Information Security and Computer Fraud, 2015, Vol. 3, No. 1, 15-24
Available online at http://pubs.sciepub.com/iscf/3/1/3

DOI:10.12691/iscf-3-1-3

Development of Self-Issuable (Divisible and Transferable) Offline Electronic Cash

Shinsuke Tamura*, Hazim A. Haddad

School of Engineering, University of Fukui, Fukui, Japan
*Corresponding author: tamura@dance.plala.or.jp

Received July 23, 2015; Revised August 19, 2015; Accepted August 23, 2015

Abstract Based on anonymous tag based credentials and linear Mix-nets, this paper develops a scheme for e-cash systems that can be used in offline environments. The developed scheme makes e-cash holders anonymous while disabling them to use e-cash dishonestly. It also makes e-cash divisible and transferable. In detail, although no one except cash holders themselves can know correspondences between them and their e-cash, e-cash issuing authority can identify dishonest cash holders that had generated and/or spent e-cash illegitimately. In addition, cash holders can generate new e-cash of arbitrary values from their holding e-cash to make purchases of amounts less than original cash values. Also, cash holders can use e-cash that they had received from others as same as the one directly issued to them from the issuing authority.

Keywords: *privacy, anonymous tags, anonymous credentials, linear mix-nets*

Cite This Article: Shinsuke Tamura, and Hazim A. Haddad, "Development of Self-Issuable (Divisible and Transferable) Offline Electronic Cash." *Information Security and Computer Fraud*, vol. 3, no. 1 (2015): 15-24. doi: 10.12691/iscf-3-1-3.

1. Introduction

Together with credit card systems e-cash systems are one of the most convenient paying schemes in e-society and many schemes had been proposed already [1-7]. Among various features that existing e-cash schemes aim to achieve, anonymity, divisibility and transferability are most important, where anonymity ensures that correspondences between e-cash and their holders are not revealed, and divisibility and transferability enable each cash holder to divide its e-cash into ones with smaller cash values and to use e-cash that it had received from others, respectively.

But in offline environments where e-cash issuing authorities do not participate in individual purchases, to efficiently satisfy these requirements is not easy. Regarding the divisibility, existing schemes must assume the unit cash value for dividing original e-cash in advance and required computation and/or communication costs increase when the unit becomes small [7]. Also, several schemes cannot achieve complete anonymity, e.g. cash holders cannot conceal links among e-cash that they generated by dividing same e-cash [1]. In the same way, many existing transferable e-cash schemes cannot achieve complete anonymity [4,5]. For example, authorities can identify cash holders that used illegitimately transferred e-cash even if they were honest [5]. In addition, they are not convenient enough, e.g. volume of information that constitutes each e-cash increases every time when the e-cash is transferred [6] or cash holders must maintain numbers of receipts obtained from payees even after they had spent their e-cash [5].

While exploiting anonymous tag based credentials [8,9,11] and linear Mix-nets [10,11], this paper proposes a scheme that efficiently and effectively achieves the above 3 features, i.e. it achieves complete anonymity while enabling authorities to identify dishonest entities. Also, each cash holder can divide its e-cash into ones with arbitrary (even decimal) cash values to pay them to others provided that the total value of divided e-cash does not exceed the value of the original e-cash. In addition, volume of information that constitutes each e-cash or computation and communication cost for handling the e-cash does not increase even the e-cash is exchanged among many cash holders. Cash holders do not need to maintain numbers of receipts either.

2. Environments and Requirements

Figure 1 shows entities involved in the proposed offline e-cash scheme, they are e-ash issuing authority A and cash holders P_1, P_2, ---, P_M. Authority A issues e-cash to cash holders P_1, P_2, ---, P_M, and each P_m makes its purchase while paying its e-cash $C(P_m, t, h)$ to other cash holder P_k (P_m may simply give $C(P_m, t, h)$ to P_k). P_m also asks A to exchange its e-cash for real cash.

About e-cash, $C(P_m, t, h)$ means that P_m pays it to P_k as the h-th division of its t-th e-cash, and P_m obtains its t-th e-cash as $C(A, i)$ that was issued by authority A or $C(P_j, s, v)$ that was paid by other cash holder P_j. Where, $C(A, i)$ represents e-cash that authority A had issued directly to P_m as the i-th e-cash. Also an expiration time is defined for

each e-cash, and cash holders must exchange their e-cash for real cash at *A* before they expire.

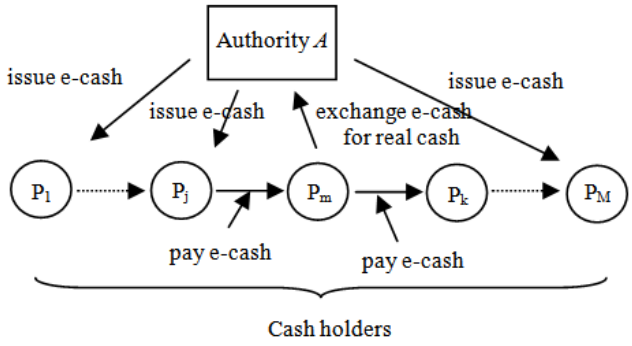

Figure 1. Entities in the proposed e-cash scheme

An important thing is offline e-cash system schemes must satisfy the following requirements under conditions that cash holders P_m, P_j and P_k are anonymous and they pay or give their e-cash to other cash holders without the presence of authority *A*, and the proposed e-cash system scheme satisfies these requirements by exploiting anonymous tag based credentials and linear Mix-nets as explained in the following sections. The requirements are,

1. *Unforgeability* Only cash issuing authority can generate valid e-cash,
2. *Anonymity* honest cash holders can conceal correspondences between them and their spending e-cash,
3. *Divisibility* cash holders can divide their e-cash into e-cash with smaller cash values,
4. *Transferability* cash holders can use e-cash that were paid to them by others,
5. *Unlinkability* cash holders can conceal links among e-cash that they had spent, and
6. *Security* cash holders cannot use e-cash illegitimately.

But about the security, illegitimate e-cash uses include double spending and using of e-cash owned by others, and in environments where individual cash holders are anonymous and authority *A* does not exist when cash holders pay their e-cash, payees cannot confirm that their receiving e-cash were certainly owned by payers or payers did not use same e-cash at other places. Therefore, to satisfy the security requirements, the proposed scheme detect illegitimately used e-cash after they were used and identify liable entities as same as other existing schemes.

3. Security Components

3.1. Anonymous Tag based Credentials

Provided that B, T_P, *k* and *g* are integers defined by authority *A*, R and *w* are secret integers defined by entity P, d_1 and d_2 are 2 secret signing keys of *A* and $S(d_1 \| d_2, x)$ is a pair of RSA signatures $\{S(d_1, x) = x^{d1}{}_{\text{mod B}}, S(d_2, x) = x^{d2}{}_{\text{mod B}}\}$, signature pair $T(A, T_P, R) = S(d_1 \| d_2, T_P{}^{R+1} K_w G_w{}^{R}{}_{\text{mod B}})$ is an anonymous tag based credential generated by *A* and given to P. Here, integers B, T_P, *k* and *g* are publicly known and they are defined so that to know integers q_1, q_{1*}, q_2, q_{2*}, q_3, q_{3*} that satisfy relations $T_P = k^{q1}{}_{\text{mod B}}$, $T_P{}^{q1*}{}_{\text{mod B}} = k$, $k = g^{q2}{}_{\text{mod B}}$, $k^{q2*}{}_{\text{mod B}} = g$, $g = T_P{}^{q3}{}_{\text{mod B}}$, $g^{q3*}{}_{\text{mod B}} = T_P$ is computationally infeasible for entities other than *A*. Different from B, *k* and *g* that are common to all credentials, T_P and R are unique to $T(A, T_P, R)$, and P calculates K_w and G_w as $K_w = k^w{}_{\text{mod B}}$ and $G_w = g^w{}_{\text{mod B}}$ based on publicly known *k*, *g* and secret integer *w*. About signing keys d_1 and d_2, they can be maintained as secrets of *A* despite multiple verification keys are publicly disclosed because the signer is only *A*. Also it is easy to maintain uniqueness of P's secret integer R as will be discussed in Sec. 4.1 [8,9,11]. In the remainder, notation $_{\text{mod B}}$ is omitted when confusions can be avoided.

Important things are, firstly P can prove that credential $T(A, T_P, R)$ is legitimate and it knows secret integer R in it by showing $T(A, T_P, R)^W = S(d_1 \| d_2, T_P{}^{R+1} K_w G_w{}^{R}{}_{\text{mod B}})^W$ without disclosing R itself, and secondly for given integer *U*, other entities can force P to calculate U^R honestly as a used seal of $T(A, T_P, R)$ while using secret integer R. Here, W is P's secret integer and entities that do not know W cannot know the correspondence between $T(A, T_P, R)$ and $T(A, T_P, R)^W$, i.e. to calculate W from $T(A, T_P, R)$ and $T(A, T_P, R)^W$ is a discrete logarithm problem. Also, only P that knows R can calculate U^R from *U*. Then, P can convince others that it is a legitimate owner of the credential without revealing its identity by showing $T(A, T_P, R)^W$. On the other hand, other entities can use used seal U^R as an evidence that P had certainly shown credential $T(A, T_P, R)$. But it must be noted that U^R and U^{R*} may have a same value despite $R \neq R^*$. Therefore to make used seals unique to credentials, actually $T(A, T_P, R)$ is implemented as a set of values $\{S(d_{11} \| d_{21}, T_P{}^{R+1} K_w G_w{}^{R}{}_{\text{mod B1}}), S(d_{12} \| d_{22}, T_P{}^{R+1} K_w G_w{}^{R}{}_{\text{mod B2}})\}$ by using different integers B_1 and B_2. Namely, relation $U^R{}_{\text{mod B1}} = U^{R*}{}_{\text{mod B1}}$ does not holed even when relation $U^R{}_{\text{mod B2}} = U^{R*}{}_{\text{mod B2}}$ holds.

In conclusion, together with used seal U^R credential $T(A, T_P, R)$ satisfies the following requirements. They are,

a) *Unforgeability* no one other than authority *A* can generate valid credentials,

b) *Soundness* entities that do not know integer R in credential $T(A, T_P, R)$ cannot prove the ownership of $T(A, T_P, R)$ to other entity Q. Also, when Q dishonestly accepts $T(A, T_P, R)$ shown by other credential holder P_* possibly while conspiring with it, *A* can detect that and identify liable entities,

c) *Anonymity* anyone except P cannot identify P from credential form $T(A, T_P, R)^W$,

d) *Unlinkability* even if P shows credential $T(A, T_P, R)$ n-times in forms $T(A, T_P, R)^{W1}$, $T(A, T_P, R)^{W2}$, ---, $T(A, T_P, R)^{Wn}$ while generating different secret integers W_1, W_2, ---, W_n, no one except P can know links between them,

e) *Revocability A* can invalidate credential $T(A, T_P, R)$ without knowing secrets of honest entities, if its holder P behaved dishonestly while showing $T(A, T_P, R)^W$ or if *A* reissued new credential to P as a replacement of $T(A, T_P, R)$, and

f) *Verifiability* anyone can verify the validity of credential $T(A, T_P, R)$, in other words, entities can verify the validity of $T(A, T_P, R)$ without knowing any secret of *A*.

3.2. Linear Mix-net

Linear Mix-net *L* consists of a sequence of mutually independent mix-servers L_1, L_2, ---, L_Z and enables authority *A* to calculate the sum of attribute values $D_P(1)$, $D_P(2)$, ---, $D_P(H_P)$ owned by same data holder P without knowing the correspondence between P and each $D_P(h)$ or

the calculated sum or links between individual attribute values $D_P(1)$, $D_P(2)$, ---, $D_P(H_P)$ [10,11].

Conceptually, P generates triplet {h, $D_P(h)$, $U(h)^R$} that includes its h-th attribute value $D_P(h)$ for each h, puts triplets {1, $D_P(1)$, $U(1)^R$}, {2, $D_P(2)$, $U(2)^R$}, ---, {H_P, $D_P(H_P)$, $U(H_P)^R$} in L separately without revealing its identity, and mix-servers L_1, L_2, ---, L_Z repeatedly encrypt each {h, $D_P(h)$, $U(h)^R$} to {h, $E(D_P(h))$, $U(h)^{R\cdot V}$} by their secret encryption keys while shuffling their encryption results. After that, authority A gathers encrypted triplets {1, $E(D_P(1))$, $U(1)^{R\cdot V}$}, {2, $E(D_P(2))$, $U(2)^{R\cdot V}$}, ---, {H_P, $E(D_P(H_P))$, $U(H_P)^{R\cdot V}$} that correspond to P and calculates sum $E(D_P(*)) = E(D_P(1))+E(D_P(2))+ \text{---} + E(D_P(H_P))$, constructs pair {$E(D_P(*))$, $U(*)^{R\cdot V}$}, and finally L_Z, L_{Z-1}, ---, L_1 repeatedly decrypt each pair {$E(D_P(*))$, $U(*)^{R\cdot V}$} to $\{D_P(*) = D_P(1)+D_P(2)+ \text{---} +D_P(H_P), U(*)^{R\cdot V\cdot Y}\}$ by their secret decryption keys while shuffling their decryption results.

In detail, P convinces L of its eligibility by showing credential $T(A, T_P, R)^{W(P, h)}$ while generating secret integer W(P, h) to put its h-th attribute value $D_P(h)$, and calculates used seal $U(h)^R$ of $T(A, T_P, R)$ from integer $U(h)$ defined by L_1, L_2, ---, L_Z. About encryptions and decryptions of $U(h)^R$ in {h, $D_P(h)$, $U(h)^R$} and $U(*)^{R\cdot V}$ in {$E(D_P(*))$, $U(*)^{R\cdot V}$}, L_1, L_2, ---, L_Z transform $U(h)^R$ and $U(*)^{R\cdot V}$ to $U(h)^{R\cdot V} = U(h)^{R\cdot V(1)\cdot V(2) \text{ ---} V(Z)}$ and $U(*)^{R\cdot V\cdot Y} = U(*)^{R\cdot V\cdot Y(Z)\cdot Y(Z-1) \text{ ---} Y(1)}$ by using their secret integers V(1), Y(1), V(2), Y(2), ---, V(Z), Y(Z).

Therefore, no one except P can know correspondences between P and each $D_P(h)$ or $D_P(*)$, or links between $D_P(1)$, $D_P(2)$, ---, $D_P(H_P)$ unless all L_1, L_2, ---, L_Z conspire (each L_z shuffles its encryption and decryption results). Nevertheless, A can identify triplets {1, $E(D_P(1))$, $U(1)^{R\cdot V}$}, ---, {N_P, $E(D_P(H_P))$, $U(H_P)^{R\cdot V}$} that correspond to same data holder P, and P can know the sum of its attribute values{$D_P(*)$, $U(*)^{R\cdot V\cdot Y}$}. To enable A to identify the triplets, provided that $U(0)$ is a publicly known integer, L_1, L_2, ---, L_Z defines their secret integers X(1, h), X(2, h), ---, X(Z, h) for each h and calculate $U(1)$, $U(2)$, ---, $U(H_P)$, as $U(1) = U(0)^{X(1, 1)\cdot X(2, 1) \text{ ---} X(Z, 1)}$, $U(2) = U(1)^{X(1, 2)\cdot X(2, 2) \text{ ---} X(Z, 2)}$, ---, $U(H_P) = U(H_{P-1})^{X(1, HP)\cdot X(2, HP) \text{ ---} X(Z, HP)}$. Then, L_1, L_2, ---, L_Z calculate $U(*)^{R\cdot V} = U(h)^{R\cdot V\cdot X(1, h+1) \text{ ---} X(Z, h+1)\cdot X(1, h+2) \text{ ---} X(Z, h+2) \text{ ---} X(1, HP) \text{ ---} X(Z, HP)}$ for each $U(h)^{R\cdot V}$, and A identifies triplets {1, $E(D_P(1))$, $U(1)^{R\cdot V}$}, ---, {H_P, $E(D_P(H_P))$, $U(H_P)^{R\cdot V}$} based on $U(*)^{R\cdot V}$ calculated from $U(1)^{R\cdot V}$, ---, $U(H_P)^{R\cdot V}$. On the other hand, P calculates $U(*)^{R\cdot V\cdot Y}$ from $U(*)^{V\cdot Y}$ by using its secret integer R to identify {$D_P(*)$, $U(*)^{R\cdot V\cdot Y}$}.

About calculation of $D_P(*) = D_P(1)+D_P(2)+ \text{---} +D_P(H_P)$, L_1, L_2, ---, L_Z encrypts each $D_P(h)$ by additive encryption functions, as a result, $E(D_P(1))+E(D_P(2))+ \text{---} +E(D_P(H_P))$ is decrypted to $D_P(*)$. A and L_1, L_2, ---, L_Z also can convince others of their honest handling of each triplet without disclosing their secret keys or integers provided that dishonesties bring losses to some entity.

4. Behaviors of the e-Cash Scheme

The proposed scheme of e-cash systems consists of 5 phases, i.e. registration, issuing, paying, reporting and verification phases. Here, authority A is accompanied by linear Mix-net L consists of mix-servers L_1, L_2, ---, L_Z. Also provided that H is the maximum number of payments that a single cash holder makes within a service period, each mix-server L_z maintains its secret integers X(z, 1), X(z, 2), ---, X(z, H), and for each h ($0 < h \le H$), L_1, L_2, ---, L_Z jointly calculates integer U_h as $U_h = U_{h-1}^{X(h)} = U_{h-1}^{X(1, h)\cdot X(2, h) \text{ ---} X(Z, h)}$ to be disclosed publicly, as in the previous section. Therefore, no one can know integer X(h) unless all L_1, L_2, ---, L_Z conspire. Then, first 4 phases proceed as shown in Figure 2.

An important thing is, different from other e-cash schemes, validity of e-cash that each cash holder P_m pays is not ensured by authority A's signature because in offline environments even A's signature cannot disable cash holders to spend same e-cash multiple times. Instead, validity of e-cash P_m pays is ensured by the anonymous signature of P_m, in other words, the issuer of e-cash P_m pays is P_m itself. Here, anonymous signatures enable signers to conceal their identities.

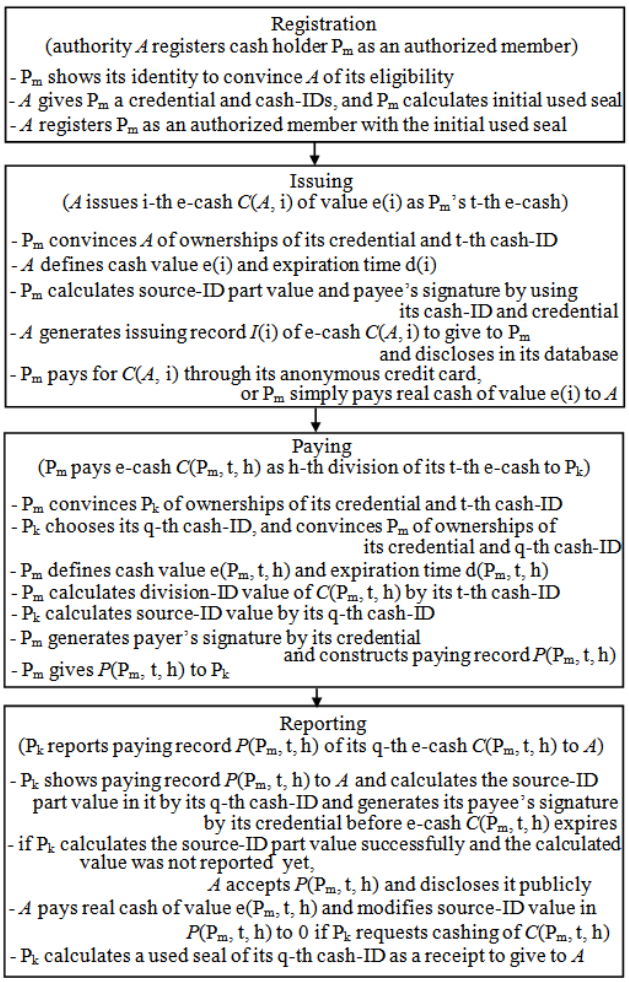

Figure 2. First 4 phases of the proposed e-cash scheme

4.1. Registration Phase

Each cash holder P_m registers itself and obtains its credential and cash-IDs while revealing its identity as below. In the following phases, P_m uses its credential and cash-IDs to show its eligibility and to divide its e-cash respectively of course without revealing its identity. P_m uses credentials also to generate its anonymous signatures. On the other hand, authority A uses anonymous signatures and cash-IDs to identify dishonest cash holders and to calculate total cash values of e-cash generated by dividing same e-cash, respectively.

1. Pm shows A its identity.
2. If Pm is eligible, A issues credential T(A, TPm, Rm) and cash-IDs T*(A, TPm(1), Rm(1)), T*(A, TPm(2), Rm(2)), ---, T*(A, TPm(H), Rm(H)) to Pm.
3. Pm calculates initial used seal U_Rm from publicly disclosed constant integer U_ by using T(A, TPm, Rm).
4. A registers Pm as an authorized member with initial used seal U_Rm.

Here, A uses initial used seal $U_{\_}^{Rm}$ to disable P_m to use credentials of other cash holders. In detail, at a time when A requests P_m to calculate a used seal of $T(A, T_{Pm}, R_m)$ from integer λ while showing its identity, A asks P_m to calculate a used seal also from integer $U_{\_}$. Namely, if P_m calculated λ^{R*} by using a credential of other entity, P_m calculates $U_{\_}^{R*}$ from $U_{\_}$ that is different from initial used seal $U_{\_}^{Rm}$. About cash-IDs, each $T_*(A, T_{Pm}(h), R_m(h))$ has the same structure as credential $T(A, T_{Pm}, R_m)$. But to discriminate credentials from cash-IDs A defines different signing keys, i.e. A uses keys pairs $\{d_1, d_2\}$ and $\{d_{1*}, d_{2*}\}$ to generate credentials and cash-IDs, respectively.

Secret unique integers $R_m = R_m(0), R_m(1), R_m(2)$, ---, $R_m(H)$ in the credential and cash-IDs are generated through the cooperation among L_1, L_2, ---, L_Z. Namely provided that Ω is a publicly known integer, for each h (h = 0, 1, 2, ---, H), each mix-server L_z generates secret integer $R_m(z, h)$ and calculates $\Omega^{Rm(z^*, h)} = \Omega^{Rm((z-1)^*, h) \cdot Rm(z, h)}$ based on $\Omega^{Rm((z-1)^*, h)}$ disclosed by L_{z-1}, and when finally disclosed $\Omega^{Rm(Z^*, h)}$ did not appear before each L_z informs P_m of $R_m(z, h)$ so that P_m can calculate unique secret integer $R_m(h)$ as product $R_m(h) = R_m(1, h) \cdot R_m(2, h)$ --- $R_m(Z, h)$. When $\Omega^{Rm(Z^*, h)}$ had appeared already, L_1, L_2, ---, L_Z generate different secret integers [8,11].

4.2, Issuing Phase

Authority A issues its i-th e-cash $C(A, i)$ to cash holder P_m as t-th e-cash of P_m without knowing P_m as below.

1. P_m generates secret integers $V_m(t)$ and $V^*_m(t)$, and convinces A of its ownerships of credential $T(A, T_{Pm}, R_m)$ and t-th cash-ID $T_*(A, T_{Pm}(t), R_m(t))$ by showing $T(A, T_{Pm}, R_m)^{Vm(t)}$ and $T_*(A, T_{Pm}(t), R_m(t))^{V^*m(t)}$.
2. A defines cash value e(i) and expiration time d(i).
3. P_m calculates source-ID part value $U_0^{Rm(t)}$ and payee's signature $\{U_0^{Rm(t)}+e(i)\|d(i)\}^{Rm}$ from integer U_0, e(i) and d(i) as used seals of $T_*(A, T_{Pm}(t), R_m(t))$ and $T(A, T_{Pm}, R_m)$. Here, notation $\|$ represents concatenation, and cash value e(i) and expiration time d(i) are integers.
4. If $U_0^{Rm(t)}$ did not appear before, A constructs cash issuing record $I(i) = <e(i), d(i), 0, U_0^{Rm(t)}, \{U_0^{Rm(t)}+e(i)\|d(i)\}^{Rm}>$ shown in Figure 3 (a) and gives $I(i)$ to P_m. A also discloses $I(i)$ publicly in its database.
5. P_m pays cash value e(i) through its anonymous credit card, or it simply pays real cash of value e(i) to A.

In the above, P_m chooses its t-th cash-ID for obtaining its t-th e-cash (actually P_m can choose an arbitrary cash-ID but step 4 disables P_m to use same cash-IDs repeatedly), as a result, P_m uses different cash IDs for different e-cash. Also, it changes values of secret integers $V_m(t)$ and $V^*_m(t)$ every time when it accesses A, therefore A cannot identify P_m or know links between individual e-cash P_m had obtained. But to maintain its anonymity P_m must buy $C(A, i)$ by its anonymous credit card or by paying real cash.

About the issuing record in Figure 3 (a), division-No. part value 0 represents that $C(A, i)$ is the original e-cash directly issued by A (i.e. it is the 0-th division). Together with division-ID part values in cash paying records shown in Figure 3 (b), source-ID part value $U_0^{Rm(t)}$ enables A to identify e-cash that are generated by dividing $C(A, i)$ without knowing links between $C(A, i)$ and individual divided e-cash. Payee's signature $\{U_0^{Rm(t)}+e(i)\|d(i)\}^{Rm}$ is used to identify dishonest cash holders without knowing secrets of honest entities, as will be discussed in Sec. 5.

Value part	Expiration time part	Division-No. part	Source-ID part
e(i)	d(i)	0	$U_0^{Rm(t)}$

Payee's signature part
$\{U_0^{Rm(t)}+e(i)\|d(i)\}^{Rm}$

(a) i-th cash issuing record $I(i)$ associated with cash-ID $T_*(A, T_{Pm}(t), R_m(t))$ and signed by credential $T(A, T_{Pm}, R_m)$

Value part	Expiration time part	Division-No. part	Division-ID part
$e(P_m, t, h)$	$d(P_m, t, h)$	h	$U_h^{Rm(t)}$

Source-ID part	Payer's signature part
$U_0^{Rk(q)}$	$\{U_h^{Rm(h)}+U_0^{Rk(q)} +e(P_m, t, h)\|d(P_m, t, h)\}^{Rm}$

(b) Paying record $P(P_m, t, h)$ associated with cash-IDs $T_*(A, T_{Pm}(t), R_m(t))$ and $T_*(A, T_{Pk}(q), R_k(q))$ and signed by credential $T(A, T_{Pm}, R_m)$

Figure 3. Configurations of cash records

Here, P_m calculates $U_0^{Rm(t)}$ and $\{U_0^{Rm(t)}+e(i)\|d(i)\}^{Rm}$ as used seals of $T_*(A, T_{Pm}(t), R_m(t))$ and $T(A, T_{Pm}, R_m)$, therefore although A does not know values of $R_m(t)$ or R_m, it can convince itself that P_m calculates them honestly. Also $\{U_0^{Rm(t)}+e(i)\|d(i)\}^{Rm}$ can be considered as an anonymous signature of P_m [11], i.e. no one except P_m can know P_m from credential form $T(A, T_{Pm}, R_m)^W$, $\{U_0^{Rm(t)}+e(i)\|d(i)\}^{Rm}$ can be calculated only by P_m that knows integer R_m, correct calculation of $\{U_0^{Rm(t)}+e(i)\|d(i)\}^{Rm}$ is ensured despite R_m is unknown, and by calculating a used seal of $T(A, T_{Pm}, R_m)$ from integer $U_0^{Rm(t)}+e(i)\|d(i)$, P_m can convince any entity that $\{U_0^{Rm(t)}+e(i)\|d(i)\}^{Rm}$ was generated by it.

About the expiration time d(i) of $C(A, i)$, because it may be used as a clue to identify P_m, all e-cash issued in a same service period have same expiration time. Also, $C(A, i)$ and all e-cash generated by dividing $C(A, i)$ have the same expiration time.

4.3. Paying Phase

Paying phase, in which P_m makes its purchase and pays e-cash $C(P_m, t, h)$ to other cash holder P_k by dividing e-cash $C(A, i)$ or $C(P_j, s, v)$, proceeds as follow. Here, $C(A, i)$ or $C(P_j, s, v)$ is P_m's t-th e-cash that it had received directly from issuing authority A or from other cash holder P_j respectively. Here as a special case, to exchange a part of $C(A, i)$ or $C(P_j, s, v)$ for real cash, P_m pays $C(P_m, t, h)$ to itself. In the following, it is assumed that P_m generates $C(P_m, t, h)$ as the h-th division of $C(A, i)$ or $C(P_j, s, v)$, and P_k receives $C(P_m, t, h)$ as its q-th e-cash. About cash-IDs,

although P_k below chooses its q-th cash-ID to receive its q-th e-cash, it can choose an arbitrary unused cash-ID.

1. P_m generates secret integers $W_m(t, h)$ and $W^*_m(t, h)$, shows credential $T(A, T_{Pm}, R_m)$ and t-th cash-ID $T_*(A, T_m(t)\ R_m(t))$ in forms $T(A, T_{Pm}, R_m)^{Wm(t, h)}$ and $T_*(A, T_{Pm}(t), R_m(t))^{W*m(t, h)}$ and convinces P_k of its eligibility and the ownership of $T_*(A, T_{Pm}(t), R_m(t))$.
2. P_k generates secret integers $V_k(q)$ and $V^*_k(q)$, chooses q-th cash-ID $T_*(A, T_k(q), R_k(q))$, shows credential $T(A, T_{Pk}\ R_k)$ and $T_*(A, T_k(q), R_k(q))$ in forms $T(A, T_{Pk}, R_k)^{Vk(q)}$ and $T_*(A, T_{Pk}(q), R_k(q))^{V*k(q)}$, and convinces P_m of its eligibility and the ownership of $T_*(A, T_{Pk}(q), R_k(q))$.
3. P_m defines cash value $e(P_m, t, h)$ and expiration time $d(P_m, t, h)$, calculates division-ID part value $U_h^{Rm(t)}$ from integer U_h by using cash-ID $T_*(A, T_{Pm}(t), R_m(t))$, and informs P_k of quadruplet $<e(P_m, t, h), d(P_m, t, h), h, U_h^{Rm(t)}>$.
4. P_k calculates source-ID part value $U_0^{Rk(q)}$ from U_0 as a used seal of $T_*(A, T_{Pk}(q), R_k(q))$.
5. P_m calculates payer's signature $S(R_m, G(P_m, t, h)) = G(P_m, t, h)^{Rm}$ as a used seal of $T(A, T_{Pm}, R_m)$, and constructs paying record $P(P_m, t, h) = <e(P_m, t, h), d(P_m, t, h), h, U_h^{Rm(t)}, U_0^{Rk(q)}, S(R_m, G(P_m, t, h))>$ to give P_k as shown in Figure 3 (b). Where, $G(P_m, t, h) = U_h^{Rm(t)} + U_0^{Rk(q)} + e(P_m, t, h) \| d(P_m, t, h)$.

In the above, P_m and P_k show their credentials and cash-IDs while modifying them by their secret integers $W_m(t, h)$, $W^*_m(t, h)$, $V_k(q)$ and $V^*_k(q)$, therefore P_m and P_k can make them anonymous as same as in the issuing phase. Also P_m can conceal links among e-cash it generates from $C(A, i)$ or $C(P_j, s, v)$. Nevertheless, P_k can confirm correct calculations of division-ID part value $U_h^{Rm(t)}$ and payer's signature $S(R_m, G(P_m, t, h))$ because P_m calculates them as used seals of its cash-ID and credential. In the same way, P_m can confirm P_k's correct calculation of source-ID part value $U_0^{Rk(q)}$.

Here as discussed in the previous subsection, together with the source-ID part value in issuing record $I(i)$ and/or paying record $P(P_j, s, v)$, division-ID part value $U_h^{Rm(t)}$ in $P(P_m, t, h)$ enables A to confirm that P_m had honestly divided its t-th e-cash $C(A, i)$ or $C(P_j, s, v)$ without knowing P_m, $C(A, i)$ or $C(P_j, s, v)$. A can also identify dishonest entities by exploiting payer's signatures in paying records together with payee's signatures that are calculated at step 2 in the reporting phase.

About dishonesties of cash holders, because P_m and P_k interact under offline environments, both of P_m and P_k can forge paying records without being noticed by P_k or P_m, e.g. P_m may calculate the division-ID part value while using a cash-ID different from $T_*(A, T_m(t)\ R_m(t))$, also P_k may modify the cash value, the expiration time or the division-ID part value in $P(P_m, t, h)$ after it had received it from P_m. But these dishonesties are detected and liable entities are identified while exploiting division-ID and source-ID parts values and payer's and payee's signatures as will be discussed in Sec. 5.

4.4. Reporting Phase

In this phase, cash holder P_k that received its q-th e-cash $C(P_m, t, h)$ from other cash holder P_m as paying record $P(P_m, t, h) = <e(P_m, t, h), d(P_m, t, h), h, U_h^{Rm(t)}, U_0^{Rk(q)}, S(R_m, G(P_m, t, h))>$, reports $P(P_m, t, h)$ to A by the expiration time of $C(P_m, t, h)$. Also, A pays real cash of value $e(P_m, t, h)$ to P_k, if P_k wants cashing of $C(P_m, t, h)$. The reporting phase proceeds as below.

1. P_k generates secret integers $W_k(q, 0)$ and $W^*_k(q, 0)$, and shows its credential and q-th cash-ID in forms $T(A, T_{Pk}, R_k)^{Wk(q, 0)}$ and $T_*(A, T_k(q), R_k(q))^{W*k(q, 0)}$ to convince A of its eligibility and the ownership of the cash-ID.
2. P_k shows paying record $P(P_m, t, h)$ to A, and calculates source-ID part value $U_0^{Rk(q)}$ from U_0 and payee's signature $S(R_k, S(R_m, G(P_m, t, h))) = S(R_m, G(P_m, t, h))^{Rk} = G(P_m, t, h)^{Rm \cdot Rk}$ from $S(R_m, G(P_m, t, h))$ by its q-th cash-ID and credential, respectively.
3. If P_k calculates source-ID part value $U_0^{Rk(q)}$ successfully and $U_0^{Rk(q)}$ did not appear before, A accepts $P(P_m, t, h)$ and discloses it publicly with the payee's signature, provided that $P(P_m, t, h)$ does not expire.
4. When P_k wants cashing and no one exchanged $C(P_m, t, h)$ for real cash yet, A pays real cash of value $e(P_m, t, h)$ to P_k. A also modifies source-ID part value in $P(P_m, t, h)$ to 0. Where, source-ID part value 0 means P_k cannot generate new e-cash from $C(P_m, t, h)$.
5. P_k calculates used seal of its q-th cash-ID from integer U_-, i.e. $U_-^{Rk(q)}$, as a receipt, and A discloses $P(P_m, t, h)$ with the receipt publicly.

In the above, because source-ID part value $U_0^{Rk(q)}$ can be calculated only by cash-ID $T_*(A, T_k(q), R_k(q))$ and P_k must calculate it honestly, anyone except P_k cannot report $P(P_m, t, h)$ or exchange $C(P_m, t, h)$ for real cash unless it is conspiring with P_k. In the same way, receipt $U_-^{Rk(q)}$ disables P_k to exchange $C(P_m, t, h)$ for real cash repeatedly. Also, P_k can convince A of correct calculation of payee's signature without revealing its identity or secret integer R_k, then it can maintain its anonymity as same as in the paying phase.

About dishonest cash holders, because A discloses paying record $P(P_m, t, h)$ at step 3, cash holders become able to use division-ID part value $U_h^{Rm(t)}$ calculated by P_m to generate or modify their paying records. But it must be noted that step 3 disables cash holders to use same source-ID part values repeatedly. As a result, all paying records have different source-ID part values, and this means cash holders including P_k cannot forge paying records that include consistent payer's signatures of P_m without the help of P_m (only P_m can generate consistent signatures of P_m on information that includes newly appearing source-ID part values). Therefore, when multiple paying records with consistent payer's signatures and same division-ID part value $U_h^{Rm(t)}$ are detected, A can regard P_m as the liable entity. In other words, entities other than P_m cannot generate a paying record with division-ID part value $U_h^{Rm(t)}$ and consistent payer's signature of P_m without the help of P_m despite $U_h^{Rm(t)}$ is disclosed.

5. Verification Phase

The verification phase that consists of dishonest record detection and dishonest entity identification stages detects dishonesties of entities and identifies entities that are liable for them.

In offline environments, payees cannot know whether payers are generating new e-cash so that their cash values

do not exceed cash values of original e-cash or not. Authority *A* cannot sign on paying records that payers generate either. As a result, cash holders can behave dishonestly without being noticed by others. As shown below, there are 8 kinds of possibilities that entities involved behave dishonestly. Here in the following it is assumed that to pay P_k cash holder P_m generates e-cash $C(P_m, t, h)$ and corresponding cash paying record $P(P_m, t, h)$ from e-cash $C(P_j, s, v)$ that P_m had obtained from P_j. But illegitimate paying records are detected and liable entities are identified in the same way also in cases where P_m generates $C(P_m, t, h)$ from $C(A, i)$ directly issued from *A*.

a. P_m generates e-cash from $C(P_j, s, v)$ more than the cash value of $C(P_j, s, v)$.

b. P_m generates paying record $P(P_m, t, h)$ to give to P_k while forging the expiration time or calculating a division-ID part value by a cash-ID different from $T_*(A, T_{Pm}(t), R_m(t))$ (but P_m must use its cash-ID if P_k is honest, because P_k examines the division-ID part value at step 3 in the paying phase).

c. P_m does not report paying record $P(P_j, s, v)$ to *A*.

d. P_k reports paying record $P(P_m, t, h)$ to *A* while modifying it, e.g. changing its cash value, expiration time or division-ID part value. Or to use e-cash $C(P_m, t, h)$ repeatedly, it reports $P(P_m, t, h)$ multiple times while changing source-ID part values (source-ID part values must be unique to individual e-cash).

e. Other cash holder P_j reports $P(P_m, t, h)$ to *A* while changing the source-ID part value in it. Where, P_j can obtain $P(P_m, t, h)$ because *A* discloses it at step 3 in the reporting phase, or P_j may steal it by eavesdropping on interactions between P_m and P_k. But P_j must calculate the source-ID part value by its cash-ID because *A* examines the consistency between the source-ID part value and the cash-ID of P_j.

f. Authority *A* arbitrarily forges paying records.

Namely, because P_m or P_k cannot know e-cash generated and used in other places in offline environments and *A* in the reporting phase can examine only source-ID part values and payee's signatures in individual paying records, P_m can generate e-cash from $C(P_j s, v)$ more than its cash value, can generate $P(P_m, t, h)$ while using a cash-ID different from $T_*(A, T_m(t), R_m(t))$ used to calculate the source-ID part value of $P(P_j, s, v)$, or may not report $P(P_j, s, v)$ to *A*. Also, P_k can report $P(P_m, t, h)$ to *A* while modifying its cash value, expiration time and/or division-ID part value.

As the more vicious dishonesty, when other cash holder P_j steals $P(P_m, t, h)$ given to P_k and reports it to *A* (of course while changing the source-ID part value), P_j becomes able to generate e-cash from P_k's e-cash $C(P_m, t, h)$. Also, when authority *A* is dishonest, it can forge paying records arbitrarily to be registered in its database. Where about conspiracy between P_m and P_k, although it enables them to forge consistent paying records, they cannot reap any benefit as a total, i.e. one of P_m and P_k must compensate losses caused by the forged e-cash.

But it must be noted that cash holders must report paying records while generating source-ID part values and payee's signatures by their legitimate cash-IDs and credentials, i.e. *A* examines them at steps 2 and 3 in the reporting phase. In addition, all paying records must have different source-ID values, and once authority *A* had accepted paying records, anyone including *A* and mix-servers in Mix-net *L* must honestly handle them, i.e. all paying records are publicly disclosed, and honest encryptions and decryptions of *L* are ensured [10].

Then, without knowing secrets of honest entities, the dishonest record detection and the dishonest entity identification stages become able to detect above dishonesties and identify liable entities as in the following subsections.

5.1. Dishonest Record Detection Stage

The dishonest record detection stage detects dishonesties while exploiting division-ID and source-ID parts values in paying records and issuing records. Namely, although additional mechanisms are necessary so that P_m can conceal links between $C(P_m, t, 1)$, $C(P_m, t, 2)$, ---, $C(P_m, t, H(m, t))$ it had generated from same e-cash $C(P_j, s, v)$ from others, while exploiting relation $U_h^{Rm(t)\cdot X(h+1)\cdot X(h+2)\ \cdots\ X(H)} = U_0^{Rm(t)\cdot X(1)\cdot X(2)\ \cdots\ X(H)} = U_0^{Rm(t)\cdot X^*}$, *A* can determine paying record $P(P_m, t, h)$ accompanied by division-ID value $U_h^{Rm(t)}$ was generated from $P(P_j, s, v)$ when its source-ID part value $U_0^{Rm(t)}$ is converted to $U_0^{Rm(t)\cdot X^*}$. Therefore, if *A* gathers $P(P_m, t, 1)$, $P(P_m, t, 2)$, ---, $P(P_m, t, H(m, t))$, and compares cash values of $C(P_m, t, 1)$, $C(P_m, t, 2)$, ---, $C(P_m, t, H(m, t))$ and $C(P_j, s, v)$, *A* can examine whether all $C(P_m, t, 1)$, $C(P_m, t, 2)$, ---, $C(P_m, t, H(m, t))$ were honestly generated and handled or not. Here, if P_m did not report $P(P_j, s, v)$ to *A* or $P(P_j, s, v)$ had expired already, *A* cannot use relation $U_h^{Rm(t)\cdot X(h+1)\ \cdots\ X(H)} = U_0^{Rm(t)\cdot X^*}$ to find $C(P_j, s, v)$ from which $C(P_m, t, h)$ was generated. But in this case, *A* identifies each $C(P_m, t, h)$ as an orphan paying record that cannot be corresponded to any paying or issuing record. As a result, every kind of dishonesties at the beginning of this section can be detected as inconsistent division-IDs and source-ID pairs or orphan paying records.

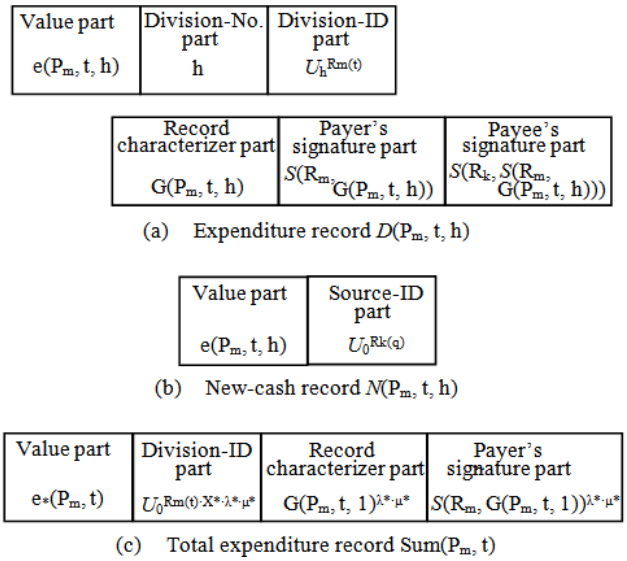

Figure 4. Expenditure, new-cash and total expenditure records

To implement the above strategies, authority *A* constructs expenditure record $D(P_m, t, h) = \langle e(P_m, t, h), h, U_h^{Rm(t)}, G(P_m, t, h), S(R_m, G(P_m, t, h)), S(R_k, S(R_m, G(P_m, t, h)))\rangle$ and new-cash record $N(P_m, t, h) = \langle e(P_m, t, h), U_0^{Rk(q)}\rangle$ from paying record $P(P_m, t, h) = \langle e(P_m, t, h), d(P_m, t, h), h, U_h^{Rm(t)}, U_0^{Rk(q)}, S(R_m, G(P_m, t, h))\rangle$ as shown in

Figure 4 (a) and (b). Here, record characterizer part value $G(P_m, t, h)$ and payer's signature $S(R_m, G(P_m, t, h))$ are $U_h^{Rm(t)}+U_0^{Rk(q)}+e(P_m, t, h)\|d(P_m, t, h)$ and $G(P_m, t, h)^{Rm}$ respectively, and payee's signature $S(R_k, S(R_m,G(P_m, t, h)))$ $= G(P_m, t, h)^{Rm\cdot Rk}$ is calculated when P_k reports $P(P_m, t, h)$ to A. In addition to the above records, total expenditure records are defined as shown in Figure 4 (c).

Authority A detects inconsistent division-IDs and source-ID pairs and orphan paying records without knowing secrets of honest entities while exploiting linear Mix-net $L = \{L_1, L_2, ---, L_Z\}$ as follows.

1. For each paying record $P(P_m, t, h) = <e(P_m, t, h), d(P_m, t, h), h, U_h^{Rm(t)}, U_0^{Rk(q)}, S(R_m, G(P_m, t, h))>$, which is valid in a considering service period, A constructs expenditure record $D(P_m, t, h) = <e(P_m, t, h), h, U_h^{Rm(t)}, G(P_m, t, h), S(R_m, G(P_m, t, h)), S(R_k, S(R_m, G(P_m, t, h)))>$ and new-cash record $N(P_m, t, h) = <e(P_m, t, h), U_0^{Rk(q)}>$, and put $D(P_m, t, h)$ in linear Mix-net L. Here, $S(R_k, S(R_m, G(P_m, t, h)))$ is the payee's signature accompanying $P(P_m, t, h)$.
2. Mix-servers $L_1, L_2, ---, L_Z$ repeatedly encrypt expenditure records put by A while shuffling their encryption results. Here, value part $e(P_m, t, h)$ in each $D(P_m, t, h)$ is encrypted to $E(e(P_m, t, h))$ by additive encryption functions of $L_1, L_2, ---, L_Z$, but L does not encrypt division-No. part value h as the attribute No. in Sec. 3.2. About division-ID part value $U_h^{Rm(t)}$, provided that $\lambda(1), \lambda(2), ---, \lambda(Z)$ are secret integers of $L_1, L_2, ---, L_Z$, $\lambda_* = \lambda(1)\cdot\lambda(2)$ --- $\lambda(Z)$, $X_*(z, h) = X(z, h)\cdot X(z, h+1)$ --- $X(z, H)$ and $X_* = X(1)\cdot X(2)$ --- $X(H)$, each L_z transforms (encrypts) $U_h^{Rm(t)\cdot\{X*(1,\ h+1)\cdot\lambda(1)\}\cdot\{X*(2,\ h+1)\cdot\lambda(2)\}---\{X*(z-1,\ h+1)\cdot\lambda(z-1)\}}$ calculated by L_{z-1} to $U_h^{Rm(t)\cdot\{X*(1,\ h+1)\cdot\lambda(1)\}\cdot\{X*(2,\ h+1)\cdot\lambda(2)\}---\{X*(z-1,\ h+1)\cdot\lambda(z-1)\}\cdot\{X*(z,\ h+1)\cdot\lambda(z)\}}$, while referring to division No. part value h. As a result $U_h^{Rm(t)}$ is transformed to $U_h^{Rm(t)\cdot X(h+1)\cdot X(h+2)---X(H)\cdot\lambda(1)\cdot\lambda(2)---\lambda(z)} = U_0^{Rm(t)\cdot X*\cdot\lambda*}$ (as shown in Sec. 4, $X(h) = X(1, h)\cdot X(2, h)$ --- $X(Z, h)$). In the same way, $L_1, L_2, ---, L_Z$ transform record characterizer part value $G(P_m, t, h)$, payer's signature $S(R_m, G(P_m, t, h))$ and payee's signature $S(R_k, S(R_m, G(P_m, t, h)))$ to $G(P_m, t, h)^{\lambda*}$, $S(R_m, G(P_m, t, h))^{\lambda*}$ and $S(R_k, S(R_m, G(P_m, t, h)))^{\lambda*}$. In conclusion, $D(P_m, t, h)$ is encrypted to $E(D(P_m, t, h)) = <E(e(P_m, t, h)), h, U_0^{Rm(t)\cdot X*\cdot\lambda*}, G(P_m, t, h)^{\lambda*}, S(R_m, G(P_m, t, h))^{\lambda*}, S(R_k, S(R_m,G(P_m, t, h)))^{\lambda*}>$.
3. A gathers encrypted expenditure records $E(D(P_m, t, 1)), E(D(P_m, t, 2)), ---, E(D(P_m, t, H(m, t))$ that include same division-ID part value $U_0^{Rm(t)\cdot X*\cdot\lambda*}$, calculates $E(e(P_m, t, 1))+E(e(P_m, t, 2))+ --- +E(e(P_m, t, H(m, t))) = E(e(P_m, t, 1)+e(P_m, t, 2)+ --- +e(P_m, t, H(m, t))) =E(e_*(P_m, t))$, and constructs encrypted total expenditure record $E(Sum(P_m, t)) = <E(e_*(P_m, t)), U_0^{Rm(t)\cdot X*\cdot\lambda*}, G(P_m, t, 1)^{\lambda*}, S(R_m, G(P_m, t, 1))^{\lambda*}>$ to put in L. Where, H(m, t) is the largest division No. part value generated from $C(P_j, s, v)$.
4. $L_Z, L_{Z-1}, ---, L_1$ repeatedly decrypt each encrypted total expenditure record $E(Sum(P_m, t))$ put by A while shuffling their decryption results, and as a result $E(Sum(P_m, t))$ is decrypted to total expenditure record $Sum(P_m, t) = <e_*(P_m, t) = e(P_m, t, 1)+e(P_m, t, 2)+ --- +e(P_m, t, H(m, t)), U_0^{Rm(t)\cdot X*\cdot\lambda*\cdot\mu*}, G(P_m, t, 1)^{\lambda*\cdot\mu*}, S(R_m, G(P_m, t, 1))^{\lambda*\cdot\mu*}>$. Here mix-servers must shuffle also their decryption results to protect their additive encryption functions which are usually weak against plain text attacks. About the division-ID part value, provided that $\mu(1), \mu(2), ---, \mu(Z)$ are secret integers of $L_1, L_2, ---, L_Z$ and $\mu_* = \mu(1)\cdot\mu(2)$ --- $\mu(Z)$, each L_z transforms (decrypts) $U_0^{Rm(t)\cdot X*\cdot\lambda*\cdot\mu(Z)\cdot\mu(Z-1)---\mu(z+1)}$ calculated by L_{z+1} to $U_0^{Rm(t)\cdot X*\cdot\lambda*\cdot\mu(Z)\cdot\mu(Z-1)---\mu(z+1)\cdot\mu(z)}$, and as a result $U_0^{Rm(t)\cdot X*\cdot\lambda*}$ is transformed to $U_0^{Rm(t)\cdot X*\cdot\lambda*\cdot\mu*}$. In the same way $G(P_m, t, 1)^{\lambda*}$ and $S(R_m, G(P_m, t, 1))^{\lambda*}$ are transformed to $G(P_m, t, 1)^{\lambda*\cdot\mu*}$ and $S(R_m, G(P_m, t, 1))^{\lambda*\cdot\mu*}$.
5. For each issuing record $I(i) = <e(i), d(i), 0, U_0^{Rm(t)}, \{U_0^{Rm(t)}+e(i)\|d(i)\}^{Rm}>$ which does not expire, A constructs new-cash record $N(A, i, 0) = <e(i), U_0^{Rm(t)}>$ and put it in L together with new cash records generated at step 1.
6. By using integers $\lambda(1), \lambda(2), ---,\lambda(Z), X(1), X(2), ---, X(H)$, and $\mu(1), \mu(2), ---, \mu(Z)$, mix-servers $L_1, L_2, ---, L_Z$ calculate $U_0^{Rk(q)\cdot X*\cdot\lambda*\cdot\mu*}$ or $U_0^{Rm(t)\cdot X*\cdot\lambda*\cdot\mu*}$ from $U_0^{Rk(q)}$ or $U_0^{Rm(t)}$ in each $N(P_j, s, v)$ or $N(A, i, 0)$ to transform it to $N_*(P_j, s, v) = <e(P_m, t, h), U_0^{Rk(q)\cdot X*\cdot\lambda*\cdot\mu*}>$ or $N_*(A, i, 0) = <e(i), U_0^{Rm(t)\cdot X*\cdot\lambda*\cdot\mu*}>$.
7. For each total expenditure record $Sum(P_m, t)$, A finds a transformed new-cash record that includes $U_0^{Rm(t)\cdot X*\cdot\lambda*\cdot\mu*}$ as its source-ID part value. Where, because all paying records have different source-ID part values, A can find at most one transformed new-cash record.
8. Provided that $N_*(P_j, s, v) = <e(P_j, s, v), U_0^{Rm(t)\cdot X*\cdot\lambda*\cdot\mu*}>$ is the found transformed new-cash record, A examines whether relation $e_*(P_m, t) \le e(P_j, s, v)$ holds or not, and when the relation does not hold it determines that $Sum(P_m, t)$ is an illegitimate total expenditure record. When no new-cash record that corresponds to $Sum(P_m, t)$ is found, A determines $Sum(P_m, t)$ is an orphan record.

In the above steps, because $U_1, U_2, ---, U_H$ are calculated as $U_1 = U_0^{X(1)}$, $U_2 = U_0^{X(1)\cdot X(2)}$,---, $U_H = U_0^{X(1)\cdot X(2) --- X(H)}$, division-ID part values of all expenditure records that correspond to e-cash generated from e-cash $C(P_j, s, v)$ and source-ID part value $U_0^{Rm(t)}$ of $N(P_j, s, v)$ are transformed to same value $U_0^{Rm(t)\cdot X*\cdot\lambda*\cdot\mu*}$. Therefore A can identify expenditure records generated from $C(P_j, s, v)$ as the ones that are accompanied by division-ID part value $U_0^{Rm(t)\cdot X*\cdot\lambda*\cdot\mu*}$. Also, because linear Mix-net L encrypts $e(P_m, t, 1), e(P_m, t, 2), ---, e(P_m, t, H(m, t))$ in expenditure records by additive encryption functions, $E(e(P_m, t, 1))+E(e(P_m, t, 2))+ --- +E(e(P_m, t, H(m, t)))$ is decrypted to $e_*(P_m, t) = e(P_m, t, 1)+e(P_m, t, 2)+ --- +e(P_m, t, H(m, t))$, i.e. the sum of cash values generated from $C(P_j, s, v)$. As a result, A can determine that cash holder P_m had honestly divided its e-cash $C(P_j, s, v)$ when $e_*(P_m, t)$ does not exceed $e(P_j, s, v)$.

Nevertheless, cash holder P_m can conceal its payments and links between its individual payments from others. Because integers R_m, $R_{m(t)}$ are P_m's secrets, other entities including A cannot know P_m from issuing record $I(i)$ or paying record $P(P_m, t, h)$. About links between $C(P_m, t, 1)$, $C(P_m, t, 2)$, ---, $C(P_m, t, H(m, t))$, to calculate $X(h)$ from $U_{h-1}^{Rm(t)}$ and $U_{h-1}^{Rm(t)\cdot X(h)}$ is a discrete logarithm problem, and entities other than P_m cannot identify sequence $U_0^{Rm(t)}$, $U_1^{Rm(t)}$, $U_2^{Rm(t)}$, ---, $U_H^{Rm(t)}$.

5.2. Dishonest Entity Identification Stage

As in the previous subsection dishonestly handled paying records are detected as inconsistent total

expenditure and new-cash records pairs or orphan total expenditure records, and once dishonesties are detected, *A* identifies liable entities without knowing secrets of honest entities by the procedure below. In the following it is assumed that total expenditure record $\mathrm{Sum}(P_m, t) = \langle e_*(P_m, t), U_0^{Rm(t)\cdot X^*\cdot\lambda^*\cdot\mu^*}, G(P_m, t, 1)^{\lambda^*\cdot\mu^*}, S(R_m, G(P_m, t, 1))^{\lambda^*\cdot\mu^*}\rangle$ that corresponds to cash holder P_m is illegitimate, i.e. $e_*(P_m, t)$ in $\mathrm{Sum}(P_m, t)$ is greater than $e(P_j, s, v)$ in transformed new-cash record $N_*(P_j, s, v) = \langle e(P_j, s, v), U_0^{Rm(t)\cdot X^*\cdot\lambda^*\cdot\mu^*}\rangle$, or $\mathrm{Sum}(P_m, t)$ is an orphan record.

Although additional steps are required to protect secret information of honest entities, conceptually, after detecting illegitimate total expenditure record $\mathrm{Sum}(P_m, t)$, *A* discloses $\mathrm{Sum}(P_m, t)$ and asks all cash holders to calculate used seals of their credentials from record characterizer part value $G(P_m, t, 1)^{\lambda^*\cdot\mu^*}$ in $\mathrm{Sum}(P_m, t)$ while revealing their identities. Then, P_m that calculates $S(R_m, G(P_m, t, 1))^{\lambda^*\cdot\mu^*} = G(P_m, t, 1)^{Rm\cdot\lambda^*\cdot\mu^*}$ is liable, i.e. only P_m that knows integer R_m can calculate $S(R_m, G(P_m, t, 1))^{\lambda^*\cdot\mu^*}$ from $G(P_m, t, 1)^{\lambda^*\cdot\mu^*}$ and P_m must honestly calculate $S(R_m, G(P_m, t, 1))^{\lambda^*\cdot\mu^*}$ from $G(P_m, t, 1)^{\lambda^*\cdot\mu^*}$ as a used seal of its own credential.

However, if record characterizer and payers' signature pair $\{G_1, G_2\}$ in $P(P_m, t, 1)$ is the one forged by P_m or someone else no one calculates $G_2^{\lambda^*\cdot\mu^*}$ from $G_1^{\lambda^*\cdot\mu^*}$. Therefore in this case, *A* finds encrypted expenditure record $E(D(P_m, t, 1))$ that was used to calculate $E(\mathrm{Sum}(P_m, t))$ at step 3 in the dishonest record detection stage, and asks all cash holders to calculate used seals of their credentials from payer's signature $G_2^{\lambda^*}$, and determines P_k is liable when it calculates $G_2^{\lambda^*\cdot Rk}$ that is equal to the payee's signature $G_2^{Rk\cdot\lambda^*}$ in $E(D(P_m, t, 1))$.

Namely, authority *A* examines payee's signature G_2^{Rk} when it accepts paying record $P(P_m, t, h)$, and P_k must honestly calculate it as a used seal of its credential. Also, P_k verifies the consistency of pair $\{G_1, G_2\}$ at a time when P_m paid $C(P_m, t, h)$ to it. In addition, anyone including *A* and mix-servers cannot handle paying records dishonestly after they are disclosed. Then, regardless that entities other than P_k itself had forged $\{G_1, G_2\}$ or not, *A* can determine P_k is liable, i.e. P_k had accepted $P(P_m, t, h)$ that includes inconsistent $\{G_1, G_2\}$. As an exception if it is conspiring with *A*, P_k does not need to calculate G_2^{Rk} honestly by using its credential when it reports $P(P_m, t, h)$. But even in this case *A* cannot impute the liability to honest cash holders, i.e. *A* cannot forge payee's signatures of honest cash holders and must compensate corresponding losses by itself.

In detail, the dishonest entity identification stage proceeds as follow.

1. *A* asks mix-servers L_Z, L_{Z-1}, ---, L_1 to trace (partial) encryption forms of illegitimate total expenditure record $\mathrm{Sum}(P_m, t) = \langle e_*(P_m, t), U_0^{Rm(t)\cdot X^*\cdot\lambda^*\cdot\mu^*}, G(P_m, t, 1)^{\lambda^*\cdot\mu^*}, S(R_m, G(P_m, t, 1))^{\lambda^*\cdot\mu^*}\rangle$ calculated at step 4 in Sec. 5.1 back to $E(\mathrm{Sum}(P_m, t)) = \langle E(e_*(P_m, t)), U_0^{Rm(t)\cdot X^*\cdot\lambda^*}, G(P_m, t, 1)^{\lambda^*}, S(R_m, G(P_m, t, 1))^{\lambda^*}\rangle$, and finds encrypted expenditure records $E(D(P_m, t, 1)) = \langle E(e(P_m, t, 1)), 1, U_0^{Rm(t)\cdot X^*\cdot\lambda^*}, G(P_m, t, 1)^{\lambda^*}, S(R_m, G(P_m, t, 1))^{\lambda^*}, S(R_k, S(R_m, G(P_m, t, 1)))^{\lambda^*}\rangle$, $E(D(P_m, t, 2))$, ---, $E(D(P_m, t, H(m, t)))$ that constitute $E(\mathrm{Sum}(P_m, t))$.

2. *A* discloses illegitimate total expenditure record $\mathrm{Sum}(P_m, t)$ with individual encrypted expenditure records $E(D(P_m, t, 1))$, $E(D(P_m, t, 2))$, ---, $E(D(P_m, t, H(m, t)))$ publicly.

3. For each illegitimate $\mathrm{Sum}(P_m, t) = \langle e_*(P_m, t), U_0^{Rm(t)\cdot X^*\cdot\lambda^*\cdot\mu^*}, G(P_m, t, 1)^{\lambda^*\cdot\mu^*}, S(R_m, G(P_m, t, 1))^{\lambda^*\cdot\mu^*}\rangle$, each P_r calculates $G(P_m, t, 1)^{\lambda^*\cdot\mu^*\cdot Rr}$ from $G(P_m, t, 1)^{\lambda^*\cdot\mu^*}$ by using its credential $T(A, T_{Pr}, R_r)$.

4. When P_m calculates $G(P_m, t, 1)^{\lambda^*\cdot\mu^*\cdot Rm}$ that coincides with $S(R_m, G(P_m, t, 1))^{\lambda^*\cdot\mu^*}$, and P_m admits $e_*(P_m, t)$ in $\mathrm{Sum}(P_m, t)$ is its correct total expenditure or expenditure records included in orphan record $\mathrm{Sum}(P_m, t)$ had already expired, P_m pays cash of value $\{e_*(P_m, t) - e(P_j, s, v)\}$ to *A* possibly with arrears but without revealing its identity. In a case where $\mathrm{Sum}(P_m, t)$ is an orphan record P_m pays $e_*(P_m, t)$.

5. On the other hand when P_m does not believe that $e_*(P_m, t)$ is its correct total expenditure or expenditure records included in $\mathrm{Sum}(P_m, t)$ had expired, it examines payer's signatures in disclosed individual encrypted expenditure records. In detail, for each encrypted record $E(D(P_m, t, h))$, P_m extracts pair $\{G(P_m, t, h)^{\lambda^*}, S(R_m, G(P_m, t, h))^{\lambda^*}\}$ to examine whether relation $G(P_m, t, h)^{\lambda^*\cdot Rm} = S(R_m, G(P_m, t, h))^{\lambda^*}$ holds or not. After that when relation $G(P_m, t, h)^{\lambda^*\cdot Rm} = S(R_m, G(P_m, t, h))^{\lambda^*}$ does not hold, P_m without revealing its identity claims that $E(D(P_m, t, h))$ is not its record while convincing *A* that $S(R_m, G(P_m, t, h))^{\lambda^*}$ is not calculated by its credential (*A* receives this claim at step 9).

6. In a case where relation $G(P_m, t, h)^{\lambda^*\cdot Rm} = S(R_m, G(P_m, t, h))^{\lambda^*}$ holds but P_m does not believe that $E(D(P_m, t, h))$ had expired, P_m picks $P(P_j, s, v)$ from which it had generated $P(P_m, t, h)$, and claims that the expiration time in $P(P_j, s, v)$ is incorrect without revealing its identity.

7. When P_m claims that the expiration time in $P(P_j, s, v) = \langle e(P_j, s, v), d(P_j, s, v), v, U_v^{Rj(s)}, U_0^{Rm(t)}, S(R_j, G(P_j, s, v))\rangle$ is incorrect, *A* requests all cash holders to calculate used seal of $G(P_j, s, v)$ while revealing their identities.

8. *A* determines cash holder P_j that calculates payer's signature $S(R_j, G(P_j, s, v))$ from $G(P_j, s, v)$ had included a wrong value as the expiration time in $P(P_j, s, v)$ to give to P_m, and P_j pays $e(P_j, s, v)$ to *A* with penalty fine. But *A* determines P_m had accepted incorrect $P(P_j, s, v)$ intentionally when no one calculates $S(R_j, G(P_j, s, v))$, and to identify P_m proceeds to step 11.

9. When anonymous P_m claims that $E(D(P_m, t, h))$ is not its expenditure record, *A* requests all cash holders to calculate used seals of their credentials from payer's signature $S(R_m, G(P_m, t, h))^{\lambda^*}$ while revealing their identities.

10. *A* determines P_k that calculates payee's signature $S(R_k, S(R_m, G(P_m, t, h)))^{\lambda^*} = S(R_m, G(P_m, t, h))^{\lambda^*\cdot Rk}$ from $S(R_m, G(P_m, t, h))^{\lambda^*}$ is liable and forces P_k to pay cash of value $e(P_m, t, h)\}$ with penalty fine.

11. If no one paid for illegitimate expenditure records corresponding to $\mathrm{Sum}(P_m, t)$ or claimed they were incorrect, or no one calculates payer's signature $S(R_j, G(P_j, s, v))$ at step 8, firstly, *A* requests all cash holders to calculate used seals of their credentials from record characterizer value $G(P_m, t, 1)^{\lambda^*}$ while revealing their identities. After that, *A* determines P_m that calculated $S(R_m, G(P_m, t, 1))^{\lambda^*}$ is liable, and forces P_m to pay cash of value $\{e_*(P_m, t) - e(P_j, s, v)\}$ with penalty fine. When $\mathrm{Sum}(P_m, t)$ is an orphan record P_m pays $e_*(P_m, t)$.

12. If no one calculates $S(R_m, G(P_m, t, 1))^{\lambda^*}$ in the previous step, *A* picks encrypted expenditure record $E(D(P_m. t, 1)) = \langle E(e(P_m, t, 1)), 1, U_0^{Rm(t)\cdot X^*\cdot\lambda^*}, G(P_m, t, 1)^{\lambda^*}, S(R_m, G(P_m, t, 1))^{\lambda^*}, S(R_k, S(R_m, G(P_m, t, 1)))^{\lambda^*}\rangle$ that

constitutes Sum(P_m, t), and requests all cash holders to calculate used seals of their credentials from payer's signature $S(R_m, G(P_m, t, 1))^{\lambda^*}$ while revealing their identities.

13. *A* forces P_k that calculates payee's signature $S(R_k, S(R_m, G(P_m, t, 1)))^{\lambda^*} = S(R_m, G(P_m, t, 1))^{Rk \cdot \lambda^*}$ from $S(R_m, G(P_m, t, 1))^{\lambda^*}$ to pay $\{e_*(P_m, t) - e(P_j, s, v)\}$ with penalty fine, but in a case where Sum(P_m, t) is an orphan record, P_k pays $e_*(P_m, t)$.

If cash holders honestly use their e-cash and handle their paying records, in other words, if cash values, expiration times, division-ID and source-ID parts values and payer's signatures in individual paying records are honestly generated, reported and disclosed, apparently the above procedure successfully identifies dishonest entities while maintaining sensitive data as their secrets provided that authority *A* is honest (about payee's signatures, cash holders must calculate them honestly as discussed before). When cash holder P_m generated new e-cash from its e-cash $C(P_j, s, v)$ excessively by mistake, firstly at step 3, P_m knows that disclosed illegitimate total expenditure record Sum(P_m, t) corresponds to it, i.e. payer's signature $S(R_m, G(P_m, t, 1))^{\lambda^* \cdot \mu^*}$ in Sum(P_m, t) is calculated from record characterizer $G(P_m, t, 1)^{\lambda^* \cdot \mu^*}$ only by using its credential, and at step 4, P_m pays the difference between its actual expenditure and the original cash value $e_*(P_m, t) - e(P_j, s, v)$ to *A*. Here, P_m is not requested to reveal its identity when it calculates $S(R_m, G(P_m, t, 1))^{\lambda^* \cdot \mu^*}$, and still can conceal the correspondence between it and its e-cash regardless that its mistake is intentional or not.

The procedure also identifies dishonest entities even when paying records are dishonestly generated and/or handled. Namely, if payee P_k is honest, because P_k examines the division-ID part value and the payer's signature in paying record $P(P_m, t, h)$ when payer P_m pays $C(P_m, t, h)$, P_m must generate the payer's signature honestly while using its credential. Therefore, *A* can determine that P_m is dishonest by examining payer's signatures at step 11 even when $P(P_m, t, h)$ was forged by P_m. On the other hand, in a case where P_k modified $P(P_m, t, h)$ after having received it from P_m, P_k cannot include consistent payer's signature of P_m in $P(P_m, t, h)$ because all paying records must have different source-ID part values despite P_k does not know integer R_m. Then P_m claims that $P(P_m, t, h)$ is incorrect at step 5, and because P_k must calculate its payee's signature honestly when it reports $P(P_m, t, h)$, *A* can determine that P_k is dishonest at steps 9 and 10, i.e. P_k had received an inconsistent paying record regardless that other entities are conspiring with it or not. In the same way, *A* can identify also other entities when they forge $P(P_m, t, h)$, i.e. anyone other than P_m cannot forge paying record $P(P_m, t, h)$ so that it becomes consistent with P_m's credential.

About a case where P_m does not report paying record $P(P_j, s, v)$ to *A*, each $P(P_m, t, h)$ becomes an orphan record, and if P_k is honest, P_m that calculates payer's signature $S(R_m, G(P_m, t, 1))^{\lambda^*}$ is determined as liable, i.e. only P_m can calculate $S(R_m, G(P_m, t, 1))^{\lambda^*}$ at step 11 as same as in the above. On the other hand when P_k is dishonest, P_m does not calculate $S(R_m, G(P_m, t, 1))^{\lambda^*}$ because P_k cannot generate P_m's payer's signature consistently, and P_k is identified as an entity that had accepted inconsistent payer's signature at step 12. In the same way, P_j that had paid expired e-cash $C(P_j, s, v)$ to P_m while modifying the expiration time is identified at step 8. Lastly in a case where authority *A* is dishonest, even *A* cannot forge payer's or payee's signatures of honest cash holders consistently. Then, it cannot identify any liable entity and must compensate the corresponding losses by itself.

About the anonymity of cash holders, cash holders in Step 5 do not reveal their identities. Although each cash holder calculates used seals of its credential from $G(P_m, t, h)^{\lambda^*}$ or $S(R_m, G(P_m, t, h))^{\lambda^*}$ while revealing its identity at Steps 7, 9, 11 and 12, it did not calculate $G(P_m, t, h)^{\lambda^* \cdot Rj}$, $G(P_m, t, h)^{\lambda^* \cdot Rm}$, $G(P_m, t, h)^{\lambda^* \cdot Rk}$ or $S(R_m, G(P_m, t, h))^{\lambda^* \cdot Rk}$ before if it is honest. Therefore no one including *A* can identify payments of honest cash holders.

As another type of dishonesty although protection of this dishonesty is out of the scope of the proposed scheme, cash holders may disappear during interactions between them, e.g. a payee can disappear after it receives e-cash despite a payer does not complete its purchase yet. But these threats also can be removed easily by the secure object exchange scheme [11].

6. Conclusion

As discussed in previous sections the proposed scheme successfully satisfy anonymity, divisibility and transferability of e-cash in offline environments. Namely, honest cash holders can conceal correspondences between them and their e-cash and links between e-cash they had spent. Nevertheless, cash issuing authority *A* can identify liable entities when e-cash were illegitimately generated or used.

In addition, P_m can generate new e-cash $C(P_m, t, 1)$, $C(P_m, t, 2)$, ---, $C(P_m, t, H(m, t))$ of any values from its e-cash $C(P_j, s, v)$ that it had received from other cash holder P_j provided that the total cash value of them does not exceed the cash value of $C(P_j, s, v)$ as above, i.e. $C(P_j, s, v)$ is divisible and transferable. But different from in other schemes, volume of information included in each e-cash does not increase even when it is transferred multiple times. Cash holders do not need to maintain receipts for transferring their e-cash either. In addition, an honest e-cash holder can conceal the correspondence between it and its e-cash even when the e-cash is a dishonestly transferred one. About divisibility, authority *A* does not need to define the minimum cash value unit in advance, and as a result, costs for handling e-cash can be decreased (the costs do not increase with the minimum cash value unit).

Drawbacks of the scheme are, firstly when cash holder P_k receives e-cash $C(P_m, t, h)$ from P_m, it must report paying record $P(P_m, t, h)$ to authority *A* through online communication channels, and secondly to generate new e-cash from existing e-cash each P_m must obtain numbers of cash-IDs in advance. But P_k is not required to report $P(P_m, t, h)$ immediately, i.e. it can report it together with other paying records at its convenient time. Therefore, inconvenience caused by the reporting can be mitigated. Also, although cash holder P_m obtains all cash-IDs in the registration phase in Sec.4, it can obtain convenient number of cash-IDs anytime, e.g. at a time when it reports paying records.

As other drawbacks, each cash holder is required to examine individual illegitimate records even if it is honest,

and different from real cash, cash holders must use their e-cash before they expire. About these drawbacks, numbers of illegitimate records can be decreased by high penalty fines. Also, although the proposed scheme defines expiration time of e-cash $C(P_m, t, h)$ as that of $C(P_j, s, v)$ from which $C(P_m, t, h)$ was generated, it is possible to make $C(P_m, t, h)$ valid during a fixed duration after $C(P_m, t, h)$ was generated.

References

[1] T. Okamoto, "An efficient divisible electronic cash scheme," *Crypto'95*, 438-451. 1995.

[2] J. Camenisch, S. Hohenberger and A. Lysyanskaya, "Compact e-cash," *Eurocrypt'05*, 302-321. 2005.

[3] S. Canard and A. Gouget, "Divisible e-cash systems can be truly anonymous," *Eurocrypt'07*, 482-497. 2007.

[4] S. Canard, A. Gouget and J. Traore, "Improvement of efficiency in (unconditional) anonymous transferable e-cash," *Financial Cryptography 2008*, 202-214. 2008.

[5] G. Fuchsbauer, D. Pointcheval and D. Vergnaud, "Transferable constant-size fair e-cash," *Proceedings of the 8th International Conference on Cryptology and Network Security*, 226-247. 2009.

[6] S. Canard and A. Gouget, "Multiple denominations in e-cash with compact transaction data," *Financial Cryptography 2010*, 82-97. 2010.

[7] M. Izabachene and B, Libert, "Divisible e-cash in the standard model," *Pairing'12*, 314-332. 2012.

[8] S. Tamura, "Anonymous Security Systems and Applications: Requirements and Solutions," *Information Science Reference*, 2012.

[9] S. Tamura and S. Taniguchi, "Enhancement of anonymous tag based credentials," *Information Security and Computer Fraud*, Vol. 2, No. 1, 10-20, 2014.

[10] S. Tamura and S. Taniguchi, "Linear mix-net and a scheme for collecting data from anonymous data holders," *Information Security and Computer Fraud*, Vol. 2, No. 3, 39-47, 2014.

[11] S. Tamura, "Elements of Schemes for Preserving Privacies in e-society Systems," *Lambert Academic Publishing*, 2015.