耐タンパ個人端末を利用し個人情報の保護を可能とした認証方式

佐藤直之† 鈴木英明†

本稿では，耐タンパ性を持つセキュアな個人端末と特殊な電子証明書を用いた新たな認証方式を提案する．ここで提案する方式は，プライバシ保護を重視し，個人情報の不必要な漏洩をもたらさない．『不必要』とは，認証で権利を主張するユーザにとって，価値がないという意味である．既存の多くの認証システムでは，認証によって，端末の前にいるユーザがあらかじめ登録されたどのユーザであるかを確認し，そのユーザについての登録情報より権利を確認する．この手法では，すでに不必要な情報漏洩が起こっている．検証者側がユーザがだれであるか知ること自体が，本来ユーザにとって不必要である．これに対して本方式では，認証はユーザと権利とを直接結びつける．認証によって確認されるのは，端末の前にいるユーザが，権利を持っているか否かという情報だけである．本方式では，裁判官と名付けた１つの特殊機関を設け，これによって，上記の個人情報の保護の機能を実現しつつ，不正を試みる者への容易な対応を実現している．

An Authentication System that Protects Privacy by Using Tamper-resistant User Terminals

Naoyuki Sato† and Hideaki Suzuki†

We have developed an authentication system that uses tamper-resistant user terminals and certificates to confirm a user's rights. In our system, a verifier cannot acquire any user's personal information (e.g., name or address) during the authentication process, without the user's consent. Users can choose what personal information will be acquired by verifiers and what will not. Today, almost all systems identify a user in order to authenticate the user's rights. But in many cases, it is not necessary for a system to know who the user is. There are many privacy violations. We define a model that includes a 'Judge' as a special authority, and use cryptographic techniques to achieve our goal.

1. まえがき

実社会においても計算機の世界においても，個人が，確かにその主張する権利を持つ存在であることを確認する認証作業は，多くの場面において実施されている．最近では，Web サイトにおいても，簡単なパスワードや電子証明書を用いた認証作業が頻繁に行われるようになった．

これらの認証の目的を考えてみると，その内容は認証後に提供されるサービスの内容に応じて大きく異なる．たとえば，銀行の ATM では，ユーザごとに異なる預金を処理するため，またプライバシの侵害を引き起こさないためにも，端末の前にいるユーザを特定する必要がある．これに対して，自動車の運転免許証や定額制のインターネットプロバイダなどのように，ユーザが万人に共通した内容の権利を行使する場面においては，認証によって個人が特定される必要はなく，権利の確認さえできればよい．

だが現状では，上記の２番目のような場面においても，個人を特定できる情報を用いて，認証が行われることが多い．個人識別子とパスワードを利用した手法や，名前などの個人情報を記載した証明書を用いる手法が一般的に利用されている．この現実は，プライバシ保護の視点から考えると，大きな問題である．ユーザにとって，このように本来不必要な情報が流出することはけっして歓迎できることではない．近年，インターネットなどの公共広域電子ネットワークが急速に普及し，情報の収集および配布をだれでも簡単に行えるようになった．このため，個人情報が不必要に漏洩することは，個人の社会生活に甚大な影響を与える可能性もある．個人情報の不必要な流出を起こさない認証方式が求められている．

このような背景の中，本稿では，耐タンパ性を持つセキュアな個人端末の存在を前提として，上記の課題を解決する新たな認証方式を提案する．本方式では，利用者は匿名のまま権利を主張しこれを行使でき，またサービス提供者は匿名者であってもその不正行為の

† NTT 情報流通プラットフォーム研究所
NTT Information Sharing Platform Laboratories

取り締まりが容易である．耐タンパ性端末には，ICカードなどのほか，特殊な携帯電話やPDAなどを利用できる．最近の目覚ましい技術の進歩によって，本稿で求めるような機能を持つ媒体の実現は可能である．

提案する方式では，ユーザの権利を証明する道具として，各種発行局から事前に発行される特殊な電子的な証明書を利用する．運転免許証やパスポートなどは，現実世界での証明書の例である．以下では，便宜上，個人情報をその内容から『特徴情報』と『行動情報』の2つに明確に区別する．

特徴情報： 名前，住所，身長，体重，容姿，収入などの社会的あるいは身体的特徴についての情報．

行動情報： 行動についての情報．行動履歴．

提案方式では，証明書にユーザの特徴情報を必要以上に掲載せず，権利についての情報と他の個人情報との管理を明確に区分する．通常，証明書からはその所有者の名前などの特徴情報を導けない．また提案方式では，耐タンパ性を持つ端末の機能を利用して，コピーなどによって流出した証明書が不正に利用されるのを防ぐと同時に，証明書の発行局でさえユーザの行動を監視できないようにしている．

また，本提案方式のように証明書で権利を証明する手法には，以下の2点のような長所もある．1）認証の検証側であるサービス提供者は，ユーザのパスワードを確認するためのデータベースを準備する必要がない．2）サービス提供者は，自分の管理する各々の資源に対して，その利用者の範囲を，様々な権利の組合せとして柔軟に指定できる．

本稿では，これまで述べてきた個人情報の保護の必要性に加えて，ユーザの利便性およびシステムの安全性などの多くの課題を考慮し，以下の要求を設定した．本稿では，これらを満たす認証方式を提示する．

（1）認証時の特徴情報の流出制御
認証実施時において，どのような特徴情報が検証者側に伝わるかを，ユーザ自身が制御できる．

（2）行動の追跡不能性とその制御
行動が把握されうる範囲をユーザが制御できる．

（3）証明書の再利用性
証明書は何度でも利用できる．

（4）証明書の利用者制限
証明書はその所有者以外は利用できない．

（5）証明書の無効化
発行局は任意の証明書を無効にできる．

（6）ユーザの特定
（不正発覚時などに）ユーザを特定する方法がある．

（7）ユーザの無力化
任意のユーザに発行されたすべての証明書を同時に無効にする方法がある．

次章では，関連研究を概観する．3章では，本稿で提案する認証方式の基本的な設計方針と，その基盤となる技術を説明する．4章では提案するプロトコルを説明する．5章で安全性などの特徴について検討し，6章でまとめを行う．

2. 関連研究

認証処理における匿名性の実現方法は，電子現金の分野において活発に議論され，採り入れられてきた（文献1），2），4），16）など）．特に，Brandsは，本稿で提案する方法と同様に，耐タンパ性を持ったユーザ端末を前提とした電子現金方式を提案している[1),2)]．この分野においては，電子現金（電子チケットを含む）が重複して利用された場合にのみその利用者が判別可能となるような手法を提案し，これによって匿名性と不正対策とを両立しているものが多い．Brandsもこの手法を採用している．だが，この手法は，電子現金のように，あらかじめ利用回数や利用可能額などのなんらかの利用制約が決定している場合にのみ採用できる方法である．これに対して，本稿で提案する方式は一般的な認証場面への適用を目的としているので，証明書に上記のような利用制約を設けていない．このため上記の手法を採用することは難しい．

本稿では，ユーザの匿名性を満たしかつ匿名ユーザが行った不正行為への対応を可能とするために，『裁判官』と名付けた概念上新しい特殊な存在をシステムの構成要素として導入する方式を提案し採用している．次章以降で詳しく説明するが，提案方式では，不正発覚時などに，証明書を利用したユーザと，そのユーザの個人情報とを結びつける役割を，他のシステム構成要素から独立した裁判官の役割としている．本方式では裁判官の力はきわめて大きい．裁判官は，証明書からそのユーザを識別し，またそのユーザに対してそれまでに発行されたすべての証明書を同時に無効にすることができる．提案方式では，裁判官がその役割を果たせるようにするために，いくつかの条件を満たした確率的暗号系と確率的性質を持った特殊な符号系などの技術を組み合わせて利用している．このアプローチは，Brandsの方式をはじめ，既存の電子現金方式のアプローチとは大きく異なったものである．

SDSI（Simple Distributed Security Infrastructure）[13)]やSPKI（Simple Public-Key Infrastructure）[7),8)]では，本稿で提案する方式と同様に，ユーザの公開鍵を認証時に識別子として利用する方法につ

いて提案されている．ただし，これらでは個人情報の保護の必要性を重視しておらず，その機能がない．文献 15) では，証明書を用いて匿名性を得る 1 つの方式が提案されている．ただしこの方式では，証明書の発行局に対する匿名性が考慮されていない．

3. 基本方針と数学的準備

3.1 基本方針と基盤技術

本稿では，5 種類の存在から成る世界モデルを想定して利用する（**図 1** 参照）．証明書発行局（**Certificate Issuer** or **CI**）はユーザ（**User**）の権利を保証する証明書を発行する．ユーザは，各々別々のカード（**Card**）を保持する．またユーザは自分の証明書を適切に管理し資源やサービスを利用する．資源管理者（**Resource Manager** or **RM**）は資源やサービスをユーザに提供する存在で，認証における検証者となる．裁判官（**Judge**）は調停役であり，主に不正に対する防衛のために存在する．裁判官は，他の 3 者が信頼する特別な存在である．発行局，資源管理者，ユーザは各々複数存在することを想定している．

カードは秘密情報を記録する記憶能力と，これを利用する計算能力を持つ．ここに記録された情報は，カードの保持者であるユーザ自身ですら自由に知ることができない．また，カードは決められた処理を必ず正しく行うものとする．すなわち，カードは，だれにとっても悪意を持つ存在となることはない．カードは耐タンパ性を持ち，秘密情報が洩れることや，アルゴリズムを狂わされることはない．また，カードはユーザの所有物であるので，カードに対する通信はすべてユーザの中継の下で行われることを前提とする（**図 2** 参照）．

本稿で提案する方式は，以下の基本的な方針に従って動作する．

(1) カードは鍵ペア (S, P) を複数個（増減可）持つ．
(2) カードは秘密鍵 S をだれにも教えない．
(3) 発行局は，証明書を 1 つの公開鍵 P と関連づけて発行する．証明書を利用するには，P に対応した秘密鍵 S，すなわちカードが必要となる．
(4) 証明書は，その所有者についての情報を必要以上に持たない．通常，証明書からは，そのユーザ名などの特徴情報を導けない．
(5) ユーザ自身が証明書および鍵を選択し利用する．
(6) 同一ユーザの持つ証明書は，すべて同一の識別情報を含む．ただし，特殊な鍵を持つ裁判官しかこの情報を読み取れない．
(7) 同一ユーザの持つ証明書は，すべて同一の識別情報に反応する性質を持つ．したがって，この鍵となる識別情報が既知であれば，だれでもそのユーザに対して発行された証明書を識別できる．

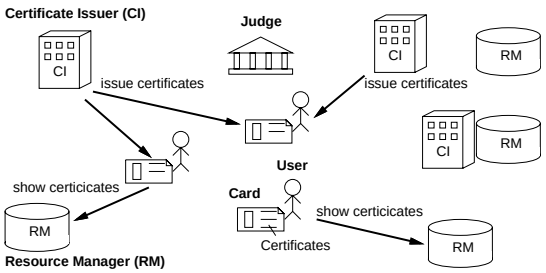

図 1 世界モデル
Fig. 1 World model.

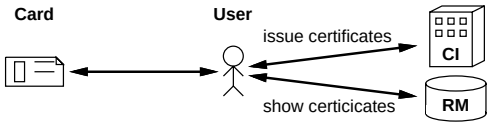

図 2 通信のモデル
Fig. 2 Communication model.

上記の (2)，(3) は，証明書をその所有者以外が不正に利用できなくするために設定した．(1)，(4) および (5) は個人情報保護のための方針である．(6) および (7) は不正者の特定や証明書の無効化の実現のためにある．

基盤的技術として，(4) を実現するためにブラインド署名技術[3]を，(6) のためにいくつかの条件を満たした確率的暗号系[9]を，(7) のために特殊な符号系を定義して利用する．他の項目については，証明書の発行や検証のプロトコル自体で実現する．

以下には，準備として，ブラインド署名の表記方法と確率的暗号技術，符号技術について述べる．

3.2 ブラインド署名

以下に，ブラインド署名に関係する 4 つの関数を定義する．上から 2 番目の関数は署名者が実行し，最後の関数は署名の検証者が実行する．残る 2 つは署名を要求するユーザが実行する．S は署名生成鍵（署名者の秘密鍵），P は署名検証鍵（同公開鍵），m は署名を行う文書，s は m に対する署名情報である．

- ブラインド関数 $m' = Blind(P, m, r)$
  署名する文書 m を隠す関数．r は乱数．関数 $Blind$ は r および m の変化に対して一方向性を持つとする☆．

☆ 関数 $Blind'$ がこの条件を満たさない場合，適当な一方向性関数 f と g を導入し，$Blind(P, m, r) = Blind'(P, f(m), g(r))$ とする．

- 署名関数 $s' = Sign(S, m')$
  ユーザの示した文書 m' に対して署名を行う．
- ブラインド削除関数 $s = Unblind(P, s', r)$
  署名 s を導出．r は関数 $Blind$ 実行時の選択値．
- 検証関数 $Verify(P, m, s) = 0$ or 1
  公開鍵 P を用いて文書 m の署名 s を検証する．

$$Verify(P, m, s) = \begin{cases} 1 & (s \text{ が正しい場合}) \\ 0 & (\text{上記以外の場合}) \end{cases}$$

$Verify(P, m, Unblind(P, Sign(S, Blind(P, m, r)), r)) = 1$ である．ブラインド署名の実現法はいくつか提案されているが，RSA 署名[3),12)]を利用した方法が有名である．

3.3 確率的暗号系

本提案方式では，発行局を問わず，同一ユーザに対して発行する証明書には，すべて同じ識別情報を埋め込む．この情報は不正発覚時などに，ユーザを特定するために利用する．この目的で，次の条件を満たす意味的安全（強秘匿）な確率的暗号系を利用する[9)]．また，この関数を以下のように記述する．

［条件］任意の暗号文について，これと同じ平文を持つ別の暗号文を公開情報のみから容易に生成できる．またこの際，同一平文を持つすべての暗号文の生成が可能とする．

- 暗号生成関数 $c = E_{crypt}(e, m, r)$
  公開鍵 e を用いて平文 m の暗号文 c を作成する．確率的暗号系では，1 つの平文の暗号文は多数存在するが，r によって一意に決定する．
- 復号関数 $m = D_{crypt}(d, c)$
  暗号文 c を秘密鍵 d で復号し平文 m を取り出す．
- 暗号変換関数 $c' = Cv_{crypt}(e, c, t)$
  公開鍵 e と暗号文 c より，c と同じ平文を持つ別の暗号文 c' を生成する．t は c' を一意に決定する．

秘密鍵を知らなければ，意味的安全性より (c, c') と $(c, E_{crypt}(e, m', r'))$ $(m \neq m')$ の区別は容易でない．上記の条件を満たす暗号系としては，ElGamal 暗号[6),14)]，Okamoto らの暗号[11)]などがある．

3.4 特殊な符号系

本提案方式では，あるユーザに対して発行されたすべての証明書を同時に無効にできるようにするために，以下の 4 つの特徴を持つ特殊な符号系を利用する．

特徴 1： 1 つの元に対して複数の符号がある．

特徴 2： 符号からは，その元の情報をほとんど得られない．特に，ある符号 c_0 に対して，2 つの符号 c_1 と c_2 があり，このどちらかが c_0 と同じ元を持っていることが分かっているとしても，c_1 と c_2 のいずれが同じ元を持つかを容易には判定できない．

特徴 3： 符号に対して，ある値が元であるかどうかは容易に判別できる．

特徴 4： 符号に対して，その元を知らなくても，その符号と同じ元を持つ別の符号を容易に生成できる．またこの際，同一の元を持つすべての符号の生成が可能とする．

上記の特徴 1 で定義される符号生成関数を $E_{code}(x, r)$ とおく．$c = E_{code}(x, r)$ は，元 x の符号 c を作成する．r は c を一意に決定するための情報である．特徴 2 は，暗号系における強秘匿性と対応する性質である．特徴 3 で定義される判定関数は $D_{code}(x, c)$ と記す．$c = E_{code}(x', r)$ の場合以下のようになる．

$$D_{code}(x, c) = \begin{cases} 1 & (x = x' \text{ の場合}) \\ 0 & (\text{上記以外の場合}) \end{cases}$$

また，特徴 4 で定義される符号変換関数を $Cv_{code}(c, t)$ とおく．t は変換を一意に指定する．

$D_{code}(x, E_{code}(x, r)) = 1$ である．上記の特徴より，元 x を知らなければ，2 つの符号 c と c' が同一の元を持つかどうか容易には分からない．この関数の具体例としては，Diffie-Hellman 決定問題の困難性[5)]に根拠をおいた方法などがある☆．

4. 提 案 方 式

4.1 各構成要素の役割と初期状態

以降において，e は公開鍵を，d は秘密鍵を示し，P は署名検証鍵（公開鍵），S は署名生成鍵（秘密鍵）を示す．また，“$_{CI}$” は証明書発行局に関係するデータに，“$_J$” は裁判官に関係するデータに付与している．

証明書発行局 CI： 証明書に対してブラインド署名を行う．各 CI は署名で利用する鍵ペア (P_{CI}, S_{CI}) を持ち，公開鍵 P_{CI} を公開している．また，安全パラメータ N を決定し公開している．

資源管理者 RM： RM は認証時の検証者となる．

ユーザ： ユーザは自分の証明書を管理して利用する．ユーザは初期化時にカードと 1 枚の証明書を受け

☆ p を素数，$1 < x < p-1$，t を $p-1$ と素な乱数とする．
$E_{code}(x, r) = (a, a^x)$（$r$ を用いて原始根 a を生成）
$D_{code}(x, (c_L, c_R)) = 1$（$c_L^x = c_R$ のとき）or 0（左記以外）
$Cv_{code}((c_L, c_R), t) = (c_L^t, c_R^t)$.
注意：計算はすべて $mod\ p$ で行う．c_L^t は原始根となる．

取る☆．この初めの証明書は，カードが正しく動作することを他者に証明するためにある．この証明書にはユーザの持つカードに対応した ID が埋め込まれている．ただし，ID は裁判官の公開鍵で暗号化されており，裁判官以外は読み取れない（後述参照）．

カード： カードは，署名を行うために鍵ペア (P, S) を複数セット持つ．この署名の具体的手法については，任意の手法を採用してよい．鍵ペアの数は所有者のユーザの意志で自由に増減することができる．ただし，ユーザは新しい鍵ペアを指定することはできず，カード自身がこれを作成する．カードは自分の持つ秘密鍵をだれにも知らせない．

裁判官： 裁判官は 3.3 節で定義した暗号系で使用するための鍵ペア (e_J, d_J) を持ち，公開鍵 e_J を公開している．また，『無効 ID リスト』と『無効証明書リスト』（ともに初めは空）を持ち，公開している．

4.2 証明書の定義

本方式での証明書は 6 つの要素から成る．

i. 署名 s：証明書発行局の発行した署名．s は紙の証明書における『印』に対応する．

ii. 公開鍵 P：P は証明書の利用者を制限する．この鍵と対応した秘密鍵を持った者のみが，この証明書を利用できる．P は紙の証明書での『顔写真』などと対応する．ユーザは 1 つの鍵を複数の証明書の利用条件として利用してもよい．

iii. 証明事項 M：証明書が保証する内容．紙の証明書での文書部分と同じ．

iv. 暗号文 E：識別子 ID の裁判官の鍵 e_J による暗号文．$E = E_{crypt}(e_J, ID, r)$.

v. 符号 D：識別子 ID の符号．$D = E_{code}(ID, r')$.

vi. 補足情報 c：詳細は下記プロトコルを参照のこと．

4.3 発行プロトコル

ユーザが新しい証明書 $C = (s, P, M, E, D, c)$ を獲得する際のプロトコルを示す．プロトコル開始前に，ユーザと発行局 CI の間で，証明事項 M に合意しているとする．プロトコル中でユーザが利用する証明書 $C_o = (s_o, P_o, M_o, E_o, D_o, c_o)$ は，ユーザが以前に獲得した証明書の 1 枚で，ユーザの持つカードが正しく動作することを保証するため，また新しく発行する証明書にもカードの ID を埋め込むためにある．これは，初期化時に与えられた証明書でもよい．関数 g は適当な一方向性ハッシュ関数とし，また "||" は結合を表す．

☆ この証明書はカードのメーカなどが発行する．

（1）ユーザは，鍵 P および P_o に対するカード内での識別番号と $(M, E_o, D_o, P_{CI}, N, e_J)$ をカードに送る．

（2）カードは乱数 $(r_k, u_k, v_k, w_k, x_k)$ $(1 \le k \le N)$ を適当に選び，以下の値 (α_k, β_k) を計算する．

$$\alpha_k = Blind(P_{CI}, Z_k, r_k) \quad (1)$$

$$\beta_k = E_{crypt}(e_J, r_k||Z_k, x_k) \quad (2)$$

ただし，

$$X_k = Cv_{crypt}(e_J, E_o, u_k) \quad (3)$$

$$Y_k = Cv_{code}(D_o, v_k) \quad (4)$$

$$Z_k = g(P||w_k)||X_k||Y_k||M. \quad (5)$$

カードは，公開鍵 P_o に対応する秘密鍵 S_o を用いて，式 (6) の文に対する署名 s_{Card} を作成し，s_{Card} とすべての $(\alpha_k, \beta_k, r_k, u_k, v_k, w_k, x_k)$ をユーザに返す．

$$\alpha_1||\alpha_2||...||\alpha_N||\beta_1||\beta_2||...||\beta_N||E_o||D_o||M \quad (6)$$

（3）ユーザはすべての (α_k, β_k) と s_{Card} および C_o を CI に送る．

（4）CI は，裁判官の公開する 2 つのリストを用いて，C_o が無効となっていないことを確認する（詳細は 4.5 節の無効化処理を参照）．CI は，C_o の発行局の公開鍵 P_{CI_o} を用いて，以下の式が成り立つことを確認する．また，CI は署名 s_{Card} も確認する．

$$Verify(P_{CI_o}, g(P_o||c_o)||E_o||D_o||M_o, s_o) = 1$$

すべてに合格した場合，CI は k' $(1 \le k' \le N)$ を選びユーザに送る．

（5）ユーザは，$k \neq k'$ となるすべての k に対して，$(r_k, u_k, v_k, x_k, g(P||w_k))$ を CI に送る．

（6）CI は，手順（5）で受け取った情報を用いて式 (1)〜式 (5) に従って α_k と β_k を計算し，これらが手順（3）での値と一致することを確認する．

これに合格した場合，CI は以下の値 s' を計算し，これをユーザに返す．また，CI は $(s', E_o, \alpha_{k'}, \beta_{k'})$ を保存する．

$$s' = Sign(S_{CI}, \alpha_{k'}) \quad (7)$$

（7）ユーザは，新しい証明書 $C = (s, P, M, E, D, c)$ を以下のように計算する．

- $s = Unblind(P_{CI}, s', r_{k'})$
- $E = X_{k'}$
- $D = Y_{k'}$
- $c = w_{k'}$

4.4 検証プロトコル

ユーザは以下のようにして証明書を利用する．

（1）ユーザは証明書 C を資源管理者 RM に送る．

（2）RM は C について 3 つの検証を行う．

- 以下の式が成り立つことを確認する．
$$Verify(P_{CI}, g(P||c)||E||D||M, s) = 1 \quad (8)$$
この検証の合格は，『公開鍵 P と対応した秘密鍵を持った存在について，CI が事項 M を保証している』ことを意味する．
- 既存の公開鍵署名を利用した認証手順を用いて，ユーザが公開鍵 P に対応した秘密鍵 S を持ったカードを保持していることを確認する．
- 裁判官の公開する 2 つのリストを用いて C の有効性を調べる．この詳細は次節で説明する．

4.5 取消し処理とユーザの特定

証明書を無効にする方法は 3 通りある．これらを以下に示す．この 3 番目の手法では，ユーザを特定することができる．これらの手法によって，1 章であげた 7 つの要件のうち，「(5) 証明書の無効化」「(6) ユーザの特定」「(7) ユーザの無力化」の 3 つが満たされる．

4.5.1 証明書の無効化 (1)

発行時点において，証明事項 M 内に有効期限などの条件を明記する．この条件を満たさなくなった場合，証明書は無効と考える．条件は，期限などのように容易に判別がつくものであることが望まれる．

4.5.2 証明書の無効化 (2)

無効としたい証明書が既知であるなら，その証明書に含まれる署名 s を『無効証明書リスト』に登録する．このリストに登録された s が含まれる証明書は無効と解釈する．

ただし，CI はブラインド署名の技術を用いて署名を作成しているので，通常自分が発行したどの証明書をどのユーザが持つか分からない．この場合，CI は，無効としたい証明書に対応した $(s', E_o, \alpha_{k'}, \beta_{k'})$ を裁判官に送る．裁判官は，秘密鍵 d_J を用いて，$\beta_{k'}$ から $(r_{k'}, Z_{k'})$ を取り出す．裁判官は以下の式 (9) を検証し，これが成立するなら署名 s を $s = Unblind(P_{CI}, s', r_{k'})$ として計算し，『無効証明書リスト』に登録する．

$$\alpha_{k'} = Blind(P_{CI}, Z_{k'}, r_{k'}) \quad (9)$$

式 (9) が成立しない場合，保存されていた $\beta_{k'}$ に偽りの情報が含まれていたことになる．この場合，E_o を用いて，次項で説明するようにユーザの無力化などの処置を行うことができる．

4.5.3 ユーザの特定と無力化

裁判官は，証明書自体から，その所有者を識別し，このユーザの持つすべての証明書を同時に無効にすることができる．裁判官は，証明書に含まれる情報 E $(= E_{crypt}(e_J, ID, r))$ より，秘密鍵 d_J を用いて，識別子 ID を導出する．4.1 節の前提により，ID からカードを特定でき，その所有者であるユーザを知ることができる．また，無力化のためには，裁判官は ID を『無効 ID リスト』に登録する．

ある証明書において，証明書に含まれる D $(= E_{code}(ID, r'))$ が，無効 ID リストに登録された ID の 1 つと対応している場合，すなわち以下の式が成り立つとき，この証明書は無効である．

$$\exists ID_k \text{ in 無効 } ID \text{リスト} : D_{code}(ID_k, D) = 1 \quad (10)$$

5. 検討・評価

本提案方式の特徴を (1) 証明書の偽造，(2) 証明書の不正利用，(3) 証明書の運ぶ情報，(4) カードの機能の 4 つの点から検討する．

5.1 証明書の偽造

証明書はそれぞれ 6 つの要素より構成されているが，署名 s を除く各要素は，公開情報のみを用いて，だれでも自由に作成できる．したがって，証明書の偽造は，署名の偽造とほぼ等価である．

署名の偽造の可能性は，下位で使われる具体的な署名方法に大きく依存する．このため，プロトコルで利用する署名方法を決定する場合には，署名方法自体の持つ本質的な欠点を考慮し，必要なら欠点を補うようにプロトコルを修正しなければならない．

本提案方式では，CI は，ブラインド署名の技法を用いて署名 s を作成する．したがって，もしユーザが CI と約束した内容とは異なった文書にブラインド処理を行い CI がこれに気づかないとすれば，ユーザは，署名の偽造と同じ効果を得ることができる．本方式では，ユーザによるこのような不正行為を防止するために，文書にブラインド処理を行うのをカードの仕事とし，この作業の証明として，カードしか発行できない署名 s_{Card} を用いている．また，カード自身が不正を行わないことについての保証は，『カードの正当性』を保証する証明書 C_o と，カードの動作に対する前提で説明している．この手法では，CI は，自分が署名する文書の一部を，カードの署名を通して確認していることになる．

ここで，カードの正当性を保証する証明書の利用条件である秘密鍵 S_o を入手した者は，カードの動作を真似て不正を行える可能性があることに注意を要する．通常，公開鍵署名法では，公開鍵から秘密鍵を導出することは計算量的に難しい．だが，物理的にカードを分解するなどの別の手法を用いて秘密鍵を入手することは可能かもしれない．

カードについての耐タンパ性の前提は，上記のような事態がめったに起こらないことを保証するためにある．しかし，このような事態が発生した場合，秘密鍵を入手した者は，次の証明書獲得時において，以下の5種類の情報の偽造が可能である．

- M：任意の内容を保証させる証明書を獲得
- P：任意の鍵を条件とした証明書を獲得
- E：不正ユーザの摘発を妨害
- D：ユーザ（カード）単位での無効化を妨害
- $\beta_{k'}$：証明書単位での無効化を妨害

このうち，P 以外の偽造については，証明書発行時に N を母数とした cut-and-choose の手法を用いることで防止している．P 以外の偽造が成功する確率は $1/N$ である．逆に，この偽造行為が発覚する確率は $(N-1)/N$ である．N の値は，CI が保証する M の内容に応じて適切に定めなければならない．S_o が物理的に簡単には導出できないことを仮定すれば，多くの場合，$N=1$ で十分だと予想する．この場合，発行プロトコルの cut-and-choose の部分はすべて省略することができる（詳細は後述）．

P の偽造については，プロトコル中では対応していない．しかし，P が偽造された証明書が発見された後には，その証明書に含まれる情報から，当該ユーザおよび証明書の無効化を行うことができる．

5.2 証明書の不正利用

証明書 C を利用するためには，これに含まれる公開鍵 P と対応した秘密鍵 S が必要である．したがって，S をカード以外が知らないとすれば，証明書を利用するためにはカードが必要となり，これを所有する正規のユーザしか証明書を利用できない．以上より，1 章であげた 4 番目の要件「（4）証明書の利用者制限」が満たされていることは明らかである．また，3 番目の要件「（3）証明書の再利用性」については，秘密鍵 S を持ったユーザがこの証明書を何度でも利用できることは明らかであり，満たされている．

カードの盗難対策としては，カード自身がユーザに対する認証手段を備えていることが好ましい．だが，いずれにせよ，証明書が不正に利用されたことが分かった後には，無効化の手段をとることができる．

5.3 証明書の運ぶ情報

証明書 $C=(s,P,M,E,D,c)$ は，裁判官以外のすべての存在に対して，同じ程度の個人情報しか与えないということがとても重要である．このことについて，まず，裁判官と発行局を除いた存在について考えてみる．これらの存在に対して証明書が直接に与える情報は，『CI が公開鍵 P に対応した秘密鍵を持った存在について，事項 M を保証している』という内容である．E や D，c はこの証明書の有効性を判定するために必要だが，秘密鍵 d_J を知らない限り，これらは乱数と区別がつかない．通常，同一ユーザの持つ各証明書は各々異なる (E,D,c) を持つので，この情報をユーザの追跡などの情報収集に利用することは難しい．

次に，証明書 C の発行局 CI に対しても，証明書が与える情報は上記の内容と同じである．CI は証明書発行時に目隠しをしたまま（ブラインドで）署名を行うので，同じ内容 M を保証する証明書を複数のユーザに対して発行した場合，自分が作成した署名 s（正確には s'）がどの証明書と対応するのかが分からない．関数 g の一方向性と乱数 w_k は，cut-and-choose の処理によって P が CI に知れてしまわないためにある．このため CI は P をユーザ判別のための道具として利用できない．さらに，式 (3) の X_k および式 (4) の Y_k，乱数 w_k の値は N 個の要素で各々異なり，3.3 節と 3.4 節の定義により，開示を行わない $k=k'$ の項について CI はこれらの値を知ることができない．したがって，これらをもとに作成される (E,D,c) は CI に情報を与えない．CI は $\beta_{k'}$ も知っているが，d_J を知らないのでこれを直接には利用できない．

裁判官に対して，証明書は特別な情報を与える．裁判官は，証明書自身からその所有者の持つカードの ID を導ける．また発行局からの情報があれば，特定の証明書の署名 s を得ることができる．本提案方式では，このように裁判官という特殊な存在を仮定している点が大きな特徴の 1 つである．

以上の考察より，裁判官を除いたすべての存在に対して，証明書がもたらす情報は，『CI が公開鍵 P に対応した秘密鍵を持った存在について，事項 M を保証している』という内容だけであることが分かる．この性質より，任意の CI と RM が結託しても，ユーザの特徴情報と行動情報とを結びつけることは困難となる．

ユーザは，ある資源を利用したい場合，その資源を利用するために最低必要な権利のみが記載され保証された証明書を提示して，資源を利用することができる．ユーザは証明書を同時に複数枚利用してもよい．また，各証明書は，ユーザの持つ複数の権利を保証してもよいし，個々の小さな権利を保証してもよい．たとえば，生まれた年を保証する証明書や，国籍を保証する証明書，会員制クラブの会員であることを保証する証明書などがある．権利とは，ユーザについての情報の一種である．上記の最低必要な権利とは，提供される資源およびサービスの内容に従って異なり一意に決定でき

ないが，ユーザを識別するに足る情報が含まれない場合も多い．提案方式はこのような場面に適用することを想定している．逆に，ユーザが提供したくない情報を RM が要求した場合，ユーザは，その資源を利用することを諦めるか，あるいは要求どおりの情報を提供するかどちらかを選ぶことになる．1 章の要件「(1) 認証時の特徴情報の流出制御」は，このように，ユーザが自分の持つどの権利をだれに証明するか選ぶことができるという意味において，満たされている．

1 章の 2 番目の要件「(2) 行動の追跡不能性とその制御」については若干注意が必要である．上記のように，ユーザが，自分が識別されるような情報を RM に提供しなかった場合，認証の時点でユーザが識別されることはない．しかし，ユーザが同一証明書を複数回利用した場合，RM や CI は，ユーザがだれであるかは識別できないとしても，証明書をインデックスとして，『特定の個人』としてユーザの行動を監視できる．実際には，RM や CI からは，証明書に記されたユーザの公開鍵 P がそれぞれ特定の個人を示しているように見える．ユーザは，鍵ペアごとに，異なった自分の分身がいることを認識していなければならない．もし，ユーザが特定の個人として監視されることをも回避したいのなら，同一の権利を保証する複数枚の証明書を各々別々の鍵ペアを用いて作成し，これらを適宜切り替えて利用する必要がある．逆に，このようにすることによって，ユーザは，自分の行動が RM や CI にどのように把握されるのかを制御できる．

以上のすべての考察より，本提案方式が 1 章でのすべての要求を満たしていることが分かる．

5.4 カードの機能と役割について

本提案方式でのカードの機能についての要求は，3.1 節に示した．カードは秘密情報を記録する記憶能力と，これを利用する計算能力とを持つ．また，カードは耐タンパ性を持ち，秘密情報が洩れることや，アルゴリズムを狂わされることはない．

このようなカードへの要求は，所有者以外の存在に秘密鍵が盗み出されることを防ぐ目的のためだけに設定されているのではない．本提案方式では，これに加えて，カードの所有者自身による意図的な不正行為を防ぐ目的でカードの機能を利用する．たとえば，カードが秘密鍵を所有者にも公開しないのは，所有者自身が秘密鍵を他者へ譲与（貸与）することや，秘密鍵を用いて証明書の偽造を図ることを防ぐためである．

しかし，これまでの検討で見てきたように，4 章で提案したプロトコルでは，わずかな確率にせよ何らかの事情でカードの秘密鍵が外部に洩れることも仮定し，これへの対処法が盛り込まれている．実際には，この仮定がシステム全体にとって大きな負担となっている．以下には，カードの秘密鍵が絶対に洩れず，また解読されず，アルゴリズムの改ざんも不可能なことが保証された場合の簡易な証明書発行プロトコルを示す（**図 3** 参照）．この簡易版と対応する検証プロトコルは，関数 g と補足情報 c を使わない点で若干簡単になるが，オリジナルとほぼ一致する．また無効化処理についても同様である．

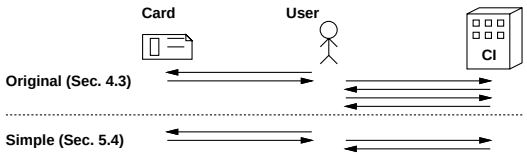

図 3 証明書発行時に発生する通信
Fig. 3 Communication at issuing a certificate.

発行プロトコル（簡易版）

証明書 $C_o = (s_o, P_o, M_o, E_o, D_o)$（$c_o$ は未使用のため削除）は，カードの正当性を証明するための証明書である．

(1) ユーザは，鍵 P および P_o に対するカード内での識別番号と $(M, E_o, D_o, P_{CI}, e_J)$ をカードに送る．

(2) カードは乱数 (r, u, v, x) を適当に選び，以下の値 (α, β) を計算する．

$$\alpha = Blind(P_{CI}, Z, r) \tag{11}$$

$$\beta = E_{crypt}(e_J, r, x) \tag{12}$$

ただし，

$$X = Cv_{crypt}(e_J, E_o, u) \tag{13}$$

$$Y = Cv_{code}(D_o, v) \tag{14}$$

$$Z = P||X||Y||M \tag{15}$$

カードは，公開鍵 P_o に対応する秘密鍵 S_o を用いて，式 (16) の文に対する署名 s_{Card} を作成し，$(s_{Card}, \alpha, \beta, r, u, v, x)$ をユーザに返す．

$$\alpha||\beta||E_o||D_o||M \tag{16}$$

(3) ユーザは $(s_{Card}, \alpha, \beta)$ と C_o を CI に送る．

(4) CI は，C_o の発行局の公開鍵と裁判官の公開する 2 つのリストを用いて，C_o の有効性を確認する．CI は署名 s_{Card} も確認する．

これに合格した場合，CI は以下の値 s' を計算し，これをユーザに返す．また，(s', E_o, β) を保存する．

$$s' = Sign(S_{CI}, \alpha) \tag{17}$$

(5) ユーザは，新しい証明書 $C = (s, P, M, E, D)$ を以下のように計算する．

- $s = Unblind(P_{CI}, s', r)$
- $E = X$
- $D = Y$

6. むすび

本稿では，耐タンパ性を持つ個人端末と電子的な証明書を利用して，個人情報を不必要に漏洩しない新たな認証方式の枠組みを提案した．提案した方式は，耐タンパ性を有効に活用している．提案した方式では，裁判官という特殊な存在を設定し，ユーザの匿名性を確保しつつかつ匿名性を利用したユーザの違反行為に対応している．裁判官を除くすべての存在に対して，ユーザが自分についてのすべての行動情報の流れを制御できる点が大きな特徴である．今後の課題としては，IC カードなどの実装媒体を考慮した具体的暗号関数の選定やプロトコルのさらなる効率化，より深い安全性の検討と追求などがあげられる．

謝辞 本研究にあたり，有益なご討論をいただいた NTT 情報流通プラットフォーム研究所の齋藤孝文氏（現サービスインテグレーション基盤研究所），桑名栄二氏，曽根岡昭直氏〔現 NTT データ（株）〕に感謝いたします．また，様々な議論の場を通してご協力いただいた同研究グループの皆様に感謝いたします．

参考文献

1) Brands, S.: Untraceable Off-line Cash in Wallet with Observers, *Proc. CRYPTO '93*, LNCS, Vol.773, pp.302–318, Springer-Verlag (1993).
2) Brands, S.: Off-Line Electronic Cash Based on Secret-Key Certificates, *Proc. LATIN '95* (1995).
3) Chaum, D.: Security without Identification: Transaction Systems to Make Big Brother Obsolete, *Comm. ACM*, Vol.28, No.10, pp.1030–1044 (1986).
4) Chaum, D., Fiat, A. and Naor, M.: Untraceable Electronic Cash, *Proc. CRYPTO '88*, LNCS, Vol.403, pp.319–327, Springer-Verlag (1988).
5) Diffie, W. and Hellman, M.: New direction in cryptography, *IEEE Trans. Information Theory*, Vol.IT-22, No.6, pp.644–654 (1976).
6) Elgamal, T.: A Public Key Cryptosystem and a Signature Scheme Based on Discrete Logarithms, *IEEE Trans. Information Theory*, Vol.IT-31, No.4, pp.469–472 (1985).
7) Ellison, C.M.: Establishing Identity Without Certification Authorities, *6th USENIX Security Symposium* (1996).
8) Ellison, C.M., Frantz, B., et al.: SPKI Certificate Theory, RFC 2693 (1999).
9) Goldwasser, S. and Micali, S.: Probabilistic Encryption, *Journal of Computer and System Sciences*, Vol.28, pp.270–299 (1984).
10) ITU-T Recommendation X.509: *Information Technology – Open Systems Interconnection – The Directory: Authentication Framework* (1997).
11) Okamoto, T. and Uchiyama, S.: A New Public-Key Cryptosystem as Secure as Factoring, *Proc. EUROCRYPT '98*, LNCS, Vol.1403, pp.308–318, Springer-Verlag (1998).
12) Rivest, R., Shamir, A. and Adleman, L.: A Method for Obtaining Digital Signatures and Public-Key Cryptosystems, *Comm. ACM*, Vol.21, No.2, pp.120–126 (1978).
13) Rivest, R.L. and Lampson, B.: SDSI – A Simple Distributed Security Infrastructure. http://theory.lcs.mit.edu/~cis/sdsi.html (1996).
14) Tsiounis, Y. and Yung, M.: On the Security of ElGamal Based Encryption, *PKC '98*, LNCS, Vol.1431, pp.117–134, Springer-Verlag (1998).
15) 満保雅浩，岡本栄司：限定匿名証明書の使用者を指定する方法について，信学技報，ISEC96-34, pp.9–20 (1996).
16) 岡本龍明，太田和夫：理想的電子現金方式の一方法，電子通信学会論文誌（D-I），Vol.J76-D-I, No.6, pp.315–323 (1993).

(平成 11 年 11 月 30 日受付)
(平成 12 年 6 月 1 日採録)

佐藤 直之

1997 年慶応大学大学院理工学研究科計算機科学専攻修士課程修了．同年日本電信電話（株）入社．入社直後は情報検索技術の研究に務め，その後，暗号理論と暗号理論を応用した新サービスについての研究および開発に従事．現在，NTT 情報流通プラットフォーム研究所所属．

鈴木 英明（正会員）

1985 年早稲田大学理工学部数学卒業．同年，NTT 武蔵野電気通信研究所入所．以来，知識獲得，定理証明系，リエンジニアリング，プログラムの抽象化検索，情報流通基盤技術の研究に従事．現在，NTT 情報流通プラットフォーム研究所主任研究員．ACM，IEEE-CS，電子情報通信学会各会員．