双線形写像を用いた墨塗り署名方式の安全性について

伊豆哲也[†,††] 國廣昇[††]
太田和夫[††] 武仲正彦[†]

部分情報がマスキング（墨塗り）された電子文書において，開示部分の完全性と非開示部分の秘匿を両立させる暗号要素技術として墨塗り署名が注目を集めている．2005 年 1 月に宮崎・花岡・今井が提案した双線形写像を用いた墨塗り署名方式（本稿では SUMI-6 と呼ぶ）は上の性質に加え，墨塗り箇所数の秘匿と墨塗り者による開示条件のコントロールが可能という特徴を持っている．しかしアルゴリズムや条件の記述が不十分であったり，条件の妥当性の議論に欠けたりするという問題があった．本稿はさらなる安全性解析を目的として SUMI-6 を詳細検討し，条件によっては可能となるいくつかの問題点を指摘する．また SUMI-6 の改良墨塗り署名方式 SUMI-6′ を提案し，その安全性を議論する．さらに，宮崎らが別に提案した墨塗り署名方式 SUMI-5 に対して，双線形写像を用いて改良を施した墨塗り署名方式 SUMI-5′ の提案・検討を行う．

On the Security of the Sanitizable Signature Schemes from Bilinear Maps

Tetsuya Izu,[†,††] Noboru Kunihiro,[††] Kazuo Ohta[††]
and Masahiko Takenaka[†]

Sanitizable signature schemes attract much attention since they assure both of the integrity of disclosed information and the secrecy of closed information in a masked (or sanitized) electronic document. On January 2005, Miyazaki, Hanaoka, and Imai proposed a sanitizable signature scheme from bilinear maps (called “SUMI-6” in this paper), in addition to the above properties, which hides the number of sanitized portions and whose disclosure conditions are controlled by sanitizers. An aim of this paper is to analyze the security of SUMI-6. This paper points out some possible security problems of SUMI-6. Then, an enhaced scheme SUMI-6′ is proposed with security discussions. Also, an enhanced scheme SUMI-5′ from another sanitizable signature scheme SUMI-5 by Miyazaki et al. will be proposed and analyzed.

1. はじめに

1999 年 5 月に施行された情報公開法[5] は行政機関に対する文書公開を義務づけており，個人情報のように開示すべきでない情報を除いた部分開示を求めている．紙文書における部分開示はマスキング（墨塗り）によって実現されているのに対し，電子文書における墨塗り技術はまだ確立されていない．これに加え 2005 年 4 月に施行された e-文書法[6] は，領収書などの紙文書をスキャナで読み込んで作成した電子文書に対し（電子署名やタイムスタンプなどの一定の要件のもと）法的な証拠能力を認めており，非開示部分の秘匿と開示部分の完全性保証を可能とした墨塗り署名方式の確立が急務となっている．しかし従来の電子署名方式は，文書に対する正当な変更（墨塗り）と不正な変更（改竄）を区別できないため，墨塗りを文書改竄と判断してしまい，部分情報の完全性を保持したままで墨塗りを実現することは困難であった．墨塗り署名方式（Sanitizable Signature Scheme）は電子文書の開示部分の完全性と非開示部分の秘匿を両立させる技術であり，これまでに具体的な方式として CES[10]，SUMI-4[9]，PIATS[4]，SUMI-5[8]，SUMI-6[7]，SSCH[1] などが提案されている．

2005 年 1 月に宮崎・花岡・今井が提案した墨塗り署名方式[7]（本稿では SUMI-6 と呼ぶ）は双線形写像による一括署名[2] を用いることで，上の性質に加えて，墨塗り箇所数の秘匿と墨塗り者による開示条件コントロールを特徴としている．また安全性の面では Privacy，Unforgeability，Invisibility の 3 つの安全性を達成していることが主張されている．

† 株式会社富士通研究所
FUJITSU LABORATORIES LTD.
†† 電気通信大学
The University of Electro-Communications

しかしアルゴリズムや条件の記述が不十分であったり，条件の妥当性の議論に欠けたりしているという問題があり，SUMI-6 の安全性に問題が生じる場面もありうることが分かってきた．そこで本稿は SUMI-6 の安全性解析を目的とし，いくつかの問題点となりうる性質を指摘する．次にこれらの問題を回避するために，改良墨塗り署名方式 SUMI-6′ を提案し，その安全性について議論する．また，宮崎らが別に提案した墨塗り署名方式 SUMI-5[8] の一括署名を用いた改良方式 SUMI-5′ を提案し，その安全性についても議論する．SUMI-6′ と SUMI-5′ には Invisibility の有無という大きな差異が見られるため，適用場面に応じた利用が可能であると考える．

本稿の構成は以下のとおりである：2 章で墨塗り署名方式 SUMI-6[7] のアルゴリズム概要を紹介した後，3 章でいくつかの攻撃法を考察し，墨塗り文書の具体的偽造方法を示す．4 章では SUMI-6 の改良方式である SUMI-6′ を提案する．また 5 章では，宮崎らが提案した墨塗り署名方式 SUMI-5 の改良方式 SUMI-5′ を提案するとともに，その安全性を議論する．

2. SUMI-6

SUMI-6 は宮崎，花岡，今井によって提案された墨塗り署名方式であり[7]，開示・非開示のステータスを表す添字集合を用いることで複数の墨塗り者による墨塗りを可能にしている．本章では SUMI-6 の処理概要と，提案者らによる安全性に関する主張をまとめる．

2.1 記　　号

署名の対象となる文書は n 個の（必ずしも同じでないビット長の）部分文書 M_i（$i = 1, \ldots, n$）から構成される集合 $\{M_1, \ldots, M_n\}$ として与えられているとする．部分文書として，たとえば XML 構文における最小構文単位[3] や JPEG 画像ファイルにおける最小符号化ユニット[11] が考えられる．3 つの群 $\mathbb{G}_1$，$\mathbb{G}_2$，$\mathbb{G}_T$ は位数 p（素数）の乗法的巡回群で，これらの群での離散対数問題は困難であると仮定する．また g_1，g_2 を $\mathbb{G}_1$，$\mathbb{G}_2$ の生成元，写像 $H\colon \{0,1\}^* \to \mathbb{G}_1$ を（フルドメイン）ハッシュ関数とする．さらに写像 $e\colon \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T$ は以下の性質を満たすとする（双線形写像）：

- 双線形性：任意の $u \in \mathbb{G}_1$，$v \in \mathbb{G}_2$，$a, b \in \mathbb{Z}$ に対し $e(u^a, v^b) = e(u, v)^{ab}$
- 非退化性：$e(g_1, g_2) \neq 1_T$（1_T は $\mathbb{G}_T$ の単位元）
- 計算可能性：任意の $u \in \mathbb{G}_1$，$v \in \mathbb{G}_2$ に対し $e(u, v)$ は効率的に計算可能

このような双線形写像の具体例として，楕円曲線上のペアリングを用いた構成法が知られている．

表 1 SUMI-6 の処理概要
Table 1 A description of SUMI-6.

◇署名者の鍵生成
入力 群 $\mathbb{G}_2$ の生成元 g_2 と位数 p
1. $\mathsf{sk} \leftarrow \mathbb{Z}/p\mathbb{Z}$ をランダムに生成し，$\mathsf{pk} \leftarrow g_2^{\mathsf{sk}}$ とする．
出力 署名者の秘密鍵・公開鍵ペア $(\mathsf{sk}, \mathsf{pk}) \in (\mathbb{Z}/p\mathbb{Z}, \mathbb{G}_2)$.

◇署名者
入力 文書 $\{M_1, \ldots, M_n\}$
1. 文書識別子 ID をランダムに生成し，$\bar{M}_0 \leftarrow ID$，$\bar{M}_i \leftarrow ID||M_i$（$i = 1, \ldots, n$）とする．
2. 各部分文書 $\bar{M}_i$（$i = 0, \ldots, n$）に対し，署名 $\sigma_i \leftarrow H(h_i)^{\mathsf{sk}}$ を求める．
3. 一括署名 $\sigma \leftarrow \prod_{i=0}^{n} \sigma_i$ を求める．
4. 添字集合 $DASA \subseteq \{1, \ldots, n\}$ を定める．ここで部分文書 $\bar{M}_i$ は，$i \in DASA$ ならば追加墨塗り可能，$i \notin DASA$ ならば追加墨塗り不可（強制開示）であることを意味する．
出力 文書 $\{\bar{M}_0, \ldots, \bar{M}_n\}$，署名 $\{\sigma_1, \ldots, \sigma_n\}$，一括署名 σ，添字集合 $DASA$☆

◇墨塗り者
入力 文書 $\{\bar{M}_0, \ldots, \bar{M}_k\}$，署名 $\{\sigma_i\}_{i \in DASA}$，一括署名 σ，添字集合 $DASA$☆☆
1. $i \in DASA$ である部分文書 $\bar{M}_i$ の新しい開示条件を決定し，それに応じた処理を行う．
墨塗り 一括署名および添字集合を $\sigma \leftarrow \sigma/\sigma_i$，$DASA \leftarrow DASA \backslash \{i\}$ により更新し，部分文書 $\bar{M}_i$，署名 σ_i を削除する．
強制開示 添字集合を $DASA \leftarrow DASA \backslash \{i\}$ により更新し，署名 σ_i を削除する．
それ以外 何もしない
2. 添字を詰めてリナンバリングする．
出力 文書 $(\bar{M}_0, \ldots, \bar{M}_{k'})$，署名 $\{\sigma_i\}_{i \in DASA}$，一括署名 σ，添字集合 $DASA$☆☆☆

◇検証者
入力 文書 $(\bar{M}_0, \ldots, \bar{M}_k)$，署名 $\{\sigma_i\}_{i \in DASA}$，一括署名 σ，添字集合 $DASA$，署名者の公開鍵 pk☆☆
1.（文書 ID の検証） 各部分文書 $\bar{M}_i$（$i = 1, \ldots, k$）の文書 ID が $\bar{M}_0$ に等しいかを検証する．失敗した場合には invalid を出力して終了する．
2.（一括署名の検証） $e(\sigma, g_2) = \prod_{i=0}^{k} e(H(\bar{M}_i), \mathsf{pk})$ が成立するかを検証する．失敗した場合には invalid を出力して終了する．
3.（個別署名の検証） $i \in DASA$ である部分文書 $\bar{M}_i$ に対し，$e(\sigma_i, g_2) = e(H(\bar{M}_i), \mathsf{pk})$ が成立するかを検証する．失敗した場合には invalid を出力して終了する．
4. valid を出力して終了する．

2.2 SUMI-6 の処理概要

SUMI-6 の処理概要を**表 1** に示す．SUMI-6 では，

☆ ある添字 i（$0 \leq i \leq n$）に対応する $\bar{M}_i$，ある添字 i（$1 \leq i \leq n$）に対応する σ_i は求められるとする．
☆☆ ある添字 i（$0 \leq i \leq k$）に対応する $\bar{M}_i$，ある添字 $i \in DASA$ に対応する σ_i は求められるとする．
☆☆☆ ある添字 i（$0 \leq i \leq k'$）に対応する $\bar{M}_i$，ある添字 $i \in DASA$ に対応する σ_i は求められるとする．

各部分文書は追加墨塗り可能（DASA，Disclosed and Additional Sanitizing is Allowed），追加墨塗り不可（または強制開示，DASP，Disclosed and Additional Sanitizing is Prohibited）のいずれかのステータスをとる．署名者は各部分文書の署名（個別署名）とともに一括署名を生成し，各部分文書のステータスを追加墨塗り可能または追加墨塗り不可のいずれかに設定したうえで，個別署名と一括署名を出力する．墨塗り者はステータスが追加墨塗り可能の部分文書のステータスを墨塗りまたは追加墨塗り不可に変更することが許されている．墨塗りする場合には部分文書と個別署名を削除し，一括署名からもその情報を削る．追加墨塗り不可にする場合には個別署名だけを削除する．そして最後に添字を詰めてリナンバリングした墨塗り文書と一括署名・個別署名を出力する．このように SUMI-6 では，ステータスが追加墨塗り可能または追加墨塗り不可の部分文書の内容と，それらに対応する個別署名・一括署名（正確には，ステータスが追加墨塗り可能な部分文書に対応する個別署名，およびステータスが追加墨塗り不可または追加墨塗り可能な部分文書に対応する一括署名）が保持される．墨塗りや追加墨塗り不可になった部分文書の情報は（部分的に）失われるため，後の墨塗り者がステータスをさらに変更することはできないように設計されている．検証者は墨塗り文書 ID と，個別署名および一括署名をそれぞれ検証する．

2.3 SUMI-6 の安全性

SUMI-6 の提案論文[7] では，SUMI-6 の達成する安全性として Privacy，Unforgeability，Invisibility の 3 つが主張されている．それぞれの安全性の概要は以下のとおりである．安全性の定義については提案論文を参照されたい．

Privacy 墨塗り（削除）された部分文書から，その部分文書のいかなる内容も復元できないこと．

Unforgeability 検証可能な墨塗り文書が偽造できないこと．ただしこの偽造墨塗り文書を構成する部分文書は，今までに署名されたことがない文書の部分文書であるか，今までに署名されたことがある文書の部分文書のうちステータスが強制開示であるものでなければならない．

Invisibility 墨塗り箇所の個数が判定できないこと．さらには墨塗りが施されているかも判定できないことが必要である．

3. SUMI-6 の安全性解析

SUMI-6 の提案論文では，ある条件下における安全性が主張されている．もちろんこの安全性は理論的に正しいものの，条件の記述が不十分であったり，条件の妥当性に議論の余地が残ったりしているものも見受けられる．SUMI-6 の厳密な安全性を示すには，これら条件を詳細に検討する必要があると考える．

一方で SUMI-6 における墨塗り者は匿名で墨塗りを行うため，検証者は誰がどの部分文書のステータスを墨塗り・追加墨塗り可能・追加墨塗り不可と設定したかを検証することができない．したがって，たとえば検証者が権限を逸脱して部分文書のステータスを変更したとしても，他の検証者はその事実に気づかない可能性が生じてしまう．以下では，本来は部分文書のステータス変更を行うべきでないエンティティが権限を逸脱してステータス変更を行う攻撃，あるいは決められた手続き以外の処理を行う攻撃者を想定して考える．

3.1 部分文書のステータス変更

SUMI-6 は，同一文書内の 2 つの部分文書の内容が同じ場合，対応する署名も同一になるという性質を持っている．したがって $\bar{M}_i = \bar{M}_j$ ならば $\sigma_i = \sigma_j$ である．それぞれのステータスが追加墨塗り可能と追加墨塗り不可ならば個別署名 σ_j は削除されているはずであるが，攻撃者は署名 σ_i をもとに $\sigma \leftarrow \sigma \times \sigma_i$，$DASA \leftarrow DASA \cup \{j\}$ とすることで，部分文書 $\bar{M}_j$ のステータスを追加墨塗り不可から追加墨塗り可能に変更することができる．

3.2 部分文書の順序交換

SUMI-6 では，オリジナル文書を構成する部分文書の集合として墨塗り文書が定義されているため，墨塗り文書の部分文書の順序を交換して得られる文書もまた墨塗り文書となる．このように部分文書の順序を考慮しない文書が必要となる場合もあろうが，反面，部分文書の順序を考慮した文書を必要とする場面が存在することも事実である．実際，SUMI-5[8] や PIATS[4] などほとんどの墨塗り署名方式では，部分文書の順序が考慮されている．

SUMI-6 において部分文書の順序を考慮した場合，部分文書・個別署名を受け取った攻撃者は，それら順序を任意に交換することで偽造文書を作成可能である．それぞれの個別署名は正当であり，一括署名は不変であるため偽造文書に対しても正当となるからである．

3.3 部分文書の繰返し

SUMI-6 の墨塗り文書において，もとの文書に含まれる部分文書 $\{\bar{M}_1, \ldots, \bar{M}_n\}$ を繰り返し使用できるかどうかは明示されていない．しかし繰返しが許容される場合には，以下のような攻撃が可能となる．

墨塗り文書 $\{\bar{M}_1, \ldots, \bar{M}_k\}$，個別署名 $\{\sigma_i\}_{i \in DASA}$，

一括署名 σ を受け取った攻撃者は，ステータスが追加墨塗り可能である部分文書 $\bar{M}_i$ と対応する個別署名 σ_i を追加することで，新しい偽造墨塗り文書 $\{\bar{M}_1, \ldots, \bar{M}_k, \bar{M}_i\}$ および一括署名 $\sigma \leftarrow \sigma \times \sigma_i$（ただし添字集合を変更する必要がある）を生成できる．

個々の部分文書に対応する署名が同一の秘密鍵 sk から生成されていることに着目すると，さらに $M' \leftarrow \{\bar{M}_0, \ldots, \bar{M}_k, \bar{M}_0, \ldots, \bar{M}_k\}$，$\sigma' \leftarrow \sigma^2$ とすることで，偽造墨塗り文書 M' 上の正当な一括署名 σ' を生成できる．このときそれぞれの個別署名も正当となっている．ただし文書 ID に対応する部分文書 $\bar{M}_0 = ID$ と文字列をまったく含まない部分文書が識別可能であれば，この攻撃は検知可能である．

本節で述べた部分文書の繰返し攻撃は，墨塗りに応じて一括署名が動的に変化するという SUMI-6 の特徴を利用しているため，他の墨塗り署名には適用できない攻撃である．

3.4 個別署名の復元

SUMI-6 では，文書 ID を意味する部分文書 $\bar{M}_0$ のステータスは追加墨塗り不可となっている．しかしこの条件に対応する処理を誰が行うかが不明であるため，場合によっては問題が生じる場合がある．

署名者が $\bar{M}_0$ のステータスを変更する場合，署名者の出力の時点では個別署名 σ_0 は削除されているが，他の部分文書のステータスがすべて追加墨塗り可能であれば，一括署名 σ と個別署名の積 $\prod_{i=1}^{n} \sigma_i$ から個別署名 σ_0 は復元可能となる．

途中の墨塗り者の出力において，ステータスが追加墨塗り可能な部分文書の個別署名の積 $\sigma_{DASA} = \prod_{i \in DASA} \sigma_i$ を考える．部分文書 $\bar{M}_0$ のステータスはつねに追加墨塗り不可，つまり $0 \notin DASA$ であることに注意すると，σ_0 と σ_{DASA} の積 $\sigma' = \sigma_0 \times \sigma_{DASA}$ は一括署名として正当になっている．このとき対応する文書はもとの文書からステータスが追加墨塗り不可の部分文書を削除したものとなっており，追加墨塗り不可の部分文書が墨塗り（削除）されたことを意味する．

3.5 ステータス変更

SUMI-6 では，署名者または墨塗り者から墨塗り文書・個別署名・一括署名を受け取った任意のエンティティは，墨塗り者として開示条件の変更，つまりステータスが追加墨塗り可能な部分文書を自由に追加墨塗り不可または墨塗り（削除）に変更することが可能である．追加墨塗り可能というステータスは，最終的に開示するか墨塗りするかを墨塗り者の判断に委ねていると考えられることから，墨塗り者が上のようなステータス変更を行えることは望ましいといえる．しかし，たとえば検証者であっても同様なステータス変更が行えるという点は，つねに望ましいとはいえない．特に検証者にとっては，ステータスが追加墨塗り可能であろうと追加墨塗り不可であろうと部分文書が開示されていることには変わりない．SUMI-6 のこのような性質がすぐに攻撃に結びつくわけではないが，他の墨塗り署名には見られない特徴的な性質であるため，明示的に指摘しておく．

4. 改良墨塗り署名方式 SUMI-6′

本章では改良墨塗り署名方式 SUMI-6′ を提案し，安全性に関する検討を行う．

4.1 設計方針

SUMI-6′ の設計方針は，SUMI-6 において手順が不明瞭であった部分を明確化し，前章で考察した攻撃法に対する防御を実現することである．特に SUMI-6′ では，部分文書の初期の開示条件の設定者が署名者から最初の墨塗り者に変更されている．これは署名者はどの部分文書が将来において墨塗りされる/されないかを決定すべきでないという墨塗り署名方式における三者モデル[7),8)] に沿わせるための変更である．この場合，署名者が最初の墨塗り者を兼ねることで SUMI-6 と同等となる．

4.2 SUMI-6′ の処理概要

SUMI-6′ の処理概要を**表 2** に示す．ただし署名者の鍵生成処理は SUMI-6 と同じであるため，記述を省略した．

SUMI-6′ と SUMI-6 の大きな違いは，文書を部分文書の順序列と考える点，各部分文書に対して部分文書 ID を導入した点，および最後の墨塗り者が個別署名の出力を行わない点である．部分文書 ID としてオリジナル文書の文書番号を用いてしまうと，墨塗り文書の添字集合から墨塗りの有無がただちに判定可能になるため，ここではランダムな値を昇順になるように付与することとした．また，SUMI-6 の検証では個別署名は二重に検証されていたため，最後の墨塗り者が個別署名を出力しないように変更してもアルゴリズム上の問題は生じない．

なお上記のように部分文書 ID を付与した場合，SUMI-6 の特徴である Invisibility は情報論的な意味では失われてしまっている．たとえば 2 つの部分文書 ID の差が D ならば，オリジナル文書でのこれら 2 つの部分文書の間にはたかだか D 個の部分文書しか存在しなかったことが類推できるからである．現実的には，そのような類推を（計算量的に）無視できるように部

表 2 SUMI-6′ の処理概要
Table 2 A description of SUMI-6′.

◇署名者
入力 文書列 $(M_1, \ldots, M_n)$
1. 文書識別子 ID と部分文書識別子 ID_i をランダムに生成し，$\bar{M}_0 \leftarrow ID$，$\bar{M}_i \leftarrow ID||ID_i||M_i$ $(i = 1, \ldots, n)$ とする．ただし ID_i は昇順となるように設定する．
2. 各部分文書 $\bar{M}_i$ $(i = 0, \ldots, n)$ に対し，署名 $\sigma_i \leftarrow H(h_i)^{\mathsf{sk}}$ を求める．
3. 一括署名 $\sigma \leftarrow \prod_{i=0}^{n} \sigma_i$ を求める．
出力 文書列 $(\bar{M}_0, \ldots, \bar{M}_n)$，署名列 $(\sigma_1, \ldots, \sigma_n)$，一括署名 σ

◇最初の墨塗り者
入力 文書列 $(\bar{M}_0, \ldots, \bar{M}_n)$，署名列 $(\sigma_1, \ldots, \sigma_n)$，一括署名 σ
1. 各部分文書 $\bar{M}_i$ の開示条件を定め，それに応じた処理を行う．ただし部分文書 $\bar{M}_0$ はつねに強制開示とする．
墨塗り 一括署名を $\sigma \leftarrow \sigma/\sigma_i$ により更新し，部分文書 $\bar{M}_i$，署名 σ_i を削除する．
強制開示 署名 σ_i を削除する．
それ以外 添字集合を $DASA \leftarrow DASA \cup \{i\}$ により更新する．
2. 添字を詰めてリナンバリングする．
出力 文書列 $(\bar{M}_0, \ldots, \bar{M}_{n'})$，署名列 $\{\sigma_i\}_{i \in DASA}$，一括署名 σ，添字集合 $DASA$

◇墨塗り者
入力 文書列 $(\bar{M}_0, \ldots, \bar{M}_k)$，署名列 $\{\sigma_i\}_{i \in DASA}$，一括署名 σ，添字集合 $DASA$
1. $i \in DASA$ である部分文書 $\bar{M}_i$ の新しい開示条件を決定し，それに応じた処理を行う．
墨塗り 一括署名および添字集合を $\sigma \leftarrow \sigma/\sigma_i$，$DASA \leftarrow DASA \backslash \{i\}$ により更新し，部分文書 $\bar{M}_i$，署名 σ_i を削除する．
強制開示 署名 σ_i を削除し，添字集合を $DASA \leftarrow DASA \backslash \{i\}$ により更新する．
それ以外 何もしない
2. 添字を詰めてリナンバリングする．
出力 文書列 $(\bar{M}_0, \ldots, \bar{M}_{k'})$，署名列 $\{\sigma_i\}_{i \in DASA}$，一括署名 σ，添字集合 $DASA$
出力（最終墨塗り者） 文書列 $(\bar{M}_0, \ldots, \bar{M}_{k'})$，一括署名 σ

◇検証者
入力 文書列 $(\bar{M}_0, \ldots, \bar{M}_k)$，一括署名 σ，署名者の公開鍵 pk
1.（文書 ID の検証） 各部分文書 $\bar{M}_i$ $(i = 1, \ldots, k)$ の文書 ID が $\bar{M}_0$ に等しいかを検証する．失敗した場合には invalid を出力して終了する．
2.（部分文書 ID の検証） 各部分文書 $\bar{M}_i$ $(i = 1, \ldots, k)$ の部分文書 ID が昇順になっているかを検証する．失敗した場合には invalid を出力して終了する．
3.（一括署名の検証） $e(\sigma, g_2) = \prod_{i=0}^{k} e(H(\bar{M}_i), \mathsf{pk})$ が成立するかを検証する．失敗した場合には invalid を出力して終了する．
4. valid を出力して終了する．

分文書 ID 空間を大きく設定することは可能である．

4.3 安全性評価

SUMI-6′ の安全性について考察する．SUMI-6′ では各部分文書は部分文書 ID を持っているため，3.1 節で指摘したような署名の一致が起きる確率は無視できるほど小さくすることができる．また部分文書 ID が昇順に制限することで，3.2 節のような順序交換も防いでいる．さらには SUMI-6′ での元の文書の部分文書列を墨塗り文書と見なしていることから，3.3 節で指摘したような部分文書の繰返しもできないようになっている．さらに $\bar{M}_0$ に対する署名は最初の墨塗り者が生成するため，最初の署名者を信用する限りにおいては，3.4 節のような攻撃も防がれている．また最後の墨塗り者が個別署名を出力しないことから，3.5 節で指摘したような，検証者によるステータス変更も不可能になっている．このように SUMI-6′ は，前章で指摘したような処理の曖昧さが排除され，また安全性も向上していることが分かる．

なお SUMI-6′ でも SUMI-6 と同様に，本来は墨塗り者の権限を持たないエンティティが権限を逸脱して追加墨塗り可能な部分文書のステータスを変更することが可能となっている．しかしこれは SUMI-6, SUMI-6′ が持つ，匿名の墨塗り者による墨塗りという性質から導かれる性質であり，墨塗り者が匿名であるべきか否かは墨塗り署名方式の設計ポリシの問題であることから，安全性における問題とは考えていない．

5. SUMI-5

本章では，宮崎らが 2004 年 1 月に提案した墨塗り署名方式 SUMI-5[8] に対し，双線形写像を用いた改良方式 SUMI-5′ を提案するとともに，その効率・安全性について議論する．本章で使用する写像 $H\colon \{0,1\}^* \to \{0,1\}^m$ は，SHA-256 のような通常の（フルドメイン）ハッシュ関数とする．

5.1 SUMI-5 の処理概要

SUMI-5 は SUMI-6 と同様な 3 種類のステータスを用いることで，複数の墨塗り者による墨塗りを可能にしつつ不正な追加墨塗り攻撃を防御している．SUMI-5 の処理概要を**表 3** に示す．

SUMI-5 では，各部分文書は墨塗り（強制非開示，Closed），追加墨塗り可能（DASA，Disclosed and Additional Sanitization is Allowed），追加墨塗り不可（DASP，Disclosed and Additional Sanitization is Prohibited）の 3 つのステータスのいずれかをとる．ステータスの初期値は追加墨塗り可能となっており，各墨塗り者は，ステータスが追加墨塗り可能の部分文書のステータスを墨塗りまたは追加墨塗り不可に変更することで，その部分文書の開示または非開示を確定させていく．

表 3 SUMI-5 の処理概要
Table 3 A description of SUMI-5.

◆署名者
入力 文書 $(M_1, \ldots, M_n)$
1. 各部分文書 M_i $(i = 1, \ldots, n)$ に対する識別子 r_i と乱数 s_i をランダムに生成し，$\bar{M}_i \leftarrow r_i || M_i$ とする．
2. 各部分文書 $\bar{M}_i$ $(i = 1, \ldots, n)$ に対し，平面上の 4 点 $(0, Q_i)$，$(1, H(\bar{M}_i))$，$(2, H(s_i))$，$(3, P_i)$ が一直線となるような P_i，Q_i を求める．
3. 2. で求めた座標値の連結に対する署名 $\sigma \leftarrow \mathrm{Sign}_{\mathrm{sk}}(Q_1 || \cdots || Q_n || P_1 || \ldots || P_n)$ を生成する（sk は署名者の秘密鍵）．
出力 文書 $(\bar{M}_1, \ldots, \bar{M}_n)$，署名 σ，値集合 $(P_1, \ldots, P_n)$，乱数値集合 $(s_1, \ldots, s_n)$

◆墨塗り者
入力 文書 $(\bar{M}_1, \ldots, \bar{M}_n)$，署名 σ，値集合 $(P_1, \ldots, P_n)$，乱数値集合 $(s_1, \ldots, s_n)$，添字集合 D，C，N☆
1. 開示・非開示が未決定な部分文書の条件を定め，添字集合 $D' \supseteq D$，$C' \supseteq C$，$N' \subseteq N$ を生成する．
2. 各部分文書 $\bar{M}_i$ および乱数情報 s_i $(i = 1, \ldots, n)$ から新しい部分文書および乱数情報

$$(\bar{M}'_i, s'_i) = \begin{cases} (\bar{M}_i, H(s_i)) & \text{if } i \in D' \backslash D \\ (H(\bar{M}_i), s_i) & \text{if } i \in C' \backslash C \\ (\bar{M}_i, s_i) & \text{otherwise} \end{cases}$$

を生成する☆☆．
出力 文書 $(\bar{M}'_1, \ldots, \bar{M}'_n)$，署名 σ，値集合 $(P_1, \ldots, P_n)$，乱数値集合 $(s'_1, \ldots, s'_n)$，添字集合 D'，C'，N'

◆検証者
入力 文書 $(\bar{M}_1, \ldots, \bar{M}_n)$，署名 σ，値集合 $(P_1, \ldots, P_n)$，乱数値集合 $(s_1, \ldots, s_n)$，添字集合 D，C，N，署名者の公開鍵 pk
1. 各部分文書 $\bar{M}_i$，乱数情報 s_i $(i = 1, \ldots, n)$ から新しい部分文書および乱数情報

$$(\bar{M}'_i, s'_i) = \begin{cases} (H(\bar{M}_i), s_i) & \text{if } i \in D \\ (\bar{M}_i, H(s_i)) & \text{if } i \in C \\ (H(\bar{M}_i), H(s_i)) & \text{otherwise} \end{cases}$$

を生成する．
2. 各 i $(i = 1, \ldots, n)$ に対し，4 点 $(0, Q'_i)$，$(1, \bar{M}'_i)$，$(2, s'_i)$，$(3, P_i)$ が一直線となるような Q'_i を求める．ただし 3 点 $(1, \bar{M}'_i)$，$(2, s'_i)$，$(3, P_i)$ が一直線上にない場合には，invalid を出力して終了する．
3. 値の連結 $Q'_1 || \cdots || Q'_n || P_1 || \ldots || P_n$ に対し，署名者の公開鍵 pk を用いて検証する．成功した場合には valid を，失敗した場合には invalid を出力して終了する．

5.2 SUMI-5′ の設計方針と処理概要

SUMI-5′ の設計方針は，SUMI-5[8] に対して（SUMI-6 とは別の方向で）双線形写像による一括署名の技法を適用することである．しかし部分文書やそれに関連する情報の削除は行わないため，部分文書の個数は一定であり，墨塗り箇所の秘匿は不可能である（つまり Invisibility は成立しない）．

SUMI-5′ の処理概要を**表 4** に示す．SUMI-5 が用いていた値情報の代わりに，SUMI-5′ は更新情報と呼ばれる 2 種類の個別署名の比を保持する．そしてステータスに応じて部分文書または墨を削除して一括署名を更新する．署名検証時には一括署名を検証すればよく，SUMI-5 のような値情報の復元が不要となっている．

表 4 SUMI-5′ の処理概要
Table 4 A description of SUMI-5′.

◆署名者の鍵生成
入力 群 $\mathbb{G}_2$ の生成元 g_2 と位数 p
1. $\mathsf{sk} \leftarrow \mathbb{Z}/p\mathbb{Z}$ をランダムに生成し，$\mathsf{pk} \leftarrow g_2^{\mathsf{sk}}$ とする．
出力 署名者の秘密鍵・公開鍵ペア $(\mathsf{sk}, \mathsf{pk}) \in (\mathbb{Z}/p\mathbb{Z}, \mathbb{G}_2)$．

◆署名者
入力 文書 $(M_1, \ldots, M_n)$
1. 各部分文書 M_i $(i = 1, \ldots, n)$ に対する部分文書識別子 ID_i をランダムに生成し，$\bar{M}_i \leftarrow ID_i || M_i$ とする．またハッシュ値 $h_i \leftarrow H(\bar{M}_i)$ と署名 $\sigma_i \leftarrow h_i^{\mathsf{sk}}$ を求める．
2. 各 i $(i = 1, \ldots, n)$ に対する乱数 R_i を生成し，ハッシュ値 $h'_i \leftarrow H(R_i)$ と署名 $\sigma'_i \leftarrow h'^{\mathsf{sk}}_i / \sigma_i$ を求める．
3. 一括署名 $\sigma \leftarrow \prod_{i=1}^{n} \sigma_i$ を求める．
出力 文書 $(\bar{M}_1, \ldots, \bar{M}_n)$，墨 $(R_1, \ldots, R_n)$，一括署名 σ，更新情報 $(\sigma'_1, \ldots, \sigma'_n)$

◆墨塗り者
入力 文書 $(\bar{M}_1, \ldots, \bar{M}_n)$，墨 $(R_1, \ldots, R_n)$，一括署名 σ，更新情報 $(\sigma'_1, \ldots, \sigma'_n)$
1. 各部分文書 $\bar{M}_i$ $(i = 1, \ldots, n)$ の開示条件を定め，それに応じた処理を行う．
強制非開示（墨塗り） 部分文書 $\bar{M}_i$ を適当なデータに置き換え，$\sigma \leftarrow \sigma \times \sigma'_i$ により一括署名を更新する．
開示・非開示が未決定 何もしない
強制開示 墨 R_i を適当なデータに置き換える．
出力 文書 $(\bar{M}_1, \ldots, \bar{M}_n)$，墨 $(R_1, \ldots, R_n)$，一括署名 σ，更新情報 $(\sigma'_1, \ldots, \sigma'_n)$（ただし最後の墨塗り者は更新情報を出力しない）．

◆検証者
入力 文書 $(\bar{M}_1, \ldots, \bar{M}_n)$，墨 $(R_1, \ldots, R_n)$，一括署名 σ，署名者の公開鍵 pk
1. 各部分文書 $\bar{M}_i$ $(i = 1, \ldots, n)$ に対し，$\bar{M}_i$ が置き換えられていれば（非開示と判断して）$h_i \leftarrow H(R_i)$，そうでなければ（開示と判断して）$h_i \leftarrow H(\bar{M}_i)$ とする．
2.（一括署名の検証） $e(\sigma, g_2) = \prod_{i=1}^{n} e(h_i, \mathsf{pk})$ が成立するかを検証する．失敗した場合には invalid を出力して終了する．
3. valid を出力して終了する．

5.3 SUMI-5′ の評価

安全性の面では，SUMI-5′ の安全性は SUMI-5 と同等であると考えられる．

☆ ここで添字集合 D，C，N は開示が確定した部分文書の集合，非開示が確定した部分文書の集合，開示・非開示が未決定な部分文書の集合に対応し，$D \cup C \cup N = \{1, \ldots, n\}$，$D \cap C \cap N = \phi$ を満たすとする．

☆☆ オリジナルの SUMI-5 では $i \in D' \backslash D$ または $i \in C' \backslash C$ のときに乱数情報または部分文書情報を削除またはマスクすることになっているが，本稿では簡単のためハッシュ値による置換を用いた．

表 5 墨塗り署名方式の比較
Table 5 A comparison of some sanitizable signature schemes.

	安全性			効率性	
	Privacy	Unforgeability	Invisibility	署名生成	検証者
SUMI-6	満たす	満たす	満たす	n	たかだか $n+1$
SUMI-6′	満たす	満たす	部分的に満たさない	n	1
SUMI-5	満たす	満たす	満たさない	1	たかだか $3n$
SUMI-5′	満たす	満たす	満たさない	$2n$	たかだか $2n$

次に効率の面では，検証者が必要とする情報量において SUMI-5′ は優れている．SUMI-5 では 3 種類の情報系列が必要で，そのままでは $3n$ 個程度（改善しても $2n$ 個程度）の入力を必要とし，また検証処理の中でもう 1 種類の情報系列を生成する必要がある．これに対して SUMI-5′ では 2 種類の情報系列で十分で，そのままでは $2n$ 個程度（改善すれば n 個程度）の入力で済み，検証処理の中でも新たな系列を必要とはしない．

しかし署名生成処理を比べると，値情報を計算しないでよい反面，SUMI-5′ は $2n$ 回の（双線形写像を用いた）署名生成が必要となり，1 回の（通常のアルゴリズムでの）署名生成しか必要としない SUMI-5 に比べて効率が大きく劣っている．

5.4 比　　較

最後に本稿でこれまでに議論した墨塗り署名方式である SUMI-6，SUMI-6′，SUMI-5，SUMI-5′ を，安全性と効率性の面で比較する．比較結果を**表 5** にまとめる．ここで効率性の比較においては，署名者が必要とする署名生成回数と，検証者が入力として必要とするデータの個数を用いた．

SUMI-6 と SUMI-6′ で検証者が必要とするデータ量の大きな違いは，最終墨塗り者が個別署名を出力するかどうかによっている．SUMI-6/6′ と SUMI-5/5′ の効率性を比較すると，SUMI-6/6′ の方が優れているように見受けられる．

一方で SUMI-6/6′ が（部分的に）持つ Invisibility という性質は，SUMI-5/5′ では失われてしまっている．しかし Invisibility という性質は適用場面によって必要性の有無が変わってくる．たとえば戸籍台帳から戸籍謄本を作成する場面では Invisibility は望ましい性質だが，公文書の公開場面では，墨塗りの有無が大きな意味を持つことから，Invisibility はむしろ不要であると考えられる．

6. ま と め

墨塗り署名方式 SUMI-6 を解析し，いくつかの問題点となりうる特徴を指摘した．また 2 つの改良墨塗り署名方式 SUMI-6′，SUMI-5′ を提案し，安全性を検討した．SUMI-6′/5′ は SUMI-6/5 の安全性や効率性を向上させられているが，SUMI-6′ と SUMI-5′ については，適用場面に応じた使い分けが必要であると考える．

オリジナルの一括署名方式[2] では，それぞれの個別署名を生成する秘密鍵はすべて異なっているのに対し，本稿で議論した SUMI-6，SUMI-6′，SUMI-5′ は同じ秘密鍵を用いている．このため，この差異が安全性に与える影響を解析する必要性が高いと考える．また他の墨塗り署名方式に対し，本稿で議論したような一括署名方式の適用もこれからの課題である．

参 考 文 献

1) Ateniese, G., Chou, D.H., Medeiros, B. and Tsudik, G.: Sanitizable Signatures, *ESORICS 2005*, LNCS 3679, pp.159–177, Springer-Verlag (2005).
2) Boneh, D., Gentry, C., Lynn, B. and Shacham, H.: Aggregate and Verifiably Encrypted Signatures from Bilinear Maps, *Eurocrypt 2003*, LNCS 2656, pp.416–432 (2003).
3) Bull, L., Stanski, P. and Squire, D.M.: Content Extraction Signatures using XML Digital Signatures and Custom Transforms On-Demand, *WWW 2003*, pp.170–177, ACM (2003).
4) Izu, T., Kanaya, N., Takenaka, M. and Yoshioka, T.: PIATS: A Partially Sanitizable Signature Scheme, *ICICS 2005*, LNCS 3783, pp.72–83, Springer-Verlag (2005). (Preliminary version was published at CSS 2004 as “検証者が署名者と墨塗り者を検証可能な電子文書の墨塗り方式”, by Takenaka, M., Yoshioka, T. and Kanaya, N.).
5) 行政機関の保有する情報の公開に関する法律，総務省 (1999). Available at http://law.e-gov.go.jp/htmldata/H11/H11HO042.html
6) 民間事業者等が行う書面の保存等における情報通信の技術の利用に関する法律，内閣府 (2004). Available at http://law.e-gov.go.jp/htmldata/H16/H16HO149.html

7) Miyazaki, K., Hanaoka, G. and Imai, H.: Digitally Signed Document Sanitizing Scheme from Bilinear Maps, *SCIS 2005*, 3E3-5, pp.1471–1476 (2005).
8) Miyazaki, K., Iwamura, M., Matsumoto, T., Sasaki, R., Yoshiura, H., Tezuka, S. and Imai, H.: Digitally Signed Document Sanitizing Scheme with Disclosure Condition Control, *IEICE Trans. Fundamentals*, Vol.E88-A, No.1, pp.239–246 (2005).
9) 宮崎邦彦，須崎誠一，岩村　充，松本　勉，佐々木良一，吉浦　裕：電子文書墨塗り問題，電子情報通信学会技術研究報告書，ISEC 2003-20, pp.61–68 (2003).
10) Steinfeld, R., Bull, L. and Zheng, Y.: Content Extraction Signatures, *ICISC 2001*, LNCS 2288, pp.285–304 (2002).
11) 武仲正彦，吉岡孝司：画像ファイルに対する部分完全性保証技術の実現，電子情報通信学会技術研究報告，ISEC 2005-69, pp.183–188 (2005).

(平成 17 年 11 月 25 日受付)
(平成 18 年 6 月 1 日採録)

伊豆　哲也（正会員）

昭和 42 年生．平成 9 年立教大学大学院理学研究科博士後期課程退学．同年（株）富士通研究所入社．情報セキュリティの研究に従事．電気通信大学電気通信学研究科博士後期課程在学中．平成 13 年 Waterloo 大学（カナダ）客員研究員．平成 11 年暗号と情報セキュリティシンポジウム（SCIS 1999）論文賞受賞．平成 14 年コンピュータセキュリティシンポジウム（CSS 2002）優秀論文賞受賞．平成 17 年電子情報通信学会基礎・境界ソサイエティ功労賞受賞．電子情報通信学会，IACR 各会員．

國廣　昇

昭和 46 年生．平成 8 年東京大学大学院工学系研究科計数工学専攻修士課程修了．同年日本電信電話（株）入社．平成 8 年より平成 14 年まで，NTT コミュニケーション科学基礎研究所に勤務．平成 14 年電気通信大学講師．平成 18 年同大学助教授．現在に至る．博士（工学）．情報セキュリティ，暗号理論，量子計算の研究に従事．著書に，『ほんとうに安全？現代の暗号』（岩波科学ライブラリー）．翻訳書に『暗号理論』（岩波，1 冊でわかるシリーズ）．平成 9 年「SCIS 論文賞」受賞．電子情報通信学会，数式処理学会，IACR 各会員．

太田　和夫（正会員）

昭和 52 年早稲田大学理工学部数学科卒業．昭和 54 年早稲田大学大学院修士課程修了．昭和 54 年から平成 13 年日本電信電話（NTT）研究所に勤務．平成 13 年から現在，電気通信大学教授．専門は情報セキュリティ，特に暗号理論．理学博士．電子情報通信学会業績賞，小林記念特別賞受賞（平成 5 年）．電気通信普及財団（テレコムシステム技術賞）受賞（平成 10 年）．電子情報通信学会，IACR，IEEE 各会員．編著書に『情報セキュリティの科学』（講談社ブルーバックス），『暗号・ゼロ知識証明・数論』（共立出版），『ほんとうに安全？現代の暗号』（岩波科学ライブラリー）等．翻訳書に，『暗号理論』（岩波，一冊でわかるシリーズ），『計算理論の基礎』（共立出版）．

武仲　正彦

昭和 42 年生．平成 2 年大阪大学工学部電気工学科卒業．平成 4 年大阪大学大学院工学研究科電気工学専攻博士前期課程修了．同年（株）富士通研究所入社．主任研究員．公開鍵・共通鍵暗号の攻撃・実装技術，サイドチャネル攻撃，ネットワークセキュリティの研究に従事．平成 14 年コンピュータセキュリティシンポジウム（CSS 2002）優秀論文賞受賞．平成 17 年（財）電気科学技術奨励会第 53 回電気科学技術奨励賞受賞．電子情報通信学会会員．