패킷 분석을 이용한 네트워크 모니터링 시스템

백유진, 정인권, 김용환, 김선명
국립금오공과대학교 컴퓨터소프트웨어공학과
e-mail: asdfgogo@lycos.co.kr

Network Monitoring System by Analysing Packets

Eu-jin Baek, In-kwon Chung, Yong-hwan Kim, Sunmyeng Kim
Department of Computer Software Engineering
Kumoh National Institute of Technology

요 약

무선 네트워크를 이용한 데이터 송수신이 일반화되면서 네트워크 트래픽에 대한 분석 및 제어에 대한 관심이 높아지고 있다. 본 논문에서는 무선 통신에서 트래픽 분석에 대한 효율성을 증진시키기 위해 패킷 캡처 기능을 이용하여 무선 통신 트래픽 모니터링 기능을 설계하였다. 특히 libpcap 라이브러리를 이용하여 Embedded Linux 시스템에서 패킷 캡처 프로그램을 구현하였다. 이를 기반으로 네트워크를 통해 송수신되는 패킷을 캡처하여 분석하고 이에 따른 트래픽 모니터링 결과를 보인다.

1. 서론

최근 인터넷의 사용자가 증가함에 따라서 새로운 프로토콜의 등장과 각종 새로운 콘텐츠 종류의 데이터들이 네트워크를 통해 전송됨으로써 기술적, 사회적인 새로운 이슈들이 등장하고, 네트워크 기반의 응용 프로그램이 다양하게 개발됨에 따라 네트워크 트래픽이 증가하고 있다. 네트워크 트래픽의 증가로 인해 네트워크 회선의 부족이나 네트워크 응답시간의 저하 등의 문제가 발생 할 수 있으며 이를 대비하기 위해 네트워크 모니터링 시스템의 중요성이 부각되고 있다.

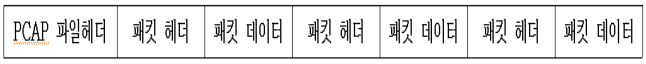
이를 고려할 때 네트워크 모니터링 시스템의 개발 필요성은 매우 크다고 할 수 있다. 네트워크 트래픽 모니터링은 네트워크를 관리하는데 있어서 매우 중요한 일이다. 그러나 네트워크 트래픽의 모니터링이 등한시 되거나 제대로 이루어지지 못하는 경우가 많이 발생된다. 네트워크 트래픽 모니터링이 필요한 이유는 네트워크 트래픽의 현재 상황 및 문제점을 파악하기 위해서이다. 네트워크에서 어느 시간대에 트래픽 발생이 높은가를 파악하려면 네트워크 트래픽 모니터링 및 분석을 통해 알 수 있어야 한다. 또한 최고 트래픽이 어느 정도인지, 어느 시간대인지, 어느 정도의 트래픽이 있는지, 그리고 어느 정도로 증가하고 있는지 등 네트워크 회선 계획을 세우는 데 필요하다. 이와 같은 이유로 인해 네트워크 트래픽 모니터링 및 분석이 필요하다.

본 논문 구성은 다음과 같다. 2장에서는 관련 연구에 대한 설명하고, 3장에서는 시스템 구성 및 모듈설계에 대해 자세히 설명한다. 이어 4장에서는 구현된 모니터링 시스템의 성능에 대해 기술한다. 마지막으로 5장에 결론과 향후 방향에 대해 기술한다.

2. 관련 연구

패킷 캡처란 네트워크를 통해 송수신되는 패킷의 내용을 확인하는 것을 의미한다. 패킷 캡처를 위해 libpcap 라이브러리를 사용한다. 각 운영체제의 밴더들이 각각의 패킷 캡처 도구들을 제공하고 있어 개발이나 포팅 등에 어려움이 있기 때문에 각 도구들의 기능을 포함하면서 시스템에 독립적인 libpcap이 등장하게 되었다. 이러한 패킷 캡처는 네트워크의 사용에 대한 통계와 보완을 목적으로 하는 모니터링, 네트워크를 디버깅하기 위한 목적, 스니퍼 등 다양한 형태로 응용이 가능하다.

네트워크상에 전송되는 패킷을 캡처하고 저장한 것을 패킷 파일이라고 한다. 여기에는 여러 가지 형태가 있다. 그 중 대표적인 것이 PCAP 파일이다. 흔히 패킷 캡처 파일이라 하면 PCAP 형태의 파일을 의미한다. PCAP (Packet Capture)은 그림 1과 같은 구조를 갖는다.



(그림 1) PCAP 파일의 구조

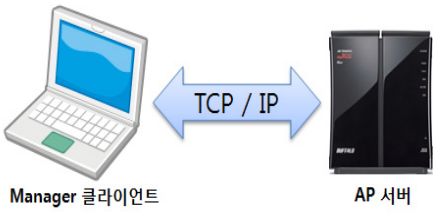
파일의 첫 부분에 PCAP 파일임을 말해주는 헤더가 있고 그 후 패킷의 헤더, 이어서 해당 패킷 데이터가 있다. 패킷 헤더와 패킷 데이터가 연속적으로 반복 기록된다.

3. 패킷분석을 이용한 네트워크 모니터링 시스템

3.1 시스템의 구성

네트워크 모니터링 시스템은 AP에서 동작하는 서버와 데스크탑에서 동작하는 매니저 어플리케이션으로 구성된다. (그림 2 참조). 네트워크 모니터링 시스템은 AP 인터

페이스의 패킷을 캡처 후, 매니저 PC와 유선 소켓 통신을 통해 패킷 정보를 전송하고 매니저 어플리케이션에서 AP에 연결된 디바이스의 전체 네트워크를 모니터링 할 수 있는 방식으로 구현하였다.



(그림 2) TCP/IP를 사용한 구조

매니저 어플리케이션으로 전송된 패킷 정보는 분석 툴을 통해서 각 패킷마다 헤더정보를 분석한다. 즉, 프로토콜 사용량, 소스 IP의 패킷 발생 수 등을 분석해서 사용자에게 보여준다. 매니저 어플리케이션의 제어명령을 AP에 전송 해당 IP 디바이스의 패킷을 차단 또는 허용 할 수 있다.

3.1.1 AP 서버

AP 서버 프로그램은 기존 펌웨어를 OPEN-WRT 오픈소스 펌웨어로 교체하고 네트워크 모니터링에 필요한 기능을 가진 패키지 형태의 프로그램으로 제작하였다. 데스크탑에서 작성해서 크로스컴파일 후 AP에서 실행하는 방식으로 구축하였다. 매니저 어플리케이션과 소켓통신을 통해 사용자의 제어 명령으로 동작하게 된다.

3.1.2 데스크탑 매니저

MFC를 이용한 윈도우 어플리케이션으로 서버로 디바이스 차단과 허용 명령을 전송할 수 있는 인터페이스와, 서버에서 전송된 패킷내용을 분석해서 사용자에게 보여주는 인터페이스로 구성되어 있다.

사용자에게 AP의 패킷 내용을 분석해서 사용자에게 헤더정보 분석과, IP별 카운터, 프로토콜 등을 보여준다.

3.2 주요기능

3.2.1 패킷 캡처

패킷 캡처 기능의 전체적인 구조는 어딘가에서 정보를 얻어오는 다른 프로그램들과 마찬가지이다. 예를 들어 파일에서 정보를 얻어오는 경우, 우선 파일을 열고, 파일을 읽고, 분석하고, 파일을 닫게 된다. 패킷 캡처도 우선 패킷 캡처할 디바이스(NIC)나 기존의 저장된 파일을 열고, 패킷을 읽고, 분석한 후 디바이스나 파일을 닫게 된다.

일반적으로 이더넷 환경에서 라우터는 내부 네트워크를 향하는 패킷들을 브로드캐스팅 하게 되고 각 컴퓨터들은 자신의 인터페이스로 들어오는 패킷 중 목적지가 자신인 경우에만 받아들여 이를 운영체제가 처리한다. 패킷 캡처는 이처럼 자신에게 전달되는 패킷을 받아 들여 패킷의 내용을 확인할 수 있음을 의미한다. 이러한 패킷 캡처는 네트워크의 사용에 대한 통계와 보안을 목적으로 하는 모니터링, 네트워크를 디버깅하기 위한 목적, 스니핑 등 다양한 형태로 응용이 가능하다. 우리는 패킷 캡처 기능을 구현하기 위해 각 도구들의 기능을 포함하면서 시스템에 독립적인 libpcap을 이용하였다. libpcap은 커널 수준이 아닌 사용자 레벨에서 저수준의 네트워크 모니터링을 가능케 하는 인터페이스를 제공한다.

AP의 인터페이스로부터 패킷의 길이와 패킷의 정보를 얻어온 후, 소켓 통신을 통해 패킷의 길이와 패킷 정보를 데스크탑 매니저 어플리케이션으로 전송한다.

3.2.2 패킷분석

앞에서 설명한 패킷 캡처기능이 활성화되면 패킷의 내용은 관리자 PC에 설치되어 있는 프로그램으로 전송된다. 패킷을 캡처하고 단순히 그 패킷의 내용을 보는 것으로는 고수준의 교육을 받은 사람이 아닌 상황에서는 사용자가 원하는 정보를 추출해내는 것이 쉽지 않다. 그렇기 때문에 패킷을 분석하여 사용자가 원하는 정보를 평소 사용하는 표현 체계로 보여주는 것으로 얻어진 정보의 활용도를 높일 수 있다. 시스템은 전송받은 패킷을 기반으로 헤더내용을 추출하고 추출된 내용은 각각 트래픽을 측정하기 위한 용도와 TCP, UDP, ARP 등의 프로토콜의 규격에 따라 분류된다. 또한 IP, Port 별 접속되어 있는 디바이스들의 리스트를 관리하고 그들이 생성하는 패킷 양을 통하여 현재 실행되고 있는 트래픽 양을 측정한다. 각 패킷의 정보는 정해진 규격으로 표현되기 때문에 규격을 프로그램화하여 분석한다.

공공장소에서 사용되거나 법적인 규제로 인하여 개인 사생활 보호를 위해 패킷의 전체적 내용은 변환하지 않고 그대로 보여주며 패킷의 헤더 정보만 이용하여 분석한다.

3.2.3 트래픽 제어

트래픽 제어 기능은 사용자가 네트워크 분석을 통해 제공된 정보를 이용하여 해당 IP의 트래픽을 차단하거나 허용하고 싶을 때 사용하는 기능이다. AP에 접속된 특정 사용자가 많은 트래픽을 사용할 시, 전체적인 네트워크 측면에서 비효율적이다. 따라서 원하는 사용자의 접속을 차단하고, 지속적인 트래픽 증가량을 파악하여 통신을 복구시키는 등 접속제어가 필요하다.

이러한 제어를 위해 ICMP(Internet Control Message Protocol)프로토콜을 사용한다. ICMP는 네트워크 계층의 한 부분으로 에러나 주의가 요구되는 상황에서 사용된다. 즉, 네트워크 환경의 운영체제에서 오류 메시지를 전송받는 데 주로 쓰이며 프로토콜의 주요 구성원 중 하나로 인터넷 프로토콜에 의존적인 작업을 수행한다. ICMP 메시지는 네크워크 계층 또는 상위 계층에 의해 호출된다.

ICMP는 0번부터 40번까지의 메시지 타입을 가지고 있

고 각 타입마다 전달하는 메시지가 다르다. 표 1은 각 타입 번호별 전달하는 메시지를 나타낸다.

<표 1> ICMP 메시지 타입

TypeField	ICMP Message Type
0	Echo Reply
3	Destination Unreachable
4	Source Quench
5	Redirect (change a route)
6	Alternate Host Address
8	Echo Request
...	
36	Mobile Registration Reply
37	Domain Name Request
38	Domain Name Reply
39	Skip
40	Photuris

사용자가 IP차단 명령을 서버로 전송할 경우 서버는 ICMP Message Type 3번-목적지 도달불가 메시지를 생성하여 해당 디바이스에게 전송한다. 목적지 도달불가 메시지의 경우 라우터가 데이터그램을 라우팅할 수 없을 때, 호스트가 데이터그램을 배달할 수 없을 때, 데이터그램을 폐기한 후 발신지 호스트에 오류를 보고할 때 사용한다. 이 메시지가 해당 디바이스로 전송됨으로써 디바이스는 접근 실패 메시지를 받아들이게 됨으로써 AP와의 접속을 차단시키는 것이다. 차단 디바이스의 트래픽이 패킷 분석을 통해 줄어들었을 경우 ICMP Message 전송을 취소함으로써 AP로의 접근차단을 해제한다. 디바이스의 AP연결을 제어함으로써 무선 통신을 분석 후 네트워크 환경을 현재보다 개선시킬 수 있다.

4. 실험결과

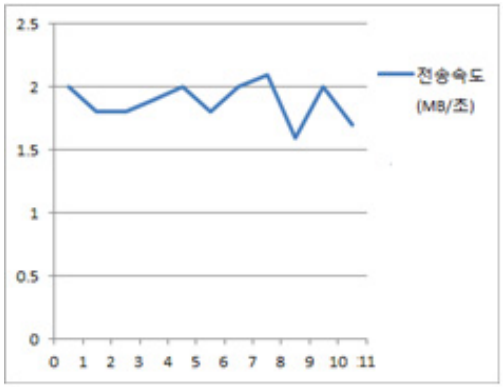
4.1 시스템 구동결과

관리자 프로그램의 캡처기능을 활성화 시킨 후에 정상적으로 AP를 통과하는 패킷이 프로그램 상으로 전달되는 것이 확인되었고, 누락된 패킷을 확인하고자 무선으로 연결된 PC상에서 와이어 샤크 프로그램을 사용하여 각각의 패킷을 5초 정도의 표본을 추출하여 확인해본 결과 해당 PC로 전송되고 있는 패킷이 중간에 누락 없이 전달되는 것이 확인 되었다.

4.2 패킷 캡처에 따른 시스템 효율

AP내에서 패킷이 복사되어 관리자 PC로 재전송이 되는 구조로 인하여 오버헤드로 인한 전체적인 시스템 성능 하락이나 확실한 감시가 행해지지 않을 수 있기에 감시가 활성화 된 상태에서의 전송비율을 확인해보았다.

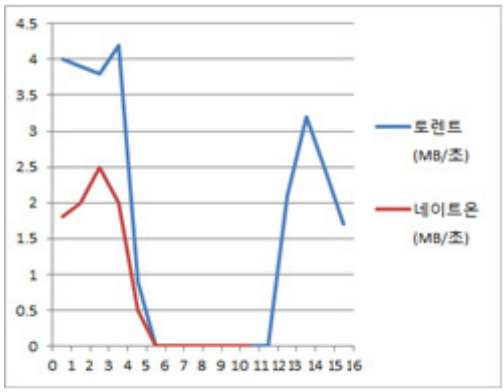
그림 3은 AP에 무선으로 연결된 장치의 다운로드 프로그램(토렌트)의 10초간의 전송을 나타낸 그래프이다. 처음 확인을 시작하고 5초 후 관리자 PC에서 트래픽 감시 기능이 켜진 후에 5초간 더 측정하였다. 측정결과 5초 이후에 패킷 캡처를 활성화시켰음에도 불구하고 관리자PC로 전송으로 인한 속도하락이나 전체적인 성능하락이 보이지는 않았다.



(그림 3) 프로그램 다운로드 속도

4.3 트래픽 차단 후의 속도변화

AP에서 트래픽 차단 후 속도변화를 측정해 보았다. 그림 4를 보면 각 프로그램별 결과가 다르게 나올 수 있기에 한 가지 프로그램으로 실험 하지 않고 메신저를 통한 파일 전송 기능, 토렌트를 이용한 파일 다운로드를 동시에 사용하는 상태에서 차단과 재개를 통해 성능을 측정하였다. 그림 4에서 3초가 되는 시점에 차단기능을 동작시켰고 토렌트와 메신저의 파일 전송 기능이 모두 중단되는 것을 확인하였다. 그리고 10초 시점에서 재개를 시킨 결과 메신저는 계속적인 재전송이 되지 않아 일단 5초 이후로 전송이 중단되었고 토렌트의 파일전송은 정상적으로 재개가 된 것을 확인하였다.



(그림 4) 차단기능 작동시의 속도변화

5. 결론

본 논문에서는 네트워크 트래픽 모니터링 시스템을 구현하기 위하여 libpcap을 이용한 패킷 캡처 프로그램을 Embedded Linux 시스템 상에서 구현하고 패킷을 캡처하여 결과를 보였다. 그러나 단순한 패킷 캡처만으로 네트워크 전체에 걸친 트래픽 모니터링 결과를 보이는 데는 부족함이 있다.

여러 가지 모니터링 연구를 위해서 운영체제, 네트워크, 보안, 데이터베이스, 프로그래밍 등 다양한 기술이 필요하다. 또한 앞으로는 시스템 자원들에 대한 트래픽 측정도 요구된다. 새로운 기술을 익히는 것도 중요하지만 현재 사용하고 있는 것도 제대로 사용하고 있는지 다시 한 번 점검해보고 운영체제와 TCP/IP등 기본기가 되는 것들에 대해서도 깊은 연구가 필요할 것이다. 네트워크 트래픽 모니터링이라는 것이 단지 시스템 설정 몇 개 바꾸어서 되는 것들이 아니라 그만큼 시스템 구성 전반에 대한 이해와 효율적인 설계가 필요한 것이고 운영체제가 작동하는 원리, 등 근본적인 통신 원리에 대한 이해가 반드시 필요할 것이다.

감사의 글

이 논문은 2010년도 정부(교육과학기술부)의 재원으로 한국연구재단의 지원을 받아 수행된 기초연구사업임 (No. 2010-0025495).

참고문헌

[1] 정진욱, 변옥환, 이재광 공역, “TCP/IP 네트워크”, 진영사, 1999.

[2] E. Comer, “TCP/IP 인터네트워킹”, 그린 출판사, 1999.

[3] 정중기, “인터넷 프로토콜 핵심 가이드”, 한빛미디어, 2000.

[4] 박창민, “TCP/IP 네트워크 관리”, 한빛미디어, 1999.

[5] 노병혁, 오성진, “유닉스 프로그래밍”, 사이버출판사, 1997.

[6] 네트워크 해킹/트래픽 완전 해결법 NTOP 사용하기, Linux@Work 2000. 9.

[7] NTOP User’s Guide, http://www.ntop.org, 2005.

[8] UNIX 피해 시스템 분석 및 침입자 모니터링: Part I v1.0, http://www.certcc.or.kr/paper/tr2001/tr2001-03/Scene-of-the-Crime.pdf, 2005.