

위성 멀티미디어 액세스망에서 인터넷 실시간 전달 프로토콜 구현

이계상* · 진광자** · 안재영** · 오덕길**
*동의대학교 · **한국전자통신연구원

An Implementation of Internet Real-Time Protocol for Satellite Multimedia Access Networks

Kye-Sang Lee* · Gwang-Ja Jin** · Jae-Young Ahn** · Deock-Gil Oh**
*Donggeui University · **ETRI
E-mail : ksl@hyomin.donggeui.ac.kr

요 약

최근 들어 위성 통신 기술의 발전으로 위성을 이용한 네트워크 액세스가 기존 액세스망의 또 다른 보완 수단으로 유력해지고 있다. 위성을 이용하여 멀티미디어 서비스를 가입자에게 제공할 수 있는 액세스망에서 인터넷 접속은 빼놓을 수 없는 중요한 서비스가 될 것이다. 특히, 실시간 트래픽이 급증하고 있는 인터넷 접속 서비스에서 이를 지원할 수 있는 실시간 전달 프로토콜의 연구는 중요하다.

본 연구에서는 그동안 IETF에서 표준 프로토콜로 개발되어 온 실시간 전달 프로토콜 (RTP: Real Time Protocol)의 위성 멀티미디어 액세스망 적용을 위해 동 프로토콜을 구현하였다. RTP는 현 인터넷을 통해 오디오/ 비디오를 전달할 때 실시간 트래픽의 엄격한 시간 제약성을 만족시키기 위해 개발된 프로토콜로서, 위성 액세스망과 같은 고유한 링크 특성을 갖는 환경에서 적용성을 확인할 필요가 있다. 본 논문에서는 RTP 프로토콜의 개요를 살펴보고, 프로토콜 설계 구조와 주요 구성 요소 및 그 구현 내용을 기술한다.

ABSTRACT

Recently, with the advent of satellite communication technologies, network access services through satellites have emerged as an alternative to existing access network services. Of satellite access services including multimedia services, the high speed internet access service will be one of the important services. Particularly, the efficient transportation of real time internet traffic seems very essential.

The aim of this research is to examine the applicability of the IETF RTP protocol in the satellite access network. The RTP has been developed for the QoS-oriented transportation protocol of real time traffic over the existing best-effort internet. This paper summarizes the RTP protocol, and describes the architecture and the data structure of the implementation of the RTP.

1. 서 론

최근 인터넷 사용이 급증함에 따라 인터넷에 고속으로 접속할 수 있는 DSL 또는 케이블모뎀과 같은 유선 인터넷 액세스망 기술의 발전이 두드러졌다. 한편, Ka 밴드와 같은 위성 통신 기술

의 발전으로 광대역 쌍방향 위성 액세스망이 상용 가능한 또 하나의 인터넷 액세스망 기술로 대두되고 있다[1,2].

인터넷에서 그동안 점점 더 비중이 높아지고 있던 실시간 응용 서비스의 지원은 위성 인터넷 접속에서는 더욱 중요해질 전망이다. 위성 인터

넷 접속에서는 특히 인터넷 방송과 원격 교육과 같은 IP 멀티캐스팅을 이용한 실시간 응용 서비스들이 중요한 서비스로 등장될 것으로 예측되기 때문이다. 한편, 인터넷 실시간 서비스지원을 실현하는 RTP 프로토콜은 거의 모든 실시간 멀티미디어 트래픽의 전달 프로토콜로 자리를 잡아갈 전망이다. 최근에 급속히 보급되고 있는 VoIP와 스트리밍 기술을 이용한 인터넷 방송 등의 응용 서비스 트래픽의 전달에 모두 RTP가 이용되고 있다. 따라서, 본 연구에서는 고유한 특성을 갖는 위성 액세스망에 적용성을 검토하기 위하여 RTP 프로토콜을 설계하고 구현하였다.

II. RTP 프로토콜 개요

RTP 프로토콜은 실시간 트래픽 전달을 위한 새로운 프로토콜로서 인터넷 표준기구인 IETF의 avt 워킹 그룹에서 개발한 표준 프로토콜이다 [3,4]. 이 프로토콜은 오디오/ 비디오 컨퍼런스를 염두에 두고 개발된 중단에 위치하는 응용 계층 프로토콜로서 실시간 트래픽의 엄격한 송수신 시간 제약성이 고려되었다.

RTP 프로토콜은 응용 계층 프레임링 (Application Layer Framing) 구조를 갖는다. 이는 RTP가 실시간 응용 프로토콜에 공통으로 필요한 기본 동작과 표준 포맷 정도를 규정하는 간단한 프레임워크를 제공함을 의미한다. 따라서, 실제 하나의 완전한 실시간 응용 프로토콜은 RTP가 제공하는 프레임워크에 응용 고유의 기능들이 추가되어야 완성된다. RTP는 통상 UDP와 함께 실시간 멀티미디어 응용을 위한 전송 계층을 구성하며 UDP 프로토콜에 타이밍, 순서 유지 기능을 추가하여 대부분 응용 프로토콜에 필요한 공통 기능을 제공한다.

RTP 프로토콜이 제공하는 기본 기능에는 타임스탬핑과 멀티캐스트의 지원을 들 수 있다. 타임스탬핑은 송신측에서 패킷들이 전송될 때 이 들간의 상대적 시간 간격이 수신측에서도 그대로 유지되어 재생될 수 있도록 하는 기능이다. 소스가 모든 메시지에 타임스탬프를 표시하면 수신측에서는 이를 받아보고 패킷간 타이밍 정보를 추출하여 원만한 재생에 이용한다. 이 기능을 통해 패킷이 인터넷을 경과하면서 받는 지연 변이 (jitter)가 보정될 수 있다.

RTP 프로토콜은 오디오/ 비디오 컨퍼런스를 염두에 두고 개발된 프로토콜로 그 동작은 멀티캐스트 환경을 기본으로 하고 있다. 여러 사용자가 하나의 컨퍼런스에 참여하여 이룬 멀티캐스트 그룹에서 한 사용자의 발신 트래픽은 나머지 모든 사용자에게 전달된다. RTP 프로토콜과 함께 개발된 제어용 프로토콜인 RTCP (Real Time Control Protocol) 프로토콜도 멀티캐스트로 동작된다.

RTP는 모든 메시지에 동일한 단일 메시지 포맷을 갖는다. 실시간 트래픽의 전달에 효율적이도

록하고 대역폭의 낭비를 막고 지연 시간을 줄이는 등 오버헤드를 방지한 매우 간결한 헤더 포맷을 갖는다. 다음에 패킷 헤더 중 중요한 필드를 소개한다.

처음 16 bit 영역은 다음과 같이 구분된다. 버전 필드는 2 bit로 RTP 버전을 표시한다. 1 bit 짜리 Padding 필드는 패킷의 응용 데이터 영역에 padding 부분이 있는지를 표시한다. 다음 1 bit Extension 필드는 RTP 고정 헤더 외에 또 다른 확장 헤더가 있는지를 나타낸다. 응용에 따라 추가로 헤더가 확장될 필요가 있을 때 이를 이용한다. Contributor counter 필드는 믹서가 사용될 때 필요한 필드로 몇 개의 소스 스트림이 합쳐진 것인지를 표시한다. 1 bit Marker 필드는 페이로드의 프레임 끝을 표시하는데 이용된다. 예를 들어 압축된 비디오 프레임의 경우 한 장의 비디오 프레임의 끝이 이 필드를 통해 표시된다. 페이로드 필드는 7 bit로 페이로드의 유형을 표시한다. 이 필드를 통해 PCM 음성, MPEG 비디오, H.261 비디오 등을 구분한다.

이상 16 bit 영역에 이어 다음 16 bit 영역은 RTP 패킷의 순서를 나타내는 Sequence 필드이다. 전송 도중 혹시 일어날 수 있는 순서 바뀔을 이 필드의 정보를 통해 바로 잡을 수 있다. 다음 32 bit는 RTP 프로토콜에서 가장 기본적 동작을 지원하는 타임스탬프 필드이다. 하나의 트래픽 스트림내에서 패킷간 동기를 맞춰주는데 이용된다. 다음 32 bit 필드들은 SSRC 필드와 CSRC 필드로 각각 트래픽의 송신자와, 믹서 사용 경우 합쳐진 트래픽 스트림을 구성하는 원 소스들을 식별한다.

이상과 같은 패킷 포맷을 통해 구현된 RTP의 기능은 다음과 같이 요약될 수 있다. 첫째, 스트림내 동기 기능이다. 이것은 타임스탬프 필드를 이용하여 패킷간 시간 동기를 맞춰 원만한 재생을 꾀한다. 둘째, 패킷 순서 맞춤 기능이다. Sequence 필드를 이용하여 패킷의 손실을 검출하고 순서 오류시 바른 순서를 회복한다. 셋째는 페이로드 식별 기능으로서, 페이로드 유형 필드를 통해 그 때 그 때 패킷 단위로 페이로드의 유형을 식별한다. 모든 패킷이 페이로드 유형을 가짐으로써 망 QoS 변화에 따라 페이로드 형태의 변경이 가능하다. 넷째, 프레임 식별 기능이다. Marker 필드를 이용하여 페이로드의 큰 단위의 시작 또는 끝을 식별하는 기능이다. 마지막으로, 소스 식별 기능을 들 수 있다. 멀티캐스트 환경에서는 여러 소스가 존재하므로 SSRC 필드를 통해 특정 소스를 식별한다.

이상 살펴 본 RTP 프로토콜은 실시간 트래픽을 전달하는 기능만을 담당하며, 프로토콜 제어에 필요한 별도의 기능은 RTCP 프로토콜을 통하여 달성한다. RTCP 프로토콜은 RTP 프로토콜과 함께 IETF avt 워킹 그룹에서 개발되었다 [3]. RTP가 응용 데이터를 전달하는 단 하나의 메시지 유형만을 제공하므로 수신측으로 부터의 피드백 또는 응답을 위해 개발된 제어 프로토콜이다. RTP와

분리된 프로토콜이지만 밀접한 관계를 가지며 RTP와 짝을 이뤄 동작한다. 즉, RTP 프로토콜은 짝수의 포트 번호를 사용하고, RTCP 프로토콜은 통상 이 보다 1이 더 큰 홀수의 포트 번호를 사용한다.

RTCP 프로토콜의 주 기능은 QoS 관리, 미디어 간 동기, 소스 스트림 정보 식별 등이다. 이를 위해 5 개의 메시지 유형을 갖는다. 송신자 보고 메시지, 수신자 보고 메시지, 소스 식별 메시지, 종료 메시지, 응용 고유 메시지가 그것이다. RTCP 프로토콜 메시지들 역시 RTP 패킷들과 함께 멀티캐스트된다.

송신자 보고 메시지는 RTP 데이터 송신자가 주기적으로 수신자들에게 보내는 메시지로서, 일정 기간 중 자신이 보낸 총 데이터 패킷 수와 바이트 수 등 송신 트래픽 정보를 수신자에게 전달한다. 송신자 보고 메시지에는 자신이 수신자이기도 한 경우 그동안 수신한 데이터에 대한 수신자 보고도 곁할 수 있다.

수신자 보고 메시지는 RTP 패킷을 수신하는 수신자가 송신자에게 주기적으로 보내는 메시지로서 패킷의 수신 통계 자료를 담고 있다. 수신자는 송신자 보고를 통해 받은 송신 패킷 정보를 갖고 실제 수신한 패킷 수와 비교하여 패킷 손실율 및 누적 손실 패킷 수를 계산한다. 이러한 수신 통계치와 그동안 수신한 최고 sequence 번호, 패킷 수신 jitter, 송신자 보고 최종 수신 시간 정보를 수신자 보고에 포함시킨다.

소스 식별 메시지는 소스가 자신의 정보를 멀티캐스트 그룹내 모든 수신자에게 주기적으로 알리는데 사용하는 메시지이다. 소스 정보에는 소스 이름, 사용자 이름, E-mail 주소, 전화번호, 주소, 응용 프로그램명 등이 있다. 이 중 소스 이름인 CNAME은 여러 RTP 스트림을 하나의 송신자에 연관시키는데 사용하여, 소위 intermedia synchronization을 달성한다. 예를 들어, 하나의 송신 호스트로부터 발신되는 오디오와 비디오 동기를 맞추는데 사용한다.

종료 (Bye) 메시지는 소스가 어떤 멀티캐스트 그룹을 떠날 때 이 사실을 다른 멤버들에게 알리기 위해 사용하는 메시지이다. 응용 고유 메시지는 RTP를 사용하여 새로운 응용 프로토콜을 개발할 때 응용 고유의 특징과 실험을 하기 위한 용도로 예비된 메시지이다. 시험 후 폭 넓은 사용이 정당화되면 새로운 RTCP 패킷으로 IANA에 등록된다.

이상에서 살펴 본 RTP/ RTCP 프로토콜은 전달 품질이 보장되지 않는 현 best-effort 인터넷에서 실시간 트래픽의 시간 제약성을 어느 정도 충족시키고자 하는 프로토콜이다. 하지만, Integrated service와 Differentiated service와 같은 IP QoS 프로토콜에서와 같이 QoS 보장을 위한 어떤 자원 예약 기능은 수행되지 않음이 주의되어야 한다.

RTP/ RTCP 프로토콜은 최근 H.323의 일부 프

로토콜로 쓰여 VoIP 서비스 제공에 응용되고 있고, 더욱 최근 들어서는 SIP/ SDP 프로토콜과 함께 쓰여 VoIP 뿐 아니라, 인터넷 방송과 원격 교육과 같은 스트리밍 서비스에도 활용되고 있고, 향후 넓은 범위의 A/V 컨퍼런스에 응용될 예정이다[7].

III. 구현 구조 및 설계

본 장에서는 프로토콜 구현을 위한 전체 구조와 데이터 구조의 설계 내용을 기술한다.

프로토콜 구현 구조는 오디오나 비디오 등 각각의 미디어 스트림에 대해 별도의 세션을 구성하는 것을 가정하였다. 이들 서로 다른 미디어 간 intermedia synchronization은 RTP 메시지의 CNAME을 통해 달성한다. 하나의 세션은 다시 RTP 전송기와 수신기, 그리고 RTCP 전송기 및 수신기로 구성한다. 물론, 수신만 하는 경우 또는 송신만 하는 경우, 각각 RTP 수신기 또는 송신기만 동작될 수 있도록 한다. 그림 1은 이러한 실시간 트랜스포트 프로토콜 구현 구조를 도시한다.

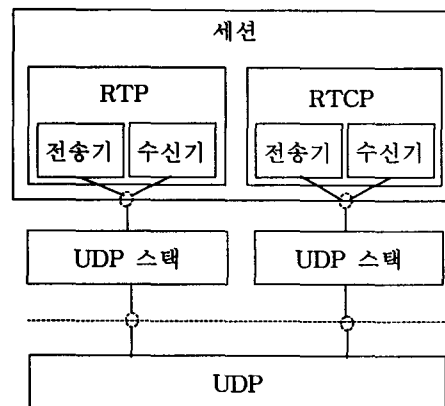


그림 1. 실시간 트랜스포트 프로토콜 구현 구조

세션 객체는 어떤 추가적인 기능을 담당하기 보다는 RTP 및 RTCP의 전송기 수신기 객체가 제공하는 기능을 encapsulation하여 API를 제공함으로써 상위 응용 프로토콜 작성을 용이하게 함을 목적으로 한다.

RTP 전송기는 RTP 패킷들을 생성하고, RTP 패킷을 전송하는 기능을 담당하며, RTP 수신기는 네트워크로부터 RTP 패킷을 수신하여 동기를 맞춰 상위 프로그램에 넘겨 주는 기능을 담당한다. RTP 전송기와 수신기는 각각 버퍼를 갖는다. 그림 2 참조. RTP 전송기의 버퍼는 상위 응용 프로그램에서 페이로드를 포함한 데이터를 API 크기로 받아 일단 저장한 뒤 네트워크 전송에 맞는 단위로 변환하기 위한 장소로 쓰인다. RTP 수신기의 버퍼는 전송기 버퍼의 패킷 크기 변환 기능 이외에 인터넷 통신망에서 발생한 각 패킷의 지

연변이를 흡수하는 기능을 담당한다.

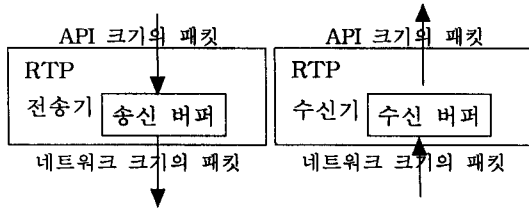


그림 2. RTP 전송기와 수신기 버퍼

RTP 전송기와 수신기는 동일한 소켓을 통해 하나의 UDP 프로토콜 스택을 공유한다. UDP 스택은 포트 번호, 소켓 번호 등을 모아 놓은 자료 구조이며, 소켓 프로시저를 간접 호출하여 RTP 및 RTCP 전송기와 수신기가 UDP 소켓을 쉽게 사용하도록 한다.

RTCP 프로토콜 스택도 RTP 스택과 유사하게 설계되었다. RTCP 전송기는 RTCP 메시지를 생성하여 전송하는 기능을 담당하며, 수신기는 RTCP 패킷을 수신하여 검사하고 유형별로 그 내용을 해독한다. RTCP 전송기와 수신기도 별도의 UDP 스택을 공유한다.

다음에 RTP와 RTCP를 구성하는 주요 데이터 구조를 살펴 본다. RTP 데이터 구조는 패킷 클래스, 패킷 헤더 클래스, 전송기 클래스 및 수신기 클래스로 크게 구성된다.

RTP 패킷은 패킷 클래스와 헤더 클래스로 구성된다. 패킷 클래스는 실제 패킷 데이터를 갖는 메모리 영역을 가리키는 포인터와 패킷 헤더를 가리키는 포인터로 구성되며, 패킷의 크기 등 패킷에 관련된 일반 정보를 포함한다. 또한 이러한 패킷 정보에 접근할 수 있는 멤버함수를 제공한다. 패킷 헤더 클래스는 구조체로 정의되며 RTP 프로토콜 패킷 헤더의 여러 필드를 담고 있다.

RTP 전송기 클래스는 RTP 패킷을 만들고 UDP 스택을 이용하여 패킷을 전송하는 기능을 담당한다. 사용자 프로그램에서 RTP API를 통해 페이로드가 넘어오면 이를 송신 버퍼에 잠시 보관하였다가 네트워크 패킷 크기에 맞추어 전송한다. 패킷 수신지에 관한 정보는 생성자 기동시 `remoteHost`와 `remotePort` 인수를 통해 얻는다. 전송기 클래스는 전송기 관련 여러 변수를 갖고 있으며, 또한, QoS 모니터링을 위해 RTCP 전송기와 수신기가 이 클래스의 변수에 접근할 수 있도록 한다. 패킷의 전송은 궁극적으로 UDP 스택을 통해 이루어지며 아울러 UDP 정보에 접근하는 방법도 제공한다.

RTP 수신기 클래스는 네트워크로부터 패킷을 수신하여 jitter 제거를 위한 입력 버퍼를 거쳐 패킷 재생 시각에 맞춰 패킷을 재생하는 과정에 필요한 기능 들을 주요 멤버함수로 갖는다. 패킷을 수신하면 jitter 값을 갱신하고 사용자 프로그램에 패킷을 넘겨 줄 재생 시각이 되었으면 이를 시행한다. 어떤 소스로부터 최초의 패킷이 도착하면 이 소스를 수신기에 등록시키고 관련 정보를 초

기화한다. QoS 모니터링을 위해 RTCP 프로토콜과 정보를 공유한다.

RTCP 데이터 구조도 RTP 데이터 구조와 유사하게 패킷 클래스, 패킷 헤더 클래스, 전송기 클래스 및 수신기 클래스로 구성된다.

IV. 구현 환경 및 테스트 프로그램

Red Hat 6.1 리눅스[5]가 탑재된 펜티엄 PC 상에서 C++ 언어와 GNU C 컴파일러 및 디버깅 툴[6]을 사용하였다. RTP/ RTCP 프로토콜 기능 구현 부분은 라이브러리화하고, 테스트 프로그램과 같은 응용 프로그램과 링크하여 사용하였다. 응용 프로그램 예제로서, PCM 코딩된 음성 송수신하는 것을 가정하여, 20 ms 마다 160 바이트씩을 갖는 패킷을 주기적으로 송수신하는 테스트 프로그램을 작성하여, 동일 호스트의 loop-back 인터페이스를 통해 시험하였다.

V. 결 론

본 논문에서는 RTP 프로토콜의 위성 액세스망에서의 적용성 검토를 위한 사전 연구로서 RTP 프로토콜을 소개하고, 이의 구현 구조와 데이터 구조 설계 내용을 요약하였다. 현재 프로토콜의 구현 기능을 시험하고, 이를 루프백 인터페이스를 통한 동일 시스템내에서 시험을 완료하였다.

향후, 모의 위성 링크를 이용하여 구현한 RTP 프로토콜의 성능 시험을 수행하여, 위성 액세스망에서 RTP 프로토콜의 적용성 연구를 진행할 예정이다.

참고문헌

- [1] 김내수, 최동준, 오덕길, "위성/ 지상 ATM 망 연동 및 서비스 제공방안 고찰", 한국통신학회 정보통신지, pp. 70-86, 1999.6
- [2] Special Issue, "Satellite ATM Network Architecture", IEEE Comm. Mag., 1999.3
- [3] H. Schulzrinne, et. al., "RTP : A Transport Protocol for Real-Time Applications", IETF RFC 1889, 1996.1
- [4] H. Schulzrinne, et. al., "RTP Profile for Audio and Video Conference with Minimal Control", IETF RFC 1890, 1996.1
- [5] C. Negus, "Red Hat Linux Bible", IDG Books, 1999
- [6] R. Stones, et. al., "Beginning Linux Programming", Wrox, 1999
- [7] 이계상, "멀티미디어 데이터 전송용 프로토콜", 정보과학회 주최 정보통신 기술 단기간좌 - 인터넷 실시간 서비스, 이화여대, 2000.2