Kernel Dependability and Assurance Microconference

Lukas Bulwahn, Shuah Khan, Kate Stewart
Dependability and Assurance

**Dependability:** *(our definition)*
- ensure a specific system property, i.e., availability, safety, security, reliability, integrity
- includes system analysis, engineering and argumentation

**Assurance:** *(our definition)*
- confidence in a system property or a quality property of the engineering process
- includes creation, exchange, review of trustable arguments and evidences to convey confidence
Kernel Dependability and Assurance

In software, claims for distinct system properties (safety, security, availability) may rely on overlapping qualities of the software.

This MC is about Dependability and Assurance of basic operating system functionality, provided by the kernel and its basic user-space libraries (libc, ld.so, ...)

We want to discuss:

- how can **we as Kernel Developers** gain confidence that certain claims for the Linux kernel hold?
- how can **we as Users** gain confidence that certain claims for the Linux kernel hold?
- And how can **we as Kernel Developers make Users** gain confidence that certain claims for the Linux kernel hold?
Verification
Fault Detection & Control
Argumentation Completeness
Development Process
Integrity

Linux Kernel dependability - Proactive & reactive thinking
Maintaining results from static analysis collaboratively?
Understanding Linux Lists
Assessing kernel system call correctness by testing
TCB safety
Safety in processes CPU execution state

Avoiding Security Flaws
Following the Linux Kernel Defence Map

Assessing kernel system call correctness by testing

LINUX PLUMBERS CONFERENCE
August 24-28, 2020