The Journal of Korean
Institute of CALS/EC
Vol. 4, No. 1, June 1999

전자화폐지불시스템의 위험요소 분석 및 프로토콜 설계

허철회,* 조성진,** 정환묵***

Danger Element Analysis and Protocol Design of Electronic Cash Payment System

Chulwhei Her, Soung-jin Joo, Hwanmok Chung

Abstract

The electronic cash, an electronic equivalent of the real paper money, has been recently proposed as one of the various payment methods for electronic commerce.

In this paper, we design an electronic cash payment system based on a new electronic cash payment protocol that can effectively provide full anonymity and avoid double-spending. The protocol is suitable for use as the electronic cash control or electronic cash database. The protocol support also security and electronic cash property.

* 성덕대학 전산정보처리과 전임강사
** 선린대학 전자계산과 겸임교수
*** 대구효성가톨릭대학교 전자정보공학부 교수

1. 서론

최근 들어 세계 최대의 컴퓨터 네트워크인 인터넷에 관한 관심이 점점 높아가고 있다. 또한, 월드 와이드 웹(WWW)의 등장은 인터넷을 이용한 다양한 형태의 서비스를 제공하게 되었다. 텍스트뿐만 아니라 멀티미디어를 이용한 많은 자료들을 초고속 통신망을 통하여, 시간과 공간에 구애받지 않고 서비스를 주고받을 수 있게 되었다. 많은 기업들이 WWW을 이용하여 상품의 판매와 홍보 활동의 수단으로 이용하려는 인식이 확산되면서 소비자와 판매자간의 주문, 판매, 대금결제 등의 전자적인 상거래를 수행할 수 있는 전자상거래에 대한 연구가 활발히 진행되고 있다[2].

전자상거래는 소비자와 판매자가 네트워크의 가상 공간에서 상품의 진열, 주문, 판매, 대금 결재의 단계를 수행하는 것이다. 전자상거래가 소비자에게 주는 이점은 첫째, 상점을 찾아가고, 상품을 선택 할 시간을 줄일 수 있다. 둘째, 다양한 상점과 상품에 쉽게 접근 할 수 있다. 셋째, 구입하고자 하는 물품의 질과 가격 비교가 쉬워 상품 선택이 용이하다. 또 판매자에게는 첫째, 전 세계의 인터넷 이용자를 대상으로 지역적, 공간적 제약 없이 상품을 홍보, 판매할 수 있다. 둘째, 홍보비용이나 물건 진열 등의 경비를 줄일 수 있다. 셋째, 거의 항상 지원될 수 있는 컴퓨팅 능력을 충분히 이용할 수 있기 때문에 품질 높은 상거래가 가능하게 된다. 이와 같은 전자상거래의 이점을 바탕으로 서비스의 요구가 점점 증가하고 있다[1][10].

전자상거래의 활성화를 위해 기본적으로 필요한 요건들은 쉬운 네트워크의 접속, 소프트웨어와 하드웨어 플랫폼, 물품의 배달, 멀티미디어 정보, 소비자 서비스, 개인 정보 및 프라이버시 보호, 쉬운 금액 지불 방법 등이 필요하다. 이러한 요건들이 점차적으로 성숙되면서 전자상거래가 인터넷에서 점점 구체화 되어가고 있으며, 그 중에서도 가장 핵심적인 요소로 떠오르고 있는 것이 인터넷상에서 행할 수 있는 안전하고 효율적인 대금 지불 수단인 전자지불시스템(Electronic Payment System)이다.

본 논문에서는 전자화폐를 이용한 전자지불시스템의 구현을 위해 프로토콜 메커니즘을 제안하였다. 제2장에서는 전자화폐지불 시스템의 기본적 개념과 필요한 기술들을 알아본다. 제3장에서는 전자화폐지불시스템에서 위험 요소에 관하여 알아보고, 제4장에서는 전자화폐지불시스템의 프로토콜에 관하여 제안하였으며, 전자화폐지불시스템의 안전성과 확장성에 관하여 기술하였다. 제5장에서는 결론을 기술하였다.

2. 전자화폐지불시스템

2.1 전자상거래에서의 전자지불시스템

현재 전자상거래에 사용되고 있는 전자지불시스템은 크게 네 가지로 나누어 볼 수 있다. 실세계에서 쓰이고 있는 화폐 시스템을 사이버 공간에서 사용할 수 있도록 구현하는 전자화폐시스템[3], 신용카드 거래를 인터넷상에서 구현한 인터넷 신용카드 지불시스템[3],[9], 장표 결제를 위한 수표를 인터넷상에서 구현

한 전자수표시스템[3],[8],[9], 전자지불을 위한 시스템은 아니지만 인터넷상의 가상은행(Cyber Bank)을 이용한 전자자금 이체[3]가 있다. 본 논문에서는 전자상거래를 위한 지불 시스템 중에서 안전하고, 효율적이며, 경제적인 전자화폐 시스템 구현을 위한 프로토콜에 관하여 연구하였다.

2.2 관련연구

2.2.1 Ecash

Ecash[14]는 Digicash사에서 은익서명(Blind Signature)기법을 사용한 익명성을 보장하는 전자화폐시스템이다. 사용자들은 Digital Wallet이라는 클라이언트 소프트웨어를 사용하여 중앙은행인 FDB(First Digital Bank)에서 전자현금을 인출, 지불, 예금할 수 있다. Digicash사는 익명성의 보장을 가장 중요시한다. 사용자는 난수를 발생시켜 은익서명 기법을 이용한 전자동전을 만들고, 사용자의 비밀키와 은행의 공개키로 암호화하여 은행에 보낸다. 은행은 서명을 확인하고 은행의 비밀키를 이용하여 전자동전에 서명하고 사용자의 공개키로 메시지를 암호화하여 사용자에게 보낸다. 사용자는 은익해제기법(Unblind function)을 행하여 전자화폐를 사용한다. 사용된 화폐는 은행에서 불법적으로 복사된 화폐인지를 확인하게 된다. 따라서, 사용된 화폐들은 모두 기록하게 된다. 그러므로, 전자화폐의 사용 기한이 정해져 있다. 즉, 한번 인출된 화폐는 기한 내에 사용하든지 다시 예금했다가 인출하여야 한다. 또한 중앙 집중적인 계좌관리 때문에 거래에 많은 처리비용이 소요되며, 사용 인원의 제한을 초래한다.

2.2.2 NetCash

NetCash[7],[12]는 복수의 서버를 도입하는 분산시스템으로 디지캐시보다 익명성은 약하다. 사용자의 계좌를 여러 대의 분산된 서버에서 관리하여 사용자들의 수를 극대화할 수 있도록 하였다. 또한 넷 체크(Net Cheque)라는 전자수표 시스템과 교환이 가능하도록 하였다.

은행 서버는 전자화폐의 발행 요청을 받으면, 고유번호를 생성하고 사용자의 계좌에서 현금을 인출 한 다음 발행할 화폐에 그 고유번호를 부여하여 전자서명을 한다. 발행된 전자 화폐 고유번호의 유일성을 보장하기 위하여, 은행 서버는 고유번호 데이터 베이스에 저장하여 관리한다. 그러나, 분산된 시스템을 이용하므로 트랜잭션 비용이 많이 들고, 유효기간이 지난 전자화폐는 교환하여야 한다.

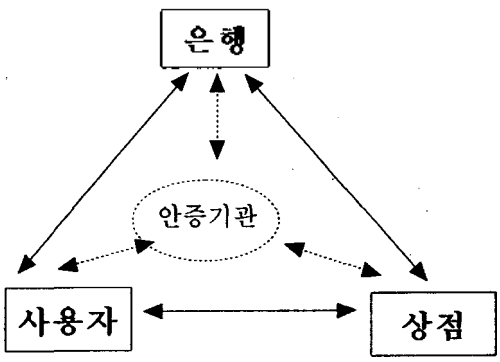
2.2.3 PayMe Transfer Protocol(PMTP)

Ecash와 Netcash의 장점을 결합하고, 단점을 보완한 익명성과 확장성을 보장하는 프로토콜이다. PMTP[20] 시스템은 온라인 전자상거래시스템으로 사용자와 은행으로 구성되어 있으며, 사용자는 소비자 또는 상인이 될 수 있으며, 은행은 전자화폐를 발행하고, 전자화폐를 보관하는 데이터베이스를 구축한다. PayMe Currency는 은행의 비밀키로 인증되며 값, 일련번호, 은행ID, 은행호스트, 포트번호, 만기일자 등을 가진다. 사용자의 완전한 익명성을 보장하지 못하며, 전자화폐는 계좌가 없는 분산된 은행에서 실세계의 현금으로 교환이 가능하다.

2.3 전자화폐지불시스템의 구성원

전자상거래에서 전자화폐 시스템의 구성원을 살펴보면 <그림 1>과 같이 구성된다.

- 사용자(User) : 온라인 쇼핑몰에서 상품을 구매하며, 대금으로 전자화폐를 사용하는 주체로서 은행에 계좌를 소유하고 있어야 한다. 사용자는 은행계좌에 예금된 금액 한도내에서 전자화폐를 발행 받을 수 있으며, 이것을 상품 대금으로 통영 시키고, 은행에 예금하여 소멸시킨다.
- 상점(Shop) : 온라인 쇼핑몰을 인터넷상에 진열하고, 판매하며, 사용자로부터 전자화폐를 구매 대금으로 받는 공급자
- 은행(Bank) : 현실 세계의 화폐와 전자화폐의 교환 역할을 한다. 즉, 사용자의 은행 계좌에서 요청 된 금액에 해당하는 전자화폐를 발급해 주는 발행기관이며, 상점에게는 전자화폐를 결제해 주는 결제기관으로 전자화폐를 소멸시킨다.



<그림 1> 전자화폐시스템 구성도

- 인증기관 : 공개키 암호화와 사용자들이 주고받는 모든 정보에 대하여 인증을 관리한다. 사용자가 공개키의 등록을 요청하면 사용자를 등록시키고 인증서를 발행하며, 인증서에는 전자서명으로 인증할 수 있도록 한다.

2.4 전자화폐 지불 시스템에 필요한 기술

(1) 암호화(Encryption)

전자지불시스템에 있어서 가장 중요한 요소 중에 하나는 암호화기법이다. 네트워크상의 자료 암호화는 주로 공개키 암호화기법(Public Key encryption Method)을 사용한다. 공개키 암호화 기법은 비대칭형 암호화 체계로서 공개키(Public Key)와 개인키(Private Key)가 한 쌍을 이룬다. 이 두개의 키로 이루어지는 암호화기법을 통해 네트워크 상에서 자료를 원하는 사람에게만 해독이 가능하게 암호화해서 전달하는 것이 가능해졌다. 공개키 암호화 알고리즘은 암호화하는 키(공개키:Public Key)와 복호화하는 키(비밀키: Private Key)가 서로 다르다는데 장점이 있다. 특히, 공개키는 누구나 접근할 수 있도록 일반에게 공개하고, 비밀키는 본인만이 알고 있도록 함으로써 부인봉쇄를 위한 전자서명에 이용한다. 알고리즘 이용 방법은 주문 메시지를 보낼 때 메시지에 비밀키를 이용해 암호화 알고리즘에 의해 계산된 값을 첨부해 보내며 수신자는 공개키를 이용해 역으로 계산해 봄으로써 이 메시지의 진위여부를 확인하는 것이다. 대표적인 공개키 방식의 알고리즘은 RSA, Diffie-Hellman, Fortezza 등이다.[6]

(2) 전자서명(Digital Signature)

컴퓨터의 디지털 정보는 완전하게 동일한 내용으로 복제가 가능하기 때문에 위조, 복제, 부인 등이 용이하다. 전자서명은 공개키 방식을 이용한다. 송신자가 수신자에게 어떤 정보를 보낼 때 정보의 체크썸(MAC)을 구하고, 자신의 암호화키로 암호화하여 수신자에게 보낸다. 수신자는 이 정보를 받아 체크 썸을 공개키로 풀어 내용이 동일하면 자료의 무결성과 송신자의 신분을 증명하게 된다. 비밀키 암호화 알고리즘에는 DES, DES40, RC2, RC4, IDEA 등이다.[5]

(3) 사용자 인증서

사용자의 인증은 권한 증명을 요구하게 된다. 인증서는 개인의 공개키를 인증하기 위한 전자 문서로서 암호화키와 관련된 정보들로 되어 있다. 인증서는 공개키의 신뢰성을 위하여 사용된다. 인증기법(Authentication Function)에는 메시지 암호화, 체크썸(Cryptographic Checksum), 해시함수(Hash Function) 등의 사용 방법이 있다. 인증서에는 인증서 번호, 사용자 이름, 공개키, 만기일, 발행기관 등의 정보가 수록되어 있다. 인증서의 형식은 X.509 표준이 사용되고 있다.[8][19]

3.전자화폐지불시스템에서 위험 요소

3.1 전자화폐 전송(Transporting Electronic Cash)시 위험

전자화폐는 전송당시, 안전하고 신뢰 할 수 있어야 한다는 전제 아래, 메시지를 암호화하여 변경(Tamper)되는 것을 방지하고, 자료 전송중의 신뢰도를 높이기 위하여 End-to-End 프로토콜을 이용함으로써 자료의 손실을 줄일 수 있다.[5]

전자화폐는 특히 관련된 파일들의 보안을 필수로 하여야 한다. 관련된 자료를 분실 또는 도난 당했을 경우 즉시 지불시스템을 중지하고, 전송 과정을 추적하여 안전하게 시스템을 관리할 수 있어야 한다.[11]

3.2 전자화폐 생성 및 관리상의 위험요소

전자화폐시스템은 반드시 고유의 ID를 갖는다. 사용자의 컴퓨터는 이진수를 이용한 64비트 이상을 사용하여 랜덤하게 생성한다. Netcash는 복수의 서버를 도입하는 분산시스템으로 사용자의 계좌를 분산된 서버에서 관리하여 확장성을 극대화시키고 있다. 그러나, 은행에서 전자화폐를 발행할 때 고유번호를 함께 부여함으로써 완전한 익명성을 보장하고 있지는 못한다.

또한, Ecash는 전자화폐 발행 요청시 사용자가 고유번호를 생성하여 고유번호를 알지 못하도록 은익서명하여 은익서명된 고유번호를 은행에 송부한다. 은행은 은익서명 된 전자화폐의 고유번호와 액면가에 전자서명을 하고 사용자에게 전자화폐를 발행한다. 그리고, 은행은 이중 지불 검사 요청이 있을 때마다, 유효한 전자 화폐의 고유번호를 고유번호 데이터베이스에 저장하기 때문에 전자화폐를 사용하는 사용자들이 증가함에 따라 DB 크기가 크게 증가한다. 또한, 고유번호를 사용자들이 만들게 되므로 고유번호의 충돌이 일어 날 수 있다. 한번 사용된 화폐를 모두 기록해야 하는 불편함

때문에 화폐의 사용 기한이 정해져 있다는 불편함이 있다. 즉, 한번 인출한 돈은 기한 내에 사용하든지, 다시 저금했다가 인출해야 한다. 그리고, 화폐사용의 모든 책임이 사용자에게 있다. 바이러스나 하드디스크의 오류로 전자화폐가 없어져도 다시 찾을 수가 없다. 따라서, 소액을 인출하여 사용하는 방식을 사용한다. 또한, 중앙집중식 계좌관리 때문에 거래의 많은 처리비용과 사용자 수의 제한이 있다.

PMTP에서는 분산된 은행의 사용자로부터 받은 전자화폐의 유효성은 실제로 발행한 은행에 연결하여 검증 받도록 되어있다. Netcash와 같이 전자화폐를 발행하는 은행마다 일련번호를 만들어 사용하고, 쓰고 나면 지우는 형태를 취하므로 유효기간이 지난 전자화폐의 갱신이 필요하며 전자화폐량의 제한을 초래한다.

4. 전자화폐시스템 프로토콜

4.1 전자화폐시스템에서의 기본적 프로토콜

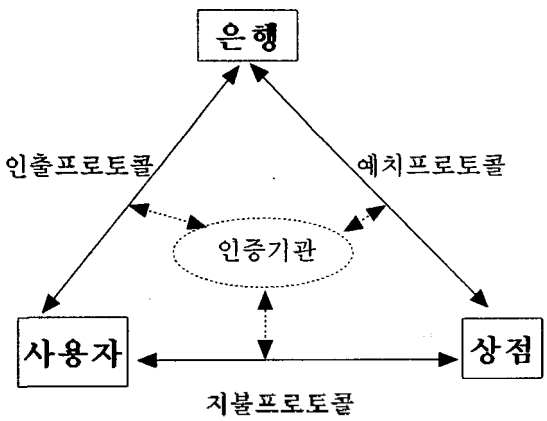
전자화폐시스템에서 기본적인 프로토콜의 요구 조건은 다음과 같다.[1][7][17]

가. 전자화폐의 건전성·안전성 확보
- -전자화폐 사용의 익명을 보장하여 프라이버시의 보장이 이루어져야 한다.
- -전자화폐의 이중사용 즉, 불법적인 화폐 복사에 대한 방지 대책이 있어야 한다.
- -사용자 인증 등을 통하여 거짓 위장 발주를 회피하는 구조가 필요하다.
- -사용자의 신용 정보(이름, 주소)의 안전성 확보가 강구되어야 한다.

나. 전자화폐 거래 비용의 현실성 확보
- -키 인증, 네트워크 비용 등의 거래 비용이 절감되어야 한다.

전자화폐시스템의 기본적인 프로토콜 요구조건을 만족하는 시스템은 구성원들 사이에서 수행되는 프로토콜에 기반을 둔다.[15] 일반적으로 구성원들의 경우와 마찬가지로 설계하고자 하는 전자화폐시스템의 요구사항과 가정이 무엇이냐에 따라 프로토콜은 달라질 수 있다. 본 논문에서는 전자화폐 발행 및 유통의 안전성과 이중 사용방지를 위한 프로토콜 설계에 관하여 논하였다.

전자화폐시스템의 프로토콜은 시스템을 구성하는 구성원과 시스템이 요구하는 기능에 따라 그 구성이 달라질 수 있다. 즉, 구성원이 많아지면 당연히 프로토콜의 수도 증가 하게 되며, 전자화폐시스템에 어떤 기능을 첨가할 경우 그것을 위한 별도의 프로토콜 반드시 필요하게 되는 것이다. <그림 2>는 구성원과 프로토콜을 나타낸 것이다.



<그림 2> 전자화폐 프로토콜 구성도

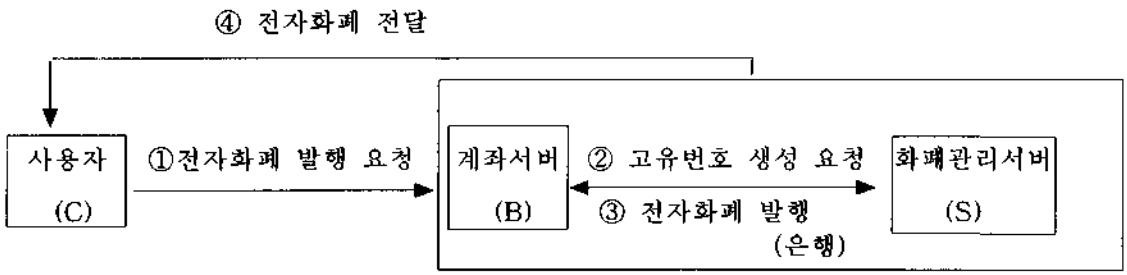
또한, <표 1>은 각 프로토콜에서 사용되는 표기법을 나타내었다.

<표 1> 표기법

구분	기호	의미
구성요소	C	사용자
	B	은행 계좌서버
	S	은행 화폐관리서버
	M	상인
암호화	PEx(M)	x의 공개키로 암호화한 메세지
	Exy(M)	x가 y에게 보내는 봉투 암호화 메시지(x의 인증 포함)
	E'xy(M)	x가 y에게 보내는 봉투 암호화 메시지(x의 인증 포함하지 않음)
	Sxy(M)	x와 y사이에 비밀키로 암호화 된 메시지
	SK	세션 키
자료	CA	금액
	EC	전자화폐 금액
	sn#	전자화폐 고유번호
	acc#	사용자 계좌번호
	pwd	사용자 패스워드
	Bid	은행 식별자 id
	Cid	해당 은행에 사용자 계좌의 Identity Number
	Ack	접수통지
	SDB(x)	은행의 디지털 서명
	Sid	전자화폐관리 서버 id

4.2 전자화폐 인출 프로토콜 (Withdrawal Protocol)

전자화폐는 은행이 가치를 보증 해주는 일종의 가치 있는 디지털 정보이다. 이러한 전자화폐의 가치를 증명해주기 위해서 은행이 이용할 수 있는 방법은 디지털 서명을 사용하는 것이다. 그러나, 일반적인 디지털 서명을 사용하게 되면 사용자 프라이버시가 보장되지 않는다는 문제점이 발생하게 된다. 즉, 은행은 사용자가 서명 받길 원하는 메시지와 그에 대한 서명문을 알게 되기 때문에, 전자화폐 발급 후 은행은 그것들을 서로 연결시킬 수 있게 되는 것이다. 그러므로, 전자화폐를 발급하기 위해서는 일반적인 디지털 서명이 아닌 새로운 종류의 디지털 서명이 필요하게 되며, 이것은 전자화폐시스템을 구성하는 프로토콜 중 인출 프로토콜을 설계하는데 기반이 된다. 인출프로토콜을 그림으로 표현하면 <그림 3>과 같이 나타낼 수 있다. 전자화폐는 사용자 계정에서 예금을 인출함으로써 발생되며, 그 프로토콜은 다음과 같다.



<그림 3> 전자화폐인출시스템

① 전자화폐 발행 요청(C -> B) : 사용자는 은행의 계좌에서 원하는 금액의 전자화폐를 인출하기 위해 은행에 전자화폐의 발행을 요청한다. 이때, 익명성을 보장받을 수 있는 은닉서명기법(blind signature technique)을 적용하며, 인증기법(SHA-1)에 의해 메시지에 대한 정보는 인증을 받게 된다.

E'cb(Sid,Bid,Cid,Ecb(pwd, acc#, CA))

② 고유번호 생성(B -> S) : 은행의 계좌서버와 화폐서버는 단 방향으로 자료처리가 가능하도록 하여 자료에 대한 보안과 익명성을 보장받도록 한다. 즉, 계좌서버에서 화폐서버로 고유번호 생성을 요청할 때 은행주소(Bid)와 해당 은행 사용자(Cid)의 주소 값만을 전달하여 고유번호를 생성하고, 화폐서버에는 사용자에 대한 자료가 남지 않도록 한다. 화폐서버는 요청한 금액에 해당하는 화폐의 금액과 고유번호를 생성한다.

Ebs(Sid,Bid,Cid,CA,SDB(Cid,),PEc(sn#),EC, SK)

③ 전자화폐 발행 (S-> B) : 사용자의 계정으로부터 금액을 인출한 후 금액에 대한 전자화폐의 금액과 고유번호가 만들어지면, 전자화폐 발행을 요청한 소비자에게 전달할 메시지를 만든다. 고유번호는 화폐관리서버의 공개키로 암호화되기 때문에 계좌서버는 전자화폐의 고유번호를 알 수 없다. 따라서, 사용자의 익명성을 보장할 수 있게 된다.

Esb(Sid,Bid,Cid,SDB(Cid),Scs(EC,PEc(sn#))

④ 전자화폐 전달(B -> C) : 은행은 사용자가 요청한 전자화폐에 은닉서명을 하여 사용자에게 전달한다. 사용자의 공개키와 자신의 비밀키를 사용하여 메시지를 암호화하여, 트랜잭션의 안전성을 보호한다. 사용자는 은행에서 보낸 메시지를 은익 해제 기법으로 전자화폐와 은닉 서명 값을 전자지갑에 저장한다.

Ebc(Bid,EC,Cid,sn#,SDB(x),Sid,SK,Ssc(EC, PEc(sn#))

은행의 계좌서버와 화폐를 발행하는 화폐관리 서버가 독립적으로 운영된다면 추적불가능한 익명성은 아니지만, 익명성은 더욱 보장받을 수 있을 것이다. 그러나, 화폐관리 서버가 독립적으로 수행될 경우 전자화폐 발행이 화폐관리 서버가 처리하는 만큼의 트랜잭션이 늘어나 처리비용이 늘어나고 처리시간이 지연될 것이다.

4.3 전자화폐 교환 단계

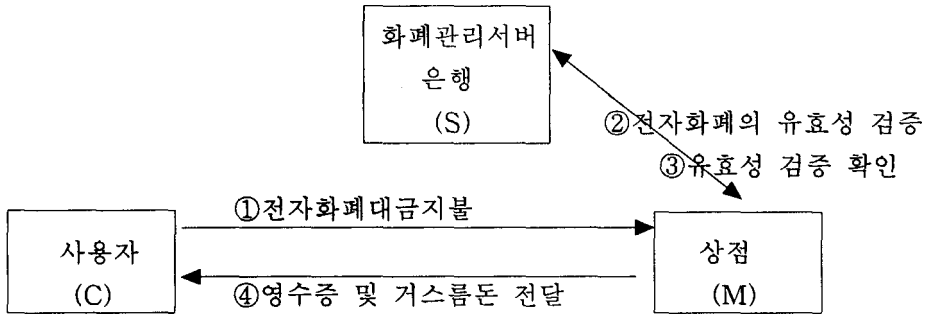
사용자는 원하는 시기에 전자화폐를 교환할 수 있다.

① 전자화폐 교환 요청(C -> S) : 사용자는 원하는 시기에 전자화폐를 교환할 수 있다. 교환 요구가 들어온 전자화폐는 난수를 사용하여 새로운 전자화폐의 고유번호를 만들고, 변경하여 사용자에게 보낼 메시지를 만든다. 그리고, 사용되었던 전자화폐의 고유번호는 삭제하여 항상 일정한 전자화폐를 유지한다.

E'cs(Sid,Bid, EC, PEs(sn#), Scs(Sid,Cid, EC, sn#, SDB(x),SK,Cid)

② 전자화폐 전달(S -> C) : 전자화폐 서버는 요청한 전자화폐를 사용자에게 전달한다.

Esc(Bid, EC, sn#, SDB(x), Sid, Ssc(EC, PE(sn#))



<그림 4> 전자화폐지불시스템

4.4 전자화폐 지불 프로토콜 (payment protocol)

사용자가 상품을 구매하고 대금을 지불할 때 사용되는 것이 지불 프로토콜이며 <그림 4>와 같이 나타낼 수 있다.

① 전자화폐 지불(C -> M) : 전자화폐를 상점(판매자)에 지불한다. 이때에도 전달되는 정보에 대한 보안을 위해 인증 메시지와 함께 상점에 보내게 된다. E'cm(EC, SDB(c), Ecm(CA, PEs(sn#),SK)

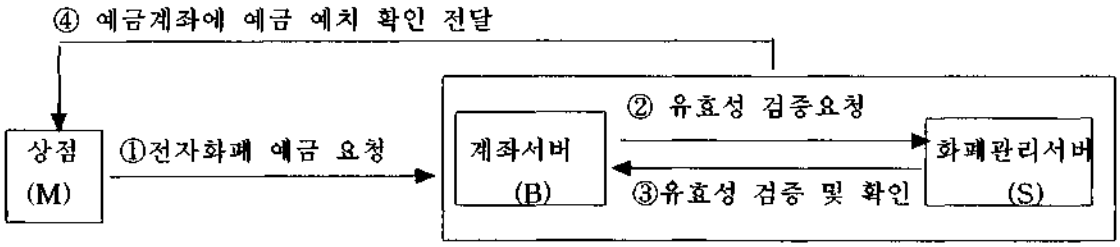
② 유효성 검증 요청(M -> S) : 상인은 전자화폐의 유효성과 이중 지불 여부를 검사하고, 거스름돈의 처리를 위해 상인이 보낸 상품 금액과 전자화폐를 화폐관리 서버로 보낸다. E'ms(Sid,Bid,Cid,EC,SDB(c),Ecm(PEs(sn#),)
전자화폐관리 서버는 은행(Bid)에 서명을 검증해 줄 것을 요구한다.

Esb(Bid,SDB(x),Sid,Cid,Ack)
은행은 비밀키를 이용하여 서명을 확인하고 화폐관리서버로 보낸다.

Ebs(Bid,SDB(x),Sid,Cis,Ack)
이때, 전자서명(SDB(x))이 일치하지 않은 다면 화폐관리 서버에 알린다. 화폐관리서버는 사용자에게 인증 서명의 오류을 알린다.
Esc(Sid,Bid,Cid,SDB(c),err_message)

③ 유효성 검증 및 지불처리(S) : 화폐관리 서버는 서명이 확인되면, 고유번호를 확인한다. 고유번호가 존재하면 유효한 전자화폐로 간주한다. 만일 지불된 전자화폐의 금액이 상품의 금액보다 클 경우에는 상품금액과 거스름돈에 해당하는 전자화폐를 발행하고 지불된 전자화폐를 화폐관리 서버에서 삭제한다. Esm(Bid, SDB(m), Sid, Ssc(EC, PE(sn#,),Cid,Bid), Ssm(EC,PEm(sn#),Cid,Bid)

④ 거스름돈 전자화폐 및 영수증 전달(M-> C) : 판매자는 영수증과 함께 거스름돈 전자화폐를 전달한다. Emc(Bid, SDB(c), Sid, Ssc(EC, PEc(sn#),Cid,SK)



<그림 5> 예치 프로토콜

4.5 전자화폐 예치 프로토콜 (Deposit protocol)

상점과 은행 사이에서 수행되는 프로토콜로서 상점이 사용자로부터 받은 전자화폐를 은행이 상점의 계좌에 입금시키고 결제해 주는 프로토콜이다. 이는 <그림 5>와 같이 나타낼 수 있다.

① 전자화폐 예금 요청(M->B) : 예금하고자 하는 전자화폐를 계좌 정보와 함께 은행에 송부한다.Emb(Bid,Sid,Cid,pwd,acc#,CA,Emb(PEs(sn#))

② 유효성 검증 요청(B->S) : 전자화폐를 수신한 은행은 유효성을 검증하고자, 전자화폐를 화폐관리 서버에 보낸다. E'bs(Bid, Sid, PEs(sn#),Ack)

③ 유효성 검증 및 확인(S->B) : 화폐관리서버는 전자화폐의 고유번호가 존재하면 은행에 유효성을 전달하고, 고유번호를 제거한다. Esb(CA, Bid, Sid, Ack)

④ 예금 예치 결과 통보(B -> M) : 은행은 전자화폐 서버로부터 유효성을 전달받아 은행 계좌에 입금 시키고, 예금 결과를 전송한다. Ebm(결과)

4.6 안전성

전자상거래에 있어 트랜잭션의 보안은 필수적이다. 또한, 전자상거래의 보안은 신원확인, 기밀성, 인증, 디지털 서명, 데이터의 무결성을 포함한다. 본 연구에서는 전자화폐를 생성하여 사용자에게 전달하는 단계와 사용자와 상점간의 지불 단계에서는 익명성을 요구하게 되므로 세션키를 이용하여 안전성을 강화 할 수 있도록 하였다. 또한, 인증기법으로는 메시지 암호화(Message Encryption) 기법을 사용하여 안전성을 보장하며, 체크 썸 해쉬 함수로는 SHA-1을 사용한다.

4.7 익명성 및 확장성

화폐관리서버는 계좌서버와 분리하여 전자화폐를 발행하고 화폐의 고유번호를 생성, 관리하므로서, 계좌서버와 화폐서버간에는 일방향으로 처리될 수 있도록 하였다. 즉, 사용자의 전자화폐 발행 요청시 계좌서버에서 화폐서버로 정보를 보낼 때 주소 정보를 이용하면 사용자에 대한 정보를 화폐서버는 알 수 없다. 또한, 은행의 계좌서버는 전자화폐의 고유번호를 알 수 없기 때문에 익명성을 보장할 수 있

다. 전자화폐 인출시 난수를 발생시켜 전자화폐의 일련번호를 만들고, RSA 기법을 사용하여 메시지를 암호화하고, 은닉서명기법을 활용한 서명을 받음으로 익명성을 보장할 수 있다.

화폐서버는 현재 통용중인 전자화폐들의 고유번호만을 가지고 있어, 이중지불 요청시 이중지불 검사가 효율적으로 수행되며, 화폐서버에서 고유번호를 생성, 소멸시키기 때문에 고유번호의 충돌을 방지할 수 있다.

분산 전자화폐 서버와 분산 은행들의 고유 ID를 이용하여 화폐를 발행할 수 있도록 유지하며, 각 서버의 데이터베이스에 중복되지 않은 전자화폐의 고유번호를 유지할 수 있도록 하며, 사용자로부터 보유하고 있지 않은 전자화폐 고유번호의 인증이나 교환 요구가 오면, 전자화폐를 소유하고 있는 서버를 호출하여 서비스할 수 있는 메커니즘을 적용하여 확장성을 높인다.

5. 결론

인터넷 사용의 증가와 전자상거래의 활성화로 전자화폐의 필요성과 운영을 위한 트랜잭션 메커니즘에 대한 요구가 증가하고 있다. 본 논문에서는 전자화폐 대금결제시스템의 프로토콜을 구현하였으며 프라이버시 보호를 위한 익명성을 제공하고, 화폐 이동의 정확성을 보장하여 전자상거래시 안전성을 높였다. 또한, 전자화폐지불시스템에서의 위험요소를 파악하여 위험에 대처할 수 있는 방법을 찾고자 하였다.

향후의 연구과제는 여러 서버들이 전자화폐를 발행할 경우 각 서버들의 연동과 처리 속도 및 트랜잭션을 줄일 수 있는 방법에 대한 연구가 필요하다.

참고문헌

[1] [이재광외 3인, 1998] "전자상거래를 위한 전자지불 시스템 분석" 1998년 한국과학재단

[2] [김영균외 3인,1998] "전자상거래 구현을 위한 요소기술 표준화 전략" 1998.5 정보과학회, pp11-18

[3] [탁승호, 1996] "전자화폐와 결제시스템", 더뱅커사, 1996

[4] [최용락, 1996] "통신망 정보보호", 도서출판 그린, 1996

[5] [홍승필,고재욱, 1998] "정보보안 기술과 구현", 도서출판 파워북, 1998

[6] [김철, 1996] "암호학의 이해", 영풍문고, 1996

[7] [정보보호센터 홈페이지, 1999] http://www.kisa.or.kr/

[8] [최인영,1998] "전자상거래 혁명", 동일출판사, 1998

[9] [L.Abad Peiro, 1998] "Designing a generic payment service" 1998. IBM System Journal http://www.almaden.ibm.com/journal/sj/371/abadpeiro.html

[10][N. Asokan, 1998] "Towards A Framework for Handling Disputes in Payment System" 1998. IBM System Journal http://www.almaden.ibm.com/technology/

[11][Benjamin Cox, 1998] "NetBill Security and Transaction Protocol" Usenix Workshop on Electronic Commerce 1998 Carnegie Mellon University

[12][Gennady Medvinsky and B. Clifford, 1993] "Netcash:A design for practical electronic currency on the internet", In processing of the First ACM conference and communication security, November, 1993

[13][HTML document, 1999] "Micro Payment Transfer Protocol" http://www.w3.org/TR/WD-mptp/

[14][Digicash, 1999] "How ecash Work Inside", http://www.digicash.com/ecash/docs/index.html 1999

[15][David Chaum,] "Online Cash Check",http://www.digicash.com/news/achive/online.html/ 1999

[16][Paul-Andre Pays, 1996] "An intermediation and payment system technology" computer network and isdn system 28(1996) pp1197-1206

[17][Rajashekar Kailar 1996] "Accountability in Electronic Commerce Protocols" IEEE Transaction on Software Engineering, Vol 22 No.5 May 1996 pp313-328

[18][Versign, 1999] "http://www.verisign.com/"

[19][Carl E. Landwehr, 1997] "Security Issues in Networks with Internet Access" proceedings of ieee. vol.85 no.12 september 1997 pp2034-2051

[20]PMAP, 1999]"Scaleable, Secure Cash Payment for WWW Resources with the PayMent Protocol Set" Michael Peirce, Donal O'mahony 1999 "http://www.w3.org/Conferences/WWW4/Papers/228/

저자소개

조성진
1988 광주대학교 학사
1998 대구효성가톨릭대학교 전산통계학과 석사과정
현재 선린대학 겸임교수
관심 분야 : 인공 지능, CALS/EC, 에이전트

허철희
1984 광운대학교 전자계산과 학사
1987 명지대학교 전자계산과 석사
1998 대구효성가톨릭대학교 전산통계학과 박사과정
현재 성덕대학 전산정보처리학과 전임강사
관심분야 : 전자상거래, 암호학, 인공지능

정환묵
1972 한양대학교 전자공학과 공학사
1987 인하대학교 대학원 이학박사
1986.12.~1987.12. 日本 東京大學 정보과학과 객원연구원
1995. 2.~1996. 2 日本 明治大學 情報科學科 객원교수
1998. 1.~현재 한국퍼지 및 지능 시스템 학회 부회장
1984. 3.~현재 대구효성가톨릭대학교 공과대학 전자정보공학부 교수
관심 분야 : 인공 지능, 퍼지 논리, 다치 논리, 지능시스템 공학, CALS/EC