VM의 자동 변수 생성 방식 기반 모바일 지급결제 시스템

(A Mobile Payment System Based-on an Automatic Random-Number Generation in the Virtual Machine)

강 경 석 † 민 상 원 †† 심 상 범 †††
(Kyoung-suk Kang) (Sang-won Min) (Sang-beom Shim)

요 약 이동전화가 전자상거래와 온라인 뱅킹의 새로운 수단으로 등장하고 있다. 모바일 기기를 통한 모바일 지급결제는 인터넷 결제나 소액결제 등에서 대중적으로 이용되며 새로운 결제수단으로 각광받고 있다. 하지만 현재의 모바일 지급결제는 소액에 치중하고 있으며 안전하고 표준화된 기술의 미비 등의 문제가 해결해 되어야할 과제로 떠오르고 있다.

본 논문에서는 현재 모바일 지급결제 서비스의 정의와 유형을 알아보고 모바일 지급결제 서비스에서의 인증의 의미와 각 지급결제 서비스의 인증방식에 대해 살펴보았다. 또한 각 인증방식들에서의 사고유형과 그 원인 그리고 그에 따른 대책을 알아보고, 현시점에서 그 대책이 실제 적용되기까지의 긴 시간동안 기존 설비나 휴대폰의 하드웨어를 추가 또는 변형하지 않는 범위 안에서 실현 가능한 새로운 인증절차를 제안하였다. 본 논문이 제안한 인증기법은 휴대폰의 VM을 이용한 결제로서 기존 SMS로 전달된 난수를 입력하는 방식의 문제점으로 지적되는 전달 내용에 대한 타인의 도용 위험성을 줄이기 위해 사용자가 직접 본인의 휴대폰에서 결제용 VM을 구동 난수를 확인하여 난수를 사이트에 입력하는 방식을 사용한다. VM 다운로드 후 처음 사용시 다운받은 VM의 S/N을 서버에 등록하여 VM을 구동할 때마다 S/N과 휴대폰 번호를 매칭한 후 난수를 부여하여 기존 휴대폰 통합 과금 서비스에서는 하기 어려운 점이었던 등록된 폰 이외의 불법 복제된 폰의 결제를 막을 수 있게 하였다. 또 난수 발급시 사용되는 매개체를 SMS발송에서 47 byte 패킷통신으로 대체하여 난수를 발급할 때 소요되는 시간을 대폭 줄이고 결제할 때 소요되는 비용을 기존의 1/3로 절감 하였다.

키워드 : 모바일폰, 지급 시스템, Virtual machine

Abstract A mobile phone has became as a payment tool in e-commerce and on-line banking areas. This trend of a payment system using various types of mobile devices is rapidly growing, especially in the Internet transaction and small-money payment. Hence, there will be a need to define its standard for secure and safe payment technology.

In this thesis, we consider the service types of the current mobile payments and the authentication method, investigate the disadvantages, problems and their solutions for smart and secure payment. Also, we propose a novel authentication method which is easily adopted without modification and addition of the existed mobile hardware platform. Also, we present a simple implementation as a demonstration version. Based on virtual machine (VM) approach, the proposed model is to use a pseudo-random number which is confirmed by the VM in a user's mobile phone and then is sent to the authentication site. This is more secure and safe rather than use of a random number received by the previous SMS. For this payment operation, a user should register the serial number at the first step after downloading the VM software, by which can prevent the illegal payment use by a mobile copy-phone. Compared with the previous SMS approach, the proposed method can reduce the amount

· 본 논문은 2005년도 광운대학교 교내 학술연구비 지원에 의해 연구되었음
† 비 회 원 : (주)하렉스인포텍 연구소
pine1404@hotmail.com
†† 종신회원 : 광운대학교 전자통신공학과 교수
min@kw.ac.kr
††† 비 회 원 : 광운대학교 대학원 전자통신공학과
paulussam@paran.com
논문접수 : 2005년 12월 23일
심사완료 : 2006년 10월 11일

of packet size to 30% as well as the time. Therefore, the VM-based method is superior to the previous approaches in the viewpoint of security, packet size and transaction time.

Key words : Mobile phone, Payment system, Virtual machine

1. 서 론

인터넷의 급속한 확산과 전자상거래 (e-commerce), 통신 기술과 같은 IT 응용기술의 발전은 e-비즈니스라는 새로운 비즈니스 형태를 만들어 냈으며, 무선 통신 시스템은 기존의 비즈니스를 모바일 비즈니스 형태로 전환시키고 있다. 특히 언제, 어디서나 접근 가능한 모바일 비즈니스의 특징으로 인해 모바일 기기를 통한 모바일 지급결제 이용자가 급격히 늘고 있다. 하지만 아직까지 모바일 지급결제 서비스는 적은 금액에 대한 결제가 중심이 되고 있으며, 모바일 지급결제 서비스 관련 업체들 간의 부조화와 안전하고 표준화된 기술의 미비 등이 당면 과제로 떠오르고 있어 현실적인 어려움이 내재하고 있다. 이러한 문제점들의 해결을 위해 정부는 모바일 지급결제 표준 단체를 만드는 것을 비롯하여, 이동통신사 및 결제대행사와 함께 좀 더 안정적인 모바일 지급결제 서비스의 표준화에 노력하고 있지만 아직까지 통일된 안을 찾지 못하고 있다[1].

본 논문에서는 모바일 지급결제 서비스 중에서 현재 가장 많이 사용되고 있는 휴대폰 통합 과금 서비스를 바탕으로 하드웨어 구조의 변경 없이 VM(Virtual Machine)을 이용한 개선된 형태의 사용자 인증절차를 제안하였다. 제안한 인증기법은 최근 휴대폰 게임을 통하여 쉽게 접할 수 있는 휴대폰의 VM을 이용한 결제 방식을 이용하였다. 기존의 SMS(Short Message Service)로 전달된 난수(One-time Password)를 입력하는 방식은 전송된 내용을 타인이 쉽게 도용할 수 있다 점에서 위험에 노출되어 있었다. 하지만 VM을 이용한 결제 방식은 사용자 본인이 직접 휴대폰에서 결제용 VM을 구동하여 난수를 확인할 수 있도록 하고, 이렇게 확인한 난수를 사이트에 입력하는 방식을 사용함으로써 타인에 의한 도용의 위험을 줄였다. VM 다운로드 후 처음 사용할 때 다운받은 VM의 S/N(Serial Number)을 서버에 등록하여 VM을 구동할 때마다 S/N과 휴대폰 번호를 매칭하여 기존에 저장한 정보와 맞을 경우에만 난수를 부여하도록 하였다. 이에 따라 기존 휴대폰 통합 과금 서비스에서는 하기 어려웠던 불법 복제된 폰을 통한 결제의 문제를 방지할 수 있게 되었다. 또한 난수 발급에 소요되는 시간을 대폭 줄이고, 결제할 때 소요되는 부가 SMS 발송비를 47 byte 패킷통신으로 대체하여 소요비용을 1/3로 줄였다.

논문의 구성은 2장에서 모바일 지급결제의 정의와 서비스 유형에 대해 간략히 기술하였고, 3장에서 모바일 지급결제에서의 인증의 개념과 각 모델에서 인증이 어떻게 이루어지고 있는지를 살펴본 후 현재 인증방식에서의 문제점과 부작용에 대해 살펴본다. 그리고 4장에서는 현재의 대책이 현실화되기까지 현재 사용하고 있는 결제 인프라나 휴대폰에서 적용 가능한 좀 더 향상된 인증기법을 제안하며, 마지막 5장에서 결론을 맺는다.

2. 모바일 지급결제 현황

2.1 모바일 지급결제의 정의

모바일 지급결제 (Mobile Payment)는 온라인과 오프라인 상에서 이루어지는 서비스와 재화 구매시에 무선 단말기기(휴대폰, PDA 등)를 이용하여 대금을 지급하는 결제서비스로 정의할 수 있다[2]. 즉, 모바일 지급결제는 상품 구입에 따른 대금 지급이 이동 통신망을 통해 이루어지는 서비스를 의미한다. 온라인 뿐 아니라 오프라인 상에서의 일반 상거래에서도 이동통신망을 이용하여 이용자 신원 확인 및 거래 정보의 전달, 인증 등의 결제 과정이 이루어져 기존 전자상거래에서 사용되던 인터넷 지급결제보다 넓은 서비스 영역에 걸쳐 서비스가 가능하며, 이는 일반 상거래에서 화폐적인 거래를 대체할 수 있는 새로운 지급결제 수단이 될 수 있다. 또한 모바일 지급결제 서비스는 이용자가 금융기관의 계좌에 직접 접근하지 않고도 서비스가 가능하다는 점에서 모바일 뱅킹(Mobile Banking)과도 구별되어지는 특징을 지닌다. 이러한 모바일 지급결제와 모바일 뱅킹 등을 포괄하는 넓은 의미로 모바일 금융 서비스(Mobile Finance)라고 하며, 이는 무선 단말기기로 무선통신망에 접속하여 금융기관 등과 거래하는 모든 형태의 금융거래를 의미하고, 여기에는 휴대폰 통합 과금 서비스, 증권 등도 포함된다[3].

모바일 지급결제는 개념상 전자상거래 유형의 하나이지만, 유선 인터넷망 기반의 전자상거래에 비해 이동성, 접근성, 보안성 등에 있어서 장점을 갖는다. 또한 오프라인 결제에 사용되는 기존의 화폐에 비교해서도 휴대성, 보안성 등의 장점을 지니고 있으며, 단순 지급결제 뿐만 아니라 이용자의 특성에 맞는 개인화 서비스가 가능하게 되어 다른 전자상거래 유형들과는 차별화된 서비스를 제공할 수 있을 것으로 보인다. 표 1은 소액과 고액으로 구분지어 결제 금액에 따라 구분지어지는 모바일 지급결제 유형에 대해 설명하고 있다[4]. 표 2는 모바일 지급결제의 이용 방법에 따른 분류를 나타내고 있다[5].

표 1 모바일 지급결제의 결제 금액에 따른 분류

구분	모바일 콘텐츠	온라인 상품	오프라인 상품
소액 결제	• 휴대폰 벨소리, 기상정보, 경기결과와 같은 몇 백원 단위의 모바일 콘텐츠 • 대체로 휴대폰 청구서에 합산되어 청구됨	• 온라인 게임 • 보통 선불형 전자화폐 내지 선불형 계정이용	• 음료수, 지하철 요금, 주차요금 • 대체로 휴대폰 청구서에 합산되어 청구됨
고액 결제	• 아직 이용 대상이 없음	• CD, 책 등 온라인 쇼핑 • 주로 신용카드 방식	• 주유 요금 등 오프라인 구매 • 신용카드, 계좌이체 방식

표 2 모바일 지급결제의 서비스 이용방법에 따른 분류

구분	종류	내용
스마트카드 (IC) 내장 또는 탈착여부	카드기반	• 칩에 결제정보를 담아 인증
	비카드기반 (S/W식)	• 무선네트워크를 통한 실시간 인증 및 대금결제 • 휴대폰 메모리에 결제정보를 저장하는 Mobile Wallet방식
무선네트워크 이용 여부	온라인 방식	• 무선인터넷에 접속, 모바일 뱅킹 또는 무선 결제 대행사를 이용한 온라인 쇼핑몰 등에서의 대금 지급결제(통화료부담) • 폰빌, Remote Payment
	오프라인 방식	• 휴대폰과 POS단말기,ATM 등 간에 근거리 통신기술(RF, IrFM, 블루투스, 2D바코드 등)을 이용, 대금 지급결제 • 통화요금 부담하지 않음
이동통신업체의 참여 정도	직접결제 방식	• 이동통신업체가 직접 지급결제 서비스 제공 주체 • 지급결제과정 전반을 이동통신업체가 관리하고 부담 • 휴대폰 통합 과금 서비스, SKT의 네모서비스
	간접결제 방식	• 금융기관과 제휴하여 간접적으로 서비스 제공 • 선불, 직불, 신용카드, 계좌이체 등

2.2 주요 모바일 지급결제 서비스

본 절에서는 대표적인 모바일 지급결제 서비스인 모바일 뱅킹, 칩카드 기반 지급결제, 휴대폰 통합 과금 에 대해 설명한다. 모바일 지급결제는 각 서비스별로 이동통신 사업자, 금융기관, 결제 대행사 등의 역할, 수익기반, 금융서비스에 대한 책임범위 등에 차이가 있다.

먼저 모바일 뱅킹 서비스는 이동전화로 이동통신사의 무선 인터넷 포털에서 제공하는 각 은행의 온라인 뱅킹 서비스 메뉴에 접속한 후, 은행이 제공하는 금융정보조회, 계좌조회, 계좌이체 등의 서비스를 이용하는 것을 말한다. 거래과정은 그림 1에서 보는 바와 같이 모바일 뱅킹 이용 단말기, 이동통신사, 송금은행, 수취은행의 거래 과정을 거치게 된다. 모바일 뱅킹 서비스는 이동전화를 기존 은행 서비스의 전달 채널을 이용하는 형태로서, 이동통신사의 무선 인터넷망에 접속하여 이용하는 것을 제외하면 은행이 PC와 유선인터넷망을 통하여 제공하는 기존의 인터넷 뱅킹 서비스와 유사하다고 할 수 있다. 은행과 이동통신사만이 거래에 참여하며, 이동통신사는 액세스 제공에 따른 수수료(무선인터넷 이용료 및 은행으로부터의 수수료)를 확보하고, 이용자의 거래정보, 거래에 대한 책임은 계좌를 보유한 은행이 담당한다.

다음으로 살펴볼 칩기반의 지급결제(Chip-Based Payment)는 이동 전화에 IC칩을 내장하여 신용카드 또는 전자 화폐 대용으로 결제하는 방식이다. 그림 2에서 보

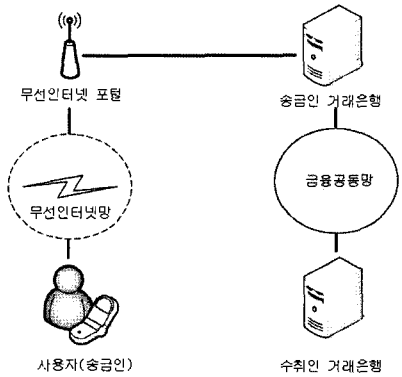


그림 1 모바일 뱅킹 서비스 흐름도

듯이 이동 통신 기기 자체가 하나의 지급결제 수단이 되는 것은 아니며, 기존의 신용카드나 전자 화폐에 관한 정보를 처리 및 전송하는 POS(Point Of Sale) 단말기와 동일한 기능(Virtual POS)을 수행한다[6].

마지막으로 휴대폰 통합 과금 서비스는 인터넷 쇼핑몰이나 게임 사이트에서 재화 및 서비스를 구매하면서 무통장 입금이나 신용카드를 통한 결제 대신에 휴대폰 번호를 입력하고, 대금은 나중에 휴대폰 이용 대금과 합산하여 결제하는 방식이다. 그림 3의 서비스 흐름도를 보면 사용자가 대금결제를 위해 휴대폰 번호를 입력하였을 때 이동통신업체는 휴대폰에 결제용 암호를 SMS

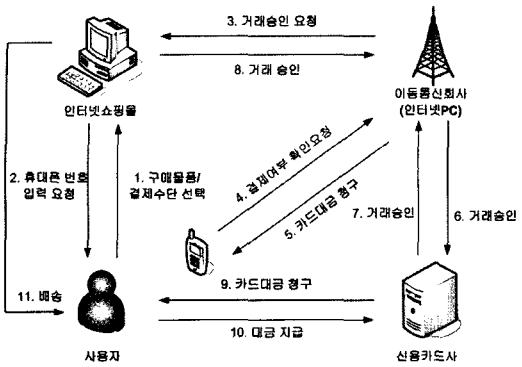


그림 2 칩카드 기반 온라인 지급결제서비스 흐름도

로 전송하거나, 소비자가 특정 ARS번호에 전화를 함으로써 대금결제가 승인되는 과정을 거치게 된다.

이동통신업체는 거래 대금 지불시 이동통신 가입자 여부 인증, 거래 대금의 통합청구 및 수납대행, 거래 대금 정산 및 결제대행사 이전까지의 송부업무 등을 수행하고, 거래금액의 일정 부분을 수수료로 취한다. 모바일 결제대행사는 거래 내역 관리, 인터넷 쇼핑몰이나 온라인 CP(Contents Provider)와의 정산 업무 등을 수행하고 추가로 CP로부터 수수료를 취한다. CP는 판매 및 서비스제공에 대한 최종적인 회수 책임을 지며, 요금납부 및 수납확인 절차가 수반하는 시간은 일반적으로 매출시점에서 대금회수까지 2.5개월 이상 소요되며, 이러한 이유로 인해 일종의 외상제공으로 볼 수 있다.

3. 모바일 지급결제의 사용자 인증

3.1 각 모델에서의 인증

모바일 지급결제에서의 인증이란 휴대폰을 통하여 결제 또는 지급을 요청한 사람이 사용이 허가된 휴대폰사용자 본인인지의 여부와 현재 폰을 가지고 있는지의 확인 절차를 거쳐 결제승인을 얻는 일련의 확인 절차를 일컫는다. 본 장에서는 2장에서 분류한 모바일 지급결제 서비스 중 휴대폰 통합 과금 서비스를 중심으로 각 방식에 따라 실제로 인증이 어떻게 이루어지는지 기술한다.

가장 먼저 살펴볼 IC칩을 이용한 인증 방식은 2장에서 소개한 모바일 뱅킹 서비스와 칩카드 기반 지급결제 서비스에서 사용되고 있다[7]. IC칩 기반 모바일 뱅킹은 은행이 발급한 금융 IC칩을 휴대폰에 장착하여 잔액 조회, 계좌 이체와 같은 각종 은행거래를 할 수 있는 서비스를 말하며, 휴대폰이 다양한 기능을 지원할 수 있는 지능형 단말기로 진화하면서 기본적인 은행거래 이외에 신용카드, 교통카드, 증권시세 조회 등 다양한 서비스가 함께 제공되고 있다.

IC칩 기반 모바일 뱅킹은 칩에 인터넷 뱅킹 ID와 각종 금융 정보를 저장하여 PIN 번호, 계좌 비밀번호와 보안카드의 3중 보안체계를 사용하고 있으며, PIN 번호를 5회 이상 잘못 입력하면 자동적으로 잠금 기능이 수행되기 때문에 기타 지급결제 서비스에 비해 보안성이 높다. 또한 IC칩을 이용한 거래에서 보고된 금융 사고가 없으며, 현재로서는 프로그램 구동까지 몇 단계를 거쳐가야 한다는 불편한 점 이외에는 인증이나 보안면에서는 거의 완벽하다고 알려져 있다[8]. 그림 4는 IC 칩기반의 모바일 뱅킹 서비스가 어떠한 과정을 통해 제공되는지를 설명하고 있다.

다음으로 SMS를 이용한 인증 방식을 살펴보면, SMS는 모바일 지급결제 서비스 중에서 가장 보편적으로 많이 쓰이는 방법으로서 현재 가장 큰 시장 규모를 가지고 있으며, 주로 온라인 서비스의 콘텐츠 사용료 결

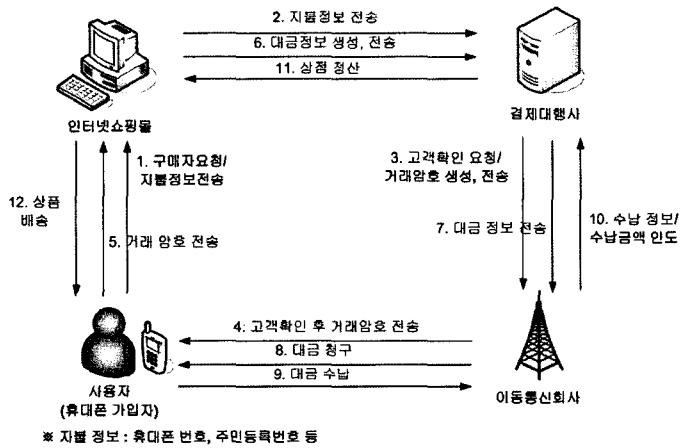


그림 3 휴대폰 통합 과금 서비스 흐름도

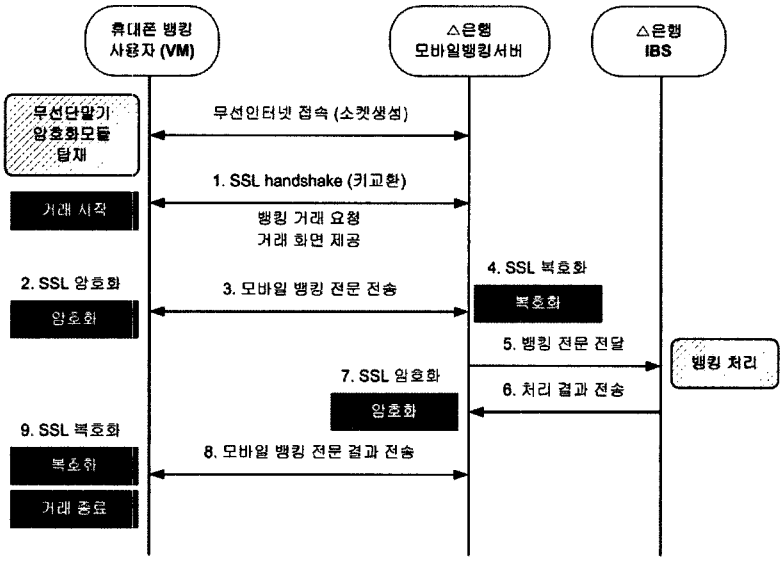


그림 4 IC 칩기반 모바일 뱅킹 서비스 제공 절차

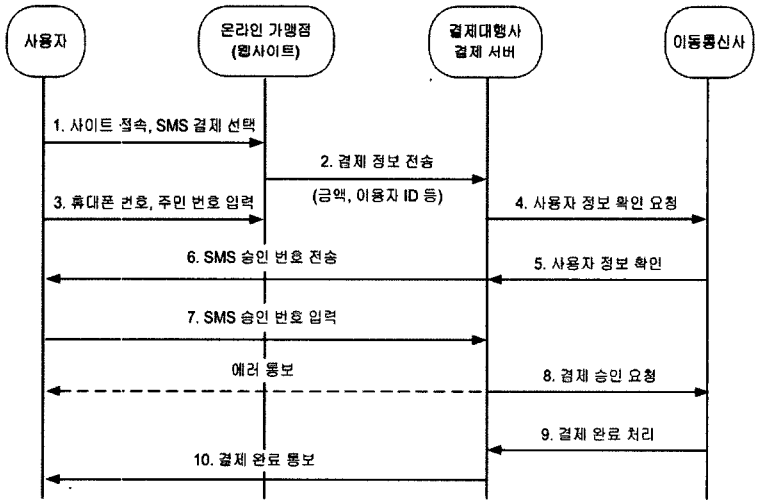


그림 5 SMS를 이용한 휴대폰 통합 과금 서비스 흐름도

제 등에서 많이 사용되고 있다. SMS를 이용한 인증의 경우 그림 5와 같이 결제를 원하는 사용자가 결제하고자 하는 사이트에 접속하여 휴대폰 번호, 휴대폰 가입자 주민번호, 이동통신사 등의 결제정보를 입력하면 휴대폰으로 도착하는 난수를 웹사이트 결제 창에 입력하여 본인 인증 및 휴대폰 소지 여부를 확인, 결제를 승인하는 절차로 이루어진다[9].

다음으로 Callback URL을 이용한 인증 방식은 기존 모바일 지급결제에서 많이 사용되고 있는 SMS를 이용한 과금 서비스가 온라인에서만 사용할 수 있다는 단점을 보완하여 나온 인증 방식이다. 이 방식은 SMS에 포함된 Callback URL을 통해 통화 버튼을 눌러 정해진 URL로 접속하면 결제를 승인하는 방식이다. 그림 6과 같이 결제를 원하는 사용자가 결제를 원하는 사이트나 오프라인 매장에서 통신사와 주민번호, 휴대폰 번호를 입력하는 등 납부 요청을 하게 되면, 사용자에게 Callback URL 정보가 포함된 SMS가 보내지고, 통화 버튼을 눌러 결제를 확인, 결제에 대한 승인 완료된다.

마지막으로 기존 칩카드 기반 지급결제 서비스에서의 전용 단말기 구입이나 전용칩 발급의 번거로움을 보완한 형태의 광리더기를 이용한 인증 방식이 있다. 광리더기를 이용한 인증 방식은 휴대폰 액정화면에 빛이 있다는 점에 착안하여 개발된 것으로써, 프로그램을 통하여 휴대폰 고유 정보를 액정화면의 깜박임을 통하여 암호

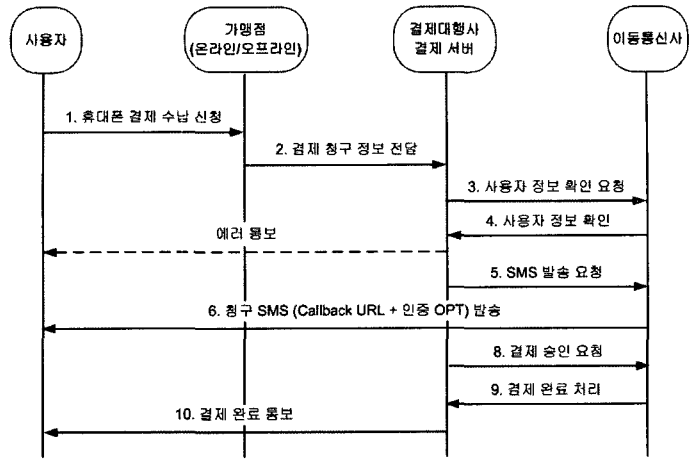


그림 6 Callback URL을 이용한 모바일 지급결제 서비스 흐름도

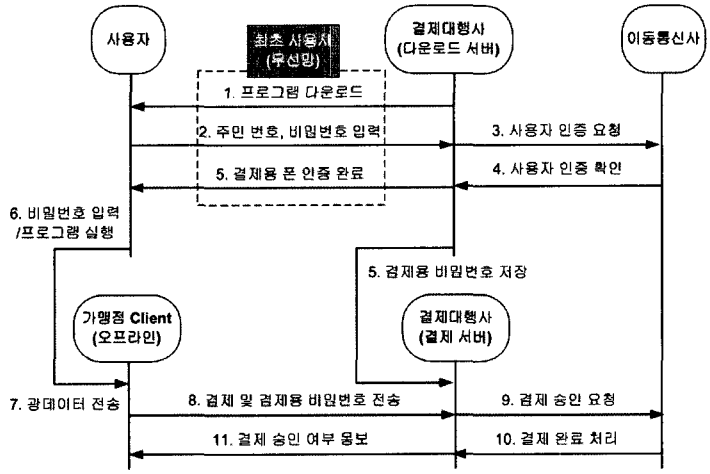


그림 7 LCD BL을 이용한 모바일 지급결제 서비스 흐름도

화하여 결제에 활용하는 기술을 사용하고 있다. 이 서비스는 그림 7에서 보듯이 휴대폰에 결제전용 VM을 다운받아 암호화된 개인결제 정보를 휴대폰 액정의 점멸(Blinking)을 통하여 전용리더기에 전달하는 방식으로 진행되며, 오프라인 전용 휴대폰 소액결제 서비스로 이용되고 있다. 이 방식을 통한 인증을 처음 사용할 때는 반드시 휴대폰에 결제전용 프로그램을 다운로드받아 주민번호와 인증번호 4자리를 입력하여 본인임을 인증 받는다. 그 후부터는 오프라인 매장에서 사용시에 전용 프로그램을 실행하여 본인이 설정한 인증번호 4자리를 입력 후 리더기에 액정화면을 접촉시키면 암호화된 깜박임이 조회기에 전달되어 본인 여부를 확인 후 결제를 승인한다.

3.2 기존 인증 방식의 문제 현황

현재 휴대폰 지급결제에서 벌어지는 사고는 인증절차가 간단한 휴대전화 통합 과금 서비스에서 일어나고 있다. 또 사고의 유형 중 많은 부분은 불법 복제된 휴대폰을 이용한 통합 과금 서비스 도용이 차지하고 있다. 이는 주로 불법 복제한 폰으로 사전에 취득한 주민번호를 통하여 온라인 콘텐츠 사이버 머니를 충전하여 이를 되파는 식으로 현금화 하는 범죄 등에서 악용되고 있다.

정통부의 발표를 보면, 정통부 산하 중앙전파관리소가 적발한 연간 불법으로 복제된 폰의 수는 지난 2000년 3대였으며 2001년에는 6대, 2002년에는 단 한대도 없었다. 하지만 지난 2003년에는 1,097대로 폭증했고 지난해에는 858대를 기록되고 있으며, 지금과 같은 추세로 나간다면 올해 적발대수로만 7,000대를 넘길 정도로 큰 문제가 되고 있다. 정보통신부는 이러한 불법복제를 통한 모바일 지급결제 서비스의 범죄를 막기 위하여 현재 2005년 3월부터 휴대폰 인증제를 실시하면서 신형 휴대

폰에는 복제 여부를 확인하는 암호키를 삽입하여, 이용자가 휴대폰 전원을 켜는 순간 휴대폰에서 암호키가 이동통신업체에 전송돼 인증센터에 보관된 암호키와 동일한지 확인절차를 거치는 '파워 온 (Power on) 인증제'를 시행하고 있다. 그러나 2004년 이전에 제작된 폰은 암호키가 들어있지 않아 문제의 여지가 남아있다.

또한 정통부는 전화를 걸때마다 암호키를 전송하여 휴대폰의 복제 여부를 확인하는 '휴대폰 발신인증제'를 내년 상반기에 도입할 예정이다. 발신 인증제가 도입된 후에는 휴대폰으로 전화를 걸었을 때 휴대폰에서 발신된 암호키가 이동통신사 인증센터에 보관되어 있는 암호키와 다르면 전화가 걸리지 않게 된다. 또 전화를 받을 때와 SMS 등 무선데이터를 주고받을 때 암호키를 확인하는 '착신 및 무선데이타 인증제'가 내년 하반기, 늦어도 2007년부터 실시한다고 한다[10].

그러나 정통부에서 추진 중인 '휴대폰 발신인증제'와 같은 경우 휴대폰 통화시마다 암호키를 주고받아야 하기 때문에 이동통신사의 엄청난 서버 증설이 불가피한 상황이라 비용의 문제로 많은 진통이 예상되어 실제 서비스까지는 많은 시간이 필요할 것으로 보인다.

4. 새로운 인증 방식 제안 및 구현

휴대폰 통합 과금 서비스는 기존의 모바일 지급결제에서 가장 많이 사용되고 있으나 휴대폰 불법복제에 따른 인증과 관련된 문제점이 노출되어 있다. 본 장에서는 이러한 문제점을 개선한 형태로, 기존 시스템이나 휴대폰 하드웨어의 변경이나 추가 없이도 구현 가능한, 좀 더 보안상 안정된 모바일 지급결제 인증 모델을 제안, 구현하고자 한다.

4.1 VM을 이용한 안전결제

VM은 여러 가지의 다양한 무선 단말기 상에서 소프트웨어와 콘텐츠가 구동할 수 있도록 해주는 기술을 의미한다. 단말기의 종류나 각 단말기가 가지고 있는 해당 운영체제에 구속을 받지 않고 소프트웨어의 수정을 통해 각 단말기에 올릴 수 있도록 해주는 미들웨어이다. 따라서 사용자가 무선 인터넷 서버에서 필요한 콘텐츠나 어플리케이션을 다운로드 받아 각각의 단말기에서 구동할 수 있게 해준다.

본 논문에서 제안하는 인증 방식은 VM을 이용한 결제로서, SMS로 전달된 난수를 입력하는 기존 방식이 정보를 타인이 쉽게 도용할 수 있다는 약점을 가지고 있기 때문에 이러한 점을 개선하고자 사용자가 휴대폰에서 결제용 VM을 구동 패킷통신을 통해 난수를 확인할 수 있도록 하였고, 이렇게 확인한 난수를 사이트에 입력함으로써 인증을 받을 수 있게 하였다. VM을 이용한 방식은 VM 다운로드 후 처음 사용할 때에 VM의 S/N을 서버에 등록하는 프로세스가 있어서 다음 사용부터는 VM만 구동하면 난수를 즉시 확인할 수 있다.

기존의 SMS를 이용한 휴대폰 통합 과금 서비스에 비해 개선된 점은 매회 결제승인 요청시 VM의 S/N과 휴대폰 번호를 매칭 후에 난수를 부여하며, 이를 통해 기존에 등록한 정상적인 폰에서의 요청인지 아닌지를 구분하여 불법 복제된 폰에서의 결제를 막는 과정이 추가되었다. 또한 난수를 발급받기위해 소요되는 대기 시간을 기존의 10초~2분에서 1~3초 이내로 줄여 결제시에 낭비되는 시간을 없앴으며, 결제할 때 부가적으로 발생하는 SMS 비용을 1건당 10~15원에서 1건당 5원으로 낮추어 경제성을 높였다.

4.2 시스템 구성 및 절차

휴대폰 상에서 패킷통신을 이용하여 난수를 받아오는 절차를 구현하기 위하여 XCE사의 SK-VM을 사용하였다. 또, 사용자정보의 인증 및 난수, S/N의 발급 및 관리를 비롯한 결제승인 역할을 하는 서버는 윈도우 2000 환경에서 Visual C++를 사용하여 구축하였으며, 데이타베이스는 오라클을 사용하였다. 웹은 ASP를 사용하였으며, 주로 신규 가입 회원의 CallBack SMS 발송 및 기존 사용자의 결제로그를 담당한다.

그림 8은 VM을 이용한 안전결제의 흐름도를 나타내며, 다음과 같은 순서로 진행된다.

1. 사용자는 CP의 웹페이지에서 결제를 요청한다.
2. CP는 결제대행사의 결제 페이지를 호출한다.
3. 사용자는 결제 페이지에 주민번호와 HP No를 입력한다. 그리고 결제 버튼을 클릭한다.
4. 결제대행사는 사용자의 휴대폰 번호를 사용자 테이블 DB에서 검색한다.
5. 사용자 테이블 DB에서 휴대폰 번호 검색이 실패한 경우
   - A. CallBack SMS를 사용자의 휴대폰 번호로 전송한다.
   - B. 사용자는 수신된 CallBack SMS를 이용하여 프로그램을 다운 받는다.
   - C. 다운받은 프로그램은 최초로 실행하는 프로그램으로 사용자인증 과정을 거친다.
   - D. 사용자인증은 서버와 소켓통신을 하여 인증 과정을 거친다.
   - E. 결제대행사는 사용자인증을 거친 사용자에게 S/N을 발급하고, 휴대폰 번호와 VM 프로그램의 일련번호를 사용자 테이블에 저장한다.
   - F. VM 프로그램은 수신된 S/N을 휴대폰의 메모리에 저장한다.
6. DB 테이블에서 휴대폰 번호 검색이 성공한 경우
   - A. 결제 요청 테이블에 사용자의 정보와 난수를 삽입한다.

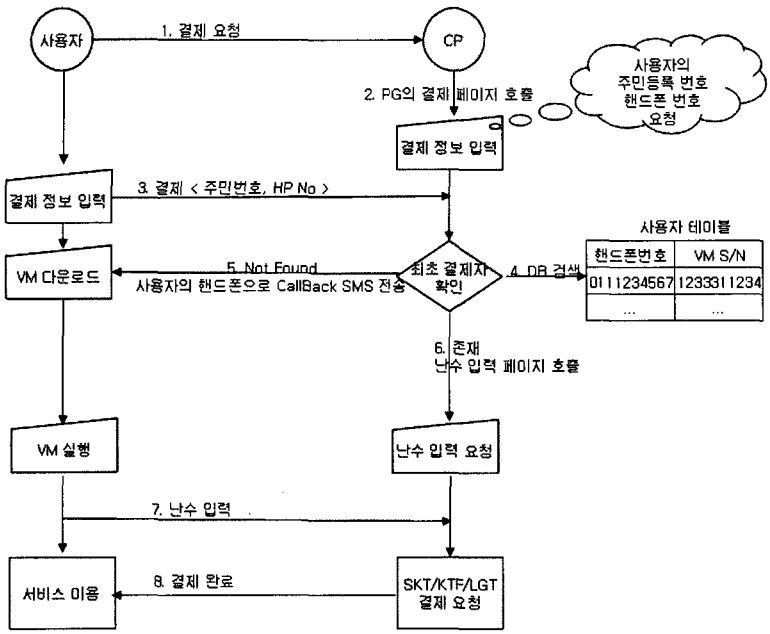


그림 8 VM을 이용한 안전결제 흐름도

B. 사용자에게 난수를 입력을 요청하는 페이지를 호출한다.

7. 사용자는 난수를 얻기 위해 VM 프로그램을 실행한다.
   A. 최초 실행이 아닌 경우, 자동으로 서버와 소켓 통신을 하여 서버로부터 난수를 받아온다. 소켓 통신에는 VM 프로그램의 Version 정보, 사용자의 휴대폰 번호, S/N을 전송한다.
   B. 서버는 사용자의 휴대폰 번호와 S/N이 유효한 경우 휴대폰으로 난수를 결제 요청 테이블로부터 읽어서 전송하여 준다.
   C. 수신된 난수를 폰 화면에 출력한다.
   D. 난수를 결제 페이지에 입력한 후 결제를 요청한다.
8. 결제완료

4.3 서버 및 단말

서버 프로그램에서 가장 중요한 프로세스는 사용자 인증, S/N 발급, 난수 발급, 난수 요청 등 크게 네 가지로 나눌 수 있다. 먼저사용자 인증 부분을 살펴보면 휴대폰 번호를 이용하여 주민번호를 추출하여 이동통신사의 호스트에 접속망 사업자가 제공한 모듈을 이용 실제 가입자에 대한 확인을 하였다. 구현 환경의 제한으로 여기서는 한 통신사와 연동을 하였다. S/N 발급 부분은 먼저 사용자의 프로파일을 확인 사용자 정보를 가져온 후 기존/신규 사용자를 구분하여, 신규사용자인 경우 해쉬 알고리즘인 MD5 함수를 이용하여 S/N을 발급 회원 DB에 삽입하고, 기존 사용자인 경우는 난수값을 업데이트한다.

난수 발급 부분을 살펴보면 그림 9에서와 같이 간단한 알고리즘으로 구성되어있지만, 중복 방지를 위해 시간과 서버의 틱카운트(Tic-Count)를 이용하여 총 6자리 숫자로 구성된 문자열을 구성하였다.

다음으로 난수 요청 부분은 그림 10과 같이 사용자 DB에서 결제요청 여부 확인과 함께 발급된 난수를 불러온 후 패킷 데이타를 통하여 난수를 전송하게 된다. 이때 난수 유효시간 3분을 웹페이지 연동을 통하여 확인 후 제한 시간 내에만 난수를 확인할 수 있도록 한다. 난수를 전송하기 위한 패킷의 구조는 그림 11과 같다.

```
CString CRunThread::MakeRandom()
{    // 인증키(난수) 생성 함수
     CTime t=CTime::GetCurrentTime();
     DWORD dwTick = GetTickCount();

     CString sRnd = "";
     sRnd.Format("%02d%d%02d%d", dwTick%100, (t.GetSecond() / 6)%10, (dwTick/100)%100, (t.GetSecond() % 10));

     return sRnd;
}
```

그림 9 난수 발급 소스 코드

```
bool CRunThread::TransactRandom(PACKET packet)
{ // 결제를 위한 난수(인증키) 값을 발급한다.
  CString sSeq=GetPaymentLog(packet.pktData.pkReqRandom.szPhone, packet.pktData.pkReqRandom.szSerialNo);
  if(sSeq.IsEmpty())
    return     SendAckFailed(4, "결제요청을 먼저하신 후 인증키를 요청하십시오");
      else
      {
          if(ValidPaymentLog((LPSTR)(LPCTSTR)sSeq) == false)
              return SendAckFailed(5, "제한 시간을 초과하여 인증키 생성이 실패하였습니다. 결제요청을 다시하여주십시오");
          else
          {
              CString sRandom = MakeRandom();
              UpdatePaymentLog((LPSTR)(LPCTSTR)sRandom, (LPSTR)(LPCTSTR)sSeq);
              return SendAckRandom((LPSTR)(LPCTSTR)sRandom);     // 난수를 전송한다.
          }
      }
      return false;
}
```

그림 10 난수요청 소스코드

```
#ifndef __DEFSTRUCT_H__
#define __DEFSTRUCT_H__

#pragma pack(1)

// 명령어 종류 (COMMAND TYPE)
enum {
    // vm결제
    CT_REQ_AUTH             = 0x0A01,
    CT_ACK_AUTH             = 0x0AA1,
    CT_REQ_RANDOM       = 0x0A02,
    CT_ACK_RANDOM       = 0x0AA2,
    CT_ACK_FAILED       = 0x0FA1,
};

// HEADER Packet의 사이즈
#define PACKET_HEADER_SIZE     sizeof(short)*2
//HEADER 구성
typedef struct _HEADER
{
    short nCmdType; // 명령어
    short nSize;    // body 사이즈
} HEADER;
//인증 요청
typedef struct _PK_REQ_AUTH
{
    char szVendorNo[4];
    char szPhone[15];
    char szSsn[15];
} PK_REQ_AUTH;
//인증요청 결과
typedef struct _PK_ACK_AUTH
{
    char szSerialNo[32];//시리얼넘버
    char szRandomNo[32];//난수값
} PK_ACK_AUTH;
//난수요청
typedef struct _PK_REQ_RANDOM
{
    char szPhone[15];
    char szSerialNo[32];
} PK_REQ_RANDOM;
//난수요청 결과
typedef struct _PK_ACK_RANDOM
{
    char szRandomNo[32];
} PK_ACK_RANDOM;
//오류발생
typedef struct _PK_ACK_FAILED
{
public:
    short errCode;
    char errMsg[80];
} PK_ACK_FAILED;
//패킷구성
typedef union _PKTDATA
{
    char                sData[100];     // 전체 크기
    PK_REQ_AUTH         pkReqAuth;      // 사용자인증 요청
    PK_ACK_AUTH         pkAckAuth;      // 사용자인증 결과
    PK_REQ_RANDOM       pkReqRandom;    // 난수요청
    PK_ACK_RANDOM       pkAckRandom;    // 난수결과
    PK_ACK_FAILED       pkAckFailed;    // 오류전송
} PKTDATA;

typedef struct _PACKET
{
    HEADER pktHdr;
    PKTDATA pktData;
} PACKET, *LPPACKET;

typedef CList<LPPACKET, LPPACKET> PACKET_LIST;
// 패킷 리스트 정의
#pragma pack()
#endif // __DEFSTRUCT_H__
```

그림 11 패킷의 구조

다음으로 살펴볼 휴대폰 내 VM에서의 프로세스는 매우 간단하게 이루어져있다. VM 프로세스는 먼저 VM의 최초 동작시에 입력받은 주민번호를 가지고 인증 서버에 접속하여 실제 사용자의 정보에 대해 인증하고, 서버로부터 부여받은 S/N을 VM에 저장한 후 서버가 보내주는 난수를 화면으로 디스플레이하거나 에러 메시지를 띄우는 역할을 하게 된다. 휴대폰에서의 동작을 간단하게 정리하면 다음과 같다.

1. 프로그램을 다운받아서 실행한다.
2. VM 실행후 주민번호를 입력받는다.
3. 주민번호와 getProperty에서 받은 휴대폰 번호를 서버에 보낸다.
4. 서버에서 가져온 S/N을 rms에 저장한다.
5. S/N과 휴대폰 번호를 서버에 보낸다.
6. 서버에서 발생시킨 난수를 폰 디스플레이어에 출력한다.
7. rms에 값이 있다면 5번부터 프로세스를 다시 시작한다.

이러한 과정 중 휴대폰에서 진행되는 프로세스 중 가장 중요한 부분은 S/N을 확인하는 과정이며, 이에 대한 소스코드는 그림 12에 나타나 있다.

4.4 구현 및 평가

앞 절에서는 본 논문이 기술한 시스템을 바탕으로 상용화되지 않은 테스트 사이트에 결제수단으로서 구현을 하였다. 이는 기술적인 과정을 구현만 해놓은 것으로서 실제 결제까지는 연동이 되지 않은 상태에 있으며, 특정 통신사에 가입된 모든 종류의 휴대폰에서 실제로 프로그램을 다운받아 가입자 인증, 사용 등록, 승인번호 발

```
                    :
                    :
private final String RMS_STORE_NAME = "PAYMENT"; // RMS의 이름
    private String sSerial; // 시리얼넘버를 받는변수
    private String sRandom; // 난수를 불러오는 변수
    private String sErrorMessage;
    private String sJuminNo = null;// 인증시 주민등록 번호
    private String serial_num;    // 인증받을때 넘어오는 시리얼 넘버

    RmsRecordStore rms; // rms를 불러온다
    ClientPacket cp; // 네트워크 설정클래스를 불러온다
    MyNetwork mn; // 네트워크처리 클래스를 불러온다.
                    :
                    :
                    :

class PaymentCanvas extends Canvas
    {
        public PaymentCanvas()
        {    // 초기화 작업
            lcd_width = this.getWidth();                    // 화면 사이즈의 넓이
            lcd_height = this.getHeight();                  // 화면 사이즈의 높이
            sJuminNo ="";

            rms = new RmsRecordStore(); // rms 클래스를 불러온다 rms에는 시리얼넘버가 저장
            try
            {    // RMS를 연다
                rms.open(RMS_STORE_NAME);
                if(rms.getNumRecords() > 0 )
                {    // rms에 값이 존재하면
                    sSerial = rms.getRecord(1);
                    System.out.println("sSerial : " + sSerial);
                    paymentState = WAITING_RANDOM; // 패스워드를 보여주는 화면으로 이동
                }
                else
                {    // rms에 값이 존재하지 않으면
                    paymentState = INPUT_SSN; // 초기화면으로 이동
                    System.out.println("No values in RMS");
                }
                rms.close(); // RMS를 닫는다.
            }
            catch(Exception e){}
        }                              <이하생략>
```

그림 12 VM 프로세스의 소스코드

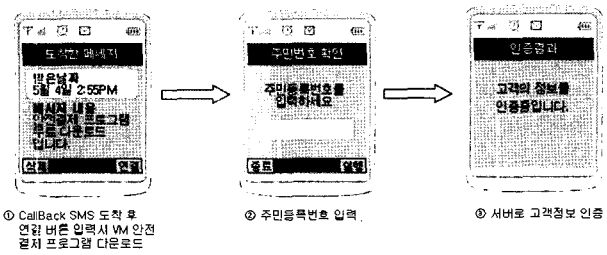
급, 사이버머니 충전까지 모든 과정을 테스트 모듈로 구현하였다.

이 과정에 있어 웹사이트에서의 결제창 동작 순서는 다음의 과정을 거친다. 먼저 신규 사용자의 경우 웹페이지에서 사용자 정보 입력창에서 통신사, 휴대폰 번호, 주민번호를 누른 후 결제요청을 선택하면, 사용자 DB에서 신규/기존 사용자의 여부를 판단하여 신규 사용자일 경우 결제용 VM을 CallBack SMS를 전송한 후 문자메시지 전송 안내창과 함께 사용방법에 대한 설명을 해준다. 다음 절차부터는 기존/신규 사용자 모두 동일한 프로세스로 진행되며, 승인번호 입력창에 부여받은 승인번호를 입력한다. 이때 승인번호는 일회성 암호로서 결제 요청후 3분이 지나면 유효성이 파기되어 다시 결제요청을 통하여 승인번호를 다시 받아야한다. 웹사이트에서의 결제창 동작 순서는 기존의 SMS를 이용한 휴대폰 통합 과금 서비스와 흡사하나 기존/신규 사용자를 구분하여 결제전용 VM을 SMS를 통하여 발급하는 프로세스가 추가된 것이 차이가 있다.

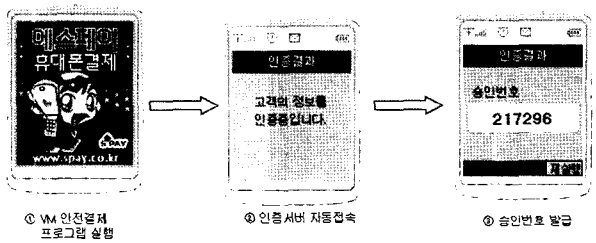
다음으로 설명할 휴대폰에서 VM에서의 동작 과정은 SMS를 이용한 휴대폰 통합 과금 서비스에서는 없었던 과정으로서, 그림 13에서처럼 (a) 신규사용자 VM 다운로드 과정과 (b) 승인번호 발급과정으로 나뉘어진다. 신규사용자는 그림 13의 (a)에서 보듯이 전송된 CallBack SMS를 통해 결제전용 VM을 다운받고, 주민번호를 입력하여 사용자 인증을 받게 된다. 이 과정에서 본 논문에서 제안한 고유 S/N을 발급하여 VM과 서버의 사용자 DB에 저장하고, 차후 결제시에 이 S/N을 사용하게 된다. 이렇게 최초 인증을 받은 사용자는 그림 13의 (b)와 같이 VM 동작 과정을 거쳐 승인번호 발급을 받아 웹의 결제창에 입력을 하게 된다.

본 논문에서 제안한 VM을 이용한 안전결제는 기존 소액결제용 휴대폰 통합 과금 서비스에서 가장 많이 사용하는 SMS를 이용한 결제와 비교하여볼 때, 휴대폰 불법복제나 잘못된 결제를 하려고 할 때에 신규가입과정에서 VM에 발급한 S/N 확인을 통하여 한 대 이상의 폰으로 결제를 원천적으로 막을 수 있는 보안 과정을 추가하였다. 이는 결제를 원하는 사용자 본인이 직접 VM을 구동하여 승인번호를 발급받아야만하기 때문에 휴대폰 소유자 본인이 모르는 사이에 결제가 이루어지는 것을 막을 수 있다.

또한 기존의 SMS를 이용한 결제에서는 사용자가 결제를 요청한 후 승인번호를 받기까지 10초~2분 이상을 수동적으로 기다리거나 SMS 사용이 빈번한 시간대에는 결제유효시간이 지난 후에 SMS가 도착이 되어 다시 결제요청을 해야 하는 불편한 점이 있었다. 하지만 VM을 이용한 안전결제는 결제요청시 사용자가 능동적으로



(a)



(b)

그림 13 VM 동작순서 (a) 신규사용자 VM다운로드 과정 (b) 승인번호 발급과정

VM을 동작하여 1~3초 이내에 바로 승인번호를 받을 수 있어 결제에 소요되는 시간을 크게 줄였다. 또한 결제 서비스에 소요되는 비용에 있어서는 1회 SMS를 이용 결제할 때 사용하는 SMS 비용을 결제대행사가 결제 1건당 최소 13원~15원 정도를 지불하여야 되었으나 VM을 이용한 안전결제는 47 byte 패킷 통신으로서 통신료 5원으로 승인번호를 발급받을 수 있어 기존의 방식보다 훨씬 경제적이다.

5. 결 론

본 논문에서는 최근 급성장하고 있는 모바일 지급결제 수단을 살펴보고 모바일 지급결제 수단 중에서 가장 많이 사용되고 있는 휴대폰 소액결제방법을 기존의 SMS를 이용한 결제방식을 벗어나 휴대폰의 VM을 이용 패킷 통신을 이용하는 결제방법을 제안, 구현 하였다. 제안한 결제방식은 단순히 매개체가 SMS인지 휴대폰의 VM인지의 차이가 아니라 결제를 원하는 사용자가 결제용 VM을 다운받았을 때에 주민번호 인증과 함께 VM에 S/N을 부여, 관리하여 결제 요청시에 기존에 등록한 정상적인 폰에서의 요청인지 아니면 불법 복제된 폰에서의 잘못된 요청인지, 혹은 폰 번호나 명의가 바뀐 휴대폰에서의 요청인지 등을 구분하여 승인번호를 발급함으로써 잘못된 결제를 막는 기능을 추가하였다.

제안한 VM을 이용한 안전결제는 기존 방식에 비하여 강화된 보안 이외에도 사용자가 결제를 요청한 후 승인번호 발급까지 소요되는 시간을 1~3초 이내로 줄였으며, 난수 발급을 수동적으로 기다리는 사용자의 입장을 빠른 시간 내에 직접 확인할 수 있도록 하였다. 또한 결제시에 부가적으로 발생하는 SMS 발송 비용을 47 byte 패킷통신을 통해 5원 이하로 대폭 낮추었다. 그러나 본 논문 구현시에는 통신사와의 업무 협의를 하지 않고 진행하여 그 패킷 통신료는 현재 사용자가 부담하게 되어있다. 이 패킷 통신료와 새로운 결제방식을 알리고 보급하기 위한 비용과 노력이 수반되어져야 한다는 점이 상용화시에 풀어야 할 과제로 남아있다.

앞으로 휴대폰을 이용한 모바일 지급결제는 좀 더 간편하고 안전한 서비스로 발전할 것이다. 물론 하드웨어 기술의 발달으로 많은 결제정보와 암호화키가 휴대폰 속에 저장되어 진보된 휴대폰과 시스템이 개발될 것이다. 그러나 기존에 보급된 휴대폰이 교체되어 서비스가 정착되기까지는 많은 시간과 노력이 필요할 것으로 예측된다. 따라서 모바일 지급결제의 개발 및 보완이 한창인 현 시점에서는 기존에 보급된 휴대폰과 시스템을 가지고 구현 가능한 보다 안전하고 보다 간편한 인증방식에 대한 시도와 연구개발이 필요한 분야가 모바일 지급결제 분야이다.

참 고 문 헌

[1] 김성현, "인터넷 기반산업으로서의 지불결제 서비스 시장의 구조 및 전망", 정보통신정책연구원 연구보고서, 01-19, 2001년 12월.

[2] Krueger, Malte, Kund Bohle, "Payment Culture Matters," IPTS, August 2001.

[3] 신성문, "모바일인터넷 지급결제 시스템의 구조", 정보통신정책원구원, 연구보고서 2000년 2월.
[4] 김희수, "모바일 지급결제 시장동향과 정책이슈", KISDI, 2003년 10월.
[5] 김시홍, "국내 Mobile-Payment 시장의 현황과 경쟁·제휴전략", KISDI, 2003년 2월.
[6] 장병환, "스마트카드 기술동향", 지급결제와 정보기술, 금융결제원, 2003년 5-6월호.
[7] 오은숙, "비접촉식 지급결제서비스 현황 및 전망", 지급결제와 정보기술, 금융결제원, 2003년 7-8월호.
[8] 장진성, "온라인 프라이빗 뱅킹 서비스 동향", 지급결제와 정보기술, 금융결제원, 2003년 3-4월호.
[9] 금융결제원, "이동통신회사의 지급결제관련 서비스 현황", 지급결제정보 제 2003-9호, 2003년 9월.
[10] 정보통신부, "정보통신망 이용촉진 및 정보 보호 등에 관한 법률 시행규칙", 2004년 7월.

강 경 석

2002년 2월 광운대학교 전자공학부 졸업
2006년 2월 광운대학교 전자정보통신공학 석사. 현재 (주)하렉스인포텍 연구소 근무

민 상 원

1988년 2월 광운대학교 전자통신공학과 졸업. 1990년 2월 KAIST 전기 및 전자공학과 석사. 1996년 2월 KAIST 전기 및 전자공학과 박사. 1990년 1월~1999년 2월~LG 정보통신. 1999년 3월~현재 광운대학교 전자통신공학과 부교수

심 상 범

2005년 2월 광운대학교 전파공학과 졸업
2005년 3월~현재 광운대학교 전자통신공학과 석사과정