Contribution ID: **272** Type: **not specified**

Source-based code coverage of Linux kernel

We have been working on measuring Linux kernel's source-based code coverage using LLVM toolchain. Prior coverage testing tools like gcov instrument the target kernel at the IR level and do not have a precise mapping back to the source code. If compiler optimizations are enabled (which is the default for building Linux kernel), coverage will not map well to the source code, and the reports are oftentimes confusing. To solve this problem, source-based code coverage [1] does instrumentation at the compiler frontend and maintains accurate mapping from coverage measure to source code locations. Source-based results can reflect exactly how the code is written and can present advanced metrics like branch coverage and Modified Condition/Decision Coverage (MC/DC) [2] in a human friendly way. Comparison of these two tools using real kernel code examples can be found in [3].

This talk is complementary to our proposed talk submitted to the Refereed Track. This talk is a deeper dive of our work on supporting advanced testing towards certifying Linux.

- We will describe our kernel patch that enables source-based code coverage tools in a freestanding environment like Linux kernel.
- We will compare kernel coverage reports generated by both gcov and source-based code coverage on real kernel code examples to demonstrate how source-based results are more informative.
- We will share some of the earliest MC/DC measurements of Linux kernel enabled by our patch.

Reference

1. Source-based Code Coverage https://clang.llvm.org/docs/SourceBasedCodeCoverage.html
2. Making Linux Fly: Towards Certified Linux Kernel, https://elisa.tech/event/elisa-seminar-making-linux-fly-towards-certified-linux-kernel/
3. Real kernel code examples where gcov can be confusing https://github.com/xlab-uiuc/linux-mcdc/issues/7

Primary authors: ZHANG, Wentao (University of Illinois Urbana-Champaign); MARINOV, Darko (University of Illinois at Urbana-Champaign); JIA, Jinghao (University of Illinois Urbana-Champaign); XU, Tianyin (University of Illinois at Urbana-Champaign)

Session Classification: Safe Systems with Linux MC

Track Classification: Safe Systems with Linux MC