Brillo Kernel Maintenance

Linux Plumber's Conference 2016, Santa Fe

Kees (“Case”) Cook
keescook@google.com
@kees_cook

https://outflux.net/slides/2016/lpc/brillo.pdf

Agenda

- Brillo
- Existing problems
- New problems
- Solutions!
- One kernel to rule them all
- Delta reduction
- LTS to LTS upgrades
- Test test test
- Sanity check

Brillo

- IoT with an Android stack
  - May not have screen for display/input
  - May have GPIO or other fun buses
- Mix of vendors from handsets to embedded
- Still relatively early in development

Existing problems

- kernel prebuilts
- multiple tree origins
  - upstream
  - Android common
  - Vendor tree
- kernel version is static to device
  - must backport fixes and features
  - must forward-port out-of-tree drivers
- exponential set of combinations to update/test

New problems

- Support must be at least 5 years after last device is sold
  - No one notices exponential work when it's small...

Solutions!

- One kernel: Reduce backporting work by keeping a single kernel
- Delta reduction: Reduce forward-porting work by keeping everything upstream
- If this is too scary … you're not testing thoroughly enough

One kernel to rule them all

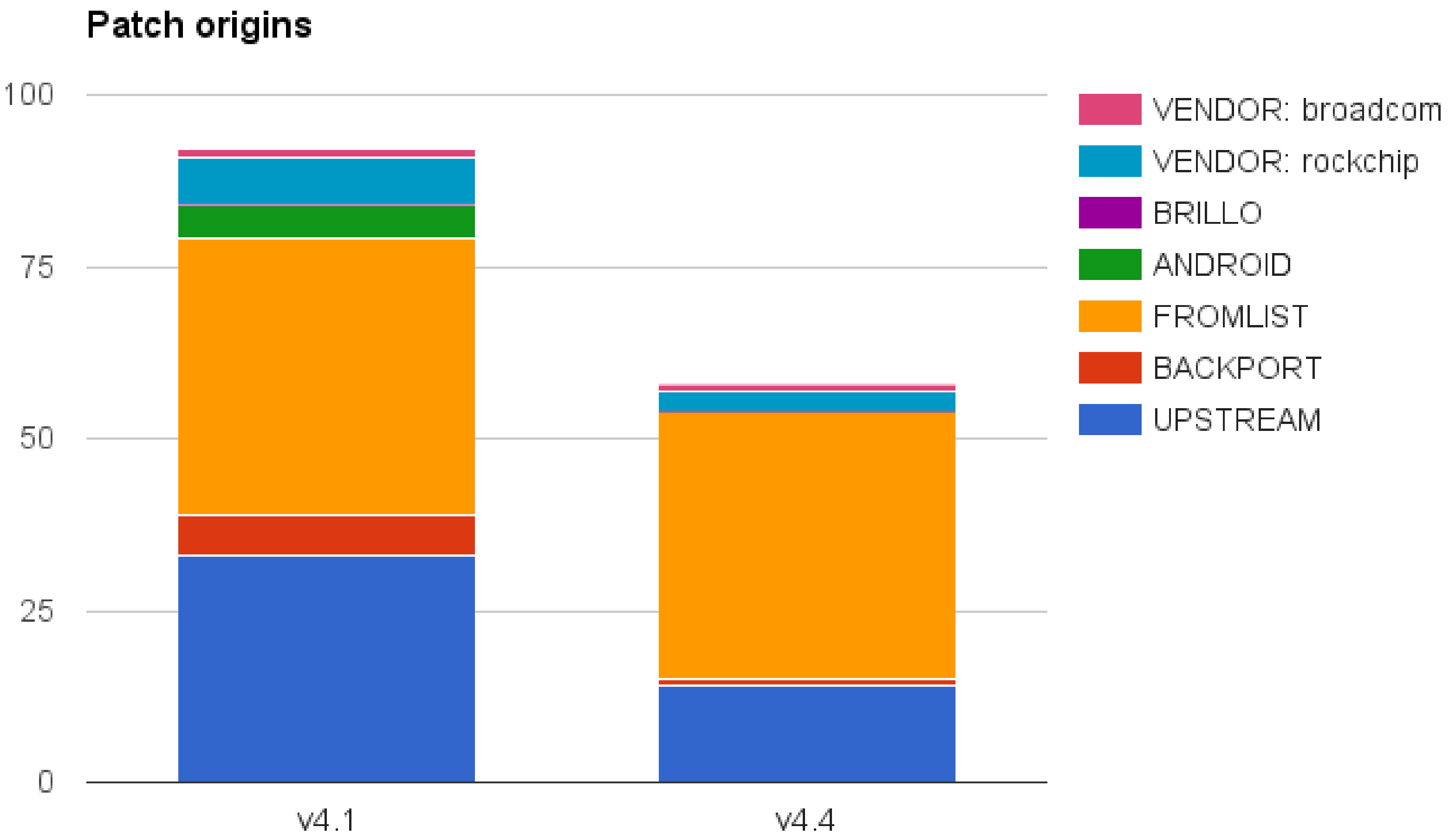
- Single in-tree kernel with Android patches and all vendor patches
- Per-product arch and CONFIG declarations
- Vendor patches must have at least been sent upstream, and are cherry-picked back
- Kernel is latest upstream LTS and receives regular -stable patches
- *Kernel version moves forward*

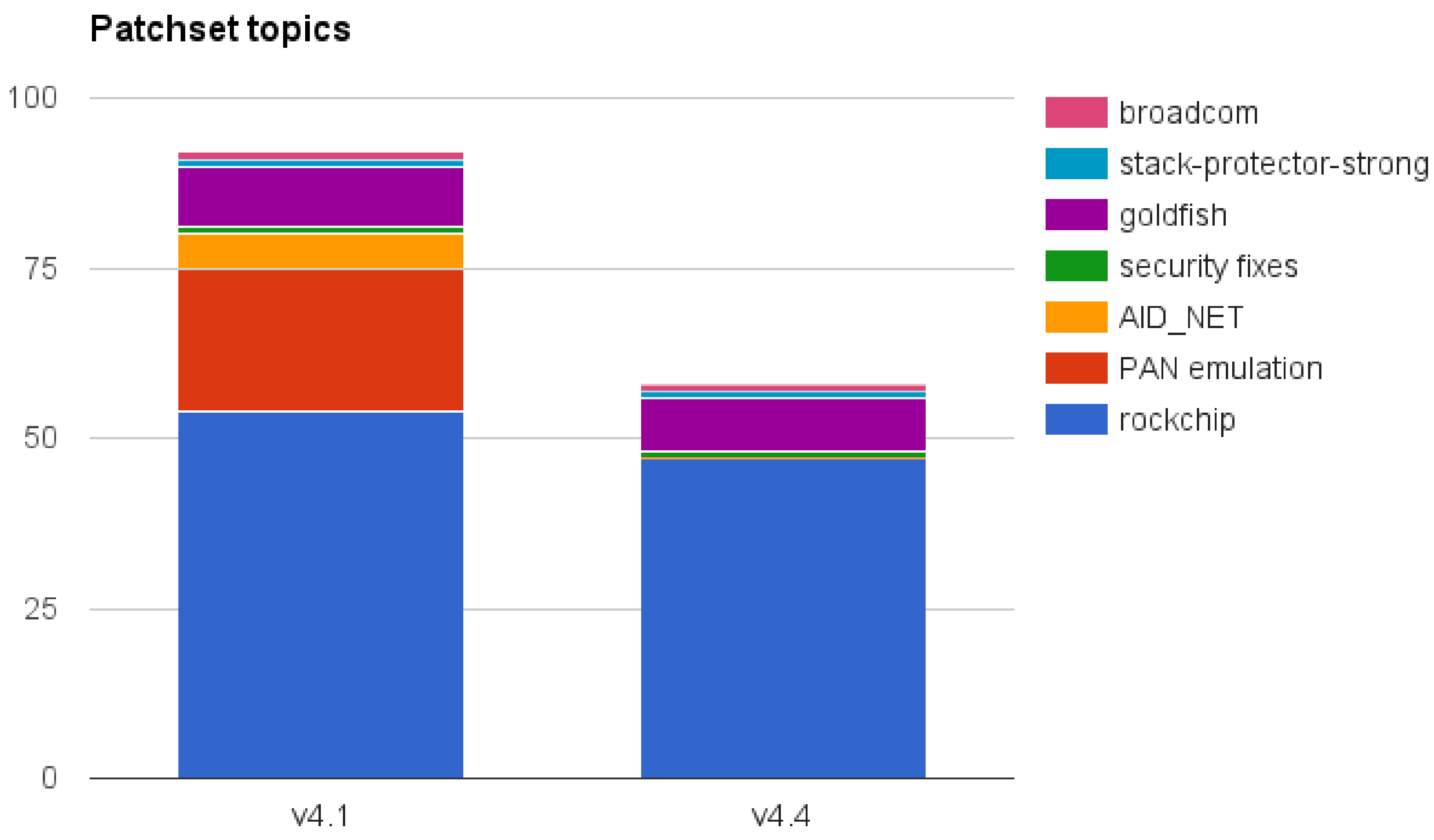
Delta reduction

- Android common kernel has over 600 patches on top of upstream
- Mattias Nissler consolidated these to under 150 patches
  - fix Android userspace to not need special cases, or use upstream alternatives
  - collapse small fixes into the corresponding feature patches and remove patch/revert pairs
  - upstream any low-hanging fruit

LTS to LTS upgrades

- Forward port anything not already in next LTS
- First attempt at this went smoothly: v4.1 → v4.4


Patch origins
100
75
50
25
0
v4.1
v4.4
VENDOR: broadcom
VENDOR: rockchip
BRILLO
ANDROID
FROMLIST
BACKPORT
UPSTREAM



Patchset topics

Test test test

- Afraid of regressions?
- Get code into upstream (ohai HiKey)
- Build automated tests
  - If you can't tell me what you're afraid of breaking then you can't tell me we shouldn't upgrade
  - If you can't test what you're afraid of breaking, how can you verify that it works in the first place?
  - Yes, it's hard. So is everything else, but you do this once
- Test linux-next and linux-stable (kernelci.org)
- Catch things before they're in the next LTS

Sanity check

- Will this work?
  - I really hope so! We have to try *something* to get out of the ancient kernel quagmire and off the backport treadmill.
- Will the vendors agree to this?
  - Most already have and are fairly proactive about upstreaming.
  - Those with reservations must decide if they're more terrified of the up-front costs of upstreaming or the long-term costs of having a support duration way beyond what they're used to.

Questions / Comments / Flames

Kees (“Case”) Cook
keescook@google.com
@kees_cook

https://outflux.net/slides/2016/lpc/brillo.pdf

https://goo.gl/IRlZ1B
https://android.googlesource.com/device/generic/brillo/+/master/docs/KernelDevelopmentGuide.md