컴퓨터 부착용 신용카드 조회기에 기반한 전자지불 승인시스템의 설계 및 구현

장 시 웅[†] · 신 병 철[††] · 김 광 백[†††]

요 약

현재 전자상거래에서 가장 많이 사용되는 전자지불 방법은 신용카드를 이용한 방법이다. 그러나 신용카드를 이용한 결제는 해킹과 spoofing 등의 방법을 통해 신용카드 정보를 해킹 당할 수 있다. 본 연구에서는 종래의 전자지불 승인의 약점을 개선하기 위한 방법으로 인터넷상에서도 신용카드를 소지한 사람만이 신용카드 결제를 수행할 수 있도록 하는 방법을 제시하고, 이를 위한 컴퓨터 부착용 신용카드 조회기와 전자지불 시스템을 구현하였다. 본 연구에서 구현한 전자지불 승인시스템은 Windows-2000, Windows-98 및 Windows-me에서 동작 가능하며, 75명 이상의 동시 사용자를 지원할 수 있다.

Design and Implementation of an Electronic Payment System based on a Credit Card Reader

Si-Woong Jang[†] · Byoung-Chul Shin[††] · Kwang-Baek Kim[†††]

ABSTRACT

Nowadays, most payments in electronic commerce are processed by using credit card. The problem with credit card payment system is that customer's credit card information may be hacked by key board hacking and spoofing. In this study, we propose a method that only credit card holders can pay with the credit card even on the Internet, which solves enhance drawbacks of conventional payment methods. We implemented the computer-attachable credit card reader and a payment system. The payment system implemented in this study can be operated on many systems such as Windows-2000, Windows-98 and Windows-me, and could support 75 concurrent users.

키워드 : 전자지불(Electronic Payment), 신용카드(Credit Card), 인터넷(Internet), 암호화(Encryption), 복호화(Decryption)

1. 서 론

전자상거래는 기업간 또는 기업과 개인, 정부간에 컴퓨터 네트워크를 통해서 이루어지는 다양한 거래를 하는 것으로 정보기술의 통합적 활용을 통하여 기업의 경영 효율을 높이고 국제시장에서 경쟁력을 강화할 수 있는 수단이다[1].

이러한 전자상거래가 활성화되기 위해서 해결되어야 하는 중요한 사항이 전자지불(Electronic Payment) 문제이다. 전자지불에서는 사용자가 네트워크를 통하여 안전하게 상품을 구매할 수 있도록 보안성 문제가 해결되어야 함은 물론, 실제 전자상거래 과정에서 생길 수 있는 문제점들을 해결할 수 있도록 설계되어야 한다. 현재 전자상거래에서 가장 많이 사용되는 전자지불 방법은 신용카드를 이용한 방법이나 타인의 신용카드 번호와 유효기간을 알면 신용카드가 없어도 쇼핑몰을 통해 물건 구입을 도용할 수 있는 약점을 가진다. 또한 키보드를 통해 인터넷 환경에서 신용카드 번호를 입력할 경우 키보드 해킹과 spoofing 등의 방법을 통해 신용카드 정보를 해킹 당할 수 있다. 본 연구에서는 종래의 전자지불 승인의 약점을 개선하기 위한 방법으로 인터넷 상에서도 신용카드를 소지한 사람만이 신용카드 결제를 수행할 수 있도록 하는 방안을 제시하고 이를 구현하였다.

전자지불 설계 과정에서 고려되어야 할 대표적인 요소들은 안정성, 이중사용 방지, 분쟁해결성, 효율성, 사생활 보호 등이다[2, 3]. 즉 안정성은 프로토콜이 외부의 공격으로

※ 본 연구는 과학기술부 · 한국과학재단 지정, 부산광역시 지원 지역협력 연구센터인 동의대학교 전자세라믹스연구센터의 지원(2000학년도 동의대학교 자체 학술연구조성비 포함)에 의한 것입니다.
† 종신회원 : 동의대학교 컴퓨터통계학과 교수
†† 정 회 원 : 동의대학교 신소재공학과 교수
††† 정 회 원 : 신라대학교 컴퓨터공학과 교수
논문접수 : 2002년 1월 29일, 심사완료 : 2002년 5월 9일

부터 안전해야 한다는 가정이 있어야 하며, 이중사용 방지는 같은 지불 데이타를 두 번 사용할 수 없어야 한다는 것이며, 분쟁해결성은 지불과정에서 생길 수 있는 분쟁 유형에 대해 효과적으로 대처할 수 있어야 한다는 것이다. 효율성은 지불 처리비용이 저렴해야 한다는 것이고, 사생활 보호는 거래 과정에서 거래자의 개인정보가 노출되는 것을 막을 수 있어야 한다는 것이다[4, 5].

인터넷상에서 거래가 성립하기 위해서는 대금지불에 관한 안전성과 확실성이 보장되어야 한다. 따라서 성공적인 전자상거래의 실현을 위해서는 반드시 대금지불에 관한 여러 방안이 강구되어야만 한다. 안전한 전자지불 보안을 위해 여러 기관에서 많은 연구가 활발하게 진행 중인 상태이다. 전자상거래에서 활용하고 있는 전자지불의 형태로는 전자화폐(Electric Cash), 스마트카드를 통한 결제, 신용카드를 통한 전자결제, 제3자 결제방식 등이 있다[6, 7].

본 논문에서는 안전한 전자지불을 구현하기 위해 컴퓨터 부착용 신용카드 조회기에 기반한 전자지불 승인시스템을 설계하고 구현하였다. 신용카드를 사용할 경우 카드번호 및 거래관련정보를 전송하여야 하기 때문에 철저한 보안이 전제가 되어야 한다. 따라서 전송할 정보에 대한 암호화를 통해 데이터의 보안에 만전을 기하여야 한다. 기존에는 이를 위해 마스터카드, 비자 등의 대규모 신용카드회사와 금융회사를 중심으로 마련된 SET을 이용해서 암호화 작업을 하는 실정이다. 그러나 본 논문에서는 컴퓨터 부착용 신용카드 조회기를 이용하여 하드웨어에서 암호화 작업을 마친 후 승인 서버에 넘겨주는 방식을 이용하므로 기존의 소프트웨어로만 암호화 작업을 한 것보다 훨씬 암호화를 안전하고 효율적으로 수행할 수 있다.

2. 전자지불 승인시스템의 관련 연구

전자지불 승인시스템에 대한 연구는 지금까지 다양하게 이루어져 왔으며, 실제로 전자지불 승인시스템에 대한 서비스도 많이 이루어지고 있다. 기존에 연구되어진 전자지불 승인시스템의 사례들을 살펴보면 데이콤 전자지불 시스템, EasyPay[8], PayGate[9], SET/SSL Protocol을 이용하는 방식, PGP를 이용한 WWW 기반에서의 전자지불 프로토콜, 웹 브라우저와 CGI 프로그램 사이의 보안통신 방법, 독립된 고유번호 서비스를 이용한 전자화폐 대금 결제 시스템 등이 있다.

데이콤 전자지불 시스템은 PC Banking 서비스를 이용하는 천리안 PPP 고객을 대상으로 하여 서비스를 처음 개시하였으며, 1999년 9월 국내 전 신용카드에 대한 대금결제 서비스를 개시하였다. 해외발행 신용카드(VISA, MASTER, JCB, AMEX, 다이너스 카드)에 대한 대금결제서비스도 제공하고 있으며 인터넷상의 계좌이체 대금결제서비스를 1999년 9월 데이콤 Shopplaza에 처음으로 적용하여 실시하였다. 이 시스템은 SET Protocol을 적용한 신용카드 전자지불 시스템을 기반으로 하고 있다[12].

EasyPay는 카드소지자와 지불 GateWay 전구간을 128bit SSL로 구현하였으며, 세계적으로 공인된 암호화 Key를 사용하고 있으며 별도의 인증(Certificate)을 필요로 하지 않는다는 장점이 있다. PayGate는 신용카드 전자지불 시스템으로서 실시간으로 VAN을 거쳐 신용카드 승인을 획득하는 방식의 서비스를 제공하고 있으며, 신용카드 승인을 획득한 이후에도 지불을 취소할 수 있는 서비스를 제공하고 있다. 이 시스템의 지불 데이터는 128bit의 암호 알고리즘으로 보호하고 있다.

SET/SSL Protocol을 이용하는 방식은 SET과 SSL을 결합하여 전자지불 보안을 높인 것이다. 즉, SET은 각각의 트랜잭션(transaction)에서 확인 수행하는 것으로 은행카드 트랜잭션(bank card transaction)에 적합하다. 그리고 SSL은 session-oriented이고 세션 중에는 endpoints를 인증하는 역할을 한다. 이러한 결합 구조로 인하여 SET의 인증 부하(authentication load)가 감소하는 장점을 가지고 있다[13].

PGP를 이용한 WWW 기반에서의 전자지불 프로토콜은 신용카드를 이용하는 시스템에서 독립적으로 작동하고, 공개키 인증이 현실적이어서 단지 전자우편이라는 제한된 분야에만 사용되는 것은 아니다. 암호화와 전자서명 등을 필요로 하는 분야에 얼마든지 응용될 수 있으며 PGP를 암호화모듈로 사용해 인터넷에서 전송되는 모든 메시지들에 대해서 전자서명과 암호화 작업을 수행할 수 있도록 했다. 현재 가장 많이 사용되는 신용카드를 이용한 전자지불 방식을 기반으로 하고 있으므로 이를 이용하여 새로운 프로토콜을 만들 수 있다[12].

웹 브라우저와 CGI 프로그램 사이의 보안통신 방법은 웹 브라우저와 웹 서버 사이의 통신뿐만 아니라 Internet에서도 보안을 지원할 수 있는 시스템을 구현한 것이다. 이 시스템은 PKI 기반의 SSL을 이용하여 보안을 지원한 것이 특징이다. 그러나 브라우저 사용자와 CGI 개발자는 같은 PKI 제품을 사용해야 한다는 제약사항이 있는 것이 단점이다[13].

독립된 고유번호 서비스를 이용한 전자화폐 대금 결제 시스템은 이중 지불검사를 담당하는 고유번호 관리 서버를 은행과 독립적으로 구성하여 익명성을 보장하며, 고유번호 관리 서버는 데이터베이스에 현재 통용중인 전자화폐들의 고유번호를 저장하고, 고유번호 데이터베이스의 크기는 일정하게 유지되어 이중지불 검사를 효율적으로 수행하는 것이 이 시스템의 특징이다. 그러나 전자화폐가 다량 존재하게 되어 관리가 어렵고 대금 지불시 지불에 필요한 전자화

폐를 직접 선택해야 하는 단점이 있다[14].

지금까지 살펴본 연구에서의 전자지불 승인시스템은 보안에 상당한 비중을 두고 있으며, 다음으로 사용자에게 간편하게 서비스하는 것이 요구되고 있다. 보안은 대부분 SSL이나 SET과 같은 Protocol을 이용하여 암호화를 하였으며, 간혹 특정의 CGI를 이용하거나 PGP와 같은 메일 기반의 색다른 방법을 이용하는 것도 있지만 현실적으로 적용하기에는 미흡한 부분이 많이 있다. 그리고 현재 사용중인 사용자 인터페이스의 경우는 신용카드 번호를 사용자가 직접 입력해야 하는 불편과 함께 키보드로 신용카드 번호를 입력시 후킹 또는 해킹을 당할 수 있다. 즉 사용자의 PC가 이미 해킹 당한 컴퓨터이면 신용카드 입력시 외부로 누출되어 심각한 보안상의 위험이 따를 수 있다.

본 논문에서는 전자상거래에서 신용카드 결제 문제를 해결하기 위해 개인용 컴퓨터에 신용카드 조회기를 부착하여 인터넷에서 신용카드 결제시에도 신용카드를 소지한 사람만이 신용카드를 결제할 수 있도록 구현한 것이다. 하드웨어와 소프트웨어를 결합한 전자지불 승인시스템으로 보안과 사용자 인터페이스 부분에 중점을 두었다. 즉 신용카드 조회기를 통하여 간편하게 신용카드 번호를 읽어서 암호화를 시킨다. 이렇게 암호화된 신용카드 정보를 전자지불 승인 서버에 보내어 복호화한 후 금융기관이 요구하는 형태로 보내어 전자지불에 대한 승인을 얻는다.

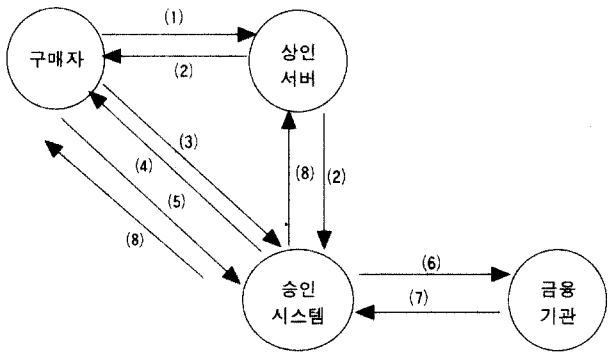
3. 전자지불 승인시스템의 구조 및 설계

본 장에서는 전자지불 승인시스템의 전체적인 구성, 암호화 및 복호화, 전자지불 승인시스템의 컴포넌트 간 통신 관계에 대해 기술한다.

3.1 전자지불 승인시스템과 외부 요소와의 관계

전자지불 승인시스템의 외부 요소는 구매자, 상인 서버 및 금융 기관으로 이루어져 있다. 전자지불 승인시스템과 외부요소와의 자료흐름은 (그림 3)과 같이 나타낼 수 있다. 구매자는 상품을 구매하거나 판매하는 주체로서 구매자는 쇼핑 도중에 구입하려는 물건을 선택한 후, 지불 페이지 요청을 한다. 상인 서버는 물건을 판매하는 주체로서 일반적으로 쇼핑몰을 들 수 있다. 승인시스템은 고유번호 관리, 신용 카드 번호 인증 및 관리를 수행한다. 금융기관은 신용카드의 신용내역 판별 및 결과를 전송한다. 구체적인 자료흐름 과정은 다음과 같다.

(그림 3)에서 처럼 구매자는 상인 서버에서 물건을 선택한 후 대금을 지불하기 위해서 지불페이지를 요청한다(1). 상인서버는 구매자의 구매내역을 제시하고 승인시스템으로의 연결 버튼을 제공한다(2). 구매자가 선택한 상품 목록을 확인하면 연결버튼을 통하여 승인시스템에 연결된다(3). 승인시스템은 난수번호를 생성한 후 난수번호를 담아 카드 삽입 요구 메시지를 전송한다(4). 사용자가 신용카드를 조회기에 삽입하면 조회기는 승인시스템으로부터 받은 난수번호를 이용하여 암호화키를 구성한 후, 구성된 암호화키로 신용정보를 암호화한 후 승인시스템에 전송한다(5). 승인시스템은 사용자의 조회기로부터 받은 암호화된 신용정보를 복호화한 후 다시 금융기관이 원하는 암호체계로 암호화한 후 금융기관에 신용정보를 전송한다(6). 금융기관은 승인시스템으로부터 받은 신용정보를 체크하여 구매자의 신용내역 판별 결과를 승인시스템에 전달한다(7). 승인시스템은 금융기관으로부터 받은 판별 내역을 확인한 후 전자 영수증을 발행하여 구매자와 상인서버에 전자 영수증을 전송한다.

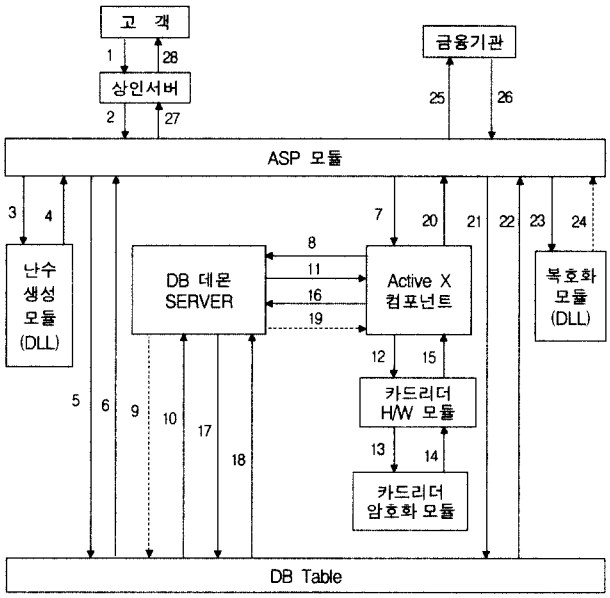


(그림 3) 전자지불 승인시스템과 외부요소와의 자료흐름

3.2 전자지불 승인시스템의 구조

전자지불 승인시스템은 ASP 모듈, 난수생성 모듈(dll), 복호화 모듈(dll), Active-X 컴포넌트, DB 데몬서버로 구성되어 있다. (그림 4)는 전자지불 시스템의 컴포넌트간 호출관계 및 파라메터 전달 관계를 나타낸다. 각 컴포넌트간의 상세한 통신 관계를 설명하면 다음과 같다.

고객이 물건을 구입한 후 카드결제를 선택하면(1), 상인 서버에서는 전자지불 승인시스템의 ASP 모듈에 카드 결제를 요구한다(2). 전자지불 승인시스템의 ASP 모듈은 결제를 요구한 고객의 카드정보를 암호화하기 위해, 난수생성 모듈을 호출하여 난수 값을 생성하고 난수 값을 DB Table에 저장한다(3, 4, 5). 이후, ASP 모듈이 Session ID를 파라메터로 하여 Active-X 컴포넌트를 호출하면(7), Active-X 컴포넌트는 Session ID를 파라메터로 하여 DB 데몬 서버에 난수 값 읽기를 요구한다(8). DB 데몬 서버는 DB Table에서 읽은 난수값을 Active-X 컴포넌트에 전달하고(9, 10, 11), Active-X 컴포넌트는 난수 값을 카드리더 모듈에 전달한다(12). 카드리더 모듈은 난수 값과 읽은 카드 정보를 카드 리더 암호화 모듈에 전달하여 암호화하고(13, 14), 암호화된 카드 정보를 Active-X 컴포넌트에 전달한다(15).



(그림 4) 전자지불 시스템의 컴포넌트간 관계

Active-X 컴포넌트는 암호화된 카드정보를 DB 데몬 서버에 전달하여 DB Table에 저장한다(16). 이후, ASP 모듈은 암호화된 카드정보를 DB Table에서 읽어(21, 22), 암호화된 카드정보와 난수를 파라메터로 복호화 모듈을 호출하여 복호화를 수행한 후(23, 24), 금융기관이 원하는 형태로 카드정보를 제출하여 금융기관의 조회결과를 받은 후(25, 26), 상인서버와 고객에게 조회결과를 전달한다.

3.3 전자지불 승인시스템의 암호화 및 복호화

본 절에서는 신용카드 조회기에서의 암호화 및 복호화에 대해 설명한다. 카드 정보를 암호화하는 방법은 quotient ring [15]에 기반한 행렬을 사용하며, 암호해독의 난이도를 높이기 위해 암호화 대상 모든 데이터에 대해 서로 다른 암호화키를 적용하는 완전 암호체계를 추구하였다. 암호화 모듈의 함수적 특성을 배제하기 위해 암호키 값은 서버에서 전송해 온 난수 값과 암호 모듈 자체에서 생성한 난수 값을 조합하여 사용한다. quotient ring에 기반한 행렬을 이용하면, 구현이 단순하여 RAM의 용량이 적은 one-chip 프로세서에 적용이 유리하며, 다양한 행렬 연산으로 복잡도를 높일 수 있는 특징이 있다.

3.3.1 호스트 시스템에서 난수 값의 구성

호스트 시스템에서 난수 값은 (그림 5)의 형태로 생성한다.

b	m	n	matrix의 컴포넌트 bits

(그림 5) 난수 값의 구조

(그림 5)에서 b는 quotient ring의 base, m은 행렬의 크기, n은 행렬의 개수를 나타낸다.

여기서, b, m, n은 각각 8bits로 구성하고, 난수 값의 총 비트 수를 T_t, 암호키를 구성할 행렬 요소(matrix element) 1개의 비트 수를 T_e라 할 때, 1개 컴포넌트의 비트 수 T_e는 식 (1)과 같다.

$$T_e = \lfloor \frac{T_t - 24}{m^2 n} \rfloor \geq 4 \qquad (1)$$

난수의 크기를 128~1024 bits로 할 때, 식 (1)에 따른 행렬의 구조는 <표 1>과 같다. <표 1>에서 볼 수 있는 것처럼, 난수의 크기가 128 bits일 때, 구성할 수 있는 행렬의 수는 2×2 행렬 6개, 3×3 행렬 3개, 4×4 행렬 1개이다.

〈표 1〉 암호 행렬의 구성

난수값의 크기	행렬의 최소개수			행렬의 최대개수		
	m	n	T_e	m	n	T_e
128 bits	2	1	28	2	6	4
	3	1	12	3	3	4
	4	1	7	4	1	7
256 bits	2	1	60	2	15	4
	3	1	26	3	6	4
	4	1	15	4	3	5
512 bits	2	1	124	2	31	4
	3	1	55	3	13	4
	4	1	31	4	7	4
1024 bits	2	1	252	2	63	4
	3	1	112	3	28	4
	4	1	63	4	15	4

3.3.2 신용카드 조회기에서의 암호화

신용카드 조회기가 서버로부터 난수 값을 전달받으면, 신용카드를 체크하라는 의미로 해석하고 신용카드 체크 LED를 ON시킨다. 이 때, 신용카드 조회기는 사용자가 체크한 카드 정보를 읽은 후, 자체에서 난수를 발생하여, 서버에서 받은 난수 값과 신용카드조회기에서 생성한 난수 값을 조합하여 암호화 행렬을 구성한다. 여기서, 암호화 행렬 구성 시, 서버에서 받은 난수 값은 행렬 요소(matrix element)로 이용하고, 자체 생성 난수 값은 암호 행렬을 구성하기 위한 조합 정보로 이용한다.

예를 들어, quotient ring의 base가 102, 행렬의 크기가 m = 2, 행렬의 개수가 n = 4이고, 난수 값이 128bits로 구성되어 있으며, 암호화할 평문의 값이 "123456781234"인 경우를 가정하자. 그러면, 행렬의 크기가 2×2인 정방 행렬이므로 암호화하고자 하는 평문의 값은 다음과 같이 6개의 1×2 행렬 값으로 구분하여 표현할 수 있다.

{(31, 32), (33, 34), (35, 36), (37, 38), (31, 32), (33, 34)}

여기서, 난수 값의 크기를 128bits로 가정하였으므로 식 (1)에 의해 행렬 요소의 값은 9bits로 구성되며, 행렬 요소가 가질 수 있는 값은 0~512가 된다. 따라서, 다음과 같은 행렬을 가정할 수 있다.

$$A=\begin{pmatrix}76 & 55\\ 9 & 77\end{pmatrix},\ B=\begin{pmatrix}252 & 169\\ 95 & 15\end{pmatrix},\ C=\begin{pmatrix}77 & 18\\ 68 & 73\end{pmatrix},\ D=\begin{pmatrix}6 & 108\\ 18 & 219\end{pmatrix} \quad (2)$$

위의 행렬은 base가 102인 상태에서 역행렬이 존재하는 행렬들이다. 개별 행렬은 서버에서 보낸 난수 값을 이용하여 식 (2)와 같이 구성할 수 있으며, 평문의 각 쌍을 암호화할 암호행렬은 신용카드 조회기에서 생성한 난수값을 이용하여 생성한다. 이를 테면, (31, 32)의 평문쌍을 암호화할 암호행렬은 A, B, C의 3개 행렬 곱(A×B×C)으로 구성될 수 있고, (33, 34)의 평문쌍을 암호화할 암호행렬은 A×A×D로 구성될 수 있다. 각 평문을 암호화할 행렬의 쌍은 신용카드 조회기에서 생성한 난수 값을 이용하여 구성된다. 따라서, 모든 평문의 쌍에 대해 서로 다른 암호 행렬을 사용함으로써, 평문의 길이가 길어도 암호 행렬을 추정하는 것은 어렵다. 또한, 동일한 카드 정보에 대해 서버에서 동일한 난수 값을 보내도 항상 다른 암호값이 생성되므로 함수성이 없어 카드정보 해킹에서 안전하다. 신용카드 조회기에서 생성된 난수 값은 암호화된 정보에 은닉되어 서버로 전달된다.

3.3.3 호스트 시스템에서의 복호화

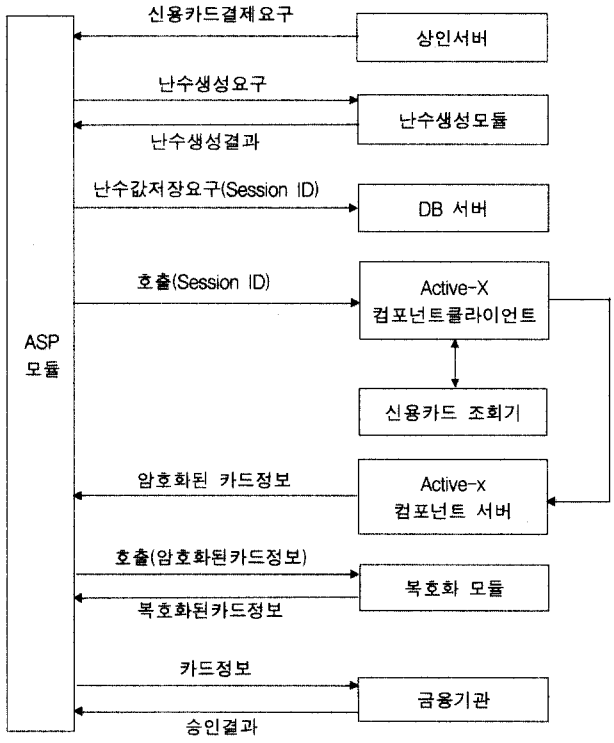
신용카드 정보가 신용카드 조회기에서 읽혀 암호화된 후, 암호문이 호스트시스템에 전달되면 호스트 시스템은 자체에서 보관하고 있는 난수 값과 암호문 내에 포함된 신용카드조회기의 난수를 이용하여 복호 행렬들을 구성하고, 암호문을 복호 행렬들을 이용하여 차례로 복호화한다.

3.4 전자지불 승인시스템의 컴포넌트간 통신관계

전자상거래 승인시스템은 Web 서버와 연동하여 수행하는 ASP 모듈, 암호키를 생성하기 위해 필요한 난수 생성 모듈(Randnum), 암호화된 카드정보를 복호화 하기 위한 복호화 모듈(Decrypt), 카드 단말기의 정보를 승인 서버로 보내기 위한 Active-X 컴포넌트 클라이언트 모듈, Active-X 클라이언트와 통신하고 데이타를 DB 서버에 저장하기 위한 Active-X 서버 모듈(DB 데몬 서버) 및 신용카드 정보를 읽기 위한 카드조회기 H/W 모듈로 구성되어 있다.

3.4.1 ASP 모듈

ASP 모듈은 상인서버로부터 신용카드 결제요구를 최초로 받는 승인시스템의 메인 모듈로서 난수생성 모듈(RandNum), Active-X 컴포넌트 모듈 및 복호화 모듈(Decrypt) 등을 호출하여 작업을 수행한다. ASP 모듈은 상인서버로부터 신용카드 결제요구를 받으면 난수 생성 모듈을 호출하여 해당 신용카드 정보를 암호화할 난수 값을 얻는다. 난수 값은 Session ID를 키로 하여 DB 서버에 저장하고, Session ID를 인수로 하여 Active-X 컴포넌트 클라이언트를 호출한다. 이때, Active-X 컴포넌트 클라이언트는 Session ID를 키로 하여 ASP 모듈에서 저장한 난수값을 읽어 카드조회기에 전달한다. 그러면, 카드조회기는 전달받은 난수값을 가지고 암호키를 생성한 후, 카드로부터 읽은 정보를 암호화하고 Active-X 컴포넌트 클라이언트 모듈에 암호화된 카드정보를 전달한다. 이후, Active-X 컴포넌트 클라이언트 모듈은 암호화된 카드정보를 Active-X 컴포넌트 서버 모듈에 전달하고, Active-X 서버 모듈은 전달받은 암호화된 카드정보를 Session ID를 키로 하여 DB 서버에 저장한다. ASP 모듈은 Session ID를 키로 하여 암호화한 카드정보를 DB 서버로부터 읽은 후 복호화 모듈을 호출하여 복호화된 카드정보를 얻은 후 관련정보를 금융기관에 보내 승인결과를 얻는다.

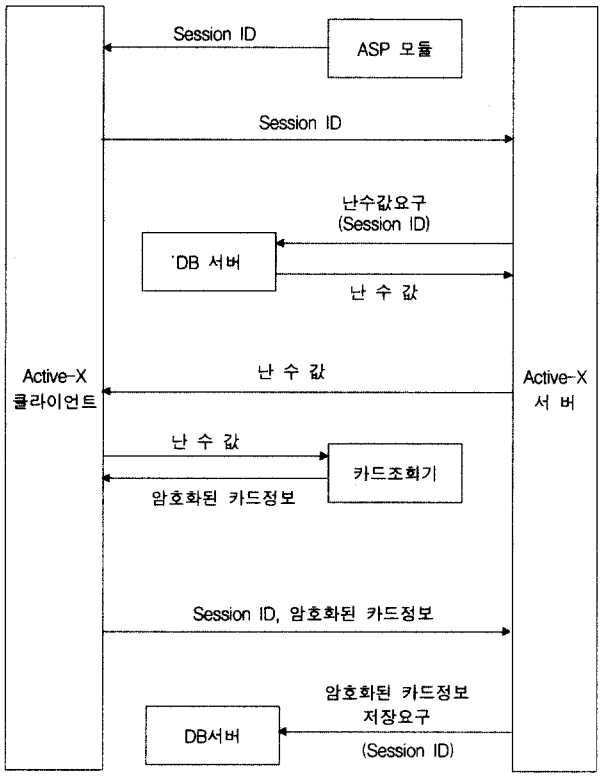


(그림 6) ASP 모듈 흐름도

3.4.2 Active-X 컴포넌트

Active-X 컴포넌트는 Active-X 클라이언트 모듈과 Active-X 서버 모듈로 구성되어 있으며, Active-X 서버와 클라이언트 사이에는 TCP/IP 프로토콜로 자료교환 및 교신을 한다. Active-X 클라이언트 모듈은 ASP 모듈과 카드조회기 사이의 교량 역할을 수행하는 모듈로서 ASP 모듈로

부터 Session ID를 전달받아 그것을 서버에 전달한다. 서버에서는 전달받은 Session ID를 이용하여 DB 서버에 접근하여 난수값을 읽은 후 클라이언트에게 난수값을 전달한다. 클라이언트는 서버로부터 받은 난수값을 카드조회기에 전달하고 암호화된 카드정보를 얻고, 암호화된 카드 정보와 Session ID를 서버에게 전달한다. 서버는 Session ID를 키 값으로 하여 DB 서버에 저장한 후 결과를 ASP 모듈에 리턴한다.



(그림 7) Active-x 컴포넌트

3.4.3 난수 생성 모듈

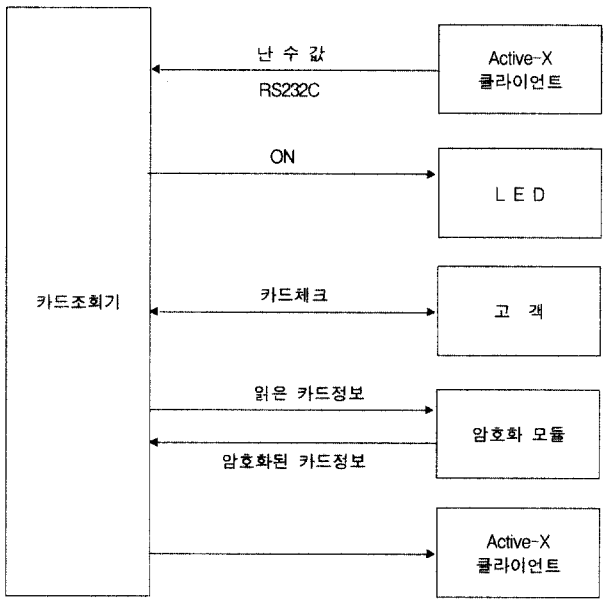
카드조회기로부터 읽은 카드정보를 암호화하기 위하여 암호화 키 값이 필요하다. 본 연구에서는 암호화 기법으로 quotient ring에 기반한 행렬연산을 사용하였는데, 암호 행렬을 구성하기 위해 난수값이 필요하다. 암호 행렬 구성 시에는 역행렬이 존재하는 난수값을 생성하여야 한다. 난수 생성 모듈은 이상의 특성을 만족하는 난수를 발생하여 ASCII 값으로 변환한 후 결과를 ASP 모듈로 리턴한다.

3.4.4 카드조회기

카드조회기는 Active-X 컴포넌트와의 통신과 암호화 모듈로 구성되어 있다. 카드조회기와 Active-X 컴포넌트간의 통신은 Serial을 이용하여 자료를 전송한다. 카드조회기는 RS232C 프로토콜을 이용하여 난수 값과 암호화된 카드정보를 통신하게 된다. Active-X 클라이언트로부터 난수 값을 전달받으면 카드를 읽으라는 명령으로 해석하고 카드조회기의 LED를 ON시킨다. 카드조회기의 LED가 ON되면 고객은 카드를 체크하게 되는데, 이때 카드로부터 카드 정보를 읽는다. 그리고, 난수 값을 파라미터로 카드조회기 암호화 모듈을 호출하여 읽은 카드관련 정보를 암호화한 후 암호화된 정보를 Active-X 클라이언트에 전달한다.

3.4.5 복호화 모듈

ASP 모듈로부터 암호화된 정보나 난수값을 받아 역 행렬을 구성하고, 이를 암호화된 정보에 적용하여 복호화된 카드 정보를 생성한다. 카드 정보의 각각의 바이트에 대해서로 다른 역 행렬을 적용한다.



(그림 8) 카드 조회기

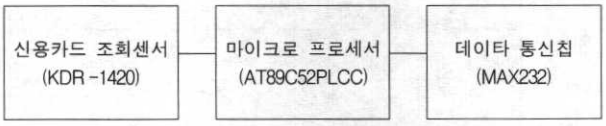
4. 전자지불 승인시스템의 구현

이번 장에서는 전자지불 승인시스템의 하드웨어 및 소프트웨어의 구현 내용을 설명한다.

4.1 신용카드 조회기

카드조회기는 별도의 전력 없이 마우스에서 사용하는 전력을 이용한다. 마우스 포트에 카드 조회기를 삽입하여 전력을 얻고 카드조회기의 마우스 포트에 마우스를 연결함으로써 마우스를 동시에 사용하여 별도의 전력이 필요하지 않도록 제작하였다. 데이터 전송은 RS232C를 이용하여 자료를 전송한다. 암호화 자료는 최대 128Bytes이내의 패킷 자료로 구성되어 있다. RS232C의 최대 전송속도는 19200 BPS이므로 신속하게 패킷을 전송할 수 있다. 신용카드 조회센서(KDR-1420, KDE Co.)는 신용카드의 정보를 읽고,

마이크로 프로세서(AT89C52PLCC)는 승인서버에서 받은 난수값을 이용하여 신용카드에서 읽은 정보를 암호화한다. 암호화가 수행된 후 데이터 통신 칩(MAX232)은 암호화된 데이터를 승인서버에 보낸다. (그림 9)는 신용카드 조회기의 블록도를 보여 준다.



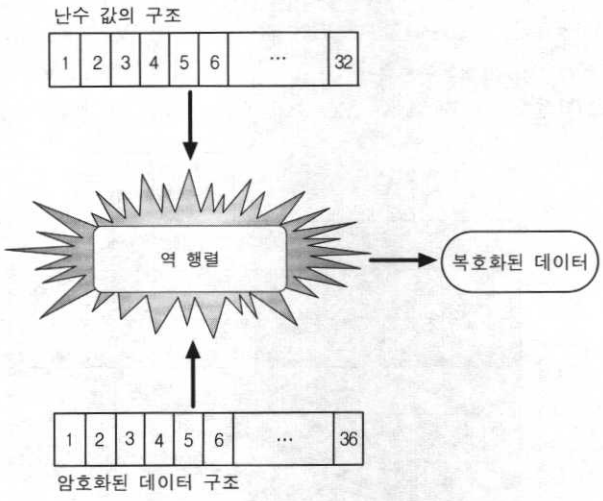
(그림 9) 신용카드 조회기의 블록도

4.2 승인시스템의 내부 구조

승인시스템은 내부적으로 보안성을 높이기 위해서 암호화와 복호화 과정을 통해서 신용카드 정보를 보호하고 있다. 이러한 처리는 DB 서버와 연동하여 처리되고 있으며 구체적인 내용은 다음과 같다.

4.2.1 복호화 처리 구조

암호화된 자료를 복호화하는 모듈인 Decypt.dll은 Visual C++로 개발된 확장 DLL이다. Decrypt.Dll은 Register에 등록되어 Web 서버와 연동하여 실행되는 것으로 데이터베이스 서버에 저장되어 있는 난수값(RandomNumber)과 암호화된 데이타를 파라미터로 넘겨받아 역 행렬을 이용하여 원래의 카드 정보인 복호화된 데이타를 생성한다.



(그림 10) 암호화 데이타 구조

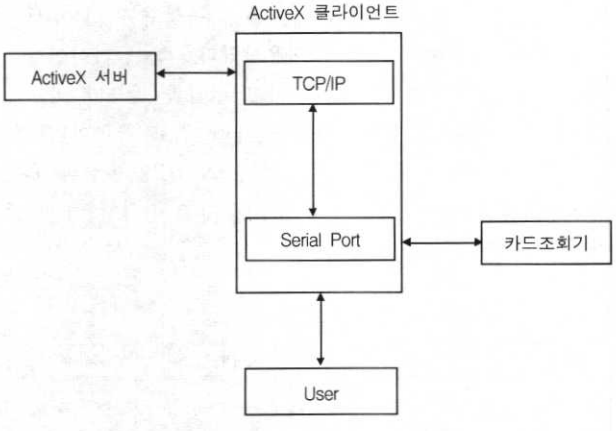
4.2.2 ActiveX 서버 처리 구조

ActiveX 서버와 카드 조회기에 접근하기 위한 ActiveX 클라이언트를 살펴보자. ActiveX 서버는 내부적으로 데이타베이스 서버에 접근하기 위한 모듈과 Socket 모듈을 가지고 있다. 데이터베이스 서버에 접근하기 위한 모듈은 기본적으로 ODBC를 이용하여 정보를 저장하거나 불러오게 된다. 이러한 처리는 DBSet 이라는 클래스에서 처리하게 되는데 구성은 <표 2>와 같다.

〈표 2〉 CDBSet의 데이터 구조

Session_ID	Enterprise_ID	Random_num	Security_data	Original_data
세션 ID	카드정보가 저장될 ID	난수값	암호화된 카드정보	복호화된 카드정보

Session_ID는 현재 생성된 세션으로 카드 정보를 추출하기 위해서 이용하는 것이고, Enterprise_ID는 카드정보의 ID를 저장하기 위한 공간이며, Random_num은 CRandom.dll인 확장 DLL에 의해 생성된 난수번호를 저장하기 위한 필드이며, Security_data는 카드 리드기로부터 읽어와 저장하게 되는 암호화된 카드 정보이다. 그리고 Original_data는 복호화된 정보가 저장될 공간이다. 다음으로 Socket 모듈을 살펴보면 Socket은 ActiveX 클라이언트와 접속하기 위한 모듈로서 클라이언트 클래스에서 ActiveX 클라이언트와 Packet 교환을 하기 위한 처리를 한다. 이러한 처리는 TCP/IP 프로토콜을 이용하여 수행되는 모듈로서 소켓을 생성하며 Receive 함수를 이용하여 ActiveX 클라이언트로부터 Session_ID의 값을 확인한 후 PostMessage를 이용하여 해당하는 정보를 받아들인다. 이렇게 받아들인 정보는 데이터베이스 서버 모듈에게 전달되어 저장된다.



(그림 11) ActiveX 클라이언트 Data 구조

4.2.3 ActiveX 클라이언트 처리 구조

ActiveX 클라이언트 모듈은 ActiveX 서버 모듈과 통신하는 부분과 카드 조회기와 통신하는 부분으로 구성되어 있다. ActiveX 서버와 통신하는 부분은 TCP/IP 프로토콜을 이용하므로 CAuthSocket 클래스를 이용하여 소켓을 생성한 후 서버 모듈과 통신하기 위해서 Connect 함수에서 ActiveX 서버와 연결하고, ActiveX 서버에서 제공하는 정보를 받기 위해서 Receive 함수를 이용하여, ActiveX 서버의 값을 받아들인다. 처음으로 서버와의 접속이 이루어졌을

때는 Session_ID를 확인한 후 난수값을 서버로부터 전송받는다. 이렇게 전송 받은 난수값은 카드조회기에 전송하게 된다. ActiveX 클라이언트와 카드조회기와 통신하는 부분은 Serial을 이용하여 자료를 전송한다. ActiveX 클라이언트는 카드조회기의 Port(COM1 or COM2)를 확인한 후 카드 조회기와 접속하기 위한 초기 설정을 한다. 카드조회기의 존재 유무를 체크한 후 ActiveX 클라이언트는 사용자가 ActiveX 클라이언트를 호출할 때까지 대기 모드에 들어간다. 사용자가 ActiveX 클라이언트를 체크를 하면 ActiveX 서버로부터 난수값을 전달받고 난수값은 다시 카드조회기로 전송하게 된다. 카드조회기는 난수값을 전달받으면 카드를 읽으라는 신호로 해석하고, 신용카드번호를 읽은 후 서버에서 전달받은 난수값과 자체 생성한 난수값을 이용하여 암호화한 후 암호화된 정보를 ActiveX 클라이언트에게 Serial Port를 이용하여 전달하게 된다.

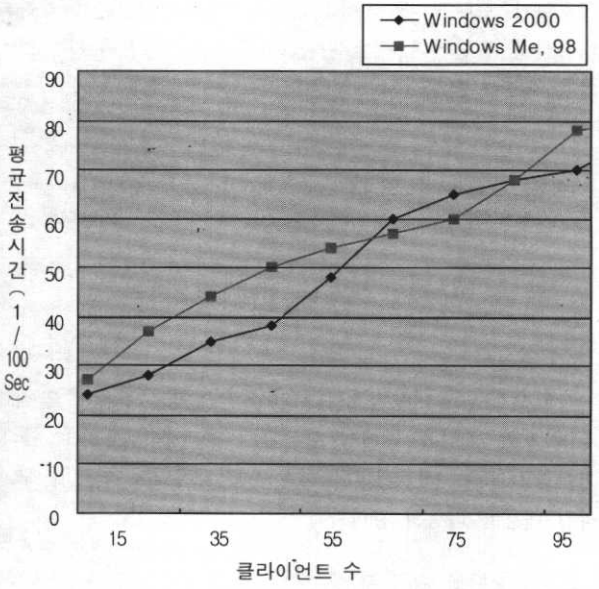
5. 성능 분석

본 연구에서는 신용카드 조회기를 이용한 전자지불 승인 시스템을 인터넷 환경에서 구현하였다. 구현한 전자지불 승인시스템을 테스트하기 위해 다수의 PC가 연결된 인트라넷 환경을 구성하였다. 성능 분석은 메인 서버(IIS 서버 + DB 서버)와 클라이언트를 <표 3>과 같은 환경으로 구성하였다. 여기서는, 서버의 성능분석을 위한 것이므로 클라이언트의 사양은 중요하지 않아서 기존에 구성되어 있는 사양과 운영체제 환경을 그대로 사용하였다. 즉, 클라이언트는 서버에 단순히 승인 요구만을 하고 결과를 받으면 되므로 시스템의 사양과 운영체제와는 관계가 없고 클라이언트의 수만 관련이 있다. 본 연구는 기본적으로 50명 이상의 동시 접속자수를 받아 들여서 처리할 수 있다는 가정 하에서, 클라이언트 접속 테스트를 하였다.

〈표 3〉 테스트 환경 구성표

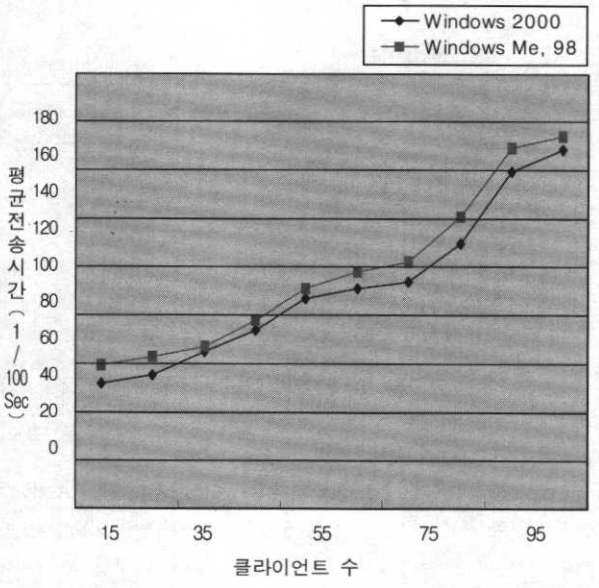
구 분	OS	CPU	수량	메모리
서버 (IIS + DB)	Windows 2000 서버	Pentium III 1Ghz	1	512M
클라이언트	Windows 2000	Pentium III 1Ghz	5	256M
	Windows Me	Pentium III 800 Mhz	20	256M
	Windows 98	Pentium II 600 Mhz	25	128M

Active-X 서버와 Active-X 클라이언트 사이의 통신은 전자지불 승인 서버와 신용카드 조회기가 장착된 사용자 PC와의 통신을 나타내므로 중요한 의미를 가진다. (그림 12)는 ActiveX 서버와 ActiveX 클라이언트간의 평균 전송 시간을 비교한 것으로 클라이언트의 환경 즉 Windows 2000제품군과 Windows Me, 98간에는 평균적으로 0.1초간의 차이를 보였다. 전체적으로는 클라이언트 수를 늘려감에 따라 0.8초까지 증가하는 현상을 보였다.



(그림 12) ActiveX 모듈에 대한 평균전송 시간 비교

(그림 13)은 전체 수행에서 클라이언트 수를 증가함에 따라 평균전송 시간의 변이와 운영체제간의 차이를 나타낸 것이다. 클라이언트 수를 늘려 가면 대체로 평균전송시간이 증가하는 것을 볼 수 있으며, Windows 2000제품군과 Windows Me, 98간의 약간의 차이를 나타내는 것을 볼 수 있다. 클라이언트 수를 15대로 하였을 때, 평균전송시간의 경우 Windows 2000은 0.5초 정도의 값을 보였고, Windows Me, 98은 0.6초 정도의 결과치를 보였다. 전체적으로 Windows 2000제품군과 Windows Me, 98은 대략 0.1초 정도의 차이를



(그림 13) 전체 수행에서 클라이언트 수에 대한 평균전송 시간 비교

보였다. 그리고 클라이언트수가 80이상일 경우에 평균전송 시간이 급격히 증가하는 현상을 보였다. 따라서, Windows 2000 서버를 탑재한 Pentium Ⅲ에서 안정적으로 처리할 수 있는 클라이언트의 수는 75 정도로 파악되었다. (그림 12)와 (그림 13)을 종합하여 보면 Active-X 모듈의 평균 전송 시간은 전체처리 시간의 50% 정도를 차지하고 있어 Active-X 모듈의 중요성을 보여 준다.

6. 결 론

본 연구에서는 종래의 전자지불 승인의 약점을 개선하기 위한 방법으로 인터넷상에서도 신용카드를 소지한 사람만이 신용카드 결제를 수행할 수 있도록 하는 방안을 제시하고, 컴퓨터 부착용 신용카드 조회기와 전자지불 승인시스템을 구현하였다.

컴퓨터 부착용 신용카드 조회기의 보안을 위해 암호화를 수행하였으며, 암호화된 신용카드 정보는 서버의 복호화 모듈에 의해 해독된다. 본 연구에서는 신용카드 조회기내에 암호화 모듈을 탑재하여 사용자가 신용카드를 읽는 즉시 암호화하여 컴퓨터에 전달함으로써 각종 해킹에도 안전하게 하였다. 신용카드 조회기는 서버로부터 부여받은 암호키와 자체 생성한 난수를 조합하여 신용카드로부터 읽은 정보를 암호화하고, 서버는 암호화된 신용정보를 받아 복호화를 수행한다.

연구를 통하여 구현한 전자지불 승인시스템의 성능 평가를 위해 인트라넷을 구축하고 한 대의 서버에 다수의 클라이언트를 동시에 접속하면서 성능을 평가하였다. 성능평가 결과, Windows 2000 서버를 탑재한 PentiumⅢ에 전자지불 승인서버를 탑재하였을 때, 클라이언트의 수가 75까지는 전체 처리시간이 서서히 증가되나 클라이언트수가 80이상일 경우에 처리시간이 급격히 증가하는 현상을 보여서, 안정적으로 처리할 수 있는 클라이언트의 수는 75 정도로 파악되었다.

전자상거래에 있어서의 철저한 보안 체계의 확립은 전자상거래의 활성화에 반드시 필요한 요소이다. 개발된 시스템은 홈쇼핑(home Shopping), 홈뱅킹(Home banking) 전자결제 시스템 등의 네트워크 상에서 이루어지고 있는 모든 사용자 인증 보안 시스템에 응용 될수 있다.

참 고 문 헌

[1] 송용욱, "지불기술, 시스템 동향", 인터넷백서, forthcoming.

[2] 손은경, 김태윤, "재사용 가능한 전자화폐 일련번호와 지불 트랜잭션 메커니즘", 정보과학회논문지, 제4권 제6호, pp.817-825, 1998.

[3] P Putland, J Hill, D. Tsapakidis, "Electronic payment systems," BT Technology Journal, Vol.15, No.2, pp.32-38, 1997.

[4] 박현동, 이은성, 송상헌, 강신각, 박적수, 류재철, "안전한 인터넷 전자지불 프로토콜의 설계 및 구현", 정보처리논문지, 제6권 제8호, pp.2145-2156, 1999.

[5] 주미리, 이보영, 양형규, 원동호, "전자상거래 인증 서비스를 위한 검증 가능한 자체인증 방식", 정보처리논문지, 제7권 제9호, pp.2894-2902, 2000.

[6] 김정은, 이형우, 김태윤, "스마트 카드를 사용한 오프-라인 전자지불 기법", 정보과학회논문지(A), 제26권 제11호, pp. 1363-1371, 1999.

[7] Michael Peirce, Donal O'Mahony, "Flexible Real-Time Payment Methods for Mobile Communications," IEEE Personal Communications, pp.44-55, Dec., 1999.

[8] 한국정보통신, http://www.kicc.co.kr.

[9] 페이게이트, http://www.paygate.net.

[10] Lucas de Carvalho Ferreira, Ricardo Dahab, "A scheme for Analyzing Electronic Payment Systems," Proceedings of the Fourteenth Annual Computer Security Applications Conference, pp.137-146, 1998.

[11] M. H. Sherif, A. Serhrouchni, A. Y. Gaid and F. Farazmandnia, "SET and SSL : Electronic payments on the Internet," Proceeding s of the Third IEEE Symposium on Computers and Communications, pp.353-358, 1998.

[12] 박현동, 강신각, 박성열, 류재철, "PGP를 이용한 WWW 기반에서의 전자지불 프로토콜 개발", 정보처리논문지, 제4권 제4호, pp.1046-1058, 1997.

[13] 이준석, "웹 브라우져와 CGI 프로그램 사이의 보안 통신을 지원하는 시스템 설계 및 구현", 정보처리논문지, 제6권 제3호, pp.641-653, 1999.

[14] 조지용 외 4인, "독립된 고유번호 서비스를 이용한 전자화폐 대금 결제 시스템의 설계 및 구현", 정보과학회논문지(C), 제4권 제5호, pp.728-737, 1998.

[15] J. B. Fraleigh, "A First course in Abstract Algebra," Addison-Wesley Publishing, Inc., 1994.

장 시 웅

e-mail : swjang@dongeui.ac.kr

1984년 부산대학교 계산통계학과 졸업(학사)

1993년 부산대학교 대학원 전자계산학과 졸업(석사)

1996년 부산대학교 대학원 전자계산학과 졸업(박사)

1986년~1993년 대우통신 종합연구소 주임연구원

1996년~현재 동의대학교 컴퓨터통계학과 부교수

관심분야 : 전자지불, 전자상거래, 데이터베이스

신 병 철

e-mail : shinbc@dongeui.ac.kr

1984년 연세대학교 세라믹공학과 졸업 (학사)

1986년 한국과학기술원 재료공학과 졸업 (석사)

1988년 한국과학기술원 재료공학과 졸업(박사)

1988년~1998년 포철 RIST 신소재연구부문 책임연구원

1996년~현재 동의대학교 신소재공학과 부교수

관심분야 : 전자세라믹스, 정보보안(CISA)

김 광 백

e-mail : gbkim@silla.ac.kr

1993년 부산대학교 대학원 전자계산학과 (이학석사)

1996년~1997년 동의공업전문대학 사무자동화과 전임강사

1999년 부산대학교 대학원 전자계산학과 (이학박사)

1997년~현재 신라대학교 컴퓨터공학과 조교수

관심분야 : 인공신경망, 영상처리, 생체신호처리, 퍼지시스템, 의료영상인식, 차량번호판인식