乱数の漏洩に対して安全な署名方式に関する研究

鈴木, 広大 / Suzuki, Kodai

(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)
15
(開始ページ / Start Page)
1
(終了ページ / End Page)
6
(発行年 / Year)
2020-03-24
(URL)
https://doi.org/10.15002/00022725

乱数の漏洩に対して安全な署名方式に関する研究
A Study of Non-Deterministic Signature Scheme Secure Against Nonce Leakage

鈴木 広大*
Kodai Suzuki
法政大学 情報科学研究科 情報科学専攻
Email: 18T0007@cis.k.hosei.ac.jp

***Abstract*—In the real world, documents are signed or stamped to prove that they have been approved at the time of contract. In order to play a similar role in electronic documents such as e-mails, there is a digital signature in which a signature is digitized. Electronic documents are easy to impersonate and falsify, and digital signatures are used to prevent them.**

Schnorr proposed an electronic signature scheme called Schnorr signature. It is considered secure based on the discrete logarithm problem, and its security has been proven by Pointcheval et al. In recent years, due to the simplicity of the system and the robustness of the security, attention has been paid to the proposal of a system applied to the security of virtual currency such as bitcoin.

The Schnorr signature generates a random number when the signature is generated. This is secret information that only the signer knows, but if it leaks, the secret key can be easily calculated. In this paper, we propose a Schnorr signature scheme that is secure even if the random numbers are leaked. We also prove that the signature scheme is secure based on the CDH problem.

***Index Terms*—digital signature, Schnorr Signature, random oracle model, bilinearity**

1. 序論

実際の社会では、契約を交わすときなどに確かに本人が承認したということを証明するために署名や捺印をする. メールや電子的な書類に同様の役割を果たすために, 署名を電子化したものとしてディジタル署名がある. ディジタル署名には公開鍵暗号技術が利用されており, 署名者のみがその秘密鍵を持っているということを利用して本人であることを証明する. 電子文書は紙の文書と比べて印影や筆跡が残らないため, 誰が作成したものか証明しにくく, なりすましや文書改ざんが容易である. ディジタル署名はこのような文書の偽造や改ざんを防止するために利用されている.

電子署名は鍵生成, 署名生成, 署名検証という 3 つのアルゴリズムからなる. 電子署名の正当性は, 鍵生成アルゴリズムで生成された秘密鍵と公開鍵のペアのうち, 秘密鍵を用いて生成された正しい署名が, 公開鍵を用いて検証アルゴリズムで正しく受理されることである. また安全な署名とは, 秘密鍵なくして署名の偽造や改ざんを行えない署名である.

Schnorr はゼロ知識型認証法という個人認証法を基に, シュノア署名と呼ばれる新たな署名方式を提案した. [1][2] シュノア署名は Pointcheval ら [4] によって, ランダムオラクルモデル [7] において離散対数問題が困難という仮定の下で, 選択メッセージ攻撃に対し安全であることが証明されている. シュノア署名は, 同時期に DSA が提唱されたことや, 特許により保護されていたなど, 政治的な理由であまり利用されてこなかった. しかし近年では, 他の署名方式より計算方法が単純でセキュリティが強固であるという点から再び注目を集めており, 特に, メディアなどで脚光を浴び, 利用者が急増しているビットコインなど仮想通貨のセキュリティへ適用する方式 [3] なども提案されている. ビットコインの処理能力には限界があり, 約 10 分間で最大 1MB 分のトランザクションまでしか処理しない仕様となっている. つまり, 利用者がされに増えていくと処理能力が限界を迎えることが明白である. このビットコインのスケーラビリティ問題に対して有効な対策の一つとして, 単純な署名方式であるシュノア署名の採用が議論されている.

シュノア署名は署名生成時に乱数 k を選ぶ. この乱数は署名者のみが知っている秘密情報であるが, 万が一漏洩した場合秘密鍵が簡単に計算できてしまう. そこで我々はこの乱数 k が漏れても安全な非決定的なディジタル署名方式を新たに提案する.

我々の提案する署名方式は, 基本的な構造はシュノア署名と類似しているが, 異なる点としてハッシュ関数を 2 つ使うという点がある. 具体的には, シュノア署名の署名生成時に計算された $s = k - cx$ の値に着目し, この値を 2 つ目のハッシュ関数 H_2 を用いて, $s = H_2(M)^{k-cx}$ とした. このように計算することで, たとえ乱数 k が漏れたとしても秘密鍵が計算できない, より強固な署名方式となる. 我々の提案する署名方式はペアリングを利用し正当性を説明することが出来, CDH 問題の困難性にもとづいてシュノア署名と同程度の安全性を得ることが出来る.

本論では, セクション 2 でディジタル署名について説明し, シュノア署名について記すとともに, 安全性について議論する. また, 新たな署名方式を提案するために必要な, 計算量的問題やペアリングの知識などもここで示す. セクション 3 で提案した署名方式について示す. まず署名のアルゴリズムを定義し, 次に署名の正当性をペアリングを用いて説明する. 最後に, CDH 問題の困難性に基づき安全性の証明をし, その中で我々の提案する署名方式が, シュノア署名と同程度の安全性が確保できることも議論する.

2. 準備と既存研究

本稿では，まずディジタル署名について紹介し，その安全性について記す. その後, シュノアが提案した認証法について紹介し, それを用いたシュノア署名方式について記す．また，新たな署名方式を提案するために必要なその他の知識を記す.

* Supervisor: Prof.　Satoshi Obana

2.1. ディジタル署名

本節ではディジタル署名について説明する． ディジタル署名とは, 書面状の手書きの署名のセキュリティ特性を電子的な文書に模倣するために用いられる公開鍵暗号技術である. メールや電子契約書など, あるメッセージがその作者によって作られたことを検証する仕組みで, 他人がその作者になりすまして署名を作ったり別のメッセージに対する偽の署名を作ったり出来ない. ディジタル署名は, 鍵ペアを生成する鍵生成アルゴリズム, 秘密鍵を用いて署名を生成するアルゴリズム, 公開鍵を用いて署名を検証するアルゴリズムの, 通常 3 つのアルゴリズムから成る. 以下, ディジタル署名の定義を挙げる.

定義 *1.* ディジタル署名とは，以下の性質を満たす三つの確率的多項式時間アルゴリズム (KeyGen， Sign, Verify) の組である.

1) KeyGen: セキュリティパラメータ λ に対して, 公開鍵と秘密鍵の対，(pk, sk) を出力する．ここでは，$(pk, sk) \leftarrow \text{KeyGen}(1^\lambda)$ と書く.
2) Sign: 秘密鍵 sk，メッセージ $m \in \{0,1\}^*$ に対して，署名 σ を出力する．ここでは，$\sigma \leftarrow \text{Sign}(sk, m)$ と書く.
3) Verify: 公開鍵 pk，メッセージ署名 $m \in \{0,1\}^*$, 署名 σ に対して，1 ビット $b \in \{0,1\}$ を出力する．ここで，1 は署名が正しいことを意味し，0 は正しくないことを意味する．また，$b \leftarrow \text{Verify}(pk, m, \sigma)$ と書く.

2.2. ディジタル署名の正当性

ディジタル署名の正当性とは, KeyGen で生成された鍵ペア (pk, sk) のうち, 秘密鍵 sk を用いて Sign で生成された正しい署名が, 公開鍵 pk を用いて Verify で検証した際正しく受理されることである. つまり, $(pk, sk) \leftarrow \text{KeyGen}(1^\lambda)$, $\sigma \leftarrow \text{Sign}(sk, m)$ のとき, $1 \leftarrow \text{Verify}(pk, m, \sigma)$ となる.

2.3. ディジタル署名の安全性

ディジタル署名の安全性の一つに, 第三者によって署名が偽造できないことがあげられる. 偽造には意味のない文書の署名の偽造から, 意味のある文書の署名の偽造まで色々なレベルが存在する. 一方で, 偽造する際に使用する攻撃方法も様々である. 安全な署名方式とは, 攻撃者にとって最も有利な手段を用いても, どんな意味のない文書の偽造も不可能な署名方式であるといえる.

2.3.1. 攻撃の種類. 攻撃者が送信者になりすまして, 署名を偽造するためにどの程度の署名に関する情報を利用できるかで, 攻撃の種類が以下のように分類される.

- 受動的攻撃 (passive attack)：公開鍵などの公開情報のみを利用する攻撃
  - 直接攻撃 (direct attack)：公開鍵のみを利用する攻撃
  - 既知平文攻撃 (known message attack)：攻撃者は平文の集合 $\{m_1, \cdots, m_n\}$ に対する正しい署名を入手し, これらの署名情報を利用して第三の平文の署名を偽造する攻撃. 攻撃者はこれらの平文を選ぶことが出来ない.
- 能動的攻撃 (active attack)：攻撃のための情報を署名者から入手できる状態の攻撃
  - 一般的選択平文攻撃 (generic chosen message attack)：攻撃者は任意に選択した平文の集合 $\{m_1, \cdots, m_n\}$ に対する正しい署名を入手し, これらの署名情報を利用して第三の平文の署名を偽造する攻撃. 攻撃者はこれらの平文を選べるが, 平文は固定されており, 署名を見る前にすべての平文のリストを作っておく必要がある.
  - 適応的選択平文攻撃 (adaptive chosen message attack)：攻撃者は, 毎回適応的に任意に選んだ平文に対して真の署名者に署名させ, そこで得た情報を利用して第三の平文の署名を偽造する攻撃.

2.3.2. 偽造の種類. 署名の偽造の解読レベルは以下のように分類される.

- 全面解読 (total break)：署名者の秘密鍵が計算できる
- 一般的偽造 (universal forgery)：署名アルゴリズムと機能的に等価なアルゴリズムを効率的に見つけられる. 任意の平文の署名が偽造可能になる.
- 選択的偽造 (selective forgery)：攻撃者が改め選んだ特定の平文に対する署名分の偽造が出来る.
- 存在的偽造 (existential forgery)：少なくとも一つの平文に対する署名分の偽造が出来る.

2.3.3. 安全性のレベル. 署名の安全性のレベルは, 前節で述べた偽造の種類と攻撃の種類を用いて議論する. つまり, (偽造の種類)×(攻撃) により安全性のレベルを決定する. 偽造に関しては存在的偽造が被害のレベルが低い. 一方, 攻撃に関しては適正的選択平文攻攻撃が攻撃能力が高い. よって, 最も安全な署名とは, 適応的選択平文攻撃のもと存在的偽造が出来ない署名と定義することが出来る.

2.4. 離散対数問題

G を位数が大きな素数 q の巡回群とし, その生成元を g とする. このとき, (g, y) から $y = g^x$ となる整数 x を求めよ, という問題を離散対数問題という.

2.5. CDH 問題

G を位数が大きな素数 q の巡回群とし, その生成元を g とする. このとき, (g, g^a, g^b) から g^{ab} とを求めよ, という問題を CDH 問題という. また, CDH 問題を解く効率的なアルゴリズムが存在しないという仮定のことを CDH 仮定と呼ぶ.

2.6. 楕円曲線上のペアリング [8]

本稿では, ペアリングと呼ばれる双線形写像について述べる. 楕円曲線とは, 一般的に $y^2 + a_1xy + a_3y = x^3 + a_2x^2 + a4x + a_6$ で与えられる (x, y) に関する方程式のことである. また, 有限体 $\mathbb{F}_q$ 上の楕円曲線は一般的に $E/\mathbb{F}_q$ と表される. 楕円曲線上の有利点の集合を $E(\mathbb{F}_q)$ で表したとき, $E(\mathbb{F}_q)$ は群をなす. ねじれ点を $E[n] := \{P \in E(\overline{\mathbb{F}}_q) \| nP \in \mathcal{O}\}$ と定義すると, $E[n]$ 上でペアリングと呼ばれる双線形写像が定義できる. ペア

リングは, $e : E[n] \times E[n] \to \mathbb{F}_{q^k}$ で定義され, 任意の点 $P, Q, R \in E[n]$ に対して,

- bilinearity

$$e(P, Q \cdot R) = e(P, Q) \cdot e(P, R)$$

$$e(P \cdot Q, R) = e(P, R) \cdot e(Q, R)$$

- non-degeneracity

$$e(P, Q) = 1 \Leftrightarrow P = 0$$

という 2 つの性質を持ち, 双線形性から直ちに, ある整数 a, b に対して,

$$e(aP, bQ) = e(bP, aQ) = e(P, Q)^{ab}$$

という関係が導ける. このような楕円曲線状のペアリングの値 $e(P, Q)$ は, Miller のアルゴリズム [9] などによって効率的に計算可能である.

ペアリングは, ID ベース暗号 [8] や放送型暗号 [10] など, 近年提案されている暗号方式で用いられていたり, BLS 署名 [11] などの電子署名方式にも応用されている. 我々が新たに提案する署名方式もペアリングを用いて署名の正当性を説明している.

2.7. ランダムオラクルモデル

多くの暗号方式ではハッシュ関数 h を使用しており, これらのセキュリティ証明には Bellare と Rogaway[7] が形式化した, ランダムオラクルモデルと呼ばれるモデルを利用することがある.

ランダムオラクルとは, 新しいクエリごとにランダムな値を生成するオラクルである. すなわち, ランダムオラクル $H : X \to Y$ は, 質問 $x \in X$ に対しランダムに選んだ $y \in Y$ を返す. 敵は, 必要に応じてランダムオラクル H に x を質問し, $H(x)$ の値 y を教えてもらうと仮定する. それ以降, $H(x)$ の値は $H(m) = y$ と確定する. このようなモデルをランダムオラクルモデルという. このモデルの証明は, ハッシュ関数に脆弱性が無い限り署名方式全体の安全性を確保するものであると主張されている.

2.8. シュノアの認証法

証明者 A が検証者 B に「自分は A だ」ということを証明したい. A の公開鍵が $y = g^x$ の場合, もっとも単純な方法は秘密鍵 x を B に送ることである. しかし, これだと B は秘密鍵 x を知ることになり, A になりすませてしまう. これを防ぐために x を秘密にしたまま x を知っている, ということだけを B に証明する. このような個人認証法をゼロ知識型認証法といい, シュノアは離散対数問題に基づき, 以下のようなゼロ知識認証法を示した.

- 登録段階
  1) 証明者 A は, 秘密鍵 $x \in Z_q$ をランダムに選び, 公開鍵 $y = g^x$ を計算する. その後,A は y を信頼できるセンタ T に登録する.
  2) T は,y を A の公開情報として公開する
- 認証プロトコル
  1) 証明者 A は $k \in Z_q$ をランダムに選び,$r = g^k$ を計算し,r を検証者 B に送る.
  2) B は,$c \in Z_q$ をランダムに選び A に送る.
  3) A は,$s = k - cx mod q$ を計算し s を B に送る.
  4) B は,$r = g^s y^c$ が成り立つかどうかチェックする. 成り立てば受理し, 成り立たなければ拒否する.

$$g^s y^c = g^{k-cx} g^{cx} = g^k = r$$

より, 認証の式が成立する.

2.9. シュノアの認証法の安全性

シュノアの認証法は離散対数問題が困難という仮定の下で受動攻撃に対して安全である.

定理 *1.* 敵は, 盗聴せずとも通信経路 (r, c, s) を生成できる.

証明 *1.*

1) c をランダムに選ぶ
2) s をランダムに選ぶ
3) $r = g^s y^c$ を計算し,r を求める.

定理 *2.* なりすましに成功するアルゴリズム A' が存在したと仮定すると,x を求めることが出来る効率的なアルゴリズム M が存在する.

証明 *2.*

1) M は,A' を証明者として認証プロトコルを実行する. ただし,M 自身が検証者となる. その通信経路を (r, c_1, s_1) とする.
2) M は A' を初期状態に戻し, 再度認証プロトコルを実行する. その通信経路を (r, c_2, s_2) とする.
3) (r, c_1, s_1),(r, c_2, s_2) が正しい通信経路で, かつ,$c_1 \neq c_2$ であると仮定すると

$$r = g^{s_1} y^{c_1} = g^{s_2} y^{c_2}$$

が成り立つ. これら 2 つの式より

$$g^{s_1 - s_2} = y^{c_2 - c_1}$$

が得られる. よって

$$y = g^x = g^{(s_1 - s_2)/(c_2 - c_1)}$$

つまり

$$x = \frac{(s_1 - s_2)}{(c_2 - c_1)} \ mod \ q$$

これは, M が離散対数問題を解いたことと同義である. つまり, シュノアの認証法は離散対数問題に基づき安全である. □

2.10. シュノア署名方式

シュノアは, シュノアの認証法を基に, シュノア署名と呼ばれるディジタル署名方式を提案した [1][2]. シュノア署名は, 認証法では検証者 B がランダムで選んでいた値 c をハッシュ関数 H と署名したいメッセージ m を用いて A が自分自身で計算する. 残りのパラメータは認証法と同様に計算し, $\sigma = (c, s)$ を署名とする. 以下 3 つのアルゴリズムを示す.

- KeyGen:秘密鍵 $sk = x \in Z_q$ をランダムに選び, 公開鍵 $pk = y = g^x$ を計算する.

- Sign:メッセージ m に対し, 署名 σ を以下のように計算する.
  1) $k \in Z_q$ をランダムに選び,$r = g^k$ を計算する.
  2) ハッシュ関数 H を用いて $c = H(m||r)$ を計算する.
  3) $s = k - cx \ mod \ q$ を計算する.

  最後に,$\sigma = (c, s)$ とする.
- Verify:受信者は, 受け取った署名 $\sigma = (c, s)$ と公開鍵 $y = g^x$ から

$$r_v = g^s y^c$$

を計算する. 次に

$$c_v = H(m||r_v)$$

を計算し,$c = c_v$ が成り立てば署名を受理し, 成り立たなければ拒否する.

2.10.1. シュノア署名の正当性. シュノア署名の正当性は以下のとおりである.

$$r_v = g^s y^c = g^{k-cx} g^{cx} = g^k = r$$
$$c_v = H(m||r_v) = H(m||r) = c$$

より, 正しい署名の時受理できる.

2.10.2. シュノア署名の安全性. シュノア署名は Pointcheval ら [4] によって, ランダムオラクルモデルにおいて, 離散対数問題が困難という仮定の下で選択メッセージ攻撃に対し安全であることが証明されている. また, Pointcheval らは安全性の証明において以下のような定理を示している.

定理 *3.* Forking Lemma : A を、入力が公開された値のみで構成される確率的多項式時間アルゴリズムとする. A がランダムオラクル H に要求できるクエリの数を Q, 署名者に要求できるクエリの数を R とする. 時間 T 内で A は, 確率 $\varepsilon \geq 10(R+1)(R+Q)/2^k$ で有効な署名 (m, r, c, s) を生成できるとする. 署名 (r, c, s) を秘密鍵を知らずにシミュレートできる場合, (m, r, c, s), (m, r, c', s') という二つの有効な署名を $c \neq c'$ という条件下で時間 $T' \leq 120686QT/\varepsilon$ で出力する.

シュノア署名の安全性は, 署名オラクルとランダムオラクルによって生成された (r, c, s) が秘密鍵を知らなくてもシミュレートできることを示せば, 定理 2 と定理 3 から結果が得られる. また, 敵 A が H オラクルに高々 n 回質問し, 偽造に成功する確率を $Adv_A(n)$ で表すとすると, 以下の定理が成り立つ.

定理 *4.* シュノア署名に対し, 署名の偽造に $Adv_A(n) > \epsilon$ で成功する効率的なアルゴリズム A が存在すると仮定すると, 受動的攻撃により, 確率 ϵ/n でなりすましに成功する効率的なアルゴリズム B が存在する.

証明 *3.* B は A をサブルーチンとして利用する. B は A の H オラクルへの n 番目の質問が偽造用の (m^*, r^*) であると推測する. このとき, B は H オラクルの応答を以下のようにシミュレートしなりすましを行う.

1) B は r^* をなりすます相手 C に送る.
2) C は何らかの c^* を返してくる.
3) B はこの c^* を $H(m^*||r^*)$ の値として A に返す.
4) A は最後に偽造 (m^*, r^*, c^*, s^*) を出力する.
5) B は s^* を C に送る.

ここで, (m^*, r^*, c^*, s^*) が正しい偽造であれば B はなりすましに成功する. 正しい偽造であれば, $r^* = g^{s^*} y^{c^*}$ が成り立つ. これは (r^*, c^*, s^*) が正しい通信系列であることを示しているため, 検証者は署名を受理する. A が m を署名オラクルに質問したとき, B は以下のようにシミュレートする.

1) B は (c, s) をランダムに選び, $\sigma = (c, s)$ を返す.
2) $r = g^s y^c$ とおき, B は (m, r, c) をテーブルに記憶しておく.

A が (m_i, r_i) を H オラクルに質問してきたとき, B は H オラクルを以下のようにシミュレートする.

- (m_i, r_i, c_i) がテーブルにある場合, B は c_i を A に返す.
- そうでない場合, B は $c_i = H(m_i||ri)$ を A に返し, (m_i, r_i, c_i) をテーブルに記憶しておく.

以上より, 署名オラクルとランダムオラクルのシミュレートが完了した. これにより, 定理 3 から B は A から偽造に成功する 2 つの署名 (m^*, r^*, c_1^*, s_1^*), (m^*, r^*, c_2^*, s_2^*) を受け取ることが出来る. 最後に B はこの 2 つの署名を用いて, 定理 2 と同様に離散対数問題を解く.

$$r^* = g^{s_1^*} y^{c_1^*} = g^{s*_2} y^{c_2^*}$$

である. よって

$$y = g^x = g^{(s_1^* - s_2^*)/(c_2^* - c_1^*)}$$

つまり,

$$x = \frac{(s_1^* - s_2^*)}{(c_2^* - c_1^*)}$$

B は入力 (g, g^x) から x を求めることが出来, 確率 $(\epsilon/n)^2$ で離散対数問題を解いたことになる. 以上より, シュノア署名はランダムオラクルモデルにおいて離散対数問題が困難であるという仮定の下選択平文攻撃に対して安全であるということが証明された. □

3. 乱数の漏洩に対して安全な署名方式

シュノア署名は離散対数問題が困難という仮定の下で安全であるが, 署名者 A がランダムに選択した値 k が漏れた場合, 公開された署名 σ から以下のように秘密鍵 x が計算できてしまう.

$$\begin{aligned} \sigma &= (c, s) \\ s &= k - cx \end{aligned}$$

より,k が漏れると,

$$x = (k - s)/c$$

が, 計算できる.
そこで本論では, 乱数 k が漏れても秘密鍵 x が漏れない, 安全な署名方式を新たに提案する.

3.1. 提案する署名方式

シュノア署名は, s を求める式が単調であったので, k が漏れた場合に秘密鍵が簡単に計算できてしまった. そこで我々は署名の s に着目し, この値をより複雑にしていくことで, 乱数 k が漏れても秘密鍵が漏れないような署名方式を提案する.

我々はまず, 離散対数問題の考えに基づき, s の値を

$$s = g^{k-cx}$$

として計算した. しかしこれは

$$s = g^{k-cx} = g^k y^{-c}$$

となるため, 乱数 k が漏れていたと仮定すると, 秘密鍵 x を知らなくても, 公開された値のみで誰でも署名を作ることが出来てしまう. つまり, 簡単に署名者のなりすましが行えてしまい, 安全ではない.

そこで次に, ハッシュ関数 $H_2 : H_2(m) \in G'$ を用いて

$$s = H_2(m)^{k-cx}$$

と定めた. これにより, 離散対数問題から $k - cx$ は求められず, かつ秘密鍵無しでは署名を生成することが出来なくなった.

この s の値を基準にして, ペアリング関数を使い r など他のパラメータや検証時の判定式などを定めた.

また, 我々の提案する署名方式は, CDH 問題が困難であるという仮定の下で安全な方式であることをのちに証明する. 以下具体的な署名方式を示す.

新たに提案する署名方式は以下の 3 つのアルゴリズムで構成される.

- KeyGen:セキュリティパラメータ 1^λ を受け取り, 公開パラメータを (G, G', g, H_1, H_2, e), 秘密鍵 sk, 公開鍵 pk を出力する. 秘密鍵は $sk = x \in Z_q$ をランダムに選び, 公開鍵は $pk = y = g^x$ を計算する. ここで q は群 G の位数, $H_2(m) \to G'$, e はペアリング関数とする.
- Sign:秘密鍵 sk とメッセージ m を受け取り, 以下のように計算された署名 σ を出力する.
  1) $k \in Z_q$ をランダムに選び,$r = e(g, H_2(m))^k$ を計算する.
  2) ハッシュ関数 H_1 を用いて $c = H_1(m||r)$ を計算する.
  3) $s = H_2(m)^{k-cx}$ を計算する.
  4) 最後に, $\sigma = (c, s)$ とする.
- Verify:受信者は, 受け取った署名 $\sigma = (c, s)$, 公開鍵 $y = g^x$ とメッセージ m から

  $$r_v = e(g, s)e(y^c, H_2(m))$$

  を計算する. 次に

  $$c_v = H(m||r_v)$$

  を計算し,$c = c_v$ が成り立てば署名を受理し, 成り立たなければ拒否する.

3.2. 署名の正当性

本署名の正当性は, 検証式 $r_v = e(g, s)e(y^c, H_2(m))$ に正しい署名 $\sigma = (c, s)$ を用いて計算すると, Sign アルゴリズムで生成した r と検証時に計算する r_v が一致することが示せればよい. 正当性は以下の式で示すことが出来る.

$$\begin{aligned} r_v &= e(g, s)e(y^c, H_2(m)) \\ &= e(g, H_2(m)^{k-cx})e(g^{cx}, H_2(m)) \\ &= e(g, H_2(m))^{k-cx} e(g, H_2(m))^{cx} \\ &= e(g, H_2(m))^k \\ &= r \end{aligned}$$

よって,

$$c_v = H_1(m||r_v) = H_1(m||r) = c$$

より, 正しい署名の時受理できる.

3.3. 署名の安全性

本稿では, 本署名の安全性について論じる. 本署名は, CDH 問題が困難であるという仮定の下で選択平文攻撃に対して安全である. 基本的な証明の流れはシュノア署名同様に, まず署名オラクルとランダムオラクルから生成された (r, c, s) を秘密鍵を知らないという前提のもとシミュレートする. シミュレートが出来ると, 定理 3 から有効な署名を 2 つ出力でき, それらを用いて CDH 問題を解くことで安全性の証明が成立する. 従来のシュノア署名と異なる点は, ハッシュ関数を H_1, H_2 と 2 つ使っているので, ランダムオラクルを 2 つシミュレートする点である. 以下, 証明を示す.

定理 5. 本署名方式は, ランダムオラクルモデルにおいて, CDH 問題が困難であるという仮定の下で選択平文攻撃に対して安全である.

証明：
まず, 署名の偽造に $Adv_A(n) > \epsilon$ で成功する効率的なアルゴリズム A が存在すると仮定すると, 確率 ϵ/n でなりすましに成功する効率的なアルゴリズム B が存在することを示す. B は公開されたパラメータ g, y, e と $h \in G'$ を入力として受け取る.

3.3.1. H_1 オラクルのシミュレート. B は A をサブルーチンとして利用する. B は A の H_1 オラクルへの n 番目の質問が偽造用の (m^*, r^*) であると推測する. このとき, B は H_1 オラクルの応答を以下のようにシミュレートしなりすましを行う.

1) B は r^* をなりすます相手 C に送る.
2) C は何らかの c^* を返してくる.
3) B はこの c^* を $H_1(m^*||r^*)$ の値として A に返す.
4) A は最後に偽造 (m^*, r^*, c^*, s^*) を出力する.
5) B は s^* を C に送る.

ここで, (m^*, r^*, c^*, s^*) が正しい偽造であれば B はなりすましに成功する. 正しい偽造であれば, $r^* = e(g, s^*)e(y^{c^*}, H_2(m^*))$ が成り立つ. これは (r^*, c^*, s^*) が正しい通信系列であることを示しているため, 検証者は署名を受理する.

A が n 番目以外の (m_i, r_i) を H_1 オラクルに質問してきたとき, B は H_1 オラクルを以下のようにシミュレートする.

- (m_i, r_i, c_i) がテーブルにある場合, B は c_i を A に返す.
- そうでない場合, B はランダムな値 $c_i = H(m_i||ri)$ を A に返し, (m_i, r_i, c_i) をテーブルに記憶しておく.

H_1 オラクルでは, 偽造用の署名が n 番目に来ると B が推測することで正しくシミュレートをしているので, この時点でなりすましに成功する確率は $1/n$ 倍されることになる.

3.3.2. H_2 オラクルのシミュレート. B は A をサブルーチンとして利用し, 以下のように H_2 オラクルをシミュレートする.

1) B は乱数 r_m を選択する. .
2) A が H_2 オラクルに m_i を質問してきたとき, B は $h^{r_m} = H_2(m_i)$ を A に返す.

このとき H_1 オラクルと同様に, 偽造に成功すると推測される n 番目の質問 m^* のときは h を返し, それ以外の時はランダムな値を返すとすると, なりすましに成功する確率は $1/n$ 倍され ϵ/n^2 となり, シュノア署名におけるなりすまし成功確率より小さくなる. いついかなる時も H_2 は同じ値 (今回は h^{r_m}) を返すとシミュレートすることができれば, 元のシュノア署名と同様のなりすまし成功確率を保つことが出来る.

3.3.3. 署名オラクルのシミュレート. A が m を署名オラクルに質問したとき, B は以下のようにシミュレートする.

1) B は (c, s) をランダムに選び, $\sigma = (c, s)$ を返す.
2) $r = e(g, s)e(y^c, H_2(m))$ とおき, B は (m, r, c) をテーブルに記憶しておく.

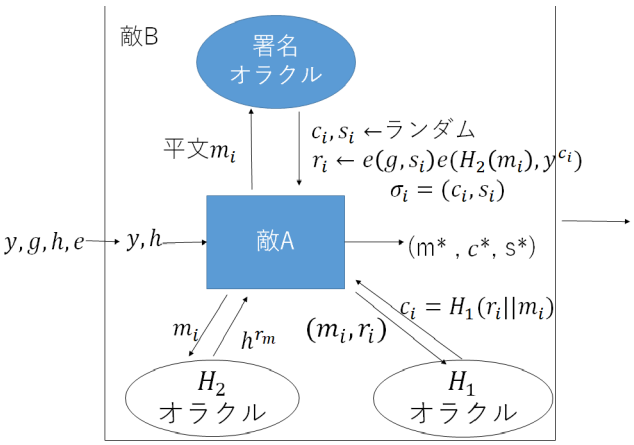

図 1. 各オラクルのシミュレートの図

以上より, 3 つのオラクルのシミュレートが完了した. これにより, 定理 3 から B は 2 つの有効な署名 (m^*, r^*, c_1^*, s_1^*), (m^*, r^*, c_2^*, s_2^*) を, $c_1^* \neq c_2^*$ の条件下で出力する. この 2 つの署名を用いて B は CDH 問題が解けることを以下に示す.

B の入力 $g, y(= g^x), h, e$ であり, 偽造された 2 つの署名は (m^*, r^*, c_1^*, s_1^*), (m^*, r^*, c_2^*, s_2^*) である. このとき,

$$s_1^* = h^{r_m(k - c_1^* x)}$$

$$s_2^* = h^{r_m(k - c_2^* x)}$$

である. 以上 2 つの式より,

$$\frac{s_1^*}{s_2^*} = h^{r_m(c_2^* - c_1^*)x}$$

よって,

$$h^x = \left(\frac{s_1^*}{s_2^*}\right)^{(r_m(c_2^* - c_1^*))^{-1}}$$

これは, 入力 $g, y(= g^x), h$ のとき B が h^x を求めたことと同義である. B は確率 (ϵ/n) で有効な署名を 1 つ出力することから, 二つの署名を出力する際は確率 $(\epsilon/n)^2$ であり, この確率で CDH 問題を解いたことになる.

以上より, 本署名方式は, ランダムオラクルモデルにおいて, CDH 問題が困難であるという仮定の下で選択平文攻撃に対して安全であることが証明された. □

4. 結び

本稿では, シュノア署名の乱数について議論するとともに, その乱数が漏れても安全である新たな署名方式を提案し, 安全性の証明を行った. 我々の提案する署名方式はハッシュ関数を 2 つ使うという点で従来の方式と大きく異なる. 具体的には, シュノア署名の署名生成時に計算された $s = k - cx$ の値を $s = H_2(M)^{k-cx}$ とすることで, たとえ乱数 k が漏れたとしても秘密鍵 x が計算できない, より強固な署名方式となった. 正当性については, ペアリングを使うことにより正しい署名が受理されることを説明した. また, 安全性の証明時にランダムオラクルのシミュレートについて考察し, シュノア署名と同等の安全性が得られることを確認した. 今後の課題として, この署名方式を用いた具体的なアプリケーションの実装などが挙げられる.

参照

[1] C. P. Schnorr. ”Efficient Identification and Signatures for Smart Cards. ” In Crypto ’89, LNCS 435, pages 235-251. Springer-Verlag, Berlin, 1990.

[2] C. P. Schnorr. ”Efficient Signature Generation by Smart Cards. ”Journal of Cryptology, 4(3):161-174, 1991.

[3] Gregory Maxwell, Andrew Poelstra, Yannick Seurin. ”Simple Schnorr Multi-Signatures with Applications to Bitcoin” Designs, Codes and Cryptography, September 2019, Volume 87, Issue 9, pp 2139-2164.

[4] David Pointcheval, Jacques Stern. ”Security Arguments for Digital Signatures and Blind Signatures. ”Journal of Cryptology, June 2000, Volume 13, Issue 3, pp 361-396

[5] David Pointcheval, Serge Vaudenay. ”On Provable Security for Digital Signature Algorithms. ”CCS ’16 Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, Pages 1651-1662.

[6] Shafi Goldwasser, Silvio Micali, and Ronald L. Rivest. ”A Digital Signature Scheme Secure Against Adaptative Chosen-Message Attacks. ”SIAM Journal of Computing, 17(2):281-308, April 1988

[7] Mihir Bellare and Phillip Rogaway. ”Random Oracles Are Practical: a Paradigm for Designing Efficient Protocols. ” In Proc. of the 1st CCCS, pages 62-73. ACM Press, New York, 1993.

[8] ”ID ベース暗号に関する調査報告書”(https://www.cryptrec.go.jp/report/cryptrec-tr-2001-2008.pdf)

[9] V. Miller, ”Short Programs for Functions on Curves” ,1986.(http://crypto.stanford.edu/miller/miller.pdf)

[10] Benny Chor, Amos Fiat, and Moni Naor. ”Tracing traitors”. In Proc. of Crypto’94, Lecture Notes in Computer Science, LNCS 839, Springer Verlag, pages 257?270, 1994.

[11] D. Boneh, B. Lynn, and H. Shacham. ”Short signatures from the weil paring”ASIACRYPT2001, LNCS 2248, pp. 514-532, 2001

[12] 黒澤 馨. ”現代暗号への招待”, サイエンス社, 2010