세션키를 이용한 효율적 소액지불시스템
(An Efficient Micropayment System using a Session Key)

정 윤 수 † 백 승 호 † 황 윤 철 † 오 충 식 †† 이 상 호 †††
(Yoon Su Jeong) (Seung-Ho Baek) (Yoon Cheol Hwang) (Chung Shick Oh) (Sang-ho Lee)

요 약 해쉬체인은 계산속도가 빠른 해쉬함수를 이용하여 체인을 구성하는 구조이다. 이 구조를 이용하여 화폐를 만들면 해쉬연산만으로 화폐의 유효성을 확인할 수 있어 지금까지 주로 적은 금액이 빈번하게 교환되는 실명 거래 환경에서 응용되었다. 그러나, 소액지불과 같은 다양한 암호학 응용에 사용되고 있는 해쉬체인 기반 시스템들은 익명성으로 인하여 지불비용이 증가하는 문제점을 가지고 있다. 따라서, 이 논문에서는 지불비용이 증가하지 않도록 인출되는 과정에서 한번의 은닉서명으로 사용자의 익명성을 보장하며, 시스템에 사용하는 인증서의 역할을 세션키로 대체하여 효율성을 향상시킨 새로운 해쉬체인 기반 소액지불시스템을 제안한다.

키워드 : 소액지불시스템, 해쉬체인, 세션키

Abstract A hash chain is highly efficient and attractive structure to use in electronic cash. Previous systems using hash chain are used extensively in various cryptography applications such as one-time passwords, server-supported signatures and microments. However, The most hash chain bas ed systems using pre-paid method provide anonymity but have the problem to increase payment cost. Therefore, in this paper, we propose a new hash chain based micromet system which improves efficiency using session key and guarantees user anonymity through blind signature in the withdrawal process of the root value without disclosing privacy information.

Key words : micropayment, hash-chain, session key

1. 서 론

초고속 정보통신망 구축에 따른 인터넷의 보급확산과 함께 인터넷 전자상거래라는 새로운 문화가 활성화되고 있다. 전자상거래는 현실 생활의 상거래 환경을 인터넷 상에서 구현함으로써 거래비용의 감소와 신속성을 높이는 효과를 가져왔다. 미래의 전자상거래에서는 고가의 상품뿐만 아니라 문서, 각종 음악화일, 동영상이나 그림화일, 증권정보와 같은 저가의 디지털 상품들이 많이 등장할 것이다[1-12].

현재까지 발표된 대부분의 해쉬체인기반 전자화폐는 트랙잭션의 처리 비용이 커지는 것을 피하기 위해 설계 단계부터 익명성을 고려하지 않거나[1-3,6], 실명으로 하는 신용기반(credit-based)의 후불 시스템으로 구성되었다[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 꺼려하며, 이것은 거래의 건전성(fressness)과도 무관하지 않다. 사이버 거래가 일반화되면 될수록 사생활 보호에 대한 고객의 관심과 요구는 더 높아질 것임은 쉽게 예측할 수 있다.

따라서, 이 논문에서는 은행으로부터 지불 권한을 위임받은 고객이 동일한 상인과 잦은 거래를 하는 경우에 판매자와 익명의 거래를 효율적으로 하기 위한 해쉬체인 기반 전용화폐를 제안한다. 제안기법은 루트값이 인출되는 과정에서 한번만 은닉서명을 하여 사용자의 익명성을 보장하고, 시스템에 사용하는 인증서의 역할을 세션키로 대체하여 효율성을 향상시킨다. 또한 기존의 분할 가능한 화폐가 익명성을 유지하면서 환불기능을 제공할 수 있도록 하기 위해 상인 유효기간 E_M, 상인의 위치정보 L_M등을 이용한다.

이 논문의 구성은 다음과 같다. 2장에서는 기존 소액

† 학생회원 : 충북대학교 전기전자컴퓨터공학부
bukmunro@netsec.cbnu.ac.kr
manitto@netsec.cbnu.ac.kr
dolpin98@netsec.cbnu.ac.kr
†† 비 회 원 : 한국과학기술정보연구원 선임기술원
ocs@kisti.re.kr
††† 종신회원 : 충북대학교 전기전자컴퓨터공학부 교수
shlee@chungbuk.ac.kr
논문접수 : 2004년 4월 8일
심사완료 : 2005년 4월 18일

지불시스템들의 지불방식에 대해 분석한다. 3장에서는 사생활 정보가 노출되지 않은 해쉬체인(hash chain) 기반 소액지불시스템을 제안하고, 4장에서는 제안 기법의 성능분석을 수행한다. 마지막으로 5장에서는 결론에 대해 기술한다.

2. 관련연구

2.1 MilliCent

MilliCent는 안전하면서도 소액거래에 사용할 수 있도록 처리비용을 줄인 전자지불시스템이다. MilliCent 전자지불 시스템은 Customer, Vendor, 그리고 Broker로 구성되며 이들 사이에는 scrip이라는 전자화폐단위를 사용한다. MilliCent의 지불과정은 먼저, Customer가 Broker로부터 Broker Script을 사놓고, 이 Broker Scip으로 원하는 Vendor의 Scrip을 산다. Broker는 미리 Vendor들에게 Vendor Scrip을 대량으로 사들였다가 Customer에게 자신의 Broker Scrip을 받고 파는 것이다. Customer는 Vendor에게서 물건을 주문한 후 Vendor Scrip을 지불한다.

MilliCent에서 처리비용을 줄이는 방법은 첫째 지불정보(Vendor의 Scrip)의 이중사용여부를 Vendor가 직접 확인할 수 있기 때문에 다른 on-line 방식의 지불시스템에 비해 인증서버와의 통신 비용이 감소된다. 대신 Vendor에서는 사용된 Scrip의 ID를 일정기간동안 기록해 두어야 한다. 둘째, 거래금액이 작기 때문에 비용이 적게 드는 암호기술을 사용했다.

MilliCent 시스템의 단점으로는 Customer가 Vendor Scrip 구입시 Broker가 Customer가 어느 Vendor에서 쇼핑한다는 정보를 알게 된다. 이는 Customer의 사생활을 침해할 수 있기 때문에 바람직하지 않다.

2.2 eCash

Cash는 네델란드의 DigiCash사에 의해 개발된 최초의 온라인형 전자현금 시스템으로 전자화폐 시스템의 주요 요구사항인 익명성을 보장한다. eCash는 익명성 보장을 위해 David Chaum이 고안한 블라인드 서명기법을 사용한다. 블라인드 서명의 기본 개념은 사용자가 은행에 인출할 현금에 대한 일련번호를 선택하며, 이 일련번호에 난수를 곱한 값을 은행에 전달하고 은행에서는 이 값에 서명하여 전자현금을 발행한다. 사용자는 이 전자현금을 받아서 자신만이 알고 있는 난수로 나누어 원래의 일련번호를 갖는 전자현금을 얻게 되는 것이다. eCash의 단점으로는 사용자가 일련번호를 생성할 때 난수 발생기를 사용하므로 아무리 큰 범위의 값을 사용한다고 할 지라도 충돌의 가능성이 있다. 이렇게 되면 이중지불을 안전하게 확인할 수 없게 된다. 또 은행은 전자화폐의 유효성을 검증하기 위해 고유번호 데이타베이스를 검색하여 일련번호가 존재하지 않으면 유효한 전자화폐로 간주한다. 따라서 이중 지불을 검사하기 위해서는 사용된 모든 화폐의 일련번호를 관리해야 한다. 이로 인해 사용자가 많아지면 일련번호 데이타베이스가 커지고 확장성이 떨어지게 되는 단점이 있다.

2.3 PayWord

Payword 지불 시스템은 소액지불에 중점을 두고 MicroMint와 함께 제안된 소액 전자지불 시스템이다.

이 시스템은 Hash Chain에 기초를 두고 있으며, 전자화폐인 Payword를 소비자가 직접 발행한다는 특징이 있다. 그림 2에서와 같이 거래를 희망하는 소비자는 브로커에게 자신의 신용카드번호를 전송하여 인증서를 발급받아 Payword를 생성한다. 이 지불시스템에서 소비자는 Hash chain을 생성하기 위해 임의의 T_n을 선택하고 Hash를 계속 수행하여 차례로 아래와 같은 T_n, T_{n-1}, ⋯ , T_0를 얻는다.

$$T_0 = \mathrm{H}(T_1)$$
$$T_1 = \mathrm{H}(T_2)$$
$$\vdots$$
$$T_{n-1} = \mathrm{H}(T_n)$$
$$T_n = \text{상품가격}$$

소비자는 상품대금으로 T_0부터 T_n까지 차례로 상점에게 전달하고 상점은 T_n을 Hash 함수를 수행하여 T_{n-1}과 비교한 뒤 지불정보의 유효성 여부를 판단한다. 그러

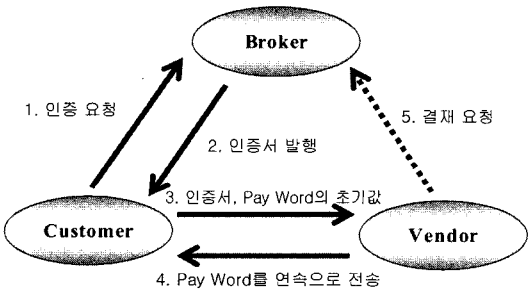

그림 1 Millicent의 시스템 구성

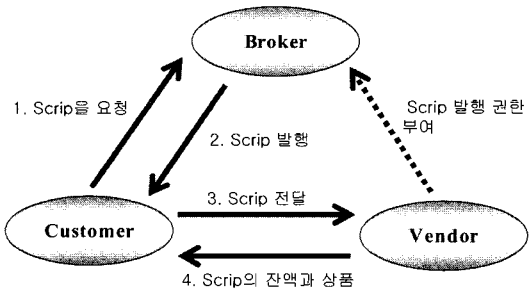

그림 2 PayWord의 시스템 구성

나, 소비자가 직접 Payword를 발행하기 때문에 다른 소비자의 Payword와 충돌을 일으킬 수 있다는 문제점을 가지고 있다[5]. 또한 PayWord를 연속으로 전송하는 단계를 제외하고, 모든 과정에서 공개키 암호화 알고리즘을 수행하여 속도가 저하되는 단점이 있다. 또한 PayWord는 연속적인 거래가 가능케하기 위해 브로커가 전자서명한 인증서를 유효기간동안 계속 사용할 수 있도록 하는 신용기반의 후불방식을 채택하였다. 이러한 후불방식은 소비자가 화폐를 남용할 소지가 있어 시스템 자체의 신용도가 낮아질 수 있다.

2.4 그 외 소액 전자지불 시스템

그 외의 소액 전자지불 시스템으로 MITLCS의 Ronald Rivest와 Weizmann Institute of Science의 ADI Shamir가 제안한 MicroMint와 PayWord방식을 변형한 MPTP[13], 그리고 PayWord방식과 Ecash방식을 혼용하여 제안한 Wenbo Payment[14]가 있다.

MicroMint는 minted라고 불리는 coin을 브로커가 생산하고 소비자는 브로커로부터 coin을 사서 상점에 지불하는 방식이다. 공개키 암호화를 사용하지 않고 공유키도 사용하지 않는다. 그러나, 소비자의 요구보다 많은 coin을 생산하기 때문에 자원의 낭비가 발생하고, 위조 방지를 위해 Hash 이외에 부가적인 연산을 수행해야 한다.

3. 제안된 해쉬체인기반 소액지불시스템

이 논문에서는 고객의 사생활 정보가 노출되지 않는 전자화폐의 동전을 구성할 때 Payword와 같이 해쉬함수를 이용한다. 전자화폐는 고객, 은행 그리고 상인 등 세 개의 구성원으로 이루어지며 동작 프로토콜은 고객과 상인이 은행에게 계정을 개설하는 등록 프로토콜, 고객과 은행사이의 인출 프로토콜, 고객과 상인간의 지불 프로토콜 그리고 상인과 은행간의 예치 프로토콜로 구성된다.

3.1 용어정의

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

- C : 고객
- B : 은행
- M : 상인
- C_M : 거래 코드
- ID_C : 거래에서 사용되는 고객의 익명 ID
- ID_{M_k} : 상인 M_k의 ID
- ID_B : 은행의 ID
- L_M : 상인의 위치정보
- E_M : 상인의 유효기간
- K_{CB} : 고객과 은행간에 공유된 비밀키
- K_{CM} : 고객과 상인간에 공유된 비밀키
- K'_{CM} : 은행에서 생성한 고객과 상인간의 공유된 one-time 세션 키
- K_{M_kB} : 상인 M_k와 은행사이의 공유된 비밀키
- $\{M\}_I$: M은 비밀키 K_{ij}에 의해 암호화
- $H(\cdot)$: MD5와 SHA-1과 같은 암호 해쉬 함수
- $H^r(W_N)$: 함수 H가 W_N을 r번 적용한 결과는 $H^r(W_N)$이고 이 결과를 표시하면 다음과 같다.

$$H^r(W_N) = \underbrace{H(H\cdots(H(W_N))\cdots)}_{\leftarrow \quad \text{r times} \quad \rightarrow}$$

3.2 등록 프로토콜(Registration protocol)

등록 프로토콜과정에서 고객은 자신의 사생활 정보와는 상관없는 익명의 신원정보 ID_C를 선택하여 은행에게 전달한다. 전달된 신원정보 ID_C는 모든 고객들 사이에서 유일하고 고객이 소유하고 있는 고객의 식별번호 카드(identification card)나 패스포트(passport)를 이용하여 은행에게 검증받게 된다. 이런 과정을 통해 은행이 고객을 검증하는 과정이 끝나면 은행은 고객에게 계정을 만들어 주고 익명의 신원정보 ID_C와 고객의 실제 정보를 묶어 저장한다. 그리고, 은행은 고객에게 은행과 공유할 수 있는 비밀키 K_{CB}을 전달한다. 고객과 동일한 과정을 통해 상인은 은행에게 계정을 받고 은행과 공유할 수 있는 비밀키 K_{MB}를 전달받는다.

이와 같은 비밀키 공유 작업이 모두 끝나면 은행은 고객과 상인사이에 사용하게 될 비밀키 K_{CM}을 생성하여 고객과 상인에게 전달한다. 마지막으로 은행은 상인이 생성한 위치정보 L_M을 $M(ID_M)$의 실제 인식 정보와 묶는다. 이런 과정은 상인이 상점 전용 화폐를 구성하여 간접적으로 특정 사용자에게 피해를 줄 수 있는 부분을 제거하기 위해 사용된다.

그림 3 등록 프로토콜의 세부적인 동작 과정을 살펴보면 다음과 같다.

• 단계 1: 고객은 자신의 사생활 정보와는 상관없는 임

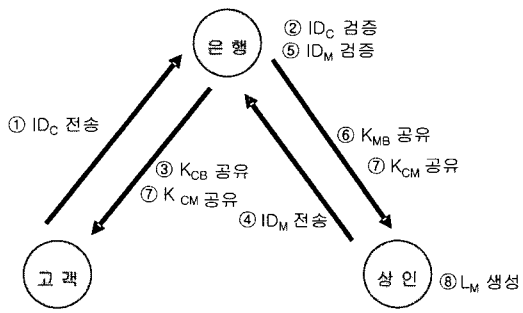

그림 3 등록 프로토콜

의의 신원정보 ID_C를 생성한 후, 생성한 정보를 검증받기 위해 ID_C를 은행에게 전달한다.

- 단계 2~3: ID_C를 전달받은 은행은 고객의 신원정보를 통해 ID_C를 검증한다. 검증이 끝나면 은행은 고객의 계정을 만들고, 고객의 고객 식별자 정보를 데이타베이스에 기록해 놓는다. 이런 과정이 모두 끝나면 고객과 은행은 비밀키 K_{CB}를 공유한다[17].
- 단계 4~5: 상인도 고객과 유사한 방법으로 은행에게 계정을 개설한다.
- 단계 6: 은행은 상인과 정보를 공유하기 위해 비밀키 K_{MB}를 전달한다.
- 단계 7: 공유 작업이 모두 끝난 후에 은행은 고객과 상인사이에 사용하게 될 비밀키 K_{CM}을 생성하여 고객과 상인에게 전달한다. 전달된 비밀키 K_{CM}은 고객과 상인사이에서만 공유되어야 하기 때문에 은행은 비밀키 K_{CM}을 삭제한다.
- 단계 8: 상인은 고객의 익명성 및 사생활 정보를 노출시키지 않으면서 서로 다른 상인과 거래를 할 수 있도록 상인의 위치정보 L_M을 생성하고, 은행은 상인의 위치정보 L_M과 $M(ID_M)$의 실제 인식 정보를 묶는다.

3.3 인출 프로토콜(Withdrawal protocol)

고객은 은행으로부터 화폐를 인출받기위해 그림 4에 기술된 프로토콜을 사용한다. 기술된 프로토콜의 세부적인 동작과정은 다음과 같다.

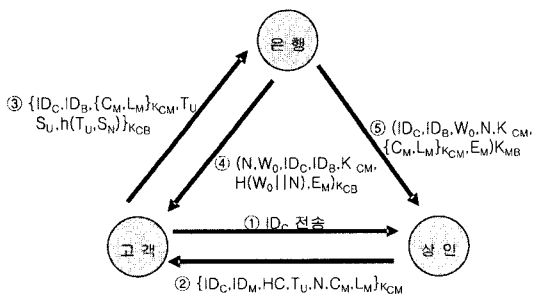

그림 4 인출 프로토콜

- 단계 1: 고객은 상인에게 고객의 신원정보 ID_C를 전송한다.

고객은 상인으로부터 서비스를 요청하기 전에, 상인을 방문하여 거래 코드, 상품 대금, 상인 위치정보등의 구매정보를 이용하여 다중의 상인에게 지불할 수 있는 해쉬체인 값을 만든다.

상인은 거래코드 C_M, 상인의 실제정보를 은행과 묶기 위해 필요한 상인위치정보 L_M, 그리고 고객이 지불의 종자값으로 사용할 해쉬 체인값을 생성한다.

상인은 해쉬 체인값을 생성하기 전에 해쉬 체인의 키 값으로 (1)을 생성한다.

$$T_u=\{ID_C,\ r_M,\ K_{CM}\} \quad (1)$$

T_u는 고객의 신원정보 ID_C와 상인이 생성한 난수 r_M, 고객과 상인사이에 공유된 비밀 세션키 K_{CM}으로 구성된다. 생성된 T_u는 종자값의 키 요소와 같은 함수의 역할을 한다. 상인에 의해 생성된 새로운 해쉬값이 상인이외에는 어느 누구도 해쉬 값을 생성할 수 없도록 하기 위해 사용된다. 또한 루트에 은닉서명을 하기 위해 T_u의 구성요소인 r_M을 사용한다.

그리고 T_u에 사용된 고객의 신원정보 ID_C는 고객의 사생활 정보와는 상관없는 익명의 신원정보를 선택하여 사용한 것이고, 모든 고객에게 유일한 신원정보이기 때문에 사용자의 익명성을 보장할 수 있다.

$$HC = \{S_i \mid S_i = h(S_{i+1},\ T_u),\ i= N-1,...,0\} \quad (2)$$

상인은 고객이 수행하는 해쉬 체인 연산과 유사한 방법으로 새로운 해쉬 체인값 HC를 생성한다. 즉 상인은 N에 대하여 임의의 S_N을 선택하고, i=N-1,...,0에 대하여 해쉬 체인값을 생성한다. 상인이 생성한 정보 중 거래코드 C_M, 임의의 수 r_M은 상인이 랜덤하게 선택한 수로서 이중사용 방지 및 고객의 다중 상품 관리를 해주는 역할을 한다. 고객은 C_M, r_M정보 이외에 S_N과 T_u를 n번 해쉬함수에 적용한 상품 대금 HC와 함께 구매정보를 구성한다.

- 단계 2: 상인은 고객에게 상인정보를 전달한다.

상인이 거래코드 C_M 및 상인의 위치정보 L_M을 고객에게 전달하게 되면 은행은 상인이 고객에게 전달한 거래코드 및 위치정보 L_M을 알 수 없어 고객의 구매정보를 추적할 수 없게 된다. 상인은 이와같이 생성된 (3)의 정보를 고객에게 전송한다.

$$\{ID_C,\ ID_M,\ HC,\ T_u,\ N,\ C_M,\ L_M\}_{KCM} \quad (3)$$

이 방법은 해쉬함수와 비밀키를 사용하기 때문에 공개키를 사용하는 방법보다 비용측면에서 소액지불시스템에 더 적합하다.

- 단계 3: 고객은 은행에게 인출요구 정보를 전달한다.

(4)의 인출요구 정보 중 $\{C_M,\ L_M\}_{KCM}$은 T_u에서 생성한 고객과 상인의 세션키로 암호화하였기 때문에 은행은 거래코드를 알 수 없으며 거래코드를 통한 구매자의 구매정보를 추적할 수 없게 된다. 이 방법은 eCash의 blind 서명기법 보다 소액지불시스템에 적합하다.

$$(ID_C,\ ID_B,\ \{C_M,\ L_M\}_{KCM},\ T_u,\ S_N,\ h(T_u,\ S_k))_{KCM} \quad (4)$$

- 단계 4: 은행은 고객에게 인출응답 정보를 전달한다.

단계 3에서 사용자가 은행에게 보낸 정보는 상점의 정보(ID_B)를 포함하고 있기 때문에 은행은 고객의 ID_B 테이블을 유지하면서 사용자가 보낸 정보를 K_{CB}로 복호

화하여 상점과 사용자 사이의 공통된 세션키를 생성한다. 그리고 은행은 고객의 요구사항을 이행하기 위해 상품대금과 은행에 남아있는 대금을 비교하여 해쉬체인의 길이 N을 결정하고, 해쉬체인의 루트인 W_N을 임의의 수로 선택하여 (5)의 식을 만족하는 해쉬체인 W_0, W_1, ..., W_N을 생성한다.

$$W_i = H(W_{i+1}),\ i = N-1, N-2, ..., 0 \quad (5)$$

W_0는 해쉬체인의 종자값(SEED)이라고 불리우는 마지막 해쉬값이다. 체인의 W_i는 소액지불을 위해 고정된 값을 미리 결정한다. 그리고, 은행은 고객과 상인간의 공유된 one-time 세션 키 K'_{CM}을 생성한다. 그 때 고객은 은행으로부터 (6)과 같은 인출응답 정보를 전달받는다.

$$(N,\ WN,\ W_0,\ K'CM,\ H(W_0||N),\ E_M)_{Kcn} \quad (6)$$

$H(W_0||N)$은 해쉬체인의 명확성을 보이면서 해쉬체인의 상태정보를 저장하기 위해 사용되고 있다. 은행은 고객의 계좌에 남아있는 금액과 상품대금에 대한 모든 정보를 가지고 상품에 대한 지불여부를 판단할 수 있다.

- 단계 5: 상인은 은행에게 권한정보를 전달받은 후, 권한정보를 체크한 후 저장한다.

은행은 상인에게 권한을 주기 위해 (7)과 같은 정보를 전달한다.

$$(N,\ W_0,\ ID_C,\ ID_B,\ K'_{CM},\ E_M\ \{C_M,\ L_M\})_{Kcm})_{Kcn} \quad (7)$$

상인은 은행에게 정보의 타당성을 체크받은 후 권한정보를 저장한다.

3.4 지불 프로토콜(Payment protocol)

고객은 인출한 화폐를 상인에게 지불할때에는 그림 5에 기술된 프로토콜을 사용한다. 그림 5에서 고객은 $W_i=H(W_{i+1})$(i=N-1, N-2, ...,0)과 같은 등식으로 은행으로부터 수신된 W_N을 N번 해쉬한다.

- 단계 1: 상인에게 지불내용을 전달한다.

인출비용의 명확성을 체크하기 위해 W_0와 일치하는지 마지막 해쉬값을 비교한다. 만일 모든 검증이 통과되면 고객은 (8)의 지불내용을 상인에게 제출한다.

$$\{N,\ ID_C,\ ID_B,\ (W_1,\ 1),\ C_M,\ L_M\}_{K'CM} \quad (8)$$

$(W_i,\ i)$는 체인의 해쉬값과 인덱스로 구성된 지불쌍으로 표현한다. 상인은 구매정보를 위해 처음 해쉬값을 재구성하여 지불의 무결성을 체크할 수 있다.

- 단계 2: 상인은 고객에게 서비스를 전달한다.

상인은 ID_C와 일치하는 은행 권한 메시지가 지불명령을 전달한 고객에 의해 제공되고 있는지를 데이타베이스를 통해 검색한다. 상인은 상인의 데이타베이스에 저장된 E_M을 체크하여 해쉬체인을 명확하게 검증하기 위해 고객으로부터 수신된 W_1을 해쉬하여 해쉬값을 계산한다.

$$H(W_1) = W_0 \quad (9)$$

W_0가 상인의 데이타베이스에 저장된 해쉬 체인의 루트이다. 만일 모든 검증이 성공적으로 검증되었다면, 상인은 아이템이나 서비스를 은행이나 고객으로 제공된 N값에 의해 나눈다. 그리고 고객에게 아이템이나 서비스가 전달된다.

첫 지불쌍 $(W_1,\ 1)$이 상인에게 수신되면 고객은 다중지불을 위해 $(W_i,\ i)$를 상인에게 전달하고, 상인은 마지막 해쉬 값 W_N까지 계속적으로 아이템이나 서비스를 전달한다. 고객과 상인 사이의 모든 세션이 끝나면 상인은 해쉬체인 W_0 값, 구매 관련정보, E_M, 고객의 ID_C, 은행의 ID_B등 고객 및 은행 권한 정보를 제거한다.

3.5 예치 프로토콜(Deposit protocol)

지불을 수신하기 위해 상인은 거래가 끝난 후에 은행과 함께 고객으로부터 수신된 해쉬체인을 되찾아야 한다. 상인은 고객의 지불 구조로부터 유추된 (10)의 예치 요구를 은행에게 전달한다.

$$(W_N,\ N,\ ID_C,\ ID_M)_{Kmn} \quad (10)$$

예치 요구를 전달받은 은행은 해쉬체인을 통해 은행의 데이타베이스에 저장된 E_M파라미터를 검증하거나 예치 요구에 관련된 각 항목의 명확성을 검증한다. 만일 검증이 성공하면 은행은 상인의 계좌에 돈을 전달하고 지불과 관련된 정보를 삭제한다.

4. 시스템 분석

이 장에서는 제안된 시스템의 분할성, 공평성, 보안성, 효율성, 익명성등을 분석하고, 기존 해쉬체인 기반 지불 시스템들과 비교하여 장단점을 논의한다.

4.1 분할성(Divisibility)

고객이 구매정보와 관련된 정보를 은행에게 전달하는 방법은 크게 두 가지가 있다. 첫째는 고객이 상인과 거

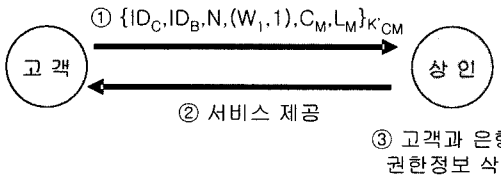

그림 5 지불 프로토콜

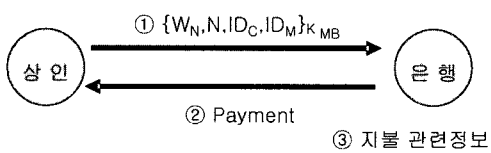

그림 6 예치 프로토콜

래하고자 할때이고, 두 번째는 고객이 은행으로부터 필요한 전자 화폐를 모두 얻기 위해 인출을 원할때이다. 은행은 고객에 의해 전달받은 총액을 나타내는 N과 W_N을 선택한다. N 동전을 얻기 위한 동전 체인은 P 하위체인으로 나누어 질 수 있다($P \le N$). 만일 동전 체인이 $\{W_{iN}\}$(i=0,1,...,N)으로 주어진다면 다음과 같은 하위체인이 만들어진다.

$$C_1,\ N,\ W_N,\ W_i,\ i = 0.1,...,C_l$$
$$C_2,\ N,\ W_N,\ W_i,\ i = C_1+1,\ C_1+2,...,C_1+C_2$$
$$C_P,\ N,\ W_N,\ W_i,\ i = C_l+...+C_{P-1}+1$$
$$C_1+...+C_{P-1}+2,...,C_l+...+C_{p-1}+C_P$$

하위체인이 만들어지면 은행은 C_1, C_2,..., C_P는 C_1, $C_2+...+C_P$=N을 만족하는 각 상인의 서비스의 값을 표현한다. 그리고, 은행은 W_0, W_{C_l}, $W_{C_l+C_2}$,...,$W_{C_l+C_2+\cdots+C_P}$을 저장하고 고객과 상인에게서 각 하위체인의 타탕성이 만족하는 동안 저장된 W값을 상인에게 전달한다. 각 상인은 고객의 구매정보와 일치하는 하위체인의 종자값만 획득할 수 있다. 예를 들어 상인 M_K는 하위체인 C_K : $Wc_1+\cdots+c_{k-1}$의 종자값을 수신받는다. 지불이 발생하면 고객은 상인에게 (11)과 같은 M_K의 지불 구조를 보낸다.

$$\{C_K,\ ID_C,\ ID_B,\ L_M,\ W_{C_l+\cdots+C_{k-1}},\ OI_K\}_{K'CM} \quad (11)$$

M_K는 $H(W_{C_1+\cdots+C_{n-1}+1}) = W_{C_1+\cdots+C_{n-1}}$이 맞는지 검증하고, 고객에게 전자 아이템이나 서비스의 단위를 보낸다. 세션이 끝난후에 M_K는 은행에게 (12)의 지불요구를 보낸다.

$$(W_{C_1+\cdots+C_{n-1}},\ W_{C_1+\cdots+C_i},\ C_k,\ ID_C,\ ID_M)_{K_{m,B}} \quad (12)$$

동전을 생성할 때 이용하는 해쉬함수는 일방향성을 만족한다. 따라서 종자값으로부터 루트의 생성과 검증은 가능하나 그 역방향으로의 계산은 불가능하다. 동전을 지불에 이용할 때 C_K: $W_{C_1+\cdots+C_{n-1}}$동전부터 지불에 이용하기 때문에 해쉬함수의 일방향성으로 인하여 이미 사용된 동전의 정보로부터 아직 사용하지 않은 동전을 만들어낼 수 없다. 또한 만약 $C_1+...+C_{P-1}$의 동전을 지불에 이용하였을 경우 제 삼자는 j<n-i-k인 보다 작은 C_K의 해쉬값은 생성할 수 있어도 j>i+k 인 C_K의 해쉬값은 생성 할 수 없으며, 반대로 j<i+k인 C_K의 해쉬값은 생성할 수 있어도 j>n-i-k인 C_K의 해쉬값은 생성할 수 없기 때문에 동전을 위조할 수 없으며 하나의 해쉬함수를 이용하여 동전을 구성하는 것보다 더 안전하다[6]. (j 번째체인, i번째동전, k: 동전갯수, n: 해쉬체인길이)

분할하여 사용될 동전은 한쪽 체인의 종자값을 계산하여 생성하여 지불해야 정당한 동전으로 입증 받을 수 있다. 만약 사용자가 분할할 동전의 종자값을 계산하지 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.

4.2 공평성(Fairness)

제안기법은 다른 구성 요소간에 은행이 신뢰성을 갖을 수 있는 선불방식이다. 제안 기법에서 고객은 요청된 아이템이나 서비스가 수신되기 전에 은행에게 돈을 지불해야 한다. 이렇게 하면 상인과 은행사이에 발생되는 불이익을 예방할 수 있다. 은행은 고객이 이중지불의 위험성을 줄이기 위해 거래에 사용된 해쉬 체인의 타당성을 생성하는 책임을 진다. 반면 고객과 동일한 은행의 권한 메시지는 상인이 고객의 구매요구가 받아들이기 전에 요구되고 은행은 데이타베이스를 통해 지불 명령의 타당성을 체크한다. 지불이 되면 상인은 고객의 이중지불을 막기위해 고객과 관련된 정보를 삭제한다. 동일한 이유로 은행은 상인이 두 번 지불요청하는 것을 예방하기 위해서 지불에 관련된 정보를 제거한다. 지불 프로토콜 실행에서 고객의 지불과 상인의 서비스는 거래가 취소되도 추가적인 이익이 발생하지 않도록 단계적으로 수행된다. 은행은 체크의 타당성 이후에 상인으로부터 수신된 해쉬체인의 일부를 회수한다. 제안 시스템은 고객과 상인 사이가 공평하다.

4.3 보안성(Security)

이 논문에서는 지불 프로토콜이 실행되면 공격자가 위조할 수 없도록 하기위해 해쉬 체인을 사용된다. 그 이유는 $h(\cdot)$가 강한 암호학인 ONE-WAY 함수를 사용하고 있기 때문이다. 그리고, 고객과 상인사이의 지불내용을 세션키 K'_{CM}를 사용하여 암호화하는 시점에 은행은 입금 과정이 끝나면 데이타베이스로부터 입금 요구의 타당성을 검증하는 정보를 제거한다. 상인은 은행에 의해 요청된 타당성 조사가 통과되지 않고는 다른 어떤것도 되찾지 못한다. 제안시스템은 고객의 이중소비와 초과소비(overspend)로부터 고객을 예방한다.

공격자가 임의로 어떤 고객의 화폐를 인출하기 위해서는 먼저 그 고객의 신원을 증명할 수 있어야 한다. 이 증명을 하기 위해서는 고객의 공개 신원정보를 통해 고객의 실제 신원정보를 알아야 하지만 제안기법에서는 고객 자신의 사생활 정보와는 상관없는 익명의 ID_C를 선택하여 사용하였기 때문에 고객의 실제 신원정보는 알 수 없다. 이런 이유로 인해 사용자의 익명성을 보장받을 수 있다. 또한 고객은 신원을 증명할 때 시간 정보를 포함하므로 공격자는 기존 증명을 다시 사용할 수 없다.

4.4 효율성(Efficiency)

제안된 소액지불시스템의 효율성은 선불구조이고 상

인은 고객의 지불 명령의 복호화를 통해 타당성을 검증한다. 따라서 4개의 비밀 대칭키 K_{CB}, K_{CM}, K_{BM} 그리고 K'_{CM}이 고객, 은행, 상인사이에 필요하다. 즉, 세션키 이외의 어떤 인증서도 요구하지 않는다. 거래가 이루어지는 동안 해쉬 동작(해쉬 체인 생성과 검증)의 수는 Payword 기법과 동일하다. 그러나 정보는 은행에 의해 저장되고, 상인은 실행이 이루어지고 난 후 정보가 지워진다. 제안 시스템은 저장과 계산, 프라이버시 보호 측면에서 효율적이다.

그러나 초기 Payword 기법에서 고객은 은행에 의해 생성된 인증서와 루트 값 W_0의 디지털 시그너처, 기타 다른 정보 및 지불을 수신받기 위한 상인의 ID등을 생성한다. 이것은 Payword가 후불시스템이기 때문이고 디지털 시그너처는 후에 고객이 지불하기 위한 약속으로써 사용되었다. 그러나 공개키 기반의 이러한 디지털 시그너처 요구는 항상 단점이 되고 있다. 이런 문제를 해결하기 위해 제안기법에서는 공개키 대신 세션키를 사용하여 연산비용을 향상시키고 있다.

4.5 익명성(Anonymity)

고객의 익명성은 거래를 하는동안 예방할 수 있는 소액지불시스템의 중요한 부분이다. 제안기법에서 사용되는 해쉬체인은 은행에 의해 매도되고 상인에 의해 매수된다. 거래에 사용되는 고객의 익명 ID_C는 고객의 실제 정보와 관련이 없다. 상인은 고객이 거래하는 것을 결정하지 못하지만 은행은 상인의 위치정보가 포함된 고객의 구매정보를 이용한다.

de Solages와 Traore의 제한적 은닉서명 정리 2에 의해 은행은 서명과정에서 얻은 정보로부터 서명한 메시지나 결과 서명에 대한 어떤 정보도 얻을 수 없다[15]. 뿐만 아니라 서명에 포함되는 C_M, L_M을 인출 과정에서 보지 못하므로 어느 판매자용 화폐를 인출했는지 알 수 없다.

4.6 기존 분할 가능한 화폐와 비교

기존 해쉬체인 기반 지불시스템과 제안시스템의 비교는 표 1에 요약되어 있다. 표 1에서 제안 시스템과 Nguyen등의 시스템은 길이가 N인 해쉬체인을 인출한다고 가정하며, 일반 동전방식의 화폐는 N개의 동전을 인출한다고 가정하고 비교한다[16,17].

해쉬체인을 기반으로 화폐를 구성하고 있는 기존 기법들은 공개키를 사용하여 거래를 수행하지만 제안기법에서는 공개키대신 세션키를 사용하고 있기 때문에 기존 기법들보다 효율적이면서 인증서를 사용하지 않기 때문에 연산비용이 적게 든다. 그리고 제안기법에 사용되고 있는 해쉬체인은 체인의 루트를 통해 식별되므로 익명성을 제공하기 위해서 Nguyen등의 시스템이나 일반 오프라인 동전에서는 n번 은닉서명을 하지만 제안기법에서는 루트값이 인출되는 과정에서 한번만 은닉되기 때문에 효율적이다.

또한, 대부분의 분할 가능한 화폐[18-21]는 이진트리 구조를 이용하고 있다. 이 이진트리 구조는 인출하기 전에 미리 만든 구조가 아니며 지불할 때 필요한 노드만 만들어 사용한다. 반면 제안 시스템에서는 어떤 임의의

표 1 기존 해쉬체인 기반 지불시스템과의 비교

	선불 : ○ 후불 : □	범용 : ○ 판매재 전용 : □	익명성	효율성	인출비용	지불비용
Payword	□	□	×	공개키, 인증서사용	×	- 체인루트에 대한 서약(서명) 확인 - 나머지 : 해쉬연산
iKP 기반 소액지불시스템	□	□	×	공개키, 인증서사용	×	- 체인루트에 대한 서약(서명) 확인 - 신용한도 확인 - 나머지 : 해쉬연산
Mao의 시스템	○	○ 상인이 거스름	△ 가명 사용	공개키, 인증서사용	한번 은닉서명	- 여러 상인에게 지불할수록 그 다음 상인이 확인하여야 하는 정보가 많음
Nyugen의 이중잠금 해쉬체인	○	○ 이중잠금 해쉬체인	×	공개키, 인증서사용	한번 일반서명	- 체인루트에 대한 서약(서명) 확인 - 나머지 : 해쉬연산 - 영수증 발급
Nyugen의 시스템	○	○ 각 동전에 서명	○ 각 동전에 은닉서명	공개키, 인증서사용	n번 은닉서명	- 첫 동전: 서명과 시도와 응답 - 나머지 : 서명
제안된 시스템	○	○ 지불대금에 서명	○ 루트에 은닉서명	비밀키, 인증서 필요없음	한번 은닉서명	- 체인루트에 대한 서약(서명) 확인 - 나머지 : 해쉬연산
일반 오프라인 동전방식	○	○ 각 동전에 서명	○ 완전한 연결불가능성	공개키, 인증서사용	n번 은닉서명	- 각 동전마다 서명과 시도와 응답

대금을 지불하기 위해 그만큼의 노드를 전달할 필요가 없으며, 그 만큼의 시도와 응답을 할 필요도 없다. 항상 최종 값 하나만 전달되며, 시도와 응답을 한번 수행하거나 서명을 하나 생성한다.

5. 결 론

해쉬체인은 저렴한 해쉬함수를 이용한 구조이므로 전자화폐에 적용하기에 매력적인 구조이다. 그러나 해쉬체인을 이용한 초기 지불 시스템은 고객의 프라이버시를 보호하지 못하며, 한 판매자에게만 지불할 수 있는 판매자 전용화폐이었다. 해쉬체인의 장점을 유지하면서 선불방식의 범용화폐로 전환하기 위한 노력이 있었지만 이 시스템들은 이중사용 문제를 효율적으로 해결하지 못하였기 때문에 해쉬체인의 장점을 제대로 반영하지 못하고 있었다.

이 논문에서는 고객의 익명성을 보장하면서 효율성을 향상시킨 소액지불 시스템을 제안하였다. 해쉬 체인의 사용때문에 복잡해지는 이중사용 문제는 익명의 화폐가 범죄에 악용되는 것을 막기 위해 추적 기능을 이용하여 인출과 지불 비용을 최소화 하였다. 또한, 시스템에 사용하는 공개키 대신 비밀키를 사용하여 인증서의 역할을 수행하지 않음으로써 효율성을 향상시켰고, 거래코드 C_M과 상인의 위치정보 L_M등을 이용하여 익명성을 유지하면서 환불 기능을 제공하고 있다. 향후 연구에서는 제안된 방식보다 효율적이면서 안전한 기법에 대한 연구가 필요하다.

참 고 문 헌

[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 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.

[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] 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.

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

[11] Jing-Jang Hwang, Tzu-Chang Yeh, Jung-Bin Lie, "Securing on-line credit card payments without disclosing privacy information," computer Standards & Interfaces 25, pp. 119-129, 2003.

[12] Network Working Group, "AAA Authorization Appllication Examples," RFC 2905, http://www.faqs.org/rfcs/rfc2905.html.

[13] Phillip M. Hallam-Baker, "Micro Payment Transfer Protocol(MPTP) Version 0.1," W3C Working Draft, 1995.

[14] Ellis Chi, "Evaluation of Micropayment Schemes," HP Lab, technical report, 1997.

[15] de Solages, A. and Traore, J., "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, FC 1998, LNCS 1465, pp. 275-295, Springer, 1998.

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

[17] 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, :NCS 1465, pp. 275-295, Feb. 1998.

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

[19] T. Okamoto and K. Ohta, "Universal Electronic Cash," In proceedings of Crypto'91, Lecture Notes in Computer Science 576, pp. 324-337, Springer-Verlag, Berlin, Germany, 1992.

[20] Chan, A., Frankel, Y., and Tsiounis, Y., "Easy Come-Easy Go Divisible Cash," Advances in Cryptology, Eurocrypt 1998, LNCS 1403, pp. 561-575, Springer, 1998.

[21] Nakanishi, T. and Sugiyama, Y., "Unlinkable Divisible Electronic Cash," Proc. of the 3rd Int. Workshop on Information Security, ISW 2000, LNCS

1975, pp. 121-134, Springer, 2000.

정 윤 수

1998년 청주대학교(이학사). 2000년 충북대학교 대학원 전자계산학과(이학석사) 2003년~현재 충북대 전기전자컴퓨터공학부 전자계산학과 박사과정. 관심분야는 암호이론, 암호알고리즘, 정보보호, Network Security, 이동통신보안, 전자상거래보안

백 승 호

2003년~현재 충북대 전기전자컴퓨터공학부 전자계산학과 석사과정. 관심분야는 침입탐지, 정보보호, Network Security

황 윤 철

1994년 한남대학교 전자계산공학과 1996년 한남대학교 전자계산공학과(공학석사). 1999년~현재 충북대 전기전자컴퓨터공학부 전자계산학과 박사수료. 관심분야는 인터넷, 정보보호, Network Security

오 충 식

2004년 충북대학교 전자계산학(이학석사) 1986년 3월 KIST 부설 시스템공학연구소. 2005년 2월~현재 한국과학기술정보연구원 재직(선임기술원). 관심분야는 인터넷, 정보보호, Network Security

이 상 호

정보과학회논문지 : 정보통신
제 32 권 제 1 호 참조