이동통신 환경에서 신뢰할 수 있는 서비스별 과금 방법

김순석*

Reliable Billing Schemes for Service Types in Mobile Communication Environments

Soon-Seok Kim*

요 약

본 논문에서는 사용자가 이동 단말기를 이용하여 유료 컨텐츠에 대해 부가 서비스를 제공받을 경우 사용자와 컨텐츠 제공자간에 상호 신뢰할 수 있는 과금 방법들을 제안하고자 한다. 제안하는 방법은 이동통신 환경에서 단문 메시지, 벨소리, 이미지와 음악 데이터 전송, 그리고 게임과 같은 다양한 종류의 서비스에 대해서 과금 방법을 지원한다. 또한 암호학적인 해쉬 체인 방법을 이용하여 이동 단말기의 계산 부하를 줄이고 사용자와 컨텐츠 제공자간에 전송되는 데이터 량을 줄였으며, 각 사용자들의 컨텐츠 이용에 대한 증거를 저장하지 않고도 신뢰성 있는 과금이 가능하기 때문에 컨텐츠 제공자 측에서 메모리 공간을 절약할 수 있다.

ABSTRACT

In this paper we propose reliable billing schemes between users and contents providers where a user is provided value-added service for a paid contents using mobile terminals. Our schemes support various types of services such as short messages, bell sounds, images or music data transmissions, and games in mobile communication environments. Using hash chain method, we also reduced the computational overhead of mobile terminals and the volume of data transmitted between the user and the content provider. Content providers can save memory space because they don't need to store each user's usage evidence but still can charge.

키워드

Mobile communication, Billing, Value added service, Cryptographic Protocol

Ⅰ. 서 론

1990년대 초반부터 널리 보급된 무선호출기로부터 시작하여 셀룰러폰(celluar phone), 개인 휴대 전화기(PCS)를 거쳐 조만간 상용화될 IMT 2000 기반의 이동통신에 이르기까지 최근 몇 년 사이에 이동 통신 환경은 매우 빠른 속도로 발전하고 있으며 더불어 단말기를 통해 가능한 부가 서비스에 대한 모색도 현재 활발히 전개되고 있다. 특히 이러한 부가 서비스를 이용함에 있어 서비스를 제공하는 컨텐츠 제공자와 무선 단말기를 가지고 여러 서비스를 이용하려는 사용자 사이에 일어나는 서비스 이용에 따른 과금에 대한 문제가 최근 이슈가 되고 있다. 현재와 같이 일방적으로 컨텐츠 제공자의 로그 정보에만 의존해서 작성되는 이용요금 통지는 바람직하지 않다. 이는 실제로 사용자가 이용한 시간 혹은 횟수와 다르게 과

*한라대학교 정보통신공학부

접수일자 : 2003. 8. 1

금이 되는 경우가 있기 때문이다. 만약 악의를 가진 컨텐츠 제공자에 의해서 로그 정보가 조작된다면 어떻게 되겠는가? 이런 경우에 사용자는 자신이 이용한 서비스에 대해 증거를 제시하기가 힘들며 또 증거를 제시하기 위해 많은 노력이 필요하다. 즉, 사용자와 컨텐츠 제공자 사이에 신뢰할 수 있는 과금을 위해 어느 한 쪽만이 서비스 이용에 대한 증거를 가지는 것은 바람직하지 못하며 무선 단말기를 이용해야한다는 특수한 상황에 따른 제반적인 문제들도 고려해야 한다. 따라서 사용자와 컨텐츠 제공자가 서로를 신뢰하여 사용자는 컨텐츠 제공자가 요구하는 사용금액을 믿을 수 있으며 만일 사용자가 이 사실에 대해 부인할 경우 컨텐츠 제공자는 또한 이에 응할 수 있는 증거를 가져야 한다.

이동통신에서의 과금과 관련하여 현재까지 알려진 다수의 논문[1,2,3,4]들은 주로 모바일 사용자가 이용하는 컨텐츠에 대한 양 즉, 컨텐츠 제공자가 제공하는 컨텐츠의 데이터 량을 기준으로 한 과금 방법들을 제안해오고 있다. 그러나 현재 제공되고 있는 서비스들을 살펴보면 우리나라와 같이 각 이동통신사에서 미리 정한 일정 시간 단위로 요금을 부과하거나 서비스를 제공한 횟수별 또는 정액제 등 실제 제공되는 서비스의 유형에 따라 여러 가지로 그 요금을 부과하고 있다. 따라서 본 논문에서는 특정 데이터량을 기준으로 한 것이 아닌 이러한 다양한 서비스의 유형에 따른 과금 방법들을 제안하는데 그 목적이 있다.

본 논문에서는 사용자가 무선 단말기를 이용하여 유료 컨텐츠에 대한 부가 서비스를 제공받을 경우 사용자와 컨텐츠 제공자 사이에 상호 신뢰할 수 있는 각종 서비스 유형별 과금 방법을 제안하고자 한다. 본 논문의 2장에서는 제안하는 방법과 관련한 연구동향에 대해 살펴보고, 3장에서는 사용자와 컨텐츠 제공자사이에 상호 신뢰할 수 있는 과금 방법을 제안한다. 4장에서는 제안한 과금 방법을 비교 및 분석해보고, 5장을 끝으로 결론 및 향후 연구 방향에 대해 기술하고자 한다.

II. 관련 연구

최근 몇 년 사이에 제안된 논문들은 대부분 Pederson[5]이 제안한 tick payment 또는 micropayment라 불리우는 암호학적인 해쉬 체인(hash chain)을 이용한 방법을 기반으로 하고 있다. 이 방법을 간략히 설명하면 다음과 같다. 예를 들어, 일정 단위의 쿠폰을 이용하여 지불하는 경우 1,000원은 100원 짜리 쿠폰 10개로 지불할 수 있다. 이처럼 물건을 사는 사람과 파는 사람은 이미 정해진 일정한 단위의 쿠폰을 이용하여 지불하는 방법을 생각할 수 있다. 암호학적인 일방향 해쉬 함수를 f라 하고 f의 초기 값을 a라 하자. 이때 총 T개의 쿠폰이 있다고 할 때 i(i는 1과 T사이의 정수이다)번째 쿠폰 $a_i=f^{T-i}(a)$로 계산할 수 있다. 따라서 구매자는 $a_i=f^{T-i}(a)$를 계산하여 판매자에게 전달하고 판매자는 전달받은 해쉬 결과값을 $a_i=f(a_{i-1})$의 관계를 이용하여 구매자가 제시한 쿠폰의 수를 확인한다. 즉 암호학적인 해쉬 체인의 방법은 판매자에게 현 해쉬값을 부여하고 이에 대한 입력값을 구매자가 계산하여 판매자에게 보냄으로써 판매자가 이를 확인하는 방법을 연속적으로 진행하는 방법이다. 이러한 방법은 프로토콜이 비교적 간단하고 빠른 특성이 있어 이동통신과 같은 무선 분야에 소액 지불방법으로 현재 널리 이용되고 있다. 또한 이 방법은 차세대 이동통신 시스템인 UMTS에 적용될 보안기술을 연구 개발하는 ASPeCT 프로젝트[1,2]에서 모바일 이용자에게 tick payment를 이용하여 부가 서비스를 제공하는 방식으로 제안된 바 있다. 아울러 본 논문에서도 이 방법을 기본적으로 채택하고 있다.

Zhou와 Lam[3]이 제안한 방법은 모바일 사용자와 통신 서비스 제공자 사이에서 사용자 자신에게 부과된 요금이 실제 사용량에 맞게 과금 되었는지에 대한 분쟁 발생 시 이를 해결할 수 있는 방법을 제안하였다. 또한 좀더 확장하여 사용자가 외부 도메인에서 통신을 시도할 경우에 일어날 수 있는 요금 분쟁 즉, 자신이 등록한 홈 도메인이 아닌 외부 도메인에서 로밍 서비스를 받을 경우에도 사용자에게 올바른 과금을 하도록 하는 방법을 제안하였다. 사용자는 자신이 외부 도메인에 있더라

도 통신을 할 수 있으며 홈 도메인과 외부 도메인 사이의 통신을 통해서 올바른 사용자인지를 확인할 수 있다. 사용자가 자신이 사용한 통화에 대한 증거를 외부 도메인에게 보낼 때 외부 도메인은 일정하게 서비스에 대한 요청을 하고 사용자는 이에 대한 증거를 계속 생성하여 전달한다. 이러한 메시지 교환으로 추후 요금에 대해 사용자와 외부 도메인 사이에 상호 부인을 방지할 수 있는 방법을 제안했다. 이 방법 또한 앞서 언급한 암호학적인 해쉬 체인의 방법을 그대로 이용하고 있으며 본 논문에서도 이러한 부인 방지 서비스를 기본적으로 지원하고 있다.

그밖에 Patel과 Crowcroft[4]는 모바일 사용자가 특정 서비스 제공자에게 서비스를 받고자 할 경우 티켓 서버로부터 제공된 티켓을 이용하여 서비스를 받을 수 있도록 제안하였다. 이때 제공된 티켓은 인증기관으로부터 인증서를 부여받아 인증된 티켓을 사용한다. 여기서 사용된 티켓의 내용에 대한 구체적인 정보는 없으나 아마도 서비스를 이용할 수 있는 권한을 획득한 것으로 생각된다. 이러한 티켓 기반의 과금 방식은 Buttyan과 Hubaux[6]의 논문에서도 제안되고 있는데 여기서 말하는 티켓은 앞서 언급한 tick payment 방식에서 암호학적인 해쉬 체인에 대한 정보라든가 티켓의 아이디 그밖에 키 정보 등이 저장되어 있다.

지금까지 살펴본 논문들은 주로 암호학적인 해쉬 체인을 이용한 과금 방식이 주류를 이루고 있다. 즉 이 방식은 소액의 각 쿠폰에 해당하는 금액만큼 이에 해당하는 서비스를 제공하는 방식이다. 그러나 실생활에서 현재 일어나고 있는 과금 방식은 서비스 이용 시간이라든가 서비스 이용 횟수 혹은 일정 기간에 n원 하는 식으로 그 유형들이 다르게 전개되고 있다. 따라서 본 논문에서는 이러한 암호학적인 해쉬 체인 방법을 부분적으로 이용하되 과금 방식을 기존과 달리 실제 발생할 수 있는 서비스 유형별로 나누어 그 각각에 대해 프로토콜을 제안하고자 한다.

III. 새로운 과금 방법 제안

새로운 과금 방법을 제안하기 전에 먼저 현재 이용중인 과금 시스템에 대해 살펴보자. 이동통신 단말기 이용자는 먼저 컨텐츠를 이용하기 위해 각 단말기에 내장된 특별한 브라우저가 필요하다. 현재 세계적으로 가장 많이 이용되는 환경은 JVM (Java Virtual Machine)환경이다. JVM 환경을 지원하는 핸드폰이나 PDA와 같은 무선 단말기에서 벨소리나 이미지에서부터 게임에 이르기까지 여러 가지 데이터와 서비스를 전송 받아 사용할 수 있다. 따라서 기본적으로 각 사용자의 단말기는 이러한 브라우저와 환경을 지원한다고 가정한다.

사용자는 브라우저나 가상머신 환경에서 네트워크 제공자나 컨텐츠 제공자로부터 필요한 데이터나 프로그램을 전송 받을 수 있다. 이때 사용자가 전송 받은 데이터나 프로그램은 게임이나 실시간으로 감상할 수 있는 음악 또는 동영상일 수 있으며 또 가상머신 상에서 수행 가능한 프로그램일 수 있다. 이러한 전송 서비스를 제공하는 네트워크 제공자나 컨텐츠 제공자는 사용자에게 서비스에 대한 비용을 청구하게 되는데 초기에 무료로 제공되는 서비스와는 다르게 차이가 있다. 즉 각 서비스마다 요금도 다르고 또 서비스의 종류도 점차 늘어가고 있는 추세이다. 이렇게 사용자가 컨텐츠 제공자로부터 서비스를 이용할 경우 컨텐츠 제공자는 사용자에게 서비스를 제공하고 사용자가 이용한 시간이나 혹은 데이터의 양에 따라서 요금을 청구하게된다.

본 논문에서 제안하는 방법은 사용자와 컨텐츠 제공자 사이에서 사용자가 컨텐츠 제공자에게 서비스를 요구하고 서비스를 받는 과정에서 일어나는 과금에 초점을 둔다. 현재의 과금 시스템과는 다르게 사용자 자신이 이용한 서비스에 대한 증거를 생성하며, 컨텐츠 제공자는 이 증거를 저장한다. 따라서 사용자가 요금에 대한 증거를 요구할 경우 컨텐츠 제공자는 저장하고 있는 증거를 제공하므로 사용자는 컨텐츠 제공자를 신뢰할 수 있다.

3.1 요구사항

이동통신 환경에서의 과금 시스템은 다음과 같은 요구사항을 가진다.

- 단말기 내에 있는 브라우저에서 사용이 가능해야한다.

가장 기본적인 요구사항으로 핸드폰, PDA와 같은 무선 단말기에서 작동이 가능해야한다. 일반 PC에서 사용하는 브라우저와는 구조, 기능, 성능면에서 많은 차이점을 가지고 있다. 따라서 무선 단말기에서 사용하는 브라우저의 특징에 맞는 과금 시스템을 제안해야 한다.

- 사용자측 단말기의 계산량을 최소화해야한다.

핸드폰이나 PDA와 같은 무선 단말기는 계산 능력에 한계가 있다. 따라서 컨텐츠 제공자에게 부과되는 계산량 보다는 사용자측 단말기에 부과되는 계산량에 관심을 둔다. 즉 사용자 쪽에서 계산량이 늘어난다면 사용자의 응답속도가 느려지기 때문에 효과적이지 못하다.

- 사용자는 컨텐츠 제공자가 제시한 사용 내역이 정확한지, 그 판별이 가능해야한다.

현재 제공하고 있는 서비스에서는 컨텐츠 제공자가 일방적으로 과금의 결과를 통보한다. 사용자가 과금에 대한 이견이 있을 경우에는 어떻게 해야하는가? 사용자는 네트워크 제공자에게 사용자 자신의 네트워크 사용시간을 확인 받고 그 시간과 컨텐츠 제공자가 제공한 시간을 비교해야 한다. 이 방법은 사용자와 서비스 제공자 사이의 과금 방식에 명확한 해답이 될 수 없다. 따라서 사용자가 생성한 증거를 가지고 판별이 가능해야한다.

- 사용자의 익명성을 보장해야한다.

컨텐츠 제공자가 사용자에 대한 정보를 아는 것은 바람직하지 못하다. 컨텐츠 제공자가 만일 현재 접속한 사용자가 누구인지를 안다면 사용자의 프라이버시를 침해할 수 있다. 과금 시스템에서는 사용자의 정보를 되도록 적게 컨텐츠 제공자에게 노출시키는 것이 바람직하다.

- 여러 가지 과금 방법에 대한 지원이 가능해야한다.

서비스 종류에 따라서 여러 가지 과금 방법이 존재한다. 여러 과금 방법의 예로는 일정한 시간 간격으로 과금이 되는 경우, 시간에 관계없이 서비스를 받는 횟수에 따라 과금이 되는 경우, 동영상과 같이 실시간으로 스트림(stream) 서비스를 받아 과금하는 경우, 그리고 정액제로 과금하는 방법이 있다. 이런 여러 가지 과금 방법에 대해서 적용 가능한 방법을 제공해야한다.

3.2 기본 가정과 표기

먼저 본 논문에서 제안하는 방법이 안전하게 수행되기 위해서는 다음의 가정이 필요하다.

- 무선 단말기의 내부에는 사용자를 포함하여 그 누구로부터의 임의 접근이나 변경이 불가능해야한다.

핸드폰이나 PDA와 같은 무선 단말기의 메모리나 기타 내부 하드웨어에 과금과 관련된 프로그램이 저장되고 또 수행된다. 그러므로 내부 하드웨어에 임의로 접근이나 조작하는 행위를 막을 수 있어야 한다. 임의로 접근하여 프로그램이나 하드웨어의 조작이 가능하다면 악의를 가진 사용자는 사실과 다른 사용에 관한 증거를 가지게 된다.

- 제 3의 신뢰할 수 있는 기관이 존재한다.

사용자의 비밀키(private key)와 공개키(public key)쌍을 저장하고 있으며 사용자와 컨텐츠 제공자가 신뢰할 수 있는 제 3의 기관이 존재한다. 컨텐츠 제공자가 사용자의 서명을 확인하기 위해 사

용자의 공개키를 요청할 때 올바른 키를 분배하는 역할을 담당한다. 여기서 제 3의 기관은 인증기관이나 신뢰할 수 있는 네트워크 제공자일 수 있다.

- 사전에 사용자와 컨텐츠 제공자의 인증 및 키분배가 일어난다.

위 두 번째 가정에서 제 3의 신뢰할 수 있는 기관을 가정하였다. 본 논문에서는 서명에 사용하는 사용자의 비밀키와 컨텐츠 제공자가 받은 서명을 확인할 때 사용하는 검증키인 공개키는 과금 프로토콜 진행 이전단계에서 신뢰할 수 있는 기관을 통해 사용자와 컨텐츠 제공자에게 제공된다고 가정한다. 사용자에 대한 응답 속도를 높이기 위해 컨텐츠 제공자는 사전에 사용자들의 공개키를 데이터베이스에 저장 가능하며 한 번 접속했던 사용자에 대한 공개키를 저장하고 있다고 가정한다. 이러한 인증 및 키분배 과정은 현재 여러 방법들이 이전에 발표된 논문들[1,2,4,6,7,8]에 제안된 바 있으며, 본 논문에서도 이들 방법을 기본적으로 이용한다고 가정한다. 이러한 인증 및 키분배 과정은 보안상 중요한 부분이다. 그러나 본 논문에서 실제 기술하는 프로토콜은 인증 및 키분배 과정보다는 실질적인 과금 방법에 그 초점을 두고 제안하였음을 재차 언급한다.

아울러 제안하는 방법에서 사용하게 될 각 기호들에 대한 정의는 다음과 같다.

- TP : 신뢰할 수 있는 제 3자(Trust Party).
- User : 무선 단말기를 이용하여 컨텐츠 제공자가 제공하는 서비스를 이용하는 사용자.
- CP : 컨텐츠 제공자(Contents Provider).
- UID[1] : User의 실제 아이디가 아닌 임시 아이디로 익명성 보장을 위해 이용된다. 예를 들어, 유럽의 2세대 이동통신시스템 표준인 GSM시스템[9]의 경우 TMSI(Temporary Mobile Subscribe Identity)가 될 수 있다.
- U_{sk} : 전자서명 알고리즘에 사용하는 User의 비밀키(Private Key)로, 본 논문에서는 DSA나 KCDSA와 같은 표준 전자서명 알고리즘에 이 키를 적용할 수 있다.
- Time : User가 처음 서비스를 요구할 때의 시간.
- *cur_time* : CP가 User에게 전달하는 현재 시간.
- Service : User가 CP에게 요구하는 서비스 내용을 담은 일종의 명세서.
- <isAccepted> : User의 서비스 요청에 대한 응답으로 해당 컨텐츠에 대한 제공의사가 있고, 이에 따른 제반 환경이 준비되었음을 나타내는 의미의 단순한 응답 메시지이다.
- $S_u(M)$: 메시지 M을 User의 비밀키로 서명.
- $f(M)$: 메시지 M에 대한 암호학적인 일방향 해쉬함수, MD5나 혹은 SHA를 사용할 수 있다.
- Proof Request : 일정시간이 경과했을 때 CP가 User에게 사용에 대한 증거를 요구하는 메시지.
- Exit : User가 더 이상 서비스 이용을 원치 않는다는 의미로 CP에게 전달하는 메시지.
- Nak : CP가 User에게 보낸 컨텐츠에 대해 연결이 끊어졌거나 하는 이상이 있을 경우 재요청을 위해 User가 CP에게 전달하는 메시지이다. 이동통신 환경하에서 User가 이동단말기를 사용하다보면 유선 환경과는 달리 접속이 자주 끊어질 수가 있다. 따라서 본 논문에서는 Nak 메시지를 이용하여 이러한 경우를 보완하고자 한다.
- $Data_t$: User가 CP에게 t+1번째 요청한 Service에 대해 CP가 제공하는 자료로 일종의 유료 컨텐츠를 말한다. 이때 t의 초기값은 0이다.

3.3 제안하는 서비스별 과금 방법

본 절에서는 과금의 유형을 크게 일정 시간단

1) 이 값은 실제 적용될 때 다양하게 생성될 수 있으며, User와 CP 모두가 사전에 알고 있는 값으로 가정한다. 예를 들어, GSM 시스템의 경우, 사용자의 송수신 서비스를 담당하는 통신서비스 제공자(Network Provider)가 임시로 랜덤 아이디인 TMSI를 생성하여 User와 CP에게 부여함으로써 두 개체가 이 값을 공유할 수 있다. TMSI 생성과 이에 관한 자세한 설명은 [9]를 참조하기 바란다.

위, 1회용 서비스, 스트림 단위, 정액제의 네 가지로 분류하여 각각에 대한 과금 방법들을 기술하고자 한다.

3.3.1 일정 시간 단위로 과금하는 경우

핸드폰 및 전화를 사용할 경우에 각 통신사에서는 미리 정한 일정 시간 단위로 요금을 부과하는 경우가 있다. 현재 우리나라에서 이 방법을 가장 많이 이용하고 있는데 예를 들어, 10초, 1분, 3분 등과 같이 CP가 지정한 시간 단위로 요금을 부과하는 방법이다. 이 경우 CP는 미리 정한 시간(이를 w라 하자)이 경과할 때마다 계속된 다음 서비스 제공을 위해 User에게 서비스 사용에 대한 증거를 요구하며, User 또한 자신의 사용에 대한 증거를 자신의 단말기에 저장한다. 이것은 나중에 분쟁이 발생할 경우에 일종의 영수증으로 이용할 수 있다.

[단계 1] Initialization

먼저 User는 자신의 UID, Time, 그리고 Service에 대한 정보를 생성한 후 CP에게 전달한다. 이 정보를 받은 CP는 사용자의 UID를 확인하고 Time을 체크한다. 특히 Time을 체크할 경우에 통신상에서 있을 수 있는 시간 오차를 감안한다. 그리고 Service를 확인한 다음 컨텐츠를 제공할 준비가 되었음을 나타내는 <isAccepted> 메시지를 이용하여 User에게 전달한다.

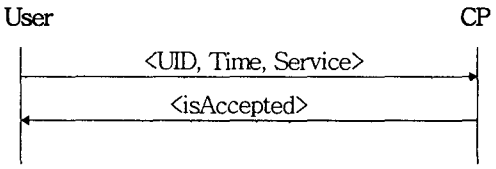

그림 1. 초기화
Fig. 1 Initialization

[단계 2] Setup Operation

CP로부터 <isAccepted> 메시지를 전달받은 User는 CP가 컨텐츠를 제공할 환경이 준비되었음을 인지하고 자신의 UID와 Time, 그리고 Service를 연접하여 서명한 후, 이 서명과 자신의 비밀키 U_{sk}를 연접하여 암호학적 해쉬 연산을 수

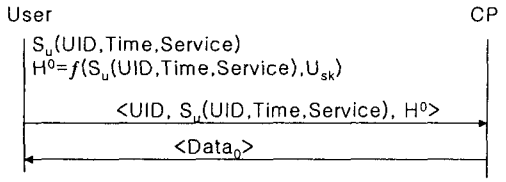

그림 2. 셋업 연산
Fig. 2 Setup Operation

행한다. 이 결과를 H^0라 하자. 연산이 끝난 후에 User는 자신의 UID, 서명 결과, 그리고 H^0를 연접하여 CP에게 보낸다. CP는 먼저 수신한 메시지에서 User의 UID를 확인한 다음 User의 공개키를 이용하여 서명을 확인한다. 이 과정에서 UID가 [단계 1]에서 받은 것과 동일한지 그리고 Time과 Service가 올바른지를 확인한다. 이때 해쉬 연산의 결과는 User에게 서비스를 제공했다는 증거로 사용되므로 CP는 이 값을 저장한다. 확인 결과 만일 User가 보낸 메시지에 이상이 없다면 CP는 User가 원하는 컨텐츠인 $Data_0$를 User에게 전달한다. 그러나 만약 중간에 연결이 끊어졌거나 제대로 수신하지 못했을 경우에는 CP에게 Nak를 보내 $Data_0$를 재요청한다.

[단계 3] Regular Operation

CP가 정한 일정 시간 w가 경과한 직후, CP는 User에게 계속된 다음 서비스를 제공하기 위해 사용에 대한 증거를 요구한다. 즉 Time과 Proof Request 메시지를 User에게 보낸다. 메시지를 받은 User는 먼저 CP가 보낸 Time과 자신 단말기의 진행 시간을 비교하여 CP가 보낸 시간이 올바른지를 확인한다. 만일 이상이 없다면 증거 제시를 위해 이전 단계에서 보낸 H^0와 자신의 비밀키 U_{sk}를 연접하여 암호학적인 해쉬 연산($f(H^0)$)을 수행, 그 결과인 H^1을 CP에게 보낸다. 증거 H^1을 받은 CP는 이전에 전달받은 H^0를 이용하여 $H'(=f(H^0))$을 계산하여, H'과 증거 H^1이 동일한지를 확인한 다음, 이에 따른 컨텐츠인 $Data_1$을 User에게 전달한다. 만일 User가 $Data_1$값을 전달받는 과정에서 연결이 끊어졌거나 제대로 수신하지 못했을 경우에는 CP에게 Nak를 보내 $Data_1$을

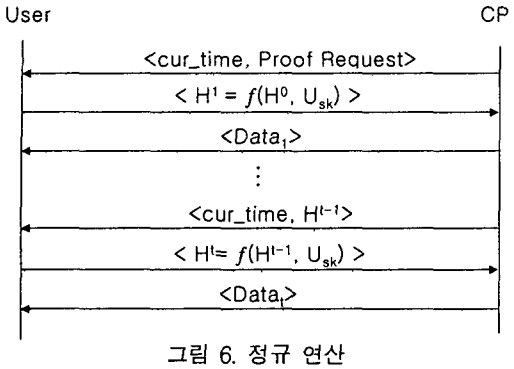

그림 6. 정규 연산
Fig. 3 Regular Operation

재요청한다. 지금까지 설명한 과정은 w시간이 경과한 직후마다 그림 3과 같이 t번 반복적으로 수행된다. 다만 CP가 다음 요청을 보낼 때부터는 기존 cur_time과 Proof Request를 보내던 것을 Proof Request 대신 가장 최근에 CP가 받은 해쉬값을 보낸다.

[단계 4] Proof Generation

만일 User가 더 이상 서비스 제공을 원치 않을 경우 User는 CP로부터 증거 요청시 즉, CP가 cur_time과 H^{t-1}(그림 3참조) 메시지를 보내올 때 이에 대한 응답으로 다음 증거를 생성하는 대신 Exit 메시지를 CP에게 보낸다. 이는 CP로부터 서비스 제공이 끝난 경우이므로, CP는 사후 분쟁에 대비하여 [단계 2]에서 받은 서명과 더불어 암호학적 해쉬 함수의 초기값 H^0와 마지막 값인 H^t값을 증거로 보관하고 이에 따른 요금 청구서를 작성한다. 이때 User 또한 자신이 사용한 서비스에 대해 Time과 H^t를 그 증거로 보관한다.

여기서 실제 지불되는 요금은 예를 들어, CP가 받은 최종 증거를 H^t라하고 일정시간 w에 대해 v원씩 부과한다고 할 경우 t*v원이며 이 요금은 그 달의 끝에 CP가 User에게 청구할 수 있다.

제안한 방법에서 CP는 w시간 간격으로 전달받은 증거를 임의로 생성할 수 없다. 왜냐하면 암호학적인 해쉬 함수의 일방향성 때문에 결과를 가지고 역으로 입력값을 유추할 수 없기 때문이다. 만일 CP가 해쉬의 입력으로 Time 정보를 안다 할

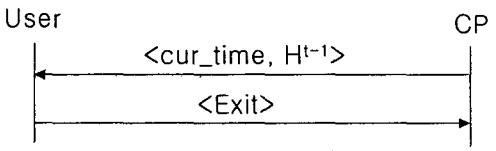

그림 7. 증거 생성
Fig. 4 Proof Generation

지라도 U_{sk}를 알 수 없기 때문에 증거를 마음대로 생성할 수 없다. 따라서 User는 CP가 제시하는 요금을 믿을 수 있다. 그러나 만일 User가 U_{sk}를 영구적으로 이용할 경우 CP는 무작위(brute-force) 공격 등의 방법을 통해 U_{sk}를 알아내 User가 제시한 증거를 조작할 수 있다. 따라서 U_{sk}에 대한 키 신규성(refreshness) 보장을 위해 주기적으로 U_{sk}를 바꿀 필요가 있다. 이 과정은 앞서 3.2절에서 가정한 키분배 과정에서 일어난다고 가정하며 이후 프로토콜에서도 동일하게 적용된다. User 또한 앞서 가정에서 언급한대로 단말기 내부에 임의 접근이 불가능하기 때문에 U_{sk}를 위변조 할 수 없다. CP 역시 User가 제시하는 증거를 믿을 수 있다.

만일 요금에 대한 분쟁이 생겼을 경우를 가정해 보자. CP는 먼저 [단계 2]에서 받은 User의 서명을 증거로 제시할 수 있으며 User가 사용한 시간에 대해서는 User가 연산한 해쉬값을 가지고 해결할 수 있다. 즉 암호학적인 해쉬의 초기값 H^0와 마지막 값인 H^t값을 이용하여 User가 얼마 동안의 서비스를 받았는지를 확인할 수 있다. User 또한 보관하고 있던 Time과 H^t를 증거로 CP에게 제시함으로써 사후 분쟁을 해결할 수 있다.

3.3.2 1회용 서비스에 대해 과금하는 경우

이 방법은 위 3.3.1보다 단순한 방법으로 User가 핸드폰을 이용하여 벨소리나 이미지 혹은 문자와 같은 서비스들을 이용하는 경우에 가장 많이 사용하는 방법이다. User는 무선인터넷을 통해 여러 CP들로부터 제공되는 모바일 서비스를 받게 되는데 그중 어느 한 CP로부터 1회용 벨소리나 이미지를 다운받는 경우이다. 즉 일정시간이나 월 단위의 과금이 아닌 단 한번의 서비스에 대해 과

금하는 경우를 말한다. 아마도 이러한 서비스는 한달에 2~3회 정도일 것이며 그때마다 특정 CP가 아닌 다른 CP로부터 서비스를 제공받을 것이다. 이 경우 앞서 제안한 방법과는 달리 일정 시간 단위로 매번 User가 CP에게 사용에 대한 증거를 생성할 필요가 없다. 따라서 User는 접속 후 CP에게 서비스를 요청하고 User가 서비스를 제대로 받았을 경우에 한해서 사용에 대한 증거를 한번만 생성하여 보내면 된다.

[단계 1] Initialization

앞서 3.3.1에서 설명한 일정시간 단위의 과금 과정과 동일하므로 참조하기 바란다.

[단계 2] Setup & Regular Operation

CP로부터 <isAccepted> 메시지를 전달받은 User는 먼저 자신의 UID와 Time, 그리고 Service를 연접하여 서명한 다음 UID와 서명 결과를 연접하여 CP에게 보낸다. CP는 수신한 메시지에서 User의 UID를 확인하고 User의 공개키를 이용하여 서명을 확인한다. 이 과정에서 UID가 [단계 1]에서 받은 것과 동일한지 그리고 Time과 Service가 올바른지를 확인한다. 그 결과 만일 User가 보낸 메시지에 이상이 없다면 CP는 User가 원하는 컨텐츠인 Data를 User에게 전달한다.

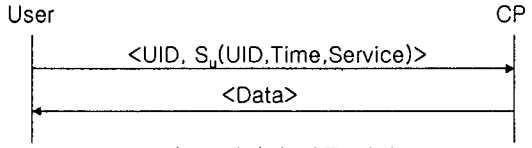

그림 5. 셋업과 정규 연산
Fig. 5 Setup & Regular Operation

[단계 3] Proof Generation

CP로부터 Data를 전달받은 User는 먼저 자신이 받은 데이터가 올바르게 수신되었는지를 확인하고 이상이 없다면 서비스 이용에 대한 증거로 자신의 UID와 P값을 연접하여 CP에게 보내고 사후 분쟁에 대비하여 이를 보관한다. 이때 P는 이전 단계에서 생성한 서명 결과에 U_{sk}를 연접하여 암호학적인 해쉬 연산을 수행한 결과이다. 그러나 만약 중간에 연결이 끊어졌거나 제대로 수신하지 못했을 경우에는 CP에게 Nak를 보내 Data를 재요청한다. 메시지를 전달받은 CP는 UID를 확인하고 증거와 서명 정보를 사후 분쟁에 대비하여 보관한다. CP는 User로부터 서명과 증거 P를 받는다. 서명은 User가 CP에게 서비스를 요청했다는 사실을 의미하며 P는 서비스 제공에 대한 증거로 일종의 영수증을 의미한다. 이때 CP는 User의 비밀키를 알 수 없기 때문에 임의로 서명을 생성할 수 없다. 그리고 암호학적인 해쉬의 일방향성 때문에 User의 비밀키를 모르고는 해쉬 결과를 생성할 수 없다. 따라서 User는 CP가 제시하는 요금을 믿을 수 있다. CP 또한 User의 서명을 확인할 수 있으므로 User가 제시한 증거를 신뢰할 수 있다.

요금에 대한 분쟁이 생겼을 경우를 가정해 보

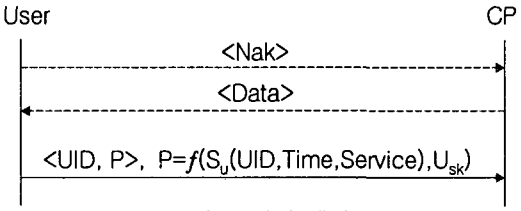

그림 9. 증거 생성
Fig. 6 Proof Generation

자. 이 방법은 첫 번째 제안한 방법과 User가 생성하는 메시지는 유사하지만 전달되는 메시지에는 차이가 있다. 일단 User는 자신이 제공받을 Service와 Time에 대해 서명을 하고 서명한 데이터를 먼저 CP에게 전달하였다. 이 메시지는 CP가 User의 실제 서비스 사용여부를 확인하기 위해서 필요하다. 본 방법에서는 데이터 수신 중간에 일어날 수 있는 장애에 대해서 User가 보상을 받을 수 있도록 하기 위해 CP에게 미리 서비스에 대한 증거로 영수증을 주진 않는다. 즉 User의 편익을 위해 컨텐츠 Data를 수신한 후에 이에 대해서 User가 영수증에 해당하는 증거를 생성하여 전달하는 것이 바람직하다. 왜냐하면 이전에 일정 시간 단위로 계속해서 부과되는 과금 방식과는 달리 본 방법은 그 사용이 1회 또는 몇 회로 한정되어 있기 때문이다. 따라서 Data를 수신한 후에 UID와 P를 증거로 생성하여 전달하도록 했다. 만일

분쟁 발생시 CP는 User의 서명과 P를 증거로 제시하며 User 또한 자신이 보관하고 있던 P를 이용하여 해결할 수 있다. 즉 CP는 User의 비밀키를 모르고서 P를 생성할 수 없으며 User 또한 CP가 자신의 서명과 P를 보관하고 있기 때문에 그 사실을 부인할 수 없다.

3.3.3 정액제로 과금하는 경우

정액제 서비스의 경우는 크게 두 가지로 생각해 볼 수 있다. 첫째, User가 사전 등록단계에서 CP와 계약하여 CP로부터 정액권을 구입한 다음 서비스 사용량에 따라 소액의 일정 금액씩 차감하는 방법이 있다. 두 번째로 User가 사전 등록단계에서 CP와 계약하여 CP로부터 예를 들어, 월 단위의 정액권을 구입한 다음 사용량이나 시간에 상관없이 계약한 그 달 동안에 자유롭게 서비스를 이용하는 방법이 있다.

예를 들어, 첫 번째 경우는 만원짜리 정액권을 구입하여 서비스 사용량에 따라 요금이 100원 단위로 부과되면서 차감되는 방법을 생각해 볼 수 있다. 이 경우는 전형적인 암호학적인 해쉬 체인의 방법을 이용하여 해결할 수 있으며 앞서 2장의 관련연구들에서 살펴본 논문들에서 제안이 된 바 있다.

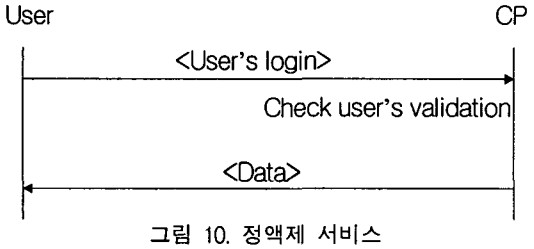

그림 10. 정액제 서비스
Fig. 7 Fixed Charge Service

두 번째 경우는 현재 우리 주변에서 케이블 인터넷 서비스를 받을 경우가 그러한데 한 달에 일정 금액을 지불하면 밤낮 가리지 않고 언제든 사용이 가능하다. 유료컨텐츠에 적용된 예는 일정 금액을 지불하면 하루 동안 그 사이트에 등록된 만화를 감상할 수 있도록 하는 서비스가 있다. 이러한 경우 전자결제 시스템을 이용하며 결제를 하면 사용자에게 아이디와 패스워드를 제공한다. 이 아이디와 패스워드로 접속하여 하루 동안 서비스를 제공받을 수 있다(그림 7 참조).

그렇다면 무선 환경에서 이러한 유료 서비스를 어떻게 안전하게 이용할 것인가? 이 방법은 앞서 제안한 방법들과는 달리 서비스 이용에 대한 증거가 필요 없다. 왜냐하면 이미 금액이 지불된 상태이므로 User와 CP가 사전에 키설정 이라든가 인증과정을 거치고 나면 그에 합당한 서비스를 CP가 제공해주면 그뿐이기 때문이다. 따라서 사후 분쟁의 소지가 없다. 그러나 여기서 가장 중요한 점은 상호간에 정확한 인증과 유효 시간이다. 서비스를 제공해주는 CP 입장에서는 이미 돈을 지불한 User가 실제 정확한 User인지 아니면 제 3자가 User인체 하는지를 구별해야한다. 이러한 인증문제는 아주 중요하다. 그러나 본 방법에서는 앞서 3.2절에서 언급한 대로 이미 기존의 인증 방법을 그대로 사용하는 것으로 가정했기 때문에 이 부분에 대한 기술은 생략한다. 즉 상호간에 확실한 인증을 거친 것으로 가정한다. 그 다음으로 유효 시간에 대한 문제인데, 만일 User가 CP와 사전 계약 당시에 합의한 일정기간동안에 접속하여 서비스를 제공받으려하는 것인지 아니면 이미 유효 기간이 지난 시간에 접속하여 서비스를 제공받으려 하는지를 면밀히 체크해야 한다. 예를 들어,

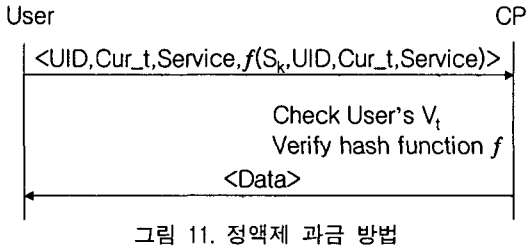

그림 11. 정액제 과금 방법
Fig. 8 Fixed charge billing scheme

User가 1월1일부터 1월 31일까지가 유효기간일 때 1월 15일 현재 서비스 요구를 했다면 CP는 이에 응할 것이다. 그러나 만일 2월 1일에 서비스 요구를 했다면 이에 응하지 않아야 한다.

이러한 유효 시간에 대한 체크는 다음과 같이 이루어질 수 있다. 먼저 User는 CP에게 접근하여

서비스를 받으려 하는 상황이며 이미 사전에 키교환이라든가 인증 과정을 거친 것으로 가정하자. 이때 상호간에 키교환을 통해 얻어진 세션키를 $S_{k,}$, 사전 계약 당시에 이용하기로 한 유효 기간을 V_t, User가 CP에게 서비스를 제공받기 위해 접근할 당시의 시간을 Cur_t이라 하자. 아래 그림 8에서 보는 바와 같이 User가 UID, Cur_t, Service와 이들에 대한 해쉬값을 CP에게 보내면 CP는 먼저 UID와 Service 내용을 확인하고 User가 보낸 메시지의 무결성을 확인하기 위해 해쉬값을 체크한다. 또한 Cur_t를 이용하여 이 값이 V_t이내의 범위인지를 체크한다. 만일 체크 결과 이상이 없다면 User가 요구하는 컨텐츠 Data를 전달한다.

그러나 여기서 다음과 같은 문제점이 발생할 수 있다. 예를 들어, 오늘밤 12시를 V_t의 끝이라 할 때 User가 밤 11시 59분에 접속하여 V_t를 넘겨 사용하려할 경우를 생각해 보자. 만일 이를 허용한다면 아마도 User는 심지어 몇 일이고 접속을 끊지 않고 사용하려할 것이다. 따라서 CP는 컨텐츠 Data를 제공한 다음에 일정 시간 간격이나 혹은 지속적으로 User의 V_t를 체크해야한다. 만일 서비스 이용 도중 V_t가 얼마 남지 않았을 경우 몇 분전에 User에게 이 사실을 메시지로 알리는 것도 한 방법일 것이다.

3.3.4 스트림 서비스에 과금하는 경우

최근에는 핸드폰을 통해서 음악이나 동영상을 실시간으로 제공받을 수 있다. 즉 무선 단말기를 통해 실시간으로 여러 가지 정보를 받아 볼 수 있는데 예를 들어, 사용자가 한 편의 영화를 실시간으로 감상하고자 할 때 서비스를 제공하는 쪽에서는 한 편의 영화를 모두 수신한 뒤에 이를 확인하고 요금을 부과할 수도 있고 또 일정 시간 단위로 요금을 부과할 수도 있다. 오랜 시간이 걸리는 영화를 끝까지 보지 않고 중간에 접속이 끊어지거나 혹은 User의 요구로 접속이 끊어질 수 있다. 이런 경우에는 한 편의 영화를 보고 난 후에 과금을 하는 것은 User의 입장에서 바람직하지 않다. 대개 이러한 스트림 서비스를 제공하는 경우에는 일정 시간 단위로 서비스를 제공하고 요금을 부과하는 방법을 사용하는 것이 바람직하다. 따라서 이 경우는 위 3.3.1에서 제안한 방법으로 해결이 가능하다.

IV. 제안한 과금 방법에 대한 분석

먼저 제안한 방법의 안전성은 암호학적인 해쉬 함수의 일방향성에 기반한다. 즉 사용자가 생성한 증거를 가지고 컨텐츠 제공자가 역으로 사용자의 비밀키 U_{sk}를 알아내기가 힘들다. 그리고 컨텐츠 제공자가 U_{sk}를 알 수 없기 때문에 사용자의 서비스 이용에 대한 증거를 생성할 수 없다. 그리고 부인방지에 대한 안전성 문제는 각 방법별로 앞서 기술한 바 있어 여기서는 생략한다. 그밖에 여러 암호학적인 해쉬 함수들 중에서 어떤 해쉬 함수를 이용하느냐는 그 선택에 따라서 안전성에 차이가 있다. 제안한 방법에서는 계산 속도 및 기타 조건에 따라 MD5나 SHA 같은 해쉬 함수를 가지고 수행할 수 있다.

아울러 제안한 첫 번째 방법인 일정 시간 단위로 과금하는 경우를 기준으로 그 계산량을 살펴보면 단계 1에서는 사용자와 컨텐츠 제공자 사이에 실제적인 연산은 수행하지 않는다. 사용자는 자신의 UID, Time, 그리고 Service 메시지를 컨텐츠 제공자에게 전달하고 컨텐츠 제공자는 사용자의 요구를 받아들일지를 결정한다. 단계 2에서 사용자는 서명 연산(예를 들어, 공개키 서명 방식을 가정하자)으로 1회의 공개키 연산을 수행하고 컨텐츠 제공자에게 전달한다. 또 컨텐츠 제공자는 받은 서명을 확인하는 과정에서 1회의 공개키 연산을 수행한다. 단계 3에서는 일정 시간 단위로 k번의 증거를 생성한다고 할 때, 사용자측에서 k회의 해쉬연산을 수행한다. 끝으로 단계 4에서는 단계 3 과정에서 생성된 증거가운데 컨텐츠 제공자인 경우 H^0와 H^t 그리고 사용자의 경우 Time과 H^t값을 단지 증거로 보관하는 과정을 수행한다. 따라서, 사용자 측에서는 1회의 서명연산과 k회 해쉬연산을 컨텐츠 제공자 측에서는 1회의 서명검증 연산을 수행한다. 제안한 방법은 사용자측 단말기의 연산량을 최소로 요구하는 무선 환경의 경우를 생각해볼 때 계산적인 면에서 충분히 실생활에 응

용될 수 있으리라 생각된다.

V. 결 론

최근 들어 모바일 기기의 발달로 컨텐츠 제공자가 게임, 벨소리, 이미지, 음악, 동영상 등 각종 매체에 따른 여러 가지 유형의 서비스를 제공할 수 있게 됨에 따라 그 과금에 대한 방법도 달라져야할 필요성이 대두되었다. 따라서 본 논문에서는 이러한 서비스 유형을 크게 4가지로 분류하여 각각에 대해 상호 신뢰할 수 있는 과금 방법을 제안하였다.

지금까지 개발되어 사용되고 있는 과금 시스템은 컨텐츠 제공자 대개가 로그 기록에 의해서 요금이 청구된다. 이 경우 사용자는 악의를 가진 컨텐츠 제공자에 의해서 실제 이용한 요금보다 더 많은 요금을 지불하도록 요구받을 수 있다. 사용자가 요금에 대한 이의를 가진다 하더라도 마땅히 사용에 대한 증거를 가질 수 없었다. 이러한 문제점을 해결하기 위해 사용자와 컨텐츠 제공자 사이에 상호 신뢰할 수 있는 증거를 생성하여 서비스를 제공하도록 하는 방법을 제안하였다. 즉 사용자는 컨텐츠 제공자가 제공한 요금을 보고 이의를 제기할 수 있고 또 이에 대한 증거를 요구하여 자신의 모바일 기기에 저장된 정보와의 비교를 통해 확인 할 수도 있다.

제안한 방법에서는 암호학적인 해쉬 함수를 이용하여 효율성과 안전성을 가지는 과금 방법을 제안하였다. 서명보다 빠른 연산으로 사용자의 계산량을 줄여주고 또 콘텐츠 제공자에게 빠른 응답이 가능하도록 하였다. 그러나 해쉬 함수를 이용할 경우, 작은 길이의 결과값을 가지지만 만약 자주 콘텐츠를 이용한다면 각각의 결과값은 무선 단말기에는 부담이 될 수 있다. 무선 단말기의 메모리 한계로 인해 장기적인 기간동안 자주 사용함으로서 많은 저장 공간이 필요하다. 그리고 콘텐츠 제공자의 사용 증거 제시 요구가 많아지면 사용자의 무선 단말기에서 계산량의 증가로 배터리 소모량에도 영향을 미친다. 따라서 향후 연구 방향으로 증거 생산과 저장을 믿을 수 있는 기관에게 위임하는 것이다. 믿을 수 있는 에이전트는 콘텐츠 제공자와 접속하여 사용 증거의 저장, 그리고 일정 시간 단위로 요구하는 사용 증거의 생성을 담당하는 것이다. 이렇게 한다면 위에서 언급한 메모리의 한계 그리고 배터리 소모량을 줄이는데 효과적일 수 있다.

참고 문헌

[1] G. Horn and B. Preneel, "Authentication and Payment in Future Mobile Systems," ESORICS '98, LNCS vol. 1485, pp. 277-293, 1998.

[2] K. M. Martin, B. Preneel, C. J. Mitchell, H. J. Hitz, G. Horn, A. Poliakova, and P. Howard, "Secure Billing for Mobile Information Services in UMTS," IS&N, LNCS vol. 1430, pp. 535-548, 1998.

[3] J. Zhou and K. Y. Lam, "Undeniable Billing in Mobile Communication," Mobicom '98, pp. 284-290, 1998.

[4] B. Patel and J. Crowcroft, "Ticket Based Service Access for the Mobile User," Mobicom '97, pp. 223-233, 1997.

[5] T. P. Pederson, "Electronic Payments of Small Accounts," Security Protocols, LNCS vol. 1361, pp. 59-68, 1997.

[6] L. Buttyán and J. Hubaux, "Accountable and Anonymous Access to Services in Mobile Communication Systems," IEEE Symposium on Reliable Distributed Systems, pp. 384-389, 1999.

[7] C. Boyd and A. Mathuria, "Key Establishment Protocols for Secure Mobile Communications: A Selective Survey," Information Security and Privacy(ACISP98), LNCS vol. 1438, pp. 344-355, 1998.

[8] M. Tatebayashi, N. Matsuzaki, and D. B. Newman, "Key Distribution Protocol for Digital Mobile Communication Systems," Advances in Cryptology: Proceedings of Crypto'89, LNCS vol. 435, pp. 324-334, 1989.

[9] ETSI, GSM Recommendations: GSM 01.02 -12.21 , Feb. 1993.

저자 소개

김순석(Soon-Seok Kim)

1997년 2월: 진주산업대학교 전자계산학과 공학사
1999년 2월 : 중앙대학교 컴퓨터공학과 공학석사
2003년 2월 : 중앙대학교 컴퓨터공학과 공학박사
2003년 3월~현재 : 한라대학교 정보통신공학부 전임강사
※ 관심분야 : 정보보호, 암호프로토콜, 이동통신