systemd v259~rc1

Pre-release

github-actions released this Nov 17 · 419 commits to main since this release · Immutable

v259-rc1 · d810cfc

Compare

CHANGES WITH 259 in spe:

Announcements of Future Feature Removals and Incompatible Changes:

* Support for System V service scripts is deprecated and will be removed in v260. Please make sure to update your software *now* to include a native systemd unit file instead of a legacy System V script to retain compatibility with future systemd releases. Following components will be removed:

  * systemd-rc-local-generator,
  * systemd-sysv-generator,
  * systemd-sysv-install (hook for systemctl enable/disable/is-enabled).

* Required minimum versions of following components are planned to be raised in v260:

  * Linux kernel >= 5.10 (recommended >= 5.14),
  * glibc >= 2.34,
  * libxcrypt >= 4.4.0 (libcrypt in glibc will be no longer supported),
  * util-linux >= 2.37,
  * elfutils >= 0.177,
  * openssl >= 3.0.0,
  * cryptsetup >= 2.4.0,
  * libseccomp >= 2.4.0,
  * python >= 3.9.0.

  Please provide feedback on systemd-devel if this would cause problems.

* The parsing of RootImageOptions= and the mount image parameters of ExtensionImages= and MountImages= will be changed in the next version so that the last duplicated definition for a given partition wins and is applied, rather than the first, to keep these options coherent with other unit settings.

Feature Removals and Incompatible Changes:

* The cgroup2 file system is now mounted with the "memory_hugetlb_accounting" mount option, supported since kernel 6.6. This means that HugeTLB memory usage is now counted towards the cgroup’s overall memory usage for the memory controller.

* The default storage mode for the journal is now 'persistent'. Previously, the default was 'auto', so the presence or lack of /var/log/journal determined the default storage mode, if no overriding configuration was provided. The default can be changed with -Djournal-storage-default=.

* systemd-networkd and systemd-nspawn no longer support creating NAT rules via iptables/libiptc APIs; only nftables is now supported.

* systemd-boot's and systemd-stub's support for TPM 1.2 has been removed (only TPM 2.0 supported is retained). The security value of TPM 1.2 support is questionable in 2025, and because we never supported it in userspace, it was always quite incomplete to the point of uselessness.

* The image dissection logic will now enforce the VFAT file system type for XBOOTLDR partitions, similar to how it already does this for the ESP. This is done for security, since both the ESP and XBOOTLDR must be directly firmware-accessible and thus cannot by protected by cryptographic means. Thus it is essential to not mount arbitrarily complex file systems on them. This restriction only applies if automatic dissection is used. If other file system types shall be used for XBOOTLDR (not recommended) this can be achieved via explicit /etc/fstab entries.

* systemd-machined will now expose "hidden" disk images as read-only by default (hidden images are those whose name begins with a dot). They were already used to retain a pristine copy of the downloaded image, while modifications were made to a 2nd, local writable copy of the image. Hence, effectively they were read-only already, and this is now official.

* The LUKS volume label string set by systemd-repart no longer defaults to the literal same as the partition and file system label, but is prefixed with "luks-". This is done so that on LUKS enabled images a conflict between /dev/disk/by-label/ symlinks is removed, as this symlink is generated both for file system and LUKS superblock labels. There's a new VolumeLabel= setting for partitions that can be used to expicitly choose a LUKS superblock label, which can be used to explicitly revert to the old naming, if required.

Service manager/PID1:

* The service manager's Varlink IPC has been extended considerably. It now exposes service execution settings and more. Its Unit.List() call now can filter by cgroup or invocation ID.

* The service manager now exposes Reload() and Reexecute() Varlink IPC calls, mirroring the calls of the same name accessible via D-Bus.

* The $LISTEN_FDS protocol has been extended to support pidfd inode IDs. The $LISTEN_PID environment variable is now augmented with a new $LISTEN_PIDFDID environment variable which contains the inode ID of the pidfd of the indicated process. This removes any ambiguity regarding PID recycling: a process which verified that $LISTEN_PID points to its own PID can now also verify the pidfd inode ID, which does not recycle IDs.

* The log message made when a service exits will now show the wallclock time the service took in addition to the previously shown CPU time.

* A new pair of properties OOMKills and ManagedOOMKills are now exposed on service units (and other unit types that spawn processes) that count the number of process kills made by the kernel or systemd-oomd.

* The service manager gained support for a new RootDirectoryFileDescriptor= property when creating transient service units. It is similar to RootDirectory= but takes a file descriptor rather than a path to the new root directory to use.

* The service manager now supports a new UserNamespacePath= setting which mirrors the existing IPCNamespacePath= and NetworkNamespacePath= options, but applies to Linux user namespaces.

* The service manager gained a new ExecReloadPost= setting to configure commands to execute after reloading of the configuration of the service has completed.

* Service manager job activation transactions now get a per-system unique 64-bit numeric ID assigned. This ID is logged as an additional log field for in messages related to the transaction.

* The service manager now keeps track of transactions with ordering cycles and exposes them in the TransactionsWithOrderingCycle D-Bus property.

systemd-sysext/systemd-confext:

* systemd-sysext and systemd-confext now support configuration files /etc/systemd/systemd-sysext.conf and /etc/systemd/systemd-confext.conf, which can be used to configure mutability or the image policy to apply to DDI images.

* systemd-sysext's and systemd-confext's --mutable= switch now accepts a new value "help" for listing available mutability modes.

* systemd-sysext now supports configuring additional overlayfs mount settings via the $SYSTEMD_SYSEXT_OVERLAYFS_MOUNT_OPTIONS environment variable. Similarly systemd-confext now supports $SYSTEMD_CONFEXT_OVERLAYFS_MOUNT_OPTIONS.

systemd-vmspawn/systemd-nspawn:

* systemd-vmspawn will now initialize the "serial" fields of block devices attached to VMs to the filename of the file backing them on the host. This makes it very easy to reference the right media in case many block devices from files are attached to the same VM via the /dev/disk/by-id/… links in the VM.

* systemd-nspawn's .nspawn file gained support for a new NamespacePath= setting in the [Network] section which takes a path to a network namespace inode, and which ensures the container is run inside that when booted. (This was previously only available via a command line switch.)

* systemd-vmspawn gained two new switches --bind-user=/--bind-user-shell= which mirror the switches of the same name in systemd-nspawn, and allow sharing a user account from the host inside the VM in a simple one-step operation.

* systemd-vmspawn and systemd-nspawn gained a new --bind-user-group= switch to add a user bound via --bind-user= to the specified group (useful in particular for the 'wheel' or 'empower' groups).

* systemd-vmspawn now configures RSA4096 support in the vTPM, if swtpm supports it.

* systemd-vmspawn now enables qemu guest agent via the org.qemu.guest_agent.0 protocol when started with --console=gui.

systemd-repart:

* repart.d/ drop-ins gained support for a new TPM2PCRs= setting, which can be used to configure the set of TPM2 PCRs to bind disk encryption to, in case TPM2-bound encryption is used. This was previously only settable via the systemd-repart command line. Similarly, KeyFile= has been added to configure a binary LUKS key file to use.

* systemd-repart's functionality is now accessible via Varlink IPC.

* systemd-repart may now be invoked with a device node path specified as "-". Instead of operating on a block device this will just determine the minimum block device size required to apply the defined partitions and exit.

* systemd-repart gained two new switches --defer-partitions-empty=yes and --defer-partitions-factory-reset=yes which are similar to --defer-partitions= but instead of expecting a list of partitions to defer will defer all partitions marked via Format=empty or FactoryReset=yes. This functionality is useful for installers, as partitions marked empty or marked for factory reset should typically be left out at install time, but not on first boot.

* The Subvolumes= values in repart.d/ drop-ins may now be suffixed with :nodatacow, in order to create subvolumes with data Copy-on-Write disabled.

systemd-udevd:

* systemd-udevd rules gained support for OPTIONS="dump-json" to dump the current event status in JSON format. This generates output similar to "udevadm test --json=short".

* The net_id builtin for systemd-udevd now can generate predictable interface names for Wifi devices on DeviceTree systems.

* systemd-udevd and systemd-repart will now reread partition tables on block devices in a more graceful, incremental fashion. Specifically, they no longer use the kernel BLKRRPART ioctl() which removes all in-memory partition objects loaded into the kernel and then recreates them as new objects. Instead they will use the BLKPG ioctl() to make minimal changes, and individually add, remove, or grow modified partitions, avoiding removal/re-adding where the partitions were left unmodified on disk. This should greatly improve behaviour on systems that make modifications to partition tables on disk while using them.

* A new udev property ID_BLOCK_SUBSYSTEM is now exposed on block devices reporting a short identifier for the subsystem a block device belongs to. This only applies to block devices not connected to a regular bus, i.e. virtual block devices such as loopback, DM, MD, or zram.

* systemd-udevd will now generate /dev/gpio/by-id/… symlinks for GPIO devices.

systemd-homed/homectl:

* homectl's --recovery-key= option may now be used with the "update" command to add recovery keys to existing user accounts. Previously, recovery keys could only be configured during initial user creation.

* Two new --prompt-shell= and --prompt-groups= options have been added to homectl to control whether to query the user interactively for a login shell and supplementary groups memberships when interactive firstboot operation is requested. The invocation in systemd-homed-firstboot.service now turns both off by default.

systemd-boot/systemd-stub:

* systemd-boot now supports log levels. The level may be set via log-level= in loader.conf and via the SMBIOS Type 11 field 'io.systemd.boot.loglevel='.

* systemd-boot's loader.conf file gained support for configuring the SecureBoot key enrollment time-out via secure-boot-enroll-timeout-sec=.

* Boot Loader Specification Type #1 entries now support a "profile" field which may be used to explicitly select a profile in multi-profile UKIs invoked via the "uki" field.

sd-varlink/varlinkctl:

* sd-varlink's sd_varlink_set_relative_timeout() call will now reset the timeout to the default if 0 is passed.

* sd-varlink's sd_varlink_server_new() call learned two new flags SD_VARLINK_SERVER_HANDLE_SIGTERM + SD_VARLINK_SERVER_HANDLE_SIGINT, which are honoured by sd_varlink_server_loop_auto() and will cause it to exit processing cleanly once SIGTERM/SIGINT are received.

* varlinkctl in --more mode will now send a READY=1 sd_notify() message once it receives the first reply. This is useful for tools or scripts that wrap it (and implement the $NOTIFY_SOCKET protocol) to know when a first confirmation of success is received.

* sd-varlink gained a new sd_varlink_is_connected() call which reports whether a Varlink connection is currently connected.

Shared library dependencies:

* Linux audit support is now implemented via dlopen() rather than regular dynamic library linking. This means the dependency is now weak, which is useful to reduce footprint inside of containers and such, where Linux audit doesn't really work anyway.

* Similarly PAM support is now implemented via dlopen() too (except for the PAM modules pam_systemd + pam_systemd_home + pam_systemd_loadkey, which are loaded by PAM and hence need PAM anyway to operate).

* Similarly, libacl support is now implemented via dlopen().

* Similarly, libblkid support is now implemented via dlopen().

* Similarly, libseccomp support is now implemented via dlopen().

* Similarly, libselinux support is now implemented via dlopen().

* Similarly, libmount support is now implemented via dlopen(). Note, that libmount still must be installed in order to invoke the service manager itself. However, libsystemd.so no longer requires it, and neither do various ways to invoke the systemd service manager binary short of using it to manage a system.

* systemd no longer links against libcap at all. The simple system call wrappers and other APIs it provides have been reimplemented directly in systemd, which reduced the codebase and the dependency tree.

systemd-machined/systemd-importd:

* systemd-machined gained support for RegisterMachineEx() + CreateMachineEx() method calls which operate like their counterparts without "Ex", but take a number of additional parameters, similar to what is already supported via the equivalent functionality in the Varlink APIs of systemd-machined. Most importantly, they support PIDFDs instead of PIDs.

* systemd-machined may now also run in a per-user instance, in addition to the per-system instance. systemd-vmspawn and systemd-nspawn have been updated to register their invocations with both the calling user's instance of systemd-machined and the system one, if permissions allow it. machinectl now accepts --user and --system switches that control which daemon instance to operate on. systemd-ssh-proxy now will query both instances for the AF_VSOCK CID.

* systemd-machined implements a resolve hook now, so that the names of local containers and VMs can be resolved locally to their respective IP addresses.

* systemd-importd's tar extraction logic has been reimplemented based on libarchive, replacing the previous implementation calling GNU tar. This completes work begun earlier which already ported systemd-importd's tar generation.

* systemd-importd now may also be run as a per-user service, in addition to the existing per-system instance. It will place the downloaded images in ~/.local/state/machines/ and similar directories. importctl gained --user/--system switches to control which instance to talk to.

systemd-firstboot:

* systemd-firstboot's and homectl's interactive boot-time interface have been updated to show a colored bar at the top and bottom of the screen, whose color can be configured via /etc/os-release. The bar can be disabled via the new --chrome= switches to both tools.

* systemd-firstboot's and homectl's interactive boot-time interface will now temporarily mute the kernel's and PID1's own console output while running, in order to not mix the tool's own output with the other sources. This logic can be controlled via the new --mute-console= switches to both tools. This is implemented via a new systemd-mute-console component (which provides a simple Varlink interface).

* systemd-firstboot gained a new switch --prompt-keymap-auto. When specified, the tool will interactively query the user for a keymap when running on a real local VT console (i.e. on a user device where the keymap would actually be respected), but not if invoked on other TTYs (such as a serial port, hypervisor console, SSH, …), where the keymap setting would have no effect anyway. The invocation in systemd-firstboot.service now uses this.

systemd-creds:

* systemd-creds's Varlink IPC API now supports a new "withKey" parameter on the Encrypt() method call, for selecting what to bind the encryption to precisely, matching the --with-key= switch on the command line.

* systemd-creds now allow explicit control of whether to accept encryption with a NULL key when decrypting, via the --allow-null and --refuse-null switches. Previously only the former existed, but null keys were also accepted if UEFI SecureBoot was reported off. This automatism is retained, but only if neither of the two switches are specified. The systemd-creds Varlink IPC API learned similar parameters on the Decrypt() call.

systemd-networkd:

* systemd-networkd's DHCP sever support gained two settings EmitDomain= and Domain= for controlling whether leases handed out should report a domain, and which. It also gained a per-static lease Hostname= setting for the hostname of the client.

* systemd-networkd now exposes a Describe() method call to show network interface properties.

* systemd-networkd now implements a resolve hook for its internal DHCP server, so that the hostnames tracked in DHCP leases can be resolved locally. This is now enabled by default for the DHCP server running on the host side of local systemd-nspawn or systemd-vmspawn networks.

systemd-resolved:

* systemd-resolved gained a new Varlink IPC method call DumpDNSConfiguration() which returns the full DNS configuration in one reply. This is exposed by resolvectl --json=.

* systemd-resolved now allows local, privileged services to hook into local name resolution requests. For that a new directory /run/systemd/resolve.hook/ has been introduced. Any privileged local service can bind an AF_UNIX Varlink socket there, and implement the simple io.systemd.Resolve.Hook Varlink API on it. If so it will receive a method call on it for each name resolution request, which it can then reply to. It can reply positively, deny the request or let the regular request handling take place.

* DNS0 has been removed from the default fallback DNS server list of systemd-resolved, since it ceased operations.

TPM2 infrastructure:

* systemd-pcrlock no longer locks to PCR 12 by default, since its own policy description typically ends up in there, as it is passed into a UKI via a credential, and such credentials are measured into PCR 12.

* The TPM2 infrastructure gained support for additional PCRs implemented via TPM2 NV Indexes in TPM2_NT_EXTEND mode. These additional PCRs are called "NvPCRs" in our documentation (even though they are very much volatile, much like the value of TPM2_NT_EXTEND NV indexes, from which we inherit the confusing nomenclature). By introducing NvPCRs the scarcity of PCRs is addressed, which allows us to measure more resources later without affecting the definition and current use of the scarce regular PCRs. Note that NvPCRs have different semantics than PCRs: they are not available pre-userspace (i.e. initrd userspace creates them and initializes them), including in the pre-kernel firmware world; moreover, they require an explicit "anchor" initialization of a privileged per-system secret (in order to prevent attackers from removing/recreating the backing NV indexes to reset them). This makes them predictable only if the result of the anchor measurement is known ahead of time, which will differ on each installed system. Initialization of defined NvPCRs is done in systemd-tpm2-setup.service in the initrd. Information about the initialization of NvPCRs is measured into PCR 9, and finalized by a separator measurement. The NV index base handle is configurable at build time via the "tpm2-nvpcr-base" meson setting. It currently defaults to a value the TCG has shown intent to assign to Linux, but this has not officially been done yet. systemd-pcrextend and its Varlink APIs have been extended to optionally measure into an NvPCR instead of a classic PCR.

* A new service systemd-pcrproduct.service is added which is similar to systemd-pcrmachine.service but instead of the machine ID (i.e. /etc/machined-id) measures the product ID (as reported by SMBIOS or Devicetree). It uses a new NvPCR called "hardware" for this.

* systemd-pcrlock has been updated to generate CEL event log data covering NvPCRs too.

systemd-analyze:

* systemd-analyze gained a new verb "dlopen-metadata" which can show the dlopen() weak dependency metadata of an ELF binary that declares that.

* A new verb "nvpcrs" has been added to systemd-analyze, which lists NvPCRs with their names and values, similar to the existing "pcrs" operation which does the same for classic PCRs.

systemd-run/run0:

* run0 gained a new --empower switch. It will invoke a new session with elevated privileges – without switching to the root user. Specifically, it sets the full ambient capabilities mask (including CAP_SYS_ADMIN), which ensures that privileged system calls will typically be permitted. Moreover, it adds the session processes to the new "empower" system group, which is respected by polkit and allows privileged access to most polkit actions. This provides a much less invasive way to acquire privileges, as it will not change $HOME or the UID and hence risk creation of files owned by the wrong UID in the user's home. (Note that --empower might not work in all cases, as many programs still do access checks purely based on the UID, without Linux process capabilities or polkit policies having any effect on them.)

* systemd-run gained support for --root-directory= to invoke the service in the specified root directory. It also gained --same-root-dir (with a short switch -R) for invoking the new service in the same root directory as the caller's. --same-root-dir has also been added to run0.

sd-event:

* sd-event's sd_event_add_child() and sd_event_add_child_pidfd() calls now support the WNOWAIT flag which tells sd-event to not reap the child process.

* sd-event gained two new calls sd_event_set_exit_on_idle() and sd_event_get_exit_on_idle(), which enable automatic exit from the event loop if no enabled (non-exit) event sources remain.

Other:

* User records gained a new UUID field, and the userdbctl tool gained the ability to search for user records by UUID, via the new --uuid= switch. The userdb Varlink API has been extended to allow server-side searches for UUIDs.

* systemd-sysctl gained a new --inline switch, similar to the switch of the same name systemd-sysusers already supports.

* systemd-cryptsetup has been updated to understand a new tpm2-measure-keyslot-nvpcr= option which takes an NvPCR name to measure information about the used LUKS keyslot into. systemd-gpt-auto-generator now uses this for a new "cryptsetup" NvPCR.

* systemd will now ignore configuration file drop-ins suffixed with ".ignore" in most places, similar to how it already ignores files with suffixes such as ".rpmsave". Unlike those suffixes, ".ignore" is package manager agnostic.

* systemd-modules-load will now load configured kernel modules in parallel.

* systemd-integrity-setup now supports HMAC-SHA256, PHMAC-SHA256, PHMAC-SHA512.

* systemd-stdio-bridge gained a new --quiet option.

* systemd-mountfsd's MountImage() call gained support for explicitly controlling whether to share dm-verity volumes between images that have the same root hashes. It also learned support for setting up bare file system images with separate Verity data files and signatures.

* journalctl learned a new short switch "-W" for the existing long switch "--no-hostname".

* system-alloc-{uid,gid}-min are now exported in systemd.pc.

* Incomplete support for musl libc is now available by setting the "libc" meson option to "musl". Note that systemd compiled with musl has various limitations: since NSS or equivalent functionality is not available, nss-systemd, nss-resolve, DynamicUser=, systemd-homed, systemd-userdbd, the foreign UID ID, unprivileged systemd-nspawn, systemd-nsresourced, and so on will not work. Also, the usual memory pressure behaviour of long-running systemd services has no effect on musl. We also implemented a bunch of shims and workarounds to support compiling and running with musl. Caveat emptor.

  This support for musl is provided without a promise of continued support in future releases. We'll make the decision based on the amount of work required to maintain the compatibility layer in systemd, how many musl-specific bugs are reported, and feedback on the desirability of this effort provided by users and distributions.

Contributors

Contributions from: Alan Brady, Alberto Planas, Aleksandr Mezin, Allison Karlitskaya, Andreas Schneider, Anton Tiurin, Antonio Alvarez Feijoo, Arian van Putten, Armin Wolf, Bastian Almendras, Charlie Le, Chen Qi, Chris Down, Christian Hesse, Christoph Anton Mitterer, Daan De Meyer, Daniel Brackenbury, Daniel Foster, Daniel Hast, Danilo Spinella, David Tardon, Dimitri John Ledkov, Dr. David Alan Gilbert, Duy Nguyen Van, Emanuele Giuseppe Esposito, Emil Renner Berthing, Eric Curtin, Erin Shepherd, Evgeny Vereshchagin, Felix Pehla, Florian, Francesco Valla, Franck Bui, Frantisek Sumsal, Gero Schwäricke, Goffredo Baroncelli, Govind Venugopal, Guido Günther, Hans de Goede, Igor Opaniuk, Ingo Franzki, Itxaka, Ivan Kruglov, Jelle van der Waa, Jim Spentzos, Joshua Krusell, Justin Kromlinger, Jörg Behrmann, Kai Lueke, Kai Wohlfahrt, Le_Futuriste, Lennart Poettering, Luca Boccassi, Lucas Adriano Salles, Lukáš Nykrýn, Managor, Mantas Mikulėnas, Marcel Leismann, Marcos Alano, Marien Zwart, Markus Boehme, Masanari Iida, Matteo Croce, Maximilian Bosch, Michal Sekletár, Mike Yuan, Miroslav Lichvar, Moisticules, Natalie Vock, Nick Labich, Nick Rosbrook, Nils K, Osama Abdelkader, Oğuz Ersen, Pascal Bachor, Peter Hutterer, Quentin Deslandes, Rafael Fontenelle, Ronan Pigott, Ryan Brue, Sebastian Gross, Septatrix, Taylan Kammer, Thomas Blume, Thomas Mühlbacher, Tobias Heider, Xarblu, Yu Watanabe, Zbigniew Jędrzejewski-Szmek, anthisfan, cvlc12, dgengtek, dramforever, gvenugo3, helpvisa, huyubiao, jouyouyun, jsks, kanitha chim, n0099, ners, nkraetzschmar, nl6720, theSillywhat, val4oss, 雪叶

— Edinburgh, 2025/11/17