Red Hat Enterprise Linux 9.0 Beta

9.0 Release Notes

Release Notes for Red Hat Enterprise Linux 9.0 Beta

Last Updated: 2021-11-24

Red Hat Enterprise Linux 9.0 Beta 9.0 Release Notes

Release Notes for Red Hat Enterprise Linux 9.0 Beta

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 Beta and document known problems in this release, as well as notable bug fixes, Technology Previews, deprecated functionality, and other details.

Table of Contents

RHEL BETA RELEASE

Red Hat provides Red Hat Enterprise Linux Beta access to all subscribed Red Hat accounts. The purpose of Beta access is to:

- Provide an opportunity to customers to test major features and capabilities prior to the general availability release and provide feedback or report issues.
- Provide Beta product documentation as a preview. Beta product documentation is under development and is subject to substantial change.

Note that Red Hat does not support the usage of RHEL Beta releases in production use cases. For more information, see What does Beta mean in Red Hat Enterprise Linux and can I upgrade a RHEL Beta installation to a General Availability (GA) release?.

MAKING OPEN SOURCE MORE INCLUSIVE

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

PROVIDING FEEDBACK ON RED HAT DOCUMENTATION

We appreciate your input on our documentation. Please let us know how we could make it better. To do so:

- For simple comments on specific passages, make sure you are viewing the documentation in the Multi-page HTML format. Highlight the part of text that you want to comment on. Then, click the **Add Feedback** pop-up that appears below the highlighted text, and follow the displayed instructions.
- For submitting more complex feedback, create a Bugzilla ticket:
  1. Go to the Bugzilla website.
  2. As the Component, use **Documentation**.
  3. Fill in the **Description** field with your suggestion for improvement. Include a link to the relevant part(s) of documentation.
  4. Click **Submit Bug**.

CHAPTER 1. OVERVIEW

1.1. MAJOR CHANGES IN RHEL 9.0 BETA

Security

OpenSSL is now provided in version 3.0.0-0.beta2, 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 8.6p1, which provides many enhancements, bug fixes, and security improvements as compared to version 8.0p1, which is distributed in RHEL 8.5.

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.

The **scap-security-guide** packages are provided in version 0.1.57, which introduces substantial changes over the versions provided in RHEL 8.

See Section 4.7, "Security" for more information.

Use of **SHA-1** is restricted in the **DEFAULT** crypto policy. With the exception of HMAC and DNSSec usage, SHA-1 is no longer allowed in TLS, DTLS, SSH, IKEv2 and Kerberos protocols.

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 for more information about security-related major differences between RHEL 9 and RHEL 8.

Networking

The WireGuard VPN technology is now available as an unsupported Technology Preview.

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.

You can use the new MultiPath TCP daemon (mptcpd) to configure MultiPath TCP (MPTCP) endpoints without using the **iproute2** utility.

The **network-scripts** package has been removed. Use NetworkManager to configure network connections.

By default, NetworkManager now uses the key file format 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 Section 4.8, “Networking”.

Dynamic programming languages, web and database servers

RHEL 9.0 Beta provides the following dynamic programming languages:

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

RHEL 9.0 Beta includes the following version control systems:

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

The following web servers are distributed with RHEL 9.0 Beta:

- **Apache HTTP Server 2.4**
- **nginx 1.20**

The following proxy caching servers are available:

- **Varnish Cache 6.5**
- **Squid 5.1**

RHEL 9.0 Beta offers the following database servers:

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

See Section 4.11, “Dynamic programming languages, web and database servers” for more information.

Compilers and development tools

Updated compiler toolsets

The following compiler toolsets are available with RHEL 9.0 Beta:

- **LLVM Toolset 12.0.1**
- **Rust Toolset 1.54.0**
- **Go Toolset 1.16.6**

Updated system toolchain

The following system toolchain components are available with RHEL 9.0 Beta:

- GCC 11.2
- glibc 2.34
- binutils 2.35

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

- GDB 10.2
- Valgrind 3.17.0
- SystemTap 4.5
- Dyninst 11.0.0
- elfutils 0.185

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

- Maven 3.6
- Ant 1.10

See Section 4.12, "Compilers and development tools" for more information.

Virtualization

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).

For more information about virtualization features introduced in this release, see Section 4.16, "Virtualization".

1.2. RED HAT CUSTOMER PORTAL LABS

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

- Registration Assistant
- Product Life Cycle Checker
- Kickstart Generator
- Kickstart Converter
- Red Hat Enterprise Linux Upgrade Helper
- Red Hat Satellite Upgrade Helper
- Red Hat Code Browser

- JVM Options Configuration Tool
- Red Hat CVE Checker
- Red Hat Product Certificates
- Load Balancer Configuration Tool
- Yum Repository Configuration Helper
- Red Hat Memory Analyzer
- Kernel Oops Analyzer
- Red Hat Product Errata Advisory Checker

1.3. ADDITIONAL RESOURCES

- Information regarding the Red Hat Enterprise Linux **life cycle** is provided in the Red Hat Enterprise Linux Life Cycle document.
- Major **differences between RHEL 8 and RHEL 9** are documented in Considerations in adopting RHEL 9.
- The **Red Hat Insights** service, which enables you to proactively identify, examine, and resolve known technical issues, is now available with all RHEL subscriptions. For instructions on how to install the Red Hat Insights client and register your system to the service, see the Red Hat Insights Get Started page.

CHAPTER 2. ARCHITECTURES

Red Hat Enterprise Linux 9.0 Beta is distributed with the kernel version 5.14, which provides support for the following architectures:

- 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 **Red Hat Customer Portal 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 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 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.

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.

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.

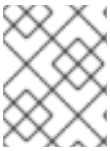
Each Application Stream component has a given life cycle, either the same as RHEL 9 or shorter.

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

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. Detailed module commands are described in the Managing software with YUM document.

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.



NOTE

Application Streams versions and formats distributed in RHEL 9.0 Beta might differ from versions and formats provided at the time of general availability of RHEL 9.0.

3.4. PACKAGE MANAGEMENT WITH YUM/DNF

Throughout this document, **YUM** and **DNF** can be used interchangeably.

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 YUM.

CHAPTER 4. NEW FEATURES

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

4.1. INSTALLER AND IMAGE CREATION

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 cart for sudo is not supported.

(JIRA:RHELPLAN-95126)

Anaconda supports the rhsm command for machine provisioning via Kickstart installations for satellite

Previously, machine provisioning was dependent on a custom **%post** script for Kickstart installation on Red Hat Satellite. This **%post** script imports the custom satellite self-signed certificate, registers the machine, attaches a subscription and installs packages residing in repositories.

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

(BZ#1951709)

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 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.

```
# yum 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.

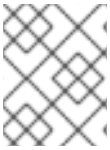
For kickstart installations, add initial-setup-gui to the packages section and enable the initial-setup service.

```
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
```

(BZ#2011448)

Image Builder now supports cross-version image building

With this enhancement, you can use Image Builder to create images of multiple RHEL minor releases that are different from the host, such as RHEL 8.4 and RHEL 8.5. As a result, you can avoid maintaining multiple Image Builder instances.

(BZ#2019003)

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

rpm-ostree rebased to version v2021.5

The **rpm-ostree** package has been upgraded to version v2021.5, 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)

OSTree rebased to version v2021.2

The **OSTree** package has been upgraded to 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)

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 **subscriptions-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

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 YUM plugin because its functionality is not limited to YUM transactions but covers also changes by RPM.

(BZ#1942549)

libmodulemd rebased to version 2.13.0

The **libmodulemd** packages have been rebased to version 2.13.0, which provides the following notable changes over the previous version:

- 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)

RPM rebased to version 4.16

RPM packages have been rebased to version 4.16, which provides the following notable changes:

- New SPEC features, most notably:
  - Fast macro-based dependency generators
  - The **%generate_buildrequires** section that allows for generating dynamic build dependencies
  - Meta (unordered) dependencies
  - 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
- The rpm database is now based on the **sqlite** library. Read-only support for Berkeley DB 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
- Increased parallelism in package builds
- Enforced UTF-8 validation of header data at build-time

(JIRA:RHELPLAN-70122)

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)

4.5. SHELLS AND COMMAND-LINE TOOLS

powerpc-utils rebased to version 1.3.9

powerpc-utils package has been upgraded to version 1.3.9. Notable bug fixes and enhancements 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)

ppc64-diag rebased to version 2.7.7

ppc64-diag package has been upgraded to version 2.7.7. Notable bug fixes and enhancements 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)

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)

mod_security_crs rebased to version 3.3

mod_security_crs has been upgraded to 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 via 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)

chrony rebased to version 4.1

chrony has been updated to version 4.1. Notable bug fixes and enhancements include:

- Added support for Network Time Security (NTS) authentication.
- By default, the Authenticated Network Time Protocol (NTP) sources are trusted over non-authenticated NTP sources. Add the **autselectmode ignore** argument in the **chrony.conf** file to restore the original behavior.
- 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.

(BZ#1961131)

4.7. SECURITY

OpenSSL now includes providers

The OpenSSL toolkit in version 3.0.0-0.beta2, which is included in RHEL 9 Beta, 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, if the **openssl.cnf** configuration file does not contain a specific provider, 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)

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). SHA-1 is also allowed for DNSSEC in the DEFAULT and LEGACY policy levels.

If your scenario requires enabling some of the disabled algorithms and ciphers, use policy modifiers or customize the policy.

(BZ#1937651)

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.

WARNING

The VPN System Role does not work correctly with the **ansible-core 2.11** package that is provided in RHEL 9.0 Beta. For more information, see Some RHEL System Roles do not work with the **ansible-core 2.11** package.

(BZ#2019341)

OpenSSL provided in version 3.0.0-0.beta2

RHEL 9 Beta provides **openssl** packages in upstream version 3.0.0-0.beta2, 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#1903209)

GnuTLS provided in version 3.7.2

In RHEL 9 Beta, the **gnutls** packages are provided in upstream version 3.7.2. This provides many improvements and bug fixes over previous versions, most notably:

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

(BZ#1966479)

OpenSSH distributed in 8.6p1

RHEL 9 Beta includes **OpenSSH** in version 8.6p1. This version provides many enhancements and bug fixes over **OpenSSH** version 8.0p1, which is distributed in RHEL 8.5, most notably:

New Features

- 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)

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)

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)

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)

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)

Notable changes in scap-security-guide

RHEL 9 Beta includes the **scap-security-guide** packages in version 0.1.57. This version introduces the following major changes over the version available in RHEL 8:

- The ***-cpe-dictionary.xml**, ***-cpe-oval.xml**, ***-ocil.xml**, ***-oval.xml**, and ***-xccdf.xml** SCAP component files have been removed to avoid data duplication and to reduce the package size.
- Removed the Legacy SCAP 1.2 source data streams.
- Removed the Bash Profile Remediation scripts to encourage safer practices. The preferred way to remediate a hardening profile is to use the **oscap xccdf eval --remediate** command that executes only the needed remediations.
- Includes only RHEL 9 content. If you need to scan systems with different versions of RHEL, use the packages that are provided for the scanned systems.

(BZ#1962564)

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)

sudo supports Python plugins

With the **sudo** program version 1.9, which is included in RHEL 9-beta, 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)

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)

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)

Making Nmstate more inclusive

Red Hat is committed to using conscious language. See details about this initiative in Making open source more inclusive. Therefore the **slave** term in the **nmstate** API has been replaced by the term **port**.

(BZ#1969941)

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-docs** package.

(BZ#1949127)

4.9. KERNEL

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)

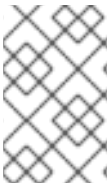
cgroup-v2 enabled by default in RHEL 9

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
```



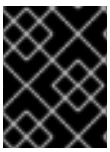
NOTE

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)

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 disabled several settings of the ftrace framework that were enabled by default. Most notably being the **tracing_on**, **trace_clock**, **set_event_pid**, and **tracing_max_latency** configurations. The new behavior of **trace-cmd reset** is to reset the mentioned configurations to their default values.

(BZ#1933980)

A new crashkernel.default file for kdump memory allocation

A new implementation of the **crashkernel.default** file is now available on the RHEL 9 version of **kdump**.

The **crashkernel.default** file is shipped with each kernel and it contains the default crash kernel value for the corresponding kernel build. The default value is used by **kdump** to control the default crash kernel memory value of each kernel. The value forms a good reference for **kdump** memory reservation. Using this value as the base to estimate the required memory, you can configure the desired **crashkernel=** value.

As a result, this improves the memory allocation for **kdump** when a system has less than 4 GB available memory.

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

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

(BZ#1942398)

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, v5.14-rt15. These updates provide a number of bug fixes and enhancements.

(BZ#1891873)

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)

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)

4.10. 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)

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 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)

4.11. 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 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 9.2.
- The **npm** package manager has been upgraded to version 7.20.3.
- A new **Timers Promises** API provides an alternative set of timer functions that return **Promise** objects.
- **Node.js** now provides a new experimental **Web Streams** API.
- **Node.js** is now compatible with **OpenSSL 3.0**.

Node.js 16 is the initial version of this Application Stream, which you can install easily as an RPM package. Additional **Node.js** versions will be provided as modules 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 3.0.2**, 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 available in RHEL 8.

Notable enhancement include:

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

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 introduces PHP 8.0

RHEL 9 is distributed with **PHP 8.0**, which provides a number of bug fixes and enhancements over version 7.4 available in RHEL 8.

Notable enhancements include:

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

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 Beta provides version 2.4.48 of the Apache HTTP Server. Notable changes over version 2.4.37 distributed with RHEL 8 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.
- 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.

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)

Varnish Cache 6.5 in RHEL 9

RHEL 9 includes **Varnish Cache 6.5**, a high-performance HTTP reverse proxy. This release provides a number of bug fixes and enhancements over version 6.0 available in RHEL 8.

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 5.1**, 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 available in RHEL 8.

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.

(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. Additional **Redis** versions will be provided as modules with a shorter life cycle in future minor releases of RHEL 9.

(BZ#1959756)

4.12. COMPILERS AND DEVELOPMENT TOOLS

GCC 11.2 is available

RHEL 9 Beta is distributed with GCC version 11.2. 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:
  - 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.

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

- The following Intel CPUs are supported: Sapphire Rapids, Alder Lake, and Rocket Lake.
- 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.

(BZ#1986836)

GCC defaults to IBM z14

RHEL 9 Beta is distributed with GCC 11.2 that defaults to the IBM z14 processor.

(BZ#1870016)

GCC defaults to IBM POWER9

RHEL 9 Beta is distributed with GCC 11.2 that defaults to the IBM POWER9 processor.

(BZ#1870028)

Link time optimization in GCC

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.

(BZ#2019811)

Updated performance tools and debuggers

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

- GDB 10.2
- Valgrind 3.17.0
- SystemTap 4.5
- Dyninst 11.0.0

- elfutils 0.185

(BZ#2019806)

DAWR functionality improved in GDB on IBM POWER10

RHEL 9 Beta 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)

Notable changes in LLVM Toolset 12.0.1

RHEL 9 Beta is distributed with LLVM Toolset 12.0.1. Notable changes include:

- The new compiler flag **-march=x86-64-v[234]** has been added.
- The compiler flag **-fasynchronous-unwind-tables** of the **clang** compiler is now the default on Linux AArch64/PowerPC.
- The **clang** compiler now supports the C++20 likelihood attributes [[likely]] and [[unlikely]].
- The new function attribute **tune-cpu** has been added. It allows microarchitectural optimizations to be applied independently from the **target-cpu** attribute or TargetMachine CPU.
- The new sanitizer **-fsanitize=unsigned-shift-base** has been added to the integer sanitizer **-fsanitize=integer** to improve security.
- Code generation on PowerPC targets has been optimized.
- The WebAssembly backend is now enabled in LLVM. With this enhancement, you can generate WebAssembly binaries with LLVM and Clang.

For more information, see Using LLVM Toolset.

(BZ#1931726)

Notable changes in CMake 3.20.2

RHEL 9 Beta 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)

Notable changes in Rust Toolset 1.54.0

RHEL 9 Beta is distributed with Rust Toolset 1.54.0. Notable changes include:

- The Rust standard library is now available for the **wasm32-unknown-unknown** target. With this enhancement, you can generate WebAssembly binaries, including newly stabilized intrinsics.
- It is now possible to use constant-value parameters to define generics. With this change, you can write functions completely generic over the values of any integer, boolean, or character type, and arrays generic over their element type as well as their length. Moreover, it is now possible to iterate items from an array by value using the new standard library's array type API **std::array::IntoIter**.
- Rust now includes the **IntoIterator** implementation for arrays. With this enhancement, you can use the **IntoIterator** trait to iterate over arrays by value and pass arrays to methods. However, **array.into_iter()** still iterates values by reference until the 2021 edition of Rust.
- The syntax for **or** patterns now allows nesting anywhere in the pattern. For example: **Pattern(1|2)** instead of **Pattern(1)|Pattern(2)**.
- Unicode identifiers can now contain all valid identifier characters as defined in the Unicode Standard Annex #31.
- Methods and trait implementations have been stabilized.

For more information, see Using Rust Toolset.

(BZ#1947202)

Notable changes in Go Toolset 1.16.6

RHEL 9 Beta is distributed with Go Toolset 1.16.6. Notable changes include:

- The **GO111MODULE** environment variable is now set to **on** by default. To revert this setting, change **GO111MODULE** to **auto**.
- The Go linker now uses less resources and improves code robustness and maintainability. The change applies to all supported CPU architectures and operating systems.
- With the new **embed** package you can access embedded files while compiling.
- All functions of the **io/ioutil** package have been moved to the **io** and **os** packages. While you can still use **io/ioutil**, the **io** and **os** packages provide better definitions.
- The Delve debugger has been rebased to 1.6.0 so that it supports Go Toolset 1.16.6.

For more information, see Using Go Toolset.

(BZ#1944737)

Go FIPS mode is supported with OpenSSL 3

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

(BZ#1984110)

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)

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)

pcp rebased to 5.3.1

The Performance Co-Pilot (PCP) package has been rebased to version 5.3.1. This release includes bug fixes, enhancements, and new features. The most notable changes include:

- Scalability improvements, which now support large number of hosts to have performance metrics centrally logged (**pmlogger** farms) and automatically monitored with performance rules (**pmie** farms).
- Supports the new **pcp-ss** tool for historical socket statistics.
- Improvements to the **pcp-htop** tool.
- Added extensions to the over-the-wire PCP protocol, which now support higher resolution timestamps.

(BZ#1957575)

grafana-pcp rebased to version 3.1.0

The **grafana-pcp** package has been rebased to version 3.1.0. The rebase provides following notable changes over previous version:

- Updated Performance Co-Pilot Vector Checklist dashboards to show new time series panel, display units in graphs, and update help texts.
- Added **pmproxy** URL and **hostspec** variables to Performance Co-Pilot Vector Host Overview and Performance Co-Pilot Checklist dashboards.
- Updated all dashboards to show **datasource** selection.
- Updated all dashboards as read only.
- Added compatibility with Grafana 8.

(BZ#1956385)

grafana rebased to version 7.5.9

The **grafana** package has been rebased to version 7.5.9. This rebase provides following notable changes over previous version:

- Supports the beta version of the new time series panel visualization.
- Supports the beta version of the new Pie chart panel visualization.
- Added alert support for Grafana Loki. It is a log aggregation tool.
- Added support for multiple new query transformations.

For more information, see: What's New in Grafana v7.4 and What's New in Grafana v7.5 .

(BZ#1956384)

python-jsonpointer rebased to version 2.0

The **python-jsonpointer** module has been updated to version 2.0.

Notable changes 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)

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)

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 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. 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)

4.13. IDENTITY MANAGEMENT

Improved the SSSD debug logging by adding a unique identifier tag for each request

As SSSD processes requests asynchronously, it is not easy to follow log entries for individual requests in the backend logs, as messages from different requests are added to the same log file. To improve the readability of debug logs, a unique request identifier is now added to log messages in the form of **RID#<integer>**. This allows you to isolate logs pertaining to an individual request, and you can track requests from start to finish across log files from multiple SSSD components.

For example, the following sample output from an SSSD log file shows the unique identifiers RID#3 and RID#4 for two different requests:

```
(2021-07-26 18:26:37): [be[testidm.com]] [dp_req_destructor] (0x0400): RID#3 Number of active DP request: 0
(2021-07-26 18:26:37): [be[testidm.com]] [dp_req_reply_std] (0x1000): RID#3 DP Request AccountDomain #3: Returning [Internal Error]: 3,1432158301,GetAccountDomain() not supported
(2021-07-26 18:26:37): [be[testidm.com]] [dp_attach_req] (0x0400): RID#4 DP Request Account #4: REQ_TRACE: New request. sssd.nss CID #1 Flags [0x0001].
(2021-07-26 18:26:37): [be[testidm.com]] [dp_attach_req] (0x0400): RID#4 Number of active DP request: 1
```

(JIRA:RHELPLAN-92473)

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 9, you can install **ansible-freeipa** without any preliminary steps. Installing **ansible-freeipa** automatically installs **ansible-core** as a dependency. Both packages are available in the **rhel-9-for-x86_64-appstream-rpms** repository.

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

(JIRA:RHELPLAN-100359)

IdM now supports the automember and server Ansible modules

With this update, the **ansible-freeipa** package contains the **ipaautomember** and **ipaserver** modules:

- Using the **ipaautomember** module, you can add, remove, and modify automember rules and conditions. As a result, future IdM users and hosts that meet the conditions will be assigned to IdM groups automatically.
- Using the **ipaserver** module, you can ensure various parameters of the presence or absence of a server in the IdM topology. You can also ensure that a replica is hidden or visible.

(JIRA:RHELPLAN-96640)

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)

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)

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)

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)

4.14. 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.

(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)

Boot loader menu hidden by default

The GRUB boot loader is now configured to hide the boot menu by default if RHEL is the only installed operating system and if the previous boot succeeded. This results in a smoother boot experience on such systems.

To access the boot menu, use one of the following options:

- Repeatedly press **Esc** after booting the system.
- Repeatedly press **F8** after booting the system.
- Hold **Shift** during boot.

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

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

(JIRA:RHELPLAN-101245)

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 on UEFI systems, is now a symbolic link to 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)

Langpacks replace comps language groups

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

In previous RHEL releases, language support was available from **comps** language groups. To enable support for a language, you previously installed the ***code*-support** package. The **langpacks-*code*** packages now replace the **comps** language groups.

(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)

4.15. RED HAT ENTERPRISE LINUX SYSTEM ROLES

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, e.g. 50% in addition to the human-readable size of the file system, for example, 10g, 50 GiB.

(BZ#1984583)

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)

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)

4.16. VIRTUALIZATION

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)

4.17. RHEL IN CLOUD ENVIRONMENTS

WALinuxAgent rebased to 2.3.0.2

The Windows Azure Linux Agent (WALinuxAgent) has been upgraded to upstream version 2.3.0.2, which introduces a number of bug fixes and enhancement. Most notably:

- Support for has been added RequiredFeatures and GoalStateAggregateStatus APIs.
- 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)

4.18. 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)

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)

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)

Updating container images with new packages

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

```
$ podman run -it registry.access.redhat.com/rhel9-beta
$ yum update -y && rm -rf /var/cache/yum
```

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

```
$ yum install <package>
```

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

Note that for RHEL 9 Beta, 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 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)

CHAPTER 5. BUG FIXES

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

5.1. INSTALLER AND IMAGE CREATION

The automatic partitioning can be scheduled in Anaconda

Previously, during automatic partitioning on LVM type disks, the installer tried to create a partition for an LVM PV on each selected disk. If these disks already had partitioning layout, the schedule of the automatic partitioning could have failed with the error message.

With this update, the problem has been fixed. Now you can schedule the automatic partitioning in the installer.

(BZ#1642391)

RHEL installer failed to start when InfiniBand network interfaces were configured using installer boot options

Previously, when you configured InfiniBand network interfaces at an early stage of RHEL installation using installer boot options (for example, downloaded installer image using PXE server), the installer failed to activate the network interfaces.

This issue occured because the RHEL NetworkManager failed to recognize the network interfaces in InfiniBand mode, and instead configured Ethernet connections for the interfaces.

As a result, connection activation failed, and if the connectivity over the InfiniBand interface was required at an early stage, RHEL installer failed to start the installation.

With this release, the installer successfully activates the InfiniBand network interfaces that you configure at an early stage of RHEL installation using installer boot options, and the installation completes successfully.

(BZ#1890009)

Configuring a wireless network using Anaconda GUI is fixed

Previously, configuring the wireless network while using Anaconda graphical user interface (GUI) caused the installation to crash.

With this update, the problem has been fixed. You can configure the wireless network during the installation while using Anaconda GUI.

(BZ#1847681)

Anaconda now shows a dialog for **ldl** or unformatted DASD disks in text mode

Previously, during an installation in text mode, Anaconda failed to show a dialog for Linux disk layout (**ldl**) or unformatted Direct-Access Storage Device (DASD) disks. As a result, users were unable to utilize those disks for the installation.

With this update, in text mode Anaconda recognizes **ldl** and unformatted DASD disks and shows a dialog where users can format them properly for the future utilization for the installation.

(BZ#1874394)

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. SHELLS AND COMMAND-LINE TOOLS

openCryptoki rebased to version 3.16.0

The **openCryptoki** package has been upgraded to version 3.16.0. Notable bug fixes and enhancements include:

- Improved the **protected-key** option and support for the **attribute-bound keys** in **EP11** core processor.
- Improved the import and export of secure key objects in the **cycle-count-accurate (CCA)** processor.

(BZ#1869533)

opal-prd rebased to version 6.7.1

The **opal-prd** package has been upgraded to version 6.7.1. Notable bug fixes and enhancements include:

- 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)

lsvpd rebased to version 1.7.22

The **lsvpd** package has been upgraded to version 1.7.22. Notable bug fixes and enhancements 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)

Red Hat Enterprise Linux 9 delivers an up-to-date modulemd-tools package

Previously, it was not possible to upgrade the **modulemd-tools** package from RHEL version 8 to version 9. The package is now upgraded to a new upstream version 0.9

(BZ#1946984)

libservicelog rebased to version 1.1.19

libservicelog has been upgraded to version 1.1.19. Notable bug fixes and enhancements include:

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

(BZ#1869568)

5.4. SECURITY

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, **yum** 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)

5.5. 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)

5.6. KERNEL

The makedumpfile utility now works as expected on a 52-bit virtual address on a 64-bit ARM architecture

Previously, the **makedumpfile** utility failed to create dump files on a 52-bit kernel virtual address on a 64-bit ARM architecture. As a consequence, the capture kernel failed to generate the **vmcore** image in the event of a kernel crash.

This update fixes the problem. As a result, **makedumpfile** can now generate **vmcore** files on a 52-bit virtual address on a 64-bit ARM architecture.

(BZ#1922023)

5.7. 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)

5.8. COMPILERS AND DEVELOPMENT TOOLS

-Wsequence-point warning behavior fixed

Previously, when 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, doesn't increase compilation time.

(BZ#1481850)

5.9. IDENTITY MANAGEMENT

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)

5.10. RED HAT ENTERPRISE LINUX SYSTEM ROLES

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)

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. 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 provides the interface for offloading TLS record encryption to Network Interface Controllers (NICs) that support 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, even if the **systemd** package provides **systemd-resolved**, this service is an unsupported Technology Preview.

(BZ#2020529)

6.2. KERNEL

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 supports SGX v1 and v1.5. The version 1 enables platforms using the **Flexible Launch Control** mechanism to use the SGX technology.

(BZ#1874182)

6.3. 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 a means for an application to directly map persistent memory into its address space. To use DAX, a system must have some form of persistent memory available, usually in the form of one or more Non-Volatile Dual In-line Memory Modules (NVDIMMs), and a file system that supports DAX 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)

6.4. RED HAT ENTERPRISE LINUX SYSTEM ROLES

HA Cluster RHEL System Role available as a Technology Preview

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

- Configuring nodes, fence device, resources, resource groups, and resource clones including meta attributes and resource operations
- Configuring cluster properties
- Configuring multi-link clusters
- Configuring custom cluster names and node names
- Configuring whether clusters start automatically on boot
- Configuring a basic corosync cluster and pacemaker cluster properties, stonith and resources.

The **ha_cluster** system role does not currently support constraints. Running the role after constraints are configured manually will remove the constraints, as well as any configuration not supported by the role.

The **ha_cluster** system role does not currently support SBD.

(BZ#1893743, BZ#1978726)

6.5. VIRTUALIZATION

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)

CHAPTER 7. DEPRECATED FUNCTIONALITY

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

Deprecated functionality continues to be supported until the end of life of 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.

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.

7.1. SECURITY

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)

7.2. 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.

For more information on migrating to nftables, see Migrating from iptables to nftables and 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.

For details about how to migrate a team to a bond, see Migrating a network team configuration to network bond.

(BZ#1935544)

NetworkManager stores new network configurations to /etc/NetworkManager/system-connections/ in a keyfile format

Previously, NetworkManager stored new network configurations to **/etc/sysconfig/network-scripts/** in the **ifcfg** format. Starting with RHEL 9.0, RHEL stores new network configurations at **/etc/NetworkManager/system-connections/** in a keyfile format. The connections for which the configurations are stored to **/etc/sysconfig/network-scripts/** in the old format still work uninterrupted. Modifications in existing profiles continue updating the older files.

(BZ#1894877)

7.3. 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)

7.4. IDENTITY MANAGEMENT

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)

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)

7.5. 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)

Creating internal snapshots of virtual machines has been deprecated

Due to their lack of optimization and stability, internal virtual machine snapshots are now deprecated. Instead, external snapshots are recommended for use.

(BZ#1621944)

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)

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)

7.6. 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)

7.7. DEPRECATED PACKAGES

The following packages have been deprecated and will probably not be included in a future major release of Red Hat Enterprise Linux:

- libdb
- mcpp
- python3-pytz

CHAPTER 8. KNOWN ISSUES

This part describes known issues in Red Hat Enterprise Linux 9.0 Beta.

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)

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 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)

Installing RHEL from the boot menu using basic graphics mode from the Troubleshooting submenu fails

Installation process may fail to enter in the basic graphics mode on the hardware with an unsupported graphic card or due to any issue in the graphic driver that is preventing starting the graphical interface.

To work around this problem and boot the installer:

- Using the text user interface, use the **inst.text** boot option.
- Using the graphical user interface via VNC, use the **inst.vnc** option.

(BZ#1961092)

8.2. SHELLS AND COMMAND-LINE TOOLS

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)

8.3. INFRASTRUCTURE SERVICES

bind does not allow the same zone file in multiple zones

bind-9.11.4-9 does not allow the same zone file in multiple zones. To use a zone file the **named** service modifies the file. Consequently, the named service would not start if a configuration includes multiple zones sharing the same file path. To workaround, this problem, use the **in-view** clause to share one zone between multiple views. For example, use different paths for different zones, and include view names in the path.

(BZ#1984982)

8.4. SECURITY

The OpenSSL TLS library does not detect if the PKCS#11 token supports creation of raw RSA or RSA-PSS signatures

The **TLS-1.3** protocol requires the support for **RSA-PSS** signature. If the **PKCS#11** token does not support **raw RSA** or **RSA-PSS** signatures, the server applications which use **OpenSSL TLS** library will fail to work with the **RSA** key if it is held by the **PKCS#11** token. As a result, **TLS** communication will fail.

To work around this problem, configure server or client to use the **TLS-1.2** version 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)

OpenSSH HW acceleration does not work on IBM Z

In RHEL 9.0 Beta, some applications do not use the engine specified in the OpenSSL configuration to perform hardware offload of cryptographic functions. For example, the OpenSSH tools do not use hardware acceleration through the IBMCA engine on the 64-bit IBM Z systems.

(BZ#2019369)

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)

usbguard-notifier logs too many error messages to the Journal

The **usbguard-notifier** service does not have inter-process communication (IPC) permissions for connecting to the **usbguard-daemon** IPC interface. Consequently, **usbguard-notifier** fails to connect to the interface, and it writes a corresponding error message to the Journal. Because **usbguard-notifier** starts with the **--wait** option, which ensures that **usbguard-notifier** attempts to connect to the IPC interface each second after a connection failure, by default, the log contains an excessive amount of these messages soon.

To work around the problem, allow a user or a group under which **usbguard-notifier** is running to connect to the IPC interface. For example, the following error message contains the UID and GID values for the GNOME Display Manager (GDM):

```
IPC connection denied: uid=42 gid=42 pid=8382, where uid and gid 42 = gdm
```

To grant the missing permissions to the **gdm** user, use the **usbguard** command and restart the **usbguard** daemon:

```
# usbguard add-user gdm --group --devices listen
# systemctl restart usbguard
```

After granting the missing permissions, the error messages no longer appear in the log.

(BZ#2009226)

8.5. NETWORKING

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)

8.6. KERNEL

modprobe fails to install some out-of-tree kernel modules

The **/etc/depmod.d/dist.conf** 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. Since **/etc/depmod.d/dist.conf** is missing due to some prior RHEL changes, **modprobe** cannot load some out-of-tree kernel modules. To work around this problem, provide a config file for your out-of-tree module, or install any out-of-tree kernel module in the **/lib/modules/$(uname -r)/updates/** directory instead of the **/lib/modules/$(uname -r)/extra/** directory.

(BZ#1985100)

RPM macros for building out-of-tree kernel module RPMs cause various problems

RPM macros for building out-of-tree kernel modules in RHEL 9 (including the **%kernel_module_package**) were broken due to several packaging changes in the kernel RPM.

The specific errors and their workarounds are detailed in Errors and workarounds when building out-of-tree kernel module RPMs using %kernel_module_package macros

As a result, it is possible to successfully build an out-of-tree kernel module RPM.

(BZ#1971748)

kdump fails to start on RHEL 9 kernel

The RHEL 9 kernel does not have the **crashkernel=auto** 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, execute:

```
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)

Audio devices that use the Use Case Manager configuration are not detected or they do not function properly

A bug in the **alsa-lib** package causes that the internal Use Case Manager (UCM) identifier is not correctly parsed. Consequently, some audio devices that use the Use Case configuration are not detected or they do not function properly. The problem occurs more when the **pipewire** sound service is used.

(BZ#2015863)

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)

8.7. FILE SYSTEMS AND STORAGE

System does not boot via iSCSI

The dependencies of the **iscsi-init.service** can create a deadlock that blocks running **iscsid** in an **initramfs** environment. Booting from iSCSI does not work because **iscsi-init** does not start and blocks **iscsid** from starting. As a consequence, no iSCSI sessions can be established in the **initramfs**.

To work around this problem, use the following steps:

1. Navigate to an **initramfs** emergency shell by adding **rd.break=initqueue** to the kernel command line in grub.
2. Verify if the **/etc/iscsi/initiatorname.iscsi** file exists. If it exists, **iscsi-init** is not required.
3. Manually start **iscsid** by executing the **/usr/sbin/iscsid** command as root.
4. Exit the shell, the system should continue to boot from iSCSI at this point.
5. Once the system has booted from iSCSI, edit the **/usr/lib/systemd/system/iscsi-init.service** file and add the "DefaultDependencies=no" in the [Unit] section.
   Do not use **systemctl edit** because it creates a new file in the **/etc/systemd** directory, while **dracut** will continue to the original one.
6. Rebuild the initramfs file by using the **dracut --rebuild** command as root.

You should now be able to successfully reboot using iSCSI

(BZ#2016482)

8.8. RED HAT ENTERPRISE LINUX SYSTEM ROLES

Some RHEL System Roles do not work with the ansible-core 2.11 package

The RHEL 9 Beta release includes the **ansible-core 2.11** package. This is a version of Ansible that has only the core functionality. That means that the modules such as **firewalld**, and plugins such as **json_query**, among many others. As a consequence, the following system roles will not work in RHEL 9 Beta with the **ansible-core 2.11** package:

- **ha_cluster**
- **kdump**
- **logging**
- **selinux**
- **storage**
- **timesync**
- **vpn**

Currently, there is no workaround available.

(JIRA:RHELPLAN-92523)

8.9. VIRTUALIZATION

RHEL 9 virtual machines cannot use DASD as virtio block storage on IBM Z

Currently, virtual machines (VMs) running RHEL 9 on IBM Z hardware are not able to use DASD storage devices attached with the virtio-blk driver. You should not upgrade your VMs to RHEL 9 Beta if you plan to use the described devices.

(BZ#2008401)

Hot-unplugging a mounted virtual disk sometimes causes the guest kernel to crash on IBM Z

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

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

When the kernel crashes, the VM automatically reboots. If you need to unplug the disk and controller fully, you can avoid the VM crashing by first shutting off the VM. In addition, remove the disk from the guest's **fstab** file if present in order to boot gracefully next time.

(BZ#1997541)

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, use a different connection protocol or a different installation source.

(BZ#2014229)

8.10. RHEL IN CLOUD ENVIRONMENTS

RHEL 9 VMs on Azure sometimes lose network connection

Currently, RHEL 9 virtual machines running on the Microsoft Azure hypervisor have problems with the ordering cycle after rebooting. This may cause certain services to terminate unexpectedly, including NetworkManager, which may in turn cause network disconnections. To work around the issue, restart the VM or access the serial console and start the NetworkManager service.

(BZ#1998445)

8.11. 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)

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
NetworkManager	BZ#1949127, BZ#1931284, BZ#1894877
WALinuxAgent	BZ#1972101
alsa-lib	BZ#2015863
anaconda	BZ#1951709, BZ#1978264, BZ#1642391, BZ#1890009, BZ#1847681, BZ#1874394, BZ#1947376, BZ#1877697, BZ#1914955, BZ#1929105, BZ#1983602, BZ#1997832
bind	BZ#1984982
chrony	BZ#1961131
clevis	BZ#1956760
cloud-init	BZ#1998445
cmake	BZ#1957948
containers-common	BZ#2020026
cyrus-sasl	BZ#1947971, BZ#1995600
distribution	BZ#1878583
edk2	BZ#1935497
fapolicyd	BZ#1932225
gcc	BZ#1986836, BZ#1870016, BZ#1870028, BZ#1481850
gdb	BZ#1870029, BZ#1870031
git	BZ#1956345
gnutls	BZ#1966479
go-toolset	BZ#1944737

Component	Tickets
golang	BZ#1984110
grafana-pcp	BZ#1845592, BZ#1956385
grafana	BZ#1956384
ipa	BZ#1952028, BZ#1957736, BZ#1966101, BZ#1988383
iptables	BZ#1945151
iscsi-initiator-utils	BZ#2016482
javapackages-tools	BZ#1951482
jigawatts	BZ#1972029
kdump-anaconda-addon	BZ#1894783
kernel-rt	BZ#1891873
kernel-srpm-macros	BZ#1971748
kernel	BZ#1844416, BZ#1851933, BZ#1780258, BZ#1874195, BZ#1953515, BZ#1960556, BZ#1948340, BZ#1942398, BZ#1978382, BZ#1957818, BZ#2002499, BZ#1613522, BZ#1874182, BZ#1995338, BZ#1570255, BZ#2000616, BZ#2013650, BZ#2008401, BZ#1997541
kexec-tools	BZ#1922023
kmod-kvdo	BZ#1949159
kmod	BZ#1985100
ksc	BZ#1923836
libmodulemd	BZ#1984403
libservicelog	BZ#1869568
llvm-toolset	BZ#1931726
lorax-templates-rhel	BZ#1961092
lsvpd	BZ#1869564

Component	Tickets
lvm2	BZ#1899214
mariadb	BZ#1971248
mod_security_crs	BZ#1947962
modulemd-tools	BZ#1946984
net-snmp	BZ#1964963
nginx	BZ#1953639
nmstate	BZ#1969941
nodejs	BZ#1953491
opal-prd	BZ#1869560
opencryptoki	BZ#1869533
openscap	BZ#1936619
openssh	BZ#1952957, BZ#2019369
openssl	BZ#1903209, BZ#1871147, BZ#1970388, BZ#1975836, BZ#1681178, BZ#1685470
oscap-anaconda-addon	BZ#1893753
ostree	BZ#1961254
pacemaker	BZ#1850145, BZ#1443666, BZ#1975388
pcp	BZ#1847808, BZ#1957575, BZ#1981223
pcs	BZ#1290830, BZ#1909901, BZ#1872378, BZ#1881064
php	BZ#1949319
podman	JIRA:RHELPLAN-77549
powerpc-utils	BZ#1873868
ppc64-diag	BZ#1869567

Component	Tickets
python-jsonpointer	BZ#1980256
python-podman	BZ#1975462
qemu-kvm	BZ#1940132, BZ#1939509, BZ#1621944, BZ#1965079, BZ#1951814, BZ#2014229
redis	BZ#1959756
rhel-system-roles	BZ#1978488, BZ#1984583, BZ#1986460, BZ#1978752, BZ#1978753, BZ#1958964, BZ#1978734, BZ#1978760, BZ#1893743
rpm-ostree	BZ#1961324
rpm	BZ#1942549, BZ#1962234
rsyslog	BZ#1992155
rust-toolset	BZ#1947202
s390utils	BZ#1932480
scap-security-guide	BZ#1962564
selinux-policy	BZ#1932752, BZ#2021529
shadow-utils	BZ#1859252
squid	BZ#1990517
sssd	BZ#1949149, BZ#1879869, BZ#1737489
subscription-manager	BZ#1898563
sudo	BZ#1981278
swig	BZ#1943580
systemd	BZ#2018112
trace-cmd	BZ#1933980
usbguard	BZ#1986785, BZ#2009226
varnish	BZ#1984185

Component	Tickets
virt-who	BZ#2008215
wpa_supplicant	BZ#1975718
other	BZ#2019811, BZ#2019806, BZ#1937651, BZ#1941810, BZ#2019341, BZ#1941595, JIRA:RHELPLAN-80758, JIRA:RHELPLAN-80759, JIRA:RHELPLAN-82578, JIRA:RHELPLAN-68364, JIRA:RHELPLAN-78673, JIRA:RHELPLAN-78675, BZ#1940863, JIRA:RHELPLAN-100497, BZ#2008558, BZ#2008575, BZ#2009455, JIRA:RHELPLAN-74542, JIRA:RHELPLAN-73678, JIRA:RHELPLAN-84168, JIRA:RHELPLAN-73697, JIRA:RHELPLAN-95126, JIRA:RHELPLAN-92473, JIRA:RHELPLAN-100359, JIRA:RHELPLAN-96640, BZ#2021262, JIRA:RHELPLAN-70122, BZ#2019003, BZ#2011448, BZ#2019318, JIRA:RHELPLAN-101240, JIRA:RHELPLAN-101241, JIRA:RHELPLAN-101242, JIRA:RHELPLAN-101245, JIRA:RHELPLAN-101246, JIRA:RHELPLAN-101247, BZ#2010291, JIRA:RHELPLAN-102552, JIRA:RHELPLAN-65217, BZ#2020529, BZ#1927780, BZ#1935544, BZ#1899167, BZ#1899170, BZ#1979521, JIRA:RHELPLAN-100087, JIRA:RHELPLAN-100639, JIRA:RHELPLAN-10304, BZ#1640697, BZ#1697896, JIRA:RHELPLAN-92523

APPENDIX B. REVISION HISTORY

0.0-1

Wed Nov 03, 2021, Lenka Špačková (lspackova@redhat.com)

- Release of the Red Hat Enterprise Linux 9.0 Beta Release Notes.