



Contribution ID: 392

Type: **not specified**

First-party kernel.org build environments

Wednesday, 18 September 2024 17:40 (40 minutes)

In its simplest cases, building a kernel requires very few dependencies and can be done with a couple of make commands. However, things can get complicated very quickly: fine-tuned toolchains such as the kernel.org ones provide a wide variety to choose from, each compiler has a particular supported version range, eBPF kselftests require a cutting-edge LLVM, the Rust code is still tied to the latest rustc compiler releases...

How can contributors find out maintainers' recommended ways to build a kernel? How can test systems determine the best way to reach optimal build coverage? How should a kernel be built to reliably reproduce a known issue? These things can be very error-prone without a structured description of the build environment.

Following an email discussion^[1] on this topic, several ideas have already been brought up and a live session at Plumbers would help pave the way to reach a true upstream solution. In particular:

- kernel.org toolchains could be made available as packages (deb, rpm, ipk...) in addition to the existing plain tarballs
- packages would express dependencies with versions for other tools and help with security updates (e.g. the recent xz issue)
- on top of this, some reference Dockerfiles and other image build recipes (e.g. Yocto) could be maintained to facilitate generating full build environments
- Kbuild could then have an option to directly invoke a container manager

Many steps have already been made in this direction by various independent parties. The proposal is now to try and consolidate this as a first-party solution for the upstream kernel, as per the quote below.

On 09/07/2024 07:30, Nathan Chancellor wrote:

I think it would be a good idea to try and solicit feedback from the greater kernel community at large to ensure that whatever solution is decided on will work for both testing systems and developers/maintainers. I think that a first party solution for having a consistent and easy to set up/work with build environment has been needed for some time but unfortunately, I am not sure how much discussion around this problem has happened directly with those folks.

[1] <https://lore.kernel.org/all/f80acb84-1d98-44d3-84b7-d976de77d8ce@gtucker.io/>

Primary author: TUCKER, Guillaume

Co-author: CHANCELLOR, Nathan

Presenter: TUCKER, Guillaume

Session Classification: Toolchains Track

Track Classification: Toolchains Track