멀티미디어 서비스 트래픽 모니터링과 분석

강훈정 O, 김명섭, 홍원기

포항공과대학교 컴퓨터공학과 분산처리 및 네트워크관리 연구실

{bluewind, mount, jwkhong}@postech.ac.kr

요 약

최근 멀티미디어 서비스 트래픽이 급격히 증가하고 있다. 이러한 경향은 인터넷 사용자 증가, 초고속 네트워크 보급, PC 성능 향상 및 멀티미디어 컨텐츠의 증가로 인해 더욱 가속화될 것이다. 따라서 멀티미디어 서비스 트래픽이 네트워크의 성능, 상태, 운영에 미치는 영향이 더욱 증대될 것이고, 이러한 트래픽을 모니터링하고 분석하는 일이 중요하게 되었다. 그러나 아직까지 대부분의 기업이나 인터넷 서비스 제공 사업자들은 멀티미디어 서비스 트래픽에 대한 정보를 정확히 파악하지 못하고 있다. 이것은 이러한 종류의 트래픽에서는 동적으로 할당된 트랜스포트 계층의 프로토콜과 포트 넘버를 이용하여 멀티미디어 데이터를 전송하기 때문이다. 따라서 잘 알려진 포트 번호로 상위 어플리케이션을 판단하던 기존 대부분의 분석 방법으로는 멀티미디어 서비스 트래픽을 분석할 수 없게 되었다. 본 논문에서는 이러한 문제를 해결하기 위해 멀티미디어 서비스 트래픽을 모니터링하고 분석하기 위한 방법을 제시한다. 이 방법은 제어 세션의 패킷을 분석해서 새롭게 동적으로 생성되는 세션의 정보를 추출해 내고 이 정보를 이용해 멀티미디어 서비스 트래픽의 어플리케이션을 판별한다. 또한 멀티미디어 서비스 트래픽을 모니터링하고 분석하기 위한 구현 시스템과 그 분석 결과에 대해서도 기술한다. 본 논문에서 제시된 방법은 현재 네트워크 상에서의 멀티미디어 서비스 트래픽 이용 현황을 파악하고, 서비스에 대한 요금 징수 및 고객관계관리를 위한 정보를 제공할 수 있다.

1. 서론

인터넷 사용자가 전세계적으로 급격히 증가하고, 네트워크 기반의 응용 프로그램이 다양하게 개발되고 사용됨에 따라 네트워크 트래픽이 급격히 증가하고 있다. 이러한 수요 증대에 맞추어 네트워크가 계속해서 증설되고 새로운 네트워크 기반 서비스들이 생겨나고 있다. 이런 상황에서 네트워크 상의 트래픽을 관찰하여 사용자들이 얼마나 트래픽을 발생시키고 어떤 서비스나 어플리케이션을 사용하는지에 대한 경향을 파악하는 일들이 요구된다. 이처럼 네트워크 트래픽 모니터링은 네트워크의 현재 상황 및 문제점을 파악하고 네트워크 회선에 대한 계획 수립, 서비스 제공 평가 등을 위한 정보를 제공할 수 있다[1,2,3].

최근의 네트워크 트래픽의 두드러진 추세 중 하나는 스트리밍이나 멀티미디어 컨퍼런싱과 관련된 트래픽의 급격한 증가에 있다. 본 논문에서는 이 두 가지 종류의 트래픽을 멀티미디어 서비스 트래픽이라 칭하기로 한다. 현재 많은 사이트에서 방송 프로그램, 영화, 음악 등의 스트리밍 컨텐츠를 제공하고 있고 이에 대한 사용자의 수요도 증가 추세이다. 또한 화상 회의와 같은 양방향성의 멀티미디어 전송 서비스도 도입되고 점차 널리 사용되어지고 있다. 이러한 경향은 인터넷 사용자 증가, 초고속 네트워크 보급, PC 성능 향상 등으로 인해 더욱 가속화되고 있다. 이렇게 멀티미디어 서비스 트래픽이 네트워크에 미치는 영향이 더욱 커지고 있고, 따라서 멀티미디어 트래픽 분석은 네트워크 이용 현황을 파악하는데 중요한 부분을 차지하게 된다.

그러나 아직까지 대부분의 기업이나 인터넷 서비스 제공 사업자(Internet Service Provider, ISP)들은 멀티미디어 서비스 트래픽에 대해서 사용자들이 어떤 서비스를 얼마나 사용하는지에 대한 정보를 파악하지 못하고 있다. 이것은 네트워크 트래픽이 대용량화되고 기존의 텍스트나 이미지 위주의 트래픽이 멀티미디어 위주의 트래픽으로 변하고 있는 특성에 맞는 모니터링 및 분석 시스템이 존재하고 있지 않기 때문이다. 기존 대부분의 모니터링 시스템에 주로 쓰인 트래픽 분석 방법은 잘 알려진 포트 번호(well-known port)만을 갖고서 그 상위 어플리케이션을 판단하던 것이었다. 그러나 멀티미디어 서비스의 경우 제어 프로토콜에서 동적으로 데이터를 전송할 세션에 대한 협상을 한 후 멀티미디어 서비스 컨텐트를 전송하게 된다. 따라서 멀티미디어 서비스를 전송할 때는 잘 알려진 포트 번호가 아니라 동적으로 할당된 프로토콜과 포트 번호를 사용하기 때문에 이러한 트래픽은 기존의 분석 방법으로 판단했을 경우 어플리케이션을 알 수 없는 트래픽으로 나타나게 되어 트래픽 분석이 어렵게 된다[4].

본 논문에서는 이러한 문제를 해결하기 위해 플로우 기반의 일반 모니터링 시스템에 적용되어 응용될 수 있는 멀티미디어 서비스 트래픽의 모니터링 및 분석 방법을 제시한다. 모니터링 및 분석은 현재 가장 많이 쓰이고 있는 스트리밍 서비스, 멀티미디어 컨퍼런싱 및 IP 네트워크에서의 전화 서비스(VoIP)의 트래픽을 그 대상으로 한다. 이 방법들은 멀티미디어 서비스를 위한 제어 프로토콜과 관련된 패킷의 페이로드를 분석하여 멀티미디어 서비스 데이터를 전송하는데 사용될 프로토콜과 포트번호를 추출한다. 이 정보를 이용해 기존 방법에서는 어떤 어플리케이션인지 알 수 없었던 멀티미디어 서비스 트래픽을 판별할 수 있게 된다.

2. 관련 연구

이 장에서는 배경 지식으로 멀티미디어 서비스 관련 프로토콜과 기존 연구를 소개한다.

2.1 멀티미디어 서비스 관련 프로토콜

스트리밍은 전송되는 데이터를 마치 끊임없고 지속적인 물 흐름처럼 처리할 수 있는 기술을 의미한다. 현재 많은 스트리밍 서비스가 시도되고 실용화되어 왔지만, 그 중에서도 가장 많이 쓰이고 있는 서비스 플랫폼으로는 WMT(Windows Media Technology)[5], RealMedia[6], QuickTime[7]을 들 수 있다. 표 1 은 이 서비스들에서 이용되는 프로토콜들을 보여준다.

스트리밍 서비스	제어 세션 프로토콜	데이터 세션 프로토콜
RealMedia	RTSP	RDT
QuickTime	RTSP	RTP
WMT	MMS	MMST/MMSU

표 1. 서비스 별 스트리밍 미디어 프로토콜

현재 멀티미디어 컨퍼런싱 및 IP 기반 전화 서비스 어플리케이션들은 SIP(Session Initiation Protocol)[8]나 H.323[9]에 기반한다. 이 때 사용되어지는 프로토콜들이 표 2에 기술되었다.

어플리케이션	제어 세션 프로토콜	데이터 세션 프로토콜
SIP 기반	SIP	RTP
H.323 기반	Q.931, H.245	RTP

표 2. 멀티미디어 컨퍼런싱 프로토콜

멀티미디어 서비스를 위해 두 종류의 세션이 형성된다. 하나는 서비스 동안 연결 설정이나 서비스 제어를 담당하는 세션이다. 스트리밍 서비스에서는 RTSP[10], MMS[5]와 같은 제어 프로토콜을 사용하고, 멀티미디어 컨퍼런싱 서비스에서는 SIP, Q.931[9], H.245[9] 프로토콜이 이용된다. 이 때 형성되는 세션을 본 논문에서는 제어 세션(control session)이라고 부르기로 한다. 나머지 하나는 RDT[6], RTP[11], MMST/MMSU[5]와 같은 전송 프로토콜을 사용해서 서버에서 클라이언트에게 혹은 양 종단간에 실제로 멀티미디어 데이터를 전송하는 세션이다. 이후 이 세션을 데이터 세션(data session)이라 칭한다.

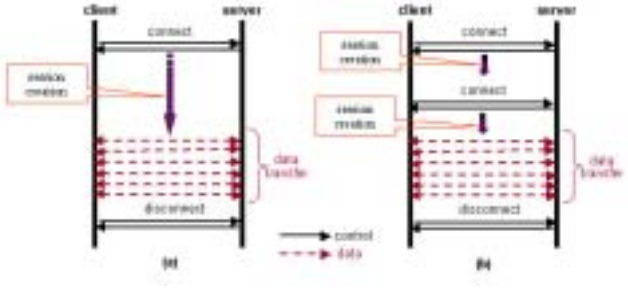

그림 1. 멀티미디어 서비스 제어 및 데이터 세션

그림 1은 제어 세션과 데이터 세션이 생성되어 멀티미디어 데이터가 전달되는 과정을 보여준다. 우선 처음에는 잘 알려진 포트 번호를 통해서 제어 세션이 생성된다. 이 때 (a)의 스트리밍 서비스나 SIP 기반 서비스의 경우에는 한 개의 제어 세션이 생기지만, (b)의 H.323 같은 경우에는 제어 세션이 한 단계 더 생긴다. 이렇게 제어 세션이 생성되고 나면 제어 프로토콜을 이용해 전송 프로토콜과 포트 번호 등과 같이 멀티미디어 데이터 전송에 대한 사항들을 협상한 후 데이터 세션을 생성시켜 멀티미디어 데이터를 전송하게 된다. 여기서 본 논문은 (a), (b)의 데이터 세션이나 (b)의 두 번째 제어 세션과 같이, 이전에 성립된 제어 세션의 협상에 의해서 동적으로 할당된 전송 프로토콜과 포트 번호를 이용하는 세션을 다이나믹 세션(dynamic session) 이라고 부르기로 한다.

이 때 제어 세션에 의해 다이나믹 세션의 정보를 협상할 경우 제어 세션 패킷의 페이로드에는 새로운 세션에서 사용할 전송 프로토콜이나 포트 번호 등의 협상 내용이 나타나게 된다. 따라서 이러한 제어 세션 패킷의 페이로드를 분석함으로써 다이나믹 세션에 대한 정보를 찾아낼 수 있게 된다.

2.2 관련 트래픽 모니터링 시스템

이 절에서는 플로우(flow)의 개념을 도입하고 본 논문에서 참고로 한 플로우 기반의 **NG-MON**(Next Generation Monitoring)[12] 시스템 구조에 대해서 기술한다. 그리고 멀티미디어 서비스 트래픽 모니터링을 시도했던 기존의 시스템으로 **FlowScan**[16]과 **mmdump**[4]를 소개한다.

플로우란 특정 두 종단(end points)간에 지나다니는 일련의 패킷(packet)들을 모아 부르는 것으로, 다양한 정의들이 있지만[13,14,15] 본 논문에서는 일정 시간 동안 근원지 IP 주소, 목적지 IP 주소, 근원지 포트 번호, 목적지 포트 번호 및 전송 프로토콜 번호 5 개 항목이 같은 패킷들의 집합으로 정의한다. 같은 플로우에 속하는 패킷들을 통합함으로써 처리해야 할 트래픽 데이터를 줄일 수 있기 때문에 많

은 시스템에서 플로우에 기반해서 트래픽을 분석하고 있다.

NG-MON 은 플로우에 기반해서 초고속 IP 네트워크 트래픽을 모니터링하기 위한 시스템이다. 대용량 네트워크 트래픽을 처리하기 위해 트래픽 모니터링 단계를 5 단계로 구분한 후, 각 단계를 파이프라인 구조로 연결하였다. 각 단계마다 한 대 이상의 시스템에서 수행이 가능하도록 병렬 구조를 적용하였다. NG-MON 의 전체 구조는 다음과 같다.

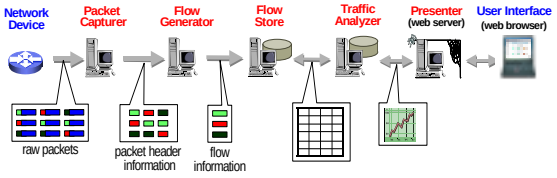

그림 2. NG-MON 시스템 구조

그림 2 에서 보는 바와 같이 플로우 정보 생성을 위해 패킷 수집, 플로우 생성, 플로우 저장의 세 단계로 나누어 처리하도록 하였다. 그리고 이러한 플로우 정보에 기반해 트래픽을 분석하는 트래픽 분석 단계와 프레젠테이션 단계를 통해 트래픽 정보로 가공되어 사용자에게 분석 결과를 보여준다. 각 단계를 각기 다른 하나의 시스템이나, 혹은 로드를 감당하지 못할 경우에는 여러 시스템에 설치되어, 병렬로 동시에 수행 가능한 유연성을 제공함으로써 전체 로드를 분산할 수 있다. 이러한 시스템 구조는 본 논문의 시스템에서 채택되어졌다.

FlowScan 도 플로우에 기반한 트래픽 모니터링 시스템이다. 이 시스템은 수집된 패킷에서 멀티미디어 서비스 제어와 관련된 잘 알려진 포트 번호가 감지되면 그 이후 두 호스트 사이에 지나다니는 트래픽 중에서 어플리케이션을 판단할 수 없는 트래픽들을 모두 멀티미디어 서비스 트래픽이라고 간주하여 분석하는 방법을 사용한다. 이러한 방법은 멀티미디어 서비스 트래픽 분석을 위한 별도의 처리 오버헤드가 없이 쉽게 분석을 할 수 있다는 장점이 있다. 그러나 두 호스트 사이에 멀티미디어 서비스와 관련된 포트 번호가 감지되고 난 후에 두 호스트 사이에 발생하는 트래픽 중에는 멀티미디어 서비스 트래픽이 아닌 경우도 있다. 그리고 이 방법은 처음에 쓰였던 서버와는 다른 서버를 이용해 멀티미디어 데이터를 전송할 경우에도 적용될 수가 없기 때문에 부정확한 분석 결과를 낼 수 있다.

mmdump 는 인터넷 상의 멀티미디어 트래픽을 모니터링하기 위한 툴로 패킷 필터를 변경해가며 멀티미디어 서비스 관련 패킷만 수집해서 분석하는 방법을 이용한다. 이 때 패킷 필터에는 수집될 패킷의 포트 번호가 등록되어 있어서, 등록된 포트 번호를 가진 패킷만 캡쳐할 수 있다. 이를 위해서 이 툴은 제어 세션 패킷들을 분석해서 데이터 세션을 위해 새롭게 할당된 포트 번호를 찾아낸다. 이후 새로 찾아낸 포트 번호에 해당하는 패킷들을 잡아낼 수 있도록 패킷 필터를 바꾸어 적용시킨다. 그러나 이 시스템은 다음과 같은 문제점들을 가지고 있다. 첫째로 현재 스트리밍 서비스에 많이 이용되고 있는 WMT 트래픽은 분석하지 않았다. 이 서비스에 이용되는 프로토콜이 비공개되어 있어서 제어 세션을 정확하게 분석 하기가 어렵기 때문이다. 또한 IP 단편화(IP fragmentation)를 고려하지 않았다. 어플리케이션에 따라서 데이터를 보낼 때 한번에 많은 메시지를 보내게 되면 IP 단편화가 발생하게 된다. 관찰 결과 Windows Media Technology 데이터 전송에 쓰이는 패킷 중에서 약 30~70%의 IP 단편화된 패킷들을 발견할 수 있었다. 이러한 패킷들은 그 자체만을 보고서는 포트 번호를 확인할 수가 없기 때문에 필터에서 인식을 하지 못하고 버려지게 되므로 멀티미디어 서비스 트래픽을 감지할 수가 없게 된다. 이와 더불어 수집하고자 하는 패킷만 잡아내도록 필터를 수시로 변경함에 기인한 문제가 발생한다. 새로운 세션이 생성되고 사라지게 됨에 따라 빈번하게 필터를 새로 컴파일해서 변경하는 동안에는 네트워크를 지나가는 패킷을 수집할 수가 없다. 이 지연 시간동안 새로 생겨나는 세션의 패킷이나, 다이나믹 세션 정보를 담고 있는 제어 세션 패킷이 지나갈 수 있다. 그리고 서비스가 끝났다고 판단되어 그와 관련된 패킷을 캡쳐하지 않도록 필터를 변경한 후에도 라우팅 상의 문제로 데이터 세션의 패킷이 지나갈 경우에는 그 패킷을 놓칠 수 있게 된다. 이러한 경우에도 역시 분석 결과가 부정확해진다.

3. 멀티미디어 서비스 트래픽 분석 방법

이 장에서는 플로우 기반 네트워크 모니터링 시스템에 적용되어 응용될 수 있는 멀티미디어 서비스 트래픽 모니터링 및 분석 방법을 제시한다. 이를 위해 패킷을 수집하고 처리하는 과정이 그림 3에 나타나 있다.

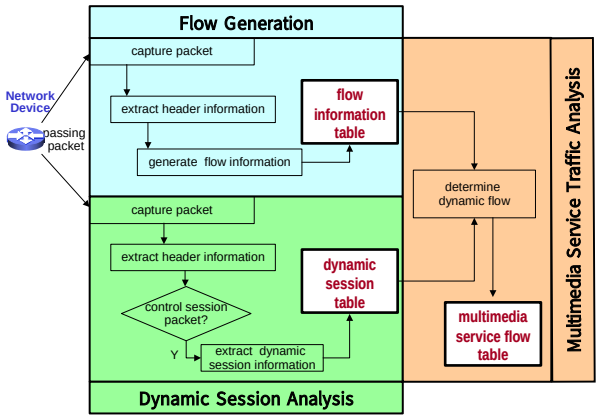

그림 3. 멀티미디어 서비스 트래픽 모니터링 및 분석 과정

전체적인 과정은 다음과 같이 크게 세 단계로 나누어 살펴볼 수 있다. 첫째는 패킷 헤더로부터 추출된 정보를 모아 플로우를 만드는 플로우 생성부분이다. 두 번째는 다이나믹 세션 정보 추출과정으

로 플로우 정보를 바탕으로 어플리케이션을 판단할 때 필요한 정보를 제공하기 위해 제어 세션의 패킷을 분석하는 부분이다. 그리고 세 번째는 이 두 정보를 바탕으로 멀티미디어 서비스 관련 플로우를 판별하는 멀티미디어 서비스 트래픽 분석 과정이다.

플로우를 생성하고 이에 기반해 트래픽을 분석하는 것은 일반적인 트래픽 모니터링 시스템과 유사하다. 그런데 다이나믹 세션 패킷들을 모아 플로우를 만들 경우에는 포트번호만을 가지고서는 그 어플리케이션을 알 수가 없다. 왜냐하면 다이나믹 세션은 잘 알려진 포트 번호를 사용하는 것이 아니라 동적으로 할당된 포트 번호를 사용하기 때문이다. 따라서 포트번호 만으로는 어플리케이션을 알 수 없는 플로우들이 과연 멀티미디어 서비스 트래픽인지를 판별하기 위한 정보가 필요하게 된다. 이 절에서는 그림 4의 다이나믹 세션 분석(dynamic session analyzer) 모듈에서 이러한 다이나믹 세션 정보를 찾아내는 과정에 대해서 기술한다.

```
1   Procedure DynamicSessionAnalyzer ( Msg )
2    BEGIN
3     if FIN Flag in Msg is NOT set then
4        if protocol in Msg = RTSP then
5           if SourcePort in Msg
                    = RTSP server port number then
6              result = ParseRTSP (payload of Msg ) ;
7        else if protocol in Msg = MMS then
8           if DestinationPort in Msg
                    = MMS server port number then
9              result = ParseMMS (payload of Msg ) ;
10       else if protocol in Msg =SIP then
11             result = ParseSIP (payload of Msg ) ;
12       else if protocol in Msg = Q.931 then
13         if SourcePort in Msg = Q.931 callee port then
14             result = ParseQ931 (payload of Msg ) ;
15       else if protocol in Msg = H.245 then
16             result = ParseH245 (payload of Msg ) ;
17       if result= TRUE then
18         create new data session information;
19         insert data session into dynamic session table;
20    else
21       delete session from dynamic session table;
22   END
```

그림 4. 다이나믹 세션 분석 알고리즘

그림 4에서 Msg 는 캡쳐된 패킷의 헤더와 트랜스포트 계층의 페이로드를 말한다. 본 논문에서는 이 정보를 합쳐 제어 세션 메시지라 칭한다. 이 메시지를 처리하는 과정을 보면, 우선 제어 세션 메시지에서 TCP FIN 플래그(flag)가 설정되어 있는지를 살핌으로써 세션 연결을 해제하는 패킷인지를 판단한다(줄 3). 연결 해제 패킷에는 다이나믹 세션 정보가 들어있지 않으므로, 그 이외의 패킷에 대해서만 페이로드를 분석한다. 이 때, 페이로드를 분석하는 일은 각 프로토콜별로 수행되며, 이 경우에도 다이나믹 세션에 대한 정보를 담고 있는 제어 세션 패킷만 골라서 분석함으로써 불필요한 분석 오버헤드를 줄일 수 있다. 이 때 선별되어지는 패킷을 살펴보면, RTSP 는 서비스 요청에 대한 서버의 응답(줄 5), MMS 는 클라이언트의 요청(줄 8), SIP 는 요청과 응답 양쪽(줄 10), H.323 의 Q.931 채널에서는 응답자의 메시지(줄 13)이다. 위에 열거한 경우에는 RTSP, MMS , SIP , Q.931 가 각각 554, 1755, 5060, 1720 으로 모두 잘 알려진 포트 번호를 사용하기 때문에 페이로드를 분석할 패킷과 프로토콜을 선별하기가 쉬워진다. 그러나 H.323 의 H.245 에서는 잘 알려진 포트 번호가 아니라 Q.931 프로토콜에서 동적으로 할당된 포트번호를 사용하게 된다. 따라서 캡쳐된 패킷의 패킷 헤더를 보고서 Q.931 에 의해 활성화된 H.245 의 제어 패킷인지를 확인하게 된다(줄 15).

이렇게 선정된 제 세션 패킷의 페이로드를 프로토콜별로 분석하게 된다. 각 제어 프로토콜의 스펙에 따라 페이로드를 파싱해가면서 다이나믹 세션에 대한 정보를 추출한다(줄 6, 9, 11, 14, 16). 표 3은 이 정보들이 있는 부분을 정규 표현식으로 표현한 것이다.

제어 프로토콜	정규 표현식
RTSP	[Tt][Rr][Aa][Nn][Ss][Pp][Oo][Rr][Tt]:[a-zA-Z]$^{+}$([/-][a-zA-Z])*.*;client_port=[1-9][0-9]$^{\{3,4\}}$(-[1-9][0-9]$^{\{3,4\}}$)$^{\{0,1\}}$;.*
MMS	\\\\[1-9][0-9]$^{\{1,2\}}$\.\\[1-9][0-9]$^{\{1,2\}}$\.\\[1-9][0-9]$^{\{1,2\}}$\.\\[1-9][0-9]$^{\{1,2\}}$\\[([Tt][Cc][Pp])|([Uu][Dd][Pp])]\\[1-9][0-9]$^{\{1,2\}}$[0-9][0-9].*
SIP	[Mm]= [a-zA-Z]+ [1-9][0-9]$^{\{3,4\}}$ [a-zA-Z]+([/-][a-zA-Z])*.*

표 3. 다이나믹 세션 정보의 정규 표현식

RTSP 의 경우에 페이로드에서 트랜스포트(transport) 헤더를 찾는다. 트랜스포트 헤더는 “ Transport: (전송프로토콜);(클라이언트포트);(서버포트) ” 형태로 되어 있으며, 이 헤더에서 규정된 정규 표현식을 갖는지를 확인해 다이나믹 세션에 대한 정보를 추출해 낸다. MMS 의 경우에도 ‘URL 스트링 형식’에 “TCP”나 “UDP”, 포트 번호 형태로부터 원하는 정보를 알아낼 수 있다. SIP 의 페이로드에는 세션에 대해 기술하는 SDP(Session Description Protocol)[17] 부분이 있는데, 여기서 M= (미디어 타입) (포트번호) (전송프로토콜) (페이로드 타입)” 형태로 되어 있는 미디어(media) 헤더를 찾아 데이터를 전송할 세션의 전송 프로토콜과 포트번호를 추출해 낸다. 반면 Q.931 이나 H.323 은 텍스트 기반의 프로토콜이 아니기 때문에 패킷에서 데이터를 전송할 때 새롭게 쓰여지는 포트 번호가 있는 특정 위치를 찾아야 한다. 그림 5는 다이나믹 세션 정보가 들어있을 때의 Q.931 과 H.245 패킷의 페이로드 중 일부

를 나타낸 것이다. Q.931 에서는 선택된 제어 메시지에서 요소 정보(element information) 중에서 사용자 간 정보(user-user info)를 찾아 IP 주소의 포트 번호를 읽어온다. H.245 세션에서 다이나믹 세션에 대한 정보를 갖고 있는 것은 논리 채널 개시(open logical channel) 메시지를 교환할 때이다. 따라서 이 메시지인지를 먼저 판별하고 순방향 채널 파라미터(forward channel parameter)나 역방향 채널 파라미터(reverse channel parameter), 혹은 네트워크 접근 파라미터(network access parameter)에서 새로운 세션에 대한 정보를 찾아내게 된다.

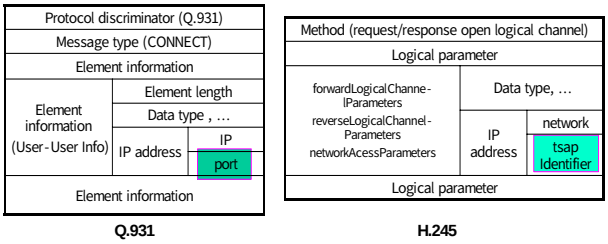

그림 5. Q.931 과 H.245 의 페이로드

분석 결과 다이나믹 세션에 대한 정보를 찾게 되면 이 정보들을 현재 활성화되어 있는 다이나믹 세션 정보를 갖고 있는 테이블에 저장한다. 그러나 세션이 끝났을 경우에는 이 정보는 필요 없는 정보가 되므로 테이블에서 제거되어야 한다. 따라서 처음 TCP 플래그 확인(그림 4 의 줄 3) 결과 세션을 종료하는 제어 세션 메시지이면(그림 4 의 줄 20) 다이나믹 세션 테이블에서 세션 정보를 삭제한다(그림 4 의 줄 21). 그러나 이러한 연결 해제 과정이 없거나 종료 패킷이 손실된 경우가 발생하게 되면 FIN 플래그를 확인할 수 없게 된다. 이런 점을 대비하기 위해 주기적으로 데이터 전송 세션 정보들을 체크해서 일정시간동안 패킷이 교환되지 않은 세션은 제거시킴으로써 세션 정보를 관리할 수 있다. 따라서 이 테이블의 내용을 참조함으로써 현재 활성중인 다이나믹 세션들에 대한 정보를 파악할 수 있고 이를 바탕으로 멀티미디어 서비스 트래픽을 판별할 수 있다.

4. 멀티미디어 서비스 트래픽 모니터링 및 분석을 위한 시스템 구조

이 장에서는 멀티미디어 서비스 트래픽 분석을 위한 플로우 기반의 시스템 구조에 대해서 기술한다. 본 논문에서는 이 시스템을 MST-MON(Multimedia Service Traffic MONistoring system)이라 명명하기로 한다. 이 시스템은 NG-MON 의 파이프 라인 구조와 로드 분산 기술을 그대로 이용하였다.

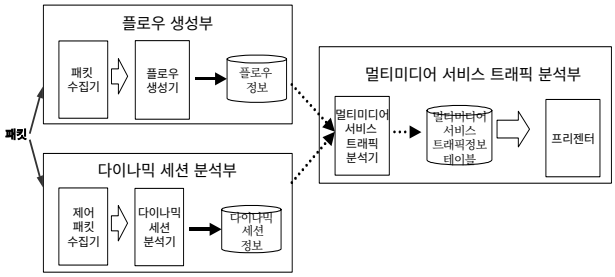

그림 6. MST-MON 시스템 구조

그림 6에서 보는 바와 같이 트래픽 모니터링 및 분석을 파이프 라인 구조로 연결된 여러 개의 단계로 나누었다. 각 분석 단계에서 로드 분산을 위해서 하나 혹은 그 이상의 시스템이 사용될 수 있다. 또한 각 단계마다 통신할 수 있는 인터페이스를 정의함으로써 다른 최적화된 모듈로 쉽게 대치할 수 있다. 또한 다이나믹 세션 분석부를 기존 모니터링 시스템에 병렬적으로 동작하게 하였다. 이것은 기존의 트래픽 모니터링 시스템에 수정을 가하지 않고 이 모듈을 추가하게 함으로써 유연성을 제공한다. 그리고 다이나믹 세션 정보 추출을 위해서는 패킷의 페이로드 전체를 캡쳐해서 분석해야 하는데, 기존 모니터링 시스템에서 패킷의 페이로드까지 모두 캡쳐하는 것은 부담이 된다. 따라서 별도로 제어 세션의 패킷을 캡쳐하는 부분을 따로 둠으로써 기존 시스템이 가질 수 있는 부담을 제거한다.

MST-MON 의 시스템 구조 및 기능은 다음과 같이 크게 세 부분으로 나뉘어진다. 플로우 생성부는 패킷 헤더로부터 추출된 정보를 모아 플로우를 만들고, 이와 동시에 동적 세션 분석부는 동적으로 생겨난 세션 플로우를 판단할 때 필요한 정보를 제공하기 위해 제어 세션의 패킷을 분석한다. 멀티미디어 서비스 트래픽 분석부는 이렇게 만들어진 이 두 종류의 정보를 바탕으로 멀티미디어 서비스 관련 플로우를 판별하고 분석하여 결과를 사용자에게 보여준다.

4.1 플로우 생성부(Flow Generation)

플로우 생성부는 플로우 생성을 위해 패킷 수집기(probe)와, 플로우 생성기(flow generator)로 이루어져 있다.

그림 7에서와 같이 패킷 수집기에서는 수집 필터가 네트워크 모니터링 지점을 지나는 패킷을 잡아내 패킷의 헤더만을 수집하고, 헤더 정보 추출기가 패킷 헤더를 분석해 트래픽 분석에 필요한 패킷 헤더 정보를 추출하여 플로우 생성기에 넘겨준다. 플로우 생성기는 패킷 헤더 정보를 이용해 일정 시간 동안 같은 플로우에 속하는 패킷끼리 정보를 모으고 통합한다. 즉, 플로우를 결정짓는 5 개 항목이 같은 플로우가 있는지 살펴서, 만일 기존에 존재하는 플로우이면 해당 플로우에 패킷이 추가될 때마다 그 플로우의 패킷 수와 플로우 길이를 갱신시킨다. 그러나 이 패킷이 속하는 플로우가 없다면 수집된 패킷으로 새로운 플로우를 만든다. 일정 시간이

지나면 모여진 플로우 정보들을 데이터베이스나 파일 형태로 저장하고, 다시 새롭게 플로우 정보를 만드는 과정을 반복한다. 그림 7의 플로우 플로우 정보가 이 때 저장되는 정보 형태이다.

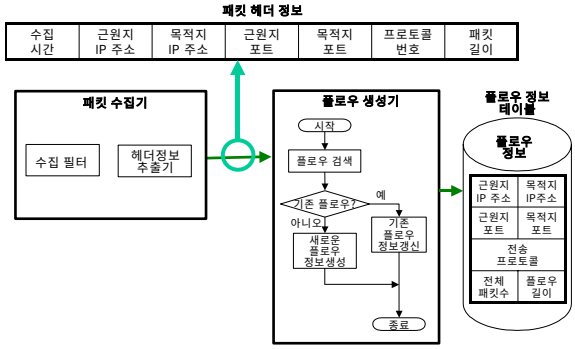

그림 7. 플로우 생성

본 논문에서 제시된 시스템 구조는 기존의 모니터링 시스템에서 만들어낸 플로우 정보를 그대로 이용할 수 있도록 하고 있다. 특히 MST-MON 의 경우에는 NG-MON 의 플로우 생성에 필요한 모듈들을 수정없이 그대로 이용하였으며, 이 때 단편화된 IP 패킷들까지 고려해서 플로우 정보를 만들기 때문에, IP 단편화로 인해 플로우의 포트 번호를 판단할 수 없게 되는 문제점을 해결하게 되었다.

4.2 다이나믹 세션 분석부(Dynamic Session Analysis)

MST-MON 은 멀티미디어 서비스 트래픽인지를 알 수 있게 해 주는 다이나믹 세션 정보를 제공해 줄 수 있다. 이를 위해 다이나믹 세션 분석부에서 제어 세션의 패킷들을 분석해서 다이나믹 세션의 정보를 찾아낸다.

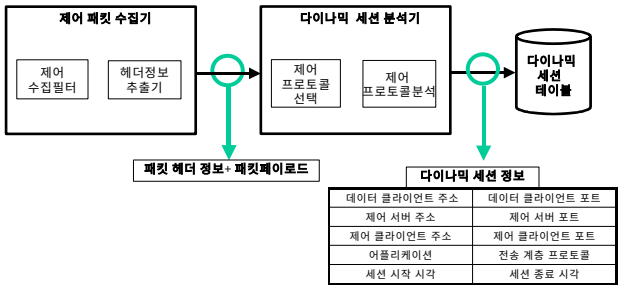

그림 8. 다이나믹 세션 분석

그림 8와 같이, 제어 패킷 수집기에서 제어 패킷 필터는 모니터링 지점을 지나가는 패킷을 헤더를 포함해서 페이로드까지 잡아내고, 헤더 정보 추출기가 분석에 필요한 정보를 추출해서 페이로드와 함께 동적 세션 분석기로 넘겨준다. 동적 세션 분석기는 패킷 헤더 정보를 보고 제어 세션의 패킷을 골라내서 그림 4의 분석 알고리즘에 따라 페이로드로부터 다이나믹 세션이 사용할 포트번호 등에 대한 정보를 추출해 낸다. 분석 결과 얻어진 정보들은 다이나믹 세션 테이블에 저장되어, 트래픽 분석기가 참조할 수 있도록 한다. 이 때 테이블에 기록되는 정보 항목이 그림 8에 나타나 있다. 데이터 클라이언트 주소와 데이트 클라이언트 포트는 다이나믹 세션에서 새롭게 만들어서 데이터를 전송 받을 주소와 포트번호이다. 제어 서버 주소와 포트는 해당 다이나믹 세션을 생성해 낸 서버의 IP 주소와 포트 번호이고, 제어 클라이언트 주소와 포트도 다이나믹 세션을 생성할 때 서버와 연결되었던 클라이언트의 주소와 포트 번호이다. 이 정보를 이용해 다이나믹 세션과 관련있는 제어 세션을 찾아낼 수 있다. 그리고 어플리케이션은 해당 세션을 이용하고 있는 멀티미디어 서비스의 종류를 나타낸다. 세션 시작 시간은 다이나믹 세션을 발견해서 해당 정보를 만든 시각이며, 세션 종료 시간은 제어 세션의 연결이 끊어짐으로 해서 다이나믹 세션이 종료된 시각이다.

4.3 멀티미디어 서비스 트래픽 분석부 (Multimedia Service Traffic Analysis)

멀티미디어 서비스 트래픽 분석부는 멀티미디어 서비스 트래픽들을 판별하고 저장하는 역할을 담당한다.

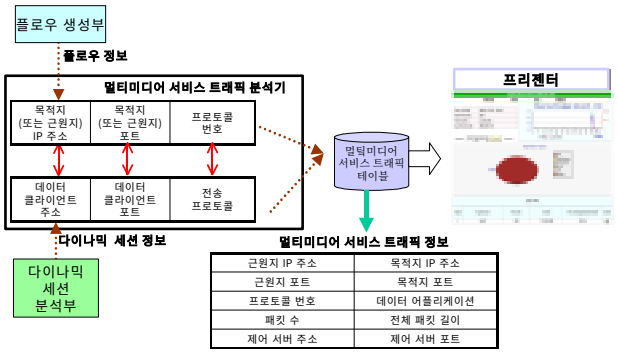

그림 9. 멀티미디어 서비스 트래픽 분석기와 프리젠터

그림 9에 도시된 바와 같이, 멀티미디어 서비스 트래픽 분석기는 플로우 생성부로부터 가져온 플로우 정보를 동적 세션 분석부으로부터 받은 동적 세션 정보와 비교한다. 동적 세션 정보의 [데이터 클라이언트 주소, 데이터 클라이언트 포트, 전송 프로토콜]의 쌍과 저장된 플로우 정보의 [목적지 IP 주소, 목적지 포트 번호, 프로토콜 번호]의 쌍이나 [근원지 IP 주소, 근원지 포트, 프로토콜 번호]의 쌍과 동일한지를 살핀다. 즉, 데이터 클라이언트 주소와 목적지 IP 주소, 데이트 클라이언트 포트와 목적지 포트 번호, 전송 프로토콜과 프로토콜 번호가 모두 같은지 살피는 것이다. 멀티미디어 서비스 제어 세션에서 사용하는 잘 알려진 포트를 사용하는 플로우들과의 비교 결과 항목이 일치하는 플로우들은 그림 9와 같은 정보를 가지고 저장되어진다. 각 항목은 패킷 헤더 정보와 다이나믹 세션 정보를 종합한 것이다. 저장된 정보는 프리젠터에 의해 웹을 통해 보여지게 된다.

그런데, 멀티미디어 서비스를 위한 세션들이 빈

번하게 발생하고 사라지게 됨에 따라서 한번 저장된 다이나믹 세션 정보가 계속해서 유효하지는 않는다. 즉 특정 시간 동안에 클라이언트에서 멀티미디어 서비스 데이터를 받기 위해 사용되었던 포트가 시간이 지나고 나서는 연결이 끊기고, 다른 어플리케이션에 의해 사용될 수도 있기 때문이다. 따라서 현재 활성화되어 있는 다이나믹 세션 정보인지를 확인하는 작업이 필요하다.

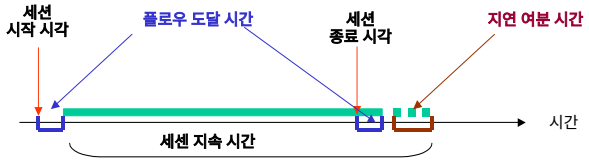

그림 10. 세션 지속 시간

그림 10은 세션 지속 시간의 유효성을 검사하기 위해 필요한 시간들을 보여주고 있다. 세션 시작 시간과 세션 종료 시각은 다이나믹 세션 정보로부터 온 정보들이다. 플로우 도달 시간은 플로우 정보와 다이나믹 세션 정보와의 동기화를 위해 필요한 시간이다. 즉, 다이나믹 세션 정보가 생성 즉시 멀티미디어 세션 분석부에 즉각 전달되어 저장되는 반면, 플로우 정보는 일정시간 동안 패킷들을 모아 플로우로 만든 후에야 트래픽 분석기로 도달하기 때문에, 다이나믹 세션도 이 시간만큼이 지난 후에야 플로우를 판별할 때 사용되어진다. 그리고, 제어 세션의 연결이 끊어져서 다이나믹 세션이 종료되어도 이 플로우 도달 시간만큼이 지날때까지는 다이나믹 세션이 유효하다고 보게 된다. 지연 여부 시간은 라우팅 경로 등의 문제로 제어 세션이 종료되고 난 후에 도착하는 데이터 세션의 패킷들을 놓치지 않도록 여분의 지연 시간을 두도록 한 것이다.

5. 시스템 구현

이 장에서는 멀티미디어 서비스 트래픽 모니터링 시스템인 MST-MON 의 프로토타입과 이를 이용해 얻은 결과에 대해서 기술한다. 구현 환경은 그림 11와 같다.

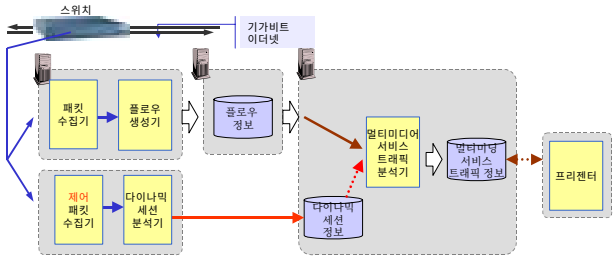

그림 11. 프로토타입 구현 환경

동일한 트래픽이 스위치의 미러링 포트를 통해 두 개의 패킷 수집기와 제어 패킷 수집기로 전달되지만, 각 모듈의 기능에 따라서 달리 처리된다. 그 중 패킷 수집기는 네트워크 인터페이스 카드로부터 들어오는 모든 패킷을 캡처하여 1 분간 패킷들을 모아 플로우를 생성한 후 저장한다. 이 때 동일한 패킷들이 제어 패킷 수집기로 넘겨지지만, 다이나믹 세션 정보를 위해 필요한 제어 세션 패킷들만을 선별되어 분석되어진다. 만일 선택된 페이로드로부터 다이나믹 세션의 포트 정보 등을 찾게 되면 멀티미디어 서비스 트래픽 분석기에 있는 다이나믹 세션 테이블에 저장시킨다.

멀티미디어 서비스 트래픽 분석기는 매 분마다 플로우 저장기에 쿼리를 보내 플로우 정보를 가져온다. 그 중에서 다이나믹 세션 테이블의 정보를 바탕으로 멀티미디어 서비스 트래픽 플로우를 판별해 데이터베이스에 저장시킨다. 이후 분석된 결과는 사용자의 요구에 의해 프리젠터를 통해 제공되어진다.

MST-MON 은 800MHz CPU, 256Mbytes 메모리 사양의 PC 를 Linux Redhat 7.2 의 운영체제하에서 사용하였다. 이 때 각 시스템간에는 플로우를 생성하고 판별하는데 있어서 시간 동기화가 필요하다. 이를 위해 LAN(Local Area Network) 환경에서 밀리 초 단위의 동기화를 제공하는 NTP(Network Time Protocol)[18]를 사용하였다.

이 프로토타입을 적용함으로써 기존에는 어플리케이션을 알 수 없었던 멀티미디어 서비스 트래픽들을 판별해낼 수 있었다.

그림 12. 멀티미디어 서비스 트래픽 분류

그림 12는 MST-MON 을 이용해 특정 1 분동안 발생한 트래픽 중에서 멀티미디어 서비스 트래픽만을 골라 어플리케이션별로 나타낸 예이다. 왼쪽 상단 표에서 트래픽 발생 시간, 전체 패킷 수, 전체 트래픽 양, 평균 이용 대역폭을 알 수 있다. 오른쪽 그래프에서는 선택된 시간 단위별로 전체 발생 트래픽의 변화를 보여준다. 예를 들어 분 단위로 분석 시간을 선택했을 경우에 한 시간 동안 각 분단위로 발생한 트래픽 양을 그래프로 보여주게 되어 있다. 중간의 원형 그래프는 멀티미디어 서비스 종류별로 발생한 트래픽 비율을 한 눈에 볼 수 있도록 나타낸 것이다. 하단의 표는 해당 시간에 발생한 각 멀

티미디어 서비스 종류 별 트래픽 정보를 트래픽 양이 순서에 따라서 보여준다. 각 서비스 종류 별로 발생한 트래픽의 패킷 수, 바이트 및 전체에서의 비율이 나타나 있다. 해당 멀티미디어 서비스에 대한 정보를 원할 경우에는 <detail> 버튼을 더욱 상세한 분석 결과를 볼 수 있다.

그림 13. WMT 트래픽 분석 예

그림 13은 그림 12에서 WMT 를 선택했을 경우 나타나는 1 분 단위의 트래픽 분석 예이다. 화면 하단에는 선택된 서비스를 이루고 있는 여러 개의 플로우 정보를 보여준다. 이러한 플로우들은 기존에는 개별적으로 분류되고 분석되었다. 그러나 MST-MON 은 특정 멀티미디어 서비스를 위해 발생한 여러 개의 플로우들을 연관시키고 종합해서 분석할 수 있다. 이 예에서는 danube.posech.ac.kr 와 mail.ex.onkino.co.kr 사이에 WMT 서비스 이용을 위해 발생한 플로우들을 종합 분석한 트래픽 정보가 나타나 있다. 특히 거의 대부분의 양을 차지하고 있는 데이터 세션은 잘 알려진 포트 번호를 사용하고 있지는 않음에도 불구하고, 멀티미디어 세션인지를 파악하고 다른 제어 세션과 연관시켜서 정보를 보여주고 있다. 이로써 멀티미디어 서비스 트래픽을 파악하고 분석할 수 있게 되었다.

6. 결론

본 논문에서는 플로우 기반 네트워크 모니터링 시스템에 적용될 수 있는 멀티미디어 서비스 트래픽 모니터링 및 분석 방법을 제시하였다. 특히 멀티미디어 서비스를 위한 제어 프로토콜이 메시지를 교환하는 과정을 상세히 분석함으로써, 각 제어 프로토콜 별로 제어 세션 패킷의 페이로드로부터 다이나믹 세션 정보를 추출하는 방법을 찾아내었다. 이렇게 얻어진 정보를 바탕으로 멀티미디어 서비스 도중 생성되는 다이나믹 세션을 파악함으로써, 기존에 포트번호만을 가지고 판단했을 경우에는 어플리케이션을 알 수 없었던 멀티미디어 서비스 트래픽들을 분석해 낼 수 있게 되었다. 또한 동일한 멀티미디어 서비스를 위해 발생한 여러 개의 제어 및 데이터 세션들을 하나로 연관시킬 수 있게 됨으로써 어플리케이션 관점에서 분석을 용이하게 만들었다.

본 논문에서는 기존 NG-MON 에 위에서 분석 방법을 적용하여 MST-MON 을 개발하였다. 이 프로토타입은 1Gbps 네트워크 환경에서 멀티미디어 서비스 트래픽을 모니터링했다. 그 결과 기존에는 알 수 없는 어플리케이션으로 분류되었던 멀티미디어 서비스 관련 트래픽을 파악하고, 어플리케이션에 근간한 분석을 함으로써 멀티미디어 사용 실태를 파악할 수 있는 기반을 제공하였다.

이러한 방법은 네트워크 트래픽 분석에 있어서 기존에 TCP 나 UDP 의 포트 번호만을 보고 어플리케이션을 추정하던 분석에서 벗어났다는 점에서 의의를 찾을 수 있다. 실제로 어플리케이션의 세션 레벨에서 메시지를 분석하고 세션 정보를 유지함으로써 기존 트랜스포트 계층의 포트 번호만을 보고서는 파악할 수 없었던 트래픽을 분석할 수 있게 되었다. 따라서 트랜스포트 계층에서 어플리케이션 계층으로 분석 수준을 높였다고 볼 수 있다.

향후 연구로 다이나믹 세션 정보 추출의 정확성을 보완하는 노력을 계속하고자 한다. 현재 본 논문에서는 MMS 패킷의 페이로드에서도 다이나믹 세션 정보를 추출해서 이용하고 있다. 그러나, MMS 프로토콜의 스펙이 비공개되어 있기 때문에 이에 대한 정확성을 입증하기 위해서는 프로토콜 스펙을 구해서 구현하고 이를 검증하는 연구가 필요할 것이다.

또한, 다이나믹 세션을 생성하고 이용하는 또 다른 어플리케이션에 본 논문에 제시된 방법을 더욱 확장시키는 연구가 필요하다. 이러한 적용 범위의 확장을 통해 기존에 포트 번호만으로는 판별이 불가능한 트래픽을 파악하고 분석할 수 있게 될 것이다.

7. 참고 문헌

[1] J. Won-Ki Hong, S. U. Park, Y. M. Kang and J. T. Park, “Enterprise Network Traffic Monitoring, Analysis and Reporting Using Web Technology”, Journal of Network and Systems Management, Plenum Press, March 2001, pp. 89-111.

[2] J. Won-Ki Hong, Soon-Sun Kwon and Jae-Young Kim, “WebTrafMon: Web-based Internet/Intranet Network Traffic Monitoring and Analysis System”, Computer Communications, Elsevier Science, Vol. 22, No. 14, September 1999, pp. 1333-1342.

[3] Soon-Hwa Hong, Jae-Young Kim, Bum-Rae Cho, J. Won-Ki Hong, “Distributed Network Traffic Monitoring and Analysis using Load Balancing Technology”, 2001 Asia-Pacific Network Operations and Management Symposium, Sydney, Australia, September 2001, pp.172-183.

[4] Jacobus van der Merwe, Ramon Caceres, Yang-hua Chu, and Cormac Sreenan “mmdump- A Tool for Monitoring Internet Multimedia Traffic,” ACM Computer Communication Review, 30(4), October 2000.

[5] Micorsoft, Windows Media Technology, http://www.microsoft.com/windows/windowsmedia/default.asp.

[6] Real Networks, Real Media Technology, http://www.realnetworks.com/.

[7] Apple, QuickTime,

http://www.apple.com/quicktime/.
[8] M. Handley, H. Schulzrinne, E.Schooler, J. Rosenberg, “SIP: Session Initiation Protocol,”RFC 2543, March 1999.
[9] ITU-T, “Recommendation H.323: Visual Telephone Systems and Equipment for Local Area Networks Which Provide a Non-guaranteed Quality of Service,” 1996.
[10] H. Schulzrinne, A. Rao, and R. Lanphier, “Real Time Streaming Protocol (RTSP),” RFC 2336, April 1998.
[11] H. Schulzrinne, S. Casner, R. Frederick, V, and Jacobson, “RTP: A Transport Protocol for Real-Time Applications,” RFC1889, January 1996.
[12] Se-Hee Han, Myung-Sup Kim, Hong-Taek Ju and James W. Hong, “The Architecture of NG-MON: A Passive Network Monitoring System”, Lecture Notes in Computer Science 2506, 13th IFIP/IEEE International Workshop on Distributed Systems: Operations and Management (DSOM 2002), Montreal, Canada, October, 2002, pp. 16-27.
[13] Siegfried Lifler, “Using Flows for Analysis and Measurement of Internet Traffic,” Diploma Thesis, Institute of Communication Networks and Computer Engineering, University of Stuttgart, 1997.
[14] J.Quittek, T. Zseby, B. Claise, K.C. Norsth, “IPFIX Requirements,” Internet Draft, http://norseth.org/ietf/ipfix/draft-ietf-ipfix-architecture-00.txt.
[15] CAIDA, "Preliminary Measurement Spec for Internet Routers," http://www.caida.org/tools/measurement/measurementspec/.
[16] Dave Plonka, “FlowScan: A Network Traffic Flow Reporting and Visualization Tool,” Proc. of 2000 LISA XIV, New Orleans, USA, December 2000, pp. 305-317.
[17] M. Handley, V. Jacobson, “SDP: Session Description Protocol,” RFC 2327, April,1998.
[18] NTP, http://www.ntp.org/.