不完全な署名情報からの偽造耐性を持つ電子署名方式の提案

笹目 大輝[1,a)]　原 啓祐[2,4]　知久 奏斗[1,4]　四方 順司[3,2]

概要：本論文では，不完全な署名情報を用いた偽造に耐性を持つ電子署名である復元不能署名を提案する．復元不能署名は，攻撃者が偽造対象の署名の一部を入手できる状況で，偽造不可能性を保証する．具体的には，復元不能署名のシンタックスとその安全性を定式化し，通常の電子署名方式に基づいた構成を提案する．さらに，復元不能署名に要求される安全性が，通常の偽造不可能性や（関連技術である）圧縮不能署名に求められている安全性より真に強いことを示す．

Proposal of a Digital Signature Scheme Resistant to Forgery from Incomplete Signature Information

Daiki Sasame[1,a)]　Keisuke Hara[2,4]　Sohto Chiku[1,4]　Junji Shikata[3,2]

Abstract: In this paper, we propose a novel digital signature scheme called non-recoverable signatures, which is resistant to forgery even when incomplete signature information is available. The non-recoverable signature guarantees unforgeability in scenarios where an adversary obtains partial information of a target signature. Specifically, We formalize the syntax and security of our proposed scheme and present a construction derived from a standard digital signature scheme. We also demonstrate that the security required for non-recoverable signatures are strictly stronger than ordinary unforgeability and the security of incompressible signatures.

1. はじめに

1.1 背景

電子署名 [7] は，（秘密の）署名鍵を持っているユーザが，文書に対して署名を付与することができる暗号技術であり，現代社会において，デジタルコミュニケーションや電子取引の基盤として不可欠な役割を広く担っている．通常，電子署名には，標準的な安全性として，「（署名者を除く）ユーザが，過去に署名が発行された文書以外に対して，署名を新たに偽造することはできない」という EUF-CMA 安全性を満たすことが求められる．しかしながら，電子署名の実社会における使用用途は多岐に渡るため，EUF-CMA 安全性だけでは捉えきれない電子署名に対する脅威が，これまでも先行研究において考えられてきている [1,2,4–6,8,9]．

本論文では，そのような新たな脅威の一例として，署名者によって正しく生成された電子署名が，検証者に渡る過程でその一部が攻撃者に漏洩し，攻撃者がその漏洩情報に基づいて，署名を復元するという脅威を考える．実社会におけるこのような脅威が考えられるシナリオとして，通信路やデータベースに対するサイドチャネル攻撃により，攻撃者に署名の一部が漏洩するようなケースが考えられる．従来の研究では，秘密鍵や署名時に用いる乱数などに対して攻撃者が部分的な情報を入手できる状況に関する安全性は広く議論されてきた [11] が，攻撃者が偽造したいターゲットの署名の部分情報を入手できる状況に焦点を当てた研究はほとんど存在しない．

上記の署名に対する漏洩を考慮した関連研究として，近年，**圧縮不能署名** [3,10] と呼ばれる技術が提案されている．

1 横浜国立大学大学院環境情報学府
Graduate School of Environment and Information Sciences, Yokohama National University
2 横浜国立大学先端科学高等研究院
Institute of Advanced Sciences, Yokohama National University
3 横浜国立大学大学院環境情報研究院
Faculty of Environment and Information Sciences, Yokohama National University
4 産業技術総合研究所
National Institute of Advanced Industrial Science and Technology (AIST)
a) sasame-daiki-xm@ynu.jp

圧縮不能署名は，署名を大きなサイズにすることで，攻撃者が偽造対象のメッセージに対して，すでに生成された署名を再利用（または，一部の漏洩した署名情報に基づいて再生成）しようとした場合，過去に生成された署名を「ほぼそのまま」保存していなければ偽造が成功しないように設計された電子署名技術である．

攻撃者がターゲット署名の部分情報を入手する状況をモデル化している点において，圧縮不能署名は本論文で考えている脅威を捉えているように思われる．しかしながら，圧縮不能署名に対する安全性では，攻撃者は（偽造対象ではないような）**全ての署名**について入手できる情報が制限されており，実社会における脅威を正確に捉えていると言えない．

1.2 貢献

上記の背景に基づき，本論文では，サイドチャネル攻撃などによるターゲット署名の一部が漏洩する状況を考慮した新しい電子署名技術として**復元不能署名**を提案する．復元不能署名では，攻撃者が，偽造対象ではない文書に対する署名を自由に入手でき，かつ，ターゲット署名の部分情報を入手できる状況であっても，ターゲット署名の偽造不可能性を保証することが可能である．

技術的な貢献として，本論文では，復元不能署名のシンタックスと厳密な安全性定義を与えるとともに，（通常の）電子署名方式に基づいて，復元不能署名を構成できることを示す．また，復元不能署名の安全性の意義を示すために，従来の圧縮不能署名の安全性や EUF-CMA 安全性との関係や差異を分析する．具体的には，以下の事実を証明することで，復元不能署名の安全性が，圧縮不能署名の安全性や EUF-CMA 安全性より真に強い概念であることを示す．

- 復元不能署名の安全性は，圧縮不能署名の安全性と EUF-CMA 安全性を含意する．
- EUF-CMA 安全性，もしくは，圧縮不能署名の安全性を満たすが，復元不能署名の安全性を満たさない方式（セパレーション）が存在する．

2. 準備

表記．本論文では，以下の表記を用いる．$x \leftarrow\!\$\ X$ は要素 x を有限集合 X から一様ランダムにサンプリングすることを示す．$\mathbb{N}$ は自然数の集合を表し，$\mathbb{Z}$ は整数の集合を表す．PPT は確率的多項式時間の略である．関数 $f(\lambda)$ が λ で無視できるとは，任意の $c \in \mathbb{Z}$ に対して $f(\lambda) = \mathrm{o}(1/\lambda^c)$ を満たすことであり，これを λ で無視できる関数 $\mathsf{negl}(\lambda)$ と表す．2 つの確率変数 X と Y について，Y 条件付きの X の平均最小エントロピーを次のように定義する．

$$H_\infty(X|Y) = -\log \mathbf{E}_{y \leftarrow\!\$ Y}[\max_x \Pr[X = x|Y = y]].$$

2.1 電子署名

ここでは，電子署名を定義する．本論文で用いる電子署名方式では，検証アルゴリズムにおいて，メッセージと署名の両方を入力として取る形式ではなく，署名にメッセージを暗黙的に組み込む形式を採用する．メッセージ空間 $\{0,1\}^{L_m}$，署名空間 $\{0,1\}^{L_\sigma}$ の電子署名方式は以下の PPT アルゴリズムから構成される：

$\mathsf{Gen}(1^\lambda) \to \mathsf{vk}, \mathsf{sk}$**：** 鍵生成アルゴリズムはセキュリティパラメータ 1^λ を入力とし， 検証鍵 vk と署名鍵 sk を出力する．

$\mathsf{Sign}(\mathsf{sk}, m) \to \sigma$**：** 署名アルゴリズムは署名鍵 sk とメッセージ m を入力として，署名 σ を出力する．

$\mathsf{Ver}(\mathsf{vk}, \sigma) \to m/\perp$**：** 検証アルゴリズムは検証鍵 vk と署名 σ を入力として，メッセージ m もしくは $\perp$ を出力する．

以下の条件を満たすとき，電子署名方式 $\mathsf{Sig} = (\mathsf{Gen}, \mathsf{Sign}, \mathsf{Ver})$ は正当性を満たす．

全ての $\lambda \in \mathbb{N}, m \in \{1,0\}^{L_m}, (\mathsf{vk}, \mathsf{sk}) \leftarrow \mathsf{Gen}(1^\lambda, 1^S)$ について，

$$\Pr[\mathsf{Ver}(\mathsf{vk}, \mathsf{Sign}(\mathsf{sk}, m)) = m] \geq 1 - \mathsf{negl}(\lambda).$$

定義 1. 電子署名方式 $\mathsf{Sig} = (\mathsf{Gen}, \mathsf{Sign}, \mathsf{Ver})$ が EUF-CMA 安全であるとき，全ての PPT 攻撃者 $\mathcal{A}$ について以下を満たす．

$$\Pr\left[\mathsf{SigForge}^{\text{EUF-CMA}}_{\mathcal{A},\mathsf{Sig}}(1^\lambda) = 1\right] \leq \mathsf{negl}(\lambda).$$

いま，$\mathsf{SigForge}^{\text{EUF-CMA}}_{\mathcal{A},\mathsf{Sig}}(1^\lambda)$ は 図 1において定義される．

2.2 圧縮不能署名

ここでは，圧縮不能署名 [10] を定義する．圧縮不能署名は以下の PPT アルゴリズムから構成される：

$\mathsf{Gen}(1^\lambda, 1^S) \to \mathsf{vk}, \mathsf{sk}$**：** 鍵生成アルゴリズムはセキュリティパラメータ $1^\lambda, 1^S$ を入力とし， 検証鍵 vk と署名鍵 sk を出力する．

$\mathsf{Sign}(\mathsf{sk}, m) \to \sigma$**：** 署名アルゴリズムは署名鍵 sk とメッセージ m を入力として，署名 σ を出力する．

$\mathsf{Ver}(\mathsf{vk}, \sigma) \to m/\perp$**：** 検証アルゴリズムは検証鍵 vk と署名 σ を入力として，メッセージ m もしくは $\perp$ を出力する．

以下の条件を満たすとき，圧縮不能署名 $\mathsf{Sig} = (\mathsf{Gen}, \mathsf{Sign}, \mathsf{Ver})$ は正当性を満たす．

全ての $\lambda \in \mathbb{N}, m \in \{1,0\}^{L_m}, (\mathsf{vk}, \mathsf{sk}) \leftarrow \mathsf{Gen}(1^\lambda, 1^S)$ について，

$$\Pr[\mathsf{Ver}(\mathsf{vk}, \mathsf{Sign}(\mathsf{sk}, m)) = m] \geq 1 - \mathsf{negl}(\lambda).$$

$\mathsf{SigForge}^{\mathrm{EUF\text{-}CMA}}_{\mathcal{A},\mathsf{Sig}}(1^\lambda)$:	$O_{\mathsf{Sign}}(m)$:
1 : $L_{\mathsf{Sign}} \coloneqq \emptyset; i = 0$	1 : **if** $(m, *) \in L_{\mathsf{Sign}}$ **then return** 0
2 : $(\mathsf{sk}, \mathsf{vk}) \leftarrow \mathsf{Gen}(1^\lambda)$	2 : **if** $m = m_{i^*}$ **then return** 0
3 : $\sigma' \leftarrow \mathcal{A}^{O_{\mathsf{Sign}}}(\mathsf{vk}, m_{i^*})$	3 : $i = i + 1$
4 : **if** $\mathsf{Ver}(\mathsf{vk}, \sigma') = m_{i^*}$ **then return** 1	4 : $\sigma_i \leftarrow \mathsf{Sign}(\mathsf{sk}, m)$
5 : **return** 0	5 : $m_i \coloneqq m$
	6 : $L_{\mathsf{Sign}} \coloneqq L_{\mathsf{Sign}} \cup \{(m_i, \sigma_i)\}$
	7 : **return** σ_i

図 1: 電子署名の安全性ゲーム

$\mathsf{SigForge}^{\mathsf{IncomSig}}_{\mathcal{A},\mathsf{Sig}}(1^\lambda)$:	$O_{\mathsf{Sign}}(m)$:
1 : $L_{\mathsf{Sign}} \coloneqq \emptyset; i = 0$	1 : **if** $(m, *) \in L_{\mathsf{Sign}}$ **then return** 0
2 : $S \leftarrow \mathcal{A}_1(1^\lambda)$	2 : $i = i + 1$
3 : $(\mathsf{sk}, \mathsf{vk}) \leftarrow \mathsf{Gen}(1^\lambda, 1^S)$	3 : $\sigma_i \leftarrow \mathsf{Sign}(\mathsf{sk}, m)$
4 : $\mathsf{aux} \leftarrow \mathcal{A}_1(\mathsf{vk})$	4 : $m_i \coloneqq m$
5 : $\mathsf{st} \leftarrow \mathcal{A}_1^{O_{\mathsf{Sign}}}(\mathsf{vk})$ $(\mathsf{st} < S)$	5 : $L_{\mathsf{Sign}} \coloneqq L_{\mathsf{Sign}} \cup \{(m_i, \sigma_i)\}$
6 : $\sigma' \leftarrow \mathcal{A}_2(\mathsf{vk}, \mathsf{st}, \mathsf{aux})$	6 : **return** σ_i
7 : **if** $\mathsf{Ver}(\mathsf{vk}, \sigma') = \perp$ **then return** 0	
8 : **return** 1	

図 2: 圧縮不能署名の安全性ゲーム

定義 2. 圧縮不能署名 $\mathsf{Sig} = (\mathsf{Gen}, \mathsf{Sign}, \mathsf{Ver})$ が圧縮不能安全であるとき，全ての PPT 攻撃者 $\mathcal{A}$ について以下を満たす．

$$\Pr\left[\mathsf{SigForge}^{\mathsf{IncomSig}}_{\mathcal{A},\mathsf{Sig}}(1^\lambda) = 1\right] \leq \mathsf{negl}(\lambda).$$

いま，$\mathsf{SigForge}^{\mathsf{IncomSig}}_{\mathcal{A},\mathsf{Sig}}(1^\lambda)$ は 図 2において定義される．

3. 復元不能署名

本章では，復元不能署名とその安全性を定義する．復元不能署名では，攻撃者が，偽造対象ではない文書に対する署名を自由に入手でき，かつ，ターゲット署名の部分情報を入手できる状況であっても，ターゲット署名の偽造不可能性を保証する．入手できるターゲット署名のサイズはパラメータ S によって定義し，復元不能署名が安全性を満たすとは，ターゲット署名を S ビットまで攻撃者が入手できる場合に，ターゲット署名の偽造に成功する確率が無視できる値以下であることを示す．

3.1 シンタックス

復元不能署名は以下の PPT アルゴリズムから構成される：

$\mathsf{Gen}(1^\lambda, 1^S) \to \mathsf{vk}, \mathsf{sk}$**:** 鍵生成アルゴリズムはセキュリティパラメータ $1^\lambda, 1^S$ を入力とし， 検証鍵 vk と署名鍵 sk を出力する．

$\mathsf{Sign}(\mathsf{sk}, m) \to \sigma$**:** 署名アルゴリズムは署名鍵 sk とメッセージ m を入力として，署名 σ を出力する．

$\mathsf{Ver}(\mathsf{vk}, \sigma) \to m/\perp$**:** 検証アルゴリズムは検証鍵 vk と署名 σ を入力として，メッセージ m もしくは $\perp$ を出力する．

3.2 正当性

以下の条件を満たすとき，復元不能署名 $\Pi = (\mathsf{Gen}, \mathsf{Sign}, \mathsf{Ver})$ は正当性を満たす．

全ての $\lambda \in \mathbb{N}, m \in \{1,0\}^{L_m}, (\mathsf{vk}, \mathsf{sk}) \leftarrow \mathsf{Gen}(1^\lambda, 1^S)$ について，

$$\Pr[\mathsf{Ver}(\mathsf{vk}, \mathsf{Sign}(\mathsf{sk}, m)) = m] \geq 1 - \mathsf{negl}(\lambda).$$

3.3 安全性定義

ここでは，復元不能署名の安全性を定義する．

定義 3. 全ての PPT 攻撃者 $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$ について以下を満たすとき，復元不能署名 $\Pi = (\mathsf{Gen}, \mathsf{Sign}, \mathsf{Ver})$ は復元不能安全であるという．

$$\Pr\left[\mathsf{SigForge}^{\mathsf{NR}}_{\mathcal{A},\Pi}(1^\lambda) = 1\right] \leq \mathsf{negl}(\lambda).$$

いま，$\mathsf{SigForge}^{\mathsf{NR}}_{\mathcal{A},\Pi}(1^\lambda)$ は 図 3において定義される．

$\mathsf{SigForge}^{\mathsf{NR}}_{\mathcal{A},\Pi}(1^\lambda)$:

1 : $L_{\mathsf{Sign}} \coloneqq \emptyset; i = 0$
2 : $S \leftarrow \mathcal{A}_1(1^\lambda)$
3 : $(\mathsf{sk}, \mathsf{vk}) \leftarrow \mathsf{Gen}(1^\lambda, 1^S)$
4 : $\mathsf{aux} \leftarrow \mathcal{A}_1(\mathsf{vk})$
5 : $(i^*, \mathsf{st}) \leftarrow \mathcal{A}_1^{O_{\mathsf{Sign}}}(\mathsf{vk})$ $(\mathsf{st} < S)$
6 : $\sigma' \leftarrow \mathcal{A}_2^{O_{\mathsf{Sign}}, O_{\mathsf{Read}}}(\mathsf{vk}, m_{i^*}, \mathsf{st}, \mathsf{aux})$
7 : **if** $\mathsf{Ver}(\mathsf{vk}, \sigma') = m_{i^*}$ **then return** 1
8 : **return** 0

$O_{\mathsf{Sign}}(m)$:

1 : **if** $(m, *) \in L_{\mathsf{Sign}}$ **then return** 0
2 : **if** $m = m_{i^*}$ **then return** 0
3 : $i = i + 1$
4 : $\sigma_i \leftarrow \mathsf{Sign}(\mathsf{sk}, m)$
5 : $m_i \coloneqq m$
6 : $L_{\mathsf{Sign}} \coloneqq L_{\mathsf{Sign}} \cup \{(m_i, \sigma_i)\}$
7 : **return** σ_i

$O_{\mathsf{Read}}(i)$:

1 : **if** $i = i^*$ **then return** 0
2 : **if** $(m_i, \sigma_i) \in L_{\mathsf{Sign}}$ **then return** (m_i, σ_i)

図 3: 復元不能署名の安全性ゲーム

4. 構成

本章では，復元不能署名を電子署名方式に基づいて構成する．

4.1 構成

λ, S をセキュリティパラメータとして設定する．また，$n = S + \mathsf{poly}(\lambda)$ として，$\mathsf{Sig} = (\mathsf{Gen}, \mathsf{Sign}, \mathsf{Ver})$ をメッセージ空間 $\{0,1\}^{n+L_m}$ の電子署名方式とする．復元不能署名 $\Pi = (\mathsf{Gen}, \mathsf{Sign}, \mathsf{Ver})$ の構成を以下で示す．

$\mathsf{Gen}(1^\lambda, 1^S)$: $\mathsf{Sig.Gen}(1^\lambda)$ を実行し，$(\mathsf{Sig.vk}, \mathsf{Sig.sk})$ を入手する．$\mathsf{vk} = \mathsf{Sig.vk}$, $\mathsf{sk} = \mathsf{Sig.sk}$ として出力する．

$\mathsf{Sign}(\mathsf{sk}, m)$: 乱数 $R \in \{0,1\}^n$ をサンプリングし，$\sigma \leftarrow \mathsf{Sig.Sign}(\mathsf{sk}, (R, m))$ を実行し，σ を出力する．

$\mathsf{Ver}(\mathsf{vk}, \sigma)$: $M \leftarrow \mathsf{Sig.Ver}(\mathsf{vk}, \sigma)$ を実行し，M を得る．$M = \perp$ であれば，$\perp$ を出力する．そうでなければ，$M = (R, m)$ として m を出力する．

4.2 安全性証明

定理 1. Sig が EUF-CMA 安全な電子署名方式であるとき，本構成 Π は復元不能安全である．

証明. 本構成 Π における復元不能安全を破る攻撃者 $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$ を用いて，Sig の EUF-CMA 安全を破る攻撃者 $\mathcal{B}$ を構成する．攻撃者 $\mathcal{B}$ は以下の通りに動作する．

(1) $\mathcal{B}$ は 1^λ を $\mathcal{A}_1$ に送る．その後，$\mathcal{B}$ は $\mathcal{A}_1$ から 1^S を受け取り，$n = S + \mathsf{poly}(\lambda)$ とする．

(2) $\mathcal{B}$ は vk を挑戦者から受け取り，それを $\mathcal{A}_1$ に送る．その後，$\mathcal{A}_1$ から aux を受け取る．

(3) $\mathcal{A}_1$ が行った署名クエリ m_i 毎に，$\mathcal{B}$ はランダムな $R_i \in \{0,1\}^n$ をサンプリングし，挑戦者に署名クエリ (R_i, m_i) を送る．挑戦者から受け取った σ_i をそのまま $\mathcal{A}_1$ へと返す．

(4) $\mathcal{A}_1$ が st と i^* を生成し，i^*, st を出力すると，$\mathcal{B}$ は $\mathsf{vk}, i^*, \mathsf{st}, \mathsf{aux}$ を $\mathcal{A}_2$ に送る．

(5) $\mathcal{A}_2$ が行った，m_{i^*} ではない署名クエリ m_i 毎に，$\mathcal{B}$ はランダムな $R_i \in \{0,1\}^n$ をサンプリングし，挑戦者に署名クエリ (R_i, m_i) を送る．挑戦者から受け取った σ_i をそのまま $\mathcal{A}_2$ へと返す．

(6) $\mathcal{B}$ は $\mathcal{A}_2$ が偽造として出力した σ' をそのまま出力する．

攻撃者 $\mathcal{A}$ が勝利するための条件が $\mathsf{Ver}(\mathsf{vk}, \sigma') = (R', m') = (R_{i^*}, m_{i^*})$ であることを考える．$m_{i^*} \notin \{m_i\}_i$ であった場合，(R', m') は $\mathcal{B}$ によってクエリされていないペアであり，$\mathcal{B}$ はこのゲームに勝利する．$m_{i^*} \in \{m_i\}_i$ であった場合，$R' = R_{i^*}$ となる必要がある．この時，$\mathcal{A}_2$ にとっての R_{i^*} のエントロピーは以下で表せる．

$$H_\infty(R_{i^*} | \mathsf{st}, \mathsf{vk}, \{m_i\}_i) \geq S + \mathsf{poly}(\lambda) - S = \mathsf{poly}(\lambda).$$

ゆえに，$\mathcal{A}_2$ にとって R_{i^*} は予測不可能であるため，$\mathcal{A}_2$ が $R' = R_{i^*}$ を生成する確率は無視できる．よって，攻撃者 $\mathcal{B}$ は Sig の EUF-CMA 安全を破る．これは，Sig の EUF-CMA 安全であるという仮定に反する．よって，背理法により，Π が復元不能安全であることが示された． □

5. EUF-CMA 安全性との関係

通常，電子署名には，「自由にメッセージを選択し，それに対する署名を得られる攻撃者でも，新たな署名の偽造が不可能である」という EUF-CMA 安全性を満たすことが求められる．復元不能署名では，この EUF-CMA 安全性

で求められる条件を満たすような安全性を定義しており，復元不能安全性を満たすとき，EUF-CMA 安全性も満たすことは定義より自明である．

本章では，EUF-CMA 安全性は満たすが，復元不能安全性は満たさない方式 (セパレーション) が存在することを証明し，復元不能安全性が EUF-CMA 安全性よりも真に強いことを示す．

5.1 構成

λ, S をセキュリティパラメータとして設定する．ここで，$\mathsf{Sig} = (\mathsf{Gen}, \mathsf{Sign}, \mathsf{Ver})$ をメッセージ空間 $\{0,1\}^{L_m}$ の電子署名方式とする．本構成 $\Pi^{\mathsf{sep1}} = (\mathsf{Gen}, \mathsf{Sign}, \mathsf{Ver})$ は以下である．

$\mathsf{Gen}(1^\lambda, 1^S)$： $\mathsf{Sig.Gen}(1^\lambda)$ を実行し，$(\mathsf{Sig.vk}, \mathsf{Sig.sk})$ を入手する．$\mathsf{vk} = \mathsf{Sig.vk}$, $\mathsf{sk} = \mathsf{Sig.sk}$ として出力する．

$\mathsf{Sign}(\mathsf{sk}, m)$： $\sigma \leftarrow \mathsf{Sig.Sign}(\mathsf{sk}, m)$ を実行し，σ を出力する．

$\mathsf{Ver}(\mathsf{vk}, \sigma)$： $m \leftarrow \mathsf{Sig.Ver}(\mathsf{vk}, \sigma)$ を実行し，m を出力する．

5.2 EUF-CMA 安全性を満たす証明

定理 2. Sig が EUF-CMA 安全な電子署名方式であるとき，本構成 Π^{sep1} は EUF-CMA 安全である．

証明. 本構成 Π^{sep1} における EUF-CMA 安全を破る攻撃者 $\mathcal{A}$ を用いて，Sig の EUF-CMA 安全を破る攻撃者 $\mathcal{B}$ を構成する．攻撃者 $\mathcal{B}$ は以下の通りに動作する．

(1) $\mathcal{B}$ は 1^λ を $\mathcal{A}$ に送る．

(2) $\mathcal{B}$ は vk を挑戦者から受け取り，それを $\mathcal{A}$ に送る．

(3) $\mathcal{A}$ が行った署名クエリ m_i 毎に，$\mathcal{B}$ は挑戦者に署名クエリ m_i を送る．挑戦者から受け取った σ_i をそのまま $\mathcal{A}$ へと返す．

(4) $\mathcal{B}$ は $\mathcal{A}$ が偽造として出力した σ' をそのまま出力する．

本構成より明らかに，$\mathcal{B}$ が Sig の EUF-CMA 安全性を破る確率は $\mathcal{A}$ が Π^{sep1} の EUF-CMA 安全性を破る確率と等しくなる．したがって，

$$\Pr\left[\mathsf{SigForge}^{\text{EUF-CMA}}_{\mathcal{A},\Pi}(1^\lambda) = 1\right] > \mathsf{negl}(\lambda).$$

のとき，

$$\Pr\left[\mathsf{SigForge}^{\text{EUF-CMA}}_{\mathcal{B},\mathsf{Sig}}(1^\lambda) = 1\right] > \mathsf{negl}(\lambda).$$

を満たし，攻撃者 $\mathcal{B}$ は Sig の EUF-CMA 安全性を破る．これは，Sig の EUF-CMA 安全であるという仮定に反する．よって，背理法により，Π^{sep1} が EUF-CMA 安全性を満たすことが示された． □

5.3 復元不能安全性を破る証明

定理 3. 本構成 Π^{sep1} の復元不能安全性を破る PPT 攻撃者が存在する

証明. 本構成 Π^{sep1} における復元不能安全性を破る攻撃者 $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$ を構成する．ここで，Π^{sep1} の署名アルゴリズムの出力サイズを $|\sigma|$ とする．攻撃者 $\mathcal{A}$ は以下の通りに動作する．

(1) 挑戦者は 1^λ を $\mathcal{A}_1$ に送る．その後，$\mathcal{A}_1$ は $S > |\sigma|$ であるような 1^S を出力する．

(2) $\mathcal{A}_1$ は vk を挑戦者から受け取り，その後，aux を出力する．

(3) $q = \mathsf{poly}(\lambda)$ ラウンドにおいて，$\mathcal{A}_1$ は挑戦者に署名クエリ m_i を送り，σ_i を受け取る．

(4) $\mathcal{A}_1$ は署名クエリに送った m_i の中から i^* を選び，$\mathsf{st} = \sigma_{i^*}$ として，$\mathsf{vk}, i^*, \mathsf{st}, \mathsf{aux}$ を挑戦者へ送る．

(5) $\mathcal{A}_2$ は $\mathsf{vk}, i^*, \mathsf{st}, \mathsf{aux}$ を受け取ると，$\mathsf{st} := \sigma_{i^*}$ と展開する．最後に，$\sigma' = \sigma_{i^*}$ を偽造として出力する．

いま，$\mathcal{A}_2$ の出力は挑戦者から受け取った署名 σ_{i^*} であり，これは，$\mathsf{Ver}(\mathsf{vk}, \sigma_{i^*}) = m_{i^*}$ となるはずである．つまり，

$$\Pr\left[\mathsf{SigForge}^{\mathsf{NR}}_{\mathcal{A},\Pi^{\mathsf{sep1}}}(1^\lambda) = 1\right] = 1.$$

が成り立つ．したがって，

$$\Pr\left[\mathsf{SigForge}^{\mathsf{NR}}_{\mathcal{A},\Pi^{\mathsf{sep1}}}(1^\lambda) = 1\right] > \mathsf{negl}(\lambda).$$

を満たし，攻撃者 $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$ は復元不能安全性を破る． □

6. 圧縮不能署名との関係

圧縮不能署名は，署名を大きなサイズにすることで，攻撃者が偽造対象のメッセージに対して，すでに生成された署名を再利用（または，一部の漏洩した署名情報に基づいて再生成）しようとした場合，過去に生成された署名を「ほぼそのまま」保存していなければ偽造が成功しないように設計された電子署名技術である．

攻撃者がターゲット署名の部分情報を入手する状況をモデル化している点は，復元不能署名と圧縮不能署名の安全性において共通しているが，両者にはターゲット以外の署名に対する制限に違いが存在する．復元不能安全性では，攻撃者はターゲット以外の署名については自由に入手できる．これに対し，圧縮不能安全性では，全ての署名について，入手できる情報に制限がある．つまり，復元不能安全性の方が圧縮不能安全性よりも攻撃者に対する制限が弱く，安全性としては強いことが予想できる．

本章では，復元不能安全性が圧縮不能安全性よりも真に

強いことを示す．具体的には，復元不能安全性が圧縮不能安全性を包含することと，圧縮不能安全性は満たすが，復元不能安全性は満たさない方式（セパレーション）が存在することを証明する．

6.1 包含関係

ここでは，復元不能安全性が圧縮不能安全性を包含するということ示す．

定理 4. 復元不能安全性は圧縮不能安全性を包含する．

証明. 圧縮不能安全性を破る攻撃者 $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$ を用いて，復元不能安全性を破る攻撃者 $\mathcal{B} = (\mathcal{B}_1, \mathcal{B}_2)$ を構成する．このとき，攻撃者 $\mathcal{B}_1, \mathcal{B}_2$ はそれぞれ $\mathcal{A}_1, \mathcal{A}_2$ を用いることができるとする．攻撃者 $\mathcal{B}$ は以下の通りに動作する．

(1) $\mathcal{B}_1$ は 1^λ を $\mathcal{A}_1$ に送る．その後，$\mathcal{B}_1$ は $\mathcal{A}_1$ から 1^S を受け取り，挑戦者に 1^S を渡す．

(2) $\mathcal{B}_1$ は vk を挑戦者から受け取り，それを $\mathcal{A}_1$ に送る．その後，$\mathcal{B}_1$ は $\mathcal{A}_1$ から aux を受け取る．

(3) $\mathcal{A}_1$ が行った署名クエリ m_i 毎に，$\mathcal{B}_1$ は挑戦者に署名クエリ m_i を送る．$\mathcal{B}_1$ は挑戦者から受け取った σ_i をそのまま $\mathcal{A}_1$ へと返す．

(4) $\mathcal{A}_1$ が st を生成したら，$\mathcal{B}_1$ は st を受け取る．次に，$\mathcal{B}_1$ は $\mathsf{vk}, \mathsf{st}, \mathsf{aux}$ を $\mathcal{B}_2$ へ送る．最後に，$\mathcal{B}_2$ は $\mathsf{vk}, \mathsf{st}, \mathsf{aux}$ を $\mathcal{A}_2$ へ送る．

(5) $\mathcal{B}_2$ は $\mathcal{A}_2$ が偽造として出力した σ' をそのまま出力する．

攻撃者 $\mathcal{A}$ が勝利するための条件が $\mathsf{Ver}(\mathsf{vk}, \sigma') = (R', m') \neq \perp$ であることを考える．$m' \notin \{m_i\}_i$ であった場合，(R', m') は $\mathcal{B}$ によってクエリされていないペアであり，$\mathcal{B}$ はこのゲームに勝利する．$m' \in \{m_i\}_i$ であった場合，$R' = R$ となる必要がある．この時，$\mathcal{A}_2$ にとっての R_j のエントロピーは以下で表せる．

$$H_\infty(R|\mathsf{st}, \mathsf{vk}, \{m_i\}_i) \geq S + \mathsf{poly}(\lambda) - S = \mathsf{poly}(\lambda).$$

ゆえに，$\mathcal{A}_2$ にとって R は予測不可能であるため，$\mathcal{A}_2$ が $R' = R$ を生成する確率は無視できる．よって，攻撃者 $\mathcal{B}$ は復元不能安全性を破る．すなわち，圧縮不能安全性を破る攻撃者が存在するとき，復元不能安全性を破る攻撃者を構成できる．この命題の対偶を考えて，復元不能安全は圧縮不能安全を包含することが示された． □

6.2 セパレーション

ここでは，圧縮不能安全性は満たすが，復元不能安全性は満たさない方式（セパレーション）が存在することを示す．

6.2.1 構成

λ, S をセキュリティパラメータとして設定する．ここで，$n = S + \mathsf{poly}(\lambda)$ として $\mathsf{Sig}_R = (\mathsf{Gen}_R, \mathsf{Sign}_R, \mathsf{Ver}_R)$ をメッセージ空間 $\{0,1\}^n$ の電子署名方式，$\mathsf{Sig}_m = (\mathsf{Gen}_m, \mathsf{Sign}_m, \mathsf{Ver}_m)$ をメッセージ空間 $\{0,1\}^{L_m}$ の電子署名方式とする．本構成 $\Pi^{\mathsf{sep2}} = (\mathsf{Gen}, \mathsf{Sign}, \mathsf{Ver})$ は以下である．

$\mathsf{Gen}(1^\lambda, 1^S)$: $\mathsf{Sig}.\mathsf{Gen}_R(1^\lambda)$ と $\mathsf{Sig}.\mathsf{Gen}_m(1^\lambda)$ を実行し，$(\mathsf{Sig}_R.\mathsf{vk}, \mathsf{Sig}_R.\mathsf{sk})$ と $(\mathsf{Sig}_m.\mathsf{vk}, \mathsf{Sig}_m.\mathsf{sk})$ を入手する．乱数 $R \in \{0,1\}^n$ をサンプリングする．$\mathsf{vk} = (\mathsf{Sig}_R.\mathsf{vk}, \mathsf{Sig}_m.\mathsf{vk})$, $\mathsf{sk} = (\mathsf{Sig}_R.\mathsf{sk}, \mathsf{Sig}_m.\mathsf{sk}, R)$ として出力する．

$\mathsf{Sign}(\mathsf{sk}, m)$: $\sigma_R \leftarrow \mathsf{Sig}_R.\mathsf{Sign}(\mathsf{Sig}_R.\mathsf{sk}, R)$ と $\sigma_m \leftarrow \mathsf{Sig}.\mathsf{Sign}_m(\mathsf{Sig}_m.\mathsf{sk}, m)$ を実行し，$\sigma = (\sigma_R, \sigma_m)$ として出力する．

$\mathsf{Ver}(\mathsf{vk}, \sigma)$: $R \leftarrow \mathsf{Sig}_R.\mathsf{Ver}(\mathsf{Sig}_R.\mathsf{vk}, \sigma_R)$ と $m \leftarrow \mathsf{Sig}_m.\mathsf{Ver}(\mathsf{Sig}_m.\mathsf{vk}, \sigma_m)$ を実行し，R と m を得る．$R = \perp$ であれば，$\perp$ を出力する．$m = \perp$ であれば，$\perp$ を出力する．そうでなければ，m を出力する．

6.2.2 圧縮不能安全性を満たす証明

定理 5. Sig が EUF-CMA 安全な電子署名方式であるとき，本構成 Π^{sep2} は圧縮不能安全である．

証明. 本構成 Π^{sep2} における圧縮不能署名の安全性を破る攻撃者 $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$ を用いて，Sig の EUF-CMA 安全を破る攻撃者 $\mathcal{B}$ を構成する．攻撃者 $\mathcal{B}$ は以下の通りに動作する．

(1) $\mathcal{B}$ は 1^λ を $\mathcal{A}_1$ に送る．その後，1^S を受け取り，$n = S + \mathsf{poly}(\lambda)$ とする．また，乱数 $R \in \{0,1\}^n$ をサンプリングする．

(2) $\mathcal{B}$ は vk を挑戦者から受け取り，それを $\mathcal{A}_1$ に送る．その後，aux を受け取る．

(3) $\mathcal{A}_1$ が行った署名クエリ m_i 毎に，$\mathcal{B}$ は挑戦者に署名クエリ (R, m_i) を送る．挑戦者から受け取った σ_i をそのまま $\mathcal{A}_1$ へと返す．

(4) $\mathcal{A}_1$ が st を生成したら，$\mathsf{vk}, \mathsf{st}, \mathsf{aux}$ を $\mathcal{A}_2$ へ送る．

(5) $\mathcal{B}$ は $\mathcal{A}_2$ が偽造として出力した σ' をそのまま出力する．

攻撃者 $\mathcal{A}$ が勝利するための条件が $\mathsf{Ver}(\mathsf{vk}, \sigma') = (R', m') \neq \perp$ であることを考える．$m' \notin \{m_i\}_i$ であった場合，(R', m') は $\mathcal{B}$ によってクエリされていないペアであり，$\mathcal{B}$ はこのゲームに勝利する．$m' \in \{m_i\}_i$ であった場合，$R' = R$ となる必要がある．この時，$\mathcal{A}_2$ にとっての R_j のエントロピーは以下で表せる．

$$H_\infty(R|\mathsf{st}, \mathsf{vk}, \{m_i\}_i) \geq S + \mathsf{poly}(\lambda) - S = \mathsf{poly}(\lambda).$$

ゆえに，$\mathcal{A}_2$ にとって R は予測不可能であるため，$\mathcal{A}_2$ が $R' = R$ を生成する確率は無視できる．よって，攻撃者 $\mathcal{B}$ は Sig の EUF-CMA 安全を破る．これは，Sig の EUF-CMA 安全であるという仮定に反する．よって，背理法により，Π^{sep2} は圧縮不能安全であることが示された． □

6.2.3 復元不能安全性を破る証明

定理 6. 本構成 Π^{sep2} の復元不能安全を破る PPT 攻撃者が存在する

証明. 本構成 Π^{sep2} における復元不能署名の安全性を破る攻撃者 $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$ を構成する．ここで，Π^{sep2} の署名アルゴリズムの出力サイズを $|\sigma|$ とする．攻撃者 $\mathcal{A}$ は以下の通りに動作する．

(1) 挑戦者は 1^λ を $\mathcal{A}_1$ に送る．その後，$\mathcal{A}_1$ は $S > |\sigma|$ であるような 1^S を出力する．挑戦者は，$n = S + \mathsf{poly}(\lambda)$ として，乱数 $R \in \{0,1\}^n$ をサンプリングする．

(2) $\mathcal{A}_1$ は vk を挑戦者から受け取り，その後，aux を出力する．

(3) $q = \mathsf{poly}(\lambda)$ ラウンドにおいて，$\mathcal{A}_1$ は挑戦者に署名クエリ m_i を送り，σ_i を受け取る．

(4) $\mathcal{A}_1$ は署名クエリに送った m_i の中から i^* を選び，$\mathsf{st} = \sigma_{m_{i^*}}$ として，$\mathsf{vk}, i^*, \mathsf{st}, \mathsf{aux}$ を $\mathcal{A}_2$ へ送る．

(5) $\mathcal{A}_2$ は $\mathsf{vk}, i^*, \mathsf{st}, \mathsf{aux}$ を受け取ると，$\mathsf{st} \coloneqq \sigma_{m_{i^*}}$ と展開する．

(6) $r = \mathsf{poly}(\lambda)$ ラウンドにおいて，$\mathcal{A}_2$ は挑戦者に m_{i^*} ではない署名クエリ m_j を送り，σ_j を受け取る．

(7) $\mathcal{A}_2$ は σ_j を受け取ると，$\sigma_j \coloneqq (\sigma_R, \sigma_{m_j})$ と展開し，σ_R を得る．最後に，$\sigma' = (\sigma_R, \sigma_{m_{i^*}})$ を偽造として出力する．

いま，$\mathcal{A}_2$ の出力は $\sigma' = (\sigma_R, \sigma_{m_{i^*}}) = \sigma_{i^*}$ であり，これは，$\mathsf{Ver}(\mathsf{vk}, \sigma_{i^*}) = m_{i^*}$ となるはずである．つまり，

$$\Pr\left[\mathsf{SigForge}^{\mathsf{NR}}_{\mathcal{A},\Pi}(1^\lambda) = 1\right] = 1.$$

が成り立つ．したがって，

$$\Pr\left[\mathsf{SigForge}^{\mathsf{NR}}_{\mathcal{A},\Pi}(1^\lambda) = 1\right] > \mathsf{negl}(\lambda).$$

を満たし，攻撃者 $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$ は復元不能安全性を破る． □

7. まとめ

本稿では，通信路やデータベースに対するサイドチャネル攻撃によって署名の一部が漏洩する状況を想定し，復元不能署名を新たに提案した．技術的には，復元不能署名の厳密な定義を与え，通常の電子署名方式に基づく具体的な構成を示した．また，従来の電子署名方式や既存研究である圧縮不能署名と比較し，本方式がより強力な安全性概念を実現していることを明らかにした．

謝辞 本研究の一部は JSPS 科研費 JP23K24846, JP24K20776, JP25KJ1319, JST CREST JPMJCR22M1 の助成を受けたものです．また，本研究の一部は，JST 経済安全保障重要技術育成プログラム【JPMJKP24U2】の支援を受けたものです．

参考文献

[1] M. Bellare, D. Cash, and R. Miller. Cryptography secure against related-key attacks and tampering. *ASIACRYPT 2011*, *LNCS* 7073, pp. 486–503.

[2] M. Bellare and S. K. Miner. A forward-secure digital signature scheme. *CRYPTO'99*, *LNCS* 1666, pp. 431–448.

[3] K. Bhushan, R. Goyal, V. Koppula, V. Narayanan, M. Prabhakaran, and M. S. Rajasree. Leakage-resilient incompressible cryptography: Constructions and barriers. *ASIACRYPT 2024, Part VII*, *LNCS* 15490, pp. 201–234.

[4] E. Boyle, G. Segev, and D. Wichs. Fully leakage-resilient signatures. *EUROCRYPT 2011*, *LNCS* 6632, pp. 89–108.

[5] K. Cohn-Gordon, C. J. F. Cremers, and L. Garratt. On post-compromise security. *CSF 2016 Computer Security Foundations Symposium*, pp. 164–178.

[6] C. Cremers, S. Düzlü, R. Fiedler, M. Fischlin, and C. Janson. BUFFing signature schemes beyond unforgeability and the case of post-quantum signatures. In *2021 IEEE Symposium on Security and Privacy*, pp. 1696–1714.

[7] W. Diffie and M. E. Hellman. New directions in cryptography. *IEEE Transactions on Information Theory*, 22(6):644–654.

[8] Y. Dodis, J. Katz, S. Xu, and M. Yung. Strong key-insulated signature schemes. *PKC 2003*, *LNCS* 2567, pp. 130–144.

[9] S. Faust, E. Kiltz, K. Pietrzak, and G. N. Rothblum. Leakage-resilient signatures. *TCC 2010*, *LNCS* 5978, pp. 343–360.

[10] J. Guan, D. Wichs, and M. Zhandry. Incompressible cryptography. *EUROCRYPT 2022, Part I*, *LNCS* 13275, pp. 700–730.

[11] Y. T. Kalai and L. Reyzin. A survey of leakage-resilient cryptography. Cryptology ePrint Archive, Report 2019/302.