

# Notes from Linux Plumbers Conference

Chris Simmonds, 2net

# Linux Plumbers Conference 2022

- Linux Plumbers Conference is a place where kernel devs and people using kernel interfaces get together
- This year it was in Dublin, Ireland from 12 to 14 September
- Plumbers has had an Android Micro Conference for the last 10 years - that is what I am reporting on here

LPC web pages: <https://lpc.events/>

LPC raw video feeds: [https://www.youtube.com/playlist?list=PLVsQ\\_xZBEyN0daQRmK04ysrjkSzaLI6go](https://www.youtube.com/playlist?list=PLVsQ_xZBEyN0daQRmK04ysrjkSzaLI6go)

(edited videos of all the talks will be available soon)

# The Android Micro Conference (MC)

- The Android MC was hosted by Karim Yaghmour (as always)
- 3.5 hours of talks and Q&A on Android kernel topics
- The following slides are a summary of the things I felt were most relevant to AOSP devs

Android MC raw video feed: [https://www.youtube.com/watch?v=PW7XA237dgI&list=PLVsQ\\_xZBEyN0daQRmK04ysrjkSzaLI6go&index=15](https://www.youtube.com/watch?v=PW7XA237dgI&list=PLVsQ_xZBEyN0daQRmK04ysrjkSzaLI6go&index=15)

# GKI experience (Elliot Berman, Qualcomm Innovation Centre)

- Now shipping full GKI 2.0 Android12-Linux5.10 (since 2021)
- Mixed build: vmlinux from ACK, kernel modules from Qualcomm tree
- Upstream patches much faster now
- Current kernel is Android 13-Linux 5.15
- Issue with devices with c. 1 GB RAM
  - using ARM32 kernel - smaller than ARM64
  - but, GKI not supported for ARM32
- Potential issue: does GKI encourage out of tree modules?

# Technical debt (Matthias Männich, Google)

- Technical Debt - patches on top of mainline Linux in the Android Common Kernel
  - aim to reduce patches on top of mainline
  - patches due to GKI are mostly merged upstream now
- 224 patches at the time of the talk
- Tracking out of tree patches
  - quilt series  
<https://android.googlesource.com/kernel/common-patches>
  - you can apply these to mainline kernel to get ACK

# Hermetic builds with Bazel (Matthias Männich, Google)

- Goal - always produce the same kernel binary each time it is built
- The old build.sh GKI build tool did not achieve this goal
- Solution: switch to bazel for Android 13 kernel builds
  - sandboxed, hermetic
  - better parallelization, faster (perhaps, still v slow for me)

```
$ mkdir android-kernel && cd android-kernel
$ repo init -u https://android.googlesource.com/kernel/manifest -b common-android13-5.15
$ tools/bazel build //common:kernel_aarch64_dist
```

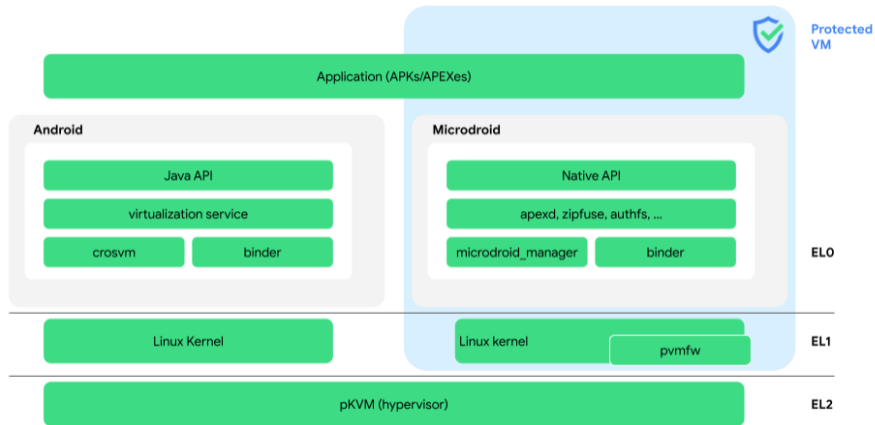
//common points to master bazel build file:

common/BUILD.bazel

# Virtualization in Android (David Brazdil, Serban Constantinescu, Google)

- New in Android 13 and android13-linux5.10 and android13-linux5.15
- Uses a type 1.5 hypervisor, **pKVM**, ARM64 only
- Runs **Microdroid** guest OS
  - platform code only; no Java
  - communicates with host OS via binder over sockets (vsock)
- Intended use: "Protected Computing Environment"
  - Machine learning models
  - Isolated Compilation - ahead-of-time compile of DEX code after OTA
- "The Truszone TEE is too privileged and too fragmented to use"

# Virtualization in Android



<https://source.android.com/docs/core/virtualization>



# Cuttlefish and Kernel Dev (Ram Muthiah, Google)

- This talk was a review of where we are with Cuttlefish - nothing new here
- Cuttlefish is an emulator used for kernel, system, BSP, and continuous integration testing
- Virtio compliant, works with QEMU, Crosvm, COQOS
  - except virtio gpu, which is not implemented in Cuttlefish for aarch64 (yet)

# eBPF-based FUSE (Paul Lawrence, Google)

- eBPF = extended Berkeley Packet Filter
  - allows bytecode modules to be loaded into the kernel
  - modules written in C (or Rust) and compiled using clang
  - "Javascript for the kernel"
- eBPF was a common theme at LPC and also at the LF OSS conference (held just down the road)
- FUSE = File system in User Space
- FUSE is slow; this talk discussed some ideas that would speed things up by implementing some parts in eBPF
- Experimental; not in shipping code

# EROFS as a replacement for EXT4 and Squashfs (David Anderson, Google)

- EROFS is a read only filesystem, a successor to Squashfs
  - main feature is the ability to do decompression in place - better memory management
- EROFS is in mainline Linux from 5.4
- This talk discussed the advantages of using EROFS for the system partitions
- Reduced size, e.g. 3.9 GB system partition -> 2.5 GB (45% saving)
- Read speed comparable to Ext4

# Rust in the kernel

- This was discussed at the Kernel Summit, not the Android MC(\*)
- Linus plans to merge basic support for Rust in 6.1
- Linux: "Rust isn't that terrible in the end; it's not Perl"
  - minimal merge just to get the infrastructure into the kernel
  - it should build, but shouldn't do much of anything beyond the "hello, world" stage
  - expect it to be used for "leaf" code, i.e. drivers
  - not intended for deeper code (not yet)

(\*) I was not at the Kernel Summit