무선 전자상거래를 위한 전자영수증 발급 및 검증 기법 구현

박 근 홍† · 조 성 제††

요 약

최근 무선 단말기 이용이 증가함에 따라 무선 환경에서의 전자상거래가 활성화되고 있다. 따라서 유선 환경에서와 마찬가지로 무선 환경에서의 전자상거래 역시 소비자와 판매자간에 서로 신뢰할 수 있는 보안 및 영수증 발급 기술이 요구된다. 무선 전자상거래를 위해 WPKI, WML 전자서명 등 여러 방법들이 연구중이나 무선 단말기 성능 제한과 WAP 게이트웨이에서의 데이터 변환 시 발생하는 보안문제 등으로 인한 문제점이 여전히 존재한다. 본 논문에서는 무선 환경에서 전자상거래의 신뢰성을 보장하기 위해 무선 환경과 유선 환경을 접목시킨 전자영수증 발급 시스템을 제안한다. 본 시스템에서는 전자서명을 이용한 전자영수증을 발급함으로써 판매자와 소비자간의 신분 확인, 무결성 및 부인봉쇄 기법을 지원한다. 또한 신뢰할 수 있는 독립된 검증 서버를 도입하여 전자영수증을 검증하고 관리함으로써 전체 성능을 향상시켰다.

A System Implementation for Issuing and Verifying the Electronic Receipt for M-Commerce

Keunhong Park† · Seongje Cho††

ABSTRACT

As cell phone and PDA have been in common use recently, there is a growing tendency to utilize the mobile terminals for M-Commerce. The information security and the receipt of e-trade are very important to support reliable digital transactions in wireless environment as in wired environment. Even though some work such as WML digital signature and WPKI has been studied for M-Commerce, there are several problems on the aspects of the functional limitation of the mobile terminals and the unsecure data transformation of WAP gateway. In this study we have designed and implemented a prototype system of issuing and verifying the electronic receipt that guarantees authentication, data integrity and non-repudiation for secure mobile e-commerce. Moreover, we have enhanced the system performance by letting the trusted independent server verify and manage the electronic receipt.

키워드 : 전자영수증(Electronic Receipt), 무선 전자상거래(M-Commerce), Wpki(Wireless Public Key Infrastructure), 전자서명(Digital Signature)

1. 서 론

현재 무선 인터넷(mobile internet) 관련 기술이 빠르게 발전하고 있으며, 무선 이동통신 환경에서 금융, 증권, 경매 등과 같은 전자상거래 관련 서비스가 일반화되었다[1]. IMT-2000 등을 통해 무선 인터넷은 더욱 발전할 것이며 2004년엔 무선 인터넷 이용자 수가 유선 인터넷 이용자의수보다 많을 것으로 전망된다. ARC 그룹에 의하면 1999년 4억 2,800만명이던 이동 통신 가입자가 2004년엔 12억 3,500만명으로증가되며, 그 보급률이 70~80%에 이를 것으로 전망된다[2]. 국내에서도 1999년부터 무선 데이터 서비스 관련 인프라 및 소프트웨어 역량을 확보한 이동전화 사업자들이 본격적인 마케팅 활동을 전개하여 무선 환경에서 전자상거래 시장을 확대시키고 있다.

무선 인터넷 기반의 전자상거래가 활성화되기 위해서는 과금 솔루션 인프라 구축뿐만 아니라, 전자서명을 통한 거래 사실에 대한 부인방지 및 인증서 기반의 사용자 인증(au-

※ 이 연구는 2003학년도 단국대학교 대학연구비의 지원으로 연구되었음.
† 정 회 원 : 단국대학교 대학원 전산통계학과
†† 정 회 원 : 단국대학교 정보컴퓨터학부 교수
논문접수 : 2002년 10월 14일, 심사완료 : 2003년 2월 24일

thentication), 데이터 무결성(integrity) 등에 대한 인프라확산도 중요하다[3, 8]. 현재 WAP(wireless application protocol) 포럼에서는 무선 환경에 적합한 프로토콜로 TLS (transport layer security) 기반의 WTLS(Wireless TLS)를 제안한다. 하지만 WTLS는 사용자 인증, 데이터 무결성 등의 서비스는 제공하지만 부인방지 기능은 제공하지 않는다[4]. 부인방지 기능을 제공하는 대표적인 서비스는 공개키 암호 방식을 이용한 전자서명이다[3]. 서명자 자신의 고유한 개인키를 이용하여 생성되는 전자서명은 해쉬 함수(hash function)와 함께 사용자 인증 및 부인방지, 데이터의 무결성 등의 기능을 제공한다. M-Commerce에서 다른 문제점은, 소비자가 물품 구매에 대한 영수증을 오프라인으로 받게 되거나 전자서명이 없는 영수증을 전자우편으로 받게 된다는 것이다. 오프라인으로 영수증을 받는것은 실시간이 아니고 M-Commerce 특성에도 부합되지 않는다. 전자서명이 없는 영수증은 법적인 효력이 없는 단순한 정보에 불과하여 소비자에게 신뢰성을 주지 못한다. 따라서, M-Commerce에서 판매자와 구매자간 신뢰성 확보를 위해 전자서명 기반의 전자영수증 발급 및 검증이 필수적이다. 이를 위해 현재 유선 환경에서 전자영수증을 발급하는 시스템[13]은 운용되고 있지만 무선 환경에서 전자영수증을 발급하고 검증하는 시스템은 확인되지 않고 있다.

본 논문에서는 무선 인터넷 환경에서 안전한 전자거래를 위해 실시간으로 전자영수증을 발급하고 검증해 주는 시스템을 제안한다. 즉, M-Commerce에서 인터넷 쇼핑몰과 소비자간에 신뢰성을 보장하기 위해 WPKI(Wireless Public Key Infrastructure) 기반의 전자서명(인증서) 및 전자영수증을 도입하여 정보의 무결성을 보장하고 인증 및 부인방지 등의 보안 문제도 해결하고자 한다. 제안한 전자영수증은 상품 대금 지불과 동시에 쇼핑몰에 의해 발급되며, 거래 내역, 판매자 정보, 구매자 정보 등을 포함하는 전자문서이다. 발급된 전자영수증은 해당 전자서명과 함께 소비자에게 실시간으로 전달된다. 뿐만 아니라 신뢰할 수 있는 별도의 검증서버를 이용하여 전자영수증 및 전자서명을 검증하고 관리하게 하여, 이동통신 단말기의 속도 문제와 저장공간 부족 등의 단점을 보완하였다. 제안된 시스템은 전자거래기본법을 충실하게 반영하였으므로 무선 환경에서 전자상거래의 활성화에 기여할 것으로 기대된다.

본 논문의 구성은 다음과 같다. 2장에서는 PKI 기반의 전자서명, 해쉬 함수, WAP과 WML(Wireless Markup Language)에 대하여 간단히 기술한다. 3장에서는 무선환경에서의 전자영수증 발급 시스템의 구조와 전자영수증의 검증 방식에 대해 설명한다. 4장에서는 전체적인 구현과정과 결과에 대하여 기술하며 구현 결과를 화면으로 보여준다. 마지막으로 5장에서는 결론 및 향후 연구과제에 대해 기술한다.

2. 관련 연구

공개키 기반 구조(PKI, Public Key Infrastructure)란 공개키 암호 시스템(Public Key Crypto-system)을 이용한 보안 서비스를 효율적이고 광범위하게 이용할 수 있도록 해주는 제반환경으로 기술, 서비스, 사회, 문화적 파급 효과, 법, 제도, 교육 등을 포함하는 인증 구조를 뜻한다[4, 14]. 공개키 암호 시스템은 암호화키와 복호화키가 서로 다른 공개키 암호 알고리즘을 사용하며, 인터넷과 같은 공개 네트워크상에서 키의 관리와 분배가 효율적이다. 그러나 키가 크고 암호화와 복호화에 많은 시간이 소요된다는 단점 때문에 일반적으로 데이터 암호화에는 사용하지 않으며 전자서명 등에 많이 사용되고 있다.

대표적인 공개키 암호 시스템으로 RSA(Rivest Shamir-Adleman) 기법이 있으며 이 밖에 ElGamal, Schnorr, LUC, ECC 등이 있다. PKI 기반의 전자서명에서 서명자는 자신의 개인키(비밀키)를 사용하여 전자문서에 서명하고, 서명검증자는 서명자의 공개키를 사용하여 서명을 검증한다. 전자서명은 전자문서의 내용이 수정 및 변조되지 않았다는 무결성을 보장하는 동시에 문서 생성의 주체를 제 3자가 확인할 수 있게 해 주며, 상호 인증 및 거래에 대한 부인방지 기능 등을 지원한다[3, 5]. 무선 환경의 인증에는 WPKI가 사용된다. 간단히 말하면, WPKI란 이동 단말기를 통해 안전한 거래를 할 수 있도록 해주는 보안 기술을 뜻하며 크게 인증기관, 등록기관, 단말기, CP/SP 서버용 프로그램으로 구성된다[3].

2.1 해쉬 함수와 전자서명

DSS(Digital Signature Standard)의 경우 160비트 메시지로부터 320비트의 서명을 생성하는데, 원문 메시지가 큰 경우 원문을 160비트 블록 단위로 나누어 서명하면 처리 시간이 길어지며 서명 후 메시지 길이는 원문의 두 배가되어 전송시간 및 저장공간을 낭비하게 된다. 그리고 전송시 서명된 메시지들 일부가 손실될 경우 처리 과정이 쉽지 않다. 따라서 전체 메시지를 간단히 서명하기 위해 해쉬 함수가 필요하며, 큰 메시지에 대해 특정 길이의 메시지 다이제스트(해쉬 함수를 통해 생성된 축약문서)를 생성한 다음 전자서명을 하면 속도도 향상된다. 현재 여러 해쉬 함수들이 제안되었는데 대표적인 것으로 MD5, SHA-1 등이 있다. MD5는 해쉬값이 128비트라는 단점 때문에 strong collision-free의 성질을 만족하지 못하는 것으로 알려져 있으며, SHA-1는 160비트 해쉬값을 가지며 지금까지 MD5와 같은 문제점

은 발견되지 않아 주로 사용되고 있다.

이처럼 성능을 위하여 해쉬 함수를 사용하여 원문을 축약시킨 다음 그 축약문에 서명한다[9]. 즉, 임의 길이의 원본 전자문서(M)를 해쉬 함수 H로 먼저 축약시킨 축약문서($H(M)$)를 서명자(송신자)가 자신의 개인키로 암호화하여 전자서명 $D(H(M))_{sk}$을 생성한다. 그 다음, 서명자는 원문 M과 전자서명 $D(H(M))_{sk}$을 함께 상대방(수신자)에 전송한다. 수신자는 전송된 M을 해쉬 함수 H로 축약하여 $H(M)'$을 생성하고, 또 전송된 $D(H(M))_{sk}$을 송신자의 공개키로 복호화 하여 $H(M)$을 생성하여 $H(M)'$와 $H(M)$를 비교하여 무결성을 검증하게 된다[4]. 전자서명 방식으로는 RSA와 DSA (Digital Signature Algorithm) 등이 있다.

2.2 WAP과 WML

WAP은 WML을 기본 구성요소로 사용하고 기존의 인터넷 망과 무선 통신망을 통합하기 위한 구조를 지칭한다[1, 6, 11]. WAP은 Phone.com(현재 Openwave.com), Ericsson, Nokia, Motorola 등 4개회사가 참여한 WAP 포럼에서 제안한 구조로 무선 데이터 서비스 사용자들이 쉽고 간편하게 인터넷에 접속할 수 있도록 고안된 표준 규격이다[7]. 현재 전세계 200여 업체들이 WAP 표준을 지원하는데, 포럼 참여 업체들이 세계 이동전화 가입자의 90%를 차지하며 표준화 활동 및 관련 응용 개발을 수행하고 있다. WAP은 이동단말기와 웹 서버 사이에 WAP 게이트웨이를 두어 WAP 프로토콜 TCP/IP 프로토콜을 무선환경의 특수성에 맞게 변환해 준다. WAP 게이트웨이는 클라이언트와는 WSP/WTP(Wireless Session Protocol/Wireless Transaction protocol)를 사용하여 서로간의 요구와 응답을 처리하며, 웹 서버와는 HTTP를 이용하며 통신할 수 있다.

WML은 XML을 기반으로 하는 마크업언어로 핸드폰, PDA 등과 같은 무선환경에서 사용자 인터페이스를 구현하고 컨텐츠를 제작하는데 사용된다. 무선 인터넷 통신을 위한 표준을 개발하기 위해 구성된 WzAP 포럼에서는 XML 문서 양식을 내용으로 하는 WML 표준을 제공하고 있다[1]. WML 문서는 XML을 기반으로 하고 있으므로 문서 정의를 위해 XML의 DTD(Document Type Definition)를 사용한다. 따라서 WML 문서는 서두에 (그림 1)과 같은 문서 선언부가 필요하며[10], 이 선언부에는 무선 단말기에서 WML 사용시 한글을 사용할 수 있도록 선언되어 있다.

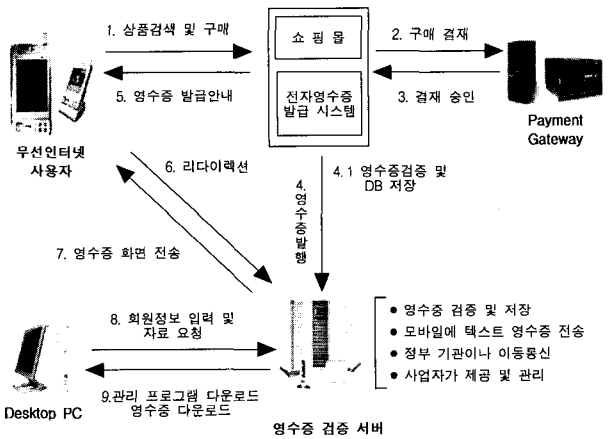
```
<?xml version ="1.0" encoding = "ks_c_5601-1987"?>
<!DOCTYPE WML PUBLIC "-//WAPFORUM//DTD WML 1.1//KO"
"http://www.wapforum.org/DTD/wml_1.1.xml">
```

(그림 1) WML 문서 선언부

WPKI의 경우 단말기 성능 제한으로 인해 암호화와 복호화에 많은 시간이 소요되는 단점이 있으며 WML 스크립트에 의해 생성된 전자서명은 WAP 게이트웨이를 통과하면서 인터넷에 적합한 형식으로 변환되어야 하기 때문에 보안상에 허점이 생긴다. 그러므로 이동통신에서의 보안은 무선 네트워크 환경에 대한 고려뿐만 아니라 이동통신 단말기 등의 성능도 고려해야 한다. 따라서 단순히 무선 인터넷을 통한 보안은 현재, 그 실효성이 미비하며 보다 효과적인 보안을 위하여 유선 인터넷과 연동되어야 한다[8].

3. 전자영수증 발급 및 검증 시스템

본 논문에서 제시하는 WPKI 기반의 전자영수증 발급 시스템이 (그림 2)에 나타나 있다. 무선 인터넷 사용자(고객)가 무선 단말기로 쇼핑몰에 접속하여 구매한뒤 결재를 신청하면 쇼핑몰은 Payment Gateway에 결재 정보를 보낸다. Payment Gateway에서 거래가 승인되면 쇼핑몰에 승인번호를 부여한다. 쇼핑몰은 승인번호를 받은 후 결재 정보를 전자영수증 발급 시스템에 전달한다. 전자영수증 발급 시스템은 전자영수증을 생성하고, 쇼핑몰의 개인키를 이용 전자영수증에 대한 전자서명도 생성하여 데이터베이스(DB)에 저장한다. 쇼핑몰의 개인키와 공개키는 인증기관에서 미리 생성하여 개인키를 쇼핑몰에 제공하고 공개키는 인터넷에 제공되어 있는 것으로 가정한다. 이제 전자영수증 발급 시스템은 파일로된 전자서명과 전자영수증을 검증 서버로 전송한다. 전자영수증의 서식은 본 논문의 관심 사항이 아니며, 여기서는 테스트를 위해 4장의 <표 2>와 같은 서식을 사용한다.



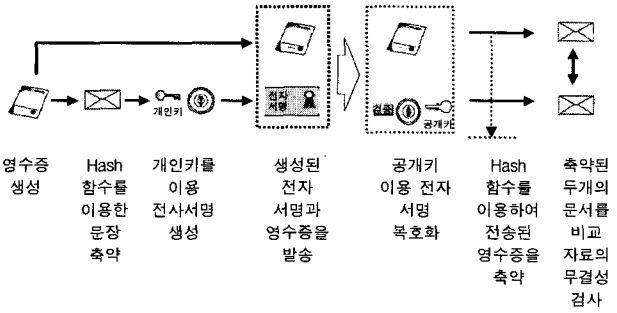
(그림 2) 전자영수증 발급 시스템

검증 서버는 발급 시스템으로부터 전자서명과 전자영수증을 수신한 다음 전자서명은 쇼핑몰의 공개키를 이용하여 복

호화하고 전자영수증은 해쉬 함수를 이용하여 축약한다. 그 다음 두 문서를 비교하여 영수증의 무결성을 검사한다. 만약 무결성이 입증되면 데이터베이스에 저장함과 동시에 리다이렉션된 무선인터넷 사용자에게 영수증을 전송한다. 만약 자료에 이상이 있으면 쇼핑몰에 재전송을 요구한다. 사용자는 무선 단말기로 전송된 영수증을 일차적으로 확인한 후, 나중에 유선 인터넷으로 검증 서버에 접속하여 자신의 PC로 영수증 및 전자서명을 다운받아 관리할 수 있다. 또한 영수증 관리 프로그램도 다운받아 PC에 설치하여 전자영수증을 쉽게 관리할 수도 있게 구성되었다. 전자영수증은 거래에 대한 증거이므로 반드시 신뢰할 수 있는 검증 서버나 자신의 PC에 안전하게 관리될 수 있다. 따라서, 기존의 인터넷 환경과 같은 전자상거래 보안 수준을 유지할 수 있다.

3.1 전자서명 생성 및 검증

전자서명을 생성하고 검증하는 과정이 (그림 3)에 나타나 있다. 구매가 완결되면 전자영수증 발급 시스템은 구매 정보가 담긴 전자영수증을 해쉬 함수를 이용하여 축약한다. 축약문서를 쇼핑몰의 개인키로 암호화하여 전자서명을 생성하여 전자영수증과 전자서명을 함께 검증 서버에게 전송한다. 검증 서버는 전자영수증과 전자서명을 수신하고 쇼핑몰의 공개키를 사용해서 수신한 전자서명을 복호화한다. 전자서명을 복호화하게 되면 축약된 전자영수증을 얻을 수 있다. 또한 수신한 전자영수증을 해쉬 함수를 이용하여 축약한다. 두 축약 문서를 비교하여 전자서명과 무결성을 검증한다. 본 논문에서는 해쉬 알고리즘으로는 SHA-1을, 전자서명 생성 방식으로는 DSA를 이용한다.

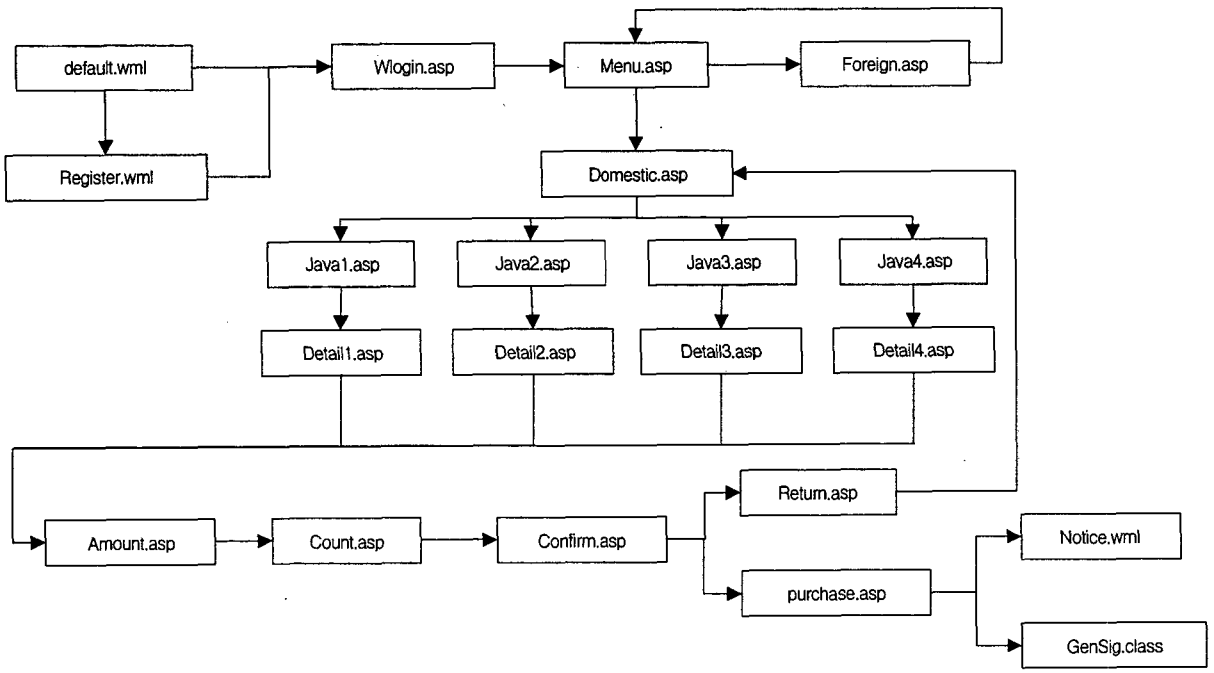


(그림 3) 전자서명의 생성 및 검증 과정

4. 구현 및 결과

4.1 구현 환경

본 논문에서 전자영수증 발급 시스템은 펜티엄 800MHz, 윈도 2000 서버에, 검증 서버는 펜티엄 700MHz, 윈도 2000 서버에 구현되었다. 웹 서버로는 IIS 5.0, 무선 쇼핑몰과 연계되는 부분의 구현언어로는 WML과 ASP를, 전자영수증 발급, 전자서명 생성 및 검증 부분의 구현 언어로는 WML, Java와 Java Servlet을 사용하며, Java Servlet을 위하여 Servlet 엔진인 Resin 2.0을 설치하였다[12]. 전자영수증 및 전자서명 보관을 위한 데이터베이스로는 MS SQL 7.0을 이



(그림 4) 무선 쇼핑몰 코드 구성도

용한다. 무선 환경에서 인터넷 쇼핑몰을 접속하는 무선 사용자 인터페이스로는 Pentium 800Hz, Windows 98 SE에서 구동되는 Phone.com사(현재 Openwave.com)의 에뮬레이터 UP.Phone 4.0을 이용하여 구현하였다.

4.2 전체 구성

테스트를 위해 WML과 ASP를 이용 서점 쇼핑몰을 구현하여 일련의 구매 과정을 수행하였다. 무선 서점 쇼핑몰 프로그램의 전체 구조는 (그림 4)와 같다. 전체 시스템을 크게 키 생성부분, 전자영수증 발급 시스템, 검증 서버 등 세 부분으로 나누어 기술한다. 키 생성부분에서는 쇼핑몰의 비밀키와 공개키를 생성하여 보관한다. 전자영수증 발급 시스템은 WML과 ASP, Java Servlet 등을 이용 전자영수증 및 전자서명을 생성하여 검증 서버로 전송한다. 검증 서버는 소켓을 통해 수신된 전자서명과 영수증을 Java Servlet을 이용하여 검증한다. 검증 서버는 전자영수증 및 전사서명의 무결성을 검사하고 데이터베이스에 저장한다.

4.2.1 키 생성

전자서명을 생성 및 검증을 위해선 개인키와 공개키가 필요하다. 본 논문에서는 키 생성 프로그램인 GenKey.java를 이용하여 한 쌍의 공개키(suepk)와 개인키(suesk)를 생성하여 각각 해당 파일에 저장한다. 생성된 키 값은 서점 쇼핑몰에 고유하며 개인키는 쇼핑몰에 비밀 보관하며 공개키는 인터넷을 통해 배포된다. 전자 상거래의 공신력을 위하여 공개키와 개인키는 신뢰성 있는 공인 인증기관에서 생성·발급되는 것으로 가정한다. (그림 5)는 키를 생성하는 GenKey.java의 의사 코드(pseudo code)이다.

```
Public static void main(String[] args)
{
        Create a key generator using DSA
        Initialize key generator
        Create a secret key(private key) with key
          generator
        Create a public key with key generator
        Save the secret and public keys in each file
}
```

(그림 5) GenKey.java의 의사 코드

4.2.2 전자영수증 발급 시스템

전자영수증 발급 시스템은 쇼핑몰로부터 받은 거래정보를 이용하여 전자영수증을 생성하는 ASP 프로그램과 DSA 공개키 암호화 방식과 SHA-1 해쉬 함수를 이용하여 전자서명을 생성하고 전송하는 Java Servlet 프로그램으로 구성된다. 전자영수증 발급 시스템중 전자서명을 생성하여 검증 서버로 송신하는 GenSig.java의 의사 코드는 (그림 6)과 같다.

```
public class GenSig extends HttpServlet
{
        Get the secret key
        Create a Digital Signature Object and initialize it with
        the private key
        Get an electronic receipt
        Generate a Digital Signature with the electronic receipt
        using DSA
        Save the Digital Signature in a file
        Send the text receipt to Mobile User
        Send the Digital Signature and the electronic receipt to
        Verifying Server
}
```

(그림 6) GenSig.java의 의사 코드

4.2.3 검증 서버

검증 서버는 전자서명과 전자영수증을 이용하여 무결성 검사를 하는 Servlet 프로그램과 사용자가 인터넷으로 접속하여 본인의 영수증 내역을 확인할 수 있는 ASP 프로그램으로 구성된다. 영수증 발급됨과 동시에 검증 서버는 전자영수증과 전자서명을 실시간으로 검증하여 무결성이 입증되면 데이터베이스에 저장한다. 또한 소비자가 추후에 자신의 Desktop PC를 이용하여 유선으로 검증 서버에 접속하여 인증 절차 후 영수증을 검색하고 다운받을 수 있도록 한다. 검증 서버는 그 역할의 중요성과 높은 신뢰성의 요구로, 이동통신사업 제공업자나 공인 인증기관 등의 신뢰할 수 있는 기관으로부터 제공되는 것으로 가정한다. (그림 7)은 검증 서버의 기능 중 전송된 전자영수증의 축약문서와 전자서명의 복호화 후 얻은 문서를 서로 비교하여 자료의 무결성을 검증하는 프로그램이다.

```
Public class VerSig extends HttpServlet {
        Receive a Digital Signature and an electronic receipt
        Get a public key from Shopping mall or Certificate
        Authority
        Create a Digital Signature Object and initialize it with
        the public key
        Decrypt the Digital Signature with the public key
        Digest the electronic receipt using SHA-1
        Compare the decrypted Digital Signature with the digested
        electronic receipt
        If the result is same then
                Save the electronic receipt in database
        Else
                Request again the electronic receipt and the Digital
                Signature
}
```

(그림 7) VerSig.java의 의사 코드

4.2.4 데이터베이스 테이블

<표 1>은 전자영수증 발급 시스템 구현시 사용된 DB 테이블들의 구조이고 <표 2>는 검증 서버 구현시 사용된 DB 테이블들의 구조이다. 전자영수증 발급 시스템은 자신의 테이블 정보들을 이용하여 <표 2>에 주어진 영수증 정보를 생성한다.

〈표 1〉 전자영수증 발급 시스템용 테이블 구조

Customer(사용자 정보)		
컬 럼 명	설 명	자 료 형
customer_id	사용자 고유번호	int(4)
customer_iden	사용자 아이디 (전화번호 이용)	char(10)
pass_wd	사용자 비밀번호	char(10)
customer_name	사용자 이름	char(10)

Purchase(구매 정보)		
컬 럼 명	설 명	자 료 형
purchase_id	구매 고유번호	int(4)
customer_id	사용자 고유번호	int(4)
purchase_date	구매 날짜	datetime(8)
amount	구매 수량	int(4)

Purchase_product_list(물건 구매 정보)		
컬 럼 명	설 명	자 료 형
list_id	리스트 고유번호	int(4)
product_id	상품(도서) 고유번호	int(4)
purchase_id	구매 고유번호	int(4)

Product(물품 정보)		
컬 럼 명	설 명	자 료 형
product_id	상품(도서) 고유번호	int(4)
product_name	상품(도서) 이름	int(4)
product_cost	상품(도서) 가격	int(4)
product_author	상품(도서) 저자	char(4)
product_public	상품(도서) 제조사(출판사)	char(4)
product_date	상품(도서) 생산날짜(출판날짜)	datetime(8)

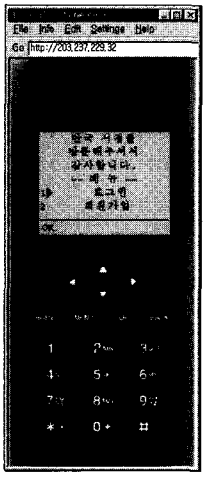
〈표 2〉 검증 서버용 테이블 설명

Customer(사용자 정보)		
컬 럼 명	설 명	자 료 형
customer_id	사용자 고유번호	int(4)
customer_iden	사용자 아이디 (전화번호 이용)	char(10)
pass_wd	사용자 비밀번호	char(10)
customer_name	사용자 이름	char(10)

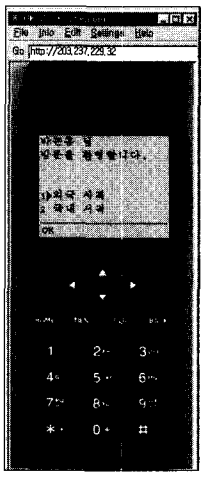
Receipt(영수증 정보)		
컬 럼 명	설 명	자 료 형
receipt_id	영수증 고유번호	int(4)
sm_code	쇼핑몰 코드	int(4)
customer_id	사용자 고유번호	int(4)
product_name	상품 이름	char(10)
product_cost	상품 가격	char(10)
purchase_date	구매 날짜	datetime(8)
amount	구매 수량	char(8)
purchase_id	구매 고유번호	int(4)

4.3 결과 및 성능

무선 인터넷 서점을 사용하는 사용자 인터페이스는 Phone.com에서 제공하는 휴대폰 에뮬레이터를 이용하여 구현하였다. 구매자는 무선 인터넷을 이용하여 도서를 구매하고 전자영수증과 전자서명을 발급받게 된다. (그림 8)은 처음 무선 인터넷 서점에 접속했을때 접하는 로그인 화면이며 (그림 9)는 로그인에 성공했을때 볼 수 있는 초기 화면이다.

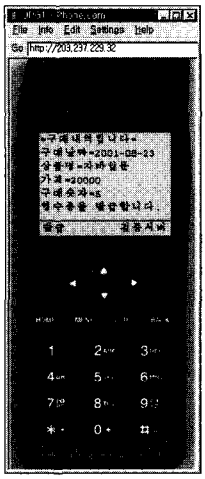


(그림 8) 로그인 화면

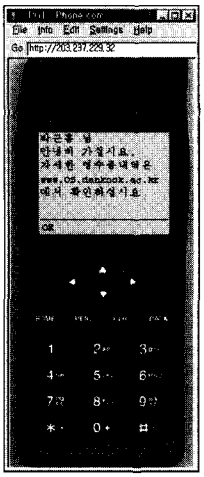


(그림 9) 구매 메뉴화면

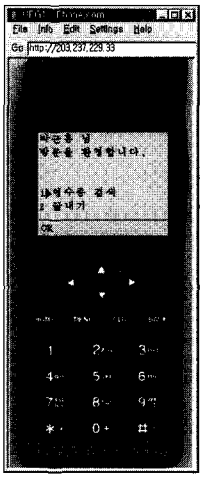
영수증 발급 화면인 (그림 10)을 보면 구매 날짜와 상품명, 구매 숫자 등의 구매 정보가 나타나 있다. '발급' 메뉴를 선택하면 전자서명이 생성되며 '검증 서버' 메뉴를 선택하면 검증 서버로 연결된다. (그림 12)와 (그림 13)은 쇼핑몰에서 검증 서버로 이동한 후 검증된 영수증을 핸드폰 상에서 확인하는 화면이다. 발급된 전자서명과 전자영수증은 검증 서버에서 검증 후 저장되며, 이후 사용자가 유선 인터넷으로 접속하여 본인의 PC에 다운받아 관리할 수 있다. PC로 검증 서버에 온라인 접속하여 전자영수증과 전자서명을 다운받는 화면이 (그림 14)에 나타나 있으며, 전자영수증을 관리를 도와주는 가계부 프로그램도 다운받을 수 있다.



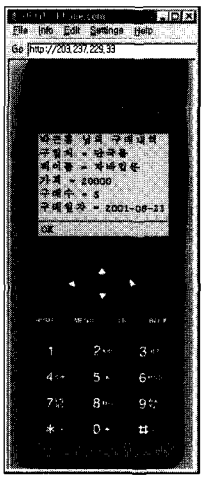
(그림 10) 영수증 화면



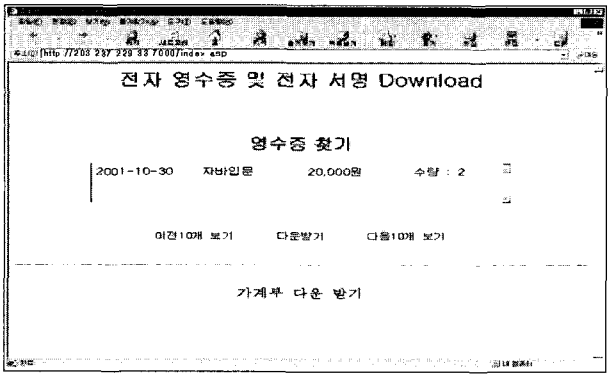
(그림 11) 쇼핑몰 구매 종료 화면



(그림 12) 검증 서버 접속 화면



(그림 13) 영수증 확인화면



(그림 14) 검증 서버의 영수증 및 전자서명 다운로드 화면

구현한 시스템의 성능을 분석하기 위해 키 생성, SHA-1 해쉬 함수 수행, DSA 전자서명 생성 및 검증에 소요되는 시간들을 10번 이상 측정하여 평균 소요시간을 계산해 보았다. DSA를 위한 512비트 크기의 키 쌍을 생성하는데 드는 평균 소요 시간은 24.0ms였다. SHA-1 수행 시간, DSA 전자서명 생성 및 검증 시간이 <표 3>에 나타나 있다. 전자 영수증의 크기가 1KB 이하라고 하였을 때, SHA-1은 전자영수증을 입력받아 항상 160비트의 해쉬값을 출력하며 표에 나타난 바와 같이 수행 시간은 전자영수증 크기에 따라 다르다. 전자영수증에 대한 전자서명 생성은 160비트 해쉬값을 입력받아 수행되며 입력 데이터 값에 따라 소요 시간은 일부 차이가 난다. 전자영수증의 전자서명 검증은 전자서명 복호화, SHA-1 수행, 두 값의 비교 등의 연산을 포함하므로, <표 3>의 결과처럼 전자서명 생성 시간보다 길며 또 복호화 시간 및 전자영수증의 크기에 따라 검증 시간이 차이가 난다. 전자서명 검증시 복호화 시간도 입력 데이터 값에 따라 일부 차이가 난다. 즉, 전자서명과 관련하여 암호화 및 복호화 시간이 다르다.

〈표 3〉 SHA-1 및 DSA의 성능

소요시간 \ 전자영수증 크기		512바이트	1024바이트
SHA-1 수행 시간		3.2ms	6.0ms
DSA 전자서명	생성 시간	20.1ms	22.5ms
	검증 시간	24.9ms	29.1ms

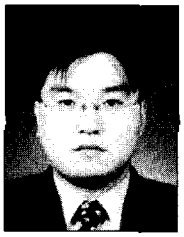
5. 결 론

본 논문에서는 무선 인터넷 기반의 전자상거래시 소비자의 권익을 보호하고 상거래를 활성화하기 위하여 전자영수증 발급 및 검증 시스템을 제안하였다. 전자서명을 이용한 전자영수증은 전자상거래의 신뢰성를 높이고 소비자에게 쇼핑몰의 신용을 보장한다. 즉, 공인기관의 인증서를 통한 거래는 상거래의 투명성과 안전성을 제공한다. 전자영수증을 공인 검증 서버의 데이터베이스에 저장할 수 있어 무선 단말기가 가지는 저장 공간 부족의 문제점을 보완하였고 성능이 좋은 검증 서버를 이용하여 전자서명과 전자영수증을 검증하는 속도도 향상시킬 수 있다. 뿐만 아니라 사용자가 편리한 시간에 PC로 검증 서버에 접속하여 자신의 전자영수증을 검색·다운받아 PC에서 거래 내역을 직접 관리할 수 있다.

전자영수증은 소비자를 보호하고 상거래의 신뢰성을 높일 수 있으며 검증 서버는 무선 인터넷의 보안상의 약점과 무선 단말기의 성능을 보완하는데 기여할 수 있다. 현재 빠르게 발전하는 이동통신 단말기의 성능과 PDA 시장의 확대, 그리고 IMT-2000 등의 무선 인터넷 시장의 활성화로 새로운 형태의 이동통신 서비스가 계속 등장할 것이다. 향후 새로운 요구와 기술력의 변화에 맞춰 전자영수증, 전자세금계산서 등의 일반적인 전자문서를 효율적으로 발급하고 관리하는 방안에 대해 연구할 계획이다.

참 고 문 헌

[1] LG-EDS 시스템 아이엔틱팀, "무선 인터넷 어플리케이션 프로그래밍", 삼양출판사, 2000.
[2] 무선인터넷백서편찬위원회, "무선인터넷 백서 2001", 소프트뱅크 미디어, 2000.
[3] 이명성 등, "특집/모바일 서비스", 정보처리학회지, 제9권 제2호, 2002.
[4] 이만영 등 5인, "전자상거래 보안기술", 생능출판사, 1999.
[5] (주)니츠 보안기술연구팀, "인터넷 보안기술", 도서출판 동서, 2000.
[6] 김기조 등 4인, "무선 응용 프로토콜 기술", 정보처리학회지, 제7권 제3호, pp.44-56, 2000.
[7] 남기범 등 2인, "무선인터넷 홈페이지 만들기", 삼양출판사, 2000.
[8] P. Niskanen, "Inside WAP Programming Applications with WML and WML Script," Addison-Wesley, 2000.
[9] J. Garms and D. Somerfield, "Professional Java Security," Wrox Press Ltd, May, 2001.
[10] S. M. Foo et. al., "Beginning WAP, WML, & WML Script," Wrox Press Ltd, 2001.
[11] C. Arehart et. al., "Professional WAP," Wrox Press Ltd, 2001.
[12] D. Hamner et. al., "Java Network Programming," Manning Press, 1999.
[13] http://www.pimstech.co.kr.
[14] http://www.pentasecurity.com/korean/faq.html.



박 근 홍

e-mail : rooty71@dankook.ac.kr
1999년 단국대학교 고분자공학과(학사)
2002년 단국대학교 대학원 전산통계학과 (이학석사)
2002년~현재 산은캐피탈 전산팀
관심분야 : 전자상거래 보안, 무선 인터넷, 운영체제 등

조 성 제

e-mail : sjcho@dankook.ac.kr
1989년 서울대학교 컴퓨터공학과(학사)
1991년 서울대학교 대학원 컴퓨터공학과 (공학석사)
1996년 서울대학교 대학원 컴퓨터공학과 (공학박사)
1996년~1997년 서울대학교 컴퓨터신기술연구소 연구원
2001년 미국 University of California, Irvine 객원 연구원
1997년~현재 단국대학교 정보컴퓨터학부 컴퓨터과학전공 조교수
관심분야 : 컴퓨터 보안, 시스템 소프트웨어, 실시간 시스템, 분산 시스템 등