Contribution ID: **18** Type: **not specified**

Ahead-of-time compiled bpftrace programs

Tuesday 21 September 2021 09:30 (40 minutes)

bpftrace was originally designed with a dynamic compilation model. While that model has worked fairly well, new developments and concerns in the eBPF ecosystem have prompted re-evaluation of the original design.

First, BPF is becoming more widely used so performance is more of a concern. Running LLVM to generate bytecode on the fly is somewhat costly, especially for bpftrace-enabled data collection in production. Binary weight is also an issue because bpftrace currently ships with LLVM and clang libraries.

Second, signed BPF programs is making its way into the kernel in response to security concerns. bpftrace is not immune to those security concerns so bpftrace must have an answer as well if it is to remain relevant in more secure environments.

Finally, CO-RE, the building blocks for portable BPF programs have become production ready and are shipping in many distros. bpftrace can build on top of these pieces to deliver ahead-of-time compiled bpftrace programs. These AOT programs will faster to run and smaller in binary size.

This talk will go into the ongoing work to enable AOT bpftrace programs. There are a lot of moving pieces because the existing code has made broad assumptions about the compilation model. Hopefully by the end of this talk, participants will have a better idea about what work has been accomplished, what remains to be done, and what unsolved issues still need to be resolved.

I agree to abide by the anti-harassment policy

I agree

Primary author: XU, Daniel (Facebook)

Presenter: XU, Daniel (Facebook)

Session Classification: BPF & Networking Summit

Track Classification: Networking & BPF Summit (Closed)