論文
4-4-12

인터넷 상거래를 위한 소액대금결제 시스템의 설계 및 성능평가
(A Design and Analysis of Micro-payment System for Internet Commerce)

성 원(Won Seong)[1] 김 의 정(Eui-Jeong Kim)[2] 박 종 원(Jong-Won Park)[3]

요 약

기존의 대금지불 방법들은 개별 정보들과 같은 소액 상품의 결제 수단으로는 부적절하다. 이유는 상품의 가치보다 결제 수단을 운영하는데 들어가는 비용이 더 크기 때문에 경제성이 전혀 없기 때문이다. 그리하여, 인터넷 상에서 활발한 거래가 예견되는 초소액의 정보 상품들을 경제적으로 처리할 수 있는 새로운 대금결제 방식이 요구된다. 최근에 제안된 몇몇 소액지불 시스템들로는 "Millicent", "PayWord", "MicroMint", "iKP" 등이 있다. 이들 각각은 나름대로의 방식으로 안전성과 소비용화를 추구하고 있다. 그러나, 이들 방식들은 핵심 메커니즘의 소비용화와 적절한 보안의 만족 등에는 문제가 없었으나 "계좌(account)의 관리"와 "지불 화폐의 집중,관리" 에는 큰 문제점을 가지고 있다. 소액대금결제 시스템에서는 계좌 관리의 복잡성과 지불 값들의 집중의 효율적인 해결이 가장 먼저 고려되어지고 우선시 되어져야 할 항목이다. 그리하여, 본 논문은 PayWord시스템을 바탕으로 적절한 보안 만족, 운영 비용의 저렴화를 이룰 수 있게 하는 핵심 메커니즘의 사용, 불필요한 계좌의 이중 사용 배제, 지불 화폐 값들의 집중관리 해결 등을 만족시켜줄 수 있는 효율적인 소액대금결제 시스템인 PayHash를 설계, 구현하였다.

ABSTRACT

for the low information goods which will be traded through Internet is impossible to manage with previously existed payment system. The reason is that it's not economic because the management cost is bigger than the benefit of the information goods trade. Therefore, recently, there have been micropaymentresearches such as "Milicent","PayWord","MicroMint", and "iKP",etc. Though these methods don't have any problem with the low cost of the mechanism and the satisfaction of adequate security, they have big problem with the use of the unnecessary account and the aggregation of payment bill.

The PayHash system which has been developed in this study simplifies the system's mechanism with "one-way hash function" which is used in generation, payment, and verification ofthe bill. And the system removed the generation and use of unnecessary account

1) 정회원 : 충남대학교 컴퓨터공학과 박사과정 수료
2) 정회원 : 공주대학교 컴퓨터교육과 조교수
3) 정회원 : 충남대학교 컴퓨터공학과 교수

논문심사 : 2003. 6. 23.
심사완료 : 2003. 7. 18.

by making one customer have one account. The system solve the problem of the payment aggregation by using the last payment hash value and its index. And the system improves its performance by reducing the use of "digital signature" drastically, as well. As the result of this study, the PayHash system made it possible for the participants of the Internet Commerce to trade the lowest cost goods through efficient maintenance.

1. 서론

최근 컴퓨터의 대량 보급과 인터넷의 급속한 확산으로 인해서 인터넷을 상업적으로 이용하려는 "전자 상거래"의 움직임이 크게 일고 있다. 그러나, 인터넷 상의 전자 상거래는 무한한 가능성이라는 긍정적인 측면과는 대조적으로 여러 가지 어려운 걸림돌들도 가지고 있다. 이러한 문제점들에는 네트웍 환경의 구축, 물품의 배달, 대금 결제, 법률적 문제 등이 있다. 이러한 문제점들이 해결되어야만 전자 상거래가 활성화될 수 있을 것이다. 이러한 전자 상거래 활성화 과제 중에서도 특히 돈을 주고 받는 메커니즘인 "대금 결제"는 기술적으로 가장 취약한 항목으로 여겨지고 있다. 그리하여 최근까지 전자 화폐(Electronic Cash)[1][2][3][4], 크레디트 기반 시스템(Credit-Card Based Systems)[5][6], 전자자금이체 시스템(Digital Fund Transfer), 전자수표(Electronic Check Systems)[7] 등의 지불 시스템들이 연구되고 있다[8][9].

저장(Storage) 기술 등을 포함한 여러 컴퓨터 기반 기술들의 발전은 도처에 산재한 여러 정보들을 디지털화하여 보관하고 서로 공유하게 하려는 움직임을 낳았다. 이러한 정보들은 정보 상품의 형태로도 거래될 수 있다. 이때 인터넷은 디지털화된 정보를 신속하고 저렴한 비용으로 전달할 수 있는 정보 전달의 훌륭한 매체일 수 있다. 인터넷 상거래에서 거래될 수 있는 많은 상품과 서비스 중에서 디지털화된 정보는 가장 중요한 상품이 될 것이다. 이러한 정보의 예로 요즘 많이 출판되고 있는 온라인 신문, 잡지, 학술지 등과 디지털화된 음악, 영화 등의 각종 데이터베이스 서비스 등을 들 수 있다. 이들 신문이나 잡지, 학술지의 경우에 소비자 개개인이 필요한 기사나 논문은 각기 다를 수 있다. 또한 원하는 부분도 극히 적은 분량일 수도 있을 것이다. 그러나, 현재 이런 상품은 여러 기사, 논문들을 묶어 판매되고 있다. 즉, 소비자는 자신이 필요한 정보뿐만 아니라 불필요한 정보도 한꺼번에 구매하고 있는 셈이다. 인터넷 상에서의 정보 거래에서는 이러한 불합리를 개선하여 고객이 실제로 필요한 부분만 살 수 있게 하고 실제로 소비한 물품에 대해서만 물건 값을 지불할 수 있게 해야 할 것이다.

기존의 대금지불 방법들은 개별 정보들과 같은 소액 상품의 결제 수단으로는 부적절하다. 이유는 상품의 가치보다 결제 수단을 운영하는데 들어가는 비용이 더 크기 때문에 경제성이 전혀 없기 때문이다. 그리하여, 인터넷 상에서 활발한 거래가 예견되는 초소액의 정보 상품들을 경제적으로 처리할 수 있는 새로운 대금결제 방식이 요구된다. 그리하여, 최근에 제안된 몇몇 소액지불 시스템들로는 "PayWord"[10], "Millicent"[11], "iKP"[12][13], "MicroMint"[10] 등이 있다. 이들 각각은 나름대로의 방식으로 안전성과 소비용화를 추구하고 있다[14][15]. 그러나, 이들 방식들은 핵심 메커니즘의 소비용화와 적절한 보안의 만족 등에는 문제가 없었으나 "계좌(account)의 관리"와 "지불 화폐의 집중,관리" 에는 큰 문제점을 가지고 있다. 소액대금결제 시스템에서는 계좌 관리의 복잡성과 지불 값들의 집중의 효율적인 해결이 가장 먼저 고려되어지고 우선시 되어져야 할 항목이다. 이러한 문제점들이 운영비용을 늘리고 시스템을 복잡하게 만들기 때문이다. 소액대금결제 시스템들에서는 이러한 항목들의 해결이 반드시 이루어져야 한다.

본 논문은 적절한 보안 만족,운영 비용의 저렴

화를 이룰 수 있게 하는 핵심 메커니즘의 사용, 불필요한 계좌의 이중 사용 배제, 지불 화폐 값들의 집중관리 해결 등을 만족시켜줄 수 있는 효율적인 소액대금결제 시스템의 설계, 구현을 목표로 하였다.

이에 본 논문은 2장에서 대금결제 시스템들이 갖추어야 할 요구 사항들을 알아보고 3장에서는 기존의 소액대금결제 시스템들의 문제점들을 해결한 효율적인 소액대금결제 시스템 PayHash의 구조와 설계 내용을 기술한다. 4장에서는 결과를, 5장에서는 본 연구의 결론과 향후 연구과제에 대해 기술한다.

2. 관련 연구

본 장에서는 전자대금결제 시스템들이 갖추어야 할 요구 사항들을 살펴본다. 1절에서는 대액대금결제시스템과 소액대금결제 시스템을 모두 포함하는 전자대금결제 시스템들이 공통적으로 갖추어야 할 보안요구 사항들에 대해서 알아본다. 2절에서는 전자대금결제 시스템들 중 소액대금결제 시스템들이 갖추어야 할 요구 사항들을 살펴본다.

2.1 전자대금결제 시스템의 보안요구 사항들

본 절에서는 전자 상거래의 보안 요구 사항들을 살펴본다[13]. 즉, 상거래를 이루고 있는 각각의 Party들(고객, 상인, 브로커 등)이 상거래에 신뢰하고 참여할 수 있기 위한 요구 사항들이다. 네트웍을 통해서 상거래에 참여하는 각 party들이 각기 다른 party들을 어떤 믿을 수 있는 근거도 없이 신뢰해 줄 수는 없는 노릇이기 때문에 상거래에 참여하는 이는 각각의 거래 참여자의 성격에 따라서 (예를 들어, 고객이면 고객의 성격) 요구되는 보안 요구 사항들을 만족할 때만이 함께 상거래를 이루는 상대방 party를 믿고 안전한 상거래를 이룰 수 있다. 이에 고객과 상인, 브로커의 입장에서 요구되는 보안 항목들을 다음과 같이 구분한다.

2.1.1 브로커의 요구 조건들

안전한 상거래는 상인을 위한 두 가지 증명들을 요구한다.

(1) B1 – 고객에 의한 거래 증명

브로커가 어떤 한 양만큼 한 신용카드 계좌(credit-card account)로부터 금액을 변제했을 때, 그 브로커는 이 신용카드의 주인이 이 지불을 인증한다는, 위조할 수 없는 증명을 소유해야만 한다. 이 증명은 다른 거래를 위한 증명으로서 "재사용"될 수 없어야만 한다. 이것은 다음 두 가지를 의미한다.

첫째, 이 증명은 적어도 거래 금액(amount), 통화량(currency), 상품항목설명(goods description), 상인 신분증명(merchant identification), 배달 주소(delivery address) 등을 확신시킨다는 것이다.

둘째, 이 증명은 재사용이 가능한 방식으로는 습득될 수 없어야 한다는 것이다.

위와 같은 의미에서, 어떤 상인이든 악의를 품은 적(adversary)이 될 수 있는데, 심지어 이런 상인이라도 위조된 변제를 생성할 수는 없어야만 한다.

(2) B2 – 상인에 의한 거래 증명

브로커가 어떤 한 상인에게 대한 지불을 인증하려고 할 때, 이 지불이 그 상인이 정말 만들어 낸 것인가에 대해서 확신할 수 있는 증명을 브로커는 가지고 있어야 한다. 즉, 브로커가 소유하게 되는 상인에 대한 증명은 바로 그 상인이 만들어 낸 것이라는, 부정할 수 없는 증명이어야 한다.

2.1.2 상인의 요구 조건들

안전한 상거래는 상인을 위한 두 가지 증명들을 요구한다.

(1) M1 – 브로커에 의한 거래 증명

상인이 현재 행해지는 상거래를 믿고 고객에게 서비스를 하려면 당연히 브로커가 보장하는 거래이

어야만 할 것이다. 즉 상인은 이 지불을 브로커가 보증한다는 확실한 증명을 필요로 한다. 이것은 브로커에 대한 보증(Certification)과 인증(Authentication)을 포함한다. 그래서, 상인은 실제 제대로 된 브로커와 접촉하고 있다는 사실과 실질적인 인증 정보들의 보증(certification)을 알 수 있다. 다시 거래 금액(amount), 통화량(currency), 거래 날짜와 시간, 거래가 동일함을 알 수 있는 정보들이 보증 됨을 알려주는 증명이다.

(2) M2 – 고객에 의한 거래 증명

거래의 효율성의 차원에서, 상인은 브로커로부터 거래 인증을 받기 전이라도 고객이 이 거래를 인증한다는 확실한 증명을 필요로 한다.

2.1.3 고객의 요구 조건들

지불을 행하는 고객에 대한 다음과 같은 보증들이 요구될 수 있다.

(1) C1 – 부정한 카드 사용의 봉쇄

고객의 신용 카드를 가지고 있는 사람이 모두 다 정당한 고객이 될 수는 없다. 이와 함께 정당한 고객임을 증명할 수 있는 부가적인 정보들을 제시하고 그것들이 확실히 인정된 고객의 지불만이 가능할 수 있다.

(2) C2 – 브로커에 의한 거래 증명

당연히 고객은 브로커가 현재 행하고 있는 거래를 보장한다는 증명을 가지기를 원한다. 이것을 "영수증(receipt)"이라고 할 수 있는데, 이것이 크게 중요성을 띠지는 않지만 가지고 있다면 그만큼 편리하다.

(3) C3 – 상인에 대한 신뢰

고객은 상인이 브로커로부터 인정되고 있다는 증명을 필요로 한다.

(4) C4 – 상인으로부터의 영수증

지불을 받은 상인이 자신이 지불을 받았다는 사실과 그 지불에 대한 상품 전달을 약속하는 "영수증(receipt)"이 필요하다.

(5) C5 – Privacy (프라이버시)

최근에 크게 부각되고 있는 요구사항으로서 고객이 자신의 주문과 지불 정보에 대한 사항들을 타인이 모르기를 당연히 바랄 것이므로 요구되는 항목이다.

(6) C6 – Anonymity (익명성)

고객은 자신의 주문과 지불 정보에 대한 비밀 보장 이외에도, 자신의 존재를 적이나 엿듣는 이들, 또는 심지어 악의를 가지고 있는 상인들까지도 모르기를 원하는데서 기인되는 요구사항이다.

2.2 소액대금결제 시스템의 요구사항

본 절에서는 소액대금 결제 시스템들이 갖추어야 할 시스템 요구 사항들을 살펴본다. 소액대금결제 시스템은 소액의 상품들을 다룬다는 특이성 때문에 기존의 대액대금결제 시스템들과는 여러 가지 면에서 다른 요구 사항들을 필요로 한다. 이는 소액대금결제 시스템은 소비용화 추구와 보안성 만족이라는 두 측면을 상호보완적으로 적절하게 조화시켜야 함을 말한다. 소액대금결제 시스템의 요구 사항은 다음과 같다.

2.2.1 인증 (Authorization)

대액지불 시스템과 마찬가지로 소액지불 시스템에서도 상거래 참여자들이 안심하고 그 상거래에 참여하기 위해서는 상대방에 대한 어떤 믿을만한 근거가 있어야 하므로 소액지불 시스템이 반드시 믿을만한 인증 메커니즘을 구성하여야 한다. 그러나, 그러한 참여자들 간의 믿을 수 있는 신뢰의 근거를 갖는다는 것은 상대방을 직접 보고 거래하지 않는다는 전자 상거래의 특성상 상당히 어렵고 부담스러운 항목이 아닐 수 없다. 그러나, 소액지불 시스템은 강도 높은 보안 인증 메커니즘

을 구성하는 것만이 최우선은 아니다. 왜냐하면 소액지불 시스템을 만족시키는 운영 비용의 최소화를 추구하는 것이 인증 정도를 높이는 것만큼 중요한 일이기 때문이다. 이에 소액지불 시스템은 소비용화를 이루면서 동시에 어느 정도의 믿을 수 있는 인증을 제공하는 방식을 갖어야 한다. 이러한 인증은 전자 사인이나 암호화(encryption)/복호화(decryption) 등을 최소한으로 이용하여 이룰 수 있다. 이때 인증은 아래의 여러 시스템 요구사항들을 포함하는 큰 개념이기도 하다.

2.2.2 위조 방지 (Unforgeable)

고객이나 상인, 그리고 브로커 중 한 참여자라도 자신의 이익을 위해서 어떤 거래 사항들을 위조할 수 있고 시스템은 이를 찾아낼 수 없다면 제대로된 지불 시스템이 될 수 없다. 어떤 참여자가 불법적으로 개인 정보나 거래 사항들을 위조한다면 시스템은 그러한 사실이 어떤 누구도 부인할 수 없도록 명백한 증거로 밝혀낼 수 있어야 한다. 그러나, 소액지불 시스템에서의 위조 방지는 값비싼 전자 사인(Digital Signature)[16]의 다량 사용 등의 방식으로 행하는 것이 바람직하지 않으며 각 참여자들 간의 전달 값들의 해쉬 적용 값들을 이용하여 쉽게 위조방지를 확인하는 방법 등의 값싼 확인 방식이 적절하다.

2.2.3 이중사용 (Double spending) 방지

돈을 주고 받는 메커니즘인 지불은 실제 세계에서의 상거래에서는 어느 정도 이중사용 방지에 안심할 수 있다. 물론 어느 정도의 지불 부인에 따른 문제점도 있지만 실제 세계에서는 한 번 지불한 돈은 지불자가 계속 가지고 있는 것은 분명히 아니므로 이중 지불의 문제를 크게 걱정하지는 않을 수 있다. 그러나, 전자 지불 시스템의 경우엔 돈이라는 것이 한 디지털화된 문자열이므로 악의를 품은 사용자가 한 번 사용한 전자화폐를 사용하지 않은 화폐인양 계속 사용할 수도 있다. 소액지불 시스템에서는 아주 적은 액수라는 특성상 대액지불 시스템에서보다는 비교적 적은 보안 강도를 가지고 이중지불 방지를 행하여야 한다.

2.2.4 효율성 (Efficiency)

효율성은 소액지불 시스템을 이루기 위해서 상당히 중요하게 여겨지는 항목이다. 소액지불 시스템은 운영 비용의 최소화를 이뤄야 하는 과제를 가지고 있으므로 결국 효율성의 문제는 얼마나 시스템의 운영 비용을 합리적으로 줄일 수 있는냐의 문제로 귀결된다. 먼저 시스템의 효율성을 높이기 위해서는 고객이나 상인이 브로커와의 잦은 온라인 (On-line) 접촉을 피하는 방법을 사용해야 한다. 이러한 오프라인(Off-line) 방식의 최대한 활용은 시스템의 효율을 높여줄 수 있다. 또다른 효율성을 높이는 방법으로는 공개키(Public-key)의 사용을 줄이는 것이다. 잦은 공개키의 사용과 확인 작업은 시스템의 보안을 높여주는 반면 효율성을 떨어뜨릴 수 있는 사항이다. 그러나, 소액지불 시스템에서 효율성을 높이기 위해서 무조건적인 보안의 소홀을 가져서는 안되는 일이므로 나름대로의 방식으로 온라인 방식과 공개키 방식의 보안을 대체시킬 수 있어야 한다.

2.2.5 지불집중 처리(Aggregation resolution)

대액지불 시스템에서라면 모든 지불 값들을 저장,기록하는 것은 당연한 일이나 소액지불 시스템에서는 고객의 한번의 지불이라는 것이 너무나도 소액이므로 수십,수백의 소액지불을 합쳐봐도 대액지불의 한번 거래량보다도 적을 수 있다. 그런데, 그러한 소액의 지불 값들을 확인을 위해서 모든 값들을 저장하고 보관하며 또한 나중에 모든 지불 값들을 일일이 전부 확인해야 한다면 이는 경제적이지 못한 일이다. 그러므로, 소액지불에서는 될 수 있는 한 지불 값들을 누적(accumulation)시켜서 하나의 대액 값을 다루듯이 저장,처리 할 수 있어야 한다.

2.2.6 계좌 관리 (Accounts)

지불값 집중 관리의 문제와 함께 소액지불의 특성상 메커니즘의 간결성과 소비용화를 위해서 요구되어지는 사항이다. 최소한의 계좌수만을 생성, 유지, 관리하는 것이 바람직한 소액지불 시스템일 수 있다. 그러므로, 소액대금결제 시스템에서는 한 고객이 여러 계좌를 가지로 거래를 행하는 것을 피하여야 한다. 그리하여 시스템의 부담을 줄여줘야 한다.

2.2.7 프라이버시(Privacy)

최근에 크게 부각되고 있는 요구사항으로서 고객이 자신의 주문과 지불 정보에 대한 사항들을 타인이 모르기를 당연히 바란다는 측면에서 크게 요구된다. 특히, 정보 상품을 사고 파는 것은 더욱더 타인에게 거래 사항을 감추고 싶을 수 밖에 없는 것일 것이다.

2.2.8 익명성(Anonymity)

고객은 자신의 주문과 거래 정보에 관련하여 그러한 거래를 행하는 존재가 정확히 누구인지를 타인들은 물론 심지어 상인까지도 모르기를 바라는 데서 요구되는 사항이다. 그러나, 익명성의 형성에도 불구하고 사용자에 대한 인증은 분명하게 이루어져야 한다.

2.3 PayWord 소액지불 모델

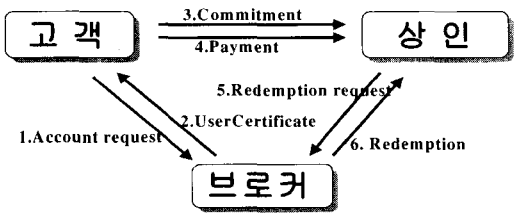
이 절에서는 기존의 대표적인 소액지불 프로토콜 중 인터넷을 통해서 소액 구매를 이룰 수 있도록 하기 위한 "PayWord"[10] 에 대하여 고찰한다.

2.3.1 핵심 메커니즘

PayWord는 크레디트-기반의 시스템이며 해쉬 값들의 사슬 메커니즘에 기반을 두고 있다. 이러한 해쉬 사슬 메커니즘을 가지고 동작되는 이 모델의 참여party들은 브로커들(brokers), 고객들(users), 그리고 상인들(vendors)이다. 전체적인 시스템 흐름은 [그림 1]과 같고 각 party별 동작 관계는 다음과 같다.

고객 U가 상인의 웹사이트를 방문하여 정보 상품을 구입하기 위해서는 먼저 고객 스스로가 브로커에게 제대로 인증받고 있는 정당한 사용자임을 상인에게 알려주어야만 한다. 그러기 위해서 U는 상인과의 거래 시작 전에 미리 브로커에게 자신의 신상 정보들을 보내서 자신이 정당한 사용자임을 증명하는 인정서인 "사용자 증명서(Certificate)"를 발급 받는다. 이때 이 Certificate에는 브로커의 전자사인을 행하므로 브로커로부터 생성된 것임을 믿을 수 있게 된다.

이후 고객은 자신이 구매하고자 하는 정보 상품을 가지고 있는 상인의 구매 웹사이트로 이동하여 상인과의 구매 관계를 맺을 수 있다. 이때, U는 루트(root) w_0를 가지고 연속적으로 해쉬 함수를 적용시켜 새로운 payword 사슬 $w_1,\ldots,w_n$을 생성하게 된다. 그리고 나서, 그 사슬의 루트 값인 w_0와 고객의 Certificate 등을 담은 commitment라는 것을 만들어 상인에게 주게 된다.



[그림 1] PayWord 시스템의 전체 흐름도
[Fig. 1] Schematic mechanism of PayWord micro-payment system

상인은 이 commitment를 통해서 고객의 정당성과 고객이 지불하는 화폐의 정당성 등을 확인할 수 있다. 지불시 고객은 첫번째 payword 값을 보내고 연속해서 payword 값을 보내게 된다. 이때, 상인은 미리 받은 payword와 나중에 보내온 payword를 해쉬 적용으로 맞춰본 후 이상이 없으면 정보 서비스를 해주게 된다.

2.3.2 문제점

PayWord 방식은 화폐의 생성과 확인 작업등이 모두 값싼 해쉬 작용으로 이루어지므로 무척 효율적인 방식으로 여겨진다. 그러나, 이 방식은 화폐로 이용되는 해쉬 값들인 payword들의 집합,유지가 문제가 되고 있다. 이 시스템에서는 먼저 고객이 거래를 하고 싶은 한 상인에게만 유효할 수 있는 증명서(Commitment)를 만들어 보내게 되는데 그 Commitment 안에는 자신이 만든 다수의 해쉬 값들의 사슬로 이루어진 막대의 첫번째 값을 포함시켜 보낸다. 그 후 연속된 해쉬 값들을 상인에게 보냄으로써 지불을 행하게 되는데 이 과정에서 고객은 자신이 지불했던 모든 payword를 저장하고 있어야 하고 지불을 받은 상인 또한 그 값들을 저장해야만 한다. 후에 상인이 자신이 모은 payword 값들을 브로커에게 보내면 브로커도 그 payword들을 저장해야 한다. 이렇게 PayWord 방식은 해쉬 값들의 저장, 유지를 위한 운영 부담이 너무나 크다고 할 수 있다. 또한, commitment마다 계좌를 구성하므로 한 고객 U에 대한 계좌를 한 상인 V가 여러 개를 가지고 있어야만 하는 문제가 있다.

3. PayHash :소액대금결제 시스템 설계

본 장에서는 PayHash 소액대금결제 시스템의 설계 내용에 대하여 기술한다. 1절에서는 PayHash 설계시의 중점 사항에 대해서 기술하고 2절에서는 PayHash의 설계 내용을 기술한다. 3절에서는 PayHash와 기존의 소액대금결제 시스템들과의 보안요구사항 만족도 비교와 소액대금결제 시스템으로서의 요구사항 만족도 비교를 행한다.

3.1 PayHash의 설계 중점 사항

본 논문에서 개발하고자 하는 시스템은 소액대금결제를 목적으로 한다. 본 논문에서는 PayWord 시스템을 바탕으로 하여 사용자 증명서와 commitment 등의 기능을 확장하고 개선하여 본 시스템을 구성하였다. 이를 이루기 위해서 가장 중요하게 고려되어야 할 점은 경제성 만족이다. 이에, 본 논문에서 개발하고자 하는 PayHash 시스템의 설계 과정에 중점적으로 고려한 사항은 다음과 같다.

3.1.1 적절한 보안 만족을 위한 설계

소액대금결제 시스템은 아주 적은 소량의 금액을 다루는 특별한 성질을 가지고 있으므로 일반적인 대액대금결제 시스템에서와 같은 값비싼 보안 장치의 마련만이 최선은 아니다. 그러나, 이것이 소액대금결제 시스템에서는 보안을 소홀히 해도 된다는 것을 말함은 아니다. 소액상품 하나가 물론 중요한 정보일 수 있으나 그것의 매매 과정이 비싼 보안 기술과 장치들로 복잡하게 얽혀 있어서 매매 과정에 드는 비용이 실제 상품 값보다 더 커서는 안된다는 것이다. 이는 강한 보안 장치의 사용이 유지 비용을 크게 증가시켜 결제 시스템을 이용하는 상거래 시장 참여자의 경제적 이득을 크게 줄일 수 있기 때문이다. 그러므로 소액대금결제 시스템은 2장에서 기술했던 전자상거래 보안 요구 조건들 중에서 가장 적절한 보안 조건들만을 선별해서 구성해야 하며 또한 만족시키고자 하는 보안 조건의 구현도 기존의 모델들과는 다르게 최소의 비용으로 동일한 효과를 낼 수 있는 메커니즘으로 구현,대체하여야 한다. 그래서, PayHash 시스템의 보안 조건은 2장의 전자상거래 보안 요구 조건들 중 C3항목과 C4항목을 제외시키면서 나머지 보안 요구 조건들을 값싼 보안 기술들로 만족시킨다.

3.1.2 시스템 운영 비용을 줄이기 위한 설계

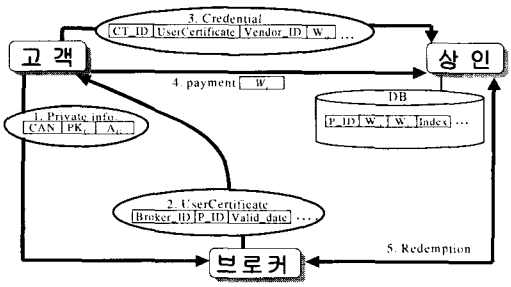
소액의 개별 정보 상품을 다루는 소액대금결제 시스템에서 운영 비용을 크게 늘리는 항목들은 다음과 같다. 첫째, 강한 보안 장치. 둘째, 지불값

들의 집중. 셋째, 불필요한 계좌(account)의 관리 등이다. 이중 강한 보안 장치의 사용은 3.1.1에서 기술하였듯이 시스템의 운영비용을 줄이기 위해서 적절하게 강도를 조절함으로써 해결할 수 있다. 그러나, 계좌의 중복 사용과 지불 값들의 집중은 기존의 소액대금결제 모델들에서 해결하지 못하고 있는 항목들이다.

최소한의 계좌수 유지와 지불 값들의 집합, 저장 문제의 해결은 소액지불 시스템들이라면 꼭 해결해야 할 문제다. 대액지불 시스템의 경우라면 분명 모든 지불 값들을 저장, 기록하여야만 할 것이고 시스템의 신뢰성 있는 운영에 필요하다면 여러 가지 계좌의 생성, 운영도 합리적이라고 할 수 있다. 그러나, 소액지불 시스템의 경우엔 고객의 지불 하나라는 것이 너무나도 적은 소액이므로 수십, 수백번의 지불을 합쳐봐야 대액지불 한번의 거래량보다도 못할 수도 있다. 그런데, 그렇게 행해지는 모든 소액의 지불 값들을 각 참여자들이 모두 집합, 저장, 유지해야 하고 여러 가지 계좌를 생성, 사용, 유지해야 한다면 불합리하다. 이에 본 PayHash는 모든 고객이 지불한 모든 지불 값들을 상인이 모두 저장하지 않고 상인이 고객의 유사 아이디를 기준으로 하나의 계좌만을 유지시켜 거기에 모든 지불 값들이 아닌 현재 사용한 지불 값과 그 값의 인덱스만을 기록하는 방식을 사용한다. 이렇게 하여 한 사용자가 여러 계좌를 생성하여서 야기되는 관리의 부담과 상인이 지불 값들을 모두 저장, 유지해야 하는 부담을 줄이고 있다.

3.2 PayHash의 설계 내용

본 논문에서 설계, 구현한 PayHash 시스템은 적절한 보안과 운영 비용의 저렴화에 초점을 맞춘 설계를 위하여 먼저 수행 중 가능한 모든 부분마다 값싼 보안 기술인 해쉬(hash) 기능을 적용시켜 지불마다 요구되는 값비싼 공용키(public key) 작용의 수를 최소화한다. 그리고, 기존의 소액지불 모델들에서 적어도 몇 개의 소액지불 때마다 반드시 행해져야 했었던 전자사인의 적용 수를 크게 줄인다. 또한 지불 값마다 브로커와 접촉하여 그 유효성을 검증한다는 것은 소액지불 시스템의 특성상 큰 부담으로 작용하므로 상인이 고객으로부터 받은 화폐의 유효성을 브로커와 off-line으로 검증할 수 있게 함으로써 브로커 접촉을 최소화 시킬 수 있도록 하였다. 또한, coin 생성과 검증 과정에 일방향 해쉬 함수의 기능을 적용시킴으로써 값싸고 간결한 지불 방식도 사용한다. 나아가, 다른 소액지불 시스템들이 모든 지불 값들을 전부 저장 보관하는 점을 개선함으로써 지불 값들의 집중 관리의 문제점을 해결하고 있다. 또한, PayWord나 MilliCent등의 기존 모델들이 너무나 많은 계좌(account)를 생성하여 시스템에 이용하는 점을 개선하여 최소한, 필요한 만큼의 계좌만을 생성, 이용한다.



[그림 2] PayHash 시스템의 전체 흐름도
[Fig. 2] Schematic mechanism of PayHash micro-payment system

PayHash 시스템의 참여자들은 고객들(users), 상인들(vendors), 그리고 브로커들(brokers)이다. 전체적인 시스템 흐름은 [그림 2]와 같고 각 참여자들의 주요 기능과 역할은 다음과 같다.

(1) 고객 (사용자)

자신이 원하는 정보 상품을 얻는 대가로 지불하여야 하는 token을 생성한다. 이를 시스템의 절차에 따라 공급자에게 지불한다.

(2) 상인 (공급자)

고객으로부터 token을 받고 그에 상응하는 정보상품 서비스를 행하여 준다. 이때 받은token의

정당성을 off-line으로 확인한다.

(3) 브로커

고객이 지불한 token의 효력을 보장한다. 이에 따라 상인이 모은 token에 대해 redemption 절차를 행한다.

3.2.1 Coin 생성 메커니즘

소액대금결제 시스템은 기존의 지불 시스템들과 마찬가지로 상품과 돈을 참여자들 간에 이동시키는 시스템이다. 상품이나 서비스의 대금이 소비자로부터 공급자에게 인터넷을 통해서 지불,이동되어져야 한다. 이를 위해서는 실세계의 화폐나 동전과 같은 가치 있는 정보형태를 만들어 내야만 한다. PayHash에서의 이러한 대금은 bit string의 디지털 정보 형태를 갖는 토큰이다. 이 토큰이 화폐의 역할을 안전하고 효율적으로 수행할 수 있게 하여야 한다.

본 연구에서 구현한 coin 생성 메커니즘은 일방향 해쉬(one-way hash) 함수의 성질에 기반을 두고 있다.

(1) 일방향 해쉬 함수의 특성 [17]

> h = H(M) (H: 일방향 해쉬 함수)
> 첫째, 주어진M에 대해서, h를 계산하기가 쉽다.
> 둘째, 주어진 h 에 대해서, H(M)=h가 되는 M을 계산하기가 어렵다.
> 세째, 주어진 M에 대해서, H(M)=H(M*)와 같은 다른 메시지M*를 찾기가 어렵다.

이와 같이 hash 함수를 사용해서 coin을 생성시키면 hash 비용이 signature보다 훨씬 저렴하기 때문에 coin생성의 비용을 크게 감소시킬 수 있다. 또 coin 각각에 전자사인(digital signa- ture)을 하지않고 coin들을 stick으로 묶어서 한꺼번에 sign함으로써 중복된 불필요한 signature을 줄일 수 있다. 이때 이용되는 hash 기술 이용은 상거래에서 상당히 중요한 위조 방지를 가능하게 하면서 비용은 값싸게 들게 하는 유용한 기술이다.

(2) Coin 지불 메카니즘 [18]

본 연구에서 구현한 소액지불 프로토콜의 방식은 PayWord[10], NetCard[19]에서의 제안 방식인 해쉬 값들의 사슬 메커니즘에 기반을 두고 있다. 소액지불을 위하여 해쉬사슬(hash-chain)을 이용하는 응용은 Anderson[19]과 Pederson[18]에 의해서도 독립적으로 제안되어져 왔다.

이 방식은 각각의 coin을 사용하기 위해 고객이 해쉬 함수를 이용해 coin들의 stick을 만들고 그 stick에 사인해서 상인에게 보낸다. stick에 들어갈 coin의 생성은 다음과 같은 과정에 따른다.

고객은 임의적으로 선택한 정보로 만든 마지막 payhash W_n을 가지고 역순으로 (식2)와 같이 해쉬함수를 적용시켜 payhash stick을 만든다.

$$w_i = h(w_i+1) \quad (i = n-1, n-2, n-3 \cdots, 0) \quad (2)$$

이제, 고객은 부가적인 정보들과 함께 w_0를 상인에게 보내고 그 후에 고객은 stick에서 차례로 각 coin들을 꺼내서 지불하고 싶은 만큼 상인에게 보내고, 상인은 간단한 해쉬 함수로 계산을 수행해 봄으로써 각 coin의 유효성을 검증할 수 있는 방식이다.

이 방식은 hash 기능 적용의 소비용화로 인하여 소액지불 프로토콜의 구현에 가장 적합한 것으로 이해된다.

3.2.2 고객-브로커 기능 설계

[그림 3]에서처럼 고객 U가 브로커 B에게 자신의 계좌(account)와 거래시 자신의 존재를 증명시켜줄 "고객 인증서(UserCertificate)"를 요구함으로써 두 참여자간의 관계가 시작된다. 이는 고객의 상거래 참여의 시작이기도 하다. U는 안전한 채널을 통해서 B에게 자신의 크레디트-카드 번호 CAN, 공개키 PK_U , 그리고 자신의 "배달 주소" A_U 등을 준다. 이후 고객이 써버린 payhash 지불들은 추후에 상인 V의 요구에 의해 P_ID 당사

자의 크레디트-카드 계좌로 부과된다. 그의 배달 주소는 그의 인터넷/전자우편 주소다.

$$P\_ID = H(R_C, CAN) \quad (3)$$

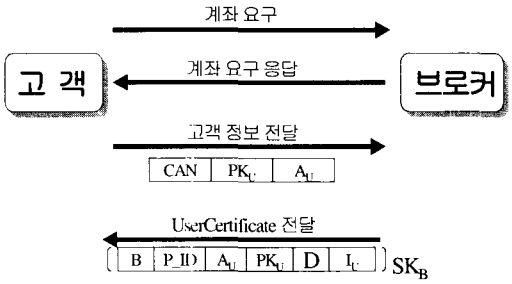
위 (식3)와 같이 고객을 나타내는 P_ID는 B가 자신이 임의로 생성해낸 랜덤값 R_C와 고객의 크레디트 번호 CAN을 합쳐서 해쉬 함수에 적용시킨 결과값으로서 고객의 의사 아이디(Pseudo ID)가 된다. B는 정보저장소에 고객명, R_C, CAN 등의 형태로 저장하여 추후 사용한다. 이 P_ID는 여타 소액지불 시스템들이 갖추지 못한 익명성의 특성을 갖게 해주는 데 쓰인다.

고객의 인증서는 고객이 보내온 위에서와 같은 고객의 정보들을 가지고 B가 만들어서 다시 고객에게 주는 것으로서 이 고객이 정당한 존재임을 증명한다는 정보들을 담고 있다.

이 인증서(UserCertificate)는 (식 4)처럼 브로커 식별자 B, 고객 의사 ID P_ID, 배달 주소 A_U, 사용자의 공개키(public key) PK_U, C_U의 만기일 E와 인증서의 시리얼 번호, 각 상인들에게 주어질 수 있는 신용 한계 등의 정보 등도 I_U에 담고 있다. 그래서, 고객의 인증서는 CU의 형태를 갖게 된다. 여기서 $\{\}SK_B$ 는 {}안의 내용을 B의 비밀키로 전자 서명한다는 것을 나타낸다.

$$C_U = \{B, P\_ID, A_U, PK_U, E, I_U\}\, SK_B \quad (4)$$

이 C_U는 고객-상인 간의 지불이 행해질 때, Credential의 내부에 포함되어져서 쓰이게 되는데 U가 구매를 하고 지불을 하기에 앞서서 Credential를 V에게 제시하게 된다. 이때 V가 자신이 알고 있는 B의 PK_B을 가지고 C_U를 복호화시켜 그 Credential의 내부를 열어서 보면 해당 U가 만기일까지 B에 의해서 정당한 존재로서 인정되는 U인지 아닌지를 확인할 수 있게 된다. 이것은 곧 정당한 존재로 밝혀진 U의 경우일 때는 해당 U가 지불해서 V가 모아둔 payhash값들을 B가 되사준다(redeem)는 것을 V가 믿어도 된다는 것을 의미한다.



[그림 3] 고객-브로커 기능
[Fig. 3] Customer-broker architecture

3.2.3 고객-상인 기능 설계

고객과 상인간의 관계는 고객이 상인의 웹사이트를 방문하여 몇 가지 정보 상품을 구입하고 그 대가로 대금을 지불하는 것으로 설명할 수 있다. U가 새로운 상인 V에 접촉했을 때, U는 루트(root) w_0를 가지고 연속적으로 해쉬 함수를 적용시켜 새로운 payhash 사슬 $w_1, \ldots, w_n$을 생성하게 된다. 여기서 n은 고객의 편리에 의해서 선택되어지고 고객은 그 사슬을 위한 Credential (CT)를 다음 (식 5)와 같이 계산한다.

$$CT = \{ CT\_id, V, C_U, w_0, D, I_M \}\, SK_U$$
$$CT\_Sig = H(P\_ID + CT) \quad (5)$$

여기서 CT_id 는 Credential의 ID, V는 상인을 나타내고 CU는 U의 인증서, w_0는 payhash 사슬의 루트, D는 유효 날짜, I_M은 부가적인 정보들(예를 들면, payhash 사슬의 길이 등), P_Id는 고객의 의사 아이디를 나타낸다. CT는 U에 의해서 비밀키로 사인되어 V에게 보내진다. V는 CT에 씌워진 U의 사인(signature)를 검사하고 CT에 포함되어져 온 C_U에 씌어진 브로커의 사인도 검사한다. 이를 통해 V 는 U의 존재를 믿게 되고 이후 행해지는 U의 지불을 받아들이게 되는 것이다. V는 U의 부정을 조사하기 위해서 그날의 끝까지 Credential를 저장해야만 한다. 그러나, 다른 해쉬 사슬 메커니즘에 기반한 유사 시스템들은 날마다 본 시스템의 Credential같은, 특정 상인에

만 유효한 지불 증명서를 만들어내고 그를 기준으로 중복되게 계좌를 만들어 관리를 한다. 그러나, 이는 소액지불 시스템으로서는 큰 부담이 아닐 수 없다. 그리하여, 본 시스템에서는 V가 고객 U의 Credential의 내부를 열어서 알아낸 P_ID를 기준으로 계좌를 만들고 여러가지 Credential마다의 계좌는 생성하지 않는다. 또한 CT를 형성하는 SK_U로의 전자서명도 처음 Credential의 생성시에만 한번 하게 되고 그 이후부터는Credential 의 전자 서명 대용인 CT_Sig를 사용, 이전의 Credential를 확인하는 방법으로 전자서명의 부담도 크게 줄이고 있다.

$$P = (w_i, i) \quad (6)$$

U와 V는 지불되어져야 할 양에 대해서 동의할 필요가 있는데 전형적인 소액지불 프로토콜에서는 상당히 적은 양을 다루게 되고 기본적인 양들을 합한 덩어리들도 취급할 수 있다. U로부터 V로 행해지는 지불 P는 (식6)과 같이 하나의 지불값과 그것의 인덱스로 구성되어 진다.

첫번째 V로의 지불은 U과 관계되는 Credential의 동봉과 함께 이루어지며 이후의 지불은 단지 payhash값과 그 인덱스만을 가지고 행한다. 이 때, 여타 소액지불 시스템 제안들은 V가 U에게 정보 서비스를 제공하고 받은 모든 지불 화폐들을 저장하고 있게 된다. 그러나, 이는 소액지불의 측면에서 바람직하지 않다. 계좌 관리의 부담이 너무 크기 때문이다.

PayHash 시스템은 해쉬 함수의 일방향성을 이용 V가 U의 P_ID 계좌에 Wo와 가장 최근에 지불된 payhash값만을 저장시킴으로써 기존의 소액대금결제 모델들에서 V가 받은 모든 지불 값들을 저장시키는 문제점을 해결하고 있다. V는 고객의 지불 정보를 자신의 정보 저장소에 P_ID를 기준으로 CT-id, C_U, Wo, CT 등의 내용 형태로 저장한다.

3.2.4 상인-브로커 기능 설계

한 상인V는 B와 미리 먼저 관계를 가지고 있을 필요는 없다. 그러나. 믿을 수 있는 채널을 통해서 B의 공개키 PK_B 을 얻어놓을 필요가 있다. 그래야만 V는 B에 의해서 사인된 인증서를 검사할 수가 있다. V는 U로부터 모은 payhash들을 B에게 되팔아(redemption)올 수 있어야 한다. 이 때 브로커들은 보통 소액지불시스템의 외적인 경로를 통해서 상인들에게 지불한다.

일반적으로 그 날의 끝에 또는 적절한 기간별로 V는 B에게 Redemption 메시지를 보내게 되는데 이 메시지는 P_ID, CT_id, Credential CT와 마지막 지불 $P = (w_l, l)$ 로 이루어진다.

4. 실험 결과

본 절에서는 2장에서 소개했던 기존의 소액대금결제 시스템들과 PayHash 시스템을 두가지 측면에서 비교를 행한다. 첫째는 전자상거래를 위한 보안 요구 사항 만족도 비교로서 비교항목의 나열과 함께 소액대금결제 시스템에서의 적절한 보안 요구 사항들을 기술하고 둘째는 소액대금결제 시스템 요구사항 만족도를 비교한다.

4.1 보안요구사항 충족 비교

〈표 1〉 소액 지불 프로토콜들의 비교
〈Table 1〉 Comparison between micro-payment systems

요구조건들\지불시스템들	PayHash	PayWord	MilliCent	iKP
브로커				
B1. 고객에 의한 거래증명	✓✓	✓✓	✓✓	✓✓
B2. 상인에 의한 거래증명	✓✓	✓✓	✓✓	✓✓
상인				
M1 브로커에 의한 거래증명	✓✓	✓✓	✓✓	✓✓
M2 고객에 의한 거래증명	✓✓	✓✓	✓✓	✓✓
고객				
C1. 부정한 카드사용의 봉쇄	✓✓	✓✓	✓✓	✓✓
C2. 브로커에 의한 거래증명	✓✓	✓✓	✓✓	✓✓
C3. 상인에 대한 신뢰			✓✓	✓✓
C4. 상인으로부터의 영수증			✓✓	✓✓
C5. 프라이버시	✓			✓✓
C6. 익명성	✓✓			✓✓

〈표 1〉에서 iKP는 3KP를 말함)

ikP(3KP)는 원래 대액 지불 프로토콜이므로 보안 요구조건 충족도는 〈표 1〉에서 보듯이 매우 만족스러우나 그만큼 복잡한 과정과 값비싼 암호화(encryption) 을 행하고 있기 때문에 소액지불 프로토콜로의 변형의 경우엔 적지않은 비용을 감수해야 하는 대가를 치뤄야 하고 Millicent는 적절한 소액지불을 만족시키나 scrip의 선매과정을 갖추기 위해서 복잡한 과정을 거쳐야 하는 부담이 크다. 이에 반해 PayHash와 PayWord는 보안요건 충족도에서 부족한 것처럼 보이나 오히려 소액지불의 프로토콜의 간결성 측면에서 이득의 차원을 따져본다면 크게 문제가 되지 않는다. C3의 요구조건 부족의 결과로 야기될 수 있는 손해(예를 들면, 고객이 지불만 하고 서비스를 받지 못하는 경우 등) 는 이 프로토콜이 소액지불이라는 차원에서 본다면 큰 문제는 아니다. 손해를 입는다고 하여도 큰 손해는 아니기 때문이다. 이러한 일이 발생되면 문제의 발생자는 나쁜 평판으로 인하여 자연스럽게 상거래 시장에서 물러나게 됨으로써 신용 있는 거래 시장을 회복하게 된다. 부가적인 보안 요구 조건인 C5, C6는 SHTTP나 SSL과 같은 여타 프로토콜의 보완으로 해결할 수 있다. 또한, C4의 경우는 소액대금결제 시스템의 동작원리와 목적상 불필요한 요구 조건일 수 있다.

4.2 소액대금결제 시스템 요구사항 충족 비교

본 절에서는 최근에 제안된 소액지불 시스템인 PayWord와 Millicent, iKP방식들과 본 논문에서 제안한 PayHash방식을 소액대금결제 시스템의 대표적인 요구 항목들에 대하여 비교한다.

비교 항목은 소액지불 시스템에서 중요시해야 할 부분들로서 계좌(Account) 축소 관리, 지불 집합(Aggregation)의 처리, 보안 강도, 익명성(Anonymity) 등이다.

〈표 2〉에서 PayHash 시스템은 여타 모델들과의 비교에서 메커니즘 간결성, Aggregation 처리, 계좌 축소 항목에서 상당히 우수한 것으로 분류하였고 익명성에서는 철저하지 않고 부분적인 익명성만을 추구하므로 보통 충족 항목으로 분류하였다. 또한, 보안 강도는 적절한 보안 요구 사항들만을 만족시키고 있으므로 강도면에서는 소액대금결제 시스템에 적합한 것으로 여겨지는 보통으로 분류하였다. 각각의 만족 사항들을 살펴보면 다음과 같다.

〈표 2〉 소액지불 시스템 비교표
〈Table 2〉 Comparison result of micro-payment requirement items

	PayHash	PayWord	Millicent	IKP
보안 강도	▲	▲	▲	●●
메커니즘 간결성	●●	●●	▲	××
Aggregation처리	●	×	×	×
계좌 축소	●	▲	×	×
익명성	▲	×	×	●●

● : 좋음, ▲ : 보통, × : 나쁨

첫째, 보안 강도 측면을 보면 PayHash 시스템은 보안요구 사항들 중 C3, C4, C5 항목들을 만족시키지 않고 있으므로 보안강도는 강하지 않은 것으로 볼 수 있다. 그러나, 이것은 오히려 소액을 다루는 시스템답게 시스템을 경량화 시키려는 목적을 만족시키는 것으로 이행되는 것이다.

둘째, 시스템의 메커니즘 간결성을 보면 PayHash는 일방향 해쉬 함수 성질을 가지고 Coin의 생성, 지불, 검증 등은 물론이거니와 인증 절차등에도 적절히 이용하여 시스템의 메커니즘을 간결화 시키고 있다. 또한, 기존의 모델들과 비교해서 전자사인의 횟수 또한 크게 줄이고 있다.

셋째, 지불집중, 계좌축소, 익명성의 측면을 보면, 기존의 모델들은 몇 개의 소액 지불 값을 위해서 지불 값의 인증서를 상인에게 보내고 그것을 받은 상인은 그 인증서 각각에 대해 계좌를 만들고 유지하며 각각의 지불 값을 모두 보관하기도 했다. 그러나, PayHash 시스템은 상인이 고객의 유사 아이디(Pseudo-ID)를 기준으로 하여 고객당 한 계좌 씩 만을 생성하여 관리하고 나아가 지불

값도 모두 저장,보관하지 않는다. 이는 부분적인 익명성의 만족과 지불집중의 해결, 필요한 계좌만의 생성, 관리를 모두 만족시키고 있다.

5. 결론 및 향후과제

소액대금결제 시스템이 제대로 만들어지기 위해서는 시스템 전반적인 보안과 핵심 메커니즘의 안전성의 추구도 중요하겠지만 무엇보다도 합리적인 소액대금지불을 성립시킬 수 있는 시스템 유지비용의 저렴화가 이뤄져야 한다. 이는 소액대금결제 시스템들은 아주 적은 소량의 금액을 다루는 특별한 성질을 가지고 있으므로 시스템의 운영에 따른 경제적인 이득을 얻으려면 일반적인 대액지불 시스템들에서와 같은 복잡하고 값비싼 보안장치의 사용은 피해야만 한다는 것이다. 또한, 보안 조건들을 모두 충족시킨다는 것은 그만큼 유지비용을 늘어나게 하기 때문에 가장 적절한 보안 조건들만을 만족시키면서 프로토콜이 이뤄져야 할 것이다. 나아가, 만족시키고자 하는 보안 조건의 구현도 최소의 비용만을 가지고 실현시킬 수 있는 적절한 메커니즘이어야 한다.

본 연구에서는 인터넷 상거래를 위한 효율적인 소액대금결제를 위해서 PayHash 시스템을 설계, 구현하였다. PayHash는 일방향 해쉬 함수의 성질을 이용한 화폐 방식을 바탕으로 하고 있고 기존의 소액지불 모델들이 가지고 있었던 불필요한 계좌의 생성과 관리를 줄였으며 지불 값들의 집중관리를 피하고 있어 운영비용을 크게 줄였다. 또한 전자 사인(Digital Signature)의 사용도 크게 줄임으로써 효율적인 소액대금결제 시스템으로서의 조건을 만족시키고 있다.

향후 연구과제로는 두 가지를 들 수 있는데, 첫째, 상인측에서의 지불 값 처리의 간결, 용이함 구축과 고객의 익명성 강화에 대한 연구를 들 수 있다. 상인측에서의 자연스럽고 간결한 지불 값 처리가 이루어지고 고객의 익명성 강화에 따른 안전한 거래가 이루어지면 보다 안전하고 효율적인 소액대금결제 시스템을 이룰 수 있을 것이다. 둘째, 제안된 시스템이 완전한 분산 형태 시스템이 될 수 있어야 할 것이다.

※ 참고 문헌

[1] Chaum, D. "Achieving Electronic Privacy", Scientific American, vol. 267, No.2, pp.76-81, 1992.

[2] Medvinsky, G., Neuman, B.C., "Net-Cash: A Design for Practical Electronic Currency on the Internet", Proc. of the 1st ACM Conference on Computer and Communications Security, 1993.

[3] Peirce, M., O'Mahony, D., "Scaleable, Secure Cash Payment for WWW Resources with the PayMe Protocol Set", Proc. of the 4th International WWW Conferences, Dec.1995.

[4] Sirbu, M., Tygar, J. D., "NetBill: An Commerce System Optimized for Network delivered Services", Proc. of the IEEE CompCon Conference, Mar.1995.

[5] Crocker, S.,Boesch, B., Hart, A., Lum, J., "CyberCash: Payment Systems for the Internet", Proc. of the INET'95 Hypermedia, 6.1995.

[6] MasterCard and Visa, "Secure Electronic Transaction: Formal Protocol Description ", Aug.1996.

[7] Neuman, B.C. , Medvinsky, G., "Requirements for Network Payment: The NetChecqueTM Perspective", Proc. Of IEEE Compcon '95, San Francisco, May.1995.

[8] JaeKyu Lee, "Analysis and Design of the Internet Based Payment System", http://iis.kaist.ac.kr/~sky/thesis/thesis.html.

[9] R. Kalakota, A. Whinston, "Frontiers of Electronic Commerce", pp.295-330,

Addison-Wesley Publishing, 1995.

[10] R. L. Rivest and Adi Shamir ,"PayWord and MicroMint: Two simple micropayment schemes", Available from authors, May 1996.

[11] Steve Glassman and Mark Manasse, "The Millicent Protocol for Inexpensive Electronic Commerce", http://www.millicent.digital.com.

[12] R. Hauser, M.Steiner, and M.Waidner, "Micro-Payments based on iKP", http://www.zurich.ibm.com/Technology/Security/publication/1996/HSW96.ps.gz.

[13] M.Bellare, J. Garay, and M.Waidner, "iKP - A Family of Secure Electronic Payment Protocols", Available from authors. July 1995.

[14] R.L.Rivest, "Electrinic Lottery Tickets as Micropayments", Proceedings of the Financial Cryptography '97 Conference.

[15] Stanislaw Jarecki and Andrew Odlyzko, "An efficient micropayment system based on probabilistic polling", Financial Crypto '97, Feb.1997.

[16] R.L. Rivest, A. Shamir, and L. Adleman, "A Method for Obtaining Digital Signatures and Public-Key Cryptosystems,", Communications of the ACM 21,2 (Feb. 1978), pp 120 - 126.

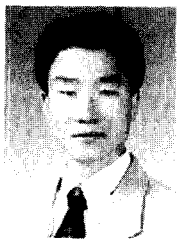
[17] Ronald L. Livest, "The MD5 message-digest algorithm", Internet Request for Comments, April 1992. RFC 1321.

[18] Torben P .Pederson, "Electronic payments of small amounts", Technical Report DAIMI PB-495, Arthus Univ, Aug 1995.

[19] R.Anderson, H.Manifavas,and C.Sutherland, "A Practical electronic cash system", Available from authors, 1995.

[20] Michael Morrison, et al., "Java UNLEASHED", Sams.net Publishing, 1996.

성 원



1997년 2월 충남대학교 컴퓨터공학과 졸업, 공학사.
1999년 2월 충남대학교 대학원 컴퓨터공학과 졸업, 공학 석사.
2002년 2월 충남대학교 대학원 컴퓨터공학과 박사과정 수료.
관심분야는 의학영상처리, 컴퓨터그래픽스, 멀티미디어

김 의 정

1993년 2월 충남대학교 컴퓨터공학과 졸업, 공학 석사.
1997년 2월 충남대학교 대학원 컴퓨터공학과 졸업, 공학 박사.
1997년 ~ 1998년 시스템공학연구소(SERI) 연구원.
1998년~ 현재 공주대학교 컴퓨터교육과 조교수.
관심분야는 패턴인식, 컴퓨터비젼, 의학영상처리

박 종 원

1979년 2월 충남대학교 전자공학과 졸업, 공학사.
1981년 2월 한국과학기술원 전산학과 졸업, 전산학 석사.
1991년 8월 한국과학기술원 전산학과 졸업, 전산학 박사.
1995년 ~ 현재, 충남대학교 공과대학 정보통신공학과 정교수.
관심분야 : 영상처리, 병렬처리공학