이중해쉬체인에 기반한 분할 가능 전자화폐의 설계
(Design of Divisible Electronic Cash based on Double Hash Chain)

용 승 림 † 이 은 경 †† 이 상 호 †††
(Seung-Lim Yong) (Eun-Kyoung Lee) (Sang-Ho Lee)

요 약 전자화폐는 안전성이 제공되어야 하고 이중사용이 방지되어야 하며, 사용자의 편의를 위해 분할성을 만족하는 것이 바람직하다. 분할성은 사용자가 발급받은 전자화폐를 화폐의 총액을 넘지 않는 범위 내에서 사용자가 원하는 대로 나누어 사용할 수 있는 성질이다. 분할성을 만족함으로써 거스름돈의 발생을 줄일 수 있고 여러 번의 인출과정을 수행하지 않아도 되는 장점이 있다.

본 논문에서는 이중해쉬체인에 기반한 분할 가능한 전자화폐 프로토콜에 대하여 제안한다. 전자화폐는 이중해쉬체인에 의해 서로 다른 액면금액을 가진 동전으로 구성된다. 전자화폐의 분할성은 지불인증을 이용하여 만족된다. 지불인증이란 은행으로부터 받은 은행의 대리서명 키 쌍으로서, 높은 액면금액의 동전을 낮은 액면금액의 동전으로 분할할 때 은행대신 서명을 하여 동전의 정당성을 인증 받을 수 있다. 제안된 방법은 사용자의 익명성을 제공하지는 않지만 해쉬함수를 이용하여 수행속도가 빠르고 위조 불가능한 동전을 생성하며, 분할성을 만족함으로써 사용자가 편리하게 이용할 수 있는 장점이 있다.

키워드 : 전자화폐, 분할성, 이중해쉬체인, 지불인증, 대리서명

Abstract An electronic cash system has to provide the security, to prevent the double spending and to support the divisibility of electronic cash for the easy of use. Divisible electronic cash system allows an electronic cash to be divided into subdivisions. Each subdivision is worth any desired value, but all values must add up to the original cash value. Divisible scheme brings some advantages. It reduces to make the change and also there is no necessity that a customer must withdraw a cash of the desired value whenever transactions occur.

In this paper, we present an electronic cash protocol which provides the divisibility based on the double hash chain technique. Electronic cash is constructed in the form of coins. Coins, generated by the double hash chain, have different denominations. The divisibility of an electronic cash is satisfied by the payment certificate, which is a pair of bank's proxy signature received from the bank. When a customer pays the coin of subdivision, the fairness of that coin is certified by a customer's signing instead of a bank. Although the proposed method does not guarantee user's anonymity, it generates coins which cannot be forged, and the customer can use an electronic cash conveniently and efficiently with its divisibility.

Key words : electronic cash, divisibility, double hash chain, payment certificate, proxy signature

1. 서 론

전자상거래란 실세계에서 이루어지는 상거래를 통신매체를 이용하여 가상의 공간에서 이루어지게 하는 상행위이다. 컴퓨터와 통신망을 이용한 전자상거래에서는 상품의 구입과 지불의 시점이 다르기 때문에 발생하는 동시성의 결여와 비대면(非對面) 거래로 인한 상대방에의 신뢰성 결여로 기존의 지불 행위와 같이 안전하고 편리하게 지불을 수행하기가 어렵다. 이러한 환경에서 안전하게 사용할 수 있는 지불 방법이 전자화폐이다.

전자화폐란 액면가치를 보증하기 위해 은행이 서명한 디지털 신호로 표현된 가치정보이다[1]. 전자화폐는 기존의 화폐가 가져야 하는 법적인 효력과 안전성 등의 기능을 그대로 가지면서 별도의 기기나 또는 컴퓨터 등

† 학생회원 : 이화여자대학교 컴퓨터학과
dragon@ewha.ac.kr
†† 비 회 원 : 이화여자대학교 컴퓨터학과
eye@sicc.co.kr
††† 종신회원 : 이화여자대학교 컴퓨터학과 교수
: shlee@ewha.ac.kr
논문접수 2002년 11월 18일
심사완료 2003년 5월 19일

에 소프트웨어 형태로 존재하는 전자지갑에 의해 관리된다. 전자화폐는 실물화폐가 가지고 있는 기능뿐만 아니라 전자화폐가 디지털 데이타로 구성되어 있음으로 발생되는 문제점을 해결할 수 있는 기능들이 요구된다. 전자화폐는 디지털 데이타인 가치정보의 위조가 불가능하도록 하는 안전성(security)을 제공해야 하고, 디지털 정보로 표시되는 화폐가 한번 이상 복사되어 사용되는 이중사용(double spending)이 방지되어야 한다. 전자화폐를 사용하는 사용자의 익명성이 보장되어야 하고 사용자 편리성이 만족되어야 한다. 또한 발급받은 전자화폐를 사용자 마음대로 나누어 사용할 수 있는 분할성(divisibility)을 제공하여 사용자 편리성과 효율성을 증대시킬 수 있다. 이러한 여러 전자화폐의 요구조건 중에서 분할성은 사용자가 전자화폐를 발급받는 경우 발급받은 전자화폐를 사용자가 원하는 대로 나누어 사용할 수 있는 성질이다. 즉 사용자가 보유하고 있는 전자화폐의 총액을 초과하지 않는 범위 내에서 사용자가 전자화폐를 나누어 사용할 수 있음을 말한다. 전자화폐 시스템이 분할성을 만족함으로써 거스름 발생에 대한 대비가 필요하지 않아 효율성을 높일 수 있으며 작은 금액에 대한 화폐를 재발행해야 하는 부담을 감소시킬 수 있는 장점이 있다.

본 논문에서는 이중 해쉬함수를 이용하여 동전을 생성하고 지불인증을 이용하여 생성된 동전을 마음대로 분할하여 사용할 수 있는 방법을 제안한다. 사용자의 익명성을 제공하지는 않지만 해쉬함수를 이용하여 전자화폐를 생성하기 때문에 해쉬함수의 일방향성에 기반하여 위조 불가능한 동전을 생성하며, 동전의 루트값에만 은행의 서명을 받음으로써 계산상 효율적이다. 또한 분할성을 만족함으로써 편리하게 이용할 수 있는 장점이 있다.

2. 관련 연구

2.1 해쉬체인에 기반한 전자화폐 시스템

전자화폐는 화폐의 정당성을 인증하기 위하여 공개키 전자서명 방식을 이용하여 동전마다 은행이 서명을 한다. 그러나 공개키 전자서명 방식은 계산상 매우 복잡하기 때문에 생성된 동전마다 공개키 전자서명을 붙이기에는 비효율적이다.

1996년 Ron Rivest와 Adi Shamir는 해쉬함수를 이용하여 동전을 체인형식으로 구성하고 체인의 루트값 하나에만 서명을 받음으로써 생성된 동전마다 서명을 받지 않도록 하여 효율성을 향상시킨 Payword 전자화폐를 제안하였다[2]. Payword 전자화폐에서 사용자는 동전을 인출할 때 임의의 수 w_n을 선택하고 $w_i = h(w_{i+1})$ $(i = n-1, n-2, \ldots, 0)$의 식을 적용하여 역방향으로 동전 $w_1, w_2, \ldots, w_n$을 생성하고 해쉬체인의 루트값 w_0에만 은행의 서명을 받는다. 사용자는 서명된 w_0를 상점에 보내고 w_1부터 지불금액만큼 동전을 지불한다. 상점은 w_0의 서명을 확인하고, 지불된 동전의 인덱스만큼 해쉬함수를 적용하여 루트값이 w_0가 되는지 확인함으로써 동전이 정당한지 검증한다.

이중해쉬체인 전자화폐는 Payword에서 제안한 동전 구성 방법에 의해 두 개의 해쉬체인을 생성하고 두 개의 해쉬체인의 원소 한 쌍을 하나의 동전을 구성하는데 이용한다[3,4,5]. 즉, [그림 1]과 같이 해쉬체인을 두 개 생성한 후 각 체인의 요소들을 서로 역순으로 번호가 같은 것끼리 쌍을 이루어 동전을 구성한다. 이 방식은 동전을 구성하는 한 체인의 종자값(seed)을 알아냈다 하더라도 다른 체인의 종자값도 알아야만 위조가 가능하다는 특징을 이용하여 동전의 위조에 대한 안전성을 향상시켰다. 해쉬함수를 기반으로 하는 전자화폐 시스템의 안전성은 해쉬함수가 역방향으로 계산하기 어렵다는 해쉬함수의 일방향성에 기반한다.

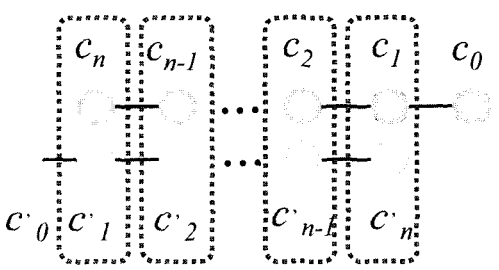

그림 1 이중해쉬체인 동전

2.2 분할 가능한 전자화폐 시스템

분할 가능한 전자화폐 시스템이란 동전의 분할성을 만족하는 전자화폐이다. 분할성은 사용자가 전자화폐를 발급 받는 경우 사용자가 보유하고 있는 전자화폐의 총액을 초과하지 않는 범위 내에서 사용자가 전자화폐를 나누어 사용할 수 있는 성질이다.

1991년 T. Okamoto 등은 이진트리 구조를 이용한 분할 가능한 효율적인 전자화폐 프로토콜을 제안하였다[6,7]. 이진 트리 구조에서 트리의 각 노드는 동전의 액면가를 나타내며 트리의 자식 노드의 금액의 합이 부모 노드의 금액이 되는 방식으로 각 노드별로 금액을 준다. [그림 2]에서 루트노드가 w원의 액면금액을 가지면 그 다음 레벨의 노드들은 $w/2$ 원의 액면금액을

갖는다.

대부분의 분할 사용 가능한 전자화폐 프로토콜에서는 이진 트리를 이용한 접근 방식을 이용하여 화폐의 분할 사용기능을 구현하고 있다. 그러나 이 방식은 계산량이 트리의 깊이(depth)에 따라 변하고 복잡한 수학식을 사용하며, 법(mod) 연산 방식을 사용함으로써 계산량이 많아지는 단점이 있다[8].

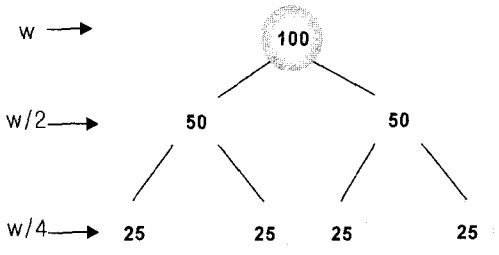

그림 2 이진 트리를 이용한 분할 가능 전자화폐

3. 이중해쉬체인에 기반한 분할 가능 전자화폐

본 장에서는 해쉬체인에 기반하여 다양한 액면금액을 가지는 분할 가능 전자화폐의 프로토콜에 대하여 기술한다. 본 논문에서 제안한 분할 가능 전자화폐는 동전을 구성할 때 Payword와 같이 해쉬함수를 이용한다. 해쉬함수를 이용하여 해쉬체인을 구성하고 생성된 해쉬체인 두 개를 묶어 서로 다른 액면금액을 가지는 이중해쉬체인을 여러 개 구성하고 이중해쉬체인의 원소 한 쌍으로 동전을 구성한다.

사용자가 동전을 분할하여 지불에 이용하고자 할 때는 은행으로부터 동전 분할에 대한 권한을 지불인증을 통하여 부여받은 후, 동전을 분할하여 지불할 때 지불인증을 이용하여 지불을 수행한다. 지불인증은 인출프로토콜에서 사용자가 은행으로부터 미리 받아둔 분할하고자 하는 동전에 대한 대리서명 키로서, 동전을 분할할 때 분할하여 새로 생성한 동전에 대리서명 키를 이용하여 서명을 함으로써 동전의 정당성을 입증 받는다. 상세한 전자화폐의 프로토콜은 다음절에서 기술한다.

3.1 용어 정의

이 절에서는 앞으로 사용하게 될 기호들을 정의하며, 이후에는 이들 기호의 부가적인 정의나 설명 없이 사용한다.

- C : 사용자
- B : 은행
- V : 상점
- PK_X , SK_X: X의 공개키, PK_X와 쌍을 이루는 X의 비밀키
- $Cert_X$: 신뢰할 수 있는 인증기관이 발행한 X의 공개키에 대한 인증서
- PPK, PSK : 지불인증으로 이용되는 대리서명의 공개키, 대리서명의 비밀키
- $\langle\ \rangle_{PK_X}$: X의 공개키로 암호화하는 함수
- $\langle\ \rangle_{SK_X}$: X의 비밀키로 복호화하는 함수
- $S(\ \ldots, SK_X)$: 서명함수
- $V(\ \ldots, PK_X)$: 검증함수
- $h(\ )$: 일방향 해쉬함수
- $h^n(\ )$: 해쉬함수를 n번 적용
- PR : 구매내역(Purchase Request)

3.2 프로토콜

전자화폐는 사용자, 은행 그리고 상점 등 세 개의 구성원으로 이루어지며 사용자와 은행사이의 인출 프로토콜, 사용자와 상점간의 지불 프로토콜 그리고 상점과 은행간의 예치 프로토콜로 구성된다.

3.2.1 인출 프로토콜

인출 프로토콜은 동전의 생성과정과 지불인증의 부여과정으로 나뉜다. 동전의 생성과정에서 사용자는 동전을 생성하고 동전의 루트값에 은행으로부터 서명을 받는다. 그리고 지불인증의 부여과정에서 동전을 분할하여 사용할 경우 필요한 지불인증으로써 은행의 대리서명 키 쌍을 부여받는다. 은행은 대리서명 키와 생성된 동전의 일부 정보, 그리고 사용자의 정보를 함께 저장해 둔다. 동전의 생성과정과 지불인증의 부여과정은 다음에 상세히 기술한다.

■ 동전 생성 과정

이중해쉬체인에 기반한 전자화폐는 [그림 3]과 같이 이중해쉬체인이 여러 개로 구성된 형태로서, 이중해쉬체인 각각마다 번호를 가지고 있고 이 번호에 따라 액면금액을 다르게 설정한다. 해쉬체인의 번호가 j인 체인내의 모든 동전은 10^{j-1}원의 액면가치를 가진다. 예를 들어 해쉬체인의 번호가 $j=1$이면 체인의 동전들의 액면금액은 1원, $j=2$이면 액면금액은 10원, $j=3$이면 100원의 액면가치를 갖는다.

사용자가 은행으로부터 인출을 요구하면 은행은 사용자 계좌의 잔액을 확인한다. 잔액이 요구액보다 많은 경우 인출프로토콜이 시작된다. 이중해쉬체인을 생성할 때 이중해쉬체인의 한쪽 체인은 사용자가 생성하고 다른

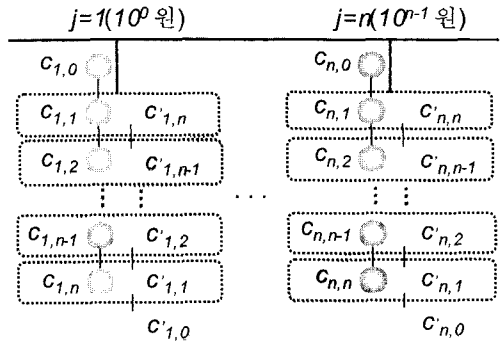

그림 3 이중해쉬체인 이용한 동전의 구성

한쪽 체인은 은행이 생성한다. 동전은 하나의 이중해쉬체인 내에서 역순으로 번호가 같은 해쉬함수 값 $(c_{j,i}, c'_{j,n-i+1})$이 쌍을 이루어 구성된다. 사용자는 종자값 $c_{j,n}$을 임의로 선택하고 해쉬함수를 적용하여 이중해쉬체인의 한쪽 체인을 생성한다. 은행도 종자값 $c'_{j,n}$을 임의로 선택하고 해쉬함수를 적용하여 다른 한쪽의 체인을 생성한다. 상세한 프로토콜은 다음과 같다.

1) 사용자가 은행에 인출을 요구하면 은행은 사용자의 계좌에서 잔액을 확인한다. 잔액이 요구액보다 많으면 인출프로토콜을 수행하고 그렇지 않으면 프로토콜을 종료한다.

2) 사용자는 해쉬체인의 번호 j에 따라 액면금액이 다른 이중해쉬체인을 생성한다. 각 해쉬체인의 한쪽 체인을 구성하기 위하여 임의의 수 $c_{j,n}$을 선택한다. 선택한 $c_{j,n}$에 해쉬함수를 적용하여 식 (1)과 같이 이중해쉬체인의 한쪽 체인을 생성한다. 해쉬체인의 루트값은 $c_{j,n}$에 해쉬함수를 n번 적용하여 $c_{j,0} = h^n(c_{j,n})$를 생성한다.

$$c_{j,i} = h(c_{j,i+1}) \ (i = n-1, n-2, \cdots, 0) \quad (1)$$

3) 은행은 이중해쉬체인의 다른 한쪽 체인을 생성하기 위하여 임의의 수 $c'_{j,n}$을 선택한 후, 해쉬함수를 적용하여 식 (2)와 같이 체인을 생성하고 생성된 $c'_{j,n}$에 n번 해쉬함수를 적용하여 $c'_{j,0} = h^n(c'_{j,n})$를 계산해 놓는다. 이렇게 생성된 해쉬체인의 종자값을 사용자의 공개키로 암호화하여 사용자에게 보내준다.

$$c'_{j,i} = h(c'_{j,i+1}) \ (i = n-1, n-2, \cdots, 0) \quad (2)$$

4) 사용자는 자신이 생성한 체인의 루트값을 은행의 공개키로 암호화한 후 자신의 비밀키로 서명하여 com_B_j를 만든다. 은행으로부터 받은 체인의 루트값 $c'_{j,0}$과 자신이 생성한 체인의 루트값 $c_{j,0}$을 이용하여 $Root_j$를 계산하여 com_B_j와 함께 은행에 보낸다.

5) 은행은 사용자에게 받은 $Root_j$에 서명한 값 $R_j = S(n \cdot Root_j, SK_B)$를 사용자에게 보낸다.

사용자가 생성된 동전에 은행으로부터 서명을 받은

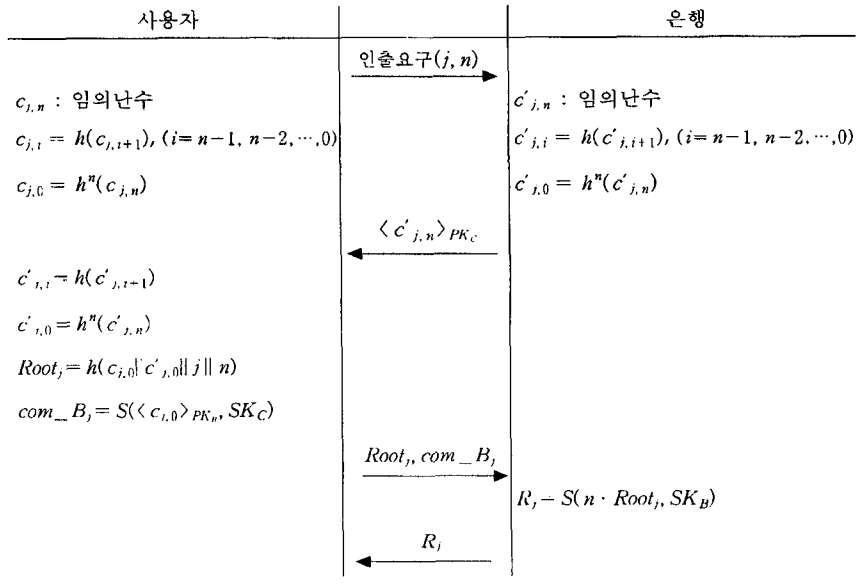

그림 4 인출프로토콜의 동전생성 과정

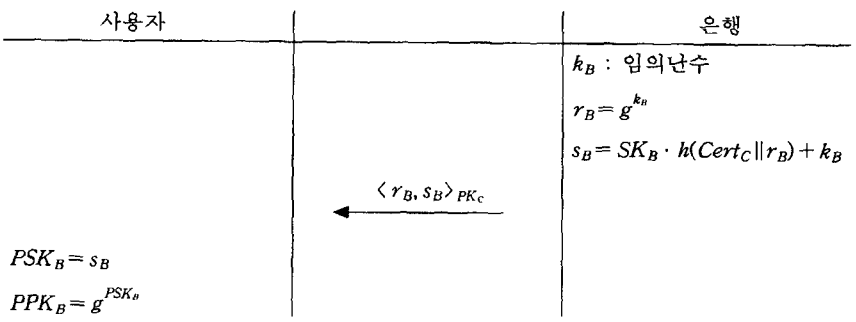

그림 5 인출프로토콜의 지불인증 부여과정

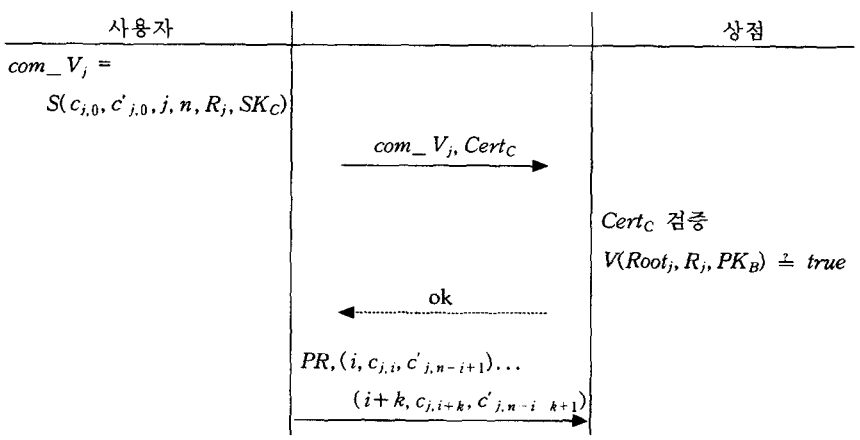

그림 6 분할하지 않고 지불하는 지불 프로토콜

후에는 분할하여 지불할 동전의 지불인증을 부여받는다. 지불인증 부여과정은 다음과 같다.

■ 지불인증 부여과정

지불인증은 동전을 분할할 때 분할하여 새로 생성한 동전에 은행대신 사용자가 은행의 서명을 할 수 있도록 은행의 대리서명 키 쌍을 부여한다[9]. 사용자는 대리서명 키 쌍을 이용하여 분할한 새 동전에 서명을 함으로써 동전에 대한 정당성을 입증 받는다. 즉 분할 가능성은 전자화폐를 인출할 당시에 그 동전보다 한 단계 낮은 액면금액에 대하여 은행으로부터 대리서명을 할 수 있는 정보를 제공받아 이루어진다. 은행은 지불인증을 부여하고 지불인증에 대한 사용자의 정보를 저장하여 놓는다.

1) 은행은 임의난수 k_B를 선택하여 $r_B = g^{k_B}$를 계산하고 $s_B = SK_B \cdot h(Cert_C \| r_B) + k_B$를 계산한다. 은행은 생성한 r_B, s_B를 사용자의 공개키로 암호화한 값 $\langle r_B, s_B \rangle_{PK}$를 사용자에게 보낸다.

2) 사용자는 지불인증으로 대리서명키 $PSK_B = s_B$, $PPK_B = g^{PSK_B}$를 얻고 이를 이용하여 상점에 동전을 분할하여 지불할 때 분할한 동전에 서명을 수행한다.

3.2.2 지불 프로토콜

사용자가 상점으로부터 물건을 구입하고 전자화폐를 지불할 때 지불금액에 맞도록 해당하는 금액의 동전들이 모두 있는 경우는 동전을 분할하지 않고 지불한다. 그러나 필요로 하는 액면금액의 동전이 없을 때에는 동전을 분할하고 인출시 부여받았던 지불인증을 이용하여 동전에 서명을 수행한 후 지불한다. 지불 프로토콜은 동전을 분할하지 않고 지불할 때와 분할하여 지불할 때의 경우를 나누어 기술한다.

■ 분할하지 않고 지불할 경우

동전을 분할하지 않고 지불할 경우의 지불 프로토콜은 [그림 6]과 같다. 사용자는 이중해쉬체인에서 $(c_{j,1}, c'_{j,n})$ 동전부터 지불금액에 해당하는 해쉬체인의 개수만큼 지불에 사용한다. [그림 6]의 프로토콜은 이전

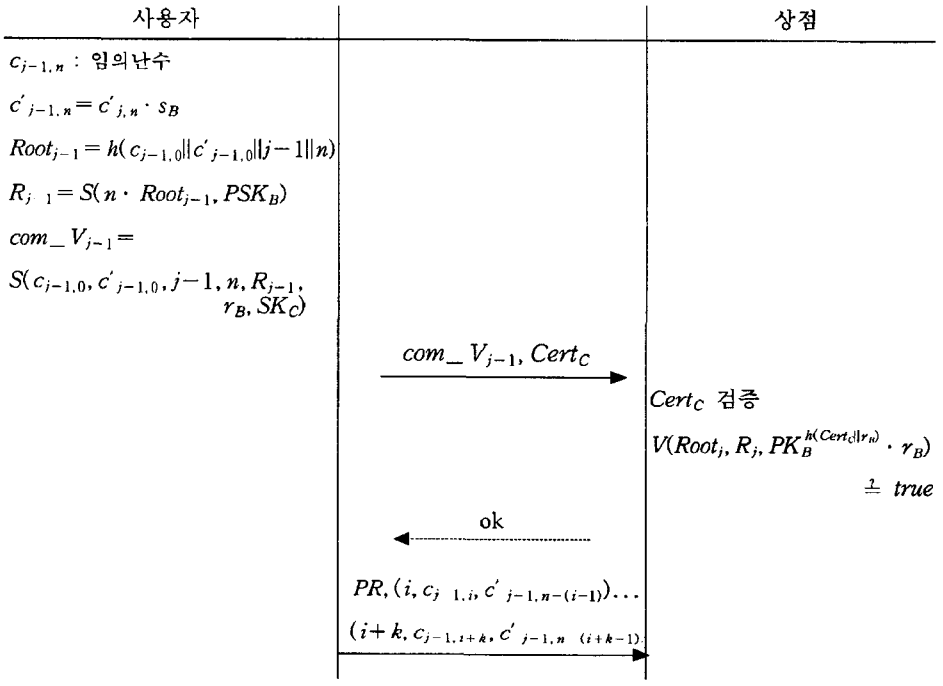

그림 7 분할하여 지불하는 지불프로토콜

에 지불했던 동전이 $i-1$번까지이고 k개의 동전을 지불할 경우이다.

1) 사용자는 지불하고자 하는 금액의 이중해쉬체인의 루트값과 이들 해쉬체인에 은행으로부터 서명받은 R_j값을 자신의 비밀키로 서명한 com_V_j값과 함께 인증서를 상점에게 보낸다.

2) 상점은 인증서 $Cert_C$를 검증하고 은행의 공개키를 이용하여 R_j를 검증한다.

3) 상점의 확인이 끝나면, 사용자는 구매 내역 PR과 함께 상품의 금액의 합계에 맞도록 동전을 지불한다. 예를 들어 $k \cdot 10^{j-1}$원의 금액을 지불해야 하고 이전에 $(i-1)$번 동전까지 이용했다면, j번째 이중해쉬체인에서 이전에 사용했던 동전 다음의 동전부터 k개만큼의 동전을 상점에 지불하면 된다.

$$PR, (i, c_{j,i}, c'_{j,n-(i-1)}) \ldots (i+k, c_{j,i+k}, c'_{j,n-(i+k-1)})$$

4) 상점은 받은 동전에 $h^{i-k}(c_{j,i-k}) = c_{j,0}$, $h^{n-(i+k-1)}(c'_{j,n-(i+k-1)}) = c'_{j,0}$과 같이 해쉬함수를 적용하여 동전의 유효성을 확인한다.

■ 분할하여 지불할 경우

액면금액이 낮은 동전이 부족할 경우 액면금액이 높은 동전을 분할하여 지불한다. 이때에는 은행으로부터 미리 받아둔 지불인증을 이용하여 동전에 서명을 수행하고 이들 동전을 지불에 이용한다.

1) 사용자는 $c'_{j-1,n} = c'_{j,n} \cdot s_B$와 임의의 수 $c_{j-1,n}$을 선택하고 이들을 n번 해쉬함수를 적용하여 다음을 생성한다.

$$c_{j-1,0} = h^n(c_{j-1,n}),\ c'_{j-1,0} = h^n(c'_{j-1,n})$$

2) 사용자는 인출프로토콜에서 부여받은 은행의 대리서명 키를 이용하여 새로운 동전에 서명을 수행한다.

$Root_{j-1} = h(c_{j-1,0} || c'_{j-1,0} || j-1 || n)$

$R_{j-1} = S(n \cdot Root_{j-1}, PSK_B)$

3) 사용자는 상점에게 아래의 항목들을 보낸다.

$S(c_{j-1,0}, c'_{j-1,0}, j-1, n, R_{j-1}, r_B, SK_C)$, $Cert_C$

4) 상점은 $Cert_C$를 확인하고 R_{j-1}을 은행의 공개키를 이용하여 검증한다.

$$V(Root_{j-1}, R_{j-1}, PK_B^{h(Cert_C||r_B)} \cdot r_B) \stackrel{?}{=} true$$

5) 상점의 확인이 끝나면, 사용자는 구매 내역과 상품의 금액의 합계에 맞도록 동전을 지불한다.

$$PR, (i, c_{j-1,i}, c'_{j-1,n-(i-1)}) \cdots (i+k, c_{j-1,i+k}, c'_{j-1,n-(i+k-1)})$$

6) 상점은 사용자의 동전들을 확인하고 구매내역과

사용자의 동전을 저장해둔다.

3.2.3 예치 프로토콜

상점은 하루단위로 사용자로부터 받은 동전들을 은행에 예금한다. j번째 체인의 i번째 동전부터 k개의 동전을 예치할 경우 상점은 사용자로부터 받은 동전들과 사용자로부터 받은 com_V_j의 값을 은행에게 넘겨준다.

$com\_V_j = S(c_{j,0}, c'_{j,0}, j, n, R_j, SK_C)$ 또는

$S(c_{j,0}, c'_{j,0}, j, n, R_j, r_B, SK_C)$

그리고 동전의 처음과 끝 $(i, c_{j,i}, c'_{j,n-(i-1)})$,

$(i+k, c_{j,i+k}, c'_{j,n-(i+k-1)})$

분할한 동전을 받은 경우에는 동전과 함께 지불인증을 은행에 넘겨준다. 은행은 상점으로부터 받은 지불인증값 r_B를 이용하여 지불인증을 부여한 사용자를 찾아낸다. 다음으로 r_B의 임의난수 x_B값을 찾아내고, s_B값을 찾아내어 분할하여 새로 생성한 동전의 종자값을 계산한다. 만약 종자값이 $c'_{j-1,n} = c'_{j,n} \cdot s_B$의 값이 되지 않으면 동전을 위조한 것이므로 사용자를 추적한다. 값이 일치하면 $c'_{j-1,0}$의 값을 계산하여 상점으로부터 받은 동전들이 유효한지 검사한다. 위의 모든 과정이 유효할 경우 상점의 계좌에 입금을 시켜주며, 그렇지 않을 경우 이 정보들을 이용하여 사용자를 추적한다.

4. 결과 및 분석

본 절에서는 본 논문에서 제안한 전자화폐 시스템의 이중사용 불가능과 위조 불가능에 대한 안전성과 분할성의 만족, 그리고 시스템의 효율성에 대하여 평가한다.

4.1 안전성

4.1.1 위조 방지

제안한 전자화폐 시스템은 동전과 지불인증에 대하여 위조가 불가능하다. 위조 방지에 대한 평가는 동전에 대한 것과 지불인증에 대한 것으로 나누어 기술한다.

■ 동전에 대한 위조방지

동전을 생성할 때 이용하는 해쉬함수는 일방향성을 만족한다. 따라서 종자값으로부터 루트로의 생성과 검증은 가능하나 그 역방향으로의 계산은 불가능하다. 동전을 지불에 이용할 때 $(c_{j,1}, c'_{j,n-1})$ 동전부터 지불에 이용하기 때문에 해쉬함수의 일방향성으로 인하여 이미 사용된 동전의 정보로부터 아직 사용되지 않은 동전을 만들어낼 수 없다. 또한 만약 (c_j, c_{n-j}),...., $(c_{j-k}, \ldots, c'_{n-j,k})$의 동전을 지불에 이용하였을 경우 제 삼자는 $j < n-i-k$인 c'_j의 해쉬값은 생성할 수 있어도 $j > i+k$인 c_j의 해쉬값을 생성할 수 없으며, 반대로 $j < i+k$인 c_j의 해쉬값은 생성할 수 있어도 $j > n-i-k$인 c'_j의 해쉬값은 생성할 수 없기 때문에 동전을 위조할 수 없으며 하나의 해쉬함수를 이용하여 동전을 구성하는 것보다 더 안전하다[5].

분할하여 사용될 동전은 한쪽 체인의 종자값을 $c'_{j-1,n} = c'_{j,n} \cdot s_B$로 계산하여 생성하여 지불해야 정당한 동전으로 입증 받을 수 있다. 만약 사용자가 분할할 동전의 종자값을 $c'_{j,n} \cdot s_B$로 하지 않을 경우 은행은 예치되는 동전들에 대하여 종자값을 검사해 보고 정당하게 분할 되었는지의 여부를 검증함으로써 그 동전의 유효성을 판단할 수 있다. 은행은 지불인증에 대한 사용자의 정보를 가지고 있으므로 사용자가 동전을 위조하여 생성할 경우 사용자를 추적할 수 있다.

■ 지불인증에 대한 위조방지

사용자는 은행으로부터 지불인증으로서 은행의 대리서명 키 쌍을 받는다. 대리서명 키의 비밀키는 $s_B = SK_B \cdot h(Cert_C \| r_B) + k_B$의 형태로서 은행의 비밀키와 은행이 임의로 선택한 정수 k_B값을 알아야 하며, 공개키를 안다 해도 비밀키를 알아내는 것은 이산대수 문제를 푸는 것과 같기 때문에 대리서명 키의 위조는 불가능하다[9]. 또한 사용자가 임의대로 지불인증을 생성하여 서명하였을 경우 상점에서는 은행의 공개키를 이용하여 $V(Root_{j-1}, R_{j-1}, PK_B^{h(Cert_C \| r_B)} \cdot r_B)$의 계산식을 수행하였을 경우 참인 값이 나오지 않기 때문에 지불인증을 임의로 만들어 분할된 동전에 서명을 할 수 없다.

4.1.2 이중 사용

사용자가 은행에서 인출받은 동전을 이중 사용할 경우 은행은 인출 시 사용자로부터 받은 정보를 이용하여 사용자의 이중사용 여부를 알아낼 수 있다. 만약 사용자가 $(c_{j,i+1}, c'_{j,n-j})$ 동전을 이중사용 했을 경우 은행은 그 동전의 루트값을 계산하여 그 동전을 생성한 사용자를 찾아낼 수 있다.

동전을 분할할 때 동전의 한쪽 체인을 이전 동전의 종자값과 은행의 대리서명 키의 비밀키를 이용하여 $c'_{j-1,n} = c'_{j,n} \cdot s_B$과 같이 생성한다. 만약 사용자가 동전을 분할하고 분할한 동전을 다시 사용하였을 때, 은행은 분할하여 생성된 동전의 정보를 이용하여 생성된 동전의 종자값을 만들고 대리서명 키를 부여할 때 저장해 놓은 사용자 정보를 이용하여 사용자를 추적할 수 있다. 또한 분할하여 새로 생성한 동전을 이중사용 할 때에도 새로 생성한 동전에 대한 사용자 정보를 저장하고 있기

때문에 사용자를 추적할 수 있다.

4.2 분할성

낮은 액면금액의 동전이 없을 때 이미 생성해 놓은 높은 액면금액의 동전을 이용하여 분할하여 지불할 수 있다. 이때 분할하여 새로 생성한 동전의 정당성은 인출 프로토콜에서 은행으로부터 받은 지불인증을 이용하여 입증 받을 수 있다. 새로 생성한 동전에 은행으로부터 서명을 다시 받지 않고 사용자가 미리 받아둔 은행의 대리서명 키를 이용하여 서명하고 지불을 수행하면 상점에서는 은행의 공개키를 이용하여 서명을 확인할 수 있다. 따라서 사용자는 지불인증인 대리서명 키를 이용하여 자신의 마음대로 화폐를 분할하여 정당하게 지불할 수 있으므로 제안한 전자화폐는 분할성을 만족한다.

4.3 효율성

제안한 전자화폐 시스템은 해쉬함수를 기반으로 하여 두 개의 체인을 구성한 후 이를 하나로 하여 동전을 구성한다. 따라서 사용자가 사용한 동전이 연계가 되기 때문에 사용자의 익명성은 보장할 수 없다. 그러나 소액(천원 이하의 돈)거래에서는 사용자의 익명성보다 효율성을 중요하게 생각할 수 있다. 제안한 논문에서는 동전을 해쉬체인으로 구성하여 동전의 루트값에만 은행의 서명을 받음으로써 생성된 동전 각각에 공개키 서명을 수행하는 방식보다 빠르게 수행될 수 있다. 또한 기존의 익명성을 보장하는 전자화폐 시스템과 비교하여 계산상 효율적이다.

T. Okamoto 등이 이진트리 구조를 이용한 분할 가능한 전자화폐 프로토콜은 계좌를 여는 인출 단계에서 4000번의 지수연산을 필요로 한다[6,7]. 또한 돈을 지불하는 단계에서 3번의 지수연산과 20번의 제곱근 연산을 필요로 한다. 그러나 본 논문에서 제안한 전자화폐 시스템에서는 서명에서 이용되는 지수연산 이외에는 해쉬연산만을 수행하면 된다. 만약 RSA 서명스킴을 이용한다고 가정하였을 때 인출단계에서는 동전을 생성할 때 $2\times j$번(j :서로다른 해쉬체인 개수)의 지수연산, 지불인증 부여과정에서 $2\times j$번의 지수연산만을 필요로 한다. 돈을 지불하는 단계에서는 동전을 분할하지 않을 경우 2번의 지수연산, 동전을 분할할 경우 3번의 지수연산을 필요로 한다. 따라서 제안한 논문은 계산상 매우 효율적이다.

또한 전자화폐 시스템에서 생성된 전자화폐는 발행기관의 서명이 있어야만 그 유효성을 인정받을 수 있다. 따라서 동전을 분할하여 사용할 경우, 새로 생성되는 낮은 액면금액의 동전들은 유효성을 인정받기 위해서 생성된 작은 단위의 동전마다 발행기관의 서명을 다시 받아야 하는 불편이 있다. 그러나 제안한 전자화폐는 새로 생성한 동전에 지불인증으로 받아둔 대리서명 키를 이용함으로써 은행으로부터 서명을 받지 않아도 된다. 또한 분할성을 만족함으로써 동전형 전자화폐에서 발생하는 거스름에 대한 대비, 작은 액면금액에 대한 은행으로부터의 재발행이 필요하지 않으므로 효율성을 증대시킬 수 있다.

5. 결론 및 향후 연구과제

전자화폐는 안전성, 이중 사용의 방지, 오프라인과 같은 기본적인 요구조건 외에 부가적으로 분할성, 양도성과 같은 요구조건들을 만족함으로서 효율성을 높일 수 있다. 본 논문에서는 이중해쉬체인을 이용하여 액면가가 서로 다른 동전들을 안전하게 생성하고, 액면가가 다른 동전들 사이에 분할이 가능하게 설계하였다. 해쉬체인을 기반으로 동전을 설계하고 은행으로부터 루트값에만 서명을 받으면 되기 때문에 빠르게 수행될 수 있다. 또한 지불인증을 이용하여 동전을 분할하여 지불할 경우 은행의 서명을 받지 않고 높은 액면금액의 동전을 낮은 액면금액의 동전으로 분할하여 지불함으로써 효율성을 증대시켰다.

참 고 문 헌

[1] 이만영 외, 전자상거래 보안 기술, 생능 출판사, 1999.

[2] R. L. Rivest and A. Shamir, "Payword and MicroMint: Two Simple Micropayment Schemes," CryptoBytes, (RSA Laboratories, Spring 1996), pp.7 11, 2(1), May 7, 1996.

[3] Q. N. Khanh, Y. Mu and V. Varadharajan, "Digital Coins based on Hash Chain," In proceeding of the ACM SIGMOD conference on Management of Data, pp.169 180, Philadelphia, 1999.

[4] Y. Mu, V. Varadharajan and L. Y. X. Lin, "New Micropayment Schemes based on Paywords," In Proceedings of 2nd Australasian Conference on Information Security and Privacy(ACISP '97), Lecture Notes in Computer Science 1270, pp. 283 293, Springer verlag, 1997.

[5] K. Q. Nguyen, Y. Mu and V. Varadharajan, "Micro Digital Money for Electronic Commerce," In Proceedings of the 13th Annual Computer Security Applications Conference(ACSAC'97), pp. 2 8, Los Alamitos, CA, USA, 1997.

[6] T. Okamoto, "An Efficient Divisible Electronic Cash Scheme," In Proceedings of Crypto'95, Lecture Notes in Computer Science, pp.438 451, Springer Verlag, Berlin, Germany, 1995.

[7] T. Okamoto and K.Ohta, "Universal Electronic

Cash," In proceedings of Crypto'91, Lecture Notes n Computer Science 576, pp.324-337, Springer-Verlag, Berlin, Germany, 1992.
[8] Y. Frankel and A. Chan, "Easy-Come-Easy-Go Divisible Cash," In Eurocrypt '98, Lecture Notes in Computer Science, pp.561-575, Springer-Verlag, Helsinki, Finland, June 1998.
[9] H. Petersen and P. Horster, "Self certified keys - Concepts and Applications," In Proceedings of Communications and Multimedia Security '97, pp.102-116, Chapman & Hall, 1997.

용 승 림
1998년 2월 이화여자대학교 공과대학 컴퓨터학과 학사. 2000년 2월 이화여자대학교 공과대학 컴퓨터학과 석사. 2000년~현재 이화여자대학교 과학기술대학원 박사과정

이 은 경
1999년 2월 이화여자대학교 자연과학대학 수학과 학사. 2001년 8월 이화여자대학교 공과대학 컴퓨터학과 석사. 현재 쌍용정보통신

이 상 호
1979년 서울대학교 계산통계학과 이학사. 1981년 한국과학기술원 전산학과 이학석사. 1987년 한국과학기술원 전산학과 공학박사. 1990년 미국 일리노이대학교 전산학과 방문교수. 현재 이화여자대학교 컴퓨터학과 교수