Red Hat Enterprise Linux 9

9.0 Release Notes

Release Notes for Red Hat Enterprise Linux 9.0

Last Updated: 2024-06-26

Red Hat Enterprise Linux 9 9.0 Release Notes

Release Notes for Red Hat Enterprise Linux 9.0

Legal Notice

Abstract

The Release Notes provide high-level coverage of the improvements and additions that have been implemented in Red Hat Enterprise Linux 9.0 and document known problems in this release, as well as notable bug fixes, Technology Previews, deprecated functionality, and other details. For information on how to install Red Hat Enterprise Linux, proceed to the Installation section.

Table of Contents

PROVIDING FEEDBACK ON RED HAT DOCUMENTATION

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

Submitting feedback through Jira (account required)

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

CHAPTER 1. OVERVIEW

1.1. MAJOR CHANGES IN RHEL 9.0

Security

The usage of the **SHA-1** message digest for cryptographic purposes has been deprecated in RHEL 9. The digest produced by SHA-1 is not considered secure because of many documented successful attacks based on finding hash collisions. The RHEL core crypto components no longer create signatures using SHA-1 by default. Applications in RHEL 9 have been updated to avoid using SHA-1 in security-relevant use cases.

Among the exceptions, the HMAC-SHA1 message authentication code and the Universal Unique Identifier (UUID) values can still be created using SHA-1 because these use cases do not currently pose security risks. SHA-1 can also be used in limited cases connected with important interoperability and compatibility concerns, such as Kerberos and WPA-2. See the List of RHEL applications using cryptography that is not compliant with FIPS 140-3 section for more details.

For solutions of compatibility problems with systems that still require SHA-1, see the following KCS articles:

- SSH from RHEL 9 to RHEL 6 systems does not work
- Packages signed with SHA-1 cannot be installed or upgraded
- Failed connection with SSH servers and clients that do not support the 'server-sig-algs' extension

OpenSSL is now provided in version 3.0.1, which adds a provider concept, a new versioning scheme, an improved HTTP(S) client, support for new protocols, formats, and algorithms, and many other improvements.

The system-wide **cryptographic policies** have been adjusted to provide up-to-date secure defaults.

OpenSSH is distributed in version 8.7p1, which provides many enhancements, bug fixes, and security improvements as compared to version 8.0p1, which is distributed in RHEL 8.5.

The SFTP protocol replaces the previously used SCP/RCP protocol in **OpenSSH**. SFTP offers more predictable filename handling and does not require expansion of **glob(3)** patterns by the shell on the remote side.

SELinux performance has been substantially improved, including time to load SELinux policy into the kernel, memory overhead, and other parameters. For additional information, see the Improving the performance and space efficiency of SELinux blog post.

RHEL 9 provides the **fapolicyd** framework in 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.

The SCAP Security Guide (SSG) packages are provided in version 0.1.60, which introduces delta tailoring, updated security profiles, and other improvements.

See Section 4.7, “Security” for more information.

The use of SHA-1 for signatures is restricted in the DEFAULT crypto policy. Except for HMAC, SHA-1 is no longer allowed in TLS, DTLS, SSH, IKEv2, DNSSEC, and Kerberos protocols.

If your scenario requires the use of SHA-1 for verifying existing or third-party cryptographic signatures, you can enable it by entering the following command:

```
# update-crypto-policies --set DEFAULT:SHA1
```

Alternatively, you can switch the system-wide crypto policies to the **LEGACY** policy. Note that **LEGACY** also enables many other algorithms that are not secure.

Cyrus SASL now uses GDBM instead of Berkeley DB, and the Network Security Services (NSS) libraries no longer support the DBM file format for the trust database.

Support for disabling SELinux through the **SELINUX=disabled** option in the **/etc/selinux/config** file has been removed from the kernel. When you disable SELinux only through **/etc/selinux/config**, the system starts with SELinux enabled but with no policy loaded. If your scenario requires disabling SELinux, add the **selinux=0** parameter to your kernel command line.

See the Security section in the *Considerations in adopting RHEL 9* document for more information about security-related major differences between RHEL 9 and RHEL 8.

Networking

You can use the new MultiPath TCP daemon (mptcpd) to configure MultiPath TCP (MPTCP) endpoints without using the **iproute2** utility. To make MPTCP subflows and endpoints persistent, use a NetworkManager dispatcher script.

By default, NetworkManager now uses the key files to store new connection profiles. Note that the **ifcfg** format is still supported.

For more information about the features introduced in this release and changes in the existing functionality, see New features - Networking.

The WireGuard VPN technology is now available as an unsupported Technology Preview. For details, see Technology Previews - Networking.

The **teamd** service and the **libteam** library are deprecated. As a replacement, configure a bond instead of a network team.

The **iptables-nft** and **ipset** are deprecated. These packages include utilities, such as **iptables**, **ip6tables**, **ebtables** and **arptables**. Use the **nftables** framework to configure firewall rules.

For more information about deprecated functionality, see Deprecated functionality - Networking.

The **network-scripts** package has been removed. Use NetworkManager to configure network connections. For more information about functionality that is no longer part of RHEL, see the Networking section in the *Considerations in adopting RHEL 9* document.

Dynamic programming languages, web and database servers

RHEL 9.0 provides the following dynamic programming languages:

- **Node.js 16**
- **Perl 5.32**
- **PHP 8.0**
- **Python 3.9**
- **Ruby 3.0**

RHEL 9.0 includes the following version control systems:

- **Git 2.31**
- **Subversion 1.14**

The following web servers are distributed with RHEL 9.0:

- **Apache HTTP Server 2.4.51**
- **nginx 1.20**

The following proxy caching servers are available:

- **Varnish Cache 6.6**
- **Squid 5.2**

RHEL 9.0 offers the following database servers:

- **MariaDB 10.5**
- **MySQL 8.0**
- **PostgreSQL 13**
- **Redis 6.2**

See Section 4.13, “Dynamic programming languages, web and database servers” for more information.

Compilers and development tools

System toolchain

The following system toolchain components are available with RHEL 9.0:

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

RHEL 9 system toolchain components include support for POWER10.

Performance tools and debuggers

The following performance tools and debuggers are available with RHEL 9.0:

- **GDB 10.2**
- **Valgrind 3.18.1**
- **SystemTap 4.6**
- **Dyninst 11.0.0**
- **elfutils 0.186**

Performance monitoring tools

The following performance monitoring tools are available with RHEL 9.0:

- **PCP 5.3.5**
- **Grafana 7.5.11**

Compiler toolsets
The following compiler toolsets are available with RHEL 9.0:

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

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

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

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

For more information, see OpenJDK documentation.

Java tools
The following Java tools are available with RHEL 9.0:

- **Maven 3.6**
- **Ant 1.10**

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

Desktop
The GNOME environment has been updated from GNOME 3.28 to GNOME 40 with many new features.

The **X.org** display server is deprecated, and will be removed in a future major RHEL release. The default desktop session is now the **Wayland** session in most cases.

When using the NVIDIA drivers, the desktop session now selects the Wayland display protocol by default, if the driver configuration supports Wayland. In previous RHEL releases, the NVIDIA drivers always disabled Wayland.

The **PipeWire** service now manages all audio output and input. **PipeWire** replaces the **PulseAudio** service in general use cases and the **JACK** service in professional use cases.

See Section 4.16, "Desktop" for more information.

Virtualization
In RHEL 9, the **libvirt** library uses modular daemons that handle individual virtualization driver sets on your host. This makes it possible to fine-grain a variety of tasks that involve virtualization drivers, such as resource load optimization and monitoring.

The QEMU emulator is now built using the Clang compiler. This enables the RHEL 9 KVM hypervisor to use a number of advanced security and debugging features. One of these features is SafeStack, which makes virtual machines (VMs) hosted on RHEL 9 significantly more secure against attacks based on Return-Oriented Programming (ROP).

In addition, Virtual Trusted Platform Module (vTPM) is now fully supported. Using vTPM, you can add a TPM virtual crypto-processor to a VM, which can then be used for generating, storing, and managing cryptographic keys.

Finally, the **virtiofs** feature has been implemented, which you can use to more efficiently share files between a RHEL 9 host and its VMs.

For more information about virtualization features introduced in this release, see Section 4.20, “Virtualization”.

1.2. IN-PLACE UPGRADE

In-place upgrade from RHEL 8 to RHEL 9

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

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 8 to RHEL 9 .

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

In-place upgrade from RHEL 7 to RHEL 9

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

1.3. RED HAT CUSTOMER PORTAL LABS

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

- Registration Assistant
- Kickstart Generator

- Red Hat Product Certificates
- Red Hat CVE Checker
- Kernel Oops Analyzer
- Red Hat Code Browser
- VNC Configurator
- Red Hat OpenShift Container Platform Update Graph
- Red Hat Satellite Upgrade Helper
- JVM Options Configuration Tool
- Load Balancer Configuration Tool
- Red Hat OpenShift Data Foundation Supportability and Interoperability Checker
- Ansible Automation Platform Upgrade Assistant
- Ceph Placement Groups (PGs) per Pool Calculator

1.4. ADDITIONAL RESOURCES

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

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

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

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

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

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

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

CHAPTER 2. ARCHITECTURES

Red Hat Enterprise Linux 9.0 is distributed with the kernel version 5.14.0, which provides support for the following architectures at the minimum required version:

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

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

CHAPTER 3. DISTRIBUTION OF CONTENT IN RHEL 9

3.1. INSTALLATION

Red Hat Enterprise Linux 9 is installed using ISO images. Two types of ISO image are available for the AMD64, Intel 64-bit, 64-bit ARM, IBM Power Systems, and IBM Z architectures:

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

NOTE

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

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

See the Performing a standard RHEL 9 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 9 installation document.

3.2. REPOSITORIES

Red Hat Enterprise Linux 9 is distributed through two main repositories:

- BaseOS
- AppStream

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

Content in the BaseOS repository is intended to provide the core set of the underlying OS functionality that provides the foundation for all installations. This content is available in the RPM format and is subject to support terms similar to those in previous releases of RHEL. For more information, see the Scope of Coverage Details document.

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

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

For more information about RHEL 9 repositories and the packages they provide, see the Package manifest.

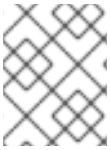
3.3. APPLICATION STREAMS

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

Application Streams are available in the familiar RPM format, as an extension to the RPM format called modules, as Software Collections, or as Flatpaks.

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

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



NOTE

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

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

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

Content that needs rapid updating, such as alternate compilers and container tools, is available in rolling streams that will not provide alternative versions in parallel. Rolling streams may be packaged as RPMs or modules.

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

3.4. PACKAGE MANAGEMENT WITH YUM/DNF

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

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

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

CHAPTER 4. NEW FEATURES

This part describes new features and major enhancements introduced in Red Hat Enterprise Linux 9.0.

4.1. INSTALLER AND IMAGE CREATION

Anaconda supports rhsm for machine provisioning through Kickstart installations for Satellite

Previously, machine provisioning depended on a custom **%post** script for Kickstart installation on Red Hat Satellite. This **%post** script imported the custom Satellite self-signed certificate, registered the machine, attached a subscription, and installed packages residing in repositories.

With RHEL 9, Satellite support has been added to the **rhsm** command for machine provisioning. You can now use **rhsm** for all provisioning tasks such as registering the system, attaching RHEL subscriptions, and installing from a Satellite instance.

(BZ#1951709)

RHEL supports localhost as a static hostname

Starting with RHEL 9, setting **localhost** as a static hostname in **/etc/hostname** is valid. In this case, NetworkManager does not try to obtain a transient hostname through DHCP or reverse DNS lookup.

(BZ#2190045)

Licensing, system, and user setting configuration screens have been disabled post standard installation

Previously, RHEL users were configuring Licensing, System (Subscription manager), and User Settings prior to the **gnome-initial-setup** and login screens. With this update, the initial setup screens have been disabled by default to improve user experience.

If you must run the initial setup for user creation or license display, install the following packages based on the requirements.

1. Install initial setup packages.

```
# dnf install initial-setup initial-setup-gui
```

2. Enable initial setup while next reboot of the system.

```
# systemctl enable initial-setup
```

3. Reboot the system to view initial setup.

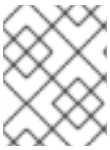
For Kickstart installations, add **initial-setup-gui** to the packages section and enable the **initial-setup** service.

```
firstboot --enable
%packages
@^graphical-server-environment
initial-setup-gui
%end
```

(BZ#1878583)

Anaconda activates network automatically for interactive installations

Previously, when performing an interactive installation without having the network activated by Kickstart or boot options, users had to activate the network manually in the network spoke. With this update, Anaconda activates the network automatically, without requiring users to visit the network spoke and activate it manually.



NOTE

This update does not change the installation experience for Kickstart installations and installations using the **ip=** boot option.

(BZ#1978264)

Image Builder now supports filesystem configuration

With this enhancement, you can specify custom filesystem configuration in your blueprints and you can create images with the desired disk layout. As a result, by having non-default layouts, you can benefit from security benchmarks, consistency with existing setups, performance, and protection against out-of-disk errors.

To customize the filesystem configuration in your blueprint, set the following customization:

```
[[customizations.filesystem]]
mountpoint = "MOUNTPOINT"
size = MINIMUM-PARTITION-SIZE
```

NOTE

After you add a file system customization to your blueprint, the file system is converted to a LVM partition.

(BZ#2011448)

New options to **Lock root account** and **Allow root SSH login with password**

The following new options have been added on the root password configuration screen in the RHEL graphical installation:

- Lock root account: Use this option to lock the root access to the machine.
- Allow root SSH login with password: Use this option to enable password-based SSH root logins.

To enable **password-based SSH root logins**, add the following line to the Kickstart file before you start the installation process.

```
%post
echo "PermitRootLogin yes" > /etc/ssh/sshd_config.d/01-permitrootlogin.conf
%end
```

(BZ#1940653)

Image Builder now supports creating bootable installer images

With this enhancement, you can use Image Builder to create bootable ISO images that consist of a **tarball** file, which contains a root file system. As a result, you can use the bootable ISO image to install the **tarball** file system to a bare metal system.

(BZ#2019318)

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 9 provides rpm-ostree v2022.2

RHEL 9 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 DNF 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.

(BZ#1961324)

RHEL 9 provides OSTree v2021.2

RHEL 9 is distributed with the **OSTree** package version v2021.2, which provides multiple bug fixes and enhancements. Notable changes include:

- New APIs for writing files, used in the new ostree-rs-ext project, to improve imports from tarballs.
- The **rofiles-fuse** command now handles **xattrs** extended attributes. Note: The **rofiles-fuse** is considered deprecated, see #2281.

- Improvements to the **introspection** API and testing.

(BZ#1961254)

The rpm-ostree rebase tool supports upgrade from RHEL 8 to RHEL 9

With this enhancement, you can upgrade your RHEL 8 system to RHEL 9 using the **rpm-ostree rebase** tool. It fully supports the default package set of RHEL for Edge upgrades between the most recent updates of RHEL 8 to the most recent updates of RHEL 9.

(BZ#2082306)

4.3. SUBSCRIPTION MANAGEMENT

Merged system purpose commands under subscription-manager syspurpose

Previously, there were two different commands to set system purpose attributes; **syspurpose** and **subscription-manager**. To unify all the system purpose attributes under one module, all the **addons**, **role**, **service-level**, and **usage** commands from subscription-manager have been moved to the new submodule, **subscription-manager syspurpose**.

Existing **subscription-manager** commands outside the new submodule are deprecated. The separate package (**python3-syspurpose**) that provides the **syspurpose** command line tool has been removed in RHEL 9.

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#1898563)

4.4. SOFTWARE MANAGEMENT

RHEL 9 provides RPM 4.16

RHEL 9 is distributed with RPM version 4.16. Notable bug fixes and enhancements over version 4.14 include:

- New SPEC features, most notably:
  - Fast macro-based dependency generators
  - The **%generate_buildrequires** section that allows for generating dynamic build dependencies
  - Meta (unordered) dependencies
  - Increased parallelism in package builds
  - Native version comparison in expressions
  - Caret version operator, opposite of tilde

    - **%elif**, **%elifos** and **%elifarch** statements
    - Optional automatic patch and source numbering
    - **%autopatch** now accepts patch ranges
    - **%patchlist** and **%sourcelist** sections
    - Enforced UTF-8 validation of header data at build-time
- The rpm database is now based on the **sqlite** library. Read-only support for **BerkeleyDB** databases has been retained for migration and query purposes.
- A new **rpm-plugin-audit** plug-in for issuing audit log events on transactions, previously built into RPM itself

(JIRA:RHELPLAN-80734)

New RPM plugin notifies fapolicyd about changes during RPM transactions

This update of the **rpm** packages introduces a new RPM plugin that integrates the **fapolicyd** framework with the RPM database. The plugin notifies **fapolicyd** about installed and changed files during an RPM transaction. As a result, **fapolicyd** now supports integrity checking.

Note that the RPM plugin replaces the DNF plugin because its functionality is not limited to DNF transactions but covers also changes by RPM.

(BZ#1942549)

RPM now supports the EdDSA public key algorithm

With this enhancement, the **rpm** command supports signing keys using the EdDSA public key algorithm. As a result, signing keys generated using EdDSA can now be used for signing and verifying packages.

Note that, however signing keys using EdDSA are now supported, RSA continues to be the default public key algorithm in GnuPG.

(BZ#1962234)

RPM now supports the Zstandard (zstd) compression algorithm

With this enhancement, the default RPM compression algorithm has switched to Zstandard (**zstd**). As a result, users can benefit from faster package installations, which can be especially noticeable during large transactions.

(JIRA:RHELPLAN-117903)

New DNF options exclude_from_weak_autodetect and exclude_from_weak

With this enhancement, the default DNF behavior does not install unwanted weak dependencies. To modify this behavior, use the following new options:

- **exclude_from_weak_autodetect**
  If enabled, the **exclude_from_weak_autodetect** option autodetects unmet weak dependencies (Recommends: or Supplements:) of packages installed on your system. As a result, providers of these weak dependencies are not installed as weak dependencies, but, if pulled in, they are installed as regular dependencies. The default value is **true**.
- **exclude_from_weak**

If enabled, the **exclude_from_weak** option prevents installing packages as weak dependencies (Recommends: or Supplements:). You can specify packages either by a package name or a glob, and separate them by a comma. The default value is **[]**.

(BZ#2005305)

RHEL 9 provides libmodulemd 2.13.0

RHEL 9 is distributed with the **libmodulemd** package version 2.13.0. Notable bug fixes and enhancements over version 2.9.4 include:

- Added support for delisting demodularized packages from a module.
- Added support for validating **modulemd-packager-v3** documents with a new **--type** option of the **modulemd-validator** tool.
- Fortified parsing integers.
- Fixed various **modulemd-validator** issues.

(BZ#1984403)

4.5. SHELLS AND COMMAND-LINE TOOLS

Bracketed paste is now enabled in bash by default

The bash **readline** library version 8.1 is now available, which enables bracketed paste mode by default. When you paste text to your terminal, **bash** highlights the text, and you must press **enter** to execute the pasted command. Bracketed paste mode is the default setting to avoid accidentally executing malicious commands.

To disable the bracketed paste mode for a specific user, add the following line to **~/.inputrc**:

```
set enable-bracketed-paste off
```

To disable the bracketed paste mode for all users, add the following line to **/etc/inputrc**:

```
set enable-bracketed-paste off
```

When you disable the bracketed paste mode, commands are directly executed on paste, and you do not need to confirm them by pressing **enter**.

(BZ#2079078)

RHEL 9 includes powerpc-utils 1.3.9

RHEL 9 provides the **powerpc-utils** package version 1.3.9. Notable bug fixes and enhancements over version 1.3.8 include:

- Increased the log size to 1 MB in **drmgr**.
- Fixed the **HCIND** array size at the boot time.
- Implemented **autoconnect-slaves** on HNV connections in **hcnmgr**.
- Improved the HNV bond list connections in **hcnmgr**.

- Use **hexdump** from **util-linux** in **hcnmgr**.
- The **hcn-init.service** starts with the 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 greater than 5 paths.
- Added missing substring extraction of **devpart** in l2of_vd() of **ofpathname**.
- Introduced **lpamumascore**.
- Fixed the remove by **index operation** in **drmgr**.
- Moved the definition of **SYS_PATH** from **l2of_vs()** to **l2of_scsi()** in **ofpathname**.
- Added **-x** option to enhance the security in partstat.
- Fixed **nroff** warnings and errors in **lparstat** man page.
- Implemented NUMA-based LMB removal in **drmgr**.
- Fixed **ofpathname** race with **udev** rename in **hcnmgr**.
- Use **NetworkManager nmcli** to check bonding interface status in **hcnmgr**.
- Use **NetworkManager nmcli** to clean the bond interface at the boot time when HNV does not exist.

(BZ#1873868)

RHEL 9 is distributed with opal-prd 6.7.1

The **opal-prd** package version 6.7.1 provides the following notable bug fixes and enhancements over the previously available version 6.6.3:

- Fixed **xscom** error logging issues caused due to **xscom OPAL** call.
- Fixed possible deadlock with the **DEBUG** build.
- Fallback to **full_reboot** if **fast-reboot** fails in **core/platform**.
- Fixed **next_ungarded_primary** in **core/cpu**.
- Improved rate limit timer requests and the timer state in Self-Boot Engine (SBE).

(BZ#1869560)

RHEL 9 provides lsvpd 1.7.12

RHEL 9 is distributed with the **lsvpd** package version 1.7.12. Notable bug fixes and enhancements over version 1.7.11 include:

- Added the UUID property in **sysvpd**.
- Improved the **NVMe** firmware version.

- Fixed PCI device manufacturer parsing logic.
- Added **recommends clause** to the **lsvpd** configuration file.

(BZ#1869564)

ppc64-diag version 2.7.7 available

The **ppc64-diag** package version 2.7.7 is provided in RHEL 9. Notable bug fixes and enhancements over version 2.7.6 include:

- Improved unit test cases.
- Added the UUID property in **sysvpd**.
- **rtas_errd** service does not run in the Linux containers.
- The obsolete logging options are no longer available in the **systemd** service files.

(BZ#1869567)

RHEL 9 includes Fetchmail 6.4.24

RHEL 9 is distributed with the **fetchmail** package version 6.4.24. **Fetchmail** is a remote-mail retrieval and forwarding utility.

For more information, see:

- the **/usr/share/doc/fetchmail/NEWS** file,
- the **fetchmail(1)** man page,
- the **/usr/share/doc/fetchmail/README.SSL** file for SSL-related information in case you need to change configuration.

(BZ#1999276)

RHEL 9 includes Eigen 3.4

RHEL 9 is distributed with the **eigen3** package version 3.4. **Eigen 3.4** is a C++ template library for linear algebra, which now supports POWER10 matrix multiplication assist instructions.

As a result, users of **Eigen 3.4** can perform optimized linear algebra computation on POWER10 systems.

(BZ#2032423)

RHEL 9 introduces the cdrskin package

RHEL 9 introduces the **cdrskin** package for burning data on CD, DVD, or BD media. The **cdrskin** package provides a replacement for the **cdrecord** executable from the **wodim** package, which is not available in RHEL 9.

The **cdrskin** package includes:

- Blanking, formatting, and burning of data on optical media.
- Multi session on CD.
- Emulated ISO-9660 multi-session on overwriteable DVD+RW, DVD-RW, DVD-RAM, BD-RE.

The **cdrskin** package also provides **cdrecord** command as a symbolic link to **cdrskin** binary, so you do not have to make any changes in user scripts. See **cdrskin(1)** manual page for the full set of features.

(BZ#2015861)

The redhat.rhel_mgmt Ansible collection is supported in the RHEL 9 release

This update provides support to the Intelligent Platform Management Interface (**IPMI**) Ansible modules. **IPMI** is a specification for a set of management interfaces to communicate with baseboard management controller (BMC) devices. The **IPMI** modules - **ipmi_power** and **ipmi_boot** - are available in the **redhat.rhel_mgmt** Collection, which you can access by installing the **ansible-collection-redhat-rhel_mgmt** package.

(BZ#2023381)

RHEL 9 introduces the util-linux-core package

In addition to the **util-linux** package, RHEL 9 provides the **util-linux-core** subpackage for scenarios where the size of installed packages is a critical feature, for example buildroots, certain containers, and boot images.

The **util-linux-core** subpackage contains a limited subset of the **util-linux** utilities, which are necessary to boot the Linux system, for example the **mount** utility.

The **util-linux-core** subpackage does not contain any external dependencies. For example, login utilities are not available due to the dependence on a PAM library.

For standard use cases, like installations, use the standard **util-linux** package. The **util-linux** package depends on **util-linux-core**, which means that if you install **util-linux**, **util-linux-core** is installed automatically.

(BZ#2079313)

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

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

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

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

(BZ#2136937)

4.6. INFRASTRUCTURE SERVICES

s-nail replaces mailx

The **s-nail** mail processing system has replaced the **mailx** utility. The **s-nail** utility is compatible with **mailx** and adds numerous new features. The **mailx** package is no longer maintained in the upstream.

(BZ#1940863)

TuneD 2.18 is available

RHEL 9 is distributed with TuneD version 2.18. Notable changes over version 2.16 include:

- The **net** plugin: added support for **txqueuelen** tuning.
- The **disk** plugin: added support for NVMe disk tuning.
- **tuned-gui** bug fixes.

(BZ#2003838)

RHEL 9 provides mod_security_crs 3.3

RHEL 9 is distributed with the **mod_security_crs** package version 3.3. Notable bug fixes and enhancements include:

- Introduced **libinjection**.
- Blocked backup files ending with ~ in filenames.
- Added new **LDAP** injection and **HTTP** splitting rules.
- Added **.swp** to restricted extensions.
- Added Common Attack Pattern Enumeration and Classification (CAPEC) tags for attack classification.
- Added support to detect **Nuclei** , **WFuzz**, and **ffuf** vulnerability scanners.
- Improved variable to lowercase (**modsec3 behavior fix**)
- Added support to detect Unix RCE bypass techniques through uninitialized variables, string concatenations, and globbing patterns.
- Removed outdated rule tags: **WASCTC**, **OWASP_TOP_10**, **OWASP_AppSensor/RE1**, and **OWASP_CRS/FOO/BAR**. **OWASP_CRS** and **attack-type** are still included in the **mod_security_crs** package.
- The format of **crs-setup.conf** variable **tx.allowed_request_content_type** has been changed to be in line with the other variables. In case the variable is overridden, please see the example in **crs-setup.conf** file for the new separator.

(BZ#1947962)

RHEL 9 provides chrony 4.1

RHEL 9 is distributed with **chrony** version 4.1. Notable bug fixes and enhancements over version 3.5 include:

- Support for Network Time Security (NTS) authentication has been added. For more information, see Overview of Network Time Security (NTS) in chrony .
- By default, the Authenticated Network Time Protocol (NTP) sources are trusted over non-authenticated NTP sources. To restore the original behavior, add the **autselectmode ignore** argument in the **chrony.conf** file.
- Support for authentication with **RIPEMD** keys - **RMD128**, **RMD160**, **RMD256**, **RMD320** - is no longer available.
- Support for long non-standard MACs in NTPv4 packets is no longer available. If you are using **chrony 2.x**, **non-MD5/SHA1** keys, you need to configure **chrony** with the **version 3** option.

In addition, the following differs from the RHEL 8 version of **chrony**:

- The **seccomp** filter is enabled by default (**-F 2** is set in **/etc/sysconfig/chronyd**). The **seccomp** filter conflicts with the **mailonchange** directive. If you have the **mailonchange** directive in **/etc/chrony.conf**, remove the **-F 2** setting from **/etc/sysconfig/chronyd**.

(BZ#1961131)

4.7. SECURITY

System-wide crypto-policies are now more secure

With this update, the system-wide cryptographic policies have been adjusted to provide up-to-date secure defaults:

- Disabled TLS 1.0, TLS 1.1, DTLS 1.0, RC4, Camellia, DSA, 3DES, and FFDHE-1024 in all policies.
- Increased minimum RSA key size and minimum Diffie-Hellman parameter size in LEGACY.
- Disabled TLS and SSH algorithms using SHA-1, with an exception of SHA-1 usage in Hash-based Message Authentication Codes (HMACs).

If your scenario requires enabling some of the disabled algorithms and ciphers, use custom policies or subpolicies.

(BZ#1937651)

RHEL 9 provides OpenSSL 3.0.1

RHEL 9 provides **openssl** packages in upstream version 3.0.1, which includes many improvements and bug fixes over the previous version. The most notable changes include:

- Added the new Provider concept. Providers are collections of algorithms, and you can choose different providers for different applications.
- Introduced the new versioning scheme in the following format: *<major>.<minor>.<patch>*.
- Added support for the Certificate Management Protocol (CMP, RFC 4210), the Certificate Request Message Format (CRMF), and HTTP transfer (RFC 6712).
- Introduced an HTTP(S) client that supports GET and POST, redirection, plain and ASN.1-encoded contents, proxies, and timeouts.
- Added new Key Derivation Function API (EVP_KDF) and Message Authentication Code API (EVP_MAC).
- Added support for Linux Kernel TLS (KTLS) through compiling with the **enable-ktls** configuration option.
- Added CAdES-BES signature verification support.
- Added CAdES-BES signature scheme and attributes support (RFC 5126) to CMS API.
- Added support for new algorithms, for example:
  - KDF algorithms "SINGLE STEP" and "SSH".
  - MAC algorithms "GMAC" and "KMAC".

- KEM algorithm "RSASVE".
- Cipher algorithm "AES-SIV"

- Added AuthEnvelopedData content type structure (RFC 5083) using AES_GCM.
- The default algorithms for PKCS #12 creation with the **PKCS12_create()** function changed to more modern PBKDF2 and AES-based algorithms.
- Added a new generic trace API.

(BZ#1990814)

OpenSSL now includes providers

The OpenSSL toolkit in version 3.0.1, which is included in RHEL 9, added the concept of providers. Providers are collections of algorithms, and you can choose different providers for different applications. OpenSSL currently includes the following providers: **base**, **default**, **fips**, **legacy**, and **null**.

By default, OpenSSL loads and activates the **default** provider, which includes commonly used algorithms such as RSA, DSA, DH, CAMELLIA, SHA-1, and SHA-2.

When the FIPS flag is set in the kernel, OpenSSL automatically loads the FIPS provider and uses only FIPS-approved algorithms. As a result, you do not have to manually switch OpenSSL to FIPS mode.

To change to a different provider on the system level, edit the **openssl.cnf** configuration file. For example, if your scenario requires using the **legacy** provider, uncomment the corresponding section.

WARNING

Explicitly activating a provider overrides the implicit activation of the default provider and may make the system remotely inaccessible, for example by the OpenSSH suite.

For information on the algorithms included in each provider, see the relevant man pages. For example, the **OSSL_PROVIDER-legacy(7)** man page for the **legacy** provider.

(BZ#2010291)

OpenSSL random bit generator now supports CPACF

This release of the **openssl** packages introduces support for the CP Assist for Cryptographic Functions (CPACF) in the OpenSSL NIST SP800-90A-compliant AES-based deterministic random bit generator (DRBG).

(BZ#1871147)

openssl-spkac can now create SPKAC files signed with SHA-1 and SHA-256

The **openssl-spkac** utility can now create Netscape signed public key and challenge (SPKAC) files signed with hashes different than MD5. You can now create and verify also SPKAC files signed with SHA-1 and SHA-256 hashes.

(BZ#1970388)

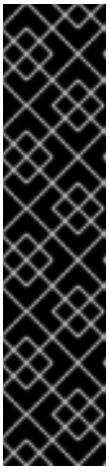
RHEL 9 provides openCryptoki 3.17.0

RHEL 9 is distributed with **openCryptoki** version 3.17.0. Notable bug fixes and enhancements over version 3.16.0 include:

- The **p11sak** utility adds a new function for listing keys.
- **openCryptoki** now supports:
  - OpenSSL 3.0.
  - Event notifications.
  - Software fallbacks in ICA tokens.
- The WebSphere Application Server no longer fails to start when the hardware crypto adapter is enabled.

RHEL 9 includes OpenSSL with additional patches, which are specific to RHEL. If the system is in Federal Information Processing Standards (FIPS) mode, OpenSSL automatically loads the FIPS provider and base provider and forces the applications to use the FIPS provider. Therefore, the behavior of **openCryptoki** on RHEL 9 differs from the upstream:

- Tokens that rely on OpenSSL's implementation of the crypto operations (soft tokens and ICA tokens software fallbacks) now support only FIPS-approved mechanisms, even though unapproved mechanisms are still listed as available.
- **openCryptoki** supports two different token data formats: the old data format, which uses non-FIPS-approved algorithms (such as DES and SHA1), and the new data format, which uses FIPS-approved algorithms only.
  The old data format no longer works because the FIPS provider allows the use of only FIPS-approved algorithms.



IMPORTANT

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

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

(BZ#1869533)

GnuTLS provided in version 3.7.3

In RHEL 9, the **gnutls** packages are provided in upstream version 3.7.3. This provides many improvements and bug fixes over previous versions, most notably:

- Introduced API for FIPS 140-3 explicit indicators.

- Hardened defaults for exporting PKCS#12 files.
- Fixed timing of the early data (zero round trip data, 0-RTT) exchange.
- The **certutil** tool no longer inherits the Certificate Revocation List (CRL) distribution point from the certificate authority (CA) when signing a certificate signing request (CSR).

(BZ#2033220)

RHEL 9 provides NSS 3.71

RHEL 9 is distributed with the Network Security Services (NSS) libraries version 3.71. Notable changes include:

- Support for the legacy DBM database format has been completely removed. NSS support only the SQLite database format in RHEL 9.
- The PKCS #12 encryption ciphers now use the AES-128-CBC with PBKDF2 and SHA-256 algorithms instead of PBE-SHA1-RC2-40 and PBE-SHA1-2DES.

(BZ#2008320)

NSS no longer support RSA keys shorter than 1023 bits

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

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

(BZ#2099438)

Minimal RSA key bit length option in OpenSSH

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

(BZ#2119694)

OpenSSH distributed in 8.7p1

RHEL 9 includes **OpenSSH** in version 8.7p1. This version provides many enhancements and bug fixes over **OpenSSH** version 8.0p1, which is distributed in RHEL 8.5, most notably:

New Features

- Support for transfers using the SFTP protocol as a replacement for the previously used SCP/RCP protocol. SFTP offers more predictable filename handling and does not require expansion of glob(3) patterns by the shell on the remote side.
  SFTP support is enabled by default. If SFTP is unavailable or incompatible in your scenario, you can use the **-O** flag to force use of the original SCP/RCP protocol.
- The **LogVerbose** configuration directive that allows forcing maximum debug logging by file/function/line pattern lists.

- Client address-based rate-limiting with the new **sshd_config PerSourceMaxStartups**, and **PerSourceNetBlockSize** directives. This provides finer control than the global **MaxStartups** limit.

- The **HostbasedAcceptedAlgorithms** keyword now filters based on the signature algorithm instead of filtering by key type.

- The **Include sshd_config** keyword in the **sshd** daemon that allows including additional configuration files by using **glob** patterns.

- Support for Universal 2nd Factor (U2F) hardware authenticators specified by the FIDO Alliance. U2F/FIDO are open standards for inexpensive two-factor authentication hardware that are widely used for website authentication. In **OpenSSH**, FIDO devices are supported by new public key types **ecdsa-sk** and **ed25519-sk** and by the corresponding certificate types.

- Support for FIDO keys that require a PIN for each use. You can generate these keys by using **ssh-keygen** with the new **verify-required** option. When a PIN-required key is used, the user will be prompted for a PIN to complete the signature operation.

- The **authorized_keys** file now supports a new **verify-required** option. This option requires FIDO signatures to assert token verification of the user's presence before making the signature. The FIDO protocol supports multiple methods for user verification, OpenSSH currently supports only PIN verification.

- Added support for verifying FIDO **webauthn** signatures. **webauthn** is a standard for using FIDO keys in web browsers. These signatures are a slightly different format to plain FIDO signatures and therefore require explicit support.

Bug fixes

- Clarified semantics of the **ClientAliveCountMax=0** keyword. Now, it entirely disables connection killing instead of the previous behavior of instantly killing the connection after the first liveness test regardless of its success.

Security

- Fixed an exploitable integer overflow bug in the private key parsing code for the XMSS key type. This key type is still experimental and support for it is not compiled by default. No user-facing autoconf option exists in portable OpenSSH to enable it.

- Added protection for private keys at rest in RAM against speculation and memory side-channel attacks like Spectre, Meltdown and Rambleed. This release encrypts private keys when they are not in use with a symmetric key that is derived from a relatively large "prekey" consisting of random data (currently 16 KB).

(BZ#1952957)

Locale forwarding disabled by default in OpenSSH

Using the **C.UTF-8** locale in small images, such as containers and virtual machines, reduces size and improves performance over using the traditional **en_US.UTF-8** locale.

Most distributions send locale environment variables by default and accept them on the server side. However, this meant that logging in through SSH from clients that used locales other than **C** or **C.UTF-8** to servers that did not have the **glibc-langpack-en** or **glibc-all-langpacks** package installed resulted in degraded user experience. Specifically, output in the UTF-8 format was broken and some tools did not work or sent frequent warning messages.

With this update, locale forwarding is switched off by default in OpenSSH. This keeps the locale viable even if clients connect to servers with minimal installations that support only a small set of locales.

(BZ#2002734)

OpenSSH supports U2F/FIDO security keys

Previously, the OpenSSH keys stored in hardware were only supported through the PKCS #11 standard, which limited the use of other security keys in SSH. Support for U2F/FIDO security keys was developed upstream and is now implemented in RHEL 9. This results in an improved usability of security keys within SSH independent of the PKCS #11 interface.

(BZ#1821501)

Libreswan provided in version 4.6

In RHEL 9, Libreswan is provided in upstream version 4.6. This version provides many bug fixes and enhancements, most notably improvements on labeled IPsec used with Internet Key Exchange version 2 (IKEv2).

(BZ#2017355)

Libreswan does not accept IKEv1 packages by default

Because the Internet Key Exchange v2 (IKEv2) protocol is now widely deployed, Libreswan no longer supports IKEv1 packets by default. IKEv2 provides a more secure environment and more resilience against attacks. If your scenario requires the use of IKEv1, you can enable it by adding the **ikev1-policy=accept** option to the **/etc/ipsec.conf** configuration file.

(BZ#2039877)

RHEL 9 provides stunnel 5.62

RHEL 9 is distributed with the **stunnel** package version 5.62. Notable bug fixes and enhancements include:

- On systems in FIPS mode, **stunnel** now always uses FIPS mode.
- The **NO_TLSv1.1**, **NO_TLSv1.2**, and **NO_TLSv1.3** options have been renamed to **NO_TLSv1_1**, **NO_TLSv1_2**, and **NO_TLSv1_3** respectively.
- The new service-level **sessionResume** option enables and disables session resumption.
- LDAP is now supported in **stunnel** clients using the **protocol** option.
- A Bash-completion script is now available.

(BZ#2039299)

RHEL 9 provides nettle 3.7.3

RHEL 9 provides the **nettle** package 3.7.3 version with multiple bug fixes and enhancements. Notable changes are the following:

- Supports new algorithms and modes, for example, **Ed448**, **SHAKE256**, **AES-XTS**, **SIV-CMAC**.
- Adds architecture-specific optimizations for existing algorithms.

(BZ#1986712)

RHEL 9 provides p11-kit 0.24

RHEL 9 provides **p11-kit** package with 0.24 version. This version provides multiple bug fixes and enhancements. Notably, the subdirectory for storing distrusted Certificate Authorities has been renamed to **blocklist**.

(BZ#1966680)

cyrus-sasl now uses GDBM instead of Berkeley DB

The **cyrus-sasl** package is now built without the **libdb** dependency, and the **sasldb** plugin uses the GDBM database format instead of Berkeley DB. To migrate your existing Simple Authentication and Security Layer (SASL) databases stored in the old Berkeley DB format, use the **cyrusbdb2current** tool with the following syntax:

```
cyrusbdb2current <sasldb_path> <new_path>
```

(BZ#1947971)

SELinux policy in RHEL 9 is up-to-date with the current kernel

The SELinux policy includes new permissions, classes, and capabilities that are also part of the kernel. Therefore, SELinux can utilize the full potential provided by the kernel. Specifically, SELinux has better granularity for granting permissions, which has subsequent security benefits. This also enables running systems with the MLS SELinux policy because the MLS policy would prevent some systems from starting if the system contained permissions unknown to the policy.

(BZ#1941810, BZ#1954145)

Default SELinux policy disallows commands with text relocation libraries

The **selinuxuser_execmod** boolean is now off by default to improve the security footprint of installed systems. As a result, SELinux users cannot enter commands using libraries that require text relocation, unless the library files have the **textrel_shlib_t** label.

(BZ#2055822)

OpenSCAP is provided in version 1.3.6

RHEL 9 includes OpenSCAP in version 1.3.6, which provides bug fixes and improvements, most notably:

- You can provide local copies of remote SCAP source data stream components instead of downloading them during the scan by using the **--local-files** option
- OpenSCAP accepts multiple **--rule** arguments to select multiple rules on the command line.
- You can skip 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#2041782)

OSCAP Anaconda Add-on now supports a new add-on name

With this enhancement, you can use the new **com_redhat_oscap** add-on name as opposed to the legacy **org_fedora_oscap** add-on name in the Kickstart file for the **OSCAP Anaconda Add-on** plugin. For example, the Kickstart section can be structured as follows:

```
%addon com_redhat_oscap
  content-type = scap-security-guide
%end
```

OSCAP Anaconda Add-on is currently compatibile with the legacy add-on name, but support for the legacy add-on name will be removed in a future major RHEL version.

(BZ#1893753)

CVE OVAL feeds now compressed

With this update, Red Hat provides CVE OVAL feeds in a compressed form. They are no longer available as XML files, but are in the **bzip2** format instead. The location of the feeds for RHEL9 has also been updated to reflect this change. Note that third-party SCAP scanners might have problems with scanning rules that use a compressed feed because referencing compressed content is not standardized.

(BZ#2028435)

SCAP Security Guide provided in version 0.1.60

RHEL 9 includes the **scap-security-guide** packages in version 0.1.60. This version provides bug fixes and enhancements, most notably:

- The rules hardening the PAM stack now use **authselect** as the configuration tool.
- SCAP Security Guide now provides a delta tailoring file for the STIG profile. This tailoring file defines a profile that represents the differences between DISA's automated STIG and SSG automated content.

(BZ#2014561)

SCAP Security Guide profiles supported in RHEL 9.0

With the SCAP Security Guide compliance profiles included in RHEL 9.0, you can harden the system to the recommendations from the issuing organizations. As a result, you can configure and automate compliance of your RHEL 9 systems according to your required hardening level by using the associated remediations and SCAP profiles.

Profile name	Profile ID	Policy version
French National Agency for the Security of Information Systems (ANSSI) BP-028 Enhanced Level	**xccdf_org.ssgproject.content_profile_anssi_bp28_enhanced**	1.2
French National Agency for the Security of Information Systems (ANSSI) BP-028 High Level	**xccdf_org.ssgproject.content_profile_anssi_bp28_high**	1.2

Profile name	Profile ID	Policy version
French National Agency for the Security of Information Systems (ANSSI) BP-028 Intermediary Level	**xccdf_org.ssgproject.content_profile_anssi_bp28_intermediary**	1.2
French National Agency for the Security of Information Systems (ANSSI) BP-028 Minimal Level	**xccdf_org.ssgproject.content_profile_anssi_bp28_minimal**	1.2
[DRAFT] CIS Red Hat Enterprise Linux 9 Benchmark for Level 2 - Server	**xccdf_org.ssgproject.content_profile_cis**	DRAFT[a]
[DRAFT] CIS Red Hat Enterprise Linux 9 Benchmark for Level 1 - Server	**xccdf_org.ssgproject.content_profile_cis_server_l1**	DRAFT[a]
[DRAFT] CIS Red Hat Enterprise Linux 9 Benchmark for Level 1 - Workstation	**xccdf_org.ssgproject.content_profile_cis_workstation_l1**	DRAFT[a]
[DRAFT] CIS Red Hat Enterprise Linux 9 Benchmark for Level 2 - Workstation	**xccdf_org.ssgproject.content_profile_cis_workstation_l2**	DRAFT[a]
[DRAFT] Unclassified Information in Non-federal Information Systems and Organizations (NIST 800-171)	**xccdf_org.ssgproject.content_profile_cui**	r2
Australian Cyber Security Centre (ACSC) Essential Eight	**xccdf_org.ssgproject.content_profile_e8**	not versioned
Health Insurance Portability and Accountability Act (HIPAA)	**xccdf_org.ssgproject.content_profile_hipaa**	not versioned
Australian Cyber Security Centre (ACSC) ISM Official	**xccdf_org.ssgproject.content_profile_ism_o**	not versioned
[DRAFT] Protection Profile for General Purpose Operating Systems	**xccdf_org.ssgproject.content_profile_ospp**	4.2.1
PCI-DSS v3.2.1 Control Baseline for Red Hat Enterprise Linux 9	**xccdf_org.ssgproject.content_profile_pci-dss**	3.2.1
[DRAFT] DISA STIG for Red Hat Enterprise Linux 9	**xccdf_org.ssgproject.content_profile_stig**	DRAFT[b]

Profile name	Profile ID	Policy version
[DRAFT] DISA STIG with GUI for Red Hat Enterprise Linux 9	**xccdf_org.ssgproject.content_profile_stig_gui**	DRAFT[b]

[a] CIS has not yet published an official benchmark for RHEL 9

[b] DISA has not yet published an official benchmark for RHEL 9

WARNING

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

(BZ#2045341, BZ#2045349, BZ#2045361, BZ#2045368, BZ#2045374, BZ#2045381, BZ#2045386, BZ#2045393, BZ#2045403)

RHEL 9 provides fapolicyd 1.1

RHEL 9 is distributed with the **fapolicyd** package version 1.1. Most notable enhancements 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#2032408)

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#2027971)

logrotate included in a separate rsyslog-logrotate package

The **logrotate** config was separated from the main **rsyslog** package into the new **rsyslog-logrotate** package. This is useful in certain minimal environments, for example where log rotation is not needed, to prevent installing unnecessary dependencies.

(BZ#1992155)

sudo supports Python plugins

With the **sudo** program version 1.9, which is included in RHEL 9, you can write **sudo** plugins in Python. This makes it easier to enhance **sudo** to more precisely suit specific scenarios.

For additional information, see the **sudo_plugin_python(8)** man page.

(BZ#1981278)

libseccomp provided in version 2.5.2

RHEL 9.0 provides the **libseccomp** packages in upstream version 2.5.2. This version provides many bug fixes and enhancements over previous versions, 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**.
- Fixed a bug where **seccomp_load()** could only be called once.
- Changed the notification **fd** handling to only request a notification **fd** if the filter has a **_NOTIFY** action.
- Added documentation about **SCMP_ACT_NOTIFY** to the **seccomp_add_rule(3)** manpage.
- Clarified the maintainers' GPG keys.

(BZ#2019887)

Clevis now supports SHA-256

With this enhancement, the Clevis framework supports the **SHA-256** algorithm as the default hash for JSON Web Key (JWK) thumbprints as recommended by **RFC 7638**. Because the older thumbprints (SHA-1) are still supported, you can still decrypt the previously encrypted data.

(BZ#1956760)

4.8. NETWORKING

The diag modules are now available in the kernel

The **diag** modules are now included with the kernel image. With this update, the **diag** modules no longer need to be dynamically loaded when the **ss** command is used. This allows better debugging of networking issues regardless of the customer policy on kernel modules. Modules included in the kernel:

```
CONFIG_INET_DIAG
CONFIG_INET_RAW_DIAG
CONFIG_INET_TCP_DIAG
CONFIG_INET_UDP_DIAG
CONFIG_INET_MPTCP_DIAG
CONFIG_NETLINK_DIAG
CONFIG_PACKET_DIAG
CONFIG_UNIX_DIAG
```

(BZ#1948340)

New core and IPv4-related networking sysctl kernel parameters

The RHEL 9.0 kernel provides the following new core and IPv4 networking **sysctl** parameters compared to previous RHEL versions:

- **net.core.devconf_inherit_init_net**
- **net.core.gro_normal_batch**
- **net.core.high_order_alloc_disable**
- **net.core.netdev_unregister_timeout_secs**
- **net.ipv4.fib_multipath_hash_fields**
- **net.ipv4.fib_notify_on_flag_change**
- **net.ipv4.fib_sync_mem**
- **net.ipv4.icmp_echo_enable_probe**
- **net.ipv4.ip_autobind_reuse**
- **net.ipv4.nexthop_compat_mode**
- **net.ipv4.raw_l3mdev_accept**
- **net.ipv4.tcp_comp_sack_slack_ns**
- **net.ipv4.tcp_migrate_req**
- **net.ipv4.tcp_mtu_probe_floor**
- **net.ipv4.tcp_no_ssthresh_metrics_save**
- **net.ipv4.tcp_reflect_tos**

For details about these parameters, install the **kernel-doc** package and see the following files:

- **/usr/share/doc/kernel-doc-<version>/Documentation/admin-guide/sysctl/net.rst**

- **/usr/share/doc/kernel-doc-<version>/Documentation/networking/ip-sysctl.rst**

(BZ#2068532)

Changed behavior in firewalld when transmitting packets between zones

In zone-based firewalls, packets enter only one zone. Implicit packet transmission is the concept violation and can allow traffic or services unexpectedly. In Red Hat Enterprise Linux 9 the **firewalld** service no longer allows implicit packet transmission between two different zones.

For more information about this change, see Changed behavior in **firewalld** when transmitting packets between zones Knowledge Article.

(BZ#2029211)

Intra-zone forwarding has been enabled by default

The **firewalld** intra-zone forwarding feature allows forwarding traffic between interfaces or sources within a firewalld zone. Starting with RHEL 9.0, this feature has been enabled by default. Use the **--add-forward** option of the **firewall-cmd** utility to enable intra-zone forwarding for a particular zone. The **firewall-cmd --list-all** command displays whether intra-zone forwarding is enabled or disabled for a zone:

```
# firewall-cmd --list-all
public (active)
  ...
  forward: no
```

(BZ#2089193)

Making Nmstate more inclusive

Red Hat is committed to using conscious language. Therefore the **slave** term in the **nmstate** API has been replaced by the term **port**.

(BZ#1969941)

NetworkManager supports interface names set in the rd.znet_ifname kernel option on IBM Z

With this enhancement, on the IBM Z platform, NetworkManager now interprets the **rd.znet** and **rd.znet_ifname** kernel command-line options when installing or booting Red Hat Enterprise Linux from the network. As a result, it is possible to specify a name of a network interface identified by the subchannels instead of the default one.

(BZ#1980387)

The hostapd package has been added to RHEL 9.0

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#2019830)

ModemManager provided in version 1.18.2

RHEL 9.0 provides the **ModemManager** packages in upstream version 1.18.2. This version includes bug fixes and enhancements over the previous version, most notably:

- Improved capabilities and modes handling for devices with 5G capabilities
- Additional devices support

(BZ#1996716)

NetworkManager allows to change queue_id of bond port

NetworkManager ports in a bond now supports the **queue_id** parameter. Assuming **eth1** is a port of bond interface, you can enable **queue_id** for a bond port with:

```
# nmcli connection modify eth1 bond-port.queue-id 1
# nmcli connection up eth1
```

Any network interface that needs to use this option should configure it with multiple calls until proper priorities are set for all interfaces. For more information, see **/usr/share/docs/kernel-doc-_<version>/Documentation/networking/bonding.rst** file that is provided by the **kernel-doc** package.

(BZ#1949127)

Support for the configuration of blackhole, prohibit and unreachable route types with latest NetworkManager

Kernel supports several route types besides the common **unicast**, **broadcast** and **local** route types. In addition, users can now configure **blackhole**, **prohibit** and **unreachable** static route types in the connection profile of the NetworkManager. The NetworkManager will add a profile when the profile is activated.

(BZ#2060013)

RoCE Express Adapters now use an improved interface naming scheme

With this enhancement, RDMA over Converged Ethernet (RoCE) Express adapters use the predictable interface naming scheme and the Peripheral Communication Interface on z-system (zPCI) connector. In this naming scheme, RHEL uses user identifier (UID) or function identifier (FID) to generate unique names. In case that no unique UID is available, RHEL uses FID to set the naming scheme.

(BZ#2091653)

4.9. KERNEL

Kernel version in RHEL 9.0

Red Hat Enterprise Linux 9.0 is distributed with the kernel version 5.14.0-70.

(BZ#2077836)

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#2091643)

Red Hat protects kernel symbols only for minor releases

Red Hat guarantees that a kernel module will continue to load in all future updates within an Extended Update Support (EUS) release, only if you compile the kernel module using protected kernel symbols. There is no kernel Application Binary Interface (ABI) guarantee between minor releases of RHEL 9.

(BZ#2059183)

RHEL 9 Beta kernels signed with trusted SecureBoot certificates

Previously, RHEL Beta releases required users to enroll a separate Beta public key using the Machine Owner Key (MOK) facility. Starting with RHEL 9 Beta, kernels are signed with trusted SecureBoot certificates, hence users no longer need to enroll a separate Beta public key to use the beta versions on systems having UEFI Secure Boot enabled.

(BZ#2002499)

cgroup-v2 enabled by default in RHEL 9

The control groups version 2 (**cgroup-v2**) feature implements a single hierarchy model that simplifies the management of control groups. Also, it ensures that a process can only be a member of a single control group at a time. Deep integration with **systemd** improves the end-user experience when configuring resource control on a RHEL system.

Development of new features is mostly done for **cgroup-v2**, which has some features that are missing in **cgroup-v1**. Similarly, **cgroup-v1** contains some legacy features that are missing in **cgroup-v2**. Also, the control interfaces are different. Therefore, third party software with direct dependency on **cgroup-v1** may not run properly in the **cgroup-v2** environment.

To use **cgroup-v1**, you need to add the following parameters to the kernel command-line:

```
systemd.unified_cgroup_hierarchy=0
systemd.legacy_systemd_cgroup_controller
```

NOTE

Both **cgroup-v1** and **cgroup-v2** are fully enabled in the kernel. There is no default control group version from the kernel point of view, and is decided by **systemd** to mount at startup.

(BZ#1953515)

Kernel changes potentially affecting third party kernel modules

Linux distributions with a kernel version prior to 5.9 supported exporting GPL functions as non-GPL functions. As a result, users could link proprietary functions to GPL kernel functions through the **shim** mechanism. With this release, the RHEL kernel incorporates upstream changes that enhance the ability of RHEL to enforce GPL by rebuffing **shim**.

IMPORTANT

Partners and independent software vendors (ISVs) should test their kernel modules with an early version of RHEL 9 to ensure their compliance with GPL.

(BZ#1960556)

The 64-bit ARM architecture has a 4 KB page size in RHEL 9

Red Hat has selected a 4 KB page size of physical memory for the 64-bit ARM architecture in Red Hat Enterprise Linux 9. This size pairs well with the workloads and memory amounts present on the majority of ARM-based systems. To employ large page sizes efficiently, use the huge pages option to address a greater amount of memory or workloads with large data sets.

For more information about huge pages see Monitoring and Managing System Status and Performance .

(BZ#1978382)

The strace utility now correctly displays SELinux context mismatches

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#2038965)

perf-top now can sort by a certain column

With this update to the **perf-top** system profiling tool, you can sort samples by an arbitrary event column. Previously, the events were sorted by the first column in case multiple events in a group were sampled. To sort the samples, use the **--group-sort-idx** command-line option and press a number key to sort the table by the matching data column. Note that column numbering starts from **0**.

(BZ#1851933)

New package: jigawatts

Checkpoint/Restore In Userspace (CRIU) is a Linux utility that allows checkpointing and restoring of processes. The **jigawatts** package contains a Java library, which aims to improve the usability of CRIU mechanisms from Java applications.

(BZ#1972029)

The trace-cmd reset command has new behavior

Previously, the **trace-cmd reset** command resetted the **tracing_on** configuration to 0. The new behavior of **trace-cmd reset** is to reset **tracing_on** to its default value 1.

(BZ#1933980)

Extended Berkeley Packet Filter is supported in RHEL 9

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. Then the bytecode is verified and translated to the native machine code with just-in-time compilation. Finally, the virtual machine executes the code.

Red Hat ships numerous components that utilize the **eBPF** virtual machine. In RHEL 9, these components include:

- The **BPF Compiler Collection (BCC)** 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 **bpftrace** tracing language.
- The **libbpf** package, which is crucial for **bpf** development and **bpf**-related applications like **bpftrace**.
  - The **XDP** and **AF_XDP** API parts of the **libbpf** library are not supported and may be removed in a future release.
- The **eBPF for Traffic Control (tc)** feature, which enables programmable packet processing inside the kernel network data path.
- The **eXpress Data Path (XDP)** feature, which provides access to the received packets before the kernel networking stack processes them. Red Hat supports **XDP** only if it is used through the **libxdp** library.

- The **xdp-tools** package, which contains user-space support utilities for the **XDP** feature and is supported on the AMD64 and Intel64 CPU architectures. The **xdp-tools** package includes:
  - The **libxdp** library.
  - The **xdp-loader** utility for loading XDP programs.
  - The **xdp-filter** example program for packet filtering.
  - The **xdpdump** utility for capturing packets from a network interface with **XDP** enabled. The **xdpdump** utility is currently supported only on AMD64 and Intel64 CPU architectures. It is available for other architectures as Technology Preview.
- The **AF_XDP** socket for connecting the **eXpress Data Path (XDP)** path to user-space.

(BZ#2070506)

RHEL 9 provides the crash utility version 8.0.0

RHEL 9 is distributed with the **crash** utility version 8.0.0. The bug fixes and and notable enhancements include:

- Adds the new **offset** parameter in the **add-symbol-file** command. This support helps to set the **kaslr_offset** to **gdb**.
- Upgrades the **gdb-7.6** to **gdb-10.2**.

(BZ#1896647)

makedumpfile now supports an improved zstd compression capability

With this enhancement, the **makedumpfile** now includes the Zstandard (**zstd**) compression capability, which provides high compression ratios. This improvement helps specifically on large memory systems.

The **zstd** compression capability now has a good balance between the **vmcore** dump size and the compression time consumption as compared to prior compression ratios. As a result, the improved compression mechanism now creates a smaller **vmcore** file with an acceptable good compression time.

Note that a good compression ratio also depends on how the system is being used and the data type stored in RAM.

(BZ#1988894)

numatop enabled on Intel Xeon scalable server processors

numatop is a tool that tracks and analyzes the behavior of the processes and threads running on NUMA systems and displays metrics which can identify NUMA-related performance bottlenecks.

numatop uses Intel performance counter sampling technologies and associates the performance data with Linux system **runtime** information, to provide analysis in production systems.

(BZ#1874125)

kexec_file_load has been added as the default option for RHEL 9

This update adds the **kexec_file_load** system call for the 64-bit ARM architecture. It provides an in-kernel **kexec** loader for **kdump**. Previously, the kernel prevented the loading of unsigned kernel images when the secure boot option was enabled. The **kdump** mechanism would first try to detect whether

secure boot is enabled and then choose the boot interface to run. Consequently, an unsigned kernel failed to load with secure boot enabled and **kexec_file_load()** specified.

This update fixes the problem and an unsigned kernel works correctly in the described scenario.

(BZ#1895232)

makedumpfile now includes improved options to get an estimated vmcore size

With this implementation, the **makedumpfile** utility now includes the following options which help to print an estimate for the dump size for the currently running kernel:

- **--dry-run** performs all operations specified by the other options but does not write the output file.
- **--show-stats** prints the report messages. This is an alternative to enabling bit 4 in the level provided to **--message-level** option.

The following example shows the **--dry-run** and **--show-stats** usage:

```
$ makedumpfile --dry-run --show-stats -l --message-level 7 -d 31 /proc/kcore dump.dummy
```

Note that the dump file size may vary depending on the system state at the time of panic and the estimate provided by the options may differ from the actual state.

(BZ#1958452)

The kexec-tools package now supports the default crashkernel memory reservation values for RHEL 9

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

To query the default crashkernel value:

```
$ kdumpctl get-default-crashkernel
```

If the memory reserved by the default **crashkernel** value is not sufficient on your system, increase the **crashkernel** parameter.

Note that the **crashkernel=auto** option in the boot command line is no longer supported in RHEL 9 and later releases.

For more information, see the **/usr/share/doc/kexec-tools/crashkernel-howto.txt** file.

(BZ#2034490)

Core scheduling is supported in RHEL 9

With the core scheduling functionality users can prevent tasks that should not trust each other from sharing the same CPU core. Likewise, users can define groups of tasks that can share a CPU core.

These groups can be specified:

- To improve security by mitigating some cross-Symmetric Multithreading (SMT) attacks

- To isolate tasks that need a whole core. For example for tasks in real-time environments, or for tasks that rely on specific processor features such as Single Instruction, Multiple Data (SIMD) processing

For more information, see Core Scheduling.

(JIRA:RHELPLAN-100497)

Performance improved on 64-bit ARM architecture using non-strict iommu mode as default

With this upgrade, the 64-bit ARM architecture defaults to using the lazy direct memory access (DMA) domain for system memory management unit (SMMU). While bringing a significant performance gain, it can introduce a window between an address unmap and a Translation Lookaside Buffer (TLB) flush on SMMU. On previous versions, the 64-bit ARM architecture configured the strict DMA domains as default, which caused the performance to drop due to the 4KB page size.

If you need to use the strict DMA domain mode, specify the **iommu.strict=1** mode using the kernel command-line. Note that using strict DMA domains can cause performance drops on 64-bit ARM architectures.

(BZ#2050415)

The kernel-rt source tree has been updated to RHEL 9.0 tree

The **kernel-rt** sources have been updated to use the latest Red Hat Enterprise Linux kernel source tree. The real-time patch set has also been updated to the latest upstream version, v5.15-rt19. These updates provide a number of bug fixes and enhancements.

(BZ#2002474)

Support for CPU hotplug in the hv_24x7 and hv_gpci PMUs

With this update, PMU counters correctly react to the hot-plugging of a CPU. As a result, if a **hv_gpci** event counter is running on a CPU that gets disabled, the counting redirects to another CPU.

(BZ#1844416)

Metrics for POWERPC hv_24x7 nest events are now available

Metrics for POWERPC **hv_24x7** nest events are now available for **perf**. By aggregating multiple events, these metrics provide a better understanding of the values obtained from **perf** counters and how effectively the CPU is able to process the workload.

(BZ#1780258)

The IRDMA driver has been introduced in RHEL 9

The IRDMA driver enables RDMA functionality on RDMA-capable Intel® network devices. Devices supported by this driver are:

- Intel® Ethernet Controller E810
- Intel® Ethernet Network Adapter X722

RHEL 9 delivers updated Intel® Ethernet Protocol Driver for RDMA (IRDMA) for the X722 Internet Wide-area RDMA Protocol (iWARP) device. RHEL 9 also introduces a new E810 device that supports iWARP and RDMA over Converged Ethernet (RoCEv2). The IRDMA module replaces the legacy i40iw

module for X722 and extends the Application Binary Interface (ABI) defined for i40iw. The change is backward compatible with legacy X722 RDMA-Core provider (libi40iw).

- The X722 device supports only iWARP and a more limited set of configuration parameters.
- The E810 device supports the following set of RDMA and congestion management features:
  - iWARP and RoCEv2 RDMA transports
  - Priority Flow Control (PFC)
  - Explicit Congestion Notification (ECN)

(BZ#1874195)

A new parameter for the kernel bonding module: lacp_active

RHEL 9 introduces the **lacp_active** parameter for the **bonding** kernel module. This parameter specifies whether to send Link Aggregation Control Protocol Data Unit (LACPDU) frames at specified intervals. The options are as follows:

- **on** (default) - enables to send the LACPDU frames along with the configured **lacp_rate** parameter
- **off** - the LACPDU frames act as "speak when spoken to"

Note that the LACPDU state frames are still sent when you initialize or unbind port.

(BZ#1951951)

4.10. BOOT LOADER

Boot loader configuration files are unified across CPU architectures

Configuration files for the GRUB boot loader are now stored in the **/boot/grub2/** directory on all supported CPU architectures. The **/boot/efi/EFI/redhat/grub.cfg** file, which GRUB previously used as the main configuration file on UEFI systems, now simply loads the **/boot/grub2/grub.cfg** file.

This change simplifies the layout of the GRUB configuration file, improves user experience, and provides the following notable benefits:

- You can boot the same installation with either EFI or legacy BIOS.
- You can use the same documentation and commands for all architectures.
- GRUB configuration tools are more robust, because they no longer rely on symbolic links and they do not have to handle platform-specific cases.
- The usage of the GRUB configuration files is aligned with images generated by CoreOS Assembler (COSA) and OSBuild.
- The usage of the GRUB configuration files is aligned with other Linux distributions.

(JIRA:RHELPLAN-101246)

4.11. FILE SYSTEMS AND STORAGE

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#2065646)

Changes in the NFS client and server in RHEL 9

- RHEL 9.0 NFS server and client no longer support the insecure GSS Kerberos 5 encryption type **des-cbc-crc**.
- NFS client no longer supports mounting filesystems using UDP transports.

(BZ#1952863)

GFS2 file systems are now created with format version 1802

GFS2 file systems in RHEL 9 are created with format version 1802. This enables the following features:

- Extended attributes in the **trusted** namespace ("trusted.* xattrs") are recognized by **gfs2** and **gfs2-utils**.
- The **rgrplvb** option is active by default. This allows **gfs2** to attach updated resource group data to DLM lock requests, so the node acquiring the lock does not need to update the resource group information from disk. This improves performance in some cases.

File systems created with the new format version will not be able to be mounted under earlier RHEL versions and older versions of the **fsck.gfs2** utility will not be able to check them.

Users can create a file system with the older format version by running the **mkfs.gfs2** command with the option **-o format=1801**.

Users can upgrade the format version of an older file system running **tunegfs2 -r 1802** ***device*** on an unmounted file system. Downgrading the format version is not supported.

(BZ#1616432)

RHEL 9 provides nvml package version 1.10.1

RHEL 9.0 updates the **nvml** package to version 1.10.1. This update adds features and fixes a potential data corruption bug on power loss.

(BZ#1874208)

Support for exFAT file system has been added

RHEL 9.0 supports Extensible File Allocation Table (exFAT) file system. You can now mount, format, and generally use this file system, which is usually used by default on flash memory.

(BZ#1943423)

rpcctl command now displays SunRPC connection information

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

(BZ#2059245)

Limiting the set of the devices for LVM

By default, LVM in RHEL 9 uses only the devices that you explicitly select. Use the new commands **lvmdevices** and **vgimportdevices** to select specific devices. Using the **pvcreate**, **vgcreate**, and **vgextend** commands indirectly selects new devices for **lvm**, if they have not already been selected. LVM ignores devices that are attached to the system until you select them by using one of these commands.

The **lvm** command saves the list of the selected devices in the devices file **/etc/lvm/devices/system.devices**. The **lvm.conf** filter or any other command-line configuration filter does not function when you enable the new devices file feature. If you remove or disable the devices file, LVM applies the filter to all attached devices. For detailed information about this feature, see the **lvmdevices(8)** man page.

(BZ#1749513)

NVMe/TCP host with nvme_tcp.ko is now fully supported

Nonvolatile Memory Express (NVMe) storage over TCP/IP networks (NVMe/TCP) with the **nvme_tcp.ko** kernel module is now fully supported. The NVMe/TCP target with the **nvmet_tcp.ko** module is available with an Unmaintained status in RHEL 9.0.

(BZ#2054441)

multipathd now supports detecting FPIN-Li events

When you add a new value **fpin** for the **marginal_pathgroups** config option, you enable **multipathd** to monitor the Link Integrity Fabric Performance Impact Notification (PFIN-Li) events and move paths with link integrity issues to a marginal pathgroup. With the **fpin** value set, **multipathd** overrides its existing marginal path detection methods and relies on the Fibre Channel fabric to identify link integrity issues.

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

(BZ#2053642)

4.12. HIGH AVAILABILITY AND CLUSTERS

The resource-stickiness resource meta-attribute now defaults to 1 instead of 0 for newly-created clusters

Previously, the default value for the **resource-stickiness** resource meta-attribute had a default value of 0 for newly-created clusters. This meta-attribute now defaults to 1.

With a stickiness of 0, a cluster may move resources as needed to balance resources across nodes. This may result in resources moving when unrelated resources start or stop. With a positive stickiness, resources have a preference to stay where they are, and move only if other circumstances outweigh the stickiness. This may result in newly-added nodes not getting any resources assigned to them without administrator intervention. Both approaches have potentially unexpected behavior, but most users prefer having some stickiness. The default value for this meta-attribute has been changed to 1 to reflect this preference.

Only newly-created clusters are affected by this change, so the behavior does not change for existing clusters. Users who prefer the old behavior for their cluster can delete the **resource-stickiness** entry from resource defaults.

(BZ#1850145)

New LVM volume group flag to control autoactivation

LVM volume groups now support a **setautoactivation** flag which controls whether logical volumes that you create from a volume group will be automatically activated on startup. When creating a volume group that will be managed by Pacemaker in a cluster, set this flag to **n** with the **vgcreate --**

setautoactivation n command for the volume group to prevent possible data corruption. If you have an existing volume group used in a Pacemaker cluster, set the flag with **vgchange --setautoactivation n**.

(BZ#1899214)

New pcs resource status display commands

The **pcs resource status** and the **pcs stonith status** commands now support the following options:

- You can display the status of resources configured on a specific node with the **pcs resource status node=*node_id*** command and the **pcs stonith status node=*node_id*** command. You can use these commands to display the status of resources on both cluster and remote nodes.
- You can display the status of a single resource with the **pcs resource status *resource_id*** and the **pcs stonith status *resource_id*** commands.
- You can display the status of all resources with a specified tag with the **pcs resource status *tag_id*** and the **pcs stonith status *tag_id*** commands.

(BZ#1290830, BZ#1285269)

New reduced output display option for pcs resource safe-disable command

The **pcs resource safe-disable** and **pcs resource disable --safe** commands print a lengthy simulation result after an error report. You can now specify the **--brief** option for those commands to print errors only. The error report now always contains resource IDs of affected resources.

(BZ#1909901)

New pcs command to update SCSI fencing device without causing restart of all other resources

Updating a SCSI fencing device with the **pcs stonith update** command causes a restart of all resources running on the same node where the stonith resource was running. The new **pcs stonith update-scsi-devices** command allows you to update SCSI devices without causing a restart of other cluster resources.

(BZ#1872378)

Ability to configure watchdog-only SBD for fencing on subset of cluster nodes

Previously, to use a watchdog-only SBD configuration, all nodes in the cluster had to use SBD. That prevented using SBD in a cluster where some nodes support it but other nodes (often remote nodes) required some other form of fencing. Users can now configure a watchdog-only SBD setup using the new **fence_watchdog** agent, which allows cluster configurations where only some nodes use watchdog-only SBD for fencing and other nodes use other fencing types. A cluster may only have a single such device, and it must be named **watchdog**.

(BZ#1443666)

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)

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)

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)

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)

Local mode version of pcs cluster setup command is now fully supported

By default, the **pcs cluster setup** command automatically synchronizes all configuration files to the cluster nodes. The **pcs cluster setup** command now fully supports the **--corosync-conf** option. Specifying this option switches the command to **local** mode. In this mode, the **pcs** command-line interface creates a **corosync.conf** file and saves it to a specified file on the local node only, without communicating with any other node. This allows you to create a **corosync.conf** file in a script and handle that file by means of the script.

(BZ#2008558)

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. By default, the location constraint that the command creates is automatically removed once the resource has been moved. This does not necessarily move the resources back to the original node; where the resources can run at that point depends on how you have configured your resources initially. If you would like to move a resource and leave the resulting constraint in place, use the **pcs resource move-with-contraint** command.

(BZ#2008575)

pcs suppport for OCF Resource Agent API 1.1 standard

The **pcs** command-line interface now supports OCF 1.1 resource and STONITH agents. As part of the implementation of this support, any agent's metadata must comply with the OCF schema, whether the agent is an OCF 1.0 or OCF 1.1 agent. If an agent's metadata does not comply with the OCF 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 now omit agents with invalid metadata from the listing.

(BZ#2018969)

pcs now accepts Promoted and Unpromoted as role names

The **pcs** command-line interface now accepts **Promoted** and **Unpromoted** anywhere roles are specified in Pacemaker configuration. These role names are the functional equivalent of the **Master** and **Slave** Pacemaker roles in previous RHEL releases, and these are the role names that are visible in configuration displays and help pages.

(BZ#2009455)

Updated version of pcsd Web UI

The **pcsd** Web UI, the graphical user interface to create and configure Pacemaker/Corosync clusters, has been updated. The updated Web UI provides an improved user experience and a standardized interface that is built with the PatternFly framework used in other Red Hat web applications.

(BZ#1996067)

4.13. DYNAMIC PROGRAMMING LANGUAGES, WEB AND DATABASE SERVERS

Python in RHEL 9

Python 3.9 is the default **Python** implementation in RHEL 9. **Python 3.9** is distributed in a non-modular **python3** RPM package in the BaseOS repository and usually installed by default. **Python 3.9** will be supported for the whole life cycle of RHEL 9.

Additional versions of **Python 3** will be distributed as RPM packages with a shorter life cycle through the AppStream repository and will be installable in parallel.

The **python** command (**/usr/bin/python**), as well as other **Python**-related commands such as **pip**, are available in the unversioned form and point to the default **Python 3.9** version.

Python 2 is not distributed with RHEL 9.

For more information about **Python** in RHEL 9, see Introduction to Python.

(BZ#1941595, JIRA:RHELPLAN-80598)

Node.js 16 available in RHEL 9

RHEL 9 provides a Long Term Support (LTS) version 16 of **Node.js**, a software development platform for building fast and scalable network applications in the JavaScript programming language.

Notable changes in **Node.js 16** over **Node.js 14** include:

- The **V8** engine has been upgraded to version 9.4.
- The **npm** package manager has been upgraded to version 8.3.1.
- A new **Timers Promises** API provides an alternative set of timer functions that return **Promise** objects.
- **Node.js** is now compatible with **OpenSSL 3.0**.
- **Node.js** now provides a new experimental **Web Streams** API and an experimental ECMAScript modules (ESM) loader hooks API.

Node.js 16 is the initial version of this Application Stream, which you can install easily as an RPM package. **Node.js 16** has a shorter life cycle than RHEL 9. For details, see the Red Hat Enterprise Linux

Application Streams Life Cycle document. Additional **Node.js** versions will be provided as modules also with a shorter life cycle in future minor releases of RHEL 9.

(BZ#1953491)

RHEL 9 provides Ruby 3.0

RHEL 9 is distributed with **Ruby 3.0.3**, which provides a number of performance improvements, bug and security fixes, and new features over **Ruby 2.7**.

Notable enhancements include:

- Concurrency and parallelism features:
  - **Ractor**, an Actor-model abstraction that provides thread-safe parallel execution, is provided as an experimental feature.
  - **Fiber Scheduler** has been introduced as an experimental feature. **Fiber Scheduler** intercepts blocking operations, which enables light-weight concurrency without changing existing code.
- Static analysis features:
  - The **RBS** language has been introduced which describes the structure of **Ruby** programs. The **rbs** gem has been added to parse type definitions written in **RBS**.
  - The **TypeProf** utility has been introduced which is a type analysis tool for **Ruby** code.
- Pattern matching with the **case/in** expression is no longer experimental.
- One-line pattern matching, which is an experimental feature, has been redesigned.
- Find pattern has been added as an experimental feature.

The following performance improvements have been implemented:

- Pasting long code to the **Interactive Ruby Shell (IRB)** is now significantly faster.
- The **measure** command has been added to **IRB** for time measurement.

Other notable changes include:

- Keyword arguments are now separated from other arguments.
- The default directory for user-installed gems is now **$HOME/.local/share/gem/** unless the **$HOME/.gem/** directory is already present.

Ruby 3.0 is the initial version of this Application Stream which you can install easily as an RPM package. Additional **Ruby** versions will be provided as modules with a shorter life cycle in future minor releases of RHEL 9.

(JIRA:RHELPLAN-80758)

RHEL 9 introduces Perl 5.32

RHEL 9 includes **Perl 5.32**, which provides a number of bug fixes and enhancements over version 5.30.

Notable enhancement include:

- **Perl** now supports Unicode version 13.0.
- The **qr** quote-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.

Perl 5.32 is the initial version of this Application Stream, which you can install easily as an RPM package. Additional **Perl** versions will be provided as modules with a shorter life cycle in future minor releases of RHEL 9.

(JIRA:RHELPLAN-80759)

RHEL 9 includes PHP 8.0

RHEL 9 is distributed with **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.
- New Just-In-Time compilation engines significantly improve application 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**.

PHP 8.0 is the initial version of this Application Stream, which you can install easily as an RPM package. Additional **PHP** versions will be provided as modules with a shorter life cycle in future minor releases of RHEL 9.

For more information, see Using the PHP scripting language.

(BZ#1949319)

RHEL 9 provides Git 2.31 and Git LFS 2.13

RHEL 9 is distributed with **Git 2.31** which provides a number of enhancements and performance improvements over version 2.27 available in RHEL 8. Notable changes include:

- The **git status** command now reports the status of sparse checkout.
- You can now use the **--add-file** option with the **git archive** command to include untracked files in a snapshot from a tree-ish identifier.
- You can use the **clone.defaultremotename** configuration variable to customize a nickname of the source remote repository.
- You can configure the maximum length of output file names created by the **git format-patch** command. Previously, the length limit was 64 bytes.
- Support for the deprecated PCRE1 library has been removed.

Additionally, the **Git Large File Storage (LFS)** extension version 2.13 is now available. Enhancements over version 2.11 distributed in RHEL 8 include:

- **Git LFS** now supports SHA-256 repositories.
- **Git LFS** now supports the **socks5h** protocol.
- A new **--worktree** option is available for the **git lfs install** and **git lfs uninstall** commands.
- A new **--above** parameter is available for the **git lfs migrate import** command.

(BZ#1956345, BZ#1952517)

Subversion 1.14 in RHEL 9

RHEL 9 is distributed with **Subversion 1.14**. **Subversion 1.14** is the initial version of this Application Stream, which you can install easily as an RPM package. Additional **Subversion** versions will be provided as modules with a shorter life cycle in future minor releases of RHEL 9.

(JIRA:RHELPLAN-82578)

Notable changes in the Apache HTTP Server

RHEL 9.0 provides version 2.4.51 of the Apache HTTP Server. Notable changes over version 2.4.37 include:

- Apache HTTP Server Control Interface (**apachectl**):
  - The **systemctl** pager is now disabled for **apachectl status** output.
  - The **apachectl** command now fails instead of giving a warning if you pass additional arguments.
  - The **apachectl graceful-stop** command now returns immediately.
  - The **apachectl configtest** command now executes the **httpd -t** command without changing the SELinux context.
  - The **apachectl(8)** man page in RHEL now fully documents differences from upstream **apachectl**.
- Apache eXtenSion tool (**apxs**):

    - The **/usr/bin/apxs** command no longer uses or exposes compiler optimisation flags as applied when building the **httpd** package. You can now use the **/usr/lib64/httpd/build/vendor-apxs** command to apply the same compiler flags as used to build **httpd**. To use the **vendor-apxs** command, you must install the **redhat-rpm-config** package first.

- Apache modules:
    - The **mod_lua** module is now provided in a separate package.
    - A new **mod_jk** connector for the Apache HTTP Server is a module that utilizes the Apache JServ Protocol (AJP) to connect web servers with Apache Tomcat and other backends.
    - A new **mod_proxy_cluster** module provides an httpd-based load balancer that uses a communication channel to forward requests from the load balancer to one of a set of application server nodes. The application server nodes use this connection to transmit server-side load balance factors and lifecycle events back to the load balancer through a custom set of HTTP methods called the Mod-Cluster Management Protocol (MCMP). This additional feedback channel allows **mod_proxy_cluster** to offer a level of intelligence and granularity not found in other load-balancing solutions. This module requires the **ModCluster** client to be installed on the backend server to successfully communicate.
- Configuration syntax changes:
    - In the deprecated **Allow** directive provided by the **mod_access_compat** module, a comment (the **#** character) now triggers a syntax error instead of being silently ignored.
- Other changes:
    - Kernel thread IDs are now used directly in error log messages, making them both accurate and more concise.
    - Many minor enhancements and bug fixes.
    - A number of new interfaces are available to module authors.

There are no backwards-incompatible changes to the **httpd** module API since RHEL 8.

Apache HTTP Server 2.4 is the initial version of this Application Stream, which you can install easily as an RPM package.

For more information, see Setting up the Apache HTTP web server.

(JIRA:RHELPLAN-68364, BZ#1931976, JIRA:RHELPLAN-80725)

nginx 1.20 available in RHEL 9

RHEL 9 includes the **nginx 1.20** web and proxy server. This release provides a number of bug fixes, security fixes, new features and enhancements over version 1.18.

New features:

- **nginx** now supports client SSL certificate validation with Online Certificate Status Protocol (OCSP).
- **nginx** now supports cache clearing based on the minimum amount of free space. This support is implemented as the **min_free** parameter of the **proxy_cache_path** directive.

- A new **ngx_stream_set_module** module has been added, which enables you to set a value for a variable.
- A new **nginx-mod-devel** package has been added, which provides all necessary files, including RPM macros and **nginx** source code, for building external dynamic modules for **nginx**.

Enhanced directives:

- Multiple new directives are now available, such as **ssl_conf_command** and **ssl_reject_handshake**.
- The **proxy_cookie_flags** directive now supports variables.

Improved support for HTTP/2:

- The **ngx_http_v2** module now includes the **lingering_close**, **lingering_time**, **lingering_timeout** directives.
- Handling connections in HTTP/2 has been aligned with HTTP/1.x. From **nginx 1.20**, use the **keepalive_timeout** and **keepalive_requests** directives instead of the removed **http2_recv_timeout**, **http2_idle_timeout**, and **http2_max_requests** directives.

nginx 1.20 is the initial version of this Application Stream, which you can install easily as an RPM package. Additional **nginx** versions will be provided as modules with a shorter life cycle in future minor releases of RHEL 9.

For more information, see Setting up and configuring NGINX.

(BZ#1953639, BZ#1991720)

Varnish Cache 6.6 in RHEL 9

RHEL 9 includes **Varnish Cache 6.6**, a high-performance HTTP reverse proxy.

Notable changes since version 6.0 include:

- Improved performance of log-processing tools, such as **varnishlog**
- Improved accuracy of statistics
- A number of optimizations in cache lookups
- Various configuration changes
- Numerous enhancements and bugs fixes

Varnish Cache 6 is the initial version of this Application Stream, which you can install easily as an RPM package.

(BZ#1984185)

RHEL 9 introduces Squid 5

RHEL 9 is distributed with **Squid 5.2**, a high-performance proxy caching server for web clients, supporting FTP, Gopher, and HTTP data objects. This release provides a number of bug fixes, security fixes, new features, and enhancements over version 4.

New features:

- **Squid** improves responsibility by using the Happy Eyeballs (HE) algorithm.
  - **Squid** now uses a received IP address as soon request forwarding requires it instead of waiting for all of the potential forwarding destinations to be fully resolved.
  - New directives are now available: **happy_eyeballs_connect_gap**, **happy_eyeballs_connect_limit**, and **happy_eyeballs_connect_timeout** directives.
  - The **dns_v4_first** directive has been removed.
- **Squid** now uses the **CDN-Loop** header as a source for loop detection in Content Delivery Networks (CDN).
- **Squid** introduces peering support for SSL bumping.
- A new Internet Content Adaptation Protocol (ICAP) trailers feature is available, which enables ICAP agents to reliably send message metadata after the message body.

Changes to configuration options:

- The **mark_client_packet** configuration option has replaced **clientside_mark**.
- The **shared_transient_entries_limit** configuration option has replaced **collapsed_forwarding_shared_entries_limit**.

Squid 5 is the initial version of this Application Stream, which you can install easily as an RPM package.

For more information, see Configuring the Squid caching proxy server.

(BZ#1990517)

MariaDB 10.5 in RHEL 9

RHEL 9 provides **MariaDB 10.5**. **MariaDB 10.5** is the initial version of this Application Stream, which you can install easily as an RPM package. Additional **MariaDB** versions will be provided as modules with a shorter life cycle in future minor releases of RHEL 9.

For more information, see Using MariaDB.

(BZ#1971248)

RHEL 9 includes MySQL 8.0

RHEL 9 is distributed with **MySQL 8.0**. **MySQL 8.0** is the initial version of this Application Stream, which you can install easily as an RPM package. **MySQL 8.0** has a shorter life cycle than RHEL 9. For details, see the Red Hat Enterprise Linux Application Streams Life Cycle document.

For information about usage, see Using MySQL.

(JIRA:RHELPLAN-78673)

RHEL 9 provides PostgreSQL 13

PostgreSQL 13 is available with RHEL 9. **PostgreSQL 13** is the initial version of this Application Stream, which you can install easily as an RPM package. Additional **PostgreSQL** versions will be provided as modules with a shorter life cycle in future minor releases of RHEL 9.

For more information, see Using PostgreSQL.

(JIRA:RHELPLAN-78675)

Redis 6.2 in RHEL 9

RHEL 9 is distributed with **Redis 6.2**, which provides a number of bug and security fixes and enhancements over version 6.0 available in RHEL 8.

Notably, **Redis** server configuration files are now located in a dedicated directory: **/etc/redis/redis.conf** and **/etc/redis/sentinel.conf**. In the RHEL 8 version, the location of these files was **/etc/redis.conf** and **/etc/redis-sentinel.conf** respectively.

Redis 6 is the initial version of this Application Stream, which you can install easily as an RPM package.

(BZ#1959756)

New package: perl-Module-Signature

RHEL 9 introduces the **perl-Module-Signature** Perl module. With this new module, you can enable signature checking for **cpan** to mitigate CVE-2020-16156. For more information, see How to mitigate CVE-2020-16154 in perl-App-cpanminus and CVE-2020-16156 in perl-CPAN.

(BZ#2039361)

4.14. COMPILERS AND DEVELOPMENT TOOLS

RHEL 9 provides support for IBM POWER10 processors

From the Linux kernel, through the system toolchain (GCC, binutils, glibc), Red Hat Enterprise Linux 9 has been updated to include support for IBM's latest POWER processor, POWER10. RHEL 9 is production ready for workloads on POWER10, with enhancements coming in future releases.

(BZ#2027596)

GCC 11.2.1 is available

RHEL 9 is distributed with GCC version 11.2.1. Notable bug fixes and enhancements include:

General improvements

- GCC now defaults to the DWARF Version 5 debugging format.
- Column numbers shown in diagnostics represent real column numbers by default and respect multicolumn characters.
- The straight-line code vectorizer considers the whole function when vectorizing.
- A series of conditional expressions that compare the same variable can be transformed into a switch statement if each of them contains a comparison expression.
- Interprocedural optimization improvements:
  - A new IPA-modref pass, controlled by the **-fipa-modref** option, tracks side effects of function calls and improves the precision of points-to analysis.
  - The identical code folding pass, controlled by the **-fipa-icf** option, was significantly improved to increase the number of unified functions and reduce compile-time memory use.

- Link-time optimization improvements:
  - Link-time optimization (LTO) enables the compiler to perform various optimizations across all translation units of your program by using its intermediate representation at link time. For more information, see Link time optimization.
  - Memory allocation during linking was improved to reduce peak memory use.
- Using a new **GCC_EXTRA_DIAGNOSTIC_OUTPUT** environment variable in IDEs, you can request machine-readable "fix-it hints" without adjusting build flags.
- The static analyzer, run by the **-fanalyzer** option, is improved significantly with numerous bug fixes and enhancements provided.

Language-specific improvements

C family

- C and C++ compilers support non-rectangular loop nests in OpenMP constructs and the allocator routines of the OpenMP 5.0 specification.
- Attributes:
  - The new **no_stack_protector** attribute marks functions that should not be instrumented with stack protection (**-fstack-protector**).
  - The improved **malloc** attribute can be used to identify allocator and deallocator API pairs.
- New warnings:
  - **-Wsizeof-array-div**, enabled by the **-Wall** option, warns about divisions of two **sizeof** operators when the first one is applied to an array and the divisor does not equal the size of the array element.
  - **-Wstringop-overread**, enabled by default, warns about calls to string functions that try to read past the end of the arrays passed to them as arguments.
- Enhanced warnings:
  - **-Wfree-nonheap-object** detects more instances of calls to deallocation functions with pointers not returned from a dynamic memory allocation function.
  - **-Wmaybe-uninitialized** diagnoses the passing of pointers and references to uninitialized memory to functions that take **const**-qualified arguments.
  - **-Wuninitialized** detects reads from uninitialized dynamically allocated memory.

C

- Several new features from the upcoming C2X revision of the ISO C standard are supported with the **-std=c2x** and **-std=gnu2x** options. For example:
  - The standard attribute is supported.
  - The **__has_c_attribute** preprocessor operator is supported.
  - Labels may appear before declarations and at the end of a compound statement.

C++

- The default mode is changed to **-std=gnu++17**.
- The C++ library **libstdc++** has improved C++17 support now.
- Several new C++20 features are implemented. Note that C++20 support is experimental. For more information about the features, see C++20 Language Features.
- The C++ front end has experimental support for some of the upcoming C++23 draft features.
- New warnings:
  - **-Wctad-maybe-unsupported**, disabled by default, warns about performing class template argument deduction on a type with no deduction guides.
  - **-Wrange-loop-construct**, enabled by **-Wall**, warns when a range-based for loop is creating unnecessary and resource inefficient copies.
  - **-Wmismatched-new-delete**, enabled by **-Wall**, warns about calls to operator delete with pointers returned from mismatched forms of operator new or from other mismatched allocation functions.
  - **-Wvexing-parse**, enabled by default, warns about the most vexing parse rule: the cases when a declaration looks like a variable definition, but the C++ language requires it to be interpreted as a function declaration.

Architecture-specific improvements

The 64-bit ARM architecture

- The Armv8-R architecture is supported through the **-march=armv8-r** option.
- GCC can autovectorize operations performing addition, subtraction, multiplication, and the accumulate and subtract variants on complex numbers.

AMD and Intel 64-bit architectures

- New ISA extension support for Intel AVX-VNNI is added. The **-mavxvnni** compiler switch controls the AVX-VNNI intrinsics.
- AMD CPUs based on the znver3 core are supported with the new **-march=znver3** option.
- Three microarchitecture levels defined in the x86-64 psABI supplement are supported with the new **-march=x86-64-v2**, **-march=x86-64-v3**, and **-march=x86-64-v4** options.

IBM Z architectures

- GCC 11.2.1 defaults to the IBM z14 processor.

IBM Power Systems

- GCC 11.2.1 defaults to the IBM POWER9 processor.
- The GCC compiler now supports POWER10 instructions with the new **-mcpu=power10** command-line option

(BZ#1986836, BZ#1870016, BZ#1870025, BZ#1870028, BZ#2019811, BZ#2047296)

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#2023422)

Notable changes to binutils

RHEL 9 introduces the following changes to **binutils**:

- **binutils** now supports Intel's AMX/TMUL instruction set, resulting in improved performance for applications which can make use of this new feature.
- The assembler, linker, and other binary utilities now support the POWER10 instructions.

(BZ#2030554, BZ#1870021)

sched_getcpu implementation can now, optionally, use rseq (restartable sequences) to improve performance on the 64-bit ARM architectures and other architectures

The previous implementation of **sched_getcpu** on the 64-bit ARM architectures uses the **getcpu** system call, which is too slow for efficient use in most parallel algorithms. Other architectures use vDSO (virtual dynamic shared object) acceleration to work around this. Implementing **sched_getcpu** using **rseq** greatly improves performance on the 64-bit ARM architectures. Other architectures see a slight improvement.

To configure **sched_getcpu** to use **rseq**, set the **GLIBC_TUNABLES=glibc.pthread.rseq=1** environment variable:

```
# GLIBC_TUNABLES=glibc.pthread.rseq=1
# export GLIBC_TUNABLES
```

(BZ#2024347)

Updated performance tools and debuggers

The following performance tools and debuggers are available with RHEL 9.0:

- GDB 10.2
- Valgrind 3.18.1
- SystemTap 4.6
- Dyninst 11.0.0
- elfutils 0.186

(BZ#2019806)

DAWR functionality improved in GDB on IBM POWER10

RHEL 9 is distributed with GDB 10.2 that provides improved DAWR functionality. New hardware watchpoint capabilities are enabled for GDB on the IBM POWER10 processors. For example, a new set of DAWR/DAWRX registers has been added.

(BZ#1870029)

GDB supports new prefixed instructions on IBM POWER10

GDB 10.2 fully supports the Power ISA 3.1 prefixed instructions on POWER10, which include eight-byte prefixed instructions. In RHEL 8.4, GDB only supported four-byte instructions.

(BZ#1870031)

RHEL 9 provides **boost** 1.75.0

RHEL 9 is distributed with the **boost** package version 1.75.0. Notable bug fixes and enhancements over version 1.67.0 include:

- The **Boost.Signals** library has been removed and replaced by the header-only **Boost.Signals2** component.
- The **bjam** tool in the **boost-jam** package has been replaced by **b2** in the **boost-b2** package.
- New libraries:
  - **Boost.Contracts**
  - **Boost.HOF**
  - **Boost.YAP**
  - **Boost.Safe Numerics**
  - **Boost.Outcome**
  - **Boost.Histogram**
  - **Boost.Variant2**
  - **Boost.Nowide**
  - **Boost.StaticString**
  - **Boost.STL_Interfaces**
  - **Boost.JSON**
  - **Boost.LEAF**
  - **Boost.PFR**

(BZ#1957950)

RHEL 9 provides LLVM Toolset 13.0.1

RHEL 9 is distributed with LLVM Toolset version 13.0.1. Notable bug fixes and enhancements over version 12.0.1 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**.
- Clang now supports guaranteed tail calls with statement attributes **[[clang::musttail]]** in C++ and **__attribute__((musttail))** in C.

In RHEL 9, you can install **llvm-toolset** easily as an RPM package.

(BZ#2001107)

Notable changes in CMake 3.20.2

RHEL 9 is distributed with CMake 3.20.2. To use CMake on a project that requires version 3.20.2 or less, use the command **cmake_minimum_required**(version 3.20.2).

Notable changes include:

- C++23 compiler modes can now be specified by using the target properties **CXX_STANDARD**, **CUDA_STANDARD**, **OBJCXX_STANDARD**, or by using the **cxx_std_23** meta-feature of the compile features function.
- CUDA language support now allows the NVIDIA CUDA compiler to be a symbolic link.
- The Intel oneAPI NextGen LLVM compilers are now supported with the **IntelLLVM** compiler ID.
- CMake now facilitates cross compiling for Android by merging with the Android NDK's toolchain file.
- When running **cmake(1)** to generate a project build system, unknown command-line arguments starting with a hyphen are now rejected.

For further information on new features and deprecated functionalities, see the CMake Release Notes.

(BZ#1957948)

RHEL 9 provides Go 1.17.7

RHEL 9 is distributed with Go Toolset version 1.17.7. Notable bug fixes and enhancements over version 1.16.7 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.

In RHEL 9, you can install **go-toolset** easily as an RPM package.

(BZ#2014087)

Go FIPS mode is supported with OpenSSL 3

You can now use the OpenSSL 3 library when in Go FIPS mode.

(BZ#1984110)

RHEL 9 provides Rust Toolset 1.58.1

RHEL 9 is distributed with Rust Toolset version 1.58.1. Notable bug fixes and enhancements over version 1.54.0 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 1.55Rust 1.56Rust 1.57Rust 1.58

In RHEL 9, you can install **rust-toolset** easily as an RPM package.

(BZ#2002885)

RHEL 9 provides the pcp package version 5.3.5

RHEL 9 is distributed with the Performance Co-Pilot (**pcp**) package version 5.3.5. Since version 5.3.1, a new **pcp-pmda-bpf** sub-package has been added which provides performance data from **eBPF** programs utilizing BPF CO-RE (**libbpf** and **BTF**).

(BZ#1991764)

Active Directory authentication for accessing SQL Server metrics in PCP

With this update, a system administrator can configure **pmdamssql(1)** to connect securely to the SQL Server metrics using Active Directory (AD) authentication.

(BZ#1847808)

The new pcp-ss PCP utility is now available

The **pcp-ss** PCP utility reports socket statistics collected by the **pmdasockets(1)** PMDA. The command is compatible with many of the **ss** command line options and reporting formats. It also offers the advantages of local or remote monitoring in live mode and historical replay from a previously recorded PCP archive.

(BZ#1981223)

RHEL 9 provides grafana 7.5.11

RHEL 9 is distributed with the **grafana** package version 7.5.11. Notable changes over version 7.5.9 include:

- Added a new **prepare time series** transformation for backward compatibility of panels that do not support the new data frame format.
- Updated password recovery functionality to use HMAC-SHA-256 instead of SHA-1 to generate password reset tokens.

(BZ#1993215)

RHEL 9 provides grafana-pcp 3.2.0

RHEL 9 is distributed with the **grafana-pcp** package version 3.2.0. Notable bug fixes and enhancements over version 3.1.0 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 didn't return all metric names.

(BZ#1993156)

Accessing remote hosts through a central pmproxy for the Vector data source in grafana-pcp

In some environments, the network policy does not allow connections from the dashboard viewer's browser to the monitored hosts directly. This update makes it possible to customize the **hostspec** in order to connect to a central **pmproxy**, which forwards the requests to the individual hosts.

(BZ#1845592)

A new package: ansible-pcp

The **ansible-pcp** package contains roles for Performance Co-Pilot (PCP) and related software, such as Redis and Grafana, used to implement the **metrics** RHEL system role.

(BZ#1957566)

RHEL 9 provides python-jsonpointer 2.0

RHEL 9 is distributed with the **python-jsonpointer** package version 2.0.

Notable changes over version 1.9 include:

- The Python versions 2.6 and 3.3 are deprecated.
- The **python-jsonpointer** module now automatically checks pointers for invalid escape sequences.
- You can now write pointers as arguments in the command line.
- Pointers can not be submitted in URL encoded format any more.

(BZ#1980256)

.NET 6.0 is available

RHEL 9 is distributed with **.NET** version 6.0. Notable improvements include:

- Support for 64-bit Arm (aarch64)
- Support for IBM Z and LinuxONE (s390x)

For more information, see Release Notes for .NET 6.0 RPM packages and Release Notes for .NET 6.0 containers.

.NET 6.0 is the initial version of this Application Stream, which you can install easily as an RPM package. **.NET 6.0** has a shorter life cycle than RHEL 9. For details, see the Red Hat Enterprise Linux Application Streams Life Cycle document.

(BZ#1986211)

Java implementations in RHEL 9

The RHEL 9 AppStream repository includes:

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

For more information, see OpenJDK documentation.

(BZ#2021262)

Java tools in RHEL 9

The RHEL 9 AppStream repository includes the following Java tools:

- **Maven 3.6.3**, a software project management and comprehension tool.
- **Ant 1.10.9**, a Java library and command-line tool for compiling, assembling, testing, and running Java applications.

Maven 3.6 and **Ant 1.10** are the initial versions of these Application Streams, which you can install easily as non-modular RPM packages.

(BZ#1951482)

SWIG 4.0 available in the CRB repository

The Simplified Wrapper and Interface Generator (SWIG) version 4.0 is available in the CodeReady Linux Builder (CRB) repository. This release adds support for **PHP 8**.

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

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

(BZ#1943580)

4.15. IDENTITY MANAGEMENT

Directory Server no longer uses a global changelog

With this enhancement, the Directory Server changelog has been integrated into the main database. Previously, Directory Server used a global changelog. However, this could cause issues if the directory used multiple databases. As a result, each suffix has now its own changelog in the same directory as the regular database files.

(BZ#1805717)

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.

(BZ#1859252)

Support for managing subID ranges is available in IdM

With this update, you can manage ID subranges for users in Identity Management. You can use the **ipa** CLI tool or IdM WebUI interface to assign automatically configured subID ranges to a user, which might be useful in a containerized environment.

(BZ#1952028)

Identity Management installation packages have been demodularized

Previously in RHEL 8, IdM packages were distributed as modules, which required you to enable a stream and install the profile that corresponds to your desired installation. IdM installation packages have been demodularized in RHEL 9, so you can use the following **dnf** commands to install IdM server packages:

For a server without integrated DNS services:

```
# dnf install ipa-server
```

For a server with integrated DNS services:

```
# dnf install ipa-server ipa-server-dns
```

(BZ#2080875)

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)

Automatic private groups for AD users support centralized configuring

You can now centrally define how compatible versions of SSSD on IdM clients manage private groups for users from trusted Active Directory domains. With this enhancement, you can now explicitly set the value for SSSD's **auto_private_groups** option for an ID range that handles AD users.

When the **auto_private_groups** option is not explicitly set, it uses a default value:

- For an **ipa-ad-trust-posix** ID range, the default value is **false**. SSSD always uses the **uidNumber** and **gidNumber** of the AD entry. A group with the **gidNumber** must exist in AD.
- For an **ipa-ad-trust** ID range, the default value is **true**. SSSD maps the **uidNumber** from the entry SID, the **gidNumber** is always set to the same value, and a private group is always mapped.

You can also set **auto_private_groups** to a third setting: **hybrid**. With this setting, SSSD maps a private group if the user entry has a GID equal to the UID but there is no group with this GID. If the UID and GID are different, a group with this GID number must exist.

This feature is useful for administrators that want to stop maintaining separate group objects for the user private groups, but also want to retain the existing user private groups.

(BZ#1957736)

Customizable logging settings for BIND

With this enhancement, you can now configure logging settings for the BIND DNS server component of an Identity Management server in the **/etc/named/ipa-logging-ext.conf** configuration file.

(BZ#1966101)

Autodiscovery of IdM servers when retrieving an IdM keytab

With this enhancement, you no longer need to specify an IdM server host name when retrieving a Kerberos keytab with the **ipa-getkeytab** command. If you do not specify a server host name, DNS discovery is used to find an IdM server. If no servers are found, the command falls back to the **host** value specified in the **/etc/ipa/default.conf** configuration file.

(BZ#1988383)

RHEL 9 provides Samba 4.15.5

RHEL 9 is distributed with Samba 4.15.5, which provides bug fixes and enhancements over version 4.14:

- 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#2013578)

Tracking client requests using the log analyzer tool

The System Security Services Daemon (SSSD) now includes a log parsing tool which tracks requests from start to finish across log files from multiple SSSD components.

The log analyzer tool allows you to more easily review SSSD debug logs to help you to troubleshoot any issues in SSSD. For example, you can extract and print SSSD logs pertaining only to certain client requests across SSSD processes. To run the analyzer tool, use the **sssctl analyze** command.

(JIRA:RHELPLAN-97899)

SSSD now logs backtraces by default

With this enhancement, SSSD now stores detailed debug logs in an in-memory buffer and appends them to log files when a failure occurs. By default, the following error levels trigger a backtrace:

- Level 0: fatal failures
- Level 1: critical failures
- Level 2: serious failures

You can modify this behavior for each SSSD process by setting the **debug_level** option in the corresponding section of the **sssd.conf** configuration file:

- If you set the debugging level to 0, only level 0 events trigger a backtrace.
- If you set the debugging level to 1, levels 0 and 1 trigger a backtrace.
- If you set the debugging level to 2 or higher, events at level 0 through 2 trigger a backtrace.

You can disable this feature per SSSD process by setting the **debug_backtrace_enabled** option to **false** in the corresponding section of **sssd.conf**:

```
[sssd]
debug_backtrace_enabled = true
debug_level=0
...

[nss]
debug_backtrace_enabled = false
...

[domain/idm.example.com]
debug_backtrace_enabled = true
debug_level=2
...

...
```

(BZ#1949149)

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#2014249)

Directory Server 12.0 is based on upstream version 2.0.14

Directory Server 12.0 is based on upstream version 2.0.14 which provides a number of bug fixes and enhancements over the previous version. For a complete list of notable changes, read the upstream release notes before updating:

- https://directory.fedoraproject.org/docs/389ds/releases/release-2-0-14.html
- https://directory.fedoraproject.org/docs/389ds/releases/release-2-0-13.html
- https://directory.fedoraproject.org/docs/389ds/releases/release-2-0-12.html
- https://directory.fedoraproject.org/docs/389ds/releases/release-2-0-11.html
- https://directory.fedoraproject.org/docs/389ds/releases/release-2-0-10.html
- https://directory.fedoraproject.org/docs/389ds/releases/release-2-0-9.html
- https://directory.fedoraproject.org/docs/389ds/releases/release-2-0-8.html
- https://directory.fedoraproject.org/docs/389ds/releases/release-2-0-7.html
- https://directory.fedoraproject.org/docs/389ds/releases/release-2-0-6.html
- https://directory.fedoraproject.org/docs/389ds/releases/release-2-0-5.html
- https://directory.fedoraproject.org/docs/389ds/releases/release-2-0-4.html
- https://directory.fedoraproject.org/docs/389ds/releases/release-2-0-3.html
- https://directory.fedoraproject.org/docs/389ds/releases/release-2-0-2.html
- https://directory.fedoraproject.org/docs/389ds/releases/release-2-0-1.html

(BZ#2024693)

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#2088414)

FreeRADIUS support is now redesigned

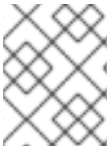
In RHEL 9, the existing FreeRADIUS offering is now streamlined and aligned more closely with the strategic direction of Identity Management (IdM). In order to provide the best support for IdM customers, Red Hat is strengthening support for these external authentication modules with FreeRADIUS:

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

The following modules are no longer supported:

- The MySQL, PostgreSQL, SQlite, and unixODBC database connectors

- The **Perl** language module
- The REST API module



NOTE

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

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

In addition, the scope of support for the **freeradius** package is now limited to the following use cases:

- Using FreeRADIUS as a wireless-authentication provider with IdM as the backend source of authentication. The authentication is happening 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.

(JIRA:RHELDOCS-17553)

4.16. DESKTOP

GNOME updated to version 40

The GNOME environment is now updated from GNOME 3.28 to GNOME 40 with many new features.

GNOME 40 includes a new and improved **Activities Overview** design. This gives the overview a more coherent look, and provides an improved experience for navigating the system and launching applications. Workspaces are now arranged horizontally, and the window overview and application grid are accessed vertically.

Other improvements to GNOME include:

- The performance and resource usage of GNOME has been significantly improved.
- The visual style, including the user interface, the icons, and the desktop, has been refreshed.
- GNOME applications no longer use the application menu, which was available from the top panel. The functionality is now located in a primary menu within the application window.
- The **Settings** application has been redesigned.
- Screen sharing and remote desktop sessions have been improved.
- If you use the proprietary NVIDIA drivers, you can now launch applications using the discrete GPU:
  a. Open the overview.
  b. Right-click the application icon in the dash.
  c. Select the **Launch on Discrete GPU**item in the menu.

- The **Power Off / Log Out** menu now includes the **Suspend** option and a new **Restart** option, which can reboot the system to the boot loader menu when you hold **Alt**.
- Flatpak applications now update automatically.
- You can now group application icons in the overview together into folders using drag and drop.
- The **Terminal** application now supports right-to-left and bi-directional text.
- The **Pointer Location** accessibility feature now works in the Wayland session. When the feature is enabled, pressing **Ctrl** highlights the pointer location on the screen.
- GNOME shell extensions are now managed by the **Extensions** application, rather than **Software**. The **Extensions** application handles updating extensions, configuring extension preferences, and removing or disabling extensions.
- The notifications popover now includes a **Do Not Disturb** button. When the button enabled, notifications do not appear on the screen.
- System dialogs that require a password now have an option to reveal the password text by clicking the eye (▌) icon.
- The **Software** application now automatically detects metered networks, such as mobile data networks. When the current network is metered, **Software** pauses updates in order to reduce data usage.
- Each connected display can now use a different refresh rate in the Wayland session.
- Fractional display scaling is available as an experimental option. It includes several preconfigured fractional ratios.
  To enable the experimental fractional scaling, add the **scale-monitor-framebuffer** value to the list of enabled experimental features:

```
$ dconf write \
    /org/gnome/mutter/experimental-features \
    "['scale-monitor-framebuffer']"
```

  As a result, fractional scaling options are accessible on the **Display** panel in **Settings**.

For more details on the changes in GNOME, see versions 3.30 to 40.0 in Release Notes.

(JIRA:RHELPLAN-101240)

PipeWire is now the default audio service

The **Pipewire** service now manages all audio output and input. **Pipewire** replaces the **PulseAudio** service in general use cases and the **JACK** service in professional use cases. The system now redirects audio from applications that use **PulseAudio**, **JACK**, or the **ALSA** framework into **Pipewire**.

Benefits of **Pipewire** over the previous solutions include:

- A unified solution for consumer and professional users
- A flexible, modular architecture
- High performance and low latency, similar to the **JACK** service

- Isolation between audio clients for better security

You no longer have to configure the **JACK** service for applications that use it. All **JACK** applications now work in the default RHEL configuration.

PulseAudio is still available in RHEL, and you can enable it instead of **PipeWire**. For details, see Switching from PipeWire to PulseAudio .

(JIRA:RHELPLAN-101241)

Power profiles are available in GNOME

You can now switch between several power profiles in the **Power** panel of **Settings** in the GNOME environment. The power profiles optimize various system settings for the selected goal.

The following power profiles are available:

Performance

Optimizes for high system performance and reduces battery life. This profile is only available on certain selected system configurations.

Balanced

Provides standard system performance and power consumption. This is the default profile.

Power Saver

Increases battery life and reduces system performance. This profile activates automatically on low battery.

Your power profile configuration persists across system reboots.

The power profiles functionality is available from the **power-profiles-daemon** package, which is installed by default.

(JIRA:RHELPLAN-101242)

Language support is now provided by langpacks

Support for various languages is now available from **langpacks** packages. You can customize the level of language support that you want to install using the following package names, where ***code*** is the short ISO code for the language, such as **es** for Spanish:

langpacks-core-*code*

Provides a basic language support, including:

- The **glibc** locale
- The default font
- The default input method if the language requires it

langpacks-core-font-*code*

Provides only the default font for the language.

langpacks-*code*

Provides the complete language support, including the following in addition to the basic language support:

- Translations

- Spell checker dictionaries
- Additional fonts

(JIRA:RHELPLAN-101247)

Lightweight, single-application environment

For graphical use cases that only present a single application, a lightweight user interface (UI) is now available.

You can start GNOME in a single-application session, also known as kiosk mode. In this session, GNOME displays only a full-screen window of an application that you have configured.

The single-application session is significantly less resource intensive than the standard GNOME session.

For more information, see Restricting the session to a single application .

(JIRA:RHELPLAN-102552)

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#2031186)

The default wallpaper adds a Red Hat logo

The default RHEL wallpaper now displays a Red Hat logo. The logo is located in the upper left corner of the screen.

To disable the logo, disable the **Background Logo** GNOME Shell extension.

(BZ#2057150)

Firefox now uses stronger encryption in PKCS#12 files

The Firefox web browser uses PKCS#12 files to establish client authentication certificates. Previously, Firefox encrypted these files using legacy algorithms:

- PBE-SHA1-RC2-40 to encrypt the certificate in the PKCS#12 file
- PBE-SHA1-3DES to encrypt the key in the PKCS#12 file

With this release, Firefox encrypts the files using stronger algorithms by default:

- AES-256-CBC with PBKDF2 to encrypt the certificate in the PKCS#12 file
- AES-128-CBC with PBKDF2 to encrypt the key in the PKCS#12 file

With this change, the PKCS#12 files are now compatible with the Federal Information Processing Standard (FIPS).

The legacy encryption algorithms remain supported in Firefox as a non-default option.

(BZ#1764205)

4.17. GRAPHICS INFRASTRUCTURES

The Wayland session is now the default with NVIDIA drivers

When using the NVIDIA drivers, the desktop session now selects the Wayland display protocol by default, if the driver configuration supports Wayland. In previous RHEL releases, the NVIDIA drivers always disabled Wayland.

To enable Wayland with the NVIDIA drivers on your system, add the following options to the kernel command line:

- **nvidia-drm.modeset=1**
- **NVreg_PreserveVideoMemoryAllocations=1**

Note that Wayland has been the default display protocol with other graphics drivers since RHEL 8.0.

Currently, the Wayland session with the NVIDIA drivers is still incomplete and presents certain known issues. Red Hat is actively working with NVIDIA to address these gaps and problems across the GPU stack.

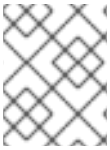
For some of the limitations of Wayland with the NVIDIA drivers, see the *Known issues* section.

(JIRA:RHELPLAN-119000)

4.18. 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)

Kernel security patches without reboot in the web console

This web console update allows users to apply kernel security patches without forcing reboots by using the **kpatch** framework. Administrators can also automatically subscribe any future kernel to the live patching stream.

(JIRA:RHELPLAN-95056)

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)

4.19. RED HAT ENTERPRISE LINUX SYSTEM ROLES

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#1993304)

The Networking system role now supports owe

The Networking RHEL system role now supports Opportunistic Wireless Encryption (owe). **owe** is a wireless authentication key management type that uses encryption between Wi-Fi clients and access points, and protects Wi-Fi clients from sniffing attacks. To use owe, set the wireless authentication key management type,**key_mgmt** field, to **owe**.

(BZ#1993377)

The Firewall system role now supports setting the firewall default zone

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#2022461)

The Storage RHEL system role now supports LVM VDO volumes

With this enhancement, you can use the Storage system role to manage Logical Manager Volumes (LVM) Virtual Data Optimizer (VDO) volumes. The LVM filesystem manages VDO volumes and with this feature, it is now possible to compress and deduplicate on LVM volumes. As a result, VDO helps to optimize the usage of the storage volumes.

(BZ#1978488)

Support for volume sizes expressed as a percentage is available in the Storage system role

This enhancement adds support to the Storage RHEL system role to express LVM volume sizes as a percentage of the pool's total size. You can specify the size of LVM volumes as a percentage of the pool/VG size, for example: 50% in addition to the human-readable size of the file system, for example, 10g, 50 GiB.

(BZ#1984583)

Support for cached volumes is available in the Storage system role

This enhancement adds support to the Storage RHEL system role to 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#2016517)

Ability to add or remove sources to the Firewall role

This update enables you to add or remove sources in the firewall settings configuration using the **source** parameter.

(BZ#2021667)

New Ansible Role for Microsoft SQL Server Management

The new **microsoft.sql.server** role is designed to help IT and database administrators automate processes involved with setup, configuration, and performance tuning of SQL Server on Red Hat Enterprise Linux.

(BZ#2013853)

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 the **microsoft.sql.server** role. This update fixes it by providing the **mssql_rpm_key**, **mssql_server_repository**, and **mssql_client_repository** variables that you can use to customize the repositories to download packages from. If no URL is provided, the **mssql** role uses the official Microsoft servers to download RPMs.

(BZ#2064648)

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

The MSSQL role generates the **/var/opt/mssql/mssql.conf** configuration file. With this update, the MSSQL 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#2064690)

Ansible Core support for the RHEL system roles

As of the RHEL 9 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 provided on previous

versions of RHEL 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 AppStream.

If you require Ansible Engine support, or otherwise need support for non-RHEL automation use cases, create a Case at Red Hat Support .

(JIRA:RHELPLAN-103540)

Support for configuring multiple elasticsearch hosts in one elasticsearch output dictionary

Previously, the **server_host** parameter used to take a string value for a single host. This enhancement adjusts it to the underlying **rsyslog omelasticsearch's** specification, so it now also takes a list of strings to support multiple hosts. Consequently, it is adjusted to hosts, following the underlying **rsyslog omelasticsearch's** specification. As a result, users can configure multiple **elasticsearch** hosts in one **elasticsearch** output dictionary.

(BZ#1986460)

RHEL system roles now support VPN management

Previously, it was difficult to set up secure and properly configured IPsec tunneling and virtual private networking (VPN) solutions on Linux. With this enhancement, you can use the VPN RHEL system role to set up and configure VPN tunnels for host-to-host and mesh connections more easily across large numbers of hosts. As a result, you have a consistent and stable configuration interface for VPN and IPsec tunneling configuration within the RHEL system roles project.

(BZ#2019341)

The SSHD RHEL system role now supports non-exclusive configuration snippets

With this feature, you can configure SSHD through different roles and playbooks without rewriting the previous configurations by using namespaces. Namespaces are similar to a drop-in directory, and define non-exclusive configuration snippets for SSHD. As a result, you can use the SSHD RHEL system role from a different role, if you need to configure only a small part of the configuration and not the entire configuration file.

(BZ#1978752)

Network Time Security (NTS) option added to the timesync RHEL system role

The **NTS** option was added to the Timesync RHEL system role to enable **NTS** on client servers. NTS is a new security mechanism specified for Network Time Protocol (NTP). NTS can secure synchronization of NTP clients without client-specific configuration and can scale to large numbers of clients. The **NTS** option is supported only with the **chrony** NTP provider in version 4.0 and later.

(BZ#1978753)

Support for HA Cluster RHEL system role

The High Availability Cluster (HA Cluster) role 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#2054401)

Support for Rsyslog username and password authentication to Elasticsearch

This update adds the Elasticsearch username and password parameters to the Logging system role. As a result, you can enable Rsyslog to authenticate to Elasticsearch using a username and password.

(BZ#1990490)

The NBDE Client system role supports static IP addresses

In previous versions of RHEL, restarting a system with a static IP address and configured with the Network Bound Disk Encryption (NBDE) Client system role would change the system's IP address. With this change, systems with static IP addresses are supported by the NBDE Client system role, and their IP addresses do not change after a reboot.

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

(BZ#2031555)

Support for specifying raid_level for LVM has been added

RHEL 9.0 supports grouping Logical Volume Management (LVM) volumes into RAIDs using the **lvmraid** feature.

(BZ#2016518)

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)

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#1978773)

Support to all bonding options added to the network system role

This update provides support to all bonding options to the **network** 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#2054435)

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#1999162)

The Network system role now directly manages the configuration files of Ansible

With this enhancement, the **network** role generates **ifcfg** files in **/etc/sysconfig/network-scripts**. Then, it inserts the comment “Ansible managed”, using the standard **ansible_managed** variable. This comment indicates that the **ifcfg** files are not directly editable as the **network** role may overwrite it. The important difference in handling the **ifcfg** file to add "Ansible managed" comment is that the **network** role uses the **initscripts** package while the NetworkManager uses the **nm** package.

(BZ#2057657)

Ansible Core support for RHEL system roles

In RHEL 9.0, 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.

(BZ#2012298)

The Cockpit 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#2021028)

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#2054367)

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#2054369)

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

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

(BZ#2057662)

The Firewall RHEL system role has been added in RHEL 9

With this enhancement, 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#2021665)

The SSH client RHEL system role now supports new configuration options in OpenSSH 8.7

With this enhancement, OpenSSH was updated to the latest version, which provides new configuration options that are available in the SSH client role for configuring new hosts.

(BZ#2029427)

4.20. VIRTUALIZATION

RHEL web console new virtualization features

With this update, the RHEL web console includes new features in the Virtual Machines page. You can now:

- Rename a VM
- Create a VM with cloud image authentication
- Add and remove USB and PCI devices to the VM
- Specify network interface model
- Share and unshare files between a host and its VM

(JIRA:RHELPLAN-102009)

QEMU uses Clang

The QEMU emulator is now built using the Clang compiler. This enables the RHEL 9 KVM hypervisor to use a number of advanced security and debugging features, and makes future feature development more efficient.

(BZ#1940132)

SafeStack for virtual machines

In RHEL 9 on AMD64 and Intel 64 hardware (x86_64), the QEMU emulator can use SafeStack, an enhanced compiler-based stack protection feature. SafeStack reduces the ability of an attacker to exploit a stack- based buffer overflow to change return pointers in the stack and create Return-Oriented Programming (ROP) attacks. As a result, virtual machines hosted on RHEL 9 are significantly more secure against ROP-based vulnerabilities.

(BZ#1939509)

virtiofs full support on Intel 64, AMD64, and IBM Z

The virtio file system (**virtiofs**) is now fully supported on Intel 64, AMD64, and IBM Z architectures. Using **virtiofs**, you can efficiently share files between your host system and its virtual machines.

(JIRA:RHELPLAN-64576)

AMD EPYC 7003 series processors supported on KVM guests

Support for AMD EPYC 7003 series processors (also known as **AMD Milan**) has now been added to the KVM hypervisor and kernel code, and to the libvirt API. This enables KVM virtual machines to use AMD EPYC 7003 series processors.

(JIRA:RHELPLAN-65223)

qemu-kvm now supports additional machine types

A set of new machine types, based on RHEL 9, has been added for use by virtual machines (VMs). To obtain all currently supported machine types on your host, use the **/usr/libexec/qemu-kvm -M help** command.

In addition, all machine types based on RHEL 7.5.0 or earlier are now unsupported. These also include **pc-i440fx-rhel7.5.0** and earlier machine types, which were default in earlier major versions of RHEL. As a consequence, attempting to start a VM with such machine types on RHEL 9 fails with an **unsupported configuration** error. If you encounter this problem after upgrading your host to RHEL 9, see the Red Hat KnowledgeBase.

(JIRA:RHELPLAN-75866)

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 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 9 guide.

(BZ#1995131)

Modular libvirt daemons

In RHEL 9, the **libvirt** library uses modular daemons that handle individual virtualization driver sets on your host. For example, the **virtqemud** daemon handles QEMU drivers. This makes it possible to fine-grain a variety of tasks that involve virtualization drivers, such as resource load optimization and monitoring.

In addition, the monolithic libvirt daemon, **libvirtd**, has become deprecated. However, if you upgrade from RHEL 8 to RHEL 9, your host will still use **libvirtd**, which you can continue using in RHEL 9. Nevertheless, Red Hat recommends switching to modular **libvirt** daemons instead.

(JIRA:RHELPLAN-113994)

Windows 11 and Windows Server 2022 guests are supported

RHEL 9 supports using Windows 11 and Windows Server 2022 as the guest operating systems on KVM virtual machines.

(BZ#2036856, BZ#2004161)

ksmtuned is now distributed separately from qemu-kvm

To decrease the footprint of the KVM hypervisor, the **ksmtuned** utility is no longer a dependency of **qemu-kvm**. As a consequence, if you require configuring kernel same-page merging (KSM), you must install the **ksmtuned** package manually.

(BZ#2069501, BZ#1971678, BZ#1972158)

New feature: vTPM

The Virtual Trusted Platform Module (vTPM) is fully supported in RHEL 9. Using vTPM, you can add a TPM virtual crypto-processor to a virtual machine (VM) running in the RHEL 9 KVM hypervisor. This makes it possible to use the VM for generating, storing, and managing cryptographic keys.

(JIRA:RHELPLAN-98617)

Virtualization support for Intel Atom P59 series processors

With this update, virtualization on RHEL 9 adds support for the Intel Atom P59 series processors, formerly known as Snow Ridge. As a result, virtual machines hosted on RHEL 9 can now use the **Snowridge** CPU model and utilise new features that the processors provide.

(BZ#1874187)

4.21. RHEL IN CLOUD ENVIRONMENTS

RHEL 9 provides WALinuxAgent 2.3.0.2

RHEL 9 is distributed with the Windows Azure Linux Agent (**WALinuxAgent**) package version 2.3.0.2. Notable bug fixes and enhancements over version 2.2.49 include:

- Support for RequiredFeatures and GoalStateAggregateStatus APIs has been added.
- Fallback locations for extension manifests have been added.
- Missing calls to str.format() have been added when creating exceptions.

(BZ#1972101)

RHEL on Azure now supports MANA

RHEL 9 virtual machines running on Microsoft Azure can now use the Microsoft Azure Network Adapter (MANA).

(BZ#1957818)

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 9 virtual machines on VMware vSphere is now more efficient and reliable.

(BZ#2040090)

RHEL 9 virtual machines are now supported on certain ARM64 hosts on Azure

Virtual machines that use RHEL 9 as the guest operating system are now supported on Microsoft Azure hypervisors running on Ampere Altra ARM-based processors.

(BZ#1949613)

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 9 virtual machine on Azure.

(BZ#2042351)

New SSH module for cloud-init

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

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

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

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

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

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

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

4. Restart the sshd service:

```
systemctl restart sshd
```

(BZ#2115791)

4.22. SUPPORTABILITY

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#2011537)

4.23. CONTAINERS

Podman now supports secure short names

Short-name aliases for images can now be configured in the **registries.conf** file in the **[aliases]** table. The short-names modes are:

- Enforcing: If no matching alias is found during the image pull, Podman prompts the user to choose one of the unqualified-search registries. If the selected image is pulled successfully, Podman automatically records a new short-name alias in the **$HOME/.cache/containers/short-name-aliases.conf** file (rootless user) and in the **/var/cache/containers/short-name-aliases.conf** (root user). If the user cannot be prompted (for example, stdin or stdout are not a TTY), Podman fails. Note that the **short-name-aliases.conf** file has precedence over **registries.conf** file if both specify the same alias.

- Permissive: Similar to enforcing mode, but Podman does not fail if the user cannot be prompted. Instead, Podman searches in all unqualified-search registries in the given order. Note that no alias is recorded.

Example:

```
unqualified-search-registries=["registry.fedoraproject.org", "quay.io"]

[aliases]

"fedora"="registry.fedoraproject.org/fedora"
```

(JIRA:RHELPLAN-74542)

Changes in the container-tools module

The **container-tools** module contains the Podman, Buildah, Skopeo, and runc tools. The rolling stream, represented by the **container-tools:rhel8** stream in RHEL 8, is named **container-tools:latest** in RHEL 9. Similarly to RHEL 8, stable versions of container tools are going to be available in numbered streams (for example, 3.0).

For more information about the Container Tools Application Stream, see Container Tools AppStream - Content Availability.

(JIRA:RHELPLAN-73678)

The containers-common package is now available

The **containers-common** package has been added to the **container-tools:latest** module. The **containers-common** package contains common configuration files and documentation for the container tools ecosystem, such as Podman, Buildah and Skopeo.

(JIRA:RHELPLAN-77549)

Updating container images with new packages

For instance, to update the **registry.access.redhat.com/rhel9** container image with the latest packages, use the following commands:

```
# podman run -it registry.access.redhat.com/rhel9
# dnf update -y && rm -rf /var/cache/dnf
```

To install a particular ***<package>*** enter:

```
# dnf install <package>
```

For more information, see Adding software to a running UBI container .

Note that for RHEL 9, updating or installing new packages in the image requires that you are running on an entitled host. You can use the Red Hat Enterprise Linux Developer Subscription for Individuals to gain access to entitled repositories at no-cost.

For more information, see No-cost Red Hat Enterprise Linux Individual Developer Subscription: FAQs .

(JIRA:RHELPLAN-84168)

The container-tools meta-package has been updated

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

(JIRA:RHELPLAN-118914)

The podman-py package is now available

The **podman-py** package has been added to the **container-tools:3.0** stable module stream and the **container-tools:latest** module. The **podman-py** package is a library of bindings to use the RESTful API of Podman.

(BZ#1975462)

Control groups version 2 is now available

The previous version of control groups, cgroups version 1 (cgroups v1) caused performance problems with a variety of applications. The latest release of control groups, cgroups version 2 (cgroups v2) enables system administrators to limit resources for any application without causing performance problems.

This new version of control groups, cgroups v2, can be enabled in RHEL 8 and is enabled by default in RHEL 9.

(JIRA:RHELPLAN-73697)

The container-tools meta-package is now available

The **container-tools** RPM meta-package includes Podman, Buildah, Skopeo, CRIU, Udica, and all required libraries, is available in RHEL 9. The stable streams are not available on RHEL 9. To receive stable access to Podman, Buildah, Skopeo, and others, use the RHEL EUS subscription.

To install the **container-tools** meta-package, enter:

```
# dnf install container-tools
```

(BZ#2000871)

Native overlay file system support in the kernel is now available

The overlay file system support is now available from kernel 5.11. The non-root users will have native overlay performance even when running rootless (as a user). Thus, this enhancement provides better performance to non-root users who wish to use overlayfs without the need for bind mounting.

(JIRA:RHELPLAN-99892)

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-74543)

The container-tools meta-package has been updated

The **container-tools** meta-package includes Podman, Buildah, Skopeo, CRIU, Udica, and all required libraries. 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 are not usable in Podman v4.0
- Native overlay file system is usable as a rootless user
- NFS storage is now supported within a container
- Control groups version 2 (cgroup v2) is enabled by default
- Downgrading from Podman v4 to v3 is not supported unless all containers are destroyed and recreated

For further information about notable changes in Podman, see the upstream release notes.

(JIRA:RHELPLAN-99889)

The crun container runtime is now the default

The **crun** container runtime is now the default runtime. The **crun** container runtime supports an annotation that allows the container to access the rootless user's additional groups. This is useful for volume mounting in a directory where setgid is set, or where the user only has group access. Both the **crun** and **runc** runtimes fully support **cgroup v2**.

(JIRA:RHELPLAN-99890)

Control group version 2 is now available

The previous version of control groups, cgroup version 1 (cgroup v1) caused performance problems with a variety of applications. The latest release of control groups, cgroup version 2 (cgroup v2) enables system administrators to limit resources for any application without causing performance problems.

In RHEL 9, cgroup v2 is enabled by default.

(JIRA:RHELPLAN-75322)

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-100032)

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/rhel9/openssl
- registry.access.redhat.com/ubi9/openssl

(JIRA:RHELPLAN-100034)

Netavark network stack is now available

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

This network stack has the following capabilities:

- 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

(JIRA:RHELPLAN-101141)

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 the section on Managing subID ranges manually in IdM documentation.

(JIRA:RHELPLAN-100020)

CHAPTER 5. BUG FIXES

This part describes bugs fixed in Red Hat Enterprise Linux 9.0 that have a significant impact on users.

5.1. INSTALLER AND IMAGE CREATION

--leavebootorder no longer changes boot order

Previously, using **--leavebootorder** for the bootloader kickstart command did not work correctly on UEFI systems and changed the boot order. This caused the installer to add RHEL at the top of the list of installed systems in the UEFI boot menu.

This update fixes the problem and using **--leavebootorder** no longer changes the boot order in the boot loader. **--leavebootorder** is now supported on RHEL for UEFI systems.

(BZ#2025953)

Anaconda sets a static hostname before running the %post scripts

Previously, when Anaconda was setting the installer environment host name to the value from the kickstart configuration (**network --hostname**), it used to set a transient hostname. Some of the actions performed during **%post** script run, for example network device activation, were causing the host name reset to a value obtained by reverse **dns**.

With this update, Anaconda now sets a static hostname of the installer environment to be stable during the run of kickstart **%post** scripts.

(BZ#2009403)

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 edge commit for the upgrade, such as adding a rpm package, would cause users to be locked out of a system, after an upgrade is applied. It caused users to 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, which creates a user on the system at installation time, rather than having the user as part of the **ostree** commit.

(BZ#2060575)

The basic graphics mode has been removed from the boot menu

Previously, the **basic graphics** mode was used to install RHEL on hardware with an unsupported graphics card or to work around issues in graphic drivers that prevented starting the graphical interface. With this update, the option to install in a **basic graphics** mode has been removed from the installer boot menu. Use the VNC installation options for graphical installations on unsupported hardware or to work around driver bugs.

For more information on installations using VNC, see the Performing a remote RHEL installation using VNC section.

(BZ#1961092)

5.2. SUBSCRIPTION MANAGEMENT

virt-who now works correctly with Hyper-V hosts

Previously, when using **virt-who** to set up RHEL 9 virtual machines (VMs) on a Hyper-V hypervisor, **virt-who** did not properly communicate with the hypervisor, and the setup failed. This was because of a deprecated encryption method in the **openssl** package.

With this update, the **virt-who** authentication mode for Hyper-V has been modified, and setting up RHEL 9 VMs on Hyper-V using **virt-who** now works correctly. Note that this also requires the hypervisor to use basic authentication mode. To enable this mode, use the following commands:

```
winrm set winrm/config/service/auth '@{Basic="true"}'
winrm set winrm/config/service '@{AllowUnencrypted="true"}'
```

(BZ#2008215)

5.3. 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#2055032)

5.4. SHELLS AND COMMAND-LINE TOOLS

RHEL 9 provides libservicelog 1.1.19

RHEL 9 is distributed with **libservicelog** version 1.1.19. Notable bug fixes include:

- Fixed output alignment issue.
- Fixed **segfault** on **servicelog_open()** failure.

(BZ#1869568)

5.5. SECURITY

Hardware optimization enabled in libgcrypt when in the FIPS mode

Previously, the Federal Information Processing Standard (FIPS 140-2) did not allow using hardware optimization. Therefore, in previous versions of RHEL, the operation was disabled in the **libgcrypt** package when in the FIPS mode. RHEL 9 enables hardware optimization in FIPS mode, and as a result, all cryptographic operations are performed faster.

(BZ#1990059)

crypto-policies now can disable ChaCha20 cipher usage

Previously, the **crypto-policies** package used a wrong keyword to disable the **ChaCha20** cipher in OpenSSL. Consequently, you could not disable **ChaCha20** for the TLS 1.2 protocol in OpenSSL through **crypto-policies**. With this update, the **-CHACHA20** keyword is used instead of **-CHACHA20-POLY1305**. As a result, you now can use the cryptographic policies for disabling **ChaCha20** cipher usage in OpenSSL for TLS 1.2 and TLS 1.3.

(BZ#2004207)

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#2013195)

GNUTLS_NO_EXPLICIT_INIT no longer disables implicit library initialization

Previously, the **GNUTLS_NO_EXPLICIT_INIT** environment variable disabled implicit library initialization. In RHEL 9, the **GNUTLS_NO_IMPLICIT_INIT** variable disables implicit library initialization instead.

(BZ#1999639)

OpenSSL-based applications now work correctly with the Turkish locale

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

(BZ#2071631)

kdump no longer crashes due to SELinux permissions

The **kdump** crash recovery service requires additional SELinux permissions to start correctly. In previous versions, therefore, SELinux prevented **kdump** from working, **kdump** reported that it is not operational, and Access Vector Cache (AVC) denials were audited. In this version, the required permissions were added to **selinux-policy** and as a result, **kdump** works correctly and no AVC denial is audited.

(BZ#1932752)

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, **dnf** can install **usbguard** correctly.

(BZ#1986785)

dnf install and dnf update now work with fapolicyd in SELinux

The **fapolicyd-selinux** package, which contains SELinux rules for fapolicyd, did not contain permissions to watch all files and directories. As a consequence, the **fapolicyd-dnf-plugin** did not work correctly, causing any **dnf install** and **dnf update** commands to make the system stop responding indefinitely. In

this version, the permissions to watch any file type were added to **fapolicyd-selinux**. As a result, the **fapolicyd-dnf-plugin** works correctly and the commands **dnf install** and **dnf update** are operational.

(BZ#1932225)

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#2037215)

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#2009226)

5.6. NETWORKING

Wifi and 802.1x Ethernet connections profiles are now connecting properly

Previously, many Wifi and 802.1x Ethernet connections profiles were not able to connect. This bug is now fixed. All the profiles are now connecting properly. Profiles that use legacy cryptographic algorithms still work but you need to manually enable the OpenSSL legacy provider. This is required, for example, when you use DES with MS-CHAPv2 and RC4 with TKIP.

(BZ#1975718)

Afterburn no longer sets an overlong hostname in /etc/hostname

The maximum length of a RHEL hostname is 64 characters. However, certain cloud providers use the Fully-Qualified Domain Name (FQDN) as the hostname, which can be up to 255 characters. Previously, the **afterburn-hostname** service wrote such an overlong hostname directly to the **/etc/hostname** file. The **systemd** service truncated the hostname to 64 characters, and NetworkManager derived an

incorrect DNS search domain from the truncated value. With this fix, **afterburn-hostname** truncates hostnames at the first dot or 64 characters, whichever comes first. As a result, NetworkManager no longer sets invalid DNS search domains in **/etc/resolv.conf**.

(BZ#2008521)

5.7. KERNEL

modprobe loads out-of-tree kernel modules as expected

The **/etc/depmod.d/dist.conf** configuration file provides a search order for the **depmod** utility. Based on the search order, **depmod** creates the **modules.dep.bin** file. This file lists module dependencies, which the **modprobe** utility uses for loading and unloading kernel modules and resolving module dependencies at the same time. Previously, **/etc/depmod.d/dist.conf** was missing. As a result, **modprobe** could not load some out-of-tree kernel modules. This update includes the **/etc/depmod.d/dist.conf** configuration file, which fixes the search order. As a result, **modprobe** loads out-of-tree kernel modules as expected.

(BZ#1985100)

alsa-lib now correctly handles audio devices that use UCM

A bug in the **alsa-lib** package caused incorrect parsing of the internal Use Case Manager (UCM) identifier. Consequently, some audio devices that used the UCM configuration were not detected or they did not function correctly. The problem occurred more often when the system used the **pipewire** sound service. With the new release of RHEL 9, the problem has been fixed by updating the **alsa-lib** library.

(BZ#2015863)

5.8. FILE SYSTEMS AND STORAGE

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#2017979)

device-mapper-multipath rebased to version 0.8.7

The **device-mapper-multipath** package has been upgraded to version 0.8.7, which provides multiple bug fixes and enhancements. Notable changes include:

- Fixed memory leaks in the **multipath** and **kpartx** commands.
- Fixed repeated trigger errors from the **multipathd.socket** unit file.
- Improved autoconfiguration of more devices, such as DELL SC Series arrays, EMC Invista and Symmetrix arrays (among others).

(BZ#2017592)

5.9. HIGH AVAILABILITY AND CLUSTERS

Pacemaker attribute manager correctly determines remote node attributes, preventing unfencing loops

Previously, Pacemaker's controller on a node might be elected the Designated Controller (DC) before its attribute manager learned an already-active remote node is remote. When this occurred, the node's scheduler would not see any of the remote node's node attributes. If the cluster used unfencing, this could result in an unfencing loop. With the fix, the attribute manager can now learn a remote node is remote by means of additional events, including the initial attribute sync at start-up. As a result, no unfencing loop occurs, regardless of which node is elected DC.

(BZ#1975388)

5.10. COMPILERS AND DEVELOPMENT TOOLS

-Wsequence-point warning behavior fixed

Previously, when compiling C++ programs with GCC, the **-Wsequence-point** warning option tried to warn about very long expressions, it could cause quadratic behavior and therefore significantly longer compilation time. With this update, **-Wsequence-point** doesn't attempt to warn about extremely large expressions and as a result, does not increase compilation time.

(BZ#1481850)

5.11. IDENTITY MANAGEMENT

MS-CHAP authentication with the OpenSSL legacy provider

Previously, FreeRADIUS authentication mechanisms that used MS-CHAP failed because they depended on MD4 hash functions, and MD4 has been deprecated in RHEL 9. With this update, you can authenticate FreeRADIUS users with MS-CHAP or MS-CHAPv2 if you enable the OpenSSL legacy provider.

If you use the default OpenSSL provider, MS-CHAP and MS-CHAPv2 authentication fails and the following error message is displayed, indicating the fix:

```
Couldn't init MD4 algorithm. Enable OpenSSL legacy provider.
```

(BZ#1978216)

Running sudo commands no longer exports the KRB5CCNAME environment variable

Previously, after running **sudo** commands, the environment variable **KRB5CCNAME** pointed to the Kerberos credential cache of the original user, which might not be accessible to the target user. As a result Kerberos related operations might fail as this cache is not accessible. With this update, running **sudo** commands no longer sets the **KRB5CCNAME** environment variable and the target user can use their default Kerberos credential cache.

(BZ#1879869)

SSSD correctly evaluates the default setting for the Kerberos keytab name in /etc/krb5.conf

Previously, if you defined a non-standard location for your **krb5.keytab** file, SSSD did not use this location and used the default **/etc/krb5.keytab** location instead. As a result, when you tried to log into the system, the login failed as the **/etc/krb5.keytab** contained no entries.

With this update, SSSD now evaluates the **default_keytab_name** variable in the **/etc/krb5.conf** and uses the location specified by this variable. SSSD only uses the default **/etc/krb5.keytab** location if the **default_keytab_name** variable is not set.

(BZ#1737489)

Authenticating to Directory Server in FIPS mode with passwords hashed with the PBKDF2 algorithm 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, prior to this update, users with a password hashed with the password-based key derivation function 2 (PBKDF2) algorithm were not able to 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, authentication with passwords hashed with the PBKDF2 algorithm now works as expected.

(BZ#1779685)

5.12. RED HAT ENTERPRISE LINUX SYSTEM ROLES

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#2004899)

Postfix role README no longer uses plain role name

Previously, the examples provided in the **/usr/share/ansible/roles/rhel-system-roles.postfix/README.md** used the plain version of the role name, **postfix**, instead of using **rhel-system-roles.postfix**. Consequently, users would consult the documentation and incorrectly use the plain role name instead of Full Qualified Role Name (FQRN). This update fixes the issue, and the documentation contains examples with the FQRN, **rhel-system-roles.postfix**, enabling users to correctly write playbooks.

(BZ#1958964)

Postfix RHEL system role README.md no longer missing variables under the "Role Variables" section

Previously, the Postfix RHEL system role variables, such as **postfix_check**, **postfix_backup**, **postfix_backup_multiple** were not available under the "Role Variables" section. Consequently, users were not able to consult the Postfix role documentation. This update adds role variable documentation

to the Postfix README section. The role variables are documented and available for users in the **doc/usr/share/doc/rhel-system-roles/postfix/README.md** documentation provided by **rhel-system-roles**.

(BZ#1978734)

Role tasks no longer change when running the same output

Previously, several of the role tasks would report as **CHANGED** when running the same input once again, even if there were no changes. Consequently, the role was not acting idempotent. To fix the issue, perform the following actions:

- Check if configuration variables change before applying them. You can use the option **--check** for this verification.
- Do not add a **Last Modified: $date** header to the configuration file.

As a result, the role tasks are idempotent.

(BZ#1978760)

The logging_purge_confs option correctly deletes unnecessary configuration files

With the **logging_purge_confs** option set to **true**, it should delete unnecessary logging configuration files. Previously, however, unnecessary configuration files were not deleted from the configuration directory even if **logging_purge_confs** was set to **true**. This issue is now fixed and the option has been redefined as follows: if **logging_purge_confs** is set to **true**, Rsyslog removes files from the **rsyslog.d** directory which do not belong to any rpm packages. This includes configuration files generated by previous runs of the Logging role. The default value of **logging_purge_confs** is **false**.

(BZ#2039106)

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#2041632)

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#2058777)

The timesync role no longer fails to find the requested service ptp4l

Previously, on some versions of RHEL, the Ansible **service_facts** module, reported service facts incorrectly. Consequently, the **timesync** role reported an error attempting to stop the **ptp4l** service. With this fix, the Ansible **service_facts** module checks the return value of the tasks to stop **timesync** services. If the returned value is **failed**, but the error message is **Could not find the requested service NAME:**, then the module assumes success. As a result, the **timesync** role now runs without errors like **Could not find the requested service ptp4l**.

(BZ#2058645)

The kernel_settings configobj is available on managed hosts

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

(BZ#2058756)

The Terminal Session Recording role tlog-rec-session is now correctly overlaid by SSSD

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

(BZ#2071804)

The SSHD system role can manage systems in FIPS mode

Previously, the SSHD system role could not create the **not allowed** HostKey type when called. As a consequence, the SSHD system role could not manage RHEL 8 and older systems in Federal Information Processing Standard (FIPS) mode. With this update, the SSHD system role detects FIPS mode and adjusts the default HostKey list correctly. As a result, the system role can manage RHEL systems in FIPS mode with the default HostKey configuration.

(BZ#2029634)

The SSHD system role uses the correct template file

Previously, the SSHD system role used a wrong template file. As a consequence, the generated **sshd_config** file did not contain the **ansible_managed** comment. With this update, the system role uses the correct template file and **sshd_config** contains the correct **ansible_managed** comment.

(BZ#2044408)

The Kdump RHEL system role is be able to reboot, or indicate that a reboot is required

Previously, the Kdump RHEL system role ignored managed nodes without any reserved memory for crash kernel. Consequently, the role finished with the "Success" status, even if it did not configure the system properly. With this update of RHEL 9, the problem has been fixed. In cases when managed nodes do not have any memory reserved for the crash kernel, the Kdump RHEL system role fails and suggests that users set the **kdump_reboot_ok** variable to **true** to properly configure the **kdump** service on managed nodes.

(BZ#2029602)

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#2038957)

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#2064391)

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#2004303)

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#2006230)

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#2057164)

The group option in the Certificate system role 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#2021025)

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#2021676)

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#2047506)

The VPN system role filter plugin vpn_ipaddr now converts to FQCN (Fully Qualified Collection Name)

Previously, the conversion from the legacy role format to the collection format was not converting the filter plugin **vpn_ipaddr** to FQCN (Fully Qualified Collection Name) **redhat.rhel_system_roles.vpn_ipaddr**. As a consequence, the VPN role could not find the plugin by the short name and reported an error. With this fix, the conversion script has been changed so that the filter is converted to FQCN format in the collection. And now the VPN role runs without issuing the error.

(BZ#2050341)

Job for kdump.service no longer fails

Previously, the Kdump role code for configuring the kernel crash size was not updated for RHEL9, which requires the use of **kdumpctl reset-crashkernel**. As a consequence, the **kdump.service** could not start and issued an error. With this update, the **kdump.service** role uses **kdumpctl reset-crashkernel** to configure the crash kernel size. Now, **kdump.service** role successfully starts the kdump service and the kernel crash size is configured correctly.

(BZ#2050419)

5.13. VIRTUALIZATION

Hot-unplugging a mounted virtual disk no longer causes the guest kernel to crash on IBM Z

Previously, when detaching a mounted disk from a running virtual machine (VM) on IBM Z hardware, the VM kernel crashed under the following conditions:

- The disk was attached with target bus type **scsi** and mounted inside the guest.
- After hot-unplugging the disk device, the corresponding SCSI controller was hot-unplugged as well.

With this update, the underlying code has been fixed and the described crash no longer occurs.

(BZ#1997541)

5.14. CONTAINERS

UBI 9-Beta containers can run on RHEL 7 and 8 hosts

Previously, the UBI 9-Beta container images had an incorrect seccomp profile set in the **containers-common** package. As a consequence, containers were not able to deal with certain system calls causing a failure. With this update, the problem has been fixed.

(BZ#2019901)

CHAPTER 6. TECHNOLOGY PREVIEWS

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

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

6.1. 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)

6.2. 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#2046653)

GIMP available as a Technology Preview in RHEL 9

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

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

(BZ#2047161)

6.3. NETWORKING

WireGuard VPN is available as a Technology Preview

WireGuard, which Red Hat provides as an unsupported Technology Preview, is a high-performance VPN solution that runs in the Linux kernel. It uses modern cryptography and is easier to configure than other VPN solutions. Additionally, the small code-basis of WireGuard reduces the surface for attacks and, therefore, improves the security.

For further details, see Setting up a WireGuard VPN.

(BZ#1613522)

KTLS available as a Technology Preview

RHEL provides Kernel Transport Layer Security (KTLS) as a Technology Preview. KTLS handles TLS records using the symmetric encryption or decryption algorithms in the kernel for the AES-GCM cipher. KTLS also includes the interface for offloading TLS record encryption to Network Interface Controllers (NICs) that provides this functionality.

(BZ#1570255)

The **systemd-resolved** service is available as a Technology Preview

The **systemd-resolved** service provides name resolution to local applications. The service implements a caching and validating DNS stub resolver, an Link-Local Multicast Name Resolution (LLMNR), and Multicast DNS resolver and responder.

Note that **systemd-resolved** is an unsupported Technology Preview.

(BZ#2020529)

6.4. KERNEL

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

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

(BZ#2030412)

SGX available as a Technology Preview

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

(BZ#1874182)

The Soft-iWARP driver is available as a Technology Preview

Soft-iWARP (siw) is a software, Internet Wide-area RDMA Protocol (iWARP), kernel driver for Linux. Soft-iWARP implements the iWARP protocol suite over the TCP/IP network stack. This protocol suite is fully implemented in software and does not require a specific Remote Direct Memory Access (RDMA)

hardware. Soft-iWARP enables a system with a standard Ethernet adapter to connect to an iWARP adapter or to another system with already installed Soft-iWARP.

(BZ#2023416)

6.5. FILE SYSTEMS AND STORAGE

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

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

(BZ#1995338)

Stratis is available as a Technology Preview

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

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

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

Stratis is provided as a Technology Preview.

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

(BZ#2041558)

NVMe-oF Discovery Service features available as a Technology Preview

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

(BZ#2021672)

6.6. COMPILERS AND DEVELOPMENT TOOLS

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

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

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

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

(BZ#1980981)

6.7. IDENTITY MANAGEMENT

DNSSEC available as Technology Preview in IdM

Identity Management (IdM) servers with integrated DNS now implement DNS Security Extensions (DNSSEC), a set of extensions to DNS that enhance security of the DNS protocol. DNS zones hosted on IdM servers can be automatically signed using DNSSEC. The cryptographic keys are automatically generated and rotated.

Users who decide to secure their DNS zones with DNSSEC are advised to read and follow these documents:

- DNSSEC Operational Practices, Version 2
- Secure Domain Name System (DNS) Deployment Guide
- DNSSEC Key Rollover Timing Considerations

Note that IdM servers with integrated DNS use DNSSEC to validate DNS answers obtained from other DNS servers. This might affect the availability of DNS zones that are not configured in accordance with recommended naming practices.

(BZ#2084180)

Identity Management JSON-RPC API available as Technology Preview

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

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

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

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

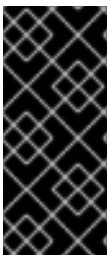
For details on using the API, see Using the Identity Management API to Communicate with the IdM Server (TECHNOLOGY PREVIEW).

(BZ#2084166)

ACME available as a Technology Preview

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

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



IMPORTANT

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

WARNING

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

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

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

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

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

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

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

(BZ#2084181)

6.8. DESKTOP

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

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

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

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

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

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

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

(JIRA:RHELPLAN-27394)

GNOME for the IBM Z architecture available as a Technology Preview

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

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

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

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

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

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

(JIRA:RHELPLAN-27737)

6.9. THE WEB CONSOLE

Stratis available as a Technology Preview in the RHEL web console

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

To learn more about Stratis, see What is Stratis.

(JIRA:RHELPLAN-122345)

6.10. VIRTUALIZATION

Creating nested virtual machines

Nested KVM virtualization is provided as a Technology Preview for KVM virtual machines (VMs) running on Intel, AMD64, and IBM Z hosts with RHEL 9. With this feature, a RHEL 7, RHEL 8, or RHEL 9 VM that runs on a physical RHEL 9 host can act as a hypervisor, and host its own VMs.

(JIRA:RHELDOCS-17040)

AMD SEV and SEV-ES for KVM virtual machines

As a Technology Preview, RHEL 9 provides the Secure Encrypted Virtualization (SEV) feature for AMD EPYC host machines that use the KVM hypervisor. If enabled on a virtual machine (VM), SEV encrypts the VM's memory to protect the VM from access by the host. This increases the security of the VM.

In addition, the enhanced Encrypted State version of SEV (SEV-ES) is also provided as Technology Preview. SEV-ES encrypts all CPU register contents when a VM stops running. This prevents the host from modifying the VM's CPU registers or reading any information from them.

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

(JIRA:RHELPLAN-65217)

Virtualization is now available on ARM 64

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

(JIRA:RHELPLAN-103993)

virtio-mem is now available on AMD64 and Intel 64

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

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

(BZ#2014487)

Intel vGPU available as a Technology Preview

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

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

(JIRA:RHELDOCS-17050)

6.11. CONTAINERS

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 7. DEPRECATED FUNCTIONALITY

This part provides an overview of functionality that has been *deprecated* in Red Hat Enterprise Linux 9.

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 9. 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 8 but has been *removed* in RHEL 9, see Considerations in adopting RHEL 9 .

7.1. INSTALLER AND IMAGE CREATION

Deprecated Kickstart commands

The following Kickstart commands have been deprecated:

- **timezone --ntpservers**
- **timezone --nontp**
- **logging --level**
- **%packages --excludeWeakdeps**
- **%packages --instLangs**
- **%anaconda**
- **pwpolicy**

Note that where only specific options are listed, the base command and its other options are still available and not deprecated. Using the deprecated commands in Kickstart files prints a warning in the logs. You can turn the deprecated command warnings into errors with the **inst.ksstrict** boot option.

(BZ#1899167)

7.2. SHELLS AND COMMAND-LINE TOOLS

Setting the TMPDIR variable in the ReaR configuration file is deprecated

Setting the **TMPDIR** environment variable in the **/etc/rear/local.conf** or **/etc/rear/site.conf** ReaR configuration file), by using a statement such as **export TMPDIR=...**, does not work and is deprecated.

To specify a custom directory for ReaR temporary files, export the variable in the shell environment before executing ReaR. For example, execute the **export TMPDIR=...** statement and then execute the **rear** command in the same shell session or script.

Jira:RHELDOCS-18049

7.3. SECURITY

SHA-1 is deprecated for cryptographic purposes

The usage of the SHA-1 message digest for cryptographic purposes has been deprecated in RHEL 9. The digest produced by SHA-1 is not considered secure because of many documented successful attacks based on finding hash collisions. The RHEL core crypto components no longer create signatures using SHA-1 by default. Applications in RHEL 9 have been updated to avoid using SHA-1 in security-relevant use cases.

Among the exceptions, the HMAC-SHA1 message authentication code and the Universal Unique Identifier (UUID) values can still be created using SHA-1 because these use cases do not currently pose security risks. SHA-1 also can be used in limited cases connected with important interoperability and compatibility concerns, such as Kerberos and WPA-2. See the List of RHEL applications using cryptography that is not compliant with FIPS 140-3 section in the RHEL 9 Security hardening document for more details.

If your scenario requires the use of SHA-1 for verifying existing or third-party cryptographic signatures, you can enable it by entering the following command:

```
# update-crypto-policies --set DEFAULT:SHA1
```

Alternatively, you can switch the system-wide crypto policies to the **LEGACY** policy. Note that **LEGACY** also enables many other algorithms that are not secure.

(JIRA:RHELPLAN-110763)

SCP is deprecated in RHEL 9

The secure copy protocol (SCP) is deprecated because it has known security vulnerabilities. The SCP API remains available for the RHEL 9 lifecycle but using it reduces system security.

- In the **scp** utility, SCP is replaced by the SSH File Transfer Protocol (SFTP) by default.
- The OpenSSH suite does not use SCP in RHEL 9.
- SCP is deprecated in the **libssh** library.

(JIRA:RHELPLAN-99136)

Digest-MD5 in SASL is deprecated

The Digest-MD5 authentication mechanism in the Simple Authentication Security Layer (SASL) framework is deprecated, and it might be removed from the **cyrus-sasl** packages in a future major release.

(BZ#1995600)

OpenSSL deprecates MD2, MD4, MDC2, Whirlpool, RIPEMD160, Blowfish, CAST, DES, IDEA, RC2, RC4, RC5, SEED, and PBKDF1

The OpenSSL project has deprecated a set of cryptographic algorithms because they are insecure, uncommonly used, or both. Red Hat also discourages the use of those algorithms, and RHEL 9 provides them for migrating encrypted data to use new algorithms. Users must not depend on those algorithms for the security of their systems.

The implementations of the following algorithms have been moved to the legacy provider in OpenSSL: MD2, MD4, MDC2, Whirlpool, RIPEMD160, Blowfish, CAST, DES, IDEA, RC2, RC4, RC5, SEED, and PBKDF1.

See the **/etc/pki/tls/openssl.cnf** configuration file for instructions on how to load the legacy provider and enable support for the deprecated algorithms.

(BZ#1975836)

/etc/system-fips is now deprecated

Support for indicating FIPS mode through the **/etc/system-fips** file has been removed, and the file will not be included in future versions of RHEL. To install RHEL in FIPS mode, add the **fips=1** parameter to the kernel command line during the system installation. You can check whether RHEL operates in FIPS mode by using the **fips-mode-setup --check** command.

(JIRA:RHELPLAN-103232)

libcrypt.so.1 is now deprecated

The **libcrypt.so.1** library is now deprecated, and it might be removed in a future version of RHEL.

(BZ#2034569)

fapolicyd.rules is deprecated

The **/etc/fapolicyd/rules.d/** directory for files containing allow and deny execution rules replaces the **/etc/fapolicyd/fapolicyd.rules** file. The **fagenrules** script now merges all component rule files in this directory to the **/etc/fapolicyd/compiled.rules** file. Rules in **/etc/fapolicyd/fapolicyd.trust** are still processed by the **fapolicyd** framework but only for ensuring backward compatibility.

(BZ#2054740)

7.4. NETWORKING

ipset and iptables-nft have been deprecated

The **ipset** and **iptables-nft** packages have been deprecated in RHEL. The **iptables-nft** package contains different tools such as **iptables**, **ip6tables**, **ebtables** and **arptables**. These tools will no longer receive new features and using them for new deployments is not recommended. As a replacement, prefer using the **nft** command-line tool provided by the **nftables** package. Existing setups should migrate to **nft** if possible.

When you load the **iptables**, **ip6tables**, **ebtables**, **arptables**, **nft_compat**, or **ipset** module, the module logs the following warning to the **/var/log/messages** file:

> Warning: <module_name> - this driver is not recommended for new deployments. It continues to be supported in this RHEL release, but it is likely to be removed in the next major release. Driver updates and fixes will be limited to critical issues. Please contact Red Hat Support for additional information.

For more information on migrating to nftables, see Migrating from iptables to nftables , as well as the **iptables-translate(8)** and **ip6tables-translate(8)** man pages.

(BZ#1945151)

Network teams are deprecated in RHEL 9

The **teamd** service and the **libteam** library are deprecated in Red Hat Enterprise Linux 9 and will be removed in the next major release. As a replacement, configure a bond instead of a network team.

Red Hat focuses its efforts on kernel-based bonding to avoid maintaining two features, bonds and teams, that have similar functions. The bonding code has a high customer adoption, is robust, and has an active community development. As a result, the bonding code receives enhancements and updates.

For details about how to migrate a team to a bond, see Migrating a network team configuration to network bond.

(BZ#1935544)

NetworkManager connection profiles in ifcfg format are deprecated

In RHEL 9.0 and later, connection profiles in **ifcfg** format are deprecated. The next major RHEL release will remove the support for this format. However, in RHEL 9, NetworkManager still processes and updates existing profiles in this format if you modify them.

By default, NetworkManager now stores connection profiles in keyfile format in the **/etc/NetworkManager/system-connections/** directory. Unlike the **ifcfg** format, the keyfile format supports all connection settings that NetworkManager provides. For further details about the keyfile format and how to migrate profiles, see NetworkManager connection profiles in keyfile format .

(BZ#1894877)

The iptables back end in firewalld is deprecated

In RHEL 9, the **iptables** framework is deprecated. As a consequence, the **iptables** backend and the **direct interface** in **firewalld** are also deprecated. Instead of the **direct interface** you can use the native features in **firewalld** to configure the required rules.

(BZ#2089200)

7.5. KERNEL

ATM encapsulation is deprecated in RHEL 9

Asynchronous Transfer Mode (ATM) encapsulation enables Layer-2 (Point-to-Point Protocol, Ethernet) or Layer-3 (IP) connectivity for the ATM Adaptation Layer 5 (AAL-5). Red Hat has not been providing support for ATM NIC drivers since RHEL 7. The support for ATM implementation is being dropped in RHEL 9. These protocols are currently used only in chipsets, which support the ADSL technology and are being phased out by manufacturers. Therefore, ATM encapsulation is deprecated in Red Hat Enterprise Linux 9.

For more information, see PPP Over AAL5, Multiprotocol Encapsulation over ATM Adaptation Layer 5 , and Classical IP and ARP over ATM .

(BZ#2058153)

v4l/dvb television and video capture devices are no longer supported

With RHEL 9, Red Hat no longer supports **Video4Linux** (**v4l**) and **Linux DVB** (**DVB**) devices that consist of various television tuner cards and miscellaneous video capture cards and Red Hat no longer provides their associated drivers.

(BZ#2074598)

7.6. FILE SYSTEMS AND STORAGE

lvm2-activation-generator and its generated services removed in RHEL 9.0

The **lvm2-activation-generator** program and its generated services **lvm2-activation**, **lvm2-activation-early**, and **lvm2-activation-net** are removed in RHEL 9.0. The **lvm.conf event_activation** setting, used to activate the services, is no longer functional. The only method for auto activating volume groups is event based activation.

(BZ#2038183)

7.7. DYNAMIC PROGRAMMING LANGUAGES, WEB AND DATABASE SERVERS

libdb has been deprecated

RHEL 8 and RHEL 9 currently provide Berkeley DB (**libdb**) version 5.3.28, which is distributed under the LGPLv2 license. The upstream Berkeley DB version 6 is available under the AGPLv3 license, which is more restrictive.

The **libdb** package is deprecated as of RHEL 9 and might not be available in future major RHEL releases.

In addition, cryptographic algorithms have been removed from **libdb** in RHEL 9 and multiple **libdb** dependencies have been removed from RHEL 9.

Users of **libdb** are advised to migrate to a different key-value database. For more information, see the Knowledgebase article Available replacements for the deprecated Berkeley DB (libdb) in RHEL .

(BZ#1927780, BZ#1974657, JIRA:RHELPLAN-80695)

7.8. IDENTITY MANAGEMENT

SHA-1 in OpenDNSSec is now deprecated

OpenDNSSec supports exporting Digital Signatures and authentication records using the **SHA-1** algorithm. The use of the **SHA-1** algorithm is no longer supported. With the RHEL 9 release, **SHA-1** in OpenDNSSec is deprecated and it might be removed in a future minor release. Additionally, OpenDNSSec support is limited to its integration with Red Hat Identity Management. OpenDNSSec is not supported standalone.

(BZ#1979521)

The SSSD implicit files provider domain is disabled by default

The SSSD implicit **files** provider domain, which retrieves user information from local files such as **/etc/shadow** and group information from **/etc/groups**, is now disabled by default.

To retrieve user and group information from local files with SSSD:

1. Configure SSSD. Choose one of the following options:

a. Explicitly configure a local domain with the **id_provider=files** option in the **sssd.conf** configuration file.

```
[domain/local]
id_provider=files
...
```

b. Enable the **files** provider by setting **enable_files_domain=true** in the **sssd.conf** configuration file.

```
[sssd]
enable_files_domain = true
```

2. Configure the name services switch.

```
# authselect enable-feature with-files-provider
```

(JIRA:RHELPLAN-100639)

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

7.9. GRAPHICS INFRASTRUCTURES

X.org Server is now deprecated

The **X.org** display server is deprecated, and will be removed in a future major RHEL release. The default desktop session is now the **Wayland** session in most cases.

The **X11** protocol remains fully supported using the **XWayland** back end. As a result, applications that require **X11** can run in the **Wayland** session.

Red Hat is working on resolving the remaining problems and gaps in the **Wayland** session. For the outstanding problems in **Wayland**, see the Known issues section.

You can switch your user session back to the **X.org** back end. For more information, see Selecting GNOME environment and display protocol.

(JIRA:RHELPLAN-121048)

Motif has been deprecated

The Motif widget toolkit has been deprecated in RHEL, because development in the upstream Motif community is inactive.

The following Motif packages have been deprecated, including their development and debugging variants:

- **motif**

- **openmotif**
- **openmotif21**
- **openmotif22**

Additionally, the **motif-static** package has been removed.

Red Hat recommends using the GTK toolkit as a replacement. GTK is more maintainable and provides new features compared to Motif.

(JIRA:RHELPLAN-98983)

7.10. RED HAT ENTERPRISE LINUX SYSTEM ROLES

The networking system role displays a deprecation warning when configuring teams on RHEL 9 nodes

The network teaming capabilities have been deprecated in RHEL 9. As a result, using the **networking** RHEL system role on an RHEL 8 controller to configure a network team on RHEL 9 nodes, shows a warning about its deprecation.

(BZ#1999770)

7.11. VIRTUALIZATION

SecureBoot image verification using SHA1-based signatures is deprecated

Performing SecureBoot image verification using SHA1-based signatures on UEFI (PE/COFF) executables has become deprecated. Instead, Red Hat recommends using signatures based on the SHA2 algorithm, or later.

(BZ#1935497)

Limited support for virtual machine snapshots

Creating snapshots of virtual machines (VMs) is currently only supported for VMs not using the UEFI firmware. In addition, during the snapshot operation, the QEMU monitor may become blocked, which negatively impacts the hypervisor performance for certain workloads.

Also note that the current mechanism of creating VM snapshots has been deprecated, and Red Hat does not recommend using VM snapshots in a production environment. However, a new VM snapshot mechanism is under development and is planned to be fully implemented in a future minor release of RHEL 9.

(JIRA:RHELPLAN-15509, BZ#1621944)

virt-manager has been deprecated

The Virtual Machine Manager application, also known as **virt-manager**, has been deprecated. The RHEL web console, also known as **Cockpit**, is intended to become its replacement in a subsequent release. It is, therefore, recommended that you use the web console for managing virtualization in a GUI. Note, however, that some features available in **virt-manager** may not be yet available in the RHEL web console.

(JIRA:RHELPLAN-10304)

libvirtd has become deprecated

The monolithic **libvirt** daemon, **libvirtd**, has been deprecated in RHEL 9, and will be removed in a future major release of RHEL. Note that you can still use **libvirtd** for managing virtualization on your hypervisor, but Red Hat recommends switching to the newly introduced modular **libvirt** daemons. For instructions and details, see the RHEL 9 Configuring and Managing Virtualization document.

(JIRA:RHELPLAN-113995)

The virtual floppy driver has become deprecated

The **isa-fdc** driver, which controls virtual floppy disk devices, is now deprecated, and will become unsupported in a future release of RHEL. Therefore, to ensure forward compatibility with migrated virtual machines (VMs), Red Hat discourages using floppy disk devices in VMs hosted on RHEL 9.

(BZ#1965079)

qcow2-v2 image format is deprecated

With RHEL 9, the qcow2-v2 format for virtual disk images has become deprecated, and will become unsupported in a future major release of RHEL. In addition, the RHEL 9 Image Builder cannot create disk images in the qcow2-v2 format.

Instead of qcow2-v2, Red Hat strongly recommends using qcow2-v3. To convert a qcow2-v2 image to a later format version, use the **qemu-img amend** command.

(BZ#1951814)

7.12. CONTAINERS

Running RHEL 9 containers on a RHEL 7 host is not supported

Running RHEL 9 containers on a RHEL 7 host is not supported. It might work, but it is not guaranteed.

For more information, see Red Hat Enterprise Linux Container Compatibility Matrix .

(JIRA:RHELPLAN-100087)

SHA1 hash algorithm within Podman has been deprecated

The SHA1 algorithm used to generate the filename of the rootless network namespace is no longer supported in Podman. Therefore, rootless containers started before updating to Podman 4.1.1 from the RHBA-2022:5951 advisory have to be restarted if they are joined to a network (and not just using **slirp4netns**) to ensure they can connect to containers started after the upgrade.

(BZ#2069279)

rhel9/pause has been deprecated

The **rhel9/pause** container image has been deprecated.

(BZ#2106816)

7.13. DEPRECATED PACKAGES

This section lists packages that have been deprecated and will probably not be included in a future major release of Red Hat Enterprise Linux.

For changes to packages between RHEL 8 and RHEL 9, see Changes to packages in the *Considerations in adopting RHEL 9* document.

IMPORTANT

The support status of deprecated packages remains unchanged within RHEL 9. For more information about the length of support, see Red Hat Enterprise Linux Life Cycle and Red Hat Enterprise Linux Application Streams Life Cycle .

The following packages have been deprecated in RHEL 9:

- iptables-devel
- iptables-libs
- iptables-nft
- iptables-nft-services
- iptables-utils
- libdb
- mcpp
- python3-pytz

CHAPTER 8. KNOWN ISSUES

This part describes known issues in Red Hat Enterprise Linux 9.0.

8.1. INSTALLER AND IMAGE CREATION

The reboot --kexec and inst.kexec commands do not provide a predictable system state

Performing a RHEL installation with the **reboot --kexec** Kickstart command or the **inst.kexec** kernel boot parameters do not provide the same predictable system state as a full reboot. As a consequence, switching to the installed system without rebooting can produce unpredictable results.

Note that the **kexec** feature is deprecated and will be removed in a future release of Red Hat Enterprise Linux.

(BZ#1697896)

Local Media installation source is not detected when booting the installation from a USB that is created using a third party tool

When booting the RHEL installation from a USB that is created using a third party tool, the installer fails to detect the **Local Media** installation source (only *Red Hat CDN* is detected).

This issue occurs because the default boot option **int.stage2=** attempts to search for **iso9660** image format. However, a third party tool might create an ISO image with a different format.

As a workaround, use either of the following solution:

- When booting the installation, click the **Tab** key to edit the kernel command line, and change the boot option **inst.stage2=** to **inst.repo=**.
- To create a bootable USB device on Windows, use Fedora Media Writer.
- When using a third party tool like Rufus to create a bootable USB device, first regenerate the RHEL ISO image on a Linux system, and then use the third party tool to create a bootable USB device.

For more information on the steps involved in performing any of the specified workaround, see, Installation media is not auto detected during the installation of RHEL 8.3 .

(BZ#1877697)

The auth and authconfig Kickstart commands require the AppStream repository

The **authselect-compat** package is required by the **auth** and **authconfig** Kickstart commands during installation. Without this package, the installation fails if **auth** or **authconfig** are used. However, by design, the **authselect-compat** package is only available in the AppStream repository.

To work around this problem, verify that the BaseOS and AppStream repositories are available to the installer or use the **authselect** Kickstart command during installation.

(BZ#1640697)

Unexpected SELinux policies on systems where Anaconda is running as an application

When Anaconda is running as an application on an already installed system (for example to perform another installation to an image file using the **–image** anaconda option), the system is not prohibited to

modify the SELinux types and attributes during installation. As a consequence, certain elements of SELinux policy might change on the system where Anaconda is running. To work around this problem, do not run Anaconda on the production system and execute it in a temporary virtual machine. So that the SELinux policy on a production system is not modified. Running anaconda as part of the system installation process such as installing from **boot.iso** or **dvd.iso** is not affected by this issue.

(BZ#2050140)

The USB CD-ROM drive is not available as an installation source in Anaconda

Installation fails when the USB CD-ROM drive is the source for it and the Kickstart **ignoredisk --only-use=** command is specified. In this case, Anaconda cannot find and use this source disk.

To work around this problem, use the **harddrive --partition=sdX --dir=**/ command to install from USB CD-ROM drive. As a result, the installation does not fail.

(BZ#1914955)

Minimal RHEL installation no longer includes the s390utils-base package

In RHEL 8.4 and later, the **s390utils-base** package is split into an **s390utils-core** package and an auxiliary **s390utils-base** package. Consequently, setting the RHEL installation to **minimal-environment** installs only the necessary **s390utils-core** package and not the auxiliary **s390utils-base** package. To work around this problem, manually install the **s390utils-base** package after completing the RHEL installation or explicitly install **s390utils-base** using a kickstart file.

(BZ#1932480)

Hard drive partitioned installations with iso9660 filesystem fails

You cannot install RHEL on systems where the hard drive is partitioned with the **iso9660** filesystem. This is due to the updated installation code that is set to ignore any hard disk containing a **iso9660** file system partition. This happens even when RHEL is installed without using a DVD.

To workaround this problem, add the following script in the kickstart file to format the disc before the installation starts.

Note: Before performing the workaround, backup the data available on the disk. The **wipefs** command formats all the existing data from the disk.

```
%pre
wipefs -a /dev/sda
%end
```

As a result, installations work as expected without any errors.

(BZ#1929105)

Anaconda fails to verify existence of an administrator user account

While installing RHEL using a graphical user interface, Anaconda fails to verify if the administrator account has been created. As a consequence, users might install a system without any administrator user account.

To work around this problem, ensure you configure an administrator user account or the root password is set and the root account is unlocked. As a result, users can perform administrative tasks on the installed system.

(BZ#2047713)

Anaconda fails to login iSCSI server using the no authentication method after unsuccessful CHAP authentication attempt

When you add iSCSI discs using CHAP authentication and the login attempt fails due to incorrect credentials, a relogin attempt to the discs with the **no authentication** method fails. To workaround this problem, close the current session and login using the **no authentication** method.

(BZ#1983602)

New XFS features prevent booting of PowerNV IBM POWER systems with firmware older than version 5.10

PowerNV IBM POWER systems use a Linux kernel for firmware, and use Petitboot as a replacement for GRUB. This results in the firmware kernel mounting **/boot** and Petitboot reading the GRUB config and booting RHEL.

The RHEL 9 kernel introduces **bigtime=1** and **inobtcount=1** features to the XFS filesystem, which kernels with firmware older than version 5.10 do not understand.

To work around this problem, you can use another filesystem for **/boot**, for example ext4.

(BZ#1997832)

Cannot install RHEL when PReP is not 4 or 8 MiB in size

The RHEL installer cannot install the boot loader if the PowerPC Reference Platform (PReP) partition is of a different size than 4 MiB or 8 MiB on a disk that uses 4 kiB sectors. As a consequence, you cannot install RHEL on the disk.

To work around the problem, make sure that the PReP partition is exactly 4 MiB or 8 MiB in size, and that the size is not rounded to another value. As a result, the installer can now install RHEL on the disk.

(BZ#2026579)

New XFS features prevent booting of PowerNV IBM POWER systems with firmware kernel older than version 5.10

PowerNV IBM POWER systems use a Linux kernel for firmware, and use Petitboot as a replacement for GRUB. This results in the firmware kernel mounting **/boot** and Petitboot reading the GRUB config and booting RHEL.

The RHEL 9 kernel introduces **bigtime=1** and **inobtcount=1** features to the XFS filesystem, which firmware with kernel older than version 5.10 do not understand. As a consequence, Anaconda prevents the installation with the following error message:

Your firmware doesn't support XFS file system features on the **/boot** file system. The system will not be bootable. Please, upgrade the firmware or change the file system type.

As a workaround, use another filesystem for **/boot**, for example **ext4**.

(BZ#2008792)

8.2. SUBSCRIPTION MANAGEMENT

virt-who cannot connect to ESX servers when in FIPS mode

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

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

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

- Do not set the RHEL 9 system you use for running **virt-who** to FIPS mode.
- Do not upgrade the RHEL system you use for running **virt-who** to version 9.0.

(BZ#2054504)

8.3. SOFTWARE MANAGEMENT

The Installation process sometimes becomes unresponsive

When you install RHEL, the installation process sometimes becomes unresponsive. The **/tmp/packaging.log** file displays the following message at the end:

```
10:20:56,416 DDEBUG dnf: RPM transaction over.
```

To workaround this problem, restart the installation process.

(BZ#2073510)

8.4. SHELLS AND COMMAND-LINE TOOLS

ReaR fails during recovery if the TMPDIR variable is set in the configuration file

Setting and exporting **TMPDIR** in the **/etc/rear/local.conf** or **/etc/rear/site.conf** ReaR configuration file does not work and is deprecated.

The ReaR default configuration file **/usr/share/rear/conf/default.conf** contains the following instructions:

```
# To have a specific working area directory prefix for Relax-and-Recover
# specify in /etc/rear/local.conf something like
#
# export TMPDIR="/prefix/for/rear/working/directory"
#
# where /prefix/for/rear/working/directory must already exist.
# This is useful for example when there is not sufficient free space
# in /tmp or $TMPDIR for the ISO image or even the backup archive.
```

The instructions mentioned above do not work correctly because the **TMPDIR** variable has the same value in the rescue environment, which is not correct if the directory specified in the **TMPDIR** variable does not exist in the rescue image.

As a consequence, setting and exporting **TMPDIR** in the **/etc/rear/local.conf** file leads to the following error when the rescue image is booted :

```
mktemp: failed to create file via template '/prefix/for/rear/working/directory/tmp.XXXXXXXXXX': No
```

```
such file or directory
cp: missing destination file operand after '/etc/rear/mappings/mac'
Try 'cp --help' for more information.
No network interface mapping is specified in /etc/rear/mappings/mac
```

or the following error and abort later, when running **rear recover**:

```
ERROR: Could not create build area
```

To work around this problem, if you want to have a custom temporary directory, specify a custom directory for ReaR temporary files by exporting the variable in the shell environment before executing ReaR. For example, execute the **export TMPDIR=...** statement and then execute the **rear** command in the same shell session or script. As a result, the recovery is successful in the described configuration.

Jira:RHEL-24847

Renaming network interfaces using ifcfg files fails

On RHEL 9, the **initscripts** package is not installed by default. Consequently, renaming network interfaces using **ifcfg** files fails. To solve this problem, Red Hat recommends that you use **udev** rules or link files to rename interfaces. For further details, see Consistent network interface device naming and the **systemd.link(5)** man page.

If you cannot use one of the recommended solutions, install the **initscripts** package.

(BZ#2018112)

The chkconfig package is not installed by default in RHEL 9

The **chkconfig** package, which updates and queries runlevel information for system services, is not installed by default in RHEL 9.

To manage services, use the **systemctl** commands or install the **chkconfig** package manually.

For more information about **systemd**, see Managing systemd. For instructions on how to use the **systemctl** utility, see Managing system services with systemctl.

(BZ#2053598)

8.5. INFRASTRUCTURE SERVICES

Both bind and unbound disable validation of SHA-1-based signatures

The **bind** and **unbound** components disable validation support of all RSA/SHA1 (algorithm number 5) and RSASHA1-NSEC3-SHA1 (algorithm number 7) signatures, and the SHA-1 usage for signatures is restricted in the DEFAULT system-wide cryptographic policy.

As a result, certain DNSSEC records signed with the SHA-1, RSA/SHA1, and RSASHA1-NSEC3-SHA1 digest algorithms fail to verify in Red Hat Enterprise Linux 9 and the affected domain names become vulnerable.

To work around this problem, upgrade to a different signature algorithm, such as RSA/SHA-256 or elliptic curve keys.

For more information and a list of top-level domains that are affected and vulnerable, see the DNSSEC records signed with RSASHA1 fail to verify solution.

(BZ#2070495)

named fails to start if the same writable zone file is used in multiple zones

BIND does not allow the same writable zone file in multiple zones. Consequently, if a configuration includes multiple zones which share a path to a file that can be modified by the **named** service, **named** fails to start. To work around this problem, use the **in-view** clause to share one zone between multiple views and make sure to use different paths for different zones. For example, include the view names in the path.

Note that writable zone files are typically used in zones with allowed dynamic updates, slave zones, or zones maintained by DNSSEC.

(BZ#1984982)

Setting the console keymap requires the libxkbcommon library on your minimal install

In RHEL 9, certain **systemd** library dependencies have been converted from dynamic linking to dynamic loading, so that your system opens and uses the libraries at runtime when they are available. With this change, a functionality that depends on such libraries is not available unless you install the necessary library. This also affects setting the keyboard layout on systems with a minimal install. As a result, the **localectl --no-convert set-x11-keymap gb** command fails.

To work around this problem, install the **libxkbcommon** library:

```
# dnf install libxkbcommon
```

(BZ#2214130)

8.6. SECURITY

OpenSSL does not detect if a PKCS #11 token supports the creation of raw RSA or RSA-PSS signatures

The TLS 1.3 protocol requires support for RSA-PSS signatures. If a PKCS #11 token does not support raw RSA or RSA-PSS signatures, server applications that use the **OpenSSL** library fail to work with an **RSA** key if the key is held by the **PKCS #11** token. As a result, TLS communication fails in the described scenario.

To work around this problem, configure servers and clients to use TLS version 1.2 as the highest TLS protocol version available.

(BZ#1681178)

OpenSSL incorrectly handles PKCS #11 tokens that does not support raw RSA or RSA-PSS signatures

The **OpenSSL** library does not detect key-related capabilities of PKCS #11 tokens. Consequently, establishing a TLS connection fails when a signature is created with a token that does not support raw RSA or RSA-PSS signatures.

To work around the problem, add the following lines after the **.include** line at the end of the **crypto_policy** section in the **/etc/pki/tls/openssl.cnf** file:

```
SignatureAlgorithms =
RSA+SHA256:RSA+SHA512:RSA+SHA384:ECDSA+SHA256:ECDSA+SHA512:ECDSA+SHA384
MaxProtocol = TLSv1.2
```

As a result, a TLS connection can be established in the described scenario.

(BZ#1685470)

Cryptography not approved by FIPS works in OpenSSL in FIPS mode

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

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

(BZ#2053289)

OpenSSL cannot use engines in FIPS mode

Engine API is deprecated in OpenSSL 3.0 and is incompatible with OpenSSL Federal Information Processing Standards (FIPS) implementation and other FIPS-compatible implementations. Therefore, OpenSSL cannot run engines in FIPS mode. There is no workaround for this problem.

(BZ#2087253)

PSK ciphersuites do not work with the FUTURE crypto policy

Pre-shared key (PSK) ciphersuites are not recognized as performing perfect forward secrecy (PFS) key exchange methods. As a consequence, the **ECDHE-PSK** and **DHE-PSK** ciphersuites do not work with OpenSSL configured to **SECLEVEL=3**, for example with the **FUTURE** crypto policy. As a workaround, you can set a less restrictive crypto policy or set a lower security level (**SECLEVEL**) for applications that use PSK ciphersuites.

(BZ#2060044)

GnuPG incorrectly allows using SHA-1 signatures even if disallowed by crypto-policies

The GNU Privacy Guard (GnuPG) cryptographic software can create and verify signatures that use the SHA-1 algorithm regardless of the settings defined by the system-wide cryptographic policies. Consequently, you can use SHA-1 for cryptographic purposes in the **DEFAULT** cryptographic policy, which is not consistent with the system-wide deprecation of this insecure algorithm for signatures.

To work around this problem, do not use GnuPG options that involve SHA-1. As a result, you will prevent GnuPG from lowering the default system security by using the non-secure SHA-1 signatures.

(BZ#2070722)

Some OpenSSH operations do not used FIPS-approved interfaces

The OpenSSL cryptographic library, which is used by OpenSSH, provides two interfaces: legacy and modern. Because of changes to OpenSSL internals, only the modern interfaces use FIPS-certified implementations of cryptographic algorithms. Because OpenSSH uses legacy interfaces for some operations, it does not comply with FIPS requirements.

(BZ#2087121)

gpg-agent does not work as an SSH agent in FIPS mode

The **gpg-agent** tool creates MD5 fingerprints when adding keys to the **ssh-agent** program even though FIPS mode disables the MD5 digest. Consequently, the **ssh-add** utility fails to add the keys to the authentication agent.

To work around the problem, create the **~/.gnupg/sshcontrol** file without using the **gpg-agent --daemon --enable-ssh-support** command. For example, you can paste the output of the **gpg --list-keys** command in the **<FINGERPRINT> 0** format to **~/.gnupg/sshcontrol**. As a result, **gpg-agent** works as an SSH authentication agent.

(BZ#2073567)

SELinux staff_u users can incorrectly switch to unconfined_r

When the **secure_mode** boolean is enabled, **staff_u** users can incorrectly switch to the **unconfined_r** role. As a consequence, **staff_u** users can perform privileged operations affecting the security of the system.

(BZ#2021529)

Default SELinux policy allows unconfined executables to make their stack executable

The default state of the **selinuxuser_execstack** boolean in the SELinux policy is on, which means that unconfined executables can make their stack executable. Executables should not use this option, and it might indicate poorly coded executables or a possible attack. However, due to compatibility with other tools, packages, and third-party products, Red Hat cannot change the value of the boolean in the default policy. If your scenario does not depend on such compatibility aspects, you can turn the boolean off in your local policy by entering the command **setsebool -P selinuxuser_execstack off**.

(BZ#2064274)

Remediating service-related rules during kickstart installations might fail

During a kickstart installation, the OpenSCAP utility sometimes incorrectly shows that a service **enable** or **disable** state remediation is not needed. Consequently, OpenSCAP might set the services on the installed system to a non-compliant state. As a workaround, you can scan and remediate the system after the kickstart installation. This will fix the service-related issues.

(BZ#1834716)

SSH timeout rules in STIG profiles configure incorrect options

An update of OpenSSH affected the rules in the following Defense Information Systems Agency Security Technical Implementation Guide (DISA STIG) profiles:

- DISA STIG for RHEL 9 (**xccdf_org.ssgproject.content_profile_stig**)
- DISA STIG with GUI for RHEL 9 (**xccdf_org.ssgproject.content_profile_stig_gui**)

In each of these profiles, the following two rules are affected:

```
Title: Set SSH Client Alive Count Max to zero
CCE Identifier: CCE-90271-8
Rule ID: xccdf_org.ssgproject.content_rule_sshd_set_keepalive_0
```

```
Title: Set SSH Idle Timeout Interval
CCE Identifier: CCE-90811-1
Rule ID: xccdf_org.ssgproject.content_rule_sshd_set_idle_timeout
```

When applied to SSH servers, each of these rules configures an option (**ClientAliveCountMax** and **ClientAliveInterval**) that no longer behaves as previously. As a consequence, OpenSSH no longer disconnects idle SSH users when it reaches the timeout configured by these rules. As a workaround, these rules have been temporarily removed from the DISA STIG for RHEL 9 and DISA STIG with GUI for RHEL 9 profiles until a solution is developed.

(BZ#2038978)

fagenrules --load does not work correctly

The **fapolicyd** service does not correctly handle the signal hang up (SIGHUP). Consequently, **fapolicyd** terminates after receiving the SIGHUP signal. Therefore, the **fagenrules --load** command does not work properly, and rule updates require manual restarts of **fapolicyd**. To work around this problem, restart the **fapolicyd** service after any change in rules, and as a result **fagenrules --load** will work correctly.

(BZ#2070655)

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#2105162)

8.7. NETWORKING

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

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

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

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

2. Display the available connection profiles:

```
# nmcli connection show
```

3. Reactive the affected connection profiles:

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

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

(BZ#2151040)

An empty rd.znet option in the kernel command line causes the network configuration to fail

An **rd.znet** option without any arguments, such as net types or subchannels, in the kernel fails to configure networking. To work around this problem, either remove the **rd.znet** option from the command line completely or specify relevant net types, subchannels, and other relevant options. For more information about these options, see the **dracut.cmdline(7)** man page.

(BZ#1931284)

Failure to update the session key causes the connection to break

Kernel Transport Layer Security (kTLS) protocol does not support updating the session key, which is used by the symmetric cipher. Consequently, the user cannot update the key, which causes a connection break. To work around this problem, disable kTLS. As a result, with the workaround, it is possible to successfully update the session key.

(BZ#2013650)

The initscripts package is not installed by default

By default, the **initscripts** package is not installed. As a consequence, the **ifup** and **ifdown** utilities are not available. As an alternative, use the **nmcli connection up** and **nmcli connection down** commands to enable and disable connections. If the suggested alternative does not work for you, report the problem and install the **NetworkManager-initscripts-updown** package, which provides a NetworkManager solution for the **ifup** and **ifdown** utilities.

(BZ#2082303)

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)

8.8. KERNEL

kdump fails to start on RHEL 9 kernel

The RHEL 9 kernel does not have the **crashkernel=auto** parameter configured as default. Consequently, the **kdump** service fails to start by default.

To work around this problem, configure the **crashkernel=** option to the required value.

For example, to reserve 256 MB of memory using the **grubby** utility, enter the following command:

```
# grubby --args crashkernel=256M --update-kernel ALL
```

As a result, the RHEL 9 kernel starts **kdump** and uses the configured memory size value to dump the **vmcore** file.

(BZ#1894783)

The kdump mechanism fails to capture vmcore on LUKS-encrypted targets

When running **kdump** on systems with Linux Unified Key Setup (LUKS) encrypted partitions, systems require a certain amount of available memory. When the available memory is less than the required amount of memory, the **systemd-cryptsetup** service fails to mount the partition. Consequently, the second kernel fails to capture the crash dump file (**vmcore**) on LUKS-encrypted targets.

With the **kdumpctl estimate** command, you can query the **Recommended crashkernel value**, which is the recommended memory size required for **kdump**.

To work around this issue, use following steps to configure the required memory for **kdump** on LUKS encrypted targets:

1. Print the estimate **crashkernel** value:

   ```
   # kdumpctl estimate
   ```

2. Configure the amount of required memory by increasing the **crashkernel** value:

   ```
   # grubby --args=crashkernel=652M --update-kernel=ALL
   ```

3. Reboot the system for changes to take effect.

   ```
   # reboot
   ```

As a result, **kdump** works correctly on systems with LUKS-encrypted partitions.

(BZ#2017401)

Allocating crash kernel memory fails at boot time

On certain Ampere Altra systems, allocating the crash kernel memory for **kdump** usage fails during boot when the available memory is below 1 GB. Consequently, the **kdumpctl** command fails to start the **kdump** service as the required memory is more than the available memory size.

As a workaround, decrease the value of the **crashkernel** parameter by a minimum of 240 MB to fit the size requirement, for example **crashkernel=240M**. As a result, the crash kernel memory allocation for **kdump** does not fail on Ampere Altra systems.

(BZ#2065013)

kTLS does not support offloading of TLS 1.3 to NICs

Kernel Transport Layer Security (kTLS) does not support offloading of TLS 1.3 to NICs. Consequently, software encryption is used with TLS 1.3 even when the NICs support TLS offload. To work around this problem, disable TLS 1.3 if offload is required. As a result, you can offload only TLS 1.2. When TLS 1.3 is in use, there is lower performance, since TLS 1.3 cannot be offloaded.

(BZ#2000616)

FADump enabled with Secure Boot might lead to GRUB Out of Memory (OOM)

In the Secure Boot environment, GRUB and PowerVM together allocate a 512 MB memory region, known as the Real Mode Area (RMA), for boot memory. The region is divided among the boot components and, if any component exceeds its allocation, out-of-memory failures occur.

Generally, the default installed **initramfs** file system and the **vmlinux** symbol table are within the limits to avoid such failures. However, if Firmware Assisted Dump (FADump) is enabled in the system, the default **initramfs** size can increase and exceed 95 MB. As a consequence, every system reboot leads to a GRUB OOM state.

To avoid this issue, do not use Secure Boot and FADump together. For more information and methods on how to work around this issue, see link:https://www.ibm.com/support/pages/node/6846531.

(BZ#2149172)

Systems in Secure Boot cannot run dynamic LPAR operations

Users cannot run dynamic logical partition (DLPAR) operations from the Hardware Management Console (HMC) if either of these conditions are met:

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

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

> HSCL2957 Either there is currently no RMC connection between the management console and the partition <LPAR name> or the partition does not support dynamic partitioning operations. Verify the

> network setup on the management console and the partition and ensure that any firewall authentication between the management console and the partition has occurred. Run the management console diagrmc command to identify problems that might be causing no RMC connection.

(BZ#2083106)

dkms provides an incorrect warning on program failure with correctly compiled drivers on 64-bit ARM CPUs

The Dynamic Kernel Module Support (**dkms**) utility does not recognize that the kernel headers for 64-bit ARM CPUs work for both the kernels with 4 kilobytes and 64 kilobytes page sizes. As a result, when the kernel update is performed and the **kernel-64k-devel** package is not installed, **dkms** provides an incorrect warning on why the program failed on correctly compiled drivers. To work around this problem, install the **kernel-headers** package, which contains header files for both types of ARM CPU architectures and is not specific to **dkms** and its requirements.

(JIRA:RHEL-25967)

8.9. BOOT LOADER

New kernels lose previous command-line options

The GRUB boot loader does not apply custom, previously configured kernel command-line options to new kernels. Consequently, when you upgrade the kernel package, the system behavior might change after reboot due to the missing options.

To work around the problem, manually add all custom kernel command-line options after each kernel upgrade. As a result, the kernel applies custom options as expected, until the next kernel upgrade.

(BZ#1969362)

8.10. FILE SYSTEMS AND STORAGE

Device Mapper Multipath is not supported with NVMe/TCP

Using Device Mapper Multipath with the **nvme-tcp** driver can result in the Call Trace warnings and system instability. To work around this problem, NVMe/TCP users must enable native NVMe multipathing and not use the **device-mapper-multipath** tools with NVMe.

By default, Native NVMe multipathing is enabled in RHEL 9. For more information, see Enabling multipathing on NVMe devices.

(BZ#2033080)

The blk-availability systemd service deactivates complex device stacks

In **systemd**, the default block deactivation code does not always handle complex stacks of virtual block devices correctly. In some configurations, virtual devices might not be removed during the shutdown, which causes error messages to be logged. To work around this problem, deactivate complex block device stacks by executing the following command:

```
# systemctl enable --now blk-availability.service
```

As a result, complex virtual device stacks are correctly deactivated during shutdown and do not produce error messages.

(BZ#2011699)

Invalid sysfs value for supported_speeds

The **qla2xxx** driver reports 20Gb/s instead of the expected 64Gb/s as one of the supported port speeds in the **sysfs supported_speeds** attribute:

```
$ cat /sys/class/fc_host/host12/supported_speeds
16 Gbit, 32 Gbit, 20 Gbit
```

As a consequence, if the HBA supports 64Gb/s link speed, the **sysfs supported_speeds** value is incorrect. This affects only the **supported_speeds** value of **sysfs** and the port operates at the expected negotiated link rate.

(BZ#2069758)

Unable to connect to NVMe namespaces from Broadcom initiator on AMD EPYC systems

By default, the RHEL kernel enables the IOMMU on AMD-based platforms. Consequently, when you use IOMMU-enabled platforms on servers with AMD processors, you might experience NVMe I/O problems, such as I/Os failing due to transfer length mismatches.

To work around this problem, add the IOMMU in passthrough mode by using the kernel command-line option, **iommu=pt**. As a result, you can now connect to NVMe namespaces from Broadcom initiator on AMD EPYC systems.

(BZ#2073541)

8.11. DYNAMIC PROGRAMMING LANGUAGES, WEB AND DATABASE SERVERS

The --ssl-fips-mode option in MySQL and MariaDB does not change FIPS mode

The **--ssl-fips-mode** option in **MySQL** and **MariaDB** in RHEL works differently than in upstream.

In RHEL 9, if you use **--ssl-fips-mode** as an argument for the **mysqld** or **mariadbd** daemon, or if you use **ssl-fips-mode** in the **MySQL** or **MariaDB** server configuration files, **--ssl-fips-mode** does not change FIPS mode for these database servers.

Instead:

- If you set **--ssl-fips-mode** to **ON**, the **mysqld** or **mariadbd** server daemon does not start.
- If you set **--ssl-fips-mode** to **OFF** on a FIPS-enabled system, the **mysqld** or **mariadbd** server daemons still run in FIPS mode.

This is expected because FIPS mode should be enabled or disabled for the whole RHEL system, not for specific components.

Therefore, do not use the **--ssl-fips-mode** option in **MySQL** or **MariaDB** in RHEL. Instead, ensure FIPS mode is enabled on the whole RHEL system:

- Preferably, install RHEL with FIPS mode enabled. Enabling FIPS mode during the installation ensures that the system generates all keys with FIPS-approved algorithms and continuous monitoring tests in place. For information about installing RHEL in FIPS mode, see Installing the system in FIPS mode.

- Alternatively, you can switch FIPS mode for the entire RHEL system by following the procedure in Switching the system to FIPS mode .

(BZ#1991500)

8.12. COMPILERS AND DEVELOPMENT TOOLS

Certain symbol-based probes do not work in SystemTap on the 64-bit ARM architecture

Kernel configuration disables certain functionality needed for **SystemTap**. Consequently, some symbol-based probes do not work on the 64-bit ARM architecture. As a result, affected **SystemTap** scripts may not run or may not collect hits on desired probe points.

Note that this bug has been fixed for the remaining architectures with the release of the RHBA-2022:5259 advisory.

(BZ#2083727)

8.13. IDENTITY MANAGEMENT

RHEL 9 Kerberos client fails to authenticate a user using PKINIT against Heimdal KDC

During the PKINIT authentication of an IdM user on a RHEL 9 Kerberos client, the Heimdal Kerberos Distribution Center (KDC) on RHEL 9 or earlier uses the SHA-1 backup signature algorithm because the Kerberos client does not support the **supportedCMSTypes** field. However, the SHA-1 algorithm has been deprecated in RHEL 9 and therefore the user authentication fails.

To work around this problem, enable support for the SHA-1 algorithm on your RHEL 9 clients with the following command:

```
# update-crypto-policies --set DEFAULT:SHA1
```

As a result, PKINIT authentication works between the Kerberos client and Heimdal KDC.

For more details about supported backup signature algorithms, see Kerberos Encryption Types Defined for CMS Algorithm Identifiers.

See also The PKINIT authentication of a user fails if a RHEL 9 Kerberos agent communicates with a non-RHEL 9 Kerberos agent.

(BZ#2068935)

The PKINIT authentication of a user fails if a RHEL 9 Kerberos agent communicates with a non-RHEL 9 Kerberos agent

If a RHEL 9 Kerberos agent interacts with another, non-RHEL 9 Kerberos agent in your environment, the Public Key Cryptography for initial authentication (PKINIT) authentication of a user fails. To work around the problem, perform one of the following actions:

- Set the RHEL 9 agent's crypto-policy to **DEFAULT:SHA1** to allow the verification of SHA-1 signatures:

  ```
  # update-crypto-policies --set DEFAULT:SHA1
  ```

- Update the non-RHEL 9 agent to ensure it does not sign CMS data using the SHA-1 algorithm. For this, update your Kerberos packages to the versions that use SHA-256 instead of SHA-1:
  - CentOS 9 Stream: krb5-1.19.1-15
  - RHEL 8.7: krb5-1.18.2-17
  - RHEL 7.9: krb5-1.15.1-53
  - Fedora Rawhide/36: krb5-1.19.2-7
  - Fedora 35/34: krb5-1.19.2-3

You must perform one of these actions regardless of whether the non-patched agent is a Kerberos client or the Kerberos Distribution Center (KDC).

As a result, the PKINIT authentication of a user works correctly.

Note that for other operating systems, it is the krb5-1.20 release that ensures that the agent signs CMS data with SHA-256 instead of SHA-1.

See also The DEFAULT:SHA1 sub-policy has to be set on RHEL 9 clients for PKINIT to work against older RHEL KDCs and AD KDCs.

(BZ#2077450)

The DEFAULT:SHA1 sub-policy has to be set on RHEL 9 clients for PKINIT to work against older RHEL KDCs and AD KDCs

The SHA-1 digest algorithm has been deprecated in RHEL 9, and CMS messages for Public Key Cryptography for initial authentication (PKINIT) are now signed with the stronger SHA-256 algorithm.

While SHA-256 is used by default starting with RHEL 7.9 and RHEL 8.7, older Kerberos Key Distribution Centers (KDCs) on RHEL 7.8 and RHEL 8.6 and earlier still use the SHA-1 digest algorithm to sign CMS messages. So does the Active Directory (AD) KDC.

As a result, RHEL 9 Kerberos clients fail to authenticate users using PKINIT against the following:

- KDCs running on RHEL 7.8 and earlier
- KDCs running on RHEL 8.6 and earlier
- AD KDCs

To work around the problem, enable support for the SHA-1 algorithm on your RHEL 9 systems with the following command:

```
# update-crypto-policies --set DEFAULT:SHA1
```

See also RHEL 9 Kerberos client fails to authenticate a user using PKINIT against Heimdal KDC .

(BZ#2060798)

FIPS support for AD trust requires the AD-SUPPORT crypto sub-policy

Active Directory (AD) uses AES SHA-1 HMAC encryption types, which are not allowed in FIPS mode on RHEL 9 by default. If you want to use RHEL 9 IdM hosts with an AD trust, enable support for AES SHA-1 HMAC encryption types before installing IdM software.

Since FIPS compliance is a process that involves both technical and organizational agreements, consult your FIPS auditor before enabling the **AD-SUPPORT** sub-policy to allow technical measures to support AES SHA-1 HMAC encryption types, and then install RHEL IdM:

```
# update-crypto-policies --set FIPS:AD-SUPPORT
```

(BZ#2057471)

Directory Server terminates unexpectedly when started in referral mode

Due to a bug, global referral mode does not work in Directory Server. If you start the **ns-slapd** process with the **refer** option as the **dirsrv** user, Directory Server ignores the port settings and terminates unexpectedly. Trying to run the process as the **root** user changes SELinux labels and prevents the service from starting in future in normal mode. There are no workarounds available.

(BZ#2053204)

Configuring a referral for a suffix fails in Directory Server

If you set a back-end referral in Directory Server, setting the state of the backend using the **dsconf <instance_name> backend suffix set --state referral** command fails with the following error:

```
Error: 103 - 9 - 53 - Server is unwilling to perform - [] - need to set nsslapd-referral before moving to referral state
```

As a consequence, configuring a referral for suffixes fail. To work around the problem:

1. Set the **nsslapd-referral** parameter manually:

```
# ldapmodify -D "cn=Directory Manager" -W -H ldap://server.example.com

dn: cn=dc\3Dexample\2Cdc\3Dcom,cn=mapping tree,cn=config
changetype: modify
add: nsslapd-referral
nsslapd-referral: ldap://remote_server:389/dc=example,dc=com
```

2. Set the back-end state:

```
# dsconf <instance_name> backend suffix set --state referral
```

As a result, with the workaround, you can configure a referral for a suffix.

(BZ#2063140)

The dsconf utility has no option to create fix-up tasks for the entryUUID plug-in

The **dsconf** utility does not provide an option to create fix-up tasks for the **entryUUID** plug-in. As a result, administrators cannot not use **dsconf** to create a task to automatically add **entryUUID** attributes to existing entries. As a workaround, create a task manually:

```
# ldapadd -D "cn=Directory Manager" -W -H ldap://server.example.com -x

dn: cn=entryuuid_fixup___<time_stamp__,cn=entryuuid task,cn=tasks,cn=config
objectClass: top
objectClass: extensibleObject
```

```
basedn: __<fixup base tree>__
cn: entryuuid_fixup___<time_stamp>__
filter: __<filtered_entry>__
```

After the task has been created, Directory Server fixes entries with missing or invalid **entryUUID** attributes.

(BZ#2047175)

Potential risk when using the default value for ldap_id_use_start_tls option

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

Currently, the SSSD configuration option to enforce TLS, **ldap_id_use_start_tls**, defaults to **false**. Ensure that your setup operates in a trusted environment and decide if it is safe to use unencrypted communication for **id_provider = ldap**. Note **id_provider = ad** and **id_provider = ipa** are not affected as they use encrypted connections protected by SASL and GSSAPI.

If it is not safe to use unencrypted communication, enforce TLS by setting the **ldap_id_use_start_tls** option to **true** in the **/etc/sssd/sssd.conf** file. The default behavior is planned to be changed in a future release of RHEL.

(JIRA:RHELPLAN-155168)

8.14. DESKTOP

Firefox add-ons are disabled after upgrading to RHEL 9

If you upgrade from RHEL 8 to RHEL 9, all add-ons that you previously enabled in Firefox are disabled.

To work around the problem, manually reinstall or update the add-ons. As a result, the add-ons are enabled as expected.

(BZ#2013247)

VNC is not running after upgrading to RHEL 9

After upgrading from RHEL 8 to RHEL 9, the VNC server fails to start, even if it was previously enabled.

To work around the problem, manually enable the **vncserver** service after the system upgrade:

```
# systemctl enable --now vncserver@:port-number
```

As a result, VNC is now enabled and starts after every system boot as expected.

(BZ#2060308)

8.15. GRAPHICS INFRASTRUCTURES

Matrox G200e shows no output on a VGA display

Your display might show no graphical output if you use the following system configuration:

- The Matrox G200e GPU

- A display connected over the VGA controller

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

To work around the problem, use the following procedure:

1. Boot the system to the boot loader menu.
2. Add the **module_blacklist=mgag200** option to the kernel command line.

As a result, RHEL boots and shows graphical output as expected, but the maximum resolution is limited to 1024x768 at the 16-bit color depth.

(BZ#1960467)

X.org configuration utilities do not work under Wayland

X.org utilities for manipulating the screen do not work in the Wayland session. Notably, the **xrandr** utility does not work under Wayland due to its different approach to handling, resolutions, rotations, and layout.

(JIRA:RHELPLAN-121049)

NVIDIA drivers might revert to X.org

Under certain conditions, the proprietary NVIDIA drivers disable the Wayland display protocol and revert to the X.org display server:

- If the version of the NVIDIA driver is lower than 470.
- If the system is a laptop that uses hybrid graphics.
- If you have not enabled the required NVIDIA driver options.

Additionally, Wayland is enabled but the desktop session uses X.org by default if the version of the NVIDIA driver is lower than 510.

(JIRA:RHELPLAN-119001)

Night Light is not available on Wayland with NVIDIA

When the proprietary NVIDIA drivers are enabled on your system, the **Night Light** feature of GNOME is not available in Wayland sessions. The NVIDIA drivers do not currently support **Night Light**.

(JIRA:RHELPLAN-119852)

8.16. 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)

8.17. VIRTUALIZATION

Installing a virtual machine over https in some cases fails

Currently, the **virt-install** utility fails when attempting to install a guest operating system from an ISO source over a https connection - for example using **virt-install --cdrom https://example/path/to/image.iso**. Instead of creating a virtual machine (VM), the described operation terminates unexpectedly with an **internal error: process exited while connecting to monitor** message.

To work around this problem, install **qemu-kvm-block-curl** on the host to enable https protocol support. Alternatively, use a different connection protocol or a different installation source.

(BZ#2014229)

Using NVIDIA drivers in virtual machines disables Wayland

Currently, NVIDIA drivers are not compatible with the Wayland graphical session. As a consequence, RHEL guest operating systems that use NVIDIA drivers automatically disable Wayland and load an Xorg session instead. This primarily occurs in the following scenarios:

- When you pass through an NVIDIA GPU device to a RHEL virtual machine (VM)
- When you assign an NVIDIA vGPU mediated device to a RHEL VM

(JIRA:RHELPLAN-117234)

The Milan VM CPU type is sometimes not available on AMD Milan systems

On certain AMD Milan systems, the Enhanced REP MOVSB (**erms**) and Fast Short REP MOVSB (**fsrm**) feature flags are disabled in the BIOS by default. Consequently, the 'Milan' CPU type might not be available on these systems. In addition, VM live migration between Milan hosts with different feature flag settings might fail. To work around these problems, manually turn on **erms** and **fsrm** in the BIOS of your host.

(BZ#2077767)

Network traffic performance in virtual machines might be reduced

In some cases, RHEL 9.0 guest virtual machines (VMs) have somewhat decreased performance when handling high levels of network traffic.

(BZ#1945040)

Disabling AVX causes VMs to become unbootable

On a host machine that uses a CPU with Advanced Vector Extensions (AVX) support, attempting to boot a VM with AVX explicitly disabled currently fails, and instead triggers a kernel panic in the VM.

(BZ#2005173)

Failover virtio NICs are not assigned an IP address on Windows virtual machines

Currently, when starting a Windows virtual machine (VM) with only a failover virtio NIC, the VM fails to assign an IP address to the NIC. Consequently, the NIC is unable to set up a network connection. Currently, there is no workaround.

(BZ#1969724)

A hostdev interface with failover settings cannot be hot-plugged after being hot-unplugged

After removing a **hostdev** network interface with failover configuration from a running virtual machine (VM), the interface currently cannot be re-attached to the same running VM.

(BZ#2052424)

Live post-copy migration of VMs with failover VFs fails

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

(BZ#1817965, BZ#1789206)

8.18. RHEL IN CLOUD ENVIRONMENTS

SR-IOV performs suboptimally in ARM 64 RHEL 9 virtual machines on Azure

Currently, SR-IOV networking devices have significantly lower throughout and higher latency than expected in ARM 64 RHEL 9 virtual machines VMs running on a Microsoft Azure platform.

(BZ#2068432)

Mouse is not usable in RHEL 9 VMs on XenServer 7 with console proxy

When running a RHEL 9 virtual machine (VM) on a XenServer 7 platform with a console proxy, it is not possible to use the mouse in the VM's GUI. To work around this problem, disable the Wayland compositor protocol in the VM as follows:

1. Open the **/etc/gdm/custom.conf** file.
2. Uncomment the **WaylandEnable=false** line.
3. Save the file.

In addition, note that Red Hat does not support XenServer as a platform for running RHEL VMs, and discourages using XenServer with RHEL in production environments.

(BZ#2019593)

Cloning or restoring RHEL 9 virtual machines that use LVM on Nutanix AHV causes non-root partitions to disappear

When running a RHEL 9 guest operating system on a virtual machine (VM) hosted on the Nutanix AHV hypervisor, restoring the VM from a snapshot or cloning the VM currently causes non-root partitions in the VM to disappear if the guest is using Logical Volume Management (LVM). As a consequence, the following problems occur:

- After restoring the VM from a snapshot, the VM cannot boot, and instead enters emergency mode.
- A VM created by cloning cannot boot, and instead enters emergency mode.

To work around these problems, do the following in emergency mode of the VM:

1. Remove the LVM system devices file: **rm /etc/lvm/devices/system.devices**
2. Recreate LVM device settings: **vgimportdevices -a**
3. Reboot the VM

This makes it possible for the cloned or restored VM to boot up correctly.

(BZ#2059545)

The SR-IOV functionality of a network adapter attached to a Hyper-V virtual machine might not work

Currently, when attaching a network adapter with single-root I/O virtualization (SR-IOV) enabled to a RHEL 9 virtual machine (VM) running on Microsoft Hyper-V hypervisor, the SR-IOV functionality in some cases does not work correctly.

To work around this problem, disable SR-IOV in the VM configuration, and then enable it again.

1. In the Hyper-V Manager window, right-click the VM.
2. In the contextual menu, navigate to **Settings/Network Adapter/Hardware Acceleration**.
3. Uncheck **Enable SR-IOV**.
4. Click **Apply**.
5. Repeat steps 1 and 2 to navigate to the **Enable SR-IOV** option again.
6. Check **Enable SR-IOV**.
7. Click **Apply**.

(BZ#2030922)

Customizing RHEL 9 guests on ESXi sometimes causes networking problems

Currently, customizing a RHEL 9 guest operating system in the VMware ESXi hypervisor does not work correctly with NetworkManager key files. As a consequence, if the guest is using such a key file, it will have incorrect network settings, such as the IP address or the gateway.

For details and workaround instructions, see the VMware Knowledge Base.

(BZ#2037657)

8.19. SUPPORTABILITY

Timeout when running sos report on IBM Power Systems, Little Endian

When running the **sos report** command on IBM Power Systems, Little Endian with hundreds or thousands of CPUs, the processor plugin reaches its default timeout of 300 seconds when collecting

huge content of the **/sys/devices/system/cpu** directory. As a workaround, increase the plugin's timeout accordingly:

- For one-time setting, run:

```
# sos report -k processor.timeout=1800
```

- For a permanent change, edit the **[plugin_options]** section of the **/etc/sos/sos.conf** file:

```
[plugin_options]
# Specify any plugin options and their values here. These options take the form
# plugin_name.option_name = value
#rpm.rpmva = off
processor.timeout = 1800
```

The example value is set to 1800. The particular timeout value highly depends on a specific system. To set the plugin's timeout appropriately, you can first estimate the time needed to collect the one plugin with no timeout by running the following command:

```
# time sos report -o processor -k processor.timeout=0 --batch --build
```

(BZ#1869561)

8.20. CONTAINERS

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

Currently, when you try to pull RHEL 9 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#2020026)

Podman fails to pull a container "X509: certificate signed by unknown authority"

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

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

Import the CA certificates on your host:

```
# cd /etc/pki/ca-trust/source/anchors/
[anchors]# curl -O <your_certificate>.crt

[anchors]# update-ca-trust
```

Then you can pull container images from the internal registry.

(BZ#2027576)

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)

podman system connection add and **podman image scp** fails

Podman uses SHA-1 hashes for the RSA key exchange. The regular SSH connection among machines using RSA keys works, while the **podman system connection add** and **podman image scp** commands do not work using the same RSA keys, because the SHA-1 hashes are not accepted for key exchange on RHEL 9:

```
$ podman system connection add --identity ~/.ssh/id_rsa test_connection
$REMOTE_SSH_MACHINE
Error: failed to connect: ssh: handshake failed: ssh: unable to authenticate, attempted methods [none
publickey], no supported methods remain
```

To work around this problem, use the ED25519 keys:

1. Connect to the remote machine:

   ```
   $ ssh -i ~/.ssh/id_ed25519 $REMOTE_SSH_MACHINE
   ```

2. Record ssh destination for the Podman service:

```
$ podman system connection add --identity ~/.ssh/id_ed25519 test_connection
$REMOTE_SSH_MACHINE
```

3. Verify that the ssh destination was recorded:

```
$ podman system connection list
```

Note that with the release of the RHBA-2022:5951 advisory, the problem has been fixed.

(JIRA:RHELPLAN-121180)

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#2024693, BZ#1805717, BZ#1779685, BZ#2053204, BZ#2063140, BZ#2047175
ModemManager	BZ#1996716
NetworkManager	BZ#1980387, BZ#1949127, BZ#2060013, BZ#1931284, BZ#1894877, BZ#2079849
RHCOS	BZ#2008521
WALinuxAgent	BZ#1972101
alsa-lib	BZ#2015863
anaconda	BZ#1951709, BZ#1978264, BZ#2025953, BZ#2009403, BZ#2050140, BZ#1877697, BZ#1914955, BZ#1929105, BZ#1983602, BZ#1997832, BZ#2008792
ansible-collection-microsoft-sql	BZ#2064648, BZ#2064690
ansible-collection-redhat-rhel_mgmt	BZ#2023381
ansible-pcp	BZ#1957566
bash	BZ#2079078
bind	BZ#1984982
binutils	BZ#2030554
boost	BZ#1957950
chrony	BZ#1961131
clevis	BZ#1956760
cloud-init	BZ#2040090, BZ#2042351
cmake	BZ#1957948

Component	Tickets
container-tools	BZ#2000871
containers-common	BZ#2019901
crash	BZ#1896647
createrepo_c	BZ#2055032
crypto-policies	BZ#2004207, BZ#2013195
cyrus-sasl	BZ#1947971, BZ#1995600
device-mapper-multipath	BZ#2017979, BZ#2017592, BZ#2011699
distribution	BZ#1878583
dnf	BZ#2005305, BZ#2073510
dotnet6.0	BZ#1986211
edk2	BZ#1935497
eigen3	BZ#2032423
fapolicyd	BZ#2032408, BZ#1932225, BZ#2054740, BZ#2070655
fence-agents	BZ#1977588
fetchmail	BZ#1999276
fido-device-onboard	BZ#1989930
firefox	BZ#1764205, BZ#2013247
firewalld	BZ#2029211
freeradius	BZ#1978216
gcc	BZ#1986836, BZ#1481850
gdb	BZ#1870029, BZ#1870031
gfs2-utils	BZ#1616432
gimp	BZ#2047161

Component	Tickets
git	BZ#1956345
glibc	BZ#2023422, BZ#2024347
gnome-shell-extension-background-logo	BZ#2057150
gnome-shell-extensions	BZ#2031186
gnupg2	BZ#2070722, BZ#2073567
gnutls	BZ#2033220, BZ#1999639
golang	BZ#2014087, BZ#1984110
grafana-pcp	BZ#1993156, BZ#1845592
grafana	BZ#1993215
grub2	BZ#2026579
grubby	BZ#1969362
hostapd	BZ#2019830
ipa	BZ#1952028, BZ#1957736, BZ#1966101, BZ#1988383, BZ#2084180, BZ#2084166, BZ#2057471
iptables	BZ#1945151
javapackages-tools	BZ#1951482
jigawatts	BZ#1972029
jmc-core	BZ#1980981
kdump-anaconda-addon	BZ#1894783, BZ#2017401
kernel-rt	BZ#2002474

Component	Tickets
kernel	BZ#1844416, BZ#1851933, BZ#1780258, BZ#1874195, BZ#1953515, BZ#1960556, BZ#1948340, BZ#1952863, BZ#1978382, BZ#1957818, BZ#2002499, BZ#2050415, BZ#1951951, BZ#1949613, BZ#2036856, BZ#2034490, BZ#1943423, BZ#2054441, BZ#2046472, BZ#2068432, BZ#1997541, BZ#1613522, BZ#1874182, BZ#1995338, BZ#1570255, BZ#2023416, BZ#2021672, BZ#2019593, BZ#2000616, BZ#2013650, BZ#2033080, BZ#2069758, BZ#2059545, BZ#2030922, BZ#1945040, BZ#2073541, BZ#1960467, BZ#2005173
kexec-tools	BZ#1988894, BZ#1895232, BZ#1958452, BZ#2065013
kmod	BZ#1985100
krb5	BZ#2060798, BZ#2068935, BZ#2077450
libburn	BZ#2015861
libcap	BZ#2037215
libgcrypt	BZ#1990059
libmodulemd	BZ#1984403
libreswan	BZ#2017355, BZ#2039877
libseccomp	BZ#2019887
libservicelog	BZ#1869568
libvirt	BZ#2014487
libxcrypt	BZ#2034569
llvm-toolset	BZ#2001107
lorax-templates-rhel	BZ#1961092
lsvpd	BZ#1869564
lvm2	BZ#1899214, BZ#1749513, BZ#2038183
mariadb	BZ#1971248
mod_security_crs	BZ#1947962

Component	Tickets
nettle	BZ#1986712
nfs-utils	BZ#2059245
nginx	BZ#1953639
nmstate	BZ#1969941
nodejs	BZ#1953491
nss	BZ#2008320, BZ#2099438
numatop	BZ#1874125
nvml	BZ#1874208
opal-prd	BZ#1869560
open-vm-tools	BZ#2037657
opencryptoki	BZ#1869533
openscap	BZ#2041782
openssh	BZ#1952957, BZ#2002734, BZ#1821501, BZ#2087121
openssl	BZ#1990814, BZ#1871147, BZ#1970388, BZ#1975836, BZ#1681178, BZ#1685470, BZ#2053289, BZ#2087253, BZ#2060044, BZ#2071631
osbuild-composer	BZ#2060575
oscap-anaconda-addon	BZ#1893753
ostree	BZ#1961254
p11-kit	BZ#1966680
pacemaker	BZ#1850145, BZ#1443666, BZ#1470834, BZ#1082146, BZ#1376538, BZ#1975388
pcp	BZ#1991764, BZ#1847808, BZ#1981223
pcs	BZ#1290830, BZ#1909901, BZ#1872378, BZ#2018969, BZ#1996067

Component	Tickets
perl-Module-Signature	BZ#2039361
php	BZ#1949319
pki-core	BZ#2084181
podman	JIRA:RHELPLAN-77549, JIRA:RHELPLAN-75322, JIRA:RHELPLAN-108830, BZ#2027576
powerpc-utils	BZ#1873868
ppc64-diag	BZ#1869567
python-jsonpointer	BZ#1980256
python-podman	BZ#1975462
qemu-kvm	BZ#1940132, BZ#1939509, JIRA:RHELPLAN-75866, BZ#1874187, BZ#1965079, BZ#1951814, BZ#2014229, BZ#2052424, BZ#1817965
redis	BZ#1959756
rhel-system-roles	BZ#1993304, BZ#1993377, BZ#2022461, BZ#1978488, BZ#1984583, BZ#2016517, BZ#2021667, BZ#1986460, BZ#1978752, BZ#1978753, BZ#1990490, BZ#2031555, BZ#2016518, BZ#2054364, BZ#1978773, BZ#2054435, BZ#1999162, BZ#2057657, BZ#2012298, BZ#2021028, BZ#2054367, BZ#2054369, BZ#2057662, BZ#2021665, BZ#2029427, BZ#2004899, BZ#1958964, BZ#1978734, BZ#1978760, BZ#2039106, BZ#2041632, BZ#2058777, BZ#2058645, BZ#2058756, BZ#2071804, BZ#2029634, BZ#2044408, BZ#2029602, BZ#2038957, BZ#2064391, BZ#2004303, BZ#2006230, BZ#2057164, BZ#2021025, BZ#2021676, BZ#2047506, BZ#2050341, BZ#2050419, BZ#1999770
rpm-ostree	BZ#1961324
rpm	BZ#1942549, BZ#1962234
rsyslog	BZ#2027971, BZ#1992155
rust-toolset	BZ#2002885
s390utils	BZ#1932480
samba	BZ#2013578, Jira:RHELDOCS-16612
scap-security-guide	BZ#2028435, BZ#2014561, BZ#2045341, BZ#2038978

Component	Tickets
selinux-policy	BZ#2055822, BZ#1932752, BZ#2021529, BZ#2064274
shadow-utils	BZ#1859252
sos	BZ#2011537, BZ#1869561
squid	BZ#1990517
sssd	BZ#1949149, BZ#2014249, BZ#1879869, BZ#1737489
strace	BZ#2038965
stratisd	BZ#2041558
stunnel	BZ#2039299
subscription-manager	BZ#1898563, BZ#2049441
sudo	BZ#1981278
swig	BZ#1943580
systemd	BZ#2018112
systemtap	BZ#2083727
tigervnc	BZ#2060308
trace-cmd	BZ#1933980
tuned	BZ#2003838
unbound	BZ#2070495
usbguard	BZ#1986785, BZ#2009226
varnish	BZ#1984185
virt-manager	BZ#1995131
virt-who	BZ#2008215, BZ#2054504
virtio-win	BZ#1969724
wpa_supplicant	BZ#1975718

Component	Tickets
other	BZ#2077836, BZ#2019806, BZ#1937651, BZ#2010291, BZ#1941810, BZ#2091643, BZ#1941595, JIRA:RHELPLAN-80758, JIRA:RHELPLAN-80759, JIRA:RHELPLAN-82578, JIRA:RHELPLAN-68364, JIRA:RHELPLAN-78673, JIRA:RHELPLAN-78675, BZ#1940863, BZ#2079313, JIRA:RHELPLAN-100497, BZ#2068532, BZ#2089193, JIRA:RHELPLAN-102009, BZ#2065646, BZ#2088414, JIRA:RHELPLAN-80734, BZ#2013853, JIRA:RHELPLAN-103540, BZ#2019341, BZ#2008558, BZ#2008575, BZ#2009455, JIRA:RHELPLAN-74542, JIRA:RHELPLAN-73678, JIRA:RHELPLAN-84168, JIRA:RHELPLAN-73697, JIRA:RHELPLAN-95126, BZ#2080875, JIRA:RHELPLAN-97899, JIRA:RHELPLAN-100359, JIRA:RHELPLAN-103147, JIRA:RHELPLAN-103146, JIRA:RHELPLAN-79161, BZ#2046325, BZ#2021262, JIRA:RHELPLAN-64576, JIRA:RHELPLAN-65223, BZ#2083036, BZ#2011448, BZ#2019318, JIRA:RHELPLAN-101240, JIRA:RHELPLAN-101241, JIRA:RHELPLAN-101242, JIRA:RHELPLAN-101246, JIRA:RHELPLAN-101247, JIRA:RHELPLAN-102552, JIRA:RHELPLAN-99892, BZ#2027596, JIRA:RHELPLAN-119000, BZ#1940653, JIRA:RHELPLAN-95056, BZ#2054401, JIRA:RHELPLAN-113994, BZ#2059183, JIRA:RHELPLAN-74543, JIRA:RHELPLAN-99889, JIRA:RHELPLAN-99890, JIRA:RHELPLAN-100032, JIRA:RHELPLAN-100034, JIRA:RHELPLAN-101141, JIRA:RHELPLAN-100020, BZ#2069501, BZ#2070506, JIRA:RHELPLAN-117903, JIRA:RHELPLAN-98617, JIRA:RHELPLAN-103855, BZ#2091653, BZ#2082306, JIRA:RHELPLAN-65217, BZ#2020529, BZ#2030412, BZ#2046653, JIRA:RHELPLAN-103993, JIRA:RHELPLAN-122345, BZ#1927780, JIRA:RHELPLAN-110763, BZ#1935544, BZ#2089200, JIRA:RHELPLAN-15509, JIRA:RHELPLAN-99136, JIRA:RHELPLAN-103232, BZ#1899167, BZ#1979521, JIRA:RHELPLAN-100087, JIRA:RHELPLAN-100639, JIRA:RHELPLAN-10304, BZ#2058153, JIRA:RHELPLAN-113995, JIRA:RHELPLAN-121048, JIRA:RHELPLAN-98983, BZ#1640697, BZ#1697896, BZ#2020026, BZ#2047713, JIRA:RHELPLAN-109067, JIRA:RHELPLAN-115603, JIRA:RHELPLAN-96940, JIRA:RHELPLAN-117234, JIRA:RHELPLAN-119001, JIRA:RHELPLAN-119852, BZ#2077767, BZ#2053598, JIRA:RHELPLAN-121180, BZ#2082303, JIRA:RHELPLAN-121049

APPENDIX B. ACKNOWLEDGEMENTS

Thank you to the below Red Hat Associates who provided feedback as part of the RHEL 9 Readiness Challenge:

- Buland Singh
- Pradeep Jagtap
- Omkar Andhekar
- Ju Ke
- Suresh Jagtap
- Prijesh Patel
- Nikhil Suryawanshi
- Amit Yadav
- Pranav Lawate
- John Pittman

APPENDIX C. REVISION HISTORY

0.1-30

Tue Jun 11 2024, Brian Angelica (bangelic@redhat.com)

- Add Deprecated Functionality RHELDOCS-18049 (Shells and command-line tools).

0.1-29

Tue Jun 11 2024, Brian Angelica (bangelic@redhat.com)

- Added an Known Issue JIRA:RHEL-24847 (Shells and command-line tools).

0.1-28

Thu Mar 14 2024, Gabriela Fialová (gfialova@redhat.com)

- Added a Known Issue JIRA:RHEL-25967 (Kernel)

0.1-27

Wed Feb 14 2024, Gabriela Fialová (gfialova@redhat.com)

- Added an Enhancement JIRA:RHELDOCS-17553 (Identity Management)

0.1-26

Thu Feb 1 2024, Gabi Fialova (gfialova@redhat.com)

- Added a KI BZ#1834716 (Security)

0.1-25

Mon Nov 13 2023, Gabriela Fialová (gfialova@redhat.com)

- Added a Tech Preview JIRA:RHELDOCS-17040 (Virtualization)

0.1-24

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

- Updated the module on Providing Feedback on RHEL Documentation.

0.1-23

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

- Added a Tech Preview JIRA:RHELDOCS-17050 (Virtualization).

0.1-22

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

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

0.1-21

September 8 2023, Marc Muehlfeld (mmuehlfeld@redhat.com)

- Added a deprecated functionality release note JIRA:RHELDOCS-16612 (Samba).

- Updated "Providing feedback on Red Hat documentation" to reflect RHEL in JIRA.

0.1-20

August 17 2023, Gabi Fialova (gfialova@redhat.com)

- Added an Enh BZ#2136937 (Plumbers).

0.1-19

August 07 2023, Gabi Fialova (gfialova@redhat.com)

- Added a KI BZ#2214130 (CS).

0.1-18

August 02 2023, Marc Muehlfeld (mmuehlfeld@redhat.com)

- Updated a deprecated functionality release note BZ#1894877 (NetworkManager).

0.1-17

Mon Jun 19, 2023, Gabi Fialova (gfialova@redhat.com)

- Fixed typo in KI BZ#2068935 (IdM).

0.1-16

Thu May 18, 2023, Gabi Fialova (gfialova@redhat.com)

- Added an Enhancement BZ#2053642 (Filesystem and storage).

0.1-15

Wed May 17, 2023, Gabi Fialova (gfialova@redhat.com)

- Update deprecated-packages.adoc with info about EOL.

0.1-14

Thu May 11, 2023, Gabi Fialova (gfialova@redhat.com)

- Added an Enhancement BZ#2190045 (Installer).

0.1-13

Thu Apr 27, 2023, Gabi Fialova (gfialova@redhat.com)

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

0.1-12

Thu Apr 13, 2023, Gabi Fialova (gfialova@redhat.com)

- Fix a broken link in a new feature JIRA:RHELPLAN-84168 (Containers).

0.1-11

Wed Mar 1, 2023, Gabi Fialova (gfialova@redhat.com)

- Modified doc text for BZ#2091643 (Kernel).

0.1-10

Mon Feb 20, 2023, Gabi Fialova (gfialova@redhat.com)

- Added information into "In-place upgrade from RHEL 8 to RHEL 9" about SAP Environments.

0.1-9

Wed Jan 18, 2023, Gabi Fialova (gfialova@redhat.com)

- Added a Known Issue doc text BZ#2083106 (Kernel).

0.1-8

Tue Jan 17, 2023, Gabi Fialova (gfialova@redhat.com)

- Updated a Tech Preview doc text BZ#2084181 (Identity Management).

0.1-7

Mon Jan 16, 2023, Gabi Fialova (gfialova@redhat.com)

- Added a Known Issue doc text BZ#2149172 (Kernel).

0.1-6

Thu Dec 22, 2022, Gabi Fialova (gfialova@redhat.com)

- Updated a Known Issue doc text BZ#1960467 (Graphics Infrastructures).

0.1-5

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

- Added a Known Issue BZ#2151040 (Networking).

0.1-4

Tue Nov 15, 2022, Gabriela Fialová (gfialova@redhat.com)

- Updated the In-place upgrade section.

0.1-3

Fri Sep 23, 2022, Gabriela Fialová (gfialova@redhat.com)

- Added a deprecated functionality BZ#2074598 (Kernel).

0.1-2

Wed Sep 21, 2022, Gabriela Fialová (gfialova@redhat.com)

- Removed a known issue BZ#2060798 (Identity Management).
- Added a bug fix BZ#2060798 (Identity Management).

0.1-1

Mon Sep 12, 2022, Gabriela Fialová (gfialova@redhat.com)

- Updated proc_providing-feedback-on-red-hat-documentation.adoc.

- Added an enhancement BZ#2119694 (Security).

0.1-0

Mon Aug 22, 2022, Lenka Špačková (lspackova@redhat.com)

- Added deprecated functionality BZ#2069279 and BZ#2106816 (Containers).
- Updated JIRA-RHELPLAN-121180 with information about a z-stream fix (Containers).

0.0-9

Wed Aug 10, 2022, Lenka Špačková (lspackova@redhat.com)

- Added a known issue BZ#1991500 (Dynamic programming languages, web and database servers).

0.0-8

Thu Aug 4, 2022, Gabriela Fialová (gfialova@redhat.com)

- Added an enhancement JIRA-RHELPLAN-118914 (Containers).
- Added a known issue BZ#2105162 (Security).
- Added a known issue BZ#1960467 (Graphics infrastructures).

0.0-7

Thu Jul 28, 2022, Lenka Špačková (lspackova@redhat.com)

- Added an enhancement BZ#2099438 (Security).
- Added a known issue BZ#2087253 (Security).
- Extended information about Application Streams in Distribution.

0.0-6

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

- Added a known issue BZ#2077450.
- Added an enhancement BZ#2091653.
- Added a bug fix BZ#2006230.

0.0-5

Wed Jun 29, 2022, Lenka Špačková (lspackova@redhat.com)

- Added known issues BZ#2087121, BZ#2073567, BZ#2083727, and BZ#2005173.
- Added an enhancement BZ#2091643.

0.0-4

Wed Jun 1, 2022, Gabriela Fialová (gfialova@redhat.com)

- Added a known issue BZ#2027576.

0.0-3

Tue May 24, 2022, Gabriela Fialová (gfialova@redhat.com)

- Updated the list of top ten popular Customer Portal Labs.
- Added and republished deprecated functionality BZ#2089200 (Networking).

0.0-2

Wed May 18, 2022, Gabriela Fialová (gfialova@redhat.com)

- Release of the Red Hat Enterprise Linux 9.0 Release Notes.

0.0-1

Wed Nov 03, 2021, Lenka Špačková (lspackova@redhat.com)

- Release of the Red Hat Enterprise Linux 9.0 Beta Release Notes.