실시간 망 관리를 위한 패킷 분석 시스템의 설계 및 구현

°정상준*, 권영헌**, 최혁수*, 이정협*, 김종근*
* 영남대학교 컴퓨터공학과
** 세경대학 컴퓨터정보통신과

Design and Implementation of Packet Analysis System for a Realtime Network Management

°Sangjoon Jung*, Younghun Kwon**, Hucksu Choi*, Junghyup Lee*, Chonggun Kim**
* Dept. of Computer Engineering, Yeungnam University
** Dept. of Computer Information & Network, Seakyung College

요 약

본 논문에서는 실시간 성능 관리를 위해 패킷 분석 시스템을 설계하고 구현하였다. 기존의 MIB 정보를 이용한 망 관리에서는 관리국의 주기적인 요청으로 각 에이전트의 MIB 정보를 가져와 분석하는 방식으로, 실시간 감시에는 적합하지 않은 단점이 있다. 따라서, 본 논문에서는 실시간 트래픽 감시를 위해 시스템을 설계하고 구현하였다. 제안된 시스템은 트래픽 상태를 감시하는 모니터링 시스템과 관측된 트래픽을 보여주는 인터페이스 부분으로 나눌 수 있다. 모니터링 시스템은 각 노드의 트래픽을 감시하여 각 패킷별로 구분하여 사용자 인터페이스에 넘겨주게 되며, 이를 사용자 인터페이스에서는 수치형 자료로 표시하거나, 범주형 자료인 그래프로 나타내게 된다. 이 시스템은 각 노드의 부하 여부를 감시하여, 비정상적인 트래픽의 폭주를 발견하게 되면 분석 모듈의 작동에 의해 해킹을 비롯한 네트워크 장애를 감지할 수 있다. 이는 실시간 망 관리의 중요한 기본 기술로 여러 분야에 활용할 수 있다.

1. 서론

현재 전세계적으로 다양한 통신망이 구축, 운용되고 있으며, 이들 통신망을 이용하여 다양한 종류의 통신을 서비스하고 있다. 이러한 통신 서비스에 대하여 가입자들은 고품질의 서비스를 요구하고 있어, 통신망 운용의 관리는 필수적이다.

일반적으로 통신 장비에 대한 관리를 수행하기 위해서 인터넷에서는 SNMP(Simple Network Management Protocol)를 사용하고 있다[1-2]. SNMP는 관리 시스템과 피관리 시스템인 에이전트 간의 관리 정보를 교환하기 위해 사용되는 프로토콜이다. SNMP에서 교환되는 정보는 MIB(Management Information Base)라는 관리 정보 집합을 사용하고 있다[1-3]. 관리 정보 집합인 MIB는 해당 피관리 시스템에 대한 정보 수집의 결정적 단서가 되는 것으로 이를 기반으로 하여 모든 망 관리 시스템들이 개발되고 있다[3]. 하지만, MIB를 이용할 경우, 에이전트에서 수집한 정보를 기반으로 해서 관리가 이루어지기 때문에, 네트워크의 상태를 정확하게 측정하기 어렵다. 주기적인 네트워크의 상태를 확인하고자 하는 경우 기존의 SNMP를 기반으로 한 관리 시스템에서 일정 시간 간격마다 관리국이 에이전트를 폴링(Polling)하여 원하는 정보를 얻어야만 한다[4]. 하지만, 이러한 방법은 네트워크 상에서 폴링으로 인한 통신량 증가를 발생시키게 되고 때로는 폴링을 위한 요구 횟수가 많아져, 정보 전송 시간이 지연되는 경우도 발생된다. 이러한 방법은 실시간 트래픽 분석이 아닌 누적 데이터의 분석을 기반으로 한 망 관리 체계이므로 고속화, 대용량화 된 근래의 네트워크의 관리에 한계점을 보인다. 해킹 또는 서비스 시스템의 다운 등의 현상은 매우 짧은 시간에 발생하여 네트워크를 마비되는 현상까지 보이고 있어, 이를 대비한 정보 수집하고 보수할 수 있는 체계의 도구도 필요하다. 본 논문에서는 노드 기반으로 실시간 네트워크 관리 체계 하에서 보다 빠른 시스템 분석을 위한 실시간 패킷 모니터링 시스템을 제안하고 구현한다. 본 시스템은 단순하고 간단한 원리와 작동 구조를 가지고 있지만, 다양한 분석 모듈의 개발을 통해 손쉽게 확장할 수 있다는 특성을 가진다.

2. 관련연구

2.1 SNMP기반의 망 관리 시스템

기존의 망 관리 시스템에서의 관리 기법은 네트워크 자원 내의 에이전트가 수집한 MIB(Management Information Base) 정보를 관리국(Management Station)이 수집하여 사용자 인터페이스에 보여 줌으로써 네트워크 관리를 수행한다[1-2].

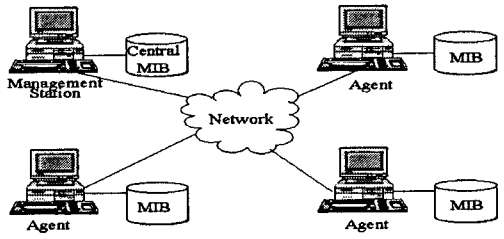

그림 1 네트워크 관리 기능 구조

이와 같은 방법은 모든 네트워크 구성 요소가 SNMP 에이전트를 탑재하고 있어야 하고, MIB 정보를 갖는 호스트에 한해 관리를 수행할 수 있다. 또한, 관리 정보를 모으기 위해서는 관리국이 에이전트에 주기적으로 정보를 요청(Polling)하여 에이전트가 수집한 데이터를 관리국에 제공함으로써, 네트워크 관리가 수행된다. 따라서, 네트워크 상에서 관리국에 의한 폴링은 통신량의 증가를 가져오고, 때로는 폴링을 위한 요구 횟수가 많아져 정보 전송 시간이 지연되는 경우도 발생한다[4]. 따라서, 실제 요구된 시간에 에이전트로부터 정보를 얻지 못하는 단점이 발생할 수도 있다.

2.2 RMON 기반의 망 관리 시스템

원격 세그먼트에 원격 관리 장치(Probe)를 설치하여 관리자에게 관리 정보를 제공하는 시스템으로 원격 네트워크 감시 MIB는 이 Probe에 보내지거나 이 Probe에서 보내 온 관리 정보를 바탕으로 네트워크 관리가 이루어진다[1-2]. 이는 SNMP 기반의 시스템의 개량된 형태로, 에이전트의 역할이 단순 수집에다 패킷을 분석할 수 있는 기능이 강화된 일종의 형태이다.

2.3 실시간 망 관리 방법의 필요성

기존의 네트워크 관리 시스템이 가지는 폴링(Polling)의 요청으로 인한 통신량 증가란 단점을 보완하기 위해서는 요구 메시지를 최소화하는 방법으로 네트워크의 관리가 이루어지는 것이 좋다[4]. 네트워크 관리국에서 실시간으로 네트워크 내의 패킷들을 받아들여 어떤 정보가 흘러가는가에 대한 감시를 수행하면 위에서 언급한 통신 요구 시간 및 수집 시간을 최소화할 수 있다. 본 논문에서는 SNMP 프로토콜을 이용하여 네트워크 감시 및 관리를 수행하는 것이 아니라, 특정 서브네트워크에 모니터링 시스템을 설치하여, 네트워크 내의 패킷을 읽어들여, 분석하고, 나아가 예측할 수 있는 시스템을 구현하고자 한다.

3. 실시간 패킷 분석 시스템의 설계

3.1 네트워크 분석 모델

네트워크를 모니터링하기 위해 특정 서브네트워크에 들어오는 모든 패킷들을 감지해서 분석할 수 있는 패킷 분석 시스템이 필요하다. 프로토콜별 분석과 패킷 헤더들의 분석을 통하여 네트워크의 상태 등을 알 수 있고, 서비스되는 패킷들의 양을 분석하여 패킷 폭주로 인한 네트워크의 장애를 알 수 있으며, 보안의 허점 유무를 파악하고, 현재의 네트워크 상태를 측정하고 과거의 기록을 토대로 앞으로의 네트워크 문제를 사전에 예견하고 제거하기 위한 시스템이 필요하다. 그림 2는 서브네트워크 내의 네트워크 분석 시스템의 위치를 보여주고 있다.

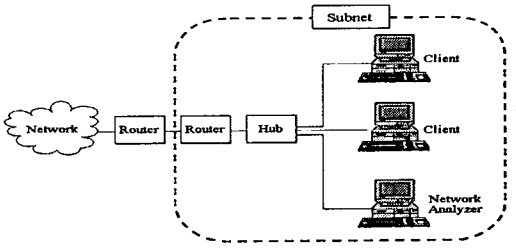

그림 2 네트워크 감시 및 분석 시스템의 위치

3.1.1 실시간 패킷 수집 구조

네트워크 상에서 돌아다니는 패킷들을 받는 패킷 드라이버는 모든 패킷들을 수집(Capturing)하여 분석기로 보내는 모듈과, 분석된 패킷들을 사용자 인터페이스에 보여주는 모듈로 나눈다. 패킷 수집 모듈 컨트롤러(Controller)는 표 1과 같이 4개의 필터를 가진다. 패킷 수집을 위한 컨트롤러 중 일반적인 수집 방법은 네트워크 내의 모든 패킷을 수집할 수 있는 PROMISCUOUS 방법으로 패킷을 모은다.

표 1 패킷 드라이버가 갖는 필터

구 분	역 할
PROMISCUOUS	들어오는 모든 패킷들을 받아들인다.
BROADCAST	Broadcast 패킷들을 받아들인다.
MULTICAST	Multicast 패킷들을 받아들인다.
INDIVISUAL	네트워크 MAC 어드레스와 일치하는 패킷들만 받아들인다.

패킷 드라이버의 구조는 4개의 계층으로 이루어지며 그 구성 요소로는 응용, 사용자 DLL, 패킷 드라이버 DLL, 패킷 드라이버 VxD이다. 응용 단계는 MFC로 구현되어 사용자에게 분석 결과를 보여주게 되며, 패킷 드라이버 DLL은 실제 패킷 드라이버의 대부분을 담당하여, 다른 어플리케이션에게 API를 제공한다. 내부적으로는 IOCTL 인터페이스를 이용해서 VxD와의 통신을 수행한다. 상위의 사용자 DLL은 Win 32 Application이 사용 가능한 API를 제공하고, MFC로 만든 확장 DLL임으로 Visual C++에서 사용하기에 최적화되어 있다. 패킷 드라이버 DLL과 통신하여 상위 Application에게 정보를 전송하게 된다. 그 구조는 그림 3과 같다.

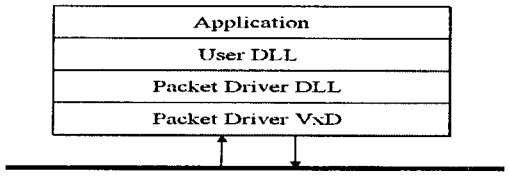

그림 3 패킷 드라이버의 구조

3.1.2 실시간 패킷 분석

본 논문에서 제안하는 실시간 패킷 분석 모델은 패킷에 포함된 정보를 2개의 레벨로 나누어 분석하며 하위 레벨에서 기본적인 분석을 하게 되고 상위 레벨에서는 어플리케이션에 가까운 분석을 하게 된다. 상위레벨에서의 분석은 네트워크 침입 및 어플리케이션에 대한 해킹 등의 정보를 분석할 수 있으며, 하위레벨에서의 분석은 네트워크 공격, 네트워크 마비에 대한 정보를 분석할 수 있어 효과적인 네트워크 감시가 가능하다. 부가적으로 상위 레벨 분석기는 향후에 지속적인 연구를 통해 패턴별로 추가될 수 있는 구조로 되어 있다. 이를 그림 4에 보인다.

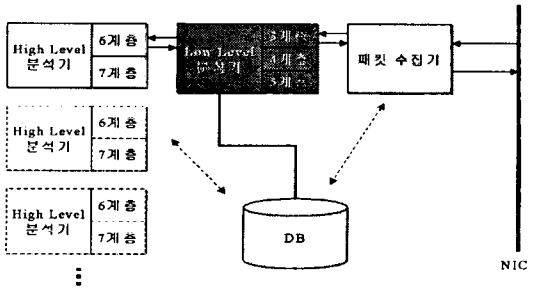

그림 4 패킷 분석 시스템의 구조

3.2 패킷 수집/분석 처리과정

패킷은 패킷드라이버를 통해 패킷 수집기에 전달된다. 패킷 수집기는 지속적으로 패킷을 수집하게 되고 이 수집된 패킷에 대한 정보를 하위레벨 분석기에 전달하여 기본적인 분석을 수행한다. 문제가 없을 경우에는 이 패턴을 반복 수행 하지만 문제가 있는 경우에는 분석에 대한 결과 정보 디스플레이하고 상황에 따라서 다른 노드 및 전체적인 관리 시스템으로 정보를 전송하게 된다. 하위레벨에서 분석한 정보는 주기적으로 모아져서 상위레벨에 전달되고 상위레벨의 분석기도 하위레벨 분석기와 같은 방식으로 작동하게 된다.

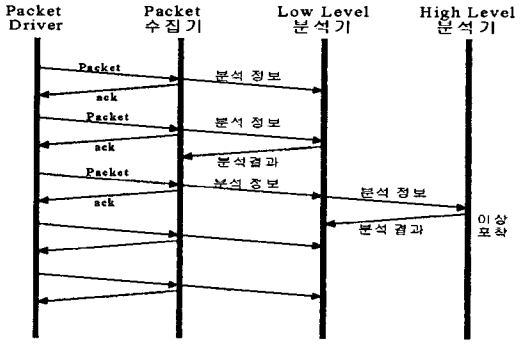

그림 5 패킷 수집 및 분석 처리 과정

4. 실시간 패킷 분석 시스템의 사양

4.1 구현 환경

본 시스템은 Windows98을 운영체제로 하는 IBM 호환 PC(Intel Pentium III 450MHz)에서 MS Visual C++ 6.0을 이용하여 구현하였다.

4.2 시스템의 기능

네트워크의 속도가 갑자기 느려질 경우 이때 네트워크의 프로토콜 별 Traffic을 분석하거나 어떤 네트워크 프로토콜의 사용이 많은지를 알 수 있다. 보통 FTP를 사용하는 컴퓨터에 의해 네트워크 속도의 저하가 발생될 확률이 높고 요즘은 네트워크 게임에 폭발적인 증가로 인한 속도의 저하도 발생한다. 본 시스템이 가지는 기능은 아래와 같이 정의할 수 있다.

- 통계(Statistics) : 한 Subnet에서 발생한 패킷/바이트 수, Broadcast/Multicast 수, 충돌(collision) 수 및 패킷 길이별 수 그리고 각종 에러(프래그먼트, CRC Alignment , jabber, 길이미달, 길이초과)에 대한 통계를 제공한다.
- 이력(Statistics) : 관리자가 설정한 시간 간격내에 발생한 각종 트래픽 및 에러에 대한 정보를 제공한다.
- 트래픽 매트릭스(Matrix): Data Link Layer, 즉 MAC Address를 기준으로 두 호스트간에 발생한 트래픽 및 에러에 대한 정보를 수집한다. 이 정보를

이용해서 특정 호스트에 가장 많은 이용자가 누구인지를 어느 정도는 알 수 있다.

● 패킷 수집(Packet Capture) : 네트워크 상에 발생한 패킷을 수집해서 관리자가 분석할 수 있도록 한다. 관리자는 패킷의 전부 또는 일정한 길이만 보관하고 읽어올 수 있도록 설정이 가능하다.

● User History : 모든 내용을 관리자가 설정한 상황 (기간별, Packet Size별, Protocol별 등)대로 History Table을 만들 수 있다.

4.3 구현 결과

모니터링이 시작되면 네트워크를 이루고 있는 패킷 드라이버의 이름과 아답터 및 패킷 개수를 파악하고, 실제 패킷 별로 분류하여 사용자에게 보여준다. 그림 6은 모니터링하고 있는 상황을 보여주는 출력 결과이다.

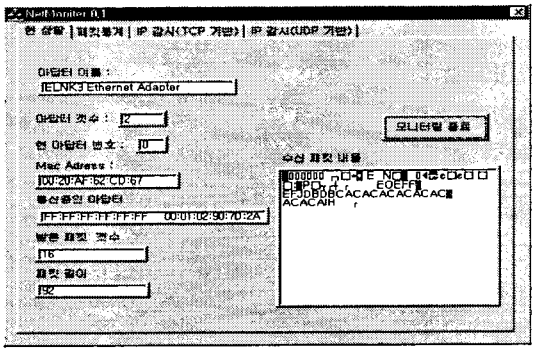

그림 6 패킷 감시 인터페이스

위 정보를 근거로 한 실제적인 패킷 정보는 그림 7에서 볼 수 있다. 패킷 별로 분류하고, 패킷의 양을 측정하게 된다. 수치적인 데이터는 그래프로 보여, 특정 임계값을 설정할 경우, 위험 상황 인식도 가능하다.

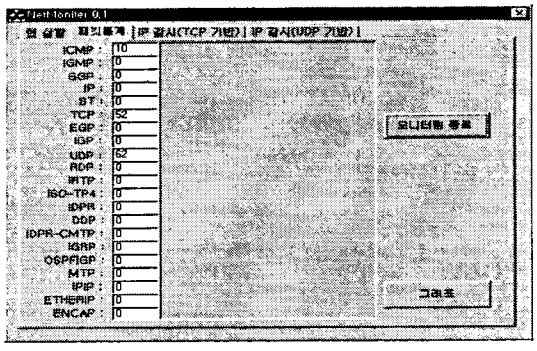

그림 7 패킷 감시 결과 출력 인터페이스

패킷 감시의 결과를 그래프로 보여, 사용자가 판단을 용이하도록 한다. 그림 8은 패킷 통계량을 그래프로 출력한 화면이다.

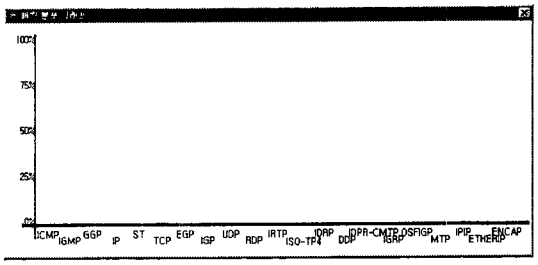

그림 8 측정된 데이터의 그래프 출력 화면

5. 결론

본 논문에서 설계하고 구현한 결과는 실시간 트래픽 캡쳐 및 분석을 목적으로 하고 있다. 본 시스템은 각 노드의 부하 여부를 감시하여, 비정상적인 트래픽의 폭주, 네트워크의 다운 등과 같은 비정상적인 작동을 발견하게 되면 분석 모듈의 작동에 의해 해킹을 비롯한 네트워크 장애를 감지할 수 있다. 이는 노드 기반 실시간 망 관리의 한가지 방법으로 여러 분야에 활용할 수 있다. 추후 연구과제로는 시계열 예측기법을 이용하여, 과거에 분석된 자료와, 현재의 측정된 데이터 값을 바탕으로 미래에 트래픽 유추를 예측할 수 있는 알고리즘을 적용하여, 예측 기능이 추가된 지능형 분석 시스템을 구현하고자 한다.

[참고문헌]

[1] William Stallings, "SNMP, SNMPv2, SNMPv3, and RMON 1 and 2", Addison Wesley, 1999.

[2] Mark A. Miller, "Managing Internetworks SNMP", M&T books, 1998.

[3] Kyung Hyu Lee, "An Agent-Manager Scheme for the Integrated Transfort Network Management", IEEE International Conference on Communications, pp.1017-1021, 1999,6.

[4] 안성진, 정진욱, "SNMP MIB-II를 이용한 인터네트 분석 파라미터계산 알고리즘에 관한 연구", 한국정보처리학회 논문지 제5권 제8호, pp.2102-2116, 1998,8.

[5] 김동수, 정태명, "실시간 네트워크 관리를 위한 SNMP 확장에 관한 연구", 한국정보처리학회 논문지 제6권 제2호, pp.449-458, 1999,2.

[6] 이강원, 김태윤, "효율적인 통신망 설계를 위한 예측 시스템 설계", 한국정보과학회 논문지, 제25권 제1호, pp.76-82, 1998,1.