Red Hat Enterprise Linux 8.7

8.7 Release Notes

Release Notes for Red Hat Enterprise Linux 8.7

Last Updated: 2024-04-29

Red Hat Enterprise Linux 8.7 8.7 Release Notes

Release Notes for Red Hat Enterprise Linux 8.7

Legal Notice

Abstract

The Release Notes provide high-level coverage of the improvements and additions that have been implemented in Red Hat Enterprise Linux 8.7 and document known problems in this release, as well as notable bug fixes, Technology Previews, deprecated functionality, and other details. For information about installing Red Hat Enterprise Linux, see Installation.

Table of Contents

MAKING OPEN SOURCE MORE INCLUSIVE

Red Hat is committed to replacing problematic language in our code, documentation, and web properties. We are beginning with these four terms: master, slave, blacklist, and whitelist. Because of the enormity of this endeavor, these changes will be implemented gradually over several upcoming releases. For more details, see our CTO Chris Wright's message.

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 8.7

Installer and image creation

Following are image builder key highlights in RHEL 8.7-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” .

Security

The DISA STIG for Red Hat Enterprise Linux 8 profile available in the **scap-security-guide** (SSG) package is now better aligned with DISA's content. This leads to fewer findings against DISA content after SSG remediations.

The Center for Internet Security (CIS) profiles available in the **scap-security-guide** (SSG) package are now aligned with CIS Red Hat Enterprise Linux 8 Benchmark version 2.0.0. This version of the benchmark adds new requirements, removed requirements that are no longer relevant, and reordered some existing requirements. The update impacts the references in the relevant rules and the accuracy of the respective profiles.

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 8.7 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 8.7:

- **opencryptoki** to version 3.18.0
- **powerpc-utils** to version 1.3.10
- **libva** to version 2.13.0

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

Infrastructure services

The following infrastructure services tools have been updated in RHEL 8.7:

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

For more information, see New Features - Infrastructure services .

Dynamic programming languages, web and database servers

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

- **Ruby 3.1**
- **Mercurial 6.2**
- **Node.js 18**

In addition, **Redis 6** has been upgraded to version 6.2.7.

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

Compilers and development tools

Updated performance tools and debuggers

The following performance tools and debuggers have been updated in RHEL 8.7:

- **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 8.7:

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

Updated compiler toolsets

The following compiler toolsets have been updated in RHEL 8.7:

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

See New features - Compilers and development tools for more information.

Java implementations in RHEL 8

The RHEL 8 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 8.7 introduces **Maven 3.8** as a new module stream.

For more information, see New features - Compilers and development tools . information.

Identity Management

Identity Management (IdM) in RHEL 8.7 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 8.7 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 AND OS CONVERSION

In-place upgrade from RHEL 7 to RHEL 8

The possible in-place upgrade paths currently are:

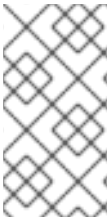
- From RHEL 7.9 to RHEL 8.4 and RHEL 8.6 on the 64-bit Intel, IBM POWER 8 (little endian), and IBM Z architectures
- From RHEL 7.6 to RHEL 8.4 on architectures that require kernel version 4.14: IBM POWER 9 (little endian) and IBM Z (Structure A). This is the final in-place upgrade path for these architectures.
- From RHEL 7.9 to RHEL 8.2 and RHEL 8.6 on systems with SAP HANA on the 64-bit Intel architecture.

To ensure your system remains supported after upgrading to RHEL 8.6, either update to the latest RHEL 8.7 version or ensure that the RHEL 8.6 Extended Update Support (EUS) repositories have been enabled.

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

For instructions on performing an in-place upgrade, see Upgrading from RHEL 7 to RHEL 8 .

For instructions on performing an in-place upgrade on systems with SAP environments, see How to in-place upgrade SAP environments from RHEL 7 to RHEL 8.



NOTE

For the successful in-place upgrade of RHEL 7.6 for IBM POWER 9 (little endian) and IBM Z (structure A) architectures, you must manually download the specific Leapp data. For more information, see the Leapp data snapshots for an in-place upgrade Knowledgebase article.

Notable enhancements include:

- The in-place upgrade of SAP Apps systems is now possible on Microsoft Azure with Red Hat Update Infrastructure (RHUI).
- The in-place upgrade is now possible on Google Cloud Platform with Red Hat Update Infrastructure (RHUI).

In-place upgrade from RHEL 6 to RHEL 8

To upgrade from RHEL 6.10 to RHEL 8, follow instructions in Upgrading from RHEL 6 to RHEL 8 .

In-place upgrade from RHEL 8 to RHEL 9

Instructions on how to perform an in-place upgrade from RHEL 8 to RHEL 9 using the Leapp utility are provided by the document Upgrading from RHEL 8 to RHEL 9 . Major differences between RHEL 8 and RHEL 9 are documented in Considerations in adopting RHEL 9 .

Conversion from a different Linux distribution to RHEL

If you are using CentOS Linux 8 or Oracle Linux 8, you can convert your operating system to RHEL 8 using the Red Hat-supported **Convert2RHEL** utility. For more information, see Converting from an RPM-based Linux distribution to RHEL.

If you are using an earlier version of CentOS Linux or Oracle Linux, namely versions 6 or 7, you can convert your operating system to RHEL and then perform an in-place upgrade to RHEL 8. Note that CentOS Linux 6 and Oracle Linux 6 conversions use the unsupported **Convert2RHEL** utility. For more information on unsupported conversions, see How to perform an unsupported conversion from a RHEL-derived Linux distribution to RHEL.

For information regarding how Red Hat supports conversions from other Linux distributions to RHEL, see the Convert2RHEL Support Policy document .

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
- Product Life Cycle Checker
- Kickstart Generator
- Kickstart Converter
- Red Hat Enterprise Linux Upgrade Helper
- Red Hat Satellite Upgrade Helper
- Red Hat Code Browser
- JVM Options Configuration Tool
- Red Hat CVE Checker
- Red Hat Product Certificates
- Load Balancer Configuration Tool
- Yum Repository Configuration Helper
- Red Hat Memory Analyzer
- Kernel Oops Analyzer
- Red Hat Product Errata Advisory Checker

1.4. ADDITIONAL RESOURCES

- **Capabilities and limits** of Red Hat Enterprise Linux 8 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 8.
- Major **differences between RHEL 7 and RHEL 8**, including removed functionality, are documented in Considerations in adopting RHEL 8 .
- Instructions on how to perform an **in-place upgrade from RHEL 7 to RHEL 8** are provided by the document Upgrading from RHEL 7 to RHEL 8 .

- The **Red Hat Insights** service, which enables you to proactively identify, examine, and resolve known technical issues, is now 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 8.7 is distributed with the kernel version 4.18.0-425, which provides support for the following architectures:

- AMD and Intel 64-bit architectures
- The 64-bit ARM architecture
- IBM Power Systems, Little Endian
- 64-bit IBM Z

Make sure you purchase the appropriate subscription for each architecture. For more information, see Get Started with Red Hat Enterprise Linux - additional architectures . For a list of available subscriptions, see Subscription Utilization on the Customer Portal.

CHAPTER 3. DISTRIBUTION OF CONTENT IN RHEL 8

3.1. INSTALLATION

Red Hat Enterprise Linux 8 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:

- Binary DVD ISO: A full installation image that contains the BaseOS and AppStream repositories and allows you to complete the installation without additional repositories.

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 Binary DVD ISO image.

See the Performing a standard RHEL 8 installation 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 Performing an advanced RHEL 8 installation 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 8 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 a list of packages distributed through BaseOS, see the Package manifest.

Content in the Application Stream repository includes additional user space applications, runtime languages, and databases in support of the varied workloads and use cases. Application Streams are available in the familiar RPM format, as an extension to the RPM format called *modules*, or as Software Collections. For a list of packages available in AppStream, see the Package manifest.

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 8 repositories, see the Package manifest.

3.3. APPLICATION STREAMS

Red Hat Enterprise Linux 8 introduces the concept of Application Streams. Multiple versions of user space components are now delivered and updated more frequently than the core operating system packages. This provides greater flexibility to customize Red Hat Enterprise Linux without impacting the underlying stability of the platform or specific deployments.

Components made available as Application Streams can be packaged as modules or RPM packages and are delivered through the AppStream repository in RHEL 8. Each Application Stream component has a given life cycle, either the same as RHEL 8 or shorter. For details, see Red Hat Enterprise Linux Life Cycle.

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.

Module streams represent versions of the Application Stream components. For example, several streams (versions) of the PostgreSQL database server are available in the **postgresql** module with the default **postgresql:10** stream. Only one module stream can be installed on the system. Different versions can be used in separate containers.

Detailed module commands are described in the Installing, managing, and removing user-space components document. For a list of modules available in AppStream, see the Package manifest.

3.4. PACKAGE MANAGEMENT WITH YUM/DNF

On Red Hat Enterprise Linux 8, installing software is ensured by the **YUM** tool, which is based on the **DNF** technology. We deliberately adhere to usage of the **yum** term for consistency with previous major versions of RHEL. However, if you type **dnf** instead of **yum**, the command works as expected because **yum** is an alias to **dnf** for compatibility.

For more details, see the following documentation:

- Installing, managing, and removing user-space components
- Considerations in adopting RHEL 8

CHAPTER 4. NEW FEATURES

This part describes new features and major enhancements introduced in Red Hat Enterprise Linux 8.7.

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#1497089)

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)

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. 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#1832510)

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 offers a simple set of shell commands that you can use in a similar way as you use UNIX commands for plain text files like **grep**, **sed**, **awk**, **diff**, **patch**, **join** and other.

(BZ#1882020)

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#2035872)

libva rebased to version 2.13.0

The **libva** library for video acceleration API has been updated to version 2.13.0. Notable improvements and new features include:

- Two new FourCC video coding formats: **X2R10G10B10** and **X2B10G10R10** for capturing, processing, and displaying video in the 10-bit RGB format (excluding Alpha).

- The VAAPI driver mapping for **iris** and **crocus** DRI drivers.
- The **vaSyncBuffer** function for output buffers synchronization.
- The **vaCopy** interface to copy surface and buffer.
- The LibVA Protected Content API for digital rights management (DRM) protected video.
- The 3DLUT Filter in Video Processing, which maps input colors to new output values.

(BZ#2099907)

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#2051330)

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 8, migrate the tokens to use 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#2043845)

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#2112435)

sysctl now matches the systemd directory order

The configuration directory order of the **sysctl** utility is now synchronized with the **systemd-sysctl** directory order. The configuration directory examines and changes kernel parameters at runtime. The configuration files in **/etc/sysctl.d** directory have higher priority than configuration files in **/run/sysctl.d**, and no more disruptions to the precedence of files between **sysctl** and **systemd** happen.

(BZ#2111915)

4.3. INFRASTRUCTURE SERVICES

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 support 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#2062356)

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 via 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#2027735)

4.4. SECURITY

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#2097837)

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 will not be applied if the PAM stack was edited by other means.

(BZ#2070564)

SSG CIS profiles aligned to the CIS RHEL 8 benchmark 2.0.0

The SCAP Security Guide (SSG) now contains changes that align the Center for Internet Security (CIS) profiles with CIS Red Hat Enterprise Linux 8 Benchmark version 2.0.0. This version of the benchmark adds new requirements, removed requirements that are no longer relevant, and reordered some existing requirements. The update impacts the references in the relevant rules and the accuracy of the respective profiles.

(BZ#2058203)

The RHEL 8 STIG profile is now better aligned with the DISA STIG content

The DISA STIG for Red Hat Enterprise Linux 8 profile (**xccdf_org.ssgproject.content_profile_stig**) available in the **scap-security-guide** (SSG) package can be used to evaluate systems according to the Security Technical Implementation Guides (STIG) by the Defense Information Systems Agency (DISA). You can remediate your systems by using the content in SSG, but you might need to evaluate them using DISA STIG automated content. With this update, the DISA STIG RHEL 8 profile is better aligned with DISA's content. This leads to fewer findings against DISA content after SSG remediation.

Note that the evaluations of the following rules still diverge:

- SV-230264r627750_rule - CCE-80790-9 (**ensure_gpgcheck_globally_activated**)
- SV-230349r833388_rule - CCE-82266-8 (**configure_bashrc_exec_tmux**)
- SV-230311r833305_rule - CCE-82215-5 (**sysctl_kernel_core_pattern**)
- SV-230546r833361_rule - CCE-80953-3 (**sysctl_kernel_yama_ptrace_scope**)
- SV-230287r743951_rule - CCE-82424-3 (**file_permissions_sshd_private_key**)
- SV-230364r627750_rule - CCE-82472-2 (**accounts_password_set_min_life_existing**)
- SV-230343r743981_rule - CCE-86107-0 (**account_passwords_pam_faillock_audit**)

(BZ#1967947)

SSG rules for mount options no longer fail incorrectly if the /tmp and /var/tmp partitions do not exist

Previously, the SCAP Security Guide (SSG) rules for mount options of **/tmp** and **/var/tmp** partitions were incorrectly reporting the **fail** result if such partitions did not exist on the system.

This enhancement makes those rules not applicable instead of failing. Now, the rules fail only when the partition exists and the system does not have correct mount options.

If these mount options are essential for a particular policy, a rule that prescribes the existence of such partitions should be present in the profile, and that one rule should fail.

(BZ#2032403)

STIG security profile updated to version V1R7

The **DISA STIG for Red Hat Enterprise Linux 8** profile in the SCAP Security Guide has been updated to align with the latest version **V1R7**.

The profile is more stable and better aligns with the RHEL 8 STIG (Security Technical Implementation Guide) manual benchmark provided by the Defense Information Systems Agency (DISA). This iteration brings updates to align the **sysctl** content to the new STIG.

You should use only the current version of this profile because older versions are no longer valid.

WARNING

Automatic remediation might render the system non-functional. Run the remediation in a test environment first.

(BZ#2112937)

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#2107081)

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#1962318)

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#2100087)

4.5. NETWORKING

The save speed of large iptables rule sets has been improved

This enhancement optimizes the **iptables-save** utility to reduce the overhead when saving large rule sets. The utility has been improved when reading entries from the **/etc/protocols** file, and it no longer searches for extension shared object files in cases where this is not necessary. As a result, the run time of **iptables-save** has been significantly improved when you save large rule sets in environments with high storage access delays.

(BZ#2058444)

NetworkManager rebased to version 1.40

The **NetworkManager** packages have been upgraded to upstream version 1.40, which provides a number of enhancements and bug fixes over the previous version:

- The device state files in the **/run/NetworkManager/devices/** directory now have new sections, **[dhcp4]** and **[dhcp6]**, which contain the DHCP options of the current lease.
- NetworkManager supports setting an IPv6 Maximum Transmission Unit (MTU) in the **ipv6.mtu** property of connections.
- NetworkManager uses the **nm.debug** kernel command line option to enable debug logging.
- Carrier detection has been improved.
- NetworkManager now restarts the DHCP client for a connection if the MAC address changes on a device.
- Wifi hotspots now use a stable random channel number unless you select a specific channel.
- NetworkManager now disables the Wi-Fi Protected Access 3 (WPA3) transition mode if you set the **wifi.key-mgmt** property to **wpa-psk** and the network interface does not support Protected Management Frames (PMF). The transition mode caused problems in certain setups in this scenario. To explicitly enable the WPA3 transitioning mode, set **wifi.key-mgmt** to **sae**.
- NetworkManager now shortens an excessively long hostname received from a DHCP server to the first dot or to 64 characters.

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

(BZ#2063109)

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)

NetworkManger now stores DHCP lease information in the /run/NetworkManager/devices/ directory

NetworkManager now stores lease information from the DHCP server in the **/run/NetworkManager/devices/** directory. Previously, the file-based API was not available and this information was only visible in the output of the **nmcli -f all devices show DEVICE** command. With this enhancement, other utilities and scripts can access DHCP options without calling **nmcli**.

(BZ#1943153)

4.6. KERNEL

Kernel version in RHEL 8.7

Red Hat Enterprise Linux 8.7 is distributed with the kernel version 4.18.0-425.

(BZ#2125545)

The default mitigation of SSBD and STIBP has been changed

The default mitigation of the **spec_store_bypass_disable** (**SSBD**) and **spectre_v2_user** (**STIBP**) boot parameters has been changed from the **seccomp** mode to **prctl**. With this update, performance of containers and applications under the control of **seccomp** improves.

(BZ#2101938)

The vmcore dump file generates correctly on the debug kernel variant

With this update, the **kdump** mechanism now uses the same version of the non-debug kernel as the capture kernel when the current kernel is debug variant. By using a non-debug kernel as the capture kernel, **kdump** consumes less memory than the debug variant. As a result, **kdump** generates the **vmcore** file correctly and captures the memory contents of the crashed kernel.

(BZ#2006000)

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)

4.7. BOOT LOADER

GRUB is signed by new keys

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

(BZ#2074762)

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#2070347)

4.8. FILE SYSTEMS AND STORAGE

nfsrahead has been added to RHEL 8

With the introduction of the **nfsrahead** tool, you can use it to modify the **readahead** value for NFS mounts, and thus affect the NFS read performance.

(BZ#1946283)

rpcctl command now displays SunRPC connection information

With this update, you can use the **rpcctl** command to display the information collected in the SunRPC **sysfs** files about the system's SunRPC objects. You can show, remove, and set objects in the SunRPC network layer through the **sysfs** file system.

(BZ#2087187)

multipath.conf can now include protocol-specific configuration overrides in DM Multipath

You can access paths of multipath devices through various protocols. Because various protocols can have various optimal configurations, it was previously not possible to set the optimal configuration for all protocols in the Device Mapper Multipath feature without a per-protocol option. With this enhancement, you can include protocol-specific configuration overrides in the **multipath.conf** file. As a result, you can now configure multipath device paths on a per-protocol basis, allowing for the correct configuration of multipath devices accessible through multiple protocols.

(BZ#2065477)

multipathd now supports detecting FPIN-Li events

When you add a new value **fpin** for the **marginal_pathgroups** config option, you enable **multipathd** to monitor the Link Integrity Fabric Performance Impact Notification (PFIN-Li) events and move paths

with link integrity issues to a marginal pathgroup. With the **fpin** value set, **multipathd** overrides its existing marginal path detection methods and relies on the Fibre Channel fabric to identify link integrity issues.

With this enhancement, the **multipathd** method becomes more robust in detecting marginal paths on Fibre Channel fabrics that can issue PFIN-Li events.

(BZ#2083077)

4.9. HIGH AVAILABILITY AND CLUSTERS

pcs command-line 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#2023845)

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#1950551)

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. With this update, the **multiple-active** resource parameter accepts a value of **stop_unexpected**, which allows you to specify 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.

(BZ#2036815)

New allow-unhealthy-node Pacemaker resource meta-attribute

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.

(BZ#2059638)

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#1182956)

Pacemaker now supports specifying Access Control Lists (ACLs) for system groups

Pacemaker previously allowed ACLs to be specified for individual users, but it is sometimes simpler and would comform better with local policies to specify ACLs for a system group, and to have them apply to all users in that group. The **pcs acl group** command was present in earlier releases but had no effect. Now, users can now specify ACLs for a system group using this command.

(BZ#1724310)

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#1909904)

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#1874624)

4.10. 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-2022:8833 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 8.7 provides numerous new features together with bug and security fixes over **Node.js 16** available since RHEL 8.5.

Notable changes include:

- The **V8** engine has been upgraded to version 10.2.
- The **npm** package manager has been upgraded to version 8.18.0.
- **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:

```
# yum module install nodejs:18
```

If you want to upgrade from the **nodejs:16** stream, see Switching to a later stream .

(BZ#2083073)

nodejs:18 rebased to version 18.14 with npm rebased to version 9

Node.js 18.14, released in RHSA-2023:1583, includes a SemVer major upgrade of **npm** from version 8 to version 9. This update was necessary due to maintenance reasons and may require you to adjust your **npm** configuration.

Notably, auth-related settings that are not scoped to a specific registry are no longer supported. This change was made for security reasons. If you used unscoped authentication configurations, the supplied token was sent to every registry listed in the **.npmrc** file.

If you use unscoped authentication tokens, generate and supply registry-scoped tokens in your **.npmrc** file.

If you have configuration lines using **_auth**, such as **//registry.npmjs.org/:_auth** in your **.npmrc** files, replace them with **//registry.npmjs.org/:_authToken=${NPM_TOKEN}** and supply the scoped token that you generated.

For a complete list of changes, see the upstream changelog.

(BZ#2178087)

A new module stream: ruby:3.1

RHEL 8.7 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 8.5.

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:

```
# yum module install ruby:3.1
```

If you want to upgrade from an earlier **ruby** module stream, see Switching to a later stream .

(BZ#2063772)

A new module stream: mercurial:6.2

RHEL 8.7 adds **Mercurial 6.2** as a new module stream. This version provides a number of bug fixes, enhancements, and performance improvements over **Mercurial 4.8** available since RHEL 8.0.

Notable changes include:

- **Mercurial 6.2** supports **Python 3.6** or later
- **Mercurial** no longer supports **Python 2**
- The **hg purge** and **hg clean** commands now provide a new **-i** option, which enables you to delete ignored files instead of untracked files
- The **hg diff** and **hg extdiff** commands now support the **--from <revision>** and **--to <revision>** arguments
- A new internal merge utility, **internal:mergediff**, is now available
- The Zstandard (ZSTD) compression is now used by default for new repositories when available
- A new way of specifying required extensions is now available that prevents **Mercurial** from starting if the required extensions are not found

In addition, a new **mercurial-chg** utility is available, which provides a C wrapper for the **hg** command. When you use the **chg** command, a **Mercurial** command server background process is created, a C program connects to that background process and executes **Mercurial** commands. As a result, the performance is significantly increased.

To install the **mercurial:6.2** module stream, use:

```
# yum module install mercurial:6.2
```

If you want to upgrade from the **mercurial:4.8** stream, see Switching to a later stream .

(BZ#2089849)

mariadb-java-client rebased to version 2.7.1

The **mariadb-java-client** package, which provides a **MariaDB** connector for applications developed in Java, has been updated to version 2.7.1.

This update introduces the following changes in services:

- Client authentication plug-ins are now defined as services. As a result, you can easily add new client authentication plug-ins. The driver includes the **caching_sha2_password** and **sha256_password** plug-ins for compatibility with **MySQL**.
- Credential plug-ins are now permitted to provide credential information. The driver includes three default plug-ins: **AWS IAM**, **Environment**, and **Property**.
- The SSL factory service now enables you to use custom SSL implementation. For example, you can create a new **HostnameVerifier** implementation.

Other notable changes include:

- The **enabledSslProtocolSuites** option now includes TLSv1.2 by default.

(BZ#2043212)

redis rebased to version 6.2.7

Redis 6, which is an advanced key-value store provided the **redis:6** module stream, has been updated to version 6.2.7. This update provides bug fixes, security fixes, and improvements over version 6.0 available since RHEL 8.4.

(BZ#1999873)

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)

4.11. COMPILERS AND DEVELOPMENT TOOLS

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:

```
# yum 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 Using GCC Toolset.

(BZ#2077276)

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#2077447)

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#2077448)

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 Clang versions 9.0 and later with the **__builtin_dynamic_object_size** builtin.

(BZ#2033684)

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#2096189)

Added support for IBM Z Series z16 in glibc

The support is now available for the **s390** instruction set with the **IBM z16** platform in **glibc**. **IBM z16** provides two additional hardware capabilities 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#2077835)

New make-latest package

This enhancement introduces the **make-latest** package which includes the latest version of the **make** utility. Previously, we provided the latest **make** version through GCC Toolset. Now, you can separately install the **make-latest** package and run the latest version with **scl enable make43 /bin/bash** (in case the **make43** version is the latest).

(BZ#2083419)

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 Aarch64 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#2077492)

libpfm now supports AMD Zen 2 and Zen 3 processors

With this enhancement, users now can access the AMD Zen 2 and Zen 3 performance monitoring hardware using **libpfm**.

(BZ#2067218)

papi now supports AMD Zen 2 and Zen 3 processors

With this enhancement, users now can access the AMD Zen 2 and Zen 3 performance monitoring hardware using **papi**.

(BZ#2071558)

Improved hardware identification for ARM processors

With this enhancement, the **papi_avail** utility now correctly reports the vendor string and code information for various ARM vendors. This utility allows the **PAPI_get_hardware_info()** function to identify processors manufactured by companies other than **ARM** limited to the **aarch64** architecture. As a result, developers can tune the code for the required architecture.

(BZ#2037427)

Updated Fujitsu A64FX event mappings

The **PAPI** library has been updated for Fujitsu A64FX processors. Users can now use additional presets in the output of **papi_avail** that can be used to analyze program performance.

These include the **IDL** event presets:

PAPI_BRU_IDL
: Branch unit idle

PAPI_FXU_IDL
: Integer unit idle

PAPI_FPU_IDL
: Floating point unit idle

PAPI_LSU_IDL
: Load store unit idle

(BZ#2037417)

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#2057676)

The systemtap package rebased to version 4.7

The **systemtap** package has been rebased to version 4.7. Notable bug fixes and enhancements include:

- A new **--sign-module** option to manually sign modules with a MOK key, for use on SecureBoot systems.
- A new **stap-profile-annotate** tool to produce system-wide profiles of annotated source code.
- A new general Python tapset for probing function entry and return.
- Extended ***foo*** processing for kernel-space probes for strings that may be in user-space.
- Extended the regular-expression language for non-capturing groups.
- Added tapset support for several recently added kernel system calls.

(BZ#2057565)

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#2075344)

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#2061042)

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#2075162)

The LLVM gold plugin is now available on the IBM Z architecture

With this enhancement, users can create LTO builds with **clang** and **ld.bfd** on the IBM Z (**s390x**) architecture. The **s390x** architecture now supports linking with **ld.bfd** and LTO.

(BZ#2088315)

A new module stream: maven:3.8

RHEL 8.7 introduces **Maven 3.8** as a new module stream.

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

```
# yum module install maven:3.8
```

If you want to upgrade from the **maven:3.6** stream, see Switching to a later stream .

(BZ#2083114, BZ#2064785, BZ#2088473)

.NET version 7.0 is available

Red Hat Enterprise Linux 8.7 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#2112096)

4.12. 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)

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)

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#782917)

IdM now indicates whether a given name is a user or a group in a trusted AD domain during a name search

With this update, new **getorigbyusername()** and **getorigbygroupname()** calls are added to **libsss_nss_idmap**, a utility library for SID-based lookups. This addition makes user and group lookup more robust when Identity Management (IdM) is in a trust with an Active Directory (AD) domain. When performing a user or group lookup, IdM can now display whether the given name belongs to a user or a group in the trusted domain.

(BZ#2062379)

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#2076554)

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#2077468)

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)

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#2052528)

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)

Replication changelog trimming is now enabled by default in Directory Server

Previously, Directory Server was not configured to automatically trim the replication **changelog** file by default. Consequently, the **changelog** file could become very large. With this update, Directory Server is configured by default to trim changelog entries that are older than seven days, preventing excessive growth of the **changelog** file.

(BZ#2062679)

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-symkey**
- **idm-pki-tools**
- **idm-pki-symkey-debuginfo**
- **idm-pki-tools-debuginfo**
- **idm-pki-acme**
- **idm-pki-base**
- **idm-pki-base-java**
- **idm-pki-ca**
- **idm-pki-kra**
- **idm-pki-server**
- **python3-idm-pki**

pki-core stays unchanged (this also includes **pki-core-debuginfo** and **pki-core-debugsource**).

(BZ#2139821)

4.13. GRAPHICS INFRASTRUCTURES

Vulkan packages are available on 64-bit IBM POWER

Packages that provide support for the Vulkan 3D graphics API are now available on the little-endian 64-bit IBM POWER architecture (**ppc64le**):

- **vulkan-headers**
- **vulkan-loader**
- **vulkan-loader-devel**
- **vulkan-tools**

With these packages, you can run software that uses a Vulkan rendering engine.

Previously, these packages were only available on the AMD64 and Intel 64 architecture.

(BZ#2012639)

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)

The force_probe option is no longer required with 12th Gen Intel Core GPUs

Prior to this release, you had to set the **i915.alpha_support=1** or **i915.force_probe=*** kernel option to enable support for the 12th Gen Intel Core GPUs, formerly known as Alder Lake-S and Alder Lake-P.

With this release, you no longer have to set the option, and full support for these GPUs is enabled by default.

(JIRA:RHELPLAN-136150)

4.14. THE WEB CONSOLE

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)

A new button in RHEL web console for installing kernel patches separately

With this update, the RHEL web console provides the **Install kpatch updates** button. You can use it to install only kernel patches without the necessity to install other updates and reboot your system.

(JIRA:RHELPLAN-121981)

The diagnostics reports page now offers new functionalities

In the updated web console diagnostics report (**sos** report) page you now can:

- label the report
- encrypt the report with a passphrase
- conceal private data within the report

Additionally, you can see a list of previously generated reports and download or delete them.

(JIRA:RHELPLAN-121983)

Crypto policies setup from the web console UI

With this update, you can change different cryptographic policy levels directly from the RHEL web console user interface (UI). You can access your cryptographic policy configuration options from the **Configuration** field in the **Overview** page of your UI.

Note that you must have the administrative access active to be able to change the settings.

(JIRA:RHELPLAN-121980)

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)

4.15. RED HAT ENTERPRISE LINUX SYSTEM ROLES

The 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#2065339, BZ#2066868)

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#2086869)

The network RHEL System 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 System Role supports routing rules so that the users have control over the packet transmission or route selection.

(BZ#1996731)

The Networking System Role now uses the Ansible managed comment in its managed configuration files

When using the **initscripts** provider, the Networking System Role now generates commented **ifcfg** files in the **/etc/sysconfig/network-scripts** directory. The Networking role inserts the **Ansible managed** comment using the Ansible standard **ansible_managed** variable. The comment declares that an **ifcfg** file is managed by Ansible, and indicates that the **ifcfg** file should not be edited directly as the Networking role will overwrite the file. The **Ansible managed** comment is added when the provider is **initscripts**. When using the Networking role with the **nm** (NetworkManager) provider, the **ifcfg** file is managed by NetworkManager and not by the Networking role.

(BZ#2065670)

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#2043009)

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#2066338, BZ#2120713, BZ#2039990, BZ#2120714)

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#2112143)

Support for thinly provisioned volumes is available in the storage RHEL System Role

The **storage** RHEL System Role can now create and manage thinly provisioned LVM logical volumes. 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#2066876)

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#2075116)

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#2079008)

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#2086934)

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#2086935)

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#2093437)

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#2079114)

The storage System 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 the **storage** role output to only produce debugging output if they are using Ansible verbosity level 1 or above.

(BZ#2056480)

The metrics System Role now generates files with the proper ansible_managed comment in the header

Previously, the **metrics** role did not add an **ansible_managed** header comment to files generated by the role. With this fix, the **metrics** role adds the **ansible_managed** header comment to files it generates, and as a result, users can easily identify files generated by the **metrics** role.

(BZ#2065215)

The postfix System Role now generates files with the proper ansible_managed comment in the header

Previously, the **postfix** role did not add an **ansible_managed** header comment to files generated by the role. With this fix, the **postfix** role adds the **ansible_managed** header comment to files it generates, and as a result, users can easily identify files generated by the **postfix** role.

(BZ#2065216)

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#2065218)

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#2100297)

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#2100939)

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#2100979)

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#2115159)

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#2115160)

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#2115162)

4.16. VIRTUALIZATION

ap-check is now available in RHEL 8

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#1660911)

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 8 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#2043830)

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#2067126)

ESXi hypervisor and SEV-ES is now fully supported

You can now enable the AMD Secure Encrypted Virtualization-Encrypted State (SEV-ES) to secure RHEL virtual machines (VMs) on VMware's ESXi hypervisor, versions 7.0.2 and later. This feature was previously introduced in RHEL 8.4 as a Technology Preview. It is now fully supported.

(BZ#1904496)

Secure Execution 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 virtual machine (VM) that has Secure Execution enabled.

Additionally, the Guest Interruption State Area (GISA) mechanism has now been enabled for Secure Execution VMs, which allows interrupts to be delivered directly into the VM by completely bypassing the host operating system.

(JIRA:RHELPLAN-98420, BZ#1984905, BZ#2043870)

4.17. RHEL IN CLOUD ENVIRONMENTS

RHEL virtual machines are now supported on the Ampere Altra architecture

With this update, running a RHEL operating system is now supported on Azure Virtual Machines with processors based on the Ampere® Altra® architecture.

(JIRA:RHELPLAN-121252)

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)

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)

4.18. 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-118463)

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-100037)

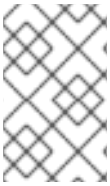
Podman now supports the --health-on-failure option

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 new network stack available starting with Podman 4.0 consists of two tools, the Netavark network setup tool and the Aardvark DNS server. In RHEL 8, the Netavark stack, previously available as a Technology Preview, is now fully supported.

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 authoritative DNS server.

(JIRA:RHELPLAN-100039)

CHAPTER 5. IMPORTANT CHANGES TO EXTERNAL KERNEL PARAMETERS

This chapter provides system administrators with a summary of significant changes in the kernel shipped with Red Hat Enterprise Linux 8.7. 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

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**.

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**.

rcutree.rcu_delay_page_cache_fill_msec = [KNL]

With this parameter you can set the page-cache refill delay in milliseconds in response to low-memory conditions. The range of permitted values is **0:100000**.

rcuscale.kfree_rcu_test_double = [KNL]

With this parameter you can test the double-argument variant of the **kfree_rcu()** function. If this parameter has the same value as **rcuscale.kfree_rcu_test_single**, both the single- and double-argument variants are tested.

rcuscale.kfree_rcu_test_single = [KNL]

With this parameter you can test the single-argument variant of the **kfree_rcu()** function. If this parameter has the same value as **rcuscale.kfree_rcu_test_double**, both the single- and double-argument variants are tested.

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 migitation
- **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**.

s390_iommu_aperture = [KNL,S390]

With this parameter you can specify the size of the per device DMA address space accessible through the DMA and IOMMU APIs as a decimal factor of the size of main memory.

- The default value is set to **1** which means that you can concurrently use as many DMA addresses as physical memory is installed, if supported by hardware, and thus map all of memory at once.
- With a value of **2** you can map all of memory twice.
- The value of **0** imposes no restrictions other than those given by hardware at the cost of significant additional memory use for tables.

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, the physical memory region can exceed 4 GB.
- If the system has less than 4 GB RAM installed, the 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.

kvm.nx_huge_pages_recovery_ratio = [KVM]

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

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

module.sig_enforce = norid [S390]

With this parameter you can ignore the RID field and force the use of one PCI domain per PCI function.

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**.

spectre_v2_user = [X86]

With this parameter you can control mitigation of Spectre variant 2 (indirect branch speculation) vulnerability between user space tasks.

- **auto**: kernel selects the mitigation depending on the available CPU features and vulnerability.
- The default mitigation is set to **prctl**.
- Not specifying this option is equivalent to **spectre_v2_user=auto**.

spec_store_bypass_disable = [X86]

With this parameter you can control whether the Speculative Store Bypass (SSB) optimization to mitigate the SSB vulnerability is used.

- Not specifying this option is equivalent to **spec_store_bypass_disable=auto**.
- The default mitigation is set to **prctl**.

New sysctl parameters

perf_user_access = [ARM64]

With this parameter you can control user space access for reading performance 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**.

force_cgroup_v2_swappiness

With this parameter you can deprecate the per-cgroup swappiness value available only in **cgroupsV1**. Due to a **systemd** design choice, most of all system and user processes are run within a **cgroup**. Furthermore these **cgroup** swappiness values default to **60**. This can lead to undesireable effects where systems swappiness value has little effect on the swap behavior of the system.
If you do want to use the per-**cgroup** swappiness feature, you can configure the system with **force_cgroup_v2_swappiness=1** to have more consistent swappiness behavior across the whole system.

Note that this is a RHEL specific feature.

CHAPTER 6. DEVICE DRIVERS

6.1. NEW DRIVERS

Network drivers

- Maxlinear Ethernet GPY Driver (**mxl-gpy**)
- Realtek 802.11ax wireless 8852A driver (**rtw89_8852a**)
- Realtek 802.11ax wireless 8852AE driver (**rtw89_8852ae**)

Graphics drivers and miscellaneous drivers

- 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**)
- 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**)
- Intel® PMT Class driver (**pmt_class**)
- Intel® PMT Crashlog driver (**pmt_crashlog**)
- Intel® PMT Telemetry driver (**pmt_telemetry**)
- Intel® speed select interface driver (**isst_if_common**)
- 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**)
- Serial multi instantiate pseudo device driver (**serial-multi-instantiate**)
- AMD® SPI Master Controller Driver (**spi-amd**)

6.2. UPDATED DRIVERS

Network drivers

- VMware vmxnet3 virtual NIC driver (**vmxnet3**) has been updated to version 1.7.0.0-k.
- Intel® PRO/1000 Network Driver (**e1000e**) has been updated to version 4.18.0-425.3.1.
- Intel® Ethernet Switch Host Interface Driver (**fm10k**) has been updated to version 4.18.0-425.3.1.
- Intel® Ethernet Connection XL710 Network Driver (**i40e**) has been updated to version 4.18.0-425.3.1.
- Intel® Ethernet Adaptive Virtual Function Network Driver (**iavf**) has been updated to version 4.18.0-425.3.1.
- Intel® Gigabit Ethernet Network Driver (**igb**) has been updated to version 4.18.0-425.3.1.
- Intel® Gigabit Virtual Function Network Driver (**igbvf**) has been updated to version 4.18.0-425.3.1.
- Intel® 2.5G Ethernet Linux Driver (**igc**) has been updated to version 4.18.0-425.3.1.
- Intel® 10 Gigabit PCI Express Network Driver (**ixgbe**) has been updated to version 4.18.0-425.3.1.
- Intel® 10 Gigabit Virtual Function Network Driver (**ixgbevf**) has been updated to version 4.18.0-425.3.1.
- Mellanox 5th generation network adapters (ConnectX series) core driver (**mlx5_core**) has been updated to version 4.18.0-425.3.1.

Storage drivers

- Emulex LightPulse Fibre Channel SCSI driver (**lpfc**) has been updated to version 14.0.0.15.
- 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 42.100.00.00.
- QLogic Fibre Channel HBA Driver (**qla2xxx**) has been updated to version 10.02.07.400-k.
- 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 8. 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	1 (bpf() syscall restricted to privileged users, without recovery)
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	n
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	y
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_ktime_get_coarse_ns, bpf_for_each_map_elem, bpf_snprintf
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_override_return, bpf_get_stack, bpf_get_current_cgroup_id, bpf_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, 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_get_current_task_btf, bpf_ktime_get_coarse_ns, bpf_for_each_map_elem, bpf_snprintf

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_ktime_get_coarse_ns, bpf_check_mtu, bpf_for_each_map_elem, bpf_snprintf
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_ktime_get_coarse_ns, bpf_check_mtu, bpf_for_each_map_elem, bpf_snprintf

Program type	Available helpers
tracepoint	bpf_map_lookup_elem, bpf_map_update_elem, bpf_map_delete_elem, bpf_probe_read, bpf_ktime_get_ns, bpf_get_prandom_u32, bpf_get_smp_processor_id, bpf_tail_call, bpf_get_current_pid_tgid, bpf_get_current_uid_gid, bpf_get_current_comm, bpf_perf_event_read, bpf_perf_event_output, bpf_get_stackid, bpf_get_current_task, bpf_current_task_under_cgroup, bpf_get_numa_node_id, bpf_probe_read_str, bpf_perf_event_read_value, bpf_get_stack, bpf_get_current_cgroup_id, bpf_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_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_get_current_task_btf, bpf_ktime_get_coarse_ns, bpf_for_each_map_elem, bpf_snprintf
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_ktime_get_coarse_ns, bpf_check_mtu, bpf_for_each_map_elem, bpf_snprintf
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_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_get_current_task_btf, bpf_ktime_get_coarse_ns, bpf_for_each_map_elem, bpf_snprintf

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_ktime_get_coarse_ns, bpf_for_each_map_elem, bpf_snprintf
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_ktime_get_coarse_ns, bpf_for_each_map_elem, bpf_snprintf
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_ktime_get_coarse_ns, bpf_for_each_map_elem, bpf_snprintf

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_ktime_get_coarse_ns, bpf_for_each_map_elem, bpf_snprintf
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_ktime_get_coarse_ns, bpf_for_each_map_elem, bpf_snprintf
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_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_ktime_get_coarse_ns, bpf_for_each_map_elem, bpf_snprintf

Program type	Available helpers
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_ktime_get_coarse_ns, bpf_for_each_map_elem, bpf_snprintf
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_ktime_get_coarse_ns, bpf_for_each_map_elem, bpf_snprintf
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_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_ktime_get_coarse_ns, bpf_for_each_map_elem, bpf_snprintf

Program type	Available helpers
raw_tracepoint	bpf_map_lookup_elem, bpf_map_update_elem, bpf_map_delete_elem, bpf_probe_read, bpf_ktime_get_ns, bpf_get_prandom_u32, bpf_get_smp_processor_id, bpf_tail_call, bpf_get_current_pid_tgid, bpf_get_current_uid_gid, bpf_get_current_comm, bpf_perf_event_read, bpf_perf_event_output, bpf_get_stackid, bpf_get_current_task, bpf_current_task_under_cgroup, bpf_get_numa_node_id, bpf_probe_read_str, bpf_perf_event_read_value, bpf_get_stack, bpf_get_current_cgroup_id, bpf_map_push_elem, bpf_map_pop_elem, bpf_map_peek_elem, bpf_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_get_current_task_btf, bpf_ktime_get_coarse_ns, bpf_for_each_map_elem, bpf_snprintf
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_ktime_get_coarse_ns, bpf_for_each_map_elem, bpf_snprintf
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_ktime_get_coarse_ns, bpf_for_each_map_elem, bpf_snprintf
lirc_mode2	not supported

Program type	Available helpers
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_ktime_get_coarse_ns, bpf_for_each_map_elem, bpf_snprintf
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_ktime_get_coarse_ns, bpf_for_each_map_elem, bpf_snprintf
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_ktime_get_coarse_ns, bpf_for_each_map_elem, bpf_snprintf
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_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_get_current_task_btf, bpf_ktime_get_coarse_ns, bpf_for_each_map_elem, bpf_snprintf

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_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_ktime_get_coarse_ns, bpf_for_each_map_elem, bpf_snprintf
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
ext	not supported
lsm	not supported

Program type	Available helpers
sk_lookup	bpf_map_lookup_elem, bpf_map_update_elem, bpf_map_delete_elem, bpf_ktime_get_ns, bpf_get_prandom_u32, bpf_get_smp_processor_id, bpf_tail_call, bpf_perf_event_output, bpf_get_current_task, bpf_get_numa_node_id, bpf_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_ktime_get_coarse_ns, bpf_for_each_map_elem, bpf_snprintf

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
hash_of_maps	yes
devmap	yes
sockmap	yes

Map type	Available
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	no

CHAPTER 8. BUG FIXES

This part describes bugs fixed in Red Hat Enterprise Linux 8.7 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#1899494)

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#1970726)

Installer asks for the passphrase missing in the Kickstart file for the encrypted devices during the installation

Previously, when running the installer in graphical mode, if the passphrase was not specified in the Kickstart file, the installer would not ask for entering the passphrase for encrypted devices. As a consequence, the partitioning specified in the Kickstart file was not applied during the installation.

This update adds a dialog window that appears during the installation and asks for the missing passphrase. As a result, the installer properly applies the partitioning scheme specified in the Kickstart file.

(BZ#2029101)

Images now build successfully for packages in blueprint that contain conditional dependencies

Previously, when using the web console to customize a blueprint with packages that contained conditional dependencies, such as **ipa-client**, **cockpit**, **podman**, would cause the build to fail because of the missing dependencies. As a consequence, the conditional dependency was not met during the dep-solve packages. This issue is fixed now, and the builds will no longer fail when dep-solving conditional dependencies.

(BZ#2065734)

8.2. 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#2060815)

8.3. SHELLS AND COMMAND-LINE TOOLS

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#2075807)

The cvSaveImage function in the opencv library no longer terminates the user application

Previously, the **opencv** library could not use the **cvSaveImage** function correctly. Consequently, the user application was terminated unexpectedly. With this update, the **cvSaveImage** function writes the image data on disk and no longer terminates the user application.

(BZ#2104776)

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#2072978)

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

In RHEL 8.5, 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#2115918)

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#2077404)

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#2021935)

Fallbacks of SR-IOV devices now complete successfully

Previously, Single Root I/O Virtualization (SR-IOV) devices did not fallback after device failover because the **hcnmgr** script used an incorrect **active_slave** attribute instead of a **primary** attribute. With this update, the **hcnmgr** script uses the correct attribute and fallbacks for SR-IOV devices complete successfully.

(BZ#2078514)

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 and bug fixes 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#2051313)

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#2051316)

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 and bug fixes include:

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

(BZ#2051319)

8.4. INFRASTRUCTURE SERVICES

The printer test page layout in RHEL 8 has changed

Previously, the print test page was not printed if the destination document format was PDF. This update introduces a new test page layout to work with a broader set of printers. Note that the test page does not contain any information regarding the printer or the test page print job.

(BZ#2064606)

The frr binary files and scripts have a new location

Previously, the **frr** package for managing dynamic routing stack contained its binary files and scripts in the **/usr/lib/frr** directory, which caused certain issues when applying the new targeted SELinux policy. Consequently, SELinux logged denial messages in access vector cache (AVC) and prevented **frr** from starting properly.

With this update, **/usr/libexec/frr** is the new location of the **frr** binary files and scripts. As a result, SELinux applies rules for binaries and scripts in **/usr/libexec/frr** and for other **frr** libraries in **/usr/lib64/frr** separately, and no longer produces denial messages.

(BZ#1714984, BZ#1941765)

8.5. SECURITY

OpenSCAP remediation sets correct permissions for /etc/tmux.conf

Previously, when remediating the SCAP rule **configure_tmux_lock_after_time**, the **/etc/tmux.conf** file was created with permissions respecting umask (600). This caused **/etc/tmux.conf** to be unreadable by regular users. If a regular user logged in, they received an error message and had to wait for several minutes before a timeout ran out and they were logged in. With this update, the remediation of rule **configure_tmux_lock_after_time** sets specific permissions of **/etc/tmux.conf** to 644. As a result, regular users no longer encounter the error message or login delay.

(BZ#2064696)

SCAP rule for Rsyslog correctly identifies .conf files

Previously, rule "Ensure System Log Files Have Correct Permissions" (**xccdf_org.ssgproject.content_rule_rsyslog_files_permissions**) did not expand glob expressions in Rsyslog include statements. As a consequence, the rule did not parse all relevant configuration files, and some log files did not have their permissions checked. With this update, the rule correctly expands the glob expressions to identify the **.conf** files it needs to parse. As a result, the rule now correctly processes the required **.conf** files to ensure that all configured log files have the correct permissions.

(BZ#2075384)

Rules for chronyd do not require explicit chrony user configuration

RHEL runs **chronyd** under the **chrony** user by default. Previously, the check and remediation for the **chronyd** service configuration user were stricter than necessary. The overly strict check led to false positives and to excessive remediations. In this version, the check and remediations of the rule **xccdf_org.ssgproject.content_rule_chronyd_run_as_chrony_user** are updated, for both the minimalistic correct configuration and legacy explicit correct configurations pass. As a result, the rule respects the default RHEL behavior and does not require explicit **chrony** user configuration.

(BZ#2077531)

Warning added to rsyslog_remote_loghost

The SCAP rule **xccdf_org.ssgproject.content_rule_rsyslog_remote_loghost** ensures that the Rsyslog daemon is configured to send log messages to a remote log host. However, the rule does not configure TCP queues. As a consequence, the system hangs if TCP queues are not configured, and the

remote log host becomes unavailable. This update adds a warning message that explains how to configure TCP queues. If you encounter system hangs while using this rule, read the warning and configure the system properly.

(BZ#2078974)

Remediation of sudo_custom_logfile works for custom sudo log files

Previously, remediation of the SCAP Security Guide rule **xccdf_org.ssgproject.content_sudo_custom_logfile** did not work for custom **sudo** log files with a different path than **/var/log/sudo.log**. With this update, the rule is fixed so that it can properly remediate if the system has a custom **sudo** log file that does not match the expected path.

(BZ#2083109)

Remediation of firewalld_sshd_port_enabled now works correctly

Previously, Bash remediation of the SCAP rule **xccdf_org.ssgproject.content_rule_firewalld_sshd_port_enabled** incorrectly handled lists of network interfaces. Additionally, configuration files had different names than required. This update has fixed the remediation. As a result, the remediation handles all network interfaces correctly, and configuration files have predictable names.

(BZ#2109602)

fagenrules --load now works correctly

Previously, the **fapolicyd** service did not correctly handle the signal hang up (SIGHUP). Consequently, **fapolicyd** terminated after receiving the SIGHUP signal, and the **fagenrules --load** command did not work properly. 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#2070639)

8.6. NETWORKING

The NetworkManager utility enforces correct ordering of IPv6 addresses from various sources

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. With this update, the problem has been fixed and the ordering priority now reflects this logic: **manual** > **dhcpv6** > **autoconf6**. However, the order of addresses under the **ipv6.addresses** setting did not change and the address added last still has the highest priority.

(BZ#2097270)

Asymmetric routing now works correctly

The previous minor version of RHEL 8 contained a change that caused connection tracking to fail in some cases. Consequently, asymmetric routing was not working correctly. This release reverts the change that was introduced in RHEL 8.6. As a result, the asymmetric routing works correctly.

(BZ#2062870)

8.7. KERNEL

A new ability to deprecate CgroupV1 memory.swappiness allowing for consistent swap behavior

CgroupV1 includes the **memory.swappiness** per-cgroup swappiness value that controls the swap behavior of the given cgroup.

However, **systemd** processes run within **cgroups** and the **sysctl** swappiness value has minimal effect on **swap** heuristics. Such cgroups ignore the values in **sysctl** or **tuned** configurations and processes running on the system are assigned a default swappiness value of **60**. As a consequence, in cases with high memory pressure and page reclamation, earlier or more aggressive swapping can occur compared to the assigned swappiness value.

This update introduces a new **sysctl** variable, **/proc/sys/vm/force_cgroupv2_swappiness**, with a default value of **0**. When set to **1**, the **memory.swappiness** value becomes deprecated and all per-cgroups swappiness values mirror the system-wide swappiness value in the **/proc/sys/vm/swappiness** file. As a result, the memory swapping behavior of cgroups is more consistent.

(BZ#2084242)

Anaconda no longer fails after entering a passphrase for encrypted devices

Previously, if **kdump** was disabled when preparing an installation, and the user selected encrypted disk partitioning, the Anaconda installer failed with a traceback after entering a passphrase for the encrypted device.

This update fixes the problem, and users no longer need to enable **kdump** to create encrypted disk partitioning.

(BZ#2086100)

The net_prio or net_cls controllers in v1 mode now work correctly

Previously, in **cgroup-v2** environments, using either **net_prio** or **net_cls** controllers in v1 mode disabled the hierarchical tracking of socket data. As a consequence, the **cgroup-v2** hierarchy for socket data tracking controllers was not active, and the **dmesg** command reported the following message:

```
cgroup: cgroup: disabling cgroup2 socket matching due to net_prio or net_cls activation
```

This update ensures **cgroup-v2** is correctly active after the reboot.

(BZ#2046396)

8.8. 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)

8.9. HIGH AVAILABILITY AND CLUSTERS

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#1786964)

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#1954099)

8.10. DYNAMIC PROGRAMMING LANGUAGES, WEB AND DATABASE SERVERS

MariaDB 10.5 now warns about dropping a non-existent table when the OQGraph plug-in is enabled

Previously, when the **OQGraph** storage engine plug-in was loaded to the **MariaDB 10.5** server, **MariaDB** did not warn about dropping a non-existent table. In particular, when the user attempted to drop a non-existent table using the **DROP TABLE** or **DROP TABLE IF EXISTS** SQL commands, **MariaDB** neither returned an error message nor logged a warning. This bug has been fixed, and a warning is now shown in the described scenario.

(BZ#1944653)

8.11. COMPILERS AND DEVELOPMENT TOOLS

Applications no longer deadlock when invoking pthread_atfork or dclose from fork handler callbacks

Previously, applications invoked **pthread_atfork** handler callbacks while **glibc** had acquired an internal lock. As a result, registering fork handlers or calling **dclose** from a fork handler could deadlock applications.

A different synchronization mechanism is now used to protect internal data structures while fork handlers are running. As a result, applications no longer deadlock when invoking **pthread_atfork** or **dclose** from fork handler callbacks.

(BZ#1888660)

Wildcard functions in Makefiles no longer return symbolic links when only directories are expected

Previously, the **GLOB_ONLYDIR** hint used by **glob()** misreported symbolic links as directories on certain XFS filesystems. When using **glob()**, **make** did not confirm that the hints were actually directories and, as a result, wildcard functions in Makefiles returned symbolic links when only directories were expected.

The bug has been fixed and wildcard functions in Makefiles no longer return symbolic links when only directories are expected.

(BZ#1982608)

popen() no longer causes multithreaded processes to crash

Previously, a defect in **popen()** caused applications to crash when using the interface from a multithreaded process. With this update, the bug has been fixed and multithreaded processes no longer crash when using **popen()**.

(BZ#2065588)

The mapping for the 0xBC code point for some IBM character sets is now U+00AF MACRON

Previously, the **IBM256**, **IBM277**, **IBM278**, **IBM280**, **IBM284**, **IBM297**, and **IBM424** character sets encoded the **EBCDIC** code point **0xBC** as the Unicode character **U+203E OVERLINE**. As a result, when using the **iconv** program provided by **glibc**, converting text in those character sets containing the **0xBC** code point failed for non-Unicode character sets such as **ISO-8859-1** because they could not encode the **U+203E OVERLINE** character.

With this update, the bug has been fixed. As a result, input in the **IBM277**, **IBM278**, **IBM280**, **IBM284**, and **IBM297** character sets can be converted to **ISO-8859-1** in all cases. For the **IBM256** and **IBM424** character sets, conversion no longer fails if the input text contains the 0xBC code point and the respective output is **U+00AF MACRON**.

(BZ#1961109)

The tempnam function now uses getrandom to increase the randomness of generated file names

Previously, the **tempnam** function in Red Hat Enterprise Linux 8.4 and later used time-derived randomness for choosing paths. As a result, the **tempnam** function was not producing the full set of possible file names when invoked repeatedly in quick succession. This bug has been fixed by a new implementation that uses the **getrandom** function to increase the randomness of the generated file names. As a result, the **tempnam** function now generates more distinct file names.

(BZ#2089247)

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#2091553)

The en_US@ampm locale is now listed correctly by locale -a

Previously, there was a defect in the listing of **en_US@ampm** in the output of the **locale -a** command. Consequently, the **setlocale** API failed when trying to set this locale using its name/alias printed by **locale -a**. With this update, **en_US@ampm** is now listed correctly and calls to **setlocale** succeed for all locales printed by **locale -a**.

(BZ#2104907)

Unit masks for events are now all included in the papi_xml_event_info output

Previously, the testing of event unit mask information in **papi_xml_event_info** was incomplete. In some cases, unit masks for events were not included in the **papi_xml_event_info** output. This bug has been fixed and as a result, **papi_xml_event_command** now prints out all the unit masks for an event.

(BZ#2037426)

8.12. IDENTITY MANAGEMENT

Debug messages no longer logged to /var/log/messages by default

Previously, the **ipa-dnskeysyncd** and **ipa-ods-exporter** daemons logged all debug messages to **/var/log/messages** by default, resulting in log files growing substantially. If required, you can now configure the debug log level by setting **debug=True** in the **/etc/ipa/dns.conf** file. For more information refer to the **default.conf(5)** man page.

(BZ#2059396)

Preserving users accounts

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"**. This message incorrectly indicates that the user was deleted and not preserved as expected. With this update, the output now returns **Preserved user "user_login"**.

(BZ#2022028)

Transferring Kerberos databases greater than 4 GB

Previously, the **kprop** service and the **kpropd** command used a 32 bit value when storing the size of the Kerberos KDC database. As a result the transfer of the Kerberos database dump file from the primary Kerberos server to a replica server failed if the database size exceeded 4 GB.

This update modifies Kerberos and it can now transfer KDC databases greater than 4 GB.

(BZ#2026462)

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#2069379)

8.13. DESKTOP

The Airplane Mode switch is always displayed

Previously, the **Airplane Mode** switch in the **Wi-Fi** section of the **Settings** application disappeared after you enabled airplane mode. With this update, the problem has been fixed, and **Settings** always display the **Airplane Mode** switch, regardless of its state.

(BZ#2079139)

8.14. GRAPHICS INFRASTRUCTURES

Hotkeys in Motif applications activate the correct item

Previously, menu hotkeys activated the wrong menu item in applications using the Motif toolkit. When a submenu was open and you pressed a hotkey associated with its item, the application activated an item in the parent menu instead.

With this update, the problem has been fixed, and hotkeys now activate the correct submenu items.

(BZ#2060571)

The desktop no longer fails to start with disabled IPv6 and DisallowTCP=false

Previously, the X11 desktop session failed to start after login under the following circumstances:

- IPv6 networking was disabled on your system.
- The **DisallowTCP=false** option was enabled in GDM configuration.

With this update, the problem has been fixed, and you can log into the X11 session as expected with the described configuration.

(BZ#2075132)

8.15. 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)

8.16. RED HAT ENTERPRISE LINUX SYSTEM ROLES

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#2064067)

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#2115884)

Configuration by the metrics RHEL System Role 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 problem 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#2060377)

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. With this fix, the **tlog** role now updates the **nsswitch.conf** to ensure **tlog-rec-session** is correctly overlaid by SSSD.

(BZ#2072749)

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#2083378)

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#2100285)

The kernel_settings RHEL System Role now correctly installs python3-configobj

Previously, the **kernel_settings** role returned an error that the **python3-configobj** package could not be found. The role failed to find the package because it did not install **python3-configobj** on managed hosts. With this update, the role now installs **python3-configobj** on managed hosts and works correctly.

(BZ#2060378)

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#2083426)

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#2100298)

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#2101607)

The nbde_client 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** System Role uses proper spacing when creating add-on Dracut configuration files. As a result, the role correctly sets Dracut command-line parameters.

(BZ#2115161)

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 **RSAMinSize** option in the **ssh** and **sshd** RHEL System Roles.

(BZ#2109997)

The NBDE Client System Role supports static IP addresses

In previous versions of RHEL, restarting a system with a static IP address and configured with the Network Bound Disk Encryption (NBDE) Client System Role would change the system's IP address.

With this change, systems with static IP addresses are supported by the NBDE Client System Role, and their IP addresses do not change after a reboot.

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

(BZ#2071011)

8.17. VIRTUALIZATION

Live pre-copy migration of VMs with failover VFs now works correctly

Previously, attempting to pre-copy migrate a running virtual machine (VM) failed if the VM used a device with the virtual function (VF) failover capability enabled. This update fixes the problem, and migrating VMs in the described scenario now works correctly.

(BZ#2054656)

8.18. RHEL IN CLOUD ENVIRONMENTS

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#2082000)

SR-IOV no longer performs suboptimally in ARM 64 RHEL 8 virtual machines on Azure

Previously, SR-IOV networking devices had significantly lower throughout and higher latency than expected in ARM 64 RHEL 8 virtual machines (VMs) running on a Microsoft Azure platform. The problem has been fixed, and the affected VMs now perform as expected.

(BZ#2068429)

Starting a RHEL 8 virtual machine on AWS using **cloud-init** no longer takes longer than expected

Previously, initializing an EC2 instance of RHEL 8 using the **cloud-init** service on Amazon Web Services (AWS) took an excessive amount of time. The Amazon Machine Images (AMIs) of RHEL 8 have been updated to include a fix for the problem, and intializing EC2 instances of RHEL 8 now works correctly.

However, you might still encounter slow intialization when customizing and uploading your own RHEL 8 image. To avoid this problem, remove the **/etc/resolv.conf** file from the image you are using for VM creation before uploading the image to AWS.

(BZ#1862930)

8.19. CONTAINERS

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)

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#2020301)

CHAPTER 9. TECHNOLOGY PREVIEWS

This part provides a list of all Technology Previews available in Red Hat Enterprise Linux 8.7.

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#1868421)

9.2. NETWORKING

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)

AF_XDP available as a Technology Preview

Address Family eXpress Data Path (**AF_XDP**) socket is designed for high-performance packet processing. It accompanies **XDP** and grants efficient redirection of programmatically selected packets to user space applications for further processing.

(BZ#1633143)

XDP features that are available as Technology Preview

Red Hat provides the usage of the following eXpress Data Path (XDP) features as unsupported Technology Preview:

- Loading XDP programs on architectures other than AMD and Intel 64-bit. Note that the **libxdp** library is not available for architectures other than AMD and Intel 64-bit.
- The XDP hardware offloading.

(BZ#1889737)

Multi-protocol Label Switching for TC available as a Technology Preview

The Multi-protocol Label Switching (MPLS) is an in-kernel data-forwarding mechanism to route traffic flow across enterprise networks. In an MPLS network, the router that receives packets decides the further route of the packets based on the labels attached to the packet. With the usage of labels, the MPLS network has the ability to handle packets with particular characteristics. For example, you can add **tc filters** for managing packets received from specific ports or carrying specific types of traffic, in a consistent way.

After packets enter the enterprise network, MPLS routers perform multiple operations on the packets, such as **push** to add a label, **swap** to update a label, and **pop** to remove a label. MPLS allows defining actions locally based on one or multiple labels in RHEL. You can configure routers and set traffic control (**tc**) filters to take appropriate actions on the packets based on the MPLS label stack entry (**lse**) elements, such as **label**, **traffic class**, **bottom of stack**, and **time to live**.

For example, the following command adds a filter to the *enp0s1* network interface to match incoming packets having the first label *12323* and the second label *45832*. On matching packets, the following actions are taken:

- the first MPLS TTL is decremented (packet is dropped if TTL reaches 0)
- the first MPLS label is changed to *549386*
- the resulting packet is transmitted over *enp0s2*, with destination MAC address *00:00:5E:00:53:01* and source MAC address *00:00:5E:00:53:02*

```
# tc filter add dev enp0s1 ingress protocol mpls_uc flower mpls lse depth 1 label 12323 lse
depth 2 label 45832 \
action mpls dec_ttl pipe \
action mpls modify label 549386 pipe \
action pedit ex munge eth dst set 00:00:5E:00:53:01 pipe \
action pedit ex munge eth src set 00:00:5E:00:53:02 pipe \
action mirred egress redirect dev enp0s2
```

(BZ#1814836, BZ#1856415)

The systemd-resolved service is now 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, even if the **systemd** package provides **systemd-resolved**, this service is an unsupported Technology Preview.

(BZ#1906489)

9.3. KERNEL

The kexec fast reboot feature is available as a Technology Preview

The **kexec** fast reboot feature continues to be available as a Technology Preview. The **kexec** fast reboot significantly speeds the boot process as the kernel enables booting directly into the second kernel without passing through the Basic Input/Output System (BIOS) first. To use this feature:

1. Load the **kexec** kernel manually.
2. Reboot the operating system.

(BZ#1769727)

The accel-config package available as a Technology Preview

The **accel-config** package is now available on Intel **EM64T** and **AMD64** architectures as a Technology Preview. This package helps in controlling and configuring data-streaming accelerator (DSA) sub-system in the Linux Kernel. Also, it configures devices through **sysfs** (pseudo-filesystem), saves and loads the configuration in the **json** format.

(BZ#1843266)

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#1660337)

eBPF available as a Technology Preview

Extended Berkeley Packet Filter (eBPF)is an in-kernel virtual machine that allows code execution in the kernel space, in the restricted sandbox environment with access to a limited set of functions.

The virtual machine includes a new system call **bpf()**, which enables creating various types of maps, and also allows to load programs in a special assembly-like code. The code is then loaded to the kernel and translated to the native machine code with just-in-time compilation. Note that the **bpf()** syscall can be successfully used only by a user with the **CAP_SYS_ADMIN** capability, such as the root user. See the **bpf(2)** manual page for more information.

The loaded programs can be attached onto a variety of points (sockets, tracepoints, packet reception) to receive and process data.

There are numerous components shipped by Red Hat that utilize the **eBPF** virtual machine. Each component is in a different development phase. All components are available as a Technology Preview, unless a specific component is indicated as supported.

The following notable **eBPF** components are currently available as a Technology Preview:

- **AF_XDP**, a socket for connecting the **eXpress Data Path (XDP)** path to user space for applications that prioritize packet processing performance.

(BZ#1559616)

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 a shared work queue with process address space ID (pasid) submission and shared virtual memory (SVM).

(BZ#1837187)

Soft-RoCE available as a Technology Preview

Remote Direct Memory Access (RDMA) over Converged Ethernet (RoCE) is a network protocol that implements RDMA over Ethernet. Soft-RoCE is the software implementation of RoCE which maintains two protocol versions, RoCE v1 and RoCE v2. The Soft-RoCE driver, **rdma_rxe**, is available as an unsupported Technology Preview in RHEL 8.

(BZ#1605216)

The stmmac driver is available as a Technology Preview

Red Hat provides the usage of **stmmac** for Intel® Elkhart Lake systems on a chip (SoCs) as an unsupported Technology Preview.

(BZ#1905243)

9.4. FILE SYSTEMS AND STORAGE

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

In Red Hat Enterprise Linux 8, the file system DAX is available as a Technology Preview. DAX provides a 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 file system that provides the capability of DAX must be created on the NVDIMM(s). Also, the file system must be mounted with the **dax** mount option. Then, a **mmap** of a file on the dax-mounted file system results in a direct mapping of storage into the application's address space.

(BZ#1627455)

OverlayFS

OverlayFS is a type of union file system. It enables you to overlay one file system on top of another. Changes are recorded in the upper file system, while the lower file system remains unmodified. This allows multiple users to share a file-system image, such as a container or a DVD-ROM, where the base image is on read-only media.

OverlayFS remains a Technology Preview under most circumstances. As such, the kernel logs warnings when this technology is activated.

Full support is available for OverlayFS when used with supported container engines (**podman**, **cri-o**, or **buildah**) under the following restrictions:

- OverlayFS is supported for use only as a container engine graph driver. Its use is supported only for container COW content, not for persistent storage. You must place any persistent storage on non-OverlayFS volumes. You can use only the default container engine configuration: one level of overlay, one lowerdir, and both lower and upper levels are on the same file system.

- Only XFS is currently supported for use as a lower layer file system.

Additionally, the following rules and limitations apply to using OverlayFS:

- The OverlayFS kernel ABI and user-space behavior are not considered stable, and might change in future updates.
- OverlayFS provides a restricted set of the POSIX standards. Test your application thoroughly before deploying it with OverlayFS. The following cases are not POSIX-compliant:
  - Lower files opened with **O_RDONLY** do not receive **st_atime** updates when the files are read.
  - Lower files opened with **O_RDONLY**, then mapped with **MAP_SHARED** are inconsistent with subsequent modification.
  - Fully compliant **st_ino** or **d_ino** values are not enabled by default on RHEL 8, but you can enable full POSIX compliance for them with a module option or mount option.
    To get consistent inode numbering, use the **xino=on** mount option.

    You can also use the **redirect_dir=on** and **index=on** options to improve POSIX compliance. These two options make the format of the upper layer incompatible with an overlay without these options. That is, you might get unexpected results or errors if you create an overlay with **redirect_dir=on** or **index=on**, unmount the overlay, then mount the overlay without these options.
- To determine whether an existing XFS file system is eligible for use as an overlay, use the following command and see if the **ftype=1** option is enabled:

```
# xfs_info /mount-point | grep ftype
```

- SELinux security labels are enabled by default in all supported container engines with OverlayFS.
- Several known issues are associated with OverlayFS in this release. For details, see *Non-standard behavior* in the Linux kernel documentation.

For more information about OverlayFS, see the Linux kernel documentation.

(BZ#1690207)

Stratis is now available as a Technology Preview

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

Stratis enables you to more easily perform storage tasks such as:

- 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.

RHEL 8.3 updated Stratis to version 2.1.0. For more information, see Stratis 2.1.0 Release Notes.

(JIRA:RHELPLAN-1212)

Setting up a Samba server on an IdM domain member is provided as a Technology Preview

With this update, you can now set up a Samba server on an Identity Management (IdM) domain member. The new **ipa-client-samba** utility provided by the same-named package adds a Samba-specific Kerberos service principal to IdM and prepares the IdM client. For example, the utility creates the **/etc/samba/smb.conf** with the ID mapping configuration for the **sss** ID mapping back end. As a result, administrators can now set up Samba on an IdM domain member.

Due to IdM Trust Controllers not supporting the Global Catalog Service, AD-enrolled Windows hosts cannot find IdM users and groups in Windows. Additionally, IdM Trust Controllers do not support resolving IdM groups using the Distributed Computing Environment / Remote Procedure Calls (DCE/RPC) protocols. As a consequence, AD users can only access the Samba shares and printers from IdM clients.

For details, see Setting up Samba on an IdM domain member.

(JIRA:RHELPLAN-13195)

NVMe/TCP host is available as a Technology Preview

Accessing and sharing Nonvolatile Memory Express (NVMe) storage over TCP/IP networks (NVMe/TCP) and its corresponding **nvme_tcp.ko** kernel module has been added as a Technology Preview. The use of NVMe/TCP as a host is manageable with tools provided by the **nvme-cli** package. The NVMe/TCP host Technology Preview is included only for testing purposes and is not currently planned for full support.

(BZ#1696451)

9.5. HIGH AVAILABILITY AND CLUSTERS

Pacemaker podman bundles available as a Technology Preview

Pacemaker container bundles now run on Podman, with the container bundle feature being available as a Technology Preview. There is one exception to this feature being Technology Preview: Red Hat fully supports the use of Pacemaker bundles for Red Hat Openstack.

(BZ#1619620)

Heuristics in corosync-qdevice available as a Technology Preview

Heuristics are a set of commands executed locally on startup, cluster membership change, successful connect to **corosync-qnetd**, and, optionally, on a periodic basis. When all commands finish successfully on time (their return error code is zero), heuristics have passed; otherwise, they have failed. The heuristics result is sent to **corosync-qnetd** where it is used in calculations to determine which partition should be quorate.

(BZ#1784200)

New fence-agents-heuristics-ping fence agent

As a Technology Preview, Pacemaker now provides the **fence_heuristics_ping** agent. This agent aims to open a class of experimental fence agents that do no actual fencing by themselves but instead exploit the behavior of fencing levels in a new way.

If the heuristics agent is configured on the same fencing level as the fence agent that does the actual fencing but is configured before that agent in sequence, fencing issues an **off** action on the heuristics agent before it attempts to do so on the agent that does the fencing. If the heuristics agent gives a negative result for the **off** action it is already clear that the fencing level is not going to succeed, causing Pacemaker fencing to skip the step of issuing the **off** action on the agent that does the fencing. A heuristics agent can exploit this behavior to prevent the agent that does the actual fencing from fencing a node under certain conditions.

A user might want to use this agent, especially in a two-node cluster, when it would not make sense for a node to fence the peer if it can know beforehand that it would not be able to take over the services properly. For example, it might not make sense for a node to take over services if it has problems reaching the networking uplink, making the services unreachable to clients, a situation which a ping to a router might detect in that case.

(BZ#1775847)

Automatic removal of location constraint following resource move available as a Technology Preview

When you execute the **pcs resource move** command, this adds a constraint to the resource to prevent it from running on the node on which it is currently running. A new **--autodelete** option for the **pcs resource move** command is now available as a Technology Preview. When you specify this option, the location constraint that the command creates is automatically removed once the resource has been moved.

(BZ#1847102)

9.6. IDENTITY MANAGEMENT

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#1664719)

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#1664718)

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#1628987)

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 8.7, 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#2101770)

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#2065692)

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#2056483)

9.7. DESKTOP

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

The GNOME desktop environment is now available for the 64-bit ARM architecture as a Technology Preview. This enables administrators to configure and manage servers from a graphical user interface (GUI) remotely, using the VNC session.

As a consequence, new administration applications are available on the 64-bit ARM architecture. For example: Disk Usage Analyzer (**baobab**), Firewall Configuration (**firewall-config**), Red Hat Subscription Manager (**subscription-manager-cockpit**), or the Firefox web browser. Using Firefox, administrators can connect to the local Cockpit daemon remotely.

(JIRA:RHELPLAN-27394, BZ#1667225, BZ#1667516, BZ#1724302)

GNOME desktop on IBM Z is available as a Technology Preview

The GNOME desktop, including the Firefox web browser, is now available as a Technology Preview on the IBM Z architecture. You can now connect to a remote graphical session running GNOME using VNC to configure and manage your IBM Z servers.

(JIRA:RHELPLAN-27737)

9.8. GRAPHICS INFRASTRUCTURES

VNC remote console available as a Technology Preview for the 64-bit ARM architecture

On the 64-bit ARM architecture, the Virtual Network Computing (VNC) remote console is available as a Technology Preview. Note that the rest of the graphics stack is currently unverified for the 64-bit ARM architecture.

(BZ#1698565)

9.9. VIRTUALIZATION

AMD SEV and SEV-ES for KVM virtual machines

As a Technology Preview, RHEL 8 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 8 includes SEV and SEV-ES encryption, but not the SEV and SEV-ES security attestation.

(BZ#1501618, BZ#1501607, JIRA:RHELPLAN-7677)

Intel vGPU

As a Technology Preview, it is now 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 only selected Intel GPUs are compatible with the vGPU feature.

In addition, it is possible to enable a VNC console operated by Intel vGPU. By enabling it, users can connect to a VNC console of the VM and see the VM's desktop hosted by Intel vGPU. However, this currently only works for RHEL guest operating systems.

(BZ#1528684)

Creating nested virtual machines

Nested KVM virtualization is provided as a Technology Preview for KVM virtual machines (VMs) running on Intel, AMD64, IBM POWER, and IBM Z systems hosts with RHEL 8. With this feature, a RHEL 7 or RHEL 8 VM that runs on a physical RHEL 8 host can act as a hypervisor, and host its own VMs.

(JIRA:RHELPLAN-14047, JIRA:RHELPLAN-24437)

Technology Preview: Select Intel network adapters now provide SR-IOV in RHEL guests on Hyper-V

As a Technology Preview, Red Hat Enterprise Linux guest operating systems running on a Hyper-V hypervisor can now use the single-root I/O virtualization (SR-IOV) feature for Intel network adapters that are supported by the **ixgbevf** and **iavf** drivers. This feature is enabled when the following conditions are met:

- SR-IOV support is enabled for the network interface controller (NIC)
- SR-IOV support is enabled for the virtual NIC
- SR-IOV support is enabled for the virtual switch
- The virtual function (VF) from the NIC is attached to the virtual machine

The feature is currently provided with Microsoft Windows Server 2016 and later.

(BZ#1348508)

Sharing files between hosts and VMs using virtiofs

As a Technology Preview, RHEL 8 now provides the virtio file system (**virtiofs**). Using **virtiofs**, you can efficiently share files between your host system and its virtual machines (VM).

(BZ#1741615)

KVM virtualization is usable in RHEL 8 Hyper-V virtual machines

As a Technology Preview, nested KVM virtualization can now be used on the Microsoft Hyper-V hypervisor. As a result, you can create virtual machines on a RHEL 8 guest system running on a Hyper-V host.

Note that currently, this feature only works on Intel and AMD systems. In addition, nested virtualization is in some cases not enabled by default on Hyper-V. To enable it, see the following Microsoft documentation:

https://docs.microsoft.com/en-us/virtualization/hyper-v-on-windows/user-guide/nested-virtualization

(BZ#1519039)

9.10. 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-122316)

9.11. CONTAINERS

Toolbox is available as a Technology Preview

Previously, the Toolbox utility was based on RHEL CoreOS coreos/toolbox. With this release, Toolbox has been replaced with containers/toolbox.

(JIRA:RHELPLAN-77238)

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-75165)

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-118470)

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

This part provides an overview of functionality that has been *deprecated* in Red Hat Enterprise Linux 8.

Deprecated functionality will likely not be supported in future major releases of this product and is not recommended for new deployments. For the most recent list of deprecated functionality within a particular major release, refer to the latest version of release documentation.

The support status of deprecated functionality remains unchanged within Red Hat Enterprise Linux 8. For information about the length of support, see Red Hat Enterprise Linux Life Cycle and Red Hat Enterprise Linux Application Streams Life Cycle .

Deprecated hardware components are not recommended for new deployments on the current or future major releases. Hardware driver updates are limited to security and critical fixes only. Red Hat recommends replacing this hardware as soon as reasonably feasible.

A package can be deprecated and not recommended for further use. Under certain circumstances, a package can be removed from a 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 7 but has been *removed* in RHEL 8, see Considerations in adopting RHEL 8 .

10.1. INSTALLER AND IMAGE CREATION

Several Kickstart commands and options have been deprecated

Using the following commands and options in RHEL 8 Kickstart files will print a warning in the logs:

- **auth** or **authconfig**
- **device**
- **deviceprobe**
- **dmraid**
- **install**
- **lilo**
- **lilocheck**
- **mouse**
- **multipath**
- **bootloader --upgrade**
- **ignoredisk --interactive**
- **partition --active**
- **reboot --kexec**

Where only specific options are listed, the base command and its other options are still available and not deprecated.

For more details and related changes in Kickstart, see the Kickstart changes section of the *Considerations in adopting RHEL 8* document.

(BZ#1642765)

The --interactive option of the ignoredisk Kickstart command has been deprecated

Using the **--interactive option** in future releases of Red Hat Enterprise Linux will result in a fatal installation error. It is recommended that you modify your Kickstart file to remove the option.

(BZ#1637872)

The Kickstart autostep command has been deprecated

The **autostep** command has been deprecated. The related section about this command has been removed from the RHEL 8 documentation.

(BZ#1904251)

10.2. SOFTWARE MANAGEMENT

rpmbuild --sign is deprecated

The **rpmbuild --sign** command is deprecated since RHEL 8.1. Using this command in future releases of Red Hat Enterprise Linux can result in an error. It is recommended that you use the **rpmsign** command instead.

(BZ#1688849)

10.3. SHELLS AND COMMAND-LINE TOOLS

The OpenEXR component has been deprecated

The **OpenEXR** component has been deprecated. Hence, the support for the **EXR** image format has been dropped from the **imagecodecs** module.

(BZ#1886310)

The dump utility from the dump package has been deprecated

The **dump** utility used for backup of file systems has been deprecated and will not be available in RHEL 9.

In RHEL 9, Red Hat recommends using the **tar**, **dd**, or **bacula**, backup utility, based on type of usage, which provides full and safe backups on ext2, ext3, and ext4 file systems.

Note that the **restore** utility from the **dump** package remains available and supported in RHEL 9 and is available as the **restore** package.

(BZ#1997366)

The ABRT tool has been deprecated

The Automatic Bug Reporting Tool (ABRT) for detecting and reporting application crashes has been deprecated in RHEL 8. As a replacement, use the **systemd-coredump** tool to log and store core dumps, which are automatically generated files after a program crashes.

(BZ#2055826)

The ReaR crontab has been deprecated

The **/etc/cron.d/rear** crontab from the **rear** package has been deprecated in RHEL 8 and will not be available in RHEL 9. The crontab checks every night whether the disk layout has changed, and runs **rear mkrescue** command if a change happened.

If you require this functionality, after an upgrade to RHEL 9, configure periodic runs of ReaR manually.

(BZ#2083301)

The SQLite database backend in Bacula has been deprecated

The Bacula backup system supported multiple database backends: PostgreSQL, MySQL, and SQLite. The SQLite backend has been deprecated and will become unsupported in a later release of RHEL. As a replacement, migrate to one of the other backends (PostgreSQL or MySQL) and do not use the SQLite backend in new deployments.

(BZ#2089399)

The hidepid=n mount option is not supported in RHEL 8 systemd

The mount option **hidepid=n**, which controls who can access information in **/proc/[pid]** directories, is not compatible with **systemd** infrastructure provided in RHEL 8.

In addition, using this option might cause certain services started by **systemd** to produce SELinux AVC denial messages and prevent other operations from completing.

For more information, see the related Knowledgebase solution Is mounting /proc with "hidepid=2" recommended with RHEL7 and RHEL8?.

(BZ#2038929)

The /usr/lib/udev/rename_device utility has been deprecated

The **udev** helper utility **/usr/lib/udev/rename_device** for renaming network interfaces has been deprecated.

(BZ#1875485)

The raw command has been deprecated

The **raw** (**/usr/bin/raw**) command has been deprecated. Using this command in future releases of Red Hat Enterprise Linux can result in an error.

(JIRA:RHELPLAN-133171)

10.4. SECURITY

NSS SEED ciphers are deprecated

The Mozilla Network Security Services (**NSS**) library will not support TLS cipher suites that use a SEED cipher in a future release. To ensure smooth transition of deployments that rely on SEED ciphers when NSS removes support, Red Hat recommends enabling support for other cipher suites.

Note that SEED ciphers are already disabled by default in RHEL.

(BZ#1817533)

TLS 1.0 and TLS 1.1 are deprecated

The TLS 1.0 and TLS 1.1 protocols are disabled in the **DEFAULT** system-wide cryptographic policy level. If your scenario, for example, a video conferencing application in the Firefox web browser, requires using the deprecated protocols, switch the system-wide cryptographic policy to the **LEGACY** level:

```
# update-crypto-policies --set LEGACY
```

For more information, see the Strong crypto defaults in RHEL 8 and deprecation of weak crypto algorithms Knowledgebase article on the Red Hat Customer Portal and the **update-crypto-policies(8)** man page.

(BZ#1660839)

DSA is deprecated in RHEL 8

The Digital Signature Algorithm (DSA) is considered deprecated in Red Hat Enterprise Linux 8. Authentication mechanisms that depend on DSA keys do not work in the default configuration. Note that **OpenSSH** clients do not accept DSA host keys even in the **LEGACY** system-wide cryptographic policy level.

(BZ#1646541)

SSL2 Client Hello has been deprecated in NSS

The Transport Layer Security (**TLS**) protocol version 1.2 and earlier allow to start a negotiation with a **Client Hello** message formatted in a way that is backward compatible with the Secure Sockets Layer (**SSL**) protocol version 2. Support for this feature in the Network Security Services (**NSS**) library has been deprecated and it is disabled by default.

Applications that require support for this feature need to use the new **SSL_ENABLE_V2_COMPATIBLE_HELLO** API to enable it. Support for this feature may be removed completely in future releases of Red Hat Enterprise Linux 8.

(BZ#1645153)

TPM 1.2 is deprecated

The Trusted Platform Module (TPM) secure cryptoprocessor standard was updated to version 2.0 in 2016. TPM 2.0 provides many improvements over TPM 1.2, and it is not backward compatible with the previous version. TPM 1.2 is deprecated in RHEL 8, and it might be removed in the next major release.

(BZ#1657927)

crypto-policies derived properties are now deprecated

With the introduction of scopes for **crypto-policies** directives in custom policies, the following derived properties have been deprecated: **tls_cipher**, **ssh_cipher**, **ssh_group**, **ike_protocol**, and **sha1_in_dnssec**. Additionally, the use of the **protocol** property without specifying a scope is now deprecated as well. See the **crypto-policies(7)** man page for recommended replacements.

(BZ#2011208)

Runtime disabling SELinux using /etc/selinux/config is now deprecated

Runtime disabling SELinux using the **SELINUX=disabled** option in the **/etc/selinux/config** file has been deprecated. In RHEL 9, when you disable SELinux only through **/etc/selinux/config**, the system starts with SELinux enabled but with no policy loaded.

If your scenario really requires to completely disable SELinux, Red Hat recommends disabling SELinux by adding the **selinux=0** parameter to the kernel command line as described in the Changing SELinux modes at boot time section of the Using SELinux title.

(BZ#1932222)

The ipa SELinux module removed from selinux-policy

The **ipa** SELinux module has been removed from the **selinux-policy** package because it is no longer maintained. The functionality is now included in the **ipa-selinux** subpackage.

If your scenario requires the use of types or interfaces from the **ipa** module in a local SELinux policy, install the **ipa-selinux** package.

(BZ#1461914)

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#2054741)

10.5. NETWORKING

Network scripts are deprecated in RHEL 8

Network scripts are deprecated in Red Hat Enterprise Linux 8 and they are no longer provided by default. The basic installation provides a new version of the **ifup** and **ifdown** scripts which call the NetworkManager service through the **nmcli** tool. In Red Hat Enterprise Linux 8, to run the **ifup** and the **ifdown** scripts, NetworkManager must be running.

Note that custom commands in **/sbin/ifup-local**, **ifdown-pre-local** and **ifdown-local** scripts are not executed.

If any of these scripts are required, the installation of the deprecated network scripts in the system is still possible with the following command:

```
# yum install network-scripts
```

The **ifup** and **ifdown** scripts link to the installed legacy network scripts.

Calling the legacy network scripts shows a warning about their deprecation.

(BZ#1647725)

The dropwatch tool is deprecated

The **dropwatch** tool has been deprecated. The tool will not be supported in future releases, thus it is not recommended for new deployments. As a replacement of this package, Red Hat recommends to use the **perf** command line tool.

For more information on using the **perf** command line tool, see the Getting started with Perf section on the Red Hat customer portal or the **perf** man page.

(BZ#1929173)

The cgdcbxd package is deprecated

Control group data center bridging exchange daemon (**cgdcbxd**) is a service to monitor data center bridging (DCB) netlink events and manage the **net_prio control** group subsystem. Starting with RHEL 8.5, the **cgdcbxd** package is deprecated and will be removed in the next major RHEL release.

(BZ#2006665)

The xinetd service has been deprecated

The **xinetd** service has been deprecated and will be removed in RHEL 9. As a replacement, use **systemd**. For further details, see How to convert xinetd service to systemd .

(BZ#2009113)

The WEP Wi-Fi connection method is deprecated

The insecure wired equivalent privacy (WEP) Wi-Fi connection method is deprecated in RHEL 8 and will be removed in RHEL 9.0. For secure Wi-Fi connections, use the Wi-Fi Protected Access 3 (WPA3) or WPA2 connection methods.

(BZ#2029338)

The unsupported xt_u32 module is now deprecated

Using the unsupported **xt_u32** module, users of **iptables** can match arbitrary 32 bits in the packet header or payload. Since RHEL 8.6, the **xt_u32** module is deprecated and will be removed in RHEL 9.

If you use **xt_u32**, migrate to the **nftables** packet filtering framework. For example, first change your firewall to use **iptables** with native matches to incrementally replace individual rules, and later use the **iptables-translate** and accompanying utilities to migrate to **nftables**. If no native match exists in **nftables**, use the raw payload matching feature of **nftables**. For details, see the **raw payload expression** section in the **nft(8)** man page.

(BZ#2061288)

The term slaves is deprecated in the nmstate API

Red Hat is committed to using conscious language. See details about this initiative in Making open source more inclusive. Therefore the **slaves** term is deprecated in the Nmstate API. Use the term **port** when you use **nmstatectl**.

(JIRA:RHELDOCS-17641)

10.6. KERNEL

Kernel live patching now covers all RHEL minor releases

Since RHEL 8.1, kernel live patches have been provided for selected minor release streams of RHEL

covered under the Extended Update Support (EUS) policy to remediate Critical and Important Common Vulnerabilities and Exposures (CVEs). To accommodate the maximum number of concurrently covered kernels and use cases, the support window for each live patch has been decreased from 12 to 6 months for every minor, major, and zStream version of the kernel. It means that on the day a kernel live patch is released, it will cover every minor release and scheduled errata kernel delivered in the past 6 months.

For more information about this feature, see Applying patches with kernel live patching .

For details about available kernel live patches, see Kernel Live Patch life cycles .

(BZ#1958250)

The crash-ptdump-command package is deprecated

The **crash-ptdump-command** package, which is a **ptdump** extension module for the crash utility, is deprecated and might not be available in future RHEL releases. The **ptdump** command fails to retrieve the log buffer when working in the Single Range Output mode and only works in the Table of Physical Addresses (ToPA) mode. **crash-ptdump-command** is currently not maintained upstream

(BZ#1838927)

Installing RHEL for Real Time 8 using diskless boot is now deprecated

Diskless booting allows multiple systems to share a root file system through the network. While convenient, diskless boot is prone to introducing network latency in real-time workloads. With a future minor update of RHEL for Real Time 8, the diskless booting feature will no longer be supported.

(BZ#1748980)

The Linux firewire sub-system and its associated user-space components are deprecated in RHEL 8

The **firewire** sub-system provides interfaces to use and maintain any resources on the IEEE 1394 bus. In RHEL 9, **firewire** will no longer be supported in the **kernel** package. Note that **firewire** contains several user-space components provided by the **libavc1394**, **libdc1394**, **libraw1394** packages. These packages are subject to the deprecation as well.

(BZ#1871863)

The rdma_rxe Soft-RoCE driver is deprecated

Software Remote Direct Memory Access over Converged Ethernet (Soft-RoCE), also known as RXE, is a feature that emulates Remote Direct Memory Access (RDMA). In RHEL 8, the Soft-RoCE feature is available as an unsupported Technology Preview. However, due to stability issues, this feature has been deprecated and will be removed in RHEL 9.

(BZ#1878207)

10.7. BOOT LOADER

The kernelopts environment variable has been deprecated

In RHEL 8, the kernel command-line parameters for systems using the GRUB2 bootloader were defined in the **kernelopts** environment variable. The variable was stored in the **/boot/grub2/grubenv** file for each kernel boot entry. However, storing the kernel command-line parameters using **kernelopts** was not robust. Therefore, with a future major update of RHEL, **kernelopts** will be removed and the kernel command-line parameters will be stored in the Boot Loader Specification (BLS) snippet instead.

(BZ#2060759)

10.8. FILE SYSTEMS AND STORAGE

VDO write modes other than async are deprecated

VDO supports several write modes in RHEL 8:

- **sync**
- **async**
- **async-unsafe**
- **auto**

Starting with RHEL 8.4, the following write modes are deprecated:

sync

Devices above the VDO layer cannot recognize if VDO is synchronous, and consequently, the devices cannot take advantage of the VDO **sync** mode.

async-unsafe

VDO added this write mode as a workaround for the reduced performance of **async** mode, which complies to Atomicity, Consistency, Isolation, and Durability (ACID). Red Hat does not recommend **async-unsafe** for most use cases and is not aware of any users who rely on it.

auto

This write mode only selects one of the other write modes. It is no longer necessary when VDO supports only a single write mode.

These write modes will be removed in a future major RHEL release.

The recommended VDO write mode is now **async**.

For more information on VDO write modes, see Selecting a VDO write mode.

(JIRA:RHELPLAN-70700)

NFSv3 over UDP has been disabled

The NFS server no longer opens or listens on a User Datagram Protocol (UDP) socket by default. This change affects only NFS version 3 because version 4 requires the Transmission Control Protocol (TCP).

NFS over UDP is no longer supported in RHEL 8.

(BZ#1592011)

cramfs has been deprecated

Due to lack of users, the **cramfs** kernel module is deprecated. **squashfs** is recommended as an alternative solution.

(BZ#1794513)

VDO manager has been deprecated

The python-based VDO management software has been deprecated and will be removed from RHEL 9. In RHEL 9, it will be replaced by the LVM-VDO integration. Therefore, it is recommended to create VDO volumes using the **lvcreate** command.

The existing volumes created using the VDO management software can be converted using the **/usr/sbin/lvm_import_vdo** script, provided by the **lvm2** package. For more information on the LVM-VDO implementation, see Deduplicating and compressing logical volumes on RHEL.

(BZ#1949163)

The elevator kernel command line parameter is deprecated

The **elevator** kernel command line parameter was used in earlier RHEL releases to set the disk scheduler for all devices. In RHEL 8, the parameter is deprecated.

The upstream Linux kernel has removed support for the **elevator** parameter, but it is still available in RHEL 8 for compatibility reasons.

Note that the kernel selects a default disk scheduler based on the type of device. This is typically the optimal setting. If you require a different scheduler, Red Hat recommends that you use **udev** rules or the TuneD service to configure it. Match the selected devices and switch the scheduler only for those devices.

For more information, see Setting the disk scheduler.

(BZ#1665295)

LVM mirror is deprecated

The LVM **mirror** segment type is now deprecated. Support for **mirror** will be removed in a future major release of RHEL.

Red Hat recommends that you use LVM RAID 1 devices with a segment type of **raid1** instead of **mirror**. The **raid1** segment type is the default RAID configuration type and replaces **mirror** as the recommended solution.

To convert **mirror** devices to **raid1**, see Converting a mirrored LVM device to a RAID1 logical volume.

LVM **mirror** has several known issues. For details, see known issues in file systems and storage.

(BZ#1827628)

peripety is deprecated

The **peripety** package is deprecated since RHEL 8.3.

The Peripety storage event notification daemon parses system storage logs into structured storage events. It helps you investigate storage issues.

(BZ#1871953)

10.9. HIGH AVAILABILITY AND CLUSTERS

pcs commands that support the clufter tool have been deprecated

The **pcs** commands that support the **clufter** tool for analyzing cluster configuration formats have been deprecated. These commands now print a warning that the command has been deprecated and sections related to these commands have been removed from the **pcs** help display and the **pcs(8)** man page.

The following commands have been deprecated:

- **pcs config import-cman** for importing CMAN / RHEL6 HA cluster configuration
- **pcs config export** for exporting cluster configuration to a list of **pcs** commands which recreate the same cluster

(BZ#1851335)

10.10. DYNAMIC PROGRAMMING LANGUAGES, WEB AND DATABASE SERVERS

The mod_php module provided with PHP for use with the Apache HTTP Server has been deprecated

The **mod_php** module provided with PHP for use with the Apache HTTP Server in RHEL 8 is available but not enabled in the default configuration. The module is no longer available in RHEL 9.

Since RHEL 8, PHP scripts are run using the FastCGI Process Manager (**php-fpm**) by default. For more information, see Using PHP with the Apache HTTP Server.

(BZ#2225332)

10.11. COMPILERS AND DEVELOPMENT TOOLS

libdwarf has been deprecated

The **libdwarf** library has been deprecated in RHEL 8. The library will likely not be supported in future major releases. Instead, use the **elfutils** and **libdw** libraries for applications that wish to process ELF/DWARF files.

Alternatives for the **libdwarf-tools dwarfdump** program are the **binutils readelf** program or the **elfutils eu-readelf** program, both used by passing the **--debug-dump** flag.

(BZ#1920624)

The gdb.i686 packages are deprecated

In RHEL 8.1, the 32-bit versions of the GNU Debugger (GDB), **gdb.i686**, were shipped due to a dependency problem in another package. Because RHEL 8 does not support 32-bit hardware, the **gdb.i686** packages are deprecated since RHEL 8.4. The 64-bit versions of GDB, **gdb.x86_64**, are fully capable of debugging 32-bit applications.

If you use **gdb.i686**, note the following important issues:

- The **gdb.i686** packages will no longer be updated. Users must install **gdb.x86_64** instead.
- If you have **gdb.i686** installed, installing **gdb.x86_64** will cause **yum** to report **package gdb-8.2-14.el8.x86_64 obsoletes gdb < 8.2-14.el8 provided by gdb-8.2-12.el8.i686**. This is expected. Either uninstall **gdb.i686** or pass **dnf** the **--allowerasing** option to remove **gdb.i686** and install **gdb.x8_64**.
- Users will no longer be able to install the **gdb.i686** packages on 64-bit systems, that is, those with the **libc.so.6()(64-bit)** packages.

(BZ#1853140)

10.12. IDENTITY MANAGEMENT

openssh-ldap has been deprecated

The **openssh-ldap** subpackage has been deprecated in Red Hat Enterprise Linux 8 and will be removed in RHEL 9. As the **openssh-ldap** subpackage is not maintained upstream, Red Hat recommends using SSSD and the **sss_ssh_authorizedkeys** helper, which integrate better with other IdM solutions and are more secure.

By default, the SSSD **ldap** and **ipa** providers read the **sshPublicKey** LDAP attribute of the user object, if available. Note that you cannot use the default SSSD configuration for the **ad** provider or IdM trusted domains to retrieve SSH public keys from Active Directory (AD), since AD does not have a default LDAP attribute to store a public key.

To allow the **sss_ssh_authorizedkeys** helper to get the key from SSSD, enable the **ssh** responder by adding **ssh** to the **services** option in the **sssd.conf** file. See the **sssd.conf(5)** man page for details.

To allow **sshd** to use **sss_ssh_authorizedkeys**, add the **AuthorizedKeysCommand /usr/bin/sss_ssh_authorizedkeys** and **AuthorizedKeysCommandUser nobody** options to the **/etc/ssh/sshd_config** file as described by the **sss_ssh_authorizedkeys(1)** man page.

(BZ#1871025)

DES and 3DES encryption types have been removed

Due to security reasons, the Data Encryption Standard (DES) algorithm has been deprecated and disabled by default since RHEL 7. With the recent rebase of Kerberos packages, single-DES (DES) and triple-DES (3DES) encryption types have been removed from RHEL 8.

If you have configured services or users to only use DES or 3DES encryption, you might experience service interruptions such as:

- Kerberos authentication errors
- **unknown enctype** encryption errors
- Kerberos Distribution Centers (KDCs) with DES-encrypted Database Master Keys (**K/M**) fail to start

Perform the following actions to prepare for the upgrade:

1. Check if your KDC uses DES or 3DES encryption with the **krb5check** open source Python scripts. See krb5check on GitHub.
2. If you are using DES or 3DES encryption with any Kerberos principals, re-key them with a supported encryption type, such as Advanced Encryption Standard (AES). For instructions on re-keying, see Retiring DES from MIT Kerberos Documentation.
3. Test independence from DES and 3DES by temporarily setting the following Kerberos options before upgrading:
    a. In **/var/kerberos/krb5kdc/kdc.conf** on the KDC, set **supported_enctypes** and do not include **des** or **des3**.
    b. For every host, in **/etc/krb5.conf** and any files in **/etc/krb5.conf.d**, set **allow_weak_crypto** to **false**. It is false by default.

c. For every host, in **/etc/krb5.conf** and any files in **/etc/krb5.conf.d**, set **permitted_enctypes**, **default_tgs_enctypes**, and **default_tkt_enctypes**, and do not include **des** or **des3**.

4. If you do not experience any service interruptions with the test Kerberos settings from the previous step, remove them and upgrade. You do not need those settings after upgrading to the latest Kerberos packages.

(BZ#1877991)

Standalone use of the ctdb service has been deprecated

Since RHEL 8.4, customers are advised to use the **ctdb** clustered Samba service only when both of the following conditions apply:

- The **ctdb** service is managed as a **pacemaker** resource with the resource-agent **ctdb**.
- The **ctdb** service uses storage volumes that contain either a GlusterFS file system provided by the Red Hat Gluster Storage product or a GFS2 file system.

The stand-alone use case of the **ctdb** service has been deprecated and will not be included in a next major release of Red Hat Enterprise Linux. For further information on support policies for Samba, see the Knowledgebase article Support Policies for RHEL Resilient Storage - ctdb General Policies .

(BZ#1916296)

Running Samba as a PDC or BDC is deprecated

The classic domain controller mode that enabled administrators to run Samba as an NT4-like primary domain controller (PDC) and backup domain controller (BDC) is deprecated. The code and settings to configure these modes will be removed in a future Samba release.

As long as the Samba version in RHEL 8 provides the PDC and BDC modes, Red Hat supports these modes only in existing installations with Windows versions which support NT4 domains. Red Hat recommends not setting up a new Samba NT4 domain, because Microsoft operating systems later than Windows 7 and Windows Server 2008 R2 do not support NT4 domains.

If you use the PDC to authenticate only Linux users, Red Hat suggests migrating to Red Hat Identity Management (IdM) that is included in RHEL subscriptions. However, you cannot join Windows systems to an IdM domain. Note that Red Hat continues supporting the PDC functionality IdM uses in the background.

Red Hat does not support running Samba as an AD domain controller (DC).

(BZ#1926114)

Indirect AD integration with IdM via WinSync has been deprecated

WinSync is no longer actively developed in RHEL 8 due to several functional limitations:

- WinSync supports only one Active Directory (AD) domain.
- Password synchronization requires installing additional software on AD Domain Controllers.

For a more robust solution with better resource and security separation, Red Hat recommends using a **cross-forest trust** for indirect integration with Active Directory. See the Indirect integration documentation.

(JIRA:RHELPLAN-100400)

The SSSD version of libwbclient has been removed

The SSSD implementation of the **libwbclient** package was deprecated in RHEL 8.4. As it cannot be used with recent versions of Samba, the SSSD implementation of **libwbclient** has now been removed.

(BZ#1947671)

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)

Limited support for FreeRADIUS

In RHEL 8, the following external authentication modules are deprecated as part of the FreeRADIUS offering:

- The MySQL, PostgreSQL, SQlite, and unixODBC database connectors
- The **Perl** language module
- The REST API module

NOTE

The PAM authentication module and other authentication modules that are provided as part of the base package are not affected.

You can find replacements for the deprecated modules in community-supported packages, for example in the Fedora project.

In addition, the scope of support for the **freeradius** package will be limited to the following use cases in future RHEL releases:

- Using FreeRADIUS as a wireless-authentication provider with Identity Management (IdM) as the backend source of authentication. The authentication occurs through the **krb5** and LDAP authentication packages or as PAM authentication in the main FreeRADIUS package.
- Using FreeRADIUS to provide a source-of-truth for authentication in IdM, through the Python 3 authentication package.

In contrast to these deprecations, Red Hat will strengthen the support of the following external authentication modules with FreeRADIUS:

- Authentication based on **krb5** and LDAP
- **Python 3** authentication

The focus on these integration options is in close alignment with the strategic direction of Red Hat IdM.

(Jira:RHELDOCS-17573)

10.13. DESKTOP

The libgnome-keyring library has been deprecated

The **libgnome-keyring** library has been deprecated in favor of the **libsecret** library, as **libgnome-keyring** is not maintained upstream, and does not follow the necessary cryptographic policies for RHEL. The new **libsecret** library is the replacement that follows the necessary security standards.

(BZ#1607766)

10.14. GRAPHICS INFRASTRUCTURES

AGP graphics cards are no longer supported

Graphics cards using the Accelerated Graphics Port (AGP) bus are not supported in Red Hat Enterprise Linux 8. Use the graphics cards with PCI-Express bus as the recommended replacement.

(BZ#1569610)

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)

10.15. THE WEB CONSOLE

The web console no longer supports incomplete translations

The RHEL web console no longer provides translations for languages that have translations available for less than 50 % of the Console's translatable strings. If the browser requests translation to such a language, the user interface will be in English instead.

(BZ#1666722)

10.16. 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#2021685)

Ansible Engine has been deprecated

Previous versions of RHEL 8 provided access to an Ansible Engine repository, with a limited scope of support, to enable supported RHEL Automation use cases, such as RHEL System Roles and Insights remedations. Ansible Engine has been deprecated, and Ansible Engine 2.9 will have no support after September 29, 2023. For more details on the supported use cases, see Scope of support for the Ansible Core package included in the RHEL 9 AppStream.

Users must manually migrate their systems from Ansible Engine to Ansible Core. For that, follow the steps:

Procedure

1. Check if the system is running RHEL 8.7:

```
# cat /etc/redhat-release
```

2. Uninstall Ansible Engine 2.9:

```
# yum remove ansible
```

3. Disable the **ansible-2-for-rhel-8-x86_64-rpms** repository:

```
# subscription-manager repos --disable
ansible-2-for-rhel-8-x86_64-rpms
```

4. Install the Ansible Core package from the RHEL 8 AppStream repository:

```
# yum install ansible-core
```

For more details, see: Using Ansible in RHEL 8.6 and later .

(BZ#2006081)

The geoipupdate package has been deprecated

The **geoipupdate** package requires a third-party subscription and it also downloads proprietary content. Therefore, the **geoipupdate** package has been deprecated, and will be removed in the next major RHEL version.

(BZ#1874892)

10.17. VIRTUALIZATION

virsh iface-* commands have become deprecated

The **virsh iface-*** commands, such as **virsh iface-start** and **virsh iface-destroy**, are now deprecated, and will be removed in a future major version of RHEL. In addition, these commands frequently fail due to configuration dependencies.

Therefore, it is recommended not to use **virsh iface-*** commands for configuring and managing host network connections. Instead, use the NetworkManager program and its related management applications, such as **nmcli**.

(BZ#1664592)

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)

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.

(BZ#1686057)

The Cirrus VGA virtual GPU type has been deprecated

With a future major update of Red Hat Enterprise Linux, the **Cirrus VGA** GPU device will no longer be supported in KVM virtual machines. Therefore, Red Hat recommends using the **stdvga**, **virtio-vga**, or **qxl** devices instead of **Cirrus VGA**.

(BZ#1651994)

KVM on IBM POWER has been deprecated

Using KVM virtualization on IBM POWER hardware has become deprecated. As a result, KVM on IBM POWER is still supported in RHEL 8, but will become unsupported in a future major release of RHEL.

(JIRA:RHELPLAN-71200)

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)

Using SPICE to attach smart card readers to virtual machines has been deprecated

The SPICE remote display protocol has been deprecated in RHEL 8. Since the only recommended way to attach smart card readers to virtual machines (VMs) depends on the SPICE protocol, the usage of smart cards in VMs has also become deprecated in RHEL 8.

In a future major version of RHEL, the functionality of attaching smart card readers to VMs will only be supported by third party remote visualization solutions.

(BZ#2059626)

SPICE has been deprecated

The SPICE remote display protocol has become deprecated. Note that SPICE will remain supported in RHEL 8, but Red Hat recommends using alternate solutions for remote display streaming:

- For remote console access, use the VNC protocol.
- For advanced remote display functions, use third party tools such as RDP, HP RGS, or Mechdyne TGX.

(BZ#1849563)

10.18. CONTAINERS

The Podman varlink-based API v1.0 has been removed

The Podman varlink-based API v1.0 was deprecated in a previous release of RHEL 8. Podman v2.0 introduced a new Podman v2.0 RESTful API. With the release of Podman v3.0, the varlink-based API v1.0 has been completely removed.

(JIRA:RHELPLAN-45858)

container-tools:1.0 has been deprecated

The **container-tools:1.0** module has been deprecated and will no longer receive security updates. It is recommended to use a newer supported stable module stream, such as **container-tools:2.0** or **container-tools:3.0**.

(JIRA:RHELPLAN-59825)

The container-tools:2.0 module has been deprecated

The container-tools:2.0 module has been deprecated and will no longer receive security updates. It is recommended to use a newer supported stable module stream, such as **container-tools:3.0**.

(JIRA:RHELPLAN-85066)

Flatpak images except GIMP has been deprecated

The **rhel8/firefox-flatpak**, **rhel8/thunderbird-flatpak**, **rhel8/inkscape-flatpak**, and **rhel8/libreoffice-flatpak** RHEL 8 Flatpak Applications have been deprecated and replaced by the RHEL 9 versions. The **rhel8/gimp-flatpak** Flatpak Application is not deprecated because there is no replacement yet in RHEL 9.

(BZ#2142499)

10.19. 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 7 and RHEL 8, see Changes to packages in the *Considerations in adopting RHEL 8* document.

The following packages have been deprecated and remain supported until the end of life of RHEL 8:

- 389-ds-base-legacy-tools
- abrt
- abrt-addon-ccpp
- abrt-addon-kerneloops
- abrt-addon-pstoreoops
- abrt-addon-vmcore
- abrt-addon-xorg
- abrt-cli
- abrt-console-notification
- abrt-dbus
- abrt-desktop
- abrt-gui
- abrt-gui-libs
- abrt-libs
- abrt-tui
- adobe-source-sans-pro-fonts
- adwaita-qt
- alsa-plugins-pulseaudio
- amanda
- amanda-client
- amanda-libs
- amanda-server
- ant-contrib
- antlr3
- antlr32
- aopalliance
- apache-commons-collections
- apache-commons-compress
- apache-commons-exec

- apache-commons-jxpath
- apache-commons-parent
- apache-ivy
- apache-parent
- apache-resource-bundles
- apache-sshd
- apiguardian
- aspnetcore-runtime-3.0
- aspnetcore-runtime-3.1
- aspnetcore-runtime-5.0
- aspnetcore-targeting-pack-3.0
- aspnetcore-targeting-pack-3.1
- aspnetcore-targeting-pack-5.0
- assertj-core
- authd
- auto
- autoconf213
- autogen
- autogen-libopts
- awscli
- base64coder
- batik
- batik-css
- batik-util
- bea-stax
- bea-stax-api
- bind-export-devel
- bind-export-libs
- bind-libs-lite

- bind-pkcs11
- bind-pkcs11-devel
- bind-pkcs11-libs
- bind-pkcs11-utils
- bind-sdb
- bind-sdb
- bind-sdb-chroot
- bluez-hid2hci
- boost-jam
- boost-signals
- bouncycastle
- bpg-algeti-fonts
- bpg-chveulebrivi-fonts
- bpg-classic-fonts
- bpg-courier-fonts
- bpg-courier-s-fonts
- bpg-dedaena-block-fonts
- bpg-dejavu-sans-fonts
- bpg-elite-fonts
- bpg-excelsior-caps-fonts
- bpg-excelsior-condenced-fonts
- bpg-excelsior-fonts
- bpg-fonts-common
- bpg-glaho-fonts
- bpg-gorda-fonts
- bpg-ingiri-fonts
- bpg-irubaqidze-fonts
- bpg-mikhail-stephan-fonts
- bpg-mrgvlovani-caps-fonts

- bpg-mrgvlovani-fonts
- bpg-nateli-caps-fonts
- bpg-nateli-condenced-fonts
- bpg-nateli-fonts
- bpg-nino-medium-cond-fonts
- bpg-nino-medium-fonts
- bpg-sans-fonts
- bpg-sans-medium-fonts
- bpg-sans-modern-fonts
- bpg-sans-regular-fonts
- bpg-serif-fonts
- bpg-serif-modern-fonts
- bpg-ucnobi-fonts
- brlapi-java
- bsh
- buildnumber-maven-plugin
- byaccj
- cal10n
- cbi-plugins
- cdparanoia
- cdparanoia-devel
- cdparanoia-libs
- cdrdao
- cmirror
- codehaus-parent
- codemodel
- compat-exiv2-026
- compat-guile18
- compat-hwloc1

- compat-libpthread-nonshared
- compat-libtiff3
- compat-openssl10
- compat-sap-c++-11
- compat-sap-c++-10
- compat-sap-c++-9
- createrepo_c-devel
- ctags
- ctags-etags
- custodia
- cyrus-imapd-vzic
- dbus-c++
- dbus-c++-devel
- dbus-c++-glib
- dbxtool
- dhcp-libs
- directory-maven-plugin
- directory-maven-plugin-javadoc
- dirsplit
- dleyna-connector-dbus
- dleyna-core
- dleyna-renderer
- dleyna-server
- dnssec-trigger
- dnssec-trigger-panel
- dotnet-apphost-pack-3.0
- dotnet-apphost-pack-3.1
- dotnet-apphost-pack-5.0
- dotnet-host-fxr-2.1

- dotnet-host-fxr-2.1
- dotnet-hostfxr-3.0
- dotnet-hostfxr-3.1
- dotnet-hostfxr-5.0
- dotnet-runtime-2.1
- dotnet-runtime-3.0
- dotnet-runtime-3.1
- dotnet-runtime-5.0
- dotnet-sdk-2.1
- dotnet-sdk-2.1.5xx
- dotnet-sdk-3.0
- dotnet-sdk-3.1
- dotnet-sdk-5.0
- dotnet-targeting-pack-3.0
- dotnet-targeting-pack-3.1
- dotnet-targeting-pack-5.0
- dotnet-templates-3.0
- dotnet-templates-3.1
- dotnet-templates-5.0
- dotnet5.0-build-reference-packages
- dptfxtract
- drpm
- drpm-devel
- dump
- dvd+rw-tools
- dyninst-static
- eclipse-ecf
- eclipse-ecf-core
- eclipse-ecf-runtime

- eclipse-emf
- eclipse-emf-core
- eclipse-emf-runtime
- eclipse-emf-xsd
- eclipse-equinox-osgi
- eclipse-jdt
- eclipse-license
- eclipse-p2-discovery
- eclipse-pde
- eclipse-platform
- eclipse-swt
- ed25519-java
- ee4j-parent
- elfutils-devel-static
- elfutils-libelf-devel-static
- enca
- enca-devel
- environment-modules-compat
- evince-browser-plugin
- exec-maven-plugin
- farstream02
- felix-gogo-command
- felix-gogo-runtime
- felix-gogo-shell
- felix-scr
- felix-osgi-compendium
- felix-osgi-core
- felix-osgi-foundation
- felix-parent

- file-roller
- fipscheck
- fipscheck-devel
- fipscheck-lib
- firewire
- fonts-tweak-tool
- forge-parent
- freeradius-mysql
- freeradius-perl
- freeradius-postgresql
- freeradius-rest
- freeradius-sqlite
- freeradius-unixODBC
- fuse-sshfs
- fusesource-pom
- future
- gamin
- gamin-devel
- gavl
- gcc-toolset-10
- gcc-toolset-10-annobin
- gcc-toolset-10-binutils
- gcc-toolset-10-binutils-devel
- gcc-toolset-10-build
- gcc-toolset-10-dwz
- gcc-toolset-10-dyninst
- gcc-toolset-10-dyninst-devel
- gcc-toolset-10-elfutils
- gcc-toolset-10-elfutils-debuginfod-client

- gcc-toolset-10-elfutils-debuginfod-client-devel
- gcc-toolset-10-elfutils-devel
- gcc-toolset-10-elfutils-libelf
- gcc-toolset-10-elfutils-libelf-devel
- gcc-toolset-10-elfutils-libs
- gcc-toolset-10-gcc
- gcc-toolset-10-gcc-c++
- gcc-toolset-10-gcc-gdb-plugin
- gcc-toolset-10-gcc-gfortran
- gcc-toolset-10-gdb
- gcc-toolset-10-gdb-doc
- gcc-toolset-10-gdb-gdbserver
- gcc-toolset-10-libasan-devel
- gcc-toolset-10-libatomic-devel
- gcc-toolset-10-libitm-devel
- gcc-toolset-10-liblsan-devel
- gcc-toolset-10-libquadmath-devel
- gcc-toolset-10-libstdc++-devel
- gcc-toolset-10-libstdc++-docs
- gcc-toolset-10-libtsan-devel
- gcc-toolset-10-libubsan-devel
- gcc-toolset-10-ltrace
- gcc-toolset-10-make
- gcc-toolset-10-make-devel
- gcc-toolset-10-perftools
- gcc-toolset-10-runtime
- gcc-toolset-10-strace
- gcc-toolset-10-systemtap
- gcc-toolset-10-systemtap-client

- gcc-toolset-10-systemtap-devel
- gcc-toolset-10-systemtap-initscript
- gcc-toolset-10-systemtap-runtime
- gcc-toolset-10-systemtap-sdt-devel
- gcc-toolset-10-systemtap-server
- gcc-toolset-10-toolchain
- gcc-toolset-10-valgrind
- gcc-toolset-10-valgrind-devel
- gcc-toolset-9
- gcc-toolset-9-annobin
- gcc-toolset-9-build
- gcc-toolset-9-perftools
- gcc-toolset-9-runtime
- gcc-toolset-9-toolchain
- gcc-toolset-11-make-devel
- GConf2
- GConf2-devel
- gegl
- genisoimage
- genwqe-tools
- genwqe-vpd
- genwqe-zlib
- genwqe-zlib-devel
- geoipupdate
- geronimo-annotation
- geronimo-jms
- geronimo-jpa
- geronimo-parent-poms
- gfbgraph

- gflags
- gflags-devel
- glassfish-annotation-api
- glassfish-el
- glassfish-fastinfoset
- glassfish-jaxb-core
- glassfish-jaxb-txw2
- glassfish-jsp
- glassfish-jsp-api
- glassfish-legal
- glassfish-master-pom
- glassfish-servlet-api
- glew-devel
- glib2-fam
- glog
- glog-devel
- gmock
- gmock-devel
- gnome-abrt
- gnome-boxes
- gnome-menus-devel
- gnome-online-miners
- gnome-shell-extension-disable-screenshield
- gnome-shell-extension-horizontal-workspaces
- gnome-shell-extension-no-hot-corner
- gnome-shell-extension-window-grouper
- gnome-themes-standard
- gnu-free-fonts-common
- gnu-free-mono-fonts

- gnu-free-sans-fonts
- gnu-free-serif-fonts
- gnupg2-smime
- gnuplot
- gnuplot-common
- gobject-introspection-devel
- google-gson
- google-noto-sans-syriac-eastern-fonts
- google-noto-sans-syriac-estrangela-fonts
- google-noto-sans-syriac-western-fonts
- google-noto-sans-tibetan-fonts
- google-noto-sans-ui-fonts
- gphoto2
- gsl-devel
- gssntlmssp
- gtest
- gtest-devel
- gtkmm24
- gtkmm24-devel
- gtkmm24-docs
- gtksourceview3
- gtksourceview3-devel
- gtkspell
- gtkspell-devel
- gtkspell3
- guile
- gutenprint-gimp
- gutenprint-libs-ui
- gvfs-afc

- gvfs-afp
- gvfs-archive
- hamcrest-core
- hawtjni
- hawtjni
- hawtjni-runtime
- HdrHistogram
- HdrHistogram-javadoc
- highlight-gui
- hivex-devel
- hostname
- hplip-gui
- httpcomponents-project
- hwloc-plugins
- hyphen-fo
- hyphen-grc
- hyphen-hsb
- hyphen-ia
- hyphen-is
- hyphen-ku
- hyphen-mi
- hyphen-mn
- hyphen-sa
- hyphen-tk
- ibus-sayura
- icedax
- icu4j
- idm-console-framework
- iptables

- ipython
- isl
- isl-devel
- isorelax
- istack-commons-runtime
- istack-commons-tools
- iwl3945-firmware
- iwl4965-firmware
- iwl6000-firmware
- jacoco
- jaf
- jaf-javadoc
- jakarta-oro
- janino
- jansi-native
- jarjar
- java-1.8.0-ibm
- java-1.8.0-ibm-demo
- java-1.8.0-ibm-devel
- java-1.8.0-ibm-headless
- java-1.8.0-ibm-jdbc
- java-1.8.0-ibm-plugin
- java-1.8.0-ibm-src
- java-1.8.0-ibm-webstart
- java-1.8.0-openjdk-accessibility
- java-1.8.0-openjdk-accessibility-slowdebug
- java_cup
- java-atk-wrapper
- javacc

- javacc-maven-plugin
- javaewah
- javaparser
- javapoet
- javassist
- javassist-javadoc
- jaxen
- jboss-annotations-1.2-api
- jboss-interceptors-1.2-api
- jboss-logmanager
- jboss-parent
- jctools
- jdepend
- jdependency
- jdom
- jdom2
- jetty
- jetty-continuation
- jetty-http
- jetty-io
- jetty-security
- jetty-server
- jetty-servlet
- jetty-util
- jffi
- jflex
- jgit
- jline
- jmc

- jnr-netdb
- jolokia-jvm-agent
- js-uglify
- jsch
- json_simple
- jss-javadoc
- jtidy
- junit5
- jvnet-parent
- jzlib
- kernel-cross-headers
- ksc
- kurdit-unikurd-web-fonts
- kyotocabinet-libs
- ldapjdk-javadoc
- lensfun
- lensfun-devel
- lftp-scripts
- libaec
- libaec-devel
- libappindicator-gtk3
- libappindicator-gtk3-devel
- libatomic-static
- libavc1394
- libblocksruntime
- libcacard
- libcacard-devel
- libcgroup
- libcgroup-tools

- libchamplain
- libchamplain-devel
- libchamplain-gtk
- libcroco
- libcroco-devel
- libcxl
- libcxl-devel
- libdap
- libdap-devel
- libdazzle-devel
- libdbusmenu
- libdbusmenu-devel
- libdbusmenu-doc
- libdbusmenu-gtk3
- libdbusmenu-gtk3-devel
- libdc1394
- libdnet
- libdnet-devel
- libdv
- libdwarf
- libdwarf-devel
- libdwarf-static
- libdwarf-tools
- libeasyfc
- libeasyfc-gobject
- libepubgen-devel
- libertas-sd8686-firmware
- libertas-usb8388-firmware
- libertas-usb8388-olpc-firmware

- libgdither
- libGLEW
- libgovirt
- libguestfs-benchmarking
- libguestfs-devel
- libguestfs-gfs2
- libguestfs-gobject
- libguestfs-gobject-devel
- libguestfs-java
- libguestfs-java-devel
- libguestfs-javadoc
- libguestfs-man-pages-ja
- libguestfs-man-pages-uk
- libguestfs-tools
- libguestfs-tools-c
- libhugetlbfs
- libhugetlbfs-devel
- libhugetlbfs-utils
- libIDL
- libIDL-devel
- libidn
- libiec61883
- libindicator-gtk3
- libindicator-gtk3-devel
- libiscsi-devel
- libjose-devel
- libkkc
- libkkc-common
- libkkc-data

- libldb-devel
- liblogging
- libluksmeta-devel
- libmalaga
- libmcpp
- libmemcached
- libmemcached-libs
- libmetalink
- libmodulemd1
- libmongocrypt
- libmtp-devel
- libmusicbrainz5
- libmusicbrainz5-devel
- libnbd-devel
- liboauth
- liboauth-devel
- libpfm-static
- libpng12
- libpurple
- libpurple-devel
- libraw1394
- libreport-plugin-mailx
- libreport-plugin-rhtsupport
- libreport-plugin-ureport
- libreport-rhel
- libreport-rhel-bugzilla
- librpmem
- librpmem-debug
- librpmem-devel

- libsass
- libsass-devel
- libselinux-python
- libsqlite3x
- libtalloc-devel
- libtar
- libtdb-devel
- libtevent-devel
- libtpms-devel
- libunwind
- libusal
- libvarlink
- libverto-libevent
- libvirt-admin
- libvirt-bash-completion
- libvirt-daemon-driver-storage-gluster
- libvirt-daemon-driver-storage-iscsi-direct
- libvirt-devel
- libvirt-docs
- libvirt-gconfig
- libvirt-gobject
- libvirt-lock-sanlock
- libvirt-wireshark
- libvmem
- libvmem-debug
- libvmem-devel
- libvmmalloc
- libvmmalloc-debug
- libvmmalloc-devel

- libvncserver
- libwinpr-devel
- libwmf
- libwmf-devel
- libwmf-lite
- libXNVCtrl
- libyami
- log4j12
- log4j12-javadoc
- lohit-malayalam-fonts
- lohit-nepali-fonts
- lorax-composer
- lua-guestfs
- lucene
- lucene-analysis
- lucene-analyzers-smartcn
- lucene-queries
- lucene-queryparser
- lucene-sandbox
- lz4-java
- lz4-java-javadoc
- mailman
- mailx
- make-devel
- malaga
- malaga-suomi-voikko
- marisa
- maven-antrun-plugin
- maven-assembly-plugin

- maven-clean-plugin
- maven-dependency-analyzer
- maven-dependency-plugin
- maven-doxia
- maven-doxia-sitetools
- maven-install-plugin
- maven-invoker
- maven-invoker-plugin
- maven-parent
- maven-plugins-pom
- maven-reporting-api
- maven-reporting-impl
- maven-resolver-api
- maven-resolver-connector-basic
- maven-resolver-impl
- maven-resolver-spi
- maven-resolver-transport-wagon
- maven-resolver-util
- maven-scm
- maven-script-interpreter
- maven-shade-plugin
- maven-shared
- maven-verifier
- maven-wagon-file
- maven-wagon-http
- maven-wagon-http-shared
- maven-wagon-provider-api
- maven2
- meanwhile

- mercurial
- mercurial-hgk
- metis
- metis-devel
- mingw32-bzip2
- mingw32-bzip2-static
- mingw32-cairo
- mingw32-expat
- mingw32-fontconfig
- mingw32-freetype
- mingw32-freetype-static
- mingw32-gstreamer1
- mingw32-harfbuzz
- mingw32-harfbuzz-static
- mingw32-icu
- mingw32-libjpeg-turbo
- mingw32-libjpeg-turbo-static
- mingw32-libpng
- mingw32-libpng-static
- mingw32-libtiff
- mingw32-libtiff-static
- mingw32-openssl
- mingw32-readline
- mingw32-sqlite
- mingw32-sqlite-static
- mingw64-adwaita-icon-theme
- mingw64-bzip2
- mingw64-bzip2-static
- mingw64-cairo

- mingw64-expat
- mingw64-fontconfig
- mingw64-freetype
- mingw64-freetype-static
- mingw64-gstreamer1
- mingw64-harfbuzz
- mingw64-harfbuzz-static
- mingw64-icu
- mingw64-libjpeg-turbo
- mingw64-libjpeg-turbo-static
- mingw64-libpng
- mingw64-libpng-static
- mingw64-libtiff
- mingw64-libtiff-static
- mingw64-nettle
- mingw64-openssl
- mingw64-readline
- mingw64-sqlite
- mingw64-sqlite-static
- modello
- mojo-parent
- mongo-c-driver
- mousetweaks
- mozjs52
- mozjs52-devel
- mozjs60
- mozjs60-devel
- mozvoikko
- msv-javadoc

- msv-manual
- munge-maven-plugin
- mythes-mi
- mythes-ne
- nafees-web-naskh-fonts
- nbd
- nbdkit-devel
- nbdkit-example-plugins
- nbdkit-gzip-plugin
- nbdkit-plugin-python-common
- nbdkit-plugin-vddk
- ncompress
- ncurses-compat-libs
- net-tools
- netcf
- netcf-devel
- netcf-libs
- network-scripts
- network-scripts-ppp
- nkf
- nodejs-devel
- nodejs-packaging
- nss_nis
- nss-pam-ldapd
- objectweb-asm
- objectweb-asm-javadoc
- objectweb-pom
- ocaml-bisect-ppx
- ocaml-camlp4

- ocaml-camlp4-devel
- ocaml-lwt
- ocaml-mmap
- ocaml-ocplib-endian
- ocaml-ounit
- ocaml-result
- ocaml-seq
- opencryptoki-tpmtok
- opencv-contrib
- opencv-core
- opencv-devel
- openhpi
- openhpi-libs
- OpenIPMI-perl
- openssh-cavs
- openssh-ldap
- openssl-ibmpkcs11
- opentest4j
- os-maven-plugin
- pakchois
- pandoc
- paps-libs
- paranamer
- parfait
- parfait-examples
- parfait-javadoc
- pcp-parfait-agent
- pcp-pmda-rpm
- pcp-pmda-vmware

- pcsc-lite-doc
- peripety
- perl-B-Debug
- perl-B-Lint
- perl-Class-Factory-Util
- perl-Class-ISA
- perl-DateTime-Format-HTTP
- perl-DateTime-Format-Mail
- perl-File-CheckTree
- perl-homedir
- perl-libxml-perl
- perl-Locale-Codes
- perl-Mozilla-LDAP
- perl-NKF
- perl-Object-HashBase-tools
- perl-Package-DeprecationManager
- perl-Pod-LaTeX
- perl-Pod-Plainer
- perl-prefork
- perl-String-CRC32
- perl-SUPER
- perl-Sys-Virt
- perl-tests
- perl-YAML-Syck
- phodav
- php-recode
- php-xmlrpc
- pidgin
- pidgin-devel

- pidgin-sipe
- pinentry-emacs
- pinentry-gtk
- pipewire0.2-devel
- pipewire0.2-libs
- platform-python-coverage
- plexus-ant-factory
- plexus-bsh-factory
- plexus-cli
- plexus-component-api
- plexus-component-factories-pom
- plexus-components-pom
- plexus-i18n
- plexus-interactivity
- plexus-pom
- plexus-velocity
- plymouth-plugin-throbgress
- pmreorder
- postgresql-test-rpm-macros
- powermock
- prometheus-jmx-exporter
- prometheus-jmx-exporter-openjdk11
- ptscotch-mpich
- ptscotch-mpich-devel
- ptscotch-mpich-devel-parmetis
- ptscotch-openmpi
- ptscotch-openmpi-devel
- purple-sipe
- pygobject2-doc

- pygtk2
- pygtk2-codegen
- pygtk2-devel
- pygtk2-doc
- python-nose-docs
- python-nss-doc
- python-podman-api
- python-psycopg2-doc
- python-pymongo-doc
- python-redis
- python-schedutils
- python-slip
- python-sqlalchemy-doc
- python-varlink
- python-virtualenv-doc
- python2-backports
- python2-backports-ssl_match_hostname
- python2-bson
- python2-coverage
- python2-docs
- python2-docs-info
- python2-funcsigs
- python2-ipaddress
- python2-mock
- python2-nose
- python2-numpy-doc
- python2-psycopg2-debug
- python2-psycopg2-tests
- python2-pymongo

- python2-pymongo-gridfs
- python2-pytest-mock
- python2-sqlalchemy
- python2-tools
- python2-virtualenv
- python3-bson
- python3-click
- python3-coverage
- python3-cpio
- python3-custodia
- python3-docs
- python3-flask
- python3-gevent
- python3-gobject-base
- python3-hivex
- python3-html5lib
- python3-hypothesis
- python3-ipatests
- python3-itsdangerous
- python3-jwt
- python3-libguestfs
- python3-mock
- python3-networkx-core
- python3-nose
- python3-nss
- python3-openipmi
- python3-pillow
- python3-ptyprocess
- python3-pydbus

- python3-pymongo
- python3-pymongo-gridfs
- python3-pyOpenSSL
- python3-pytoml
- python3-reportlab
- python3-schedutils
- python3-scons
- python3-semantic_version
- python3-slip
- python3-slip-dbus
- python3-sqlalchemy
- python3-syspurpose
- python3-virtualenv
- python3-webencodings
- python3-werkzeug
- python38-asn1crypto
- python38-numpy-doc
- python38-psycopg2-doc
- python38-psycopg2-tests
- python39-numpy-doc
- python39-psycopg2-doc
- python39-psycopg2-tests
- qemu-kvm-block-gluster
- qemu-kvm-block-iscsi
- qemu-kvm-block-ssh
- qemu-kvm-hw-usbredir
- qemu-kvm-tests
- qpdf
- qpdf-doc

- qpid-proton
- qrencode
- qrencode-devel
- qrencode-libs
- qt5-qtcanvas3d
- qt5-qtcanvas3d-examples
- rarian
- rarian-compat
- re2c
- recode
- redhat-lsb
- redhat-lsb-core
- redhat-lsb-cxx
- redhat-lsb-desktop
- redhat-lsb-languages
- redhat-lsb-printing
- redhat-lsb-submod-multimedia
- redhat-lsb-submod-security
- redhat-lsb-supplemental
- redhat-lsb-trialuse
- redhat-menus
- redhat-support-lib-python
- redhat-support-tool
- reflections
- regexp
- relaxngDatatype
- rhsm-gtk
- rpm-plugin-prioreset
- rpmemd

- rsyslog-udpspoof
- ruby-hivex
- ruby-libguestfs
- rubygem-abrt
- rubygem-abrt-doc
- rubygem-bson
- rubygem-bson-doc
- rubygem-bundler-doc
- rubygem-mongo
- rubygem-mongo-doc
- rubygem-net-telnet
- rubygem-xmlrpc
- s390utils-cmsfs
- samba-pidl
- samba-test
- samba-test-libs
- samyak-devanagari-fonts
- samyak-fonts-common
- samyak-gujarati-fonts
- samyak-malayalam-fonts
- samyak-odia-fonts
- samyak-tamil-fonts
- sane-frontends
- sanlk-reset
- sat4j
- scala
- scotch
- scotch-devel
- SDL_sound

- selinux-policy-minimum
- sendmail
- sgabios
- sgabios-bin
- shrinkwrap
- sisu-inject
- sisu-mojos
- sisu-plexus
- skkdic
- SLOF
- smc-anjalioldlipi-fonts
- smc-dyuthi-fonts
- smc-fonts-common
- smc-kalyani-fonts
- smc-raghumalayalam-fonts
- smc-suruma-fonts
- softhsm-devel
- sonatype-oss-parent
- sonatype-plugins-parent
- sos-collector
- sparsehash-devel
- spax
- spec-version-maven-plugin
- spice
- spice-client-win-x64
- spice-client-win-x86
- spice-glib
- spice-glib-devel
- spice-gtk

- spice-gtk-tools
- spice-gtk3
- spice-gtk3-devel
- spice-gtk3-vala
- spice-parent
- spice-protocol
- spice-qxl-wddm-dod
- spice-server
- spice-server-devel
- spice-qxl-xddm
- spice-server
- spice-streaming-agent
- spice-vdagent-win-x64
- spice-vdagent-win-x86
- sssd-libwbclient
- star
- stax-ex
- stax2-api
- stringtemplate
- stringtemplate4
- subscription-manager-initial-setup-addon
- subscription-manager-migration
- subscription-manager-migration-data
- subversion-javahl
- SuperLU
- SuperLU-devel
- supermin-devel
- swig
- swig-doc

- swig-gdb
- swtpm-devel
- swtpm-tools-pkcs11
- system-storage-manager
- tcl-brlapi
- testng
- tibetan-machine-uni-fonts
- timedatex
- tpm-quote-tools
- tpm-tools
- tpm-tools-pkcs11
- treelayout
- trousers
- trousers-lib
- tuned-profiles-compat
- tuned-profiles-nfv-host-bin
- tuned-utils-systemtap
- tycho
- uglify-js
- unbound-devel
- univocity-output-tester
- univocity-parsers
- usbguard-notifier
- usbredir-devel
- utf8cpp
- uthash
- velocity
- vinagre
- vino

- virt-dib
- virt-p2v-maker
- vm-dump-metrics-devel
- weld-parent
- wodim
- woodstox-core
- wqy-microhei-fonts
- wqy-unibit-fonts
- xdelta
- xmlgraphics-commons
- xmlstreambuffer
- xinetd
- xorg-x11-apps
- xorg-x11-drv-qxl
- xorg-x11-server-Xspice
- xpp3
- xsane-gimp
- xsom
- xz-java
- xz-java-javadoc
- yajl-devel
- yp-tools
- ypbind
- ypserv

10.20. DEPRECATED AND UNMAINTAINED DEVICES

This section lists devices (drivers, adapters) that

- continue to be supported until the end of life of RHEL 8 but will likely not be supported in future major releases of this product and are not recommended for new deployments. Support for devices other than those listed remains unchanged. These are **deprecated** devices.
- are available but are no longer being tested or updated on a routine basis in RHEL 8. Red Hat

may fix serious bugs, including security bugs, at its discretion. These devices should no longer be used in production, and it is likely they will be disabled in the next major release. These are **unmaintained** devices.

PCI device IDs are in the format of *vendor:device:subvendor:subdevice*. If no device ID is listed, all devices associated with the corresponding driver have been deprecated. To check the PCI IDs of the hardware on your system, run the **lspci -nn** command.

Table 10.1. Deprecated devices

Device ID	Driver	Device name
	bnx2	QLogic BCM5706/5708/5709/5716 Driver
	hpsa	Hewlett-Packard Company: Smart Array Controllers
0x10df:0x0724	lpfc	Emulex Corporation: OneConnect FCoE Initiator (Skyhawk)
0x10df:0xe200	lpfc	Emulex Corporation: LPe15000/LPe16000 Series 8Gb/16Gb Fibre Channel Adapter
0x10df:0xf011	lpfc	Emulex Corporation: Saturn: LightPulse Fibre Channel Host Adapter
0x10df:0xf015	lpfc	Emulex Corporation: Saturn: LightPulse Fibre Channel Host Adapter
0x10df:0xf100	lpfc	Emulex Corporation: LPe12000 Series 8Gb Fibre Channel Adapter
0x10df:0xfc40	lpfc	Emulex Corporation: Saturn-X: LightPulse Fibre Channel Host Adapter
0x10df:0xe220	be2net	Emulex Corporation: OneConnect NIC (Lancer)
0x1000:0x005b	megaraid_sas	Broadcom / LSI: MegaRAID SAS 2208 [Thunderbolt]
0x1000:0x006E	mpt3sas	Broadcom / LSI: SAS2308 PCI-Express Fusion-MPT SAS-2
0x1000:0x0080	mpt3sas	Broadcom / LSI: SAS2208 PCI-Express Fusion-MPT SAS-2
0x1000:0x0081	mpt3sas	Broadcom / LSI: SAS2208 PCI-Express Fusion-MPT SAS-2
0x1000:0x0082	mpt3sas	Broadcom / LSI: SAS2208 PCI-Express Fusion-MPT SAS-2
0x1000:0x0083	mpt3sas	Broadcom / LSI: SAS2208 PCI-Express Fusion-MPT SAS-2
0x1000:0x0084	mpt3sas	Broadcom / LSI: SAS2208 PCI-Express Fusion-MPT SAS-2

Device ID	Driver	Device name
0x1000:0x0085	mpt3sas	Broadcom / LSI: SAS2208 PCI-Express Fusion-MPT SAS-2
0x1000:0x0086	mpt3sas	Broadcom / LSI: SAS2308 PCI-Express Fusion-MPT SAS-2
0x1000:0x0087	mpt3sas	Broadcom / LSI: SAS2308 PCI-Express Fusion-MPT SAS-2
	myri10ge	Myricom 10G driver (10GbE)
	netxen_nic	QLogic/NetXen (1/10) GbE Intelligent Ethernet Driver
0x1077:0x2031	qla2xxx	QLogic Corp.: ISP8324-based 16Gb Fibre Channel to PCI Express Adapter
0x1077:0x2532	qla2xxx	QLogic Corp.: ISP2532-based 8Gb Fibre Channel to PCI Express HBA
0x1077:0x8031	qla2xxx	QLogic Corp.: 8300 Series 10GbE Converged Network Adapter (FCoE)
	qla3xxx	QLogic ISP3XXX Network Driver v2.03.00-k5
0x1924:0x0803	sfc	Solarflare Communications: SFC9020 10G Ethernet Controller
0x1924:0x0813	sfc	Solarflare Communications: SFL9021 10GBASE-T Ethernet Controller
	Soft-RoCE (rdma_rxe)	
	HNS-RoCE	HNS GE/10GE/25GE/50GE/100GE RDMA Network Controller
	liquidio	Cavium LiquidIO Intelligent Server Adapter Driver
	liquidio_vf	Cavium LiquidIO Intelligent Server Adapter Virtual Function Driver

Table 10.2. Unmaintained devices

Device ID	Driver	Device name
	e1000	Intel® PRO/1000 Network Driver
	mptbase	Fusion MPT SAS Host driver
	mptsas	Fusion MPT SAS Host driver
	mptscsih	Fusion MPT SCSI Host driver
	mptspi	Fusion MPT SAS Host driver
0x1000:0x0071 [a]	megaraid_sas	Broadcom / LSI: MR SAS HBA 2004
0x1000:0x0073 [a]	megaraid_sas	Broadcom / LSI: MegaRAID SAS 2008 [Falcon]
0x1000:0x0079 [a]	megaraid_sas	Broadcom / LSI: MegaRAID SAS 2108 [Liberator]
	nvmet_tcp	NVMe/TCP target driver

[a] Disabled in RHEL 8.0, re-enabled in RHEL 8.4 due to customer requests.

CHAPTER 11. KNOWN ISSUES

This part describes known issues in Red Hat Enterprise Linux 8.7.

11.1. INSTALLER AND IMAGE CREATION

Installation fails on IBM Power 10 systems with LPAR and secure boot enabled

RHEL installer is not integrated with static key secure boot on IBM Power 10 systems. Consequently, when logical partition (LPAR) is enabled with the secure boot option, the installation fails with the error, **Unable to proceed with RHEL-x.x Installation**.

To work around this problem, install RHEL without enabling secure boot. After booting the system:

1. Copy the signed Kernel into the PReP partition using the **dd** command.
2. Restart the system and enable secure boot.

Once the firmware verifies the bootloader and the kernel, the system boots up successfully.

For more information, see https://www.ibm.com/support/pages/node/6528884

(BZ#2025814)

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 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)

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)

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)

Network access is not enabled by default in the installation program

Several installation features require network access, for example, registration of a system using the Content Delivery Network (CDN), NTP server support, and network installation sources. However, network access is not enabled by default, and as a result, these features cannot be used until network access is enabled.

To work around this problem, add **ip=dhcp** to boot options to enable network access when the installation starts. Optionally, passing a Kickstart file or a repository located on the network using boot options also resolves the problem. As a result, the network-based installation features can be used.

(BZ#1757877)

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)

IBM Power systems with HASH MMU mode fail to boot with memory allocation failures

IBM Power Systems with **HASH memory allocation unit (MMU)** mode support **kdump** up to a maximum of 192 cores. Consequently, the system fails to boot with memory allocation failures if **kdump** is enabled on more than 192 cores. This limitation is due to RMA memory allocations during early boot in **HASH MMU** mode. To work around this problem, use the **Radix MMU** mode with **fadump** enabled instead of using **kdump**.

(BZ#2028361)

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#2126506)

The --size parameter of composer-cli compose start treats values as bytes instead of MiB

When using the **composer-cli compose start --size *size_value blueprint_name image_type*** command, the **--size** parameter should use its parameter in the MiB format. However, a bug in the settings causes the **composer-cli** tool to treat this parameter as bytes units.

To work around this issue, multiply the size value by 1048576. Alternatively, use the in your blueprint. The customization allows a more granular control over filesystems and accepts units like MiB or GiB. See Supported image customizations.

(BZ#2033192)

11.2. SUBSCRIPTION MANAGEMENT

syspurpose addons have no effect on the subscription-manager attach --auto output.

In Red Hat Enterprise Linux 8, four attributes of the **syspurpose** command-line tool have been added: **role**,**usage**, **service_level_agreement** and **addons**. Currently, only **role**, **usage** and **service_level_agreement** affect the output of running the **subscription-manager attach --auto** command. Users who attempt to set values to the **addons** argument will not observe any effect on the subscriptions that are auto-attached.

(BZ#1687900)

11.3. SOFTWARE MANAGEMENT

cr_compress_file_with_stat() can cause a memory leak

The **createrepo_c** C library has the API **cr_compress_file_with_stat()** function. This function is declared with **char **dst** as a second parameter. Depending on its other parameters, **cr_compress_file_with_stat()** either uses **dst** as an input parameter, or uses it to return an allocated string. This unpredictable behavior can cause a memory leak, because it does not inform the user when to free **dst** contents.

To work around this problem, a new API **cr_compress_file_with_stat_v2** function has been added, which uses the **dst** parameter only as an input. It is declared as **char *dst**. This prevents memory leak.

Note that the **cr_compress_file_with_stat_v2** function is temporary and will be present only in RHEL 8. Later, **cr_compress_file_with_stat()** will be fixed instead.

(BZ#1973588)

YUM transactions reported as successful when a scriptlet fails

Since RPM version 4.6, post-install scriptlets are allowed to fail without being fatal to the transaction. This behavior propagates up to YUM as well. This results in scriptlets which might occasionally fail while the overall package transaction reports as successful.

There is no workaround available at the moment.

Note that this is expected behavior that remains consistent between RPM and YUM. Any issues in scriptlets should be addressed at the package level.

(BZ#1986657)

A security YUM upgrade fails for packages that change their architecture through the upgrade

The patch for BZ#2088149, released with the RHBA-2022:7711 advisory, introduced the following regression: The YUM 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#2088149)

11.4. SHELLS AND COMMAND-LINE TOOLS

ipmitool is incompatible with certain server platforms

The **ipmitool** utility serves for monitoring, configuring, and managing devices that support the Intelligent Platform Management Interface (IPMI). The current version of **ipmitool** uses Cipher Suite 17 by default instead of the previous Cipher Suite 3. Consequently, **ipmitool** fails to communicate with certain bare metal nodes that announced support for Cipher Suite 17 during negotiation, but do not actually support this cipher suite. As a result, **ipmitool** aborts with the **no matching cipher suite** error message.

For more details, see the related Knowledgebase article.

To solve this problem, update your baseboard management controller (BMC) firmware to use the Cipher Suite 17.

Optionally, if the BMC firmware update is not available, you can work around this problem by forcing **ipmitool** to use a certain cipher suite. When invoking a managing task with **ipmitool**, add the **-C** option to the **ipmitool** command together with the *number* of the cipher suite you want to use. See the following example:

```
# ipmitool -I lanplus -H _myserver.example.com_ -P _mypass_ -C 3 chassis power status
```

(BZ#1873614)

ReaR fails to recreate a volume group when you do not use clean disks for restoring

ReaR fails to perform recovery when you want to restore to disks that contain existing data.

To work around this problem, wipe the disks manually before restoring to them if they have been previously used. To wipe the disks in the rescue environment, use one of the following commands before running the **rear recover** command:

- The **dd** command to overwrite the disks.
- The **wipefs** command with the **-a** flag to erase all available metadata.

See the following example of wiping metadata from the **/dev/sda** disk:

```
# wipefs -a /dev/sda[1-9] /dev/sda
```

This command wipes the metadata from the partitions on **/dev/sda** first, and then the partition table itself.

(BZ#1925531)

coreutils might report misleading EPERM error codes

GNU Core Utilities (**coreutils**) started using the **statx()** system call. If a **seccomp** filter returns an EPERM error code for unknown system calls, **coreutils** might consequently report misleading EPERM error codes because EPERM can not be distinguished from the actual *Operation not permitted* error returned by a working **statx()** syscall.

To work around this problem, update the **seccomp** filter to either permit the **statx()** syscall, or to return an ENOSYS error code for syscalls it does not know.

(BZ#2030661)

11.5. INFRASTRUCTURE SERVICES

Postfix TLS fingerprint algorithm in the FIPS mode needs to be changed to SHA-256

By default in RHEL 8, **postfix** uses MD5 fingerprints with the TLS for backward compatibility. But in the FIPS mode, the MD5 hashing function is not available, which may cause TLS to incorrectly function in the default postfix configuration. To workaround this problem, the hashing function needs to be changed to SHA-256 in the postfix configuration file.

For more details, see the related Knowledgebase article Fix postfix TLS in the FIPS mode by switching to SHA-256 instead of MD5.

(BZ#1711885)

rsync fails while using the --delete and the --filter '-x *string*.*' option together

The **rsync** utility for transferring and synchronizing files is unable to handle extended attributes in RHEL 8 correctly. Consequently, if you pass the **--delete** option together with the **--filter '-x *string*.*'** option for extended attributes to the **rsync** command, and a file on your system satisfies the regular expression, an error stating protocol incompatibilities occurs. For example, if you use the **--filter '-x system.*'** option, the filter finds the **system.mwmrc** file, which is present on your system, and **rsync** fails. See the following error message that occurs after using the **--filter '-x system.*'** option:

```
# /usr/bin/rsync -a --delete --filter '-x system.*' / 192.0.2.2::some/test/dir/
ERROR: rejecting excluded file-list name: path/to/excluded/system.mwmrc
rsync error: protocol incompatibility (code 2) at flist.c(912) [receiver=3.1.3]
rsync error: protocol incompatibility (code 2) at io.c(1649) [generator=3.1.3])
```

To prevent this problem, use regular expressions for extended attributes with caution.

(BZ#2139118)

The brltty package is not multilib compatible

It is not possible to have both 32-bit and 64-bit versions of the **brltty** package installed. You can either install the 32-bit (**brltty.i686**) or the 64-bit (**brltty.x86_64**) version of the package. The 64-bit version is recommended.

(BZ#2008197)

11.6. SECURITY

File permissions of /etc/passwd- are not aligned with the CIS RHEL 8 Benchmark 1.0.0

Because of an issue with the CIS Benchmark, the remediation of the SCAP rule that ensures permissions on the **/etc/passwd-** backup file configures permissions to **0644**. However, the **CIS Red Hat Enterprise Linux 8 Benchmark 1.0.0** requires file permissions **0600** for that file. As a consequence, the file permissions of **/etc/passwd-** are not aligned with the benchmark after remediation.

(BZ#1858866)

libselinux-python is available only through its module

The **libselinux-python** package contains only Python 2 bindings for developing SELinux applications and it is used for backward compatibility. For this reason, **libselinux-python** is no longer available in the default RHEL 8 repositories through the **yum install libselinux-python** command.

To work around this problem, enable both the **libselinux-python** and **python27** modules, and install the **libselinux-python** package and its dependencies with the following commands:

```
# yum module enable libselinux-python
# yum install libselinux-python
```

Alternatively, install **libselinux-python** using its install profile with a single command:

```
# yum module install libselinux-python:2.8/common
```

As a result, you can install **libselinux-python** using the respective module.

(BZ#1666328)

udica processes UBI 8 containers only when started with --env container=podman

The Red Hat Universal Base Image 8 (UBI 8) containers set the **container** environment variable to the **oci** value instead of the **podman** value. This prevents the **udica** tool from analyzing a container JavaScript Object Notation (JSON) file.

To work around this problem, start a UBI 8 container using a **podman** command with the **--env container=podman** parameter. As a result, **udica** can generate an SELinux policy for a UBI 8 container only when you use the described workaround.

(BZ#1763210)

SELINUX=disabled in /etc/selinux/config does not work properly

Disabling SELinux using the **SELINUX=disabled** option in the **/etc/selinux/config** results in a process in which the kernel boots with SELinux enabled and switches to disabled mode later in the boot process. This might cause memory leaks.

To work around this problem, disable SELinux by adding the **selinux=0** parameter to the kernel command line as described in the Changing SELinux modes at boot time section of the Using SELinux title if your scenario really requires to completely disable SELinux.

(JIRA:RHELPLAN-34199)

sshd -T provides inaccurate information about Ciphers, MACs and KeX algorithms

The output of the **sshd -T** command does not contain the system-wide crypto policy configuration or other options that could come from an environment file in **/etc/sysconfig/sshd** and that are applied as arguments on the **sshd** command. This occurs because the upstream OpenSSH project did not support the Include directive to support Red-Hat-provided cryptographic defaults in RHEL 8. Crypto policies are applied as command-line arguments to the **sshd** executable in the **sshd.service** unit during the service's start by using an **EnvironmentFile**. To work around the problem, use the **source** command with the environment file and pass the crypto policy as an argument to the **sshd** command, as in **sshd -T $CRYPTO_POLICY**. For additional information, see Ciphers, MACs or KeX algorithms differ from **sshd -T** to what is provided by current crypto policy level . As a result, the output from **sshd -T** matches the currently configured crypto policy.

(BZ#2044354)

OpenSSL in FIPS mode accepts only specific D-H parameters

In FIPS mode, TLS clients that use OpenSSL return a **bad dh value** error and abort TLS connections to servers that use manually generated parameters. This is because OpenSSL, when configured to work in compliance with FIPS 140-2, works only with Diffie-Hellman parameters compliant to NIST SP 800-56A rev3 Appendix D (groups 14, 15, 16, 17, and 18 defined in RFC 3526 and with groups defined in RFC 7919). Also, servers that use OpenSSL ignore all other parameters and instead select known parameters of similar size. To work around this problem, use only the compliant groups.

(BZ#1810911)

crypto-policies incorrectly allow Camellia ciphers

The RHEL 8 system-wide cryptographic policies should disable Camellia ciphers in all policy levels, as stated in the product documentation. However, the Kerberos protocol enables the ciphers by default.

To work around the problem, apply the **NO-CAMELLIA** subpolicy:

```
# update-crypto-policies --set DEFAULT:NO-CAMELLIA
```

In the previous command, replace **DEFAULT** with the cryptographic level name if you have switched from **DEFAULT** previously.

As a result, Camellia ciphers are correctly disallowed across all applications that use system-wide crypto policies only when you disable them through the workaround.

(BZ#1919155)

Smart-card provisioning process through OpenSC pkcs15-init does not work properly

The **file_caching** option is enabled in the default OpenSC configuration, and the file caching functionality does not handle some commands from the **pkcs15-init** tool properly. Consequently, the smart-card provisioning process through OpenSC fails.

To work around the problem, add the following snippet to the **/etc/opensc.conf** file:

```
app pkcs15-init {
        framework pkcs15 {
                use_file_caching = false;
        }
}
```

The smart-card provisioning through **pkcs15-init** only works if you apply the previously described workaround.

(BZ#1947025)

Connections to servers with SHA-1 signatures do not work with GnuTLS

SHA-1 signatures in certificates are rejected by the GnuTLS secure communications library as insecure. Consequently, applications that use GnuTLS as a TLS backend cannot establish a TLS connection to peers that offer such certificates. This behavior is inconsistent with other system cryptographic libraries.

To work around this problem, upgrade the server to use certificates signed with SHA-256 or stronger hash, or switch to the LEGACY policy.

(BZ#1628553)

IKE over TCP connections do not work on custom TCP ports

The **tcp-remoteport** Libreswan configuration option does not work properly. Consequently, an IKE over TCP connection cannot be established when a scenario requires specifying a non-default TCP port.

(BZ#1989050)

RHV hypervisor may not work correctly when hardening the system during installation

When installing Red Hat Virtualization Hypervisor (RHV-H) and applying the Red Hat Enterprise Linux 8 STIG profile, OSCAP Anaconda Add-on may harden the system as RHEL instead of RVH-H and remove essential packages for RHV-H. Consequently, the RHV hypervisor may not work. To work around the problem, install the RHV-H system without applying any profile hardening, and after the installation is complete, apply the profile by using OpenSCAP. As a result, the RHV hypervisor works correctly.

(BZ#2075508)

Red Hat provides the CVE OVAL reports in compressed format

Red Hat provides CVE OVAL feeds in the **bzip2-compressed** format, and they are no longer available in the XML file format. The location of feeds for RHEL 8 has been updated accordingly to reflect this change. Because referencing compressed content is not standardized, third-party SCAP scanners can have problems with scanning rules that use the feed.

(BZ#2028428)

Certain sets of interdependent rules in SSG can fail

Remediation of **SCAP Security Guide** (SSG) rules in a benchmark can fail due to undefined ordering of rules and their dependencies. If two or more rules need to be executed in a particular order, for example, when one rule installs a component and another rule configures the same component, they can run in the wrong order and remediation reports an error. To work around this problem, run the remediation twice, and the second run fixes the dependent rules.

(BZ#1750755)

Server with GUI and Workstation installations are not possible with CIS Server profiles

The CIS Server Level 1 and Level 2 security profiles are not compatible with the **Server with GUI** and **Workstation** software selections. As a consequence, a RHEL 8 installation with the **Server with GUI** software selection and CIS Server profiles is not possible. An attempted installation using the CIS Server Level 1 or Level 2 profiles and either of these software selections will generate the error message:

```
package xorg-x11-server-common has been added to the list of excluded packages, but it can't be removed from the current software selection without breaking the installation.
```

If you need to align systems with the **Server with GUI** or **Workstation** software selections according to CIS benchmarks, use the CIS Workstation Level 1 or Level 2 profiles instead.

(BZ#1843932)

Kickstart uses org_fedora_oscap instead of com_redhat_oscap in RHEL 8

The Kickstart references the Open Security Content Automation Protocol (OSCAP) Anaconda add-on as **org_fedora_oscap** instead of **com_redhat_oscap**, which might cause confusion. This is necessary for backwards compatibility backward compatibility with Red Hat Enterprise Linux 7.

(BZ#1665082)

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 8 (**xccdf_org.ssgproject.content_profile_stig**)
- DISA STIG with GUI for RHEL 8 (**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-83405-1
Rule ID: xccdf_org.ssgproject.content_rule_sshd_set_keepalive_0
STIG ID: RHEL-08-010200

Title: Set SSH Idle Timeout Interval
CCE Identifier: CCE-80906-1
Rule ID: xccdf_org.ssgproject.content_rule_sshd_set_idle_timeout
STIG ID: RHEL-08-010201
```

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 8 and DISA STIG with GUI for RHEL 8 profiles until a solution is developed.

(BZ#2038977)

Bash remediations of certain Audit rules do not work correctly

SCAP Security Guide (SSG) Bash remediations for the following SCAP rules do not add the Audit key:

- **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**

Consequently, remediation scripts fix access bits and paths in the remediated rules, but the rules without the Audit key do not conform to the OVAL check. Therefore, scans after remediations of such rules report **FAIL**. To work around the problem, add the keys to the affected rules manually.

(BZ#2119356)

Certain rsyslog priority strings do not work correctly

Support for the GnuTLS priority string for **imtcp** that allows fine-grained control over encryption is not complete. Consequently, the following priority strings do not work properly in **rsyslog**:

```
NONE:+VERS-ALL:-VERS-TLS1.3:+MAC-ALL:+DHE-RSA:+AES-256-GCM:+SIGN-RSA-SHA384:+COMP-ALL:+GROUP-ALL
```

To work around this problem, use only correctly working priority strings:

```
NONE:+VERS-ALL:-VERS-TLS1.3:+MAC-ALL:+ECDHE-RSA:+AES-128-CBC:+SIGN-RSA-SHA1:+COMP-ALL:+GROUP-ALL
```

As a result, current configurations must be limited to the strings that work correctly.

(BZ#1679512)

Negative effects of the default logging setup on performance

The default logging environment setup might consume 4 GB of memory or even more and adjustments of rate-limit values are complex when **systemd-journald** is running with **rsyslog**.

See the Negative effects of the RHEL default logging setup on performance and their mitigations Knowledgebase article for more information.

(JIRA:RHELPLAN-10431)

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)

11.7. NETWORKING

NetworkManager does not support activating bond and team ports in a specific order

NetworkManager activates interfaces alphabetically by interface names. However, if an interface appears later during the boot, for example, because the kernel needs more time to discover it, NetworkManager activates this interface later. NetworkManager does not support setting a priority on bond and team ports. Consequently, the order in which NetworkManager activates ports of these devices is not always predictable. To work around this problem, write a dispatcher script.

For an example of such a script, see the corresponding comment in the ticket.

(BZ#1920398)

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#2132754)

Systems with the IPv6_rpfilter option enabled experience low network throughput

Systems with the **IPv6_rpfilter** option enabled in the **firewalld.conf** file currently experience suboptimal performance and low network throughput in high traffic scenarios, such as 100-Gbps links. To work around the problem, disable the **IPv6_rpfilter** option. To do so, add the following line in the **/etc/firewalld/firewalld.conf** file.

```
IPv6_rpfilter=no
```

As a result, the system performs better, but also has reduced security.

(BZ#1871860)

RoCE interfaces on IBM Z lose their IP settings due to an unexpected change of the network interface name

In RHEL 8.6 and earlier, the **udev** device manager assigns on the IBM Z platform unpredictable device names to RoCE interfaces that are enumerated by a unique identifier (UID). However, in RHEL 8.7 and later, **udev** assigns predictable device names with the **eno** prefix to these interfaces.

If you update from RHEL 8.6 or earlier to 8.7 or later, these UID-enumerated interfaces have new names and no longer match the device names in NetworkManager connection profiles. Consequently, these interfaces have no IP configuration after the update.

For workarounds you can apply before the update and a fix if you have already updated the system, see RoCE interfaces on IBM Z lose their IP settings after updating to RHEL 8.7 or later .

(BZ#2169382)

11.8. KERNEL

Secure boot on IBM Power Systems does not support migration

Currently, on IBM Power Systems, logical partition (LPAR) does not boot after successful physical volume (PV) migration. As a consequence, any type of automated migration with secure boot enabled on a partition fails.

(BZ#2126777)

Using page_poison=1 can cause a kernel crash

When using **page_poison=1** as the kernel parameter on firmware with faulty EFI implementation, the operating system can cause the kernel to crash. By default, this option is disabled and it is not recommended to enable it, especially in production systems.

(BZ#2050411)

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)

Reloading an identical crash extension may cause segmentation faults

When you load a copy of an already loaded crash extension file, it might trigger a segmentation fault. Currently, the crash utility detects if an original file has been loaded. Consequently, due to two identical files co-existing in the crash utility, a namespace collision occurs, which triggers the crash utility to cause a segmentation fault.

You can work around the problem by loading the crash extension file only once. As a result, segmentation faults no longer occur in the described scenario.

(BZ#1906482)

vmcore capture fails after memory hot-plug or unplug operation

After performing the memory hot-plug or hot-unplug operation, the event comes after updating the device tree which contains memory layout information. Thereby the **makedumpfile** utility tries to access a non-existent physical address. The problem appears if all of the following conditions meet:

- A little-endian variant of IBM Power System runs RHEL 8.
- The **kdump** or **fadump** service is enabled on the system.

Consequently, the capture kernel fails to save **vmcore** if a kernel crash is triggered after the memory hot-plug or hot-unplug operation.

To work around this problem, restart the **kdump** service after hot-plug or hot-unplug:

```
# systemctl restart kdump.service
```

As a result, **vmcore** is successfully saved in the described scenario.

(BZ#1793389)

Debug kernel fails to boot in crash capture environment on RHEL 8

Due to the memory-intensive nature of the debug kernel, a problem occurs when the debug kernel is in use and a kernel panic is triggered. As a consequence, the debug kernel is not able to boot as the capture kernel and a stack trace is generated instead. To work around this problem, increase the crash kernel memory as required. As a result, the debug kernel boots successfully in the crash capture environment.

(BZ#1659609)

Allocating crash kernel memory fails at boot time

On some Ampere Altra systems, allocating the crash kernel memory during boot fails when the 32-bit region is disabled in BIOS settings. Consequently, the **kdump** service fails to start. This is caused by memory fragmentation in the region below 4 GB with no fragment being large enough to contain the crash kernel memory.

To work around this problem, enable the 32-bit memory region in BIOS as follows:

1. Open the BIOS settings on your system.

2. Open the **Chipset** menu.
3. Under **Memory Configuration**, enable the **Slave 32-bit** option.

As a result, crash kernel memory allocation within the 32-bit region succeeds and the **kdump** service works as expected.

(BZ#1940674)

The QAT manager leaves no spare device for LKCF

The Intel® QuickAssist Technology (QAT) manager (**qatmgr**) is a user space process, which by default uses all QAT devices in the system. As a consequence, there are no QAT devices left for the Linux Kernel Cryptographic Framework (LKCF). There is no need to work around this situation, as this behavior is expected and a majority of users will use acceleration from the user space.

(BZ#1920086)

The kernel ACPI driver reports it has no access to a PCIe ECAM memory region

The Advanced Configuration and Power Interface (ACPI) table provided by firmware does not define a memory region on the PCI bus in the Current Resource Settings (_CRS) method for the PCI bus device. Consequently, the following warning message occurs during the system boot:

```
[    2.817152] acpi PNP0A08:00: [Firmware Bug]: ECAM area [mem 0x30000000-0x31ffffff] not
reserved in ACPI namespace
[    2.827911] acpi PNP0A08:00: ECAM at [mem 0x30000000-0x31ffffff] for [bus 00-1f]
```

However, the kernel is still able to access the **0x30000000-0x31ffffff** memory region, and can assign that memory region to the PCI Enhanced Configuration Access Mechanism (ECAM) properly. You can verify that PCI ECAM works correctly by accessing the PCIe configuration space over the 256 byte offset with the following output:

```
03:00.0 Non-Volatile memory controller: Sandisk Corp WD Black 2018/PC SN720 NVMe SSD (prog-
if 02 [NVM Express])
 ...
        Capabilities: [900 v1] L1 PM Substates
                L1SubCap: PCI-PM_L1.2- PCI-PM_L1.1- ASPM_L1.2+ ASPM_L1.1- L1_PM_Substates+
                        PortCommonModeRestoreTime=255us PortTPowerOnTime=10us
                L1SubCtl1: PCI-PM_L1.2- PCI-PM_L1.1- ASPM_L1.2- ASPM_L1.1-
                        T_CommonMode=0us LTR1.2_Threshold=0ns
                L1SubCtl2: T_PwrOn=10us
```

As a result, you can ignore the warning message.

For more information about the problem, see the "Firmware Bug: ECAM area **mem 0x30000000-0x31ffffff** not reserved in ACPI namespace" appears during system boot solution.

(BZ#1868526)

The tuned-adm profile powersave command causes the system to become unresponsive

Executing the **tuned-adm profile powersave** command leads to an unresponsive state of the Penguin Valkyrie 2000 2-socket systems with the older Thunderx (CN88xx) processors. Consequently, reboot the system to resume working. To work around this problem, avoid using the **powersave** profile if your system matches the mentioned specifications.

(BZ#1609288)

The HP NMI watchdog does not always generate a crash dump

In certain cases, the **hpwdt** driver for the HP NMI watchdog is not able to claim a non-maskable interrupt (NMI) generated by the HPE watchdog timer because the NMI was instead consumed by the **perfmon** driver.

The missing NMI is initiated by one of two conditions:

1. The **Generate NMI** button on the Integrated Lights-Out (iLO) server management software. This button is triggered by a user.

2. The **hpwdt** watchdog. The expiration by default sends an NMI to the server.

Both sequences typically occur when the system is unresponsive. Under normal circumstances, the NMI handler for both these situations calls the **kernel panic()** function and if configured, the **kdump** service generates a **vmcore** file.

Because of the missing NMI, however, **kernel panic()** is not called and **vmcore** is not collected.

In the first case (1.), if the system was unresponsive, it remains so. To work around this scenario, use the virtual **Power** button to reset or power cycle the server.

In the second case (2.), the missing NMI is followed 9 seconds later by a reset from the Automated System Recovery (ASR).

The HPE Gen9 Server line experiences this problem in single-digit percentages. The Gen10 at an even smaller frequency.

(BZ#1602962)

Using irqpoll causes vmcore generation failure

Due to an existing problem with the **nvme** driver on the 64-bit ARM architecture that run on the Amazon Web Services Graviton 1 processor, causes **vmcore** generation to fail when you provide the **irqpoll** kernel command line parameter to the first kernel. Consequently, no **vmcore** file is dumped in the **/var/crash/** directory upon a kernel crash. To work around this problem:

1. Append **irqpoll** to **KDUMP_COMMANDLINE_REMOVE** variable in the **/etc/sysconfig/kdump** file.

```
# KDUMP_COMMANDLINE_REMOVE="hugepages hugepagesz slub_debug quiet
log_buf_len swiotlb"
```

2. Remove **irqpoll** from **KDUMP_COMMANDLINE_APPEND** variable in the **/etc/sysconfig/kdump** file.

```
# KDUMP_COMMANDLINE_APPEND="irqpoll nr_cpus=1 reset_devices
cgroup_disable=memory udev.children-max=2 panic=10 swiotlb=noforce novmcoredd"
```

3. Restart the **kdump** service:

```
# systemctl restart kdump
```

As a result, the first kernel boots correctly and the **vmcore** file is expected to be captured upon the kernel crash.

Note that the Amazon Web Services Graviton 2 and Amazon Web Services Graviton 3 processors do not require you to manually remove the **irqpoll** parameter in the **/etc/sysconfig/kdump** file.

The **kdump** service can use a significant amount of crash kernel memory to dump the **vmcore** file. Ensure that the capture kernel has sufficient memory available for the **kdump** service.

For related information on this Known Issue, see The irqpoll kernel command line parameter might cause vmcore generation failure article.

(BZ#1654962)

Connections fail when attaching a virtual function to virtual machine

Pensando network cards that use the **ionic** device driver silently accept VLAN tag configuration requests and attempt configuring network connections while attaching network virtual functions (**VF**) to a virtual machine (**VM**). Such network connections fail as this feature is not yet supported by the card's firmware.

(BZ#1930576)

The OPEN MPI library may trigger run-time failures with default PML

In OPEN Message Passing Interface (OPEN MPI) implementation 4.0.x series, Unified Communication X (UCX) is the default point-to-point communicator (PML). The later versions of OPEN MPI 4.0.x series deprecated **openib** Byte Transfer Layer (BTL).

However, OPEN MPI, when run over a **homogeneous** cluster (same hardware and software configuration), UCX still uses **openib** BTL for MPI one-sided operations. As a consequence, this may trigger execution errors. To work around this problem:

- Run the **mpirun** command using following parameters:

```
-mca btl openib -mca pml ucx -x UCX_NET_DEVICES=mlx5_ib0
```

where,

- The **-mca btl openib** parameter disables **openib** BTL
- The **-mca pml ucx** parameter configures OPEN MPI to use **ucx** PML.
- The **x UCX_NET_DEVICES=** parameter restricts UCX to use the specified devices

The OPEN MPI, when run over a **heterogeneous** cluster (different hardware and software configuration), it uses UCX as the default PML. As a consequence, this may cause the OPEN MPI jobs to run with erratic performance, unresponsive behavior, or crash failures. To work around this problem, set the UCX priority as:

- Run the **mpirun** command using following parameters:

```
-mca pml_ucx_priority 5
```

As a result, the OPEN MPI library is able to choose an alternative available transport layer over UCX.

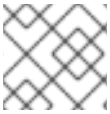
(BZ#1866402)

The Solarflare fails to create maximum number of virtual functions (VFs)

The Solarflare NICs fail to create a maximum number of VFs due to insufficient resources. You can check the maximum number of VFs that a PCIe device can create in the **/sys/bus/pci/devices/PCI_ID/sriov_totalvfs** file. To workaround this problem, you can either adjust the number of VFs or the VF MSI interrupt value to a lower value, either from **Solarflare Boot Manager** on startup, or using Solarflare **sfboot** utility. The default VF MSI interrupt value is **8**.

- To adjust the VF MSI interrupt value using **sfboot**:

```
# sfboot vf-msix-limit=2
```



NOTE

Adjusting VF MSI interrupt value affects the VF performance.

For more information about parameters to be adjusted accordingly, see the **Solarflare Server Adapter user guide**.

(BZ#1971506)

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#2106341)

Memory allocation for kdump fails on the 64-bit ARM architectures

On certain 64-bit ARM based systems, the firmware uses the non-contiguous memory allocation method, which reserves memory randomly at different scattered locations. Consequently, due to the unavailability of consecutive blocks of memory, the crash kernel cannot reserve memory space for the **kdump** mechanism.

To work around this problem, install the kernel version provided by RHEL 8.8 and later. The latest version of RHEL supports the **fallback** dump capture mechanism that helps to find a suitable memory region in the described scenario.

(BZ#2214235)

Hardware certification of the real-time kernel on systems with large core-counts might require passing the skew-tick=1 boot parameter to avoid lock contentions

Large or moderate sized systems with numerous sockets and large core-counts can experience latency spikes due to lock contentions on **xtime_lock**, which is used in the timekeeping system. As a consequence, latency spikes and delays in hardware certifications might occur on multiprocessing

systems. As a workaround, you can offset the timer tick per CPU to start at a different time by adding the **skew_tick=1** boot parameter.

To avoid lock conflicts, enable **skew_tick=1**:

1. Enable the **skew_tick=1** parameter with **grubby**.

```
# grubby --update-kernel=ALL --args="skew_tick=1"
```

2. Reboot for changes to take effect.

3. Verify the new settings by running the **cat /proc/cmdline** command.

Note that enabling **skew_tick=1** causes a significant increase in power consumption and, therefore, it must be enabled only if you are running latency sensitive real-time workloads.

(BZ#2214508)

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#1900829)

11.10. FILE SYSTEMS AND STORAGE

Limitations of LVM writecache

The **writecache** LVM caching method has the following limitations, which are not present in the **cache** method:

- You cannot name a **writecache** logical volume when using **pvmove** commands.

- You cannot use logical volumes with **writecache** in combination with thin pools or VDO.

The following limitation also applies to the **cache** method:

- You cannot resize a logical volume while **cache** or **writecache** is attached to it.

(JIRA:RHELPLAN-27987, BZ#1798631, BZ#1808012)

XFS quota warnings are triggered too often

Using the quota timer results in quota warnings triggering too often, which causes soft quotas to be enforced faster than they should. To work around this problem, do not use soft quotas, which will prevent triggering warnings. As a result, the amount of warning messages will not enforce soft quota limit anymore, respecting the configured timeout.

(BZ#2059262)

LVM mirror devices that store a LUKS volume sometimes become unresponsive

Mirrored LVM devices with a segment type of **mirror** that store a LUKS volume might become unresponsive under certain conditions. The unresponsive devices reject all I/O operations.

To work around the issue, Red Hat recommends that you use LVM RAID 1 devices with a segment type of **raid1** instead of **mirror** if you need to stack LUKS volumes on top of resilient software-defined storage.

The **raid1** segment type is the default RAID configuration type and replaces **mirror** as the recommended solution.

To convert **mirror** devices to **raid**, see Converting a mirrored LVM device to a RAID1 logical volume .

(BZ#1730502)

The /boot file system cannot be placed on LVM

You cannot place the **/boot** file system on an LVM logical volume. This limitation exists for the following reasons:

- On EFI systems, the *EFI System Partition* conventionally serves as the **/boot** file system. The uEFI standard requires a specific GPT partition type and a specific file system type for this partition.
- RHEL 8 uses the *Boot Loader Specification* (BLS) for system boot entries. This specification requires that the **/boot** file system is readable by the platform firmware. On EFI systems, the platform firmware can read only the **/boot** configuration defined by the uEFI standard.
- The support for LVM logical volumes in the GRUB 2 boot loader is incomplete. Red Hat does not plan to improve the support because the number of use cases for the feature is decreasing due to standards such as uEFI and BLS.

Red Hat does not plan to support **/boot** on LVM. Instead, Red Hat provides tools for managing system snapshots and rollback that do not need the **/boot** file system to be placed on an LVM logical volume.

(BZ#1496229)

LVM no longer allows creating volume groups with mixed block sizes

LVM utilities such as **vgcreate** or **vgextend** no longer allow you to create volume groups (VGs) where the physical volumes (PVs) have different logical block sizes. LVM has adopted this change because file systems fail to mount if you extend the underlying logical volume (LV) with a PV of a different block size.

To re-enable creating VGs with mixed block sizes, set the **allow_mixed_block_sizes=1** option in the **lvm.conf** file.

(BZ#1768536)

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)

VDO driver bug can cause device freezes through journal blocks

While tracking a device-mapper **suspend** operation, a bug in the VDO driver causes the system to mark some journal blocks as waiting for metadata updates. The updates already apply since the **suspend** call.

When the journal wraps around back to the same physical block, the block stops being available. Eventually, all writes stop until the block is available again. The **growPhysical**, **growLogical**, and **setWritePolicy** operations on VDO devices include a suspend/resume cycle, which can lead to the device freezing after a number of journal updates.

Increasing the size of the VDO pool or the logical volume on top of it or using the **pvmove** and **lvchange** operations on LVM tools managed VDO devices can also trigger this problem.

For a workaround, change the VDO device settings in any way that involves a suspend/resume cycle, shut down the VDO device completely and then start it again. This clears the incorrect in-memory state and resets the journal blocks. As a result, the device is not frozen anymore and works correctly.

(BZ#2109047)

System hangs due to soft lockup while starting a VDO volume

Due to fixing the kernel ABI breakage in the **pv_mmu_ops** structure, RHEL 8.7 systems with kernel version **4.18.0-425.10.1.el8_7**, that is RHEL-8.7.0.2-BaseOS, hang or encounter a kernel panic due to soft lockup while starting a Virtual Data Optimizer (VDO) volume. To work around this issue, disable any enabled VDO volumes before booting into **kernel-4.18.0-425.10.1.el8_7** to prevent system hangs, or downgrade to the previous version of the kernel, which is **4.18.0-425.3.1.el8**, to retain VDO functionality.

(BZ#2158783)

11.11. DYNAMIC PROGRAMMING LANGUAGES, WEB AND DATABASE SERVERS

getpwnam() might fail when called by a 32-bit application

When a user of NIS uses a 32-bit application that calls the **getpwnam()** function, the call fails if the **nss_nis.i686** package is missing. To work around this problem, manually install the missing package by using the **yum install nss_nis.i686** command.

(BZ#1803161)

PAM plug-in version 1.0 does not work in MariaDB

MariaDB 10.3 provides the Pluggable Authentication Modules (PAM) plug-in version 1.0. **MariaDB 10.5** provides the plug-in versions 1.0 and 2.0, version 2.0 is the default.

The **MariaDB** PAM plug-in version 1.0 does not work in RHEL 8. To work around this problem, use the PAM plug-in version 2.0 provided by the **mariadb:10.5** module stream.

(BZ#1942330)

Symbol conflicts between OpenLDAP libraries might cause crashes in httpd

When both the **libldap** and **libldap_r** libraries provided by OpenLDAP are loaded and used within a single process, symbol conflicts between these libraries might occur. Consequently, Apache **httpd** child processes using the PHP **ldap** extension might terminate unexpectedly if the **mod_security** or **mod_auth_openidc** modules are also loaded by the **httpd** configuration.

Since the RHEL 8.3 update to the Apache Portable Runtime (APR) library, you can work around the problem by setting the **APR_DEEPBIND** environment variable, which enables the use of the **RTLD_DEEPBIND** dynamic linker option when loading **httpd** modules. When the **APR_DEEPBIND** environment variable is enabled, crashes no longer occur in **httpd** configurations that load conflicting libraries.

(BZ#1819607)

11.12. IDENTITY MANAGEMENT

Using the cert-fix utility with the --agent-uid pkidbuser option breaks Certificate System

Using the **cert-fix** utility with the **--agent-uid pkidbuser** option corrupts the LDAP configuration of Certificate System. As a consequence, Certificate System might become unstable and manual steps are required to recover the system.

(BZ#1729215)

The /var/log/lastlog sparse file on IdM hosts can cause performance problems

During the IdM installation, a range of 200,000 UIDs from a total of 10,000 possible ranges is randomly selected and assigned. Selecting a random range in this way significantly reduces the probability of conflicting IDs in case you decide to merge two separate IdM domains in the future.

However, having high UIDs can create problems with the **/var/log/lastlog** file. For example, if a user with the UID of 1280000008 logs in to an IdM client, the local **/var/log/lastlog** file size increases to almost 400 GB. Although the actual file is sparse and does not use all that space, certain applications are not designed to identify sparse files by default and may require a specific option to handle them. For example, if the setup is complex and a backup and copy application does not handle sparse files correctly, the file is copied as if its size was 400 GB. This behavior can cause performance problems.

To work around this problem:

- In case of a standard package, refer to its documentation to identify the option that handles sparse files.
- In case of a custom application, ensure that it is able to manage sparse files such as **/var/log/lastlog** correctly.

(JIRA:RHELPLAN-59111)

FIPS mode does not support using a shared secret to establish a cross-forest trust

Establishing a cross-forest trust using a shared secret fails in FIPS mode because NTLMSSP authentication is not FIPS-compliant. To work around this problem, authenticate with an Active Directory (AD) administrative account when establishing a trust between an IdM domain with FIPS mode enabled and an AD domain.

(BZ#1924707)

FreeRADIUS server fails to run in FIPS mode

By default, in FIPS mode, OpenSSL disables the use of the MD5 digest algorithm. As the RADIUS protocol requires MD5 to encrypt a secret between the RADIUS client and the RADIUS server, this causes the FreeRADIUS server to fail in FIPS mode.

To work around this problem, follow these steps:

Procedure

1. Create the environment variable, **RADIUS_MD5_FIPS_OVERRIDE** for the **radiusd** service:

```
systemctl edit radiusd

[Service]
Environment=RADIUS_MD5_FIPS_OVERRIDE=1
```

2. To apply the change, reload the **systemd** configuration and start the **radiusd** service:

```
# systemctl daemon-reload
# systemctl start radiusd
```

3. To run FreeRADIUS in debug mode:

```
# RADIUS_MD5_FIPS_OVERRIDE=1 radiusd -X
```

Note that though FreeRADIUS can run in FIPS mode, this does not mean that it is FIPS compliant as it uses weak ciphers and functions when in FIPS mode.

For more information on configuring FreeRADIUS authentication in FIPS mode, see How to configure FreeRADIUS authentication in FIPS mode.

(BZ#1958979)

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#2125182)

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)

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#2120572)

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#2122919)

Actions required when running Samba as a print server and updating from RHEL 8.4 and earlier

With this update, the **samba** package no longer creates the **/var/spool/samba/** directory. If you use Samba as a print server and use **/var/spool/samba/** in the **[printers]** share to spool print jobs, SELinux prevents Samba users from creating files in this directory. Consequently, print jobs fail and the **auditd** service logs a **denied** message in **/var/log/audit/audit.log**. To avoid this problem after updating your system from 8.4 and earlier:

1. Search the **[printers]** share in the **/etc/samba/smb.conf** file.
2. If the share definition contains **path = /var/spool/samba/**, update the setting and set the **path** parameter to **/var/tmp/**.
3. Restart the **smbd** service:

```
# systemctl restart smbd
```

If you newly installed Samba on RHEL 8.5 or later, no action is required. The default **/etc/samba/smb.conf** file provided by the **samba-common** package in this case already uses the **/var/tmp/** directory to spool print jobs.

(BZ#2009213)

Downgrading authselect after the rebase to version 1.2.2 breaks system authentication

The **authselect** package has been rebased to the latest upstream version **1.2.2**. Downgrading **authselect** is not supported and breaks system authentication for all users, including **root**.

If you downgraded the **authselect** package to **1.2.1** or earlier, perform the following steps to work around this problem:

1. At the GRUB boot screen, select **Red Hat Enterprise Linux** with the version of the kernel that you want to boot and press **e** to edit the entry.
2. Type **single** as a separate word at the end of the line that starts with **linux** and press **Ctrl+X** to start the boot process.
3. Upon booting in single-user mode, enter the root password.
4. Restore authselect configuration using the following command:

```
# authselect select sssd --force
```

(BZ#1892761)

The default keyword for enabled ciphers in the NSS does not work in conjunction with other ciphers

In Directory Server you can use the **default** keyword to refer to the default ciphers enabled in the network security services (NSS). However, if you want to enable the default ciphers and additional ones using the command line or web console, Directory Server fails to resolve the **default** keyword. As a consequence, the server enables only the additionally specified ciphers and logs an error similar to the following:

```
Security Initialization - SSL alert: Failed to set SSL cipher preference information: invalid ciphers
<default,+cipher_name>: format is +cipher1,-cipher2... (Netscape Portable Runtime error 0 - no error)
```

As a workaround, specify all ciphers that are enabled by default in NSS including the ones you want to additionally enable.

(BZ#1817505)

pki-core-debuginfo update from RHEL 8.6 to RHEL 8.7 fails

Updating the **pki-core-debuginfo** package from RHEL 8.6 to RHEL 8.7 fails. To work around this problem, run the following commands:

1. **yum remove pki-core-debuginfo**
2. **yum update -y**
3. **yum install pki-core-debuginfo**
4. **yum install idm-pki-symkey-debuginfo idm-pki-tools-debuginfo**

(BZ#2134093)

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)

11.13. DESKTOP

Disabling flatpak repositories from Software Repositories is not possible

Currently, it is not possible to disable or remove **flatpak** repositories in the Software Repositories tool in the GNOME Software utility.

(BZ#1668760)

Generation 2 RHEL 8 virtual machines sometimes fail to boot on Hyper-V Server 2016 hosts

When using RHEL 8 as the guest operating system on a virtual machine (VM) running on a Microsoft Hyper-V Server 2016 host, the VM in some cases fails to boot and returns to the GRUB boot menu. In addition, the following error is logged in the Hyper-V event log:

```
The guest operating system reported that it failed with the following error code: 0x1E
```

This error occurs due to a UEFI firmware bug on the Hyper-V host. To work around this problem, use Hyper-V Server 2019 or later as the host.

(BZ#1583445)

Drag-and-drop does not work between desktop and applications

Due to a bug in the **gnome-shell-extensions** package, the drag-and-drop functionality does not currently work between desktop and applications. Support for this feature will be added back in a future release.

(BZ#1717947)

11.14. GRAPHICS INFRASTRUCTURES

radeon fails to reset hardware correctly

The **radeon** kernel driver currently does not reset hardware in the kexec context correctly. Instead, **radeon** falls over, which causes the rest of the **kdump** service to fail.

To work around this problem, disable **radeon** in **kdump** by adding the following line to the **/etc/kdump.conf** file:

```
dracut_args --omit-drivers "radeon"
force_rebuild 1
```

Restart the machine and **kdump**. After starting **kdump**, the **force_rebuild 1** line may be removed from the configuration file.

Note that in this scenario, no graphics will be available during **kdump**, but **kdump** will work successfully.

(BZ#1694705)

Multiple HDR displays on a single MST topology may not power on

On systems using NVIDIA Turing GPUs with the **nouveau** driver, using a **DisplayPort** hub (such as a laptop dock) with multiple monitors which support HDR plugged into it may result in failure to turn on. This is due to the system erroneously thinking there is not enough bandwidth on the hub to support all of the displays.

(BZ#1812577)

GUI in ESXi might crash due to low video memory

The graphical user interface (GUI) on RHEL virtual machines (VMs) in the VMware ESXi 7.0.1 hypervisor with vCenter Server 7.0.1 requires a certain amount of video memory. If you connect multiple consoles or high-resolution monitors to the VM, the GUI requires at least 16 MB of video memory. If you start the GUI with less video memory, the GUI might terminate unexpectedly.

To work around the problem, configure the hypervisor to assign at least 16 MB of video memory to the VM. As a result, the GUI on the VM no longer crashes.

If you encounter this issue, Red Hat recommends that you report it to VMware.

See also the following VMware article: VMs with high resolution VM console may experience a crash on ESXi 7.0.1 (83194).

(BZ#1910358)

VNC Viewer displays wrong colors with the 16-bit color depth on IBM Z

The VNC Viewer application displays wrong colors when you connect to a VNC session on an IBM Z server with the 16-bit color depth.

To work around the problem, set the 24-bit color depth on the VNC server. With the **Xvnc** server, replace the **-depth 16** option with **-depth 24** in the **Xvnc** configuration.

As a result, VNC clients display the correct colors but use more network bandwidth with the server.

(BZ#1886147)

Unable to run graphical applications using sudo command

When trying to run graphical applications as a user with elevated privileges, the application fails to open with an error message. The failure happens because **Xwayland** is restricted by the **Xauthority** file to use regular user credentials for authentication.

To work around this problem, use the **sudo -E** command to run graphical applications as a **root** user.

(BZ#1673073)

Hardware acceleration is not supported on ARM

Built-in graphics drivers do not support hardware acceleration or the Vulkan API on the 64-bit ARM architecture.

To enable hardware acceleration or Vulkan on ARM, install the proprietary Nvidia driver.

(JIRA:RHELPLAN-57914)

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#2130159)

11.15. 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)

11.16. RED HAT ENTERPRISE LINUX SYSTEM ROLES

Unable to manage **localhost** by using the **localhost** hostname in the playbook or inventory

With the inclusion of the **ansible-core 2.13** package in RHEL, if you are running Ansible on the same host you manage your nodes, you cannot do it by using the **localhost** hostname in your playbook or inventory. This happens because **ansible-core 2.13** uses the **python38** module, and many of the libraries are missing, for example, **blivet** for the **storage** role, **gobject** for the **network** role. To workaround this problem, if you are already using the **localhost** hostname in your playbook or inventory, you can add a connection, by using **ansible_connection=local**, or by creating an inventory file that lists **localhost** with the **ansible_connection=local** option. With that, you are able to manage resources on **localhost**. For more details, see the article RHEL System Roles playbooks fail when run on localhost .

(BZ#2041997)

11.17. VIRTUALIZATION

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#2020133)

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#2077770)

Attaching LUN devices to virtual machines using virtio-blk does not work

The q35 machine type does not support transitional virtio 1.0 devices, and RHEL 8 therefore lacks support for features that were deprecated in virtio 1.0. In particular, it is not possible on a RHEL 8 host to send SCSI commands from virtio-blk devices. As a consequence, attaching a physical disk as a LUN device to a virtual machine fails when using the virtio-blk controller.

Note that physical disks can still be passed through to the guest operating system, but they should be configured with the **device='disk'** option rather than **device='lun'**.

(BZ#1777138)

Virtual machines with iommu_platform=on fail to start on IBM POWER

RHEL 8 currently does not support the **iommu_platform=on** parameter for virtual machines (VMs) on IBM POWER system. As a consequence, starting a VM with this parameter on IBM POWER hardware results in the VM becoming unresponsive during the boot process.

(BZ#1910848)

IBM POWER hosts may crash when using the ibmvfc driver

When running RHEL 8 on a PowerVM logical partition (LPAR), a variety of errors may currently occur due problems with the **ibmvfc** driver. As a consequence, the host's kernel may panic under certain circumstances, such as:

- Using the Live Partition Mobility (LPM) feature
- Resetting a host adapter
- Using SCSI error handling (SCSI EH) functions

(BZ#1961722)

Using perf kvm record on IBM POWER Systems can cause the VM to crash

When using a RHEL 8 host on the little-endian variant of IBM POWER hardware, using the **perf kvm record** command to collect trace event samples for a KVM virtual machine (VM) in some cases results in the VM becoming unresponsive. This situation occurs when:

- The **perf** utility is used by an unprivileged user, and the **-p** option is used to identify the VM - for example **perf kvm record -e trace_cycles -p 12345**.
- The VM was started using the **virsh** shell.

To work around this problem, use the **perf kvm** utility with the **-i** option to monitor VMs that were created using the **virsh** shell. For example:

```
# perf kvm record -e trace_imc/trace_cycles/  -p <guest pid> -i
```

Note that when using the **-i** option, child tasks do not inherit counters, and threads will therefore not be monitored.

(BZ#1924016)

Windows Server 2016 virtual machines with Hyper-V enabled fail to boot when using certain CPU models

Currently, it is not possible to boot a virtual machine (VM) that uses Windows Server 2016 as the guest operating system, has the Hyper-V role enabled, and uses one of the following CPU models:

- EPYC-IBPB
- EPYC

To work around this problem, use the **EPYC-v3** CPU model, or manually enable the **xsaves** CPU flag for the VM.

(BZ#1942888)

Migrating a POWER9 guest from a RHEL 7-ALT host to RHEL 8 fails

Currently, migrating a POWER9 virtual machine from a RHEL 7-ALT host system to RHEL 8 becomes unresponsive with a **Migration status: active** status.

To work around this problem, disable Transparent Huge Pages (THP) on the RHEL 7-ALT host, which enables the migration to complete successfully.

(BZ#1741436)

Using virt-customize sometimes causes guestfs-firstboot to fail

After modifying a virtual machine (VM) disk image using the **virt-customize** utility, the **guestfs-firstboot** service in some cases fails due to incorrect SELinux permissions. This causes a variety of problems during VM startup, such as failing user creation or system registration.

To avoid this problem, add **--selinux-relabel** to the kernel command line of the VM after modifying its disk image with **virt-customize**.

(BZ#1554735)

Deleting a forward interface from a macvtap virtual network resets all connection counts of this network

Currently, deleting a forward interface from a **macvtap** virtual network with multiple forward interfaces also resets the connection status of the other forward interfaces of the network. As a consequence, the connection information in the live network XML is incorrect. Note, however, that this does not affect the functionality of the virtual network. To work around the issue, restart the **libvirtd** service on your host.

(BZ#1332758)

Virtual machines with SLOF fail to boot in netcat interfaces

When using a netcat (**nc**) interface to access the console of a virtual machine (VM) that is currently waiting at the Slimline Open Firmware (SLOF) prompt, the user input is ignored and VM stays unresponsive. To work around this problem, use the **nc -C** option when connecting to the VM, or use a telnet interface instead.

(BZ#1974622)

Attaching mediated devices to virtual machines in virt-manager in some cases fails

The **virt-manager** application is currently able to detect mediated devices, but cannot recognize whether the device is active. As a consequence, attempting to attach an inactive mediated device to a running virtual machine (VM) using **virt-manager** fails. Similarly, attempting to create a new VM that uses an inactive mediated device fails with a **device not found** error.

To work around this issue, use the **virsh nodedev-start** or **mdevctl start** commands to activate the mediated device before using it in **virt-manager**.

(BZ#2026985)

RHEL 9 virtual machines fail to boot in POWER8 compatibility mode

Currently, booting a virtual machine (VM) that runs RHEL 9 as its guest operating system fails if the VM also uses CPU configuration similar to the following:

```
<cpu mode="host-model">
  <model>power8</model>
</cpu>
```

To work around this problem, do not use POWER8 compatibility mode in RHEL 9 VMs.

In addition, note that running RHEL 9 VMs is not possible on POWER8 hosts.

(BZ#2035158)

Restarting the OVS service on a host might block network connectivity on its running VMs

When the Open vSwitch (OVS) service restarts or crashes on a host, virtual machines (VMs) that are running on this host cannot recover the state of the networking device. As a consequence, VMs might be completely unable to receive packets.

This problem only affects systems that use the packed virtqueue format in their **virtio** networking stack.

To work around this problem, use the **packed=off** parameter in the **virtio** networking device definition to disable packed virtqueue. With packed virtqueue disabled, the state of the networking device can, in some situations, be recovered from RAM.

(BZ#1792683)

Virtual machines sometimes fail to start when using many virtio-blk disks

Adding a large number of virtio-blk devices to a virtual machine (VM) may exhaust the number of interrupt vectors available in the platform. If this occurs, the VM's guest OS fails to boot, and displays a **dracut-initqueue[392]: Warning: Could not boot** error.

(BZ#1719687)

SUID and SGID are not cleared automatically on virtiofs

When you run the **virtiofsd** service with the **killpriv_v2** feature, your system may not automatically clear the SUID and SGID permissions after performing some file-system operations. Consequently, not clearing the permissions might cause a potential security threat. To work around this issue, disable the **killpriv_v2** feature by entering the following command:

```
# virtiofsd -o no_killpriv_v2
```

(BZ#1966475)

SMT CPU topology is not detected by VMs when using host passthrough mode on AMD EPYC

When a virtual machine (VM) boots with the CPU host passthrough mode on an AMD EPYC host, the **TOPOEXT** CPU feature flag is not present. Consequently, the VM is not able to detect a virtual CPU topology with multiple threads per core. To work around this problem, boot the VM with the EPYC CPU model instead of host passthrough.

(BZ#1740002)

11.18. RHEL IN CLOUD ENVIRONMENTS

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)

kdump sometimes does not start on Azure and Hyper-V

On RHEL 8 guest operating systems hosted on the Microsoft Azure or Hyper-V hypervisors, starting the **kdump** kernel in some cases fails when post-exec notifiers are enabled.

To work around this problem, disable crash kexec post notifiers:

```
# echo N > /sys/module/kernel/parameters/crash_kexec_post_notifiers
```

(BZ#1865745)

The SCSI host address sometimes changes when booting a Hyper-V VM with multiple guest disks

Currently, when booting a RHEL 8 virtual machine (VM) on the Hyper-V hypervisor, the host portion of the *Host, Bus, Target, Lun* (HBTL) SCSI address in some cases changes. As a consequence, automated tasks set up with the HBTL SCSI identification or device node in the VM do not work consistently. This occurs if the VM has more than one disk or if the disks have different sizes.

To work around the problem, modify your kickstart files, using one of the following methods:

Method 1: Use persistent identifiers for SCSI devices.

You can use for example the following powershell script to determine the specific device identifiers:

```
# Output what the /dev/disk/by-id/<value> for the specified hyper-v virtual disk.
# Takes a single parameter which is the virtual disk file.
# Note: kickstart syntax works with and without the /dev/ prefix.
param (
    [Parameter(Mandatory=$true)][string]$virtualdisk
)

$what = Get-VHD -Path $virtualdisk
$part = $what.DiskIdentifier.ToLower().split('-')

$p = $part[0]
$s0 = $p[6] + $p[7] + $p[4] + $p[5] + $p[2] + $p[3] + $p[0] + $p[1]

$p = $part[1]
$s1 =  $p[2] + $p[3] + $p[0] + $p[1]

[string]::format("/dev/disk/by-id/wwn-0x60022480{0}{1}{2}", $s0, $s1, $part[4])
```

You can use this script on the hyper-v host, for example as follows:

```
PS C:\Users\Public\Documents\Hyper-V\Virtual hard disks> .\by-id.ps1 .\Testing_8\disk_3_8.vhdx
/dev/disk/by-id/wwn-0x60022480e00bc367d7fd902e8bf0d3b4
PS C:\Users\Public\Documents\Hyper-V\Virtual hard disks> .\by-id.ps1 .\Testing_8\disk_3_9.vhdx
/dev/disk/by-id/wwn-0x600224807270e09717645b1890f8a9a2
```

Afterwards, the disk values can be used in the kickstart file, for example as follows:

```
part / --fstype=xfs --grow --asprimary --size=8192 --ondisk=/dev/disk/by-id/wwn-
0x600224807270e09717645b1890f8a9a2
part /home --fstype="xfs" --grow --ondisk=/dev/disk/by-id/wwn-
0x60022480e00bc367d7fd902e8bf0d3b4
```

As these values are specific for each virtual disk, the configuration needs to be done for each VM instance. It may, therefore, be useful to use the **%include** syntax to place the disk information into a separate file.

Method 2: Set up device selection by size.

A kickstart file that configures disk selection based on size must include lines similar to the following:

```
...

# Disk partitioning information is supplied in a file to kick start
%include /tmp/disks

...

# Partition information is created during install using the %pre section
%pre --interpreter /bin/bash --log /tmp/ks_pre.log
```

```
# Dump whole SCSI/IDE disks out sorted from smallest to largest ouputting
# just the name
disks=(`lsblk -n -o NAME -I -b -x SIZE -d -I 8,3`) || exit 1

# We are assuming we have 3 disks which will be used
# and we will create some variables to represent
d0=${disks[0]}
d1=${disks[1]}
d2=${disks[2]}

echo "part /home --fstype="xfs" --ondisk=$d2 --grow" >> /tmp/disks
echo "part swap --fstype="swap" --ondisk=$d0 --size=4096" >> /tmp/disks
echo "part / --fstype="xfs" --ondisk=$d1 --grow" >> /tmp/disks
echo "part /boot --fstype="xfs" --ondisk=$d1 --size=1024" >> /tmp/disks

%end
```

(BZ#1906870)

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)

11.19. SUPPORTABILITY

The **getattachment** command fails to download multiple attachments at once

The **redhat-support-tool** command offers the **getattachment** subcommand for downloading attachments. However, **getattachment** is currently only able to download a single attachment and fails to download multiple attachments.

As a workaround, you can download multiple attachments one by one by passing the case number and UUID for each attachment in the **getattachment** subcommand.

(BZ#2064575)

redhat-support-tool does not work with the **FUTURE** crypto policy

Because a cryptographic key used by a certificate on the Customer Portal API does not meet the requirements by the **FUTURE** system-wide cryptographic policy, the **redhat-support-tool** utility does not work with this policy level at the moment.

To work around this problem, use the **DEFAULT** crypto policy while connecting to the Customer Portal API.

(BZ#1802026)

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#2011413)

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)

CHAPTER 12. INTERNATIONALIZATION

12.1. RED HAT ENTERPRISE LINUX 8 INTERNATIONAL LANGUAGES

Red Hat Enterprise Linux 8 supports the installation of multiple languages and the changing of languages based on your requirements.

- East Asian Languages - Japanese, Korean, Simplified Chinese, and Traditional Chinese.
- European Languages - English, German, Spanish, French, Italian, Portuguese, and Russian.

The following table lists the fonts and input methods provided for various major languages.

Language	Default Font (Font Package)	Input Methods
English	dejavu-sans-fonts	
French	dejavu-sans-fonts	
German	dejavu-sans-fonts	
Italian	dejavu-sans-fonts	
Russian	dejavu-sans-fonts	
Spanish	dejavu-sans-fonts	
Portuguese	dejavu-sans-fonts	
Simplified Chinese	google-noto-sans-cjk-ttc-fonts, google-noto-serif-cjk-ttc-fonts	ibus-libpinyin, libpinyin
Traditional Chinese	google-noto-sans-cjk-ttc-fonts, google-noto-serif-cjk-ttc-fonts	ibus-libzhuyin, libzhuyin
Japanese	google-noto-sans-cjk-ttc-fonts, google-noto-serif-cjk-ttc-fonts	ibus-kkc, libkkc
Korean	google-noto-sans-cjk-ttc-fonts, google-noto-serif-cjk-ttc-fonts	ibus-hangul, libhangul

12.2. NOTABLE CHANGES TO INTERNATIONALIZATION IN RHEL 8

RHEL 8 introduces the following changes to internationalization compared to RHEL 7:

- Support for the **Unicode 11** computing industry standard has been added.
- Internationalization is distributed in multiple packages, which allows for smaller footprint installations. For more information, see Using langpacks.

- A number of **glibc** locales have been synchronized with Unicode Common Locale Data Repository (CLDR).

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#2052528, BZ#2057063, BZ#2057066, BZ#2062679, BZ#1817505
NetworkManager	BZ#2097270, BZ#2082000, BZ#2063109, BZ#1943153, BZ#1920398, BZ#2132754
SLOF	BZ#1910848
accel-config	BZ#1843266
anaconda	BZ#1899494, BZ#1970726, BZ#2029101, BZ#2050140, BZ#1914955, BZ#1929105, BZ#2126506
ansible-collection-microsoft-sql	BZ#2066338
ansible-collection-redhat-rhel_mgmt	BZ#2112435
ansible-freeipa	BZ#2076554
apr	BZ#1819607
authselect	BZ#1892761
bacula	BZ#2089399
brltty	BZ#2008197
chrony	BZ#2062356
clevis	BZ#2107081
cloud-init	BZ#1750862
cockpit-appstream	BZ#2030836
cockpit	BZ#2056786, BZ#1666722
coreutils	BZ#2030661
corosync-qdevice	BZ#1784200

Component	Tickets
crash-ptdump-command	BZ#1838927
crash	BZ#1906482
createrepo_c	BZ#1973588
cronie	BZ#1832510
crypto-policies	BZ#1919155, BZ#1660839
cups-filters	BZ#2064606
device-mapper-multipath	BZ#2065477, BZ#2011699
distribution	BZ#2063772, BZ#1657927
dnf	BZ#2060815, BZ#1986657
dotnet7.0	BZ#2112096
dyninst	BZ#2057676
ec2-images	BZ#1862930
edk2	BZ#1741615, BZ#1935497
fapolicyd	BZ#2070639, BZ#2100087, BZ#2054741
fence-agents	BZ#1775847
firewalld	BZ#1871860
freeradius	BZ#1958979
frr	BZ#1714984
gcc-toolset-12-annobin	BZ#2077447
gcc-toolset-12-binutils	BZ#2077448
gcc-toolset-12-gcc	BZ#2077276
gcc-toolset-12-gdb	BZ#2077492
gdb	BZ#1853140

Component	Tickets
glibc	BZ#1888660, BZ#1982608, BZ#2065588, BZ#1961109, BZ#2089247, BZ#2091553, BZ#2104907, BZ#2033684, BZ#2096189, BZ#2077835
gnome-control-center	BZ#2079139
gnome-shell-extensions	BZ#1717947
gnome-software	BZ#1668760
gnutls	BZ#1628553
golang	BZ#2075162
grub2	BZ#2074762, BZ#1583445
grubby	BZ#1978226, BZ#1900829
initscripts	BZ#1875485
ipa	BZ#2059396, BZ#2022028, BZ#782917, BZ#2062379, BZ#1924707, BZ#2120572, BZ#2122919, BZ#1664719, BZ#1664718, BZ#2101770
ipmitool	BZ#1873614
iptables	BZ#2058444
kdump-anaconda-addon	BZ#2086100
kernel	BZ#2084242, BZ#2062870, BZ#2068429, BZ#2101938, JIRA:RHELPLAN-121252, BZ#2096127, BZ#2054656, BZ#1868526, BZ#1694705, BZ#1730502, BZ#1609288, BZ#1602962, BZ#1865745, BZ#1906870, BZ#1924016, BZ#1942888, BZ#1812577, BZ#1910358, BZ#1930576, BZ#2046396, BZ#1793389, BZ#1654962, BZ#1940674, BZ#1920086, BZ#1971506, BZ#2059262, BZ#2050411, BZ#2106341, BZ#2130159, BZ#1605216, BZ#1519039, BZ#1627455, BZ#1501618, BZ#1633143, BZ#1814836, BZ#1696451, BZ#1348508, BZ#1837187, BZ#1904496, BZ#1660337, BZ#1905243, BZ#1878207, BZ#1665295, BZ#1871863, BZ#1569610, BZ#1794513
kexec-tools	BZ#2006000
kmod-kvdo	BZ#2109047
kmod	BZ#2103605
krb5	BZ#2026462, BZ#2125182, BZ#1877991

Component	Tickets
libdnf	BZ#2088149
libgnome-keyring	BZ#1607766
libguestfs	BZ#1554735
libpfm	BZ#2067218
libreswan	BZ#1989050
libselinux-python-2.8-module	BZ#1666328
libva	BZ#2099907
libvirt	BZ#1664592, BZ#1332758, BZ#2067126, BZ#1528684
libvpd	BZ#2051319
llvm-toolset	BZ#2061042, BZ#2088315
lsvpd	BZ#2051316
lvm2	BZ#1496229, BZ#1768536
make43	BZ#2083419
mariadb-java-client	BZ#2043212
mariadb	BZ#1944653, BZ#1942330
maven	BZ#2083114
mercurial	BZ#2089849
mesa	BZ#1886147
motif	BZ#2060571
nfs-utils	BZ#1946283, BZ#2087187, BZ#2081114, BZ#1592011
nispor	BZ#1848817
nodejs	BZ#2083073

Component	Tickets
nss_nis	BZ#1803161
nss	BZ#1817533, BZ#1645153, BZ#2097837
open-vm-tools	BZ#2061193
opencryptoki	BZ#2043845
opencv	BZ#2104776, BZ#1886310
openmpi	BZ#1866402
opensc	BZ#1947025
openssh	BZ#2044354
openssl	BZ#1810911
osbuild-composer	BZ#2065734
oscap-anaconda-addon	BZ#2075508, BZ#1843932, BZ#1665082
pacemaker	BZ#2036815, BZ#2059638, BZ#1182956, BZ#1724310
papi	BZ#2037426, BZ#2071558, BZ#2037427, BZ#2037417
pcs	BZ#1786964, BZ#1954099, BZ#2023845, BZ#1950551, BZ#1909904, BZ#1874624, BZ#1619620, BZ#1847102, BZ#1851335
pki-core	BZ#1729215, BZ#2134093, BZ#1628987
podman	BZ#2097708, JIRA:RHELPLAN-77238
postfix	BZ#1711885
powerpc-utils	BZ#2078514, BZ#2051330
ppc64-diag	BZ#2051313
procps-ng	BZ#2111915
pykickstart	BZ#1637872
qemu-kvm	BZ#2043830, BZ#2020133, BZ#1740002, BZ#1719687, BZ#1966475, BZ#1792683, BZ#1651994

Component	Tickets
rear	BZ#2072978, BZ#2115918, BZ#2077404, BZ#2021935, BZ#2035872, BZ#1925531, BZ#1868421, BZ#2083301
redhat-support-tool	BZ#2064575, BZ#1802026
redis	BZ#1999873
restore	BZ#1997366
rhel-system-roles	BZ#2064067, BZ#2115884, BZ#2060377, BZ#2072749, BZ#2083378, BZ#2100285, BZ#2060378, BZ#2083426, BZ#2100298, BZ#2101607, BZ#2115161, BZ#2109997, BZ#2065339, BZ#2086869, BZ#1996731, BZ#2065670, BZ#2043009, BZ#2112143, BZ#2066876, BZ#2071011, BZ#2075116, BZ#2079008, BZ#2086934, BZ#2086935, BZ#2093437, BZ#2079114, BZ#2056480, BZ#2065215, BZ#2065216, BZ#2065218, BZ#2100297, BZ#2100939, BZ#2100979, BZ#2115159, BZ#2115160, BZ#2115162, BZ#2021685, BZ#2006081
rpm	BZ#1688849
rsync	BZ#2139118
rsyslog	BZ#1962318, BZ#1679512, JIRA:RHELPLAN-10431
rust-toolset	BZ#2075344
s390utils	BZ#1660911
samba	BZ#2077468, BZ#2009213, JIRA:RHELPLAN-13195, Jira:RHELDOCS-16612
sblim-wbemcli	BZ#2075807
scap-security-guide	BZ#2064696, BZ#2075384, BZ#2077531, BZ#2078974, BZ#2083109, BZ#2109602, BZ#2070564, BZ#2058203, BZ#1967947, BZ#2032403, BZ#2112937, BZ#2028428, BZ#1858866, BZ#1750755, BZ#2038977, BZ#2119356
selinux-policy	BZ#1461914
sos	BZ#2011413
spice	BZ#1849563
sssd	BZ#2065692, BZ#2056483, BZ#1947671

Component	Tickets
systemtap	BZ#2057565
ubi8-container	BZ#2120378
udica	BZ#1763210
unbound	BZ#2027735
vdo	BZ#1949163
virt-manager	BZ#2026985
vulkan-loader	BZ#2012639
wayland	BZ#1673073
weldr-client	BZ#2033192
xmlstarlet	BZ#1882020
xorg-x11-server	BZ#1698565
xorg-x11-xtrans-devel	BZ#2075132

Component	Tickets
other	JIRA:RHELPLAN-109067, JIRA:RHELPLAN-115603, BZ#2020301, BZ#2125545, BZ#2128016, JIRA:RHELPLAN-121982, JIRA:RHELPLAN-118463, JIRA:RHELPLAN-100037, BZ#1497089, JIRA:RHELPLAN-121981, JIRA:RHELPLAN-121983, JIRA:RHELPLAN-121980, BZ#2049492, JIRA:RHELPLAN-98420, JIRA:RHELPLAN-100039, JIRA:RHELPLAN-123369, JIRA:RHELPLAN-130379, JIRA:RHELPLAN-130376, JIRA:RHELPLAN-122735, BZ#2070793, BZ#2122716, JIRA:RHELPLAN-135602, JIRA:RHELPLAN-136150, BZ#2139821, BZ#2025814, BZ#2077770, BZ#1777138, BZ#1640697, BZ#1697896, BZ#1961722, BZ#1659609, BZ#1687900, BZ#1757877, BZ#1741436, JIRA:RHELPLAN-59111, JIRA:RHELPLAN-27987, JIRA:RHELPLAN-34199, JIRA:RHELPLAN-57914, JIRA:RHELPLAN-96940, BZ#1974622, BZ#2028361, BZ#2041997, BZ#2035158, JIRA:RHELPLAN-109613, BZ#2126777, BZ#1690207, JIRA:RHELPLAN-1212, BZ#1559616, BZ#1889737, JIRA:RHELPLAN-14047, BZ#1769727, JIRA:RHELPLAN-27394, JIRA:RHELPLAN-27737, BZ#1906489, JIRA:RHELPLAN-75165, JIRA:RHELPLAN-118470, JIRA:RHELPLAN-122316, BZ#1642765, JIRA:RHELPLAN-10304, BZ#1646541, BZ#1647725, BZ#1932222, BZ#1686057, BZ#1748980, JIRA:RHELPLAN-71200, BZ#1827628, JIRA:RHELPLAN-45858, BZ#1871025, BZ#1871953, BZ#1874892, BZ#1916296, JIRA:RHELPLAN-100400, BZ#1926114, BZ#1904251, BZ#2011208, JIRA:RHELPLAN-59825, BZ#1920624, JIRA:RHELPLAN-70700, BZ#1929173, JIRA:RHELPLAN-85066, BZ#2006665, JIRA:RHELPLAN-98983, BZ#2009113, BZ#1958250, BZ#2038929, BZ#2029338, BZ#2061288, BZ#2060759, BZ#2055826, BZ#2059626, JIRA:RHELPLAN-133171, BZ#2142499

APPENDIX B. REVISION HISTORY

0.2-6

Thu Feb 29 2024, Gabriela Fialová (gfialova@redhat.com)

- Added a deprecated functionality JIRA:RHELDOCS-17641 (Networking).

0.2-5

Tue Feb 13 2024, Lucie Vařáková (lvarakova@redhat.com)

- Added a deprecated functionality JIRA:RHELDOCS-17573 (Identity Management)

0.2-4

Fri Feb 2 2024, Lucie Vařáková (lvarakova@redhat.com)

- Added a known issue BZ#1834716 (Security).

0.2-3

Fri Nov 10 2023, Gabriela Fialová (gfialova@redhat.com)

- Updated the module on Providing Feedback on RHEL Documentation.

0.2-2

Fri Oct 13 2023, Gabriela Fialová (gfialova@redhat.com)

- Added a Tech Preview JIRA:RHELDOCS-16861 (Containers).

0.2-1

October 9 2023, Lucie Vařáková (lvarakova@redhat.com)

- Updated a known issue BZ#2169382 (Networking).

0.2-0

Fri September 8 2023, Lucie Vařáková (lvarakova@redhat.com)

- Added a deprecated functionality release note JIRA:RHELDOCS-16612 (Samba).
- Updated the Providing feedback on Red Hat documentation section.

0.1-9

Thu August 24 2023, Lucie Vařáková (lvarakova@redhat.com)

- Added a known issue BZ#2214508 (Kernel).

0.1-8

Fri August 4 2023, Lenka Špačková (lspackova@redhat.com)

- Fixed section for BZ#2225332.

0.1-7

Tue August 1 2023, Lenka Špačková (lspackova@redhat.com)

- Added deprecated functionality BZ#2225332.
- Improved abstract.

0.1-6

Mon Jul 17 2023, Gabriela Fialová (gfialova@redhat.com)

- Fixed a mistake in BZ#2072749 (System Roles).

0.1-5

Thu Jun 29 2023, Marc Muehlfeld (mmuehlfeld@redhat.com)

- Added a Technology Preview BZ#1570255 (Kernel).

0.1-4

Fri Jun 16 2023, Lucie Vařáková (lvarakova@redhat.com)

- Added a known issue BZ#2214235 (Kernel).

0.1-3

Thu Jun 15 2023, Lucie Vařáková (lvarakova@redhat.com)

- Added an enhancement BZ#2070347 (Boot loader).

0.1-3

Thu May 18 2023, Gabriela Fialová (gfialova@redhat.com)

- Added an enhancement BZ#2083077 (File systems and storage).

0.1-2

Wed May 10 2023, Jaroslav Klech (jklech@redhat.com)

- Added a known issue BZ#2169382 (Networking).

0.1-1

Thu Apr 27 2023, Gabriela Fialová (gfialova@redhat.com)

- Added a known issue JIRA:RHELPLAN-155168 (Identity Management).

0.1-0

Tue Apr 18 2023, Lenka Špačková (lspackova@redhat.com)

- Added an enhancement from an asynchronous update BZ#2178087 (Dynamic programming languages, web and database servers).

0.0-9

Thu Apr 13 2023, Gabriela Fialová (gfialova@redhat.com)

- Fixed 2 broken links in DFs and KIs.

0.0-8

Feb 17 2023, Lucie Vařáková (lvarakova@redhat.com)

- Added a known issue BZ#2158783 (File systems and storage).

0.0-7

Feb 14 2023, Marc Muehlfeld (mmuehlfeld@redhat.com)

- Added an enhancement BZ#2144898 (Networking).

0.0-6

Feb 08 2023, Lucie Vařáková (lvarakova@redhat.com)

- Added bug fixes BZ#2046396 (Kernel) and BZ#2069379 (Identity Management).
- Other minor updates.

0.0-5

Jan 24 2023, Lucie Vařáková (lvarakova@redhat.com)

- Added a known issue BZ#2115791 (RHEL in cloud environments).

0.0-4

Jan 18 2023, Lucie Vařáková (lvarakova@redhat.com)

- Added a known issue BZ#1920086 (Kernel).

0.0-3

Dec 07 2022, Lucie Vařáková (lvarakova@redhat.com)

- Moved the **nodejs:18** module stream BZ#2083073 from Technology Previews to fully supported features (Dynamic programming languages, web and database servers).
- Added a known issue BZ#2132754 (Networking).

0.0-2

Nov 23 2022, Gabriela Fialová (gfialova@redhat.com)

- Release of Directory Server RNs and subsequent republishing of RHEL 8.7 RNs.

0.0-1

Nov 09 2022, Lucie Vařáková (lvarakova@redhat.com)

- Release of the Red Hat Enterprise Linux 8.7 Release Notes.

0.0-0

Sep 28 2022, Lucie Vařáková (lvarakova@redhat.com)

- Release of the Red Hat Enterprise Linux 8.7 Beta Release Notes.