Linux Plumbers Conference 2025

Contribution ID: **135** Type: **not specified**

ePass: A Framework for Enhancing Flexibility and Runtime Safety of eBPF Programs

eBPF enables safely extending kernel functionality for various applications, but its static verifier is overly restrictive, preventing many useful and valid programs in practice from running. It can also miss safety violations in complex conditions. Recent work proposes adding runtime checks to mitigate these limitations, but they narrowly target specific cases. Their instrumentations require significant effort and are error-prone.

We present ePass, a framework that provides systematic and *verifier-cooperative* runtime checking for enhancing eBPF flexibility and safety. ePass introduces a novel Intermediate Representation (IR) that lifts eBPF bytecode into an SSA (Static Single Assignment) form, enabling systematic instrumentation of runtime checks. It provides intuitive APIs for developers to easily implement diverse transformation passes. ePass ensures these passes preserve existing safety rules while enhancing runtime safety.

To showcase ePass' versatility, we develop 12 passes that address different verifier limitations and safety gaps, such as instruction limit enforcement, memory sanitization, and helper function argument validation. They most take under 100 lines of code. Our evaluation further shows that ePass enables real-world programs that are previously rejected to execute safely, mitigates known vulnerabilities, and incurs low overhead.

ePass's toolchain is completely open-source at https://github.com/OrderLab/ePass.

Primary authors: XIANG, Yiming (University of Michigan); Ms HE, Wanning (University of Michigan); AL-QUVI, Mehbubul Hasan; Prof. HUANG, Ryan (University of Michigan); Prof. WITCHEL, Emmett (University of Texas at Austin)

Presenter: XIANG, Yiming (University of Michigan)

Session Classification: eBPF Track

Track Classification: eBPF Track