Red Hat Enterprise Linux 10

10.0 Release Notes

Release Notes for Red Hat Enterprise Linux 10.0

Last Updated: 2025-09-03

Red Hat Enterprise Linux 10 10.0 Release Notes

Release Notes for Red Hat Enterprise Linux 10.0

Legal Notice

Abstract

The Release Notes provide high-level coverage of the improvements and additions that have been implemented in Red Hat Enterprise Linux 10.0 and document known issues in this release, as well as notable fixed issues, Technology Previews, deprecated functionalities, functionalities removed in RHEL 10, and other details. For information about installing Red Hat Enterprise Linux, see Installation.

Table of Contents

PROVIDING FEEDBACK ON RED HAT DOCUMENTATION

We appreciate your feedback on our documentation. Let us know how we can improve it.

Submitting feedback through Jira (account required)

1. Log in to the Jira website.
2. Click **Create** in the top navigation bar
3. Enter a descriptive title in the **Summary** field.
4. Enter your suggestion for improvement in the **Description** field. Include links to the relevant parts of the documentation.
5. Click **Create** at the bottom of the dialogue.

CHAPTER 1. OVERVIEW

1.1. MAJOR CHANGES IN RHEL 10.0

Key highlights for RHEL installer:

- The newly created users will have administrative privileges by default, unless you deselect the option.
- You can now set the required time zone by using new options instead of the time zone map.
- The remote desktop protocol (RDP) for graphical remote access replaces VNC.

Key highlights for RHEL image builder:

- RHEL image builder **cockpit-composer** package has been deprecated and replaced with the new **cockpit-image-builder** plugin.
- Disk images, such as AWS or KVM, do not have a separate /**boot** partition.

For more information, see New features and enhancements - Installer and image creation .

Security

As a Technology Preview, system-wide cryptographic policies (**crypto-policies**), the OpenSSL, GnuTLS, and NSS toolkits, and the OpenSSH suite now work with **post-quantum (PQ) algorithms**.

With the new **sudo RHEL system role**, you can consistently manage **sudo** configuration at scale across your RHEL systems.

RHEL 10 introduces **Sequoia PGP** tools **sq** and **sqv** that complement the existing GnuPG tools for managing OpenPGP encryption and signatures.

The **OpenSSL** TLS toolkit introduces creation of FIPS-compliant PKCS #12 files, the **pkcs11-provider** for using hardware tokens, and many additional improvements.

RHEL 10 contains the **OpenSSH** suite in version 9.9, which provides many fixes and improvements over OpenSSH 8.7, which was provided in RHEL 9.

The **SELinux** userspace release 3.8 introduces a new option for **audit2allow** that provides CIL output mode, Wayland support for the SELinux sandbox, and other improvements.

The **Keylime** agent component is provided in version 0.2.5, which provides support for Initial Device Identity (IDevID) and Initial Attestation Key (IAK) for device identity and uses TLS 1.3 by default. In addition, the new **keylime-policy** tool integrates all management tasks of Keylime policies.

The **security compliance** offering has evolved substantially compared to RHEL 9 in both the tooling and content. You can still perform all the actions you need to bring your systems close to a compliant state although you might need to use different tools than in previous versions of RHEL.

See New features - Security for more information.

Kernel

This release delivers several important improvements and new features to the kernel. We have expanded graphical capabilities by introducing limited support for **virtio-gpu** on IBM Z (**s390x**) systems for enhanced virtualization experiences.

To improve usability and deployment, the **rteval** utility has been containerized, simplifying its integration into various environments. You can run the **rteval** utility with all its runtime dependencies from a container image publicly available through the Quay.io container registry.

A dynamic **EFIVARS** pstore backend is now supported. You can switch between supported backends such **NVMe** and **EFIVARS** without rebooting the system.

The **rh_waived** kernel command-line boot parameter is now supported to enable waived features in RHEL. However, waived features are disabled by default in RHEL 10.

Dynamic programming languages, web and database servers

RHEL 10.0 provides the following dynamic programming languages:

- **Python 3.12**
- **Ruby 3.3**
- **Node.js 22**
- **Perl 5.40**
- **PHP 8.3**

RHEL 10.0 includes the following version control systems:

- **Git 2.45**
- **Subversion 1.14**

The following web servers are distributed with RHEL 10.0:

- **Apache HTTP Server 2.4.62**
- **nginx 1.26**

The following proxy caching servers are available:

- **Varnish Cache 7.4**
- **Squid 6.10**

RHEL 10.0 offers the following database servers:

- **MariaDB 10.11**
- **MySQL 8.4**
- **PostgreSQL 16**
- **Valkey 7.2**

See New features - Dynamic programming languages, web and database servers for more information.

Compilers and development tools

System toolchain

The following system toolchain components are available with RHEL 10.0:

- GCC 14.2
- glibc 2.39
- Annobin 12.55
- binutils 2.41

Performance tools and debuggers
The following performance tools and debuggers are available with RHEL 10.0:

- GDB 14.2
- Valgrind 3.23.0
- SystemTap 5.1
- Dyninst 12.3.0
- elfutils 0.192
- libabigail 2.6

Performance monitoring tools
The following performance monitoring tools are available with RHEL 10.0:

- PCP 6.3.0
- Grafana 10.2.6

Compiler toolsets
The following compiler toolsets are available with RHEL 10.0:

- LLVM Toolset 19.1.7
- Rust Toolset 1.84.1
- Go Toolset 1.23

For detailed changes, see New features - Compilers and development tools .

Identity Management
Key highlights for Identity Management:

- The IdM server functions only partially or not at all. Specifically, you cannot install the **ipa-server-dns** package, and the embedded DNS server cannot be configured using the **-setup-dns** option. Until the necessary updates to **bind-dyndb-ldap** and other impacted components are completed, the integrated DNS feature remains unavailable.

See Known Issues - Identity Management for more information.

The web console
With the new **File browser** provided by the **cockpit-files** package, you can manage files and directories in the RHEL web console.

See New features - The web console for more information.

Red Hat Enterprise Linux system roles

Notable new features in 10.0 RHEL system roles:

- With the new RHEL system role **aide**, you can detect unauthorized changes to files, directories, and system binaries.
- With the **systemd** RHEL system role you can now manage user units in addition to system units
- You can use the **ha_cluster** RHEL system role to export the **corosync** configuration of an existing cluster in a format that can be fed back to the role to create the same cluster.
- You can use the **podman** RHEL system role to manage the quadlet units of type **Pod**.
- The **metrics** RHEL system role now supports Valkey as an alternative to Redis.

See New features and enhancements - Red Hat Enterprise Linux System Roles for more information.

1.2. IN-PLACE UPGRADE

In-place upgrade from RHEL 9 to RHEL 10

The supported in-place upgrade paths currently are:

- From RHEL 9.6 to RHEL 10.0 on the following architectures:
  - AMD and Intel 64-bit architectures (x86-64-v3)
  - The 64-bit ARM architecture (ARMv8.0-A)
  - IBM Power Systems, Little Endian (POWER9)
  - 64-bit IBM Z (z14)

For more information, see Supported in-place upgrade paths for Red Hat Enterprise Linux .

For instructions on performing an in-place upgrade, see Upgrading from RHEL 9 to RHEL 10 .

In-place upgrade from RHEL 8 to RHEL 10

It is not possible to perform an in-place upgrade directly from RHEL 8 to RHEL 10. However, you can perform an in-place upgrade from RHEL 8 to RHEL 9 and then perform a second in-place upgrade to RHEL 10. For more information, see In-place upgrades over multiple RHEL major versions by using Leapp.

1.3. RED HAT CUSTOMER PORTAL LABS

Red Hat Customer Portal Labs is a set of tools in a section of the Customer Portal available at https://access.redhat.com/labs/. The applications in Red Hat Customer Portal Labs can help you improve performance, quickly troubleshoot issues, identify security problems, and quickly deploy and configure complex applications. Some of the most popular applications are:

- Registration Assistant
- Kickstart Generator
- Red Hat Product Certificates
- Red Hat CVE Checker

- Kernel Oops Analyzer
- Red Hat Satellite Upgrade Helper
- JVM Options Configuration Tool
- Load Balancer Configuration Tool
- Ceph Placement Groups (PGs) per Pool Calculator
- Yum Repository Configuration Helper
- Red Hat Out of Memory Analyzer
- Postfix Configuration Helper
- System Unit Generator
- Rsyslog Configuration Helper

1.4. ADDITIONAL RESOURCES

The **Red Hat Insights** service, which enables you to proactively identify, examine, and resolve known technical issues, is available with all RHEL subscriptions. For instructions on how to install the Red Hat Insights client and register your system to the service, see the Red Hat Insights Get Started page.

NOTE

Public release notes include links to access the original tracking tickets, but private release notes are not viewable so do not include links.[1]

[1] Public release notes include links to access the original tracking tickets, but private release notes are not viewable so do not include links.

CHAPTER 2. ARCHITECTURES

Red Hat Enterprise Linux 10.0 is distributed with the kernel version 6.12.0, which provides support for the following architectures at the minimum required version (stated in parentheses):

- AMD and Intel 64-bit architectures (x86-64-v3)
- The 64-bit ARM architecture (ARMv8.0-A)
- IBM Power Systems, Little Endian
- 64-bit IBM Z

Make sure you purchase the appropriate subscription for each architecture.

CHAPTER 3. DISTRIBUTION OF CONTENT IN RHEL 10

3.1. INSTALLATION

Red Hat Enterprise Linux 10 is installed using ISO images. Two types of ISO image are available for the AMD64, Intel 64-bit, 64-bit ARM, IBM Power Systems, and IBM Z architectures:

- Installation ISO: A full installation image that contains the BaseOS and AppStream repositories and allows you to complete the installation without additional repositories. On the Product Downloads page, the **Installation ISO** is referred to as **Binary DVD**.

NOTE

The Installation ISO image is in multiple GB size, and as a result, it might not fit on optical media formats. A USB key or USB hard drive is recommended when using the Installation ISO image to create bootable installation media. You can also use the Insights image Builder tool to create customized RHEL images. For more information about Insights image Builder, see the *Composing a customized RHEL system image* document.

- Boot ISO: A minimal boot ISO image that is used to boot into the installation program. This option requires access to the BaseOS and AppStream repositories to install software packages. The repositories are part of the Installation ISO image. You can also register to Red Hat CDN or Satellite during the installation to use the latest BaseOS and AppStream content from Red Hat CDN or Satellite.

3.2. REPOSITORIES

Red Hat Enterprise Linux 10 is distributed through two main repositories:

- BaseOS
- AppStream

Both repositories are required for a basic RHEL installation, and are available with all RHEL subscriptions.

Content in the BaseOS repository is intended to provide the core set of the underlying operating system functionality that provides the foundation for all installations. This content is available in the RPM format and is subject to support terms similar to those in previous releases of RHEL.

Content in the AppStream repository includes additional user-space applications, runtime languages, and databases in support of the varied workloads and use cases.

In addition, the CodeReady Linux Builder repository is available with all RHEL subscriptions. It provides additional packages for use by developers. Packages included in the CodeReady Linux Builder repository are unsupported.

For more information about RHEL 10 repositories and the packages they provide, see the Package manifest.

3.3. APPLICATION STREAMS

Multiple versions of user-space components are delivered as Application Streams and updated more frequently than the core operating system packages. This provides greater flexibility to customize RHEL without impacting the underlying stability of the platform or specific deployments.

Application Streams are available in the following formats:

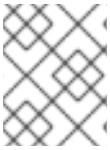
- RPM format
- Software Collections
- Flatpaks

NOTE

In previous RHEL major versions, some Application Streams were available as modules as an extension to the RPM format. In RHEL 10, Red Hat does not intend to provide any Application Streams that use modularity as the packaging technology and, therefore, no modular content is being distributed with RHEL 10.

Each Application Stream component has a given life cycle, either the same as RHEL 10 or shorter.

RHEL 10 improves the Application Streams experience by providing initial Application Stream versions that can be installed as RPM packages using the **dnf install** command.



NOTE

Certain initial Application Streams in the RPM format have a shorter life cycle than Red Hat Enterprise Linux 10.

Always determine what version of an Application Stream you want to install.

Content that needs rapid updating, such as alternate compilers and container tools, is available in Rolling Streams that will not provide alternative versions in parallel.

CHAPTER 4. IMPORTANT CHANGES TO EXTERNAL KERNEL PARAMETERS

This chapter provides system administrators with a summary of significant changes in the kernel distributed with Red Hat Enterprise Linux 10.0. These changes could include, for example, added or updated **proc** entries, **sysctl**, and **sysfs** default values, boot parameters, kernel configuration options, or any noticeable behavior changes.

New kernel parameters

accept_memory=

[MM]

Values:

lazy (default)

By default, unaccepted memory is accepted lazily to avoid prolonged boot times. The lazy option adds some runtime overhead until all memory is eventually accepted. In most cases, the overhead is negligible.

eager

For some workloads or for debugging purposes, you can use **accept_memory=eager** to accept all memory at once during boot.

arm64.nomops

[ARM64]

Unconditionally disable Memory Copy and Memory Set instructions support.

cgroup_favordynmods=

[KNL]

Enable or disable **favordynmods**.

Values:

- **true**
- **false**

Defaults to the value of **CONFIG_CGROUP_FAVOR_DYNMODS**.

early_page_ext

[KNL]

Enforces **page_ext** initialization to earlier stages to cover more early boot allocations.

Note that as side effect, some optimizations might be disabled to achieve that: for example, parallelized memory initialization is disabled. Therefore, the boot process might take longer, especially on systems with much memory.

Available with **CONFIG_PAGE_EXTENSION=y**.

fw_devlink.sync_state=

[KNL]

When all devices that could probe have finished probing, this parameter controls what to do with devices that have not yet received their **sync_state()** calls.

Values:

strict (default)

Continue waiting on consumers to probe successfully.

timeout

Give up waiting on consumers and call **sync_state()** on any devices that have not yet received their **sync_state()** calls after **deferred_probe_timeout** has expired or by **late_initcall()** if **CONFIG_MODULES** is **false**.

ia32_emulation=

[X86-64]

Values:

true

Allows loading 32-bit programs and executing 32-bit syscalls, essentially overriding **IA32_EMULATION_DEFAULT_DISABLED** at boot time.

false

Unconditionally disables IA32 emulation.

kunit.enable=

[KUNIT]

Enable executing KUnit tests. Requires **CONFIG_KUNIT** to be set to be fully enabled.

You can override the default value by using **KUNIT_DEFAULT_ENABLED**.

The default is 1 (enabled).

mtrr=debug

[X86]

Enable printing debug information related to MTRR registers at boot time.

rcupdate.rcu_cpu_stall_cputime=

[KNL]

Provide statistics on the CPU time and count of interrupts and tasks during the sampling period. For multiple continuous RCU stalls, all sampling periods begin at half of the first RCU stall timeout.

rcupdate.rcu_exp_stall_task_details=

[KNL]

Print stack dumps of any tasks blocking the current expedited RCU grace period during an expedited RCU CPU stall warning.

spec_rstack_overflow=

[X86]

Control RAS overflow mitigation on AMD Zen CPUs.

Values:

off

Disable mitigation

microcode

Enable only microcode mitigation.

safe-ret (default)

Enable software-only safe RET mitigation.

ibpb

Enable mitigation by issuing IBPB on kernel entry.

ibpb-vmexit

Issue IBPB only on VMEXIT. This mitigation is specific to cloud environments.

workqueue.unbound_cpus=

[KNL,SMP]

Specify to constrain one or some CPUs to use in unbound workqueues.

Value: A list of CPUs.

By default, all online CPUs are available for unbound workqueues.

Updated kernel parameters

amd_iommu=

[HW, X86-64]

Pass parameters to the AMD IOMMU driver in the system.

Values:

fullflush

Deprecated, equivalent to **iommu.strict=1**.

off

Do not initialize any AMD IOMMU found in the system.

force_isolation

Force device isolation for all devices. The IOMMU driver is not allowed anymore to lift isolation requirements as needed. This option does not override **iommu=pt**.

force_enable

Force enable the IOMMU on platforms known to be buggy with IOMMU enabled. Use this option with care.

New: pgtbl_v1 (default)

Use version 1 page table for DMA-API.

New: pgtbl_v2

Use version 2 page table for DMA-API.

New: irtcachedis

Disable Interrupt Remapping Table (IRT) caching.

nosmt

[KNL, **PPC**, S390]

Disable symmetric multithreading (SMT). Equivalent to **smt=1**.

[KNL, X86, **PPC**]

Disable symmetric multithreading (SMT).

nosmt=force

Force disable SMT. Cannot be undone by using the **sysfs** control file.

page_reporting.page_reporting_order=

[KNL]

Minimal page reporting order.

Value: integer.

Adjust the minimal page reporting order.

New: The page reporting is disabled when it exceeds **MAX_ORDER**.

tsc=

Disable clocksource stability checks for TSC.

Values:

[x86] reliable

Mark tsc clocksource as reliable. This disables clocksource verification at runtime, and the stability checks done at bootup. Used to enable high-resolution timer mode on older hardware, and in virtualized environment.

[x86] noirqtime

Do not use TSC to do **irq** accounting. Used to run time disable **IRQ_TIME_ACCOUNTING** on any platforms where RDTSC is slow and this accounting might add overhead.

[x86] unstable

Mark the TSC clocksource as unstable. This marks the TSC unconditionally unstable at bootup and avoids any further wobbles once the TSC watchdog notices.

[x86] nowatchdog

Disable clocksource watchdog. Used in situations with strict latency requirements, where interruptions from clocksource watchdog are not acceptable.

[x86] recalibrate

Force recalibration against a HW timer (HPET or PM timer) on systems whose TSC frequency was obtained from hardware or firmware by using either an MSR or CPUID(0x15). Warn if the difference is more than 500 ppm.

New: [x86]watchdog

Use TSC as the watchdog clocksource with which to check other HW timers (HPET or PM timer), but only on systems where TSC has been deemed trustworthy.
An earlier **tsc=nowatchdog** suppresses this. A later **tsc=nowatchdog** overrides this. A console message flags any such suppression or overriding.

usbcore.authorized_default=

[USB]

Default USB device authorization.

Values:

New: -1 (default)

Authorized (same as 1).

0

Not authorized.

1

Authorized.

2

Authorized if the device connects to an internal port.

Removed kernel parameters

- **cpu0_hotplug**
- **sysfs.deprecated**

New sysctl parameters

io_uring_group

Values:

1

A process must either be privileged (**CAP_SYS_ADMIN**) or be in the **io_uring_group** group to create an **io_uring** instance.

-1 (default)

Only processes with the **CAP_SYS_ADMIN** capability can create **io_uring** instances.

numa_balancing_promote_rate_limit_MBps

Too high promotion or demotion throughput between different memory types might hurt application latency. You can use this parameter to rate-limit the promotion throughput. The per-node maximum promotion throughput in MB/s is limited to be no more than the set value.

A rule is to set this to less than 1/10 of the PMEM node write bandwidth.

Updated sysctl parameters

io_uring_disabled

Prevents all processes from creating new **io_uring** instances. Enabling this shrinks the attack surface of the kernel.

Values:

New: 0

All processes can create **io_uring** instances as normal.

New: 1

io_uring creation is disabled for unprivileged processes not in the io_uring_group group. **io_uring_setup()** fails with **-EPERM**. Existing **io_uring** instances can still be used. See the documentation for **io_uring_group** for more information.

New: 2 (default)

io_uring creation is disabled for all processes. **io_uring_setup()** always fails with **-EPERM**. Existing **io_uring** instances can still be used.

CHAPTER 5. DEVICE DRIVERS

5.1. NEW DRIVERS

Table 5.1. Cryptographic drivers

Description	Name	Limited to architectures
IAA Compression Accelerator Crypto Driver	iaa_crypto	AMD and Intel 64-bit architectures
Intel® QuickAssist Technology - 0.6.0	intel_qat	AMD and Intel 64-bit architectures
Intel® QuickAssist Technology - 0.6.0	qat_4xxx	AMD and Intel 64-bit architectures
Intel® QuickAssist Technology - 0.6.0	qat_c3xxx	AMD and Intel 64-bit architectures
Intel® QuickAssist Technology - 0.6.0	qat_c3xxxvf	AMD and Intel 64-bit architectures
Intel® QuickAssist Technology - 0.6.0	qat_c62x	AMD and Intel 64-bit architectures
Intel® QuickAssist Technology - 0.6.0	qat_c62xvf	AMD and Intel 64-bit architectures
Intel® QuickAssist Technology - 0.6.0	qat_dh895xcc	AMD and Intel 64-bit architectures
Intel® QuickAssist Technology - 0.6.0	qat_dh895xccvf	AMD and Intel 64-bit architectures

Table 5.2. Network drivers

Description	Name	Limited to architectures
	bcm-phy-ptp	64-bit ARM architecture, IBM Power Systems, AMD and Intel 64-bit architectures
	mt7925-common	64-bit ARM architecture, AMD and Intel 64-bit architectures
	mt7925e	64-bit ARM architecture, AMD and Intel 64-bit architectures

Description	Name	Limited to architectures
	mt792x-lib	64-bit ARM architecture, AMD and Intel 64-bit architectures
CAN bus driver for Bosch M_CAN controller on PCI bus	m_can_pci	IBM Power Systems, AMD and Intel 64-bit architectures
CAN bus driver for Bosch M_CAN controller	m_can	IBM Power Systems, AMD and Intel 64-bit architectures
CAN driver for 8 devices USB2CAN interfaces	usb_8dev	IBM Power Systems, AMD and Intel 64-bit architectures
CAN driver for EMS Dr. Thomas Wuensche CAN/USB interfaces	ems_usb	IBM Power Systems, AMD and Intel 64-bit architectures
CAN driver for Kvaser CAN/USB devices	kvaser_usb	IBM Power Systems, AMD and Intel 64-bit architectures
CAN driver for PEAK-System USB adapters	peak_usb	IBM Power Systems, AMD and Intel 64-bit architectures
Intel® Infrastructure Data Path Function Linux Driver	idpf	64-bit ARM architecture, IBM Power Systems, AMD and Intel 64-bit architectures
Marvell 88Q2XXX 100/1000BASE-T1 Automotive Ethernet PHY driver	marvell-88q2xxx	64-bit ARM architecture, IBM Power Systems, AMD and Intel 64-bit architectures
Marvell Octeon EndPoint NIC Driver	octeon_ep	64-bit ARM architecture, IBM Power Systems, AMD and Intel 64-bit architectures
Microchip 251x/25625 CAN driver	mcp251x	AMD and Intel 64-bit architectures
Microchip MCP251xFD Family CAN controller driver	mcp251xfd	AMD and Intel 64-bit architectures
NXP imx8 DWMAC Specific Glue layer	dwmac-imx	64-bit ARM architecture
	bcm-phy-ptp	64-bit ARM architecture, IBM Power Systems, AMD and Intel 64-bit architectures

Description	Name	Limited to architectures
Realtek 802.11ax wireless 8852C driver	rtw89_8852c	64-bit ARM architecture, AMD and Intel 64-bit architectures
Realtek 802.11ax wireless 8852CE driver	rtw89_8852ce	64-bit ARM architecture, AMD and Intel 64-bit architectures
serial line CAN interface	slcan	IBM Power Systems, AMD and Intel 64-bit architectures
Socket-CAN driver for PEAK PCAN PCIe/M.2 FD family cards	peak_pciefd	IBM Power Systems, AMD and Intel 64-bit architectures
	bcm-phy-ptp	64-bit ARM architecture, IBM Power Systems, AMD and Intel 64-bit architectures
	mt7925-common	64-bit ARM architecture, AMD and Intel 64-bit architectures
	mt7925e	64-bit ARM architecture, AMD and Intel 64-bit architectures
	mt792x-lib	64-bit ARM architecture, AMD and Intel 64-bit architectures

Table 5.3. Platform drivers

Description	Name	Limited to architectures
AMD HSMP Platform Interface Driver - 2.0	amd_hsmp	AMD and Intel 64-bit architectures
AMD Platform Management Framework Driver	amd-pmf	AMD and Intel 64-bit architectures
Intel TPMI enumeration module	intel_vsec_tpmi	AMD and Intel 64-bit architectures
Intel TPMI SST Driver	isst_tpmi	AMD and Intel 64-bit architectures
Intel TPMI UFS Driver	intel-uncore-frequency-tpmi	AMD and Intel 64-bit architectures

Description	Name	Limited to architectures
Intel Uncore Frequency Common Module	intel-uncore-frequency-common	AMD and Intel 64-bit architectures
Intel Uncore Frequency Limits Driver	intel-uncore-frequency	AMD and Intel 64-bit architectures
Intel WMI Thunderbolt force power driver	intel-wmi-thunderbolt	AMD and Intel 64-bit architectures
Mellanox PMC driver	mlxbf-pmc	64-bit ARM architecture
	intel-hid	AMD and Intel 64-bit architectures
	isst_tpmi_core	AMD and Intel 64-bit architectures

Table 5.4. Graphics drivers and miscellaneous drivers

Description	Name	Limited to architectures
AMD XCP Platform Devices	amdxcp	64-bit ARM architecture, IBM Power Systems, AMD and Intel 64-bit architectures
DRM execution context	drm_exec	
Range suballocator helper	drm_suballoc_helper	64-bit ARM architecture, IBM Power Systems, AMD and Intel 64-bit architectures
	regmap-ram	64-bit ARM architecture, IBM Power Systems, AMD and Intel 64-bit architectures
	regmap-raw-ram	64-bit ARM architecture, IBM Power Systems, AMD and Intel 64-bit architectures
	regmap-ram	64-bit ARM architecture, IBM Power Systems, AMD and Intel 64-bit architectures

Description	Name	Limited to architectures
	regmap-raw-ram	64-bit ARM architecture, IBM Power Systems, AMD and Intel 64-bit architectures
	regmap-ram	64-bit ARM architecture, IBM Power Systems, AMD and Intel 64-bit architectures
	regmap-raw-ram	64-bit ARM architecture, IBM Power Systems, AMD and Intel 64-bit architectures
Arm FF-A interface driver	ffa-module	64-bit ARM architecture
NVIDIA BlueField-3 GPIO Driver	gpio-mlxbf3	64-bit ARM architecture
I/O Address Space Management for passthrough devices	iommufd	
CS42L43 Core Driver	cs42l43	AMD and Intel 64-bit architectures
CS42L43 SoundWire Driver	cs42l43-sdw	AMD and Intel 64-bit architectures
MEI GSC Proxy	mei_gsc_proxy	AMD and Intel 64-bit architectures
	pwrseq_emmc	64-bit ARM architecture
	pwrseq_simple	64-bit ARM architecture
SDHCI platform driver for Synopsys DWC MSHC	sdhci-of-dwcmshc	64-bit ARM architecture
	arm_cspmu_module	64-bit ARM architecture
NVIDIA pinctrl driver	pinctrl-mlxbf3	64-bit ARM architecture
NXP i.MX93 power domain driver	imx93-pd	64-bit ARM architecture
Intel RAPL TPMI Driver	intel_rapl_tpmi	AMD and Intel 64-bit architectures

Description	Name	Limited to architectures
Mellanox BlueField power driver	pwr-mlxbf	64-bit ARM architecture
NXP i.MX93 src driver	imx93-src	64-bit ARM architecture
Provide Trusted Security Module attestation reports via configfs	tsm	AMD and Intel 64-bit architectures

5.2. UPDATED DRIVERS

Table 5.5. Storage driver updates

Description	Name	Current version	Limited to architectures
Broadcom MegaRAID SAS Driver	megaraid_sas	07.727.03.00-rc1	64-bit ARM architecture, IBM Power Systems, AMD and Intel 64-bit architectures
Driver for Microchip Smart Family Controller	smartpqi	2.1.24-046	64-bit ARM architecture, IBM Power Systems, AMD and Intel 64-bit architectures
Emulex LightPulse Fibre Channel SCSI driver	lpfc	0:14.2.0.16	64-bit ARM architecture, IBM Power Systems, AMD and Intel 64-bit architectures
MPI3 Storage Controller Device Driver	mpi3mr	8.5.0.0.50	

CHAPTER 6. NEW FEATURES AND ENHANCEMENTS

This version adds the following major new features and enhancements.

6.1. INSTALLER AND IMAGE CREATION

bootc-image-builder now supports creating image mode disk images with advanced partitioning

With this enhancement, the **bootc-image-builder** tool gained more options for customizing partitioning. You can use the **bootc-image-builder** tool to create disk images of image-mode RHEL with custom mountpoints, including custom mount options, LVM-based partitions and LVM-based SWAP to, for example, change the size of the / and the /**boot** directories by using the `config.toml`As a consequence, you can create disk images with advanced partitioning layout.

Jira:RHELDOCS-18532[1]

RHEL 10 disk images will have predictable network interface names

The net.ifnames=0 will be removed from kernel arguments, causing all systems to use predictable network interface names. As a consequence, from RHEL 10.0 ongoing, disk images created with RHEL image builder will now have predictable network interface names. There are no plans for backporting this update to older RHEL versions. As a workaround for older versions, remove the kernel argument after the first boot and reboot the system. See Configuring kernel command-line parameters for more details.

Jira:RHELDOCS-18880[1]

New users created in Anaconda are administrators by default

Previously, while creating new users from the instllation program, the **Add administrative privileges to this user account** option in graphical installation was deselected. Starting RHEL 10, this option is selected by default. As a result, the newly created users will have administrative privileges in the system by default. You can deselect this option to remove the administrative privileges of the new users, if needed.

Jira:RHELDOCS-18425[1]

Added Kickstart support for CA certificates to enable encrypted DNS configuration during installation

Support for the **%certificate** section in the Kickstart file is added to enable the installation of CA certificates into the instllation program environment and the installed system. This simplifies the setup process and ensures that the encrypted DNS is operational after installation, reducing manual configuration and security gaps. The certificates are inlined in the Base64 ASCII format and imported through the **--dir** and **--filename** options. This enhancement facilitates encrypted DNS configuration as part of **Zero Trust Architecture** requirements. The encrypted DNS set up during installation ensures secure DNS resolution from the start, improving security and compliance in automated deployments.

Jira:RHEL-61434[1]

NVMe over Fabrics devices are now available in the RHEL installation program

You can now add NVMe over Fabrics devices to your RHEL installation to extend the benefits of NVMe storage beyond local devices, enabling the same high-performance, low-latency access over a network. In the RHEL installation program, you can select these devices under the NVMe Fabrics Devices section

while adding disks on the Installation Destination screen.

Jira:RHELDOCS-18819[1]

Remote Desktop Protocol (RDP) replaces VNC for graphical remote access

The protocol for graphical remote access has been replaced from VNC to remote desktop protocol (RDP), a more robust, and secure graphical remote access. It offers a reliable and encrypted connection, overcoming the limitations of VNC, which lacked encryption support and enforced password length restrictions.

You can now securely connect to graphical installation sessions. As part of this change, the **inst.vnc**, **inst.vncpassword**, and **inst.vncconnect** kernel boot options have been removed and the new options **inst.rdp**, **inst.rdp.password**, and **inst.rdp.username** have been introduced.

Jira:RHEL-38407

RHEL image builder supports [customization.installer] to inject Kickstart files into a built artifact

With this enhancement, you can use the new **[customization.installer]** blueprint customization field in RHEL image builder to add your own kickstart file. You can use the customizations for ISO instllation program such as **image installer** or **edge installer**, and can choose one of the following options:

- Set all values during the installation process.
- Enable the **unattended = true** field in Kickstart to get a fully unattended installation.
- Inject your own Kickstart by using the Kickstart field.

Depending on the fields that you specify, you can get an unattended installation, or the instllation program asks for the required fields. Alternatively, you can choose a fully unattended installation based on predefined configuration defaults. As a result, you gain more flexibility when building ISO images for bare metal deployments.

Jira:RHELDOCS-19583[1]

bootc-image-builder now supports creating image mode disk images with advanced partitioning

With this enhancement, the **bootc-image-builder** tool gained more options for customizing partitioning and creating disk images with advanced partitioning layout. You can use the **bootc-image-builder** tool to create disk images of image-mode RHEL with custom mountpoints, including custom mount options, LVM-based partitions and LVM-based SWAP to, for example, change the size of the / and the /**boot** directories by using the **config.toml**.

Jira:RHELDOCS-19291[1]

A new cockpit-image-builder plugin for RHEL image builder

For RHEL 10, RHEL image builder has a new user interface. As a result, you can benefit from new customization options, integrations with Insights services, and compatibility to share blueprints between RHEL image builder and Insights image builder.

Jira:RHELDOCS-20166[1]

RHEL disk images have the same default locale and time zone

Previously, RHEL disk images had inconsistent default locales and time zones sets. With this enhancement, RHEL disk images have the same locale and time zones by default, that is, the default locale is **C.UTF-8** is, and the default time zone is UTC.

Jira:RHELDOCS-20168[1]

Ability to build RHEL images on AWS with UEFI by default

Previously, you could only boot RHEL images on AWS by using legacy BIOS boot. With this enhancement, you can now boot RHEL images on AWS with UEFI by default. As a result, Secure Boot now improves security of your workloads.

Jira:RHELDOCS-20169[1]

RHEL 10 disk images no longer have a separate /boot partition

RHEL 10 Public disk images, such as AWS images, or KVM images, for example, do not have a separate **/boot** partition. In RHEL images, the **/boot/** partition removal targets confidential computing.

This change prevents the **/boot** partition from exceeding disk space, which was often the case when **/boot** was on a separate partition. As a result, operational failures are less likely to occur.

Jira:RHELDOCS-18902[1]

RHEL image builder now supports blueprint customization to creating disk images with advanced partitioning

With this enhancement, RHEL image builder gained more options for customizing partitioning and thus creating disk images with advanced partitioning layout. You can customize your blueprint with custom mount options, LVM-based partitions and LVM-based SWAP to, for example, change the size of the / and the **/boot** directories in the blueprint file.

Jira:RHELDOCS-19106[1]

6.2. SECURITY

keylime-agent-rust provided in version 0.2.5

The **keylime-agent-rust** package, which contains the Keylime agent, is provided in version 0.2.5 in RHEL 10. This version offers important enhancements and bug fixes, most importantly the following:

- Added support for Initial Device Identity (IDevID) and Initial Attestation Key (IAK) for device identity. The following configuration options have been added:

  enable_iak_idevid

  (default: **false**) Enables the use of IDevID and IAK certificates to identify the device.

  iak_idevid_template

  (default: **detect**) Specifies the template that sets the algorithms to be used for IDevID and IAK (defined in TPM 2.0 Keys for Identity and Attestation, section 7.3.4). The **detect** keyword sets the template according to the algorithms used in the configured certificates.

  iak_idevid_name_alg

(default: **sha256**) Specifies the digest algorithm used in IDevID and IAK. Used only if the **iak_idevid_template** option is not set as **detect**.

iak_idevid_asymmetric_alg

(default: **rsa**) Specifies the signing algorithm used in IDevID and IAK. Used only if the **iak_idevid_template** option is not set as **detect**.

iak_cert

(default: **default**) Specifies the path to the file that contains the X509 IAK certificate. The default path is **/var/lib/keylime/iak-cert.crt**.

idevid_cert

(default: **default**) Specifies the path to the file that contains the X509 IDevID certificate. The default path is **/var/lib/keylime/idevid-cert.crt**.

- Configurable IMA and measured boot event log locations are supported by using the new **ima_ml_path** and **measuredboot_ml_path** configuration options.
- Local DNS name, local IP, and configured contact IP are included as part of the Subject Alternative Name of the generated self-signed X509 certificate.
- IPv6 addresses with or without brackets are supported in the **registrar_ip** configuration option.
- Hexadecimal encoded values are supported in the **tpm_ownerpassword** configuration option.
- TLS 1.3 is enabled in connections to the agent.

Jira:RHEL-38409

libreswan provided in version 4.15

The **libreswan** packages are provided in version 4.15 in RHEL 10. This version offers substantial improvements over the previous version 4.12 that was provided in previous releases:

- Removed a dependency on **libxz** through **libsystemd**.
- In IKEv1, default proposals have been set to **aes-sha1** for Encapsulating Security Payload (ESP) and **sha1** for Authentication Header (AH).
- IKEv1 rejects ESP proposals that combine Authenticated Encryption with Associated Data (AEAD) and non-empty INTEG.
- IKEv1 rejects exchange when a connection has no proposals.
- IKEv1 has a more limited default cryptosuite:

```
IKE={AES_CBC,3DES_CBC}-{HMAC_SHA2_256,HMAC_SHA2_512HMAC_SHA1}-
{MODP2048,MODP1536,DH19,DH31}
ESP={AES_CBC,3DES_CBC}-
{HMAC_SHA1_96,HMAC_SHA2_512_256,HMAC_SHA2_256_128}-
{AES_GCM_16_128,AES_GCM_16_256}
AH=HMAC_SHA1_96+HMAC_SHA2_512_256+HMAC_SHA2_256_128
```

- Failures of the **libcap-ng** library are no longer unrecoverable.
- TFC padding is set for AEAD algorithms in the **pluto** utility.

Jira:RHEL-52935[1]

Libreswan is faster in adding large numbers of connections

Before this update, it took around 30 minutes for the **libreswan** IPsec implementation to add 1,000 connections in certain circumstances. The latest version of **libreswan** skips the **getservbyname()** function on numbered connections, and offloading validation of existing connections to the **pluto** daemon substantially reduces the loading times for large configuration files. As a result, the time to add 1,000 connections should be about 50 seconds instead of 30 minutes on the same configuration.

Jira:RHEL-74850[1]

GnuTLS provided in version 3.8.9

RHEL 10 provides the **gnutls** packages in version 3.8.9. Among other improvements, this version contains the following security-related changes that are not compatible with earlier versions:

- Certificate compression in TLS is supported (RFC 8879).
- Optimal Asymmetric Encryption Padding scheme (RSA-OAEP) is supported (RFC 8017).
- API for incremental calculation of SHAKE hashes of arbitrary length across multiple calls has been added.
- RSA encryption and decryption with PKCS #1 v1.5 padding is deprecated and disallowed by default.
- In FIPS mode, **gnutls** now defaults to exporting PKCS #12 files with Password-Based Message Authentication Code 1 (PBMAC1) as defined in RFC 9579. If you need interoperability with systems running in FIPS mode, use PBMAC1 explicitly.
- GnuTLS now checks all records in an Online Certificate Status Protocol (OCSP) response. Before this update, when multiple records were provided in a single OCSP response, only the first record was checked. This version of GnuTLS examines all records until the server certificate matches.
- The minimum RSA key size for verification to be approved in FIPS mode has been increased to 2048 bits.

Jira:RHEL-69524[1]

OpenSSH provided in version 9.9

RHEL 10 provides OpenSSH in version 9.9, which introduces many fixes and improvements over OpenSSH 8.7 which was provided in RHEL 9. For the complete list of changes, see the **openssh-9.9p1/ChangeLog** file. The most important changes are as follows:

- A system for restricting forwarding and use of keys that were added to the **ssh-agent** program has been added to **ssh**, **sshd**, **ssh-add**, and **ssh-agent** programs.
- Improvements to the use of the FIDO standard:
  - The **verify-required** certificate option has been added to **ssh-keygen**.
  - Fixes to FIDO key handling reduce unnecessary PIN prompts for keys that support intrinsic user verification.

    - A check for existing matching credentials in the **ssh-keygen** program prompts the user before overwriting the credential.

- New **EnableEscapeCommandline** option in the **ssh_config** configuration file enables the command line option in the **EscapeChar** menu for interactive sessions.
- New **ChannelTimeout** keyword specifies whether and how quickly the **sshd** daemon should close inactive channels.
- The **ssh-keygen** utility generates Ed25519 keys by default except in FIPS mode, where the default is RSA.
- The **ssh** client performs keystroke timing obfuscation by sending interactive traffic at fixed intervals, every 20 ms by default, when only a small amount of data is being sent. It also sends fake keystrokes for a random interval after the last real keystroke, defined by the **ObscureKeystrokeTiming** keyword.
- Support for DSA keys has been removed.
- The **pam-ssh-agent** subcomponent has been removed.
- The **ssh-keysign** tool is now in a separate subpackage.
- With the new **ChannelTimeout** type, **ssh** and **sshd** close all open channels if all channels lack traffic for a specified interval. This is in addition to the existing per-channel timeouts.
- The **sshd** server blocks client addresses that repeatedly fail authentication, repeatedly connect without ever completing authentication, or that crash the server.
- The **sshd** server penalizes client addresses that do not successfully complete authentication. The penalties are controlled by the new **PerSourcePenalties** keyword in **sshd_config**.
- The **sshd** server is split into a listener binary **sshd** and a per-session binary **sshd-session**. This reduces the listener binary size that does not need to support the SSH protocol. This also removes support for disabling privilege separation and disabling re-execution of **sshd**
- In portable OpenSSH, **sshd** no longer uses **argv[0]** as the PAM service name. You can select the service name at runtime with the new **PAMServiceName** directive in the **sshd_config** file. This defaults to "sshd".
- The **HostkeyAlgorithms** keyword allows **ssh** to disable implicit fallback from certificate host key to plain host keys.
- The components have been hardened in general and work better with the PKCS #11 standard.
- As a Technology Preview, OpenSSH supports post-quantum cryptography (PQC).

Jira:RHEL-60564

Added custom configuration for pkcs11-provider

The **pkcs11-provider** allows direct access to hardware tokens by using **pkcs11** URIs from OpenSSL programs. Upon installation, the **pkcs11-provider** is automatically enabled and loads tokens detected by the **pcscd** daemon by using the **p11-kit** driver by default. As a result, you can use tokens available to the system if you provide a key URI by using the **pkcs11** URI specification to an application that supports

that format by installing the package without the need to further change OpenSSL configuration. Uninstalling the package also removes the OpenSSL configuration snippet, which prevents errors when OpenSSL parses the configuration files.

Jira:RHEL-29672

File context equivalency set to /var/run = /run in the SELinux policy

The previous **/run = /var/run** file context equivalency is now inverted to **/var/run = /run** and the SELinux policy sources have been updated accordingly. The equivalency has been inverted to match the actual filesystem state and to prevent some userspace tools from reporting an error. This change should not be visible from the user or administrator perspective. If you have any custom modules that contain file specification for files in **/var/run**, change them to **/run**.

Jira:RHEL-36094[1]

OpenSSL uses pkcs11-provider for hardware tokens

Because OpenSSL 3.0 deprecated engines and replaced them with providers, RHEL 10 replaces the **openssl-pkcs11** engine with the **pkcs11-provider**. This allows OpenSSL to use hardware tokens in applications such as **apache** HTTPD, **libssh**, **bind**, and other applications that are linked with OpenSSL and use asymmetric private keys stored in an HSM, smart card or other tokens with a PKCS #11 driver available.

Jira:RHEL-40124

New capability.conf(5) man page

The **capability.conf(5)** man page has been added. It provides descriptions for the **capability.conf** configuration file and the **pam_cap.so** module arguments.

Jira:RHEL-31988

libkcapi provided in version 1.5.0

In RHEL 10.0, the **libkcapi** packages are provided in upstream version 1.5.0. This version provides various bug fixes, optimizations and enhancements, most notably:

- The **sha*** applications have been removed and replaced with a single application called **kcapi-hasher**. Symlinks to **kcapi-hasher** with equivalent names as the original **sha*** applications have been added into the **bin** and **libexec** directories. This change does not cause any known regressions.
- The **sha3sum** command, which prints checksums of files that use SHA-3, has been added.
- The **kcapi_md_sha3_*** wrapper APIs have been added.

Jira:RHEL-50457[1]

Stricter SSH host key permissions have been restored

The necessary host key permissions have been changed from the previous less strict value of **0640** to **0600**, which is also the value used upstream. The **ssh_keys** group, which previously owned all SSH keys, has also been removed. Therefore, the **ssh-keysign** utility uses the SUID bit instead of the SGID bit.

Jira:RHEL-59102[1]

libssh provided in version 0.11.1

The **libssh** SSH library is provided in version 0.11.1, which brings new functionalities, most importantly the following:

- Better asynchronous SFTP IO
- PKCS #11 provider support for OpenSSL 3.0
- Testing for GSSAPI authentication
- Proxy jump

Jira:RHEL-64319

p11-kit provided in version 0.25.5

The **p11-kit** packages are provided in version 0.25.5 in RHEL 10. This version provides enhancements and fixes over the previous version, most importantly, the following:

- Support for recursive attributes has been added to the **p11-kit** RPC protocol.
- A function to check run time version of the library has been added.
- Version information is no longer accessible through macros.
- With the new **--id** option, you can assign an ID to key pairs generated with the **generate-keypair** command or imported with the **import-object** command.
- With the new **--provider** option, you can specify a PKCS #11 module when using **p11-kit** commands.
- Fixed a bug in **p11-kit** where the EdDSA mechanism was not recognized in **generate-keypair**.
- **p11-kit** falls back to the **C_GetFunctionList** function when the **C_GetInterface** function is not supported.

Jira:RHEL-46898[1]

pkeyutil now supports encapsulation and decapsulation

The **pkeyutil** OpenSSL subcommand supports performing encapsulation and decapsulation cryptographic operations. The new post-quantum cryptographic (PQC) algorithm ML-KEM (FIPS 203) permits only encapsulation and decapsulation operations, and you can now use algorithms such as RSASVE and ML-KEM through **pkeyutil**.

Jira:RHEL-54156

GnuTLS can use certificate compression

GnuTLS compresses client and server certificates with the **zlib**, **brotli** or **zstd** compression method according to RFC 8879 if both client and server support and enable it. This method reduces data usage, and should otherwise be unnoticeable to users.

Jira:RHEL-42514[1]

New no-atexit option in OpenSSL

OpenSSL is now built with the **no-atexit** option, so that the **OPENSSL_cleanup** function is no longer registered as an **atexit** handler. Using this option might cause the **valgrind** debugging tool to report one-time memory leaks of the resources allocated on OpenSSL startup.

Jira:RHEL-40408

setools provided in version 4.5.0

The **setools** packages are provided in version 4.5.0 in RHEL 10. This version provides bug fixes and enhancements, most notably the following:

- Graphical results for information flow analysis and domain transition analysis have been added to the **apol**, **sedta**, and **seinfoflow** tools.
- Tooltips and detail popups in **apol** have been added to help cross-referencing query and analyzing results along with context-sensitive help.

Jira:RHEL-29967

RHEL 10 provides NSS in version 3.101

The NSS cryptographic toolkit packages are provided in version 3.101 in RHEL 10, which provides many bug fixes and enhancements. The most notable changes are the following:

- DTLS 1.3 protocol is now supported (RFC 9147).
- PBMAC1 support has been added to PKCS #12 (RFC 9579).
- Experimental support for X25519Kyber768Draft00 hybrid post-quantum key agreement has been added (**draft-tls-westerbaan-xyber768d00**). It will be removed in a future release.
- **lib::pkix** is the default validator in RHEL 10.
- RSA certificates with keys shorter than 2048 bits stop working in SSL servers, in accordance with the system-wide cryptographic policy.

Jira:RHEL-46839

OpenSSL can create FIPS-compliant PKCS #12 files

The OpenSSL secure communication suite has been updated and can now create PKCS #12 files in accordance with the RFC 9579 document.

Jira:RHEL-36659

The DEFAULT cryptographic policy uses additional scopes

The **crypto-policies** package now offers additional scopes **@pkcs12**, **@pkcs12-legacy**, **@smime**, and **@smime-legacy**, and uses them in the **DEFAULT** system-wide cryptographic policy. The selection of cryptographic algorithms used for PKCS #12 and S/MIME when network security services (NSS) is the underlying cryptographic library now follows system-wide cryptographic policies. Therefore, you can more easily select algorithms with higher granularity by using custom policies and subpolicies. The scopes use the following ciphers, hashes, and key exchanges:

```
cipher@pkcs12 = AES-256-CBC AES-128-CBC
cipher@pkcs12-import = 3DES-CBC+ RC2-CBC+
cipher@smime = AES-256-CBC AES-128-CBC 3DES-CBC
cipher@smime-import = RC2-CBC+
```

```
hash@{pkcs12,smime} = SHA2-256 SHA2-384 SHA2-512 SHA3-256 SHA3-384 SHA3-512 \
 SHA2-224 SHA3-224
hash@{pkcs12-import,smime} = SHA1+
key_exchange@smime = RSA DH ECDH
```

The **LEGACY** cryptographic policy uses a less strict selection of ciphers, hashes, and key exchanges than the **DEFAULT** policy, whereas the **FUTURE** policy is stricter. As a result, you can customize the algorithms used in NSS for importing and exporting PKCS #12 files and S/MIME encryption and decryption. NSS is currently the only cryptographic library linked to the newly offered scopes.

Jira:RHEL-50655

OpenSSH in FIPS mode generates RSA keys by default

In previous versions, the **ssh-keygen** utility in OpenSSH generated RSA keys by default. In the versions provided with RHEL 10, **ssh-keygen** generates ed25519 keys by default in non-FIPS mode and RSA keys by default in FIPS mode.

Jira:RHEL-37324

NSS creates FIPS-compliant PKCS #12 in FIPS mode

PKCS #12 uses an ad hoc mechanism for integrity checks. Since the publication of PKCS #12 version 1.1, more rigorous methods of integrity checks have been created in PKCS #5 Version 2.0: the password-based message authentication code 1 (PBMAC1). This update adds PBMAC1 support in PKCS #12 files to Network Security Services (NSS) in accordance with the RFC 9579 document. As a result, NSS can now read any **.p12** file that uses RFC 9579 and can generate RFC-9579-compliant message authentication codes (MAC) when requested by the user. For compatibility, NSS generates old MACs by default when not in FIPS mode. For more information on generating new MACs, see the **pk12util(1)** man page on your system.

Jira:RHEL-39732

OpenSC provided in version 0.26.1

RHEL 10 provides the **opensc** packages in the upstream version 0.26.1. The most notable enhancements and bug fixes are:

- Additional fixes for removing the time side-channel leakage related to the RSA PKCS #1 v1.5 padding removal after decryption
- Unified OpenSSL logging
- Support for the HKDF, RSA OEAP encryption, AES GCM, and AES GMAC mechanisms in the **pkcs11-tool** utility
- Fixes for CVEs targeting uninitialized memory problems: CVE-2024-45615, CVE-2024-45616, CVE-2024-45617, CVE-2024-45618, CVE-2024-45619, and CVE-2024-45620
- A fix of allocations of aligned memory that caused crashes in the Chromium web browser
- A fix of reading certificates in the TeleSec Chipcard Operating System (TCOS) card driver

Jira:RHEL-71523

OpenSC packages split into opensc and opensc-lib

In RHEL 10, the **opensc** packages have been split into the **opensc** and **opensc-lib** subpackages to enable support for smart cards in Flatpak applications.

Jira:RHEL-73314

New package: tpm2-openssl

RHEL 10 includes the new **tpm2-openssl** package, which contains the TPM2 provider for the OpenSSL TLS toolkit. The TPM2 provider enables using cryptographic keys from a Trusted Platform Module (TPM) 2.0 chip through the OpenSSL API.

Jira:RHEL-30799[1]

Rule-based filtering and forwarding of Audit events

With the new **audisp-filter** plugin, you can suppress specific Audit events based on custom **ausearch** expressions in a flexible way, which should reduce unnecessary output to downstream plugins.

This plugin acts as a bridge between Audit and other plugins. It filters out certain Audit events and forwards only those events that correspond to the rules specified in the configuration file.

As a result, you can selectively filter Audit events by using allowlist or blocklist modes. Each plugin that uses the **audisp-filter** can define its own configuration file that contains matching rules. One common use case is to exclude noisy or irrelevant Audit events and forward only significant events to the syslog plugin. This allows the filtered events to be logged by syslog, making Audit logs more manageable.

Jira:RHEL-5199

Additional services confined in the SELinux policy

This update adds additional rules to the SELinux policy that confine the following **systemd** services:

- **iio-sensor-proxy**
- **samba-bgqd**
- **tlshd**
- **gnome-remote-desktop**
- **pcm-sensor-server**

As a result, these services no longer run with the **unconfined_service_t** SELinux label, which violated the CIS Server Level 2 benchmark "Ensure No Daemons are Unconfined by SELinux" rule, and run successfully in SELinux enforcing mode.

Jira:RHEL-62355

The selinux-policy Git repository for CentOS Stream 10 is now publicly accessible

CentOS Stream contributors now can participate in the development of the SELinux policy by contributing to the **c10s** branch of the **fedora-selinux/selinux-policy** Git repository. These contributions can then be used to improve the SELinux policy of RHEL 10.

Jira:RHEL-33844

setroubleshoot provided in version 3.3.35

The **setroubleshoot** packages are provided in version 3.3.35 in RHEL 10. This version provides various fixes and enhancements, most importantly the following:

- Backtrace on CoreOS has been fixed.
- Broken AppStream metadata have been updated.
- The paths of used icons have been fixed to recently updated paths.

Jira:RHEL-68957

Rules for additional libvirt services added to the SELinux policy

The following SELinux types related to the **libvirt** services have been added to the SELinux policy:

- **virt_dbus_t**
- **virt_hook_unconfined_t**
- **virt_qmf_t**
- **virtinterfaced_t**
- **virtnetworkd_t**
- **virtnodedevd_t**
- **virtnwfilterd_t**
- **virtproxyd_t**
- **virtqemud_t**
- **virtsecretd_t**
- **virtstoraged_t**
- **virtvboxd_t**
- **virtvzd_t**
- **virtxend_t**

Jira:RHEL-46893

SELinux policy modules related to EPEL packages moved to selinux-policy-epel

The SELinux policy modules related only to packages contained in the Extra Packages for Enterprise Linux (EPEL) repository and not to any RHEL package were moved from the **selinux-policy** package to the new **selinux-policy-epel** package. As a result, **selinux-policy** is smaller, and the system performs operations such as rebuilding and loading the SELinux policy faster.

Jira:RHEL-73505

SELinux userspace provided in version 3.8

RHEL 10 contains the SELinux user-space components in version 3.8. This version introduces enhancements and fixes over the previous version, most importantly, the following:

- A new **audit2allow -C** option has been added to the CIL output mode.
- The **semanage** utility allows modifying records on **add**.
- The **semanage** utility no longer sorts local **fcontext** definitions.
- The **checkpolicy** program supports the CIDR notation for **nodecon** statements.
- The SELinux **sandbox** utility supports the Wayland display protocol.
- File context and ownership in the policy store are preserved during SELinux policy rebuild.
- The format of the binary **file_contexts.bin** file has been changed, and files that use the old format are ignored. The new format is optimized and not architecture-dependent. You can create the binary **file_contexts.bin** file in the new format by rebuilding the SELinux policy.
- The performance of the **selabel_lookup** library call has been improved significantly.

Jira:RHEL-69451

Rsyslog is provided in version 8.2412.0

The **rsyslog** packages are provided in version 8.2412.0 in RHEL 10.0. Among other fixes and enhancements, you can bind a ruleset to the **imjournal** module. With this optimization, log messages can be filtered and processed at the input stage, which reduces the load on the main message queue. By minimizing resource utilization, this feature ensures smoother handling of high-volume logs.

Jira:RHEL-70110[1]

Clevis provided in version 21 with support for PKCS #11

RHEL 10 provides the **clevis** packages in version 21. This version contains many enhancements and bug fixes, notably:

- Added the **clevis-pin-pkcs11** subpackage which provides the **pkcs11** pin for unlocking LUKS-encrypted volumes using a PKCS #11 device (smart card).
- Added two checks to the **clevis-udisks2** subpackage.
- Added a fix that prevents "Address in use" errors.

Jira:RHEL-60113

jose provided in version 14

The **jose** package is provided in version 14 in RHEL 10. The **jose** utility is a C-language implementation of the Javascript Object Signing and Encryption (JOSE) standards. The most important enhancements and fixes include the following:

- Improved bound checks for the **len** function for the **oct** JWK Type in OpenSSL, as a fix to an error reported by the SAST (Static Application Security Testing) process.
- The protected JSON Web Encryption (JWE) headers no longer contain **zip**.

- The **jose** utility avoids potential denial of service (DoS) attacks by using high decompression chunks.

Jira:RHEL-38084

Keylime provided in version 7.12

RHEL 10 provides Keylime in version 7.12, which provides important fixes and enhancements, most importantly:

- The new **keylime-policy** tool integrates all management tasks of Keylime runtime policies and measured boot policies and improves the performance of generating policies.
- The **verifier** and **tenant** Keylime components no longer require payloads for the **agent** component.

Jira:RHEL-75794

Libreswan provided in version 5.2

In RHEL 10, Libreswan is provided in upstream version 5.2. This version provides many bug fixes and enhancements, most importantly the following:

- Duplicate **--ctlsocket** option for the **whack** command is fixed (RHEL-75605).
- An expectation failure with crossing streams is fixed (RHEL-73236).
- Parsing protoport configuration has been optimized (RHEL-74850).
- Incorrect outputs for the **ipsec showhostkey** command are fixed (RHEL-75975).
- Crashes on executing **ipsec --rereadsecrets** are fixed (RHEL-69403).
- The **keyingtries** and **dpd*** options are ignored.
- The **ipsec-interface-managed=no** option for network namespaces has been introduced.
- Linux-specific updates:
  - Added support for packet offload counters in Linux kernel 6.7 and above.
  - Implemented IP-TFS (IP Traffic Flow Security) support according to RFC 9347.
  - Ensured compatibility with Linux kernel 6.10+ by setting the replay window to 0 on outbound SAs.
  - Fixed issues related to the **nopmtudisc** setting on inbound security associations (SA). IKEv2 enhancements:
  - Introduced support for RFC 5723 IKE Session Resumption, enabling session resumption without re-authentication.
  - Added support for **draft-ietf-ipsecme-ikev2-qr-alt-04**, enhancing key exchange mechanisms.
  - Implemented PPK (Post-quantum Pre-shared Key) in the INTERMEDIATE exchange to improve security.

NOTE

Peer authentication that uses PKCS #1 1.5 RSA with SHA-1 requires explicit allowing of SHA-1 signatures in NSS by using a custom cryptographic policies subpolicy. This is necessary when **authby=rsa-sha1** is configured or in a default configuration when an authenticated peer does not support RFC 7427.

Jira:RHEL-81045

ssh now provides a link with additional details about SSH login error messages

In case of an early error, the **ssh** command-line tool provides a link to the Red Hat Customer Portal page that contains additional details about common error messages and steps for resolving them. This helps troubleshoot SSH login problems when you use interactive mode.

Jira:RHEL-62718[1]

nettle provided in version 3.10.1

RHEL 10 contains the **nettle** library package in version 3.10.1. This version provides various bug fixes, optimizations and enhancements, most notably:

- SHA-256 hashing, AES-GCM encryption, and AES decryption in general have gained optimizations on 64-bit PowerPC.
- DRBG-CTR-AES256, a new deterministic random bit generator, has been added.
- SHAKE-128, an arbitrary length hash function of the SHA-3 family, has been added.
- Support for the RSA-OAEP scheme has been added.
- Incremental interface for SHAKE hashing algorithms has been added.

Jira:RHEL-79116[1]

OpenSCAP rebased to 1.3.12

The OpenSCAP packages have been rebased to upstream version 1.3.12. This version provides bug fixes and various enhancements. For additional information, see the OpenSCAP release notes.

Jira:RHEL-88845

SCAP Security Guide provided in 0.1.76

For details, see the SCAP Security Guide release notes .

Jira:RHEL-74239

6.3. RHEL FOR EDGE

RHEL provides the greenboot package in version 0.15.8

The **greenboot** packages have been updated to version 0.15.8, which provides bug fixes and enhancements. Notable changes include:

- Fixed the **bootc** compatibility with **rpm-ostree** when **bootc** is available alongside **rpm-ostree**.
- General bug fix: If **bootc** is not available, rollback using **rpm-ostree**.

Jira:RHEL-80003

6.4. SUBSCRIPTION MANAGEMENT

Ability to control feature enablement during rhc connect using CLI options for better control

With the enhanced **rhc connect** command, you can now enable or disable specific features by using the **--enable-feature** and **--disable-feature** CLI options. By default, the following features are enabled:

- Content: Provides access to Red Hat CDN repositories.
- Analytics: Triggers system registration with Red Hat Insights.
- Remote-management: Starts the **yggdrasil.service**.

Additionally, feature dependencies are enforced to prevent invalid configurations. When using **--format json**, the output now includes feature enablement details, improving automation and visibility.

Jira:RHEL-65517[1]

The subscription-manager status command describes only the registration status

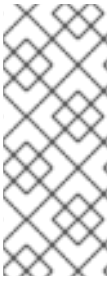
Previously, the output of the **subscription-manager status** command in Simple Content Access (SCA) mode included several details such as the compliance status. With this enhancement, the output of the **subscription-manager status** command has been simplified to state only the registration status.

Jira:RHEL-78003[1]

6.5. SOFTWARE MANAGEMENT

The repository metadata is now not downloaded by default

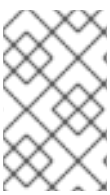
Previously, when you downloaded a repository's metadata, the filelists metadata was downloaded by default. The filelists metadata is large and is typically not needed. With this update, this metadata is not downloaded by default, which improves responsiveness and saves disk space. The filelists metadata is also no longer downloaded or updated from repositories and is not loaded into the DNF transaction when you run a **dnf** command. If the **dnf** command requires the filelists metadata or includes a file-related argument, the metadata is loaded automatically.



NOTE

When a package has a filepath dependency that requires filelists metadata to be resolved, the transaction fails with a dependency resolution error and the following hint:

```
(try to add '--skip-broken' to skip uninstallable packages or '--
setopt=optional_metadata_types=filelists' to load additional filelists metadata)
```



NOTE

If you want to re-enable the default filelist metadata downloading, you can add the **filelists** value to the **optional_metadata_types** option in the **/etc/dnf/dnf.conf** configuration file.

Jira:RHEL-12355[1]

DNF now uses librpmio for processing PGP keys

To verify RPM package signatures, RPM uses the **rpm-sequoia** library instead of the previously-used custom PGP parser. With this update, the **librepo** library, which can verify PGP signatures on DNF repositories, now also uses **rpm-sequoia** through the **librpmio** library. As a result, to provide consistent user experience, the **dnf**, **librpm**, and **rpm** components now use the same PGP implementation.

Jira:RHEL-47106

dnf-plugins-core provided in version 4.7.0

RHEL 10 provides the **dnf-plugins-core** package in version 4.7.0 that includes a new **python3-dnf-plugin-pre-transaction-actions** package. This package includes a new **pre-transaction-actions** DNF plugin that allows you to run a command upon starting an RPM transaction. For more information, see the **dnf-pre-transaction-actions(8)** manual page on your system.

Jira:RHEL-38831

createrepo_c provided in version 1.0.0

RHEL 10 provides the **createrepo_c** package in version 1.0.0. Notable changes over the previous version include:

- Default compression switched from **gz** to **zstd**, which provides smaller metadata that is faster to decompress. Note that the **gz** compression is still supported.
- To save time and disk space, metadata in the SQLite database format is no longer generated by default. Note that you can still create this metadata by using the **--database** switch or the **sqliterepo_c** tool.
- Managing the **group.xml** metadata has been standardized. Previously, this metadata was present twice, as compressed and uncompressed. With this update, the group metadata is present only once as compressed and has the **group** metadata type.

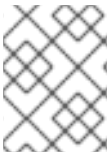
NOTE

The **group.xml** metadata is not compatible with YUM in RHEL 7. If required, you can still create repositories with the old layout by using the **modifyrepo_c** command.

Jira:RHELDOCS-18997[1]

DNF, PackageKit, and microdnf tools now install only newly recommended packages during an upgrade

The **exclude_from_weak_autodetect** option can auto-detect unmet weak dependencies of installed packages and block installation of packages that satisfy already unmet dependencies. Before this update, this option was set to **False** by default. Consequently, all existing weak dependencies of a package were installed when upgrading that package, even if some weak dependencies were not previously installed. With this update, the default value for the **exclude_from_weak_autodetect** option has been set to **true**. As a result, only newly recommended packages are now installed during an upgrade with the DNF, PackageKit, or **microdnf** tools.



NOTE

You can manually change the default value of **exclude_from_weak_autodetect** in the **/etc/dnf/dnf.conf** configuration file.

Jira:RHELDOCS-19415[1]

The RPM database relocated to /usr

With this update, the RPM database has been moved from the **/var/lib/rpm** directory to the **/usr/lib/sysimage/rpm** directory. Storing the database in **/usr** simplifies the creation and rollback of system snapshots because the contents of **/var** no longer have to be considered. It also aligns RHEL with **rpm-ostree** based systems, such as RHEL CoreOS, which already store the RPM database under the **/usr** directory.

NOTE

This change has no visible effect on the majority of users because RPM has not changed in its functionality. However, advanced users who perform OS-level snapshots, which usually include the **/usr** directory, no longer have to include the RPM database located in **/var/lib/rpm** in the snapshot to preserve the system state upon rollback.

Jira:RHELDOCS-19417[1]

A new --exclude-services flag to exclude systemd services from the list of stale processes

You can use the **dnf needs-restarting --services** to list **systemd** services that need restarting. With this update, a new **--exclude-services** flag has been added to **dnf needs-restarting**. You can use this flag to exclude **systemd** services from the list of stale processes.

Jira:RHEL-56137

Image mode for RHEL users can now use dnf --transient to perform package transactions that reset on reboot

Previously, Image mode for RHEL users could transiently install, remove, and upgrade packages by running the **bootc usr-overlay** command to unlock the system and then make changes by running DNF commands. If you use **bootc usr-overlay**, when the system reboots, the **/usr** directory overlay disappears and all changes made to it will reset. Changes to other directories, including configuration in **/etc** and program state in **/var**, persist across reboots.

With this update, a new **--transient** flag and a new **persistence** configuration option have been added to DNF to improve the user experience on bootc systems. You can now skip the **bootc usr-overlay** step by using either of the following options:

- Use the **dnf --transient** command.
- Set the **persistence** option to **transient** in the **dnf.conf** file.

NOTE

Unlike when using **bootc usr-overlay**, **--transient** and **persistence=transient** ensure that the **/usr** directory remains read-only to other processes before, during, and after the transaction.

For example, to transiently install the **make** package, enter:

```
# dnf install --transient make
```

Jira:RHEL-76849

6.6. SHELLS AND COMMAND-LINE TOOLS

RHEL 10 provides polkit in version 125

The polkit package is upgraded to version 125. Notable enhancements include the following:

- polkit uses the **tmpfiles.d** file to store configuration in the **/etc/polkit-1** directory.
- polkit now supports **syslog-style** log levels and LogControl protocol for dynamic loglevel changing.

The rebase allows the removal of **/etc/polkit-1/<subdirs>** directories and their automatic recreation with appropriate access rules on the next boot. It aligns polkit with the **reset OS to factory settings by deleting /etc** approach. Now, the user does not have to reinstall polkit, if the **/etc/polkit-1** directory was deleted.

Additionally, the **polkit.service** unit file now contains a new parameter specified in the call of polkitd daemon, that is, **--log-level=<level>**. By default in RHEL 10, this parameter is set to **--log-level=err**, logging only error messages. If the parameter **--log-level** is omitted, only critical messages are logged.

This change allows users to control how verbose polkit should be in logs and especially in the journal. The enhancement addresses the requirement to log every loaded **.rules** file for debug purposes, preventing the journal from being flooded with unnecessary information.

Jira:RHEL-55287

RHEL 10 provides ksh in version 93u+m/1.0.10

The **KornShell** (ksh) shell is upgraded to the 93u+m/1.0.10 version. The notable changes are:

- The **alarm** command, a shell built-in part of ksh, is no longer supported and will be removed. The replacement is the **cron** daemon, a utility for tasks that must run at fixed intervals.
- The ksh shell is now capable of handling more than 32767 simultaneous background jobs, subject to system limitations.
- Fixes a bug that caused an incorrect default exit status for **exit** within a trap action and a race condition occurring on some systems when running an external command with a redirection from a command substitution.
- Various other bug fixes

Jira:RHEL-45981

Traceroute now defaults to IPv6

Previously, traceroute defaulted to IPv4 addresses even when IPv6 addresses were available. With this enhancement, traceroute now defaults to IPv6 if available.

Jira:RHEL-58449

Changes in the polkit-rules visibility

Previously, in the version polkit-123, the default file mode for files in the **/usr/share/polkit-1/rules.d** directory was set explicitly, so it did not inherit the mode from the parent directory. The default file mode for files in the **/etc/polkit-1/rules.d** directory was previously owned by the **polkitd**. With this enhancement, the notable changes include the following:

The /usr/share/polkit-1/rules.d directory

- The default permission mask for files in **/usr/share/polkit-1/rules.d** is changed from **700 polkitd root** to **755 root root**, and is now visible to all users.
- The reason behind the change is that files in this directory are endorsed by various packages and are accessible in the project's public repositories.
- Previously, the permission mask or file mode was non-standard. The new file permission mask is also aligned with the Filesystem Hierarchy Standard (FHS).

The /etc/polkit-1/rules.d directory

- Files in the **/etc/polkit-1/rules.d** directory represent adjustments created by the system administrator (custom rules that are different from the vendored rules that reside in the **/usr/share/polkit-1/rules.d**). These files can contain customer-specific data about specific personnel and their privileges.
- The default permission mask for files in the **/etc/polkit-1/rules.d** directory has been changed to **0750 root polkitd** for increased security. The polkit daemon is in the **polkitd** group and this group only has **read** access to the files instead of the **write** access. Even in the case of unauthorized access to the polkit daemon, the attacker cannot change the rules and cannot grant someone any other privileges. The files are also invisible to any user other than **root** or **polkitd** group.

NOTE

Do not store your **custom .rules** files in **/usr/share/polkit-1/rules.d**. For safety reasons, store or migrate your custom rules to the **/etc/polkit-1/rules.d directory**.

Jira:RHELDOCS-16414[1]

RHEL 10 provides systemd version 257

The **systemd** package has been rebased to version 257. Notable changes include:

- Support for **cgroup v1**, including **legacy** and **hybrid** hierarchies, is now considered obsolete. Now, **systemd** always uses **cgroup v2**, even if **systemd.legacy_systemd_cgroup_controller=yes** is set on the kernel command line.
- Support for **System V** service scripts is deprecated and will be removed in future versions.
- Default configuration files are now located under the **/usr/lib/systemd/** directory instead of **/etc/systemd/**. The default configuration files can be overridden by a user configuration from **/etc** or extended by using drop-in files similarly to unit files. For more details, see the CONFIGURATION DIRECTORIES AND PRECEDENCE section in systemd-system.conf(5) man pages of the specific configuration files.

Note: Update your software **now** to include a native **systemd** unit file instead of a legacy **System V** script to maintain compatibility with future **systemd** releases.

Jira:RHELDOCS-19411[1]

RHEL 10 provides ReaR in version 2.9

The ReaR utility has been upgraded to version 2.9 . The notable changes include :

- On IBM Z, the **IPL** output method is now deprecated. The **RAMDISK** output method is provided as an alternative. The **OUTPUT=RAMDISK** functionality is identical on all the supported hardware architectures, unlike the deprecated **OUTPUT=IPL** functionality, which is specific to IBM System Z.

Note that the names of the recovery RAM disk image and the kernel that are generated by ReaR are different with **OUTPUT=RAMDISK**. The kernel is named **kernel-$RAMDISK_SUFFIX** and the ramdisk image is named **initramfs-$RAMDISK_SUFFIX.img**. The **RAMDISK_SUFFIX** is a configuration variable that you can set in **/etc/rear/local.conf**. If the variable is not set, it defaults to the hostname of the system. If you used the **OUTPUT=IPL** setting with previous ReaR versions, change it to **OUTPUT=RAMDISK** and adjust any automation that uses the resulting kernel and RAM disk image files according to the new naming convention described above to be compatible with future ReaR versions when the **IPL** output method is removed.

- The default value of the **ISO_VOLID** configuration variable, which specifies the label of the resulting ISO image when using the **OUTPUT=ISO** setting, was changed to **REAR-ISO**. The default in previous ReaR versions was **RELAXRECOVER**. If you need to mount the resulting ISO 9660 file system by label, adjust the **mount** command for the label change. Alternatively, you can set the **ISO_VOLID** variable in **/etc/rear/local.conf** to **RELAXRECOVER** to restore the former behavior.

Jira:RHEL-72557[1]

The tmux service is now available

The system administrator can now set up a **tmux** session for specific users at boot. This is useful on systems, where the **KillUserProcesses=yes** parameter is set and users are not configured to linger.

Jira:RHEL-62152

RHEL 10 provides openCryptoki version 3.24.0

The **openCryptoki** packages are provided in version 3.24.0. Support has been added for the following:

- CCA token on non-IBM Z platforms (x86_64, ppc64)
- IBM Dilithium
- RSA-OAEP with SHA-224, SHA-384, and SHA-512 on encryption and decryption
- PKCS #11 v3.0 SHA-3 mechanisms
- SHA-2 mechanisms
- SHA-based key derivation mechanisms
- Protecting tokens with a token specific user group

- New **libica** AES-GCM API using the KMA instruction on z14 and later

Jira:RHEL-58996[1]

6.7. INFRASTRUCTURE SERVICES

tuned-ppd, Valkey, libcpuid and dnsconfd packages are now available

The following packages are included in Red Hat Enterprise Linux:

- **tuned-ppd** : The **tune-ppd** is a replacement of **drop-in power-profiles-daemon** which uses **TuneD** as a backend.
- **Valkey** : Replaces Redis and provides the same features.
- **libcpuid** : Enables accurate CPU model identification in **TuneD**.
- **dnsconfd** : A local DNS cache configuration daemon that simplifies setting up DNS caching, split DNS, DNS over TLS, and other DNS features.

Jira:RHELDOCS-18925[1]

GECOS field for root user is changed to Super User

Previously, an application output for the GECOS/description appeared as **root** . Now, the GECOS/description for user **root** in the **/etc/passwd** file has been changed from **root** to **Super User**.

Jira:RHELDOCS-18776[1]

dnsconfd daemon can now be installed

With this enhancement, you can now install the **dnsconfd**, a local DNS cache configuration daemon. The newly configured daemon provides an easy way to set up DNS caching, split DNS, DNS over TLS, and other DNS features.

Jira:RHEL-34791[1]

The Kea DHCP server replaces ISC DHCP

Kea is a new Dynamic Host Configuration Protocol (DHCP) server solution in RHEL. Kea DHCP is an implementation from Internet Systems Consortium (ISC) that includes fully functional DHCPv4, DHCPv6, and Dynamic DNS servers. The Kea DHCP server has the following advantages:

- It is an extensible server solution with module hooks.
- It allows re-configuration through the REST API.
- It has a design that allows separation of data (leases) and execution environment.

Jira:RHEL-9306[1]

Weak ciphers can be now disabled in CUPS configuration

Previously, when you disabled the weak cipher in the system-wide cryptographic policy followed by CUPS configurations, the configuration changes did not take effect. With this enhancement, if a user wants to disable a certain cryptographic algorithm via system policy, CUPS honors the system settings,

unless **SSLOptions NoSystem** is set in CUPS configuration files. In that case CUPS does not offer the system-wide disabled algorithm anymore.

As a result, by default, now **Cupsd** and **libcups** follow system crypto policy. You can opt-out from crypto policy by setting **SSLOptions NoSystem** in the following configuration files:

- **/etc/cups/client.conf**: for applications using **libcups**
- **/etc/cups/cupsd.conf**: for **cupsd** daemon

It is not secure to set the **NoSystem** value, as it allows weaker algorithms to be enabled if they are disabled by system crypto policy. It should be used only if the other part in communication does not support any better crypto algorithms.

Jira:RHEL-68415[1]

6.8. NETWORKING

RHEL 10 provides nftables version 1.1.1

The RHEL **nftables** framework has implemented changes from upstream versions 1.1.0 and 1.1.1. This update provides multiple bug fixes and enhancements. Notable changes include:

- Added support for multiple devices in JSON format.
- Increased performance when listing tables.
- Added virtual local area network (VLAN) ID match and set support, including the 802.1ad (Q-in-Q) standard.
- Enabled zero burst in byte rate limiter.
- Added egress support for **list hooks**.
- Fixed listing inconsistencies in the **nft list hooks** command.

For more details and a full list of changes, see:

- 1.1.0. upstream release notes.
- 1.1.1. upstream release notes.

Jira:RHEL-65346

RHEL 10 provides iptables version 1.8.11

The **iptables** framework has been upgraded to version 1.8.11, which provides multiple bug fixes and enhancements. Notable changes include:

- New **arptables-translate** utility
- **ebtables-nft**:
  - Print negations (exclamation marks) before the match they invert for consistency with **iptables**.

- Support **--replace** and **--list-rules** command options.

- **iptables-translate**:
  - Align protocol name lookups with **iptables**.
  - Support socket match with **TPROXY** target.

- **iptables**:
  - Enable implicit extension lookup for **dccp** and **ipcomp** protocols so that no extra **-m <proto>** command option is needed after **-p <proto>**.

- **iptables-save**:
  - Avoid calls to the **getprotobynumber()** function for consistency and improved performance with huge rule sets.

- **arptables-nft**:
  - Fixed wrong formatting of **--h-type** values and **--proto-type** masks which caused misinterpretation by **arptables-restore**.
  - Improved ineffective masks when specified in **--h-type**, **--opcode** and **--proto-type** matches.

- **iptables-nft**:
  - Fixed wrong error messages in corner-case error conditions.
  - Fixed incorrect combination of inverted payload matches.

For more details, see the upstream documentation.

Jira:RHEL-66725

RHEL 10 provides firewalld version 2.3.0

The **firewalld** service version 2.3.0 provides multiple enhancements. Notable changes include:

- Added the **StrictForwardPorts** (boolean, defaults to "no") configuration option that allows **firewalld** to be strict about Destination NAT traffic. When enabled, only forward ports explicitly enabled in **firewalld** are allowed. This means that container-published ports will be blocked. For more information about the feature, see StrictForwardPorts.

- Added support for the following services:
  - client/server on Advanced Linux Sound Architecture (ALSA) sequencer (**aseqnet**)
  - Music Player Daemon (MPD)
  - Radsec
  - SlimeVR

For more details about the release updates, see the upstream repository.

Jira:RHEL-65865

RHEL 10 provides xdp-tools version 1.5.1

The **xdp-tools** package has been upgraded to version 1.5.1, which provides multiple enhancements and bug fixes. Notable changes include:

- Added the **xdp-forward** utility that enables XDP-accelerated packet forwarding between supported network devices.
- Updated the **xdp-trafficgen** utility to support specifying User Datagram Protocol (UDP) packet sizes.
- Added a new option-based API for creating XDP sockets (XSK) and user memory (UMEM) objects.

Jira:RHEL-45730

The RHEL kernel supports the netkit network device type

The RHEL kernel now supports the **netkit** network device type, which enables Berkeley Packet Filter (BPF) based high performance networking for containers. This change should positively impact efficiency, scalability, and responsiveness of containerized applications that are deployed with a Container Network Interface (CNI) that supports the **netkit** network device type, particularly in cloud environments and high-throughput systems.

Jira:RHEL-51429[1]

The i40e driver supports automatic reset behavior on MDD events

The Intel® Network Adapter Driver for PCIe* 40 Gigabit Ethernet can now reset problematic Single Root I/O Virtualization (SR-IOV) virtual functions (VFs) when it detects a malicious driver detection (MDD) event. You can activate this automatic reset behavior through the new **mdd-auto-reset-vf** option as in the following example command:

```
ethtool --set-priv-flags _ethX_ *mdd-auto-reset-vf* on
```

When the VF sends malformed packets classified as malicious, it can cause the Tx queue to freeze, which makes it unusable for several minutes. However, with **mdd-auto-reset-vf** enabled, a graceful VF reset automatically restores operational state when an MDD event occurs.

Jira:RHEL-73034[1]

nmstate supports the require-id-on-certificate setting on Libreswan configuration

With this enhancement, **libreswan**, an implementation of Internet Protocol Security (IPsec) specification, now supports the **require-id-on-certificate** setting for VPN configurations by using NetworkManager. With this feature, you can configure Subject Alternative Name (SAN) validation by using the **require-id-on-certificate** option. As a result, this implementation correctly enforces SAN validation based on the specified setting:

- No SAN validation is performed when set to **no**
- SAN are validated when set to **yes**

Jira:RHEL-58812[1]

RHEL 10 provides wpa_supplicant version 2.11

The **wpa_supplicant** service has been upgraded to version 2.11, which provides multiple enhancements and bug fixes. Notable changes include:

- Added support for Device Provisioning Protocol (DPP) release 3.
- Added support for GCM-AES-256 cipher suite.
- Added support for Basic Service Set (BSS) Color updates.
- Implemented OpenSSL 3.0 API changes.

For more information and the full list of changes, see the upstream announcement.

Jira:RHEL-59010[1]

6.9. KERNEL

Kernel version in RHEL 10.0

Red Hat Enterprise Linux 10.0 is distributed with the kernel version 6.12.0.

Dynamic EFIVARS pstore backend is now supported

With this release, you can dynamically enable the **EFIVARS** pstore backend at runtime to efficiently manage the system storage.

Previously, the pstore storage backend required a reboot to modify its configuration. With this release, you can switch between supported backends such as **NVMe** and **EFIVARS** without rebooting the system.

Also, enhancements in pstore logging provide better clarity on indications of the currently active backend.

If there is no pstore backend registered on your system, enable the **efi_pstore** for UEFI boot:

```
# echo "N" > /sys/module/efi_pstore/parameters/pstore_disable
[   90.116913] pstore: Using crash dump compression: deflate
[   90.118433] pstore: Registered efi_pstore as persistent store backend
```

Jira:RHELDOCS-19988[1]

Containerization of the rteval utility

With this update, you can run the **rteval** utility with all its runtime dependencies from a container image publicly available through the Quay.io container registry. You can:

- Enjoy the deployment flexibility, where older RHEL versions can get newer versions of **rteval**.
- Create an isolated environment to ensure the performance evaluations do not disrupt other system processes or consume excessive resources.
- Run multiple **rteval** instances on the same or multiple hosts.
- Allocate specific system resources to **rteval**, ensuring better resource usage control.

Alternatively, you can use the related Docker file to build your own container image with **rteval**. This Docker file is located in the upstream repository and provided as part of the source RPM (SRPM).

Jira:RHEL-28059[1]

New option to disable idle states locally on CPUs during rtla-timerlat testing: deepest-idle-state

The arguments for the **deepest-idle-state** are the number of the deepest allowed idle state. If -1 is the value in the argument, it * disables all idle states. In the **rtla-timerlat** instead of using **/dev/cpu_dma_latency** to disable the CPUs in the idle state globally, the **deepest-idle-state** option is added to set the deepest allowed idle state for CPUs where measurements are running.

As a result, you can save power and reflect the real-time workload during **rtla-timerlat** testing and use the **deepest-idle-state** instead of using the **/dev/cpu_dma_latency** to disable them globally.

Jira:RHEL-40744[1]

Deadline (DL) server implements a two-stage scheduler for CFS tasks

RHEL 10 introduces a new in-kernel Deadline (DL) server that implements a two-stage scheduler. It provides guaranteed execution time for Completely Fair Scheduler (CFS) tasks, mitigating potential starvation caused by Real Time (RT) or Deadline (DL) tasks.

The new DL server, running at deadline priority, schedules CFS tasks every 1 second, allocating an initial 50-millisecond runtime window for the execution. This ensures that CFS tasks receive periodic CPU time even when preempted by higher-priority RT or DL tasks. The runtime and period parameters can be adjusted on a per-CPU basis by using **/sys/kernel/debug/sched/fair_server/cpu*/{runtime, period}**. Setting a runtime of **0** disables the DL server for the specified CPU.

The DL server eliminates the need for external tools, such as **stalld**, for starvation prevention and removes the requirement for manual configuration and tuning of such tools.

This provides a robust, integrated, and transparent solution for CFS task scheduling directly within the kernel.

Jira:RHEL-58211[1]

Landlock, a new Linux Security Module (LSM) is released

RHEL 10.0 introduces Landlock, a new security feature that makes your containers safer. Landlock sets strict rules for processes such as Podman to limit access to the file system through the kernel API, defining rules for themselves regardless of privilege level and allowing users to create hard limits over the accessible scope of the processes.

With Landlock, you can build programs that mitigate potential risks associated with misconfigured or maliciously targeted processes. This makes containers and the whole system more secure.

Jira:RHEL-40283[1]

rh_waived kernel command-line boot parameter is now supported

With this release, the **rh_waived** kernel command-line boot parameter is supported. **rh_waived** is used for enabling waived features in RHEL. The waived features are kernel features considered unmaintained, insecure, rudimentary, or deprecated. These features are disabled by default in RHEL 10. To use waived features, you must enable them manually.

Jira:RHEL-26170[1]

New timerlat-interval INTV_US and cyclictest-interval INTV_US options

With this enhancement, you can use the following new options of the **rteval** command to modify the base or periodic interval option in running **timerlat** or **cyclictest** threads:

- **timerlat-interval INTV_US**
- **cyclictest-interval INTV_US**

Note that if you do not use either of these options with **rteval**, the default value is applied.

Jira:RHEL-67424[1]

New option to disable idle states locally on latency testing with cyclictest

- The **cyclictest** tool sets **/dev/cpu_dma_latency** to 0 by default to avoid increased latency when waking up from idle, which disables idle states on all CPUs.
- The new **deepest-idle-state** option only disables idle states on CPUs which are selected for the testing. The argument specifies the deepest allowed idle state, setting it to **-1** disables all idle states on the measured CPUs.
- Tuning with the **cyclictest** is supposed to reflect the real-time workload testing, and thus using the **deepest-idle-state** instead of using the **/dev/cpu_dma_latency** to disable the CPU idle states reflects a use case where the real-time workload only disables idle states on the CPU where it is running.
- As a result, the **cyclictest** coverage of addressing all use cases is increased, and power consumption decreases.

Jira:RHEL-65488[1]

New integration testing to validate kdump procedures to prevent system failure

With this enhancement, you can check the log file for **kdump** procedures after any software or hardware updates to prevent system failure. After the analysis of the output log files, the configuration entries, such as **memory issues** or **blacklist of some drivers**, are corrected to validate the **kdump** procedures and generate the **vmcore**. This ensures that the **kdump** procedures are validated and corrected before a system crash after any software or hardware update.

Jira:RHEL-29941[1]

6.10. BOOT LOADER

RHEL 10 provides grub2 in version 2.12

grub2 version rc2.12 provides many bug fixes and enhancements. The notable changes are:

- GCC 13 support.
- clang 14 support.
- binutils 2.38 support.

- Support for dynamic GRUB runtime memory addition using firmware calls.
- PCI and MMIO UARTs support.
- SDL2 support.
- LoongArch support.
- TPM driver fixes.
- Many filesystems fixes.
- Many CVE and Coverity fixes.
- Debugging support improvements.
- Tests improvements.
- Documentation improvements.
- VLAN support

Jira:RHEL-15032[1]

6.11. FILE SYSTEMS AND STORAGE

RHEL 10 provides python-blivet version 3.10

The **python-blivet** package has been rebased to version 3.10, providing various bug fixes and enhancements. The most notable changes are:

- Removed support for Python 2.
- Support for adding disks to the existing Stratis pool.
- Support for Stratis encryption with Clevis or Tang.
- Support for semi-automatic resizing of the **lvmpv** format to fill underlying block devices.

Jira:RHEL-45175

RHEL 10 provides cryptsetup version 2.7

The **cryptsetup** package has been rebased to version 2.7. This version provides various bug fixes and enhancements, most notably:

- Improvements for the **libcryptsetup** package to support LUKS encrypted devices in the **kdump** enabled systems.
- Critical fixes for LUKS2 SED OPAL feature.
- Avoids known or already fixed issues with LUSK2 SED OPAL feature.

Jira:RHEL-33395[1]

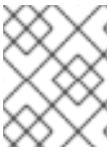
GPT is now the default partition table for IBM Power Systems, Little Endian and 64-bit IBM Z architectures

The GPT partition table is now selected by default instead of MS-DOS when installing RHEL 10 for all newly partitioned disks during the installation.

IMPORTANT

The GPT partition table is not selected by default for direct access storage device (DASD) drives on 64-bit IBM Z architecture, where the DASD partition table remains unchanged.

This update simplifies and standardizes the default partitioning behavior across different architectures and platforms.



NOTE

AMD and Intel 64-bit architectures and other products, such as RHEL Image Mode, already use the GPT partition table by default.

Jira:RHEL-52200

snapm is now available in RHEL

Snapshot Manager (**snapm**) is a new component designed to assist in managing system state snapshots. You can use it to roll back updates or changes, and boot into previous system snapshots. Managing snapshots across multiple volumes and configuring boot entries for snapshot boot and snapshot rollback can often be complex and prone to errors. Snapshot Manager automates these common tasks and integrates seamlessly with Boom Boot Manager, simplifying the process. With this update, you can easily take snapshots of the system state, apply updates, and revert to the previous system state if necessary.

Jira:RHEL-59006[1]

RHEL 10 provides device-mapper-multipath version 0.9.9

The **device-mapper-multipath** package has been updated from version 0.8.7 to 0.9.9. Notable enhancements include:

- The **multipathd.socket systemd** unit is no longer enabled by default. **multipathd** continues to run automatically on boot. However, if stopped, it does not restart automatically if there is a block device **uevent** or certain **multipath** commands are run. To keep it enabled, restart manually or uncomment the following in the **multipathd.socket systemd** file:

```
# WantedBy=sockets.target
```

- **multipathd** now attempts to run as a real-time process with a moderate priority (10) by default. If unsuccessful, it continues to run as a normal process, but with an increased priority. You can control this, by modifying the standard **systemd** options, for example, LimitRTPRIO and CPUWeight in the **multipathd.service systemd** file.

- **systemctl reload multipathd.service** or **multipathd reconfigure** commands now reload a device only if something has changed, instead of reloading every **multipath** device including the ones that are unchanged. To force a reload of all devices, run :

multipathd reconfigure all

- The following **multipath.conf** options were deprecated and are not recognized in RHEL 10. **multipath** triggers a warning message if they are included in the **multipath** file:
  - RHEL 9:
    - **multipath_dir**
    - **config_dir**
    - **bindings_file**
    - **wwids_file**
    - **prkeys_file**
    - **getuid_callout**
    - **disable_changed_wwids**
  - RHEL 8:
    - **default_selector**
    - **default_path_grouping_policy**
    - **default_uid_attribute**
    - **default_getuid_callout**
    - **default_features**
    - **default_path_checker**
- Path grouping policy, **group_by_tpg**, is introduced to group paths by their ALUA target port group. This ensures that all paths with the same target port group belong to the same pathgroup. It functions similar to the group_by_prio policy, but prevents misgrouping when paths change priorities.

IMPORTANT

All the paths in the multipath device must have their priority function set to **alua** or **syfs** to use this policy.

- Configuration settings **detect_pgpolicy** and **detect_pgpolicy_use_tpg** are introduced which can be set in overrides, devices, and defaults sections.
  - If **detect_pgpolicy** is enabled, **multipath** sets **path_grouping_policy** to **group_by_prio** or **group_by_tpg** for **alua** or **sysfs** prioritizer. If it is disabled, **path_grouping_policy** configuration set for the device is used. **detect_pgpolicy** is enabled by default.
  - If **detect_pgpolicy_use_tpg** is enabled, **detect_pgpolicy** sets **path_grouping_policy** to **group_by_tpg**. If it is disabled, **detect_pgpolicy** sets **path_grouping_policy** to **group_by_prio**. **detect_pgpolicy_use_tpg** is disabled by default.

- New wildcards for formatted output in **multipathd**:
  - New maps format wildcard:
    - k: **max_sectors_kb**
  - New paths format wildcards:
    - I: **init state**
    - L: LUN hex
    - A: **alua** target port group
    - k: **max_sectors_kb**

Jira:RHELDOCS-19812[1]

The dm-vdo module has been added to the kernel

With this update, the **kmod-kvdo** module has been replaced with the **dm-vdo** module in the RHEL 10 kernel. In addition, the Virtual Data Optimizer (VDO) **sysfs** parameters have been removed. For more information on the removed **sysfs** parameters, see Removed features in File systems and storage .

Jira:RHELDOCS-19842[1], Jira:RHELDOCS-19066

nvme-cli and cryptsetup are now available for Opal automation on NVMe SEDs

NVMe Self-Encrypting Drives (SED) support the Opal storage specification of hardware encryption technology to secure data stored in the drive. Previously, Opal support for NVMe SEDs required manual interaction to manage passwords to access the data.

With this update, you can use **nvme-cli** and **cryptsetup** to automate encryption management and drive unlocking.

Run the following commands to use NVMe SED options on NVMe SSD:

- To discover SED Opal locking features:

```
# nvme sed discover /dev/nvme0n1
Locking Features:
 Locking Supported: Yes
 Locking Feature Enabled: No
 Locked: No
```

- To initialize an SED Opal device for locking:

```
# nvme sed initialize /dev/nvme0n1
New Password:
Re-enter New Password:
# nvme sed discover /dev/nvme0n1
Locking Features:
 Locking Supported: Yes
 Locking Feature Enabled: Yes
 Locked: No
```

- To lock a SED Opal device:

```
# nvme sed lock /dev/nvme0n1
# nvme sed discover /dev/nvme0n1
Locking Features:
 Locking Supported: Yes
 Locking Feature Enabled: Yes
 Locked: Yes
```

- To unlock a SED Opal device:

```
# nvme sed unlock /dev/nvme0n1
# nvme sed discover /dev/nvme0n1
Locking Features:
 Locking Supported: Yes
 Locking Feature Enabled: Yes
 Locked: No
```

- To change the SED Opal device password:

```
# nvme sed password /dev/nvme0n1
Password:
New Password:
Re-enter New Password:
```

- To revert an SED Opal device from locking:

```
# nvme sed lock /dev/nvme0n1
# nvme sed discover /dev/nvme0n1
Locking Features:
        Locking Supported:         Yes
        Locking Feature Enabled:   Yes
        Locked:                    Yes
# nvme sed unlock /dev/nvme0n1
# nvme sed discover /dev/nvme0n1
Locking Features:
        Locking Supported:         Yes
        Locking Feature Enabled:   Yes
        Locked:                    No
# nvme sed revert /dev/nvme0n1
```

- To reset an SED Opal device to disable locking with destructive revert:

```
# nvme sed lock /dev/nvme0n1
# nvme sed discover /dev/nvme0n1
Locking Features:
        Locking Supported:       Yes
        Locking Feature Enabled: Yes
        Locked: Yes
# nvme sed revert -e /dev/nvme0n1
Destructive revert erases drive data. Continue (y/n)? y
   Are you sure (y/n)? y
   Password:
   # nvme sed discover /dev/nvme0n1
```

```
    Locking Features:
      Locking Supported:       Yes
      Locking Feature Enabled: No
      Locked:                  No
```

Note: Use **nvme sed revert** without the **-e** parameter to avoid erasing data on the NVMe disk.

The device might be either an NVMe character device such as **/dev/nvme0**, an NVMe block device such as **/dev/nvme0n1**, or an **mctp** address in the form **mctp:<net>,<eid>[:ctrl-id]**.

Example command to use an NVMe OPAL device on RHEL 10 with nvme-cli:

- Initialize, lock, and unlock an NVMe disk, and verify that data on the disk remains unchanged after unlocking:

```
# mount /dev/nvme0n1p1 /mnt/
# dd if=/dev/urandom of=/mnt/test.file bs=1M count=1024
1024+0 records in
1024+0 records out
1073741824 bytes (1.1 GB, 1.0 GiB) copied, 3.65616 s, 294 MB/s
# md5sum /mnt/test.file
57edc80dab5bf803d0944e281bf2e9dd  /mnt/test.file
# umount /dev/nvme0n1p1
# nvme sed discover /dev/nvme0n1
Locking Features:
 Locking Supported:         Yes
 Locking Feature Enabled:   No
 Locked:                    No
# nvme sed initialize /dev/nvme0n1
New Password:
Re-enter New Password:
# nvme sed lock /dev/nvme0n1
# nvme sed discover /dev/nvme0n1
Locking Features:
 Locking Supported:         Yes
 Locking Feature Enabled:   Yes
 Locked:                    Yes
# mount /dev/nvme0n1p1 /mnt/
mount: /mnt: can't read superblock on /dev/nvme0n1p1.
       dmesg[8] may have more information after a failed mount system call.
# nvme sed unlock /dev/nvme0n1
# mount /dev/nvme0n1p1 /mnt/
# md5sum /mnt/test.file
57edc80dab5bf803d0944e281bf2e9dd  /mnt/test.file
# umount /dev/nvme0n1p1
# nvme sed discover /dev/nvme0n1
Locking Features:
   Locking Supported:         Yes
   Locking Feature Enabled:   Yes
   Locked:                    No
# nvme sed revert /dev/nvme0n1
Password:
# nvme sed discover /dev/nvme0n1
Locking Features:
```

```
Locking Supported:        Yes
Locking Feature Enabled:  No
Locked:                   No
```

Jira:RHELDOCS-19877[1]

RHEL 10 provides NFS with TLS support

Network File System (NFS) with Transport Layer Security (TLS) is fully supported. This feature enhances NFS security by enabling TLS for Remote Procedure Call (RPC) traffic, ensuring encrypted communication between clients and servers. For details, see Configuring an NFS server with TLS support.

Note that NFS with TLS relies on support from kernel TLS (kTLS). The kTLS feature for general use is provided as a Technology Preview. For details see the release notes in the Technology Preview features chapter.

Jira:RHEL-74415[1]

CIFS client provides the ability to create special files under SMB shares

Common Internet File System (CIFS) client has the ability to create native Server Message Block (SMB) symlinks by default. You can also create special files, such as character devices, block devices, pipes, and sockets, through Network File System (NFS) or Windows Subsystem for Linux (WSL) reparse points by using the **reparse=default|nfs|wsl** mount option.

Jira:RHEL-78152[1]

Atomic write is now available

RHEL 10 introduces atomic write as a cross-subsystems enhancement across the file system, block layer, and drivers. The **RWF_ATOMIC** flag is used to enable torn-write protection. This ensures that after a system crash or power failure, either all or none of the written data is present on stable storage. In this scenario, partial data writes or torn writes do not occur.

Existing write operations are not atomic, and can be interrupted mid-operation. This can result in partially written data in case of crash and power failures.

This enhancement enables applications that provide critical data consistency guarantees, such as databases, to optimize the performance of their consistency algorithms.

Jira:RHEL-60811[1]

Automatic RAID checks are enabled by default

With this update, the **raid-check** service is enabled by default. This ensures that **raid-check.service** runs automatically at scheduled intervals after the system boots, performing periodic RAID consistency checks without requiring manual intervention.

Jira:RHEL-86165[1]

6.12. HIGH AVAILABILITY AND CLUSTERS

pcs now validates resource parameters when creating or updating a resource

When you create or update a cluster resource, the **pcs** command-line interface now automatically asks

the resource agent to validate the parameters you entered. If you specify **--agent-validation**, an invalid parameter yields an error. To maintain backward compatibility, if you do not specify **--agent-validation**, an invalid parameter prints a warning but does not prevent misconfiguration.

Jira:RHEL-35670

New --yes flag to confirm potentially destructive actions

To confirm potentially destructive actions such as destroying a cluster, unblocking quorum, or confirming a node being fenced, the **pcs** command-line interface now supports the **--yes** flag. Previously, you could confirm these actions by using the **--force** flag, which is also used for overriding validation errors. With these two functions combined in a single flag, a user could inadvertently confirm a potentially destructive action when the intention is only to override a validation error. You should now use the **--force** flag to override validation errors, and you should use the **--yes** flag to confirm potentially destructive actions.

Jira:RHEL-36612

New pcs status wait command

The **pcs** command-line interface now provides a **pcs status wait** command. This command ensures that Pacemaker has completed any actions required by changes to the Cluster Information Base (CIB) and does not need to take any further actions to make the actual cluster state match the requested cluster state.

Jira:RHEL-38491[1]

pcs support for new commands to query the status of a resource in a cluster

The **pcs** command-line interface now provides **pcs status query resource** commands to query various attributes of a single resource in a cluster. These commands query:

- the existence of the resource
- the type of the resource
- the state of the resource
- various information about the members of a collective resource
- on which nodes the resource is running

You can use these commands for pcs-based scripting since there is no need to parse plain text outputs.

Jira:RHEL-38489[1]

New pcs resource defaults and pcs resource op defaults option for displaying configuration in text, JSON, and command formats

The **pcs resource defaults** and **pcs resource op defaults** commands and their aliases **pcs stonith defaults** and **pcs stonith op defaults** now provide the **--output-format** option.

- Specifying **--output-format=text** displays the configured resource defaults or operation defaults in plain text format, which is the default value for this option.
- Specifying **--output-format=cmd** displays the **pcs resource defaults** or **pcs resource op defaults** commands created from the current cluster defaults configuration. You can use these commands to re-create configured resource defaults or resource operation defaults on a

different system.

- Specifying **--output-format=json** displays the configured resource defaults or resource operation defaults in JSON format, which is suitable for machine parsing.

Jira:RHEL-38487[1]

pcsd Web UI now available as a RHEL web console add-on

The **pcsd** Web UI is now available as the HA Cluster Management RHEL web console add-on when the **cockpit-ha-cluster** package is installed. It is no longer operated as a standalone interface.

Jira:RHEL-23048

New Pacemaker option to leave a panicked node shut down without rebooting automatically

You can now set the **PCMK_panic_action** variable in the **/etc/sysconfig/pacemaker** configuration file to **off** or **sync-off**. When you set this variable to **off** or **sync-off**, a node remains shut down after a panic condition instead of rebooting automatically.

Jira:RHEL-39057

New pcs tag command option for displaying cluster resource tags in text, JSON, and command formats

The **pcs tag [config]** command now supports the **--output-format** option for the following use cases:

- Displaying the configured text in plain text format by specifying **--output-format=text**. This is the default value for this option.
- Displaying the commands created from the current cluster tags configuration by specifying **--output-format=cmd**. You can use these commands to re-create configured tags on a different system.
- Displaying the configured tags in JSON format by specifying **--output-format=json**, which is suitable for machine parsing.

Jira:RHEL-21047

Support for exporting fencing level configuration in JSON format and as pcs commands

The **pcs stonith config** and the **pcs stonith level config** commands now support the **--output-format=** option to display the fencing level configuration in JSON format and as **pcs** commands.

- Specifying **--output-format=cmd** displays the **pcs** commands created from the current cluster configuration that configure fencing levels. You can use these commands to re-create configured fencing levels on a different system.
- Specifying **--output-format=json** displays the fencing level configuration in JSON format, which is suitable for machine parsing.

Jira:RHEL-38483

Deleting multiple resources with a single pcs command

Before this update, the **pcs resource delete**, the **pcs resource remove**, the **pcs stonith delete** and the **pcs stonith remove** commands supported the removal of only one resource at a time. With this update, you can now delete multiple resources at once with a single command.

Jira:RHEL-61889

Simplified configuration of globally unique cluster resource clones

To configure a cluster resource clone to be globally unique, it is now sufficient to configure the clone option **clone-node-max > 1** when creating the clone of a previously created resource or resource group. It is no longer necessary to configure the clone option **globally-unique="true"** as well.

Jira:RHEL-56675

Support for encryption of Pacemaker remote connections using SL/TLS certificates

You can now encrypt Pacemaker remote connections by using X.509 (SSL/TLS) certificates. Previously, only pre-shared keys (PSK) were supported for encryption. With support for SL/TLS certificates, you can use existing host certificates for Pacemaker remote connections.

To configure SSL/TLS certificates for Pacemaker remote connections:

1. Create a remote connection with the **pcs cluster node add-guest** command or the **pcs cluster node add-remote command**. When you create a remote connection, the connection uses PSK encryption.
2. Convert the remote connection to use certificates by updating the **PCMK_ca_file**, **PCMK_cert_file**, **PCMK_key_file**, and, optionally, the **PCMK_crl_file** variables on all cluster nodes and Pacemaker remote nodes.

For information on configuring encryption with SL/TLS certificates, see Host and guest authentication of **pacemaker_remote** nodes.

Jira:RHEL-7600

Updated date specification and duration options in Pacemaker rules

Pacemaker rules no longer support the following options:

- Invalid **duration** options: **monthdays**, **moon**, **weekdays**, **weekyears**, **yearsdays**
- Invalid **date-spec** options: **moon**, **yearsdays**

Pacemaker rules now support the following options:

- The supported **duration** options are now **seconds**, **minutes**, **hours**, **days**, **weeks**, **months**, and **years**.
- The supported **date-spec** options are now **seconds**, **minutes**, **hours**, **monthdays**, **weekdays**, **yeardays**, **months**, **weeks**, **years**, and **weekyears**.

You can configure rules that incorporate **duration** and **date-spec** options in the following **pcs** commands:

- **pcs resource defaults**
- **pcs stonith defaults**

- **pcs resource op defaults**
- **pcs stonith op defaults**
- **pcs constraint location**

Jira:RHEL-49527, Jira:RHEL-49524

Removing Booth cluster tickets from the CIB after removal from the Booth configuration

After you remove a Booth cluster ticket by using the **pcs booth ticket remove** command, the state of the Booth ticket remains loaded in the Cluster Information Base (CIB). This is also the case after you remove a ticket from the Booth configuration on one site and pull the Booth configuration to another site by using the **pcs booth pull** command. This might cause problems when you configure a ticket constraint, because a ticket constraint can be granted even after a ticket has been removed. As a consequence, the cluster might freeze or fence a node. You can prevent this by removing a Booth ticket from the CIB with the **pcs booth ticket cleanup** command.

For information about removing a Booth ticket from the CIB, see Removing a Booth ticket.

Jira:RHEL-12709, Jira:RHEL-7602

Support for new Ha Cluster Management features

For RHEL 10, the **pcsd** Web UI is now available as a RHEL web console add-on as the HA Cluster Management application. It is no longer operated as a standalone interface. The HA Cluster Management application now supports the following features:

- When you set the **placement-strategy** cluster property to **default**, the HA Cluster Management application displays a warning near the utilization attributes for nodes and resources. This warning notes that the utilization has no effect due to **placement-strategy** configuration.
- The HA Cluster Management application supports dark mode, which you can set through the user menu in the masthead.

Jira:RHEL-38493[1], Jira:RHEL-38496

6.13. DYNAMIC PROGRAMMING LANGUAGES, WEB AND DATABASE SERVERS

Python 3.12 in RHEL 10

Python 3.12 is the default Python implementation in RHEL 10. Python 3.12 is distributed as a non-modular **python3** RPM package in the BaseOS repository and is usually installed by default. Python 3.12 will be supported for the whole life cycle of RHEL 10.

Additional versions of Python 3 will be distributed as RPM packages with a shorter life cycle through the AppStream repository and will be installable in parallel. The **python** command (**/usr/bin/python**), and other Python-related commands, such as **pip**, are available in the unversioned form and point to the default Python 3.12 version.

Notable enhancements compared to the previously released Python 3.11 include:

- Python introduces a new **type** statement and new type parameter syntax for generic classes and functions.

- Formatted string literal (f-strings) have been formalized in the grammar and can now be integrated into the parser directly.
- Python now provides a unique per-interpreter global interpreter lock (GIL).
- You can now use the buffer protocol from Python code.
- Dictionary, list, and set comprehensions in **CPython** are now inlined. This significantly increases the speed of a comprehension execution.
- **CPython** now supports the Linux **perf** profiler.
- **CPython** now provides stack overflow protection on supported platforms.
- Python 3.12 is compiled with GCC's **-O3** optimization flag, which has been used by default in upstream. As a result, you can observe increased performance of your Python applications and the interpreter.

To install packages from the Python 3.12 stack, you can use, for example, the following commands:

```
# dnf install python3
# dnf install python3-pip
```

To run the interpreter, you can use, for example, the following commands:

```
$ python
$ python3
$ python3 -m pip --help
```

Jira:RHELDOCS-18402[1], Jira:RHEL-45315

RHEL 10 introduces Perl 5.40

RHEL 10 includes Perl 5.40, which provides various enhancements over the previously available version 5.32.

- Core enhancements:
  - Perl now supports Unicode 15.0.
  - You can now use a new **-g** command-line option, which is an alias for the umask option **-0777**.
  - The **-M** command-line option now accepts a space.
  - A new **builtin** module now provides documentation for new always-present functions.
  - A new **try/catch** feature has been added.
  - Deprecation warnings now have specific subcategories to provide finer-grained control. Note that you can still disable all deprecation warnings in a single statement.
  - The **@INC** hooks have been enhanced, including the **$INC** variable and the new **INCDIR** method.

- Forbidden control flow out of the **defer** and **finally** modules is now detected at compile-time.
- The use of **(?{ ... })** and **(??{ ... })** in a pattern now disables various optimisations globally in that pattern.
- The limit for the **REG_INF** regex engine quantifier has been increased from 65,536 to 2,147,483,647.
- A new regexp variable **${^LAST_SUCCESSFUL_PATTERN}** allows access to the last successful pattern that matched in the current scope.
- A new **__CLASS__** keyword has been introduced.
- Perl now supports a new **^^** logical XOR operator.

- Incompatible changes:
  - A physically empty **sort** function now triggers a compile-time error.
  - The **readline()** function no longer clears the stream error and EOF flags.
  - **INIT** blocks no longer run after an **exit()** function inside a **BEGIN** block.
  - Calling the **import** method on an unknown package now produces a warning.
  - The **return** function no longer allows an indirect object.
  - Changes in errors and warnings can now cause failures in tests.
- Deprecations:
  - The use of the **'** character as a package name separator is deprecated.
  - The **switch** feature and the smartmatch operator **~~** are deprecated.
  - Using the **goto** function to jump from an outer scope into an inner scope is deprecated.
- Internal changes:
  - Multiple deprecated C functions have been removed.
  - Internal C API functions are now hidden with the **__attribute__((hidden))** attribute on the platforms that support it. This means they are no longer callable from XS modules on those platforms.
- Modules:
  - The **Term::Table** and **Test2::Suite** modules have been added to Perl Core.
  - Most modules have been updated.

For more information, see the **perl5340delta**, **perl5360delta**, **perl5380delta**, and **perldelta** man pages.

Jira:RHELDOCS-18869[1]

RHEL 10 introduces Ruby 3.3

RHEL 10 includes Ruby 3.3.7. This version provides several performance improvements, bug and security fixes, and new features:

Notable enhancements include:

- You can use the new **Prism** parser instead of **Ripper**. **Prism** is a portable, error tolerant, and maintainable recursive descent parser for the Ruby language.
- YJIT, the Ruby just-in-time (JIT) compiler implementation, is no longer experimental and it provides major performance improvements.
- The **Regexp** matching algorithm has been improved to reduce the impact of potential Regular Expression Denial of Service (ReDoS) vulnerabilities.
- The new experimental RJIT (a pure-Ruby JIT) compiler replaces MJIT. Use YJIT in production.
- A new M:N thread scheduler is now available.

Other notable changes:

- You must now use the **Lrama** LALR parser generator instead of **Bison**.
- Several deprecated methods and constants have been removed.
- The **Racc** gem has been promoted from a default gem to a bundled gem.

To install Ruby 3.3, enter:

```
# dnf install ruby
```

For information about the length of support of Ruby 3.3, see Red Hat Enterprise Linux Application Streams Life Cycle.

Jira:RHELDOCS-19658[1]

RHEL 10 provides Node.js 22

RHEL 10 is distributed with **Node.js 22**. This version provides numerous new features, bug fixes, security fixes, and performance improvements over previously available **Node.js 20**.

Notable changes include:

- The **V8** JavaScript engine has been upgraded to version 12.4.
- The **V8 Maglev** compiler is now enabled by default on architectures where it is available (AMD and Intel 64-bit architectures and the 64-bit ARM architecture).
- **Maglev** improves performance for short-lived CLI programs.
- The **npm** package manager has been upgraded to version 10.8.1.
- The **node --watch** mode is now considered stable. In **watch** mode, changes in watched files cause the **Node.js** process to restart.
- The browser-compatible implementation of **WebSocket** is now considered stable and enabled by default. As a result, a WebSocket client to Node.js is available without external dependencies.

- **Node.js** now includes an experimental feature for execution of scripts from **package.json**. To use this feature, run the **node --run <script-in-package.json>** command.

To install Node.js 22 enter:

```
# dnf install nodejs
```

Jira:RHEL-35992

RHEL 10 introduces PostgreSQL 16

RHEL 10 is distributed with PostgreSQL version 16.

Notable enhancements include:

- The enhanced bulk loading improves the performance.
- The new **load_balance_hosts** option in the **libpq** library supports more efficient load balancing.
- Configuration files in the **/var/lib/pgsql/data/** directory support including custom **pg_hba.conf** and **pg_ident.conf** files.
- The **/var/lib/pgsql/data/pg_hba.conf** file supports regular expression matching on database and role entries.

Other changes include:

- Absence of the **postmaster** binary. Use the **postgres** binary instead. This change affects only user who use **postmaster** to start the service.
- Absence of the PDF documentation within the package. Use the upstream documentation instead.

For more information, see Using PostgreSQL.

To install PostgreSQL 16, enter:

```
# dnf install postgresql16
```

Jira:RHEL-62694

RHEL 10 introduces MySQL 8.4

RHEL 10 is distributed with MySQL 8.4. Notable changes over the previously available version 8.0 include:

- The deprecated **mysql_native_password** authentication plugin is no longer enabled by default.
- When upgrading to MySQL 8.4, user accounts or roles that have the **BINLOG_ADMIN** privilege are automatically granted the **TRANSACTION_GTID_TAG** privilege.
- When you install MySQL 8.4, the **mysql_upgrade_history** file is created or updated in the server's data directory. The file is in JSON format and includes information about the version installed, date and time of installation, and whether the release was part of a Long-Term Support (LTS series) or an Innovation series.

- The use of the **%** and _ characters as wildcards in database grants has been deprecated, and the wildcard functionality will be removed in a future MySQL release. These characters will be treated as literals. They are already treated as literals when the **partial_revokes** server system variable is set to **ON**.
- The treatment of the **%** character by the server as a synonym for localhost when checking privileges has been deprecated.
- The deprecated **--ssl** and **--admin-ssl** server options and **have_ssl** and **have_openssl** server system variables have been removed. Use the **--tls-version** and **--admin-tls-version** server system variables instead.
- The deprecated **default_authentication_plugin** system variable has been removed. Use the **authentication_policy** server system variable instead.
- The deprecated **SET_USER_ID** privilege has been removed. Instead, you can use the **SET_ANY_DEFINER** privilege for definer object creation and the **ALLOW_NONEXISTENT_DEFINER** privileges for orphan object protection.
- The deprecated **mysql_upgrade** utility has been removed.

For more information, see the upstream MySQL documentation.

Jira:RHEL-36050

RHEL 10 provides PostgreSQL 16 with the pgvector extension

RHEL 10 is distributed with PostgreSQL 16. In addition to the **pgaudit**, **pg_repack**, and **decoderbufs** extensions, the Postgresql stack now provides the **pgvector** extension. With the **pgvector** extension, you can store and query high-dimensional vector embeddings directly within PostgreSQL databases and perform a vector similarity search. Vector embeddings are numerical representations of data that are often used in machine learning and AI applications to capture the semantic meaning of text, images, or other data types.

Jira:RHEL-35993[1]

RHEL 10 introduces MariaDB 10.11

RHEL 10 is distributed with MariaDB 10.11. Notable changes include:

- A new sys_schema feature.
- Atomic Data Definition Language (DDL) statements.
- A new GRANT ... TO PUBLIC privilege.
- Separate SUPER and READ ONLY ADMIN privileges.
- A new UUID database data type.
- Support for the Secure Socket Layer (SSL) protocol version 3; the MariaDB server now requires correctly configured SSL to start.
- Support for the natural sort order through the natural_sort_key() function.
- A new SFORMAT function for arbitrary text formatting.

- Changes to the UTF-8 charset and the UCA-14 collation.
- systemd socket activation files available in the /usr/share/ directory. Note that they are not a part of the default configuration in RHEL as opposed to upstream.
- Error messages containing the MariaDB string instead of MySQL.
- Error messages available in the Chinese language.
- Changes to the default logrotate file.
- For MariaDB and MySQL clients, the connection property specified on the command line (for example, **--port=3306**), now forces the protocol type of communication between the client and the server, such as tcp, socket, pipe, or memory.

Jira:RHELDOCS-19550[1]

6.14. COMPILERS AND DEVELOPMENT TOOLS

RHEL 10 introduces GCC 14.2

RHEL 10 is distributed with the GNU Compiler Collection (GCC) version 14.2.

Notable changes since GCC 13 include:

- Optimization and diagnostic improvements
- A new **-fhardened** umbrella option, which enables a set of hardening flags
- A new **-fharden-control-flow-redundancy** option to detect attacks that transfer control into the middle of functions
- A new **strub** type attribute to control stack scrubbing properties of functions and variables
- A new **-finline-stringops** option to force inline expansion of certain **mem*** functions
- Support for new OpenMP 5.1, 5.2, and 6.0 features
- Several new C23 features
- Multiple new C++23 and C++26 features
- Several resolved C++ defect reports
- New and improved experimental support for C++20, C++23, and C++26 in the C++ library
- Support for new CPUs in the 64-bit ARM architecture
- Multiple new instruction set architecture (ISA) extensions in the 64-bit Intel architecture, for example: AVX10.1, AVX-VNNI-INT16, SHA512, and SM4
- New warnings in the GCC's static analyzer
- Certain warnings changed to errors; for details, see Porting to GCC 14
- Various bug fixes

For more information about changes in GCC 14, see the upstream GCC release notes .

Jira:RHEL-45041

GCC 14 defaults to x86-64-v3

GCC 14 in RHEL 10 defaults to the x86-64-v3 microarchitecture level. This level enables certain capabilities by default, such as the AVX and AVX2 instruction sets and the fused multiply-add (FMA) instruction set. See the related article for more details.

Jira:RHEL-33254

GCC defaults to using the IEEE128 floating point format on IBM Power Systems

In RHEL10, GCC uses the **IEEE128** floating point format by default for all long double floating point numbers on IBM Power Systems instead of the earlier software-only **IBM-DOUBLE-DOUBLE** code. As a result, you can notice performance improvements in C or C++ code that performs computations by using long double floating point numbers.

Note that this 128-bit long double floating point ABI is incompatible with the floating point ABI used in RHEL 8 and earlier versions. Support for hardware instructions to perform **IEEE128** operations is available since IBM POWER9.

Jira:RHEL-24760[1]

GCC 14 supports the FUJITSU-MONAKA CPU

Starting with RHEL 10.0, the GNU Compiler Collection (GCC) supports the FUJITSU-MONAKA. As a result, you can use the **-mcpu=fujitsu-monaka** command-line option to create code for this platform.

Jira:RHEL-65765[1]

GCC 14 supports the POWER 11 architecture

Starting with RHEL 10.0, the GNU Compiler Collection (GCC) supports the POWER 11 architecture. As a result, you can use the **-mcpu=power11** command-line option to create code for POWER 11.

Jira:RHEL-24762[1]

RHEL 10 includes annobin version 12.55

RHEL 10 is distributed with **annobin** version 12.55. Notable changes over the previously available version 12.32 include:

- Updated tools to build and work with newer versions of the GCC, Clang, LLVM, and Go compilers
- Recording and testing for the use of the GCC command-line options **-Wimplicit-int** and **-Wimplicit-function-declaration**
- Improved support for LLVM
- New tests
- A new check to identify if the deprecated OpenSSL Engine code is used
- Multiple **--debug-rpm** options are now supported

- Various bug fixes

Jira:RHEL-526[1]

RHEL 10 includes binutils version 2.41

RHEL 10 is distributed with **binutils** version 2.41. Notable changes over the previously available version 2.40 include:

- **binutils** tools support architecture extensions in the 64-bit Intel and ARM architectures.
- The linker now accepts the **--remap-inputs <PATTERN>=<FILE>** command-line option to replace any input file that matches **<PATTERN>** with **<FILE>**. In addition, you can use the **--remap-inputs-file=<FILE>** option to specify a file containing any number of these remapping directives.
- For ELF targets, you can use the linker command-line option **--print-map-locals** to include local symbols in a linker map.
- For most ELF-based targets, you can use the **--enable-linker-version** option to insert the version of the linker as a string into the **.comment** section.
- The linker script syntax has a new command for output sections, **ASCIZ "<string>"**, which inserts a zero-terminated string at the current location.
- You can use the new **-z nosectionheader** linker command-line option to omit ELF section header.

Jira:RHELDOCS-18761[1]

GCC can generate ROP protection instructions for Power 10 or later

The IBM Power 10 and later platforms have a protection against Return-Oriented Programming (ROP), which is a common primitive used to exploit vulnerabilities in programs. With this enhancement, you can use the **{{-mrop-protect}}** flag and GCC creates ROP protection instructions for these platforms. Note that, because there is no runtime support, the generated instructions have currently no effect, and the CPU treats them as no operation (NOP) instructions. However, developers can use the **{{-mrop-protect}}** flag to incorporate ROP protection mechanisms so that, in future, when ROP protection is enabled for these platforms, the applications will be more secure.

Jira:RHEL-36791[1]

binutils now supports the arch15 extension of the IBM Z instruction set

With this enhancement, **binutils** supports the **arch15** extensions of CPUs on the IBM Z platform. Developers can now use the new features provided by the **arch15** extension in assembler source files or, when an updated compiler is available, also in compiled programs. This can result in smaller and faster programs.

Jira:RHEL-56896[1]

The ld linker of binutils supports the --section-ordering-file option

You can now use the new **--section-ordering-file** command-line option with **ld.bfd**, the default system linker, to group sections of code or data that can benefit from being in proximity to each other.

This feature improves performance of programs by reducing cache misses. You can use profiling tools to analyze use of your program's code over time, and then improve code grouping in the executable image. As a result, you have more control over the layout of your programs in memory.

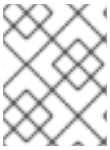
The **--section-ordering-file** option also enhances compatibility with the **gold** and **lld** linkers, which already provide this feature.

For details, see the blog post A practical guide to linker section ordering .

Jira:RHEL-36305

glibc now supports dynamic linking of Intel APX-enabled functions

An incompatible dynamic linker trampoline was identified as a potential source of incompatibilities for Intel Advanced Performance Extensions (APX) applications. As a workaround, it was possible to use the **BIND_NOW** executable or use only the standard calling convention. With this update, the dynamic linker of **glibc** preserves APX-related registers.



NOTE

Because of this change, additional space is needed beyond the top of the stack. Users who strictly limit this space might need to adjust or evaluate the stack limits.

Jira:RHEL-25045

RHEL 10 provides glibc version 2.39

RHEL 10 introduces GNU C Library (**glibc**) version 2.39.

Jira:RHEL-25850

Optimization of AMD Zen 3 and Zen 4 performance in glibc

Previously, AMD Zen 3 and Zen 4 processors sometimes used the Enhanced Repeat Move String (ERMS) version of the **memcpy** and **memmove** library routines regardless of the most optimal choice. With this update to **glibc**, AMD Zen 3 and Zen 4 processors use the most optimal versions of **memcpy** and **memmove**.

Jira:RHEL-25530

RHEL 10 provides GDB version 14.2

GDB has been updated to version 14.2. The following paragraphs list notable changes since GDB 12.1.

General:

- The **info breakpoints** command now displays enabled breakpoint locations of disabled breakpoints as in the **y-** state.
- Added support for debug sections compressed with Zstandard (**ELFCOMPRESS_ZSTD**) for ELF.
- The Text User Interface (TUI) no longer styles the source and assembly code highlighted by the current position indicator by default. To re-enable styling, use the new command **set style tui-current-position**.

- A new **$_inferior_thread_count** convenience variable contains the number of live threads in the current inferior.
- For breakpoints with multiple code locations, GDB now prints the code location using the **<breakpoint_number>.<location_number>** syntax.
- When a breakpoint is hit, GDB now sets the **$_hit_bpnum** and **$_hit_locno** convenience variables to the hit breakpoint number and code location number. You can now disable the last hit breakpoint by using the **disable $_hit_bpnum** command, or disable only the specific breakpoint code location by using the **disable $_hit_bpnum.$_hit_locno** command.
- Added support for the **NO_COLOR** environment variable.
- Added support for integer types larger than 64 bits.
- You can use new commands for multi-target feature configuration to configure remote target feature sets (see the **set remote <name>-packet** and **show remote <name>-packet** in Commands).
- Added support for the Debugger Adapter Protocol.
- You can now use the new **inferior** keyword to make breakpoints inferior-specific (see **break** or **watch** in Commands).
- You can now use the new **$_shell()** convenience function to run a shell command during expression evaluation.

Changes to existing commands:

- **break**, **watch**
  - Using the **thread** or **task** keywords multiple times with the **break** and **watch** commands now results in an error instead of using the thread or task ID of the last instance of the keyword.
  - Using more than one of the **thread**, **task**, and **inferior** keywords in the same **break** or **watch** command is now invalid.
- **printf**, **dprintf**
  - The **printf** and **dprintf** commands now accept the **%V** output format, which formats an expression the same way as the **print** command. You can also modify the output format by using additional print options in brackets **[...]** following the command, for example: **printf "%V[-array-indexes on]", <array>**.
- **list**
  - You can now use the **.** argument to print the location around the point of execution in the current frame, or around the beginning of the **main()** function if the inferior has not started yet.
  - Attempting to list more source lines in a file than are available now issues a warning, referring the user to the **.** argument.
- **document user-defined**
  - It is now possible to document user-defined aliases.

New commands:

- **set print nibbles [on|off]** (default: **off**), **show print nibbles** - controls whether the **print/t** command displays binary values in groups of four bits (nibbles).
- **set debug infcall [on|off]** (default: **off**), **show debug infcall** - prints additional debug messages about inferior function calls.
- **set debug solib [on|off]** (default: **off**), **show debug solib** - prints additional debug messages about shared library handling.
- **set print characters <LIMIT>**, **show print characters**, **print -characters <LIMIT>** - controls how many characters of a string are printed.
- **set debug breakpoint [on|off]** (default: **off**), **show debug breakpoint** - prints additional debug messages about breakpoint insertion and removal.
- **maintenance print record-instruction [N]** - prints the recorded information for a given instruction.
- **maintenance info frame-unwinders** - lists the frame unwinders currently in effect in the order of priority (highest first).
- **maintenance wait-for-index-cache** - waits until all pending writes to the index cache are completed.
- **info main** - prints information on the main symbol to identify an entry point into the program.
- **set tui mouse-events [on|off]** (default: **on**), **show tui mouse-events** - controls whether mouse click events are sent to the TUI and Python extensions (when **on**), or the terminal (when **off**).

Machine Interface (MI) changes:

- MI version 1 has been removed.
- MI now reports **no-history** when reverse execution history is exhausted.
- The **thread** and **task** breakpoint fields are no longer reported twice in the output of the **-break-insert** command.
- Thread-specific breakpoints can no longer be created on non-existent thread IDs.
- The **--simple-values** argument to the **-stack-list-arguments**, **-stack-list-locals**, **-stack-list-variables**, and **-var-list-children** commands now considers reference types as simple if the target is simple.
- The **-break-insert** command now accepts a new **-g thread-group-id** option to create inferior-specific breakpoints.
- Breakpoint-created notifications and the output of the **-break-insert** command can now include an optional **inferior** field for the main breakpoint and each breakpoint location.
- The async record stating the **breakpoint-hit** stopped reason now contains an optional field **locno** giving the code location number in case of a multi-location breakpoint.

Changes in the GDB Python API:

- Events
  - A new **gdb.ThreadExitedEvent** event.
  - A new **gdb.executable_changed** event registry, which emits the **ExecutableChangedEvent** objects that have **progspace** and **reload** attributes.
  - New **gdb.events.new_progspace** and **gdb.events.free_progspace** event registries, which emit the **NewProgpspaceEvent** and **FreeProgspaceEvent** event types. Both of these event types have a single attribute **progspace** to specify the **gdb.Progspace** program space that is being added to or removed from GDB.
- The **gdb.unwinder.Unwinder** class
  - The **name** attribute is now read-only.
  - The name argument of the **__init__** function must be of the **str** type, otherwise a **TypeError** is raised.
  - The **enabled** attribute now accepts only the **bool** type.
- The **gdb.PendingFrame** class
  - New methods: **name**, **is_valid**, **pc**, **language**, **find_sal**, **block**, and **function**, which mirror similar methods of the **gdb.Frame** class.
  - The **frame-id** argument of the **create_unwind_info** function can now be either an integer or a **gdb.Value** object for the **pc**, **sp**, and **special** attributes.
- A new **gdb.unwinder.FrameId** class, which can be passed to the **gdb.PendingFrame.create_unwind_info** function.
- The **gdb.disassembler.DisassemblerResult** class can no longer be sub-classed.
- The **gdb.disassembler** module now includes styling support.
- A new **gdb.execute_mi(COMMAND, [ARG]…)** function, which invokes a GDB/MI command and returns result as a Python dictionary.
- A new **gdb.block_signals()** function, which returns a context manager that blocks any signals that GDB needs to handle.
- A new **gdb.Thread** subclass of the **threading.Thread** class, which calls the **gdb.block_signals** function in its **start** method.
- The **gdb.parse_and_eval** function has a new **global_context** parameter to restrict parsing on global symbols.
- The **gdb.Inferior** class
  - A new **arguments** attribute, which holds the command-line arguments to the inferior, if known.
  - A new **main_name** attribute, which holds the name of the inferior's **main** function, if known.

  - New **clear_env**, **set_env**, and **unset_env** methods, which can modify the inferior's environment before it is started.

- The **gdb.Value** class
  - A new **assign** method to assign a value of an object.
  - A new **to_array** method to convert an array-like value to an array.
- The **gdb.Progspace** class
  - A new **objfile_for_address** method, which returns the **gdb.Objfile** object that covers a given address (if exists).
  - A new **symbol_file** attribute holding the **gdb.Objfile** object that corresponds to the **Progspace.filename** variable (or **None** if the filename is **None**).
  - A new **executable_filename** attribute, which holds the string with a filename that is set by the **exec-file** or **file** commands, or **None** if no executable file is set.
- The **gdb.Breakpoint** class
  - A new **inferior** attribute, which contains the inferior ID (an integer) for breakpoints that are inferior-specific, or **None** if no such breakpoints are set.
- The **gdb.Type** class
  - New **is_array_like** and **is_string_like** methods, which reflect whether a type might be array- or string-like regardless of the type's actual type code.
- A new **gdb.ValuePrinter** class, which can be used as the base class for the result of applying a pretty-printer.
- A newly implemented **gdb.LazyString.__str__** method.
- The **gdb.Frame** class
  - A new **static_link** method, which returns the outer frame of a nested function frame.
  - A new **gdb.Frame.language** method that returns the name of the frame's language.
- The **gdb.Command** class
  - GDB now reformats the doc string for the **gdb.Command** class and the **gdb.Parameter** sub-classes to remove unnecessary leading whitespace from each line before using the string as the help output.
- The **gdb.Objfile** class
  - A new **is_file** attribute.
- A new **gdb.format_address(ADDRESS, PROGSPACE, ARCHITECTURE)** function, which uses the same format as when printing address, symbol, and offset information from the disassembler.
- A new **gdb.current_language** function, which returns the name of the current language.

- A new Python API for wrapping GDB's disassembler, including **gdb.disassembler.register_disassembler(DISASSEMBLER, ARCH)**, **gdb.disassembler.Disassembler**, **gdb.disassembler.DisassembleInfo**, **gdb.disassembler.builtin_disassemble(INFO, MEMORY_SOURCE)**, and **gdb.disassembler.DisassemblerResult**.
- A new **gdb.print_options** function, which returns a dictionary of the prevailing print options, in the form accepted by the **gdb.Value.format_string** function.
- The **gdb.Value.format_string** function
  - **gdb.Value.format_string** now uses the format provided by the **print** command if it is called during a **print** or other similar operation.
  - **gdb.Value.format_string** now accepts the **summary** keyword.
- A new **gdb.BreakpointLocation** Python type.
- The **gdb.register_window_type** method now restricts the set of acceptable window names.

Architecture-specific changes:

- AMD and Intel 64-bit architectures
  - Added support for disassembler styling using the **libopcodes** library, which is now used by default. You can modify how the disassembler output is styled by using the **set style disassembler *** commands. To use the Python Pygments styling instead, use the new **maintenance set libopcodes-styling off** command.
- The 64-bit ARM architecture
  - Added support for dumping memory tag data for the Memory Tagging Extension (MTE).
  - Added support for the Scalable Matrix Extension 1 and 2 (SME/SME2). Some features are still considered experimental or alpha, for example, manual function calls with ZA state or tracking Scalable Vector Graphics (SVG) changes based on DWARF.
  - Added support for Thread Local Storage (TLS) variables.
  - Added support for hardware watchpoints.
- The 64-bit IBM Z architecture
  - Record and replay support for the new **arch14** instructions on IBM Z targets, except for the specialized-function-assist instruction **NNPA**.
- IBM Power Systems, Little Endian
  - Added base enablement support for POWER11.

For changes since the RHEL 9 system version of GDB 10.2, see the release notes for the GCC Toolset 12 version of GDB 11.2 and the GCC Toolset 13 version of GDB 12.1.

Jira:RHEL-33256, Jira:RHEL-39324, Jira:RHEL-24764

RHEL 10 provides elfutils version 0.191

The **elfutils** package has been updated to version 0.191. Notable improvements include:

- Changes in the **libdw** library:
  - The **dwarf_addrdie** function now supports binaries lacking a **debug_aranges** section.
  - Support for DWARF package files has been improved.
  - A new **dwarf_cu_dwp_section_info** function has been added.
- Caching eviction logic in the **debuginfod** server has been enhanced to improve retention of small, frequent, or slow files, such as **vdso.debug**.
- The **eu-srcfiles** utility can now fetch the source files of a DWARF/ELF file and place them into a **zip** archive.

Jira:RHEL-29197

RHEL 10 provides SystemTap version 5.1

RHEL 10 includes the **SystemTap** tracing and probing tool version 5.1. Notable changes since version 5.0 include:

- An experimental **--build-as=USER** flag to reduce privileges during script compilation.
- Improved support for probing processes running in containers, identified by host PID.
- New probes for userspace hardware breakpoints and watchpoints.
- Support for the **--remote** operation of **--runtime=bpf** mode.
- Improved robustness of kernel-user transport.

Jira:RHEL-29529

RHEL 10 provides Valgrind version 3.23.0

The **Valgrind** suite has been updated to version 3.23.0. Notable enhancements include:

- The **--track-fds=yes** option now warns against double closing of file descriptors, generates suppressible errors, and supports XML output.
- The **--show-error-list=no|yes** option now accepts a new value, **all**, to also print the suppressed errors.
- On the 64-bit IBM Z architecture, **Valgrind** now supports neural network processing assist (NNPA) facility vector instructions: **VCNF**, **VCLFNH**, **VCFN**, **VCLFNL**, **VCRNF**, and **NNPA** (z16/arch14).
- On the 64-bit ARM architecture, **Valgrind** now supports **dotprod** instructions (**sdot**/**udot**).
- On the AMD and Intel 64-bit architectures, **Valgrind** now provides more accurate instruction support for the x86_64-v3 microarchitecture.
- **Valgrind** now provides wrappers for the **wcpncpy**, **memccpy**, **strlcat**, and **strlcpy** functions that can detect memory overlap.

- **Valgrind** now supports the following Linux syscalls: **mlock2**, **fchmodat2**, and **pidfd_getfd**.

Jira:RHEL-29535

RHEL 10 introduces Dyninst version 12.3.0

RHEL 10 is distributed with the **Dyninst** library version 12.3.0.

Jira:RHEL-49597[1]

SystemTap provided in version 5.2

RHEL 10.0 provides the **SystemTap** tracing and probing tool in version 5.2.

A notable enhancement is the full activation of **debuginfod-metadata** based probes, based on **elfutils** 0.192. With this feature, you can write a **systemtap** script to target a full range of versions of a given binary or library by searching a **debuginfod** server for all matching names.

Jira:RHEL-64042

RHEL 10 introduces debugedit 5.1

RHEL 10 is distributed with **debugedit** 5.1. The most notable changes are:

- The **debugedit** utility now uses the faster **xxhash** algorithm to generate the **buildid**.
- The **find-debuginfo** utility supports the following new options:
- **-v** and **--verbose** to add more output for all files processed
- **-q** and **--quiet** to silence all non-error output
- The **find-debuginfo** utility now passes the **-j** option also to the **dwz** tool, which enables parallelized processing.
- The **debugedit** utility now handles compressed DWARF debugging ELF sections.
- The **debugedit** utility now handles more DWARF5 constructs as used by the **clang** compiler.

Jira:RHEL-64137

RHEL 10 provides elfutils version 0.192

The **elfutils** package has been updated to version 0.192. Notable improvements include:

- **debuginfod**:
  - Added per-file signature verification for integrity checking, by using the RPM IMA scheme from Fedora and RHEL.
  - New API for metadata queries: file name → **buildid**.
  - The server-side extraction of files from kernel **debuginfo** packages is significantly faster. It takes now less than 0.25 seconds, down from ~50 seconds.
- **libdw**:

    - New functions **dwfl_set_sysroot**, **dwfl_frame_unwound_source**, and **dwfl_unwound_source_str**.

- **stacktrace**:

    - Experimental new tool that can process a stream of stack samples from the **Sysprof** profiler and unwind them into call chains. Enable on x86 with **--enable-stacktrace**. See the **README.eu-stacktrace** file in the development branch for detailed usage instructions.

    - The **eu-stacktrace** utility is available as a Technology Preview. For details, see **eu-stacktrace** available as a Technology Preview .

Jira:RHEL-64046

RHEL 10 provides libabigail 2.6

RHEL 10 provides version 2.6 of the **libabigail** library. Notable changes include:

- Better support for Linux kernel module analysis by using the BPF Type Format (BTF) and Common Trace Format (CTF).
- Improved internal type comparison algorithms in the middle end.
- Improved logging in **abipkgdiff**, **abidw**, and **abilint** utilities
- Numerous bug fixes.

For further changes, see the upstream release notes.

Jira:RHEL-64063

valgrind provided in version 3.24.0

RHEL 10.0 provides the **valgrind** suite in version 3.24.0. Notable enhancements include:

- The **--track-fds=yes** option now shows suppressible errors when using bad file descriptors, and the errors are written to the XML output. The warnings shown, if you do not use the option, are deprecated and will be removed in a future version.
- Error messages now support Ada name demangling.
- The **deflate-conversion** facility (z15/arch13) now supports the deflate compression call (DFLTCC) instruction on the IBM Z platform.
- On the IBM Z platform, **valgrind** now supports the instructions provided by the message security assist (MSA) facility and its 1-9 extensions.
- **Valgrind** now supports the following new Linux system calls:
    - **open_tree**
    - **move_mount**
    - **fsopen**
    - **fsconfig**

- **fsmount**
- **fspick**
- **landlock_create_ruleset**
- **landlock_add_rule**
- **landlock_restrict_self**

Jira:RHEL-64056

Go Toolset provided in version 1.23

RHEL 10.0 provides Go Toolset in version 1.23. Notable enhancements include:

- The **for-range** loop accepts iterator functions of the following types:
  - **func(func() bool)**
  - **func(func(K) bool)**
  - **func(func(K, V) bool)**
    Calls of the iterator argument function create the iteration values for the **for-range** loop. For reference links, see the upstream release notes.
- The Go Toolchain can collect usage and breakage statistics to help the Go team to understand how the Go Toolchain is used and working. By default, Go Telemetry does not upload telemetry data and stores it only locally. For further information, see the upstream Go Telemetry documentation.
- The **go vet** subcommand includes the **stdversion** analyzer which flags references to symbols that are too new for the version of Go you use in the referring file.
- The **cmd** and **cgo** features support the **-ldflags** option to pass flags to the C linker. The **go** command uses this flag automatically to avoid **argument list too long** errors when you use a very large **CGO_LDFLAGS** environment variable.
- The **trace** utility tolerates partially broken traces and attempts to recover the trace data. This is especially useful in case of crashes, because you can get the trace leading up to the crash.
- The traceback printed by the runtime after an unhandled panic or other unrecoverable error carries indentation to distinguish the stack trace of the **goroutine** from the first **goroutine**.
- The compiler build time overhead of using profile-guided optimization was reduced to single-digit percentage.
- The new **-bindnow** linker flag enables immediate function binding when building a dynamically-linked ELF binary.
- The //**go:linkname** linker directive no longer refer to internal symbols in the standard library and the runtime that are not marked with //**go:linkname** on their definition.
- If a program no longer refers to a **Timer** or **Ticker**, garbage collection cleans them up immediately even if their **Stop** method has not been called. The timer channel associated with a **Timer** or **Ticker** is now unbuffered with capacity 0. This ensures that, every time a **Reset** or

Stop method is called, no stale values are not sent or received after the call.

- The new **unique** package provides facilities for canonicalizing values, such as **interning** or **hash-consing**.
- The new **iter** package provides the basic definitions to work with user-defined iterators.
- The **slices** and **maps** packages introduce several new functions that work with iterators.
- The new **structs** package provides types for struct fields that modify properties of the containing struct type, such as memory layout.
- Minor changes are made in the following packages:
    - **archive/tar**
    - **crypto/tls**
    - **crypto/x509**
    - **database/sql**
    - **debug/elf**
    - **encoding/binary**
    - **go/ast**
    - **go/types**
    - **math/rand/v2**
    - **net**
    - **net/http**
    - **net/http/httptest**
    - **net/netips**
    - **path/filepath**
    - **reflect**
    - **runtime/debug**
    - **runtime/pprof**
    - **runtime/trace**
    - **slices**
    - **sync**
    - **sync/atomic**

- **syscall**
- **testing/fstest**
- **text/template**
- **time**
- **unicode/utf16**

For more information, see the upstream release notes.

Go Toolset is a rolling Application Stream, and Red Hat supports only the latest version. For more information, see the Red Hat Enterprise Linux Application Streams Life Cycle document.

Jira:RHEL-34260

RHEL 10 introduces LLVM Toolset 19.1.7

RHEL 10 is distributed with the LLVM Toolset version 19.1.7.

Notable changes of the LLVM compiler:

- LLVM now uses debug records, a more efficient representation for debug information.

Notable updates of the Clang:

- C++14 sized deallocation is now enabled by default.
- C++17 support has been completed.
- Improvements to C++20 support, especially around modules, concepts, and Class Template Argument Deduction (CTAD) have been added.
- Improvements to C++23, C++2c, C23, and C2y support have been added.

For more information, see the LLVM release notes and Clang release notes.

LLVM Toolset is a rolling Application Stream, and only the latest version is supported. For more information, see the Red Hat Enterprise Linux Application Streams Life Cycle document.

Jira:RHEL-57456

RHEL 10.0 includes Rust Toolset version 1.84.1

RHEL 10.0 is distributed with the Rust Toolset version 1.84.1. Notable enhancements since the previously available version 1.79.0 include:

- The new **LazyCell** and **LazyLock** types delay the initialization until the first use. These extend the earlier **OnceCell** and **OnceLock** types with the initialization function included in each instance.
- The new sort implementations in the standard library improve the runtime performance and compile times. They also try to detect cases where a comparator is not producing a total order, making that panic instead of returning unsorted data.
- Precise capturing for opaque return types have been added. The new **use<..>** syntax specifies the generic parameters and lifetimes used in an **impl Trait** return type.

- Many new features for **const** code have been added, for example:
  - Floating point support
  - **const** immediates for inline assembly
  - References to statics
  - Mutable reference and pointers
- Many new features for **unsafe** code have been added, for example:
  - Strict provenance APIs
  - **&raw** pointer syntax
  - Safely addressing statics
  - Declaring safe items in unsafe **extern** blocks
- The Cargo dependency resolver is now version aware. If a dependency crate specifies its minimum supported Rust version, Cargo uses this information when it resolves the dependency graph instead of using the latest **semver**-compatible crate version.

Compatibility notes:

- The WebAssembly System Interface (WASI) target is changed from **rust-std-static-wasm32-wasi** to **rust-std-static-wasm32-wasip1**. You can select the WASI target also by using the **--target wasm32-wasip1** parameter on the command line. For more information, see the Changes to Rust's WASI targets upstream blog post.
- The split panic hook and panic handler arguments **core::panic::PanicInfo** and **std::panic::PanicInfo** are now different types.
- **extern "C"** functions stops on uncaught panics. Use **extern "C-unwind"** instead to allow unwinding across ABI boundaries.

Rust Toolset is a rolling Application Stream, and Red Hat only supports the latest version. For more information, see the Red Hat Enterprise Linux Application Streams Life Cycle document.

Jira:RHEL-59689[1]

RHEL 10 includes PCP version 6.3.0

RHEL 10 is distributed with Performance Co-Pilot (PCP) version 6.3.0. Notable changes over the previously available version 6.2.0 include:

New tools and agents

- **pcp2openmetrics**: a new tool to push PCP metrics in Open Metrics format to remote end points
- **pcp-geolocate**: a new tool to report latitude and longitude metric labels
- **pmcheck**: a new tool to interrogate and control PCP components
- **pmdauwsgi**: a new PCP agent that exports instrumentation from uWSGI servers

Enhanced tools

- **pmdalinux**: added new kernel metrics (hugepages, filesystems, TCP, softnet, virtual machine balloon)
- **pmdalibvirt**: added support for metric labels, added new balloon, vCPU, and domain info metrics
- **pmdabpf**: improved eBPF networking metrics for use with the **pcp-atop** utility

Jira:RHELDOCS-18787[1]

RHEL 10 provides Grafana version 10.2.6

The **Grafana** platform has been updated to version 10.2.6.

Notable enhancements include:

- Support for zooming in on the y axis of time series and candlestick visualizations by holding shift while clicking and dragging.
- Streamlined data source selection when creating a dashboard.
- Updated User Interface, including updates to navigation and the command palette.
- Various improvements to transformations, including the new unary operation mode for the **Add field from calculation** transformation.
- Various improvements to dashboards and data visualizations, including a redesigned empty dashboard and dashboard panel.
- New geomap and canvas panels.

Other changes:

- Various improvements to users, access, authentication, authorization, and security.
- Alerting improvements along with new alerting features.
- Public dashboards now available.

For a complete list of changes since the previously available **Grafana** version 9.2, see the upstream documentation.

Jira:RHEL-35761

RHEL 10 provides grafana-pcp in version 5.2.2

RHEL 10 is distributed with the **grafana-pcp** plugin version 5.2.2. Notable changes include:

- The plugin now uses Valkey as a data source instead of Redis. As a consequence, the **PCP Redis** data source was renamed to **PCP Valkey**.
- New dashboards:
  - **PCP Vector Top Consumers**
  - **PCP Vector UWSGI overview**

- The metric search is unavailable until a replacement for the RediSearch module is available for the Valkey data source.

Jira:RHEL-67043

Grafana, PCP, and grafana-pcp now use Valkey to store data

In RHEL 10, the **Valkey** key-value store replaces **Redis**. As a result, **Grafana**, PCP, and the **grafana-pcp** plug-in now use **Valkey** to store data instead of **Redis**. The **PCP Redis** data source in the **grafana-pcp** plug-in is now named **PCP Valkey**.

Jira:RHEL-45646

zlib-ng-compat replaces zlib in RHEL 10

The new **zlib-ng-compat** package provides a general-purpose lossless data compression library that is used by many different programs. This implementation provides various benefits over **zlib** distributed in RHEL 9. For example, **zlib-ng-compat** supports hardware acceleration when available and enhances compression efficiency and performance. **zlib-ng-compat** is built-in API and ABI compatible mode to ensure a smooth transition from **zlib**.

Jira:RHEL-24058[1]

SWIG 4.3.0 available in the CRB repository

The Simplified Wrapper and Interface Generator (SWIG) version 4.2.1 is now available in the CodeReady Linux Builder (CRB) repository. Notable changes include:

- Python Standard Template Library (STL) container wrappers now use the Python Iterator Protocol.
- SWIG now supports:
  - Python stable Application Binary Interface (ABI)
  - Python 3.12 and Python 3.13
  - Ruby 3.2 and Ruby 3.3
  - Tcl 9.0
  - PHP 8; support for PHP 7 has been removed.
- Support for the C++14 auto variable without trailing return type for the C++11 auto variable has been added.
- Constructors, destructors, and assignment operators have been fixed, including implicit, default, and deleted, and related non-assignable variable wrappers.
- A new Javascript generator targeting Node.js binary stable ABI Node-API is now available.
- Multiple deprecated features have been removed.
- Experimental support for C as a target language has been added.
- Handling of namespaces when using the **nspace** feature has been enhanced.

- The STL wrapper has been enhanced for the **std::unique_ptr**, **std::string_view**, **std::filesystem objects**.
- Support for C++17 fold expressions and C++11 trailing return types has been added.
- Handling of string and character literals has been improved.

Note that packages included in the CodeReady Linux Builder repository are unsupported.

Jira:RHELDOCS-19059[1]

Red Hat build of OpenJDK 21 is the default Java implementation in RHEL 10

The default RHEL 10 Java implementation is OpenJDK 21. Use the **java-21-openjdk** packages, which provide the OpenJDK 21 Java Runtime Environment and the OpenJDK 21 Java Software Development Kit. For more information, see the OpenJDK documentation.

Jira:RHEL-51248

Clang and LLVM now support zstd for debug section compression

By default, Clang and LLVM tools use **Zlib** as the algorithm for debug section compression. With this enhancement, users can alternatively use the Zstandard (**zstd**) algorithm which can reach a higher compression rate than **Zlib**.

For example, if you want to use **zstd** compression when you compile a program with Clang, use the following command:

```
$ clang -Wa,-compress-debug-sections=zstd -Wl,--compress-debug-sections=zstd ...
```

Jira:RHEL-70325

The llvm-doc package now contains only a reference to the upstream documentation.

In previous versions, the **llvm-doc** package contained the LLVM documentation in HTML format. With this update, the package provides only the **/usr/share/doc/llvm/html/index.html** file which contains a reference to the upstream documentation.

Jira:RHEL-58900

RHEL 10 provides cmake in version 3.30.5

RHEL 10 is distributed with **cmake** version 3.30.5. For notable changes, see the upstream release notes.

Jira:RHEL-65234

RHEL 10 provides .NET in versions 9.0 and 8.0

The most recent version of .NET (9.0) and the current long-term support of .NET (8.0), a general-purpose development platform featuring automatic memory management and modern programming languages, are supported on Red Hat Enterprise Linux (RHEL) 10. Using .NET, you can build high-quality applications efficiently.

For details on installation and usage, see the documentation for .NET 9.0 and .NET 8.0.

Jira:RHELDOCS-20066[1]

RHEL 10 provides Go Toolset in version 1.24.4

Go Toolset has been updated to version 1.24.4 with the release of the RHSA-2025:10677 advisory.

Notable enhancements and changes include:

- **Language**:
  - Generic type aliases are now fully supported, allowing type aliases to be parameterized for increased flexibility with generics.
- **Tools**:
  - The Go module system supports **tool** directives in **go.mod** files, enabling direct management of executable dependencies.
  - The **go build**, **go install**, and **go test** commands now support the **-json** flag for structured output.
  - The new **GOAUTH** environment variable provides enhanced authentication for private modules.
- **Runtime and Performance**:
  - Runtime improvements reduce CPU overhead by 2–3% on average.
  - Notable changes include a new map implementation based on Swiss Tables and more efficient memory allocation.
- **Standard Library**:
  - The new **os.Root** type enables directory-limited filesystem access.
  - The **testing.B.Loop** method improves benchmarking.
  - The **runtime.AddCleanup** function provides a more flexible finalization mechanism.
  - The new **weak** package introduces weak pointers.
- **Cryptography**:
  - New packages for ML-KEM post-quantum key exchange (**crypto/mlkem**), HKDF, PBKDF2, and SHA-3 are now available.
  - The Go Cryptographic Module is now under review for FIPS 140-3 certification.
- **Additional updates**:
  - The **vet** tool includes a new analyzer for detecting common mistakes in tests and examples.
  - The objdump tool now supports more architectures.
  - **Cgo** introduces annotations for improved performance and correctness.

For more information, see the upstream release notes.

Go Toolset is a rolling Application Stream, and Red Hat supports only the latest version. For more information, see the Red Hat Enterprise Linux Application Streams Life Cycle document.

Jira:RHEL-101075[1]

IBM Semeru JDK is available in RHEL 10

IBM Semeru is a Technology Compatibility Kit (TCK) certified Java Runtime Environment (JRE) implementation included with Red Hat Enterprise Linux (RHEL) 10 and maintained by IBM. The **java-21-ibm-semeru-certified-jdk-devel** package is available in the RHEL Supplementary repository as an alternative to Open Java Development Kit (OpenJDK) distribution, which is available in the RHEL AppStream repository.

Jira:RHELDOCS-20591[1]

6.15. IDENTITY MANAGEMENT

RHEL 10 provides python-jwcrypto version 1.5.6

The **python-jwcrypto** package has been updated to version 1.5.6. This version includes a security fix to an issue where an attacker could cause a denial of service attack by passing in a malicious JWE Token with a high compression ratio.

Jira:RHELDOCS-20100[1]

RHEL 10 provides ansible-freeipa package version 1.14.5

The **ansible-freeipa** package has been updated to version 1.14.5. Notable enhancements and bug fixes include:

- You can use **module_defaults** to define variables for multiple **ansible-freeipa** tasks
  The **freeipa.ansible_freeipa** collection now provides the **module_defaults** action group that simplifies the use of **ansible-freeipa** modules. By using **module_defaults**, you can set default values to be applied to all modules of the collection used in a playbook. To do so, use the **action_group** named **freeipa.ansible_freeipa.modules**. For example:

```
- name: Test
  hosts: localhost
  module_defaults:
    group/freeipa.ansible_freeipa.modules:
      ipaadmin_password: Secret123
  tasks:
…
```

  As a result, the playbook is more concise.

- Multiple IdM **sudo** rules can now be managed in a single Ansible task
  With this enhancement in **ansible-freeipa**, you can add, modify, and delete multiple Identity Management (IdM) **sudo** rules by using a single Ansible task. To do this, use the **sudorules** option of the **ipasudorule** module. As a result, you can define your **sudo** rules more easily, and execute them more efficiently.

  Using the **sudorules** option, you can specify multiple **sudo** rule parameters that apply to a particular **sudo** rule. This **sudo** rule is defined by the **name** variable, which is the only mandatory variable for the **sudorules** option.

- Removing external members by using the **ipagroup** module now works correctly

Previously, attempting to ensure the absence of an external member from an IdM group by using the **ansible-freeipa ipagroup** module with the **externalmember** parameter did not remove the members from the group, even though Ansible presented the result of the task as **changed**. With this fix, using the **ipagroup** module with **externalmember** correctly ensures the absence of an external member from an IdM group. The fix also allows the use of either DOM\name or name@domain to identify AD users.

Jira:RHEL-67567

New tool to manage IdM ID range inconsistencies

With this update, Identity Management (IdM) provides the **ipa-idrange-fix** tool. You can use **ipa-idrange-fix** tool to analyze existing IdM ID ranges, identify users and groups outside these ranges, and propose to create new **ipa-local** ranges to include them.

The **ipa-idrange-fix** tool performs the following:

- Read and analyze existing ranges from LDAP.
- Search for users and groups outside of **ipa-local** ranges.
- Propose new **ipa-local** ranges to cover the identified users and groups.
- Prompt the user to apply the proposed changes.

By default, the tool excludes IDs below 1000 to prevent conflicts with system accounts. Red Hat strongly recommends creating a full system backup before applying any suggested changes.

For more information, see the **ipa-idrange-fix(1)** man page.

Jira:RHEL-56917[1]

Automated removal of expired certificates is enabled by default

With this update, automated removal of expired certificates is now enabled by default in Identity Management (IdM) on new replicas. A prerequisite for this is the generation of random serial numbers for certificates using RSNv3, which is now also enabled by default.

As a result, certificates are now created with random serial numbers and are removed automatically when expired, after a default retention period of 30 days after expiry.

Jira:RHEL-57674

RHEL 10 provides python-pyasn1 version 0.6.1

The **python-pyasn1** package has been updated to version 0.6.1. The update includes various enhancements and bug fixes, including:

- Support of Python 3.13
- Removed support of Python 2.7, 3.6, 3.7
- Improved error handling and consistency
- Runtime deprecation of tagMap and typeMap aliases
- Support of the previously missing RELATIVE-OID construct

Jira:RHEL-67667

The ldap_id_use_start_tls option is now enabled by default

To improve security, the default value for **ldap_id_use_start_tls** has changed from **false** to **true**. When using **ldap://** without TLS for identity lookups, it can pose a risk for an attack vector. Particularly a man-in-the-middle (MITM) attack which could allow an attacker to impersonate a user by altering, for example, the UID or GID of an object returned in an LDAP search.

As unencrypted communication is not secure, the default **ldap_id_use_start_tls** option is now set to **true**.

Jira:RHELDOCS-19185[1]

RHEL 10 provides certmonger version 0.79.20

The **certmonger** package has been updated to version 0.79.20. The update includes various bug fixes and enhancements, most notably:

- Enhanced handling of new certificates in the internal token and improved the removal process on renewal.
- Removed restrictions on tokens for **CKM_RSA_X_509** cryptographic mechanism.
- Fixed the documentation for the **getcert add-scep-ca**, **--ca-cert**, and **--ra-cert** options.
- Renamed the D-Bus service and configuration files to match canonical name.
- Added missing **.TP** tags in the **getcert-resubmit** man page.
- Migrated to the SPDX license format.
- Included owner and permissions information in the **getcert list** output.
- Removed the requirement for an NSS database in the **cm_certread_n_parse** function.
- Added translations using Webplate for Simplified Chinese, Georgian, and Russian.

Jira:RHEL-40922[1]

RHEL 10 provides python-jwcrypto in version 1.5.6

The **python-jwcrypto** package has been updated to version 1.5.6. This version includes a security fix to an issue where an attacker could cause a denial of service attack by passing in a malicious JWE Token with a high compression ratio.

Jira:RHELDOCS-19191[1]

Kerberos now supports the Elliptic Curve Diffie-Hellman key agreement algorithm

The Elliptic Curve Diffie-Hellman (ECDH) key agreement algorithm for PKINIT, as defined by RFC5349, is now supported. With this update, the **pkinit_dh_min_bits** setting in **krb5.conf`file can now be configured with `P-256**, **P-384**, or **P-521** to use ECDH by default.

Jira:RHEL-71881[1]

RHEL 10 provides 389-ds-base version 3.0.6

The **389-ds-base** package has been updated to version 3.0.6. The update includes various enhancements and bug fixes, including:

- Log buffering for the error log
- An option to write the audit log in JSON format
- An option to defer updating group members when the group is updated
- An option to configure a number of PBKDF2 iterations
- The **logconv.py** log analyzer tool

Jira:RHEL-67196

389-ds-base now fully supports LMDB

The Lightning Memory-Mapped Database (LMDB), previously available as a Technology Preview in the **389-ds-base** package, is now fully supported.

Key benefits include:

- LMDB is highly optimized for read operations.
- LMDB avoids memory allocations and memory-to-memory copies.
- LMDB requires minimal configuration.
- LMDB supports multi-threaded and multi-process environments with no deadlocks.
- Readers never block writers, and vice versa.
- LMDB does not require transaction logs.

Starting with RHEL 10, all new Directory Server instances use only LMDB as the database type, and a standard installation with BDB is no longer possible.

To migrate your existing BDB instances to LBDM, create a new LMDB instance and import the database contents by using an LDIF file or replication method.

Directory Server stores LMDB settings under the **cn=mdb,cn=config,cn=ldbm database,cn=plugins,cn=config** entry that includes the following new configuration parameters:

- **nsslapd-mdb-max-size** sets the database maximum size in bytes.
  Important: Make sure that **nsslapd-mdb-max-size** is large enough to store all intended data. However, the parameter value must not be too high to impact the performance because the database file is memory-mapped.
- **nsslapd-mdb-max-readers** sets the maximum number of read operations that can be opened at the same time. Directory Server autotunes this setting.
- **nsslapd-mdb-max-dbs** sets the maximum number of named database instances that can be included within the memory-mapped database file.

Along with the new LMDB settings, you can still use the **nsslapd-db-home-directory** database configuration parameter.

Jira:RHEL-67595

RHEL 10 provides openldap version 2.6.8

The **openldap** package has been updated to version 2.6.8. The update includes various enhancements and bug fixes, including:

- Handling of TLS connections has been improved.
- Kerberos **SASL** works with **STARTTLS** even when the Active Directory certificate is an Elliptic Curve Cryptography (ECC) certificate and **SASL_CBINDING** is set to **tls-endpoint**.

Jira:RHEL-71052

Directory Server now provides buffering of the error, audit, and audit fail logs

Before this update, only the access and security logs had log buffering. With this update, Directory Server provides buffering of the error, audit, and audit fail logs. Use the following settings to configure log buffering:

- **nsslapd-errorlog-logbuffering** for the error log. Disabled by default.
- **nsslapd-auditlog-logbuffering** for the audit and audit fail log. Enabled by default.

For details, see nsslapd-errorlog-logbuffering and nsslapd-auditlog-logbuffering in the Red Hat Directory Server Configuration and schema reference documentation.

Jira:RHEL-1681

Now you can configure hashing iterations values in PBKDF2-* Password Storage Schemes plugin entries

Before this update, the number of hashing iterations was hard-coded (**10000**) for all PBKDF2-* entries of the Password Storage Schemes plugin. With this update, the hashing iterations value is now configured by using the new **nsslapd-pwdpbkdf2numiterations** attribute that is **100000** by default.

You can configure **nsslapd-pwdpbkdf2numiterations** by using the command line or the web console.

For example, to set the value to **150000** and see the current value in different password storage schemes, run:

```
# dsconf <instance_name> plugin pwstorage-scheme pbkdf2-sha512 set-num-iterations 150000
# dsconf <instance_name> plugin pwstorage-scheme pbkdf2-sha512 get-num-iterations
```

In the web console, go to menu:[Database → Password Policies → Global Policy] to configure hashing iterations.

Consider the following before changing the default value:

- Old passwords have an old hashing iterations setting until the passwords are updated.
- An increased number of iterations can impact BIND operation performance.

Jira:RHEL-42485

dsctl healthcheck now warns about creating a substring index on the membership attribute

An entry that contains a membership attribute is usually a group with many members. When changing the value set, substring index is very expensive even for a minor change such as deleting a single member. Now, when you add the substring index type, **dsctl healthcheck** warns about possible high cost of substring index on membership attributes and displays the following error message:

DSMOLE0002. If the substring index is configured for a membership attribute, the removal of a member from the large group can be slow.

Jira:RHEL-76841

The service type of gssproxy systemd service has been changed

The **gssproxy systemd** service type has been changed from "forking" to "notify". This update removes the dependency on PIDFile, which is necessary for improved compatibility with **bootc**. With this update, the **gssproxy** service uses the "notify" type, providing more reliable service state monitoring.

Jira:RHEL-71651

ACME is now fully supported in IdM

The Automated Certificate Management Environment (ACME) service is now fully supported in Identity Management (IdM). ACME is a protocol for automated identifier validation and certificate issuance. Its goal is to improve security by reducing certificate lifetimes and avoiding manual processes from certificate lifecycle management.

In RHEL, the ACME service uses the Red Hat Certificate System (RHCS) PKI ACME responder. The RHCS ACME subsystem is automatically deployed on every certificate authority (CA) server in the IdM deployment, but it does not service requests until the administrator enables it. RHCS uses the **acmeIPAServerCert** profile when issuing ACME certificates. The validity period of issued certificates is 90 days. Enabling or disabling the ACME service affects the entire IdM deployment.

Jira:RHELDOCS-19405[1]

HSM is now fully supported in IdM

Hardware Security Modules (HSM) are now fully supported in Identity Management (IdM). You can store your key pairs and certificates for your IdM Cerificate Authority (CA) and Key Recovery Authority (KRA) on an HSM. This adds physical security to the private key material.

IdM relies on the networking features of the HSM to share the keys between machines to create replicas. The HSM provides additional security without visibly affecting most IdM operations. When using low-level tools, the certificates and keys are handled differently but this is seamless for most users.

NOTE

Migration of an existing CA or KRA to an HSM-based setup is not supported. You need to reinstall the CA or KRA with keys on the HSM.

You need the following:

- A supported HSM.
- The HSM Public-Key Cryptography Standard (PKCS) #11 library.
- An available slot, token, and the token password.

To install a CA or KRA with keys stored on an HSM, you must specify the token name and the path to the PKCS #11 library. For example:

```
ipa-server-install -r EXAMPLE.TEST -U --setup-dns --allow-zone-overlap --no-forwarders -N --auto-reverse --random-serial-numbers -–token-name=HSM-TOKEN --token-library-path=/opt/nfast/toolkits/pkcs11/libcknfast.so --setup-kra
```

Jira:RHELDOCS-17465[1]

6.16. SSSD

Support for group merging added in authselect

If you are using the **authselect** utility, you no longer need to manually edit the **nssswitch.conf** file to enable group merging. With this update, It is now integrated into **authselect** profiles, eliminating the need for manual changes.

Jira:RHELDOCS-19936[1]

authselect is now required by PAM and cannot be uninstalled

With this enhancement, the **authselect-libs** package now owns **/etc/nsswitch.conf** and selected PAM configuration, including **system-auth**, **password-auth**, **smartcard-auth**, **fingerprint-auth**, and **postlogin** in **/etc/pam.d/**. Ownership of these files has been transferred to **authselect-libs** package, with **/etc/nsswitch.conf`previously owned by the `glibc** package and the PAM configuration files previously owned by the **pam** package. Since **authselect** is required by the **pam** package, it cannot be uninstalled.

For system upgrades from previous RHEL versions:

- If an **authselect** configuration already exists, **authselect apply-changes** automatically updates the configuration to the latest version. If there was no previous **authselect** configuration on your system, no changes are made.
- On systems managed by **authselect**, any non-authselect configurations are now forcefully overwritten without a prompt during the next **authselect** call. The **--force** option is no longer required.

If you require a special configuration, create a custom **authselect** profile. Note that you must manually update custom profiles to keep them up to date with your system.

You can opt-out from using **authselect**:

```
# authselect opt-out
```

Jira:RHELDOCS-19197[1]

Local profile is the new default authselect profile

Due to the removal of the SSSD files provider, a new **authselect local** profile has been introduced to handle local user management without relying on SSSD. The **local** profile replaces the previous **minimal** profile and becomes the default **authselect** profile for new installations instead of the **sssd** profile.

During upgrades, the **authselect** utility automatically migrates existing configurations from **minimal** to **local** profile.

Additionally, the **sssd authselect** profile has been updated to remove the **with-files-domain** and **with-files-access-provider** options and it no longer handles local user accounts directly via these options. If you relied on these options, you must update your SSSD configuration to use **proxy provider** instead of **files provider**.

The **sssd** profile now supports the **--with-tlog** option, which enables session recording for users managed by SSSD.

Jira:RHELDOCS-19263[1]

Support for dynamic DoT updates in SSSD

SSSD supports performing all dynamic DNS (dyndns) queries using DNS-over-TLS (DoT). You can securely update DNS records when IP addresses change, such as Identity Management (IdM) and Active Directory servers. To enable this functionality, you must install the **nsupdate** tool from the **bind9.18-utils** package.

You can use the following new options in the **sssd.conf** file to enable DoT and configure custom certificates for secure DNS updates:

- dyndns_dns_over_tls
- dyndns_tls_ca_cert
- dyndns_tls_cert
- dyndns_tls_key

For more details about these options, see the **sssd-ad(5)** and **sssd-ad(5)** man pages on your system.

Jira:RHELDOCS-20014[1]

New SSSD option: exop_force

You can use the **exop_force** option to force a password change even if no grace logins are left. Previously, SSSD did not attempt password changes if the LDAP server indicated that there were no grace logins remaining. Now, if you set **ldap_pwmodify_mode = exop_force** in the **[domain/...]** section of the **sssd.conf** file, SSSD tries to change the password even if no grace logins are left.

Jira:RHELDOCS-19863[1]

Running SSSD with reduced privileges

To support general system hardening (running software with least privileges possible), the System Security Services Daemon (SSSD) service is configured to run under **sssd** or **root** using the **systemd** service configuration files (service user). This service user now defaults to **sssd** and irrespective of what service user is configured, **root** or **sssd**, all root capabilities are dropped with the exception of a few privileged helper processes.

Note that you must ensure the correct ownership of configuration files. The **sssd.conf** file must be owned by the same user that is used to run the SSSD service. By default, in RHEL 10, this is the **sssd** user. If you create your **sssd.conf** file either manually or via an Ansible script, ensure the ownership is

correct. For example, if you create a **sssd.conf** file under the **root** user, you must change the ownership to **sssd:sssd** using the **chown** command.

Jira:RHELDOCS-18882[1]

Support for KnownHostsCommand has been added to SSSD

With this update, support for **KnownHostsCommand** has been added to SSSD. You can use the tool **sss_ssh_knownhosts** with the SSH **KnownHostsCommand** configuration option to retrieve the host's public keys from a remote server, such as FreeIPA, LDAP, and others. The **sss_ssh_knownhosts** tool replaces the less reliable **sss_ssh_knownhostsproxy** tool. **sss_ssh_knownhostsproxy** is no longer available and a message is displaying indicating the tool is obsolete.

Jira:RHELDOCS-19162[1]

6.17. DESKTOP

Window overview added to GNOME Classic

In previous versions, the overview of open windows was not available while using the GNOME Classic session. With this update, you can use the overview in both the standard GNOME and GNOME Classic mode sessions. This makes the overview's features, including system search, available to GNOME Classic mode users. Users can now also use GNOME Classic mode extensions with the default GNOME session.

Jira:RHELDOCS-19060[1]

RHEL 10 provides enhanced fonts in GNOME desktop

The appearance of fonts has been improved in RHEL 10, with most languages that use variable fonts (VF):

- The GNOME default fonts have changed to Red Hat fonts (previously Abattis Cantarell for Sans and Adobe Source Code Pro for Mono).
- The default core fonts have changed from Deja Vu to the Google Noto VF family for most languages.
- The default installed Chinese, Japanese, and Korean Noto fonts are now VF, though the static fonts are still available.
- The default fonts for Indic (India), Thai, and Khmer have changed to Noto VF which also have the Serif face.
- The default Malayalam fonts have been improved.
- The **default-fonts** meta-packages have been introduced to pull in the appropriate default fonts for each language, making it easier to install default font coverage for particular languages. These meta-packages are installed by default for GNOME desktop.

Other enhancements include the following:

- Indic input methods for India follow the newer Inscript 2 Government standard.
- New **bash-color-prompt** package sets up a default colored Bash shell prompt.

Jira:RHELDOCS-19579

GNOME Online Accounts can restrict which features providers can use

You can use the new **goa.conf** file in the system configuration directory, usually named **/etc/goa.conf**, to limit what features each provider can use.

In the **goa.conf** file, the group name defines the provider type, and the keys define boolean switches to disable the specific features. If you do not set any key or section for a feature, the feature is enabled.

For example, to disable the mail feature for Google accounts, use the following setting:

```
[google]
mail=false
```

You can use the **all** special section name to cover every provider. The value in the specific provider has precedence, if it exists and contains a valid boolean value. Note that some combinations of disabled features can lead to incomplete or invalid accounts being read by the GOA users, such as the Evolution application. Always test the changes first. Restart the GNOME Online Accounts for the changed configuration to take effect.

Jira:RHEL-40831

RHEL Flatpak Mozilla Firefox, Mozilla Thunderbird, Runtime, and SDK are supported

In RHEL 10.0, the following applications are fully supported in RHEL Flatpak:

- Mozilla Firefox
- Flatpak Runtime
- SDK
- Mozilla Thunderbird

In addition, RHEL Flatpak is also supported in Satellite 6.17, see Satellite 6.17 Release notes for more information.

To learn more about RHEL Flatpak, see the Introducing the Red Hat Flatpak Runtime for desktop containers blog post.

You can install RHEL Flatpak application on RHEL 10 systems by performing the following steps:

1. Log in to the Red Hat Container Catalog. Provide the credentials to your Red Hat Customer Portal account or your registry service account tokens:

   ```
   podman login registry.redhat.io


   Username: _<your_user_name>_
   Password: _<your_password>_
   ```

   By default, Podman saves your credentials until you log out.

2. Optional: Save your credentials permanently. Use one of the following options:

   a. Save the credentials for the current user:

```
# cp $XDG_RUNTIME_DIR/containers/auth.json \
    $HOME/.config/flatpak/oci-auth.json
```

b. Save the credentials system-wide:

```
# cp $XDG_RUNTIME_DIR/containers/auth.json \
    /etc/flatpak/oci-auth.json
```

For best practices, log in to the Red Hat Container Catalog by using registry account tokens when installing credentials system-wide.

3. Install the Mozilla Firefox RHEL 10 Flatpak:

```
$ flatpak install rhel org.mozilla.firefox
```

NOTE

For RHEL 10.0, the ID of the Mozilla Firefox RHEL Flatpak has been changed from **org.mozilla.Firefox** to **org.mozilla.firefox**

4. Run Mozilla Firefox

a. From the command line:

```
$ flatpak run org.mozilla.firefox
```

b. Launch Firefox from GNOME Activities Overview.

Jira:RHEL-53563[1]

RHEL 10 provides Papers

Papers is a document viewer application for the GNOME desktop. Papers supports thumbnails, outlines, PDF, Tiff, and the comic book formats. Other features include:

- Displaying signatures.
- Modernized user interface (UI) with the GTK4 toolkit and the **libadwaita** library to handle desktop and mobile use cases.
- Signing of PDF files.

NOTE

You cannot use Papers to open PostScript files. To open PostScript files, convert them to PDF and open the PDF. Papers is not able to open XPS files.

Jira:RHELDOCS-19661[1]

6.18. THE WEB CONSOLE

New package: cockpit-files

The **cockpit-files** package provides the File manager page in the RHEL web console. With the File manager, you can perform the following actions:

- Browse files and directories on file systems you can access
- Sort files and directories by various criteria
- Filter displayed files by a sub-string
- Copy, move, delete, and rename files and directories
- Create directories
- Upload files
- Bookmark file paths
- Use keyboard shortcuts for the actions

Jira:RHELDOCS-16362[1]

6.19. RED HAT ENTERPRISE LINUX SYSTEM ROLES

Support for new ha_cluster system role features

The **ha_cluster** system role now supports the following features:

- Configuring utilization attributes for node and primitive resources.
- Configuring node addresses and SBD options by using the **ha_cluster_node_options** variable. If both **ha_cluster_node_options** and **ha_cluster** variables are defined, their values are merged, with values from **ha_cluster_node_options** having precedence.
- Configuring access control lists (ACLs).
- Configuring Pacemaker alerts to take an external action when a cluster event such as node failure or resource starting or stopping occurs.
- Easy installation of agents for cloud environments by setting the **ha_cluster_install_cloud_agents** variable to **true**.

Jira:RHEL-34893[1], Jira:RHEL-34894, Jira:RHEL-34898, Jira:RHEL-34885

Support for exporting corosync configuration of an existing cluster

The **ha_cluster** RHEL system role now supports exporting the **corosync** configuration of an existing cluster in a format that can be fed back to the role to re-create the same cluster. If you did not use the **ha_cluster** RHEL system role to create your cluster, or if you have lost the original playbook for the cluster, you can use this feature to build a new playbook for the cluster.

Jira:RHEL-46219

New sudo RHEL system role

sudo is a critical part of RHEL system configuration. With the new **sudo** RHEL system role, you can consistently manage sudo configuration at scale across your RHEL systems.

Jira:RHEL-37551

The storage RHEL system role can now manage Stratis pools

With this enhancement, you can use the **storage** RHEL system role to complete the following tasks:

- Create a new encrypted and unencrypted Stratis pool
- Add new volumes to the existing Stratis pool
- Add new disks to the Stratis pool

For details on how to manage Stratis pools and other related information, see the resources in the **/usr/share/doc/rhel-system-roles/storage/** directory.

Jira:RHEL-40798[1]

New variables in the podman RHEL system role: podman_registry_certificates and podman_validate_certs

The following two variables have been added to the **podman** RHEL system role:

- **podman_registry_certificates** (list of dictionary elements): Enables you to manage TLS certificates and keys used to connect to the specified container image registry.
- **podman_validate_certs** (boolean, defaults to null): Controls whether pulling images from container image registries will validate TLS certificates or not. The default null value means that it is used whatever the default configured by the **containers.podman.podman_image** module is. You can override the **podman_validate_certs** variable on a per-specification basis with the **validate_certs** variable.

As a result, you can use the **podman** RHEL system role to configure TLS settings for connecting to container image registries.

For more details, see the resources in the **/usr/share/doc/rhel-system-roles/podman/** directory. Alternatively, you can review the **containers-certs(5)** manual page.

Jira:RHEL-34884[1]

New variables in the podman RHEL system role: podman_registry_username and podman_registry_password

The **podman** RHEL system role now enables you to specify the container image registry credentials either globally or on a per-specification basis. For that purpose, you must configure both role variables:

- **podman_registry_username** (string, defaults to unset): Configures the username for authentication with the container image registry. You must also set the **podman_registry_password** variable. You can override **podman_registry_username** on a per-specification basis with the **registry_username** variable. Each operation involving credentials would then be performed according to the detailed rules and protocols defined in that specification.
- **podman_registry_password** (string, defaults to unset): Configures the password for authentication with the container image registry. You must also set the **podman_registry_username** variable. You can override **podman_registry_password** on a

per-specification basis with the **registry_password** variable. Each operation involving credentials would then be performed according to the detailed rules and protocols defined in that specification. For security, encrypt the password using the Ansible Vault feature.

As a result, you can use the **podman** RHEL system role to manage containers with images, whose registries require authentication for access.

For more details, see the resources in the /**usr**/**share**/**doc**/**rhel-system-roles**/**podman**/ directory.

Jira:RHEL-34890[1]

New variable in the podman RHEL system role: podman_credential_files

Some operations need to pull container images from registries in an automated or unattended way and cannot use the **podman_registry_username** and **podman_registry_password** variables.

Therefore, the **podman** RHEL system role now accepts the **containers-auth.json** file to authenticate against container image registries. For that purpose, you can use the following role variable:

podman_credential_files (list of dictionary elements)

Each dictionary element in the list defines a file with user credentials for authentication to private container image registries. For security, encrypt these credentials by using the Ansible Vault feature. You can specify file name, mode, owner, group of the file, and can specify the contents in different ways. See the role documentation for more details.

As a result, you can input container image registry credentials for automated and unattended operations.

For more details, see the resources in the /**usr**/**share**/**doc**/**rhel-system-roles**/**podman**/ directory. Alternatively, you can review the **containers-auth.json(5)** and **containers-registries.conf(5)** manual pages.

Jira:RHEL-34891[1]

New variables in the journald RHEL system role: journald_rate_limit_interval_sec and journald_rate_limit_burst

The following two variables have been added to the **journald** RHEL system role:

- **journald_rate_limit_interval_sec** (integer, defaults to 30): Configures a time interval in seconds, within which only the **journald_rate_limit_burst** log messages are handled. The **journald_rate_limit_interval_sec** variable corresponds to the **RateLimitIntervalSec** setting in the **journald.conf** file.

- **journald_rate_limit_burst** (integer, defaults to 10 000): Configures the upper limit of log messages, which are handled within the time defined by **journald_rate_limit_interval_sec**. The **journald_rate_limit_burst** variable corresponds to the **RateLimitBurst** setting in the **journald.conf** file.

As a result, you can use these settings to tune the performance of the **journald** service to handle applications that log many messages in a short period of time.

For more details, see the resources in the /**usr**/**share**/**doc**/**rhel-system-roles**/**journald**/ directory.

Jira:RHEL-34892[1]

The ssh RHEL system role now recognizes the ObscureKeystrokeTiming and ChannelTimeout configuration options

The **ssh** RHEL system role has been updated to reflect addition of the following configuration options in the OpenSSH utility suite:

- **ObscureKeystrokeTiming** (yes|no|interval specifier, defaults to 20): Configures whether the **ssh** utility should obscure the inter-keystroke timings from passive observers of network traffic.
- **ChannelTimeout**: Configures whether and how quickly the **ssh** utility should close inactive channels.

When using the **ssh** RHEL system role, you can use the new options such as in this example play:

```
- name: Non-exclusive ssh configuration
  hosts: managed-node-01.example.com
  tasks:
    - name: Configure ssh to obscure keystroke timing and set 5m session timeout
      ansible.builtin.include_role:
        name: rhel-system-roles.ssh
      vars:
        ssh_ObscureKeystrokeTiming: "interval:80"
        ssh_ChannelTimeout: "session=5m"
```

Jira:RHEL-40181

The storage RHEL system role can now resize LVM physical volumes

If the size of a block device has changed and you use this device in an LVM, you can adjust the LVM physical volume as well. With this enhancement, you can use the **storage** RHEL system role to resize LVM physical volumes to match the size of the underlying block devices after you resized it. To enable automatic resizing, set **grow_to_fill: true** on the pool in your playbook.

Jira:RHEL-40797[1]

The nbde_client RHEL system role now enables you to skip running certain configurations

With the **nbde_client** RHEL system role you can now disable the following mechanisms:

- Initial RAM disk
- NetworkManager flush module
- Dracut flush module

The **clevis-luks-askpass** utility unlocks some storage volumes late in the boot process after the NetworkManager service puts the operating system on the network. Therefore, no configuration changes to the mentioned mechanisms are necessary.

As a result, you can disable the mentioned configurations from being run to support advanced networking setups, or volume decryption to occur late in the boot process.

Jira:RHEL-45718[1]

New variable in the postfix RHEL system role: postfix_files

The **postfix** RHEL system role now enables you to configure extra files for the Postfix mail transfer agent. For that purpose, you can use the following role variable:

postfix_files

Defines a list of files to be placed in the **/etc/postfix/** directory that can be converted into Postfix Lookup Tables if needed. This variable enables you to configure Simple Authentication and Security Layer (SASL) credentials, and similar. For security, encrypt files that contain credentials and other secrets by using the Ansible Vault feature.

As a result, you can use the **postfix** RHEL system role to create these extra files and integrate them in your Postfix configuration.

For more details, see the resources in the **/usr/share/doc/rhel-system-roles/postfix/** directory.

Jira:RHEL-46855[1]

The snapshot RHEL system role now supports managing snapshots of LVM thin pools

With thin provisioning, you can use the **snapshot** RHEL system role to manage snapshots of LVM thin pools. These thin snapshots are space-efficient and only grow as data is written or modified after the snapshot is taken. The role automatically detects if the specified volume is scheduled for a thin pool. The added feature could be useful in environments where you need to take frequent snapshots without consuming much of physical storage.

Jira:RHEL-48230[1]

New option in the logging RHEL system role: reopen_on_truncate

The **files** input type of the **logging_inputs** variable now supports the following option:

reopen_on_truncate (boolean, defaults to false)

Configures the **rsyslog** service to re-open the input log file if it was truncated, such as during log rotation. The **reopen_on_truncate** role option corresponds to the **reopenOnTruncate** parameter for **rsyslog**.

As a result, you can configure **rsyslog** in an automated fashion through the **logging** RHEL system role to re-open an input log file if it was truncated.

For more details, see the resources in the **/usr/share/doc/rhel-system-roles/logging/** directory.

Jira:RHEL-48609[1]

New variable in the logging RHEL system role: logging_custom_config_files

You can provide custom logging configuration files by using the following variable for the **logging** RHEL system role:

logging_custom_config_files (list)

Configures a list of configuration files to copy to the default logging configuration directory. For example, for the **rsyslog** service it is the **/etc/rsyslog.d/** directory. This assumes the default logging configuration loads and processes the configuration files in that directory. The default **rsyslog** configuration has a directive such as **$IncludeConfig /etc/rsyslog.d/*.conf**.

As a result, you can use customized configurations not provided by the **logging** RHEL system role.

For more details, see the resources in the **/usr/share/doc/rhel-system-roles/logging/** directory.

Jira:RHEL-50288[1]

The logging RHEL system role can set ownership and permissions for rsyslog files and directories

The **files** output type of the **logging_outputs** variable now supports the following options:

- **mode** (raw, defaults to null): Configures the **FileCreateMode** parameter associated with the **omfile** module in the **rsyslog** service.
- **owner** (string, defaults to null): Configures the **fileOwner** or **fileOwnerNum** parameter associated with the **omfile** module in **rsyslog**. If the value is an integer, it sets **fileOwnerNum**. Otherwise, it sets **fileOwner**.
- **group** (string, defaults to null): Configures the **fileGroup** or **fileGroupNum** parameter associated with the **omfile** module in **rsyslog**. If the value is an integer, it sets **fileGroupNum**. Otherwise, it sets **fileGroup**.
- **dir_mode** (defaults to null): Configures the **DirCreateMode** parameter associated with the **omfile** module in **rsyslog**.
- **dir_owner** (defaults to null): Configures the **dirOwner** or **dirOwnerNum** parameter associated with the **omfile** module in **rsyslog**. If the value is an integer, it sets **dirOwnerNum**. Otherwise, it sets **dirOwner**.
- **dir_group** (defaults to null): Configures the **dirGroup** or **dirGroupNum** parameter associated with the **omfile** module in **rsyslog**. If the value is an integer, it sets **dirGroupNum**. Otherwise, it sets **dirGroup**.

As a result, you can set ownership and permissions for files and directories created by **rsyslog**.

Note that the file or directory properties are the same as the corresponding variables in the Ansible **file** module.

For more details, see the resources in the **/usr/share/doc/rhel-system-roles/logging/** directory. Alternatively, review the output of the **ansible-doc file** command.

Jira:RHEL-50289[1]

Using the storage RHEL system role creates fingerprints on managed nodes

If not already present, **storage** creates a unique identifier (fingerprint) every time you run this role. The fingerprint has the form of the **# system_role:storage** string written to the **/etc/fstab** file on your managed nodes. As a result, you can track which nodes are managed by **storage**.

Jira:RHEL-50291[1]

New src parameter is added to the network RHEL system role

The **src** parameter to the **route** sub-option of the **ip** option for the **network_connections** variable has been added. This parameter specifies the source IP address for a route. It is useful typically for the multi-WAN connections. There you get setups where a machine has multiple public IP addresses, and

you want to ensure that outbound traffic uses a specific IP address tied to a particular network interface. As a result, support for the **src** parameter provides better control over traffic routing and ensures a more robust and flexible network configuration capability in the described scenarios

For more details, see the resources in the **/usr/share/doc/rhel-system-roles/network/** directory.

Jira:RHEL-53901[1]

Support for configuring GFS2 file systems on RHEL 9 clusters by using RHEL system roles

Red Hat Enterprise Linux 10 supports the configuration and management of the Red Hat Global File System 2 (GFS2) by using the **gfs2** RHEL system role on a RHEL 10 control node to manage RHEL 9 systems. The Red Hat Enterprise Linux (RHEL) Resilient Storage Add-On, which includes the GFS2 file system, is itself not supported on RHEL 10 systems. The role creates GFS2 file systems in a Pacemaker cluster managed with the **pcs** command-line interface.

Previously, setting up GFS2 file systems in a supported configuration required you to follow a long series of steps to configure the storage and cluster resources. The **gfs2** role simplifies the process. Using the role, you can specify only the minimum information needed to configure GFS2 file systems in a RHEL high availability cluster.

The gfs2 role performs the following tasks:

- Installing the packages necessary for configuring a GFS2 file system in a Red Hat high availability cluster
- Setting up the **dlm** and **lvmlockd** cluster resources
- Creating the LVM volume groups and logical volumes required by the GFS2 file system
- Creating the GFS2 file system and cluster resources with the necessary resource constraints

Jira:RHEL-34828[1]

New variables in the microsoft.sql.server system role: mssql_tools_versions and mssql_tls_self_sign

The new **mssql-tools18** package brings functionality that is not backwards-compatible with the previous versions of the **mssql-tools** package. Therefore the following variables have been added to the **microsoft.sql.server** system role to adapt to the changes:

- **mssql_tools_versions** (list, defaults to version 18): Enables you to install different versions of **mssql-tools**.
- **mssql_tls_self_sign** (boolean): Specifies whether the certificates that you use are self-signed or not. Applicable when you also set the **mssql_tls_enable: true** variable.

IMPORTANT

When you use **mssql-tools18** with self-signed TLS certificates, you have to set **mssql_tls_self_sign: true** so that the role sets the **-C** flag in the **sqlcmd** command-line utility so that your certificates can be trusted.

As a result, you can use these configurations to install **mssql_tools** version 17; 18; or both in parallel.

For more details, see the resources in the **/usr/share/ansible/roles/microsoft.sql-server/** directory.

Jira:RHEL-68468

New variable in the sudo RHEL system role: sudo_check_if_configured

The **sudo** RHEL system role now has the following variable:

- **sudo_check_if_configured** (boolean): Provides a semantic check of an already configured **sudoers** file in case the Ansible setup is not needed and is skipped.

As a result, you can use this setting to ensure the **sudo** role idempotence if Ansible intervention is not required.

For more details, see the resources in the **/usr/share/doc/rhel-system-roles/sudo/** directory.

Jira:RHEL-67419[1]

New variable in the systemd RHEL system role: systemd_units_user

With this update, the **systemd** RHEL system role can now also manage user units through the following variable:

- **systemd_units_user** (dictionary): Each key is a name of a user given in one of the lists passed to the role, and **root** (even if **root** is not given). Each value is a dictionary of **systemd** units for that user, or system units for **root**.

IMPORTANT

The role does not create new users and it will return an error if you specify a non-existent user.

As a result, you can use this setting to manage user units with the **systemd** RHEL system role.

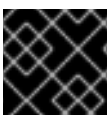
For more details, see the resources in the **/usr/share/doc/rhel-system-roles/systemd/** directory.

Jira:RHEL-67420[1]

New RHEL system role: aide

aide is a new RHEL system role for detecting unauthorized changes to files, directories, and system binaries. With this role, you can accomplish for example the following tasks:

- Install the **aide** package on the managed node.
- Generate the **/etc/aide.conf** file and template it out to the managed node.
- Initialize the (Advanced Intrusion Detection Environment) AIDE database.
- Run AIDE integrity checks on the managed node.



IMPORTANT

The role does not explain how to create a suitable AIDE configuration.

As a result, you can manage AIDE at scale in an automated fashion to address your security, compliance or auditing needs.

For more details, see the resources in the **/usr/share/doc/rhel-system-roles/aide/** directory.

Jira:RHEL-67411[1]

The microsoft.sql.server system role enables AES 128-bit and AES 256-bit encryption for AD users

Since version 1.1.83, the **adutil** utility supports the Kerberos protocol with AES 128-bit and AES 256-bit encryption when creating and modifying an Active Directory (AD) user. With this update, the **microsoft.sql.server** system role automates enabling AES 128-bit and AES 256-bit encryption provided by the Kerberos protocol when creating or modifying AD users. As a result, manual post-configuration tasks are not necessary.

Jira:RHEL-68490

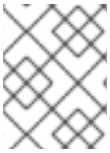
sshd RHEL system role validates commands and configurations

The **sshd** role uses the **quote** command when using the **command** or **shell** plugins to ensure you can use these commands safely. The role also validates certain user-supplied role variables passed to these plugins. This improves the security and robustness of using the role because, without validation, user-supplied variables that contain white space could split and not function correctly.

Jira:RHEL-73441[1]

RHEL 10 provides the postfix RHEL system role with a new variable postfix_default_database_type

The **postfix** system role can determine the default database type used by **postfix** and export it as a variable **postfix_default_database_type**. As a result, you can set configuration parameters based on the default database type.



NOTE

Using **postfix_default_database_type** in a configuration parameter value is not supported on Ansible 2.9.

Jira:RHEL-70554[1]

The podman RHEL system role can manage the quadlet units of type Pod

The **podman** utility of version 5 added support for **Pod** quadlet types. Consequently, the **podman** RHEL system role now enables you to also manage the quadlet units of type **Pod**.

For more details, see the upstream article.

Jira:RHEL-67417[1]

New property added to the network RHEL system role network_connections variable: autoconnect_retries

There is no fine-grained control over the number of automatic retries to reconnect a network connection in the **network** RHEL system role. This limitation could be problematic for certain use cases where

extending the retry process is critical, particularly in environments with unstable networks. The **autoconnect_retries** property added to the to the **network_connections** role variable configures how many times NetworkManager attempts to reconnect a network connection after an autoconnect failure. As a result, the **network** RHEL system role now allows configuring the number of automatic reconnection attempts after an autoconnect failure by using the **autoconnect_retries** property in the **network_connections** variable. This enhancement provides greater control over network stability and performance, especially in environments with unstable networks.

For more details, see the resources in the **/usr/share/doc/rhel-system-roles/network/** directory.

Jira:RHEL-67416[1]

New property added to the network RHEL system role network_connections variable: wait_ip

This update provides added support for the **wait_ip** property of the **ip** option in the **network_connections** role variable. The property specifies if the system should consider the network connection as activated only when a specific IP stack is configured. You can configure **wait_ip** with the following values:

- **any**: The system considers the connection activated once any IP stack is configured.
- **ipv4**: The system waits until IPv4 is configured.
- **ipv6**: The system waits until IPv6 is configured.
- **ipv4+ipv6**: The system waits until both IPv4 and IPv6 are configured.

As a result, the **network** RHEL system role now allows you to configure network connections based on specific IP stack configurations. This enables the connection to remain activated even if an IP address is not assigned, depending on the selected **wait_ip** setting.

For more details, see the resources in the **/usr/share/doc/rhel-system-roles/network/** directory.

Jira:RHEL-67415[1]

Added support for Valkey as an alternative to Redis

This update provides added support for the Valkey in-memory data structure store. It is an alternative to Redis, which is no longer open source and is being removed from Linux distributions. Valkey is typically used as a high-performance caching layer. It stores data in memory, which accelerates applications by caching frequently accessed data. Additionally, you can use Valkey for other performance-critical operations, for example:

- Storing and retrieving user session data.
- Real-time communication between different application parts.
- Providing fast data access for analytics and monitoring.

Jira:RHEL-67413[1]

New variable in the logging RHEL system role: logging_custom_templates

The following variable has been added to the **logging** RHEL system role:

- **logging_custom_templates**: A list of custom template definitions. You can use it with the

logging_outputs variable when its option is **type: files** or **type: forwards**. You can specify this custom template for each output by setting the **template** option in a particular **logging_outputs** specification. Alternatively, you can set this custom template to be used by default for all files and forwards outputs by using the **logging_files_template_format** and **logging_forwards_template_format** global options.

As a result, you can format log entries differently than what the built-in defaults provide.

For more details, see the resources in the **/usr/share/doc/rhel-system-roles/logging/** directory.

Jira:RHEL-67286[1]

6.20. VIRTUALIZATION

Virtualization support for IBM z17 processors

With this update, virtualization on RHEL adds support for the IBM z17 CPUs. As a result, virtual machines hosted on an IBM Z system with RHEL can now use new features that the z17 processors provide.

Jira:RHEL-33137[1]

Retrievable secrets are supported for Secure Execution on IBM Z

With this update, you can use generalized host-based secrets for cryptographic devices in Secure Execution virtual machines (VMs) on IBM Z. As a result, it is no longer needed to store secrets in an **initramfs** image when configuring Secure Execution, which simplifies creating a secure VM image. Note that this feature is currently only supported on IBM z17 processors.

Jira:RHEL-25204[1]

RHEL on HPE can run up to 4096 vCPUs

With this feature, a RHEL virtual machine (VM) instance running with the RHEL hypervisor on Hewlett Packard Enterprise Compute Scale-Up Server now supports up to 4096 virtual CPUs, 32 sockets, and 64 TB of memory to handle in-memory databases and other large compute intensive workloads.

Jira:RHEL-57668[1]

RHEL 10 provides nbdkit version 1.38

The **nbdkit** package has been updated to upstream version 1.38, which provides various bug fixes and enhancements. The most notable changes are the following:

- Block size advertising has been enhanced and a new read-only filter has been added.
- The Python and OCaml bindings support more features of the server API.
- Internal struct integrity checks have been added to make the server more robust.

For a complete list of changes, see the upstream release notes.

Jira:RHEL-32748

KVM on IBM Z now supports more than one boot device

Guest operating systems running on KVM on IBM Z hosts can attempt booting from additional devices when the primary boot device is not bootable. This feature is supported for the following device types:

- **virtio-net**
- **virtio-blk**
- **virtio-scsi/cdrom**

To configure the order of the boot devices for the VM, use the **order** parameter on the **<boot>** line of their XML configuration. The VM will now attempt up to 8 devices for booting.

In addition, these devices now support the **loadparm** parameter for the **<boot>** line of their XML configuration. By using **loadparm**, it is possible to configure which boot entry the device uses when the guest operating system boots from the device.

Jira:RHEL-68444, Jira:RHEL-24070

Newly supported features for virtual machines on 64-bit ARM hosts

The following features are now supported for virtual machines on RHEL hosts that use the 64-bit ARM architecture, also known as aarch64:

- Migrating VMs between 64-bit ARM hosts. Note, however, that the migration currently only works when both hosts use the same CPU type and memory page size.
- The Trusted Platform Module (TPM) Interface Specification (TIS) hardware interface
- Non-volatile dual inline memory module (NVDIMM) memory device
- The virtio-iommu device

Jira:RHELDOCS-19832[1]

RHEL supports live migrating a VM with a Mellanox virtual function

With this update, you can perform live migration of a virtual machine (VM) with an attached virtual function (VF) of a Mellanox networking device.

However, this feature is currently only supported with a Mellanox CX-7 networking device with a specific firmware version. The VF on the Mellanox CX-7 networking device uses a new **mlx5_vfio_pci** driver, which adds functionality that is necessary for the live migration, and **libvirt** binds the new driver to the VF automatically.

For more details and limitations, see: Live migrating a virtual machine with an attached Mellanox virtual function

Jira:RHELDOCS-19210[1]

Support for USO in virtio-net

This update adds the User Datagram Protocol (UDP) Segmentation Offload (USO) feature for the Windows **virtio-net** driver. This makes it possible for Windows VMs to offload the segmentation of large UDP packets to the underlying **virtio-net** device. As a result, this reduces CPU usage in the VMs and improves overall UDP networking performance, especially in workloads that generate high volumes of UDP traffic.

Jira:RHEL-1300[1]

virt-install now supports creating VMs with SEV-SNP

You can now use the **virt-install** utility to create a virtual machine (VM) that uses the AMD Secure Encrypted Virtualization with Secure Nested Paging (SEV-SNP) feature. To do so, use the **launchSecurity sev-snp,policy=0x30000** option.

Note that SEV-SNP is currently provided as a Technology Preview.

Jira:RHEL-62960

Support for VM live migration with shared virtiofs directory that provides write access to other parties

With this update, you can live migrate a virtual machine (VM) with a **virtiofs** shared directory, even if multiple other parties, such as the host and other VMs, have write access to that directory.

Jira:RHEL-29027

Virtual machines supported in RHEL for Real Time

This update introduces full support for real-time virtualization in RHEL for Real Time. You can configure the host and guest operating systems to achieve low-latency and deterministic behavior for virtual machines (VMs). This makes real-time VMs suitable for applications that require real-time performance, such as industrial automation, telecommunications, and automotive systems.

Jira:RHELDOCS-20116[1]

6.21. RHEL IN CLOUD ENVIRONMENTS

cloud-init now uses NetworkManager as the default network renderer

With this update, the **cloud-init** utility uses **NetworkManager** (NM) as the back end for network configuration when initializing a cloud instance. As a result, using NM keyfiles in **cloud-init** setup no longer requires reconfiguring **/etc/cloud/cloud.cfg**.

Jira:RHEL-29720[1]

RHEL 10 provides Unified Kernel Image

Unified Kernel Image (UKI) for RHEL fully supported. To use RHEL UKI, you must first install the **kernel-uki-virt** package. RHEL UKI can enhance SecureBoot protection in virtualized and cloud environments.

Jira:RHELDOCS-19840[1]

Enhanced automatic registration for eligible RHEL images

When purchasing certain eligible cloud marketplace subscriptions for RHEL 9.6 or later and for RHEL 10.0 or later, an improved version of the auto-registration function is available.

With the enhanced auto-registration, any RHEL instances on the eligible marketplaces will be automatically registered to Red Hat and automatically receive content updates from Red Hat Update Infrastructure (RHUI) after you establish a trusted connection between your Red Hat account and your account for the specific cloud platform, even if you did not have the trusted connection when you set launched the instance.

For additional details, see Understanding auto-registration.

Jira:RHELDOCS-19664[1]

WSL images of RHEL 8 - 10 are available on the Customer Portal

RHEL 8, RHEL 9, and RHEL 10 images for the Windows Subsystem for Linux (WSL) can now be downloaded from the Red Hat Customer Portal. These images are available for all RHEL subscriptions, including no-cost developer subscriptions. By using the WSL images, you can create RHEL instances on your Windows system.

Note that the WSL images are provided as self-supported. As such, they are not supported by Red Hat, and are intended for application development purposes only.

In addition, the following issues are currently present in the RHEL guest operating system if you use a WSL image with a Windows WSL host:

- WSL instances of RHEL might work incorrectly in a graphical interface. Using a text user interface is recommended instead.
- To use podman, you must add the following lines to the **/etc/containers/containers.conf** file, in addition to the standard configuration steps:

```
[network]
firewall_driver="iptables"
```

- To use cloud-init, you must create the **/etc/cloud/cloud.cfg.d/99_wsl.cfg** file and add the following content to it, in addition to the standard configuration steps:

```
datasource_list: [WSL]
network: {config: disabled}
```

- It is not possible to set SELinux to enforcing mode.
- FIPS mode is not available in WSL instances of RHEL.

Jira:RHELDOCS-19876

6.22. SUPPORTABILITY

The --api-url option is now available

With the **--api-url** option you can call another API as required. For example, the API for an OCP cluster. Example: **sos collect --cluster-type=ocp --cluster-option ocp.api-url=_<API_URL> --alloptions**.

Jira:RHEL-24523

The new --skip-cleaning-files option is now available

The **--skip-cleaning-files** option for the **sos report** command allows you to skip cleaning selected files. The option supports globs and wildcards. Example: **sos report -o host --batch --clean --skip-cleaning-files 'hostname'**.

Jira:RHEL-30893[1]

The plugin option names now use only hyphens instead of underscores

To ensure consistency across **sos** global options, the plugin option names now use only hyphens instead of underscores For example, the networking plugin **namespace_pattern** option is now **namespace-pattern** and must be specified by using the **--plugin-option networking.namespace-pattern=**

<pattern> syntax.

Jira:RHELDOCS-18655[1]

6.23. CONTAINERS

Image mode for RHEL supports FIPS mode

With this enhancement, you can enable the FIPS mode when building a bootc image to configure the system to use only FIPS-approved modules. You can use **bootc-image-builder**, which requires enabling the FIPS cryptographic policy in the Containerfile configuration, or use the RHEL Anaconda installation, that additionally to enabling FIPS mode in the Containerfile, also requires adding the **fips=1** kernel argument when booting the system installation. See Installing the system with FIPS mode enabled for more details.

The following is a Containerfile with instructions to enable the **fips=1** kernel argument.

```
FROM registry.redhat.io/rhel9/rhel-bootc:latest#
# Enable fips=1 kernel argument:
https://bootc-dev.github.io/bootc//building/kernel-arguments.html
COPY 01-fips.toml /usr/lib/bootc/kargs.d/
# Install and enable the FIPS crypto policy
RUN dnf install -y crypto-policies-scripts && update-crypto-policies --no-reload --set FIPS
```

The content of **01-fips.toml** is:

```
kargs = ["fips=1"]
```

Jira:RHELDOCS-18585[1]

Support to creating and deploying VMDK with bootc-image-builder

With this enhancement, you can create a Virtual Machine Disk (VMDK) from a bootc image, by using the **bootc-image-builder** tool, and deploy VMDK images to VMware vSphere.

Jira:RHELDOCS-18398[1]

Podman and Buildah support adding OCI artifacts to image indexes

With this update, you can create artifact manifests and add them to image indexes.

The **buildah manifest add** command supports the following options:

- the **--artifact** option to create artifact manifests
- the **--artifact-type**, **--artifact-config-type**, **--artifact-layer-type**, **--artifact-exclude-titles**, and **--subject** options to configure the contents of the artifact manifests it creates.

The **buildah manifest annotate** command supports the following options:

- the **--index** option to set annotations on the index itself instead of a one of the entries in the image index
- the **--subject** option for setting the subject field of an image index.

The **buildah manifest create** command supports the **--annotation** option to add annotations to the new image index.

Jira:RHEL-33571

Option is available to disable Podman health check event

This enhancement adds a new **healthcheck_events** option in the **containers.conf** configuration file under the **[engine]** section to disable the generation of **health_status** events. Set **healthcheck_events=false** to disable logging health check events.

Jira:RHEL-34604

Runtime resource changes in Podman are persistent

The updates of container configuration by using the **podman update** command are persistent. Note that this enhancement is for both SQLite and BoltDB database backends.

Jira:RHEL-33566

Building multi-architecture images is fully supported

The **podman farm build** command that creates multi-architecture container images is fully supported.

A farm is a group of machines that have a UNIX Podman socket running in them. The nodes in the farm can have different machines of various architectures. The **podman farm build** command is faster than the **podman build --arch --platform** command.

You can use **podman farm build** to perform the following actions:

- Build an image on all nodes in a farm.
- Bundle an image on all nodes in a farm up into a manifest list.
- Run the **podman build** command on all the farm nodes.
- Push the images to the registry specified by using the **--tag** option.
- Locally create a manifest list.
- Push the manifest list to the registry.

The manifest list contains one image per native architecture type present in the farm.

Jira:RHEL-34611

Quadlets for pods in Podman are available

Beginning with Podman v5.0, you can use Quadlet to automatically generate a **systemd** service file from a pod description.

Jira:RHEL-33573

The Podman v2.0 RESTful API has been updated

The new fields has been added to the **libpod/images/json** endpoint:

- The **isManifest** boolean field to determine if the target is a manifest or not. The **libpod** endpoint returns both images and manifest lists.
- The **os** and **arch** fields for image listing.

Jira:RHEL-34613

Kubernetes YAML supports a data volume container as an init container

A list of images to automatically mount as volumes can be specified in Kubernetes YAML by using the **"io.podman.annotations.kube.image.automount/$ctrname"** annotation. Image-based mounts using **podman run --mount type=image,source=<image>,dst=<path>,subpath=<path>** support a new option, **subpath**, to mount only part of the image into the container.

Jira:RHEL-34606

The containers.conf file is read-only

The system connections and farm information stored in the **containers.conf** file is read-only. The system connections and farm information will be stored in the **podman.connections.json** file, managed only by Podman. Podman continues to support the old configuration options such as **[engine.service_destinations]** and the **[farms]** section. You can still add connections or farms manually if needed however, it is not possible to delete a connection from the **containers.conf** file with the **podman system connection rm** command.

You can still manually edit the **containers.conf** file if needed. System connections that were added by Podman v4.0 remain unchanged after the upgrade to Podman v5.0.

Jira:RHEL-40639

Default settings changes for Podman v5.0

In RHEL 10.0, the following default settings changes for Podman v5.0:

- cgroups v2 is used by default instead of cgroups v1
- **pasta** is the default network used by rootless containers instead of **slirp4netns**

Jira:RHEL-40643

A new rhel10/rteval container image

The real-time **registry.redhat.io/rhel10/rteval** container image is available in the Red Hat Container Registry to run latency analysis on either a standalone RHEL installation. With **rhel10/rteval** container image, you can perform latency testing within a containerized setup to determine if such a solution is viable for your real-time workloads or to compare results against a bare metal run of **rteval**. To use this feature, subscribe to RHEL with real-time support. No tuning guidelines are provided.

Jira:RHELDOCS-18522[1]

The --compat-volumes option is available for Podman and Buildah

You can use the new **--compat-volumes** option with the **buildah build**, **podman build**, and **podman farm build** commands. This option triggers special handling for the contents of directories marked using the **VOLUME** instruction such that their contents can subsequently only be modified by **ADD** and **COPY** instructions. Any changes made in those locations by **RUN** Instructions will be discarded. Previously, this behavior was the default, but it is disabled by default.

Jira:RHEL-52240

macvlan and ipvlan network interface names are configurable in containers.conf

To specify **macvlan** and **ipvlan** networks, you can adjust the name of the network interface created inside containers by using the new **interface_name** field in the **containers.conf** configuration file.

Jira:RHELDOCS-18769[1]

Support to building GCP images by using bootc-image-builder

By using the **bootc-image-builder** tool you can generate **.gce** disk images and provision the instances on the Google Compute Engine (GCE) platform.

Jira:RHELDOCS-18472[1]

Podman supports pushing and pulling images compressed with zstd:chunked

You can push images compressed with the **zstd:chunked** format to reduce the image size and use partial pulls.

Jira:RHEL-67260

The Container Tools packages have been updated

The updated Container Tools RPM meta-package, which contains the Podman, Buildah, Skopeo, **crun**, and **runc** tools, is available. The Buildah has been updated to version 1.39.0, Skopeo has been updated to version 1.18.0. Podman v5.4 contains the following notable bug fixes and enhancements over the previous version:

- The **podman update** command supports a wide variety of options related to health checks: the **--health-cmd** to define a new health check and **--no-healthcheck** to disable an existing health check. These options make it easier to add, modify, or disable health checks on running containers. For more information, see the **podman-update(5)** man page.
- The **--mount type=volume** option for the **podman run**, **podman create**, and **podman volume create** commands supports a new option, **subpath=**, to make only a subset of the volume visible in the container.
- The **--userns=keep-id** option for the **podman run**, **podman create**, and **podman pod create** commands supports a new option, **--userns=keep-id:size=**, to configure the size of the user namespace.
- The **podman kube play** command supports Container Device Interface (CDI) devices.
- The **podman run**, **podman create**, and **podman pod create** commands support a new option, **--hosts-file**, to define the base file used for **/etc/hosts** in the container.
- The **podman run**, **podman create**, and **podman pod create** commands support a new option, **--no-hostname**, which disables the creation of **/etc/hostname** in the container.
- The **podman network create** command supports a new option for bridge networks, **--opt mode=unmanaged**, which allows Podman to use an existing network bridge on the system without changes.

- The **--network** option for **podman run**, **podman create**, and **podman pod create** accepts a new option for bridge networks, **host_interface_name**, which specifies a name for the network interface created outside the container.
- The **podman manifest rm** command supports a new option, **--ignore**, to proceed successfully when removing manifests that do not exist.
- The **podman system prune** command supports a new option, **--build**, to remove build containers leftover from prematurely terminated builds.
- Podman passes container hostnames to Netavark, which uses them for any DHCP requests for the container.
- Packagers can set the **BUILD_ORIGIN** environment variable when building podman from the Makefile. This provides information on who built the Podman binary, and this information is displayed in the **podman version** and **podman info** commands. Including this information can assist with bug reports by helping maintainers to identify the source and method of the build and installation.
- The **podman kube generate** and **podman kube play** commands can create and run Kubernetes Job YAML.
- The **podman kube generate** command includes information on the user namespaces for pods and containers in the generated YAML. The **podman kube play** command uses this information to duplicate the user namespace configuration when creating new pods based on the YAML.
- The **podman kube play** command supports Kubernetes volumes of type image.
- The service name of **systemd** units generated by Quadlet can be set with the **ServiceName** key in all supported Quadlet files.
- Quadlets can disable their implicit dependency on **network-online.target** by using a new key, **DefaultDependencies**, supported by all Quadlet files.
- Quadlet **.container** and **.pod** files support a new key, **AddHost**, to add hosts to the container or pod.
- The **PublishPort** key in Quadlet **.container** and **.pod** files can accept variables in its value.
- Quadlet **.container** files support two new keys, **CgroupsMode** and **StartWithPod**, to configure control groups for the container and whether the container will be started with the pod that it is part of.
- Quadlet **.container** files can use the network of another container by specifying the **.container** file of the container to share within the Network key.
- Quadlet **.container** files can mount images managed by **.image** files into the container by using the **Mount=type=image** key with an **.image** target.
- Quadlet **.pod** files support six new keys, **DNS**, **DNSOption**, **DNSSearch**, **IP**, **IP6**, and **UserNS**, to configure **DNS**, static IPs, and user namespace settings for the pod.
- Quadlet **.image** files can give an image multiple times by specifying the **ImageTag** key multiple times.

- Quadlets can be placed in the **/run/containers/systemd** directory as well as existing directories, such as **$HOME/containers/systemd** and **/etc/containers/systemd/users**.
- Quadlet properly handles subdirectories of a unit directory that is a symlink.
- The **podman manifest inspect** command includes the manifest's annotations in its output.
- The **--add-host** option for **podman create**, **podman run**, and **podman pod create** supports specifying multiple hostnames, semicolon-separated (for example **podman run --add-host test1;test2:192.168.1.1**).
- The **podman run** and **podman create** commands support three new options for configuring health check logging: **--health-log-destination** (specifies where logs are stored), **--health-max-log-count** (specifies how many health checks worth of logs are stored), and **--health-max-log-size** (specifies the maximum size of the health check log).

For more information about notable changes, see upstream release notes.

Jira:RHEL-66762

Container tools use sigstore signatures for container image verification

With this update, sigstore signatures are used for container image verification instead of GPG signatures, also known as simple signing.

Jira:RHEL-32724

Podman health check log output can be customized

Before this update, when a container was configured with a health check, the output was only recorded in the container state file accessible by using the **podman inspect** command. It complicated the debugging process. With this enhancement, you can use the **podman update** command with the **--health-log-destination**, **--health-max-log-count**, and **--health-max-log-size** options to configure health check log output.

For more information, see the **podman-update** man page.

Jira:RHEL-24623[1]

Deploying a container image by using a single command is available

You can deploy a container image into a RHEL cloud instance by using a signal command. The **system-reinstall-bootc** command installs performs the following actions:

- Pull the supplied image to set up SSH keys or access the system.
- Run the **bootc install to-existing-root** command with all the bind mounts and SSH keys configured.

Jira:RHELDOCS-19516[1]

Creating custom bootc images from scratch is supported

You can create **bootc** images from scratch and fully control the contents of the image and tailor the system environment to meet specific requirements. With the **bootc-base-imgectl** command, you can create custom **bootc** images based on an existing **bootc** base image. Bootc Image from Scratch are derived from container images and do not automatically receive updates from the default base image.

To include such updates, you must incorporate them manually as part of your container pipeline. Additionally, you can use the **rechunk** subcommand in **bootc-base-imgectl** on any bootc container image to optimize or restructure the image as needed.

Jira:RHELDOCS-19825[1]

A new image build progressing bar available for bootc-image-builder

Previously, you could not check if an image build was progressing by looking into the logs. With this enhancement, you can check the progress of the image build that you created by using **bootc-image-builder**. You can revert to the previous behavior by using the **--progress=verbose** argument when building images.

Jira:RHELDOCS-20170[1]

The podman pod inspect command provides a JSON array regardless of the number of pods

Previously, the **podman pod inspect** command omitted the JSON array when inspecting a single pod. With this update, the **podman pod inspect** command produces a JSON array in the output regardless of the number of pods inspected.

Jira:RHELDOCS-18770[1]

6.24. LIGHTSPEED

The command line assistant powered by RHEL Lightspeed is now available in RHEL

The command line assistant powered by RHEL Lightspeed is available within the RHEL command line as an optional AI tool. The command line assistant includes knowledge from several Red Hat resources. It provides you with interactive workflows to solve issues, implement new RHEL features, find information, and more. As a result, you can experience more accessible and proactive guidance, and thus, enable your further adoption of RHEL.

Jira:RHELDOCS-20020[1]

The command-line assistant powered by RHEL Lightspeed is generally available in RHEL

The command-line assistant powered by RHEL Lightspeed is available within the RHEL command line. The generative AI that powers the assistant is trained on information from the RHEL product documentation and Red Hat Knowledgebase, and can help you to understand, configure, and troubleshoot your RHEL systems in a more accessible way, whether you are new to RHEL or already an experienced user.

Jira:RHELDOCS-20019[1]

The command-line assistant supports using the systemd-creds as a password store manager

The command-line assistant powered by RHEL Lightspeed integrates CLAD by using the **systemd-creds**, a password store manager shipped with RHEL. By using the assistant, you can securely store your passwords by using databases such as PostgreSQL or MySQL as your history backend. As a result, you can listing, showing, encrypting and decrypting unit credentials in a secure manner.

Jira:RHELDOCS-20023[1]

CHAPTER 7. TECHNOLOGY PREVIEW FEATURES

This part provides a list of all Technology Preview features available in Red Hat Enterprise Linux 10.

For information on Red Hat scope of support for Technology Preview features, see Technology Preview Features Support Scope.

7.1. SECURITY

System-wide post-quantum cryptography is available through crypto-policies-pq-preview as a Technology Preview

The **TEST-PQ** subpolicy contained in the new **crypto-policies-pq-preview** package provides system-wide post-quantum cryptography (PQC) as a Technology Preview. You can enable PQC by switching to the **TEST-PQ** subpolicy and restarting the system, for example:

```
# update-crypto-policies --set DEFAULT:TEST-PQ
# reboot
```

Note that all PQC algorithms in RHEL 10 are provided as a Technology Preview feature. The package and system-wide cryptographic policy name are subject to change when post-quantum cryptography exits the Technology Preview state. See the Post-quantum cryptography in Red Hat Enterprise Linux 10 article (Red Hat Blog) for more information.

Jira:RHEL-58241

RHEL 10 packages liboqs, oqsprovider, nss, openssh, and gnutls provide PQC as a Technology Preview

The RHEL 10.0 packages **liboqs**, **oqsprovider**, **nss**, **openssh**, and **gnutls** provide post-quantum cryptography (PQC) as a Technology Preview. To enable the PQC algorithms, install the **crypto-policies-pq-preview** package and apply the **TEST-PQ** cryptographic subpolicy.

For details, see the Interoperability of RHEL 10 post-quantum cryptography article (Red Hat Knowledgebase).

Jira:RHEL-65426, Jira:RHEL-65422, Jira:RHEL-58245, Jira:RHEL-58246

Encrypted DNS in RHEL is available as a Technology Preview

You can enable encrypted DNS to secure DNS communication that uses DNS-over-TLS (DoT). Encrypted DNS (eDNS) encrypts all DNS traffic end-to-end, with no fallback to insecure protocols, and aligns with zero trust architecture (ZTA) principles.

To perform a new installation with eDNS, specify the DoT-enabled DNS server by using the kernel command line. This ensures encrypted DNS is active during the installation process, boot time, and on the installed system. If you require a custom CA certificate bundle, you can install it only by using the **%certificate** section in the Kickstart file. Currently, the custom CA bundle can be installed only through Kickstart installation.

On an existing system, configure NetworkManager to use a new DNS plugin, **dnsconfd**, which manages the local DNS resolver (unbound) for eDNS. Add kernel arguments to configure eDNS for the early boot process, and optionally install a custom CA bundle.

Additionally, Identity Management (IdM) deployments can also use encrypted DNS, with the integrated DNS server supporting DoT.

See Securing system DNS traffic with encrypted DNS for more details.

Jira:RHELDOCS-20058[1], Jira:RHEL-67912

7.2. SOFTWARE MANAGEMENT

Support for signing packages with Sequoia PGP is available as a Technology Preview

The **macros.rpmsign-sequoia** macro file that configures RPM to use Sequoia PGP instead of GnuPG for signing packages is now available as a Technology Preview. To enable its usage, perform the following steps:

1. Install the following packages:

```
# dnf install rpm-sign sequoia-sq
```

2. Copy the **macros.rpmsign-sequoia** file to the **/etc/rpm/** directory:

```
$ cp /usr/share/doc/rpm/macros.rpmsign-sequoia /etc/rpm/
```

Jira:RHEL-56363[1]

7.3. SHELLS AND COMMAND-LINE TOOLS

The systemd-resolved service is available as a Technology Preview

The **systemd-resolved** service provides name resolution to local applications. The service implements a caching and validating DNS stub resolver, a Link-Local Multicast Name Resolution (LLMNR), and Multicast DNS resolver and responder.

Note that **systemd-resolved** is an unsupported Technology Preview.

Jira:RHEL-88550

7.4. KERNEL

The Red Hat Enterprise Linux for Real Time on ARM64 is now available as a Technology Preview

With this Technology Preview, the Red Hat Enterprise Linux for Real Time is now enabled for ARM64. The ARM64 is enabled on ARM (AARCH64), for both 4k and 64k ARM kernels.

Jira:RHELDOCS-19635[1]

7.5. FILE SYSTEMS AND STORAGE

ublk_drv driver is available as a Technology Preview

The **ublk_drv** kernel module is now enabled as a Technology Preview. It provides the **ublk** framework with which you can create and build high-performance block devices from userspace. Currently, **ublk**

requires userspace implementations, such as the Userspace Block Driver (**ublksrv**) or the Rust-based **ublk** (**rublk**), to function effectively.

Jira:RHELDOCS-19891[1]

NVMe/TCP using TLS is available as a Technology Preview

Encrypting Non-volatile Memory Express (NVMe) over TCP (NVMe/TCP) network traffic using TLS configured with Pre-Shared Keys (PSK) has been added as a Technology Preview in RHEL 10.0. For instructions, see Configuring an NVMe/TCP host using TLS with Pre-Shared-Keys .

Jira:RHELDOCS-19968[1]

xfs_scrub utility is available as a Technology Preview

You can check all the metadata on a mounted XFS file system by using the **xfs_scrub** utility as a Technology Preview. It functions similarly to the **xfs_repair -n** command for an unmounted XFS filesystem. For details, see the **xfs_scrub(8)** man page on your system. Note that currently only the scrub feature is available in RHEL 10 kernels and online repair is not enabled.

Jira:RHELDOCS-20041[1]

Limited shrinking of XFS file systems is available as Technology Preview

You can reduce the size of XFS file systems by using the **xfs_growfs** utility as a Technology Preview. You can remove blocks from the end of the file system by using **xfs_growfs**, provided that all of the following conditions are true:

- No metadata or data is allocated within the range to be removed.
- The requested size is within the last allocation group.

Jira:RHELDOCS-20042[1]

Mounting XFS file systems with blocks larger than system page is available as Technology Preview

You can now mount XFS file systems created with a block size larger than the system page size as a Technology Preview. For example, a file system with 16-KB blocks can now be mounted on a system with a 4-KB page size, such as x86_64.

Jira:RHELDOCS-20043[1]

io-uring interface is available as a Technology Preview

The **io_uring**, which is an asynchronous I/O interface, is available as a Technology Preview. By default, this feature is disabled in RHEL 10. You can enable this interface by setting the **kernel/io_uring_disabled** variable:

- For all users:

```
# echo 0 > /proc/sys/kernel/io_uring_disabled
```

- For root only:

```
# echo 1 > /proc/sys/kernel/io_uring_disabled
```

You can also disable **io_uring** for all processes:

```
# echo 2 > /proc/sys/kernel/io_uring_disabled
```

Jira:RHEL-65347

7.6. COMPILERS AND DEVELOPMENT TOOLS

eu-stacktrace available as a Technology Preview

The **eu-stacktrace** utility, which has been distributed through the **elfutils** package since version 0.192, is available as a Technology Preview feature. **eu-stacktrace** is a prototype utility that uses the **elfutils** toolkit's unwinding libraries to support a sampling profiler to unwind frame pointer-less stack sample data.

Jira:RHELDOCS-19072[1]

7.7. IDENTITY MANAGEMENT

DNSSEC available as Technology Preview in IdM

Identity Management (IdM) servers with integrated DNS now implement DNS Security Extensions (DNSSEC), a set of extensions to DNS that enhance security of the DNS protocol. DNS zones hosted on IdM servers can be automatically signed using DNSSEC. The cryptographic keys are automatically generated and rotated.

Users who decide to secure their DNS zones with DNSSEC are advised to read and follow these documents:

- DNSSEC Operational Practices, Version 2
- Secure Domain Name System (DNS) Deployment Guide
- DNSSEC Key Rollover Timing Considerations

Note that IdM servers with integrated DNS use DNSSEC to validate DNS answers obtained from other DNS servers. This might affect the availability of DNS zones that are not configured in accordance with recommended naming practices.

Jira:RHELPLAN-121751[1]

DNS over TLS (DoT) in IdM deployments is available as a Technology Preview

Encrypted DNS using DNS over TLS (DoT) is now available as a Technology Preview in Identity Management (IdM) deployments. You can now encrypt all DNS queries and responses between DNS clients and IdM DNS servers.

To start using this functionality, install the **ipa-server-encrypted-dns** package on IdM servers and replicas, and the **ipa-client-encrypted-dns** package on IdM clients. Administrators can enable DoT during the installation by using the **--dns-over-tls** option.

IdM configures Unbound as a local caching resolver and BIND to receive DoT requests. This functionality is available through the command-line interface (CLI) and noninteractive installations of IdM.

The following options were added to installation utilities for IdM servers, replicas, clients, and the integrated DNS service:

- **--dot-forwarder** to specify an upstream DoT-enabled DNS server.
- **--dns-over-tls-key** and **--dns-over-tls-cert** to configure DoT certificates.
- **--dns-policy** to set a DNS security policy to either allow fallback to unencrypted DNS or enforce strict DoT usage.

By default, IdM uses the **relaxed** DNS policy, which allows fallback to unencrypted DNS. You can enforce encrypted-only communication by using the new **--dns-policy** option with the **enforced** setting.

You can also enable DoT on an existing IdM deployment by reconfiguring the integrated DNS service by using **ipa-dns-install** with the new DoT options.

See Securing DNS with DoT in IdM for more details.

Jira:RHEL-67912, Jira:RHELDOCS-20058

IdM-to-IdM migration is available as a Technology Preview

IdM-to-IdM migration is available in Identity Management as a Technology Preview. You can use a new **ipa-migrate** command to migrate all IdM-specific data, such as SUDO rules, HBAC, DNA ranges, hosts, services, and more, to another IdM server. This can be useful, for example, when moving IdM from a development or staging environment into a production one or when migrating IdM data between two production servers.

Jira:RHELDOCS-18408[1]

logconv.py is available as a Technology Preview

The **logconv.py** utility is available in Directory Server as a Technology Preview. **logconv.py** is a future replacement for the old **logconv.pl** utility that you could use to analyze Directory Server access logs, extract usage statistics, and count occurrences of significant events.

The utility syntax:

```
logconv.py /var/log/dirsrv/slapd-<instance_name>/access
```

For more details about the utility options and usage examples, run the **logconv.py -h** command.

Jira:RHEL-59513

7.8. VIRTUALIZATION

AMD SEV, SEV-ES, and SEV-SNP for KVM virtual machines are available as a Technology Preview

As a Technology Preview, RHEL provides the Secure Encrypted Virtualization (SEV) feature for AMD EPYC host machines that use the KVM hypervisor. If enabled on a virtual machine (VM), SEV encrypts the VM's memory to protect the VM from access by the host. This increases the VM security.

In addition, the enhanced Encrypted State version of SEV (SEV-ES) is also provided as Technology Preview. SEV-ES encrypts all CPU register contents when a VM stops running. This prevents the host from modifying the VM's CPU registers or reading any information from them.

RHEL also provides the Secure Nested Paging (SEV-SNP) feature as Technology Preview. SNP enhances SEV and SEV-ES by improving its memory integrity protection, which helps to prevent hypervisor-based attacks, such as data replay or memory re-mapping.

Note that: * SEV and SEV-ES work only on the 2nd generation of AMD EPYC CPUs (codenamed Rome) or later. * SEV-SNP works only on 3rd generation AMD EPYC CPUs (codenamed Milan) or later.

Also note that RHEL includes SEV, SEV-ES, and SEV-SNP encryption, but not the SEV, SEV-ES, and SEV-SNP security attestation and live migration.

Jira:RHELDOCS-16800[1]

Creating nested virtual machines

Nested KVM virtualization is provided as a Technology Preview for KVM virtual machines (VMs) running on Intel, AMD64, and IBM Z hosts with RHEL 10. With this feature, a RHEL 7, RHEL 8, or RHEL 9 VM that runs on a physical RHEL 10 host can act as a hypervisor, and host its own VMs.

Jira:RHELDOCS-20080[1]

New package: trustee-guest-components

As a Technology Preview, this update adds the **trustee-guest-components** package. This makes it possible for confidential virtual machines to attest themselves and get confidential resources from a Trustee server.

Jira:RHEL-73770[1]

7.9. CONTAINERS

composefs filesystem is available as a Technology Preview

The key technologies **composefs** uses are:

- OverlayFS as the kernel interface
- Enhanced Read-Only File System (EROFS) for a mountable metadata tree
- The **fs-verity** feature (optional) from the lower filesystem

Key advantages of **composefs**:

- Separation between metadata and data. **composefs** does not store any persistent data. The underlying metadata and data files are stored in a valid lower Linux filesystem such as **ext4**, **xfs**, **btrfs**, and so on.
- Mounting multiple **composefs** with a shared storage.
- Data files are shared in the page cache to enable multiple container images to share their memory.
- Support **fs-verity** validation of the content files.

Jira:RHEL-52238

The composefs file system is available as Technology Preview

The composefs read-only file system available as Technology Preview is generally intended only to be used by the bootc/ostree and podman projects at the current time. With composefs, you can use these projects to create and use read-only images, share file data between images, and validate images on runtime. As a result, you have a fully verified file-system tree mounted, with opportunistic fine-grained sharing of identical files.

Jira:RHEL-18157[1]

Partial pulls for zstd:chunked are available as a Technology Preview

You can pull only the changed parts of the container images compressed with the **zstd:chunked** format, reducing network traffic and necessary storage. You can enable partial pulls by adding the **enable_partial_images = "true"** setting to the **/etc/containers/storage.conf** file. This functionality is available as a Technology Preview.

Jira:RHEL-32266

The podman artifact command is available as a Technology Preview

The **podman artifact** command, which you can use to work with OCI artifacts at the command-line level, is available as a Technology Preview. For further informal, reference the man page.

Jira:RHEL-70218

The vrf option for the podman network create is available as a Technology Preview

The **podman network create** command provides the **vrf** value for the **--opt** option, as a Technology Preview. The **vrf** value assigns a virtual routing and forwarding instance (VRF) to the bridge interface. It accepts the name of the VRF and defaults to none.

This option can only be used with the Netavark network backend.

Jira:RHEL-89373

7.10. TECHNOLOGY PREVIEW FEATURES IDENTIFIED IN PREVIOUS RELEASES

This part provides a list of all Technology Previews available in Red Hat Enterprise Linux 10.

For information on Red Hat scope of support for Technology Preview features, see Technology Preview Features Support Scope.

7.10.1. Networking

WireGuard VPN is available as a Technology Preview

WireGuard, which Red Hat provides as an unsupported Technology Preview, is a high-performance VPN solution that runs in the Linux kernel. It uses modern cryptography and is easier to configure than other VPN solutions. Additionally, the small code-basis of WireGuard reduces the surface for attacks and, therefore, improves the security.

For further details, see Setting up a WireGuard VPN .

Jira:RHELDOCS-20056[1]

KTLS available as a Technology Preview

In RHEL, Kernel Transport Layer Security (KTLS) is provided as a Technology Preview. KTLS handles TLS records by using the symmetric encryption or decryption algorithms in the kernel for the AES-GCM cipher. KTLS also includes the interface for offloading TLS record encryption to Network Interface Controllers (NICs) that provides this functionality.

Note that specific uses cases of kernel TLS offload might have a higher support status. For details see the release notes in the New features and enhancements chapter.

Jira:RHELDOCS-20440[1]

NetworkManager enables configuring HSR and PRP interfaces

High-availability Seamless Redundancy (HSR) and Parallel Redundancy Protocol (PRP) are network protocols that provide seamless failover against failure of any single network component. Both protocols are transparent to the application layer, meaning that users do not experience any disruption in communication or any loss of data, because a switch between the main path and the redundant path happens very quickly and without awareness of the user. Now it is possible to enable and configure HSR and PRP interfaces using the **NetworkManager** service through the **nmcli** utility and the DBus message system.

Jira:RHEL-5852

The PRP and HSR protocols are now available as a Technology Preview

This update adds the **hsr** kernel module that provides the following protocols:

- Parallel Redundancy Protocol (PRP)
- High-availability Seamless Redundancy (HSR)

The IEC 62439-3 standard defines these protocols, and you can use this feature to configure redundancy with zero-time recovery in Ethernet networks.

Jira:RHELDOCS-20472[1]

CHAPTER 8. REMOVED FEATURES

All removed features were deprecated in earlier releases and are no longer supported. For information regarding functionality that is present in RHEL 9 but has been *removed* in RHEL 10, see Considerations in adopting RHEL 10.

8.1. INSTALLER AND IMAGE CREATION

auth or authconfig commands are removed

The **auth** or **authconfig** Kickstart commands which were deprecated in Red Hat Enterprise Linux 8, are removed now. As a replacement, use the **authselect** kickstart command.

Jira:RHELDOCS-18839[1]

The inst.xdriver and inst.usefbx options have been removed

The graphical system for the installation image switched from the Xorg server to a Wayland compositor. As a consequence, the **inst.xdriver** boot option has been removed. Wayland operates without relying on X drivers, making it incompatible with loading any such drivers. As a result, the **inst.xdriver** option is no longer applicable.

Additionally, the **inst.usefbx** boot option, previously used to load a generic framebuffer X driver, has also been removed.

Jira:RHELDOCS-18818[1]

The OpenStack image type has been deprecated from RHEL image builder

From the RHEL 10.0 onward, RHEL image builder will no longer support the OpenStack image type. You can use the **.qcow2** image type to build OpenStack images.

Jira:RHELDOCS-18736[1]

Capturing screenshots from the Anaconda GUI with a global hot key is removed

Previously, users could capture screenshots of the Anaconda GUI by using a global hot key. Consequently, users could extract the screenshots manually from the installation environment for any further usage. This functionality has been removed.

Jira:RHELDOCS-18492[1]

Removed inst.nompath, dmraid and nodmraid boot options

The **inst.nompath**, **dmraid** and **nodmraid** boot options have been removed now and are no longer available for use.

Jira:RHELDOCS-18485[1]

Removed automatic bug reporting system from Anaconda

The installation program no longer supports automatically reporting problems to the Red Hat issue tracking system. You can collect the installation logs and report problems manually, as described in the troubleshooting section.

Jira:RHELDOCS-18426[1]

Removed a few options of the timezone Kickstart command

The following options of the **timezone** Kickstart command has been removed in Red Hat Enterprise Linux 10:

- **--isUtc**: Use the option **--utc** instead.
- **--ntpservers**: Use the option **--ntp-server** of the timesource Kickstart command instead.
- **--nontp**: Use the option **--ntp-disable** of the **timesource** Kickstart command instead.

Jira:RHELDOCS-18423[1]

Removed the --level parameter of the logging Kickstart command

The **--level** parameter of the logging Kickstart command has been removed. It is no longer possible to set the level of logging of the installation process.

Jira:RHELDOCS-18417[1]

The support for %anaconda Kickstart command has been removed

The support for the deprecated %anaconda Kickstart command has been removed. You can use the kernel arguments and command line options to update the configuration in the **Anaconda configuration files**.

Jira:RHELDOCS-18416[1]

Removed pwpolicy Kickstart command

The support for the deprecated **pwpolicy** Kickstart command has been removed in Red Hat Enterprise Linux 10.

Jira:RHELDOCS-18415[1]

Removed support for adding additional repositories from GUI

Previously, when configuring the installation source, you could configure the additional repositories for the package installation. Starting in RHEL 10, this support has been removed. However, you can use the Kickstart installation method or **inst.addrepo** boot option if you want to specify additional repositories.

Jira:RHELDOCS-18413[1]

Removed support of the LUKS version selection from Anaconda

Previously, you could select the LUKS version from the Manual Installation screen. Starting in RHEL 10, the installation program uses the **luks2** version by default for all the new devices. No changes are made to the existing devices' LUKS version. You can also use the Kickstart method to select different LUKS versions.

Jira:RHELDOCS-18412[1]

The initial-setup package now has been removed

The initial-setup package has been removed in Red Hat Enterprise Linux 10. As a replacement, use **gnome-initial-setup** for the graphical user interface.

Jira:RHELDOCS-18411[1]

Redesigned the Time & Date spoke in the installation program GUI

Previously, Anaconda users were able to select the time zone by using the time zone map. This screen is now redesigned and the time zone map has been replaced with the options where users can set the required time zone.

For more information, refer to the installation documentation.

Jira:RHELDOCS-18410[1]

Anaconda built-in help has been removed

The built-in documentation from spokes and hubs of all Anaconda user interfaces, which was available during Anaconda installation, has been removed. Instead, refer to the official RHEL documentation.

Jira:RHELDOCS-18414[1]

Removed teaming options from the network Kickstart command

The **--teamslaves** and **--teamconfig** options used for configuring team devices in the **network** Kickstart command have been removed. To configure similar network settings, use the **--bondslaves** and **--bondopts** options to set up a **Bond** device.

Jira:RHEL-33892

Removed NVDIMM reconfiguration support during the installation process

The support for reconfiguring NVDIMM devices during the Kickstart and GUI installation has been removed in RHEL-10. However, the NVDIMM devices in the sector mode can still be usable in the installation program.

Jira:RHELDOCS-19084

The --excludeWeakdeps and --instLangs options from %packages have been removed

In RHEL-10, the **--excludeWeakdeps** and **--instLangs** options used in the **%packages** section have been removed. To maintain similar functionality, use the updated options **--exclude-weakdeps** and **--inst-langs** instead. These replacements ensure compatibility and provide the same dependency and language control within package management.

Jira:RHELDOCS-19083

8.2. SECURITY

scap-workbench is removed

The **scap-workbench** package is removed in RHEL 10. The **scap-workbench** graphical utility was designed to perform configuration and vulnerability scans on a single local or remote system. As an alternative, you can scan local systems for configuration compliance by using the **oscap** command and remote systems by using the **oscap-ssh** command. For more information, see Configuration compliance scanning.

Jira:RHELDOCS-19009[1]

oscap-anaconda-addon is removed

The **oscap-anaconda-addon**, which provided means to deploy baseline-compliant RHEL systems by using the graphical installation, is removed in RHEL 10. As an alternative, you can build RHEL images that comply with a specific standard by Creating pre-hardened images with RHEL image builder OpenSCAP integration.

Jira:RHELDOCS-19010[1]

CVE OVALv2 feed no longer provided

The Common Vulnerabilities and Exposures (CVE) Open Vulnerability Assessment Language (OVAL) version 2 feed, which contained declarative security data processed by the OpenSCAP suite, is not provided for RHEL 10. Red Hat continues to provide declarative security data in the Common Security Advisory Framework (CSAF) format and Vulnerability Exploitability eXchange (VEX) files, which are the successors of the CVE OVALv2 feed. The OpenSCAP suite retains the OVAL module, and therefore can still consume the OVAL data format.

For more information, see the OVAL v2 Announcement.

Alternatively, you can us the Insights for RHEL vulnerability service, for more information, follow Assessing and Monitoring Security Vulnerabilities on RHEL Systems .

Jira:RHELDOCS-19071[1]

DSA and SEED algorithms have been removed from NSS

The Digital Signature Algorithm (DSA), which was created by the National Institute of Standards and Technology (NIST) and is now completely deprecated by NIST, is removed from the Network Security Services (NSS) cryptographic library. You can instead use algorithms such as RSA and ECDSA.

The SEED algorithm, which was created by the Korea Information Security Agency (KISA) and has been previously disabled upstream, is removed from the NSS cryptographic library.

Jira:RHEL-44995

fips-mode-setup is removed

The **fips-mode-setup** command is removed from RHEL. To enable the cryptographic module self-checks mandated by the Federal Information Processing Standard (FIPS) 140, enable FIPS mode during the system installation. See the Switching RHEL to FIPS mode chapter in the *Security hardening* document for more information.

Jira:RHEL-65652

/etc/system-fips removed

Support for indicating FIPS mode through the **/etc/system-fips** file has been removed from RHEL. To install RHEL in FIPS mode, add the **fips=1** parameter to the kernel command line during the system installation. You can check whether RHEL operates in FIPS mode by displaying the **/proc/sys/crypto/fips_enabled** file.

Jira:RHELDOCS-19357[1]

HeartBeat removed from TLS

The support for the HeartBeat extension in TLS has been removed to reduce the attack surface.

Jira:RHEL-59212[1]

SRP authentication removed from TLS

Authentication that uses Secure Remote Password protocol (SRP) in TLS has been removed from the **gnutls** package and is no longer supported. SRP authentication is considered insecure because it cannot be used with TLS 1.3 and relies on Cipher block chaining (CBC) and SHA-1 as a key exchange.

Jira:RHEL-58640[1]

Keylime no longer supports HTTP for revocation notifications

The Keylime components no longer support the HTTP protocol for revocation notification webhooks. Use HTTPS instead. As a consequence, the Keylime verifier now requires the revocation notification webhook server CA certificate. You can add it to the **trusted_server_ca** configuration option or add it to the system truststore.

Jira:RHEL-51279

DEFAULT cryptographic policy rejects TLS ciphers with RSA key exchange

TLS ciphers that use the RSA key exchange are no longer accepted in the **DEFAULT** system-wide cryptographic policy in RHEL 10. These ciphers do not provide perfect forward secrecy and are not considered as secure as ciphers that use other key exchanges, for example, the Elliptic-curve Diffie-Hellman (ECDH) key exchange.

This change also reduces the exposure to side-channel attacks because the RSA key exchange uses PKCS #1 v1.5 encryption padding, which can cause vulnerability to timing side-channel attacks.

If you need the RSA key exchange for interoperability with legacy systems, you can re-enable it by using the LEGACY system-wide cryptographic policy or by applying a custom subpolicy.

Jira:RHEL-50464[1]

ca-certificates truststore moved

The **/etc/pki/tls/certs** truststore is converted to a different format better optimized for OpenSSL. As a consequence, if you use the files in **/etc/pki/tls/certs** directly, switch to the **/etc/pki/ca-trust/extracted** directory, where the same data is stored. For example, software that accesses the trust bundle at **/etc/pki/tls/certs/ca-bundle.crt** should switch to using **/etc/pki/ca-trust/extracted/pem/tls-ca-bundle.pem** instead.

Jira:RHEL-50293

The LEGACY cryptographic policy disallows SHA-1 signatures in TLS

The **LEGACY** system-wide cryptographic policy in RHEL 10 no longer allows creating or verifying signatures that use SHA-1 in TLS contexts. Therefore, libraries other than OpenSSL might no longer accept or create any signatures that use SHA-1 regardless of use case. OpenSSL continues to accept signatures that use SHA-1 when not used for TLS if the system is in **LEGACY** or this functionality is re-enabled with a custom subpolicy.

Jira:RHEL-50106

pam_ssh_agent_auth is removed

The **pam_ssh_agent_auth** package has been removed from RHEL 10.

Jira:RHEL-45002

OpenSSL no longer permits SHA-1 at SECLEVEL=2 in TLS

OpenSSL does not accept the SHA-1 algorithm at **SECLEVEL=2** in TLS in RHEL 10. If your scenario requires using TLS 1.0/1.1, you must explicitly set **SECLEVEL=0** and switch to the LEGACY system-wide cryptographic policy. In the LEGACY policy, applications that use SHA-1 in signatures outside of TLS will continue to work.

Jira:RHEL-39962

stunnel does not support OpenSSL ENGINE API

The **stunnel** TLS offloading and load-balancing proxy no longer supports the previously deprecated OpenSSL ENGINE API. The most common use case was accessing hardware security tokens by using PKCS #11 through the **openssl-pkcs11** package. As a replacement, you can use the **pkcs11-provider**, which uses the new OpenSSL provider API.

Jira:RHEL-33749

OpenSSL Engines removed from OpenSSL

OpenSSL Engines have been deprecated and will soon be removed from upstream. Therefore, the **openssl-pkcs11** package has been removed from OpenSSL in RHEL 10. Use providers instead, such as the **pkcs11-provider**, which is supported in this version.

Jira:RHEL-30437

Keylime policy management scripts are removed and replaced with keylime-policy

In RHEL 10, Keylime is provided with the **keylime-policy** tool, which replaces the following policy management scripts:

- **keylime_convert_runtime_policy**
- **keylime_create_policy**
- **keylime_sign_runtime_policy**
- **create_mb_refstate**
- **create_allowlist.sh**

The scripts have been removed and are no longer provided in RHEL 10.

Jira:RHEL-79831

8.3. SUBSCRIPTION MANAGEMENT

Several subscription-manager modules have been removed

Because of a simplified customer experience in Red Hat subscription services, which have transitioned to the Red Hat Hybrid Cloud Console and to account level subscription management with Simple Content Access, the following previously deprecated modules have been removed:

- **addons**
- **attach**

- **auto-attach**
- **import**
- **remove**
- **redeem**
- **role**
- **service-level**
- **usage**
- **syspurpose addons**

For more information about these changes, see the Transition of Red Hat's subscription services to the Red Hat Hybrid Cloud Console article.

Jira:RHELDOCS-18989[1]

8.4. SOFTWARE MANAGEMENT

The support for the libreport library has been removed

The support for the **libreport** library has been removed from DNF. If you want to attach DNF logs to your bug reports, you need to do it manually or by using a different mechanism.

Jira:RHEL-40382

The DNF debug plugin has been removed

The DNF **debug** plugin, which included the **dnf debug-dump** and **dnf debug-restore** commands, has been removed from the **dnf-plugins-core** package. Depending on your scenario, you can use one of the following commands instead:

- **dnf list --installed** or **dnf repoquery --installed** to list packages installed on your system.
- **dnf repolist -v** to list repositories enabled on your system.
- **dnf install $(</tmp/list)** to replicate packages installed on a source system to the target system. For example:

  1. Save a list of packages installed on a source system into the **/tmp/list** file:

     ```
     $ dnf repoquery --installed >/tmp/list
     ```

  2. Copy the **/tmp/list** file to the target system.
  3. Replicate packages on the target system:

     ```
     $ dnf install $(</tmp/list)"
     ```

Jira:RHEL-23706[1]

The numberless %patch syntax has been removed

Using the **%patch** directive without a number specified as a shorthand for **%patch 0** to apply the **zero-th** patch has been removed. If you want to use **%patch**, a warning message suggests you to use the explicit syntax, for example, **%patch 0** or **%patch -P 0** to apply the **zero-th** patch.

Jira:RHELDOCS-19811[1]

8.5. SHELLS AND COMMAND-LINE TOOLS

The perl(Mail::Sender) module has been removed

The **perl(Mail::Sender)** module is removed from RHEL 10 without any replacement. As a consequence, the **checkbandwidth** script from **net-snmp-perl** package does not support email alerts when bandwidth high or low levels for a host or interface are reached.

Jira:RHEL-44478[1]

8.6. INFRASTRUCTURE SERVICES

Significant changes in the package set for infrastructure services

The following packages are no longer included in Red Hat Enterprise Linux:

- **sendmail** : Red Hat recommends migrating to the postfix mail daemon, that is supported.
- **redis** : Red Hat recommends migrating to the **valkey** package.
- **dhcp** : Red Hat recommends migrating to the available alternatives such as **dhcpcd** and **ISC Kea**.
- **mod_security**: The **mod_security** directive is now available in the EPEL repository.
- **spamassassin** : The Spamassassin mail filter is now available in the EPEL repository instead of the standard RHEL repository as it depends on the **libdb** (Berkeley DB) library, which is unavailable due to licensing issues.
- **xsane** : The API is not yet ported to **Gtk3**.

The following packages have been renamed:

- **gpsd** : It was previously included as **gpsd-minimal**.

Jira:RHEL-22424[1]

The ISC Kea DHCP server solution is now available

Kea is a new Dynamic Host Configuration Protocol (DHCP) server solution in RHEL. **ISC DHCP** was removed in favor of Kea to ensure future delivery of bug fixes and enhancements. The upstream version of **ISC DHCP** no longer develops new features and bug fixes.

Jira:RHEL-14710[1]

8.7. NETWORKING

Network team driver was removed

The **teamd** service and the **libteam** library were removed in Red Hat Enterprise Linux 10. As a replacement, configure a bond instead of a network team.

Red Hat focuses its efforts on kernel-based bonding to avoid maintaining two features, bonds and teams, that have similar functions. The bonding code has a high customer adoption, is robust, and has an active community development. As a result, the bonding code receives enhancements and updates.

If you use RHEL 9 with a network team and plan to upgrade to RHEL 10, migrate the network team configuration to network bond before you upgrade.

Jira:RHELDOCS-20862[1]

ATM encapsulation is removed from RHEL 10

Asynchronous Transfer Mode (ATM) encapsulation enables Layer-2 (Point-to-Point Protocol, Ethernet) or Layer-3 (IP) connectivity for the ATM Adaptation Layer 5 (AAL-5). Red Hat has not been providing support for ATM NIC drivers since RHEL 7. The support for ATM implementation is being dropped in RHEL 9. These protocols are currently used only in chipsets that support the ADSL technology and are being phased out by manufacturers. Therefore, ATM encapsulation was deprecated in Red Hat Enterprise Linux 9, and it is removed from Red Hat Enterprise Linux 10.

For more information, see PPP Over AAL5, Multiprotocol Encapsulation over ATM Adaptation Layer 5 , and Classical IP and ARP over ATM .

Jira:RHELDOCS-20102[1]

The dhcp-client package has been removed

The **dhcp-client** package has been removed from RHEL 10, because the ISC DHCP client is no longer maintained upstream. As a consequence, the **dhclient** utility is no longer available and you cannot use it as DHCP client in NetworkManager. As an alternative, use the NetworkManager-internal DHCP client, which was also the default in previous RHEL versions.

Jira:RHEL-46211

The mlx4 driver is removed from RHEL 10.0

With the RHEL 10.0 release, the **mlx4** driver for the Mellanox ConnectX-3 network interface controller (NIC) is removed. You must use another NIC that is compatible with newer drivers.

Jira:RHEL-40070[1]

8.8. KERNEL

The kexec_load system call is removed

The **kexec_load** system call, which was deprecated in RHEL 9, is removed. In RHEL 10, the **kexec_file_load** system call replaces **kexec_load** and is the default system call on all architectures. Also, **kexec_file_load** is required for a secure boot.

For more information, see Is kexec_load supported in RHEL9?

Jira:RHEL-29272[1]

The crash --log dumpfile option is deprecated

The **crash --log dumpfile** option no longer works for kernel version 5.10 and above. As a consequence, the **crash --log** command fails with error message 'crash:VMCOREINFO: no log buffer data'.

You can use **makedumpfile --dump-dmesg** as a workaround, to dump the kernel dmesg info with only **vmcore**.

Jira:RHEL-52221[1]

8.9. FILE SYSTEMS AND STORAGE

Support for NVMe devices has been removed from the lsscsi package

Support for Non-volatile Memory Express (NVMe) devices has been removed from the **lsscsi** package. Use native tools such as **nvme-cli**, **lsblk**, and **blkid** instead. Report any missing functionality against the **nvme-cli** package.

Jira:RHEL-32144[1]

Support for NVMe devices has been removed from the sg3_utils package

Support for Non-volatile Memory Express (NVMe) devices has been removed from the **sg3_utils** package. Use native tools such as the **nvme-cli** package instead and report any missing functionality against **nvme-cli**.

Jira:RHEL-412[1]

The VDO sysfs parameters have been removed

The Virtual Data Optimizer (VDO) **sysfs** parameters have been removed. Except for **log_level**, all module-level **sysfs** parameters for the **kvdo** module are removed. For individual **dm-vdo** targets, all **sysfs** parameters specific to VDO are also removed. There is no change for the parameters that are common to all DM targets. Configuration values for **dm-vdo** targets that are currently set by updating the removed module-level parameters, can no longer be changed.

Statistics and configuration values for **dm-vdo** targets are no longer be accessible through **sysfs**. But these values are still accessible by using **dmsetup message stats**, **dmsetup status**, and **dmsetup table** dmsetup commands.

Jira:RHELDOCS-19066[1]

Support for GFS2 file systems has been removed

The Red Hat Enterprise Linux (RHEL) Resilient Storage Add-On will no longer be supported starting with Red Hat Enterprise Linux 10. This includes the GFS2 file system, which is also no longer supported. The RHEL Resilient Storage Add-On will continue to be supported with earlier versions of RHEL (7, 8, 9) and throughout their specific maintenance support lifecycles.

Jira:RHELDOCS-19024[1]

Support for the block translation table driver has been removed

Support for the block translation table driver (btt.ko) has been removed. As a consequence, you cannot use sector mode when configuring Non-Volatile Dual In-line Memory Modules (NVDIMM) namespaces.

Jira:RHEL-68504[1]

The nvme_core.multipath parameter has been removed

In RHEL 10, the use of DM multipath with NVMe devices over RDMA and FC is no longer supported. As a consequence, the **nvme_core.multipath** parameter has been removed, the native NVMe multipath is enabled by default, and it can no longer be disabled.

Bug fixes and support for using DM multipath with NVMe devices over RDMA and FC are provided only through the end of the RHEL 9 lifecycle. Note that DM multipath was never supported with NVMe over TCP in any version of RHEL.

Jira:RHEL-78133[1]

The md-faulty and md-multipath modules have been removed

In RHEL 10, the **md-faulty** and **md-multipath** MD RAID kernel modules are no longer available. Bug fixes and support are provided only through the end of the RHEL 9 lifecycle.

Jira:RHELDOCS-19828[1]

8.10. HIGH AVAILABILITY AND CLUSTERS

pcsd Web UI no longer available as a standalone user interface

The **pcsd** Web UI has been modified to be usable as a RHEL web console add-on and is no longer operated as a standalone interface.

Jira:RHEL-29739

Support for the RHEL Resilient Storage Add-On has been removed

The Red Hat Enterprise Linux (RHEL) Resilient Storage Add-On will no longer be supported starting with Red Hat Enterprise Linux 10 and any subsequent releases after RHEL 10. The RHEL Resilient Storage Add-On will continue to be supported with earlier versions of RHEL (7, 8, 9) and throughout their respective maintenance support lifecycles.

Jira:RHELDOCS-19023[1]

Removed and updated Pacemaker CIB elements

The following configuration components of the Pacemaker CIB have been removed or modified in RHEL 10. When you upgrade to RHEL 10, these components are automatically removed, modified, or replaced as described. Before you upgrade, ensure that the Pacemaker CIB has a supported value for the **validate-with** attribute. Although you should not edit the cluster configuration file directly, you can view the raw cluster configuration with the **pcs cluster cib** command.

An upgrade modifies the following CIB components:

- The **validate-with** attribute of the **cib** element, which is set to **pacemaker-4.0**
- The **stonith-action** cluster property, which is set to **off** if it was previously set to **poweroff**
- Legacy promotable clone (master) resources, which are changed to standard promotable clones by changing the **master** XML element to the **clone** XML element and by setting the **promotable** meta attribute

- Location constraints with more than one top-level rule, which are converted to separate location constraints for each top-level rule

An upgrade renames the following components:

- The **crmd-finalization-timeout** cluster property, which is renamed to **join-finalization-timeout**
- The **crmd-integration-timeout** cluster property, which is renamed to **join-integration-timeout**
- The **crmd-transition-delay** cluster property, which is renamed to **transition-delay**

An upgrade removes the following components from the CIB:

- **nagios-class** and **upstart-class** resources
- **bundle** resources based on an **rkt** container.
- The **restart-type** resource meta-attribute
- The **can_fail** operation meta-attribute
- The **role_after_failure** operation **meta-attribute**
- **moon** attributes in **date_spec** elements of rules
- The **remove-after-stop** cluster property.
- Ping nodes, which are changed to cluster member nodes with all resources banned and probes disabled
- NVpairs without a value attribute
- Duplicate NVpairs with a given name within an NVset, for which only the first NVpair is kept

An upgrade changes the following default values:

- An action configured as a fence device parameter is now ignored rather than treated as a default fencing action.
- The **concurrent-fencing** cluster option now defaults to **true** and is deprecated.
- The **globally-unique** clone option now defaults to **true** when **clone-node-max** is greater than 1.

An upgrade removes **lifetime** elements, and modifies the CIB as follows:

- **lifetime** elements in a location constraint are removed.
  - If the **lifetime** element in a location constraint has no top-level rules, the **lifetime**-based rule becomes the constraint's top-level rule.
  - If the **lifetime** element in a location constraint has multiple top-level rules, they are nested inside a single **or** rule.
  - If the **lifetime** element in a location constraint has a single top-level rule, a new and top-level constraint rule is added that contains the existing top-level constraint rule and the **lifetime**-based rule.

- **lifetime** elements in a colocation or order constraint are removed. If any rules contained in the colocation or order constraint are referenced elsewhere, they are put in a new location constraint that does not apply to any resources. They are put in a location constraint since a rule in a **lifetime** element might contain a node attribute expression, which is now allowed only within a location constraint rule.
- Following an upgrade, invalid fencing levels display a warning when the CIB is loaded.

Jira:RHELDOCS-19813[1]

Removed functionality for the Red Hat High Availability Add-On

The following Red Hat High Availability Add-On features are no longer supported in RHEL 10.

- RKT containers in bundles. Docker and Podman containers are still supported.
- The **upstart** and **nagios** resource classes.
- Location constraints with multiple top-level rules. Only one rule per constraint is allowed. The **pcs constraint rule add**, **pcs constraint rule delete** and the **pcs constraint rule remove** commands have been removed. If you have configured constraints with multiple rules, run the **pcs cluster cib-upgrade** command to update to the latest CIB schema. During the update, Pacemaker creates a constraint for each rule, so that there will be only one rule in each constraint.
- The **monthdays**, **moon**, **weekdays**, **weekyears**, and **yearsdays** duration options for Pacemaker rules.
- Using spaces in dates in location constraint rules.
- Delimiting stonith devices with a comma in **pcs stonith level add | clear | delete | remove** commands.
- Ambiguous syntax of the **pcs stonith level clear | delete | remove** command. The command has been clarified to distinguish a target from a stonith device.
- The legacy role names of **master** and **slave** are no longer accepted by the **pcs** command-line interface. Use **Promoted**, **Unpromoted**, --promoted, **promotable**, and **promoted-max** instead.
- Using stonith resources in **pcs resource** commands and resources in **pcs stonith** commands, and the **--brief**, **--no-strict**,**--safe** and **--simulate** flags of the **pcs stonith disable** command.
- Ability to create a stonith resource in a group with the **pcs stonith create** command.
- The **stonith.create_in_group** command from API v1 and v2.
- The **pcs cluster pcsd-status** command. Use the **pcs status pcsd** or **pcs pcsd status** command.
- The **pcs cluster certkey** command. Use the **pcs pcsd certkey** command.
- The **pcs resource | stonith [op] defaults <name>=<value>...** command. Use the **pcs resource | stonith [op] defaults update** command.
- The **pcs acl show** command. Use the **pcs acl config** command.

- The **pcs alert show** command. Use the **pcs alert config** command.
- The **pcs constraint [location | colocation | order | ticket] show | list** commands. Use the **pcs constraint [location | colocation | order | ticket] config** command.
- The **pcs property show** and the **pcs property list** commands. Use the **pcs property config** command.
- The **pcs tag list** command. Use the **pcs tag config** command.
- The **--autodelete** flag of the **pcs resource move** command.

Jira:RHEL-49521, Jira:RHEL-62719, Jira:RHEL-49524, Jira:RHEL-49520

8.11. COMPILERS AND DEVELOPMENT TOOLS

32-bit packages have been removed in RHEL 10

Linking against 32-bit multilib packages has been removed. The ***.i686** packages remain supported for the life cycle of Red Hat Enterprise Linux 9.

Jira:RHELDOCS-19269

8.12. IDENTITY MANAGEMENT

The pam_console module has been removed

The **pam_console** module has been removed from RHEL 10. The **pam_console** module granted file permissions and authentication capabilities to users logged in at the physical console or terminals, and adjusted these privileges based on console login status and user presence. As an alternative to **pam_console**, you can use the **systemd-logind** system service instead. For configuration details, see the **logind.conf(5)** man page.

Jira:RHELDOCS-18159[1]

The RSA PKINIT method has been removed

The private key-based RSA method is no longer supported in the MIT Kerberos. It has been removed for security reasons, especially for its vulnerability to the Marvin attack. As a result, the **-X flag_RSA_PROTOCOL** parameter of the **kinit** commands has no effect anymore. The Diffie-Hellman key agreement method is used as the default PKINIT mechanism.

Jira:RHEL-56070[1]

The NIS server emulator has been removed

RHEL Identity Management (IdM) does not provide the NIS functionality anymore.

Jira:RHEL-34186

Other removed functionality for RHEL Identity Management

The following packages were part of RHEL 9 but are not distributed with RHEL 10:

- **compat-hesiod**

- **fontawesome-fonts**: consider using **fontawesome4-fonts** instead
- **libnsl2**
- **python3-netifaces**: consider using **python-ifaddr** instead

Jira:RHEL-33818

BDB is no longer supported in 389-ds-base

The **libdb** library that implements the Berkeley Database (BDB) version used by **389-ds-base** is no longer available in RHEL 10. As a result, Directory Server no longer supports BDB.

As a replacement, Directory Server creates instances with Lightning Memory-Mapped Database (LMDB).

Jira:RHEL-30640

8.13. SSSD

The enumeration feature has been removed for AD and IdM

Support for the **enumeration** feature was deprecated for AD and IdM in Red Hat Enterprise Linux (RHEL) 9. The **enumeration** feature has been removed for AD and IdM in RHEL 10.

Jira:RHELDOCS-19005

The libsss_simpleifp subpackage has been removed

The **libsss_simpleifp** subpackage that provided the **libsss_simpleifp.so** library was deprecated in Red Hat Enterprise Linux (RHEL) 9. The **libsss_simpleifp** subpackage has been removed in RHEL 10.

Jira:RHELDOCS-19094

The SSSD files provider has been removed

The SSSD files provider has been removed from RHEL 10.0. Previously, the SSSD files provider was responsible for smart card authentication and session recording for local users. As a replacement, you can configure the SSSD proxy provider.

Due to the removal of the files provider, the **authselect minimal** profile has been replaced by a new **local** profile.

Jira:RHELDOCS-19267[1]

The ad_allow_remote_domain_local_groups option has been removed from SSSD

Support for the **ad_allow_remote_domain_local_groups** option in **sssd.conf** was deprecated in Red Hat Enterprise Linux (RHEL) 9.6. The **ad_allow_remote_domain_local_groups** option has been removed in RHEL 10.

Jira:RHEL-68319[1]

The reconnection_retries option has been removed

The **reconnection_retries** option has been removed from the **sssd.conf** file in SSSD in RHEL 10.0. Because SSSD switched to a new architecture using internal IPC between SSSD processes and responders no longer connect to the backend, the **reconnection_retries** option is no longer used.

Jira:RHELDOCS-18965[1]

8.14. DESKTOP

TigerVNC has been removed

The TigerVNC remote desktop solution has been removed in RHEL 10.

TigerVNC provided the server and client implementation of the Virtual Network Computing (VNC) protocol in RHEL 9.

The following packages have been removed:

- **tigervnc**
- **tigervnc-icons**
- **tigervnc-license**
- **tigervnc-selinux**
- **tigervnc-server**
- **tigervnc-server-minimal**
- **tigervnc-server-module**

The **Connections** application (**gnome-connections**) continues to be supported as an alternative VNC client, but it does not provide a VNC server. TigerVNC is replaced by the **gnome-remote-desktop** daemon, which is a remote desktop server that uses the RDP protocol. You can use the **gnome-remote-desktop** in the following modes:

- Desktop sharing: provides sharing of your physical session by using Assisted Access
- Headless session: provides a single user remote headless session
- Remote login: provides a graphical remote login and replaces functionality of XDMCP

Jira:RHELDOCS-18388[1]

Totem media player has been removed in RHEL 10

The RHEL 10 installation does not contain any media player by default. You can use any third party media player available, for example, on Flathub.

Jira:RHELDOCS-18389[1]

power-profiles-daemon is removed in RHEL 10

The **power-profiles-daemon** package that provided power mode configuration in GNOME has been removed in RHEL 10. In RHEL 10, you can manage power profiles with the Tuned daemon.

The **tuned-ppd** package provides a drop-in replacement for **power-profiles-daemon**, which allows it to be used with GNOME desktop and applications that use **power-profiles-daemon** API. You can also use it to override the three basic power profiles, including **power-saver**, **balanced**, and **performance** through the **/etc/tuned/ppd.conf** configuration file. If you want to use a customized profile, you can edit the configuration file and map the custom profile to the three basic **power-profiles-daemon** profile names.

Jira:RHELDOCS-18390[1]

gedit is removed in RHEL 10

gedit, the default graphical text editor in Red Hat Enterprise Linux, is removed in RHEL 10. As an alternative, you can use GNOME Text Editor.

Jira:RHELDOCS-19148[1]

Tweaks is no longer available as a RHEL package in RHEL 10

Instead of the Tweaks desktop application, you can use the default GNOME Settings app, which has been expanded to include many options previously only found in Tweaks.

Jira:RHELDOCS-19125[1]

Qt5 libraries are removed in RHEL 10

Qt5 libraries are replaced with Qt6 libraries, with new functionality and better support.

For more information, see Porting to Qt 6.

Jira:RHELDOCS-19132[1]

WebKitGTK is removed in RHEL 10

The WebKitGTK web browser engine is removed in RHEL 10. As a consequence, you can no longer build applications that depend on WebKitGTK. Desktop applications other than Mozilla Firefox can no longer display web content. There is no alternative web browser engine provided in RHEL 10.

Jira:RHELDOCS-19170[1]

Evolution is removed in RHEL 10

Evolution is a GNOME application that provides integrated email, calendar, contact management, and communications functionality. The application and its plugins are removed in RHEL 10. You can find an alternative in a third party source, for example on Flathub.

You can back up your Evolution data directly in Evolution by using the **Back up Evolution data** item in the **File** menu.

Jira:RHELDOCS-19146[1]

Festival is not supported in RHEL 10

With support for the Festival speech synthesizer removed in RHEL 10, the Festival binaries, libraries and the plugin for Speech Dispatcher are also removed.

As an alternative, you can use the Espeak NG speech synthesizer.

Jira:RHELDOCS-19138[1]

The Eye of GNOME is removed

The Eye of GNOME (**eog**) image viewer application is removed in RHEL 10.

As an alternative, you can use the Loupe application.

Jira:RHELDOCS-19134[1]

Cheese is removed

The Cheese camera application is removed in RHEL 10.

As an alternative, you can use the Snapshot application.

Jira:RHELDOCS-19136[1]

Devhelp has been removed

Devhelp, a graphical developer tool for browsing and searching API documentation, has been removed in RHEL 10. You can now find API documentation online in specific upstream projects.

Jira:RHELDOCS-19153[1]

gtkmm based on GTK 3 has been removed

gtkmm is a C++ interface for the GTK graphical toolkit. The **gtkmm** version that was based on GTK 3 has been removed in RHEL 10 with all its dependencies. To access **gtkmm** in RHEL 10, migrate to the **gtkmm** version based on GTK 4.

Jira:RHELDOCS-19142[1]

LibreOffice is removed in RHEL 10

The LibreOffice RPM packages are removed from RHEL 10. LibreOffice continues to be fully supported through the entire life cycle of RHEL 7, 8, and 9.

As a replacement for the RPM packages, Red Hat recommends that you install LibreOffice from either of the following sources provided by The Document Foundation:

The official Flatpak package in the Flathub repository:
link:https://flathub.org/apps/org.libreoffice.LibreOffice. The official RPM packages:
link:https://www.libreoffice.org/download/download-libreoffice/.

Jira:RHELDOCS-19152[1]

GNOME Terminal is removed in RHEL 10

GNOME Terminal has been replaced with Ptyxis in RHEL 10.

Ptyxis is a container-oriented terminal that provides transparent support for container systems such as Podman or Toolbx and robust support for user profiles.

Jira:RHELDOCS-19155[1]

Inkscape vector graphics editor is removed in RHEL 10

The RHEL 10 installation does not contain any vector graphics editor. You can use any third party vector graphics editor available, for example, on Flathub.

Jira:RHELDOCS-19150[1]

GNOME Classic session has been removed from the default installation

If your scenario requires the GNOME classic session, install it manually:

1. Install the **gnome-classic-session** package:

```
# dnf install gnome-classic-session
```

2. Log out of your current session.
3. On the login screen (GDM), click the gear icon next to your username.
4. Select "GNOME Classic" from the session list.
5. Log in as usual.

Jira:RHEL-4137

Evince is removed in RHEL 10

Evince, a document viewer for the GNOME desktop, is removed in RHEL 10. You can use the Papers application instead. Papers is a fork of Evince ported to Gtk 4, which aims to move at a more rapid pace with adding of new features, such as listing of signatures in PDF documents. Papers is partially written in Rust for improved stability.

Jira:RHELDOCS-19140[1]

8.15. GRAPHICS INFRASTRUCTURES

The PulseAudio daemon is removed in RHEL 10

The PulseAudio daemon, and its packages **pulseaudio** and **alsa-plugins-pulseaudio**, have been removed in RHEL 10.

Note that the PulseAudio client libraries and tools are not deprecated, this change only impacts the audio daemon that runs on the system.

You can use the PipeWire audio system as a replacement, which has also been the default audio daemon since RHEL 9.0. PipeWire also provides an implementation of the PulseAudio APIs.

Jira:RHELDOCS-17682[1]

Motif is removed

Motif is an X11-based Desktop Environment (DE), which consists of a toolkit and the **mwm** X11 window manager. It was previously deprecated and has been removed from RHEL 10. As a replacement, you can use the GTK or Qt toolkit.

Jira:RHELDOCS-19221[1]

xorg-x11-server is removed from RHEL 10

The X.Org server, an implementation of the X Window System, was previously deprecated and is removed from RHEL 10. Note that the X11 protocol is not removed, which means that most applications will remain compatible through the Xwayland compositor. For more information, see Red Hat Enterprise Linux 10 plans for Wayland and Xorg server (Red Hat Blog).

Jira:RHELDOCS-19222[1]

8.16. RED HAT ENTERPRISE LINUX SYSTEM ROLES

The mssql_accept_microsoft_odbc_driver_17_for_sql_server_eula variable has been deprecated

With a future major update of RHEL, the **mssql_accept_microsoft_odbc_driver_17_for_sql_server_eula** variable will no longer be supported in the **mssql** system role because the role can now install the **odbc** driver for **mssql_tools** version 17 and 18. Therefore, you must use the **mssql_accept_microsoft_odbc_driver_for_sql_server_eula** variable without the version number instead.

Important: If you use the deprecated variable with the version number **mssql_accept_microsoft_odbc_driver_17_for_sql_server_eula**, the role notifies you to use the new variable **mssql_accept_microsoft_odbc_driver_for_sql_server_eula**. However, the deprecated variable continues to work.

Jira:RHEL-69315

8.17. VIRTUALIZATION

The virt-v2v tool can no longer convert Xen virtual machines from RHEL 5

It is no longer possible to use the **virt-v2v** tool to convert virtual machines from a RHEL 5 Xen host to KVM. For details, see the Red Hat Knowledge Base.

Jira:RHEL-37687

Red Hat Virtualization compatibility has been removed from virt-v2v

Because the maintenance support for Red Hat Virtualization (RHV) has ended, the **virt-v2v** utility no longer supports exporting virtual machines to RHV. As a consequence, the following options are no longer available in **virt-v2v**:

- **-o rhv-upload**
- **-o rhv**
- **-o vdsm**

Jira:RHEL-36712

Persistent memory device passthrough cannot be used in RHEL 10

Because the **nvml** package was removed in RHEL 10, persistent memory (**pmem**) device passthrough cannot be used anymore. **pmem** device passthrough allows a virtual machine to directly access a host's physical persistent memory hardware with minimal emulation overhead.

Jira:RHEL-23771

RDMA-based migration is unsupported

In RHEL 10, migrating virtual machines (VMs) by using Remote Direct Memory Access (RDMA) is no longer supported. Therefore, Red Hat highly discourages using the **rdma** URI for VM migration.

Jira:RHELDOCS-20094

NIC device drivers related to iPXE have been removed

The Internet Preboot eXecution Environment (iPXE) firmware provides a range of network boot options for remotely booting machines. iPXE also provides a large number of device drivers. The following iPXE drivers are no longer in use in the RHEL 10 release, and have therefore been removed:

- The complete **ipxe-roms** sub-RPM package
- Binary files containing device drivers from **ipxe-bootimgs-x86** sub-RPM package:
  - **/usr/share/ipxe/ipxe-i386.efi**
  - **/usr/share/ipxe/ipxe-x86_64.efi**
  - **/usr/share/ipxe/ipxe.dsk**
  - **/usr/share/ipxe/ipxe.iso**
  - **/usr/share/ipxe/ipxe.lkrn**
  - **/usr/share/ipxe/ipxe.usb**

Instead, iPXE now depends on the platform firmware to provide a NIC driver for the network boot. The **/usr/share/ipxe/ipxe-snponly-x86_64.efi** and **/usr/share/ipxe/undionly.kpxe** iPXE binary files are a part of the **ipxe-bootimgs** package and use the NIC driver provided by the platform firmware.

Jira:RHEL-37610

8.18. RHEL IN CLOUD ENVIRONMENTS

cloud-init no longer uses python-jsonschema

This update has removed the **cloud-init** dependency on the **python-jsonschema** package. As a consequence, it is no longer possible use the **cloud-init** schema validator to verify **cloud-init** configuration.

Jira:RHEL-65849[1]

8.19. CONTAINERS

The rsyslog container image has been removed

The **rsyslog** container image has been removed. Instead, you can use the **support-tools container** image, which includes diagnostic and troubleshooting tools such as **sos report**, **strace**, and **tcpdump**. With the **support-tools** image, you can have access to many of the functionalities previously covered by the **rsyslog** image, along with additional utilities to enhance system support and maintenance workflows.

Jira:RHELDOCS-19363[1]

The cgroupv1 has been removed

The **cgroupv1** control group mechanism has been removed, use **cgroupv2** instead. The **cgroupv2** provides a single control group hierarchy against which all resource controllers are mounted. The default in RHEL 10 is **cgroupv2**.

Jira:RHEL-67064

The runc container runtime has been removed

The **runc** container runtime has been removed. The container runtime in RHEL 10 is crun. The crun is a fast and low-memory footprint OCI container runtime written in C. The crun binary is up to 50 times smaller and up to twice as fast as the runc binary. Using crun, you can also set a minimal number of processes when running your container. The crun runtime also supports OCI hooks.

Jira:RHEL-67063

CHAPTER 9. DEPRECATED FEATURES

Deprecated functionalities are fully supported, which means that they are tested and maintained, and their support status remains unchanged within Red Hat Enterprise Linux 10. However, they will likely not be supported in a future major version release, and are not recommended for new deployments on the current or future major versions of Red Hat Enterprise Linux.

Features can be deprecated during a major version's release cycle.

A deprecated feature is listed in all future release notes until it is removed. For a complete list of deprecated features, see the release notes for the latest minor version. For information about the length of support, see Red Hat Enterprise Linux Life Cycle and Red Hat Enterprise Linux Application Streams Life Cycle.

9.1. INSTALLER AND IMAGE CREATION

The cockpit-composer package has been deprecated

The **cockpit-composer** package has been deprecated, and will be removed in future major RHEL releases. From now on, use **cockpit-image-builder**.

Jira:RHELDOCS-20167[1]

The squashfs package has been deprecated

The **squashfs** package has been deprecated, and will be removed in a future major RHEL release. As an alternative, **dracut** has support for mounting **erofs**.

Jira:RHELDOCS-18903[1]

gdisk has been deprecated from the boot.iso

gdisk has been deprecated from the **boot.iso** image type. You still can use **gdisk** in your Kickstarts. For the **boot.iso** image type, other tools are available for handling GPT disks, for example, the **parted** utility.

Jira:RHELDOCS-18904[1]

The module Kickstart command has been deprecated

Anaconda has deprecated its support for DNF modularity, and as a consequence the **module** Kickstart command has been deprecated. This might impact you if you are using modules in the **%packages** section of your Kickstart files or the **module** Kickstart command. This change is implemented for simplifying the installation process and ensuring a more consistent experience moving forward.

Jira:RHEL-34829

The inst.gpt boot option is now deprecated

The **inst.gpt** boot option is now deprecated and will be removed in the future releases. To specify a preferred disk label type, use the **inst.disklabel** boot option. Specify **gpt** or **mbr** to create GPT or MBR disk labels.

Jira:RHELDOCS-18491[1]

9.2. SECURITY

ENGINE API in OpenSSL is deprecated

In RHEL 10, ENGINE API is deprecated and is planned to be removed in a future major release. No new applications should be built by using the ENGINE API. To keep application binary interface (ABI) and existing applications working, OpenSSL still exports the ENGINE symbols. To prevent new applications from using ENGINE API, OpenSSL sets the **OPENSSL_NO_ENGINE** flag system-wide, and the header **engine.h** that exposes the ENGINE API has been removed.

Jira:RHEL-45704

crypto-policies now set allow-rsa-pkcs1-encrypt = false for GnuTLS

In RHEL 10, the GnuTLS library blocks encryption and decryption with the RSA PKCS #1 v1.5 padding by default. Except for the LEGACY policy, the **allow-rsa-pkcs1-encrypt = false** option is specified in all system-wide cryptographic policies (DEFAULT, FUTURE, and FIPS).

Jira:RHEL-64746

HMAC-SHA-1 in FIPS mode is deprecated

The HMAC-SHA-1 cryptographic algorithm is deprecated in FIPS mode, and it might be removed in a future release. Outside FIPS mode, support for HMAC-SHA-1 is preserved.

Jira:RHELDOCS-18674

9.3. NETWORKING

ipset has been unmaintained

In RHEL 10, the **ipset** utility is unmaintained and is planned to be removed in a future major release. Red Hat will provide only critical bug fixes during the current release lifecycle. As an alternative to **ipset**, you can use the **nftables** sets functionality instead.

Jira:RHELDOCS-20147[1]

9.4. FILE SYSTEMS AND STORAGE

The squashfs package has been deprecated

SquashFS is deprecated and will be removed in the next major release. It will no longer receive enhancements and is in RHEL 10 for specific use cases that are internal to Red Hat. Consider using EROFS as an alternative solution.

Jira:RHELDOCS-18450[1]

9.5. HIGH AVAILABILITY AND CLUSTERS

Deprecated High Availability Add-On features

The following features have been deprecated in Red Hat Enterprise Linux 10 and will be removed in the next major release:

- Specifying rules as multiple arguments. Use a single string argument instead.

- Specifying **score** as a standalone value in **pcs constraint location add** and **pcs constraint colocation ad**. Use **score=value** instead.
- Specifying the **--wait** option in resource commands except **pcs resource restart | move**, and in the commands **pcs cluster node add-guest | add-remote**. Use the following commands instead:
  - **pcs status wait** to wait for the cluster to settle into stable state.
  - **pcs status query resource** commands to verify that the resource is in the expected state after the wait.
- Using the **--force** flag to confirm potentially destructive actions such as **pcs cluster destroy**, **pcs quorum unblock**, **pcs stonith confirm**, **pcs stonith sbd device setup**, and **pcs stonith sbd watchdog test** commands. You should now use the **--yes** flag to confirm potentially destructive actions and reserve use of the **--force** flag to override validation errors.
- Using the **--force** flag to confirm overwriting files in **pcs cluster report**. Use the **--overwrite** flag instead.
- Assigning and unassigning ACL roles without specifying the **user** or **group** keyword.
- Configuring a score parameter in order constraints. The **pcs** command-line interface now produces a warning when a user attempts to configure a score parameter in order constraints.

Jira:RHELDOCS-19607[1]

9.6. COMPILERS AND DEVELOPMENT TOOLS

The utmp and utmpx interfaces in glibc are deprecated

The **utmp** and **utmpx** interfaces provided by the **glibc** library include a counter that counts time since the UNIX epoch. This counter will overflow on February 07, 2106. Therefore, **utmp** and **utmpx** are deprecated in RHEL 10 and will be removed in RHEL 11.

Jira:RHELDOCS-18080[1]

9.7. INFRASTRUCTURE SERVICES

FTP clients and Servers software are now deprecated

The following FTP clients and servers software are deprecated and will be removed in the future major version of RHEL:

- **ftp**
- **lftp**
- **vsftpd**

These FTP protocol implementations are no longer under active development. We recommend that customers plan to migrate workflows based on FTP to one of either:

- OpenSSH and the **sftp** command, which provides an interactive interface for secure file transfer over the SSH protocol.

- WebDAV based on Apache httpd - various client implementations are available.

Jira:RHELDOCS-20610[1]

9.8. THE WEB CONSOLE

The host switcher in the RHEL web console is deprecated

The host switcher that provides connections to multiple machines through SSH from a single RHEL web console session is deprecated and disabled by default. Due to the web technology limitations, this feature cannot be secure.

In the short term, you can enable the host switcher after assessing the risks in your scenario with the **AllowMultiHost** option in the **cockpit.conf** file:

```
[WebService]
AllowMultiHost=yes
```

As more secure alternatives, you can use:

- the web console login page (with the secure limit of one host in a web browser session)
- the Cockpit Client flatpak

Jira:RHEL-4032[1]

9.9. RED HAT ENTERPRISE LINUX SYSTEM ROLES

The sshd variable deprecated and replaced by sshd_config

To unify coding standards across the RHEL system roles, the **sshd** variable has been replaced by the **sshd_config** variable. The **sshd** variable is now deprecated and might be removed from the **sshd** Ansible role in a future major version of RHEL.

Jira:RHEL-73440[1]

9.10. VIRTUALIZATION

libslirp has been deprecated

In RHEL 10, the **libslirp** networking back end has become deprecated, and will be removed in a future major version release.

Jira:RHEL-45147

The i440fx virtual machine type has been deprecated

In RHEL 10, the **i440fx** machine types for virtual machines (VMs) have become deprecated, and will be removed in a future major version of RHEL.

In addition, the **i440fx-rhel7.6** machine type has been replaced by **i440fx-rhel10.0**. As a consequence, a VM with a **i440fx-rhel7.6** machine type will not boot correctly after live migrating to a RHEL 10 host. Workaround: Restart the VM after live migration.

Jira:RHELDOCS-18672[1]

Legacy vCPU models are now deprecated

Several virtual CPU models are now deprecated and will become unsupported for use in virtual machines (VMs) in a future major release of RHEL. Notably, the deprecated models include the following:

- Intel Xeon 55xx and 75xx Processor families (also known as Nehalem)
- Intel Xeon v2 (also known as Ivy Bridge)
- AMD Opteron G4 and G5

To view the complete list of deprecated CPU models, use the following command:

```
# /usr/libexec/qemu-kvm -cpu help | grep depre | grep -v - -v
```

To check whether a running VM is using a deprecated CPU model, use the **virsh dominfo** utility, and look for a line similar to the following in the **Messages** section:

```
tainted: use of deprecated configuration settings
deprecated configuration: CPU model 'Nehalem'
```

Jira:RHEL-28971[1]

virt-manager has been deprecated

The Virtual Machine Manager application, also known as **virt-manager**, has been deprecated. The RHEL web console, also known as **Cockpit**, is intended to become its replacement in a subsequent release. It is, therefore, recommended that you use the web console for managing virtualization in a GUI. Note, however, that some features available in **virt-manager** might not be yet available in the RHEL web console.

Jira:RHELPLAN-10304[1]

libvirtd has become deprecated

The monolithic **libvirt** daemon, **libvirtd**, has been deprecated in RHEL 9, and will be removed in a future major release of RHEL. Note that you can still use **libvirtd** for managing virtualization on your hypervisor, but Red Hat recommends switching to the newly introduced modular **libvirt** daemons. For instructions and details, see the RHEL 9 Configuring and Managing Virtualization document.

Jira:RHELPLAN-113995[1]

SecureBoot image verification using SHA1-based signatures is deprecated

Performing SecureBoot image verification using SHA1-based signatures on UEFI (PE/COFF) executables has become deprecated. Instead, Red Hat recommends using signatures based on the SHA-2 algorithm, or later.

Jira:RHELPLAN-69533[1]

The virtual floppy driver has become deprecated

The **isa-fdc** driver, which controls virtual floppy disk devices, is now deprecated, and will become unsupported in a future release of RHEL. Therefore, to ensure forward compatibility with migrated virtual machines (VMs), Red Hat discourages using floppy disk devices in VMs hosted on RHEL 10.0.

Jira:RHELPLAN-81033[1]

qcow2-v2 image format is deprecated

With RHEL 10.0, the qcow2-v2 format for virtual disk images has become deprecated, and will become unsupported in a future major release of RHEL. In addition, the RHEL 10.0 Image Builder cannot create disk images in the qcow2-v2 format.

Instead of qcow2-v2, Red Hat strongly recommends using qcow2-v3. To convert a qcow2-v2 image to a later format version, use the **qemu-img amend** command.

Jira:RHELPLAN-75969[1]

9.11. CONTAINERS

The runc container runtime has been removed

The **runc** container runtime is removed. The default container runtime is **crun**. If you upgrade from the previous RHEL versions to RHEL 10.0, you have to run the **podman system migrate --new-runtime=crun** command to set a new OCI runtime for all containers.

Jira:RHELDOCS-19051[1]

tzdata package is no longer installed by default in the minimal container images

The **tzdata** package is no longer installed in the **registry.access.redhat.com/ubi10-minimal** container image. As a consequence, if you migrate your minimal container builds from a previous RHEL release to RHEL 10.0, and you enter the **microdnf reinstall tzdata** command to reinstall the **tzdata** package, you get an error message because the **tzdata** package is no longer installed by default. In this case, enter the **microdnf install tzdata** command to install **tzdata**.

Jira:RHELDOCS-18700[1]

The Podman v5.0 deprecations

In RHEL 10.0, the following is deprecated in Podman v5.0:

- The system connections and farm information stored in the **containers.conf** file are now read-only. The system connections and farm information will now be stored in the **podman.connections.json** file, managed only by Podman. Podman continues to support the old configuration options such as **[engine.service_destinations]** and the **[farms]** section. You can still add connections or farms manually if needed; however, it is not possible to delete a connection from the **containers.conf** file with the **podman system connection rm** command.
- The **slirp4netns** network mode is deprecated and will be removed in a future major release of RHEL. The **pasta** network mode is the default network mode for rootless containers.
- The **containernetworking-plugins** package and the CNI network stack are no longer supported.
    - If you upgrade from the previous RHEL versions to RHEL 10.0 or if you have a fresh installation of RHEL 10.0, the CNI is no longer available. As a result, you have to run the

podman rmi --all --force command to remove all images and containers that are using those images.

- If present, the **cni** value in the containers.conf file for the **network_backend** option must be changed to **netavark** or can be unset.

Jira:RHEL-40641

The podman-tests package has been deprecated

The **podman-tests** package has been deprecated in the AppStream repository. The package is now available in the CodeReady Linux Builder (CRB). More information about the CRB repository can be found at https://docs.redhat.com/en/documentation/red_hat_enterprise_linux/9/html/package_manifest/repositc repository.

Jira:RHEL-67860

nodejs-18 and nodejs-18-minimal are deprecated

The **nodejs-18** and **nodejs-18-minimal** container images are now deprecated and will no longer receive feature updates. Use **nodejs-22** and **nodejs-22-minimal** instead.

Jira:RHELDOCS-20283[1]

9.12. DEPRECATED FEATURES IDENTIFIED IN PREVIOUS RELEASES

This part provides an overview of functionality that has been *deprecated* in Red Hat Enterprise Linux 10.

9.12.1. SSSD

The SMB1 protocol is deprecated in Samba

Starting with Samba 4.11, the insecure Server Message Block version 1 (SMB1) protocol is deprecated and will be removed in a future release.

To improve the security, by default, SMB1 is disabled in the Samba server and client utilities.

Jira:RHELDOCS-16612[1]

9.13. DEPRECATED PACKAGES

This section lists packages that have been deprecated and will probably not be included in a future major release of Red Hat Enterprise Linux.

IMPORTANT

The support status of deprecated packages remains unchanged within RHEL 10.

The following packages have been deprecated in RHEL 10:

- daxio
- gvisor-tap-vsock-gvforwarder

- libpmem
- libpmem2
- libpmemblk
- libpmemlog
- libpmemobj
- libpmemobj-cpp
- libpmempool
- libslirp
- nvml
- pmempool
- pmreorder
- sdl2-compat
- wget

CHAPTER 10. KNOWN ISSUES

This version of Red Hat Enterprise Linux 10.0 is affected by the following newly identified and previously known issues. A known issue is listed in all future release notes until resolved, at which point it is published as a fixed issue. If you encountered an issue that is not listed in this section, please report it by using the button in the top right corner of this page.

10.1. INSTALLER AND IMAGE CREATION

Unable to build ISOs from a signed container

Trying to build an ISO disk image from a GPG or a simple signed container results in an error, similar to the following:

```
manifest - failed
Failed
Error: cannot run osbuild: running osbuild failed: exit status 1
2024/04/23 10:56:48 error: cannot run osbuild: running osbuild failed: exit status 1
```

This happens because the system fails to get the image source signatures.

Workaround: You can either remove the signature from the container image or build a derived container image. For example, to remove the signature, you can run the following command:

```
 $ sudo skopeo copy --remove-signatures containers-storage:registry.redhat.io/rhel9/rhel-bootc:9.4
containers-storage:registry.redhat.io/rhel9/rhel-bootc:9.4
$ sudo podman run \
      --rm \
      -it \
      --privileged \
      --pull=newer \
      --security-opt label=type:unconfined_t \
      -v /var/lib/containers/storage:/var/lib/containers/storage \
      -v ~/images/iso:/output \
      quay.io/centos-bootc/bootc-image-builder \
      --type iso --local \
      registry.redhat.io/rhel9/rhel-bootc:9.4
```

To build a derived container image, and avoid adding a simple GPG signatures to it, see the Signing container images product documentation.

Jira:RHEL-34807

Hostname resolution fails with encrypted DNS and custom CA in boot options

While using the **inst.repo=** or **inst.stage2=** boot options in the kernel command line along with a remote installation URL, an encrypted DNS, and a custom CA certificate in the Kickstart file, the installation program attempts to download the **install.img** stage2 image before processing the Kickstart file. Consequently, the hostname resolution fails, leading to display of some errors before successfully fetching the stage2 image. Workaround: Define the installation source in the Kickstart file instead of the kernel command line.

Jira:RHEL-80672

The installation program becomes unresponsive during final RPM installation stage

An installation program might become unresponsive during the RPM installation process at the final stage. Before the issue occurs, you might see the repeated **Configuring rootfiles.noarch** messages. Workaround: Restart the installation process.

Jira:RHEL-67865[1]

Disabled keyboard layout switching by using shortcut during installation

To prevent confusion caused by a broken keyboard shortcut to change keyboard layout, this feature has been disabled in Anaconda. You cannot change keyboard layouts by using shortcuts during installation. Workaround: Use the keyboard layout icon on the top bar to switch layouts.

Jira:RHEL-74504

Bonding device with LACP takes longer to become operational, causing subscription failures

When configuring a bonding device with LACP by using both kernel command-line boot options and a Kickstart file, the connection is created during the **initramfs** stage but reactivated in Anaconda. As a consequence, it causes a temporary disruption that leads to system subscription failure via the **rhsm** Kickstart command.

Workaround: Add **--no-activate** to the Kickstart network configuration to keep the network operational. As a result, the system subscription completes successfully.

Jira:RHELDOCS-19853[1]

The services Kickstart command fails to disable the firewalld service

A bug in Anaconda prevents the **services --disabled=firewalld** command from disabling the **firewalld** service in Kickstart. Workaround: Use the **firewall --disabled** command instead. As a result, the **firewalld** service is disabled properly.

Jira:RHEL-83577

Installation program fails if /boot partition is not created when using ostreecontainer

When using the **ostreecontainer** Kickstart command to install a bootable container, the installation fails if the **/boot** partition is not created. This issue occurs because the installation program requires a dedicated **/boot** partition to proceed with the container deployment.

Workaround: Ensure that a **/boot** partition is defined in the Kickstart file or manually created during the installation process.

Jira:RHEL-66155

Kickstart installation fails with an unknown disk error when 'ignoredisk' command precedes 'iscsi' command

Installing RHEL by using the kickstart method fails if the **ignoredisk** command is placed before the **iscsi** command. This issue occurs because the **iscsi** command attaches the specified iSCSI device during command parsing, while the **ignoredisk** command resolves device specifications simultaneously. If the **ignoredisk** command references an iSCSI device name before it is attached by the **iscsi** command, the installation fails with an "unknown disk" error.

Workaround: Ensure that the **iscsi** command is placed before the **ignoredisk** command in the Kickstart file to reference the iSCSI disk and enable successful installation.

Jira:RHEL-58827

The USB CD-ROM drive is not available as an installation source in Anaconda

Installation fails when the USB CD-ROM drive is the source for it and the Kickstart **ignoredisk --only-use=** command is specified. In this case, Anaconda cannot find and use this source disk.

Workaround: Use the **harddrive --partition=sdX --dir=**/ command to install from USB CD-ROM drive. As a result, the installation does not fail.

Jira:RHEL-58829

Driver disk menu fails to display user inputs on the console

When you start RHEL installation by using the **inst.dd** option on the kernel command line with a driver disk, the console fails to display the user input. Consequently, it seems that the application does not respond to the user input and stops responding, but displays the output which is confusing for users. However, this behavior does not affect the functionality, and user input gets registered after pressing **Enter**.

Workaround: To see the expected results, ignore the absence of user inputs in the console and press **Enter** when you finish adding inputs.

Jira:RHEL-58828

Anaconda might not work correctly on s390x and ppc64le architectures

Image mode for RHEL supports **pp64le** and **s390x** architectures besides the already supported **x86_64** and ARM architectures. However, Anaconda might not function correctly on s390x and ppc64le architectures.

Jira:RHELDOCS-19496[1]

Anaconda installation program seems as unresponsive in the rescue mode

When booting into a rescue mode and selecting the **Continue** or **Skip to shell** options, you might experience an issue where the Anaconda installation program seems to be frozen. Despite the lack of visible response, the installation program is still functional and reacting to your inputs; however, the prompt does not display on the screen, leading to confusion.

Continue with your tasks as normal, as the installation program is still operational despite the absence of a visible prompt.

Jira:RHEL-58834[1]

10.2. SECURITY

SELinux policy rules for four libvirt services temporarily changed into permissive mode

Previously, the SELinux policy was changed to reflect the replacement of the legacy monolithic **libvirtd** daemon with a new set of modular daemons. Because this change requires testing of several scenarios, the following services have been temporarily changed into SELinux permissive mode:

- **virtqemud**
- **virtvboxd**

- **virtstoraged**
- **virtsecretd**

To prevent harmless AVC denials, **dontaudit** rules have been added to the SELinux policy for these services.

Jira:RHEL-77808[1]

Cryptographic tokens do not work in FIPS mode with pkcs11-provider

When the system runs in FIPS mode, the **pkcs11-provider** OpenSSL provider does not work correctly and the OpenSSL TLS toolkit falls back to the default provider. Consequently, OpenSSL fails to load PKCS #11 keys, and cryptographic tokens do not work in this scenario.

Workaround: Set the **pkcs11-module-assume-fips = true** parameter in the PKCS #11 section of the **openssl.cnf** file. See the **pkcs11-provider(7)** man page on your system for more information. With this configuration change, **pkcs11-provider** works in FIPS mode.

Jira:RHEL-68621

The Extended Master Secret TLS Extension is now enforced on FIPS-enabled systems

With the release of the RHSA-2023:3722 advisory, the TLS **Extended Master Secret** (EMS) extension (RFC 7627) is mandatory for TLS 1.2 connections on FIPS-enabled RHEL 9 and 10 systems. This is in accordance with FIPS-140-3 requirements. TLS 1.3 is not affected.

Legacy clients that do not support EMS or TLS 1.3 now cannot connect to FIPS servers running on RHEL 9 and 10. Similarly, RHEL 9 and 10 clients in FIPS mode cannot connect to servers that only support TLS 1.2 without EMS. This in practice means that these clients cannot connect to servers on RHEL 6, RHEL 7 and non-RHEL legacy operating systems. This is because the legacy 1.0.x versions of OpenSSL do not support EMS or TLS 1.3.

In addition, connecting from a FIPS-enabled RHEL client to a hypervisor such as VMWare ESX now fails with a **Provider routines::ems not enabled** error if the hypervisor uses TLS 1.2 without EMS. To work around this problem, update the hypervisor to support TLS 1.3 or TLS 1.2 with the EMS extension. For VMWare vSphere, this means version 8.0 or later.

For more information, see TLS Extension "Extended Master Secret" enforced with Red Hat Enterprise Linux 9.2 and later.

Jira:RHEL-13340

10.3. SHELLS AND COMMAND-LINE TOOLS

pass:uname command produces an unknown output

The uname command displays unknown output with flags **pass:--hardware-platform** and **pass:--processor**. In the previous RHEL versions, **pass:uname -i** and **pass:uname -p** were aliases for **pass:uname -m** and are not portable even across GNU/Linux distributions.

As a workaround, you can use the **pass:-m** flag instead of the **pass:-i** and **pass:-p** flags.

Jira:RHEL-74146

10.4. INFRASTRUCTURE SERVICES

Nginx does not support PKCS #11 and TPM

The OpenSSL engines API was deprecated in RHEL 9 and removed from Nginx in RHEL 10. The corresponding functionality using the current OpenSSL providers API is not yet available. As a consequence, the Nginx HTTP server does not work with hardware security modules (HSMs) through PKCS #11 and Trusted Platform Module (TPM) devices.

Jira:RHEL-33742

Using the incorrect Perl database driver for MariaDB and MySQL can lead to unexpected results

The MariaDB database is a fork of MySQL. Over time, these services developed independently and are no longer fully compatible. These differences also affect the Perl database drivers. Consequently, if you use the **DBD::mysql** driver in a Perl application to connect to a MariaDB database, or the **DBD::MariaDB** driver to connect to a MySQL database, operations can lead to unexpected results. For example, the driver can return incorrect data from read operations. To avoid such problems, use the Perl driver in your application that matches the database service.

Red Hat only supports the following scenarios:

- The Perl **DBD::MariaDB** driver with a MariaDB database
- The Perl **DBD::mysql** driver with a MySQL database

Note that RHEL 8 contained only the **DBD::mysql** driver. If you plan to upgrade to RHEL 9 and then to RHEL 10 and your application uses a MariaDB database, install the **perl-DBD-MariaDB** package after the upgrade and modify your application to use the **DBD::MariaDB** driver.

For further details, see the Red Hat Knowledgebase solution Support of MariaDB/MySQL cross-database connection from Perl db drivers.

Jira:RHELDOCS-19770[1]

VMware vCenter cannot correctly remove a SATA disk from a running RHEL VM

When using the VMware vCenter interface to remove a SATA disk from a running RHEL 10 guest on the VMware ESXi hypervisor, the disk currently does not get removed fully. It stops being functional and disappears from the guest in the vCenter interface, but the SCSI interface still detects the disk as attached in the guest.

Jira:RHEL-79913[1]

10.5. NETWORKING

The wpa_supplicant service no longer relies on the OpenSSL Engine API

In RHEL 10, engines are not compatible according to Federal Information Processing Standards (FIPS) therefore the corresponding OpenSSL Engine API has been removed. Consequently, the dependent **wpa_supplicant** service cannot load X509 certificates and keys that are stored in PKCS11 URI format. As a result, any EAP-TLS authentication method and variants using PKCS11 will not be able to connect to the relevant network anymore.

Jira:RHEL-33750

The kernel can panic if you reduce the number of SR-IOV VFs at runtime

If all of the following conditions apply, the Linux kernel can panic:

- The host has Input-Output Memory Management Unit (IOMMU) enabled.
- A network driver uses a page pool.
- You reduce the number of Single Root I/O Virtualization (SR-IOV) Virtual Functions (VFs) of the network interface that uses this driver.

Workaround: Do not reduce the number of VFs at runtime. Reboot the machine to reset the number of VFs of all interfaces to 0. Afterwards, you can set a new number of VFs because increasing the number does not cause the kernel panic.

Jira:RHEL-68401[1]

10.6. KERNEL

crashkernel boot parameter does not load in rhel-guest-image

Presently, RHEL cloud image built by **osbuild** misses the **crashkernel** kernel parameter. As a result, **kdump.service** fails to start.

Workaround: Run **kdumpctl** manually to set up the **crashkernel** kernel parameter and reboot the system. **kdump.service** will start successfully.

Jira:RHEL-63071[1]

The kdump service fails during boot

After the installation of **registry.redhat.io/rhel9/rhel-bootc** container image to a physical system, the **kdump.service** fails.

Workaround: Ensure the **PrivateTmp** service is disabled:

```
# cat /etc/systemd/system/kdump.service.d/override.conf
[Service]
PrivateTmp=no
```

Then rebuild and restart the kdump service:

```
# touch /etc/kdump.conf
# systemctl restart kdump
```

Jira:RHEL-50736

10.7. FILE SYSTEMS AND STORAGE

Reverse Mapping B+Tree (rmapbt) performance impact

By default, the XFS file system enables the **rmapbt** feature, which has potential performance regressions in write-heavy workloads with small block sizes. Evaluate performance-sensitive applications carefully, particularly those that heavily rely on writing small data blocks.

Workaround: To disable the **rmapbt** feature during file system creation, use the **-m rmapbt=0** option. This will revert the default behavior.

Jira:RHEL-33653[1]

Inconsistent NVMe device names after reboot

A new kernel feature that enables asynchronous NVMe namespace scans is introduced in RHEL 10, to accelerate NVMe disk detection. As a consequence of the asynchronous scans, the **/dev/nvmeXnY** device files might point to different namespaces after each reboot. This can lead to inconsistent device names. At this time, there is no known workaround for this issue.

Jira:RHEL-85845[1]

mpi3mr driver does not support CPU offlining correctly

Because the **mpi3mr** driver does not support CPU offlining correctly, you might experience issues such as timeout warnings in the kernel log, issues with storage disk communication, or even a system hang. There is currently no workaround for this issue. To avoid it, do not use CPU offlining.

Jira:RHEL-65655[1]

10.8. HIGH AVAILABILITY AND CLUSTERS

ACL roles should not reference location constraints with two rules

In Red Hat Enterprise Linux 10, more than one top-level rule in a location constraint is not supported. When upgrading from RHEL 9 to RHEL 10, verify that any ACL roles you have configured do not reference a location constraint with two rules and are still valid.

Jira:RHEL-62722

10.9. COMPILERS AND DEVELOPMENT TOOLS

The new version of TBB is incompatible

RHEL 10 includes the Threading Building Blocks (TBB) library version 2021.11.0, which is incompatible with the versions distributed with previous releases of RHEL. You must rebuild applications that use TBB to make them run on RHEL 10.

Jira:RHEL-33633

10.10. IDENTITY MANAGEMENT

IdM in FIPS mode does not support using the NTLMSSP protocol to establish a two-way cross-forest trust

Establishing a two-way cross-forest trust between Active Directory (AD) and Identity Management (IdM) with FIPS mode enabled fails because the New Technology LAN Manager Security Support Provider (NTLMSSP) authentication is not FIPS-compliant. IdM in FIPS mode does not accept the RC4 NTLM hash that the AD domain controller uses when attempting to authenticate.

Jira:RHEL-12154[1]

Installing a RHEL 7 IdM client with a RHEL 10 IdM server in FIPS mode fails due to EMS enforcement

The TLS **Extended Master Secret** (EMS) extension (RFC 7627) is now mandatory for TLS 1.2 connections on FIPS-enabled RHEL 10 systems. This is in accordance with FIPS-140-3 requirements. However, the **openssl** version available in RHEL 7.9 and lower does not support EMS. In consequence, installing a RHEL 7 Identity Management (IdM) client with a FIPS-enabled IdM server running on RHEL 10 fails.

Workaround: Upgrade the host to RHEL 8 or later before installing an IdM client on it.

Jira:RHELDOCS-19015[1]

DNSSEC not working correctly in RHEL IdM

The DNS Security Extensions (DNSSEC) do not function correctly in Identity Management (IdM) in RHEL 10.0 because of multiple unresolved issues stemming from the replacement of the **openssl-pkcs11** OpenSSL engine with the **pkcs11-provider** OpenSSL provider.

The changes introduced by OpenSSL have impacted the integrated DNS functionality within RHEL IdM. Specifically, the changes are affecting multiple components in IdM, including **ipa**, **bind**, **bind-dyndb-ldap**, **softhsm**, and **python-cryptography**, and how these components interact with security modules.

Jira:RHEL-30556

Automatic host keytab renewal via adcli run by SSSD is failing

In direct SSSD-AD integration, SSSD checks daily if the machine account password is older than the configured age in days and, if needed, tries to renew it. The configured age is set by the **ad_maximum_machine_account_password_age** value, with a default of **30** days. A value of **0** disables the renewal attempt.

However, currently there is an issue and the automatic renewal of the machine account password fails. If the password expires, this might result in the host losing access to the AD domain.

Workaround: Renew the password manually or via another means. Do not rely on the SSSD automatic renewal.

Jira:RHELDOCS-19172[1]

dsctl healthcheck can report a wrong database type

If you created an instance with the Lightning Memory-Mapped Database Manager (LMDB) database type, running the **dsctl healthcheck** command can result in one of the following error messages, because Directory Server checks a wrong configuration parameter:

- **DSBLE0005**. Backend configuration attributes mismatch.
- **DSBLE0006**. BDB is still used as a backend.

Workaround: Set the **NSSLAPD_DB_LIB** environment variable to **mdb** before running **dsctl healthcheck**.

Jira:RHELDOCS-19014[1]

An error message is displayed during migration from BDB to LMDB

When you run the **dsctl dblib bdb2mdb** command to migrate from Berkeley Database (BDB) to Lightning Memory-Mapped Database Manager (LMDB) and you have not enabled the replication, the following error message is displayed in the output:

```
Error: 97 - 1 - 53 - Server is unwilling to perform - [] - Unauthenticated binds are not allowed
```

Note that you can ignore the error message. The error occurs because Directory Server attempts to find the **replication_changelog.db** file that is not mandatory when the replication is disabled. This error does not prevent the migration from BDB to LMDB.

There is currently no workaround for this issue.

Jira:RHELDOCS-19016[1]

ldapmodify does not delete a single specific value from any attribute in cn=config

Currently, when you try to delete a value from any attribute in **cn=config**, the value remains in the attribute and the server might require a restart to fully remove it.

Workaround: Remove the entire attribute, including all its values, by performing a modify operation without specifying any values. Then re-add the values you need. Alternatively, use the following **dsconf** command to remove a specific value without a server restart:

```
# dsconf <instance_name> config delete <attribute_name>=<undesired_value>
```

Jira:RHEL-25071

10.11. SSSD

SSSD retrieves incomplete list of members if the group size exceeds 1500 members

During the integration of SSSD with Active Directory, SSSD retrieves incomplete group member lists when the group size exceeds 1500 members. This issue occurs because Active Directory's MaxValRange policy, which restricts the number of members retrievable in a single query, is set to 1500 by default.

Workaround: Change the MaxValRange setting in Active Directory to accommodate larger group sizes.

Jira:RHELDOCS-19603[1]

10.12. DESKTOP

Standard mouse cursor is offset in VMs when using Mutter

When you use a standard mouse within a virtual machine (VM) configuration in the Mutter compositing window manager, you might notice an offset between the physical mouse cursor and the actual pointer within the virtual environment. The actual pointer might not even be visible in the virtual environment.

Workaround: If your scenario requires precise input, use a tablet as an input device in the VM configuration.

Jira:RHEL-69291

10.13. GRAPHICS INFRASTRUCTURES

Standard mouse cursor is offset in VMs when using Mutter

When you use a standard mouse within a virtual machine (VM) configuration in the Mutter compositing window manager, you might notice an offset between the physical mouse cursor and the actual pointer within the virtual environment. The actual pointer might not even be visible in the virtual environment.

Workaround: If your scenario requires precise input, use a tablet as an input device in the VM configuration.

Jira:RHEL-45898

10.14. THE WEB CONSOLE

VNC console in the RHEL web console does not work correctly on ARM64

Currently, when you import a virtual machine (VM) in the RHEL web console on ARM64 architecture and then you try to interact with it in the VNC console, the console does not react to your input.

Additionally, when you create a VM in the web console on ARM64 architecture, the VNC console does not display the last lines of your input.

Jira:RHEL-31993[1]

10.15. RED HAT ENTERPRISE LINUX SYSTEM ROLES

ansible-core does not install **sshpass** as a dependency

The **ansible-core** package does not install the **sshpass** package as a dependency. Consequently, you cannot use Ansible to manage systems over SSH with an SSH password.

Workaround: On the control node, manually install **sshpass** after you install **ansible-core**. As a result, you can use Ansible in the scenario described above.

Jira:RHEL-86829[1]

10.16. VIRTUALIZATION

Installing the VirtIO-Win bundle cannot be canceled

Currently, if you start the installation of **virtio-win** drivers from the VirtIO-Win installer bundle in a Windows guest operating system, clicking the **Cancel** button during the installation does not correctly stop it. The installer wizard interface displays a "Setup Failed" screen, but the drivers are installed and the IP address of the guest is reset.

Jira:RHEL-53962, Jira:RHEL-53965

Secure Execution VMs cannot boot with file-backed memory backing

If you configure a virtual machines (VMs) with enabled Secure Execution to use file-backed memory backing, the VM currently fails to boot, and instead displays a **Protected boot has failed** error.

Workaround: Edit the **/etc/libvirt/qemu.conf** file and set the **memory_backing_dir** line to the following value:

```
memory_backing_dir = "/dev/shm/"
```

Afterwards, the affected VMs can boot as expected.

Jira:RHEL-58218

VMs sending discard I/O requests might pause when discard_granularity is not configured

The host kernel fails misaligned discard I/O requests and QEMU uses the **werror=** ***policy*** parameter to respond to such failures. When **werror** is set to **stop**: **werror=stop**, a failed discard request causes the virtual machine (VM) to pause. This is usually undesirable because there is no way to correct this situation and resume the VM again.

Workaround: Ensure that the **discard_granularity** parameter on **virtio-blk** and **virtio-scsi** disks is set and matches the host's /**sys**/**block**/***<blkdev>***/**queue**/**discard_granularity** value. This makes the VM aware of the alignment constraints and ensures discard requests will be properly aligned, so they do not fail.

Jira:RHEL-87642[1]

The --migrate-disks-detect-zeroes option might not work for VM migration

Currently, when migrating virtual machines (VMs) on RHEL 10, the **--migrate-disks-detect-zeroes** option might not work and the migration might proceed without zeroed block detection on the specified disk. This problem is caused by a bug in QEMU where mirroring jobs had been relying on punching holes, which results in a sparse destination file.

Jira:RHEL-88435

A virtual machine with a large amount of bootable data disks might fail to start

If you attempt to start a virtual machine (VM) with a large amount of bootable data disks, the VM might fail to boot with this error: **Something has gone seriously wrong: import_mok_state() failed: Volume Full**

Workaround: Decrease the number of bootable data disks and use one system disk. To ensure the system disk is first in the boot order, add **boot order=1** to the device definition of the system disk in the XML configuration. For example:

```
<disk type='file' device='disk'>
  <driver name='qemu' type='qcow2'/>
  <source file='/path/to/disk.qcow2'/>
  <target dev='vda' bus='virtio'/>
  <boot order='1'/>
</disk>
```

Set boot order only for the system disk.

Jira:RHEL-68418

Too many open files in a virtiofs shared directory can crash the vrtiofsd process

When accessing a **virtiofs** shared directory with a large amount of open files from a virtual machine (VM), the operation might fail with the following error: **Too many open files** and the **virtiofsd** process might crash.

Workaround: Try any of the following steps:

- Run **virtiofsd** as root and use the **--inode-file-handles=mandatory** command-line option.

- Use the **--cache=never** command-line option.
- Increase the number of file descriptors **virtiofsd** is permitted to use with the **--rlimit-nofile** command-line option.

Jira:RHEL-87161[1]

VMs with large memory cannot boot on SEV-SNP host with AMD Genoa CPUs

Currently, virtual machines (VMs) cannot boot on hosts that use a 4th Generation AMD EPYC processor (also known as Genoa) and have the AMD Secure Encrypted Virtualization with Secure Nested Paging (SEV-SNP) feature enabled. Instead of booting, a kernel panic occurs in the VM.

Jira:RHEL-32892[1]

The virtio balloon driver sometimes does not work on Windows 10 and Windows 11 VMs

Under certain circumstances, the **virtio-balloon** driver does not work correctly on virtual machines (VMs) that use a Windows 10 or Windows 11 guest operating system. As a consequence, such VMs might not use their assigned memory efficiently.

Jira:RHEL-12118

Windows 11 VMs with a memory balloon device set might close unexpectedly during reboot

Currently, rebooting virtual machines (VMs) that use a Windows 11 guest operating system and a memory balloon device in some cases fails with a **DRIVER POWER STAT FAILURE** blue-screen error.

Jira:RHEL-935[1]

Windows VM with VBS and IOMMU device fails to boot

When you boot a Windows VM with Virtualization Based Security (VBS) enabled and an Input-Output Memory Management Unit (IOMMU) device by using the **qemu-kvm** utility, the booting sequence only shows the boot screen, resulting in an incomplete booting process.

Workaround: Ensure the VM domain XML is configured as below:

```
<features>
  <ioapic driver='qemu'/>
</features>
<devices>
<iommu model='intel'>
   <driver intremap='on' eim='off' aw_bits='48'/>
   <alias name='iommu0'/>
</iommu>
<memballoon model='virtio'>
   <alias name='balloon0'/>
   <address type='pci' domain='0x0000' bus='0x03' slot='0x00' function='0x0'/>
   <driver iommu='on' ats='on'/>
</memballoon>
</devices>
```

Otherwise, the Windows VM cannot boot.

Jira:RHEL-45585[1]

Windows VM running on Sapphire Rapids CPU with hypervisor launch type set to auto might fail to boot when restarted

If you set the hypervisor launch type to **auto** in a Windows virtual machine (VM) running on a Sapphire Rapids CPU, the VM might fail to boot when it is restarted. For example, you can set the hypervisor launch type to **auto** by using the **bcdedit /set hypervisorlaunchtype Auto** command.

Workaround: Do not set the hypervisor launch type to **auto** in the Windows VM.

Jira:RHEL-67699

Hot-plugging vCPUs and memory to Windows guests with VBS does not work

Currently, Windows Virtualization-based Security (VBS) is not compatible with hot-plugging CPU and memory resources. As a consequence, attempting to attach memory or vCPUs to a running Windows virtual machine (VM) with VBS enabled only adds the resources to the VM after the guest system is restarted.

Jira:RHEL-66229, Jira:RHELDOCS-19066

Migrating virtual machines on IBM Z sometimes removes network configuration

Currently, after migrating a virtual machine (VM) between IBM Z hosts, the VM's network configuration is in some cases reset, which causes the network to become unavailable on the VM. To work around the issue, disable the **vhost-net** service before you start the VM migration.

Jira:RHEL-42486[1]

10.17. RHEL IN CLOUD ENVIRONMENTS

RDMA devices currently do not work on vSphere

When using a RHEL 10 instance on the VMware vSphere platform, the **vmw_pvrdma** module currently does not install properly. As a consequence, VMware paravirtual remote direct memory access (PVRDMA) devices do not work on the affected instances.

Jira:RHEL-41133[1]

The leapp upgrade fails when upgrading from RHEL 9.6 to RHEL 10.0 for the cloud-init network configuration

If you deploy RHEL 9.6 with the **cloud-init** default configuration and with **sysconfig** as the default network configuration directory, the **sysconfig** configuration files do not support the **ifcfg** legacy format for RHEL 10.0. Consequently, the **leapp** upgrade fails when upgrading from RHEL 9.6 to RHEL 10.0 for the legacy network configuration files, such as ifcfg-*<enp1s0>*.

Workaround: Convert the **sysconfig** configuration files into the NetworkManager native **keyfile** format:

1. Modify the connection:

```
# nmcli connection modify "System <enp1s0>" connection.id "cloud-init <enp1s0>"
```

2. Migrate the connection:

```
# nmcli connection migrate /etc/sysconfig/network-scripts/ifcfg-<enp1s0>
```

3. Move the connection profile:

```
# sudo mv /etc/NetworkManager/system-connections/"cloud-init <enp1s0>.nmconnection"
/etc/NetworkManager/system-connections/cloud-init-<enp1s0>.nmconnection
```

4. Reload the network connection settings:

```
# nmcli conn reload
```

As a result, the leapp upgrade from RHEL 9.6 to RHEL 10.0 now works with the updated configuration.

Jira:RHEL-82209[1]

Upgrading a RHEL 9.6 guest on VMware ESXi to RHEL 10.0 causes **cloud-init** to rewrite the network configuration

After a upgrading a RHEL guest on the VMware ESXi hypervisor from RHEL 9.6 to RHEL 10.0, the **cloud-init** tool currently cannot detect the VMware data source and cannot restore its configuration from the cache. As a consequence, **cloud-init** reverts to the **None** data source, and rewrites the network configuration of the guest.

Workaround: Remove the **disable_vmware_customization** flag from the **/etc/cloud/cloud.cfg** file before you reboot the guest during the upgrade process. As a result, the upgraded guest will retain its previous network configuration.

Jira:RHEL-82210[1]

Nested VM with KVM virtualization and OVMF fails to boot on Azure or Hyper-V when using AMD EPYC processor

A nested VM with Open Virtual Machine Firmware (OVMF) fails to boot when run on a RHEL VM with KVM virtualization enabled in the Azure cloud or Hyper-V using the AMD EPYC processor. The VM fails to boot up with following log message:

```
Code=qemu-kvm: ../hw/core/cpu-sysemu.c:76  Aborted (core dumped) .
```

Workaround: Try booting without using the AMD EPYC processor.

Jira:RHEL-29919[1]

BIOS or UEFI supported Hyper-V Windows Server 2016 VM fails to boot if a host uses the AMD EPYC CPU processor

With the Hyper-V enabled setting, Hyper-V Windows Server 2016 VM fails to boot on the AMD EPYC CPU host.

Workaround: Check for the following log message:

```
kvm: Booting SMP Windows KVM VM with !XSAVES && XSAVEC.
If it fails to boot try disabling XSAVEC in the VM config.
```

And try adding **xsavec=off** to **-cpu cmdline** to boot Hyper-V Windows Server 2016 VM.

Jira:RHEL-38957[1]

10.18. CONTAINERS

Podman and bootc do not share the same registry login process

Podman and **bootc** use different registry login processes when pulling images. As a consequence, if you login to an image by using Podman, logging to a registry for **bootc** will not work on that image. When you install an image mode for RHEL system, and login to registry.redhat.io by using the following command:

```
# podman login registry.redhat.io <username_password>
```

And then you attempt to switch to the **registry.redhat.io/rhel9/rhel-bootc** image with the following command:

```
# bootc switch registry.redhat.io/rhel9/rhel-bootc:9.4
```

You should be able to see the following message:

```
Queued for next boot: registry.redhat.io/rhel9/rhel-bootc:9.4
```

However, an error is displayed:

```
ERROR Switching: Pulling: Creating importer: Failed to invoke skopeo proxy method OpenImage: remote error: unable to retrieve auth token: invalid username/password: unauthorized: Please login to the Red Hat Registry using your Customer Portal credentials. Further instructions can be found here: https://access.redhat.com/RegistryAuthentication
```

Workaround: Follow the steps Configuring container pull secrets to use authenticated registries with **bootc**.

Jira:RHELDOCS-18471[1]

cloud-init growpart skips with composefs is enabled

When composefs is enabled, if you generate an image from the generic base image, then the rootfs will note grow the filesystem, prompting an error similar to:

```
2024-04-30 17:27:53,543 - cc_growpart.py[DEBUG]: '/' SKIPPED: stat of 'overlay' failed: [Errno 2] No such file or directory: 'overlay'
```

Workaround: You can add a custom growpart, by specifying the **rootfs** default size in the container, instead of dynamically choosing 100G at instance creation time to be able to write a partitioning config in the container.

Jira:RHEL-34859

FIPS bootc image creation fails on FIPS enabled host

Building a disk image on a host by using Podman with enabled the FIPS mode fails with the exit code 3 because of the update-crypto-policies package:

```
# Enable the FIPS crypto policy
# crypto-policies-scripts is not installed by default in RHEL-10
RUN dnf install -y crypto-policies-scripts && update-crypto-policies --no-reload --set FIPS
```

Workaround: Build the bootc image with FIPS mode disabled.

Jira:RHELDOCS-19539

Insufficient disk space can cause deployment failure

Deploying a bootc container image on a package mode system without enough free disk space can result in installation errors and prevent the system from booting. Ensure adequate disk space is available for the image to install and adjust the provision logical volume before deployment.

Jira:RHELDOCS-19948[1]

RHEL images on Azure marked as LVM require default layout resizing

When using **system-reinstall-bootc** or **bootc install** on Azure, RHEL images marked as LVM will require resizing the default layout.

Workaround: Use RHEL images labeled as RAW. This does not require resizing the default layout.

Jira:RHELDOCS-19945[1]

10.19. LIGHTSPEED

Configuration file changes are not applied immediately

When making changes in the **etc/xdg/command-line-assistant/config.toml** configuration file, it takes around 30 to 60 seconds for the command line assistant daemon to recognize the changes, instead of applying the changes immediately. The command line assistant is also missing the **reload** functionality.

Workaround: Follow the steps:

1. Make the changes that you need to the **config.toml** configuration file.
2. Run the following command:

```
# systemctl restart clad
```

Jira:RHELDOCS-19734[1]

10.20. KNOWN ISSUES IDENTIFIED IN PREVIOUS RELEASES

This part describes known issues in Red Hat Enterprise Linux 10.0.

10.20.1. Networking

Failure to update the session key causes the connection to break

Kernel Transport Layer Security (kTLS) protocol does not support updating the session key, which is used by the symmetric cipher. Consequently, the user cannot update the key, which causes a connection break.

Workaround: Disable kTLS. As a result, with the workaround, it is possible to successfully update the session key.

Jira:RHELPLAN-99859[1]

kTLS does not support offloading of TLS 1.3 to NICs

Kernel Transport Layer Security (kTLS) does not support offloading of TLS 1.3 to NICs. Consequently, software encryption is used with TLS 1.3 even when the NICs support TLS offload.

Workaround: Disable TLS 1.3 if offload is required. As a result, you can offload only TLS 1.2. When TLS 1.3 is in use, there is lower performance, since TLS 1.3 cannot be offloaded.

Jira:RHELPLAN-96004[1]

CHAPTER 11. FIXED ISSUES

This version provides the following fixed issues and other problems that have a significant impact.

11.1. INSTALLER AND IMAGE CREATION

Improved installation program stability during virtual network devices configuration

Previously, the installation program could crash when creating a VLAN network device over an existing virtual network device (for example, Team or Bond) in the GUI. This occurred when the underlying device's state changed during the configuration update to the user interface for the new device state.

With this update, the process of refreshing the state of networking in GUI optimized to handle changes in the virtual device state. As a result, the installation program no longer crashes due to changes regarding virtual network devices configured in GUI.

Jira:RHEL-56141

11.2. SECURITY

IPsec ondemand connections no longer fail to establish

Previously, when an IPsec connection with the **ondemand** option was configured by using the TCP protocol, the connection failed to establish. With this update, the new Libreswan package makes sure that the initial IKE negotiation completes over TCP. As a result, Libreswan successfully establishes the connection even in TCP mode of IKE negotiation.

Jira:RHEL-51880[1]

NSS now enforce EMS in FIPS mode

The Network Security Services (NSS) libraries now contain the **TLS-REQUIRE-EMS** keyword to require the Extended Master Secret (EMS) extension (RFC 7627) for all TLS 1.2 connections as mandated by the FIPS 140-3 standard. NSS use the new keyword when the system-wide cryptographic policies are set to **FIPS**.

If your scenario requires interoperating with legacy systems without support for EMS or TLS 1.3, you can apply the **NO-ENFORCE-EMS** system-wide cryptographic subpolicy. However, this change violates the FIPS-140-3 requirements.

Jira:RHEL-36299

shlibsign now works in FIPS mode

Before this update, the **shlibsign** program did not work in FIPS mode. Consequently, when you rebuilt an NSS library in FIPS mode, you had to leave FIPS mode to sign the library. The program has been fixed, and you can now use **shlibsign** in FIPS mode.

Jira:RHEL-61291[1]

OpenSSL cipher suites no longer enable cipher suites with disabled hashes or MACs

Previously, applying custom cryptographic policies could leave certain TLS 1.3 cipher suites enabled even if their hashes or MACs were disabled, because the OpenSSL TLS 1.3-specific **Ciphersuites** option values were controlled only by the **ciphers** option of the cryptographic policy. With this update,

crypto-policies takes more algorithms into account when deciding whether to enable a cipher suite. As a result, OpenSSL on systems with custom cryptographic policies might refuse to negotiate some of the previously enabled TLS 1.3 cipher suites in better accordance with the system configuration.

Jira:RHEL-76526

update-ca-trust extract no longer fails to extract certificates with long names

When extracting certificates from the truststore, the **trust** tool internally derives the file name from the certificates' object label. For long enough labels, the resulting path might previously have exceeded the system's maximum file name length. As a consequence, the **trust** tool failed to create a file with a name that exceeded the maximum file name length of a system. With this update, the derived name is always truncated to within 255 characters. As a result, file creation does not fail when the object label of a certificate is too long.

Jira:RHEL-64915[1]

Binary tests for libcap are waived

The **annocheck** tool discovered binary packages in the **libcap** library function that were built without the required flags for RHEL 10 architectures. We examined the flags for potential problems and did not find any. After careful investigation, we have waived the results for **libcap**. As a result, all tests for **libcap** passed.

Jira:RHEL-33498[1]

11.3. SHELLS AND COMMAND-LINE TOOLS

ReaR now interprets square brackets enclosing IPv6 addresses in URLs as expected

Previously, square brackets in **OUTPUT_URL** and **BACKUP_URL** were not interpreted correctly. Specifying an IPv6 address instead of a hostname requires enclosing the address in square brackets, for example, [::1] for localhost. Since the brackets were not interpreted correctly, using an IPv6 address in a **sshfs:**// or **nfs:**// URL was not possible.

As a consequence, if the user used a **sshfs:**// or **nfs:**// scheme in the **BACKUP_URL** or **OUTPUT_URL** with an IPv6 address enclosed in square brackets, ReaR aborted prematurely with an error message, for example:

```
ERROR: Invalid scheme '' in BACKUP_URL
```

With this update, ReaR is now fixed to not interpret square brackets as shell metacharacters when parsing **sshfs:**// and **nfs:**// URLs. Now, you can use IPv6 addresses enclosed in brackets in **BACKUP_URL** and **OUTPUT_URL** that use the **sshfs:**// or **nfs:**// scheme . For example:

```
OUTPUT_URL=nfs://[2001:db8:ca2:6::101]/root/REAR
```

Before this fix was implemented, it was possible to work around the bug by using quoting and backslash characters, for example:

```
OUTPUT_URL="nfs://\[2001:db8:ca2:6::101\]/root/REAR"
```

Note: If you have been using the workaround, remove the backslash characters after applying the update.

Jira:RHEL-46613[1]

11.4. INFRASTRUCTURE SERVICES

cups-filters project is now split into several projects

The **cups-filters** project is split into several projects . The notable packages are mentioned below :

- **libcupsfilters**: replacement for cups-filters-libs RPM.
- **libppd** PPD library for retrofitting PPD support is added as a new component.
- **cups-browsed**: the daemon which was previously shipped in cups-filters.
- **cups-filters**: filters needed for various printing.
- **cups-filters-driverless**: ships driver less utilities, split from cups-filters to prevent additional dependencies for customers, who do not want to use the driver less utilities.

The customers who have disabled weak dependencies will not receive the **cups-browsed** and **cups-filters-driverless** packages, as they are weak dependencies of CUPS in RHEL 10. The **cups-browsed** package is part of the Server comps data and is installed by default in Server variants.

Jira:RHELDOCS-17679[1]

11.5. NETWORKING

NetworkManager can mitigate the impact of CVE-2024-3661 (TunnelVision) in VPN connection profiles

VPN connections rely on routes to redirect traffic through a tunnel. However, if a DHCP server uses the classless static route option (121) to add routes to a client's routing table, and the routes propagated by the DHCP server overlap with the VPN, traffic can be transmitted through the physical interface instead of the VPN. CVE-2024-3661 describes this vulnerability, which is also know as TunnelVision. As a consequence, an attacker can access traffic that the user expects to be protected by the VPN.

On RHEL, this problem affects LibreSwan IPSec and WireGuard VPN connections. Only LibreSwan IPSec connections with profiles in which both the **ipsec-interface** and **vt-interface** properties are undefined or set to **no** are not affected.

The CVE-2024-3661 document describes steps to mitigate the impact of TunnelVision by configuring VPN connection profiles to place the VPN routes in a dedicated routing table with a high priority. The steps work for both LibreSwan IPSec and WireGuard connections.

Jira:RHEL-64719[1]

RHEL 10 provides libnftnl version 1.2.8

The **libnftnl** library version 1.2.8 provides a few bug fixes. Notable changes include:

- Fixes incorrect validation of the **dynset** Netlink attribute from the kernel.
- No longer appends a newline when printing a rule.

Jira:RHEL-66276

11.6. BOOT LOADER

The GRUB2 net_del_dns command deletes the DNS server correctly

Previously, if you attempted to delete the DNS server by using the **net_del_dns** command, it added the DNS server back erroneously because of incorrect implementation, and returned an error. With this fix, the **add** command was replaced by the **remove** command in the **net_del_dns** implementation. As a result, you can delete the DNS server by using the **net_del_dns** command.

Jira:RHEL-4378

11.7. FILE SYSTEMS AND STORAGE

The Kickstart file now correctly sets the required device size for installation when using LVM partitioning with LUKS

Before this update, when you specified the **--size=1 --grow --encrypted** option in the Kickstart file for a new device, the installation program failed to correctly expand the encrypted device to a valid size. Consequently, the automated installation stopped with an error message, for example:

> "Kickstart insufficient" "('device cannot be smaller than 16 MiB', 'luks5'

You would then have to proceed with manual installation without the Kickstart file.

With this update, the installation starts successfully with the device specified in the Kickstart file with **--size=1 --grow --encrypted**. As a result, the installation proceeds without errors.

Jira:RHEL-45180

multipathd no longer crashes because of errors encountered by the ontap prioritizer

Before this update, **multipathd** crashed when it was configured to use the ontap prioritizer on an unsupported path, because the prioritizer only works with NetApp storage arrays. This failure occurred due to a bug in the prioritizer's error logging code, which caused it to overflow the error message buffer. With this update, the error logging code has been fixed, and **multipathd** no longer crashes because of errors encountered by the ontap prioritizer.

Jira:RHEL-49747[1]

Native NVMe multipathing no longer causes a memory leak when enable_foreign is set to monitor natively multipathed NVMe devices

Before this update, enabling native NVMe multipathing caused a memory leak if the **enable_foreign** configuration parameter was set to monitor natively multipathed NVMe devices. With this update, the memory leak was fixed in **multipathd** monitoring code. As a result, **multipathd** can now monitor natively multipathed NVMe devices without increasing memory usage.

Jira:RHEL-73410[1]

RHEL installation program now discovers and uses iSCSI devices as boot devices on aarch64

Previously, the absence of the **iscsi_ibft** kernel module in RHEL installation programs running on **aarch64** prevented the automatic discovery of iSCSI devices defined in firmware. As a result, these devices were not automatically visible nor selectable as boot devices in the installation program during

manual addition GUI.

This issue has been resolved by including the **iscsi_ibft** kernel module in newer **aarch64** builds of RHEL. As a result, the iSCSI devices are now automatically detected and available as boot options during installation.

Jira:RHEL-75491[1]

fstrim enabled by default on LUKS2 root in ostree-based new installations done by Anaconda

Previously, installing ostree-based systems, such as Image Mode, by using **ostreesetup** or **ostreecontainer** Kickstart commands with LUKS2 encryption enabled on the / (root) mount point resulted in systems where **fstrim** was not enabled. This could cause issues such as unresponsive systems or broken file chooser dialogs. With this fix, **fstrim** (discards) is now enabled by default in the LUKS2 metadata on newly installed systems.

To fix this issue in the existing installations, run the following command: .... **cryptsetup --allow-discards --persistent refresh <luks device>** .... **<luks device>** is the path to the root LUKS2 device.

Jira:RHEL-82884

11.8. HIGH AVAILABILITY AND CLUSTERS

pcs validation of SBD options

Previously, when you enabled SBD with the **pcs stonith sbd enable** command and specified values for SBD options that are not valid, it resulted in SBD misconfiguration. The **pcs** command-line interface has been updated to validate the values for SBD options. When the values are not valid, **pcs** reports the error and does not create or update an SBD configuration.

Jira:RHEL-38484[1]

Ability to remove Booth configuration from a Booth arbitrator node

Previously, running the **pcs booth destroy** command to remove Booth configuration from a Booth arbitrator node yielded an error. This happened because the command did not remove Booth configuration from nodes that are not part of the cluster. It is now possible to remove Booth configuration from Booth arbitrators.

Jira:RHEL-38486[1]

pcsd processes now consistently stop correctly and promptly

Previously, the creation method for **pcsd** processes sometimes caused a deadlock during process termination. The processes were then terminated only after a **systemd** timeout. This fix changes the process creation method and there is no longer a deadlock when the processes are stopped. As a result, **pcsd** consistently stops correctly within a short time.

Jira:RHEL-38478[1]

pcs no longer validates fencing topology with fencing levels greater than 9

The Pacemaker cluster resource manager ignores fencing topology levels greater than 9. Configuring levels greater than 9 might lead to failed fencing. With this update, you can configure fencing levels with values of only 1 to 9 in the **pcs** command-line interface and fencing topology works correctly.

Jira:RHEL-38479[1]

The syntax for specifying a score value is now consistent across all pcs constraint commands

Previously, some commands for creating constraints required you to specify a score value as **score=*value***, whereas others expected just ***value*** without **score=**. With this update, all constraint commands accept a score value in the form **score=*value***, with the exception of **pcs constraint location prefers** and **pcs constraint location avoids**, which expect ***node=score*** where ***score*** is the score value.

Jira:RHEL-34792[1]

The CIB manager no longer increases in size indefinitely with each request from an asynchronous client

Previously, when the CIB manager received a request from an asynchronous client, it leaked a small amount of memory. This caused the CIB manager process gradually to grow in size. With this fix, the relevant memory is freed for asynchronous clients and the CIB manager process does not grow in size indefinitely.

Jira:RHEL-40117

Resource constraints with expired rules no longer display

Before this update, the **pcs constraint location config resources** command displayed resource constraints with expired rules in the output. With this update, the command no longer displays constraints with expired rules if you do not specify the **--all** option.

Jira:RHEL-33386

Cluster status of a disaster recovery site now displays correctly

Before this update, when you configured a disaster recovery site and ran the **pcs dr status** command to display the status of the local and remote cluster sites, the command displayed an error instead of the cluster status. With this update, the cluster status of the local and remote sites displays correctly when you run this command.

Jira:RHEL-61747

Status of a cloned resource running with only one instance now displays properly

Before this update, when you queried the status of the instances of a cluster resource clone with only one running instance, the **pcs status query** command displayed an error message. With this update, the command reports the resource status properly.

Jira:RHEL-55723

11.9. COMPILERS AND DEVELOPMENT TOOLS

Go applications no longer panic if OpenSSL is not installed

Previously, if the OpenSSL library was not installed, applications created with Go panicked even if the Federal Information Processing Standard (FIPS) mode was disabled. This update solves this problem. As a result, you can now run applications created with Go if OpenSSL is not installed.

Jira:RHEL-52486[1]

Go now uses ld.bfd as the default linker on the 64-bit ARM platform

In previous RHEL versions, Go used the **ld.gold** linker only on 64-bit ARM platforms and **ld.bfd** on other platforms. Because **ld.gold** is deprecated in the **binutils** project, Go now also uses **ld.bfd** on 64-bit ARM platforms.

Jira:RHEL-49036

11.10. IDENTITY MANAGEMENT

The ipa idrange-add command now warns that Directory Server must be restarted on all IdM servers

Previously, the **ipa idrange-add** command did not warn the administrator that they must restart the Directory Server (DS) service on all IdM servers after creating a new range. As a consequence, the administrator sometimes created a new user or group with a UID or GID belonging to the new range without restarting the DS service. The addition resulted in the new user or group not having an SID assigned. With this update, a warning that DS needs to be restarted on all IdM servers is added to the command output.

Jira:RHELDOCS-18201[1]

The ipa-replica-manage command no longer resets the nsslapd-ignore-time-skew setting during forced replication

Previously, the **ipa-replica-manage force-sync** command reset the **nsslapd-ignore-time-skew** setting to **off**, regardless of the configured value. With this update, the **nsslapd-ignore-time-skew** setting is no longer overwritten during forced replication.

Jira:RHEL-4879

certmonger now correctly renews KDC certificates on hidden replicas

Previously, when the certificate was about to expire, **certmonger** failed to renew the KDC certificate on hidden replicas. This happened because the renewal process only considered non-hidden replicas as active KDCs. With this update, the hidden replicas are treated as active KDCs, and **certmonger** renews the KDC certificate successfully on these servers.

Jira:RHEL-46607[1]

Bypassing two-factor authentication by using an expired token is no longer possible

Previously, it was possible to bypass two-factor authentication by creating an OTP token with a specific end-validity period.

In cases where two-factor authentication is enforced, a user without an OTP token could use their password to log in **once** and configure an OTP token. Subsequently, they would be required to use both their password and the OTP token for authentication. However, if a user created an OTP token with an

expired end-validity date, IdM would incorrectly fall back to password-only authentication, effectively bypassing two-factor authentication. This was due to IdM not differentiating between non-existent and expired OTP tokens.

With this update, IdM now correctly differentiates between these scenarios. Consequently, two-factor authentication is now correctly enforced, preventing this bypass.

Jira:RHEL-63325[1]

The Account Policy plugin now uses a proper flag for an update in a replication topology

Before this update, the Account Policy plugin did not use the proper flag for an update. As a result, in a replication topology, the Account Policy plugin updated the login history, but this update failed on a consumer server logging the following error message:

```
{{ERR - acct_update_login_history - Modify error 10 on entry
}}
```

With this update, the internal update succeeds and no errors are logged.

Jira:RHEL-74164

TLS 1.3 can now be used to connect to an LDAP server running in FIPS mode

Before this update, when you tried to explicitly set TLS 1.3 when connecting to an LDAP server in FIPS mode, the used TLS version still remained 1.2. As a result, an attempt to connect to the LDAP server by using TLS 1.3 failed. With this update, the upper limit of the TLS version in FIPS mode was changed to 1.3, and the attempt to connect to an LDAP server with TLS 1.3 no longer fails.

Jira:RHEL-79498[1]

A race condition with paged result searches no longer closes the connection with a T3 error code

Before this update, Directory Server did not use the proper thread protection when checking the connection's paged result data for a timeout event. As a consequence, the paged result timeout value changed unexpectedly and triggered a false timeout when a new operation arrived. This caused a time out error and the connection was closed with the following **T3** error code:

The server closed the connection because the specified time limit for a paged result search has been exceeded.

With this update, the proper thread protection is used, and paged result searches no longer close the connection with a **T3** error code.

Jira:RHEL-76020[1]

ldapsearch now respects the NETWORK_TIMEOUT setting as expected

Before this update, an **ldapsearch** command ignored the timeout when the server was unreachable and, as a consequence, the search hung indefinitely instead of timing out. With this update, the logic error in TLS handling was fixed by adjusting connection retries and socket options.

As a consequence, the **ldapsearch** command no longer ignores the NETWORK_TIMEOUT setting and returns the following error when the timeout is reached:

> `ldap_sasl_bind(SIMPLE): Can't contact LDAP server (-1)`.

Jira:RHEL-68773

OpenLDAP library no longer fails when trying to free resources

Before this update, the OpenLDAP library tried to release memory by using the **SSL_CTX_free()** function in its destructor when an application had already cleaned up these resources by invoking the **OPENSSL_cleanup()** function, either directly or via the **atexit()** function. As a consequence, users experienced failures or undefined behavior when the invalid **SSL_CTX_free()** call tried to release already-cleaned-up SSL context resources.

With this update, a safe cleanup function has been added to skip SSL context cleanup in the OpenLDAP's destructor. As a result, the SSL context now leaks if not explicitly freed, ensuring a stable application shutdown.

Jira:RHEL-68424[1]

Reindexing no longer fails when an entry RDN has the same value as the suffix DN

Before this update, if an entry's relative distinguished name (RDN) had the same value as the suffix distinguished name (DN) in the directory, then the **entryrdn** index got broken. As a result, Directory Server could perform slow search requests, get invalid results, and write alarming messages in the error log.

With this update, reindexing works as expected.

Jira:RHEL-69819[1]

11.11. SSSD

sssd-polkit-rules package content moved to sssd-common

Previously, if you needed to enable smart card support when the system security services daemon (SSSD) did not run as **root**, you had to install the **sssd-polkit-rules** package. The package provided **polkit** integration with SSSD. To resolve this issue, the **sssd-common** package now includes the content of the **sssd-polkit-rules** package and installation of a separate package is no longer required.

Jira:RHEL-50243

11.12. RED HAT ENTERPRISE LINUX SYSTEM ROLES

No property conflicts between the NetworkManager service and the NetworkManager plugin

Before this update, the **network** RHEL system role did not request user consent to restart the **NetworkManager** service when updates were available to networking packages, particularly, due to wireless interface changes. Consequently, this led to potential conflicts between the **NetworkManager** service and the **NetworkManager** plugin. Alternatively, the **NetworkManager** plugin was failing to run correctly. The problem has been fixed by making the **network** RHEL system role ask user for their consent to restart the **NetworkManager** service. As a result, there are no property conflicts between the **NetworkManager** service and the **NetworkManager** plugin in the described scenario.

Jira:RHEL-34887[1]

Implementation of multiple sets of key-value pairs of node attributes is now consistent with other cluster configuration components

The **ha_cluster** RHEL system role supports only one set of key-value pairs for each configuration item. Previously, when you configured multiple sets of node attributes, the sets were merged into a single set. With this update, the role uses only the first set you define and ignores the other sets. This behavior is now consistent with how the role implements multiple sets of key-value pairs for other configuration components that use a key-value pair structure.

Jira:RHEL-34886[1]

The postgresql RHEL system role no longer fails to set the paths to a TLS certificate and private key

The **postgresql_cert_name** variable of the **postgresql** RHEL system role defines the base path to the TLS certificate and private key without suffix on the managed node. Before this update, the role did not define internal variables for the certificate and private key. As a consequence, if you set **postgresql_cert_name**, the Ansible task failed with the following error message:

> The task includes an option with an undefined variable. The error was: '__pg_server_crt' is undefined. '__pg_server_crt' is undefined

With this update, the role correctly defines these internal variables, and the task sets the paths to the certificate and private key in the PostgreSQL configuration files.

Jira:RHEL-67418[1]

The bootloader RHEL system role generates the missing /etc/default/grub configuration file if necessary

Before this update, the **bootloader** RHEL system role expected the **/etc/default/grub** configuration file to be present. In some cases, for example on OSTtree systems, **/etc/default/grub** can be missing. As a consequence, the role failed unexpectedly. With this update, the role generates the missing file with default parameters if necessary.

Jira:RHEL-34881[1]

The podman RHEL system role can set the ownership of the host directory again

Before this update, the **podman** RHEL system role was using the **become** keyword with the user when setting the ownership of the host directory. As a consequence, the role could not properly set the ownership. With this update, the **podman** RHEL system role does not use **become** with the ordinary user. Instead, it uses the **root** user. As a result, **podman** can set the ownership of the host directory.

As a complement to this fix, the following role variables have been added to the **podman** RHEL system role:

- **podman_subuid_info** (dictionary): Exposes information used by the role from the **/etc/subuid** file. This information is needed to properly set the owner information for host directories.
- **podman_subgid_info** (dictionary): Exposes information used by the role from the **/etc/subgid** file. This information is needed to properly set the group information for host directories.

For more details about the newly added variables, see the resources in the **/usr/share/doc/rhel-system-roles/podman/** directory.

Jira:RHEL-34888[1]

The linger feature can be canceled for the correct users

When processing the instruction list of configuration items from kube files or Quadlet files, the **podman** RHEL system role was incorrectly using the user ID associated with the entire list. It did not use the user ID associated with the list item to compile the linger file name. Consequently, the linger file was not created and therefore the **podman** RHEL system role could not cancel the linger feature for the actual user if necessary. With this update, **podman** uses the correct username to construct the linger file name. As a result, the linger feature can be canceled for the correct users.

Jira:RHEL-34889[1]

The storage RHEL system role is idempotent again

The **storage** RHEL system role in some cases incorrectly calculated sizes of existing devices. Consequently, running the same playbook again without changes caused the role to attempt resizing the device that already had the correct size, instead of passing without errors. With this update, the size calculation was fixed. As a result, the role now correctly identifies that the device already has the size specified by the playbook and does not try to resize it.

Jira:RHEL-34895[1]

Running the storage RHEL system role on a system with a pre-existing Stratis pool works as expected

Before this update, the **storage** RHEL system role could not process the existing devices and device formats. This caused the role to fail on systems with a pre-existing Stratis pool, when checking if Stratis format conformed to the configuration specified by the playbook. Consequently, the playbook failed with an error, however the Stratis pool itself was not damaged or changed. This update makes the **storage** RHEL system role work correctly with Stratis devices and other formats without labelling support. As a result, running a playbook on a system with a pre-existing Stratis pool no longer fails.

Jira:RHEL-34907[1]

You cannot set the name parameter for the imuxsock input type

Before this update, the **logging** RHEL system role incorrectly set a name parameter for the **imuxsock** input type. As a consequence, this input type did not support the **name** parameter and the **rsyslog** utility on the managed node printed this error **...parameter 'name' not known — typo in config file? ...**. This update fixes the **logging** RHEL system role to ensure that the **name** parameter is not associated with the **imuxsock** input type.

Jira:RHEL-38456

GRUB2 on RHEL 10 and RHEL 9 UEFI managed nodes correctly prompts for a password

Before this update, the **bootloader** RHEL system role incorrectly placed the password information in the **/boot/efi/EFI/redhat/user.cfg** file on managed nodes that ran RHEL 10 and RHEL 9 with UEFI Secure Boot feature. The correct location was the **/boot/grub2/user.cfg** file. Consequently, when you rebooted the managed node to modify any boot loader entry, GRUB2 did not prompt you for a password. This update fixes the problem by setting the path for **user.cfg** to **/boot/grub2/** in the source code. When you reboot the operating system on a UEFI Secure Boot managed node to modify any boot loader entry, GRUB2 prompts you to input your password.

Jira:RHEL-40759[1]

Removing Quadlet-defined networks by using podman works irrespective of a custom NetworkName directive

When removing networks, the **podman** RHEL system role was using the "systemd- + name of the Quadlet file" syntax for the network name. Consequently, if the Quadlet file had a different **NetworkName** directive in it, the removal would fail. With this update, the **podman** source code has been updated to use "the Quadlet file name + the **NetworkName** directive from that file" as a name of the network to remove. As a result, removal of networks defined by Quadlet files by using the **podman** RHEL system role works both with and without a custom **NetworkName** directive in the Quadlet file.

Jira:RHEL-40760

The podman RHEL system role creates new secrets if necessary

The **podman** RHEL system role incorrectly did not check whether a secret with the same name already existed if you used the **skip_existing: true** option of the **podman_secrets** role variable. Consequently, the role did not create any new secret if using that option. This update fixes the **podman** RHEL system role to check for existing secrets if you use **skip_existing: true**. As a result, the role properly creates new secrets if they do not exist. Conversely, it does not create a secret of the same name if you use **skip_existing: true**.

Jira:RHEL-40795[1]

The network units in the Quadlet unit files are now properly cleaned up

The **podman** RHEL system role was not correctly managing the network units defined under the **[Network]** section in the Quadlet unit files. Consequently, the network units were not stopped and disabled and subsequent runs would fail due to those units not being cleaned up properly. With this update, **podman** manages the **[Network]** units, including stopping and removing. As a result, the **[Network]** units in the Quadlet unit files are properly cleaned up.

Jira:RHEL-50104[1]

The podman RHEL system role now correctly searches for subgid values

Subordinate group IDs (**subgid**) is a range of group ID values assigned to non-root users. By using these values, you can run processes with different group IDs inside a container compared to the host system. Before this update, the **podman** RHEL system role was incorrectly searching in the **subgid** values by using the group name instead of using the user name. Consequently, the difference between the user name and the group name made **podman** fail to look up the **subgid** values. This update fixes **podman** to correctly search for **subgid** values and the problem no longer occurs in this scenario.

Jira:RHEL-57100[1]

The certificate RHEL system role correctly reports an error when an issued certificate is missing the private key

When the private key of a certificate was removed, the **certmonger** utility on a managed node entered an infinite loop. Consequently, the **certificate** RHEL system role on the control node became unresponsive when re-issuing a certificate that had the private key deleted. With this update, the **certificate** RHEL system role stops processing and provides an error message with instructions for remedy. As a result, **certificate** no longer becomes unresponsive in the described scenario.

Jira:RHEL-70536[1]

The firewall RHEL system role reports changed: True when there were changes applied

During playbook processing, the **firewall_lib.py** module from the **firewall** RHEL system role was replacing the **changed** message with **False** when using the **interface** variable in the playbook and a pre-existing networking interface on the managed node. As a consequence, **firewall** reported the **changed: False** message even when there had been changes done, and the contents from the **forward_port** variable were not saved as permanent. With this update, the **firewall** RHEL system role ensures the **changed** value is not reset to **False**. As a result, the role reports **changed: True** when there are changes, and **forward_port** contents are saved as persistent.

Jira:RHEL-67412[1]

The podman RHEL system role no longer fails to process secrets when using the run_as_user variable

Before this update, the **podman** RHEL system role failed to process secrets that were specified for a particular user using the **run_as_user** variable due to missing user information. This caused errors when attempting to process secrets which have **run_as_user** set. The issue has been fixed, and the **podman** RHEL system role correctly handles secrets which are specified for a particular user using the **run_as_user** variable.

Jira:RHEL-73443[1]

The cockpit RHEL system role installs all cockpit-related packages that match a wildcard pattern

Before this update, the **dnf** module used through the **cockpit** RHEL system role did not install all **cockpit**-related packages. As a consequence, some requested packages were not installed. With this update, the source code of the **cockpit** RHEL system role was changed to use the **dnf** module directly with an asterisk wildcard package name and a list of packages to exclude. As a result, the role correctly installs all requested packages that match the wildcard pattern.

Jira:RHEL-45944[1]

The sshd RHEL system role can configure the second sshd service correctly

Running the **sshd** RHEL system role to configure the second **sshd** service on your managed nodes caused an error if you did not specify the **sshd_config_file** role variable. Consequently, your playbook would fail and the **sshd** service would not be configured correctly. To fix the problem, deriving of the main configuration file has been improved. Also, the documentation resources in the **/usr/share/doc/rhel-system-roles/sshd/** directory have been made clearer to avoid this problem. As a result, configuring the second **sshd** service as described in the above scenario works as expected.

Jira:RHEL-34879[1]

The network RHEL system role prioritizes permanent MAC address matching

When all of the following conditions were met:

- A network connection specified both an interface name and a media access control (MAC) address for configuring a parent and a virtual local area network (VLAN) connection.
- The physical interface had the same permanent and current MAC address.
- The networking configuration was applied multiple times.

The **network** RHEL system role compared the user-specified MAC address against either the permanent MAC or the current MAC address from the **sysfs** virtual filesystem. The role then treated a match with the current MAC as valid even if the interface name was different from what the user specified. As a consequence, the "no such interface exists" error occurred. With this update, the **link_info_find()** method prioritizes matching links by permanent MAC address when it is valid and available. If the permanent MAC is unavailable (None or "00:00:00:00:00:00"), the method falls back to matching the current MAC address. As a result, this change improves the robustness of MAC address matching by ensuring that permanent addresses are prioritized while maintaining a reliable fallback mechanism for interfaces with no permanent address.

Jira:RHEL-73442[1]

The new sshd_allow_restart variable enables the sshd service to be restarted when needed

Before this update, the **sshd** RHEL system role was not restarting the **sshd** service on a managed node when required. As a consequence, some changes related to configuration files from the`/etc/sysconfig/` directory and environment files were not applied. To fix the problem, the **sshd_allow_restart** (boolean, defaults to **true**) variable has been introduced to restart the **sshd** service on the managed node when necessary. As a result, the **sshd** RHEL system role now correctly applies all changes and ensures the **sshd** service actually uses those changes.

Jira:RHEL-73439[1]

The ansible-doc command provides the documentation again for the redhat.rhel_system_roles collection

Before this update, the **vpn** RHEL system role did not include documentation for the internal Ansible filter **vpn_ipaddr**. Consequently, using the **ansible-doc** command to list documentation for the **redhat.rhel_system_roles** collection would trigger an error. With this update the **vpn** RHEL system role includes the correct documentation in the correct format for the **vpn_ipaddr** filter. As a result, **ansible-doc** does not trigger any error and provides the correct documentation.

Jira:RHEL-67421[1]

The storage RHEL system role correctly resizes logical volumes

The physical volume was not resized to its maximum size when using the **grow_to_fill** feature in the **storage** RHEL system role to automatically resize LVM physical volumes after resizing the underlying virtual disks. Consequently, not all of the storage free space was available when resizing existing or creating new additional logical volumes; and the **storage** RHEL system role failed. This update fixes the problem in the source code to ensure the role always resizes the physical volumes to their maximum size when using **grow_to_fill**.

Jira:RHEL-76504[1]

The storage RHEL system role now runs as expected on RHEL 10 managed nodes with VDO

Before this update, the **blivet** module required the **kmod-kvdo** package on RHEL 10 managed nodes by using Virtual Data Optimizer (VDO). However, **kmod-kvdo** failed to install, and as a consequence caused even the **storage** RHEL system role to fail. The fix to this problem ensures that **kmod-kvdo** is not a required package for managed nodes with RHEL 10. As a result, **storage** no longer fails when managed nodes with RHEL 10 use VDO.

Jira:RHEL-81963[1]

11.13. VIRTUALIZATION

vGPU live migration no longer reports excessive amount of dirty pages

Previously, when performing virtual machine (VM) live migration with an attached NVIDIA vGPU, an excessive amount of dirty pages could have been incorrectly reported during the migration. This problem could have increased the required VM downtime during the migration and the migration could have potentially failed.

With this update, the underlying problem has been fixed and the correct amount of dirty pages is reported during the migration, which can reduce the required VM downtime during vGPU live migration in some cases.

Jira:RHEL-64308[1]

QEMU no longer prevents using SEV-SNP

Previously, when attempting to start a virtual machine (VM) with AMD SEV-SNP enabled, QEMU checked the incorrect capability of KVM, and the guest failed to start. As a consequence, running VMs with AMD SEV-SNP configured was not possible with RHEL10. This problem has been fixed, and running VMs with SEV-SNP works as expected now.

Jira:RHEL-58928[1]

Network boot for VMs now works correctly without an RNG device

Previously, when a virtual machine (VM) did not have an RNG device configured and its CPU model did not support the RDRAND feature, it was not possible to boot the VM from the network. With this update, the problem has been fixed, and VMs that do not support RDRAND can boot from the network even without an RNG device configured.

Note, however, that adding an RNG device is highly encouraged for VMs that use a CPU model that does not support RDRAND to increase security when booting from the network.

Jira:RHEL-66234

RHEL 10 guests no longer crash on restart in GCP and Alibaba

When using a RHEL 10.0 instance on Google Cloud Platform or the Alibaba Cloud, restarting the instance previously caused a kernel panic in the guest operating system if the **virtio-net** driver was in use. This issue has been fixed and RHEL 10 guests no longer crash in the described scenario.

Jira:RHEL-56981[1]

11.14. RHEL IN CLOUD ENVIRONMENTS

The mana driver with Azure Accelerated Networking assigns a correct IP address to a VM

Previously, when launching a Red Hat Enterprise Linux VM on the Azure platform with Accelerated Networking enabled, the **NetworkManager-wait-online.service** service might failed to start on boot. Consequently, the VM might failed to acquire IP address from a DHCP server when using Azure Accelerated Networking with the **mana** driver. With this fix, you need to install the latest version of the **WALinuxAgent-udev** package. As a result, Azure VMs with Accelerated Networking along with the **mana** driver will be assigned with a correct IP address at boot time.

Jira:RHEL-68796[1]

11.15. SUPPORTABILITY

The sos now obfuscates proxy passwords in several places

Previously, the **sos** utility did not obfuscate passwords from proxy links. For example **HTTP_PROXY** and **HTTPS_PROXY** in the **/etc/environment** file. As a consequence, the **sos** utility could collect sosreports with customer proxy passwords unless cleaned up before submitting. This may pose a security concern. Several of those places were discovered and fixed to obfuscate the passwords.

Red Hat continually improves the sos utility to enhance obfuscation capabilities; however, the complete removal of sensitive information is not guaranteed. Users are responsible for reviewing and manually cleaning up any confidential data before sharing it with Red Hat.

Jira:RHEL-67712[1]

The sos clean on an existing archive no longer fails

Previously, an existing archive could not be cleaned by running **sos clean** due to a regression in the **sos** code that incorrectly detected the root directory of a tarball and prevented it from cleaning data. As a consequence, **sos clean** running on an existing sosreport tarball does not clean anything within the tarball. This update adds an implementation of a proper detection of the root directory in the reordered tarball content. As a result, **sos clean** performs sensitive data obfuscation on an existing sosreport tarball correctly.

Jira:RHEL-35945

The sos stops collecting user's .ssh configuration

Previously, the **sos** utility collected the **.ssh** configuration by default from a user. As a consequence, this action caused a broken system for users that are mounted by using automount utility. With this update, the **sos** utility no longer collects the **.ssh** configuration.

Jira:RHEL-22389

11.16. CONTAINERS

Netavark no longer fails resolving DNS TCP queries

Previously, when you ran a container in a Podman network, some domain names would not resolve even though they worked on the host system or in a container not using the Podman network. With this update, Netavark supports TCP DNS queries and the problem is fixed.

Jira:RHEL-52247

CHAPTER 12. AVAILABLE BPF FEATURES

This chapter provides the complete list of Berkeley Packet Filter (BPF) features available in the kernel of this minor version of Red Hat Enterprise Linux 10. The tables include the lists of:

- System configuration and other options
- Available program types and supported helpers
- Available map types

This chapter contains automatically generated output of the **bpftool feature** command.

Table 12.1. System configuration and other options

Option	Value
unprivileged_bpf_disabled	2 (bpf() syscall restricted to privileged users, admin can change)
JIT enable	1 (enabled)
JIT harden	1 (enabled for unprivileged users)
JIT kallsyms	1 (enabled for root)
Memory limit for JIT for unprivileged users	69267617742848
CONFIG_BPF	y
CONFIG_BPF_SYSCALL	y
CONFIG_HAVE_EBPF_JIT	y
CONFIG_BPF_JIT	y
CONFIG_BPF_JIT_ALWAYS_ON	y
CONFIG_DEBUG_INFO_BTF	y
CONFIG_DEBUG_INFO_BTF_MODULES	y
CONFIG_CGROUPS	y
CONFIG_CGROUP_BPF	y
CONFIG_CGROUP_NET_CLASSID	y
CONFIG_SOCK_CGROUP_DATA	y

Option	Value
CONFIG_BPF_EVENTS	y
CONFIG_KPROBE_EVENTS	y
CONFIG_UPROBE_EVENTS	y
CONFIG_TRACING	y
CONFIG_FTRACE_SYSCALLS	y
CONFIG_FUNCTION_ERROR_INJECTION	n
CONFIG_BPF_KPROBE_OVERRIDE	n
CONFIG_NET	y
CONFIG_XDP_SOCKETS	y
CONFIG_LWTUNNEL_BPF	y
CONFIG_NET_ACT_BPF	m
CONFIG_NET_CLS_BPF	m
CONFIG_NET_CLS_ACT	y
CONFIG_NET_SCH_INGRESS	m
CONFIG_XFRM	y
CONFIG_IP_ROUTE_CLASSID	y
CONFIG_IPV6_SEG6_BPF	y
CONFIG_BPF_LIRC_MODE2	n
CONFIG_BPF_STREAM_PARSER	y
CONFIG_NETFILTER_XT_MATCH_BPF	m
CONFIG_BPFILTER	n
CONFIG_BPFILTER_UMH	n

Option	Value
CONFIG_TEST_BPF	m
CONFIG_HZ	100
bpf() syscall	available
Large insn size limit	available
Bounded loop support	available
ISA extension v2	available
ISA extension v3	available

Table 12.2. Available program types and supported helpers

Program type	Available helpers
socket_filter	bpf_map_lookup_elem, bpf_map_update_elem, bpf_map_delete_elem, bpf_ktime_get_ns, bpf_get_prandom_u32, bpf_get_smp_processor_id, bpf_tail_call, bpf_perf_event_output, bpf_skb_load_bytes, bpf_get_current_task, bpf_get_numa_node_id, bpf_get_socket_cookie, bpf_get_socket_uid, bpf_skb_load_bytes_relative, bpf_get_current_cgroup_id, bpf_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_spin_lock, bpf_spin_unlock, bpf_strtol, bpf_strtoul, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_jiffies64, bpf_get_current_ancestor_cgroup_id, bpf_ktime_get_boot_ns, bpf_ringbuf_output, bpf_ringbuf_reserve, bpf_ringbuf_submit, bpf_ringbuf_discard, bpf_ringbuf_query, bpf_skc_to_tcp6_sock, bpf_skc_to_tcp_sock, bpf_skc_to_tcp_timewait_sock, bpf_skc_to_tcp_request_sock, bpf_skc_to_udp6_sock, bpf_snprintf_btf, bpf_per_cpu_ptr, bpf_this_cpu_ptr, bpf_get_current_task_btf, bpf_ktime_get_coarse_ns, bpf_for_each_map_elem, bpf_snprintf, bpf_timer_init, bpf_timer_set_callback, bpf_timer_start, bpf_timer_cancel, bpf_task_pt_regs, bpf_skc_to_unix_sock, bpf_loop, bpf_strncmp, bpf_kptr_xchg, bpf_map_lookup_percpu_elem, bpf_skc_to_mptcp_sock, bpf_dynptr_from_mem, bpf_ringbuf_reserve_dynptr, bpf_ringbuf_submit_dynptr, bpf_ringbuf_discard_dynptr, bpf_dynptr_read, bpf_dynptr_write, bpf_dynptr_data, bpf_ktime_get_tai_ns, bpf_user_ringbuf_drain, bpf_cgrp_storage_get, bpf_cgrp_storage_delete

Program type	Available helpers
kprobe	bpf_map_lookup_elem, bpf_map_update_elem, bpf_map_delete_elem, bpf_probe_read, bpf_ktime_get_ns, bpf_get_prandom_u32, bpf_get_smp_processor_id, bpf_tail_call, bpf_get_current_pid_tgid, bpf_get_current_uid_gid, bpf_get_current_comm, bpf_perf_event_read, bpf_perf_event_output, bpf_get_stackid, bpf_get_current_task, bpf_current_task_under_cgroup, bpf_get_numa_node_id, bpf_probe_read_str, bpf_perf_event_read_value, bpf_get_stack, bpf_get_current_cgroup_id, bpf_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_spin_lock, bpf_spin_unlock, bpf_strtol, bpf_strtoul, bpf_send_signal, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_send_signal_thread, bpf_jiffies64, bpf_get_ns_current_pid_tgid, bpf_get_current_ancestor_cgroup_id, bpf_ktime_get_boot_ns, bpf_ringbuf_output, bpf_ringbuf_reserve, bpf_ringbuf_submit, bpf_ringbuf_discard, bpf_ringbuf_query, bpf_get_task_stack, bpf_copy_from_user, bpf_snprintf_btf, bpf_per_cpu_ptr, bpf_this_cpu_ptr, bpf_task_storage_get, bpf_task_storage_delete, bpf_get_current_task_btf, bpf_for_each_map_elem, bpf_snprintf, bpf_timer_init, bpf_timer_set_callback, bpf_timer_start, bpf_timer_cancel, bpf_get_func_ip, bpf_get_attach_cookie, bpf_task_pt_regs, bpf_get_branch_snapshot, bpf_find_vma, bpf_loop, bpf_strncmp, bpf_copy_from_user_task, bpf_kptr_xchg, bpf_map_lookup_percpu_elem, bpf_dynptr_from_mem, bpf_ringbuf_reserve_dynptr, bpf_ringbuf_submit_dynptr, bpf_ringbuf_discard_dynptr, bpf_dynptr_read, bpf_dynptr_write, bpf_dynptr_data, bpf_ktime_get_tai_ns, bpf_user_ringbuf_drain, bpf_cgrp_storage_get, bpf_cgrp_storage_delete

Program type	Available helpers
sched_cls	bpf_map_lookup_elem, bpf_map_update_elem, bpf_map_delete_elem, bpf_ktime_get_ns, bpf_get_prandom_u32, bpf_get_smp_processor_id, bpf_skb_store_bytes, bpf_l3_csum_replace, bpf_l4_csum_replace, bpf_tail_call, bpf_clone_redirect, bpf_get_cgroup_classid, bpf_skb_vlan_push, bpf_skb_vlan_pop, bpf_skb_get_tunnel_key, bpf_skb_set_tunnel_key, bpf_redirect, bpf_get_route_realm, bpf_perf_event_output, bpf_skb_load_bytes, bpf_csum_diff, bpf_skb_get_tunnel_opt, bpf_skb_set_tunnel_opt, bpf_skb_change_proto, bpf_skb_change_type, bpf_skb_under_cgroup, bpf_get_hash_recalc, bpf_get_current_task, bpf_skb_change_tail, bpf_skb_pull_data, bpf_csum_update, bpf_set_hash_invalid, bpf_get_numa_node_id, bpf_skb_change_head, bpf_get_socket_cookie, bpf_get_socket_uid, bpf_set_hash, bpf_skb_adjust_room, bpf_skb_get_xfrm_state, bpf_skb_load_bytes_relative, bpf_fib_lookup, bpf_skb_cgroup_id, bpf_get_current_cgroup_id, bpf_skb_ancestor_cgroup_id, bpf_sk_lookup_tcp, bpf_sk_lookup_udp, bpf_sk_release, bpf_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_spin_lock, bpf_spin_unlock, bpf_sk_fullsock, bpf_tcp_sock, bpf_skb_ecn_set_ce, bpf_get_listener_sock, bpf_skc_lookup_tcp, bpf_tcp_check_syncookie, bpf_strtol, bpf_strtoul, bpf_sk_storage_get, bpf_sk_storage_delete, bpf_tcp_gen_syncookie, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_jiffies64, bpf_get_current_ancestor_cgroup_id, bpf_sk_assign, bpf_ktime_get_boot_ns, bpf_ringbuf_output, bpf_ringbuf_reserve, bpf_ringbuf_submit, bpf_ringbuf_discard, bpf_ringbuf_query, bpf_csum_level, bpf_skc_to_tcp6_sock, bpf_skc_to_tcp_sock, bpf_skc_to_tcp_timewait_sock, bpf_skc_to_tcp_request_sock, bpf_skc_to_udp6_sock, bpf_snprintf_btf, bpf_skb_cgroup_classid, bpf_redirect_neigh, bpf_per_cpu_ptr, bpf_this_cpu_ptr, bpf_redirect_peer, bpf_get_current_task_btf, bpf_ktime_get_coarse_ns, bpf_check_mtu, bpf_for_each_map_elem, bpf_snprintf, bpf_timer_init, bpf_timer_set_callback, bpf_timer_start, bpf_timer_cancel, bpf_task_pt_regs, bpf_skc_to_unix_sock, bpf_loop, bpf_strncmp, bpf_skb_set_tstamp, bpf_kptr_xchg, bpf_map_lookup_percpu_elem, bpf_skc_to_mptcp_sock, bpf_dynptr_from_mem, bpf_ringbuf_reserve_dynptr, bpf_ringbuf_submit_dynptr, bpf_ringbuf_discard_dynptr, bpf_dynptr_read, bpf_dynptr_write, bpf_dynptr_data, bpf_tcp_raw_gen_syncookie_ipv4, bpf_tcp_raw_gen_syncookie_ipv6, bpf_tcp_raw_check_syncookie_ipv4, bpf_tcp_raw_check_syncookie_ipv6, bpf_ktime_get_tai_ns, bpf_user_ringbuf_drain, bpf_cgrp_storage_get, bpf_cgrp_storage_delete

Program type	Available helpers
sched_act	bpf_map_lookup_elem, bpf_map_update_elem, bpf_map_delete_elem, bpf_ktime_get_ns, bpf_get_prandom_u32, bpf_get_smp_processor_id, bpf_skb_store_bytes, bpf_l3_csum_replace, bpf_l4_csum_replace, bpf_tail_call, bpf_clone_redirect, bpf_get_cgroup_classid, bpf_skb_vlan_push, bpf_skb_vlan_pop, bpf_skb_get_tunnel_key, bpf_skb_set_tunnel_key, bpf_redirect, bpf_get_route_realm, bpf_perf_event_output, bpf_skb_load_bytes, bpf_csum_diff, bpf_skb_get_tunnel_opt, bpf_skb_set_tunnel_opt, bpf_skb_change_proto, bpf_skb_change_type, bpf_skb_under_cgroup, bpf_get_hash_recalc, bpf_get_current_task, bpf_skb_change_tail, bpf_skb_pull_data, bpf_csum_update, bpf_set_hash_invalid, bpf_get_numa_node_id, bpf_skb_change_head, bpf_get_socket_cookie, bpf_get_socket_uid, bpf_set_hash, bpf_skb_adjust_room, bpf_skb_get_xfrm_state, bpf_skb_load_bytes_relative, bpf_fib_lookup, bpf_skb_cgroup_id, bpf_get_current_cgroup_id, bpf_skb_ancestor_cgroup_id, bpf_sk_lookup_tcp, bpf_sk_lookup_udp, bpf_sk_release, bpf_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_spin_lock, bpf_spin_unlock, bpf_sk_fullsock, bpf_tcp_sock, bpf_skb_ecn_set_ce, bpf_get_listener_sock, bpf_skc_lookup_tcp, bpf_tcp_check_syncookie, bpf_strtol, bpf_strtoul, bpf_sk_storage_get, bpf_sk_storage_delete, bpf_tcp_gen_syncookie, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_jiffies64, bpf_get_current_ancestor_cgroup_id, bpf_sk_assign, bpf_ktime_get_boot_ns, bpf_ringbuf_output, bpf_ringbuf_reserve, bpf_ringbuf_submit, bpf_ringbuf_discard, bpf_ringbuf_query, bpf_csum_level, bpf_skc_to_tcp6_sock, bpf_skc_to_tcp_sock, bpf_skc_to_tcp_timewait_sock, bpf_skc_to_tcp_request_sock, bpf_skc_to_udp6_sock, bpf_snprintf_btf, bpf_skb_cgroup_classid, bpf_redirect_neigh, bpf_per_cpu_ptr, bpf_this_cpu_ptr, bpf_redirect_peer, bpf_get_current_task_btf, bpf_ktime_get_coarse_ns, bpf_check_mtu, bpf_for_each_map_elem, bpf_snprintf, bpf_timer_init, bpf_timer_set_callback, bpf_timer_start, bpf_timer_cancel, bpf_task_pt_regs, bpf_skc_to_unix_sock, bpf_loop, bpf_strncmp, bpf_skb_set_tstamp, bpf_kptr_xchg, bpf_map_lookup_percpu_elem, bpf_skc_to_mptcp_sock, bpf_dynptr_from_mem, bpf_ringbuf_reserve_dynptr, bpf_ringbuf_submit_dynptr, bpf_ringbuf_discard_dynptr, bpf_dynptr_read, bpf_dynptr_write, bpf_dynptr_data, bpf_tcp_raw_gen_syncookie_ipv4, bpf_tcp_raw_gen_syncookie_ipv6, bpf_tcp_raw_check_syncookie_ipv4, bpf_tcp_raw_check_syncookie_ipv6, bpf_ktime_get_tai_ns, bpf_user_ringbuf_drain, bpf_cgrp_storage_get, bpf_cgrp_storage_delete

Program type	Available helpers
tracepoint	bpf_map_lookup_elem, bpf_map_update_elem, bpf_map_delete_elem, bpf_probe_read, bpf_ktime_get_ns, bpf_get_prandom_u32, bpf_get_smp_processor_id, bpf_tail_call, bpf_get_current_pid_tgid, bpf_get_current_uid_gid, bpf_get_current_comm, bpf_perf_event_read, bpf_perf_event_output, bpf_get_stackid, bpf_get_current_task, bpf_current_task_under_cgroup, bpf_get_numa_node_id, bpf_probe_read_str, bpf_perf_event_read_value, bpf_get_stack, bpf_get_current_cgroup_id, bpf_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_spin_lock, bpf_spin_unlock, bpf_strtol, bpf_strtoul, bpf_send_signal, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_send_signal_thread, bpf_jiffies64, bpf_get_ns_current_pid_tgid, bpf_get_current_ancestor_cgroup_id, bpf_ktime_get_boot_ns, bpf_ringbuf_output, bpf_ringbuf_reserve, bpf_ringbuf_submit, bpf_ringbuf_discard, bpf_ringbuf_query, bpf_get_task_stack, bpf_copy_from_user, bpf_snprintf_btf, bpf_per_cpu_ptr, bpf_this_cpu_ptr, bpf_task_storage_get, bpf_task_storage_delete, bpf_get_current_task_btf, bpf_for_each_map_elem, bpf_snprintf, bpf_timer_init, bpf_timer_set_callback, bpf_timer_start, bpf_timer_cancel, bpf_get_func_ip, bpf_get_attach_cookie, bpf_task_pt_regs, bpf_get_branch_snapshot, bpf_find_vma, bpf_loop, bpf_strncmp, bpf_copy_from_user_task, bpf_kptr_xchg, bpf_map_lookup_percpu_elem, bpf_dynptr_from_mem, bpf_ringbuf_reserve_dynptr, bpf_ringbuf_submit_dynptr, bpf_ringbuf_discard_dynptr, bpf_dynptr_read, bpf_dynptr_write, bpf_dynptr_data, bpf_ktime_get_tai_ns, bpf_user_ringbuf_drain, bpf_cgrp_storage_get, bpf_cgrp_storage_delete
xdp	bpf_map_lookup_elem, bpf_map_update_elem, bpf_map_delete_elem, bpf_ktime_get_ns, bpf_get_prandom_u32, bpf_get_smp_processor_id, bpf_tail_call, bpf_redirect, bpf_perf_event_output, bpf_csum_diff, bpf_get_current_task, bpf_get_numa_node_id, bpf_xdp_adjust_head, bpf_redirect_map, bpf_xdp_adjust_meta, bpf_xdp_adjust_tail, bpf_fib_lookup, bpf_get_current_cgroup_id, bpf_sk_lookup_tcp, bpf_sk_lookup_udp, bpf_sk_release, bpf_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_spin_lock, bpf_spin_unlock, bpf_skc_lookup_tcp, bpf_tcp_check_syncookie, bpf_strtol, bpf_strtoul, bpf_tcp_gen_syncookie, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_jiffies64, bpf_get_current_ancestor_cgroup_id, bpf_ktime_get_boot_ns, bpf_ringbuf_output, bpf_ringbuf_reserve, bpf_ringbuf_submit, bpf_ringbuf_discard, bpf_ringbuf_query, bpf_skc_to_tcp6_sock, bpf_skc_to_tcp_sock, bpf_skc_to_tcp_timewait_sock, bpf_skc_to_tcp_request_sock, bpf_skc_to_udp6_sock, bpf_snprintf_btf, bpf_per_cpu_ptr, bpf_this_cpu_ptr, bpf_get_current_task_btf, bpf_ktime_get_coarse_ns, bpf_check_mtu, bpf_for_each_map_elem, bpf_snprintf, bpf_timer_init, bpf_timer_set_callback, bpf_timer_start, bpf_timer_cancel, bpf_task_pt_regs, bpf_skc_to_unix_sock, bpf_loop, bpf_strncmp, bpf_xdp_get_buff_len, bpf_xdp_load_bytes, bpf_xdp_store_bytes, bpf_kptr_xchg, bpf_map_lookup_percpu_elem, bpf_skc_to_mptcp_sock, bpf_dynptr_from_mem, bpf_ringbuf_reserve_dynptr, bpf_ringbuf_submit_dynptr, bpf_ringbuf_discard_dynptr, bpf_dynptr_read, bpf_dynptr_write, bpf_dynptr_data, bpf_tcp_raw_gen_syncookie_ipv4, bpf_tcp_raw_gen_syncookie_ipv6, bpf_tcp_raw_check_syncookie_ipv4, bpf_tcp_raw_check_syncookie_ipv6, bpf_ktime_get_tai_ns, bpf_user_ringbuf_drain, bpf_cgrp_storage_get, bpf_cgrp_storage_delete

Program type	Available helpers
perf_event	bpf_map_lookup_elem, bpf_map_update_elem, bpf_map_delete_elem, bpf_probe_read, bpf_ktime_get_ns, bpf_get_prandom_u32, bpf_get_smp_processor_id, bpf_tail_call, bpf_get_current_pid_tgid, bpf_get_current_uid_gid, bpf_get_current_comm, bpf_perf_event_read, bpf_perf_event_output, bpf_get_stackid, bpf_get_current_task, bpf_current_task_under_cgroup, bpf_get_numa_node_id, bpf_probe_read_str, bpf_perf_event_read_value, bpf_perf_prog_read_value, bpf_get_stack, bpf_get_current_cgroup_id, bpf_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_spin_lock, bpf_spin_unlock, bpf_strtol, bpf_strtoul, bpf_send_signal, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_send_signal_thread, bpf_jiffies64, bpf_read_branch_records, bpf_get_ns_current_pid_tgid, bpf_get_current_ancestor_cgroup_id, bpf_ktime_get_boot_ns, bpf_ringbuf_output, bpf_ringbuf_reserve, bpf_ringbuf_submit, bpf_ringbuf_discard, bpf_ringbuf_query, bpf_get_task_stack, bpf_copy_from_user, bpf_snprintf_btf, bpf_per_cpu_ptr, bpf_this_cpu_ptr, bpf_task_storage_get, bpf_task_storage_delete, bpf_get_current_task_btf, bpf_for_each_map_elem, bpf_snprintf, bpf_timer_init, bpf_timer_set_callback, bpf_timer_start, bpf_timer_cancel, bpf_get_func_ip, bpf_get_attach_cookie, bpf_task_pt_regs, bpf_get_branch_snapshot, bpf_find_vma, bpf_loop, bpf_strncmp, bpf_copy_from_user_task, bpf_kptr_xchg, bpf_map_lookup_percpu_elem, bpf_dynptr_from_mem, bpf_ringbuf_reserve_dynptr, bpf_ringbuf_submit_dynptr, bpf_ringbuf_discard_dynptr, bpf_dynptr_read, bpf_dynptr_write, bpf_dynptr_data, bpf_ktime_get_tai_ns, bpf_user_ringbuf_drain, bpf_cgrp_storage_get, bpf_cgrp_storage_delete
cgroup_skb	bpf_map_lookup_elem, bpf_map_update_elem, bpf_map_delete_elem, bpf_ktime_get_ns, bpf_get_prandom_u32, bpf_get_smp_processor_id, bpf_tail_call, bpf_perf_event_output, bpf_skb_load_bytes, bpf_get_current_task, bpf_get_numa_node_id, bpf_get_socket_cookie, bpf_get_socket_uid, bpf_skb_load_bytes_relative, bpf_skb_cgroup_id, bpf_get_current_cgroup_id, bpf_get_local_storage, bpf_skb_ancestor_cgroup_id, bpf_sk_lookup_tcp, bpf_sk_lookup_udp, bpf_sk_release, bpf_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_spin_lock, bpf_spin_unlock, bpf_sk_fullsock, bpf_tcp_sock, bpf_skb_ecn_set_ce, bpf_get_listener_sock, bpf_skc_lookup_tcp, bpf_strtol, bpf_strtoul, bpf_sk_storage_get, bpf_sk_storage_delete, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_jiffies64, bpf_get_current_ancestor_cgroup_id, bpf_ktime_get_boot_ns, bpf_sk_cgroup_id, bpf_sk_ancestor_cgroup_id, bpf_ringbuf_output, bpf_ringbuf_reserve, bpf_ringbuf_submit, bpf_ringbuf_discard, bpf_ringbuf_query, bpf_skc_to_tcp6_sock, bpf_skc_to_tcp_sock, bpf_skc_to_tcp_timewait_sock, bpf_skc_to_tcp_request_sock, bpf_skc_to_udp6_sock, bpf_snprintf_btf, bpf_per_cpu_ptr, bpf_this_cpu_ptr, bpf_get_current_task_btf, bpf_ktime_get_coarse_ns, bpf_for_each_map_elem, bpf_snprintf, bpf_timer_init, bpf_timer_set_callback, bpf_timer_start, bpf_timer_cancel, bpf_task_pt_regs, bpf_skc_to_unix_sock, bpf_loop, bpf_strncmp, bpf_kptr_xchg, bpf_map_lookup_percpu_elem, bpf_skc_to_mptcp_sock, bpf_dynptr_from_mem, bpf_ringbuf_reserve_dynptr, bpf_ringbuf_submit_dynptr, bpf_ringbuf_discard_dynptr, bpf_dynptr_read, bpf_dynptr_write, bpf_dynptr_data, bpf_ktime_get_tai_ns, bpf_user_ringbuf_drain, bpf_cgrp_storage_get, bpf_cgrp_storage_delete

Program type	Available helpers
cgroup_sock	bpf_map_lookup_elem, bpf_map_update_elem, bpf_map_delete_elem, bpf_ktime_get_ns, bpf_get_prandom_u32, bpf_get_smp_processor_id, bpf_tail_call, bpf_get_current_pid_tgid, bpf_get_current_uid_gid, bpf_get_current_comm, bpf_get_cgroup_classid, bpf_perf_event_output, bpf_get_current_task, bpf_get_numa_node_id, bpf_get_socket_cookie, bpf_get_current_cgroup_id, bpf_get_local_storage, bpf_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_spin_lock, bpf_spin_unlock, bpf_strtol, bpf_strtoul, bpf_sk_storage_get, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_jiffies64, bpf_get_netns_cookie, bpf_get_current_ancestor_cgroup_id, bpf_ktime_get_boot_ns, bpf_ringbuf_output, bpf_ringbuf_reserve, bpf_ringbuf_submit, bpf_ringbuf_discard, bpf_ringbuf_query, bpf_snprintf_btf, bpf_per_cpu_ptr, bpf_this_cpu_ptr, bpf_get_current_task_btf, bpf_ktime_get_coarse_ns, bpf_for_each_map_elem, bpf_snprintf, bpf_timer_init, bpf_timer_set_callback, bpf_timer_start, bpf_timer_cancel, bpf_task_pt_regs, bpf_loop, bpf_strncmp, bpf_get_retval, bpf_set_retval, bpf_kptr_xchg, bpf_map_lookup_percpu_elem, bpf_dynptr_from_mem, bpf_ringbuf_reserve_dynptr, bpf_ringbuf_submit_dynptr, bpf_ringbuf_discard_dynptr, bpf_dynptr_read, bpf_dynptr_write, bpf_dynptr_data, bpf_ktime_get_tai_ns, bpf_user_ringbuf_drain, bpf_cgrp_storage_get, bpf_cgrp_storage_delete
lwt_in	bpf_map_lookup_elem, bpf_map_update_elem, bpf_map_delete_elem, bpf_ktime_get_ns, bpf_get_prandom_u32, bpf_get_smp_processor_id, bpf_tail_call, bpf_get_cgroup_classid, bpf_get_route_realm, bpf_perf_event_output, bpf_skb_load_bytes, bpf_csum_diff, bpf_skb_under_cgroup, bpf_get_hash_recalc, bpf_get_current_task, bpf_skb_pull_data, bpf_get_numa_node_id, bpf_lwt_push_encap, bpf_get_current_cgroup_id, bpf_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_spin_lock, bpf_spin_unlock, bpf_strtol, bpf_strtoul, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_jiffies64, bpf_get_current_ancestor_cgroup_id, bpf_ktime_get_boot_ns, bpf_ringbuf_output, bpf_ringbuf_reserve, bpf_ringbuf_submit, bpf_ringbuf_discard, bpf_ringbuf_query, bpf_skc_to_tcp6_sock, bpf_skc_to_tcp_sock, bpf_skc_to_tcp_timewait_sock, bpf_skc_to_tcp_request_sock, bpf_skc_to_udp6_sock, bpf_snprintf_btf, bpf_per_cpu_ptr, bpf_this_cpu_ptr, bpf_get_current_task_btf, bpf_ktime_get_coarse_ns, bpf_for_each_map_elem, bpf_snprintf, bpf_timer_init, bpf_timer_set_callback, bpf_timer_start, bpf_timer_cancel, bpf_task_pt_regs, bpf_skc_to_unix_sock, bpf_loop, bpf_strncmp, bpf_kptr_xchg, bpf_map_lookup_percpu_elem, bpf_skc_to_mptcp_sock, bpf_dynptr_from_mem, bpf_ringbuf_reserve_dynptr, bpf_ringbuf_submit_dynptr, bpf_ringbuf_discard_dynptr, bpf_dynptr_read, bpf_dynptr_write, bpf_dynptr_data, bpf_ktime_get_tai_ns, bpf_user_ringbuf_drain, bpf_cgrp_storage_get, bpf_cgrp_storage_delete

Program type	Available helpers
lwt_out	bpf_map_lookup_elem, bpf_map_update_elem, bpf_map_delete_elem, bpf_ktime_get_ns, bpf_get_prandom_u32, bpf_get_smp_processor_id, bpf_tail_call, bpf_get_cgroup_classid, bpf_get_route_realm, bpf_perf_event_output, bpf_skb_load_bytes, bpf_csum_diff, bpf_skb_under_cgroup, bpf_get_hash_recalc, bpf_get_current_task, bpf_skb_pull_data, bpf_get_numa_node_id, bpf_get_current_cgroup_id, bpf_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_spin_lock, bpf_spin_unlock, bpf_strtol, bpf_strtoul, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_jiffies64, bpf_get_current_ancestor_cgroup_id, bpf_ktime_get_boot_ns, bpf_ringbuf_output, bpf_ringbuf_reserve, bpf_ringbuf_submit, bpf_ringbuf_discard, bpf_ringbuf_query, bpf_skc_to_tcp6_sock, bpf_skc_to_tcp_sock, bpf_skc_to_tcp_timewait_sock, bpf_skc_to_tcp_request_sock, bpf_skc_to_udp6_sock, bpf_snprintf_btf, bpf_per_cpu_ptr, bpf_this_cpu_ptr, bpf_get_current_task_btf, bpf_ktime_get_coarse_ns, bpf_for_each_map_elem, bpf_snprintf, bpf_timer_init, bpf_timer_set_callback, bpf_timer_start, bpf_timer_cancel, bpf_task_pt_regs, bpf_skc_to_unix_sock, bpf_loop, bpf_strncmp, bpf_kptr_xchg, bpf_map_lookup_percpu_elem, bpf_skc_to_mptcp_sock, bpf_dynptr_from_mem, bpf_ringbuf_reserve_dynptr, bpf_ringbuf_submit_dynptr, bpf_ringbuf_discard_dynptr, bpf_dynptr_read, bpf_dynptr_write, bpf_dynptr_data, bpf_ktime_get_tai_ns, bpf_user_ringbuf_drain, bpf_cgrp_storage_get, bpf_cgrp_storage_delete
lwt_xmit	bpf_map_lookup_elem, bpf_map_update_elem, bpf_map_delete_elem, bpf_ktime_get_ns, bpf_get_prandom_u32, bpf_get_smp_processor_id, bpf_skb_store_bytes, bpf_l3_csum_replace, bpf_l4_csum_replace, bpf_tail_call, bpf_clone_redirect, bpf_get_cgroup_classid, bpf_skb_get_tunnel_key, bpf_skb_set_tunnel_key, bpf_redirect, bpf_get_route_realm, bpf_perf_event_output, bpf_skb_load_bytes, bpf_csum_diff, bpf_skb_get_tunnel_opt, bpf_skb_set_tunnel_opt, bpf_skb_under_cgroup, bpf_get_hash_recalc, bpf_get_current_task, bpf_skb_change_tail, bpf_skb_pull_data, bpf_csum_update, bpf_set_hash_invalid, bpf_get_numa_node_id, bpf_skb_change_head, bpf_lwt_push_encap, bpf_get_current_cgroup_id, bpf_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_spin_lock, bpf_spin_unlock, bpf_strtol, bpf_strtoul, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_jiffies64, bpf_get_current_ancestor_cgroup_id, bpf_ktime_get_boot_ns, bpf_ringbuf_output, bpf_ringbuf_reserve, bpf_ringbuf_submit, bpf_ringbuf_discard, bpf_ringbuf_query, bpf_csum_level, bpf_skc_to_tcp6_sock, bpf_skc_to_tcp_sock, bpf_skc_to_tcp_timewait_sock, bpf_skc_to_tcp_request_sock, bpf_skc_to_udp6_sock, bpf_snprintf_btf, bpf_per_cpu_ptr, bpf_this_cpu_ptr, bpf_get_current_task_btf, bpf_ktime_get_coarse_ns, bpf_for_each_map_elem, bpf_snprintf, bpf_timer_init, bpf_timer_set_callback, bpf_timer_start, bpf_timer_cancel, bpf_task_pt_regs, bpf_skc_to_unix_sock, bpf_loop, bpf_strncmp, bpf_kptr_xchg, bpf_map_lookup_percpu_elem, bpf_skc_to_mptcp_sock, bpf_dynptr_from_mem, bpf_ringbuf_reserve_dynptr, bpf_ringbuf_submit_dynptr, bpf_ringbuf_discard_dynptr, bpf_dynptr_read, bpf_dynptr_write, bpf_dynptr_data, bpf_ktime_get_tai_ns, bpf_user_ringbuf_drain, bpf_cgrp_storage_get, bpf_cgrp_storage_delete

Program type	Available helpers
sock_ops	bpf_map_lookup_elem, bpf_map_update_elem, bpf_map_delete_elem, bpf_ktime_get_ns, bpf_get_prandom_u32, bpf_get_smp_processor_id, bpf_tail_call, bpf_perf_event_output, bpf_get_current_task, bpf_get_numa_node_id, bpf_get_socket_cookie, bpf_setsockopt, bpf_sock_map_update, bpf_getsockopt, bpf_sock_ops_cb_flags_set, bpf_sock_hash_update, bpf_get_current_cgroup_id, bpf_get_local_storage, bpf_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_spin_lock, bpf_spin_unlock, bpf_tcp_sock, bpf_strtol, bpf_strtoul, bpf_sk_storage_get, bpf_sk_storage_delete, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_jiffies64, bpf_get_netns_cookie, bpf_get_current_ancestor_cgroup_id, bpf_ktime_get_boot_ns, bpf_ringbuf_output, bpf_ringbuf_reserve, bpf_ringbuf_submit, bpf_ringbuf_discard, bpf_ringbuf_query, bpf_skc_to_tcp6_sock, bpf_skc_to_tcp_sock, bpf_skc_to_tcp_timewait_sock, bpf_skc_to_tcp_request_sock, bpf_skc_to_udp6_sock, bpf_load_hdr_opt, bpf_store_hdr_opt, bpf_reserve_hdr_opt, bpf_snprintf_btf, bpf_per_cpu_ptr, bpf_this_cpu_ptr, bpf_get_current_task_btf, bpf_ktime_get_coarse_ns, bpf_for_each_map_elem, bpf_snprintf, bpf_timer_init, bpf_timer_set_callback, bpf_timer_start, bpf_timer_cancel, bpf_task_pt_regs, bpf_skc_to_unix_sock, bpf_loop, bpf_strncmp, bpf_kptr_xchg, bpf_map_lookup_percpu_elem, bpf_skc_to_mptcp_sock, bpf_dynptr_from_mem, bpf_ringbuf_reserve_dynptr, bpf_ringbuf_submit_dynptr, bpf_ringbuf_discard_dynptr, bpf_dynptr_read, bpf_dynptr_write, bpf_dynptr_data, bpf_ktime_get_tai_ns, bpf_user_ringbuf_drain, bpf_cgrp_storage_get, bpf_cgrp_storage_delete
sk_skb	bpf_map_lookup_elem, bpf_map_update_elem, bpf_map_delete_elem, bpf_ktime_get_ns, bpf_get_prandom_u32, bpf_get_smp_processor_id, bpf_skb_store_bytes, bpf_tail_call, bpf_perf_event_output, bpf_skb_load_bytes, bpf_get_current_task, bpf_skb_change_tail, bpf_skb_pull_data, bpf_get_numa_node_id, bpf_skb_change_head, bpf_get_socket_cookie, bpf_get_socket_uid, bpf_skb_adjust_room, bpf_sk_redirect_map, bpf_sk_redirect_hash, bpf_get_current_cgroup_id, bpf_sk_lookup_tcp, bpf_sk_lookup_udp, bpf_sk_release, bpf_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_spin_lock, bpf_spin_unlock, bpf_skc_lookup_tcp, bpf_strtol, bpf_strtoul, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_jiffies64, bpf_get_current_ancestor_cgroup_id, bpf_ktime_get_boot_ns, bpf_ringbuf_output, bpf_ringbuf_reserve, bpf_ringbuf_submit, bpf_ringbuf_discard, bpf_ringbuf_query, bpf_skc_to_tcp6_sock, bpf_skc_to_tcp_sock, bpf_skc_to_tcp_timewait_sock, bpf_skc_to_tcp_request_sock, bpf_skc_to_udp6_sock, bpf_snprintf_btf, bpf_per_cpu_ptr, bpf_this_cpu_ptr, bpf_get_current_task_btf, bpf_ktime_get_coarse_ns, bpf_for_each_map_elem, bpf_snprintf, bpf_timer_init, bpf_timer_set_callback, bpf_timer_start, bpf_timer_cancel, bpf_task_pt_regs, bpf_skc_to_unix_sock, bpf_loop, bpf_strncmp, bpf_kptr_xchg, bpf_map_lookup_percpu_elem, bpf_skc_to_mptcp_sock, bpf_dynptr_from_mem, bpf_ringbuf_reserve_dynptr, bpf_ringbuf_submit_dynptr, bpf_ringbuf_discard_dynptr, bpf_dynptr_read, bpf_dynptr_write, bpf_dynptr_data, bpf_ktime_get_tai_ns, bpf_user_ringbuf_drain, bpf_cgrp_storage_get, bpf_cgrp_storage_delete

Program type	Available helpers
cgroup_device	bpf_map_lookup_elem, bpf_map_update_elem, bpf_map_delete_elem, bpf_ktime_get_ns, bpf_get_prandom_u32, bpf_get_smp_processor_id, bpf_tail_call, bpf_get_current_pid_tgid, bpf_get_current_uid_gid, bpf_get_current_comm, bpf_get_cgroup_classid, bpf_perf_event_output, bpf_get_current_task, bpf_get_numa_node_id, bpf_get_current_cgroup_id, bpf_get_local_storage, bpf_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_spin_lock, bpf_spin_unlock, bpf_strtol, bpf_strtoul, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_jiffies64, bpf_get_current_ancestor_cgroup_id, bpf_ktime_get_boot_ns, bpf_ringbuf_output, bpf_ringbuf_reserve, bpf_ringbuf_submit, bpf_ringbuf_discard, bpf_ringbuf_query, bpf_snprintf_btf, bpf_per_cpu_ptr, bpf_this_cpu_ptr, bpf_get_current_task_btf, bpf_for_each_map_elem, bpf_snprintf, bpf_timer_init, bpf_timer_set_callback, bpf_timer_start, bpf_timer_cancel, bpf_task_pt_regs, bpf_loop, bpf_strncmp, bpf_kptr_xchg, bpf_map_lookup_percpu_elem, bpf_dynptr_from_mem, bpf_ringbuf_reserve_dynptr, bpf_ringbuf_submit_dynptr, bpf_ringbuf_discard_dynptr, bpf_dynptr_read, bpf_dynptr_write, bpf_dynptr_data, bpf_ktime_get_tai_ns, bpf_user_ringbuf_drain, bpf_cgrp_storage_get, bpf_cgrp_storage_delete
sk_msg	bpf_map_lookup_elem, bpf_map_update_elem, bpf_map_delete_elem, bpf_ktime_get_ns, bpf_get_prandom_u32, bpf_get_smp_processor_id, bpf_tail_call, bpf_get_current_pid_tgid, bpf_get_current_uid_gid, bpf_get_cgroup_classid, bpf_perf_event_output, bpf_get_current_task, bpf_get_numa_node_id, bpf_msg_redirect_map, bpf_msg_apply_bytes, bpf_msg_cork_bytes, bpf_msg_pull_data, bpf_msg_redirect_hash, bpf_get_current_cgroup_id, bpf_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_msg_push_data, bpf_msg_pop_data, bpf_spin_lock, bpf_spin_unlock, bpf_strtol, bpf_strtoul, bpf_sk_storage_get, bpf_sk_storage_delete, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_jiffies64, bpf_get_netns_cookie, bpf_get_current_ancestor_cgroup_id, bpf_ktime_get_boot_ns, bpf_ringbuf_output, bpf_ringbuf_reserve, bpf_ringbuf_submit, bpf_ringbuf_discard, bpf_ringbuf_query, bpf_skc_to_tcp6_sock, bpf_skc_to_tcp_sock, bpf_skc_to_tcp_timewait_sock, bpf_skc_to_tcp_request_sock, bpf_skc_to_udp6_sock, bpf_snprintf_btf, bpf_per_cpu_ptr, bpf_this_cpu_ptr, bpf_get_current_task_btf, bpf_ktime_get_coarse_ns, bpf_for_each_map_elem, bpf_snprintf, bpf_timer_init, bpf_timer_set_callback, bpf_timer_start, bpf_timer_cancel, bpf_task_pt_regs, bpf_skc_to_unix_sock, bpf_loop, bpf_strncmp, bpf_kptr_xchg, bpf_map_lookup_percpu_elem, bpf_skc_to_mptcp_sock, bpf_dynptr_from_mem, bpf_ringbuf_reserve_dynptr, bpf_ringbuf_submit_dynptr, bpf_ringbuf_discard_dynptr, bpf_dynptr_read, bpf_dynptr_write, bpf_dynptr_data, bpf_ktime_get_tai_ns, bpf_user_ringbuf_drain, bpf_cgrp_storage_get, bpf_cgrp_storage_delete

Program type	Available helpers
raw_tracepoint	bpf_map_lookup_elem, bpf_map_update_elem, bpf_map_delete_elem, bpf_probe_read, bpf_ktime_get_ns, bpf_get_prandom_u32, bpf_get_smp_processor_id, bpf_tail_call, bpf_get_current_pid_tgid, bpf_get_current_uid_gid, bpf_get_current_comm, bpf_perf_event_read, bpf_perf_event_output, bpf_get_stackid, bpf_get_current_task, bpf_current_task_under_cgroup, bpf_get_numa_node_id, bpf_probe_read_str, bpf_perf_event_read_value, bpf_get_stack, bpf_get_current_cgroup_id, bpf_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_spin_lock, bpf_spin_unlock, bpf_strtol, bpf_strtoul, bpf_send_signal, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_send_signal_thread, bpf_jiffies64, bpf_get_ns_current_pid_tgid, bpf_get_current_ancestor_cgroup_id, bpf_ktime_get_boot_ns, bpf_ringbuf_output, bpf_ringbuf_reserve, bpf_ringbuf_submit, bpf_ringbuf_discard, bpf_ringbuf_query, bpf_get_task_stack, bpf_copy_from_user, bpf_snprintf_btf, bpf_per_cpu_ptr, bpf_this_cpu_ptr, bpf_task_storage_get, bpf_task_storage_delete, bpf_get_current_task_btf, bpf_for_each_map_elem, bpf_snprintf, bpf_timer_init, bpf_timer_set_callback, bpf_timer_start, bpf_timer_cancel, bpf_get_func_ip, bpf_task_pt_regs, bpf_get_branch_snapshot, bpf_find_vma, bpf_loop, bpf_strncmp, bpf_copy_from_user_task, bpf_kptr_xchg, bpf_map_lookup_percpu_elem, bpf_dynptr_from_mem, bpf_ringbuf_reserve_dynptr, bpf_ringbuf_submit_dynptr, bpf_ringbuf_discard_dynptr, bpf_dynptr_read, bpf_dynptr_write, bpf_dynptr_data, bpf_ktime_get_tai_ns, bpf_user_ringbuf_drain, bpf_cgrp_storage_get, bpf_cgrp_storage_delete
cgroup_sock_addr	bpf_map_lookup_elem, bpf_map_update_elem, bpf_map_delete_elem, bpf_ktime_get_ns, bpf_get_prandom_u32, bpf_get_smp_processor_id, bpf_tail_call, bpf_get_current_pid_tgid, bpf_get_current_uid_gid, bpf_get_current_comm, bpf_get_cgroup_classid, bpf_perf_event_output, bpf_get_current_task, bpf_get_numa_node_id, bpf_get_socket_cookie, bpf_setsockopt, bpf_getsockopt, bpf_bind, bpf_get_current_cgroup_id, bpf_get_local_storage, bpf_sk_lookup_tcp, bpf_sk_lookup_udp, bpf_sk_release, bpf_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_spin_lock, bpf_spin_unlock, bpf_skc_lookup_tcp, bpf_strtol, bpf_strtoul, bpf_sk_storage_get, bpf_sk_storage_delete, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_jiffies64, bpf_get_netns_cookie, bpf_get_current_ancestor_cgroup_id, bpf_ktime_get_boot_ns, bpf_ringbuf_output, bpf_ringbuf_reserve, bpf_ringbuf_submit, bpf_ringbuf_discard, bpf_ringbuf_query, bpf_skc_to_tcp6_sock, bpf_skc_to_tcp_sock, bpf_skc_to_tcp_timewait_sock, bpf_skc_to_tcp_request_sock, bpf_skc_to_udp6_sock, bpf_snprintf_btf, bpf_per_cpu_ptr, bpf_this_cpu_ptr, bpf_get_current_task_btf, bpf_ktime_get_coarse_ns, bpf_for_each_map_elem, bpf_snprintf, bpf_timer_init, bpf_timer_set_callback, bpf_timer_start, bpf_timer_cancel, bpf_task_pt_regs, bpf_skc_to_unix_sock, bpf_loop, bpf_strncmp, bpf_get_retval, bpf_set_retval, bpf_kptr_xchg, bpf_map_lookup_percpu_elem, bpf_skc_to_mptcp_sock, bpf_dynptr_from_mem, bpf_ringbuf_reserve_dynptr, bpf_ringbuf_submit_dynptr, bpf_ringbuf_discard_dynptr, bpf_dynptr_read, bpf_dynptr_write, bpf_dynptr_data, bpf_ktime_get_tai_ns, bpf_user_ringbuf_drain, bpf_cgrp_storage_get, bpf_cgrp_storage_delete

Program type	Available helpers
lwt_seg6local	bpf_map_lookup_elem, bpf_map_update_elem, bpf_map_delete_elem, bpf_ktime_get_ns, bpf_get_prandom_u32, bpf_get_smp_processor_id, bpf_tail_call, bpf_get_cgroup_classid, bpf_get_route_realm, bpf_perf_event_output, bpf_skb_load_bytes, bpf_csum_diff, bpf_skb_under_cgroup, bpf_get_hash_recalc, bpf_get_current_task, bpf_skb_pull_data, bpf_get_numa_node_id, bpf_lwt_seg6_store_bytes, bpf_lwt_seg6_adjust_srh, bpf_lwt_seg6_action, bpf_get_current_cgroup_id, bpf_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_spin_lock, bpf_spin_unlock, bpf_strtol, bpf_strtoul, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_jiffies64, bpf_get_current_ancestor_cgroup_id, bpf_ktime_get_boot_ns, bpf_ringbuf_output, bpf_ringbuf_reserve, bpf_ringbuf_submit, bpf_ringbuf_discard, bpf_ringbuf_query, bpf_skc_to_tcp6_sock, bpf_skc_to_tcp_sock, bpf_skc_to_tcp_timewait_sock, bpf_skc_to_tcp_request_sock, bpf_skc_to_udp6_sock, bpf_snprintf_btf, bpf_per_cpu_ptr, bpf_this_cpu_ptr, bpf_get_current_task_btf, bpf_ktime_get_coarse_ns, bpf_for_each_map_elem, bpf_snprintf, bpf_timer_init, bpf_timer_set_callback, bpf_timer_start, bpf_timer_cancel, bpf_task_pt_regs, bpf_skc_to_unix_sock, bpf_loop, bpf_strncmp, bpf_kptr_xchg, bpf_map_lookup_percpu_elem, bpf_skc_to_mptcp_sock, bpf_dynptr_from_mem, bpf_ringbuf_reserve_dynptr, bpf_ringbuf_submit_dynptr, bpf_ringbuf_discard_dynptr, bpf_dynptr_read, bpf_dynptr_write, bpf_dynptr_data, bpf_ktime_get_tai_ns, bpf_user_ringbuf_drain, bpf_cgrp_storage_get, bpf_cgrp_storage_delete
lirc_mode2	not supported
sk_reuseport	bpf_map_lookup_elem, bpf_map_update_elem, bpf_map_delete_elem, bpf_ktime_get_ns, bpf_get_prandom_u32, bpf_get_smp_processor_id, bpf_tail_call, bpf_skb_load_bytes, bpf_get_current_task, bpf_get_numa_node_id, bpf_get_socket_cookie, bpf_skb_load_bytes_relative, bpf_get_current_cgroup_id, bpf_sk_select_reuseport, bpf_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_spin_lock, bpf_spin_unlock, bpf_strtol, bpf_strtoul, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_jiffies64, bpf_get_current_ancestor_cgroup_id, bpf_ktime_get_boot_ns, bpf_ringbuf_output, bpf_ringbuf_reserve, bpf_ringbuf_submit, bpf_ringbuf_discard, bpf_ringbuf_query, bpf_snprintf_btf, bpf_per_cpu_ptr, bpf_this_cpu_ptr, bpf_get_current_task_btf, bpf_ktime_get_coarse_ns, bpf_for_each_map_elem, bpf_snprintf, bpf_timer_init, bpf_timer_set_callback, bpf_timer_start, bpf_timer_cancel, bpf_task_pt_regs, bpf_loop, bpf_strncmp, bpf_kptr_xchg, bpf_map_lookup_percpu_elem, bpf_dynptr_from_mem, bpf_ringbuf_reserve_dynptr, bpf_ringbuf_submit_dynptr, bpf_ringbuf_discard_dynptr, bpf_dynptr_read, bpf_dynptr_write, bpf_dynptr_data, bpf_ktime_get_tai_ns, bpf_user_ringbuf_drain, bpf_cgrp_storage_get, bpf_cgrp_storage_delete

Program type	Available helpers
flow_dissector	bpf_map_lookup_elem, bpf_map_update_elem, bpf_map_delete_elem, bpf_ktime_get_ns, bpf_get_prandom_u32, bpf_get_smp_processor_id, bpf_tail_call, bpf_skb_load_bytes, bpf_get_current_task, bpf_get_numa_node_id, bpf_get_current_cgroup_id, bpf_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_spin_lock, bpf_spin_unlock, bpf_strtol, bpf_strtoul, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_jiffies64, bpf_get_current_ancestor_cgroup_id, bpf_ktime_get_boot_ns, bpf_ringbuf_output, bpf_ringbuf_reserve, bpf_ringbuf_submit, bpf_ringbuf_discard, bpf_ringbuf_query, bpf_skc_to_tcp6_sock, bpf_skc_to_tcp_sock, bpf_skc_to_tcp_timewait_sock, bpf_skc_to_tcp_request_sock, bpf_skc_to_udp6_sock, bpf_snprintf_btf, bpf_per_cpu_ptr, bpf_this_cpu_ptr, bpf_get_current_task_btf, bpf_ktime_get_coarse_ns, bpf_for_each_map_elem, bpf_snprintf, bpf_timer_init, bpf_timer_set_callback, bpf_timer_start, bpf_timer_cancel, bpf_task_pt_regs, bpf_skc_to_unix_sock, bpf_loop, bpf_strncmp, bpf_kptr_xchg, bpf_map_lookup_percpu_elem, bpf_skc_to_mptcp_sock, bpf_dynptr_from_mem, bpf_ringbuf_reserve_dynptr, bpf_ringbuf_submit_dynptr, bpf_ringbuf_discard_dynptr, bpf_dynptr_read, bpf_dynptr_write, bpf_dynptr_data, bpf_ktime_get_tai_ns, bpf_user_ringbuf_drain, bpf_cgrp_storage_get, bpf_cgrp_storage_delete
cgroup_sysctl	bpf_map_lookup_elem, bpf_map_update_elem, bpf_map_delete_elem, bpf_ktime_get_ns, bpf_get_prandom_u32, bpf_get_smp_processor_id, bpf_tail_call, bpf_get_current_pid_tgid, bpf_get_current_uid_gid, bpf_get_current_comm, bpf_get_cgroup_classid, bpf_perf_event_output, bpf_get_current_task, bpf_get_numa_node_id, bpf_get_current_cgroup_id, bpf_get_local_storage, bpf_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_spin_lock, bpf_spin_unlock, bpf_sysctl_get_name, bpf_sysctl_get_current_value, bpf_sysctl_get_new_value, bpf_sysctl_set_new_value, bpf_strtol, bpf_strtoul, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_jiffies64, bpf_get_current_ancestor_cgroup_id, bpf_ktime_get_boot_ns, bpf_ringbuf_output, bpf_ringbuf_reserve, bpf_ringbuf_submit, bpf_ringbuf_discard, bpf_ringbuf_query, bpf_snprintf_btf, bpf_per_cpu_ptr, bpf_this_cpu_ptr, bpf_get_current_task_btf, bpf_ktime_get_coarse_ns, bpf_for_each_map_elem, bpf_snprintf, bpf_timer_init, bpf_timer_set_callback, bpf_timer_start, bpf_timer_cancel, bpf_task_pt_regs, bpf_loop, bpf_strncmp, bpf_kptr_xchg, bpf_map_lookup_percpu_elem, bpf_dynptr_from_mem, bpf_ringbuf_reserve_dynptr, bpf_ringbuf_submit_dynptr, bpf_ringbuf_discard_dynptr, bpf_dynptr_read, bpf_dynptr_write, bpf_dynptr_data, bpf_ktime_get_tai_ns, bpf_user_ringbuf_drain, bpf_cgrp_storage_get, bpf_cgrp_storage_delete

Program type	Available helpers
raw_tracepoint_writable	bpf_map_lookup_elem, bpf_map_update_elem, bpf_map_delete_elem, bpf_probe_read, bpf_ktime_get_ns, bpf_get_prandom_u32, bpf_get_smp_processor_id, bpf_tail_call, bpf_get_current_pid_tgid, bpf_get_current_uid_gid, bpf_get_current_comm, bpf_perf_event_read, bpf_perf_event_output, bpf_get_stackid, bpf_get_current_task, bpf_current_task_under_cgroup, bpf_get_numa_node_id, bpf_probe_read_str, bpf_perf_event_read_value, bpf_get_stack, bpf_get_current_cgroup_id, bpf_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_spin_lock, bpf_spin_unlock, bpf_strtol, bpf_strtoul, bpf_send_signal, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_send_signal_thread, bpf_jiffies64, bpf_get_ns_current_pid_tgid, bpf_get_current_ancestor_cgroup_id, bpf_ktime_get_boot_ns, bpf_ringbuf_output, bpf_ringbuf_reserve, bpf_ringbuf_submit, bpf_ringbuf_discard, bpf_ringbuf_query, bpf_get_task_stack, bpf_copy_from_user, bpf_snprintf_btf, bpf_per_cpu_ptr, bpf_this_cpu_ptr, bpf_task_storage_get, bpf_task_storage_delete, bpf_get_current_task_btf, bpf_for_each_map_elem, bpf_snprintf, bpf_timer_init, bpf_timer_set_callback, bpf_timer_start, bpf_timer_cancel, bpf_get_func_ip, bpf_task_pt_regs, bpf_get_branch_snapshot, bpf_find_vma, bpf_loop, bpf_strncmp, bpf_copy_from_user_task, bpf_kptr_xchg, bpf_map_lookup_percpu_elem, bpf_dynptr_from_mem, bpf_ringbuf_reserve_dynptr, bpf_ringbuf_submit_dynptr, bpf_ringbuf_discard_dynptr, bpf_dynptr_read, bpf_dynptr_write, bpf_dynptr_data, bpf_ktime_get_tai_ns, bpf_user_ringbuf_drain, bpf_cgrp_storage_get, bpf_cgrp_storage_delete
cgroup_sockopt	bpf_map_lookup_elem, bpf_map_update_elem, bpf_map_delete_elem, bpf_ktime_get_ns, bpf_get_prandom_u32, bpf_get_smp_processor_id, bpf_tail_call, bpf_get_current_pid_tgid, bpf_get_current_uid_gid, bpf_get_current_comm, bpf_get_cgroup_classid, bpf_perf_event_output, bpf_get_current_task, bpf_get_numa_node_id, bpf_get_current_cgroup_id, bpf_get_local_storage, bpf_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_spin_lock, bpf_spin_unlock, bpf_tcp_sock, bpf_strtol, bpf_strtoul, bpf_sk_storage_get, bpf_sk_storage_delete, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_jiffies64, bpf_get_netns_cookie, bpf_get_current_ancestor_cgroup_id, bpf_ktime_get_boot_ns, bpf_ringbuf_output, bpf_ringbuf_reserve, bpf_ringbuf_submit, bpf_ringbuf_discard, bpf_ringbuf_query, bpf_snprintf_btf, bpf_per_cpu_ptr, bpf_this_cpu_ptr, bpf_get_current_task_btf, bpf_for_each_map_elem, bpf_snprintf, bpf_timer_init, bpf_timer_set_callback, bpf_timer_start, bpf_timer_cancel, bpf_task_pt_regs, bpf_loop, bpf_strncmp, bpf_get_retval, bpf_set_retval, bpf_kptr_xchg, bpf_map_lookup_percpu_elem, bpf_dynptr_from_mem, bpf_ringbuf_reserve_dynptr, bpf_ringbuf_submit_dynptr, bpf_ringbuf_discard_dynptr, bpf_dynptr_read, bpf_dynptr_write, bpf_dynptr_data, bpf_ktime_get_tai_ns, bpf_user_ringbuf_drain, bpf_cgrp_storage_get, bpf_cgrp_storage_delete
tracing	
struct_ops	
ext	
lsm	

Program type	Available helpers
sk_lookup	bpf_map_lookup_elem, bpf_map_update_elem, bpf_map_delete_elem, bpf_ktime_get_ns, bpf_get_prandom_u32, bpf_get_smp_processor_id, bpf_tail_call, bpf_perf_event_output, bpf_get_current_task, bpf_get_numa_node_id, bpf_get_current_cgroup_id, bpf_sk_release, bpf_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_spin_lock, bpf_spin_unlock, bpf_strtol, bpf_strtoul, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_jiffies64, bpf_get_current_ancestor_cgroup_id, bpf_sk_assign, bpf_ktime_get_boot_ns, bpf_ringbuf_output, bpf_ringbuf_reserve, bpf_ringbuf_submit, bpf_ringbuf_discard, bpf_ringbuf_query, bpf_skc_to_tcp6_sock, bpf_skc_to_tcp_sock, bpf_skc_to_tcp_timewait_sock, bpf_skc_to_tcp_request_sock, bpf_skc_to_udp6_sock, bpf_snprintf_btf, bpf_per_cpu_ptr, bpf_this_cpu_ptr, bpf_get_current_task_btf, bpf_ktime_get_coarse_ns, bpf_for_each_map_elem, bpf_snprintf, bpf_timer_init, bpf_timer_set_callback, bpf_timer_start, bpf_timer_cancel, bpf_task_pt_regs, bpf_skc_to_unix_sock, bpf_loop, bpf_strncmp, bpf_kptr_xchg, bpf_map_lookup_percpu_elem, bpf_skc_to_mptcp_sock, bpf_dynptr_from_mem, bpf_ringbuf_reserve_dynptr, bpf_ringbuf_submit_dynptr, bpf_ringbuf_discard_dynptr, bpf_dynptr_read, bpf_dynptr_write, bpf_dynptr_data, bpf_ktime_get_tai_ns, bpf_user_ringbuf_drain, bpf_cgrp_storage_get, bpf_cgrp_storage_delete
syscall	bpf_map_lookup_elem, bpf_map_update_elem, bpf_map_delete_elem, bpf_probe_read, bpf_ktime_get_ns, bpf_get_prandom_u32, bpf_get_smp_processor_id, bpf_tail_call, bpf_get_current_pid_tgid, bpf_get_current_uid_gid, bpf_get_current_comm, bpf_perf_event_read, bpf_perf_event_output, bpf_get_stackid, bpf_get_current_task, bpf_current_task_under_cgroup, bpf_get_numa_node_id, bpf_probe_read_str, bpf_get_socket_cookie, bpf_perf_event_read_value, bpf_get_stack, bpf_get_current_cgroup_id, bpf_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_spin_lock, bpf_spin_unlock, bpf_strtol, bpf_strtoul, bpf_sk_storage_get, bpf_sk_storage_delete, bpf_send_signal, bpf_skb_output, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_send_signal_thread, bpf_jiffies64, bpf_get_ns_current_pid_tgid, bpf_xdp_output, bpf_get_current_ancestor_cgroup_id, bpf_ktime_get_boot_ns, bpf_ringbuf_output, bpf_ringbuf_reserve, bpf_ringbuf_submit, bpf_ringbuf_discard, bpf_ringbuf_query, bpf_skc_to_tcp6_sock, bpf_skc_to_tcp_sock, bpf_skc_to_tcp_timewait_sock, bpf_skc_to_tcp_request_sock, bpf_skc_to_udp6_sock, bpf_get_task_stack, bpf_d_path, bpf_copy_from_user, bpf_snprintf_btf, bpf_per_cpu_ptr, bpf_this_cpu_ptr, bpf_task_storage_get, bpf_task_storage_delete, bpf_get_current_task_btf, bpf_sock_from_file, bpf_for_each_map_elem, bpf_snprintf, bpf_sys_bpf, bpf_btf_find_by_name_kind, bpf_sys_close, bpf_timer_init, bpf_timer_set_callback, bpf_timer_start, bpf_timer_cancel, bpf_get_func_ip, bpf_task_pt_regs, bpf_get_branch_snapshot, bpf_skc_to_unix_sock, bpf_kallsyms_lookup_name, bpf_find_vma, bpf_loop, bpf_strncmp, bpf_xdp_get_buff_len, bpf_copy_from_user_task, bpf_kptr_xchg, bpf_map_lookup_percpu_elem, bpf_skc_to_mptcp_sock, bpf_dynptr_from_mem, bpf_ringbuf_reserve_dynptr, bpf_ringbuf_submit_dynptr, bpf_ringbuf_discard_dynptr, bpf_dynptr_read, bpf_dynptr_write, bpf_dynptr_data, bpf_ktime_get_tai_ns, bpf_user_ringbuf_drain, bpf_cgrp_storage_get, bpf_cgrp_storage_delete

Program type	Available helpers
netfilter	bpf_map_lookup_elem, bpf_map_update_elem, bpf_map_delete_elem, bpf_ktime_get_ns, bpf_get_prandom_u32, bpf_get_smp_processor_id, bpf_tail_call, bpf_get_current_task, bpf_get_numa_node_id, bpf_get_current_cgroup_id, bpf_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_spin_lock, bpf_spin_unlock, bpf_strtol, bpf_strtoul, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_jiffies64, bpf_get_current_ancestor_cgroup_id, bpf_ktime_get_boot_ns, bpf_ringbuf_output, bpf_ringbuf_reserve, bpf_ringbuf_submit, bpf_ringbuf_discard, bpf_ringbuf_query, bpf_snprintf_btf, bpf_per_cpu_ptr, bpf_this_cpu_ptr, bpf_get_current_task_btf, bpf_for_each_map_elem, bpf_snprintf, bpf_timer_init, bpf_timer_set_callback, bpf_timer_start, bpf_timer_cancel, bpf_task_pt_regs, bpf_loop, bpf_strncmp, bpf_kptr_xchg, bpf_map_lookup_percpu_elem, bpf_dynptr_from_mem, bpf_ringbuf_reserve_dynptr, bpf_ringbuf_submit_dynptr, bpf_ringbuf_discard_dynptr, bpf_dynptr_read, bpf_dynptr_write, bpf_dynptr_data, bpf_ktime_get_tai_ns, bpf_user_ringbuf_drain, bpf_cgrp_storage_get, bpf_cgrp_storage_delete

Table 12.3. Available map types

Map type	Available
hash	yes
array	yes
prog_array	yes
perf_event_array	yes
percpu_hash	yes
percpu_array	yes
stack_trace	yes
cgroup_array	yes
lru_hash	yes
lru_percpu_hash	yes
lpm_trie	yes
array_of_maps	yes
hash_of_maps	yes

Map type	Available
devmap	yes
sockmap	yes
cpumap	yes
xskmap	yes
sockhash	yes
cgroup_storage	yes
reuseport_sockarray	yes
percpu_cgroup_storage	yes
queue	yes
stack	yes
sk_storage	yes
devmap_hash	yes
struct_ops	yes
ringbuf	yes
inode_storage	yes
task_storage	yes
bloom_filter	yes
user_ringbuf	yes
cgrp_storage	yes
arena_map	yes

APPENDIX A. LIST OF TICKETS BY COMPONENT

Bugzilla and JIRA tickets are listed in this document for reference. The links lead to the release notes in this document that describe the tickets.

Component	Tickets
389-ds-base	Jira:RHEL-67196, Jira:RHEL-67595, Jira:RHEL-1681, Jira:RHEL-42485, Jira:RHEL-76841, Jira:RHEL-74164, Jira:RHEL-79498, Jira:RHEL-76020, Jira:RHEL-69819, Jira:RHEL-59513, Jira:RHEL-30640, Jira:RHEL-25071
NetworkManager	Jira:RHEL-64719, Jira:RHEL-46211
NetworkManager-libreswan	Jira:RHEL-58812
Release Notes	Jira:RHELDOCS-18787, Jira:RHELDOCS-19988, Jira:RHELDOCS-19185, Jira:RHELDOCS-19191, Jira:RHELDOCS-19863, Jira:RHELDOCS-19162, Jira:RHELDOCS-19060, Jira:RHELDOCS-19579, Jira:RHELDOCS-19411, Jira:RHELDOCS-19059, Jira:RHELDOCS-20102, Jira:RHELDOCS-19550, Jira:RHELDOCS-19840, Jira:RHELDOCS-19842, Jira:RHELDOCS-19877, Jira:RHELDOCS-19661, Jira:RHELDOCS-20066, Jira:RHELDOCS-20166, Jira:RHELDOCS-20168, Jira:RHELDOCS-20169, Jira:RHELDOCS-20170, Jira:RHELDOCS-19072, Jira:RHELDOCS-19891, Jira:RHELDOCS-19968, Jira:RHELDOCS-20041, Jira:RHELDOCS-20042, Jira:RHELDOCS-20043, Jira:RHELDOCS-19635, Jira:RHELDOCS-19009, Jira:RHELDOCS-19010, Jira:RHELDOCS-19071, Jira:RHELDOCS-19357, Jira:RHELDOCS-19066, Jira:RHELDOCS-18389, Jira:RHELDOCS-18390, Jira:RHELDOCS-19148, Jira:RHELDOCS-19125, Jira:RHELDOCS-19132, Jira:RHELDOCS-19170, Jira:RHELDOCS-19146, Jira:RHELDOCS-19138, Jira:RHELDOCS-19134, Jira:RHELDOCS-19136, Jira:RHELDOCS-19153, Jira:RHELDOCS-19142, Jira:RHELDOCS-19152, Jira:RHELDOCS-19155, Jira:RHELDOCS-19150, Jira:RHELDOCS-17682, Jira:RHELDOCS-19221, Jira:RHELDOCS-19222, Jira:RHELDOCS-19023, Jira:RHELDOCS-19813, Jira:RHELDOCS-19024, Jira:RHELDOCS-19094, Jira:RHELDOCS-18839, Jira:RHELDOCS-18492, Jira:RHELDOCS-18485, Jira:RHELDOCS-18414, Jira:RHELDOCS-18159, Jira:RHELDOCS-19051, Jira:RHELDOCS-18989, Jira:RHELDOCS-19084, Jira:RHELDOCS-19083, Jira:RHELDOCS-19269, Jira:RHELDOCS-19140, Jira:RHELDOCS-19828, Jira:RHELDOCS-20167, Jira:RHELDOCS-18080, Jira:RHELDOCS-19607, Jira:RHELDOCS-18674, Jira:RHELDOCS-18672, Jira:RHELDOCS-18450, Jira:RHELDOCS-20147, Jira:RHELDOCS-16612, Jira:RHELDOCS-19015, Jira:RHELDOCS-19172, Jira:RHELDOCS-19603, Jira:RHELDOCS-19016, Jira:RHELDOCS-19770
WALinuxAgent	Jira:RHEL-68796

Component	Tickets
anaconda	Jira:RHEL-61434, Jira:RHEL-38407, Jira:RHEL-56141, Jira:RHEL-33892, Jira:RHEL-80672, Jira:RHEL-67865, Jira:RHEL-74504, Jira:RHEL-83577, Jira:RHEL-66155, Jira:RHEL-58827, Jira:RHEL-58829, Jira:RHEL-58828, Jira:RHEL-58834
annobin	Jira:RHEL-526
ansible-collection-microsoft-sql	Jira:RHEL-68468, Jira:RHEL-68490, Jira:RHEL-69315
ansible-core	Jira:RHEL-86829
ansible-freeipa	Jira:RHEL-67567
audit	Jira:RHEL-5199
bind-dyndb-ldap	Jira:RHEL-30556
binutils	Jira:RHEL-56896, Jira:RHEL-36305
bootc-image-builder-container	Jira:RHEL-34807
ca-certificates	Jira:RHEL-50293
certmonger	Jira:RHEL-40922
clevis	Jira:RHEL-60113
cloud-init	Jira:RHEL-29720, Jira:RHEL-65849, Jira:RHEL-82209, Jira:RHEL-82210
cmake	Jira:RHEL-65234
cockpit	Jira:RHEL-4032
cockpit-machines	Jira:RHEL-31993
container-tools	Jira:RHEL-33571, Jira:RHEL-33573, Jira:RHEL-67260, Jira:RHEL-66762, Jira:RHEL-32724, Jira:RHEL-67064, Jira:RHEL-67063, Jira:RHEL-67860
coreutils	Jira:RHEL-74146
crash	Jira:RHEL-52221

Component	Tickets
crypto-policies	Jira:RHEL-50655, Jira:RHEL-76526, Jira:RHEL-58241, Jira:RHEL-65652, Jira:RHEL-50464, Jira:RHEL-50106, Jira:RHEL-64746
cryptsetup	Jira:RHEL-33395
cups	Jira:RHEL-68415
debugedit	Jira:RHEL-64137
device-mapper-multipath	Jira:RHEL-49747, Jira:RHEL-73410
dhcp	Jira:RHEL-14710
distribution	Jira:RHEL-30799, Jira:RHEL-18157, Jira:RHEL-59006, Jira:RHEL-73770
dnf	Jira:RHEL-12355, Jira:RHEL-38831, Jira:RHEL-76849, Jira:RHEL-40382
dnf-plugins-core	Jira:RHEL-56137, Jira:RHEL-23706
dnsconfd	Jira:RHEL-34791
dotNET	Jira:RHELDOCS-20066
dyninst	Jira:RHEL-49597
edk2	Jira:RHELPLAN-69533, Jira:RHEL-66234, Jira:RHEL-68418
elfutils	Jira:RHEL-29197, Jira:RHEL-64046
firewalld	Jira:RHEL-65865
gcc	Jira:RHEL-45041, Jira:RHEL-33254, Jira:RHEL-24760, Jira:RHEL-65765, Jira:RHEL-24762, Jira:RHEL-36791
gdb	Jira:RHEL-33256
glibc	Jira:RHEL-25045, Jira:RHEL-25850, Jira:RHEL-25530
gnome-online-accounts	Jira:RHEL-40831

Component	Tickets
gnome-shell-extensions	Jira:RHEL-4137
gnutls	Jira:RHEL-69524, Jira:RHEL-42514, Jira:RHEL-59212, Jira:RHEL-58640
golang	Jira:RHEL-34260, Jira:RHEL-52486, Jira:RHEL-49036
grafana	Jira:RHEL-35761
grafana-pcp	Jira:RHEL-67043, Jira:RHEL-45646
greenboot	Jira:RHEL-80003
grub2	Jira:RHEL-15032, Jira:RHEL-4378
gssproxy	Jira:RHEL-71651
ipa	Jira:RHEL-56917, Jira:RHEL-57674, Jira:RHEL-4879, Jira:RHEL-46607, Jira:RHEL-63325, Jira:RHELPLAN-121751, Jira:RHEL-67912, Jira:RHEL-33818, Jira:RHEL-12154
iptables	Jira:RHEL-66725
ipxe	Jira:RHEL-37610
jose	Jira:RHEL-38084
kdump-utils	Jira:RHEL-63071, Jira:RHEL-50736, Jira:RHEL-29941
kea	Jira:RHEL-9306
kernel	Jira:RHELPLAN-99859, Jira:RHELPLAN-96004
kernel / Debugging-Tracing / kexec - kdump	Jira:RHEL-29272
kernel / Debugging-Tracing / rtla	Jira:RHEL-40744
kernel / File Systems / CIFS	Jira:RHEL-78152
kernel / File Systems / NFS	Jira:RHEL-74415
kernel / File Systems / XFS	Jira:RHEL-33653

Component	Tickets
kernel / Networking	Jira:RHEL-68401
kernel / Networking / NIC Drivers	Jira:RHEL-73034, Jira:RHEL-40070, Jira:RHEL-56981
kernel / Networking / eBPF	Jira:RHEL-51429
kernel / Other	Jira:RHEL-65347
kernel / Platform Enablement / NVMe	Jira:RHEL-78133, Jira:RHEL-85845
kernel / Security	Jira:RHEL-26170
kernel / Security / Other	Jira:RHEL-40283
kernel / Storage / Block Layer	Jira:RHEL-60811
kernel / Storage / Persistent Memory (NVDIMM)	Jira:RHEL-68504
kernel / Storage / Storage Drivers	Jira:RHEL-75491
kernel / Virtualization / ESXi	Jira:RHEL-41133
kernel / Virtualization / Hyper-V	Jira:RHEL-29919
kernel / Virtualization / KVM	Jira:RHEL-25204, Jira:RHEL-58218, Jira:RHEL-32892, Jira:RHEL-45585, Jira:RHEL-38957
kernel-rt / Core / Scheduler	Jira:RHEL-58211
keylime	Jira:RHEL-75794, Jira:RHEL-51279, Jira:RHEL-79831
keylime-agent-rust	Jira:RHEL-38409
krb5	Jira:RHEL-71881, Jira:RHEL-56070
ksh	Jira:RHEL-45981
libabigail	Jira:RHEL-64063

Component	Tickets
libcap	Jira:RHEL-31988, Jira:RHEL-33498
libkcapi	Jira:RHEL-50457
libnftnl	Jira:RHEL-66276
liboqs	Jira:RHEL-65426
librepo	Jira:RHEL-47106
libreswan	Jira:RHEL-52935, Jira:RHEL-74850, Jira:RHEL-51880, Jira:RHEL-81045
libslirp	Jira:RHEL-45147
libssh	Jira:RHEL-64319, Jira:RHEL-30437
llvm	Jira:RHEL-57456, Jira:RHEL-70325, Jira:RHEL-58900
lsscsi	Jira:RHEL-32144
mesa	Jira:RHEL-45898
mutter	Jira:RHEL-69291
mysql	Jira:RHEL-36050
nbdkit	Jira:RHEL-32748
net-snmp	Jira:RHEL-44478
nettle	Jira:RHEL-79116
nftables	Jira:RHEL-65346
nginx	Jira:RHEL-33742
nodejs	Jira:RHEL-35992
nss	Jira:RHEL-46839, Jira:RHEL-39732, Jira:RHEL-36299, Jira:RHEL-61291, Jira:RHEL-44995
opencryptoki	Jira:RHEL-58996

Component	Tickets
openldap	Jira:RHEL-71052, Jira:RHEL-68773, Jira:RHEL-68424
opensc	Jira:RHEL-71523, Jira:RHEL-73314
openscap	Jira:RHEL-88845
openssh	Jira:RHEL-60564, Jira:RHEL-37324, Jira:RHEL-62718, Jira:RHEL-45002
openssl	Jira:RHEL-54156, Jira:RHEL-40408, Jira:RHEL-36659, Jira:RHEL-39962, Jira:RHEL-45704
p11-kit	Jira:RHEL-46898, Jira:RHEL-64915
pacemaker	Jira:RHEL-39057, Jira:RHEL-56675, Jira:RHEL-7600, Jira:RHEL-40117, Jira:RHEL-62722
pcs	Jira:RHEL-35670, Jira:RHEL-36612, Jira:RHEL-38491, Jira:RHEL-38489, Jira:RHEL-38487, Jira:RHEL-23048, Jira:RHEL-21047, Jira:RHEL-38483, Jira:RHEL-61889, Jira:RHEL-49527, Jira:RHEL-12709, Jira:RHEL-38493, Jira:RHEL-38484, Jira:RHEL-38486, Jira:RHEL-38478, Jira:RHEL-38479, Jira:RHEL-34792, Jira:RHEL-33386, Jira:RHEL-61747, Jira:RHEL-55723, Jira:RHEL-29739, Jira:RHEL-49521
pkcs11-provider	Jira:RHEL-29672, Jira:RHEL-40124, Jira:RHEL-68621
podman	Jira:RHEL-34604, Jira:RHEL-33566, Jira:RHEL-34611, Jira:RHEL-34613, Jira:RHEL-34606, Jira:RHEL-40639, Jira:RHEL-40643, Jira:RHEL-52238, Jira:RHEL-52240, Jira:RHEL-24623, Jira:RHEL-52247, Jira:RHEL-32266, Jira:RHEL-70218, Jira:RHEL-89373, Jira:RHEL-40641
policycoreutils	Jira:RHEL-69451
polkit	Jira:RHEL-55287
postgresql	Jira:RHEL-35993
postgresql16	Jira:RHEL-62694
pykickstart	Jira:RHEL-34829
python-blivet	Jira:RHEL-45175, Jira:RHEL-52200, Jira:RHEL-45180, Jira:RHEL-82884

Component	Tickets
python-pyasn1	Jira:RHEL-67667
qemu-kvm	Jira:RHEL-68444, Jira:RHEL-23771, Jira:RHELPLAN-81033, Jira:RHELPLAN-75969, Jira:RHEL-58928, Jira:RHEL-87642, Jira:RHEL-88435, Jira:RHEL-67699, Jira:RHEL-66229
qemu-kvm / Devices / CPU Models	Jira:RHEL-28971
qemu-kvm / Devices / Machine Types	Jira:RHEL-57668
qemu-kvm / Live Migration	Jira:RHEL-64308
realtime-tests	Jira:RHEL-65488
rear	Jira:RHEL-72557, Jira:RHEL-46613
rhc	Jira:RHEL-65517
rhel-bootc-container	Jira:RHEL-34859
rhel-system-roles	Jira:RHEL-34893, Jira:RHEL-46219, Jira:RHEL-37551, Jira:RHEL-40798, Jira:RHEL-34884, Jira:RHEL-34890, Jira:RHEL-34891, Jira:RHEL-34892, Jira:RHEL-40181, Jira:RHEL-40797, Jira:RHEL-45718, Jira:RHEL-46855, Jira:RHEL-48230, Jira:RHEL-48609, Jira:RHEL-50288, Jira:RHEL-50289, Jira:RHEL-50291, Jira:RHEL-53901, Jira:RHEL-34828, Jira:RHEL-67419, Jira:RHEL-67420, Jira:RHEL-67411, Jira:RHEL-73441, Jira:RHEL-70554, Jira:RHEL-67417, Jira:RHEL-67416, Jira:RHEL-67415, Jira:RHEL-67413, Jira:RHEL-67286, Jira:RHEL-34887, Jira:RHEL-34886, Jira:RHEL-67418, Jira:RHEL-34881, Jira:RHEL-34888, Jira:RHEL-34889, Jira:RHEL-34895, Jira:RHEL-34907, Jira:RHEL-38456, Jira:RHEL-40759, Jira:RHEL-40760, Jira:RHEL-40795, Jira:RHEL-50104, Jira:RHEL-57100, Jira:RHEL-70536, Jira:RHEL-67412, Jira:RHEL-73443, Jira:RHEL-45944, Jira:RHEL-34879, Jira:RHEL-73442, Jira:RHEL-73439, Jira:RHEL-67421, Jira:RHEL-76504, Jira:RHEL-81963, Jira:RHEL-73440
rpm	Jira:RHEL-56363
rsyslog	Jira:RHEL-70110
rteval	Jira:RHEL-28059, Jira:RHEL-67424
rust	Jira:RHEL-59689

Component	Tickets
scap-security-guide	Jira:RHEL-74239
selinux-policy	Jira:RHEL-36094, Jira:RHEL-62355, Jira:RHEL-33844, Jira:RHEL-46893, Jira:RHEL-73505, Jira:RHEL-77808
setools	Jira:RHEL-29967
setroubleshoot	Jira:RHEL-68957
sg3_utils	Jira:RHEL-412
slapi-nis	Jira:RHEL-34186
sos	Jira:RHEL-24523, Jira:RHEL-30893, Jira:RHEL-67712, Jira:RHEL-35945, Jira:RHEL-22389
sssd	Jira:RHEL-50243, Jira:RHEL-68319
stunnel	Jira:RHEL-33749
subscription-manager	Jira:RHEL-78003
systemtap	Jira:RHEL-29529, Jira:RHEL-64042
tbb	Jira:RHEL-33633
tmux	Jira:RHEL-62152
traceroute	Jira:RHEL-58449
trustee-guest-components	Jira:RHEL-73770
tuned	Jira:RHEL-79913
valgrind	Jira:RHEL-29535, Jira:RHEL-64056
virt-manager / Common	Jira:RHEL-62960
virt-v2v	Jira:RHEL-37687, Jira:RHEL-36712
virtio-win	Jira:RHEL-1300

Component	Tickets
virtio-win / virtio-win-prewhql	Jira:RHEL-53962, Jira:RHEL-12118, Jira:RHEL-935
virtiofsd	Jira:RHEL-29027, Jira:RHEL-87161
wpa_supplicant	Jira:RHEL-59010, Jira:RHEL-33750
xdp-tools	Jira:RHEL-45730
zlib	Jira:RHEL-24058

Component	Tickets
other	Jira:RHELDOCS-18402, Jira:RHELDOCS-18869, Jira:RHELDOCS-20020, Jira:RHELDOCS-18761, Jira:RHELDOCS-18997, Jira:RHELDOCS-19415, Jira:RHELDOCS-19417, Jira:RHELDOCS-19988, Jira:RHELDOCS-20100, Jira:RHELDOCS-19185, Jira:RHELDOCS-19191, Jira:RHELDOCS-19936, Jira:RHELDOCS-19197, Jira:RHELDOCS-19263, Jira:RHELDOCS-20014, Jira:RHELDOCS-19863, Jira:RHEL-59102, Jira:RHELDOCS-18585, Jira:RHELDOCS-18398, Jira:RHELDOCS-18522, Jira:RHELDOCS-18769, Jira:RHELDOCS-19162, Jira:RHELDOCS-19405, Jira:RHELDOCS-18532, Jira:RHELDOCS-18880, Jira:RHELDOCS-18425, Jira:RHELDOCS-19579, Jira:RHELDOCS-18925, Jira:RHELDOCS-18776, Jira:RHELDOCS-16414, Jira:RHELDOCS-19411, Jira:RHELDOCS-16362, Jira:RHELDOCS-18819, Jira:RHELDOCS-19059, Jira:RHELDOCS-18472, Jira:RHELDOCS-19812, Jira:RHELDOCS-19842, Jira:RHELDOCS-19877, Jira:RHELDOCS-19832, Jira:RHELDOCS-20019, Jira:RHELDOCS-20023, Jira:RHELDOCS-20066, Jira:RHELDOCS-19210, Jira:RHELDOCS-19664, Jira:RHELDOCS-19516, Jira:RHELDOCS-19583, Jira:RHELDOCS-19825, Jira:RHELDOCS-19291, Jira:RHELDOCS-20116, Jira:RHELDOCS-20166, Jira:RHELDOCS-20168, Jira:RHELDOCS-20169, Jira:RHELDOCS-20170, Jira:RHELDOCS-18902, Jira:RHELDOCS-19106, Jira:RHELDOCS-18201, Jira:RHELDOCS-18770, Jira:RHELDOCS-17679, Jira:RHELDOCS-19876, Jira:RHELDOCS-16800, Jira:RHELDOCS-17465, Jira:RHELDOCS-18408, Jira:RHELDOCS-20058, Jira:RHELDOCS-19891, Jira:RHELDOCS-19968, Jira:RHELDOCS-20041, Jira:RHELDOCS-20042, Jira:RHELDOCS-20043, Jira:RHELDOCS-20080, Jira:RHEL-88550, Jira:RHELDOCS-19635, Jira:RHELDOCS-19009, Jira:RHELDOCS-19010, Jira:RHELDOCS-19071, Jira:RHELDOCS-19066, Jira:RHELDOCS-18388, Jira:RHELDOCS-18389, Jira:RHELDOCS-18390, Jira:RHELDOCS-17682, Jira:RHELDOCS-19221, Jira:RHELDOCS-19222, Jira:RHELDOCS-19005, Jira:RHELDOCS-19094, Jira:RHELDOCS-19267, Jira:RHELDOCS-18965, Jira:RHELDOCS-18839, Jira:RHELDOCS-18818, Jira:RHELDOCS-18736, Jira:RHELDOCS-18492, Jira:RHELDOCS-18485, Jira:RHELDOCS-18426, Jira:RHELDOCS-18423, Jira:RHELDOCS-18417, Jira:RHELDOCS-18416, Jira:RHELDOCS-18415, Jira:RHELDOCS-18413, Jira:RHELDOCS-18412, Jira:RHELDOCS-18411, Jira:RHELDOCS-18410, Jira:RHELDOCS-18414, Jira:RHELDOCS-19811, Jira:RHELDOCS-19051, Jira:RHELDOCS-18989, Jira:RHELDOCS-19828, Jira:RHELDOCS-19363, Jira:RHELDOCS-20094, Jira:RHELDOCS-20167, Jira:RHELDOCS-18700, Jira:RHELDOCS-18903, Jira:RHELDOCS-18904, Jira:RHELDOCS-18491, Jira:RHELDOCS-18672, Jira:RHELDOCS-18450, Jira:RHELPLAN-113995, Jira:RHELDOCS-20147, Jira:RHELDOCS-20283, Jira:RHELDOCS-16612, Jira:RHELDOCS-19015, Jira:RHELDOCS-19172, Jira:RHELDOCS-19603, Jira:RHELDOCS-18471, Jira:RHELDOCS-19770, Jira:RHELDOCS-19539, Jira:RHELDOCS-19734, Jira:RHELDOCS-19948, Jira:RHELDOCS-19496, Jira:RHELDOCS-19945

APPENDIX B. REVISION HISTORY

0.0-6

Tue 2 Sep 2025, Gabriela Fialová (gfialova@redhat.com)

- Added an Enhancement RHEL-86165 (Storage)
- Updated the Architectures section
- Updated a Removed Functionality RHELDOCS-19071 (Security)

0.0-5

Mon 25 Aug 2025, Marc Muehlfeld (mmuehlfeld@redhat.com)

- Added a Removed Functionality RHELDOCS-20862 (Networking)

0.0-4

Tue 12 Aug 2025, Gabriela Fialová (gfialova@redhat.com)

- Added a New Feature RHELDOCS-20591 (Compilers and development tools)

0.0-3

Wed 30 Jul 2025, Gabriela Fialová (gfialova@redhat.com)

- Added a Known Issue RHEL-42486 (Virtualization)
- Added a Known Issue RHEL-65655 (Storage)
- Updated an Enhancement RHEL-69524 (Security)

0.0-2

Tue 15 Jul 2025, Gabriela Fialová (gfialova@redhat.com)

- Added a New Feature RHEL-101075 (Compilers and development tools)
- Added a Deprecated Feature RHELDOCS-20610 (Core services)

0.0-1

Tue 01 Jul 2025, Marc Muehlfeld (mmuehlfeld@redhat.com)

- Added two Technology Previews RHEL-5852 and RHELDOCS-20472 (Networking)

0.0-0

Tue 20 May 2025, Gabriela Fialová (gfialova@redhat.com)

- Release of the Red Hat Enterprise Linux 10.0 Release Notes.