What's new in the Linux kernel

and what's missing in Debian

Ben Hutchings · DebConf 19

Ben Hutchings

- Working on Linux kernel and related code for Debian and in paid jobs for over 10 years
- Debian kernel and LTS team member, doing about half the kernel packaging work
- Maintaining Linux 3.16.y stable update series on kernel.org

Linux releases early and often

- Linux is released about 5 times a year (plus stable updates every week or two)
- ...though some features aren't ready to use when they first appear in a release
- I'll talk about new features in Linux 4.19 to 5.2 inclusive
- Good news: we have lots of new kernel features in testing/unstable (and some made it into buster)
- Bad news: some of them will need changes elsewhere to be useful

Recap of previous years' features

- schedutil: still not the default cpufreq governor
- Zoned recording: dm-zoned-tools is packaged but needs a sponsor (#882640)
- Block layer replacement: completed [5.0]
- RISC-V: enough hardware support upstream that we can build a usable kernel image [4.19]
- GPUs: free drivers for Arm Mali – lima, panfrost [5.2]

Security hardening [ongoing] (1)

- Protected FIFOs and regular files in sticky directories (`/tmp`) – should we enable these by default?
- No variable-length arrays on kernel stack, reducing risk of overflow
- Kernel stack protector has per-task canaries on PowerPC, 32-bit Arm (needs compiler plugin), 64-bit Arm (needs gcc 9)
- More use of checked reference count type (`refcount_t`)

Security hardening [ongoing] (2)

- Mapping a kernel heap page to user-space now fails
- Some support for user-space Pointer Authentication Codes and Memory Tagging Extension on 64-bit Arm
- User-space access prevention (KUAP) enabled on Power 9
- `userfaultfd()` can be restricted to privileged users – should we do that by default?
- KASLR implemented on System z (s390x)

Speculation leak mitigation [ongoing] (1)

- Spectre variant 1: more array bounds checks protected
- Spectre variant 2: better protection for user-space (opt-in) and system firmware on x86 using IBPB, IBRS, STIBP
- L1TF: Intel x86 CPUs speculatively load from L1 cache using address bits in a non-Present (swapped) page table entry
  - For regular processes, mitigated by inverting physical address bits in non-Present entries (limits max. RAM and swap file size)
  - For VMs, problem is worse as this bypasses EPT. Mitigated by flushing L1 on VMEXIT, but full mitigation will require VMs never share a core with other processes

Speculation leak mitigation [ongoing] (2)

- MDS: Intel x86 CPUs speculatively load stale data from various buffers
  - Hard to exploit as attacker can't easily control what's left in these buffers
  - Mitigated by flushing buffers on kernel exit, with aid of new microcode – but older CPUs remain vulnerable
  - Full mitigation will require restrictions on processes sharing a core
- Running untrusted code on Intel CPUs with HT is risky, but still enabled by default
- Arch-independent parameter to control all mitigations: `mitigations=auto` / `=auto,nosmt` / `=off`

Y2038 [4.18-5.1]

- Kernel internally uses 64-bit `time_t` (or `ktime_t`) almost everywhere
  - Most filesystems still store 32-bit seconds; ext3/ext4 and btrfs are only exceptions so far
- New system calls using 64-bit `time_t` enabled on all 32-bit architectures
- glibc can provide time ABIs using both 32-bit and 64-bit `time_t` but ABI exposed through headers is chosen at glibc build time
  - Switching to 64-bit `time_t` on armhf or i386 will require an soversion bump in all other libraries exposing `time_t`
  - Discussion on debian-devel: “Options for 64-bit time_t support on 32-bit architectures”

CAKE [4.19]

- New network scheduler (`sch_cake` module), can be selected using `tc` command
- Based on existing FQ-CoDel which avoids bufferbloat
- Adds fairness between hosts, no matter how many flows they use
- Adds traffic shaping and limited priority (IP DiffServ) support
- Adds TCP ACK coalescing, important for connections with uplink much slower than downlink
- Should be a good choice for home NAT gateways

Pressure stall information [4.20]

- System load (`/proc/loadavg`) is weighted average number of tasks running, ready to run, or waiting for block I/O
  - If greater than number of CPUs, *might* indicate CPU contention
  - But doesn't distinguish CPU from I/O contention
- Pressure stall information shows average time spent waiting, divided into three categories:
  - `cpu` – ready to run, but waiting to be scheduled
  - `memory` – swapping in, re-faulting mmapped file, or reclaiming memory
  - `io` – waiting for block I/O to complete
- Information exposed under `/proc/pressure` and cgroupfs
- User-space can monitor by writing threshold to the file and polling [5.2]

Energy-Aware Scheduling [5.0]

- Arm-based SoCs can have mixture of fast cores and slower, more power-efficient cores — “big.LITTLE”
- Default task scheduler (CFS) originally designed for systems where all CPUs running Linux are the same
  - It gained support for differing CPU “capacities” earlier, but only used this to improve accounting and fair sharing
- EAS attempts to schedule tasks on CPU in an energy-efficient way:
  - Uses task's CPU load to predict which CPUs will have enough spare cycles
  - Uses an energy model populated from device tree to predict energy cost
  - On symmetric systems, or if all CPUs are busy, reverts to standard behaviour

Packaging changes

- Code signing completed and in production for buster
- Build logs are verbose by default (per policy)
- Replaced private patch system used for orig tarball
- Fixed almost all warnings/errors from gcc, dpkg-dev, lintian, pycodestyle, pyflakes
- libbpf packages on most architectures
- Changelog split – old entries in source package only
- Even more build profiles – `pkg.linux.no{kernel,source}`
- Reorganised udebs to be more consistent across architectures
- RNG drivers and more graphics drivers included in udebs

Questions?

Credits & License

- Content by Ben Hutchings
  www.decadent.org.uk/ben/talks/
  License: GPL-2+
- Original OpenOffice.org template by Raphaël Hertzog
  raphaelhertzog.com/go/ooo-template
  License: GPL-2+
- Background based on “Serenity” theme by Edward Padilla
  wiki.debian.org/DebianArt/Themes/serenity
  License: GPL-2