

## **BPF tracing: exploring additional debugging capabilities**

*Monday, September 20, 2021 7:00 AM (40 minutes)*

Sometimes using tracing - instead of traditional kernel debuggers - to investigate kernel issues can be necessary. Some problems such as races are inherently time-sensitive, so minimally invasive tracing is ideal in such cases. However, it is also true that debuggers have capabilities that BPF-based tracing does not (or has recently acquired) - printing data structures, tracking local variable values, inline function visibility, etc. In fact looking at gdb capabilities is a great way to explore some possibilities for future tracing functionality. Here we explore some of the possibilities, the potential cost-benefit, and seek feedback and discussion with the community on the potential value of the approaches explored.

### **I agree to abide by the anti-harassment policy**

I agree

**Primary author:** MAGUIRE, Alan (Oracle)

**Presenter:** MAGUIRE, Alan (Oracle)

**Session Classification:** BPF & Networking Summit

**Track Classification:** Networking & BPF Summit (Closed)