Linux Plumbers Conference 2022

Contribution ID: **78**

Type: **not specified**

The Integration of Rust with Kernel Testing Service

Monday, 12 September 2022 17:30 (30 minutes)

Rust for Linux aims to bring Rust into the kernel as the second programming language. With the great advancing of this target, a corresponding testing service for Rust is becoming a potential requirement.

0-Day CI team has been working closely with the maintainers of Rust for Linux to integrate Rust into kernel test robot. We'd like to share our experience of enabling Rust test. Here are some of the progress we have made:

- Kernel test robot is a bisection-driven CI, we not only scan build errors, but also run bisections to look for the first bad commits which introduced the errors. To maintain the capability of bisection, we setup automatic upgrade and adaptive selection for Rust toolchain, thus to match the required toolchain version of different commits during the process of bisection.

- We provide both random config and a specific config with all Rust samples enabled to have different level of code coverage for Rust in kernel.

Most of the work we have done is about building kernel with Rust enabled, and we are considering runtime test in the next step. We are also interested in various topics which may help to enhance Rust test. Some further work we are looking forward to happen are:

- Boot/fuzzing testing for Rust code such as leveraging syzkaller.

- Functional testing for core Rust code and modules, which could be part of common framework like kunit/kselftests to be easily used in CI service.

- Collect and aggregate Rust code coverage data in kernel to better design and execute tests.

- Wrap a tool to setup Rust environment based on min-tool-version.sh for consistent compiling and issue reproducing.

- Testing for the potential impact of different compiling options of Rust, such as optimization level and build assert config.

We hope that our work can give inspiration to other CIs wishing to integrate it, and help to facilitate the development of Rust for Linux.

I agree to abide by the anti-harassment policy

Yes

Primary authors: Mr LIU, Yujie (Intel); Mr LI, Philip (Intel)

Presenter: Mr LIU, Yujie (Intel)

Session Classification: Rust MC

Track Classification: LPC Microconference: Rust MC