

인터넷 텔레포니(VoIP) 서비스의 설계 및 구현

정회원 이종화*, 강신각**

Design and Implementation of Internet Telephony Services

Jong-Hwa Yi*, Shin-Gak Kang** *Regular Members*

요 약

인터넷 텔레포니(VoIP) 관련 기술 개발이 빠르게 진행되면서 인터넷 전화 서비스를 비롯한 다양한 부가 서비스 개발이 한창 진행중이다. 현재 활발히 개발되고 있는 VoIP 기술로서 ITU-T H.323과 IETF SIP 프로토콜을 들 수 있는데, 이들은 종단간의 호 설정 및 해지 기능을 처리하는 응용 계층의 호 시그널링 프로토콜에 해당된다. 현재까지는 H.323 기반으로 구현된 VoIP 제품과 서비스가 널리 사용되고 있으나, 인터넷 환경에서의 장점을 갖고 있는 SIP 를 이용한 다양한 VoIP 서비스가 빠르게 개발되고 있는 실정이다. 본 논문에서는 먼저, SIP 프로토콜을 이용하여 개발 가능한 차세대 응용 서비스 유형 및 특성을 분석하고, VoIP 기본 응용 중에 하나인 인터넷 전화 서비스의 설계 및 구현에 대해 기술하고자 한다. SIP 기반의 인터넷 전화 서비스는 사용자 인터페이스 및 음성 통화 기능을 포함한 응용 모듈 그리고 User Agent는 윈도우 98/2000상에서 비주얼 C/C++를 이용하여 구현되었으며, SIP Proxy 서버와 Registrar는 Linux 7.0상에서 구현되었다.

ABSTRACT

The fast advance in the VoIP technologies gives a rich opportunity to create different kind of VoIP applications such as IP telephony services. The application level call signaling protocols such as ITU-T H.323 and IETF SIP provide the communication functions of end-to-end call setup and release. Currently, there is a lot of H.323 based VoIP products in the market, however SIP is considered as a suitable protocol for supporting applications in IP environments, so SIP based VoIP products and services begin to appear. In this paper, firstly we present the characteristics of some possible SIP based applications and describe the design and implementation of a VoIP example service named PC-to-PC Internet telephony service using the developed SIP network components. The PC-to-PC Internet telephony service and User Agent are developed in MS window 98/2000 using visual C/C++, and Proxy server and Registrar in Linux 7.0 using C, respectively.

1. 서론

지난 10월 12일에 인터넷 전화 서비스의 선두 사업체인 다이얼패드 커뮤니케이션은 MS사의 차세대 운영체제인 윈도우 XP에 다이얼패드를 탑재, 유료 인터넷 전화 서비스를 제공한다고 발표를 했다. 많은 국내 통신 및 서비스 사업자들이 인터넷 텔레포니 (VoIP: Voice over IP) 서비스 시장의 확대

전망에 대해 기대감을 표시하면서도 아직까지 성공적인 비즈니스 모델이 창출되지 못하고 있어 진정한 VoIP의 성공을 의심하는 사람들도 있는 현 시점에서, 다이얼패드사의 이러한 발표는 향후 인터넷 전화 시장의 활성화에 크게 영향을 미치리라 생각된다. VoIP 분야에서 인터넷 전화 서비스는 가장 기본 서비스 중 하나라 할 수 있는데, 향후 사용자들은 윈도우 XP 환경에서 별도의 프로그램을 다운

* ETRI 표준연구센터 통신프로토콜표준연구팀 선임연구원(jhyice@etri.re.kr),

** ETRI 표준연구센터 통신프로토콜표준연구팀 팀장(sgkang@etri.re.kr)

논문번호 : 010332-1113, 접수일자 : 2001년 11월 13일

※ 본 연구는 정보통신부 국가정책사업 "인터넷텔레포니 기술표준화 연구"에 의해 수행되었습니다.

로드 받지 않고도 쉽고 편리한 방법으로 인터넷 전화 서비스를 제공받을 수 있게 될 것이다.

인터넷 텔레포니란 기존의 음성 전화 서비스를 인터넷상에서 제공하는 기술을 의미한다. VoIP에 대한 용어를 이렇게 간단히 정의할 수 있지만, 사실상 현재의 인터넷 기술을 이용하여 기존의 음성 전화서비스와 동일한 품질의 서비스를 제공하기에는 아직도 해결해야 할 기술들이 많은 실정이다. 사용자 입장에서는 저가 이용료, 고품질, 손쉬운 인터페이스, 인터넷 전화 서비스를 비롯한 다양한 서비스 등이 우선적으로 제공되어야 하며, 사업자 입장에서는 이를 해결할 수 있는 관련 기술 개발, 기존 전화 서비스와의 연동, 기존 인터넷 서비스와의 통합, 인터넷 자원 관리, 차세대 서비스 수용을 위한 통합 네트워크 구축 등 많은 개발 및 연구 이슈에 당면하고 있는 실정이다.

VoIP 기술은 ITU-T, IETF, IMTC, ETSI TIPHON을 중심으로 국제 표준화 작업이 활발히 진행되고 있으며, 그 결과로써 현재 VoIP 기술의 장벽이라 할 수 있는 서비스 품질(QoS), 신뢰 전송(Reliability), 이기종 망간의 인터워킹 등을 보장하기 위한 여러가지 해결방안이 제안되고 있다.

ITU-T와 IETF에서 개발된 H.323과 SIP(Session Initiation Protocol) 프로토콜은 종단간의 통화를 제공해 주기 위해 호 설정과 해지 기능을 처리하는 응용계층의 호 시그널링 프로토콜에 해당된다^{1,2,3}. 현재까지는 H.323 표준기술을 기반으로 하여 VoIP 제품이 개발되고 서비스가 제공되고 있는 실정이지만, H.323은 멀티미디어 화상회의 시스템을 위한 프로토콜로 개발되었기 때문에 오디오 데이터만을 처리하는 VoIP 응용을 개발하는 경우 상대적으로 프로토콜 기능이 복잡할 뿐만 아니라 인터넷에서의 다른 응용과의 접목이 용이하지 않다. 이와는 반대로 SIP 기술은 H.323에 비해 프로토콜이 단순하고 기존의 인터넷 환경에서의 용이성, 확장성, 재사용성의 장점을 갖고 있어 이를 이용한 제품과 서비스 개발이 빠르게 진행되고 있다.

현재 SIP는 여러 분야의 응용 서비스에서 호 시그널링 프로토콜로 사용되고 있거나 개발중에 있다. 인터넷 전화를 비롯하여 PSTN에서 제공되고 있는 부가 서비스, 멀티미디어 화상 회의, 3rd party 호 제어 서비스, 이동성 지원 서비스, 인스턴트 메시징 서비스, 위치 통보 서비스, 전자상거래 등 매우 폭넓은 분야에서 개발되고 있다.

SIP에 대한 표준화는 IETF SIP 워킹그룹을 중심

으로 이루어지고 있으며, 현재 RFC2543bis09 버전의 표준이 발표되어 있는 상태이다. 본 논문의 2장에서는 먼저 SIP 프로토콜 기능을 이용하여 개발될 수 있는 차세대 응용 서비스의 특성을 기술한다. 3장에서는 RFC2543bis03에 따라 설계 및 구현된 SIP 네트워크 구성요소 즉, User Agent, Proxy 서버, Registrar에 대해 기술하고, 마지막으로 구현된 SIP를 이용하여 개발된 인터넷 전화 서비스에 대해 기술한다.

II. SIP기반 VoIP 서비스 특성

VoIP의 대표적 서비스로 인터넷 전화 서비스를 꼽을 수 있다. 국내 서비스 사업자들이 제공하고 있는 인터넷 전화 서비스는 아직까지 이용 편리성이나 품질에 문제가 있지만 무료 혹은 저가로 제공됨에 따라 많은 사용자들에게 각광을 받고 있다. 그러나 무료서비스 제공에 점점 어려움을 겪고 있는 국내 사업자들은 좀 더 새로운 VoIP 서비스 창출을 위해 한편으로는 PSTN에서 제공되는 동일한 품질의 인터넷 전화 서비스를 개발하고 있으며, 다른 한편으로는 통합 메시징 서비스, 멀티미디어 화상 회의, 전자 상거래 등 기존 인터넷 응용과의 접목을 통해 통합 VoIP 서비스 모델을 개발하고 있는 것으로 보인다.

SIP 프로토콜을 이용하여 구현 가능한 인터넷 전화 서비스로는 삼자 통화 뿐만 아니라 호 전환, 통화 중 착신 전환, 통화 중 대기, 부재중 안내, 호 예약, 메시지 알림 기능 등 기존 전화 부가 서비스가 인터넷상에서 제공될 수 있다.

IETF에서 SIP 프로토콜에 대한 표준화가 진행되면서 SIP가 제공하는 단순성, 범용성, 확장성, 모듈성 등의 특성을 이용하여 많은 인터넷 응용이 개발중에 있다. 하지만 현재 RFC2543에서 정의되어 있는 SIP 기능은 인터넷 응용들이 필요로 하는 모든 요구사항을 만족시키지 못하고 있어, SIP 프로토콜에 대한 확장이 계속적으로 이루어지고 있는 상태이며, 이와 관련된 많은 드래프트 문서가 제안되고 있다.

본 장에서는 SIP의 기본 기능 혹은 확장 기능을 이용하여 개발 가능한 서비스들의 주요 특성을 기술하고자 한다⁴.

2.1. 인터넷 전화 서비스

PSTN상에서 제공되는 전화 서비스를 인터넷 상

으로 전이하여 제공할 수 있으며, SIP 프로토콜의 기본 기능을 이용하여 개발 가능하다. 이밖에 구현 가능한 서비스로는 호 전환, 무 응답 혹은 통화 중 착신 전환, 호 예약, 통화중 대기, 삼자 통화, 개별 혹은 그룹 당겨 받기 서비스 등이 있으며, 이러한 서비스들은 SIP 확장 기능이 요구된다.

2.2. 통합 메시징(Unified Messaging) 서비스

PSTN과 인터넷망을 기반으로 음성, 팩스, 전자 메일과 같은 메시지를 통합 관리하는 서비스이다. SIP 는 단순한 세션 제어 뿐만 아니라 여러 멀티미디어 타입도 지원한다. MIME 타입을 지원하고 메시지 바디 부분에 SDP(Session Description Protocol) 을 포함시킴으로써, 텍스트, 오디오, 비디오 등의 멀티미디어 데이터 전송을 지원한다. IP망 혹은 PSTN 망에서만 가능했던 다양한 멀티미디어 서비스들이 가능해지며, 이들 서비스로는 폰과 PC간의 일반 전화통화, 팩스, 화상전화 등이 있다.

2.3. 호 필터링(Call Screening) 서비스

송신자가 차단하고자 하는 수신측 위치정보를 등록하여 호가 설정되지 않도록 하는 서비스이다. 송신자가 호를 요청할 경우 SIP proxy 서버 혹은 호 필터링 서버가 수신자의 위치 정보를 데이터베이스에서 찾는다. 그리고 송수신자의 위치 정보를 이용하여 해당 호 요청을 포워딩 시킬 것인지를 결정한다. 유효하지 않은 서비스 요청에 대해서는 송신자에게 에러 응답을 보내고, 유효한 서비스 요청에 대해서는 해당 호 요청을 데이터베이스에서 찾은 위치로 포워딩 시킨다.

2.4. 위치 정보(Presence) 서비스

사용자들에게 사용자들의 최근 위치 정보를 동시적으로 알 수 있게 해 주는 서비스이다^{5,6}. 즉, 사용자들은 등록(subscription)을 통해 상대방의 정보를 얻고, 통보(notification) 기능을 통해 자신에게 등록된 사용자들에게 변화된 위치 정보를 알려준다. Notification은 위치 정보가 변화했을 때 곧 바로 보내지기 때문에 등록된 사용자는 항상 최신의 상대방 위치정보를 유지할 수 있다. 이 서비스를 구현하기 위해서는 기본적인 SIP 기능 이외에 SUBSCRIBE 및 NOTIFY 메시지와 관련 헤더들이 확장되어야 하며, REGISTER 메시지의 헤더 확장도 요구된다.

2.5. 인스턴트 메시징 서비스

온라인 상태에 있는 사용자들간에 서로 간단한

메시지를 주고 받을 수 있도록 해 주는 서비스이다⁷. 현재 많은 인스턴트 메시징 시스템은 위치 정보 서비스와 같이 구현되고 있다. 위치정보 서비스의 사용자 리스트를 통해 인스턴트 메시징 메시지를 보낼 수 있는 온라인상의 사용자들을 알 수 있다. SIP의 응용계층의 라우팅, 사용자 이동성 지원 기능, 인증, 보안 등에 대한 기능은 그대로 이용하고, 멀티파트 세션 설정은 그룹간 메시지 전송에 이용될 수 있다. 인스턴트 메시징 서비스를 구현하기 위해서는 새로운 MESSAGE 메시지와 이를 처리하는 기능만 확장하면 가능하다.

III. SIP기반 인터넷 전화 서비스

SIP에 대한 표준화가 IETF SIP 워킹그룹에서 진행되어 1999년 3월에 RFC2543이 제정되었으며, 최근에는 RFC2543bis09 버전이 발표되어 있는 상태이다³.

본 장에서는 RFC2543bis03 버전에 따라 설계 및 구현된 인터넷 전화 서비스 및 SIP 네트워크 구성요소인 User Agent, Proxy 서버 그리고 Registrar 기능에 대해 기술한다.

3.1. 설계

3.1.1. 전체 시스템 구조

인터넷 전화 서비스를 제공하기 위한 시스템은 크게 사용자와의 인터페이스를 담당하고 음성 통화 서비스를 제공하는 응용 서비스 모듈, 응용의 요청에 따라 SIP 기반의 기능을 처리하는 User Agent, SIP 메시지 처리를 담당하는 Proxy 서버 그리고 사용자 정보를 관리하는 Registrar로 구성된다⁸. <그림 1> 은 이러한 구성요소로 연결된 전체 시스템 구성도를 보여주고 있으며, 구성요소간 동작 시나리오를 보여주고 있다.

본 논문에서 사용하는 User Agent는 응용 서비스 모듈 (즉, 인터넷 전화 서비스 모듈)을 제외한 SIP 메시지를 요청 혹은 응답하는 클라이언트 혹은 서버 역할을 담당하는 모듈을 의미하며, 처리 기능에 대한 API를 제공한다. 이것은 User Agent가 독립적인 프로세스 모듈로 설계 및 구현되었음을 의미하며, 제공되는 API를 이용하여 다양한 일대일 유형의 VoIP 서비스를 개발할 수 있다.

3.1.2. 응용 서비스 모듈

다음의 세부 기능 모듈로 구성된다.

는 기능을 수행한다.

- **등록 기능** : 등록에 대한 5가지 세부 기능 (처음 등록, 수정, 검색, 삭제, 모든 주소 삭제)을 수행하며 결과를 전달하는 기능을 수행한다.

(3) 음성 통화

호가 설정된 후에는 미디어 세션을 통해 음성 정보를 교환하게 되며, 이 때 RTP(Real Time Transfer) 프로토콜을 이용한다. RTP 세션에서 사용되는 포트 번호 및 코덱 정보는 호 설정시 협상을 통해 결정된 값을 이용한다^[11]. 마이크를 통해 입력된 사용자의 음성 정보는 선택된 코덱에 따라 인코딩되어 RTP에 실려 전송되어지며, 상대 시스템은 이와 반대의 절차에 따라 디코딩되어 스피커를 통해 전달된다. GUI를 통해 사용자가 선택한 코덱의 우선 순위는 User Agent에 사용자의 우선순위 정보로서 전달되어 다음 호 설정시부터 반영되어 진다.

(4) 에러 메시지 처리 기능

SIP 프로토콜은 서버가 처리한 기능에 대한 응답을 6개의 그룹(1xx ~ 6xx)으로 구분하여 전달하는데, 응용 모듈은 수신한 각 응답 메시지에 대한 의미를 분석하여 적절한 동작을 처리해야 한다. 이 중 4xx, 5xx, 6xx 응답 메시지들은 에러 발생시 전달되는 응답 메시지에 해당된다. 인증에 관한 응답 메시지만 401이나 407 메시지가 도착하는 경우 응용 모듈은 사용자에게 인증에 필요한 정보를 요청하여 입력 받은 후 앞서 전달한 요청 메시지를 재 전송하는 기능이 처리되어야 한다.

(5) 상태 정보 처리 기능

인터넷 전화 서비스를 사용할 수 있는 지, 요청한 기능에 대해 진행 상황, 에러 정보 혹은 부가 정보 등이 사용자에게 전달되도록 하는 기능이다.

표 1. 응용 모듈과 User Agent 의 API

| | API | 기능 |
|----------------------|----------------------------------|----------------------------|
| 사용자 → 응용 모듈 | EstablishConnection(destination) | 상대방과의 통화 요청 |
| | ReleaseConnection() | 상대방과의 통화 종료 요청 |
| | AcceptConnection() | 전화 받기 허락 |
| | RejectConnection() | 전화 받기 거절 |
| | GetCapabilities(destination) | 상대방 능력정보 요청 |
| | Registration() | 등록 요청 |
| | SetBellSound() | 벨 소리 설정 |
| | SetCodec() | 코덱 선호도 선택 |
| | SetPreference() | 통화 선호도 선택 |
| | SetNetwork() | 서버 정보 설정 |
| 응용 모듈 → UA | InitializeUA(UAInfo t) | UA 구동 및 사용자 정보 전달 |
| | Update(UAInfo t) | 사용자 선호도 정보 전달 |
| | TerminateUA(UAInfo t) | UA 종료 요청 |
| | CallSetupReq(CallSetup t) | 호 설정 요청 |
| | CallSetupRsq(CallSetup t) | 호 설정에 대한 응답 전달 |
| | CallReleaseReq(CallSetup t) | 호 종료 (Bye) 요청 |
| | CallCancelReq(CallSetup t) | 호 해지 (Cancel) 요청 |
| | CapabilitiesReq(Capabilities t) | 능력 정보 (Options) 요청 |
| | RegisterReq(RegisterInfo t) | 등록 요청 |
| UA → 응용 모듈 | InformationInd(information_t) | 전화받은 응답 메시지 (예:180.301) 전달 |
| | CallSetupInd(CallSetup t) | 호 설정 요청이 왔음 통보 |
| | CallSetupCnf(CallSetup_t) | 호 설정에 대한 상대방 응답 전달 |
| | CallReleaseInd(CallSetup t) | 호 종료 요청이 왔음 통보 |
| | CallCancelInd(CallSetup t) | 호 해지 요청이 왔음 통보 |
| | CapabilitiesCnf(Capabilities t) | 요청한 능력 정보 전달 |
| | RegisterCnf(RegisterInfo t) | 요청한 등록 정보 전달 |

3.1.3. User Agent 모듈

RFC2543bis 표준에는 SIP 네트워크의 세부 기능에 대해 정의하고 있는데, 이 중 User Agent는 클라이언트 혹은 서버의 역할을 담당하는 UAC 와 UAS로 구분되며, 주요 기능으로는 사용자의 요청을 수행하기 위하여 UAC인 경우 SIP 메시지를 생성 및 전송, 그리고 UAS인 경우 SIP 메시지 수신 및 처리 기능이다. 본 설계에서는 일대일 유형의 호를 동시에 여러 개 처리할 수 있는 VoIP 서비스를 지원하는 User Agent 기능이 고려되었으며, 세부 기능별 API가 제공된다. User Agent의 API를 이용하여 본 논문에서 기술한 인터넷 전화 서비스의외에 다양한 부가 서비스를 개발할 수 있다. <표 1>은 응용 서비스 모듈과 User Agent가 제공하는 주요 API를 보여주고 있다.

User Agent은 다음의 세부 기능 모듈로 구성된다.

(1) API 기능 처리 모듈

UA에서 정의하고 있는 각 API 기능을 제공하기 위한 처리 모듈로서, 처리하는 주요 기능은 다음과 같다.

- **구조체 값에 대한 검증** : 각 API 호출시 응용이 전달한 구조체 값에 대한 문법과 의미를 검증하는 모듈이다. 반드시 필요한 값이 전달되었는지도 검증하여 결과를 응용에게 알려준다.
- **SIP 메시지 생성 및 파싱** : 내부적인 검증 절차

가 이루어진 후에는 RFC2543bis에 정의되어 있는 SIP 메시지를 생성 혹은 파싱을 거치게 된다^[3]. 예를 들면 호 설정, 해지 등 처리해야 하는 기능에 따라 SIP 메시지 (INVITE, BYE, ACK, CANCEL, OPTIONS, REGISTER)를 호출하기 위해 필요한 구조체 (SIPMsg_t) 에 값을 설정하고 각 메시지를 호출하게 된다. 각 SIP 메시지를 처리하기 위한 상태도가 정의되었으며, 이에 따라 UA는 내부 호 정보 관리, 타이머 관리, 응용 모듈 그리고 하위 모듈과의 인터페이스 제어를 통해 상대방과의 프로토콜 기능을 수행하게 된다.

(2) 전송 스트림 생성 및 수신 모듈

SIP 메시지가 생성되면 UAC인 경우에는 이 메시지를 스트림으로 변환 (Generator 모듈) 하여 상대방에게 전달하고, UAS인 경우에는 소켓을 통해 전달된 스트림을 다시 SIP 메시지가 관리하는 SIPMsg_t 구조체로 변환 (Parser 모듈) 하게 된다. 이 모듈은 SIP 스택이라는 명칭으로 불리우며, UA에서 뿐만 아니라 서버 모듈에서도 공통적으로 사용되는 모듈이다.

(3) SDP 모듈

호 설정시 사용자가 미디어 세션에서 사용할 정보를 협상할 수 있다. 본 설계에서는 RFC2327에 따라 SDP 모듈이 설계 및 구현되었으며^[12], 세부 기능별 API가 제공된다. 이외에 음성 정보만을 처리하는 VoIP 서비스를 위해 호 설정시 손쉽게 사용할 수 있는 인터페이스가 추가적으로 정의되었는데, UAC인 경우 SDP 정보를 설정하기 위해 SetSDPMessage(SDPInfo_t) 그리고 UAS인 경우 전달된 SDP 정보를 획득하기 위해 GetSDPMessage(SDPInfo_t)가 정의되었다.

(4) 인증 모듈

RFC2543bis03에서 정의되어 있는 두 가지 인증 방식 (HTTP Basic & Digest) 을 이용하여 사용자를 인증할 수 있다. Proxy 서버인 경우 사용자 인증 요구시 407 응답 메시지를 보내오면 혹은 Registrar가 401 메시지를 보내오는 경우, UA는 응용 모듈에게 전달하여 사용자로부터 추가 인증 정보를 전달 받는다. 이 정보는 Proxy-Authorization 헤더 혹은 Authorization 헤더로 인코딩하여 서버로 전달된다.

3.1.4. Proxy 서버

User Agent가 전송한 SIP 메시지는 Proxy가 내부적으로 처리하거나 (Proxy 서버 역할) 다른 Proxy로 전달 (Proxy 클라이언트 역할)하여 준다. 자신이 처리하고 있는 호에 대한 상태 정보를 관리하느냐에 따라 Stateful Proxy 혹은 Stateless Proxy로 구분되며, 본 설계 및 구현에서는 두 가지 형태의 Proxy기능이 모두 반영되었다^[13].

- **Stateless Proxy 기능** : 자신이 처리하는 호에 대한 상태나 관련 정보를 관리하지 않는다. 주요 기능으로는 수신된 SIP 메시지가 요청 메시지인 경우 Registrar가 관리하는 위치 서버의 데이터베이스 (Location Database) 로부터 해당되는 위치 정보를 획득한다. 이 때 여러 위치 정보가 존재하는 경우 q value가 가장 높은 하나를 취한다. 여기에 Proxy 자신의 주소를 Via 헤더에 기록하여 수신 메시지에 추가시켜 Request-URI에 있는 다음 위치로 전송한다. 수신된 SIP 메시지가 응답 메시지인 경우에는 수신 메시지서 최상위 Via 헤더 정보를 삭제시킨 후 Request-URI에 있는 다음 위치로 전송한다.
- **Stateful Proxy 기능** : 자신이 처리하는 호에 대한 상태나 관련 정보를 관리하기 때문에 호에 대한 정보를 Call Leg 관리 테이블을 통해 처리한다. Call Leg에 관리되는 정보로는 From, To, Call-ID, Contact, Cseq 헤더들의 값이다. SIP 메시지를 수신한 경우 앞서 언급한 헤더 정보들을 Call Leg에 저장하고, 위치 서버의 데이터베이스로부터 해당되는 모든 위치 주소를 획득한다. Proxy는 포킹(Forking) 기능을 이용하여 각 위치 주소에 대해 자신의 주소를 Via 헤더에 실어 SIP 메시지에 추가시킨 후 Request-URI 주소로 전달하며, 제일 먼저 응답이 온 위치 정보를 취하여 호를 설정하게 된다.

Proxy 서버는 다음의 기능 모듈로 구성된다.

- **스트림 처리 모듈** : 소켓을 통해 송/수신되는 스트림을 처리하는 모듈이며, 수송계층 프로토콜로서 UDP와 TCP가 지원된다. 또한 UDP에서 TCP로 혹은 TCP에서 UDP로 변환이 가능하도록 한다.
- **SIP 메시지 생성/파싱 모듈** : 소켓을 통해 수신된 스트림을 SIP 메시지 형태로 생성(Parser 모듈)하거나 이와 반대로 Proxy가 처리하는 SIP 메

시지를 스트림으로 변환(Generator 모듈) 시키는 모듈로 메시지에 대한 의미 및 구문을 검증하는 기능이 포함된다.

- **호 처리 모듈** : 수신 받은 SIP 메시지에 대해 Stateless Proxy 혹은 Stateful Proxy 기능을 처리하는 모듈이다. Proxy 기능 수행시 필요한 정보는 위에서 언급한 바와 같이 위치 서버의 데이터베이스를 통해 얻어 처리한다.
- **인증 기능** : Proxy는 수신하는 SIP 메시지에 대해 인증 기능을 제공하지 않는 경우(none) 혹은 HTTP Basic 방식 또는 Digest 인증 방식으로 동작할 수 있다. 이러한 선택은 Proxy 서버 인스턴스 시 설정할 수 있도록 한다.

3.1.5. Registrar

모든 사용자는 Registrar에 등록되어야 하며, 하나 이상의 위치정보를 등록할 수 있다. 따라서, Registrar의 주요 기능은 사용자들의 위치 정보를 저장하고 이를 검색, 수정, 삭제시키는 기능을 담당한다. User Agent는 사용자의 위치정보를 전달하기 위해 SIP Register 메시지를 생성하고 관련 정보 즉, 위치 정보, 선호도 (q value), 유효 시간 (expires time) 정보를 Contact: 헤더에 실어 RegisterReq()를 호출함으로써 서버로 전송된다^[13].

Registrar가 수행하는 세부 등록 기능은 다음과 같으며, RegisterReq() 함수를 통해 수행되며 전달되는 구조체 값을 분석하여 해당 기능을 수행한다.

- **등록** : 새로운 위치정보를 등록하기 위한 기능으로, From, To, Contact list 정보를 전달 받아 자신의 DB에 저장한다. 사용자가 위치 정보를 등록하는 방법으로는 로컬 outbound Proxy 서버에게 전달하여 Registrar로 재전달되는 방법과 Registrar 주소를 이용하여 직접 전달하는 방식이 있을 수 있다.
- **수정** : 이미 등록된 정보를 수정하는 기능으로, 수정 대상의 Contact와 수정 정보(q value 혹은 expires time)를 전달받아 데이터베이스를 수정한다.
- **검색** : 전달된 From, To 헤더에 대해 등록된 모든 위치 정보를 전달한다.
- **삭제** : 삭제할 Contact 헤더에 expires time을 0으로 설정하여 전달되어 지면 해당 위치 정보를 삭제한다.
- **모든 정보 삭제** : Contact 헤더가 * 값을 갖고

expires time이 0을 설정된 경우 등록된 모든 위치정보를 삭제시킨다.

- **사용자 인증** : Basic 및 Digest 방식의 인증을 제공한다. 인증 정보를 요구하기 위해 User Agent에게 401 응답정보에 인증 방식(challenge)을 포함하여 전달하게 된다. User Agent는 Authorization 헤더에 사용자 인증 정보를 전달함으로써 등록 기능이 수행된다.

등록기능은 사용자 자신이 수행할 수도 있고, 제삼자가 다른 사용자의 위치 정보를 등록하여 줄 수도 있다. 이런 경우에는 From 헤더 정보는 현재 등록을 대신 요청하는 제삼자의 SIP URL에 해당되며, To 헤더 정보는 등록해 주고자 하는 사용자의 SIP URL 그리고 Contact 헤더는 To 사용자의 위치 정보에 해당된다.

3.2. 구현

3.2.1. 구현 구조

<그림 3>은 개발된 인터넷 전화 응용 서비스 및 User Agent의 구현 API 구성도를 보여주고 있다. 응용 서비스는 window 98/2000버전에서 비주얼 C++ 6.0을, 그리고 User Agent 기능은 C/C++ 를 사용하여 구현되었다.

- **GUI Manager**: 사용자 인터페이스 처리 부분을 수행한다. 사용자가 선택한 각 서비스를 수행하기 위해 응용 서비스 API를 호출하고, 결과나 상태 정보를 전달해 준다.
- **Application Manager**: 인터넷 전화 서비스 API를 제공하며, 내부적으로는 정의된 상태도에 따라 호 설정, 해지, 사용자 인증 등 해당 기능을 수행하기 위해 SIP UA API를 호출하여 처리한다. 호 설정 이후에 음성 통화를 위해 세션 관리자 (Session Manager)를 동작하게 하며, 호 해지를 위한 메시지를 전달받으면 세션 관리자 기능을 종료한다.
- **SIP UA Process Module**: 정의된 UA 상태도에 따라 호 설정, 해지, 재전송, DNS lookup, 사용자 인증, 메시지 오류 검증 등의 기능을 수행한다. 기본적으로 일대일 형태의 여러 호를 동시에 관리 가능하다. 이 모듈에서 제공하는 API는 두 그룹으로 구성되는데 먼저, 일대일 VoIP 서비스

개발을 위한 API(high level functions)가 제공되며, 이것은 <표 1>에서 정의된 API에 해당된다. 또 다른 API (Low level functions)는 내부적으로 SIP 메시지를 생성, 검증, 파싱, 로컬 에러 처리 등을 수행하는 기능을 담당한다. 개발하고자 하는 VoIP 서비스 특성에 따라 이 두 API 중 선택하여 이용할 수 있다. SIP 메시지(SIPMsg_t)는 StartLine(메시지유형, 메시지명, RequestURI, 버전, 상태코드 등의 정보), HeaderLine(헤더명과 헤더값들의 리스트 형태) 그리고 Body_t(SDP 정보)로 구성된다.

- **SIP Stack Module:** 상위 모듈에서 생성된 SIP 메시지를 전송하기 위하여 스트림으로 변환시키는 모듈과 소켓을 통해 전달된 스트림을 정의된 SIP 메시지로 변환시키는 모듈로 구성되며, 생성 혹은 파싱 에러 처리 기능도 포함된다.
- **Session Manager:** 호가 설정된 이후 음성 통화를 할 수 있도록 총괄적인 제어 기능을 수행하는 모듈에 해당한다. 이를 위해 오디오 입출력 처리 모듈(Audio I/O Module), 코덱 처리 모듈(Codec

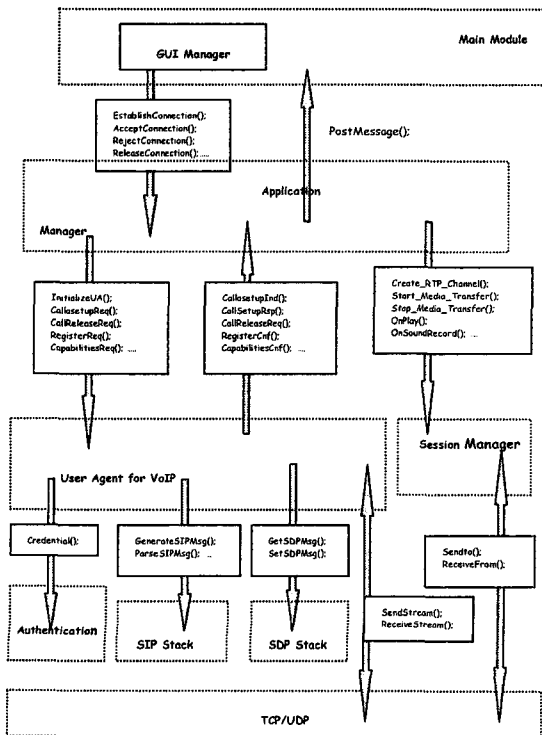


그림 3. 응용 및 UA 구현구조

Module) 그리고 오디오 전송 모듈(RTP Module)을 제어한다. 오디오 입출력 처리 모듈은 사용자의 음성 정보를 녹음 혹은 재생하기 위해 사용자가 사용하고 있는 오디오 디바이스를 제어하는 기능을 수행한다. 사용자의 음성 정보는 코덱 처리 모듈을 통해 인코딩 혹은 디코딩 된다. 앞에서 언급한 바와 같이 4개의 코덱을 이용하여 음성 통화를 할 수 있는데, 이 중 사용자가 입력한 우선순위에 따라 해당 코덱 모듈이 수행된다. 오디오 전송 모듈은 RFC1889에 정의되어 있는 RTP 프로토콜 기능이 구현되었다.

<그림 4>는 Proxy 및 Registrar의 구현 구조를 보여 주고 있다. 기본적으로 Out bound Proxy나 Registrar에 대한 IP 주소, UDP 혹은 TCP 사용 여부 그리고 포트 번호는 미리 지정되어 통신하게 된다. 본 구현에서는 GUI 화면을 통해 사용자가 통신하고자 하는 임의의 서버들의 정보를 설정하여 통신할 서버를 선택할 수도 있다.

Proxy와 Registrar는 Linux 7.0상에서 C로 구현되었으며, 자신들이 관리하는 정보를 Location 서버의 데이터베이스에 저장, 검색, 수정, 삭제 등의 기능을 통해 처리하며, 이 때 MySQL API를 사용한다.

- **Proxy Process Module:** 정의된 상태도에 따라 Stateless 혹은 Stateful Proxy 기능을 수행하는 모듈이다. 수송계층 프로토콜로서 UDP를 기본적으로 사용하나 본 구현에서는 멀티 쓰레드 방식을 이용하여 동시에 UDP와 TCP를 통해 통신이 가능하며, 또한 여러 호에 대한 메시지를 동시 송신, 수신, 제어가 가능하다.

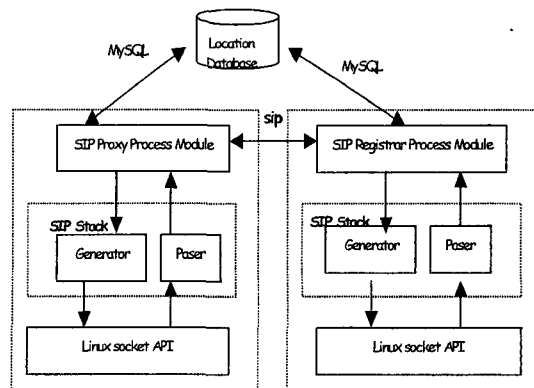


그림 4. Proxy & Registrar 구현 구조

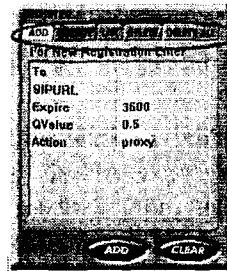
- **SIP Stack:** Proxy가 전송하고자 하는 SIP 메시지를 스트림으로 변환하거나 소켓을 통해 전달된 스트림을 SIP 메시지로 변환하여 Proxy에게 전달해주는 Generator 및 Parser 기능을 수행한다. 변환시 발생하는 에러 처리 기능도 포함되어 있다.
- **Registrar Process Module:** 정의된 상태도에 따라 SIP 메시지 중 REGISTER 메시지 기능을 수행하는 모듈이다. 요청된 Register 메시지에 대해 사용자 위치 정보 및 부가 정보를 검증하여 Location 서버의 데이터베이스에 추가, 수정, 삭제 기능을 수행하며 결과를 전달하여 준다.
- **SIP Stack:** Proxy 경우와 동일한 방법으로 Registrar가 사용하는 구조체를 이용하여 변화 기능을 수행한다.

3.2.2. 인터넷 전화 서비스의 GUI

<그림 5>은 사용자가 인터넷 전화 서비스를 실제로 사용할 수 있는 GUI 화면을 보여주고 있다.

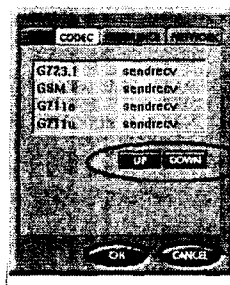
3.2.3. 상호운용성 시험

SIP forum이나 IMTC(International Multimedia Telecommunications Consortium) 그룹은 매년 3,4 차례에 걸쳐 상호운용성 시험 이벤트를 주최하는데 [13][14], 이러한 이벤트의 주요 목적은 개발되는 SIP



등록(Register) 메뉴
- 신규등록, 수정, 검색, 삭제, 모두 삭제 기능이 제공됨

(c) 등록 메뉴



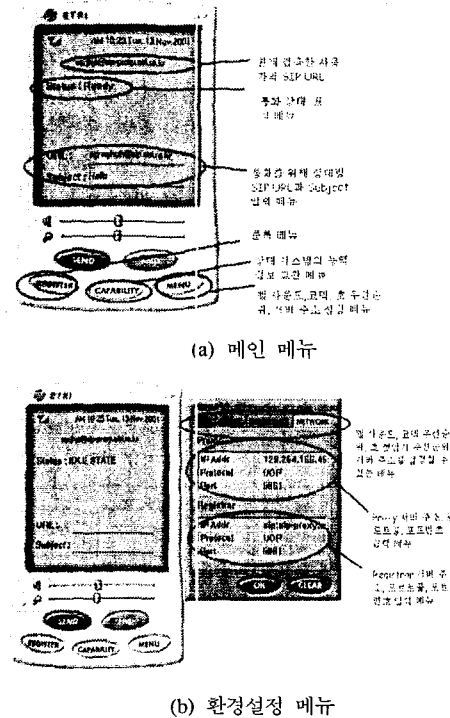
기스핑이 제공하는 4가지 코덱중에 우선순위를 결정할 수 있는 메뉴

(d) 코덱 우선순위 선택 메뉴

그림 5. 인터넷 전화 서비스 - GUI 메뉴

프로토콜 구성요소들의 기능 및 상호운용성 시험을 실시하여 제품간의 호환성을 보장하기 위함이다. 현재 국외 주요 SIP 프로토콜 개발 업체들은 이러한 이벤트에 참가하여 자신들의 제품을 검증 받고, 또한 구현자들은 개발시 혹은 동작시 발생하는 에러들을 볼 수 있는 유용한 기회로 이용하고 있다.

본 논문에서 개발된 SIP 프로토콜 및 이를 이용하여 구현된 인터넷 전화 서비스도 자체적으로 그리고 위에 언급한 이벤트에 참가하여 타 업체들의



(a) 메인 메뉴

(b) 환경설정 메뉴

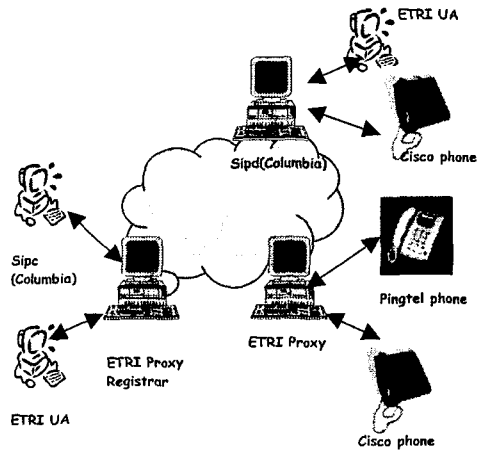


그림 6. SIP 시험 환경

제품과 상호운용성 시험을 여러 시나리오를 통해 실시하였다. 먼저, 자체 시험은 그림 6에서 나타나는 바와 같이 콜롬비아 대학에서 개발한 Sipc(UA 모듈)와 Sipc(서버 모듈)^[15], Pingtel사 SIP 폰^[16], Cisco사의 SIP 폰, vovida의 오픈 소스^[17] 등을 설치하여, 이들간의 호 설정 절차, 음성통화 그리고 호 해지 절차 등에 대한 시험을 완료하였다.

SIP Forum과 IMTC에서는 상호운용성 시험시 활용되는 시험 시나리오, 기능, 메시지, 에러 등을 포함하는 시험 스펙을 정의하여 발표하고 있는데, 참여하고자 하는 모든 구현 제품들은 이러한 시험 스펙을 이용하여 타 제품과의 시험을 수행하게 된다. 시험 스펙은 SIP 구성요소별(UA, Proxy, Registrar 등)로 세부 시험 항목을 정의하고 있는데^[18], 이 중 표2는 User Agent와 Proxy에 대한 시험 항목을 보여 주고 있다. 각 시험 항목에 대해서 구현한 User Agent와 Proxy 가 해당 SIP 메시지를 올바르게 생성하는지, 전달받은 메시지를 파싱하여 정확한 기능을 수행하는지, 중복 응답 메시지가 전달되는 경우의 처리 능력, 에러 발생시 제대로 대처하는지 등에 대한 세부 기능을 시험하였다.

본 SIP 프로토콜은 IMTC 의 시험 이벤트 (2001년 12월, winter Interoperability Test)와 SIP Forum 의 시험 이벤트(2002년 4월, SIPit)에 참가하여 User Agent와 Proxy 서버에 대해 위의 시험 항목에 대한 상호운용성 시험을 완료하였다^{[19][20]}.

표 2. 상호운용성 시험항목 리스트

| User Agent 시험 항목 |
|---|
| 1. UDP |
| 2. TCP |
| 3. basic authorization |
| 4. digest authorization |
| 5. basic proxy authorization |
| 6. digest proxy authorization |
| 7. receive CANCEL |
| 8. send CANCEL if outgoing call hung up before response comes |
| 9. DNS SRV records |
| 10. re-INVITE to change codecs |
| 11. re-INVITE to add media streams |
| 12. re-INVITE to change media address(mobility) |
| 13. recursion |
| 14. unicast REGISTER |
| 15. multicast REGISTER |
| 16. Record-Route/Route |
| 17. Require/Proxy-Require |
| 18. multiple codecs (send/receive) |
| 19. multiple media streams (send/receive) |
| 20. dynamic RTP port numbers (send/receive) |
| 21. can receive and process Expires in INVITE |
| 22. can send Expires in INVITE |
| 23. can send Via hiding headers |
| 24. can send bodies in INVITE for display |
| 25. can receive bodies in INVITE for display |
| 26. send multipart |
| 27. receive multipart |
| 28. multi-proxy authorization |
| 29. can receive and do something useful with non-SIP URL in 3xx (like pass to browser) |
| 30. can receive and do something useful with bodies in 3xx responses (like pass to browser) |
| 31. does something intelligent when multiple 200 OK received |

Proxy 시험 항목

| |
|--|
| 1. UDP to TCP conversion |
| 2. TCP to UDP conversion |
| 3. basic proxy authorization |
| 4. digest proxy authorization |
| 5. record routing |
| 6. forking - parallel |
| 7. forking - sequential |
| 8. receive unicast registers |
| 9. receive multicast registers |
| 10. supports non SIP URLs in Contact in REGISTER |
| 11. obeys action tag in REGISTER |
| 12. can always redirect |
| 13. can receive CPL in REGISTER |
| 14. obeys Expires in INVITE |
| 15. forking of non-INVITE methods |
| 16. SRV records |
| 17. tel URL handling (can send a request somewhere when one arrives with a tel URL in request URI) |
| 18. drops requests when Max-Forwards is zero |
| 19. can detect loops |
| 20. can act as local outbound proxy (i.e., forward to URL in request URI when domain is not its own) |
| 21. understands Proxy-Require |
| 22. can process Hide: Hop |
| 23. can process Hide: All |
| 24. recursion |
| 25. generates CANCEL on 200 OK when forking |
| 26. can receive and propagate CANCEL |
| 27. forwards all provisional responses upstream |
| 28. can forward multiple 200 OK upstream |

V. 결론

본 논문에서는 VoIP분야에서의 가장 기본 응용 서비스의 하나인 인터넷 전화 서비스에 대한 설계 및 구현에 대하여 기술하였다. 본 인터넷 전화 서비스는 하위 호 시그널링 프로토콜로서 IETF SIP 프로토콜을 사용하는데, 본 논문에서는 SIP 프로토콜의 주요 구성요소인 User Agent, Proxy 서버, Registrar에 대한 설계 및 구현에 대해 기술하였다. 설계 및 구현은 SIP의 표준문서인 RFC2543bis03 버전을 기반으로 이루어졌다.

SIP 프로토콜은 현재 사용자들이 쉽게 접하고 있는 다른 인터넷 응용 서비스와의 접목이 용이함과 확장성이 높다는 장점 때문에 VoIP 분야에서 뿐만 아니라 정보가전, 전자상거래, 화상 회의 등 폭 넓은 응용 분야에서 활용되고 있다.

마지막으로 서론에서 언급한 바와 같이 VoIP 분야에서 대부분의 통신/서비스 사업자 및 장비업체들은 SIP 기반의 VoIP 관련 시장이 조속히 열리기를 희망하고 있으며, 다행히도 국내외 전문가들이 2002년 후반부터 관련 시장이 빠른 속도로 활성화되리라 전망하고 있다. 기술적인 측면에서 본다면 인터넷 망에서의 서비스 품질과 신뢰성 보장, 보안 그리고 유용한 서비스 개발 등이 조속히 해결된다면 VoIP 관련 시장 활성화를 조금 더 앞당길 수 있리라 전망된다.

참 고 문 헌

[1] IETF SIP Working Group Home Page, <<http://www.ietf.org/html.charters/sip-charter.html>>

[2] IETF RFC 2543, SIP: Session Initiation Protocol, March 1999.

[3] IETF Internet Draft, draft-ietf-sip-rfc2543bis-03.txt, SIP: Session Initiation Protocol, May 2001.

[4] 이종화, 안상현, SIP기반 차세대 응용 기술, 한국정보처리학회지 제8권 제2호, 3.2001.

[5] M. Day, J. Rosenberg and H. Sugano, "A model for Presence and Instant Messaging", Request For Comment 2778, February 2000.

[6] J. Rosenberg, et al., SIP Extension for Instance Messaging, Internet Draft, February 2000.

[7] 홍용기, 문성남, SIP기반의 VoIP 시스템, 한국정보처리학회지 제8권 제2호, 3. 2001.

[8] Johnston, S.Donovan, et all, "SIP Telephony Call Flow Examples", Internet Draft, November 2000.

[9] Inmaculada Espigares del Pozo, An Implementation of the Internet Call Waiting Service using SIP,. Master's thesis at Helsinki University of Technology and Polytechnic University of Valencia. December 1999.

[10] IETF RFC 1889, RTP: A Transport Protocol for Real-Time Applications, January 1996.

[11] IETF RFC 2327, SDP: Session Description Protocol, April 1998.

[12] 현욱, 허미영, 이종화, 강신각, SIP 기반 Stateless Proxy 서버와 Registrar 서버의 설계 및 구현, COMSW2001, 7.28.2001.

[13] SIP forum home page, <http://www.sipforum.org/>

[14] IMTC home page, <http://www.imtc.org/>

[15] Columbia University, <http://www.cs.columbia.edu/~hgs/sip/implementations.html>

[16] Pingtel home page, <http://www.pingtel.com/>

[17] Vovida Open source web site, <http://www.vovida.org/>

[18] <http://www.cs.columbia.edu/~hgs/sip/sipit/classification.html>

[19] SIPit Interoperability Test, <http://www.sipforum.org/sipit.php>

[20] IMTC winter Interoperability Test, <http://www.ttc.or.jp/winter2001/>

이 종 화(Jong-Hwa Yi)

정회원



1990년 2월 : 한양대학교 전자공학과 졸업 (석사)

1990년 2월 : 한국전자통신연구원 통신프로토콜표준연구팀 선임연구원

1996년 : Technical University of Madrid, Spain 통신공학과 졸업 (박사)

<주관심 분야> VoIP, 인터넷 응용 프로토콜, 멀티미디어 통신 등

강 신 각(Shin-Gak Kang)

정회원



1984년 2월 : 충남대학교 전자공학과 졸업 (학사)

1998년 2월 : 충남대학교 전자공학과 졸업 (박사)

1984년 : 한국전자통신연구원 통신프로토콜표준연구팀 팀장

<주관심 분야> 멀티캐스트 통신, VoIP, 인터넷 보안