Contribution ID: **236** Type: **not specified**

Measuring Test Coverage of Kernel Object Code

Test coverage is a measurement of how much code is executed by a given test or test suite. Current implementations in the kernel are measured against source code with tools such as “gcov” or “llvm-cov”. However, source-based coverage measurements are unable to account for additional code not present in the original source, such as code inserted by the build system (compiler, linker, build scripts, etc.). To supplement source-based coverage, The Boeing Company and the University of Illinois Urbana-Champaign (UIUC) have investigated, created a tool, and measured coverage of the object code of the Linux kernel.

Object code coverage is a requirement in several safety critical industries, such as Automotive and Aerospace software. In addition to certification, object code coverage is effective at identifying machine code which ends up in an executable but cannot be traced back to any source code. As an example, build toolchain exploits which maliciously insert object code (such as the XZ Utils backdoor) can be identified by object code coverage.

This presentation will cover:
- An overview of object code coverage and why it is useful
- The need for object code coverage in safety critical applications (e.g. Automotive, Aerospace, Medical)
- The development of an open-source object code coverage tool which works on the Linux kernel
- Approaches used to measure object code coverage on emulated targets (QEMU) and real hardware
- Measurement differences between x86-64 and ARM64 kernels
- Challenges collecting coverage of the Linux kernel and getting accurate results
- Object code coverage results of the kernel when run with existing test suites

Ken Thompson’s lecture “Reflections on Trusting Trust” summarized the issue stating, “No amount of source-level verification or scrutiny will protect you from using untrusted code.” Many issues can only be identified by inspecting object code, and object code coverage is one metric to assist object code inspection.

Primary author: OPPELT, Andrew (The Boeing Company)

Presenter: OPPELT, Andrew (The Boeing Company)

Session Classification: LPC Refereed Track

Track Classification: LPC Refereed Track