# An Information System Platform for Anonymous Product Recycling

Shinsuke Tamura
University of Fukui/Graduate School of Engineering, Fukui, Japan
tamura@fuis.fuis.fukui-u.ac.jp

Kazuya Kouro, Masatoshi Sasatani, Kazi MD. Rokibul Alam and Hazim Anas Haddad
University of Fukui/Graduate School of Engineering, Fukui, Japan
Email: {h7kouro, h9rokibu}@radio.fuis.fukui-u.ac.jp

***Abstract*—Recycling is one of the most important solutions for reducing material and energy consumption, and that of end products is considered as the most effective one. To promote recycling of end products, this paper discusses the requirements for information system platforms, in which people can anonymously exchange their owning end products through auctions, while assuming that the major barrier for people to be involved in these auctions is that people who sell, buy or use products can be traced by entities that manage auctions. It is also shown that these requirements can be satisfied by the state of the art technologies.**

***Index Terms*—anonymous, authentication, auction, account calculation, network**

## I. Introduction

Recycling is one of the most important issues for reducing material and energy consumption, and that of (end) products is considered as the most effective one, because it enables recycling of products without or with less processing of them. This paper proposes an information system platform for product recycling systems that encourage people to exchange their owning products to be recycled.

Information systems play one of the most important roles in product recycling to encourage people to exchange their owning products. They enable people who want to buy products to get precise information about conditions of the same kind of products that are owned by people who want to replace them by new ones, and enables manufacturers to provide users of their products with information about their new types of products with higher performances, etc. Manufacturers also can inform their customers of expected lifetimes of their products while remotely monitoring their running conditions.

However, while these kinds of information make product-recycling processes efficient and convenient, privacies of individual persons become difficult to protect. People who want to sell or buy products must disclose information about when they bought them, how they used them, what they bought and sold until now, etc. These privacy disclosures make people reluctant to use these information systems, and consequently, product-recycling systems lose opportunities to invite people who want to sell or buy their owning products. To develop efficient and convenient product-recycling systems while securely protecting privacies of individuals, in this paper, an information system platform that enables people to put and get information about their owning products and exchange products through auctions without disclosing their identities is proposed.

Several electric auction systems, in which advanced security technologies are applied to protect privacies of auction participants, are proposed already. However in these systems absolutely trustworthy neutral entities are assumed to maintain secret information of individuals securely; therefore they are not suitable for practical applications. Because no entity is absolutely trustworthy in the real world, maintaining such neutral entities is expensive, especially for large-scale applications. The information system platform discussed in this paper enables to exclude absolutely trustworthy entities completely, and therefore it becomes possible to protect privacies of people while maintaining practicality and scalability of product recycling systems.

## II. Privacy protection in auctions for recycling products

The total system considered consists of 3 parts, i.e. physical product management, product database and product exchange management parts as shown in Fig. 1 [18], and the information system platform relates to the product database and the product exchange management parts. The physical product management part is responsible for handling of physical products to be recycled. Different from products placed at factories, that are placed at recycle centers are not well arranged; various and many kinds of products with different features may be placed in their arrival order without

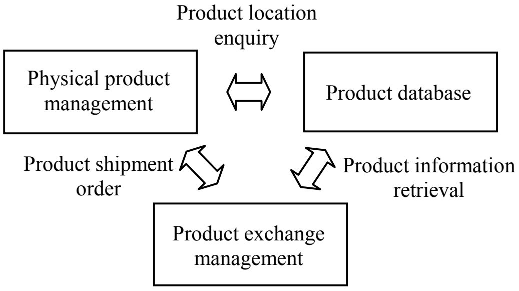

Product location enquiry
Physical product management
Product database
Product shipment order
Product information retrieval
Product exchange management

Figure 1. Configuration of a system for product recycling

considering troubles that may occur when they are searched and picked for their recycling. Therefore, new technologies that enable their quick and automatic handlings are necessary.

The product database part is responsible for information retrieval about products to be exchanged and about their state histories, e.g. running and maintenance histories. Because various and many kinds of products are put into recycle centers, it is impossible to define data structure in advance, and databases with dynamic data structure become necessary [20]. Regarding to the protection of privacies, the product database must be able to maintain these various data without knowing owners of products. The product exchange management part is responsible for the achievement of the anonymous product exchanges among people who intend to sell them (sellers) and buy them (buyers).

Issues about privacy protection strongly relate not only to the product exchange management part but also to the product database part, however in the followings, discussions are focused on mechanisms to protect privacies of individual sellers and buyers in the product exchange management part, as the extension of discussions in [18], while assuming that the one of the major reasons that makes people reluctant to exchange products for their recycling is the fact that sellers and buyers can be traced by others, e.g. by entities that are managing the recycle center. Mechanisms necessary for the product database part are discussed in another paper as anonymous memories [21].

The proposed product exchange management system consists of registration, authentication, auction and bill making and payment parts as shown in Fig. 2.

Firstly, the registration part takes care of maintaining authorized members of the system, i.e. new members are added and leaving members are removed from the member list of the system by this part. Authentication part authenticates members authorized by the registration part and that are intending to participate in auctions to sell or buy products. Only authenticated members can bring products to the recycle center, and products brought to the center are sold through auctions among buyers that are authenticated by this part. The auction part is responsible for running of auctions. Then lastly, the bill making and payment part charges buyers for their winning products in auctions and pays costs for these products to sellers on behalf of buyers.

Requirements about privacy protections in the product exchange management system can be summarized as follows.

1) Only authorized members can bring products to be sold in auctions to recycle centers, but identities of sellers must be anonymous,
2) Only authorized members can participate in auctions as buyers,
3) Auctions for exchanging recycled products must be conducted anonymously and fairly,
4) Sellers can receive their exact sales amounts that correspond to the winning prices of auctions from recycle centers without disclosing their identities,
5) Recycle centers can receive winning prices from auction winners without knowing their identities, and
6) No absolutely trustworthy entity is assumed.

The *1*st and *2*nd requirements concern with the safe operations of recycle centers, i.e. in order to protect centers from various threats caused by anonymous persons, only authorized persons are allowed to bring and buy products to or from recycle centers. However, sellers or buyers of individual products must be concealed from others in order to maintain their privacy. Regarding to auctions, i.e. about the *3*rd requirement, in order to maintain privacies of buyers, not only identities of buyers who make bids in an auction but also sequences of bids made by same buyers in auctions must be concealed from others, because these sequences may become strong supports to identify buyers. Nevertheless, to make auctions fair enough, it must be ensured that anonymous auction winners are forced to buy their winning products and recycle centers cannot sell their products to persons except auction winners.

About the *4*th and *5*th requirements, although identities of sellers and winners of individual auctions are unknown, recycle centers must be able to calculate total amount to pay to individual sellers or to charge to individual buyers correctly at the end of its every business period. Finally, to convince sellers and buyers that their identities are not disclosed, absolutely trustworthy entities cannot be assumed; because no entity exists that is absolutely trustworthy in the real world.

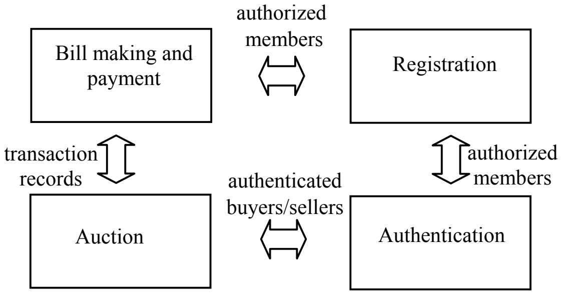


Figure 2. Configuration of the product exchange management system

To satisfy the above requirements, the authentication, bill making and payment, and auction parts must have mechanisms for anonymous authentication, anonymous account calculation, and anonymous auction, respectively. Moreover, to encourage more people to be involved in product recycling, the environment that enables sellers and buyers to communicate remotely with the recycle center through wide area networks like Internet is inevitable. Through this kind of environment sellers and buyers can sell and buy products to be recycled without visiting recycle centers. However when persons access the recycle center through the current Internet, identities of them can be disclosed easily by tracing message paths back to them. Therefore mechanisms for anonymous communications are also necessary.

## III. Related works

Regarding to the anonymous authentication, an authentication authority can authenticate entities without knowing their identities easily, when the authority assigns the same password to all entities. A major drawback of this scheme is that all entities should change their password every time when even a one entity leaves the system. Anonymous tokens remove this difficulty [2]. In the anonymous token mechanism, an entity prepares its token; and the authentication authority signs on the token blindly after checking the eligibility of the entity, i.e. the authentication authority signs on the token without knowing the content of the token. Therefore the entity can show its eligibility by the correctness of the signature on its token without disclosing its identity. Different from the scheme with the same passwords, the authentication authority can invalidate tokens when their owners leave the system by only recording invalid tokens that are returned to the authority. When refreshable tokens are used [12], in which an entity can change its token every time when it is authenticated, token identifications of frequently visiting entities also can be concealed. However anonymous tokens have a difficulty in treating token oblivions, i.e. when an entity requests reissuing of its token while claiming that it had lost its token, all other entities must change their tokens to new ones. Because the lost token is anonymous, the authority cannot identify the exact token that should be invalidated; therefore it should invalidate all tokens. To solve this problem an anonymous authentication mechanism proposed in [14] is adopted in this paper.

A primary objective of anonymous credit card systems is just to establish the anonymous account calculation. However not many mechanisms are proposed until now, also almost all mechanisms assume the existence of absolutely trustworthy neutral entities. In [5], anonymity of transactions made by card holders are maintained by assuming trustworthy multiple banks and anonymous accounts of cardholders, i.e. anonymities can be achieved when the multiple banks do not conspire. To remove absolutely trustworthy entities and anonymous accounts that are expensive to maintain in the real world and that are allowed not in all countries, respectively, this paper adopts the mechanism proposed in [16, 17].

Although various anonymous auction mechanisms had been proposed already [4, 6, 9, 10], they do not satisfy the requirements for anonymous auctions completely. Group signature adopted in [9] cannot provide participants of auctions with their complete anonymity, i.e. group managers (trustworthy neutral entities) can identify participants that make bids in auctions. Although a mechanism used in [10] does not assume any absolutely trustworthy entity, it can neither force buyers in auctions to buy their winning products nor to force sellers to sell their products to auction winners. Therefore, this paper discusses the feasibility of the mechanism that satisfy the above requirements based on the one proposed in [15].

About the anonymous communication, among of various methods [7, 8], only DCnet [3] and Mix-net [1, 11, 13] achieve the complete anonymity. In DCnet, a message-sending node sends its message while adding its secret random number to the message, and at the same time other ($n$-$1$) nodes send their secret random numbers. Then the message receiver can reconstruct the message without knowing the sender by adding messages sent from the sender and the other ($n$-1) nodes, when random numbers are assigned to these $n$ nodes so that the sum of them becomes $0$. However, DCnet is not practical apparently; multiple nodes must agree with the random numbers every time when a message is sent. Although Mix-net, in which multiple servers transferring messages while shuffling and encrypting incoming messages, is more practical than DCnet, it is still not efficient enough, because individual servers must encrypt all messages they transfer based on asymmetric key encryption algorithms. To achieve efficient anonymous communications, a symmetric key encryption algorithm based Mix-net [19] is enhanced in this paper.

## IV. Potential solutions for the product exchange management system

This section proposes mechanisms that satisfy the previously discussed requirements for the individual parts in the product exchange management system. As described below, it can be considered that state of the art technologies can provide solutions to satisfy the requirements.

### A. Registration Part

Because members added or removed to or from the system are not anonymous, no specific function is required for this part. Namely, registrations can be done by providing new members with their identification codes (IDs) and passwords, and adding them to the member list. Also removals of members can be done by simply deleting their IDs from the list.

### B. Authentication Part

This part must provide the mechanism to authenticate authorized members who intend to participate in auctions without knowing their identities. This mechanism can be implemented easily as shown in Fig.3 [14]. In the following, the authentication authority is the recycle center and entities to be authenticated are the members of the center.

Firstly member $M$ sends an ID list $D$ that includes $M$'s own ID to the recycle center $C$, and $C$ makes $P$, a list of passwords that correspond to IDs in $D$, while encrypting individual passwords by using random key $r$, which is not known to $M$. Then $C$ asks $M$ to show the key $r$ that was used for encrypting passwords in the list. Here, it is assumed that IDs and passwords of the same member appear in the same position in $D$ and $P$, and $M$ knows the password encryption algorithm. Then, because $M$ can calculate the correct key only when it knows at least one of passwords in the list, and it knows only its own password, $C$ can determine that $M$ is eligible when it returns the correct key $r$. However, $C$ cannot identify $M$, because all members that know their passwords in the list can calculate the key correctly.

In the figure, an encryption key is a secret random bit pattern $r$, and $C$ encrypts password $p$ into $q = p$ XOR $r$, and sends $q$ with E($r$). Then, $M$ can reconstruct $r$ by calculating $r' = q$ XOR $p = p$ XOR $r$ XOR $p$. Here, E() is an one way function, i.e. it is easy to calculate E($r$) from $r$ but it is difficult to find $r$ from E($r$). It must be noticed that member $M$ returns $r'$ only when E($r$) = E($r'$) is confirmed, in order to force $C$ to use the same key for encrypting all passwords in the list. When they do not coincide, $M$ decides that $C$ assigns different keys to different passwords to identify the member that is requesting the authentication.

To make the mechanism practical enough the following problems must be solved, i.e.

- Password protection: The password list sent from the center includes passwords of other members and member $M$ can know them. Therefore password $p$ must be transformed into $\underline{p}$ = F($p$) before being encrypted by key $r$ and put in the password list. Here, F() is an one way function, therefore $M$ can calculate F($p$) from $p$, but cannot calculate $p$ from F($p$). However, because $M$ can try to calculate password $w$ of other person from F($w$) without being connected to $C$, function F must be difficult enough to find $w$ from F($w$).
- Frequently visiting member protection: IDs of members that request authentication many times appear frequently in ID lists sent from members, and then $C$ can identify frequently visiting members from logs of ID lists. Therefore mechanisms that reduce appearances of their IDs are necessary.

Fortunately the state of the art technologies are mature enough to address these issues. Sophisticated encryption mechanisms can solve the former problem, and aliases combined with implicit transaction links [16, 17] explained later can solve the latter problem, for example.

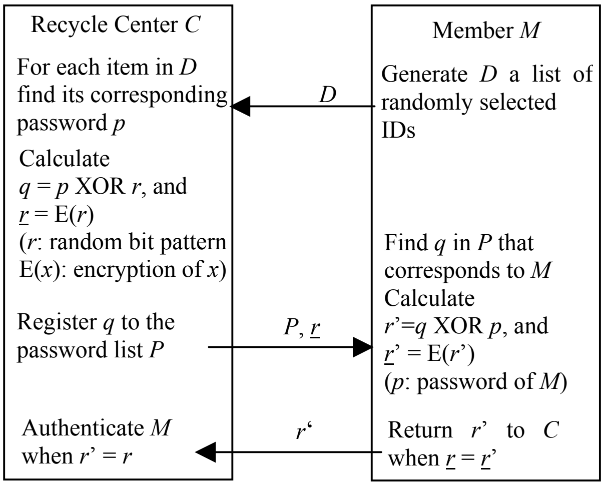

Recycle Center $C$
For each item in $D$ find its corresponding password $p$
Calculate $q = p$ XOR $r$, and $\underline{r}$ = E($r$) ($r$: random bit pattern E($x$): encryption of $x$)
Register $q$ to the password list $P$
Authenticate $M$ when $r' = r$
Member $M$
Generate $D$ a list of randomly selected IDs
Find $q$ in $P$ that corresponds to $M$ Calculate $r'$=$q$ XOR $p$, and $\underline{r}'$ = E($r'$) ($p$: password of $M$)
Return $r'$ to $C$ when $\underline{r} = \underline{r}'$
$D$
$P$, $\underline{r}$
$r$‘

Figure 3. Anonymous authentication

### C. Bill Making and Payment Part

Requirements for this part are, a) to make individual transactions of sellers and buyers secret from others including the recycle center, and to disable the recycle center to trace transaction sequences made by same sellers or buyers, b) to enable the recycle center to calculate total sales amounts of individual sellers and total expenditures of individual buyers correctly, and c) to enable the recycle center to detect dishonest sellers and buyers who intend to delete, modify or forge their transaction records.

These requirements can be satisfied by additive encryption, implicit transaction links, and blind signature mechanisms used in anonymous credit card systems. Namely the anonymous credit card system has the following properties. In the following, transaction IDs, a credit card company or a server, and cardholders or clients in [16, 17] are replaced by tokens, a recycle center and members (sellers or buyers in auctions), respectively.

1) The center can neither identify members that execute individual transactions, nor link transactions executed by same members,
2) The center can calculate the total expenditures or sales amounts of individual members at the end of its every business period (e.g. end of every month),
3) The center can identify members that execute dishonest transactions and charge them for correct amounts without information about other members,
4) Members can detect the center's dishonest operations, and
5) No absolutely trustworthy entity is assumed.

The mechanism consists of 3 phases, i.e. transaction, account calculation and state recovery phases, as shown in Fig.4.

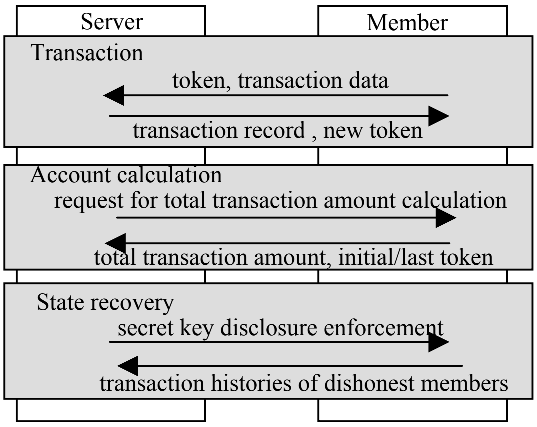

Server
Member
Transaction
token, transaction data
transaction record , new token
Account calculation
request for total transaction amount calculation
total transaction amount, initial/last token
State recovery
secret key disclosure enforcement
transaction histories of dishonest members

Figure 4. Three phases in the anonymous credit card system

*Transaction phase*: The *1*st requirement is satisfied as follows. Firstly, the authentication part authenticates member *M* without identifying it as described in the previous sub-section. Although the recycle center *C* confirms that *M* had completed its previous transaction successfully at the beginning of the transaction phase in order to detect dishonest operations of *M*, this is achieved anonymously by checking tokens, which are unique in the system and attached to individual transaction requests. Namely, for every transaction request, *C* confirms that the token attached to the request has its signature and it is not used repeatedly, and signs on a new token of *M* for its next transaction in exchange for the current one. However, *C* signs on a new token without knowing its content based on the blind signature mechanism [2]; therefore *C* cannot trace consecutive transactions executed by *M*.

In more detail, a member *M* chooses a number $N_i$ that is unique in the system at its (*i-1*)-th transaction, and encrypts $N_i$ into $E_{MS}(N_i)$ by its secret encryption key $K_{MS}$, and the recycle center *C* signs on $E_{MS}(N_i)$ by its secret key $K_{CS}$ to generate $S_{CS}(E_{MS}(N_i))$. Here, the result $S_{CS}(E_{MS}(N_i))$ is the blind signature of *C* on $N_i$, i.e. *C* signs on $E_{MS}(N_i)$ without knowing $N_i$. Finally, *M* decrypts $S_{CS}(E_{MS}(N_i))$ into $S_{CS}(N_i)$ by its secret decryption key $K_{MS}^{-1}$, then $S_{CS}(N_i)$ is the signature of *C* on $N_i$, i.e. anyone can generate $N_i$ from $S_{CS}(N_i)$ by using public key $K_{CS}^{-1}$.

Member *M* can generate number $N_i$ that is unique in the system by picking it from the token-list prepared by *C*. Of course *M* should enter the system anonymously through the anonymous authentication to pick numbers, and to force members to pick numbers only from the token-list, numbers in the list must have signatures of *C*.

*Account calculation phase*: In the transaction phase, member *M* records its individual transactions by itself in order to maintain its anonymity. Although *C* can save individual transaction records, it cannot calculate the total amount of transactions executed by *M*, because *C* does not know identities of members that execute transactions, and also the correspondences between current and new tokens are concealed from *C*. Therefore at the account calculation phase, *M* itself is responsible for calculation of the total amount of its transactions. Transaction records in their additively veiled forms disable *M* to maliciously modify its transaction records while enabling *M* to calculate its total transaction amounts. Here, V(*A*) and V(*B*), additively veiled forms of transaction records *A* and *B*, satisfy the relation V(*A*) + V(*B*) = V(*A*+*B*). Therefore, when individual transaction records are encrypted into additively veiled forms by *C*'s secret key, *C* can reconstruct total transaction amount of *M* without knowing its individual transactions, by decrypting the sum of additively veiled forms calculated by *M*, i.e. the *2*nd requirement is satisfied.

In the anonymous credit card system, an additively veiled form transaction record of *M* is constituted as a set of mutually independent linear combinations of *E* (a transaction amount), $\{L_0, L_1, ---, L_v\}$ (an implicit transaction link explained later), *N* (the number of transactions executed by *M*; to disable *C* to identify frequent visiting members, *N* is initialized when it reaches the predetermined value), and *R* (a random number secret from *M* that obscures the additive property of the encryption mechanism). Namely, a transaction record is calculated by (1). Then, *C* that knows coefficients $\{a_{ij}\}$ can calculate *E* from $\{S_1, S_2, ---, S_w\}$ by solving (1), at the same time, members cannot modify additively veiled forms consistently because $\{a_{ij}\}$ are secret from them.

$$\begin{aligned}
S_1 &= a_{11}E + a_{12}L_0 + --- + a_{1(2+v)}L_v + a_{1(3+v)}N \\
&\qquad + a_{1(4+v)}R \\
S_2 &= a_{21}E + a_{22}L_0 + --- + a_{2(2+v)}L_v + a_{2(3+v)}N \\
&\qquad + a_{2(4+v)}R \\
&\qquad --- \qquad\qquad\qquad\qquad (1) \\
S_w &= a_{w1}E + a_{w2}L_0 + --- + a_{w(2+v)}L_v + a_{w(3+v)}N \\
&\qquad + a_{w(4+v)}R \\
&,\text{where } w = 4 + v
\end{aligned}$$

It is well known that encryption mechanisms based on linear combinations are weak against plain text attacks, i.e. when enough number of encrypted data of known information are available, coefficients $\{a_{ij}\}$ of linear combinations can be calculated by solving inverse equations. This drawback of linear combination based encryption mechanisms can be removed by inserting dummy equations. When a set of data $\{S_1, ---, S_w\}$ is divided into *n* elements $\{S'_1, S'_2, ---, S'_n\}$, and *m* dummy elements are added while shuffling the merged elements, the probability that the correct *n* meaningful elements are extracted out of (*n* + *m*) elements is $1/_{n+m}C_n$. Therefore, by adjusting the numbers *n* and *m*, linear combination based encryptions can achieve any desired strength.

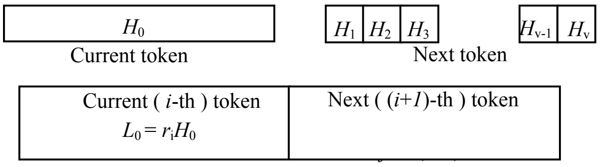
*State recovery phase*: State recovery phase starts when inconsistency is detected in some transaction, so that the recycle center *C* can calculate correct transaction amounts of dishonest members. Here, C can detect dishonest members, i.e. the *3*rd requirement is satisfied, by implicit transaction links as follow.

Let assume that $\{H_0, H_1, ---, H_v\}$ is the current and the next (future) tokens of the *i*-th transaction of member *M*, i.e. $H_0$ is the current token, and the next token is divided into *v* parts $\{H_1, ---, H_v\}$. Then the implicit transaction link $\{L_0, L_1, ---, L_v\}$ for the *i*-th transaction of *M* is calculated based on a pair of the current and the next tokens as shown in Fig. 5. Namely, $L_0 = r_i H_0$ and $\{L_j = r_{i+1}(b_{j1}H_1 + --- + b_{jv}H_v)\ ;\ j = 1, ---, v\}$, i.e. member *M* encrypts the next token $\{H_1, ---, H_v\}$ into a set of their mutually independent linear combinations, by using coefficients $\{b_{jk}\}$ that are secret from others.

An important thing is that tokens in an implicit transaction link are multiplied by random numbers $r_i$ and $r_{i+1}$ secret from members. Here, recycle center *C* determines these random numbers based on the number of transactions executed by *M*, so that the same random numbers $r_{i+1}$ is assigned to the next and the current tokens in the *i*-th and the (*i*+*1*)-th implicit transaction links of *M*. Consequently, the sums of the current tokens and the next tokens in *M*'s implicit transaction links must coincide when *M* maintains its transaction records honestly, as shown in Fig. 6. Also, because *M* knows neither of the random numbers in the implicit transaction links nor the coefficients of transaction records, it is impossible for *M* to modify its transaction record consistently. Therefore, *C* can detect dishonest operations of *M*, e.g. deletion addition, modification or forgery of transaction records, by checking implicit transaction links. However, *C* cannot know linkages of transactions executed by *M*, even *C* generates and maintains implicit transaction links of *M*.

Recycle center *C* detects dishonest operations of member *M* as follow; firstly, *C* decrypts the sum of transaction records that is sent from *M* to extract $T_C = (t_{C1} + t_{C2} + --- + t_{C(Q-1)})$, the sum of the current tokens in implicit transaction links, and $T_N = (t_{N1} + t_{N2} + --- + t_{N(Q-1)})$, the sum of the next tokens in implicit transaction links, and sends $T_N$ to *M*. Here, $t_{Ci}$ and $t_{Ni}$ are the current and the next tokens in the implicit transaction link of *M*'s *i*-th transaction record (when $t_{Ni}$ is decrypted to $D(t_{Ni})$ by *M*'s secret key, $D(t_{Ni})$ coincides with $t_{C(i+1)}$). Then *M* decrypts $T_N$ into $D(T_N)$ by its secret key, and finally *C* compares $(T_C + t_{CQ})$ with $(t_{C1} + D(T_N))$, and detects dishonesties when they do not match (*M* shows its initial and last tokens $t_{C1}$ and $t_{CQ}$ at the account calculation phase to enable *C* to execute this comparison).

Implicit transaction links included in transaction records enable the center to detect malicious operations of members also in the auction part, as described in the next sub-section.



Implicit transaction link for the *i*-th transaction

Figure 5. Implicit transaction link

| | |
|---|---|
| *1*st current token<br>$t_{C1}$ | *1*st next token<br>$t_{N1}$ |
| *2*nd current token<br>$t_{C2}$ (=$D(t_{N1})$) | *2*nd next token<br>$t_{N2}$ |
| *3*rd current token<br>$t_{C3}$ (=$D(t_{N2})$) | *3*rd next token<br>$t_{N3}$ |
| (*Q*-*1*)-th current token<br>$t_{C(Q-1)}$ (=$D(T_{N(Q-2)})$) | (*Q*-*1*)-th next token<br>$t_{N(Q-1)}$ |

$$\sum_{i=1}^{Q-1} t_{Ci} + t_{CQ} = t_{C1} + \sum_{i=1}^{Q-1} D(t_{Ni})$$

Figure 6. Detection of dishonest transaction record

When *C* detects discrepancy between $T_C$ and $D(T_N)$, it can easily trace all transactions executed by *M*, by decrypting its maintaining implicit transaction links, because *M* that cannot show $D(T_N)$ consistently must disclose its secret key for decrypting next tokens in order to prove its honesty. About the *4*th requirement, it is straightforward to detect dishonest operations of the center, when a member receives a signed receipt that includes implicit transaction links from the center for its every transaction. Also it is obvious that no absolutely trustworthy entity is included in Fig. 4.

### *D. Auction Part*

Products to be recycled are exchanged between sellers and buyers through auctions, and these auctions must be conducted anonymously and fairly, i.e. they must satisfy the following properties.

1. Only authorized buyers can make bids in the auctions, but identities of buyers that make individual bids including winning bids must be concealed from others,
2. Individual bids that are made by same buyers in auctions must not be linked each other,
3. Winners of auctions must be forced to buy their winning products, and
4. The recycle center must be forced to sell products to auction winners.

In the following, the anonymous auction mechanism in [15] is enhanced to satisfy the above requirements. The blind signature and implicit transaction links play main roles. Namely, the anonymous authentication allows only authorized entities to participate in auctions while maintaining their anonymities, and the blind signature disables anyone to link sequences of bids made by same buyers. Also it enables recycle center *C* to force auction winners to buy their winning products. Implicit transaction links together with the blind signature force *C* to sell products to auction winners. In addition, in order to enable all participants to confirm that auctions are carried out fairly, every state of auctions, e.g. the current bid price, is disclosed in the public bulletin board (BBS). Here, only the recycle center *C* can write date on the BBS, but anyone can read them at anytime; therefore C cannot modify data in the BBS dishonestly, because always someone may be watching the BBS.

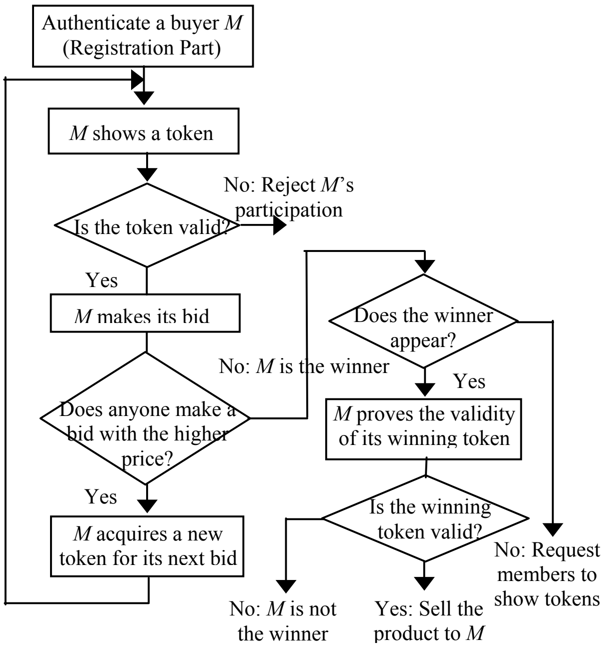

Authenticate a buyer *M* (Registration Part)
*M* shows a token
Is the token valid?
No: Reject *M*'s participation
Yes
*M* makes its bid
Does anyone make a bid with the higher price?
No: *M* is the winner
Yes
*M* acquires a new token for its next bid
Does the winner appear?
Yes
*M* proves the validity of its winning token
Is the winning token valid?
No: Request members to show tokens
No: *M* is not the winner
Yes: Sell the product to *M*

Figure 7. Behavior of the auction part

Fig. 7 describes the behavior of the auction part. In the figure, bids in auctions are considered as transactions; therefore buyer *M* can make bids only by showing unused tokens that have signatures of *C*, and it can get a new token for its next bid after the completion of its previous bid as in the bill making and payment part (*C* sends the new token to *M* as a multicast message while encrypting it by keys designated by *M* to maintain anonymity of *M*). The important thing is that a new token is given to *M* only when other buyer makes a bid for the product with the higher price, except the initial token for the auction.

The requirements listed in the beginning of this sub-section are satisfied as follows, i.e. recycle center *C*'s signature on tokens together with the anonymous authentication, allows only authorized members to participate in auctions, and because tokens that are attached to individual bids are blindly signed by *C*, buyer *M* can conceal not only its identity but also the linkage between its bids from others. Moreover, *M* is forced to buy the product when it makes the highest bid, because *M* discloses its tokens at its bids and can get its new and unused tokens only when other buyers make bids with the higher prices. Namely, even when *M* does not appear to buy its winning product, *C* can determine that *M* is an auction winner by asking all authorized members to show their last tokens. *M* cannot show its unused token, because it does not have an unused token when no one made bid with the higher price than it. Here, participants are not required to show their last tokens every time when a buyer with the highest bid does not appear to buy its winning product. Because bids in auctions are considered as transactions, this last token showing process can be combined with the total transaction amount calculation process in the bill making and payment part, i.e. it is enough for *C* to ask all members to show their last tokens as a part of bill making and payment process at the end of its every business period.

Winners of auctions can force *C* to sell them their winning products by showing their winning tokens (tokens that are attached to the winning bids). Although any entity can show winning tokens to *C*, because winning tokens are disclosed to the public through the BBS, *C* can determine the correct owner of the winning token by checking consistencies of implicit transaction links maintained by the winners in the same way as described in the bill making and payment part.

### *E. Anonymous Network*

Among of various kinds of mechanisms that enable entities to send messages without disclosing their identities, Mix-net is the most known one [1, 11, 13]. It conceals senders of messages as follow. A sender encrypts its message repeatedly by using public keys of multiple mix-servers while combining secret random numbers with the keys, and put the result into a sequence of mix-servers as shown in Fig.8. Then, each mix-server in the sequence decrypts its receiving messages by using its secret key and sends the decrypted results to its neighboring mix-server while randomly changing the message sending order from the receiving order. Therefore, the message is successfully decrypted to be sent to its receiver. On the other hand, it is not possible for any entity except the sender itself to identify the message sender unless all mix-servers conspire. However, because senders must know encryption keys of individual mix-servers, conventional Mix-nets must adopt asymmetric key encryption algorithms; therefore although they are appropriate for applications with short message exchanges such as ones in electric voting, it is difficult to use them for general applications with heavy message traffics because of large overheads in asymmetric key encryption and decryption processes.

To reduce this overhead, this sub-section enhances the performance of the symmetric key encryption based Mix-net (SEBM) proposed in [19]. Different from in usual Mix-nets, SEBM uses symmetric key encryption algorithms, e.g. messages are simply multiplied by secret random numbers; therefore entities can send large amount of messages anonymously with much less overheads.

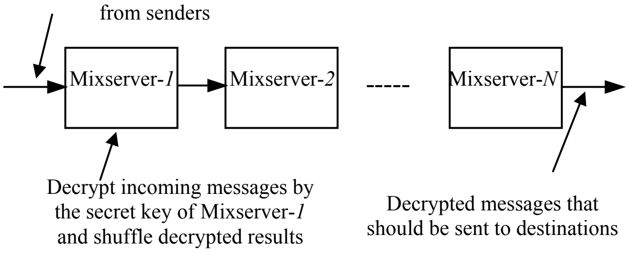

Repeatedly encrypted messages from senders
Mixserver-*1*
Mixserver-*2*
-----
Mixserver-*N*
Decrypt incoming messages by the secret key of Mixserver-*1* and shuffle decrypted results
Decrypted messages that should be sent to destinations

Figure 8. Mix-net

*SEBM*: SEBM behaves as shown in Fig.9. A fact that enables the use of symmetric key encryption algorithms is that individual message senders participate as mix-servers. Namely, the network consists of encryption and decryption parts both of them consist of the same set of servers (in the following, server-*i* in the encryption part and the decryption part is represented as encryption server-*i* and decryption server-*i*, respectively), and entity-$i_1$ that intends to start communication sends its message from encryption server-$i_1$ in the encryption part to randomly selected encryption server-$i_2$ while encrypting the message by its secret key $K_{i1}$. Then this process is iterated for *HTL* (hops to live) times (*HTL* is determined at random by server-$i_1$, and attached to the message, in the figure *HTL* is set to *N*), i.e. server-$i_q$ ($1 \leq q \leq N$) encrypts the received message by its secret key $K_{iq}$ and forwards the result to randomly selected server-$i_{(q+1)}$ (server-$i_N$ forwards the message to servers in the decryption part as shown later). Moreover, each server in the encryption part generates dummy messages and forwards them in the same way as its own messages and its receiving messages that come from other servers, instead of shuffling incoming messages.

When the message is forwarded to servers in the encryption part for *HTL* times, it is multicast to servers in the decryption part, and individual servers in the decryption part decrypt the message by their decryption keys and multicast the decryption results to other decryption servers. This process is iterated for the predefined number of times that is randomly determined by the last server in the encryption part and large enough compared with *HTL*; then there exists at least one message forwarding pass, in which a sequence of encryption servers and their corresponding decryption servers encrypt and decrypt the message. Therefore, the message repeatedly encrypted in the encryption part is decrypted successfully to be forwarded to the receiver. Here, encryption algorithms used in individual servers are selected so that repeatedly encrypted messages can be decrypted successfully regardless of the order of decryptions. An encryption algorithm used in SEBM, in which messages are simply multiplied by secret random numbers, satisfies this property.

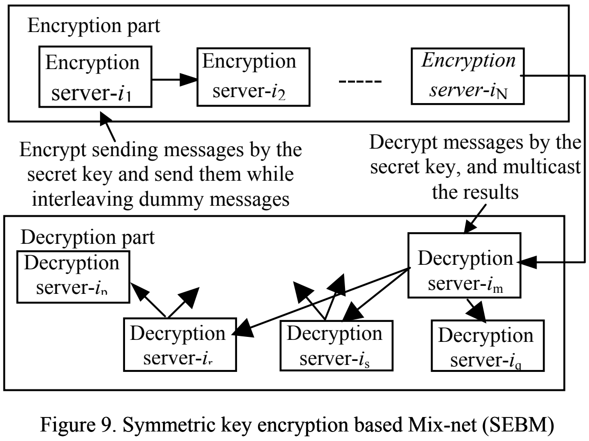


Figure 9. Symmetric key encryption based Mix-net (SEBM)

*Enhanced SEBM*: Two drawbacks of SEBM are that firstly it requires large number of multicasts and secondary the symmetric key encryption mechanisms, in which repeatedly encrypted messages are decrypted successfully regardless of the order of decryptions, are easy to break. In the case of SEBM, encryption based on simple secret number multiplication can be broken easily when same secret numbers are used for different messages. The enhanced SEBM proposed in this section removes these drawbacks by using anonymous tag lists.

An anonymous tag list is an array of anonymous tags that are attached to a message as shown in Fig. 10, and enables both removal of multicasts and use of different secret numbers for different messages. By the tag list, decryption server *S* can determine whether its receiving messages were encrypted by *S* in the encryption part or not, and the secret numbers that *S* used when it encrypted the messages in the encryption part. An encryption server adds an anonymous tag to the tag list of its receiving message, and encrypts the message and each tag in the tag list individually to be forwarded to its neighboring server (to conceal the number of servers that the message visited in the encryption part, the message sending server may attach dummy tags).

Different form the original SEBM in the enhanced SEBM, decryption servers are arrayed in a sequence as shown in Fig. 11, and repeatedly encrypted messages are forwarded through this sequence while being decrypted. Based on the anonymous tags, server *S* in the decryption part forwards its receiving message without any operation when its attached tag list does not include its tag (a tag that is generated by *S*). Also *S* determines the secret random number that it used in the encryption part, when the tag list includes its tag. Then because only and all servers in the decryption part, which encrypted the message in the encryption part, decrypt the message based on secret numbers that were used for encryptions, the repeatedly encrypted message can be successfully decrypted to the original one without any multicast. Moreover, although the massage encryption mechanism is simple, it is hard to break it, because different secret numbers are applied to different messages.

Anonymous tags must satisfy the following requirements, i.e.

1) Either of a server that generates an anonymous tag and the secret number used in the encryption part cannot be identified by any entity except the server that generates that tag, and
2) A server can identify its tags and the secret numbers used in the encryption part even the tags are encrypted and decrypted repeatedly by other servers.

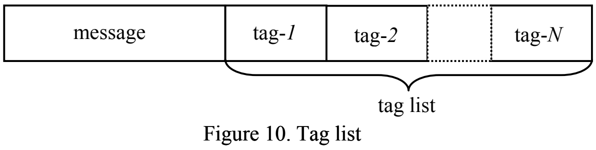


Figure 10. Tag list

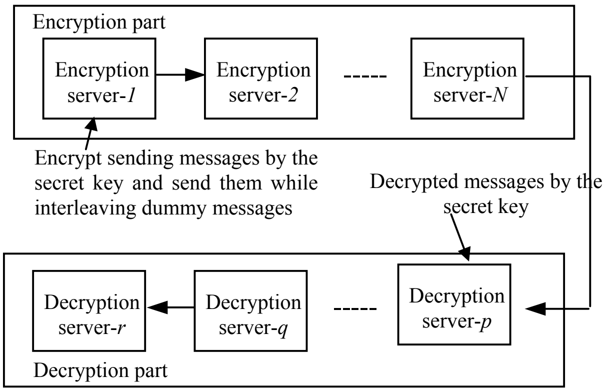


Figure 11. Enhanced SEBM

| Tag part | Encrypted tag part |
|---|---|
| $T$ | $\mathrm{E}_{\mathrm{K}}(T)$ |

Figure 12. Anonymous tag

In Fig. 12, an anonymous tag is implemented as a pair of tag and encrypted tag parts. In the tag part, tag $T$ is placed, and $T$ is encrypted to $\mathrm{E}_{\mathrm{K}}(T)$ by the server's secret key $K$ to be put in the encrypted tag part. Here, a server in the encryption or the decryption part encrypts/decrypts messages and individual tags in their tag lists separately, i.e. it encrypts/decrypts the massages by multiplying secret numbers while changing them for different messages, and encrypts/decrypts individual tags by secret keys that are corresponded to secret numbers used in message encryptions. Then the tag satisfies the above requirements as follow. Namely, although a tag attached to a message is encrypted or decrypted by servers repeatedly, server $S$ that generates the tag can identify its tag by decrypting the encrypted tag part by its secret keys and comparing the results with the tag part, because decryption result of the encrypted tag part coincides with the tag part only when the correct decryption key is used. Here, tags repeatedly encrypted by different servers can be successfully decrypted regardless of the order of servers that decrypt them. Also $S$ can identify the secret number used for the message encryption from the key that decrypts the encrypted tag part into the data coincides with the tag part, because there is an one to one mapping between keys for tag encryptions and secret numbers for message encryptions.

On the other hand, entities except $S$ cannot identify either of the secret number used for the message encryption and the linkage between the original and the repeatedly encrypted or decrypted tags, because the secret key used for the encrypted tag part calculation is known only to $S$. Here, it must be noticed that in order to maintain the strength of the encryption for the encrypted tag part calculation where the original and encrypted values are paired as the tag and the encrypted tag parts, asymmetric key encryption algorithms such as Elgamal must be used to generate encrypted tag parts and to encrypt and decrypt anonymous tags, instead of simple secret random number multiplications. However, the length of tags is not long; therefore communication efficiency can be maintained even asymmetric key encryption algorithms are used.

## V. Preliminary Feasibility Evaluation

Mechanisms adopted in the proposed information system platform such as blind signature are based on already established technologies except the linear combination based encryption algorithm and SEBM. Therefore, this section discusses the performances of the linear combination based encryption algorithm and SEBM based on a preliminary prototype system to evaluate the feasibility of the platform.

Fig. 13 shows the relation between the dimension of the linear equations adopted for the transaction record calculation and the single transaction execution time for a PC with *2.25*GHz CPU in the bill making and payment part, i.e. the duration from the member authentication to the transaction record registration. According to the results, a single transaction can be completed within less than a second even the dimension of the linear equations is *400*. Therefore, it is possible to execute transactions within practical times while maintaining enough strength of the encryption, i.e. when it is assumed that an actual transaction record is divided into *200* items and these items are mixed and shuffled with the other *200* dummy items, ${}_{400}\mathrm{C}_{200}$ number of variations must be examined to break the encryption mechanism.

According to Fig. 14, the relation between the dimension of the linear equations and the solution accuracy, messages encrypted by linear combination based encryption algorithms can be decrypted accurately enough even when the dimension of the equations becomes high, because all items in transaction records such as transaction amounts, tokens, dates, etc. can be represented as integers.

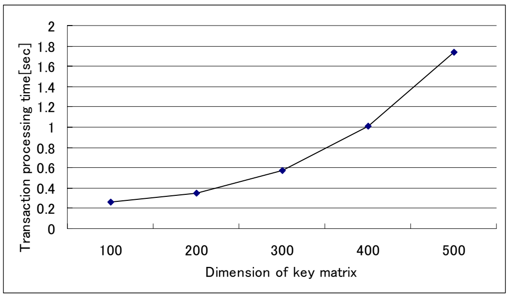


Figure 13. Transaction execution time

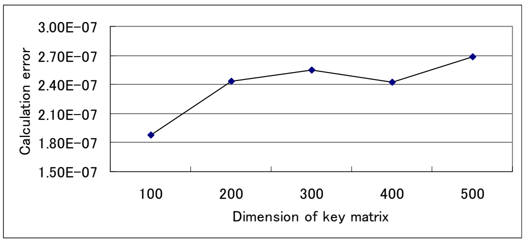


Figure 14. Solution accuracy of linear equations

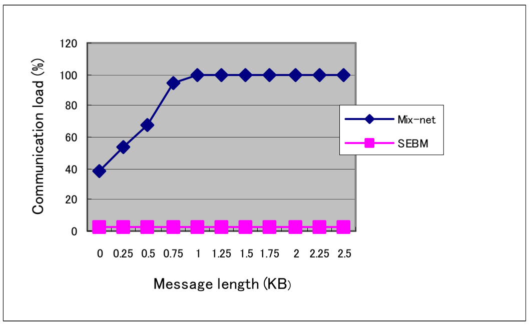


Figure 15. Communication processing load

To evaluate the feasibility of anonymous networks, experimental networks based on the conventional Mix-net and SEBM (multicast version), in which individual servers send their massages every *300*msec., are developed. In both networks, messages are forwarded through *3* servers equipped with *1.6*GHz CPUs. In SEBM case, this means that *HTL = 3*, i.e. messages are forwarded through *3* servers randomly selected from *9* servers in the encryption part, and servers in the decryption part forward their receiving messages to *8* other servers. Fig. 15 is the comparison of communication processing load between the conventional Mix-net and multicast version SEBM (vertical and horizontal axes represent the communication processing load and the message length, respectively). Different from the conventional Mix-net case, in which the communication load increases to *100*% as message length increases (this means that the conventional Mix-net cannot forward messages when their sizes become more than *1*K bytes), in SEBM case, the communication load stays low even the message size becomes long. Considering that messages are multicast frequently in the multicast version SEBM, the results shown in Fig.15 exhibit that the enhanced SEBM is practical enough, although the figure does not include affects caused by the anonymous tag processing required in enhanced SEBM.

## VI. Conclusion

Requirements for information systems that promote recycling of end products were discussed, and the anonymity was extracted as one of the most important issues to be considered. Namely, when products can be exchanged anonymously without any troubles that accompany this anonymity, primary barrier that makes people reluctant to exchange products to be recycled can be removed. This paper also showed that this requirement could be satisfied by the state of the art technologies.

To use the proposed mechanism for actual systems, discussions about the issue below are necessary. That is, to protect people from accidents caused by various defects of products, product manufacturers must be able to trace their products exchanged while maintaining anonymity of their owners.

## References

[1] D. Chaum, "Untraceable electronic mail, return address and digital pseudonyms," *Communications of the ACM*, Vol. 24, No.2, 1981, pp. 84-88.

[2] D. Chaum, "Security without identification: Transaction systems to make gig brother obsolete," *Communications of ACM*, Vol.28, No.10, 1985, pp.1030-1044.

[3] D. Cahum, "The dining cryptographers problem," *Journal of Cryptology*, 1988, pp.65-75.

[4] M. Franklin and M. Reiter, "The design and implementation of a secure auction service," *IEEE Trans. on Software Engineering*, Vol.22, 1996, pp.302-312

[5] S. Low, et al., "Anonymous credit cards and their collusion analysis," *IEEE/ACM Trans. on Networking*, Vol.4, No.6, 1996, pp809-816.

[6] M. Naor, B. Pinkas and R. Sumner, "Privacy preserving auctions and mechanism design," *The 1st ACM Conference on Electronic Commerce*, 1999.

[7] M. Reiter and A. Rubin, "Anonymous web transactions with Crowds," *Communications of the ACM*, Vol.42, No.2, 1999, pp.32-38.

[8] D. Goldschlag, M. Reed and P. Syverson, "Onion-Routing for anonymous and private internet connections," *Communications of the ACM*, Vol.42, No.2, 1999, pp.39-41.

[9] K. Omote and A. Miyaji, "A practical electronic English auction by using bulletin board," *Technical Report of IEICE,* ISEC2000-85*,* 2000.

[10] F. Stajano, "Security for ubiquitous computing," John Wiley & Sons, 2003, pp.152-165.

[11] B. LeeB. B. Dawson, K. Kim, J. Yang and S. Yoo, "Providing receipt-freeness in Mixnet-based voting protocols," *Information Security and Cryptography ICISC2003 6th International Conference*, 2003.

[12] R. Shigetomi, A. Otsuka, T. Ogawa and H. Imai, "Refreshable tokens and its applications to anonymous loans," *SCIS 2003*, 2003.

[13] T. Carroll and D. Grosu, "A secure and efficient voter-controlled anonymous election schemes," *International Conference of Information Technology*: *Coding and Computing (ITCC'05)*, Vol.1, 2005, pp.721-726.

[14] .S. Tamura and T. Yanase, "Information sharing among untrustworthy entities," *IEEJ Trans. EIS*, Vol.125, No.11, 2005, pp1767-1772.

[15] K. Hassan, S. Tamura and T. Yanase, "A mechanism for anonymous auction," *Proceedings of Asian Simulation Conference 2006 (Systems Modeling and Simulation)*, Springer, pp.233-237, 2006.

[16] S. Tamura, K. Kouro and T. Yanase, "Expenditure limits in anonymous credit card systems," *IEEE SMC 2006*, pp.1238-1243, 2006.

[17] S. Tamura and T. Yanase, "A mechanism for anonymous credit card systems," *IEEJ Trans. EIS*, Vol. 127, No.1, 2007, pp.81-87.

[18] S. Tamura and T. Yanase, "An information system for anonymous recycling of end products," *IEEE SMC 2007*, pp.2290-2295, 2007.

[19] K. Kouro, S. Tamura and T. Yanase, "Anonymous network for product recycling," *IEEE SMC 2007*, pp.2308-2312, 2007.

[20] T. Tsuji, T. Tsuchida and K. Higuchi, "Considerations on information strage and retrieval systems for recycling objects," *IEEE SMC 2007*, 2007.

[21] S. Tamura, H. Haddad, K. Kouro, H. Tsurugi, A. Rokibul, T. Yanase, and S. Taniguchi, “The anonymous memory,” unpublished.

**Shinsuke Tamura** was born in Hyogo, Japan on Jan. 16 1948, and received the B.S., M.S. and Dr. (Eng.) degrees in Control Engineering from Osaka University, Japan in 1970, 1972 and 1991, respectively.

During 1972 to 2001 he worked for Toshiba Corporation. He is currently a professor of the department of information science, University of Fukui, Fukui, Japan.

Prof. Tamura is a member of IEEJ, SICE and JSST.

**Kazuya Kouro** was born in Fukui, Japan on Oct. 26 1983, and received the B.S. and M.S. degrees in Information Engineering from University of Fukui, Japan in 2006 and 2008, respectively.

He is currently a student of a master’s program in graduate school of engineering, University of Fukui, Fukui, Japan.

**Masatoshi Sasatani** was born in Ishikawa, Japan on May 26 1984, and received the B.S. degree in Information Engineering from University of Fukui, Japan in 2007.

He is currently a student of a master’s program in graduate school of engineering, University of Fukui, Fukui, Japan.

**Kazi MD. Rokibul Alam** was born in Khulna, Bangladeshi on Oct. 31 1976, and received the B.S. degree from Khulna University and M.S. degree from Bangladeshi University of Engineering and Technology both in Computer Science and Engineering in 1999 and 2004, respectively.

He is currently a student of a doctoral program in graduate school of engineering, University of Fukui, Fukui, Japan.

Mr. Alam is a member of IEB.

**Hazim Anas Haddad** was born in Hama, Syria on Jan. 1 1982, and received the B.S. degree in Computer Science and Engineering from Ittihad University, United Arab Emirates in 2004 and M.S. degree in Nuclear and Safety Engineering from University of Fukui, Japan in 2008.

He is currently a student of a doctoral program in graduate school of engineering, University of Fukui, Fukui, Japan.