数値 x を文脈的に応じてビット列や有限体上の要素として解釈する。例えば、数値 2^{m-1} をビット列 $\underbrace{100\ldots00}_{m-1}$ と解釈することもあれば、有限体 $\mathbb{F}_{2^m}(=\mathbb{F}_2(\alpha))$ 上の要素 α^{m-1} と解釈することもある。

2.2 (k,n) 閾値秘密分散方式

(k,n) 閾値秘密分散方式はディーラー D が秘密 s を n 人の参加者 $P_1,\ldots,P_n$ に分散し、k 人以上の参加者が秘密を復元できる方法である。しかし、$k-1$ 人以下の参加者は秘密の部分情報も得られない。参加者 P_i に分散された秘密の部分情報をシェアと呼び、v_i と表す。

Shamir は有限体 $\mathbb{F}_p$ 上の Lagrange 補完を用いて (k,n) 閾値秘密分散方式 $\mathtt{SS}=(\mathtt{ShareGen},\mathtt{Reconst})$ を提案した。分散アルゴリズム $\mathtt{ShareGen}$ は秘密 $s\in\mathbb{F}_p$ を入力として、n 個のシェア $(v_1,\ldots,v_n)$ を生成する。復元アルゴリズム $\mathtt{Reconst}$ は $j\geq k$ を満たすシェア $(v_{i_1},\ldots,v_{i_j})$ を入力として、秘密 $\hat{s}$ を復元する。分散アルゴリズムと復元アルゴリズムは以下のように定義される。

$\mathtt{ShareGen}(s)$:

Input: 秘密 $s\in\mathbb{F}_p$
Output: シェアのリスト $(v_1,\ldots,v_n)$

1: $f(0)=s$ となるような次数 $k-1$ のランダムな多項式 $f(x)\in\mathbb{F}_p[X]$ を生成
2: $v_i=f(i)$ を計算
3: **return** $(v_1,\ldots,v_n)$

$\mathtt{Reconst}(v_{i_1},\ldots,v_{i_j})$:

Input: $j\geq k$ を満たすシェアのリスト $(v_{i_1},\ldots,v_{i_j})$
Output: 復元された秘密 $\hat{s}$

1: Lagrange 補完を用いて $v_{i_1},\ldots,v_{i_j}$ から多項式 $\hat{f}(x)$ を復元
2: **return** $\hat{s}=\hat{f}(0)$

また、全ての閾値秘密分散方式は以下の正当性条件を満たす必要がある。

$$\Pr[(v_1,\ldots,v_n)\leftarrow\mathtt{ShareGen}(s);\ s'\leftarrow\mathtt{Reconst}(v_{i_1},\ldots,v_{i_k}):s'=s]=1$$

2.3 不正検出可能な秘密分散方式

CDSS は不正者による秘密の改竄を防ぐために Tompa と Woll によって初めて提案された [8]。通常の秘密分散方式と比較して、CDSS は復元アルゴリズムに改竄されたシェアを提出する共謀した不正者による不正を検出する機能を有する。復元アルゴリズムが不正を検出した場合、秘密を出力する代わりに特別な記号 $\perp$ を出力する。不正者は、復元アルゴリズムが不正を検出に失敗し、復元された秘密が分散された秘密と異なる場合に不正に成功する。

OKS モデルにおける CDSS $\mathtt{SS}=(\mathtt{ShareGen},\mathtt{Reconst})$ の安全性は以下の実験 $\mathbf{Exp}^{\mathrm{OKS}}_{\mathtt{SS},\mathcal{A}}$ ($\mathcal{A}=(\mathcal{A}_1,\mathcal{A}_2)$ は任意の攻撃者) を通して定義される。攻撃者 $\mathcal{A}_1$ は、$\mathtt{Reconst}$ にシェアを提出する参加者の添え字のリスト $(i_1,\ldots,i_{k-1})$ を出力するオラクル $\mathcal{A}_1^{\mathcal{O}_1}$ を与えられる。また、攻撃者 $\mathcal{A}_2$ には、シェア $(v_{i_1},\ldots,v_{i_{k-1}})$ を入力として、改竄されたシェア $(v'_{i_1},\ldots,v'_{i_{k-1}})$ を出力するオラクル $\mathcal{A}_2^{\mathcal{O}_2}$ を与えられる。実験は以下のように定義される。

$\mathbf{Exp}^{\mathrm{OKS}}_{\mathtt{SS},\mathcal{A}}$:

1: $s\leftarrow\mathcal{S}$
2: $(v_1,\ldots,v_n)\leftarrow\mathtt{ShareGen}(s)$
3: $(i_1,\ldots,i_{k-1})\leftarrow\mathcal{A}_1^{\mathcal{O}_1}$
4: $(v'_{i_1},\ldots,v'_{i_{k-1}},i_k)\leftarrow\mathcal{A}_2^{\mathcal{O}_2}(v_{i_1},\ldots,v_{i_{k-1}})$
5: $s'\leftarrow\mathtt{Reconst}(v'_{i_1},\ldots,v'_{i_{k-1}},v_{i_k})$
6: **if** $s'=s\vee s'=\perp$ **:**
7: **return** 0
8: **return** 1

不正者の優位性は以下のように定義される。

$$\mathbf{Adv}^{\mathrm{OKS}}_{\mathtt{SS},\mathcal{A}}=\Pr\left[\mathbf{Exp}^{\mathrm{OKS}}_{\mathtt{SS},\mathcal{A}}\to 1\right]$$

任意の攻撃者 $\mathcal{A}$ に対して、$\mathbf{Adv}^{\mathrm{OKS}}_{\mathtt{SS},\mathcal{A}}\leq\epsilon$ となるとき、CDSS $\mathtt{SS}$ は OKS モデルにおいて (k,n,ϵ) 安全であると定義する。

$\mathcal{S}$ を秘密の取り得る値の集合、$\mathcal{V}_i$ を参加者 P_i のシェアの取り得る値の集合としたとき、尾形、黒沢、Stinson は OKS モデルにおける (k,n,ϵ) 安全な秘密分散方式のシェアサイズの下界が $|\mathcal{V}_i|=(|\mathcal{S}|-1)/\epsilon+1$ であると示した。

2.4 Becerra-Vega の方式

Becerra と Vega は $\mathbb{Z}_{2^m}$ 上の加法を用いて OKS モデルにおける CDSS $\mathtt{SS}=(\mathtt{ShareGen},\mathtt{Reconst})$ を提案した。分散アルゴリズム $\mathtt{ShareGen}$ は秘密 $s\in\mathbb{F}_{2^{mN}}$ を入力として、s からチェックディジット $a\in\mathbb{F}_{2^m}$ を計算し、n 個のシェア $(v_1,\ldots,v_n)$ を生成する。シェア v_i は秘密 s のシェア $f_s(i)$ とチェックディジット a のシェア $f_a(i)$ の組である。復元アルゴリズム $\mathtt{Reconst}$ は k 個以上のシェア $(v_{i_1},\ldots,v_{i_j})$ を入力として、Lagrange 補完を用いて秘密 $\hat{s}$ と $\hat{a}$ を復元する。復元された秘密 $\hat{s}$ が元の秘密 s と一致するかどうかを判定するために、$\mathtt{ShareGen}$ と同様に $\hat{s}$ からチェックディジットを計算し、計算されたチェックディジットが $\hat{a}$ と一致するかどうか

を確認する。彼らの提案する復元アルゴリズムでは、$\hat{f}_s$ または $\hat{f}_a$ の次数が $k-1$ 未満の場合に $\perp$ を出力する。しかし、このような処理を追加すると、シェア数が k 未満の場合に秘密の部分情報が漏洩するため、その処理を省略する。各アルゴリズムは以下のように定義される。

$\texttt{ShareGen}(s)$:

Input: $m, N \in \mathbb{N}$、 $N > 1$ を満たす秘密 $s \in \mathbb{F}_{2^{mN}}$

Output: シェアのリスト $(v_1, \ldots, v_n)$.

1: $f_s(0) = s$ となるようなランダムな $k-1$ 次多項式 $f_s(x) \in \mathbb{F}_{2^{mN}}[X]$ を生成
2: s を分解して N 個のビット列 $s_1, s_2, \ldots, s_N \in \mathbb{Z}_{2^m}$ を生成
3: $a = s_1 \boxplus s_2 \boxplus \ldots \boxplus s_N$ を計算
4: $f_a(0) = a$ となるようなランダムな $k-1$ 次多項式 $f_a(x) \in \mathbb{F}_{2^m}[X]$ を生成
5: $v_i = (f_s(i), f_a(i))$ を計算
6: **return** $(v_1, \ldots, v_n)$

$\texttt{Reconst}(v_{i_1}, \ldots, v_{i_j})$:

Input: シェアのリスト $(v_{i_1}, \ldots, v_{i_j})$ ただし、$j \geq k$ を満たす

Output: 復元されたシェア $\hat{s}$ または $\perp$.

1: Lagrange 補完を用いて $v_{i_1}, \ldots, v_{i_j}$ から多項式 $\hat{f}_s(x)$ と $\hat{f}_a(x)$ を復元
2: $\hat{s} = (\hat{s}_1, \hat{s}_2, \ldots, \hat{s}_N) = \hat{f}_s(0)$ と $\hat{a} = \hat{f}_a(0)$ を計算
3: **if** $\hat{a} \neq \hat{s}_1 \boxplus \hat{s}_2 \boxplus \ldots \boxplus \hat{s}_N$ **:**
4: **return** $\perp$
5: **return** $\hat{s}$

さらに、彼らは BV 方式が OKS モデルにおいて安全であることを証明した。しかし、その証明から BV 方式が $\epsilon = 2^{-m}$ であることを保証するのは容易でない。

3 提案する攻撃

この章では、BV 方式に対する不正方法について示し、その不正成功確率について議論する。

BV 方式で使用されている Shamir の秘密分散方式は線形秘密分散であるため、BV 方式のシェアを持つ不正者はシェアを改竄することで復元された秘密に任意の値を加算 ($\mathbb{F}_{2^m}$ 上) することができる。

不正者が i_j 番目のシェア $(f_s(i_j), f_a(i_j))$ を $(f_s(i_j) \oplus \delta_{s,i_j}, f_a(i_j) \oplus \delta_{a,i_j})$ に改竄したとする。このとき、Lagrange 補完を用いて復元された秘密 $\hat{s}$ は以下のように計算される。

$$\begin{aligned}\hat{s} = \hat{f}_s(0) &= \bigoplus_{1\leq j\leq k} (f_s(i_j) \oplus \delta_{s,i_j}) l_j \\ &= \left\{ \bigoplus_{1\leq j\leq k} f_s(i_j) l_j \right\} \oplus \left\{ \bigoplus_{1\leq j\leq k} \delta_{s,i_j} l_j \right\}\end{aligned}$$

ここで上式の l_j は以下のように定義している。

$$l_j = \prod_{\substack{1\leq r\leq k \\ r\neq j}} \frac{i_r}{i_r - i_j}$$

$\Delta_s = \bigoplus_{1\leq j\leq k} \delta_{s,i_j} l_j$ とすると、以下のように表せる。

$$\hat{s} = s \oplus \Delta_s.$$

同様に、$\hat{a}$ は Lagrange 補完を用いて以下で計算される。

$$\begin{aligned}\hat{a} = \hat{f}_a(0) &= \bigoplus_{1\leq j\leq k} (f_a(i_j) \oplus \delta_{a,i_j}) l_j \\ &= \left\{ \bigoplus_{1\leq j\leq k} f_a(i_j) l_j \right\} \oplus \left\{ \bigoplus_{1\leq j\leq k} \delta_{a,i_j} l_j \right\}\end{aligned}$$

$\Delta_a = \bigoplus_{1\leq j\leq k} \delta_{a,i_j} l_j$ とする。ここで、不正者の不正成功の定義は $(\Delta_1, \ldots, \Delta_N) \neq (0, \ldots, 0)$ と Δ_a よって表される以下を満たすことである。

$$a \oplus \Delta_a = \bigboxplus_{1\leq i\leq N} (s_i \oplus \Delta_{s_i}) \tag{1}$$

3.1 不正成功確率 1 の攻撃

この節では、BV 方式に対して確率 1 で不正が成功するような方法を二つ示す。一つ目の方法は s と a 、二つ目の方法は s のみに改竄を行う。

始めに、Δ_a と Δ_{s_j} $(j \in \{1, \cdots, N\})$ の最上位ビットが 1 で、他が 0 であるとき常に不正が成功することを示す。補題 1 は最上位ビットのみが 1 のとき $\mathbb{Z}_{2^m}$ 上の加算と $\mathbb{F}_{2^m}$ 上の加算が等価であることを示している。

補題 1. $x \in \mathbb{F}_{2^m}$ とすると、以下が成り立つ。

$$x \oplus 2^{m-1} = x \boxplus 2^{m-1}$$

証明. $x_{[m]}$ を二つの場合に分けて考える。

(1) $x_{[m]} = 0$ のとき、$x = x_{[m-1:1]}$ を満たすため、以下が得られる。

$$\begin{aligned}x \boxplus 2^{m-1} &= x_{[m-1:1]} \boxplus 2^{m-1} \\ &= 1 \| x_{[m-1:1]} \\ &= x_{[m-1:1]} \oplus 2^{m-1} \\ &= x \oplus 2^{m-1}\end{aligned}$$

(2) $x_{[m]} = 1$ のとき、$x = 1 \| x_{[m-1:1]} = x_{[m-1:1]} \boxplus 2^{m-1} = x_{[m-1:1]} \oplus 2^{m-1}$ を満たすため、以下が得られる。

$$\begin{aligned}x \boxplus 2^{m-1} &= (x_{[m-1:1]} \boxplus 2^{m-1}) \boxplus 2^{m-1} \\ &= x_{[m-1:1]} \boxplus 2^{m} \\ &= x_{[m-1:1]} \\ &= (x_{[m-1:1]} \oplus 2^{m-1}) \oplus 2^{m-1} \\ &= x \oplus 2^{m-1}\end{aligned}$$

したがって $x \oplus 2^{m-1} = x \boxplus 2^{m-1}$ は常に満たす。 □

以下の補題は補題 1 から導かれる。

補題 2. $x, y \in \mathbb{F}_{2^m}$ とすると、以下が成り立つ。

$$(x \oplus 2^{m-1}) \boxplus (y \oplus 2^{m-1}) = x \boxplus y.$$

証明. 補題 1 より、以下が成り立つ。

$$\begin{aligned}(x \oplus 2^{m-1}) \boxplus (y \oplus 2^{m-1}) &= (x \boxplus 2^{m-1}) \boxplus (y \boxplus 2^{m-1}) \\ &= (x \boxplus y) \boxplus \underbrace{(2^{m-1} \boxplus 2^{m-1})}_{=0} \\ &= x \boxplus y\end{aligned}$$

□

定理 3. 不正者がいくつかの i に対して $\Delta_{s_i} = 2^{m-1}$ を選択したとする。このとき、不正者は確率 1 で不正に成功することができる。

証明. D を不正された i の数とする。$\boxplus$ は可換であるため、一般性を失わずに不正者が以下のように Δ_{s_i} を選択すると仮定できる。

$$\Delta_{s_i} = \begin{cases} 2^{m-1} & 1 \le i \le D \text{ の場合} \\ 0 & \text{その他の場合} \end{cases}$$

このとき、D が偶数か奇数によって二つの場合に分けて考える。

(1) D が偶数のとき、補題 1 と補題 2 を複数回適用することで、式 (1) の右辺は以下のようになる。

$$\begin{aligned}&\boxplus_{1 \le i \le D} (s_i \oplus 2^{m-1}) \boxplus \boxplus_{D+1 \le i \le N} s_i \\ &= \boxplus_{1 \le i \le D/2} \left((s_{2i-1} \oplus 2^{m-1}) \boxplus (s_{2i} \oplus 2^{m-1})\right) \boxplus \boxplus_{D+1 \le i \le N} s_i \\ &= \boxplus_{1 \le i \le D/2} \left((s_{2i-1} \boxplus 2^{m-1}) \boxplus (s_{2i} \boxplus 2^{m-1})\right) \boxplus \boxplus_{D+1 \le i \le N} s_i \\ &= \boxplus_{1 \le i \le N} s_i = a\end{aligned}$$

したがって、$\Delta_a = 0$ とすることで、不正成功確率 $\epsilon = 1$ となる。

(2) D が奇数のとき、補題 1 と補題 2 を複数回適用することで、式 (1) の右辺は以下のようになる。

$$\begin{aligned}&\boxplus_{1 \le i \le D} (s_i \oplus 2^{m-1}) \boxplus \boxplus_{D+1 \le i \le N} s_i \\ &= (s_1 \oplus 2^{m-1}) \boxplus \boxplus_{1 \le i \le \lfloor D/2 \rfloor} \left((s_{2i} \oplus 2^{m-1}) \boxplus (s_{2i+1} \oplus 2^{m-1})\right) \\ &\quad \boxplus \boxplus_{D+1 \le i \le N} s_i \\ &= (s_1 \boxplus 2^{m-1}) \boxplus \boxplus_{2 \le i \le N} s_i \\ &= \left(\boxplus_{1 \le i \le N} s_i\right) \boxplus 2^{m-1} = a \boxplus 2^{m-1} = a \oplus 2^{m-1}\end{aligned}$$

したがって、$\Delta_a = 2^{m-1}$ とすることで、不正成功確率 $\epsilon = 1$ となる。

上記の攻撃によって、不正成功確率 $\epsilon = 1$ となる $(\Delta_{s_1}, \Delta_{s_2}, \ldots, \Delta_{s_N})$ の組み合わせの数は $\sum_{1 \le i \le N} \binom{N}{i} = 2^N - 1$ である。 □

3.2 一般化

この節では、不正成功確率が $2^{-w_H(\Delta_{[m-1:1]})}$ となる Δ_a と Δ_{s_i} $(i \in \{1, \cdots, N\})$ の組み合わせが存在することを示す。

補題 4. $m \in \mathbb{N}, x, y, \Delta \in \mathbb{F}_{2^m}$ とすると、以下を満たす (x, y) の組み合わせは $2^{2m - w_H(\Delta_{[m-1:1]})}$ 個存在する。

$$(x \oplus \Delta) \boxplus y = (x \boxplus y) \oplus \Delta. \tag{2}$$

証明. ℓ についての数学的帰納法によって証明する。つまり、任意の $\Delta \in \mathbb{F}_{2^m}^{\le \ell}$ に対して、以下の等式を満たす $(x, y) \in (\mathbb{F}_{2^m}^{\le \ell})^2$ の組み合わせが $2^{2\ell - w_H(\Delta)}$ 個存在することを示す。

$$(x \oplus \Delta) \boxplus y = (x \boxplus y) \oplus \Delta \tag{3}$$

$\ell = 1$ のとき、Δ は 0 または 1 である。$\Delta = 0$ の場合、任意の $(x, y) \in (\mathbb{F}_{2^m}^{\le 1})^2 (= \{0, 1\}^2)$ に対して $(x \oplus 0) \boxplus y = (x \boxplus y) \oplus 0$ が明らかに成り立つため、式 (3) を満たす (x, y) の組み合わせは $2^2 = 2^{2 \cdot 1 - w_H(0)}$ 個存在する。$\Delta = 1$ の場合、以下の等式が成り立つ。

$$(x \oplus 1) \boxplus y = \begin{cases} x \boxplus y \boxplus 1 & x = 0 \text{ の場合} \\ x \boxplus y \boxplus -1 & x = 1 \text{ の場合} \end{cases}$$

$$(x \boxplus y) \oplus 1 = \begin{cases} x \boxplus y \boxplus 1 & x = y \text{ の場合} \\ x \boxplus y \boxplus -1 & x \ne y \text{ の場合} \end{cases}$$

よって、$(x \oplus 1) \boxplus y = (x \boxplus y) \oplus 1$ は (x, y) が $(0, 0)$ または $(1, 0)$ のときに成り立つため、式 (3) を満たす (x, y) の組み合わせは $2^2 = 2^{2 \cdot 1 - w_H(1)}$ 個存在する。

両方の場合において、式 (3) を満たす (x, y) の組み合わせは $2^{2 \cdot 1 - w_H(\Delta)}$ 個存在する。

任意の $\Delta \in \mathbb{F}_{2^m}^{\le k}$ $(k < m - 1)$ について、式 (3) を満たす $(x, y) \in (\mathbb{F}_{2^m}^{\le k})^2$ の組み合わせは $2^{2k - w_H(\Delta_{[k:1]})}$ 個存在すると仮定する。

$\Delta \in \mathbb{F}_{2^m}^{\le k}$ について、$\mathcal{Z}_{k,\Delta}$ を以下のように定義する。

$$\mathcal{Z}_{k,\Delta} = \left\{(x, y) \in (\mathbb{F}_{2^m}^{\le k})^2 \mid x, y, \Delta \text{ は } \ell = k \text{ のとき、式 (3) を満たす}\right\}$$

仮定より、任意の $\Delta \in \mathbb{F}_{2^m}^{\le k}$ について、$|\mathcal{Z}_{k,\Delta}| = 2^{2k - w_H(\Delta)}$ を満たす。$c \in \{0, 1\}$ に対して以下のように $\mathcal{Z}_{k,\Delta}^{(c)}$ を定義する。

$$\mathcal{Z}_{k,\Delta}^{(c)} = \left\{(x, y) \in \mathcal{Z}_{k,\Delta} \mid ((x \oplus \Delta) \boxplus y)_{[k+1]} = c\right\}$$

$\mathcal{Z}_{k,\Delta}^{(c)}$ は式 (3) を満たすような (x, y) の集合であり、$(x \oplus \Delta) \boxplus y$ の $(k+1)$ 番目のビット (および $(x \boxplus y) \oplus \Delta$ の $(k+1)$ 番

目のビット) が c と等しいことに注意する。そして、c_k を $(x \oplus \Delta) \boxplus y$ の $(k+1)$ 番目のビットを表すために用いる。明らかに、$\mathcal{Z}^{(0)}_{k,\Delta} \cap \mathcal{Z}^{(1)}_{k,\Delta} = \emptyset$ と $|\mathcal{Z}^{(0)}_{k,\Delta}| + |\mathcal{Z}^{(1)}_{k,\Delta}| = 2^{2k-w_H(\Delta)}$ を満たす。

ここで、任意の $\Delta \in \mathbb{F}_{2^m}^{\leq k+1}$ について $|\mathcal{Z}_{k+1,\Delta}| = 2^{2(k+1)-w_H(\Delta_{[k+1:1]})}$ が成り立つことを示す。

$\Delta \in \mathbb{F}_{2^m}^{\leq k+1}$、$(x_{[k:1]}, y_{[k:1]}) \in \mathcal{Z}_{k,\Delta}$ を満たすような $(x, y) \in (\mathbb{F}_{2^m}^{\leq k+1})^2$ とする。$(x_{[k:1]}, y_{[k:1]}) \in \mathcal{Z}_{k,\Delta}$ であるため、以下が成り立つ。

$$((x \oplus \Delta) \boxplus y)_{[k:1]} = ((x \boxplus y) \oplus \Delta)_{[k:1]}$$

したがって、以下が成り立つとき、$(x, y) \in \mathcal{Z}_{k+1,\Delta}$ となる。

$$((x \oplus \Delta) \boxplus y)_{[m:k+1]} = ((x \boxplus y) \oplus \Delta)_{[m:k+1]}$$

$((x \oplus \Delta) \boxplus y)_{[m:k+1]}$ と $((x \boxplus y) \oplus \Delta)_{[m:k+1]}$ は以下のように計算される。

$$((x \oplus \Delta) \boxplus y)_{[m:k+1]} = (x_{[k+1]} \oplus \Delta_{[k+1]}) \boxplus y_{[k+1]} \boxplus c_k \quad (4)$$

$$((x \boxplus y) \oplus \Delta)_{[m:k+1]} = (x_{[k+1]} \boxplus y_{[k+1]} \boxplus c_k) \oplus \Delta_{[k+1]} \quad (5)$$

$\Delta_{[k+1]}$ が 0 か 1 かによって、二つの場合に分けて考える。

(1) $\Delta_{[k+1]} = 0$ のとき、式 (4) と式 (5) の両方の右辺は $x_{[k+1]} \boxplus y_{[k+1]} \boxplus c_k$ となり、$x_{[k+1]}, y_{[k+1]}$ および c_k の値に関係なく $(x \oplus \Delta) \boxplus y = (x \boxplus y) \oplus \Delta$ が成り立つ。したがって、以下が成り立つ。

$$\begin{aligned} |\mathcal{Z}_{k+1,\Delta}| &= |\{0,1\}^2| \times |\mathcal{Z}_{k,\Delta_{[k:1]}}| \\ &= 2^2 \cdot 2^{2k-w_H(\Delta_{[k:1]})} \\ &= 2^{2(k+1)-w_H(0\|\Delta_{[k:1]})} \;=\; 2^{2(k+1)-w_H(\Delta)} \end{aligned}$$

(2) $\Delta_{[k+1]} = 1$ のとき、表 1 に式 (4) と式 (5) の両方の右辺の計算結果を示す。

表 1　$\Delta_{[k+1]} = 1$ における $(x_{[k+1]} \oplus \Delta_{[k+1]}) \boxplus y_{[k+1]} \boxplus c_k$ と $(x_{[k+1]} \boxplus y_{[k+1]} \boxplus c_k) \oplus \Delta_{[k+1]}$ の計算結果

$x_{[k+1]}$	$y_{[k+1]}$	c_k	$(x_{[k+1]} \oplus 1) \boxplus y_{[k+1]} \boxplus c_k$	$(x_{[k+1]} \boxplus y_{[k+1]} \boxplus c_k) \oplus 1$
0	0	0	1	1
1	0	0	0	0
0	1	0	10	0
1	1	0	1	11
0	0	1	10	0
1	0	1	1	11
0	1	1	11	11
1	1	1	10	10

表 1 より、$c_k = 0$ のとき、式 (4) と式 (5) の両方の右辺が等しいのは $(x_{[k+1]}, y_{[k+1]})$ が $(0,0)$ または $(1,0)$ のときである。また、$c_k = 1$ のとき、式 (4) と式 (5) の両方の右辺が等しいのは $(x_{[k+1]}, y_{[k+1]})$ が $(0,1)$ または $(1,1)$ のときである。

以上から数学的帰納法より、$\ell = m-1$ のとき、式 (3) が成り立つ。したがって、以下が成り立つ。

$$\begin{aligned} |\mathcal{Z}_{k+1,\Delta}| &= |\{(0,0),(1,0)\}| \times |\mathcal{Z}^{(0)}_{k,\Delta_{[k:1]}}| \\ &\quad +|\{(0,1),(1,1)\}| \times |\mathcal{Z}^{(1)}_{k,\Delta_{[k:1]}}| \\ &= 2 \cdot (|\mathcal{Z}^{(0)}_{k,\Delta_{[k:1]}}| + |\mathcal{Z}^{(1)}_{k,\Delta_{[k:1]}}|) \\ &= 2 \cdot |\mathcal{Z}_{k,\Delta_{[k:1]}}| \\ &= 2^{2(k+1)-(w_H(1\|\Delta_{[k:1]})+1)} \;=\; 2^{2(k+1)-w_H(\Delta)} \end{aligned}$$

ここまでで、任意の $\Delta \in \mathbb{F}_{2^m}^{\leq \ell}$ $(1 \leq \ell \leq m-1)$ について、$|\mathcal{Z}_{\ell,\Delta}| = 2^{2\ell - w_H(\Delta)}$ が成り立つことを証明した。以降で $|\mathcal{Z}_{m,\Delta}| = 2^{2m-w_H(\Delta_{[m-1:1]})}$ を証明する。(x, y) を $(x_{[m-1:1]}, y_{[m-1:1]}) \in \mathcal{Z}_{m-1,\Delta_{[m-1:1]}}$ のような $(\mathbb{F}_{2^m})^2$ の要素とする。このとき、$(x_{[m-1:1]} \oplus \Delta_{[m-1:1]}) \boxplus y_{[m-1:1]} = (x_{[m-1:1]} \boxplus y_{[m-1:1]}) \oplus \Delta_{[m-1:1]}$ が成り立つ。$z = (x_{[m-1:1]} \oplus \Delta_{[m-1:1]}) \boxplus y_{[m-1:1]}$ とすると、以下が成り立つ。

$$\begin{aligned} (x \oplus \Delta) \boxplus y &= ((x_{[m]} \oplus \Delta_{[m]}) \boxplus y_{[m]} \boxplus c_{m-1}) 2^{m-1} \\ &\quad \boxplus (z \boxplus -2^{m-1} c_{m-1}) \\ (x \boxplus y) \oplus \Delta &= ((x_{[m]} \boxplus y_{[m]} \boxplus c_{m-1}) \oplus \Delta_{[m]}) 2^{m-1} \\ &\quad \boxplus (z \boxplus -2^{m-1} c_{m-1}) \end{aligned}$$

任意の $a, b \in \{0,1\}$ について、$(a \oplus b) 2^{m-1} = (a \boxplus b) 2^{m-1}$ が成り立つため、$((x_{[m]} \oplus \Delta_{[m]}) \boxplus y_{[m]} \boxplus c_{m-1}) 2^{m-1}$ と $((x_{[m]} \boxplus y_{[m]} \boxplus c_{m-1}) \oplus \Delta_{[m]}) 2^{m-1}$ は $x_{[m]}, y_{[m]}, \Delta_{[m]}$ および c_{m-1} に関わらず等しい。したがって、任意の $\Delta \in \mathbb{F}_{2^m}$ について、式 (3) を満たす $(x, y) \in (\mathbb{F}_{2^m})^2$ は $2^2 \cdot 2^{2(m-1)-w_H(\Delta_{[m-1:1]})} = 2^{2m-w_H(\Delta_{[m-1:1]})}$ 個存在する。 □

定理 5. 不正者が一つの i に対して $\Delta_{s_i} = \Delta_a$ を選択したとする。このとき、不正者は確率 $2^{-w_H(\Delta_{a[m-1:1]})}$ で不正に成功する。

証明. $\boxplus$ は可換であるため、一般性を失わずに不正者が以下のように Δ_{s_i} を選択すると仮定できる。

$$\Delta_{s_i} = \begin{cases} \Delta_a & (i = 1) \\ 0 & \text{その他の場合} \end{cases}$$

$S = \boxplus_{2 \leq i \leq N} s_i$ とすると、式 (1) の右辺は $(s_1 \oplus \Delta_a) \boxplus S$ となる。さらに、補題 4 より、以下の式を満たす (s_1, S) の組み合わせは $2^{2m-w_H(\Delta_{a[m-1:1]})}$ 個存在する。

$$(s_1 \oplus \Delta_a) \boxplus S = (s_1 \boxplus S) \oplus \Delta_a \quad (6)$$

式 (6) が成り立つとき、式 (1) の右辺は以下のように変形できる。

$$(s_1 \oplus \Delta_a) \boxplus S = (s_1 \boxplus S) \oplus \Delta_a = a \oplus \Delta_a$$

(s_1, S) の組み合わせは 2^{2m} 個存在する。したがって、式 (1) が成り立つ確率は $2^{2m-w_H(\Delta_{a[m-1:1]})} \cdot 2^{-2m} = 2^{-w_H(\Delta_{a[m-1:1]})}$ となる。 □

補題 6. $m \in \mathbb{N}, x, y, \Delta \in \mathbb{F}_{2^m}$ とすると、以下を満たす (x, y) の組み合わせは $2^{2m-w_H(\Delta_{[m-1:1]})}$ 個存在する。

$$(x \oplus \Delta) \boxplus (y \oplus \Delta) = x \boxplus y \tag{7}$$

補題 6 の証明は補題 4 と同様であるため省略する。

定理 7. $\Delta \in \mathbb{F}_{2^m}$ とする。不正者が二つの i に対して $\Delta_{s_i} = \Delta$ を選択したとする。このとき、不正者は確率 $2^{-w_H(\Delta_{s_i[m-1:1]})}$ で不正に成功する。

定理 7 は補題 6 を用いて定理 5 と同様に証明することができるため、本論文では定理 7 の証明を省略する。

定理 3、定理 5、定理 7 より、更に一般的な場合についても不正に成功する確率を求めることができる。

系 8. 不正者が一つの i に対して $\Delta_{s_i} = \Delta_a$ または $\Delta_{s_i} = \Delta_a \oplus 2^{m-1}$ を選択し、いくつかの j に対して $\Delta_{s_j} = 2^{m-1}$ を選択したとする。このとき、不正者は確率 $2^{-w_H(\Delta_{a[m-1:1]})}$ で不正に成功する。

系 8 の証明は本論文では省略する。

系 9. $\Delta \in \mathbb{F}_{2^m}$ とする。不正者が二つの i に対して $\Delta_{s_i} = \Delta$ を選択し、いくつかの j に対して $\Delta_{s_j} = 2^{m-1}$ を選択したとする。このとき、不正者は確率 $2^{-w_H(\Delta_{[m-1:1]})}$ で不正に成功する。

系 9 の証明は本論文では省略する。

系 8 と系 9 の $\Delta_{s_i} = 2^{m-1}$ の場合、各系は定理 3 と等価であり、不正に成功する確率は $2^{-w_H(\Delta_{[m-1:1]})}$ である。このことから、一つ目の攻撃（定理 3）は、二つ目の攻撃（系 8, 9）の特別な場合とみなすことができる。

3.3 不正の例 (定理 5)

分散される秘密 $s = s_1 \| s_2$ を $\mathbb{F}_{2^{3\cdot 2}}$ の要素とする。$\boxplus$ を $\mathbb{Z}_{2^3}$ 上の加算と定義し、$a = s_1 \boxplus s_2$ とする。不正者が定理 5 に記述された攻撃を行う状況を考える。表 2 は定理 5 の不正に成功するための条件を満たす Δ と s の値をまとめたものである。定理と結果が一致していることが確認できる。

表 2　Δ と $(s_1 \oplus \Delta) \boxplus s_2 = (s_1 \boxplus s_2) \oplus \Delta$ を満たす確率の関係

Δ	$(s_1 \oplus \Delta) \boxplus s_2 = (s_1 \boxplus s_2) \oplus \Delta$ を満たす $s(= s_1 \| s_2)$	$w_H(\Delta_{[2:1]})$	ϵ
001 101	000000 000010 000100 000110 001000 001010 001100 001110 010000 010010 010100 010110 011000 011010 011100 011110 100000 100010 100100 100110 101000 101010 101100 101110 110000 110010 110100 110110 111000 111010 111100 111110	1	$2^{-1}\left(=\frac{32}{2^6}\right)$
010 110	000000 000001 000100 000101 001000 001011 001100 001111 010000 010001 010100 010101 011000 011011 011100 011111 100000 100001 100100 100101 101000 101011 101100 101111 110000 110001 110100 110101 111000 111011 111100 111111	1	$2^{-1}\left(=\frac{32}{2^6}\right)$
011 111	000000 000100 001000 001100 010000 010100 011000 011100 100000 100100 101000 101100 110000 110100 111000 111100	2	$2^{-2}\left(=\frac{16}{2^6}\right)$
100	任意の $s \in \mathbb{F}_{2^{3\cdot 2}}$	0	$2^{-0}\left(=\frac{64}{2^6}\right)$

4 結論

本研究では、$\mathbb{F}_{2^m}$ と $\mathbb{Z}_{2^m}$ 上の加法の類似性に注目し、BV 方式に対する二つの攻撃を示した。不正者がこれらの攻撃を用いることで、$\epsilon = 2^{-m}$ よりも高い確率で復元アルゴリズムへの不正を成功することができる。一つ目の攻撃は、秘密 $s = (s_1, \ldots, s_N)$ の複数の s_i の最上位ビットに対して任意の操作ができる。この攻撃は確率 1 で不正に成功する。二つ目の攻撃は、一つ目の攻撃を一般化した方法であり、一つ目の攻撃に加えて最大二つの s_i の任意のビットに対して操作ができる。この攻撃は確率 $2^{-w_H(\Delta_{[m-1:1]})}$ ($w_H(\Delta_{[m-1:1]}) < m$ が明らかに成り立つ) で不正に成功する。よって、BV 方式は $\epsilon = 2^{-m}$ よりも有意に高い不正成功確率を持つため、BV 方式が OKS モデルにおいて安全であるとは言えない。

参考文献

[1] D. Becerra and G. Vega. Secret sharing scheme with efficient cheating detection. In *Proceedings of the 2nd International Conference on Networking, Information Systems & Security*, number Article 5 in NISS19, pages 1–7. Association for Computing Machinery, Mar. 2019.

[2] G. R. Blakley. Safeguarding cryptographic keys. In *1979 International Workshop on Managing Requirements Knowledge (MARK)*, pages 313–318, June 1979.

[3] M. Carpentieri, A. De Santis, and U. Vaccaro. Size of shares and probability of cheating in threshold schemes. In *Advances in Cryptology — EUROCRYPT '93*, Lecture notes in computer science, pages 118–125. Springer Berlin Heidelberg, Berlin, Heidelberg, 1994.

[4] H. Hoshino and S. Obana. Almost optimum secret sharing schemes with cheating detection for random bit strings. In *Proceedings of the 10th International Workshop on Advances in Information and Computer Security - Volume 9241*, IWSEC 2015, pages 213–222, Berlin, Heidelberg, Aug. 2015. Springer-Verlag.

[5] H. Hoshino and S. Obana. Cheating detectable secret sharing scheme suitable for implementation. In *2016 Fourth International Symposium on Computing and Networking (CANDAR)*, pages 623–628, Nov. 2016.

[6] W. Ogata, K. Kurosawa, and D. R. Stinson. Optimum secret sharing scheme secure against cheating. *SIAM J. Discrete Math.*, 20(1):79–95, Jan. 2006.

[7] A. Shamir. How to share a secret. *Commun. ACM*, 22(11):612–613, Nov. 1979.

[8] M. Tompa and H. Woll. How to share a secret with cheaters. *J. Cryptology*, 1(3):133–138, Oct. 1989.