Embedded Linux Community Update
Feb 2022
Tim Bird
Principal Software Engineer, Sony Electronics
Nature of this talk…

- Quick overview of lots of embedded topics
- A springboard for further research
  - If you see something interesting, you have a link or something to search for
- Some overlap with material given previously
  - I may go quickly over some older slides
- Not comprehensive!
  - Just stuff that I saw
Outline

- Linux Kernel
- Technology Areas
- Conferences
- Industry News
- Resources
Outline

Linux Kernel
Technology Areas
Conferences
Industry News
Resources
Linux Kernel

- Versions
- Stuff In Progress
- Development Stats
- Development Tools and Workflow
Kernel Versions

- Linux v5.12 – 25 Apr 2021 – 70 days
- Linux v5.13 – 27 Jun 2021 – 63 days
- Linux v5.14 – 29 Aug 2021 – 63 days
- Linux v5.15 – 31 Oct 2021 – 63 days
  - “Trick or Treat” release
- Linux v5.16 – 9 Jan 2022 – 70 days
- Linux v5.17-rc2 – current version this week
  - Probably 5.17 will be released on 13 Mar 2022
Linux v5.12 (Apr 2021)

- Support for oprofile removed
  - superceded by perf events
- “PREEMPT_DYNAMIC” allows selecting preemption mode at boot or run time
- Dynamic thermal power management
  - Allow power usage of groups of devices to be capped to meet thermal constraints
- Nintendo 64 support (finally!)
  - Not sure how useful this is when the console only supports 8MB RAM, but hey!
Linux v5.12 (cont.)

- Build system can use Clang’s link-time optimization (LTO) features on ARM64 and x86 architectures
- kfence memory debugging tool has been added
- Some new perf-events features:
  - Can report on instruction latency
  - Daemon mode for long-running sessions
  - See [https://git.kernel.org/pub/scm/linux/kernel/git/torvalds/linux.git/commit/?id=3a36281a1719](https://git.kernel.org/pub/scm/linux/kernel/git/torvalds/linux.git/commit/?id=3a36281a1719)
- Support for Playstation DualSense game controllers *(yeah Sony!)*
Linux v5.13 (June 2021)

- Support for Control-Flow Integrity (CFI)
- Software-interrupt processing code from the Preempt RT tree was mainlined
- logbuf_lock used by printk has been removed
  - working towards lockless printk
- Support for Generic USB Display (GUD) driver
- BPF programs can now call (some) kernel functions directly
- /dev/kmem was removed (!!!)
- Added new WWAN networking framework
- Landlock security module

(Many of these explained later)
Linux v5.14 (August 2021)

- memfd_secret system call was added
  - Details on next slide
- new tracers
  - osnoise - show application delays caused by kernel activity
  - timerlat – detailed info about timer-based wakeups
- A fair amount of Qualcomm and MediaTek driver code
  - clocks, pin controllers, sound
- "simplesdrm" driver
  - direct-rendering interface for simple framebuffer devices
- Kunit can run tests under QEMU (in addition to native and UML)
memfd_secret system call

• Creates a region of memory that even the kernel cannot directly access
  • Pages are removed from the kernel’s direct map
  • Intended to be used for cryptographic info (e.g. keys)
• Makes it difficult for other processes or even the kernel to unintentionally (or even intentionally) access the memory
  • See https://lwn.net/Articles/835342/
  • For many more details, see https://git.kernel.org/pub/scm/linux/kernel/git/torvalds/linux.git/commit/?id=1507f51255c9
• Realtime preemption locking code – “Sleeping spinlocks”
  • It’s a big deal - More on this later
• More io_uring performance enhancements
  • “BIO recycling”
• Core scheduler support for asymmetric systems
  • Cores on the same chip that can run either 64-bit or 32-bit
    • How to deal with scheduling when the processor can’t even execute some code
  • See https://lwn.net/Articles/838339/
• ksmbd – in-kernel SMB server (!!)
  • Not a replacement for Samba, but provides better optimization for Linux in some situations
Linux v5.15 (cont.)

- printk indexing
  - Can extract all printk messages from kernel
    - Is used to detect changes that could break log-parsing tools
- DAMON system merged (Data Access Monitor)
  - See next page
- kernel new uses –Werror flag during build, by default
  - A compiler warning will cause the build to fail
- If LLVM=1 (env var), you don’t need to specify CROSS_COMPILE
- Minimum gcc version is now 5.1
DAMON system

- DAMON = Data Access Monitoring tool
- Provides tools to record data access and show visualizations of access patterns
- Different visualizations available
  - A heatmap of memory access for your workload
  - Graphs showing information about working set size
- See https://damonitor.github.io/doc/html/v17/admin-guide/mm/damon/index.html
- Nice diagnostic tool... but is it actually more?
Linux v5.16 (January 2022)

- EROFS (Enhanced Read-Only FS) continues to get new features
  - Multiple-device support
  - io_uring operations can now have security policies enforced by SELinux or Smack
- First set of patches for folios
  - New memory data type (described later)
- DAMON operation schemes added
  - DAMON can perform pro-active page reclaim, and monitor the physical address space
- See https://lwn.net/Articles/863753/
Linux v5.17 (expected in March)

- Random number generator replaced SHA1 with BLAKE2 hash function
  - Results in 370% increase in RNG performance
- RTLA – realtime analysis tools have been added
  - osnoise and timerlat (not sure how these are different from the tracers added in 5.4)
- Some changes to flags fields used in FUSE_INIT call
  - Check your FUSE filesystems and tools for compatibility
Stuff In Progress

- A few things being worked on
  - Page folios
  - Multi-generational LRU
  - Rust for the kernel
  - Zero-copy network transmissions with io_uring
- See https://lwn.net/Articles/881675/
Folios

• Folio = New data type to indicate a pointer to a page that is NOT the tail of a compound page
  • Basically, it’s an internal typing improvement to memory management
  • To avoid issues with passing a pointer to a page (part of a compound page) with wrong attributes
    • Sometimes, a routine doesn’t work properly with a tail page
  • Some kernel devs like them, and some don’t
• See https://lwn.net/Articles/849538/
• Some parts of folio code adopted in 5.16 and 5.17
  • Still not actually used yet
Multi-generational LRU (MGLRU)

- Currently have 2 queues for managing page eviction
  - Active and inactive
- MGLRU propose multiple queues and a more complex algorithm
  - Less CPU overhead
  - Better working set estimation
  - Proactive reclaim (which lowers memory pressure)
- Big debate about whether it could be merged in 5.17
  - It wasn’t merged, but it has supporters
    - Google says they’re already using it and it’s working well
Rust for the Linux kernel

- People want to support Rust code in the Linux kernel
- Third version of the Rust support patch was posted in January
- It's a difficult problem because Rust programs don't normally have to deal with memory allocation failures
  - Requires a modified memory allocator that can handle failures
- Lots of features are still unstable
- Has not been accepted into kernel yet, but work is proceeding
Zero-copy networking with io_uring

- RFC patch by Pavel Begunkov
  - Not sure when it would be ready for upstreaming
- About 1.5 to 2 times faster than current (socket-based) zero-copy networking
- See https://lwn.net/Articles/879724/
Linux 5.16 developer stats

• Most active 5.16 developers
• By changesets:

<table>
<thead>
<tr>
<th>Person</th>
<th>Changesets</th>
<th>Percent</th>
</tr>
</thead>
<tbody>
<tr>
<td>Michael Straube</td>
<td>286</td>
<td>2.0%</td>
</tr>
<tr>
<td>Cai Huoquing</td>
<td>232</td>
<td>1.6%</td>
</tr>
<tr>
<td>Jakub Kicinski</td>
<td>200</td>
<td>1.4%</td>
</tr>
<tr>
<td>Christoph Hellwig</td>
<td>158</td>
<td>1.1%</td>
</tr>
<tr>
<td>Bart Van Assche</td>
<td>157</td>
<td>1.1%</td>
</tr>
</tbody>
</table>

• Michael Straube worked on the r8188eu wireless network driver

Table data: https://lwn.net/Articles/880699/
Linux 5.16 developer stats

• Most active 5.16 developers
• By lines of code:

<table>
<thead>
<tr>
<th>Person</th>
<th>Lines changed</th>
<th>Percent</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ping-Ke Shih</td>
<td>91116</td>
<td>11.4%</td>
</tr>
<tr>
<td>Zhan Liu</td>
<td>34501</td>
<td>4.3%</td>
</tr>
<tr>
<td>Nick Terrell</td>
<td>28611</td>
<td>3.6%</td>
</tr>
<tr>
<td>Sameer Pujar</td>
<td>15121</td>
<td>1.9%</td>
</tr>
<tr>
<td>Johan Almbladh</td>
<td>13901</td>
<td>1.7%</td>
</tr>
</tbody>
</table>

Ping-Ke Shih submitted the Realtek rtw89 driver (which skipped the staging tree)
Most active employers for 5.16

- Most active active employees for the 5.16 kernel:

<table>
<thead>
<tr>
<th>Company</th>
<th>Changsets</th>
<th>Percent</th>
</tr>
</thead>
<tbody>
<tr>
<td>Intel</td>
<td>1454</td>
<td>10.2%</td>
</tr>
<tr>
<td>(Unknown)</td>
<td>1196</td>
<td>8.4%</td>
</tr>
<tr>
<td>Google</td>
<td>932</td>
<td>6.6%</td>
</tr>
<tr>
<td>(None)</td>
<td>781</td>
<td>5.5%</td>
</tr>
<tr>
<td>Red Hat</td>
<td>765</td>
<td>5.4%</td>
</tr>
</tbody>
</table>

- “Unknown” consisted of about 400 developers, most with one or two patches
  - Some are “drive by” contributors, and some will stay around

Table data: https://lwn.net/Articles/880599/
Kernel commit log entries

- Number of commit log entries, per kernel version

<table>
<thead>
<tr>
<th>Company</th>
<th>git log count</th>
<th>developer count</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.12</td>
<td>14167</td>
<td>1938</td>
</tr>
<tr>
<td>5.13</td>
<td>17187</td>
<td>2131</td>
</tr>
<tr>
<td>5.14</td>
<td>15871</td>
<td>1982</td>
</tr>
<tr>
<td>5.15</td>
<td>13473</td>
<td>1856</td>
</tr>
<tr>
<td>5.16</td>
<td>15384</td>
<td>2074</td>
</tr>
<tr>
<td>5.17-rc2*</td>
<td>12264*</td>
<td>1715*</td>
</tr>
</tbody>
</table>

- Extracted using ‘git log v5.yy..v5.zz –oneline | wc –l’ and ‘author-stats v5.yy..v5.zz | wc –l’

Table data: [https://lwn.net/Articles/880599/](https://lwn.net/Articles/880599/)
Kernel Development

- Lore + Lei
- Using github for kernel work
Lore + Lei

- Lore archive has become indispensable for kernel devs
  - [https://lore.kernel.org/](https://lore.kernel.org/) and [https://lore.kernel.org/all](https://lore.kernel.org/all)
  - Searches are very fast, even over all repositories
  - Can do smart searches (AND, OR, NOT, regexes, and context qualifiers)
    - s: = match within Subject
    - nq: = in non-quoted portion of text
    - dfn: = match filename from diff
    - dfhh: = match diff hunk header (usually a function name)
  - See [https://lore.kernel.org/all/_/text/help/](https://lore.kernel.org/all/_/text/help/)
- LEI = local email interface
  - Tool to search lore, and create a local mail box with results
Lei workflow

• Lei is great for kernel developers
  • It allows them to avoid subscribing to email lists
    • While still catching items of interest (no matter which list it appears on)
• Developer can create a query for items of interest
  • Lei remembers query, so user can issue it again
  • Example query on next page
• Can process local mailbox offline, using familiar mail tools
  • e.g. Can use ‘mutt’ instead of web interface or GUI
• See https://lwn.net/Articles/878205/
Sample lei search

• Here’s a query that a floppy driver maintainer might use:
  • lei q -l https://lore.kernel.org/all/ -o ~/Mail/floppy \
    --threads --dedupe=mid \n    ' (dfn:drivers/block/floppy.c OR dfhh:floppy_* OR s:floppy \n      OR ((nq:bug OR nq:regression) AND nq:floppy)) \n      AND rt:1.month.ago.’

• Which means to store emails that:
  • have a patch that touches floppy.c (dfn:drivers/block/floppy.c)
  • have a patch that changes a function whose name starts with floppy_ (dfhh:floppy_*)
  • have "floppy" in the subject (s:floppy)
  • mention both "floppy" and either "bug" or "regression" in non-quoted text ((nq:bug OR nq:regression) AND nq:floppy)
  • were received in the last month (rt:1.month.ago)
Konstantin Ryabitsev is working on a bot that can turn a github pull request into an email patch series.

Idea is to allow a developer using Github* to submit a patch:

- *(or Gitlab or some other git “forge”)
- Works around difficulty of setting up an e-mail-based workflow
- It handles things like running checkpatch.pl and get_maintainers

Some maintainers are wary about allowing this workflow:

- In their view, Github doesn’t promote some desirable practices
  - e.g. There’s no way to review a commit message on github, so commit messages tend to be poor

See https://lwn.net/Articles/860607/
Outline

Linux Kernel
Technology Areas
Conferences
Industry News
Resources
Technology Areas

- Audio
- Core Kernel
- Filesystems
- Graphics
- Languages

- Networking
- Security
- Testing
- Toolchains
Audio

- PipeWire continues to gain ground
  - Intended to replace PulseAudio and JACK
  - Is higher performance
  - WirePlumber is a new session manager for PipeWire
    - Scriptable in LUA
  - See Talk by Geoge Kiagiadakis (Collabora) from ELC 2021
Core Kernel

- memfd_secret (v5.14)
- printk_indexing (v5.15)
- scheduling for asymmetric processors (v5.15)
- RNG speedups (using BLAKE2 hash) (v5.17)
Filesystems

- io_uring continues to mature
  - As a reminder: see https://lwn.net/Articles/810414/
  - Performance enhancements (v5.15)
  - Can be security-regulated by SELinux or Smack (v5.16)
  - Support for zero-copy networking (coming)
- EROFS and F2FS continue to mature
  - Better compression support
  - Better xattr support
- FUSE_INIT flag changes (v5.17)
Graphics

- Generic USB Display driver mainlined in v5.13
  - Allows to push graphics and video over USB
  - Can be used to turn a raspberry PI zero into a USB-to-HDMI adapter
    • See https://github.com/notro/gud/wiki
- “simpledrm” driver merged in 5.14
- legacy fbdev sub-system gets a maintainer (Jan, 2022)
  • But some people didn’t like it (see next page)
Reviving fbdev

- New maintainer was found for fbdev code: Helge Deller
  - Maintainer change was a bit too quick for some people
    - But the status had been listed as “Orphan” in MAINTAINERS
- Controversy was over restoration of hardware acceleration for scrolling
  - Revert of a patch removing hardware acceleration for 2D scrolling for legacy fbdev devices
- Quite frankly, it looks like the DRM devs having a turf war over something they weren’t maintaining
- IMHO should let the new maintainer try and see what happens
- See https://lwn.net/Articles/881827/
Languages

- Rust
- Python
Rust

• Rust being applied to security-sensitive tools
  • Rust version of GNU coreutils
    • Work is progressing
    • See https://www.phoronix.com/scan.php?page=news_item&px=Rust-Coreutils-Jan-2022
  • MIT license (!)

• Rust in the kernel
Rust in the Linux kernel

- People are working to support the Rust language for kernel driver development
  - Have been talking about it for a while
  - Posted RFC to kernel mailing list in April
- Big advantage (claimed) is memory safety features of Rust built in to the language
- Reaction of kernel developers is mixed:
  - Don’t want to require kernel developers to know more languages
  - Not sure benefit is worth the cost
  - Most maintainers seem to have a “wait and see” attitude
- 3rd version of system posted in January
Python

- Some distros have disabled python2 by default (SUSE)
- Cpython is getting faster
  - [https://lwn.net/Articles/857754/](https://lwn.net/Articles/857754/)
- Python 3.10.0 released Oct. 4, 2021
  - Better error messages
  - Better debugging
  - Structural pattern matching (case statements with matches)
- Having issues with removal of deprecated functions
  - Breaks backwards compatibility
  - Developers can’t seem to leave the language alone
  - See [https://lwn.net/Articles/883391/](https://lwn.net/Articles/883391/)
Networking

- Wireless WAN (WWAN) framework added to kernel (v5.13)
  - Also known as “Mobile Broadband”
  - Usually provided by cellular networks such as 2G, 3G, 4G LTE or 5G and cellular modems
  - First user is a Qualcomm wwan modem driver
  - Adds the concept of a WWAN port, which is a logical pipe to a modem control protocol
  - See code in drivers/net/wwan
    - Unfortunately, there doesn’t seem to be any docs under Documentation
  - See https://www.phoronix.com/scan.php?page=news_item&px=Linux-WWAN-Subsystem-v9
• Always a stream of oddball networking features and enhancements:
  • Custom configuration of hash policies for multipath IP traffic
  • Support for per-VLAN multicast
  • Support for Management Component Transport Protocol (MCTP)
  • Unix-domain sockets now support out-of-band data
  • SO_RESERVE_MEM can reserve kernel memory and speed up some operations
  • Support for Automatic Multicast Tunneling (ATM)
  • New sysctl knobs for tuning the ARP cache behavior
  • And so on...
Real-Time

- PREEMPT_DYNAMIC allows selecting preemption mode at boot time or run time (v5.12)
  - Can be “none”, “voluntary”, or “full”
  - There’s an option under debugfs for controlling the mode at run time
- rtda – real-time Linux analysis tool (5.17)
  - See https://www.phoronix.com/scan.php?page=news_item&px=Linux-5.17-RTLA
- PREEMPT_RT status
  - Software-interrupt processing code mainlined (v5.13)
  - Sleeping locks was mainlined (v5.16)
RT preemption locking code

- Provides “sleeping spinlocks” (and sleeping rwlocks)
  - Allows for process switch (schedule) while a lock is held, which is the core feature of PREEMPT_RT
- Must turn on CONFIG_PREEMPT_RT config option
  - Extensively tested to verify that non-RT kernels are not affected
  - See also CONFIG_RT_MUTEXES
- See the commit for details: https://git.kernel.org/pub/scm/linux/kernel/git/torvalds/linux.git/commit/?id=e5e726f7bb9f
- Merged in kernel v5.15
  - After 17 years of development effort and many reworks and refactorings
What’s left

• What’s left in PREEMPT_RT patches out of mainline:
  • About 3,000 lines of code, affecting 133 files (in 101 patches) (!!)
    • Some big changes to printk
    • Some changes to the random number generator, 8250 serial driver, the core scheduler, cgroups, tracing, memcontrol, i915 DRM driver
  • People are anxious for Linux RT without having to apply a patch

• See
Security

- Kernel hardening
- Control-flow integrity
- Landlock security module
Kernel hardening

- Control-Flow Integrity (v5.13)
- strict memcpy() bounds checking (v5.16)
  - https://lwn.net/Articles/864521/
- Spectre mitigations
  - There always seem to be new speculative execution vulnerability mitigations
  - Interestingly, v5.16 removed some Spectre-mitigation behavior for seccomp()
    - Devs decided that the extra mitigations weren’t really buying more security
Control-Flow Integrity (CFI)

- Mainlined in v5.13 kernel
- CFI checks that indirect call goes to function with same signature as expected
  - At the start of the function
- Detects if malicious code has changed the destination site for an indirect call
  - There are literally thousands of indirect calls in the kernel
- Feature merged in v5.13 is referred to as “forward-edge CFI”
  - There are separate mechanisms for guarding returns (“backward-edge CFI”)
Control-Flow Integrity (cont.)

- Requires Clang and Link Time Optimization
  - All indirect call targets are analyzed at compile time and put into jump tables
    - Loadable modules are handled separately
- Check is made at execution time that indirect call is to a valid target
  - Incurs less than 1% overhead (claimed)
- ARM64 supported now
  - x86 support is in the works
- See https://lwn.net/Articles/856514/ and https://lwn.net/Articles/810077/
Other security bits

- Landlock Security Module (v5.13)
- polkit vulnerability fix
  - in-progress patch to prevent argc == 0 problem
Landlock security module

- Landlock security module (new in v5.13)
  - Allows a process to be converted to a secure mode where the kernel can do additional validation of file system operations
  - Allows attaching a ruleset to a process, that allows the system to manage access control to files
  - Allows regular users to create secure execution environments
    - Not just sysadmins and root
  - Accepting a new security module is a big deal
    - There are not many Linux security modules in the kernel
    - This one took over 5 years and 74 revisions to get accepted
- See https://lwn.net/Articles/703876/
Landlock sample

- See demonstration program in kernel source tree at samples/landlock/sandboxer.c
  - Set LANDLOCK_ALLOWED environment variable to a list of directories
  - Start a program with the ‘sandboxer’ app
  - Program can only access files under the specified directories
Testing

- Systems
- Suites of tests
- Board farms and test hardware
Test Systems

- **Systems:**
  - KernelCI – has added kselftest git repo to list of trees it tests
  - Syzbot – always producing more fuzzing failure cases
  - CKI – providing many reports to upstream
  - Fuego
  - CompassCI
  - LKFT – providing many reports to upstream
Test Suites

- **LTP**
  - Latest release: 20220121 (Jan 21, 2022)
    - New tests for quotactl, statx, epoll, creat, dup, and fixes for others
    - Features (e.g. metadata parser) to make the test executor smarter
    - See https://github.com/linux-test-project/ltp/releases

- **kselftest**
  - New tests for ALSA, arm64, BPF, DAMON, network drivers, ftrace, gpio, kvm, network forwarding, netfilter, powerpc, RCU and others

- **Kunit**
  - Documentation cleanup, output format cleanup
  - Ability to run under QEMU
Board Farm APIs

- Standardized Board Farm and hardware testing APIs proposed by TimeSys and Sony
  - Proposed APIs for Power Measurement, Camera and video, and Serial port testing

- Multi-function DUT control board by Pengutronix
Toolchains

- **GCC**
  - Kernel now requires gcc 5.1 to build
  - GCC 11.2 released July 28, 2021
    - See https://gcc.gnu.org/gcc-11/changes.html
- **LLVM**
  - For kernel builds, use of LLVM does not require CROSS_COMPILE environment variable (or make variable)
  - LLVM 13.0.0 released October 4, 2021
    - https://releases.llvm.org/13.0.0/docs/ReleaseNotes.html
Outline

- Linux Kernel
- Technology Areas
- Conferences
- Industry News
- Resources
Conferences – 2021

- ELC 2021
- Open Source Summit Japan, 2021
Embedded Linux Conference 2021

- Hybrid (Seattle and virtual), September 2021
  - Was “ELC Europe”, but had to combine into a single event in 2021
- Was challenging:
  - Many people could not attend ELC in-person
  - Only a few talks on-site
- Still good content:
  - [https://elinux.org/ELC_2021_Presentations](https://elinux.org/ELC_2021_Presentations)
  - Slides and videos are available (with a few exceptions), linked from that page
Open Source Summit Japan, 2021

• Changed to December, and went virtual
• Schedule (with slides) at: https://ossalsjp21.sched.com/
• Video playlist: https://www.youtube.com/playlist?list=PLbzoR-pLrL6rfdmASBlf5hioXQF26gJhQ
Conferences - 2022

- Embedded Linux Conference North America
  - June 22-24, Austin, Texas, USA
  - Hybrid event – in-person and virtual
  - CFP is open now (closes March 14)

- Embedded Linux Conference Europe
  - Sept 13-16, Dublin, Ireland
  - Hybrid event – in-person and virtual
  - First in-person event in Europe after COVID-19 pandemic
COVID-19 issues?

- ELC 2021 was first in-person event by Linux Foundation
  - (well, Open Source Summit 2021, in Seattle)
- Things looked like they would open up (June 2021), but then didn’t
  - Delta and Omicron variants caused resurgences
  - Continuing pandemic posed problems for ELC 2021
- Events will continue with hybrid style for many months
- We are still learning how to handle hybrid events
  - How to give all attendees (both in-person and virtual) as good an experience as possible?
  - How to help in-person and virtual attendees mingle, for that “hallway networking” effect that is so valuable?
Becoming an “umbrella” event

- In Austin, there are 13 sub-events, including ELC
  - LinuxCon, ELC, Embedded IOT Summit, CloudOpen, ContainerCon
  - OSPOCon, Community Leadership Conference
  - Supplychain SecurityCon, Critical Software Summit
  - Open AI + Data Forum
  - Diversity Empowerment Summit, Open Source On-Ramp
  - Emerging OS Forum
    - Metaverse, WebAssembly, Public health, Climate, Crypto/Blockchain
Outline

Linux Kernel
Technology Areas
Conferences
Industry News
Resources
Industry News

- Trade associations
  - Linux Foundation
- Security news
- Interesting cases of embedded Linux
Linux Foundation

- Training and mentorship has really ramped up:
  - 2M trainings and exams delivered as of Jan 2022
- LFX tools for managing projects
  - Project insights, security, mentorship, crowdfunding, events, training, control center
  - See https://lfx.dev/
- 2021 Annual Report at:
<table>
<thead>
<tr>
<th>Metric</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Contributing Developers</td>
<td>739,010</td>
</tr>
<tr>
<td>Contributing Companies</td>
<td>11,169</td>
</tr>
<tr>
<td>Lines of Code Added Weekly</td>
<td>30.6M</td>
</tr>
<tr>
<td>Lines of Code Deleted Weekly</td>
<td>14.6M</td>
</tr>
<tr>
<td>Project Builds</td>
<td>2.1M</td>
</tr>
<tr>
<td>Logged Issues</td>
<td>963,390</td>
</tr>
<tr>
<td>Repositories</td>
<td>13,055</td>
</tr>
<tr>
<td>Vulnerabilities Detected</td>
<td>263,499</td>
</tr>
<tr>
<td>Vulnerabilities Fixed</td>
<td>13,802</td>
</tr>
<tr>
<td>Free Course Enrollments</td>
<td>200,000</td>
</tr>
<tr>
<td>Event Attendees In Last 12 Months</td>
<td>67,661</td>
</tr>
<tr>
<td>Email Messages Sent</td>
<td>2.7M</td>
</tr>
<tr>
<td>Chat Messages Sent</td>
<td>2.1M</td>
</tr>
<tr>
<td>CLAs Signed</td>
<td>35,158</td>
</tr>
<tr>
<td>Mentees Applied</td>
<td>10,511</td>
</tr>
<tr>
<td>2020 Community Meetings</td>
<td>26,998</td>
</tr>
</tbody>
</table>
LF Projects

- Initiatives with recent interesting activities:
  - Linux Foundation Research
  - Open Source Security Foundation
  - Internet Security Research Group

- Projects relevant to embedded
  - Yocto Project, Zephyr, Linux kernel, KernelCI, Automotive Grade Linux, Automated Compliance Toolkit
Linux Foundation Research

• New initiative to measure, analyze and describe the impact of open source collaborations
• Will use data from LF projects and tools (e.g. LFX), as well as other sources, to prepare reports
• Recent reports:
  • The state of SBOM and Cybersecurity Readiness
  • Hyperledger Brand Study
  • Diversity, Equity, and Inclusion in Open Source
  • Data and Storage Trends
  • The State of Open Source in Financial Services Report
Open Source Security Foundation

- **Initiatives:**
  - Security Scorecard
  - Security Reviews
  - Security Metrics Dashboard
  - Package Feeds / Package Analysis
  - CII Best Practices Badge Program

- **Standards:**
  - Open Source Vulnerability Schema
  - Supply-Chain Levels for Software Artifacts

- **Guides and Training:**
  - OSS Vulnerability Guide
  - Free Security Software Development courses: see https://openssf.org/training/courses/
Internet Security Research Group

- Operates “Let’s encrypt”
  - The world’s largest certificate authority
  - To encourage https over http
- Prossimo
  - Improving memory safety for security-sensitive software infrastructure
  - Utilizing memory-safe languages
- Prio
  - Privacy-preserving metrics service
  - Used by Apple and Google for Covid-19 Exposure Notification app
Industry Security news

- polkit security vulnerability
- log4j
- Software Bill of Materials (SBOMs)
Polkit security vulnerability

• vulnerability due to polkit being setuid, and not handling case where argc == 0
  • polkit assumes there is always argv[0], and started processing at argv[1]
    • Allowed attackers to inject items from environment as if they were part of command line (and change them on the fly)
  • POSIX is ambiguous about where argc == 0 is allowed
  • Easy to fix: chmod 755 /usr/bin/pkexec (remove setuid bit)
  • Kernel may support a fix also
• It’s an old bug, that almost got fixed 12 years ago
• See https://lwn.net/Articles/882799
log4j

• Is a java logging library used by over 30,000 projects
• Vulnerability was due to calling an external service as part of interpolating the logging string
  • The loggin string included elements provided by the user
• Fix to log4j itself was actually pretty fast and easy
• However, the way log4j was embedded in projects made it difficult for server administrators to detect if they were affected, and fix it
• Need better methods to track component dependencies
  • And also, better security audits and practices for widely-use modules
Software Bill of Materials

• Is the new buzzword for supply chain security
• SBOMs = way to track and validate source of software as it moves through supply chain
  • Intended to detect and prevent a “Solar Winds” style attack
• LF research did a survey on SBOM and Cybersecurity readiness
• See Automating Compliance Tooling (ACT)
  https://github.com/act-project/TAC
  • Umbrella project including: Fossology, OSS Review Toolkit, SPDX Tools and Tern (SBOMS for containers)
White House summit on Open Source security

- Linux Foundation was invited to attend Open Source security summit at White House in the US
  - Discussed security issue with OSS in the supply chain
  - Presented status of Open Source Security Foundation projects
    - Best Practices
    - Identifying Critical Projects
    - Metrics and Scorecards
  - Described Project Sigstore
- Many important companies and groups in attendance
  - Google issued a statement of support for public (government) investment in OSS security
Interesting embedded Linux uses

- Starlink satellite count
  - 1552 as of Jan 10, 2022
    - That’s over 74000 Linux nodes in low earth orbit
  - Skynet - it’s under construction
- GregKH’s LTS status signboard
  - See https://twitter.com/gregkh/status/1483330885655183360
- Mars helicopter
Mars Helicopter - Ingenuity
Mars Helicopter

- Mars Ingenuity Helicopter landed in February, 2021 on Mars
- Performed tests and demonstrations in April & May
  - First 5 flights were part of “Technology Demonstration”
  - Didn’t formulate a plan for continued flights until later
- Is still flying...
  - Has performed 18 flights so far
Mars Helicopter Software Bugs

- Flight 4 (first attempt) - Failure to transition to flight mode
  - Helicopter did “spin up”, but failed to fly
  - NASA said this was from a “watchdog timer” expiration
    - Unexpected delay transitioning between operating modes
- Flight 6 – Navigation image timestamp issue
  - Helicopter flight was very wobbly and unstable
    - Helicopter rocked back and forth during flight
  - A frame was dropped during flight 6, causing timestamps to be off for subsequent frames
    - Caused problems for navigation and flight stability
- Issues have received software updates (on Mars!)
Ingenuity flights on Mars
Helicopter operation on Mars

- Continuing mission for Ingenuity ("Operations demonstration")
  - Will continue that mission until August, 2021
  - Helicopter survived Martian change of seasons
  - Can fly in "summer" atmospheric conditions
    - Uses higher speed for propellers, due to thinner air
- Will continue doing stuff indefinitely
  - Is acting as a scout for items of interest for Rover
- January 2022 status
  - Flight 19 delayed due to dust storm
  - First time a flight was delayed due to bad weather on another planet
Sources for Mars helicopter

- Talk by Tim Canham at ELC 2021
  - Slides: https://elinux.org/images/5/5a/1._TIMOTHY_CANHAM.pdf
  - Video: https://youtu.be/0_GfMcBmbCg

- https://mars.nasa.gov/technology/helicopter/
Outline

Linux Kernel Technology Areas Conferences Industry News Resources
Resources

- LWN.net – https://lwn.net
  - If you are not subscribed, please do so
  - Some content is delayed by 2 weeks for non-subscribers (some links in this presentation)
- Phoronix - https://www.phoronix.com/
- eLinux wiki – elinux.org
  - Especially: https://elinux.org/ELC_2021_Presentations
- Google
Thanks!