CL-005

排他的論理和を用いた消失訂正符号に基づく条件付き閾値秘密分散

Conditional threshold secret sharing based on erasure code using XOR

今井 淳太 *　三村 守 *　田中 秀磨 *
Junta Imai　Mamoru Mimura　Hidema Tanaka

1. はじめに

秘密情報 S を漏洩・破壊・故障などに対して安全に保管する手法として、秘密分散がある。S を n 個のシェア W_i $(i \in \{0,1,\cdots,n\})$ に分散し、そのうちの k 個 $(k \leq n)$ 以上のシェアを集めることで元の秘密情報 S を復元できる時、(k,n) 閾値秘密分散と呼ぶ。この時、各シェア W_i から元の秘密情報 S を推測できないこと、k 個未満のシェアから秘密情報 S の復元もしくは推測できないことが主な安全性要件となる。秘密分散の使用にあたり、シェアを配布するメンバー間に重みをもたせる条件を与える場合がある。簡潔な方法としては重要なメンバー (VIP) に複数個のシェアを渡すことで実現できるが、配布されたシェア数により各メンバーは自分が VIP であるかどうかを知ることが可能となる。また、シェアを配布する各メンバー同士に結託する可能性が場合は、そのメンバーを 1 つのグループとしてシェアを複数個持っているとみなすことができる。これらのことは結託攻撃の実現が容易になる問題を生じさせる。そこで本稿では、秘密情報のシェアの組み合わせに重みをつけ、k 個のシェアの組み合わせ次第では元の秘密情報が復元できなくなる条件付き (k,n) 閾値秘密分散を提案する。この方法においては、k 個のシェアを集めるほかに、ある条件を満たさなければ元の秘密情報を復元することはできない。本論文で設定する条件は以下の 2 種類である。1 つ目は、シェアを配布するメンバーに VIP が存在するパターンである。この条件では、秘密情報を復元する際に必ず VIP の持つシェアが必要となる。2 つ目は、各メンバーがいずれかのグループに属するパターンである。この条件では、各グループから最低 1 人のメンバーのシェアがなければ復元することができない。

本研究ではこれを実現するため、消失訂正符号の 1 つであるチェーン符号を応用する。消失訂正符号は符号語のうち、いくつかのビットが消失しても訂正できる符号の構成法である。しかしながら、ある一定数以上の消失誤りがあった場合は復元できない。また、消失するビットの位置に応じて復号可能な故障耐性が変わるという特徴がある。これらの消失訂正符号の特徴を用いることで、上述した安全性要件を満たした条件付き (k,n) 閾値秘密分散の構成を提案する。

2. チェーン符号 [1]

本研究では消失訂正符号の 1 つであるチェーン符号に注目する。消失訂正符号は、s 個の n[bit] データ要素 $d_i (i \in \{0,1,\cdots,s\})$ と t 個の冗長な n[bit] パリティ要素 $p_j (j \in \{0,1,\cdots,t\})$ で構成される。符号内のある要素が消失した場合には、他のいくつかの要素を用いて復元させることのできる符号である。この際、符号語間のハミング距離が d であるならば、最低でも $d-1$ 個までの要素の同時消失に耐えうることができる。このことを、故障耐性が $d-1$ であると表現する。

パリティ要素の生成には様々な方法があるが、本稿では、いくつかのデータ要素の XOR 演算により算出する方法を用いる。このパリティ要素を算出する式を、パリティ方程式と呼ぶ。消失訂正符号の構築に XOR を用いた場合、ハミング距離 d を達成するためには、各データ要素が最低でも $d-1$ 個のパリティ方程式に含まれていなければならない。

チェーン符号の構造は $s=t$ となっており、各パリティ要素 p_j は、$d-1$ 個のデータ要素の XOR となっている。すなわち、p_j のパリティ方程式は以下の式で与えられる。

$$p_j = \bigoplus_{i=j}^{j+d-2} d_{(i \bmod s)} \tag{1}$$

このようにして構成されたチェーン符号は、非最大距離離散符号となる。非最大距離離散符号においては、各符号間のハミング距離に差が生じる。このため、各要素を復元するための回復方程式のサイズに違いが発生する。回復方程式のサイズとは、その方程式に含まれる項数で定義される。以下に、$(s,t,d)=(3,3,3)$ の場合の各要素の

*防衛大学校情報工学科

回復方程式を列挙する。ここで、故障耐性は 2 である。

$$
\begin{aligned}
&d_0 = d_1 \oplus p_0,\ d_2 \oplus p_0 \oplus p_1,\ d_2 \oplus p_2,\ d_1 \oplus p_1 \oplus p_2\\
&d_1 = d_0 \oplus p_0,\ d_2 \oplus p_1,\ d_2 \oplus p_0 \oplus p_2,\ d_0 \oplus p_1 \oplus p_2\\
&d_2 = d_1 \oplus p_1,\ d_0 \oplus p_0 \oplus p_1,\ d_0 \oplus p_2,\ d_1 \oplus p_0 \oplus p_2\\
&p_0 = d_0 \oplus d_1,\ d_0 \oplus d_2 \oplus p_1,\ d_1 \oplus d_2 \oplus p_2,\ p_1 \oplus p_2\\
&p_1 = d_1 \oplus d_2,\ d_0 \oplus d_2 \oplus p_0,\ d_0 \oplus d_1 \oplus p_2,\ p_0 \oplus p_2\\
&p_2 = d_0 \oplus d_2,\ d_1 \oplus d_2 \oplus p_0,\ d_0 \oplus d_1 \oplus p_1,\ p_0 \oplus p_1
\end{aligned}
\tag{2}
$$

上記の例では、例えば d_0 の要素が消失した場合は $\{d_1, p_0\}$、$\{d_2, p_0, p_1\}$、$\{d_2, p_2\}$ または $\{d_1, p_1, p_2\}$ のいずれかの組み合わせの XOR により、d_0 を復元することができる。この例では回復方程式のサイズは、2 または 3 となっている。

消失訂正符号が最大距離離散符号である場合、もし故障耐性が 2 であれば要素の消失は 2 個までしか耐えることができない。しかし、非最大距離離散符号である場合は、消失した要素の組み合わせによっては 3 個以上の消失にも耐えることが可能となる。これは、上述したように回復方程式のサイズが異なるためである。

3. 安全性要件と提案手法の概要

一般的に、秘密分散に求められる安全性要件は以下にまとめられる。

要件 1. 各シェア W_i から秘密情報 S を予測できない
要件 2. 設定した閾値 n 未満では秘密情報を復元できない

これに加えて、本研究では VIP を設定した場合、以下の要件を加える。

要件 3. VIP が持つシェアが無ければ秘密情報 S を復元できない

VIP は既に本人が自覚もしくはディーラから伝えられるケースも考えられるが、VIP の持つシェアのサイズを各シェアと等しくし、あらかじめ本人が VIP であるか伝えないことで以下の利点が生じる。

- 誰が VIP であるか本人であっても各シェアから予測できない。
- VIP が管理するデータが増えないことで、運用上のリスクを低減できる。
- 誰が VIP であるか不明なため結託などの不正が未然に防げ、運用上のリスクを低減できる。
- 誰を VIP にするのかの決定を TTP であるディーラが決定できる。

要件 3 をさらに拡大することで、グループ単位でのシェアも可能となる。この場合、以下の要件を加える。これは議事における決選投票的な意味合いがある。

要件 4. 各グループから最低 1 つのシェアが無ければ秘密情報 S を復元できない

本提案手法では、要件 1 を満たすためにバーナム暗号の仕組みを採用する。例えば、$(s,t,d) = (3,3,3)$ の時、平文 P、乱数 R_1、$R_2(|P| = |R_1| = |R_2|)$ として、以下のように秘密情報 S を定義する。

$$
S = (d_0||d_1||d_2)\ ,\begin{cases} d_0 = P \oplus R_1 \oplus R_2 \\ d_1 = R_1 \\ d_2 = R_2 \end{cases} \tag{3}
$$

各ビット位置の入れ替えを行い、これを秘密情報とすることで、単純に復元された秘密情報 S から平文 P を得ることが情報理論的に不能にすることもできる。要件 2 を満たすため、第 2 節で述べたチェーン符号を応用する。前述したように、チェーン符号が消失符号語を復元できる条件は故障耐性で示される。逆に言えば、故障耐性分だけあらかじめ符号語を破棄し、残りをシェアとすることで容易に秘密分散として機能する。

一方で、チェーン符号は非最大距離離散符号である。この性質から導かれる回復方程式のサイズの違いを利用することで、要件 3 及び要件 4 を満たすことができる。具体的な構成法を第 4 節に示す。

4. 提案手法

4.1 概要

この節では、チェーン符号をもとに条件付き秘密分散を実現するための、提案する 3 つの構成方法を説明する。前節で説明したように、チェーン符号は非最大距離離散符号であるため、消失した要素の数が同じであっても、その組み合わせによって復元が可能な場合と不可能な場合とに分かれる。この違いを利用することにより、メンバーに配布するシェアの組み合わせに重みをつけている。

例として、$3n$[bit] 入力 S に対する、$(s,t,d) = (3,3,3)$ における構成を示す。

$$
S \ = \ d_0 \ || \ d_1 \ || \ d_2 \tag{4}
$$

と分割し、これを元に (1) を用いて p_0、p_1 及び p_2 を作成する。従って、符号語として 6 個の n[bit] 符号語 $\{d_0\,, d_1\,, d_2\,, p_0\,, p_1\,, p_2\}$ が得られる。この時、$d = 3$ であるため、6 個の符号語から任意の 2 個の符号語が消失しても全体を復元できる。しかし、3 個の符号語が消失した場合には、その組み合わせによって復元ができる場合

とできない場合がある。図 1 に、復元が不可能となる組み合わせを示す。$(s,t,d)=(3,3,3)$ の時、図に示すように 4 通りの組み合わせが存在する。この図から、各符号語をシェアとしてメンバーに配布すれば、条件付き閾値秘密分散を実現することができる。以降では、次の 3 つのパターンを実現させる方法を説明する。

提案手法 1：閾値秘密分散
提案手法 2：メンバーの中に VIP が存在する場合
提案手法 3：各メンバーがそれぞれ 3 つのグループに所属

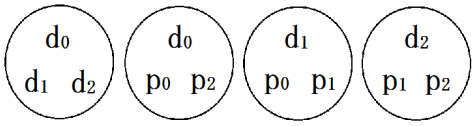

図 1. $(s,t,d)=(3,3,3)$ で復元不能になる符号語の組み合わせ

4.2 提案手法 1

秘密情報の復元に条件が存在しない、通常の閾値秘密分散の構築方法を示す。$d=3$ として構成されたチェーン符号は、任意の 2 個の符号語の消失には耐えることができる。そこで、ディーラーは 6 個の符号語から 2 個の符号語を予め取り除き、残りの 4 個の符号語をシェアとして配布することを考える。取り除く 2 個の符号語を選ぶ際、各人のシェアに重みをつけないようにするためには、図 1 で示した各組み合わせの中からそれぞれ 1 個の符号語が消失するように選べばよい。そのような符号語の組み合わせは、$\{d_0,p_1\}$、$\{d_1,p_2\}$ または $\{d_2,p_0\}$ の 3 通り存在する。例として、$\{d_0,p_1\}$ を取り除いたものを図 2 に示す。丸で囲まれた各組の中にはそれぞれ 2 個の符号語が存在しているので、ディーラーが配布した 4 個の符号語 $\{d_1,d_2,p_0,p_2\}$ から任意の 1 個の符号語が消失したとしても、残りの 3 個の符号語から消失したすべての符号語を復元することができる。すなわち、$\{d_1,d_2,p_0,p_2\}$ をシェアとした $(3,4)$ 閾値秘密分散を実現できる。ここで TTP は、各シェア $W_j(j\in\{0,1,2,3\})$ に $d_i(i\in\{1,2\})$ 及び $p_h(h\in\{0,2\})$ を割り当て、4 人のメンバーに配布する。各シェア W_i にどの符号語が割り振られているかは秘密とする。

秘密情報 S の復元には、W_i が符号語として取り得る全ての組み合わせに対する全数探索を実行する。例えば、

$$W_0=d_1,\quad W_1=d_2,\quad W_2=p_0,\quad W_3=p_2 \tag{5}$$

として、d_0 及び p_1 の一時的な候補 d_0' 及び p_1' を (2) を用いて復元する。復元できた情報 S' に対して、式 (3) の処理の逆を実行し、平文候補 P' を得る。すなわち、

$$\begin{aligned} P' &= d_0'\oplus R_1\oplus R_2 \\ &= d_0'\oplus W_0\oplus W_1 \end{aligned} \tag{6}$$

この平文 P' が semantics であれば成功であり、そうでなければ繰り返す。全会一致（4 つのシェアが集まった場合）の時は、各メンバーに配布されるシェアとして $\frac{6!}{2!}$ 通りの組み合わせがある。1 個の消失した符号語を復元するのに最大で 2 回の XOR 演算を必要とし、平文候補 P' を計算するのに 2 回の XOR 演算を必要とするので、全体の計算量としては $\frac{6!}{2!}\cdot((2\cdot2)+2)$ 回の XOR 演算が必要となる。次に、(3,4) の場合の復元を考える。集まったシェアは 3 つなので、シェアの組み合わせとしては $\frac{6!}{3!}$ 通りである。消失した符号語は 3 個あるので、全体の計算量としては $\frac{6!}{3!}\cdot((2\cdot3)+2)$ 回の XOR 演算が必要となる。ここで注目すべきは、平文 P、秘密情報 S、シェア W_i のサイズにかかわらず、組み合わせ数及び計算量が確定的に決定されることである。ただし、消失した条件を与える符号語の選び方によっては、上述した XOR 演算の回数は異なる。これは式 (2) に示したように回復方程式のサイズが異なるからである。上述した例は、各符号語の復元にサイズが 3 の回復方程式を利用した場合である。実際にはすべての符号語の復元に最大サイズの回復方程式を利用することはないため、最大でも $\frac{6!}{2!}\cdot((2\cdot2)+2)$ 回以下の XOR 演算により計算されることがわかる。

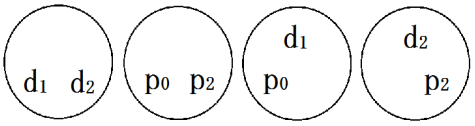

図 2. 提案手法 1 の構成準備

4.3 提案手法 2

メンバーの中に VIP が存在する閾値秘密分散の構築方法を説明する。通常の閾値秘密分散と同じように、2 個の符号語を取り除いたものをシェアとして配布するが、VIP を作るためには取り除く符号語に偏りをもたせなければならない。このような 2 個の組み合わせは全部で 12 通り存在するが、一例として、$\{d_0,d_1\}$ を取り除いたものを図 3 に示す。図の一番左の組をみると、d_2 しか残っていないのが確認できる。$\{d_0,d_1\}$ は既に消失しているので、ここからさらに d_2 が消失すれば符号語全体の復元は不可能となる。つまり、$\{d_2,p_0,p_1,p_2\}$ をシェアとして配布した場合、d_2 は全体の復元のために必ず必要なシェアとなる。すなわち、d_2 を配られたメンバーを VIP とした $(3,4)$ 閾値秘密分散となっている。

秘密情報 S の復元には必ず VIP が参加していなければならない。

$$W_i\in\{d_2,p_0,p_1,p_2\}\ (i\in\{0,1,2,3\}),\ W_j\neq W_k\ (j\neq k) \tag{7}$$

とした時、全会一致で全てのシェアが揃っている場合は復元の手順は提案手法 1 と同様である。$(3,4)$ の場合の復元も同様であるが、VIP が含まれていない時、平文候補 P' が semantics を持たず、復号に失敗する。このように、VIP が存在するという条件を与えても、提案手法 1 と同様の手順であり、組み合わせ回数及び計算量は同様に確定的に決定される。

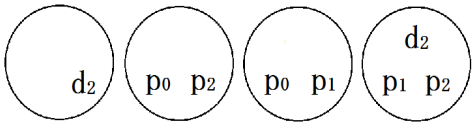

図 3. 提案手法 2 の構成準備

4.4 提案手法 3

メンバーがいくつかあるグループに所属しているとする。秘密情報の復元にはすべてのグループから最低 1 つのシェアを必要とし、かつ、閾値を満たさなければならないという条件を与える。必要なシェア W_i が増えるので、$(s,t,d)=(4,4,3)$ を例に構成を示す。式 (4) を拡張し、式 (1) を用いることで以下のような符号語が得られる。

$$S=(d_0||d_1||d_2||d_3)\ ,\begin{cases} d_0 = P \oplus R_1 \oplus R_2 \oplus R_3 \\ d_1 = R_1 \\ d_2 = R_2 \\ d_3 = R_2 \end{cases} \tag{8}$$

$$(p_0,p_1,p_2,p_3)=(d_0\oplus d_1, d_1\oplus d_2, d_2\oplus d_3, d_3\oplus d_0) \tag{9}$$

各符号語の回復方程式の一部を以下に示す。

$$\begin{aligned} d_0 &= d_1\oplus p_0,\ d_3\oplus p_3,\ d_2\oplus p_0\oplus p_1,\cdots \\ d_1 &= d_0\oplus p_0,\ d_2\oplus p_1,\ d_3\oplus p_0\oplus p_3,\cdots \\ d_2 &= d_1\oplus p_1,\ d_3\oplus p_2,\ d_0\oplus p_0\oplus p_1,\cdots \\ d_3 &= d_2\oplus p_2,\ d_0\oplus p_3,\ d_1\oplus p_1\oplus p_2,\cdots \\ p_0 &= d_0\oplus d_1,\ d_1\oplus d_3\oplus p_3,\ d_0\oplus d_2\oplus p_1,\cdots \\ p_1 &= d_1\oplus d_2,\ d_0\oplus d_2\oplus p_0,\ d_1\oplus d_3\oplus p_2,\cdots \\ p_2 &= d_2\oplus d_3,\ d_1\oplus d_3\oplus p_1,\ d_0\oplus d_2\oplus p_3,\cdots \\ p_3 &= d_0\oplus d_3,\ d_1\oplus d_3\oplus p_0,\ d_0\oplus d_2\oplus p_2,\cdots \end{aligned} \tag{10}$$

図 4 に復元が不可能となる符号語の組み合わせを示す。図中の斜線は $\{d_0,d_2\}$ を取り除いた場合を意味する。この時、1 組、3 組及び 5 組に注目すると、$\{d_0,d_2\}$ 以外の符号語が重複なく含まれている。従って、以下の 3 つのグループへそれぞれの符号語をシェアとして配布する。

グループ A：$\{p_0,p_3\}$
グループ B：$\{p_1,p_2\}$
グループ C：$\{d_1,d_3\}$

この例からは 3 つのグループ A、B、C に対し、各グループから最低 1 つのシェアが得られ、かつ総数が 4 以上にならなければ秘密情報 S が復元できない。

提案手法 1 及び 2 と同様にディーラーが各メンバーにシェアを割り振ったとする。各メンバーがどのグループに属するかを既知とするか未知とするかは、TTP が自由に決定できる。シェアが 6 つ集まった時、秘密情報 S の復元に最も計算量が必要となる。各シェアの符号語への割り振れる組み合わせ数は $\frac{8!}{2!}$ である。前述したように、シェアのサイズに関わらず計算量は XOR 演算回数で見積もることができる。回復方程式の最大サイズは 5 であり、P の復号に 3 回の XOR 演算を行うので、最大 $\frac{8!}{2!}\cdot((4\cdot2)+3)$ 回の XOR 演算が必要となる。

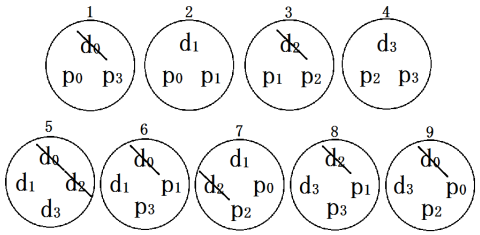

図 4. 提案手法 3 の構成準備

5. まとめ

消失訂正符号の 1 つであるチェーン符号を用いて条件付き (k,n) 閾値秘密分散の構成法を 3 つ示した。第 3 節に安全性要件を示した。要件 1 は式 (3) 及び式 (8) に示したようにバーナム暗号と同じ構成であり、乱数が安全であれば満たすことができる。要件 2 はチェーン符号の復元不可能になる組み合わせから満たされている。従ってチェーン符号の誤り訂正限界で保証している。要件 3 は要件 1 と図 3 から満たしていることが確認できる。要件 4 は要件 1～ 要件 3 及び図 4 から満たしていることが確認できる。本研究では具体的構成を述べたのみであり、条件付き (k,n) 閾値秘密分散の提案手法 1～3 の一般化した構成法の導出が今後の課題である。

参考文献

[1] Kevin M. Greenan, Xiaozhou Li and Jay J. Wylie, “Flat XOR-based erasure codes in storage systems: Constructions, efficient recovery, and tradeoffs,” MSST ‘10 Proceedings of the 2010 IEEE 26th Symposium on Mass Storage Systems and Technologies, May 2010, pp. 1-14.

[2] A.Wilner, “Multiple drive failure tolerant RAID system,” United States Trademark and Patent Office, December 2001, patent number 6,327,627B1.

[3] J.L.Hafner, “WEAVER Codes: Highly fault tolerant erasure codes for storage systems,” in FAST-2005. USENIX Association, December 2005, pp. 212-224.

[4] 栗原正純, 桑門秀典, “分散ストレージにおける再生成符号と秘密分散について,” 電子情報通信学会技術研究報告.IT, 情報理論 110(363), 13-18, January 2011.

[5] E.D.Karnin, J.W.Greene and M.Hellman, “On secret sharing systems,” IEEE Trans. on Information Theory, vol.29, no.1, pp.231-241, January 1983.