THE LINUX FOUNDATION
OPEN SOURCE SUMMIT
EUROPE

Embedded Linux Conference

rsinit – a Tiny Initramfs Toolbox for Embedded Systems

Michael Olbrich – m.olbrich@pengutronix.de



https://www.pengutronix.de

About Me

- Michael Olbrich
- Embedded Linux developer
- @ Pengutronix since 2006
- Graphics team lead







Motivation

- Historically no initramfs in embedded systems
- New requirements:
  - rootfs integrity (dm-verity)
  - rootfs encryption (dm-crypt)
  - Development in a verified boot world





Requirements

- Minimal overhead (size & time)
- Robust
- Extensible





Existing Solutions

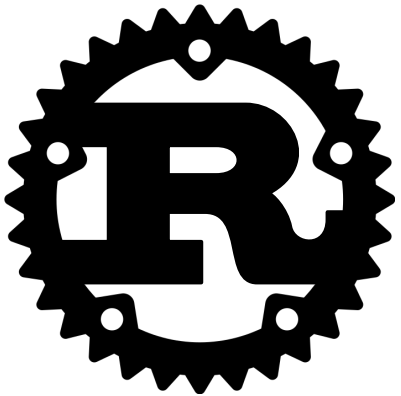
Yocto initramfs-framework

- Ready to use in Yocto ✅
- Extensible ✅
- Large Size ❌
- Shell scripts ❌





rsinit



https://github.com/pengutronix/rsinit

https://crates.io/crates/rsinit





rsinit – Design Choices

- Small binary size
- Rust only
- Default binary for basic use-cases
- Toolbox to build custom binaries for other use-cases





rsinit – Mounting the Rootfs

- Regular partitions
- dm-verity
- NFS
- 9p over virtio
- 9p over USB
- More to come...





rsinit – Writable rootfs

Problem:

- With dm-verity the rootfs is readonly

Solution:

- Mount volatile or persistent overlays 🛠️
- Bonus: trivial to go back to the original





Binary Size

- Static binary
- Musl as libc
- Optimize for size
  - ~450k
- Abort on panic etc. [1]
  - ~130k

[1] https://github.com/johnthagen/min-sized-rust





rsinit – Kernel Development

- Regular rootfs from mass storage
- Custom kernel / initramfs via tftp/nfs
- Kernel modules are bind-mounted from the initramfs
- `rsinit.bind_modules`

https://fosdem.org/2026/schedule/event/7XHGHX-netboot_without_throwing_a_fit/





Testing & CI

- Some unit tests
- Basic integration tests (with qemu)
- Integrated in Github Workflows

- TODO: coverage





rsinit – readahead (WIP)

- based on Steven Rostedt's ideas
- `mm_filemap_add_to_page_cache` trace event for recording
- posix_fadvise() to trigger readahead
- ~10% faster userspace for a simple test-case
- Draft PR #28





rsinit – Future

- Encrypted rootfs
- Boot splash
- ...





Thank You!

Questions?

https://github.com/pengutronix/rsinit

https://crates.io/crates/rsinit

