

eBPF를 활용한 Linux 커널 I/O 스택의 계층별 지연 시간 분석

김유리¹, 이찬용², 강동현¹

¹동국대학교 컴퓨터·AI학부

²동국대학교 일반대학원 컴퓨터·AI학과

최근 인공지능 및 빅데이터 분석과 같이 입출력 집약적인 응용 프로그램이 확산됨에 따라, 빠르고 효율적인 I/O 처리의 중요성이 부각되고 있다[1]. 이를 위해 NVMe SSD와 같은 고성능 저장 장치가 도입되었으나, 저장 장치의 성능 향상에 비해 커널 수준에서 발생하는 I/O 처리 오버헤드의 비중이 전체 지연 시간에서 점차 증가하고 있다[1]. 이러한 오버헤드는 시스템 호출, 인터럽트 처리, 데이터 복사 등 커널이 사용자 영역과 저장 장치 간의 I/O 요청을 중재하는 과정에서 발생하며, 고속 저장 장치의 잠재적 성능을 저해하는 주요 병목 요인으로 작용한다[1].

본 연구에서는 Linux 커널의 I/O 스택을 파일 시스템, 블록 계층, 장치 드라이버 등 주요 계층별로 세분화하여 I/O 경로를 추적하고, 각 계층의 지연 시간을 정량적으로 분석하였다. 이를 통해 고성능 저장 장치 환경에서 커널 내부의 주요 병목 지점을 식별하고, 효율적인 I/O 처리 구조 설계를 위한 기반을 마련하고자 한다. 우리는 Linux 커널에서 제공하는 동적 분석 도구인 eBPF를 활용하여 사용자 영역의 시스템 호출부터 하드웨어 드라이버까지 전 구간을 통합 추적하는 방법론을 구현하였다. eBPF 맵을 통해 각 I/O 요청의 타임스탬프와 지연 시간을 실시간으로 수집하고, 프로세스 ID와 커널 구조체의 주소를 키로 사용하여 요청을 추적하였다. 이를 통해 기존 I/O 추적 도구로는 파악하기 어려웠던 파일시스템 계층의 오버헤드를 정량적으로 측정하였다.

1MB 파일 쓰기 작업을 반복 실험한 결과, NVMe 하드웨어 처리 시간(약 30-40μs) 대비 파일 시스템 계층에서 5-25배(200μs-1ms)의 오버헤드가 발생하였다. 이를 통해 커널 I/O 스택이 고성능 저장 장치 환경에서 주요 병목 지점으로 작용함을 확인하였다. 또한, I/O 요청이 프로세스 컨텍스트에서 시작되어 인터럽트 컨텍스트에서 완료되는 비동기적 특성을 확인하였다.

향후 연구에서는 bio 및 request 구조체를 활용하여 요청 식별 체계를 구축하여, I/O 요청의 모든 과정을 분석할 수 있는 통합 프로파일링 도구를 개발할 계획이다. 이를 통해 시스템 관리자와 개발자가 커널 I/O 경로상의 병목을 정확히 진단하고 최적화할 수 있는 기반을 제공하고자 한다.

Acknowledgments 이 성과는 정부(과학기술정보통신부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임 (RS-2023-00251730).

References [1] G. Lee, S. Shin, W. Song, T. J. Ham, J. W. Lee, and J. Jeong, “Asynchronous I/O Stack: A Low-latency Kernel I/O Stack for Ultra-Low Latency SSDs”, *Proc. Of the USENIX Annual Technical Conference (USENIX ATC 19)*, 2019, pp. 603–616.