The 29th Symposium on Information Theory
and its Applications (SITA2006)
Hakodate, Hokkaido, Japan, Nov. 28 – Dec. 1, 2006

リモートバイオメトリクス認証に有効な「近い」ことを示す零知識証明プロトコル

Zero-knowledge interactive proofs for proving nearness of biometrics and its application

尾形わかは * 菊池 浩明 † 西垣 正勝 ‡
Wakaha Ogata Hiroaki Kikuchi Masakatsu Nishigaki

Abstract— We provide zero-knowledge interactive proof protocols for proving "nearness" of two committed vectors. Such protocols are very useful for construction of cancelable biometric authentication systems.

Keywords— biometrics, cryptographic protocol, zero-knowledge interactive proof

1 はじめに

近年，個人認証方式として，生体情報を用いた方式が増加している．生体認証方式は，パスワード認証における忘却，持ち物認証における紛失がないなど，利便性が高いが，生体情報の取り扱いにおいて課題が多い．第一に，プライバシー情報である生体情報を認証サーバに登録することに抵抗を感じるユーザが多い．第二に，生体情報は変更することができないため，生体情報が流出してしまうと二度とその生体情報を使用できなくなるという大きな問題をはらんでいる．後者に関しては，近年，流出しにくい静脈認証などが注目を浴びているが，認証者が生体情報自体を管理している限り，認証者からの流出の可能性がある．

これに対し，Ratha らはキャンセラブルという概念を導入し，画像ブロック置換，マニューシャ非線形変換などの方式を提案した [1]．キャンセラブルな方式では，乱数を用いて加工した生体情報を認証者に登録することによって，登録情報の流出時にはこれを取り消すことができる．また，登録されるのは加工後の生体情報であるため，プライバシーの問題も緩和される．他のキャンセラブルな生体認証方式としては，[2, 3, 4] などがある．

本研究では，非対称暗号システムを利用することにより，認証サーバからの情報漏洩に対して安全なキャンセラブル生体情報認証システムを提案する．

2 生体認証方式

生体認証方式は，あらかじめ生体情報をテンプレートとして登録しておき，認証時には生体情報をテンプレートと比較して差異が小さい場合に限り本人であるとみなす．生体情報の種類によって比較手法・差異の計測手法が異なり，多くの研究者によって誤認証率（FAR，FRR）を小さく抑える手法の開発がされている．

被認証者（ユーザ）と検証者（サーバ）がネットワークを介して認証を行うリモート生体認証においては，IC カードと組み合わせることにより，安全性や利便性の向上を図る試みも多数研究されている．例えば，IC カードがローカルで生体認証を行いサーバは IC カードのみを認証するようにすれば，生体情報のテンプレートは IC カード内に封入することができる上に，IC カードの機器認証にはセキュリティプロトコルを利用できる．しかし，IC カードの盗難やサイドチャネルアタックなどを考えた場合，生体情報の秘匿性を IC カードの耐タンパー性のみに頼ることは妥当ではない．本研究では，IC カードの耐タンパー性を仮定せず，サーバが生体情報と IC カードの双方を認証するモデルを研究対象とする．

2.1 キャンセラブル生体認証

キャンセラブルな生体認証では，何らかの乱数を用いて生体情報をマスクし，マスクされた情報をテンプレートとしてサーバに登録する．安全性のため，生体情報をマスクする乱数はある程度以上の情報量を持つ必要があり，IC カード内で保管される．

認証時には，乱数によってマスクされた生体情報を用いて 2 つの生体情報の比較を行うため，使用する生体情報の比較手法にあわせ，マスク方法を考える必要がある．

太田らによる方式 [2] では，ハミング距離によって比較を行うことができる虹彩情報を対象としたキャンセラブル認証方式を構築している．具体的には，以下のとおりである．

虹彩情報は，m ビット列とみることができる．登録時の虹彩情報 $X \in \{0,1\}^m$ に対して，認証時の虹彩情報が $X' \in \{0,1\}^m$ が

$$W_H(X \oplus X') \leq m\theta$$

を満たすとき，本人であると判定する．ただし，$W_H(\cdot)$ はビット列のハミング重み，$\theta \in [0,1]$ は閾値である．[2] の方式では，テンプレートとして $Y = X \oplus R$ を登録

* 東京工業大学 東京都目黒区大岡山, Tokyo Institute of Technology, Ookayama, Meguro-ku, Tokyo, Japan
† 東海大学 神奈川県平塚市北金目, Tokai University, Kitakaname, Hiratsuka, Kanagawa
‡ 静岡大学 静岡県浜松市城北, Shizuoka University, Johoku, Hamamatsu, Japan,

し，ランダムなビット列 $R \in \{0,1\}^m$ は IC カード内に保管される．認証においては，ユーザは虹彩情報 X' と R を用いて $Y' = X' \oplus R$ を計算し，これをサーバに送信する．サーバはテンプレート Y と受信した Y' から $Y \oplus Y' = X \oplus X'$ を計算し，ハミング重みがしきい値以下であれば本人であると判定する．

比良田らは，２次元画像からなる生体情報を取り扱うことのできるキャンセラブル認証方式を提案している [4]．この方式では，生体情報（２次元画像のフーリエ変換画像）$G(u,v)$ とランダムな $R(u,v)$ の積 $R(u,v)G(u,v)$ がテンプレート，検証時の生体情報（２次元画像のフーリエ変換画像の複素共役）$G^*(u,v)$ に対して $R^{-1}(u,v)G^*(u,v)$ がサーバに送信される情報となる．この場合，サーバで $R^{-1}(u,v)G^*(u,v)R(u,v)G(u,v) = G^*(u,v)G(u,v)$ によって乱数要素が相殺され，２つの２次元画像の相互相関関数を計算することが可能である．

これらの方式では，テンプレートは乱数によってマスクされているため，テンプレートは生体情報について何の情報も漏らさない．また，IC カードからは乱数しか得ることができず，IC カードからも生体情報は漏洩しない．しかし，通信路を盗聴して得た情報を再送（再送攻撃）することによってなりすましが可能であるという欠点を持つ．また，認証プロトコルを行うことによってサーバは２つの生体情報の差異を知ることができるため，サーバがヒルクライミング攻撃 [5] を行うなどの方法により，サーバに生体情報が漏れる可能性がある．

2.2 生体認証の安全性

従来，生体認証の性能は本人拒否率（FRR）と他人受入率（FAR）によって評価されてきた．しかし，キャンセラブル方式のように，乱数要素が付加されサーバによる不正を考慮した方式においては，通常のユーザ認証システムと同様の安全性の評価が必要である．ここでは，攻撃モデルを示すことによって本研究で考慮する安全性について述べる．

攻撃モデルは，攻撃者の目的と，アタックモデルに分類できる．

攻撃者の目的 通常のユーザ認証システムにおける攻撃者の目的である「なりすまし」と，生体認証特有である「生体情報そのものを得ること」の２つとする．従来の生体認証システムでは，本人の生体情報を持つことと，なりすましができることはほぼ等価と考えることができた．しかし，キャンセラブル方式においては，先に示した再送攻撃のように，生体情報を全く知らずになりすましができる場合もあるため，なりすましと生体情報の漏洩を独立に考える必要がある．

アタックモデル 「サーバによる不正」「IC カードの盗難」「通信路の盗聴」のいずれかとする．（同時に２つ以上のアタックはできないとする．）サーバは認証プロトコル中に通信路を流れる情報をすべて見ることができるため，サーバによる不正を防ぐことができれば盗聴者による不正も防ぐことができる．

表 1: 攻撃の種類

	生体情報を得る	なりすまし
サーバによる不正	(A)	—
IC カードの盗難	(B)	(C)
通信路の盗聴	—	(D)

以上を整理すると，表 1 に示す４つの攻撃 (A)〜(D) に対して安全性を考慮する必要がある．

3 提案方式

3.1 提案方式のアウトライン

キャンセラブル方式では生体情報を乱数によってマスクしテンプレートとするが，提案方式ではマスクとしてコミットメント方式を使用する．コミットメントは，以下の性質を持つ関数 $E(\cdot,\cdot)$ である．

- コミットメント $E(x,r)$ は，コミットされる値 x と乱数 r から一意に計算される．
- （Blind）$E(x,r)$ から x を求めることは困難．
- （Bind）任意の $x, x'(\neq x)$ に対して，$E(x,r) = E(x',r')$ を満たす (r,r') を求めることは困難．

生体情報の登録時には，ユーザは生体情報のコミットメントを計算し，これをテンプレートとしてサーバに登録する．Blind の性質より，テンプレートからは生体情報が漏れることはない．

リモート認証の際には，ユーザは取得した生体情報のコミットメントをサーバに送信する．ユーザが本人である場合，２つのコミットメントにコミットされている２つの値は「近い」．そこでユーザは,「コミットされている２つの値が十分に近いこと」を証明する零知識証明プロトコルを実行する．

零知識証明プロトコルは検証者（この場合はサーバ）には証明される命題（この場合は「コミットされている２つの値が十分に近いこと」）以外の情報を与えないため，サーバは認証プロトコルからは生体情報について何も得られない．

3.2 対象とする生体情報

本研究では，生体情報を $X = (x_1, \ldots, x_m)$（$x_i \in \{0, \ldots, d-1\}$）で表わすことができ，登録時の生体情報 X と認証時の生体情報 $X' = (x'_1, \ldots, x'_m)$ のユーク

リッド距離が閾値以下である場合に本人であると判定するような生体情報を取り扱う．つまり，

$$|X - X'|^2 = (x_1 - x'_1)^2 + \ldots + (x_m - x'_m)^2 \leq \theta^2 \quad (1)$$

であれば本人であると判定する生体情報を対象とする．

生体認証は登録された生体情報と認証時の生体情報の「近さ」を測るものであるため，2 つの生体情報の距離を適切に定義することにより，基本的には任意の生体情報に対して，このような判定式によって本人性の判定が可能であると考えられる．

3.3 コミットメント方式

誰も素因数を知らない十分大きな合成数 n，Z_n^* の要素である g, h を用意する．ユーザは h の離散対数 $\log_g h$ を知らないとする．このとき，$E(x, r) = g^x h^r \bmod n$ を整数 x のコミットメントとする [6]．ただし，r は $[-2^s n + 1, 2^s n - 1]$ から選ばれた乱数，s はセキュリティパラメータ（例えば $s = 40$）である，

このコミットメントは準同型性

$$E(x_1, r_1) \times E(x_2, r_2) = E(x_1 + x_2, r_1 + r_2)$$

を満たす．なお，Z_n^* の位数は誰にも分らないので，$x_1 + x_2$ などは mod 演算ではない．

3.4 方式の詳細

最終目標としては式（1）によって本人判定をする方式の構築をめざすが，以下では，簡易的にチェック式を

$$\forall i \in \{1, \ldots, m\} : x_i - x'_i \in \{0, \pm 1, \ldots, \pm\theta\} \quad (2)$$

とする認証方式を提案する．

登録: ユーザは $(x_1, \ldots, x_m)$ から各 x_i のコミットメント

$$E_i = E(x_i, r_i) = g^{x_i} h^{r_i} \bmod n$$

を計算する．$E_1, \ldots, E_m$ をサーバに登録する．ユーザは $r_1, \ldots, r_m$ を IC カードに保管する．

認証プロトコル: ユーザは $(x'_1, \ldots, x'_m)$ から各 x'_i のコミットメント

$$E'_i = E(x'_i, r'_i) = g^{x'_i} h^{r'_i} \bmod n$$

を計算しサーバに送る．

全ての $i \in \{1, \ldots, m\}$ について下記を行う．ユーザは，全ての $\tilde{x}_i \in \{x'_i - \theta, \ldots, x'_i + \theta\}$ について，$E_i = E(\tilde{x}_i, r_i)$ であると仮定して，E_i / E'_i が $|\tilde{x}_i - x'_i| \leq \theta$ を満たす $(\tilde{x}_i - x'_i)$ のコミットメントであることを証明する区間の ZKIP[7, 8] を実行する．このとき，各 i に対して独立に $2\theta + 1$ 回の ZKIP を同時に行うが，$\tilde{x}_i$ の順はランダムに入れ替えて行う．（具体的なプロトコルについては，付録を参照.）サーバは，各 i に対して，$2\theta + 1$ 回の ZKIP のうち少なくとも 1 つで accept であれば，accept とする．

4 提案方式の評価

4.1 安全性

前章で提案した方式は，2.2 章で述べた安全性の条件を満たす．

攻撃 (A): サーバは登録時の生体情報 X のコミットメントと，認証プロトコルにおける通信を見ることができ，認証プロトコルは認証時の生体情報 X' のコミットメントと ZKIP からなる．したがって，サーバは生体情報に関する情報は何も得られない．

攻撃 (B): IC カードには，コミットメントに用いた乱数のみが保管されている．したがって，ここからは生体情報に関する情報は何も得られない．

攻撃 (C): ユーザの IC カードを持つ攻撃者がなりすましできる確率を考える．

E'_i でコミットされた値 x'_i が，登録時にコミットされた値 x_i に対して $|x'_i - x_i| \leq \theta$ を満たさないとき，区間の ZKIP では必ず reject となる．したがって，$2\theta + 1$ 回の ZKIP のうち少なくとも 1 回 accept とするためには，攻撃者は E'_i として $|x'_i - x_i| \leq \theta$ を成立させる x'_i のコミットメントをサーバに送る必要がある．x_i が Z_d 上に一様ランダムに分布していると仮定すると，x_i を全く知らない攻撃者が選んだ x'_i が $|x'_i - x_i| \leq \theta$ を満たす確率は $(2\theta + 1)/d$．したがって，全ての $i \in \{1, \ldots, m\}$ に対して少なくとも 1 回 accept される確率は $((2\theta + 1)/d)^m$ となり，十分大きい d と m に対してこの確率は小さい．

攻撃 (D): 認証プロトコルでは，X' のコミットメント及び ZKIP のみが通信され，盗聴者は何の情報も得ることができない．この盗聴者がユーザになりすますことのできる確率は，ZKIP の soundness 確率を ϵ とすると $((2\theta + 1)\epsilon)^m$ であり，十分小さい．（r_i を知らない攻撃者は，たとえ x_i を知っていたとしても ZKIP で accept されることはない.）

4.2 効率

区間の ZKIP は $O(1)$ で行うことができるため，認証プロトコル全体の通信量と計算量は $O(m\theta)$ となる．

5 まとめ

本稿では，式 (2) によって本人性を判定する生体認証システムを提案した．x_i と x'_i の線形式による判定を追加することにより，より式 (1) に近い判定を行うことも可能である．

今後の課題は，通信量・計算量の削減と，効率を下げることなく式 (1) によって本人性を判定する方式の構築が挙げられる．

参考文献

[1] N.K.Ratha, J.H.Connell, and R.M.Bolle, "Enhancing security and privacy in biometrics-based authentication system," IBM System Journal, Vol.40, No.3 (2001)

[2] 太田陽基，清本晋作，田中俊昭「虹彩コードを秘匿する虹彩認証方式の提案」情報処理学会論文誌, Vol.45，No.8，pp, 1845–1855 (2004)

[3] 高橋健太，三村昌弘,「キャンセラブル指紋照合方式の提案」，コンピュータセキュリティシンポジウム 2005 論文集，pp.379–384 (2005)

[4] 比良田真史，高橋健太，三村昌弘,「画像マッチングに基づく生体認証に適用可能なキャンセラブルバイオメトリクスの提案」2006-CSEC-34, pp. 45–440 (2006)

[5] Hill C.J., "Risk of masquerade arising from the storage of biometrics", Bachelor thesis, Dept. of CS, Australian National University (2002)

[6] E. Fujisaki, T. Okamoto, "Statistical Zero Knowledge Protocols ro Prove Modular Polynomial Relations," Prpc. of CRYPTO '99, LNCS vol. 1666, pp. 413–430 (1999)

[7] F. Boudot, "Efficient Proofs that a Committed Number Lies in an Interval," Eurocrypt 2000, LNCS vol. 807, pp. 431–444 (2000)

[8] A. Chan, Y. Frankel, Y. Tsiounis, "Easy Come – Easy Go Divisible Cash," Proc. of Eurocrypt '87, LNCS vol. 304, pp. 127–141 (1998)

A 区間の ZKIP

ここでは，提案方式の認証プロトコルで使用する区間の ZKIP の詳細を示す [7, 8].

前提：サーバは $E/E' = E(x,r)/E(x',r')$ を持っている．ユーザは r, r', x' および x の予測値 $\tilde{x}$ を知っている．

証明したい事柄：E/E' が $|\tilde{x} - x'| \leq \theta$ を満たす $\tilde{x} - x'$ のコミットメントであること．

パラメータ：τ, ℓ:セキュリティパラメータ．正当なユーザが reject される確率が $1/2^{\ell}$, 不正なユーザが accept される確率が $1/2^{\tau-1}$ となるため，共に十分大きい必要がある．また，$T = 2(\tau + \ell + 1) + \lfloor \log 2\theta \rfloor + 1$ とする．

なお，全ての式で $\pm$ は復号同順とする．

Step 1: ユーザは

$$\tilde{y}_{\pm} = 2^T(\theta \pm (\tilde{x} - x')), \quad \tilde{y}_{1\pm} = \lfloor \sqrt{\tilde{y}_{\pm}} \rfloor$$

とし，サーバと次のプロトコルを行う．

Step 1.1: ユーザは $r_{1\pm}, r'_{1\pm}$ をランダムに選び,

$$F_{1\pm} = E(\tilde{y}_{1\pm}^2, r_{1\pm}), \quad G_{1\pm} = E(\tilde{y}_{1\pm}, r'_{1\pm})$$

を計算する．さらに $w_{1\pm}, t_{1\pm}, t'_{1\pm}$ をランダムに選び,

$$H_{1\pm} = G_{1\pm}^{w_{1\pm}} h^{t_{1\pm}}, \quad H'_{1\pm} = E(w_{1\pm}, t'_{1\pm})$$

を計算する．$F_{1+}, F_{1-}, G_{1+}, G_{1-}, H_{1+}, H_{1-}, H'_{1+}, H'_{1-}$ をサーバに送る．

Step 1.2: サーバは τ ビットの乱数 c_1 をユーザに送る．

Step 1.3: ユーザは

$$\begin{aligned} u_{1\pm} &= t_{1\pm} + c_1(r_{1\pm} - r'_{1\pm}\tilde{y}_{1\pm}), \\ u'_{1\pm} &= t'_{1\pm} + c_1 r'_{1\pm}, \\ z_{1\pm} &= w_{1\pm} + c_1 \tilde{y}_{1\pm} \end{aligned}$$

を計算し $u_{1+}, u_{1-}, u'_{1+}, u'_{1-}, z_{1+}, z_{1-}$ をサーバに送る．

Step 1.4: サーバは

$$\left.\begin{array}{l} G_{1\pm}^{z_{1\pm}} h^{u_{1\pm}} \equiv H_{1\pm} F_{1\pm}^{c_1} \pmod{n} \\ g^{z_{1\pm}} h^{u'_{1\pm}} \equiv H'_{1\pm} G_{1\pm}^{c_1} \pmod{n} \end{array}\right\} \quad (3)$$

が成り立つかチェックする．

Step 2: 次に，ユーザは

$$\tilde{y}_{2\pm} = \tilde{y}_{\pm} - \tilde{y}_{1\pm}^2$$

を計算し，サーバは

$$F_{2\pm} = g^{2^T\theta}(E/E')^{\pm 2^T}/F_{1\pm}$$

を計算し，次のプロトコルを実行する．

Step 2.1: ユーザは $w_{2\pm}, t_{2\pm}$ をランダムに選び,

$$H_{2\pm} = E(w_{2\pm}, t_{2\pm})$$

を計算し，H_{2+}, H_{2-} をサーバに送る．

Step 2.2: サーバは τ ビットの乱数 c_2 をユーザに送る．

Step 2.3: ユーザは

$$\begin{aligned} u_{2\pm} &= t_{2\pm} + c_2(2^T(r - r') - r_{1\pm}), \\ z_{2\pm} &= w_{2\pm} + c_2 \tilde{y}_{2\pm} \end{aligned}$$

を計算し $u_{2+}, u_{2-}, z_{2+}, z_{2-}$ をサーバに送る．

Step 2.4: サーバは

$$\left.\begin{array}{l} g^{z_{2\pm}} h^{u_{2\pm}} \equiv H_{2\pm} F_{2\pm}^{c_2} \pmod{n} \\ z_{2\pm} \in [c_2 2\sqrt{2^{T+1}\theta}, 2^{\tau+\ell+1}\sqrt{2^{T+1}\theta}] \end{array}\right\} \quad (4)$$

が成り立つかチェックする．

Step 3: サーバは，チェック式 (3) および (4) が共に成り立つときのみ，accept とする．