Contribution ID: **371** Type: **not specified**

Making Sense of State Pruning

State pruning allows the BPF verifier to mitigate the path explosion problem and scale to large programs. With its underlying algorithms, precision tracking, strongly connected components computation, and liveness analysis, state pruning accounts for around 15% of the verifier. Its many heuristics have been tuned over a decade of trial and error.

While state pruning inefficiencies can lead to programs being rejected, bugs can lead to bytecode being incorrectly identified as dead code and eliminated. Nevertheless, despite its key role and complexity, this critical pruning logic remains not widely understood and would benefit from broader reviews.

This talk therefore aims to demystify state pruning, covering:
- how state pruning works;
- the underlying algorithms, the heuristics and their evolution;
- limitations & shortcomings.

To conclude this talk, we will describe existing and ongoing works to test the pruning logic, and discuss our own propositions to improve its debuggability.

Primary authors: TARDY, Mahé; CHAIGNON, Paul (Isovalent)

Presenters: TARDY, Mahé; CHAIGNON, Paul (Isovalent)

Session Classification: eBPF Track

Track Classification: eBPF Track