# Digital Payment Systems with Passive Anonymity-Revoking Trustees


Jan Camenisch[1]* Ueli Maurer[1] Markus Stadler[2]

[1] Department of Computer Science
ETH Zurich
CH-8092 Zurich, Switzerland
Email: {camenisch|maurer}@inf.ethz.ch

[2] UBILAB
Union Bank of Switzerland
Bahnhofstrasse 45
CH-8021 Zurich, Switzerland
Email: stadler@ubilab.ubs.ch



**Abstract.** Anonymity of the participants is an important requirement for some applications in electronic commerce, in particular for payment systems. Because anonymity could be in conflict with law enforcement, for instance in cases of blackmailing or money laundering, it has been proposed to design systems in which a trustee or a set of trustees can selectively revoke the anonymity of the participants involved in suspicious transactions. From an operational point of view, it can be an important requirement that such trustees are neither involved in payment transactions nor in the opening of an account, but only in case of a justified suspicion. In this paper we propose the first efficient anonymous digital payment systems satisfying this requirement. The described basic protocol for anonymity revocation can be used in on-line or off-line payment systems.



**Keywords:** Digital payment systems, electronic money, cryptography, privacy, anonymity revocation.


## 1 Introduction

In most presently-used payment systems the protection of the user's privacy relies exclusively on administrative and legal measures. Using cryptographic tools, in particular blind signature schemes [9], it is possible to design electronic payment systems that allow the customers to remain anonymous, without affecting the other security requirements of the system (e.g. [2, 6, 10, 11]). However, while protecting the honest customers' privacy, the anonymity also opens the door for misuse by criminals, for instance for perfect blackmailing [19] or for money laundering.


* Supported by the Swiss Commission for Technology and Innovation (KTI), and by the Union Bank of Switzerland.

Therefore, in order to make anonymous payment systems acceptable to governments and banks, they must provide mechanisms for revoking a participant's anonymity under certain well-defined conditions. Such anonymity revocation must be possible only for an authorised trusted third party or a set of such parties. In this paper we refer to trusted third parties as *trustees*. In a concrete scenario a trustee could be a judge or a law enforcement agency.

The concept of *anonymity-revocable payment systems*, sometimes called fair payment systems, was introduced independently in [4] and [18]. The customer's privacy cannot be compromised by the bank nor by the payee, even if they collaborate, but the trustee or a specified set of trustees can (in cooperation with the bank) revoke a customer's anonymity. It is understood that the trustee(s) answer a request only if there exists sufficient evidence that a transaction is not lawful.

All previously proposed anonymity-revocable systems [4, 7, 8, 13, 18] are either inefficient because they are based on the cut-and-choose paradigm, or they require the trustee's participation in the opening of accounts or even in withdrawal transactions.

From an operational point of view, it is an important requirement that a trustee can be passive, i.e., that he need not be involved in regular transactions nor when a customer opens a new account. The goal of this paper is the design of the first efficient anonymous digital payment systems satisfying this requirement.

## 2 Digital payment systems

An electronic payment system consists of a set of protocols between three interacting parties: a bank, a customer (the payer), and a shop (the payee). The customer and the shop have accounts with the bank. The goal of the system is to transfer money in a secure way from the customer's account to the shop's account. It is possible to identify three different phases: a *withdrawal phase* involving the bank and the customer, a *payment phase* involving the customer and the shop, and a *deposit phase* involving the shop and the bank. In an *off-line* system, each phase occurs in a separate transaction, whereas in an *on-line* system, payment and deposit take place in a single transaction involving all three parties.

The bank, the shop and the customer have different security requirements. The bank must ensure that money can be deposited only if it has previously been withdrawn. In particular, double-spending of digital money must be impossible. The shop, upon receiving a payment in an off-line system, must be assured that the bank will accept the payment. Finally, the customer must be assured that the withdrawn money will later be accepted for a payment and that the bank is not able to claim that the money has already been spent (framing), i.e., falsely accuse him of double-spending. Furthermore, the customer may require that his privacy be protected. We refer to [6] for a detailed discussion of security requirements for payment systems.

Anonymous electronic payment systems (e.g. [2, 6, 10, 11]) are based on a cryptographic mechanism called a blind signature scheme [5, 9]. Such a signature scheme allows a signer (the bank) to sign a message without seeing its content. Furthermore, while anyone, including a shop or the bank, is able to verify such a signature, even the bank is not able to link a particular signature with a particular instance of signing a message. In order to implement an anonymous payment system based on a blind signature scheme, any message signed (blindly) by the bank with the secret key corresponding to a particular public key is agreed to have a certain value (e.g. $10).

An obvious problem with such a scheme is that money can in principle be spent more than once. In an on-line system, double-spending can be prevented by checking for multiple deposits. This requires that all deposit transactions (at least within the validity period of the bank's public key) are stored by the bank. In an off-line system, double-spending cannot be prevented, but it is possible to design systems that allow to revoke a customer's anonymity when the money is spent more than once. This can be achieved by assuring that the customer's identity is properly encoded in the signed message and by having the customer answer a challenge message during the payment such that the identity can be computed from the answers to two different challenges. Alternatively, the anonymity revoking mechanism of this paper can be used.

## 3 Anonymity revocation by a trustee

Anonymity revocation by a trustee means that, when the need arises, the trustee can link a withdrawal transaction with the corresponding deposit transaction. There are two types of anonymity revocation, depending on which kind of information is available to the trustee:

- *Withdrawal-based* anonymity revocation: Based on the bank's view of a withdrawal transaction, the trustee can compute a piece of information that can be used (by the bank or a payee) to recognize the money when it is spent later. This type of anonymity revocation can for instance be used in case of blackmailing. When the owner of an account is forced to withdraw money and to transfer it to an anonymous criminal, the account owner could secretly inform the bank and the trustee could be asked to compute a value that can be put on a black list and linked with the money when it is deposited.
- *Payment-based* anonymity revocation: Based on the bank's view of a deposit transaction, the trustee determines the identity of the person who had withdrawn the money. This may for instance be needed when the suspicion of money laundering arises.

One of the security requirements of such a payment system is that the trustee must be capable only of anonymity revocation but that he cannot play a different role in the system. In particular, the trustee must be unable to forge money.

It it possible to distinguish three different approaches to achieving the above goals according to the type of the trustee's involvement.

1. The trustee is involved in every withdrawal. In such systems [7, 13] the trustee plays the role of an intermediary during the withdrawal protocol and performes the blinding operation on behalf of the customer. The trustee can then trivially revoke the anonymity if needed.
2. The trustee is involved in the opening of accounts, but not in transactions (e.g. [8]). Such systems are potentially more efficient because normally an account is used for more than a single transaction.
3. The trustee is not involed in any protocols of the payment system but is needed only for anonymity-revocation. In such systems the customer proves the bank in the withdrawal protocol that the coin and the exchanged messages contain information, encrypted under the public key of a trustee, that allow revoking the anonymity. This can in principle be achieved by application of the well-known cut-and-choose paradigm, as described independently in [4] and [18]. However, such a system would be inefficient as explained in the following rough description of the scheme of [18]. A more efficient scheme is proposed in this paper.

We now describe the scheme of [18]. In order to obtain a blind signature on a message $m$, the customer prepares $2K$ blinded messages, each of which contains $m$ encrypted with the trustee's public key as well as a session identifier encrypted with the trustee's public key. $K$ is a security parameter. These encryptions are probabilistic (i.e. the text is padded with a random string of at least 64 bits before encryption) in order to prevent decryption by an exhaustive search over a small set of possible values. To check that these messages are properly formed, the bank chooses a random subset of $K$ blinded messages and asks the customer to open all of them, where "open" means presenting the random padding used for encrypting the session identifier. For the purpose of possible later anonymity revocation, the bank stores the corresponding $K$ encryptions of $m$. Then it blindly signs the remaining $K$ messages that were not opened. The verification of such a coin (a blind signature for the message $m$) consists of the verification of the bank's signature as well as the verification that $m$ had correctly been encrypted for the trustee.

In the described system, withdrawal-based revocation can be achieved by asking the trustee to open the encryptions of $m$ the bank obtained during the withdrawal protocol. Payment-based anonymity revocation can be achieved by asking the trustee to decrypt the encrypted session ID contained in each of the $K$ components of the signature. The probability that a dishonest customer manages to escape payment-based anonymity revocation is $1/\binom{2K}{K} \approx 2^{-2K}/\sqrt{\pi K}$. The same holds for withdrawal-based revocation. To achieve a reasonable security, $K$ should be at least 20; hence both signatures and the revocation information stored by the bank are long.

The goal of this paper is to propose an efficient anonymity-revocable payment system that allows both types of anonymity revocation and in which, in contrast to the previously proposed efficient systems, the trustee is completely passive unless he is asked to revoke the anonymity of a person. In particular, after initially publishing a public key, the trustee need neither be involved in the opening of an account nor in any withdrawal or deposit transaction.

## 4 Building blocks

We briefly describe a few well-known cryptographic building blocks based on the computational difficulty of the discrete logarithm problem and then describe our main building block (protocol **P**). Variations of this protocol **P** have previously been proposed in [2] and [12].

Let $G$ be a finite cyclic group of order $q$ and let $g \in G$ be a generator of $G$, such that computing discrete logarithms to the base $g$ is infeasible. Let $\mathcal{H}_\ell : \{0,1\}^* \rightarrow \{0,1\}^\ell$ ($\ell \approx 128$) denote a cryptographically strong hash function. For a number of different cryptographic schemes, a public key is constructed by computing $y = g^x$ for a secret key $x$ chosen at random from $\mathbb{Z}_q$.

We will make use of extensions of the Schnorr signature scheme [16]. A Schnorr signature for a message $m$ is a pair $(c, s)$ with $c \in \{0,1\}^\ell$ and $s \in \mathbb{Z}_q$, satisfying the verification equation

$$c = \mathcal{H}_\ell(m \| g^s y^c).$$

Such a signature can be generated only if one knows the secret key $x$, by choosing $r$ at random from $\mathbb{Z}_q$ and computing $c$ and $s$ according to

$$c = \mathcal{H}_\ell(m \| g^r)$$

and

$$s \equiv r - cx \pmod{q}.$$

Basically, a Schnorr signature with respect to a public-key $(g, y)$ is a proof (depending on the message $m$ to be signed) that the signer knows the discrete logarithm of his public key $y$ to the base $g$.

We now give definitions for two cryptographic primitives for proving knowledge and equality of discrete logarithms, respectively. A proof of knowledge of the discrete logarithm of a group element $h$ to the base $g$, denoted $PKLOG(g, h)$ consists of a Schnorr signature with respect to a public-key $(g, h)$ for the message $g \| h$, i.e.,

$$PKLOG(g, h) = (c, s)$$

with

$$c = \mathcal{H}_\ell(g \| h \| g^s h^c).$$

A (message-dependent) proof of equality of the discrete logarithm of $h_1$ to the base $g_1$ and the discrete logarithm of $h_2$ to the base $g_2$, denoted $PLOGEQ(m, g_1, h_1, g_2, h_2)$, is a pair $(c, s)$ satisfying the following condition:

$$PLOGEQ(m, g_1, h_1, g_2, h_2) = (c, s)$$

with

$$c = \mathcal{H}_\ell(m \| g_1 \| g_2 \| h_1 \| h_2 \| g_1^s h_1^c \| g_2^s h_2^c).$$

Such a proof can be obtained if and only if one knows the discrete logarithms $\log_{g_1} h_1$ and $\log_{g_2} h_2$ and if they are both equal to some value $x$. One first chooses $r$ at random from $\mathbb{Z}_q$ and computes $c = \mathcal{H}_\ell(m \| g_1 \| g_2 \| h_1 \| h_2 \| g_1^r \| g_2^r)$ and $s \equiv r - cx \pmod{q}$. Note that the message $m$ can be the empty string.

The following protocol is a blind Schnorr signature protocol [15]. When a message $m$ is signed by this protocol, the signer $\mathcal{B}$ learns neither $m$ nor the resulting signature $(c, s)$.

$$
\begin{array}{lcl}
\mathcal{C}(m, g, y) & & \mathcal{B}(g, x) \\
\hline
 & & \tilde{r} \in_R \mathbb{Z}_q \\
 & & \tilde{t} := g^{\tilde{r}} \\
 & \xleftarrow{\quad \tilde{t} \quad} & \\
\gamma, \delta \in_R \mathbb{Z}_q & & \\
t := \tilde{t} g^{\gamma} y^{\delta} & & \\
c := \mathcal{H}_\ell(m \| t) & & \\
\tilde{c} := c - \delta \pmod{q} & \xrightarrow{\quad \tilde{c} \quad} & \\
 & & \tilde{s} := \tilde{r} - \tilde{c}x \pmod{q} \\
 & \xleftarrow{\quad \tilde{s} \quad} & \\
s := \tilde{s} + \gamma \pmod{q} & & \\
\hline
\downarrow & & \downarrow \\
(c, s) & & (\tilde{r}, \tilde{t}, \tilde{c}, \tilde{s})
\end{array}
$$

If both $\mathcal{C}$ and $\mathcal{B}$ follow the protocol, $\mathcal{C}$ obtains a valid Schnorr signature $(c, s)$ of the message $m$:

$$g^s y^c = g^{\tilde{s}+\gamma} y^{\tilde{c}+\delta} = g^{\tilde{r}-\tilde{c}x+\gamma+\tilde{c}x} y^{\delta} = \tilde{t} g^{\gamma} y^{\delta} = t \ .$$

The signature is valid because $c = \mathcal{H}_\ell(m \| t)$ holds. $\mathcal{B}$'s output of the protocol is the entire view consisting of $\tilde{r}$, $\tilde{t}$, $\tilde{c}$, and $\tilde{s}$. Note that the pair $(c, s)$ is statistically independent of the pair $(\tilde{c}, \tilde{s})$ because $\gamma$ and $\delta$ are randomly and uniformly chosen from $\mathbb{Z}_q$, and that therefore the message-signature pair and $\mathcal{B}$'s view are unlinkable.

This basic blind issuing protocol for Schnorr signatures is now extended to a protocol that not only proves $\mathcal{B}$'s knowledge of the secret key $x$, but simultaneously that the discrete logarithm of a value $z_w$ to the base $h_w$ is equal to $x$. $\mathcal{C}$ can then modify this proof in order to obtain a message-dependent proof of equality of $\mathcal{B}$'s secret key and the logarithm of a value $z_p$ to a base $h_p$, with $h_p = h_w^{\alpha} g^{\beta}$, and $z_p = z_w^{\alpha} y^{\beta}$ for some $\alpha, \beta \in \mathbb{Z}_q$.

**Protocol P:**

| $\mathcal{C}(m, g, y, h_w, \alpha, \beta)$ | | $\mathcal{B}(g, x, h_w)$ |
|---|---|---|
| | | $z_w := h_w^x$ |
| | $\xleftarrow{\quad z_w \quad}$ | |
| $h_p := h_w^\alpha g^\beta$ <br> $z_p := z_w^\alpha y^\beta$ | | $\tilde{r} \in_R \mathbb{Z}_q$ <br> $\tilde{t}_g := g^{\tilde{r}}$ <br> $\tilde{t}_h := h_w^{\tilde{r}}$ |
| | $\xleftarrow{\quad \tilde{t}_g,\, \tilde{t}_h \quad}$ | |
| $\gamma, \delta \in_R \mathbb{Z}_q$ <br> $t_g := \tilde{t}_g g^\gamma y^\delta$ <br> $t_h := \tilde{t}_h^\alpha \tilde{t}_g^\beta h_p^\gamma z_p^\delta$ <br> $c := \mathcal{H}_\ell(m\|g\|h_p\|y\|z_p\|t_g\|t_h)$ <br> $\tilde{c} := c - \delta \pmod q$ | | |
| | $\xrightarrow{\quad \tilde{c} \quad}$ | |
| | | $\tilde{s} := \tilde{r} - \tilde{c}x \pmod q$ |
| | $\xleftarrow{\quad \tilde{s} \quad}$ | |
| $s := \tilde{s} + \gamma \pmod q$ | | |
| $\downarrow$ <br> $(m, h_p, z_p, c, s)$ | | $\downarrow$ <br> $(h_w, z_w, \tilde{r}, \tilde{t}_g, \tilde{t}_h, \tilde{c}, \tilde{s})$ |

Note that $(c, s)$ is a valid message-dependent $PLOGEQ(m, g, y, h_p, z_p)$ for message $m$. It can easily be proved that $\mathcal{B}$'s view of protocol **P** is unlinkable to (i.e., statistical independent of) $\mathcal{C}$'s output $(m, h_p, z_p, c, s)$.

An important property of protocol **P** is that $\mathcal{C}$ can obtain a valid output only if he computes $h_p$ as $h_w^\alpha g^\beta$ for some $\alpha, \beta \in \mathbb{Z}_q$. The following payment system will make use of this property to construct an anonymity-revocation mechanism.

## 5 An efficient anonymous payment system with a passive anonymity-revoking trustee

For simplicity, we describe only a simple on-line payment scheme with a single denomination of coins. An extension to multiple denominations is trivial. The scheme can also be extended to off-line payments, as described in Section 6. The withdrawal protocol described in this section is based on a fair blind signature scheme proposed in [17].

System setup:

1. The bank chooses a finite group $G$ of prime order $q > 2^{170}$, such that computing discrete logarithms in $G$ is infeasible. Note that such a group is cyclic and every element (except the neutral element) is a generator of the group. Three elements $g$, $g_1$ and $g_2$ are chosen by a publicly verifiable pseudo-random mechanism which guarantees that the discrete logarithms of none of these elements with respect to another one is known. Finally, the bank

chooses a secret key $x \in_R \mathbb{Z}_q$ and computes the public key $y = g^x$. The bank publishes $G$, $g$, $g_1$, $g_2$, and $y$.

2. The trustee randomly chooses his secret key $\omega \in \mathbb{Z}_q^*$ and computes his public key $g_T = g_2^{\omega}$. He publishes $g_T$.

The withdrawal protocol, which makes use of our building block (protocol **P**), is described below ($\epsilon$ denotes the empty string).

| Customer$(g, y, g_1, g_2, g_T)$ | | Bank$(g, x, g_1, g_2, g_T)$ |
|---|---|---|
| | $\xrightarrow{\text{Identification}}$ | |
| choose a random coin number $c\#$<br>$\alpha \in_R \mathbb{Z}_q^*$<br>$h_w := g_1^{\alpha^{-1}} g_2$<br>$d := g_T^{\alpha}$<br>$U := PLOGEQ(\epsilon, g_1, (h_w/g_2), d, g_T)$ | | |
| | $\xrightarrow{h_w,\, d,\, U}$ | verify $U$ and stop if verification fails |
| $\mathcal{C}(c\#, g, y, h_w, \alpha, 0)$ | | $\mathcal{B}(g, x, h_w)$ |
| | **Protocol P** | |
| $\downarrow$ | | $\downarrow$ |
| $(c\#, h_p, z_p, c, s)$ | | $(h_w, z_w, \tilde{r}, \tilde{t}_g, \tilde{t}_h, \tilde{c}, \tilde{s})$ |
| $V := PKLOG(g_2, h_p/g_1)$<br>$W := (c, s) = PLOGEQ(c\#, g, y, h_p, z_p)$ | | debit customer's account |
| $\downarrow$ | | $\downarrow$ |
| $(c\#, h_p, z_p, V, W)$ | | $(h_w, z_w, U, d, \tilde{r}, \tilde{t}_g, \tilde{t}_h, \tilde{c}, \tilde{s})$ |

The withdrawn coin consists of the coin number $c\#$ and the values $h_p$, $z_p$, $V$, and $W$ and can be verified by checking the two proofs $V$ and $W$. For the purpose of later anonymity revocation, the bank keeps the value $d$.

Note that (in contrast to protocol **P**) now the bank's view and the generated signature (coin) are only computationally unlinkable: i.e., the bank could link by testing whether $\log_{g_T} d = \log_{g_2}(h_p/g_1)$. However, this is intractable because the bank does not know $\log_{g_2} g_T$ (see [2] for a discussion about the so called decision-Diffie-Hellman problem).

In the computation of $U$ we have made use of the fact that by exchanging base and input element of a discrete logarithm computation, the resulting discrete logarithm is inverted modulo the group order:

$$\log_g h \;\equiv\; (\log_h g)^{-1} \pmod{q}.$$

Therefore, the proof $U = PLOGEQ(\epsilon, g_1, (h_w/g_2), d, g_T)$ in the withdrawal protocol proves that the discrete logarithm $\log_{g_1}(h_w/g_2)$ is inverse to $\log_{g_T} d$.

We now discuss the anonymity-revocation mechanism. The facts that the customer

- knows $\log_{g_1}(h_w/g_2)$ as can be verified by checking $U$,
- knows $\log_{g_2}(h_p/g_1)$ as can be verified by checking $V$,
- neither knows $\log_{g_1} g$, $\log_{g_2} g$, nor $\log_{g_1} g_2$ (which is guaranteed by the way these elements were generated),

imply that the customer has executed protocol **P** with

$$\alpha = (\log_{g_1}(h_w/g_2))^{-1} = \log_{g_T} d = \log_{g_2}(h_p/g_1)$$

and $\beta = 0$. This relationship can now be used for anonymity revocation.

Withdrawal-based anonymity revocation is achieved as follows. Given the value $d$ observed in a withdrawal transaction, the trustee computes

$$g_1 d^{\omega^{-1}} = g_1 g_2^{\alpha} = h_p.$$

This value can be put on a black list and recognised when the coin is spent.

Payment-based revocation is achieved as follows. Given the component $h_p$ observed in a payment transaction, the trustee can compute the value

$$(h_p/g_1)^{\omega} = (g_2^{\alpha})^{\omega} = d$$

which can be compared with the corresponding value in the revocation database obtained from the withdrawal transactions.

## 6 Efficiency considerations and extensions

We now compare the efficiency of the proposed scheme with the previously proposed schemes based on the cut-and-choose paradigm. In a scheme of the latter type, a blind signature consists of $K \approx 20$ components, each of which consists of a random padding string and a public-key encrypted value. In order to achieve a reasonable security level, the lengths of these two values must be at least 64 and 512 bits, respectively, resulting in a total signature length of close to $12,000$ bits. The value stored by the bank for each withdrawal transaction is of a comparable size. The withdrawal transaction requires $4K$ public-key encryption operations, which in general is quite inefficient, but could be as fast as our scheme if RSA with small exponents is used.

In contrast, the signature in the proposed scheme consists of two group elements, two hash values, and two numbers smaller than $q$. When the group allows for a compact representation of its elements, the signatures can be quite short. For instance, elements of an elliptic curve with order $q$ over a field of cardinality close to $q$ can be represented by two field elements. Hence for $q \approx 2^{170}$, the total signature length is roughly $6 \log_2 q + 256 \approx 1300$ bits. This could even be reduced to about 1000 bits if the representation of group-elements is compressed and the challenges for the proofs $V$ and $W$ are chosen to be the same. The signature length compares favourably with a cut-and-choose based scheme.

To achieve higher security against fraudulent anonymity revocation, the protocol described in the previous section can be extended to incorporate several trustees who can only in cooperation revoke a customer's anonymity. This is achieved by letting each trustee choose a secret key $\omega_i$ and defining $\omega$ to be the product of the $\omega_i$. Raising a value to the power $\omega$ or $\omega^{-1}$ during anonymity revocation is achieved by asking all trustees to consecutively compute the $\omega_i$-th or $\omega_i^{-1}$-th powers, respectively.

To extend our scheme to off-line payments, the customer replaces the coin number $c\#$ by $t = g_2^r$ for $r$ chosen at random from $\mathbb{Z}_q$. To spend a coin the customer must provide a Schnorr-signature $(c, s)$, where $c$ must be $\mathcal{H}(m\|t)$, and the public key is $h_p/g_1$. The message $m$ must depend on (or be chosen by) the shop. This signature is a message-dependent proof of knowledge with $t$ as commitment. If the customer spends a coin twice the bank can, upon receiving both signatures, calculate $\alpha$ and thereby identify the double-spender.

Another method for extending our scheme to off-line payments would be that the customer replaces the coin number $c\#$ by a randomly chosen public key of any (fixed) signature scheme. To spend the coin he signs a message containing some shop-dependent data. Thus double-spending can be detected by the fact that more than one message was signed with respect to the same public-key. However the offender can be identified only by invoking the anonymity-revocation mechanism which is acceptable if it happens rarely. This could be guaranteed by so-called observers (as proposed in [3] and [12]) which would imply that double-spending required breaking a tamper-proof component. This method has the advantage that an arbitrary and hence very efficient signature scheme could be used (e.g. [1, 14]).

As is the case with most complex cryptographic protocols, the proposed protocol can quite convincingly be argued to be secure if computing discrete logarithms in the underlying group is infeasible, but the security cannot be proved rigorously. It is an open problem to prove that the protocol is as secure as the discrete logarithm problem.

## Acknowledgements

Some ideas of this paper are based on results of a previous cooperation with Jean-Marc Piveteau.

## References

1. D. Bleichenbacher and U. Maurer. Directed acyclic graphs, one-way functions and digital signature. In Y. Desmedt, editor, *Advances in Cryptology — CRYPTO '94*, volume 839 of *Lecture Notes in Computer Science*, pages 75–82. Springer Verlag Berlin, 1994.
2. S. Brands. An efficient off-line electronic cash system based on the representation problem. Technical Report CS-R9323, CWI, Apr. 1993.

3. S. Brands. Untraceable off-line cash in wallets with observers. In D. R. Stinson, editor, *Advances in Cryptology — CRYPTO '93*, volume 773 of *Lecture Notes in Computer Science*, pages 302–318, 1993.
4. E. Brickell, P. Gemmel, and D. Kravitz. Trustee-based tracing extensions to anonymous cash and the making of anonymous change. In *Proceedings of the 6th Annual Symposium on Discrete Algorithms*, pages pp 457–466, Jan. 1995.
5. J. Camenisch, J.-M. Piveteau, and M. Stadler. Blind signatures based on the discrete logaritm problem. In A. D. Santis, editor, *Advances in Cryptology — EUROCRYPT '94*, volume 950 of *Lecture Notes in Computer Science*, pages 428–432. Springer Verlag Berlin, 1994.
6. J. Camenisch, J.-M. Piveteau, and M. Stadler. An efficient payment system protecting privacy. In D. Gollmann, editor, *Computer Security — ESORICS 94*, volume 875 of *Lecture Notes in Computer Science*, pages 207–215. Springer Verlag, 1994.
7. J. Camenisch, J.-M. Piveteau, and M. Stadler. Faire Anonyme Zahlungssysteme. In F. Huber-Wäschle, H. Schauer, and P. Widmayer, editors, *GISI 95*, Informatik aktuell, pages 254–265. Springer Verlag Berlin, Sept. 1995.
8. J. Camenisch, J.-M. Piveteau, and M. Stadler. An efficient fair payment system. In *3rd ACM Conference on Computer and Communicatons Security*, pages 88–94, New Delhi, Mar. 1996. acm press.
9. D. Chaum. Blind signature systems. In D. Chaum, editor, *Advances in Cryptology — CRYPTO '83*, page 153. Plenum, 1983.
10. D. Chaum. Security without identification: Transaction systems to make big brother obsolete. *Communications of the ACM*, 28(10):1030–1044, Oct. 1985.
11. D. Chaum, A. Fiat, and M. Naor. Untraceable electronic cash. In S. Goldwasser, editor, *Advances in Cryptology — CRYPTO '88*, volume 403 of *Lecture Notes in Computer Science*, pages 319–327. Springer Verlag, 1990.
12. D. Chaum and T. Pedersen. Wallet databases with observers. In E. F. Brickell, editor, *Advances in Cryptology — CRYPTO '92*, volume 740 of *Lecture Notes in Computer Science*, pages 89–105. Springer-Verlag, 1993.
13. M. Jakobsson and M. Yung. Revokable and versatile electronic money. In *3rd ACM Conference on Computer and Communicatons Security*, pages 76–87, New Delhi, Mar. 1996. acm press.
14. R. Merkle. A certified digital signature. In G. Brassard, editor, *Advances in Cryptology — CRYPTO '89*, volume 435 of *Lecture Notes in Computer Science*, pages 218–238. Springer Verlag Berlin, 1990.
15. T. Okamoto. Provable secure and practical identification schemes and corresponding signature schemes. In E. F. Brickell, editor, *Advances in Cryptology — CRYPTO '92*, volume 740 of *Lecture Notes in Computer Science*, pages 31–53. Springer-Verlag, 1993.
16. C. P. Schnorr. Efficient signature generation for smart cards. *Journal of Cryptology*, 4(3):239–252, 1991.
17. M. Stadler. *Cryptographic Protocols for Revocable Privacy.* PhD Thesis, ETH Zürich, 1996. Diss. ETH No. 11651.
18. M. Stadler, J.-M. Piveteau, and J. Camenisch. Fair blind signatures. In L. C. Guillou and J.-J. Quisquater, editors, *Advances in Cryptology — EUROCRYPT '95*, volume 921 of *Lecture Notes in Computer Science*, pages 209–219. Springer Verlag, 1995.
19. S. von Solms and D. Naccache. On blind signatures and perfect crimes. *Computer & Security*, 11(6):581–583, 1992.