SNMP 를 이용한 실시간 장애관리 시스템의 개발

정아주, 박준상, 이상우, 김명섭
고려대학교 컴퓨터정보학과
e-mail : {yesvery, runtoyou, ilovejam, tmskim}@korea.ac.kr

Real-time Fault Management System based on SNMP

Ah-Joo Jung, Jun-Sang Park, Sang-Woo Lee, Myung-Sup Kim
Dept. of Computer and Information Science, Korea University

요 약

최근 인터넷의 급속한 발전과 컴퓨터 및 네트워크 기술이 점차 보편화 됨에 따라 다양한 정보를 제공하는 서비스들이 증가하고, 서비스를 제공하는 시스템 또한 사용 영역이 지속적으로 확대되고 있다. 이렇듯 다양한 서비스를 제공하는 시스템을 효율적으로 관리하고 여러 이유로 발생하는 장애를 능동적으로 감지하여 해결할 수 있는 기술은 최근 중요한 연구 이슈 중 하나가 되었다. 본 논문은 관리자의 편의성을 고려하여 웹 기반 방식의 관리 시스템을 제공함과 동시에 실시간 장애 관리 시스템을 설계 및 구현하였다. 또한 장애가 발생한 시스템을 정확하고 신속히 감지하기 위해 실시간으로 장애 정보를 수집 후, 이를 관리자에게 SMS 서비스로 알려주는 시스템을 제안한다.

1. 서론

오늘날 점점 복잡하고 다양해지는 서비스 속에서 분산되어 있는 많은 시스템들을 체계적이고 효율적으로 관리하는 것은 중요한 일이다. 하지만 다수의 분산된 시스템들을 관리자가 직접 관리하기에는 많은 인적자원 및 비용이 필요하다. 따라서, 이를 해결하기 위해 실시간 모니터링 시스템의 필요성이 대두되고 있다.

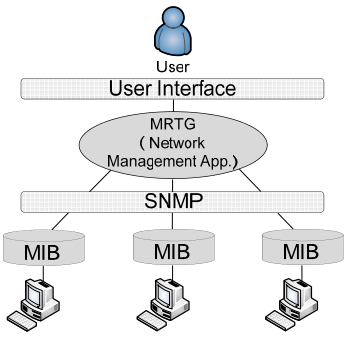
실시간 모니터링 시스템은 분산된 각 서버들로부터 상태 정보를 수집하여 그 정보를 가공 후 수치나 그래프 등을 통해 제공하는 것이 기본 목적이다. 하지만 실시간 모니터링 시스템은 해당 시스템의 하드웨어와 소프트웨어의 내적 요인에 의해 언제든지 장애가 발생할 수 있으며, 서버의 폭주나 정전사고와 같은 외적 요인도 배제할 수 없는 것 또한 문제이다. 비효율적으로 수행되는 시스템은 사용자의 관점에서 치명적인 문제가 될 수 있기 때문에 시스템의 흐름과 상태를 정확히 모니터링하며 장애가 발생 하였을 시 해당 장애 정보를 즉각 관리자에게 알려줄 수 있어야 한다.

본 논문에서는 시스템의 장애상태를 정확히 감지하여 관리자에게 해당 장애 정보를 SMS(Short Message System)로 통지 해주는 장애 관리 시스템을 설계 및 구현하였다.

기존 연구[1-2]에서는 시스템의 감시 및 장애발생 여부 확인, SMS 를 이용한 결과 보고, 저장 및 관리, 웹 페이지를 통한 결과 조회 등 실시간 장애 관리 시스템의 해당 기능들을 충족하고 있다. 하지만 장애 여부 상태를 파악하는 기존 기술들은 그 내용이 불충분하다. 또한 장애 여부를 확인하는 것 만으로는 정확한 장애원인을 파악할 수 없고 비정상적인 상황이 발생 하더라도 문제를 인지하고 대처하기까지 많은 노력과 시간이 필요하게 된다. 따라서 본 논문에서는 다양한 장애 여부를 파악할 수 있는 기술들을 소개하고, SMS 서비스에 장애발생 원인에 대한 모듈을 추가함으로서 관리자로 하여금 즉각적인 장애 조치를 취하게 하는 방법을 기술 한다. 이로써 본 논문에서 제시하는 시스템은 여러 장애상황에 효율적으로 대처할 수 있는 장점을 가지고, 모니터링 하는 시스템의 안정성과 높은 신뢰성을 제공한다.

논문의 구성은 다음과 같다. 서론에 이어 2 장에서는 실시간 모니터링 시스템의 구조에 대해 알아보고, 3 장에서는 실시간 장애 관리 시스템의 설계 및 시스템 구현을 위한 알고리즘 기술들을 설명한다. 4 장에서는 실제 구현 예시를 보고 5 장에서 제안한 시스템의 문제점과 향후 과제를 언급하며 결론을 맺는다.

2. 실시간 장애 관리 시스템의 구조



(그림 1) 실시간 모니터링 시스템 구조

* 이 논문은 2007 년 정부(교육인적자원부)의 재원으로 한국학술진흥재단의 지원을 받아 수행된 연구임.(KRF-2007-331-D00387)

실시간 모니터링 시스템은 (그림 1)과 같은 서비스 처리로 구성된다.

시스템은 사용자에게 웹 인터페이스를 제공하고 서버의 상황을 실시간으로 모니터링 하기 위해 MRTG(Multi-Router Traffic Grapher)[3]를 사용한다.

MRTG 는 SNMP(Simple Network Management Protocol)[4]라는 프로토콜을 사용하며, SNMP 는 MIB(Management Information Base)[5] 자원의 객체데이터베이스에 정의된 값들을 가져오거나 설정이 가능하다. 이 때문에 MRTG 는 SNMP 를 통해 자원의 분석 및 실시간으로 서버 모니터링을 가능하게 해준다.

실시간 모니터링 시스템은 서버/클라이언트 모델을 기반으로, MRTG 가 설치되어 운용되며 모니터링 결과를 이미지로 생성하여 HTML 페이지로 보여주기 때문에 어디서든지 정보를 공유할 수 있는 웹 페이지의 장점을 적극 활용한다.

또한 실시간 처리를 위해 실행 명령을 리눅스 쉘 파일에 저장하고 Crontab 에 등록하여 5 분마다 MRTG 를 갱신하도록 한다. 갱신주기는 컴퓨터의 성능에 따라 Interval 값을 다르게 할 수 있다. 각 시스템들은 5 분을 주기로 네트워크 트래픽, CPU 사용률, 메모리 사용률, 디스크 사용률 등을 UCD-SNMP[6]가 제공하는 MIB 값을 가져와 그래프와 웹 페이지로 자동 생성해준다. MRTG 를 이용한 시스템 분석 작업 내용과 관련한 MIB 정보는 다음과 같다.

<표 1> 네트워크 트래픽관련 MIB 정보

Inbps	mib-2.interfaces.ifTable.ifEntry.ifInOctets
outbps	mib-2.interfaces.ifTable.ifEntry.ifOutOctets
Memory Real Total	enterprises.ucdavis.memory.memTotalReal
Memory Real Avail	enterprises.ucdavis.memory.memAvailReal
CPU	enterprises.ucdavis.laTable.laEntry.laLoad
Disk	enterprises.ucdavis.dskTable.dskEntry.dskUsed

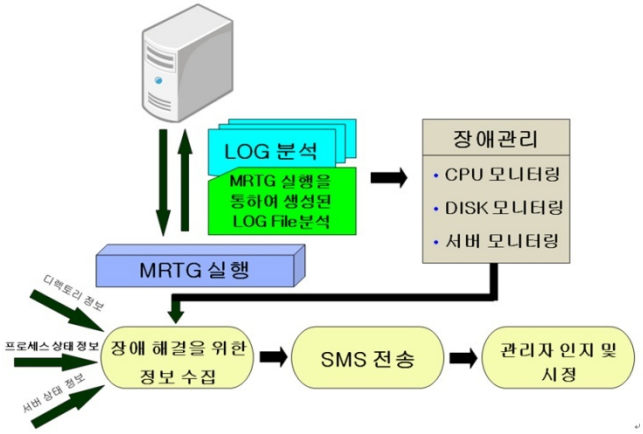
MRTG 는 모니터링뿐만 아니라 MIB 값을 이용하여 외부 프로그램을 사용할 수 있기 때문에 다양한 변형이 가능하고 MRTG 가 생성하는 로그값을 가공하여 새로운 데이터베이스를 구축하여 응용할 수도 있다.

실시간 장애 관리 시스템은 이러한 앞서 설명한 실시간 모니터링 시스템의 장점을 이용하여 시스템의 장애나 부하 등 다양한 서비스 검사를 수행한다.

3. 알고리즘

실시간 장애 관리 시스템은 (그림 2)와 같은 구조를 갖는다. 관리시스템은 MRTG 실행을 통해서 생성된 각 장치 별 로그 파일을 읽어 들여 이를 분석하고, 분석된 자료를 바탕으로 장애 판단 알고리즘을 통하여 시스템의 장애 발생 여부를 확인 한다. 시스템의 장애 상태가 확인된 경우 서버는 장애를 해결하기 위한 정보를 수집한다. 그리고 해당 정보를 관리자에게 SMS 를 통해 전송하고 관리자는 장애발생 정보를 확인 후 신속한 대처를 할 수 있게 한다.

이 장에서는 SMS 전송 기능을 포함하여 앞서 설명한 시스템을 구현하기 위해 CPU 장애 판단, 디스크 장애 판단, 시스템 동작 유무에 대한 알고리즘 기술들을 제안 한다. 또한 제시한 알고리즘을 통하여 장애 상태를 확인한 후 관리자가 이를 쉽게 해결하기 위해 해당 장애에 대한 정보(디렉토리 목록, CPU 점유율이 높은 프로세서 정보)를 수집하여 SMS 메시지를 전송한다.



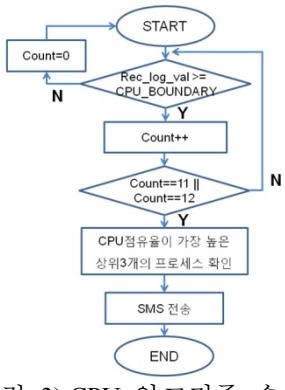
(그림 2) 시스템 구성도

3.1 CPU 장애 상태 판단 알고리즘

CPU 의 장애는 CPU 의 점유율이 비정상적인 수치를 보이며 불안정할 때 장애의 원인으로 판단한다.

(그림 3)과 같이 CPU 의 최근 로그값(Rec_log_val)이 CPU 임계값(CPU_BOUNDARY) 보다 높은 수치를 가지고, 이 수치가 지속적으로 높아진다면 비정상적인 상태임을 확인할 수 있다. 이 경우 Count 값을 증가시켜 장애가 일어난 시점부터 지속시간을 체크하여 Count 값이 11(55 분)이 될 경우 경고 SMS 를 전송하고, 12(60 분) 가 된 경우에는 장애 발생 SMS 를 전송한다.

SMS 메시지는 관리자의 장애 상태에 대한 상세한 정보를 위해 CPU 점유율이 높은 상위 3 개 프로세스들의 이름, CPU 사용률에 대한 정보를 포함한다. 프로세서의 정보는 관리 대상 시스템의 proc 파일 시스템을 이용하며, proc 은 프로세서의 정보를 가지고 있는 파일 시스템으로서 실행중인 프로세서의 자세한 정보, 즉 CPU 점유율에 대한 상태정보를 얻을 수 있다.



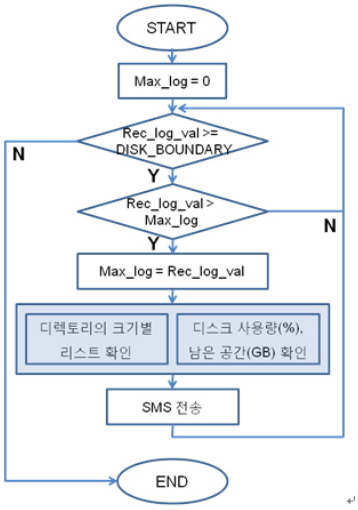
(그림 3) CPU 알고리즘 순서도

이러한 프로세서들의 상세 정보를 제공함으로써 관리자는 어떤 프로세서에 의해 CPU 가 비정상적인 작동을 했는지 문제 원인을 신속하게 파악하여 처리할 수 있다.

3.2 디스크 장애 상태 판단 알고리즘

(그림 4)에서와 같이 디스크 사용률(Rec_log_val)이 임계값(DISK_BOUNDARY)의 범위를 벗어날 경우 디스크 Full 이 될 가능성이 높으므로 경고 SMS 메시지를 전송 한다. 이후 90% 이상의 가장 최근 로그 값(Rec_log_val)을 이전 최대 로그 값(Max_log)과 비교하여 값이 증가 할 때마다 SMS 메시지를 전송하고 최대 로그값으로 대체한다. 최근 로그값이 90% 미만으로 내려 갈 경우 최대 로그값은 0 으로 초기화 된다.

디스크 관련 SMS 메시지는 디스크 용량에 큰 영향을 미치는 디렉토리의 목록과 디스크의 사용량(%), 남은 디스크 공간(GB)을 포함 한다. 디렉토리의 목록은 Ftw 함수를 이용하여 루트 디렉토리에서 부터 레벨 3 위치까지 탐색 후 레벨 3 에 있는 모든 디렉토리의 정보를 얻어 그 중 상위 3 개의 목록을 SMS 로 전송한다. 또한 디스크 사용량은 dskUsed 라는 MIB 값을 이용하여 % 단위로 표시하고, 디스크의 남은 공간은 Proc 파일 시스템의 정보를 이용하여 GB 단위로 보여준다.

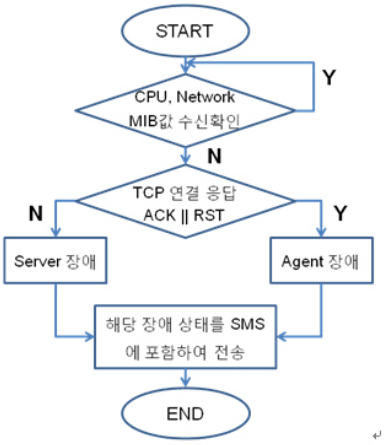


(그림 4) 디스크 알고리즘 순서도

이렇게 디스크의 정보를 관리자에게 제공함으로써 디스크가 Full 이 났을 때 발생하는 문제에 대해 미리 예방할 수 있다.

3.3 관리대상 시스템의 동작 유무 판단 알고리즘

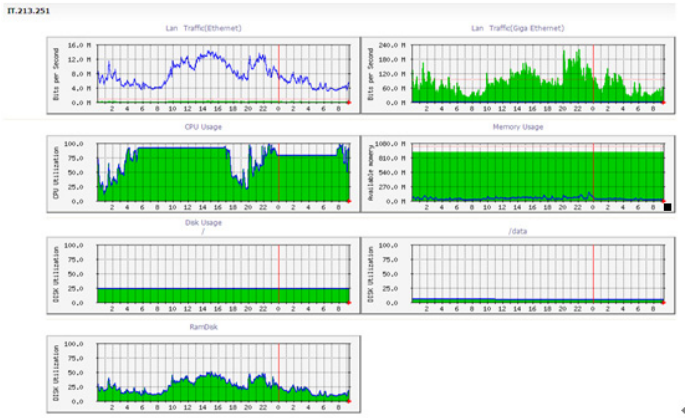
(그림 5)는 시스템 동작 유무를 확인하는 알고리즘의 순서도이다. 관리대상 시스템의 동작 유무를 판단하기 위해 앞서 제시한 두 알고리즘과는 다른 알고리즘이 요구된다. 먼저, 관리대상 시스템이 작동하지 않을 경우의 상황을 알기 위해 해당 관리대상 시스템의 MIB 값을 제대로 가져올 수 있는지를 확인한다. MIB 값의 수신 여부는 값의 변동이 심한 CPU 와 네트워크 트래픽의 MIB 값으로 판단하며, CPU 와 네트워크 트래픽의 MIB 값이 일정시간 동안 변동됨이 없다면 MIB 값을 제대로 가져오지 못함을 뜻한다. 이렇게 두 상태 모두 MIB 값을 가져오지 못한다는 것은 SNMP-Agent 가 작동하지 않을 경우와 시스템이 작동하지 않을 경우가 있다. 이 두 가지 상황의 정확한 판단을 위해 해당 시스템의 동작의 유무를 확인하는데, 시스템 동작의 유무는 Well-known 포트로 TCP 연결요청 후 응답상태를 확인한다. 응답상태에 대해 ACK 또는 RST 플래그를 포함한 패킷을 전송 받으면 서버는 정상적인 상태로 판단하고, 이는 snmp agent 의 문제가 발생한 것으로 처리한다. 반대로 응답이 확인되지 않으면 해당 시스템은 작동하지 않음으로 판단한다. 즉, 단순히 MIB 값 만으로는 SNMP-Agent 의 작동 상태에 대해서만 알 수 있을 뿐 시스템의 작동 여부는 정확히 판단 할 수 없다.



(그림 5) 시스템의 동작 여부 판단 알고리즘 순서도

4. 시스템 구현

4.1 구현환경



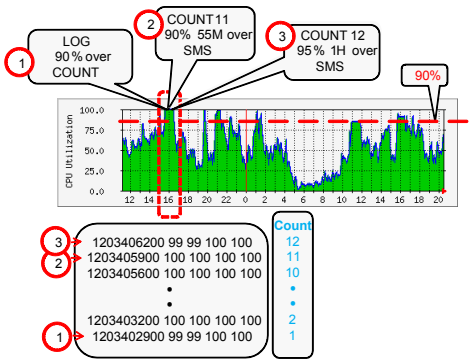
(그림 6) 시스템 모니터링 화면

총 18 대의 시스템에 대한 네트워크 트래픽, CPU 사용률, 메모리 사용률, 디스크 사용률을 웹 서버 환경에서 구축하여 모니터링 하도록 구성하였으며 웹의 구성은 MRTG 에서 표준으로 분석한 결과의 상태변화를 그래프로 보여준다. 각각의 시스템은 5 분 단위로 모니터링하며 장애 상태를 인지한다.

4.2 결과 고찰

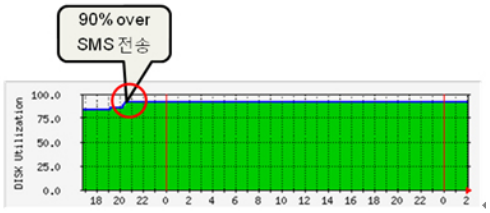
다음은 시스템의 장애발생 상황을 장치 별로 보여준다.

(그림 7)은 CPU 의 장애 상태에 대한 것으로 15 시와 17 시 사이에 CPU 점유율이 90% 이상을 차지하는 동시에 60 분 동안 높은 수치로 지속되고 있음을 나타낸 경우이다. 이 경우 55 분과 60 분에 장애 발생 메시지를 보냄으로써 시스템의 부하를 막고 비정상적으로 실행되는 프로세스를 확인 할 수 있다.



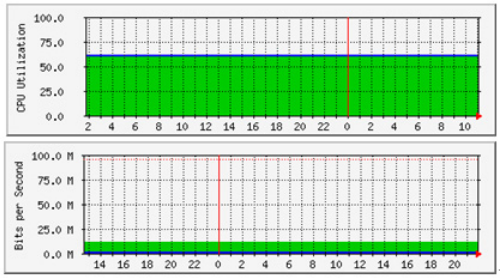
(그림 7) CPU 장애감지

(그림 8)은 디스크 사용률을 보여 주고 있으며 디스크 사용률이 90% 이상인 경우이다. 현 상황에서 디스크를 계속 사용한다면 단시간내에 디스크가 Full 이 될 가능성이 높다. 그러므로 사전에 SMS 메시지를 통해 현재 디스크 상황을 미리 알려준다면 디스크가 Full 이 났을 경우 발생하는 문제를 미리 예방할 수 있다.



(그림 8) 디스크 장애감지

(그림 9)에서는 시스템 또는 SNMP-Agent 가 작동하지 않을 경우의 CPU 와 Network 상태를 보여주고 있다. 그래프가 변동하지 않고 일정한 값만 계속 유지되고 있는데 이것은 작동 상태에 이상이 있다는 것을 나타내며 시스템과 SNMP-Agent 의 이상 유무를 확인하여 관리자에게 장애정보를 알려 빠른 조치를 취할 수 있도록 한다.



(그림 9) Server 상태에 대한 장애감지

장애가 발생하였을 때 관리자에게 발송된 문자 메시지의 전송 기록은 SMS 정보 페이지에서 확인 할 수 있으며, 이 페이지는 장애가 발생한 관리대상 웹 서버의 장애 시점을 알 수 있고 어떤 자원에서 문제가 발생 하였는지, 장애 통보가 어떻게 이루어 졌는지에 대한 정보도 확인할 수 있다

5. 결론

복잡하고 다양한 환경에서 시스템이 얼마만큼의 안정성을 갖고 여러 가지 상황에 효율적으로 대처할 수 있는가 하는 점은 실시간 모니터링 시스템 운영에 직접적인 영향을 끼친다. 이러한 개념을 바탕으로 본 논문에서는 실시간 모니터링 기술을 이용한 장애 관리 시스템을 제시하였다.

실시간 장애관리 시스템은 시스템들의 데이터 흐름과 다양한 이벤트의 변화를 곧바로 감지하여 관리자에게 반영될 수 있도록 설계 되었다. 이는 각각의 시스템에 이상이 생겼을 경우 관리자가 즉시 상태를 파악할 수 있으며 시스템 입장에서 볼 때 여러 상황에 대처할 수 있어 자원을 절약하는 데 큰 장점으로 작용한다. 또한 실시간 모니터링 장애관리 시스템은 관리자가 현재 실시간 모니터링 및 시스템의 장애 발생 파악뿐만이 아니라 장애 상태에 대한 문제를 해결할 수 있어 효율적인 관리에 큰 도움을 준다.

하지만 본 논문에서 제시하는 장애 관리 시스템은 MRTG 를 기본 툴로 사용하며 MRTG 는 최소 5 분 단위의 모니터링 결과만을 제공한다. 하지만 현재의 대용량 네트워크 트래픽과 응용프로그램의 복잡성을 고려했을 때 MRTG 는 신속한 대처가 불가능하다. 따라서 SNMP 를 기반으로 모니터링 시간을 최소화 할 수 있는 시스템을 구축하는 연구를 계획하고있다.

참고문헌

[1] 김승남, 박민현, 한욱표, 정영준, “Real-timed web-Server Fault Management System”, 기초과학연구 17 (2009) 167-182.

[2] 조승환, Implementation of Web-based Service Observation System(SOS), 2005. 09.

[3] Tobias Oetiker, Dave Rand, Multi Router Traffic Grapher, http://www.mrtg.org

[4] William Stallings, SNMP, SNMPv2, SNMPv3, RMON 1 and 2, 3/E, Addison Wesley

[5] RFC-1156 "Management Information Base for Network Management of TCP/IP-based internets", ftp://ftp.isi.edu/in-notes/rfc1156

[6] http://net-snmp.sourceforge.net