

# Fixing the year 2038 problem

**MiniDebConf Cambridge 2023**

Wookey

[wookey@wookware.org](mailto:wookey@wookware.org)

# The problem

- Time (time\_t) on unix 32-bit systems is 32-bit signed int seconds since 1<sup>st</sup> Jan 1970. Rolls over at 2147483647 to negative number.
- Happens at 2038-01-19 03:14:07 UTC. Next second is 1901-12-13 20:45:52 UTC
- This is only a bit more than 14 years away
- There are other issues: Y2106 (unsigned 32-bit Epoch-based times), Y2107 (FAT/vfat), Y2036 (unsigned 32-bit 1900-based RFC 868 times (NTP)) or Y9999 (four-digit years)

# This session

- I am not much of an 'expert'.
- Explain the problem
- Last update was FOSDEM
- Discussion occurred: we now have a plan

# How big a problem?

- Real computing mostly 64 bit already
- Lots of cheap computing still 32 bit
- Cars, TVs, controllers (buildings, plant, IOT), cheap phones
- A lot of stuff still running in 2038 is already installed
- There will still be new things. We should fix them. Soon.
- A lot more OpenEmbedded/Android than binary distros.

# Who cares?

- riscv32, x32, loong32, arc already 64-bit time
- RHEL already dropped 32-bit arches
- armv5, i386, mips32 and ppc32 already close to obsolete
- Source distros 'just' rebuild:  
openwrt/buildroot/yocto/gentoo/adelie
- Leaving debian armv7 ('armhf'), Rasbian. (Anyone else?)

# Initial Work

- Arnd Bergman and Deepa Dinamani – Kernel, 2017.  
<https://lwn.net/Articles/717076/>
- Perl fixed 5.12 (works on 32-bit system support) (2010)
- Musl is fixed in 1.2 (2020): <http://musl.libc.org/time64.html>  
(time\_t always 64bit, but things can still run using old ABI)
- Glibc fixed in 3.34 (2021)
- Lots of other software fixed – how much is still broken?

# Distro work

- Adelle (musl):  
<https://web.archive.org/web/20220301175235/https://wiki.adelielinux.org/wiki/Project:Time64>
- Gentoo: [https://wiki.gentoo.org/wiki/Project:Toolchain/time64\\_migration](https://wiki.gentoo.org/wiki/Project:Toolchain/time64_migration)
- OpenSuse: [https://www.reddit.com/r/linux/comments/xjtf3q/in\\_the\\_year\\_2038/](https://www.reddit.com/r/linux/comments/xjtf3q/in_the_year_2038/)
- Ubuntu: Library analysis [https://people.canonical.com/~vorlon/armhf-time\\_t/](https://people.canonical.com/~vorlon/armhf-time_t/)
- OpenEmbedded: Assorted patches
- Debian: Rebuild attempt in 2020 (too much broken)  
Rebootstrap base (2022), ABI analysis
- Ubuntu: (2022-2023) more thorough ABI analysis.  
Foundations team (Steve Langasek) driving debian transition

# Glibc

- Glibc 3.34 – supports old and new.
- Does not force 64bit so other stuff just rebuilds
- File format changes: utmp, wtmp, btmp have a time\_t in
- LFS is tied to 64bit time\_t
  - Sets FILE\_OFFSET\_BITS=64 if TIME\_BITS=64



# LFS (Large Filesystem Support) is involved

- Glibc 3.34 enforces `_FILE_OFFSET_BITS=64` if `_TIME_BITS=64`
- `_LARGEFILE_SOURCE` enables both 32 and 64bit ABI
- `_FILE_OFFSET_BITS=64` enables 64bit ABI only.
- LFS is similar transition to 64bit `time_t` in this regard

# Gnulib and autoconf

- gnulib released between 2022-07-02 and 2022-12-25 automatically enabled time64 support if the system supports it. (Now reverted)
  - set `gl_cv_type_time_t_bits_macro=no` to stop it
- Autoconf 2.72 (Nov 2022) release tried to tie LFS and 64bit-time. (Reverted).

# ABIs and files

- ABI changes if `time_t` used in struct
- File and disk formats contain 32-bit times
- New ABI is just like any other ABI bump, but HUGE
- Supporting/transitioning old file formats important for apps
- File formats mostly fixed for i386→amd64 transition

# Fundamental question

- **Update existing architecture?**
  - Most efforts so far
  - Easy for source distros
  - Much harder for binary distros
  - Changes the ABI
- **New ABI → new triplet?**
  - More 'correct'?
  - Easier for binary distros? (Some stuff will break with new names)
  - Unhelpful for users who want to just upgrade, not re-install

# Debian migration

- Just rebuilding against newer glibc doesn't cause transition something has to set `_TIME_BITS=64` and `_FILE_OFFSET_BITS=64`
- Dpkg or glibc could set them.
- Big transition like `libc5→libc6` within existing arch  
Process affects all arches.
- Minor-arch ABI transition example: long double changed from 64-128bits on alpha, powerpc, sparc, s390 (2007)

# All 32bit arches?

- Just armhf/all 32-bit arches?
- What about i386?
- Rough consensus that i386 is most used for compatibility
- Could make a new 'i686' 32-bit x86 if anyone cared enough
- So all 32-bit arches except i386 will transition (x32 is already 64-bit time)

# How big a problem?

- 6429 packages of Debian's 35960 (18%) have `time_t` in
- How many public ABIs change?
- You have to compile the headers to find out
- Abi-compliance-checker via 163,000 line shell-script
  - [https://salsa.debian.org/vorlon/armhf-time\\_t](https://salsa.debian.org/vorlon/armhf-time_t)
- Initially ~1650 did not compile. Now down to ~900
  - <https://wiki.debian.org/ArmhfTimeTTodo>
  - Ordered by reverse dependencies. 450 have zero revdeps.

# How many libraries?

- **-dev packages: 10323**
  - 4963 are rust-\*, golang-\* and libghc-\* which can be ignored, leaving 5237 to consider
    - Time\_t changes ABI: 495 (9%)
    - LFS changes ABI: 101 (2%)
    - Didn't change: 2586 (50%)
    - 1388 failed the checker (26%) (down to 900 now)
- **Reverse dependencies of changed ABIs: 5063-5975**
- **Perlapi-5.x.x virtual package adds 600**
- **Need to assume the 'failed' ones changed ABI, and add their revdeps (up to 1500).**



# Transition Plan

- **abi=time64 and abi=ifs in dpkg 1.22 onwards** (used to be future=)
  - Enabled by default on flag-day
- **~500 libs NMUed with t64 suffix** (replaces any existing suffix)  
so **libfoo → libfoot64**, **libfoo-dev stays libfoo-dev**
  - + up to 900 unknown – work here still very helpful
- **Once built, rebuild reverse deps with binNMU on all arches**
  - ~6200 (including 700 perl).
- **Next SONAME bump will remove 't64' suffix**
- **Coordination with FTPmasters needed**
- **Current plan is do this Jan 2024**
- **Plan: <https://lists.debian.org/debian-devel/2023/05/threads.html>**

# What else breaks?

- NFSv3 (some clients signed so OK till 2108)?
- INN CNFS is OK, but has time\_t in older timecaf disk format
- 32-bit Wine OK as i386 not changing
- Perl 'XS modules' C ABI (600 rdeps on perlapi-5.x.x)
- datefudge/faketime need fixing (reproducible builds) (#1028587)

# What else breaks?

- apr needs SONAME bump for `-DFILE_OFFSET_BITS=64` (#1031034)
- Fixed: `gdnsd`, `who`, `tar`
- PHP ints same size as `DEB_HOST_ARCH_BITS`, so old PHP not yet using `DateTime` API breaks (how much is that?)
- This work fixes ABI, not 2038-wrap issues which may still occur.
- Whatever else we didn't notice yet.

# ModernC update is entangled

- Wrinkle that looking up functions in non-standard ways (e.g. FFI or implicit declarations) could get wrong-size definition
  - <https://fedoraproject.org/wiki/Changes/PortingToModernC>
- Can be avoided by `-Wno-implicit-function-declaration`
- Should we set `-Werror=implicit-function-declaration` in `dpkg-buildflags` for everything, or just if `abi=timet64` is set?

# Questions/Issues

- Test coverage unknown
- Not everything respects dpkg-buildflags/environment
- Are FTPmasters ready for the deluge?
- Are there any time\_t- and lfs- using libs which depend on lfs- using libs which are still on 32-bit?
- How many implicit function declaration issues?
  - Gentoo has 568 in bug tracker
- Does dpkg emit -Werror=implicit-function-declaration already?

## Further info/Discussion

- <https://wiki.debian.org/ReleaseGoals/64bit-time>
- <https://wiki.debian.org/ArmhfTimeTToDo>
- [time64 bug list](#)
  
- [distributions@lists.linux.dev](mailto:distributions@lists.linux.dev)
- <https://subspace.kernel.org/lists.linux.dev.html>

Please join in if you are interested/worried/understand specific issues.

# Modern C porting also implicated

- Implicit function defines are going away
- Errors from gcc14 and clang 16 (next year)
- [https://wiki.gentoo.org/wiki/Modern\\_C\\_porting](https://wiki.gentoo.org/wiki/Modern_C_porting)
- Implicit functions break glibc foo→foo64 macros
- Some configure tests depend on implicit functions