Red Hat Enterprise Linux 8.6

8.6 Release Notes

Release Notes for Red Hat Enterprise Linux 8.6

Last Updated: 2024-04-29

Red Hat Enterprise Linux 8.6 8.6 Release Notes

Release Notes for Red Hat Enterprise Linux 8.6

Legal Notice

Abstract

The Release Notes provide high-level coverage of the improvements and additions that have been implemented in Red Hat Enterprise Linux 8.6 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 8.6

Security

In RHEL 8.6, SELinux, the **fapolicyd** framework, and Policy-Based Decryption (PBD) for automated unlocking of LUKS-encrypted drives support the SAP HANA database management system. See the Red Hat Enterprise Linux Security Hardening Guide for SAP HANA 2.0 Knowledgebase article for more information.

Packages for **fapolicyd** have been upgraded to the upstream version 1.1. Among other improvements, you can now use the new **rules.d/** and **trust.d/** directories, the **fagenrules** script, and new options for the **fapolicyd-cli** command.

OpenSSH servers now support drop-in configuration files.

The **pcsc-lite** packages have been rebased to upstream version 1.9.5, which provides many enhancements and bug fixes.

You can now verify the versions of installed SELinux policy modules with the newly added **--checksum** option to the **semodule** command.

The SCAP Security Guide (SSG) packages have been rebased to upstream version 0.1.60, and the OpenSCAP packages have been rebased to upstream version 1.3.6.

See New features - Security for more information.

Dynamic programming languages, web and database servers

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

- **PHP 8.0**
- **Perl 5.32**

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

Compilers and development tools

Updated compiler toolsets

The following compiler toolsets have been updated:

- **GCC Toolset 11**
- **LLVM Toolset 13.0.1**
- **Rust Toolset 1.58.1**
- **Go Toolset 1.17.7**

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 8.6 introduces a new **log4j:2** module, which contains **Apache Log4j 2**, which is a Java logging utility and a library enabling you to output log statements to a variety of output targets.

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

Identity Management
The **ansible-freeipa** roles and modules are now available in the Ansible Automation Hub, which provides fast updates of the **ansible-freeipa** content.

1.2. IN-PLACE UPGRADE AND OS CONVERSION

In-place upgrade from RHEL 7 to RHEL 8
The supported 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 with SAP HANA remains supported after upgrading to RHEL 8.2, enable the RHEL 8.2 Update Services for SAP Solutions (E4S) repositories.

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.

Notable enhancements include:

- With the release of RHEL 8.6, multiple upgrade paths are now available for the in-place upgrade from RHEL 7 to RHEL 8. This allows you to decide which RHEL 8 minor version you want to upgrade your system to instead of upgrading to the latest RHEL 8 minor version by default. Note that the available upgrade paths differ between RHEL systems and RHEL systems with SAP HANA.
- The **Leapp** utility now runs significantly faster during the pre-upgrade and the initial stages of the in-place upgrade.
- The in-place upgrade is also supported for SAP hosting systems for the following cloud image types:
  - Bring-your-own-subscription (BYOS) systems on any public cloud platform which uses Red Hat Subscription Manager (RHSM) for a RHEL subscription.

- Pay-as-you-go (PAYG) instances on Amazon Web Services (AWS) and Microsoft Azure 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 8.6 is distributed with the kernel version 4.18.0-372, 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 Binary DVD ISO image is larger than 4.7 GB, and as a result, it might not fit on a single-layer DVD. A dual-layer DVD or USB key is recommended when using the Binary DVD 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.

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 8.6.

4.1. INSTALLER AND IMAGE CREATION

Image Builder supports customized file system partition on LVM

With this enhancement, if you have more than one partition, you can create images with a customized file system partition on LVM and resize those partitions at runtime. For that, you can specify a customized filesystem configuration in your blueprint and then create images with the desired disk layout. The default filesystem layout remains unchanged - if you use plain images without file system customization, the root partition is resized by **cloud-init**.

(JIRA:RHELPLAN-102505)

4.2. RHEL FOR EDGE

RHEL for Edge now supports Greenboot built-in health checks by default

With this update, RHEL for Edge **Greenboot** now includes built-in health checks with **watchdog** feature to ensure that the hardware does not hang or freeze while rebooting. With that, you can benefit from the following features:

- It makes it simple for **watchdogs** hardware users to adopt the built-in health checks
- A set of default health checks that provide value for built-in OS components
- The **watchdog** is now present as default presets, which makes it easy to enable or disable this feature
- Ability to create custom health checks based on the already available health checks.

(BZ#2083036)

RHEL 8 rebased to rpm-ostree v2022.2

RHEL 8 is distributed with the **rpm-ostree** version v2022.2, which provides multiple bug fixes and enhancements. Notable changes include:

- Kernel arguments can now be updated in an idempotent way, by using the new **--append-if-missing** and **--delete-if-present** kargs flags.
- The **Count Me** feature from YUM is now fully disabled by default in all repo queries and will only be triggered by the corresponding **rpm-ostree-countme.timer** and **rpm-ostree-countme.service** units. See countme.
- The post-processing logic can now process the **user.ima** IMA extended attribute. When an **xattr** extended attribute is found, the system automatically translates it to **security.ima** in the final **OSTree** package content.
- The **treefile** file has a new **repo-packages** field. You can use it to pin a set of packages to a specific repository.
- Ability to use modularity on the compose and client side.

- Container images are now used as a compose target and also as an upgrade source.

(BZ#2032594)

4.3. SUBSCRIPTION MANAGEMENT

Merged system purpose commands under subscription-manager syspurpose

Previously, there were multiple subscription-manager modules (**addons**, **role**, **service-level**, and **usage**) for setting attributes related to system purpose. These modules have been moved under the new **subscription-manager syspurpose** module.

The original subscription-manager modules (**addons**, **role**, **service-level**, and **usage**) are now deprecated. Additionally, the package (**python3-syspurpose**) that provides the **syspurpose** command line tool has been deprecated in RHEL 8.6. All the capabilities of this package are covered by the new **subscription-manager syspurpose** module.

This update provides a consistent way to view, set, and update all system purpose attributes using a single command of subscription-manager; this replaces all the existing system purpose commands with their equivalent versions available as a new subcommand. For example, **subscription-manager role --set SystemRole** becomes **subscription-manager syspurpose role --set SystemRole** and so on.

For complete information about the new commands, options, and other attributes, see the **SYSPURPOSE OPTIONS** section in the **subscription-manager** man page.

(BZ#2000883)

4.4. SOFTWARE MANAGEMENT

The modulesync command is now available to replace certain workflows in RHEL 8

In Red Hat Enterprise Linux 8, modular packages cannot be installed without modular metadata. Previously, you could use the **yum** command to download packages, and then use the **createrepo_c** command to redistribute those packages.

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

(BZ#1868047)

A new --path CLI option is added to RPM

With this update, you can query packages by a file that is currently not installed using a new **--path** CLI option. This option is similar to the existing **--file** option, but matches packages solely based on the provided path. Note that the file at that path does not need to exist on disk.

The **--path** CLI option can be useful when a user excludes all documentation files at install time by using the **--nodocs** option with **yum**. In this case, by using the **--path** option, you can display the owning package of such an excluded file, whereas the **--file** option will not display the package because the requested file does not exist.

(BZ#1940895)

4.5. SHELLS AND COMMAND-LINE TOOLS

The lsvpd package rebased to version 1.7.13

The **lsvpd** package has been rebased to version 1.7.13. Notable bug fixes and enhancements include:

- Added support for SCSI location code.
- Fixed length of absolute path **getDevTreePath** in **sysfstreecollector**.

(BZ#1993557)

The net-snmp-cert gencert tool now uses the SHA512 encryption algorithm instead of SHA1

In order to increase security, the **net-snmp-cert gencert** tool has been updated to generate certificates using SHA512 encryption algorithm by default.

(BZ#1908331)

The dnn and text modules are available in the opencv package

The **dnn** module containing Deep Neural Networks for image classification inference and the **text** module for scene text detection and recognition are now available in the **opencv** package.

(BZ#2007780)

The powerpc-utils package rebased to version 1.3.9

The **powerpc-utils** package has been upgraded to version 1.3.9. Notable bug fixes, and enhancements include:

- Increased log size to 1MB in **drmgr**.
- Fixed checking **HCNID** array size at boot time.
- Implemented **autoconnect-slaves** on HNV connections in **hcnmgr**.
- Improved the HNV bond list connections in **hcnmgr**.
- Uses **hexdump** from **util-linux** instead of **xxd** from **vim** in **hcnmgr**.
- The **hcn-init.service** starts together with NetworkManager.
- Fixed OF to logical FC lookup for multipath in **ofpathname**.
- Fixed OF to logical lookup with partitions in **ofpathname**.
- Fixed bootlist for multipath devices with more than 5 paths.
- Introduced **lparnumascore** command to detect the NUMA affinity score for the running LPAR.
- Added the **-x** option in **lpartstat** to enhance security.
- Fixed **ofpathname** race with **udev** rename in **hcnmgr**.
- Fixed **qrydev** in HNV, and removed **lsdevinfo**.

(BZ#2028690)

The powerpc-utils package now supports vNIC as a backup device

The **powerpc-utils** package now supports Virtual Network Interface cards (vNIC) as a backup **vdevice** for Hybrid Network Virtualization (HNV).

(BZ#2022225)

The opencryptoki package rebased to version 3.17.0

The **opencryptoki** package has been rebased to version 3.17.0. Notable bug fixes and enhancements include:

- The **p11sak** tool offers a new function of listing keys.
- Added support for **OpenSSL 3.0**.
- Added support for event notifications.
- Added SW fallbacks in ICA tokens.
- The WebSphere Application Server no longer fails to start with the hardware crypto adapter enabled.
- The **opencryptoki.module** was removed, and the **p11-kit list-modules** command no longer causes error messages.

(BZ#1984993)

Certain network interfaces and IP addresses can be excluded when creating a rescue image

You can use the **EXCLUDE_IP_ADDRESSES** variable to ignore certain IP addresses, and the **EXCLUDE_NETWORK_INTERFACES** variable to ignore certain network interfaces when creating a rescue image.

On servers with floating addresses, you need to stop the ReaR rescue environment from configuring floating addresses that are moved to a fail-over server until the original server is recovered. Otherwise, a conflict with the fail-over server would occur and cause a consequent disruption of the services running on the fail-over server. To prevent conflicts, you can perform the following actions in the ReaR configuration file **/etc/rear/local.conf**:

- exclude the IP addresses in the ReaR by providing the **EXCLUDE_IP_ADDRESSES** variable as a bash array of addresses. For example: **EXCLUDE_IP_ADDRESSES=(192.0.2.27 192.0.2.10)**,
- exclude the network interfaces in the ReaR by providing the **EXCLUDE_NETWORK_INTERFACES** variable as a bash array of interfaces. For example: **EXCLUDE_NETWORK_INTERFACES=(eno1d1)**.

(BZ#2035939)

4.6. INFRASTRUCTURE SERVICES

New bind9.16 package version 9.16.23 introduced

A new **bind9.16** package version 9.16.23 has been introduced as an alternative to **bind** component version 9.11.36. Notable enhancements include:

- Introduced new Key and Signing Policy feature in DNSSEC.

- Introduced the QNAME minimisation to improve privacy.
- Introduced the **validate-except** feature to Permanent.
- Negative Trust Anchors to temporarily disable DNSSEC validation.
- Refactored the response policy zones (RPZ).
- Introduced new naming conventions for zone types: *primary* and *secondary* zone types are used as synonyms to *master* and *slave*.
- Introduced a supplementary YAML output mode of **dig**, **mdig**, and **delv** commands.
- The **filter-aaaa** functionality was moved into separate **filter-a** and **filter-aaaa** plugins.
- Introduced a new zone type mirror support (RFC 8806).

Removed features:

- The **dnssec-enabled** option has been removed, DNSSEC is enabled by default, and the *dnssec-enabled* keywords are no longer accepted.
- The **lwresd** lightweight resolver daemon, and **liblwres** lightweight resolver library have been removed.

(BZ#1873486)

CUPS is available as a container image

The Common Unix Printing System (CUPS) is now available as a container image, and you can deploy it from the Red Hat Container Catalog.

(BZ#1913715)

The bind component rebased to version 9.11.36

The **bind** component has been updated to version 9.11.36. Notable bug fixes and enhancements include:

- Improved the **lame-ttl** option to be more secure.
- A multiple threads bug affecting RBTDB instances no longer results in assertion failure in **free_rbtdb()**.
- Updated implementation of the ZONEMD RR type to match RFC 8976.
- The maximum supported number of NSEC3 iterations has been reduced to 150. Records with more iterations are treated as insecure.
- An invalid direction field in a LOC record no longer results in a failure.

(BZ#2013993)

CUPS driverless printing is available in CUPS Web UI

CUPS driverless printing, based on the IPP Everywhere model, is available in the CUPS Web UI. In addition to the **lpadmin** command used in the CLI, you can create an IPP Everywhere queue in the CUPS Web UI to print to network printers without special software.

(BZ#2032965)

4.7. SECURITY

The pcsc-lite packages rebased to 1.9.5

The **pcsc-lite** packages have been rebased to upstream version 1.9.5. This update provides new enhancements and bug fixes, most notably:

- The **pcscd** daemon no longer automatically exits after inactivity when started manually.
- The **pcsc-spy** utility now supports Python 3 and a new **--thread** option.
- Performance of the **SCardEndTransaction()** function has been improved.
- The **poll()** function replaced the **select()** function, which allows file descriptor numbers higher than **FD_SETSIZE**.
- Many memory leaks and concurrency problems have been fixed.

(BZ#2014641)

Crypto policies support diffie-hellman-group14-sha256

You can now use the **diffie-hellman-group14-sha256** key exchange (KEX) algorithm for the **libssh** library in RHEL system-wide cryptographic policies. This update also provides parity with OpenSSH, which also supports this KEX algorithm. With this update, **libssh** has **diffie-hellman-group14-sha256** enabled by default, but you can disable it by using a custom crypto policy.

(BZ#2023744)

OpenSSH servers now support drop-in configuration files

The **sshd_config** file supports the **Include** directive, which means you can include configuration files in another directory. This makes it easier to apply system-specific configurations on OpenSSH servers by using automation tools such as Ansible Engine. It is also more consistent with the capabilities of the **ssh_config** file. In addition, drop-in configuration files also make it easier to organize different configuration files for different uses, such as filter incoming connections.

(BZ#1926103)

sshd_config:ClientAliveCountMax=0 disables connection termination

Setting the SSHD configuration option **ClientAliveCountMax** to **0** now disables connection termination. This aligns the behavior of this option with the upstream. As a consequence, OpenSSH no longer disconnects idle SSH users when it reaches the timeout configured by the **ClientAliveInterval** option.

(BZ#2015828)

libssh rebased to 0.9.6

The **libssh** package has been rebased to upstream version 0.9.6. This version provides bug fixes and enhancements, most notably:

- Support for multiple identity files. The files are processed from the bottom to the top as listed in the **~/.ssh/config** file.
- Parsing of sub-second times in SFTP is fixed.

- A regression of the **ssh_channel_poll_timeout()** function returning **SSH_AGAIN** unexpectedly is now fixed.
- A possible heap-buffer overflow after key re-exchange is fixed.
- A handshake bug when AEAD cipher is matched but there is no HMAC overlap is fixed.
- Several memory leaks on error paths are fixed.

(BZ#1896651)

Libreswan rebased to 4.5

Libreswan has been rebased to upstream version 4.5. This version provides many bug fixes and enhancements, most notably:

- Support of Internet Key Exchange version 2 (IKEv2) for Labeled IPsec.
- Support for childless initiation of Internet Key Exchange (IKE) Security Association (SA).

(BZ#2017352)

New option to verify SELinux module checksums

With the newly added **--checksum** option to the **semodule** command, you can verify the versions of installed SELinux policy modules.

Because Common Intermediate Language (CIL) does not store module name and module version in the module itself, there previously was no simple way to verify that the installed module is the same version as the module which was supposed to be installed.

With the new command **semodule -l --checksum**, you receive a SHA256 hash of the specified module and can compare it with the checksum of the original file, which is faster than reinstalling modules.

Example of use:

```
# semodule -l --checksum | grep localmodule
localmodule sha256:db002f64ddfa3983257b42b54da7b182c9b2e476f47880ae3494f9099e1a42bd

# /usr/libexec/selinux/hll/pp localmodule.pp | sha256sum
db002f64ddfa3983257b42b54da7b182c9b2e476f47880ae3494f9099e1a42bd  -
```

(BZ#1731501)

OpenSCAP can read local files

OpenSCAP can now consume local files instead of remote SCAP source data stream components. Previously, you could not perform a complete evaluation of SCAP source data streams containing remote components on systems that have no internet access. On these systems, OpenSCAP could not evaluate some of the rules in these data streams because the remote components needed to be downloaded from the internet. With this update, you can download and copy the remote SCAP source data stream components to the target system before performing the OpenSCAP scan and provide them to OpenSCAP by using the **--local-files** option with the **oscap** command.

(BZ#1970529)

SSG now scans and remediates rules for home directories and interactive users

OVAL content to check and remediate all existing rules related to home directories used by interactive users was added to the SCAP Security Guide (SSG) suite. Many benchmarks require verification of properties and content usually found within home directories of interactive users. Because the existence and the number of interactive users in a system may vary, there was previously no robust solution to cover this gap using the OVAL language. This update adds OVAL checks and remediations that detect local interactive users in a system and their respective home directories. As a result, SSG can safely check and remediate all related benchmark requirements.

(BZ#1884687)

SCAP rules now have a warning message to configure Audit log buffer for large systems

The SCAP rule **xccdf_org.ssgproject.content_rule_audit_basic_configuration** now displays a performance warning that suggests users of large systems where the Audit log buffer configured by this rule might be too small and can override the custom value. The warning also describes the process to configure a larger Audit log buffer. With this enhancement, users of large systems can stay compliant and have their Audit log buffer set correctly.

(BZ#1993826)

SSG now supports the /etc/security/faillock.conf file

This enhancement adds support for the **/etc/security/faillock.conf** file in SCAP Security Guide (SSG). With this update, SSG can assess and remediate the **/etc/security/faillock.conf** file for definition of **pam_faillock** settings. The **authselect** tool is also used to enable the **pam_faillock** module while ensuring the integrity of **pam** files. As a result, the assessment and remediation of the **pam_faillock** module is aligned with the latest versions and best practices.

(BZ#1956972)

SCAP Security Guide rebased to 0.1.60

The SCAP Security Guide (SSG) packages have been rebased to upstream version 0.1.60. This version provides various enhancements and bug fixes, most notably:

- Rules hardening the PAM stack now use **authselect** as the configuration tool.
- Tailoring files that define profiles which represent the differences between DISA STIG automated SCAP content and SCAP automated content (delta tailoring) are now supported.
- The rule **xccdf_org.ssgproject.content_enable_fips_mode** now checks only whether the FIPS mode has been enabled properly. It does not guarantee that system components have undergone FIPS certification.

(BZ#2014485)

DISA STIG profile supports Red Hat Virtualization 4.4

The **DISA STIG for Red Hat Enterprise Linux 8** profile version V1R5 has been enhanced to support Red Hat Virtualization 4.4. This profile aligns with the RHEL 8 Security Technical Implementation Guide (STIG) manual benchmark provided by the Defense Information Systems Agency (DISA). However, some configurations are not applied on hosts where Red Hat Virtualization (RHV) is installed because they prevent Red Hat Virtualization from installing and working properly.

When the STIG profile is applied on a Red Hat Virtualization Host (RHVH), on a self-hosted install (RHELH), or on a host with RHV Manager installed, the following rules result in 'notapplicable':

- **package_gss_proxy_removed**

- **package_krb5-workstation_removed**
- **package_tuned_removed**
- **sshd_disable_root_login**
- **sudo_remove_nopasswd**
- **sysctl_net_ipv4_ip_forward**
- **xwindows_remove_packages**

WARNING

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

(BZ#2021802)

OpenSCAP rebased to 1.3.6

The OpenSCAP packages have been rebased to upstream version 1.3.6. This version provides various bug fixes and enhancements, most notably:

- You can provide local copies of remote SCAP source data stream components by using the **--local-files** option.
- OpenSCAP accepts multiple **--rule** arguments to select multiple rules on the command line.
- OpenSCAP allows skipping evaluation of some rules using the **--skip-rule** option.
- You can restrict memory consumed by OpenSCAP probes by using the **OSCAP_PROBE_MEMORY_USAGE_RATIO** environment variable.
- OpenSCAP now supports the OSBuild Blueprint as a remediation type.

(BZ#2041781)

clevis-systemd no longer depends on nc

With this enhancement, the **clevis-systemd** package no longer depends on the **nc** package. The dependency did not work correctly when used with Extra Packages for Enterprise Linux (EPEL).

(BZ#1949289)

audit rebased to 3.0.7

The **audit** packages have been upgraded to version 3.0.7 which introduces many enhancements and bug fixes. Most notably:

- Added **sudoers** to Audit base rules.

- Added the **--eoe-timeout** option to the **ausearch** command and its analogous **eoe_timeout** option to **auditd.conf** file that specifies the value for end of event timeout, which impacts how **ausearch** parses co-located events.
- Introduced a fix for the 'audisp-remote' plugin that used 100% of CPU capacity when the remote location was not available.

(BZ#1939406)

Audit now provides options for specifying the end of the event timeout

With this release, the **ausearch** tool supports the **--eoe-timeout** option, and the **auditd.conf** file contains the **end_of_event_timeout** option. You can use these options to specify the end of the event timeout to avoid problems with parsing co-located events. The default value for the end of the event timeout is set to two seconds.

(BZ#1921658)

Adding sudoers to Audit base rules

With this enhancement, the **/etc/sudoers** and the **etc/sudoers.d/** directories are added to Audit base rules such as the Payment Card Industry Data Security Standard (PCI DSS) and the Operating Systems Protection Profile (OSPP). This increases the security by monitoring configuration changes in privileged areas such as **sudoers**.

(BZ#1927884)

Rsyslog includes the mmfields module for higher-performance operations and CEF

Rsyslog now includes the **rsyslog-mmfields** subpackage which provides the **mmfields** module. This is an alternative to using the property replacer field extraction, but in contrast to the property replacer, all fields are extracted at once and stored inside the structured data part. As a result, you can use **mmfields** particularly for processing field-based log formats, for example Common Event Format (CEF), and if you need a large number of fields or reuse specific fields. In these cases, **mmfields** has better performance than existing Rsyslog features.

(BZ#1947907)

libcap rebased to version 2.48

The **libcap** packages have been upgraded to upstream version 2.48, which provides a number of bug fixes and enhancements over the previous version, most notably:

- Helper library for POSIX semantic system calls (**libpsx**)
- Support for overriding system call functions
- IAB abstraction for capability sets
- Additional **capsh** testing features

(BZ#2032813)

fapolicyd rebased to 1.1

The **fapolicyd** packages have been upgraded to the upstream version 1.1, which contains many improvements and bug fixes. Most notable changes include the following:

- 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. See the new **fagenrules(8)** man page for more details.

- In addition to the **/etc/fapolicyd/fapolicyd.trust** file for marking files outside of the RPM database as trusted, you can now use the new **/etc/fapolicyd/trust.d** directory, which supports separating a list of trusted files into more files. You can also add an entry for a file by using the **fapolicyd-cli -f** subcommand with the **--trust-file** directive to these files. See the **fapolicyd-cli(1)** and **fapolicyd.trust(13)** man pages for more information.
- The **fapolicyd** trust database now supports white spaces in file names.
- **fapolicyd** now stores the correct path to an executable file when it adds the file to the trust database.

(BZ#1939379)

libseccomp rebased to 2.5.2

The **libseccomp** packages have been rebased to upstream version 2.5.2. This version provides bug fixes and enhancements, most notably:

- Updated the syscall table for Linux to version **v5.14-rc7**.
- Added the **get_notify_fd()** function to the Python bindings to get the notification file descriptor.
- Consolidated multiplexed syscall handling for all architectures into one location.
- Added multiplexed syscall support to the PowerPC (PPC) and MIPS architectures.
- Changed the meaning of the **SECCOMP_IOCTL_NOTIF_ID_VALID** operation within the kernel.
- Changed the **libseccomp** file descriptor notification logic to support the kernel's previous and new usage of **SECCOMP_IOCTL_NOTIF_ID_VALID**.

(BZ#2019893)

4.8. NETWORKING

CleanUpModulesOnExit firewalld global configuration option is now available

Previously, when restarting or otherwise shutting down **firewalld**, **firewalld** recursively unloaded kernel modules. As a result, other packages attempting to use these modules or dependent modules would fail. With this upgrade, users can set the **CleanUpModulesOnExit** option to **no** to stop **firewalld** from unloading these kernel modules.

(BZ#1980206)

Restoring large nftables sets requires less memory

With this enhancement, the **nftables** framework requires significantly less memory when you restore large sets. The algorithm which prepares the **netlink** message has been improved, and, as a result, restoring a set can use up to 40% less memory.

(BZ#2047821)

The nmstate API now supports OVS-DPDK

This enhancement adds the schema for the Open vSwitch (OVS) Data Plane Development Kit (DPDK) to the **nmstate** API. As a result, you can use **nmstate** to configure OVS devices with DPDK ports.

(BZ#2003976)

The nmstate API now supports VLAN and QoS ID in SR-IOV virtual functions

This update enhances the **nmstate** API with support for local area network (VLAN) and quality of service (QoS) in single root I/O virtualization (SR-IOV) virtual functions. As a result, you can use **nmstate** to configure these features.

(BZ#2004006)

NetworkManager rebased to version 1.36.0

The **NetworkManager** packages have been upgraded to upstream version 1.36.0, which provides a number of enhancements and bug fixes over the previous version:

- The handling of layer 3 configurations has been reworked to improve the stability, performance, and memory usage.
- NetworkManager now supports the **rd.znet_ifnames** kernel command line option on the IBM Z platform.
- The **blackhole**, **unreachable**, and **prohibit** route types have been added.
- NetworkManager now ignores routes managed by routing services.
- The Wi-Fi Protected Access version 3 (WPA3) network security has been improved by enabling the hash-to-element (H2E) method when generating simultaneous authentication of equals (SAE) password elements.
- The service now correctly handles replies from DHCP servers that send duplicate address or mask options.
- You can now turn off MAC aging on bridges.
- NetworkManager no longer listens for **netlink** events for traffic control objects, such as **qdiscs** and **filters**.
- Network bonds now support setting a queue ID for bond ports.

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

- NetworkManager 1.36.0
- NetworkManager 1.34.0

(BZ#1996617)

The hostapd package has been added to RHEL 8.6

With this release, RHEL provides the **hostapd** package. However, Red Hat supports **hostapd** only to set up a RHEL host as an 802.1X authenticator in Ethernet networks. Other scenarios, such as Wi-Fi access points or authenticators in Wi-Fi networks, are not supported.

For details about configuring RHEL as an 802.1X authenticator with a FreeRADIUS back end, see Setting up an 802.1x network authentication service for LAN clients using hostapd with FreeRADIUS backend.

(BZ#2016946)

NetworkManager now supports setting the number of receiving queues (rx_queue) on OVS-DPDK interfaces

With this enhancement, you can use NetworkManager to configure the **n_rxq** setting of Open vSwitch (OVS) Data Plane Development Kit (DPDK) interfaces. Use the **ovs-dpdk.n-rxq** attribute in NetworkManager to set the number of receiving queues on OVS-DPDK interfaces.

For example, to configure 2 receiving queues in OVS interface named **ovs-iface0**, enter:

```
# nmcli connection modify ovs-iface0 ovs-dpdk.nrxq 2
```

(BZ#2001563)

The nftables framework now supports nft set elements with attached counters

Previously, in the **netfilter** framework, **nftables** set counters were not supported. The **nftables** framework is configurable by the **nft** tool. The kernel allows this tool to count the network packets from a given source address with a statement **add @myset {ip saddr counter}**. In this update, you can count packets that match a specific criteria with a dynamic set and elements with attached counters.

(BZ#1983635)

The nispor packages are now fully supported

The **nispor** packages, previously available as a Technology Preview, are now fully supported. This enhancement adds support for **NetStateFilter** to use the kernel filter on network routes and interfaces.

With this release, the **nispor** packages single Root Input and Output Virtualization (SR-IOV) interfaces can query SR-IOV Virtual Function (SR-IOV VF) information per (VF), support new bonding options: **lacp_active**, **arp_missed_max**, and **ns_ip6_target**.

(BZ#1848817)

4.9. KERNEL

Kernel version in RHEL 8.6

Red Hat Enterprise Linux 8.6 is distributed with the kernel version 4.18.0-372.

See also Important changes to external kernel parameters and Device Drivers.

(BZ#1839151)

Extended Berkeley Packet Filter for RHEL 8.6

The **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 executes a special assembly-like code.

The **eBPF** bytecode first loads to the kernel, followed by its verification, code translation to the native machine code with just-in-time compilation, and then the virtual machine executes the code.

Red Hat ships numerous components that utilize the **eBPF** virtual machine. Each component is in a different development phase, and thus not all components are currently fully supported. In RHEL 8.6, the following **eBPF** components are supported:

- The **BPF Compiler Collection (BCC)** tools package, which provides tools for I/O analysis, networking, and monitoring of Linux operating systems using **eBPF**.
- The **BCC** library which allows the development of tools similar to those provided in the **BCC** tools package.
- The **eBPF for Traffic Control (tc)** feature, which enables programmable packet processing inside the kernel network data path.
- The **bpftrace** tracing language
- The **eXpress Data Path (XDP)** feature, which provides access to received packets before the kernel networking stack processes them, is supported under specific conditions. For more information see, XDP is conditionally supported and Overview of networking eBPF features in RHEL.
- The **libbpf** package, which is crucial for bpf related applications like **bpftrace** and **bpf/xdp** development.
- The **xdp-tools** package, which contains userspace support utilities for the **XDP** feature, is now supported on the AMD and Intel 64-bit architectures. This includes the **libxdp** library, the **xdp-loader** utility for loading XDP programs, the **xdp-filter** example program for packet filtering, and the **xdpdump** utility for capturing packets from a network interface with XDP enabled.

Note that all other **eBPF** components are available as Technology Preview, unless a specific component is indicated as supported.

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

- The **AF_XDP** socket for connecting the **eXpress Data Path (XDP)** path to user space

For more information regarding the Technology Preview components, see **eBPF** available as a Technology Preview.

(BZ#1780124)

Red Hat, by default, enables eBPF in all RHEL versions for privileged users only

Extended Berkeley Packet Filter (**eBPF**) is a complex technology which allows users to execute custom code inside the Linux kernel. Due to its nature, the **eBPF** code needs to pass through the verifier and other security mechanisms. There were Common Vulnerabilities and Exposures (CVE) instances, where bugs in this code could be misused for unauthorized operations. To mitigate this risk, Red Hat by default enabled **eBPF** in all RHEL versions for privileged users only. It is possible to enable **eBPF** for unprivileged users by using the kernel.command-line parameter **unprivileged_bpf_disabled=0**.

However, note that:

- Applying **unprivileged_bpf_disabled=0** disqualifies your kernel from Red Hat support and opens your system to security risks.
- Red Hat urges you to treat processes with the **CAP_BPF** capability as if the capability was equal to **CAP_SYS_ADMIN**.

- Setting **unprivileged_bpf_disabled=0** will not be sufficient to execute many BPF programs by unprivileged users as loading of most BPF program types requires additional capabilities (typically **CAP_SYS_ADMIN** or **CAP_PERFMON**).

For information on how to apply kernel command-line parameters, see Configuring kernel command-line parameters.

(BZ#2089409)

The osnoise and timerlat tracers were added in RHEL 8

The **osnoise** tracer measures operating system noise. That is, the interruptions of applications by the OS and hardware interrupts. It also provides a set of tracepoints to help find the source of the OS noise. The **timerlat** tracer measures the wakeup latencies and helps to identify the causes of such latencies of real-time (RT) threads. In RT computing, latency is absolutely crucial and even a minimal delay can be detrimental. The **osnoise** and **timerlat** tracers enable you to investigate and find causes of OS interference with applications and wakeup delay of RT threads.

(BZ#1979382)

The strace utility can now display mismatches between the actual SELinux contexts and the definitions extracted from the SELinux context database

An existing **--secontext** option of **strace** has been extended with the **mismatch** parameter. This parameter enables to print the expected context along with the actual one upon mismatch only. The output is separated by double exclamation marks (**!!**), first the actual context, then the expected one. In the examples below, the **full,mismatch** parameters print the expected full context along with the actual one because the user part of the contexts mismatches. However, when using a solitary **mismatch**, it only checks the type part of the context. The expected context is not printed because the type part of the contexts matches.

```
[...]
$ strace --secontext=full,mismatch -e statx stat /home/user/file
statx(AT_FDCWD, "/home/user/file"
[system_u:object_r:user_home_t:s0!!unconfined_u:object_r:user_home_t:s0], ...

$ strace --secontext=mismatch -e statx stat /home/user/file
statx(AT_FDCWD, "/home/user/file" [user_home_t:s0], ...
```

SELinux context mismatches often cause access control issues associated with SELinux. The mismatches printed in the system call traces can significantly expedite the checks of SELinux context correctness. The system call traces can also explain specific kernel behavior with respect to access control checks.

(BZ#2038992, BZ#2038810)

The --cyclictest-threshold option has been added to the rteval utility

With this enhancement, the **--cyclictest-threshold=USEC** option has been added to the **rteval** test suite. Using this option you can specify a threshold value. The **rteval** test run ends immediately if any latency measurements exceed this threshold value. When latency expectations are not met, the run aborts with a failure status.

(BZ#2012285)

4.10. FILE SYSTEMS AND STORAGE

RHEL 8.6 is compatible with RHEL 9 XFS images

With this update, RHEL 8.6 is now able to use RHEL 9 XFS images. RHEL 9 XFS guest images must have **bigtime** and inode btree counters (**inobtcount**) on-disk capabilities allowed in order to mount the guest image with RHEL 8.6. Note that file systems created with **bigtime** and **inobtcount** features are not compatible with versions earlier than RHEL 8.6.

(BZ#2022903, BZ#2024201)

Options in Samba utilities have been renamed and removed for a consistent user experience

The Samba utilities have been improved to provide a consistent command-line interface. These improvements include renamed and removed options. Therefore, to avoid problems after the update, review your scripts that use Samba utilities, and update them, if necessary.

Samba 4.15 introduces the following changes to the Samba utilities:

- Previously, Samba command-line utilities silently ignored unknown options. To prevent unexpected behavior, the utilities now consistently reject unknown options.
- Several command-line options now have a corresponding **smb.conf** variable to control their default value. See the man pages of the utilities to identify if a command-line option has an **smb.conf** variable name.
- By default, Samba utilities now log to standard error (**stderr**). Use the **--debug-stdout** option to change this behavior.
- The **--client-protection=off|sign|encrypt** option has been added to the common parser.
- The following options have been renamed in all utilities:
  - **--kerberos** to **--use-kerberos=required|desired|off**
  - **--krb5-ccache** to **--use-krb5-ccache=*CCACHE***
  - **--scope** to **--netbios-scope=*SCOPE***
  - **--use-ccache** to **--use-winbind-ccache**
- The following options have been removed from all utilities:
  - **-e** and **--encrypt**
  - **-C** removed from **--use-winbind-ccache**
  - **-i** removed from **--netbios-scope**
  - **-S** and **--signing**
- To avoid duplicate options, certain options have been removed or renamed from the following utilities:
  - **ndrdump**: **-l** is no longer available for **--load-dso**
  - **net**: **-l** is no longer available for **--long**
  - **sharesec**: **-V** is no longer available for **--viewsddl**

- **smbcquotas**: **--user** has been renamed to **--quota-user**
- **nmbd**: **--log-stdout** has been renamed to **--debug-stdout**
- **smbd**: **--log-stdout** has been renamed to **--debug-stdout**
- **winbindd**: **--log-stdout** has been renamed to **--debug-stdout**

(BZ#2062117)

Compiler barrier changed to static inline function compiler_barrier to avoid name conflict with function pointers

This enhancement provides additional features and a patch for a potential data corruption bug. The compiler barrier is now set to a static inline function **compiler_barrier**. No name conflict occurs with the hardware store barrier, when implementing hardware fencing for non-temporal memcpy variants, while using a function pointer. As a result, RHEL 8.6 now includes **pmdk** version 1.11.1.

(BZ#2009889)

4.11. HIGH AVAILABILITY AND CLUSTERS

The pcmk_delay_base parameter may now take different values for different nodes

When configuring a fence device, you now can specify different values for different nodes with the **pcmk_delay_base parameter**. This allows a single fence device to be used in a two-node cluster, with a different delay for each node. This helps prevent a situation where each node attempts to fence the other node at the same time. To specify different values for different nodes, you map the host names to the delay value for that node using a similar syntax to pcmk_host_map. For example, node1:0;node2:10s would use no delay when fencing node1 and a 10-second delay when fencing node2.

(BZ#1082146)

Specifying automatic removal of location constraint following resource move

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, previously available as a Technology Preview, is now fully supported. When you specify this option, the location constraint that the command creates is automatically removed once the resource has been moved.

(BZ#1990784)

Detailed Pacemaker status display for internal errors

If Pacemaker can not execute a resource or fence agent for some reason, for example the agent is not installed or there has been an internal timeout, the Pacemaker status displays now show a detailed exit reason for the internal error.

(BZ#1470834)

Support for special characters inside pcmk_host_map values

The **pcmk_host_map** property now supports special characters inside **pcmk_host_map** values using a backslash (\) in front of the value. For example, you can specify **pcmk_host_map="node3:plug\ 1"** to include a space in the host alias.

(BZ#1376538)

pcs suppport for OCF Resource Agent API 1.1 standard

The **pcs** command-line interface now supports OCF 1.1 resource and STONITH agents. An OCF 1.1 agent's metadata must comply with the OCF 1.1 schema. If an OCF 1.1 agent's metadata does not comply with the OCF 1.1 schema, **pcs** considers the agent invalid and will not create or update a resource of the agent unless the **--force** option is specified. The **pcsd** Web UI and **pcs** commands for listing agents omit OCF 1.1 agents with invalid metadata from the listing.

An OCF agent that declares that it implements any OCF version other than 1.1, or does not declare a version at all, is validated against the OCF 1.0 schema. Validation issues are reported as warnings, but for those agents it is not necessary to specify the **--force** option when creating or updating a resource of the agent.

(BZ#1936833)

New fencing agent for OpenShift

The **fence_kubevirt** fencing agent is now available for use with RHEL High Availability on Red Hat OpenShift Virtualization. For information on the **fence_kubevirt** agent, see the **fence_kubevirt**(8) man page.

(BZ#1977588)

4.12. DYNAMIC PROGRAMMING LANGUAGES, WEB AND DATABASE SERVERS

A new module stream: php:8.0

RHEL 8.6 adds **PHP 8.0**, which provides a number of bug fixes and enhancements over version 7.4

Notable enhancements include:

- New named arguments are order-independent and self-documented, and enable you to specify only required parameters.
- New attributes enable you to use structured metadata with PHP's native syntax.
- New union types enable you to use native union type declarations that are validated at runtime instead of PHPDoc annotations for a combination of types.
- Internal functions now more consistently raise an Error exception instead of warnings if parameter validation fails.
- The Just-In-Time compilation has improved the performance.
- The **Xdebug** debugging and productivity extension for PHP has been updated to version 3. This version introduces major changes in functionality and configuration compared to **Xdebug 2**.

To install the **php:8.0** module stream, use:

```
# yum module install php:8.0
```

If you want to upgrade from the **php:7.4** stream, see Switching to a later stream .

For details regarding PHP usage on RHEL 8, see Using the PHP scripting language .

(BZ#1978356, BZ#2027285)

A new module stream: perl:5.32

RHEL 8.6 introduces **Perl 5.32**, which provides a number of bug fixes and enhancements over **Perl 5.30** distributed in RHEL 8.3.

Notable enhancement include:

- **Perl** now supports unicode version 13.0.
- The **qr** qoute-like operator has been enhanced.
- The **POSIX::mblen()**, **mbtowc**, and **wctomb** functions now work on shift state locales and are thread-safe on C99 and above compilers when executed on a platform that has locale thread-safety; the length parameters are now optional.
- The new experimental **isa** infix operator tests whether a given object is an instance of a given class or a class derived from it.
- Alpha assertions are no longer experimental.
- Script runs are no longer experimental.
- Feature checks are now faster.
- **Perl** can now dump compiled patterns before optimization.

To upgrade from an earlier **perl** module stream, see Switching to a later stream .

(BZ#2021471)

A new package: nginx-mod-devel

A new **nginx-mod-devel** package has been added to the **nginx:1.20** module stream. The package provides all necessary files, including RPM macros and **nginx** source code, for building external dynamic modules for **nginx**.

(BZ#1991787)

MariaDB Galera now includes an upstream version of the garbd systemd service and a wrapper script

MariaDB 10.3 and MariaDB 10.5 in RHEL 8 include a Red Hat version of **garbd** systemd service and a wrapper script for the **galera** package in the **/usr/lib/systemd/system/garbd.service** and **/usr/sbin/garbd-wrapper** files, respectively.

In addition to the Red Hat version of these files, RHEL 8 now also provides an upstream version. The upstream files are located at **/usr/share/doc/galera/garb-systemd** and **/usr/share/doc/galera/garbd.service**.

RHEL 9 provides only the upstream version of these files, located at **/usr/lib/systemd/system/garbd.service** and **/usr/sbin/garb-systemd**.

(BZ#2042306, BZ#2042298, BZ#2050543, BZ#2050546)

4.13. COMPILERS AND DEVELOPMENT TOOLS

New command for capturing glibc optimization data

The new **ld.so --list-diagnostics** command captures data that influences **glibc** optimization decisions, such as IFUNC selection and **glibc-hwcaps** configuration, in a single machine-readable file.

(BZ#2023420)

glibc string functions are now optimized for Fujitsu A64FX

With this update, **glibc** string functions exhibit increased throughput and reduced latency on A64FX CPUs.

(BZ#1929928)

New UTF-8 locale en_US@ampm with 12-hour clock

With this update, you can now use a new UTF-8 locale **en_US@ampm** with a 12-hour clock. This new locale can be combined with other locales by using the **LC_TIME** environment variable.

(BZ#2000374)

New location for libffi's self-modifying code

With this update **libffi**'s self-modifying code takes advantage of a feature in the RHEL 8 kernel to create a suitable file independent of any file system. As a result, **libffi**'s self-modifying code no longer depends on making part of the filesystem insecure.

(BZ#1875340)

Updated GCC Toolset 11

GCC Toolset 11 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.

Notable changes introduced with RHEL 8.6 include:

- The GCC compiler has been updated to version 11.2.1.
- **annobin** has been updated to version 10.23.

The following tools and versions are provided by GCC Toolset 10:

Tool	Version
GCC	11.2.1
GDB	10.2
Valgrind	3.17.0
SystemTap	4.5
Dyninst	11.0.0
binutils	2.36.1

Tool	Version
elfutils	0.185
dwz	0.14
make	4.3
strace	5.13
ltrace	0.7.91
annobin	10.23

To install GCC Toolset 11, run the following command as root:

```
# yum install gcc-toolset-11
```

To run a tool from GCC Toolset 11:

```
$ scl enable gcc-toolset-11 tool
```

To run a shell session where tool versions from GCC Toolset 11 override system versions of these tools:

```
$ scl enable gcc-toolset-11 bash
```

For more information about usage, see Using GCC Toolset.

The GCC Toolset 11 components are available in the two container images:

- **rhel8/gcc-toolset-11-toolchain**, which includes the GCC compiler, the GDB debugger, and the **make** automation tool.
- **rhel8/gcc-toolset-11-perftools**, which includes the performance monitoring tools, such as SystemTap and Valgrind.

To pull a container image, run the following command as root:

```
# podman pull registry.redhat.io/<image_name>
```

Note that only the GCC Toolset 11 container images are now supported. Container images of earlier GCC Toolset versions are deprecated.

For details regarding the container images, see Using the GCC Toolset container images.

(BZ#1996862)

GDB disassembler now supports the new arch14 instructions

With this update, GDB is able to disassemble new arch14 instructions.

(BZ#2012818)

LLVM Toolset rebased to version 13.0.1

LLVM Toolset has been upgraded to version 13.0.1. Notable changes include:

- Clang now supports guaranteed tail calls with statement attributes **[[clang::musttail]]** in C++ and **__attribute__((musttail))** in C.
- Clang now supports the **-Wreserved-identifier** warning, which warns developers when using reserved identifiers in their code.
- Clang's **-Wshadow** flag now also checks for shadowed structured bindings.
- Clang's **-Wextra** now also implies **Wnull-pointer-subtraction**.

(BZ#2001133)

Rust Toolset rebased to 1.58.1

The **Rust Toolset** has been rebased to version 1.58.1. Notable changes include:

- The Rust compiler now supports the 2021 edition of the language, featuring disjoint capture in closure, **IntoIterator** for arrays, a new Cargo feature resolver, and more.
- Added Cargo support for new custom profiles.
- Cargo deduplicates compiler errors.
- Added new open range patterns.
- Added captured identifiers in format strings.

For further information, see:

- Rust 1.55
- Rust 1.56
- Rust 1.57
- Rust 1.58

(BZ#2002883)

Go Toolset rebased to version 1.17.7

Go Toolset has been upgraded to version 1.17.7. Notable changes include:

- Added an option to convert slices to array pointers.
- Added support for //go:build lines.
- Improvements to function call performance on amd64.
- Function arguments are formatted more clearly in stack traces.
- Functions containing closures can be inlined.
- Reduced resource consumption in x509 certificate parsing.

(BZ#2014088)

pcp rebased to 5.3.5

The **pcp** package has been rebased to version 5.3.5. Notable changes include:

- Added new **pmieconf(1)** rules for CPU and disk saturation.
- Improved stability and scalability of **pmproxy(1)** service.
- Improved service latency and robustness of **pmlogger(1)** service.
- Added new performance metrics related to electrical power.
- Added new features in the **pcp-htop(1)** utility.
- Added new features in the **pcp-atop(1)** utility.
- Updated Nvidia GPU metrics.
- Added new Linux kernel KVM and networking metrics.
- Added a new MongoDB metrics agent.
- Added a new sockets metrics agent and **pcp-ss(1)** utility.
- Disabled **pmcd(1)** and **pmproxy(1)** Avahi service advertising by default.

(BZ#1991763)

The grafana package rebased to version 7.5.11

The **grafana** package has been rebased to version 7.5.11. Notable changes include:

- Added a new **prepare time series** transformation for backward compatibility of panels that do not support the new data frame format.

(BZ#1993214)

grafana-pcp rebased to 3.2.0

The **grafana-pcp** package has been rebased to version 3.2.0. Notable changes include:

- Added a new MS SQL server dashboard for PCP Redis.
- Added visibility of empty histogram buckets in the PCP Vector eBPF/BCC Overview dashboard.
- Fixed a bug where the **metric()** function of PCP Redis did not return all metric names.

(BZ#1993149)

js-d3-flame-graph rebased to 4.0.7

The **js-d3-flame-graph** package has been rebased to version 4.0.7. Notable changes include:

- Added new blue and green color scheme.
- Added functionality to display flame graph context.

(BZ#1993194)

Power consumption metrics now available in PCP

The new **pmda-denki** Performance Metrics Domain Agent (PMDA) reports metrics related to power consumption. Specifically, it reports:

- Consumption metrics based on Running Average Power Limit (RAPL) readings, available on recent Intel CPUs
- Consumption metrics based on battery discharge, available on systems which have a battery

(BZ#1629455)

A new module: log4j:2

A new **log4j:2** module is now available in the AppStream repository. This module contains **Apache Log4j 2**, which is a Java logging utility and a library enabling you to output log statements to a variety of output targets.

Log4j 2 provides significant improvements over **Log4j 1**. Notably, **Log4j 2** introduces enhancements to the **Logback** framework and fixes some inherent problems in the **Logback** architecture.

To install the **log4j:2** module stream, use:

```
# yum module install log4j:2
```

(BZ#1937468)

4.14. IDENTITY MANAGEMENT

ansible-freeipa is now available in the AppStream repository with all dependencies

Previously in RHEL 8, before installing the **ansible-freeipa** package, you first had to enable the Ansible repository and install the **ansible** package. In RHEL 8.6 and RHEL 9, you can install **ansible-freeipa** without any preliminary steps. Installing **ansible-freeipa** automatically installs the **ansible-core** package, a more basic version of **ansible**, as a dependency. Both **ansible-freeipa** and **ansible-core** are available in the **rhel-9-for-x86_64-appstream-rpms** repository.

ansible-freeipa in RHEL 8.6 and RHEL 9 contains all the modules that it contained in RHEL 8.

(JIRA:RHELPLAN-100359)

IdM now supports the automountlocation, automountmap, and automountkey Ansible modules

With this update, the **ansible-freeipa** package contains the **ipaautomountlocation**, **ipaautomountmap**, and **ipaautomountkey** modules. You can use these modules to configure directories to be mounted automatically for IdM users logged in to IdM clients in an IdM location. Note that currently, only direct maps are supported.

(JIRA:RHELPLAN-79161)

The support for managing subID ranges is available in the shadow-utils

Previously, **shadow-utils** configured the subID ranges automatically from the **/etc/subuid** and **/etc/subgid** files. With this update, the configuration of subID ranges is available in the **/etc/nsswitch.conf** file by setting a value in the **subid** field. For more information, see **man subuid** and

man subgid. Also, with this update, an SSSD implementation of the **shadow-utils** plugin is available, which provides the subID ranges from the IPA server. To use this functionality, add the **subid: sss** value to the **/etc/nsswitch.conf** file. This solution might be useful in the containerized environment to facilitate rootless containers.

Note that in case the **/etc/nsswitch.conf** file is configured by the **authselect** tool, you must follow the procedures described in the **authselect** documentation. When it is not the case, you can modify the **/etc/nsswitch.conf** file manually.

(JIRA:RHELPLAN-103579)

An alternative to the traditional RHEL ansible-freeipa repository: Ansible Automation Hub

With this update, you can download **ansible-freeipa** modules from the Ansible Automation Hub (AAH) instead of downloading them from the standard RHEL repository. By using AAH, you can benefit from the faster updates of the **ansible-freeipa** modules available in this repository.

In AAH, **ansible-freeipa** roles and modules are distributed in the collection format. Note that you need an Ansible Automation Platform (AAP) subscription to access the content on the AAH portal. You also need **ansible** version 2.9 or later.

The **redhat.rhel_idm** collection has the same content as the traditional **ansible-freeipa** package. However, the collection format uses a fully qualified collection name (FQCN) that consists of a namespace and the collection name. For example, the **redhat.rhel_idm.ipadnsconfig** module corresponds to the **ipadnsconfig** module in **ansible-freeipa** provided by a RHEL repository. The combination of a namespace and a collection name ensures that the objects are unique and can be shared without any conflicts.

(JIRA:RHELPLAN-103147)

ansible-freeipa modules can now be executed remotely on IdM clients

Previously, **ansible-freeipa** modules could only be executed on IdM servers. This required your Ansible administrator to have **SSH** access to your IdM server, causing a potential security threat. With this update, you can execute **ansible-freeipa** modules remotely on systems that are IdM clients. As a result, you can manage IdM configuration and entities in a more secure way.

To execute **ansible-freeipa** modules on an IdM client, choose one of the following options:

- Set the **hosts** variable of the playbook to an IdM client host.
- Add the **ipa_context: client** line to the playbook task that uses the **ansible-freeipa** module.

You can set the **ipa_context** variable to **client** on an IdM server, too. However, the server context usually provides better performance. If **ipa_context** is not set, **ansible-freeipa** checks if it is running on a server or a client, and sets the context accordingly. Note that executing an **ansible-freeipa** module with **context** set to **server** on an IdM client host raises an error of **missing libraries**.

(JIRA:RHELPLAN-103146)

The ipadnsconfig module now requires action: member to exclude a global forwarder

With this update, excluding global forwarders in Identity Management (IdM) by using the **ansible-freeipa ipadnsconfig** module requires using the **action: member** option in addition to the **state: absent** option. If you only use **state: absent** in your playbook without also using **action: member**, the playbook fails. Consequently, to remove all global forwarders, you must specify all of them individually in the playbook. In contrast, the **state: present** option does not require **action: member**.

(BZ#2046325)

Identity Management now supports SHA384withRSA signing by default

With this update, the Certificate Authority (CA) in IdM supports the SHA-384 With RSA Encryption signing algorithm. SHA384withRSA is compliant with the Federal Information Processing Standard (FIPS).

(BZ#1731484)

SSSD default SSH hashing value is now consistent with the OpenSSH setting

The default value of **ssh_hash_known_hosts** has been changed to false. It is now consistent with the OpenSSH setting, which does not hash host names by default.

However, if you need to continue to hash host names, add **ssh_hash_known_hosts = True** to the **[ssh]** section of the **/etc/sssd/sssd.conf** configuration file.

(BZ#2015070)

samba rebased to version 4.15.5

The *samba* packages have been upgraded to upstream version 4.15.5, which provides bug fixes and enhancements over the previous version:

- Options in Samba utilities have been renamed and removed for a consistent user experience
- Server multi-channel support is now enabled by default.
- The **SMB2_22**, **SMB2_24**, and **SMB3_10** dialects, which were only used by Windows technical previews, have been removed.

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#2013596)

Directory Server rebased to version 1.4.3.28

The **389-ds-base** packages have been upgraded to upstream version 1.4.3, which provides a number of bug fixes and enhancements over the previous version:

- A potential deadlock in replicas has been fixed.
- The server no longer terminates unexpectedly when the **dnaInterval** is set to **0**.
- The performance of connection handling has been improved.
- Improved performance of **targetfilter** in access control instructions (ACI).

(BZ#2016014)

Directory Server now stores memory-mapped files of databases on a tmpfs file system

In Directory Server, the **nsslapd-db-home-directory** parameter defines the location of memory-mapped files of databases. This enhancement changes the default value of the parameter from **/var/lib/dirsrv/slapd-*instance_name*/db/** to **/dev/shm/**. As a result, with the internal databases stored on a **tmpfs** file system, the performance of Directory Server increases.

(BZ#1780842)

4.15. DESKTOP

Security classification banners at login and in the desktop session

You can now configure classification banners to state the overall security classification level of the system. This is useful for deployments where the user must be aware of the security classification level of the system that they are logged into.

The classification banners can appear in the following contexts, depending on your configuration:

- Within the running session
- On the lock screen
- On the login screen

The classification banners can take the form of either a notification that you can dismiss, or a permanent banner.

For more information, see Displaying the system security classification.

(BZ#1751336)

4.16. GRAPHICS INFRASTRUCTURES

Intel Alder Lake-P GPUs are now supported

This release adds support for the Intel Alder Lake-P CPU microarchitecture with integrated graphics. This includes Intel UHD Graphics and Intel Xe integrated GPUs found with the following CPU models:

- Intel Core i7-1280P
- Intel Core i7-1270P
- Intel Core i7-1260P
- Intel Core i5-1250P
- Intel Core i5-1240P
- Intel Core i3-1220P

Support for Alder Lake-P graphics is disabled by default. To enable it, add the following option to the kernel command line:

```
i915.force_probe=PCI_ID
```

Replace *PCI_ID* with either the PCI device ID of your Intel GPU, or with the * character to enable support for all alpha-quality hardware that uses the **i915** driver.

(BZ#1964761)

4.17. THE WEB CONSOLE

Smart card authentication for sudo and SSH from the web console

Previously, it was not possible to use smart card authentication to obtain sudo privileges or use SSH in the web console. With this update, Identity Management users can use a smart card to gain sudo privileges or to connect to a different host with SSH.

NOTE

It is only possible to use one smart card to authenticate and gain sudo privileges. Using a separate smart card for sudo is not supported.

(JIRA:RHELPLAN-95126)

RHEL web console provides Insights registration by default

With this update, when you use the Red Hat Enterprise Linux web console to register a RHEL system, the **Connect this system to Red Hat Insights.**check box is checked by default. If you do not want to connect to the Insights service, uncheck the box.

(BZ#2049441)

Cockpit now supports using an existing TLS certificate

With this enhancement, the certificate does not have strict file permission requirements any more (such as **root:cockpit-ws 0640**), and thus it can be shared with other services.

(JIRA:RHELPLAN-103855)

4.18. RED HAT ENTERPRISE LINUX SYSTEM ROLES

The Firewall RHEL System Role has been added in RHEL 8

The **rhel-system-roles.firewall** RHEL System Role was added to the **rhel-system-roles** package. As a result, administrators can automate their firewall settings for managed nodes.

(BZ#1854988)

Full Support for HA Cluster RHEL System Role

The High Availability Cluster (HA Cluster) role, previously available as a Technology Preview, is now fully supported. The following notable configurations are available:

- Configuring fence devices, resources, resource groups, and resource clones including meta attributes and resource operations
- Configuring resource location constraints, resource colocation constraints, resource order constraints, and resource ticket constraints
- Configuring cluster properties
- Configuring cluster nodes, custom cluster names and node names

- Configuring multi-link clusters
- Configuring whether clusters start automatically on boot

Running the role removes any configuration not supported by the role or not specified when running the role.

The HA Cluster System Role does not currently support SBD.

(BZ#1893743)

The Networking System Role now supports OWE

Opportunistic Wireless Encryption (OWE) is a mode of opportunistic security for Wi-Fi networks that provides encryption of the wireless medium but no authentication, such as public hot spots. OWE uses encryption between Wi-Fi clients and access points, protecting them from sniffing attacks. With this enhancement, the Networking RHEL System role supports OWE. As a result, administrators can now use the Networking System Role to configure connections to Wi-Fi networks which use OWE.

(BZ#1993379)

The Networking System Role now supports SAE

In Wi-Fi protected access version 3 (WPA3) networks, the simultaneous authentication of equals (SAE) method ensures that the encryption key is not transmitted. With this enhancement, the Networking RHEL System role supports SAE. As a result, administrators can now use the Networking System Role to configure connections to Wi-Fi networks, which use WPA-SAE.

(BZ#1993311)

The Cockpit RHEL System Role is now supported

With this enhancement, you can install and configure the web console in your system. Consequently, you can manage web console in an automated manner.

(BZ#2021661)

Add support for raid_level for LVM volumes

The Storage RHEL System Role can now specify the **raid_level** parameter for LVM volumes. As a result, LVM volumes can be grouped into RAIDs using the **lvmraid** feature.

(BZ#2016514)

The NBDE client System Role supports systems with static IP addresses

Previously, restarting a system with a static IP address and configured with the 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.

(BZ#1985022)

Support for cached volumes is available in the Storage System Role

Storage RHEL System Role can now create and manage cached LVM logical volumes. LVM cache can be used to improve performance of slower logical volumes by temporarily storing subsets of an LV's data on a smaller, faster device, for example an SSD.

(BZ#2016511)

Support to add Elasticsearch username and password for authentication from rsyslog

This update adds the **Elasticsearch** username and password parameters to the **logging** System Role, to enable the **rsyslog** to authenticate to Elasticsearch using username and password.

(BZ#2010327)

Ansible Core support for the RHEL System Roles

As of RHEL 8.6 GA release, Ansible Core is provided, with a limited scope of support, to enable RHEL supported automation use cases. Ansible Core replaces Ansible Engine which was previously provided in a separate repository. Ansible Core is available in the AppStream repository for RHEL. For more details on the supported use cases, see Scope of support for the Ansible Core package included in the RHEL 9 and RHEL 8.6 and later AppStream repositories. Users must manually migrate their systems from Ansible Engine to Ansible Core.

For details on that, see Using Ansible in RHEL 8.6 and later .

(BZ#2012316)

The network RHEL System Role now supports both named and numeric routing tables in static routes.

This update adds support for both the **named** and **numeric** routing tables in static routes, which is a prerequisite for supporting the policy routing (for example, source routing). The users can define policy routing rules later to instruct the system which table to use to determine the correct route. As a result, after the user specifies the **table** attribute in the **route**, the system can add routes into the routing table.

(BZ#2031521)

The Certificate role consistently uses "Ansible_managed" comment in its hook scripts

With this enhancement, the Certificate role generates pre-scripts and post-scripts to support providers, to which the role inserts the "Ansible managed" comment using the Ansible standard "ansible_managed" variable:

- **/etc/certmonger/pre-scripts/script_name.sh**
- **/etc/certmonger/post-scripts/script_name.sh**

The comment indicates that the script files should not be directly edited because the Certificate role can overwrite the file. As a result, the configuration files contain a declaration stating that the configuration files are managed by Ansible.

(BZ#2054364)

The Terminal session recording System Role uses the "Ansible managed" comment in its managed configuration files

The Terminal session recording role generates 2 configuration files:

- **/etc/sssd/conf.d/sssd-session-recording.conf**
- **/etc/tlog/tlog-rec-session.conf**

With this update, the Terminal session recording role inserts the **Ansible managed** comment into the configuration files, using the standard Ansible variable **ansible_managed**. The comment indicates that the configuration files should not be directly edited because the Terminal session recording role can

overwrite the file. As a result, the configuration files contain a declaration stating that the configuration files are managed by Ansible.

(BZ#2054363)

Microsoft SQL System Role now supports customized repository for disconnected or Satellite subscriptions

Previously, users in disconnected environments that needed to pull packages from a custom server or Satellite users that needed to point to Satellite or Capsule had no support from Microsoft SQL Role . This update fixes it, by enabling users to provide a customized URL to use for **RPM** key, **client** and **server** mssql repositories. If no URL is provided, the **mssql** role uses the official Microsoft servers to download RPMs.

(BZ#2038256)

The Microsoft SQL System Role consistently uses "Ansible_managed" comment in its managed configuration files

The **mssql** role generates the following configuration file:

- **/var/opt/mssql/mssql.conf**

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

(BZ#2057651)

Support to all bonding options added to the Networking System Role

This update provides support to all bonding options to the Networking RHEL System Role. Consequently, it enables you to flexibly control the network transmission over the bonded interface. As a result, you can control the network transmission over the bonded interface by specifying several options to that interface.

(BZ#2008931)

NetworkManager supports specifying a network card using its PCI address

Previously, during setting a connection profile, NetworkManager was only allowed to specify a network card using either its name or MAC address. In this case, the device name is not stable and the MAC address requires inventory to maintain record of used MAC addresses. Now, you can specify a network card based on its PCI address in a connection profile.

(BZ#1695634)

A new option auto_gateway controls the default route behavior

Previously, the **DEFROUTE** parameter was not configurable with configuration files but only manually configurable by naming every route. This update adds a new **auto_gateway** option in the **ip** configuration section for connections, with which you can control the default route behavior. You can configure **auto_gateway** in the following ways:

- If set to **true**, default gateway settings apply to a default route.
- If set to **false**, the default route is removed.

- If unspecified, the **network** role uses the default behavior of the selected **network_provider**.

(BZ#1897565)

The VPN role consistently uses Ansible_managed comment in its managed configuration files

The VPN role generates the following configuration file:

- **/etc/ipsec.d/mesh.conf**
- **/etc/ipsec.d/policies/clear**
- **/etc/ipsec.d/policies/private**
- **/etc/ipsec.d/policies/private-or-clear**

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

(BZ#2054365)

New source parameter in the Firewall System Role

You can now use the **source** parameter of the Firewall System Role to add or remove sources in the firewall configuration.

(BZ#1932678)

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#2057656)

The Firewall System Role now supports setting the firewall default zone

You can now set a default firewall zone in the Firewall System role. Zones represent a concept to manage incoming traffic more transparently. The zones are connected to networking interfaces or assigned a range of source addresses. Firewall rules for each zone are managed independently enabling the administrator to define complex firewall settings and apply them to the traffic. This feature allows setting the default zone used as the default zone to assign interfaces to, same as **firewall-cmd --set-default-zone zone-name**.

(BZ#2022458)

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#2057645)

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#2057661)

4.19. VIRTUALIZATION

Mediated devices are now supported by virtualization CLIs on IBM Z

Using **virt-install** or **virt-xml**, you can now attach mediated devices to your virtual machines (VMs), such as vfio-ap and vfio-ccw. This for example enables more flexible management of DASD storage devices and cryptographic coprocessors on IBM Z hosts. In addition, using **virt-install**, you can create a VM that uses an existing DASD mediated device as its primary disk. For instructions to do so, see the Configuring and Managing Virtualization in RHEL 8 guide.

(BZ#1995125)

Virtualization support for Intel Atom P59 series processors

With this update, virtualization on RHEL 8 adds support for the Intel Atom P59 series processors, formerly known as Snow Ridge. As a result, virtual machines hosted on RHEL 8 can now use the **Snowridge** CPU model and utilise new features that the processors provide.

(BZ#1662007)

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)

Windows 11 and Windows Server 2022 guests are supported

RHEL 8 now supports using Windows 11 and Windows Server 2022 as the guest operating systems on KVM virtual machines.

(BZ#2036863, BZ#2004162)

4.20. RHEL IN CLOUD ENVIRONMENTS

RHEL 8 virtual machines are now supported on certain ARM64 hosts on Azure

Virtual machines that use RHEL 8.6 or later as the guest operating system are now supported on Microsoft Azure hypervisors running on Ampere Altra ARM-based processors.

(BZ#1949614)

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)

cloud-init supports user data on Microsoft Azure

The **--user-data** option has been introduced for the **cloud-init** utility. Using this option, you can pass scripts and metadata from the Azure Instance Metadata Service (IMDS) when setting up a RHEL 8 virtual machine on Azure.

(BZ#2023940)

cloud-init supports the VMware GuestInfo datasource

With this update, the **cloud-init** utility is able to read the datasource for VMware guestinfo data. As a result, using **cloud-init** to set up RHEL 8 virtual machines on VMware vSphere is now more efficient and reliable.

(BZ#2026587)

4.21. SUPPORTABILITY

A new package: rig

RHEL 8 introduces the **rig** package, which provides the **rig** system monitoring and event handling utility.

The **rig** utility is designed to assist system administrators and support engineers in diagnostic data collection for issues that are seemingly random in their occurrence, or occur at inopportune times for human intervention.

(BZ#1888705)

sos report now offers an estimate mode run

This **sos report** update adds the **--estimate-only** option with which you can approximate the disk space required for collecting an **sos** report from a RHEL server. Running the **sos report --estimate-only** command:

- executes a dry run of **sos report**
- mimics all plugins consecutively and estimates their disk size.

Note that the final disk space estimation is very approximate. Therefore, it is recommended to double the estimated value.

(BZ#1873185)

Red Hat Support Tool now uses Hydra APIs

The **Red Hat Support Tool** has moved from the deprecated Strata APIs to the new Hydra APIs. This has no impact on functionality. However, if you have configured the firewall to allow only the Strata API /**rs**/ path explicitly, update it to /**support**/ to ensure the firewall works correctly.

In addition, due to this change, you can now download files greater than 5 GB when using the **Red Hat Support Tool**.

(BZ#2018194)

Red Hat Support Tool now supports Red Hat Secure FTP

When using **Red Hat Support Tool**, you can now upload files to the case by the **Red Hat Secure FTP**. **Red Hat Secure FTP** is a more secure replacement of the deprecated **Dropbox** utility that **Red Hat Support Tool** used to support in its earlier versions.

(BZ#2018195)

Red Hat Support Tool now supports S3 APIs

The **Red Hat Support Tool** now uses S3 APIs to upload files to the Red Hat Technical Support case. As a result, users can upload a file greater than 1 GB to the case directly.

(BZ#1767195)

4.22. CONTAINERS

container-tools:4.0 stable stream is now available

The **container-tools:4.0** stable module stream, which contains the Podman, Buildah, Skopeo, and runc tools is now available. This update provides bug fixes and enhancements over the previous version.

For instructions on how to upgrade from an earlier stream, see Switching to a later stream .

(JIRA:RHELPLAN-100175)

The NFS storage is now available

You can now use the NFS file system as a backend storage for containers and images if your file system has xattr support.

(JIRA:RHELPLAN-75169)

The container-tools:rhel8 module has been updated

The **container-tools:rhel8** module, which contains the Podman, Buildah, Skopeo, crun, and runc tools is now available. This update provides a list of bug fixes and enhancements over the previous version.

Notable changes include:

- Due to the changes in the network stack, containers created by Podman v3 and earlier will not be usable in v4.0
- The native overlay file system is usable as a rootless user
- Support for NFS storage within a container
- Downgrading to earlier versions of Podman is not supported unless all containers are destroyed and recreated

Podman tool has been upgraded to version 4.0, for further information about notable changes, see the upstream release notes.

(JIRA:RHELPLAN-100174)

Universal Base Images are now available on Docker Hub

Previously, Universal Base Images were only available from the Red Hat container catalog. With this enhancement, Universal Base Images are also available from Docker Hub as a Verified Publisher image.

(JIRA:RHELPLAN-101137)

A podman container image is now available

The **registry.redhat.io/rhel8/podman** container image, previously available as a Technology Preview, is now fully supported. The **registry.redhat.io/rhel8/podman** container image is a containerized implementation of the **podman** package. The **podman** tool manages containers and images, volumes mounted into those containers, and pods made of groups of containers.

(JIRA:RHELPLAN-57941)

Podman now supports auto-building and auto-running pods using a YAML file

The **podman play kube** command automatically builds and runs multiple pods with multiple containers in the pods using a YAML file.

(JIRA:RHELPLAN-108830)

Podman now has ability to source subUID and subGID ranges from IdM

The subUID and subGID ranges can now be managed by IdM. Instead of deploying the same **/etc/subuid** and **/etc/subgid** files onto every host, you can now define range in a single central storage. You have to modify the **/etc/nsswitch.conf** file and add **sss** to the services map line: **services: files sss**.

For more details, see Managing subID ranges manually in IdM documentation.

(JIRA:RHELPLAN-101133)

The openssl container image is now available

The **openssl** image provides an **openssl** command-line tool for using the various functions of the OpenSSL crypto library. Using the OpenSSL library, you can generate private keys, create certificate signing requests (CSRs), and display certificate information.

The **openssl** container image is available in these repositories:

- registry.redhat.io/rhel8/openssl
- registry.access.redhat.com/ubi8/openssl

(JIRA:RHELPLAN-101138)

Netavark network stack is now available

The new network stack available starting with Podman 4.1.1-7 consists of two tools, the Netavark network setup tool and the Aardvark DNS server. The Netavark stack, previously available as a Technology Preview, is with the release of the RHBA-2022:7127 advisory 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-137623)

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

With the release of the RHBA-2022:7127 advisory. 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#2130912)

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 8.6. 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

fw_devlink.strict = [KNL]

Format: <bool>
With this parameter you can treat all inferred dependencies as mandatory dependencies. This setting only applies if **fw_devlink=on|rpm**.

no_hash_pointers

With this parameter you can force pointers that are printed to the console or buffers to be unhashed. By default, when a pointer is printed using the **%p** format string that pointer's value is obscured by hashing. This is a security feature that hides actual kernel addresses from unprivileged users. However, it also makes debugging the kernel more difficult since you cannot compare unequal pointers. If this command-line parameter is specified, then all normal pointers will have their true value printed. Pointers that are printed using the **%pK** format string can still be hashed. Specify **no_hash_pointers** only when debugging the kernel and do not use it in production.

no_entry_flush = [PPC]

With this parameter it is possible to avoid flushing the L1-D cache when entering the kernel.

no_uaccess_flush = [PPC]

With this parameter it is possible to avoid flushing the L1-D cache after accessing user data.

rcutorture.nocbs_nthreads = [KNL]

With this parameter you can set the number of Read-copy-update (RCU) callback-offload togglers. The default value is 0 (zero) and it disables toggling.

rcutorture.nocbs_toggle = [KNL]

With this parameter you can set the delay in milliseconds between successive callback-offload toggling attempts.

refscale.verbose_batched = [KNL]

With this parameter you can batch the additional **printk()** statements.
You can print everything, by specifying zero (the default) or a negative value. Otherwise, print every Nth verbose statement, where N is the value specified.

strict_sas_size = [X86]

Format: <bool>
With this parameter you can enable or disable strict **sigaltstack** size checks against the required signal frame size which depends on the supported floating-point unit (FPU) features. You can use this parameter to filter out binaries, which have not yet been made aware of the **AT_MINSIGSTKSZ** auxiliary vector.

torture.verbose_sleep_frequency = [KNL]

This parameter specifies how many verbose **printk()** statements should be emitted between each sleep.

The default value of 0 (zero) disables the verbose-printk() sleeping.

torture.verbose_sleep_duration = [KNL]

This parameter specifies the duration of each verbose-printk() sleep in jiffies.

tsc_early_khz = [X86]

Format: <unsigned int>
This parameter enables to skip the early Time Stamp Counter (TSC) calibration and use the given value instead. The parameter proves useful when the early TSC frequency discovery procedure is not reliable. Such as on overclocked systems with CPUID.16h support and partial CPUID.15h support.

Updated kernel parameters

amd_iommu = [HW,X86-64]

You can pass parameters to the AMD IOMMU driver in the system.
Possible values are:

- **fullflush** - Enable flushing of IO/TLB entries when they are unmapped. Otherwise they are flushed before they will be reused, which is a lot of faster.
- **off** - Do not initialize any AMD IOMMU found in the system.
- **force_isolation** - Force device isolation for all devices. The IOMMU driver is not allowed anymore to lift isolation requirements as needed. This option does not override **iommu=pt**.
- **force_enable** - Force enable the IOMMU on platforms known to be buggy with IOMMU enabled. Use this option with care.

acpi.debug_level = [HW,ACPI,ACPI_DEBUG]

Format: <int>
CONFIG_ACPI_DEBUG must be enabled to produce any Advanced Configuration and Power Interface (ACPI) debug output. Bits in **debug_layer** correspond to a **_COMPONENT** in an ACPI source file. For example **#define _COMPONENT ACPI_EVENTS** Bits in debug_level correspond to a level in **ACPI_DEBUG_PRINT** statements. For example **ACPI_DEBUG_PRINT((ACPI_DB_INFO, …**

The debug_level mask defaults to "info". See **Documentation/acpi/debug.txt** for more information about debug layers and levels.

Enable processor driver info messages:

acpi.debug_layer=0x20000000

Enable AML "Debug" output, for example, stores to the Debug object while interpreting AML:

acpi.debug_layer=0xffffffff, **acpi.debug_level=0x2** Enable all messages related to ACPI hardware: **acpi.debug_layer=0x2**, **acpi.debug_level=0xffffffff**

Some values produce so much output that the system is unusable. The **log_buf_len** parameter is useful if you need to capture more output.

acpi_mask_gpe = [HW,ACPI]

Format: <byte> or <bitmap-list>

Due to the existence of **_Lxx**/**_Exx**, some general purpose events (GPEs) triggered by unsupported hardware or firmware features can result in GPE floodings that cannot be automatically disabled by the GPE dispatcher. You can use this facility to prevent such uncontrolled GPE floodings.

cgroup_disable = [KNL]

Format: <name of the controller(s) or feature(s) to disable>
With this parameter you can disable a particular controller or optional feature.

The effects of **cgroup_disable = <controller/feature>** are:

- **controller/feature** is not auto-mounted if you mount all **cgroups** in a single hierarchy
- **controller/feature** is not visible as an individually mountable subsystem
- if **controller/feature** is an optional feature then the feature is disabled and corresponding **cgroups** files are not created
  Currently only memory controller deals with this and cut the overhead, others just disable the usage. So only **cgroup_disable=memory** is actually worthy.

  Specifying "pressure" disables per-cgroup pressure stall information accounting feature.

clearcpuid = BITNUM[,BITNUM...] [X86]

With this parameter you can disable CPUID feature X for the kernel. See **arch/x86/include/asm/cpufeatures.h** for the valid bit numbers. Linux specific bits are not necessarily stable over kernel options, but the vendor specific ones should be. User programs calling CPUID directly or using the feature without checking anything will still see it. This just prevents it from being used by the kernel or shown in **/proc/cpuinfo**. Also note the kernel could malfunction if you disable some critical bits.

iommu.strict = [ARM64, X86]

Format: <"0" | "1">
With this parameter you can configure translation look-aside buffer (TLB) invalidation behavior.

Possible values are:

- 0 - lazy mode, requests that use of Direct Memory Access (DMA) unmap operations is deferred
- 1 - strict mode (default), DMA unmap operations invalidate IOMMU hardware TLBs synchronously.
  On AMD64 and Intel 64, the default behavior depends on the equivalent driver-specific parameters. However, a strict mode explicitly specified by either method takes precedence.

rcutree.use_softirq = [KNL]

If this parameter is set to zero, it moves all **RCU_SOFTIRQ** processing to per-CPU rcuc kthreads. The default is a non-zero value. It means that **RCU_SOFTIRQ** is used by default.
Specify **rcutree.use_softirq = 0** to use rcuc kthreads. But note that **CONFIG_PREEMPT_RT=y** kernels disable this kernel boot parameter (forcibly setting it to zero).

rcupdate.rcu_normal_after_boot = [KNL]

This parameter enables to use only normal grace-period primitives once boot has completed. That is after the **rcu_end_inkernel_boot()** call has been invoked. There is no effect on **CONFIG_TINY_RCU** kernels.

The kernels with the **CONFIG_PREEMPT_RT=y** setting, enable this kernel boot parameter and forcibly they set it to the value one. That is, converting any post-boot attempt at an expedited Read-copy-update (RCU) grace period to instead use normal non-expedited grace-period processing.

spectre_v2 = [X86]

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

Possible values are:

- on - unconditionally enable, implies **spectre_v2_user=on**
- off - unconditionally disable, implies **spectre_v2_user=off**
- auto - the kernel detects whether your CPU model is vulnerable
  Selecting 'on' will, and 'auto' may, choose a mitigation method at run time according to the CPU. The available microcode, the setting of the **CONFIG_RETPOLINE** configuration option, and the compiler with which the kernel was built.

  Selecting 'on' will also enable the mitigation against user space to user space task attacks.

  Selecting 'off' will disable both the kernel and the user space protections.

  You can also select specific mitigations manually:

- retpoline - replace indirect branches
- retpoline,generic - Retpolines
- retpoline,lfence - LFENCE; indirect branch
- retpoline,amd - alias for retpoline,lfence
- eibrs - enhanced indirect branch restricted speculation (IBRS)
- eibrs,retpoline - enhanced IBRS + Retpolines
- eibrs,lfence - enhanced IBRS + LFENCE
- ibrs - use IBRS to protect kernel
- ibrs_always - use IBRS to protect both kernel and userland
- retpoline,ibrs_user - replace indirect branches with retpolines and use IBRS to protect userland

Not specifying this option is equivalent to **spectre_v2=auto**.

CHAPTER 6. DEVICE DRIVERS

6.1. NEW DRIVERS

Network drivers

- MT7921E 802.11ax wireless driver (mt7921e.ko.xz)
- Realtek 802.11ax wireless core module (rtw89_core.ko.xz)
- Realtek 802.11ax wireless PCI driver (rtw89_pci.ko.xz)
- ntb_netdev (ntb_netdev.ko.xz)
- Intel® Ethernet Protocol Driver for RDMA (irdma.ko.xz)
- Intel® PCI-E Non-Transparent Bridge Driver (ntb_hw_intel.ko.xz)

Graphics drivers and miscellaneous drivers

- Generic Counter interface (counter.ko.xz)
- Intel Quadrature Encoder Peripheral driver (intel-qep.ko.xz)
- AMD ® PCIe MP2 Communication Driver (amd_sfh.ko.xz)
- Driver to initialize some steering wheel joysticks from Thrustmaster (hid-thrustmaster.ko.xz)
- HID over I2C ACPI driver (i2c-hid-acpi.ko.xz)
- Intel PMC Core Driver (intel_pmc_core.ko.xz)
- ThinkLMI Driver (think-lmi.ko.xz)
- Processor Thermal Reporting Device Driver (int3401_thermal.ko.xz)
- Processor Thermal Reporting Device Driver (processor_thermal_device_pci.ko.xz)
- Processor Thermal Reporting Device Driver (processor_thermal_device_pci_legacy.ko.xz)
- TI TPS6598x USB Power Delivery Controller Driver (tps6598x.ko.xz)

6.2. UPDATED DRIVERS

Network drivers

- Intel® PRO/1000 Network Driver (e1000e.ko.xz) has been updated.
- Intel® Ethernet Switch Host Interface Driver (fm10k.ko.xz) has been updated.
- Intel® Ethernet Connection XL710 Network Driver (i40e.ko.xz) has been updated.
- Intel® Ethernet Adaptive Virtual Function Network Driver (iavf.ko.xz) has been updated.
- Intel® Gigabit Ethernet Network Driver (igb.ko.xz) has been updated.

- Intel® Gigabit Virtual Function Network Driver (igbvf.ko.xz) has been updated.
- Intel® 2.5G Ethernet Linux Driver (igc.ko.xz) has been updated.
- Intel® 10 Gigabit PCI Express Network Driver (ixgbe.ko.xz) has been updated.
- Intel® 10 Gigabit Virtual Function Network Driver (ixgbevf.ko.xz) has been updated.
- Mellanox 5th generation network adapters (ConnectX series) core driver (mlx5_core.ko.xz) has been updated.
- VMware vmxnet3 virtual NIC driver (vmxnet3.ko.xz) has been updated to version 1.6.0.0-k.

Storage drivers

- Emulex LightPulse Fibre Channel SCSI driver (lpfc.ko.xz) has been updated to version 0:14.0.0.4.
- Broadcom MegaRAID SAS Driver (megaraid_sas.ko.xz) has been updated to version 07.719.03.00-rh1.
- LSI MPT Fusion SAS 3.0 Device Driver (mpt3sas.ko.xz) has been updated to version 39.100.00.00.
- QLogic Fibre Channel HBA Driver (qla2xxx.ko.xz) has been updated to version 10.02.06.200-k.
- Driver for Microchip Smart Family Controller (smartpqi.ko.xz) has been updated to version 2.1.12-055.

Graphics and miscellaneous driver updates

- Standalone drm driver for the VMware SVGA device (vmwgfx.ko.xz) has been updated to version 2.18.1.0.

CHAPTER 7. BUG FIXES

This part describes bugs fixed in Red Hat Enterprise Linux 8.6 that have a significant impact on users.

7.1. INSTALLER AND IMAGE CREATION

The network --defroute option now works correctly in the %include script

Previously, the **network --defroute** option got ignored when used in the **%include** script during the kickstart installation. As a consequence, the device was set as the default route.

With this update, the kickstart installation does not ignore the **network --defroute** option added in the **%include** script and the network connection is configured as expected.

(BZ#1990145)

Users can now specify user accounts in the RHEL for Edge Installer blueprint

Previously, performing an update on your blueprint without a user account defined in the RHEL for Edge Commit for the upgrade, such as adding a rpm package, would cause users to be locked out of their system, after the upgrade was applied. It caused users to have redefine user accounts when upgrading an existing system. This issue has been fixed to allow users to specify user accounts in the RHEL for Edge Installer blueprint, that creates a user on the system at installation time, rather than having the user as part of the **ostree** commit.

(BZ#1951936)

osbuild no longer fails to build an ISO image bigger than 4GB

Image Builder users can create a customized image by adding additional packages. If the total size of the packages and their dependencies exceeded 4GB size, users of RHEL 8.5 and earlier releases would see the following error:

```
ubprocess.CalledProcessError: Command '['/usr/bin/xorrisofs', '-verbose', '-V', 'RHEL-8-5-0-BaseOS-x86_64', '-sysid', 'LINUX', '-isohybrid-mbr', '/usr/share/syslinux/isohdpfx.bin', '-b', 'isolinux/isolinux.bin', '-c', 'isolinux/boot.cat', '-boot-load-size', '4', '-boot-info-table', '-no-emul-boot', '-rock', '-joliet', '-eltorito-alt-boot', '-e', 'images/efiboot.img', '-no-emul-boot', '-isohybrid-gpt-basdat', '-o', '/run/osbuild/tree/installer.iso', '/run/osbuild/inputs/tree']' returned non-zero exit status 32.
```

The problem happened because the ISO 9660 Level Of Interchange **-isolevel 3** argument was not passed to the **xorrisofs** command. To work around the problem, users had to permanently alter the ISO level value to 3.

With the RHEL 8.6 release, the problem has been fixed, and users no longer need to permanently alter the ISO level value.

(BZ#2056451)

7.2. SOFTWARE MANAGEMENT

Running createrepo_c --update on a modular repository now preserves modular metadata in it

Previously, when running the **createrepo_c --update** command on an already existing modular repository without the original source of modular metadata present, the default policy was to remove all additional metadata including modular metadata from this repository, which, consequently, broke it. To

preserve metadata, it required running the **createrepo_c --update** command with the additional **--keep-all-metadata** option.

With this update, you can preserve modular metadata on a modular repository by running **createrepo_c --update** without any additional option.

To remove additional metadata, you can use the new **--discard-additional-metadata** option.

(BZ#1992209)

7.3. SHELLS AND COMMAND-LINE TOOLS

Errors during the installation of the info subpackage do not happen anymore

Previously the **fix-info-dir** script expected the existence of a **/dev/null** file. With a new version of the **texinfo** package for software documentation, the installation of the **info** subpackage does not fail on systems that do not contain the **/dev/null** special file. Now the **fix-info-dir** script does not expect the existence of the **/dev/null** file, and avoids the possibility of an infinite loop.

(BZ#2022201)

ReaR backs up a system with an unused LVM physical volume correctly

Previously, **ReaR** produced an incorrect disk layout when an unused LVM physical volume (PV) was present on the system. As a result, ReaR commands that need to produce the disk layout, such as the **mkrescue**, **mkbackup**, **mkbackuponly**, **savelayout** commands, aborted with the error message:

```
ERROR: LVM 'lvmdev' entry in /var/lib/rear/layout/disklayout.conf where volume_group or device is empty or more than one word
```

With this update, **ReaR** now comments out unused PVs in the disk layout file and is thus able to back up a system with unused PVs correctly.

(BZ#2048454)

ReaR does not incorrectly exclude multipath devices from the backup

Previously, **ReaR** was incorrectly excluding certain multipath devices whose names contained the names of multipath devices that should have been excluded from the backup.

For example, if a device named **/dev/mapper/mpatha** was excluded from the backup, then a second device named **/dev/mapper/mpathaa** would be incorrectly excluded as well. This would occur with more than 26 multipath devices.

The bug has been fixed and **ReaR** now does not exclude multipath devices from the backup unless they should be excluded. Note that you have to specify **AUTOEXCLUDE_MULTIPATH=n** in the **ReaR** configuration file if there are multipath devices that should be included in the backup, otherwise **ReaR** excludes all multipath devices automatically. This behavior has not changed.

(BZ#2049091)

7.4. SECURITY

Remote users are no longer repetitively prompted to access smart cards

Previously, the **polkit** policy for the **pcscd** daemon incorrectly requested user interaction. As a

consequence, non-local and non-privileged users could not access smart cards and encountered large numbers of prompts. With this update, the **pcsc-lite** package policy no longer includes the interactive prompts. As a result, remote card users are no longer repeatedly asked for privilege escalation.

For additional information about adjusting the policy to escalate privileges of non-privileged users, see Controlling access to smart cards using polkit in Security hardening in RHEL product documentation.

(BZ#1928154)

64-bit IBM Z systems no longer become unbootable when installing in FIPS mode

Previously, the **fips-mode-setup** command with the **--no-bootcfg** option did not execute the **zipl** tool. Because **fips-mode-setup** regenerates the initial RAM disk (**initrd**), and the resulting system needs an update of **zipl** internal state to boot, this put 64-bit IBM Z systems into an unbootable state after installing in FIPS mode. With this update, **fips-mode-setup** now executes **zipl** on 64-bit IBM Z systems even if invoked with **--no-bootcfg**, and as a result, the newly installed system boots successfully.

(BZ#2020295)

crypto-policies can disable ChaCha20 in OpenSSL

Previously, the **crypto-policies** component used a wrong keyword to disable the ChaCha20 cipher in OpenSSL. As a consequence, use of ChaCha20 in TLS 1.2 in OpenSSL could not be disabled through **crypto-policies**. With this update, **crypto-policies** use the **-CHACHA20** keyword instead of the **-CHACHA20-POLY1305** keyword. As a result, you can now use **crypto-policies** to disable the use of the ChaCha20 cipher in OpenSSL for both TLS 1.2 and TLS 1.3.

(BZ#2023734)

systemd can now execute files from /home/*user*/bin

Previously, **systemd** services could not execute files from the **/home/*user*/bin/** directory because the SELinux policy did not include the policy rules that allow such access. Consequently, the **systemd** services failed and eventually logged the Access Vector Cache (AVC) denial Audit messages. This update adds the missing SELinux rules that allow access, and **systemd** services can now correctly execute commands from **/home/*user*/bin/**.

(BZ#1860443)

STIG-specific default banner text removed from other profiles

Previously, banner text from the STIG profile was used as default by other profiles that did not have a default text defined, such as CIS. As a consequence, systems using these profiles were configured with the specific text required by DISA. With this update, a generic default text was created and a standard CIS banner aligned with the guidelines was defined. As a result, profiles based on guidelines which explicitly require a text banner are now aligned with the requirements and set the correct text.

(BZ#1983061)

ANSSI Enhanced Profile correctly selects the "Ensure SELinux State is Enforcing" rule

Previously, the ANSSI Enhanced profile (**anssi_bp28_enhanced**) did not select the "Ensure SELinux State is Enforcing" (**selinux_state**) rule. This update modified the rule selection and now the ANSSI Enhanced Profile selects the "Ensure SELinux State is Enforcing" rule.

(BZ#2053587)

Descriptions for restorecon and seunshare SSG rules fixed

Previously, descriptions for rules "Record Any Attempts to Run restorecon" (CCE-80699-2) and "Record Any Attempts to Run seunshare" (CCE-80933-5) were incorrect. With this update, the descriptions of these rules are aligned with the automated OVAL check. As a result, applying the fix recommended in the description now correctly fixes these rules.

(BZ#2023569)

The CIS profile no longer automatically disables IPv6

Previously, the CIS profile for RHEL 8 provided inappropriate automated remediation for recommendation "3.6 Disable IPv6", which disabled IPv6 by configuring **/etc/modprobe.d/ipv6.conf** to prevent the IPv6 module from loading. This could have undesired effects on the dependent features and services. In RHEL 8 CIS Benchmark v1.0.1, the recommendation 3.6 must be implemented manually, and therefore the RHEL8 CIS profiles do not apply any remediation for this configuration item. As a result, the CIS profile is aligned with the benchmark and does not disable IPv6 automatically. To disable IPv6 manually by configuring GRUB2 or sysctl settings as recommended by CIS, see How do I disable or enable the IPv6 protocol in Red Hat Enterprise Linux?.

(BZ#1990736)

CIS profile no longer blocks the SSH service

Previously, the **xccdf_org.ssgproject.content_rule_file_permissions_sshd_private_key** rule by default set the permissions to **640** on SSH private keys. As a consequence, the SSH daemon did not start. This update removes the **file_permissions_sshd_private_key** rule from the CIS profile and as a result, the SSH service works correctly.

(BZ#2002850)

Files in /usr/share/audit/sample-rules are now accepted by SCAP rules

Previously, according to the description of SCAP rules **xccdf_org.ssgproject.content_rule_audit_ospp_general** and **xccdf_org.ssgproject.content_rule_audit_immutable_login_uids**, users were able to make systems compliant by copying appropriate files from the **/usr/share/audit/sample-rules** directory. However, OVAL checks of these rules failed, and the system was consequently marked as non-compliant after the scan. With this update, the OVAL checks now accept the files from **/usr/share/audit/sample-rules**, and the SCAP rules pass successfully.

(BZ#2000264)

ANSSI Kickstart now reserves enough disk space

Previously, GUI installation required more disk space than ANSSI Kickstart reserved in the **/usr** partition. As a consequence, RHEL 8.6 GUI installations failed, with an error message stating that **At least 429 MB more space needed on the /usr filesystem**. This update increases the disk space for the **/usr** partition, and RHEL 8.6 installations using the ANSSI Kickstarts provided in the **scap-security-guide** now completes successfully.

(BZ#2058033)

Remediations of GRUB2 arguments are now persistent

Previously, the remediations for GRUB2 rules that set kernel arguments were using incorrect procedures and the configuration changes were not persistent across kernel upgrades. As a consequence, the remediations had to be reapplied with every kernel upgrade. With this update, remediations use the **grubby** tool that configures GRUB2 in a persistent way.

(BZ#2030966)

scap-workbench no longer hangs when scanning remote systems from RHEL 8 hosts

Previously, sending content files to the scanned system would hang and the **scap-workbench** utility could not complete the scan. This was due to a bug in the kernel which blocked executed Qt subprocesses. As a consequence, scanning of remote systems using the **scap-workbench** command from RHEL 8 hosts did not work. With this update, the underlying kernel bug is fixed, and therefore remote scans no longer hang on copying files to a remote system and successfully finish.

(BZ#2051890)

usbguard-notifier no longer logs too many error messages to the Journal

Previously, the **usbguard-notifier** service did not have inter-process communication (IPC) permissions for connecting to the **usbguard-daemon** IPC interface. Consequently, **usbguard-notifier** failed to connect to the interface, and it wrote a corresponding error message to the Journal. Because **usbguard-notifier** started with the **--wait** option, which ensured that **usbguard-notifier** attempted to connect to the IPC interface each second after a connection failure, by default, the log contained an excessive amount of these messages soon.

With this update, **usbguard-notifier** does not start with **--wait** by default. The service attempts to connect to the daemon only three times in the 1-second intervals. As a result, the log contains three such error messages at maximum.

(BZ#2000000)

Ambient capabilities are now applied correctly to non-root users

As a safety measure, changing a UID (User Identifier) from root to non-root nullifies permitted, effective, and ambient sets of capabilities.

However, the **pam_cap.so** module is unable to set ambient capabilities because a capability needs to be in both the permitted and the inheritable set to be in the ambient set. In addition, the permitted set gets nullified after changing the UID (for example by using the **setuid** utility), so the ambient capability cannot be set.

To fix this problem, the **pam_cap.so** module now supports the **keepcaps** option, which allows a process to retain its permitted capabilities after changing the UID from root to non-root. The **pam_cap.so** module now also supports the **defer** option, which causes **pam_cap.so** to reapply ambient capabilities within a callback to **pam_end()**. This callback can be used by other applications after changing the UID.

Therefore, if the **su** and **login** utilities are updated and PAM-compliant, you can now use **pam_cap.so** with the **keepcaps** and **defer** options to set ambient capabilities for non-root users.

(BZ#1950187)

The usbguard-selinux package is no longer dependent on usbguard

Previously, the **usbguard-selinux** package was dependent on the **usbguard** package. This, in combination with other dependencies of these packages, led to file conflicts when installing **usbguard**. As a consequence, this prevented the installation of **usbguard** on certain systems. With this version, **usbguard-selinux** no longer depends on **usbguard**, and as a result, **yum** can install **usbguard** correctly.

(BZ#1963271)

audisp-remote now correctly detects the availability of the remote locations

Previously, the **audisp-remote** plugin did not detect that remote services became unavailable. As a consequence, the **audisp-remote** process would enter a state with high CPU usage. With this update,

audisp-remote can properly detect remote services becoming unavailable. As a result, the process no longer enters a high-CPU-usage state.

(BZ#1906065)

Clevis no longer stops on certain configurations before automated unlocking

Previously, the Clevis utility, which performs automated unlocking of LUKS-encrypted volumes, stopped on certain system configurations. Consequently, encrypted volumes were not unlocked automatically, and the administrator had to provide a passphrase manually. In some cases, Clevis restarted after the administrator pressed Enter and unlocked the encrypted volumes. With this update, the utility has been fixed to not stop on these configurations, and the process of automated unlocking now works properly.

(BZ#2018292)

7.5. NETWORKING

NetworkManager now uses a static IPv4 IP address as primary

The main purpose of primary and secondary addresses is to enable source address selection for connections that are not yet bound to an IP address. For these connections, the kernel automatically chooses an address. In a NetworkManager connection profile, you can configure a static IPv4 address and DHCP at the same time for one connection. Previously, if you configured a connection with DHCP and a static IPv4 address from the same range as the one provided by the DHCP server, NetworkManager incorrectly assigned the IP address that it received from the DHCP server as primary and the static IP address as secondary.

RHEL 8.6 changes this to the intended behavior. As a result, if you configure both a static IPv4 address and DHCP in one connection profile, the static IP address is now always the primary and the address received from the DHCP server the secondary. Additionally, NetworkManager now also sets the **src** attribute for routes assigned by a DHCP server. With this functionality, destinations reachable through these routes use the IP address received from the DHCP server as a source.

(BZ#2096256)

7.6. KERNEL

The dmidecode --type 17 command now successfully decodes DDR5 memory information

Previously, the **dmidecode** command failed to decode the DDR5 memory information. Consequently, **dmidecode --type 17** returned the **<OUT OF SPEC>** message. The latest update of the package (**dmidecode-3.3-3.el8**) has fixed this problem. As a result, **dmidecode --type 17** now successfully decodes DDR5 memory information.

(BZ#2027665)

kdump no longer fails on KVM virtual machines that use the default amount of memory

Previously, **kdump** failed on some kernel-based virtual machines (KVM) that uses the default amount of memory. Consequently, the crash kernel failed to capture the crash dump file with following error:

/bin/sh: error while loading shared libraries: libtinfo.so.6: cannot open shared object file: No such file or directory

With this update, the problem has been fixed and **kdump** works correctly on KVM virtual machines that use the default amount of memory.

(BZ#2004000)

Tunnel offloading now works as expected and supports the available hardware

Previously, the driver was not setting certain feature flags. Hence, tunnel offloading was not working as expected. In this update, the driver sets the required flags to enable tunnel offloading and works as expected.

(BZ#1910885)

Fixed the kernel warning while setting the rx ring buffer to max

Previously, an internal function expecting clean input was called with a reused and already initialized structure. It caused the kernel to give the warning message: “missing unregister, handled but fix driver”. This update fixes the bug, reinitializing the structure before trying to register it again.

(BZ#2040171)

7.7. FILE SYSTEMS AND STORAGE

xfsrestore command works correctly while restoring a backup

Previously, while restoring a backup created using the **xfsdump** command, **xfsrestore** created an orphanage directory. As a consequence, a few files were moved into the created orphanage directory with the following messages:

```
# xfsdump -L test -M test -f /scratch.dmp /mnt/test
...
xfsdump: NOTE: root ino 128 differs from mount dir ino 1024, bind mount?
...
xfsdump: Dump Status: SUCCESS

# xfsrestore -f /scratch.dmp /mnt/restore/
...
xfsrestore: restoring non-directory files
xfsrestore: NOTE: ino 128 salvaging file, placing in orphanage/1024.0/dir17/file60
xfsrestore: NOTE: ino 129 salvaging file, placing in orphanage/1024.0/dir17/file61
xfsrestore: NOTE: ino 130 salvaging file, placing in orphanage/1024.0/dir17/file62
xfsrestore: NOTE: ino 131 salvaging file, placing in orphanage/1024.0/dir17/file63
xfsrestore: NOTE: ino 132 salvaging file, placing in orphanage/1024.0/dir17/file64
xfsrestore: NOTE: ino 133 salvaging file, placing in orphanage/1024.0/dir17/file65
xfsrestore: NOTE: ino 134 salvaging file, placing in orphanage/1024.0/dir17/file66
...
```

With this update, the problem has been fixed and **xfsrestore** now works correctly.

(BZ#2020494)

The multipathd.socket unit file no longer disables multipathd after too many startup attempts

Previously, the starting conditions for **multipathd** in the **multipath.service** unit file differed from the triggering conditions in **multipathd.socket**. Consequently, the unit file repeatedly tried to start **multipathd** and failed. This resulted in disabling **multipathd** after too many failed attempts. With this fix, the starting conditions for **multipathd.socket** and **multipathd.service** have been set to the same values. As a result, the **multipathd.socket** unit file no longer attempts to start **multipathd** where the starting conditions for **multipathd.service** are not met.

(BZ#2008101)

Protection uevents no longer cause reload failure of multipath devices

Previously, when a **read-only** path device was rescanned, the kernel sent out two write protection uevents - one with the device set to **read/write**, and the following with the device set to **read-only**. Consequently, upon detection of the **read/write** uevent on a path device, **multipathd** tried to reload the multipath device, which caused a reload error message. With this update, **multipathd** now checks that all the paths are set to **read/write** before reloading a device read/write. As a result, **multipathd** no longer tries to reload **read/write** whenever a **read-only** device is rescanned.

(BZ#2009624)

7.8. COMPILERS AND DEVELOPMENT TOOLS

The -j flag now works when used in a Makefile

Previously, when you added the **-j** flag to MAKEFLAGS inside the Makefile, the targets were built sequentially instead of in parallel. This bug has been fixed, and now the targets are built at the same time when you use the **-j** flag in the Makefile.

(BZ#2004246)

Statically linked applications no longer crash

Previously, the initialization code of the dynamic loader, which is linked into statically linked binaries, did not initialize a link map variable correctly. Consequently, statically linked applications crashed if **LD_LIBRABY__PATH** contained a dynamic token string. With this update statically linked applications no longer crash.

(BZ#1934162)

pthread_once() in glibc has been fixed to correctly support C++ exceptions

Previously, the **pthread_once()** implementation could result in a hang when using **libstdc++** library functions. For example **libstdc++**'s **std::call_once()** called a function that threw an exception which would result in a hang. With this update, **pthread_once()** is fixed and no longer hangs when an exception is thrown.

(BZ#2007327)

7.9. IDENTITY MANAGEMENT

Certmonger can now automatically renew SCEP certificates with AD when challengePassword is required for enrollment

Previously, requests for renewal of SCEP certificates sent by **certmonger** to an Active Directory (AD) Network Device Enrollment Service (NDES) server included the **challengePassword** used to originally obtain the certificate. However, AD treats **challengePassword** as a one-time password (OTP). As a consequence, the renewal request was rejected.

This update adds the **challenge_password_otp** option to **certmonger**. When enabled, this option prevents **certmonger** from sending the OTP with the SCEP renewal request. The administrator must also add the **DisableRenewalSubjectNameMatch** entry with a value of **1** to the **HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Cryptography\MSCEP** subkey in the AD registry. With this modification, AD no longer requires the signer certificate and requested certificate subject names to match. As a result, the SCEP certificate renewal is successful.

To configure **certmonger** and the AD server for SCEP renewals to work:

1. Open **regedit** on the AD server.
2. In the **HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Cryptography\MSCEP** subkey, add a new 32-bit REG_DWORD entry **DisableRenewalSubjectNameMatch** and set its value to **1**.
3. On the server where **certmonger** is running, open the **/etc/certmonger/certmonger.conf** file and add the following section:

```
[scep]
challenge_password_otp = yes
```

4. Restart certmonger:

```
# systemctl restart certmonger
```

(BZ#1577570)

FreeRADIUS proxy server no longer stops working when a second FreeRADIUS server is unavailable

When a FreeRADIUS server is configured as a proxy server it forwards request messages to another FreeRADIUS server. Previously, if the connection between these two servers was interrupted, the FreeRADIUS proxy server stopped working. With this fix, the FreeRADIUS proxy server is now able to reestablish a connection when the other server becomes available.

(BZ#2030173)

Authenticating to Directory Server in FIPS mode with PBKDF2-hashed passwords now works as expected

When Directory Server runs in Federal Information Processing Standard (FIPS) mode, the **PK11_ExtractKeyValue()** function is not available. As a consequence, users with a password-based key derivation function 2 (PBKDF2) hashed password could not authenticate to the server when FIPS mode was enabled. With this update, Directory Server now uses the **PK11_Decrypt()** function to get the password hash data. As a result, authenticating to Directory Server in FIPS mode now works for users with PBKDF2-hashed passwords.

(BZ#2033398)

Socket activation of SSSD succeeds when the SSSD cache is mounted in tmpfs as the SSSD user

Previously, socket activation of SSSD would fail if the SSSD cache was mounted in a **tmpfs** temporary file system because the **/var/lib/sss/db/config.ldb** SSSD configuration file was not owned by the **sssd** user. With this fix, SSSD creates the **config.ldb** file as the **sssd** user and socket activation succeeds. If you have mounted the **/var/lib/sssd/db/** SSSD cache directory in **tpmfs**, you must remount it as the **sssd** user so SSSD can create the **config.ldb** file in that location.

WARNING

Perform the following steps only if you have mounted your SSSD cache into **tmpfs** for faster performance according to the steps in the Tuning performance in Identity Management guide. In standard circumstances, Red Hat recommends using the default location for the SSSD cache, on standard disk storage, instead.

Procedure

1. Confirm that **/var/lib/sss/db** is a mount point:

```
# mount -t tmpfs | grep /var/lib/sss/db
tmpfs on /var/lib/sss/db type tmpfs
(rw,relatime,rootcontext=system_u:object_r:sssd_var_lib_t:s0,seclabel,size=307200k,mode=700
)
```

2. If **/var/lib/sss/db** is a valid mount point, check if it is owned by the **root** user:

```
# ls -l /var/lib/sss | grep db
drwx------. 2 *root root* 40 Jul 26 04:48 db
```

3. If the **db** directory is a mount point and it is owned by the **root** user, add **uid=sssd,gid=sssd** to the corresponding entry in the **/etc/fstab** file to mount it as the SSSD user:

```
tmpfs /var/lib/sss/db/ tmpfs
size=300M,mode=0700,*uid=sssd,gid=sssd*,rootcontext=system_u:object_r:sssd_var_lib_t:s0
0 0
```

4. Remount the directory and restart the SSSD service:

```
# systemctl stop sssd
# umount /var/lib/sss/db
# mount /var/lib/sss/db
# systemctl start sssd
```

Verification

- Verify that the **/var/lib/sss/db** directory is owned by the **sssd** user:

```
# ls -l /var/lib/sss | grep db
drwx------. 2 sssd sssd 160 Jul 26 05:00 db
```

(BZ#2108316)

7.10. GRAPHICS INFRASTRUCTURES

Matrox GPU with a VGA display now works as expected

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

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

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

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

(BZ#1953926)

7.11. RED HAT ENTERPRISE LINUX SYSTEM ROLES

A playbook using the Metrics role completes successfully on multiple runs even if the Grafana admin password is changed

Previously, changes to the Grafana **admin** user password after running the Metrics role with the **metrics_graph_service: yes** boolean caused failure on subsequent runs of the Metrics role. This led to failures of playbooks using the Metrics role, and the affected systems were only partially set up for performance analysis. Now, the Metrics role uses the Grafana **deployment** API when it is available and no longer requires knowledge of username or password to perform the necessary configuration actions. As a result, a playbook using the Metrics role completes successfully on multiple runs even if the administrator changes the Grafana **admin** password.

(BZ#1967321)

The SSHD System Role uses the correct template file

In RHEL 8.5, the SSHD System Role used a wrong template file. As a consequence, the generated **sshd_config** file did not contain the **# Ansible managed** comment. The missing comment did not affect any functionality on the system. With this update, the system role uses the correct template file and **sshd_config** contains the correct **# Ansible managed** comment.

(BZ#2040038)

The Networking System Role no longer fails to set a DNS search domain if IPv6 is disabled

Previously, the **nm_connection_verify()** function of the **libnm** library did not ignore the DNS search domain if the IPv6 protocol was disabled. As a consequence, when you used the Networking RHEL System Role and set **dns_search** together with **ipv6_disabled: true**, the System Role failed with the following error:

```
nm-connection-error-quark: ipv6.dns-search: this property is not allowed for 'method=ignore' (7)
```

With this update, the **nm_connection_verify()** function ignores the DNS search domain if IPv6 is disabled. As a consequence, you can use **dns_search** as expected, even if IPv6 is disabled.

(BZ#2041627)

The nm provider in the Networking System Role now correctly manages bridges

Previously, if you used the **initscripts** provider, the Networking System Role created an **ifcfg** file which

configured NetworkManager to mark bridge interfaces as unmanaged. Also, NetworkManager failed to detect followup **initscript** actions. For example, the **down** and **absent** actions of initscript provider will not change the NetworkManager's understanding on unmanaged state of this interface if not reloading the connection after the **down** and **absent** actions. With this fix, the Networking System Role uses the **NM.Client.reload_connections_async()** function to reload NetworkManager on managed hosts with NetworkManager 1.18. As a result, NetworkManager manages the bridge interface when switching the provider from **initscript** to **nm**.

(BZ#2034908)

The SSH server role now detects FIPS mode and handles tasks correctly in FIPS mode

Previously, when managing RHEL8 and older systems in FIPS mode, one of the default hostkeys was not allowed to be created. As a consequence, the SSH server role operation failed to generate the **not allowed key** type when invoked. With this fix, the SSH server role detects FIPS mode and adjusts default hostkey list accordingly. As a result, the SSH server role can now manage systems in FIPS mode with default hostkeys configuration.

(BZ#1979714)

The Logging System Role no longer calls tasks multiple times

Previously, the Logging role was calling tasks multiple times that should have been called only once. As a consequence, the extra task calls slowed down the execution of the role. With this fix, the Logging role was changed to call the tasks only once, improving the Logging role performance.

(BZ#2005727)

RHEL System Roles now handle multi-line ansible_managed comments in generated files

Previously, some of the RHEL System Roles were using **# {{ ansible_managed }}** to generate some of the files. As a consequence, if a customer had a custom multi-line **ansible_managed** setting, the files would be generated incorrectly. With this fix, all of the system roles use the equivalent of **{{ ansible_managed | comment }}** when generating files so that the **ansible_managed** string is always properly commented, including multi-line **ansible_managed** values. Consequently, generated files have the correct multi-line **ansible_managed** value.

(BZ#2006231)

The Logging role no longer misses quotes for the immark module interval value

Previously, the "interval" field value for the **immark** module was not properly quoted, because the **immark** module was not properly configured. This fix ensures that the "interval" value is properly quoted. Now, the **immark** module works as expected.

(BZ#2021678)

The group option no longer keeps certificates inaccessible to the group

Previously, when setting the group for a certificate, the **mode** was not set to allow group read permission. As a consequence, group members were unable to read certificates issued by the Certificate role. With this fix, the group setting now ensures that the file mode includes group read permission. As a result, the certificates issued by the Certificate role for groups are accessible by the group members.

(BZ#2021683)

The /etc/tuned/kernel_settings/tuned.conf file has a proper ansible_managed header

Previously, the Kernel settings RHEL System Role had a hard-coded value for the **ansible_managed**

header in the **/etc/tuned/kernel_settings/tuned.conf** file. Consequently, users could not provide their custom **ansible_managed** header. In this update, the problem has been fixed so that **kernel_settings** updates the header of **/etc/tuned/kernel_settings/tuned.conf** with user's **ansible_managed** setting. As a result, **/etc/tuned/kernel_settings/tuned.conf** has a proper **ansible_managed** header.

(BZ#2047504)

The logging_purge_confs option no longer fails to delete unnecessary configuration files

Previously, the **logging_purge_confs** variable was prepared to delete unnecessary logging configuration files, but failed to clean them up. Consequently, even though the **logging_purge_confs** variable was set to true, unnecessary configuration files were not cleaned up, but left in the configuration directory. This issue is now fixed and the **logging_purge_confs** variable has been redefined to work as follows.

- If **logging_purge_confs** is set to **true**, it removes files in **rsyslog.d** which do not belong to any rpm packages. That includes configuration files generated by the previous **logging** role run. The **logging_purge_confs** default value is **false**.

(BZ#2040812)

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#2064388)

Configuration by the Metrics role now follows symbolic links correctly

When the **mssql pcp** package is installed, the **mssql.conf** file is located in **/etc/pcp/mssql/** and is targeted by the symbolic link **/var/lib/pcp/pmdas/mssql/mssql.conf**. Previously, however, the Metrics role overwrote the symbolic link instead of following it and configuring **mssql.conf**. Consequently, running the Metrics role changed the symbolic link to a regular file and the configuration therefore only affected the **/var/lib/pcp/pmdas/mssql/mssql.conf** file. This resulted in a failed symbolic link, and the main configuration file **/etc/pcp/mssql/mssql.conf** was not affected by the configuration. The issue is now fixed and the **follow: yes** option to follow the symbolic link has been added to the Metrics role. As a result, the Metrics role preserves the symbolic links and correctly configures the main configuration file.

(BZ#2058655)

The Kernel settings 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#2058772)

The Kdump System Role does not ignore hosts anymore

Previously, the Kdump role ignored managed nodes that do not have memory reserved for crash kernel, and consequently completed with the "Success" status even when not configuring the system correctly. The role has been redesigned to fail in cases where managed nodes do not have memory reserved for crash kernel, and to prompt the user to set the **kdump_reboot_ok** variable to **true** to correctly

configure kdump on managed nodes. As a result, the Kdump role now does not ignore hosts, and either completes successfully with the correct configuration, or fails with an error message describing what users need to do to fix the issue.

(BZ#2029605)

The Firewall System Role now reloads the firewall immediately when target changes

Previously, the Firewall System Role was not reloading the firewall when the **target** parameter has been changed. With this fix, the Firewall role reloads the firewall when the **target** changes, and as a result, the **target** change is immediate and available for subsequent operations.

(BZ#2057172)

Default pcsd permissions for HA Cluster System Role now allow access for group haclient

Previously, when a user ran the HA Cluster System Role with the default **pcsd** permissions that were set with the **ha_cluster_pcs_permission_list** variable, only members of the group **hacluster** had access to the cluster. With this fix, the default **pcsd** permissions allow the group **haclient** to manage the cluster and all members of **haclient** can now access and manage the cluster.

(BZ#2049747)

7.12. VIRTUALIZATION

strict NUMA binding policy no longer allows for moving runtime memory

Previously, when the **strict** NUMA binding policy was enabled in a VM (**<memory mode='strict'/>**), attempting to move runtime memory from that VM to another NUMA node in some cases partly or completely failed. To avoid this problem, the **strict** policy now completely prohibits moving runtime memory.

In addition, the **restrictive** policy has been added, which works like the **strict** policy did previously. This means that it does allow for moving runtime memory to other NUMA nodes, but cannot ensure that the memory is moved completely.

(BZ#2014369)

multifd migration now works reliably

Previously, attempting to migrate a virtual machine (VM) using the **multifd** feature of QEMU caused the migration to fail and the VM to terminate unexpectedly. The underlying code has been fixed, and **multifd** migration now works as expected.

(BZ#1982993)

VM migration and snapshots no longer failing due to virtio-balloon

Previously, attempting to migrate a virtual machine (VMs) with a more recent guest operating system (such as RHEL 9) failed if the VM was using the **virtio-balloon** device. Similarly, creating a snapshot of such a VM failed. This update fixes a bug in the **page poison** feature of **virtio-balloon**, which prevents the described problem from occurring.

(BZ#2004416)

Hot unplugging an IBMVFC device on PowerVM now works as expected

Previously, when using a virtual machine (VM) with a RHEL 8 guest operating system on the PowerVM

hypervisor, attempting to remove an IBM Power Virtual Fibre Channel (IBMVFC) device from the running VM failed. Instead, it displayed an **outstanding translation** error. The underlying code has been fixed and live hot unplugs of IBMVFC device now work correctly on PowerVM.

(BZ#1959020)

7.13. CONTAINERS

Rootless containers created in RHEL 8.5 and earlier using fuse-overlayfs now recognize removed files

Previously, in RHEL 8.4 and earlier, rootless images and containers were created or stored using the fuse-overlayfs file system. Using such images and containers in RHEL 8.5 and later introduced problems for unprivileged users using the overlayfs implementation provided by the kernel and who had removed files or directories from a container or from an image in RHEL 8.4. This problem did not apply to containers created by the root account.

As a consequence, files or directories that were removed from a container or from an image were marked as such using the whiteout format when using the fuse-overlayfs file system. However, due to differences in the format, the kernel overlayfs implementation did not recognize the whiteout format created by fuse-overlayfs. As a result, any removed files or directories still appeared. This problem did not apply to containers created by the root account.

With this update, the problem is solved.

(JIRA:RHELPLAN-92741)

CHAPTER 8. TECHNOLOGY PREVIEWS

This part provides a list of all Technology Previews available in Red Hat Enterprise Linux 8.6.

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

8.1. RHEL FOR EDGE

FDO process available as a Technology Preview

The FDO process for automatic provisioning and onboarding RHEL for Edge images is available as a Technology Preview. With that, you can build a RHEL for Edge Simplified Installer image, provision it to a RHEL for Edge image, and use the FDO (FIDO device onboarding) process to automatically provision and onboard your Edge devices, exchange data with other devices and systems connected on the networks. As a result, the FIDO device onboarding protocol performs device initialization at the manufacturing stage and then late binding to actually use the device.

(BZ#1989930)

8.2. 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)

8.3. 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, an 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)

8.4. 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)

8.5. 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: https://www.kernel.org/doc/Documentation/filesystems/overlayfs.txt.

For more information about OverlayFS, see the Linux kernel documentation: https://www.kernel.org/doc/Documentation/filesystems/overlayfs.txt.

(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)

8.6. 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)

8.7. 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)

8.8. 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**), 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)

8.9. 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)

8.10. THE WEB CONSOLE

Stratis available as a Technology Preview in the RHEL web console

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

To learn more about Stratis, see What is Stratis.

(JIRA:RHELPLAN-108438)

8.11. 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)

8.12. CONTAINERS

Toolbox is available as a Technology Preview

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

(JIRA:RHELPLAN-77238)

The Netavark network stack is available as a Technology Preview

Before Podman version 4.1.1-7, the Netavark network stack for containers is available as a Technology Preview.

This network stack has the following capabilities:

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

NOTE

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

(JIRA:RHELPLAN-137622)

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 9. 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 .

9.1. 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)

9.2. 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)

9.3. 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 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 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)

9.4. 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 version 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)

9.5. 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 8.6 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. In 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)

9.6. 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 will be 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 example, 8.4.x will have a one-year support window, but 8.4.x+1 will have 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)

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)

9.7. 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)

9.8. 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)

9.9. 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)

9.10. 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)

9.11. 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 **dnf** 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)

9.12. 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.

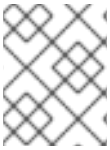
Jira:RHELDOCS-16612

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

9.13. 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)

9.14. 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)

9.15. 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)

The remotectl command is deprecated

The **remotectl** command has been deprecated and will not be available in future releases of RHEL. You can use the **cockpit-certificate-ensure** command as a replacement. However, note that **cockpit-certificate-ensure** does not have feature parity with **remotectl**. It does not support bundled certificates and keychain files and requires them to be split out.

(JIRA:RHELPLAN-147538)

9.16. 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 8.6:

```
# 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)

9.17. 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)

9.18. 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)

9.19. 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
- 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
- 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-emf
- eclipse-license
- 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-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-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
- 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
- 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
- jffi
- jflex
- jgit
- jline
- 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

- 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
- 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
- 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-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-mongo
- rubygem-mongo-doc
- 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
- 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

9.20. 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 9.1. 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)

Device ID	Driver	Device name
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
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

Device ID	Driver	Device name
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 9.2. 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

Device ID	Driver	Device name
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 10. KNOWN ISSUES

This part describes known issues in Red Hat Enterprise Linux 8.6.

10.1. 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)

10.2. 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)

10.3. 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 DNF upgrade can skip obsoleted packages

The patch for BZ#2095764, released with the RHBA-2022:5816 advisory, introduced the following regression: The DNF upgrade using security filters, such as the **--security** option, can skip upgrading obsoleted packages. This issue happens specifically when an installed package is obsoleted by a different available package, and an advisory exists for the available package.

Consequently, **dnf** leaves the obsoleted package in the system, and the security upgrade is not fully performed, potentially leaving the system in a vulnerable state.

To work around this problem, perform the full upgrade without security filters, or, first, verify that there are no obsoleted packages involved in the upgrade process.

(BZ#2095764)

10.4. SHELLS AND COMMAND-LINE TOOLS

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)

10.5. 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)

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)

10.6. 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)

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)

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)

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)

Ansible remediations require additional collections

With the replacement of Ansible Engine by the **ansible-core** package, the list of Ansible modules

provided with the RHEL subscription is reduced. As a consequence, running remediations that use Ansible content included within the **scap-security-guide** package requires collections from the **rhc-worker-playbook** package.

For an Ansible remediation, perform the following steps:

1. Install the required packages:

```
# dnf install -y ansible-core scap-security-guide rhc-worker-playbook
```

2. Navigate to the **/usr/share/scap-security-guide/ansible** directory:

```
# cd /usr/share/scap-security-guide/ansible
```

3. Run the relevant Ansible playbook using environment variables that define the path to the additional Ansible collections:

```
# ANSIBLE_COLLECTIONS_PATH=/usr/share/rhc-worker-playbook/ansible/collections/ansible_collections/ ansible-playbook -c local -i localhost, rhel9-playbook-cis_server_l1.yml
```

Replace ***cis_server_l1*** with the ID of the profile against which you want to remediate the system.

As a result, the Ansible content is processed correctly.

NOTE

Support of the collections provided in **rhc-worker-playbook** is limited to enabling the Ansible content sourced in **scap-security-guide**.

(BZ#2114981)

10.7. NETWORKING

The nm-cloud-setup service removes manually-configured secondary IP addresses from interfaces

Based on the information received from the cloud environment, the **nm-cloud-setup** service configures network interfaces. Disable **nm-cloud-setup** to manually configure interfaces. However, in certain cases, other services on the host can configure interfaces as well. For example, these services could add secondary IP addresses. To avoid that **nm-cloud-setup** removes secondary IP addresses:

1. Stop and disable the **nm-cloud-setup** service and timer:

```
# systemctl disable --now nm-cloud-setup.service nm-cloud-setup.timer
```

2. Display the available connection profiles:

```
# nmcli connection show
```

3. Reactive the affected connection profiles:

```
# nmcli connection up "<profile_name>"
```

As a result, the service no longer removes manually-configured secondary IP addresses from interfaces.

(BZ#2132754)

The primary IP address of an instance changes after starting the nm-cloud-setup service in Alibaba Cloud

After launching an instance in the Alibaba Cloud, the **nm-cloud-setup** service assigns the primary IP address to an instance. However, if you assign multiple secondary IP addresses to an instance and start the **nm-cloud-setup** service, the former primary IP address gets replaced by one of the already assigned secondary IP addresses. The returned list of metadata verifies the same. To work around the problem, configure secondary IP addresses manually to avoid that the primary IP address changes. As a result, an instance retains both IP addresses and the primary IP address does not change.

(BZ#2079849)

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)

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 lose their IP settings due to an unexpected change of the network interface name

The RDMA over Converged Ethernet (RoCE) interfaces lose their IP settings due to an unexpected change of the network interface name if both conditions are met:

- User upgrades from a RHEL 8.6 system or earlier.
- The RoCE card is enumerated by UID.

To workaround this problem:

1. Create the **/etc/systemd/network/98-rhel87-s390x.link** file with the following content:

```
[Match]
Architecture=s390x
KernelCommandLine=!net.naming-scheme=rhel-8.7

[Link]
NamePolicy=kernel database slot path
AlternativeNamesPolicy=database slot path
MACAddressPolicy=persistent
```

2. Reboot the system for the changes to take effect.

3. Upgrade to RHEL 8.7 or newer.

Note that RoCE interfaces that are enumerated by function ID (FID) and are non-unique, will still use unpredictable interface names unless you set the **net.naming-scheme=rhel-8.7** kernel parameter. In this case, the RoCE interfaces will switch to predictable names with the "ens" prefix.

(BZ#2169382)

10.8. KERNEL

Using net_prio or net_cls controllers in v1 mode deactivates some controllers of the cgroup-v2 hierarchy

In **cgroup-v2** environments, using either **net_prio** or **net_cls** controllers in v1 mode disables the hierarchical tracking of socket data. As a result, the **cgroup-v2** hierarchy for socket data tracking controllers is not active, and the **dmesg** command reports the following message:

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

(BZ#2046396)

Anaconda in some cases fails after entering the passphrase for encrypted devices

If **kdump** is disabled when preparing an installation and the user selects encrypted disk partitioning, the Anaconda installer fails with a traceback after entering the passphrase for the encrypted device.

To work around this problem, do one of the following:

- Create the encrypted disk partitioning before disabling **kdump**.
- Keep **kdump** enabled during the installation and disable it after the installation process is complete.

(BZ#2086100)

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 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 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)

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)

10.9. 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)

Using Device mapper multipath with the NVMe/TCP driver causes system instability

DM multipath is not supported with the NVMe/TCP driver. Using it causes sleeping functions in the kernel to be called in an atomic context, which then results in system instability.

To workaround the problem, enable native NVMe multipath. Do not use DM multipath tools. For RHEL 8, add the option **nvme_core.multipath=Y** to the kernel command line.

(BZ#2022359)

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)

10.10. 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)

MariaDB 10.5 does not warn about dropping a non-existent table when the OQGraph plug-in is enabled

When the **OQGraph** storage engine plug-in is loaded to the **MariaDB 10.5** server, **MariaDB** does not warn about dropping a non-existent table. In particular, when the user attempts to drop a non-existent table using the **DROP TABLE** or **DROP TABLE IF EXISTS** SQL commands, **MariaDB** neither returns an error message nor logs a warning.

Note that the **OQGraph** plug-in is provided by the **mariadb-oqgraph-engine** package, which is not installed by default.

(BZ#1944653)

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)

10.11. IDENTITY MANAGEMENT

Windows Server 2008 R2 and earlier no longer supported

In RHEL 8.4 and later, Identity Management (IdM) does not support establishing trust to Active Directory with Active Directory domain controllers running Windows Server 2008 R2 or earlier versions. RHEL IdM now requires SMB encryption when establishing the trust relationship, which is only available with Windows Server 2012 or later.

(BZ#1971061)

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)

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 the following error:

```
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)

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)

10.12. 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)

10.13. 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)

10.14. THE WEB CONSOLE

Removing USB host devices using the web console does not work as expected

When you attach a USB device to a virtual machine (VM), the device number and bus number of the USB device might change after they are passed to the VM. As a consequence, using the web console to remove such devices fails due to the incorrect correlation of the device and bus numbers. To workaround this problem, remove the **<hostdev>** part of the USB device, from the VM's XML configuration.

(JIRA:RHELPLAN-109067)

Attaching multiple host devices using the web console does not work

When you select multiple devices to attach to a virtual machine (VM) using the web console, only a single device is attached and the rest are ignored. To work around this problem, attach only one device at a time.

(JIRA:RHELPLAN-115603)

10.15. 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 2.12** 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 2.12** 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)

10.16. VIRTUALIZATION

Network traffic performance in virtual machines might be reduced

In some cases, RHEL 8.6 guest virtual machines (VMs) have somewhat decreased performance when handling high levels of network traffic.

(BZ#2069047)

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)

Live post-copy migration of VMs with failover VFs does not work

Currently, attempting to post-copy migrate a running virtual machine (VM) fails if the VM uses a device with the virtual function (VF) failover capability enabled. To work around the problem, use the standard migration type, rather than post-copy migration.

(BZ#2054656)

Live migrating VMs to a RHEL 8.6 Intel host from an earlier minor version of RHEL 8 does not work

Because the Intel Transactional Synchronization Extensions (TSX) feature has become deprecated, RHEL 8.6 hosts on Intel hardware no longer use the **hle** and **rtm** CPU flags. As a consequence, live migrating a virtual machine (VM) to a RHEL 8.6 host fails if the source host uses RHEL 8.5 or an earlier minor version of RHEL 8.

For more information on TSX deprecation, see the Red Hat KnowledgeBase .

(BZ#2134184)

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 to 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 the **--selinux-relabel** option to the **virt-customize** command.

(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)

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)

Windows Server 2022 guests in some cases boot very slowly on AMD Milan

Virtual machines (VMs) that use the Windows Server 2022 guest operating system and the **qemu64** CPU model currently take a very long time to boot on hosts with an AMD EPYC 7003 series processor (also known as **AMD Milan**).

To work work around the problem, do not use **qemu64** as the CPU model, because it is an unsupported setting for VMs in RHEL 8. For example, use the **host-model** CPU instead.

(BZ#2012373)

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)

10.17. RHEL IN CLOUD ENVIRONMENTS

SR-IOV performs suboptimally in ARM 64 RHEL 8 virtual machines on Azure

Currently, SR-IOV networking devices have significantly lower throughout and higher latency than expected in ARM 64 RHEL 8 virtual machines (VMs) running on a Microsoft Azure platform.

(BZ#2068429)

Setting static IP in a RHEL 8 virtual machine on a VMware host does not work

Currently, when using RHEL 8 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)

Starting a RHEL 8 virtual machine on AWS using cloud-init takes longer than expected

Currently, initializing an EC2 instance of RHEL 8 using the **cloud-init** service on Amazon Web Services (AWS) takes an excessive amount of time. 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)

10.18. SUPPORTABILITY

The getattachment command fails to download multiple attachments

The **getattachment** command is able to download only a single attachment, but 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** command.

(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)

10.19. 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)

Container images signed with a Beta GPG key can not be pulled

Currently, when you try to pull RHEL Beta container images, **podman** exits with the error message: **Error: Source image rejected: None of the signatures were accepted**. The images fail to be pulled due to current builds being configured to not trust the RHEL Beta GPG keys by default.

As a workaround, ensure that the Red Hat Beta GPG key is stored on your local system and update the existing trust scope with the **podman image trust set** command for the appropriate beta namespace.

If you do not have the Beta GPG key stored locally, you can pull it by running the following command:

```
sudo wget -O /etc/pki/rpm-gpg/RPM-GPG-KEY-redhat-beta
https://www.redhat.com/security/data/f21541eb.txt
```

To add the Beta GPG key as trusted to your namespace, use one of the following commands:

```
$ sudo podman image trust set -f /etc/pki/rpm-gpg/RPM-GPG-KEY-redhat-beta
registry.access.redhat.com/namespace
```

and

```
$ sudo podman image trust set -f /etc/pki/rpm-gpg/RPM-GPG-KEY-redhat-beta
registry.redhat.io/namespace
```

Replace *namespace* with *ubi9-beta* or *rhel9-beta*.

(BZ#2020301)

CHAPTER 11. INTERNATIONALIZATION

11.1. 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

11.2. 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#2033398, BZ#2016014, BZ#1817505, BZ#1780842
NetworkManager	BZ#1996617, BZ#2001563, BZ#2079849, BZ#1920398
SLOF	BZ#1910848
accel-config	BZ#1843266
anaconda	BZ#1990145, BZ#2050140, BZ#1914955, BZ#1929105
ansible-collection-microsoft-sql	BZ#2038256, BZ#2057651
apr	BZ#1819607
audit	BZ#1906065, BZ#1939406, BZ#1921658, BZ#1927884
authselect	BZ#1892761
bind9.16	BZ#1873486
bind	BZ#2013993
brltty	BZ#2008197
certmonger	BZ#1577570
clevis	BZ#1949289, BZ#2018292
cloud-init	BZ#2023940, BZ#2026587, BZ#1750862
cockpit	BZ#1666722
coreutils	BZ#2030661
corosync-qdevice	BZ#1784200
crash	BZ#1906482
createrepo_c	BZ#1992209, BZ#1973588

Component	Tickets
crypto-policies	BZ#2020295, BZ#2023734, BZ#2023744, BZ#1919155, BZ#1660839
cups-container	BZ#1913715
cups	BZ#2032965
device-mapper-multipath	BZ#2008101, BZ#2009624, BZ#2011699
distribution	BZ#1657927
dmidecode	BZ#2027665
dnf-plugins-core	BZ#1868047
dnf	BZ#1986657
ec2-images	BZ#1862930
edk2	BZ#1741615, BZ#1935497
fapolicyd	BZ#1939379, BZ#2054741
fence-agents	BZ#1977588, BZ#1775847
fido-device-onboard	BZ#1989930
firewalld	BZ#1980206, BZ#1871860
freeradius	BZ#2030173, BZ#1958979
galera	BZ#2042306
gcc	BZ#1996862
gdb	BZ#2012818, BZ#1853140
glibc	BZ#1934162, BZ#2007327, BZ#2023420, BZ#1929928, BZ#2000374
gnome-shell-extensions	BZ#1751336, BZ#1717947
gnome-software	BZ#1668760
gnutls	BZ#1628553
golang	BZ#2014088

Component	Tickets
grafana-pcp	BZ#1993149
grafana	BZ#1993214
grub2	BZ#1583445
hostapd	BZ#2016946
initscripts	BZ#1875485
ipa	BZ#1731484, BZ#1924707, BZ#1664719, BZ#1664718
js-d3-flame-graph	BZ#1993194
kdump-anaconda-addon	BZ#2086100
kernel	BZ#1953926, BZ#2068429, BZ#1910885, BZ#2040171, BZ#2022903, BZ#2036863, BZ#1979382, BZ#1949614, BZ#1983635, BZ#1964761, BZ#2069047, 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#1971506, BZ#2022359, BZ#2059262, 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#2004000
krb5	BZ#1877991
libcap	BZ#1950187, BZ#2032813
libffi	BZ#1875340
libgnome-keyring	BZ#1607766
libguestfs	BZ#1554735
libreswan	BZ#2017352, BZ#1989050
libseccomp	BZ#2019893

Component	Tickets
libselinux-python-2.8-module	BZ#1666328
libssh	BZ#1896651
libvirt	BZ#2014369, BZ#1664592, BZ#1332758, BZ#1528684
llvm-toolset	BZ#2001133
log4j-2-module	BZ#1937468
lsvpd	BZ#1993557
lvm2	BZ#1496229, BZ#1768536
make	BZ#2004246
mariadb	BZ#1944653, BZ#1942330
mesa	BZ#1886147
net-snmp	BZ#1908331
nfs-utils	BZ#1592011
nftables	BZ#2047821
nginx-1.20-module	BZ#1991787
nispor	BZ#1848817
nmstate	BZ#2003976, BZ#2004006
nss_nis	BZ#1803161
nss	BZ#1817533, BZ#1645153
opencryptoki	BZ#1984993
opencv	BZ#2007780, BZ#1886310
openmpi	BZ#1866402
opensc	BZ#1947025

Component	Tickets
openscap	BZ#1970529, BZ#2041781
openssh	BZ#1926103, BZ#2015828, BZ#2044354
openssl	BZ#1810911
osbuild-composer	BZ#1951936, BZ#2056451
oscap-anaconda-addon	BZ#1834716, BZ#2075508, BZ#1843932, BZ#1665082
pacemaker	BZ#1082146, BZ#1470834, BZ#1376538
pcp	BZ#1991763, BZ#1629455
pcs	BZ#1990784, BZ#1936833, BZ#1619620, BZ#1847102, BZ#1851335
pcsc-lite	BZ#1928154, BZ#2014641
perl	BZ#2021471
php	BZ#1978356
pki-core	BZ#1729215, BZ#1628987
pmdk-1_fileformat_v6-module	BZ#2009889
podman	JIRA:RHELPLAN-92741, JIRA:RHELPLAN-108830, JIRA:RHELPLAN-77238
policycoreutils	BZ#1731501
postfix	BZ#1711885
powerpc-utils	BZ#2028690, BZ#2022225
pykickstart	BZ#1637872
qemu-kvm	BZ#1982993, BZ#2004416, BZ#1662007, BZ#2020133, BZ#2012373, BZ#1740002, BZ#1719687, BZ#1651994
rear	BZ#2048454, BZ#2049091, BZ#2035939, BZ#1868421, BZ#2083301
redhat-support-tool	BZ#2018194, BZ#2018195, BZ#1767195, BZ#2064575, BZ#1802026

Component	Tickets
restore	BZ#1997366
rhel-system-roles	BZ#1967321, BZ#2040038, BZ#2041627, BZ#2034908, BZ#1979714, BZ#2005727, BZ#2006231, BZ#2021678, BZ#2021683, BZ#2047504, BZ#2040812, BZ#2064388, BZ#2058655, BZ#2058772, BZ#2029605, BZ#2057172, BZ#2049747, BZ#1854988, BZ#1893743, BZ#1993379, BZ#1993311, BZ#2021661, BZ#2016514, BZ#1985022, BZ#2016511, BZ#2010327, BZ#2012316, BZ#2031521, BZ#2054364, BZ#2054363, BZ#2008931, BZ#1695634, BZ#1897565, BZ#2054365, BZ#1932678, BZ#2057656, BZ#2022458, BZ#2057645, BZ#2057661, BZ#2021685, BZ#2006081
rig	BZ#1888705
rpm-ostree	BZ#2032594
rpm	BZ#1940895, BZ#1688849
rsyslog	BZ#1947907, BZ#1679512, JIRA:RHELPLAN-10431
rteval	BZ#2012285
rust-toolset	BZ#2002883
samba	BZ#2013596, BZ#2009213, JIRA:RHELPLAN-13195, Jira:RHELDOCS-16612
scap-security-guide	BZ#1983061, BZ#2053587, BZ#2023569, BZ#1990736, BZ#2002850, BZ#2000264, BZ#2058033, BZ#2030966, BZ#1884687, BZ#1993826, BZ#1956972, BZ#2014485, BZ#2021802, BZ#2028428, BZ#1858866, BZ#1750755, BZ#2038977
scap-workbench	BZ#2051890
selinux-policy	BZ#1860443, BZ#1461914
sos	BZ#1873185, BZ#2011413
spice	BZ#1849563
sssd	BZ#2015070, BZ#1947671
strace	BZ#2038992
subscription-manager	BZ#2000883, BZ#2049441

Component	Tickets
texinfo	BZ#2022201
udica	BZ#1763210
usbguard	BZ#2000000, BZ#1963271
vdo	BZ#1949163
virt-manager	BZ#1995125, BZ#2026985
wayland	BZ#1673073
xfsdump	BZ#2020494
xorg-x11-server	BZ#1698565
other	BZ#1839151, BZ#1780124, BZ#2089409, JIRA:RHELPLAN-100359, JIRA:RHELPLAN-103147, JIRA:RHELPLAN-103146, JIRA:RHELPLAN-79161, BZ#2046325, JIRA:RHELPLAN-108438, JIRA:RHELPLAN-100175, BZ#2083036, JIRA:RHELPLAN-102505, BZ#2062117, JIRA:RHELPLAN-75169, JIRA:RHELPLAN-100174, JIRA:RHELPLAN-101137, JIRA:RHELPLAN-57941, JIRA:RHELPLAN-101133, JIRA:RHELPLAN-101138, JIRA:RHELPLAN-95126, JIRA:RHELPLAN-103855, JIRA:RHELPLAN-103579, BZ#2025814, BZ#2077770, BZ#1777138, BZ#1640697, BZ#1697896, BZ#1971061, BZ#1959020, 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#2020301, BZ#2028361, BZ#2041997, BZ#2035158, JIRA:RHELPLAN-109067, JIRA:RHELPLAN-115603, BZ#1690207, JIRA:RHELPLAN-1212, BZ#1559616, BZ#1889737, JIRA:RHELPLAN-14047, BZ#1769727, JIRA:RHELPLAN-27394, JIRA:RHELPLAN-27737, BZ#1906489, JIRA:RHELPLAN-100039, 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

APPENDIX B. REVISION HISTORY

0.2-9

Thu February 29 2024, Lucie Vařáková (lvarakova@redhat.com)

- Added a deprecated functionality JIRA:RHELDOCS-17641 (Networking).

0.2-8

Tue February 13 2024, Lucie Vařáková (lvarakova@redhat.com)

- Added a deprecated functionality JIRA:RHELDOCS-17573 (Identity Management).

0.2-7

Fri November 10 2023, Gabriela Fialová (gfialova@redhat.com)

- Updated the module on Providing Feedback on RHEL Documentation.

0.2-6

Fri October 13 2023, Gabriela Fialová (gfialova@redhat.com)

- Added a Tech Preview JIRA:RHELDOCS-16861 (Containers).

0.2-5

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.

0.2-4

Tue September 05 2023, Jaroslav Klech (jklech@redhat.com)

- Fixed an ordered list for known issue BZ#2169382 (Networking).

0.2-3

Thu August 24 2023, Lucie Vařáková (lvarakova@redhat.com)

- Added a known issue BZ#2214508 (Kernel).

0.2-2

Fri August 4 2023, Lenka Špačková (lspackova@redhat.com)

- Fixed section for BZ#2225332.

0.2-1

Tue August 1 2023, Lenka Špačková (lspackova@redhat.com)

- Added deprecated functionality BZ#2225332.
- Improved abstract.

0.2-0

Tue Aug 01 2023, Lucie Vařáková (lvarakova@redhat.com)

- Added deprecated functionality JIRA:RHELPLAN-147538 (The web console).

0.1-9

Thu Jun 29 2023, Marc Muehlfeld (mmuehlfeld@redhat.com)

- Added a Technology Preview BZ#1570255 (Kernel).

0.1-8

Fri Jun 16 2023, Lucie Vařáková (lvarakova@redhat.com)

- Added a known issue BZ#2214235 (Kernel).

0.1-7

Wed May 10 2023, Jaroslav Klech (jklech@redhat.com)

- Added a known issue BZ#2169382 (Networking).

0.1-6

Thu Apr 27 2023, Gabriela Fialová (gfialova@redhat.com)

- Added a known issue JIRA:RHELPLAN-155168 (Identity Management).

0.1-5

Thu Apr 13 2023, Gabriela Fialová (gfialova@redhat.com)

- Fixed 2 broken links in DFs and KIs.

0.1-4

Thu Mar 2 2023, Lucie Vařáková (lvarakova@redhat.com)

- Updated a new feature BZ#2089409 (Kernel).

0.1-4

Tue Jan 24 2023, Lucie Vařáková (lvarakova@redhat.com)

- Added a known issue BZ#2115791 (RHEL in cloud environments).

0.1-3

Thu Dec 08 2022, Marc Muehlfeld (mmuehlfeld@redhat.com)

- Added a known issue BZ#2132754 (Networking).

0.1-2

Tue Nov 08 2022, Lucie Vařáková (lvarakova@redhat.com)

- Added new features JIRA:RHELPLAN-137623 and BZ#2130912 (Containers).
- Added a Technology Preview JIRA:RHELPLAN-137622 (Containers).
- Added a known issue BZ#2134184 (Virtualization).

0.1-1

Wed Sep 07 2022, Lucie Vařáková (lvarakova@redhat.com)

- Added bug fix BZ#2096256 (Networking).
- Other minor updates.

0.1-0

Fri Aug 19 2022, Lucie Vařáková (lvarakova@redhat.com)

- Added bug fix BZ#2108316 (Identity Management).

0.0-9

Fri Aug 05 2022, Lucie Vařáková (lvarakova@redhat.com)

- Added known issue BZ#2114981 (Security).

0.0-8

Wed Aug 03 2022, Lenka Špačková (lspackova@redhat.com)

- Added known issue BZ#2095764 (Software management).

0.0-7

Fri Jul 22 2022, Lucie Vařáková (lvarakova@redhat.com)

- Added bug fix BZ#2020494 (File systems and storage).
- Added known issue BZ#2054656 (Virtualization).
- Other minor updates.

0.0-6

Mon Jul 11 2022, Lenka Špačková (lspackova@redhat.com)

- Added bug fix BZ#2056451 (Installer and image creation).
- Added bug fix BZ#2051890 (Security).
- Other minor updates.

0.0-5

Jun 08 2022, Lucie Vařáková (lmanasko@redhat.com)

- Added new feature BZ#2089409 (Kernel).

0.0-4

May 31 2022, Lucie Vařáková (lmanasko@redhat.com)

- Added known issues BZ#2075508 (Security) and BZ#2077770 (Virtualization).
- Added Technology Previews BZ#1989930 (RHEL for Edge) and JIRA:RHELPLAN-108438 (The web console).

- Added information about the in-place upgrade from RHEL 8 to RHEL 9 to the In-place upgrade and OS conversion section.
- Other minor updates.

0.0-3

May 18 2022, Lucie Maňásková (lmanasko@redhat.com)

- Added new feature BZ#2049441 (The web console).
- Added known issues BZ#2086100 (Kernel) and BZ#2020133 (Virtualization).
- Other small updates.

0.0-2

May 16 2022, Lucie Maňásková (lmanasko@redhat.com)

- Added bug fix BZ#2014369 (Virtualization).
- Added known issue BZ#1554735 (Virtualization).
- Other small updates.

0.0-1

May 11 2022, Lucie Maňásková (lmanasko@redhat.com)

- Release of the Red Hat Enterprise Linux 8.6 Release Notes.

0.0-0

Mar 30 2022, Lucie Maňásková (lmanasko@redhat.com)

- Release of the Red Hat Enterprise Linux 8.6 Beta Release Notes.