

사용자 평등성을 제공하는 웹 서버 허용 제어 장치 설계

⁰이철 임상석 황주영 박규호
한국과학기술원 전자 전산학과
{chullee, sslim, jhwang}@core.kaist.ac.kr kpark@ee.kaist.ac.kr

Design of a Ticket Based Admission Controller for Users' Fairness of Web Server

⁰Chul Lee, Sang-Seok Lim, Joo-Young Hwang Kyu-Ho Park
Dept. of EECS, Korea Advanced Institute of Science and Technology

요 약

본 논문에서는 웹 서버의 허용 제어 장치를 설계하였다. 웹 서버 허용 제어 장치는 웹 서버의 처리 용량을 초과하는 사용자 요청을 차단하여, 과부하로부터 웹 서버를 보호한다. 허용 제어 장치에 의하여 요청이 차단된 사용자는 재시도하는 특성을 보이는데, 이로 인하여 웹 서버의 부하는 가중된다. 기존의 웹 서버 허용 제어 장치는 이러한 사용자의 재시도 특성을 고려하지 않음으로써, 재시도하는 사용자가 다시 허용이 차단될 수 있다. 본 논문에서 제안하는 티켓 기반 웹 서버 허용 제어 장치는 재시도하는 사용자를 구분하고 우선적으로 서비스하여, 웹 서버의 과부하에 사용자간 평등함을 제공한다.

1. 서 론

초고속 인터넷 망의 대중적인 보급으로 인하여 인터넷 사용자는 급격히 증가하였다. 광섬유를 기반으로 전송 기술이 비약적으로 발달하였고, 이로 인하여 물리적인 계층의 대역폭은 충분하게 되었다. 하지만, 인터넷 서버의 성능은 그에 미치지 못하고 있으며, 과도해진 인터넷 트래픽의 폭주는 서버의 성능 저하나 고장을 야기시킨다. 트래픽 폭주로 인한 서버의 성능 저하 혹은 고장을 방지하기 위하여, 웹 서버 시스템에 허용 제어 장치를 적용하고 있다. 웹 서버의 허용 제어 장치는 서버의 폭주가 탐지 되었을 때 사용자 요청을 차단한다. 웹 서버의 부하가 최대 용량 이내로 유지되도록 충분한 양의 사용자 요청을 차단해야만 서버의 고장을 막고 최대 성능을 유지할 수 있다[1].

웹 서버 허용 제어 장치에 의하여 요청이 차단된 사용자는 재시도 하는 특성을 갖는다[2]. 재시도하는 요청은 웹 서버의 과부하를 가중 시킬 뿐 아니라, 또다시 차단될 확률이 매우 크다. 기존의 허용 제어 장치와 같이 재시도 하는 사용자를 구분하지 않으면, 여러 차례 차단되는 사용자로 인한 서비스 기아 현상이 발생한다.

본 논문에서 제안하는 티켓 기반 허용 제어 장치는 차단된 사용자에게 티켓을 부여한다. 재시도 하는 사용자는 서버로 전송하는 요청에 티켓을 포함하여 차별화가 가능하다. 재시도 요청을 차별화함으로써 서비스 기아 현상을 방지할 수 있다.

본 논문의 2장에서는 티켓 기반 허용 제어 기법에 대하여 설명하며, 3장에서는 제안된 방법을 리눅스 운영체제의 커널 모듈로 설계하는 과정을 설명한다. 4장에서는 재시도 회수를 측정하기 위한 실험 환경을 설명하고 실험 결과를 분석한다. 5장에서 결론을 내린다.

2. 티켓 기반 허용 제어 기법

제안된 티켓 기반 허용 제어 장치는 허용이 차단된 사용자에게 티켓을 부여한다. 사용자의 웹 브라우저는 자동적으로 요청을 재차 전송하는데, 이때는 허용 제어 장치로부터 부여 받은 티켓을 포함한다. 티켓은 HTTP 헤더의 쿠키 필드에 저장되며, 재시도 횟수와 관련된 정보를 포함한다.

티켓 기반 허용 제어 장치는 사용자 요청을 3계층의 등급으로 분류한다. 하급의 요청은 티켓을 포함하지 않은 사용자로부터 전송된 것이다. 중급의 요청은 티켓을 포함하여 전송되는 것으로 과거 한 번의 허용 차단은 당한 것이며 첫 번째 재시도임을 나타낸다. 상급의 요청은 과거 두 번 이상의 허용 차단을 당하여 두 번째 이상의 재시도하는 요청임을 나타낸다.

그림 1은 기존 웹 서버 허용 제어 장치의 문제점을 도시한 것이다. 사용자 B는 첫 번째 요청에 의하여 서비스를 받는 반면, 사용자 A는 여러 번 요청을 재시도 함에도 불구하고 서비스를 받지 못하여 결국 기아상태에 빠진다. 이와 같이 기존의 방법은 사용자간의 평등함을 제공하지 못한다.

제안된 티켓 방식을 사용하는 경우, 사용자 A는 첫 번째 요청에서 차단당하지만 티켓을 부여 받는다. 그림 2와 같이 사용자 A의 웹 브라우저는 티켓을 포함하여 요청을 재시도 한다. 사용자 A의 재시도 요청은 사용자 B의 요청보다 요청의 등급이 높기 때문에 우선적으로 서비스 받는다. 사용자의 웹 브라우저는 사용자의 간섭 없이 자동적으로 요청을 재시도하며, 허용 제어 장치로부터 티켓을 부여 받았을 때 티켓에 포함된 지연 시간만큼

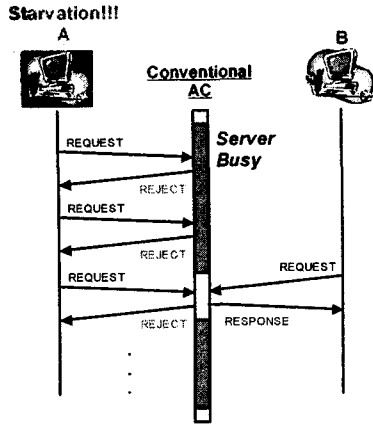


그림 1. 기존의 허용 제어 장치

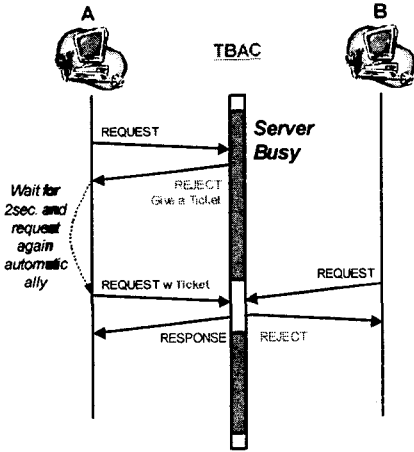


그림 2. 티켓 기반 허용 제어 장치

지연 후에 재시도하는 기능을 갖는다. 이와 같이 재시도를 지연하는 이유는 요청의 폭주로 인하여 상급의 요청의 급격한 증가를 방지하기 위한 것이다.

이와 같이 제안된 기법을 HTTP의 쿠키와 HTML 문법의 자바스크립트로 구현할 수 있다. 티켓 허용 제어 장치가 사용자의 요청을 차단하게 될 때, 사용자의 브라우저가 쿠키를 설정하도록 하여, 재시도 요청을 구분할 수 있으며, 자바 스크립트를 이용하여 사용자의 웹 브라우저의 재시도를 지연시킬 수 있다. 쿠키와 자바 스크립트는 최근의 웹 서비스 환경에서 널리 쓰이고 있다.

3. 티켓 기반 허용 제어 장치 설계

제안된 티켓 기반 허용 제어 장치를 리눅스 운영체제의 커널 모듈로 구현하였다. 웹 서버는 티켓 기반 허용 제어 장치(TBAC)과 아파치 웹 서버로 구성된다. 티켓 기반 허용 제어 장치가 사용자 요청을 전송받고, 허용 여부를 결정하는 과정을 설명하면 다음과 같다. 먼저, 사

용자 요청을 전송 받기 위해 접속을 설립한다. 이후, 사용자 요청 데이터를 전송받으면 요청의 등급을 분류하기 위하여 과실 과정을 거친다. 웹 서버의 과부하가 탐지되면 현재 사용자 요청에 대한 허용을 차단하고 티켓을 포함한 차단 메시지를 사용자에게 전달한다. 허용이 결정될 경우 아파치 웹 서버로 요청을 전달하여 올바른 서비스가 될 수 있도록 한다.

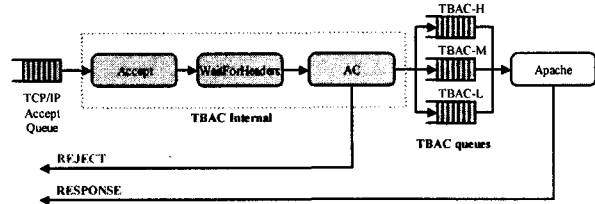


그림 3 티켓 기반 허용 제어 장치 구조

티켓 기반 허용 제어 장치는 단일 프로세스 이산 사건 기반 프로세스로 구현 되었다. 그림 3에서 보는 것과 같이 3개의 스테이지로 구성되어 루프를 돌면서 이산 사건의 발생에 따라 해당 코드를 수행한다. 티켓의 정보를 바탕으로 구분되어진 사용자 요청의 등급에 따라 우선순위 기반 스케줄링을 수행하는데, 이를 위하여 허용 제어 장치 모듈과 아파치 웹 서버 사이에 다중 큐를 사용하였다. 그림 3에서와 같이 3개의 다중 큐가 존재하며 각각 상,중,하급으로 분류된 요청이 대기하는 큐이다. 아파치 웹 서버는 상급의 큐로부터 먼저 요청을 처리 하게 됨으로써, 우선순위 기반 스케줄링이 가능하다. 전체 큐의 길이가 정해진 한계 값을 초과할 때를 허용을 차단하는 정책을 사용하였다.

4. 실험

4.1 실험 환경

허용 제어 기능을 하는 웹 서버에서는 성능 지표가 단순한 응답 지연 시간이 아닌, 올바른 응답을 받을 때까지 요청을 재시도한 회수가 된다. 기존의 벤치마크 도구들은 제안된 티켓 방식을 사용한 허용 제어의 동작을 확인하거나, 성능을 측정하기 위한 통계 자료들을 추출 할 수가 없어서, 새로운 성능 측정 도구가 제안되었다. 과부하 상태에 도달한 웹 서버의 특성을 확인하기 위해서는, 서버의 과부하 상태를 유지할 수 있도록 httpf[3]와 같이 일정한 비율로 계속해서 요청을 할 수 있어야 한다. 또한, 서버로부터 분주 페이지를 응답 받았을 때 사용자가 인내하는 범위 내에서 올바른 응답을 받을 때까지 요청을 재시도 해야 한다. 새로 요청을 시도하는 사용자는 최대 5번까지 요청을 재시도하고, 그 이후에도 서비스를 받지 못할 경우 포기한다고 가정하였다. 워크로드는 SpecWeb99[4]의 동적 요청파일인 specweb99-cgi.pl을 사용하였다. 그림 4와 같은 패턴으로 100초동안 최대 서버 처리 용량의 150%에 해당하는 요청을 발생시켰다.

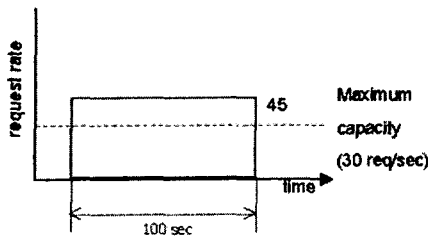


그림 4 요청 패턴

웹 서버로는 PentiumII 450 SMP CPU, 128MB 메모리, 100Mbps NIC와 리눅스 운영체제를 사용하였다. 워크로드 생성을 위하여 3대의 클라이언트 머신을 사용하였으며 웹 서버와 같은 스펙을 갖는다.

4.2 실험 결과

그림 5는 올바른 응답을 받기까지 재시도 한 회수에 따른 요청의 수에 대한 분포이다. 다섯번의 재시도에도 올바른 응답을 받지 못하는 사용자는 서비스 요청을 포기하는 것으로 가정하고, 6+로 표시하였다. 그래프에서 볼 수 있듯이, 티켓 방식을 사용하는 경우, 재시도되는 요청이 우선권을 가지므로 첫번째 시도에 의하여 올바른 응답을 받은 요청의 개수는 비교적 적은 반면, 제한된 재시도 회수 이내에 대부분의 요청이 처리 되는 것을 볼 수 있다. 표2를 보면, 티켓 방식의 경우 모든 요청이 다섯번의 재시도 회수 이내에 서비스를 받을 수 있지만, 티켓 방식을 사용하지 않을 때는 여러 차례 요청을 재시도 해도 서비스를 받지 못하고 결국 요청을 포기하는 사용자가 3.54%정도 존재 한다.

표 1의 결과에서 티켓 방식의 사용할 때 재시도 회수

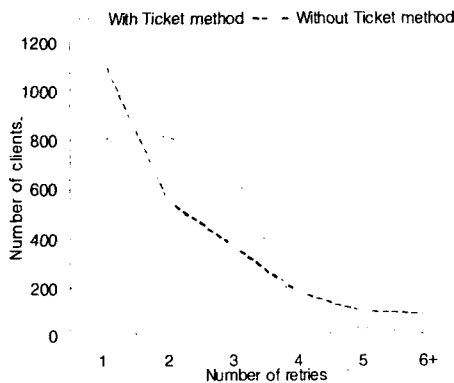


그림 5 재시도 회수에 따른 사용자수 분포

재시도 회수	티켓 방식	기존의 방식
평균	2.05	2.10
표준편차	0.92	1.36
손실 사용자 수	0.04%	3.54%

표 1 종합적인 결과

에대한 표준 편차가 0.4 정도 줄어든 것을 확인 할 수 있는데, 이것은 티켓방식을 사용할 때 사용자간에 평등한 재시도 회수를 갖는다고 볼 수 있다. 따라서 티켓 기반의 허용 제어 장치는 사용자 평등성을 제공한다.

4. 결 론

본 논문에서는 웹 서버 시스템의 성능 감소를 막기 위한 허용 제어 장치에서 허용이 거부되어 요청을 재시도 하는 사용자에게 우선권을 부여하는 것을 제안하였다. 제안된 기법을 이용하여 웹 서버 허용 제어 장치를 구성할 경우 서버의 과부하로 인하여 서비스를 거부 받은 사용자는 재시도 하는 요청에 우선권을 갖게 되어 제한된 재시도 회수 이내에 서비스 받는다. 이것은 일시적인 요청의 집중으로 여러 차례 서비스를 거부 받은 사용자가 서버가 다운된 것으로 착각하고 웹 사이트를 떠날 경우를 방지한다. 실험 결과에서는 제안된 기법을 사용하지 않은 허용 제어 장치와 비교하여 서비스를 포기하는 사용자가 3.5% 정도 줄어들었다. 이와 같이 제안된 웹 서버 허용 제어 장치는 사용자간의 평등성을 제공한다.

4. 참고 문헌

[1] A. Iyengar, E. MacNari and T. Nguyen, "An Analysis of Web Server Performance", In Proceedings of the IEEE Global Telecommunications Conference, 1997
 [2] Zona Research Inc., <http://www.zonaresearch.c>
 [3] D. Mosberger and T. Jin, "A tool fore measuring web server performance", In Proceedings of the Internet Server Performance Workshop , 19
 [4] SPECWEB, <http://www.spec.org/>