Red Hat Enterprise Linux 9

9.1 Release Notes

Release Notes for Red Hat Enterprise Linux 9.1

Last Updated: 2025-05-12

Red Hat Enterprise Linux 9 9.1 Release Notes

Release Notes for Red Hat Enterprise Linux 9.1

Legal Notice

Abstract

The Release Notes provide high-level coverage of the improvements and additions that have been implemented in Red Hat Enterprise Linux 9.1 and document known problems in this release, as well as notable bug fixes, Technology Previews, deprecated functionality, and other details. For information on how to install Red Hat Enterprise Linux, proceed to 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 9.1

Installer and image creation

Following are image builder key highlights in RHEL 9.1 GA:

- Image builder on-premise now supports:
  - Uploading images to GCP
  - Customizing the /**boot** partition
  - Pushing a container image directly to a registry
  - Users can now customize their blueprints during the image creation process.

For more information, see Section 4.1, “Installer and image creation” .

RHEL for Edge

Following are RHEL for Edge key highlights in RHEL 9.1-GA:

- RHEL for Edge now supports installing the services and have them running with the default configuration, by using the **fdo-admin** CLI utility

For more information, see Section 4.2, “RHEL for Edge” .

Security

RHEL 9.1 introduces **Keylime**, a remote machine attestation tool using the trusted platform module (TPM) technology. With Keylime, you can verify and continuously monitor the integrity of remote machines.

SELinux user-space packages have been upgraded to version 3.4. The most notable changes include:

- Improved relabeling performance through parallel relabeling
- Support for SHA-256 in the **semodule** tool
- New policy utilities in the **libsepol-utils** package

Changes in the system configuration and the **clevis-luks-systemd** subpackage enable the Clevis encryption client to unlock also LUKS-encrypted volumes that mount late in the boot process without using the **systemctl enable clevis-luks-askpass.path** command during the deployment process.

See New features - Security for more information.

Shells and command-line tools

RHEL 9.1 introduces a new package **xmlstarlet**. With **XMLStarlet**, you can parse, transform, query, validate, and edit XML files.

The following command-line tools have been updated in RHEL 9.1:

- **opencryptoki** to version 3.18.0
- **powerpc-utils** to version 1.3.10

- **libvpd** to version 2.2.9
- **lsvpd** to version 1.7.14
- **ppc64-diag** to version 2.7.8

For more information, see New Features - Shells and command-line tools

Infrastructure services

The following infrastructure services tools have been updated in RHEL 9.1:

- **chrony** to version 4.2
- **unbound** to version 1.16.2
- **frr** to version 8.2.2

For more information, see New Features - Infrastructure services .

Networking

NetworkManager supports migrating connection profiles from the deprecated **ifcfg** format to keyfile format.

NetworkManager now clearly indicates that WEP support is not available in RHEL 9.

The MultiPath TCP (MPTCP) code in the kernel has been updated from upstream Linux 5.19.

For further details, see New features - Networking .

Dynamic programming languages, web and database servers

Later versions of the following components are now available as new module streams:

- **PHP 8.1**
- **Ruby 3.1**
- **Node.js 18**

In addition, the **Apache HTTP Server** has been updated to version 2.4.53.

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

Compilers and development tools

Updated system toolchain

The following system toolchain components have been updated in RHEL 9.1:

- **GCC 11.2.1**
- **glibc 2.34**
- **binutils 2.35.2**

Updated performance tools and debuggers

The following performance tools and debuggers have been updated in RHEL 9.1:

- **GDB 10.2**
- **Valgrind 3.19**

- **SystemTap 4.7**
- **Dyninst 12.1.0**
- **elfutils 0.187**

Updated performance monitoring tools
The following performance monitoring tools have been updated in RHEL 9.1:

- **PCP 5.3.7**
- **Grafana 7.5.13**

Updated compiler toolsets
The following compiler toolsets have been updated in RHEL 9.1:

- **GCC Toolset 12**
- **LLVM Toolset 14.0.6**
- **Rust Toolset 1.62**
- **Go Toolset 1.18**

For detailed changes, see Section 4.14, "Compilers and development tools" .

Java implementations in RHEL 9
The RHEL 9 AppStream repository includes:

- The **java-17-openjdk** packages, which provide the OpenJDK 17 Java Runtime Environment and the OpenJDK 17 Java Software Development Kit.
- The **java-11-openjdk** packages, which provide the OpenJDK 11 Java Runtime Environment and the OpenJDK 11 Java Software Development Kit.
- The **java-1.8.0-openjdk** packages, which provide the OpenJDK 8 Java Runtime Environment and the OpenJDK 8 Java Software Development Kit.

For more information, see OpenJDK documentation.

Java tools
RHEL 9.1 introduces **Maven 3.8** as a new module stream.

See Section 4.14, "Compilers and development tools" for more information.

Identity Management
Identity Management (IdM) in RHEL 9.1 introduces a Technology Preview where you can delegate user authentication to external identity providers (IdPs) that support the OAuth 2 Device Authorization Grant flow. When these users authenticate with SSSD, and after they complete authentication and authorization at the external IdP, they receive RHEL IdM single sign-on capabilities with Kerberos tickets.

For more information, see Technology Previews - Identity Management

Red Hat Enterprise Linux system roles
Notable new features in 9.1 RHEL system roles:

- RHEL system roles are now available also in playbooks with fact gathering disabled.

- The **ha_cluster** role now supports SBD fencing, configuration of Corosync settings, and configuration of bundle resources.
- The **network** role now configures network settings for routing rules, supports network configuration using the **nmstate API**, and users can create connections with IPoIB capability.
- The **microsoft.sql.server** role has new variables, such as variables to control configuring a high availability cluster, to manage firewall ports automatically, or variables to search for **mssql_tls_cert** and **mssql_tls_private_key** values on managed nodes.
- The **logging** role supports various new options, for example **startmsg.regex** and **endmsg.regex** in files inputs, or **template**, **severity** and **facility** options.
- The **storage** role now includes support for thinly provisioned volumes, and the role now also has less verbosity by default.
- The **sshd** role verifies the include directive for the drop-in directory, and the role can now be managed through /etc/ssh/sshd_config.
- The **metrics** role can now export postfix performance data.
- The **postfix** role now has a new option for overwriting previous configuration.
- The **firewall** role does not require the state parameter when configuring masquerade or icmp_block_inversion. In the **firewall** role, you can now add, update, or remove services using absent and present states. The role can also provide Ansible facts, and add or remove an interface to the zone using PCI device ID. The **firewall** role has a new option for overwriting previous configuration.
- The **selinux** role now includes setting of **seuser** and **selevel** parameters.

1.2. IN-PLACE UPGRADE

In-place upgrade from RHEL 8 to RHEL 9

The supported in-place upgrade paths currently are:

- From RHEL 8.6 to RHEL 9.0 on the following architectures:
  - 64-bit Intel
  - 64-bit AMD
  - 64-bit ARM
  - IBM POWER 9 (little endian)
  - IBM Z architectures, excluding z13
- From RHEL 8.6 to RHEL 9.0 on systems with SAP HANA

To ensure your system remains supported after upgrading to RHEL 9.0, either update to the latest RHEL 9.1 version or enable the RHEL 9.0 Extended Update Support (EUS) repositories.

For instructions on performing an in-place upgrade, see Upgrading from RHEL 8 to RHEL 9 .

For instructions on performing an in-place upgrade on systems with SAP environments, see How to in-place upgrade SAP environments from RHEL 8 to RHEL 9.

Notable enhancements include:

- In-place upgrades on Microsoft Azure and Google Cloud Platform with Red Hat Update Infrastructure (RHUI) are now possible.
- The OpenSSH and OpenSSL configurations are now migrated during the in-place upgrade.

In-place upgrade from RHEL 7 to RHEL 9
It is not possible to perform an in-place upgrade directly from RHEL 7 to RHEL 9. However, you can perform an in-place upgrade from RHEL 7 to RHEL 8 and then perform a second in-place upgrade to RHEL 9. For more information, see Upgrading from RHEL 7 to RHEL 8 .

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
- VNC Configurator
- Red Hat Satellite Upgrade Helper
- JVM Options Configuration Tool
- Load Balancer Configuration Tool
- Ceph Placement Groups (PGs) per Pool Calculator
- Red Hat Out of Memory Analyzer

1.4. ADDITIONAL RESOURCES

Capabilities and limits of Red Hat Enterprise Linux 9 as compared to other versions of the system are available in the Knowledgebase article Red Hat Enterprise Linux technology capabilities and limits .

Information regarding the Red Hat Enterprise Linux **life cycle** is provided in the Red Hat Enterprise Linux Life Cycle document.

The Package manifest document provides a **package listing** for RHEL 9, including licenses and application compatibility levels.

Application compatibility levels are explained in the Red Hat Enterprise Linux 9: Application Compatibility Guide document.

Major **differences between RHEL 8 and RHEL 9**, including removed functionality, are documented in Considerations in adopting RHEL 9 .

Instructions on how to perform an **in-place upgrade from RHEL 8 to RHEL 9** are provided by the document Upgrading from RHEL 8 to RHEL 9 .

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.

CHAPTER 2. ARCHITECTURES

Red Hat Enterprise Linux 9.1 is distributed with the kernel version 5.14.0-162, which provides support for the following architectures at the minimum required version:

- AMD and Intel 64-bit architectures (x86-64-v2)
- The 64-bit ARM architecture (ARMv8.0-A)
- IBM Power Systems, Little Endian (POWER9)
- 64-bit IBM Z (z14)

Make sure you purchase the appropriate subscription for each architecture. For more information, see Get Started with Red Hat Enterprise Linux - additional architectures .

CHAPTER 3. DISTRIBUTION OF CONTENT IN RHEL 9

3.1. INSTALLATION

Red Hat Enterprise Linux 9 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 Image Builder tool to create customized RHEL images. For more information about 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.

See the Interactively installing RHEL from installation media document for instructions on downloading ISO images, creating installation media, and completing a RHEL installation. For automated Kickstart installations and other advanced topics, see the Automatically installing RHEL document.

For a list of users and groups created by RPMs in a base RHEL installation, and the steps to obtain this list, see the What are all of the users and groups in a base RHEL installation? Knowledgebase article.

3.2. REPOSITORIES

Red Hat Enterprise Linux 9 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 OS 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. For more information, see the Scope of Coverage Details document.

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 9 repositories and the packages they provide, see the Package manifest.

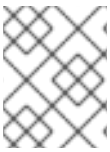
3.3. APPLICATION STREAMS

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 familiar RPM format, as an extension to the RPM format called modules, as Software Collections, or as Flatpaks.

Each Application Stream component has a given life cycle, either the same as RHEL 9 or shorter. For RHEL life cycle information, see Red Hat Enterprise Linux Life Cycle .

RHEL 9 improves the Application Streams experience by providing initial Application Stream versions that can be installed as RPM packages using the traditional **dnf install** command.



NOTE

Certain initial Application Streams in the RPM format have a shorter life cycle than Red Hat Enterprise Linux 9.

Some additional Application Stream versions will be distributed as modules with a shorter life cycle in future minor RHEL 9 releases. Modules are collections of packages representing a logical unit: an application, a language stack, a database, or a set of tools. These packages are built, tested, and released together.

Always determine what version of an Application Stream you want to install and make sure to review the Red Hat Enterprise Linux Application Stream Lifecycle first.

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. Rolling streams may be packaged as RPMs or modules.

For information about Application Streams available in RHEL 9 and their application compatibility level, see the Package manifest. Application compatibility levels are explained in the Red Hat Enterprise Linux 9: Application Compatibility Guide document.

3.4. PACKAGE MANAGEMENT WITH YUM/DNF

In Red Hat Enterprise Linux 9, software installation is ensured by **DNF**. Red Hat continues to support the usage of the **yum** term for consistency with previous major versions of RHEL. If you type **dnf** instead of **yum**, the command works as expected because both are aliases for compatibility.

Although RHEL 8 and RHEL 9 are based on **DNF**, they are compatible with **YUM** used in RHEL 7.

For more information, see Managing software with the DNF tool.

CHAPTER 4. NEW FEATURES

This part describes new features and major enhancements introduced in Red Hat Enterprise Linux 9.1.

4.1. INSTALLER AND IMAGE CREATION

Automatic FCP SCSI LUN scanning support in installer

The installer can now use the automatic LUN scanning when attaching FCP SCSI LUNs on IBM Z systems. Automatic LUN scanning is available for FCP devices operating in NPIV mode, if it is not disabled through the **zfcp.allow_lun_scan** kernel module parameter. It is enabled by default. It provides access to all SCSI devices found in the storage area network attached to the FCP device with the specified device bus ID. It is not necessary to specify WWPN and FCP LUNs anymore and it is sufficient to provide just the FCP device bus ID.

(BZ#1937031)

Image builder on-premise now supports the /boot partition customization

Image builder on-premise version now supports building images with custom **/boot** mount point partition size. You can specify the size of the **/boot** mount point partition in the blueprint customization, to increase the size of the **/boot** partition in case the default boot partition size is too small. For example:

```
[[customizations.filesystem]]
mountpoint = "/boot"
size = "20 GiB"
```

(JIRA:RHELPLAN-130379)

Added the --allow-ssh kickstart option to enable password-based SSH root logins

During the graphical installation, you have an option to enable password-based SSH root logins. This functionality was not available in kickstart installations. With this update, an option **--allow-ssh** has been added to the **rootpw** kickstart command. This option enables the root user to login to the system using SSH with a password.

(BZ#2083269)

Boot loader menu hidden by default

The GRUB boot loader is now configured to hide the boot menu by default. This results in a smoother boot experience. The boot menu is hidden in all of the following cases:

- When you restart the system from the desktop environment or the login screen.
- During the first system boot after the installation.
- When the **greenboot** package is installed and enabled.

If the previous system boot failed, GRUB always displays the boot menu during the next boot.

To access the boot menu manually, use either of the following options:

- Repeatedly press **Esc** during boot.
- Repeatedly press **F8** during boot.

- Hold **Shift** during boot.

To disable this feature and configure the boot loader menu to display by default, use the following command:

```
# grub2-editenv - unset menu_auto_hide
```

(BZ#2059414)

Minimal RHEL installation now installs only the s390utils-core package

In RHEL 8.4 and later, the **s390utils-base** package is split into an **s390utils-core** package and an auxiliary **s390utils-base** package. As a result, setting the RHEL installation to **minimal-environment** installs only the necessary **s390utils-core** package and not the auxiliary **s390utils-base** package. If you want to use the **s390utils-base** package with a minimal RHEL installation, you must manually install the package after completing the RHEL installation or explicitly install **s390utils-base** using a kickstart file.

(BZ#1932480)

Image builder on-premise now supports uploading images to GCP

With this enhancement, you can use image builder CLI to build a **gce** image, providing credentials for the user or service account that you want to use to upload the images. As a result, image builder creates the image and then uploads the **gce** image directly to the GCP environment that you specified.

(BZ#2049492)

Image builder on-premise CLI supports pushing a container image directly to a registry

With this enhancement, you can push RHEL for Edge container images directly to a container registry after it has been built, using the image builder CLI. To build the container image:

1. Set up an upload provider and optionally, add credentials.
2. Build the container image, passing the container registry and the repository to **composer-cli** as arguments.
   After the image is ready, it is available in the container registry you set up.

(JIRA:RHELPLAN-130376)

Image builder on-premise users now customize their blueprints during the image creation process

With this update, the **Edit Blueprint** page was removed to unify the user experience in the image builder service and in the image builder app in **cockpit-composer**. Users can now create their blueprints and add their customization, such as adding packages, and create users, during the image creation process. The versioning of blueprints has also been removed so that blueprints only have one version: the current one. Users have access to older blueprint versions through their already created images.

(JIRA:RHELPLAN-122735)

4.2. RHEL FOR EDGE

RHEL for Edge now supports the fdo-admin cli utility

With this update, you can configure the FDO services directly across all deployment scenarios by using the CLI.

Run the following commands to generate the certificates and keys for the services :

NOTE

This example takes into consideration that you already installed the **fdo-admin-cli** RPM package. If you used the source code and compiled it, the correct path is **./target/debug/fdo-admin-tool** or **./target/debug/fdo-admin-tool**, depending on your build options.

```
$ mkdir keys
$ for i in "diun" "manufacturer" "device_ca" "owner"; do fdo-admin-tool generate-key-and-cert $i; done
$ ls keys
device_ca_cert.pem  device_ca_key.der  diun_cert.pem  diun_key.der  manufacturer_cert.pem
manufacturer_key.der  owner_cert.pem  owner_key.der
```

As a result, after you install and start the service, it runs with the default settings.

(JIRA:RHELPLAN-122776)

4.3. SUBSCRIPTION MANAGEMENT

The subscription-manager utility displays the current status of actions

The **subscription-manager** utility now displays with progress information while it is processing the current operation. This is helpful when **subscription-manager** takes more than usual time to complete its operations related to server communication, for example, registration.

To revert to the previous behavior, enter:

```
# subscription-manager config --rhsm.progress_messages=0
```

(BZ#2092014)

4.4. SOFTWARE MANAGEMENT

The modulesync command is now available to replace certain workflows in RHEL 9

In RHEL 9, modular packages cannot be installed without modular metadata. Previously, you could use the **dnf** command to download packages, and then use the **createrepo_c** command to redistribute those packages.

This enhancement introduces the **modulesync** command to ensure the presence of modular metadata, which ensures package installability. This command downloads RPM packages from modules and creates a repository with modular metadata in a working directory.

(BZ#2066646)

4.5. SHELLS AND COMMAND-LINE TOOLS

Cronie adds support for a randomized time within a selected range

The **Cronie** utility now supports the ~ (random within range) operator for cronjob execution. As a result, you can start a cronjob on a randomized time within the selected range.

(BZ#2090691)

ReaR adds new variables for executing commands before and after recovery

With this enhancement, ReaR introduces two new variables for easier automation of commands to be executed before and after recovery:

- **PRE_RECOVERY_COMMANDS** accepts an array of commands. These commands will be executed before recovery starts.
- **POST_RECOVERY_COMMANDS** accepts an array of commands. These commands will be executed after recovery finishes.

These variables are an alternative to **PRE_RECOVERY_SCRIPT** and **POST_RECOVERY_SCRIPT** with the following differences:

- The earlier **PRE_RECOVERY_SCRIPT** and **POST_RECOVERY_SCRIPT** variables accept a single shell command. To pass multiple commands to these variables, you must separate the commands by semicolons.
- The new **PRE_RECOVERY_COMMANDS** and **POST_RECOVERY_COMMANDS** variables accept arrays of commands, and each element of the array is executed as a separate command.

As a result, providing multiple commands to be executed in the rescue system before and after recovery is now easier and less error-prone.

For more information, see the **default.conf** file.

(BZ#2111059)

A new package: xmlstarlet

XMLStarlet is a set of command-line utilities for parsing, transforming, querying, validating, and editing XML files. The new **xmlstarlet** package provides a simple set of shell commands that you can use in a similar way as you use UNIX commands for plain text files such as **grep**, **sed**, **awk**, **diff**, **patch**, **join**, and other.

(BZ#2069689)

opencryptoki rebased to version 3.18.0

The **opencryptoki** package, which is an implementation of the Public-Key Cryptography Standard (PKCS) #11, has been updated to version 3.18.0. Notable improvements include:

- Default to Federal Information Processing Standards (FIPS) compliant token data format (tokversion = 3.12).
- Added support for restricting usage of mechanisms and keys with a global policy.
- Added support for statistics counting of mechanism usage.
- The **ICA**/**EP11** tokens now support **libica** library version 4.
- The **p11sak** tool enables setting different attributes for public and private keys.
- The **C_GetMechanismList** does not return **CKR_BUFFER_TOO_SMALL** in the EP11 token.

openCryptoki supports two different token data formats:

- the earlier data format, which uses non-FIPS-approved algorithms (such as DES and SHA1)
- the new data format, which uses FIPS-approved algorithms only.

The earlier data format no longer works because the FIPS provider allows the use of only FIPS-approved algorithms.

IMPORTANT

To make openCryptoki work on RHEL 9, migrate the tokens to the new data format before enabling FIPS mode on the system. This is necessary because the earlier data format is still the default in **openCryptoki 3.17**. Existing **openCryptoki** installations that use the earlier token data format will no longer function when the system is changed to FIPS-enabled.

You can migrate the tokens to the new data format by using the **pkcstok_migrate** utility, which is provided with **openCryptoki**. Note that **pkcstok_migrate** uses non-FIPS-approved algorithms during the migration. Therefore, use this tool before enabling FIPS mode on the system. For additional information, see Migrating to FIPS compliance - pkcstok_migrate utility .

(BZ#2044179)

powerpc-utils rebased to version 1.3.10

The **powerpc-utils** package, which provides various utilities for a PowerPC platform, has been updated to version 1.3.10. Notable improvements include:

- Added the capability to parsing the Power architecture platform reference (PAPR) information for energy and frequency in the **ppc64_cpu** tool.
- Improved the **lparstat** utility to display enhanced error messages, when the **lparstat -E** command fails on max config systems. The **lparstat** command reports logical partition-related information.
- Fixed reported online memory in legacy format in the **lparstat** command.
- Added support for the **acc** command for changing the quality of service credits (QoS) dynamically for the NX GZIP accelerator.
- Added improvements to format specifiers in **printf()** and **sprintf()** calls.
- The **hcnmgr** utility, which provides the HMC tools to hybrid virtual network, includes following enhancements:
  - Added the **wicked** feature to the Hybrid Network Virtualization **HNV FEATURE** list. The **hcnmgr** utility supports wicked hybrid network virtualization (HNV) to use the **wicked** functions for bonding.
  - **hcnmgr** maintains an **hcnid** state for later cleanup.
  - **hcnmgr** excludes NetworkManager (NM) **nmcli** code.
  - The NM HNV **primary slave** setting was fixed.
  - **hcnmgr** supports the virtual Network Interface Controller (vNIC) as a backup device.
- Fixed the invalid hexadecimal numbering system message in **bootlist**.

- The **-l** flag included in **kpartx** utility as **-p** delimiter value in the **bootlist** command.
- Fixes added to **sslot** utility to prevent memory leak when listing IO slots.
- Added the DRC type description strings for the latest peripheral component interconnect express (PCIe) slot types in the **lsslot** utility.
- Fixed the invalid config address to RTAS in **errinjct** tool.
- Added support for non-volatile memory over fabrics (NVMf) devices in the **ofpathname** utility. The utility provides a mechanism for converting a logical device name to an open firmware device path and the other way round.
- Added fixes to the non-volatile memory (NVMe) support in asymmetric namespace access (ANA) mode in the **ofpathname** utility.
- Installed **smt.state** file as a configuration file.

(BZ#1920964)

The Redfish modules are now part of the redhat.rhel_mgmt Ansible collection

The **redhat.rhel_mgmt** Ansible collection now includes the following modules:

- **redfish_info**
- **redfish_command**
- **redfish_config**

With that, users can benefit from the management automation, by using the Redfish modules to retrieve server health status, get information about hardware and firmware inventory, perform power management, change BIOS settings, configure Out-Of-Band (OOB) controllers, configure hardware RAID, and perform firmware updates.

(BZ#2112434)

libvpd rebased to version 2.2.9

The **libvpd** package, which contains classes for accessing the Vital Product Data (VPD), has been updated to version 2.2.9. Notable improvements include:

- Fixed database locking
- Updated **libtool** utility version information

(BZ#2051288)

lsvpd rebased to version 1.7.14

The **lsvpd** package, which provides commands for constituting a hardware inventory system, has been updated to version 1.7.14. With this update, the **lsvpd** utility prevents corruption of the database file when you run the **vpdupdate** command.

(BZ#2051289)

ppc64-diag rebased to version 2.7.8

The **ppc64-diag** package for platform diagnostics has been updated to version 2.7.8. Notable improvements include:

- Updated build dependency to use **libvpd** utility version 2.2.9 or higher
- Fixed **extract_opal_dump** error message on unsupported platform
- Fixed build warning with **GCC-8.5** and **GCC-11** compilers

(BZ#2051286)

sysctl introduces identic syntax for arguments as systemd-sysctl

The **sysctl** utility from the **procps-ng** package, which you can use to modify kernel parameters at runtime, now uses the same syntax for arguments as the **systemd-sysctl** utility. With this update, **sysctl** now parses configuration files that contain hyphens (**-**) or globs (*****) on configuration lines. For more information about the **systemd-sysctl** syntax, see the **sysctl.d(5)** man page.

(BZ#2052536)

Updated systemd-udevd assigns consistent network device names to InfiniBand interfaces

Introduced in RHEL 9, the new version of the **systemd** package contains the updated **systemd-udevd** device manager. The device manager changes the default names of InfiniBand interfaces to consistent names selected by **systemd-udevd**.

You can define custom naming rules for naming InfiniBand interfaces by following the Renaming IPoIB devices procedure.

For more details of the naming scheme, see the **systemd.net-naming-scheme(7)** man page.

(BZ#2136937)

4.6. INFRASTRUCTURE SERVICES

chrony now uses DHCPv6 NTP servers

The NetworkManager dispatcher script for **chrony** updates the Network time protocol (NTP) sources passed from Dynamic Host Configuration Protocol (DHCP) options. Since RHEL 9.1, the script uses NTP servers provided by DHCPv6 in addition to DHCPv4. The DHCP option 56 specifies the usage of DHCPv6, the DHCP option 42 is DHCPv4-specific.

(BZ#2047415)

chrony rebased to version 4.2

The **chrony** suite has been updated to version 4.2. Notable enhancements over version 4.1 include:

- The server interleaved mode has been improved to be more reliable and supports multiple clients behind a single address translator (Network Address Translation - NAT).
- Experimental support for the Network Time Protocol Version 4 (NTPv4) extension field has been added to improve time synchronization stability and precision of estimated errors. You can enable this field, which extends the capabilities of the protocol NTPv4, by using the **extfield F323** option.
- Experimental support for NTP forwarding over the Precision Time Protocol (PTP) has been added to enable full hardware timestamping on Network Interface Cards (NIC) that have

timestamping limited to PTP packets. You can enable NTP over PTP by using the **ptpport 319** directive.

(BZ#2051441)

unbound rebased to version 1.16.2

The **unbound** component has been updated to version 1.16.2. **unbound** is a validating, recursive, and caching DNS resolver. Notable improvements include:

- With the ZONEMD Zone Verification with **RFC 8976** support, recipients can now verify the zone contents for data integrity and origin authenticity.
- With **unbound**, you can now configure persistent TCP connections.
- The SVCB and HTTPS types and handling according to the Service binding and parameter specification through the DNS **draft-ietf-dnsop-svcb-https** document were added.
- **unbound** takes the default TLS ciphers from crypto policies.
- You can use a Special-Use Domain **home.arpa.** according to the **RFC8375**. This domain is designated for non-unique use in residential home networks.
- **unbound** now supports selective enabling of **tcp-upstream** queries for stub or forward zones.
- The default of **aggressive-nsec** option is now **yes**.
- The **ratelimit** logic was updated.
- You can use a new **rpz-signal-nxdomain-ra** option for unsetting the **RA** flag when a query is blocked by an Unbound response policy zone (RPZ) nxdomain reply.
- With the basic support for Extended DNS Errors (EDE) according to the **RFC8914**, you can benefit from additional error information.

(BZ#2087120)

The password encryption function is now available in whois

The **whois** package now provides the **/usr/bin/mkpasswd** binary, which you can use to encrypt a password with the **crypt** C library interface.

(BZ#2054043)

frr rebased to version 8.2.2

The **frr** package for managing dynamic routing stack has been updated to version 8.2.2. Notable changes and enhancements over version 8.0 include:

- Added Ethernet VPN (EVPN) route type-5 gateway IP Overlay Index.
- Added Autonomous system border router (ASBR) summarization in the Open-shortest-path-first (OSPFv3) protocol.
- Improved usage of stub and not-so-stubby-areas (NSSA) in OSPFv3.
- Added the graceful restart capability in OSPFv2 and OSPFv3.

- The link bandwidth in the border gateway protocol (BGP) is now encoded according to the IEEE 754 standard. To use the previous encoding method, run the **neighbor PEER disable-link-bw-encoding-ieee** command in the existing configuration.
- Added the long-lived graceful restart capability in BGP.
- Implemented the extended administrative shutdown communication **rfc9003**, and the extended optional parameters length **rfc9072** in BGP.

(BZ#2069563)

TuneD real-time profiles now auto determine initial CPU isolation setup

TuneD is a service for monitoring your system and optimizing the performance profile. You can also isolate central processing units (CPUs) using the **tuned-profiles-realtime** package to give application threads the most execution time possible.

Previously, the real-time profiles for systems running the real-time kernel did not load if you did not specify the list of CPUs to isolate in the **isolated_cores** parameter.

With this enhancement, TuneD introduces the **calc_isolated_cores** built-in function that automatically calculates housekeeping and isolated cores lists, and applies the calculation to the **isolated_cores** parameter. With the automatic preset, one core from each socket is reserved for housekeeping, and you can start using the real-time profile without any additional steps. If you want to change the preset, customize the **isolated_cores** parameter by specifying the list of CPUs to isolate.

(BZ#2093847)

4.7. SECURITY

New packages: keylime

RHEL 9.1 introduces Keylime, a tool for attestation of remote systems, which uses the trusted platform module (TPM) technology. With Keylime, you can verify and continuously monitor the integrity of remote systems. You can also specify encrypted payloads that Keylime delivers to the monitored machines, and define automated actions that trigger whenever a system fails the integrity test.

See Ensuring system integrity with Keylime in the RHEL 9 Security hardening document for more information.

(JIRA:RHELPLAN-92522)

New option in OpenSSH supports setting the minimum RSA key length

Accidentally using short RSA keys makes the system more vulnerable to attacks. With this update, you can set minimum RSA key lengths for OpenSSH servers and clients. To define the minimum RSA key length, use the new **RequiredRSASize** option in the **/etc/ssh/sshd_config** file for OpenSSH servers, and in the **/etc/ssh/ssh_config** file for OpenSSH clients.

(BZ#2066882)

crypto-policies enforce 2048-bit RSA key length minimum for OpenSSH by default

Using short RSA keys makes the system more vulnerable to attacks. Because OpenSSH now supports limiting minimum RSA key length, the system-wide cryptographic policies enforce the 2048-bit minimum key length for RSA by default.

If you encounter OpenSSH failing connections with an **Invalid key length** error message, start using longer RSA keys.

Alternatively, you can relax the restriction by using a custom subpolicy at the expense of security. For example, if the **update-crypto-policies --show** command reports that the current policy is **DEFAULT**:

1. Define a custom subpolicy by inserting the **min_rsa_size@openssh = 1024** parameter into the **/etc/crypto-policies/policies/modules/RSA-OPENSSH-1024.pmod** file.
2. Apply the custom subpolicy using the **update-crypto-policies --set DEFAULT:RSA-OPENSSH-1024** command.

(BZ#2102774)

New option in OpenSSL supports SHA-1 for signatures

OpenSSL 3.0.0 in RHEL 9 does not support SHA-1 for signature creation and verification by default (SHA-1 key derivation functions (KDF) and hash-based message authentication codes (HMAC) are still supported). However, to support backwards compatibility with RHEL 8 systems that still use SHA-1 for signatures, a new configuration option **rh-allow-sha1-signatures** is introduced to RHEL 9. This option, if enabled in the **alg_section** of **openssl.cnf**, permits the creation and verification of SHA-1 signatures.

This option is automatically enabled if the LEGACY system-wide cryptographic policy (not legacy provider) is set.

Note that this also affects the installation of RPM packages with SHA-1 signatures, which may require switching to the LEGACY system-wide cryptographic policy.

(BZ#2060510, BZ#2055796)

crypto-policies now support sntrup761x25519-sha512@openssh.com

This update of the system-wide cryptographic policies adds support for the **sntrup761x25519-sha512@openssh.com** key exchange (KEX) method. The post-quantum **sntrup761** algorithm is already available in the OpenSSH suite, and this method provides better security against attacks from quantum computers. To enable **sntrup761x25519-sha512@openssh.com**, create and apply a subpolicy, for example:

```
# echo 'key_exchange = +SNTRUP' > /etc/crypto-policies/policies/modules/SNTRUP.pmod
# update-crypto-policies --set DEFAULT:SNTRUP
```

For more information, see the Customizing system-wide cryptographic policies with subpolicies section in the RHEL 9 Security hardening document.

(BZ#2070604)

NSS no longer support RSA keys shorter than 1023 bits

The update of the Network Security Services (NSS) libraries changes the minimum key size for all RSA operations from 128 to 1023 bits. This means that NSS no longer perform the following functions:

- Generate RSA keys shorter than 1023 bits.
- Sign or verify RSA signatures with RSA keys shorter than 1023 bits.
- Encrypt or decrypt values with RSA key shorter than 1023 bits.

(BZ#2091905)

SELinux policy confines additional services

The **selinux-policy** packages have been updated, and therefore the following services are now confined by SELinux:

- **ksm**
- **nm-priv-helper**
- **rhcd**
- **stalld**
- **systemd-network-generator**
- **targetclid**
- **wg-quick**

(BZ#1965013, BZ#1964862, BZ#2020169, BZ#2021131, BZ#2042614, BZ#2053639, BZ#2111069)

SELinux supports the self keyword in type transitions

SELinux tooling now supports type transition rules with the **self** keyword in the policy sources. Support for type transitions with the **self** keyword prepares the SELinux policy for labeling of anonymous inodes.

(BZ#2069718)

SELinux user-space packages updated

SELinux user-space packages **libsepol**, **libselinux**, **libsemanage**, **policycoreutils**, **checkpolicy**, and **mcstrans** were updated to the latest upstream release 3.4. The most notable changes are:

- Added support for parallel relabeling through the **-T** option in the **setfiles**, **restorecon**, and **fixfiles** tools.
  - You can either specify the number of process threads in this option or use **-T 0** for using the maximum of available processor cores. This reduces the time required for relabeling significantly.
- Added the new **--checksum** option, which prints SHA-256 hashes of modules.
- Added new policy utilities in the **libsepol-utils** package.

(BZ#2079276)

SELinux automatic relabeling is now parallel by default

Because the newly introduced parallel relabeling option significantly reduces the time required for the SELinux relabeling process on multi-core systems, the automatic relabeling script now contains the **-T 0** option in the **fixfiles** command line. The **-T 0** option ensures that the **setfiles** program uses the maximum of available processor cores for relabeling by default.

To use only one process thread for relabeling as in the previous version of RHEL, override this setting by entering either the **fixfiles -T 1 onboot** command instead of just **fixfiles onboot** or the **echo "-T 1" > /.autorelabel** command instead of **touch /.autorelabel**.

(BZ#2115242)

SCAP Security Guide rebased to 0.1.63

The SCAP Security Guide (SSG) packages have been rebased to upstream version 0.1.63. This version provides various enhancements and bug fixes, most notably:

- New compliance rules for **sysctl**, **grub2**, **pam_pwquality**, and build time kernel configuration were added.
- Rules hardening the PAM stack now use **authselect** as the configuration tool. Note: With this change, the rules hardening the PAM stack are not applied if the PAM stack was edited by other means.

(BZ#2070563)

Added a maximum size option for Rsyslog error files

Using the new **action.errorfile.maxsize** option, you can specify a maximum number of bytes of the error file for the Rsyslog log processing system. When the error file reaches the specified size, Rsyslog cannot write any additional errors or other data in it. This prevents the error file from filling up the file system and making the host unusable.

(BZ#2064318)

clevis-luks-askpass is now enabled by default

The **/lib/systemd/system-preset/90-default.preset** file now contains the **enable clevis-luks-askpass.path** configuration option and the installation of the **clevis-systemd** sub-package ensures that the **clevis-luks-askpass.path** unit file is enabled. This enables the Clevis encryption client to unlock also LUKS-encrypted volumes that mount late in the boot process. Before this update, the administrator must use the **systemctl enable clevis-luks-askpass.path** command to enable Clevis to unlock such volumes.

(BZ#2107078)

fapolicyd rebased to 1.1.3

The **fapolicyd** packages have been upgraded to version 1.1.3. Notable improvements and bug fixes include:

- Rules can now contain the new subject PPID attribute, which matches the parent PID (process ID) of a subject.
- The OpenSSL library replaced the Libgcrypt library as a cryptographic engine for hash computations.
- The **fagenrules --load** command now works correctly.

(BZ#2100041)

4.8. NETWORKING

The act_ctinfo kernel module has been added

This enhancement adds the **act_ctinfo** kernel module to RHEL. Using the **ctinfo** action of the **tc** utility, administrators can copy the **conntrack** mark or the value of the differentiated services code point (DSCP) of network packets into the socket buffer's **mark** metadata field. As a result, you can use conditions based on the **conntrack** mark or the DSCP value to filter traffic. For further details, see the **tc-ctinfo(8)** man page.

(BZ#2027894)

cloud-init updates network configuration at every boot on Microsoft Azure

Microsoft Azure does not change the instance ID when an administrator updates the network interface configuration while a VM is offline. With this enhancement, the **cloud-init** service always updates the network configuration when the VM boots to ensure that RHEL on Microsoft Azure uses the latest network settings.

As a consequence, if you manually configure settings on interfaces, such as an additional search domain, **cloud-init** may override them when you reboot the VM. For further details and a workaround, see the cloud-init-22.1-5 updates network config on every boot solution.

(BZ#2144898)

The PTP driver now supports virtual clocks and time stamping

With this enhancement, the Precision Time Protocol (PTP) driver can create virtual PTP Hardware Clocks (PHCs) on top of a free-running PHC by writing to **/sys/class/ptp/ptp*/n_vclocks**. As a result, users can run multiple domain synchronization with hardware time stamps on one interface.

(BZ#2066451)

firewalld was rebased to version 1.1.1

The **firewalld** packages have been upgraded to version 1.1.1. This version provides multiple bug fixes and enhancements over the previous version:

New features:

- Rich rules support NetFilter-log (NFLOG) target for user-space logging. Note that there is not any NFLOG capable logging daemon in RHEL. However, you can use the **tcpdump -i nflog** command to collect the logs you need.
- Support for port forwarding in policies with **ingress-zones=HOST** and **egress-zones={ANY, *source based zone*}**.

Other notable changes include:

- Support for the **afp**, **http3**, **jellyfin**, **netbios-ns**, **ws-discovery**, and **ws-discovery-client** services
- Tab-completion and sub-options in Z Shell for the **policy** option

(BZ#2040689)

NetworkManager now supports advmss, rto_min, and quickack route attributes

With this enhancement, administrators can configure the **ipv4.routes** setting with the following attributes:

- **rto_min** (TIME) - configure the minimum TCP re-transmission timeout in milliseconds when communicating with the route destination
- **quickack** (BOOL) - a per-route setting to enable or disable TCP quick ACKs
- **advmss** (NUMBER) - advertise maximum segment size (MSS) to the route destination when establishing TCP connections. If unspecified, Linux uses a default value calculated from the maximum transmission unit (MTU) of the first hop device

Benefit of implementing the new functionality of **ipv4.routes** with the mentioned attributes is that there is no need to run the **dispatcher** script.

Note that once you activate a connection with the mentioned route attributes, such changes are set in the kernel.

(BZ#2068525)

Support for the 802.ad vlan-protocol option in nmstate

The **nmstate** API now supports creating the **linux-bridge** interfaces using the 802.ad **vlan-protocol** option. This feature enables the configuration of Service-Tag VLANs. The following example illustrates usage of this functionality in a **yaml** configuration file.

```
---
interfaces:
  - name: br0
    type: linux-bridge
    state: up
    bridge:
      options:
        vlan-protocol: 802.1ad
      port:
        - name: eth1
          vlan:
            mode: trunk
            trunk-tags:
            - id: 500
```

(BZ#2084474)

The firewalld service can forward NAT packets originating from the local host to a different host and port

You can forward packets sent from the localhost that runs the **firewalld** service to a different destination port and IP address. The functionality is useful, for example, to forward ports on the **loopback** device to a container or a virtual machine. Prior to this change, **firewalld** could only forward ports when it received a packet that originated from another host. For more details and an illustrative configuration, see Using DNAT to forward HTTPS traffic to a different host .

(BZ#2039542)

NetworkManager now supports migration from ifcfg-rh to key file

Users can migrate their existing connection profile files from the **ifcfg-rh** format to the key file format. This way, all connection profiles will be in one location and in the preferred format. The key file format has the following advantages:

- Closely resembles the way how NetworkManager expresses network configuration
- Guarantees compatibility with future RHEL releases
- Is easier to read
- Supports all connection profiles

To migrate the connections, run:

```
# nmcli connection migrate
```

Note that the **ifcfg-rh** files will work correctly during the RHEL 9 lifetime. However, migrating the configuration to the key file format guarantees compatibility beyond RHEL 9.

For more details, see the **nmcli(1)**, nm-settings-keyfile(5), and nm-settings-ifcfg-rh(5) manual pages.

(BZ#2059608)

More DHCP and IPv6 auto-configuration attributes have been added to the nmstate API

This enhancement adds support for the following attributes to the nmstate API:

- **dhcp-client-id** for DHCPv4 connections as described in RFC 2132 and 4361.
- **dhcp-duid** for DHCPv6 connections as described in RFC 8415.
- **addr-gen-mode** for IPv6 auto-configuration. You can set this attribute to:
  - **eui64** as described in RFC 4862
  - **stable-privacy** as described in RFC 7217

(BZ#2082043)

NetworkManager now clearly indicates that WEP support is not available in RHEL 9

The **wpa_supplicant** packages in RHEL 9.0 and later no longer contain the deprecated and insecure Wired Equivalent Privacy (WEP) security algorithm. This enhancement updates NetworkManager to reflect these changes. For example, the **nmcli device wifi list** command now returns WEP access points at the end of the list in gray color, and connecting to a WEP-protected network returns a meaningful error message.

For secure encryption, use only wifi networks with Wi-Fi Protected Access 2 (WPA2) and WPA3 authentication.

(BZ#2030997)

The MPTCP code has been updated

The MultiPath TCP (MPTCP) code in the kernel has been updated and upstream Linux 5.19. This update provides a number of bug fixes and enhancements over the previous version:

- The **FASTCLOSE** option has been added to close MPTCP connections without a full three-way handshake.
- The **MP_FAIL** option has been added to enable fallback to TCP even after the initial handshake.
- The monitoring capabilities have been improved by adding additional Management Information Base (MIB) counters.
- Monitor support for MPTCP listener sockets has been added. Use the **ss** utility to monitor the sockets.

(BZ#2079368)

4.9. KERNEL

Kernel version in RHEL 9.1

Red Hat Enterprise Linux 9.1 is distributed with the kernel version 5.14.0-162.

(BZ#2125549)

Memory consumption of the list_lru has been optimized

The internal kernel data structure, **list_lru**, tracks the "Least Recently Used" status of kernel inodes and directory entries for files. Previously, the number of **list_lru** allocated structures was directly proportional to the number of mount points and the number of present memory **cgroups**. Both these numbers increased with the number of running containers leading to memory consumption of **O(n^2)** where **n** is the number of running containers. This update optimizes the memory consumption of **list_lru** in the system to **O(n)**. As a result, sufficient memory is now available for the user applications, especially on the systems with a large number of running containers.

(BZ#2013413)

BPF rebased to Linux kernel version 5.16

The Berkeley Packet Filter (BPF) facility has been rebased to Linux kernel version 5.16 with multiple bug fixes and enhancements. The most notable changes include:

- Streamlined internal BPF program sections handling and **bpf_program__set_attach_target()** API in the **libbpf** userspace library.
  The **bpf_program__set_attach_target()** API sets the BTF based attach targets for BPF based programs.
- Added support for the **BTF_KIND_TAG** kind, which allows you to tag declarations.
- Added support for the **bpf_get_branch_snapshot()** helper, which enables the tracing program to capture the last branch records (LBR) from the hardware.
- Added the legacy **kprobe** events support in the **libbpf** userspace library that enables **kprobe** tracepoint events creation through the legacy interface.
- Added the capability to access hardware timestamps through BPF specific structures with the **__sk_buff** helper function.
- Added support for a batched interface for RX buffer allocation in **AF_XDP** buffer pool, with driver support for **i40e** and **ice**.
- Added the legacy **uprobe** support in **libbpf** userspace library to complement recently merged legacy **kprobe**.
- Added the **bpf_trace_vprintk()** as variadic **printk** helper.
- Added the **libbpf** opt-in for stricter BPF program section name handling as part of **libbpf** 1.0 effort.
- Added the **libbpf** support to locate specialized maps, such as **perf RB** and internally delete BTF type identifiers while creating them.
- Added the **bloomfilter** BPF map type to test if an element exists in a set.
- Added support for kernel module function calls from BPF.
- Added support for typeless and weak **ksym** in light skeleton.

- Added support for the **BTF_KIND_DECL_TAG** kind.

For more information on the full list of BPF features available in the running kernel, use the **bpftool feature** command.

(BZ#2069045)

BTF data is now located in the kernel module

BPF Type Format (BTF) is the metadata format that encodes the debug information related to BPF program and map. Previously, the BTF data for kernel modules was stored in the **kernel-debuginfo** package. As a consequence, it was necessary to install the corresponding **kernel-debuginfo** package in order to use BTF for kernel modules. With this update, the BTF data is now located directly in the kernel module. As a result, you do not need to install any additional packages for BTF to work.

(BZ#2097188)

The kernel-rt source tree has been updated to RHEL 9.1 tree

The **kernel-rt** sources have been updated to use the latest Red Hat Enterprise Linux kernel source tree. The real-time patch set has also been updated to the latest upstream version, **v5.15-rt**. These updates provide a number of bug fixes and enhancements.

(BZ#2061574)

Dynamic preemptive scheduling enabled on ARM and AMD and Intel 64-bit architectures

RHEL 9 provides the dynamic scheduling feature on the ARM and AMD and Intel 64-bit architectures. This enhancement enables changing the preemption mode of the kernel at boot or runtime instead of the compile time. The **/sys/kernel/debug/sched/preempt** file contains the current setting and allows **runtime** modification.

Using the **DYNAMIC_PREEMPT** option, you can set the **preempt=** variable at boot time to either **none**, **voluntary** or **full** with **voluntary** preemption being the default. Using dynamic preemptive handling, you can override the default preemption model to improve scheduling latency.

(BZ#2065226)

stalld rebased to version 1.17

The **stalld** program, which provides the **stall** daemon, is a mechanism to prevent the starvation state of operating system threads in a Linux system. This version monitors the threads for the starvation state. Starvation occurs when a thread is on a CPU run queue for longer than the starvation threshold.

This **stalld** version includes many improvements and bug fixes over the previous version. The notable change includes the capability to detect runnable dying tasks.

When **stalld** detects a starving thread, the program changes the scheduling class of the thread to the **SCHED_DEADLINE** policy, which gives the thread a small slice of time for the specified CPU to run the thread. When the **timeslice** is used, the thread returns to its original scheduling policy and **stalld** continues to monitor the thread states.

(BZ#2107275)

The tpm2-tools package has been rebased to tpm2-tools-5.2-1 version

The **tpm2-tools** package has been rebased to version **tpm2-tools-5.2-1**. This upgrade provides many significant enhancements and bug fixes. Most notable changes include:

- Adds support for public-key output at primary object creation using the **tpm2_createprimary** and **tpm2_create** tools.
- Adds support for the **tpm2_print** tool to print public-key output formats. **tpm2_print** decodes a Trusted Platform Module (TPM) data structure and prints enclosed elements.
- Adds support to the **tpm2_eventlog** tool for reading logs larger than 64 KB.
- Adds the **tpm2_sessionconfig** tool to support displaying and configuring session attributes.

For more information on notable changes, see the **/usr/share/doc/tpm2-tools/Changelog.md** file.

(BZ#2090748)

Intel E800 devices now support iWARP and RoCE protocols

With this enhancement, you can now use the **enable_iwarp** and **enable_roce** devlink parameters to turn on and off iWARP or RoCE protocol support. With this mandatory feature, you can configure the device with one of the protocols. The Intel E800 devices do not support both protocols simultaneously on the same port.

To enable or disable the iWARP protocol for a specific E800 device, first obtain the PCI location of the card:

```
$ lspci | awk '/E810/ {print $1}'
44:00.0
44:00.1
$
```

Then enable, or disable, the protocol. You can use use **pci/0000:44:00.0** for the first port, and **pci/0000:44:00.1** for second port of the card as argument to the devlink command

```
$ devlink dev param set pci/0000:44:00.0 name enable_iwarp value true cmode runtime
$ devlink dev param set pci/0000:44:00.0 name enable_iwarp value false cmode runtime
```

To enable or disable the RoCE protocol for a specific E800 device, obtain the PCI location of the card as shown above. Then use one of the following commands:

```
$ devlink dev param set pci/0000:44:00.0 name enable_roce value true cmode runtime
$ devlink dev param set pci/0000:44:00.0 name enable_roce value false cmode runtime
```

(BZ#2096127)

4.10. BOOT LOADER

GRUB is signed by new keys

Due to security reasons, GRUB is now signed by new keys. As a consequence, you need to update the RHEL firmware to version FW1010.30 (or later) or FW1020 to be able to boot the little-endian variant of IBM Power Systems with the Secure Boot feature enabled.

(BZ#2074761)

Configurable disk access retries when booting a VM on IBM POWER

You can now configure how many times the GRUB boot loader retries accessing a remote disk when a logical partition (**lpar**) virtual machine (VM) boots on the IBM POWER architecture. Lowering the number of retries can prevent a slow boot in certain situations.

Previously, GRUB retried accessing disks 20 times when disk access failed at boot. This caused problems if you performed a Live Partition Mobility (LPM) migration on an **lpar** system that connected to slow Storage Area Network (SAN) disks. As a consequence, the boot might have taken very long on the system until the 20 retries finished.

With this update, you can now configure and decrease the number of disk access retries using the **ofdisk_retries** GRUB option. For details, see Configure disk access retries when booting a VM on IBM POWER.

As a result, the **lpar** boot is no longer slow after LPM on POWER, and the **lpar** system boots without the failed disks.

(BZ#2070725)

4.11. FILE SYSTEMS AND STORAGE

Stratis now enables setting the file system size upon creation

You can now set the required size when creating a file system. Previously, the automatic default size was 1 TiB. With this enhancement, users can set an arbitrary filesystem size. The lower limit must not go below 512 MiB.

(BZ#1990905)

Improved overprovision management of Stratis pools

With the improvements to the management of thin provisioning, you can now have improved warnings, precise allocation of space for the pool metadata, improved predictability, overall safety, and reliability of thin pool management. A new distinct mode disables overprovisioning. With this enhancement, the user can disable overprovisioning to ensure that a pool contains enough space to support all its file systems, even if these are completely full.

(BZ#2040352)

Stratis now provides improved individual pool management

You can now stop and start stopped individual Stratis pools. Previously, **stratisd** attempted to start all available pools for all devices it detected. This enhancement provides more flexible management of individual pools within Stratis, better debugging and recovery capabilities. The system no longer requires a reboot to perform recovery and maintenance operations for a single pool.

(BZ#2039960)

Enabled protocol specific configuration of multipath device paths

Previously due to different optimal configurations for the different protocols, it was impossible to set the configuration correctly without setting an option for each individual protocol. With this enhancement, users can now configure multipath device paths based on their path transport protocol. Use the **protocol** subsection of the **overrides** section in the **/etc/multipath.conf** file to correctly configure multipath device paths, based on their protocol.

(BZ#2084365)

New libnvme feature library

Previously, the NVMe storage command line interface utility (**nvme-cli**) included all of the helper functions and definitions. This enhancement brings a new **libnvme** library to RHEL 9.1. The library includes:

- Type definitions for NVMe specification structures
- Enumerations and bit fields
- Helper functions to construct, dispatch, and decode commands and payloads
- Utilities to connect, scan, and manage NVMe devices

With this update, users do not need to duplicate the code and multiple projects and packages, such as **nvme-stas**, and can rely on this common library.

(BZ#2099619)

A new library libnvme is now available

With this update, nvme-cli is divided in two different projects: * **nvme-cli** now only contains the code specific to the **nvme** tool * **libnvme** library now contains all type definitions for NVMe specification structures, enumerations, bit fields, helper functions to construct, dispatch, decode commands and payloads, and utilities to connect, scan, and manage NVMe devices.

(BZ#2090121)

4.12. HIGH AVAILABILITY AND CLUSTERS

Support for High Availability on Red Hat OpenStack platform

You can now configure a high availability cluster on the Red Hat OpenStack platform. In support of this feature, Red Hat provides the following new cluster agents:

- **fence_openstack**: fencing agent for HA clusters on OpenStack
- **openstack-info**: resource agent to configure the **openstack-info** cloned resource, which is required for an HA cluster on OpenStack
- **openstack-virtual-ip**: resource agent to configure a virtual IP address resource
- **openstack-floating-ip**: resource agent to configure a floating IP address resource
- **openstack-cinder-volume**: resource agent to configure a block storage resource

(BZ#2121838)

pcs supports updating multipath SCSI devices without requiring a system restart

You can now update multipath SCSI devices with the **pcs stonith update-scsi-devices** command. This command updates SCSI devices without causing a restart of other cluster resources running on the same node.

(BZ#2024522)

Support for cluster UUID

During cluster setup, the **pcs** command now generates a UUID for every cluster. Since a cluster name is not a unique cluster identifier, you can use the cluster UUID to identify clusters with the same name when you administer multiple clusters.

You can display the current cluster UUID with the **pcs cluster config [show]** command. You can add a UUID to an existing cluster or regenerate a UUID if it already exists by using the **pcs cluster config uuid generate** command.

(BZ#2054671)

New pcs resource config command option to display the pcs commands that re-create configured resources

The **pcs resource config** command now accepts the **--output-format=cmd** option. Specifying this option displays the **pcs** commands you can use to re-create configured resources on a different system.

(BZ#2058251)

New pcs stonith config command option to display the pcs commands that re-create configured fence devices

The **pcs stonith config** command now accepts the **--output-format=cmd** option. Specifying this option displays the **pcs** commands you can use to re-create configured fence devices on a different system.

(BZ#2058252)

Pacemaker rebased to version 2.1.4

The Pacemaker packages have been upgraded to the upstream version of Pacemaker 2.1.4. Notable changes include:

- The **multiple-active** resource parameter now accepts a value of **stop_unexpected**, The **multiple-active** resource parameter determines recovery behavior when a resource is active on more than one node when it should not be. By default, this situation requires a full restart of the resource, even if the resource is running successfully where it should be. A value of **stop_unexpected** for this parameter specifies that only unexpected instances of a multiply-active resource are stopped. It is the user's responsibility to verify that the service and its resource agent can function with extra active instances without requiring a full restart.

- Pacemaker now supports the **allow-unhealthy-node** resource meta-attribute. When this meta-attribute is set to **true**, the resource is not forced off a node due to degraded node health. When health resources have this attribute set, the cluster can automatically detect if the node's health recovers and move resources back to it.

- Users can now specify Access Control Lists (ACLS) for a system group using the **pcs acl group** command. Pacemaker previously allowed ACLs to be specified for individual users, but it is sometimes simpler and would conform better with local policies to specify ACLs for a system group, and to have them apply to all users in that group. This command was present in earlier releases but had no effect.

(BZ#2072108)

Samba no longer automatically installed with cluster packages

As of this release, installing the packages for the RHEL High Availability Add-On no longer installs the Samba packages automatically. This also allows you to remove the Samba packages without automatically removing the HA packages as well. If your cluster uses Samba resources you must now manually install them.

(BZ#1826455)

4.13. DYNAMIC PROGRAMMING LANGUAGES, WEB AND DATABASE SERVERS

The nodejs:18 module stream is now fully supported

The **nodejs:18** module stream, previously available as a Technology Preview, is fully supported with the release of the RHSA-2022:8832 advisory. The **nodejs:18** module stream now provides **Node.js 18.12**, which is a Long Term Support (LTS) version.

Node.js 18 included in RHEL 9.1 provides numerous new features together with bug and security fixes over **Node.js 16**.

Notable changes include:

- The **V8** engine has been upgraded to version 10.2.
- The **npm** package manager has been upgraded to version 8.19.2.
- **Node.js** now provides a new experimental **fetch** API.
- **Node.js** now provides a new experimental **node:test** module, which facilitates the creation of tests that report results in the Test Anything Protocol (TAP) format.
- **Node.js** now prefers IPv6 addresses over IPv4.

To install the **nodejs:18** module stream, use:

```
# dnf module install nodejs:18
```

(BZ#2083072)

A new module stream: php:8.1

RHEL 9.1 adds **PHP 8.1** as a new **php:8.1** module stream.

With **PHP 8.1**, you can:

- Define a custom type that is limited to one of a discrete number of possible values using the Enumerations (Enums) feature
- Declare a property with the **readonly** modifier to prevent modification of the property after initialization
- Use fibers, full-stack, interruptible functions

To install the **php:8.1** module stream, use:

```
# dnf module install php:8.1
```

For details regarding PHP usage on RHEL 9, see Using the PHP scripting language.

(BZ#2070040)

A new module stream: ruby:3.1

RHEL 9.1 introduces **Ruby 3.1.2** in a new **ruby:3.1** module stream. This version provides a number of performance improvements, bug and security fixes, and new features over **Ruby 3.0** distributed with RHEL 9.0.

Notable enhancements include:

- The **Interactive Ruby** (IRB) utility now provides an autocomplete feature and a documentation dialog
- A new **debug** gem, which replaces **lib/debug.rb**, provides improved performance, and supports remote debugging and multi-process/multi-thread debugging
- The **error_highlight** gem now provides a fine-grained error location in the backtrace
- Values in the hash literal data types and keyword arguments can now be omitted
- The pin operator (**^**) now accepts an expression in pattern matching
- Parentheses can now be omitted in one-line pattern matching
- YJIT, a new experimental in-process Just-in-Time (JIT) compiler, is now available on the AMD and Intel 64-bit architectures
- The **TypeProf For IDE** utility has been introduced, which is an experimental static type analysis tool for **Ruby** code in IDEs

The following performance improvements have been implemented in Method Based Just-in-Time Compiler (MJIT):

- For workloads like **Rails**, the default maximum JIT cache value has increased from 100 to 10000
- Code compiled using JIT is no longer canceled when a **TracePoint** for class events is enabled

Other notable changes include:

- The **tracer.rb** file has been removed
- Since version 4.0, the **Psych** YAML parser uses the **safe_load** method by default

To install the **ruby:3.1** module stream, use:

```
# dnf module install ruby:3.1
```

(BZ#2063773)

httpd rebased to version 2.4.53

The Apache HTTP Server has been updated to version 2.4.53, which provides bug fixes, enhancements, and security fixes over version 2.4.51 distributed with RHEL 9.0.

Notable changes in the **mod_proxy** and **mod_proxy_connect** modules include:

- **mod_proxy**: The length limit of the name of the controller has been increased
- **mod_proxy**: You can now selectively configure timeouts for backend and frontend

- **mod_proxy**: You can now disable TCP connections redirection by setting the **SetEnv proxy-nohalfclose** parameter
- **mod_proxy** and **mod_proxy_connect**: It is forbidden to change a status code after sending it to a client

In addition, a new **ldap** function has been added to the expression API, which can help prevent the LDAP injection vulnerability.

(BZ#2079939)

A new default for the LimitRequestBody directive in httpd configuration

To fix CVE-2022-29404, the default value for the **LimitRequestBody** directive in the Apache HTTP Server has been changed from **0** (unlimited) to 1 GiB.

On systems where the value of **LimitRequestBody** is not explicitly specified in an **httpd** configuration file, updating the **httpd** package sets **LimitRequestBody** to the default value of 1 GiB. As a consequence, if the total size of the HTTP request body exceeds this 1 GiB default limit, **httpd** returns the **413 Request Entity Too Large** error code.

If the new default allowed size of an HTTP request message body is insufficient for your use case, update your **httpd** configuration files within the respective context (server, per-directory, per-file, or per-location) and set your preferred limit in bytes. For example, to set a new 2 GiB limit, use:

```
LimitRequestBody 2147483648
```

Systems already configured to use any explicit value for the **LimitRequestBody** directive are unaffected by this change.

(BZ#2128016)

New package: httpd-core

Starting with RHEL 9.1, the **httpd** binary file with all essential files has been moved to the new **httpd-core** package to limit the Apache HTTP Server's dependencies in scenarios where only the basic **httpd** functionality is needed, for example, in containers.

The **httpd** package now provides **systemd**-related files, including **mod_systemd**, **mod_brotli**, and documentation.

With this change, the **httpd** package no longer provides the **httpd** Module Magic Number (MMN) value. Instead, the **httpd-core** package now provides the **httpd-mmn** value. As a consequence, fetching **httpd-mmn** from the **httpd** package is no longer possible.

To obtain the **httpd-mmn** value of the installed **httpd** binary, you can use the **apxs** binary, which is a part of the **httpd-devel** package. To obtain the **httpd-mmn** value, use the following command:

```
# apxs -q HTTPD_MMN
20120211
```

(BZ#2065677)

pcre2 rebased to version 10.40

The **pcre2** package, which provides the Perl Compatible Regular Expressions library v2, has been updated to version 10.40.

With this update, the use of the **\K** escape sequence in lookaround assertions is forbidden, in accordance with the respective change in **Perl 5.32**. If you rely on the previous behavior, you can use the **PCRE2_EXTRA_ALLOW_LOOKAROUND_BSK** option. Note that when this option is set, **\K** is accepted only inside positive assertions but is ignored in negative assertions.

(BZ#2086494)

4.14. COMPILERS AND DEVELOPMENT TOOLS

The updated GCC compiler is now available for RHEL 9.1

The system GCC compiler, version 11.2.1, has been updated to include numerous bug fixes and enhancements available in the upstream GCC.

The GNU Compiler Collection (GCC) provides tools for developing applications with the C, C++, and Fortran programming languages.

For usage information, see Developing C and C++ applications in RHEL 9 .

(BZ#2063255)

New GCC Toolset 12

GCC Toolset 12 is a compiler toolset that provides recent versions of development tools. It is available as an Application Stream in the form of a Software Collection in the **AppStream** repository.

The GCC compiler has been updated to version 12.1.1, which provides many bug fixes and enhancements that are available in upstream GCC.

The following tools and versions are provided by GCC Toolset 12:

Tool	Version
GCC	12.1.1
GDB	11.2
binutils	2.35
dwz	0.14
annobin	10.76

To install GCC Toolset 12, run the following command as root:

```
# dnf install gcc-toolset-12
```

To run a tool from GCC Toolset 12:

```
$ scl enable gcc-toolset-12 tool
```

To run a shell session where tool versions from GCC Toolset 12 override system versions of these tools:

```
$ scl enable gcc-toolset-12 bash
```

For more information, see GCC Toolset 12.

(BZ#2077465)

GCC Toolset 12: Annobin rebased to version 10.76

In GCC Toolset 12, the Annobin package has been updated to version 10.76.

Notable bug fixes and enhancements include:

- A new command line option for annocheck tells it to avoid using the **debuginfod** service, if it is unable to find debug information in another way. Using **debuginfod** provides annocheck with more information, but it can also cause significant slow downs in annocheck's performance if the **debuginfod** server is unavailable.
- The Annobin sources can now be built using **meson** and **ninja** rather than configure and make if desired.
- Annocheck now supports binaries built by the Rust 1.18 compiler.

Additionally, the following known issue has been reported in the GCC Toolset 12 version of Annobin:

Under some circumstances it is possible for a compilation to fail with an error message that looks similar to the following:

```
cc1: fatal error: inaccessible plugin file
opt/rh/gcc-toolset-12/root/usr/lib/gcc/architecture-linux-gnu/12/plugin/gcc-annobin.so
expanded from short plugin name gcc-annobin: No such file or directory
```

To work around the problem, create a symbolic link in the plugin directory from **annobin.so** to **gcc-annobin.so**:

```
# cd /opt/rh/gcc-toolset-12/root/usr/lib/gcc/architecture-linux-gnu/12/plugin
# ln -s annobin.so gcc-annobin.so
```

Where *architecture* is replaced with the architecture being used:

- **aarch64**
- **i686**
- **ppc64le**
- **s390x**
- **x86_64**

(BZ#2077438)

GCC Toolset 12: binutils rebased to version 2.38

In GCC Toolset 12, the **binutils** package has been updated to version 2.38.

Notable bug fixes and enhancements include:

- All tools in the **binutils** package now support options to display or warn about the presence of multibyte characters.

- The **readelf** and **objdump** tools now automatically follow any links to separate **debuginfo** files by default. This behavior can be disabled by using the **--debug-dump=no-follow-links** option for **readelf** or the **--dwarf=no-follow-links** option for **objdump**.

(BZ#2077445)

GCC 12 and later supports _FORTIFY_SOURCE level 3

With this enhancement, users can build applications with **-D_FORTIFY_SOURCE=3** in the compiler command line when building with GCC version 12 or later. **_FORTIFY_SOURCE** level 3 improves coverage of source code fortification, thus improving security for applications built with **-D_FORTIFY_SOURCE=3** in the compiler command line. This is supported in GCC versions 12 and later and all Clang in RHEL 9 with the **__builtin_dynamic_object_size** builtin.

(BZ#2033683)

DNS stub resolver option now supports no-aaaa option

With this enhancement, **glibc** now recognizes the **no-aaaa** stub resolver option in **/etc/resolv.conf** and the **RES_OPTIONS** environment variable. When this option is active, no AAAA queries will be sent over the network. System administrators can disable AAAA DNS lookups for diagnostic purposes, such as ruling out that the superfluous lookups on IPv4-only networks do not contribute to DNS issues.

(BZ#2096191)

Added support for IBM Z Series z16

The support is now available for the **s390** instruction set with the **IBM z16** platform. **IBM z16** provides two additional hardware capabilities in **glibc** that are **HWCAP_S390_VXRS_PDE2** and **HWCAP_S390_NNPA**. As a result, applications can now use these capabilities to deliver optimized libraries and functions.

(BZ#2077838)

Applications can use the restartable sequence features through the new glibc interfaces

To accelerate the **sched_getcpu** function (especially on aarch64), it is necessary to use the restartable sequences (rseq) kernel feature by default in **glibc**. To allow applications to continuously use the shared rseq area, **glibc** now provides the **__rseq_offset**, **__rseq_size** and **__rseq_flags** symbols which were first added in **glibc** 2.35 upstream version. With this enhancement, the performance of the **sched_getcpu** function is increased and applications can now use the restartable sequence features through the new **glibc** interfaces.

(BZ#2085529)

GCC Toolset 12: GDB rebased to version 11.2

In GCC Toolset 12, the GDB package has been updated to version 11.2.

Notable bug fixes and enhancements include:

- New support for the 64-bit ARM architecture Memory Tagging Extension (MTE). See new commands with the **memory-tag** prefix.

- **--qualified** option for **-break-insert** and **-dprintf-insert**. This option looks for an exact match of the user's event location instead of searching in all scopes.

For example, **break --qualified *foo*** will look for a symbol named *foo* in the global scope. Without **--qualified**, GDB will search all scopes for a symbol with that name.

- **--force-condition**: Any supplied condition is defined even if it is currently invalid.
- **-break-condition --force**: Likewise for the MI command.
- **-file-list-exec-source-files** accepts optional **REGEXP** to limit output.
- **.gdbinit** search path includes the config directory. The order is:
  a. **$XDG_CONFIG_HOME/gdb/gdbinit**
  b. **$HOME/.config/gdb/gdbinit**
  c. **$HOME/.gdbinit**
- Support for ~/**.config/gdb/gdbearlyinit** or ~/**.gdbearlyinit**.
- **-eix** and **-eiex** early initialization file options.

Terminal user interface (TUI):

- Support for mouse actions inside terminal user interface (TUI) windows.
- Key combinations that do not act on the focused window are now passed to GDB.

New commands:

- **show print memory-tag-violations**
- **set print memory-tag-violations**
- **memory-tag show-logical-tag**
- **memory-tag with-logical-tag**
- **memory-tag show-allocation-tag**
- **memory-tag check**
- **show startup-quietly** and **set startup-quietly**: A way to specify **-q** or **-quiet** in GDB scripts. Only valid in early initialization files.
- **show print type hex** and **set print type hex**: Tells GDB to print sizes or offsets for structure members in hexadecimal instead of decimal.
- **show python ignore-environment** and **set python ignore-environment**: If enabled, GDB's Python interpreter ignores Python environment variables, much like passing **-E** to the Python executable. Only valid in early initialization files.
- **show python dont-write-bytecode** and **set python dont-write-bytecode**: If **off**, these commands suppress GDB's Python interpreter from writing bytecode compiled objects of imported modules, much like passing **-B** to the Python executable. Only valid in early initialization files.

Changed commands:

- **break *LOCATION* if *CONDITION***: If *CONDITION* is invalid, GDB refuses to set a breakpoint. The **-force-condition** option overrides this.
- ***CONDITION* -force N COND**: Same as the previous command.
- **inferior [ID]**: When ID is omitted, this command prints information about the current inferior. Otherwise, unchanged.
- **ptype[*/FLAGS*] TYPE | EXPRESSION**: Use the /**x** flag to use hexadecimal notation when printing sizes and offsets of struct members. Use the /**d** flag to do the same but using decimal.
- **info sources**: Output has been restructured.

Python API:

- Inferior objects contain a read-only **connection_num** attribute.
- New **gdb.Frame.level()** method.
- New **gdb.PendingFrame.level()** method.
- **gdb.BreakpoiontEvent** emitted instead of **gdb.Stop**.

(BZ#2077494)

GDB supports Power 10 PLT instructions

GDB now supports Power 10 PLT instructions. With this update, users are able to step into shared library functions and inspect stack backtraces using GDB version 10.2-10 and later.

(BZ#1870017)

The dyninst packaged rebased to version 12.1

The **dyninst** package has been rebased to version 12.1. Notable bug fixes and enhancements include:

- Initial support for **glibc-2.35** multiple namespaces
- Concurrency fixes for DWARF parallel parsing
- Better support for the **CUDA** and **CDNA2** GPU binaries
- Better support for IBM POWER Systems (little endian) register access
- Better support for PIE binaries
- Corrected parsing for catch blocks
- Corrected access to 64-bit Arm (**aarch64**) floating point registers

(BZ#2057675)

A new fileset /etc/profile.d/debuginfod.*

Added new fileset for activating organizational debuginfod services. To get a system-wide **debuginfod** client activation you must add the URL to **/etc/debuginfod/FOO.urls** file.

(BZ#2088774)

Rust Toolset rebased to version 1.62.1

Rust Toolset has been updated to version 1.62.1. Notable changes include:

- Destructuring assignment allows patterns to assign to existing variables in the left-hand side of an assignment. For example, a tuple assignment can swap to variables: **(a, b) = (b, a);**
- Inline assembly is now supported on 64-bit x86 and 64-bit ARM using the **core::arch::asm!** macro. See more details in the "Inline assembly" chapter of the reference, **/usr/share/doc/rust/html/reference/inline-assembly.html** (online at https://doc.rust-lang.org/reference/inline-assembly.html).
- Enums can now derive the **Default** trait with an explicitly annotated **#[default]** variant.
- **Mutex**, **CondVar**, and **RwLock** now use a custom **futex**-based implementation rather than pthreads, with new optimizations made possible by Rust language guarantees.
- Rust now supports custom exit codes from **main**, including user-defined types that implement the newly-stabilized **Termination** trait.
- Cargo supports more control over dependency features. The **dep:** prefix can refer to an optional dependency without exposing that as a feature, and a **?** only enables a dependency feature if that dependency is enabled elsewhere, like **package-name?/feature-name**.
- Cargo has a new **cargo add** subcommand for adding dependencies to **Cargo.toml**.
- For more details, please see the series of upstream release announcements:
  - Announcing Rust 1.59.0
  - Announcing Rust 1.60.0
  - Announcing Rust 1.61.0
  - Announcing Rust 1.62.0
  - Announcing Rust 1.62.1

(BZ#2075337)

LLVM Toolset rebased to version 14.0.6

LLVM Toolset has been rebased to version 14.0.6. Notable changes include:

- On 64-bit x86, support for **AVX512-FP16** instructions has been added.
- Support for the Armv9-A, Armv9.1-A and Armv9.2-A architectures has been added.
- On PowerPC, added the **__ibm128** type to represent IBM double-double format, also available as **__attribute__((mode(IF)))**.

clang changes:

- **if consteval** for **C++2b** is now implemented.
- On 64-bit x86, support for **AVX512-FP16** instructions has been added.
- Completed support of OpenCL C 3.0 and **C++** for OpenCL 2021 at experimental state.

- The **-E -P** preprocessor output now always omits blank lines, matching GCC behavior. Previously, up to 8 consecutive blank lines could appear in the output.
- Support **-Wdeclaration-after-statement** with **C99** and later standards, and not just C89, matching GCC's behavior. A notable use case is supporting style guides that forbid mixing declarations and code, but want to move to newer C standards.

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

(BZ#2061041)

Go Toolset rebased to version 1.18.2

Go Toolset has been rebased to version 1.18.2.

Notable changes include:

- The introduction of generics while maintaining backwards compatibility with earlier versions of Go.
- A new fuzzing library.
- New **debug**/**buildinfo** and **net**/**netip** packages.
- The **go get** tool no longer builds or installs packages. Now, it only handles dependencies in **go.mod**.
- If the main module's **go.mod** file specifies **go 1.17** or higher, the **go mod download** command used without any additional arguments only downloads source code for the explicitly required modules in the main module's **go.mod** file. To also download source code for transitive dependencies, use the **go mod download all** command.
- The **go mod vendor** subcommand now supports a **-o** option to set the output directory.
- The **go mod tidy** command now retains additional checksums in the **go.sum** file for modules whose source code is required to verify that only one module in the build list provides each imported package. This change is not conditioned on the Go version in the main module's **go.mod** file.

(BZ#2075169)

A new module stream: maven:3.8

RHEL 9.1 introduces **Maven 3.8** as a new module stream.

To install the **maven:3.8** module stream, use:

```
# dnf module install maven:3.8
```

(BZ#2083112)

.NET version 7.0 is available

Red Hat Enterprise Linux 9.1 is distributed with **.NET** version 7.0. Notable improvements include:

- Support for IBM Power (**ppc64le**)

For more information, see Release Notes for .NET 7.0 RPM packages and Release Notes for .NET 7.0 containers.

(BZ#2112027)

4.15. IDENTITY MANAGEMENT

SSSD now supports memory caching for SID requests

With this enhancement, SSSD now supports memory caching for SID requests, which are GID and UID lookups by SID and vice versa. Memory caching results in improved performance, for example, when copying large amounts of files to or from a Samba server.

(JIRA:RHELPLAN-123369)

The ipaservicedelegationtarget and ipaservicedelegationrule Ansible modules are now available

You can now use the **ipaservicedelegationtarget** and **ipaservicedelegationrule ansible-freeipa** modules to, for example, configure a web console client to allow an Identity Management (IdM) user that has authenticated with a smart card to do the following:

- Use **sudo** on the RHEL host on which the web console service is running without being asked to authenticate again.
- Access a remote host using **SSH** and access services on the host without being asked to authenticate again.

The **ipaservicedelegationtarget** and **ipaservicedelegationrule** modules utilize the Kerberos **S4U2proxy** feature, also known as constrained delegation. IdM traditionally uses this feature to allow the web server framework to obtain an LDAP service ticket on the user's behalf. The IdM-AD trust system uses the feature to obtain a cifs principal.

(JIRA:RHELPLAN-117109)

SSSD support for anonymous PKINIT for FAST

With this enhancement, SSSD now supports anonymous PKINIT for Flexible Authentication via Secure Tunneling (FAST), also called Kerberos armoring in Active Directory. Until now, to use FAST, a Kerberos keytab was needed to request the required credentials. You can now use anonymous PKINIT to create this credential cache to establish the FAST session.

To enable anonymous PKINIT, perform the following steps:

1. Set **krb5_fast_use_anonymous_pkinit** to **true** in the **[domain]** section of the **sssd.conf** file.
2. Restart SSSD.
3. In an IdM environment, you can verify that anonymous PKINIT was used to establish the FAST session by logging in as the IdM user. A cache file with the FAST ticket is created and the **Default principal: WELLKNOWN/ANONYMOUS@WELLKNOWN:ANONYMOUS** indicates that anonymous PKINIT was used:

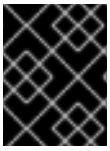
```
klist /var/lib/sss/db/fast_ccache_IPA.VM
Ticket cache: FILE:/var/lib/sss/db/fast_ccache_IPA.VM
Default principal: WELLKNOWN/ANONYMOUS@WELLKNOWN:ANONYMOUS
```

```
Valid starting Expires Service principal
03/10/2022 10:33:45 03/10/2022 10:43:45 krbtgt/IPA.VM@IPA.VM
```

(JIRA:RHELPLAN-123368)

IdM now supports Random Serial Numbers

With this update, Identity Management (IdM) now includes **dogtagpki 11.2.0**, which allows you to use Random Serial Numbers version 3 (RSNv3). You can enable RSNv3 by using the **--random-serial-numbers** option when running **ipa-server-install** or **ipa-ca-install**. With RSNv3 enabled, IdM generates fully random serial numbers for certificates and requests in PKI without range management. Using RSNv3, you can avoid range management in large IdM installations and prevent common collisions when reinstalling IdM.



IMPORTANT

RSNv3 is supported only for new IdM installations. If enabled, it is required to use RSNv3 on all PKI services.

(BZ#747959)

IdM now supports a limit on the number of LDAP binds allowed after a user password has expired

With this enhancement, you can set the number of LDAP binds allowed when the password of an Identity Management (IdM) user has expired:

-1
: IdM grants the user unlimited LDAP binds before the user must reset the password. This is the default value, which matches the previous behavior.

0
: This value disables all LDAP binds once a password is expired. In effect, the users must reset their password immediately.

1-MAXINT
: The value entered allows exactly that many binds post-expiration.

The value can be set in the global password policy and in group policies.

Note that the count is stored per server.

In order for a user to reset their own password they need to bind with their current, expired password. If the user has exhausted all post-expiration binds, then the password must be administratively reset.

(BZ#2091988)

New ipasmartcard_server and ipasmartcard_client roles

With this update, the **ansible-freeipa** package provides Ansible roles to configure Identity Management (IdM) servers and clients for smart card authentication. The **ipasmartcard_server** and **ipasmartcard_client** roles replace the **ipa-advise** scripts to automate and simplify the integration. The same inventory and naming scheme are used as in the other **ansible-freeipa** roles.

(BZ#2076567)

IdM now supports configuring an AD Trust with Windows Server 2022

With this enhancement, you can establish a cross-forest trust between Identity Management (IdM) domains and Active Directory forests that use Domain Controllers running Windows Server 2022.

(BZ#2122716)

The ipa-dnskeysyncd and ipa-ods-exporter debug messages are no longer logged to /var/log/messages by default

Previously, **ipa-dnskeysyncd**, the service that is responsible for the LDAP-to-OpenDNSSEC synchronization, and **ipa-ods-exporter**, the Identity Management (IdM) OpenDNSSEC exporter service, logged all debug messages to **/var/log/messages** by default. As a consequence, log files grew substantially. With this enhancement, you can configure the log level by setting **debug=True** in the **/etc/ipa/dns.conf** file. For more information, refer to **default.conf(5)**, the man page for the IdM configuration file.

(BZ#2083218)

samba rebased to version 4.16.1

The **samba** packages have been upgraded to upstream version 4.16.1, which provides bug fixes and enhancements over the previous version:

- By default, the **smbd** process automatically starts the new **samba-dcerpcd** process on demand to serve Distributed Computing Environment / Remote Procedure Calls (DCERPC). Note that Samba 4.16 and later always requires **samba-dcerpcd** to use DCERPC. If you disable the **rpc start on demand helpers** setting in the **[global]** section in the **/etc/samba/smb.conf** file, you must create a **systemd** service unit to run **samba-dcerpcd** in standalone mode.
- The Cluster Trivial Database (CTDB) **recovery master** role has been renamed to **leader**. As a result, the following **ctdb** sub-commands have been renamed:
  - **recmaster** to **leader**
  - **setrecmasterrole** to **setleaderrole**
- The CTDB **recovery lock** configuration has been renamed to **cluster lock**.
- CTDB now uses leader broadcasts and an associated timeout to determine if an election is required.

Note that the server message block version 1 (SMB1) protocol is deprecated since Samba 4.11 and will be removed in a future release.

Back up the database files before starting Samba. When the **smbd**, **nmbd**, or **winbind** services start, Samba automatically updates its **tdb** database files. Note that Red Hat does not support downgrading **tdb** database files.

After updating Samba, verify the **/etc/samba/smb.conf** file using the **testparm** utility.

For further information about notable changes, read the upstream release notes before updating.

(BZ#2077487)

SSSD now supports direct integration with Windows Server 2022

With this enhancement, you can use SSSD to directly integrate your RHEL system with Active Directory forests that use Domain Controllers running Windows Server 2022.

(BZ#2070793)

Improved SSSD multi-threaded performance

Previously, SSSD serialized parallel requests from multi-threaded applications, such as Red Hat Directory Server and Identity Management. This update fixes all SSSD client libraries, such as **nss** and **pam**, so they do not serialize requests, therefore allowing requests from multiple threads to be executed in parallel for better performance. To enable the previous behavior of serialization, set the environment variable **SSS_LOCKFREE** to **NO**.

(BZ#1978119)

Directory Server now supports canceling the Auto Membership plug-in task.

Previously, the Auto Membership plug-in task could generate high CPU usage on the server if Directory Server has complex configuration (large groups, complex rules and interaction with other plugins). With this enhancement, you can cancel the Auto Membership plug-in task. As a result, performance issues no longer occur.

(BZ#2052527)

Directory Server now supports recursive delete operations when using ldapdelete

With this enhancement, Directory Server now supports the **Tree Delete Control [1.2.840.113556.1.4.805]** OpenLDAP control. As a result, you can use the **ldapdelete** utility to recursively delete subentries of a parent entry.

(BZ#2057063)

You can now set basic replication options during the Directory Server installation

With this enhancement, you can configure basic replication options like authentication credentials and changelog trimming during an instance installation using an **.inf** file.

(BZ#2057066)

Directory Server now supports instance creation by a non-root user

Previously, non-root users were not able to create Directory Server instances. With this enhancement, a non-root user can use the **dscreate ds-root** subcommand to configure an environment where **dscreate**,**dsctl**,**dsconf** commands are used as usual to create and administer Directory Server instances.

(BZ#1872451)

pki packages renamed to idm-pki

The following **pki** packages are now renamed to **idm-pki** to better distinguish between IDM packages and Red Hat Certificate System ones:

- **idm-pki-tools**
- **idm-pki-acme**
- **idm-pki-base**
- **idm-pki-java**
- **idm-pki-ca**

- **idm-pki-kra**
- **idm-pki-server**
- **python3-idm-pki**

(BZ#2139877)

4.16. GRAPHICS INFRASTRUCTURES

Wayland is now enabled with Matrox GPUs

The desktop session now enables the Wayland back end with Matrox GPUs.

In previous releases, Wayland was disabled with Matrox GPUs due to performance and other limitations. These problems have now been fixed.

You can still switch the desktop session from Wayland back to Xorg. For more information, see Overview of GNOME environments.

(BZ#2097308)

12th generation Intel Core GPUs are now supported

This release adds support for several integrated GPUs for the 12th Gen Intel Core CPUs. This includes Intel UHD Graphics and Intel Xe integrated GPUs found with the following CPU models:

- Intel Core i3 12100T through Intel Core i9 12900KS
- Intel Pentium Gold G7400 and G7400T
- Intel Celeron G6900 and G6900T
- Intel Core i5-12450HX through Intel Core i9-12950HX
- Intel Core i3-1220P through Intel Core i7-1280P

(JIRA:RHELPLAN-135601)

Support for new AMD GPUs

This release adds support for several AMD Radeon RX 6000 Series GPUs and integrated graphics of the AMD Ryzen 6000 Series CPUs.

The following AMD Radeon RX 6000 Series GPU models are now supported:

- AMD Radeon RX 6400
- AMD Radeon RX 6500 XT
- AMD Radeon RX 6300M
- AMD Radeon RX 6500M

AMD Ryzen 6000 Series includes integrated GPUs found with the following CPU models:

- AMD Ryzen 5 6600U

- AMD Ryzen 5 6600H
- AMD Ryzen 5 6600HS
- AMD Ryzen 7 6800U
- AMD Ryzen 7 6800H
- AMD Ryzen 7 6800HS
- AMD Ryzen 9 6900HS
- AMD Ryzen 9 6900HX
- AMD Ryzen 9 6980HS
- AMD Ryzen 9 6980HX

(JIRA:RHELPLAN-135602)

4.17. THE WEB CONSOLE

Update progress page in the web console now supports an automatic restart option

The update progress page now has a **Reboot after completion** switch. This reboots the system automatically after installing the updates.

(BZ#2056786)

4.18. RED HAT ENTERPRISE LINUX SYSTEM ROLES

The network RHEL system role supports network configuration using the nmstate API

With this update, the **network** RHEL system role supports network configuration through the **nmstate** API. Users can now directly apply the configuration of the required network state to a network interface instead of creating connection profiles. The feature also allows partial configuration of a network. As a result, the following benefits exist:

- decreased network configuration complexity
- reliable way to apply the network state changes
- no need to track the entire network configuration

(BZ#2072385)

Users can create connections with IPoIB capability using the network RHEL system role

The **infiniband** connection type of the **network** RHEL system role now supports the Internet Protocol over Infiniband (IPoIB) capability. To enable this feature, define a value to the **p_key** option of **infiniband**. Note that if you specify **p_key**, the **interface_name** option of the **network_connections** variable must be left unset. The previous implementation of the **network** RHEL system role did not properly validate the **p_key** value and the **interface_name** option for the **infiniband** connection type. Therefore, the IPoIB functionality never worked before. For more information, see a README file in the **/usr/share/doc/rhel-system-roles/network/** directory.

(BZ#2086965)

HA Cluster RHEL system role now supports SBD fencing and configuration of Corosync settings

The HA Cluster system role now supports the following features:

SBD fencing

Fencing is a crucial part of HA cluster configuration. SBD provides a means for nodes to reliably self-terminate when fencing is required. SBD fencing can be particularly useful in environments where traditional fencing mechanisms are not possible. It is now possible to configure SBD fencing with the HA Cluster system role.

Corosync settings

The HA Cluster system role now supports the configuration of Corosync settings, such as transport, compression, encryption, links, totem, and quorum. These settings are required to match cluster configuration with customers' needs and environment when the default settings are not suitable.

(BZ#2065337, BZ#2070452, BZ#2079626, BZ#2098212, BZ#2120709, BZ#2120712)

The network RHEL role now configures network settings for routing rules

Previously, you could route the packet based on the destination address field in the packet, but you could not define the source routing and other policy routing rules. With this enhancement, **network** RHEL role supports routing rules so that the users have control over the packet transmission or route selection.

(BZ#2079622)

The new previous:replaced configuration enables firewall system role to reset the firewall settings to default

System administrators who manage different sets of machines, where each machine has different pre-existing firewall settings, can now use the **previous: replaced** configuration in the **firewall** role to ensure that all machines have the same firewall configuration settings. The **previous: replaced** configuration can erase all the existing firewall settings and replace them with consistent settings.

(BZ#2043010)

New option in the postfix RHEL system role for overwriting previous configuration

If you manage a group of systems which have inconsistent **postfix** configurations, you may want to make the configuration consistent on all of them. With this enhancement, you can specify the **previous: replaced** option within the **postfix_conf** dictionary to remove any existing configuration and apply the desired configuration on top of a clean **postfix** installation. As a result, you can erase any existing **postfix** configuration and ensure consistency on all the systems being managed.

(BZ#2065383)

Enhanced microsoft.sql.server RHEL system role

The following new variables are now available for the **microsoft.sql.server** RHEL system role:

- Variables with the **mssql_ha_** prefix to control configuring a high availability cluster.
- The **mssql_tls_remote_src** variable to search for **mssql_tls_cert** and **mssql_tls_private_key** values on managed nodes. If you keep the default **false** setting, the role searches for these files on the control node.

- The **mssql_manage_firewall** variable to manage firewall ports automatically. If this variable is set to **false**, you must enable firewall ports manually.
- The **mssql_pre_input_sql_file** and **mssql_post_input_sql_file** variables to control whether you want to run the SQL scripts before the role execution or after it. These new variables replace the former **mssql_input_sql_file** variable, which did not allow you to influence the time of SQL script execution.

(BZ#2066337)

The logging RHEL system role supports options startmsg.regex and endmsg.regex in files inputs

With this enhancement, you can now filter log messages coming from files by using regular expressions. Options **startmsg_regex** and **endmsg_regex** are now included in the files' input. The **startmsg_regex** represents the regular expression that matches the start part of a message, and the **endmsg_regex** represents the regular expression that matches the last part of a message. As a result, you can now filter messages based upon properties such as date-time, priority, and severity.

(BZ#2112145)

The sshd RHEL system role verifies the include directive for the drop-in directory

The **sshd** RHEL system role on RHEL 9 manages only a file in the drop-in directory, but previously did not verify that the directory is included from the main **sshd_config** file. With this update, the role verifies that **sshd_config** contains the include directive for the drop-in directory. As a result, the role more reliably applies the provided configuration.

(BZ#2052081)

The sshd RHEL system role can be managed through /etc/ssh/sshd_config

The **sshd** RHEL system role applied to a RHEL 9 managed node places the SSHD configuration in a drop-in directory (**/etc/ssh/sshd_config.d/00-ansible_system_role.conf** by default). Previously, any changes to the **/etc/ssh/sshd_config** file overwrote the default values in **00-ansible_system_role.conf**. With this update, you can manage SSHD by using **/etc/ssh/sshd_config** instead of **00-ansible_system_role.conf** while preserving the system default values in **00-ansible_system_role.conf**.

(BZ#2052086)

The metrics role consistently uses "Ansible_managed" comment in its managed configuration files

With this update, the **metrics** role inserts the "Ansible managed" comment to the configuration files, using the Ansible standard **ansible_managed** variable. The comment indicates that the configuration files should not be directly edited because the **metrics** role can overwrite the file. As a result, the configuration files contain a declaration stating that the configuration files are managed by Ansible.

(BZ#2065392)

The storage RHEL system role now supports managing the pool members

The **storage** RHEL system role can now add or remove disks from existing LVM pools without removing the pool first. To increase the pool capacity, the **storage** RHEL system role can add new disks to the pool and free currently allocated disks in the pool for another use.

(BZ#2072742)

Support for thinly provisioned volumes is now available in the storage RHEL system role

The **storage** RHEL system role can now create and manage thinly provisioned LVM logical volumes (LVs). Thin provisioned LVs are allocated as they are written, allowing better flexibility when creating volumes as physical storage provided for thin provisioned LVs can be increased later as the need arises. LVM thin provisioning also allows creating more efficient snapshots because the data blocks common to a thin LV and any of its snapshots are shared.

(BZ#2072745)

Better support for cached volumes is available in the storage RHEL system role

The **storage** RHEL system role can now attach cache to existing LVM logical volumes. LVM cache can be used to improve performance of slower logical volumes by temporarily storing subsets of an LV's data on a smaller, faster device, for example an SSD. This enhances the previously added support for creating cached volumes by allowing adding (attaching) a cache to an existing, previously uncached volume.

(BZ#2072746)

The logging RHEL system role now supports template, severity and facility options

The **logging** RHEL system role now features new useful **severity** and **facility** options to the files inputs as well as a new **template** option to the files and forwards outputs. Use the **template** option to specify the traditional time format by using the parameter **traditional**, the syslog protocol 23 format by using the parameter **syslog**, and the modern style format by using the parameter **modern**. As a result, you can now use the **logging** role to filter by the severity and facility as well as to specify the output format by template.

(BZ#2075119)

RHEL system roles now available also in playbooks with fact gathering disabled

Ansible fact gathering might be disabled in your environment for performance or other reasons. Previously, it was not possible to use RHEL system roles in such configurations. With this update, the system detects the **ANSIBLE_GATHERING=explicit** parameter in your configuration and **gather_facts: false** parameter in your playbooks, and use the **setup:** module to gather only the facts required by the given role, if not available from the fact cache.

NOTE

If you have disabled Ansible fact gathering due to performance, you can enable Ansible fact caching instead, which does not cause a performance hit of retrieving them from source.

(BZ#2078989)

The storage role now has less verbosity by default

The storage role output is now less verbose by default. With this update, users can increase the verbosity of storage role output to only produce debugging output if they are using Ansible verbosity level 1 or above.

(BZ#2079627)

The firewall RHEL system role does not require the state parameter when configuring masquerade or icmp_block_inversion

When configuring custom firewall zones, variables **masquerade** and **icmp_block_inversion** are boolean settings. A value of **true** implies **state: present** and a value of **false** implies **state: absent**. Therefore, the **state** parameter is not required when configuring **masquerade** or **icmp_block_inversion**.

(BZ#2093423)

You can now add, update, or remove services using absent and present states in the firewall RHEL system role

With this enhancement, you can use the **present** state to add ports, modules, protocols, services, and destination addresses, or use the **absent** state to remove them. Note that to use the **absent** and **present** states in the **firewall** RHEL system role, set the **permanent** option to **true**. With the **permanent** option set to **true**, the state settings apply until changed, and remain unaffected by role reloads.

(BZ#2100292)

The firewall system role can add or remove an interface to the zone using PCI device ID

Using the PCI device ID, the **firewall** system role can now assign or remove a network interface to or from a zone. Previously, if only the PCI device ID was known instead of the interface name, users had to first identify the corresponding interface name to use the **firewall** system role. With this update, the **firewall** system role can now use the PCI device ID to manage a network interface in a zone.

(BZ#2100942)

The firewall RHEL system role can provide Ansible facts

With this enhancement, you can now gather the **firewall** RHEL system role's Ansible facts from all of your systems by including the **firewall:** variable in the playbook with no arguments. To gather a more detailed version of the Ansible facts, use the **detailed: true** argument, for example:

```
vars:
  firewall:
    detailed: true
```

(BZ#2115154)

Added setting of seuser and selevel to the selinux RHEL system role

Sometimes, it is necessary to set **seuser** and **selevel** parameters when setting SELinux context file system mappings. With this update, you can use the **seuser** and **selevel** optional arguments in **selinux_fcontext** to specify SELinux user and level in the SELinux context file system mappings.

(BZ#2115157)

New cockpit system role variable for setting a custom listening port

The **cockpit** system role introduces the **cockpit_port** variable that allows you to set a custom listening port other than the default 9090 port. Note that if you decide to set a custom listening port, you will also need to adjust your SELinux policy to allow the web console to listen on that port.

(BZ#2115152)

The metrics role can export postfix performance data

You can now use the new **metrics_from_postfix** boolean variable in the **metrics** role for recording and detailed performance analysis. With this enhancement, setting the variable enables the **pmdapostfix** metrics agent on the system, making statistics about **postfix** available.

(BZ#2051737)

The postfix role consistently uses "Ansible_managed" comment in its managed configuration files

The **postfix** role generates the **/etc/postfix/main.cf** configuration file. With this update, the **postfix** role inserts the "Ansible managed" comment to the configuration files, using the Ansible standard **ansible_managed** variable. The comment indicates that the configuration files should not be directly edited because the **postfix** role can overwrite the file. As a result, the configuration files contain a declaration stating that the configuration files are managed by Ansible.

(BZ#2065393)

The nbde-client RHEL system role supports static IP addresses

In previous versions of RHEL, restarting a system with a static IP address and configured with the **nbde_client** RHEL system role changed the system's IP address. With this update, systems with static IP addresses are supported by the **nbde_client** role, and their IP addresses do not change after a reboot.

Note that by default, the **nbde_client** role uses DHCP when booting, and switches to the configured static IP after the system is booted.

(BZ#2070462)

4.19. VIRTUALIZATION

RHEL web console now features RHEL as an option for the Download an OS VM workflow

With this enhancement, the RHEL web console now supports the installation of RHEL virtual machines (VMs) using the default **Download an OS** workflow. As a result, you can download and install the RHEL OS as a VM directly within the web console.

(JIRA:RHELPLAN-121982)

Improved KVM architectural compliance

With this update, the architectural compliance of the KVM hypervisor has now been enhanced and made stricter. As a result, the hypervisor is now better prepared to address future changes to Linux-based and other operating systems.

(JIRA:RHELPLAN-117713)

ap-check is now available in RHEL 9

The **mdevctl** tool now provides a new **ap-check** support utility. You can use **mdevctl** to persistently configure cryptographic adapters and domains that are allowed for pass-through usage into virtual machines as well as the **matrix** and **vfio-ap** devices. With **mdevctl**, you do not have to reconfigure these adapters, domains, and devices after every IPL. In addition, **mdevctl** prevents the distributor from inventing other ways to reconfigure them.

When invoking **mdevctl** commands for **vfio-ap** devices, the new **ap-check** support utility is invoked as part of the **mdevctl** command to perform additional validity checks against **vfio-ap** device configurations.

In addition, the **chzdev** tool now provides the ability to manage the system-wide Adjunct Processor (AP) mask settings, which determine what AP resources are available for **vfio-ap** devices. When used, **chzdev** makes it possible to persist these settings by generating an associated **udev** rule. Using **lszdev**,

you can can now also query the system-wide AP mask settings.

(BZ#1870699)

open-vm-tools rebased to 12.0.5

The **open-vm-tools** packages have been upgraded to version 12.0.5, which introduces a number of bug fixes and new features. Most notably, support has been added for the Salt Minion tool to be managed through guest OS variables.

(BZ#2061193)

Selected VMs on IBM Z can now boot with kernel command lines longer than 896 bytes

Previously, booting a virtual machine (VM) on a RHEL 9 IBM Z host always failed if the kernel command line of the VM was longer than 896 bytes. With this update, the QEMU emulator can handle kernel command lines longer than 896 bytes. As a result, you can now use QEMU direct kernel boot for VMs with very long kernel command lines, if the VM kernel supports it. Specifically, to use a command line longer than 896 bytes, the VM must use Linux kernel version 5.16-rc1 or later.

(BZ#2044218)

The Secure Execution feature on IBM Z now supports remote attestation

The Secure Execution feature on the IBM Z architecture now supports remote attestation. The **pvattest** utility can create a remote attestation request to verify the integrity of a guest that has Secure Execution enabled.

Additionally, it is now possible to inject interrupts to guests with Secure Execution through the use of GISA.

(BZ#2001936, BZ#2044300)

VM memory preallocation using multiple threads

You can now define multiple CPU threads for virtual machine (VM) memory allocation in the domain XML configuration, for example as follows:

```
<memoryBacking>
  <allocation threads='8'/>
</memoryBacking>
```

This ensures that more than one thread is used for allocating memory pages when starting a VM. As a result, VMs with multiple allocation threads configured start significantly faster, especially if the VMs has large amounts of RAM assigned and backed by hugepages.

(BZ#2064194)

RHEL 9 guests now support SEV-SNP

On virtual machines (VMs) that use RHEL 9 as a guest operating system, you can now use AMD Secure Encrypted Virtualization (SEV) with the Secure Nested Paging (SNP) feature. Among other benefits, SNP enhances SEV by improving its memory integrity protection, which helps prevent hypervisor-based attacks such as data replay or memory re-mapping. Note that for SEV-SNP to work on a RHEL 9 VM, the host running the VM must support SEV-SNP as well.

(BZ#2169738)

4.20. RHEL IN CLOUD ENVIRONMENTS

New SSH module for cloud-init

With this update, an SSH module has been added to the **cloud-init** utility, which automatically generates host keys during instance creation.

Note that with this change, the default **cloud-init** configuration has been updated. Therefore, if you had a local modification, make sure the /etc/cloud/cloud.cfg contains "ssh_genkeytypes: ['rsa', 'ecdsa', 'ed25519']" line.

Otherwise, **cloud-init** creates an image which fails to start the **sshd** service. If this occurs, do the following to work around the problem:

1. Make sure the **/etc/cloud/cloud.cfg** file contains the following line:

```
ssh_genkeytypes:  ['rsa', 'ecdsa', 'ed25519']
```

2. Check whether **/etc/ssh/ssh_host_*** files exist in the instance.
3. If the **/etc/ssh/ssh_host_*** files do not exist, use the following command to generate host keys:

```
cloud-init single --name cc_ssh
```

4. Restart the sshd service:

```
systemctl restart sshd
```

(BZ#2115791)

4.21. CONTAINERS

The Container Tools packages have been updated

The Container Tools packages which contain the Podman, Buildah, Skopeo, crun, and runc tools are now available. This update provides a list of bug fixes and enhancements over the previous version.

Notable changes include:

- The **podman pod create** command now supports setting the CPU and memory limits. You can set a limit for all containers in the pod, while individual containers within the pod can have their own limits.
- The **podman pod clone** command creates a copy of an existing pod.
- The **podman play kube** command now supports the security context settings using the **BlockDevice** and **CharDevice** volumes.
- Pods created by the **podman play kube** can now be managed by systemd unit files using a **podman-kube@<service>.service** (for example **systemctl --user start podman-play-kube@$(systemd-escape my.yaml).service**).
- The **podman push** and **podman push manifest** commands now support the sigstore signatures.

- The Podman networks can now be isolated by using the **podman network --opt isolate** command.

Podman has been upgraded to version 4.2, for further information about notable changes, see the upstream release notes.

(JIRA:RHELPLAN-118462)

GitLab Runner is now available on RHEL using Podman

Beginning with GitLab Runner 15.1, you can use Podman as the container runtime in the GitLab Runner Docker Executor. For more details, see GitLab's Release Note .

(JIRA:RHELPLAN-101140)

Podman now supports the --health-on-failure option

The **podman run** and **podman create** commands now support the **--health-on-failure** option to determine the actions to be performed when the status of a container becomes unhealthy.

The **--health-on-failure** option supports four actions:

- **none**: Take no action, this is the default action.
- **kill**: Kill the container.
- **restart**: Restart the container.
- **stop**: Stop the container.

NOTE

Do not combine the **restart** action with the **--restart** option. When running inside of a systemd unit, consider using the **kill** or **stop** action instead to make use of systemd's restart policy.

(BZ#2097708)

Netavark network stack is now available

The Netavark stack is a network configuration tool for containers. In RHEL 9, the Netavark stack is fully supported and enabled by default.

This network stack has the following capabilities:

- Configuration of container networks using the JSON configuration file
- Creating, managing, and removing network interfaces, including bridge and MACVLAN interfaces
- Configuring firewall settings, such as network address translation (NAT) and port mapping rules
- IPv4 and IPv6
- Improved capability for containers in multiple networks
- Container DNS resolution using the aardvark-dns project

NOTE

You have to use the same version of Netavark stack and the **aardvark-dns** authoritative DNS server.

(JIRA:RHELPLAN-132023)

New package: catatonit in the CRB repository

A new **catatonit** package is now available in the CodeReady Linux Builder (CRB) repository. The **catatonit** package is used as a minimal init program for containers and can be included within the application container image. Note that packages included in the CodeReady Linux Builder repository are unsupported.

Note that since RHEL 9.0, the **podman-catonit** package is available in the AppStream repository. The **podman-catatonit** package is used only by the Podman tool.

(BZ#2074193)

CHAPTER 5. 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 9.1. 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

allow_mismatched_32bit_el0 = [ARM64]

With this parameter you can allow systems with mismatched 32-bit support at the EL0 level to run 32-bit applications. The set of CPUs supporting 32-bit EL0 is indicated by the **/sys/devices/system/cpu/aarch32_el0** file. Also, you can restrict hot-unplug operations.
For more information, see **Documentation/arm64/asymmetric-32bit.rst**.

arm64.nomte = [ARM64]

With this parameter you can unconditionally disable Memory Tagging Extension (MTE) support.

i8042.probe_defer = [HW]

With this parameter you can allow deferred probing on **i8042** probe errors.

idxd.tc_override = [HW]

With this parameter in the **<bool>** format, you can allow override of default traffic class configuration for the device.
The default value is set to **false** (**0**).

kvm.eager_page_split = [KVM,X86]

With this parameter you can control whether or not a KVM proactively splits all huge pages during dirty logging. Eager page splitting reduces interruptions to vCPU execution by eliminating the write-protection faults and Memory Management Unit (MMU) lock contention that is otherwise required to split huge pages lazily.
VM workloads that rarely perform writes or that write only to a small region of VM memory can benefit from disabling eager page splitting to allow huge pages to still be used for reads.

The behavior of eager page splitting depends on whether the **KVM_DIRTY_LOG_INITIALLY_SET** option is enabled or disabled.

- If disabled, all huge pages in a **memslot** are eagerly split when dirty logging is enabled on that **memslot**.
- If enabled, eager page splitting is performed during the **KVM_CLEAR_DIRTY ioctl()** system call, and only for the pages being cleared.
  Eager page splitting currently only supports splitting huge pages mapped by the two dimensional paging (TDP) MMU.

  The default value is set to **Y** (**on**).

kvm.nx_huge_pages_recovery_period_ms = [KVM]

With this parameter you can control the time period at which KVM zaps 4 KiB pages back to huge pages.

- If the value is a non-zero **N**, KVM zaps a portion of the pages every **N** milliseconds.

- If the value is **0**, KVM picks a period based on the ratio, such that a page is zapped after 1 hour on average.
  The default value is set to **0**.

l1d_flush = [X86,INTEL]

With this parameter you can control mitigation for L1D-based snooping vulnerability.
Certain CPUs are vulnerable to an exploit against CPU internal buffers which can, under certain conditions, forward information to a disclosure gadget. In vulnerable processors, the speculatively forwarded data can be used in a cache side channel attack, to access data to which the attacker does not have direct access.

The available option is **on**, which means **enable the interface for the mitigation**.

mmio_stale_data = [X86,INTEL]

With this parameter you can control mitigation for the Processor Memory-mapped I/O (MMIO) Stale Data vulnerabilities.
Processor MMIO Stale Data is a class of vulnerabilities that can expose data after an MMIO operation. Exposed data could originate or end in the same CPU buffers as affected by metadata server (MDS) and Transactional Asynchronous Abort (TAA). Therefore, similar to MDS and TAA, the mitigation is to clear the affected CPU buffers.

The available options are:

- **full**: enable mitigation on vulnerable CPUs
- **full,nosmt**: enable mitigation and disable SMT on vulnerable CPUs.
- **off**: unconditionally disable mitigation
  On MDS or TAA affected machines, **mmio_stale_data=off** can be prevented by an active MDS or TAA mitigation as these vulnerabilities are mitigated with the same mechanism. Thus, in order to disable this mitigation, you need to specify **mds=off** and **tsx_async_abort=off**, too.

  Not specifying this option is equivalent to **mmio_stale_data=full**.

  For more information, see **Documentation/admin-guide/hw-vuln/processor_mmio_stale_data.rst**.

random.trust_bootloader={on,off} = [KNL]

With this parameter you can enable or disable trusting the use of a seed passed by the boot loader (if available) to fully seed the kernel's CRNG. The default behavior is controlled by the **CONFIG_RANDOM_TRUST_BOOTLOADER** option.

rcupdate.rcu_task_collapse_lim = [KNL]

With this parameter you can set the maximum number of callbacks present at the beginning of a grace period that allows the RCU Tasks flavors to collapse back to using a single callback queue. This switching only occurs when the **rcupdate.rcu_task_enqueue_lim** option is set to the default value of **-1**.

rcupdate.rcu_task_contend_lim = [KNL]

With this parameter you can set the minimum number of callback-queuing-time lock-contention events per jiffy required to cause the RCU Tasks flavors to switch to per-CPU callback queuing. This switching only occurs when the **rcupdate.rcu_task_enqueue_lim** option is set to the default value of **-1**.

rcupdate.rcu_task_enqueue_lim = [KNL]

With this parameter you can set the number of callback queues to use for the RCU Tasks family of RCU flavors. You can adjust the number of callback queues automatically and dynamically with the default value of **-1**.

This parameter is intended for use in testing.

retbleed = [X86]

With this parameter you can control mitigation of Arbitrary Speculative Code Execution with Return Instructions (RETBleed) vulnerability. The available options are:

- **off**: no mitigation
- **auto**: automatically select a mitigation
- **auto,nosmt**: automatically select a mitigation, disabling SMT if necessary for the full mitigation (only on Zen1 and older without STIBP).
- **ibpb**: mitigate short speculation windows on basic block boundaries too. Safe, highest performance impact.
- **unret**: force enable untrained return thunks, only effective on AMD f15h-f17h based systems.
- **unret,nosmt**: like the **unret** option, will disable SMT when STIBP is not available. Selecting the **auto** option chooses a mitigation method at run time according to the CPU.

  Not specifying this option is equivalent to **retbleed=auto**.

sev=option[,option...] = [X86-64]

For more information, see **Documentation/x86/x86_64/boot-options.rst**.

Updated kernel parameters

acpi_sleep = [HW,ACPI]

Format: { s3_bios, s3_mode, s3_beep, s4_hwsig, s4_nohwsig, old_ordering, nonvs, sci_force_enable, nobl }

- For more information on **s3_bios** and **s3_mode**, see **Documentation/power/video.rst**.
- **s3_beep** is for debugging; it makes the PC's speaker beep as soon as the kernel real-mode entry point is called.
- **s4_hwsig** causes the kernel to check the ACPI hardware signature during resume from hibernation, and gracefully refuse to resume if it has changed. The default behavior is to allow resume and simply warn when the signature changes, unless the **s4_hwsig** option is enabled.
- **s4_nohwsig** prevents ACPI hardware signature from being used, or even warned about, during resume. **old_ordering** causes the ACPI 1.0 ordering of the **_PTS** control method, with respect to putting devices into low power states, to be enforced. The ACPI 2.0 ordering of **_PTS** is used by default.
- **nonvs** prevents the kernel from saving and restoring the ACPI NVS memory during suspend, hibernation, and resume.
- **sci_force_enable** causes the kernel to set **SCI_EN** directly on resume from S1/S3. Even though this behavior is contrary to the ACPI specifications, some corrupted systems do not work without it.

- **nobl** causes the internal denylist of systems known to behave incorrectly in some ways with respect to system suspend and resume to be ignored. Use this option wisely.
  For more information, see **Documentation/power/video.rst**.

crashkernel=size[KMG],high = [KNL, X86-64, ARM64]

With this parameter you can allocate physical memory region from top as follows:

- If the system has more than 4 GB RAM installed, a physical memory region can exceed 4 GB.
- If the system has less than 4 GB RAM installed, a physical memory region will be allocated below 4 GB, if available.
  This parameter is ignored if the **crashkernel=X** parameter is specified.

crashkernel=size[KMG],low = [KNL, X86-64]

When you pass **crashkernel=X,high**, the kernel can allocate a physical memory region above 4 GB. This causes the second kernel crash on systems that require some amount of low memory (for example, **swiotlb** requires at least 64M+32K low memory) and enough extra low memory to make sure DMA buffers for 32-bit devices are not exhausted. Kernel tries to allocate at least 256 M below 4 GB automatically. With this parameter you can specify the low range under 4 GB for the second kernel instead.

- **0:** disables low allocation. It will be ignored when **crashkernel=X,high** is not used or memory reserved is below 4 GB.

crashkernel=size[KMG],low = [KNL, ARM64]

With this parameter you can specify a low range in the DMA zone for the crash dump kernel. It will be ignored when **crashkernel=X,high** is not used or memory reserved is located in the DMA zones.

kvm.nx_huge_pages_recovery_ratio = [KVM]

With this parameter you can control how many 4 KiB pages are periodically zapped back to huge pages:

- **0** disables the recovery
- **N** KVM will zap **1**/**Nth** of the 4 KiB pages every period.
  The default is set to **60**.

kvm-arm.mode = [KVM,ARM]

With this parameter you can select one of KVM modes of operation:

- **none**: forcefully disable KVM.
- **nvhe**: standard nVHE-based mode, without support for protected guests.
- **protected**: **nVHE**-based mode with support for guests whose state is kept private from the host. Not valid if the kernel is running in the EL2 level.
  The default value is set to **VHE**/**nVHE** based on hardware support.

mitigations = [X86,PPC,S390,ARM64]

With this parameter you can control optional mitigations for CPU vulnerabilities. This is a set of curated, arch-independent options, each of which is an aggregation of existing arch-specific options:

- **off**: disable all optional CPU mitigations. This improves system performance, but it may also expose users to several CPU vulnerabilities.

    - Equivalent to: **nopti [X86,PPC]**, **kpti=0 [ARM64]**, **nospectre_v1 [X86,PPC]**, **nobp=0 [S390]**, **nospectre_v2 [X86,PPC,S390,ARM64]**, **spectre_v2_user=off [X86]**, **spec_store_bypass_disable=off [X86,PPC]**, **ssbd=force-off [ARM64]**, **l1tf=off [X86]**, **mds=off [X86]**, **tsx_async_abort=off [X86]**, **kvm.nx_huge_pages=off [X86]**, **no_entry_flush [PPC]**, **no_uaccess_flush [PPC]**, **mmio_stale_data=off [X86]**.

    - Exceptions: This does not have any effect on **kvm.nx_huge_pages** when the **kvm.nx_huge_pages=force** option is specified.

- **auto** (default): mitigate all CPU vulnerabilities, but leave SMT enabled, even if it is vulnerable.

    - Equivalent to: (default behavior)

- **auto,nosmt**: mitigate all CPU vulnerabilities, disabling SMT if needed.

    - Equivalent to: **l1tf=flush,nosmt [X86]**, **mds=full,nosmt [X86]**, **tsx_async_abort=full,nosmt [X86]**, **mmio_stale_data=full,nosmt [X86]**

rcu_nocbs[=cpu-list] = [KNL]

The optional argument is a CPU list.
In kernels built with **CONFIG_RCU_NOCB_CPU=y**, you can enable the no-callback CPU mode, which prevents such CPUs callbacks from being invoked in softirq context. Invocation of such CPUs' RCU callbacks will instead be offloaded to **rcuox/N kthreads** created for that purpose, where **x** is **p** for RCU-preempt, **s** for RCU-sched, and **g** for the **kthreads** that mediate grace periods; and **N** is the CPU number. This reduces OS jitter on the offloaded CPUs, which can be useful for HPC and real-time workloads. It can also improve energy efficiency for asymmetric multiprocessors.

- If a **cpulist** is passed as an argument, the specified list of CPUs is set to no-callback mode from boot.

- If the **=** sign and the **cpulist** arguments are omitted, no CPU will be set to no-callback mode from boot but you can toggle the mode at runtime using **cpusets**.

rcutree.kthread_prio = [KNL,BOOT]

With this parameter you can set the **SCHED_FIFO** priority of the RCU per-CPU **kthreads** (**rcuc/N**). This value is also used for the priority of the RCU boost threads (**rcub/N**) and for the RCU grace-period **kthreads** (**rcu_bh**, **rcu_preempt**, and **rcu_sched**).

- If **RCU_BOOST** is set, valid values are 1-99 and the default is **1**, the least-favored priority.

- If **RCU_BOOST** is not set, valid values are 0-99 and the default is **0**, non-realtime operation. When **RCU_NOCB_CPU** is set, you should adjust the priority of **NOCB** callback **kthreads**.

rcutorture.fwd_progress = [KNL]

With this parameter you can specify the number of **kthreads** to be used for RCU grace-period forward-progress testing for the types of RCU supporting this notion.
The default is set to **1 kthread**. Values less than zero or greater than the number of CPUs cause the number of CPUs to be used.

spectre_v2 = [X86]

With this parameter you can control mitigation of Spectre variant 2 (indirect branch speculation) vulnerability. The default operation protects the kernel from user space attacks.

- **on**: unconditionally enable, implies **spectre_v2_user=on**

- **off**: unconditionally disable, implies **spectre_v2_user=off**
- **auto**: kernel detects whether your CPU model is vulnerable
- Selecting **on** will, and **auto** may, choose a mitigation method at run time according to the CPU, the available microcode, the setting of the **CONFIG_RETPOLINE** configuration option, and the compiler with which the kernel was built.
- Selecting **on** will also enable the mitigation against user space to user space task attacks.
- Selecting **off** will disable both the kernel and the user space protections.
- Specific mitigations can also be selected manually:
  - **retpoline**: replace indirect branches
  - **retpoline,generic**: Retpolines
  - **retpoline,lfence**: LFENCE; indirect branch
  - **retpoline,amd**: alias for retpoline,lfence
  - **eibrs**: enhanced IBRS
  - **eibrs,retpoline**: enhanced IBRS + Retpolines
  - **eibrs,lfence**: enhanced IBRS + LFENCE
  - **ibrs**: use IBRS to protect kernel
    Not specifying this option is equivalent to **spectre_v2=auto**.

New sysctl parameters

max_rcu_stall_to_panic

When you set **panic_on_rcu_stall** to **1**, you determine the number of times that RCU can stall before **panic()** is called. When you set **panic_on_rcu_stall** to **0**, this value has no effect.

perf_user_access = [ARM64]

With this parameter you can control user space access for reading **perf** event counters.

- When set to **1**, user space can read performance monitor counter registers directly.
- The default is set to **0**, which means **access disabled**.
  For more information, see **Documentation/arm64/perf.rst**.

gro_normal_batch

With this parameter you can set the maximum number of the segments to batch up on output of GRO. When a packet exits GRO, either as a coalesced superframe or as an original packet which GRO has decided not to coalesce, it is placed on a per-NAPI list. This list is then passed to the stack when the number of segments reaches the **gro_normal_batch** limit.

high_order_alloc_disable

With this parameter you can choose order-0 allocation. By default, the allocator for page fragments tries to use high order pages, that is order-3 on X86 systems. While the default behavior returns good results, in certain situations a contention in page allocations and freeing occurs. This was

especially true on older kernels (version 5.14 and higher) when high-order pages were not stored on per-CPU lists. This parameter exists now mostly of historical importance.
The default value is **0**.

page_lock_unfairness

By specifying the value for this parameter you can determine the number of times that the page lock can be stolen from under a waiter. After the lock is stolen the number of times specified in this file, the **fair lock handoff** semantics will apply, and the waiter will only be awakened if the lock can be taken.
The default value is **5**.

Changed sysctl parameters

urandom_min_reseed_secs

You can use this parameter to determine the minimum number of seconds between **urandom** pool reseeding. This file is writable for compatibility purposes, but writing to it has no effect on any RNG behavior.

write_wakeup_threshold

When the entropy count sinks below this threshold in a number of bits, you can wake up processes waiting to write to the /**dev**/**random** file. This file is writable for compatibility purposes, but writing to it has no effect on any RNG behavior.

CHAPTER 6. DEVICE DRIVERS

6.1. NEW DRIVERS

Network drivers

- Platform Firmware Runtime Update Telemetry driver (**pfr_telemetry**)
- Platform Firmware Runtime Update device driver (**pfr_update**)
- Bluetooth support for MediaTek devices ver 0.1 (**btmtk**)
- MHI Host Interface (**mhi**)
- Modem Host Interface (MHI) PCI controller driver (**mhi_pci_generic**)
- IDXD driver dsa_bus_type driver (**idxd_bus**)
- AMD PassThru DMA driver (**ptdma**)
- Mellanox FAN driver (**mlxreg-fan**)
- Mellanox LED regmap driver (**leds-mlxreg**)
- Intel® LPSS ACPI driver (**intel-lpss-acpi**)
- Intel® LPSS PCI driver (**intel-lpss-pci**)
- Intel® LPSS core driver (**intel-lpss**)
- Maxlinear Ethernet GPY Driver (**mxl-gpy**)
- Realtek 802.11ax wireless 8852A driver (**rtw89_8852a**)
- Realtek 802.11ax wireless 8852AE driver (**rtw89_8852ae**)
- Intel® PMT Class driver (**pmt_class**)
- Intel® PMT Crashlog driver (**pmt_crashlog**)
- Intel® PMT Telemetry driver (**pmt_telemetry**)
- Intel® speed select interface mailbox driver (**isst_if_mbox_msr**)
- Intel® speed select interface pci mailbox driver (**isst_if_mbox_pci**)
- Intel® speed select interface mmio driver (**isst_if_mmio**)
- Intel® Software Defined Silicon driver (**intel_sdsi**)
- Intel® Extended Capabilities auxiliary bus driver (**intel_vsec**)
- ISH ISHTP eclite client opregion driver (**ishtp_eclite**)
- Acer Wireless Radio Control Driver (**acer-wireless**)
- AMD HSMP Platform Interface Driver (**amd_hsmp**)

- DESIGNWARE HS OTG Core (**dwc2**)
- Synopsys HAPS PCI Glue Layer (**dwc3-haps**)
- DesignWare USB3 PCI Glue Layer (**dwc3-pci**)
- DesignWare USB3 DRD Controller Driver (**dwc3**)
- xHCI Platform Host Controller Driver (**xhci-plat-hcd**)
- ON Semiconductor FSA4480 driver (**fsa4480**)
- Richtek RT1719 Sink Only USBPD Controller Driver (**rt1719**)
- Willsemi WUSB3801 Type-C port controller driver (**wusb3801**)
- Core driver for VFIO based PCI devices (**vfio-pci-core**)
- AMD SEV Guest Driver (**sev-guest**)
- Mellanox watchdog driver (**mlx_wdt**)

Graphics drivers and miscellaneous drivers

- Cirrus Logic DSP Support (**cs_dsp**)
- DRM DisplayPort helper (**drm_dp_helper**)
- DRM Buddy Allocator (**drm_buddy**)
- DRM SHMEM memory-management helpers (**drm_shmem_helper**)
- DRM driver using bochs dispi interface (**bochs**)
- Letsketch tablet driver (**hid-letsketch**)
- Intel® speed select interface driver (**isst_if_common**)
- SiGma Micro HID driver (**hid-sigmamicro**)
- Fixing side buttons of Xiaomi Mi Silent Mouse (**hid-xiaomi**)
- Driver for DEC VSXXX-AA and -GA mice and VSXXX-AB tablet (**vsxxxaa**)
- Nvidia line card platform driver (**mlxreg-lc**)
- Intel PCH Thermal driver (**intel_pch_thermal**)
- Intel LPSS UART driver (**8250_lpss**)

6.2. UPDATED DRIVERS

Network driver updates

- VMware vmxnet3 virtual NIC driver (**vmxnet3**) has been updated to version 1.7.0.0-k.

Storage driver updates

- Emulex LightPulse Fibre Channel SCSI driver (**lpfc**) has been updated to version 14.2.0.5.
- MPI3 Storage Controller Device Driver (**mpi3mr**) has been updated to version 8.0.0.69.0.
- LSI MPT Fusion SAS 3.0 Device Driver (**mpt3sas**) has been updated to version 40.100.00.00.
- Driver for Microchip Smart Family Controller (**smartpqi**) has been updated to version 2.1.18-045.

Graphics and miscellaneous driver updates

- Standalone drm driver for the VMware SVGA device (**vmwgfx**) has been updated to version 2.20.0.0.

CHAPTER 7. 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 9. 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 7.1. System configuration and other options

Option	Value
unprivileged_bpf_disabled	2 (bpf() syscall restricted to privileged users, admin can change)
JIT compiler	1 (enabled)
JIT compiler hardening	1 (enabled for unprivileged users)
JIT compiler kallsyms exports	1 (enabled for root)
Memory limit for JIT for unprivileged users	264241152
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	y
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	n
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	1000
bpf() syscall	available
Large program size limit	available

Table 7.2. 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_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_spin_lock, bpf_spin_unlock, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_jiffies64, 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
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_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_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

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_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_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_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

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_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_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_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
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_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_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

Program type	Available helpers
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_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_tcp_gen_syncookie, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_jiffies64, 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
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_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_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

Program type	Available helpers
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_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_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_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
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_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
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_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_spin_lock, bpf_spin_unlock, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_jiffies64, 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

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_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_spin_lock, bpf_spin_unlock, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_jiffies64, 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
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_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_spin_lock, bpf_spin_unlock, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_jiffies64, 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

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_local_storage, bpf_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_spin_lock, bpf_spin_unlock, bpf_tcp_sock, 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_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
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_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_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_jiffies64, 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
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_uid_gid, 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_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_jiffies64, 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

Program type	Available helpers
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_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
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_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_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

Program type	Available helpers
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_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
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_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_spin_lock, bpf_spin_unlock, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_jiffies64, 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
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_sk_select_reuseport, bpf_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_spin_lock, bpf_spin_unlock, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_jiffies64, 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

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_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_spin_lock, bpf_spin_unlock, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_jiffies64, 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
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_uid_gid, 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_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
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_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_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

Program type	Available helpers
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_uid_gid, 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_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_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
tracing	not supported

Program type	Available helpers
struct_ops	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_skb_store_bytes, bpf_l3_csum_replace, bpf_l4_csum_replace, bpf_tail_call, bpf_clone_redirect, bpf_get_current_pid_tgid, bpf_get_current_uid_gid, bpf_get_current_comm, bpf_get_cgroup_classid, bpf_skb_vlan_push, bpf_skb_vlan_pop, bpf_skb_get_tunnel_key, bpf_skb_set_tunnel_key, bpf_perf_event_read, bpf_redirect, bpf_get_route_realm, bpf_perf_event_output, bpf_skb_load_bytes, bpf_get_stackid, 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_current_task_under_cgroup, 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_xdp_adjust_head, bpf_probe_read_str, bpf_get_socket_cookie, bpf_get_socket_uid, bpf_set_hash, bpf_setsockopt, bpf_skb_adjust_room, bpf_redirect_map, bpf_sk_redirect_map, bpf_sock_map_update, bpf_xdp_adjust_meta, bpf_perf_event_read_value, bpf_perf_prog_read_value, bpf_getsockopt, bpf_override_return, bpf_sock_ops_cb_flags_set, bpf_msg_redirect_map, bpf_msg_apply_bytes, bpf_msg_cork_bytes, bpf_msg_pull_data, bpf_bind, bpf_xdp_adjust_tail, bpf_skb_get_xfrm_state, bpf_get_stack, bpf_skb_load_bytes_relative, bpf_fib_lookup, bpf_sock_hash_update, bpf_msg_redirect_hash, bpf_sk_redirect_hash, bpf_lwt_push_encap, bpf_lwt_seg6_store_bytes, bpf_lwt_seg6_adjust_srh, bpf_lwt_seg6_action, bpf_rc_repeat, bpf_rc_keydown, bpf_skb_cgroup_id, bpf_get_current_cgroup_id, bpf_get_local_storage, bpf_sk_select_reuseport, 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_msg_push_data, bpf_msg_pop_data, bpf_rc_pointer_rel, 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_sysctl_get_name, bpf_sysctl_get_current_value, bpf_sysctl_get_new_value, bpf_sysctl_set_new_value, bpf_strtol, bpf_strtoul, bpf_sk_storage_get, bpf_sk_storage_delete, bpf_send_signal, bpf_tcp_gen_syncookie, bpf_skb_output, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_tcp_send_ack, bpf_send_signal_thread, bpf_jiffies64, bpf_read_branch_records, bpf_get_ns_current_pid_tgid, bpf_xdp_output, bpf_get_netns_cookie, bpf_get_current_ancestor_cgroup_id, bpf_sk_assign, bpf_ktime_get_boot_ns, bpf_seq_printf, bpf_seq_write, 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_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_get_task_stack, bpf_load_hdr_opt, bpf_store_hdr_opt, bpf_reserve_hdr_opt, bpf_inode_storage_get, bpf_inode_storage_delete, bpf_d_path, bpf_copy_from_user, bpf_snprintf_btf, bpf_seq_printf_btf, bpf_skb_cgroup_classid, bpf_redirect_neigh, bpf_per_cpu_ptr, bpf_this_cpu_ptr, bpf_redirect_peer, bpf_task_storage_get, bpf_task_storage_delete, bpf_get_current_task_btf, bpf_bprm_opts_set, bpf_ktime_get_coarse_ns, bpf_ima_inode_hash, bpf_sock_from_file, bpf_check_mtu, 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_get_attach_cookie, bpf_task_pt_regs, bpf_get_branch_snapshot, bpf_skc_to_unix_sock, bpf_kallsyms_lookup_name

Program type	Available helpers
ext	not supported
lsm	not supported
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_sk_release, bpf_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_spin_lock, bpf_spin_unlock, bpf_probe_read_user, bpf_probe_read_kernel, bpf_probe_read_user_str, bpf_probe_read_kernel_str, bpf_jiffies64, 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

Table 7.3. 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

Map type	Available
hash_of_maps	yes
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	no
ringbuf	yes
inode_storage	yes
task_storage	yes

CHAPTER 8. BUG FIXES

This part describes bugs fixed in Red Hat Enterprise Linux 9.1 that have a significant impact on users.

8.1. INSTALLER AND IMAGE CREATION

The installer no longer installs earlier versions of packages

Previously, the installer did not correctly load the DNF configuration file during the installation process. As a consequence, the installer sometimes installed earlier versions of select packages in the RPM transaction.

This bug has been fixed, and only the latest versions of packages are now installed from the installation repositories. In cases where it is impossible to install the latest versions of the packages, the installation fails as expected.

(BZ#2053710)

Anaconda installation is successful even if changing the network configuration in stage2

Previously, when using the **rd.live.ram** boot argument, Anaconda did not unmount an NFS mount point that is used in **initramfs** to fetch the installation image into memory. As a consequence, the installation process could become unresponsive or fail with a timeout error if the network configuration was changed in stage2.

To fix this problem, the NFS mount point used to fetch the installation image into memory is unmounted in **initramfs** before switchroot. As a result, the installation process is completed without any interruption.

(BZ#2082132)

8.2. SUBSCRIPTION MANAGEMENT

virt-who now connects to ESX servers correctly when in FIPS mode

Previously, when using the **virt-who** utility on a RHEL 9 system in FIPS mode, **virt-who** could not connect to ESX servers. As a consequence, **virt-who** did not report any ESX servers, even if configured for them, and logged the following error message:

```
ValueError: [digital envelope routines] unsupported
```

With this update, **virt-who** has been fixed to handle FIPS mode correctly, and the described problem no longer occurs.

(BZ#2054504)

8.3. SOFTWARE MANAGEMENT

DNF now correctly rolls back a transaction containing an item with the Reason Change Action type

Previously, running the **dnf history rollback** command on a transaction containing an item with the **Reason Change** Action type failed. With this update, the issue has been fixed, and **dnf history rollback** now works as expected.

(BZ#2053014)

8.4. SHELLS AND COMMAND-LINE TOOLS

The vi command in ReaR no longer results in an infinite loop

Previously, the ReaR rescue system did not contain the **vi** executable, only the **/bin/vi** script. As a consequence, the **/bin/vi** script caused an infinite loop when invoked. With this update, the ReaR rescue system contains the actual **vi** executable **/usr/libexec/vi**, and running the **vi** command no longer leads to an endless loop.

(BZ#2097437)

ReaR with the PXE output method no longer fails to store the output files in the rsync OUTPUT_URL location

Previously, the handling of the **OUTPUT_URL** variable with the **OUTPUT=PXE** and **BACKUP=RSYNC** options was removed. As a consequence, when using an rsync location for **OUTPUT_URL**, ReaR failed to copy the **initrd** and kernel files to this location, although it uploaded them to the location specified by **BACKUP_URL**. With this update, the behavior from RHEL 8.4 and earlier releases is restored. ReaR creates the required files at the designated **OUTPUT_URL** destination using rsync.

(BZ#2115958)

ReaR no longer fails to display an error message if it does not update the UUID in /etc/fstab

Previously, ReaR did not display an error message during recovery when it failed to update the universally unique identifier (UUID) in **/etc/fstab** to match the UUID of the newly created partition in case the UUIDs were different. This could have happened if the rescue image was out of sync with the backup. With this update, an error message occurs during recovery if the restored basic system files do not match the recreated system.

(BZ#2083272)

ReaR now supports restoring a system using NetBackup version 9

Previously, restoring a system using the NetBackup (NBU) method with NetBackup version 9 or later failed due to missing libraries and other files. With this update, the **NBU_LD_LIBRARY_PATH** variable contains the required library paths and the rescue system now incorporates the required files, and ReaR can use the NetBackup method.

(BZ#2120736)

ReaR no longer displays a false error message about missing symlink targets

Previously, ReaR displayed incorrect error messages about missing symlink targets for the **build** and **source** symlinks under **/usr/lib/modules/** when creating the rescue image. This situation was harmless, and you could safely ignore the error message. With this update, ReaR does not report a false error message about missing symlink targets in this situation.

(BZ#2119501)

The cmx operation with no parameter no longer crashes the CIM Client

The **cmx** operation calls a method and returns XML, a parameter specifies the name of the called method. Previously, the command line **sblim-wbemcli** Common Information Model (CIM) Client crashed when running the **cmx** operation without an additional parameter. With this update, the **cmx**

operation requires the parameter that defines the name of the called method. Invoking the **cmx** operation without this parameter results in an error message, and the CIM Client no longer crashes.

(BZ#2083577)

free command uses a new calculation method for used memory

Previously, the calculation of used memory in the **free** utility subtracted free space, cache space and buffer space from the total memory. Consequently, a discrepancy occurred when you compared the value of used memory with outcome of another tool because the **free** utility did not calculate shared memory. With this update, the **free** command uses a new calculation method that provides clear state of free memory and considers the unreclaimable cache. Used memory is now any memory that is not available, and includes also **tmpfs** objects that are in the virtual memory.

(BZ#2003033)

8.5. INFRASTRUCTURE SERVICES

Unbound no longer validates SHA-1-based RSA signatures

Previously, OpenSSL did not validate SHA-1-based RSA signatures in the DEFAULT system-wide cryptographic policy. As a consequence, when Unbound tried to validate such signatures, the error from OpenSSL caused the resolution to fail. With this update, Unbound disables validation support of all RSA/SHA1 (algorithm number 5) and RSASHA1-NSEC3-SHA1 (algorithm number 7) signatures, which resolves the query. Note that this makes the result insecure under all system-wide cryptographic policies.

(BZ#2071543)

8.6. SECURITY

OpenSSH key generation uses FIPS-compatible interfaces

The OpenSSL cryptographic library, which is used by OpenSSH, provides two interfaces: legacy and modern. Previously, OpenSSH used the legacy interface for key generation, which did not comply with Federal Information Processing Standards (FIPS) requirements. With this update, the **ssh-keygen** utility uses the FIPS-compliant API instead of the low-level FIPS-incompatible API. As a result, OpenSSH key generation is FIPS-compliant.

(BZ#2087121)

Cryptography not approved by FIPS no longer works in OpenSSL in FIPS mode

Previously, cryptography that was not FIPS-approved worked in the OpenSSL toolkit regardless of system settings. Consequently, you could use cryptographic algorithms and ciphers that should be disabled when the system is running in FIPS mode, for example:

- TLS cipher suites using the RSA key exchange worked.
- RSA-based algorithms for public-key encryption and decryption worked despite using the PKCS #1 and SSLv23 paddings or using keys shorter than 2048 bits.

This update contains fixes ensuring that cryptography not approved by FIPS no longer works in OpenSSL in FIPS mode.

(BZ#2053289)

Specifying arbitrary curves removed from OpenSSL

Previously, the checks of explicit curve parameters safety were incomplete. As a consequence, arbitrary elliptic curves with sufficiently large **p** values worked in RHEL. With this update, the checks now verify that the explicit curve parameters match one of the well-known supported curves. As a result, the option to specify arbitrary curves through the use of explicit curve parameters has been removed from OpenSSL. Parameter files, private keys, public keys, and certificates that specify arbitrary explicit curves no longer work in OpenSSL. Using explicit curve parameters to specify one of the well known and supported curves such as P-224, P-256, P-384, P-521, and **secp256k1** remains supported in non-FIPS mode.

(BZ#2066412)

OpenSSL req uses AES-256-CBC for private keys encryption

Previously, the OpenSSL **req** tool encrypted private key files by using the 3DES algorithm. Because the 3DES algorithm is insecure and disallowed in the current FIPS 140 standard for cryptographic modules, **req** now generates private key files encrypted using the AES-256-CBC algorithm instead. The overall PKCS#8 file format remains unchanged.

(BZ#2063947)

OpenSSL no longer fails to connect when FFDHE is used

Previously, TLS connections that use the finite-field-based Diffie-Hellman ephemeral (FFDHE) key exchange mechanism sometimes failed when processing FFDHE key shares from a client. This was caused by overly restrictive checks in OpenSSL. As a consequence, the OpenSSL server aborted the connection with an **internal_error** alert. With this update, OpenSSL accepts smaller but still compliant client key shares. As a result, connections between OpenSSL and other implementations no longer randomly abort when using FFDHE key exchanges.

(BZ#2004915)

OpenSSL-based applications now work correctly with the Turkish locale

Because the **OpenSSL** library uses case-insensitive string comparison functions, OpenSSL-based applications did not work correctly with the Turkish locale, and omitted checks caused applications using this locale to crash. This update provides a patch to use the Portable Operating System Interface (POSIX) locale for case-insensitive string comparison. As a result, OpenSSL-based applications such as curl work correctly with the Turkish locale.

(BZ#2071631)

Permissions for insights-client added to the SELinux policy

The new **insights-client** service requires permissions which were not in the previous **selinux-policy** versions. As a consequence, some components of **insights-client** did not work correctly and reported access vector cache (AVC) error messages. This update adds new permissions to the SELinux policy. As a result, **insights-client** runs correctly without reporting AVC errors.

(BZ#2081425, BZ#2077377, BZ#2087765, BZ#2107363)

SELinux staff_u users no longer can incorrectly switch to unconfined_r

Previously, when the **secure_mode** boolean was enabled, **staff_u** users could switch to the **unconfined_r** role, which was not expected behavior. As a consequence, **staff_u** users could perform privileged operations affecting the security of the system. With this update, the SELinux policy has been fixed, and **staff_u** users no longer can incorrectly switch to **unconfined_r**.

(BZ#2076681)

OpenSCAP no longer produces incorrect errors when checking available memory

Previously, when evaluating some XCCDF rules, OpenSCAP incorrectly showed the error message **Failed to check available memory** and produced invalid scan results. For example, this occurred for rules **accounts_user_dot_no_world_writable_programs**, **accounts_user_dot_group_ownership** and **accounts_users_home_files_permissions**. With this update, the bug in error handling is fixed and the error message appears only for real failures.

(BZ#2109485)

fagenrules --load now works correctly

Previously, the **fapolicyd** service did not correctly handle the signal hang up (SIGHUP). Consequently, **fapolicyd** terminated after receiving SIGHUP, and the **fagenrules --load** command did not work correctly. This update contains a fix for the problem. As a result, **fagenrules --load** now works correctly, and rule updates no longer require manual restarts of **fapolicyd**.

(BZ#2070655)

8.7. NETWORKING

An instance now retains the primary IP address even after starting the nm-cloud-setup service in Alibaba Cloud

Previously, after launching an instance in the Alibaba Cloud, the **nm-cloud-setup** service configured the incorrect IP address as the primary IP address in case of multiple IPv4 addresses. Consequently, this affected the selection of the IPv4 source address for outgoing connections. With this update, after configuring secondary IP addresses manually, the **NetworkManager** package fetches the primary IP address from **primary-ip-address** metadata and configures both primary and secondary IP addresses correctly.

(BZ#2079849)

The NetworkManager utility enforces correct ordering of manually added IPv6 addresses

In general, the ordering of IPv6 addresses affects the priority for source address selection. For example when you make an outgoing TCP connection. Previously, the relative priority of IPv6 addresses added through the **manual**, **dhcpv6**, and **autoconf6** methods was not correct. This update fixes the problem and the ordering priority now reflects this logic: **manual** > **dhcpv6** > **autoconf6**. Also, the order of addresses under the **ipv6.addresses** setting was reversed so that the address added first has the highest priority.

(BZ#2097293)

8.8. KERNEL

Network socket tagging works again

Certain legacy **cgroup** v1 controllers that have no **cgroup** v2 equivalent, such as **net_prio** or **net_cls**, previously interfered with the **cgroup** v2 socket tagging when they were mounted together with other **cgroup** v2 controllers in a mixed **cgroup** v1/v2 environment. As a consequence, a mixed **cgroup** v1/v2 environment using either the **net_prio** or **net_cls** v1 controller disabled proper network socket tagging with **cgroup** v2. This update eliminates this limitation, which makes it possible to use a mixed cgroup v1/v2 environment network socket tagging.

(BZ#2060150)

The kexec-tools package now supports the default crashkernel memory reservation values

The **kexec-tools** package now maintains the default **crashkernel** memory reservation values. The **kdump** service uses the default value to reserve the crash kernel memory for each kernel. This implementation also improves memory allocation for **kdump** when a system has less than 4 GB of available memory.

If the memory reserved by the default **crashkernel** value is not sufficient on your system, you can use the **kdumpctl estimate** command to get an estimated value without triggering a crash. The estimated **crashkernel=** value may not be accurate and can serve as a reference to set an appropriate **crashkernel=** value.

(BZ#1959203)

Systems can successfully run dynamic LPAR operations

Previously, users could not run dynamic logical partition (DLPAR) operations from the Hardware Management Console (HMC) if either of these conditions were met:

- The Secure Boot feature was enabled that implicitly enables kernel **lockdown** mechanism in integrity mode.
- The kernel **lockdown** mechanism was manually enabled in integrity or confidentiality mode.

In RHEL 9, kernel **lockdown** completely blocked Run Time Abstraction Services (RTAS) access to system memory accessible through the **/dev/mem** character device file. Several RTAS calls required write access to **/dev/mem** to function properly. Consequently, RTAS calls did not execute correctly and users would see the following error message:

> HSCL2957 Either there is currently no RMC connection between the management console and the partition <LPAR name> or the partition does not support dynamic partitioning operations. Verify the network setup on the management console and the partition and ensure that any firewall authentication between the management console and the partition has occurred. Run the management console diagrmc command to identify problems that might be causing no RMC connection.

With this update, the problem has been fixed by providing a very narrow PowerPC-specific exception to **lockdown**. The exception permits RTAS to access the required **/dev/mem** areas. As a result, the problem no longer manifests in the described scenario.

(BZ#2046472)

No kernel warnings after setting the ring buffer value from rx to max

The kernel was producing a warning message **Missing unregister, handled but fix driver** when an internal function expecting a clean input was called with a reused, already initialized structure. With this update, the problem has been fixed by reinitializing the structure before registering it again.

(BZ#2054379)

8.9. BOOT LOADER

grubby now passes arguments to future kernels

When installing a newer version of the kernel, the **grubby** tool did not pass the kernel command-line arguments from the previous kernel version. As a consequence, the GRUB boot loader ignored user settings. With this fix, the user settings now persist after installing the new kernel version.

(BZ#1978226)

8.10. FILE SYSTEMS AND STORAGE

Journal entries no longer stop the journal writes

Previously, in the VDO driver during device-mapper suspend operation and after resuming device operation, some journal blocks could still be marked as waiting for some metadata updates to be made before they could be reused, even though those updates had already been done. When enough journal entries were made for the journal to wrap around back to the same physical block, it was not available. Journal writes would stop, waiting for the block to become available, which never happened. Consequently, when some operations on a VDO device included a suspend or resume cycle, the device was in a frozen state after some journal updates. The journal updates before this device state were unpredictable because it was depended on previous allocation patterns within VDO, and the incoming write or discard patterns. With this update, after the suspend or resume cycle saving data to storage, the internal data structure state is reset and lockups no longer happened.

(BZ#2064802)

Adding a data device no longer triggers assertion failure

Previously, when adding additional devices to the cache, Stratis did not use cache immediately after initialization. As a consequence, the **stratisd** service returned an assertion failure message whenever a user attempted to add additional data devices to a pool. With this fix, cache is now used immediately after initialization and no assertion failures occur.

(BZ#2007018)

Resolved errors when adding new data devices to the encrypted pool

Previously, whenever the user initialized an encrypted pool with encrypted data devices, using a Clevis bind command on a tang server, specified with the **--trust-url** option, **stratisd** did not include the thumbprint part of the Clevis tang configuration in the internal data structures. Consequently, a failure occurred when attempting to add new data devices to the pool. With this update, the internal data structures of **stratisd** now include the thumbprint part of the Clevis tang configuration.

(BZ#2005110)

Connecting to NVMe namespaces from Broadcom initiators on AMD EPYC systems no longer require non-default IOMMU settings

By default, the RHEL kernel enables the IOMMU on AMD-based platforms. Previously, the **lpfc** driver did not use the scatter-gather list accessor macros. Consequently, certain servers with AMD processors encountered NVMe I/O problems, such as I/Os failing due to transfer length mismatches.

With this update, you do not need to put IOMMU into passthrough mode with a kernel command-line option in order to connect to NVMe namespaces from Broadcom initiators.

(BZ#2073541)

8.11. HIGH AVAILABILITY AND CLUSTERS

pcs now validates the value of stonith-watchdog-timeout

Previously, it was possible to set the **stonith-watchdog-timeout** property to a value that is incompatible with SBD configuration. This could result in a fence loop, or could cause the cluster to consider a fencing action to be successful even if the action is not finished. With this fix, **pcs** validates the value of **stonith-watchdog-property** when you set it, to prevent incorrect configuration.

(BZ#2058246)

pcs now recognizes the mode option when creating a new Booth ticket

Previously, when a user specified a **mode** option when adding a new Booth ticket, **pcs** reported the error **invalid booth ticket option 'mode'**. With this fix, you can now specify the **mode** option when creating a Booth ticket.

(BZ#2058243)

pcs now distinguishes between resources and stonith resources

Previously, some **pcs** commands did not distinguish between resources and stonith resources. This allowed users to use **pcs resource** sub-commands for stonith resources, and to use **pcs stonith** sub-commands for resources that are not stonith resources. This could lead to user confusion or resource misconfiguration. With this update, **pcs** displays a warning when there is a resource type mismatch.

(BZ#1301204)

8.12. COMPILERS AND DEVELOPMENT TOOLS

glibc now restores errno after loading an NSS module

Previously, the Name Service Switch (NSS) implementation in **glibc** set errno incorrectly during database enumeration using functions such as **getpwent()** if the last NSS module did not provide any data. As a result, applications using these enumeration functions incorrectly observed errors and failed. **glibc** now restores errno after loading an NSS module and, as a result, applications using these functions no longer fail.

(BZ#2063142)

The auditing interface now saves and restores the x8 register and the full width of the NEON registers for AArch64

Previously, a bug in the implementation of the dynamic loader's audit interface caused the **AArch64** saved register state to be incomplete compared to the procedure call standard. This bug has been fixed and the auditing interface now saves and restores the x8 register and the full width of the NEON registers for **AArch64**. Applications using the dynamic loader auditing interface can now inspect and influence the x8 register for **AArch64**. To use this new x8 register and have access to the full width of the NEON registers on **AArch64**, the audit modules must be recompiled to use the new version of the interface (LAV_CURRENT is 2).

(BZ#2003291)

POWER9-optimized strncpy function no longer gives incorrect results

Previously, the POWER9 strncpy function did not use the correct register as the source of the NUL bytes for padding. Consequently, the output buffer contained uninitialized register content instead of the NUL padding. With this update, the strncpy function has been fixed, and the end of the output buffer is now correctly padded with NUL bytes.

(BZ#2091549)

Valgrind override of glibc memmem function installed on IBMz15 architecture

Previously, a missing valgrind override of the **glibc memmem** function lead to false positive warnings of:

> Conditional jump or move depends on uninitialised value(s)

This update includes a valgrind override of the **glibc memmem** function and, as a result, there are no longer false positive warnings when using the **memmem** function in programs running under valgrind on the IBMz15 architecture.

(BZ#1993976)

8.13. IDENTITY MANAGEMENT

The ipa user-del --preserve user_login output no longer indicates that the user was deleted

Previously, if you ran the **ipa user-del --preserve user_login** command to preserve a user account, the output incorrectly returned the message **Deleted user "user_login"**. With this update, the output now returns **Preserved user "user_login"**.

(BZ#2100227)

PKINIT user authentication now works correctly in the RHEL 9 Kerberos client - Heimdal KDC scenario

Previously, the PKINIT authentication of an IdM user on a RHEL 9 Kerberos client against the Heimdal Kerberos Distribution Center (KDC) failed. This failure occurred because the Kerberos client did not support the **supportedCMSTypes** field required in the context of the deprecation of the SHA-1 algorithm in RHEL 9.

With this update, the RHEL 9 Kerberos client sends a list of signature algorithms including **sha512WithRSAEncryption**, and **sha256WithRSAEncryption** as **supportedCMSTypes** during PKINIT to Heimdal KDC. Heimdal KDC uses **sha512WithRSAEncryption** and, as a result, PKINIT authentication works correctly.

(BZ#2068935)

Handling unreadable objects in an LDAP group's member list

Before this update, SSSD inconsistently handled the unreadable objects in an LDAP group's member list and this resulted in unreadable objects causing an error or in certain situations unreadable objects were ignored.

With this update, SSSD has a new option **ldap_ignore_unreadable_references** to modify this behavior. If the **ldap_ignore_unreadable_references** option is set to **false**, unreadable objects cause an error and if set to **true**, unreadable objects are ignored. The default is set to **false** and because of the original inconsistent behavior, after the update, some group lookups may fail. In this case, set **ldap_ignore_unreadable_references = True** in the corresponding **[domain/*name of the domain*]** section in the **/etc/sssd/sssd.conf** file.

This allows unreadable objects to be handled in a consistent manner and the behavior can be tuned using the new **ldap_ignore_unreadable_references** option.

(BZ#2069376)

8.14. DESKTOP

Subscription enrolling with Activation keys has been fixed

Previously, you could not enroll your Red Hat subscription in **Settings** using Activation keys. **Settings** displayed the following error after pressing **Register**:

> Failed to register system; Failed to RegisterWithActivationKeys: Unknown arguments: dict_keys(['enable_content'])

With this update, the problem has been fixed, and you can now enroll your subscription using Activation keys as expected in **Settings**.

(BZ#2100467)

8.15. GRAPHICS INFRASTRUCTURES

X.org now enables the X11 SECURITY extension

Previously, the X.org display server did not provide the X11 **SECURITY** extension. As a consequence, applications that used this extension terminated unexpectedly.

With this update, X.org enables the X11 **SECURITY** extension. As a result, applications that depend on the extension now work as expected.

(BZ#1894612)

Matrox GPU with a VGA display now works as expected

Prior to this release, your display showed no graphical output if you used the following system configuration:

- A GPU in the Matrox MGA G200 family
- A display connected over the VGA controller
- UEFI switched to legacy mode

As a consequence, you could not use or install RHEL on this configuration.

With this update, the **mgag200** driver has been significantly rewritten, and as a result, the graphics output now works as expected.

(BZ#2100898)

8.16. THE WEB CONSOLE

Removing USB host devices using the web console now works as expected

Previously, when you attached a USB device to a virtual machine (VM), the device number and bus number of the USB device changed after they were passed to the VM. As a consequence, using the web console to remove such devices failed due to the incorrect correlation of the device and bus numbers. With this update, the issue has been fixed and you can remove the USB host devices using the web console.

(JIRA:RHELPLAN-109067)

Attaching multiple host devices using the web console now works as expected

Previously, when you selected multiple devices to attach to a virtual machine (VM) using the web console, only a single device was attached and the rest were ignored. With this update, the issue has been fixed and you can now simultaneously attach multiple host devices using the web console.

(JIRA:RHELPLAN-115603)

8.17. RED HAT ENTERPRISE LINUX SYSTEM ROLES

The network RHEL role manages ansible_managed parameter in the configuration files

Previously, the Ansible role was unable to provide the correct **ansible_managed** header for the **network** role managed configuration files. As a consequence, system administrators were uncertain about which files were managed by Ansible. With this fix, the role managed files have a correct **ansible_managed** header, and system administrators can reliably tell about which files are managed Ansible.

(BZ#2065382)

Fixed a typo to support active-backup for the correct bonding mode

Previously, there was a typo,**active_backup**, in supporting the InfiniBand port while specifying **active-backup** bonding mode. Due to this typo, the connection failed to support the correct bonding mode for the InfiniBand bonding port. This update fixes the typo by changing bonding mode to **active-backup**. The connection now successfully supports the InfiniBand bonding port.

(BZ#2065394)

The IPRouteUtils.get_route_tables_mapping() function now accepts any whitespace sequence

Previously, a parser for the **iproute2** routing table database, such as **/etc/iproute2/rt_tables**, asserted that entries in the file were of the form **254 main** and only a single space character separated the numeric id and the name. Consequently, the parser failed to cache all the mappings between the route table name and table id.Therefore the user could not add a static route into the route table by defining the route table name. With this update, the parser accepts any whitespace sequence in between the table ID and table name. As a result, as the parser caches all the mapping between the route table name and table ID, users can add a static route into the route table by defining the route table name.

(BZ#2115886)

The forward_port parameter now accepts both the string and dict option

Previously, in the **firewall** RHEL system role, the **forward_port** parameter only accepted the **string** option. However, the role documentation claimed that both **string** and **dict** options were supported. Consequently, the users reading and following the documentation were getting an error. This bug has been fixed by making **forward_port** accept both options. As a result, the users can safely follow the documentation to configure port forwarding.

(BZ#2100605)

Configuration by the metrics role now follows symbolic links correctly

When the **mssql pcp** package is installed, the **mssql.conf** file is located in **/etc/pcp/mssql/** and is targeted by the symbolic link **/var/lib/pcp/pmdas/mssql/mssql.conf**. Previously, however, the **metrics** role overwrote the symbolic link instead of following it and configuring **mssql.conf**. Consequently, running the **metrics** role changed the symbolic link to a regular file and the configuration therefore only affected the **/var/lib/pcp/pmdas/mssql/mssql.conf** file. This resulted in a failed symbolic link, and the

main configuration file **/etc/pcp/mssql/mssql.conf** was not affected by the configuration. The issue is now fixed and the **follow: yes** option to follow the symbolic link has been added to the **metrics** role. As a result, the **metrics** role preserves the symbolic links and correctly configures the main configuration file.

(BZ#2060523)

The kernel_settings configobj is available on managed hosts

Previously, the **kernel_settings** role did not install the **python3-configobj** package on managed hosts. As a consequence, the role returned an error stating that the **configobj** Python module could not be found. With this fix, the role ensures that the **python3-configobj** package is present on managed hosts and the **kernel_settings** role works as expected.

(BZ#2060525)

The mount_options parameter for volumes is now valid for a volume

Previously, the parameter was accidentally removed from the list of valid parameters for a volume. Consequently, users were unable to set the **mount_options** parameter for volumes. With this bug fix, the **mount_options** parameter has been added back to the list of valid parameters and the code has been refactored to catch the errors. As a result, the **storage** RHEL system role can set the **mount_options** parameter for volumes.

(BZ#2083376)

The storage RHEL system role now correctly supports striped and raid0 levels for LVM volumes

The **storage** RHEL system role previously incorrectly reported RAID levels **striped** and **raid0** as not supported for LVM volumes. This is now fixed and the role can now correctly create LVM volumes of all RAID levels supported by LVM: **raid0**, **raid1**, **raid4**, **raid5**, **raid6**, **raid10**, **striped** and **mirror**.

(BZ#2083410)

The metrics RHEL system role README and documentation now clearly specifies supported Redis and Grafana versions on specific versions of RHEL by the role

Previously, when trying to use the **metrics** role with unsupported versions of Redis and Grafana on unsupported platforms, the role failed. This update clarifies the documentation about which versions of Redis and Grafana are supported on which versions of RHEL by the role. As a result, you can avoid trying to use unsupported versions of Redis and Grafana on unsupported platforms.

(BZ#2100286)

Minimal RSA key bit length option in the ssh and sshd RHEL system roles

Accidentally using short RSA keys might make the system more vulnerable to attacks. With this update, you can set RSA key minimal bit lengths for OpenSSH clients and servers by using the **RequiredRSASize** option in the **ssh** and **sshd** RHEL system roles.

(BZ#2109998)

The nbde_client RHEL system role now uses proper spacing when specifying extra Dracut command line-parameters

The Dracut framework requires proper spacing when specifying additional parameters, such as kernel command-line parameters. If the parameters are not specified with proper spacing, Dracut might not

append the specified extra parameters to the kernel command line. With this update, the **nbde_client** RHEL system role uses proper spacing when creating add-on Dracut configuration files. As a result, the role correctly sets Dracut command-line parameters.

(BZ#2115156)

The tlog RHEL system roles is now correctly overlaid by SSSD

Previously, the **tlog** RHEL system role relied on the System Security Services Daemon (SSSD) files provider and on enabled **authselect** option **with-files-domain** to set up correct **passwd** entries in the **nsswitch.conf** file. In RHEL 9.0, SSSD did not implicitly enable the files provider by default, and consequently the **tlog-rec-session** shell overlay by SSSD did not work. With this fix, the **tlog** role now updates the **nsswitch.conf** to ensure **tlog-rec-session** is correctly overlaid by SSSD.

(BZ#2071804)

The metrics RHEL system role automatically restarts pmie and pmlogger services after an update to their configuration

Previously, the **pmie** and **pmlogger** services did not restart after their configuration was changed and waited for handler execution. This caused errors with other **metrics** services, which required **pmie** and **pmlogger** configuration to match their runtime behavior. With this update, the role restarts **pmie** and **pmlogger** immediately after a configuration update, their configuration matches runtime behavior of dependent metrics services, and they work correctly.

(BZ#2100294)

8.18. VIRTUALIZATION

Network traffic performance in virtual machines is no longer reduced when under heavy load

Previously, RHEL virtual machines had, in some cases, decreased performance when handling high levels of network traffic. The underlying code has been fixed and network traffic performance now works as expected in the described circumstances.

(BZ#1945040)

8.19. RHEL IN CLOUD ENVIRONMENTS

The SR-IOV functionality of a network adapter attached to a Hyper-V VM now works reliably

Previously, when attaching a network adapter with single-root I/O virtualization (SR-IOV) enabled to a RHEL 9 virtual machine (VM) running on Microsoft Hyper-V hypervisor, the SR-IOV functionality in some cases did not work correctly. A bug in the Hyper-V specific memory-mapped I/O (MMIO) allocation code has been fixed and the SR-IOV functionality now works as expected on Hyper-V VMs.

(BZ#2030922)

SR-IOV no longer performs suboptimally in ARM 64 RHEL 9 virtual machines on Azure

Previously, SR-IOV networking devices had significantly lower throughout and higher latency than expected in ARM 64 RHEL 9 virtual machines (VMs) running on a Microsoft Azure platform. The problem has been fixed, and the affected VMs now perform as expected.

(BZ#2068432)

8.20. CONTAINERS

podman system connection add and podman image scp no longer fail

Podman uses SHA-1 hashes for the RSA key exchange. Previously, the regular SSH connection among machines using RSA keys worked, while the **podman system connection add** and **podman image scp** commands did not work using the same RSA keys, because the SHA-1 hashes were not accepted for key exchange on RHEL 9. With the update, the problem has been fixed.

(JIRA:RHELPLAN-121180)

Container images signed with a Beta GPG key can now be pulled

Previously, when you pulled RHEL Beta container images, Podman failed with the error message: **Error: Source image rejected: None of the signatures were accepted**. The images failed to be pulled due to current builds being configured to not trust the RHEL Beta GPG keys by default. With this update, the **/etc/containers/policy.json** file supports a new **keyPaths** field which accepts a list of files containing the trusted keys. Because of this, the container images signed with GA and Beta GPG keys are now accepted in the default configuration.

(BZ#2094015)

Podman no longer fails to pull a container "X509: certificate signed by unknown authority"

Previously, if you had your own internal registry signed by our own CA certificate, then you had to import the certificate onto your host machine. Otherwise, an error occurs:

```
x509: certificate signed by unknown authority
```

With this update, the problem has been fixed.

(BZ#2027576)

DNF and YUM no longer fail because of non-matching repository IDs

Previously, DNF and YUM repository IDs did not match the format that DNF or YUM expected. For example, if you ran the following example, the error occurred:

```
# podman run -ti ubi8-ubi
# dnf debuginfo-install dnsmasq
...
This system is not registered with an entitlement server. You can use subscription-manager to register.
```

With this update, the problem has been fixed. Suffix **--debug-rpms** was added to all debug repository names (for example **ubi-8-appstream-debug-rpms**), and also the suffix **-rpms** was added to all UBI repository names (for example **ubi-8-appstream-rpms**).

For more information, see Universal Base Images (UBI): Images, repositories, packages, and source code.

(BZ#2120378)

CHAPTER 9. TECHNOLOGY PREVIEWS

This part provides a list of all Technology Previews available in Red Hat Enterprise Linux 9.

For information on Red Hat scope of support for Technology Preview features, see Technology Preview Features Support Scope.

9.1. SHELLS AND COMMAND-LINE TOOLS

ReaR available on the 64-bit IBM Z architecture as a Technology Preview

Basic Relax and Recover (ReaR) functionality is now available on the 64-bit IBM Z architecture as a Technology Preview. You can create a ReaR rescue image on IBM Z only in the z/VM environment. Backing up and recovering logical partitions (LPARs) has not been tested.

The only output method currently available is Initial Program Load (IPL). IPL produces a kernel and an initial ramdisk (initrd) that can be used with the **zIPL** bootloader.

WARNING

Currently, the rescue process reformats all the DASDs (Direct Attached Storage Devices) connected to the system. Do not attempt a system recovery if there is any valuable data present on the system storage devices. This also includes the device prepared with the **zIPL** bootloader, ReaR kernel, and initrd that were used to boot into the rescue environment. Ensure to keep a copy.

For more information, see Using a ReaR rescue image on the 64-bit IBM Z architecture .

(BZ#2046653)

GIMP available as a Technology Preview in RHEL 9

GNU Image Manipulation Program (GIMP) 2.99.8 is now available in RHEL 9 as a Technology Preview. The **gimp** package version 2.99.8 is a pre-release version with a set of improvements, but a limited set of features and no guarantee for stability. As soon as the official GIMP 3 is released, it will be introduced into RHEL 9 as an update of this pre-release version.

In RHEL 9, you can install **gimp** easily as an RPM package.

(BZ#2047161)

9.2. SECURITY

gnutls now uses KTLS as a Technology Preview

The updated **gnutls** packages can use Kernel TLS (KTLS) for accelerating data transfer on encrypted channels as a Technology Preview. To enable KTLS, add the **tls.ko** kernel module using the **modprobe** command, and create a new configuration file **/etc/crypto-policies/local.d/gnutls-ktls.txt** for the system-wide cryptographic policies with the following content:

```
[global]
ktls = true
```

Note that the current version does not support updating traffic keys through TLS **KeyUpdate** messages, which impacts the security of AES-GCM ciphersuites. See the RFC 7841 - TLS 1.3 document for more information.

(BZ#2042009)

9.3. 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.

(BZ#1613522)

Configuring Multipath TCP using NetworkManager is available as a Technology Preview

With this update, the NetworkManager utility provides you with the Multipath TCP (MPTCP) functionality. You can use **nmcli** commands to control MPTCP and make its settings persistent.

For more information, see Understanding Multipath TCP: High availability for endpoints and the networking highway of the future and RFC 8684: TCP Extensions for Multipath Operation with Multiple Addresses.

(BZ#2029636)

KTLS available as a Technology Preview

RHEL provides Kernel Transport Layer Security (KTLS) as a Technology Preview. KTLS handles TLS records 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.

(BZ#1570255)

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.

(BZ#2020529)

9.4. KERNEL

The Intel data streaming accelerator driver for kernel is available as a Technology Preview

The Intel data streaming accelerator driver (IDXD) for the kernel is currently available as a Technology Preview. It is an Intel CPU integrated accelerator and includes the shared work queue with process address space ID (pasid) submission and shared virtual memory (SVM).

(BZ#2030412)

SGX available as a Technology Preview

Software Guard Extensions(SGX) is an Intel® technology for protecting software code and data from disclosure and modification. The RHEL kernel partially provides the SGX v1 and v1.5 functionality. The version 1 enables platforms using the **Flexible Launch Control** mechanism to use the SGX technology.

(BZ#1874182)

The Soft-iWARP driver is available as a Technology Preview

Soft-iWARP (siw) is a software, Internet Wide-area RDMA Protocol (iWARP), kernel driver for Linux. Soft-iWARP implements the iWARP protocol suite over the TCP/IP network stack. This protocol suite is fully implemented in software and does not require a specific Remote Direct Memory Access (RDMA) hardware. Soft-iWARP enables a system with a standard Ethernet adapter to connect to an iWARP adapter or to another system with already installed Soft-iWARP.

(BZ#2023416)

9.5. FILE SYSTEMS AND STORAGE

DAX is now available for ext4 and XFS as a Technology Preview

In RHEL 9, the DAX file system is available as a Technology Preview. DAX provides means for an application to directly map persistent memory into its address space. To use DAX, a system must have some form of persistent memory available, usually in the form of one or more Non-Volatile Dual In-line Memory Modules (NVDIMMs), and a DAX compatible file system must be created on the NVDIMM(s). Also, the file system must be mounted with the **dax** mount option. Then, an **mmap** of a file on the dax-mounted file system results in a direct mapping of storage into the application's address space.

(BZ#1995338)

Stratis is available as a Technology Preview

Stratis is a local storage manager. It provides managed file systems on top of pools of storage with additional features to the user:

- Manage snapshots and thin provisioning
- Automatically grow file system sizes as needed
- Maintain file systems

To administer Stratis storage, use the **stratis** utility, which communicates with the **stratisd** background service.

Stratis is provided as a Technology Preview.

For more information, see the Stratis documentation: Setting up Stratis file systems.

(BZ#2041558)

NVMe-oF Discovery Service features available as a Technology Preview

The NVMe-oF Discovery Service features, defined in the NVMexpress.org Technical Proposals (TP) 8013 and 8014, are available as a Technology Preview. To preview these features, use the **nvme-cli 2.0** package and attach the host to an NVMe-oF target device that implements TP-8013 or TP-8014. For more information about TP-8013 and TP-8014, see the NVM Express 2.0 Ratified TPs from the https://nvmexpress.org/developers/nvme-specification/ website.

(BZ#2021672)

nvme-stas package available as a Technology Preview

The **nvme-stas** package, which is a Central Discovery Controller (CDC) client for Linux, is now available as a Technology Preview. It handles Asynchronous Event Notifications (AEN), Automated NVMe subsystem connection controls, Error handling and reporting, and Automatic (**zeroconf**) and Manual configuration.

This package consists of two daemons, Storage Appliance Finder (**stafd**) and Storage Appliance Connector (**stacd**).

(BZ#1893841)

NVMe over Fibre Channel devices are now available in RHEL installation program as a Technology Preview

You can now add NVMe over Fibre Channel devices to your RHEL installation as a Technology Preview. In RHEL installation program, you can select these devices under the NVMe Fabrics Devices section while adding disks on the Installation Destination screen.

(BZ#2107346)

9.6. COMPILERS AND DEVELOPMENT TOOLS

jmc-core and owasp-java-encoder available as a Technology Preview

RHEL 9 is distributed with the **jmc-core** and **owasp-java-encoder** packages as Technology Preview features.

jmc-core is a library providing core APIs for Java Development Kit (JDK) Mission Control, including libraries for parsing and writing JDK Flight Recording files, as well as libraries for Java Virtual Machine (JVM) discovery through Java Discovery Protocol (JDP).

The **owasp-java-encoder** package provides a collection of high-performance low-overhead contextual encoders for Java.

(BZ#1980981)

9.7. 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.

(BZ#2084180)

Identity Management JSON-RPC API available as Technology Preview

An API is available for Identity Management (IdM). To view the API, IdM also provides an API browser as a Technology Preview.

Previously, the IdM API was enhanced to enable multiple versions of API commands. These enhancements could change the behavior of a command in an incompatible way. Users are now able to continue using existing tools and scripts even if the IdM API changes. This enables:

- Administrators to use previous or later versions of IdM on the server than on the managing client.
- Developers can use a specific version of an IdM call, even if the IdM version changes on the server.

In all cases, the communication with the server is possible, regardless if one side uses, for example, a newer version that introduces new options for a feature.

For details on using the API, see Using the Identity Management API to Communicate with the IdM Server (TECHNOLOGY PREVIEW).

(BZ#2084166)

RHEL IdM allows delegating user authentication to external identity providers as a Technology Preview

In RHEL IdM, you can now associate users with external identity providers (IdP) that support the OAuth 2 device authorization flow. When these users authenticate with the SSSD version available in RHEL 9.1, they receive RHEL IdM single sign-on capabilities with Kerberos tickets after performing authentication and authorization at the external IdP.

Notable features include:

- Adding, modifying, and deleting references to external IdPs with **ipa idp-*** commands
- Enabling IdP authentication for users with the **ipa user-mod --user-auth-type=idp** command

For additional information, see Using external identity providers to authenticate to IdM .

(BZ#2069202)

sssd-idp sub-package available as a Technology Preview

The **sssd-idp** sub-package for SSSD contains the **oidc_child** and krb5 **idp** plugins, which are client-side components that perform OAuth2 authentication against Identity Management (IdM) servers. This feature is available only with IdM servers on RHEL 8.7 and higher, and RHEL 9.1 and higher.

(BZ#2065693)

SSSD internal krb5 idp plugin available as a Technology Preview

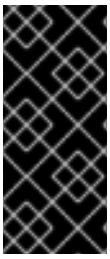
The SSSD krb5 **idp** plugin allows you to authenticate against an external identity provider (IdP) using the OAuth2 protocol. This feature is available only with IdM servers on RHEL 8.7 and higher, and RHEL 9.1 and higher.

(BZ#2056482)

ACME available as a Technology Preview

The Automated Certificate Management Environment (ACME) service is now available in Identity Management (IdM) as a Technology Preview. 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.



IMPORTANT

It is recommended to enable ACME only in an IdM deployment where all servers are running RHEL 8.4 or later. Earlier RHEL versions do not include the ACME service, which can cause problems in mixed-version deployments. For example, a CA server without ACME can cause client connections to fail, because it uses a different DNS Subject Alternative Name (SAN).

WARNING

Currently, RHCS does not remove expired certificates. Because ACME certificates expire after 90 days, the expired certificates can accumulate and this can affect performance.

- To enable ACME across the whole IdM deployment, use the **ipa-acme-manage enable** command:

```
# ipa-acme-manage enable
The ipa-acme-manage command was successful
```

- To disable ACME across the whole IdM deployment, use the **ipa-acme-manage disable** command:

```
# ipa-acme-manage disable
The ipa-acme-manage command was successful
```

- To check whether the ACME service is installed and if it is enabled or disabled, use the **ipa-acme-manage status** command:

```
# ipa-acme-manage status
ACME is enabled
The ipa-acme-manage command was successful
```

(BZ#2084181)

9.8. DESKTOP

GNOME for the 64-bit ARM architecture available as a Technology Preview

The GNOME desktop environment is available for the 64-bit ARM architecture as a Technology Preview.

You can now connect to the desktop session on a 64-bit ARM server using VNC. As a result, you can manage the server using graphical applications.

A limited set of graphical applications is available on 64-bit ARM. For example:

- The Firefox web browser
- Red Hat Subscription Manager (**subscription-manager-cockpit**)
- Firewall Configuration (**firewall-config**)
- Disk Usage Analyzer (**baobab**)

Using Firefox, you can connect to the Cockpit service on the server.

Certain applications, such as LibreOffice, only provide a command-line interface, and their graphical interface is disabled.

(JIRA:RHELPLAN-27394)

GNOME for the IBM Z architecture available as a Technology Preview

The GNOME desktop environment is available for the IBM Z architecture as a Technology Preview.

You can now connect to the desktop session on an IBM Z server using VNC. As a result, you can manage the server using graphical applications.

A limited set of graphical applications is available on IBM Z. For example:

- The Firefox web browser
- Red Hat Subscription Manager (**subscription-manager-cockpit**)
- Firewall Configuration (**firewall-config**)
- Disk Usage Analyzer (**baobab**)

Using Firefox, you can connect to the Cockpit service on the server.

Certain applications, such as LibreOffice, only provide a command-line interface, and their graphical interface is disabled.

(JIRA:RHELPLAN-27737)

9.9. THE WEB CONSOLE

Stratis available as a Technology Preview in the RHEL web console

With this update, the Red Hat Enterprise Linux web console provides the ability to manage Stratis storage as a Technology Preview.

To learn more about Stratis, see What is Stratis.

(JIRA:RHELPLAN-122345)

9.10. VIRTUALIZATION

RHEL VMs can now be deployed to VMware ESXi instances running on ARM64 processors

As a Technology Preview, it is now possible to deploy RHEL virtual machines to VMware ESXi hypervisor instances running on 64-bit ARM-based processors.

(JIRA:RHELPLAN-95456)

AMD SEV and SEV-ES for KVM virtual machines

As a Technology Preview, RHEL 9 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 security of the VM.

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.

Note that SEV and SEV-ES work only on the 2nd generation of AMD EPYC CPUs (codenamed Rome) or later. Also note that RHEL 9 includes SEV and SEV-ES encryption, but not the SEV and SEV-ES security attestation.

(JIRA:RHELPLAN-65217)

Virtualization is now available on ARM 64

As a Technology Preview, it is now possible to create KVM virtual machines on systems using ARM 64 CPUs.

(JIRA:RHELPLAN-103993)

virtio-mem is now available on AMD64, Intel 64, and ARM 64

As a Technology Preview, RHEL 9 introduces the **virtio-mem** feature on AMD64, Intel 64, and ARM 64 systems. Using **virtio-mem** makes it possible to dynamically add or remove host memory in virtual machines (VMs).

To use **virtio-mem**, define **virtio-mem** memory devices in the XML configuration of a VM and use the **virsh update-memory-device** command to request memory device size changes while the VM is running. To see the current memory size exposed by such memory devices to a running VM, view the XML configuration of the VM.

(BZ#2014487, BZ#2044162, BZ#2044172)

Intel vGPU available as a Technology Preview

As a Technology Preview, it is possible to divide a physical Intel GPU device into multiple virtual devices referred to as **mediated devices**. These mediated devices can then be assigned to multiple virtual machines (VMs) as virtual GPUs. As a result, these VMs share the performance of a single physical Intel GPU.

Note that this feature is deprecated and will be removed entirely in a future RHEL release.

(JIRA:RHELDOCS-17050)

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 9. With this feature, a RHEL 7, RHEL 8, or RHEL 9 VM that runs on a physical RHEL 9 host can act as a hypervisor, and host its own VMs.

(JIRA:RHELDOCS-17040)

9.11. RHEL IN CLOUD ENVIRONMENTS

RHEL confidential VMs are now available on Azure as a Technology Preview

With the updated RHEL kernel, you can now create and run confidential virtual machines (VMs) on Microsoft Azure as a Technology Preview. However, it is not yet possible to encrypt RHEL confidential VM images during boot on Azure.

(JIRA:RHELPLAN-122321)

9.12. CONTAINERS

The capability for multiple trusted GPG keys for signing images is available as a Technology Preview

The **/etc/containers/policy.json** file supports a new **keyPaths** field which accepts a list of files containing the trusted keys. Because of this, the container images signed with GA and Beta GPG keys are now accepted in the default configuration.

For example:

```
"registry.redhat.io": [
        {
            "type": "signedBy",
            "keyType": "GPGKeys",
            "keyPaths": ["/etc/pki/rpm-gpg/RPM-GPG-KEY-redhat-release", "/etc/pki/rpm-gpg/RPM-GPG-
KEY-redhat-beta"]
        }
]
```

(JIRA:RHELPLAN-129327)

The sigstore signatures are now available as a Technology Preview

Beginning with Podman 4.2, you can use the sigstore format of container image signatures. The sigstore signatures are stored in the container registry together with the container image without the need to have a separate signature server to store image signatures.

(JIRA:RHELPLAN-74672)

The podman-machine command is unsupported

The **podman-machine** command for managing virtual machines, is available only as a Technology Preview. Instead, run Podman directly from the command line.

(JIRA:RHELDOCS-16861)

CHAPTER 10. DEPRECATED FUNCTIONALITY

Deprecated devices are fully supported, which means that they are tested and maintained, and their support status remains unchanged within Red Hat Enterprise Linux 9. However, these devices will likely not be supported in the next major version release, and are not recommended for new deployments on the current or future major versions of RHEL.

For the most recent list of deprecated functionality within a particular major release, see the latest version of release documentation. For information about the length of support, see Red Hat Enterprise Linux Life Cycle and Red Hat Enterprise Linux Application Streams Life Cycle .

A package can be deprecated and not recommended for further use. Under certain circumstances, a package can be removed from the product. Product documentation then identifies more recent packages that offer functionality similar, identical, or more advanced to the one deprecated, and provides further recommendations.

For information regarding functionality that is present in RHEL 8 but has been *removed* in RHEL 9, see Considerations in adopting RHEL 9 .

10.1. INSTALLER AND IMAGE CREATION

Deprecated Kickstart commands

The following Kickstart commands have been deprecated:

- **timezone --ntpservers**
- **timezone --nontp**
- **logging --level**
- **%packages --excludeWeakdeps**
- **%packages --instLangs**
- **%anaconda**
- **pwpolicy**

Note that where only specific options are listed, the base command and its other options are still available and not deprecated. Using the deprecated commands in Kickstart files prints a warning in the logs. You can turn the deprecated command warnings into errors with the **inst.ksstrict** boot option.

(BZ#1899167)

10.2. SHELLS AND COMMAND-LINE TOOLS

Setting the TMPDIR variable in the ReaR configuration file is deprecated

Setting the **TMPDIR** environment variable in the **/etc/rear/local.conf** or **/etc/rear/site.conf** ReaR configuration file), by using a statement such as **export TMPDIR=...**, does not work and is deprecated.

To specify a custom directory for ReaR temporary files, export the variable in the shell environment before executing ReaR. For example, execute the **export TMPDIR=...** statement and then execute the **rear** command in the same shell session or script.

Jira:RHELDOCS-18049

10.3. SECURITY

SHA-1 is deprecated for cryptographic purposes

The usage of the SHA-1 message digest for cryptographic purposes has been deprecated in RHEL 9. The digest produced by SHA-1 is not considered secure because of many documented successful attacks based on finding hash collisions. The RHEL core crypto components no longer create signatures using SHA-1 by default. Applications in RHEL 9 have been updated to avoid using SHA-1 in security-relevant use cases.

Among the exceptions, the HMAC-SHA1 message authentication code and the Universal Unique Identifier (UUID) values can still be created using SHA-1 because these use cases do not currently pose security risks. SHA-1 also can be used in limited cases connected with important interoperability and compatibility concerns, such as Kerberos and WPA-2. See the List of RHEL applications using cryptography that is not compliant with FIPS 140-3 section in the RHEL 9 Security hardening document for more details.

If your scenario requires the use of SHA-1 for verifying existing or third-party cryptographic signatures, you can enable it by entering the following command:

```
# update-crypto-policies --set DEFAULT:SHA1
```

Alternatively, you can switch the system-wide crypto policies to the **LEGACY** policy. Note that **LEGACY** also enables many other algorithms that are not secure.

(JIRA:RHELPLAN-110763)

SCP is deprecated in RHEL 9

The secure copy protocol (SCP) is deprecated because it has known security vulnerabilities. The SCP API remains available for the RHEL 9 lifecycle but using it reduces system security.

- In the **scp** utility, SCP is replaced by the SSH File Transfer Protocol (SFTP) by default.
- The OpenSSH suite does not use SCP in RHEL 9.
- SCP is deprecated in the **libssh** library.

(JIRA:RHELPLAN-99136)

Digest-MD5 in SASL is deprecated

The Digest-MD5 authentication mechanism in the Simple Authentication Security Layer (SASL) framework is deprecated, and it might be removed from the **cyrus-sasl** packages in a future major release.

(BZ#1995600)

OpenSSL deprecates MD2, MD4, MDC2, Whirlpool, RIPEMD160, Blowfish, CAST, DES, IDEA, RC2, RC4, RC5, SEED, and PBKDF1

The OpenSSL project has deprecated a set of cryptographic algorithms because they are insecure, uncommonly used, or both. Red Hat also discourages the use of those algorithms, and RHEL 9 provides them for migrating encrypted data to use new algorithms. Users must not depend on those algorithms for the security of their systems.

The implementations of the following algorithms have been moved to the legacy provider in OpenSSL: MD2, MD4, MDC2, Whirlpool, RIPEMD160, Blowfish, CAST, DES, IDEA, RC2, RC4, RC5, SEED, and PBKDF1.

See the **/etc/pki/tls/openssl.cnf** configuration file for instructions on how to load the legacy provider and enable support for the deprecated algorithms.

(BZ#1975836)

/etc/system-fips is now deprecated

Support for indicating FIPS mode through the **/etc/system-fips** file has been removed, and the file will not be included in future versions of 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 using the **fips-mode-setup --check** command.

(JIRA:RHELPLAN-103232)

libcrypt.so.1 is now deprecated

The **libcrypt.so.1** library is now deprecated, and it might be removed in a future version of RHEL.

(BZ#2034569)

fapolicyd.rules is deprecated

The **/etc/fapolicyd/rules.d/** directory for files containing allow and deny execution rules replaces the **/etc/fapolicyd/fapolicyd.rules** file. The **fagenrules** script now merges all component rule files in this directory to the **/etc/fapolicyd/compiled.rules** file. Rules in **/etc/fapolicyd/fapolicyd.trust** are still processed by the **fapolicyd** framework but only for ensuring backward compatibility.

(BZ#2054740)

10.4. NETWORKING

Network teams are deprecated in RHEL 9

The **teamd** service and the **libteam** library are deprecated in Red Hat Enterprise Linux 9 and will be removed in the next major release. 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.

For details about how to migrate a team to a bond, see Migrating a network team configuration to network bond.

(BZ#1935544)

NetworkManager connection profiles in ifcfg format are deprecated

In RHEL 9.0 and later, connection profiles in **ifcfg** format are deprecated. The next major RHEL release will remove the support for this format. However, in RHEL 9, NetworkManager still processes and updates existing profiles in this format if you modify them.

By default, NetworkManager now stores connection profiles in keyfile format in the **/etc/NetworkManager/system-connections/** directory. Unlike the **ifcfg** format, the keyfile format supports all connection settings that NetworkManager provides. For further details about the keyfile

format and how to migrate profiles, see NetworkManager connection profiles in keyfile format . (BZ#1894877)

The iptables back end in firewalld is deprecated

In RHEL 9, the **iptables** framework is deprecated. As a consequence, the **iptables** back end and the **direct interface** in **firewalld** are also deprecated. Instead of the **direct interface** you can use the native features in **firewalld** to configure the required rules.

(BZ#2089200)

10.5. KERNEL

ATM encapsulation is deprecated in RHEL 9

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, which support the ADSL technology and are being phased out by manufacturers. Therefore, ATM encapsulation is deprecated in Red Hat Enterprise Linux 9.

For more information, see PPP Over AAL5, Multiprotocol Encapsulation over ATM Adaptation Layer 5 , and Classical IP and ARP over ATM .

(BZ#2058153)

10.6. FILE SYSTEMS AND STORAGE

lvm2-activation-generator and its generated services removed in RHEL 9.0

The **lvm2-activation-generator** program and its generated services **lvm2-activation**, **lvm2-activation-early**, and **lvm2-activation-net** are removed in RHEL 9.0. The **lvm.conf event_activation** setting, used to activate the services, is no longer functional. The only method for auto activating volume groups is event based activation.

(BZ#2038183)

10.7. DYNAMIC PROGRAMMING LANGUAGES, WEB AND DATABASE SERVERS

libdb has been deprecated

RHEL 8 and RHEL 9 currently provide Berkeley DB (**libdb**) version 5.3.28, which is distributed under the LGPLv2 license. The upstream Berkeley DB version 6 is available under the AGPLv3 license, which is more restrictive.

The **libdb** package is deprecated as of RHEL 9 and might not be available in future major RHEL releases.

In addition, cryptographic algorithms have been removed from **libdb** in RHEL 9 and multiple **libdb** dependencies have been removed from RHEL 9.

Users of **libdb** are advised to migrate to a different key-value database. For more information, see the Knowledgebase article Available replacements for the deprecated Berkeley DB (libdb) in RHEL .

(BZ#1927780, BZ#1974657, JIRA:RHELPLAN-80695)

10.8. COMPILERS AND DEVELOPMENT TOOLS

Smaller size of keys than 2048 are deprecated by openssl 3.0

Key sizes smaller than 2048 bits are deprecated by **openssl** 3.0 and no longer work in Go's FIPS mode.

(BZ#2111072)

Some PKCS1 v1.5 modes are now deprecated

Some **PKCS1** v1.5 modes are not approved in **FIPS-140-3** for encryption and are disabled. They will no longer work in Go's FIPS mode.

(BZ#2092016)

10.9. IDENTITY MANAGEMENT

SHA-1 in OpenDNSSec is now deprecated

OpenDNSSec supports exporting Digital Signatures and authentication records using the **SHA-1** algorithm. The use of the **SHA-1** algorithm is no longer supported. With the RHEL 9 release, **SHA-1** in OpenDNSSec is deprecated and it might be removed in a future minor release. Additionally, OpenDNSSec support is limited to its integration with Red Hat Identity Management. OpenDNSSec is not supported standalone.

(BZ#1979521)

The SSSD implicit files provider domain is disabled by default

The SSSD implicit **files** provider domain, which retrieves user information from local files such as **/etc/shadow** and group information from **/etc/groups**, is now disabled by default.

To retrieve user and group information from local files with SSSD:

1. Configure SSSD. Choose one of the following options:
   a. Explicitly configure a local domain with the **id_provider=files** option in the **sssd.conf** configuration file.

```
[domain/local]
id_provider=files
...
```

   b. Enable the **files** provider by setting **enable_files_domain=true** in the **sssd.conf** configuration file.

```
[sssd]
enable_files_domain = true
```

2. Configure the name services switch.

```
# authselect enable-feature with-files-provider
```

(JIRA:RHELPLAN-100639)

-h and -p options were deprecated in OpenLDAP client utilities.

The upstream OpenLDAP project has deprecated the **-h** and **-p** options in its utilities, and recommends using the **-H** option instead to specify the LDAP URI. As a consequence, RHEL 9 has deprecated these two options in all OpenLDAP client utilities. The **-h** and **-p** options will be removed from RHEL products in future releases.

(JIRA:RHELPLAN-137660)

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

10.10. DESKTOP

GTK 2 is now deprecated

The legacy GTK 2 toolkit and the following, related packages have been deprecated:

- **adwaita-gtk2-theme**
- **gnome-common**
- **gtk2**
- **gtk2-immodules**
- **hexchat**

Several other packages currently depend on GTK 2. These have been modified so that they no longer depend on the deprecated packages in a future major RHEL release.

If you maintain an application that uses GTK 2, Red Hat recommends that you port the application to GTK 4.

(JIRA:RHELPLAN-131882)

10.11. GRAPHICS INFRASTRUCTURES

X.org Server is now deprecated

The **X.org** display server is deprecated, and will be removed in a future major RHEL release. The default desktop session is now the **Wayland** session in most cases.

The **X11** protocol remains fully supported using the **XWayland** back end. As a result, applications that require **X11** can run in the **Wayland** session.

Red Hat is working on resolving the remaining problems and gaps in the **Wayland** session. For the outstanding problems in **Wayland**, see the Known issues section.

You can switch your user session back to the **X.org** back end. For more information, see Selecting GNOME environment and display protocol.

(JIRA:RHELPLAN-121048)

Motif has been deprecated

The Motif widget toolkit has been deprecated in RHEL, because development in the upstream Motif community is inactive.

The following Motif packages have been deprecated, including their development and debugging variants:

- **motif**
- **openmotif**
- **openmotif21**
- **openmotif22**

Additionally, the **motif-static** package has been removed.

Red Hat recommends using the GTK toolkit as a replacement. GTK is more maintainable and provides new features compared to Motif.

(JIRA:RHELPLAN-98983)

10.12. RED HAT ENTERPRISE LINUX SYSTEM ROLES

The networking system role displays a deprecation warning when configuring teams on RHEL 9 nodes

The network teaming capabilities have been deprecated in RHEL 9. As a result, using the **networking** RHEL system role on an RHEL 8 controller to configure a network team on RHEL 9 nodes, shows a warning about its deprecation.

(BZ#1999770)

10.13. VIRTUALIZATION

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 SHA2 algorithm, or later.

(BZ#1935497)

Limited support for virtual machine snapshots

Creating snapshots of virtual machines (VMs) is currently only supported for VMs not using the UEFI firmware. In addition, during the snapshot operation, the QEMU monitor may become blocked, which negatively impacts the hypervisor performance for certain workloads.

Also note that the current mechanism of creating VM snapshots has been deprecated, and Red Hat does not recommend using VM snapshots in a production environment. However, a new VM snapshot

mechanism is under development and is planned to be fully implemented in a future minor release of RHEL 9.

(JIRA:RHELPLAN-15509, BZ#1621944)

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** may not be yet available in the RHEL web console.

(JIRA:RHELPLAN-10304)

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)

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 9.

(BZ#1965079)

qcow2-v2 image format is deprecated

With RHEL 9, 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 9 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.

(BZ#1951814)

Legacy CPU models are now deprecated

A significant number of CPU models have become deprecated and will become unsupported for use in virtual machines (VMs) in a future major release of RHEL. The deprecated models are as follows:

- For Intel: models prior to Intel Xeon 55xx and 75xx Processor families (also known as Nehalem)
- For AMD: models prior to AMD Opteron G4
- For IBM Z: models prior to IBM z14

To check whether your 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 'i486'
```

-

(BZ#2060839)

10.14. CONTAINERS

Running RHEL 9 containers on a RHEL 7 host is not supported

Running RHEL 9 containers on a RHEL 7 host is not supported. It might work, but it is not guaranteed.

For more information, see Red Hat Enterprise Linux Container Compatibility Matrix .

(JIRA:RHELPLAN-100087)

SHA1 hash algorithm within Podman has been deprecated

The SHA1 algorithm used to generate the filename of the rootless network namespace is no longer supported in Podman. Therefore, rootless containers started before updating to Podman 4.1.1 or later have to be restarted if they are joined to a network (and not just using **slirp4netns**) to ensure they can connect to containers started after the upgrade.

(BZ#2069279)

rhel9/pause has been deprecated

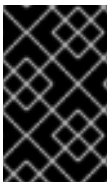
The **rhel9/pause** container image has been deprecated.

(BZ#2106816)

10.15. 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.

For changes to packages between RHEL 8 and RHEL 9, see Changes to packages in the *Considerations in adopting RHEL 9* document.



IMPORTANT

The support status of deprecated packages remains unchanged within RHEL 9. For more information about the length of support, see Red Hat Enterprise Linux Life Cycle and Red Hat Enterprise Linux Application Streams Life Cycle .

The following packages have been deprecated in RHEL 9:

- iptables-devel
- iptables-libs
- iptables-nft
- iptables-nft-services
- iptables-utils
- libdb

- mcpp
- mod_auth_mellon
- python3-pytz
- xorg-x11-server-Xorg

CHAPTER 11. KNOWN ISSUES

This part describes known issues in Red Hat Enterprise Linux 9.1.

11.1. INSTALLER AND IMAGE CREATION

The reboot --kexec and inst.kexec commands do not provide a predictable system state

Performing a RHEL installation with the **reboot --kexec** Kickstart command or the **inst.kexec** kernel boot parameters do not provide the same predictable system state as a full reboot. As a consequence, switching to the installed system without rebooting can produce unpredictable results.

Note that the **kexec** feature is deprecated and will be removed in a future release of Red Hat Enterprise Linux.

(BZ#1697896)

Local Media installation source is not detected when booting the installation from a USB that is created using a third party tool

When booting the RHEL installation from a USB that is created using a third party tool, the installer fails to detect the **Local Media** installation source (only *Red Hat CDN* is detected).

This issue occurs because the default boot option **int.stage2=** attempts to search for **iso9660** image format. However, a third party tool might create an ISO image with a different format.

As a workaround, use either of the following solution:

- When booting the installation, click the **Tab** key to edit the kernel command line, and change the boot option **inst.stage2=** to **inst.repo=**.
- To create a bootable USB device on Windows, use Fedora Media Writer.
- When using a third party tool like Rufus to create a bootable USB device, first regenerate the RHEL ISO image on a Linux system, and then use the third party tool to create a bootable USB device.

For more information on the steps involved in performing any of the specified workaround, see, Installation media is not auto detected during the installation of RHEL 8.3 .

(BZ#1877697)

The auth and authconfig Kickstart commands require the AppStream repository

The **authselect-compat** package is required by the **auth** and **authconfig** Kickstart commands during installation. Without this package, the installation fails if **auth** or **authconfig** are used. However, by design, the **authselect-compat** package is only available in the AppStream repository.

To work around this problem, verify that the BaseOS and AppStream repositories are available to the installer or use the **authselect** Kickstart command during installation.

(BZ#1640697)

Driver disk menu fails to display user inputs on the console

When you start RHEL installation using the **inst.dd** option on the Kernel command line with a driver disk, the console fails to display the user input. Consequently, it appears that the application does not

respond to the user input and freezes, 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**.

As a workaround, to see the expected results, ignore the absence of user inputs in the console and press **Enter** when you finish adding inputs.

(BZ#2109231)

Unexpected SELinux policies on systems where Anaconda is running as an application

When Anaconda is running as an application on an already installed system (for example to perform another installation to an image file using the **–image** anaconda option), the system is not prohibited to modify the SELinux types and attributes during installation. As a consequence, certain elements of SELinux policy might change on the system where Anaconda is running. To work around this problem, do not run Anaconda on the production system and execute it in a temporary virtual machine. So that the SELinux policy on a production system is not modified. Running anaconda as part of the system installation process such as installing from **boot.iso** or **dvd.iso** is not affected by this issue.

(BZ#2050140)

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.

To work around this problem, use the **harddrive --partition=sdX --dir=**/ command to install from USB CD-ROM drive. As a result, the installation does not fail.

(BZ#1914955)

Hard drive partitioned installations with iso9660 filesystem fails

You cannot install RHEL on systems where the hard drive is partitioned with the **iso9660** filesystem. This is due to the updated installation code that is set to ignore any hard disk containing a **iso9660** file system partition. This happens even when RHEL is installed without using a DVD.

To workaround this problem, add the following script in the kickstart file to format the disc before the installation starts.

Note: Before performing the workaround, backup the data available on the disk. The **wipefs** command formats all the existing data from the disk.

```
%pre
wipefs -a /dev/sda
%end
```

As a result, installations work as expected without any errors.

(BZ#1929105)

Anaconda fails to verify existence of an administrator user account

While installing RHEL using a graphical user interface, Anaconda fails to verify if the administrator account has been created. As a consequence, users might install a system without any administrator user account.

To work around this problem, ensure you configure an administrator user account or the root password is set and the root account is unlocked. As a result, users can perform administrative tasks on the installed system.

(BZ#2047713)

New XFS features prevent booting of PowerNV IBM POWER systems with firmware older than version 5.10

PowerNV IBM POWER systems use a Linux kernel for firmware, and use Petitboot as a replacement for GRUB. This results in the firmware kernel mounting **/boot** and Petitboot reading the GRUB config and booting RHEL.

The RHEL 9 kernel introduces **bigtime=1** and **inobtcount=1** features to the XFS filesystem, which kernels with firmware older than version 5.10 do not understand.

To work around this problem, you can use another filesystem for **/boot**, for example ext4.

(BZ#1997832)

Cannot install RHEL when PReP is not 4 or 8 MiB in size

The RHEL installer cannot install the boot loader if the PowerPC Reference Platform (PReP) partition is of a different size than 4 MiB or 8 MiB on a disk that uses 4 kiB sectors. As a consequence, you cannot install RHEL on the disk.

To work around the problem, make sure that the PReP partition is exactly 4 MiB or 8 MiB in size, and that the size is not rounded to another value. As a result, the installer can now install RHEL on the disk.

(BZ#2026579)

The installer displays an incorrect total disk space while custom partitioning with multipath devices

The installer does not filter out individual paths of multipath devices while custom partitioning. This causes the installer to display individual paths to multipath devices and users can select individual paths to multipath devices for the created partitions. As a consequence, an incorrect sum of the total disk space is displayed. It is computed by adding the size of each individual path to the total disk space.

As a workaround, use only the multipath devices and not individual paths while custom partitioning, and ignore the incorrectly computed total disk space.

(BZ#2052938)

RHEL for Edge installer image fails to create mount points when installing an rpm-ostree payload

When deploying **rpm-ostree** payloads, used for example in a RHEL for Edge installer image, the installer does not properly create some mount points for custom partitions. As a consequence, the installation is aborted with the following error:

```
The command 'mount --bind /mnt/sysimage/data /mnt/sysroot/data' exited with the code 32.
```

To work around this issue:

- Use an automatic partitioning scheme and do not add any mount points manually.

- Manually assign mount points only inside /**var** directory. For example, /**var**/***my-mount-point***), and the following standard directories: /, /**boot**, /**var**.

As a result, the installation process finishes successfully.

(BZ#2125542)

NetworkManager fails to start after the installation when connected to a network but without DHCP or a static IP address configured

Starting with RHEL 9.0, Anaconda activates network devices automatically when there is no specific **ip=** or kickstart network configuration set. Anaconda creates a default persistent configuration file for each Ethernet device. The connection profile has the **ONBOOT** and **autoconnect** value set to **true**. As a consequence, during the start of the installed system, RHEL activates the network devices, and the **networkManager-wait-online** service fails.

As a workaround, do one of the following:

- Delete all connections using the **nmcli** utility except one connection you want to use. For example:

  a. List all connection profiles:

  ```
  # nmcli connection show
  ```

  b. Delete the connection profiles that you do not require:

  ```
  # nmcli connection delete <connection_name>
  ```

  Replace <connection_name> with the name of the connection you want to delete.

- Disable the auto connect network feature in Anaconda if no specific **ip=** or kickstart network configuration is set.

  a. In the Anaconda GUI, navigate to **Network & Host Name**

  b. Select a network device to disable.

  c. Click **Configure**.

  d. On the **General** tab, deselect the **Connect automatically with priority**

  e. Click **Save**.

(BZ#2115783)

RHEL installer does not process the inst.proxy boot option correctly

When running Anaconda, the installation program does not process the **inst.proxy** boot option correctly. As a consequence, you cannot use the specified proxy to fetch the installation image.

To work around this issue: * Use the latest version of RHEL distribution. * Use **proxy** instead of **inst.proxy** boot option.

(JIRA:RHELDOCS-18764)

RHEL installation fails on IBM Z architectures with multi-LUNs

RHEL installation fails on IBM Z architectures when using multiple LUNs during installation. Due to the multipath setup of FCP and the LUN auto-scan behavior, the length of the kernel command line in the configuration file exceeds 896 bytes.

To work around this problem, you can do one of the following:

- Install the latest version of RHEL (RHEL 9.2 or later).
- Install the RHEL system with a single LUN and add additional LUNs post installation.
- Optimize the redundant **zfcp** entries in the boot configuration on the installed system.
- Create a physical volume (**pvcreate**) for each of the additional LUNs listed under **/dev/mapper/**.
- Extend the VG with PVs, for example, **vgextend <vg_name> /dev/mapper/mpathX**.
- Increase the LV as needed for example, **lvextend -r -l +100%FREE /dev/<vg name>/root**.

For more information, see the KCS solution.

(JIRA:RHELDOCS-18638)

RHEL installer does not automatically discover or use iSCSI devices as boot devices on aarch64

The absence of the **iscsi_ibft** kernel module in RHEL installers running on aarch64 prevents automatic discovery of iSCSI devices defined in firmware. These devices are not automatically visible in the installer nor selectable as boot devices when added manually by using the GUI. As a workaround, add the "inst.nonibftiscsiboot" parameter to the kernel command line when booting the installer and then manually attach iSCSI devices through the GUI. As a result, the installer can recognize the attached iSCSI devices as bootable and installation completes as expected.

For more information, see KCS solution.

(JIRA:RHEL-56135)

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.

As a 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-13837)

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. To work around this problem, use the **firewall --disabled** command instead. As a result, the **firewalld** service is disabled properly.

(JIRA:RHEL-82566)

11.2. SUBSCRIPTION MANAGEMENT

The subscription-manager utility retains nonessential text in the terminal after completing a command

Starting with RHEL 9.1, the **subscription-manager** utility displays progress information while processing an operation. For some languages (typically non-Latin), progress messages might not be cleared after the operation finishes. As a result, you might see parts of old progress messages in the terminal.

Note that this is not a functional failure for **subscription-manager**.

To work around this problem, perform either of the following steps:

- Include the **--no-progress-messages** option when running `subscription-manager`commands in the terminal
- Configure **subscription-manager** to operate without displaying progress messages by entering the following command:

```
# subscription-manager config --rhsm.progress_messages=0
```

(BZ#2136694)

11.3. SOFTWARE MANAGEMENT

The Installation process sometimes becomes unresponsive

When you install RHEL, the installation process sometimes becomes unresponsive. The **/tmp/packaging.log** file displays the following message at the end:

```
10:20:56,416 DDEBUG dnf: RPM transaction over.
```

To workaround this problem, restart the installation process.

(BZ#2073510)

A security DNF upgrade fails for packages that change their architecture through the upgrade

The patch for BZ#2108969, released with the RHBA-2022:8295 advisory, introduced the following regression: The DNF upgrade using security filters fails for packages that change their architecture from or to **noarch** through the upgrade. Consequently, it can leave the system in a vulnerable state.

To work around this problem, perform the regular upgrade without security filters.

(BZ#2108969)

11.4. SHELLS AND COMMAND-LINE TOOLS

ReaR fails during recovery if the TMPDIR variable is set in the configuration file

Setting and exporting **TMPDIR** in the **/etc/rear/local.conf** or **/etc/rear/site.conf** ReaR configuration file does not work and is deprecated.

The ReaR default configuration file **/usr/share/rear/conf/default.conf** contains the following instructions:

```
# To have a specific working area directory prefix for Relax-and-Recover
# specify in /etc/rear/local.conf something like
#
# export TMPDIR="/prefix/for/rear/working/directory"
#
# where /prefix/for/rear/working/directory must already exist.
# This is useful for example when there is not sufficient free space
# in /tmp or $TMPDIR for the ISO image or even the backup archive.
```

The instructions mentioned above do not work correctly because the **TMPDIR** variable has the same value in the rescue environment, which is not correct if the directory specified in the **TMPDIR** variable does not exist in the rescue image.

As a consequence, setting and exporting **TMPDIR** in the **/etc/rear/local.conf** file leads to the following error when the rescue image is booted :

```
mktemp: failed to create file via template '/prefix/for/rear/working/directory/tmp.XXXXXXXXXX': No
such file or directory
cp: missing destination file operand after '/etc/rear/mappings/mac'
Try 'cp --help' for more information.
No network interface mapping is specified in /etc/rear/mappings/mac
```

or the following error and abort later, when running **rear recover**:

```
ERROR: Could not create build area
```

To work around this problem, if you want to have a custom temporary directory, specify a custom directory for ReaR temporary files by exporting the variable in the shell environment before executing ReaR. For example, execute the **export TMPDIR=...** statement and then execute the **rear** command in the same shell session or script. As a result, the recovery is successful in the described configuration.

Jira:RHEL-24847

Renaming network interfaces using ifcfg files fails

On RHEL 9, the **initscripts** package is not installed by default. Consequently, renaming network interfaces using **ifcfg** files fails. To solve this problem, Red Hat recommends that you use **udev** rules or link files to rename interfaces. For further details, see Consistent network interface device naming and the **systemd.link(5)** man page.

If you cannot use one of the recommended solutions, install the **initscripts** package.

(BZ#2018112)

The chkconfig package is not installed by default in RHEL 9

The **chkconfig** package, which updates and queries runlevel information for system services, is not installed by default in RHEL 9.

To manage services, use the **systemctl** commands or install the **chkconfig** package manually.

For more information about **systemd**, see Managing systemd. For instructions on how to use the **systemctl** utility, see Managing system services with systemctl.

(BZ#2053598)

11.5. INFRASTRUCTURE SERVICES

Both bind and unbound disable validation of SHA-1-based signatures

The **bind** and **unbound** components disable validation support of all RSA/SHA1 (algorithm number 5) and RSASHA1-NSEC3-SHA1 (algorithm number 7) signatures, and the SHA-1 usage for signatures is restricted in the DEFAULT system-wide cryptographic policy.

As a result, certain DNSSEC records signed with the SHA-1, RSA/SHA1, and RSASHA1-NSEC3-SHA1 digest algorithms fail to verify in Red Hat Enterprise Linux 9 and the affected domain names become vulnerable.

To work around this problem, upgrade to a different signature algorithm, such as RSA/SHA-256 or elliptic curve keys.

For more information and a list of top-level domains that are affected and vulnerable, see the DNSSEC records signed with RSASHA1 fail to verify solution.

(BZ#2070495)

named fails to start if the same writable zone file is used in multiple zones

BIND does not allow the same writable zone file in multiple zones. Consequently, if a configuration includes multiple zones which share a path to a file that can be modified by the **named** service, **named** fails to start. To work around this problem, use the **in-view** clause to share one zone between multiple views and make sure to use different paths for different zones. For example, include the view names in the path.

Note that writable zone files are typically used in zones with allowed dynamic updates, slave zones, or zones maintained by DNSSEC.

(BZ#1984982)

Setting the console keymap requires the libxkbcommon library on your minimal install

In RHEL 9, certain **systemd** library dependencies have been converted from dynamic linking to dynamic loading, so that your system opens and uses the libraries at runtime when they are available. With this change, a functionality that depends on such libraries is not available unless you install the necessary library. This also affects setting the keyboard layout on systems with a minimal install. As a result, the **localectl --no-convert set-x11-keymap gb** command fails.

To work around this problem, install the **libxkbcommon** library:

```
# dnf install libxkbcommon
```

(BZ#2214130)

11.6. SECURITY

OpenSSL does not detect if a PKCS #11 token supports the creation of raw RSA or RSA-PSS signatures

The TLS 1.3 protocol requires support for RSA-PSS signatures. If a PKCS #11 token does not support raw RSA or RSA-PSS signatures, server applications that use the **OpenSSL** library fail to work with an **RSA**

key if the key is held by the **PKCS #11** token. As a result, TLS communication fails in the described scenario.

To work around this problem, configure servers and clients to use TLS version 1.2 as the highest TLS protocol version available.

(BZ#1681178)

OpenSSL incorrectly handles PKCS #11 tokens that does not support raw RSA or RSA-PSS signatures

The **OpenSSL** library does not detect key-related capabilities of PKCS #11 tokens. Consequently, establishing a TLS connection fails when a signature is created with a token that does not support raw RSA or RSA-PSS signatures.

To work around the problem, add the following lines after the **.include** line at the end of the **crypto_policy** section in the **/etc/pki/tls/openssl.cnf** file:

```
SignatureAlgorithms =
RSA+SHA256:RSA+SHA512:RSA+SHA384:ECDSA+SHA256:ECDSA+SHA512:ECDSA+SHA384
MaxProtocol = TLSv1.2
```

As a result, a TLS connection can be established in the described scenario.

(BZ#1685470)

scp empties files copied to themselves when a specific syntax is used

The **scp** utility changed from the Secure copy protocol (SCP) to the more secure SSH file transfer protocol (SFTP). Consequently, copying a file from a location to the same location erases the file content. The problem affects the following syntax:

scp localhost:/myfile localhost:/myfile

To work around this problem, do not copy files to a destination that is the same as the source location using this syntax.

The problem has been fixed for the following syntaxes:

- **scp /myfile localhost:/myfile**
- **scp localhost:~/myfile ~/myfile**

(BZ#2056884)

PSK ciphersuites do not work with the FUTURE crypto policy

Pre-shared key (PSK) ciphersuites are not recognized as performing perfect forward secrecy (PFS) key exchange methods. As a consequence, the **ECDHE-PSK** and **DHE-PSK** ciphersuites do not work with OpenSSL configured to **SECLEVEL=3**, for example with the **FUTURE** crypto policy. As a workaround, you can set a less restrictive crypto policy or set a lower security level (**SECLEVEL**) for applications that use PSK ciphersuites.

(BZ#2060044)

GnuPG incorrectly allows using SHA-1 signatures even if disallowed by crypto-policies

The GNU Privacy Guard (GnuPG) cryptographic software can create and verify signatures that use the

SHA-1 algorithm regardless of the settings defined by the system-wide cryptographic policies. Consequently, you can use SHA-1 for cryptographic purposes in the **DEFAULT** cryptographic policy, which is not consistent with the system-wide deprecation of this insecure algorithm for signatures.

To work around this problem, do not use GnuPG options that involve SHA-1. As a result, you will prevent GnuPG from lowering the default system security by using the non-secure SHA-1 signatures.

(BZ#2070722)

gpg-agent does not work as an SSH agent in FIPS mode

The **gpg-agent** tool creates MD5 fingerprints when adding keys to the **ssh-agent** program even though FIPS mode disables the MD5 digest. Consequently, the **ssh-add** utility fails to add the keys to the authentication agent.

To work around the problem, create the **~/.gnupg/sshcontrol** file without using the **gpg-agent --daemon --enable-ssh-support** command. For example, you can paste the output of the **gpg --list-keys** command in the **<FINGERPRINT> 0** format to **~/.gnupg/sshcontrol**. As a result, **gpg-agent** works as an SSH authentication agent.

(BZ#2073567)

Default SELinux policy allows unconfined executables to make their stack executable

The default state of the **selinuxuser_execstack** boolean in the SELinux policy is on, which means that unconfined executables can make their stack executable. Executables should not use this option, and it might indicate poorly coded executables or a possible attack. However, due to compatibility with other tools, packages, and third-party products, Red Hat cannot change the value of the boolean in the default policy. If your scenario does not depend on such compatibility aspects, you can turn the boolean off in your local policy by entering the command **setsebool -P selinuxuser_execstack off**.

(BZ#2064274)

Remediating service-related rules during kickstart installations might fail

During a kickstart installation, the OpenSCAP utility sometimes incorrectly shows that a service **enable** or **disable** state remediation is not needed. Consequently, OpenSCAP might set the services on the installed system to a non-compliant state. As a workaround, you can scan and remediate the system after the kickstart installation. This will fix the service-related issues.

(BZ#1834716)

Remediation of SCAP Audit rules fails incorrectly

Bash remediation of some SCAP rules related to Audit configuration does not add the Audit key when remediating. This applies to the following rules:

- **audit_rules_login_events**
- **audit_rules_login_events_faillock**
- **audit_rules_login_events_lastlog**
- **audit_rules_login_events_tallylog**
- **audit_rules_usergroup_modification**
- **audit_rules_usergroup_modification_group**

- **audit_rules_usergroup_modification_gshadow**
- **audit_rules_usergroup_modification_opasswd**
- **audit_rules_usergroup_modification_passwd**
- **audit_rules_usergroup_modification_shadow**
- **audit_rules_time_watch_localtime**
- **audit_rules_mac_modification**
- **audit_rules_networkconfig_modification**
- **audit_rules_sysadmin_actions**
- **audit_rules_session_events**
- **audit_rules_sudoers**
- **audit_rules_sudoers_d**

In consequence, if the relevant Audit rule already exists but does not fully conform to the OVAL check, the remediation fixes the functional part of the Audit rule, that is, the path and access bits, but does not add the Audit key. Therefore, the resulting Audit rule works correctly, but the SCAP rule incorrectly reports FAIL. To work around this problem, add the correct keys to the Audit rules manually.

(BZ#2120978)

SSH timeout rules in STIG profiles configure incorrect options

An update of OpenSSH affected the rules in the following Defense Information Systems Agency Security Technical Implementation Guide (DISA STIG) profiles:

- DISA STIG for RHEL 9 (**xccdf_org.ssgproject.content_profile_stig**)
- DISA STIG with GUI for RHEL 9 (**xccdf_org.ssgproject.content_profile_stig_gui**)

In each of these profiles, the following two rules are affected:

```
Title: Set SSH Client Alive Count Max to zero
CCE Identifier: CCE-90271-8
Rule ID: xccdf_org.ssgproject.content_rule_sshd_set_keepalive_0

Title: Set SSH Idle Timeout Interval
CCE Identifier: CCE-90811-1
Rule ID: xccdf_org.ssgproject.content_rule_sshd_set_idle_timeout
```

When applied to SSH servers, each of these rules configures an option (**ClientAliveCountMax** and **ClientAliveInterval**) that no longer behaves as previously. As a consequence, OpenSSH no longer disconnects idle SSH users when it reaches the timeout configured by these rules. As a workaround, these rules have been temporarily removed from the DISA STIG for RHEL 9 and DISA STIG with GUI for RHEL 9 profiles until a solution is developed.

(BZ#2038978)

Keylime might fail attestation of systems that access multiple IMA-measured files

If a system that runs the Keylime agent accesses multiple files measured by the Integrity Measurement Architecture (IMA) in quick succession, the Keylime verifier might incorrectly process the IMA log additions. As a consequence, the running hash does not match the correct Platform Configuration Register (PCR) state, and the system fails attestation. There is currently no workaround.

(BZ#2138167)

Keylime measured boot policy generation script might cause a segmentation fault and core dump

The **create_mb_refstate** script, which generates policies for measure boot attestation in Keylime, might incorrectly calculate the data length in the **DevicePath** field instead of using the value of the **LengthOfDevicePath** field when handling the output of the **tpm2_eventlog** tool depending on the input provided. As a consequence, the script tries to access invalid memory using the incorrectly calculated length, which results in a segmentation fault and core dump. The main functionality of Keylime is not affected by this problem, but you might be unable to generate a measured boot policy.

To work around this problem, do not use a measured boot policy or write the policy file manually from the data obtained using the **tpm2_eventlog** tool from the **tpm2-tools** package.

(BZ#2140670)

Some TPM certificates cause Keylime registrar to crash

The **require_ek_cert** configuration option in **tenant.conf**, which should be enabled in production deployments, determines whether the Keylime tenant requires an endorsement key (EK) certificate from the Trusted Platform Module (TPM). When performing the initial identity quote with **require_ek_cert** enabled, Kelime attempts to verify whether the TPM device on the agent is genuine by comparing the EK certificate against the trusted certificates present in the Keylime TPM certificate store. However, some certificates in the store are malformed x509 certificates and cause the Keylime registrar to crash. There is currently no simple workaround to this problem, except for setting **require_ek_cert** to **false**, and defining a custom script in the **ek_check_script** option that will perform EK validation.

(BZ#2142009)

OpenSSH in RHEL 9.0-9.3 is not compatible with OpenSSL 3.2.2

The **openssh** packages provided by RHEL 9.0, 9.1, 9.2, and 9.3 strictly check for the OpenSSL version. Consequently, if you upgrade the **openssl** packages to version 3.2.2 and higher and you keep the **openssh** packages in version 8.7p1-34.el9_3.3 or earlier, the **sshd** service fails to start with an **OpenSSL version mismatch** error message.

To work around this problem, upgrade the **openssh** packages to version 8.7p1-38.el9 and later. See the sshd not working, OpenSSL version mismatch solution (Red Hat Knowledgebase) for more information.

(JIRA:RHELDOCS-19626)

11.7. NETWORKING

The nm-cloud-setup service removes manually-configured secondary IP addresses from interfaces

Based on the information received from the cloud environment, the **nm-cloud-setup** service configures network interfaces. Disable **nm-cloud-setup** to manually configure interfaces. However, in certain cases, other services on the host can configure interfaces as well. For example, these services could add secondary IP addresses. To avoid that **nm-cloud-setup** removes secondary IP addresses:

1. Stop and disable the **nm-cloud-setup** service and timer:

```
# systemctl disable --now nm-cloud-setup.service nm-cloud-setup.timer
```

2. Display the available connection profiles:

```
# nmcli connection show
```

3. Reactive the affected connection profiles:

```
# nmcli connection up "<profile_name>"
```

As a result, the service no longer removes manually-configured secondary IP addresses from interfaces.

(BZ#2151040)

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. To work around this problem, disable kTLS. As a result, with the workaround, it is possible to successfully update the session key.

(BZ#2013650)

The initscripts package is not installed by default

By default, the **initscripts** package is not installed. As a consequence, the **ifup** and **ifdown** utilities are not available. As an alternative, use the **nmcli connection up** and **nmcli connection down** commands to enable and disable connections. If the suggested alternative does not work for you, report the problem and install the **NetworkManager-initscripts-updown** package, which provides a NetworkManager solution for the **ifup** and **ifdown** utilities.

(BZ#2082303)

11.8. KERNEL

The mlx5 driver fails while using Mellanox ConnectX-5 adapter

In Ethernet switch device driver model (**switchdev**) mode, **mlx5** driver fails when configured with device managed flow steering (DMFS) parameter and **ConnectX-5** adapter supported hardware. As a consequence, you can see the following error message:

```
BUG: Bad page cache in process umount pfn:142b4b
```

To workaround this problem, you need to use the software managed flow steering (SMFS) parameter instead of DMFS.

(BZ#2180665)

FADump enabled with Secure Boot might lead to GRUB Out of Memory (OOM)

In the Secure Boot environment, GRUB and PowerVM together allocate a 512 MB memory region, known as the Real Mode Area (RMA), for boot memory. The region is divided among the boot components and, if any component exceeds its allocation, out-of-memory failures occur.

Generally, the default installed **initramfs** file system and the **vmlinux** symbol table are within the limits to avoid such failures. However, if Firmware Assisted Dump (FADump) is enabled in the system, the default **initramfs** size can increase and exceed 95 MB. As a consequence, every system reboot leads to a GRUB OOM state.

To avoid this issue, do not use Secure Boot and FADump together. For more information and methods on how to work around this issue, see https://www.ibm.com/support/pages/node/6846531.

(BZ#2149172)

weak-modules from kmod fails to work with module inter-dependencies

The **weak-modules** script provided by the **kmod** package determines which modules are kABI-compatible with installed kernels. However, while checking modules' kernel compatibility, **weak-modules** processes modules symbol dependencies from higher to lower release of the kernel for which they were built. As a consequence, modules with inter-dependencies built against different kernel releases might be interpreted as non-compatible, and therefore the **weak-modules** script fails to work in this scenario.

To work around the problem, build or put the extra modules against the latest stock kernel before you install the new kernel.

(BZ#2103605)

The kdump service fails to build the initrd file on IBM Z systems

On the 64-bit IBM Z systems, the **kdump** service fails to load the initial RAM disk (**initrd**) when **znet** related configuration information such as **s390-subchannels** reside in an inactive **NetworkManager** connection profile. Consequently, the **kdump** mechanism fails with the following error:

```
dracut: Failed to set up znet
kdump: mkdumprd: failed to make kdump initrd
```

As a workaround, use one of the following solutions:

- Configure a network bond or bridge by re-using the connection profile that has the **znet** configuration information:

  ```
  $ nmcli connection modify enc600 master bond0 slave-type bond
  ```

- Copy the **znet** configuration information from the inactive connection profile to the active connection profile:

  a. Run the **nmcli** command to query the **NetworkManager** connection profiles:

  ```
  # nmcli connection show

  NAME                UUID              TYPE   Device

  bridge-br0          ed391a43-bdea-4170-b8a2 bridge   br0
  bridge-slave-enc600  caf7f770-1e55-4126-a2f4 ethernet enc600
  enc600              bc293b8d-ef1e-45f6-bad1 ethernet --
  ```

  b. Update the active profile with configuration information from the inactive connection:

  ```
  #!/bin/bash
  ```

```
inactive_connection=enc600
active_connection=bridge-slave-enc600
for name in nettype subchannels options; do
field=802-3-ethernet.s390-$name
val=$(nmcli --get-values "$field"connection show "$inactive_connection")
nmcli connection modify "$active_connection" "$field" $val"
done
```

c. Restart the **kdump** service for changes to take effect:

```
# kdumpctl restart
```

(BZ#2064708)

The kdump mechanism fails to capture the vmcore file on LUKS-encrypted targets

When running **kdump** on systems with Linux Unified Key Setup (LUKS) encrypted partitions, systems require a certain amount of available memory. When the available memory is less than the required amount of memory, the **systemd-cryptsetup** service fails to mount the partition. Consequently, the second kernel fails to capture the crash dump file (**vmcore**) on LUKS-encrypted targets.

With the **kdumpctl estimate** command, you can query the **Recommended crashkernel value**, which is the recommended memory size required for **kdump**.

To work around this problem, use following steps to configure the required memory for **kdump** on LUKS encrypted targets:

1. Print the estimate **crashkernel** value:

```
# kdumpctl estimate
```

2. Configure the amount of required memory by increasing the **crashkernel** value:

```
# grubby --args=crashkernel=652M --update-kernel=ALL
```

3. Reboot the system for changes to take effect.

```
# reboot
```

As a result, **kdump** works correctly on systems with LUKS-encrypted partitions.

(BZ#2017401)

Allocating crash kernel memory fails at boot time

On certain Ampere Altra systems, allocating the crash kernel memory for **kdump** usage fails during boot when the available memory is below 1 GB. Consequently, the **kdumpctl** command fails to start the **kdump** service.

To workaround this problem, do one of the following:

- Decrease the value of the **crashkernel** parameter by a minimum of 240 MB to fit the size requirement, for example **crashkernel=240M**.
- Use the **crashkernel=x,high** option to reserve crash kernel memory above 4 GB for **kdump**.

As a result, the crash kernel memory allocation for **kdump** does not fail on Ampere Altra systems.

(BZ#2065013)

The Delay Accounting functionality does not display the SWAPIN and IO% statistics columns by default

The **Delayed Accounting** functionality, unlike early versions, is disabled by default. Consequently, the **iotop** application does not show the **SWAPIN** and **IO%** statistics columns and displays the following warning:

```
CONFIG_TASK_DELAY_ACCT not enabled in kernel, cannot determine SWAPIN and IO%
```

The **Delay Accounting** functionality, using the **taskstats** interface, provides the delay statistics for all tasks or threads that belong to a thread group. Delays in task execution occur when they wait for a kernel resource to become available, for example, a task waiting for a free CPU to run on. The statistics help in setting a task's CPU priority, I/O priority, and **rss** limit values appropriately.

As a workaround, you can enable the **delayacct** boot option either at runtime or boot.

- To enable **delayacct** at runtime, enter:

  ```
  echo 1 > /proc/sys/kernel/task_delayacct
  ```

  Note that this command enables the feature system wide, but only for the tasks that you start after running this command.

- To enable **delayacct** permanently at boot, use one of the following procedures:
  - Edit the **/etc/sysctl.conf** file to override the default parameters:
    a. Add the following entry to the **/etc/sysctl.conf** file:

    ```
    kernel.task_delayacct = 1
    ```

    For more information, see How to set sysctl variables on Red Hat Enterprise Linux .

    b. Reboot the system for changes to take effect.

  - Edit the GRUB 2 configuration file to override the default parameters:
    a. Append the **delayacct** option to the **/etc/default/grub** file's **GRUB _CMDLINE_LINUX** entry.

    b. Run the **grub2-mkconfig** utility to regenerate the boot configuration:

    ```
    # grub2-mkconfig -o /boot/grub2/grub.cfg
    ```

    For more information, see How do I permanently modify the kernel command line? .

    c. Reboot the system for changes to take effect.

As a result, the **iotop** application displays the **SWAPIN** and **IO%** statistics columns.

(BZ#2132480)

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. To work around this problem, 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.

(BZ#2000616)

The iwl7260-firmware breaks Wi-Fi on Intel Wi-Fi 6 AX200, AX210, and Lenovo ThinkPad P1 Gen 4

After updating the **iwl7260-firmware** or **iwl7260-wifi** driver to the version provided by RHEL 8.7 and/or RHEL 9.1 (and later), the hardware gets into an incorrect internal state. reports its state incorrectly. Consequently, Intel Wifi 6 cards may not work and display the error message:

```
kernel: iwlwifi 0000:09:00.0: Failed to start RT ucode: -110
kernel: iwlwifi 0000:09:00.0: WRT: Collecting data: ini trigger 13 fired (delay=0ms)
kernel: iwlwifi 0000:09:00.0: Failed to run INIT ucode: -110
```

An unconfirmed work around is to power off the system and back on again. Do not reboot.

(BZ#2129288)

dkms provides an incorrect warning on program failure with correctly compiled drivers on 64-bit ARM CPUs

The Dynamic Kernel Module Support (**dkms**) utility does not recognize that the kernel headers for 64-bit ARM CPUs work for both the kernels with 4 kilobytes and 64 kilobytes page sizes. As a result, when the kernel update is performed and the **kernel-64k-devel** package is not installed, **dkms** provides an incorrect warning on why the program failed on correctly compiled drivers. To work around this problem, install the **kernel-headers** package, which contains header files for both types of ARM CPU architectures and is not specific to **dkms** and its requirements.

(JIRA:RHEL-25967)

11.9. BOOT LOADER

The behavior of grubby diverges from its documentation

When you add a new kernel using the **grubby** tool and do not specify any arguments, **grubby** passes the default arguments to the new entry. This behavior occurs even without passing the **--copy-default** argument. Using **--args** and **--copy-default** options ensures those arguments are appended to the default arguments as stated in the **grubby** documentation.

However, when you add additional arguments, such as **$tuned_params**, the **grubby** tool does not pass these arguments unless the **--copy-default** option is invoked.

In this situation, two workarounds are available:

- Either set the **root=** argument and leave **--args** empty:

```
# grubby --add-kernel /boot/my_kernel --initrd /boot/my_initrd --args "root=/dev/mapper/rhel-root" --title "entry_with_root_set"
```

- Or set the **root=** argument and the specified arguments, but not the default ones:

```
# grubby --add-kernel /boot/my_kernel --initrd /boot/my_initrd --args "root=/dev/mapper/rhel-root some_args and_some_more" --title "entry_with_root_set_and_other_args_too"
```

(BZ#2127453)

11.10. FILE SYSTEMS AND STORAGE

RHEL instances on Azure fail to boot if provisioned by cloud-init and configured with an NFSv3 mount entry

Currently, booting a RHEL virtual machine (VM) on the Microsoft Azure cloud platform fails if the VM was provisioned by the **cloud-init** tool and the guest operating system of the VM has an NFSv3 mount entry in the **/etc/fstab** file.

(BZ#2081114)

Anaconda fails to login iSCSI server using the no authentication method after unsuccessful CHAP authentication attempt

When you add iSCSI discs using CHAP authentication and the login attempt fails due to incorrect credentials, a relogin attempt to the discs with the **no authentication** method fails. To workaround this problem, close the current session and login using the **no authentication** method.

(BZ#1983602)

Device Mapper Multipath is not supported with NVMe/TCP

Using Device Mapper Multipath with the **nvme-tcp** driver can result in the Call Trace warnings and system instability. To work around this problem, NVMe/TCP users must enable native NVMe multipathing and not use the **device-mapper-multipath** tools with NVMe.

By default, Native NVMe multipathing is enabled in RHEL 9. For more information, see Enabling multipathing on NVMe devices.

(BZ#2033080)

The blk-availability systemd service deactivates complex device stacks

In **systemd**, the default block deactivation code does not always handle complex stacks of virtual block devices correctly. In some configurations, virtual devices might not be removed during the shutdown, which causes error messages to be logged. To work around this problem, deactivate complex block device stacks by executing the following command:

```
# systemctl enable --now blk-availability.service
```

As a result, complex virtual device stacks are correctly deactivated during shutdown and do not produce error messages.

(BZ#2011699)

supported_speeds sysfs attribute reports incorrect speed values

Previously, due to an incorrect definition in the **qla2xxx** driver, the **supported_speeds sysfs** attribute for the HBA reported 20 Gb/s speed instead of the expected 64 Gb/s speed. Consequently, if the HBA supported 64 Gb/s link speed, the **sysfs supported_speeds** value was incorrect, which affected the reported speed value.

But now the **supported_speeds** sysfs attribute for the HBA returns a 100 Gb/s speed instead of the intended 64 Gb/s, and 50 Gb/s speed instead of the intended 128 Gb/s speed. This only affects the reported speed value, and the actual link rates used on the Fibre connection are correct.

(BZ#2069758)

11.11. DYNAMIC PROGRAMMING LANGUAGES, WEB AND DATABASE SERVERS

The --ssl-fips-mode option in MySQL and MariaDB does not change FIPS mode

The **--ssl-fips-mode** option in **MySQL** and **MariaDB** in RHEL works differently than in upstream.

In RHEL 9, if you use **--ssl-fips-mode** as an argument for the **mysqld** or **mariadbd** daemon, or if you use **ssl-fips-mode** in the **MySQL** or **MariaDB** server configuration files, **--ssl-fips-mode** does not change FIPS mode for these database servers.

Instead:

- If you set **--ssl-fips-mode** to **ON**, the **mysqld** or **mariadbd** server daemon does not start.
- If you set **--ssl-fips-mode** to **OFF** on a FIPS-enabled system, the **mysqld** or **mariadbd** server daemons still run in FIPS mode.

This is expected because FIPS mode should be enabled or disabled for the whole RHEL system, not for specific components.

Therefore, do not use the **--ssl-fips-mode** option in **MySQL** or **MariaDB** in RHEL. Instead, ensure FIPS mode is enabled on the whole RHEL system:

- Preferably, install RHEL with FIPS mode enabled. Enabling FIPS mode during the installation ensures that the system generates all keys with FIPS-approved algorithms and continuous monitoring tests in place. For information about installing RHEL in FIPS mode, see Installing the system in FIPS mode.
- Alternatively, you can switch FIPS mode for the entire RHEL system by following the procedure in Switching the system to FIPS mode .

(BZ#1991500)

11.12. COMPILERS AND DEVELOPMENT TOOLS

Certain symbol-based probes do not work in SystemTap on the 64-bit ARM architecture

Kernel configuration disables certain functionality needed for **SystemTap**. Consequently, some symbol-based probes do not work on the 64-bit ARM architecture. As a result, affected **SystemTap** scripts may not run or may not collect hits on desired probe points.

Note that this bug has been fixed for the remaining architectures with the release of the RHBA-2022:5259 advisory.

(BZ#2083727)

11.13. IDENTITY MANAGEMENT

MIT Kerberos does not support ECC certificates for PKINIT

MIT Kerberos does not support ECC certificates for PKINIT

MIT Kerberos does not implement the RFC5349 request for comments document, which describes the design of elliptic-curve cryptography (ECC) support in Public Key Cryptography for initial authentication (PKINIT). Consequently, the MIT **krb5-pkinit** package, used by RHEL, does not support ECC certificates. For more information, see Elliptic Curve Cryptography (ECC) Support for Public Key Cryptography for Initial Authentication in Kerberos (PKINIT).

(BZ#2106043)

The DEFAULT:SHA1 sub-policy has to be set on RHEL 9 clients for PKINIT to work against AD KDCs

The SHA-1 digest algorithm has been deprecated in RHEL 9, and CMS messages for Public Key Cryptography for initial authentication (PKINIT) are now signed with the stronger SHA-256 algorithm.

However, the Active Directory (AD) Kerberos Distribution Center (KDC) still uses the SHA-1 digest algorithm to sign CMS messages. As a result, RHEL 9 Kerberos clients fail to authenticate users by using PKINIT against an AD KDC.

To work around the problem, enable support for the SHA-1 algorithm on your RHEL 9 systems with the following command:

```
# update-crypto-policies --set DEFAULT:SHA1
```

(BZ#2060798)

The PKINIT authentication of a user fails if a RHEL 9 Kerberos agent communicates with a non-RHEL-9 and non-AD Kerberos agent

If a RHEL 9 Kerberos agent, either a client or Kerberos Distribution Center (KDC), interacts with a non-RHEL-9 Kerberos agent that is not an Active Directory (AD) agent, the PKINIT authentication of the user fails. To work around the problem, perform one of the following actions:

- Set the RHEL 9 agent's crypto-policy to **DEFAULT:SHA1** to allow the verification of SHA-1 signatures:

  ```
  # update-crypto-polices --set DEFAULT:SHA1
  ```

- Update the non-RHEL-9 and non-AD agent to ensure it does not sign CMS data using the SHA-1 algorithm. For this, update your Kerberos client or KDC packages to the versions that use SHA-256 instead of SHA-1:
  - CentOS 9 Stream: krb5-1.19.1-15
  - RHEL 8.7: krb5-1.18.2-17
  - RHEL 7.9: krb5-1.15.1-53
  - Fedora Rawhide/36: krb5-1.19.2-7
  - Fedora 35/34: krb5-1.19.2-3

As a result, the PKINIT authentication of the user works correctly.

Note that for other operating systems, it is the krb5-1.20 release that ensures that the agent signs CMS data with SHA-256 instead of SHA-1.

See also The DEFAULT:SHA1 sub-policy has to be set on RHEL 9 clients for PKINIT to work against AD KDCs.

(BZ#2077450)

Heimdal client fails to authenticate a user using PKINIT against RHEL 9 KDC

By default, a Heimdal Kerberos client initiates the PKINIT authentication of an IdM user by using Modular Exponential (MODP) Diffie-Hellman Group 2 for Internet Key Exchange (IKE). However, the MIT Kerberos Distribution Center (KDC) on RHEL 9 only supports MODP Group 14 and 16.

Consequently, the pre-autentication request fails with the **krb5_get_init_creds: PREAUTH_FAILED** error on the Heimdal client and **Key parameters not accepted** on the RHEL MIT KDC.

To work around this problem, ensure that the Heimdal client uses MODP Group 14. Set the **pkinit_dh_min_bits** parameter in the **libdefaults** section of the client configuration file to 1759:

```
[libdefaults]
pkinit_dh_min_bits = 1759
```

As a result, the Heimdal client completes the PKINIT pre-authentication against the RHEL MIT KDC.

(BZ#2106296)

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.

(BZ#2124243)

IdM to AD cross-realm TGS requests fail

The Privilege Attribute Certificate (PAC) information in IdM Kerberos tickets is now signed with AES SHA-2 HMAC encryption, which is not supported by Active Directory (AD).

Consequently, IdM to AD cross-realm TGS requests, that is, two-way trust setups, are failing with the following error:

```
"Generic error (see e-text) while getting credentials for <service principal>"
```

(BZ#2060421)

IdM Vault encryption and decryption fails in FIPS mode

The OpenSSL RSA-PKCS1v15 padding encryption is blocked if FIPS mode is enabled. Consequently, Identity Management (IdM) Vaults fail to work correctly as IdM is currently using the PKCS1v15 padding for wrapping the session key with the transport certificate.

(BZ#2089907)

Migrated IdM users might be unable to log in due to mismatching domain SIDs

If you have used the **ipa migrate-ds** script to migrate users from one IdM deployment to another, those users might have problems using IdM services because their previously existing Security Identifiers

(SIDs) do not have the domain SID of the current IdM environment. For example, those users can retrieve a Kerberos ticket with the **kinit** utility, but they cannot log in. To work around this problem, see the following Knowledgebase article: Migrated IdM users unable to log in due to mismatching domain SIDs.

(JIRA:RHELPLAN-109613)

Directory Server terminates unexpectedly when started in referral mode

Due to a bug, global referral mode does not work in Directory Server. If you start the **ns-slapd** process with the **refer** option as the **dirsrv** user, Directory Server ignores the port settings and terminates unexpectedly. Trying to run the process as the **root** user changes SELinux labels and prevents the service from starting in future in normal mode. There are no workarounds available.

(BZ#2053204)

Configuring a referral for a suffix fails in Directory Server

If you set a back-end referral in Directory Server, setting the state of the backend using the **dsconf <instance_name> backend suffix set --state referral** command fails with the following error:

```
Error: 103 - 9 - 53 - Server is unwilling to perform - [] - need to set nsslapd-referral before moving to referral state
```

As a consequence, configuring a referral for suffixes fail. To work around the problem:

1. Set the **nsslapd-referral** parameter manually:

```
# ldapmodify -D "cn=Directory Manager" -W -H ldap://server.example.com

dn: cn=dc\3Dexample\2Cdc\3Dcom,cn=mapping tree,cn=config
changetype: modify
add: nsslapd-referral
nsslapd-referral: ldap://remote_server:389/dc=example,dc=com
```

2. Set the back-end state:

```
# dsconf <instance_name> backend suffix set --state referral
```

As a result, with the workaround, you can configure a referral for a suffix.

(BZ#2063140)

The dsconf utility has no option to create fix-up tasks for the entryUUID plug-in

The **dsconf** utility does not provide an option to create fix-up tasks for the **entryUUID** plug-in. As a result, administrators cannot not use **dsconf** to create a task to automatically add **entryUUID** attributes to existing entries. As a workaround, create a task manually:

```
# ldapadd -D "cn=Directory Manager" -W -H ldap://server.example.com -x

dn: cn=entryuuid_fixup_<time_stamp>,cn=entryuuid task,cn=tasks,cn=config
objectClass: top
objectClass: extensibleObject
```

```
basedn: <fixup base tree>
cn: entryuuid_fixup_<time_stamp>
filter: <filtered_entry>
```

After the task has been created, Directory Server fixes entries with missing or invalid **entryUUID** attributes.

(BZ#2047175)

Potential risk when using the default value for ldap_id_use_start_tls option

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.

Currently, the SSSD configuration option to enforce TLS, **ldap_id_use_start_tls**, defaults to **false**. Ensure that your setup operates in a trusted environment and decide if it is safe to use unencrypted communication for **id_provider = ldap**. Note **id_provider = ad** and **id_provider = ipa** are not affected as they use encrypted connections protected by SASL and GSSAPI.

If it is not safe to use unencrypted communication, enforce TLS by setting the **ldap_id_use_start_tls** option to **true** in the **/etc/sssd/sssd.conf** file. The default behavior is planned to be changed in a future release of RHEL.

(JIRA:RHELPLAN-155168)

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.

To work around this problem, change the MaxValRange setting in Active Directory to accommodate larger group sizes.

(JIRA:RHELDOCS-19603)

11.14. DESKTOP

Firefox add-ons are disabled after upgrading to RHEL 9

If you upgrade from RHEL 8 to RHEL 9, all add-ons that you previously enabled in Firefox are disabled.

To work around the problem, manually reinstall or update the add-ons. As a result, the add-ons are enabled as expected.

(BZ#2013247)

User Creation screen is unresponsive

When installing RHEL using a graphical user interface, the User Creation screen is unresponsive. As a consequence, creating users during installation is more difficult.

To work around this problem, use one of the following solutions to create users:

- Run the installation in VNC mode and resize the VNC window.
- Create users after completing the installation process.

(BZ#2122636)

VNC is not running after upgrading to RHEL 9

After upgrading from RHEL 8 to RHEL 9, the VNC server fails to start, even if it was previously enabled.

To work around the problem, manually enable the **vncserver** service after the system upgrade:

```
# systemctl enable --now vncserver@:port-number
```

As a result, VNC is now enabled and starts after every system boot as expected.

(BZ#2060308)

11.15. GRAPHICS INFRASTRUCTURES

Matrox G200e shows no output on a VGA display

Your display might show no graphical output if you use the following system configuration:

- The Matrox G200e GPU
- A display connected over the VGA controller

As a consequence, you cannot use or install RHEL on this configuration.

To work around the problem, use the following procedure:

1. Boot the system to the boot loader menu.
2. Add the **module_blacklist=mgag200** option to the kernel command line.

As a result, RHEL boots and shows graphical output as expected, but the maximum resolution is limited to 1024x768 at the 16-bit color depth.

(BZ#1960467)

X.org configuration utilities do not work under Wayland

X.org utilities for manipulating the screen do not work in the Wayland session. Notably, the **xrandr** utility does not work under Wayland due to its different approach to handling, resolutions, rotations, and layout.

(JIRA:RHELPLAN-121049)

NVIDIA drivers might revert to X.org

Under certain conditions, the proprietary NVIDIA drivers disable the Wayland display protocol and revert to the X.org display server:

- If the version of the NVIDIA driver is lower than 470.
- If the system is a laptop that uses hybrid graphics.
- If you have not enabled the required NVIDIA driver options.

Additionally, Wayland is enabled but the desktop session uses X.org by default if the version of the NVIDIA driver is lower than 510.

(JIRA:RHELPLAN-119001)

Night Light is not available on Wayland with NVIDIA

When the proprietary NVIDIA drivers are enabled on your system, the **Night Light** feature of GNOME is not available in Wayland sessions. The NVIDIA drivers do not currently support **Night Light**.

(JIRA:RHELPLAN-119852)

11.16. THE WEB CONSOLE

VNC console works incorrectly at certain resolutions

When using the Virtual Network Computing (VNC) console under certain display resolutions, you might experience a mouse offset issue or you might see only a part of the interface. Consequently, using the VNC console might not be possible. To work around this issue, you can try expanding the size of the VNC console or use the Desktop Viewer in the Console tab to launch the remote viewer instead.

(BZ#2030836)

11.17. VIRTUALIZATION

Installing a virtual machine over https or ssh in some cases fails

Currently, the **virt-install** utility fails when attempting to install a guest operating system (OS) from an ISO source over a https or ssh connection - for example using **virt-install --cdrom https://example/path/to/image.iso**. Instead of creating a virtual machine (VM), the described operation terminates unexpectedly with an **internal error: process exited while connecting to monitor** message.

Similarly, using the RHEL 9 web console to install a guest OS fails and displays an **Unknown driver 'https'** error if you use an https or ssh URL, or the **Download OS** function.

To work around this problem, install **qemu-kvm-block-curl** and **qemu-kvm-block-ssh** on the host to enable https and ssh protocol support, respectively. Alternatively, use a different connection protocol or a different installation source.

(BZ#2014229)

Using NVIDIA drivers in virtual machines disables Wayland

Currently, NVIDIA drivers are not compatible with the Wayland graphical session. As a consequence, RHEL guest operating systems that use NVIDIA drivers automatically disable Wayland and load an Xorg session instead. This primarily occurs in the following scenarios:

- When you pass through an NVIDIA GPU device to a RHEL virtual machine (VM)
- When you assign an NVIDIA vGPU mediated device to a RHEL VM

(JIRA:RHELPLAN-117234)

The Milan VM CPU type is sometimes not available on AMD Milan systems

On certain AMD Milan systems, the Enhanced REP MOVSB (**erms**) and Fast Short REP MOVSB (**fsrm**) feature flags are disabled in the BIOS by default. Consequently, the **Milan** CPU type might not be available on these systems. In addition, VM live migration between Milan hosts with different feature flag

settings might fail. To work around these problems, manually turn on **erms** and **fsrm** in the BIOS of your host.

(BZ#2077767)

Disabling AVX causes VMs to become unbootable

On a host machine that uses a CPU with Advanced Vector Extensions (AVX) support, attempting to boot a VM with AVX explicitly disabled currently fails, and instead triggers a kernel panic in the VM.

(BZ#2005173)

VNC is unable to connect to UEFI VMs after migration

If you enable or disable a message queue while migrating a virtual machine (VM), the Virtual Network Computing (VNC) client will fail to connect to the VM after the migration is complete.

This problem affects only UEFI based VMs that use the Open Virtual Machine Firmware (OVMF).

(JIRA:RHELPLAN-135600)

Failover virtio NICs are not assigned an IP address on Windows virtual machines

Currently, when starting a Windows virtual machine (VM) with only a failover virtio NIC, the VM fails to assign an IP address to the NIC. Consequently, the NIC is unable to set up a network connection. Currently, there is no workaround.

(BZ#1969724)

Windows VM fails to get IP address after network interface reset

Sometimes, Windows virtual machines fail to get an IP address after an automatic network interface reset. As a consequence, the VM fails to connect to the network. To work around this problem, disable and re-enable the network adapter driver in the Windows Device Manager.

(BZ#2084003)

Broadcom network adapters work incorrectly on Windows VMs after a live migration

Currently, network adapters from the Broadcom family of devices, such as Broadcom, Qlogic, or Marvell, cannot be hot-unplugged during live migration of Windows virtual machines (VMs). As a consequence, the adapters work incorrectly after the migration is complete.

This problem affects only those adapters that are attached to Windows VMs using Single-root I/O virtualization (SR-IOV).

(BZ#2090712, BZ#2091528, BZ#2111319)

A **hostdev** interface with failover settings cannot be hot-plugged after being hot-unplugged

After removing a **hostdev** network interface with failover configuration from a running virtual machine (VM), the interface currently cannot be re-attached to the same running VM.

(BZ#2052424)

Live post-copy migration of VMs with failover VFs fails

Currently, attempting to post-copy migrate a running virtual machine (VM) fails if the VM uses a device with the virtual function (VF) failover capability enabled. To work around the problem, use the standard migration type, rather than post-copy migration.

(BZ#1817965)

Host network cannot ping VMs with VFs during live migration

When live migrating a virtual machine (VM) with a configured virtual function (VF), such as a VMs that uses virtual SR-IOV software, the network of the VM is not visible to other devices and the VM cannot be reached by commands such as **ping**. After the migration is finished, however, the problem no longer occurs.

(BZ#1789206)

Using a large number of queues might cause Windows virtual machines to fail

Windows virtual machines (VMs) might fail when the virtual Trusted Platform Module (vTPM) device is enabled and the *multi-queue virtio-net* feature is configured to use more than 250 queues.

This problem is caused by a limitation in the vTPM device. The vTPM device has a hardcoded limit on the maximum number of opened file descriptors. Since multiple file descriptors are opened for every new queue, the internal vTPM limit can be exceeded, causing the VM to fail.

To work around this problem, choose one of the following two options:

- Keep the vTPM device enabled, but use less than 250 queues.
- Disable the vTPM device to use more than 250 queues.

(BZ#2020146)

PCIe ATS devices do not work on Windows VMs

When you configure a PCIe Address Translation Services (ATS) device in the XML configuration of virtual machine (VM) with a Windows guest operating system, the guest does not enable the ATS device after booting the VM. This is because Windows currently does not support ATS on **virtio** devices.

For more information, see the Red Hat KnowledgeBase .

(BZ#2073872)

Kdump fails on virtual machines with AMD SEV-SNP

Currently, kdump fails on RHEL 9 virtual machines (VMs) that use the AMD Secure Encrypted Virtualization (SEV) with the Secure Nested Paging (SNP) feature.

(JIRA:RHEL-10019)

11.18. RHEL IN CLOUD ENVIRONMENTS

Cloning or restoring RHEL 9 virtual machines that use LVM on Nutanix AHV causes non-root partitions to disappear

When running a RHEL 9 guest operating system on a virtual machine (VM) hosted on the Nutanix AHV hypervisor, restoring the VM from a snapshot or cloning the VM currently causes non-root partitions in the VM to disappear if the guest is using Logical Volume Management (LVM). As a consequence, the following problems occur:

- After restoring the VM from a snapshot, the VM cannot boot, and instead enters emergency mode.
- A VM created by cloning cannot boot, and instead enters emergency mode.

To work around these problems, do the following in emergency mode of the VM:

1. Remove the LVM system devices file: **rm /etc/lvm/devices/system.devices**
2. Recreate LVM device settings: **vgimportdevices -a**
3. Reboot the VM

This makes it possible for the cloned or restored VM to boot up correctly.

Alternatively, to prevent the issue from occurring, do the following before cloning a VM or creating a VM snapshot:

1. Uncomment the **use_devicesfile = 0** line in the **/etc/lvm/lvm.conf** file
2. Reboot the VM

(BZ#2059545)

Customizing RHEL 9 guests on ESXi sometimes causes networking problems

Currently, customizing a RHEL 9 guest operating system in the VMware ESXi hypervisor does not work correctly with NetworkManager key files. As a consequence, if the guest is using such a key file, it will have incorrect network settings, such as the IP address or the gateway.

For details and workaround instructions, see the VMware Knowledge Base.

(BZ#2037657)

Setting static IP in a RHEL virtual machine on a VMware host does not work

Currently, when using RHEL as a guest operating system of a virtual machine (VM) on a VMware host, the DatasourceOVF function does not work correctly. As a consequence, if you use the **cloud-init** utility to set the VM's network to static IP and then reboot the VM, the VM's network will be changed to DHCP.

(BZ#1750862)

11.19. SUPPORTABILITY

Timeout when running sos report on IBM Power Systems, Little Endian

When running the **sos report** command on IBM Power Systems, Little Endian with hundreds or thousands of CPUs, the processor plugin reaches its default timeout of 300 seconds when collecting huge content of the **/sys/devices/system/cpu** directory. As a workaround, increase the plugin's timeout accordingly:

- For one-time setting, run:

```
# sos report -k processor.timeout=1800
```

- For a permanent change, edit the **[plugin_options]** section of the **/etc/sos/sos.conf** file:

```
[plugin_options]
# Specify any plugin options and their values here. These options take the form
# plugin_name.option_name = value
#rpm.rpmva = off
processor.timeout = 1800
```

The example value is set to 1800. The particular timeout value highly depends on a specific system. To set the plugin's timeout appropriately, you can first estimate the time needed to collect the one plugin with no timeout by running the following command:

```
# time sos report -o processor -k processor.timeout=0 --batch --build
```

(BZ#1869561)

11.20. CONTAINERS

Running systemd within an older container image does not work

Running systemd within an older container image, for example, **centos:7**, does not work:

```
$ podman run --rm -ti centos:7 /usr/lib/systemd/systemd
 Storing signatures
 Failed to mount cgroup at /sys/fs/cgroup/systemd: Operation not permitted
 [!!!!!!] Failed to mount API filesystems, freezing.
```

To work around this problem, use the following commands:

```
# mkdir /sys/fs/cgroup/systemd
# mount none -t cgroup -o none,name=systemd /sys/fs/cgroup/systemd
# podman run --runtime /usr/bin/crun --annotation=run.oci.systemd.force_cgroup_v1=/sys/fs/cgroup -
-rm -ti centos:7 /usr/lib/systemd/systemd
```

(JIRA:RHELPLAN-96940)

APPENDIX A. LIST OF TICKETS BY COMPONENT

Bugzilla and JIRA IDs are listed in this document for reference. Bugzilla bugs that are publicly accessible include a link to the ticket.

Component	Tickets
389-ds-base	BZ#2052527, BZ#2057063, BZ#2057066, BZ#1872451, BZ#2053204, BZ#2063140, BZ#2047175
NetworkManager	BZ#2068525, BZ#2059608, BZ#2030997, BZ#2079849, BZ#2097293, BZ#2029636, BZ#1894877, BZ#2151040
anaconda	BZ#2059414, BZ#2053710, BZ#2082132, BZ#2050140, BZ#1877697, BZ#1914955, BZ#1929105, BZ#1997832, BZ#2052938, BZ#2107346, BZ#2125542, BZ#2115783
ansible-collection-microsoft-sql	BZ#2066337
ansible-collection-redhat-rhel_mgmt	BZ#2112434
ansible-freeipa	BZ#2076567
bind	BZ#1984982
catatonit	BZ#2074193
chrony	BZ#2047415, BZ#2051441
clevis	BZ#2107078
cloud-init	BZ#1750862
cockpit-appstream	BZ#2030836
cockpit	BZ#2056786
cronie	BZ#2090691
crypto-policies	BZ#2102774, BZ#2070604
cyrus-sasl	BZ#1995600
device-mapper-multipath	BZ#2084365, BZ#2033080, BZ#2011699
distribution	BZ#2063773

Component	Tickets
dnf-plugins-core	BZ#2066646
dnf	BZ#2053014, BZ#2073510
dotnet7.0	BZ#2112027
dyninst	BZ#2057675
edk2	BZ#1935497
elfutils	BZ#2088774
fapolicyd	BZ#2100041, BZ#2054740, BZ#2070655
firefox	BZ#2013247
firewalld	BZ#2040689, BZ#2039542
frr	BZ#2069563
gcc-toolset-12-annobin	BZ#2077438
gcc-toolset-12-binutils	BZ#2077445
gcc-toolset-12-gcc	BZ#2077465
gcc-toolset-12-gdb	BZ#2077494
gcc	BZ#2063255
gdb	BZ#1870017
gdm	BZ#2097308
gimp	BZ#2047161
glibc	BZ#2033683, BZ#2096191, BZ#2063142, BZ#2077838, BZ#2085529, BZ#2003291, BZ#2091549
gnome-settings-daemon	BZ#2100467
gnupg2	BZ#2070722, BZ#2073567
gnutls	BZ#2042009

Component	Tickets
golang	BZ#2075169, BZ#2111072, BZ#2092016
grub2	BZ#2074761, BZ#2026579
grubby	BZ#1978226, BZ#1969362, BZ#2127453
httpd	BZ#2079939, BZ#2065677
ipa	BZ#747959, BZ#2091988, BZ#2083218, BZ#2100227, BZ#2084180, BZ#2084166, BZ#2069202, BZ#2057471, BZ#2124243, BZ#2089907
jmc-core	BZ#1980981
kdump-anaconda-addon	BZ#1959203, BZ#2017401
kernel-rt	BZ#2061574
kernel	JIRA:RHELPLAN-117713, BZ#2027894, BZ#2066451, BZ#2079368, BZ#2065226, BZ#2013413, BZ#2069045, BZ#2001936, BZ#2097188, BZ#2096127, BZ#2054379, BZ#2073541, BZ#2030922, BZ#1945040, BZ#2100898, BZ#2068432, BZ#2046472, BZ#1613522, BZ#1874182, BZ#1995338, BZ#1570255, BZ#2023416, BZ#2021672, BZ#2000616, BZ#2013650, BZ#2132480, BZ#2060150, BZ#2059545, BZ#2069758, BZ#1960467, BZ#2005173, BZ#2129288
kexec-tools	BZ#2064708, BZ#2065013
keylime	BZ#2138167, BZ#2140670, BZ#2142009
kmod-kvdo	BZ#2064802
kmod	BZ#2103605
krb5	BZ#2068935, BZ#2106043, BZ#2060798, BZ#2077450, BZ#2106296, BZ#2060421
libdnf	BZ#2108969
libnvme	BZ#2099619
libsepol	BZ#2069718, BZ#2079276
libvirt	BZ#2064194, BZ#2014487
libvpd	BZ#2051288

Component	Tickets
libxcrypt	BZ#2034569
llvm-toolset	BZ#2061041
lsvpd	BZ#2051289
lvm2	BZ#2038183
maven	BZ#2083112
mysql	BZ#1991500
nfs-utils	BZ#2081114
nmstate	BZ#2084474, BZ#2082043
nodejs	BZ#2083072
nss	BZ#2091905
nvme-cli	BZ#2090121
nvme-stas	BZ#1893841
open-vm-tools	BZ#2061193, BZ#2037657
opencryptoki	BZ#2044179
openscap	BZ#2109485
openssh	BZ#2066882, BZ#2087121, BZ#2056884
openssl	BZ#2060510, BZ#2053289, BZ#2066412, BZ#2063947, BZ#2004915, BZ#2058663, BZ#1975836, BZ#1681178, BZ#1685470, BZ#2060044, BZ#2071631
pacemaker	BZ#2121838, BZ#2072108
pause-container	BZ#2106816
pcre2	BZ#2086494
pcs	BZ#2024522, BZ#2054671, BZ#2058251, BZ#2058252, BZ#2058246, BZ#2058243, BZ#1301204

Component	Tickets
php	BZ#2070040
pki-core	BZ#2084181
podman	BZ#2097708, BZ#2027576, BZ#2069279
policycoreutils	BZ#2115242
powerpc-utils	BZ#1920964
ppc64-diag	BZ#2051286
procps-ng	BZ#2052536, BZ#2003033
pykickstart	BZ#2083269
qemu-kvm	BZ#2044218, BZ#1965079, BZ#1951814, BZ#2060839, BZ#2014229, BZ#2052424, BZ#1817965, BZ#1789206, BZ#2090712, BZ#2020146
rear	BZ#2111059, BZ#2097437, BZ#2115958, BZ#2083272, BZ#2120736, BZ#2119501
resource-agents	BZ#1826455
rhel-system-roles	BZ#2072385, BZ#2086965, BZ#2065337, BZ#2079622, BZ#2043010, BZ#2065383, BZ#2112145, BZ#2052081, BZ#2052086, BZ#2065392, BZ#2072742, BZ#2072745, BZ#2072746, BZ#2075119, BZ#2078989, BZ#2079627, BZ#2093423, BZ#2100292, BZ#2100942, BZ#2115154, BZ#2115157, BZ#2115152, BZ#2051737, BZ#2065382, BZ#2065394, BZ#2115886, BZ#2100605, BZ#2060523, BZ#2060525, BZ#2065393, BZ#2070462, BZ#2083376, BZ#2083410, BZ#2100286, BZ#2109998, BZ#2115156, BZ#2071804, BZ#2100294, BZ#1999770
rsyslog	BZ#2064318
rust	BZ#2075337
s390utils	BZ#1870699, BZ#1932480
samba	BZ#2077487, Jira:RHELDOCS-16612
sblim-wbemcli	BZ#2083577
scap-security-guide	BZ#2070563, BZ#2120978, BZ#2038978
selinux-policy	BZ#1965013, BZ#2081425, BZ#2076681, BZ#2064274

Component	Tickets
sos	BZ#1869561
sssd	BZ#1978119, BZ#2065693, BZ#2056482
stalld	BZ#2107275
stratisd	BZ#1990905, BZ#2040352, BZ#2039960, BZ#2007018, BZ#2005110, BZ#2041558
subscription-manager	BZ#2092014, BZ#2136694
systemd	BZ#2018112
systemtap	BZ#2083727
tigervnc	BZ#2060308
tpm2-tools	BZ#2090748
tuned	BZ#2093847
ubi8-container	BZ#2120378
udisks2	BZ#1983602
unbound	BZ#2087120, BZ#2071543, BZ#2070495
valgrind	BZ#1993976
virt-who	BZ#2054504
virtio-win	BZ#1969724, BZ#2084003
whois	BZ#2054043
xmlstarlet	BZ#2069689
xorg-x11-server	BZ#1894612

Component	Tickets
other	JIRA:RHELPLAN-92522, BZ#2125549, BZ#2128016, BZ#1937031, JIRA:RHELPLAN-121982, JIRA:RHELPLAN-95456, JIRA:RHELPLAN-122321, JIRA:RHELPLAN-118462, JIRA:RHELPLAN-101140, JIRA:RHELPLAN-132023, JIRA:RHELPLAN-123369, JIRA:RHELPLAN-117109, JIRA:RHELPLAN-130379, BZ#2049492, JIRA:RHELPLAN-130376, JIRA:RHELPLAN-122735, BZ#2070793, BZ#2122716, JIRA:RHELPLAN-123368, JIRA:RHELPLAN-135601, JIRA:RHELPLAN-135602, BZ#2139877, JIRA:RHELPLAN-122776, JIRA:RHELPLAN-121180, BZ#2094015, JIRA:RHELPLAN-109067, JIRA:RHELPLAN-115603, JIRA:RHELPLAN-65217, BZ#2020529, BZ#2030412, BZ#2046653, JIRA:RHELPLAN-103993, JIRA:RHELPLAN-122345, JIRA:RHELPLAN-129327, JIRA:RHELPLAN-74672, BZ#1927780, JIRA:RHELPLAN-110763, BZ#1935544, BZ#2089200, JIRA:RHELPLAN-15509, JIRA:RHELPLAN-99136, JIRA:RHELPLAN-103232, BZ#1899167, BZ#1979521, JIRA:RHELPLAN-100087, JIRA:RHELPLAN-100639, JIRA:RHELPLAN-10304, BZ#2058153, JIRA:RHELPLAN-113995, JIRA:RHELPLAN-121048, JIRA:RHELPLAN-98983, JIRA:RHELPLAN-131882, JIRA:RHELPLAN-137660, BZ#1640697, BZ#1697896, BZ#2047713, JIRA:RHELPLAN-96940, JIRA:RHELPLAN-117234, JIRA:RHELPLAN-119001, JIRA:RHELPLAN-119852, BZ#2077767, BZ#2053598, BZ#2082303, JIRA:RHELPLAN-121049, JIRA:RHELPLAN-109613, JIRA:RHELPLAN-135600, BZ#2149172

APPENDIX B. REVISION HISTORY

0.3-8

Mon May 12 2025, Gabriela Fialová (gfialova@redhat.com)

- Updated the Customer Portal labs section

0.3-7

Tue March 18 2025, Gabriela Fialová (gfialova@redhat.com)

- Added a Known Issue in JIRA:RHEL-82566 (Installer)

0.3-6

Mon February 24 2025, Gabriela Fialová (gfialova@redhat.com)

- Added a Known Issue in JIRA:RHELDOCS-19626 (Security)

0.3-5

Thu Jan 30 2025, Gabriela Fialová (gfialova@redhat.com)

- Added an Known Issue JIRA:RHELDOCS-19603 (IdM SSSD)

0.3-4

Mon Jan 20 2025, Gabriela Fialová (gfialova@redhat.com)

- Added an Known Issue JIRA:RHEL-13837 (Installer)

0.3-3

Wed Dec 11 2024, Gabriela Fialová (gfialova@redhat.com)

- Removed a Known Issue BZ-2107346 (Installer)
- Added a Technology Preview BZ#2107346 (Installer)

0.3-2

Wed Dec 4 2024, Gabriela Fialová (gfialova@redhat.com)

- Updated the Customer Portal labs section
- Updated the Installation section

0.3-1

Tue Nov 19 2024, Gabi Fialova (gfialova@redhat.com)

- Removed a Known Issue BZ-2057471 (IdM)

0.3-0

Thu Oct 03 2024, Gabriela Fialová (gfialova@redhat.com)

- Added an Known Issue JIRA:RHEL-56135 (Installer)

0.2-9

Wed Aug 28 2024, Gabriela Fialová (gfialova@redhat.com)

- Added an Known Issue JIRA:RHELDOCS-18638 (Installer)

0.2-8

Thu Aug 22 2024, Gabriela Fialová (gfialova@redhat.com)

- Added an Known Issue JIRA:RHELDOCS-18764 (Installer)

0.2-7

Thu Jul 18 2024, Gabriela Fialová (gfialova@redhat.com)

- Updated the abstract in the Deprecated functionalities section

0.2-6

Tue Jun 11 2024, Brian Angelica (bangelic@redhat.com)

- Add Deprecated Functionality RHELDOCS-18049 (Shells and command-line tools)

0.2-5

Tue Jun 11 2024, Brian Angelica (bangelic@redhat.com)

- Added an Known Issue JIRA:RHEL-24847 (Shells and command-line tools)

0.2-4

Thu May 16 2024, Gabriela Fialová (gfialova@redhat.com)

- Added an Known Issue JIRA:RHEL-10019 (Virtualization)

0.2-3

Thu Mar 14 2024, Gabriela Fialová (gfialova@redhat.com)

- Added a Known Issue JIRA:RHEL-25967 (Kernel)

0.2-2

Thu Feb 1 2024, Gabriela Fialová (gfialova@redhat.com)

- Added a KI BZ#1834716 (Security).

0.2-1

Mon Nov 13 2023, Gabriela Fialová (gfialova@redhat.com)

- Added a Tech Preview JIRA:RHELDOCS-17040 (Virtualization)

0.2-0

Fri Nov 10 2023, Gabriela Fialová (gfialova@redhat.com)

- Updated the module on Providing Feedback on RHEL Documentation

0.1-9

Fri Nov 10 2023, Gabriela Fialová (gfialova@redhat.com)

- Added a Tech Preview JIRA:RHELDOCS-17050 (Virtualization)

- Added a Tech Preview JIRA:RHELDOCS-17050 (Virtualization)

0.1-8

Fri Oct 13 2023, Gabriela Fialová (gfialova@redhat.com)

- Added a Tech Preview JIRA:RHELDOCS-16861 (Containers)

0.1-7

September 25 2023, Gabriela Fialová (gfialova@redhat.com)

- Added a KI BZ#2122636 (Desktop)

0.1-6

September 8 2023, Marc Muehlfeld (mmuehlfeld@redhat.com)

- Added a deprecated functionality release note JIRA:RHELDOCS-16612 (Samba)
- Updated the "Providing feedback on Red Hat documentation" to reflect RHEL in JIRA.

0.1-5

August 17 2023, Gabriela Fialová (gfialova@redhat.com)

- Added an Enh BZ#2136937 (Plumbers)

0.1-4

August 07 2023, Gabriela Fialová (gfialova@redhat.com)

- Added a KI BZ#2214130 (CS)

0.1-3

August 02 2023, Marc Muehlfeld (mmuehlfeld@redhat.com)

- Updated a deprecated functionality release note BZ#1894877 (NetworkManager).

0.1-2

July 25 2023, Gabriela Fialová (gfialova@redhat.com)

- Added a Known Issue BZ#2109231 (Installer)

0.1-1

Thu Jun 15, 2023, Lucie Vařáková (lvarakova@redhat.com)

- Added a new feature BZ#2070725 (Boot loader)
- Other minor updates.

0.1-0

Wed May 17, 2023, Gabriela Fialová (gfialova@redhat.com)

- Updated the Deprecated packages section with life cycle information.

0.0-9

Thu Apr 27, 2023, Gabriela Fialová (gfialova@redhat.com)

- Added a known issue JIRA:RHELPLAN-155168 (Identity Management)

0.0-8

Tue Apr 25, 2023, Lucie Vařáková (lvarakova@redhat.com)

- Added a known issue BZ#2180665 (Kernel).

0.0-7

Mon Feb 20, 2023, Gabriela Fialová (gfialova@redhat.com)

- Added information about SAP environments to In-place upgrade from RHEL 8 to RHEL 9 .

0.0-6

Thu Feb 16, 2023, Gabriela Fialová (gfialova@redhat.com)

- Updated a known issue BZ#2132480 (Kernel).

0.0-5

Tue Feb 14, 2023, Gabriela Fialová (gfialova@redhat.com)

- Made a small formatting change in Important changes to external kernel parameters .

0.0-4

Tue Feb 14, 2023, Marc Muehlfeld (mmuehlfeld@redhat.com)

- Added an enhancement BZ#2144898 (Networking).

0.0-3

Wed Dec 07, 2022, Gabriela Fialová (gfialova@redhat.com)

- Moved the **nodejs:18** module stream BZ#2083072 from Technology Previews to fully supported features (Dynamic programming languages, web and database servers).

0.0-2

Wed Nov 16, 2022, Gabriela Fialová (gfialova@redhat.com)

- Release of the Red Hat Enterprise Linux 9.1 Release Notes.

0.0-1

Wed Sep 28, 2022, Gabriela Fialová (gfialova@redhat.com)

- Release of the Red Hat Enterprise Linux 9.1 Beta Release Notes.