

BPF Memory Allocation

Harry (Hyeonggon) Yoo

오늘의 주제

BPF 프로그램의 요청으로 인해 커널에서* 메모리를 할당해야 하는 경우 발생하는 문제들을 다룬다.

*커널을 거치지 않고 BPF 프로그램 내에서 메모리를 할당하는 경우는 다루지 않는다.

BPF란 무엇인가

BPF는 커널 내의 가상머신으로, 커널을 다시 컴파일할 필요 없이 **userspace**에서 제공한 **BPF** 프로그램 코드를 검증 및 컴파일 후 커널 모드에서 실행하는 인프라를 제공한다.

BPF 프로그램 실행 컨텍스트의 특수성

BPF 프로그램은 일반적인 태스크와 독립적인 스케줄링 단위가 아니다.

대신, 커널 코드의 특정 지점에 **attach**되어 실행된다.

- Kprobe / kretprobe
- Tracepoint
- fentry / fexit
- Perf events

BPF에서 커널 코드를 호출하는 경우

BPF 프로그램은 BPF 프로그램 코드 내의 함수를 호출하기도 하지만,
BPF helper, kfunc 등 일부 API는 커널 코드를 호출한다.

문제점: 임의의 지점에서 호출될 수 있는 커널 코드는 할 수 있는 일이 매우 제한된다*. 임의의 락을 획득하면 데드락이 발생한다*.

*예를 들어, 버디 할당자에 attach된 BPF 프로그램에서 버디 할당자를 호출하면 데드락이 생길 수 있다.

*실제로 BPF은 초기에 hash map에서 kcalloc()을 사용하다가 데드락이 발생했다: [commit 6c9059817432](#)

현실적인 해결방안들

1. BPF 프로그램 실행 전에 필요한 메모리를 미리 할당 (preallocation)
2. BPF 내에 lockless 메모리 할당을 지원하는 특수한 할당자를 구현
3. 버디, 슬랩 할당자가 스핀락을 busy waiting 하지 않고 trylock으로만 메모리 할당을 시도한다. (alloc_pages_nolock(), kmalloc_nolock())

Q&A

감사합니
다.