

SIP 시스템 분석 및 사용자 에이전트 모듈 구현

이 종 열⁰, 노강래, 김준일, 신동일, 신동규
세종대학교 컴퓨터공학과
{leemaster, krnoh, junil, dshin, shindk}@gce.sejong.ac.kr

Analysis of an SIP System and Implementation of an User Agent Module

Jong-Youl Lee⁰, Kang-Rae Noh, Jun-il Kim, Dongil Shin, Dongkyoo Shin
Dept. of Computer Engineering, Sejong University

요 약

본 논문에서는 현재 VOIP(Voice Over IP)기술로서 ITU-T의 H.323에 대응되는 텍스트 기반 프로토콜인 SIP(Session Initiation Protocol)를 구현한 오픈 소스인 VOCAL(Vovida Open Communication Application Library) 시스템의 구조를 분석하고 SIP의 시스템을 구성하는 여러 모듈들을 테스트를 통하여 검증하였다. VOCAL의 사용자 에이전트에 RTP(Real-time Transport Protocol) 포맷의 데이터를 전송/수신할 수 있는 모듈을 구현하고 검증하여, SIP 시스템에서 사용할 수 있는 사용자 에이전트의 활용에 대한 방안을 제시하였다.

관리, 가입자들의 인증, 과금 정보의 추적 등도 지원한다.

1. 서 론

SIP[1]는 ITU-T의 H.323에 대응되는 프로토콜로서 단말간 또는 사용자들간에 기존의 VoIP 서비스뿐만 아니라 다양한 서비스의 호 설정 프로토콜로서 사용자간의 멀티미디어 세션에 대한 세션간의 변경, 초기화 및 종료를 정의한 응용계층의 프로토콜을 의미한다. SIP를 구현한 Open project로는 콜럼비아대학[2]과 VOVIDA사의 VOCAL이 가장 많이 알려져 있다. 본 논문에서는 VOCAL 시스템의 SIP구현을 분석하고 테스트 모듈을 구현 하여 RTP 포맷 데이터의 송신/수신을 검증하였다.

③ 기능과 애플리케이션 플랫폼

- call forward, call blocking, call transfer, 그리고 call waiting 과 같은 기본적인 기능을 지원 하며, 새로운 기능과 애플리케이션 생성을 위한 소프트웨어 라이브러리(C++, Call processing language(CPL), Java telephony API (JTAPI))등을 지원한다.

2. VOCAL 소개

VOCAL[3]은 Open project이며 IP중심의 통신 소프트웨어이며, Linux, Solaris와 같은 운영체제에서 컴파일 및 실행 가능하다. VOCAL은 SIP에 필요한 각종 라이브러리를 지원하며, 다음의 기능을 지원한다.

VOCAL의 시스템 구조[4]는 VoIP(Voice over IP) telephony services를 지원하기 위한 분산 네트워크로 구성되며, SIP(Session Initiation Protocol), MGCP(Media Gateway Control Protocol), H.323 프로토콜들을 포함하는 다음의 Device들을 지원한다.

① SIP 기반의 Call Control and Switching으로써 사용자 등록(registration), Call 초기화, Call 수정(modification), 종료(termination)등을 지원한다.

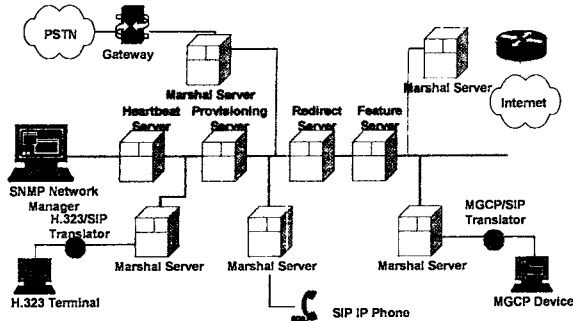
- SIP IP phones
- H.323 terminals
- MGCP endpoint
- Analog phone or the PSTN connected via SIP gateways

② Operation Support Services

- Web GUI(Graphic User Interface)로부터 시스템을 구성하고, 조정·통제 할 수 있다. SNMP 네트워크 관리자로부터 네트워크 요소들을 관찰할 수 있으며 가입자와 그들의 특정 가입신청의 추가와

[그림1]과 같이 Vocal의 분산 네트워크 구조를 구성하는 주요 서버들의 기능은 다음과 같다.

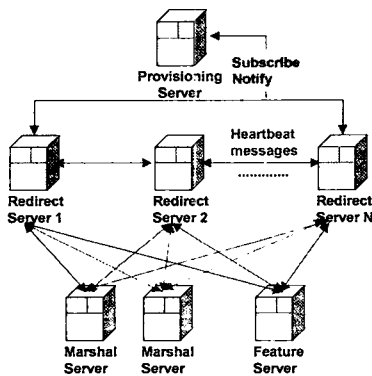
1) Marshal Server :SIP IP phones, user agents ,gateways와 routers과 같은 외부 개체들과의 접촉 점으로써의 역할만을 하며, 각 단말의 종류(PSTN,



[그림 1 Vocal 시스템 구조]

Internet, multiple 사용자 에이전트, Conference Bridge)에 따라 Marshal 서버의 4가지 형태가 존재하며, SIP 표준[5]에서 Proxy서버의 역할을 한다.

- 2) Heartbeat Server : 다른 서버들에 의해 보내지는 heartbeat들을 모니터링하고 SNMP 네트워크 관리자에게 이런 정보들을 forward한다.
- 3) Provisioning Server : 시스템 설정(provision)을 위해 사용되며, GUI로부터 분산구조의 시스템을 구성하고 관리한다.
- 4) Redirect Server : location, registration 그리고 redirection 기능지원하며 [그림 2]에서처럼 시스템 내의 Redirect server들은 같은 정보를 유지하기 위해 Heartbeat에 정보를 보내어 동기를 맞추며 주기적으로 Provisioning서버로 알려져서 정보 등을 갱신 한다.
- 5) Feature Server : call forwarding와 call blocking 과 같은 기본적인 특징을 구현 및 지원한다.



[그림 2 각 서버들의 상호 작용]

3. 사용자 에이전트(User Agent)

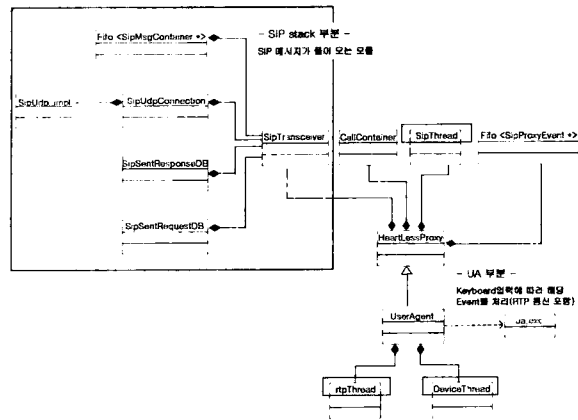
사용자 에이전트란 Vocal이나 SIP에서 Request 메시지나 Response 메시지를 송/수신하여 SIP 트랜잭션을 개시 및 종료하는 논리적 실체이며 사용자 에이전트에는 다음과 같은 것들이 될 수 있다.

- Cisco7960 SIP IP Phone
- Pingtel xpressa
- PC with softphone applicaion
- Komodo ATA 182/186

Vocal에서는 테스트를 위한 3가지 mode가 다음과 같이 설정되어있다.

- 1) Quicknet card : Linux Platform에서 특성화된 card(Quicknet card)를 사용하여 테스트 가능하다.
- 2) Sound card : Linux Platform에서 사용 가능하다.
- 3) NullHardware mode : 어떠한 장치도 없이 사용자 에이전트 소프트웨어로만 세션의 설정 모습을 볼 수 있다.

본 논문에서는 사용자 에이전트의 NullHardware mode를 선택하여 하드웨어 독립적으로 RTP통신[6] 모듈을 구현하여 텍스트 문서를 송수신하고 결과를 검증하였다. Vocal에서 구현된 사용자 에이전트의 클래스 구조도는 [그림 3]과 같으며 주요 클래스 흐름은 [그림 4]에 나타나 있다.



[그림 3 UA 전체 클래스 구조도]

3.1 RTP 테스트 모듈 구현

사용자 에이전트의 NullHardware 모드에서는 실제로 RTP 관련 함수의 API Prototype만 존재하고 함수의 내부는 구현되어 있지 않다. 따라서 [그림 5]에서 처럼 INVITE, ACK 메시지를 통한 SIP세션 설정만 이루어지고, 데이터의 송/수신은 이루어지지 않는다. 본 논문에서는 NullHardware 모드에서 RTP 관련 함수를 구현하여 RTP 통신을 가능하도록 하였다. 사용자 에이전트에서 INVITE를 보내고 ACK 메시지를 받은 후 서로 RTP 정보를 교환한 후 RTP 포맷의 데이터를 송/수신한다.

3.2 사용자 에이전트 테스트[7]

우선 사용자 에이전트의 테스트 전 상대 IP, ID, Speed

