Contribution ID: **106**　　　　　　　　　　　　　　　　　　　Type: **not specified**

Revisiting How Kernels Invoke initrds

Thursday 19 September 2024 10:00 (25 minutes)

The current way most Linux systems use initrds is via the initramfs mechanism: a compressed cpio archive is generated via an initrd generator and passed to the kernel at boot. The kernel then decompresses this cpio archive at boot into a fresh tmpfs file system. This file system is then booted.

This is not ideal for various reasons: the tmpfs can never be unmounted, hence must be emptied when transitioning into the host. Morever, the a lot of work needs to be done ahead of time, touch the whole data in the initrd, even if not all files will be needed on a specific system. The fact that the tmpfs is writable is not ideal either.

Let's hence investigate how we can improve the situation, to make boots quicker, safer and the initrd image process reproducible and attestable.

This session is more about discussing various options and challenges, there's no ready-made proposal behind this session.

Ideas: mounting erofs from memory instead of cpio, making initial superblock unmountable, using pmem= on the kernel command line, and more.

Primary author:　POETTERING, Lennart

Presenter:　POETTERING, Lennart

Session Classification:　Kernel <-> Userspace/Init/System Management boundaries and APIs MC

Track Classification:　Kernel <-> Userspace/Init/System Management boundaries and APIs MC