通信情報保護學會論文誌
第9卷 第1號, 1999. 3

익명 통신로를 이용한 Escrow 전자화폐

김 춘수*, 박 춘식*, 전 희종**,

Implementation of an Integrated Access Control Rule Script Language and Graphical User Interface for Hybrid Firewalls

Choon Soo Kim*, Choon Sik Park*, Hee Jong Jun**

요 약

기존에 제안된 대부분의 전자화폐는 개인의 프라이버시를 보호하기 위하여 추적 불가능 특성을 채택하고 있으나, 역효과로서 범죄에 사용되기 쉽다는 단점이 있다. 이를 보완하기 위하여 본 논문에서는 Escrow 방법을 도입하였다. 이 방법을 통하여 개인의 프라이버시 보호뿐만 아니라 범죄에 대한 예방책을 마련하였다. 또한 기존에 제안된 방법들을 설명하고 본 논문에서 제안한 방법과 비교하였다.

Abstract

Most of the previous researches for the electronic cash system guarantee unconditional untraceability for the purpose of individual privacy. Such untraceable electronic cash system that only focuses on untraceability, however, has side effect such as money laundering, criminal activities. We present a escrow cash model using anonymous channel that supports not only untaceability but also crime prevention, and prove the efficiency of our scheme relative to previous escrow cash systems.

1. 서 론

일반적으로 전자화폐를 위한 지불 시스템 또는 전자화폐 프로토콜은 사용자, 상점 및 은행으로 구성되어 있으며, 세 가지의 프로토콜을 가지고 있다. 즉, 인출단계, 지불단계, 예금단계의 프로토콜을 가지고 있으며 각 단계의 주체는 다음과 같다.

① 인출단계 : 은행 ↔ 사용자
② 지불단계 : 사용자 ↔ 상점, 사용자A ↔ 사용자B(양도)
③ 예금단계 : 상점 ↔ 은행, 사용자 ↔ 은행

* 한국전자통신연구원
** 숭실대학교 전기공학과

현재까지 제안된 대부분의 전자화폐는 개인의 프라이버시를 고려하여 설계되었기 때문에 정상적으로 사용된 전자화폐에 대하여는 추적 불가능(Untraceability)한 특성을 가지고 있다.[1,2,3,4,5,6] 이 방법들은 모두 개인의 프라이버시 보호에 대하여 무조건적인 안전을 보장하고 있다. 즉 이중(초과) 사용된 경우를 제외하고는 사용자의 ID 및 구매정보를 임의로 추적 수 없으며, 범법자들에 의한 구매, 증여 및 지불정보 등 화폐의 흐름도 추적할 수 없다. 또한 S. Von Solms와 D. Naccache[7]등은 추적 불가능한 익명 전자화폐는 기존의 종이 화폐 보다 돈 세탁, 납치 등의 범법 행위에 쉽게 이용될 수 있다고 지적하였다. 따라서 정부나 제정을 담당하는 국가기관에서는 불가능한 익명 전자화폐의 채택을 꺼리고 있는 실정이다. 그러나 다른 한편으로는 정상적인 사용자의 프라이버시는 반드시 보호 되어야 한다. 이러한 문제를 해결하기 위하여 개인의 프라이버시를 보호하면서 범죄행위를 방지 또는 추적할 수 있는 전자화폐 시스템들이 현재 활발히 연구중이다. 이를 실현하기 위하여 공정한 내용은 닉서명 방법[8]을 이용하는 방법과 Key-Escrow 기법[9]을 적용하는 전자화폐 등 여러 가지 방법이 제안되었고, 상기한 일반적인 지불 시스템에 법적으로 추적권을 보장 받을 수 있는 법원 등 공정기관이나 Escrow를 실행할 수 있는 제3자(Trusted Third Party;TTP) 즉 신탁기관(Trustee)이나 신탁법인 또는 법적 효력을 갖는 공정기관이 필요하며 5단계의 프로토콜이 기본적으로 필요하다.

① 등록단계 : 제 3자(법원 또는 신탁기관) ↔ 사용자
② 인출단계 : 은행 ↔ 사용자
③ 지불단계 : 사용자 ↔ 상점, 사용자A ↔ 사용자B
④ 예금단계 : 상점 ↔ 은행, 사용자 ↔ 은행
⑤ 추적단계 : 은행 ↔ 법원 또는 신탁회사 (법적인 자격을 갖춘 기관이 추적요청시 수행)

전자화폐의 요구사항으로서 개인의 프라이버시와 사회적인 안전성(범죄방지)의 양면이 고려되어야 한다. 이를 실현하기 위하여 Escrow 전자화폐를 도입하였다. Escrow 전자화폐의 요구사항으로서 기존의 전자화폐 요구사항에 4가지 사항을 추가하였다.[9]

[기존의 전자화폐 요구사항]

- 안전(이중사용 금지, 위조 방지, 사기 방지) : 만약 사용자가 전자화폐를 두 번 이상 사용한다면, 은행은 이를 추적하여 사용자의 ID를 알 수 있어야 하며, 사용자는 전자화폐나 예금 정보를 위조할 수 없어야 한다.
- 분할성(Divisibility) : 각 전자화폐는 작은 단위로 나누어질 수 있어야 한다.
- 양도성(Transferability) : 분할된 전자화폐는 사용자 사이에 서로 유통될 수 있어야 한다.
- Off-Line : 전자화폐는 Off-Line으로 지불 가능해야 한다.

[Escrow 전자화폐 구현 시 추가되는 요구사항]

- 프라이버시 보호 : 적어도 임의의 수의 TTP가 협조하지 않는 한 사용자의 프라이버시는 보존 되어야 한다. 즉 사용자가 범법행위를 하지 않았을 경우, 일정 수 이상의 TTP가 불법적으로 협조하지 않는 다면 사용자의 프라이버시는 보존되어야 한다.
- 지불정보추적 : 법원 등의 명령에 의하여 법 집행이나 범죄 방지를 위하여 모든 TTP가 협조한다면, 주어진 범죄자의 ID만으로 범죄행위를 위해 사용된 지불

정보를 알 수 있어야 한다.

- 지불자의 ID추적 : 모든 TTP가 협조한다면, 주어진 범죄행위와 관련되어 사용된 지불 정보만으로 지불자의 ID를 알 수 있어야 한다.
- 비밀키 노출에 대한 대책 : 강제 납치 등을 통하여, 은행이나 수탁기관의 관계자를 고문하여 돈을 인출하거나 비밀키 등이 노출 되었을 때, 이를 방지할 수 있는 기술적 방법이 있어야 한다.

범죄행위 추적을 수행하기 위한 기관은 신탁기관 또는 법원 등 공정기관으로 구성될 수 있으며, 구성에 의한 분류는 다음과 같다.

- 방법A : 1개의 기관으로 구성하여 독단적으로 모든 권한을 가지고 있는 방법
- 방법B : 여러 개의 기관으로 구성되어 있고 구성된 기관이 모두 동의(협조)해야만 추적이 가능한 방법
- 방법C : 여러 개(n)의 기관으로 구성되어 있으나 그 중 특정 수(p) 이상의 동의(협조)만 있어도 추적이 가능한 방법

위의 세가지 방법은 모두 장단점이 있으며, 경우에 따라서 또는 적용대상의 요구사항에 따라서는 이중 한 방법이 다른 방법보다 월등히 우수할 수 있다. 본 논문에서는 위의 방법B 경우에 있어서 적합하며, 방법A나 B로의 전환이 용이하고, 기존의 전자화폐의 기본 요구 사항과 추가된 요구사항을 만족하는 새로운 Escrow 전자화폐 시스템을 제안한다.

2. 기존의 연구결과

개인의 프라이버시와 사회적 안전을 고려한 전자화폐 시스템을 실현하기 위하여 Escow가 가능한 전자화폐를 구성하여 여러 편의 연구결과가 발표되었다. E. Brickell, P. Gemmell, D. Kravitz[10]은 두 명의 TTP가 협조했을 때 개인의 프라이버시를 밝힐 수 있는 방법을 제안하였으며, J. Camenisch, J. Piveteau, M. Stadler[11]은 그들이 제안한 내용은닉서명 방법[8]을 이용하여 공정한 내용은닉서명 방법을 제안하였다. 이 방법들은 TTP의 도움을 받으면 메시지에 대한 익명을 추적할 수 있는 방법이다. 또한 E. Fujisaki와 T. Okamoto가 D. Chaum의 Mix Net을 사용하여Practical Escrow Cash[9]을 제안하였다.

이중 Fujisaki, Okamoto가 제안한 프로토콜 중에서 Escorw의 핵심인 등록 프로토콜을 소개한다. 사용자 U는 RSA 모듈 N과 대응되는 지수 d를 비밀리에 생성한다. 그리고 U는 N을 다수의 신탁자 $T=(T_1, \ldots\ldots, T_t)$에게 등록하고 T로부터 라이센스 L을 받는다. 익명으로 TTP T_i 즉 $T_1, \ldots\ldots, T_t$ 에게 Mix net 방법[18]을 이용하여N을 보내고 Mix의 역방향을 통하여T_i 의 서명 L_i를 얻는다. ($E((\cdot)), D((\cdot))$ 는 비밀키 k를 이용한 암/복호화 방법을 나타내고, $E_1oE_2((\cdot))$는 공개키 암호시스템 $E_1(E_2((\cdot)))$를 나타낸다.

[프로토콜]

사용자 U는 RSA모듈 N과 이에 대응하는 비밀 지수 d를 생성한다. U와 $(T_1,\ldots..T_t)$ 사이의 등록 프로토콜은 다음과 같다. 단, 연산자 $||$는 연접(Concatenation) 을 의미함

① U는 $(k_1,\ldots..k_t)$를 선택하여 c_t를 재귀적(Recursive Computation)으로 계산한다.

$$c_i = E_i(k_i||c_{i-1}) \text{for } 1 \le i \le t, \text{ 단 } c_0 := ID_U(U\text{의 ID}) \quad (1)$$

U는 또한 $C_1 = E_1 o \cdots\cdots o E_t(N||c_t)$를 계산하고 U의 서명과 함께 T_1에게 보낸다.

② $1 \le i \le t$ 까지 T_i는 T_{i-1}의 서명을 확인하고, 복호화 한다.

$C_{i+1} = D_i(C_i)$, 단, $T_0 := U$ (2)

T_i는 자신의 서명과 함께 C_{i+1}을 T_{i+1}에게 보낸다.

③ T_i는 T_{i-1}의 서명을 확인한 후 식(3)으로부터 N, k_i, c_{i-1}을 얻는다.

$(N||c_i) = D_i(C_i)$, $(k_i||c_{i-1}) = D_i(C_i)$ (3)

T_i는 N을 위한 라이센스 L을 계산한고 $C_i = E_{ki}(L)$로 암호화 하고 자신의 서명과 함께 c_{i-1}과 C_i를 T_{i-1}에게 보낸다.

④ $1 \le i \le t-1$ 까지 T_i는 T_{i+1}의 서명을 확인하고 복호화 한다.

$(k_i||c_{i-1}) = D_i(c_i)$ (4)

T_i는 $C_i = E_{ki}(C_{i+1})$을 계산하고 자신의 서명과 함께 c_{i-1}, C_i를 T_{i-1}에게 보낸다.

⑤ T_1은 T_2의 서명을 확인하고 복호화 한다.

$(k_1||ID_U) = D_1(c_1)$ (5)

T_1은 $C_1 = E_{k1}(D_2)$를 계산하고 C_1을 자신의 서명과 함께 U에게 보낸다.

⑥ U는 T_1의 서명을 확인한 후 식(6)으로부터 라이센스 L을 얻는다.

$L = Dk_t$ o……o $Dk_1(D_1)$ $(C_1 = Ek_1$ o……o $E_{kt}(L))$ (6)

[단점]

Mix Net을 이용한 등록방법은 Park, Itoh, Kurosawa[15]가 지적한 바와 같이 Mix의 수가 증가하면 사용자 U가 만들어야 하는 $|c_i|$의 크기는 증가한다. 또한 사용자의 계산량이 많으며 각 Mix와 사용자는 두 가지의 알고리즘 즉, 공개키 알고리즘과 비밀키 알고리즘을 가지고 있어야 하며, 최종 Mix T_i는 3개의 암호 알고리즘을 가져야 한다. 따라서 키 관리가 복잡해지게 되어 이를 운영하는 데 어려움이 뒤따르게 된다.

3. 제안된 방법의 특징

본 논문에서는 Shuffle형 익명통신로[15]를 이용한 실질적인 Escrow 전자화폐 시스템을 제안하였다. 기본적인 개념은 사용자가 TTP로부터 자신의 정보와 관련된 라이센스를 받아 모든 지불행위에 사용하는 방법이며 그 특성은 다음과 같다.

① 프라이버시 보호 : 모든 TTP가 협조하지 않는다면 사용자의 프라이버시는 보장된다.

② 지불정보 추적 : 모든 TTP가 협조(또는 결탁)한다면 주어진 범죄자의 ID(이름)만으로 범죄행위를 위해 사용된 지불 정보를 알 수 있다.

③ ID 추적 : 모든 TTP가 협조한다면 주어진 범죄행위와 관련되어 사용된 지불 정보만으로 지불자의 ID(이름)을 알 수 있다.

④ 비밀정보 누출에 대한 대책 : 고문 및 납치에 의한 공격에 대하여 부분적으로 기술적인 보호가 가능하다.

⑤ 안전한 Off-Line 분할 가능한 현금 : 제안된 전자화폐는 Off-Line 지불환경에서 분할이 가능하며, 분할된 현금은 원래 분할되기 전 현금과 연결이 가능하다(모든 현금 및 분할된 현금은 추적 불가능하다.).

4. Shuffle형 익명통신로를 이용한 Escrow 현금

기본 개념은 등록, 인출, 지불, 예금 및 Transferring 프로토콜로 구성되어 있다.

[등록 프로토콜]

사용자 U는 자신이 사용하게 될 RSA 서명의 N과 대응되는 지수 d를 비밀리에 생성한다. 그리고 U는 N을 다수의 신탁자 $T=(T_1, \ldots\ldots, T_k)$를 통하여 등록하고 T_k로부터 ElGamal 서명 Pair인 $L=(L_r, L_s)$을 N의 유일한 라이센스로 받는다. 물론 사용자 U와 RSA 모듈 N과의 관계는 모든 TTP가 협조하지 않는다면 비밀로 유지된다. 또한 등록과정에서 T_1은 사용자의 ID를 얻게 되므로 TTP에게 자신의 신분과 거래를 위한 서명의 Modular N값 을 상호 위탁하게 된다.

[인출 프로토콜]

사용자 U는 은행 B로부터 현금 C를 인출한다.

[지불 프로토콜]

사용자 U는 C를 (N, L)과 함께 상점 V에게 지불한다.

[예금 프로토콜]

상점 V는 은행 B에게 지불된 값 C를 예금할 때 (N, L)를 함께 제시한다.

[양도 프로토콜]

사용자 U_1은 다른 사용자 U_2에게 현금 C를 전달하며 U_1 로부터 받은 (N, L)를 함께 전달한다.

본 연구에서 제안된 방식으로 메시지 N에 대한 TTP의 라이센스 (L_r, L_s)은 N에 대한 ElGamal 암호 시스템을 필요 충분조건으로 만족한다면 메시지 N에 대하여 TTP가 부여하는 라이센스로 유효하다. (e_B, n_B)는 은행 B의 RSA 공개키를 나타낸다. (e, N)은 사용자의 공개키에 해당되고 N은 사용자에 의하여 비밀로 유지된다. (시스템에서 e가 공개정보로 사용된다.) 또한 TTP의 서명 L은 T_k의 비밀키를 이용한 ElGamal서명방식이 사용된다. RSA 등 다른 내용은닉 서명방식으로 대체도 가능하나 다른 서명방식을 사용한다면 앞서 저적한 바와 같이 T_k가 자신의 ElGamal 공개키 암호 시스템용 비밀키 이외에 다른 서명용 알고리즘과 키가 필요하게 된다. 키 관리의 단순화를 위하여 본 논문에서는 ElGamal 공개키 암호시스템[16]을 사용하여 지불정보 N에 대한 라이센스를 제공 한다.

4.1 등록 프로토콜

다음의 프로토콜은 Shuffle형 익명통신로를 수정(식 7~27)하여 양방향으로 전개한 것이다. 즉 순방향 T_1에서 T_k까지 실행하여 T_k는 사용자의 N값을 얻고 역방향 T_k에서 T_1까지 실행하여 T_1은 U의 ID를 얻으며 U에게 T_k의 서명 L을 제공한다. 익명성과 안전성을 보장하기 위하여 동시에 등록되는 사용자의 수는 많을수록 유리하며 이는 익명통신로의 특징이기도 하다. 즉 각 단계에서의 TTP는 전달 받은 등록정보를 자신의 비밀키와 난수로 처리한 후 다음 TTP에게 사전식 배열로 정리하여 전달한다. 이 때 등록되는 정보는 각 TTP를 거쳐 완전히 뒤섞이게(Shuffle) 된다. 프로토콜의 식 및 설명을 간단히 하기 위하여 한 명의 사용자가 등록되는 과정을 설명하지만 실제로는 다수의 사용자가 동시에 등록한다. Shuffle형 익명통신로를 이용하여 등록 프로토콜을 실행하기 위해서 Mix 즉 TTP T_i와 사용자 U가 사용하는 관련 변수는 다음과 같으며 등록 단계의 모든 연산은 Modular q상의 연산이고 모든 변수는 q의 크기 $|q|$ 보다는 작다.

ElGamal Cryptosystem (q, g)

단, g는 GF(q) 상의 원시원(Primitive Element)
사용자 : U
TTP : T_j ($j=1$ ..... k)
U의 ID : ID_U
U의 비밀키 : $X_U \in \{1, ..., q-1\}$
U의 공개키 $Y_U (= g^{X_U} \bmod q)$
T_j의 비밀키 $X_j \in \{1, ..., q-1\}$
Tj의 공개키 $Y_j (= g^{X_j} \bmod q)$

[준비 단계]

사용자 U는 RSA 모듈 자신이 사용하게 될 RSA 서명의 N을 선택하고 이와 대응되는 지수 즉 비밀키d를 비밀리에 생성한다. 각 TTP는 자신이 사용할 난수 r_j를 선택한다.

[단계1]

U는 Shuffle형 순방향 익명통신로를 위한 난수 R을 선택하고 역방향 익명통신로를 위한 난수이며 라이센스를 숨기는 데 사용되는 R'를 선택한다. 사용자 U는 자신이 선택한 난수 R, R'를 비밀로 취급해야 한다. 단, R, R'의 조건은 식(7)과 같다.

$$R, R' \in Z^*_q \tag{7}$$

자신의 ID를 사용하여 c_k를 계산하며 이는 역방향으로 전달되면서 전개되어 T_1에게 ID_U를 제공하게 된다.

$$c_k = ID_U (\prod_{i=1}^{k} Y_i)^{R'} \bmod q \tag{8}$$

U는 자신의 RSA 서명으로 사용할 Modular N값과 ID가 숨겨져 있는 c_k및 c_k를 풀기 위한 난수 R'를 형태로 T_k에게 보내려고 한다. 이를 위하여 다음 식들을 계산한다.

$$t_1 = g^R \bmod q, \tag{9}$$

$$u_1 = N(\prod_{i=1}^{k} Y_i)^R \bmod q \tag{10}$$

$$v_1 = c_k(\prod_{i=1}^{k} Y_i)^R \bmod q \tag{11}$$

$$w_1 = g^{R'}(\prod_{i=1}^{k} Y_i)^R \bmod q \tag{12}$$

그리고 T_1에게 익명성이 보장되는 방법으로 (t_1, u_1, v_1, w_1)를 보낸다. 이 때 주의해야 할 사항은 다수의 사용자가 동시에 등록해야 한다는 사항이다.

[단계2]

T_1는 자신이 선택한 난수 r_1과 자신의 비밀키 X_1을 이용하여 식(13)을 계산한다.

$$(t_2, u_2, v_2, w_2) = (t_1, g^{r_1}, u_1 \frac{(\prod_{i=1}^{k} Y_i)^{r_1}}{t_1^{X_1}}, v_1 \frac{(\prod_{i=1}^{k} Y_i)^{r_1}}{t_1^{X_1}}, w_1 \frac{(\prod_{i=1}^{k} Y_i)^{r_1}}{t_1^{X_1}})$$
$$= (g^{R+r_1}, N(\prod_{i=2}^{k} Y_i)^{R+r_1}, c_k(\prod_{i=2}^{k} Y_i)^{R+r_1}, g^{R'}(\prod_{i=2}^{k} Y_i)^{R+r_1}) \bmod q \tag{13}$$

T_1은 생성된 (t_2, u_2, v_2, w_2)를 T_2에게 전달한다. 전달하는 방법은 앞서 설명한 바와 같이 다수의 등록정보를 상기 식으로 계산한 후 사전식 배열 순서로 재 배열하여 T_2에게 전달한다. 이를 일반화 하면 식(14)로 표현할 수 있다.

$$(t_{j+1}, u_{j+1}, v_{j+1}, w_{j+1}) =$$
$$(t_j \cdot g^{r_j}, u_j \cdot \frac{(\prod_{i=j+1}^{k} Y_i)^{r_j}}{t_j^{X_j}}, v_j \frac{(\prod_{i=j+1}^{k} Y_i)^{r_j}}{t_j^{X_j}}, w_j \frac{(\prod_{i=j+1}^{k} Y_i)^{r_j}}{t_j^{X_j}}) \bmod q \tag{14}$$

그리고 $1 \le j \le k-1$ 구간에서는 $(t_{j+1}, u_{j+1}, v_{j+1}, w_{j+1})$를 계산한 후 T_{j+1} 에게 보낸다.

또한 T_2에서 T_{k-1} 까지 위 과정을 거치면

(t_k, u_k, v_k, w_k)를 얻을 수 있다.

$$(t_k, u_k, v_k, w_k) = (g^{R+r_1+r_2+\cdots\cdots r_{k-1}},\ N \cdot Y_k^{R+r_1+r_2+\cdots\cdots r_{k-1}},\ c_k \cdot Y_k^{R+r_1+r_2+\cdots\cdots r_{k-1}},\ g^{R'} \cdot Y_k^{R+r_1+r_2+\cdots\cdots r_{k-1}}) \bmod q \quad (15)$$

순방향 단계에서 각 TTP는 추적단계를 위하여 별도로 저장하는 것은 없으며 단지 정해진 순서에 따라 자신의 변수로 계산하여 다음 TTP에게 전달하면 된다.

[단계3]

T_k는 U가 보내려고 하는 내용을 얻기 위하여 다음 식들을 계산하고 N, c_k, $g^{R'}$를 얻는다.

$$\frac{u_k}{t_k^{X_k}} = \frac{N \cdot Y_k^{R+r_1+r_2+\cdots\cdots r_{k-1}}}{g^{(R+r_1+r_2+\cdots\cdots r_{k-1})X_k}} = N \bmod q \quad (16)$$

$$\frac{v_k}{t_k^{X_k}} = \frac{c_k \cdot Y_k^{R+r_1+r_2+\cdots\cdots r_{k-1}}}{g^{(R+r_1+r_2+\cdots\cdots r_{k-1})X_k}} = c_k \bmod q \quad (17)$$

$$\frac{w_k}{t_k^{X_k}} = \frac{g^{R'} \cdot Y_k^{R+r_1+r_2+\cdots\cdots r_{k-1}}}{g^{(R+r_1+r_2+\cdots\cdots r_{k-1})X_k}} = g^{R'} \bmod q$$

단, $(Y_k = g^{X_k})$ (18)

여기서 N은 사용자가 사용할 지불정보이며 c_k에는 사용자의 ID가 숨겨져 있으며 $g^{R'}$ 는 c_k를 풀기위한 일종의 사용자 공개키이다.

또한 T_k는 N에 대하여 ElGamal 서명을 하여 $L=(L_r, L_s)$을 생성하기 위하여 다음 단계를 수행한다.

① $1 \le kl \le q-2$ 와 $\gcd(kl, q-1) = 1$인 조건을 만족하는 난수 kl을 선택한다.

② $L_r = g^{kl} \bmod q$ 와 $kl^{-1} \bmod q-1$ 을 계산한다.

③ $L_r = g^{kl} \bmod q$ 와 $L_s = kl^{-1}(N - X^k \cdot L_r) \bmod q-1$ 를 계산한다.

여기서 계산된 (L_r, L_s)은 N에 대한 T_k의 서명이고 이를 사용자 U에게 안전하게 전달하기 위하여 공개 게시판에 $N||(L_r, L_s)$형태로 게시한다. 다수의 사용자로 확장하면 공개 게시판에 다수의 $N||(L_r, L_s)$이 존재하게 되며 사용자는 자신이 알고 있는 N값을 검색하여 N에 대한 서명 (L_r, L_s)을 얻을 수 있고 ElGamal 서명값을 검증한 후 이상이 없으면 이를 받아들인다. 검증은 다음과 같은 단계를 거친다.

① $1 \le L_r \le q-1$ (19)

② $(Y_k)^{L_r} \cdot (L_r)^{L_s} \stackrel{?}{=} g^N \bmod q$ (20)

또한 모든 참여자 즉 은행, 사용자, 상점 등은 이를 열람할 수 있으므로 모든 참여자는 단순히 공개 게시판을 열람함으로써 N의 정당성을 확증할 수 있다. 그러나 이 방법은 On-Line 방법이고 본 논문의 기본 취지와 다르므로 그 사용에 대한 설명을 생략한다. 라이센스 L을 얻은 사용자는 자신만 알고 있는 비밀키 d를 사용하여 지불행위를 할 수 있으며 다른 제3자는 N에 대한 d를 알 수 없기 때문에 이를 사용할 수 없다. 또한 N에 숨겨져 있는 사용자의 신분은 그대로 비밀로 유지될 수 있으며 다음 역방향 익명통신로 처리과정을 통하여 사용자의 신분은 T_1에게 위탁된다.

역방향 전개 과정은 다음 단계에서 T_k에서 T_1까지 수행한다.

[단계4]

T_k가 전달하고자 하는 내용은 T_1에게 사용자의 신분 ID_U이고 먼저 전달할 내용을 다음과 같이 설정한다.

$$A_k = g^{R'} \bmod q \quad (21)$$

$$B_k = c_k = ID_U(\prod_{i=1}^{k} Y_i)^{R'} \bmod q \quad (22)$$

또한 식(23) 계산한다

$$(A_{k-1}, B_{k-1}) = (A_k \cdot g^{r_k},\ B_k \frac{(\prod_{i=1}^{k} Y_i)^{r_k}}{A_k^{X_k}}) \quad (23)$$

$$= (g^{R'+r_k},\ ID_U \prod_{i=k-1}^{i=1} Y_i)^{R'+r_k}) \bmod q$$

결국 T_k은 (A_{k-1}, B_{k-1})를 생성하여 T_{k-1}에 전달한다. 전달하는 방식은 순방향과 동일하게 사전식 배열로 재정열하여 다음 단계의 TTP에게 보낸다.

T_{k-1}는 식(24)를 계산하여 (A_{k-2}, B_{k-2})를 T_{k-2}에게 전달한다.

$$(A_{k-2}, B_{k-2}) = (A_{k-1} \cdot g^{r_{i-1}}, B_{k-1}\frac{\left(\prod_{i=k-2}^{i=1} Y_i\right)^{r_{k-1}}}{A_{k-1}^{X_{k-1}}}) \bmod q \quad (24)$$

이를 일반화된 식으로 표시하면 식(25)와 같다.

$$(A_{j-1}, B_{j-1}) = (A_j \cdot g^{r_j}, B_j \frac{\left(\prod_{i=j-1}^{i=1} Y_i\right)^{r_j}}{A_j^{X_j}}) \bmod q \quad (25)$$

T_k에서 T_2까지 위 과정을 거치면 (A_1, B_1)를 얻을 수 있으며 이를 T_1에게 전달한다.

$$(A_1, B_1) = (g^{R' + r_k + r_{k-1} + \cdots + r_2}, ID_u \cdot Y_k^{R' + r_k + r_{k-1} + \cdots + r_2}) \bmod q \quad (26)$$

[단계5]

T_1는 식(27)을 계산하여 사용자의 ID를 얻는다.

$$\frac{B_1}{A_1^{X_1}} = \frac{ID_u \cdot Y_1^{R' + r_k + r_{k-1} + \cdots + r_2}}{g^{(R' + r_k + r_{k-1} + \cdots + r_2)X_1}} = ID_U \bmod q$$
$$\text{단, } (Y_1 = g^{X_1}) \quad (27)$$

T_k에서 T_1 까지 위 과정에서 각 TTP가 향후 ID_U나 N값을 추적하기 위하여 보관해야 하는 것은 난수 r_j와 그와 관계된 모든 값들이다. 만약 TTP가 매 등록 시마다 다른 난수 r_j를 사용할지라도 추적을 위해서 TTP가 최소한 보관해야 할 정보는 입출력에 대한 연결만 가지고 있으면 된다. 이는 TTP에 대한 부담을 덜어주는 것이며 DB 구축 시 매우 유리한 방법이다. 또한 DB를 열람하는 방식으로서 추적이 가능하므로 계산적인 복잡도를 매우 경감시키는 구조가 된다. 이 외에 특별히 보관해야 될 정보는 없으며 자신의 비밀키와 난수만 잘 관리하면 된다. TTP에 의한 불법행위를 막기 위하여 TTP 자신이 사용한 난수를 보관하는 것은 필수적이다. 그러나 안전성을 고려할 때 난수들을 어떻게 하는 것이 효율적인 지는 시스템이 설치되는 환경과 사용자 그룹의 종류 등에 따라 선택하는 것이 바람직하다.

[단계6]

등록의 마지막 단계로서 T_1은 자신이 얻은 사용자들의 ID를 은행에게 열람시킨다(또는 전송). 이는 등록을 하지 않는 사용자는 인출을 막아주는 역할을 하며 사용자가 자신의 신분을 숨기기 위하여 사용자에 의하여 조작된 ID를 위탁했을 때에는 인출행위를 할 수 없다. 사용자가 하나는 자신의 ID를 또 다른 하나는 제3자의 ID로 이중으로 등록하여 사용하려고 할지라도 제3자와의 마찰이 발생하여 이중 등록이 검출되거나 2개의 동일한 N값이 검출되어 이중 등록이 추적될 수 있으므로 사용자에 의한 위법적인 등록은 사전에 검출될 수 있다.

4.2 인출 프로토콜

사용자 U가 현금 ϖ 원을 자신의 계좌에서 인출하고자 할 때 은행과의 인출 프로토콜을 통하여 ϖ 원에 해당하는 전자화폐 C를 얻을 수 있다. 인출 프로토콜에서 은행 B는 사용자 U의 공개키 N을 이용하여 내용 은닉 서명을 통하여 은행의 현금을 받고 자신이 생성한 Threshold Secret Sharing 방법(식 28, 29, 30, 31)을 통하여 λ를 TTP에게 등록한다. 은행이나 TTP의 비밀키가 유출되어 다량의 위조된 화폐가 통용될 때 λ의 등록여부를 판단하여 위조 화폐와 정상적인 화폐를 분별하게 된다. 본 논문에서는 사용자가 관리해야 하는 변수의 개수와 TTP가 관리해야 하는 변수 및 키의

개수를 줄이기 위하여 ElGamal 방식을 사용하고 있는 Shuffle형 익명통신로를 이용한 (h, k) Threshold Secret Sharing 방법을 사용하여 보다 효율적인 방법을 적용시킬 수 있었다. 단 새로운 키나 변수를 사용해야 하는 문제가 있으나 은행은 ε를 T_i에게 Off-Line 방식으로 전송될 수 있다.

U는 난수 λ를 생성하고 GF(q) 상의 차수 $h-1$ 인 다항식 $\lambda(x)$를 선택한다.

$$\lambda(x)=\lambda+\lambda_1 x+\cdots\cdots+\lambda_{h-1}x^{h-1} \bmod q$$
$$\text{단, } 0<\lambda<q,\ h\le k-1$$

또한 U는 다음의 식을 계산하여 익명통신로를 이용하는 방식으로 T_2에서부터 T_k까지 다음식을 각각 보낸다. 각 식들은 익명통신로를 거치면서 각 TTP에게 $\varepsilon=\lambda(i)$가 전달된다. 즉 T_i에게 전달되는 ε는 $T_1 \sim T_{i-1}$을 통하여 T_i에게 전달되는데 다음식을 계산한 후 ε를 Z와 함께 B에게 전송한다.

$$\varepsilon_1=g^{R'}$$
$$\varepsilon_2=\lambda(2)\left(\prod_1^2 Y_i\right)^{R'} \bmod q \quad (28)$$
$$\cdots\cdots\cdots$$
$$\varepsilon_k=\lambda(k)\left(\prod_1^k Y_i\right)^{R'} \bmod q$$
$$\varepsilon=\{\varepsilon_1,\ \varepsilon_2,\ \cdots\cdots\ \varepsilon_k\} \quad (29)$$
$$Z=\gamma^{e_B} H(N||\lambda||\varpi) \bmod n_B \quad (30)$$

여기서 $\gamma \in Z_{n_B}$은 사용자가 은행으로부터 내용은닉 서명을 받기 위하여 선택된 난수 이며 H는 일방향 해쉬함수이고 모든 참여자에게 공개된다. 그리고 나머지 인출을 위한 프로토콜의 단계는 아래와 같다.

[단계1]

B는 $W=Z^{1/e_B} \bmod n_B$을 U에게 보내고 U의 계좌에서 ϖ 원을 인출 시킨다. 또한 ε와 인출 값 ϖ 를 T_i에게 보낸다.

[단계2]

U는 다음을 계산하여 γ 을 제거하고 전자화폐 C 를 얻는다.

$$C=W/\gamma \bmod n_B=(H(N||\lambda||\varpi))^{1/e_B} \bmod n_B \quad (31)$$

[단계3]

각 T_i는 자신의 데이터 베이스(λ-Database) 안에 ε 을 복호화 한 값 $\lambda(i)$를 ϖ 와 함께 저장한다. 물론 이 단계에서 TTP가 얻는 ε 에 대하여 은행은 그 사용자를 알 수 있으나 ε에 숨겨진 값은 알 수 없으며 TTP중 h 명 이상의 협조가 있다면 ε 에 숨겨져 있는 λ 값을 알 수 있다.

4.3 지불 프로토콜

사용자 U는 지불 프로토콜을 사용하여 상점 V에게 y ($\le \varpi$)원을 사용한다고 가정한다. 지불 프로토콜은 인증과 사용자 서명 두 단계로 구성되어 있다. 이 단계에서 사용자는 자신의 서명키인 N을 사용하며 이는 지불정보로 인식되고 향후 사용자의 신분을 추적할 수 있는 근거가 된다. 또한 사용자는 N과 함께 N에 대한 서명값 L을 사용함으로써 상점은 N의 유효성을 판단하고 사용자는 자신의 정당성을 입증하게 되는데 각 단계는 다음과 같다.

[단계1]

U는 (y, N, L, λ, C)를 V에게 보낸다.

[단계2]

V는 $y \le \varpi$ 을 확인하고 (N, L)가 유효한지를 검증한다. 또한 $C^{e_B} \equiv H(N||\lambda||\varpi) \bmod n_B$ 여부를 검사하여 서명L과 C의 유효성을 검증한다. 만약 모두 유효하다면V는 U에게 Time

Stamp τ 와 V의 ID ID_V 를 보낸다. 그렇지 유효하지 않으면 V는 지불 프로토콜을 중지한다.

[단계3]

U는 N과 대응되는 자신의 비밀키로 서명 S를 계산하여 V에게 보낸다.

$$S = H(y||ID_V||\tau)^d \bmod N \quad (32)$$

[단계4]

V는 사용자의 공개키을 사용하여 다음 식이 만족된다면 $(y||ID_V||\tau)$와 S를 y원에 대한 유효한 지불로 받아들인다.

$$S^e = H(y||ID_U||\tau) \bmod N \quad (33)$$

지불 단계에서 은행에서 인출된 ϖ는 여러 종류의 y로 나눠질 수 있다. 본 논문에서 제시하는 방식은 인출된 돈 그 자체를 가공(예: 돈의 가치에 대하여 암호화 또는 서명)하지 않기 때문에 사용자의 책임하에 $0 \sim \varpi$까지 지불이 가능하게 된다. 물론 사용자와 상점은 ϖ, y값을 알 수 있으며, 잔여금액은 사용자에 의하여 관리되어야 한다. 초과되는 금액은 일차적으로 [단계2]에서 검증되며, 사용자가 사용한 총액은 은행에 의하여 검증된다. 즉 기존의 전자화폐 시스템은 재사용에 대하여 은행이 검증하지만 본 논문에서 제한한 방식은 초과사용을 검증하게 된다.

[초과사용]

초과 사용에 대한 검출을 위하여 은행은 사본 $\{y_i, N, L, \lambda, C, Id_i, \tau_i, S_i\}$ 정보를 자신의 DB에 보관한다. 새로운 예금행위가 발생하면 은행은 최종 취득자로부터 받은 사본 $\{y, N, L, \lambda, C, ID_v, \tau, S\}$ 정보 중 N값을 자신의 DB에 보관되어 있는 N값과 비교한다. 그리고 같은 N값이 검출되면 $\Sigma y_i \leq \varpi$ 를 비교하고 이상이 없으면 새로운 DB로서 보관한다. 만약 초과사용이 검출되면 은행은 N에 해당하는 사본을 이를 평가할 수 있는 기관에 제출한다. 이 사본 중 C은 U 이외는 어느 누구도 위조할 수 없기 때문에 은행은 초과한 사본을 위조할 수 없다. 따라서 은행이 제시한 초과된 사본에 대한 진위는 논의할 문제가 아니다. 즉 이를 평가하는 기관은 은행의 사본을 전적으로 신뢰할 수 밖에 없으며 이 경우 정해진 방법에 따라 모든 TTP의 협조를 요청한다. 이 때 은행은 N값을 제시하며 모든 TTP들의 협조하에 T_k로 부터 T_1 순서로 N에 해당하는 사용자의 ID를 추적하여 초과사용자를 색출한다. 이 때는 N과 관련하여 보관중인 (A_{k-1}, B_{k-1})를 T_{k-1}에게 전달한다.

4.4 예금 프로토콜

상점 V는 y원을 예금하기 위하여 지불 사본$(y, N, L, \lambda, C, ID_V, \tau, S)$을 은행 B에게 보낸다. 각 항목에 대한 내용을 고찰하면 N, L, τ은 사용자의 등록여부를 판단하는 정보이고 C는 은행이 서명한 전자화폐에 해당한다. ID_V는 전자화폐의 최종 취득자에 대한 정보이며 τ와 함께 S를 검증할 때 사용되었는지 여부를 판단하는 정보가 된다. 만약 U가 초과사용 했다면 은행 B는 사본$(y, N, L, \lambda, C, ID_V, \tau, S)$을 통하여 $\Sigma y_i > \varpi$ 여부를 판단할 수 있으며 은행 B는 초과 사용이 검출된 사본들을 증거로 제시하면서 N으로 자신을 은닉하고 있는 불법사용자를 밝혀 줄 것을 TTP 에게 요구한다. 이 경우 TTP는 정당한 요구에 대하여 모든 TTP가 협조하여 사용자의 ID를 추적한다. 여기서 주목할 만한 사항은 불법 사용자를 찾아내는 추적 방법은 기존의 전자화폐 시스템에 비하여 매우 간단하다는 것이다. 앞서 살펴본 TTP를 사용하지 않는 다른 전자화폐 프로

토콜에서는 이중사용 방지를 위하여 매우 복잡한 수식을 사용한다[4][17][18]. 이는 TTP를 사용하는 방법의 또 하나의 장점이며 주목할 만한 사항이며, C를 초과 사용한 사본은 U를 제외한 어느 누구도 위조할 수 없다.

4.5 Transferring 프로토콜

사용자 U_1이 또 다른 사용자 U_2에게 $y(\leq \varpi)$원을 전달하려고 할 때 U_1은 U_2와 함께 전달 프로토콜을 실행하면 된다. 상점의 ID ID_V가 U_2의 공개키 N_2로 바뀌는 것을 제외하면 지불 프로토콜과 같다. 전달된 화폐의 내용은 $(y, N_1, L_1, \lambda_1, C, N_2, \tau, S_1)$이며 S_1은 익명의 사용자 N_1이 또 다른 익명의 사용자 N_2에게 y원을 전달 했다는 서명을 나타낸다. U_2가 전달 받은 y원을 사용하여 V에게 z원을 지불하고자 한다면 V는 서명 N_2와 $S_2 \equiv H(z||ID_V||\tau) \bmod N$을 받기 전에 $(N_1, L_1, \lambda_1, C, N_2)$을 점검하고 (y, N_2, τ_1, S_1)의 유효성을 점검한다. 만약 $y \leq \varpi$이고 $S_1^e \equiv H(y||N_2||\tau_1) \bmod N_1$ 이면 V는 (y, N_2, τ_1, S_1)를 받아들인다.

4.6 추적 프로토콜

앞서 설명한 바와 같이 본 연구에서 제안한 방식은 네 가지의 불법행위를 추적할 수 있다. 그 중 초과 사용은 4.3절에서 설명하였다. TTP에게 추적을 요청하기에 앞서 추적에 대한 정당한 절차가 선행되어야 하며 이에 대한 방법을 기술하였다.

[사용자 ID 추적]

만약 상점 또는 특정인과의 불법적인 거래가 전자화폐를 통하여 사용되었다면 불법적인 거래를 적발하기 위하여 법적인 자격을 가지고 있는 기관에서 거래당시 사용되었던 N값 즉 지불정보를 TTP에게 제시하여 사용자의 ID 추적을 요구할 수 있다. 이 경우 초과사용의 경우와 동일하게 사용자의 ID를 추적할 수 있다.

T_k로부터 T_1순서로 N에 해당하는 사용자의 ID_U를 추적하기 위하여 T_k는 N과 관련하여 보관중인 (A_k, B_k)를 T_{k-1}에게 전달하고 T_{k-1}는 (A_k, B_k)의 출력값인 (A_{k-1}, B_{k-1})을 찾아서 T_{k-2}에게 전달하는 방법으로 T_1까지 진행된다. T_1는 (A_1, B_1)와 연관된 ID_U을 찾아 공개한다. 공개된 ID_U을 근거로 특정 지불정보에 대한 사용자의 ID를 찾아낼 수 있다.

[지불정보 추적]

범죄에 가담했거나 가담할 가능성이 높은 특정인에 대한 지불정보 즉 지불정보(거래내역)를 감시하기 위하여 법적인 정당성이 부여된 기관으로부터 TTP에게 특정인의 ID를 제공하고 전자화폐의 지불단계에서 사용되는 N값을 추적해 줄 것을 요청할 수 있다. 이 때 모든 TTP가 동의 한다면 사용자의 ID로부터 지불정보 N을 추적할 수 있다.

T_1로부터 T_k순서로 ID_U에 해당하는 사용자의 N값을 추적하기 위하여 T_1는 ID_U과 관련하여 보관중인 (A_1, B_1)를 T_2에게 전달하고 T_2는 (A_1, B_1)의 입력값인 (A_2, B_2)을 찾아서 T_3에게 전달하는 방법으로 T_k까지 진행된다. T_k는 (A_k, B_k)와 연관된 N을 찾아 공개한다. 공개된 N을 근거로 은행을 통하여 특정인의 최종 거래 내역을 알 수도 있고 각 상점에 수배를 위한 조치를 내릴 수도 있다.

[비밀키 노출시 추적]

납치나 고문 등을 통하여 TTP나 은행의 비밀 키가 범죄자에게 드러난다면 TTP는 발행된 모든 현금의 λ값(현금 번호)을 복구하기 위하여 협조한다. 즉 h명 이상의 TTP를 신뢰할 수 있다면 복구는 성공적으로 수행될 수 있다. 그리고 복구된 λ를 λ-Database에 저장한다. 이

후로는 정상적인 지불 프로토콜을 중지하고 모든 지불 행위를 On-Line 검사 방법으로 처리한다. On-Line 검사 방법을 통하여 상점 V에게 전달된 모든 λ 값은 이미 저장된 λ-Database에 저장되어 있는지 여부를 검사한다. 만약 현금의 λ 즉 C가 λ-Database에 저장되어 있지 않다면 위조된 현금으로 간주한다.(즉 지불자는 체포된다.) 발행된 현금이 모두 사용되면 λ 값을 λ-Database에서 지운다. 이러한 On-Line 검사 방법은 비상시 임시적인 방법이며 모든 현금이 은행이나 TTP의 새로운 키로 변경되기 전까지 제한적인 기간(비상시 키 교체 기간)동안 수행된다. 거의 모든 키 교체가 끝나면 은행의 새로운 키를 사용하여 Off-Line 방식으로 사용되며 교체되지 않은 키를 사용하는 사용자는 On-Line 검사 방법을 계속 수행한다. 인출 과정에서 사용자 U가 유효하지 못한 λ를 TTP에게 보낸다면 이는 U에게 불이익을 초래할 뿐이며 아무런 이점이 없다. 따라서 λ의 유효성을 검사할 필요는 없다.

5. 안전성/효율성 및 구현

본 논문에서 제시한 프로토콜은 기본적으로 전자화폐에서 요구하는 여섯 가지 조건을 만족해야 한다. 즉 물리적 요구조건에 대한 독립성, 위조 및 복사 방지, 추적 불가능성, 상점과 사용자와의 Off-Line 거래, 분할성 및 양도성을 만족해야 한다. 이중 위조 및 복사 방지는 시스템 구현방식에 의존하기 때문에 실현문제에 해당하나 위조 및 복사하여 사용하더라도 사후에 검출된다. 본 절에서는 이러한 불법 또는 고의적인 행위에 대하여 제시된 프로토콜이 얼마나 안전한 가를 평가한다. 기본적으로 제안한 프로토콜의 안전성은 소인수 분해 문제와 이산대수 문제 및 난수에 의한 암호화에 의존한다. 따라서 안전성이 수학적으로 입증된 Tool들을 사용하기 때문에 각 단계에서의 프로토콜은 안전성에 대한 증명이 입증되므로 전반적으로 안전하다고 볼 수 있다. 즉, 본 논문에서 제시한 프로토콜에서 Modular 값 크기는 그 시스템의 안전성을 평가할 수 있는 수학적 근거자료가 될 수 있다. 그러나 그 크기가 커지게 되면 전달해야 되는 정보량이 증가하므로 효율성 측면에서는 손상을 입는다. 따라서 Modular 값을 결정하는 문제는 적용되는 시스템의 주변환경과 관계가 있으며 최적값을 찾는 문제도 쉽지 않다. 본 논문에서 제안된 방식은 사용되는 환경에 맞추어 q값의 크기를 가변할 수 있다. 각 단계에서 사용한 대표적인 Modular 값을 나열하면 표1과 같다. 변수 하나의 최대 크기를 512Bit로 선정한다면 전송되는 순수 정보량은 최대 $|n_m| \times 6$(=3,072Bit)를 넘지 않기 때문에 전송해야 되는 정보량은 기존에 제안된 전자화폐 시스템에 비하여 매우 적기 때문에 효율적인 방안이다. 512Bit 소인수 분해에 대한 안전성은 2^{256} 정도의 계산량으로 표시할 수 있으며 상용 시스템에서 현재까지 안전한 것으로 평가되고 있다. 또한 전자화폐 구현시 전자화폐 Media(스마트 카드)의 계산 처리 능력이 문제가 되고 있다. 그러나 본 논문에서 제안된 방법은 계산량을 최소화 하였기 때문에 현재 개발된 Biprocessor Type의 스마트 카드에서 실시간 처리가 가능하다.

표1 단계별 Modular 값

단 계	변 수	크 기
등록단계	q	512Bit
	N	$\langle\|q\|$, 256Bit
	g, R, R', IDU, L_r, L_s	각각에 대하여 $\langle\|q\|$
	(t_1, u_1, v_1, w_1)	최대 $\|q\|$ x 4
	(L^c_r, L^c_s)	최대 $\|q\|$ x 2
인출단계	n_{ω}	512Bit
	y	수십 Bit
	Z	$\langle\|n_{\omega}\|$
	ε	수십 Bit x TTP의 수
	(Z, ε)	$\|n_{\omega}\|$ 보다 수십 Bit 큼
	C, W	각각에 대하여 $\langle\|n_{\omega}\|$
지불단계	(y, N, L, λ, C)	최대 $\|n_{\omega}\|$x5
	(τ, ID_v)	수십 Bit
	S	$\langle$ |N|
예금단계	$(y, N, L, \lambda, C, ID_v, \tau, S)$	최대 $\|n_{\omega}\|$ x 6

5.1 사용자에 의한 불법행위

전자화폐 시스템에 대한 불법행위 중 사용자에 의한 불법행위의 발생확률이 가장 높으며 이에 대한 대책을 아래에 기술 하였다.

[위조]

은행의 공개키 e_{ω}만 알고 비밀키 $1/e_{\omega}$ 를 모르는 사람이 자신의 선택한 γ 값을 가지고 생성한 Z에 대한 은행이 서명한 값 W를 생성하는 것은 거의 불가능하며 이는 소인수 분해 문제에 해당한다. 따라서 Modulo n을 사용한다면 약 $2^{n/2}$ 정도의 계산량이 필요하므로 위조하는 것은 현실적으로 매우 어려운 문제로 알려져 있다. 예로서 n의 크기를 512Bit라고 했을 때 $2^{512/2}=2^{256}$ 정도의 계산량이 필요하게 된다. 이 계산량을 십진으로 환산하면 약 10^{85}정도이고 500Mhz속도를 갖는 컴퓨터가 1 Clock(또는 1 State)에 하나의 계산을 한다고 가정할 경우 최대$2x10^{76}$초가 소요되며 이를 년으로 환산하면 0.6×10^{67}년이라는 엄청난 시간이 소요된다. 물론 제 3자가 통신로 상에서 가로채기로 W값을 얻었을 지라도 γ 값을 모르기 때문에 C를 생성할 수 없으며 이는 One-Time Key와 같은 계산량을 가지므로 무조건 안전하다고 볼 수 있다. 여기서 γ 의 크기는 $|\gamma|\langle|n_{\omega}|$ 이다

[재사용]

재사용과 복사의 경우는 같은 범주에 속한다. 따라서 복사하여 사용한 것이나 재사용한 화폐의 경우에는 이중 사용 검출에 의하여 사용자의 신분이 밝혀지게 된다. 제 3자에 의하여 복사된 경우에도 지불 프로토콜 단계에서 제 3자는 사용자의 N과 연결되는 사용자의 비밀키 d를 생성할 수 없기 때문에 제3자는 이를 지불 프로토콜 단계에서 사용할 수 없다. 복사는 전자화폐의 물리적인 전달 수단인 스마트 카드 등의 Temper Proof 능력에 일차적으로 의존한다. 그러나 전자적 데이터 형태로 저장된 전자화폐와 관련된 정보는 완벽하게 복사할 수 있다고 가정하고 전자화폐 시스템

을 설계하기 때문에 물리적 전달 수단에 전적으로 의존하고 있는 전자화폐 시스템을 제안한 경우는 매우 드문 편이다.

[도난]

전자화폐를 도난 당한 경우에도 훔친 전자화폐를 사용할 수 없다. 그 이유는 복사된 경우와 마찬가지로 훔친 사람은 사용자의 N과 연결되는 비밀키 d를 생성할 수 없기 때문에 지불 프로토콜 단계에서 자신이 정당한 사용자라는 것을 입증할 수 없다.

5.2 은행에 의한 불법행위

은행에 의한 불법행위는 발생할 확률이 매우 적으며 대부분 은행에 의한 경우 보다는 은행에 근무하는 개인이 의도적으로 행하는 행위이다.

[이중사용누명]

일회 사용한 정당한 사용자에게 은행은 이중 사용하였다고 누명을 씌울 수 있다. 그러나 본 논문에서 제안한 방법에서는 사용자의 비밀키인 d값을 모르는 은행이 또 다른 τ' 값에 대한 사용자의 서명인 $S=H(y||ID_V||\tau')^d \bmod N$를 생성할 수 없기 때문에 이중사용에 대한 누명은 소인수 분해 문제에 해당한다.

[인출거부]

인출거부에 대한 대책은 본 논문이 제시한 프로토콜에는 포함되어 있지 않지만 인출 거부에 대한 문제 해결은 현재 은행의 입출금 시스템인 사용자의 통장 등으로 해결되는 정도의 일반적인 문제로 간주한다.

[사용자 신분 노출]

은행은 인출 시 사용자와 사용자가 인출하는 돈의 양을 알 수 있다. 이는 일반적인 경우로서 이를 숨기는 것은 비밀은행(비밀 금고)의 기능이기 때문에 본 논문에서 제시한 방법으로는 이에 대한 보호 대책은 마련할 수 없지만 상점 또는 최종 취득자로부터 받은 지불 사본(y, N, L, λ, C, ID_V, τ, S) 값으로부터 예금단계에서 돈에 대한 사용자의 신분은 알 수 없다. 단지 정당한 절차에 따라 N값을 제시하면 모든 TTP의 협조하에 법집행 차원에서 최초 은행으로부터 인출한 사용자의 ID를 찾을 수 있으며 이는 본 논문에서 제안된 방식의 요구조건이다.

[예금거부]

예금거부 문제는 예금으로 입금하려고 하는 C값에 대한 정당성 문제이다. 만약 은행이 상점에서 받은 N, λ, C값으로부터 다음식을 검증하였을 때 일치하지 않으면 사용자에 의하여 C가 위조 되었거나 C값이 손상을 받은 것으로 간주한다.

$$C \stackrel{?}{=} (H(N||\lambda||\varpi))^{1/e_\varpi} \bmod n_\varpi \qquad (4.45)$$

만약 일치한다면 정당한 사용자이며 그 금액도 정당한 것이다. 그러나 일치하는 경우에 은행이 의도적으로 인출을 거부하면 법적인 문제로 확대될 수 있으며 은행은 자신의 비밀키인 $1/e^\varpi$ 를 공개해야 하는 경우 까지도 있을 수 있다. 다시 말하면 $H(N||\lambda||m))$ 값에 대하여 C를 생성할 수 있는 기관은 $1/e^\varpi$ 를 알고 있는 은행뿐이다. 이는 은행과 같은 상당히 공신력이 있는 기관에서는 자신의 비밀키를 공개하는 위험을 감수하면서까지 부당한 예금거부를 하지는 않는다는 것이 상식이다

5.3 상점에 의한 불법행위

상점은 사용자의 범주에 속하며 다음과 같은 불법행위가 있다.

[위조]

사용자의 경우와 마찬가지 이유로 상점은 C를 위조할 수 없다. 즉 은행의 공개키 e_m만 알고 비밀키 $1/e_m$를 모르는 사람이 자신의 선택한 γ값을 가지고 생성한 Z에 대한 은행이 서명한 값 W를 생성하는 것은 매우 어려우며 이는 소인수 분해 문제에 해당한다.

[재사용]

사용자와의 프로토콜의 수행결과 생긴 지불 사본(y, N, L, λ, C, ID_V, τ, S)을 복사하거나 재사용했을 경우에는 재사용한 C에 대하여 자신의 ID인 ID_V가 드러나게 되므로 재사용 및 의도적인 복사는 범법행위로 드러난다. 또한 ID_V만 바꿔 치기 했을 때는 사용자의 비밀키 d를 모르는 상점은 위조한 ID_V 에 대한 S'를 계산하는 것은 소인수 분해 문제에 해당한다. 또한 사용자로부터 받은 (y, N, L, λ, C)를 이용하여 지불 사본(y, N, L, λ, C, ID_V, τ, S)을 생성하여 사용하는 것 또한 사용자의 비밀키 d를 찾아내야 하는 문제 이므로 위에서 열거한 문제에 해당한다.

[사용자 신분 노출]

수표와는 달리 현재 화폐 시스템에서와 마찬가지로 사용자에 대한 상세 신분은 알 수 없다. 단지 사용자의 용모정도를 알 수 있게 되며 이는 법적인 구속력을 갖지 못한다. 즉 상점에 의한 사용자의 구매정보 노출은 상점의 매상과 관계된 것이며 이는 막을 수 없다. 그러나 수표 시스템에서와 같이 사용자의 ID 등 세부정보는 사용자로부터 받은 (y, N, L, λ, C)로부터 얻을 수 없다.

[구매거부]

사용자로부터 받은 C를 검증하여 정당하지 않으면 상점은 합법적으로 구매를 거부할 수 있다. 그러나 그 외의 경우 상점이 정당한 화폐(C)에 대하여 구매를 거부할 이유가 없다.

5.4 결탁 및 TTP에 의한 불법행위

기본적으로 TTP는 객관적으로 공정성이 입증된 기관이나 개인을 선정한다. 그러나 외부의 압력에 의한 불법행위가 있을 수 있으며 아래 열거한 사항들은 대부분 TTP의 결탁에 의한 불법행위이다.

[자진결탁]

TTP 전체가 법적인 정당한 요구 없이 결탁하더라도 사용자의 ID와 라이센스 L과의 관계를 알 수 있으며 이를 근거로 사용자의 거래내역을 쉽게 알 수 있기 때문에 전자화폐의 기본 요구 조건인 추적불가능성을 상실하게 된다. 그러나 이 경우는 TTP 모두가 법집행의 사유 없이 불법으로 모든 TTP가 결탁하기는 매우 어려울 것으로 보며 수탁을 맡는 기관이나 개인의 자질이 문제가 될 것이다. 더 많은 TTP를 두어 공정성을 보장할 수 있으나 효율적인 문제가 뒤 따른다. TTP의 자진결탁에 대한 대책은 TTP를 선정 시 얼마나 공정한(정직한) TTP를 선정하는가 하는 문제에 달려 있다.

[외부압력에 의한 결탁]

TTP 전체가 법적인 정당한 요구 없이 외부의 압력에 의해 결탁하여 사용자와 거래내역을 불법으로 외부에 유출하는 것으로 이 또한 자진 결탁과 같은 문제이다.

[등록방해]

특정 TTP T_i가 등록을 방해할 목적으로 사용자로부터 전달되는 정보를 변형하여 다른 TTP T_{i+1}또는 T_{i-1}에게 전달할 경우 T_i은 존재하지 않는 사용자의 ID를 얻게 되며 이 경우 발행된 라이센스 L은 사용할 수 없게 된다. 사

용자가 등록을 신청하고 응답의 없으면 문제를 제기하게 되며 만약 사용자가 정당한 등록 절차를 밟았다면 추적에 의하여 정보를 변형한 T_i를 찾아낼 수 있다.

[등록위조]

사용자와 결탁한 특정 T_i에 의해서 추적 불가능한 라이센스를 생성할 수 있는 가능성의 문제이다. 이 문제는 이산대수의 수학적 문제이므로 안전하다. 즉 특정 T_i가 최종 TTP를 거치지 않고 N과 대응되는 L을 찾아내는 문제로서 함수 V_T의 수학적 특성에 의존한다.

[등록위조]

사용자와 결탁한 특정 T_i에 의해서 추적 불가능한 라이센스를 생성할 수 있는 가능성의 문제이다. 이 문제는 이산대수의 수학적 문제이므로 안전하다. 즉 특정 T_i가 최종 TTP를 거치지 않고 N과 대응되는 L을 찾아내는 문제로서 함수 V_T의 수학적 특성에 의존한다.

6. 결 론

본 논문에서는 지불정보 추적 및 ID 추적을 동시에 실현하기 위한 방법을 제시하였다. 수탁기관을 통하여 발행한 라이센스를 사용하기 때문에 초과사용을 위하여 복잡한 정수이론을 사용하는 수학적 기법을 사용하지 않아도 되며, Key-Escrow 기법을 사용하는 기존의 Escrow 전자화폐를 보다 쉽게 구현할 수 있다. 또한 분할성과 Transferable 을 해결하기 위하여 기존의 방법에서는 많은 양의 수학적 계산을 필요로 하는 반면에 본 논문에서 제안한 방법은 매우 쉽게 실현할 수 있다. 모든 사용자는 자신의 전자화폐의 번호를 다수의 센터에 비밀 공유 방법으로 등록해야 한다. 즉 인출단계에서 사용자는 은행을 통하여 TTP에게 자신의 현금 비밀을 등록한다. 등록된 번호는 사용된 현금의 불법여부를 확인하는 데 사용된다. 제시된 전자화폐 시스템의 특징은 다음과 같다.

① 일반적인 전자화폐의 요구조건 만족
② 사용자의 ID 및 거래 내역 추적 가능
③ 비밀정보 유출에 대한 대책
④ 현실적으로 실현 가능한 Escrow 전자화폐 제시
⑤ 사용자의 계산량 감소(스마트카드로 구현 가능)
⑥ TTP의 수에 관계없이 등록되는 정보의 크기는 일정
⑦ TTP는 한 종류의 키만 관리

참고문헌

[1] S. Brands, " Untraceable Off-line Cash in Wallet with Observers", Proceedings of Crypto 93, pp.302-318, 1994.

[2] D. Chaum, A. Fiat, and M. Naor, "Untraceable Electronic Cash", Proceedings of Crypto 88, pp.319-327, 1990.

[3] T. Eng and T. Okamoto, "Single-Term Divisible Coins", Proceedings of Eurocrypt 94

[4] M. Franklin and M. Yung, "Secure and Efficient Off-Line Digital Money", Proceedings of ICALP 93, pp. 449-460, 1993.

[5] T. Okamoto and K. Ohta, "Disposable Zero-Knowledge Authentication and Their Applications to Untraceable Electronic Cash", Proceedings of Crypto 89, pp. 481-496, 1990.

[6] T. Okamoto and K. Ohta, "Universal Electronic Cash", Proceedings of Crypto 91, pp. 324-337, 1992.

[7] S. von Solm and D. Naccache, "On

Blind Signatures and Perfect Crimes", Computer and Security, pp. 581-583, 1992. 11.

[8] M. Stadler, J. Pivetau and J. Camenisch, "Fair Blind Signature", Proceedings of Eurocrypt 95, pp. 209-219, 1995.

[9] E. Fujisaki and T. Okamoto, "Practical Escrow Cash System", Pre-Proceedings of 1996 Cambridge Workshop on Secure Protocols, 1996.

[10] E. Brickell, P. Gemmell and D. Kravitz, "Trustee-based Tracing Extensions to Anonymous Cash and the Making of Anonymous Change", Proceedings of SODA 95, pp.457-466, 1995.

[11] J. Camenisch, J. Piveteau and M. Stadler, " Blind Signatures Based on the Discrete Logarithm Problem", Proceedings of Eurocrypt 94, pp. 428-432, 1994

[12] T. Okamoto, "An Efficient Divisible Electronic Cash Scheme", Proceedings of Crypto 95, pp.438-451, 1995.

[13] T. P. Pedersen, "Non-Interactive and Information-Theoretic Secure Verifiable Secret Sharing", Proceedings of Crypto 91, pp.129-140, 1992.

[14] D. Chaum, "Untraceble Electronic Mail, Return Address, and Digital Pseudonyms", Communications of the ACM, Vol.24, No.2, pp. 84-88, 1981.

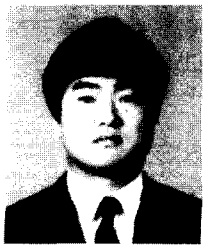
[15] C. S. Park, K. Itoh and K. Kurosawa, "Efficient Anonymous Channel and All/Nothing Election Scheme", Advances in Cryptology, Proceedings of Eurocrypt '93, pp. 248-259, 1993.

[16] T. ElGamal, "A Public Key Cryptosystem and Signature Scheme Based on Discrete Logarithm", IEEE Trans. Inform. Theory, Vol. 31, No. 4, pp. 469-472, 1985.

[17] D. Chaum, "Blind Signatures for Untraceable Payments", Advances in Cryptology, Proceedings of Crypto 82, Plenum Press, pp. 199-203, 1983.cha82

[18] S. Brands, " Untraceable Off-line Cash in Wallet with Observers", Advances in Cryptology, Proceedings of Crypto 93, pp.302-318, 1994.

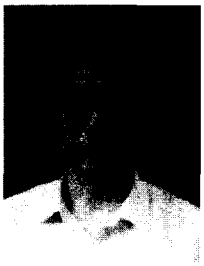
□ 著者紹介



김 춘 수

1987년 2월 숭실대학교 전기과 졸업(학사)
1989년 2월 숭실대학교 대학원 전기과 졸업(석사)
1998년 2월 숭실대학교 대학원 전기과 졸업(공학박사)
1990년 ~ 현재 한국전자통신연구원 선임연구원

※ 주관심분야 : 통신정보보호, 정보이론, 전자화폐



박 춘 식

광운대학교 전자통신과 졸업(학사)
한양대학교 대학원 전자통신과 졸업(석사)
일본 동경공업대학 전기전자공학과 졸업(암호학 전공, 공학박사)
1989년 10월 ~ 1990년 9월 일본 동경공업대학 객원 연구원
1989년 ~ 현재 한국전자통신연구원 책임연구원

※ 주관심분야 : 암호이론, 정보이론, 통신이론

전 희 종

1975년 2월 숭실대학교 전기공학과 졸업
1977년 2월 서울대학교 대학원 전기공학과 졸업(석사)
1987년 2월 중앙대학교 대학원 전기공학과 졸업(공학박사)
1977년 8월 ~ 1981년 8월 공군사관학교 교수
1995년 9월 ~ 1996년 5월 University of Victoria 객원교수
1998년 2월 ~ 1999년 2월 전력전자 학회 조사이사
1981년 8월 ~ 현재 숭실대학교 교수