コンピュータセキュリティ　18－23
(2002.　7.　19)

期限付き匿名貸し出しプログラムにおける一考察

繁富　利恵†　　大塚　　玲††　　小川　貴英†　　今井　秀樹††

† 津田塾大学 〒187-8577 東京都小平市津田町 2-1-1
†† 東京大学生産技術研究所 〒153-8505 東京都目黒区駒場 4-6-1
E-mail: †{m01shige,ogawa}@tsuda.ac.jp, ††otsuka@imailab.iis.u-tokyo.ac.jp, †††imai@iis.u-tokyo.ac.jp

あらまし　現在、情報の電子化が進んでいることは周知の事実である。こういった電子情報は、非電子情報に比べ、情報の収集、検索や統合などが容易となるため、蓄積情報に対する配慮がより重要となっている。こういった中、貸し出しサービス (図書館、レンタルビデオ、借金など) は情報の電子化が非常に進んでいるサービスの一つである。また、貸し出しサービスにおける情報は、個人の趣味指向だけではなく、生活レベルさえも推測できるような情報を含んでいる。つまり、貸し出しサービスの情報の管理者は貸し出し情報を手にいれることができるだけでなく、各個人の個人的な情報を手にいれることもできる。また、情報が全て電子化されているため一度上記の情報がユーザの意図に反して貸出機関から流出してしまうと、電子情報の持つ特徴により、サービス利用者に関連する他の情報との統合が進む可能性が高く、このような電子化された情報が持つプライバシ問題は大きい。こういった問題を解決するため、"匿名貸し出し" を提案する。この" 匿名貸し出し" は、貸し出しの際には匿名で貸し出し、返却期限を過ぎても未返却だった場合にはその匿名性を破るようなシステムである。この論文では、貸し出し機関が匿名で貸していた場合でもユーザー人当たりの権利の数を制限することができ、(つまり図書館であれば何冊貸しているかを制限できること) しかも、ユーザのプライバシは保たれるような新しいシステムを提案し、耐タンパーデバイスおよびオフライン電子マネー方式を利用してこれを実現した。

キーワード　電子マネー, プライバシ保護, Optimistic protocol, Timed-release encryption, Anonymity Revocation

A Note on an Anonymous Loan System

Rie SHIGETOMI†, Akira OTSUKA††, Takahide OGAWA†, and Hideki IMAI††

† Tsuda College　2-1-1 Tsuda-machi, Kodaira-shi, Tokyo 187-8577 JAPAN
†† Institute of Industrial Science, University of Tokyo　4-6-1 Komaba, Meguro-ku, Tokyo 153-8505 JAPAN
E-mail: †{m01shige,ogawa}@tsuda.ac.jp, ††otsuka@imailab.iis.u-tokyo.ac.jp, †††imai@iis.u-tokyo.ac.jp

Abstract　Recently, paper based transactions are being replaced by digitized transactions in rapid pace. These kind of digitized data is useful compared to paper based data in sense of the flexibility of the data. Loan Service, (for example, Library, Rental video, debt etc.,) is among the services that makes use of sophisticated digitized transactions. Loan services handle a lot of personal information, which enables the administrators of the information, the analysis of personal hobby and tastes, or even living levels. This leads to a large privacy problem. We examined ways to avoid this privacy problem. One solution is to use "An Anonymous Loan" that the user can be anonymous while borrowing and returning, but the anonymity is unveiled on the due date without return. In this paper, we will suggest a new version "An Anonymous Loan System" that the loan party is allowed to control the number of rights transact by one user, using tamper resistance device and electronic cash scheme for off-line payment.

Key words　Electronic cash, Privacy Protection, Optimistic Protocol, Timed-release Encryption, Anonymity Revocation

1. Introduction

Today, personal hobbies and tastes to entertainment are growing rapidly in their variety. To meet demands of the consumers, more and more companies are increasing the variety of services they are offering. Digitized transaction is one of the key elements that has enabled these growing. Loan service is one of the most sophisticated services offered digitally. Take the example of borrowing a book from a university library as a student: You used to have to give the librarian your student identification card and a sign a little card for each book you borrow. The little card had to be kept until you return the books. Today, when you borrow something from the library, all you have to do is hand your library card (or a student ID) and the book to the librarian. The librarian scans the library card, and the information of you borrowing those books is stored to the database of library computer. When personal loan information was recorded in paper-based systems, it was difficult for anyone to extract useful information from the system. In addition, the fact that most these kinds of information were throw away due some term because of the limit of libraries to store paper-based information, made it harder to re-build a personal loan history at the library.

However, these kinds of data are stored digitally in computer now, and information technology related developments have enabled us to save and control these digital data more easily. This means if a librarian wants to know who is borrowing a specific book, he (or she) can do so by only a few clicks on the computer.

In general, loan data in electronic form makes it easier for the administrator –compared to data in the paper form– to analyze the tastes of each person.

This leads to a privacy problem where there is a possibility for the administrator to take control of the activities of the people whom he(or she) has the loan data.

To avoid this problem, any loaning should be done anonymously.

On the other hand, anonymousness must be unveiled when, the date due has passed. Libraries, rental video shops, money loaning, and rental car are only few examples that utilize these kinds of loans.

Therefore, there is a requirement for a system that grant permission anonymously, but identifies the user when he (or she) delays.

2. Related Work

This work has some ideas in common with Anonymity Revocation and Timed-Release Encryption.

We have chosen Optimistic Approach as Anonymity Revocation. Optimistic approach was originally introduced by Askan et al. [1] [2]. This protocol is used in the nonreturning case, where due date has passed and the user's name must be unveiled.

On the other hand, electronic cash scheme has been a fertile area of research. Electronic cash scheme has consideration of privacy protection and anonymity revocation to keep off double spender [4] [5].

Time-Release Encryption Problem was first discussed by May [7]and was extended by Rivest et al. [8] and Crescenzo et al. [6].Revest suggested that the idea has three schemes: Time-Lock Puzzle, Online Time-Server and Off-line Time-Server.

The first one, Time-Lock Puzzle, uses computational complexity; that is the difficulty to solve the encryption. The idea is that the time to recover a secret is given by the minimum computational effort needed by any machine, serial or parallel, to perform some computation which enables one to recover the secret. However this idea depends on CPU-machine of receiver. As the CPU-machine of the receiver increases, the sender must choose a longer key, thus taking a longer time to encrypt the secret. The receiver must also consumpt an amount of CPU-machine to decrypt any message. Using an IC card as the receiver is unrealistic for this cost of calculation.

The second one, Online Time-server, uses a trusted third party (TTP) to release message at the appointed time. The idea is to use TTP to store the message M until its desired release time t. However in this model, TTP has to store M, which is inefficient. The same thing could be accomplished with less cost by the next model: TTP encrypts the document and hands the encrypted document to the receiver, but publishes the secret key only at time t. This could be accomplished with less cost, as all TTP has to keep is the secret key. In addition, TTP using secret sharing is also suggested, but to keep the story simple we will only use TTP. Crescenzo et al. [6] has suggested a more efficient model. However this model has a problem that it is hard to detect a malicious prover creates x and w by himself and uses when he (or she) verifies his (or her) borrowing rights.

The third one, off-line Time-Server, also uses a trusted third party(TTP) but the time-server is on a tamper-device, so it is difficult to control the exact time.

In Timed-Release Encryption, a message sent to the future must be unveiled. This means the message cannot be canceled. The second case of Rivest using only TTP enables this operation, but with a large encumbrance. We suggest a more efficient system using tamper resistance device and TTP who has small encumbrance, for services where the date due is passed returned and unreturned at almost the same

rate.

3. Preliminaries

We will define two schemes, Electronic Cash Scheme and Timed-Release Encryption.

3.1 Electronic Cash Scheme

Many off-line electronic cash scheme have ever been proposed in literature. In this paper, we will show Brand's [4] electronic cash scheme. Brand's electronic cash scheme is based on representation problem. Let G_q be a group of prime order q.

3.1.1 Representation Problem

[Definition 1] Let $k \geqq 2$. A generator-tuple of length k is a k-tuple$(g_1, \ldots, g_k)$ with $g_i \in G_q$ and $g_i \neq g_j$ if $i \neq j$. For any $h \in G_q$, representation of h with respect to generator-tuple $(g_1, \ldots, g_k)$ is a tuple $(a_1, \ldots, a_k)$, with $a_i \in Z_q$for all $1 \leqq i \leqq k$, such that $\prod_{i=1}^{k} g_i^{a_i} = h$.

[Definition 2] Repelesenation problem is to find a represenation of h with respect to $(g_1, \ldots g_k)$ from a group G_q , a generator-tuple $(g_1, \ldots, g_k)$ and $h \in G_q$.

3.1.2 Restrictive blind signature scheme

In this section, we will define Restrictive blind signature scheme based on representation problem.

Let u_1 be user ID, $m = g_1^{u_1} g_2$ be message, $A = m^s$ be message blinded.

[Definition 3] Let $m \in G_q$ such that the receiver at the start of a blind signature protocol knows a representation (a_1, a_2) of m with respect to a generator-tuple (g_1, g_2). Let(b_1, b_2) be the representation the receiver knows of the blinded number A of m after the protocol has finished. If there exists two functions I_1 and I_2 such that

$$I_1(a_1, a_2) = I_2(b_1, b_2)$$

regardless of m and the blinding transformations applied by the receiver, then the protocol is called a restrictive blind signature protocol. The functions I_1 and I_2 are called blinding-invariant functions of the protocol with respect to (g_1, g_2)

Therefore, after blind signature protocol, we have $I_1(a_1, a_2) = I_2(b_1, b_2) = u_1$.

3.1.3 Protocol

We will explain the electronic cash protocol which is three party protocols, Bank, Alice(consumer) and Charlie(merchant).

Let H and H_0 be a one-way hash function.

[setup]

Let G_q be a group of prime order q. Bank generates $g, g_1, g_2 \in_R G_q$ for public key and a number $x \in G_q$ for secret key. Alice generates at random a number $u_1 \in Z_q$, and computes $I = g_1^{u_1}$. If $g_1^{u_1} \neq 1$, then Alice transmits I to Bank, and keeps u_1 secret. Bank stores I. Bank computes $z = (Ig_2)^x$, and transmits it to Alice.

[The withdrawal protocol]

To withdraw a coin, Alice and Bank follow the protocol shown in Fig.1.

If Alice accepts in the payment protocol, then $A, B, (z', a', b', r')$ is a coin of which she knows a representation.

[Payment Protocol]

To spend a coin, Alice and Charlie follow the protocol below. See Fig.2.

After sometime, say at night, Charlie deposits the coin by sending $A, B, sign(A, B), (r_1, r_2), date/time$ to Bank.

3.1.4 Properties

Any electronic cash scheme satisfying following properties: (a) Unforgeability, (b) Revokablity and (c)Untraceability, are realized computationally.

[Unforgeability]

Any adversary cannot compute $(k+1)$ coins with the transcript of k protocols using an efficient algorithm.

[Revokability]

When Alice is trying to spend the same coin twice, Alice has to respond to two different challenges d, d' $(d \neq d')$ for one coin in the payment protocol with overwhelming probability of $(1 - 1/q)$. Since Bank now has at its disposal a pair (d, r_1, r_2) from the new transcript and a pair (d', r_1', r_2') from the deposited information, it can compute

$$g_1^{r_1 - r_1'/r_2 - r_2'} = g_1^{u_1}.$$

Then, Bank can gets u_1's ID from his public key $g_1^{u_1}$.

[Untraceability]

Given a coin from Alice, no adversary can find whether another coin is from Alice or not, using polynomial-time algorithm.

Such electronic cash schemes have already been proposed [4], [5], our protocol is available to extend another electronic cash scheme.

3.2 Timed-Release Encryption

The setting is as follows. There are three participants: the sender, the receiver and the server [6] [7] [8]. First, the sender transmits to the receiver an encrypted messages and a release-time. Then, the receiver is allowed to in a conversation with a server. The server and the receiver engage in a conditional oblivious transfer such that if the release-time is not less than the current time defined by the server, the receiver gets the message. Otherwise, the receiver gets nothing. Furthermore, the server does not lean any information about the release-time or the identity of the sender. In particular, the server does not lean whether the release-time is less than, equal to, or greater than the current time.

The sender can get $pkey$ from the server for encryption his

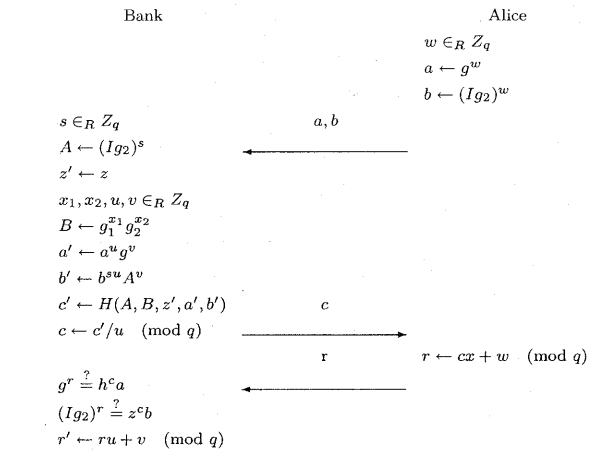

Fig. 1 Withdrawal Protocol

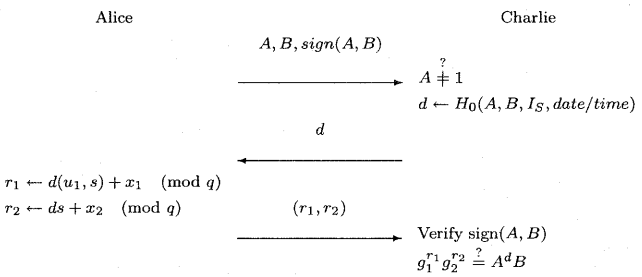

Fig. 2 Payment Protocol

or her messages. At the release-time, the receiver can get *skey* from the server if he or she wants to. Hence, the server gives the pair, *pkey* and *skey*, which first is the public key and the other is the secret key that opens at release-time.

Let R be relation, then $(pkey, skey) \in R$

3. 2. 1 The Model

Timed-Release Encryption consists of the following algorithms:

- TS_p: A encrypted algorithm that takes the time server's public key *pkey* and the sender's message m as the input. The output is the encrypted message.
- TS_s: A decrypted algorithm that takes the time server's secret key *skey* as the input which opens at release time from the time server. the outputs is the decrypted message, that means the sender's message m.

The following informally stated security requirements must hold:

- TS security: It is not feasible to find out the sender's message m before release time without the time server.

4. Definition

4. 1 Services with time limit

An User P who has the right is offered a main service from an organization V, under the condition that the user does the returning service on a specific time S_T in future.

For example, *LoanService* is a kind of service where the User has the right to transact an item from the Loan Party under the time limit S_T. S_T is, in other words, the due date.

4. 1. 1 Description of States

This service takes several states as in figure[3].

Let S be a set of states. Each $s \in S$ has five states:

(0) Non-Member s_0

P is not a member of V.

(1) Member s_1

P has applied for the membership of V for the service with time limit S_T from V.

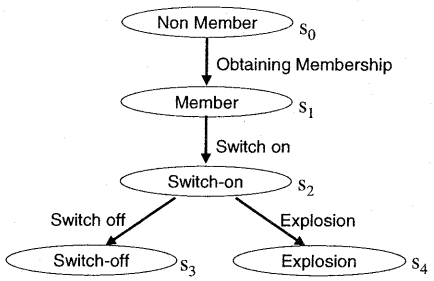

Fig. 3 Flowchart

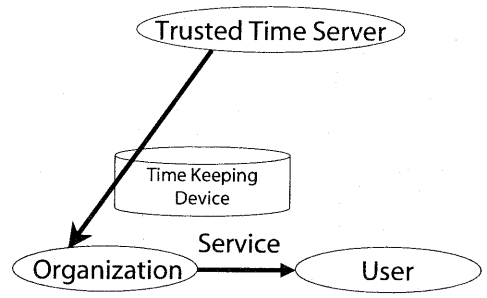

Fig. 4 Construction

(2) Switch on s_2

V provided P the main service.

(3) Switch off s_3

P did the returning service to V before S_T.

(4) Explosion s_4

User P did not do the returning service to V before S_T.

We divide S into five elements, s_0, s_1, s_2, s_3 and s_4 such that s_0 = a state of (0), s_1 = a state of (1), s_2 = a state of (2), s_3 = a state of (3) and s_4 = a state of (4).

4.2 The Model

(1) *UserID*

UserID is an ID to identify each user. UserID may be published by multiple organizations, (i.e. users may choose where they have their ID published) thus UserID is unique for each pair of user and organization.

Let U be User Group then UserID is $u_1, u_2, \ldots u_n \in U$.

(2) *TransactionID*

TransactionID is an ID unique for each transaction. The Organization publishes the Transaction ID. The transaction ID also act as the identification for the service itself. Transaction ID includes the date and the service information.

Let T be a set of transactionID then TransactionID is $t_1, t_2, \ldots \in T$, and f_S be a function such that $f_S(T) = S$.

We divide T into four subsets T_1, T_2, T_3 and T_4 such that

$T_1 = \{t |\ t \in T, f_S(t) = s_1\}$,

$T_2 = \{t |\ t \in T, f_S(t) = s_2\}$,

$T_3 = \{t |\ t \in T, f_S(t) = s_3\}$ and

$T_4 = \{t |\ t \in T, f_S(t) = s_4\}$.

4.3 Honest-user Anonymity

(1) Untraceability : Given $t \in T_2 \cup T_3$, and $U = \{u_1, u_2, \ldots u_n\}$, it is computationally infeasible for an probabilistic polynomial-time adversary to find $u_k \in U$ ($k \in \{1, \ldots, n\}$)who engaged in t.

(2) Unlinkability: Given $t_i, t_j \in T_2 \cup T_3$ ($i, j \in \{1, \ldots, n\}$), and $U = \{u_1, u_2, \ldots u_n\}$, it is computationally infeasible for an probabilistic polynomial-time adversary to decide whether there exists same user u_k who engaged in t_i, t_j.

Under the condition of Untraceability and Unlinkability, it is said that the *honest-user Anonymity* is available.

4.4 Over-due date Traceability

Traceability : Given $t \in T_4$, and $U = \{u_1, u_2, \ldots u_n\}$, it is unveiled user u_k who engaged in t.

Under the condition of Traceability, it is said that the *Over-duedate Traceability* is available.

4.5 Upper limit

For a given user $u \in U$, no more than k transactions related to u can be in state s_2 simultaneously.

4.6 Problem Setting

We will define the Problem Setting in this paper.

(1) Honest-user Anonymity

(2) Over-duedate Traceability

(3) Upperlimit

When these three requirements are satisfied, this scheme called "An Anonymous Loan System".

5. Construction

In this section, we will suggest a new scheme for "An Anonymous Loan System".

5.1 Definition of Players

Our scheme is three-party protocol, User P, Organization V and Trusted Time Server TS. We give a definition for each of the four players and relations using figure [4].

5.1.1 User P

User P is the person receiving the service with time limit.

P should receive the permission to be provided the service with time limit from the Organization.

5.1.2 Organization V

The Organization V provides the service with time limit to the permitted members V holds a Time Keeping Device published by the Trusted Time Server. V corresponds to the Bank on electronic cash scheme who generates coins as borrowing rights for each user

5.1.3 Time Server TS

The Time Server TS has the next two facilities in this protocol.

- Time Server
- Time Keeping Device

(1) Time Server TS

On time S_T, Time Server TS announces a secret key $skey$. The public key $pkey$(corresponding to $skey$) is announced before S_T. $skey$ and $pkey$ is on Timed-Release Encryption.

(2) Time Keeping Device IC

The Time Keeping Device IC is a tamper resistance device like an IC card.

When the information and $pkey$ are inputted, IC saves those informations. IC is allowed to do the following three operations:

(a) When information and $pkey$ are inputted, IC is allowed to save the information encrypted using TS_p.

(b) It is allowed to delete the information saved at (a).

(c) It is allowed to output the information stored in (a) when a valid skey is inputted.

5.1.4 Assumption 1

We assume that the Time Server TS, including the Time Keeping Device IC is honest.

5.2 Proposed protocol

In this section, we will define the protocol described in section 4.1.

5.2.1 Obtaining Membership

P must first apply for the service with time limit to have service with time limit from V. For example, when you want to borrow some books from the library, you might have to register the library your address and telephone number to get a library card.

We will discuss based on figure [5].

First, P asks for a membership to V, V then decides whether to grant membership to P.

If V decides to reject, P cannot obtain membership. On the other hand, when V decides to grant membership to P, V issues α, which is the condition of service. α includes informations such as the number of books P is allowed to get service. Then, P and V does *setup* on electronic cash scheme, so V has got $g, g_1, g_2 \in G_q$ for public key and $x \in G_q$ for secret key. P generates at random a number $ID_p \in G_q$ and computes $I = g_1^{ID_p}$. If $g_1^{ID_p} \neq 1$, then P transmits I to V and keeps ID_p secret. V stores I. V computes $z = (Ig_2)^x$, and transmits it to P

When all of this succeeds, P obtains membership of V.

Next, we will discuss about P getting *ServiceRight* for the service with time limit. *ServiceRight* is the same as coins for electronic cash scheme. P can get k coins from V. In this paper, we will discuss about the case of $k = 1$. If $k > 1$, then V generates coins k times.

This is P and V doing *withdrawalprotocol* on electronic cash scheme.

As the result, P has got $< A, B >$ and $< z', a', b', r' >$ as Service Right.

5.2.2 Switch-on

Next, we will discuss about P using the main service provided by V with figure[6].

The procedures are as follows:

(1) P sends the *ServiceRight* to V and IC.

(2) IC saves *ServiceRight*.

(3) V provides service to P.

First, P sends *ServiceRight* (that is, a coin in electronic cash scheme) to V and IC. Note that a malicious P might send a diffrent *ServiceRight* to V and IC. To avoid this, P first sends *ServiceRight* to IC, then IC sends the given *ServiceRight* to V. This is possible because of Assumtion 1.

V generates a Transaction ID y_m which must be a secret number only known to V. Then, V calculates $h(y_m)$ using a one-way hush function h and sends $h(y_m)$ and pkey of TS to IC.

V computes c for challenge, and sends c to P. Also, IC computes and sends c' to P. Then P computes the responses r_1, r_2 for V and r_1', r_2' for IC, and sends them.

Also, IC saves the following:$pkey$, $TS_p((r_1', r_2'), pkey), h(y_m)$.

5.2.3 Switch-off

In this section, we will discuss about Switch-off with figure[7]. This is the case where P does the returning service before S_T, for a main service provided by V.

V must delete the data related to y_m in IC. This is done by sending y_m and $h(y_m)$ to IC. Then, IC searches the saved data related to $h(y_m)$ and deletes the following: $pkey$, $TS_p((r_1', r_2'), pkey), h(y_m)$.

Hence, the information about the user who received the service is deleted.

IC creates a message γ, representing the deletition has finished successively, and sign γ. Then IC sends γ and the signed γ, $Sign_{IC}(\gamma)$ to P and V.

5.2.4 Explosion

In the case where P does not proceed the returning service

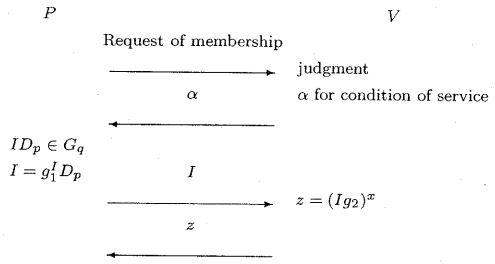

Fig. 5 Obtaining Membership

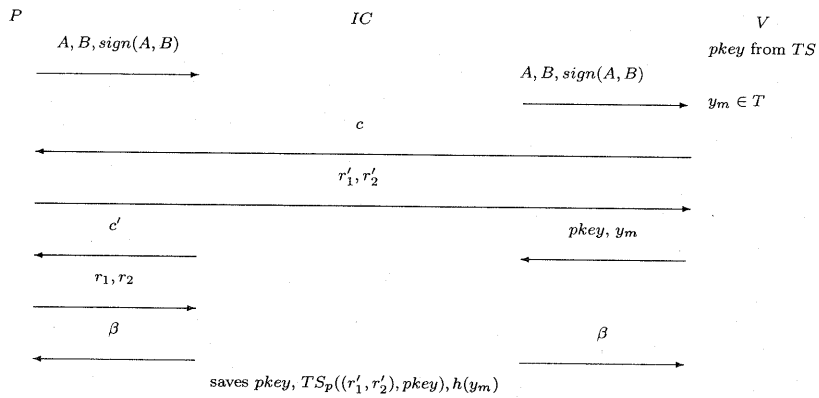

Fig. 6 Switch-on

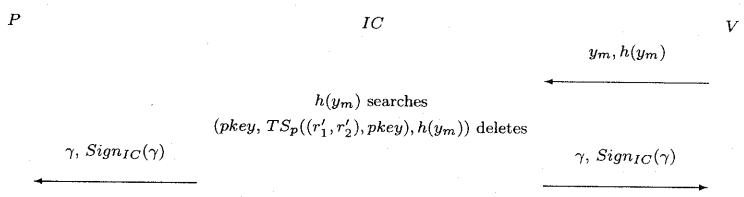

Fig. 7 Switch-off

in due date, V unveils the User ID as in figure[8].

V sends to IC, $pkey$ and $skey$ obtained from TS.

If IC decides the inputs, that is $pkey$ and $skey$, is valid, IC searches and outputs following: $pkey$, $TS_p((r'_1, r'_2), pkey), h(y_m)$.

V can find r'_1, r'_2 from TS_s because $TS_s(TS_p((r'_1, r'_2), pkey), skey) = (r'_1, r'_2)$.

Then, V can compute

$$g_1^{r_1 - r'_1 / r_2 - r'_2} \equiv g_1^{ID_p}$$

Hence, V knows ID_p.

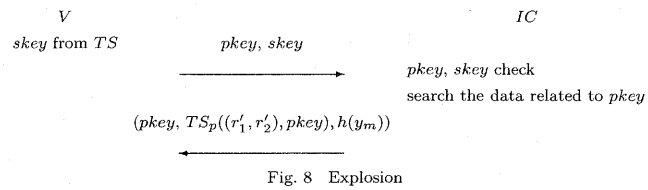

Fig. 8 Explosion

6. Theorem and Proof

6.1 Theorem 1

We defined the following problem setting in section 4.5:

- Honest-user Anonymity
- Over-due date Traceability.
- Upper limit

Construction in section 5 confirms these problem setting.

6.2 Proof

6.2.1 Honest-user Anonymity

In this case, we have to discuss about the situations in the proposed protocol: $t \in T_1$ and $t \in T_2$.

(1) $t \in T_2$

(a) Untraceability:

The data of the relation of t and u_k is encrypted by the Timed-Release Encryption and stored in IC. Therefore, Malicious V cannot trace the identity of the honest user because of TS security of the Timed-Release Encryption and tamper resistance device. In addition, V cannot trace the User ID because of Untraceability of the Electronic cash scheme.

(b) Unlinkability:

This case concludes Untraceability of Electronic cash scheme because P uses different coin for each transactions.

(2) $t \in T_3$

(a) Untraceability:

From Assumption 1, all the transcripts in IC correctly returned by an honest user, the data which t engaged u_k, are deleted. Therefore, malicious V cannot trace the identity of the honest user once items are returned.

(b) Unlinkability:

This case concludes Untraceability of Electronic cash scheme because P uses different coin for each transactions.

6.2.2 Over-due date Traceability

In this case, we have to discuss about the situation, $t \in T_4$. V can get the User ID from a Time Keeping Device. This case concludes Revokability of Electronic Cash Scheme.

In the situation where malicious user sends fake coin to IC, he (or she) is not able to create a fake coin. This case concludes Unforgeability of Electronic Cash Scheme.

6.2.3 Upper limit

P is not able to create a fake coin because of Unforgeability of Electronic Cash Scheme.

7. Conclusion

We have made an analysis on the digitized form of loan systems, and stated a privacy problem that would not have happened with paper based systems. We have suggested a system named "An Anonymous Loan System" that could solve this privacy problem using electronic cash scheme. By using electronic cash scheme, we can also solve the "Upper-limit" problem to control the number of service rights. Our protocol heavily depends on tamper-proof, and methods to reduce the dependency to tamper-proof should be necessary.

However, our protocol does not mention about redestribution of the coins, so this remains as a future work.

References

[1] N. Asokan, V. Shoup, and M. Waidner: Optimistic fair exchange of digital signatures. IEEE Journal on Selected Areas in Communications, vol. 18, pages 591-610, Apr. 2000.

[2] N. Asokan, M. Schunter, and M. Waidner: Optimistic protofol for fair exchange. Proc. of ACM-CCS '97, pages 8-17, 1997.

[3] G. Ateniese, J. Camenisch, M. Joye, and G. Tsudik: A Practical and Provably Secure Coalition-Resistant Group Signature Scheme, CRYPTO 2000, LNCS 1880. pp. 255-270, 2000.

[4] Brands S.: Untraceable off-line cash in wallets with observers, Proc. of CRYPTO '93, pp 302–318 (1994).

[5] Chaum D., Fiat A.,Naor M.: Untraceable Electronic Cash, Proc. of CRYPTO '88, pp. 319-327 (1988).

[6] G. D. Crescenzo, R. Ostrovsky and S. Rajagopalan: Conditional Oblivious Transfer and Timed-Release Encryption. Advances in Cryptology-EUROCRYPT '99, pages 74-89, Springer, LNCS vol. 1592. 1999.

[7] T. C. May: Timed-release cryoto, Febuary 1993. http://www.hks.net/cpunks/cpunks-0/1460.html.

[8] R. L. Rivest, A. Shamir, and D. A. Wagner: Time-Lock Puzzles and Timed-Release Crypto, http://theory.lcs.mit.edu/ rivest