



Contribution ID : 139

Type : **not specified**

Compile-Once Run-Everywhere BPF Programs?

Thursday, 15 November 2018 09:30 (20)

Compile-once and run-everywhere can make deployment simpler and may consume less resources on the target host, e.g., without llvm compiler and kernel devel package. Currently bpf programs for networking can compile once and run over different kernel versions. But bpf programs for tracing cannot since it accesses kernel internal headers and these headers are subject to change between kernel versions.

But compile-once run-everywhere for tracing is not easy. BPF programs could access anything in the kernel headers, including data structures, macros and inline functions. To achieve this goal, we need (1) preserving header-level accesses for the bpf program, and (2) abstracting header info of vmlinux. Right before program load on the target host, some kind of resolution is done for bpf program against the running kernel so the resulted program is just like to that compiled against host kernel headers.

In this talk, we will explore how BTF could be used by both bpf program and vmlinux to advance the possibility of bpf program compile-once and run-everywhere.

I agree to abide by the anti-harassment policy

Presenter(s) : SONG, Yonghong (Facebook); STAROVOITOV, Alexei (Facebook)

Session Classification : BPF MC