# 원격 웹 서버 로깅 시스템의 구현

정기훈[0] 노삼혁
홍익대학교 컴퓨터공학과, 정보컴퓨터공학부

khchong@cs.hongik.ac.kr, samhnoh@hongik.ac.kr

# Implementation Study of a Remote Logging System for Web Servers

Kihun Chong[0] Sam H. Noh
Dept. of Computer Engineering, Computer Science, Information and Computer Engineering, Hongik Univ.

요 약

본 논문에서는 웹 서버에게 부하를 주지 않으면서 네트워크를 통해 원격으로 로깅하는 시스템인 원격 웹 서버 로깅 시스템을 구현하였다. 구현된 로깅 시스템은 웹 서버의 종류와는 관계 없이 로그 데이터를 수집할 수 있으며, 여러 개의 서버 군으로 이루어진 시스템에서도 각 시스템의 웹 서버나 환경과는 관계 없이 독립적으로 로그 데이터를 수집할 수 있다는 장점을 갖고 있다. 뿐만 아니라 웹 서버에게 미치는 오버헤드를 측정해본 결과, 부하를 거의 주지 않음으로써 로깅 시스템으로 인한 웹 서버의 성능 저하를 일으키지 않는다는 것을 알 수 있었다.

## 1. 서론

1990년에 만들어지고, 1993년에는 미국에서 사용하는 인터넷 서비스 중 127위를 차지했던 웹은 1997년 이후로 엄청난 속도로 발전하였으며, 2004년 2월 현재 4천7백만 이상의 도메인에서 웹 서비스를 하고 있다[1]. 이와 같이 기하급수적인 웹의 성장에 힘입어 웹 서버 시스템 또한 폭발적으로 증가하고 있으며, 웹 서비스를 이용하는 사용자도 꾸준히 증가하는 추세이다. 따라서 안정적인 웹 서비스를 위해서는 서버 시스템의 성능 향상도 중요하지만 그에 못지않게 중요한 것이 바로 사용자의 요청 패턴을 정확하게 분석하여 서버 시스템의 성능 향상과 최적화에 따르는 비용을 줄이는 것이다. 이를 위해 일반적으로 가장 많이 사용하는 것이 웹 서버 프로그램의 로그 데이터(Log Data)이다. 하지만 로그 데이터를 기록하는 것은 곧 디스크 쓰기를 의미하기 때문에 사용자의 요청이 증가할수록 기록해야 하는 로그 데이터도 많아지며, 이것은 역설적으로 로그 데이터로 인한 서버 시스템의 성능 저하를 부르게 된다. 따라서 본 논문에서는 웹 서버 소프트웨어의 실행에 부하를 주지 않으면서 웹 서버의 실행 로그를 네트워크를 통해 원격으로 로깅하는 시스템인 http_netlog를 구현하였다.

원격 웹 서버 로깅 시스템이 웹 서버에 얼마나 많은 영향을 미치는가를 조사하기 위하여 리눅스(Linux) 운영체제 상에서 웹 서버 벤치 마크인 WebStone을 이용하여 아파치 웹 서버 시스템에서의 http_netlog가 미치는 오버헤드를 측정하였으며 http_netlog는 아파치 웹 서버에 거의 부담을 주지 않는 것으로 나타났다.

논문의 구성은 다음과 같다. 먼저 2절에서는 관련 연구에 대하여 논한다. 3절에서는 원격 웹 서버 로깅 시스템의 구성 및 구동 방식에 대하여 알아보고 4절에서는 원격 웹 서버 로깅 시스템의 부하를 측정하고 그 결과에 대하여 논한다. 마지막으로 5절에서 결론 및 향후 과제에 대하여 언급한다.

## 2. 관련 연구

서버 시스템의 성능 향상은 보통 단일 서버 시스템 보다는 다수의 서버 군으로 이루어진 서버 시스템 전체의 성능의 향상을 목표로 하고 있으며 가장 대표적인 분야는 웹 클러스터링 서버 시스템이다. 이러한 시스템에서 가장 관심을 갖는 것은 사용자 요청으로 인한 부하를 어떻게 하면 효과적으로 분산하여 시스템의 성능을 극대화 시키는가 이다. 이러한 문제를 해결하기 위하여 제시된 시스템으로 Distributed Cooperative Web Servers[2], LARD[3] 등이 있다.

서버 자체의 성능 향상을 위하여 관심을 갖는 부분은 다시 운영체제영역의 성능 향상과 사용자영역의 성능 향상으로 나눌 수 있는데, 운영체제영역에서 관심을 갖는 것은 디스크 또는 네트워크간의 액세스 시간을 줄이거나 메모리/프로세스의 효과적인 관리를 통하여 웹 서버 시스템의 성능을 극대화 시키는 것을 목표로 연구가 진행되고 있다. 사용자 영역에서는 웹 서버 프로그램의 단독적인 성능 향상을 목표로 하는 것이 대부분이다[4]. 이러한 웹 서버 프로그램의 성능 향상을 위하

여 보통 최적화 시킨 시스템 콜(System Call)을 통한 운영체제와의 통신을 들 수 있으며, 사용자 영역 자체 내에서도 멀티 프로세싱, 멀티 스레딩, 이벤트 드리븐 등 여러 가지 방법을 동원하여 웹 서버 프로그램의 성능 향상을 위하여 효과적인 방법을 모색하고 있다. 실제로 사용자가 가장 많이 사용하는 아파치 웹 서버의 경우 웹 서버의 성능을 향상시키기 위해서 로깅부분까지도 최적화시킨 모습을 볼 수 있다[5].

### 3. 원격 웹 서버 로깅 시스템

이 절에서는 원격 웹 서버 로깅 시스템인 http_netlog에 대하여 자세히 살펴보겠다. 특히 웹 서버와 어떻게 독립적이 되는가에 대하여 실제 시스템에 설치된 모델을 통하여 다루도록 하겠다.

원격 웹 서버 로깅 시스템은 크게 웹 로그를 생성하는 로그 생성기(http_netlog_agent)와 http_netlog_agent에 의해 생성된 데이터를 수집하는 수집기(http_netlog_manager)로 구성된다.

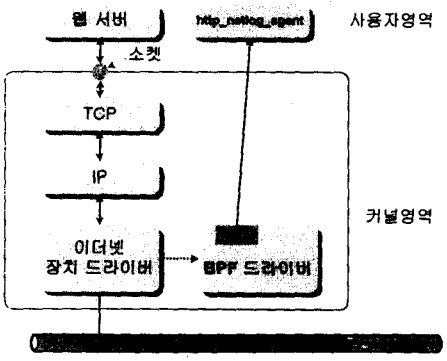


그림 1. http_netlog_agnet의 구조

#### 3.1 로그 생성기(http_netlog_agent)

로그 생성기는 원격 웹 서버 로깅 시스템에서 웹 로그 데이터를 추출해내는 일을 한다. http_netlog_agent가 다른 로그 데이터 추출기와 차이가나는 가장 큰 특징은 웹 서버와 독립적으로 동작한다는 것이다. 즉, 웹 서버 프로그램의 종류와 관계 없이, 웹 서버가 동작을 하는지 여부와 상관 없이 http_netlog_agent를 붙였다 떼었다 할 수 있다. 이러한 것이 가능한 이유는 BPF(BSD Packet Filter)[6]를 이용한 패킷 필터링에 의한 HTTP 프로토콜의 추적때문이며, 그림 1에서 http_netlog_agent의 구성에 대하여 자세히 보여주고 있다.

UNIX시스템에서 기본적으로 지원하는 BPF는 이더넷(Ethernet) 인터페이스를 통과하는 모든 패킷에 대하여 그 복사본을 받는데, 이러한 정보를 사용자가 원하는 데로 필터링을 할 수 있도록 지원한다. 따라서 http_netlog_agent는 HTTP 서비스를 이용하는 모든 패킷에 대해서만 필터링을 하였으며, 연결의 맺음부터 시작하여 연결이 종료될 때까지 모든 과정을 추적하여 그에 따른 HTTP 로그 데이터를 수집할 수 있는 것이다. 그렇기 때문에 웹 서버 프로세스와 독립적으로 동작할 수 있는 것이다.

또한, 커널 영역에서 연결이 실패하는 경우와 같이 웹 서버 프로세스가 알아낼 수 없는 정보까지도 http_netlog_agent는 파악할 수 있기 때문에 웹 서버에서 제공하는 것보다 더욱 자세한 정보를 제공할 수 있다. 그리고 웹 서버에서는 보통 사용자가 요청한 파일의 크기에 대해서만 기록을 하는데 반하여 로그 생성기는 연결 시작부터 종료까지의 모든 과정이 이루어지는 동안 교환된 모든 패킷의 크기를 수집하기 때문에 훨씬 더 실제적인 네트워크의 트래픽(Traffic) 정보를 제공할 수 있다. 더욱이 웹 서버가 남기는 로그와 동일한 형식의 로그 파일을 만들어낼 수 있기 때문에 필요한 목적에 맞게 로그 파일을 생성할 수도 있다.

로그 생성기에게 있어서 가장 중요한 제한은 바로 서버와 네트워크에 최소한의 오버헤드를 주는 것이다. 로그 생성기의 기능이 아무리 좋다고 하더라도 로그 생성기로 인하여 웹 서버 시스템의 성능 저하를 불러 일으킨다면 로깅 시스템의 목표에서 벗어나는 것이기 때문에 http_netlog_agent는 가능한 가벼워야 한다.

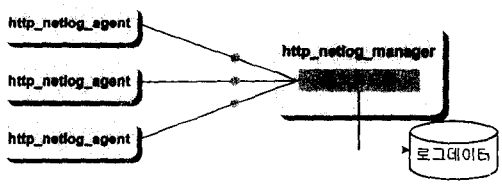


그림 2. 로그 생성기와 로그 수집기의 구성

#### 3.2 로그 수집기(http_netlog_manager)

로그 수집기는 http_netlog_agent가 생성한 로그 데이터를 원격으로 수집하여 저장하는 일을 한다. 로그 수집기는 원격으로 로그 데이터를 수집하기 때문에 같은 웹 서버에서 동작할 수도 있으며 웹 서버와는 독립된 시스템에서도 동작이 가능하다. 따라서 여러 개의 서버로 이루어진 웹 서버 시스템의 경우에도 로그 데이터를 수집하는 것이 가능하며, 자동으로 동기화까지 시켜주기 때문에 관리자 입장에서는 편하게 로그 데이터를 수집할 수 있도록 해준다.

그림 2는 로그 수집기와 로그 생성기와의 구성을 보여주고 있다. 그림 2에서 보는 바와 같이 각 웹 서버 시스템에서 로그 생성기에 의해서 만들어진 로그 데이터는 미리 생성된 로그 수집기와의 커넥션을 통하여 로그 수집기로 전해지는데, http_netlog_manager는 각 서버의 http_netlog_agent가 보내주는 로그 데이터의 시간을 동기화 하기 위하여 타임스탬프(timestamp) 값을 이용한다.

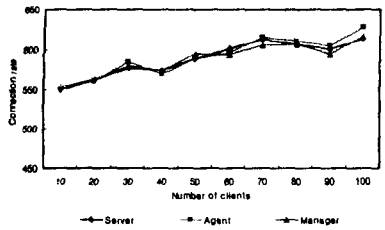


그림 3. 초당 접속률 (connections/sec)

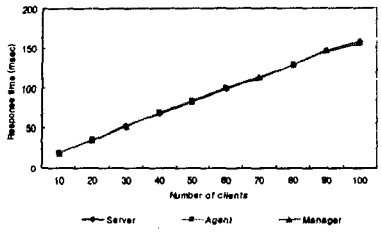


그림 4. 응답시간 (miliseconds)

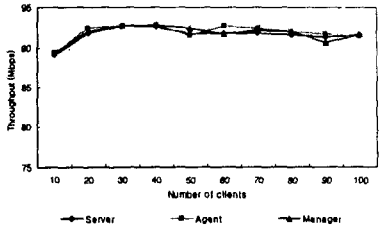


그림 5. 처리율 (Mbits/sec)

**4. 원격 웹 서버 로깅 시스템의 부하 측정**

이번 절에서는 http_netlog가 실제로 웹 서버 시스템에는 어느 정도의 영향을 미치는가에 대하여 알아보기 위하여 몇 가지 성능 평가를 하였으며, 실험 환경 및 결과를 차례로 논하기로 한다.

**4.1 실험 환경**

원격 웹 서버 로깅 시스템의 부하를 측정하기 위한 실험 환경은 리눅스 환경으로 구성하였다. 서버 시스템은 750MHz P-III/256MB/Linux 2.4.20을 사용하였으며, 클라이언트 시스템은 1.7GHz P-IV/512MB/Linux 2.4.20을 사용하였다. 웹 서버는 아파치 웹 서버 1.3.29을 사용하였으며, WebStone 2.5를 사용하여 웹 서버의 벤치마크를 하였다[7][8]. 결과는 각 집단마다 클라이언트의 수를 10에서 100까지 증가 시키면서 세 번씩 실험하여 그 평균을 구하였다. 아파치 웹 서버에서는 설정 파일에서 사용하는 기본값을 그대로 적용하였다. 다만, http_netlog가 동작하는 경우에만 아파치 웹 서버의 로그 기록을 사용하지 않았다.

**4.2 실험 결과**

총 세 가지 모델로 나누어서 측정을 하였으며, 각 모델당 클라이언트 수에 따른 초당 접속률(connection rate), 응답시간(response time), 처리율(throughput) 결과를 각각 그림 3, 4, 5에 나타내었다. Server는 로깅 시스템을 사용하지 않은 웹 서버를 나타내며, Agent는 로그 수집기 없이 로그 생성기만으로 웹 서버 디스크에 로깅하는 모델이며, Manager는 로그 생성기와 수집기를 모두 사용한 모델을 나타낸다. Agent와 Manager 모델 모두 로깅 시스템을 사용하지 않을 때의 웹 서버의 성능을 떨어뜨리지 않고 있으며, 경우에 따라서는 더 좋은 성능을 보여주기도 하였다. 이것은 원격 웹 서버 로깅 시스템의 부하가 크지 않다는 것을 의미하며, 그럼에도 불구하고 더욱 많은 정보를 제공하는 로깅 시스템의 이점을 잘 반영하는 결과라고 할 수 있다

**5. 결론**

본 논문에서는 웹 서버에게 부하를 주지 않으면서 네트워크를 통해 원격으로 로깅하는 시스템인 http_netlog를 구현하였다. 원격 웹 서버 로깅 시스템은 웹 서버의 종류와는 관계없이 로그 데이터를 수집할 수 있으며, 특히 클러스터링 시스템과 같은 여러 개의 서버 군으로 이루어진 시스템에서도 각 시스템의 웹 서버나 환경과는 관계 없이 독립적으로 로그 데이터를 수집할 수 있다는 장점을 갖고 있다.

또한 원격 웹 서버 로깅 시스템이 웹 서버에 얼마나 많은 영향을 미치는가에 대한 측정을 실시한 결과 웹 서버에 거의 부담을 주지 않는 것으로 나타나 웹 서버의 성능 저하 없이 더 자세한 로그 데이터를 수집할 수 있음을 보여주었다.

**참고 문헌**

[1] Netcraft Web Server Survey, "http://www.netcraft.com/survey"

[2] Scott M. Baker and Bongki Moon, "Distributed cooperative web servers," Computer Networks, 31(11-16):1215-1229, 1999

[3] Vivek S. Pai, Mohit Aron, Gaurav Banga, Michael Svendsen, Peter Druschel, Willy Zwaenepoel, and Erich Nahum, "Locality-aware Request Distribution in Cluster-based Network Servers," In Proceedings of the Eighth International Conference on Architectural Support for Programming Languages and Operating Systems, 1998

[4] Erich M. Nahum, Tsipora Barzilai, and Dilip Kandlur, "Performance issues in WWW servers," In Proceedings of ACM SIGMETRICS Conference on Measurement and Modeling of Computer Systems, 1999

[5] Yiming Hu, Ashwini Nanda, and Qing Yang, "Measurement, Analysis and Performance Improvement of the Apache Web Server," In Proceedings of 18th IEEE International Performance, Computing and Communications Conference, 1999

[6] W. Richard Stevens, "TCP/IP Illustrated, vol. 1," Addison Wesley, 491-497, 1994

[7] The Apache Group, Apache http Server Project, http://www.apache.org.

[8] Mindcraft, Inc., "WebStone—the Benchmark for Web Servers," http://www.mindcraft.com/webstone.