

로드 분산 방법을 이용한 대용량 네트워크 트래픽 모니터링 및 분석

홍순화, 김재영, 홍원기
{padosori, jay, jwkhong}@postech.ac.kr

요 약

최근 인터넷 사용자가 계속해서 증가하고 네트워크 기반의 응용 프로그램이 다양하게 개발됨에 따라 네트워크 트래픽이 증가하고 있다. 이에 따라 네트워크 트래픽을 분석하고 모니터링할 수 있는 네트워크 분석 시스템들이 소개되고 있다. 그러나 기존의 네트워크 분석 시스템들은 실시간 혹은 매 시간 단위의 정보만을 확인할 수 있어 장기간의 네트워크 트래픽 모니터링에는 적합하지 않다. 또, 하나의 시스템에서 패킷 캡처와 패킷 분석을 동시에 하므로 시스템 과부하 시 패킷 손실이 일어나 10 Mbps 이상의 대용량 인터넷 회선에 적당치 않고 여러 네트워크 노드의 트래픽을 합쳐서 분석할 수 없는 단점이 있다. 이뿐 아니라 기존 네트워크 분석 시스템은 P2P, 스트리밍 등의 멀티미디어 트래픽을 분석하지 못한다. 본 논문에서는 이러한 단점을 보완한 네트워크 모니터링 및 분석 시스템인 WebTrafMon II의 설계 및 구현을 설명한다. WebTrafMon II는 웹기반에서 호스트 정보, 프로토콜 정보, 어플리케이션 정보를 하루 동안 매 시간 별로 분석해 주는 WebTrafMon의 장점을 살리면서 데이터베이스를 이용하여 장기간의 네트워크 트래픽 정보를 저장한다. 또, 멀티미디어 트래픽의 유형을 파악하여 멀티미디어 트래픽 방법을 연구하고 로드 분산 방법을 사용하여 패킷 캡처, 패킷 분석, 웹 뷰어 등의 모듈을 분산 시스템에서 독립적으로 실행하여 시스템 과부하로 인한 패킷 손실을 막고 패킷 캡처를 독립적으로 실행하여 여러 네트워크 노드의 트래픽을 합쳐서 분석할 수 있다.

1. 서론

오늘날 네트워크 트래픽은 계속해서 증가하고 있다. 이러한 네트워크 트래픽의 증가로 인해 네트워크 회선의 부족이나 네트워크 응답 시간의 저하 등의 문제가 발생할 수 있으며 이를 대비하기 위해 네트워크 모니터링 및 분석 시스템의 중요성이 부각되고 있다.

여러 네트워크 분석 시스템이 개발되어 사용되고 있으나 단순한 실시간 정보로 인해 네트워크 모니터링에 적합하지 못하거나 장기간의 네트워크 계획을 세우는데 적합하지 않다. 예를 들어, tcpdump [1]는 패킷 한 개의 정보를 실시간으로 나열하고 있어 네트워크 모니터링에 적합하지 않으며 ntop [2]은 네트워크 트래픽 정보를 하루 동안 매 시간 별로

모아 분석하므로 하루 동안의 모니터링에는 적합하나 하루 이상의 장기간 모니터링에는 적합하지 않다.

이뿐 아니라, 기존의 네트워크 분석 시스템들은 최근에 급격히 증가하고 있는 Peer-to-Peer(P2P) [3], 스트리밍 서비스 등의 멀티미디어 트래픽을 분석할 수 없다.

또, 기존의 네트워크 분석 시스템들은 한 시스템에서 패킷 캡처와 패킷 분석을 동시에 하므로 시스템 과부하 시 패킷 손실이 발생하며 한 시스템에서만 패킷 캡처가 가능하도록 설계되어 여러 네트워크 노드의 트래픽을 합쳐서 분석할 수 없다.

이에 따라 본 논문에서는 기존의 네트워크 분석 시스템들의 장단점을 파악하여 네트워크 트래픽을 사용자에게 정확하고 편리하게

분석해 줄 수 있는 네트워크 모니터링 및 분석 시스템의 요구사항을 살펴보고 그러한 요구사항을 반영한 WebTrafMon II의 설계 및 구현을 설명한다.

WebTrafMon II의 기반이 되는 WebTrafMon [4]은 웹기반에서 하루 동안 매 시간 별로 호스트 정보, 프로토콜 정보, 어플리케이션 정보 등을 분석해 주는 장점을 갖는 네트워크 분석 시스템이다. WebTrafMon도 기존의 네트워크 분석 시스템들처럼 장기간의 네트워크 모니터링이 지원되지 않는 단점과 한 시스템에서 패킷 캡처와 패킷 분석을 동시에 하여 시스템 과부하 시 패킷 손실이 발생하고 여러 네트워크 노드의 트래픽을 합쳐서 분석할 수 없는 단점이 있다.

WebTrafMon II는 WebTrafMon의 장점을 살리면서 단점을 극복하기 위해 데이터베이스를 사용하여 장기간의 네트워크 트래픽 정보를 저장하고 멀티미디어 트래픽을 분석하며 분산 시스템에서의 로드 분산 방법을 사용하여 패킷 캡처와 패킷 분석을 독립시켜 시스템 과부하로 인한 패킷 손실을 막았다. 또, 패킷 캡처를 독립적으로 설계해 여러 네트워크 노드의 트래픽을 합쳐서 분석할 수 있다.

2. 관련연구

2.1 네트워크 분석 시스템 조사

여러 네트워크 분석 시스템들이 사용되고 있다. 각 네트워크 분석 시스템들의 특징을 살펴보고 장단점을 파악한다.

2.1.1 Ntop

ntop [2]은 Deri Luca가 1998년부터 오픈 프로젝트로 개발하고 있는 네트워크 모니터링 및 분석 시스템이다. ntop은 'network top'이라는 뜻으로 유닉스의 시스템 자원 상태를 보여주는 top 명령어를 네트워크 모니터링 및 분석 시스템에 응용했다는 뜻이다. ntop은 웹기반에서 호스트 정보, 프로토콜 정보, 어플리케이션 정보 등을 분석해 준다. 호스트 중심으로 호스트에서 나가고 들어오는 패킷의 양이라든가 호스트에서 많이 사용되는 프로토콜을 확인하기 편리하다. ntop은 시스템 과부하로 인한 패킷 손실을 막기 위해 패킷 분석 모듈에 쓰레드를 사용하는 등 시스템 자원을 적게 사용하도록 만들었다. 현재 ntop은 하루 24시간을 기준으로 해서 매 시간 별 네

트워크 트래픽을 분석해 주는 기능이 있다. 호스트 중심의 분석 기능은 조사한 네트워크 분석 시스템들 중 가장 뛰어났으나 장기간 네트워크 트래픽 분석을 제공하지 못하는 단점이 있다. 또 포항공대 인터넷 회선에서 테스트해 본 결과 60 Mbps의 트래픽을 패킷 손실로 인해 20 Mbps로 분석하고 2만개의 호스트를 1000개 정도밖에는 표시해 주지 못했다.

2.1.2 MRTG

MRTG(Multi-Router Traffic Grapher) [5]는 네트워크 링크 간의 트래픽 부하량을 측정하는 도구로서 5분 단위에서 1년 단위까지 네트워크 트래픽의 총 양을 확인하는 도구이다. 웹기반으로 동작하여 네트워크 트래픽을 확인하기에 편리할 뿐 만 아니라 C와 Perl로 작성되어 유닉스 플랫폼 뿐 만 아니라 윈도우즈 NT에서도 동작하는 장점이 있다. 또 MRTG는 SNMP MIB 정보를 사용하여 패킷 캡처를 하지 않으므로 패킷 손실 없이 정확한 정보를 제공해 준다. 그러나 MRTG는 네트워크 트래픽의 총 양에 대한 정보를 제공해 줄 뿐 어떤 호스트에서 어느 정도의 트래픽을 발생시켰는지, 어떤 어플리케이션이 어느 정도의 트래픽을 발생시켰는지, 어떤 프로토콜이 사용되었는지 등의 정보를 제공해 주지 못하는 단점이 있다. 이러한 단점으로 인해 네트워크의 문제점을 파악할 수 있는 네트워크 분석 시스템으로는 적당하지 않다.

2.1.3 Ethereal

Ethereal [6]은 실시간 혹은 파일에 저장해 놓은 네트워크 트래픽 정보를 분석해 주는 프로그램이다. GTK+ 기반으로 유닉스 플랫폼의 X 윈도우 화면에서 동작하며 MS 윈도우즈용도 있다. 여러 다양한 프로토콜 분석을 지원하는 장점이 있다. Gerald Comb가 50명 이상의 프로그래머들과 함께 오픈 프로젝트로 진행되고 있다. Ethereal은 보안 등을 위한 실시간 네트워크 모니터링에 적합한 프로그램으로 장기간의 네트워크 모니터링에는 적합하지 않다.

2.1.4 NNStat

NNStat [7]은 Robert Braden과 Annette DeSchon이 1988년에 발표한 프로그램으로 여러 네트워크 노드에서의 트래픽을 합쳐서 분석할 수 있으며 사용자가 지정하는 장기간의

네트워크 트래픽 분석도 지원한다. NNStat은 SAA(Statistics Acquisition Agent)와 SCH(Statistics Collection Host)로 구성되어 있다. SAA는 여러 네트워크 노드에서 네트워크 트래픽을 수집하는 모듈이고 SCH는 분석하는 모듈이다. 각 노드에 SAA가 하나씩 있게 되고 각각의 SAA에서 모은 데이터를 SCH에 보내준다. SAA에는 분석 시간을 설정하는 기능이 있어 사용자가 분석 시간을 설정할 수 있다. NNStat은 SunOS 4.0 NIT(Network Interface Tap) 인터페이스를 사용하는 시스템에만 지원되는 단점이 있다.

2.1.5 UniMon

OSI 7 계층의 프로토콜을 정확히 분석하는 것을 목표로 갖는 UniMon [8]은 NNStat처럼 여러 네트워크 노드의 트래픽을 합쳐서 분석할 수 있는 장점이 있다. UniMon은 실시간 네트워크 모니터링과 문제점 파악에 적당한 분석 시스템으로 장기간의 네트워크 트래픽 분석에는 적당하지 않다.

2.1.6 기타

tcpslice [9]는 실시간 네트워크 분석 도구로 다양한 포맷으로 네트워크 트래픽 정보를 파일에 저장할 수 있는 특징을 가지고 있다. snoop [10]은 Etherfind의 새로운 버전으로 SunOS 5.x에 기본적인 패키지로 들어가 있다. snoop은 tcpdump와 기능이 비슷하며 역시 실시간 분석 도구이다. 그외 argus [11], arpwatc [12], nsfwatc [13], drawbridge [14] 등은 네트워크 보안을 목적으로 하기에 약간의 네트워크 분석 기능을 지원하나 네트워크 모니터링에는 불편하다. 또 MS 윈도우즈 전용 프로그램으로 ewatch [15]와 sniffer pro [16]같은 프로그램이 있다. ewatch와 sniffer pro는 상용 프로그램으로 실시간 분석을 위주로 편리한 UI를 제공해 주는 것이 장점이거나 장기간의 네트워크 트래픽을 분석할 수 없다.

2.1.7 프로그램 비교 사항

위에서 소개한 내용을 정리해 보면 다음 표 1과 같다.

(표 1) 네트워크 분석 시스템 비교

	Capture Method	Analysis Method	Analysis Interval	Analysis Scope	Support for Multiple network points	Distributed Architecture	User Interface
tcpdump	libpcap	real time	current	layer 7	no	no	Text
ntop	libpcap	batch	5 second, hourly	layer 7	no	no	Web
etherreal	libpcap	real time, batch	current, user-specified time	layer 7	no	no	X- Windows
MRTG	snmp agent	batch	5 minute, hourly, daily, weekly, monthly	layer 2	yes	yes	Web
WebTrafMon	libpcap	real time, batch	current, hourly	layer 7	no	no	Web
WebTrafMon II	libpcap	real time, batch	current, hourly, daily, monthly, yearly	layer 7	yes	yes	Web

표 1에서 ‘Capture Method’는 패킷을 캡처하는 방식을 말한다. libpcap으로 ‘Analysis Scope’을 응용 계층(layer 7)까지 분석할 수 있는 방법과 SNMP를 사용해서 링크 계층(layer 2)까지 분석할 수 있는 방법이 있다. ‘Analysis Method’는 ‘real time’과 ‘batch’로 나누었다. real time은 패킷 하나가 들어올 때마다 정보를 보여 주는 방식을 말하며 batch는 일정한 시간 동안의 트래픽을 수집해서 분석한 뒤 보여 주는 방식을 말한다. ‘Analysis Interval’은 분석하여 보여 줄 수 있는 시간 범위를 말하며 ‘Support for Multiple Network points’는 여러 네트워크 노드의 트래픽을 합쳐서 분석해 줄 수 있는지의 여부이다. 또, ‘Distributed Architecture’는 분석 시스템의 로드를 분산할 수 있는지의 여부이다.

표 1에서 보면 분석 범위가 장기간이면서 응용 계층까지를 분석해 주는 프로그램이 없다는 것을 발견할 수 있다. 일정한 간격의 장기간 네트워크 트래픽 정보를 제공하는 것은 MRTG밖에 없는데 MRTG는 호스트 정보 혹은 프로토콜 정보를 제공하지 못한다. 따라서 장기간의 네트워크 트래픽 정보를 제공해 주면서 호스트 정보 혹은 프로토콜 정보를 분석해 줄 수 있는 프로그램이 필요하여 WebTrafMon II는 분석 방식을 batch, 분석 범위는 매시, 매일, 매달, 매년으로 정했다. 또 여러 노드에서의 패킷 캡처를 지원하는 프로그램이 NNStat과 UniMon밖에 없기에 여러 네트워크 노드에서의 패킷 캡처를 지원하도록 정하였다.

2.2 멀티미디어 서비스 포트 번호 조사

멀티미디어 서비스를 분석하기 위해 Real

Server와 Real Player에서 오고 가는 패킷을 조사하여 분석하였다. 조사해 본 결과 클라이언트 포트가 6972 번으로 고정되어 있었다. 멀티미디어 서비스도 telnet, ftp 등과 같이 TCP 혹은 UDP 위에서 동작하는 응용 서비스이므로 포트 번호 조사를 통한 멀티미디어 분석을 시도한다. 스트리밍 서비스와 P2P 도구들의 포트 번호에 대해 조사한다. Real Player [16]의 RTP [17], Windows Media Player [18]의 MMST [19], FastTrack [20], Gnutella [21], FreeNet [22]에 대해 조사했다.

(표 2) 멀티미디어 서비스의 포트 번호 조사

서비스	프로토콜	포트 번호
Real Media	RTP (UDP)	클라이언트 6972
Windows Media	MMST (TCP)	서버 1751
FastTrack	fasttrack (TCP)	서버 1214
Gnutella	gnutella (TCP)	6347
FreeNet	freenet (TCP)	19114

2.2절에서 조사한 스트리밍 도구와 P2P 도구의 포트 번호를 네트워크 분석 시스템에서 처리하면 멀티미디어 트래픽을 분석할 수 있다.

3. 요구사항

네트워크 트래픽이 증가하는 원인으로는 첫째, 점점 더 많은 시스템이 네트워크에 연결되어 사용자가 증가하는 것과 둘째, 웹을 비롯한 다양한 네트워크 응용 프로그램의 개발 및 사용으로 인한 네트워크 사용량의 증가를 들 수 있다. 특히 웹의 발전은 네트워크를 사용하지 않던 사람들에게 네트워크를 접해 볼 기회를 제공했을 뿐 만 아니라 이미 네트워크를 사용하던 사람들의 네트워크 사용량도 크게 높이고 있다.

네트워크 트래픽을 분석하기 위해서는 네트워크 분석 시스템이 호스트 정보와 네트워크 트래픽 정보를 가지고 네트워크 트래픽의 주된 양을 어떤 호스트가 차지하는지, 사용되고 있는 호스트의 수는 어느 정도인지, 호스트가 주로 사용하고 있는 서비스는 무엇인지 등의 정보를 알려 줄 수 있어야 한다. 그러나

기존의 분석 시스템들은 텍스트 기반에서 실시간으로 호스트 정보, 프로토콜 정보, 트래픽 양 정보 등을 화면에 보여 주는 것에 그치고 있어서 어떤 호스트가 네트워크 트래픽의 대부분을 차지하는지, 사용되고 있는 호스트 수는 어느 정도인지 등의 문제를 파악하기 어렵다. 이러한 문제점을 생각해 볼 때 네트워크 모니터링 및 분석 시스템은 기본적으로 다음과 같은 요구사항을 만족해야 한다.

1) 호스트 정보를 분석할 수 있어야 한다. 특정한 시간 동안 어떤 호스트들이 네트워크 트래픽을 발생시켰고 그 중 가장 많은 트래픽을 발생시킨 호스트는 무엇인지 알 수 있어야 네트워크의 문제점을 파악할 수 있다.

2) 사용되고 있는 전송 프로토콜(예: arp, ip, udp, tcp) 및 어플리케이션 계층의 프로토콜(예: ftp, http, mms, rtp, snmp, telnet)을 분석할 수 있어야 한다. 프로토콜을 분석해야 어떤 서비스가 어느 정도 사용되는 지 알 수 있다.

3) 웹기반의 사용자 인터페이스를 제공해야 한다. 웹기반의 사용자 인터페이스를 통해 여러 사람이 언제 어디서나 정보를 쉽게 확인할 수 있다.

요구사항 1), 2), 3)을 만족하는 네트워크 분석 시스템으로는 ntop과 WebTrafMon이 있다.

ntop은 웹기반에서 호스트를 기준으로 해서 각 호스트들이 어떤 프로토콜을 사용하는지 분석해 주는 기능이 있고 WebTrafMon도 웹기반에서 시간 별로 호스트 정보와 프로토콜 정보를 제공하므로 둘 다 요구사항 1), 2), 3)을 만족한다.

그런데 ntop이나 WebTrafMon으로는 장기간의 네트워크 트래픽을 분석할 수 없다. 왜냐하면 ntop이나 WebTrafMon은 하루 동안 매 시간 별 네트워크 트래픽 정보만을 제공하기에 24시간이 지나가면 분석해 놓은 데이터가 없어지기 때문이다. 장기적인 네트워크 계획을 세우기 위해서는 매 시간 별 데이터 뿐만 아니라 하루 단위, 한달 단위, 일년 단위 등으로 네트워크 트래픽을 분석해 줄 수 있어야 한다.

장기적인 네트워크 트래픽 정보를 제공하는 네트워크 분석 시스템으로는 MRTG가 있으나 관련연구에서 소개했듯이 MRTG는 네트워크 트래픽의 총 양을 보여 줄 뿐 요구사항 1), 2)같은 호스트 정보, 프로토콜 정보 등을

보여 주지 못해 네트워크 트래픽의 원인을 정확히 진단할 수 없다.

ntop과 WebTrafMon으로는 나누어진 네트워크의 트래픽을 분석할 수 없다. 나누어진 네트워크의 트래픽을 분석하기 위해서는 한 노드와 다른 노드의 네트워크 트래픽을 합쳐서 분석할 수 있어야 한다. NNStat 이 여러 네트워크 노드의 트래픽을 분석할 수 있는 기능을 제공한다. 그러나 NNStat은 SunOS 4.*의 NIT 기반에서 개발되어 SunOS 4.*에서만 동작하는 단점이 있다.

ntop이나 WebTrafMon은 패킷 캡처와 패킷 분석을 동시에 하므로 시스템 과부하 시 패킷 손실이 발생한다. ntop의 경우 패킷 분석 모듈이 시스템 자원을 적게 쓰도록 만들었으나 패킷이 많이 들어 올 경우 패킷 캡처를 독립적으로 하는 시스템 보다 시스템 과부하로 인한 패킷 손실의 위험이 높다.

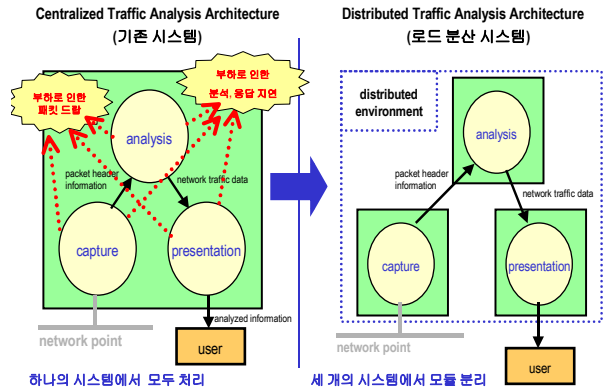
이런 문제들을 고려해 볼 때 요구사항 1), 2), 3) 이외에 다음과 같은 요구사항이 필요하다.

- 4) 네트워크 트래픽을 매시, 매일, 매달, 매년 단위 등으로 장기간 분석해서 자동으로 보여 줄 수 있어야 한다.
- 5) 여러 네트워크 노드에서의 네트워크 트래픽을 합쳐서 분석할 수 있어야 한다.
- 6) 패킷 캡처가 독립적으로 이루어져 패킷 분석으로 인한 시스템 과부하 시의 패킷 손실이 없어야 한다.
- 7) 멀티미디어 트래픽을 분석할 수 있어야 한다.

본 논문에서는 요구사항 1), 2), 3) 뿐 만 아니라 4), 5), 6), 7)도 반영할 수 있는 네트워크 모니터링 및 분석 시스템을 설계 및 구현하였다. 이 시스템의 가장 큰 특징은 요구사항 4), 5), 6), 7)을 반영하기 위해 분산 시스템에서의 로드 분산 방법을 사용했다는 점이다. 패킷 캡처, 패킷 분석, 웹 뷰어 등으로 모듈을 나누고 이 세 모듈을 각각의 시스템에서 동작하게 함으로써 여러 네트워크 노드에서 패킷 캡처를 할 수 있고 패킷 분석 시의 시스템 과부하가 패킷 캡처에 영향을 주지 않는다.

4. 설계

기본적으로 WebTrafMon II는 그림 1과 같이 로드 분산 기법을 적용한다.

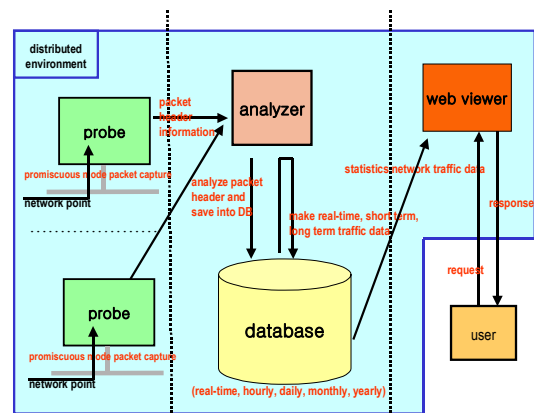


(그림 1) WebTrafMon II에 적용한 로드 분산 방법

그림 1에서와 같이 기존 시스템은 한 시스템에서 패킷 캡처 및 분석을 하므로 패킷 손실이 발생하고 패킷을 빠른 시간에 분석하기가 어렵다. 따라서 그림 1의 로드 분산 시스템과 같이 각 모듈을 독립시키면 시스템 과부하로 인한 패킷 손실을 막고 빠른 분석을 할 수 있다.

또, 관련 연구에서 조사한 멀티미디어 포트 번호를 분석 시에 반영 하면 멀티미디어 트래픽 분석이 가능하다.

앞에서 살펴본 요구조건들과 위의 로드 분산 방법을 기본으로 해서 WebTrafMon II를 설계한다. 설계 구조는 그림 2와 같이 분산된 시스템을 기본으로 한다.



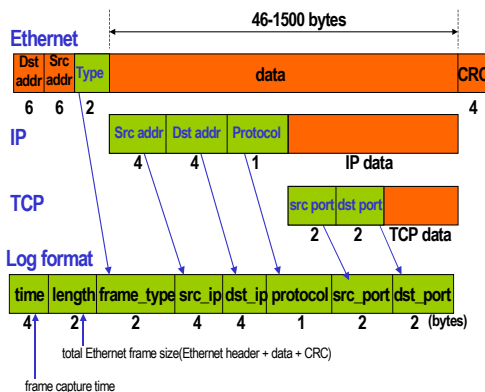
(그림 2) WebTrafMon II의 구조

그림 2와 같이 WebTrafMon II는 패킷을 캡처해서 파일에 저장하는 패킷 캡처 모듈 (probe)과 파일에 저장된 패킷 정보를 분석해

서 데이터베이스에 저장한 뒤 매시, 매일, 매달, 매년 단위로 분석하는 패킷 분석 모듈(analyzer), 그리고 분석된 정보를 웹에서 보여주는 웹 뷰어(web viewer) 모듈로 구성되어 있다.

4.1 패킷 캡처 모듈(probe)의 설계

패킷 캡처 모듈은 네트워크 상의 패킷을 캡처한 뒤 패킷 헤더로부터 원하는 정보를 추출하여 로그 파일에 저장하는 모듈이다. 그림 3은 프레임에서 정보를 추출해서 로그 파일에 저장하는 내용을 설명한다.



(그림 3) 패킷 캡처 모듈의 로그 정보

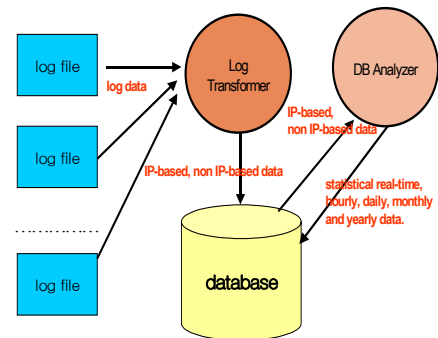
로그 포맷의 time은 패킷을 캡처해서 로그 파일에 저장할 때의 시간이며 length는 패킷 전체의 크기에 CRC값(4 바이트)을 더한 값이다. time과 length를 제외한 나머지 정보는 모두 패킷에서 가져오는데 ether_type은 이더넷 헤더에서 정보를 가져오고 src_ip, dst_ip, protocol 등은 IP 프로토콜 헤더에서, src_port, dst_port 등은 TCP/UDP 헤더에서 정보를 가져온다. 로그 포맷의 일정 필드를 사용하지 않는 프로토콜은 사용하지 않는 필드를 0으로 채운다. 예를 들어, ARP같이 IP기반이 아닌 프로토콜은 time, length, ether_type 등의 값만 저장한뒤 사용하지 않는 src_ip, dst_ip, protocol, src_port, dst_port 등은 모두 0으로 저장한다. 또, ICMP같이 IP기반이지만 포트 정보를 사용하지 않는 프로토콜은 time, length, ether_type, src_ip, dst_ip 등의 값만 저장한뒤 src_port, dst_port 등을 0으로 저장한다. 포트 번호 중 0번이 있으나 분석할 때 TCP/UDP 프로토콜 여부를 먼저 체크한 뒤 포트 분석을 하므로 TCP/UDP의 포트 번호 0번과 빈값을 채워넣는 0값과 구분할 수 있다.

그림 1과 같이 여러 네트워크 노드의 트래픽을 합쳐서 분석하려면 패킷 캡처 모듈을

여러곳에서 실행시킬 수 있어야 한다. 따라서 패킷 캡처 모듈을 패킷 분석 모듈과 독립적인 모듈로 설계하였다. 독립적으로 설계된 패킷 캡처 모듈과 패킷 분석 모듈이 로그 정보를 공유하기 위해 패킷 캡처 모듈은 매 시간마다 로그 파일을 새로 만들고 패킷 분석 모듈은 한 시간 전의 로그 파일을 읽어 분석한다.

4.2 패킷 분석 모듈(analyzer)의 설계

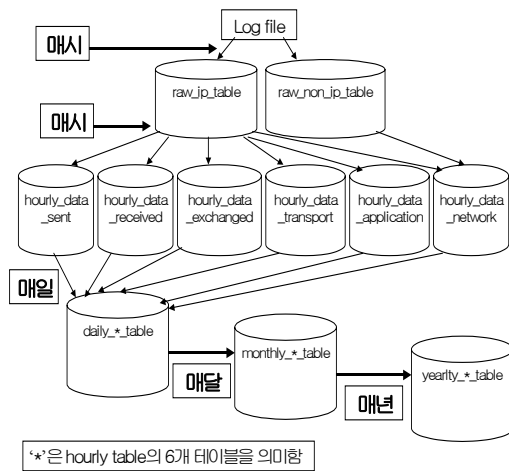
패킷 분석 모듈은 패킷 캡처 모듈이 저장한 파일 정보를 IP와 non IP 데이터로 분류해서 데이터베이스에 저장하는 로그 변환기와 분류된 데이터를 가지고 분석하고 통계를 내서 매시, 매일, 매달, 매년의 데이터를 만들어주는 DB분석 시스템으로 나눈다. 그림 4는 패킷 분석 모듈의 구조이다.



(그림 4) 패킷 분석 모듈의 구조

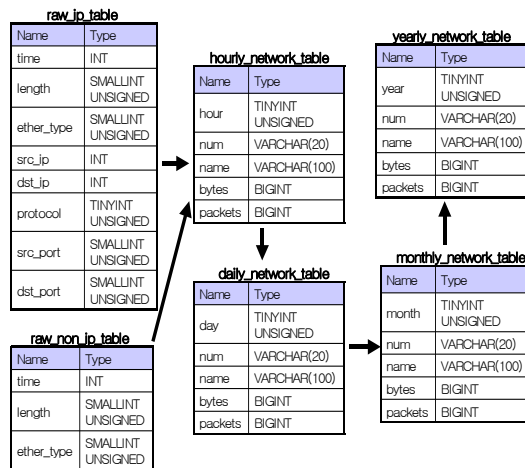
여러 노드에서 패킷 캡처 모듈을 실행해서 매 시간마다 각각의 로그 파일이 만들어지면 로그 변환기가 각각의 로그 파일들의 정보를 합쳐서 데이터베이스에 넣는다. 그 후 DB분석 시스템이 데이터베이스에 저장된 정보를 매 시간마다 분석하여 호스트 정보, 프로토콜 정보, 어플리케이션 정보를 만들어서 데이터베이스에 저장한다. 그리고 이렇게 매 시간마다 저장된 데이터를 가지고 하루, 한달, 일년 주기로 통계를 내서 데이터베이스에 저장한다. 멀티미디어 트래픽 분석은 데이터베이스에 저장된 포트 번호를 반영하므로 쉽게 분석할 수 있다.

그림 5는 로그 변환기와 DB분석 시스템에 의한 네트워크 트래픽 데이터베이스 테이블의 생성과 변화를 나타낸다.



(그림 5) 패킷 분석 모듈의 데이터 이동

로그 변환기가 로그 파일을 분석해서 raw_ip_table과 raw_non_ip_table을 만든다. raw 테이블 두 개를 이용해서 DB분석 시스템이 호스트, 프로토콜, 어플리케이션 등에 관련된 데이터를 분석하여 hourly 테이블을 만들어 낸다. 그 다음 부터는 hourly 테이블을 가지고 차례로 통계를 내서 daily, monthly, yearly 테이블을 만든다. 호스트 정보는 data_sent table, data_received_table, data_exchanged_table 등 3개의 테이블로 나누었고 프로토콜 정보는 network_table과 transport_table로 나누었다. 그림 6에 raw 테이블과 네트워크 테이블을 예로 해서 hourly, daily, monthly, yearly 테이블의 데이터베이스 스키마를 나타냈다.

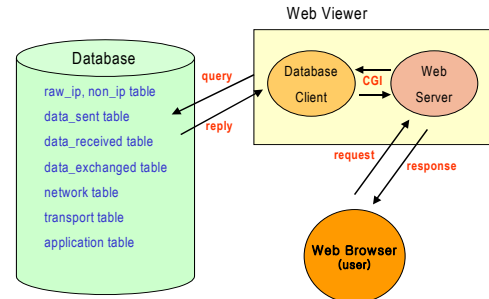


(그림 6) 장기간 network 정보 분석을 위한 데이터베이스 스키마

4.3 웹 뷰어(web viewer) 모듈의 설계

웹 뷰어는 패킷 분석 모듈이 분석 및 통계를 내서 저장한 매시, 매일, 매달, 매년의 네

트워크 트래픽 데이터를 웹에서 보여주는 모듈이다. 데이터베이스의 스키마 내용을 적절하게 보여 줄 수 있도록 웹 뷰어에서도 매시, 매일, 매달, 매년으로 나누어 정보를 확인할 수 있도록 설계했다. 그림 7은 웹뷰어의 설계를 설명한다.



(그림 7) 웹 뷰어의 설계

사용자가 웹 브라우저를 통해 WebTrafMon II에 접속하면 웹 뷰어에서 매시, 매일, 매달, 매년의 정보를 데이터베이스에서 가져온 뒤 정보를 가공해 각각의 네트워크 트래픽의 총 패킷 수와 총 패킷 사용량을 보여 준다. 그 뒤 사용자가 매시, 매일, 매달, 매년의 정보 각각에서 호스트 정보, 프로토콜 정보, 어플리케이션 정보를 요청하면 데이터베이스 클라이언트를 통해 데이터베이스 질의를 데이터베이스 서버에 요청해 정보를 가공한 뒤 사용자에게 보여 준다. 사용자가 웹 뷰어를 통해 데이터베이스에 멀티미디어 포트 정보를 입력할 수 있으며 이것이 반영되어 멀티미디어 트래픽을 분석한다.

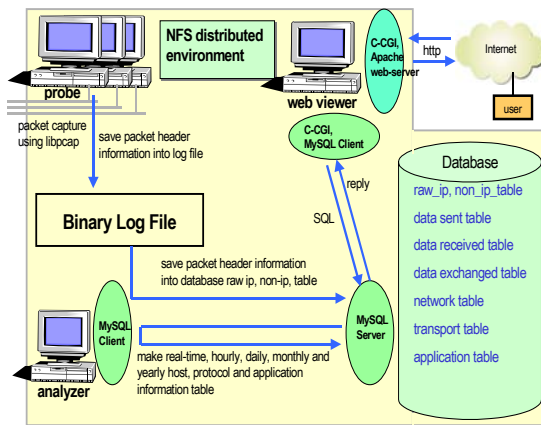
5. 구현

WebTrafMon II의 구현은 분산 시스템 환경에서 로드 분산 방법을 사용했다. 로드 분산 방법을 사용한 이유 중 하나가 시스템 과부하로 인한 패킷 손실을 막기 위함이다. 하나의 시스템에서 패킷을 캡처 하면서 동시에 분석 할 경우 시스템 과부하로 인한 패킷 손실의 위험이 높아진다. 따라서 분산 시스템에서 패킷 캡처 하는 모듈과 패킷 분석하는 모듈을 각각 하나의 시스템에서 동작하게 하여 이러한 위험을 제거했다. 분산 시스템을 사용한 다른 이유는 여러 네트워크 노드의 트래픽을 합쳐서 분석할 수 있는 환경을 만들기 위해서 이다.

WebTrafMon II는 Pentium III 800, 516 MB의 PC 여러 대에 리눅스를 설치하여 NSF [24]환

경을 만든 뒤 C, C-CGI, Mysql 3.22.32, Apache Web Server 1.3.14를 이용해서 구현하였다.

그림 8은 WebTrafMon II의 로드 분산 방법을 설명해 준다.



(그림 8) WebTrafMon II의 시스템 구조

그림 8과 같이 여러개의 패킷 캡처 모듈이 네트워크 노드의 트래픽 정보를 NFS 환경에서 각각의 로그 파일에 저장한다. 이렇게 저장된 각각의 로그 파일들을 패킷 분석 모듈이 분석하여 데이터베이스에 저장하고 웹 뷰어는 분석된 데이터를 사용자에게 보여준다. 로그 파일과 데이터베이스도 패킷 캡처 모듈, 패킷 분석 모듈, 웹 뷰어 등과 독립되어야 하므로 NFS 파일 서버를 사용하여 하나의 시스템에서 동작시켰다.

5.1 패킷 캡처 모듈(probe)의 구현

패킷 캡처 모듈을 구현하기 위해서 libpcap [25] 라이브러리의 promiscuous 모드를 사용했다. libpcap 라이브러리는 유닉스 계열의 다양한 운영체제에서 쓸 수 있도록 운영체제에 독립적인 API를 제공하는 장점이 있어 대부분의 네트워크 분석 시스템에 사용된다. libpcap이 운영체제에 독립적이기에 WebTrafMon II도 다양한 운영체제에서 사용이 가능하다.

패킷 캡처 모듈에서 time을 저장함으로써 매시, 매일, 매달, 매년의 데이터 뿐만 아니라 특정 시간대의 사용자가 요구하는 시간 간격의 정보도 보여 줄 수 있다.

캡처한 패킷 정보를 데이터베이스에 저장하는 것보다 파일에 저장하는 것이 더 빠르고 데이터베이스에 직접 저장할 경우 패킷 캡처 모듈이 동작하는 시스템에서 과부하로 인한 패킷 손실이 발생할 수 있기에 로그 파일을 거쳐 데이터베이스에 정보를 저장하도

록 구현했다. 로그 파일을 사용할 경우에는 패킷 분석 모듈이 동작하는 시스템에서 로그 정보를 데이터베이스로 저장하므로 패킷 캡처 모듈이 동작하는 시스템에 부하를 주지 않는다. 로그 파일을 텍스트 파일이나 바이너리 파일로 만들 수 있는데 텍스트 파일은 눈으로 로그 정보를 확인할 수 있는 장점은 있으나 저장하는데 바이너리 파일보다 시간이 많이 걸리고 저장한 내용을 데이터베이스로 옮기는 작업이 쉽지 않아 텍스트 파일보다 더 빠르게 저장되고 데이터베이스에 쉽게 저장할 수 있는 바이너리 파일을 사용하였다.

5.2 패킷 분석 모듈(analyzer)의 구현

패킷 분석 모듈은 C 언어와 Mysql 3.22.32 데이터베이스를 이용해서 구현했다. 패킷 분석 모듈은 메시가 되면 패킷 캡처 모듈에서 저장한 바이너리 파일을 읽어 IP기반 패킷과 non IP기반 패킷으로 나누어 데이터베이스에 저장한다.

패킷 분석 모듈은 매시, 매일, 매달, 매년의 네트워크 트래픽 정보를 자동으로 분석해서 데이터베이스에 저장해야 하므로 Unix의 cron명령어를 이용해서 매시, 매일, 매달, 매년마다 패킷 분석 모듈이 자동으로 동작하도록 했다.

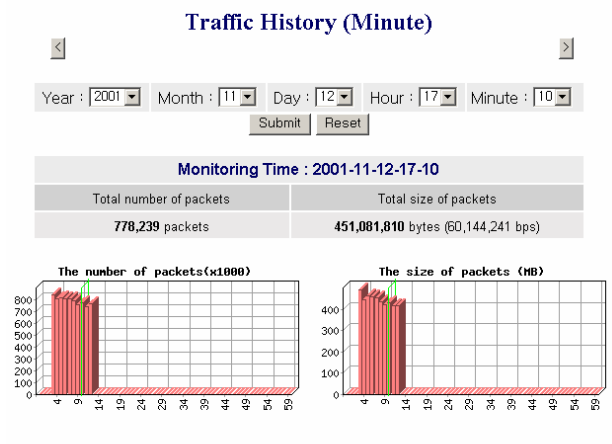
또 웹기반에서 사용자에게 TCP/UDP 포트 번호와 포트 이름을 입력 받을 수 있도록 했다. 이렇게 구현한 이유는 다양한 어플리케이션이 개발되는 것을 고려해서 어플리케이션 계층 분석의 확장성과 정확성을 높이기 위함이다. 사용자가 특정한 포트 번호와 어플리케이션 이름을 입력해 주면 분석 할 때 이것이 반영되어 잘 알려진 포트 번호 외에 최근 개발된 어플리케이션까지 분석할 수 있다.

5.3 웹 뷰어(web viewer) 구현

웹 뷰어는 Apache Web Server 1.3.14를 이용해서 동작시켰으며 네트워크 트래픽 정보를 사용자에게 편리하게 제공하기 위해 C 언어와 gd [26]라이브러리를 사용하여 네트워크 트래픽의 총 양을 보여 주는 그래프를 그렸다. 또 데이터베이스의 내용을 아파치 웹 서버로 전달하기 위해 C-CGI를 사용하여 웹과 데이터베이스를 연동하였다.

5.4 구현 화면

분 별 데이터를 가지고 예를 든다. 다음 그림 9는 매 분 별 데이터를 보여 준다.



(그림 9) Minute View의 첫 화면

그림 9의 두 그래프는 일분 동안의 총 패킷 수와 총 네트워크 트래픽 양을 보여 준다. 제일 위의 시간 창에서 2001년 11월 12일 17시 10분의 데이터를 선택했다는 것을 알 수 있고 11월 12일 17시의 매 분 별 데이터를 보여 준다.

그림 9에서 Data Sent를 선택하면 그림 10와 같이 2001년 11월 12일 17시 10분에 데이터를 가장 많이 보낸 호스트 10개를 보여 준다. 그림 10에서 141.223.5.20 호스트가 가장 많은 네트워크 트래픽을 보냈고 그 다음으로 141.223.12.101이며 다음으로 브로드캐스트 트래픽이 많음을 알 수 있다.

Monitoring Time : 2001-11-12-17-10

Host Information : Data Sent (TOP 10)

Order by Byte	Source	Packets	Bytes
1	141.223.5.20 home.postech.ac.kr	20,130 (2.59 %)	30,134,098 (6.68 %)
2	141.223.12.102 nova.postech.ac.kr	21,326 (2.74 %)	22,130,189 (4.91 %)
3	255.255.255.127 unknown	15,972 (2.05 %)	20,815,714 (4.61 %)
4	255.255.255.127 unknown	12,607 (1.62 %)	19,270,710 (4.27 %)
5	255.255.255.127 unknown	11,927 (1.53 %)	17,372,598 (3.85 %)
6	211.175.250.8 unknown	11,367 (1.46 %)	16,558,748 (3.67 %)
7	141.223.163.50 jupiter.postech.ac.kr	21,738 (2.79 %)	14,466,954 (3.21 %)
8	141.223.2.112 stkim2.postech.ac.kr	5,256 (0.68 %)	7,792,615 (1.73 %)
9	64.71.189.34 jsplanning.he.net	5,185 (0.67 %)	7,731,384 (1.71 %)
10	141.223.170.122 unknown	4,979 (0.64 %)	7,715,257 (1.71 %)

[Top 100] [Show All]

(그림 10) 2001년 11월 12일 17시 10분의 Data Sent 정보

그림 10에서 3, 4, 5 번의 255.255.255.127은 서로 다른 곳에서 발생한 브로드캐스트 메시지이다. 브로드캐스트 메시지가 많다는 것을 알 수 있다. 또, 그림 10의 호스트가 사용한 프로토콜 정보를 확인하기 위하여 141.223.5.20을 선택하면 그림 11의 정보를 볼 수 있다.

Monitoring Time : 2001-11-12-17-10

Data Sent : 141.223.5.20 (home.postech.ac.kr)

Network Layer

Order by Byte	ether type (number)	Packets	Bytes
1	Internet IP (IPv4) (2048)	20,130 (100.00 %)	30,134,098 (100.00 %)

Transport Layer

Order by Byte	protocol (number)	Packets	Bytes
1	TCP (6)	20,130 (100.00 %)	30,134,098 (100.00 %)

Application Layer

Order by Byte	source port	Packets	Bytes
1	http (80)	20,130 (100.00 %)	30,134,098 (100.00 %)

(그림 11) 141.223.5.20이 사용한 프로토콜 정보

그림 11에서 141.223.5.20은 트래픽 모니터링 시간 범위 내에서 TCP/IP 기반의 http 서비스만을 사용했음을 알 수 있다.

6. 결론 및 향후 과제

현재 WebTrafMon II는 네트워크 트래픽을 장기간 분석하여 때에 따라 보여 줄 수 있도록 분산 시스템에서 구현되어 있다. WebTrafMon II는 다른 네트워크 분석 시스템들과 달리 시간 단위에서 매일, 매달, 매년 단위의 장기간 네트워크 트래픽 데이터를 분석해 줄 수 있으며 로드 분산 방법을 사용하여 하나의 시스템에서 독립적인 패킷 캡처를 하므로 시스템 과부하 시의 패킷 손실을 막을 수 있고 여러 네트워크 노드에서의 트래픽을 합쳐서 분석할 수 있다는 장점이 있다. 또 사용자에게 어플리케이션 계층의 정보를 입력 받아 어플리케이션 계층 분석의 확장성을 갖는 것도 장점이다. 그 예로 스트리밍, P2P 등의 멀티미디어 트래픽도 분석할 수 있는 기능을 제공한다.

앞으로 사용자에게 정보를 좀 더 빨리 보여 줄 수 있도록 패킷 분석 모듈과 웹 뷰어 등의 성능을 개선하고 사용자가 편리하게 사용할 수 있는 화면 설계 및 개선이 요구된다. 또, IPoATM 등의 다양한 플랫폼에서의 트래픽 분석 및 모니터링과 1 Gbps 이상의 초고속 통신망에서의 트래픽 분석 및 모니터링에 관한 연구도 필요하다. 서비스를 좀 더 정확히 확인하기 위한 서비스 내용 분석도 최근의 어플리케이션이 늘어나는 추세에 맞는 좋은 연구가 될 것이다.

[참고문헌]

- [1] Lawrence Berkley National Laboratory, "tcpdump 3.6", <http://www.tcpdump.org>.
- [2] L. Deri and R. Carbone, "Monitoring Networks Using Ntop", Released paper in <http://luca.ntop.org>, Jan 29th 2001.
- [3] Peer-to-Peer Service, "<http://www.peer-to-peerwg.org/>".
- [4] Soon Sun Kwun, Jae Young Kim, James Won-Ki Hong, "Internet/Intranet Network Traffic Monitoring and Analysis System, KNOM Review Vol 2, No 1, April 1999, pp.1-10.
- [5] Tobias Oetiker and Dave Rand, "MRTG: Multi-Router Traffic Grapher", <http://www.mrtg.org>.
- [6] Ethereal, <http://www.ethereal.com>.
- [7] Robert T. Braden and Annette L. DeSchon, "NNStat: Internet Statistics Collection Package", USC/Information Sciences Institute Marina del Rey, California, November 28, 1988.
- [8] Werner Erhard, Michael M. Gutzmann and Hastings M. Libati, "Network Traffic Analysis and Security Monitoring with UniMon", Proceedings of the IEEE Conference on, 2000, pp.439-446.
- [9] Lawrence Berkeley National Laboratory, "tcplice-1.1a3", <ftp://ftp.ee.lbl.gov/tcplice.tar.Z>.
- [10] sun, "<http://www.sun.com/products/sunray1/ts-sysmon.html>".
- [11] Carter Bullard, "argus-1.7.beta.1b", <ftp://ftp.sei.cmu.edu/pub/argus>.
- [12] Lawrence Berkley National Laboratory, "arpwatch 2.0", <ftp://ftp.ee.lbl.gov/arpwatch.tar.Z>.
- [13] Dave Curry and Jeff Mogul, "nfswatch-4.3", <ftp://ftp.lip6.fr/pub2/networking/nfs>.
- [14] David K. Hess and Douglas Lee Schales, David R. Safford, "drawbridge 2.0", <http://www.certcc.or.kr/tools/index.html>.
- [15] ewatch, "<http://ewatch.hangkong.ac.kr>".
- [16] sniffer pro, "<http://www.softseek.com>".
- [17] Real Player, "<http://www.real.com>".
- [18] Audio-Video Transport Working Group, "RTP: A Transport Protocol for Real-Time Applications", IETF RFC 1889, January 1996.
- [19] Windows Media Player, "<http://www.Microsoft.com>".
- [20] MMST, "<http://www.microsoft.com/windows2000/en/server/help>".
- [21] FastTrack, "<http://www.fasttrack.nu>".
- [22] Gnutella, "<http://www.gnutellaworld.net/gw/stories.php>".
- [23] FreeNet, "<http://freenet.sourceforge.net/>".
- [24] sun, "<http://www.sun.com/software/white-papers/wp-nfs.sw>".
- [25] Lawrence Berkley National Laboratory, "libpcap 0.6.2", <http://www.tcpdump.org>.
- [26] BOUTELL.COM, "gd 1.8.2", <http://www.boutell.com/gd/>.
- [27] R. Enger and J. Reynolds, "FYL on a Network Management Tool Catalog", IETF RFC 1470, June 1993.



홍 순 화

2000 숭실대학교, 컴퓨터공학과 학사

2000-현재 포항공과대학교 컴퓨터공학과 석사
과정

관심분야: 인터넷 트래픽 모니터링, 분산처리



김 재 영

1994 포항공과대학교, 전자계산학 학사

1996 포항공과대학교, 전자계산학 석사

1996-1998 포항공과대학교 학술정보센터 연구
원

1998-현재 포항공과대학교 컴퓨터공학과 박사
과정

관심분야: 네트워크 및 분산 시스템 관리, 분
산처리, CORBA, 인터넷 서비스 관리



홍 원 기

1983 Univ. of Western Ontario, 전산학 학사

1985 Univ. of Western Ontario, 전산학 석사

1985-1986 Univ. of Western Ontario, 전산학과 강
사

1986-1991 Univ. of Waterloo, 전산학 박사

1991-1992 Univ. of Waterloo, Post-Doc fellow

1992-1995 Univ. of Western Ontario, 연구교수

1995-현재 포항공과대학교 컴퓨터공학과 부교
수

관심분야: 분산처리, 네트워크 및 분산 시스
템 관리, CORBA, Internet 관리.