Contribution ID: **320** Type: **not specified**

Measuring and Understanding Linux Kernel Tests

Friday, 20 September 2024 10:30 (30 minutes)

We have been working on an LLVM-based toolchain for measuring test adequacy of existing kernel tests from test suites including KUnit [1], kselftest [2], LTP [3], test suites from RHEL [4] and more in KCIDB [5]. We measure different adequacy metrics including basic metrics statement coverage and branch coverage, and advanced metric Modified Condition/Decision Coverage (MC/DC) [6].

This talk is complementary to our proposed talk submitted to the Refereed Track. In this talk, we would like to present our results and share our analysis on the measured adequacy with different metrics, with a focus on the following aspects:

- The coverage measures of different test suites and their combinations (including statement coverage, branch coverage, and MC/DC);
- An analysis of uncovered code (by different metrics) and the implications to existing testing practices;
- An analysis of different test suites and their "distance" from recent kernel bugs.

Reference
1. KUnit - Linux Kernel Unit Testing, https://docs.kernel.org/dev-tools/kunit/index.html
2. Linux Kernel Selftests, https://docs.kernel.org/dev-tools/kselftest.html
3. Linux Test Project, https://github.com/linux-test-project/ltp
4. Red Hat Kernel QE and CKI kernel tests repository, https://gitlab.com/redhat/centos-stream/tests/kernel/kernel-tests
5. Catalog of tests used by KCIDB, https://github.com/kernelci/kcidb/blob/main/tests.yaml
6. Making Linux Fly: Towards Certified Linux Kernel, https://elisa.tech/event/elisa-seminar-making-linux-fly-towards-certified-linux-kernel/

Primary authors: REN, Tingxu (University of Illinois at Urbana-Champaign); 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: Kernel Testing & Dependability MC

Track Classification: Kernel Testing & Dependability MC