UNIVERSITÄT KARLSRUHE

FAKULTÄT FÜR INFORMATIK

Postfach 6980, D 7500 Karlsruhe

Value Transfer Systems Enabling Security and Unobservability

Holger Bürk, Andreas Pfitzmann

Institut für Rechnerentwurf und Fehlertoleranz
Fakultät für Informatik
Universität Karlsruhe

Interner Bericht 2/87

Value Transfer Systems Enabling Security and Unobservability

Holger Bürk, Andreas Pfitzmann

Institut für Rechnerentwurf und Fehlertoleranz
Fakultät für Informatik
Universität Karlsruhe

Interner Bericht 2/87

Written spring 1987, editorially corrected July 1988

Abstract

In present-day cashless payment systems, the banks and (by installing a Trojan Horse) even the manufacturers of the computer equipment used could easily observe who pays what amount to whom and when. With the increasing digitization of these systems, e.g. point-of-sale terminals and home banking, the amount of transaction data and their computerization drastically increases. Thereby, these payment systems become completely unacceptable, since compiling dossiers on the lifestyle and whereabouts of all clients will become easy.

Therefore, we describe the *digital payment systems enabling unobservability* of clients and arrange them in a general model to compare their different degrees of unobservability and their different levels of security. Since no single system has all desired features, we propose a suitable synthesis.

The use of such payment systems for *unobservable value exchange* over a network (e.g. money against information) between two remote parties poses an additional security problem: the lack of a simultaneous exchange causes a temporary advantage for one party, who could then stop communication. But third parties can be used to overcome this problem. We compare two rather different approaches using third parties. The first tries to provide security by third parties identifying perpetrators in case of detected fraud, whereas the second uses a third party as trustee who takes an active part in the value exchange and can be completely controlled by each absolutely anonymous party.

Keywords

Cashless payment, digital payment systems enabling unobservability of clients, anonymity cut, relation anonymity, clearing anonymity, linkability of actions, anonymous numbered accounts, anonymously transferable standard values, mutually trusted tamper-resistant devices, blindly signed digital banknotes, unobservable value exchange

----

This is an extended version of a paper presented at IFIP/Sec'86.

Table of Contents

Table of Figures

1 Introduction

In present-day cashless payment systems, the banks and (by installing a Trojan Horse [14]) even the manufacturers of the computer equipment used could easily observe who pays what amount to whom and when. With the increasing digitization of these systems, e.g. point-of-sale terminals and home banking, the amount of transaction data and their computerization drastically increases.
Thereby, these payment systems become completely unacceptable, since compiling dossiers on the lifestyle and whereabouts of all clients will become easy. Measures of law and jurisdiction alone are insufficient to prevent from collecting and using these data, since infringements can hardly be discovered and restoration of privacy is impossible in most cases. Hence, *payment systems* not only guaranteeing *security against fraud* but also enabling *unobservability of clients* are necessary. Unobservability of clients means three things:

1. Each client may do his business *anonymously*, i.e. the banks and the client's business partners (e.g. international providers of information services) together are not able to find out his identity without his cooperation.
2. Each payment is *unobservable* for inquisitive unconcerned third parties.
3. Each action of a client is *unlinkable* to actions which have taken place earlier, unless such a concatenation of actions is intentionally enabled by the client.

In view of these harder requirements, it should be noted that some cashless payment systems, e.g. home banking via BTX (= interactive videotex in Germany) or all automatic teller machines using passive (for example magnetic stripe) cards, do not even provide sufficient, i.e. both-sided, security: Clients are forced to trust banks completely, since the handwritten signature (which till now gave at least a minimum level of protection to the clients) is replaced by "proofs" the bank could imitate very easily and perfectly. To give an example, appending a TAN (Transaction Number) or a PIN (Personal Identification Number) to messages, both generated by the bank, nowadays is supposed to create legally binding declarations.

The body of this report is organized in five chapters. Chapter 2 presents a brief overview of necessary and feasible technical capabilities to solve the security and unobservability problems.
In chapter 3 we develop a general model for payment systems satisfying the above listed requirements and arrange all published payment systems of this kind in our model.
In chapter 4 we describe two different approaches to solve the additional problems of security against fraud arising when such payment systems are to be used for unobservable *value exchange* over a network between two remote parties; e.g. if one anonymous party buys useful information from another one for digital money, it should neither be possible that it receives the information and never pays the money nor that it has to pay first and never receives the information.
Finally, we discuss some socio-political impacts when these digital payment systems will work in practice.

2 Necessary Technical Capabilities

2.1 Networks Without User Observability

To enable unobservable (i.e. anonymous, unobservable by unconcerned third parties, and unlinkable) digital payments, networks without user observability are necessary. Such networks guarantee that nobody, not even the network manager, is able to figure out sender and recipient of transmitted messages, unless the sender and receiver of the respective message are cooperating. For this, it is not sufficient to hide the content of a message only, which can be done by *end-to-end-encryption* [35], since an attacker could still see who sends how many messages to whom and at what time (*traffic analysis*). To hide these traffic data too, the relation between sender and recipient must be kept secret (*unlinkability*), or stronger, both sender and recipient of a message are kept secret.

Unlinkability of sender and recipient can be realized by special network stations, so called *MIXes* [5]. A MIX collects a number of messages from different senders, changes their encodings and forwards them to the recipients in a different order. This measure hides the relation between sender and recipient from everybody but the MIX. By using more than one MIX to forward a message from the sender to the recipient, the relation is hidden from everybody who does not control all the MIXes used.

Receiving a message can be made anonymous to the network by delivering the message to all stations (*broadcast*) or at least to sufficiently many (*multicast*). A participating station can decide which messages are actually for it by scanning every message for a special attribute, called implicit address [30].

A powerful scheme for sender anonymity is *superposed sending*, which is published in [7] and is called DC-network (Dining Cryptographers network) there. Unfortunately superposed sending requires either a tremendous amount of secret communication for key exchange between the participating stations before messages can be sent, thereby complicating work in practice, or very fast generation of cryptographically strong pseudorandom keys [3, 29 p. 18].

Such expenditure can be avoided if the network is designed for preventing attackers from physically observing all lines connecting a user with the rest of the world. A simple and efficient way is to connect the user stations by *RINGs*, which are in wide use for local area networks anyway. If suitable medium access protocols are used, the sending of a user station is only observable if its two neighbor stations collude or the lines are tapped. The latter attack can be prevented by running the cable in an appropriate way. To achieve high performance in large networks, many RINGs or DC-networks can be interconnected by a backbone switching network [28, 29, 30].

2.2 Authentication

New applications of networks like digital payment systems require elaborate measures for authentication, e.g. transmitted messages may be legally binding declarations which are effective only if a proof of authorization is shown.

For authentication, digital signatures [14] are used, ensuring that a declaration of an individual can only be signed by that individual and that everybody can verify if a declaration is actually signed with the signature belonging to that individual. Digital signatures can be formed e.g. by encryption of the declaration with the best known but old-fashioned public key cryptosystem (PKCS) RSA [14] or by using any efficient digital signature system which is provably as secure as factoring is difficult [21, 20]. The latter alternative is more secure, since RSA is not provably as secure as factoring is difficult, but is clearly broken if factoring the modulus is feasible.

In the following, we use

$\mathbf{sign}_X(m)$ to describe the result of a message m signed by X, and

$\mathbf{auth}_X(m')$ to describe the check of authentication whether a message m' is signed by X, i.e. whether $m'=\mathrm{sign}_X(m)$ for some m.

If a PKCS is to be used for secrecy, the sender usually encrypts the messages with the public key of the recipient and the recipient decrypts the received messages with his corresponding private, i.e. secret, key. The PKCS used may be RSA (which has the additional property that encryption and decryption operations are commutative, see section 3.2.3) or any other PKCS or probabilistic encryption scheme [2] (a special case of a PKCS) which is provably as secure as factoring is difficult.

In order to guarantee anonymity of individuals, a digital signature must not be associated with an individual directly but with a *digital pseudonym* [5, 7] which is used by that individual. Digital pseudonyms are just (almost unique) numbers (or pairs thereof) which are created by individuals through a special random process. It is assumed that every individual uses different digital pseudonyms wherever possible to restrict linkability to a minimum.

There are two forms of authentication which should be distinguished: *self-authentication* and *granted authentication* [31].
The form of authentication when someone refers to previous declarations of his own is called *self-authentication.* This form corresponds to one's hand-written signature. With self-authentication linkability of different declarations of the same client is intentionally enabled by the client: he uses the same digital pseudonym and digital signature for the corresponding declarations.
The form of authentication corresponding to showing one's identity card, driver's license, etc., is *granted authentication*. Granted authentication is applied in those cases when a declaration can be made only when an authorization from another (e.g. superior) instance can be shown. These authorizations or *credentials* should be issued to digital pseudonyms, not directly to the individuals. Special measures [7] can prevent the third parties granting authentication, e.g. banks, from deriving undesired concatenations, even in cooperation with those to whom the credentials are shown.

3 Digital Payment Systems Enabling Security and Unobservability

In the following we assume that networks without user observability are used. Then the degree of anonymity only depends on the form of authentication used (cf. section 2.2), which is discussed in detail for each payment system.
To understand thoroughly how to design payment systems which meet all three requirements stated in chapter 1, a general model is developed first.

3.1 A General Model and General Remarks on Unobservable Payment Systems

3.1.1 Towards a General Model

Starting point of our analysis are the relations between banks and their clients from today's point of view. This leads us to a general model for any cashless payment system. Since clients and banks are different acting entities, we arrange them on two different levels, namely a level of banks and a level of clients. Some of the clients will have business relations with each other. These relations are called *dynamic*, since they often change. Each client has a relation to a bank, that is, he has established at least one account at this bank, and of course for clearing, the banks must have relations with each other, but we are not concerned whether this happens via a central bank CB or directly. All these relations are called *static*, since they are relatively enduring. Moreover, unlike the dynamic relations where only a very small fraction of all possible pairs of clients have relations with each other, the static relations are those, which are always present to a much larger extent.
Figure 1 shows the relation graph for an example with three clients X, Y, and Z, who presently have business relations with each other. Their banks are B_X, B_Y, and B_Z, respectively.

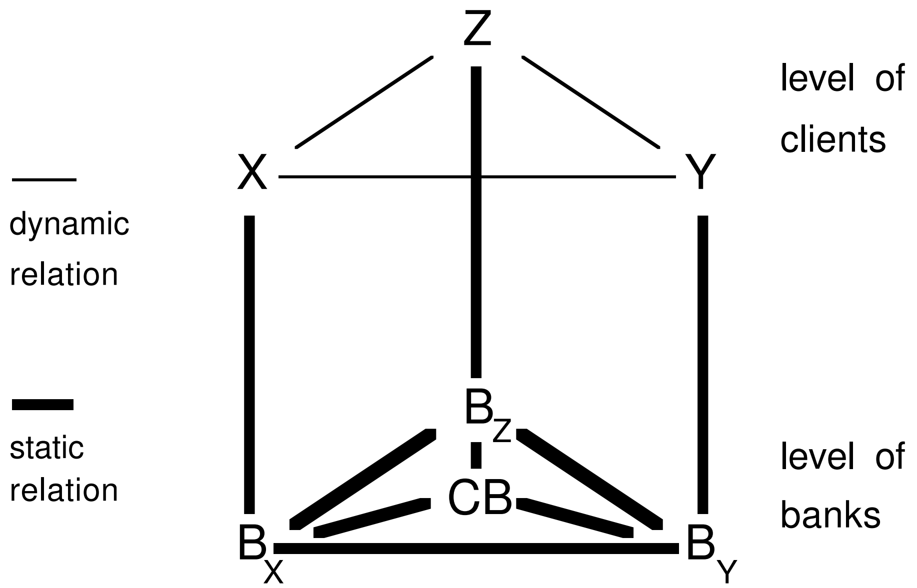


Figure 1: Relations between banks and their clients

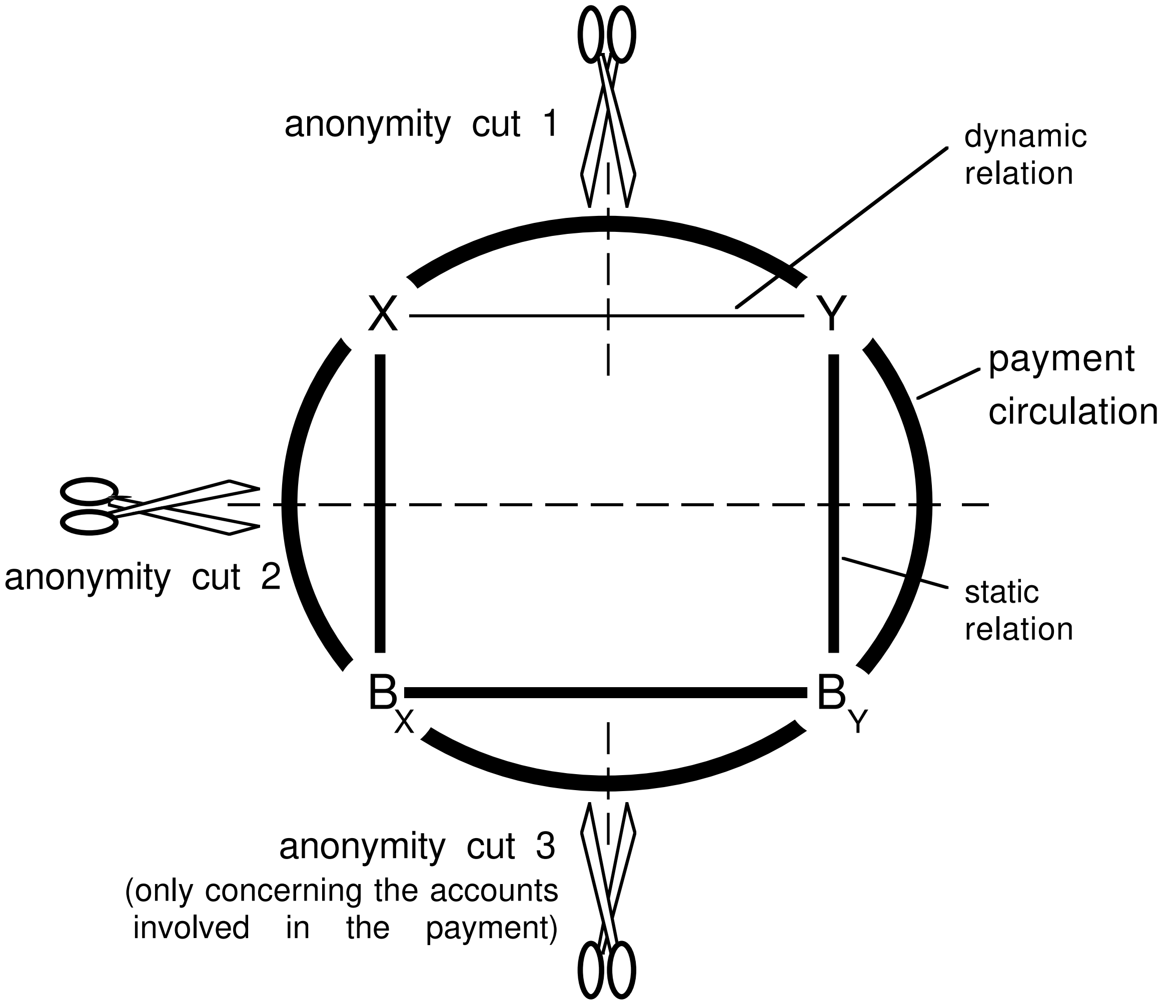


Figure 2: Anonymity cuts within payment circulation graph

The dynamic relations are those which must be particularly protected with respect to the payment system, since only they can give substantial information about the lifestyle and whereabouts of the clients. If a payment between two clients, say X and Y, is made, in general at least four instances X, Y, B_X, and B_Y, and possibly CB are involved. We obtain a payment circulation graph as shown in figure 2, which is specified in detail by the payment system used. To guarantee unobservability of the clients X and Y, we will show that the payment circulation graph has to be entirely partitioned by the three *anonymity cuts* 1, 2, and 3, as shown in figure 2.

Anonymity cut 1 stands for mutual anonymity of clients and unobservability of their payments for inquisitive third parties. Anonymity cut 2 stands for anonymity of clients towards their banks. Both anonymity cuts affect mainly the relations of the clients to the other parties involved in the payment. Therefore we call this form of anonymity *relation anonymity*. The relation graph of figure 2 shows that both anonymity cuts together totally isolate the clients.[1]

Networks without user observability (cf. section 2.1) enabling anonymous communication are necessary to realize anonymity cut 1. Since authentication is necessary in most applications, digital pseudonyms and credentials (cf. section 2.2) are used as well. With the usual authentication techniques, mutual anonymity may be reduced to one-sided anonymity or no anonymity at all

[1] But this by itself does neither prevent a bijection clients ↔ accounts nor observability of the money transfer at the level of banks. Therefore, in the long run, the clients can become completely identifiable and observable. Anonymity cut 3 is needed to overcome this problem.

whenever desired. Special forms of anonymous accounts (see subsections 3.2.1, 3.2.2, 3.2.4) are introduced to realize anonymity cut 2. Care has to be taken that the payment systems in use do not undermine the relation anonymity established by anonymity cuts 1 and 2. Therefore anonymity cut 3 is necessary.
Anonymity cut 3 signifies that for the banks payments are unlinkable to other payments (or any other actions), though the banks are actually involved in the payment process. (Please note, as shown in figure 2, that anonymity cut 3 does not cut the static relations between the banks.) Payments are unlinkable if all banks together have no better chance to find out from where money comes and to where it goes than observing when and how balances change (provided that there are any, see subsections 3.2.2, 3.2.3). We call this form of anonymity *clearing anonymity*. It is important to have clearing anonymity because otherwise any established relation anonymity of the clients would gradually be forfeited, since the quantity of information about the anonymous holders of the accounts increases with each payment, whereby at some point de-anonymization becomes feasible.

3.1.2 Clearing Anonymity – Who Should Realize It?

Clearly, clearing anonymity (anonymity cut 3) and relation anonymity towards the banks (anonymity cut 2) would not be necessary if the banks (including their computer equipment!) are absolutely trustworthy with regard to the privacy of the payments made by their clients. But can we assume that the banks are trustworthy in this sense? Even if we assume that the employees of the banks are absolutely trustworthy, the answer must be "no" for technical reasons:
It seems to be a kind of natural law that more and more complex computer systems are designed, produced and maintained with more and more complex tools and (international) teamwork. This prevents adequate validation, not to speak about rigorous verification, that their functionality is restricted to what they should do, e.g. that they contain no *Trojan Horses*.
These Trojan Horses cannot only be generated by members of the team, but also by the complex tools used, if their "output" has sufficient design complexity (which usually is the case!). Moreover, the Trojan Horse in the first tool may be implanted by another tool used to generate the first one and so on. An example for this technically unavoidable possibility is given by Ken Thompson [34]. (The more general attack by a *virus* as described by Fred Cohen [11] can be restricted to the one just described by signing programs digitally, as proposed – for slightly other reasons – in [14 p.317], and checking the signature before execution.) For all these technical reasons, the complex computer equipment used for digital payments cannot be trustworthy, and thus, neither can the banks using such equipment.

Apart from these technical problems to realize trustworthy computer equipment and the additional problem whether people would believe we had solved all these technical problems, it is for socio-political reasons (destabilization of society) too dangerous, to let anybody (irrespectively how trustworthy he seems to be) have too much power disabling thorough control. With the increasing digitization of everydays transactions, the banks would have the possibility to compute dossiers on the lifestyle and whereabouts of their clients. Thus, they and anybody who could come into possession of these data, e.g. secret services of the same or another country, a dictator, etc., can become omniscent "Big Brothers" [27]. Only the realization of anonymity cut 3 (and of course anonymity cut 1 as well) can prevent that such a scenario comes true.

If anonymity cut 3 is to be realized without participation of third parties, then either the banks can realize it, thereby substantiating their trustworthiness, or the clients can realize it, implying that their payments are made unlinkable for banks.

Let us consider the first possibility where banks realize anonymity cut 3. This possibility always has a serious drawback: How can the bank convince its clients that their payments are indeed unlinkable? It seems very unlikely that solutions to this problem exist, since on the one hand the computer equipment used is completely under control of the banks, on the other hand the clients must be sure that no information which makes their payments linkable to other actions is revealed to the banks. Using tamper-resistant containers to enclose the complex computer equipment of the bank seems not to be appropriate to overcome this problem, since their security will never be beyond doubt, they are inconvenient to maintain, and they are costly – and, of course, they do not solve the Trojan Horse problem described above.

Thus, the second possibility where clients are responsible for realizing anonymity cut 3 seems to be the right one. Since with accounts all payments involving an account are naturally linkable, the first step to realize anonymity cut 3 is to substitute the payment systems with accounts by payment systems which are based upon more appropriate representations of money. In the following we denote these representations simply as *digitally transferable currencies*, as long as they are not defined in detail. Obviously, digitally transferable currencies must be transferable between any two clients and must in some way be unlinkable for the banks. It is the latter quality which provides clearing anonymity and makes the difference between digitally transferable currencies and accounts. Therefore this latter quality is discussed in detail in the following.

There are two possibilities by which the clients can establish clearing anonymity:

1. *Unlinkability of different currencies:*
   This possibility proceeds from the natural fact that each digitally transferable currency can be properly identified and thereby traced. To disable the banks nevertheless from deriving undesired concatenations when these currencies are used for payments, the currencies should be unlinkable. In particular, special care has to be taken that they are not linkable to the person who owns them or to other currencies. Unfortunately, the latter cannot be realized completely, even if the payment system itself does so, since currencies which are transfered simultaneously are thereby linkable.
2. *Unlinkability of each currency with itself:*
   To obtain a second possibility, measures have to be taken to overcome the natural fact that each digitally transferable currency can be properly identified and thereby traced. Therefore, their identity must be changed, so that they are not linkable for anybody, except the one who made the changes. To prevent the banks from tracing, the clients must change the identity of the currencies while they are withdrawn from the banks and before or after they are spent.

If clearing anonymity is to be established by unlinkability of different currencies, it has to be made sure that, with exception of linkability by simultaneous transfer, the digitally transferable currencies are unlinkable. Obviously, this is the case, when the pseudonym of the owner of each currency is unlinkable to any other pseudonym (known to the banks). Hence, whenever a payment is made, that is, the ownership of the currencies used for payments is transfered, the corresponding pseudonyms have to be changed and must not have been used for anything else before. If they are used only once, i.e. for exactly one payment and then never again, the currencies are indeed unlinkable. The situation is shown in figure 3.

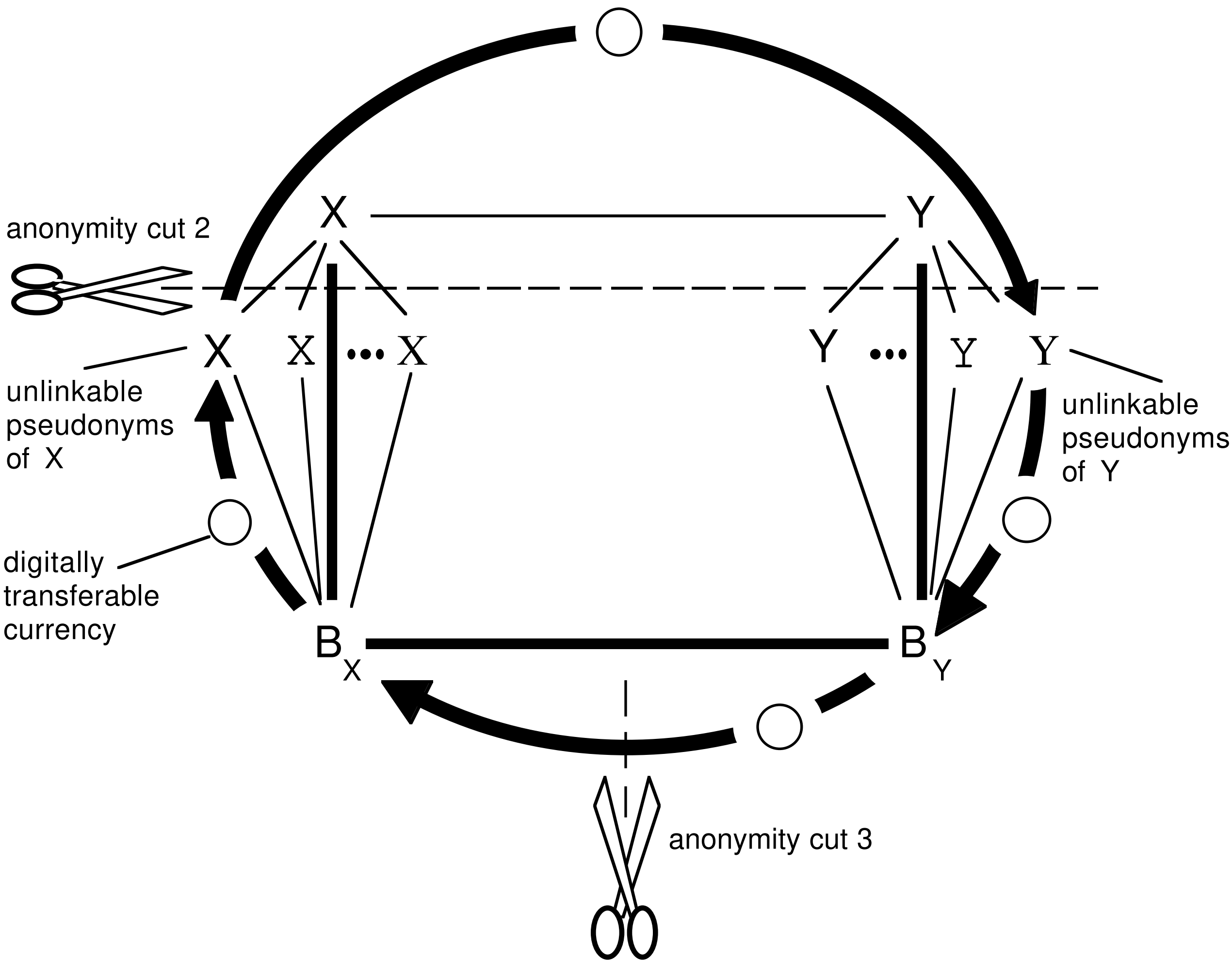


Figure 3: Unlinkability of different currencies

If the second possibility, that is, unlinkability of each currency with itself, is used to establish clearing anonymity, the identity of the digitally transferable currencies has to be changed. This is possible in case the currencies are represented by specially encoded messages so that their identity can be changed by changing their encodings, however, without thereby invalidating them. Therefore special methods (see subsection 3.2.3) are necessary. The situation is shown in figure 4. Since the bank should not, by observation of the withdrawn and deposited currencies, be able to find out which parties are actually involved in a particular payment, a varying amount of time should pass between withdrawal and deposit and the values of the currencies should not be arbitrary. Instead, a few standardized ones should be used as today.

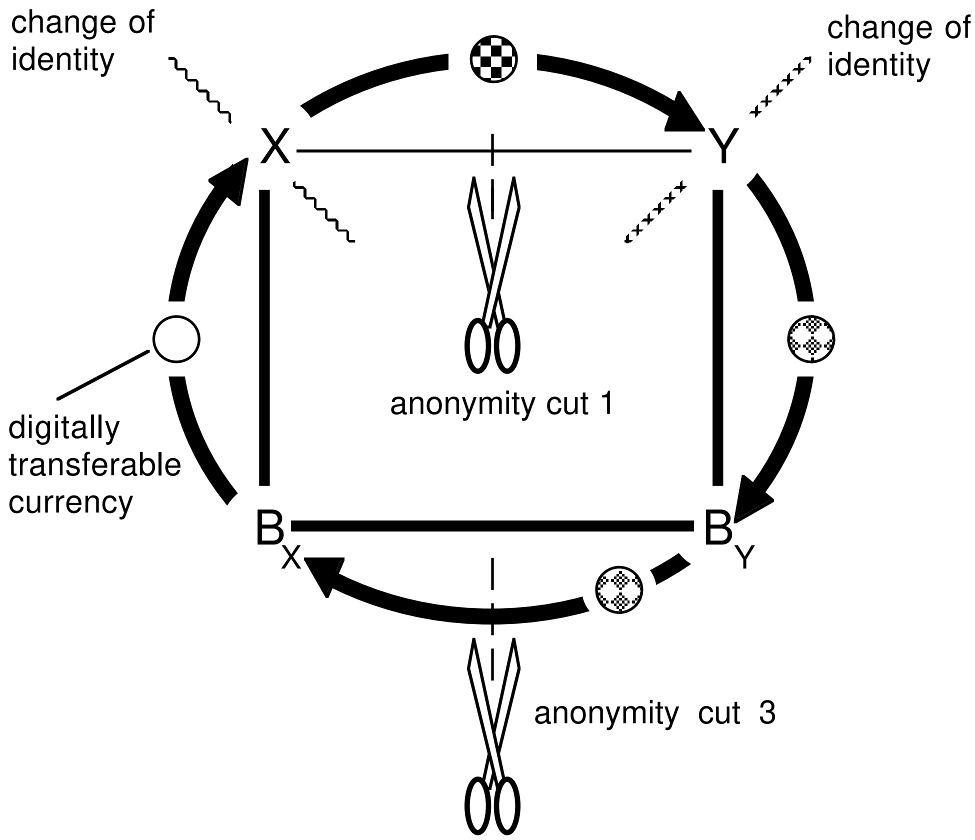


Figure 4: Unlinkability of each currency with itself

3.1.3 Security Problems posed by "Unlinkability of each Currency with Itself" and How They Can Be Solved

In this section, we discuss the security problem arising when the digitally transferable currencies are (temporarily) not under control of the bank as in the case just considered in figure 4. Of course, unauthorized persons must not have the possibility to increase the number or value of the digitally transferable currencies. This can be accomplished by using the following two methods:

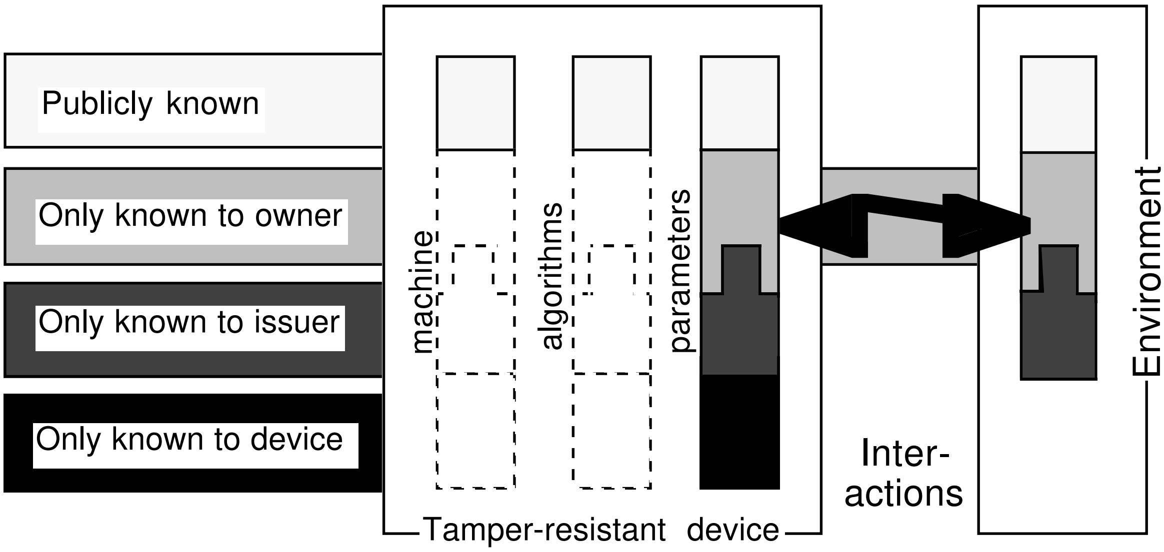
1. Use of *tamper-resistant devices* trusted by mutually suspicious groups.
2. *Coding* only.

The security of both methods (like of seemingly all other digital payment systems, also those not enabling unobservability) depends on cryptography, though unfortunately as yet the security of cryptosystems cannot be proved from scratch.

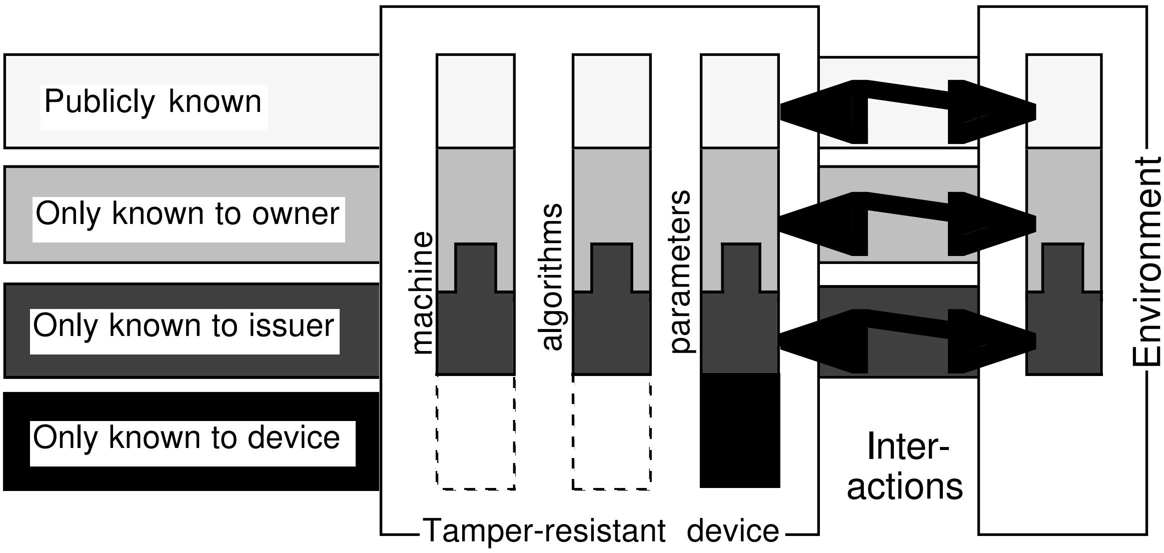
The first method additionally uses tamper-resistant devices [13] (e.g. chipcards) where the digitally transferable currencies are kept in. However, the existence of appropriate devices is admittedly questionable, since they must be trusted by mutually suspicious groups, so that the same, but to some extent reverse problematic arising when banks realize anonymity cut 3, is present: On the one hand their function has to be checkable by their users, especially they must not contain Trojan Horses which reveal information about the payments, on the other hand they must be tamper-resistant against their users, who must not be able to increase the money in the devices nor, not even by destroying the device, to extract their secret information, e.g. keys of a cryptosystem, in order to build imitations.

General Model of Tamper-resistant Devices

(Unused combinations are depicted by dashed lines, impossible combinations are omitted.)



Ideal state of affairs: Parameters secret, design publicly known, interactions under control of owner



Problematic state of affairs: Parameters and design secret, interactions not completely under control of owner

Figure 5: Mutually trusted tamper-resistant devices: How to do them and how not to do them

Figure 5 shows two rather different possibilities how tamper-resistant devices can be built: one which shows the problematic state of affairs implying that they are not mutually trustworthy, and the other which gives an idea of how an appropriate design would look like.
The second method merely uses cryptography. In this case the digitally transferable currencies are encoded messages. Since encoded messages can be copied, but must represent money only once, in each payment a bank must be involved to guarantee the payee that the information still represents money. So usually the payments have to take place on-line; only if the payee is known in advance and specified as sole recipient in the digitally transferable currencies he can verify locally whether they had not been received by him some time ago. Thus he can be sure that they still represent money.

3.1.4 Levels of Money Transfer

In summary there are three levels of money transfer, which are shown in figure 6:

1. *Level of banks*: Money transfer is completely under control of the banks. Such payments cause no special problems concerning security but they are linkable by the banks if the money is transfered using conventional accounts. Only in case the money is represented by unlinkable digitally transferable currencies, the payments are unlinkable as well.
2. *Level of mutually trusted tamper-resistant devices*: They may be regarded as minibanks, since they administer money like a bank would do. Thus they guarantee secure and unlinkable payments if proper protocols are used which only check that the partner is an authentic device but not which device exactly.
3. *Level of clients*: Usable for secure and unlinkable payments with encoded messages representing values.

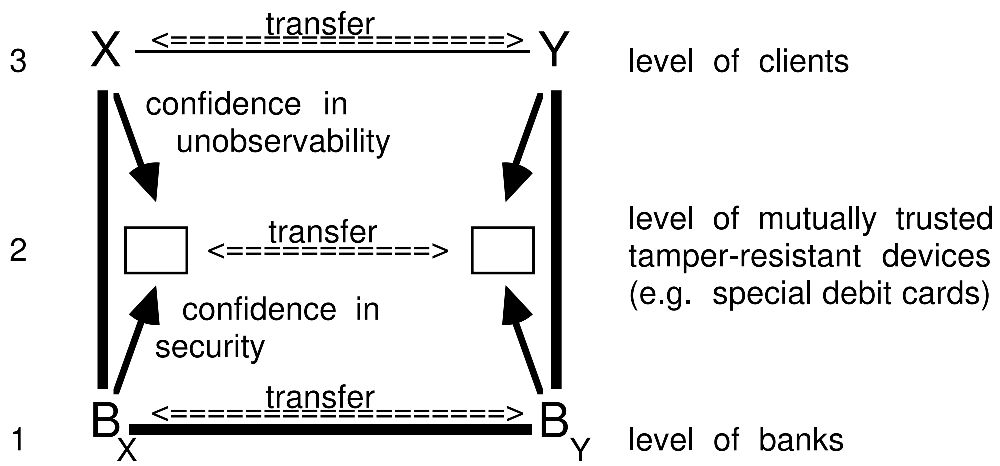


Figure 6: Levels of money transfer

3.2 Arranging the Payment Systems in the General Model

The anonymity cut model developed in the preceeding paragraph allows us to arrange all known digital payment systems enabling unobservability to a certain degree according to which anonymity cuts they realize. All these payment systems have the basic quality to enable payments between mutually anonymous individuals according to anonymity cut 1. It is important that in spite of anonymity the payer is sure to get a receipt for payment. This is similar to the value exchange problem because the payee should receive the payment only when the payer receives a receipt from the payee and vice versa. However, there is a great difference to value exchange in general, because a receipt of payment can be made out by an arbitration court after the payer has proved that he has really made a payment, but no court is able to recreate lost values (or to summon anonymous individuals without additional precautions, cf. section 4.2). Thus, the payment system should enable payers to collect evidences for the payments made. This problem is addressed below.

Three further criteria have to be considered:

1. Is it possible for payees to verify if they really got money without using the network?
   (We will see that this can be arranged using any of the known digital payment systems enabling unobservability if the payee is known to the payer in advance and the payer makes special pre-arrangements using the network. Therefore, the more interesting criterion is:)
2. Is it possible for the payer to make off-line payments to arbitrary payees in a way that the payees can verify without using the network that they really got money?
3. What is the security and anonymity of the payment system based upon:
   Tamper-resistance of devices against their owners, strength of cryptosystems used (and, if a special cryptosystem is needed, how secure is this believed or proven to be?) or, best of all, information theoretic proofs?

First, we consider a simple payment system realizing anonymity cut 2, then those realizing anonymity cut 3. The payment systems realizing anonymity cut 3 are arranged in increasing order according to how much checkable unobservability they provide for the clients. It should be mentioned, that it is appropriate to have combinations of the specific payment systems which will be discussed in section 3.4.

3.2.1 Anonymous Numbered Accounts

This proposal [28] is a "digital version" of the very popular anonymous numbered accounts already existing in Austria and in a not really anonymous form in Switzerland for decades. A client establishes an account anonymously, e.g. by means of a network. He never has to identify himself since he randomly chooses a digital pseudonym (cf. section 2.2) which is made known to the bank in order that the bank can subsequently use it to authenticate his messages (cf. figure 7). Money is transfered in the usual way in the level of banks [13]. However, in lieu of special banking forms and handwritten signatures, digitally signed messages (in standard formats) are used. The bank signs messages digitally as well, which can be verified by every party in the system.

In the following we show a protocol for a payment of k currency units (CU) from payer X to payee Y, for simplicity both clients of the same bank B.

Besides the notation introduced in section 2.2, the following notation is used to formalize the protocols throughout this and the next sections:

X* --- *m* ---> *B means X sends message m to B. m may contain text to clarify its meaning. To distinguish explanatory text from parameters, it is written in capital letters whereas parameters are written in small letters, e.g. m = ("MY NAME IS x").

If the message is signed to enable the receiver to check the authentication (which is done by applying the corresponding operator auth_x), its origin is recorded in the message by specifying that operator (which is not shown in the protocols).

***X* : operation** means X performs an operation described by "operation" locally. Some of these operations may not be formalized. In these cases the operation is explained by text enclosed in quotation-marks.

=?= denotes a boolean condition that is true only if the expressions on both sides have the same value. If a boolean condition yields "false", the protocol is aborted and, if necessary, appropriate recovery measures are taken.

For simplicity, no secrecy mechanisms for messages are shown in the protocols.

Prerequisite:
We assume that payer X has established an anonymous numbered account a_x. X signs his messages to the bank by applying the operator sign_x (which he has randomly chosen). The corresponding operator auth_x to verify authentication of such messages from X is made known to the bank and is interpreted as the digital pseudonym of the holder of account a_x.
Analogously, payee Y, the holder of an anonymous numbered account a_y, uses sign_y with the corresponding operator auth_y.
The bank uses the operator sign_B for signing its messages. The operator auth_B to check whether a message has been signed by B is publicly known.

[1] X gives order to B to transfer k CU from his account a_x to a_y, the account of Y:
  [1.1] X --- sign_x ("TRANSFER k CU FROM ACCOUNT a_x TO ACCOUNT a_y") ---> B

[2] The bank B checks the authentication of the transfer order received in step [1.1] and executes the transfer order if the account is covered:
  [2.1] B : auth_x ([1.1])
  [2.2] B : "specified account a_x belonging to auth_x ?"
  [2.3] B : "account a_x covered ?"
  [2.4] B : "transfer k CU from a_x to a_y"
  [2.5] B --- sign_B ("STATEMENT OF ACCOUNT a_x") ---> X
  [2.6] B --- sign_B ("STATEMENT OF ACCOUNT a_y") ---> Y

Note: To realize anonymity cut 1 completely, the account number a_y in step [1.1] (specified by Y in a previous step) may be enciphered with a public key of the bank to prevent X from knowing (without the cooperation of the bank) to which account the money is transfered.
To reduce the storage overhead at the bank caused by storing all the transfer orders, the messages of step [2.5] and [2.6] should be signed by X and Y, respectively, and sent back to the bank. This can be done by including them in the next transfer order (step [1.1]). Additionally, messages must now be time-stamped by the sender to prevent banks and clients from insisting on low or high statements of accounts which were only valid in the past.

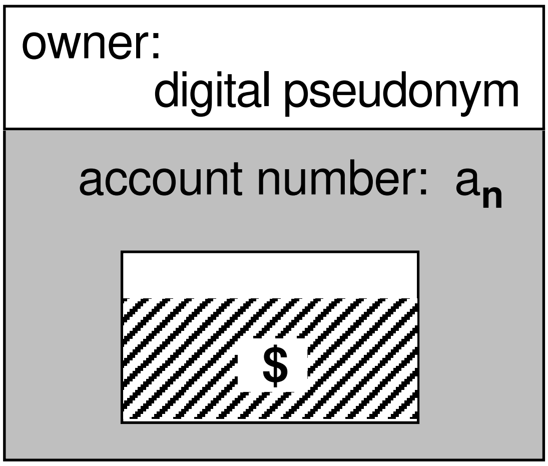


Figure 7: Digital version of anonymous numbered account

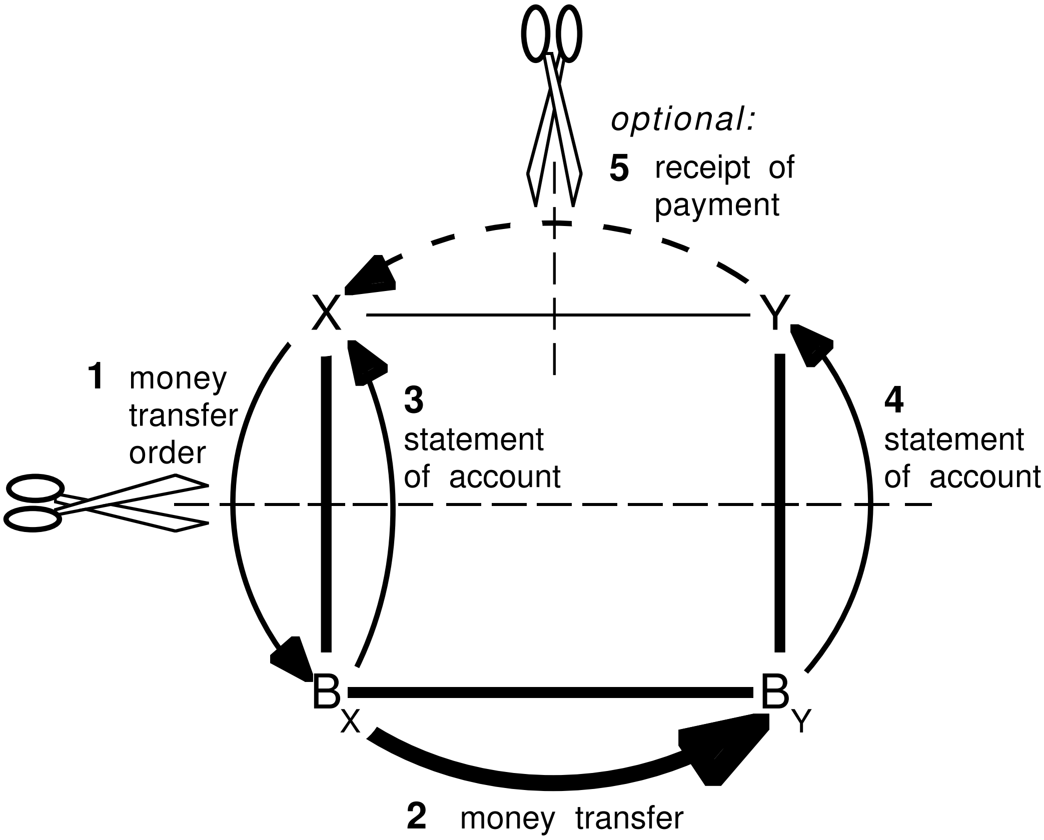


Figure 8: Payment circulation for anonymous numbered accounts

Note that the distinction between flow of signed messages (thin arrows) and flow of money (thick arrows) in figure 8 is pragmatic and is introduced for clarification only.
Since this payment system realizes only anonymity cuts 1 and 2, no clearing anonymity has been established (cf. figure 8). Thus, it is recommended to use this system only in combination with the following proposals realizing anonymity cut 3. Further characteristics of this system are:

* Payers can always prove that a payment was made, since the bank is obliged to store and confirm the money transfer orders. (If the bank refuses to confirm the money transfer order, it is sent again via a judge as witness.)

* The off-line verification of pre-arranged off-line payments is possible: the payer reserves a certain amount of money on his anonymous numbered account exclusively for payments to a specified other anonymous numbered account until an expiry date. The payer gets a corresponding confirmation signed by his bank which he can give to the payee, who checks whether he is the owner of the specified account and whether the expiry date does not exceed too early. If both conditions are satisfied, he can be sure that every money transfer order signed by the payer and drawn to the account of the payee is covered, as long as their overall amount does not exceed the amount specified in the confirmation of the reservation.
* Off-line payments to arbitrary payees cannot be verified by the payee without using the network.
* A payment is linkable to all other actions which have earlier affected an account.

3.2.2 Mutually Trusted Tamper-Resistant Devices

Chipcards are the prototypes of mass-produced tamper-resistant devices, which may be used especially for off-line money transfer between anonymous clients. Chipcards used in this way can be seen as "electronic wallets" [16, 25] or minibanks with one account [28], since the tasks of a bank are substituted and manipulation is (or should be) impossible. Thus, a chipcard is a substitute for a trusted third party by whom the payments are conducted. The trustworthiness of these devices has been discussed earlier (cf. section 3.1.3).

To transfer money with chipcards, they have to be connected by their interfaces (directly or by means of a network without user observability). A protocol allowing mutual verification if the partner chipcard is an authentic chipcard too, is executed, and then the money is transfered by transmitting digital cheques (signed messages with unique transaction numbers, cf. [13, p. 331]). The transaction is completed when the paying chipcard receives a receipt of payment from the chipcard of the payee. If the receipt of payment should fail to come, the paying chipcard is not able to decide whether the chipcard of the payee has sent a receipt which got lost or whether the transmitted digital cheque got already lost before, since this is a classical "two-army-problem" [33, p. 477]. Since the paying chipcard should not wait forever for the receipt, it should switch into a special state after a time-out which serves as evidence that a payment was made without receiving a receipt. Chipcards in this special state can be shown to a court who makes a broadcast call to reach the chipcard of the original payee (e.g. by use of an electronic blackboard). If the payment should not have reached the payee, it can be assumed that he is interested to accept the call whereby the transaction can be finished normally. In the other case a receipt of payment is issued to the payee by the court.

In principle, chipcards (suitable for off-line money transfer as well) can realize all anonymity cuts depending on implementation (cf. figure 9). If we assume that the banks issue the chipcards to their clients, anonymity cut 2 is realized if neither the identity of the holder of a chipcard (which is not needed to conduct transactions) is noted nor any of his physiological characteristics are registered. A chipcard with identity issued in this way is a kind of substitute for a digital pseudonym of its holder. Obviously, in this case and in case the chipcard has no identity at all, anonymity cut 1 is realized.

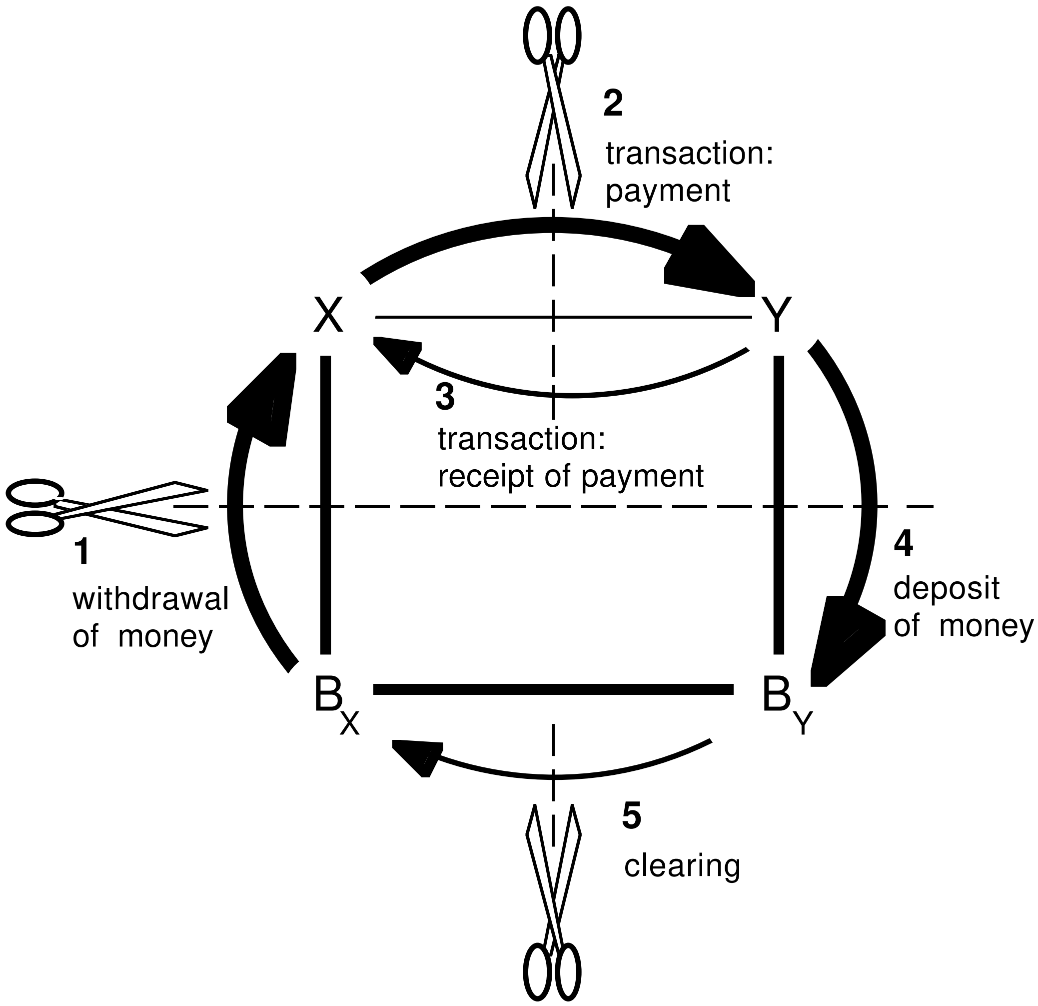


Figure 9: Payment circulation for mutually trusted tamper-resistant devices

Clearly, if chipcards with identity are used, e.g. a serial number (exchanged within each transaction of that chipcard) or an unique keypair of a PKCS [22] which never changes during the life cycle of the card, all actions made with a single chipcard are linkable in principle. Therefore, the cards should not have identities (or at least conceal them from their transaction partners) which implies that all authentic chipcards should have a key in common. This yields huge problems if someone else but the bank should be able to learn the key, but otherwise means no loss of security during authentication or money transfer, since secure authentication schemes for chipcards without identity exist [26 p. 351] and for money transfer only the transaction number, not the identity of chipcards is needed.
If all actions made by a single chipcard are unlinkable for banks without the cooperation of other parties, i.e. clearing anonymity is established in a weak form, it suffices to avoid generating an audit trail within the chipcard, or if an audit trail has to be maintained for security reasons, to make it unintelligible for banks without the cooperation of the chipcard's holder.

A main problem with chipcards for off-line money transfer which has not been considered so far is how to tolerate complete failures or losses which would imply a loss of money if no measures for fault tolerance were taken. Promising measures [36, 37], taking into account not only the user's security but also their unobservability, are based upon the generation of anonymous audit trails. A completely anonymous audit trail can be generated using a pair of pseudorandomly chosen transaction numbers for each transaction, instead of the amount of the transaction, the state of accounts, time and date etc. Each element of these pairs of transaction numbers is the next element of a sequence generated from a seed by a cryptographically strong pseudorandom number generator. Each chipcard uses exactly one seed chosen by its holder which defines its identity. As long as no seeds are revealed to the bank, this audit trail is completely worthless to it. But when someone looses his chipcard or someone's chipcards fails, he can reveal the seed of his chipcard to the bank. Now the bank can check whether no future element of that card's sequence is in any transaction number pair and therefore whether the most recent recovery point established and signed by but stored outside the chipcard, is still valid. Another possibility is to include the amount of each transaction in the audit trail, which allows to use more distant recovery points. This and different possibilities to guarantee that the bank receives all transaction number pairs, is described in full detail in [36].

3.2.3 Blindly Signed Digital Banknotes

Next, we describe a payment system invented by David Chaum [6, 7] realizing anonymity cut 3 only, i.e. independently whether anonymity cuts 1 and 2 are realized or not. This payment system establishes clearing anonymity by unobservability of each currency with itself, cf. section 3.1.2. Consequently, the digitally transferable currencies in this payment system are encoded messages which are transfered in the level of the clients, cf. figure 6. We refer to them as *digital banknotes* in analogy to banknotes in payment systems using cash, though this denotation is slightly inaccurate because these digital "banknotes" are rather digital bank cheques in standard values issued by the banks. Their full denotation *blindly signed digital banknotes* results from the fact that the bank signs the banknotes *blindly*, i.e. without actually seeing them, which is a precondition that their identity can be changed by the respective owner, cf. section 3.1.2.

A blindly signed digital banknote consists of a randomly chosen number n providing uniqueness and a redundant part enabling distinction of banknotes from other information. A banknote becomes valid when the bank signs it. Its nominal value is determined by the key used for the signature, i.e. different keys are attached to different nominal values.

To get a digital banknote, a client has to form a note number as described above and to submit it to the bank for signing. The note number, however, is encrypted in a special way. This disables the bank to recognize the note number whilst signing it, but enables the client to decrypt it after signing without invalidating the signature of the bank, i.e. he changes the identity of the note known (only in encrypted form) to the bank. (This procedure is given more formally at the end of this subsection.) After the bank has signed the digital banknote, its nominal value is withdrawn from the client's bank account.

If the banknote will sometime be deposited, the bank is not able to find out to whom of its clients the banknote was originally issued, so that anonymity cut 3 is realized, as shown in figure 10. As mentioned in section 3.1.2, it is necessary to have some supply of unspent notes to avoid linkability of withdrawal and deposit by unusual amounts or time. Clearly, all withdrawals and deposits on the same account are linkable.

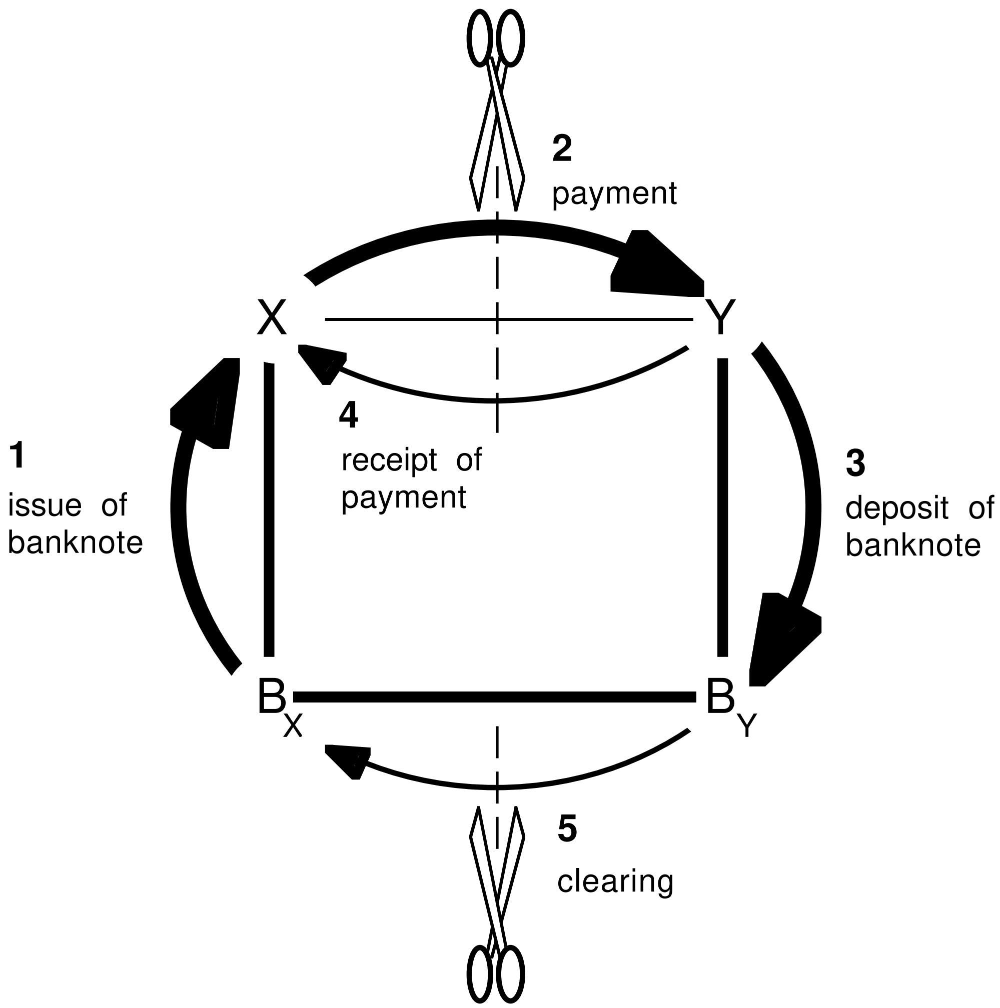


Figure 10: Payment circulation for blindly signed digital banknotes

Only the client who formed the note number is able to decrypt it after the bank signed it. By decrypting it, he can establish clearing anonymity (independently whether he is payer or payee), even against a coalition of the other and the bank. Clearly, the protocols for a payment vary slightly depending on who encrypts and later decrypts the banknote [8]. To allow payer and payee to verify clearing anonymity, both parties have to encrypt and later decrypt the banknote.[2] In the following, we will only consider in detail the variation, where the payer encrypts the note number. Therefore, it seems appropriate to outline the other variations.

If the payee forms the note number, it has to be sent to the payer first, who submits it to the bank for signing. The payer may submit the note number unchanged or encrypted by him a second time implying that he has to decrypt it before passing the signed note back to the payee. Since in any case only the payee has the information to decrypt the digital banknote yielding a valid banknote, the bank (even with the cooperation of the payer) cannot recognize the note number when it is submitted for deposit. For the same reason, the payee can be sure, that the digital banknote received from the payer has not already been deposited. If the payer would form the banknote (as described in full detail in the following), he and the bank together could of course link withdrawal and deposit, so that the payee could not verify that clearing anonymity is given.

---

[2] The dual is, that the bank and either payer or payee may cancel clearing anonymity due to certain circumstances. This can be achieved if the party who encrypts the note number gives the information necessary to decrypt it to the other.

With this payment system usually only on-line payments are possible (cf. section 3.1.3), independently of who formed the note number. In case the payee forms the note number, the payer has to submit the note number to the bank when the payment takes place. In the other case where the payer forms the note number, the payee has to ask the bank first if a received digital banknote has not already been deposited.
Only in case the payee is known to the payer in advance, the account number of the payee together with an expiry date of the reservation may be encoded into the note number, allowing the digital banknotes to be deposited only to the specified account during reservation. Before the expiry date the payee can verify the cover of the banknote locally. Afterwards the banknote may be deposited to arbitrary accounts, so that it may be used for other payments than the originally intended one, in case the payer changed his mind or the intended business comes to nothing.

Care has to be taken to prevent the banks from falsely claiming that a particular banknote has already been deposited. Therefore, the note numbers contain some secret information known to the payer only, who forms the note number. This information has to be revealed to the payee who discloses it to the bank only after he has received a confirmation that the banknote was not yet deposited. In order to prove that a banknote had already been deposited, the bank has to show this secret information.
The problem to form messages which contain secret information is one of the basic problems of cryptography and can be solved in various ways. The method applied in this case uses a public one-way-function f [13]. The note numbers are of the form $f(x)$ completed by redundancy for some argument x known to the payer only. The argument x is called *precursor* to the argument $f(x)$ and to the resulting note number. By definition of a one-way-function, it is impossible to compute the precursor out of the note number efficiently.
The following protocol shows withdrawal, payment and deposit of a digital banknote. This is an extended version of Chaum's original and still unpublished *bank-proof payer anonymity protocol* [8]. The extension is necessary to use this payment system for payments between two remote and anonymous parties. The substantial improvements are the introduction of an expiry date for each reservation and receipts of payments. Note that the secrecy and authentication mechanisms for the messages are not shown.

Signing *blindly* exploits a special property of the public key cryptosystem RSA and is explained in the following:

A keypair of RSA is denoted as (e_x, d_x) where e_x is the key for encryption and d_x the key for decryption. Encryption and decryption of a message m is denoted as $e_x(m)$ and $d_x(m)$, respectively. Encryption and decryption are inverse operations. The cryptosystem RSA has the additional property that both operations are commutative. Thus we have the following identities which hold for all keypairs and all messages m from the allowed message space:

$$d_x(e_x(m)) = e_x(d_x(m)) = m$$

The above mentioned special property of RSA is denoted as multiplicative homomorphism, i.e. for all keypairs (e_x, d_x) and all messages m_1, m_2 we have:

$$e_x(m_1 \bullet m_2) = e_x(m_1) \bullet e_x(m_2) \quad \text{and}$$
$$d_x(m_1 \bullet m_2) = d_x(m_1) \bullet d_x(m_2).$$

(Note, that the messages are represented in binary digits and are interpreted as binary numbers in the above operations.) The property that RSA is a multiplicative homomorphism is used to get *blind signatures* by multiplying the message m to be signed by a random "blinding factor" r encrypted with the encryption key e_x. The resulting product will be signed by applying the decryption key d_x. Thus, the signing party sees only $m \bullet e_x(r)$ which gives no clue about the message m. By applying d_x on the product, we have:

$$d_x(m \bullet e_x(r)) = d_x(m) \bullet d_x(e_x(r)) = d_x(m) \bullet r.$$

The blinding factor r can be removed from the resulting product by multiplication with its multiplicative inverse r^{-1} which yields the signed message $d_x(m)$.

A few special primitive operators are used:

x := random denotes the operation of choosing an argument x at random from the allowed message space.

x^* := special (x) denotes the operation to form x^* out of x by addition of redundancy, such that check (x^*) becomes true.

check (x) denotes a predicate that is true only if x is the result of a preceeding operation special (x).

$f(x)$ denotes the image of an argument x under f, a public one-way-function.

Prerequisite:
We assume that bank B uses the operator sign_B for signing its messages. The operator auth_B to check whether a message has been signed by B is publicly known. The bank uses a keypair (e_B, d_B) of the RSA public key cryptosystem exclusively to sign banknotes; d_B is kept secret, e_B is publicly known and determines the nominal value of a banknote signed with d_B.

[1] X forms a blinded note number and submits it to the bank for signing in step [1.5]:
[1.1] X : n' := random
[1.2] X : n := $f(n')$
[1.3] X : n^* := special (n)
[1.4] X : r := random
[1.5] X --- $n^* \bullet e_B(r)$ ---> B

Note:
n' is the secret information which prevents the bank from falsely claiming a particular note had already been deposited, since n' cannot be obtained from n.
n is the actual note number. Since n is formed under a one-way-function, attacks to obtain the banks signature on $n = f(n')$ when n' is given or vice versa are ineffective.
n^* is the note number extended by redundancy, e.g. $n^* = n, f(n)$. This will be used to prevent reservation of fake notes for deposit (cf. step [6] of the protocol).
r is a random "blinding factor" which disables the bank to recognize the note number n^* whilst signing it.

[2] The bank checks the authentication of the message received in step [1.5] and signs the submitted blinded note number. At the same time, the nominal value of the note is withdrawn from the account of X:
[2.1] B : "account of X covered?"
[2.2] B : "withdraw from account of X"
[2.3] B : "form $d_B(n^*) \bullet r = d_B$ ([1.5])"
[2.4] B --- $d_B(n^*) \bullet r$ ---> X
[2.5] B --- "STATEMENT OF ACCOUNT" ---> X

Note: The procedure in step [2.3] describes the bank's "blind signature" as explained in the preliminary remark at the begin of the protocol.

[3] *X* removes the blinding factor and checks whether he has received a valid banknote or not:
[3.1] X : $d_B(n^*)$:= [2.4] • r^{-1}
[3.2] X : e_B ([3.1]) =?= n^*

[4] *X* transfers the note to *Y*:
[4.1] X --- $d_B(n^*)$ ---> Y

[5] *Y* checks the signature on the note and submits it to the bank for deposit to one of his accounts:
[5.1] Y : n^* := e_B ([4.1])
[5.2] Y : check (n^*)
[5.3] Y --- $d_B(n^*)$ ---> B

[6] The bank verifies the validity of the submitted note:
[6.1] B : n^* := e_B ([5.3])
[6.2] B : check (n^*)
[6.3] B : "n^* already been cashed or reserved?"
[6.4] B : "register n^* as reserved for *Y* until *t*"
[6.5] B --- sign_B ("n^* RESERVED FOR *Y* UNTIL *t*") ---> Y

Note: Notes which are submitted for deposit but whose precursor to the note number has not been shown yet, will be registered as reserved for deposit. Therefore it is important that notes cannot be reserved without having the bank's signature which is checked in step [6.1] and [6.2]. If the note numbers n^* submitted for reservation are just random numbers not having any redundancy, it will be easy to reserve a multitude of notes by submitting randomly chosen numbers m. Each result $e_B(m)$ of encryption by e_B could be reserved. This would cause the banks a lot of unnecessary work and storage overhead, since the precursor to $e_B(m)$ cannot be shown (otherwise f would be no one-way-function).
In order to limit the time span of the above mentioned storage overhead and in order to prevent the bank from refusing forever to accept a submitted note argueing the note has already been reserved for someone else, an expiry date t is used. If the precursor to the note number is not shown until this expiry date, the reservation will be cancelled.
The specification of Y in step [6.5] may be any digital pseudonym of Y, e.g. the digital pseudonym under which Y is known to the bank as holder of an (anonymous numbered) account.

[7] *Y* checks the confirmation of reservation received in step [6.5] and forwards it to X:
[7.1] Y : auth_B ([6.5])
[7.2] Y --- [6.5] ---> X

[8] *X* checks the confirmation of reservation received from *Y* in step [7.2] and sends *Y* the precursor to the note number:
[8.1] X : auth_B ([7.2])
[8.2] X --- n' ---> Y

[9] Y verifies the precursor received from X in step [8.2] and sends a receipt of payment to X and the precursor to the bank:

[9.1] Y : special (f([8.2])) =?= n^*

[9.2] Y --- "RECEIPT OF PAYMENT" ---> X

[9.3] Y --- [8.2] ---> B

[10] The bank verifies the supplied precursor and deposits the note to the account of Y:

[10.1] B : special (f([8.2])) =?= n^*

[10.2] B : "n^* marked for Y?"

[10.3] B : "mark n^* as cashed"

[10.4] B : "deposit to account of Y"

[10.5] B --- "NOTE NUMBER n^* CASHED, STATEMENT OF ACCOUNT" ---> Y

If an anonymous payee refuses to give a receipt of payment to the payer (cf. step [9.2]), though the precursor has already been sent, then the payer can prove that he has made a payment since he can show the confirmation of the bank saying that the note has been reserved for the payee. Of course this serves as proof only before the expiration of reservation has elapsed. However, it does not prove whether the payer has sent the precursor to the note number to the payee or not. Therefore the same procedure descibed in subsection 3.2.2 for chipcards is used, i.e. the precursor to the note number is made known to the payee by publishing it (encrypted and therefore only readable for the intended payee) on an electronic blackboard. Thus, the payer is sure to get at least a substitutional receipt of payment issued by a court.

It should be mentioned that a slight modification of the basic format of blindly signed digital banknotes offers a more flexible handling when the banknotes should be deposited, concerning expiry date of reservation, deposit to intended payee, etc.. As already mentioned, there are various ways to form messages such that they contain secret information. Instead of using one-way-functions, an operator auth_x usable to authenticate messages signed by the operator sign_x (which has been randomly chosen) can be encoded in the note number. That way, the precursor to the note number can be replaced by a standard message signed with the operator sign_x. Instead of showing the precursor to the note number (which is effective only once) a standard message has to be signed (which can be done with a sequence of standard messages as well). Thus it is possible to have a sequence of precursors which are effective one after another.

3.2.4 Anonymously Transferable Standard Values

This payment system establishes clearing anonymity by unlinkability of different currencies, cf. section 3.1.2. It uses digitally transferable currencies called *anonymously transferable standard values* (which can be thought of as accounts upon which standard values are deposited [4]). The value (in standard denominations) is encoded in the *value number*. The anonymously transferable standard values are administered by the bank similar to anonymous numbered accounts, cf. subsection 3.2.1. The owners of these anonymously transferable standard values are known to the bank only by their digital pseudonyms which are unlinkable to anything else which is known to the banks. Payments are made by transfering the ownership of these values in the level of banks, cf. figure 6.

If a payment is made, the payee first has to choose a new digital pseudonym and to sign a declaration proving that he wants to become (under this newly generated identity) the owner of an anonymously transferable standard value currently held by the payer. The signed declaration can be checked by applying the corresponding auth-operator which is an older digital pseudonym under which the payee is known to the payer. Obviously, this older digital pseudonym should differ from the newly generated digital pseudonym to disable the bank from deriving undesired concatenations without the cooperation of the payer. To disable the payer, too, from linking pseudonyms without the cooperation of the bank, the newly generated pseudonym can be shown to him encrypted only with a public key of the bank. After the payer has received this declaration, he orders the bank to register the digital pseudonym of the payee, as specified in the declaration, as new owner of the anonymously transferable standard value held by him. The bank confirms execution of the order either to the payer who then forwards this confirmation to the payee, or to payer and payee. In any case the payer should receive a receipt of payment from the payee as soon as the payee has received the confirmation of execution. If the payee refuses to give a receipt of payment, the payer has sufficient evidence proving that a payment was made: the declaration of the payee and the confirmation of the bank.

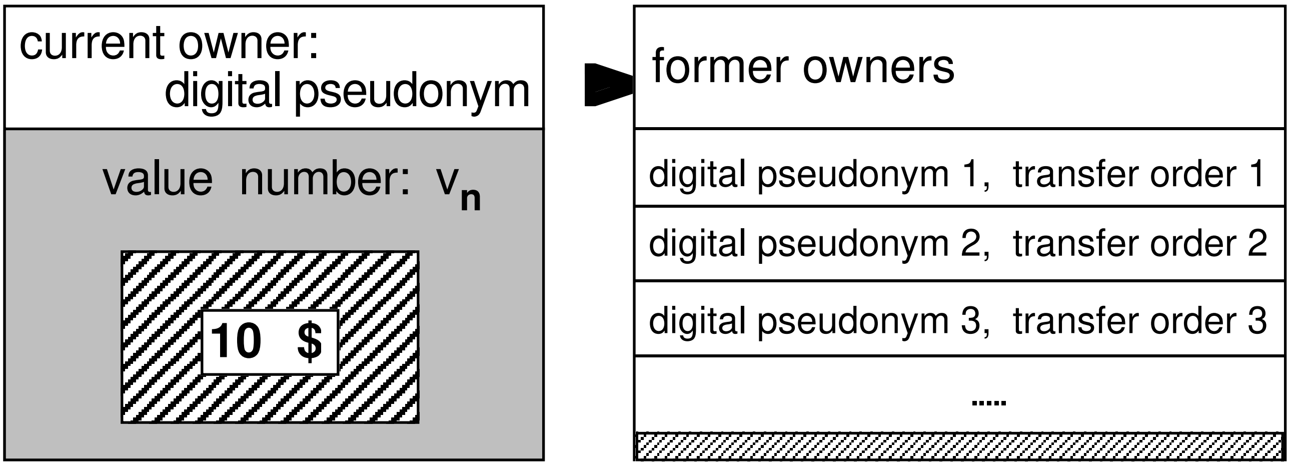


Figure 11: Anonymously transferable standard value

The protocol for the transfer of ownership of an anonymously transferable standard value from payer X to payee Y is as follows:

Prerequisite:
We assume that X, the current owner of an anonymously transferable standard value v_n, has chosen the operator sign_x to sign the transfer order when this value is to be transfered. Therefore auth_x, the corresponding operator to verify the authentication of the transfer order, is registered at the bank B administering this value as the digital pseudonym of the current holder.
The bank uses the operator sign_B for signing its messages. The operator auth_B to check whether a message has been signed by B is publicly known.
Messages from Y to X are signed by sign_{yx} and checked by auth_{yx}, which is a digital pseudonym of Y known to X.

[1] Y gives his consent to X to become the new owner of the anonymously transferable standard value v_n currently held by X. Y has to choose the operators sign_y and auth_y at random, which will be used for signing and verifying authentication of messages, respectively. The auth-operator is communicated to X:
    [1.1] Y --- sign_{yx} ("AGREEMENT TO REGISTER auth_y AS NEW OWNER OF v_n") ---> X

[2] X checks the authentication of the message received in step [1.1] and gives order to the bank to register auth_y, the randomly chosen digital pseudonym of Y, as new owner:
    [2.1] X : auth_{yx} ([1.1])
    [2.2] X --- sign_x ("REGISTER auth_y AS CURRENT OWNER OF v_n") ---> B

[3] The bank checks the authentication of the transfer order received in step [2.2] and executes it:
[3.1] B : auth_x ([2.2])
[3.2] B : "insert auth_x and [2.2] in the list of former owners of v_n"
[3.3] B : "register auth_y as current owner of v_n"
[3.4] B --- sign_B ("auth_y REGISTERED AS CURRENT OWNER OF v_n") ---> X
[3.5] B --- sign_B ("auth_y REGISTERED AS CURRENT OWNER OF v_n") ---> Y

[4] X checks the confirmation of transference from B received in step [3.4].
[4.1] X : auth_B ([3.4])

[5] Y checks the confirmation of transference from B received step [3.5] and sends a receipt of payment:
[5.1] Y : auth_B([3.5])
[5.2] Y --- sign_{yx} ("RECEIPT OF PAYMENT") ---> X

Two modifications of this protocol should be mentioned:
The bank could send the confirmation of transference to X only who then has to forward it to Y. This possibility has the advantage that in case the payer has not obtained a receipt of payment and has to prove that a payment was made, it is not necessary to ask the bank whether it has sent the confirmation of execution to the payee, since the payer is responsible for sending the confirmation of execution to the payee.
It is possible to save the expense of generating a new pair of operators (recall that the sign-operator stands for a digital signature) for each transfer of an anonymously transferable standard value: instead of auth-operators, which are taken as the digital pseudonyms of the owners, arguments under a public one-way function are used. It is appropriate to have only one public one-way function f implying that all digital pseudonyms are of the form $f(x)$ for some argument x. The precursor x to the digital pseudonym $f(x)$ can serve as authentication, since initially each precursor is only known to the holder of the pseudonym. Of course, the precursor can be used as authentication only once, but this is sufficient since a proof of authentication is required for the transfer order only. Thus, the signed transfer order of step [2.2] can be replaced by sending first the (unsigned) order of transference to the bank. Thereupon the payer should get a signed message from the bank confirming both, the digital pseudonym of the current, and the intentionally next owner. This is to prevent the bank from registering another digital pseudonym than the one which is specified in the (unsigned) transfer order. Only after receiving this message from the bank, the precursor to the current digital pseudonym is sent. Additionally, the message signed by the bank contains an expiry date to prevent it from forever refusing to accept an order to transfer the account. Now it is essential that the payer gets a confirmation of the timely receipt of the precursor from the bank, when the precursor to the current digital pseudonym has been shown to it. If the expiry date is chosen appropriately, the confirmation can be enforced, if necessary, by the help of a court, since with the above mentioned signed message of the bank the payer has sufficient evidence that the transfer order had been accepted. In this case the payer has to present the precursor to the court in time. The court forwards it to the bank and forces the bank to issue a corresponding confirmation.

Obviously, this payment system provides both, relation anonymity and clearing anonymity, cf. figure 12. However, it should be mentioned that whenever the ownership of such a value is transfered, the digital pseudonyms of the current owner X and the intentionally next owner Y of the anonymously transferable value become linkable to the digital pseudonyms under which they are known to each other if one of them has the cooperation of the bank. This linkability can be avoided if special measures (cf. section 3.4) are taken.

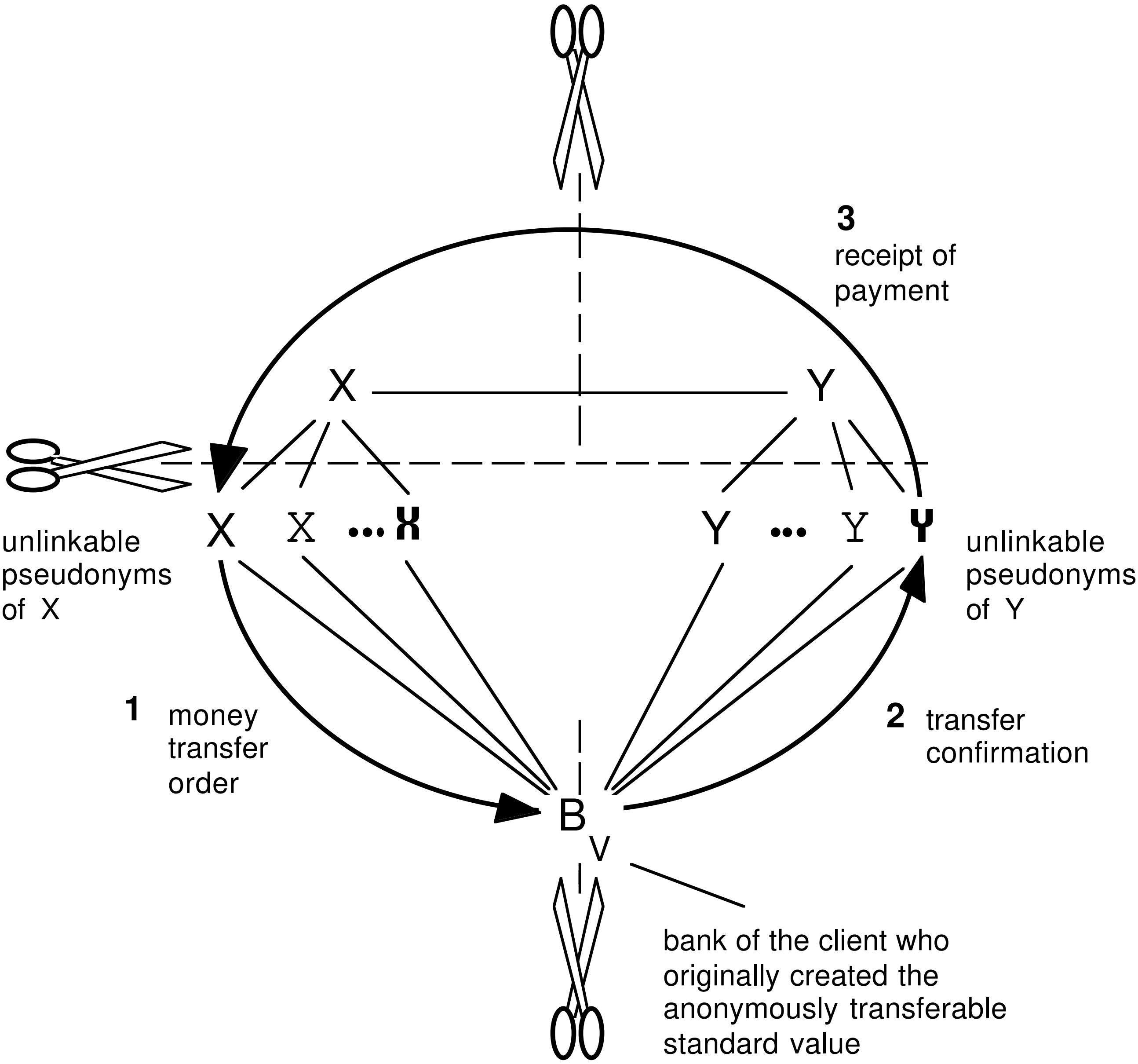


Figure 12: Payment circulation for anonymously transferable standard values

The further characteristics with respect to the possibility for off-line payments of this system are:

* Off-line verification of pre-arranged off-line payments is possible: the payer reserves some anonymously transferable standard values exclusively for payments to specified digital pseudonyms of the payee until an expiry date. The payer gets a corresponding confirmation signed by his bank which he can give to the payee, who checks whether he is the owner of the specified pseudonyms and whether the expiry date does not exceed too early. If both conditions are satisfied, he can be sure that every transfer of ownership signed by the payer is covered.
* Off-line payments to arbitrary payees cannot be verified by the payee without using the network.

3.3 Comparison of Anonymously Transferable Standard Values and Blindly Signed Digital Banknotes

In this section we compare blindly signed digital banknotes and anonymously transferable standard values, since both systems provide the desired clearing anonymity without using tamper-resistant devices, and the measures used by each system to establish clearing anonymity are in some way duals of each other, cf. section 3.1.2.

The following fundamental criteria will be discussed:
1. Unobservability
2. Security
3. Efficiency

1. Anonymously transferable standard values can provide nearly maximal unobservability of clients, since the holder of such a value may be completely anonymous and a payment can be made unlinkable to any other action of the same person if anonymity cut 1 is realized completely. With blindly signed digital banknotes, however, first of all linkability between withdrawal and deposit of a digital banknote can be made impossible. But even if additionally anonymity cut 1 and 2 are realized completely, all actions involving the bank accounts are still linkable, merely the payment-relations are unobservable for the banks.
Combining the measures of both systems, as described in detail in section 3.4, yields even more unobservability of clients as any single system yet described.

2. Blindly signed digital banknotes rely on the special property of RSA to be a multiplicative homomorphism (cf. subsection 3.2.3). There is no other secure signature system known which is a multiplicative homomorphism, too, nor are other ways known to implement *blind signatures* without using that property.[3] RSA, however, is today not provably as secure as factoring is difficult, but is clearly broken if factoring the modulus is feasible. Another possibility to break the system might be a chosen signature attack. Since anonymously transferable standard values may be implemented using any digital signature system and there are efficient digital signature systems which are provably as secure as factoring is difficult even against a chosen signature attack [21, 20], today anonymously transferable standard values must be considered to be more secure. Since any signature system usable to sign messages *blindly* is a signature scheme, the security of blindly signed digital banknotes can never exceed the security of anonymously transferable standard values.
Another class of cryptographic systems can be used as well: By letting everybody be his own key-generation center, any identity-based signature scheme [17] can trivially be used as a normal signature scheme. This is appropriate, if it yields superior performance to the other known signature schemes which are provably as secure as factoring is difficult even against a chosen signature attack.

3. The cost per transaction is approximately the same for both payment systems. This can be seen from the table below where the storage and processing requirements per standard value transaction for both payment systems are listed. In both systems we distinguished whether the payer uses a digital signature or a precursor for authentication. Note that for clearness, the processing requirements do not include the verification of authentication. It should be mentioned that a blind signature requiring an RSA exponentiation takes about 25 times the processing of a "normal" signature using the Fiat-

[3] This was true at the time we wrote this report (spring 1987). In the meantime, universal translation techniques have been invented [38, 39] which can be used to implement blind signatures using an arbitrary signature system [40]. But the expense of this universal translation techniques seems to be extraordinarily high.

Shamir scheme as stated in [17]. However, the expense of the necessary application of cryptographically strong one-way functions has admittedly been neglected there and would cause a substantial expense probably in the order of a RSA exponentiation. But using unexpensive DES-like one-way functions would undermine any of the known proofs of security.

Payment / Cost		anonymously transferable standard values		blindly signed digital banknotes	
		authentication mechanism		authentication mechanism	
		precursor	digital signature	precursor	digital signature
bank	storage	precursor to pseudonym	signature on transference order	precursor to note number	signed message proving ownership
bank	processing	two signatures (acceptance confirmation, transference confirmation)	one signature (transference confirmation)	two signatures (reservation confirmation, cashing confirmation) and one RSA exponentiation (blind signature)	
client	storage	confirmation concerning registration of pseudonym		precursor to note number	secret key for signing message proving ownership
		precursor to pseudonym	secret key for signing transference order		
client	processing	evaluation of one-way function	generation of corresponding signing and authentication operators	generation of blinded note, especially one RSA exponentiation of the blinding factor	
				evaluation of one-way function	generation of corresponding signing and authentication operators

Table: Cost (storage and processing requirements) per standard value transaction

In summary, it has been shown that anonymously transferable standard values are better (from today's point of view) or at least as good (if breaking RSA is proven to be as hard as factoring) as blindly signed digital banknotes with respect to all desiderata given in [9, 10].

A problem arising from the extent of anonymity provided by anonymously transferable standard values, which has not been considered so far, is that the possession of a person cannot be located anymore (e.g. to collect progressive tax rates, etc.). Blindly signed digital banknotes do not have this property. As already shortly mentioned in section 3.2.3, various protocols enable the bank in cooperation with either the

1. payer or the
2. payee or
3. both together or
4. either party

to trace the origin and destination of a payment in order to identify both parties [8]. However, this can also be achieved if anonymously transferable standard values are used. The key idea is to use a *mixing transformation* whereby a set of anonymously transferable standard values with the same nominal values is randomly transfered from one set of digital pseudonyms to another. The pseudonyms of each set may be either all anonymous or all nonanonymous ones. For nonanonymous pseudonyms, the bank knows the identity of the holder and testifies this by its signature. Then the above listed variations to trace the origin and destination of a payment with blindly signed digital banknotes may be simulated, as follows:

1. The payer uses an anonymous pseudonym to transfer the anonymously transferable standard value but transfers it only to a nonanonymous pseudonym.
2. The payee uses an anonymous pseudonym to receive the anonymously transferable standard value but accepts a transfer only from a nonanonymous pseudonym.
3. Payer and payee use anonymous pseudonyms.
4. Payer and payee use anonymous pseudonyms, but each party lets the other party sign a declaration with a nonanonymous pseudonym saying that the anonymous pseudonym used is theirs.

The natural way to implement a mixing transformation is to use MIXes (cf. section 2.1), which can yield complexity theoretic anonymity only. Based on the DC-network (cf. section 2.1), more complicated and costly solutions providing information theoretic anonymity are conceivable. In any case these mixing transformations and the protocols as a whole only use normal digital signatures as cryptographic transformations for security. Therefore, they can be made as secure as factoring is difficult if suitable signature systems are used.

3.4 Combinations and Interest

It is clear from section 3.1.1 that it is appropriate to unite the system realizing anonymity cut 2 only (anonymous numbered accounts) with the one realizing anonymity cut 3 only (blindly signed digital banknotes) to get both relation and clearing anonymity. Further and more elaborate combinations are described in the following two subsections.

3.4.1 Combinations To Get Maximal Anonymity

In section 3.1, we started the discussion about digital payment systems enabling unobservability of clients in view of the fact that the bank can clearly link all actions involving an account. Then we showed two dual possibilities by which the clients can establish the desired clearing anonymity. These dual possibilities are unlinkability of different currencies and unlinkability of each currency with itself. Obviously, a payment system which combines both possibilities will provide a maximum of clearing anonymity, as shown in figure 13.

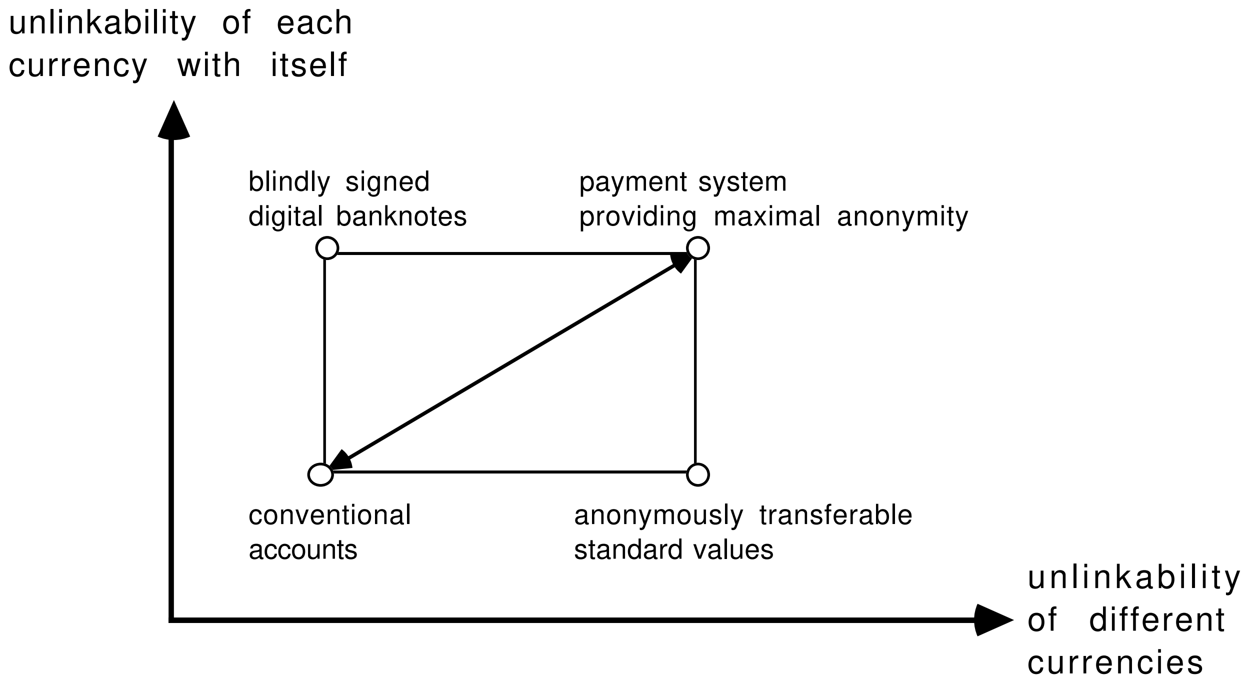


Figure 13: Combination to get maximal anonymity

In order to obtain a payment system which provides maximal anonymity, the measures by which the payment systems with blindly signed digital banknotes and anonymously transferable standard values, respectively, establish clearing anonymity have to be combined suitably. But how should such a combination look like?

Consider the case that anonymously transferable standard values are used for a payment. It is quite obvious, that the digital pseudonym under which a client is known to the bank as the holder of an anonymously transferable standard value is linkable to the digital pseudonym under which he is known to his transaction partner if bank and transaction partner collude. This would cause no harm as long as the digital pseudonym under which a client is known to the bank as the holder of an anonymously transferable standard value is used for one transaction only. Then it cannot be linked to anything else and each holder of an anonymously transferable standard value must clearly be known to the bank by a digital pseudonym anyway. As a matter of fact, this digital pseudonym is however used for a second transaction, that is, when the payee wants to use the received anonymously transferable standard value for another payment. To unlink the two transactions described above, clients should be able to change the identity of their digital pseudonyms known to the bank the same way as they can with blindly signed digital banknotes. This can be done by using the same special measure [7] which is applied to hinder the credentials issuing organizations from deriving undesired concatenations (cf. section 2.2) and which is based on blind signatures as well.

3.4.2 Combinations Enabling Off-line Payments Even Without Absolutely Tamper-resistant Devices

Obviously, the usability of digital payment systems increases when they can be used not only for secure on-line but also for secure off-line payments. Therefore, it is appropriate to use two systems side by side, namely

* the only one enabling off-line payments in general (tamper-resistant devices) but in a modified form for reasons of damage confinement in case the devices are not as tamper-resistant as supposed, and
* one not depending on tamper-resistant devices for on-line payments of greater security.

A modification of tamper-resistant devices cannot only consist in restricting the maximum value of off-line payments since in cases the tamper-resistance is severely injured, e.g. if the secret parameters could be derived by recording and decoding the electromagnetic radiation, any restricting measure cannot limit the damage. It has to be made sure that any damage arising when someone has broken the mechanism of the tamper-resistant device can be ascertained quickly and that the knowledge of the secret parameters stored within the device is not sufficient to produce an unlimited amount of money. Both can be achieved if the devices contain e.g. blindly signed digital banknotes. Then their function is restricted to the storage and transfer of digital money, which will be erased from the storage of the device as soon as it is spent. Yet their main purpose is to guarantee that each banknote is spent only once. Then in case someone breaks the mechanism of the tamper-resistant devices, he can only cheat by spending banknotes several times. This fraud is discovered when somebody tries in vain to deposit a banknote which has been deposited before. As in off-line payments anonymity cut 1 is not completely realized anyway (because the clients see each other), one can try to trace the origin of the duplicated banknote. To increase the chance of discovery, the number of off-line transactions with each banknote should be restricted to one.

3.4.3 Interest

Obviously, in any digital payment system where banks or tamper-resistant devices administer the money, interest can be paid without any technical problems. Special methods even allow to obtain interest for digital banknotes [8].

4 Value Exchange Systems Enabling Security and Unobservability

The main problem arising in value exchange over a network, e.g. in the exchange of digital money against other valuable information, is the lack of simultaneity of the exchange, yielding a temporary advantage for one of the business partners. Thus, if he stops communication prematurely, a claim against him will remain valid, which should be called in if necessary.

Solutions for the simultaneity problem developed for signing contracts [15, 1] are not suitable for value exchange in general, since one cannot define that e.g. a message only has a meaning if it is signed n times (with prefixes 1 through n) whereas one can define that a contract is only binding if it is signed n times. Goldreich's solutions of the concurrent identification problem [18, 19] cannot be adapted either, because both solutions presuppose a fixed identity of persons which can be established and approved beforehand. This is in contrast to the value exchange, where new values (e.g. new information, a new cheque, etc.) are to be exchanged.

In the following, we describe two approaches to enable unobservable value exchange which is secure against fraud. Both approaches use third parties in order to bypass the simultaneity problem.

4.1 Security by Identification of Perpetrators in Case of Detected Fraud

This concept guarantees security by identification of debtors who refuse to pay, thereby enabling availability of all their property for satisfaction of their creditors as in the non-anonymous case.

To allow identification under these circumstances, all parties concerned with the business must have identified themselves against a trusted and non-anonymous third party, who gives them a credential which proves that the holder of a particular digital pseudonym can be identified if necessary. This is an example for granted authentication, since before any business is done, the participants have to authenticate themselves mutually by means of these credentials. Later on business proceeds as usual.

For authentication a single trusted third party [22] (who may be different for different participants as shown in figure 14) or a chain of trusted third parties [5] may be used. The second method has the property that a participant can be identified only if all authenticating third parties are cooperating. It can be extended to tolerate faults or sabotage from a few third parties with the techniques described in [29]. For brevity, we only describe the first method.

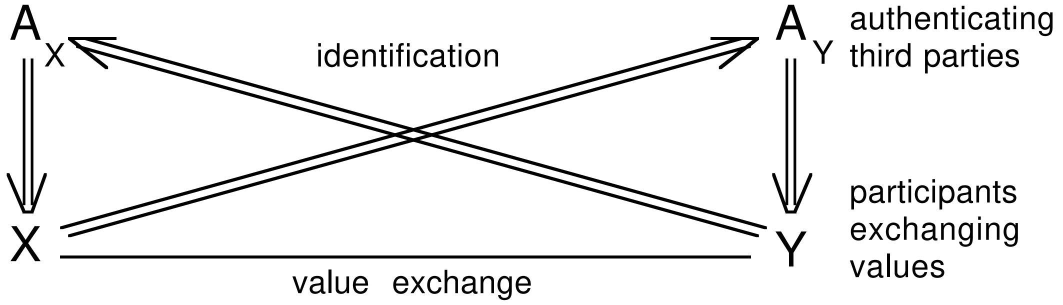


Figure 14: Security by authenticating third parties

In case that one participant complains (falsely or correctly) to the authenticating third party of his business partner that this "bad guy" has broken off the communication prematurely, he first has to submit signed messages of his partner to show that a communication between him and the accused existed. After this has been verified, the accused is requested to continue the communication. If he refuses, he will be identified by his authenticating third party and examination proceeds as in the non-anonymous case.
The worst thing that can happen if someone is falsely accused to have broken off communication prematurely is that he is obliged to send the rest of the messages again. However, this time the messages are sent via the authenticating third party to the complainant. Thus, the complainant cannot claim again falsely that he has not received the messages. By charging a small fee for each complaint to the authenticating third party, abuse can be reduced to a tolerable extent.

The advantages (+) and disadvantages (–) of this concept are:

- (+) Normally, no active participation of the authenticating third party is needed to exchange the values.
- (–) Who should control that the identities of the participants are revealed by the authenticating third party only in the assigned cases?
- (–) Uncovered claims may arise, since it is never verified that the participants are able to pay and identification of such participants does not help very much.

4.2 Security Between Absolutely Anonymous Parties by Active Participation of a Trustee

The second approach to security and anonymity for value exchange does not have the disadvantages of the first one. This is achieved by active participation of a non-anonymous trustee *T* by whom the value exchange between the anonymous parties *X* and *Y* is executed, as shown in figure 15.

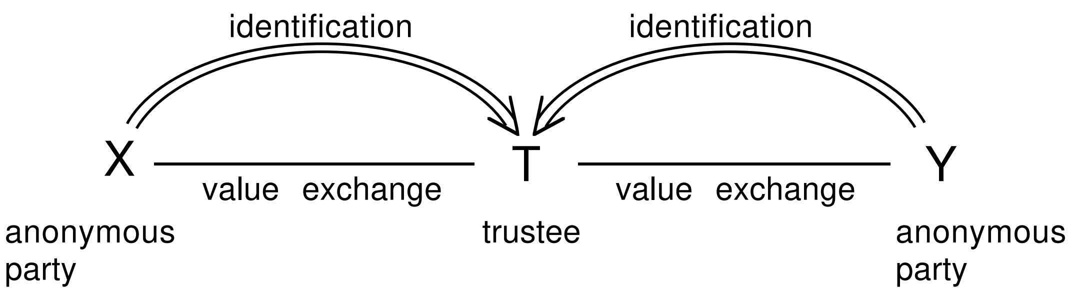


Figure 15: Security by an active trustee

To see how this works, we consider the frequently appearing case that some ware, e.g. useful information (from *Y*) is exchanged against money (from *X*). The case that the values to be transfered are both wares which the trustee should not or cannot inspect can then be solved by splitting the transfer in two parts "ware against money" and "money against ware".
Before any value exchange takes place, the anonymous parties *X* and *Y* inform the trustee what they wish to exchange and agree upon form and extent of the trustee's service concerning the value exchange. By this, *X* and *Y* who are both known by their digital pseudonyms only, are able to control the actions of the trustee and bring him into court if he does not adhere to the agreement. Then

X sends the money and *Y* the ware to *T*, who checks whether what he receives is according to what has been agreed upon. According to the result of this check, money and ware are passed on to the intended recipients or the exchange is broken off and money and ware are returned to their respective senders together with a justification for the breaking off. In order to make a coalition between the trustee and *X* against *Y* ineffective, the ware should not be sent till the trustee has confirmed to *Y* that he has received the money from *X*. Otherwise it would be possible for the trustee to return the ware to *Y* with the justification that *X* has not sent the money while actually sending a copy of the ware to *X* in secret. The situation is shown in figure 16. Another possibility is to pass ware and money to the trustee concurrently, but let *Y* choose the trustee, so that coalitions of *X* and the trustee are very unlikely.

X <--> Y
agreement about the value exchange

X <---------------------------> T <---------------------------> Y
agreement specifying the trustee's service | agreement specifying the trustee's service

X ----------------------------> T
money

T ----------------------------> Y
confirmation of reception of money

T <---------------------------- Y
ware

ascertainment and prevention of frauds by X and Y

T checks whether ware and money are according to the agreement

X <---------------------------- T ----------------------------> Y
ware | money

ascertainment of frauds by T; prosecution if the case should arise

Figure 16: Procedure of value exchange by an active trustee

In cases that the trustee is not or shall not be able to check the transmitted ware (i.e. for reasons of privacy), *X* and *Y* may agree upon a complaint time during which the trustee retains *X*'s money but passes the ware from *Y* to *X* without checking it. If *X* complains that what he has received does not meet his expectations (or precisely: the agreement between *X* and *Y* specifying ware and price), he may forbid *T* to pass the money to *Y* and has to prove that the ware is indeed faulty. If an eventually called court works fast enough, no appreciable damages, i.e. loss of interest, will arise.

If *X* claims that he has not received the ware, it can be sent again with negligible costs for *T*, since information can be copied nearly for free. If we assume that the payment is made in two parts, i.e. *X* pays *T* first and *T* pays *Y* afterwards, it is also impossible to cheat by falsely claiming not to have received the money, since with all payment systems allowing unobservable payments (cf. chapter 3) the payer can prove that he has made a payment.

The advantages (+) and disadvantages (–) of this concept are:

- (–) The trustee always has to be active. But since the PTTs may offer the additional service to act as trustees using some standard contracts for value exchange, there is no performance problem.
- (+) No confidence concerning the trustee's service in the value exchange is necessary, since the trustee can be controlled by the participants *X* and *Y* and brought into court if frauds or irregularities are ascertained.
- (+) All charges from *X* to *Y* or vice versa may be called in by recourse to the values deposited to the trustee.
- (+) The participants *X* and *Y* may be completely anonymous.

5 What Needs To Be Shortly Mentioned and Persistently Done

Of course the existence of a payment system enabling unobservability does not necessarily mean that it must be used for all kinds of payments, but that each client determines for himself how much unobservability of his payments he wishes, unless this freedom of choice is restricted by law. For instance, such restrictions could be decreed because collecting progressive taxes will become even more problematical than today (see below) when people receive several incomes under different pseudonyms. But also for this problem solutions enabling unobservability using the credential mechanism are conceivable. The other way round, such a payment system may be a strong impetus to rethink our overcomplicated tax system – simplification and automation preventing fraud would both be very welcome (not to all people, but to a, hopefully, vaste majority). An advantage of an anonymous payment system can be that it may be more generally accepted as substitute for cash, so that cash could even be abolished. This would make theft more difficult, since active cooperation of the victim is required. Black markets and illicit work would possibly thin out, since sales and income tax could be collected automatically whenever the payment system is used and/or possession of money could be taxed.

So far, our arguments mainly used the interests of individuals. But the state should have an interest in payment systems enabling unobservability as well, since enforcing its own policy concerning security and privacy is possible only then. Without the use of such payment systems, national sovereignty (at least concerning privacy) would be heavily endangered whenever computer equipment containing secret "know how" is used, especially when bought from abroad.

Before these payment systems and other appropriate anonymous transaction protocols can be used in practice, the legal framework has to be adapted appropriately [31]. But this would be necessary for any digital payment and transaction system anyway.

Last but not least we want to remind that today the state bears the full risk of the cash system without installing a nearly perfect supervision and control system encircling everyone everywhere. If banks de facto replace cash, they have to take over this risk from the state without installing payment systems with nearly perfect supervision and control to quickly locate the weak spot if the system should be misused or broken. As a matter of fact, most digital payment systems used today, e.g home banking via BTX (= interactive videotex in Germany) are immature and unsecure from a technical point of view. The circumstance that great frauds were rare (or at least not reported) up to now and could have been covered from the insurances [32], together with the missing awareness of the clients that their privacy is extremely endangered by these systems (and cannot be covered by any insurance) should not induce anybody to hold on to insufficient payment systems.

So let us (in contrast to the NSA, cf. [24, 23], but in the future hopefully together with the NBS again [12]) strive to further advance technical security from an art to a science by proving all cryptosystems and protocols used and validating all devices (including tamper-resistant ones) publicly -- and if this is not (yet) possible or deemed to be too expensive, then let us build no digital payment system rather than a bad one!

Acknowledgements

We are indebted to Dr. Klaus Dittrich, Dr. Klaus Echtle, Prof. Winfried Görke, Birgit Pfitzmann, and Michael Waidner for lots of valuable suggestions and to the German Science Foundation (DFG) for financial support.

References

[1] Michael Ben-Or, Oded Goldreich, Silvio Micali, Ronald L. Rivest: A Fair Protocol for Signing Contracts; LNCS 194, Automata, Languages and Programming (ICALP), 12th Colloquium, Nafplion, Greece, July 15-19, 1985, Winfried Bauer (ed.), Springer Verlag, Heidelberg, pp. 43-52

[2] Manuel Blum, Shafi Goldwasser: An Efficient Probabilistic Public-Key Encryption Scheme Which Hides All Partial Information; Advances in Cryptology, Proceedings of Crypto 84, A Workshop on the Theory and Application of Cryptographic Techniques, August 19-22, 1984, University of California, Santa Barbara, Edited by G. R. Blakley and David Chaum, Lecture Notes in Computer Science LNCS 196, Springer-Verlag Heidelberg, 1985, pp. 289-299

[3] Manuel Blum, Silvio Micali: How to Generate Cryptographically Strong Sequences of Pseudo-Random Bits; SIAM J. Comput. Vol. 13, No. 4, November 1984, pp. 850-864

[4] Holger Bürk: Digitale Zahlungssysteme und betrugssicherer, anonymer Wertetransfer; Studienarbeit am Institut für Informatik IV, Universität Karlsruhe, April 1986

[5] David Chaum: Untraceable Electronic Mail, Return Addresses, and Digital Pseudonyms; Communications of the ACM Vol. 24, No. 2, February 1981, pp. 84-88

[6] David Chaum: Blind Signatures for untraceable payments; Advances in Cryptology, Proc. of Crypto 82, D.Chaum, R.L.Rivest, A.T.Sherman (eds.), Plenum Press, New York, 1983, pp. 199-203

[7] David Chaum: Security without Identification: Transaction Systems to make Big Brother Obsolete; Communications of the ACM Vol. 28, No. 10, October 1985, pp. 1030-1044

[8] David Chaum: Privacy Protected Payments. Unconditional Payer and/or Payee Untraceability; received October 31, 1986

[9] David Chaum: Highly Secure But Untraceable Transactions; IFIP/Sec´86, Pre-prints of the Fourth International Conference and Exhibition on Computer Security: "Information Security: The Challenge", Monte Carlo, December 2-4, 1986, A. Grissonnanche (ed.), pp. 264-266

[10] David Chaum: A Convenient, Economical, Auditable, Secure, and Yet Untraceable Payment System; Proc. of SECURICOM 87, 5th Worldwide Congress on Computer and Communications, Security and Protection, Paris, March 4-6, 1987, pp.1-6

[11] Fred Cohen: Computer Viruses – Theory and Experiments; Proc. of the 7th National Computer Security Conference, 1984, National Bureau of Standards, Gaithersburg, MD; USA, pp. 240-263; Computers & Security, North-Holland, Vol. 6, No. 1, February 1987, pp. 22-35

[12] Computer Security Act Stresses Encryption Standard; Communications of the ACM Vol. 30, No. 6, June 1987, p. 572

[13] D. W. Davies, W. L. Price: Security for Computer Networks, An Introduction to Data Security in Teleprocessing and Electronic Funds Transfer; John Wiley & Sons, Chichester, New York, 1984

[14] Dorothy Denning: Cryptography and Data Security; Addison-Wesley Publishing Company, Reading, Mass.; 1982

[15] Shimon Even, Oded Goldreich, Abraham Lempel: A Randomized Protocol for Signing Contracts; Communications of the ACM Vol. 28, No. 6, June 1985, pp. 637-647

[16] S. Even, O. Goldreich, Y. Yacobi: Electronic Wallet; 1984 International Zurich Seminar on Digital Communications, March 6-8, Zurich, Switzerland, IEEE, pp. 199-201

[17] Amos Fiat, Adi Shamir: How To Prove Yourself: Practical Solutions to Identification and Signature Problems; Advances in Cryptology – CRYPTO '86, August 1986, A. M. Odlyzko (ed.), Lecture Notes in Computer Science LNCS 263, Springer-Verlag, Heidelberg, 1987, pp. 186-194

[18] Oded Goldreich: On Concurrent Identification Protocols; Laboratory for Computer Science, Massachusetts Institute of Technology, MIT/LCS/TM-250, December 1983

[19] Oded Goldreich: On Concurrent Identification Protocols; Advances in Cryptology, Proceedings of EUROCRYPT 84, A Workshop on the Theory and Application of Cryptographic Techniques, April 9-11, 1984, Paris, France, T. Beth, N. Cot, I. Ingemarsson (eds.), Lecture Notes in Computer Science LNCS 209, Springer-Verlag Heidelberg, 1985, pp.387-396

[20] Oded Goldreich: Two Remarks Concerning the Goldwasser-Micali-Rivest Signature Scheme; Advances in Cryptology – CRYPTO '86, August 1986, A. M. Odlyzko (ed.), Lecture Notes in Computer Science LNCS 263, Springer-Verlag, Heidelberg, 1987, pp. 104-110

[21] Shafi Goldwasser, Silvio Micali, Ronald L. Rivest: A "Paradoxical" Solution to the Signature Problem; 25th Annual Symposium on Foundations of Computer Science, IEEE Computer Society, October 24-26, 1984, pp. 441-448; revised and extended version entitled "A digital signature scheme secure against adaptive chosen-message attacks" appeared in SIAM J. Comput. Vol. 17, No. 2, 1988, pp. 281-308

[22] S. Herda: Authenticity, Anonymity and Security in OSIS. An Open System for Information Services; Proc. 1. GI Fachtagung Datenschutz und Datensicherung im Wandel der Informationstechnologien, P.P.Spies (ed.), Informatik-Fachberichte Band 113, Springer-Verlag Heidelberg 1985, pp. 35-50

[23] John Horgan: NSA explains encryption program to IEEE Privacy Subcommittee; The Institute, IEEE, Vol. 10, No. 9, September 1986, p. 2

[24] Gina Kolata: NSA to Provide Secret Codes; Science Vol. 230, October 1985, pp. 45-46

[25] H. Maurer, N. Rozsenich, I. Sebestyen: Videotex without "Big Brother"; IIG, Universität Graz, Bericht F128, January 1984

[26] Carl H. Meyer, Stephan M. Matyas: Cryptography - A New Dimension in Computer Data Security; John Wiley & Sons, 1982

[27] George Orwell: 1984;

[28] Andreas Pfitzmann: A switched/broadcast ISDN to decrease user observability; 1984 International Zurich Seminar on Digital Communications, March 6-8, Zurich, Switzerland, IEEE, pp. 183-190

[29] Andreas Pfitzmann: How to implement ISDNs without user observability - Some remarks; Fakultät für Informatik, Universität Karlsruhe, Interner Bericht 14/85

[30] Andreas Pfitzmann, Michael Waidner: Networks without user observability -- design options; Eurocrypt 85, April 1985, F.Pichler (ed.), Lecture Notes in Computer Science LNCS 219, Springer-Verlag, Heidelberg, 1986, pp. 245-253; revised and extended version entitled "Networks without user observability" appeared in Computers & Security, North-Holland, Vol. 6, No. 2, April 1987, pp. 158-166

[31] Birgit Pfitzmann, Michael Waidner, Andreas Pfitzmann: Rechtssicherheit trotz Anonymität in offenen digitalen Systemen; Computer und Recht (CR), Verlag Dr. Otto Schmidt KG, Köln, 3. Jahrgang, Oct., Nov., Dec. 1987, Hefte 10, 11, 12, pp. 712-717, 796-803, 898-904

[32] Roy Martin Richards, Jan L. Guynes: A Strategic Plan for Reducing Consumer Anxiety About EFTS Security; acm SIGSAC Security, Audit&Control Review Vol. 4, No. 2, Spring 1986, pp. 4-8

[33] Andrew S. Tanenbaum: Computer Networks; Prentice-Hall, Englewood Cliffs, N. J., 1981

[34] Ken Thompson: Reflections on Trusting Trust; Communications of the ACM, Vol. 27, No. 8, August 1984, pp. 761-763

[35] Victor L. Voydock, Stephen T. Kent: Security Mechanisms in High-Level Network Protocols; acm computing surveys Vol. 15, No. 2, June 1983, pp. 135-171

[36] Michael Waidner, Birgit Pfitzmann: Verlusttolerante anonyme elektronische Brieftaschen; Fakultät für Informatik, Universität Karlsruhe, Interner Bericht 1/87, 1987

[37] Michael Waidner, Birgit Pfitzmann: Verlusttolerante elektronische Brieftaschen; Fault-Tolerant Computing-Systems, Sept. 9-11, 1987, Bremerhaven, Informatik-Fachberichte 147, Springer-Verlag, Heidelberg, pp. 36-50, revised version appeared in DuD, Datenschutz und Datensicherung, Vieweg Verlag, Braunschweig, Heft 10, October 1987, pp. 487-497

[38] G. Brassard, D. Chaum, C. Crépeau: Minimum Disclosure Proofs of Knowledge; Draft July 1987

[39] D. Chaum, I. B. Damgård, J. van de Graaf: Multiparty Computations ensuring privacy of each party´s input and correctness of the result; Crypto '87, LNCS 293, Springer-Verlag, Berlin 1988, pp. 87-119

[40] Michael Waidner: Betrugssicherheit durch kryptographische Protokolle beim Wertetransfer über Kommunikationsnetze; Arbeitsbericht über das DFG-Projekt "Betrugssicherheit" September 1986 bis April 1988; Interner Bericht 7/88, Fakultät für Informatik, Univ. Karlsruhe