

# Freedesktop Planet - Latest News

- [GNU Guix: A Planet for Guix](#) (2025/12/05 10:48)

I am pleased to announce the availability of Planet Guix, an Atom and RSS aggregator covering all things Guix. You can browse posts on the website or use your favourite feed reader to subscribe to the aggregate feed. Planet Guix already has subscriptions to 19 blogs from around the community; if you write about Guix (no matter how infrequently) and would like your blog to be included, or if you would like to suggest another blog I missed, please create a pull request against the repository in Codeberg — you'll see that the subscriptions are simply configured as association lists in `planet/config.scm`. Background Back in September, Sébastien Gendre asked on the `help-guix` mailing list if there were any plans to create a Planet website for Guix. The discussion drifted into how this might be implemented in Guile, and I thought it sounded like an interesting project for the dark autumn evenings. The original Planet aggregator was written in Python and many Planet websites are still using its successor, Venus. The Venus code base has not seen much activity in the last decade and still uses Python 2, which was sunset in 2020. This was all the incentive I needed to implement a new Planet aggregator and static site generator in Guile. Implementation We already know from the likes of Haunt that Guile has all the tools needed to generate a static web site. Both Atom and RSS are XML formats, and Guile also has great support for working with XML. The Guile Planet implementation uses the following built-in modules: `(web client)` to fetch the feeds. `(sxml simple)` for reading the Atom/RSS feeds and writing the aggregate Atom feed. `(sxml xpath)` for searching the feeds to extract the data of interest. `(sxml transform)` for sanitizing HTML in the entry summaries. Many feeds include HTML content in the entry summary, which we need to parse. This is where `htmlprag` from `guile-lib` comes in. I used this both to parse HTML embedded in feeds and to generate the static content from an SXML data structure. With these libraries to hand the code for the planet aggregator almost wrote itself! I was trying to keep dependencies to a minimum, but `guile-filesystem` is too useful to do without and, later in the development process, I pulled in `guile-srfi-235` which provides some useful combinators. At the moment I'm only using `apply-chain` to build a function for post-processing one of the feeds, but why re-invent the wheel? Deployment I initially deployed the Planet to a test site running on one of my servers, but the idea was received enthusiastically by the Guix maintainers and I was happy that they wanted to host it on their infrastructure. Of course they are using Guix to manage their virtual machines in Hetzner cloud! While they could have picked up the Planet code and run with it, instead they pointed me at the server configuration and invited me to make a pull request against `hydra/guix-hetzner-2.scm`. They suggested I base the configuration on their existing `static-web-site-configuration` so I started reading the code which proved very educational (I admit that I had to sleep on it for a week before coming up with a plan!) The `static-web-site-configuration` did almost everything needed to build the Planet aggregator, only the build step runs like a Guix package build in an isolated environment with no network - so we cannot fetch the feeds in this build step. Luckily, I had already implemented functionality in the Planet code base to build the static site from feeds cached on disk. So it was simply a case of adding support for a pre-build script to the `static-web-site-configuration` and using this step to download the feeds. The pull request was merged after some short discussion, and a few days later the site was live in its new home. Community This was my second time contributing to the Guix project and I'm pleased to report that it was a smooth experience both times. When it came to the deployment, I was glad that I was encouraged to add the service configuration myself instead of being spoon-fed: working with computers, you learn best by doing. I'd like to give a shout-out to [@civodul](#), [@cbaines](#), and [@apteryx](#) for

their help with the deployment, and to the several people who sent merge requests to add their blogs before I even got around to writing this announcement. I think the Planet site is already a great place to discover people writing about Guix, and I hope it grows and becomes an asset to the community. Happy reading!

- [FSF Blogs: November GNU Spotlight with Amin Bandali featuring twelve new GNU releases: Coreutils, Gnuastro, and more!](#) (2025/12/02 14:32)

Twelve new GNU releases in the last month (as of November 30, 2025):

- [FSF Blogs: We support your freedom — support the Licensing and Compliance Lab](#) (2025/12/01 22:52)
- [GNU Guile: GNU Guile 3.0.11 released](#) (2025/12/01 14:00)

We are pleased to announce the release of GNU Guile 3.0.11! This release is mainly a bug-fix release, though it does include a number of new features, including support for SRFI 197: Pipeline Operators, support for SRFI 207: String-notated bytevectors (bytestrings), and JIT (just-in-time) compilation for the RISC-V architecture. It also overhauls SRFI-64 (testing) and includes many other improvements. For full details, see the release notes and check out the download page. Happy Guile hacking!

- [Jose E. Marchesi: Algol 68 Front-End merged in GCC](#) (2025/11/30 00:00)

I am very happy to announce that, after almost a year in the works, today the Algol 68 front-end has been merged in GCC proper in its development trunk branch. This means that we are no longer off-tree, and that GCC 16 will be featuring a full-fledged and modern Algol 68 compiler once it gets released. An Algol 68 module This doesn't mean the work is done. The modules system, which is already functional, needs to be completed and polished, parallel clauses are still to be implemented, and the quality of the generated code should be generally improved. We also want to replace the Boehm GC with a tightly integrated customized, exact one, and to continue modernizing and expanding the language, always carefully and respectfully, via GNU extensions: exception handling, FFI, etc. At this point I would like to thank Marcel van der Veer, Pietro Monteiro, Mohammad-Reza Nabipoor, Thomas Schwinge, Sam James, Matthias Klose, Iain Buclaw, Andrew Pinski, Segher Boessenkool, Iain Sandoe, the GCC global reviewers and the overall GCC community. Without their help, and Marcel's nifty Algol 68 parser, this front-end would simply not exist. And now, the real fun starts... ;) Happy algoling!

- [Simon Josefsson: Container Images for Debian with Guix](#) (2025/11/28 16:32)

The debian-with-guix-container project build and publish container images of Debian GNU/Linux stable with GNU Guix installed. The images are like normal Debian stable containers but have the guix tool and a reasonable fresh guix pull. Supported architectures include amd64 and arm64. The multi-arch container is called: `registry.gitlab.com/debdistutils/guix/debian-with-guix-container:stable` It may also be accessed via `debian-with-guix` at Docker Hub as: `docker.io/jas4711/debian-with-guix:stable` The container images may be used like this: `$ podman run --privileged -it --hostname guix --rm registry.gitlab.com/debdistutils/guix/debian-with-guix-container:stable root@guix:/# hello bash: hello: command not found root@guix:/# guix describe guix c9eb69d repository URL: https://gitlab.com/debdistutils/guix/mirror.git branch: master commit: c9eb69ddb05e77300b59f49f4bb5aa50cae0892 root@guix:/# LC_ALL=C.UTF-8 /root/.config/guix/current/bin/guix-daemon --build-users-group=guixbuild & [1] 21 root@guix:/# GUIX_PROFILE=/root/.config/guix/current; . "$GUIX_PROFILE/etc/profile" root@guix:/# guix describe Generation 2 Nov 28 2025 10:14:11 (current) guix c9eb69d repository URL: https://gitlab.com/debdistutils/guix/mirror.git branch: master commit: c9eb69ddb05e77300b59f49f4bb5aa50cae0892 root@guix:/# guix install --verbosity=0 hello accepted connection from pid 55, user root The following package will be installed: hello 2.12.2 hint: Consider setting the necessary environment variables by running: GUIX_PROFILE="/root/.guix-profile" . "$GUIX_PROFILE/etc/profile" Alternately, see `guix package --search-paths -p "/root/.guix-profile".`

root@guix:/# GUIX\_PROFILE="/root/.guix-profile" root@guix:/# . "\$GUIX\_PROFILE/etc/profile" root@guix:/# hello Hello, world! root@guix:/# Below is an example GitLab pipeline job that demonstrate how to run guix install to install additional dependencies, and then download and build a package that pick up the installed package from the system. test-wget-configure-make-libksba-amd64: image: registry.gitlab.com/debdistutils/guix/debian-with-guix-container:stable before\_script: - env LC\_ALL=C.UTF-8 /root/.config/guix/current/bin/guix-daemon --build-users-group=guixbuild \$GUIX\_DAEMON\_ARG & - GUIX\_PROFILE=/root/.config/guix/current; . "\$GUIX\_PROFILE/etc/profile" - guix describe - guix install libgpg-error - GUIX\_PROFILE="/root/.guix-profile"; . "\$GUIX\_PROFILE/etc/profile" - apt-get install --update -y --no-install-recommends build-essential wget ca-certificates bzip2 script: - wget https://www.gnupg.org/ftp/gcrypt/libksba/libksba-1.6.7.tar.bz2 - tar xfa libksba-1.6.7.tar.bz2 - cd libksba-1.6.7 - ./configure - make V=1 - make check VERBOSE=t V=1 The images were initially created for use in GitLab CI/CD Pipelines but should work for any use. The images are built in a GitLab CI/CD pipeline, see .gitlab-ci.yml. The containers are derived from official Debian stable images with Guix installed and a successful run of guix pull, built using buildah invoked from build.sh using image/Containerfile that runs image/setup.sh. The pipeline also push images to the GitLab container registry, and then also to Docker Hub. Guix binaries are downloaded from the Guix binary tarballs project because of upstream download site availability and bandwidth concerns. Enjoy these images! Hopefully they can help you overcome the loss of Guix in Debian which made it a mere apt-get install guix away before. There are several things that may be improved further. An alternative to using podman --privileged is to use --security-opt seccomp=unconfined --cap-add=CAP\_SYS\_ADMIN,CAP\_NET\_ADMIN which may be slightly more fine-grained. For ppc64el support I ran into an error message that I wasn't able to resolve: guix pull: error: while setting up the build environment: cannot set host name: Operation not permitted For riscv64, I can't even find a Guix riscv64 binary tarball for download, is there one anywhere? For arm64 containers, it seems that you need to start guix-daemon with --disable-chroot to get something to work, at least on GitLab.com's shared runners, otherwise you will get this error message: guix install: error: clone: Invalid argument Building the images themselves also require disabling some security functionality, and I was not able to build images with buildah without providing --cap-add=CAP\_SYS\_ADMIN,CAP\_NET\_ADMIN otherwise there were errors like this: guix pull: error: cloning builder process: Operation not permitted guix pull: error: clone: Operation not permitted guix pull: error: while setting up the build environment: cannot set loopback interface flags: Operation not permitted Finally on amd64 it seems --security-opt seccomp=unconfined is necessary, otherwise there is an error message like this, even if you use --disable-chroot: guix pull: error: while setting up the child process: in phase setPersonality: cannot set personality: Function not implemented This particular error is discussed upstream, but I think generally that these error suggest that guix-daemon could use more optional use of features: if some particular feature is not available, gracefully fall back to another mode of operation, instead of exiting with an error. Of course, it should never fall back to an insecure mode of operation, unless the user requests that. Happy Hacking!

- [gnuastro @ Savannah: Gnuastro 0.24 released](#) (2025/11/27 01:35)

The 24th release of GNU Astronomy Utilities (Gnuastro) is now available. See the full announcement for all the new features in this release and the many bugs that have been found and fixed: [https://lists.gnu.org ... -11/msg00001.html](https://lists.gnu.org/archive/html/astropack-devel/2025/11/msg00001.html)

- [FSF Events: Free Software Directory meeting on IRC: Friday, December 5, starting at 12:00 EST \(17:00 UTC\)](#) (2025/11/26 18:15)

Join the FSF and friends on Friday, December 5 from 12:00 to 15:00 EST (17:00 to 20:00 UTC) to help improve the Free Software Directory.

- [FSF Events: Free Software Directory meeting on IRC: Friday, November 28, starting at 12:00 EST \(17:00 UTC\)](#) (2025/11/26 18:13)

Join the FSF and friends on Friday, November 28 from 12:00 to 15:00 EST (17:00 to 20:00 UTC) to help improve the Free Software Directory.

- [FSF Blogs: Hundreds of free software supporters tuned in for FSF40 hackathon](#) (2025/11/25 21:05)
- [Gary Benson: Slow boot?](#) (2025/11/24 08:41)

Does your Linux box take forever to boot? The command you're looking for is systemd-analyze blame

- [parallel @ Savannah: GNU Parallel 20251122 \('Mamdani'\) released \[stable\]](#) (2025/11/24 05:27)

GNU Parallel 20251122 ('Mamdani') has been released. It is available for download at: [lrybry://@GnuParallel](https://lrybry.org/@GnuParallel):4 Quote of the month: ainda não inventaram palavras capazes de expressar minha gratidão aos desenvolvedores do GNU Parallel -- @nueidris.kawaii.social New in this release: No new features. Bug fixes. GNU Parallel - For people who live life in the parallel lane. If you like GNU Parallel record a video testimonial: Say who you are, what you use GNU Parallel for, how it helps you, and what you like most about it. Include a command that uses GNU Parallel if you feel like it. About GNU Parallel GNU Parallel is a shell tool for executing jobs in parallel using one or more computers. A job can be a single command or a small script that has to be run for each of the lines in the input. The typical input is a list of files, a list of hosts, a list of users, a list of URLs, or a list of tables. A job can also be a command that reads from a pipe. GNU Parallel can then split the input and pipe it into commands in parallel. If you use xargs and tee today you will find GNU Parallel very easy to use as GNU Parallel is written to have the same options as xargs. If you write loops in shell, you will find GNU Parallel may be able to replace most of the loops and make them run faster by running several jobs in parallel. GNU Parallel can even replace nested loops. GNU Parallel makes sure output from the commands is the same output as you would get had you run the commands sequentially. This makes it possible to use output from GNU Parallel as input for other programs. For example you can run this to convert all jpeg files into png and gif files and have a progress bar: `parallel --bar convert {1} {1.}.{2} ::: *.jpg ::: png gif` Or you can generate big, medium, and small thumbnails of all jpeg files in sub dirs: `find . -name '*.jpg' | parallel convert -geometry {2} {1} {1//}/thumb{2}_{1//} :::: - :::: 50 100 200` You can find more about GNU Parallel at: <http://www.gnu.org/s/parallel/> You can install GNU Parallel in just 10 seconds with: `$ (wget -O - pi.dk/3 || lynx -source pi.dk/3 || curl pi.dk/3/ || \ fetch -o - http://pi.dk/3 ) > install.sh $ sha1sum install.sh | grep c555f616391c6f7c28bf938044f4ec50 12345678 c555f616 391c6f7c 28bf9380 44f4ec50 $ md5sum install.sh | grep 707275363428aa9e9a136b9a7296dfe4 70727536 3428aa9e 9a136b9a 7296dfe4 $ sha512sum install.sh | grep b24bfe249695e0236f6bc7de85828fe1f08f4259 83320d89 f56698ec 77454856 895edc3e aa16feab 2757966e 5092ef2d 661b8b45 b24bfe24 9695e023 6f6bc7de 85828fe1 f08f4259 6ce5480a 5e1571b2 8b722f21 $ bash install.sh` Watch the intro video on <http://www.youtube.com/watch?v=L284C9FF2488BC6D1> Walk through the tutorial (man parallel\_tutorial). Your command line will love you for it. When using programs that use GNU Parallel to process data for publication please cite: O. Tange (2018): GNU Parallel 2018, March 2018, <https://doi.org/10.1101/281146>. If you like GNU Parallel: Give a demo at your local user group/team/colleagues Post the intro videos on Reddit/Diaspora\*/forums/blogs/Identi.ca/Google+/Twitter/Facebook/Linkedin/ mailing lists Get the merchandise <https://gnuparall.org/merch/> Request or write a review for your favourite blog or magazine Request or build a package for your favourite distribution (if it is not already there) Invite me for your next conference If you use programs that use GNU Parallel for research: Please cite GNU Parallel in you publications (use --citation) If GNU Parallel saves you money: (Have your company) donate to FSF <https://my.fsf.org/donate/> About GNU SQL GNU sql aims to give a simple, unified interface for accessing databases through all the different databases' command line clients. So far the focus has been on giving a common way to specify login information (protocol, username, password, hostname, and port number), size (database and table size), and running queries. The database is addressed using a DBURL. If commands are left out you will get that database's interactive shell. When using GNU SQL for a publication please cite: O. Tange (2011): GNU SQL - A Command Line Tool for Accessing Different Databases Using DBURLs, ;login: The USENIX

Magazine, April 2011:29-32. About GNU Niceload GNU niceload slows down a program when the computer load average (or other system activity) is above a certain limit. When the limit is reached the program will be suspended for some time. If the limit is a soft limit the program will be allowed to run for short amounts of time before being suspended again. If the limit is a hard limit the program will only be allowed to run when the system is below the limit.

- [Jose E. Marchesi: Version 6 of the Algol 68 GCC Front-End posted](#) (2025/11/22 03:00)

Today I submitted the version 6 of the patch series for the Algol 68 GCC Front-End:

<https://gcc.gnu.org/pipermail/gcc-patches/2025-November/701589.html> Since last submission we have added a modules system based on the Modules and Separate Compilation Facility designed by Charles Lindsey and Hendrik Boom and released by the IFIP Working Group 2.1 Standing Subcommittee on ALGOL 68 Support. To our knowledge, this is the first time the modules facility ever gets implemented. This is the deal: Jose E. Marchesi (50): a68: top-level misc files a68: build system a68: build system (regenerated files) a68: documentation a68: command-line options a68: DWARF language codes a68: darwin specific support a68: powerpc specific support a68: gcc/algol68 misc files a68: ga68 compiler driver a68: a681 compiler proper a68: unicode support routines a68: front-end diagnostics a68: modules exports a68: modules imports a68: parser: entry point a68: parser: AST nodes attributes/types a68: parser: scanner a68: parser: keyword tables management a68: parser: top-down parser a68: parser: parenthesis checker a68: parser: bottom-up parser a68: parser: syntax check for declarers a68: parser: standard prelude definitions a68: parser: parsing of modes a68: parser: symbol table management a68: parser: static scope checker a68: parser: debug facilities a68: parser: extraction of tags from phrases a68: parser: dynamic stack usage in serial clauses a68: parser: pragmas infrastructure a68: low: lowering entry point and misc handlers a68: low: plain values a68: low: stowed values a68: low: standard prelude a68: low: clauses and declarations a68: low: runtime a68: low: builtins a68: low: ranges a68: low: units and coercions a68: low: modes a68: libga68: sources, spec and misc files a68: libga68: build system a68: libga68: build system (generated files) a68: testsuite: infrastructure a68: testsuite: execution tests 1/2 a68: testsuite: execution tests 2/2 a68: testsuite: compilation tests a68: testsuite: revised MC Algol 68 test set a68: testsuite: mcgt tests

- [FSF Blogs: The FSF40 hackathon is this weekend. Here's what you need to know](#) (2025/11/19 20:30)

- [FSF Blogs: FSD meeting recap 2025-11-14](#) (2025/11/17 21:33)

Check out the important work our volunteers accomplished at Friday's Free Software Directory (FSD) IRC meeting.

- [remotecontrol @ Savannah: Google is collecting troves of data from downgraded Nest thermostats](#) (2025/11/17 15:06)

[https://www.theve ... d-data-collection](https://www.theve...d-data-collection)

- [GNUnet News: GNUnet 0.26.1](#) (2025/11/14 23:00)

GNUnet 0.26.1 This is a bugfix release for gnunet 0.26.0. It fixes some regressions and minor bugs. Links Source:

<https://ftpmirror.gnu.org/gnunet/gnunet-0.26.1.tar.gz> ( <https://ftpmirror.gnu.org/gnunet/gnunet-0.26.1.tar.gz.sig> ) Detailed list of changes:

<https://git.gnunet.org/gnunet.git/log/?h=v0.26.1> NEWS: <https://git.gnunet.org/gnunet.git/tree/NEWS?h=v0.26.1> The GPG key used to sign is:

3D11063C10F98D14BD24D1470B0998EF86F59B6A Note that due to mirror synchronization, not all links may be functional early after the release. For direct access try <https://ftp.gnu.org/gnu/gnunet/>

- [Simon Josefsson: Introducing the Debian Libre Live Images](#) (2025/11/12 23:16)

The Debian Libre Live Images allows you to run and install Debian GNU/Linux without non-free software. The general goal is to provide a way to use Debian without reliance on non-free software, to the extent possible within the Debian project. One challenge are the official Debian live and



installer images. Since the 2022 decision on non-free firmware, the official images for bookworm and trixie contains non-free software. The Debian Libre Live Images project provides Live ISO images for Intel/AMD-compatible 64-bit x86 CPUs (amd64) built without any non-free software, suitable for running and installing Debian. The images are similar to the Debian Live Images distributed as Debian live images. One advantage of Debian Libre Live Images is that you do not need to agree to the distribution terms and usage license agreements of the non-free blobs included in the official Debian images. The rights to your own hardware won't be crippled by the legal restrictions that follows from relying on those non-free blobs. The usage of your own machine is no longer limited to what the non-free firmware license agreements allows you to do. This improve your software supply-chain situation, since you no longer need to consider their implication on your computing environment for your liberty, privacy or security. Inclusion of non-free firmware is a vehicle for xz-style attacks. For more information about the advantages of free software, see the FSF's page on What is Free Software?. Enough talking, show me the code! Err, binaries! Download images: `wget https://gitlab.com/api/v4/projects/74667529/packages/generic/debian-libre-live/main/live-image-amd64.hybrid.iso` `wget https://gitlab.com/api/v4/projects/74667529/packages/generic/debian-libre-live/main/live-image-amd64.hybrid.iso.SHA256SUMS` `sha256sum -c live-image-amd64.hybrid.iso.SHA256SUMS` Run in a virtual machine: `kvm -cdrom live-image-amd64.hybrid.iso -m 8G` Burn to an USB drive for installation on real hardware: `sudo dd if=live-images-amd64.hybrid.iso of=/dev/sdX # use sdX for USB drive` Images are built using live-build from the Debian Live Team. Inspiration has been taken from Reproducible Live Images and Kali Live. The images are built by GitLab CI/CD shared runners. The pipeline `.gitlab-ci.yml` container job creates a container with live-build installed, defined in `container/Containerfile`. The build job then invokes `run.sh` that includes a run to `lb build`, and then upload the image to the package registry. This is a first initial public release, calibrate your expectations! The primary audience are people already familiar with Debian. There are known issues. I have performed successful installations on a couple of different machines including laptops like Lenovo X201, Framework AMD Laptop 13" etc. Are you able to install Debian without any non-free software on some hardware using these images? Happy Hacking!

- [GNUnet News: GNUnet 0.26.0](#) (2025/11/12 23:00)

GNUnet 0.26.0 released We are pleased to announce the release of GNUnet 0.26.0. GNUnet is an alternative network stack for building secure, decentralized and privacy-preserving distributed applications. Our goal is to replace the old insecure Internet protocol stack. Starting from an application for secure publication of files, it has grown to include all kinds of basic protocol components and applications towards the creation of a GNU internet. This is a new major release. Major versions may break protocol compatibility with the 0.25.X versions. Please be aware that Git master is thus henceforth (and has been for a while) INCOMPATIBLE with the 0.25.X GNUnet network, and interactions between old and new peers will result in issues. In terms of usability, users should be aware that there are still a number of known open issues in particular with respect to ease of use, but also some critical privacy issues especially for mobile users. Also, the nascent network is tiny and thus unlikely to provide good anonymity or extensive amounts of interesting information. As a result, the 0.26.0 release is still only suitable for early adopters with some reasonable pain tolerance . If it were not for compatibility-breaking changes in the crypto API of `libgnunetutil` this would only be a maintenance release. The changes hopefully protect users of the library from misuse of GNUnet's cryptographic key objects in ways that may be detrimental to security. Since this change breaks backwards compatibility, this is a new major release. Download links `gnunet-0.26.0.tar.gz` ( signature ) `gnunet-fuse-0.26.0.tar.gz` ( signature ) The GPG key used to sign is: `3D11063C10F98D14BD24D1470B0998EF86F59B6A` Note that due to mirror synchronization, not all links might be functional early after the release. For direct access try `http://ftp.gnu.org/gnu/gnunet/` Changes A detailed list of changes can be found in the git log , the NEWS . Known Issues There are known major issues with the TRANSPORT subsystem.

There may be some regressions in the new CORE subsystem. There are known moderate implementation limitations in CADET that negatively impact performance. There are known moderate design issues in FS that also impact usability and performance. There are minor implementation limitations in SET that create unnecessary attack surface for availability. The RPS subsystem remains experimental. In addition to this list, you may also want to consult our bug tracker at [bugs.gnunet.org](https://bugs.gnunet.org) which lists about 190 more specific issues. Thanks This release was the work of many people. The following people contributed code and were thus easily identified: Christian Grothoff, Florian Dold, TheJackiMonster, ch3, and Martin Schanzenbach.

- [health @ Savannah: GNU Health HIS server 5.0.3 patchset bundle released](#) (2025/11/12 17:52)

Dear community I am happy to announce that the GNU Health Hospital Information System 5.0.3 patchset has been released! In addition to the patches, you will notice a tarball significantly smaller than previous bundles, mainly from not excluding large translation files that need work. You can of course always download the most current translation of any GNU Health package from our weblate instance at Codeberg. ([https://translate ... ojects/gnuhealth/](https://translate.ojects.gnuhealth/)) If you use the vanilla / standard installation, you can update the server and the dependencies from the gnuhealth control center (<https://docs.gnuh ... ontrolcenter.html>) Backup As usual, before you upgrade your instance, make sure you have made a backup of your DB instance and "attach" resource ! Happy hacking ♥ Changelog for 5.0.3 2025-11-11 \* [SKIP CI] Backport po files from devel branch. (c7f8f3a3c) (Feng Shu) 2025-11-11 \* Add tryton/health\_icd9procs/locale/el.po (5acbb83f2) (Feng Shu) 2025-11-10 \* [SKIP CI] Backport po files from devel branch. (57aeec18f) (Feng Shu) 2025-11-07 \* [SKIP CI] Backport po files from devel branch. (489f1b225) (Feng Shu) 2025-11-05 \* Fix bug core:Dx command should filter by arguments #153 (dc410b014) (Luis Falcon) 2025-11-05 \* [SKIP CI] Backport po files from devel branch. (40a327c3b) (Feng Shu) 2025-11-05 \* Update tryton/health/locale/health.pot (a41f324ed) (Feng Shu) 2025-11-05 \* [SKIP CI] Backport po files from devel branch. (583837f1b) (Feng Shu) 2025-11-05 \* fix bug core: healthprof attribute duplicated in gnuhealth.patient.disease model (2c6976f6b) (Luis Falcon) 2025-11-04 \* Readd sv.po file, for Umeaman will become a team leader for Swedish. [do not merge] (31ef5422c) (Feng Shu) 2025-11-04 \* [SKIP CI] Backport po files from devel branch. (bc914fc3d) (Feng Shu) 2025-11-02 \* Delete health module po files which translate progress < 30%, [do not merge] (48e76223a) (Feng Shu) 2025-10-29 \* [SKIP CI] Backport po files from devel branch. (c2497b4de) (Feng Shu) 2025-10-28 \* translation: Missing translation terms. #156. Fix spanish string for time in health\_surgery (c76d20d36) (Luis Falcon) 2025-10-28 \* Issue #155. Clean po files, backport from devel branch [do not merge] (59b8f3eb5) (Feng Shu) 2025-10-28 \* Issue #155. genetics\_uniprot: Remove empty translation po files (01806d006) (Luis Falcon) 2025-10-28 \* [SKIP CI] Backport po files from devel branch. (e719e0f9c) (Feng Shu) 2025-10-28 \* Update tryton/health/locale/health.pot [do not merge] (3b9ec6200) (Feng Shu) 2025-10-28 \* Merge remote-tracking branch 'origin/po-backport' into patchset/5.0.3 (b6847391a) (Feng Shu) 2025-10-27 \* [SKIP CI] Backport po files from devel branch. (32c1903b2) (po-backport) 2025-10-25 \* Update package version to 5.0.3 (bbcc6c9ed) (Luis Falcon) 2025-10-25 \* Fix bug #150. core: Include children in the family history (46f6293ad) (Luis Falcon) For a more detailed list, please go to our project page at Codeberg: <https://codeberg. ... org/gnuhealth/his>

- [GNU Taler news: GNU Taler 1.2 released](#) (2025/11/11 23:00)

We are happy to announce the release of GNU Taler v1.2.

- [health @ Savannah: GNUHealthCon 2025 - December 20th](#) (2025/11/11 08:51)

The GNU Health Conference 2025 will take place December 20th, and it will be online! Join us and get the latest on Free Software in healthcare and social medicine! #GHCon2025 More info and registration: [https://www.gnuh ... h.org/ghcon/2025/](https://www.gnuh... h.org/ghcon/2025/)

- [unrtf @ Savannah: unrtf 0.21.11](#) (2025/11/11 06:59)

unrtf 0.21.11 is released, fixing recently submitted security issues and a number of older bugs. Until the tar file can be uploaded to the proper location on <ftp.gnu.org>, you can find it on the project home page

- [coreutils @ Savannah: coreutils-9.9 released \[stable\]](#) (2025/11/10 14:16)

This is to announce coreutils-9.9, a stable release. This is primarily a stabilization release, details of which are summarized in the NEWS below. There have been 106 commits by 10 people in the 7 weeks since 9.8. Thanks to everyone who has contributed! The following people contributed changes to this release: Bernhard Voelker (4) Mathieu Bordere (1) Bruno Haible (4) Nicolas Boichat (1) Collin Funk (28) Paul Eggert (9) Grisha Levit (1) Pádraig Brady (57) Hannes Braun (1) Sylvestre Ledru (1) Pádraig [on behalf of the coreutils maintainers]

===== Here is the GNU coreutils home page: <https://gnu.org/s/coreutils/> Here are the compressed sources: <https://ftp.gnu.org/gnu/coreutils/coreutils-9.9.tar.gz> (15MB)

<https://ftp.gnu.org/gnu/coreutils/coreutils-9.9.tar.xz> (6.1MB) Here are the GPG detached signatures:

<https://ftp.gnu.org/gnu/coreutils/coreutils-9.9.tar.gz.sig> <https://ftp.gnu.org/gnu/coreutils/coreutils-9.9.tar.xz.sig> Use a mirror for higher download bandwidth: <https://www.gnu.org/order/ftp.html> Here are the SHA1 and SHA256 checksums: File: coreutils-9.9.tar.gz SHA1 sum:

c66ec935ab7e0ef32c40153fcf67dcf67579171a SHA256 sum: 91a719fc923de686016f2c8d084a8be1f793f34173861273c4668f7c65af94a File: coreutils-9.9.tar.xz SHA1 sum: 456b5c69f3ce8fbdbe926a11652673ecf12bfc44 SHA256 sum:

19bcb6ca867183c57d77155eae946c5eced88183143b45ca51ad7d26c628ca75 Use a .sig file to verify that the corresponding file (without the .sig suffix) is intact. First, be sure to download both the .sig file and the corresponding tarball. Then, run a command like this: `gpg --verify coreutils-9.9.tar.gz.sig` The signature should match the fingerprint of the following key: `pub rsa4096/0xDF6FD971306037D9 2011-09-23 [SC]`

Key fingerprint = 6C37 DC12 121A 5006 BC1D B804 DF6F D971 3060 37D9 uid [ultimate] Pádraig Brady <P@draigBrady.com> uid [ultimate] Pádraig Brady <pixelbeat@gnu.org> If that command fails because you don't have the required public key, or that public key has expired, try the following commands to retrieve or refresh it, and then rerun the 'gpg --verify' command. `gpg --locate-external-key P@draigBrady.com gpg --recv-keys DF6FD971306037D9 wget -q -O-`

`'https://savannah.gnu.org/project/release-gpgkeys.php?group=coreutils&download=1' | gpg --import -` As a last resort to find the key, you can try the official GNU keyring: `wget -q https://ftp.gnu.org/gnu/gnu-keyring.gpg gpg --keyring gnu-keyring.gpg --verify coreutils-9.9.tar.gz.sig` This release is based on the coreutils git repository, available as `git clone https://https.git.savannah.gnu.org/git/coreutils.git` with commit 0ae5bdc7a8311efd3efe43363050710d6ea1c367 tagged as v9.9. For a summary of changes and contributors, see:

<https://gitweb.git.savannah.gnu.org/gitweb/?p=coreutils.git;a=shortlog;h=v9.9> or run this command from a git-cloned coreutils directory: `git shortlog v9.8..v9.9` This release was bootstrapped with the following tools: Autoconf 2.72.97-cf8b9 Automake 1.18.1 Gnulib 2025-11-06

862a81c0e15448adde6a6e7473ec47e3a4bd91a6 Bison 3.8.2 NEWS \* Noteworthy changes in release 9.9 (2025-11-10) [stable] \*\* Bug fixes `'basenc --base58'` would not operate correctly with input > 15561475 bytes. [bug introduced with --base58 in coreutils-9.8] `'cksum --check'` now supports base64 encoded input in untagged format: `-` for all length adjustable algorithms (blake2b, sha2, sha3), `-` if that base64 input starts with a tag like "SHA1" etc. Previously an error was given, about invalid input format. [bug introduced in coreutils-9.2] `'cksum --check -a sha2'` has better support for tagged format. Previously an unneeded but explicit `'-a sha2'` did not match standard tags like SHA256. Also non standard SHA2 tags with a bad length resulted in undefined behavior. [bug introduced in coreutils-9.8] `'cp'` restores performance with



transparently compressed files, which regressed due to the avoidance of copy offload, seen with OpenZFS at least. [bug introduced in coreutils-9.8] `env` on macOS, for now only when built with --disable-nls, will no longer always set a `__CF_USER_TEXT_ENCODING` environment variable. [bug introduced in coreutils-9.8] 'nice' now limits the adjusted niceness value to its supported range on GNU/Hurd. [This bug was present in "the beginning".] 'numfmt' no longer reads out-of-bounds memory with trailing blanks in input. [bug introduced with numfmt in coreutils-8.21] 'numfmt' no longer outputs invalid characters with multi-byte blanks in input. [bug introduced in coreutils-9.5] 'rm -d DIR' no longer fails on Ceph snapshot directories. Although these directories are nonempty, 'rmdir DIR' succeeds on them. [bug introduced in coreutils-8.16] 'sort --compress-program' now diagnoses if it can't write more data to an exited compressor. Previously sort could have exited silently in this case. [bug introduced in coreutils-6.8] 'tail' outputs the correct number of lines again for non-small -n values. Previously it may have output too few lines. [bug introduced in coreutils-9.8] 'unexpand' no longer triggers a heap buffer overflow with --tabs arguments that use the GNU extension /NUM or +NUM formats. [bug introduced in coreutils-8.28] \*\* Changes in behavior 'cp' with default options may again, like with versions before v9.8, miss opportunities to create holes with file systems that support SEEK\_HOLE only trivially. This change is a consequence of the abovementioned copy offload fix. 'sort --compress-program' will continue without compressing temporary files if the specified program cannot be executed. Also malformed shell scripts without a "shebang line" will no longer be executed. \*\* New Features 'numfmt' now accepts the --unit-separator=SEP option, to output or accept a separator between the number and unit. For e.g. "1234 M". \*\* Improvements 'fmt', 'date', 'nl', and 'pr' will now exit promptly upon receiving a write error, which is significant when reading large / unbounded inputs. install, sort, and split now use posix\_spawn() to invoke child programs more efficiently and more independently from their own memory usage. 'numfmt': - parses numbers with a non-breaking space character before a unit - parses numbers containing grouping characters from the current locale - supports a multi-byte --delimiter character - no longer processes input indefinitely in the presence of write errors wc -l now operates 10% faster on hosts that support AVX512 instructions. \*\* Build-related chcon and runcon are not built by default if selinux headers are not present, or if the --without-selinux configure option is specified. This can be overridden with the --with-selinux configure option. nproc no longer fails to build with Android API level <= 20. [build issue introduced in coreutils-9.8]

- [GNU Taler news: GNU Taler 1.1 released](#) (2025/11/09 23:00)

We are happy to announce the release of GNU Taler v1.1.

- [FSF Events: Free Software Directory meeting on IRC: Friday, November 14, starting at 12:00 EST \(17:00 UTC\)](#) (2025/11/07 15:42)

Join the FSF and friends on Friday, November 14 from 12:00 to 15:00 EST (17:00 to 20:00 UTC) to help improve the Free Software Directory.

- [Gary Benson: Terminal colours](#) (2025/11/06 22:48)

\x1B[38;5;105m and \x1B[38;5;141m are a great combo.

- [FSF News: Announcing the winners of the FSF40 Photo Contest](#) (2025/11/06 21:20)

BOSTON, Massachusetts, USA (Thursday, November 6, 2025) – The Free Software Foundation (FSF) today announced the winning photo submissions in the FSF40 Photo Contest held in August.

- [FSF Events: Help us with the GNU Press Shop on Wednesday, January 7, 2026](#) (2025/11/04 17:17)

- [FSF Events: Help us with the GNU Press Shop on Wednesday, December 3, 2025](#) (2025/11/04 17:15)

- [GNU Guix: Update on the Guix Fundraising](#) (2025/11/03 11:00)

We're on our way! It's been a month since we started the fundraising campaign to Sustain and Strengthen Guix. So far we've raised €6562 which is

around 40% of our €15000 annual goal. If you'd like to support the project's fundraiser there's still time, pop over to the donate page now! **DONATE NOW** There have been a range of donations, both one-off and recurring. A few people have made large one-off donations, one of over €2150! There have been a couple between €500-€250 and a few more in the €100 range. These are big contributions to our goal, so I want to thank those individuals for helping out so generously. Just over 100 people (115 right now) have stepped forward to become recurring supporters, pledging a monthly amount to help the project. This is key because it means the project knows there's a regular stream of donations that can pay for the shared resources that we all use. There's been great support with a few people donating €30-€50 a month which is fantastic, the rest at the €10-€15 a month - and one person managed to use the recurring button multiple times to get precisely the amount they wanted to donate monthly! The result is that Open Collective estimates €657.50 a month of recurring donations, and Stripe estimates €720 a month of recurring donations. This is significant because if each person is able to continue giving monthly then annually we'd estimate around €16500 of donations. The maths is simple, the impact significant - a recurring donation of €10 a month is worth €120 a year, that's why recurring donations make such a difference! Of course, people's situations change and they may stop supporting Guix - we've had a couple of cancellations already. So in terms of the actual money we've received we're at ~40% of the €15000 target which I think is pretty good! Thanks to everyone who's supported Guix by donating, you're making a difference and we really appreciate it! If you haven't done it yet, and would like to jump in to support the project then now's a great time! A recurring donation is ideal, but we appreciate any support you can give and every donation gets us a bit closer! **DONATE NOW** Spreading the word Guix is a global community of people, we've had donations from so many places. Where ever you are, it's amazing to think of so many people enjoying, supporting and contributing to Guix. As we're distributed all over the globe we don't have that many ways to keep people informed about the project. I'm sure there are many Guix users who don't know the project needs support. You can help spread the word that Guix is running a fundraiser by talking about it and using this badge. Put it on your social media, your web site or your Git forge account! Thanks to Luis Felipe for creating it. What's next The next few weeks will tell us how many people are able to donate to Guix and the annual budget the project has so that it's sustainable. Then we'll be able plan where we can sustain Guix and where we can do new things to strengthen the project. My goal is for the next blog post is to provide an update on both our fundraising campaign and how we're using the donations that we've received.

- [unifont @ Savannah: Unifont 17.0.03 Released](#) (2025/11/01 19:13)

1 November 2025 Unifont 17.0.03 is now available. This is a minor release aligned with Unicode 17.0.0. This release updates and adds over 100 Chinese ideographs. Download this release from GNU server mirrors at: [https://ftpmirror ... /unifont-17.0.03/](https://ftpmirror.../unifont-17.0.03/) or if that fails, [https://ftp.gnu.org ... /unifont-17.0.03/](https://ftp.gnu.org.../unifont-17.0.03/) or, as a last resort, [ftp://ftp.gnu.org ... /unifont-17.0.03/](ftp://ftp.gnu.org.../unifont-17.0.03/) These files are also available on the unifondry.com website: [https://unifoundr ... /unifont-17.0.03/](https://unifoundr.../unifont-17.0.03/) Font files are in the subdirectory [https://unifoundr ... 0.03/font-builds/](https://unifoundr...0.03/font-builds/) A more detailed description of font changes is available at [https://unifoundr ... nifont/index.html](https://unifoundr...nifont/index.html) and of utility program changes at [https://unifoundr ... nt-utilities.html](https://unifoundr...nt-utilities.html) Enjoy! Paul Hardy GNU Unifont Maintainer

- [Parabola GNU/Linux-libre: dovecot >= 2.4 requires manual intervention](#) (2025/11/01 19:07)

From Arch: The dovecot 2.4 release branch has made breaking changes which result in it being incompatible with any <= 2.3 configuration file. Thus, the dovecot service will no longer be able to start until the configuration file was migrated, requiring manual intervention. For guidance on the 2.3-to-2.4 migration, please refer to the following upstream documentation: Upgrading Dovecot CE from 2.3 to 2.4 Furthermore, the dovecot 2.4 branch no longer supports their replication feature, it was removed. For users relying on the replication feature or who are unable to perform

the 2.4 migration right now, we provide alternative packages available in [extra]: dovecot23 pigeonhole23 dovecot23-fts-elastic dovecot23-fts-xapian The dovecot 2.3 release branch is going to receive critical security fixes from upstream until stated otherwise.

- [Amin Bandali: Free software activities in October 2025](#) (2025/10/31 23:44)

Hello and welcome to my October free software activities report. GNU & FSF GNU Spotlight: I prepared and sent the October GNU Spotlight to the FSF campaigns team, who will review and publish it on the FSF's community blog and as part of the next issue of the monthly Free Software Supporter newsletter. GNU Emacs: bug#79629: I noticed that I was unable to customize the holiday-other-holidays variable using the setopt macro: my change did not seem to take effect. As Eli Zaretskii helpfully pointed out, this was because customizing holiday-other-holidays did not recompute the value of calendar-holidays, which is computed once, when the package is loaded. So I prepared and sent a patch 500a2d0cc55 to recompute calendar-holidays when its components are set. bbabc1db258: While reading about custom-reevaluate-setting in the Startup Summary node of the GNU Emacs Lisp reference manual I noticed a small typo, so I committed a patch to fix it. Misc The Free Software Foundation celebrated its fortieth birthday on 4 October 2025 online and in person in Boston! I was not able to attend the event in person, so I recorded a video for the FSF40 volunteer panel held at the venue. This month at work one of our Elasticsearch clusters experienced partial failure, and we needed to extract document IDs from a backup of one of the cluster's shards. Elasticsearch uses Lucene under the hood and each shard is a standalone Lucene index, so I used Lucene's Java API to write a little GetIDS class to query the index for all of its documents, and for each document print its \_id field, decoding the binary-valued BytesRef as needed. The gotcha was that all of the BytesRefs seemed to have a -1 byte in the beginning, throwing off the recommended BytesRef.utf8ToString() method, so I had to reimplement that method's logic in my program and have it use an adjusted offset + 1 and length - 1 instead. That's about it for this month's report. Take care, and so long for now.

- [parallel @ Savannah: GNU Parallel 20251022 \('Goodall'\) released \[stable\]](#) (2025/10/30 22:58)

GNU Parallel 20251022 ('Goodall') has been released. It is available for download at: [lbrary:///GnuParallel:4](http://lbrary:///GnuParallel:4) Quote of the month: idk who built GNU parallel but I owe them a beer -- ram @h4x0r1ng New in this release: No new features. Bug fixes. GNU Parallel - For people who live life in the parallel lane. If you like GNU Parallel record a video testimonial: Say who you are, what you use GNU Parallel for, how it helps you, and what you like most about it. Include a command that uses GNU Parallel if you feel like it. About GNU Parallel GNU Parallel is a shell tool for executing jobs in parallel using one or more computers. A job can be a single command or a small script that has to be run for each of the lines in the input. The typical input is a list of files, a list of hosts, a list of users, a list of URLs, or a list of tables. A job can also be a command that reads from a pipe. GNU Parallel can then split the input and pipe it into commands in parallel. If you use xargs and tee today you will find GNU Parallel very easy to use as GNU Parallel is written to have the same options as xargs. If you write loops in shell, you will find GNU Parallel may be able to replace most of the loops and make them run faster by running several jobs in parallel. GNU Parallel can even replace nested loops. GNU Parallel makes sure output from the commands is the same output as you would get had you run the commands sequentially. This makes it possible to use output from GNU Parallel as input for other programs. For example you can run this to convert all jpeg files into png and gif files and have a progress bar: `parallel --bar convert {1} {1.}.{2} ::: *.jpg ::: png gif` Or you can generate big, medium, and small thumbnails of all jpeg files in sub dirs: `find . -name '*.jpg' | parallel convert -geometry {2} {1} {1//}/thumb{2}_{1/} :::: - ::: 50 100 200` You can find more about GNU Parallel at: <http://www.gnu.org/s/parallel/> You can install GNU Parallel in just 10 seconds with: `$ (wget -O - pi.dk/3 || lynx -source pi.dk/3 || curl pi.dk/3 || \ fetch -o - http://pi.dk/3 ) > install.sh $ sha1sum install.sh | grep c555f616391c6f7c28bf938044f4ec50 12345678 c555f616391c6f7c 28bf9380 44f4ec50 $ md5sum install.sh | grep 707275363428aa9e9a136b9a7296dfe4 70727536 3428aa9e 9a136b9a 7296dfe4`

\$ sha512sum install.sh | grep b24bfe249695e0236f6bc7de85828fe1f08f4259 83320d89 f56698ec 77454856 895edc3e aa16feab 2757966e 5092ef2d 661b8b45 b24bfe24 9695e023 6f6bc7de 85828fe1 f08f4259 6ce5480a 5e1571b2 8b722f21 \$ bash install.sh Watch the intro video on <http://www.youtube.com/watch?v=L284C9FF2488BC6D1> Walk through the tutorial (man parallel\_tutorial). Your command line will love you for it. When using programs that use GNU Parallel to process data for publication please cite: O. Tange (2018): GNU Parallel 2018, March 2018, <https://doi.org/10.1101/281146>. If you like GNU Parallel: Give a demo at your local user group/team/colleagues Post the intro videos on Reddit/Diaspora\*/forums/blogs/ Identi.ca/Google+/Twitter/Facebook/Linkedin/ mailing lists Get the merchandise <https://gnuparall.com/merch> ... igns/gnu-parallel Request or write a review for your favourite blog or magazine Request or build a package for your favourite distribution (if it is not already there) Invite me for your next conference If you use programs that use GNU Parallel for research: Please cite GNU Parallel in your publications (use --citation) If GNU Parallel saves you money: (Have your company) donate to FSF <https://my.fsf.org/donate/> About GNU SQL GNU sql aims to give a simple, unified interface for accessing databases through all the different databases' command line clients. So far the focus has been on giving a common way to specify login information (protocol, username, password, hostname, and port number), size (database and table size), and running queries. The database is addressed using a DBURL. If commands are left out you will get that database's interactive shell. When using GNU SQL for a publication please cite: O. Tange (2011): GNU SQL - A Command Line Tool for Accessing Different Databases Using DBURLs, ;login: The USENIX Magazine, April 2011:29-32. About GNU Niceload GNU niceload slows down a program when the computer load average (or other system activity) is above a certain limit. When the limit is reached the program will be suspended for some time. If the limit is a soft limit the program will be allowed to run for short amounts of time before being suspended again. If the limit is a hard limit the program will only be allowed to run when the system is below the limit.

- [libc @ Savannah: The GNU C Library version 2.42 is now available](#) (2025/10/27 16:44)

The GNU C Library ===== The GNU C Library version 2.42 is now available. The GNU C Library is used as the C library in the GNU system and in GNU/Linux systems, as well as many other systems that use Linux as the kernel. The GNU C Library is primarily designed to be a portable and high performance C library. It follows all relevant standards including ISO C23 and POSIX.1-2024. It is also internationalized and has one of the most complete internationalization interfaces known. The GNU C Library website is at <http://www.gnu.org/software/libc/> Packages for the 2.42 release may be downloaded from: <http://ftpmirror.gnu.org/libc/> <http://ftp.gnu.org/gnu/libc/> The mirror list is at <http://www.gnu.org/order/ftp.html> Distributions are encouraged to track the release/\* branches corresponding to the releases they are using. The release branches will be updated with conservative bug fixes and new features while retaining backwards compatibility. NEWS for version 2.42 ===== Major new features: The following ISO C23 function families (introduced in TS 18661-4:2015) are now supported in <math.h>. Each family includes functions for float, double, long double, \_FloatN and \_FloatNx, and a type-generic macro in <tgmath.h>. - Power and absolute-value functions: compoundn, pown, powr, rootn, rsqrt. On Linux, the pthread\_gettid\_np function has been added. The ISO C2Y family of unsigned abs functions, i.e. uabs, labs, ullabs, and uimaxabs, is now supported. On Linux, the <termios.h> interface now supports arbitrary baud rates; speed\_t is redefined to simply be the baud rate specified as an unsigned int, which matches the kernel interface. The thread-local cache in malloc (tcache) now supports caching of large blocks. This feature can be enabled by setting the tunable glibc malloc.tcache\_max to a larger value (max 4194304). Tcache is also significantly faster for small sizes. A new configure option, "--enable-sframe", can be used to enable SFrame support of the GNU C Libraries. SFrame is a new stack trace information format which can be used by backtrace. It requires binutils with a minimum version of 2.45. Support for lightweight stack guard pages via madvise and the

MADV\_GUARD\_INSTALL flag has been added to pthread\_create. Additional optimized and correctly rounded mathematical functions have been imported from the CORE-MATH project, in particular acospi, asinpi, atanpi, atan2pi, cospi, sinpi, tanpi. The testsuite has been significantly extended, including coverage of the functionality of the printf and scanf function families in many variants. The manual has been significantly extended and updated, particularly the threads, terminal, filesystem, resource, and math chapters. Code has been added to detect the x86-64 Intel Arrow Lake, Panther Lake, Clearwater Forest, and Diamond Rapids microarchitectures. Regarding S390, support for the new z17 platform has been added. Deprecated and removed features, and other changes affecting compatibility: The glibc.rtdl.execstack tunable now supports a compatibility mode to allow programs that require an executable stack through dynamically loaded shared libraries. On Linux, the <termio.h> header and the definition of struct termio in <sys/ioctl.h> have been removed. The termio interface has been obsolete since the very first version of POSIX.1 in 1988, replaced with <termios.h>. The support for TX lock elision of pthread mutexes has been deprecated on all architectures and will be removed in the next release. On AArch64 Linux targets supporting the Scalable Matrix Extension (SME), setjmp and sigsetjmp will disable the ZA state of SME. Changes to build and runtime requirements: GCC 12.1 or later is now required to build the GNU C Library. GNU Binutils 2.39 or later is now required to build the GNU C Library. Security related changes: The following CVEs were fixed in this release, details of which can be found in the advisories directory of the release tarball: GLIBC-SA-2025-0001: assert: Buffer overflow when printing assertion failure message (CVE-2025-0395) GLIBC-SA-2025-0003: power10: strcmp fails to save and restore nonvolatile vector registers (CVE-2025-5702) GLIBC-SA-2025-0004: power10: strncmp fails to save and restore nonvolatile vector registers (CVE-2025-5745) GLIBC-SA-2025-0005: posix: Fix double-free after allocation failure in regcomp (CVE-2025-8058) The following bugs were resolved with this release: [5994] stdio: fflush after ungetc on seekable input stream [12724] stdio: fclose violates POSIX 2008 on seekable input streams [25263] dynamic-link: ldd and ld.so fail to resolve \$ORIGIN with cross dir symlink [27880] nptl: Please provide a pthread pid accessor [29190] dynamic-link: Symbols with version hash zero lead to crashes, not matched correctly [29459] stdio: fwrite does not return EPIPE when underlying write fails with EPIPE. [31791] nss: [Regression] nss: memory for >8 elements in nsswitch.conf is not freed [32058] libc: qsort leaks memory if C++ exception is thrown from comparison function [32269] dynamic-link: RISC-V IFUNC resolver cannot access gp pointer [32369] stdio: fflush(NULL) doesn't properly flush files opened in read mode [32411] math: THREEp96 seems wrong [32412] dynamic-link: Initial DTV is reallocated using main realloc in auditing mode [32483] locale: ctype.h macros segfault in multithreaded programs with multiple libc.so [32529] stdio: fseek failure on file opened with "rm" mode after ungetc [32535] stdio: fflush failure on file opened with "rm" mode after ungetc [32541] libc: getenv cannot be overridden in static builds [32574] libc: pthread\_attr\_getstacksize/pthread\_attr\_getstack return incorrect main stack size [32612] dynamic-link: [aarch64 PAC] \_dl\_tlsdesc\_dynamic can't be unwound through with \_Unwind\_Backtrace [32626] math: math: log10p1f is not correctly rounded [32627] math: math: sinh is not correctly rounded [32630] math: math: tanf is not correctly rounded for all rounding modes [32653] dynamic-link: Review options for improving both security and backwards compatibility of glibc 2.41 dlopen / execstack handling [32694] math: wrong clang version 3.4 prereq checks in bits/floatn.h for \_\_float128 support, should be 3.9 [32708] libc: Inclusion of sys/mount.h triggers many gcc warnings using -Wshift-overflow=2 -Wsystem-headers [32711] math: math: remainder incorrect sign of zero result [32717] libc: glibc tests fail when bfd is built with --enable-error-execstack=yes [32723] math: [2.41 Regression] /usr/include/bits/floatn.h doesn't work with Intel SYCL compiler [32763] dynamic-link: Static PIE with more than one PT\_LOAD segments at offset 0 segfault [32777] crypt: The performance of the rand() function degradation [32781] libc: Incorrect attribute access for sched\_getattr [32782] nptl: Race conditions in pthread cancellation causing



[GNU Artanis: Why you should port Scheme application to GNU Artanis](#) (2025/10/27 10:51)  
[GNU Artanis: MQTT over Websocket for web application](#) (2025/10/27 10:50)  
[GNU Artanis: How to write a plugin for GNU Artanis](#) (2025/10/27 10:50)  
[GNU Artanis: A secure design of crypto custody with TEE](#) (2025/10/27 10:49)  
[GNU Artanis: The best practice to port Scheme application to Artanis](#) (2025/10/27 10:49)

- [health @ Savannah: GNU Health Hospital Information System 5.0 now available for Raspberry Pi OS](#) (2025/10/24 10:07)  
Dear community I am happy to announce the release of the GNU Health Hospital Information System 5.0 series image for Raspberry Pi OS (version 6.0). This image is the latest of our "GNU Health in a Box" project, which provides a is a full Hospital and Laboratory Information System server in Single Board Computers (like the Raspi or Olimex LIME2). The latest image provides a ready-to-run server with: \* Raspberry Pi OS Desktop \* Debian 13 "trixie" \* GNU Health server HIS 5.0 series server and client \* PostgreSQL 17 database server \* Python 3.13 For more information about the "GNU Health in a box" project, you can visit <https://www.gnuhealth.org/embedded.html> For the download and installation instructions, go to our official documentation page: <https://docs.gnuhealth.org/embedded.html> Happy hacking!
- [GNUnet News: GNUnet 0.25.2](#) (2025/10/19 22:00)  
GNUnet 0.25.2 This is a bugfix release for gnunet 0.25.1. It fixes some regressions and minor bugs. Links Source: <https://ftpmirror.gnu.org/gnunet/gnunet-0.25.2.tar.gz> ( <https://ftpmirror.gnu.org/gnunet/gnunet-0.25.2.tar.gz.sig> ) Detailed list of changes: <https://git.gnunet.org/gnunet.git/log/?h=v0.25.2> NEWS: <https://git.gnunet.org/gnunet.git/tree/NEWS?h=v0.25.2> The GPG key used to sign is: 3D11063C10F98D14BD24D1470B0998EF86F59B6A Note that due to mirror synchronization, not all links may be functional early after the release. For direct access try <https://ftp.gnu.org/gnu/gnunet/>
- [unifont @ Savannah: Unifont 17.0.02 Released](#) (2025/10/18 22:21)  
18 October 2025 Unifont 17.0.02 is now available. This is a minor release aligned with Unicode 17.0.0. - This release includes several glyph updates and many new Chinese ideographs; see the ChangeLog file for details. - src/Makefile now uses CPPFLAGS and LDFLAGS definitions for C program compilation. Download this release from GNU server mirrors at: <https://ftpmirror.gnu.org/unifont-17.0.02/> or if that fails, <https://ftp.gnu.org/pub/unifont-17.0.02/> or, as a last resort, <ftp://ftp.gnu.org/pub/unifont-17.0.02/> These files are also available on the unifondry.com website: <https://unifondry.com/unifont-17.0.02/> Font files are in the subdirectory <https://unifondry.com/0.02/font-builds/> A more detailed description of font changes is available at <https://unifondry.com/unifont/index.html> and of utility program changes at <https://unifondry.com/unifont-utilities.html> Information about Hangul modifications is at <https://unifondry.com/hangul/index.html> and <http://unifondry.com/l-generation.html> Enjoy! Paul Hardy GNU Unifont Maintainer
- [FSF News: FSF announces Librephone project](#) (2025/10/14 21:50)  
BOSTON, Massachusetts, USA (Tuesday, October 14, 2025) – The Free Software Foundation (FSF) today announced its project to bring mobile phone freedom to users. "Librephone" is an initiative to reverse-engineer obstacles preventing mobile phone freedom until its goal is achieved.
- [GNU Taler news: GNU Taler enables digital payments at Datenspuren 2025](#) (2025/10/12 22:00)  
by Mikolai Gütschow
- [Jose E. Marchesi: Version 2 of the Algol 68 GCC Front-End posted](#) (2025/10/12 21:10)  
Today I submitted the version 2 of the patch series for the Algol 68 GCC Front-End: <https://gcc.gnu.org/pipermail/gcc-patches/2025-October/697255.html> This is the deal: Jose E. Marchesi (47): a68: top-level misc files a68: build system a68: build system (regenerated files) a68: documentation a68: command-line options a68: DWARF language codes a68: darwin specific support a68: powerpc specific support a68: gcc/algol68 misc files a68: ga68 compiler driver a68: a681 compiler proper a68: unicode support routines a68: front-end diagnostics a68: parser: entry point a68: parser: AST nodes attributes/types a68: parser: scanner a68: parser: keyword tables management a68: parser: top-down parser a68: parser: parenthesis checker a68: parser: bottom-up parser a68: parser: syntax check for

declarers a68: parser: standard prelude definitions a68: parser: parsing of modes a68: parser: symbol table management a68: parser: static scope checker a68: parser: debug facilities a68: parser: extraction of tags from phrases a68: parser: dynamic stack usage in serial clauses a68: low: lowering entry point and misc handlers a68: low: plain values a68: low: stowed values a68: low: standard prelude a68: low: clauses and declarations a68: low: runtime a68: low: builtins a68: low: ranges a68: low: units and coercions a68: low: modes a68: libga68: sources, spec and misc files a68: libga68: build system a68: libga68: build system (generated files) a68: testsuite: infrastructure a68: testsuite: execution tests 1/2 a68: testsuite: execution tests 2/2 a68: testsuite: compilation tests a68: testsuite: revised MC Algol 68 test set a68: testsuite: mcgt tests

- [FSF News: FSF turns forty with a groundbreaking new project and a new president](#) (2025/10/04 23:45)

- [Amin Bandali: FSF40 volunteer panel](#) (2025/10/04 23:44)

The Free Software Foundation celebrated its fortieth birthday today online and in person in Boston. Forty years of commitment to software freedom and fighting for the freedoms of all computer users. As part of the FSF40 celebration, the FSF held a roundtable panel with FSF volunteers both in-person and online. I was not able to attend the event in person, so I prepared a video recording for the volunteer roundtable that was played at the venue. In the video I talk about my free software journey and how I first got involved, some of the issues facing the world of free software today, and how you can get involved and contribute to free software projects and help the free software movement. Sorry, this embedded video will not work, because your web browser does not support HTML5 video. [ please watch the video in your favourite streaming media player ] Happy 40th birthday, FSF, and here's to 40+ more years of software freedom. Happy hacking! The video recording is Copyright © 2025 Amin Bandali, and is licensed under CC BY-SA 4.0.

- [GNU Guix: Fundraising campaign to sustain GNU Guix](#) (2025/10/03 11:00)

Today we're launching a fundraising campaign to sustain and strengthen GNU Guix. Guix is completely independent from any company or institution, we rely on the support of our community to fund the project. If you can, please help sustain Guix by making a donation. **DONATE NOW** Why we need your support Like many Free Software projects we need financial support because running a project is expensive. We incur costs for development infrastructure, facilitating developer collaboration and supporting the community around the project. As a package manager and GNU/Linux distribution Guix has some unique needs. As the distribution grows and becomes more popular our costs also grow. Each package that's added to the distribution increases the number of builds. And, as more people use Guix the cost of delivering those packages also grows. Sustain Guix To be sustainable we need to match our expenses with our incoming donations. This gives the project certainty that there won't be a sudden funding shortfall. Currently, shortfalls happen. Even recently individual volunteers have had to step in and fund services from their own pockets. That's risky and unsustainable, so we're aiming for stable financial foundations. To achieve that goal we need €15,000 (roughly \$17,500) of donations a year. This would pay for the current infrastructure and project expenses. To be sustainable recurring donations are critical because they provide a regular stream of income that can pay for the ongoing shared resources that we all use. For example, to have a better build farm we'd need more hosting and bandwidth which is a recurring cost. So is the goal achievable? Well, it's definitely a big goal. But, it's only €1,250 a month - so if 125 people contribute €10 a month that would get us to the target and make all the difference! Strengthen Guix We would love to do more, and if there's support from the community then we will. There's lots more we could do! If there's more funding then we'll be able to strengthen Guix by expanding the infrastructure, investing more to support the project and promote Guix. With more support we'd be able to do things like: Improve the overall resilience of our infrastructure so that services are more reliable. Do more to increase the substitute infrastructure's bandwidth and distribution. Tell more people about Guix by attending events, organising user sprints and conferences. Donate Now

to Sustain GuixNow's the time where we ask for your help. Please donate to sustain and strengthen Guix. You can donate through either the FSF or the Guix Foundation using a variety of payment methods. If you haven't heard of the Guix Foundation it's an EU-based non-profit that's dedicated to supporting the development and promotion of GNU Guix. It's a members-driven association, so by becoming a member you'll be supporting Guix and will have a voice in it's activities. Every donations helps, Recurring donations are ideal, but we appreciate any support you can give. Every donation gets us a step closer to being sustainable. DONATE NOW Thank you for your support!

- [FSF News: FSF confirms Ian Kelling as its new president](#) (2025/10/02 20:05)

BOSTON, Massachusetts, USA (Thursday, October 2, 2025) â€” The Free Software Foundation (FSF) announced today that Ian Kelling, senior systems administrator for the organization and the first union member to hold a seat on its board, has been elected as the new president of the FSF.

- [Amin Bandali: Free software activities in September 2025](#) (2025/09/30 23:44)

Summer has been winding down here in southern Ontario and beautiful fall colours have been slowly appearing around us, and with that, it's time for my September free software activities report — albeit a very short one, as it turned out to be a busy month and I had few free software contributions. GNU & FSF GNU Spotlight: I prepared and sent the September GNU Spotlight to the FSF campaigns team, who will review and publish it on the FSF's community blog and as part of the next issue of the monthly Free Software Supporter newsletter. Debian recutils: Last month, I adopted the recutils package in Debian, and uploaded version 1.9-4 to unstable to fix FTBFS bug #1066370. Today, I uploaded 1.9-4~bpo13+1 to trixie-backports to provide an easy way for users of Debian 13 (Trixie) to install recutils. The upload is currently in the backports NEW queue pending review by the backports team, and will appear in the archive if approved. Misc I had a lovely chat with Prot earlier this month for his 'Prot Asks' series, where we talked about free software, free knowledge, the importance of community and the commons, and life in Canada. That's about it for this month's report. I hope to have more to share next month. Take care, and so long for now.

- [libunistring @ Savannah: GNU libunistring-1.4 released](#) (2025/09/27 21:26)

Download from <https://ftp.gnu.org/ftp-gnu/string-1.4.tar.gz> This is a stable release. New in this release: The data tables and algorithms have been updated to Unicode version 17.0.0. Fixed a bug: The functions `u*_grapheme_next` and `u*_grapheme_prev` did not work right for strings with Indic characters, Emojis, or regional indicators.

- [GNUnet News: GNUnet 0.25.1](#) (2025/09/23 22:00)

GNUnet 0.25.1 This is a bugfix release for gnunet 0.25.0. It fixes some regressions and minor bugs. Links Source: <https://ftpmirror.gnu.org/gnunet/gnunet-0.25.1.tar.gz> ( <https://ftpmirror.gnu.org/gnunet/gnunet-0.25.1.tar.gz.sig> ) Detailed list of changes: <https://git.gnunet.org/gnunet.git/log/?h=v0.25.1> NEWS: <https://git.gnunet.org/gnunet.git/tree/NEWS?h=v0.25.1> The GPG key used to sign is: 3D11063C10F98D14BD24D1470B0998EF86F59B6A Note that due to mirror synchronization, not all links may be functional early after the release. For direct access try <https://ftp.gnu.org/gnu/gnunet/>

- [FSF News: FSF confirms Alexandre Oliva to board of directors](#) (2025/09/22 23:05)

BOSTON, Massachusetts, USA (Monday, September 22, 2025) â€” The Free Software Foundation (FSF) today announced the addition of Alexandre Oliva to its board of directors after a three month trial period.

- [parallel @ Savannah: GNU Parallel 20250922 \('Iryna Zarutskaya'\) released \[stable\]](#) (2025/09/22 19:53)

GNU Parallel 20250922 ('Iryna Zarutskaya') has been released. It is available for download at: [lbrary://@GnuParallel:4](https://lbrary.gnu.org/parallel/) Quote of the month: GNU

parallel is awesome. Use it more often in your scripts! -- Wade @WadeGrimshire@twitter New in this release: No new features. Bug fixes and man page updates. GNU Parallel - For people who live life in the parallel lane. If you like GNU Parallel record a video testimonial: Say who you are, what you use GNU Parallel for, how it helps you, and what you like most about it. Include a command that uses GNU Parallel if you feel like it. About GNU Parallel GNU Parallel is a shell tool for executing jobs in parallel using one or more computers. A job can be a single command or a small script that has to be run for each of the lines in the input. The typical input is a list of files, a list of hosts, a list of users, a list of URLs, or a list of tables. A job can also be a command that reads from a pipe. GNU Parallel can then split the input and pipe it into commands in parallel. If you use xargs and tee today you will find GNU Parallel very easy to use as GNU Parallel is written to have the same options as xargs. If you write loops in shell, you will find GNU Parallel may be able to replace most of the loops and make them run faster by running several jobs in parallel. GNU Parallel can even replace nested loops. GNU Parallel makes sure output from the commands is the same output as you would get had you run the commands sequentially. This makes it possible to use output from GNU Parallel as input for other programs. For example you can run this to convert all jpeg files into png and gif files and have a progress bar: `parallel --bar convert {1} {1.}.{2} ::: *.jpg ::: png gif` Or you can generate big, medium, and small thumbnails of all jpeg files in sub dirs: `find . -name '*.jpg' | parallel convert -geometry {2} {1} {1//}/thumb{2}_{1/} ::: - ::: 50 100 200` You can find more about GNU Parallel at: <http://www.gnu.org/s/parallel/> You can install GNU Parallel in just 10 seconds with:

```
$ (wget -O - pi.dk/3 || lynx -source pi.dk/3 || curl pi.dk/3/ || \
  fetch -o - http://pi.dk/3 ) > install.sh $ sha1sum install.sh | grep
c555f616391c6f7c28bf938044f4ec50 12345678 c555f616 391c6f7c 28bf9380 44f4ec50 $ md5sum install.sh | grep
707275363428aa9e9a136b9a7296dfe4 70727536 3428aa9e 9a136b9a 7296dfe4 $ sha512sum install.sh | grep
b24bfe249695e0236f6bc7de85828fe1f08f4259 83320d89 f56698ec 77454856 895edc3e aa16feab 2757966e 5092ef2d 661b8b45
b24bfe24 9695e023 6f6bc7de 85828fe1 f08f4259 6ce5480a 5e1571b2 8b722f21 $ bash install.sh
```

Watch the intro video on <http://www.youtube.com/watch?v=L284C9FF2488BC6D1> Walk through the tutorial (man parallel\_tutorial). Your command line will love you for it. When using programs that use GNU Parallel to process data for publication please cite: O. Tange (2018): GNU Parallel 2018, March 2018, <https://doi.org/10.1146014>. If you like GNU Parallel: Give a demo at your local user group/team/colleagues Post the intro videos on Reddit/Diaspora\*/forums/blogs/Identi.ca/Google+/Twitter/Facebook/Linkedin/ mailing lists Get the merchandise <https://gnuparall.igns.gnu-parallel.com> Request or write a review for your favourite blog or magazine Request or build a package for your favourite distribution (if it is not already there) Invite me for your next conference If you use programs that use GNU Parallel for research: Please cite GNU Parallel in you publications (use --citation) If GNU Parallel saves you money: (Have your company) donate to FSF <https://my.fsf.org/donate/> About GNU SQL GNU sql aims to give a simple, unified interface for accessing databases through all the different databases' command line clients. So far the focus has been on giving a common way to specify login information (protocol, username, password, hostname, and port number), size (database and table size), and running queries. The database is addressed using a DBURL. If commands are left out you will get that database's interactive shell. When using GNU SQL for a publication please cite: hh O. Tange (2011): GNU SQL - A Command Line Tool for Accessing Different Databases Using DBURLs, ;login: The USENIX Magazine, April 2011:29-32. About GNU Niceload GNU niceload slows down a program when the computer load average (or other system activity) is above a certain limit. When the limit is reached the program will be suspended for some time. If the limit is a soft limit the program will be allowed to run for short amounts of time before being suspended again. If the limit is a hard limit the program will only be allowed to run when the system is below the limit.

- [coreutils @ Savannah: coreutils-9.8 released \[stable\]](#) (2025/09/22 18:12)



This is to announce coreutils-9.8, a stable release. Major new features include: - SHA3 hashing with `cksum -a sha3` - nproc now honors Linux cgroup v2 configured CPU quotas - fold is now multi-byte character aware - Base58 encoding with `basenc --base58` - stty can set arbitrary baud rates on supported systems There have also been many bug fixes and other changes as summarized in the NEWS below. There have been 348 commits by 8 people in the 24 weeks since 9.7. Thanks to everyone who has contributed! The following people contributed changes to this release: Bernhard Voelker (1) H. Peter Anvin (4) Bruno Haible (8) Harry Fellowes (1) Collin Funk (61) Paul Eggert (179) Grisha Levit (1) Pádraig Brady (93) Pádraig [on behalf of the coreutils maintainers]

===== Here is the GNU coreutils home page: <https://gnu.org/s/coreutils/> Here are the compressed sources: <https://ftp.gnu.org/gnu/coreutils/coreutils-9.8.tar.gz> (15MB) <https://ftp.gnu.org/gnu/coreutils/coreutils-9.8.tar.xz> (6.0MB) Here are the GPG detached signatures: <https://ftp.gnu.org/gnu/coreutils/coreutils-9.8.tar.gz.sig> <https://ftp.gnu.org/gnu/coreutils/coreutils-9.8.tar.xz.sig> Use a mirror for higher download bandwidth: <https://www.gnu.org/order/ftp.html> Here are the SHA1 and SHA256 checksums: File: coreutils-9.8.tar.gz SHA1 sum: 058b9777cfc78abc4d8893e74ebddfdb344076a9 SHA256 sum: 1be88d53f694671cf7fb85e7723dbd1de9131d031880161b452a0685b986296e File: coreutils-9.8.tar.xz SHA1 sum: a6a58e00688fe0705a7051b36592f1e43e89a9ef SHA256 sum: e6d4fd2d852c9141a1c2a18a13d146a0cd7e45195f72293a4e4c044ec6ccca15 Use a .sig file to verify that the corresponding file (without the .sig suffix) is intact. First, be sure to download both the .sig file and the corresponding tarball. Then, run a command like this: `gpg --verify coreutils-9.8.tar.gz.sig` The signature should match the fingerprint of the following key: `pub rsa4096/0xDF6FD971306037D9 2011-09-23 [SC]`  
 Key fingerprint = 6C37 DC12 121A 5006 BC1D B804 DF6F D971 3060 37D9 uid [ultimate] Pádraig Brady <P@draigBrady.com>  
 uid [ultimate] Pádraig Brady <pixelbeat@gnu.org> If that command fails because you don't have the required public key, or that public key has expired, try the following commands to retrieve or refresh it, and then rerun the 'gpg --verify' command. `gpg --locate-external-key P@draigBrady.com gpg --recv-keys DF6FD971306037D9 wget -q -O-`  
'<https://savannah.gnu.org/project/release-gpgkeys.php?group=coreutils&download=1>' | `gpg --import` - As a last resort to find the key, you can try the official GNU keyring: `wget -q https://ftp.gnu.org/gnu/gnu-keyring.gpg gpg --keyring gnu-keyring.gpg --verify coreutils-9.8.tar.gz.sig` This release is based on the coreutils git repository, available as `git clone https://https.git.savannah.gnu.org/git/coreutils.git` with commit 76e161dc2d70f2d3e0b908e756d62be91bb72c41 tagged as v9.8. For a summary of changes and contributors, see: <https://gitweb.git.savannah.gnu.org/gitweb/?p=coreutils.git;a=shortlog;h=v9.8> or run this command from a git-cloned coreutils directory: `git shortlog v9.7..v9.8` This release was bootstrapped with the following tools: Autoconf 2.72.70-9ff9 Automake 1.17 Gnulib 2025-09-22 4e944641037342d36f31547bec6a73ca910d5bd9 Bison 3.8.2 NEWS \* Noteworthy changes in release 9.8 (2025-09-22) [stable] \*\* Bug fixes 'b2sum' will diagnose --length values that are too big. Previously it would have silently assumed 512 for any larger values. [bug introduced in coreutils-9.6] 'base32' and 'base64' when decoding will again diagnose partially padded data that ends with a newline. [bug introduced in coreutils-9.5] 'basenc -d -i' will now strip '=' characters from the input in encodings where padding characters are not valid. [bug introduced with the basenc program in coreutils-8.31] 'cp -p' had spurious "Operation not supported" failures when copying to non-NFS files from NFSv4 files with trivial ACLs. [bug introduced in coreutils-9.6] 'cp --sparse=always' missed some opportunities to create holes. That is, although the copies had the correct data, sometimes data zeros used extents rather than holes. [This bug was present in "the beginning".] cp missed opportunities to create holes when copying from file systems like squashfs that support SEEK\_HOLE only trivially. [bug introduced in

coreutils-9.0] `cp`, `install`, and `mv` now avoid possible data corruption on `glibc` 2.41 and 2.42 systems when `copy_file_range` is used with ranges > 2GiB, avoiding <https://sourceware.org/PR33245> [bug triggered since coreutils-9.0] `'date'` supports specifying multiple named formats with the last taking precedence. Previously multiple specifications would induce an error. [bug introduced in coreutils-5.90] `'dd oflag=seek_bytes'` no longer mistakenly reports errors when the output file exists on GNU/Hurd. [bug introduced in coreutils-8.16] `'fold'` no longer exhausts memory when processing large inputs with a very large `--width` argument. [This bug was present in "the beginning".] `'install -d'` now produces the correct diagnostic upon failure to create a directory. Previously it would have produced a confusing error about changing permissions. [This bug was present in "the beginning".] `'ls --size --block-size=\k'` could misalign output in locales with multi-byte thousands grouping characters. [This bug was present in "the beginning".] `'nohup'` avoids implementation defined behavior setting `umask`, avoiding a FORTIFY runtime failure on Bionic libc. [This bug was present in "the beginning".] `'od --strings'` with `'-N'` now works correctly. Previously `od` might write a NUL byte after a heap buffer, or output invalid addresses. [These bugs were present in "the beginning".] `'od -w0'` will now issue a diagnostic and exit gracefully. Previously it would have aborted. [bug introduced in coreutils-9.3] `'od -w'` no longer silently mishandles enormous widths like 3037000500. Instead, it either outputs correctly or diagnoses a too-large width. [This bug was present in "the beginning".] `'od +N.'` (where N is a decimal number) works again as per POSIX. [bug introduced in textutils-2.0] `'od /dev/null ++0'` no longer mistakenly treats the `++0` as an offset. [This bug was present in "the beginning".] `'sort'` with key character offsets of `SIZE_MAX`, could induce a read of 1 byte before an allocated heap buffer. For example: `'sort +0.18446744073709551615R input'` on 64 bit systems. [bug introduced in coreutils-7.2] `stdbuf` now works on AIX. Previously it would have been ineffective. [bug introduced with the `stdbuf` program in coreutils-7.5] `'tail -n NUM'` no longer can output more than NUM lines if `stdin` is a largish regular file with a nonzero initial offset, and grows while `'tail'` is reading it. [This bug was present in "the beginning".] `'tail -f -n +NUM'` no longer mishandles NUM values  $\geq$  `UINTMAX_MAX` when the input is seekable. [bug introduced in coreutils-9.6] `'tail --pid'` avoids some unlikely races if the kernel reuses PIDs. [bug introduced in coreutils-9.5] `'tty'` now exits with status 4 with a special diagnostic if `ttname` fails even though standard input is a tty. Formerly it quietly pretended that standard input was not a tty. [This bug was present in "the beginning".] \*\* New Features `basenc` supports the `--base58` option to encode and decode the visually unambiguous Base58 encoding. `'cksum -a'` now supports the `'sha3'` argument, to use the SHA3-224, SHA3-256, SHA3-384, SHA3-512 message digest algorithms depending on the argument passed to the required `--length (-l)` option. `'cksum -a'` now supports the `'sha2'` argument, as a more consistent interface than the existing `'sha224'`, `'sha256'`, `'sha384'`, `'sha512'` arguments, which are now selected with the `--length (-l)` option. `'date'` now outputs dates in the country's native calendar for the Iranian locale (`fa_IR`) and for the Ethiopian locale (`am_ET`), and also does so more consistently for the Thailand locale (`th_TH.UTF-8`). `fold` now supports multi-byte characters, honoring their column width. Also the `--characters (-c)` option was added to wrap at a certain number of characters, similarly to `--bytes` in uni-byte locales. `nproc` now honors any cgroup v2 configured CPU quotas, which may reduce the effective number of processors available. `stty` supports setting arbitrary baud rates on supported systems, like Hurd, Linux with `glibc`  $\geq$  2.42, and some BSDs. Also on other systems the full set of supported baud rates is determined at build time if possible. Commands that support hardware acceleration like `cksum` and `wc` can now disable this acceleration at runtime through the commonly used `GLIBC_TUNABLES` environment variable. For example to disable the use of AVX512 instructions in `cksum`, you can: `export GLIBC_TUNABLES='glibc.cpu.hwcaps=-AVX512F'` \*\* Changes to conform better to POSIX.1-2024 `readlink` now defaults to being verbose if the `POSIXLY_CORRECT` environment variable is set. `realpath` now supports `-E`, which specifies the default behavior. The corresponding long option is `--canonicalize`. `tsort` now accepts and ignores `-w`. \*\* Improvements `'factor'` is now much faster at identifying large

prime numbers, and significantly faster on composite numbers greater than  $2^{128}$ . fold now exits immediately upon receiving a write error, which is significant when reading large / unbounded inputs. 'seq' is more accurate with large integer start values. Previously 'seq 18446744073709551617 inf | head -n1' would output the number before the user specified start value. \*\* Build-related cksum was not compilable by Apple LLVM 10.0.0 x86-64, which lacks support for checking for the VPCLMULQDQ instruction. [bug introduced in coreutils-9.6]

- [gcal @ Savannah: GNU gcal 4.2.0 released](#) (2025/09/21 08:39)

I would like to thank Giuseppe Scrivano for maintaining GNU gcal up to now and I am pleased to take the wheel now. I am happy to also announce the release of version 4.2.0 of gcal. Besides making GNU gcal compilable with gcc15, this release is basically a bug fix release. It contains fixes for bugs reported on the bug-gcal mailing list and filed in the Debian BTS (Bug Tracking Sytem). While becoming the GNU gcal maintainer on savannah went very well, it was not possible to give me upload rights for GNU ftp server. As I am not confident that I get access to the server in the near future, I am publishing the new version at: [https://www.alteh ... gcal-4.2.0.tar.gz](https://www.alteh...gcal-4.2.0.tar.gz) [https://www.alteh ... gcal-4.2.0.tar.xz](https://www.alteh...gcal-4.2.0.tar.xz) and the GPG detached signatures using the key 79DED521FD7B916F726130A6250D40F0A6A2C5C8: [https://www.alteh ... -4.2.0.tar.gz.sig](https://www.alteh...-4.2.0.tar.gz.sig) [https://www.alteh ... -4.2.0.tar.xz.sig](https://www.alteh...-4.2.0.tar.xz.sig) Major changes in release 4.2 New entry added for Good Friday as Czech holiday. New entry added for Good Friday as Hungarian holiday. New entry added for Three Kings Day in Poland since 2011. Print note in case of starting-day different from monday and --iso-week-number is set to yes. No longer report Christmas day as the 26th in Idaho. No longer report Christmas day as the 26th in Ohio.

- [libtool @ Savannah: libtool-2.6.0 released \[alpha\]](#) (2025/09/18 14:55)

Libtoolers! The Libtool Team is pleased to announce the release of libtool 2.6.0, a alpha release. GNU Libtool hides the complexity of using shared libraries behind a consistent, portable interface. GNU Libtool ships with GNU libltdl, which hides the complexity of loading dynamic runtime libraries (modules) behind a consistent, portable interface. There have been 68 commits by 19 people in the 43 weeks since 2.5.4. See the NEWS below for a brief summary. Thanks to everyone who has contributed! The following people contributed changes to this release: Anthony Mallet (1) Bruno Haible (2) Christian Feld (1) Collin Funk (1) Elizabeth Figura (1) Evgeny Grin (1) Frédéric Bérat (1) Gleb Popov (1) Ileana Dumitrescu (47) Julien ÉLIE (1) Karl Berry (1) Kirill Makurin (1) Manoj Gupta (1) Martin Storsjö (1) Michael Haubenwallner (2) Mintsuki (1) Mitch (1) Pierre Ossman (2) Takashi Yano (1) Ileana [on behalf of the libtool maintainers]

===== Here is the GNU libtool home page:

[https://gnu. ... g/s/libtool/](https://gnu...g/s/libtool/) Here are the compressed sources: [https://alpha.gnu ... tool-2.6.0.tar.gz](https://alpha.gnu...tool-2.6.0.tar.gz) (2.1MB) [https://alpha.gnu ... tool-2.6.0.tar.xz](https://alpha.gnu...tool-2.6.0.tar.xz) (1.1MB) Here are the GPG detached signatures: [https://alpha.gnu ... -2.6.0.tar.gz.sig](https://alpha.gnu...-2.6.0.tar.gz.sig) [https://alpha.gnu ... -2.6.0.tar.xz.sig](https://alpha.gnu...-2.6.0.tar.xz.sig) Use a mirror for higher download bandwidth: <https://www.gnu.org/order/ftp.html> Here are the SHA1 and SHA256 checksums: File: libtool-2.6.0.tar.gz SHA1 sum: 681b41409762acb07232d7af7c289cc8a2f851c9 SHA256 sum: ac9d4b8e072aa20d030d5ae039aba3b52bd7967b09e0c21f91f3e3dc1bbb8755 File: libtool-2.6.0.tar.xz SHA1 sum: dff32acab30a9262d19c3559092c2415b6a2b23e SHA256 sum: 69e6d28ae880fda08e0dc080ef2e38077ea2765a0d84e1afcfce1e605c911ac Use a .sig file to verify that the corresponding file (without the .sig suffix) is intact. First, be sure to download both the .sig file and the corresponding tarball. Then, run a command like this: `gpg --verify libtool-2.6.0.tar.gz.sig` The signature should match the fingerprint of the following key: pub rsa4096 2021-09-23 [SC] FA26 CA78 4BE1 8892 7F22 B99F 6570 EA01 146F 7354 uid Ileana Dumitrescu <ileanadumi95@protonmail.com> uid Ileana Dumitrescu <ileanadumitrescu95@gmail.com> If that command fails because you don't have the required public key, or that public key has expired, try the following commands to retrieve or refresh it, and then rerun the 'gpg --verify'

command. `gpg --locate-external-key ileanadumi95@protonmail.com gpg --recv-keys 6570EA01146F7354 wget -q -O- 'https://savannah. ... ol&download=1' | gpg --import -` As a last resort to find the key, you can try the official GNU keyring: `wget -q https://ftp.gnu.o ... u/gnu-keyring.gpg gpg --keyring gnu-keyring.gpg --verify libtool-2.6.0.tar.gz.sig` This release is based on the libtool git repository, available as `git clone https://git.savan ... g/git/libtool.git` with commit `e40fdc22cf727fb885f9df7d9affa827e3253d1c` tagged as `v2.6.0`. For a summary of changes and contributors, see: `https://git.sv.gn ... shortlog;h=v2.6.0` or run this command from a git-cloned libtool directory: `git shortlog v2.5.4..v2.6.0` This release was bootstrapped with the following tools: Autoconf 2.72e Automake 1.18.1 Gnulib 2025-07-29 `f0773a994eb75ceeeb44e39145c842044357a1ad` NEWS Noteworthy changes in release 2.6.0 (2025-09-18) [alpha] \*\* New features: - Add a new tool, `libtool-next-version`, to guide users through updating library versions. - Add tagging for Objective-C and Objective-C++, OBJC and OBJCXX. - Increase 5 digit limit on revision value for libraries to 19 digits, which is referencing Unix epoch time in nanoseconds. - Add configuration options to choose whether to use `'-nostdlib'` to let the compiler frontend decide what standard libraries to link when building C++ shared libraries and modules, `--enable-cxx-stdlib` and `--disable-cxx-stdlib`. - Allow statically linking GCC and Clang compiler support libraries into shared libraries. - Add linking `clang_rt` static archives compiler internal libraries by their absolute path. - Set `'mklink'` as the symlinking tool for MSVC. - Pass `'--target'` architecture flag for Clang. - Support MSYS and MSYS2 file path conversions. \*\* Bug fixes: - Fix wrongly deduplicated compiler dependencies on linux. - Fix NetBSD postdeps for shared libraries. - Fix statically linking dependencies into shared C++ libraries when utilizing clang builtins or g++ options like, `-static-libstdc++`, by using a new configuration option, `--enable-cxx-stdlib`. - Ensure `*-linux-mlibc` host matches to `mlibc` userland rather than matching to GNU/Linux and similar userlands. - Fix hang with `cmd.exe` in MSYS. - For MSVC, fix mishandling compiler flags, symlinking, `cl.exe` `'exp'` extension collision, symbol names, and numerous testsuite bugs. - Fix undeclared reference to `access` on Windows in `libltdl`. - Fix `flang -Wl` flags on FreeBSD. - Fix reordering `'--as-needed'` flag. - Fix `libltdl` early failures for multi-arch. \*\* Changes in supported systems or compilers: - Support additional Intel OneAPI compilers, `'icx'`, `'icpx'`, and `'ifx'`. - Support ML64 (Microsoft Macro Assembler). Enjoy!

- [www @ Savannah: Malware in Proprietary Software - Latest Additions](#) (2025/09/17 15:56)

The initial injustice of proprietary software often leads to further injustices: malicious functionalities. The introduction of unjust techniques in nonfree software, such as back doors, DRM, tethering, and others, has become ever more frequent. Nowadays, it is standard practice. We at the GNU Project show examples of malware that has been introduced in a wide variety of products and dis-services people use everyday, and of companies that make use of these techniques. Here are our latest additions August 2025 Google's Software is Malware Google has announced the inclusion of a "security" measure in Android "smartphones," which will require any software installed in certified Android devices to come from a developer who has gone through Google's new developer verification program. The problem here is not that there's a system that provides trust on the origin of the software. A system like that might be useful, but the end user should still be able to select which organization provides that service, or maybe set up such an organization or renounce the service altogether. Making this verification exclusive to Google makes us question which is the threat here. Is it a user installing malware inadvertently? Or is it the user installing software that makes Google lose money? Academic researchers have published an attack that led Google's supposed "intelligence" [\*] to obey malicious commands to manipulate devices in the user's home. Giving Google control of your devices, or control of your own computing that you do on their servers, inevitably makes you vulnerable to Google. This announcement shows that the vulnerability includes third-party crackers [\*\*] too. The article says that the crack discoverers worked with Google to "mitigate" the danger. What, concretely, does "mitigate" mean here? Probably in this case it is

a weasel word to suggest fixing a problem without claiming to have fixed it. [\*] Let's not call these systems “artificial intelligence.” Intelligence is something they do not have. [\*\*] Please note that the article wrongly refers to crackers as “hackers.” July 2025 Malware in Mobile Devices Samsung disabled the option to unlock the bootloader in the new version of its proprietary Android distribution, making it impossible for users to install a third-party operating system on their mobile devices. This takes away people's right to run the system of their choice (which we hope is a free/libre system), and to extend the life of their device once Samsung stops supporting its software.

From:

<https://wiki.tromjaro.alexio.tf/> - **TROMjaro wiki**

Permanent link:

<https://wiki.tromjaro.alexio.tf/doku.php?id=news:planet:gnu&rev=1583652621>

Last update: **2021/10/30 11:38**

