# 안전한 인스턴트 메신저의 설계와 구현

정 보 고[†] · 이 광 수[††]


## 요 약

컴퓨터와 네트워크의 보급이 일반화되면서 인터넷을 통한 정보 전달이 일상 생활처럼 되고 있다. 기존에는 정보를 전달하기 위한 방법이 주로 전자메일에 한정되어 있던 것에 반해, 요즘은 좀 더 즉각적으로 메시지를 전달해주는 인스턴트 메신저를 많이 사용하고 있다. 인스턴트 메신저는 이러한 장점으로 인해 국내에서도 사용자가 급속하게 늘고 있다. 현재 사용되고 있는 대부분의 인스턴트 메신저는 전송되는 정보가 아무런 보호장치 없이 네트워크를 통해 전송되어 제 3자에 의한 도청이 가능하게 된다. 따라서 전송되는 정보의 암호화를 포함하는 안전한 인스턴트 메신저 서비스의 필요성이 대두되고 있다. 본 논문에서 제안된 안전한 인스턴트 메신저는 사용자 개인이 메신저 서버에게 전송하는 개인 정보를 암호화하고, 사용자간에 전송되는 정보를 선택적으로 암호화하여 잠재적인 보안문제를 해결하였다. 그리고 시스템 설계는 일반 사용자도 쉽게 사용할 수 있도록 사용자 편의성에 중점을 두었다.


# A Design and Implementation of Secure Instant Messenger

Bo-go Jung[†] · Gwang-soo Rhee[††]


## ABSTRACT

As computers and networks become popular, distributing information on the Internet is common in our daily life. In the past, e-mail has been the primary choice of exchanging information, but instant messengers are gaining popularity abroad and domestically because of their nature of getting immediate responses. Most of existing instant messengers don't have any protection on the transmitted information, endangering the privacy of the users. As a counter measure, instant messengers need to provide security service including message encryption. In this paper, we designed a secure instant messenger with encryption capability. Our secure instant messenger solves the security problem by encrypting private information, which individual users send to the instant messenger server, and messages, which are exchanged among users. Another important design goal is the user transparency.


**키워드 : 인스턴트 메신저(Instant messenger), Diffie-Hellman(Diffie-Hellman), RSA, 암호화(Encryption), 보안(Security)**

## 1. 서 론

인스턴트 메신저란 네트워크를 통하여 실시간으로 메시지를 주고받을 수 있는 프로그램을 말한다. 대부분의 인스턴트 메신저 프로그램은 백그라운드로 실행되어 최소의 메모리와 네트워크 리소스를 사용하는 장점이 있으며, 메시지를 전송하는 기능 외에 부가적으로 파일 전송·일대일 대화·대화방·사용자 검색 기능을 제공한다. 또한 사용자의 메신저 서버 접속 상태를 알려주는 접속상태 통보도 함께 제공한다.

인스턴트 메신저는 전자메일이 사용자가 메일 서버에 접속하여 메일들을 읽어오는 과정을 요하는데 비해, 자동적으로 사용자 화면에 메시지를 전달함으로써 보다 간편하고 즉각적인 메시지의 전달을 기대할 수 있다. 인스턴트 메신저는 이러한 장점으로 인해 그 이용자 수가 빠르게 증가하고 있으며, 인터넷의 가장 보편적인 서비스의 하나로 정착될 것이라고 전망된다.

그러나 현재 사용되고 있는 대부분의 인스턴트 메신저는 전송되는 정보에 대한 보안 기능이 없는 상태로 운영되고 있다. 이렇게 인스턴트 메신저 서비스 사용자의 개인정보나 전송되는 메시지가 아무런 보호장치 없이 네트워크 상에 노출되어 있는 상황은 제 3자에 의해 그 정보가 도청될 수 있는 잠재적 보안 위험을 내포하고 있다. 따라서 전송되는 정보를 안전하게 전달하는 안전한 인스턴트 메신저 시스템 개발이 필요하다.

제안된 안전한 인스턴트 메신저는 보안을 요하는 정보를 암호화하여 전송하였고, 클라이언트-서버 구조에 비해 통신

※ 본 연구는 KISTEP 숙명여자대학교 연구기반 확충사업의 지원에 의해 수행되었음.(과제번호: 00-B-WB-02-A-02)
† 준 회 원 : SK Teletech 연구원
†† 종신회원 : 숙명여자대학교 정보과학부 교수
논문접수 : 2000년 12월 4일, 심사완료 : 2001년 3월 15일

오버헤드가 경감되는 자바 RMI에 의한 분산 처리 환경을 기반으로 설계되었으며, 사용자 정보 관리는 Oracle 7.3.3과 JDBC 인터페이스를 이용하였다. 그리고 사용 계층이 특정 컴퓨터 전문가 집단이 아니라 일반 사용자라는 측면을 고려하여, 보안 서비스의 사용자 투명성을 우선 고려하였다. 전송되는 내용을 암호화하기 위하여 DES 암호화 알고리즘을 사용하였고, DES 암호키의 교환은 Diffie-Hellman 키 교환 방식과 RSA 암호화 알고리즘을 사용하였다.

본 논문에서는 2장에서 현재 서비스되고 있는 인스턴트 메신저들의 일반적인 기능과 몇몇 제품들의 특징을 조사하고, 3장에서 안전한 인스턴트 메신저의 보안 요구사항에 대하여 알아보고 안전한 인스턴트 메신저 모형을 설계하였다. 4장에서는 구현된 안전한 인스턴트 메신저의 구현환경과 구현된 내용을 기능별로 나누어 설명하고, 5장에서 결론을 기술하였다.

## 2. 관련연구

현재 국내에 소개되어 있는 인스턴트 메신저의 종류는 매우 많다. 또한 계속해서 많은 곳에서 인스턴트 메신저 서비스를 제공하고 있어, 앞으로 그 종류와 이를 이용하는 사용자 수는 더욱 늘어날 것으로 보인다[5]. 인스턴트 메신저는 보편적으로 메시지 전송, 파일 전송, 일대일 대화, 대화방, 그리고 사용자 검색 기능을 갖추고 있다. 현재 나와있는 국내외 인스턴트 메신저 제품은 이런 일반적인 기능들을 갖추고 있으며, 부가적인 서비스의 종류와 오프라인 상태에 있는 상대와의 통신지원 여부 등에 조금씩 차이를 보인다. 이런 차이를 살펴보면서 현재 많이 사용되고 있는 ICQ[9], MSN 메신저[8], 그리고 UIN 메신저[6]에 대하여 알아본다. 또한 인스턴트 메신저 서비스 중에서 암호화 기능을 제공해주고 있는 MaXIM[7]에 대하여 알아보도록 한다.

### 2.1 ICQ

ICQ[9]는 미라빌리스사의 제품으로 인스턴트 메신저 서비스의 원조이며, 그 사용자 수도 많다. ICQ의 특징으로는 사용자가 대화 모드를 선택하여, 현재 자신의 상태를 다양하게 표시할 수 있으며, 대화 모드에는 온라인, 오프라인, 방해금지, 비공개 등으로 모드에 따라서 메시지 수신 방법 등의 차이가 있다. 또한 ICQ는 상대방이 접속 중이 아니더라도 메시지나 파일 전송, 채팅 요구 등이 가능한데, 이는 통신 요청 내용을 보관해 두었다가 상대방이 접속할 때 처리해주는 방법을 사용하기 때문이다. ICQ는 메일 전송 기능이 있는데, 이때 메일 클라이언트는 윈도우에 등록된 메일 클라이언트를 사용하거나, ICQ가 자체적으로 제공하는 메일 클라이언트를 통하여 메일을 읽고 보낼 수 있다. 또한 ICQ는 음성메시지 기능이 있어서 상대방에게 음성 메시지를 녹음하여 전송할 수 있다. 음성 메시지를 받은 상대방은 녹음된 메시지를 듣고, 파일로 저장하여 보관할 수 있다.

### 2.2 MSN 메신저

MSN 메신저[8]는 마이크로소프트사의 제품으로 메일 서비스 계정으로 메신저에 접속한다. MSN 메신저는 접속한 사용자에게 실시간으로 메시지 전송을 할 수 있는데, 수신자가 도착된 메시지를 읽으면 MSN 인스턴트 메시지 창이 뜨게 된다. 인스턴트 메시지 창은 일대일 대화 기능처럼 두 사용자가 주고받는 메시지를 한 화면에 보여주고, 두 사용자간에 일대일 대화를 하는 동안 다른 사람을 초대하여 대화방 기능처럼 사용할 수 있다. 메시지 전송 중에는 음성채팅 기능을 이용하여 문자 메시지를 전달함과 동시에 음성 메시지를 전달할 수 있다. 음성채팅 기능을 사용하면 수신자에게 음성채팅을 허가할 것인지 묻게 되고, 수신자로부터 요청이 승인되면 문자 메시지 전송과 함께 음성 메시지 전송이 가능하다. 또한 인스턴트 메시지 창에서 파일 보내기를 선택하면 현재 대화중인 상대방에게 파일을 전송할 수 있다. 수신자에게는 파일 저장 여부를 묻게 되고, 저장을 선택하면 파일저장을 위해 사전에 등록된 폴더에 저장된다.

### 2.3 UIN 메신저

UIN 메신저[6]는 유아이엔(주)사의 제품으로 웹 메일 서비스를 제공하는 다음 커뮤니케이션의 한메일과 연결되어, 한메일 계정을 가진 사용자라면 새로운 계정을 받지 않고도 UIN 메신저를 사용할 수 있다. 또한 새롭게 UIN 메신저에 가입하면 한메일에 자동으로 가입되어 전자메일 주소를 얻을 수 있다. 따라서 UIN 메신저는 한메일에 전자메일이 도착했을 때 메신저에서 메일 수신여부를 알려주며, 한번의 클릭으로 브라우저를 띄우지 않고 간편하게 메일을 확인할 수 있고, 메일 전송의 경우도 브라우저 없이 메일을 전송할 수 있다. 또한 메일을 읽어올 pop3 서버와 확인 간격을 설정하면, 메일이 도착하였는지 일정 간격마다 확인하여 메일 도착여부를 알려준다. pop3로 도착한 메일은 윈도우에 등록된 메일 클라이언트 프로그램을 이용하여 읽을 수 있다.

### 2.4 MaXIM

MaXIM Server/Client[7]는 (주)드림시큐리티사의 제품으로 인증 서버인 Magic CA 서버와 연계하여 암호화 기능과 인증 기능을 제공해주는 인스턴트 메신저 서비스이다.

MaXIM은 전자상거래 지불 프로토콜인 SET 프로토콜을 응용하여 개발되어, 인터넷을 통해 전송되는 통신 내용을 보호한다. 또한 대화방을 사용할 때, 메모리 관리를 대화방 개설자의 컴퓨터에 위임하여 서버의 로드를 감소하게 하였고, 신원 확인 및 자료의 암호화가 당사자간의 전자서명과

전자봉투에 의해서 이루어짐으로써 통신 내용의 보안이 가능하게 하였다. MaXIM은 사용자 아이디와 비밀번호가 유출되어도 인증서가 없으면 통신할 수가 없으며, 인증서를 분실하게 되면 다시 발급 받아야 한다. 사용자가 접속 중이 아닌 상태에서 전송되는 메시지와 파일은 서버가 안전하게 관리하며 수신자만이 암호화된 메시지 또는 파일을 읽을 수 있다.

MaXIM은 SET 프로토콜의 표준알고리즘인 RSA, DES, SHA1을 채택하였고, SEED 등의 국산 알고리즘도 탑재 가능하다. SHA1은 데이터를 입력받아 그에 해당하는 인덱스를 계산하는 해쉬 알고리즘이고, SEED는 128비트의 키를 사용하는 국내에서 개발된 대칭키 알고리즘이다.

MaXIM은 메시지 및 파일 전송의 암호화는 공개키 기반의 암호화를 선택하였고, 대화방에서는 전자서명과 전자봉투를 이용하여 가입자 신원확인 및 비밀키를 교환한 뒤, 전송되는 대화방 내용의 암호화는 비밀키 암호화 방식이 이용된다. 이때 대화방 개설자는 임의의 암호 키를 생성하고, 대화방 참여자에게 그 키를 배포한다.

## 3. 안전한 인스턴트 메신저

본 장에서는 일반적인 인스턴트 메신저가 갖고 있는 보안문제를 해결하기 위한 안전한 인스턴트 메신저를 제안하고 설계한다. 3.1절에서는 안전한 인스턴트 메신저가 갖추어야 할 보안 요구사항에 대하여 알아보고, 3.2절에서는 실제로 구현된 안전한 인스턴트 메신저 모형을 제시하고 설계한다.

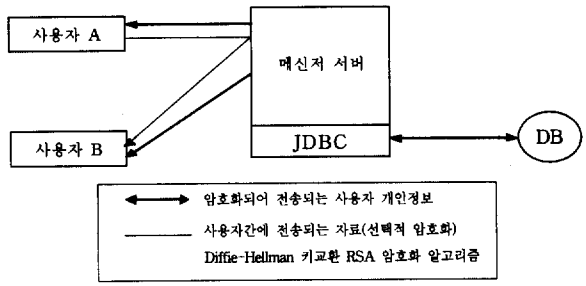
### 3.1 안전한 인스턴트 메신저의 보안 요구사항

인스턴트 메신저의 표준화 작업을 추진하고 있는 IETF 응용 영역 IMPP (Instant Messaging and Presence Protocol) 그룹에서 만든 RFC 2779 문서[4]에서는 인스턴트 메신저 보안과 관련하여 메시지의 기밀성과 재전송 공격에 대한 대비, 메시지 무결성 등을 기본 요건으로 규정하고 있다. 본 논문에서는 네트워크를 통해 전달되는 정보 가운데 보안을 요하는 모든 정보를 암호화하여 메시지의 기밀성을 보장하고, 매번 암호화에 사용되는 키를 통신 당사자들끼리 새로 정하게 함으로써 재전송 공격과 메시지 무결성 문제를 해결한다.

### 3.2 안전한 인스턴트 메신저 모형

본 논문에서는 전송되는 정보의 암호화를 포함하는 안전한 인스턴트 메신저를 제안한다.

본 논문에서 구현한 안전한 인스턴트 메신저 모형을 제시하면 위의 (그림 1)과 같다. 사용자가 서버에게 전송하는 개인정보는 항상 암호화하여 안전하게 전달되고, 사용자간에 전송되는 정보는 선택적으로 암호화를 하도록 하였다. 또한 3-tier구조를 이용하여 사용자는 인스턴트 메신저 서버의 데이터베이스에 직접 접근하지 못하고 메신저 서버를 통해서만 접근할 수 있도록 하였다. 안전한 인스턴트 메신저의 설계는 일반 사용자라도 불편함이 없도록 사용자 투명성을 우선적으로 고려하여 설계하였다.




(그림 1) 안전한 인스턴트 메신저 모형

사용자가 서버에게 전송하는 개인정보를 암호화하는 경우, 암호화에 사용되는 키를 교환하기 위하여 Diffie-Hellman 키 교환 방식을 사용한다. Diffie-Hellman 키 교환 방식을 사용하면 사용자에게 아무런 작업을 요청하지 않기 때문에 일반 사용자도 쉽게 사용할 수 있다. 사용자간에 전송되는 정보는 사용자의 선택에 따라 암호화가 이루어지도록 하였고, 암호화에 사용되는 키를 교환하기 위하여 Diffie-Hellman 키 교환 방식과 RSA 암호화 알고리즘 가운데 선택하도록 하였다. Diffie-Hellman 키 교환 방식을 사용하면 사용자의 투명성이 보장되고, RSA 암호화 알고리즘을 사용하면 수신자와 상호작용 없이 비밀키를 얻을 수 있으며 키 교환을 위해 필요한 시간을 줄일 수 있다. RSA 암호화 알고리즘을 사용하기 위해서는 개인키와 공개키 쌍을 필요로 하게 되는데, 이는 처음 안전한 인스턴트 메신저에 가입할 때 생성되어 키 저장소에 보관된다. 키 저장소는 사용자의 컴퓨터에 생성되기 때문에 다른 컴퓨터에서 동일한 사용자로 인스턴트 메신저를 사용하는 경우는 키 저장소를 이동해야 하는 번거로움이 있다.

본 논문에서 제안하는 안전한 인스턴트 메신저의 기능을 암호화 측면에서 나누어 설명하면 다음과 같다.

#### 3.2.1 개인정보의 암호화

사용자가 메신저 서버에 등록하거나 접속할 때 입력하는 개인 정보는 서버와 사용자가 공유한 키를 이용하여 항상 암호화하여 전달된다. 키를 공유하기 위해서는 Diffie-Hellman 키 교환 방식을 사용하고, 사용자와 서버가 공유하고 있는 비밀키는 제 3자가 알 수 없으므로 서버와 사용자 외에는 개인정보를 해독할 수 없다. 서버와 사용자가 개인 정보를 전송하기 위해 사용하는 키는 일회용이며, 매번 사용자와 서버가 협력하여 새로운 키를 생성하여 사용한다.

#### 3.2.2 접속 중인 사용자간에 전송되는 정보의 암호화

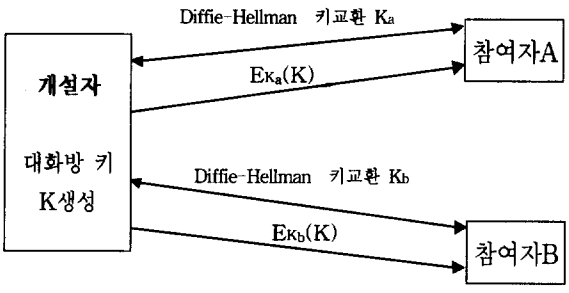
사용자간에 전송되는 정보는 항상 암호화되는 것이 아니라 사용자의 선택에 따라 암호화가 이루어진다. 암호화를 선택하

면, 두 사용자는 정보를 전송하기 전에 Diffie-Hellman 키 교환 방식을 사용하여 키를 공유하고, 그 공유된 키를 이용하여 정보를 암호화하여 전송한다. 이 때 사용되는 키는 통신 당사자만이 알 수 있고, 제 3자와 메신저 서버는 알 수 없다.

메시지 전송에서 암호화를 선택하면, 암호화를 위해 공유된 키는 일회용으로 제한되어 한번밖에 사용되지 않는다. 따라서 메시지 전송을 두 번 하게 되면, 키 교환 역시 두 번 이루어지게 된다. 그러나 일대일 대화에서 암호화를 선택한 경우는 교환된 키의 수명을 일대일 대화가 종료되기 전까지 계속되게 함으로써 매번 일대일 대화의 내용이 전달될 때마다 키를 공유해야 하는 번거로움을 줄이도록 하였다. 암호화된 정보를 수신하는 경우, 암호화 메시지는 알림 메시지를 사용하여 수신자에게 암호화된 메시지의 도착을 알리고, 암호화 일대일 대화는 암호화 전송이 이루어지고 있음을 타이틀바와 화면 하단의 아이콘을 이용하여 표시하였다.

3.2.3 대화방에서의 암호화

대화방에서 암호화를 선택한 경우는 암호화된 내용을 전송하기 위해서 두 명 이상의 사용자가 키를 공유해야 한다. 따라서 이 경우는 메시지 전송과 일대일 대화에서 사용하던 방법과는 조금 다른 차이가 있다. 대화방의 경우 Diffie-Hellman 키 교환 방식을 N명의 사용자를 대상으로 적용하기 위해서는 기존과는 다른 방법을 적용하는데, 그 방법에 대하여 설명하면 (그림 2)와 같다.



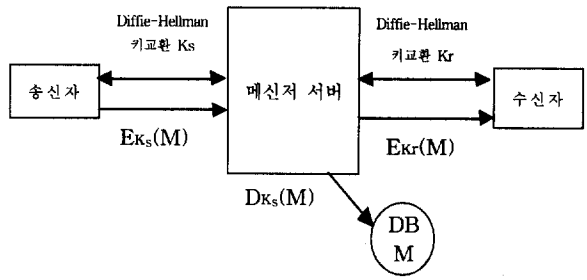

(그림 2) 암호화 대화방 키 교환

위의 (그림 2)에서 보는 것처럼 암호화 대화방을 개설하는 개설자는 임의의 대화방 암호 키 K를 생성한다. 암호화된 대화방에 참여하고자 하는 사용자 A가 들어오면, 개설자는 사용자 A의 정보를 확인하여 대화방 참여 승인 여부를 결정해 줄 수 있다. 대화방 개설자로부터 확인을 받고 대화방 참여가 승인되면, 개설자와 참여자 A는 Diffie-Hellman 방식을 사용하여 임시 키 $K_a$를 교환하고, 개설자는 그 임시 키 $K_a$를 사용하여 대화방 키 K를 암호화하여 A에게 전달한다. 이렇게 암호화 대화방에서는 Diffie-Hellman 키 교환 방식을 응용하여 개설자가 생성한 대화방 키를 각 참여자에게 안전한 방법으로 전달하였다.

3.2.4 접속 중이 아닌 사용자에게 Diffie-Hellman 키 교환 방식을 이용하여 암호화 메시지 전송

Diffie-Hellman 키 교환 방식은 접속 중인 두 명의 사용자간에 키를 교환하기 위해 사용된다. 따라서 수신자가 접속 중이 아닌 경우에 암호화 메시지를 전송하는 경우는 통신 당사자간에 Diffie-Hellman 키 교환 방식을 사용할 수 없다. 이런 경우는 두 가지 선택사항 중에서 사용자가 원하는 방법을 선택하여 암호화하도록 한다. 첫 번째는 기존의 방법처럼 Diffie-Hellman 키 교환 방식을 사용하는 방법이고, 두 번째는 RSA 암호화 알고리즘을 사용하는 방법이다.

접속 중이 아닌 사용자에게 암호화 메시지를 전송하기 위해 Diffie-Hellman 키 교환 방식을 이용하는 경우는 두 명의 사용자가 비밀키를 교환하는 것이 아니라 송신자와 서버가 협력하여 비밀키 Ks를 생성하게 된다. 송신자는 서버와 교환한 비밀키를 이용하여 암호화된 메시지 $E_{Ks}(M)$을 서버에게 전송하고, 서버는 이 내용을 해독하여 데이터베이스에 안전하게 저장한다. 나중에 메시지의 수신자가 접속하여 메시지보기를 선택하면 서버는 수신자와 협력하여 비밀키 Kr을 생성하고, 메시지 M을 비밀키 Kr로 암호화하여 보내준다. 이 과정을 그림으로 살펴보면 (그림 3)과 같다.



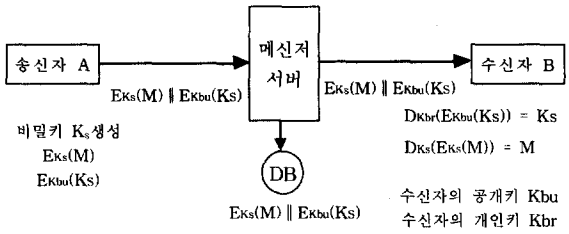

(그림 3) 접속 중이 아닌 사용자에게 암호화 메시지 전송 (방식 1)

이렇게 기존의 방법과 마찬가지로 Diffie-Hellman 키 교환 방식을 사용하면, 일반 사용자도 안전한 인스턴트 메신저를 쉽게 사용할 수 있다. 그러나 이 경우는 전송되는 메시지의 내용을 메신저 서버가 해독하여 보관하기 때문에 서버의 신뢰성에 영향을 받는다. 따라서 서버의 신뢰여부에 따라서 Diffie-Hellman 키 교환 방식 사용여부를 결정할 수 있다.

3.2.5 접속 중이 아닌 사용자에게 RSA 암호화 알고리즘을 이용하여 암호화 메시지 전송

접속 중이 아닌 사용자에게 RSA를 사용하여 암호화 메시지를 전송하는 경우는 아래의 (그림 4)와 같다.

송신자는 임의의 비밀키 Ks를 생성하고, 이를 이용하여 메시지를 암호화한다. 이 때 사용된 비밀키는 수신자의 공개키 $K_{bu}$로 암호화하고, 메시지를 암호화한 결과와 비밀키를 암호화한 결과를 함께 서버에 전송한다.

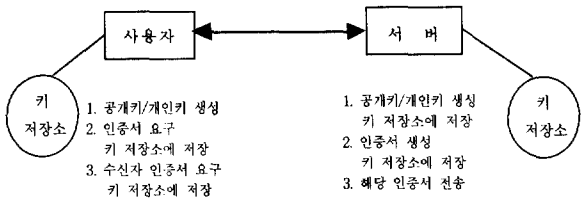



(그림 4) 접속 중이 아닌 사용자에게 암호화 메시지 전송 (방식 2)

수신자의 공개키를 얻기 위해서 송신자는 자신의 키 저장소에 수신자의 인증서가 있는지 확인하고, 없는 경우는 인증 서버(CA Server)에게 수신자의 인증서를 요구하여 받을 수 있다. 메시지를 받은 서버는 비밀키를 해독하기 위한 수신자의 개인키와 송신자가 임의로 생성한 비밀키 중 어느 것도 알지 못하므로, 받은 메시지를 해독하지 못하고 암호화된 상태로 데이터베이스에 저장한다. 메시지의 수신자가 접속하여 메시지 보관함을 통해 메시지를 읽게 되면 메시지는 암호화된 형태 그대로 수신자에게 전송되고, 이를 받은 수신자는 그 내용을 자신의 개인키 $K_{br}$로 복호화하여 메시지를 암호화하는데 사용된 비밀키 Ks를 얻고, 이를 이용하여 암호화된 메시지를 해독하여 그 내용을 볼 수 있다.

3.2.6 인증서 관리

접속 중이 아닌 사용자에게 암호화 메시지를 전송하기 위하여 RSA를 사용하면, 암호화와 복호화를 위해 공개키와 개인키 쌍이 필요하다. 이 키 쌍은 사용자가 메신저 서버에 등록할 때 생성되어, 각 사용자 컴퓨터의 키 저장소에 저장된다. 사용자는 키를 생성하면서 서버에게 공개키에 대한 인증서를 요구하는데, 인증서란 그 공개키가 확실하고 유효함을 나타내주는 것이다. 서버는 각 사용자로부터 인증서 요구가 들어오면 사용자의 공개키를 인증해주는 인증서를 사용자에게 돌려주고, 또한 서버의 키 저장소에 사용자의 인증서를 저장해둔다. 수신자의 인증서를 받은 사용자는 키 저장소에 인증서를 저장하여 다음에 같은 수신자에게 RSA를 이용하여 암호화할 때 서버에게 인증서를 다시 요구해야 하는 번거로움을 줄였다. 인증서 관리에 대한 내용을 그림으로 살펴보면 (그림 5)와 같다.




(그림 5) 인증서 관리

(그림 5)에서 보듯이 우선 사용자와 서버는 공개키와 개인키를 생성하고, 서버는 생성한 개인키와 인증서 형태의 공개키를 키 저장소에 저장한다. 2에서 사용자는 서버에게 공개키에 대한 인증서를 요구하는데, 서버는 각 사용자로부터 인증서 요구가 들어오면 사용자의 공개키를 인증해주는 인증서를 생성하여 사용자에게 돌려주고, 또한 서버의 키 저장소에 사용자의 인증서를 저장해둔다. 인증서를 받은 사용자는 1에서 생성한 개인키와 인증서를 키 저장소에 보관한다. 3의 경우는 메시지의 암호화에 RSA를 사용하는 경우로 수신자의 인증서가 필요한 경우, 송신자는 수신자의 인증서를 서버에 요구한다. 서버는 서버의 키 저장소에서 수신자의 인증서를 찾아 돌려주고, 이를 받은 송신자는 송신자의 키 저장소에 수신자의 인증서를 저장한다. 이렇게 수신자의 인증서를 저장함으로서 같은 수신자에게 RSA를 이용하여 여러 번 암호화할 때 서버에게 인증서를 다시 요구해야 하는 번거로움을 줄였다.

## 4. 구 현

본 장에서는 제 3장에서 살펴본 안전한 인스턴트 메신저에 대한 구현을 기술한다. 4.1절에서는 구현에 필요한 환경에 대하여 기술하고, 4.2절에서는 실제 구현된 안전한 인스턴트 메신저 시스템에 대하여 살펴본다.

### 4.1 구현 환경

안전한 인스턴트 메신저는 서버와 클라이언트간의 통신을 위해 클라이언트-서버 구조에 비해 통신 오버헤드가 경감되는 자바 RMI에 의한 분산 처리 환경을 기반으로 설계되었으며, 구현언어로는 자바를 이용하였다. 암호화 모듈을 사용하기 위해서 Sun JCE 규격을 따르는 오스트리아 그라쯔 대학에서 제공하는 IAIK-JCE[10] 패키지를 사용하였다.

클라이언트는 PentiumIII PC의 윈도우즈9x 운영 체제에서 구현 및 시험되었으며, 메신저 서버는 Sun UltraSparc에서의 Solaris 2.6 운영체제에서 구현 및 시험되었다.

전송되는 정보를 암호화하게 위해서는 DES 암호화 알고리즘을 사용하였고, DES 키 교환은 Diffie-Hellman 키 교환 방식과 RSA를 사용하였다. 대부분의 경우 Diffie-Hellman 키 교환 방식을 사용하였고, 수신자가 접속 중이 아닌 경우 암호화 메시지를 전송하는 경우는 Diffie-Hellman 키 교환 방식과 RSA 암호화 알고리즘 중에 사용자가 비밀키 교환 방법을 선택하도록 하였다. RSA를 사용할 때 공개키를 인증해주는 인증 서버의 역할은 메신저 서버가 그 기능을 담당하도록 하였다.

### 4.2 구현내용

본 절에서는 구현된 안전한 인스턴트 메신저를 암호화 기능별로 나누어 살펴본다.

4.2.1 등록 및 접속

처음 사용자가 안전한 인스턴트 메신저를 실행하면 (그

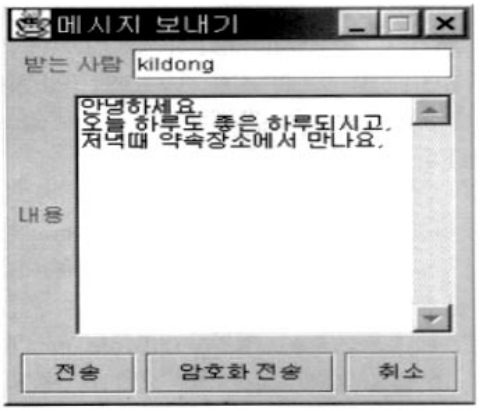
림 6)의 왼쪽과 같이 등록과 접속 버튼이 비활성화된 접속 화면이 나타난다.


(그림 6) 접속 화면

사용자가 안전한 인스턴트 메신저에 가입하거나 메신저에 접속하는 경우에 전송되는 개인 정보는 암호화되어 전송되어야 한다. 암호화에 사용되는 비밀키를 교환하기 위해 Diffie-Hellman 키 교환 방식이 사용되는데, 이때 필요한 키 교환 요소를 생성하는데 시간이 필요하다. 따라서 처음에는 등록과 접속 버튼을 비활성화 상태였다가 키 교환 요소의 생성이 완료되면 (그림 6)의 오른쪽 화면처럼 등록과 접속 버튼이 활성화된다.

4.2.2 접속 중인 사용자에게 암호화 메시지 전송

사용자간에 전송되는 정보는 사용자의 선택에 따라 암호화를 결정할 수 있다. 메시지 전송의 경우도 사용자의 선택에 따라 암호화가 이루어지는데, (그림 7)에서 보는 것처럼 사용자는 전송할 내용을 입력하고 '전송'과 '암호화 전송' 가운데 선택할 수 있다.

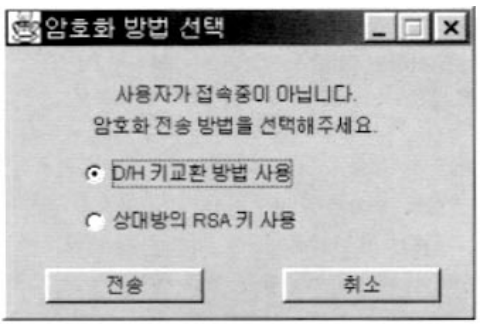



(그림 7) 메시지 전송

수신자가 접속 중이고 암호화 전송을 선택한 경우는, 두 통신 당사자간에 Diffie-Hellman 키 교환 방식을 이용하여 비밀키를 교환하고, 그 비밀키를 이용하여 메시지를 암호화하여 전송한다. 수신자에게는 암호화 메시지가 도착하였다는 알림 메시지가 뜨고, 메시지 내용을 해독하여 보여준다.

4.2.3 접속 중이 아닌 사용자에게 암호화 메시지 전송

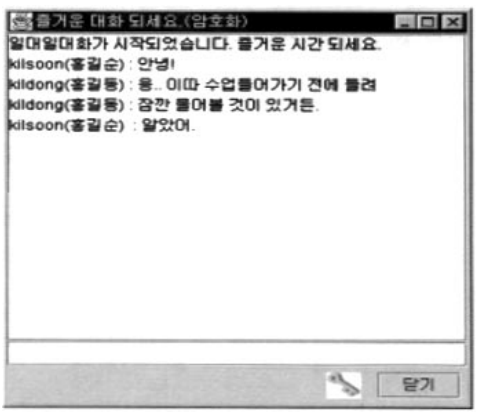
메시지 전송의 경우, 우선 받는 사람이 현재 인스턴트 메신저에 접속되어 있는지 확인을 한다. 만약 수신자가 접속 중이 아니라면 메시지는 실시간으로 전송되지 않고, 서버의 데이터베이스에 보관된다. 이때 송신자에게는 수신자가 접속 중이 아니라는 알림 메시지가 뜨고, 메시지를 전송할 것인지 묻게 된다. 접속 중이 아닌 사용자에게 암호화 메시지를 전송할 경우는 사용자가 (그림 8)과 같이 송신자가 비밀키를 얻는 과정을 선택할 수 있다.




(그림 8) 알고리즘 선택

4.2.4 암호화 일대일 대화

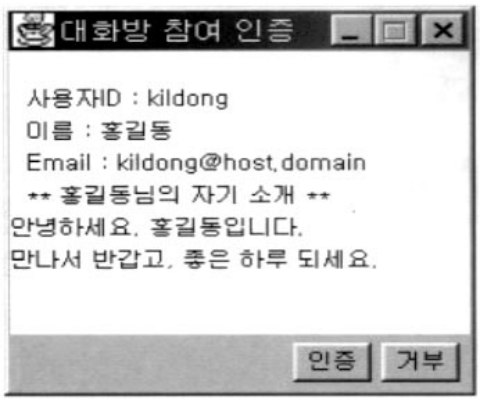
일대일 대화에서 암호화를 선택하면 대화 내용을 암호화하는데 필요한 비밀키를 Diffie-Hellman 키 교환 방식을 사용하여 교환하고, 암호화 일대일 대화가 시작된다. 암호화 일대일 대화는 (그림 9)와 같이 일대일 대화 창에 타이틀 바에서 "(암호화)"라는 설명을 추가하고 화면 하단에 열쇠 아이콘을 추가하여 암호화 일대일 대화가 이루어지고 있음을 나타내었다.




(그림 9) 암호화 일대일 대화

4.2.5 대화방

대화방에서 암호화를 선택한 경우, 대화방 내용을 암호화하는데 필요한 비밀키 K를 대화방 개설자가 임의로 생성한다. 대화방에 참여하고자 하는 사용자가 들어오면 우선 대화방 개설자에게 (그림 10)과 같이 대화방에 참여하고자 하는 사용자의 정보가 보여지게 된다. 개설자는 이를 바탕으로 사용자의 대화방 참여를 승인해줄 수 있다.

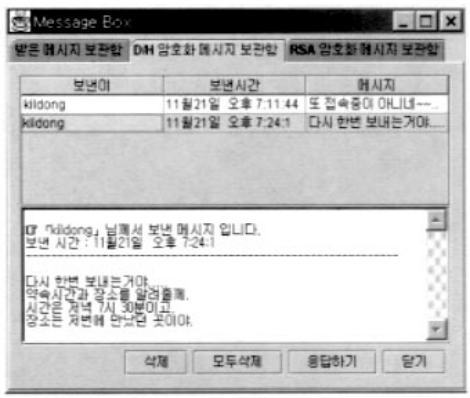



(그림 10) 대화방 인증

암호화 대화방의 경우 암호화 일대일 대화와 마찬가지로 처음 참여할 때 받은 비밀키를 대화방을 종료할 때까지 계속 사용하여 매번 키를 공유해야 하는 번거로움을 줄였고, 암호화 대화방에 아이콘을 추가하여 암호화 대화가 이루어지고 있음을 나타내었다.

4.2.6 메시지 보관함

메시지 보관함은 사용자가 접속 중이 아닌 경우에 전송된 메시지를 보관한다. 메시지 보관함은 (그림 11)과 같이 받은 메시지를 세 분류로 나누어 보여준다.




(그림 11) 메시지 보관함

메시지 보관함은 평문으로 전송된 메시지와 Diffie-Hellman 키 교환 방식을 이용하여 전송된 메시지와 RSA를 사용하여 전송된 메시지로 나누어서 보관된다. 메시지 보관함을 선택하면 서버에 보관된 사용자의 메시지를 읽어오게 되는데, 평문으로 전송된 메시지는 평문으로 가져오게 된다. Diffie- Hellman 키 교환 방식을 이용하여 전송된 메시지는 서버와 사용자간에 Diffie-Hellman 키 교환을 이용해 비밀키를 공유한 뒤, 그 비밀키로 암호화되어 전송된다. RSA를 이용하여 전송되었던 메시지는 암호화된 상태로 서버에 보관되어 있고, 사용자는 메시지 보관함을 이용하여 암호화된 메시지를 읽어온다. 읽어온 메시지는 사용자의 개인키로 해독되어 보여진다.

## 5. 결 론

본 논문에서는 인스턴트 메신저를 사용하여 네트워크를 통해 정보를 전송할 경우에 생기는 문제점을 지적하고, 이를 해결하는 방법으로 암호화 기능이 첨가된 안전한 인스턴트 메신저를 제안하였다. 안전한 인스턴트 메신저에서는 Diffie-Hellman 키 교환 방식을 사용하여 매번 사용되는 암호 키를 통신 참가자의 협력에 의해 새롭게 정하므로, 메시지를 가로채 보관해 두었다가 재전송하는 공격이 불가능하다. RSA를 사용하여 메시지를 전송하는 경우도 메시지 전달 이전에 Diffie-Hellman 방식의 암호화 로그인 과정을 거치므로 재전송 공격이 가능하지 않다. 또한 암호 키에 대한 보안으로 메시지의 변조나 위조가 불가능한 시스템으로 구현하였다.

안전한 인스턴트 메신저는 서버로 전송되는 개인정보를 암호화하여 개인정보가 제 3자에 의하여 도청되는 것을 불가능하게 하였다. 또한 사용자간에 전송되는 자료 가운데 보안을 요하는 자료에 한해 사용자가 선택적으로 암호화를 할 수 있어, 보안을 요하는 자료들은 수신자에게 안전하게 전송되도록 하였다.

본 논문에서 구현한 안전한 인스턴트 메신저는 수신자가 접속 중이 아닌 경우 Diffie-Hellman 키 교환 방식을 적용하기 어려운 경우를 고려하였다. 따라서 수신자가 접속 중이 아닌 경우의 메시지 전송은 송신자가 수신자가 아닌 메신저 서버와 Diffie-Hellman 키 교환 방식을 적용하던가 RSA 암호화 알고리즘을 사용하도록 하였다.

다수 사용자나 사용자 그룹을 대상으로 메시지나 파일을 전송할 때 효율적인 키 분배 및 관리방법은 그룹의 변동 등으로 인해 복잡한 문제를 내포하고 있으며, 이의 해결은 향후 연구과제로 남긴다.

## 참 고 문 헌

[1] "Data Encryption Standard(DES)," National Bureau of Standards FIPS Publication 46, 1977.
[2] W.Diffie and M.E.Hellman. "Multiuser cryptographic techniques," AFIPS Conference Proceedings, 45 : 109-112, 1976.
[3] R. L. Rivest, A. Shamir, and L. Adleman, "A method for obtaining digital signatures and public key cryptosystems," Communications of the ACM, 21 (1978), 120-126.
[4] M. Day, S. Aggarwal, G. Mohr, and J. Vincent, "Instant Messaging /Presence Protocol Requirements," RFC 2779, Feb 2000.
[5] 킬러 애플리케이션 "인스턴트 메시징", PC Week 4(19) : 66-67, 1999.
[6] UIN 메신저, http : //uin.com/.
[7] MaXIM, http : //www.dreamsecurity.com/products/products_frame.htm/.
[8] MSN Messenger Service, http : //messenger.msn.com
[9] ICQ, http : //www.icq.com/.
[10] IAIK-JCE, http : //jcewww.iaik.tu-graz.ac.at/.

## 정 보 고

e-mail : bogojung@hanmail.net

1999년 숙명여자대학교 전산학과 졸업 (학사)

2001년 숙명여자대학교 대학원 컴퓨터과학과 (이학석사)

2001년~현재 SK Teletech 연구원

관심분야 : 네트워크 보안

## 이 광 수

e-mail : rhee@sookmyung.ac.kr

1981년 서울대학교 계산통계학과 졸업 (학사)

1986년 Washington University 대학원 컴퓨터과학과(이학석사)

1990년 Washington University 대학원 컴퓨터과학과(이학박사)

1990년~현재 숙명여대 정보과학부 교수

관심분야 : 알고리즘, 네트워크 보안