해쉬체인을 이용한 익명의 판매자 전용화폐

최형섭[O†] 정명규[‡] 김상진[†] 오희국[†]
한양대학교 컴퓨터공학과[†], (주)아이캐시[‡]
{hschoi[O], sangjin, hkoh}@cse.hanyang.ac.kr[†], faker@icash.co.kr[‡]

Anonymous Vendor-specific Cash Using Hash Chain

Hyungsup Choi[O†] Myungkyu Jung[‡] Sangjin Kim[†] Heekuck Oh[†]
Dept. of Computer Science and Engineering, Hanyang University[†], iCash[‡]

요 약

해쉬체인을 이용한 전자화폐는 효율적인 해쉬함수를 이용하여 화폐의 유효성을 빠르고 저렴하게 확인할 수 있다. 해쉬체인을 이용한 기존 후불방식의 시스템은 근본적으로 익명성을 제공하지 않는 문제점이 있으며, 선불방식의 시스템들은 익명성을 제공하지만 이것으로 인하여 지불비용이 증가하여 소액지불에는 적합하지 않다. 이 논문에서는 이런 문제를 해결한 해쉬체인을 이용하는 새로운 익명의 판매자 전용화폐를 제안한다. 새 시스템은 해쉬체인과 표현문제를 이용하여 화폐를 구성하고 제한적 은닉서명을 사용하여 익명성을 제공한다.

1. 서 론

소액지불시스템(micropayment system)[1-8]은 뉴스, 신문, 잡지, 음악, 증권정보와 같은 서가의 디지털 상품을 거래할 때 적합한 지불 시스템이다. 많은 소액지불시스템이 지불 과정에서 화폐의 유효성을 확인하는 비용을 줄이기 위해 해쉬체인을 사용하여 화폐를 구성한다 [4-8]. 또한 트랜잭션의 처리 비용이 커지는 것을 피하기 위해 설계단계부터 익명을 고려하지 않거나 [1-3,6], 실명으로 하는 신용기반(credit-based)의 후불 시스템으로 구성하기도 한다 [4,5]. 익명 거래가 가능한 [7,8]은 해쉬체인을 이용한 범용화폐로 연산비용이 작지 않아 소액지불에는 부담스럽다. 그러나 거래의 규모가 아무리 작다 하더라도 대다수의 고객은 자신의 거래가 알려지기를 꺼려한다. 이것은 거래의 건전성과도 무관하다. 사이버 거래가 일반화되면 될수록 사생활 보호에 대한 고객의 관심과 요구는 더 높아질 것은 쉽게 예측할 수 있다. 이 논문에서는 고객의 사생활 보호에 중점을 두고, 값은 판매자와 잦은 거래를 하는 경우에 적합한 익명의 판매자 전용화폐를 제안한다. 새 시스템은 해쉬체인과 표현문제(representation problem)[9]를 사용하여 화폐를 구성하였으며, 익명성을 제공하기 위해 제한적 은닉서명(restrictive blind signature)[10]을 사용한다. 비용면에서는 인출과정을 제외하고는 기존의 후불방식의 판매자 전용화폐와 같은 수준을 갖는다.

2. 제안된 시스템

2.1 시스템 설정

이 논문에서 사용되는 수학 연산은 군(group)의 위수(order)가 매우 큰 소수 q인 G_q군에서 이루어진다. 이 군은 큰 소수 p를 선택하고 $p-1$의 소인수 중 하나인 q를 선택하여 구성한다. G_q군은 Z_p^*의 부분군(subgroup)으로 1을 제외한 모든 원소는 법 p에서 G_q군의 생성자가 된다. 시스템의 안전성은 이 군에서 이산대수(discrete logarithm)를 구하는 것이 계산적으로 어렵다는 것에 근거한다. 이 논문에서 지수요소와 관련된 연산은 법 q에서 이루어지고 나머지 군 연산은 모두 법 p에서 이루어진다. 이후, 논문에서는 법 p와 q에 대한 연산 표기를 생략한다. 이 시스템에서는 충돌회피 해쉬함수 $H:\{0,1\}^* \rightarrow Z_q$를 사용한다. 또한 임의의 값 x를 k번 해쉬함수에 적용하는 것을 $H^k(x)$로 표기한다.

시스템의 참여자는 은행, 고객, 판매자로 구성된다. 시스템을 설정하기 위해 은행은 표 1에 설명되어 있는 5개의 G_q군의 생성자(generator)를 임의로 선택한다. 그 다음 개인키 $x_B \in_R Z_q^*$를 선택하고 대응되는 공개키 $y_B = g_B^{x_B}$를 계산한다. 은행은 $p, q, g_B, g_r, g_v, g_K, g_L, y_B$를 공개한다. 은행은 또한 이중청구, 이중환불 등을 발견하기 위해 입금 데이터베이스를 유지한다.

이 시스템에서 사용하는 화폐는 표현문제를 이용하여 다음과 같이 구성한다.

$$C = g_K^{c_0} g_L^{n}$$

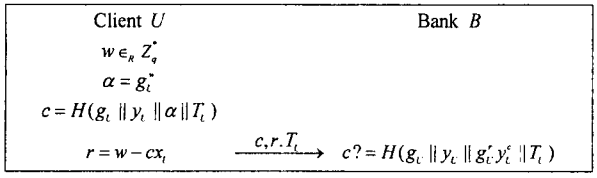
여기에서 c_0는 해쉬체인의 루트이고 n은 체인의 길이를 의미한다. 생성자 튜플 (g_K, g_L)에 대한 C의 표현은 (c_0, n)이다. 표현문제는 G_q에서 이산대수를 계산하는 것이 계산적으로 어렵다고 가정할 때 C의 표현을 찾는 것이 어렵다는 것에 바탕을 두고 있다.

2.2 계정 개설

고객은 자신의 비밀신원정보 $x_i \in_R Z_q^*$를 선택하고 대응되는 공개신원정보 $y_i = g_r^{x_i}$를 계산하여 은행에게 보내준다. 고객은 x_i를 알고 있음을 영지식으로 증명[11]하면 은행은 y_i를 고객의 식별자로 기록하고 계정을 개설한다. 영지식 증명은 그림 1과 같이 진행되며 프로토콜

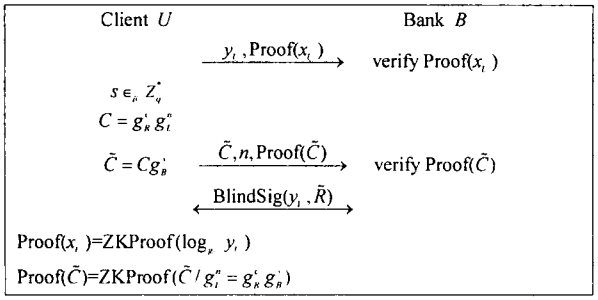
<표 1> 시스템 설정과 계정 설정에 필요한 파라미터

은행	g_B	은행의 공개키를 위한 생성자
	g_r	고객의 식별자를 위한 생성자
	g_v	판매자의 식별자를 위한 생성자
	g_K	해쉬체인의 루트를 표현하기 위한 생성자
	g_L	해쉬체인의 길이를 표현하기 위한 생성자
	$x_B \in_R Z_q^*$	은행의 개인키
	$y_B = g_B^{x_B}$	은행의 공개키
고객	$x_i \in_R Z_q^*$	고객의 개인키
	$y_i = g_r^{x_i}$	고객의 식별자
판매자	$x_v \in_R Z_q^*$	판매자의 개인키
	$g_v = g_v^{x_v}$	판매자의 식별자

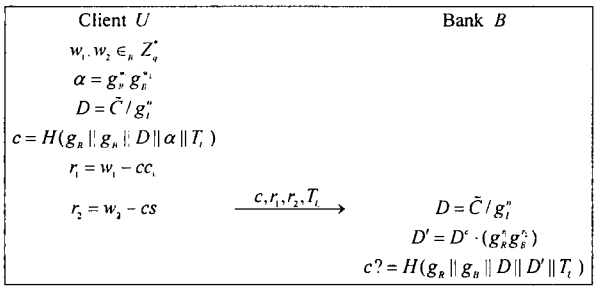


<그림 1> 이산대수 영지식 증명 프로토콜 $\mathrm{ZKProof}(\log_{g_t} y_t)$

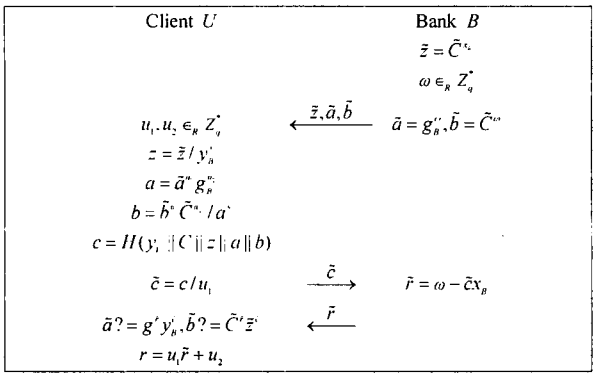
상에서는 $\mathrm{ZKProof}(\log_{g_t} y_t)$로 표기한다. 그림 1의 프로토콜에서 c는 $c = H(g_t \| y_t \| \alpha \| T_t)$와 같이 만들어지며, 이 때 T_t는 고객이 생성한 타임스탬프이다. 판매자도 $x_t \in_R Z_q^*$를 선택하여 $y_t = g_t^{x_t}$를 계산하고 y_t를 은행에게 보내어 계정을 개설한다. y_t는 판매자의 식별자를 나타내는 동시에 환불 과정에서는 공개키로도 사용된다.



<그림 2> 인출 프로토콜



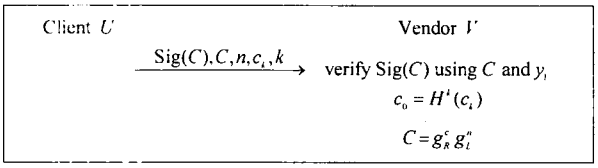
<그림 3> 표현 영지식 증명 프로토콜 $\mathrm{ZKProof}(\tilde{C} / g_t^n = g_R^s g_B^s)$



<그림 4> 제한적 은닉서명 BlindSig($y_t, \tilde{C}$)

2.3 인출 프로토콜

이 시스템의 인출 프로토콜은 그림 2와 같다. 고객은 n개의 동전을 인출하고자 할 경우 임의의 값 c_n을 선택하고 $c_{i-1} = H(c_i)$의 연산을 반복하여 해쉬체인 $\{c_0, c_1, \ldots, c_n\}$을 구한다. 여기서 c_0는 체인의 루트이며, 각 $c_i (1 \le i \le n)$가 하나의 동전을 나타낸다. 고객은 먼저 y_t를 은행에게 전달하고 x_t를 알고 있음을 영지식으로 증명한다. 신원을 입증한 다음 은닉요소 $s \in Z_q$를 임의로 선택하여 C를 은닉한다. 그 다음 그림 3의 프로토콜을 수행하여 $\tilde{C}$에 해쉬체인의 길이 정보 n이 포함되어 있음을 $\tilde{C}$ 표현의 다른 값은 보이지 않고 증명한다. 은행은 이 증명을 확인하고 고객과 그림 4의 제한적 은닉서명을 수행한다. 이 서명은 de Solages와 Traore[10]가 제안한 서명으로 BlindSig로 표기한다. 고객은 제한적 은닉서명의 은닉제한 특성 때문에 $\tilde{C}$가 결정된 후에는 $\tilde{C}, g_t^n$값으로만 서명을 받을 수 있다. y_t는 이 화폐를 판매자 전용으로 만들기 위해 포함하는 것이다. 그러나 은행은 이 값을 보지 못하므로 고객이 어느 판매자용 화폐를 인출하였는지 알 수 없다. 프로토콜이 성상적으로 종료되면 고객은 C에 대한 은행의 서명 $\mathrm{Sig}(C) = (z = C^{x_B}, c, r)$을 얻으며, 이 서명 값은 $c = H(y_t \| C \| z \| g_R^r y_B^c \| C' z^c)$을 만족한다.



<그림 5> 지불 프로토콜

2.4 지불 프로토콜

고객이 C를 판매자에게 처음 지불할 경우에는 그림 5의 프로토콜을 수행한다. 이 그림에서 고객은 k개의 동전을 지불한다. 이를 위해 고객은 Sig(C), C, n, c_k, k를 판매자에게 전달한다. 판매자는 C에 대한 은행의 서명을 확인하고, 이 서명에 자신의 식별자 y_t가 포함되어 있는지를 확인한다. 판매자는 c_k를 k번 해쉬함수에 적용하여 c_0를 구한 후 C와 $g_R^s g_t^n$가 같은지 비교하여 화폐의 유효성을 확인한다. 판매자는 고객의 이중사용 검출을 위해 Sig(C), C, n, c_k, k를 자신의 데이터베이스에 저장한다. 그림 5의 지불 프로토콜은 고객이 판매자에게 처음으로 지불할 경우에 사용하는 프로토콜이다. 같은 판매자에게 C에 있는 나머지 동전을 지불할 경우에는 C, c_i, i만 전달하여 지불한다.

2.5 입금 프로토콜

판매자는 은행에게 자신의 식별자 y_t와 함께 고객으로부터 받은 Sig(C), C, n, c_i, i를 은행에게 전달하여 지불대금을 청구한다. 여기서 c_i는 고객으로부터 받은 마지막 동전이다. 은행은 Sig(C)을 확인하고 이 화폐가 판매자 y_t 전용화폐인지 검사한다. 그 다음 c_i가 이 화폐에 포함된 동전이 맞는지 해쉬체인을 이용하여 확인한다. 그 다음 C가 입금 데이터베이스에 있는지 검사하여 이중청구를 확인한다. 은행은 입금된 정보가 정당하다면 해당 금액을 판매자의 계좌에 입금시키고 받은 정보는 입금 데이터베이스에 저장한다. 해쉬체인의 특성 때문에 판매자는 고객으로부터 받은 동전 외에는 입금을 할 수 없다. 판

매자는 입금을 한 후에도 고객의 이중사용을 확인하기 위해 C, c_i, i는 계속 유지한다. 판매자 전용화폐이므로 고객의 이중사용은 판매자가 확인하며, 판매자의 이중청구는 은행이 확인한다.

2.6 환불 프로토콜

선불방식이므로 사용하고 남은 동전은 은행에 반납하여 환불받을 수 있어야 한다. 그러나 해쉬체인으로 연결되어 있으므로 단순하게 반납하면 익명성이 깨진다. 고객은 판매자에게 잔액확인서를 받아 이것을 은행에 전달하고 [12]에 제시된 기법을 이용하여 환불티켓을 은행으로부터 받는다. 실제 환불은 이 티켓을 이용하여 나중에 받는다. 환불티켓은 익명으로 발행되므로 고객은 미리 환불을 받은 후에 그것을 지불에 다시 사용할 수가 있다. 이것을 막기 위해 판매자로부터 잔액확인서를 받아 환불받을 때 제출하여야 한다. 이 확인서는 오직 상점만이 만들 수 있어야 하며, 고객은 이것을 조작할 수 없어야 한다.

3. 시스템의 안전성

3.1 신분위장

공격자가 다른 어떤 고객을 행세하여 화폐를 인출하기 위해서는 먼저 그 고객으로 신원을 증명할 수 있어야 한다. 이 증명을 하기 위해서는 기저 g_i에 대한 그 고객의 공개신원정보의 이산대수를 알아야 하지만 이것은 계산적으로 어렵다. 또한 고객은 신원을 증명할 때 시간 정보를 포함하므로 공격자는 기존 증명을 다시 사용할 수 없다.

3.2 화폐위조

이산대수 가정에 의해 은행을 제외한 누구도 은행의 서명키를 계산하기 어려우므로 화폐를 직접 위조하는 것은 어렵다. 또한 de Solages와 Traore[10]의 제한적 은닉서명 가정 1에 의해 서명을 여러 번 병행 또는 순차적으로 수행하여도 원래 받아야 하는 것 이상의 서명을 얻을 수 없으므로 이런 방식으로 화폐를 위조하는 것 역시 어렵다.

3.3 화폐조작

인출단계에서 고객은 해쉬체인의 길이 n을 속일 수 없어야 한다. 고객이 이것을 속일 수 있는 길은 은닉서명을 수행하기 전에 전달하는 증명을 공격하거나 은닉서명 자체를 공격하여야 한다. 전자는 증명에 사용하는 해쉬함수의 일방향성을 깨야하기 때문에 어렵고, 후자는 사용하는 은닉서명의 가정 2에 의해 기저 g_n에 대한 g_i의 이산대수를 계산할 수 있어야지만 가능하다. 그러나 이산대수 가정에 의해 이것은 어려우므로 후자의 공격도 가능하지 않다.

3.4 익명성

고객은 은행으로부터 은닉서명으로 화폐를 인출하거나 환불티켓을 받는다. de Solages와 Traore[10]의 제한적 은닉서명 정리 2에 의해 은행은 서명과정에서 얻은 정보로부터 서명한 메시지나 결과 서명에 대한 어떤 정보도 얻을 수 없다. 뿐만 아니라 서명에 포함하는 y_i를 인출 과정에서 보지 못하므로 어느 판매자용 화폐를 인출했는지 알 수 없다. 또한 환불을 하여도 환불티켓을 이용하므로 고객의 익명성은 유지된다.

3.5 이중사용 및 이중청구

이 시스템은 판매자 전용화폐로 하나의 해쉬체인은 한 판매자에게만 지불할 수 있다. 그러므로 판매자는 사전에 고객의 이중사용을 막을 수 있다. 해쉬체인의 특성 때문에 판매자는 고객으로부터 받은 동전만 입금할 수 있으며, 은행은 입금된 정보를 기록하고 있으므로 판매자는 이중청구를 할 수도 없다.

3.6 이중환불 및 초과환불

은행은 환불된 티켓을 보관하므로 이중으로 환불하는 것은 쉽게 발각된다. 고객은 환불티켓을 인출하기 전에 잔액확인서를 판매자로부터 받아야 한다. 이 확인서는 판매자가 서명하여 만듦으로 고객이 자신에게 유리하도록 변경할 수 없다. 따라서 고객은 자신이 받아야 하는 것 이상으로 환불을 받을 수 없다. 또한 잔액확인서가 필요하므로 미리 환불을 받은 후에 그것을 다시 사용할 수 없다. 판매자는 잔액확인서를 발급한 후에는 그 체인에 있는 동전은 받지 않는다.

4. 결 론

이 논문에서는 해쉬체인을 이용한 익명이 보장되는 판매자 전용화폐를 제안하였다. 이 시스템은 표현문제를 사용하여 화폐를 구성하였고, 제한적 은닉서명을 사용하여 익명성을 제공하였다. 또한 남은 동전을 이미 사용한 동전의 익명성을 해치지 않고 환불받을 수 있도록 하였다. 제안된 시스템은 기존의 해쉬체인에 기반한 판매자 전용 화폐의 효율성을 그대로 유지하면서 익명성이 제공되는 시스템이다.

참고문헌

[1] M.S. Manasse, "The Millicent Protocols for Electronic Commerce," *Proc. of the 1st USENIX Workshop on Electronic Commerce*, pp. 117-123, Jul. 1995.

[2] A. Herzberg and H. Yochai, "Mini-Pay: Charging per Click on the Web," *Proc. of the 6th Int. World Wide Web Conf.*, Apr. 1997.

[3] C. Jutla and M. Yung, "PayTree: Amortized-Signature for Flexible MicroPayments," *Proc. of the 2nd USENIX Workshop on Electronic Commerce*, pp. 213-221, Nov. 1996.

[4] R.L. Rivest and A. Shamir, "PayWord and MicroMint - Two Simple Micropayment Schemes," *Proc. of 1996 Int. Workshop on Security Protocols*, LNCS 1189, pp. 69-87, Apr. 1996.

[5] Y. Mu, V. Varadharajan, and Y. Lin "New Micropayment Schemes Based on PayWords," *Proc. of the 2nd ACISP*, LNCS 1270, pp. 283-293, Jul. 1997.

[6] K.Q. Nguyen, Y. Mu, and V. Varadharajan, "Micro-Digital Money for Electronic Commerce," *Proc. of the 13th IEEE ACSAC*, pp. 2-8, Dec. 1997.

[7] W. Mao, "Lightweight Micro-Cash for the Internet," *Proc. of the ESORICS'96*, LNCS 1146, pp. 15-32, Sep. 1996.

[8] K.Q. Nguyen, Y.Mu, and V. Varadharajan, "Secure and Efficient Digital Coins," *Proc. of the 13th IEEE ACSAC*, pp. 9-15, Dec. 1997.

[9] S. Brands, "Untraceable Off-line Cash in Wallets with Observers," Crypto'93, LNCS 773, pp. 302-318, Aug. 1993.

[10] A. de Solages and J. Traore, "An Efficient Fair Off-Line Electronic Cash System with Extensions to Checks and Wallets with Observers," *Proc. of the 2nd Int. Conf. on Financial Cryptography*, LNCS 1465, pp. 275-295, Feb. 1998.

[11] C.P. Schnorr, "Efficient Signature Generation by Smart Cards," *J. of Cryptology*, Vol. 4, No. 3, pp. 161-174, 1991.

[12] 최형섭, 김상진, 오희국 "분할 가능한 화폐를 위한 새로운 환불방식," 한국정보보호학회 2001년도 종합학술발표회, pp. 177-180, 2001년 11월