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


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 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-cash 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_s, v) that was paid by other cash holder P_s. 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

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.
identify liable entities as same as other existing schemes.

Therefore, to make used seals unique to credentials, actually $T(A, T_p, R)$ is implemented as a set of values $\{S(d_1\|d_2, T_P\|K_\alpha G_w^R \mod B), S(d_1\|d_2, T_P\|K_\alpha G_w^R \mod B_2)\}$ by using different integers $B_1$ and $B_2$. Namely, relation $\text{mod } B_1 \neq \text{mod } B_2$ does not hold even when relation $\text{mod } B_1 = \text{mod } B_2$ 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 $T(A, T_p, R)$, d) **Unlinkability** even if $P$ shows credential $T(A, T_p, R)$ n-times in forms $T(A, T_p, R)\|\|1$, $T(A, T_p, R)\|\|2$, $T(A, T_p, R)\|\|n$ 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 $T(A, T_p, R)$ 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. 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 such that $S(d_1\|d_2, x) = x^{d_1 \mod B_1} \| x^{d_2 \mod B_2}$, signature pair $T(A, T_p, R) = S(d_1\|d_2, T_P \| K_\alpha G_w^R \mod B)$ 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 $T_P = k^{q_1 \mod B_1}$, $T_P^* = k^{q_1^* \mod B_1}$, $T_P = k^{q_2 \mod B_2}$, $T_P^* = k^{q_2^* \mod B_2}$, $g = T_P^{q_3 \mod B}$, $g = T_P^{q_3^* \mod B}$ 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 $K_w = k^{q_1 \mod B}$ and $G_w = g^{q_2 \mod B}$ 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 $\mod B$ 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 $T(A, T_p, R)^W = S(d_1\|d_2, T_P^{R^1}K_wG_w^R \mod B)^W$ 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 $\{S(d_1\|d_2, T_P\|K_wG_w^R \mod B), S(d_1\|d_2, T_P\|K_wG_w^R \mod B_2)\}$ by using different integers $B_1$ and $B_2$.

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 $T(A, T_p, R)^W$, $T(A, T_p, R)^W_1$, $T(A, T_p, R)^W_2$, $T(A, T_p, R)^W_n$ 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 $T(A, T_p, R)$ 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 $L_1$, $L_2$, $\ldots$, $L_2$ and enables authority $A$ to calculate the sum of attribute values $D_1(1)$, $D_2(2)$, $\ldots$, $D_P(H_P)$ 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 $D_p(1)$, $D_p(2)$, ..., $D_p(H)$ [10, 11].

Conceptually, $P$ generates triplet $[h, D_p(h), U(h)]^k$ that includes its $h$-th attribute value $D_p(h)$ for each $h$, puts triplets $[1, D_p(1), U(1)]^k$, $[2, D_p(2), U(2)]^k$, ..., $[H_p, D_p(H_p), U(H_p)]^k$ in $L$ separately without revealing its identity, and mix-servers $L_1$, $L_2$, ..., $L_z$ repeatedly encrypt each $[h, D_p(h), U(h)]^k$ to $[h, E(D_p(h)), (U(h))^z]$ by their secret encryption keys while shuffling their encryption results. After that, authority $A$ gathers encrypted triplets $[1, E(D_p(1)), U(1)]^k$, $[2, E(D_p(2)), U(2)]^k$, ..., $[H_p, E(D_p(H_p)), U(H_p)]^k$ that correspond to $P$ and calculates sum $E(D_p(*)^+ = E(D_p(1)) + E(D_p(2))^+ + ... + E(D_p(H_p))$, constructs pair $[E(D_p(*)^+, U(*)^k)]$, and finally $L_1, L_2, ..., L_z$ repeatedly decrypt each pair $[E(D_p(*)^+, U(*)^k)]$ to $(D_p(*)) = D_p(1) + D_p(2) + ... + D_p(H_p)$, $(U(*))^z$ by their secret decryption keys while shuffling their decryption results.

In detail, $P$ convinces $L$ of its eligibility by showing credential $T(A, T_p, R)^{X(0, h)}$ while generating secret integer $W(P)$ to put its $h$-th attribute value $D_p(h)$, and calculates used seal $U(h)^k$ of $T(A, T_p, R)$ from integer $U(h)$ defined by $L_1$, $L_2$, ..., $L_z$. About encryptions and decryptions of $U(h)^k$ in $[h, D_p(h), U(h)]^k$ and $(U(h))^k$ in $E(D_p(*)^+$, $U(*)^k$ to $U(*)^z = U(h)^{X(1, z)}$, $U(*)^z = U(*)^{X*(V(1), V(2), ..., V(z))}$ by using their secret integers $V(1)$, $V(2)$, $V(2)$, ..., $V(z)$.

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 $L$ shuffles its encryption and decryption results). Nevertheless, $A$ can identify triplets $[1, E(D_p(1)), U(1)]^k$, ..., $[H_p, E(D_p(H_p)), U(H_p)]^k$ that correspond to same data holder $P$, and $P$ can know the sum of its attribute values $(D_p(*)^+, U(*)^k)$. 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 $U(1) = U(0)X(1, 1)$, $U(2) = U(1)X(2, 2)$, ..., $U(H_p) = U(Z-1, Z-1)X(Z, Z)$.

To calculate $U(h)$ for each $h$, $L_1, L_2, ..., L_z$ calculate $(U(h))^{X(1, h)} = U(0)X(1, h)$, $(U(h))^{X(2, h)} = U(1)X(2, h)$, ..., $(U(h))^{X(Z, h)}$ for each $h$, and $A$ identifies triplets $[1, E(D_p(1)), U(1)]^k$, $[2, E(D_p(2)), U(2)]^k$, ..., $[H_p, E(D_p(H_p)), U(H_p)]^k$ based on $(U(*)^k)$ calculated from $U(*)^z$. On the other hand, $P$ calculates source-ID and its 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.

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 $L_j$ maintains its secret integers $X(z, 1)$, $X(z, 2)$, ..., $X(z, H)$, and for each $h (0 < h < H)$, $L_1, L_2, ..., L_z$ jointly calculates integer $U_h = U_h^X(h) = U_h^X(h) = U_h^X(h) + U_h^X(h)$ 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](#)
1. \( P_m \) shows \( A \) its identity.

2. If \( P_m \) is eligible, \( A \) issues credential \( T(A, T_{Pm}, R_m) \) and cash-IDs \( T^*(A, T_{Pm}(1), R_m(1)), T^*(A, T_{Pm}(2), R_m(2)), \ldots, T^*(A, T_{Pm}(H), R_m(H)) \) to \( P_m \).

3. \( P_m \) calculates initial used seal \( U_{R_m} \) from publicly disclosed constant integer \( U_\_ \) by using \( T(A, T_{Pm}, R_m) \).

4. \( A \) registers \( P_m \) as an authorized member with initial used seal \( U_{R_m} \).

Here, \( A \) uses initial used seal \( U_{R_m} \) to disable \( P_m \) from using 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 \( \lambda \) without showing its identity, \( A \) asks \( P_m \) to calculate a used seal also from integer \( U_\_. \) Namely, if \( P_m \) calculated \( \lambda^{R_m} \) by using a credential of other entity, \( P_m \) calculates \( U_\_^{R_m} \) from \( U_\_ \) that is different from initial used seal \( U_{R_m} \). 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 \( R_m = R_m(0), R_m(1), R_m(2), \ldots, R_m(H) \) in the credential and cash-IDs are generated through the cooperation among \( L_1, L_2, \ldots, L_Z \). Namely, provided that \( \Omega \) is a publicly known integer, for each \( h \) \((=0, 1, 2, \ldots, H)\), each mix-server \( L_z \) generates secret integer \( R_m(z, h) \) and calculates \( \Omega^{R_m(z-1)*} h \Omega^{R_m(z)*} \) based on \( \Omega^{R_m(z-1)*} h \Omega^{R_m(z)*} \) disclosed by \( L_{z-1} \), and when finally disclosed \( \Omega^{R_m(z)*} \) 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 \( R_m(h) = R_m(1, h)R_m(2, h) \ldots R_m(H, h) \). When \( \Omega^{R_m(z)*} \) had appeared already, \( L_1, L_2, \ldots, L_Z \) 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 \( T^*(A, T_{Pm}(t), R_m(t)) \) and \( T^*(A, T_{Pm}(t), R_m(t)) \).

2. \( A \) defines cash value \( e(i) \) and expiration time \( d(i) \).

3. \( P_m \) calculates source-ID part value \( 0 \) represents that \( P_m \) has no t-th cash-ID. \( C(A, i) \) or \( C(A, i) \) have the same expiration time. Also, \( C(A, i) \) or \( C(A, i) \) have the same service period have same expiration time. Also, \( C(A, i) \) or \( C(A, i) \) have the same service period have same expiration time. Also, \( C(A, i) \) or \( C(A, i) \) have the same service period have same expiration time.

Here, \( P_m \) calculates \( U_0^{R_m(t)} \) and \( \{U_0^{R_m(t)} , e(i) , d(i) \}_R \) as used seals of \( T(A, T_{Pm}(t), R_m(t)) \) and \( T(A, T_{Pm}(t), R_m(t)) \) for real cash, \( P_m \) can convince any entity that \( P_m \) calculates \( U_0^{R_m(t)} \) and \( \{U_0^{R_m(t)} , e(i) , d(i) \}_R \) as used seals of \( T(A, T_{Pm}(t), R_m(t)) \) and \( T(A, T_{Pm}(t), R_m(t)) \).

4. \( P_m \) pays cash value \( e(i) \) through its anonymous credit card, or it simply pays real cash of value \( e(i) \) to \( A \).

5. \( P_m \) calculates source-ID part value \( 0 \) represents that \( P_m \) has no t-th cash-ID. \( C(A, i) \) or \( C(A, i) \) have the same expiration time. Also, \( C(A, i) \) or \( C(A, i) \) have the same service period have same expiration time. Also, \( C(A, i) \) or \( C(A, i) \) have the same service period have 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 \( P_j \) 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.
although $P_m$ 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 $T(A, T_{fm}(R_m)))$ in forms $T(A, T_{pm}(R_m))^{W_{m}^*(t, h)}$ and $T(A, T_{pm}(R_m))^{W_{m}(t, h)}$, and convinces $P_k$ of its eligibility and the ownership of $T(A, T_{pm}(R_m))$.

2. $P_k$ generates secret integers $V_{m}(q)$ and $V_{m}^*(q)$, chooses $q$-th cash-ID $T(A, T_{km}(R_k))$, shows credential $T(A, T_{rm}(R_m))$ and $T(A, T_{km}(R_k))$ in forms $(T(A, T_{km}(R_k))^{V_{m}^*(q)})$ and $(T(A, T_{km}(R_k))^{V_{m}(q)})$, and convinces $P_m$ of its eligibility and the ownership of $T(A, T_{km}(R_k))$.

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 $U_0$ by using cash-ID $T(A, T_{pm}(R_m)(t))$, 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 $T(A, T_{pm}(R_m), R_k)$, and calculates used seal of its $q$-th cash-ID from $T(A, T_{pm}(R_m), R_k)$, and informs $P_m$ of its eligibility and the ownership of $T(A, T_{pm}(R_m), R_k)$.

5. $P_m$ defines $e(P_m, t, h)$ and expiration time $d(P_m, t, h)$, calculates division-ID part value $U_{h}^{Rm(t)}$ as a used seal of $T(A, T_{pm}(R_m), R_k)$, and informs $P_k$ of used seal of $T(A, T_{pm}(R_m), R_k)$, and informs $P_m$ of quadruplet $(e(P_m, t, h), d(P_m, t, h), h, U_{h}^{Rm(t)})$.

In the above, $P_m$ and $P_k$ show their credentials and cash-IDs while modifying them by their secret integers $W_{m}(t, h)$, $W_{m}^*(t, h)$, $V_{m}(q)$ and $V_{m}^*(q)$, 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 $C(A, t)$ or $C(P_i, s, v)$. Nevertheless, $P_m$ can perform correct calculations of division-ID part value $U_{h}^{Rm(t)}$ and payer’s signature $S(R_m, G(P_m, t, h))$ provided that $P_m$ calculates them as used seals of its cash-ID and credential. In the same way, $P_k$ can perform $P_m$’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 $t(i)$ and/or paying record $P(P_i, s, v)$, 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 $C(A, t)$ or $C(P_i, s, v)$ without knowing $P_m$, $C(A, t)$ or $C(P_i, s, v)$. $A$ can also identify dishonesty by exploiting payer’s signatures in paying records together with payer’s signatures that are calculated as in the reporting phase.

About dishonesty 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 $P_i$ or $P_m$, e.g., $P_m$ may calculate the division-ID part value while using a cash-ID different from $T(A, T_{pm}(t) R_m(t))$, 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 $P_i$ that received its $q$-th e-cash $C(P_m, t, h)$ from other cash holder $P_m$ as paying record $P(P_m, t, h) = (e(P_m, t, h), d(P_m, t, h), h, U_{0}^{Rm(t)}, U_{0}^{Rk(q)}, S(R_m, G(P_m, t, h)))$, 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 $T(A, T_{pk}(R_k), R_q)$ and $T(A, T_{pk}(R_k), R_q)^{W_{k}^*(q, 0)}$, and convinces $A$ 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 $S(R_m, G(P_m, t, h)) = S(R_m, G(P_m, t, h))^{R_k}$ 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 $U_0$, i.e., $U_{0}^{Rk(q)}$ 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 $T(A, T_{pk}(q), R_q)$ 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 $U_{0}^{Rk(q)}$ 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 phasing 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 $P_m$. (only $P_m$ can generate consistent signature of $P_m$ on information that includes newly appearing source-ID part 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 \( P(R_m, t) \) from e-cash \( C(P_s, s, v) \) that \( P_p \) 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 \( A \).

1. \( P_m \) generates e-cash from \( C(P_j, s, v) \) more than the cash value of \( C(P_p, s, v) \).

2. \( P_m \) generates paying record \( P(R_m, t, h) \) to give to \( P_k \) while forging the expiration time or calculating a division-ID part value by a cash-ID different from \( T(A, T_m(t), R_m(t)) \) (but \( P_m \) must use its cash-ID if \( P_p \) is honest, because \( P_p \) examines the division-ID part value at step 3 in the paying phase).

3. \( P_m \) does not report paying record \( P(P_p, s, v) \) to \( A \).

4. \( P_p \) reports paying record \( P(R_m, t, h) \) 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 \( P(R_m, t, h) \) multiple times while changing source-ID part values (source-ID part values must be unique to individual e-cash).

5. Other cash holder \( P_j \) reports \( P(R_m, t, h) \) to \( A \) while changing the source-ID part value in it. Where, \( P_j \) can obtain \( P(R_m, t, h) \) 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 \( P_j \).

6. Authority \( A \) arbitrarily forges paying records. Namely, because \( P_m \) or \( P_p \) 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 \( P(R_m, t, h) \) while using a cash-ID different from \( T(A, T_m(t), R_m(t)) \) used to calculate the source-ID part value of \( P(R_p, s, v) \), or may not report \( P(R_p, s, v) \) to \( A \). Also, \( P_j \) can report \( P(R_m, t, h) \) 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 \( P(R_m, t, h) \), 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 \( P_j \)’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 \( U^R(t)·X(h+1)·X(h+2) --- X(H) = U^R(t)·X(h+1)·X(h+2) --- X(H) = U^R(t)·X(h+1)\cdot X(h+2) --- X(H) = U^R(t)·X(h+1)\cdot X(h+2) --- X(H) \), \( A \) can determine paying record \( P(R_m, t, h) \) accompanied by division-ID value \( U^R(t) \) was generated from \( P(R_j, s, v) \) when its source-ID part value \( U^R(0) \) is converted to \( U^R(0) \). Therefore, if \( A \) gathers \( P(R_m, t, 1) \), \( P(R_m, t, 2) \), \( P(R_m, t, H(m, t)) \), 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 \( P(R_p, s, v) \) to \( A \) or \( P(R_j, s, v) \) had expired already, \( A \) cannot use relation \( U^R(t)·X(h+1)\cdot X(h+2) --- X(H) = U^R(t)·X(h+1)\cdot X(h+2) --- X(H) = U^R(t)·X(h+1)\cdot X(h+2) --- X(H) = U^R(t)·X(h+1)\cdot X(h+2) --- X(H) \) 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](image-url) 

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

To implement the above strategies, authority \( A \) constructs expenditure record \( D(P_m, t, h) = <e(P_m, t, h), h, U^R(t), G(P_m(t, h), S(R_m, G(P_m(t, h))), S(R_m, S(R_m, G(P_m(t, h)))), P_m(t, h), h, U^R(t), U^R(t), S(R_m, G(P_m(t, h)))> \) and new-cash record \( N(P_m, t, h) = <e(P_m, t, h), d(P_m(t, h), h, U^R(t), U^R(t), S(R_m, G(P_m(t, h)))) \) 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 \(U_h R_{\text{Mix}}(t) N_i R(t)\) \(e(P_m, t, h)\) \(d(P_m t, h)\) and \(G(P_m t, h) R_{\text{Mix}}(t, h)\) respectively, and payee’s signature \(S(R_m, G(P_m t, h))\) \(= G(P_m t, h) R_{\text{Mix}}(t, h)\) is calculated when Pm reports \(P_m t, h\) 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 source-ID pairs and orphan paying records without.

\[G(P_m t, h) \times R_{\text{Mix}}(t) N_i R(t) \times d(P_m t, h)\]

\(= G(P_m t, h) R_{\text{Mix}}(t, h)\) is calculated when \(P_k\) reports respectively, and payee's signature \(S(R_m, G(P_m t, h))\) \(= G(P_m t, h) R_{\text{Mix}^R}(t, h)\) is calculated when \(P_k\) reports to \(A\). In addition to the above records, total expenditure records are defined as shown in Figure 4 (c).

5. For each issuing record \(i(0) = \langle e(i), d(i), 0, U_h R_{\text{Mix}(t)}\rangle\) \(\{0\}_{R_{\text{Mix}(t)}}\) \(\| d(i) _{R_{\text{Mix}(t)}}\rangle\) which does not exist, A constructs new-cash record \(N(A, i, 0) = \langle e(i), U_h R_{\text{Mix}}(t)\rangle\) and put it in \(L\) together with new cash records generated at step 1.

6. By using integers \(λ(1), λ(2), ---, λ(Z), X(1), X(2), ---, X(H), \) and \(μ(1), μ(2), ---, μ(Z), \) mix-servers \(L_1, L_2, ---, L_2\) calculate \(U_h R_{\text{Mix}(t)}(X^h\times X^h\times μ(Z), X(1), X(2), ---, X(H))\) \(U_h R_{\text{Mix}(t)}(X^h\times X^h\times μ(Z), X(1), X(2), ---, X(H))\)

7. For each total expenditure record \(\text{Sum}(P_m t)\) A finds a transformed new-cash record that includes \(U_h R_{\text{Mix}(t)}(X^h\times X^h\times μ(Z), X(1), X(2), ---, X(H))\) 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 \(N(P_j, s, v) = \langle e(P_j, s, v)\rangle\), \(U_h R_{\text{Mix}(t)}(X^h\times X^h\times μ(Z), X(1), X(2), ---, X(H))\) is the found transformed new-cash record. A examines whether relation \(e(P_m t) \leq e(P_j, s, v)\) holds or not, and when the relation does not hold it determines that \(\text{Sum}(P_m t)\) is an illegitimate total expenditure record. When new-cash record that corresponds to \(\text{Sum}(P_m t)\) is found, A determines \(\text{Sum}(P_m t)\) is an orphan record.

In the above steps, because \(U_1, U_2, ---, U_i\) are calculated as \(U_i = U_{h R_{\text{Mix}(t)}}(X(1) |X(2) = \cdots = X(1)|X(2) = \cdots = X(H))\), division-ID part values of all expenditure records that correspond to e-cash generated from e-cash \(C(P, s, v)\) and source-ID part value \(U_h R_{\text{Mix}(t)}(X^h\times X^h\times μ(Z), X(1), X(2), ---, X(H))\) of \(N(P_j, s, v)\) are transformed to same value \(U_h R_{\text{Mix}(t)}(X^h\times X^h\times μ(Z), X(1), X(2), ---, X(H))\). Therefore A can identify expenditure records generated from \(C(P, s, v)\) as the ones that are accompanied by division-ID part value \(U_h R_{\text{Mix}(t)}(X^h\times X^h\times μ(Z), X(1), X(2), ---, X(H))\).

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, \(e(P_m t, t)\) is decrypted to \(e(P_m t, t)\) \(= e(P_m t, 1) + e(P_m t, 2) + \cdots + e(P_m t, H(m, t))\). As a result, A can determine that cash holder \(P_m\) had honestly divided its e-cash \(C(P, s, v)\) when \(e(P_m t, t)\) does not exceed \(e(P_m t, t)\).

Nevertheless, cash holder \(P_m\) can conceal its payments and links between its individual payments from others. Because integers \(R_{\text{Mix}(t)}\) \(R_{\text{Mix}(t)}\) are \(P_m\)’s secrets, other entities including \(A\) cannot know \(P_m\) from issuing record \(i(t)\) or paying record \(P(P_m t, h)\). About links between \(C(P_m t, t), C(P_m t, 2), ---, C(P_m t, H(m, t))\), to calculate \(X(h)\) from \(U_{h R_{\text{Mix}(t)}}(X^h\times X^h\times μ(Z), X(1), X(2), ---, X(H))\) \(U_{h R_{\text{Mix}(t)}}(X^h\times X^h\times μ(Z), X(1), X(2), ---, X(H))\)

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 Sum(Pm, t) = <e*(Pm, t), U0(Rm(t))X > is calculated at step 4 in Sec. 5.1, calculated new-cash record N(P(s), v) = <e(P(s), s, v), U0(R(s)) >, and Sum(Pm, t) is an orphan record.

Although additional steps are required to protect secret information of honest entities, conceptually, after detecting illegitimate total expenditure record Sum(Pm, t), A discloses Sum(Pm, t) and asks all cash holders to calculate used seals of their credentials from record characterizer part value G(Pm, t, 1)λ*μ* from G(Pm, t, 1) to reveal their identities. Then, Pm that calculates G(Rm(t), G(Pm, t, 1)) is liable, i.e. only Pm that knows integer Rm can calculate S(Rm(t), G(Pm, t, 1)) from G(Pm, t, 1)λ*μ* and Pm must honestly calculate G(Rm(t), G(Pm, t, 1))λ*μ* as a used seal of its own credential.

However, if record characterizer and payers’ signature pair {G1, G2} in P(Pm, t, 1) is the one forged by Pm or someone else no one calculates G1, G2λ*μ* from G1λ*μ*.

Therefore in this case, A finds encrypted expenditure record E(D(Pm, t, 1)) that was used to calculate E(Sum(Pm, 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 G1, and determines Pk is liable when it calculates G1λ*μ*t by itself.

Namely, authority A examines payee’s signature G1 when it accepts paying record P(Pm, t, h), and Pk must honestly calculate it as a used seal of its credential. Also, Pk verifies the consistency of pair {G1, G2} at a time when Pm paid C(Pm, 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 Pk itself had forged {G1, G2} or not, A can determine Pk is liable, i.e. if Pk had accepted P(Pm, t, h) that includes inconsistent {G1, G2}.

As an exception if it is conspiring with A, Pk does not need to calculate G1λ*μ* honestly by using its credential when it reports P(Pm, 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 L1, L2, L3, ..., L1 to trace (partial) encryption forms of illegitimate total expenditure record Sum(Pm, t) = <e(Pm, t), U0(Rm(t))Xλ*μ*, G(Pm, t, 1)λ*μ*, S(Rm(t), G(Pm, t, 1))λ*μ* > calculated at step 4 in Sec. 5.1 back to E(Sum(Pm, t)) = <e(Pm, t), U0(Rm(t))Xλ*μ*, G(Pm, t, 1)λ*μ*, S(Rm(t), G(Pm, t, 1))λ*μ* >, and finds encrypted expenditure records E(D(Pm, t, 1)) = <e(D(Pm, t, 1)), 1, U0(Rm(t))Xλ*μ*, G(Pm, t, 1)λ*μ*, S(Rm(t), G(Pm, t, 1))λ*μ* >, E(D(Pm, t, 2)), ..., E(D(Pm, t, H(m(t)))) that constitute E(Sum(Pm, t)).

2. A discloses illegitimate total expenditure record Sum(Pm, t) with individual encrypted expenditure records E(D(Pm, t, 1)), E(D(Pm, t, 2)), ..., E(D(Pm, t, H(m(t)))) publicly.

3. For each illegitimate Sum(Pm, t) = <e(Pm, t), U0(Rm(t))Xλ*μ*, G(Pm, t, 1)λ*μ*, S(Rm(t), G(Pm, t, 1))λ*μ*>, each Pk calculates G(Pm, t, 1)λ*μ*τ from G(Pm, t, 1)λ*μ* by using its credential T(A, TRP, Rj).

4. When Pm calculates G(Pm, t, 1)λ*μ*τ that coincides with S(Rm(t), G(Pm, t, 1))λ*μ*, and Pm admits e(Pm, t) in Sum(Pm, t) is its correct total expenditure or expenditure records included in orphan record Sum(Pm, t) had already expired, Pm pays cash of value <e(Pm, t) - e(Pj, s, v), t> to A possibly with arrears but without revealing its identity. In a case where Sum(Pm, t) is an orphan record Pm pays e(Pm, t).

5. On the other hand when Pm does not believe that e(Pm, t) is its correct total expenditure or expenditure records included in Sum(Pm, t) had expired, it examines payer’s signatures in disclosed individual encrypted expenditure records. In detail, for each encrypted record E(D(Pm, t, h)), Pm extracts pair <G(Pm(t), h)λ*, S(Rm(t), G(Pm(t), h)λ*)> that examine whether relation G(Pm, t, h)λ*τ = S(Rm(t), G(Pm, t, h)λ*) holds or not. After that when relation G(Pm, t, h)λ*τ = S(Rm(t), G(Pm, t, h)λ*) does not hold, Pm without revealing its identity claims that E(D(Pm, t, h)) is not its record while convincing A that S(Rm(t), G(Pm, t, h)λ*) is not calculated by its credential (A receives this claim at step 9).

6. In a case where relation G(Pm, t, h)λ*τ = S(Rm(t), G(Pm, t, h)λ*) holds but Pm does not believe that E(D(Pm, t, h)) had expired, Pm picks P(Pm, s, v) from which it had generated P(Pm, t, h), and claims that the expiration time in P(Pm, s, v) is incorrect without revealing its identity.

7. When Pm claims that the expiration time in P(Pm, s, v) = <e(Pm, s, v), d(Pm, s, v), v, U0(Rm(t))λ*τ, U0(Rm(t))λ*τ, S(Rm(t), G(Pm, t, h)λ*)> is incorrect, A requests all cash holders to calculate used seal of G(Pm, t, h) while revealing their identities.

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

9. When anonymous Pm claims that E(D(Pm, t, h)) is not its expenditure record, A requests all cash holders to calculate used seals of their credentials from payer’s signature S(Rm(t), G(Pm, t, h)λ*) while revealing their identities.

10. A determines Pk that calculates payee’s signature S(Rk, S(Rm(t), G(Pm, t, h)λ*))λ*τ = S(Rm(t), G(Pm, t, h)λ*)λ*τ from S(Rm(t), G(Pm, t, h)λ*)λ*τ is liable and forces Pk to pay cash of value e(Pm, t, h) with penalty fine.

11. If no one paid for illegitimate expenditure records corresponding to Sum(Pm, t) or claimed they were incorrect, or no one calculates payer’s signature S(Rp, G(P, s, v)) at step 8, firstly, A requests all cash holders to calculate used seals of their credentials from record characterizer part value G(Pm, t, 1)λ* while revealing their identities. After that, A determines Pm that calculated S(Rm(t), G(Pm(t), t)λ*) is liable, and forces Pm to pay cash of value <e(Pm, t) - e(Pj, s, v)> with penalty fine. When Sum(Pm, t) is an orphan record Pm pays e(Pm, t).

12. If no one calculates S(Rm(t), G(Pm(t), t)λ*) in the previous step, A picks encrypted expenditure record E(D(Pm, t, 1)) = <e(e(Pm, t), 1), 1, U0(Rm(t))Xλ*, G(Pm(t), t)λ*, S(Rm(t), S(Rm(t), G(Pm(t), t)λ*))λ*τ>
constitutes \( \text{Sum}(P_m, t) \), and requests all cash holders to calculate used seals of their credentials from payer’s signature \( S(R_m, G(P_m, t, 1))^{x^\ast} \) while revealing their identities.

13. \( A \) forces \( P_k \) that calculates payee’s signature \( S(R_k, S(R_m, G(P_m, t, 1))^{x^\ast}) = S(R_m, G(P_m, t, 1))^{x^\ast} \) from \( S(R_m, G(P_m, t, 1))^x \) \( G(P_m, t, 1))^{x^\ast} \) to pay \( \{e(P_m, t) - e(P_j, s, v)\} \) with penalty fine, but in a case where \( \text{Sum}(P_m, t) \) 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 \( \text{Sum}(P_m, t) \) corresponds to it, i.e. payer’s signature \( S(R_m, G(P_m, t, 1))^{x^\ast} \) in \( \text{Sum}(P_m, t) \) is calculated from record characterizer \( G(P_m, t, 1))^{x^\ast} \) 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 \( S(R_m, G(P_m, t, 1))^{x^\ast} \), 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 \( P_j \) is honest, \( P_m \) that calculates payer’s signature \( S(R_m, G(P_m, t, 1))^{x^\ast} \) is determined as liable, i.e. only \( P_m \) can calculate \( S(R_m, G(P_m, t, 1))^{x^\ast} \) at step 11 as same as in the above. On the other hand when \( P_j \) is dishonest, \( P_m \) does not calculate \( S(R_m, G(P_m, t, 1))^{x^\ast} \) 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 \( G(P_m, t, h) \), \( S(R_m, G(P_m, t, h))^{x^\ast} \) while revealing its identity at Steps 7, 9, 11 and 12, it did not calculate \( G(P_m, t, h))^{x^\ast} \) of \( G(P_m, t, h))^{x^\ast} \) or \( S(R_m, G(P_m, t, h))^{x^\ast} \) 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 \( C(P_m, t, 1), C(P_m, t, 2), \ldots, C(P_m, t, 1, H(m, t)) \) 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,
Information Security and Computer Fraud

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 $C(P_s, s, v)$ 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