



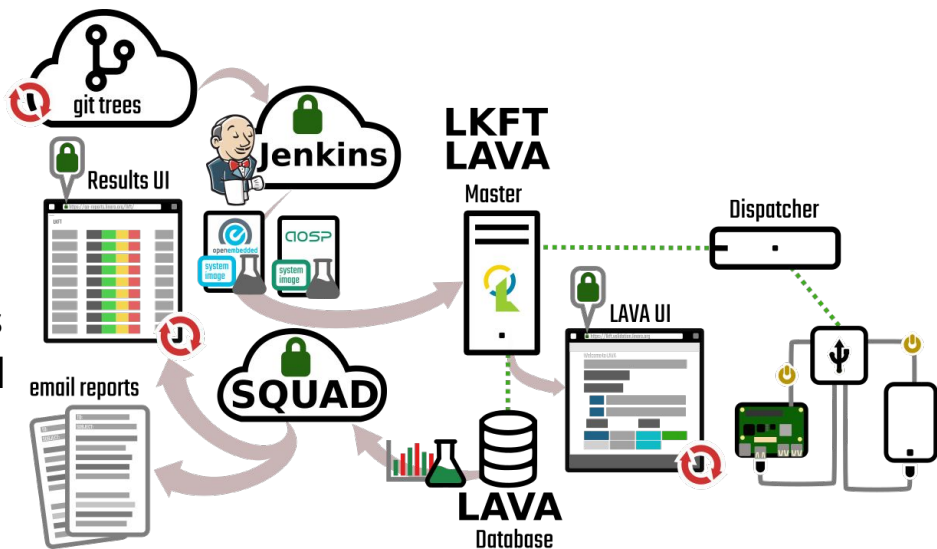
Linux Kernel Functional Testing (LKFT)

Functional Testing of Mainline, LTS and Android
Common Kernels

Linux Kernel Functional Test (LKFT) Components

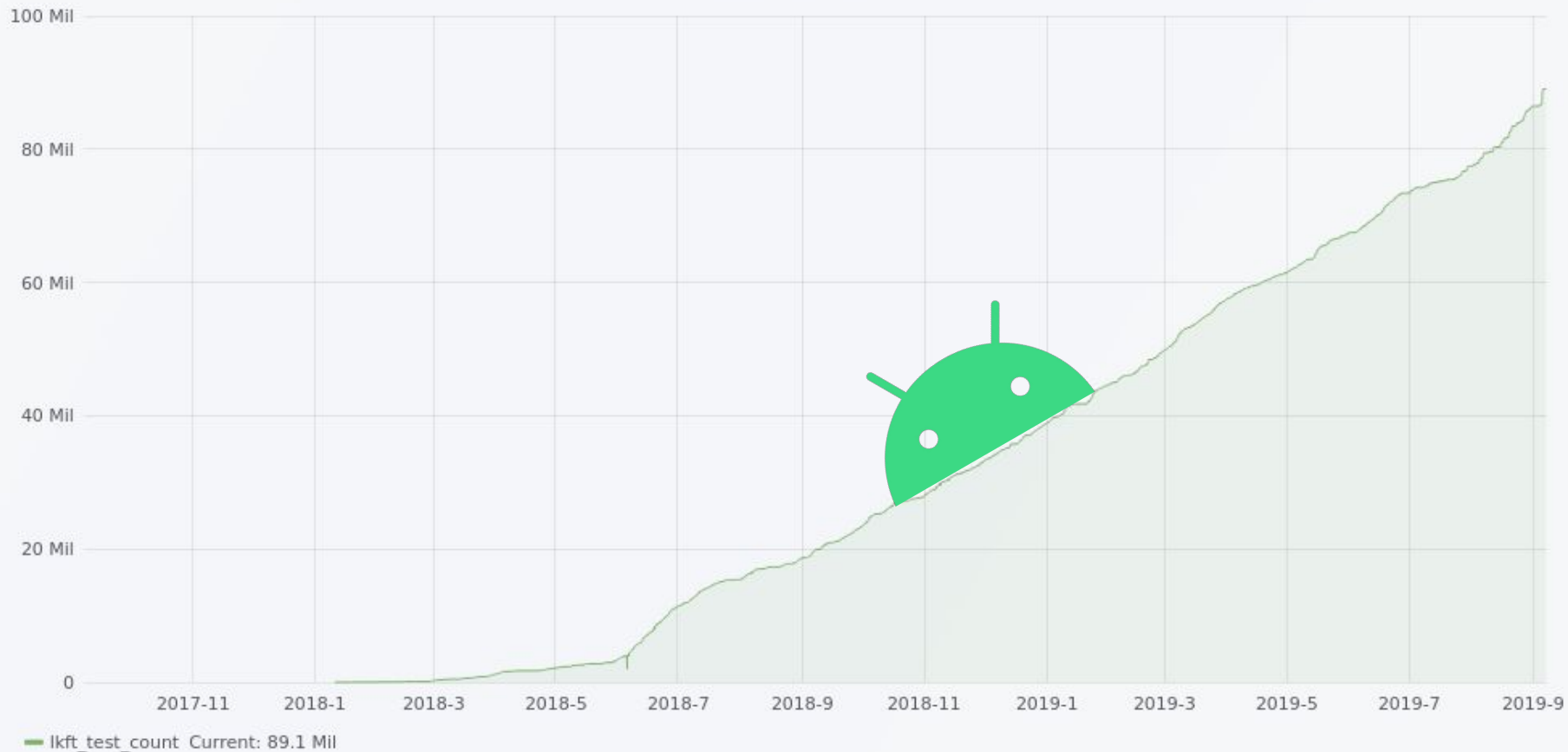
The [LKFT](#) framework is a collection of software tools and hardware devices. LKFT is composed of

- Lab - [test targets](#)
 - You can host your own remote lab
- Jenkins - Builds -> on demand builders
- Linaro Automated Validation Architecture ([LAVA](#)) - Manage the DUTs
- [SQUAD](#) - Scheduling, data analysis and reporting framework
- [Test definitions](#)



Backed by Linaro's Kernel Validation team to maintain, triage results, and escalate

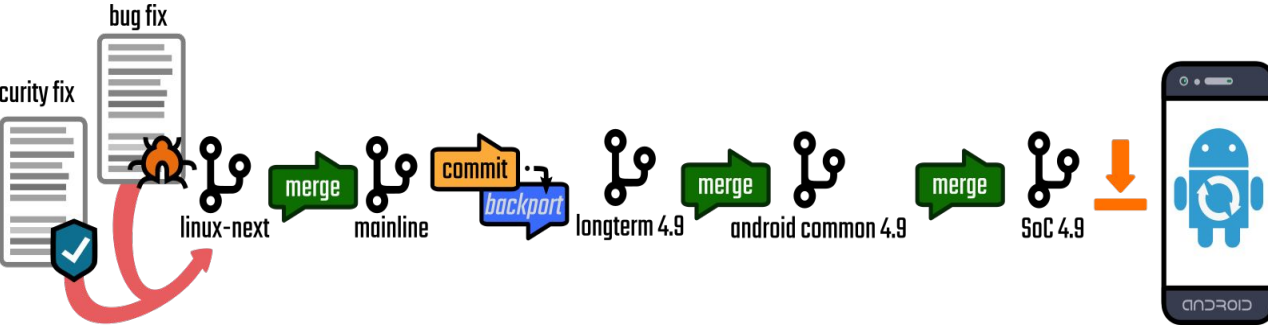
LKFT Tests Run



The Android robot is reproduced or modified from work created and shared by Google and used according to terms described in the [Creative Commons 3.0 Attribution License](https://creativecommons.org/licenses/by/3.0/).

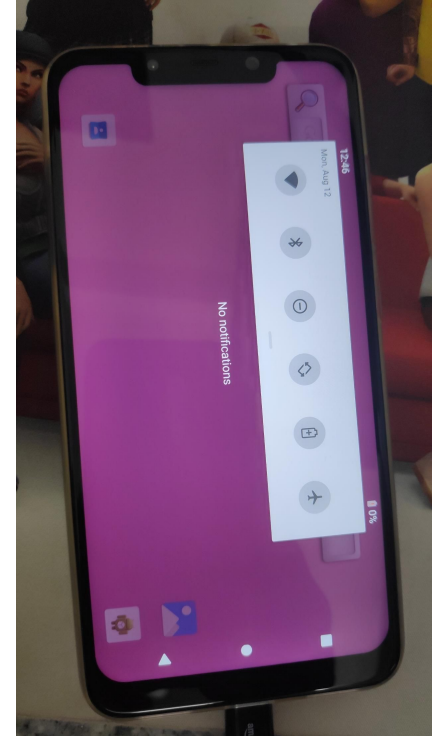


Mainline -> Stable -> Android Common Kernel



To the right is a Poco F1, booting Android with a mainline linux kernel (and a couple of patches but not many!) Pixel 3 boots mainline too! (with patches)

Being able to boot/run Android on mainline means we can test mainline, future stable and future LTS kernels at RC.



Boot testing is tired, Functional testing is wired

- CTS / VTS (kselftest, LTP)
 - Remember exercising the kernel is what we want
 - CTS pushes on Camera, BT, Networking
- Add energy probes, run benchmarks, workloads and look for kernel regressions in EAS
- Add a [Chamelium](#) board to capture video / sound
 - IGT
 - [Synthmark](#)
 - V4I2
 - libcamera
- Regressions get to be more than pass/fail, EAS regression uses statistical tests that spot anomalies

